

**T.C.
MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**İSTATİSTİK BÖLÜMÜ ÖĞRENCİLERİNİ VE ÖĞRETİM
ELEMENLERİNİ İZLEYEN BİR KARAR DESTEK SİSTEMİ:
BAYKUŞ PROGRAMI**

DOKTORA TEZİ

Semra ERPOLAT

İstatistik Anabilim Dalı

İstatistik Programı

Tez Danışmanı: Prof. Dr. Nalan CİNEMRE

NİSAN 2007

ÖZET

Geçen yüzyılın sonlarına gelindiğinde, geçmişe kıyasla, dünyada her konuda büyük değişimler yaşanmıştır. Bu değişimlerden birisi de veri kaynaklarındaki artıştır. Veri kaynaklarındaki artışa bağlı olarak verilerdeki artış beraberinde “doğru ve gerçekçi verilerden yararlanarak problemlere sistematik bir yaklaşım çerçevesinde çözüm getirme” zorunluluğunu doğurmuştur. Bu bağlamda, çok sayıdaki verinin gereksinimlere göre ayrılıp yararlı bilgiye ulaşılması hemen herkes için sorun olmuştur. Bu sorunun çözümü için önerilen sayısız yöntem vardır. Bu yöntemlerden birisi de karar alıcıların yarı yapılanmış ve yapılanmamış veri ve modellerden yararlanmalarına yardım ederek, onları görevlerinde destekleyen bilgisayar tabanlı etkileşimli bir sistem olan Karar Destek Sistemi'dir.

Bu çalışmada, bilgidен yararlı bilgi edinmeyi sağlayan değişik sistemler üzerinde durulmuş ve bu sistemlerden en çok kullanılanlardan biri olan Karar Destek Sistemleri'ne bir örnek verilmiştir. Çalışma dokuz bölümden oluşmuştur. Birinci Bölümde karar, karar alma süreci, karar kuramı, karar çözümlemesi ve Karar Destek kavramları açıklanmıştır. İkinci Bölümde modelleme ve problem çözümleme süreci kavramları ayrıntılandırılmıştır. Üçüncü Bölümde Bilgi Sistemleri, Dördüncü Bölümde Karar Destek Sistemleri, Beşinci Bölümde Bilgi Tabanlı Karar Destek Sistemleri, Altıncı Bölümde ise Uzman Sistemler ve bilgi edinimi hakkında bilgi verilmiştir. Yedinci Bölümde Karar Destek Sistemleri, Uzman Sistemler, Bilgi Tabanlı Karar Destek Sistemleri'nin nasıl inşa edildikleri ayrıntılı biçimde ele alınmıştır. Sekizinci Bölümde Mimar Sinan Güzel Sanatlar Üniversitesi Fen Edebiyat Fakültesi İstatistik Bölümü öğretim elemanlarına ve bu bölümün lisans, lisansüstü programlarına kayıtlı öğrencilerine çeşitli konularda yardımcı olmak amacıyla geliştirilen ve Baykuş ismi verilen bir Karar Destek Sistemi anlatılmıştır. Son Bölüm sonuç ve önerilere ayrılmıştır.

SUMMARY

In compare with the past there are incredible changes almost in every aspect of the world especially in the number of data sources. The increase of data sources caused the increase in the quantity of data. As a result, “solving problems in a systematic way by using correct and real data” became inevitable big problem. In this context, to obtain knowledge from the categorization of the data with respect to needs became common problem for everybody. Many methods are available to solve this common problem. One of these methods is known as Decision Support System which is computer based and interactive one. This system helps decision makers and support them at their tasks for benefit from semi structured and unstructured data and models.

In this study, different systems are used to obtain knowledge from information are explained briefly, and the well known and widely used of them “Decision Support Systems” is explained in detail and exemplified. This study is composed of nine chapters. In the First Chapter, some important concepts such as decision, decision making process, decision theory, decision analysis, and decision support were explained. In the Second Chapter modelling and problem solving process were investigated. Information Systems were explained in the Third Chapter while Decision Support Systems in the Fourth. The Fifth Chapter is about Knowledge Based Decision Support Systems. In the Sixth Chapter Expert Systems and knowledge acquisition were explained. The construction of Decision Support Systems, Expert Systems, Knowledge Based Decision Support Systems are explained in the Seventh Chapter in detail. In the Eight Chapter Decision Support System which is named as Baykuş was introduced. The aim of Baykuş is to help the instructors of Mimar Sinan Fine Arts University Faculty of Arts and Letters Department of Statistics, and the registered students of the licence and the high licence programs of Statistics. In the Last Chapter results and comments were placed.

ÖNSÖZ

Değerli katkı ve yardımlarından dolayı danışmanım Sayın Prof. Dr. Nalan CİNEMRE'ye, desteğinden dolayı anneme sonsuz teşekkür ederim.

Nisan 2007

Semra ERPOLAT

İÇİNDEKİLER DİZİNİ

ÖZET	i
SUMMARY	ii
ÖNSÖZ	iii
İÇİNDEKİLER DİZİNİ	ivv
ŞEKİL LİSTESİ	ixx
EKRAN LİSTESİ	x
KISALTMALAR LİSTESİ	xii

BÖLÜM 1

KARAR, KARAR ALMA SÜRECİ, KARAR KURAMI KARAR ANALİZİ VE KARAR DESTEK

1.1.Giriş.....	1
1.2. Karar.....	1
1.2.1. Karar ve Karar Alma	1
1.2.2. Karar Çeşitleri	2
1.3. Karar Alma Süreci	5
1.3.1. Zeka Evresi	6
1.3.2. Tasarım Evresi	9
1.3.3. Seçim Evresi	10
1.3.4. Uygulama Evresi	12
1.3.5. Bir KD Sistemi Ortamındaki Geleneksel Karar Süreci	12
1.4. Karar Kuramı	13
1.5. Karar Çözümlemesi.....	13
1.5.1. Karar Çözümlemesi Süreci	13
1.5.2. Karar Çözümlemesi'nin Uygulama Alanları ve Kapsamı	16
1.5.2.1. Genel Uygulama Alanları	16
1.5.2.2. Karar Çözümlemesinin Kapsamı.....	16
1.5.3. Karar Analiziyle İlgili Bilim Dalları ve Metodolojiler	17
1.5.4. Bir Karar Modelini Oluşturan Temel Elemanlar	17
1.5.5. Karar Çözümlemesinde Kullanılan Teknikler	18
1.5.5.1. Tek Amaç Olması Durumunda Kullanılan Teknikler	18
1.5.5.2. Çok Amaç Olması Durumunda Kullanılan Teknikler	18
1.6. Karar Destek	21
1.6.1. KD'in Çeşitli Tanımları	21
1.6.2. KD Nedir?	22
1.6.3. KD'in Sınıflandırılması	22
1.6.4. KD'de Kullanılan Bilim Dalları	22
1.7. Karar Alan Elemanlar	23
1.7.1. Karar Almada Bilgisayarlar	23

İÇİNDEKİLER DİZİNİ (Devam Ediyor)

1.7.2. Karar Almada İnsanlar.....	23
1.7.2.1. İnsanların Karar Alma Süreci.....	23
1.7.2.2. Karar Alma Stilleri	24
1.8. Karar Almada Bilgi Teknolojisi Kullanımı.....	25

BÖLÜM 2

MODELLEME VE PROBLEM ÇÖZÜMLEME SÜRECİ

2.1. Giriş.....	27
2.2. Karar Almada Modelleme Yaklaşımı	27
2.2.1. Grafikselsel Paradigma ve Modelleme	28
2.3. Modellemenin Karakteristikleri ve Yararları	29
2.4. Modellerin Sınıflandırılması.....	30
2.4.1. Yapılarına Göre Model Çeşitleri	30
2.4.2. İçeriklerine Göre Model Çeşitleri	30
2.5. KD Sistemleri'nde Kullanılan Modeller	31
2.6. Matematiksel Model Çeşitleri ve Problemleri.....	32
2.7. Matematiksel Teknikler.....	33
2.8. Matematiksel Modellerin Kullanım Süreci	36
2.9. Problem Çözümleme Süreci	37
2.10. Neyi Ne Zaman Kullanmalı?	39

BÖLÜM 3

BİLGİ SİSTEMLERİ

3.1. Giriş.....	41
3.2. Bilgi Sistemleri Nedir?.....	41
3.3. BS'nin Sınıflandırılması.....	42

BÖLÜM 4

KARAR DESTEK SİSTEMLERİ

4.1. Giriş.....	48
4.2. KD Sistemleri'nin Gelişimi	48
4.3. KD Sistemleri'nin Çeşitli Tanımları	51
4.4. KD Sistemleri'nin Karakteristik ve Yetenekleri.....	53
4.5. KD Sistemleri'nin Yararları	56
4.6. KD Sistemleri'ni Oluşturan Araçlarlar	57
4.7. KD Sistemleri'nin Mimarisi	60
4.7.1. Veri Yönetimi Altsistemi.....	61
4.7.2. Model Yönetimi Altsistemi	63
4.7.2.1. KD Sistemi Modelleri.....	63
4.7.2.2. KD Sistemi'nin Model Yönetim Alt Sistemi	63

İÇİNDEKİLER DİZİNİ (Devam Ediyor)

4.7.2.3. Model Tabanındaki Modellerin Farklı Sınıfları	66
4.7.2.4. KD Sistemi Modellerinin Yedi Grubu.....	69
4.7.3. Diyalog Yönetimi Alt Sistemi veya Kullanıcı Arayüzleri.....	74
4.7.4. Yararlı Bilgi Yönetimi Altsistemi.....	75
4.8. KD Sistemleri'nin Sınıflandırılması	75
4.8.1. Alter'in Çıktı Sınıflandırması	77
4.8.2. Holsapple ve Whinston'un Sınıflandırması.....	80
4.8.3. KD Sistemleri'nin Bilgiye ve Zekaya Dayalı Olarak Sınıflandırılması.....	82
4.8.4. KD Sistemleri'nin Diğer Sınıflandırmaları	86
4.9. KD Sistemleri'nin Sağladıkları Destek.....	87
4.10. KD Sistemleri'nin Organizasyonla Bütünleştirilmesi.....	88
4.11. Organizasyonel Karar Almaya Ayrıntılı Bir Bakış.....	88
4.12. Grup Karar Destek Sistemi.....	90
4.13. KD Sistemleri'nin Kullanım Alanları	91
4.14. Gelecek Kuşak KD Sistemleri	92
4.14.1. KD Sistemleri'nin Gelecekteki Tasarımları	92
4.14.2. KD Sistemleri'nin İçerdikleri Elemanların Gelecekteki Özellikleri.....	93
4.14.3. Bilgi Dünyasından Beklenen Gelişmeler.....	95

BÖLÜM 5

BİLGİ TABANLI KARAR DESTEK SİSTEMLERİ

5.1. Giriş.....	99
5.2. Bilgi Tabanlı KD Sistemleri	99
5.2.1. Bilgi Tabanlı KD Sistemleri'nin Özellikleri.....	101
5.2.2. Bilgi Tabanlı KD Sistemleri Mimarisi	103
5.3. Akıllı Sistem Yaklaşımlarını Gerçekleştirme.....	105

BÖLÜM 6

UZMAN SİSTEMLER VE BİLGİ EDİNİMİ

6.1. Giriş.....	106
6.2. US'in Çeşitli Tanımları	106
6.3. US'i Kavrama	107
6.3.1. Yapay Zeka İle Doğal Zekanın Karşılaştırılması.....	107
6.3.2. Geleneksel Programlar	108
6.3.3. Geleneksel Programlara Sahip US'in Karakteristikleri.....	109
6.4. Uzman Bir Sisteme Ne Zaman İhtiyaç Duyulur?	110
6.5. US'in Avantaj ve Dezavantajları	111
6.5.1. US'in Avantajları	111
6.5.2. US'in Dezavantajları	113
6.6. US'in Mimarisi	114
6.6.1. US'in Yapısı.....	114

İÇİNDEKİLER DİZİNİ (Devam Ediyor)

6.6.2. US'in Bileşenleri	115
6.6.3. US Süreci	122
6.7. US ve Uzman Kabuklar	123
6.7.1. Kabukların Mimarisi	123
6.7.2. Kabuk Kullanmanın Avantajları	124
6.8. Bilgi Edinimi ve Bilgi Tabanının Kurulması	124
6.8.1. İlişkisel Veri Tabanı Sistemleri İle US	125
6.9. US ve İnternet/İntranet	127

BÖLÜM 7

KD SİSTEMLERİ, US, BTKD SİSTEMLERİ İNŞA ETME

7.1. Giriş	128
7.2. KD Sistemleri'ni Kurma	128
7.2.1. KD Sistemleri'ni Kurma Yaklaşımları	128
7.2.2. Kullanıcı	131
7.2.3. Kullanıcı Yapımlı KD Sistemlere Karşı Hazır Yapım Sistemler	131
7.2.4. KD Sistemleri Yazılım Araçları	132
7.2.4.1. KD Sistemi Yazılım Kategorileri	132
7.2.4.2. KD Sistemi Teknoloji Düzeyleri	132
7.2.4.3. KD Sistemleri'nde Kullanılan Programlama Dilleri	137
7.2.4.4. KD Sistemleri Kullanıcı Arayüzleri	137
7.2.5. KD Sistemleri'ndeki Rollerin Değerlendirilmesi	138
7.2.6. Bir KD Sistemi Tasarlarken Düşünülmesi Gereken Faktörler	139
7.2.7. KD Sistemi Geliştirme Platformları	140
7.2.8. KD Sistemleri'ni Geliştirme Aracını Seçme	141
7.2.8.1. Bileşenlerin Biraraya Getirilmesi	143
7.2.8.2. Kaynak Dışı	143
7.2.9. Bir KD Sistemi Jeneratörü Yaratma	143
7.2.10. Klasik Gelişim Döngüsü İle Prototip'in Karşılaştırılması	144
7.2.10.1. KD Sistemleri'nin Klasik Gelişim Döngüsü	144
7.2.10.2. Son Kullanıcılar Tarafından Geliştirilen Kd Sistemleri'nin Gelişim	146
Döngüsü	
7.3. Bir US Kurma	148
7.4. Bir BTKD Sistemi Kurma	149
7.5. KD Sistemleri, US Ve BTKD Sistemleri'nin Geleceği	154

BÖLÜM 8

BAYKUŞ PROGRAMI

8.1. Giriş	155
8.2. Baykuş Programının Amacı	155
8.3. Baykuş Programının Mimarisi	155
8.3.1. Baykuş Programının Sağlayacağı Destek	156
8.3.2. Baykuş Programını Geliştirme Stratejisi	156

İÇİNDEKİLER DİZİNİ (Devam Ediyor)

8.3.3. Baykuş Programını Geliştirme Süreci	156
8.3.4. Baykuş Programının İçerdiği Elemanlar	158
8.4. Baykuş Programının İşleyişi.....	159
8.4.1. Lisans Öğrenci Girişi.....	160
8.4.1.1. Kişisel Bilgiler.....	162
8.4.1.2. Bölüm Bilgileri.....	165
8.4.2. Lisans Üstü Öğrenci Girişi	177
8.4.3. Öğretim Elemanı Girişi	177
8.4.3.1. Kişisel Bilgiler Düğmesi.....	178
8.4.3.2. Lisans Öğrencileri Ayırımı	179
8.4.3.3. Lisans Üstü Öğrencileri Ayırımı	205
8.4.4. Bölüm Başkanı Girişi	206
8.4.4.1. Öğrenciler Düğmesi.....	207
8.4.4.1.1. Lisans Öğrencileri Ayırımı:.....	207
8.4.4.1.2. Lisans Üstü Öğrencileri Ayırımı	223
8.4.4.2. Dersler Düğmesi	223
8.4.4.2.1. Dönem Dersleri Düğmesi.....	224
8.4.4.2.2. Ara Düğmesi.....	225
8.4.4.2.3. Yeni Ayırımı.....	226
8.4.4.2.4. Listeler Ayırımı	227
8.4.4.2.5. Bilgi Yenileme Ayırımı	228
8.4.4.3. Öğretim Elemanı Düğmesi.....	229
8.4.4.3.1. Yeni Öğretim Elemanı Girişi Düğmesi.....	230
8.4.4.3.2. Kişisel Bilgiler Düğmesi	231
8.4.5. Çıkış Düğmesi.....	236

BÖLÜM 9

SONUÇLAR VE TARTIŞMA.....	239
KAYNAKLAR.....	241
EKLER DİZİNİ	246
ÖZGEÇMİŞ.....	249

ŞEKİL LİSTESİ

Şekil 1.1. Karar alma /modelleme süreci (Turban, 2005).....	6
Şekil 1.2. Planlama ve kontrol için bir yararlı bilgi modeli (Finlay, 1994).....	8
Şekil 1.3. Bir KD Sistemi'nin karar alma süreci (Shim ve diğ., 2002).....	12
Şekil 1.4. Karar çözümlemesi süreci (Howard, 1988).....	14
Şekil 1.5. Karar temelinin ortaya çıkarılması ve değerlendirilmesi (Howard, 1988).....	15
Şekil 1.6. Karar almak için bilgisayarlaştırılmış destek (YDS) (Turban ve diğ. 2001).....	26
Şekil 2.1. Matematiksel modeller ve kullanım alanları (Davis, 1988).....	32
Şekil 2.2. Nicel destek için araçlar (Davis, 1988).....	34
Şekil 2.3. Bir problemi ölçen adımlar (Davis, 1988).....	36
Şekil 3.1. Bir bilgi sisteminin şematik görüntüsü (Turban ve diğ., 1996).....	42
Şekil 4.1. KD Sistemleri'nin ideal karakteristikleri ve yetenekleri (Turban, 1995).....	54
Şekil 4.2. Modern geniş kapsamlı bir KD Sistemi.....	59
Şekil 4.3. KD Sistemleri'nin kavramsal modeli (Turban, 1995).....	61
Şekil 4.4. Matematiksel modellerin "Aile Ağacı" (Mallach, 2000).....	66
Şekil 4.5. KD Sistemleri arasındaki ilişkiler (Finlay, 1994).....	82
Şekil 4.6. KD Sistemi türleri (Finlay, 1994).....	84
Şekil 4.7. KD Sistemleri tarafından sağlanan destek (Turban, 1995).....	87
Şekil 4.8. KD Sistemleri için yeni bir karar paradigması (Courtney, 2001).....	89
Şekil 4.9. GKD Sistemleri'nin düzeni (Stair, 1992).....	90
Şekil 5.1. Genel bir KD Sistemi'nin veri akış diyagramı (Mallach, 1994).....	99
Şekil 5.2. BTKD Sistemleri uygulamasının fonksiyonel çözümlemesi.....	102
(Klein ve Methlie, 1995). (BT: Bilgi Tabanı)	
Şekil 5.3. Kullanıcı ile BTKD Sistemleri kaynakları arasındaki tipik etkileşim	103
(Klein ve Methlie, 1995)	
Şekil 5.4. BTKD Sistemleri mimarisi (Turban, 1995).....	104
Şekil 5.5. Kaynaklar ve uygulama dosyası arasındaki bilgi transferi.....	104
(Klein ve Methlie, 1995)	
Şekil 6.1. US'in yapısı (Turban ve diğ., 2001).....	115
Şekil 6.2. İdeal bir uzman sistem kabuk mimarisi (Beynon-Davies, 1991).....	124
Şekil 7.1. KD Sistemleri'nin gelişimi ve adaptasyonu (Zwass, 1998).....	131
Şekil 7.2. KD Sistemleri'nin teknolojik düzeyleri (Turban ve diğ., 1996).....	133
Şekil 7.3. KD Sistemleri'nin üç teknoloji düzeyi ile yönetici ve teknisyenlerin beş	139
farklı rolü (Sprague, 1980)	

EKRAN LİSTESİ

Ekran 8.1. Baykuş programının tanıtım ekranı.	159
Ekran 8.2. Telif hakkı ekranı.	159
Ekran 8.3. Baykuş programının ana menüsü.	160
Ekran 8.4. Öğrenci Sign On ekranı.	161
Ekran 8.5. “Lisans Öğrenci Main” ekranı.	162
Ekran 8.6. “Kişisel Bilgiler Öğrenci” ekranı.	163
Ekran 8.7. “Özgeçmiş” sekmesi.	163
Ekran 8.8. “Akademik Bilgiler” sekmesi.	164
Ekran 8.9. “Sınıf” sekmesi.	164
Ekran 8.10. “Dersler” sekmesi.	164
Ekran 8.11. “Kullanıcı Girişi” sekmesi.	165
Ekran 8.12. “Dönem Seçimi” ekranı.	165
Ekran 8.13. “Bölüm Bilgileri” ekranı.	166
Ekran 8.14. “Ders Seçimi” ekranı.	166
Ekran 8.15. “Zorunlu” sekmesi.	168
Ekran 8.16. “Seçilen Dersler” ekranı.	170
Ekran 8.17. “Alınan Dersler” ekranı.	172
Ekran 8.18. Not girişlerinin yapılması sonunda “Alınan Dersler” ekranı.	172
Ekran 8.19. “Sınav Bilgileri” ekranı.	176
Ekran 8.20. “Öğretim Elemanı Main” ekranı.	178
Ekran 8.21. “Lisans Öğrencileri Listesi Danışmanlık” ekranı.	180
Ekran 8.22. “Kişisel Bilgiler” ekranında yeni ders ekleme işlemi.	182
Ekran 8.23. “Bölüm Bilgileri” ekranı.	183
Ekran 8.24. “Ders Seçimi” ekranı.	184
Ekran 8.25. “Ders Seçimi” ekranında “Onay” işlemi.	185
Ekran 8.26. “Lisans Öğrenci İstatistikleri” ekranı.	187
Ekran 8.27. “Lisans Öğrenci Arama” ekranı.	191
Ekran 8.28. “Ders Listesi” ekranı.	191
Ekran 8.29. “Öğrenci Listesi” ekranı.	192
Ekran 8.30. “Öğrenci Listesi” raporu.	193
Ekran 8.31. “Sınav Listesi” ekranı.	193
Ekran 8.32. Not hesaplaması yapıldıktan sonraki “Sınav Listesi” ekranı.	205
Ekran 8.33. “Sınav Listesi” raporu.	205
Ekran 8.34. “Bölüm Başkanı Main” ekranı.	206
Ekran 8.35. “Lisans ve Lisans Üstü Öğrencileri Main” ekranı.	207
Ekran 8.36. “Lisans Öğrencileri” ekranı.	208
Ekran 8.37. “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu” ekranı	209
Ekran 8.38. “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu Düzenle” ekranı	210
Ekran 8.39. Kayıt gezinti düğmeleri.	211
Ekran 8.41. “Yeni Lisans Öğrenci Girişi” ekranı.	213
Ekran 8.42. “Kişisel Bilgiler” sekmesi ve takvimin görünüşü.	214
Ekran 8.43. “Kişisel Bilgiler” sekmesine fotoğraf ekleyen ekran.	215
Ekran 8.45. “Özgeçmiş” sekmesi.	216

EKRAN LİSTESİ (Devam Ediyor)

Ekran 8.46. “Akademik Bilgiler” sekmesi.....	216
Ekran 8.47. “Akademik Bilgiler” sekmesi.....	217
Ekran 8.48. “Dersler” sekmesi.....	217
Ekran 8.49. “Kullanıcı Girişi” sekmesi.....	218
Ekran 8.50. “Öğrenci Raporları” print ekranı.....	220
Ekran 8.51. “Kişisel Bilgiler” raporu.....	221
Ekran 8.52. “Akademik Bilgiler” raporu.....	222
Ekran 8.53. “Akademik Bilgiler” raporu.....	222
Ekran 8.54. “Lisans ve Lisans Üstü Dersleri Main” ekranı.....	223
Ekran 8.55. “Ders Listesi” ekranı.....	224
Ekran 8.57. “Ders Ekle” ekranı.....	226
Ekran 8.59. “Ders Düzenle” ekranı.....	228
Ekran 8.60. “Ders İçeriği Print” ekranı.....	229
Ekran 8.61. “Ders İçeriği Raporu” ekranı.....	229
Ekran 8.62. “Öğretim Elemanları” ekranı.....	230
Ekran 8.63. “Öğretim Elemanının Kişisel Bilgileri” ekranı.....	231
Ekran 8.64. “Lisans Desleri” sekmesi.....	232
Ekran 8.65. “Kullanıcı Girişi” sekmesi.....	233
Ekran 8.66. “Öğretim Elemanı Raporları” print ekranı.....	236

KISALTMALAR LİSTESİ

BS	: Bilgi Sistemleri
BT	: Bilgi Teknolojisi
BTBS	: Bilgisayar Tabanlı Bilgi Sistemleri
BTKD Sistemleri:	Bilgi Tabanlı Karar Destek Sistemleri
DDS	: Doğal Dil Süreci
DZ	: Doğal Zeka
GD Sistemleri	: Grup Destek Sistemleri
GKD	: Grup Karar Destek
GKD Sistemleri	: Grup Karar Destek Sistemleri
İİS	: İş İşlemleri Sistemi
İVT	: İlişkisel Veri Tabanı
İVTS	: İlişkisel Veri Tabanı Sistemleri
İVTYS	: İlişkisel Veri Tabanı Yönetim Sistemleri
KD	: Karar Destek
KD Sistemleri	: Karar Destek Sistemleri
OOS	: Ofis Otomasyon Sistemleri
US	: Uzman Sistemler
UBS	: Uzman Bilgi Sistemleri
VT	: Veri Tabanı
VTYS	: Veri Tabanı Yönetim Sistemi
YBİS	: Yönetici Bilgi Sistemleri

KISALTMALAR LİSTESİ (Devam Ediyor)

YBS : Yönetim Bilgi Sistemleri

YZ : Yapay Zeka

YZS : Yönetim Zeka Sistemleri

YZ : Yapay Zeka

BÖLÜM 1

KARAR, KARAR ALMA SÜRECİ, KARAR KURAMI KARAR ÇÖZÜMLEMESİ VE KARAR DESTEK

1.1.GİRİŞ

Bu bölümde karar, karar alma süreci, karar kuramı, karar çözümlemesi ve karar destek gibi önemli kavramlar üzerinde durulacaktır.

1.2. KARAR

1.2.1. KARAR VE KARAR ALMA

Karar, herhangi bir sorunu çözüme ulaştırmak amacıyla edinilen bilgilerin, uygun karar modellerinde kullanılmasıyla oluşturulan alternatif çözümlerden en iyi olanının seçilmesi ve yürütülmesi süreci olarak tanımlanabilir.

Karar alma sadece seçilmesi düşünülen alternatiflerin belirlenmesi ve seçilmesi değil aynı zamanda karar alıcının hedeflerinin, amaçlarının, isteklerinin, değerlerinin ve benzer yargılarının gerçekleşmesini sağlayacak en uygun alternatifin seçilmesi anlamına da gelmektedir.

Kuruüzüm (1998)'e göre karar veya karar alma,

- Bir süreçtir.
- Bir problem çözme işlemidir.
- Amaca uygun değişik davranış biçimlerinden birini seçme işlemidir.

Karar, alternatifler arasından uygun bir seçimdir (Mallach, 1994; Simon, 1960). Karar alma ise, bir hedef ya da hedeflere ulaşmak için gerçekleştirilmesi gereken hareket yönünü alternatifler arasından seçme süreci olarak tanımlanabilir (Turban, 1995).

Mevcut bir problemle ilgili karar almanın, problemin çözümlenmesi anlamına gelmediği önemli bir husustur. Problem çözüme, sistemin içinde bulunduğu gerçek durum ile olması istenen durum arasındaki farklılığı, kapatmaya çalışma sürecidir.

Çok sayıdaki alternatifler arasından bir seçim yapma sürecinin tümü olarak tanımlanan karar alma aşağıda sıralanan ifadeleri içerir (Bohanec, 2006).

- Problemi değerlendirme,
- Bilgi toplama ve bu bilgilerin doğruluğunu kanıtlama,
- Alternatifleri tanımlama,
- Alınacak kararların sonuçlarını tasarlama,
- Mevcut bilgiye dayalı olarak güvenilir ve mantıksal yargı kullanımının seçimini yapma,
- Diğer açıklama ve kararlardan haberdar olma,
- Kararları değerlendirme.

1.2.2. KARAR ÇEŞİTLERİ

Yöneticilerin üstlendikleri görevlerden bir tanesi de kararları, yeterince bilgi edindikten sonra almaktır (Turban ve diğ., 2001). Kararları sınıflandırmanın birçok yolu vardır. Kararlar genellikle sahip oldukları ortak karakteristiklere göre sınıflandırılırlar. Anthony (1965), yönetsel faaliyetleri; stratejik planlama, yönetsel kontrol ve işlemsel kontrol kategorilerine ayırmıştır. Bu ayrım aynı zamanda karar almanın faaliyet alanlarını da oluşturmaktadır (Mallach, 1994). Garry ve Scot Morton'a göre kararlar, yapı ve kapsam bakımından olmak üzere iki ana boyuta ve bunların da her biri kendi aralarında üç alt boyuta ayrılmaktadır (Mallach, 2000). Yapılarına göre kararların alt boyutları yapısal, yarı yapısal, yapısal olmayan şeklindedir. Kapsama göre alt boyutlar ise işlemsel, taktik ve stratejiktir.

Karar türü olarak nitelendirilen alınan kararın yapısı ile faaliyet alanı olarak nitelendirilen kararın kapsamı birbirleriyle ilişkili boyutlardır. Genellikle işlemsel kararlar daha yapısal iken stratejik kararlar daha az yapısal olma eğilimindedir.

İki ana boyutun (yapı, kapsam) her birinin üç alt boyuta ayrılması ve bu alt boyutların birbirleriyle kombinasyonları sonucunda Çizelge 1.1’de gösterilen dokuz çeşit karar elde edilir. Turban ve diğ. (2001), karar desteğe ihtiyaç duyulan alanları belirlemek üzere Çizelge 1.1’de sunulan iskeleti geliştirmiştir. Bu iskelet, ilk olarak Gorry ve Scott-Morton (1971) tarafından önerilmiş olup Turban ve diğ. (2001) iskeleti oluştururken, Simon (1960; 1977)’nin programlı olandan programlı olmayana veya iyi yapısallaşmıştan az yapısallaşmışa doğru uzanan karar türleri ile Anthony (1965)’nin stratejik, yönetsel ve operasyonel kontrol biçiminde adlandırdığı yönetsel aktivitelerinin en geniş üç kategorisini kullanmıştır. İskelette, çeşitli karar türlerini ve yönetsel kontrolü destekleyen teknolojileri göstermiştir.

Çizelge 1.1. Karar destek iskeleti (Turban, 1995).

KARAR	KONTROL ÇEŞİDİ			
	İşlemsel	Taktik	Stratejik	Gereksinilen Destek
Yapısal	1	2	3	Yönetim Bilgi Sistemleri (YBS) İşlemleri, Araştırma Modelleri, İş İşlemleri
	Anlaşılabilir hesaplamalar, emir girişi	Bütçe çözümlenmeleri, kısa dönem tahminler, personel raporları yapma/satın alma	Finansal yönetim (yatırım), ambar yerini saptama, dağıtım sistemleri	
Yarı Yapısal	4	5	6	KD Sistemleri
	Üretim planlama, envanter kontrol	Kredi değerlendirme, bütçe hazırlama, fabrika kurma, proje planlama	Yeni fabrika kurma, şirketlerin birleşmesi ve eklenmesi, yeni ürün planlama, hayat sigortası planlama	
Yapısal Olmayan	7	8	9	KD Sistemleri, Uzman Sistemler (US), Sinir Ağları
	Bir depo için koruma seçimi, yazılım satın alma, faiz karşılığında ödünç para vermeyi onaylama	Ciro işlemleri, bir uygulamayı düzenleme, donanım satın alma	Yeni teknoloji geliştirme, sosyal sorumlulukları planlama	
Gereksinilen Destek	YBS, Yönetim Bilimi	Yönetim Bilimi, KD Sistemleri, US, Uzman Bilgi Sistemleri (UBS)	UBS, US, Sinir Ağları	

Belirlenen amaçları gerçekleştirecek bir Karar Destek Sistemi (KD Sistemi) planlanırken, eğer bu KD Sistemi’ni kullanarak alınacak kararların hangi karar sınıflarına girdikleri belirlenebiliyorsa, geçmişte bu çeşit kararlar için kullanılmış Karar Destek Sistemleri’nden (KD Sistemleri) iyi sonuçlar alınmış olanlarını

incelemek mümkün olabilir ve mevcut problemi çözmek için buna benzer bir KD Sistemi oluşturulabilir (Mallach, 2000).

Yapısına Göre Karar Çeşitleri: Yukarıda açıkladığımız gibi kararları, problemin yapısına göre yapısal, yarı yapısal ve yapısal olmayan şeklinde sınıflandırmak mümkündür (Mallach, 2000). Yapısal kararlar, tekrarlamalı ve rutindir. Kolaylıkla çözülebilir ve standart bir model ile temsil edilebilir. Genellikle iyi tanımlanmış kararlar yapısal kararlardır. Yapısal kararlar, programlama algoritmalarına benzedikleri için kolaylıkla bir bilgisayar programına dönüştürülebilir. Yapısal kararlarda karar sürecinin zeka, tasarım ve seçim evresi yapısal olup yapısal karar evresi olarak adlandırılır. Yarı yapısal kararlar, tamamen olmasa da yapısal yönleri içeren problemlerdir. Organizasyonların karşı karşıya oldukları kararların birçoğu bu tiptedir (Mallach, 1994). Yarı yapısal kararlarda karar sürecinin zeka, tasarım ve seçim evrelerinden en az biri yapısal olmayan niteliktedir. Yapısal olmayan kararlarda zeka, tasarım ve seçim evrelerinin tümü yapısal değildir. Bu tür kararlar genellikle yeni, alışılmamış ve çözümleri zor kararlardır (Shim ve diğ., 2002).

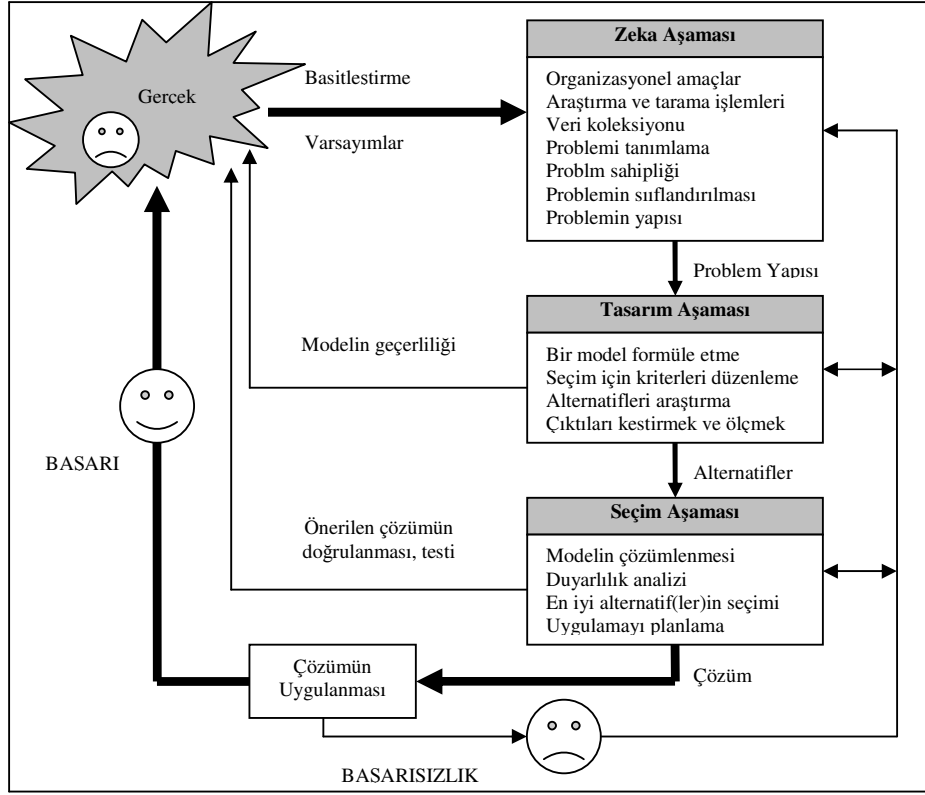
Doğasına Göre Karar Çeşitleri: Anthony (1965)'e göre yönetsel faaliyetler; işlemsel kontrol, taktik veya yönetsel kontrol ve stratejik planlama olmak üzere üç kategoriye ayrılmaktadır. Mallach (1994) ise bu üç kategoriye karar alma sürecinin faaliyet alanları olarak kabul etmiştir. İşlemsel kararlar, içinde bulunulan zamanda organizasyonlarda oluşan aktiviteleri etkileyen kararlardır. Daha önce stratejik ve taktik kararlarla görevleri, kaynakları ve hedefleri belirlenen organizasyonların, belirlenen günlük işlemleri (Finlay, 1994) veya özel görevleri (Antony, 1965) işlemsel kararlar ile gerçekleştirilir. İşlemsel kararlar rutin olarak alınan kararlar olup sonuçları organizasyonların geleceğini etkilemez. Alt düzey yöneticiler veya yönetici olmayan çalışanlar tarafından alınan kararlar işlemsel kararlardır. Taktik veya yönetsel kararlar, organizasyonun sınırlı bir süre içinde nasıl çalışması gerektiğini düzenleyen kararlardır. Taktik kararlar, işlemsel aktivitelerden ziyade daha uzun zaman mesafesi olan aktiviteleri ilgilendirmektedir. Bu kararlar, içinde bulunulan anda organizasyonların sahip oldukları kaynakları en etkin biçimde kullanabilmelerini sağlamaktadır (Antony, 1965; Finlay, 1994). Genellikle orta düzey yöneticiler tarafından alınan kararlardır. Stratejik kararlar, uzun bir zaman periyodu içinde organizasyonun tümünü veya büyük bir kısmını etkileyen kararlardır

(Antony, 1965; Mallach, 2000). Bu tip kararlar organizasyonların amaçlarını ve politikalarını etkilemektedir. Genellikle üst düzey yöneticiler tarafından alınır. Stratejik kararlar, belirlenen hedeflere ulaşmak için ihtiyaç duyulan kaynakların düzeyleriyle ilgilenmekte ve organizasyon ile çevresi arasındaki uzun süreli ilişkileri içermektedir (Finlay, 1994).

1.3. KARAR ALMA SÜRECİ

Bütün kararların ilk defa Herbet Simon tarafından tanımlanmış olan zeka, tasarım, seçim ve uygulama evresi vardır (Turban ve diğ., 2001). Simon (1960)'a göre karar alma bu dört evreden oluşan ve tekrar incelenebilme olanağına sahip bir modeldir. Bu model, yani karar alma/modelleme süreci Şekil 1.1.'de gösterilmiştir.

Karar almanın zeka evresi problemler için araştırma yapıldığı sırada, tasarım evresi alternatifleri geliştirme sırasında, seçim evresi ise alternatifleri çözümlenme ve uygulama evresi için bunlardan birini seçme sırasında devreye girer (Shim ve diğ., 2002). Zeka, tasarım ve seçim evreleri arasında sürekli olarak aktivitelerin akışı söz konusudur. Burada önemli olan nokta, herhangi bir evredeyken bir önceki evreye geri dönülebilme imkanının olabilmesidir. Her bir evreye verilen önem ve evreler arasındaki ilişki; her bir karar alıcının farklı bir evreyi önemseyebileceği, karar alma evreleri üzerinde harcanacak zamanın karar alınması gereken problemin çeşidine göre az veya çok olabilmesi ve evreler arasında ileriye veya geriye doğru gidişlerin mümkün olabilmesi gibi nedenlerden dolayı genellikle bir karardan diğerine farklılık göstermektedir.



Şekil 1.1. Karar alma /modelleme süreci (Turban, 2005).

Finlay (1994) karar alma süreci için, Simon (1960)'nun modelinin biraz değiştirilmiş ve genişletilmiş hali olan bir modeli önermektedir. Bu model yapısalılık, anlama ve yedi aşamalı hareketten oluşan üç evreden oluşmaktadır. Bu yedi aşama arasında geri dönüşler ve ileri gidişler mümkündür. Finlay (1994)'in yaklaşımı Çizelge 1.2.'de ayrıntılandırılmıştır.

Çizelge 1.2. Finlay (1994)'in karar alma süreci.

EVRELER	AŞAMALAR
Yapısallaşma	Problemi bulma Problemi tanımlama
Anlama	Detaylandırılmış sistemlerin tasarımı Hareket konularını inceleme Karar verme
Hareket	Değişim uygulamaları Tekrar gözden geçirme

1.3.1. ZEKA EVRESİ

Problem belirleme ve tanımlama olarak da bilinen zeka evresinde, karara olan ihtiyaç belirlenir. Simon (1960)'a göre zeka evresi, karar alınmasını gerektiren koşullar için çevre araştırması iken, Finlay (1994)'e göre problemlerin belirlenmesi için yapılan

bir arařtırmadır. Finlay (1994)'in karar alma süreci modelinde zeka evresi, problem bulma aşaması olarak adlandırılmıştır.

Karar alıcının, öncelikle karar probleminin neye hüküm vereceğini veya neyi formüle edeceğini kararlařtırması gerekir. Bu formülasyon Mallach (1994) tarafından bir karar durumu veya problem durumu olarak adlandırılırken, Finlay (1994) tarafından problem tanımlama aşaması olarak adlandırılmaktadır. İlk kez tanımlanacak olan problemin amacının ve kısıtlarının belirlenmiş olması gerekir (Finlay, 1994).

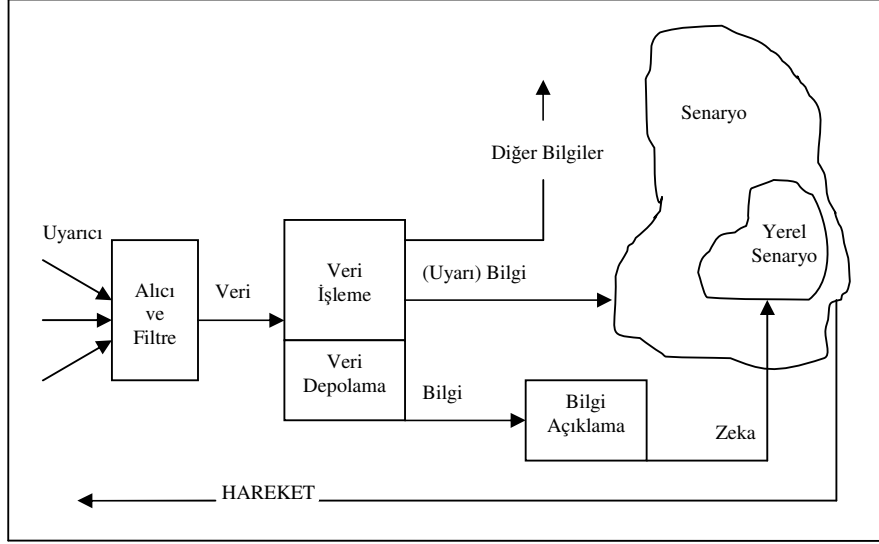
Karar ifadesi ile kararlařtırılmak istenen şey belirtilir. Akıllı karar almada karar ifadesinin açık bir şekilde belirtilmesinin önemi büyüktür. Çünkü karar ifadesi, üzerinde çalışılan konunun önemsiz yanlarını önemli yanlarından ayırarak, karar alıcının düşüncesini kolaylıkla esas konu üzerine odaklamasına yardımcı olur. KD Sistemi geliştirme projeleri, önerilen sistemin yardımıyla alınmış olan kararların kolay bir şekilde anlaşılmasını sağladıktan sonra başlamalıdır (Turban, 1993).

Zeka evresindeki “zeka” terimi, kararın sonucunu bilmeden bilgiyi toplama veya biraraya getirme anlamına gelmektedir (Mallach, 1994). Zeka evresi, içinde modelleme sürecinin de başladığı evredir. Turban ve diğ. (2001)'ne göre bu evre, gerçeğin belirlendiği, problemin teşhis edildiği ve sınındığı evredir. Mallach (2000)'a göre bu evre bir karar gerektiren problem veya durumun bulunması, tanımlanması ve formüle edilmesidir. Zeka evresi boyunca problemin sınıflandırılması, problemin ayrıştırılması ve problem sahipliğinin belirlenmesi işlemlerinin gerçekleştirilmesi gerekebilir (Turban, 1993). Zeka evresinin sonunda karar ifadesine ulaşılır.

Problemin Sınıflandırılması: Problemin sınıflandırılmasında, kararın alanı ve kararın yapılaşdırılma derecesi önemli rol oynayan iki unsurdur. Mallach (1994) problemleri kararın yapılaşdırılma derecesine göre “yapısal, yapısal olmayan” şeklinde sınıflandırılırken, Simon (1960) “programlı, programlı olmayan” şeklinde sınıflandırmıştır.

Bilgi, zeka için hammaddedir. Yöneticiler veya karar alıcılar, ele aldıkları problemin durum veya senaryo düzenini açıklamada, bilgiye ihtiyaç duyabilirlerler (Finlay, 1994). Veri, bilgi, zeka ve bunların biçimlendirildiği sürecin tümü yararlı bilginin (knowledge) görünüşlerini oluşturmaktadır. Finlay (1994), karar alıcının kararlarını

daha iyi bir pozisyonda planlamasını ve kontrol etmesini sağlayacak şekilde karar alıcının zekasını kullanan bir yararlı bilgi modeli önermiştir. Bu model Şekil 1.2.'de gösterildiği gibidir.



Şekil 1.2. Planlama ve kontrol için bir yararlı bilgi modeli (Finlay, 1994).

Senaryo, alınan kararın sonucunu en kötü durum, en iyi durum, en olası durum, vb. gibi varılması olası bir sonuç olduğunu varsayıp değerlendirir. Böylece seçim evresinde kullanılmış olan kriterlerin değerlendirilmesi büyük ölçüde gerçekleştirilmiş olur (Turban, 1993). Finlay (1994)'in önermiş olduğu bu model sadece bireysel ya da grup halinde planlama ve karar alma gerektiren sistemlerde değil aynı zamanda herhangi bir bilişsel sistem türünde de uygulanabilir.

Üstlendikleri görev itibarıyla yöneticiler sürekli olarak bilgiye gereksinim duyar. Finlay (1994), farklı düzeylerdeki problem veya karar alanları için farklı bilgi gereksinimlerinin önemli olabileceğini savunmaktadır. Karar alanının farklı düzeylerindeki farklı gereksinim karakteristikleri Çizelge 1.3'de gösterilmektedir. Bu çizelgede boyut olarak sıralanan ifadeler kişiden kişiye farklılık gösterebilir. Gereksinim duyulan bilgiye kolaylıkla ulaşılması bilgi kaynağının objektif olduğunu gösterir. Genellikle, zor veri (hard data) olarak adlandırılan veri türü nicel ve yazılı iken kolay veri (soft data) olarak adlandırılan veri nitel ve sözlü olma eğilimindedir.

Çizelge 1.3. Bilgi gereksinimi düzeyleri (Finlay, 1994).

BOYUT	İŞLEMSEL	TAKTİK	STRATEJİK
Problem türü	Yapısal	-----➔	III-yapısal
Zaman çerçevesi	Şimdiki geçmiş	-----➔	Gelecek
Kaynak	Büyük ölçüde dahili	-----➔	Büyük ölçüde harici
Organizasyon	Sıkışık	-----➔	Gevşek
Alan	Ayrıntılı	-----➔	Geniş-alanlı
Yaş	Şimdi (current)	-----➔	Eski
Zorluk	Zor	-----➔	Hafif
Kesinlik	Tam ve kesin	-----➔	Tam
Beklenti	Emredilmiş	-----➔	Süpriz
Kullanım sıklığı	Sık	-----➔	Seyrek

Problem Ayırıştırma ve Sahiplik: Anlaşılması ve çözümü zor olan problemler, anlaşılması ve çözümü daha kolay olan basit alt problemlere ayırıştırılarak çözülebilir. Problemin sahipliğinin belirlenmesi ise problemin organizasyon içinde çözülebilir olup olmadığının veya problemin kontrol edilemeyen bir faktörü olup olmadığının belirlenmesinde önem taşımaktadır (Turban, 1993).

Problem İfadesi: Zeka evresindeki problemin sınıflandırılması, problem ayırıştırma ve sahiplik alt aşamalarının gerçekleştirilmesi sonucunda ulaşılan son noktadır (Mallach, 1994).

1.3.2. TASARIM EVRESİ

Simon (1960) tasarım evresini, mümkün hareket yönlerini bulma, geliştirme ve çözümlenme olarak açıklamıştır. Turban ve diğ. (2001) ise bu evreyi, modelin oluşturulup onaylandığı ve daha önce belirlenmiş olan potansiyel çözümlerin değerlendirilebilmesi için kriterin düzenlendiği yer olarak tanımlamıştır. Karar alma kriterini düzenleme, karar alıcılara herhangi bir kararı optimize etmelerinde yardımcı ve faydalı bir işlemdir. Karar alıcılar çoğunlukla kendi karar alma kriterlerini tam olarak tanımlayamayabilirler. Kriter açık bir şekilde tanımlanmasa bile yine de vardır ve bu evrenin sonunda kriterin açık bir şekilde oluşturulması gerekir.

Tasarım evresinde mevcut alternatifler veya elde edilebilir tercihler belirlenirken büyük çapta bir araştırmaya gereksinim duyulabilir (Mallach, 1994). Mallach (1994)'a göre alternatiflerin seçiminde yaratıcılık önemli rol oynamaktadır. Bu evre,

alternatif sonuçların araştırılmasında modellerin kullanımını gerektiren evredir (Finlay, 1994). Bu evrede problemin durumu modellenir, model test edilir ve onaylanır. Modelleme, problemin kavramsallığını ve nitel ve/veya nicel şekilde soyutlanmasını gerektirir, yani modeller gerçeği basitleştirmek için kullanılır. Matematiksel bir model oluşturulmak istenmesi durumunda, modele ilişkin bağımlı ve bağımsız değişkenlerin tanımlanması ve bunlar arasındaki ilişkilerin eşitlikler şeklinde oluşturulması gerekir (Turban, 1993).

1.3.3. SEÇİM EVRESİ

Seçim evresi, probleme ilişkin bütün alternatiflerin araştırıldığı, değerlendirildiği ve bunlardan uygun bulunan bir tanesinin çözüm olarak seçildiği evredir (Simon, 1960). Kararın uygulanabilmesi için öncelikle önerilen çözümün “kağıt üzerinde” test edilip akla uygun olduğunun görülmesi gerekir (Turban ve diğ., 2001). Alternatiflerin belirlenmesi ve değerlendirilmesi bakımından genellikle seçim evresi ile tasarım evresi arasındaki sınır çok belirgin olmayabilmektedir.

Gerçek problemi çözecek uygun bir hareket tarzının seçimi, kriter kümesine bağlı olan birçok yaklaşımı içerebilir. Problemin çözümünü gerçekleştirecek modellerden normatif olanlar ya optimal çözüm veren analitik modeli ya da geniş kapsamlı ve ayrıntılı bir liste modelini, analitik yöntemler ise yapısal modellerde optimum bir çözüm üreten veya kesin bir sonuç kestirimi yapan matematiksel formülasyonları kullanır (Mallach, 1994).

Araştırma: Araştırma, problemlere çözüm bulmak, doğruyu ortaya koymak, bilinmeyen öğrenmek amacıyla gerçekleştirilen çalışmalar olarak tanımlanabilir. Araştırmanın etkinliğini artırmanın yollarından biri algoritma kullanmaktır. Algoritma kullanılarak karar almak için gerçekleştirilecek işlemlerin gözlemlenmesi ve tasarlanması daha kolay olmaktadır. Araştırma, bir hedef ve izlenmesi gereken adımların belirtilmesiyle yürütülebilir. “Kör araştırma” ve “deneme-yanılma yöntemiyle araştırma” olmak üzere iki tür araştırma vardır (Turban, 1993).

Kör Araştırma: Kör araştırma, isteğe bağlı olup yol gösterici nitelikte değildir. Tamsayım, tamsayım olmayan olmak üzere iki tür kör araştırma vardır.

Tamsayım durumunda, probleme ilişkin bütün alternatifler gözönüne alınır. Tamsayım olmayan kör arařtırmada ise işlemler yeterince iyi olan bir çözüme ulařılıncaya kadar tekrarlanır.

Kör arařtırma, çözüm için birçok düğümün kontrol edilmesini gerektirdiğinden, büyük problemler için pratik değıldir. Bununla birlikte zaman ve bilgisayar belleğı miktarı üzerine koyduğı sınırlama yüzünden uygulanması kolay bir yöntemdir.

Deneme-Yanılma Yöntemiyle Arařtırma: Deneme-yanılma yöntemiyle arařtırma, bir problemin nasıl çözüleceğini içeren karar kurallarından oluşan bir bütündür (Turban, 1993). Problemin çözümüne ilişkin yaklaşık sonuca genellikle deneme-yanılma yoluyla ulařılır. Deneme-yanılma yöntemli arařtırmada tatmin edici bir çözüme ulařılıncaya kadar işlemler adım adım izlenir. Bu arařtırma kör arařtırmaya oranla daha ucuz ve daha hızlıdır.

Değerlendirme: Bir problem için önerilecek çözüme ulaşmadaki son adımdır. Bu adım; deneme-yanılma, “what-if” çözümlenmeleri ve hedef arařtırma gibi duyarlılık çözümlenmeleri ve çoklu amaçlar ile gerçekleştirilebilir.

Çoklu Amaçlar: Çoklu amaçların değerlendirilmesinde, genellikle her bir alternatifin ve bunun birkaç amaç üzerindeki potansiyel etkisinin çözümlenmesi gerekir. Çok amaçlı karar alma problemlerinin çözümünde genellikle bilgisayarlařtırılmış modeller kullanılır.

Duyarlılık Çözümlenmeleri: Bilginin doğruluğı veya görelî önemi hakkında emin olunamadığı durumlarda karar alıcılara yardım etmede kullanılan bir tekniktir. Duyarlılık çözümlenmeleri genellikle doğrusal programlama gibi standart nicel modellerle gerçekleştirilir. Duyarlılık çözümlenmeleri, aralıkları belirleme ve sınırların düzenlenme işlemlerini oldukça hızlı gerçekleştirebildiğinden etkili bir yöntemdir. Deneme-yanılma, “what-if” çözümlenmeleri ve hedef arařtırma gibi duyarlılık çözümlenmeleri de vardır.

Deneme-yanılma, çok daha iyi bir çözüm elde edebilmek için girdileri defalarca değıştirmeyi içerir. Bu deneysellik, “what-if” ve hedef arama olarak meydana çıkar (Turban, 1993). “What-if” senaryosunda sonuçlar, girdi verisine ve belirsiz olan geleceğı değerlendiren modele bağılıdır. Uygun bir kullanıcı arayüzü ile girdi

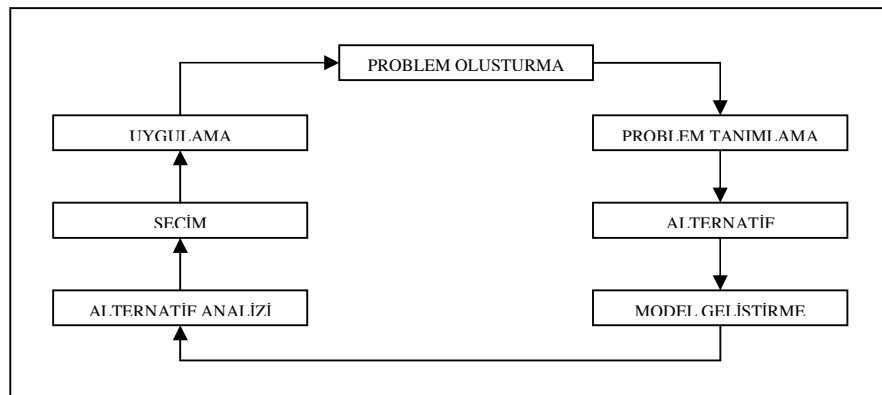
verisinin model üzerindeki etkisi çözümlenebilir. “What-if” çözümlenmeleri Uzman Sistemler (US) ile uygulanabilirken geleneksel sistemlerde “what-if” senaryolarının test edilmesi, bu senaryoların daha önceden yazılmış iş programları yüzünden zordur. Hedef araştırma çözümlenmeleri, çıktı olarak nitelendirilen bir hedefin istenen bir düzeyi için gerekli olan girdileri kontrol eder. Geriye dönük çözüm yaklaşımını içerir.

1.3.4. UYGULAMA EVRESİ

Bu evre, öneri olarak sunulan çözümün uygun olduğunun kararlaştırılması durumunda gerçekleştirilir. Uygulamanın başarılı olduğunun kanıtı orjinal problemin çözümüdür. Problemin uygulama sonucunda çözülememesi, yani başarısızlık durumunda karar alma sürecinde bir önceki aşamalara geri dönülür. Şekil 1.1.’in ortaya koyduğu gibi karar alma süreci tekrarlı bir süreç olup görüldüğü kadar kolay değildir. Süreç boyunca herhangi bir evreye tekrar tekrar gidilebileceği açıktır.

1.3.5. BİR KD SİSTEMİ ORTAMINDAKİ GELENEKSEL KARAR SÜRECİ

Karar alma modellerini kullanan daha geleneksel bir süreç Şekil 1.3’de sunulmuştur. Bu geleneksel modeli önemli kılan etken, problem çözümlenmesini ve model geliştirmeyi gerçekleştiriyor olmasıdır. Bu süreçte ilk olarak problem oluşturulur ve daha sonra problem, model yaradılışıyla uyuşan bir yolla tanımlanır. Modeller çeşitli alternatifleri ve gerçekleştirilen seçimleri çözümlenmede kullanılır. Süreci oluşturan aşamalar birbirlerini kapsamaktadır. Süreç, problem anlaşılıp başarılı çözümler seçilinceye kadar bir önceki aşamalara geçen tekrarlara sahiptir.



Şekil 1.3. Bir KD Sistemi’nin karar alma süreci (Shim ve diğ., 2002).

1.4. KARAR KURAMI

Karar kuramı, karar vericiye karar alma ve karar sürecini geliştirme konularında yol gösteren bir yaklaşımdır. Söz konusu yaklaşım, karar alınacak duruma ilişkin bilgi miktarına göre üç başlık altında incelenir (Cinemre, 2004).

Belirlilik Durumunda Karar Alma: Karar vericinin, aralarından seçim yapacağı karar seçeneklerine ilişkin sonuç değerleri ile olayın yapısı hakkındaki bilgisinin eksiksiz olduğu durumdur. Doğrusal programlama belirlilik durumunda karar alırken kullanılan yöntemlerden bir tanesidir.

Risk Durumunda Karar Alma: Olayların gerçekleşme olasılıklarının bilinmesi durundaki karar problemidir. Cinemre (2004), risk durumunda kullanılan başlıca ölçütleri en yüksek olasılık, beklenen değer, beklenen fırsat kaybı veya beklenen pişmanlık şeklinde sıralamıştır.

Belirsizlik Durumunda Karar alma: Olayların gerçekleşme olasılıklarının bilinmemesi veya belirlenememesi koşullarında ortaya çıkan belirsizlik durumunda alınan kararlardır. Belli başlı ölçütleri Laplace, minimaks (maksimin), maksimaks (minimin), Savage, Hurwicz şeklinde sıralanabilir.

1.5. KARAR ÇÖZÜMLEMESİ

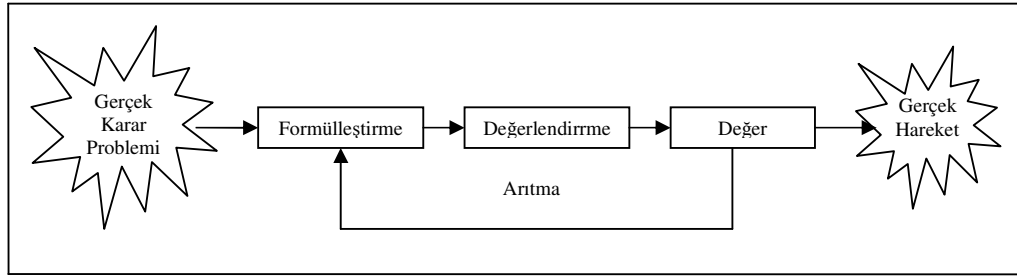
Karar çözümlemesi, özellikle belirsizlik içeren karmaşık durumlarda karar alıcılara yardımcı olan tekniklerdir. Alınan karara verilen önem ve karar alıcıların durumsal tercihleri kararın kalitesini belirleyen belli başlı unsurlardır.

Karar çözümlemesi; bireyler, gruplar veya organizasyonlar tarafından alınan kararları modelleme, optimize etme ve çözümlemeyle ilgilenen bilim dalları olan yöneylem araştırması ile istatistiğin birleşimini içeren geniş bir nicel alanı ilgilendirmektedir (Covaliu, 2001).

1.5.1. KARAR ÇÖZÜMLEMESİ SÜRECİ

Karar çözümlemesi, Şekil 1.4'de açıklanan süreci kullanmak suretiyle yerine getirilir. Sürecin amacı, karar alıcının kendisine önerilen hareketle işe başlaması

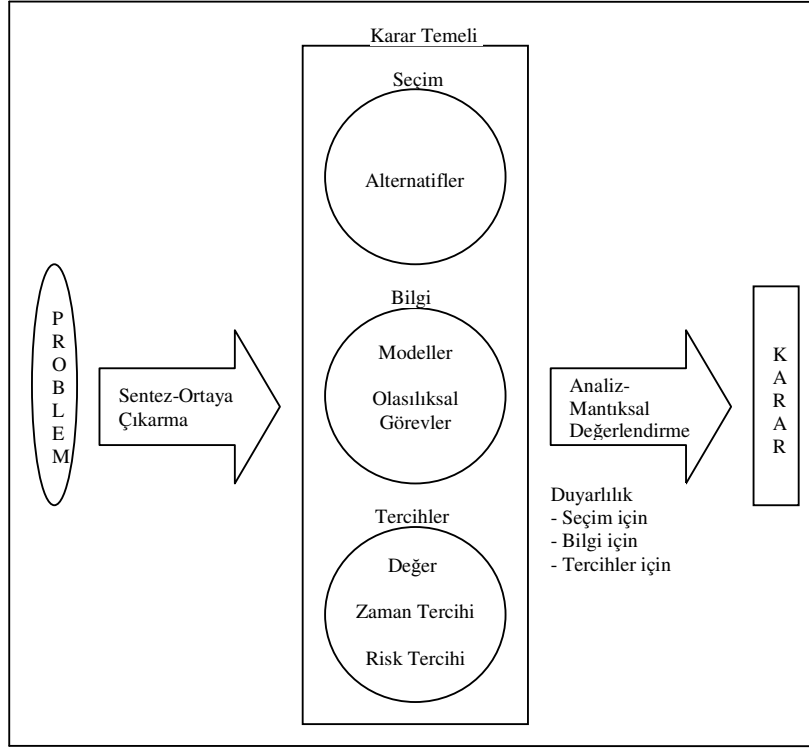
durumunda problem içindeki kavrayışları anlayabilmek için ardışık biçimde gerçekleşecek belli adımlara başvuracaktır.



Şekil 1.4. Karar çözümlenmesi süreci (Howard, 1988).

Karar çözümlenmesi sürecine, karar alıcının karşı karşıya olduğu gerçek bir karar problemi ile başlanır. Daha sonra ele alınan probleme uygun bir model bulmayı içeren formülleme aşaması gerçekleştirilir. Problemin şekilsel gösterimi “kararın temeli” olarak adlandırılır. Bir sonraki adım, oluşturulan karar temelinin, alternatifleri belli olan bir hesaplama süreci tarafından değerlendirilmesidir. Bu noktada uygulanması düşünülen alternatifin doğru seçim olduğunun anlaşılmasını sağlayan değer biçme işlemi gerçekleştirilir. Değer biçmenin yeterince tatmin edici bulunması durumunda seçim uygulamaya konur.

Karar çözümlenmesinin işleyişi, karar temeli olarak adlandırılan formülleme aşamasına dayanmaktadır. Karar temelinin karar çözümlenmesi içindeki rolü Şekil 1.5’te ayrıntılı biçimde gösterilmiştir.



Şekil 1.5. Karar temelinin ortaya çıkarılması ve değerlendirilmesi (Howard, 1988).

Şekil 1.5.'te görüleceği gibi karar temeli üç bölümden oluşmaktadır. Bu bölümler; seçimler veya karar alıcının karşı karşıya olduğu alternatifler, konuyla ilgili bilgi ve karar alıcının tercihleridir (Howard, 1988).

Alternatifler kolaylıkla görülebilmelidir veya formülasyonun temel bir aktivitesiymiş gibi oluşturulabilmelidir. Bilgi ile; modeller, ilişkiler veya kararlar ve çıktılar arasındaki iletişimi karakterize etmede önemli olabilecek olasılıksal görevlerden herhangi biri ifade ediliyor olabilir. Karar alıcının tercihleri; değer, zaman önceliği ve risk önceliği faktörlerine göre üç boyutlu biçimde değerlendirilmelidir. Değerler çıktılardan elde edilir. Zaman önceliği girdilerin öncelik sırasını, risk sırası ise çıktılarının sahip oldukları farklı kesinlik derecelerini ifade etmektedir.

Temeldeki işlemlerin gerçekleştirilmesinin ardından bu işlemlerin değerlendirilmesine geçilir. Daha sonra en iyi karara ulaşmak için mantıksal işlemler yürütülür. Değerlendirme ile seçim, bilgi ve tercihlere duyarlılık sağlanması hedeflenir.

1.5.2. KARAR ÇÖZÜMLEMESİ'NİN UYGULAMA ALANLARI VE KAPSAMI

1.5.2.1. Genel Uygulama Alanları

Karar çözümlemesi, daha çok iş ve idari alanlarda karar alıcılara yardım etmek için kullanılır. Covaliu (2001) karar çözümlemesinin genel uygulama alanlarını aşağıdaki şekilde gruplandırmıştır.

İş Uygulamaları: Hava alanı ve otel yönetimi, petrol araştırma, kalite kontrolü ve sigorta, güvenilirlik ve bakım, mahsul koruma, kredi ve ödünç para verme yönetimi, proje seçimi, yeni ürün geliştirme, yeni girişim başlatma.

İdari Uygulamalar: Acil yönetim, çevresel risk yönetimi, yeni enerji kaynaklarının seçimi, plan araştırma ve geliştirme.

Genel Uygulamalar: Medikal tanı ve tedavi, teşebbüs ve ciro işlemleri.

1.5.2.2. Karar Çözümlemesinin Kapsamı

Karar çözümlemesi, insanlara daha iyi kararlar almalarında yardım etmek amacıyla tasarlanmış nicel bir disiplindir. Karar çözümlemesinin nicel olma özelliği karmaşık ve zor kararları modellemede, optimize ve çözümlemede yardımcı olan normatif bir iskelet sunması ve bu amaçları gerçekleştirmek üzere çeşitli araç ve prosedürleri sunabilmesinden gelmektedir.

Karar almanın nicel yönünün yanında bir de nitel, yani betimsel yönü vardır. Karar almanın betimsel yönü insanların gerçekte kararları nasıl aldıklarıyla ilgilidir. Betimsel yön, daha çok insan psikolojisine dayanır (Covaliu, 2001). Betimsel olarak karar alma, insanların bilgiyi işlemelerinde, olasılıkları hesaplamalarında ve karar çözümlemesinin rasyonel emirleriyle çelişen kararlarında yardımcı olacak deneysel kanıtlar sağlar. Deneysel sonuçlara dayanan karar almanın betimsel yönü de göz önüne alınarak, özellikle zor ve önemli durumlarda iyi kararların alınabilmesi için nicel kanıtlara dayanan karar çözümlemesi araçlarının önemi daha da artmaktadır.

Karar çözümlemesinin kapsadığı kararların birçoğu belirsizlik içerdiğinden iyi (yüksek kaliteli) karar ile şanslı sonuç (lucky outcome) arasındaki farkın ayırt edilmesi gerekmektedir (Covaliu, 2001). İyi karar, karar çözümlemesinin önerdiği

rasyonel yaklaşım neticesinde elde edilen sonuç olup karar probleminin bütün yönlerini anlama ve problemi doğru modellemeye dayanır. Şanslı sonuç ise sadece şans eseri oluşmuş bir sonuç olup kalitesinin mutlaka çözümlenmesi gerekir.

1.5.3. KARAR ANALİZİYLE İLGİLİ BİLİM DALLARI VE METODOLOJİLER

Karar çözümlemesi, birçok bilim dalı ve metodoloji ile birlikte hareket etmektedir. Covaliu (2001) bu bilim dalları ve metodolojileri olasılık teorisi, istatistiksel karar teorisi, bayes, grafik teorisi ve paradigma, stokastik süreçler, stokastik dinamik programlama, bilgi teorisi, fayda teorisi (çok nitelikli), oyun teorisi, ciro işlemi, matematiksel programlama, hedef programlama olarak sıralamıştır.

1.5.4. BİR KARAR MODELİNİ OLUŞTURAN TEMEL ELEMANLAR

İster basit ister karmaşık olsun bütün karar modelleri aşağıda sıralanmış olan unsurları içermektedir (Covaliu, 2001).

Karar Alıcı Tarafından Alınmış Olan Kararları (Kararları ve Alternatifleri): Karar, karar alıcının mevcut alternatifler dışında kesikli veya sürekli, basit veya birleşik olabilecek başka bir alternatif seçmesini gerektiren zamanı belirler (Covaliu, 2001).

Alternatif, çok sayıdaki karar değişkenleri için belli bir karar noktasında seçilmiş olan değerlerin bir kombinasyonu olabilir (Covaliu, 2001).

Doğal Durumu Açıklayan Belirsiz (Kesin Olmayan) Olayları: Belirsiz olaylar genellikle birbirini izleyen kararlar arasında oluşan boşlukta meydana gelir (Covaliu, 2001). Eğer belirsizlik karar alınmadan önce oluşmuşsa karar alma işlemine başlamadan önce bu belirsizliğin incelenmesi gerekmektedir.

Kararlar ve Belirsiz Olaylar Neticesinde Oluşan Sonuçları: Sonuçlar, karar alıcıların tanımladıkları bütün karar noktalarındaki seçtikleri alternatifleri ve belirsiz olayların neticelerini içerir. Sonuçlar, karar alıcının amaçlarıyla yakından ilgilidir.

Karar Alıcıların Sonuçlarla İlgili Ne Düşündüklerini Gösteren Amaçları ve Tercihleri: Bir karar problemini çözümlenmeye çalışan karar alıcının belirlediği

amaçlar, tercih ettiği direktifleri de içeren, problemin üzerinde düşündüğü niceliklerdir.

Genelde karar problemleri birden fazla amaç içerir. Birden fazla amacın olduğu problemlerde bu amaçların her birini optimal yapmak için farklı strateji kullanımı gerekebilir. Böyle durumlarda karar alıcı belirlediği amaçlardan bazılarını eler. İşte bu eleme işlemi karar alıcının tercihlerini ifade etmesi anlamına gelir.

Birden fazla amacın olduğu durumlarda, tercihler belirlenirken amaçların elenmesi yoluna gidilebilir. Tek bir amacın olduğu durumlarda bile eğer bir belirsizlik söz konusu ise karar alıcı tercihlerini risk unsurunu gözönüne alarak belirler (Covaliu, 2001).

1.5.5. KARAR ÇÖZÜMLEMESİNDE KULLANILAN TEKNİKLER

1.5.5.1. Tek Amaç Olması Durumunda Kullanılan Teknikler

Karar alıcının çözümlenmeye çalıştığı problemlerin çok azında tek bir amaç söz konusu olabilmektedir. Tek amaçlı karar problemlerinde bile karar alıcının bir takım tercihlerle karşı karşıya kalması kaçınılmazdır. Bu tercihlerden en önemlisini, uygulanması düşünülen stratejinin seçimi oluşturmaktadır. Bu tür bir seçimin rasyonel olabilmesi için stratejilere ilişkin risk profillerini oluşturmak ve daha sonra farklı stratejilerin belirlenen amacı elde etmede nasıl değerlendirileceğini belirten uygun bir “karar kuralı”nı işletmek gerekmektedir. Böylelikle ulaşılabilecek faydanın maksimum olması sağlanacaktır.

Tek amacın olması durumunda kullanılan karar kurallarını Covaliu (2001) maksimum beklenen değer, maksimum beklenen değer ile birlikte minimum varyans, etkin strateji seçimi, maksimum beklenen fayda olarak sıralamıştır.

1.5.5.2. Çok Amaç Olması Durumunda Kullanılan Teknikler

Çok amacın söz konusu olduğu durumlarda bir karar kuralı yardımıyla bu amaçların tek bir performans ölçümü içinde birleştirilmesi gerekmektedir. Birleşim için kullanılan karar kuralının subjektif ve yargısal olması karar alıcının seçimine bağlıdır. Anlaşmazlık yaratan amaçlar hakkında uzlaşmaya varmanın bir yolu da o amaçlardan vazgeçmektir (Covaliu, 2001). Amaçlardan vazgeçmeyi etkili bir

biçimde gerçekleştirmek için kullanılan en yaygın yöntem ise etkin önceliklerin gözönünde bulundurulmasıdır.

Karar çözümlemesi teknikleri, alternatifler arasından seçimin çok açık olmadığı durumlarda bilgi, veri ve dengeli bir karar almayı sağlayacak deneyimler hakkında eleştirel düşünmeyi sağlayacak rasyonel süreçler veya sistematik prosedürlerdir (Baker ve diğ., 2001). Bu teknikler, problemle ilgili soruların yanıtlarını oluştururken eleştirel düşünme yollarını kullanan yolları düzenlemeyi sağlar. Karar çözümlemesi teknikleri ile kararların analitik yapısı oluşturulur. Bu yapı, karar alıcılara amaçlarını, tercihlerini, problemin yapısını ve belirsizliklerini düşünmelerini sağlayacak sistematik bir ortam sağlar (Covaliu, 2001). Daha sonra probleme ilişkin belirlenen bu amaçlar, tercihler, yapı ve belirsizlik ile problemin içerdiği diğer önemli unsurlar göz önünde tutularak problemin nicel modellenmesi gerçekleştirilir. Baker ve diğ. (2001) karar çözümlemesi tekniklerini aşağıdaki gibi sıralamıştır.

Lehte ve Aleyhte Olanlar Çözümlemesi : Lehte ve aleyhte olanlar çözümlemesi, her bir alternatif için iyi şeyler (lehte) ile kötü şeylerin (aleyhte) tanımlandığı nitel bir karşılaştırma yöntemidir (Baker ve diğ., 2001).

Lehte ve aleyhte olan şeylerin listesi uzman kişilerin problem ile ilgili oluşturdukları girdiler olup her bir alternatif için birbirleriyle karşılaştırılır. Karşılaştırma sonucunda, genellikle en güçlü lehte olan girdiler ile en zayıf aleyhte olan girdiler tercih edilir. Lehte ve aleyhte olanlar çözümlemesi, alternatiflerin sayısının en fazla 2 ile 4, 1 ile 5 arasında olduğu basit karar problemleri için uygundur. Matematiksel yapıları içermemesi çözümlemenin uygulanmasını basit kılmaktadır.

Kepner-Tregoe (K-T) Karar Çözümlemesi: K-T karar çözümlemesi, kriter ve alternatiflerin uzmanlardan oluşan bir grup tarafından bireysel yargıya göre sayısal olarak değerlendirildiği nicel bir karşılaştırma yöntemidir.

Çok nitelikli bir problem ile karşı karşıya kalındığı zaman, bütün niteliklerin aynı anda en iyilenmeleri güç olduğundan, en iyi alternatifin seçimi de o ölçüde zordur. Bu tür problemlerde karar alma süreci, alternatifleri önem derecelerine göre sıralayıp az önemli olan alternatiflerden bazılarını elemek suretiyle kolaylaştırılabilir. Sistematik karar alma yaklaşımı, karar alma sürecinin tüm yönlerini sergileyerek doğru kararlar alınmasında yardımcı olmaktadır. Bilgisayar kullanımının söz konusu

olmadığı günlerde karar alma problemlerinin çözümünde yaygın biçimde kullanılan yöntem K-T yöntemidir. Bu yöntem kararın amacını belirlemek, amaçları saptamak, amaçları önem derecelerine göre sıralamak, alternatifleri oluşturmak, alternatifleri amaçlara göre değerlendirmek, deneme niteliğinde bir seçim yapmak, bu seçimin potansiyel karşıt sonuçlarına değer biçmek, final bir seçim yapmak aşamalarını içermektedir (Mallach, 1994).

Analitik Hiyerarşi Yöntemi (AHY): AHY, istenen alternatifi tüm alternatifleri kriterlere karşı olan görelî performanslarına dayalı olarak karşılaştırmak suretiyle belirleyen nicel bir yöntemdir (Baker ve diğ., 2001).

AHY'nin, yalnızca alternatiflerin kendi aralarında iyi tanımlanmış matematiksel fonksiyonlarla (genellikle doğrusal) ilişkilendirildiği diğêr Karar Destek (KD) araçlarından farkı; fikirlerin, duygu ve heyecanların, karar alternatiflerinin sayısal bir ölçekle sıralanmasını sağlayacak şekilde ölçüldüğü durumlara farklı bir yaklaşım getirmesidir (Erpolat ve Cinemre, 2006).

Çok Amaçlı Fayda Teorisi (ÇAFT): ÇAFT, benzer olmayan maliyetler, riskler ve faydaların ölçümlerini, bireysel ve grup tercihleriyle birleştiren nicel bir karşılaştırma yöntemidir (Baker ve diğ., 2001). ÇAFT'nin temeli fayda teorisine dayanmaktadır. ÇAFT, her bir alternatif hakkında daha güçlü alternatif performans kestirimlerini sağlayacak nicel bilginin olduğu durumlarda kullanılır.

Maliyet-Fayda Çözümlemesi (M-F): M-F çözümlemesi, hükümet projeleri veya politikalarının istenebilirliğini değerlendiren sistematik nicel bir yöntemdir. Bu yöntem, karar almanın esas amacının uygun alternatifi belirlemekten ziyade alternatifin parasal maliyetini belirlemek olduğunda kullanılır (Baker ve diğ., 2001).

Alışkanlık Değıştirme Araçları (ADA): Alışılmış araçlara, sistem içindeki karmaşık davranışların anlaşılmasında ihtiyaç duyulabilir. Alışkanlık değıştirme araçlarının geliştirilmesi oldukça uzun zaman alır ve oldukça fazla kaynak kullanımı gerektirir.

1.6. KARAR DESTEK

Karar Destek (KD), karar almada kullanılan ve genellikle geniş bir alana sahip bir kavramdır. Oldukça basit ve sezgisel bir kavram gibi görünmekle birlikte gerçekte kişiden kişiye veya alandan alana farklı şeylere karşılık gelen bağımsız bir ifadedir. Bu nedenle KD’i tam olarak açıklayan genel kabul görmüş bir tanım henüz bulunamamıştır.

1.6.1. KD’İN ÇEŞİTLİ TANIMLARI

Yaygın kullanım alanına sahip olan KD, kullanıldığı alana bağlı olarak farklı biçimde tanımlanmaktadır. Bu yüzden KD’in üzerinde mutabık olunan ortak bir tanımı bulunmamaktadır. KD’in genel tanımı genellikle KD’in temel karakteristiklerini içeren kısa ve öz tanımlarıdır. KD’in farklı tanımları ve karakteristikleri aşağıdaki gibi sıralanabilir (Bohanec, 2006).:

1. Veriye farklı açılardan bakmak suretiyle karar almaya yardımcı olmaktır.
2. Karar alabilmek için gerekli olan bütün veriyi tanımlamak ve bunu anlamlı bir bilgi halinde biraraya getirmektir.
3. Karar almak için kullanılan kimi zaman yapısal kimi zaman da matematiksel tabanlı yaklaşımdır: Kepner-Tregoe, akış diyagramları, analitik hiyerarşi, nominal grup süreci, vb.
4. Oluşturulan bütün alternatiflerin etkilerini anlamak suretiyle iyi kararların alınmasına yardımcı olmaktır. “What will happen if ... ?” sorusuna verilecek yanıtın geniş alanlı senaryolarına izin vermektir.
5. Karar sürecinin değerini artırmak üzere geliştirilmiş, veri çözümlemesinin uzmanlaştırılmış halidir.
6. KD, karar almadaki veri, model ve yapısal karar sürecinin kullanımını kolaylaştıran bilgisayar tabanlı sistemlerden (karar teorisi, karar çözümlemesi, yöneylem araştırması, yönetim bilimi ve yapay zeka) yararlanır.

Yukarıdaki (1), (2) ve kısmen de (5) nolu tanımlar, veri ambarı ve veri madenciliğini içeren veri merkezli yaklaşım üzerinde durmaktadır. (3) nolu tanım modelleme, (4)

nolu tanım ise benzetim üzerine odaklanmaktadır. Son tanım KD'in ilgili olduğu bütün yöntem bilim ve tekniklerden söz etmesi bakımından en kapsamlı olanıdır.

1.6.2. KD NEDİR?

KD, karar ve destek kelimelerinden oluşmaktadır. Bu kelimelerden “karar” karar alma, “destek” ise insanları desteklemek anlamındadır.

KD ilk olarak ortaya çıktığı zaman, yöneylem araştırması ve karar çözümlemesi ile birlikte, sonraları KD Sistemi ile aynı anlamda kullanılmıştır (Bohanec, 2006). Günümüzde ise “veri ambarları, on-line analitik süreç ve veri madenciliği ile birlikte anılmaktadır (Watson, 1998).

1.6.3. KD'İN SINIFLANDIRILMASI

KD'i farklı şekillerde sınıflandırmak mümkündür. Herhangi bir KD'in aşağıdaki sınıflandırmalardan en az birinin içinde yer alacağı kesindir (Bohanec, 2006).

Kullanılan Bilim Dalına Göre Sınıflandırma: Yöneylem araştırması, karar çözümlemesi, KD Sistemi, veri ambarı, veri madenciliği, Grup Karar Destek (GKD).

Karar Türüne Göre Sınıflandırma: Yapısal, yarı yapısal, yapısal olmayan.

Organizasyonel Düzeye Göre Sınıflandırma: İşlemsel, taktik, stratejik.

Karar Alıcı Türüne Göre Sınıflandırma: Bireysel, grup, organizasyonel.

Geçerli Olan KD Elemanına Göre Sınıflandırma: Veri, model, yararlı bilgi, iletişim.

Kullanılan Yönteme Göre Sınıflandırma: Raporlama, görsellik, modelleme (nicel ya da nitel), benzetim, optimizasyon.

1.6.4. KD'DE KULLANILAN BİLİM DALLARI

KD, birçok alanda kullanılabildiğinden çok sayıda bilim dallarını içerebilmektedir. Bunlardan en çok kullanılanları yöneylem araştırması, karar çözümlemesi, KD Sistemi, veri ambarlama ve GKD şeklinde sıralanabilir.

1.7. KARAR ALAN ELEMANLAR

Fain, karar alan elemanları “Bilgisayar” ve “insan” olmak üzere iki gruba ayırmıştır (Andriole, 1986).

1.7.1. KARAR ALMADA BİLGİSAYARLAR

Karar alırken bilgisayarlara olan gereksinim Fain tarafından dört genel sınıfa ayrılmıştır (Andriole, 1986):

Bilgiyi Yeniden Düzeltme: Yeniden düzeltilmiş bilgi, daha sonraki kullanımlar için bilgisayar sistemi içinde yeniden depolanmış olmalı; kullanıcı için kopyalanmak üzere yazıcıya gönderilebilmeli; tablolar, grafikler, diyagramlar, vb. bir ekran üzerinde görüntülenebilmelidir.

Hesaplama: Genellikle sayısal veri üzerinde bir veya daha çok aritmetik işlemin yapılmasıyla ilgilidir. Kullanıcının belirlediği işlemleri yerine getirmeyi içerir.

Tercüme: Bu süreç, hesaplama gibi, bir kurallar kümesine göre bilginin kullanımını içermektedir. Bu kurallar kullanıcı tarafından program uygulaması boyunca ya sunulmuş (gösterilmiş) ya da seçilmiştir.

Öğrenme: Program uygulaması boyunca depolanmış bilgisayar öğrenimlerinin değişikliğidir.

1.7.2. KARAR ALMADA İNSANLAR

Karar alırken gerekli olan insan ihtiyacı Fain tarafından profesyonel bilgisayarıcı ve çözümleyici olmak üzere iki genel sınıfa ayrılmıştır (Andriole, 1986).

1.7.2.1. İnsanların Karar Alma Süreci

İnsanlara yardım etmek amacıyla karar alma sürecini bilgisayar desteğiyle otomatikleştirmeye çalışırken insanların tamamıyla rasyonel olmadıkları unutulmaması gereken önemli bir husustur. İnsanların olaylara tepki gösterme yolları, problemleri kavrama tarzları, değer ve yargıları kararların farklı alınmasına neden olabilir (Turban, 1993). İnsanların psikolojik kişilikleri karar alma sürecinde önemli rol oynamaktadır. Öyle ki insanların gerçekler karşısındaki davranış biçimleri kişiliklerine göre farklılık gösterir. Karar alıcının kişilik tipini bilme, bu kişilere

destek sağlayacak uygun araçların tasarımlarında yardımcı olacaktır (Mallach,1994). Çizelge 1.4'deki "Kişi türlerine göre tercih edilen karar türleri" çizelgesinde kişilik türlerine göre tercih edilen karar alma teknikleri özetlenmiştir. KD Sistemi'nin yararlı olabilmesi için sistemin karar alıcının tercih ettiği karar alma tekniklerinden bazılarını içermesi gerekmektedir.

İnsanlar karar aldıklarında genellikle rasyonellik, politik durum, esneklik kategorilerini gözönünde bulundururlar. Rasyonellikle, bilginin tarafsız biçimde toplanarak çözümlenmesi ve belirlenen amaçlara göre son bir seçim yapılması anlatılmaktadır. Bir grup halinde karar alan üyeler arasında bir anlaşmazlık söz konusu olduğunda, bağlı bulunan organizasyonun hedef ve gücüne göre karar alınması uygun olur. Politik karar alma olarak bilinen bu durum, uzlaşma yöntemi olarak karakterize edilir ve amaç kazan kazan (win-win) hedefini gerçekleştirmektir. Esneklik, karar almanın, geleneksel kalıp dışında gerçekleştirilmesini sağlayan yetenektir.

1.7.2.2. Karar Alma Stilleri

Herhangi bir problemle karşı karşıya kalındığında çözüm için seçilen yöntemin ve uygulanması düşünülen kararın insandan insana değişim göstermesi, sahip olunan psikolojik kişiliklerin farklı olmasıyla ilgilidir. Seçilen yöntemin farklılık göstermesinin bir diğer nedeni ise kararların bir birey ya da grup tarafından alınmış olmasıdır. Sauter (1999), karar alma stillerini sol beyin, sağ beyin, yardım, birleştirme şeklinde ayırmıştır. Çizelge 1.4.'de karar alma stillerine göre tercih edilen teknikler verilmiştir.

Çizelge 1.4. Kişi türlerine göre tercih edilen karar türleri (Sauter, 1999).

KARAR ALMA STİLLERİ	TERCİH EDİLEN TEKNİKLER
Sol Beyin	Analitik ve nicel teknikler.
Sağ Beyin	Bütünün bir bölümünden ziyade tümünü ilgilendiren yapısal olmayan ve doğal prosedürler: Beyin fırtınası.
Yardım	Gereksinim duyulan başka karar alma stillerini benimseyen üstün stiller.
Birleştirme	Sezgi, sürekli olarak kararın uygunluğunu onaylayarak karar alıcıya belirsizlik ve karmaşıklığın üstesinden gelebilmeyi sağlarken sol ve sağ beynin birleştirilmesi bilginin analitik biçimde sembollerle filtreleme avantajını sağlar.

1.8. KARAR ALMADA BİLGİ TEKNOLOJİSİ KULLANIMI

Herhangi bir kararın alımı için kullanılabilir alternatiflerin sayısı binlerce olabilir. Karar alıcının doğru karar alabilmesi için çok sayıdaki bu alternatifleri makul bir sayıya indirilmesi gerekmektedir. İşte bu aşamada KD devreye girer. Bilgisayarlar, özellikle alternatiflerin sayısal biçimde ifade edilebileceği durumlarda, alternatifleri değerlendirmede kullanılabilir. Eğer bu mümkün olmuyorsa bilgisayarlar, en azından alternatifleri karar almayı kolaylaştıracak biçimde sunarak karar alıcıya yardımcı olabilir. Karar alma sürecinin bir grup tarafından gerçekleşmesi gerektiği durumlarda KD Sistemi'nin kullanımıyla grup üyelerinin aynı konu üzerine odaklanmaları sağlanır. Elektronik posta gibi KD araçlarının kullanımıyla ise farklı yerleşim yerlerindeki insanların ortak bir karar almalarını sağlamak amacıyla, iletişim kurmaları sağlanabilir.

Karar almayı gerektiren birçok faaliyet mevcuttur. Bu faaliyetleri desteklemek için sistemin aşağıda sıralanan ifadelerden bir veya birkaçına sahip olması gerekmektedir (Finlay, 1994).

- Mevcut veya yeni başlayan problemleri bulmaya yardım etmek,
- Probleme açıklık getirerek problemin durumunu modellemeye yardım etmek,
- Seçenekleri göz önünde bulunduracak araçları sağlamak,
- Değişimleri uygulamaya ve bunların yeniden gözden geçirilmesine yardım etmek.

Bilgi Teknolojisi (BT)'nin başarısındaki önemli nokta, kullanıcıya doğru zamanlarda doğru bilgiyi sağlama yeteneğidir (Turban vd., 2001). KD; programlama dilleri, veri tabanları, ağ ve sistem çözümleri gibi bütün Bilgi Sistemleri (BS) alanlarını kapsar. KD Sistemi, yukarıda sıralanan bu teknolojileri özel bir uygulama türü geliştirmek amacıyla bir araya getirebilir. BS'nin her biri aslında bir KD yeteneğine sahiptir.

Yöneticileri veya karar alıcıları desteklemek için kullanılan dört BT vardır (Turban ve diğ., 2001).

1. Analitik ve nicel kararları destekleyen KD Sistemleri.

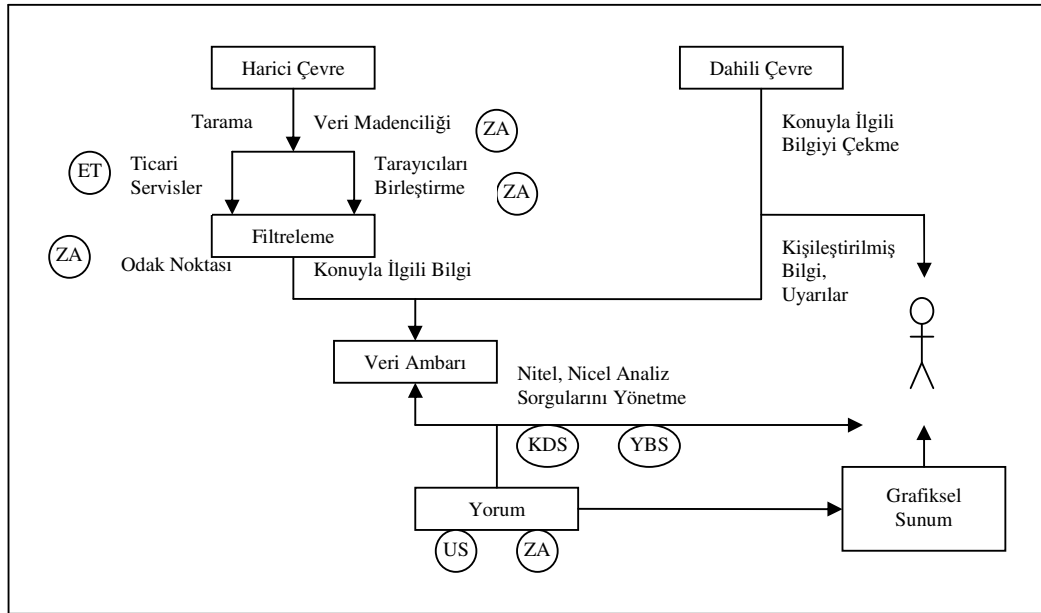
2. Yönetim Bilgi Sistemleri (YBS) olarak adlandırılan, bilgiyi desteklemek ve karar alıcılara çözümlenme ve iletişim araçları sağlamak amacıyla geliştirilmiş sistemler.

3. Grup Karar Destek Sistemleri (GKD Sistemleri) veya Grup Destek Sistemleri (GD Sistemleri) olarak adlandırılan kararların grup halinde alınmasını sağlayan sistemler.

4. Akıllı destek sistemler olarak adlandırılan, herhangi bir kararı optimize etmede uzman bir bileşen yardımıyla uzman bilgi sağlayan sistemler.

Yukarıda sıralanan sistemlerin tümüne birden Yönetim Destek Sistemleri (YDS) denir. Her biri farklı yetenekler içeren bu sistemler bağımsız biçimde veya birlikte kullanılabilir.

Karar alıcılara yardım etmek amacıyla kullanılan bilgisayarlaştırılmış destek Şekil 1.6'da gösterilmiştir.



Şekil 1.6. Karar almak için bilgisayarlaştırılmış destek (YDS)¹
(Turban ve diğ., 2001).

¹ Şekil 1.4'de kullanılan kısaltmaların açıklımları aşağıdaki gibidir (Turban ve diğ., 2001).

ZA : Zeki Ajanlar
US : Uzman Sistemler
ET : Elektronik Ticaret
KDS : Karar Destek Sistemleri
YBS : Yönetimsel Bilgi Sistemleri

BÖLÜM 2

MODELLEME VE PROBLEM ÇÖZÜMLEME SÜRECİ

2.1.GİRİŞ

İnsan yaşamı bir dizi karar alma durumunu içeren bir süreçtir. Her insan günlük yaşantısında ne yiyeceği, ne giyeceği, saat kaçta uyuyacağı, TV’de hangi programı izleyeceği gibi son derece basit konuların yanı sıra, tasarrufunu nasıl değerlendireceği, bütçesini nasıl ayarlayacağı, çocuğunu hangi okula göndereceği gibi daha önemli konularda, zaman zaman da sözgelimi, hastalığın tedavisinde ne yapması gerektiği gibi yaşamsal öneme sahip konular hakkında karar almak durumundadır. Alınan bu kararların bir kısmı işleri kolaylaştırırken bir kısmı daha da karmaşıklştırabilmektedir. Çünkü doğal ve kolay gibi görünse de, aslında karar almak özellikle de iyi bir karar almak gerçekten zor bir görevdir. Bu görevi yerine getirirken belli kurallara uyulması ve belli işlemlerin gerçekleştirilmesi gerekmektedir. Organizasyonların karşı karşıya oldukları problemlerin hemen hepsi oldukça karmaşık yapıda olduklarından çözümleri için ileri düzey bilgi ve teknolojiye gereksinim vardır. Problemlerin çözümünü gerçekleştirmek ve iyi kararlar alabilmek için kullanılan yardımcı araçların başında modeller gelir. Modelleme, gerçeği kolaylaştırır ve karar alternatiflerinin kavramsallığına yardım eder (Mallach, 1994). Model, gerçeğin basitleştirilmiş bir temsilidir (Turban ve diğ., 2005). Model kullanmanın amacı karar alma sürecini ilerletmek ve kolaylaştırmaktır.

Bu bölümde, modelleme üzerinde durulacak ve karar alma sürecinde kullanılan model çeşitleri açıklanmaya çalışılacaktır.

2.2. KARAR ALMADA MODELLEME YAKLAŞIMI

Model kullanarak problem çözümünü ve karar çözümlemesini gerçekleştirmek, eskiden beri kullanılan etkili bir yöntemdir. Gerçekleştirilmesi için bilgisayar desteği

şart olmasa da, günümüz problemlerinin bir çoğunun elle çözülemeyecek kadar karmaşık yapıda olması, zaman ve maliyet gibi çeşitli unsurlardan dolayı modelleme, bilgisayar destekli olarak gerçekleştirilmektedir.

Modelleme, karar alma sürecinde genellikle alternatiflerin çok olduğu durumlarda ihtiyaçlara karşılık verecek ve amacı gerçekleştirecek en iyi alternatifin seçimi, yani çözüm için kullanılmaktadır. Kimi problemlerde alternatif sayısının az ve basit, kimilerinde ise çok ve karmaşık olmasından dolayı çözüm için kullanılan modeller yapı bakımından farklıdır. Örneğin, bir odanın eşya düzenini gerçekleştirmek için mental model kullanımı yeterli olabilirken, büyük bir binanın mimarisini gerçekleştirmek için görsel bir modelin kullanımı gerekmektedir. Elektronik sektöründe ise bir başka model yöntemi olan fiziksel veya ölçek modelleri kullanılırken, karar problemlerinde matematiksel modeller kullanılmaktadır.

2.2.1. GRAFİKSEL PARADİGMA VE MODELLEME

Karar problemlerini modellemede grafiksel paradigmaları kullanmak, problemin daha anlaşılır ve basit kılınması bakımından büyük önem taşımaktadır. Yaygın kullanılan grafiksel paradigmalar akış diyagramları ve karar ağaçlarıdır.

Paradigmalarda farklı olaylar farklı sembollerle temsil edilir. Diktörtgenler, karar düğümleri olup kararları, yumurta şeklindeki ovaler değişim düğümleri olup belirsiz olayları, köşegenleri yuvarlak diktörtgenler sonuç veya değer düğümleri olup sonuçları temsil eder. Bu düğümler arasındaki ilişkiler ise oklarla gösterilir.

Akış Diyagramları: Akış diyagramı, bir çevrimsiz grafik olup grafikte kullanılan her bir düğüm karar probleminin tek bir değişkenini, oklar ise değişkenler arasındaki ilişkileri gösterir. Akış diyagramları karar problemlerinin modellenmesinde kullanılan en etkili yoldur. Bunun başlıca nedeni bu diyagramın oldukça karmaşık problemleri bile değişkenler arasındaki varsayımsal bağımlılık veya bağımsızlığı, kararların ardışıklığını ve karar alıcının bilgi akışını belirgin biçimde ifade eden yoğlaştırılmış bir grafik içinde şekillendirip görselleştirebilmesidir (Covaliu, 2001). Bu tür diyagramlar veri ve problem ile ilgili fazla detayın olmadığı yapısal olmayan problemlerin çözümünde oldukça büyük bir yardım sağlar. Ayrıca, duyarlılık çözümlemesiyle birlikte kullanıldığında problemde sorun olan ve olmayan unsurların belirlenmesinde de yararlıdır.

Karar Ağaçları: Akış diyagramlarının tersine karar ağaçları, karar probleminin yapısındaki asimetriyi açık biçimde gösterir (Covaliu, 2001). Bu diyagramlar aynı zamanda karşılık gelen dallar üzerindeki her bir düğüm için fonksiyonel ve sayısal bilgileri de gösterebilmektedir.

2.3. MODELLEMENİN KARAKTERİSTİKLERİ VE YARARLARI

Karar alma sürecinde model kullanmanın yararları, aşağıdaki şekilde sıralanabilir (Turban ve diğ. 2001; Mallach, 1994).

- Uygulamanın gerçek bir sistemle yönetilmesinden bir model üzerinde yönetilmesinin maliyeti daha düşüktür.
- Modeller, gerçekleştirilmesi uzun yıllar alabilecek işlemlerin bilgisayar ile birkaç saniyede benzetimine izin verir.
- Modeli kullanma, gerçek sistemi kullanmadan çok daha kolaydır: deneyselliğin yönetimi model kullanıldığı zaman daha kolay olur ve organizasyonun günlük işlerini engellemez.
- Model kullanılarak, deneme-yanılma deneyselliği boyunca oluşacak hataların maliyeti, gerçek sistemin kullanıldığı zamankinden daha düşüktür.
- Modelleme, özel hareket durumlarındaki riskin hesaplanmasına izin verir: deneysellik, belirsizliğin oldukça fazla olduğu alanlarda uygulanabilir.
- Matematiksel modeller, çok büyük veya neredeyse sonsuz sayıda alternatifler ile çözümlenmelere izin verir.
- Modeller, bilgi ve destek eğitimini artırır ve güçlendirir.
- Bir modele erişme ve modeli kullanma, gerçek yaşamdaki alternatif seçimlere başvurulduğu durumdansa alternatiflerin incelenildiği durumda daha kolaydır: Birçok karar seçeneği bilgisayar modelleri aracılığıyla incelenebilir.
- Veriyi gerçek bir sistemden toplamaktansa bir bilgisayar modelinden toplamak daha kolaydır.

- Model, zamandan tasarruf sağlar ve sonuca gerçek dünyanınkinden daha çabuk bir şekilde ulaştırır.

2.4. MODELLERİN SINIFLANDIRILMASI

Modeller yapılarına, kullanım alanlarına ve içeriklerine göre çeşitli sınıflamalara tabi tutulabilir.

2.4.1. YAPILARINA GÖRE MODEL ÇEŞİTLERİ

Bazı sistemler karmaşık davranışlara sahip olabilir. Bu tip sistemleri en iyi biçimde açıklamanın yolu soyut modellerin en bilineni ve yaygın biçimde kullanılan olan matematiksel modeli kullanmaktır. Mallach (1994)'ın yapılarını dikkate alarak yaptığı model sınıflandırması şöyledir.

Grafiksel Modeller: Grafiksel modeller, veri akış diyagramı olabilir.

Anlatımsal Modeller: Anlatımsal modeller, sistemi İngilizce gibi doğal bir dil ile açıklayan modellerdir.

Fiziksel Modeller: Fiziksel modeller, gerçek sistemin daha küçük bir şekilde temsil edilmesidir.

Sembolik Modeller: Sembolik modeller, genellikle KD Sistemleri tarafından kullanılan modellerdir. Bu modeller matematiksel modeller veya bilgi tabanlı modeller olarak da adlandırılır.

Yukarıda sıralanan modellerden ilk üçü KD Sistemleri'nde kullanılamazken sembolik modeller KD Sistemleri'nin en önemli araçlarındandır.

2.4.2. İÇERİKLERİNE GÖRE MODEL ÇEŞİTLERİ

Soyutlamanın çeşitli dereceleri ile yapılabilen modelleme; ölçek, analog, mental ve matematiksel olmak üzere dört başlık altında toplanabilir (Turban ve diğ., 2001).

Ölçek Modeller: Ölçek modeller, soyutluğu en az olan modelleme türü olup sistemin fiziksel kopyalarını içerir. Genellikle orjinalden daha küçük bir ölçeğe dayalıdır. Ölçek modellere uçak, araba köprü, vb. modelleri örnek olarak verilebilir.

Analog Modeller: Analog modeller, gerçek bir sistem gibi görünmemesine rağmen gerçek bir sistem gibi davranan modellerdir. Analog modellerin şekli gerçek modelinkinden farklılık gösterir. Dağları veya suları renkleri kullanarak gösteren haritalar, makina veya ev planları gibi yönetim veya sorumluluk ilişkilerini gösteren organizasyonel yapıları grafikler, analog modellere örnek olarak verilebilir.

Mental Modeller: Mental modeller, kişinin bir durum hakkında nasıl düşündüğünü ifade eder. Bu modeller, bir birey tarafından algılanan güven, varsayımlar, ilişkiler ve ışakışını içerir. Mental modeller, önemli olan bilginin hangisi olduğunun belirlenmesini gerektiren çevresel incelemeler için gereklidir. Mental modeller subjektiftirler ve sıkça değıştiklerinden belgelenmeleri zordur. Bu modeller sadece karar alma için değil ayrıca bilgisayar-insan etkileşimi için de önemlidir.

Matematiksel Modeller: Matematiksel (nicel) modeller fonksiyonel ilişkileri açıklamada kullanılan modeller olup, gerçek bir sistemin bilgi tabanlı bir temsildir. Bu modeller, KD Sistemleri sürecinde karar almaya yardım etmek için kullanılır. Matematiksel modellerde sonuç değışkenler, karar değışkenleri, kontrol edilemeyen değışkenler ve ara (orta) değışkenler olmak üzere dört farklı değışken türü kullanılabilir (Turban ve diğ., 2001). Matematiksel modelleri Turban ve diğ. (2005) emirsel, tahmin ve betimsel (tanımlayıcı) olmak üzere üç grupta incelerken, Mallach (2000) ise matematiksel modelleri, emirsel ve betimsel olmak üzere iki grupta incelemiştir. Matematiksel modeller konusu Kesim 4.7.2.3'de ayrıntılı bir biçimde ele alınmıştır.

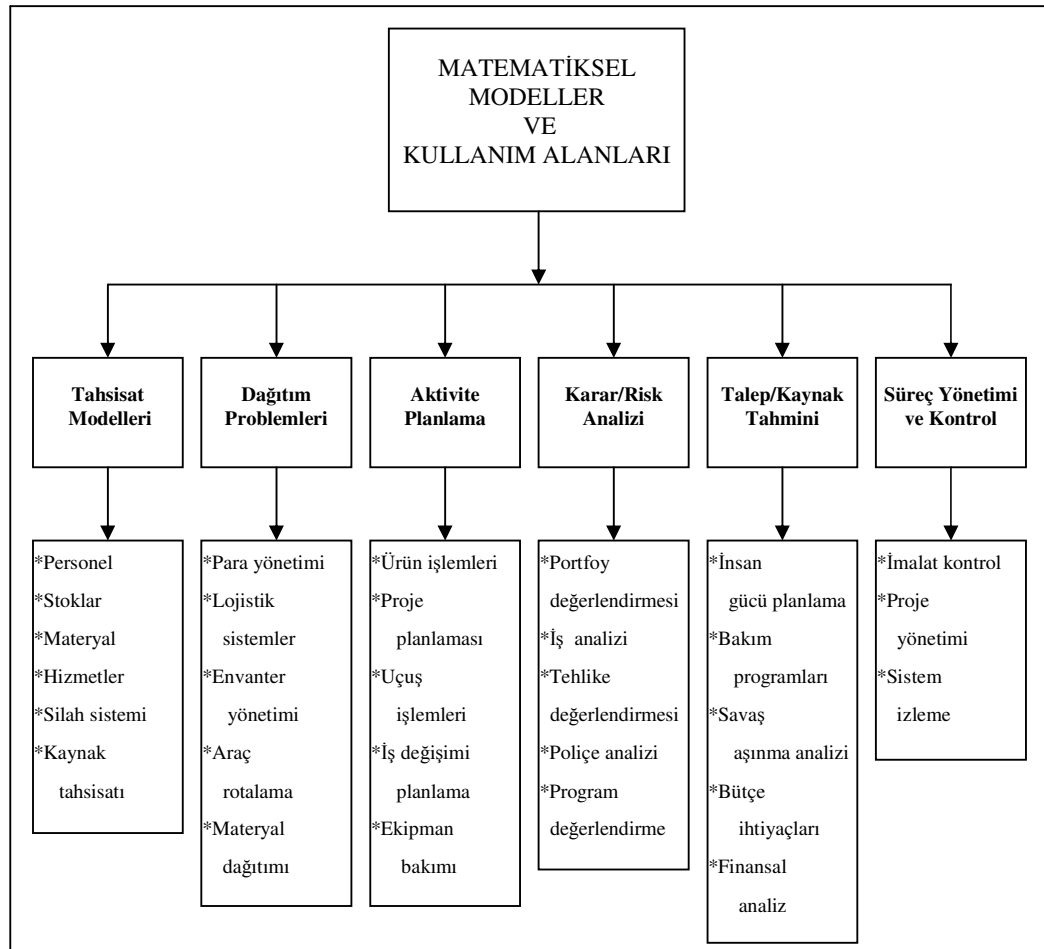
2.5. KD SİSTEMLERİ'NDE KULLANILAN MODELLER

En basitinden en karmaşığına kadar bütün KD Sistemler modellere dayanır. Modellerin amacı, genellikle KD Sistemleri'ni desteklemek ve onu kullanan karar alıcıya, eğer özel seçimler yapmış olsaydı ne gibi bir sonuçla karşılaşacağını tahmin edip sunmaktır. Böylece karar alıcı alternatif durumları denemeye gerek kalmadan bir seçim yapabilecektir. KD Sistemleri çok sayıda model içerebilir. Bu modellerden bazıları standart olup KD Sistemleri geliştirme yazılımları içinde oluşturulur. Diğer modeller ise standart olmalarına karşın fonksiyonlar içinde oluşturulmaya uygun değildir. Bu tip modeller KD Sistemleri ile arayüz kurabilecek, tek başına mevcut olan yazılımlar olabilir. Standart olmayan modeller ise mevcut modellerin

özelliklerinden yararlanılarak oluşturulabilir. Turban (1995)'a göre KD Sistemi'nin kurucusu; hangi modellerin KD Sistemi içinde olması gerektiğini, modellerin oluşturulup oluşturulamayacağı, kullanıma hazır olan modellerin mi yoksa mevcut modellerin biraz değiştirilerek mi kullanılması gerektiğini bir karara bağlamalıdır. KD Sistemleri'nde kullanılan model türlerine, Kesim 4.7.2'de ayrıntılı bir biçimde değinilmiştir.

2.6. MATEMATİKSEL MODEL ÇEŞİTLERİ VE PROBLEMLERİ

Karar problemlerini çözmek için kullanılan matematiksel modeller yapısal benzerliklerine göre farklı kategoriler içinde gruplandırılabilir. Şekil 2.1'de matematiksel modeller ve kullanım alanları gösterilmiştir.



Şekil 2.1. Matematiksel modeller ve kullanım alanları (Davis, 1988).

Tahsisat Modelleri: Karar vericilerin çok sık karşılaştıkları problemlerden bir tanesi, çeşitli programlar karşısında kıt kaynakların nasıl dağıtılacağıdır. Tahsisat

modelleri, yöneticilere mevcut alternatifler arasında bir değerlendirme imkanı vererek en iyi dengeye ulaşmayı sağlar.

Dağıtım Problemleri: Organizasyonlar, nesnelere bir yerden başka bir yere taşırlarken “en iyi yol” belirleme problemi ile karşılaşır.

Aktivite Planlama: Planlama, birbiriyle ilgili aktivitelerin bir ardışıdır. Karar alıcı kaynakların optimum kullanımının, maksimum çıktı düzeylerinin iş yükü taleplerini tatmin etmek için en düşük maliyetli yöntemler sağlamanın vb. sonucu olacak olayların karmaşık bir kümesi için düzen ve planlama belirtmelidir.

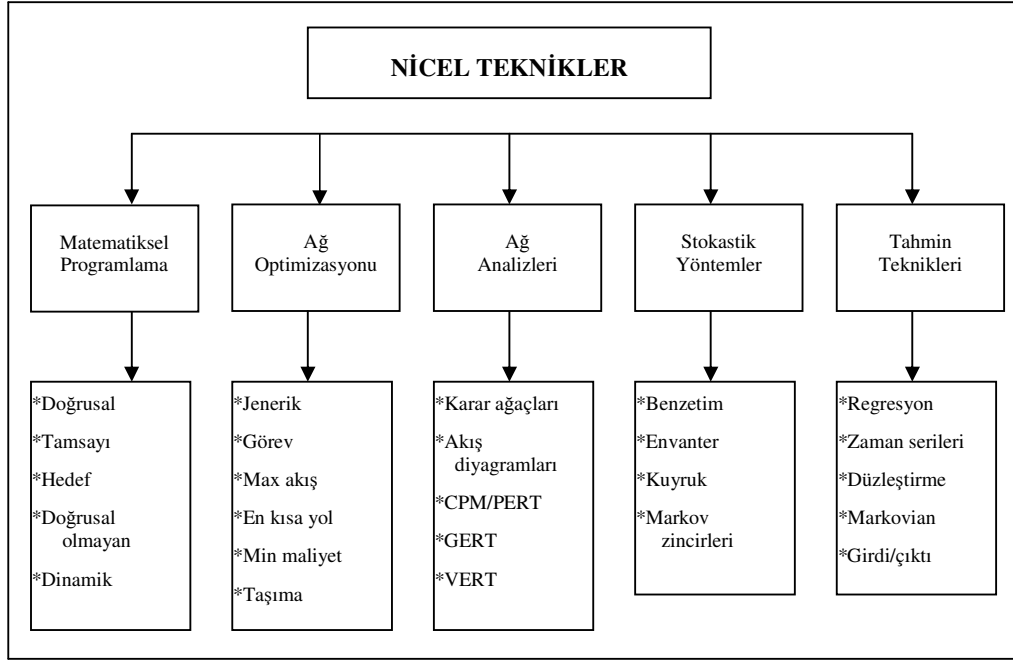
Karar/Risk Çözümlemesi: Yöneticiler genellikle tehlikeler, kazançlar, kayıp fırsatı, zaman kaybı, vb. durumları minimal risk içinde sonuçlandırmayı düşünmek zorundadır. Bu tip problemlerin zorluğu, gerçekleşecek olan mümkün olayın belirsizliğinden kaynaklanır.

Talep/Kaynak Tahmini: Bütün yönetim aktivitelerinde ilk yapılması gereken şey, organizasyonun ne kadarlık bir talep ile karşılaşacağını tahminidir. Talep tahminlendikten sonra sıra talebi karşılamaya yetecek kaynak miktarının tahmin edilmesine gelir.

Süreç Yönetimi ve Kontrol: Yöneticiler yapılan iş süreci boyunca personelin görev, planlama, düzen ve gelişimini izleyebilmeli, koordine edebilmeli ve kontrolü sağlayabilmelidir.

2.7. MATEMATİKSEL TEKNİKLER

Matematiksel teknikler, karmaşık problemlerin çözümlenmesi için gereklidir. Her bir teknik belirli problem çeşitlerini sonuca ulaştırmak için geliştirilmiştir. Şekil 2.2’de en çok kullanılan yöntemlerden bazıları ve bu yöntemlerin daha çok hangi alanlarda kullanıldıkları gösterilmektedir.



Şekil 2.2. Nicel destek için araçlar (Davis, 1988).

Matematiksel Programlama: Kararların çoğu, çok sayıda kaynak sınırlaması ve politik kısıtlamalar altında yöneticinin optimum hareket noktasını bulmalarını gerektirir. Yönetim amaçları (maliyet minimizasyonu, çıktı maksimizasyonu, üretim zamanının azaltılması, vb.) ve zorlama faktörleri (mevcut kaynaklar, vb.) bir denklemler kümesi ile açıklanabildiği zaman matematiksel programlama en etkin yöntemdir. Doğrusal, tamsayı, hedef, doğrusal olmayan ve dinamik programlama belli başlı matematiksel programlama teknikleridir.

Ağ Optimizasyonu: Matematiksel programlamanın çözüm sürecindeki zaman gereksiniminin uzun olduğu durumlarda, ağ optimizasyonu tercih edilir. Bu yöntemde problemler özel bir “ağ” içinde temsil edilip açıklanarak daha çabuk bir şekilde çözümlenir. Jenerik, görev, en yüksek akış, en kısa yol, en düşük maliyet ve ulaştırma ağ optimizasyonu tekniklerindedir.

Ağ Çözümlemesi: Ağ çözümlemesinde genellikle belirlenen hareketin sonucu ve alternatifleri etkileyen çeşitli faktörler ile ilgili belirsizliğin derecesi önemlidir. Ağ çözümlemesi yöntemleri; tehlike değerlendirmesi, kapital yatırımları sınama, büyük ölçekli projeleri sınama ve yönetme, güvenlik çözümlemelerini yapma, vb. durumlarda kullanılabilir. Karar ağaçları, akış diyagramları, CPM (Critical Path

Method)/PERT (Project Evaluation and Review Technique), GERT (Good Enough Reliability Tool) ve VERT ağ çözümlemesi teknikleridir.

Karar ağacı, bir yönetici her biri olayların bir önceki zincirine bağımlı olan seçimlerin karmaşık bir kümesini yapmak zorunda olduğu zaman, alternatif sonuçların tasarlanması ve çözümlenmesi için etkili bir yöntem sağlar. Akış diyagramı, çeşitli olaylar ve aktiviteler arasındaki ilişkinin çok karmaşık olduğu küçük problemler için kullanılır. CPM, kritik yolun bulunmasında; PERT, proje değerlendirmesinde kullanılır. GERT, problemin boyutu büyüdükçe akış çözümlemesi yerine kullanılan yöntemdir. Akış diyagramı ve GERT, olayların ve ilişkilerin matematiksel olarak tahmin edilebilir bir form içinde ifade edilebildiği durumlarda kullanılır. VERT, daha karmaşık problemlerin çözümlenmesinde kullanılır.

Stokastik Yöntemler: Bir sistemin rasgele davranışını tahmin eden yöntemlerdir. Benzetim, envanter, kuyruk ve markov zincirleri stokastik yöntemler arasındadır.

Envanter modelleri, uygun ikmal politikasını oluşturma, envanter maliyetini ve talebe cevap verme zamanını kontrol etmede kullanılır. Bu modeller envanter kararlarını desteklemede önemli bir nicel yöntem sağlar. Kuyruk modelleri, farklı tipteki bir veya daha fazla servisin, müşteri bekleme zamanı veya boşta kalma zamanından birinin uygun olarak oluşturulmasını kolaylaştırır. Markov modelleri, bazı gelecek noktalarında sürecin koşullarıyla ilgili bilginin karar alıcı için önemli olduğu durumlarda kullanılır. Bu modellerde bir durumdan diğerine geçişteki değişimin bilinmesi gerekir. Benzetim, probleme ilişkin analitik çözümlerinin bulunmadığı zaman veya çözümlerinin doğrulukları tatmin edici olmadığı zaman kullanılır. Benzetim sonuçları doğrudan fonksiyonel bilgi sağlamaz.

Tahmin Teknikleri: Organizasyon, işlem veya sistem niteliklerini tahmin etmede kullanılan tekniklerdir. Yaygın biçimde kullanılan tahmin teknikleri; regresyon, zaman serileri, düzleştirme, Markovian ve girdi/çıkı teknikleridir.

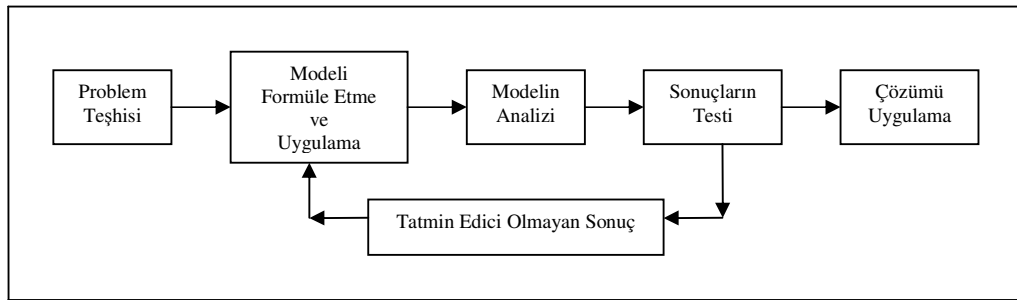
Regresyon, bilinmeyen faktörler için hesaplama ve bir sistemin veya sürecin diğer koşullardaki niteliklerini tanımlamada kullanılır. Zaman serileri, zaman ile ilgili problemleri çözümlenmek için karmaşık bir yöntem sağlar. Zamana bağlı birçok olayın dengesiz bir mizacı olduğundan verinin saçılım grafiği dışında olan anlayışa

parçaların birbiriyle bağlanmasından oluşan bir matematiksel forma dönüştürülür. Oluşturulan matematiksel formülasyon ise uygun bulunan matematiksel teknikler yardımıyla çeşitli algoritmik işlemler sonucunda çözümlenir. Böylece problemin uygulanabilirliği ve analitik çözümü sağlanmış olur.

2.9. PROBLEM ÇÖZÜMLEME SÜRECİ

Bilindiği gibi modeller, karar alıcılara karşı karşıya oldukları problemleri çözmelerinde yardım etmek amacıyla oluşturulur. Çözümü etkili biçimde (en kısa zaman, en düşük maliyet, en az hata) gerçekleştirmek için mevcut problemin yapısına uygun model veya modellerin seçimi önemlidir.

Herhangi bir karar probleminin bir modelle temsil edilmesi mümkün olduğundan, karar alıcının problem çözümü sırasında oluşan olayın görsel modelini oluşturabilmesi gerekmektedir. İşte bu sürece “problem çözümü süreci” denir (Turban ve diğ., 2005). Problem çözümü sürecinde kullanılan birçok model aynı işlevi gerçekleştirebilir veya aynı geçerliliğe sahip olabilir. Şekil 2.4’de problem çözümü sürecinin adımları ve bu adımların birbirleriyle etkileşimleri yer almaktadır.



Şekil 2.4. Problem çözümü süreci (Turban ve diğ., 2005).

Problem çözümü sürecinin ilk adımı olan problemin teşhis edilmesi, sürecin en önemli adımını oluşturmaktadır. Çünkü, çözümü aranan herhangi bir sorunun doğru biçimde problemleştirilmesi veya sorun oluşturan olayların doğru biçimde belirlenmesi, harcanan emek, zaman ve maliyetin en düşük düzeyde olmasına yardımcı olabilmektedir. Çözümlemesi tasarlanan problemi tanımlamak, her zaman kolay bir işlem olmayabilir.

Problem, herhangi bir işlemi yerine getirirken istenen durum ile mevcut durum arasında farklılıkların olmasıyla oluşur. Problemi tanımlamak için problemle ilgili olduğu düşünülen bütün bilgilerin toplanması ve birçok insanla iletişim içinde bulunulması gerekmektedir. Bu şekilde edinilen bilgiler daha sonra ayrıntılı biçimde incelenerek gereksiz görülenler elenmelidir. Böylece temel problem veya problemler üzerine yoğunlaşma sağlanmalıdır.

Problemin iyi ve basit biçimde tanımlanması, çözümünü oldukça kolaylaştırır. Problemi tanımlama işleminden sonra, yapısına göre ölçek, analog, mental ve matematiksel modellerden biri kullanılarak problemin modellenmesi sağlanır. Model formülasyonu olarak bilinen bu adımın temel amacı, problemi daha önceden belirlenen bir modelleme tekniğinin formatına uydurmaya çalışmaktansa, probleme uygun olan modelleme tekniğini seçmektir (Turban ve diğ., 2005). En iyi veya uygun model, mevcut problemin karakteristiklerini veya özünü tam olarak yansıtan model olup aynı zamanda en basit olandır. Problemin karakteristiklerini ise problemi etkileyen bağımsız değişken değerlerinin karar alıcının kontrolü altında olup olmaması, bağımlı değişken ile bağımsız değişkenler arasındaki fonksiyonel ilişkinin iyi tanımlanıp tanımlanmamış olması belirlemektedir.

Problemin çözümü sürecinin bir sonraki adımı, problemi temsil eden modelin çözümlenmesidir. Bu adımda probleme çözüm getirecek alternatifler oluşturulur ve değerlendirilir. Bunun için genellikle birçok senaryo oluşturulur veya çok sayıda “what if?” sorusu sorulur. “What if” çözümlenmesi, matematiksel olmayan modellerle de kullanıma elverişli bir çözümleme türüdür.

Modelin çözümlenmesi sonucunda elde edilen sonuçların testi bir sonraki adımı oluşturmaktadır. Kimi durumlarda model çözümlenmesi sonucunda ulaşılan en son sonuç, problem için bir çözüm olmayabilmektedir. Model çözümlenmesi “what if?” sorularıyla gerçekleştirildiğinden, mümkün çözümlerin her birinin olabilirliğinin ve kalitesinin test edilmesi gerekmektedir. Sürecin test edilmesi aynı zamanda modelin de testini içerdiğinden, problem çözümü süreci içinde kullanılan modelin geçerliliği iki kez sınanmış olur. Bu aşamada oluşturulan alternatiflerden daha iyi bir alternatifin farkına varılması da mümkün olabilir. Bu durum, model için önemli olan bazı varsayımların göz ardı edildiğine işaret eder. Modeli, sonucu bilinen verilerle

test etme, modelin yapısal doğruluğundan ve geçerliliğinden emin olunması için karar alıcılara yol gösterecektir.

Modelin test edilme işlemi gerçekleştirildikten sonra elde edilen sonuçların yeterince iyi bulunmaması durumunda geri dönülüp modelin değiştirilmesi veya geliştirilmesi mümkündür. Bu işlemin ardından süreç, modelin tekrar çözümlenmesi ve elde edilen sonuçların sınanması şeklinde gerçekleşecektir. Test sonucunda elde edilen sonuçların karar alıcı için yeterince iyi olması durumunda elde edilen çözümün uygulanması olan son aşamaya geçilir. Bu aşama genellikle problem çözümü sürecinin en zor aşamasıdır. Çünkü, çözümün uygulanabilmesi için insan gücüne gerksinim duyulur ve çözüm zaman içinde genellikle bir takım yenilikler gerektirebilir. Çözümün uygulanması aşamasında karşı karşıya kalınabilecek bu tip sorunların giderilmesi için problem çözümü sürecinin bütün adımlarında alınan kararları etkileyebilecek birinin bulundurulması yararlı olabilir.

2.10. NEYİ NE ZAMAN KULLANMALI?

Etkili bir KD Sistemi uygulamasında en kritik ve zor işlemlerden bir tanesi, doğru nicel tekniğinin seçimidir. Zira, çok sayıda nicel işlem olmasına rağmen belirli bir problemi çözmek için bunlardan sadece bir tanesi en uygundur. Ne yazık ki en iyi nicel aracı seçmede başvurulabilecek sabit bir kural yoktur. Birçok durumda bir KD Sistemi oluşturmada kullanılan analitik metodoloji, geliştiricilerin deneyimlerine ve eğitimine dayalı olup meslek içinde mevcut olan genel kurallara çok fazla bağlı değildir. Yapılan araştırmalar KD Sistemi kurucusunun seçmiş olduğu yolun gidilecek yöne en iyi yaklaşımı değil, kurucu için en rahat yaklaşım anlayışını içerdiğini göstermiştir. Nicel teknik seçiminde Davis (1988) tarafından önerilen kriterlerin dikkate alınması bu konudaki zorluğu bir ölçüde azaltacaktır. Bu kriterler aşağıda kısaca açıklanmıştır.

Problemin Yapısı: Problemin yapısı, hem alınmış olan kararların hem de kullanılabilir nicel işlemlerin çeşidini belirlemede yardımcı olur.

Karmaşıklık: İçinde bulunulan durumun karmaşıklığı, genellikle alınması gereken en basit yaklaşımdan farklı bir hareket yönü gerektirebilir.

Problemin Büyüklüğü: Problemlerin bazıları çok karmaşık olduğundan geleneksel tekniklerle çözümleri saatler alabilir. Oysa KD Sistemleri'nden beklenen, sonuçların kısa zamanda alınmasıdır. Kullanışlı bir nicel işlemi seçmede problemin büyüklüğü önemli bir rol oynayabilir. Önemli olan, seçilen nicel tekniğin problemi en kısa zamanda çözebilmesidir.

Kesinlik (Doğruluk): Kesin bir tahminde bulunabilecek nicel bir tekniğin kullanılmasıdır.

Yazılım Mevcudiyeti: Kullanılacak tekniğin saptanması, belirlenmiş probleme hangi işlemin en iyi şekilde uyacağına göre değil, bu konuda mevcut olan ticari yazılımlardan hangisine en kolay şekilde ulaşılabileceğine göre gerçekleştirilir.

Tasarımcı Tercih: Bir nicel işlemin seçilmesinde KD Sistemi geliştiricisinin deneyimi ve eğitimi de önemli bir etkidir. Problemin karakteristiğinin nicel işlem seçiminde etkili olmadığı unutulmamalıdır. Tasarımcı kendi tercihinine göre nicel yöntemleri seçebilir.

Yapısal olmayan yönetim problemlerinin çözümü standart modellerle gerçekleştirilemez. Bu tip problemler genellikle taktik veya stratejik problemler olarak sınıflandırılır ve çözümleri için KD Sistemleri'nin kullanılması uygun olur. Yapısal yönetim problemleri ise standart modellerle çözülebilir. Çizelge 2.1'de yapısal yönetim bilimi problemlerinden bazıları ve bu problemlerin çözümü için kullanılacak mevcut modelleme araçları listelenmektedir (Turban ve diğ., 2001).

Çizelge 2.1. Tipik yapısal yönetim bilimi problemleri ve çözüm araçları (Turban ve diğ., 2001).

PROBLEM	ÇÖZÜM ARACI
Kaynak dağıtımı	1. Doğrusal programlama 2. Doğrusal olmayan programlama
Proje yönetimi	1. PERT 2. CPM
Envanter kontrol	1. Envanter yönetim modelleri 2. Benzetim
Kestirim Sonuçları	1. Kestirim modelleri 2. Regresyon çözümlemesi
Bekleme hattı yönetimi	1. Kuyruk teorisi 2. Benzetim
Mal taşımacılığı ve dağıtımı	1. Taşıma modelleri
Parçaları birbirleriyle eşleştirme	1. Tahsis etme modelleri
Satış payını tahmin etme ve aktif biçimde yönlendirilmiş diğer durumlar	1. Markov zinciri çözümlemesi 2. Dinamik programlama 3. Benzetim

BÖLÜM 3

BİLGİ SİSTEMLERİ

3.1.GİRİŞ

Bu bölümde Bilgi Sistemleri (BS) üzerinde durulacak ve BS'nden en yaygın biçimde kullanılan ve en bilinenlerin başında gelen KD Sistemleri'ne giriş yapılacaktır.

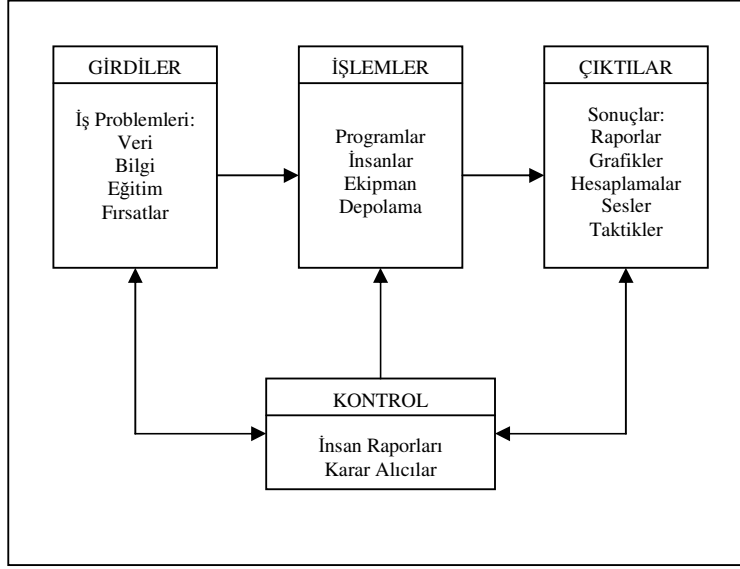
3.2. BİLGİ SİSTEMLERİ NEDİR?

BS, belirli bir amaç için bilgiyi toplayan, depolayan, çözümleyen ve dağıtan parçaların tümüne verilen addır (Turban ve diğ., 1996).

Stair (1992) ise BS'ni, belirlenen amaçlar için gerekli olan veriyi ve bilgiyi geribesleme mekanizmasına benzer biçimde toplayan (girdi), düzeltip depolayan (işleme) ve yayan (çıkıtı), birbiriyle bağlantılı elemanlar veya bileşenler topluluğu olarak tanımlamaktadır.

Bütün sistemlerde olduğu gibi, BS de “girdileri” (veri) ve “çıkıtları” (raporlar, hesaplamalar) içerir. Belli bir çevre (ortam) içinde işlev gören BS, kullanıcılara veya diğer sistemlere gönderilen girdileri ve oluşan çıkıtları işler. BS'nde “girdi”, ham veriyi yakalama ve toplama aktivitesi; “işleme”, ham veriyi kullanışlı çıkıtlara dönüştürme veya kullanışlı çıkıtlarla değiştirme aktivitesi; “geribesleme” ise girdi veya işlem aktivitelerinin düzenlenmesinde veya bu aktivitelerin değişimlerinin gerçekleştirilmesinde kullanılan aktivitedir. Geribesleme, sistemin başarısında önemli role sahip olup yapılan işlemleri kontrol eder.

Turban ve diğ., (1996)'ne göre BS'nin şematik görüntüsü Şekil 3.1'deki gibidir.



Şekil 3.1. Bir bilgi sisteminin şematik görüntüsü (Turban ve diğ., 1996).

3.3. BS'NİN SINIFLANDIRILMASI

BS değişik özelliklerine göre farklı şekillerde sınıflandırılmaktadır. Turban ve diğ. (1996) BS'ni kullandıkları birimin düzeyi, fonksiyonel alanları, mimarileri ve sağladıkları desteğe göre dört sınıfa ayırmış ve bu sınıfların her birini de kendi aralarında alt sınıflarda incelemiştir.

Organizasyonel Düzeylere Göre Sınıflandırma: BS'nin, kullandıkları organizasyon bölümüne göre sınıflandırılmasıdır. Kendi içinde üç sınıfa ayrılır. Bu sınıflar aşağıdaki gibidir.

- Bölümsel Bilgi Sistemleri
- Girişim Bilgi Sistemleri
- Organizasyon İçi Bilgi Sistemleri

Fonksiyonel Alanlara Göre Sınıflandırma: BS'nin organizasyonun önemli işlevlerini gerçekleştiren alanlardaki kullanımlarına göre sınıflandırılmasıdır. Bu sınıflandırma aşağıdaki gibidir.

- Hesaplama Sistemi
- Finansal Sistem

- İmalat (işlem/ürün) Sistemi
- Satış Sistemi
- İnsan Kaynakları Sistemi

Mimarilerine Göre Sınıflandırma: BS'nin kuruluş mimarilerine göre sınıflandırılmasıdır. Alt sınıflar aşağıdaki gibidir.

- Anabilgisayar Tabanlı Sistem
- Tek Başına Kullanılan Kişisel Bilgisayar (PC)
- Dağıtılmış Sistem

Sağladıkları Desteğe Göre Sınıflandırma: BS, farklı organizasyonlarda farklı amaçları gerçekleştirmek üzere kullanılabilir. BS sağladıkları desteklere göre aşağıdaki alt sınıflara ayrılmaktadır.

- İş İşlemleri Sistemleri
- Yönetim Bilgi Sistemleri
- Karar Destek Sistemleri
- Yönetici Bilgi Sistemleri
- Uzman Sistemler
- Ofis Otomasyon Sistemleri
- Grup Destek Sistemleri

Bu kategoride yer alan BS'nin en önemli özellikleri bunların, işin bütün fonksiyonel alanlarında kullanılabilmesidir. GD Sistemleri dışındaki BS'nin; yaptıkları iş, üstlendikleri yükümlülükler, sağladıkları destek ve kullanıcıları bakımından karşılaştırılması Çizelge 3.1'de gerçekleştirilmiştir.

Çizelge 3.1. BS'nin altı çeşidinin karşılaştırılması (Alter, 1992).

BS ÇEŞİDİ	SİSTEMİN YAPTIĞI İŞ	ÜSTLENDİKLERİ YÜKÜMLÜLÜK DERECESESİ	İŞİ KOORDİNE ETMEDE SAĞLADIĞI DESTEK	TİPİK KULLANICILARI
İŞ İŞLEMLERİ SİSTEMLERİ	İşler ile ilgili bilgi toplama ve depolama	Raporları koruma, karar almada işlemleri ve standartları yürütme	Kim gereksiniyorsa, iş verisine ulaşmayı sağlama	İşleri yürüten insanlar
YÖNETİM BİLGİ SİSTEMLERİ	İş işleme sistemlerindeki veriyi bir organizasyon yönetimi ve performans izleme için kullanılan bilgi ile değiştirme	Organizasyonun hedeflerini performans ölçerek ve bunu beklentilerle karşılaştırarak kuvvetlendirme	Performansın ölçümlerini vurgulama	Kendi gerçek işleri hakkında geribesleme kabul eden yöneticiler ve insanlar
KARAR DESTEK SİSTEMLERİ	Bilgi, modeller veya bilginin çözümlenmesi için araçlar sağlayarak insanlara karar almalarında yardım etme	Kullanıcılara, sistemi kullanmada ve kararlar almada akıl verme; bir karar sürecinin evreleri için yöntemler ve formatlar sağlama	Bir kararı çözümlmek ve açıklamak için genel bir iskelet sağlama	Çözümleyiciler, yöneticiler ve diğer profesyoneller
YÖNETİCİ BİLGİ SİSTEMLERİ	Veri çözümleme uzmanlarının yetki gücüne gerek kalmadan veriye etkileşimli formatta ve hızlı bir şekilde ulaşmayı sağlama	Bazen, organizasyonel planlama ve kontrol süreçlerinin yapısal bölümleri olarak kullanılma	Bazen performans ölçümlerini vurgulamada kullanılma	İdareciler ve yüksek düzey yöneticiler
UZMAN SİSTEMLER	Uzmanların bilgilerini diğerleri için elde edilebilir kılmak; uzman bilgiye gereksinim duyulan alanlarda problem çözümüne yardım etme	Karar sürecine rehberlik etme ve tasarlanmış anahtar faktörleri temin etme	Karar almada organizasyonlara yardım etme	Uzman bilginin var olduğu bir alanda problemleri çözen kişiler
OFİS OTOMASYON SİSTEMLERİ	İnsanlara belgeleri ve mesajları işlemede yardım etme; genel ofis işlerini daha etkili ve verimli yapmak için araçlar sağlama	Günlük işleri yapmak için format veya yöntemleri tanımlama veya sınırlandırma; nadiren bilginin hacmini etkileme	Bir organizasyonun her tarafında kullanılabilen araç veya yöntemler sağlama	Ofis çalışanları ve arasıra ofis işini yapan diğer insanlar

Çizelgeki BS ile ilgili açıklamalar aşağıda verilmiştir (Alter, 1992):

İş İşlemleri Sistemleri: Herhangi bir organizasyonda yapılan işler ile ilgili veriyi toplayan ve depolayan sistemler olan İş İşlemleri Sistemleri (İİS), alınmış kararları kontrol etmek amacıyla da kullanılmaktadır. Bilgisayarlaştırılmış ilk BS olarak bilinen İİS daha çok fatura ödemelerinde, çek yazmada veya kredi kartı işlemlerinde kullanılır. Oldukça yapısal olmaları yüzünden İİS'nde kullanılan modeller, işin gerçekleştirilme yöntemine bağlı olarak belirlenir. İİS'nin işleyiş tarzı, belirli formattaki ve uygunluktaki veriyi organizasyonun kurallarına, siyasetine ve hedeflerine göre kullanarak işlemleri gerçekleştirmek şeklindedir. İyi tasarlanmış bir İİ Sistemi, her bir iş için aşağıdaki işlemleri kolaylıkla gerçekleştirebilmelidir.

- Eksik veri tarzındaki hataları bulabilme,

- Verinin göze çarpan en büyük ve en küçük değerlerini bulabilme,
- Veri tabanın diğer veriler ile uyuşmayan verilerini bulabilme,
- Yanlış formattaki verileri bulabilme.

Yönetim Bilgi Sistemleri: YBS, bir organizasyonun yönetimi için gerekli olan bilgiyi sağlayan sistemlerdir. YBS'nin geçmişi bilgisayarların geçmişinden daha eskiye dayanmaktadır. Bilgisayar desteğini de içine alan YBS'nde bilgi toplamanın amaçları; çalışanların performanslarını izleyebilme, çalışanlar arasında koordinasyonu sağlayabilme ve organizasyonların geçmiş işlemleri hakkında bilgi edinebilme şeklinde sıralanabilir. Bilgisayarlaşmış YBS ile İİS'nden edinilen veri özetlenerek, organizasyonu izleme ve yönetmede yöneticilere yardımcı olmak amacıyla sunulur. Ayrıca bu sistemler yöneticilere çalışanların gereksinimlerini ileterek, iş verimliliğinin artmasında da yardımcı olurlar. YBS, performans ölçümüyle ilgili bazı işlemleri gerçekleştirmek suretiyle yönetimin yapısal olmayan görevini az da olsa yapısallaştırır.

Karar Destek Sistemleri: KD Sistemleri, işin nasıl yapılması gerektiğini kesin olarak hiç kimsenin bilmediği durumlarda, insanların kararlar almalarında ve yargıya varmalarında yardımcı olan etkileşimli sistemlerdir. Yarı yapısal ve yapısal olmayan durumlarda karar almayı destekleyecek bilgileri, modelleri veya verileri birleştiren çeşitli araçlar sağlayan KD Sistemleri, mevcut problemi çözerken çeşitli bölümlere ayırarak çözüm için hüküm veya tecrübenin gerektiği yerlerde destek sağlar. Ayrıca, profesyonellerin veya yöneticilerin zamanlarını ve veriyi açıklama çabalarını boşa harcayan karmaşık hesaplamaları veya grafik çizimlerini otomatik olarak gerçekleştirerek, yöneticilerin sadece gerçek iş problemlerine odaklanmalarına da yardımcı olur. KD Sistemleri, tekrar eden veya etmeyen karar alma durumlarını da destekleyebilir. Tekrarlı karar almayı, işlemleri ve formatları tanımlayarak, tekrarlı olmayan karar almayı ise verileri, modelleri ve istenildiği zaman kullanılabilen arayüz yöntemlerini kullanıcılara sağlamak suretiyle destekler. KD Sistemleri'ne Bölüm 4'de ayrıntılı biçimde değinilmiştir.

İİS; rapor koruma, koordinasyon ve organizasyon içinde daha düşük düzeylerde tekrar eden karar almayı kontrol etmeye odaklanmıştır. YBS; yönetim için raporlar sağlar fakat bu raporlar genellikle kullanışlı olmayan yapıda olduklarından

yöneticilerin gereksinimlerine karşılık vermez. Buna karşın KD Sistemleri en az yapısallık içinde çalışan yöneticileri ve profesyonelleri, başarı için net olmayan, rutin durumlarda desteklemeyi amaçlar. KD Sistemleri, verinin görüntülenmesi ve çözümlenmesinde kullanıcıya kontrol edebileceği yöntemler sağlayarak, kullanıcının alternatif kararları formüle edebilmesini ve değerlendirebilmesini sağlar. KD Sistemleri daha iyi kararlar almada düzeltilmiş kullanıcı arayüzlerine, grafiksel ve istatistiksel yöntemlere ve benzetim-optimizasyon modellerine gereksinir.

Yönetici Bilgi Sistemleri: Yönetici Bilgi Sistemleri (YBİS), yönetici ve yönetimlerin işlenmiş sonuçları ve genel iş koşullarını izleyebilmeleri için bilgi girişi sağlayan etkileşimli sistemlerdir. Yönetim Destek Sistemleri olarak da adlandırılırlar.

YBS, yöneticilere problemleri çözerek ve standart formattaki raporları sağlayarak yardım ederken; KD Sistemleri, kullanımı çok fazla tecrübe isteyen araçları sağlayarak, YBİS ise bilgiyi ne zaman isterlerse bulmayı ve işlerine yarayacak en kullanışlı formda elde etmelerine olanak vererek yardım eder.

Uzman Sistemler: İyi tanımlanmış alanlar içinde uzman bilgi gerektiren karmaşık durumların tasarımı, teşhisi veya değerlendirilmesinden sorumlu profesyonellerin entelektüel işlerini destekleyerek kolaylaştıran US, çok çeşitli problemlerde kullanılır. US hastalık teşhislerini, kimyasal çözümlenmeleri, bilgisayar ayarlamalarını, jeolojik veriyi açıklamayı ve daha birçok problem çözümü süreçlerini destekler. US'e Bölüm 6'da ayrıntılı biçimde değinilmiştir.

Ofis Otomasyon Sistemleri: Ofis Otomasyon Sistemleri (OOS), ofisler ve iş organizasyonları içindeki günlük iletişim ve bilgi işleme görevlerini kolaylaştırır. Bu sistemler, kelime işlemciler, hesap tabloları ve sistemleri gibi çok miktarda araç içerir. OOS, işleri genel ofis çalışmaları olan sekreterler, daktilocular ve resepsiyonistler tarafından çok kullanılır. Ayrıca işleri genel ofis görevleri olan mesajlar bırakma, sunumlar hazırlama, vb. olan yöneticiler ve profesyoneller de OOS'ni kullanırlar. OOS'ni oluşturan araçlar üç kategoriye ayrılır:

Kişisel Verimlilik Araçları: Hesap tabloları, kişisel veri tabanı sistemleri, not alma sistemleri, sunum paketleri, çevrimiçi telefon rehberleri, çevrimiçi randevu takvimleri.

Metin ve Görüntü İşleme Sistemleri: Metin veya görüntülü veri içeren belgeleri depolayan, düzelten ve yazan programlardır.

Elektronik İletişim Sistemleri: İletişim ağları elektronik posta, sesli posta (v-mail) ve fax makinalarında kullanılan sistemlerdir.

BÖLÜM 4

KARAR DESTEK SİSTEMLERİ

4.1. GİRİŞ

Tıpkı KD gibi Karar Destek Sistemleri'nin (KD Sistemleri) de genel kabul görmüş evrensel bir tanımı yoktur (Turban, 1993). Genellikle nerede ve ne tür problemlerde kullanıldığına bağlı olarak kabaca, insanlara karar almalarında yardımcı sistemler olarak açıklanabilir. Daha çok, modellemenin esas olduğu karar alma sürecinin birçok görevini otomatikleştirmek için kullanılan (Turban ve diğ., 2001) KD Sistemleri, karar almayı kendi başına gerçekleştiremez (Mallach, 1994). KD Sistemleri'nin temeli kararlara ve karar almaya dayanır. Bu nedenle KD Sistemleri'ni anlamak için ilk önce karar alma sürecinin anlaşılması gerekir.

KD Sistemleri, Bilgisayar Tabanlı Bilgi Sistemleri'nin (BTBS) bir alt kümesidir. BTBS, OOS, İş Yönetim Sistemleri (İYS), YBS ve YDS gibi çeşitli BS'nin genel adlandırmasıdır (Eom, 2001). YDS ise KD Sistemleri, US ve UBS içeren sistemlerdir.

4.2. KD SİSTEMLERİ'NİN GELİŞİMİ

KD Sistemleri, bilgisayarlara dayalı sistemler olduklarından bu sistemlerin gelişimi hiç kuşkusuz bilgisayarların gelişimine paralel biçimde olmuştur. Bu nedenle, KD Sistemleri'ndeki gelişimi anlayabilmek için bilgisayar alanındaki gelişimi incelemek gerekmektedir.

İlk elektronik bilgisayarların kullanımı 1950'li yıllara dayanmaktadır. Bu bilgisayarlarla sayısal verilerin kolayca işlenmesi ve tekrar eden hesaplamaların otomatik biçimde gerçekleştirilmesi sağlanmıştır. Özellikle büyük organizasyonlara yarar sağlayan bu işlem "Otomatik Veri İşleme" veya "Elektronik Veri İşleme" olarak anılmıştır. Otomatik veya Elektronik Veri İşleme terimi, bilgisayarların kullanım alanlarının artması ve verinin işlenmesinin eski bilgilere dayanmaya başlamasından sonra kısaltılarak sadece "Veri İşleme" adını almıştır.

Veri İşleme'nin, organizasyonlar için, vazgeçilmez olmaya başlamasıyla bu işlem bilgisayar uygulamalarının tümünde yerini almıştır. Veri İşleme daha sonraları genel iş olaylarının tekrar eden süreçlerini ve ortak verilerin kaydedilme süreçlerini açıklamak amacıyla "İş İşleme Süreci" ile birleştirilmiştir. Böylece, maaş bodroları, envanter ve faturalama gibi rutin büro işlemleri otomatik biçimde gerçekleştirilmeye başlanmıştır.

Kısa bir zaman sonra yöneticiler, uygun biçimde özetlenmiş iş verisinin aslında potansiyel bir karar alma yeteneğine sahip olduğunun farkına varmışlardır. Bu nedenle, organizasyonunun veri işleme aygıtlarını ve bilgisayarlarında saklı verileri kullanarak çeşitli sorguları gerçekleştirmeye çalışmışlardır.

Önceleri herhangi bir konu hakkında veri elde edebilmenin tek yolu, konuyla ilgili kayıtlı çok miktardaki bilgilerden, amacı gerçekleştirmede yararlı olabilecek verileri elde etmek veya üretmektir. Bilgi Raporlama Sistemi (BRS), bu amacı gerçekleştirmek üzere kullanılan sistemlerden biridir. Ancak BRS'nin, sayfa temelli oluşu ve çok kullanışlı olmaması nedeniyle gelişen teknoloji ile birlikte kullanımı da azalmıştır. Bu sistem yerini YBS'ne bırakmıştır.

1960'lı yıllarda veriyi toplama, düzenleme, saklama ve veriye yeniden erişim işlemlerini gerçekleştirme çabaları sonucunda veri tabanı yönetiminde önemli gelişmeler kaydedilmiştir. İşte YBS bu gelişmeler sonucunda ortaya çıkmıştır. YBS ile büyük miktardaki iş verisini toplayıp özetleyerek yönetim için hayati önem taşıyan bilgilere ulaşmak ve kullanıcı etkileşimli yönetici sorgularını gerçekleştirmek amaçlanmıştır (Eom, 2001). Başlangıçta kullanıcı yanlısı olmamaları ve kullandıkları veri tabanlarının dağınık olmaları yüzünden veri edinimini tam olarak gerçekleştiremeyen YBS, görevlerini tam anlamıyla yerine getirememiştir.

1970'li yıllarda yukarıda sözü edilen sistemlerin en gelişmiş sayılan ve belirli tipteki kararların alınmasına yardımcı olmak amacıyla kullanılan "Karar Destek Sistemleri (KD Sistemleri)" olarak adlandırılan sistemler ile tanışılmıştır. İlk defa Michael S. Scott Morton tarafından sözü edilen KD Sistemleri, "Yönetim Karar Sistemleri" olarak anılmıştır (Scott, 1971). Daha sonra KD Sistemleri, bazı organizasyonlar ve öğrencilerin girişimleriyle geliştirilmeye çalışılmıştır. Bu KD Sistemleri, yarı yapısal veya yapısal olmayan problemleri çözmek amacıyla gerekli veri ve modelleri

kullanmak suretiyle karar alıcılara yardım eden bilgisayar tabanlı sistemlerdi (Sprague, 1980).

KD Sistemleri, son 25 yılda esnek olmayan ana bilgisayar (mainframe) sistemleri olarak bilinen izole edilmiş bilgisayar araçları, alıcı/dağıtıcı veri azaltıcıları ve genellikle organizasyonun intranet'ini kullanan ve büyük ölçüde BT departmanı ile kullanıcı arasında karşılıklı işbirliği gerektiren yüksek performanslı, geliştirilebilen KD uygulamaları ile geliştirilmeye çalışılmıştır (www.doc.ic.ac.uk/~frk/frank/kmt/DSS %20 handout.pdf, 2006).

KD Sistemleri'nin ilk uygulamaları organizasyonlarının satış ve pazarlama verilerini çözmek olmuştur. KD Sistemleri'nin üzerinde geliştirildiği ana bilgisayar sistemleri, bu yıllarda güçlü ve güvenilir olmalarına karşın, henüz kullanıcı isteklerini tam anlamıyla karşılayabilecek yetenekte olmadıklarından, işlemleri gerçekleştirirken zaman kaybına neden oluyorlardı. Bu durum aynı zamanda kullanıcıyla BT departmanı arasında da sürtüşmeye neden oluyordu. Bu soruna bir çözüm oluşturmak amacıyla kullanıcılar 1980'li yıllarda hesap tablolarını ve KD için kişisel veri tabanlarını kullanabilen bilgisayarlar almaya başladı. Bu aslında kullanıcılar için bir engelleme sayılmaktaydı. Çünkü, sadece çok gelişmiş bilgisayarlar KD'i gerçekleştirebilecek araçları ve birleşik halde bulunan veriden gereksinilen veriyi ayıracak yetenekleri içerebiliyordu. Bu tür bilgisayarları edinmek ise yüksek maliyet gerektiriyordu. Bu nedenle uygun olmayan veri ve uzlaşmaz sonuçlar tarzındaki hatalara neden olabilen bilgisayarlar, gittikçe önemlerini yitirmeye başlamıştı.

1990'lı yıllarda kullanıcılar BT departmanından sistemin veri tabanı kısmını yönetmesini istemişlerdir. Böylece iki tabakalı (two-tiered) alıcı/dağıtıcı "veri azaltımı" dönemi başlamıştır. Bu dönemde BT departmanının merkezileştirilmiş bir veri tabanından sorumlu olması, buna karşılık kullanıcıların ise veri tabanı içinde olması gereken bilgisayar araçlarından sorumlu olmaları kararlaştırılmıştır. BT departmanı ve kullanıcılar arasındaki iletişim ise ağlar aracılığıyla gerçekleştirilmiştir. İlk başlarda bu çözüm oldukça iyi sonuçlar vermiş ve her iki tarafı da memnun etmiş olmakla birlikte sonraları bu işbirliği sürecinde kullanıcıların bilgisayar esnekliğine sahip olmaları yüzünden BT departmanının güvenilirlik ve veri tutarlılığını garanti edememesi büyük bir hayal kırıklığı yaşatmıştır.

Yaşanan gelişmeler sonucunda bu kez, küresel yarışın gereksinimlerini karşılama, organizasyonları tek bir çatı altında toplama ve hızlı değişime çabucak uyum sağlama gibi hedefler, aşılması gereken yeni sorunlar yaratmıştır. Bu aşamada üst düzey yöneticiler, KD’i stratejik bir silah olarak kullanmaya başlamıştır. Böylelikle, çalışanların işlerini daha etkili ve doğru biçimde gerçekleştirebilmeleri için doğrudan KD Sistemleri’ne erişebilmeleri ve bu sistemleri kullanabilmeleri gereksinimi ortaya çıkmıştır.

4.3. KD SİSTEMLERİ’NİN ÇEŞİTLİ TANIMLARI

KD Sistemleri kullanıldıkları alanlara göre çok farklı şekillerde tanımlanmıştır. Turban (1995)’a göre, bu farklı tanımlar KD Sistemleri’nin geniş çerçevesini farklı yollarla daraltmaya çalışmaktadır. Aslında çerçevenin daraltılmasıyla, yapılan tanımlamaları birbirine ters düşürmeyecek şekilde uygun bir tanım fonksiyonu kast edilmektedir. Bu tanımların tümünde KD Sistemleri’nin esas amacı olan karar almayı destekleme ve geliştirmeye fazla önem verilmemektedir.

Turban (1995)’a göre KD Sistemleri’nin ideal bir tanımını yapmak için ilk önce KD Sistemleri için yapılan basit tanımdan yola çıkılmalı ve bu tanıma KD Sistemleri’nin karmaşık tanımı eklenmelidir. Turban (1995) basit tanımı şöyle oluşturmuştur: “KD Sistemi, özellikle özel bir yönetim probleminin çözümü için geliştirilen karar almayı desteklemek için oluşturulmuş interaktif, esnek ve uyarlanabilir bilgisayar tabanlı bir bilgi sistemidir. Bu sistem veriyi kullanır, kullanıcı yanlısı bir arayüz sağlar ve karar alıcılara kendi kavrayışları için izin verir.” Turban’ın bu tanıma eklediği karmaşık tanım ise şöyledir: “KD Sistemi, modeller kullanır (ya standart modeller ya da kullanıcı tarafından oluşturulan modeller), bir iteratif süreç sonucunda kurulur (genellikle son kullanıcılar tarafından), karar almanın bütün evrelerini destekler ve bir bilgi tabanı içerir”.

Finlay (1994), KD Sistemleri’nin YBS’ni ve Yönetim Zeka Sistemleri (YZS) olarak adlandırılan zekayı kapsamı gerektiğini vurgulamıştır. Ayrıca Finlay (1994)’a göre KD Sistemleri’nin kişisel olarak kullanılan veya karar alıcı tarafından etkileşimli veya rasgele kullanılan sistemleri sınırlandırmaması büyük önem taşımaktadır. Bu nedenle Finlay (1994) KD Sistemleri’ni karar alma sürecine yardım eden bilgisayar

tabanlı sistemler olarak belirten tanımını tercih etmiş ve bu tanıma ayrıca karar alıcının problemin yapısını algılamış olmasının önemini de katmıştır.

Taylor (1999)'a göre KD Sistemleri'nin YBS'nden farklılığı yönetici veya karar alıcının KD Sistemleri içinde dahili, YBS içinde ise harici bir bileşen olarak yer almasıdır. Yönetici, bir karar almak için kullandığı iteratif diyalog sürecinde BTBS ile etkileşim halindedir.

Shim ve diğ. (2002) KD Sistemleri'ni karmaşık karar alma ve problem çözümlemeyi desteklemede kullanılan tipik bilgisayar teknolojisi çözümleri olarak ifade etmiştir.

“KD Sistemi, bilgisayar temelli bilgi sistemi olup en önemli amacı bilgi kullanan çalışanlarına bildirilen kararlara dayalı olan bilgiyi sağlamaktır” (Mallach, 2000). Bu yolla kullanıcıların durumları çözebilmelerine ve kararlar alabilmelerine böylece işleri daha etkili biçimde yapabilmelerine olanak sağlanır (Poe, 1996).

“KD Sistemleri, yöneticilere bir veri tabanından veriyi çözümleyerek ve sonuçları yöneticiye sağlayarak kararlar almalarında yardım eder” (Nickerson,1998).

“KD Sistemleri, bilgisayar tabanlı bir bilgi sistemi olup insanların kararlara ulaşmalarında yardımcı olur” (Reynolds, 1995).

“KD Sistemleri, genellikle karar alma sürecini desteklemek için geliştirilmiş olan BS'nin bir çeşididir” (Zwass, 1998).

“KD Sistemleri, yapısal olmayan problemleri çözmek için karar alıcıların veri ve modellerden yararlanmalarına yardım eden etkileşimli bilgisayar tabanlı sistemlerdir” (Scott-Morton, 1971).

“KD Sistemleri, kararların kalitesini geliştirmede bireylerin entelektüel kaynaklarıyla bilgisayarın yeteneklerini birleştirir. Böylece, yarı yapısal problemler ile ilgilenen yöneticilerin kararlar alabilmelerinde bilgisayar tabanlı destek sisteminden yararlanmaları sağlanır” (Keen ve Scott-Morton, 1978).

“KD Sistemleri, özellikle yapısal olmayan yönetim problemlerinin çözümünü desteklemede karar almayı kolaylaştırmak için geliştirilmiş etkileşimli, kullanışlı ve yeni koşullara uyurlanabilen bilgisayar tabanlı BS'dir” (Turban, 1995).

“KD Sistemleri, tamamen alışılmış olmayan görevler için yönetsel karar almanın etkinliğini geliştiren sistemlerdir” (Simmons ve Poulos, 1988).

“KD Sistemleri, bilgiyi organizasyonlar için etkili hareketlere dönüştürmede yöneticilere yardım eden bilgisayarlaştırılmış yardımcılardır” (Khoong, 1995).

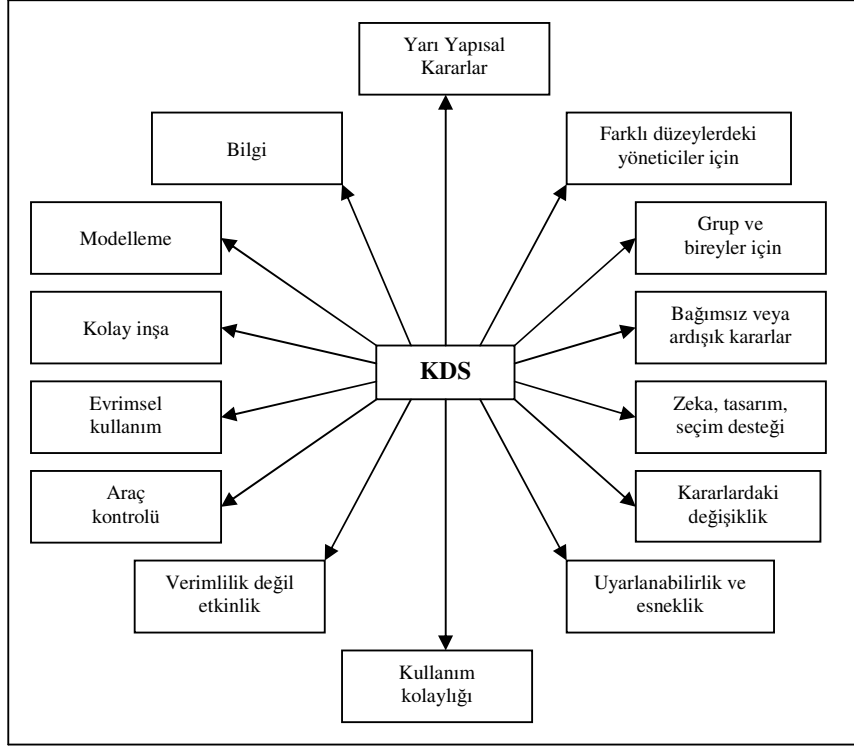
4.4. KD SİSTEMLERİ’NİN KARAKTERİSTİK VE YETENEKLERİ

KD Sistemleri’nin daha önce yapılan tanımlarında KD Sistemleri, yönetici pozisyonundaki karar alıcıları, yarı yapısal karar durumlarında destekleyen, yeteneklerinin gelişmesine yardımcı olan fakat onların yerini alamayan sistemler olarak tanımlanmıştır. Karar alma aslında yargı gerektiren bir süreçtir. Bu nedenle karar almada kullanılan algoritmalar tek başlarına yeterli değildir. KD Sistemleri ile ilgili yapılan ilk tanımlar incelendiğinde sistemlerin bilgisayarlaştırılmış, on-line ve bazı grafiksel çıktı yeteneklerine sahip olmaları gerektiği anlaşılmaktadır (Turban, 1993; Shim ve diğ., 2002). Turban (1995), çeşitli yazarlar tarafından yapılan KD Sistemi tanımlarının dayandığı genel kavramları Çizelge 4.1’deki gibi özetlemiştir. Gorry ve Scott-Morton (1971), Simon’nun (1960,1977) karar türlerini Anthony’nin (1965) yönetim aktivitelerinin düzeyleri ile birleştirerek orjinal KD Sistemi kavramını veya iskeletini oluşturmuştur.

Çizelge 4.1. KD Sistemi tanımlamalarını açıklayıcı kavramlar (Turban, 1995).

KAYNAK	KD SİSTEMİ’NİN İFADE ETTİĞİ TERİM
Gorry ve Scott-Morton	Problem türü, sistem fonksiyonu (destek)
Little	Sistem fonksiyonu, arayüz karakteristikleri
Alter	Kullanım şekli, sistem amaçları
Moore ve Chang	Kullanım şekli, sistem yetenekleri
Bonczek vd.	Sistem bileşenleri
Keen	Gelişme süreci

Çizelge 4.1’deki kavramların, KD Sistemleri’nin karar almayı desteklemek ve geliştirmek olan asıl amaçlarına zıt düştükleri ve bu amaçları önemsemedikleri görülebilir. Turban (1995), KD Sistemleri’nin karakteristikleri ve yetenekleri ile ilgilenmenin çok daha yararlı olacağını vurgulamış ve bunları Şekil 4.1’deki gibi ifade etmiştir.



Şekil 4.1. KD Sistemleri'nin ideal karakteristikleri ve yetenekleri (Turban, 1995).

Turban (1995)'a göre KD Sistemleri'nin karakteristik ve yetenekleri şöyledir:

- Yarı yapısal veya yapısal olmayan bir durumda, insan yargılarının ve bilgisayarlaştırılmış bilginin biraraya getirilmesini destekler. Problem, sadece bir bilgisayar sistemi kullanılarak çözümlenemez. Bu sisteme insan yargıları da dahil edilmelidir.
- Yönetimin her düzeyi için destek sağlanabilir.
- Hem bireyler hem de gruplar için destek sağlanabilir.
- Birbirine bağlı ve/veya ardışık kararların bir çoğuna destek sağlanabilir.
- Karar alma sürecinin evreleri olan zeka, tasarım, seçim ve uygulamanın her biri için destek sağlanabilir.
- Çeşitli karar alma süreçleri ve stilleri için destek sağlanabilir.
- KD Sistemleri değişen koşullara göre değiştirilebilmelidir. KD Sistemleri'ne basit parçaların eklenebilmesi, değiştirilebilmesi, birleştirilebilmesi, yeniden

düzenlenebilmesi ve beklenmedik durumlarda hızlı yanıtlar sağlayabilmesi için ayarlanabilmesi mümkün olmalıdır

- KD Sistemleri özellikle bilgisayar kullanmayı iyi bilmeyen kullanıcılar tarafından da kolayca kullanılabilir. Kullanım kolaylığı interaktif bir tarzı beraberinde getirir.
- Doğruluk, zamanlılık ve kalite unsurlarını içeren karar alma sürecinin etkinliğini geliştirmeyi sağlayabilir.
- Karar alıcı karar alma sürecinin bütün adımları üzerinde tam bir kontrole sahiptir. Sistem karar alıcıyı destekler, fakat yerini alamaz.
- KD Sistemleri öğrenmeye imkan vererek, KD Sistemleri'ni kurma ve geliştirme sürecini başlatır.
- KD Sistemleri'nin kurulumları oldukça kolaydır. Son kullanıcılar kendi başlarına basit sistemleri kurabilir.
- KD Sistemleri'nin modelleme yeteneği, yeni kavramlar ve öğrenme sağlayan farklı biçimlerdeki farklı stratejiler altında çeşitli denemeler yapılmasına imkan tanır.
- KD Sistemleri, farklı BS'ni kullanabilen veya bu BS'ne gerektiğinde başvurabilen BS'dir.
- Zor problemlerin çözümü için bir bilgi bileşeni ile donatılan KD Sistemi, ileri düzey bir KD Sistemi'dir.
- KD Sistemleri; veri tabanlarını, ağları ve programları kullanıcı arayüzleri ile birleştirir.
- KD Sistemleri genellikle oldukça açık sistemlerdir². Oldukça açık olan sistemler, üzerlerindeki etki ve ortamdan dolayı daha karmaşık bir yapıya sahip olabilir. Karmaşıklığını engellemek için sistemin işleyişi süresince bu etkilerin kontrol edilmesi gerekir.

² Açık sistemler, kendi ortamına (çevresine) ve/veya diğer sistemlere oldukça bağlı olan sistemlerdir (Turban, 1993).

Turban ve diğ. (2001) KD Sistemleri'nin yukarıda sıralanan karakteristiklerine, modelin bir bölümünün diğer bölüm üzerindeki etkilerini inceleyen ve “what-if” çözümlenmesi olarak da adlandırılan duyarlılık çözümlenmelerini de eklemiştir.

4.5. KD SİSTEMLERİ'NİN YARARLARI

KD Sistemleri'nin en önemli yararı, karar alıcıların en az hata olasılığıyla hızlı hükümler vermelerine yardım ederek problemi çözmelerini sağlayabilmesidir. Böylece karar alıcının hızlı ve verimli bir şekilde çalışması sağlanmış olur. Turban (1995) ve Mallach (1994)'a göre KD Sistemleri'nin karar alıcıya sağlayabileceği faydalar aşağıda sıralandığı gibidir.

- Karmaşık problemlerin çözümlerini destekleyebilecek yeteneğe sahiptir.
- Karar alıcıları, daha önce alınmış benzer kararlar hakkında bilgilendirerek problemin çözümünü kolaylaştırır. Böylece benzer kararların gelecekteki tutarlılığını artırmayı sağlar. US, kararları duygusal olmayan bir tarzda aldırılmaya yardımcı olarak KD Sistemleri'nin tutarlılığına katkıda bulunabilir. Bu, US ile KD Sistemleri arasındaki çok yakın ilişkilerden bir tanesidir.
- Beklenmedik durumlarda hızlı cevaplar sağlar.
- Birçok farklı stratejiyi farklı biçimlerde deneme yeteneğine sahiptir.
- Öğrenme veya öğretmede yeni anlayışlar sağlar. Bunun için bir uzman sistem bileşeninden yararlanılabilir. US'in çoğu, kullanıcıya sistemden ne tür bir sorun için özel bir öneride bulunmasını istediğini soran ve yanıtını teknik olmayan ifadelerle alabilen bir arayüz sağlar. Kullanıcı bu arayüz sayesinde önerilen çözüme ulaşmada uzmanların izledikleri düşünce tarzını kavrayacaktır. Böylece kullanıcı için öğrenme gerçekleşecektir.
- İletişimi kolaylaştırır. “What-if” çözümlenmesi kuşkuyu gidermede ve takım çalışmasını geliştirmede kullanılabilir.
- Organizasyon içindeki personelin iletişimini kolaylaştırır. “Kötü kullanım” olarak adlandırılan, gelecekte üstlenilebilecek belirli bir hareketi örneklemede veya “savunucu kullanım” olarak adlandırılan geçmişte uygun bir şekilde üstlenilmiş

belirli bir hareketi örneklemede kullanılan bir ikna aracı olabilir. “GroupWare”, grup halinde karar almayı sağlamak için tasarlanmış bir KD Sistemi’dir. Elektronik mail, ilan tahtaları veya elektronik olarak görüşmeyi sağlayan diğer araçlar bu tür KD Sistemleri’ne örnek olarak verilebilir.

- KD Sistemleri’ni rutin bir uygulama olarak kullanmak yanlış kararların doğurduğu maliyeti giderecektir.
- KD Sistemleri, yönetsel etkinliği geliştirmek ve yöneticilere herhangi bir görevi daha kısa bir zamanda ve/veya daha az bir çabayla gerçekleştirmelerine olanak sağlamak suretiyle objektif kararlar almayı kolaylaştırır.
- KD Sistemleri organizasyonel kontrolü artırır. KD Sistemleri, bireysel olarak alınan kararların organizasyonun normlarına, ana hatlarına veya iskeletine uygun biçimde alınmasını isteyebilir. Bunun için organizasyon içinde bir tutarlılık düzeyi belirlenebilir. Ayrıca bireysel olarak alınan kararlar, yöneticilere rapor edilebilir ve daha sonra bu raporlar çalışanların verimliliğini ölçmede kullanılabilir.

Doğru ve güvenilir kararlar almak için bilginin mevcut olması gerekmektedir. Karar almada bilginin elle oluşturulması aşağıda sıralanan eğilimlerden dolayı gittikçe zorlaşmaktadır (Turban ve diğ., 2001).

- Bir karar içinde olması gereken alternatif sayısının; iletişimdeki gelişmeler, küresel satışlara ulaşılabilirlik, internet ve elektronik ticaret kullanımıyla artış göstermesi.
- Kararların çoğunun belli bir zaman kısıtı altında alınması.
- BT ve sağladığı karmaşık çözümlenin, karar almanın önemli bir faktörü olmaya başlaması.
- Bilgiye hızlı biçimde erişimin, uzmanlara danışma veya grup halinde karar toplantısı yapma suretiyle sağlanması.

4.6. KD SİSTEMLERİ’Nİ OLUŞTURAN ARAÇLARLAR

KD Sistemleri, veriyi bilgiye, bilgiyi de yararlı bilgiye dönüştürme sürecini gerçekleştiren sistemler olup içlerinde çok çeşitli sistemler, araçlar ve teknolojiler

bulundurur (Rhia, 2000). Rhia (2000), KD Sistemleri araçlarını, farklı gereksinimleri karşılayan dört kategoride incelemiştir. Bu kategoriler aşağıdaki gibidir.

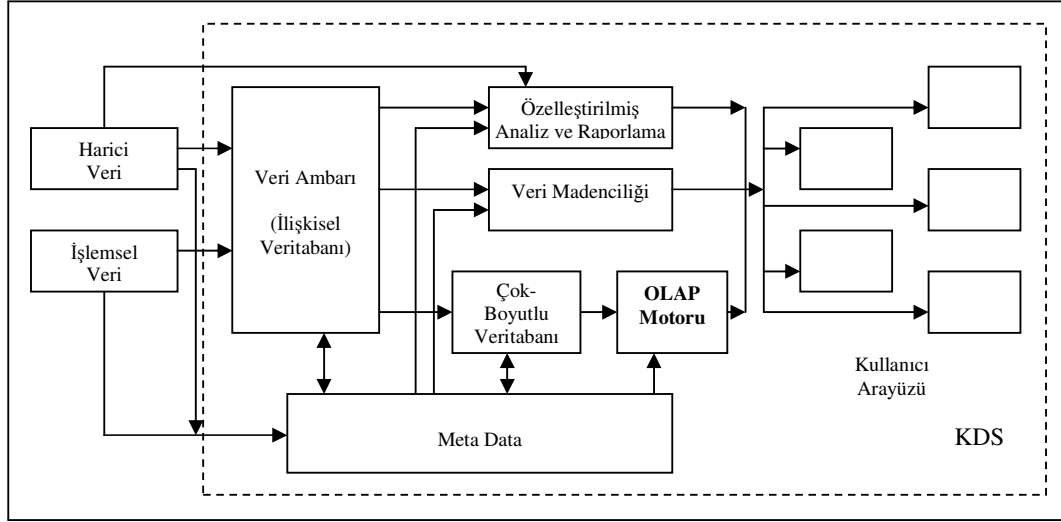
- Sorgu ve raporlama araçları
- OLAP araçları
- YBS araçları
- Veri madenciliği araçları

Modern geniş kapsamlı KD Sistemleri'nin içerdiği araçlar şunlardır:

- Veri ambarı (ilişkisel veri tabanı)
- Meta data
- Özelleştirilmiş çözümlene ve raporlama
- Veri madenciliği
- Çok boyutlu veri tabanı
- OLAP motoru
- Kullanıcı arayüzleri

KD Sistemleri'nde bu araçların tümü olabileceği gibi sadece birkaçı ya da ilave araçlar da bulunabilir. KD Sistemleri'nin görevini yerine getirebilmesi için bu araçların etkileşimli biçimde çalışması gerekmektedir. Etkileşimli çalışma için veriye gereksinim vardır. Verinin, KD Sistemleri'ne girişi harici ve işlemsel biçimde gerçekleşmektedir.

Yukarıda sıralanan araçların tümünün yer aldığı, modern geniş kapsamlı bir KD Sistemi'nin şematik gösterimi Şekil 4.2'deki gibidir ([www.doc.ic.ac.uk/~frk/frank/kmt/DSS %20handout. pdf](http://www.doc.ic.ac.uk/~frk/frank/kmt/DSS%20handout.pdf), 2006).



Şekil 4.2. Modern geniş kapsamlı bir KD Sistemi.

Veri Ambarı: İşlemsel veri tabanlarından farklı bir özelliğe sahip olan veri ambarı, KD’i optimize etmek amacıyla geliştirilmiş nesne tabanlı, bütünleşik, zamana göre değişim gösteren ilişkisel ya da çok boyutlu veri tabanlarının değişmeyen bir parçasıdır (Eom, 2001). Veri ambarları içinde, harici ve dahili veri kaynaklarından edinilmiş çok miktardaki veri bulunabilir.

Çözümleme ve Raporlama: Çözümleme araçlarının birçoğu matematiksel ve istatistiksel modelleri uyguladıktan sonra bunlara ilişkin çıktılar grafikler ve tablolar içeren raporlar biçiminde kullanıcılarına sunabilme yeteneğine sahiptir. Raporlara ek bilgiler, başka kaynaklardan eklenebileceği gibi raporun sadece kullanıcının gereksindiği kısmı da alınabilir.

Meta Data: Meta data, veri hakkında veri olup veri ambarının bütün bileşenlerini ve görünüşlerini birarada tutmaya yarayan adeta bir tutkaldır (www.doc.ic.ac.uk/~frk/frank/kmt/DSS%20 handout. pdf, 2006). Meta data’nın bir diğer görevi ise niteliklerin, boyutların, metriklerin, hiyerarşilerin, tabloların ve kolonların dizinlerini oluşturmaktır.

Veri Madenciliği: Veri keşfi olarak da bilinen veri madenciliği, veri içinde gizlenmiş modelleri, eğilimleri, sınıfları, kavrayışları, ilişkileri keşfeder (Eom, 2001). Büyük veri tabanlardan, veri ambarlarından veya herhangi bir bilgi kaynağından istenilen bilgiyi arayarak bulmaya çalışır. Veri madenciliği, modelleri tanımlamada

veya tahmin modelleri oluşturmada istatistiksel veya kural tabanlı yöntemleri kullanabilir (www.doc.ic.ac.uk/~frk/frank/kmt/DSS%20handout.pdf, 2006).

Çok Boyutlu Veri Tabanı: Çok boyutlu veri tabanı, her biri çok boyutlu uzayda bir noktayı temsil eden nesnelerin bir topluluğu olarak tanımlanabilir (Shoshani, 1997). Veri tabanlarının içerdikleri verilerin birçoğu çok boyutlu yapıya sahiptir. Veriyi çok boyutlu uzayda göstermenin birçok avantajı vardır. Bunlardan bazıları; verinin sahip olduğu bazı özelliklerin daha kolay görünmesi, çok boyutlu yapıya kendiliğinden sahip olan verilerin çok boyutlu uzayda gösterilmesinin daha kolay olması, boyut kategorilerinin çapraz çarpımını kolaylaştırma, veri üzerinde özetlemenin yapılabilmesi şeklinde sıralanabilir.

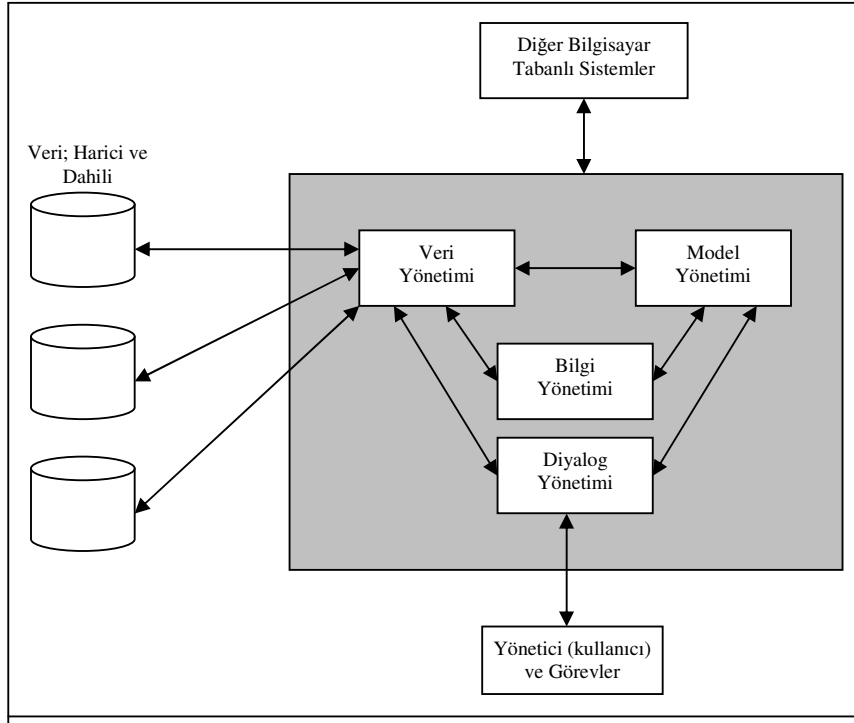
OLAP: OLAP, uzman kişinin sisteme olan bakış açısını sistemi kullanan kişilerin anlayacağı biçimde açıklayarak, ham veriden dönüştürülmüş bilgi üzerinde mümkün olan incelemelerin büyük çoğunluğuna hızlı, tutarlı, interaktif biçimde erişip çözümleyicilere, yöneticilere ve uygulayıcılara veriden çeşitli kavrayışlar sağlayan bir yazılım teknolojisi kategorisidir (Vassiliadis ve Sellis, 1999). OLAP sisteminin kural veya gereksinimleri, çok boyutlu kavramsal görünüş, saydamlık, erişilebilirlik, tutarlı raporlama performansı, alıcı/dağıtıcı mimarisi, genel boyutsallık, dinamik seyrek matris yönetimi, çok kullanıcı desteği, sınırlandırılmamış çapraz boyutsallık işlemleri, sezgisel veri kullanımı, esnek raporlama, kısıtsız boyut ve toplam düzeyleri şeklinde sıralanabilir (Codd E.F. ve diğ., 2006).

4.7. KD SİSTEMLERİ'NİN MİMARİSİ

KD Sistemleri, BS'nin özel bir türüdür (Mallach, 1994; Turban, 1993; Turban ve diğ., 2001). KD Sistemleri'nin bileşenleri aşağıda sıralanmıştır. Bu bileşenler, KD Sistemleri'nin yazılım bölümünü oluşturmaktadır. Alt sistemlerin birbirleriyle olan iletişimleri ve aynı zamanda KD Sistemleri'nin kavramsal modeli Şekil 4.3'de gösterildiği gibidir.

- Veri Yönetim Alt Sistemi
- Model Yönetim Alt Sistemleri
- Diyalog Alt Sistemi veya Kullanıcı Arayüzü

- Bilgi Yönetim Alt Sistemi



Şekil 4.3. KD Sistemleri'nin kavramsal modeli (Turban, 1995).

4.7.1. VERİ YÖNETİMİ ALTSİSTEMİ

Bir KD Sistemi içinde, gerekli bilginin sağlanması için veri depoları olarak adlandırılan bir veya birden fazla veri tabanı, dosya kümesi ve/veya veri ambarı kullanılabilir. Bu veri depolarından bazıları KD Sistemi içinde bazıları ise harici veri kaynakları biçiminde kullanılabilir. Veri tabanlarından bazıları, KD Sistemleri uygulamalarında kendi Veri Tabanı Yönetim Sistemleri (VTYS) ile birlikte başka BS tarafından kullanılıp, korunurlarken yani ayrıık bulunabilirlerken bazıları ise ayrıık olmayabilirler.

KD Sistemleri için veri, geleneksel veri tabanlarından ve çeşitli harici, dahili ve kişisel kaynaklardan elde edilebilir (Turban, 1995). Dahili veri, organizasyon içindeki veri tabanlarını kapsar. Bu veri tabanlarından bazıları organizasyon içinde kullanılan diğer BS ile yürütülürler. Harici veri, ticari veri tabanlarından toplanan veriden, algılayıcılar veya uydular tarafından edinilen bilgiye kadar geniş bir yelpaze içinde, organizasyonun dışında bulunan kaynaklardan toplanan veridir. KD Sistemi uygulamalarının çoğunda veri, “veri ambarı”ndan elde edilir. Bir veri ambarı KD

Sistemleri'nin gerektirdiđi veriyi çeřitli kaynaklardan toplayıp kiřitsel bir veri tabanı biçiminde düzenleyebilir (Turban ve diđ., 2001). Kiřitsel veri, kullanıcının deneyimini ieren veridir. Kiřitsel veri ile rakiplerin ne yapabileceđi tahmin edilebilir.

Bazı harici veriler organizasyona, düzgün bir kaynak üzerindeki elektronik veri deđiřimi yoluyla veya bir organizasyondan diđer organizasyona diđer veri aktarım yollarıyla ulařabilir. Ayrıca birçok veri de internet aracılıđıyla sađlanabilir.

Ham veri, elle veya araç ve alıcılar tarafından toplanabilir. Veri toplama iřlemi tamamlandıđında bu iřlemin hangi yolla gerekleřtirildiđine bakılmaksızın verinin geerli kılınması gerekir. KD Sistemleri iin veri kalitesi ve güvenilirliđi ok önemlidir. Bu nedenle karřılařılabilecek çeřitli sorunların önüne gemek iin veri kalitesini sađlayan çeřitli korumalar tasarlanmıřtır (Turban ve diđ., 2001).

Büyük KD Sistemleri'nde veri ile ilgili ařađdaki problemler ile karřılařılır.

- Dođru olmayan veri
- Uygun olmayan veri
- Uygun biimde ölçülmemiř veya indekslenmemiř veri
- Ařırı veri gereksinimi
- Mevcut olmayan veriye gereksinim

Kurulan KD Sistemi'nin ayrı veri tabanlarının kurulumunu gerektirmesi durumunda sistem kurucusunun gerekli olan veriyi tasarlaması ve hazırlaması gerekmektedir. Veri; iliřkisel veri tabanı, hiyerarřik veri tabanı, ađ ve nesne yönelimli veri tabanı biimlerinde düzenlenebilir. Düzenlenen veri tabanlara eriřim, alıcı/dađıtıcı gibi teknolojileri kullanan ađlar aracılıđıyla gerekleřebilir. Birok organizasyon geniř-giriřim (wide-enterprise) türü veri tabanları geliřtirmektedir. İliřkisel Veri Tabanı Yönetim Sistemi (İVYS), KD Sistemleri iin daha uygun veri tabanı türüdür. ünkü bu tip veri tabanları, belgeleri diđer dosyalardaki belgelerle birleřtirmek iin önceden tanımlanmıř bađlantılar gerektirmez. Bu özellik sayesinde İliřkisel Veri Tabanları'ndaki (İVT) veriye yeniden eriřimde büyük bir esneklik sađlanır. İVT'nın

Yapısal Sorgu Dili (Structured Query Language, SQL) olarak adlandırılan standart bir arayüz kullanmaları bir başka avantajıdır.

4.7.2. MODEL YÖNETİMİ ALTSİSTEMİ

Bu alt sistem, nitel ve nicel modelleri içeren ve yöneten bir yazılımdan oluşmaktadır. Nicel modeller, sistemin analitik yeteneklerini sağlamaktadır.

Kurulması düşünülen bir KD Sistemi'nin bir model üzerinde sınanmasıyla düşük deneme maliyeti, zamandan tasarruf, hata maliyetinde azalma, modelin kendi kendini işletebilmesi, öğrenme desteği ve eğitimde artış sağlanır. Model yönetim alt sistemi, aşağıda ayrıntılı biçimde incelenecektir.

4.7.2.1. KD Sistemi Modelleri

Mallach (1994)'a göre KD Sistemleri, bilgi tabanlı model olarak da adlandırılan modellerin dördüncü türü olan sembolik/matematikselsel modelleri kullanır. Gerçek, bilgi gibi işlenebilen veya yorumlanabilen veri ile gösterilir. Bu modelde kullanılan veri elemanları, bilgisayar veya programlama dilleri tarafından algılanabilecek türler olan doğru/yanlış yapısını içeren değerleri, karakter dizgileri veya sayısal değerleri içerir.

Sembolik model, veri elemanlarını kullanan prosedürleri ve formülleri birleştirir, yeni veri elemanlarına değerler türetir. Model, veri tabanından, kullanıcılardan veya diğer bilgi sistemlerinden elde edilen harici bilgileri kullanabilir.

KD Sistemleri'nde farklı türdeki çok sayıda matematikselsel modelin birleştirilmesi gerçekleştirilebilir. KD Sistemi kurucuları genellikle KD Sistemi içinde olması gereken modelleri belirlerken, yeni modeller oluşturmanın gerekip gerekmediği, hazır modelleri kullanmanın yeterli olup olmadığı ve mevcut modeller üzerinde değişiklik yapmanın gerekip gerekmediği sorunlarıyla karşı karşıya kalır.

4.7.2.2. KD Sistemi'nin Model Yönetim Alt Sistemi

Model tabanındaki modeller dört kategoriye ayrılır. Bu kategoriler aşağıdaki gibidir:

- Stratejik modeller
- Taktik modeller

- İşlemsel (operasyonel) modeller
- Model yapı blokları ve altyordam modelleri

Stratejik modeller, üst düzey yöneticilerin stratejik planlarını (ortak amaç geliştirme, fabrika yeri seçimi, vb.) desteklemek amacıyla kullanılan modellerdir. Bu modellerin esas amacı, organizasyonun çevresiyle uyum içinde olmasını sağlamaktır (Finlay, 1994). Bilgi, zekanın üretimi için gerekli olan ham maddeyi düzenler. Girdi, harici veri tabanları, harici söylentiler, dahili biçimsel raporlar ve biçimsel görüşmeler gibi çok çeşitli kaynaklardan elde edilir. Üst düzeyde veri doğruluğuna ulaşmak, genellikle mümkün değildir. Burada ilgilenilen şey, yanlış yapmamaya özen göstermektir. Stratejik planlama periyodik olarak gerçekleştirilir ve bu periyot genellikle bir aydan daha kısa değildir. İyi yönetilen bir organizasyonun beklenmedik durumlarda uygulamaya koyabileceği çok sayıda planının olması gerekir.

Taktik modeller, çoğunlukla orta düzey yöneticiler tarafından kullanılan modellerdir. Bu modeller aracılığıyla satış promosyonları ve teçhizat düzeni ayrıntıları gibi organizasyonel kaynakların dağıtım ve kontrolü sağlanabilir. Taktik modeller, genellikle muhasebe bölümü gibi bir alt sisteme uygulanabilir.

Operasyonel modeller; borçlanmayı onaylama, öğrencilere önerilerde bulunma ve stok kontrolü gibi organizasyonel aktiviteleri günü gününe desteklemede kullanılır. Bu modeller genellikle tam ve kesin olan dahili veriyi kullanır ve organizasyonun çevresiyle etkileşiminde tampon görevi üstlenir.

Model yapı blokları ve alt iş programı modelleri; rastgele sayı üretim jeneratörünün gerçekleştirdiği işlemleri, hesaplamalı iş programları veya regresyon çözümlemesi yoluyla yapılan işlemleri gerçekleştirir. İş programları kendilerine ait modellerde veya daha büyük modellerde bir bileşen gibi bulunabilir.

Model tabanındaki modeller, fonksiyonel alanlara göre finansal modeller veya üretim kontrol modelleri şeklinde, bağlı buldukları bilim dalına göre ise istatistiksel veya yönetim bilimi modelleri olarak sınıflandırılabilir. Bir model tabanındaki modellerin sayısı birkaç tane olabileceği gibi birkaç yüz tane de olabilir. Model tabanında bulunan modeller standart olarak bulunan modeller olabileceği gibi COBOL veya

dördüncü kuşak diller (fourth-generation languages, 4GL) ya da özel modelleme dilleri gibi yüksek düzey diller aracılığıyla da oluşturulabilirler.

Verinin yönetilmesi gerektiği gibi modellerin de yönetilmesi gerekmektedir. Model yönetimi, Model Tabanı Yönetimi Yazılımları (MTYY) aracılığıyla gerçekleştirilir (Turban, 1993). MTYY birbirleriyle ilgili modelleri veri tabanlarına uygun bağlantılarla bağlayabilme yeteneğine sahiptir. MTYY'nın, aşağıda sıralanan özellikleri sağlaması gerekmektedir (Turban, 1995).

Kontrol: KD Sistemi kullanıcısı, kontrolü gerçekleştirmek üzere çeşitli seçeneklere sahip olmalıdır. Sistem, gerçekleştirilmesi düşünülen uygulama için uygun modellerin seçimini elle gerçekleştirildiği zamanki kadar iyi bir biçimde ve tamamen otomatik gerçekleştirebilmelidir. Bu yolla kullanıcı, mevcut bir problemi çeşitli yollardan çözümlene seçeneklerini uygulayıp sonuçları karşılaştırabilecek çok rahat bir ortam elde etmiş olur. Subjektif bilginin sisteme girişi yapılabilir. Gereksinilen bilginin tamamı sistemde bulunmayabilir. Bu nedenle kullanıcı denemelerini gerçekleştirirken “what-if” ve hedef arama gibi duyarlılık çözümlenmelerini uygulamalıdır.

Esneklik: KD Sistemi, bir problemi çözümlene süreci (bir oturum) boyunca gerektiğinde modelleme yaklaşımlarını değiştirebilme yeteneğine sahip olmalıdır.

Geribildirim: Model, herhangi bir aşamadaki problem çözümlene sürecinin durumunu kullanıcıya gösterebilecek yeterli düzeyde bir geribildirim sağlayabilmelidir.

Arayüz: Özel model kullanımıyla KD Sistemi kullanıcısının rahat hissetmesi sağlanabilir. Kullanıcıdan edinilen bilgi, sistemde olabildiğince minimum düzeyde ve isteğe bağlı olarak saklanmalıdır.

Gereksizliği Azaltma: Paylaşımlı modellerin kullanımıyla gerçekleştirilebilir.

Tutarlılığı Artırma: Aynı model veya bilginin birçok kullanıcı tarafından kullanımı sağlanabilmelidir.

Yukarıda sıralanan özelliklerin sağlanabilmesi için MTYY aşağıdaki işlemlerin gerçekleştirilebilmesine izin vermelidir.

- Mevcut modellere erişebilme ve yeniden düzenleyebilme
- Mevcut modelleri deneyebilme ve uygulayabilme
- Mevcut modelleri saklayabilme
- Yeni modeller oluşturabilme

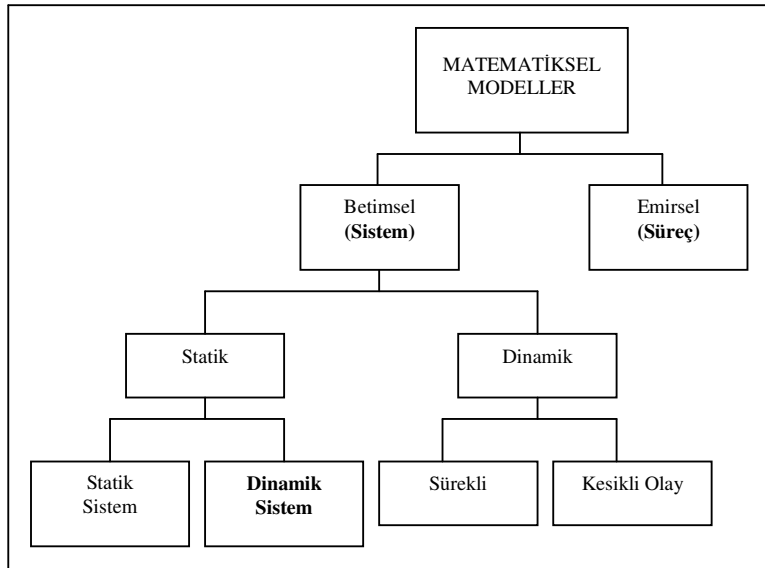
Modeller, veri tabanı dizinine benzer bir model dizini kullanılarak listelenir. Model yönetimi,

- Modelin gerçek çalışmasını kontrol etme ve model birleşimi
- İhtiyaç duyulduğunda birçok modelin işlevlerini birleştirme

gibi işlevleri içeren, model uygulama aktivitelerini kontrol edebilmelidir. Ayrıca bu tip bir modelden elde edilen sonuçları da çözümleyebilmeli ve kestirebilmelidir.

4.7.2.3. Model Tabanındaki Modellerin Farklı Sınıfları

Mallach (1994) modelleri “sistem” veya “süreç” modelleri olmak üzere farklı bir biçimde de sınıflandırmıştır. Şekil 4.4’de Mallach (2000)’ın matematiksel modeller için düzenlediği aile ağacı yer almaktadır.



Şekil 4.4. Matematiksel modellerin “Aile Ağacı” (Mallach, 2000).

Sistemi ihtiyaç duyulan çalışma tarzına göre düzenleyen modeller, sistem modelleridir. Bu modeller hem problemi çözmeye kullanılacak doğru formülleri ve algoritmaları hem de KD Sistemleri tarafından alınabilecek farklı türdeki kararlara uyacak farklı matematiksel modelleri içermelidir.

Süreç modeller ise insanların, sistem ile ilgili bir karar alırken izledikleri süreci düzenleyen modellerdir. Süreç modeller birçok uzman sistemin temelini oluştururlar (Mallach, 1994). Bu modeller, özel bir alandaki belirli bir sorunu saptayacak uzmanlar tarafından kullanılan önemli kuralları içerir. Süreç modeller çözüme daha az adımda ulaşılmasını sağladıklarından bu anlamda verimlilik üst düzeydedir. Süreç modellerin şimdi ve yeni arasında kesin ilişkiler olduğu varsayımına dayanan özel kuralları nedeniyle, bu modellerin beklenmedik durumlara kolaylıkla uyarlanabilmesini sağlayacak derin bilgi gereksinimini içermesi söz konusu değildir. Bunun yerine herhangi bir beklenmedik durum oluştuğunda model, formül değerlerini verilen duruma uyarlayacak ve böylelikle doğru yanıtı sağlayacaktır. Süreç modeller girdi verilerine bağlı olarak betimsel modeller ile emirsel veya tahminsel modeller olmak üzere sınıflandırılabilir.

Emirsel veya tahminsel modeller, hayali veya istatistiksel olarak tanımlanan geleceğe ilişkin veriyi kullanarak model kurucuların gelecekteki beklentilerinin ne olduğunu anlatan modellerdir.

Betimsel modeller, sistemin geçmişte ne yaptığını veya bulunulan zamanda ne yapıyor olduğunu anlatan modellerdir. “Statik” veya “dinamik” olabilirler (Mallach, 1994; Turban 1995). Statik modeller, dengedeki bir sistemin değerlerini göstermek için kullanılır. Bir statik model ya statik bir sistemi ya da dinamik bir sistemin enstantanesini (snapshot) modeller. Statik modellerle herhangi bir zaman dilimindeki veri değerleri arasındaki ilişkiler açıklanabilmektedir. Statik modeller, sürekliliği uzun veya kısa olan tek bir zaman aralığına dayanır. Herhangi bir veri elemanı statik modellerle bir kez hesaplandıktan veya işlendikten sonra, sistemde bu haliyle kalır. Karar alıcı girdiyi değiştirmedeği sürece sonuç değişmeyecektir. Dinamik modeller, bir periyot üzerindeki sistemin değişimini neden-sonuç ilişkilerini kullanarak gösterir. Periyot, bir dizi neden-sonuç ilişkisi ile bir sonraki periyoda bağlanır. Bir dinamik modelde veri değerleri zamana bağlı olarak değişir. Değişim dinamik modeller için gereklidir ve zamana bağlı olarak gerçekleşir. Zamanın

herhangi bir anında meydana gelen bir olay, modellenmiş sistemi gelecekteki anlarda etkileyecektir. Bu ise zamana bağlı eğilimlerin ve modellerin bir göstergesidir. Statik modeller, zamana bağlı değişim miktarını göstermek için kullanılabilir. Farklılık zamana bağlı verinin gösteriminden kaynaklanır. Eğer birden fazla zaman periyodu sonuçları modelde gösterilmişse ve model içinde periyottaki gelişim ve değişimi etkileyen formüller mevcut ise kullanılan bu model dinamiktir. Bazı kararlar için kullanılan veri değerlerinde değişim söz konusu değildir. Bu tür kararların alımında bir ya da bir dizi statik model kullanımı yeterli olacaktır. Dinamik modeller, zamana bağlı olarak sistemin davranışlarını taklit eder, sınırlar ve böylece sistemin optimizasyonu üzerine odaklanır (Mallach, 1994).

Dinamik modeller, sürekli modeller ve kesikli olay modeller olmak üzere iki başlık altında incelenir. Sürekli modeller, hesap ve diferansiyel eşitlikler aracılığıyla işletilir. Karmaşık sistemlerin sayısal olarak çözümlenmesi daha kolaydır. Sürekli modeller, bir değerden bir sonraki değere sorun çıkarmadan ilerler. Kesikli olay modeller, sistemi girilen her bir değere göre anlık biçimde değiştirerek modeller. Bu modeller daha çok iş planlama uygulamalarında kullanılmaktadır.

İki tür kesikli olay modeli vardır. Bir kesikli-olay modeli ya deterministik ya da stokastiktir. Eğer, belirli bir girdi kümesi için modelin çıktıları sabit ise model deterministiktir. Bu modellerin çözümü için genellikle hesap tabloları kullanılır. Çıktıların belirsizlik göstermesi durumunda ise model stokastiktir. Belirli bir girdi kümesi için bir stokastik modelin çıktısı, mümkün sonuçlar üzerinde rastlantısal olarak değişir. Bu model birkaç kez uygulandığında bir ortalama etrafında kümelenmiş birçok farklı sonuç elde edilecektir. Söz konusu ortalamanın varyansı modelleme süreci boyunca atılan adımlara bağlı olacaktır.

Bir modeli deterministik veya stokastik model olarak nitelendirme kararı, modelin içine ve dışına ne koyulduğuna bağlıdır (Mallach, 1994). Sistemin sınırı üzerinde anlaşmak suretiyle model içindeki belirsizlik kontrol altında tutulabilir. Hesaplama modelleri ve betimsel modeller sırasıyla deterministik ve stokastik modellere karşılık gelirler. **Kesinlik** modelleri, optimum sonuçlar verir. Finansal modellerin birçoğu kesinlik varsayımı altında kurulur (Turban, 1995). Eğer çözüm için yeterli bilgi yoksa mevcut problem **belirsiz** problem olarak ele alınır. Eğer problemin çözümü için daha çok bilgi edinimi sağlanmışsa problem hesaplanmış veya varsayılan **risk**

altında ele alınabilir. Risk altında karar almada karar alıcının karar değişkenlerinin gerektirdiği bazı olasılıkları bildiği varsayılır (Lawrence ve Pasternack, 2002). Risk çözümlemesini gerçekleştirmek üzere olasılıksal benzetim, zamana bağlı ve bağlı olmayan benzetim ve görsel benzetim teknikleri kullanılabilir.

Olasılıksal benzetim, olasılıkları bir veya daha fazla bağımsız değişkene ayırır. İki alt kategorisi vardır: Farklı değerlerin aynı olasılık değeri ile ayrıldığı **kesikli** dağılım ve değerlerin aynı ortalama ve varyansla istatistiksel bir dağılım gösterdiği **sürekli** dağılım. Olasılıksal benzetim, Monte Carlo olarak adlandırılan bir teknik yardımı ile gerçekleştirilir.

4.7.2.4. KD Sistemi Modellerinin Yedi Grubu

Modeller, veri elemanlarını kullanmak veya elde etmek için prosedür ve formülleri kullanır. Türetim için gerekli olan veri, harici bir veri tabanı veya diğer harici veri kaynakları aracılığıyla elde edilebilir. Kullanılan bu modellerin önemli bir özelliği, veri değişimine bağlı olarak modelde bir değişimi gerektirmemesidir (Malach, 1994).

Daha önce açıklandığı gibi KD Sistemleri'nin önemli özelliklerinden birisi de farklı türdeki modelleri birleştirerek kullanabilmesidir. Mallach (1994), KD Sistemi modellerini aşağıdaki şekilde incelemiştir.

- Sistem modelleriyle süreç modelleri
- Statik modeller ile dinamik modeller
- Sürekli modeller ile kesikli olay modeller
- Deterministik modeller ile stokastik modeller

Turban (1995) ise KD Sistemi modellerini aşağıda sıralanan yedi gruba ayırmıştır.

- Tam sayım, az alternatif
- Algoritma yoluyla optimizasyon
- Analitik formül yoluyla optimizasyon
- Benzetim

- Deneme-yanılma
- Diğer betimsel modeller
- Emirsel modeller

Turban (1995) tarafından belirtilen modeller aşağıda kısaca açıklanmıştır.

Tam Sayım: Bu modeller, sonlu sayıdaki ve az miktardaki alternatiflerin modellemesinde kullanılır. Tam sayım modelleri amaç sayısına göre; tek amaçlı veya çok amaçlı olabilir. Tek amaçlı karar çözümlemesinde karar tabloları veya karar ağaçları kullanılmaktadır. Çok amaçlı karar çözümlemesinde ise analitik hiyerarşi yöntemi, hedef programlama, çok amaçlı programlama, vb. çok çeşitli teknikler söz konusudur.

Matematiksel Algoritma Yoluyla Optimizasyon: Matematiksel programlama, problem çözümede oldukça yansız bir yaklaşım sunan bir tekniktir. Optimizasyon problemlerinin çözümünde yaygın biçimde kullanılan doğrusal programlama, matematiksel algoritma yoluyla çözüm bulan tekniklerden en bilinenidir. Doğrusal olmayan programlama, hedef programlama ve dağıtım problemleri yaygın biçimde kullanılan diğer programlama çeşitleridir. Bu modellerde amaç, çok miktardaki veya sonsuz sayıdaki alternatiflerden en iyi olanını seçmektir. Matematiksel algoritma kullanımını gerektiren durumlardan bazıları şunlardır:

- Dağıtım için sınırlı sayıda ekonomik kaynaklar mevcutsa (örneğin; işçi, sermaye, makina, vb.)
- Ürün veya hizmet üretiminde kullanılan kaynaklar
- Her biri bir çözüm programı olarak adlandırılan kaynaklar için iki veya daha fazla yolun bulunması
- Belirlenen bir hedef doğrultusunda her bir ürün veya hizmetin bir sonuç vermesi
- Dağıtımın birçok kısıt ile sınırlanması veya kısıtlanması
- Çeşitli kaynaklardan elde edilen kazançların karşılaştırılabilir genel bir birimle ölçülmesi

- Özel bir dağıtım kazancının diğer dağıtımlardan bağımsız olması
- Toplam kazancın, farklı hizmet veya ürün kazançlarının toplamı olması
- Bütün verinin kesinlikle biliniyor olması
- Kaynakların en ekonomik biçimde kullanılması

Analitik Formül Yoluyla Optimizasyon: Analitik formül yoluyla optimizasyon modellerinin amacı, tek bir hat formülü kullanarak en iyi çözüme ulaşmaktır. Genellikle envanter modellerini içerir. Bu kategoride matematiksel bir algoritma yardımıyla optimizasyon tekniğinin de kullanımı mümkündür. Envanter modellerinin içinde istatistik, finansal çözümlene, muhasebe, yönetim bilimi gibi modelleri bulunduran diğer modeller de vardır.

İstatistiksel ve finansal fonksiyonlar birçok KD Sistemi jeneratörü içinde oluşturulabilir. KD Sistemleri'nin birçoğu optimizasyon ve benzetim özelliklerini içermektedir. KD Sistemi jeneratörü, tek bir emir ile bu modelleri çağırabilir. Ayrıca bir KD Sistemi jeneratörü güçlü nicel yöntemlere bir arayüz olabilir. Önceden programlanmış nicel modellere erişim, kalıplar aracılığıyla gerçekleştirilir. Bu modellerden bazıları diğer modeller için yapı blokları olabilir. Örneğin, regresyon modeli, finansal planlama modelini destekleyen kestirim modelinin bir parçası olabilir.

Benzetim: Benzetim, genellikle denemeler yapmak için kullanılan bir tekniktir. Benzetim, gerçeği açıklayan modellerin aksine gerçeği taklit eder (Turban, 1995). Bu ise benzetim modellerinin diğer modellere göre gerçeği daha az basitleştirdiği anlamına gelmektedir. Normatif modellere oranla daha açıklayıcı olan benzetim modelleri kesin durumlar için kestirim yapar. Süreç, birçok deneme yapılarak gerçekleşir. Denemeler arasından en iyi olanı seçilerek çözüme ulaşılır.

Benzetim modelleri, genellikle çözümlenecek problemin sayısal optimizasyon teknikleri için çok karmaşık olması durumunda uygulanır. Doğru sonuçlar verebilmesi için mevcut problem ile ilgili ayrıntılı yararlı bilginin olmasını gerektiren benzetim modelleri, karar alıcının istekleri doğrultusunda oluşturulur ve oluşturulan bu modeller başka problemlerin çözümleri için kullanılamaz.

Karar alıcı, elde edilen benzetim modelleriyle farklı girdi deęişkenlerini deneyerek hangilerinin önemli olduklarını belirleyebilir. Ayrıca, farklı alternatifleri deneyerek hangi veri kümesinin en iyi olduğunu sınavabilir. Benzetim modelleri, karar alıcının “what-if” sorularını sormasına izin verir. Problem çözümü sürecinde deneme-yanılma yaklaşımını kullanan karar alıcı, benzetim modelleri aracılığıyla bu süreci en az riskle daha hızlı ve daha ucuz biçimde gerçekleştirebilir.

Benzetim modellerinin dezavantajları; optimal çözümü garanti etmemesi, oluşturulmasının genellikle yavaş ve pahalı bir süreç olması, çözümlerinin ve çıkarımlarının diğer problemlere devredilemez olması olarak sayılabilir.

Deneme-Yanılma Yöntemi İle Programlama: Benzetim süreci kimi durumlarda uzun, karmaşık ve doğru sonuçlar vermeyen bir yöntem olabilir. Bu gibi durumlarda deneme-yanılma yöntemine başvurularak daha uygun bir çözüme hızlı ve daha az maliyetle ulaşılabilir.

Deneme-yanılma, çeşitli kurallar yardımıyla karmaşık bir problemin yeterince iyi bir çözümünü bulma yöntemidir. Bu yöntemde kullanılan teknikler ise deneme-yanılma yöntemi ile programlama ve US’dir. Deneme-yanılma, genellikle kötü yapılaşmış problemlerin çözümünde kullanılırken iyi yapılaşmış karmaşık problemlerin çözümü için de uygun çözümler vermektedir. Alternatif oluşturmayı otomatik olarak gerçekleştirmek mümkün iken birçok KD Sistemi’nde bu işlem genellikle elle yapılmaktadır. Alternatifler deneme-yanılma ile oluşturulabilir. Bu durumda her bir alternatif için gelecek sonuçlarının kestirilmesi gerekmektedir (Turban, 1993). Deneme-yanılma yöntemlerini kullanmanın temel zorluğu bunların algoritmalar kadar genel olmamalarıdır. Bu nedenle deneme-yanılma yöntemleri sadece tasarlandıkları özel durumlar için kullanılabilirler. Süreç sonunda elde edilen sonuç yeterli bir çözüm olmayabilir.

Deneme-yanılmada, model kullanımını gerektiren durumlar şunlardır:

- Yanlış veya sınırlı sayıda girdi verisi varsa,
- Gerçek, optimizasyon modelini bozacak kadar karmaşıksa,
- Tam güvenilir bir model yoksa,

- Optimizasyonun hesaplanma zamanı çok uzunsa,
- Deneme-yanılma ile optimizasyon sürecinin verimliliği artırıyorsa,
- Problemler sık sık çözümleniyorsa ve çok fazla bilgisayar zamanını alıyorsa,
- Optimizasyon için ekonomik olmayan karmaşık problemler varsa,
- Sayısal yöntemden ziyade sembolik yöntem gerekliyse.

Diğer Betimsel Modeller: Nicel olmayan modelleri gerektiren diğer betimsel modeller, kurallar ve formülasyonların koşullarına göre ifade edilir. Bu modeller ayrıık olarak düzenlenebilir veya finansal modeller ve nicel modellerle birleştirilebilir.

Emirsel Modeller: Emirsel modellerin temel amacı, belirlenen bir senaryo için geleceği tahmin etmedir. Bu modellere örnek markov çözümlemesi ve tahmin modelleri verilebilir. Tahmin, geleceğin herhangi bir noktasında model değışkenlerinin değerini kestirmektir. Turban (1995), tahmleri deterministik modelleri kullanan “kısa zamanlı tahminler” ve hem deterministik hem de olasılıksal modelleri kullanan “uzun zamanlı tahminler” olarak ikiye ayırmıştır. Tahmin, kontrol edilemeyen faktörlerin çok olduğu durumlarda zorlaşır. Tahmin yöntemleri dört grupta toplanabilir.

Yargı Modelli Yöntemler: Subjektif tahminlere ve uzman görüşüne dayalı modelleri içeren yöntemlerdir.

Sayım Yöntemleri: Genellikle zaman serileri ve nedensel yöntemler arasında paylaşılan zor tarihsel veriye dayalı piyasa deneyimlerini veya arařtırmalarını içerir.

Zaman Serileri Çözümleme Yöntemleri: Tarihsel veriyle gelecek olayları tahmin etmek için kullanılan yöntemlerdir.

İlişkişel veya Nedensel Yöntemler: Zaman serilerinden daha fazla değışken içeren ve alternatifleri belirlemede karmaşık istatistiksel teknikler kullanan yöntemlerdir.

4.7.3. DİYALOG YÖNETİMİ ALT SİSTEMİ VEYA KULLANICI ARAYÜZLERİ

Diyalog alt sistemi, karar alıcının KD Sistemi ile iletişim kurmasını sağlayan alt sistemdir. Kullanıcı, bu alt sistemi kullanmak suretiyle KD Sistemi'ne ihtiyaç duyduğu bilgi ile ilgili ayrıntıları verir. KD Sistemi bu ayrıntıları kullanarak, veri kaynaklarından hangi bilginin çıkarılması gerektiğini belirler.

Diyalog alt sistemi veya kullanıcı arayüzü, KD Sistemi'nin etkili bir biçimde kullanımını sağlayan bölümdür. Bilginin nasıl görüneceğini ve kullanılacağını açıklayan çok çeşitli arayüz modelleri mevcuttur. Diyalog stilleri, diyalog biçimleri ve iletişim biçimlerinin tümünün amacı KD Sistemi'nin kullanımını kolaylaştırmaktır.

Menü etkileşimi, emir dili, soru-cevaplı grafik etkileşimi, doğal dil ve nesne kullanımı gibi stiller KD Sistemi'nin diyalog alt sistemini geliştirmek için kullanılan tekniklerdir. Problem çözümü için grafik kullanımı veriyi, ilişkileri ve özetleri görselleştirerek sunduğu için önemlidir. Grafiksel Kullanıcı Arayüzleri, karmaşık emir söz dizimlerinin yerine ikonlar ve butonlar gibi görsel nesnelere yardımcı işlemleri gerçekleştiren sistemlerdir. Bu sistemlerde çok çeşitli grafikler mevcuttur. Kullanıcı arayüzleri, interaktif multimedia'ların kullanımıyla zenginleştirilebilir. Hypermedia olarak adlandırılan multimedia'nın yeni bir türü, yararlı bilgi ve veriyi yönlendirebilen ve sonuçları yakalayabilen metin, grafik, audio, video gibi araçlardan birkaçını içerir. Eğer kullanıcı kullandığı dilde bilgisayar ile iletişim kurabilirse bu durumda problemlerin birçoğunu minimize edebilir hatta eleyebilir de. Bunun için bilgisayarın kullanıcının kendi diliyle girdiği verileri anlayacak kadar zeki olması gerekmektedir. Doğal Dil Süreci (DDS), Yapay Zeka (YZ)'nin bir parçasıdır. DDS'nde iki teknik kullanılmaktadır.

Anahtar kelime çözümlemesi: Benzerleri eşleştirmedir. Seçilen kelime veya ifadeleri arar.

Karmaşık dil süreci: Sözdizimsel ve anlamsal çözümlemedir. İletişim için dil gerekmektedir ve bilgisayarın konuşulan dili anlaması oldukça zor bir işlem olduğundan hala üzerinde çalışılan bir tekniktir.

KD Sistemleri'nin kullanıcıları problemi algılamada farklı kavrayış tercihlerine ve yeteneklerine, bir karara ulaşmada farklı yollara, dolayısıyla farklı destek türlerine gereksinim duyduklarından KD Sistemleri'nin kullanım şekilleri homojen değildir (Turban ve diğ., 2001).

4.7.4. BİLGİ YÖNETİMİ ALTSİSTEMİ

KD Sistemi içinde bulunup bulunmaması isteğe bağlı olan bir alt sistemdir. Yararlı bilgi yönetim alt sistemi, diğer alt sistemlerden herhangi birini destekleyebilir veya KD Sistemi içinde bağımsız bir bileşen gibi de bulunabilir. Amacı, problemin çözümü için yararlı bilgi sağlamaktır.

Karar almada bilgi yönetim desteği için aşağıdaki ifadeler sıralanabilir (Holsapple, 2001).

- Bilgiye dayalı karar almayı içeren bir aktivitedir.
- Bilgisayar tabanlı KD Sistemleri; betimsel bilgi, prosedürel bilgi ve nedensel bilgiyi içeren bir sistem olup karar alıcıların ilgilendikleri bilgiyi gösteren ve yöneten çeşitli yararlı bilgi yönetim tekniklerini kullanır.
- Bilgi yönetimi; insanlar, makinalar, organizasyonlar ve topluluklar tarafından yararlı bilginin temsil edilmesi ve yönetilmesiyle ilgilenir.
- Bilgi yönetiminin amacı, doğru maliyetler için doğru zamanlarda doğru varlıklarla doğru biçimlerde var olan doğru yararlı bilgiyi sağlamaktır.
- Bilgi yönetimi, küresel bilgi toplumu içindeki hızlı değişimde karar alıcıların geri kalmamasını sağlar.
- Yararlı bilgi bileşenin temelini US oluşturmaktadır.

4.8. KD SİSTEMLERİ'NİN SINIFLANDIRILMASI

KD Sistemleri kuruluş biçimlerine göre; tek kullanıcı hesap tablolarından, çok kullanıcı veri tabanlarına, yeniden kullanılabilir ajanları kullanan internet tabanlı KD Sistemleri'ne kadar geniş bir yelpazede çeşitlilik gösterir (Bui ve Lee, 1999). KD Sistemleri'nin amaçlarına, destekledikleri karar türlerine ve kullanıcılarına göre

farklı sınıflandırmaları da söz konusudur (Mallach, 1994). Finlay (1994), KD Sistemleri'ni farklı türlere ayırmanın gerekliliğini iki nedene bağlamıştır:

- KD Sistemleri'nin türlerine göre, bu sistemleri geliştirmede kullanılan yöntemlerin farklılaşması,
- KD Sistemleri'nin sınıflandırılmasıyla, araştırma için bir iskeletin sağlanması.

Kurulacak KD Sistemi türüne göre, kullanıcı ve kurucu etkileşimi de farklılık gösterecektir. Mantıksal bir modele dayalı olarak geliştirilen bir KD Sistemi'nin kurucusu, veri modeline dayalı biçimde geliştirilen KD Sistemi kurucularından farklı nitelikte olmak zorundadır. Sınıflandırma veya iskelet olmadan, farklı türlerdeki KD Sistemleri'nin sonuçları ve bulguları etkili biçimde açıklanamaz ve iletişim olmaksızın faktörlerin önem derecelerinin anlaşılması söz konusu olamaz (Mallach, 1994). KD Sistemleri farklı şekillerde sınıflandırılmaktadır. Uygulamada KD Sistemleri'ni yapılan bu sınıflamalardan birine koymanın mümkün olmadığı durumlarla da karşılaşmaktadır. KD Sistemleri'nin çeşitli sınıflarını Turban (2005) aşağıdaki biçimde belirtmiştir.

- Alter'in çıktı sınıflandırması
- Holsapple ve Whinston'un sınıflandırması
 - Metin yönlendirmeli KD Sistemleri
 - Veri tabanı yönlendirmeli KD Sistemleri
 - Hesap tablosu yönlendirmeli KD Sistemleri
 - Çözücü yönlendirmeli KD Sistemleri
 - Kural yönlendirmeli KD Sistemleri
 - Birleşik KD Sistemleri
- Bilgiye ve zekaya dayalı olarak KD Sistemleri'nin sınıflandırılması (Finlay, 1994)
- KD Sistemleri'nin diğer sınıflandırmaları, kurumsal ve ad-hoc KD Sistemleri

- o Kişisel, grup ve organizasyonel destek
- o Grup Destek Sistemi (GDS)'ne karşı bireysel KD Sistemi
- o Hazır yapım sistemlere karşı kullanıcı yapımlı sistemler

Bu sınıflandırmalardan en çok kullanılanı ilk defa Alter (1977) tarafından ortaya atılan ve KD Sistemleri'ni yaptıkları işlere göre kategorilere ayıran sınıflandırmadır.

4.8.1. ALTER'İN ÇIKTI SINIFLANDIRMASI

Yaptıkları işlerin türüne göre KD Sistemleri'nin sınıflandırılmasından Alter (1980), KD Sistemleri'nin kategorileri olarak söz etmiş ve bu sınıflandırmayı yedi kategori altında toplamıştır. Mallach (1994)'a göre bu kategoriler KD Sistemleri'nin farklı türlerini oluşturmaktadır. Finlay (1994)'a göre ise kategoriler, bilgi veya zekayla ilgili olup olmamalarına bağlı olarak YBS veya Yönetim Zeka Sistemleri (YZS) olarak sınıflandırılabilir. YZS'nin kullanımı büyük ölçüde içeriğe bağlı iken YBS ise genellikle içerikten bağımsızdır. YZS etkinlikle ilgili iken, YBS verimlilikle ilgilidir (Finlay, 1994).

KD Sistemleri'nin yedi kategorisi aşağıdaki şekilde sıralanabilir.

1. Dosya yürütme sistemleri
2. Veri çözümleme sistemleri
3. Bilgi çözümleme sistemleri
4. Hesap modelleri
5. Betimsel modeller
6. Optimizasyon sistemleri
7. Öneri sistemleri

Bu sınıflandırmanın ilk iki türü veri yönlendirmeli olup verinin yeniden edinimini veya çözümlenmesini gerçekleştirir. Üçüncü tür hem veri hem de modelleri ilgilendirmektedir. Geri kalan dört tür ise model yönlendirmeli olup benzetim

yeteneklerini, optimizasyon veya bir yanıt öneren hesaplamaları yürütmektedir. Turban (2005), Alter'in sınıflandırmasını Çizelge 4.2'deki gibi özetlemiştir.

Çizelge 4.2. KD Sistemleri'nin farklı sınıflarının karakteristikleri (Turban, 2005).

Yönlendirme	Kategori	İşlem Türü	Görev Türü	Kullanıcı	Kullanım Biçimi	Zaman
Veri	Dosya yürütme sistemleri	Veri parçalarına erişim	İşlemsel	Yönetici olmayan personel	Basit sorgular	Düzensiz
	Veri çözümlene sistemleri	Veri dosyalarının ad-hoc çözümlenmesi	İşlemsel çözümlene	Çözümleyici veya yönetici personel	Verinin düzenlenmesi veya gösterimi	Düzensiz veya periyodik
Veri veya Modeller	Bilgi çözümlene sistemleri	Çoklu veri tabanlarını ve küçük modelleri gerektiren ad-hoc çözümlenmesi	Çözümleme, planlama	Çözümleyici	Özel raporları programlama, küçük modeller geliştirme	Düzensiz, isteğe bağlı
Modeller	Hesap modelleri	Gelecek ile ilgili sonuçları, hesap tanımlarına bağlı olarak tahmin eden standart hesaplar	Planlama, bütçeleme	Çözümleyici veya yönetici	Etkinlik kestirimleri (Girdi) Kestirilmiş parasal sonuçlar (Çıktı)	Periyodik
	Betimsel modeller	Belirli hareketlerin sonuçlarını kestirme	Planlama, bütçeleme	Çözümleyici	Olası karar (Girdi) Kestirilmiş sonuçlar (Çıktı)	Periyodik veya düzensiz (ad-hoc çözümlenmesi)
	Optimizasyon sistemleri	Kombinasyonel bir probleme optimal çözüm bulma	Planlama, kaynak dağıtımı	Çözümleyici	Kısıtlar ve amaçlar (Girdi) Cevap (Çıktı)	Periyodik veya düzensiz (ad-hoc çözümlenmesi)
	Öneri sistemleri	Önerilen bir kararı oluşturacak hesaplamaları gerçekleştirme	İşlemsel	Yönetici olmayan personel	Karar durumunun yapısal bir açıklaması. (Girdi) Önerilen bir karar (Çıktı)	Günlük veya periyodik

Dosya Yürütme Sistemleri: Veri parçalarına çabuk erişimi sağlar. Bu sistemde karar alıcının harekete geçtiği nokta, bilginin istenen belirli bir parçasına yeniden erişimin istendiği andır. Dosya yürütme sistemleri en etkili ve genellikle en önemli KD Sistemleri'dir (Mallach, 2000).

Veri Çözümlene Sistemleri: Verinin kullanımına izin veren sistemlerdir. Dosya yürütme kategorisindeki verilerin hemen hemen hepsi aynı zamanda dosya çözümlene yeteneklerine sahiptir. Veri çözümlene sistemleri, seçilen veriye koşullu olarak yeniden erişim ve bu verinin aritmetik özetleri şeklindeki işlemleri yerine getirebilir (Mallach, 2000).

Bilgi Çözümlene Sistemleri: Bir dizi veri tabanları ve küçük modellere erişimi sağlayan sistemlerdir. Çözümleme, BS ve aynı düzeydeki harici kaynaklar tarafından gerçekleştirilir (Mallach, 2000).

Hesap Modelleri: Tasarlanmış hareketlerin sonucunu hesap tanımlamalarına dayalı olarak hesaplayan modellerdir. Hesap modelleri, her bir dönemdeki hesaplamaların sadece o dönemdeki diğer verilere bağlı olduğu ve belirsizliğin söz konusu olmadığı sistem modelleridir.

Betimsel Modeller: Betimsel modeller, genellikle bireysel veya kollektif insan davranışlarında görünen belirsizliği veya fazla çalışan sistemin dinamik davranışını temsil eder. Bu modeller daha çok bir kararın etkisini kestirmek için kullanılır (Mallach, 2000).

Optimizasyon Sistemleri: Önceden belirlenmiş kısıtlarla tutarlı olan optimal çözümün oluşturulması için ana hatları sağlayan sistemlerdir. Optimizasyon sistemlerinde, belirlenen alternatifler arasından en uygun olanının seçilmesi hedeflenir (Mallach, 2000).

Öneri Sistemleri: Oldukça yapısal bir görev için önerilen mekanik işleri gerçekleştirerek belirli bir karara ulaştıran sistemlerdir. İki tür öneri sistemi vardır (Mallach, 2000):

Betimsel sistemler: Karar alıcıya bir karar önerebilen sistemlerdir. Bu tip sistemler kararların oldukça yapısal olduğu durumlarda kullanışlıdır.

Emirsel sistemler: Uzman kişilerin muhakeme sürecini taklit ederek bir karara ulaşmayı sağlayan sistemlerdir. Makul sayıdaki kuralları kodlar.

KD Sistemleri'ni yaptıkları işlerin tiplerine göre değerlendiren Çizelge 4.2'deki sınıflandırmada ilk üç sınıflama (dosya yürütme, veri çözümlene, bilgi çözümlene), neredeyse tamamen veri tabanlarına odaklıdır. Bu sistemlerden **veri yönlendirmeli** sistemler veya veriyi yeniden edinen sistemler olarak söz edilir. Son dört sınıflama ise daha çok iş sistemi ilgili olup veri tabanlarına fazla odaklı değildir. Bu KD Sistemleri'ndeki veri tabanları model ile kullanılabilmesi için genellikle küçük, kişisel içerikli ve özgün biçimde yapılanmışlardır. Bu tür KD Sistemleri **model yönlendirmeli** veya dış değer bulucu sistemler olarak adlandırılır (Finlay, 1994). Optimizasyon ve öneri sistemleri, genellikle tamamen **süreç** modellerine dayalıdır. Bu nedenle bu iki sistemden süreç yönlendirmeli sistemler olarak söz

etmek mümkündür. Süreç yönlendirmeli sistemler, karar alma sürecinde insanın davranışlarını taklit etmektedir.

4.8.2. HOLSAPPLE VE WHINSTON'UN SINIFLANDIRMASI

Holsapple ve Whiston (1996), KD Sistemleri'ni altı kategoride sınıflandırmıştır.

- Metin yönlendirmeli KD Sistemleri
- Veri tabanı yönlendirmeli KD Sistemleri
- Hesap tablosu yönlendirmeli KD Sistemleri
- Çözücü yönlendirmeli KD Sistemleri
- Kural yönlendirmeli KD Sistemleri
- Bileşik KD Sistemleri

Metin Yönlendirmeli KD Sistemleri: Veri ve yararlı bilgiden oluşan bilgi, veri tabanlarında genellikle metinsel olarak ve karar alıcıların kolaylıkla erişebilecekleri biçimde saklanır. Bu nedenle, bilgiyi oluşturacak metin tarzındaki dökümanların ve bilgi parçalarının etkin ve etkili biçimde açıklanması ve işlenmesi çok önemlidir. Metin yönlendirmeli KD Sistemleri, metinsel olarak saklanmış ve alınan kararları etkilemiş olabilecek bilginin elektronik biçimde takip edilmesini sağlayarak karar alıcılara destek sağlamaktadır. Bu sistemler, daha çok nakliye şirketleri tarafından siparişleri düzenlemede, müşterilerine en iyi taşıma yollarını göstermede, müşterilerin ve şirketlerin gönderdikleri paketlerin durumunu izleyebilmelerinde yardımcı olmaktadır. Elektronik belge yönetim sistemleri, bilgi yönetim sistemleri, içerik yönetim sistemi, iş kuralları sistemleri metin yönlendirmeli KD Sistemleri'ne örnek olarak verilebilir (Turban, 2005).

Veri Tabanı Yönlendirmeli KD Sistemleri: Veri tabanı yönlendirmeli KD Sistemleri, içerdikleri veri taban(lar)ının yapılarına göre oluşan sistemlerdir. Bu sistemlerin ilk kuşaklarında İVT'nın düzeni kullanılmıştır. İVT hacimli, betimsel ve yapısal yapıdaki bilgiyi sakladığından veri tabanı yönlendirmeli KD Sistemleri de aynı tür verileri saklamaktadır. Güçlü rapor ve sorgu oluşturabilmesi önemli özelliklerindedir.

Hesap Tablosu Yönlendirmeli KD Sistemleri: Hesap tablosu bir modelleme sistemi olup kullanıcılarına KD Sistemi çözümlerini uygulayabilecekleri modelleri geliştirmelerinde yardımcı olur. Geliştirilen modeller sadece prosedürel yararlı bilgiyi oluşturmak, görüntülemek ve değiştirmekle kalmaz, aynı zamanda sisteme kullandığı modellerin makrolarını (kendiliğinden içerdikleri bilgi) uygulayabilmesi için yardımcı da olmaktadır (Turban, 2005). Son kullanıcılar tarafından geliştirilen KD Sistemleri genellikle hesap tablolarını kullanan sistemlerdir. Bu tür sistemleri geliştirmek için kullanılan en yaygın araç Excel'dir. Excel, içinde çok sayıda istatistiksel paketler, çözücü olarak adlandırılan bir doğrusal programlama paketi ve birçok finansal ve yönetsel modeller bulundurduğundan özellikle tercih edilmektedir. Excel ayrıca VTYS içermesi veya bu tür bir sisteme arayüz oluşturabilmesi bakımından da önem taşımaktadır. Hesap tablosu geliştirmek amacıyla kullanılan araçlardan bazıları da what-if çözümlenmesi ve hedef arama yeteneklerini de içerebilmektedir (Turban, 2005).

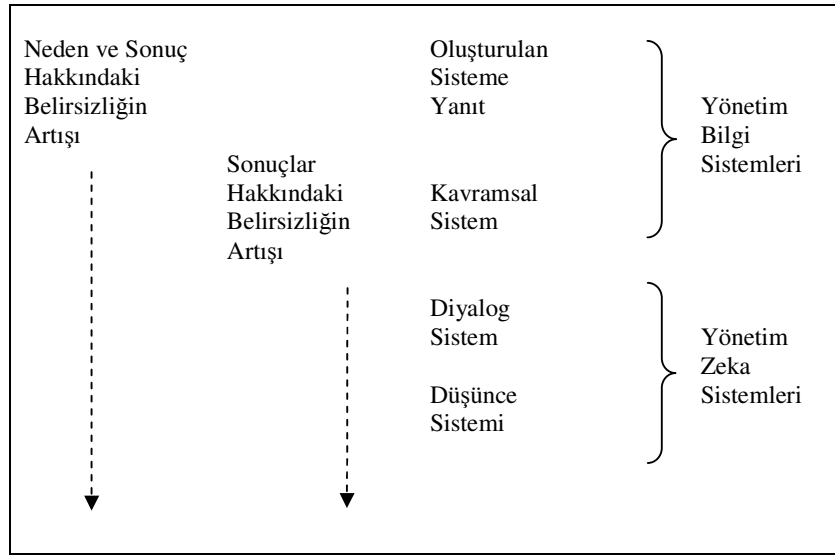
Çözücü Yönlendirmeli KD Sistemleri: Çözücü, özel bir problemi çözmek üzere bilgisayar programı şeklindeki belirli hesaplamaları gerçekleştiren bir çeşit algoritma veya prosedürdür (Turban, 2005). Çözücü, Excel gibi sistem geliştirme yazılımları içinde hazır biçimde bulunabileceği gibi C++ gibi bir programlama dili kullanılarak hesap tablosu üzerine doğrudan yazılmış ya da bir "add-in" aracı şeklinde hesap tablosuna sonradan eklenebilecek biçimde de olabilir. Bir diğer alternatif ise çözücülerin Lingo gibi özelleştirilmiş bir modelleme dili içine gömülü olabilmesidir. Daha karmaşık işlemleri gerçekleştiren çözücüler ise bağımsız biçimde bulunabilirler. Bu tür çözücüler KD Sistemleri içine yerleştirmek mümkündür.

Kural Yönlendirmeli KD Sistemleri: KD Sistemleri'nin daha önceden açıklanan yararlı bilgi bileşenleri hem işlemsel hem de sonuç çıkarsamalı kuralları (nitel veya nicel) içeren genellikle US formatındaki bileşenlerdir. Bu tarz bir bileşen nicel modellerin yerine kullanılabilir veya modeller ile birleştirilebilir (Turban, 2005). Kural yönlendirmeli KD Sistemleri betimsel, prosedürel, mantıksal, dilbilimsel, gösterimsel ve özümsemeli olmak üzere altı çeşittir (Turban, 2005). Bunlardan ilk üçü "ana" türler olup diğerleri bunlardan türetilmiştir

Bileşik KD Sistemleri: Yukarıda sözü edilen beş basit yapıdan iki veya daha fazlasının bileşimiyle, oluşturulan sistemlerdir (Turban, 2005).

4.8.3. KD SİSTEMLERİ'NİN BİLGİYE VE ZEKAYA DAYALI OLARAK SINIFLANDIRILMASI

KD Sistemleri'nin oluşumundaki zeka ifadesi büyük önem taşımaktadır. Finlay (1994), KD Sistemleri'nin türleri arasındaki en büyük farklılığın, bilgiyi ve zekayı sağlayan alt sistemlerinden kaynaklandığını vurgulamıştır. Finlay (1994), ayrıca KD Sistemleri'nin bilgi veya zeka ile ilgili olmalarına bağlı olarak farklı önem derecelerine de sahip olduklarına dikkat çekmiştir. Bilgiye dayalı KD Sistemleri, YBS'ni kullanırken zeka ile ilgili KD Sistemleri, YZS'ni kullanır. Şekil 4.5'de YBS ile YZS arasındaki ilişkiler gösterilmiştir.



Şekil 4.5. KD Sistemleri arasındaki ilişkiler (Finlay, 1994).

Yönetim Bilgi Sistemleri: YBS'nin iki ayrı başlık altında incelenmesi uygundur.

Veriyi Yeniden Edinme Sistemleri: Geçmiş hakkında bilgi edinimini sağlayan YBS'dir. Bu sistemler belli başlı üç sınıfta toplanır (Finlay, 1994).

Dosya Yürütme Sistemleri: Yeniden yapılaşmış veriye hızlı ad-hoc erişimi sağlayan sistemlerdir.

Veri Çözümleme Sistemleri: Yeniden tanımlanmış durumların ileriye doğru çözümlenmelerini, raporlarını ve grafiksel çıktılarını sağlayan sistemlerdir.

Uygulamalı Bilgi Sistemleri: Daha önceki uygulamalardan seçilmiş ve özetlenmiş bilgiyi ve eğilim tahminlerini uygun biçimde sağlayan, veriyi yeniden edinme sistemlerinin gelişmiş bir türüdür.

Dış Değer Bulucu Sistemler: Geçmiş verilere dayalı olarak extrapolasyon yaparak veya ilişkileri kullanarak gelecek hakkında bilgi sağlayan YBS'dir. Yönetim bilimi tekniklerinin bir çoğu bu sistemler aracılığıyla kullanılmaktadır. Finlay (1994), dış değer bulucu sistemleri üç alt sınıfa ayırmıştır.

Tanımsal Sistemler: Değişkenler arasındaki tanımsal ilişkileri yeniden tanımlayan sistemlerdir.

Nedensel Sistemler: Neden ve sonuç ilişkilerini tanımlayan sistemlerdir. Bu ilişkiler olanakların elverdiği ölçüde geçmişteki istatistiksel veriden türetilbilir ve bir sonuç elde etmek için varsayılan gelecek verisiyle kullanılabilir. Bu tür sistemlerde sistem kurucusunun bir matematiksel modelleme uzmanı olması gerekir.

Olasılıksal Sistemler: İçlerinde olasılıksal özellikler bulduklarından dış değer bulucu sistemler tarafından seçilen sistemlerdir. Olasılıksal sistemlerde veri modelleri olasılıksal ve/veya istatistiksel dağılımlara gerek duyduğundan sistem kurucusunun gerekli düzeyde matematiksel ve istatistiksel bilgiye sahip olması gerekmektedir. Benzetim ve istatistiksel kestirim, sistem tarafından gerçekleştirilen işlemlere örnek olarak verilebilir.

Yönetim Zeka Sistemleri (YZS): Zekaya dayalı KD Sistemleri'nde kullanılan YZS'nin iki farklı türü vardır.

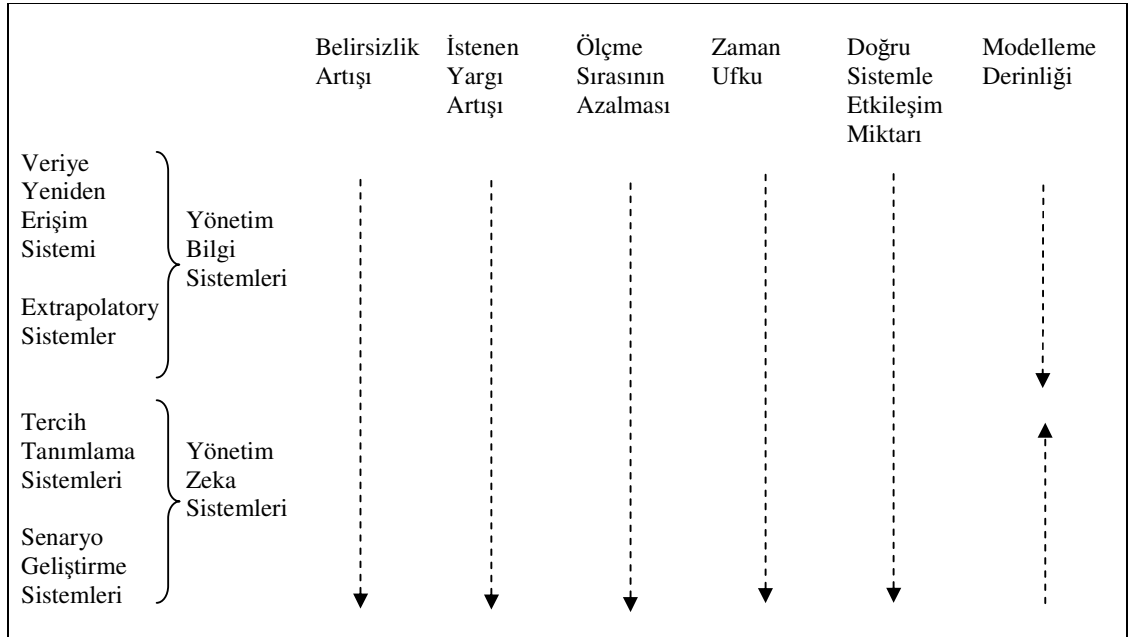
Tercih Tanımlama Sistemleri: Karar alıcıların biçimsel olarak düşündükleri neden ve sonuçlar olmadan, seçenekler arasından seçimin yapıldığı YZS'dir. Olası seçeneklerin ve ilişkilendirilmiş kriterlerin ayrıntılı bir listesini gerektirir. Bu tür sistemler, karar ağaçları ve çok nitelikli kriterler gibi ileri düzey karar alma tekniklerine dayalı olarak kategorize edilebilir. Karar ağaçları ile bir dizi karar, açık ve uygun biçimde grafiksel olarak açıklanabilir; mümkün seçenekler ve çıktıların ayrıntılı bir listesiyle olasılıksal özellikler birleştirilebilir. Çok nitelikli KD Sistemleri, açık biçimde belirtilmiş olan birçok kriteri gerektirir. Bu kriterler ve seçenekler arasındaki etkileşim, karar alıcının yargısını içeren bir matrisin

oluşturulmasıyla sağlanır. Tercih tanımlama sistemleriyle problemin durum yapısı ortaya çıkartılır. Kullanıcılardan edinilen subjektif girdi ve belirsizliğin boyutu da sisteme dahil edilebilir. Bu KD Sistemleri türünün mevcut araçları, grup karar almayı destekler.

Senaryo Geliştirme Sistemleri: Karar alıcının bütün görüşlerini gerçekleştirebildiği YZS'dir. Bu sistemlerde insanın kavrayış ve etkileşim süreçlerini çoğaltacak araçlar önemli yer tutmaktadır. Karar alıcıya aktarılan bilginin çokluğu problemin çözümü için oluşturulabilecek senaryoların daha iyi olmasını ve karar alıcıya zeka için gerekli olan hammaddenin kolaylıkla edinilmesini sağlar. Karar alıcı, senaryosunu yorumlamada kullanacağı zekanın oluşturulmasında yardıma gerek duyabilir. Bu amaçla kullanacağı bilgi modelindeki verinin en azından, bilgi ve zekanın oluşturulduğu süreç kadar iyi olması gerekir. Senaryo geliştirme sistemleri iki alt sınıfa ayrılır.

Kavrayış Planlama Sistemleri: Karar alıcıların veya karar alıcı grubunun genel düşüncelerini düzenlemelerine yardımcı olan sistemlerdir.

Fikir Üretme Sistemleri: Beyin fırtınası gibi oturumları destekleyen sistemler olup serbestçe fikir alışverişine olanak verir.



Şekil 4.6. KD Sistemi türleri (Finlay, 1994).

Şekil 4.6’da YBS’nin veriye yeniden erişim sistemi, dış değer bulma sistemleri ve YZS’nin tercih tanımlama sistemleri, senaryo geliştirme sistemleri karşılaştırılmıştır.

YBS ve YZS arasındaki farklılıklar, yukarıdaki açıklamalar doğrultusunda, aşağıdaki gibi özetlenebilir.

- YBS genellikle içerikten bağımsız iken YZS içeriğe oldukça bağımlıdır.
- YBS verimli çalışma ile ilgili iken YZS etkinlikle ilgilidir.

YBS ve YZS’nin önem verdikleri işlevlere göre karşılaştırılması Çizelge 4.3’de gerçekleştirilmiştir.

Çizelge 4.3. YBS ve YZS’nin karşılaştırılması (Finlay, 1994).

	Yönetim Bilgi Sistemleri (YBS)	Yönetim Zeka Sistemleri (YZS)
Sistemin Türü	Dahili kontrol/bütçeleme	Planlama
Odak noktaları	Hızlı ve verimli, yapısal bilgi akışı ve veri yapıları Hızlı ve verimli çalışma	Etkili kararlar, esneklik, uyarlanabilirlik ve hızlı yanıt verme Etkinlik
Amaçları	Önceden belirleme	Ad-hoc/bağlılık
Bulunduğu yerin türü	Değişmeyen politikalar içinde	Belirli bir senaryo içinde
Yaratıcısı	IT uzmanı/problem çözümleyicisi	Kullanıcılar/problem çözümleyicisi
Tasarım perspektifi	Organizasyonel	Bireysel/küçük gruplar
Kullanılma tasarımı metodolojisi	“Klasik” sistem yaklaşımı ve girdi ve çıktıların prototipi	“Breadboarding”
Yazılım/donanım yönlendirme	Yazılım ve donanım	Yazılım
Modeller		
i)	Sabit mantık	Evrimsel mantık
ii)	Çoğunlukla deterministik ilişkiler	Yargısal ilişkiler
iii)	Çoğunlukla aritmetiksel ve matematiksel	Çoğunlukla mantıksal
iv)	Çoğunlukla deterministik veri	Olasılıksal veri
v)	Oran ve aralık ölçekleri	Nominal ve ordinal ölçekler
Çıktı		
i)	Genel format	Kullanıcının oluşturduğu
ii)	Standartlaştırılmış/sorulu raporlar	İteratif/interaktif iyi yapılaşmamış raporlar
iii)		Anlayış, bilgi, diyalog
iv)	Bir yanıt Yönetim ilgisi	zeka
Zaman dilimi	Geçmiş, şimdi ve gelecek	Şimdi ve gelecek
İçerik	İçerikten bağımsız Yapısal	İçeriğe bağımlı İyi yapılaşmamış
Kesinlik	Kesinlik ve doğruluk	Doğruluk
Geçerlilik	Büyük ölçüde gerekli	İsteğe bağlı
Kullanım		

4.8.4. KD SİSTEMLERİ'NİN DİĞER SINIFLANDIRMALARI

KD Sistemleri'nin diğ er bir sınıflandırması şöyledir:

- Kurumsal ve ad-hoc KD Sistemleri
- Kişisel, grup ve organizasyonel destek
- Grup Destek Sistemi'ne (GDS) karşı Bireysel KD Sistemleri
- Hazır yapım sistemlere karşı kullanıcı yapımlı sistemler

Kurumsal ve Ad-Hoc KD Sistemleri: Kurumsal KD Sistemleri, tekrarlamalı yapıya sahip kararların alınmasında kullanılan sistemlerdir (Donovan ve Madnick, 1977). Bu tür sistemler benzer problemleri tekrarlı biçimde çözümledikleri için yıllarca hiç bir değışiklik yapılmadan kullanılabilir. Portföy yönetim sistemi kurumsal KD Sistemleri'ne örnek olarak verilebilir.

Ad-hoc KD Sistemleri, genellikle önceden tahmin edilemeyen ve tekrarlamalı olmayan özel problemlerle ilgilenmektedir (Turban, 2005). Ad-hoc KD Sistemleri, sadece bir kez kullanılmak üzere geliştirilen sistemlerdir ve tek bir birey tarafından kullanılır. Stratejik planlama problemleri ile kimi yönetim kontrol problemleri ad-hoc kararlar olarak nitelendirilir. Ad-hoc KD Sistemleri uygulamalarının birçoğı kurumsal KD Sistemleri içinde geliştirilmiştir (Turban, 2005).

Kişisel, Grup ve Organizasyonel Destek: KD Sistemleri tarafından verilen destek birbiriyle ilgili üç farklı kategoriye ayrılabilir (Hackathorn ve Keen, 1981). Bu kategoriler kişisel, grup ve organizasyonel destek şeklinde sayılabilir. Kişisel destek, farklı bir görev veya karar içindeki bir faaliyeti gerçekleştiren bireysel kullanıcı üzerine odaklanmıştır. Bu görev diğ er görevlerden oldukça farklıdır. Grup destek, ayrı fakat birbirleriyle oldukça ilgili görevleri yerine getiren bir grup insan üzerine odaklanmaktadır. Organizasyonel destek ise ardışık işlemleri, farklı fonksiyonel alanları, imkan dahilindeki farklı konumları ve güçlü kaynakları içeren organizasyonel görevler veya aktiviteler üzerine odaklanmıştır.

Grup Destek Sistemi'ne (GDS) Karşı Kişisel KD Sistemi: Birçok organizasyonda kararlar ortak biçimde alınır. Kimi zaman karmaşık bir süreç olarak

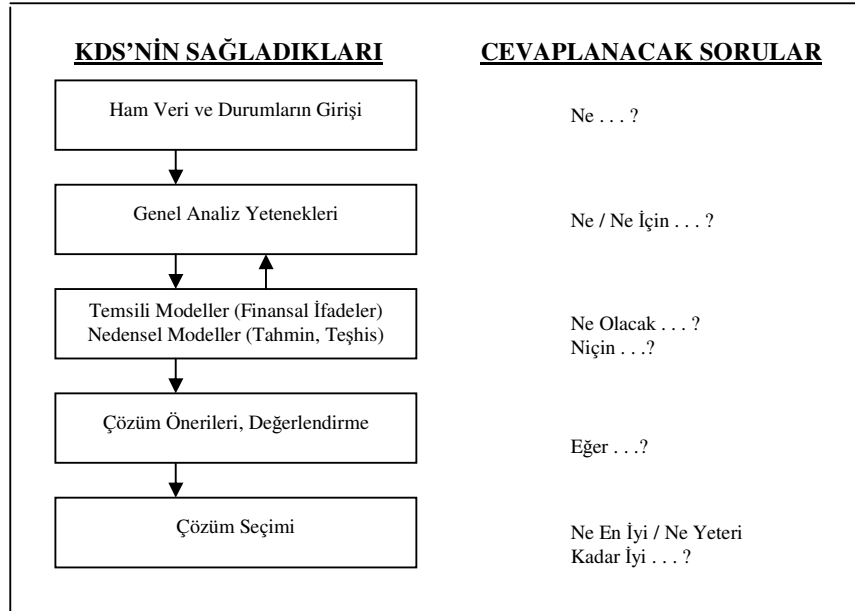
nitelendirilebilecek grup halinde çalışma “Grup Destek Sistemi (GDS)” olarak adlandırılan sistemler tarafından desteklenebilir (Turban, 2005).

GDS ile grup destek farklı kavramlardır. Grup destekte kararlar, görevleri birbirleriyle ilişkili olan bireyler tarafından alınmaktadır. Bu bireyler kararların diğer kararlar üzerindeki etkilerini kontrol edebilir, fakat kararları bir grup biçiminde almak zorunda değildir. GDS’nde ise her bir karar veya kararlar, bir grup tarafından ortaklaşa alınır.

Hazır Yapım Sistemlere Karşı Kullanıcı Yapımlı Sistemler: Bireysel kullanıcıların ve organizasyonların kullandıkları KD Sistemleri’nin birçoğu hazır yapım KD Sistemleri’dir. Farklı organizasyonlar benzer problemlere sahip olabilir. Bu nedenle organizasyonların doğrudan veya ufak değişikliklerle kullanabilecekleri genel KD Sistemleri’ni edinmeleri daha doğru olacaktır. Bu tür KD Sistemleri’ne “hazır yapım KD Sistemler” denir ve satışı birçok şirket tarafından gerçekleştirilebilmektedir.

4.9. KD SİSTEMLERİ’NİN SAĞLADIKLARI DESTEK

KD Sistemleri çok çeşitli destek sağlayabilir. KD Sistemleri tarafından sağlanan desteği gösteren Şekil 4.7’deki yapı Alter (1980)’e dayanmaktadır.



Şekil 4.7. KD Sistemleri tarafından sağlanan destek (Turban, 1995).

Şekil 4.7. ile açıklanan yapıda, destek düzeylerinin her biri bir önceki düzeyi içerebilir ve bir önceki düzeye eklenebilir (Turban, 1995).

4.10. KD SİSTEMLERİ’NİN ORGANİZASYONLA BÜTÜNLEŞTİRİLMESİ

Organizasyonların ve KD Sistemleri içindeki kullanıcı gereksinimlerinin, uygulanan genel stratejilerin ve organizasyonel politikaların farklılık göstermesi olağandır. Bu farklılıklar eğitim, donanım ve değerlendirmeden oluşan bir bütünleştirme süreci ile giderilebilir (Sprague ve Carlson, 1982). Bütünleştirme sürecinin amacı, başarısızlık riskini azaltmaktır. Alter (1980), KD Sistemleri’nin başarısızlık olasılığını artıran sekiz tane risk faktörü tanımlamıştır. Bu faktörler Çizelge 4.4’de sıralanmışlardır.

Çizelge 4.4. KD Sistemleri’nin risk faktörleri ve bütünleşme süreci (Sprague ve Carlson, 1982).

RİSK FAKTÖRÜ	RİSKİ AZALTACAK BÜTÜNLEŞME SÜRECİ
Mevcut veya gönüllü olmayan kullanıcı	Eğitim, değerlendirme
Çoklu kullanıcılar veya uygulayıcılar	Eğitim, donanım
Görünmeyen kullanıcılar, uygulayıcılar veya koruyucular	Eğitim, donanım, değerlendirme
Özel amaç veya örnek kullanımdaki yetersizlik	Eğitim, değerlendirme
Etkinin kestiriminde ve azaltılmasında yetersizlik	Eğitim, değerlendirme
Desteğin kaybedilmesi veya destek eksikliği	Eğitim, donanım, değerlendirme
KD Sistemleri ile deneyim eksikliği	Eğitim, donanım, değerlendirme
Teknik problemlerin ve maliyetin etkinliği	Donanım, değerlendirme

Kullanıcı eğitimi, kullanılan KD Sistemleri’nin nasıl işletilmesi gerektiğinin eğitimidir. Kullanıcının KD Sistemleri’ni rahat kullanabilmesi için, KD Sistemleri’ni kullanarak problemleri çözme, KD Sistemleri’ni yeni problemlere uyarlama ve KD Sistemleri’nin özel niteliklerini kullanma konularında eğitilmesi gerekir.

4. 11. ORGANİZASYONEL KARAR ALMAYA AYRINTILI BİR BAKIŞ

İnternet ve telekomünikasyon alanındaki gelişmeler sonucunda, organizasyonlar işlerini daha küresel, birleşik ve birbirleriyle bağlantılı bir şekilde gerçekleştirmektedir. Böyle bir ortamda görev yapan yöneticiler oldukça farklı düşünce biçimleriyle karşı karşıya kalacaktır. Bu ise, karar alma sürecinde kültürel, organizasyonel, kişisel, etik ve estetik faktörlerin daha ayrıntılı bir biçimde düşünülmesini gerektirecektir. Kullanılacak KD Sistemleri’nin ise daha yumuşak

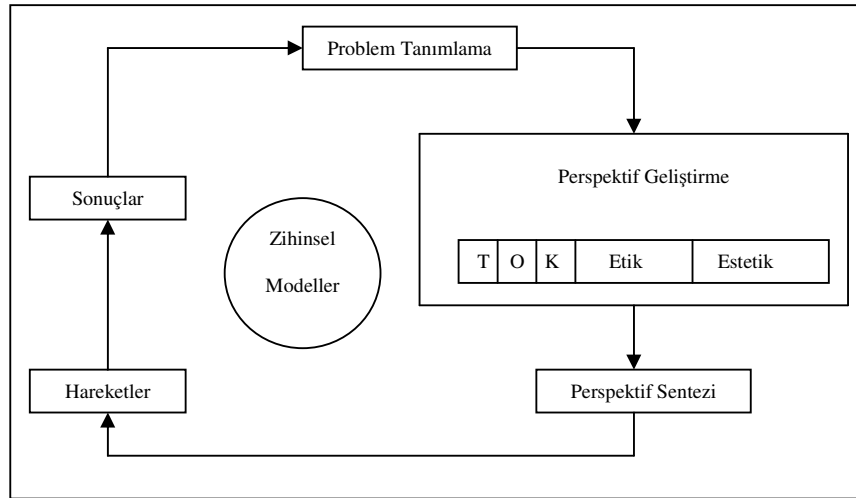
bilgiyi matematiksel sistemler ve bilgi tabanlı sistemlere göre daha kapsamlı bir içerik içinde yönetebilme yeteneğine sahip olması gerekmektedir.

Organizasyonel karar alma çeşitli şekillerde sınıflandırılabilir. Bu sınıflandırmalar aşağıdaki gibi özetlenebilir (Eom, 2001).

Organizasyonel karar alma,

- Tek bir karar alıcıyı tamamen bilgilendiren, rasyonellik varsayımına dayalı olarak en etkili alternatiflerin seçimi üzerine odaklanan rasyonel bir modeldir.
- Herhangi bir organizasyon içindeki çeşitli birimlerin alt bölümlere ayrılmasını önemseyen organizasyonel bir süreçtir.
- Makul, yeterince iyi bir çözüm bulmak amacıyla “sınırlı rasyonellik” ifade eden tatmin edici bir modeldir.

Organizasyonel karar alma Şekil 4.8’de ayrıntılı biçimde gösterilmiştir (Courtney, 2001). Problem; formülasyon aşamasının bakış açısını, çoklu ve değişik perspektifleri (Organizasyonel (O), Kişisel (K), Teknik (T)) içermeli, ayrıca etik ve estetik faktörleri de dikkate almalıdır.



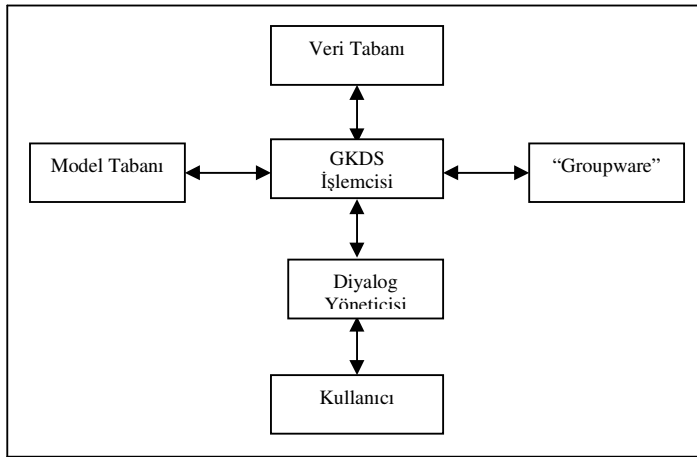
Şekil 4.8. KD Sistemleri için yeni bir karar paradigması (Courtney, 2001).

Problemin tanımlandığı noktadan, çözümüne kadar olan süreç problem perspektifine dayanır. Teknik perspektif, süreci geçmişte yöneten ve KD Sistemi içindeki veri tabanlarının ve modellerin gelişimini içerir. Kişiler, karar alırken farklı yöntemler

kullanabileceklerinden, yapısal olmayan karmaşık bir durumda, farklı kişilerin aynı harici bilgilerle farklı sonuçlara ulaşmaları sürpriz sayılmamalıdır. Bu nedenle organizasyonel ve kişisel perspektiflerin, özellikle de etik ve estetik ile ilgili olanlarının, ölçümü gerçekleştirilemeyebilir.

4.12. GRUP KARAR DESTEK SİSTEMİ

KD Sistemleri bireysel olarak insanların karar almalarına yardımcı olabileceği gibi, bir grup olarak çalışan insanların karar almalarına da yardımcı olabilmektedir. Grup Karar Destek Sistemleri (GKD Sistemleri), grup olarak alınan kararları desteklemek için gereken bütün donanım, yazılım, insan, veri tabanları ve işlemleri içeren sistemlerdir (Nickerson, 1998). Stair (1992)'e göre GKD Sistemleri'nin düzeni Şekil 4.9'daki gibidir.



Şekil 4.9. GKD Sistemleri'nin düzeni (Stair, 1992).

Stair (1992) GKD Sistemleri'nin tipik özelliklerini aşağıdaki gibi özetlemiştir.

- Özel tasarım
- Kolay kullanım
- Belirli ve genel destek
- Negatif grup davranışlarını yok etme
- Pozitif grup davranışlarını destekleme

Yine Stair (1992)'e göre GKD Sistemleri'nin yetenekleri şunlardır:

- Kelime işlemci ve metin kullanımı
- Veri tabanı ve dosya kullanımı
- İş programlama veya hesap tablosu yetenekleri
- Grafik paketleri
- Karar alma yardımları
- İletişim kolaylıkları
- Yardım kolaylıkları

4.13. KD SİSTEMLERİ'NİN KULLANIM ALANLARI

KD Sistemleri çok geniş bir uygulama alanına sahiptir. Eom (2001) bu uygulama alanlarını aşağıdaki başlıklar altında özetlemiştir.

- Hesaplama veya denetim
- Finans
- İnsan kaynakları yönetimi
- Uluslararası ticaret
- Bilgi sistemleri: Veri iletişimi; KD Sistemi jeneratörleri; sistem çözümlemesi, tasarım, gelişme ve değerlendirme; bilgi kaynakları yönetimi
- Satış
- Üretim ve operasyon yönetimi: Talep planlaması, yönetici zamanlaması, işlemleri zamanlama ve kontrol, yönetsel tasarım, kapasite planlama, envanter planlama, kaynak yönetimi
- Stratejik planlama
- Çok fonksiyonlu yönetim

4.14. GELECEK KUŞAK KD SİSTEMLERİ

Gelecek kuşak KD Sistemleri iki yönden incelenebilir. Tasarım ile ilgili değişim ve gelişmeleri kapsayan birinci yön, KD Sistemleri'nin içerdiği elemanlarla ilgilidir. İkinci yön ise KD Sistemleri'nin içerdiği ya da içereceği elemanların donanım, yazılım ve uygulama alanlarındaki değişim ya da gelişimleri ilgilendirmektedir.

4. 14.1. KD SİSTEMLERİ'NİN GELECEKTEKİ TASARIMLARI

Veri tabanı yetenekleri olan modelleme fonksiyonelliği ve arayüz tasarım bileşenleri, KD Sistemleri'nin klasik tasarım araçları olarak görülebilir (Shim ve diğ., 2002). KD Sistemleri'ni gelecek için tasarlamada bu bileşenlere ek olarak veri ambarları, on-line analitik süreç (OLAP), veri madenciliği ve web tabanlı KD Sistemleri gibi araçlar kullanılabilir. KD Sistemleri'nin gelişimi için işbirlikçi destek sistemleri, görsel gruplar, optimizasyon tabanlı KD Sistemleri ve aktif KD odaklı konular da önemlidir.

Veri Ambarı: Veri ambarı, karar alma sürecinin yönetimini desteklemek üzere kullanılan konu yönlendirmeli, bütünleşik, zaman değişimli, değişken olmayan veri koleksiyonudur (Inmon ve Hackathorn, 1994). Barquin (1996)'a göre veriyi ambarlama, organizasyonların veri ambarlarını kullanmaları sonucunda bilgi içeren varlıklarından anlam çıkarma sürecidir.

On-Line Analitik Süreç (OLAP): OLAP, kullanıcılara verinin çok boyutlu görünüşlerini sağlamak için geliştirilen ve grafiksel bir arayüz kullanarak verinin kolaylıkla çözümlenmesini sağlayan araçlar kümesidir (Hoffer ve diğ., 2002). Genellikle, çok boyutlu çözümlenme denilince ilk akla gelen OLAP olmaktadır. OLAP, veri ambarında saklanan eski verinin çözümlenmesinde yardımcı olmaktadır. OLAP araçları bilginin çeşitli mümkün görünüşlerine hızlı, tutarlı, interaktif biçimde erişimi sağlar.

Veri Madenciliği: Veri madenciliği; istatistik, yapay zeka ve ayrıntılı bilgisayar tekniklerini kullanarak bilgi keşfi veya veri tabanı araştırmasıdır (Weldon, 1996).

Veri madenciliği araçları, veri içindeki modelleri bulur ve bunlar için çeşitli kurallar koyar. Yeterli boyut ve kalitedeki veri tabanları, veri madenciliği teknolojisi ile eğilim ve davranışların kestirimini, veri tabanı içindeki verinin daha önceden

bilinmeyen biçimlerinin keşfini otomatikleştirmek suretiyle yeni iş imkanları sağlayabilir (Turban ve diğ., 2000). Veri madenciliğinde sinirsel hesaplama, zeki ajanlar ve işbirliği çözümleri gibi birçok teknik kullanılabilir (Turban ve diğ., 2000). Sinirsel hesaplama, çeşitli modeller için eski verinin sınırlanabildiği bir bilgi üretme yaklaşımıdır. Zeki ajanlar kullanıcıya rutin görevleri yönetmede, bilgiyi araştırmada ve ilk kez elde etmede, karar almayı desteklemede, alan uzmanlığını gerçekleştirebilmede yardım eden ve böylece zamandan tasarruf sağlayan bilgisayar programlarıdır. İşbirliği çözümleri, büyük veri kümelerini sıralamak için özel olarak geliştirilmiş algoritmaları kullanan ve maddeler arasındaki istatistiksel kuralları açıklayan bir yaklaşımdır.

Web Tabanlı KD Sistemleri: Web Tabanlı KD Sistemleri, netscape veya internet explorer gibi web araştırma motorlarını kullanarak karar alıcıya KD bilgisi veya herhangi bir KD aracı sağlayabilen bilgisayarlaştırılmış sistemlerdir.

Intranet Kullanımıyla KD: Internet, Yerel Alan Ağı (Local Area Network, LAN) veya Internet Servis Sağlayıcısı (Internet Service Provider, ISP) üzerinden, herhangi bir kimseye doğrudan bağlanmayı sağlayan genel ve küresel bir iletişim ağıdır. Kullanıcılar, internetten elde edilen genel ve reklam bilgisinin engin alanını yönlendirmek için etkili, hızlı ve verimli çalışan araçlara gereksinim duyar (Turban ve diğ., 2000). Intranet, internet teknolojisini kullanan ortak bir LAN olup organizasyonların “firewall”larını³ sağlamlaştırır.

Alıcı KD Sistemleri: Alıcı KD Sistemleri, bir firma ile müşterileri arasında bağlantı kuran ve onların karar alma sürecine destek sağlayan web tabanlı bir pazarlama modelidir (O’Keefe ve McEacher, 1998).

4.14.2. KD SİSTEMLERİ’NİN İÇERDİKLERİ ELEMANLARIN GELECEKTEKİ ÖZELLİKLERİ

Yönetim ve davranış bilimleri ile ilgili araştırmalar sürüp bilgisayar teknolojisi geliştikçe, gelecek kuşak KD Sistemleri de bu yeniliklerden faydalanacaktır. Adriole

³ Firewall, genel ağlardan özel bir ağı ayıran yazılım ve donanımı içeren bir ağ düğümüdür.

(1986), KD Sistemleri'nin gelişimini; donanım, yazılım ve uygulamalar olmak üzere üç kategoride incelemiştir.

Donanım: Adriole (1986) şu anki KD Sistemleri'ni destekleyen donanımı "geleneksel" (merkezi işlem birimleri, disk sürücüler, ışıklı kalemler, dokunmalı ekranlar, vb.) olarak adlandırmış ve gelecek kuşak KD Sistemleri'nin donanım özelliklerini şöyle sıralamıştır:

- Gelecek kuşak KD Sistemleri, daha küçük ve daha ucuz olacak ve böylece kullanımları yaygınlaşacaktır. Bu sistemler birbirlerine bağlanmış olacaklarından daha büyük ve daha küçük sistemlere yüklenebilecektir.
- KD Sistemleri'nin çoğu veri tabanlarıyla, KD ağlarındaki diğer sistemler ile ve dış ağlar üzerinden yapılan geleneksel iletişim sistemleriyle iletişim bağlantısına sahip olacaktır.
- Evrak çantası ve daha küçük boyuttaki KD Sistemleri yaygınlaşacaktır.
- Gelecek KD Sistemleri çeşitli video gösterim sistemleri ile bütünleştirilmiş olabilecektir.

Yazılım: Bu kategorinin özellikleri aşağıda açıklanmıştır.

- Yapay zeka, KD Sistemleri'nin tasarım ve gelişimi için büyük bir umut vermektedir. Doğal dil işlemci sistemleri, KD'in etkinliğini artıracak ve KD Sistemleri'nin geniş yayılımına katkıda bulunacaktır.
- US karar alma sürecini rutin hale getirecektir. Zwass (1998) KD Sistemleri ile US'in birleştirilmesine "akıllı KD Sistemleri" adını vermiştir. Yatırım, yönetim, kaynak tahsisatı ve ofis yönetimi ile ilgili kurallar akıllı KD Sistemleri içine yerleştirilecektir.
- Gelecek kuşak KD Sistemleri ayrıca sosyo-psikolojik çevreye de yardım edebilecektir. KD Sistemleri, kullanıcısı eğer çok fazla hata yapmışsa, sorguları cevaplaması çok uzun zaman almışsa, vb. durumlarda kullanıcısını uyarabilecektir. Böylece kullanıcının girdisine ve davranışına göre sistemin hızı düşürülebilir veya artırılabilir.

- Gelecek kuşak KD yazılımlarının bir kısmı üretici adı taşımayan, problemlere özel yazılımlar olacaktır.

Uygulamalar: KD Sistemleri'nin gerçekleştirdiği uygulamaların gelecekte aşağıdaki özelliklere sahip olacakları düşünülür.

- Profesyonel problemleri çözümlen danışma büroları gibi iyi yapılanmış olacaklardır.
- Kullanıcı için problemleri önceliklendirecek ve bunlardan bazılarını çözümlenebileceklerdir.
- Problem çözümlenme partneri olmaya başlayacaklardır.
- Organizasyonun bütün düzeylerine yerleşmiş olacaklardır.
- Dünyanın farklı yerlerindeki diğer organizasyonlar ile iletişim kurabileceklerdir.

Sonuç olarak, gelecek kuşak KD Sistemleri, kullanıcılarının profesyonel dünyaları ile kişisel dünyaları arasındaki boşluğa bir köprü olacaktır.

4.14.3. BİLGİ DÜNYASINDAN BEKLENEN GELİŞMELER

Mallach (2000), teknolojinin gelişmesiyle gelecekte bilgi dünyasında olacak gelişmeleri şöyle sıralamıştır:

KD Sistemleri, bilgi sistemlerinin standart ve vazgeçilmez bir parçası olacaktır:

Yeni uygulamalar geliştikçe, KD'in gereksinimlerinin de hesaba katılması gerekir. KD Sistemi'nde yapılan değişiklik veri tabanlarının planlanma ve organize edilme yolunu, sisteme girişi kontrol etme yolunu, bir uygulamanın standart arayüzünü kullanma yolunu ve hesaplamada açık sistemler yaklaşımını etkileyebilecektir.

Karar alıcıların bilgisayar kullanımları artacaktır: İş dünyasında kullanılan birçok program bilgi sistemlerine ve dolayısıyla bilgisayara bağlıdır. Bu nedenle karar alıcıların bilgisayarı ve bilgi sistemlerini kullanmayı iyi bilmeleri gerekir.

Donanım teknolojisi gelişmeye devam edecektir: Teknoloji geliştikçe yeni yaklaşımların ortaya çıkacağı ve bunları edinme gerekliliği unutulmamalıdır.

“Yeni” yazılım teknolojileri seli başlayacaktır: Oluşturulan yeni yazılım teknolojileri ilk olarak “öncüler” veya “erken edinciler” tarafından kullanılır. Yazılımlar başarılı olduklarında kullanıcılarına çok kazandırırken, başarısız olduklarında çok kaybettirir. US, ses tanıma ve veri madenciliği “Yeni” olarak adlandırılan yazılım teknolojisindedirler.

Kullanıcı arayüzleri gelişmeye devam edecektir: Karakter tabanlı kullanıcı arayüzleri internet ile birleştirilmektedir. Böylece elde edilen internet tarayıcılı arayüz çok daha fazla uygulamaya erişimi sağlayacaktır.

İş perspektifli iş satışı, bilgi sistemleri profesyonelleri için güçlü olacaktır: Bilgi sistemleri teknolojisi, organizasyonel amaçları daha etkili kılmak için işlerini yeniden tasarlamada diğer sistemlere göre daha fazla imkana sahiptir. Bunun nedeni teknolojinin çok hızlı bir şekilde değişmesi ve BT'nin organizasyonların bütün alanlarına uygulanabilir olmasıdır.

Gelecek kuşak KD Sistemleri'nin özellikleri aşağıdaki gibi sıralanabilir (Sprague ve Watson, 1996).

- Özellikle iyi yapılaşmamış problemler ile ilgili iş gören bilgi çalışanlarının performanslarını artıran çeşitli teknolojilerin birleşimi olacaktır.
- Karar alma ve problem çözmede birlikte çalışacak insanlar arasında gerekli bağlantıyı sağlayabilecek, model tabanlı içinde daha zengin veri kaynakları ve daha fazla “zeka” geliştirebilecek şekilde kurulacaktır.
- Bütün bilgi kaynaklarına giriş, genel bir diyalog arayüzü ile sağlanacak ve bilgi kaynakları bu genel arayüz tarafından çalıştırılacaktır.
- Bilgi çalışanlarının iş istasyonları kullanıcılar ile her zaman işbirliği içinde veya iletişim halinde olacaktır.
- Veri tabanı içindeki iyi yapılaşmış veri, KD için uzun süre faydalı olacaktır. Gittikçe genişleyen bilgi kaynaklarının kümesi KD Sistemleri'nin güç ve etkinliği üzerinde önemli bir etkiye sahip olacaktır.

- Bazı geliřtiriciler, KD Sistemleri'nin yerine US'i geliřtirmeye ve kullanmaya bařlayacaktır.
- US'in bileřenleri ve diđer yapay zeka yaklařımları KD Sistemleri iinde birleřtirilecektir.

Khoong (1995) KD Sistemleri'nin geliřim eđilimleri iin bazı temel tařlar vermiřtir. Bu temel ifadeler izelge 4.5'deki gibidir.

Çizelge 4.5. KD Sistemleri'nin temel taşlarının çeşitleri (Khoong, 1995).

TEMEL TAŞ	KARAKTERİSTİKLERİ	ÖRNEK UYGULAMALARI
1. Araştırma yönetim sistemleri	Kaynak dağıtım kararlarında yardım eden KD Sistemleri	İşgücü nöbet çizelgeleme sistemleri, araç planlama sistemleri
2. Süreç uygulama destek aletleri	Aktivitelerin uygulanmasının kontrolüne yardım eden KD Sistemleri	Ürün süreç kontrol sistemleri, uzay aracı kontrol sistemleri
3. Aktivite planlama sistemleri	Plan ve kontrol aktivitelerini kaynak dağıtım hesabı içine alan KD Sistemleri	Zaman tabloları sistemleri
4. Oyun planlama sistemleri	Bilinmeyen bir çevrede oyuncu için stratejileri planlayan KD Sistemleri	Araştırma ve yardım işlemlerini planlama
5. İş akışı destek aletleri	Organizasyonel iş akışlarını anlayan ve destekleyen KD Sistemleri	Organizasyonel yönetim
6. Ortak oyun simülatörü	Bilinmeyen fakat düşman olmayan çevrede çok çeşitli oyuncuların etkileşimlerini taklit eden KD Sistemleri	Birleşmiş kuvvetler savaş işlemlerini planlama
7. Çoklu aktivite planlama sistemleri	Birbirine bağlı aktivitelerin birçok bağlantılarını planlayan ve kontrol eden KD Sistemleri	Çoklu model taşımacılığını planlama ve kontrol
8. Dağıtılmış kaynak yönetim sistemleri	Dağıtılmış kaynaklar için tahsisat kararlarına yardım eden KD Sistemleri	Küresel yükleme aracı dağıtımı
9. Süreç düzenleme destek aletleri	Organizasyonel iş akışlarını yeniden düzenlemeye yardım eden KD Sistemleri	Sürecin yeniden düzenlenmesi
10. Toplantı planlama sistemleri	Organizasyonel aktivitelerde birlikte çalışan insanlar arasındaki iletişimi planlamada ve kontrol etmede yardım eden KD Sistemleri	Konferans planlayıcılar, savaş koordinatörleri
11. Çoklu kültürel destek sistemleri	Farklı karar alma kültürleri karşısında güçlü olan KD Sistemleri	Uluslararası çevresel koruma sistemleri
12. Uyarlanabilir destek sistemleri	Kullanıcıların karar alma çevrelerindeki değişimlere adapte olmalarına yardım eden KD Sistemleri	Öğrenme organizasyonları
13. Rekabete dayanan oyun stratejileri	Bilinmeyen ve düşman bir çevredeki çeşitli oyuncuların etkileşimlerini taklit eden KD Sistemleri	Ürün satış sistemleri, uluslararası ticari görüşme
14. Rakipli oyun simülatörleri	Bilinmeyen ve düşman bir çevredeki çeşitli oyuncuların etkileşimlerini taklit eden KD Sistemleri	Etkileşimli grup savaş oyunu sistemleri
15. Otonom koordinasyon temsilcileri	İnsanlar ve/veya makineler arasında iletişimlerini koordine eden zeka temsilcileri	Organizasyonlar içindeki gerçek sekreterler
16. Asıl karar alıcılar	Kararlar almada ve hareketi yürürlüğe koymada insanlar adına hareket eden zeka temsilcileri	Organizasyonel bilgi ağındaki asıl karar alıcılar
17. Meta KD Sistemleri	Diğer KD Sistemleri'ni kontrol eden ve eğer gerekliyse KD Sistemleri'ni oluşturan asıl karar alıcılar	Organizasyonel bilgi ağındaki KD Sistemi jeneratörleri
18. Çoklu rakip	Potansiyel olarak düşman zeka temsilcileri içeren çevredeki güçlü (sağlam) KD Sistemleri	Küresel bilgi ana hattındaki asıl organizasyonlar.

BÖLÜM 5

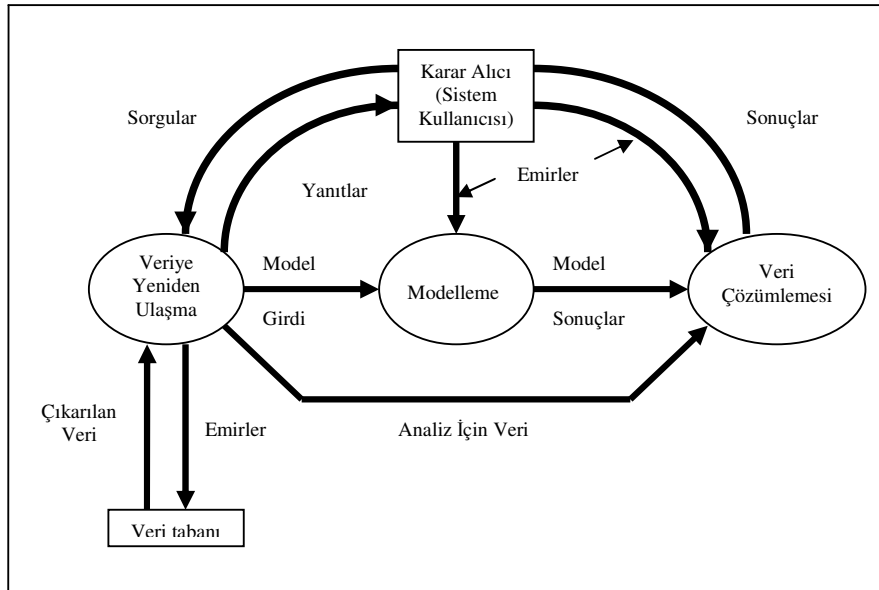
BİLGİ TABANLI KARAR DESTEK SİSTEMLERİ

5.1. GİRİŞ

Bu bölümde Bilgi Tabanlı KD Sistemleri (BTKD Sistemleri) hakkında ayrıntılı bilgi verilecektir.

5.2. BİLGİ TABANLI KD SİSTEMLERİ

BTKD Sistemleri, geleneksel KD Sistemler ile US'in birleşiminden oluşmaktadır. Bilindiği gibi geleneksel bir KD Sistemi; veri yönetimi, modelleme, karar metodolojisi ve sayısal verinin gösteriminden oluşup US'in sembolik düşünme ve açıklama yeteneklerini kapsamaktadır (Klein ve Methlie, 1995). Karar almayı destekleyen bir KD Sistemi örneği Şekil 5.1'de gösterilmiştir.



Şekil 5.1. Genel bir KD Sistemi'nin veri akış diyagramı (Mallach, 1994).

KD Sistemleri içinde bulunan bilgi tabanı sistemlerini geliştirme yöntemleri, zeki KD Sistemleri olarak bilinen KD araçlarının yeni jenerasyonu içinde yavaş yavaş

gelişmiştir. Çeşitli alternatiflerle ilgili olarak bilginin yorumunu sağlayan mevcut KD Sistemi'ni geliştirmede; kural tabanlı düşünme, durum tabanlı düşünme, karışık düşünme olmak üzere üç önemli yaklaşım söz konusudur (Pal ve Palmer, 2000). Sıralanan üç yaklaşımın her biri geleneksel bilgi tabanlı iş KD Sistemleri'nin bazı yönlerini zenginleştirmeye odaklanmıştır. Klein ve Methlie (1995), KD Sistemleri'nin iskeletini US ile birleştirip genişletmede farklı yönergeleri izlemiştir. Bu yönergeler aşağıdaki gibidir.

Özel Bir Problem Alanı İçin Uzman Tavsiyesi: KD Sistemleri, kullanıcıya kavramları, hesaplama işlemlerini, işleyen karar modellerini ve sunulacak raporları tanımlamasında uzman önerilerde bulunmaz, sadece yardımcı olur. KD Sistemleri'ne kullanıcının dikkate aldığı sonuçları veya önerileri sunabilecek bir uzman fonksiyon ekleme, KD Sistemi kullanıcısı ve uzman bir asistanın işbirliği ile mümkün olabilir. Bu yetenek, alan bilgisinin bilgi tabanını ve uzman asistanı, KD Sistemleri'nin bir parçası olarak öneren ayrı bir düşünme mekanizmasını gerektirir.

Uzmanın Sonuçlarını Açıklama: İyi bir KD Sistemi, kullanıcısının öğrenme sürecini geliştirmelidir. US'in açıklama yeteneği sayesinde; kullanıcının ulaşılan sonuçlara olan güveni artar böylece sisteme daha fazla itibar eder, sistemin temelini oluşturan varsayımlar açıkça belirtilir, sistem kolayca hatasızlaştırılabildiğinden geliştirilmesi daha hızlı olabilir.

Karar Çözümlemesi Metodolojisini Destekleyecek Akıllı Asistan: Karar çözümlemesi, karmaşık problemler ile ilgili karar alma sürecini kolaylaştırmaya yardımcı olan güçlü bir tekniktir. Metodolojik bilgiye sahip bir bilgi tabanı, karar alıcıya bir karar modeli veya akış diyagramı tanımlamada, bir olasılık dağılımını belirlemede, değer fonksiyonlarını oluşturmada yardımcı olabilir.

Model Sonuçlarını Açıklama ve/veya Model Davranışı: Basit bir problemin çözümünde bile karmaşık bir veri tabanına gereksinim duyulabilir. Uzman bir sistem, modellerin mantık yapılarını ve modellerdeki değişkenler arasındaki nedensel ilişkileri açıklayabilir. Nicel veya nitel olabilen ve karşılaştırmalı olarak durum değerlendirmesi yapabilen modeller US'i ve "what-if" senaryolarını yorumlayarak, değişkenlerdeki eğilim ve değişimleri vurgulayarak değerlendirme algoritmasını açıklayabilir.

İstatistiksel, Optimizasyonel veya Diğer Yöneylem Araştırması Tekniklerinin Kullanımında Yardım: US acemi bir kullanıcıya, araçların doğru ve yarar sağlayacak stratejiler doğrultusunda kullanımını sağlayarak, büyük veri tabanlardan ilgi alanı bilgisini ortaya çıkararak yardımcı olurlar.

KD Sistemi Kaynaklarının Kullanımında Rehberlik veya Akıllı Kullanıcı Arayüzleri Geliştirme: KD Sistemleri'nin kurulumları veri tabanı, karar modelleri ve rapor sayısında oldukça büyük bir artışa yol açabilir. Veri kaynaklarındaki bu artış, problem çözümü için uygun olanın seçilip kullanılmasını zorlaştırabilir. Uygun veri kaynağının kullanılabilmesi için sistem kullanıcısının eğitimi gerekmektedir. Eğer kullanıcı eğitimi zaman ve/veya parasal açıdan olanaksızsa akıllı kullanıcı arayüzüne başvurabilir. Akıllı kullanıcı arayüzü, kullanıcıya sistemin kaynaklarını doğru bir şekilde seçmesinde ve kullanmasında yardımcı olabilmektedir.

Kesin Soruların Formüle Edilmesinde Yardım: Veri tabanları büyük veya ilişkiler karmaşık olduğunda, kullanıcı isteklerini formüle etmede US'den yardım alınabilir.

Özel Bir Karar Sınıfı İçin Model Kurma Süreci Boyunca Akıllı Destek: Analitik modellerin düzenlenmesi sürecinde uzman bir sistem kullanılabilir. Akıllı bir bileşen bir uzman modelleme danışmanı gibi davranabilme, benzetim modelinin açıklamalarını edinebilme, bilgi alanı içindeki modellerin terimlerini otomatik olarak belgelendirebilme, modelleri otomatik olarak yazabilme, derleyebilme ve çözebilme yeteneklerine sahiptir. Çözümleri uzmanlık gereken özel problemlerin çözümü için karar alma işlemi tamamlandığında, bilgi tabanları ve düşünme biçimindeki uzmanlığa ve açıklama yeteneklerine başvurulabilir (Klein ve Methlie, 1995).

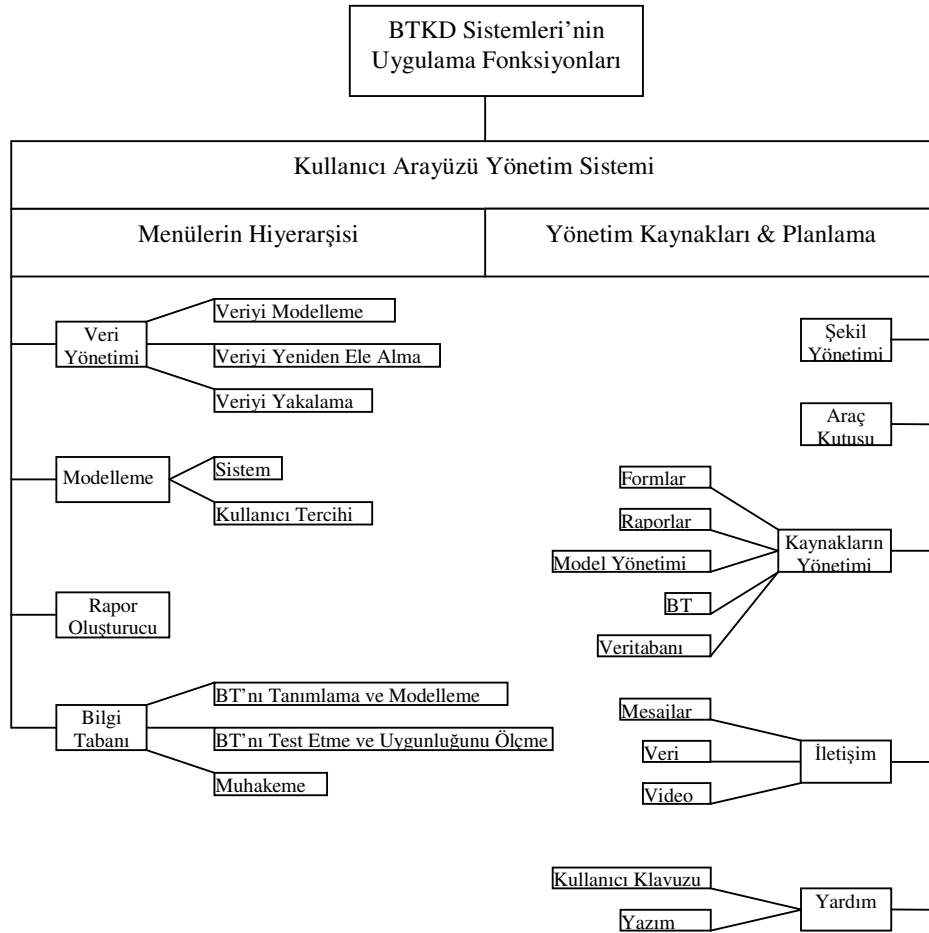
5.2.1. BİLGİ TABANLI KD SİSTEMLERİ'NİN ÖZELLİKLERİ

BTKD Sistemleri aşağıdaki yetenekleri sağlayabilir (Klein ve Methlie, 1995).

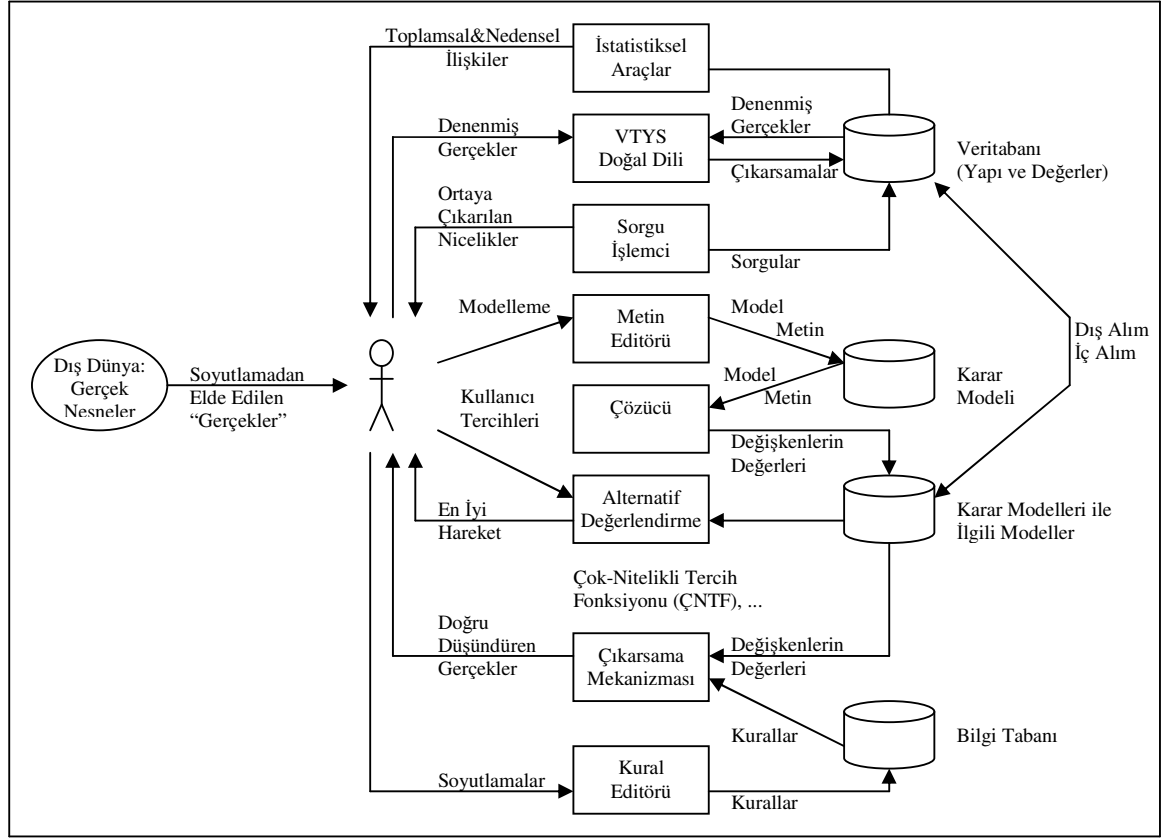
- Problem çözümlene görevi süresince makina-insan işbirliğini destekleyecek bir arayüz
- Problemi çözme sırasında konu ile ilgili bilgiden yararlanma olanağı sağlama
- Problemin tanınması için destek

- Problemin düzenlenmesi için destek
- Problemin formülasyonu ve çözümlenmesi için destek
- Kullanıcıya uzman yardım sağlayacak bir çıkarsama motoru ve bilgi tabanlı yönetim sistemleri
- Modelleme alt sisteminin içerdiği bir düşünme algoritması

Çeşitli bilgi tabanları, gerçekleştirilen görevle ilgili olarak farklı ilgi alanlarında bulunan KD Sistemleri'ne yardımcı olabilir. BTKD Sistemleri'nin fonksiyonel çözümlenmesi Şekil 5.2'de, kullanıcı ile BTKD Sistemleri kaynakları arasındaki tipik etkileşim Şekil 5.3'de verilmiştir.



Şekil 5.2. BTKD Sistemleri uygulamasının fonksiyonel çözümlenmesi (Klein ve Methlie, 1995). (BT: Bilgi Tabanı)

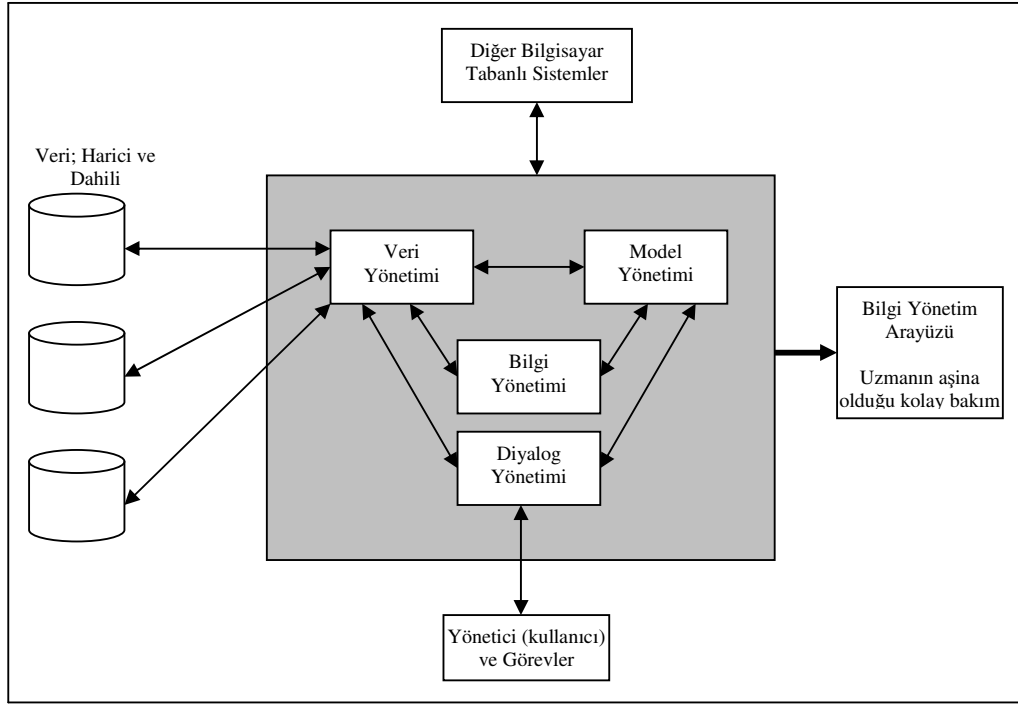


Şekil 5.3. Kullanıcı ile BTKD Sistemleri kaynakları arasındaki tipik etkileşim (Klein ve Methlie, 1995).

5.2.2. BİLGİ TABANLI KD SİSTEMLERİ MİMARİSİ

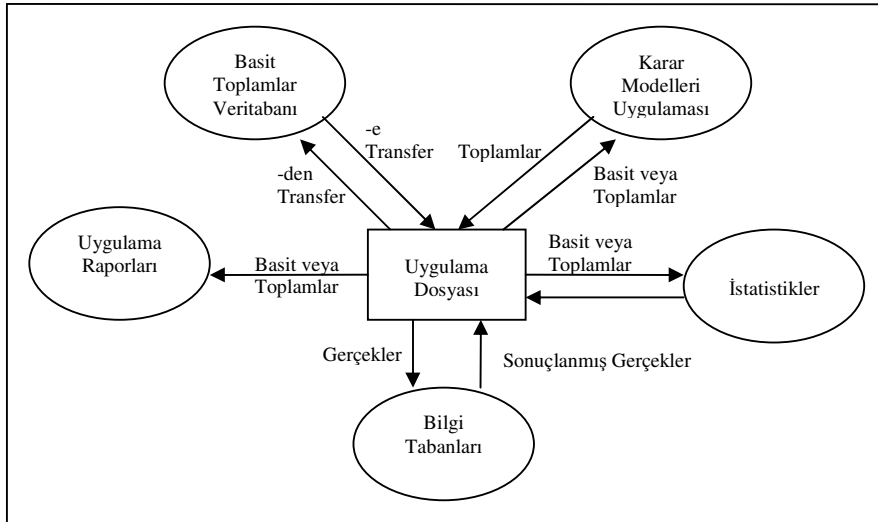
BTKD Sistemleri, bir uzman sistem veya zeka bileşeni olan KD Sistemleri'dir. Turban (1995)'nın geliştirdiği Şekil 5.3'de gösterilen KD Sistemi mimarisine uzman sistem veya zeka bileşeni özelliğinin eklenmesiyle elde edilen BTKD Sistemleri'nin çeşitli bileşenleri Şekil 5.4'de gösterilmiştir.

BTKD Sistemleri ortamı, uzman modül ve KD Sistemleri geliştirme araçlarının geleneksel modülleri arasındaki iletişimi kolaylaştırır. KD Sistemleri içinde bulunan modüller ve bilgi tabanı arasında tam bir iletişim ve bütünleşme istenir (Klein ve Methlie, 1995).



Şekil 5.4. BTKD Sistemleri mimarisi (Turban, 1995).

Veri tabanı içinde bulunan kurallar, bildirilen bir sonuca ulaşmak için modellerin değişken değerleri bilgilerine gereksinim duyar. Kaynaklar ve uygulama dosyası arasındaki bilgi transferi Şekil 5.5'de verilmiştir.



Şekil 5.5. Kaynaklar ve uygulama dosyası arasındaki bilgi transferi (Klein ve Methlie, 1995).

Klein ve Methlie (1995), BTKD Sistemleri'ni, US'in yüzeysel sembolik bilgisini nedensel modelin derin bilgisi ile birleştiren ve mevcut kaynakların seçimine veya

modellerle etkileşimine yardım eden, akıllı kullanıcı arayüzlerini geliştirme yeteneklerini içeren bir bileşen olarak tanımlamıştır.

5.3. AKILLI SİSTEM YAKLAŞIMLARINI GERÇEKLEŞTİRME

Kullanışlı bir akıllı sistemde aranan karakteristikler aşağıda sıralanmıştır (Bielawski ve Lewand, 1991).

- Problem kavramsaldır (hesaplanabilir olmanın tersi): Kesin olmayan veya eksik veri ile düşünmeyi gerektirebilir, genel düşünce tarzını gerektirmeyebilir.
- Problemin konusu yönetilebilirdir: Düşünceleri açık bir şekilde tam olarak ifade edebilmelidir, bilgi alanı dar ve özel olmalıdır.
- Problemin çözümü için kaynaklar mevcuttur: En azından alanında uzman bir kişinin olması, uzmanın, iradesini kullanarak projeyi gerçekleştirebilmesi ve projeye katkıda bulunabilmesi, birden fazla uzmanın gerekliliği durumunda, bütün uzmanların çözüm için belirli bir yaklaşım için görüş birliğine varmaları.
- Bir problemin çözümlenmeye değer olması için aşağıdaki özellikleri içermesi gerekir: Sonuçları zamandan tasarruf sağlıyorsa, verimliliği artırıyorsa, çalışanlar üzerindeki fiziksel riski azaltıyorsa, bilgi ürettiyse, tehlikede bulunan bilgiyi koruyorsa, tutarlılığı sağlıyorsa.

BÖLÜM 6

UZMAN SİSTEMLER

VE

BİLGİ EDİNİMİ

6.1. GİRİŞ

Esas amacı, karar alıcılara özel durumlarda uzman öneri ve bilgi sağlamak olan Uzman Sistemler (US) (Turban, 1995), Yapay Zeka'nın (YZ) alt disiplinlerinden biridir (Turban ve diğ., 2001).

KD Sistemleri, kullanıcıya sonuçlara ulaşmada yardımcı olacak ortam sağlarken, US uzmanlık gerektiren bir alanda çıkarsamalar yaparak sonuçlara ulaştırır (Turban, 1995). Ayrıca, US olarak adlandırılan sistemlerin kullanıcının hatalarını algılama ve yanlışlıkları bularak kullanıcıyı yönlendirme yeteneklerine de sahip olması gerekmektedir (Nabiyev, 2003). US'in çıkarsamalarda bulunabilmesi, açıklamalar yapması ve tavsiyelerde bulunabilmesi için bilgiyi işleme ve saklaması gerekmektedir. Gerçekler ve bir takım kurallardan oluşan bilginin yeri, bilgi tabanıdır.

6.2. US'İN ÇEŞİTLİ TANIMLARI

Olson ve Courtney (1992)'e göre US, uzman bir insanın ulaştığı sonuçlara benzer sonuçlara ulaşmak amacıyla insan düşüncesini taklit etmek için belirli bir miktar YZ gerektiren özel bir alanda kullanılmak üzere geliştirilmiş bilgisayar programlarıdır.

Turban ve diğ. (2001) US, zor problemleri çözümlenmek amacıyla uzmanların düşünme süreçlerini taklit etmeye çalışan, bilgisayara yol gösterici programlar olarak tanımlamıştır.

US, insan gibi mantıklı düşünen bilgisayar programları geliştirecek YZ araştırmalarıyla uğraşan YZ'nin bir alt dalı şeklinde de tanımlanabilir (Raaggad ve Gargano, 1999).

Goodall (1985), US'ı ařađıdaki iki yaklařıma gre tanımlamıřtır.

İnsan/YZ Ynelimli Yaklařım: US, iřlemleri genelde uzman bir insanın yaptıđı tarzda gerekleřtiren bilgisayar sistemleridir.

Teknoloji Ynelimli Yaklařım: US, zel bir alandaki iřlemleri bu alandaki insan uzmanlıđının bir betimlemesini kullanarak, genelde bir uzman insanın yaptıđına benzer řekilde gerekleřtiren bilgisayar sistemleridir.

US'in, sınırlı sayıdaki problem alanları iin uygun olduđu yaklařımda, matematiksel gsterim ve/veya sonulara ulařmada eldeki verilerin yol gsterici olamadıđı durumlarda yargılara dayalı US'in kullanımı uygun olacaktır (Raggad ve Gargano, 1999; Mallach 1994; Goodall, 1985).

6.3. US'İ KAVRAMA

zel bir alandaki bilgiyi, uzman kiřilerin dřnce tarzında sunarak bir uzmanlık dzeyi gerekleřtirme yeteneđine sahip olan US, sonulara ulařmada tamamlanmamıř ve belirsiz veri ile ilgilenebilir ve sistemin dřnce sreci iin iyi bir aıklamayı kapsayabilir.

YZ'yı Dođal Zeka (DZ) ve geleneksel programlarla karřılařtırmak suretiyle, US'in YZ'nın alt disiplini olarak ele alınmasıyla ne kadar avantaj sađlayacađı kavranabilir.

6.3.1. YAPAY ZEKA İLE DOĐAL ZEKANIN KARřILAřTIRILMASI

YZ ile DZ birbirlerine olan stnlkleri bakımından ařađıdaki řekilde karřılařtırılabilir (Turban ve diđ., 2001).

- DZ'ya gre daha kalıcı olan YZ, kullanıcısı deđiřtiđinde veya kullanılan bilgi unutulduđunda bilginin yeniden edinimini sađlar. Bilgisayar sistemleri ve programları deđiřmediđi mddete bilgi aynı kalacaktır.
- YZ, kopyalama ve yayılmayı kolaylařtırır. Bilginin bir yerden bařka bir yere transferi uzun sreceđi gibi bazı bilgilerin transferi mmkn olamayabilir. Bilginin kolay biimde transferi onun bir bilgisayar sistemi iine dahil edilmesiyle sađlanır.

- Kimi durumlarda bir insan uzmandan yardım almaktansa bir bilgisayardan yardım almak daha ucuz olabilir.
- YZ, bir bilgisayar teknolojisi olduğundan tutarlı ve eksiksizdir. Oysa, kaynağının insan olması ve insanların olaylar karşısında genellikle değişken ve tutarsız davranışlar göstermesi nedeniyle DZ değişken ve tutarsızdır.
- YZ, belgelendirilebilir. Sistem aktivitelerinin kopyalanmasıyla bilgisayarın almış olduğu kararlar kolaylıkla belgelendirilebilir.
- İnsanların çeşitli nesnelere birbirleriyle nasıl ilişkilendirileceğini açıklayan becerileri, duyu kalitesi ve örnek modeller arasındaki ilişkilerin farkına varmalarını olanaklı kılan DZ'sı, duygusal deneyimleri doğrudan kullanırken, YZ kullanıcısı dolaylı kullanır.
- DZ, bireysel bir problemin çözümü için oldukça büyük bir deneysel içerik kullanırken, YZ sahip olduğu çok dar odakla çözümü gerçekleştirir. DZ yaratıcıdır. Bilginin YZ sistemine dikkatli biçimde kurulması gerekirken, insanlar doğaları gereği zaten bilgiye sahiptirler.

6.3.2. GELENEKSEL PROGRAMLAR

Problemlerin çözümü için veriyi kullanan algoritmalara veya matematiksel formlere ve bir sonuca ulaştıran ardışık prosedürlere dayalı olan ve BTBS olarak adlandırılan geleneksel programlar, veriyi ve bilgiyi dolaysız bir kavramsal model gibi işler. Belirgin amacı, etkili bir algoritma tasarımı ile etkili sonuçlara ulaşmak olan BTBS, tekrarlı bir süreçtir. Bu süreçte sistemin işleyişi veya sistemin kontrol mekanizması ile sistem bilgisi içiçe geçmiştir (Raggad ve Gargano, 1999; Turban ve diğ., 2001).

BTBS'nde uyulması gereken kurallar, işleyen mekanizma veya sistem kontrolü içinde oluşturulur. Kurallar, program kodunun bir parçası olup talimatların ardışıklık bölümünü oluşturur. Programcının görevi, oluşturulan bu kuralları program algoritması olarak adlandırılan bir düzen içinde sabitleştirmektir. Programcı, kuralları veya alan bilgisini değiştirmek istediğinde programı yeniden düzenlemeye ve bilgiyi doğru yere iletecek yapıları eklemeye veya var olan yapıları değiştirmeye

gereksinim duyabilir. Program veya programların doğru çalışabilmeleri için yapılan değişiklikler sırasında temel adımların unutulmaması ve kullanılan verinin tam ve kesin olması gerekmektedir.

6.3.3. GELENEKSEL PROGRAMLARA SAHİP US'İN KARAKTERİSTİKLERİ

Geleneksel programlara sahip US'in karakteristikleri aşağıdaki şekilde sıralanabilir (Turban ve diğ., 2001).

- US'in kavramsallığını oluşturan kaynak bilgidir. US'e, veri ve bilgiyi kurallara uygun biçimde işleyen bir bilgi edinme bileşeni eklenerek genişletilebilir. Veri ve bilgi US'in doğal kavramsal kaynaklarını oluşturmaktadır.
- US, sahip oldukları bilgiyi algoritmik bir yapı olan belirli bir prosedür içinde sunmaz. YZ, bilginin sembolik işleyişine dayalıdır. Harfler, kelimeler veya objeleri temsil eden sayılar, süreçler ve bunların ilişkileri bilgiyi temsil eder. Bilgi tabanı; gerçekler, kavramlar ve bunlar arasındaki ilişkileri içerir. Süreçler, problemlerin çözümünü sağlayacak tavsiye veya önerileri oluşturmak için kullanılan bilgi tabanındaki sembollerin beceriyle kullanımını sağlar.
- US'in amacı, çıkarsama mekanizmasını destekleyerek etkili bir bilgi betimlemesi sağlamaktır.
- US kontrol bölümünü farklı kurallardan sağlar. Kontrol işlemlerinin büyük bir kısmı çıkarsama motoruyla gerçekleştirilebilirken bilgi tabanı da kurallar veya bilgiyi içerir.
- Bilgi, veri olarak ifade edilir ve US'in kontrolü dışında kodlanır. Bu sayede bilgi kullanıcıya US'den elde edilen sonuçların açıklaması biçiminde geri döner.
- Bilgi tabanında bulunan kuralların sırasının bir önemi yoktur.
- US, açıklama kolaylığı sağlar.
- Bilgi içeren kurallara, istenildiği anda erişim sağlanabilmelidir. US; kusurlu, tamamlanmamış ve kesin olmayan iş kurallarına veya alan bilgisine müsaade edebilir.

- US, bilgi tabanı tamamlanmamış olsa bile kesin olmayan sonuçları üretir.
- US, arama işlemini genellikle karışık deneme-yanılma yöntemi ile gerçekleştirir.
- US, düşünme yeteneğine sahiptir.
- Verinin bakımı ve güncellenmesi oldukça kolaydır. Gerekli değişimler başka bir makineye gereksinim olmadan gerçekleştirilebilir.

6.4. UZMAN BİR SİSTEME NE ZAMAN İHTİYAÇ DUYULUR?

US, problem alanları sınırlı sayıda olan uygulamalar için geliştirilmiş sistemlerdir. US ile gerçekleştirilmesi düşünülen işin alanı uygun biçimde sınırlandırılmış olmalı ve çok sayıda olay kombinasyonu içermemelidir. Yerine getirilmesi düşünülen iş, konuyla ilgili fazla bilgisi olmayan birisi tarafından bile kolaylıkla çözülebiliyorsa, uzman bir sistemin kurulmasına gereksinim yoktur. Uzman bir sistemin kurulmasını gerektiren durumların belli başlıcaları şunlardır (Raggad ve Gargano, 1999; Mallach, 1994):

- Kararın alınması büyük maliyetler gerektiriyorsa.
- Görevin yerine getirilebilmesi için sık sık aynı problem alanlarına başvuruluyorsa.
- En iyi çözüm ile en kötü çözüm arasında büyük farklılıklar varsa.
- US'in test edilip onaylanabilmesi için test verisine erişmek kolaysa.
- Sistemin sonuçları üzerinde genel bir görüş varsa. Sistemden elde edilen çıktılarda, girdilerdeki hatalara göz yumulmuş olabilir. Bu nedenle uzmanların çıktılar üzerinde görüş birliğine varmaları gerekmektedir.
- US'in gelişim süreci boyunca, uzman desteğinin bulunmasını gerektiren bir ihtiyaç olduğu görüşü varsa.
- Problem uzmanlık alanı içinde kolaylıkla özelleştirilebiliyorsa.
- Problemin çözümü bir insan uzmanının uzmanlığını gerektiriyorsa. Burada dikkat edilmesi gereken nokta, sistemin mümkün olduğunca az insan uzmanlığı gerektirecek biçimde kurulmasıdır.

- Problemin alanı iyi biçimde sınırlandırılmışsa.
- Problemin çözümü, ortak duyu mantığından ziyade bilgiye dayalıysa. Çünkü US, bilgiye ve düşünmeyi gerçekleştiren saf sistemlere dayalıdır.
- Çözüm; mümkün, ispatlanabilir ve kısaca oluşturulabilmelidir (gerçekleştirilebilmesi için sadece bir kaç yüz kural içermek).
- Bilgi, kavrama tarzında olmalıdır. Fiziksel aktiviteler kural dışıdır.
- US ile yorum, kestirim, tanı, hatasızlaştırma, tasarım, planlama, görüntüleme, öğretim, kontrol görevleri yerine getirilebilmelidir (Goodall, 1985).

Turban ve diğ. (2001), Goodall (1985)'in yukarıda sıraladığı görev türlerini US'in genel kategorileri olarak nitelendirmiştir. US'in birçoğu bu görevlerden birini esas amaç olarak kabul ederken, diğerlerini yan amaçlar olarak benimser.

6.5. US'İN AVANTAJ VE DEZAVANTAJLARI

Bilgisine danışılan uzman, sistemi kullanacak olan bilgi çalışanları ve US'in kurulması aşamasında gerekli olacak programcılar için bir uzman sistem geliştirme, zaman alıcı bir süreçtir. US, dar bir problem alanına odaklandıkları için sınırlandırılmış olup, alan dışı problemlerin çözümü için kullanılamamaktadır.

6.5.1. US'İN AVANTAJLARI

Goodall (1985), Mallach (1994), Turban (1995) ve Turban ve diğ. (2001)'ne göre akıllı düşünme yeteneği olmayan geleneksel BS'ni kullanmaktansa akıllı düşünme yeteneği olan bir uzman sistem kullanılarak aşağıdaki avantajlar sağlanabilir.

- US, sistemin kalitesini geliştirir. Program, önceden programlanamayan görevleri eksik ve kesin olmayan veriyle bile işleyecektir. Bu yetenek, özellikle bilginin sürekli değiştiği alanlarda, yararlıdır. Önerilerin tutarlılığını sağlamak için hata oranının olabildiğince küçük olması gerekir. Mevcut bilgiye, çoğalan alanlara ilişkin birçok kural eklenebilir.
- US, olasılıklarla açıklanan belirsizlikleri kontrol altında tutabilir. Kontrolün gerçekleştirilmesi, US'in çıkarsama motoruna bağlıdır.

- Önerilerin arkasındaki mantığı açıklamak suretiyle bilgiyi belirginleştirir.
- Uzmanlığı olmayan kullanıcılar için bir eğitim aracı olarak kullanılabilir.
- Kullanılmadığı zaman herhangi bir maliyete neden olmayan US yürütüldüğünde parasal tasarruf sağlar. Maliyetleri azaltmanın bir yolu da insan uzmanlara olan gereksinimleri mümkün olduğunca azaltmaktır.
- Uzmanlar, uzmanlıklarını gerektiren görevler üzerine odaklanabilmeleri için US'i serbestçe kullanabilme imkanlarına sahiptir.
- Özel bir alandaki problemin bilgisini düzenleyebilir ve saklayabilir. Az sayıda uzmanın olduğu veya uzmanın işten ayrılması gibi durumlarda kullanıcılarına çözüm önerisinde bulunur.
- Bilgisayar programcısının verimliliğini artırır. Bilgi tabanı yaratıcısı, sistemin nihai kullanıcılarına göre daha avantajlı tekniklere ve araçlara ulaşabilmektedir. Uzman bilgisinin modüler kurallar biçimine dönüştürülmesi, geliştirme hatalarını azaltır ve sistemin sürdürülebilirliğini geliştirir.
- Yeni bilgi keşfine olanak sağlar.
- Çıktının ve verimliliğin artmasını sağlayabilir. US, insana göre daha hızlı çalışır ve daha çok çıktı elde eder. Çıktılardaki artış ise daha az çalışan ve daha az maliyet anlamına gelir.
- İnsan uzmanların kullandıkları, maliyetleri yüksek olan izleme ve kontrol araçlarına olan ihtiyaçtan kurtarabilir.
- Yararlı bilgiye ve bilgiye erişimi mümkün kılar.
- Yorulmaması veya sıkılmaması ve hiç bir zaman uykuya gereksinim duymaması sonuçları gözden kaçırmaması, büyük boyuttaki veriyi yönetebilmesi hızlı cevap verebilmesi, birçok uzmanın bilgisine sahip olabileceğinden, uzmanlara fazla gereksinim duymaması, daha az hata yapması gibi özelliklerinden dolayı insan uzmanlığına gerek duymayabilir.

US'in, bilgiyi düzenlemek amacıyla kullanılması durumunda sağladığı avantajlar şunlardır:

- Yeni bilgi keşfi (yeni bir asal sayı bulma gibi).
- Bilgi tabanı yaratıcısının gücüyle, uzman kişiyi kurallar ve diğer yapılar konusunda bilgilendirmek üzere, eski bilgiyi düzenlemek ve açıklayıcı hale getirmek. Elde edilen bilginin bu son hali ihtiyaç duyan herkes tarafından istenildiği takdirde kullanılabilir. Bilgiyi düzenleyen ve açık bir hale getiren çeşitli araçlar vardır. Bu araçlardan bir tanesi de “teiresias”dır.
- Düzenlenmiş bilginin öğrenimine yardımcı olmak.
- Bilginin çözümlenmesine yardım etmek.

6.5.2. US'İN DEZAVANTAJLARI

US'in bir önceki kesimde açıklanan avantajları yanında kuşkusuz bazı dezavantajları da vardır. Mallach (1994), Turban (1995) ve Turban ve diğ. (2001) US'i kullanmanın dezavantajlarını aşağıdaki gibi sıralamıştır.

- Uzmanlık alanı genellikle dardır. US genellikle özel amaçlar için tasarlanır ve bu amaçlar dışındaki diğer amaçlar için kullanımları uygun değildir.
- US, genel düşünceyi temsil etmez sadece kurallarını uygular.
- Uzmanlar, sınırlarını zorladıklarının farkına varmadan, uygun olmayan önerilerde bulunabilir.
- US'i geliştirmek, sürecin gerektirdiği insan uzmanlar ve diğer insanlar açısından pahalı olabilir.
- Bilgi her zaman hazır biçimde bulunmayabilir.
- İnsanlardan uzmanlık bilgilerinin alınması zordur.
- Son kullanıcılara olan güven eksikliği, US'in kullanımını engelleyebilir.

- Sayıları az olan ve yüksek ücretler karşılığı çalışan bilgi mühendislerine ihtiyaç duyulabilir.
- Elde edilen sonuçların makul olup olmadığının araştırılmasında uzmanların kullandıkları araçlar farklı olabilir.

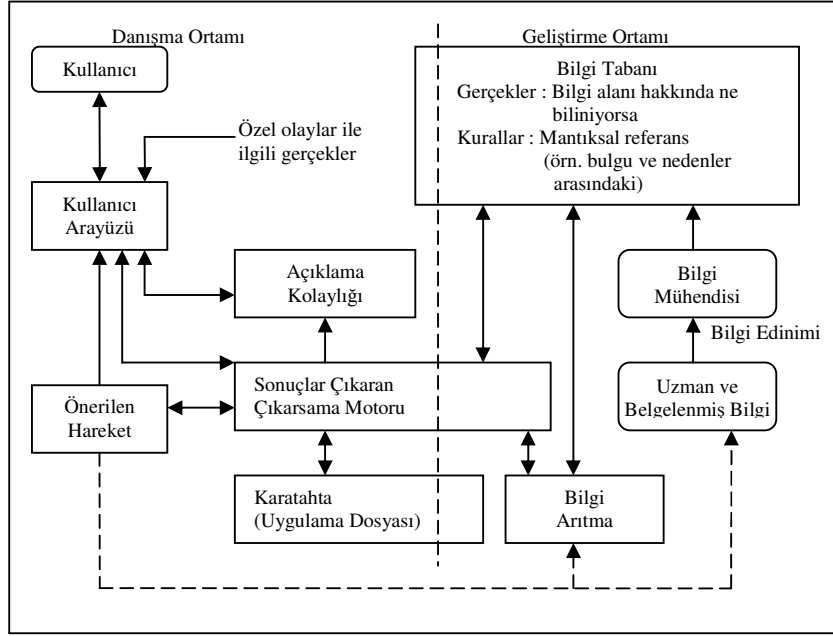
6.6. US'İN MİMARİSİ

6.6.1. US'İN YAPISI

Bilgilerin bir uzmandan bilgisayara ve bilgisayarlardan da kullanıcıya aktarılması için öncelikle bilginin edinilmesi gerekir. Bilginin edinilmesinden sonra bilginin temsili, daha sonra bilgiden sonuç çıkarma en son da bilginin transferi gerekir (Turban ve diğ., 2001).

Bilgi, ya insan uzmanlardan veya belgelendirilmiş kaynaklardan edinilebilir. Edinilen bilgi daha sonra kurallar veya çeşitli veri yapıları şeklinde düzenlenir veya temsil edilir. Kurallar veya yapılara dönüşen bilgi, bilgi tabanı biçiminde, elektronik olarak depolanır. Bu bilgi tabanı şekilleri US'in farklı yönlerinden birini temsil eder. US'i diğer sistemlerden ayıran bir diğer yönü, mevcut bilgiden sonuçlar çıkarmaya yarayan düşünme mekanizmasının bilgi tabanı ve kullanıcı formlarından sağlanan bilgiden elde edilmesidir. Düşünme mekanizması genellikle sonuç çıkarsama motoru olarak bilinir. Çıkarsama motoru, acemilere önerilerde bulunmaya yarar. US'in açıklamaları sayesinde önerilerin anlaşılması, bilginin kullanıcılara ulaşması sağlanır.

US'in farklı bileşenleri ve bunların etkileşimleri Şekil 6.1'de gösterilmiştir.



Şekil 6.1. US'in yapısı (Turban ve diğ., 2001).

6.6.2. US'İN BİLEŞENLERİ

Uzman bir sistemin temel bileşenleri; diyalog yapısı (kullanıcı arayüzü), bilgi tabanı, çıkarsama motoru, karatahta, açıklama alt sistemi, bilgi arıtma sistemidir. Bu bileşenlerden ilk üçü Goodall (1985), son üçü Turban ve diğ. (2001) tarafından oluşturulmuştur.

Kullanıcı arayüzü, sistemin kullanıcı ile iletişimini sağlar. Kullanıcı arayüzleri; kullanıcıya sorular yöneltmek, kullanıcının yanıtlarını ve bir sonuca ulaştıracak kuralları yorumlamak amacıyla kullanılabilir. US üzerinde yapılan incelemelere göre, sistemi oluşturan program kodlarının US'deki bileşenlere göre dağılımı şöyledir (Goodall, 1985):

- Kullanıcı ile iletişim için %44
- Bilgi tabanını düzenlemek için %22
- Çıkarsama motoru için %8
- Diğer işlemler için %26

Açıklama yeteneği, US'in en önemli özelliklerinden biri olup kullanıcı arayüzü içinde gerçekleşir. Turban ve diğ. (2001), açıklama yeteneğini US'in davranışını izleyebilen ve açıklayabilen ayrı bir bileşen olarak açıklamıştır (Şekil 6.1). Bilgi tabanı ve sonuç çıkarsama motoru US'in birlikte çalışan iki ayrı alt sistemidir. Bu alt sistemler US'in ana işlevselliğini sağlar. Bilgi tabanı, özel bir form içindeki özel alan bilgisini temsil eder. Sonuç çıkarsama motoru, kullanıcı girdisine ve bilgi tabanı ve/veya diğer kaynaklardan edinilen gerçeklere dayalı olan bilgi tabanından sonuçlar elde eder veya gerçeklerden sonuç çıkarır.

Kullanıcı Arayüzü: Kullanıcı (bilgi çalışanı veya karar alıcı), US ile kullanıcı arayüzünün sağladığı diyalog yapısı aracılığıyla iletişim kurabilmektedir. Kullanıcı, girdilerini US'e girer ve bu girdileri kullanarak sistemden aldığı sonucu açıklamasını ister. Kullanıcıya verilen çıktı genellikle bilgi tabanından alınan metinleri ve bu metinlerin önceden tanımlanmış cümle formatları biçiminde düzenlenmiş biçimlerini içerir.

Sistemin girdileri doğal dille girilmesine izin vermesi (doğal dilin belirsiz karakteristiğinden dolayı) yanlış girdi girişine neden olabilir. İletişimin doğal dille sağlanması, kullanıcının kullandığı ifadelerin anlamına göre gerçekleşecektir. Kullanıcının sistem tarafından doğru anlaşılabilmesi, kullanıcı ifadelerinin belli bir içeriğe dayanmasına bağlıdır. Sistemin doğal dili anlayabilmesi yeteneği, US'nin kendisine söylenen herşeyi tam olarak anladığı anlamına gelmemektedir. Bunu bir ölçüde sağlayabilmenin en iyi yolu US'in, herhangi bir cümlenin aynı anlamı ifade eden, fakat farklı şekillerde söylenebilecek kombinasyonlarını içermesi ve kullanıcıya bu kombinasyonları sunarak gerçekte ne demek istediğini sormasıdır.

Bilgi Tabanı: US'in gücü ve etkinliği içerdiği bilginin kalitesiyle eşdeğerdir. Sistemin sahip olduğu bilgi, karmaşık problemleri bile çözebilecek düzeyde olmalı ve en iyi yargıyı sağlayabilmelidir. Herhangi bir konu hakkında uzmanlık düzeyinde bilgi edinmek, işlemleri doğru gerçekleştirebilmek açısından çok önemlidir. İnsanlardan bilgi almanın zor, uzman bilginin kaynağının ise uzman kişiler olmasından dolayı bilginin edinimi kimi zaman güç bir süreç olabilmektedir. Hayes-Roth ve diğ. (1983)'a göre bilgi mühendisliği olarak da adlandırılan bilgi edinme;

- Problemin teşhis edilmesi, tanımlanması ve ana karakteristiklerinin belirlenmesi,

- Kavramsallık veya uygun bir sunum ortamının seçimi (örneğin, kurallar ve bilgi edinimi),
- Bir mimari tasarlama veya edinme metodolojisini karşılaştırma (örneğin, kurallar ve uzmanlardan edinilecek bilgi sürecini karşılaştırma),
- Uygulama, bilginin bulunması veya programlanması ve bir prototip geliştirme,
- Bilgi tabanını örneklerle bağlı olarak test etme, arıtma ve bilginin geçerliliğini sınaama aşamalarından oluşmaktadır.

Bilgi edinme yöntemleri; elle, yarı otomatik, otomatik olmak üzere üç şekilde gerçekleştirilebilir. Elle bilgi edinimi, tam olarak yapısal olmayan görüşmelerden oldukça yapısal olan görüşmelere kadar bütün yapısallık düzeyindeki görüşmeleri içeren bir röportaj yaklaşımıdır. Otomatik bilgi edinimi, olayların tarihlerini içeren bir tümevarım sistemini ve bilgi tabanını oluşturacak girdi biçimindeki örnekleri kullanır. Bilgi mühendisinin ve uzmanının üstlendiği görevleri yerine getirir. Otomatik bilgi edinimi makina öğrenimi olarak da bilinmektedir. Bu yolla toplanan bilginin, çözümlenmesinden ve bilgisayarda sunumunun gerçekleştirilmesinden önce kodlanması gerekmektedir. Otomatik bilgi edinme yöntemlerinin onaylanması ve doğrulanması diğer bilgi edinme yöntemlerine göre daha kolaydır. Yarı otomatik bilgi edinimi ise elle ve otomatik bilgi edinimlerinin karışımıdır. Bu yaklaşımlardan hangisinin kullanılacağı bilgi tabanında tutulacak olan bilginin türüne bağlıdır. Bazı bilgi türleri şunlardır:

Yüzeysel Bilgi: Üretim kurallarını kullanan US için temel oluşturan bilgidir.

Deneme-Yanılma Bilgisi: Bildirisel ve prosedürel bilgidir.

Derin Bilgi: Tanımları, aksiyomları, genel kuralları, prensipleri ve bilginin onayladığı nedensel ilişkileri içeren bilgidir.

Bilgi sunumu, insan uzmanın bilgisinin uygun bir ortam içinde kodlanmasının sistematik bir yoludur (Beynon-Davies, 1991). Bilgi; hesap veya formal mantığa dayandırılarak, üretim kuralları formundaki iş uygulamaları, düğümlerdeki bilgiyi veri yapıları şeklinde düzenlemektense grafik şeklinde düzenleyen ve düğümler arasındaki bağlantıları kullanarak gerçekler arasındaki ilişkileri sunan anlamsal ağlar,

herhangi bir yerdeki özel bir nesnenin bütün anlamsal bilgilerini saklayacak veri yapılarını kullanan çerçeveler veya yapısal nesnelere ile sunulabilir.

Mantık, bilgisayarın bilgiyi saklamasını sağlayacak bir yöntem olmasa da, bilginin nasıl saklanacağı konusunda yol gösteren bir araçtır. Mantık, matematiğin bir parçası olup hesaplamalı gösterim ve çıkarsamanın doğruluğu hakkında düşünmeyi sağlayacak çeşitli formlarda kullanılabilir (Goodall, 1995). Mantık formları aşağıda sıralanan yapıları içerir.

- PROLOG (PROgramming in LOGic) gibi programlama dilleri.
- “ve (and)”, “veya (or)”, “değil (not)” bağlaçlarıyla bağlanan temel cümlecikler şeklindeki yapı bloklarını içeren yan görevli mantık veya hesaplamalar.
- Mantığı, basit yapı blok nesnelere ve “is-a”, “has-a” şeklindeki ilişkilere dayandırarak bunlar arasındaki yapıları bulmak. İlişkiler belirleyiciler olarak adlandırılır ve “ve (and)”, “veya (or)”, “değil (not)” mantıksal operatörleri aracılığıyla paylaşılır.

Yukarıda anlatılanlar sistemin düşünme biçimini ve geçerli bilgi sunumlarını saptama yaklaşımlarını biçimlendiren teoriyi oluşturmaktadır. Bu teori, sonuç çıkarsamada kullanılan yöntemlerin doğruluğunun kontrolünde kullanılır.

US’in sıklıkla kullandığı bilgi sunum yöntemleri aşağıda açıklanmıştır.

Kurallar: Kurallar, üretim kuralları olarak da adlandırılmaktadır. Kurallarla düşünen sisteme ise üretim sistemi denilmektedir. Üretim sistemlerinde sonuç elde edebilmek için kuralların zincirleme olarak veri yönetimli biçimde gerçekleştirilmesi gerekmektedir. US’in çoğu bilgiyi kurallar biçiminde sunduğu için bilgi tabanından genellikle kural tabanı olarak söz edilmektedir. Sistemin içerdiği bilginin tümü kurallar biçiminde yazılabilir. Mevcut US’in tümü, sisteme girdi olarak dahil edilecek bilginin kurallar biçiminde yazılmış olmasını gerektirmekte ve bunun için bilgi çalışanlarını yönlendirmektedir.

Kurallar doğal olup, bazı bilgi türlerinin kodlanması (düzenlenmesi) için kullanılabilir tek yoldur. Bilgi sunumu bu yolla gerçekleştirilirken aynı zamanda insan davranışlarının bir modeli de sunulmuş olur. Kurallar, insan uzmanların

kavramsal davranışlarının bir benzetimi olarak da tanımlanabilir. Kuralların bir insan uzman tarafından okunması, anlaşılması ve sürdürülmesi kolaydır. Eğer bilgi, sistemde veri biçiminde tutulup programın kontrol mekanizması içinde kodlanmazsa, sistemin bilgiyi tam olarak anlayabilmesi için sürekli olarak kullanıcıya sorular yöneltilip yanıtlar istemesi kaçınılmazdır. Kuralların kodlanması, esnek ve anlaşılması kolay olan ve “if-then” yapısından oluşan bir yazılım gerektirir.

Bir bilgi tabanında bulunan kurallardan her biri, önemli miktarda uzmanlık bilgisi içerir. Problemlerin çözümü için bu uzmanlık bilgisinin tek başına kullanılmasındansa çıkarsama motoru ile güçlendirilmesi, sonuçların çok daha iyi ve doğru olmasını sağlayabilir. Kurallar birbirlerine bağımlı olduklarından sisteme yeni bir kuralın eklenmesi diğer kuralları etkileyebilir. Eklenen bir kuralın diğer kurallarla çelişip çelişmediğini çözümlmek amacıyla diğer kuralların tekrar gözden geçirilmesi gerekebilir. Kural sistemleri, genellikle basitliklerine göre sınıflandırılır. Bütün kurallar aynı düzeydedir. Kurallar, etkinliği veya araştırmayı geliştirecek, bütün araştırma döngüleri ve araştırmaların kural tabanı segmentleriyle sınırlandırıldığı yapısal kural tabanı sistemleri için kullanılabilir. Bir kural kümesi, kural tabanını anlamaya ve sürdürmeye yardımcı olan problemin farklı yönleriyle ilgili bireysel kuralların bulunduğu bir kümedir. Klein ve Methlie (1995), kural kümesini bir çeşit meta bilgi olarak tanımlamış ve bu bilgiyi bilgi tabanı içine almıştır.

Anlamsal Ağlar: Herhangi bir kural içermeyen bilgiyi anlatmak için sunum yolunu kullanan bir yöntemdir. Anlamsal ağlarda, düğümlerden elde edilen bilgiler bir veri yapısı içinde değil bir grafik içinde düzenlenmektedir. Grafikler, düğüm ve oklardan oluşur. Bağlantılar veya oklar her birinin bir adı olan düğümler arasındaki ilişkileri gösterir. Bu ilişkiler “is-a, has, is, own, needs” biçiminde olup, kavramlar arasındaki ilişkileri etkilemektedir.

Bilgiyi anlamsal ağlar biçiminde saklayan US'in içerdiği sonuç çıkarsama motoru, işleri ilgili bölümlere bölüştürme yeteneğine sahiptir. Bu tip bir çıkarsama motoru iki bölümden oluşur. Birinci bölüm, ileriye doğru zincirleme, geriye doğru zincirleme veya diğer yöntemlerden bazılarıyla ilgili kuralları içerir. İkinci bölüm, gerçeklerden sonuç çıkarmak amacıyla ağ işlemlerini ilgili bağlantılarla eşleştirme işlemini gerçekleştirir.

Çerçeveler: Herhangi bir obje, çok sayıda özellik veya nitelik ile açıklanabilir. Bu özellik veya niteliklerin birçoğu genellikle sabit olup farklı uygulamalarda kullanılabilir. Çerçeve; bir obje, hareket veya olayın tipik karakteristikleri hakkındaki yapısal veri parçasıdır. Kuralların olduğu gibi, çerçevelerin de belirsiz ve kayıp değerleri işleyebilmesi mümkün olmalıdır.

Durumlar: Durum tabanlı düşünme, herhangi bir problemi daha önce çözülmüş benzer problemlerin çözümlerini kullanarak çözen bir süreçtir. Süreç iki adımdan oluşur. Çözülecek probleme uyan ve daha önce çözülmüş benzer problemlerin hafızada bulunması. Bulunan çözümlerin mevcut probleme uyarlanması ve test edilmesidir. Durum tabanlı düşünmenin en önemli bileşeni durum kütüphanesidir. Burada önemli olan nokta, ilgilenilen durumun durum kütüphanesinden bulunup yeniden ele alınmasıdır. Durumların durum kütüphanesinde saklanması indeksler kullanılarak gerçekleştirilmektedir. Saklanan durumların her biri çözümlerini de içerir. Çözümlerin saklanabilmesi için test edilmesi ve testin sonunda başarılı bulunması gerekmektedir. Durum tabanlı düşünmede bilgi edinimi, diğer düşünme yapılarına göre daha kolaydır. Bu düşünme tarzında bilginin sunumu, geçmiş örnekler veya bu örneklerin sonuçları kullanılarak oluşan durumlar şeklindedir.

Sonuç Çıkarılma Motoru: Kural tabanlı sistemlerde araştırma ve düşünme işlemlerini gerçekleştiren mekanizma, sonuç çıkarılma motoru olarak adlandırılır. Kullanıcının, danışma oturumunu başlatmasıyla sonuç çıkarılma motoru aktif hale gelir. Bu motor, belirlenen gerçeklere uyan kuralları bulur, bunlar arasından uygun olan birini seçer ve bu kurala çalışan hafıza yoluyla ortaya çıkarılmış gerçeği ekleyip uygulamaya koyar.

Sonuç çıkarılma motoru, sınırlandırılmış kuralların seçimini örnek eşleştirmesi yaparak gerçekleştirir. Hangi kuralın çıkarılacağına seçiminde ise anlaşmazlık çözümü kullanılır (Klein ve Methlie, 1995). Bu çözüme göre, uygulanabilir ilk kural gerçekleştirilir veya kural üretimi ya da formal mantık kullanılarak çıkarılacak kural belirlenir. Yeni bir gerçeğin çalışan hafızayla sonuçlandırılıp diğerlerine eklenmesi ve bu yeni gerçekle eşleşen yeni örneklerin bulunması gerekir. Bu ardışık kurallar, bağlantılı gerçekler ve kurallar zinciri olarak bilinir. Sonuç çıkarılma motoru, gerçekleri bilgi tabanından çıkarırken kullanıcı girdisini, bilgi tabanı ve/veya diğer

kaynaklardan edinilen gerçekleri kullanır. Gerçeklerden sonuç çıkarma veya bilgi tabanından yorumlar edinmede aşağıda açıklanan üç temel teknik kullanılır.

İleriye doğru zincir: Veri yönlendirmeli olarak da bilinen bu teknik, mevcut durumu bilgi tabanında bulunan kuralların koşullarıyla eşleştirir. Eğer koşul doğru ise sonuç çıkarsama motoru elde ettiği sonucu bilinen gerçekler listesine ekler. Bu teknikte kullanıcı tarafından girilen girdi, kural koşullarının bir çoğu ile eşleşebileceğinden çözüm için çok da etkili bir teknik olmayabilir.

Geriye doğru zincir: Hedef yönlendirmeli veya yönetimli olarak da bilinen bu teknik, hedef sayısı sınırlandırılmışsa ve özel koşulları kullanıcıdan öğrenebilmek için soru sorma şeklinde bilgi edinimi gerçekleştirilecekse etkili olur. Bu teknik daha çok, tanılayıcı görevleri doğrulamada kullanılır. Doğrulama işlemi gerçekleştirirken görevleri etkilediği düşünülen doğruluğu kanıtlanmış gerçeklere geri dönüşler yapar. Bu işlem, kuralların hedef yönlendirmeli geriye doğru zincir işlemi olup sistemin oluşturduğu hipotezi kanıtlamaya çalışır. Geriye doğru zincir tekniği, çok amaçlı hedeflerin söz konusu olmadığı durumlarda koşulları açıklamak ve kanıtlamaya çalışmak için uygun bir teknik değildir. Böyle bir durumda ileriye doğru zincir tekniğinin kullanımı tercih edilmelidir.

Karma zincir: Problem alanının çok geniş olması durumunda hem ileriye hem de geriye doğru zincir tekniklerini kullanan karma zincir tekniği tercih edilmelidir. Bir insan uzman tarafından kullanılan stratejileri modellemede US genellikle karma zincir tekniğini kullanmaktadır.

Karatahta Alt Sistemi: Karatahta, mevcut problemin açıklamaları, girdi verisi ve elde edilen ara sonuçları için çalışan hafızanın bir bölümünün düzenlendiği bir tür veri tabanıdır (Turban ve diğ., 2001). Karatahta, bir sonraki adımda ne yapılacağına dair seçeneklerin, tahminlerin ve kararların tutulduğu, karmaşık bir yapıya sahip bir yöntemdir. Bu yöntem daha çok birden fazla bilgi kaynağının kullanımını gerektiren, anlaşılması güç işlemleri yerine getirmek üzere tasarlanmış sistemlerde kullanılmaktadır. Bilgi tabanlarının sistem tarafından kullanılabilmesi için mutlaka paylaşımına açık bir veri tabanına bağlanabilmeleri zorunludur. Birbirlerinden bağımsız bilgi kaynaklarının iletişimlerini, merkezi bir bilgi kaynağından sağlamaları için kullanılan aygıt karatahtadır. Kullanılmak üzere bir bilgi kaynağına erişildiğinde,

bilgi kaynağının yaptığı ilk iş karatahtanın durumunu gözden geçirmek ve daha sonra içerdiği bilgiyi kullanarak ya yeni bir hipotez oluşturmak ya da mevcut hipotezi değiştirmektir.

Açıklama Alt Sistemi: Turban ve diğ. (2001) açıklama alt sistemini, US'in davranış tarzının açıklanabildiği bir alan olarak tanımlamıştır. Açıklama alt sistemi, sonuç çıkarsama motorunun bir yan ürünü olup yararlı bilginin kullanımıyla elde edilen sonuçları yorumlamayı hedefler. Bu yorumlar, US'in kullanıcılarına diyalog bileşeni içinde sunulabilir. Açıklama alt sisteminde; “Kesin bir sonuca nasıl ulaşılır?”, “Niçin belirli bir soru sorulur?”, “Çözüme ulaşma planı nedir?”, “Niçin kesin olan bir alternatif reddedilir?” gibi sorular yanıtlanabilir.

Bilgi Arıtma Sistemi: Bilgi arıtma sistemi, insan uzmanların kendi performansları hakkında bilgi edinebilecekleri bir sistemdir. Bu yolla uzman, performansını değerlendirip ne düzeyde olduğunu görerek eksik yanlarını iyileştirebilir. US'in başarı veya başarısızlık nedenlerini belirleyip değerlendirebileceği bir programa sahip olması kıymetini oldukça artırmaktadır.

6.6.3. US SÜRECİ

Turban ve diğ. (2001) US sürecini iki bölüme ayırmıştır. Sistem geliştirme ve danışma süreçleri olarak belirledikleri bu bölümler Şekil 6.1'deki gibidir.

Sistem Geliştirme Süreci: US'in kurulduğu süreç olup, bilgi mühendisinin bilgiyi uzmandan edinmesiyle başlar. Bu bilgi daha sonra konu ve bilgi ilişkileri hakkındaki gerçekler olarak “if-then” kuralları şeklinde bilgi tabanı içinde programlanır.

Danışma Süreci: Sistem ile kullanıcılarının birlikte çalışmasını gerektiren danışma süreci, kullanıcının verilerini sisteme girip karşılığında öneriler aldığı bir süreçtir. US'in kullanıcılara verdiği tavsiyeler, sonuç çıkarsama motorunun kullanılmasıyla sağlanır. Sonuç çıkarsama motoru, kullanıcının sisteme girdiği veriyle eşleşen bilgi tabanı içindeki gerçekler ve kuralları arar. Bu işlem bir uygulama dosyası veya geçici veri tabanı olan karatahtanın kullanılmasıyla gerçekleştirilir. Elde edilen bütün ara sonuçlar burada saklanır. Sonuç çıkarsama motoru tarafından gerçekleştirilen işlemlerin bir açıklaması kullanıcıya sağlanabilir. Bilgi tabanındaki bilgi, kullanıcı ile sistemin karşılıklı iletişimi sonucunda rafine edilmelidir.

6.7. US VE UZMAN KABUKLAR

Kabuk, sadece US'deki bilgilerden yararlanarak elde edilmiş açıklama ve çıkarsamaları içeren bir mekanizmadır (Turban, 1995). Beynon-Davies (1991), uzman kabuğu özel alan bilgisi olmayan US olarak tanımlamıştır. Mallach (1994) kabuğu, sonuç çıkarsama motorunun ön paketi olarak tanımlamıştır. US'i kabuk kullanarak oluşturmak, zaman açısından hız kazandırmaktadır (Beynon-Davies, 1991).

6.7.1. KABUKLARIN MİMARİSİ

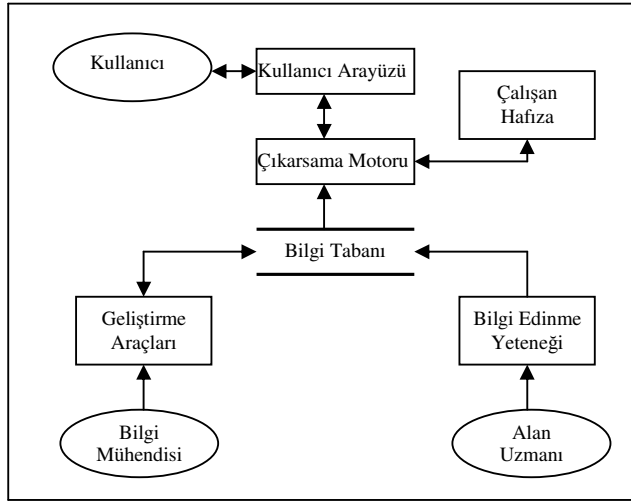
US'de kullanılan bir kabuğun mimarisi aşağıdaki ifadeleri içerir (Beynon-Davies, 1991).

- Özel alan bilgisi için kuralların bulunduğu ve bir depo gibi çalışan bilgi tabanı.
- Bilgi tabanı içindeki bilgiyi kullanarak sistemin sonuçlar elde etmesini sağlayan çıkarsama motoru.
- Çoğunlukla bilgi mühendisleri tarafından kullanılan bilgi tabanını oluşturma ve test etmede yardım sağlayan çeşitli geliştirme araçları.
- Problemin çözümlenmiş kısmi sonuçlarının saklandığı veri alanı olarak adlandırılan çalışan hafıza.
- Kullanıcının US ile iletişim kurduğu ve US'i çalıştırdığı yer olan kullanıcı arayüzü. Açıklama yeteneği US'in en önemli kullanıcı arayüzlerinden biridir.

Kabukların tümü yukarıda sıralanan bu beş bileşeni içerir. US'den bazıları, alan uzmanının uzmanlığını sisteme doğrudan uygulayan bilgi edinme bileşeni gibi başka bileşenleri de içerebilir (Beynon-Davies, 1991).

US'i kabuk kullanarak tasarlamak hız sağlamakla birlikte bu sistemlerin sonuç yollarını sınırlandırabilir. Gerçekte karşı karşıya olunan problemler kabukların destekleyebileceği problem düzeylerinden daha karmaşık olmaktadır. Bu gibi durumlarda PROLOG, LISP veya JESS gibi programlama dillerini kullanmak problemi çözümlenmede destek sağlamaktadır. Ancak destek için programlama dillerinin kullanımı zaman alıcı bir işlem olup tamamen yeni bir uzman sistem ve

yapısının tasarlanmasını gerektirir. Şekil 6.2’de ideal bir uzman sistem kabuğunun mimarisi verilmiştir.



Şekil 6.2. İdeal bir uzman sistem kabuk mimarisi (Beynon-Davies, 1991).

6.7.2. KABUK KULLANMANIN AVANTAJLARI

Beynon-Davies (1991), bir kabuk kullanmanın avantajlarını aşağıdaki şekilde sıralamıştır.

Sistemin Prototipini Hızlandırabilir: Uzman kabuk, sistem geliştiricisinin US’in şeklinden ziyade anlamsal içeriğine, programcının ise diğer bileşenlerden çok sadece bilgi bileşeni üzerine odaklanmasına yardımcı olabilir.

Sistemin Geliştirilmesini Hızlandırabilir: Uzman kabuk, geliştiriciye önceki yapıyı kabul ettirerek, sistemin şeklinden ziyade anlamsal içeriğine konsantre olmasını sağlayabilir.

Ustalık Düzeyini Azaltabilir: Uzman kabuklar sağladıkları yetenekler sayesinde uzman sistem kurucularının sahip olmaları gereken ustalık düzeylerinin bir kısmını azaltabilmektedir.

6.8. BİLGİ EDİNİMİ VE BİLGİ TABANININ KURULMASI

US’in hızlı ve verimli (etkili) olmasını sağlayan en önemli faktör, sistemin içerdiği bilgidir. Bilgi tabanındaki bilginin kalitesi US’in etkinliğini belirler. Bilgi, bilgi mühendisi tarafından edinilir. Bilgi mühendisi sadece bilgi edinmez, ayrıca

problemin tanımlanması, uzman bilginin edinilmesi, mimarinin tasarlanması, programın test edilmesi ve artırılması işlemlerini de gerçekleştirir.

Yukarıda sıralanan işlemlerden en önemlisi uzman bilgi edinimidir. Uzman bilgi edinimi, belirli bir alan hakkında gerek duyulan bilginin çoğunlukla uzman bir kişiden toplandığı, zor ve zaman alıcı bir süreçtir. Bilgi edinildikten sonra toplanan tüm bilgilerin derlenmesi gerekmektedir. Bu işlem için genellikle bilgisayar programları kullanılmaktadır. Uzman bir sistemin bilgi edinimi ve test edilmesi aşamalarında sistemin kurucuları yanında onlarla işbirliği içinde olan uzman bir kişinin de bulunması gerekmektedir. Sistemin doğru çalışıp, yararlı önerilerde bulunabilmesi, içerdiği bilginin doğruluğuna bağlıdır. Bunun için bilginin edinildiği kaynakların son derece güvenilir olduklarından emin olunmalıdır. Aksi takdirde sistem, içerdiği yanlış bilgileri kullanarak yanlış sonuçlar üretecek ve bu sonuçları kullanacak olan karar alıcıların da yanlış kararlar almalarına neden olacaktır. Bilginin hatalı olmasına yol açabilecek durumlar şunlardır:

- Problem alanı ile ilgili ayrıntılı bilgiye sahip olunmaması,
- Problemin çözümlenebilmesi için edinilen bilgiden daha fazlasına ihtiyaç duyulması,
- Bulunan bilginin elde tutulmasına gerek duyulduğunda, karar ortamının değiştirilebilir olması.

6.8.1. İLİŞKİSEL VERİ TABANI SİSTEMLERİ İLE US

Bilginin sunumu, kurallar ve gerçekler olmak üzere iki şekilde gerçekleştirilebilir (Beynon-Davies, 1991). Gerçekler, nesnel arasındaki ilişkileri gösterirken, kurallar eski gerçekleri kullanarak yeni gerçekleri elde etme sürecini tanımlar. İlişkisel Veri Tabanı Sistemleri (İVTS), gerçekler olarak adlandırılan bilgiyi düzenlemede kullanılabilen yöntemlerden biridir. Bu yöntem kullanılarak elde edilen sonuçlar, gerçeklerin bilgi tabanlı sistemler tarafından düzenlendiği sonuçlardan farklı olacaktır. Beynon-Davies (1991), İVTS'nin özelliklerini aşağıdaki şekilde sıralamıştır.

- Veri, veri tabanı olarak adlandırılan bir bileşende saklanır ve tablo veya ilişki olarak adlandırılan bir veri yapısını kullanır.

- Objeler arasındaki bildirisel bilgi veya ilişkileri temsil eden gerçekleri yönetir.
- Kurallar, değerler ve değerler arasındaki ilişkilerin ayrı bir tablo içinde saklandığı bildirisel biçimde temsil edilebilir. Değerler arasında çok sayıda farklı ilişki oluşturulabildiğinden, bu ilişkilerin saklanabilmesi için çok sayıda tabloya gereksinim olacaktır.
- Kısıtlar veya ek tablolar şeklinde gösterilebilen, prosedürel bilgi biçimindeki kurallar veri güvenilirliğini etkiler. Veri tabanı ile iletişim içinde olan yüksek düzeyli bir dil, prosedürel bilgiyi içerir ve yönetir.
- Sınırlandırılmış bir bilgi tabanı sistemi olan veri tabanına eklenen bir kural, veri tabanının yapısını değiştirebilir.
- İlişkisel Veri Tabanı Yönetim Sistemi (İVTYS), bağımsız program verisidir. Veri, kendisini kullanan programdan bağımsız olarak korunabilir.
- Herhangi bir açıklamada bulunabilmesi için yapacağı açıklamaların programlanmış olması gerekmektedir.
- Veri, geçerli olmalıdır. Kurallar, veri tabanının mantıksal sürekliliğini yönetir.
- Veri tabanı, veri tabanı sistemlerini kuran ve VTYS olarak adlandırılan bir aleti kullanır.
- VTYS, bir veri modeli veya mimarisi üzerine kurulur.
- VTYS, veri tabanının ayrılmaz bir parçası olup gerçeklerin büyük bir çoğunluğunu yönetir.

İVT'nın yukarıda sıralanan özelliklerine karşılık Beynon-Davies (1991) US'in sahip olduğu özellikleri aşağıdaki gibi sıralamıştır.

- Gerçekler ve kurallardan oluşan bilgi, bilgi tabanı olarak adlandırılan ayrı bir bileşende saklanır.
- Bilgi sunumuna bağlı olan veri yapısı üretim kurallarını, yapısal objeleri veya çerçeveleri seçer, hesaplamalar veya diğer işlemleri belirler.

- US, mevcut gerçeklerden yeni gerçeklerin üretimini gösteren çok sayıdaki kuralları yönetir. Yeni kural eklemek, diğer kurallardan veya kuralları kullanacak süreçten bağımsız biçimde gerçekleştirilir.
- Prosedürel bilgi şeklinde saklanan ve çoğunlukla veri biçiminde olan kurallar, süreç tarafından, mevcut gerçeklerden yeni gerçekler üretmek için kullanılır.
- Bilgi, hem kurallar hem de gerçekler biçiminde sunulabilir.
- Bilgi tabanının genel amaçlı işlemcisi, işlemsel bilgiyi harekete geçirir.
- US, bağımsız bilgi sürecidir. Kurallar, süreçten bağımsız biçimde sürdürülür.
- Bir kural eklemek, bir koşul veya kuralın silinmesini gerektirebilir.
- US, sonuç çıkarsama ile bilgiyi birbirinden ayırır.
- Bilgi tamamlanmamış olabilir.
- US, uzman kabuk adı verilen bir araç kullanılarak oluşturulabilir.
- Uzman kabuk, bilgi sunumunu şekilsel biçimde gerçekleştirir.
- US'in ayrılmaz bir parçası olan sonuç çıkarsama mekanizması, çok sayıdaki kuralları harekete geçirir.

6.9. US VE INTERNET/INTRANET

Uzman bilgiyi çok sayıdaki kullanıcılara ulaştırma yeteneği, US'i kullanmayı haklı kılan nedenlerden biridir. US'in kalabalık kitlelere ulaşması internet ve intranetin yaygın kullanımı ile ilgilidir. US web üzerinden, sadece insan kullanıcılara değil aynı zamanda robot ve veri tabanları gibi KD Sistemleri'ni içeren diğer bilgisayarlaştırılmış sistemlere de transfer edilebilir. Hatta web kullanılarak US'i oluşturmak da olanaklıdır. Intranet tabanlı GroupWare; kurucular, uzmanlar ve bilgi mühendisleri arasındaki işbirliği sürecini kolaylaştırabilir. Web, multimedia tabanlı US'in kitlelere ulaşmasını destekleyebilir. Bu sistemler ayrıca "Intelimedia Sistemler" olarak da adlandırılmakta olup, kapsamlı multimedia uygulamaları ile US'in birleşimini desteklemektedir.

BÖLÜM 7

KD SİSTEMLERİ, US, BTKD SİSTEMLERİ İNŞA ETME

7.1. GİRİŞ

Bu bölümde KD Sistemleri'ni inşa etmek amacıyla gerçekleştirilmesi gereken adımlar ve ihtiyaç duyulan gereksinimlere yer verilecektir. Ayrıca bölümün sonuna doğru US ve BTKD Sistemleri'nin de inşası için gerekli olan adımlar kısaca özetlenecektir.

7.2. KD SİSTEMLERİ'Nİ KURMA

KD Sistemleri'ni kurmak karmaşık bir süreç olup donanım seçimi, insan-makine arayüzleri ve KD Sistemleri'nin bireyler ve gruplar üzerindeki potansiyel etkisi gibi teknik ve davranışsal konular hakkında bilgi sahibi olmayı gerektirir (Turban, 1995). Bu bakış açısıyla KD Sistemleri'nin kurulmasının pek de kolay olmadığı söylenebilir.

7.2.1. KD SİSTEMLERİ'Nİ KURMA YAKLAŞIMLARI

Bilindiği gibi KD Sistemleri, bir birey veya bir gruba karar almalarında yardım edecek bir yetenekler topluluğudur. Bu nedenle insanların ihtiyaçları değiştikçe KD Sistemleri'nin de değişmesi gerekmektedir. Bu durum dikkate alınarak, geliştirilen KD Sistemleri'nin değişikliklere kolayca uyum sağlayacak özellikte olması sağlanmalıdır. KD Sistemleri'nin değiştirilebilir olup olmaması veya değişiminin kolay olup olmaması kuruluş biçimine bağlı olarak değişim gösterebilmektedir. KD Sistemleri'ni kurmada kullanılan çok çeşitli yaklaşımlar vardır. Bu yaklaşımlar Zwass (1998) tarafından üç kategoriye ayrılmıştır.

- Quick-Hit yaklaşımı
- Geleneksek yaşam döngüsü yaklaşımı

- İteratif yaklaşım

Quick-Hit Yaklaşımı: Quick-Hit yaklaşımında, özel bir KD Sistemi kurmak için belirlenmiş bir ihtiyacın ve yüksek potansiyelli bir faydanın var olması veya problemin çok zor olması gerekir. Bu yaklaşımda maliyet ve risk yüksek olmakla birlikte KD Sistemleri'nin inşası oldukça hızlıdır. Mevcut jeneratörleri kullanabilmesi Quick-Hit yaklaşımının önemli bir avantajıdır. Bu önemli avantajı yanında KD Sistemleri'nin genellikle bir kişi veya bir amaç için kuruluyor olması, diğer KD Sistemleri ile ilişki kurmaması ve deneyimlerin bir sonraki KD Sistemleri'ne iletilmesinde sınırlamanın olması göz ardı edilmemesi gereken dezavantajlarıdır. Hedeflerin ve işlemlerin belli olması, kullanıcıların az, verinin mevcut ve sistemin bağımsız olması durumunda Quik-Hit yaklaşımı tercih edilmelidir (Zwass, 1998).

Geleneksel Yaşam Döngüsü Yaklaşımı: Bu süreç ayrıntılı sistem planlaması ve çözümlemesi ile başlar, kodlama ve sınamayla tasarım aşamasına doğru ilerler ve yürütme işlevine geçer. Süreç uzun olup, sürecin sistemin tamamlanmasından önce çalışacak kısmi bir sistemi yoktur. Geleneksel yaşam döngüsü yaklaşımı, birçok kullanıcıyı etkileyen ve bildirilen isteklerin çözümleme sürecinden önce belirlenebileceği karmaşık sistemler için uygundur. Yaşam döngüsü, genellikle bir KD Sistemi jeneratörü veya bir organizasyondaki çok sayıda fonksiyonel birimleri veya iş süreçlerini etkileyecek model tabanlı çok büyük bir organizasyonel KD Sistemi geliştirileceği zaman kullanılır.

İteratif Yaklaşım: İteratif tasarım süreci, kararlar ve karar alıcılar tarafından ihtiyaç duyulan esneklik ve kısa geliştirme döngüsüne sahip olduğundan, özel bir KD Sistemi kurmadaki en uygun yaklaşım olarak görülmektedir (Sprague ve Carlson, 1982).

Esneklik, hem özel KD Sistemleri'nin kolaylıkla geliştirilebilme hem de kullanıcının karar alma sürecindeki değişiklikleri yanıtlayabilme yeteneği olarak görülebilir. İteratif tasarım, özel KD Sistemleri'nin yinelenen versiyonlarını oluşturan sistem yaşam döngüsünün geleneksel düzeylerini özetler. Bu tasarım, özel bir KD Sistemi'ni geliştirme zamanını azaltan KD Sistemi jeneratörlerinin kullanımını kolaylaştırır. Bütün düzeylerde iteratif tasarım sürecine başvurmanın sonucunda

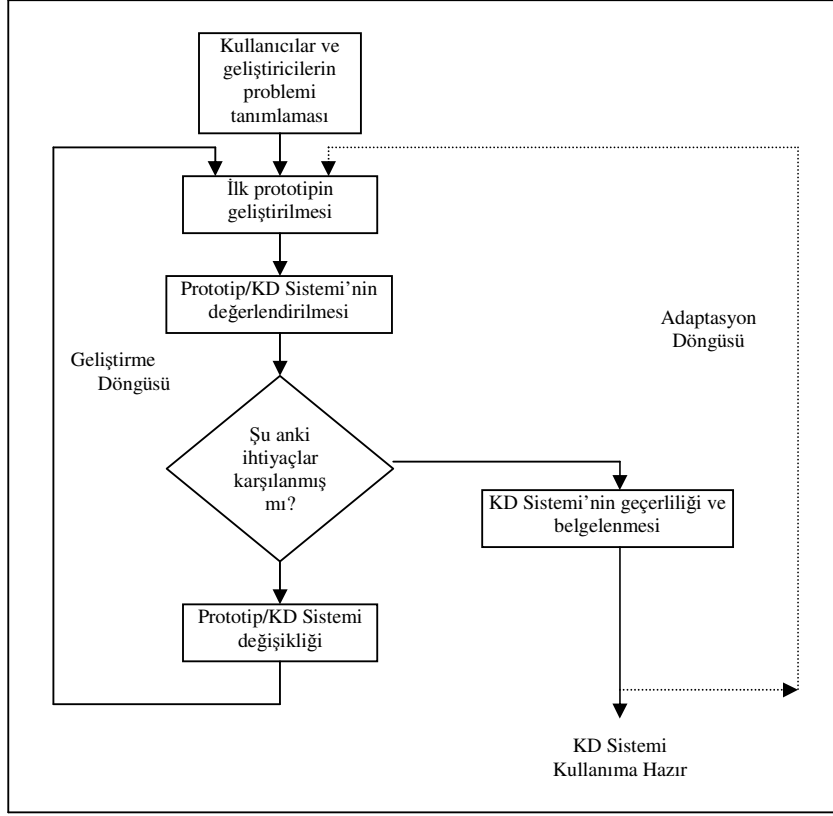
uyarlanabilir bir KD Sistemi elde edilir. iteratif tasarımın öne çıkan özellikleri şunlardır (Sprague ve Carlson,1982):

- Bir alt problem üzerine odaklanması,
- Küçük, fakat kullanışlı bir özel KD Sistemi üzerine odaklanması,
- İlave ya da değişiklik döngülerini planlayabilmesi,
- Sürekli değerlendirme yapabilmesi.

Courbon' a göre iteratif yaklaşım dört aktivite içerir (Turban ve diğ., 1996). Bunlar :

- Önemli bir altproblemin seçimi,
- Karar alıcıya yol gösterecek küçük fakat kullanışlı bir sistemin geliştirilmesi,
- Sistemin değerlendirilmesi ve geliştirilmesi, sonra tekrar değerlendirilmesi,
- Bir döngü içinde sistemin artırılması, genişletilmesi ve değiştirilmesidir.

KD Sistemleri'nin kullanıcıları/kullanıcı grupları genellikle gelecekte sistemden ne isteyeceğini bilemeyebilirler. Ayrıca, sistemin çözümlenmesi sürecinde kullanıcı gereksinimleri çok belirgin olmayabilir. Günlük yaşamda genellikle bir aktiviteden istenilen şey ancak o aktivite yaşama geçirildiğinde öğrenilir. Geliştirilen bir KD Sistemi'nden ne istenildiğini veya ne elde edilebileceğini öğrenebilmek için ise sistemin bir prototipinin oluşturulması ve bu prototipin çözümlenmesi gerekir. Prototip, sistemden istenen özellikler hakkında bilgi edinmeyi ve üzerinde deneyim kazanmayı sağlayan sistemin basitleştirilmiş ilk versiyonudur. Diğer süreçlere göre daha kapsamlı ve sağlam olan bu süreç sistemin gelişimini yavaş yavaş sağladığından birkaç kez tekrar eder. İteratif yaklaşım sürecinin adımları Şekil 7.1'de gösterilmiştir.



Şekil 7.1. KD Sistemleri'nin gelişimi ve adaptasyonu (Zwass, 1998).

7.2.2. KULLANICI

Kullanıcı, genel bir KD Sistemi'nin nihai bir karara ulaşma biçimini etkileyen bileşenlerinden sonuncu olanıdır. Herhangi bir konu ile ilgili kavrayış öncelikleri ile yetenekler ve herhangi bir karara varma, kullanıcıdan kullanıcıya farklılık gösterir. Kullanıcı, bazı sistemlerde yönetici bazılarında ise karar alıcı pozisyonunda olabilir. Kurulacak KD Sistemi'nin kimin tarafından ve hangi amaçla kullanılacağını bilmek sistemin tasarımı için önemlidir. Kimi KD Sistemleri bireyler tarafından kendi kişisel sorunlarına yardım almak için kullanılırken, kimileri bir organizasyon veya grup tarafından daha kapsamlı sorunlara yardım almak için kullanılmaktadır. Kimi sistemler ise hem bireysel hem de grup olarak yardım almak için kullanılmaktadır.

7.2.3. KULLANICI YAPIMLI KD SİSTEMLERE KARŞI HAZIR YAPIM SİSTEMLER

KD Sistemleri ya kullanıcı yapılımlı veya hazır yapılımlıdır. Rutin olmayan ve yapısal olmayan bir problem ile karşı karşıya kalındığında organizasyonun ihtiyaç duyduğu

KD Sistemi, kullanıcı yapımlı olacaktır. Kullanılan birçok KD Sistemi, kullanıcı yapımlıdır.

Farklı organizasyonlarda benzer fonksiyonel problemlerle karşı karşıya kalındığında çözüm için genel bir KD Sistemi kurma yeterli olacaktır. Bu tarz KD Sistemleri hazır yapım KD Sistemleri olarak adlandırılmaktadır.

7.2.4. KD SİSTEMLERİ YAZILIM ARAÇLARI

7.2.4.1. KD Sistemi Yazılım Kategorileri

Herhangi bir yazılımı edinmenin Mallach (2000)'a göre dört farklı yolu vardır:

- Bir hazır paketi satın almak.
- Bir paketi özelleştirmek.
- Eldeki görev için tasarlanmış özelleştirilmiş araçlar veya jeneratörler kullanmak.
- Gerekli programları yazmak.

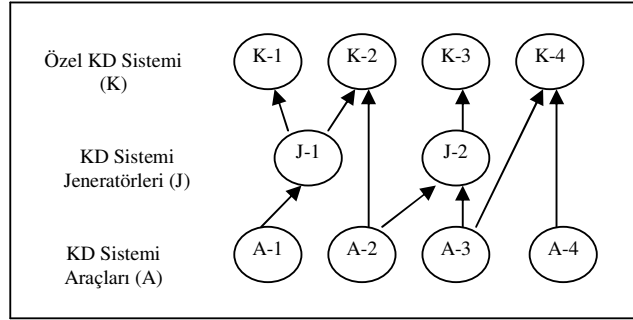
Standart Paketler, özel kararlar almaya yardımcı olmak için geliştirilmiştir. Özelleştirilmiş KD Sistemi aletleri ve KD Sistemi jeneratörleri, KD Sistemi geliştiricilerine standartlaştırılmış “yapı blokları” sağlar. KD Sistemi aletleri genel olarak KD Sistemi jeneratörlerini oluşturmak için kullanılır. KD Sistemi jeneratörlerinin kullanımı ise daha çok işe yönelik daha üst düzey roller ile ilgilidir.

7.2.4.2. KD Sistemi Teknoloji Düzeyleri

Sprague ve Carlson (1982) bir KD Sistemi kurarken kullanılacak teknoloji düzeylerini özel KD Sistemi, KD Sistemi jeneratörleri ve KD Sistemi araçları olmak üzere üçe ayırmıştır.

KD Sistemi araçları, özel KD Sistemi'ni kurmada kullanılan KD Sistemi jeneratörlerinin oluşturulmasında kullanılan araçlardır. Bu araçlar aynı zamanda daha karmaşık özelliklere sahip KD Sistemi araçlarını oluşturmada da kullanılabilir. KD Sistemi jeneratörü kullanarak KD Sistemleri geliştirme yaklaşımı, zaman ve para yönünden tasarruf sağladığından KD Sistemleri'ni finansal bakımından uygulanabilir kılmaktadır. KD Sistemleri'nin bu üç teknolojik düzeyi,

KD Sistemi uygulamalarının inşa sorunlarını anlayabilmek için kullanışlı bir iskelet sağlar. KD Sistemleri'nin üç teknoloji düzeyi arasındaki ilişki Şekil 7.2'deki gibi özetlenebilir. Şekil incelendiğinde, KD Sistemi araçlarının özel KD Sistemi geliştirmede doğrudan kullanılabilirdiği (A-2'den K-2'ye, A-3 ve A-4'den K-4'e uzanan doğrudan oklar); özel KD Sistemi geliştirmek için bir diğer yaklaşımın KD Sistemi araçları ile KD Sistemi jeneratörlerini birlikte kullanmak olduğu (örneğin, A-1'den J-1'e ardından K-1'e uzanan oklar) görülebilir.



Şekil 7.2. KD Sistemleri'nin teknolojik düzeyleri (Turban ve diğ., 1996).

KD Sistemleri kurulum teknolojileri aşağıda kısaca açıklanmıştır.

Özel KD Sistemi: İşi gerçekten başaran “son ürün” veya KD Sistemi uygulaması özel KD Sistemi olarak adlandırılır. Özel KD Sistemi, BS'nin bir uygulaması olmakla birlikte sahip olduğu karakteristikler bakımından diğer tipik veri süreçleri uygulamalarından ayrılır (Sprague, 1980). Sprague (1980)'ye göre özel KD Sistemi, belirli bir karar alıcı veya karar alıcı grubuna benzer problemlerin özel kümeleriyle ilgilenmesine olanak veren yazılım veya donanımlardır.

KD Sistemi Jeneratörleri: KD Sistemi jeneratörü, özel bir KD Sistemi'ni çabuk, ucuz ve kolay biçimde kurmak için gerekli yetenekleri içeren, bütünleştirilmiş bir yazılım paketidir (Turban, 1995). Bu jeneratörler; modelleme, rapor oluşturma, risk çözümlenmesini gerçekleştiren grafikleri oluşturma gibi birçok işlemi gerçekleştirebilen, bütünleşik, kolay kullanımlı paketlerdir. KD Sistemi jeneratörleri kod jeneratörleri gibi de düşünülebilir (Bhargava ve diğ., 1999). İdeal bir KD Sistemi jeneratörü özel amaçlı bir dil olabilir. Özel amaçlı dil, bir KD Sistemi uygulaması oluşturmada veya hesap tablosu teknolojisine dayalı bütünleşik bir yazılım sistemi olarak kolaylıkla kullanılabilir.

Birçok kullanıcı KD Sistemleri'ni geliştirirken KD Sistemi jeneratörlerini kullanmaktadır. KD Sistemleri'ni bir KD Sistemi jeneratörü kullanarak oluşturmak, programlama işleminden ziyade karar çözümlemesi ve matematiksel modellemeyi kullanan karar çözümlemesi işlemlerini azaltmak bakımından yarar sağlayacaktır (Bhargava ve diğ., 1999). Popüler mikrobilgisayar tabanlı jeneratörler Lotus 1-2-3' tür. Express, Excel, Focus, QuattroPro ve IFPS Plus, iyi bilinen jeneratörlerden bazılarıdır. Finansal ve üretim gibi özel KD Sistemi kategorileri için tasarlanmış jeneratörler de mevcuttur.

Bir KD Sistemi jeneratörü aşağıda sıralanan ifadeleri içermelidir (Bhargava ve diğ., 1999).

- Özel yöntemler içeren bir modelleme paradigması,
- Model şemalarını tanımlayan ve veri girişi için kullanılan model ve veri tanımlama özellikleri,
- Modellerin süreklilik, saklanma, düzeltilme ve uygulanma işlemlerini gerçekleştiren, verinin tanımlanması ve kullanımı için uzman bir mekanizma içeren model ve veri yönetim özellikleri,
- Sonuçları etkileyen sonuç değişkenler ve çeşitli kontrol edilebilen değişkenler ve/veya kontrol edilemeyen değişkenler arasındaki ilişkileri araştırmak için analitik yöntemleri ve görsel sunumları,
- Ya genel amaçlı ve/veya özel bir kullanıcı arayüzünü oluşturacak yetenekli kullanıcı arayüzleri,
- Veri tabanları veya harici kullanıcı arayüzleri gibi harici programlarla iletişimi sağlayacak model ve veri takas özelliği.

KD Sistemi Araçları: KD Sistemi araçları, KD Sistemi teknolojisinin temel düzeyi olarak da adlandırılan en alt düzeyi olup özel KD Sistemi veya KD Sistemi jeneratörlerini geliştirmeyi kolaylaştıran yazılım veya donanım elemanlarıdır (Sprague, 1980). Grafikler, editörler, sorgu sistemleri, rastgele sayı üreticileri, hesap tabloları hem KD Sistemi jeneratörlerini hem de özel KD Sistemleri'ni kurmada kullanılabilen KD Sistemi araçlarından biridir.

Sadece KD Sistemi araçlarını kullanarak özel bir KD Sistemi geliştirmek mümkündür. KD Sistemi jeneratörlerinden, çok fazla bir zaman ve çaba kullanımına gerek kalmadan, özel bir KD Sistemi'ni sabit biçimde geliştirmeye olanak tanıyacak bir platform yaratması beklenir (Sprague ve Carlson, 1982). KD Sistemleri'ni jeneratörler kullanmadan sadece KD Sistemi araçları ile inşa etmek, özellikle eğer araçlar kendi kendilerini geliştirmeye gereksinim duyuyorsa, çok uzun ve pahalı bir süreç olabilir.

Mallach (2000), KD Sistemleri'nin geliştirilmesinde kullanılan özelleştirilmiş yazılımları altı ana kategoride toplamıştır:

Veri Tabanı Yönetim Paketleri: Hiyerarşik, ağ veya İVT yapılarından biri ile desteklenen paketlerdir. Hiyerarşik yapı masaüstü mikro bilgisayarlar ile ana bilgisayarlarda, ilişkisel yapı ise mini bilgisayarlar ile iş istasyonlarında yaygın kullanıma sahiptir. Her ne kadar İVTYS, mikrobilgisayarların düzeyini düşürse de ilişkisel yapı beklenmeyen bilginin yeniden düzeltilme isteğini yanıtlamada diğerlerine göre daha çok esneklik sağlar. Bu durum ilişkisel yapının birçok KD Sistemi'nde kullanılabilirliğini mümkün kılar. İVT'nın birçoğuna Yapısal Sorgu Dili (Structured Query Language, SQL) ile erişilir. Ne yazık ki karmaşık sorguların SQL ile oluşturulması zor olup bu tip sorguların çalıştırılması uzun zaman almaktadır. Bu nedenle İVTYS kurucularının çoğu veri tabanlarına erişimi kolaylaştırmak için son kullanıcı sorgu paketlerini de önermektedir.

Bilgileri Yeniden Düzeltme Sorgu ve Raporlama Paketleri: "Bilgiyi yeniden düzelten paketler" in birçoğu özelleştirilmiş kullanıcı verisini bir dosya veya bir veri tabanından çıkarmak için tasarlanmıştır. Bilgiyi yeniden düzeltme yeteneği sisteme VTYS'nin bir parçası gibi dahil edilebilir veya ayrı paketler gibi satın alınabilir. Bu yeteneğin ayrı bir paket olarak satın alınması VTYS içinde oluşturulmasına göre kullanımı daha kolay, daha genel (birden çok veri tabanına erişim sağlama) veya bazı durumlarda daha ucuzdur.

Özelleştirilmiş Model Paketleri: KD Sistemleri'nin çoğu çeşitli modeller içerir. Modeller birçok ortak karakteristiklere sahiptir. Bu nedenden dolayı, standart paketler bilinen tipteki birçok model ile bağlantılı olarak geliştirilir. Bu paketler

ayrıca kullanıcının oluşturduğu modelin karakteristiğini bilgisayara tanımlamayı da kolaylaştırır.

Mevcut çok sayıdaki finansal modelleme araçlarından en basiti “Hesap tabloları”dır. Hesap tabloları hücrelerden oluşur. Her bir hücre metin, sayı veya formül içerebilir. Formüller diğer hücrelerin ve ürünün içerikleri üzerinde işlem yapar. Sonuçlar formlar veya grafikler ile gösterilir. Hesap tabloları birçok nedenden dolayı hata eğilimlidir. Bu nedenle ihtiyaçları karşılamada iki farklı yazılım; “Hesap tablosu denetleme paketleri”, “Finansal model dilleri” kullanılır. Bunlardan hesap tablosu denetleme paketleri hesap tablosunun hatalarını bulmada, finansal model dilleri ise “kayıplı” hesap tablosunu finansal bir model yaklaşımı ile açıklamada kullanılır. IFPS (Financial Modeling Languages) örnek olarak verilebilir.

İstatistiksel Veri Çözümleme Paketleri: Hareketin genel konusu seçilmişse, genellikle bütün kararlar gelecekte ne olacağını tahmin edilmesini gerektirir. Karar almanın öncül temeli, kontrol edilen faktörler üzerinden geleceğe bağımlı değildir. Sistemin bu faktörleri geçmişte nasıl yaptığı belirlenir, gelecekte nasıl tepki göstereceği tahmin edilir ve böylece alınan kararların olası çıktıları ölçülür. Bu ilişkiler çoğunlukla istatistikseldir. Özelleştirilmiş “İstatistiksel Yazılımlar” istatistiksel işlemleri yapmada kullanıcıya yardım eder. İstatistiksel veri çözümleme paketleri; ortalamaların bulunması, yayılımlar, örneklemeler ve korelasyonlar, istatistiksel güven sınırlarını belirleme, çapraz tablo oluşturma, vb. standart istatistiksel işlemleri otomatikleştirir. Burada önemli olan nokta, kullanıcının istatistiksel paket programlarını kullanmadan önce istatistiksel olarak ne yapması gerektiğini biliyor olmasıdır. Bunun için kullanıcının yeterli düzeyde istatistik bilmesi gerekir. İstatistiksel veri çözümleme paketlerine örnek olarak SAS, SPSS, SYSTAT, vb. verilebilir. Sonuç olarak KD Sistemleri istatistiksel paketlere ihtiyaç duyar.

Tahmin Paketleri: Şu an bilinen gerçeklere bağlı olarak gelecek hakkında bilgi vermeye “tahmin” denir. Tahmin, iş hayatında iki farklı anlam kazanır: Birincisi, gelecekte ortaya çıkması olası bir olayın zayıf tahminidir. Müşteri siparişleri, piyango sayı tahminleri buna örnek olarak verilebilir. Bu tip tahminler genelde bazı eski veri tiplerine ve istatistiksel eğilim çözümlerine dayanır. Tahmin edilmesi kolay olmayan, bilinen (veya belki istatistiksel olarak tahmin edilmiş) faktörlerin diğer faktörleri nasıl etkilediğini gösteren bir model tarafından elde edilen

tahminlerdir. İkinci anlamıyla tahmin, istatistiksel tekniklerin bir uygulamasıdır. Bütün endüstriyel yöneticiler tarafından aynı yöntemlerin kullanıldığı gerçeğinden dolayı, gerçek dünya hakkında bilgi veren tahmin yöntemlerine olan ihtiyaç bu yöntemleri içeren paketlerin çok miktarda olmasına yol açmıştır.

Grafik Paketleri: Kararların çoğu, grafikler üzerinde kolayca görülebilen farklılıklara veya eğilimlere dayalı olarak verilmektedir. Hesap tablolarının, istatistiksel ve tahmin paketlerinin birçoğu grafiksel çıktı içerir. Fakat asıl amaçları grafik oluşturmak olmadığından grafik yetenekleri özel grafik paketlerindeki göre sınırlıdır. Özelleştirilmiş grafik paketler, çok sayıdaki standartlaştırılmış veri değişim formatları içinden çıktılarını alabilir ve orijinal paketlerin yapabildiklerinden daha değişik diyagramlar ve grafikler oluşturabilir. Grafik paketlerine olan gereksinim, diğer yazılım çeşitlerinin grafik yetenekleri geliştikçe azalmaktadır.

7.2.4.3. KD Sistemleri'nde Kullanılan Programlama Dilleri

KD Sistemleri'ni geliştirmede kullanılan programlama dilleri üçüncü ve dördüncü kuşak programlama dilleridir. C, Pascal, vb. üçüncü kuşak programlama dillerinden en bilinenleridir. Bu diller, uzun ve hata eğilimli programlama sürecini içerirler. Dördüncü kuşak programlama dilleri; hesap tablosu yazılımları (EXCEL; LOTUS), veri tabanı sorgulama dilleri (SQL), istatistiksel programlar, grafiksel programlar, benzetim, optimizasyon, karar çözümlenmesi, vb. dillerdir. Bu diller, çok esnek olmamalarına ve bilgisayar zamanı bağlamında çok hızlı ve verimli çalışmamalarına rağmen daha hızlı bir şekilde uygulama geliştirmeye izin verirler. KD Sistemleri'nin gereksinimlerini karşılayacak yeteneklere sahip olmasından dolayı bu dillerin, bilgisayarı kullanımlarındaki yavaşlık ve verimsizlikleri göz ardı edilir. Bundan dolayı KD Sistemleri'ni geliştirmede, genellikle üçüncü kuşak programlama dillerinden daha uygundur. Dördüncü kuşak programlama dillerinin çoğu bilgiyi bir veri tabanından yeniden düzeltbilme, bunu sıralama, tablolaştırma ve kişilerin karar alma ihtiyaçlarını destekleyen basit raporlar oluşturma kolaylıklarını içerir.

7.2.4.4. KD Sistemleri Kullanıcı Arayüzleri

Herhangi bir KD Sistemi planlamanın önemli bir bölümünü kullanıcı arayüzü planlaması oluşturur. KD Sistemleri kullanıcıları diğer BS'i kullanıcılarının birçoğunun yaptığından daha değişken ve pek sık meydana gelmeyen organizasyonel

etkisi yüksek ihtiyaçlara sahiptir. KD Sistemi kullanıcı arayüzlerini planlamada düşünölebilecek faktörler;

- Görev performans zamanını minimize etme,
- Öğrenme ve geri çağırma zamanını minimize etme,
- Sistemin çok yönlölüğünü maksimize etme,
- Hataları minimize etme,
- Farklı kullanıcı gereksinimlerine adapte olabilmeyi ve bunlar için yardım isteyenlere yardım etme yeteneğini içerir.

Mimariye eklenen faktörler; kullanıcının birçok şeyi akılda tutma yeteneği, kullanıcı yorgunluğu ve kullanıcının iyi bildiği diğör sistemler ile tutarlı eğilimleridir.

En genel kullanıcı arayüzleri; kullanıcı tarafından çeşitlendirilmiş emirler, kullanıcı için seçimlerin menüleri, kullanıcı için içi doldurulacak boşluklar, grafiksel kullanıcı arayüzleridir. Bu arayüzlerin her biri hem KD Sistemleri'ne hem de KD Sistemleri altında çalışan işletim sistemlerine başvurur. Kullanıcı arayüz teknolojisi her geçen gün daha da gelişmektedir. Yaygın kullanılan arayüzler arasında kalem ile veri girme, ses tanıma hypertext ve hypermedia, vb. sayılabilir.

7.2.5. KD SİSTEMLERİ'NDEKİ ROLLERİN DEĞERLENDİRİLMESİ

KD Sistemleri teknoloji düzeyleri olan özel KD Sistemleri, KD Sistemi jeneratörleri ve KD Sistemi araçlarının üçü de, KD Sistemleri'nin geliştirilmesi ve işlenmesinde kullanılabilir. Yönetici veya teknisyenlerin KD Sistemleri'nin kurulumunda üstlendikleri rollere göre bazı ilginç geliştirme yaklaşımları oluşabilir.

Şekil 7.3'de KD Sistemleri'nin inşasında ihtiyaç duyulan insan rolleri ile KD Sistemleri teknoloji düzeyleri birlikte ele alınmıştır. Bu roller aşağıda sıralanmıştır (Sprague, 1980).

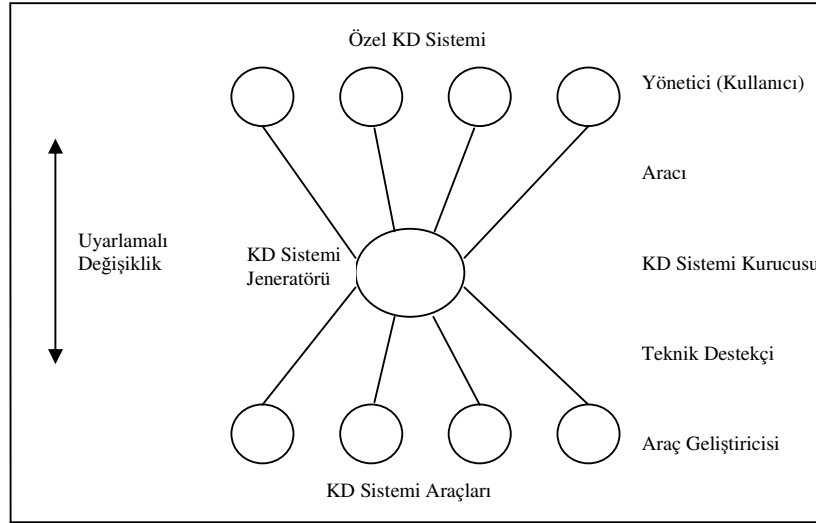
Yönetici ya da Kullanıcı : Herhangi bir problem veya karar ile karşı karşıya kalan kişidir.

Aracı: Kullanıcıya yardım eden kişidir.

KD Sistemi Kurucusu: Kullanıcı veya aracının doğrudan etkileneceği özel bir KD Sistemi'ni oluşturacak yetenekleri KD Sistemi jeneratörlerinden edinen kişidir.

Teknik Destekçi: İhtiyaç duyulduğunda ilave bilgi sistem yetenek veya bileşenlerini jeneratörün bir parçası gibi geliştiren kişidir.

Araç Geliştiricisi: Yeni teknoloji, yeni diller, yeni donanım, yeni yazılım geliştiren ve alt sistemler arasındaki bağlantıların etkinliğini artıran kişidir.



Şekil 7.3. KD Sistemleri'nin üç teknoloji düzeyi ile yönetici ve teknisyenlerin beş farklı rolü (Sprague, 1980).

Yukarıda kısaca açıklanan görevler farklı kişiler tarafından üstlenilebileceği gibi bir kişi birden fazla görevi veya birden fazla görev birkaç kişi tarafından da üstlenilebilir (Sprague, 1980). Sprague (1980)'ya göre görev tahsisi; problemin özelliğine (problemin dar ya da geniş kapsamlı olması), kişinin özelliğine (bireylerin bilgisayar teçhizatı, dili ve genel kavramlarıyla rahat ediş tarzları), teknolojinin gücüne (kullanıcının yönlendiriliş tarzına) bağlıdır.

7.2.6. BİR KD SİSTEMİ TASARLARKEN DÜŞÜNÜLMESİ GEREKEN FAKTÖRLER

Mallach (1994)'a göre, kurulması planlanan KD Sistemi'nin tasarımına başlamadan önce aşağıda sıralanan unsurların göz önünde bulundurulması gerekir.

- KD Sistemi'nden alınması beklenen kararlar ve sağlanması beklenen çıktılar bakımından, ilk olarak KD Sistemi'nin amacının belirtilmesi.

- KD Sistemi'nin kendileriyle iletişim kuracağı ve herhangi bir veri akışını sağlayacağı harici kaynakların tanımlanması.
- Herhangi bir veri dosyasının tanımlanmış olması, tanımlanan bu veri dosyasındaki verilerin harici kaynaklardan elde edilip edilmediği ve bunun harici kaynaklarının belirtip belirtmediğinin açıklanması.
- KD Sistemi içinde bulunan esas süreçlerin belirtilmesi.

Yukarıda sıralanan unsurların tümü, sistem kurucusu tarafından anlaşılabilirse, kurulması planlanan KD Sistemi'nin bir sistem olduğu kanısına varılır.

7.2.7. KD SİSTEMİ GELİŞTİRME PLATFORMLARI

KD Sistemi geliştirme teknoloji düzeylerine dayalı olarak sistem geliştirmek için kullanılan birçok yazılım platformu (stratejisi) mevcuttur. Turban (2005) bu yazılım platformlarını aşağıdaki biçimde sıralamıştır.

- COBOL veya PASCAL gibi genel amaçlı programlama dilleri aracılığıyla özel bir KD Sistemi yazmak.
- Dördüncü kuşak bir KD Sistemi kullanmak: Veriye yönelik diller, hesap tabloları ve mali işlere yönelik diller dördüncü kuşak dillere örnek olarak verilebilir.
- Bir veri ambarı veya büyük bir veri tabanına sahip OLAP kullanılarak KD Sistemi geliştirmek: OLAP motorları sadece çok boyutlu veri küplerini oluşturmakla kalmaz, aynı zamanda etkin biçimde işlev gören çözümlene araçlarını da sağlamaktadır.
- KD Sistem jeneratörü kullanmak: Lotus 1-2-3, Excell ve Quattro Pro KD Sistemi jeneratörlerine örnek olarak verilebilirler. KD Sistemi'nin bireysel dördüncü kuşak dilleri kullanarak geliştirilmesi jeneratörleri kullanarak geliştirilmesinden daha verimli (randımanlı) olmasına rağmen, daha fazla sınırlandırma gerektirmektedir.
- Özel alanlı bir KD Sistemi jeneratörü kullanmak: Özel alanlı KD Sistemi jeneratörleri genellikle fonksiyonel bir alanda oldukça yapısal olan bir sistemi kurmak için tasarlanmıştır. Satış, imalat ve diğer alanlardaki çözümleneleri gerçekleştirmek üzere özel olarak tasarlanmış OLAP sistemlerini içerir.

- Durum metodolojisini kullanmak: KD Sistemleri'ni karşılaşılan durumların zorluklarına göre geliştirme yaklaşımıdır.
- Yukarıdaki yaklaşımların birkaçının birleştirilmesiyle karmaşık bir KD Sistemi geliştirme: Bu yaklaşım özellikle karmaşık KD Sistemleri için uygun bir yaklaşımdır.

7.2.8. KD SİSTEMLERİ'Nİ GELİŞTİRME ARACINI SEÇME

Piyasada KD Sistemleri'nin geliştirilmesinde kullanılacak çok sayıda araç ve jeneratör bulunmaktadır. Ortak özellikleri WEB üzerinden işletilebilirlik olan bu yazılımların bir kısmı sadece büyük bilgisayarlarda, bir kısmı sadece PC'ler üzerinde çalışmaktadır.

Organizasyonlar KD Sistemleri'ni kuracakları araç veya araçları belirlerken birçok soruya yanıt vermek zorunda kalabilir. Turban (2005) bu sorulardan birkaçını aşağıdaki şekilde sıralamıştır.

- Hangi aracın kullanılacağı.
- Kullanımları düşünülen araçlar üzerinden hangi donanımların çalıştığı.
- Hangi işletim sisteminin kullanılacağı.
- Kullanımı düşünülen işletim sistemi üzerinden hangi ağların çalıştırılabileceği.

Donanım Seçimi: KD Sistemleri, bireysel PC'lerden oldukça büyük çok işlemcili bilgisayarlara kadar uzanan çok geniş seçeneklere sahip platformlar üzerinde çalıştırılabilmektedir. Bu platformlardan bir kısmı unix iş istasyonlu ağlar üzerinde çalıştırılabilmekteyken, bir kısmı ise WEB sunucusuna bağlı PC'leri olan bir WEB sunucusu gibi işlev görmektedir (Turban, 2005). KD Sistemleri'ni geliştirmek üzere kullanılacak olan donanımın seçimi genellikle organizasyonun mevcut donanım mimarisine ve kurulacak sistemin her bir kullanıcıya ne kadarlık bir kullanım hakkı tanıyacağına bağlıdır.

Yazılım Seçimi: Turban (2005) aşağıda sıraladığı nedenlerden dolayı KD Sistemi araçlarını veya jeneratörlerini seçme işlemini karmaşık bir süreç olarak nitelendirmiştir.

- Seçimin gerçekleştirileceği sırada, KD Sistemleri'nin bilgi gereksinimlerinin ve çıktılarının tamamen bilinmesi mümkün olmayabilir.
- Piyasada satışa sunulan çok sayıda KD Sistemi geliştirme yazılım paketlerinin mevcut olması.
- Yazılım paketlerinin çok hızlı biçimde güncellenmesi.
- Fiyatların çok sık değişmesi.
- Yazılım paketlerini değerlendiren kişilerin çok sayıda olması.
- Herhangi bir dilin birçok KD Sistemi'nin kuruluşunda kullanılabilmesi. Böylelikle geliştirilen araçların bazı yeteneklerinin bir uygulamadan diğerine değişmesinin söz konusu olabilmesi.
- Büyük bir KD Sistemi'nin çeşitli bölümlerinin farklı araçlar ile geliştirilebilmesi.
- Seçilen karar, karşılaştırılan paketlerinin yeteneklerine bağlı olarak birçok kriter gerektirebilir. Kriterlerin çoğu fiziksel bir varlığa sahip değilken; diğerleri bir başkası ile doğrudan uyumsuzluk içinde olabilir.
- Teknik, fonksiyonel, son kullanıcı ve yönetimsel amaçların tümü düşünülür.
- Data Decisions, Data Pro ve Software Digest Inc. gibi şirketler tarafından yapılan değerlendirmelerin, KD Sistemi içermeyen uygulamalar için bile, subjektif ve genellikle yüzeysel olmaları.
- Birkaç satıcıya bağlı olmanın çekiciliği ile birçok satıcıdan edinilen ürünlerin birleştirilmesine ve karşılaştırılmasına izin veren alıcı/satıcı açık sistem ortamlarının olmadığı göz önünde bulundurulması.

Bir KD Sistemi Jeneratörünün Seçimi: Organizasyonlar KD Sistemi jeneratörlerini genellikle KD Sistemi uygulamaları için kullanmaktadır. Sadece bir jeneratörle KD Sistemi kurma kısıtlaması şirketler için söz konusu olmadığından birden fazla jeneratörün kullanımı mümkün olabilmektedir.

7.2.8.1. Bileşenlerin Biraraya Getirilmesi

Organizasyonların bir veya birkaç geliştirme platformu kullanması yazılım seçimlerini daha da kolaylaştıracaktır (Turban, 2005). Turban (2005), bileşenlerin kolaylıkla yeniden biraraya getirilebilmelerinin, bu bileşenlerin tasarımlarına bağlı olduğuna işaret etmiştir. Eğer bileşenler standart görevlerini yerine getirecekleri şekilde tasarlanabiliyorlarsa, bu bileşenlerin yeniden biraraya getirilebilmeleri mümkün olabilir. Diğer taraftan bileşenlerin yönetimi ve birleşimi çok karmaşık bir süreç de olabilir. Yeniden biraraya getirilebilirlik, bilgi yönetiminin temel dayanak noktalarından biri olmasına rağmen kod geliştirme açısından mikro düzey oluşturmaktadır.

7.2.8.2. Kaynak Dışı

KD Sistemleri'ni geliştirmek için herhangi bir yazılım seçilirken bu yazılımın kaynaklar içinde veya kaynaklar dışında olup olmayacağı gözönünde bulundurulmalıdır.

Kaynak dışıyla ilgili esas karışıklık, yazılımın geliştirilmesini isteyen ve bunu yabancı olmayan bir şirket ile geliştiren organizasyonun, ürün tamamlanmamış olsa bile sorumluluğuna sahip olmasıdır (Turban, 2005).

Kullanıcılar, yöneticiler, yürütücü sponsorlar olarak sıralanabilecek organizasyon üyeleri, organizasyonun sistem için güvenilir olduğu sürece sistemin sanki kaynaklar içindeymiş gibi geliştirilmiş olmasını isteyebilir (Turban, 2005).

7.2.9. BİR KD SİSTEMİ JENERATÖRÜ YARATMA

Sprague ve Carlson (1982)'a göre KD Sistemi jeneratörleri kullanım kolaylığı, erişim kolaylığı ve çözümlene yeteneklerine sahip olmalıdır. Aşağıda bu yeteneklere kısaca değinilmiştir.

Kullanım Kolaylığı: KD Sistemi jeneratörleri, teknik bilgisi olmayan insanların bile aktif ve denetlenebilir bir şekilde kullanabilsini sağlayacak basitlik ve uygunlukta olan özel bir KD Sistemi'ni yaratabilmelidir. KD Sistemi jeneratörleri, kurucuların özel bir KD Sistemi kurabilmeleri ve değiştirebilmelerine olanak verecek kolaylık ve uygunlukta olmalıdır.

Erişim Kolaylığı: KD Sistemi jeneratörleri, çok sayıdaki kullanıcılar, problemler ve içerikler için problem çözümlenmeyi ve karar almayı desteklemek amacıyla geniş veri kaynaklarına erişimi sağlayabilmelidir.

Çözümleme: KD Sistemi jeneratörleri, çok sayıdaki kullanıcılar, problemler ve içerikler için problem çözümlenmeyi ve karar almayı destekleyecek çözümlenmeleri sağlayabilmelidir.

7.2.10. KLASİK GELİŞİM DÖNGÜSÜ İLE PROTOTİP'İN KARŞILAŞTIRILMASI

Daha önce açıklandığı gibi, KD Sistemleri'ni geliştirme teknolojilerinden olan iteratif tasarım süreci, kararlar ve karar alıcılar tarafından ihtiyaç duyulan kısa geliştirme döngüsü kadar esnek bir yapıya sahip olduğundan, kısa geliştirme döngüsünden daha iyi bir alternatif olarak öne çıkmaktadır (Sprague ve Carlson, 1982). Prototip oluşturma süreci, genellikle kurucular ve son kullanıcılar arasında oluşabilecek yanlış anlaşılmalara bir açıklık getirir. Bu süreçte, KD Sistemleri'nin gelişimi sürdükçe, kurucular ve kullanıcılar arasında karşılıklı bir öğrenme süreci gerçekleşir. Sistemin prototipinin oluşturulmasının avantajları şunlardır (Turban, 1995).

- Geliştirme zamanı kısalmır.
- Kullanıcının reaksiyon gösterme (kullanıcıdan geribildirimde bulunulması) zamanı kısalmır.
- Kullanıcıya sistemin gereksinim duyduğu bilgiyi ve sistemin yeteneklerini bildirmek suretiyle kullanıcının sistem hakkındaki algılayışında gelişme sağlanır.
- Maliyet düşer.

7.2.10.1. KD Sistemleri'nin Klasik Gelişim Döngüsü

Klasik bir KD Sistemi geliştirme süreci, karmaşık bir KD Sistemi geliştirme sürecinin bütün aktivitelerini içerir. Bu sürecin şematik gösterimi Şekil 7.4'de verilmiştir. Şeklin anlaşılabilmesi bakımından gelişim döngüsünün unsurları aşağıda açıklanmıştır.

A aşaması (Planlama): Planlama, çoğunlukla ihtiyaç değerlendirmesi ve problem teşhisi ile ilgilidir. Bu aşamada KD'in amaç ve hedefleri belirlenir.

B aşaması (Araştırma): Araştırma aşaması, kullanıcı ihtiyaçlarını ve mevcut kaynakları (donanım, yazılım, satıcılar, sistemler, diğer organizasyonlar içindeki araştırmalar veya benzer tecrübeler ve konuyla ilgili araştırmaların gözden geçirilmesi) belirlemek için konuyla ilgili bir yaklaşımın oluşturulmasını içerir.

C aşaması (Çözümleme ve Kavramsal Tasarım): Bu aşama en iyi yaklaşımın belirlenmesini ve bunun gerçekleştirilmesi için gerekli olan özel kaynakları belirlemeyi içerir. Ayrıca; teknik, personel, finansal ve organizasyonel kaynakları da içermektedir.

D aşaması (Tasarım): Bu aşamada sistem bileşenlerinin ayrıntılı tanımlamaları yapılır, yapı ve özellikleri tanımlanır. KD Sistemleri esas bileşenlerine göre veri tabanı ve veri tabanı yönetimi, model tabanı ve model tabanı yönetimi, bilgi yönetimi alt sistemi, diyalog alt sistemi parçalarından oluşur.

E aşaması (İnşa): İnşa, tasarımın teknik olarak yerine getirilmesidir.

F aşaması (Yürütme): Yürütme aşaması aşağıdaki görevleri içerir:

Test: Sistemin çıktılarından elde edilen veriler toplanarak tasarım aşamasında yapılan tanımlamalarla karşılaştırılır.

Değerlendirme: Yerine getirilen sistemin kullanıcı gereksinimlerinin ne kadarını karşıladığının değerlendirilmesidir.

Gösterim: Kullanıcılara sistemin gerçekleştirebileceği işlemsel yeteneklerin gösterimidir.

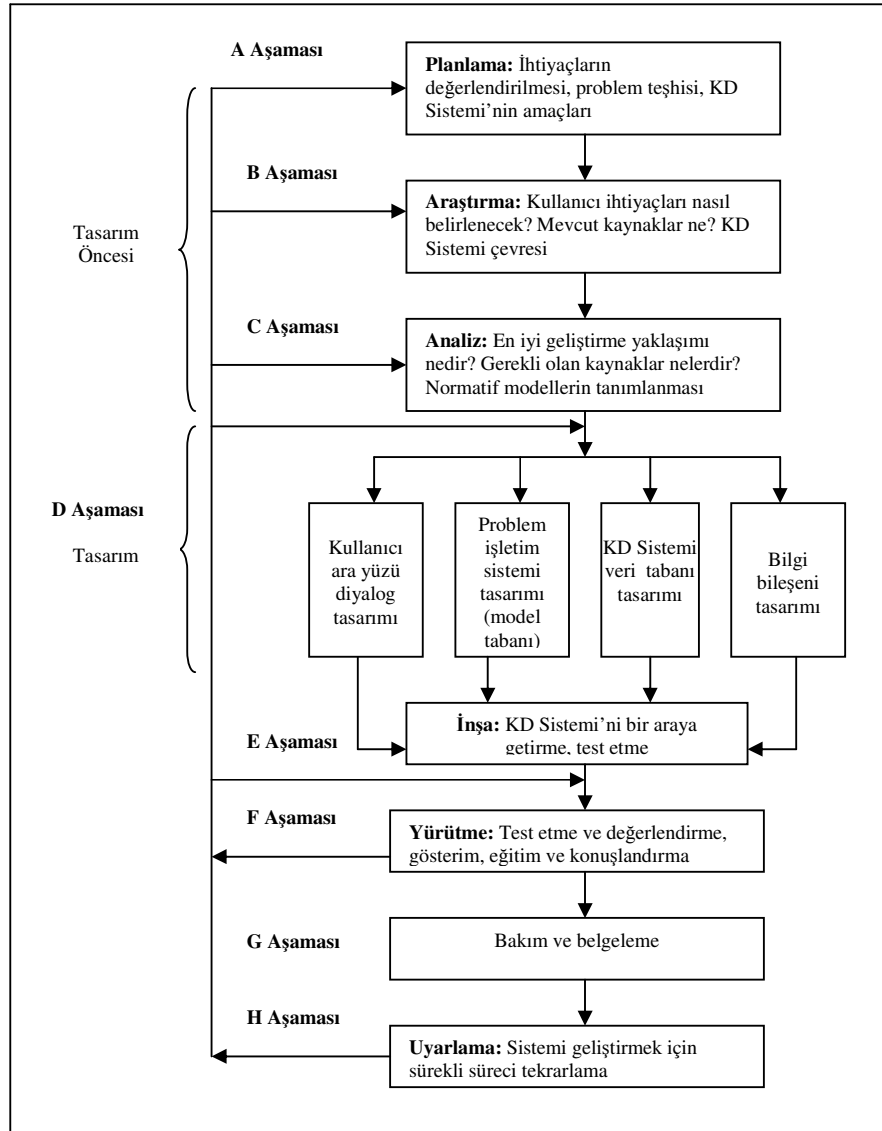
Yönlendirme : Yönetimsel kullanıcıların öğretimini içerir.

Eğitim: İşlemsel kullanıcıların eğitimidir. Ayrıca sistemi onaracak kullanıcıların eğitimini de içerir.

Yaygınlaştırma: Tam sistemin bütün kullanıcılar tarafından kullanılabilmesidir.

G aşaması (Bakım ve Belgeleme): Bakım, kullanılan sistemin planlanmasını ve kullanıcılarının eğitimini içerir. Belgeleme ise sistemin bakımı ve kullanımı için uygun belgelerin oluşturulmasıdır.

H aşaması (Uyarılama): Uyarılama, değişen kullanıcı gereksinimlerine bağlı olarak yukarıda açıklanan önceki adımlara geri dönüşü sağlar.



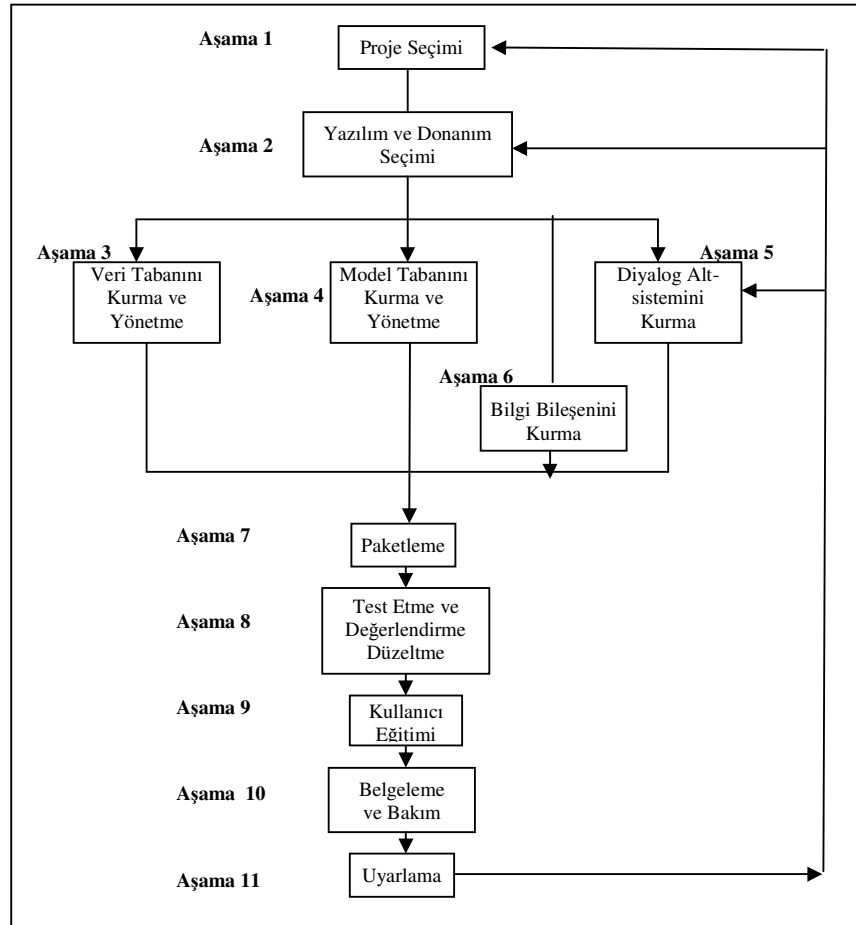
Şekil 7.4. Bir KD Sistemi kurma evreleri (Turban, 1995).

7.2.10.2. Son Kullanıcılar Tarafından Geliştirilen KD Sistemleri'nin Gelişim Döngüsü

Yarı yapısal veya yapısal olmayan problemler değişken yapıda olduklarından, kendileri için tasarlanan KD Sistemleri'nin de değiştirilebilir olması gerekir. Bu

nedenle birçok KD Sistemi, BS'nin geleneksel yaşam zinciri gelişim sürecinden farklı bir yapıya sahip olan ve bu değişimlere kolaylıkla uyarlanabilen model süreci tarafından geliştirilmiştir. Burada sözü edilen gelişim süreci bütün aktiviteleri içeren bir KD Sistemi gelişim sürecidir. Bu süreç aynı zamanda klasik KD Sistemi gelişim süreci olarak da bilinmektedir (Turban, 1995).

Son kullanıcılara, kendilerinin veya içinde buldukları grubun karar gereksinimlerini karşılayacak özel bir KD Sistemi geliştirme olanağı verildiğinde, son kullanıcıların görüşüne uygun bir inşa etme sürecinin izlenmesi daha doğru olacaktır. Turban (1995), son kullanıcıların görüş açlarına göre geliştirdiği bir inşa etme sürecini Şekil 7.5'deki gibi özetlemiş, bu sürecin aşamalarını ise aşağıdaki gibi sıralamıştır.



Şekil 7.5. Son kullanıcılar tarafından inşa edilmiş bir KD Sistemi geliştirme süreci (Turban, 1995).

Birinci Aşama (Proje veya Problemin Seçimi): Uygun bir çözüm bulacak süreci gerçekleştirmek için organizasyon içinde bölümlere ihtiyaç vardır.

İkinci Aşama (Yazılım ve Donanım Seçimi): Bu aşama, uygun yazılım ve donanımları seçmeyi kapsamaktadır.

Üçüncü Aşama (Veri Edinme ve Yönetme): Verinin bilgi tabanından edinilmesini ve korunmasını içerir.

Dördüncü Aşama (Model Alt Sistemi Edinme ve Yönetme): Model tabanı kurma aşamasıdır. Model tabanından, ilgilenilen modelleri edinme ve kullanmayı içerir.

Beşinci Aşama (Diyalog Alt Sistemi ve Yönetimi): Kullanıcı arayüzü geliştirme aşamasıdır.

Altıncı Aşama (Bilgi Bileşeni): Bilgi mühendisliğini gerçekleştirir.

Yedinci Aşama (Paketleme): KD Sistemi'nin çeşitli yazılım bileşenlerini test edilmesi ve kullanım için bir araya getirilmesidir.

Sekizinci Aşama (Test Etme, Değerlendirme ve Geliştirme): Örnek bir girdi kullanımıyla KD Sistemi'ni test etme ve test sonucunda KD Sistemi'nin güvenilir olduğunu görmek suretiyle testi geçerli kılmaktır.

Dokuzuncu Aşama (Kullanıcı Eğitimi): Kullanıcıların KD Sistemi'ni doğru kullanabilmeleri için eğitilmeleridir.

Onuncu Aşama (Belgeleme ve Bakım): Belgelerin üretilmesini ve KD Sistemi'nin bakımını içerir.

Onbirinci Bölüm (Adaptasyon): KD Sistemi'ni kullanıcının gereksinimlerine uyacak biçimde uyarlamayı içerir.

7.3. BİR US KURMA

Herhangi bir US projesine başlamadan önce sistemin esas amacı, kısıtlamaları, mevcut destek yetenekleri, uzman kişilerin varlığı, kullanıcı etkisinin güvenilirliği, sürdürülebilirliği, gereksinilen çözüm ve uygulama gibi birçok faktörün araştırılıp belirlenmesi gerekmektedir. Görevini tam olarak yerine getirebilen başarılı bir uzman

sistem kurmanın esası, bilgi tabanını güncelleme ve geliştirmeye ve çıkarsama motorunun gücünü artırmaya dayanmaktadır (Raggad ve Gargano, 1999). US'in kullanım süreci boyunca, etkisinin sürekli olarak değerlendirilmesi ve problem alanının değişip değişmediğinin test edilmesi gerekmektedir. Problem alanı değişen bir uzman sistem, kullanıcıya yanlış önerilerde bulunacak ve yanlış sonuçlar verecektir.

7.4. BİR BTKD SİSTEMİ KURMA

Bir BTKD Sistemi kurmak, hem KD Sistemleri'nin hem de US'in yeteneklerini, fonksiyonelliklerini ve yapılarını içeren bir sistem kurmak demektir. Daha önce KD Sistemleri ve US için anlatılan kurma süreci, sahip olunan özellikler ve yetenekler gibi bilgilerin tümü, BTKD Sistemleri için de geçerlidir. Klein ve Methlie (1995), BTKD Sistemleri'nin yürütme süreci için Şekil 7.6'de gösterilen metodolojiyi önermişlerdir. Bu tasarım metodolojisinin amacı, kullanıcının bilgi sürecini desteklemek ve kullanıcının uygun tasarım stratejilerini belirlemesine yardımcı olmaktır. Metodolojide tasarımın temel karakteristikleri sunulmakta ve tasarımcıya çeşitli tasarım stratejilerini kullanma ve birleştirebilme imkanı sağlanmaktadır. Kullanıcı için başlangıç noktası modeller, veri tabanı, bilgi tabanı veya sonuçları bilinen bir uygulama olabilir. Klein ve Methlie (1995), sistemin tasarımına kullanıcı arayüzleri veya sistemin küresel mantığı ile ilişkilendirilmiş değişkenlerle başlanılmasını, ardından isteğe bağlı olarak diğer kaynakların sisteme birer fonksiyonellik biçiminde eklenmesini önermiştir. BTKD Sistemleri'nin yürütme süreci metodolojisinin adımları aşağıdaki başlıklarla ifade edilebilir (Klein ve Methlie, 1995).

Kullanıcının Hedeflerini Kavrama: Hedeflerin kavranması ile, bireyler arasında iyi bir iletişim kurularak problemin kısa sürede çözümlenmesi veya bir çözüme ulaşılması amaçlanmaktadır. Kullanıcının hedeflerini kavramayla aşağıda sıralanan ifadelerin gerçekleştirilmesi kolaylaşır.

- Problem durumunu belirleme.
- Problemi teşhis etme.
- Alternatifleri oluşturma.

- Kriter hesaplama.
- Alternatifleri değerlendirme.
- Bir alternatif seçme.

Problemin Sınırlarını Anlama ve Kavrama: Bu adım, alternatifleri oluşturmada yol gösterici bir özelliğe sahiptir. Problemin sınırlarını anlama ve kavramayla aşağıda sıralanan ifadelerle bir açıklık getirilmiştir olacaktır.

- Karar alıcılara.
- Karar alıcıların organizasyon içindeki karar yapılarıyla olan ilişkilerine.
- Karar alıcı tarafından kabul gören ve karşı çıkılan sabit ve kontrollü sınırlara.
- Bir alternatifin seçilmesiyle çözülmüş olacak probleme.
- Diğer karar alıcı veya uzmanların birlikte çalışma ve çözümleme için girdiler sağlama isteklilik ve yeteneklerine.
- Organizasyon içinde egemen olan kültüre.

Gerçek Karar Süreçlerini Anlama ve Kavrama: Kullanıcıların, birbirlerine benzer karar süreçlerini kullanarak çözüme ulaşmaları genellikle zayıf bir olasılık olarak görülmektedir. Aşağıda karar sürecini açıklayan alt adımlar sıralanmıştır.

- Karar sürecinin meydana geldiği genel görevleri tanımlamak.
- Karar sürecinin gerektirdiği kişileri ve bu kişilerin gerçekleştirdikleri alt görevleri tanımlamak.
 - Problem üzerinde çalışılması için gerekli olan alan bilgisi.
 - Problemin tanı metodolojisi ve görev bilgisi.
 - Alternatif oluşturma.
 - Kriter oluşturmada kullanılan gerçekler ve belgeler.

- Kriterin hesaplanma yöntemi.
- Kriterin sunumu.
- Hesaplamaya alınacak kısıtlar.

Problem İçin Normatif Bir Karar Süreci Tanımlamak: Tasarımcının görevi karar süreçlerini çözümlmek, bir tanı koymak ve süreci geliştirmektir.

Karar Sürecindeki Değişiklikleri Tanımlamak: Geliştirilecek olan BTKD Sistemleri birçok kişinin kullanımına açık olacaktır. KD Sistemleri'nin benimsenmesi sosyal bir süreç olduğundan kullanıcılarının tümü karar sürecinin gelişimini özümsemelidir.

Karar Sürecinin Desteklenecek Bölümünün Seçimi: Kullanıcının başlangıç ortamının tanımlanmış olması gerekmektedir.

BTKD Sistemleri'nin Fonksiyonel Çözümlemesi: Bu aşamada, BTKD Sistemleri'nin temel fonksiyonları ve bütün mimari veya kavramsal modelleri tanımlanır. Bunun gerçekleştirilebilmesi için genellikle problem çözüleme süreci boyunca kullanılmış olan raporların, karar modellerinin, veri yapılarının girdi formlarının ve bilgi tabanlarının ilk listelerine gereksinim duyulur.

Bir Geliştirme Ortamı Seçme: KD Sistemleri, US veya BTKD Sistemleri'nin tasarımını gerçekleştirecek beş farklı yol vardır (Klein ve Methlie, 1995).

- VB, DELPHI, PASCAL, BASIC, C veya Java gibi standart üçüncü ve dördüncü kuşak programlama dillerini kullanmak.
- LISP, PROLOG VEYA Smalltalk gibi YZ dillerini kullanmak.
- KD Sistemleri'nin fonksiyonelliğini karşılayan uzman sistem kabukları.
- Uzmana gereksinim duyulmayan KD Sistem geliştirme ortamları.
- KD Sistemleri ve US'i birleştiren BTKD Sistemleri geliştirme ortamları.

KD Sistemleri geliştirmek üzere yukarıda sıralanan tasarım geliştirme ortamlarından birinin seçimi, sistemlerden istenen fonksiyonelliğin türüne bağlıdır. Geliştirilecek

olan sistemden çeşitli işlevleri gerçekleştirmeleri istenebilir. İşlevselliğe bağlı sistem geliştirme alanı seçmede gereksinilen unsurları Klein ve Methlie (1995) aşağıdaki şekilde sıralamıştır.

Grafiksel Kullanıcı Arayüzü Ortamı: Grafiksel nesne tabanlı ortam, kullanıcıya sistemi ikonlar, menüler ve diğer grafiksel bileşenler yardımıyla kolaylıkla kullanabilme olanağı vermektedir. Bu ortam uygulamayı kullanan kişi ile uygulamanın kaynakları arasında küresel bir mantık etkileşimine izin verir.

Rapor Üretici: Kullanıcının, etkileşimli biçimde çeşitli nesnelere birleşik olan raporları tanımlamasına izin verir.

Modelleme Dili: Karar alıcıya yardım edecek bir modelleme şekli sunar.

Şekil Tanımlaması: Çeşitli denetimler sonucunda girilen veriden sistemin zeka bölümü tarafından bir açıklama raporu elde edilmelidir.

Veri tabanı Yönetim Sistemi: Gerekliği zaman veri tabanı ile bağlantı kurulabilmelidir.

Bilgi Tabanı Yönetim Sistemi: Bu sistemin esas amacı, bilginin sunuş yöntemi ile sunulan bilgiyi eşleştirmektir.

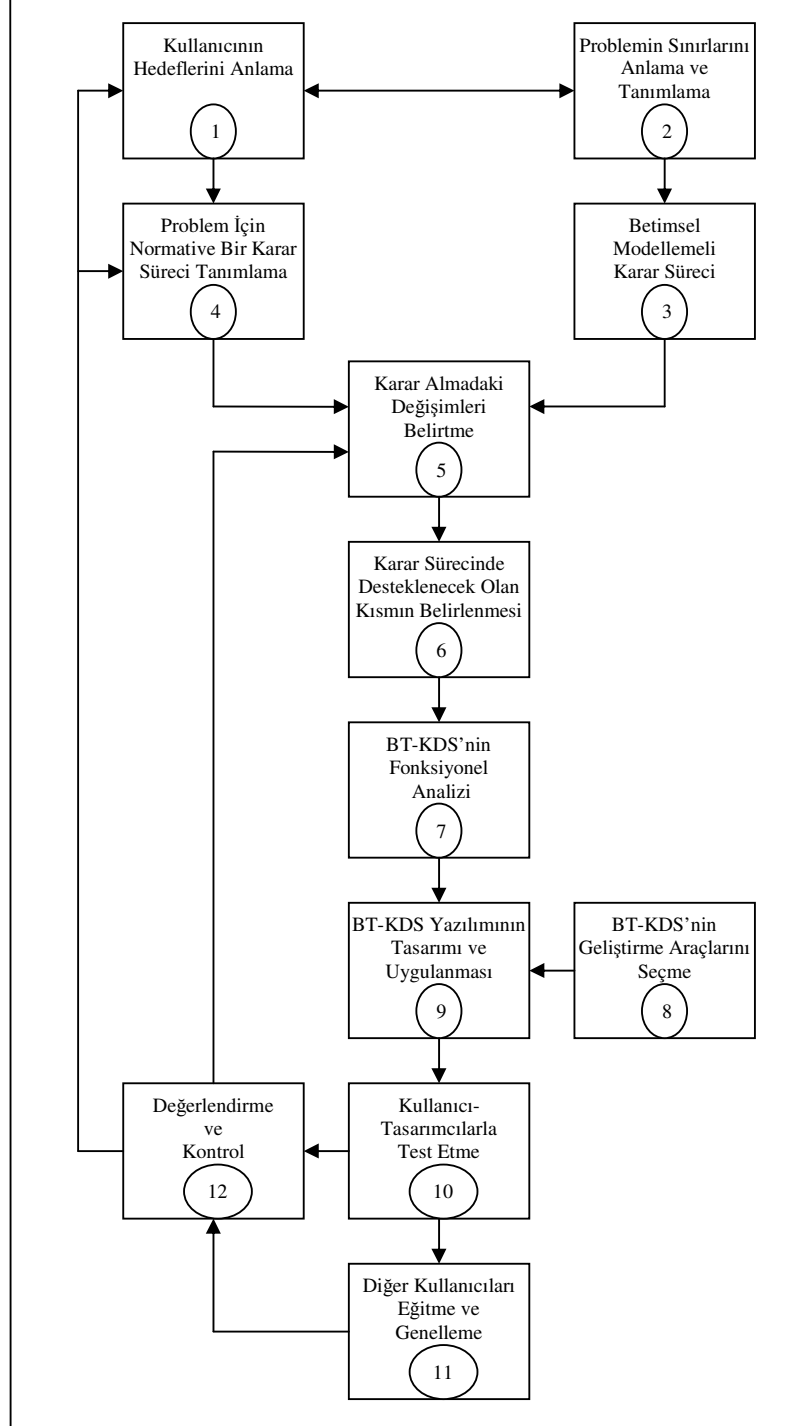
Araç Kutusu: Finans, istatistik ve tahmin gibi planlama uygulamaları için gerekli olan çeşitli algoritmaları içerir.

İletişim Arayüzü, Lokal Alan Ağı (Local Area Network, LAN) ve Alıcı/Dağıtıcı Mimarisi: İstenen iletişimi sağlayacak donanımı eşleştirir.

BTKD Sistemleri'nin Tasarımı ve Yürütülmesi: BTKD Sistemleri'nin tasarımı ve yürütülmesi aşağıda sıralanan adımları içermektedir (Klein ve Methlie, 1995).

- Veri çözümlemesi ve modelleme.
- Şekil tanımlaması ve girdi sağlama.
- Karar modelinin tasarımı ve sınanması.
- Rapor tanımlama.

- Bilgi tabanının modellenmesi ve testi.
- Bütün kullanıcı arayüzleri tasarımları ve küresel uygulama üzerinden mantık tanımlaması.



Şekil 7.6. BT-KD Sistemleri'nin yürütme süreci için metodoloji (Klein ve Methlie, 1995).

Sprague ve Carlson (1982)'a göre, iteratif tasarım süreci, kararlar ve karar alıcıların ihtiyaç duydukları esneklik ve kısa geliştirme zamanı özelliklerini sağladığından sistem geliştirmede kullanılabilir en uygun yöntemlerden biri olarak düşünülebilir. İteratif tasarım sürecinin her bir adımının prototiplendirilmesiyle süreç daha açıklayıcı hale getirilebilir.

BTKD Sistemi kurulduktan sonra oluşturulan sistemin doğru çalışıp çalışmadığının test edilmesi gerekmektedir. Bunun için sistemin bütün bölümleri test edilmelidir. Sistemin içerdiği karar modellerinin ve bilgi tabanlarının bitmiş halini test etmek için tasarıma dahil edilen kullanıcılara gereksinim duyulur. Sistemin testi ve değerlendirilmesi aşamaları boyunca kullanıcının işlemleri elle yapması, kullanıcıya iyi bir pratik kazandıracaktır. Kullanıcı eğitimi, çoğunlukla kullanıcıların tasarımdan çıkardıkları anlama, uzmanlık düzeylerine ve BTKD Sistemi'nin amacına bağlı olarak farklılık göstermektedir.

7.5. KD SİSTEMLERİ, US VE BTKD SİSTEMLERİ'NİN GELECEĞİ

KD Sistemleri, uzman BS, GKD Sistemleri ve US gibi BS'nin tümünün birden YDS olarak adlandırılan ve yeni teknolojik gelişmeleri kullanan bütünleşik bir sistemde toplanması, yöneticilerin bu sistemlere yaptıkları yatırımlardan maksimum fayda edinmelerini sağlayabilir.

Sistemleri içinde akıllı aracı bileşeninin bulunması, sistemin fonksiyonelliğini büyük ölçüde artıracaktır (Turban ve diğ., 2001). Akıllı aracı, bağımsız veya özerk bir şekilde kullanıcı veya bir başka program adına işlemleri gerçekleştirme potansiyeline sahip yeni bir teknolojidir. Turban ve diğ. (2001)'ne göre aracı; öneride bulunabilme, uyarabilme, bildirebilme, karıştırabilme, eleştirebilme, dağıtabilme, yardım sağlayabilme, yetki verebilme, açıklayabilme, süzebilme, yol gösterebilme, teşhis edebilme, eşleştirebilme, izleyebilme, yönlendirebilme, görüşebilme, düzenleyebilme, sunabilme, sorgulayabilme, rapor edebilme, hatırlatabilme, geri alabilme, planlayabilme, araştırabilme, koruyabilme, talep edebilme, sınıflandırabilme, depolayabilme, önerebilme, özetleyebilme, öğretebilme, dönüştürebilme, izleyebilme işlemlerini gerçekleştirebilir.

BÖLÜM 8

BAYKUŞ PROGRAMI

8.1. GİRİŞ

Bu bölümde daha önceki bölümlerde sözü edilen KD Sistemleri'ne örnek oluşturması amacıyla geliştirilmiş olan “Baykuş” adındaki program anlatılacaktır.

8.2. BAYKUŞ PROGRAMININ AMACI

Baykuş programının amacı, Mimar Sinan Güzel Sanatlar Üniversitesi (MSGSÜ) Fen Edebiyat Fakültesi İstatistik Bölümü lisans, bilimsel hazırlık, yüksek lisans ve doktora programlarına kayıtlı öğrenciler ile öğretim elemanlarına bölümle ilgili sorumlu oldukları hususlarda yardım etmek, bilgilenmek istedikleri konularda ise yetkileri dahilinde bilgi vermektir.

Baykuş ile bölümde okuyan lisans, bilimsel hazırlık, yüksek lisans ve doktora programlarına kayıtlı öğrenciler, ders seçimlerini gerçekleştirmek ve kişisel bilgilerinin doğru olup olmadığını kontrol edip yapılması gereken değişiklikleri danışmanlarına veya bölüm başkanına bildirmek ile sorumludurlar. Öğrencilerin bilgilendirildiği husus ise sınav sonuçlarıdır.

Programda öğretim elemanlarının sorumlu olduğu hususlar ise kendilerine ait bilgilerin doğruluğunu sınamak, danışmanı oldukları öğrencilerin durumlarını izlemek ve onlara ders seçimlerine yardımcı olup seçimlerini onaylamak, yürüttükleri derslerin sınav sonuçlarını girmek şeklinde sıralanabilir. Öğretim elemanları Baykuş programında bölümde okuyan tüm öğrencilerin kişisel bilgileri konusunda bilgilendirilmektedir.

8.3. BAYKUŞ PROGRAMININ MİMARİSİ

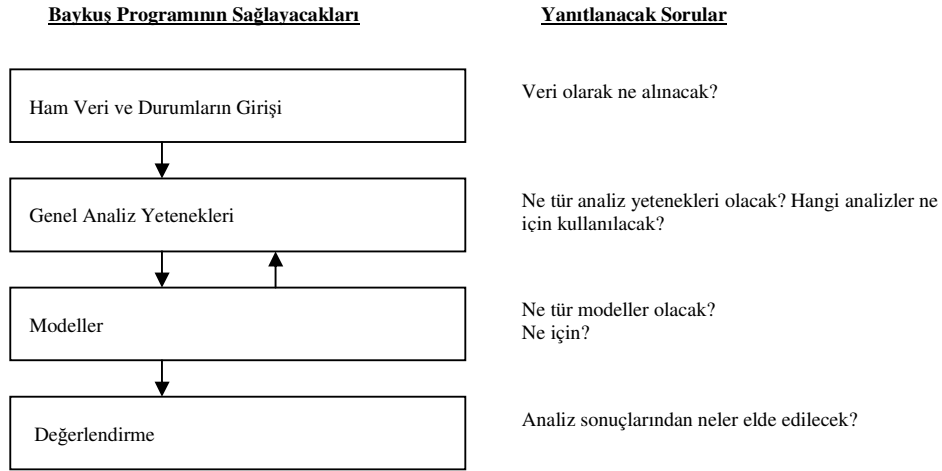
“Baykuş Programı”, Access 2003 veri tabanına dayanan bir KD Sistemi'dir. Bu programda kullanıcı ile iletişimi kolaylaştırmak amacıyla oluşturulan ekran

görüntüleri için Visual Basic (VB) programlama dili ve SQL (Structured Query Language) sorgu dilinden yararlanılmıştır.

8.3.1. BAYKUŞ PROGRAMININ SAĞLAYACAĞI DESTEK

Baykuş programının sağlayacağı destek Çizelge 8.1.'de ile özetlenmiştir.

Çizelge 8.1. Baykuş programının sağlayacağı destek.



8.3.2. BAYKUŞ PROGRAMINI GELİŞTİRME STRATEJİSİ

Baykuş programı, Kesim 8.2'de sıralanan amaçları gerçekleştirmek üzere Access 2003 paket programı, buna bağlı olarak VB programlama dili ve SQL sorgularından yararlanılarak oluşturulmuştur. Oluşturulmuş biçimine göre Baykuş programı, dördüncü kuşak KD Sistemleri'nden olup veri tabanına dayalı olma özelliğini taşımaktadır.

8.3.3. BAYKUŞ PROGRAMINI GELİŞTİRME SÜRECİ

Baykuş adındaki KD Sistemi'nin kuruluş adımları aşağıdaki gibidir.

A Aşaması (Planlama): Programın planlanmasında gerçekleştirilmek istenenler belirlenmiştir.

- İstatistik Bölümü'nde kayıtlı olan lisans, bilimsel hazırlık, yüksek lisans ve doktora öğrencilerinin ders seçimlerini gerçekleştirmelerini ve sınav sonuçlarını takip etmelerini sağlayacak, kişisel bilgilerini güncelleştirebilecek,

- İstatistik Bölümü öğretim elemanlarına, danışmanı oldukları öğrencilerin ders seçimlerini onaylamak ve durumlarını takip etmek, diğer öğrenciler hakkında bilgi edinmek ile verdikleri dersler ile ilgili durumları göstermek, bu derslere ilişkin not girişlerini gerçekleştirmek, kişisel bilgilerini güncelleştirmek,

- İstatistik bölüm başkanına bölümdeki tüm öğrencilerin ders seçimlerini, kişisel bilgilerini izlemek ve bölümü yeni kazanan öğrencilerin girişlerini sağlamak, bölümde okutulan dersleri düzenlemek ve yenilerini eklemek, bölümde görevli öğretim elemanlarının danışmanlık ve ders durumlarını düzenlemek, yeni öğretim elemanı girişini sağlamak,

işlevlerini gerçekleştirecek bir sistem oluşturmak.

B Aşaması (Araştırma): Planlama aşamasında sıralanan amaçları gerçekleştirmek üzere sistemde kullanılması gereken çözümlene ve modellerin belirlenmesi, bu çözümleneleri yapabilecek bilgisayar yazılım paketlerinin araştırılması aşamasıdır.

Baykuş programının belirlenen amaçları gerçekleştirebilmesi için VB kodlarından yararlanılması yeterli görülmüştür.

C Aşaması (Çözümleme ve Kavramsal Tasarım): Baykuş programını oluşturmak üzere yararlanılacak programların elde edilmesi aşamasıdır.

D Aşaması (Tasarım): Baykuş programı Access veri tabanına dayalı olarak geliştirileceğinden, tasarım aşamasında programın sağladığı form ve raporlardan yararlanılacaktır.

E Aşaması (İnşa): Yazılımın aktif hale getirilmesi ve ayarların yapılmasıdır.

F Aşaması (Yürütme): Veri girişleri yapılarak veri tabanının oluşturulması, problemi çözüme ulaştıracak çözümlenelerin yapılması ve çözümleneler sonucunda ulaşılan bulguların değerlendirilmesinden oluşmaktadır.

G Aşaması (Bakım ve Belgeleme): Baykuş programının kullanılmasıyla elde edilen raporlardan oluşmaktadır.

H Aşaması (Uyarılama): Kullanıcı ihtiyaçlarına bağlı olarak yukarıdaki aşamalara geri dönüşün mümkün olabilmesidir.

8.3.4. BAYKUŞ PROGRAMININ İÇERDİĞİ ELEMANLAR

Baykuş programı, veri tabanına dayalı bir KD Sistemi olup sistemin veri yönetimi alt sistemini oluşturan veri tabanları Access paket programı aracılığıyla oluşturulmuştur. Oluşturulan bu veri tabanlarına bilgilerin girişi ise programda “Bölüm Başkanı Girişi” düğmesiyle ulaşılan formlar aracılığıyla yapılmaktadır. Programın model yönetimi alt sistemi, VB programlama dilinde bulunan matematiksel yapılar aracılığıyla oluşturulmuştur. Programın iletişim alt sistemi uygulamanın kullanıcı ile iletişimini sağlayacak formlar aracılığıyla sağlanmıştır. Sistemin bilgi yönetimi alt sistemi ise oluşturulan rapor ve çıktı ekranları ile oluşturulmuştur.

Baykuş programının çalıştığı klasör içinde aşağıda sıralanan elemanlar yer almaktadır.

1. MSGSÜ adındaki uygulama programı,
2. MSGSÜData adında veri tabanı programı,
3. Resimler klasörü,
4. BackupData klasörü.

MSGSÜ Uygulama Programı: MSGSÜ adındaki uygulama programı Baykuş programının dayandığı esas veri tabanı olup uygulama boyunca kullanıcıların gördükleri bütün formları, bu formlara ait sorguları, raporları ve VB kodlarını içerir.

MSGSÜData Veri Tabanı: MSGSÜ uygulamasının kullandığı tüm tabloların yer aldığı veri tabanı “MSGSÜData” veri tabanıdır. MSGSÜ veri tabanındaki tablolar MSGSÜData veri tabanından dış veri alımıyla oluşturulmuştur. Dış veri alımıyla, programda yapılan veri güncellemelerinin kullanıcıları etkilemeden gerçekleştirilmesinin sağlanması amaçlanmıştır.

MSGSÜData veri tabanında yer alan tabloların listesi EK 1’de verilmiştir. Tablolar arasında kurulan ilişkiler Ek 2’de gösterildiği gibidir. EK 1’de sıralanan tablolardan 4, 5, 6, 9, 10, 20, 21 ve 28 nolu tablolar diğer tablolara bilgi girişlerinde veya istenen bilgilere erişimlerde kullanılan yardımcı tablolardır.

Resimler Klasörü: MSGSÜ uygulamasında kullanılan ve İstatistik Bölümü lisans, bilimsel hazırlık, yüksek lisans, doktora öğrencileri ile öğretim elemanlarının resimlerini içeren klasördür.

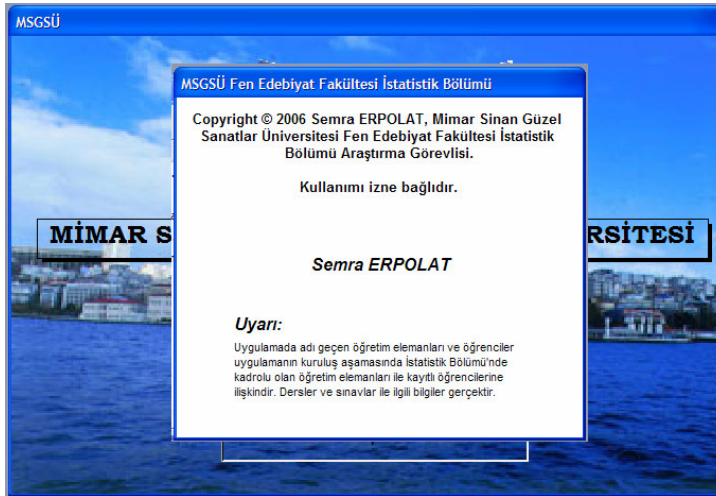
BackupData Klasörü: Bu klasör uygulamanın kullanımı süresince yapılan yedekleme işlemleri sonucunda elde edilen veritabanlarının saklandığı klasördür.

8.4. BAYKUŞ PROGRAMININ İŞLEYİŞİ

Baykuş programı çalıştırıldığında Ekran 8.1.'deki tanıtım ekranı ile karşılaşılır. Bu ekranın görünmesinin ardından birkaç dakika ekranda kalacak olan Ekran 8.2'deki telif hakkı ekranı açılır. Bu ekranların ardından kullanıcıyı yönlendirecek olan Ekran 8.3'deki ana menü ile karşılaşılır.



Ekran 8.1. Baykuş programının tanıtım ekranı.



Ekran 8.2. Telif hakkı ekranı.



Ekran 8.3. Baykuş programının ana menüsü.

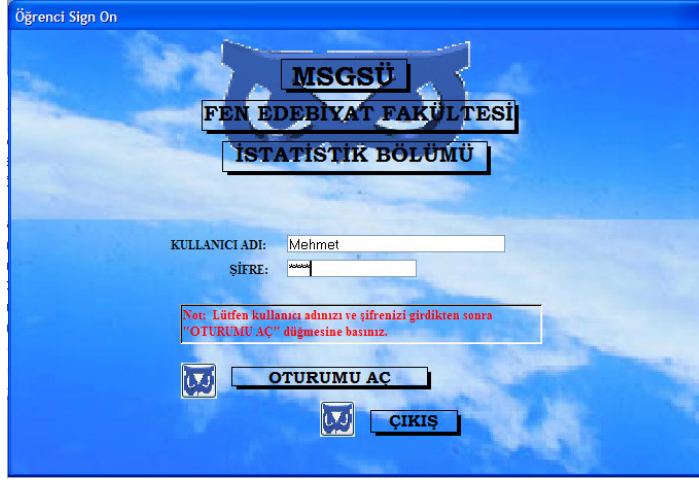
Ana menü; “Lisans Öğrenci Girişi”, “Lisans Üstü Öğrenci Girişi”, “Öğretim Elemanı Girişi” ve “Bölüm Başkanı Girişi” olmak üzere dört tip kullanıcı türüne hitap eden düğmelerden oluşmaktadır.

Baykuş programının bu dört farklı kullanıcı türüne göre işleyişi aşağıda açıklanmıştır.

8.4.1. LİSANS ÖĞRENCİ GİRİŞİ

“Lisans Öğrenci Girişi”, İstatistik lisans programı öğrencilerinin Baykuş programını kullanmalarına olanak vererek bilgilendirilmelerini sağlayacak giriş düğmesidir. Bu düğmeye basıldığında kullanıcı adı ve şifreyi sorgulayan “Öğrenci Sign On” adındaki ekran ile karşılaşılır (Ekran 8.4). Öğrenci kullanıcı, adını ve şifresini girdikten sonra “Oturumu Aç” düğmesine tıklar. Kullanıcı adı ve şifre, öğrencinin programa ilk kez kaydı yapıldığı sırada Bölüm Başkanı kullanıcısı tarafından belirlenmektedir.

Aşağıdaki ekran görüntüsü İstatistik Bölümü’nde okumakta olup Baykuş programına girişi yapılmış olan bir öğrenciye ait kullanıcı adı ve şifre değerleri girildikten sonra alınmıştır.



Ekran 8.4. Öğrenci Sign On ekranı.

Öğrenci Sign On ekranı, doğru kullanıcı adı ve şifre girilmediği sürece Baykuş programının çalıştırılmasına izin vermeyen bir şifre ekranıdır. Bu ekran kullanıcıya doğru kullanıcı adı ve şifreyi girmesi için üç hak tanımaktadır. Üç hak sonunda eğer doğru kullanıcı adı ve şifre girilmemişse program ana menüye geri dönmektedir. Doğru kullanıcı adı ve şifre girildiği durumda ise Ekran 8.5’de gösterilen “Lisans Öğrenci Main” ekranı devreye girer. İşlemin gerçekleşmesini sağlayan VB kodları aşağıdaki gibidir.

```

Private Sub cmdSignOn_Click()
    Dim cnThisConnect As ADODB.Connection
    Dim rcdKullanicilarOgrenciler As New ADODB.Recordset
    Dim i As Integer
    Static intPswdCount As Integer
    Set cnThisConnect = CurrentProject.Connection
    rcdKullanicilarOgrenciler.Open "tblKullanicilarOgrenciler", cnThisConnect, _
        adOpenKeyset, adLockOptimistic, adCmdTable
    If Me.txtKullanicıAdı = "" Or Me.txtŞifre = "" Then
        MsgBox "Lütfen Kullanıcı adınızı ve şifrenizi girdikten sonra Oturumu Aç düğmesine basınız.",
            vbInformation, gstrAppTitle
        Me.txtKullanicıAdı = ""
        Me.txtŞifre = ""
        Me.txtKullanicıAdı.SetFocus
        Exit Sub
    End If
    For i = 1 To rcdKullanicilarOgrenciler.RecordCount
        If (rcdKullanicilarOgrenciler![KullanicıAdı] = Me.txtKullanicıAdı) And
            (rcdKullanicilarOgrenciler!
                [Şifre] = Me.txtŞifre) Then
            gstrThisUser = Me!txtKullanicıAdı
            DoCmd.OpenForm "frmÖğrenciMain"
            [Forms]!frmÖğrenciMain.RecordSource = "qryÖğrenciMain"
            DoCmd.Close acForm, Me.Name
            Exit Sub
        Else
            rcdKullanicilarOgrenciler.MoveNext
        End If
    Next i
End Sub

```

```

End If
Next i
MsgBox "Yanlış Kullanıcı Adı veya Şifre girdiniz. Lütfen tekrar deneyiniz.", vbCritical,
gstrAppTitle
Me.txtKullanıcıAdı = ""
Me.txtŞifre = ""
Me.txtKullanıcıAdı.SetFocus
intPswdCount = intPswdCount + 1
If intPswdCount < 3 Then Exit Sub
Me.Undo
DoCmd.Close acForm, Me.Name
DoCmd.OpenForm "frmMain"
End Sub

```

Bu kodlarda kullanıcının girmiş olduğu kullanıcı adı ve şifre, sistemde yer alan ve lisans öğrencilerinin kullanıcı adı, şifre gibi bilgilerinin saklandığı "tblKullanıcılarÖğrenciler" tablosundaki verilerle karşılaştırılır. Eğer öğrencinin girdiği değerler bu tablodaki değerlerle uyuyorsa Baykuş programı çalışır.



Ekran 8.5. "Lisans Öğrenci Main" ekranı.

Lisans Öğrenci Main ekranı "Kişisel Bilgiler" ve "Bölüm Bilgileri" olmak üzere iki düğmeyi içerir.

8.4.1.1. Kişisel Bilgiler

Kişisel Bilgiler düğmesi kullanıcıyı Ekran 8.6'da gösterilen ekrana ulaştırmaktadır.

Kişisel Bilgiler Öğrenci

Sayın, Mehmet Can ERPOLAT

KİŞİSEL BİLGİLER ÖZ GEÇMİŞ AKADEMİK BİLGİ SINIF DERSLER KULLANICI GİRİŞİ

Öğrenci No: 200614 Cinsiyet: Erkek Fotoğrafi: [Portrait Photo]

Adi: Mehmet İkinci Adı: Can Soyadı: ERPOLAT

KAYIT OLDUĞU

Tarih: 09/18/2006 Eğitim Yılı: 2006-2007 Donem: GÜZ

ADRES BİLGİLERİ

Adres: Bolu

Posta Kodu: 14000

Semt: Merkez İl: Bolu

Ev Tel: Cep Tel: Email:

NÜFUS BİLGİLERİ

T.C. Kimlik No: 0

Vergi No: 0

Doğum Yeri İl: Bolu İlçe: Merkez

Doğum Tarihi: 6/14/2006

Kan Grubu: 0 Rh+ Medeni Hali: Bekar

GERİ ÇIKIŞ

Ekran 8.6. “Kişisel Bilgiler Öğrenci” ekranı.

Kişisel Bilgiler Öğrenci ekranı; “Kişisel Bilgiler”, “Özgeçmiş”, “Akademik Bilgiler”, “Sınıf”, “Alacağı Dersler” ve “Kullanıcı Girişi” adlarındaki altı sekmeden oluşmaktadır. Bu ekran, üzerinde değişiklik yapılmasına izin vermeyen tamamen bilgi vermek amacıyla kullanılan bir ekrandır. Kullanıcısı, İstatistik lisans programı öğrencisi olan bu ekran üzerinde herhangi bir değişiklik veya bilgi güncellemesi, danışman veya bölüm başkanının yetkisi dahilinde mümkün olabilir.

Kişisel Bilgiler Sekmesi: Bu sekme kullanıcının adres ve nüfus bilgilerini kapsayan kişisel bilgilerinden oluşmaktadır. Sekmenin görüntüsü Ekran 8.6’daki “Kişisel Bilgiler Öğrenci” ekranı ile aynıdır.

Özgeçmiş Sekmesi: Öğrencinin özgeçmişini içeren metin niteliğinde bir sekmedir.

Kişisel Bilgiler Öğrenci

Sayın, Mehmet Can ERPOLAT

KİŞİSEL BİLGİLER ÖZ GEÇMİŞ AKADEMİK BİLGİ SINIF ALAĞI DERSLER KULLANICI GİRİŞİ

Mehmet

GERİ ÇIKIŞ

Ekran 8.7. “Özgeçmiş” sekmesi.

Akademik Bilgiler Sekmesi: Bu sekme öğrencinin mezun olduğu lise adı ve türü, mezuniyet tarihi ve akademik ortalaması ile ilgili bilgi vermektedir.

Kişisel Bilgiler Öğrenci

Sayın, Mehmet Can ERPOLAT

KİŞİSEL BİLGİLER ÖZ GEÇMİŞ AKADEMİK BİLGİ SINIF DERSLER KULLANICI GİRİŞİ

MEZUN OLDUĞU LİSE:

Adı: Bolu Anadolu Lisesi

Türü: Anadolu Lisesi

Mezuniyet Tarihi: 9/15/2006

Akademik Ortalama: 5

GERİ ÇIKIŞ

Ekran 8.8. “Akademik Bilgiler” sekmesi.

Sınıf Sekmesi: Sınıf sekmesi eğitim yılını, dönemi, öğrencinin kaçınıcı yarıyılı olduğunu ve danışmanını gösteren sekmedir.

Kişisel Bilgiler Öğrenci

Sayın, Mehmet Can ERPOLAT

KİŞİSEL BİLGİLER ÖZ GEÇMİŞ AKADEMİK BİLGİ SINIF ALACAĞI DERSLER KULLANICI GİRİŞİ

EĞİTİM YILI, DÖNEM, YARIYIL DANIŞMAN BİLGİLERİ

Eğitim Yılı: 2006-2007

Dönem: GÜZ

Yarıyıl: 1

Danışman: Prof. Dr. Nalan CİNEMRE

GERİ ÇIKIŞ

Ekran 8.9. “Sınıf” sekmesi.

Dersler Sekmesi: Öğrencinin İstatistik Bölümü’ne ilk kaydından itibaren aldığı ve alabileceği dersleri gösteren sekmedir.

Kişisel Bilgiler Öğrenci

Sayın, Mehmet Can ERPOLAT

KİŞİSEL BİLGİLER ÖZ GEÇMİŞ AKADEMİK BİLGİ SINIF DERSLER KULLANICI GİRİŞİ

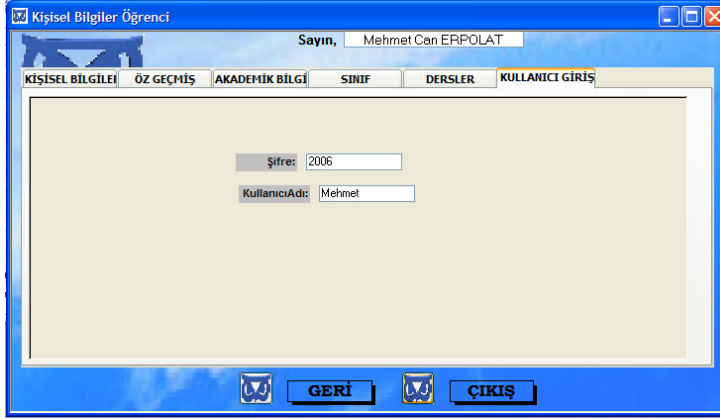
KODU:	ADı:	KREDİSİ:	Z/S:	ÖĞRETİM ÜYESİNİN ADI:	YARIYILI:
IST101	İSTATİSTİĞE GİRİŞ I	3	Z	Yrd. Doç. Dr. Meral YAY	1
IST103	OLASILIK I	2	Z	Prof. Dr. Gülay KIROĞLU	1
IST105	BİLGİ İŞLEM I	2	Z	Prof. Dr. Aydın ERAR	1
IST109	İKTİSAT I	2	Z	Yrd. Doç. Dr. Yeşim REEL	1
MAT181	ANALİZ I	4	Z	Yrd. Doç. Dr. NIKOĞLU	1
SOS123	SOSYOLOJİ I	2	Z	Yrd. Doç. Dr. Fidevs GÜMÜŞOĞLU	1
ING111	TEMEL İNGİLİZCE I	0	OZ	Ökt. Ayman DURANSOY	1
TUR102	TÜRK DİLİ I	0	OZ	Ökt. Dr. Nüket ÖZER	1
IST381	ÇOK AMAÇLI PROGRA	2	S	Prof. Dr. Nalan CİNEMRE	5
IST483	KARAR DESTEK SİSTE	2	S	Gülser DONDURMACI	7

Record: 14 of 10

GERİ ÇIKIŞ

Ekran 8.10. “Dersler” sekmesi.

Kullanıcı Girişi Sekmesi: Bu sekme öğrenciye Baykuş programını çalıştırabilmesi için “Bölüm Başkanı” girişi kullanıcısı tarafından belirlenen kullanıcı adı ve şifreden oluşan hatırlatıcı bir sekmedir.



Ekran 8.11. “Kullanıcı Girişi” sekmesi.

8.4.1.2. Bölüm Bilgileri

Bölüm Bilgileri düğmesinin tıklanmasıyla ulaşılan ekran, Ekran 8.12’de gösterilen “Dönem Seçimi” ekranıdır. Aşağıdaki ekran görüntüsü, eğitim yılı olarak “2006-2007”, dönem olarak “GÜZ”, yarıyıl olarak ise “1” seçildikten sonra alınmıştır.



Ekran 8.12. “Dönem Seçimi” ekranı.

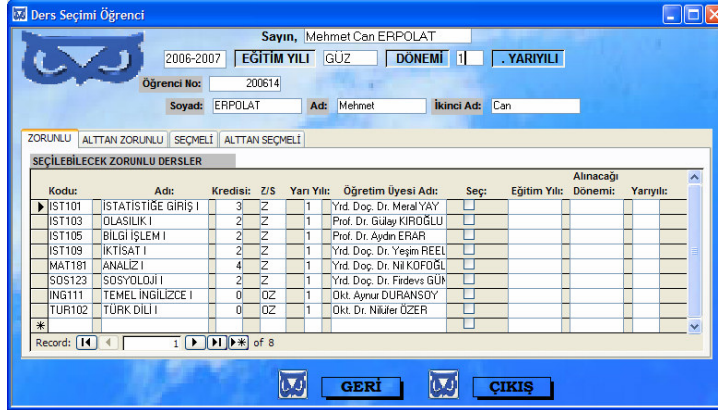
Bu ekran öğrenciyi, seçtiği “eğitim yılı”, “dönem” ve “yarıyıl” belirleyici kriterlerine göre bölüm bilgilerine ulaştıran ekrandır. “Dönem Seçimi” ekranında kriterler belirlenip “Göster” düğmesine tıklanınca Ekran 8.13’deki “Bölüm Bilgileri” ekranına ulaşılır.



Ekran 8.13. “Bölüm Bilgileri” ekranı.

Bu ekran lisans öğrencisine, o dönemde seçebileceği dersleri gösteren “Ders Seçimi”, seçtiği dersleri gösteren “Seçilen Dersler”, seçtiği derslere ilişkin not bilgilerini gösteren “Sınav Bilgileri” ve daha önceki dönemlerde seçtiği dersler ve bu derslerdeki başarı durumu hakkında bilgi veren “Alınan Dersler” ekranlarına ulaştıracak olan düğmelerden oluşmaktadır.

Ders Seçimi Ekranı: “Ders Seçimi” ekranı, öğrencinin mevcut eğitim yılı ve dönemde seçebileceği dersleri gösteren ve istediği dersleri seçmesine izin veren ekrandır. Ekranın görüntüsü Ekran 8.14’deki gibidir.



Ekran 8.14. “Ders Seçimi” ekranı.

Ders Seçimi ekranı “Zorunlu”, “Altta Zorunlu”, “Seçmeli” ve “Altta Seçmeli” adlarındaki dört sekmeden oluşmaktadır.

Zorunlu Sekmesi: Zorunlu sekmesi, öğrencinin içinde bulunduğu dönemde seçebileceği zorunlu dersleri gösteren sekmedir. Bu sekmedeki formun kayıt kaynağının (record source) dayandığı sorgu, Baykuş programında “qryDersSeçimiÖğrenci” olarak adlandırılmış olup bu sorgunun SQL ifadesi aşağıdaki gibidir.

```
SELECT tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID,
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID,
tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu,
tblÖğrencilerinTümAlmasıGerekenDersleri.[Seç/Bırak], tblDersler.DersID, tblDersler.DersKodu,
tblDersler.DersAdı, tblDersler.MSGSÜKredisi, tblDersler.[Zorunlu/Seçmeli],
tblDersler.ÖğrÜyesiAdı, tblDersler.[Eğitim-Yılı], tblDersler.Dönemi, tblDersler.YarıYılı,
tblDersler.AçıkKapalı, tblÖğrencilerinTümAlmasıGerekenDersleri.EğitimYılı,
tblÖğrencilerinTümAlmasıGerekenDersleri.Dönem,
tblÖğrencilerinTümAlmasıGerekenDersleri.Yarıyıl,
tblÖğrencilerinTümAlmasıGerekenDersleri.AlttanTekrar, tblDersler.ÖğretimElemanıID
FROM tblDersler INNER JOIN tblÖğrencilerinTümAlmasıGerekenDersleri ON tblDersler.DersID =
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID
WHERE
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu)="Y") AND
((tblDersler.[Zorunlu/Seçmeli])="Z" Or (tblDersler.[Zorunlu/Seçmeli])="OZ") AND
((tblDersler.[Eğitim-Yılı])=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblDersler.Dönemi)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem]) AND
((tblDersler.AçıkKapalı)=True));
```

Sorgu “tblDersler” ile “tblÖğrencilerinTümAlmasıGerekenDersleri” tablolarının “DersID” alanına göre ilişkilendirilip aşağıda sıralanan kriterlerin seçimiyle oluşturulmuştur.

- 1.“tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki her bir öğrenci için benzersiz bir alan olan “ÖğrenciID” ile kullanıcının ÖğrenciID bilgisinin uyuşması.
2. “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanının “Y” olması.
3. “tblDersler” tablosundaki “Zorunlu/Seçmeli” alanının “Z” veya “OZ” olması.
4. “tblDersler” tablosundaki “Açık/Kapalı” alanının “True” olması.
5. “tblDersler” tablosundaki “Eğitim-Yılı” alanı ile “Dönemi” alanının öğrencinin dönem seçimi ekranıyla seçtiği eğitim yılı ve dönem bilgileriyle aynı olması.

Öğrenci bu sekmede sıralanan zorunlu derslerden seçmek istediklerini “Seç” onay kutusuna tıklayıp çek atmak suretiyle gerçekleştirir. Bu işlem sonucunda dersin

alınacağı eğitim yılı, dönem ve yarıyıl bilgileri ilgili alanlara işlenir. Seç işlemi sonucunda devreye giren VB kodları aşağıdaki gibidir.

```
Private Sub Seç_Bırak_Click()  
If Seç_Bırak = True Then  
    Me.BaşarıDurumu = "S"  
    Me.EğitimYılı.SetFocus  
    Me.EğitimYılı = gstrEğitimYılı  
    Me.Dönem.SetFocus  
    Me.Dönem = gstrDönem  
    Me.Yarıyıl.SetFocus  
    Me.Yarıyıl = gstrYarıyıl  
End If  
If Seç_Bırak = False Then  
    Me.BaşarıDurumu = "Y"  
    Me.EğitimYılı.SetFocus  
    Me.EğitimYılı = ""  
    Me.Dönem.SetFocus  
    Me.Dönem = ""  
    Me.Yarıyıl.SetFocus  
    Me.Yarıyıl = ""  
End If  
End Sub
```

Öğrenci almaktan vazgeçtiği dersleri “Seç” onay kutusunu tekrar tıklayıp boşaltmak suretiyle gerçekleştirir.

Burada dikkat edilmesi gereken önemli nokta, ders seçimlerinin gerçekleştirilip ekrandan çıkıldıktan sonra öğrenci için geri dönüşün söz konusu olmamasıdır. Seçimin yanlış olması veya seçimden vazgeçilmesi durumunda düzeltme için öğrencinin danışmanına veya bölüm başkanına başvurması gerekmektedir. Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adlı öğrencinin “İstatistiğe Giriş I”, “Olasılık I”, “Bilgi İşlem I”, “İktisat I” zorunlu derslerini seçtikten sonra alınmıştır.

Kodu	Adı	Kredisiz	Z/S	Yarı Yılı	Öğretim Üyesi Adı	Seç	Eğitim Yılı	Dönemi	Yarıyılı
İST1101	İSTATİSTİĞE GİRİŞ I	3	Z	1	Yrd. Doç. Dr. Meral YAY	<input checked="" type="checkbox"/>	2006-2007	GÜZ	1
İST1103	OLASILIK I	2	Z	1	Prof. Dr. Gülay KIROĞLU	<input checked="" type="checkbox"/>	2006-2007	GÜZ	1
İST1105	BİLGİ İŞLEM I	2	Z	1	Prof. Dr. Aydın ERAR	<input checked="" type="checkbox"/>	2006-2007	GÜZ	1
İKT1109	İKTİSAT I	2	Z	1	Yrd. Doç. Dr. Yeşim REEL	<input checked="" type="checkbox"/>	2006-2007	GÜZ	1
MAT1181	ANALİZ I	4	Z	1	Yrd. Doç. Dr. Nil KÖFÜLÜ	<input type="checkbox"/>			
SOS123	SOSYOLOJİ I	2	Z	1	Yrd. Doç. Dr. Feride GÜN	<input type="checkbox"/>			
İNG111	TEMEL İNGİLİZCE I	0	OZ	1	Ok. Ayhan DURANSOY	<input type="checkbox"/>			
TUR102	TÜRK DİLİ I	0	OZ	1	Ok. Dr. Nüket ÖZER	<input type="checkbox"/>			

Ekran 8.15. “Zorunlu” sekmesi.

Alttan Zorunlu Sekmesi: Görüntüsü ve kullanımı, zorunlu sekmesi gibi olan bu sekmede, öğrencinin alıp da başaramadığı zorunlu derslerin listesi yer almaktadır. Öğrenci “Zorunlu” sekmesinde olduğu gibi tekrar almak istediği dersleri “Seç” sütununa çek atarak belirler.

“Mehmet Can ERPOLAT” adlı öğrencinin alttan alması gereken zorunlu dersi olmadığı için öğrenci bu sekmede herhangi bir seçim yapmamıştır.

Sekmenin kayıt kaynağını oluşturan SQL ifadesi, “Zorunlu” sekmesindeki gibi olup tek farkı “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanının “F” olmasıdır.

Alttan Zorunlu sekmesinde “Seç” düğmesine tıklandığında işleyen VB kodları da “Zorunlu” sekmesindeki benzer olup farklılıkları “Seç” düğmesi seçili olan derse ait “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanının “SF” olması, seçili olmaması durumunda ise “F” olmasıdır.

Seçmeli Sekmesi: Görünüşü ve kullanımı, diğer sekmeler gibi olan “Seçmeli” sekmesi, öğrencinin mevcut dönemde ilk kez alacağı seçmeli derslerin listesini içeren sekmedir.

Bu sekmenin kayıt kaynağını oluşturan SQL ifadesinde “Zorunlu” sekmesinde olduğu gibi “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanı “Y” dir.

“Seçmeli” sekmesinde “Seç” düğmesine tıklandığında işleyen VB kodu diğer sekmelerdekine benzemekle birlikte farklı olarak “Seç” düğmesi seçili olan derse ait “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanının “SÇ” olması, seçili olmaması durumunda ise “Y” olmasıdır.

“Mehmet Can ERPOLAT” adlı öğrenci “Seçmeli” sekmesinde listelenen derslerden “Çok Amaçlı Programlama” ve “Karar Destek Sistemleri” seçmeli derslerini seçmiştir.

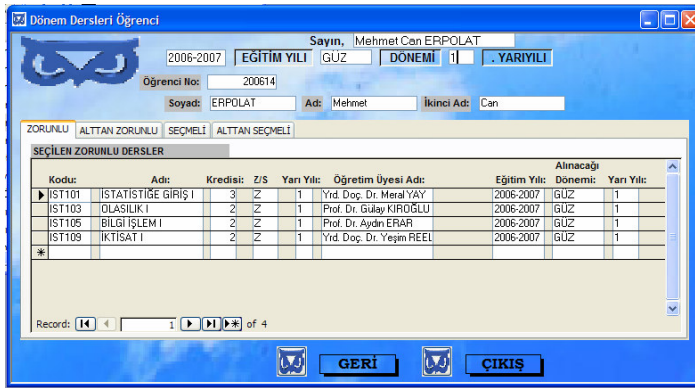
Alttan Seçmeli Sekmesi: Diğer sekmelerle benzer görünüşe ve kullanıma sahip bu sekme, öğrencinin daha önceki dönemlerde alıp da başaramadığı seçmeli derslerin listesini içeren sekmedir.

Bu sekmenin kayıt kaynağını oluşturan SQL ifadesinde “Alttan Zorunlu” sekmesinde olduğu gibi “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanı “F” olmasıdır.

“Alttan Seçmeli” sekmesinde “Seç” düğmesine tıkladığında işleyen VB kodunda “Seç” düğmesi seçili olan derse ait “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanı “SÇF”, seçili olmayanlara ait olan alanın değeri ise “F” dir.

“Mehmet Can ERPOLAT” adlı öğrencinin alttan seçmeli dersi olmadığı için bu sekmede herhangi bir seçim yapması gerekmemiştir.

Seçilen Dersler Ekranı: Öğrencinin seçtiği dersler hakkında bilgi veren ekrandır. Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adlı öğrencinin “Ders Seçimi” ekranında ders seçimlerini gerçekleştirip “Bölüm Bilgileri” ekranında yer alan “Seçilen Dersler” düğmesine tıklandıktan sonra alınmıştır.



Ekran 8.16. “Seçilen Dersler” ekranı.

Zorunlu Sekmesi: Zorunlu sekmesi, öğrencinin mevcut dönemde seçtiği zorunlu dersleri listeleleyen ekrandır. Bu nedenle sekmedeki formun kayıt kaynağını oluşturan SQL ifadesinde “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanı “S” olanlar sorgulanmaktadır. Sorgu aşağıda verildiği gibidir.

```
SELECT tblDersler.SicilNo, tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID,
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID,
tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu,
tblÖğrencilerinTümAlmasıGerekenDersleri.[Seç/Bırak],
tblÖğrencilerinTümAlmasıGerekenDersleri.Onay, tblDersler.DersID, tblDersler.DersKodu,
tblDersler.DersAdı, tblDersler.MSGSÜKredisi, tblDersler.[Zorunlu/Seçmeli],
tblDersler.ÖğrÜyesiAdı, tblDersler.[Eğitim-Yılı], tblDersler.Dönemi, tblDersler.YarıYılı,
tblDersler.AçıkKapalı, tblÖğrencilerinTümAlmasıGerekenDersleri.EğitimYılı,
```



```

tblÖğrencilerinTümAlmasıGerekenDersleri.Dönem,
tblÖğrencilerinTümAlmasıGerekenDersleri.Yarıyıl,
tblÖğrencilerinTümAlmasıGerekenDersleri.AlttanTekrar, tblDersler.ÖğretimElemanıID
FROM tblDersler INNER JOIN tblÖğrencilerinTümAlmasıGerekenDersleri ON tblDersler.DersID =
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID
WHERE
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu)="S") AND
((tblÖğrencilerinTümAlmasıGerekenDersleri.Onay)=False) AND
((tblDersler.[Zorunlu/Seçmeli]="Z" Or (tblDersler.[Zorunlu/Seçmeli]="OZ")) AND
((tblDersler.[Eğitim-Yılı]=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblDersler.Dönemi)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem]) AND
((tblDersler.AçıkKapalı)=True));

```

Ekran 8.16'daki "Seçilen Dersler" ekranı aynı zaman da öğrencinin seçmiş olduğu zorunlu dersleri de listeleyen ekrandır.

Altan Zorunlu Sekmesi: Öğrencinin alttan almak istediği zorunlu dersleri listeleyen sekmedir. Sekmedeki formun kayıt kaynağını oluşturan SQL ifadesi "Zorunlu" sekmesindeki benzer olup "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosundaki "BaşarıDurumu" alanı "SF"dir.

Seçmeli Sekmesi: Bu sekme öğrencinin mevcut dönemde seçtiği seçmeli dersleri listeler. Sekmedeki formun kayıt kaynağını oluşturan SQL ifadesinde "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosundaki "BaşarıDurumu" alanı "SÇ"dir.

Altan Seçmeli Sekmesi: Öğrencinin alttan almak isteyip seçtiği seçmeli dersleri listeleyen sekmedir. Sekmedeki formun kayıt kaynağını oluşturan SQL ifadesinde "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosundaki "BaşarıDurumu" alanı "SÇF"dir.

Alınan Dersler Ekranı: Öğrencinin mevcut eğitim yılı ve dönemde aldığı dersler ile önceki dönemlerde aldığı derslerin başarı durumunu gösteren ekrandır.

Aşağıdaki ekran görüntüsü öğrencinin ders seçimini tammlamasının, danışmanın da uygun gördüğü derslerin onaylanmasından sonra alınmıştır. Öğrencinin danışmanı tarafından onaylanan dersler "İstatistiğe Giriş I" ve "Olasılık I" zorunlu dersleri ile "Çok Amaçlı Programlama" ve "Karar Destek Sistemleri" seçmeli dersleri olduğundan ekranda sadece bu dersler listelenmektedir.

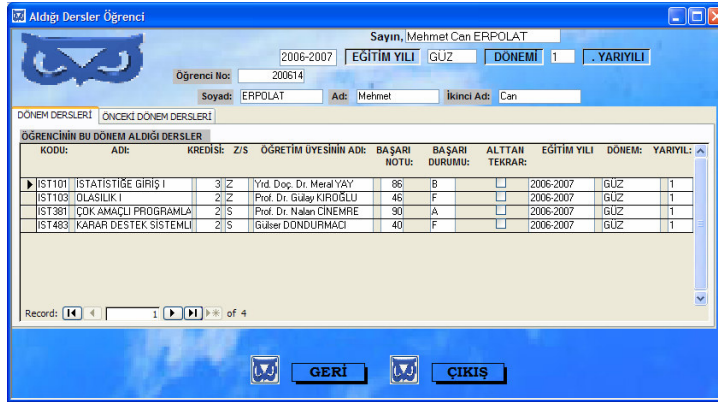


Ekran 8.17. “Alınan Dersler” ekranı.

Bu ekran “Dönem Dersleri” ve “Önceki Dönem Dersleri” sekmelerinden oluşmaktadır.

Dönem Dersleri Sekmesi: Öğrencinin seçip de danışmanı veya bölüm başkanı tarafından onaylanan derslerinin listelendiği sekmedir. Sekmede ayrıca dönem derslerinin başarı durumları da yer almaktadır (Ekran 8.17 ve Ekran 8.18).

Aşağıdaki ekran görüntüsü dönem sonunda notların girilmesi sonrasında karşılaşılan ekran görüntüsüdür.



Ekran 8.18. Not girişlerinin yapılması sonunda “Alınan Dersler” ekranı.

Ekranın kayıt kaynağının dayandığı SQL aşağıdaki gibidir.

```
SELECT tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID,
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID,
tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu,
tblÖğrencilerinTümAlmasıGerekenDersleri.[Seç/Bırak], tblDersler.DersID, tblDersler.DersKodu,
tblDersler.DersAdı, tblDersler.MSGSÜKredisi, tblDersler.[Zorunlu/Seçmeli],
tblDersler.ÖğrÜyesiAdı, tblDersler.[Eğitim-Yılı], tblDersler.Dönemi, tblDersler.YarıYılı,
tblDersler.AçıkKapalı, tblSınavlar.[Vize I], tblSınavlar.[Vize II], tblSınavlar.Final,
```

```

tblSnavlar.BaşarıNotu, tblSnavlar.BaşarıDurumu,
tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu,
tblÖğrencilerinTümAlmasıGerekenDersleri.Onay,
tblÖğrencilerinTümAlmasıGerekenDersleri.EğitimYılı,
tblÖğrencilerinTümAlmasıGerekenDersleri.Dönem,
tblÖğrencilerinTümAlmasıGerekenDersleri.Yarıyıl, tblSnavlar.EğitimYılı, tblSnavlar.Dönem,
tblSnavlar.Yarıyıl, tblSnavlar.AlttanTekrar, tblDersler.ÖğretimElemanıID
FROM tblDersler INNER JOIN (tblÖğrencilerinTümAlmasıGerekenDersleri INNER JOIN
tblSnavlar ON tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID = tblSnavlar.ÖğrenciID) ON
(tblDersler.DersID = tblSnavlar.DersID) AND (tblDersler.DersID =
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID)
WHERE
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblDersler.AçıkKapalı)=True) AND
((tblÖğrencilerinTümAlmasıGerekenDersleri.Onay)=True) AND
((tblSnavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblSnavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem]) AND
((tblSnavlar.Yarıyıl)=[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl])) OR
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu)="F") AND
((tblDersler.AçıkKapalı)=True) AND ((tblÖğrencilerinTümAlmasıGerekenDersleri.Onay)=False)
AND ((tblSnavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblSnavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem]) AND
((tblSnavlar.Yarıyıl)=[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl])) OR
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu)="SF") AND
((tblDersler.AçıkKapalı)=True) AND ((tblÖğrencilerinTümAlmasıGerekenDersleri.Onay)=False)
AND ((tblSnavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblSnavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem]) AND
((tblSnavlar.Yarıyıl)=[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl])) OR
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu)="SÇF") AND
((tblDersler.AçıkKapalı)=True) AND ((tblÖğrencilerinTümAlmasıGerekenDersleri.Onay)=False)
AND ((tblSnavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblSnavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem]) AND
((tblSnavlar.Yarıyıl)=[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl]));

```

Sorgu “tblDersler”, “tblÖğrencilerinTümAlmasıGerekenDersleri” ile “tblSnavlar” tablolarının ilişkilendirilmesiyle oluşturulmuştur. Sorgu aşağıda sıralanan kriterlerden oluşmaktadır.

1. “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki herbir öğrenci için benzersiz bir alan olan “ÖğrenciID” ile kullanıcının ÖğrenciID bilgisinin uyuşması.
2. “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanının “Y”, “F”, “SF” veya “SÇF” olması.
3. “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki “BaşarıDurumu” alanının “Y” olanlar için “Onay” alanının “False” olması buna karşılık “BaşarıDurumu” alanının “F”, “SF” veya “SÇF” olanların “Onay” alanının ise “True” olması.

4. "tblDersler" tablosundaki "Zorunlu/Seçmeli" alanının "Z" veya "OZ" olması.
5. "tblDersler" tablosundaki "Açık/Kapalı" alanının "True" olması.
6. "tblSnavlar" tablosundaki "Eğitim Yılı", "Dönem" ve "Yarıyıl" alanlarının öğrencinin dönem seçimi ekranıyla seçtiği eğitim yılı, dönem ve yarıyıl bilgileriyle aynı olması.

Önceki Dönem Dersleri Sekmesi: Öğrencinin önceki dönemlerde aldığı tüm dersleri ve bu derslerdeki sınav sonuçlarını içeren listeyi veren ekrandır. Görünüş bakımından Ekran 8.17'deki "Dönem Dersleri" sekmesiyle aynıdır. Öğrenci daha önce ders almadığı için bu sekmede herhangi bir ders bulunmamaktadır.

Ekranın kayıt kaynağının dayandığı SQL ise aşağıdaki gibidir.

```
SELECT tblDersler.SicilNo, tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID,
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID,
tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu,
tblÖğrencilerinTümAlmasıGerekenDersleri.[Seç/Bırak], tblDersler.DersID, tblDersler.DersKodu,
tblDersler.DersAdı, tblDersler.MSGSÜKredisi, tblDersler.[Zorunlu/Seçmeli],
tblDersler.ÖğrÜyesiAdı, tblDersler.[Eğitim-Yılı], tblDersler.Dönemi, tblDersler.YarıYılı,
tblDersler.AçıkKapalı, tblSnavlar.[Vize I], tblSnavlar.[Vize II], tblSnavlar.Ödev, tblSnavlar.Final,
tblSnavlar.BaşarıNotu, tblSnavlar.BaşarıDurumu, tblSnavlar.AlttanTekrar, tblSnavlar.EğitimYılı,
tblSnavlar.Dönem, tblSnavlar.Yarıyıl, tblÖğrencilerinTümAlmasıGerekenDersleri.Onay,
tblÖğrencilerinTümAlmasıGerekenDersleri.EğitimYılı,
tblÖğrencilerinTümAlmasıGerekenDersleri.Dönem,
tblÖğrencilerinTümAlmasıGerekenDersleri.Yarıyıl, tblDersler.ÖğretimElemanıID
FROM tblDersler INNER JOIN (tblÖğrencilerinTümAlmasıGerekenDersleri INNER JOIN
tblSnavlar ON tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID = tblSnavlar.ÖğrenciID) ON
(tblDersler.DersID = tblSnavlar.DersID) AND (tblDersler.DersID =
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID)
WHERE
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblDersler.AçıkKapalı)=True) AND ((tblSnavlar.BaşarıDurumu)="A") AND
((tblSnavlar.Yarıyıl)<[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl]) OR
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblDersler.AçıkKapalı)=True) AND ((tblSnavlar.BaşarıDurumu)="B") AND
((tblSnavlar.Yarıyıl)<[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl]) OR
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblDersler.AçıkKapalı)=True) AND ((tblSnavlar.BaşarıDurumu)="C") AND
((tblSnavlar.Yarıyıl)<[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl]) OR
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblDersler.AçıkKapalı)=True) AND ((tblSnavlar.BaşarıDurumu)="D") AND
((tblSnavlar.Yarıyıl)<[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl]) OR
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblDersler.AçıkKapalı)=True) AND ((tblSnavlar.BaşarıDurumu)="E") AND
((tblSnavlar.Yarıyıl)<[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl]) OR
(((tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![Öğr
enciID]) AND ((tblDersler.AçıkKapalı)=True) AND ((tblSnavlar.BaşarıDurumu)="F") AND
((tblSnavlar.Yarıyıl)<[Forms]![frmBölümBilgileriÖğrenci]![txtYarıyıl]));
```

Sorgu “tblDersler”, “tblÖğrencilerinTümAlmasıGerekenDersleri” ile “tblSınavlar” tablolarının ilişkilendirilmesiyle oluşturulmuştur. Sorgu aşağıda sıralanan kriterlerden oluşmaktadır.

1. “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki herbir öğrenci için benzersiz bir alan olan “ÖğrenciID” ile kullanıcının ÖğrenciID bilgisinin uyuşması.
2. “tblSınavlar” tablosundaki “BaşarıDurumu” alanının “A”, “B”, “C”, “D”, “E” veya “F” olması.
3. “tblSınavlar” tablosundaki “Yarıyıl” alanlarının öğrencinin dönem seçimi ekranıyla seçtiği yarıyıl bilgisinden küçük olması.

Sınav Bilgileri Ekranı: “Sınav Bilgileri” ekranı, öğrencinin mevcut eğitim yılı ve dönemde aldığı bütün derslere ilişkin not bilgilerinin gösterildiği bilgi verici ekrandır. Ekranın SQL kaynağı aşağıdaki gibidir.

```
SELECT tblSınavlar.SınavlarID, tblSınavlar.ÖğrenciID,
tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID,
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID,
tblÖğrencilerinTümAlmasıGerekenDersleri.BaşarıDurumu,
tblÖğrencilerinTümAlmasıGerekenDersleri.[Seç/Bırak], tblDersler.DersID, tblDersler.DersKodu,
tblDersler.DersAdı, tblDersler.MSGSÜKredisi, tblDersler.[Zorunlu/Seçmeli],
tblDersler.ÖğrÜyesiAdı, tblDersler.[Eğitim-Yılı], tblDersler.Dönemi, tblDersler.YarıYılı,
tblDersler.AçıkKapalı, tblSınavlar.[Vize I], tblSınavlar.Ödev, tblSınavlar.[Vize II], tblSınavlar.Final,
tblSınavlar.BaşarıNotu, tblSınavlar.BaşarıDurumu, tblÖğrencilerinTümAlmasıGerekenDersleri.Onay,
tblSınavlar.EğitimYılı, tblSınavlar.Dönem, tblSınavlar.Yarıyıl, tblSınavlar.[Vize I], tblSınavlar.Final,
tblÖğrencilerinTümAlmasıGerekenDersleri.EğitimYılı,
tblÖğrencilerinTümAlmasıGerekenDersleri.Dönem,
tblÖğrencilerinTümAlmasıGerekenDersleri.Yarıyıl, tblSınavlar.AlttanTekrar,
tblDersler.ÖğretimElemanıID
FROM tblDersler INNER JOIN (tblÖğrencilerinTümAlmasıGerekenDersleri INNER JOIN
tblSınavlar ON tblÖğrencilerinTümAlmasıGerekenDersleri.ÖğrenciID = tblSınavlar.ÖğrenciID) ON
(tblDersler.DersID = tblSınavlar.DersID) AND (tblDersler.DersID =
tblÖğrencilerinTümAlmasıGerekenDersleri.DersID)
WHERE (((tblSınavlar.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![ÖğrenciID]) AND
((tblSınavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblSınavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem])) OR
(((tblSınavlar.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![ÖğrenciID]) AND
((tblSınavlar.BaşarıDurumu)="A") AND
((tblSınavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblSınavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem])) OR
(((tblSınavlar.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![ÖğrenciID]) AND
((tblSınavlar.BaşarıDurumu)="B") AND
((tblSınavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblSınavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem])) OR
(((tblSınavlar.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![ÖğrenciID]) AND
((tblSınavlar.BaşarıDurumu)="C") AND
((tblSınavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
((tblSınavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem])) OR
```

```

(((tblSnavlar.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![ÖğrenciID]) AND
(tblSnavlar.BaşarıDurumu)="D") AND
(tblSnavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
(tblSnavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem])) OR
(((tblSnavlar.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![ÖğrenciID]) AND
(tblSnavlar.BaşarıDurumu)="E") AND
(tblSnavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
(tblSnavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem])) OR
(((tblSnavlar.ÖğrenciID)=[Forms]![frmBölümBilgileriÖğrenci]![ÖğrenciID]) AND
(tblSnavlar.BaşarıDurumu)="F") AND
(tblSnavlar.EğitimYılı)=[Forms]![frmBölümBilgileriÖğrenci]![txtEğitimYılı]) AND
(tblSnavlar.Dönem)=[Forms]![frmBölümBilgileriÖğrenci]![txtDönem]]);

```

Sorgu “tblDersler”, “tblÖğrencilerinTümAlmasıGerekenDersleri” ile “tblSnavlar” tablolarının ilişkilendirilmesiyle oluşturulmuştur. Sorgu aşağıda sıralanan kriterlerden oluşmaktadır.

1. “tblÖğrencilerinTümAlmasıGerekenDersleri” tablosundaki herbir öğrenci için benzersiz bir alan olan “ÖğrenciID” ile kullanıcının ÖğrenciID bilgisinin uyuşması.
2. “tblSnavlar” tablosundaki “BaşarıDurumu” alanının “A”, “B”, “C”, “D”, “E”, “F” veya “ ” olması.
3. “tblSnavlar” tablosundaki “EğitimYılı” ve “Dönem” alanlarının öğrencinin dönem seçimi ekranıyla seçtiği eğitim yılı ve dönem bilgileriyle aynı olması.

Aşağıdaki ekran görüntüsü dönem sonunda not bilgilerinin girilmesinden sonra alınmıştır.

KODU:	ADI:	ÖĞRETİM ÜYESİ ADI:	ALTTAN TEKRAR:	Vize I:	Vize II:	FİNAL:	BAŞARI NOTU:	BAŞARI DURUM
IST101	İSTATİSTİĞE GİRİŞ I	Yrd. Doç. Dr. Meral YAY	<input type="checkbox"/>	85	80	90	86	B
IST101	OLASILIK I	Prof. Dr. Gülay KIROĞLU	<input type="checkbox"/>	95	30	30	46	F
IST381	ÇOK AMAÇLI PROGRAMLA	Prof. Dr. Nalan CINEMRE	<input type="checkbox"/>	100	80	90	90	A
IST481	KARAR DESTEK SİSTEMLİ	Gülser DONDURMACI	<input type="checkbox"/>	50	30	40	40	F

Ekran 8.19. “Sınav Bilgileri” ekranı.

8.4.2. LİSANS ÜSTÜ ÖĞRENCİ GİRİŞİ

“Lisans Üstü Öğrenci Girişi”, İstatistik Bölümü’nde okuyan bilimsel hazırlık, yüksek lisans ve doktora öğrencilerin bilgilendirilmelerini sağlayacak giriş düğmesidir.

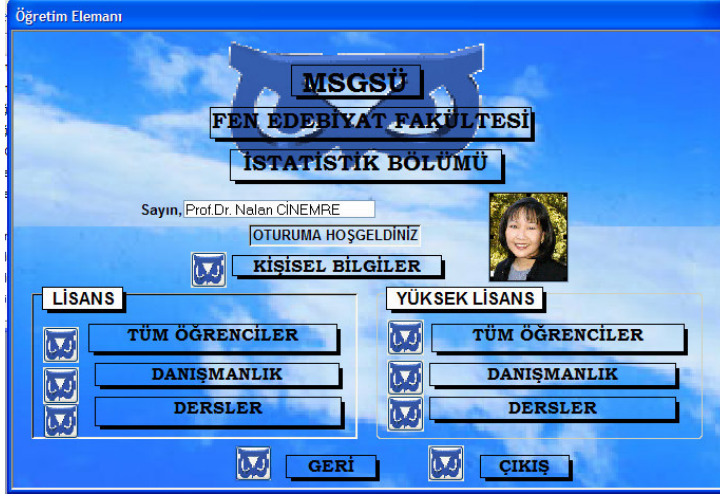
Bu giriş Kesim 8.4.1’de sözü edilen “Lisans Öğrenci Girişi” ile benzer ekranlara ve düğmelere sahiptir. Bu nedenle “Lisans Üstü Öğrenci Girişi” kısmına değinilmeyecektir. Bu giriş, Baykuş programında lisans üstü öğrencilere ait bilgilerin yer aldığı tablolar ile lisans üstü derslerinin yer aldığı tablolar üzerinden işlem görmektedir.

8.4.3. ÖĞRETİM ELEMANI GİRİŞİ

Öğretim Elemanı Girişi, İstatistik Bölümü’ndeki öğretim elemanlarının kendi kişisel bilgilerine, verdikleri lisans ve lisans üstü derslerine, danışmanı oldukları lisans ve lisans üstü öğrencilerin kişisel bilgileri ile bölüm bilgilerine, tüm lisans ve lisans üstü öğrencilere ilişkin kişisel bilgilere erişimini sağlayan ekranlara ulaştıran düğmedir.

“Öğretim Elemanı Girişi” düğmesine tıklandığında, diğer düğmelerde olduğu gibi, bir kullanıcı adı ve şifre doğrulama ekranı ile karşılaşılır. “Öğretim Elemanı Sign On” adı verilen bu ekranın görünüşü ve dayandığı VB kodları diğer “Sign On” ekranların VB kodlarının benzeri olup tek farkı girilen kullanıcı adı ve şifrenin “tblÖğretimelemanlarıKullanıcıları” tablosunda aranıyor olmasıdır. “Sign On” penceresine doğru kullanıcı adı ve şifre girildiği takdirde “Öğretim Elemanı Main” ekranına ulaşılır.

Aşağıdaki ekran görüntüsü, “Öğretim Elemanı Sign On” ekranında İstatistik Bölümü’nde öğretim elemanı olan “Prof. Dr. Nalan CİNEMRE”ye ait kullanıcı adı ve şifre bilgileri girilip “Oturumu Aç” düğmesine tıklandıktan sonra alınmıştır.



Ekran 8.20. “Öğretim Elemanı Main” ekranı.

“Öğretim Elemanı Main” ekranı “Lisans” ve “Lisans Üstü” olmak üzere iki ayrımı ve her bir ayrımında da “Tüm Öğrenciler”, “Danışmanlık” ve “Dersler” olmak üzere toplam altı düğmeyi içermektedir. Ekranda ayrıca öğretim elemanının kişisel bilgilerini gösteren “Kişisel Bilgiler” düğmesi de yer almaktadır.

8.4.3.1. Kişisel Bilgiler Düğmesi

Bu düğme öğretim elemanının kendisine ait bilgilere ulaşmasını sağlayan “Kişisel Bilgiler Öğretim Elemanı” ekranını açan düğmedir. Ekranın görünüşü diğer “Kişisel Bilgiler” ekranları gibidir.

Ekranın “Kişisel Bilgiler”, “Özgeçmiş”, “Akademik Bilgiler”, “Yayımlar”, “Lisans Dersleri”, “Lisans Üstü Dersleri” sekmeleri, Kesim 8.4.4.3.2.’de değinilen “Öğretim Elemanlarının Kişisel Bilgileri” ekranındaki karşılık gelen sekmeleriyle aynı iken “Kullanıcı Girişi” sekmesi farklılık göstermektedir.

Ekranın değiştirilmesine müsaade edilen alanlarında herhangi bir değişiklik yapılması durumunda “Kaydet” düğmesine basılması gerekmektedir. Aksi takdirde yapılan değişiklikler geçici olacaktır.

Kişisel Bilgiler Sekmesi: “Kişisel Bilgiler Sekmesi”, öğretim elemanının kişisel bilgilerini gösteren sekme olup öğretim elemanı bu sekme üzerinde değişiklik yapabilme yetkisine sahiptir.

Özgeçmiş Sekmesi: Kullanıcısının özgeçmişini gösteren sekmedir. Üzerinde değişiklik yapılabilir.

Akademik Bilgiler Sekmesi: Öğretim elemanın akademik durumu hakkında bilgi veren sekmedir. Bu sekmede öğretim elemanın mezun olduğu lise, üniversite, varsa yüksek lisans, doktora, doçentlik derecelerini aldığı üniversiteler ve bu derecelere ilişkin akademik ortalamalar ile mezuniyet tarihleri yer almaktadır. Öğretim elemanı tarafından değiştirilebilir bir sekmedir.

Yayınlar Sekmesi: Yayınlar sekmesi, öğretim elemanın yapmış olduğu akademik çalışmaları yayın adı, tarihi, yayınlandığı yer ve ortak yazar bilgilerini verecek şekilde sıralayan sekmedir. Değişiklik yapılabilir.

Lisans Dersleri Sekmesi: Öğretim elemanının bölümde verdiği lisans derslerini listeleyen sekmedir. Değişiklik yapılamaz.

Lisans Üstü Dersleri Sekmesi: Öğretim elemanının bölümde verdiği lisans üstü derslerini listeleyen sekmedir. Değişiklik yapılamaz.

Kullanıcı Girişi Sekmesi: Öğretim elemanının Baykuş programını çalıştırabilmesi için ihtiyaç duyduğu kullanıcı adı ve şifresini gösteren sekmedir. Bu sekme bilgi vermek amaçlı olup kullanıcısı tarafından değiştirilemez. Değişimi ancak bölüm başkanı tarafından gerçekleştirilebilir. Ekran görüntüsü Kesim 8.4.1.1.'de sözü edilen lisans öğrencilerine ilişkin “Kullanıcı Girişi” sekmesiyle aynıdır.

8.4.3.2. Lisans Öğrencileri Ayırımı

Bu ayırım “Tüm Öğrenciler”, “Danışmanlık” ve “Dersler” olmak üzere toplam üç düğmeyi kapsamaktadır.

Tüm Öğrenciler Düğmesi: Bu düğme öğretim elemanının, İstatistik bölümünde okumakta olan tüm öğrencilere ilişkin kişisel bilgilere erişimini sağlayan ekrana ulaşmasını sağlar.

Ekranın kullanımı ve bilgileri gösterme mantığı Kesim 8.4.1.1.'de sözü edilen öğrencilere ilişkin “Kişisel Bilgiler” ekranında olduğu gibidir. Ekran üzerinde öğretim elemanının herhangi bir değişiklik yapmasına izin verilmemiştir.

Danışmanlık Düğmesi: Bu düğme öğretim elemanını, danışmanı olduğu öğrencilerin kişisel bilgileri ile bölüm bilgilerini gösteren ekranlara ulaştıracak olan başlangıç düğmesidir. Düğme tıklandığında öğretim elemanına danışmanlığının hangi eğitim yılı, dönem ve yarıyıl için olduğunu soran bir dönem seçimi ekranı açılır.

8.4.1.2. kesiminde değinilen “Dönem Seçimi” ekranının benzeri olan bu ekranda “Göster” düğmesine basıldığında “Lisans Öğrencileri Listesi Danışmanlık” ekranı açılır.

Aşağıdaki ekran görüntüsü “Prof. Dr. Nalan CİNEMRE”nin “Dönem Seçimi” ekranında eğitim yılı olarak “2006-2007”, dönem olarak “Güz”, yarıyıl olarak “1” seçip “Göster” düğmesine tıklamasından sonra alınmıştır.

Öğrenci No	Soyad	Ad
200701	Erpolat	Semra
101	q	q
2008	EE	BB
200814	ERPOLAT	Mehmet
	ERPOLAT	Mehmet

Ekran 8.21 . “Lisans Öğrencileri Listesi Danışmanlık” ekranı.

“Lisans Öğrencileri Listesi Danışmanlık” ekranı, öğretim elemanının danışmanlığını yaptığı öğrencilerin listesini içermektedir. Öğretim elemanının listelenen öğrencilerden herhangi birine ait kişisel bilgilerine veya bölüm bilgilerine ulaşabilmesi için listeden bir seçip yapıp ilgili düğmeye basması gerekmektedir.

Bu ekranda ayrıca istenen öğrenciyi bulmaya yarayan “Ara” düğmesi ve bölümde okuyan öğrenci sayılarına ilişkin bilgiler veren “İstatistik” düğmeleri de yer almaktadır.

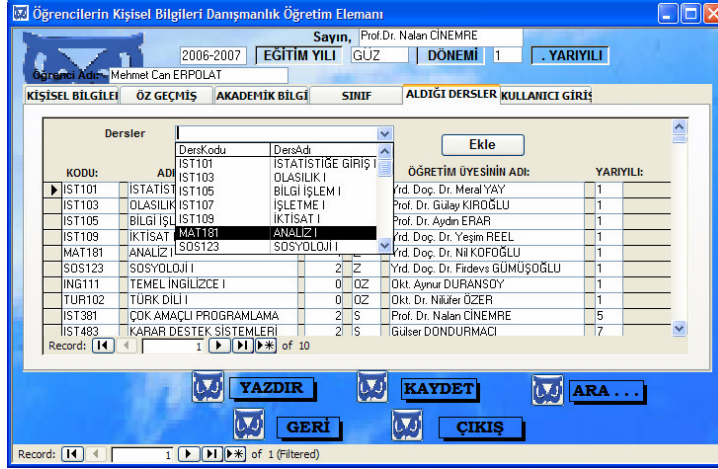
Kişisel Bilgiler Düğmesi: Bu düğme öğretim elemanının, danışmanlığını yaptığı öğrencilere ilişkin kişisel bilgilere erişimi sağlayan “Öğrencilerin Kişisel Bilgileri Danışmanlık” olarak adlandırılan ekranı açar.

Öğretim elemanı “Öğrencilerin Kişisel Bilgileri Danışmanlık” ekranında yer alan öğrenci listesinden “Mehmet Can ERPOLAT” adlı öğrenciyi seçip “Kişisel Bilgiler” düğmesine bastığında daha önce Kesim 8.4.1.1.’de sözü edilen “Kişisel Bilgiler” ekranına benzer bir ekran ile karşılaşır.

Görünüş bakımından daha önce sözü edilen öğrencilere ilişkin “Kişisel Bilgiler” ekranıyla aynı olan bu ekran üzerinde öğretim elemanı değişiklik yapabilme yetkisine sahiptir. “Kişisel Bilgiler”, “Özgeçmiş”, “Akademik Bilgiler” ve “Sınıf” sekmeleri üzerinde herhangi bir değişiklik yapmak isteyen öğretim elemanının değişiklik yapmak istediği alan üzerine gelip gerekli değişikliği yapıp form üzerinde yer alan “Kaydet” düğmesine basması yeterlidir.

Form üzerinde yer alan “Dersler” sekmesi daha önce sözü edilen “Kişisel Bilgiler” ekranındaki “Dersler” sekmesinden farklıdır. Daha önce de söz edildiği gibi, “Dersler” sekmesi, öğrencinin aldığı veya ilerki dönemlerde alabileceği dersleri listeleyen ekrandır. İlk kaydı yapılırken seçebileceği dersleri belirlenen öğrencinin bu derslerine yenilerini ekleyebilmek için bu sekmeden yararlanır. Bunun için sekme üzerinde yer alan açılabilir liste içinde sıralanan derslerden birinin seçilip “Ekle” düğmesine basılması yeterlidir. Şayet açılabilir listede sıralanan derslerden aşağıdaki listede yer alan derslerden biri seçilip “Ekle” düğmesine basılırsa Baykuş programı kaydın zaten mevcut olduğunu belirten bir uyarı mesajı verecektir.

Aşağıdaki ekran görüntüsü, “Mehmet Can ERPOLAT” adlı öğrencinin “Kişisel Bilgiler” ekranındaki “Dersler” sekmesindeki açılabilir listeden “Analiz I” dersi seçili iken alınmıştır.



Ekran 8.22. “Kişisel Bilgiler” ekranında yeni ders ekleme işlemi.

“Ekle” düğmesinin VB kodu aşağıdaki gibidir.

```

Private Sub cmdEkle_Click()
Dim cnThisConnect As ADODB.Connection
Dim rcdTümDersler As New ADODB.Recordset
Dim i As Integer
Dim a As Integer
Dim b As Integer
a = Me.cmbDersEkle.Column(0)
b = Me.ÖğrenciID
Set cnThisConnect = CurrentProject.Connection
rcdTümDersler.Open "tblÖğrencilerinTümAlmasıGerekenDersleri", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
For i = 1 To rcdTümDersler.RecordCount
    If (rcdTümDersler![DersID] = a) And (rcdTümDersler![ÖğrenciID] = b) Then
        MsgBox "Zaten Seçili olan bir dersi tekrar seçiyorsunuz. " & _
            "Lütfen aşağıdaki listede mevcut olmayan bir dersi seçiniz.", vbCritical, gstrAppTitle
    Exit Sub
    End If
rcdTümDersler.MoveNext
Next i
rcdTümDersler.AddNew
rcdTümDersler![DersID] = Me.cmbDersEkle.Column(0)
rcdTümDersler![ÖğrenciID] = Me.ÖğrenciID
rcdTümDersler.Update
Me.RecordSource = "qryAlmasıGerekenDerslerÖğretimElemanı"
On Error Resume Next
Screen.PreviousControl.SetFocus
If Err <> 0 Then
    Me.SetFocus
    Err.Clear
End If
If Not SaveIt() Then Exit Sub
End Sub

```

Bu kodlar aracılığıyla "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosunda öğrenciye ait “ÖğrenciID” ve “DersID” alanlarına göre arama yapılmaktadır. Açılan

listeden eklenmesi istenen ders seçilip “Ekle” düğmesine basıldığında, yapılan bu arama sonucunda eğer seçilen ders tabloda mevcutsa ekleme işlemi gerçekleşmeyip kullanıcı, “Zaten seçili olan bir dersi tekrar seçiyorsunuz. Lütfen aşağıdaki listede mevcut olmayan bir dersi seçiniz.” Mesajıyla uyarılır ve herhangi bir işlem yapılmaz. Aksi durumda seçilen ders "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosuna eklenir.

“Kullanıcı Girişi” sekmesi, öğrencinin ilk kaydı sırasında kendisine verilen kullanıcı adı ve şifre bilgilerinin yenileriyle değiştirilmesine imkan tanır. Bu sekmenin kullanışı Kesim 8.4.3.1’de sözü edilecek olan öğretim elemanlarına ait “Kişisel Bilgiler” ekranındaki “Kullanıcı Adı ve Şifre” sekmesiyle aynı olduğundan burada sözü edilmeyecektir. Tek farklılık yapılan değişikliklerin lisans öğrencilerini ilgilendiren bilgilerin saklanması olduğu tablolara yansıtan VB kodlarının işletiliyor olmasıdır.

Bölüm Bilgileri Düğmesi: Bu düğme kullanıcıyı öğrencilerin bölüm bilgilerine ulaştıracak olan ana menüye götürür. Aşağıdaki ekran görüntüsü, öğretim elemanının “Lisans Öğrencileri Listesi Danışmanlık” ekranında “Mehmet Can ERPOLAT” adlı öğrenciyi seçip “Bölüm Bilgileri” düğmesine tıklamasından sonra alınmıştır.



Ekran 8.23. “Bölüm Bilgileri” ekranı.

“Bölüm Bilgileri” ekranındaki fotoğraf danışmana öğrenciyi hatırlatılır. Bu ekranda “Mevcut Dönem” ayırımında “Ders Seçimi”, “Seçilen Dersler” düğmeleri ile “Ders

Bilgileri” ayırımında ise “Alınan Dersler” ve “SınavBilgileri” düğmeleri yer almaktadır.

Ders Seçimi Düğmesi: Bu düğme öğrencinin seçebileceği zorunlu, alttan zorunlu, seçmeli ve alttan seçmeli dersleri listeleyen sekmelerden oluşmaktadır. Kullanım mantığı daha önce Kesim 8.4.1.2’de sözü edilen öğrencilerin “Ders Seçimi” ekranıyla aynıdır. Bu ekranın amacı, öğrencinin seçmeyi de danışmanı tarafından alması uygun veya zorunlu görülen derslerin öğrenciye seçimini gerçekleştirmektir.

Seçilen Dersler Düğmesi: Tıklandığında “Ders Seçimi” ekranını açan düğmedir. Bu ekran öğrenci tarafından seçilen veya danışmanı tarafından kendisi için seçilen derslerin görüntülediği ekran olup amacı öğrenci tarafından seçilen veya seçtirilen zorunlu, alttan zorunlu, seçmeli ve alttan seçmeli derslerin onay işlemlerinin, yani seçilen derslerin kesinleştirilme işlemlerinin gerçekleştirilmesini sağlamaktır.

Aşağıdaki ekran görüntüsü “Seçilen Dersler” düğmesine tıklandıktan sonra alınmıştır.

Kod	Ad	Kredisi	Z/S	Yan Yılı	Öğretim Üyesi Adı	Onay	Eğitim Yılı	Dönemi	Yan Yılı
İST101	İSTATİSTİĞE GİRİŞ I	3	Z	1	Yrd. Doç. Dr. Meral YAY	<input type="checkbox"/>	2006-2007	GLZ	1
İST103	OLASILIK I	2	Z	1	Prof. Dr. Gülay KIROĞLU	<input type="checkbox"/>	2006-2007	GLZ	1
İST105	BİLGİ İŞLEM I	2	Z	1	Prof. Dr. Aydın ERAR	<input type="checkbox"/>	2006-2007	GLZ	1
İST109	İKTİSAT I	2	Z	1	Yrd. Doç. Dr. Yeşim REEL	<input type="checkbox"/>	2006-2007	GLZ	1

Ekran 8.24. “Ders Seçimi” ekranı.

“Ders Seçimi” ekranında Zorunlu, Alttan Zorunlu, Seçmeli ve Alttan Seçmeli sekmelerinde listelenen derslerden onaylanması istenen derslerin “Onay” çek kutucuğunun seçilip “Kaydet” düğmesine ardından da “Onay” düğmesine basılması gerekmektedir. Böylece, öğrencinin mevcut dönemde alacağı derslerin kesinleştirilme işlemi gerçekleştirilmiş olur.

Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adlı öğrenci için danışmanının Zorunlu sekmesinden “İstatistiğe Giriş I” ve “Olasılık I” derslerini onaylamasından

sonra alınmıştır. Ayrıca ekran görüntüsü Zorunlu sekmesine benzediği için tekrar verilmeyen Seçmeli sekmesinden de danışman, öğrencinin seçmiş olduğu seçmeli derslerden “Çok Amaçlı Programlama” ve “Karar Destek Sistemleri” adlı dersleri tıklayarak onaylamıştır.

Kodu:	Adı:	Kredisi:	Z/S	Yarı Yılı:	Öğretim Üyesi Adı:	Onay:	Eğitim Yılı:	Alınacağı Dönem:	Yarı Yılı:
IST101	İSTATİSTİĞE GİRİŞ I	3	Z	1	Yrd. Doç. Dr. Meral YAY	<input checked="" type="checkbox"/>	2006-2007	GÜZ	1
IST103	OLASILIK I	2	Z	1	Prof. Dr. Gülay KIROĞLU	<input checked="" type="checkbox"/>	2006-2007	GÜZ	1
IST105	BİLGİ İŞLEM I	2	Z	1	Prof. Dr. Aydın ERAR	<input type="checkbox"/>			
IST109	İKTİSAT I	2	Z	1	Yrd. Doç. Dr. Yeşim REEL	<input type="checkbox"/>			

Ekran 8.25. “Ders Seçimi” ekranında “Onay” işlemi.

Aşağıda “Zorunlu” ders seçimlerinin onaylanmasını gerçekleştirmek için “Onay” düğmesine tıkladığında işletilen VB kodları yer almaktadır.

```

Private Sub cmdOnay_Click()
Dim cnThisConnect As ADODB.Connection
Dim rcdSnavlar As New ADODB.Recordset
Dim rcdTümDersler As New ADODB.Recordset
Dim rcdDersler As New ADODB.Recordset
Dim i As Integer
Dim j As Integer
Dim b As Integer
b = Forms!frmDönemDersleriÖğrenci1.ÖğrenciID
Set cnThisConnect = CurrentProject.Connection
rcdSnavlar.Open "tblSnavlar", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdTümDersler.Open "tblÖğrencilerinTümAlmasıGerekenDersleri", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdDersler.Open "tblDersler", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
For i = 1 To rcdTümDersler.RecordCount
If (rcdTümDersler![ÖğrenciID] = b) And (rcdTümDersler![Onay] = True) And
(rcdTümDersler![BaşarıDurumu] = "S") Then
rcdSnavlar.AddNew
rcdSnavlar![DersID] = rcdTümDersler![DersID]
rcdSnavlar![ÖğretimElemanıID] = rcdDersler![ÖğretimElemanıID]
rcdSnavlar![ÖğrenciID] = rcdTümDersler![ÖğrenciID]
rcdSnavlar![EğitimYılı] = gstrEğitimYılı
rcdSnavlar![Dönem] = gstrDönem
rcdSnavlar![Yarıyıl] = gstrYarıyıl
rcdSnavlar.Update
rcdTümDersler![BaşarıDurumu] = "SS"
rcdTümDersler![Seç/Bırak] = True

```

```

rcdTümDersler![EğitimYılı] = gstrEğitimYılı
rcdTümDersler![Dönem] = gstrDönem
rcdTümDersler![Yarıyıl] = gstrYarıyıl
rcdTümDersler.Update
Else
If (rcdTümDersler![ÖğrenciID] = b) And (rcdTümDersler![Onay] = False) And
(rcdTümDersler![BaşarıDurumu] = "S") Then
rcdTümDersler![BaşarıDurumu] = "Y"
rcdTümDersler![Seç/Bırak] = False
rcdTümDersler![EğitimYılı] = ""
rcdTümDersler![Dönem] = ""
rcdTümDersler![Yarıyıl] = ""
rcdTümDersler.Update
rcdSınavlar.Update
End If
End If
rcdTümDersler.MoveNext
Next i
rcdSınavlar.Update
Me.DersKodu.SetFocus
Me.cmdOnay.Enabled = False
    On Error Resume Next
    Screen.PreviousControl.SetFocus
    If Err <> 0 Then
        Me.SetFocus
        Err.Clear
    End If
    If Not SaveIt() Then Exit Sub
End Sub

```

VB kodlarının işletilmesiyle, “Onay” kutucuğu seçili olan zorunlu derslere ilişkin "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosundaki “BaşarıDurumu” alanı “SS” olurken “EğitimYılı”, “Dönem”, “Yarıyıl” alanları da daha önce belirlenen eğitim yılı, dönem ve yarıyıl bilgilerine göre işlenir. Ayrıca onaylanan ders “tblSınavlar” tablosuna eklenir. “Onay” kutucuğu seçili olmayan zorunlu derslere ilişkin "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosundaki “BaşarıDurumu” alanı ise tekrar “Y” olur. “EğitimYılı”, “Dönem” ve “Yarıyıl” alanlarında ise herhangi bir değişiklik olmaz.

“Alttan Zorunlu”, “Seçmeli” ve “Alttan Seçmeli” ders seçimlerinin onaylanmasına ilişkin VB kodları da “Zorunlu” sekmesindeki gibi olup farklı olarak “Alttan Zorunlu” sekmesi için "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosundaki “BaşarıDurumu” alanının “SSF”, “Seçmeli” için SSC ve “Alttan Seçmeli” için SSCF olmasıdır.

Alınan Dersler Düğmesi: Bu düğme görüntüsü ve kullanım mantığı daha önce Kesim 8.4.1.2.’de sözü edilen ve öğrencilerin kullanımına sunulan “Alınan Dersler”

ekranına benzyen bir ekranı açar. Öğretim elemanı bu ekran üzerinde herhangi bir değişiklik yapma yetkisine sahip değildir.

Sınav Bilgileri Düğmesi: “Sınav Bilgileri” ekranını açan bu düğme, Alınan Dersler düğmesi gibi, öğretim elemanın danışmanı olduğu öğrenci hakkında bilgi edinmesi amacıyla kullanılmaktadır. Bu ekranda öğrencinin mevcut dönemde aldığı derslerin sınav sonuçları yer almaktadır. Görünüşü ve çalışma mantığı 8.4.1.2. kesimindeki “Sınav Bilgileri” ekranıyla aynıdır.

İstatistik Düğmesi: İstatistik düğmesi, bölümde okuyan birinci, ikinci, üçüncü, dördüncü ve uzatmalı lisans öğrencilerinin genel toplamlarını, cinsiyetlerine göre toplamlarını veren Ekran 8.26’daki “Lisans Öğrenci İstatistikleri” adındaki bilgi ekranını açar. Öğretim elemanı, bu ekranda eğitim yılı ve dönem seçimi yaptıktan sonra “Göster” düğmesine bastığında, seçtiği eğitim yılı ve dönemde İstatistik Bölümü’nde kayıtlı lisans öğrencilerinin sınıflarına ve cinsiyetlerine göre toplamları ile genel toplamlarını görecektir.

Aşağıdaki ekran görüntüsü “Lisans Öğrenci İstatistikleri” ekranında eğitim yılı olarak “2006-2007”, dönem olarak “Güz” seçimleri yapıp “Göster” düğmesine tıklandıktan sonra alınmıştır.

	TOPLAM ÖĞRENCİ SAYISI	KIZ ÖĞRENCİ SAYISI	ERKEK ÖĞRENCİ SAYISI
I. SINIF:	6	2	4
II. SINIF:	0	0	0
III. SINIF:	0	0	0
IV. SINIF:	0	0	0
+ SINIF:	0	0	0
GENEL TOPLAM:	6	2	4

Ekran 8.26. “Lisans Öğrenci İstatistikleri” ekranı.

“Göster” düğmesine ilişkin VB kodları aşağıda gösterildiği gibidir.

```

Private Sub cmdGöster_Click()
Dim cnThisConnect As ADODB.Connection
Dim rcdÖğrenci As New ADODB.Recordset
Dim rcdÖğrenciSınıf As New ADODB.Recordset
Dim i As Integer
Dim j As Integer
Dim BHKadınI As Integer
Dim BHKadınII As Integer
Dim YLDKadın As Integer
Dim YLTKadın As Integer
Dim DDKadın As Integer
Dim DTKadın As Integer
Dim BHERkekI As Integer
Dim BHERkekII As Integer
Dim YLDERkek As Integer
Dim YLTERkek As Integer
Dim DDERkek As Integer
Dim DTERkek As Integer
Set cnThisConnect = CurrentProject.Connection
rcdÖğrenci.Open "tblYLÖğrencileri", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdÖğrenciSınıf.Open "tblYLÖğrenciSınıf", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
If IsNothing(Me.cmbEğitimYılı) Then
    MsgBox "Lütfen Eğitim Yılımı seçiniz.", vbCritical, gstrAppTitle
    Exit Sub
End If
If IsNothing(Me.CmbDönem) Then
    MsgBox "Lütfen Dönemi seçiniz.", vbCritical, gstrAppTitle
    Exit Sub
End If
For i = 1 To rcdÖğrenci.RecordCount
    rcdÖğrenciSınıf.MoveFirst
    For j = 1 To rcdÖğrenciSınıf.RecordCount
        If rcdÖğrenciSınıf![YLÖğrenciID] = rcdÖğrenci![YLÖğrenciID] And rcdÖğrenciSınıf![EğitimYılı]
= Me.cmbEğitimYılı And rcdÖğrenciSınıf![Dönem] = Me.CmbDönem And
rcdÖğrenciSınıf![ProgramTürü] = "BİLİMSEL HAZIRLIK" Then
            If rcdÖğrenci![Cinsiyet] = "KADIN" Then
                If rcdÖğrenciSınıf![YarıyılTürü] = "BİLİMSEL HAZIRLIK I. DÖNEM" Then
                    BHKadınI = BHKadınI + 1
                ElseIf rcdÖğrenciSınıf![YarıyılTürü] = "BİLİMSEL HAZIRLIK II. DÖNEM" Then
                    BHKadınII = BHKadınII + 1
                End If
            ElseIf rcdÖğrenci![Cinsiyet] = "ERKEK" Then
                If rcdÖğrenciSınıf![Yarıyıl] = "BİLİMSEL HAZIRLIK I. DÖNEM" Then
                    BHERkekI = BHERkekI + 1
                ElseIf rcdÖğrenciSınıf![Yarıyıl] = "BİLİMSEL HAZIRLIK II. DÖNEM" Then
                    BHERkekI = BHERkekI + 1
                End If
            End If
        End If
    End If
End If
If rcdÖğrenciSınıf![YLÖğrenciID] = rcdÖğrenci![YLÖğrenciID] And rcdÖğrenciSınıf![EğitimYılı] =
Me.cmbEğitimYılı And rcdÖğrenciSınıf![Dönem] = Me.CmbDönem And
rcdÖğrenciSınıf![ProgramTürü] = "YÜKSEK LİSANS" Then
    If rcdÖğrenci![Cinsiyet] = "KADIN" Then
        If rcdÖğrenciSınıf![YarıyılTürü] = "DERS AŞAMASI" Then
            YLDKadın = YLDKadın + 1
        ElseIf rcdÖğrenciSınıf![YarıyılTürü] = "TEZ AŞAMASI" Then
            YLTKadın = YLTKadın + 1
        End If
    End If
End If

```

```

ElseIf rcdÖğrenci![Cinsiyet] = "ERKEK" Then
  If rcdÖğrenciSınıf![Yarıyıl] = "DERS AŞAMASI" Then
    YLDERkek = YLDERkek + 1
  ElseIf rcdÖğrenciSınıf![Yarıyıl] = "TEZ AŞAMASI" Then
    YLTERkek = YLTERkek + 1
  End If
End If
End If
End If
If rcdÖğrenciSınıf![YLÖğrenciID] = rcdÖğrenci![YLÖğrenciID] And rcdÖğrenciSınıf![EğitimYılı] =
Me.cmbEğitimYılı And rcdÖğrenciSınıf![Dönem] = Me.CmbDönem And
rcdÖğrenciSınıf![ProgramTürü] = "DOKTORA" Then
  If rcdÖğrenci![Cinsiyet] = "KADIN" Then
    If rcdÖğrenciSınıf![YarıyılTürü] = "DERS AŞAMASI" Then
      DDKadın = DDKadın + 1
    ElseIf rcdÖğrenciSınıf![YarıyılTürü] = "TEZ AŞAMASI" Then
      DTKadın = DTKadın + 1
    End If
  ElseIf rcdÖğrenci![Cinsiyet] = "ERKEK" Then
    If rcdÖğrenciSınıf![Yarıyıl] = "DERS AŞAMASI" Then
      DDERkek = DDERkek + 1
    ElseIf rcdÖğrenciSınıf![Yarıyıl] = "TEZ AŞAMASI" Then
      DTERkek = DTERkek + 1
    End If
  End If
End If
End If
rcdÖğrenciSınıf.MoveNext
Next j
rcdÖğrenci.MoveNext
Next i
Me.txtBHERkekI = BHERkekI
Me.txtBHERkekII = BHERkekII
Me.txtBHGenelErkek = BHERkekI + BHERkekII
Me.txtBHKızI = BHKadınI
Me.txtBHKızII = BHKadınII
Me.txtBHGenelKız = BHKadınI + BHKadınII
Me.txtBHToplamI = BHERkekI + BHKadınI
Me.txtBHToplamII = BHERkekII + BHKadınII
Me.txtBHGenelToplam = BHERkekI + BHKadınI + BHERkekII + BHKadınII
Me.txtDRErkekI = DDERkek
Me.txtDRErkekII = DTERkek
Me.txtDRGenelErkek = DDERkek + DTERkek
Me.txtDRKızI = DDKadın
Me.txtDRKızII = DTKadın
Me.txtDRGenelKız = DDKadın + DTKadın
Me.txtDRToplamDers = DDERkek + DDKadın
Me.txtDRToplamTez = DTERkek + DTKadın
Me.txtDRGenelToplam = DDERkek + DDKadın + DTERkek + DTKadın
Me.txtYLERkekI = YLDERkek
Me.txtYLERkekII = YLTERkek
Me.txtYLGeneErkek = YLDERkek + YLTERkek
Me.txtYLKızI = YLDKadın
Me.txtYLKızII = YLTKadın
Me.txtYLGeneKız = YLDKadın + YLTKadın
Me.txtYLToplamDers = YLDERkek + YLDKadın
Me.txtYLToplamTez = YLTERkek + YLTKadın
Me.txtYLGeneToplam = YLDERkek + YLDKadın + YLTERkek + YLTKadın
End Sub

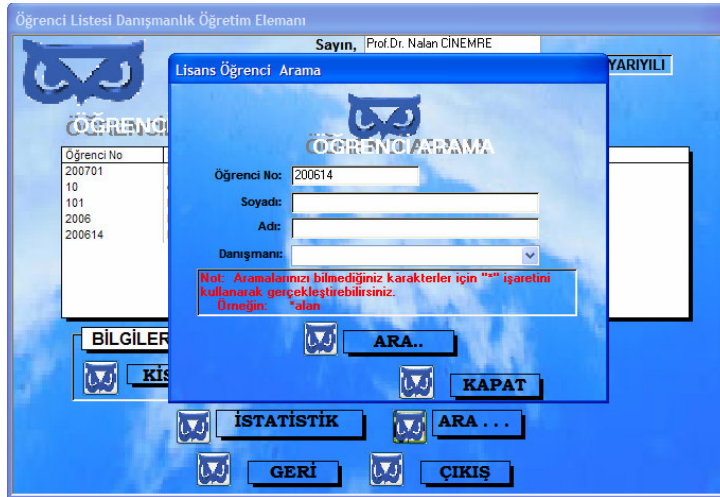
```

Ara Düğmesi: “Ara” düğmesi, Baykuş programına girişi yapılmış lisans öğrencilerini “Öğrenci No” veya “Soyad” veya “Ad” veya “Danışman” kriterlerine göre arayan Ekran 8.27’deki “Lisans Öğrenci Arama” ekranını açar. Arama işleminin gerçekleşebilmesi için en az bir alana değer girilmesi gerekmektedir. Ekrandaki “Ara” düğmesinin işlettiği VB kodları aşağıdaki gibidir.

```
Private Sub cmdSearch_Click()  
Dim varWhere As Variant  
varWhere = Null  
If Not IsNothing(Me.txtÖğrenciNo) Then  
varWhere = "[ÖğrenciNo] LIKE " & Me.txtÖğrenciNo & "*"  
End If  
If Not IsNothing(Me.txtSoyad) Then  
varWhere = (varWhere + " AND ") & "[Soyad] LIKE " & Me.txtSoyad & "*"  
End If  
If Not IsNothing(Me.txtAd) Then  
varWhere = (varWhere + " AND ") & "[Ad] LIKE " & Me.txtAd & "*"  
End If  
If Not IsNothing(Me.CmbÖğretimÜyesi) Then  
varWhere = (varWhere + " AND ") & "[ÖğrÜyesiAdı]= " & Me.CmbÖğretimÜyesi  
End If  
If IsNothing(varWhere) Then  
MsgBox "Aramanın gerçekleşebilmesi için en az bir alana değer girmelisiniz.", vbInformation,  
gstrAppTitle  
Exit Sub  
End If  
If IsNothing(DLookup("ÖğrenciNo", "tblÖğrenci", varWhere)) Then  
MsgBox "Girdiğiniz kriterlere uyan bir öğrenci bulunamadı.", vbInformation, gstrAppTitle  
Exit Sub  
End If  
DoCmd.OpenForm f, WhereCondition:=varWhere  
DoCmd.Close acForm, ff  
DoCmd.Close acForm, Me.Name  
End Sub
```

VB kodları ile girilen kriter veya kriterler ile eşleşen bir kaydın olup olmadığı “tblÖğrenci” tablosundaki bilgilerde aranır. Eğer eşleşen bir kayıt bulunursa, yani girilen kritere sahip bir öğrenci bulunursa, bu öğrenciye ait bilgiler daha önce Kesim 8.4.1.2.’de sözü edilen “Kişisel Bilgiler” ekranına benzer bir ekranda öğretim elemanına sunulur. Aksi takdirde kullanıcıya "Girdiğiniz kriterlere uyan bir öğrenci bulunamadı" şeklinde uyarıcı bir mesaj iletilir.

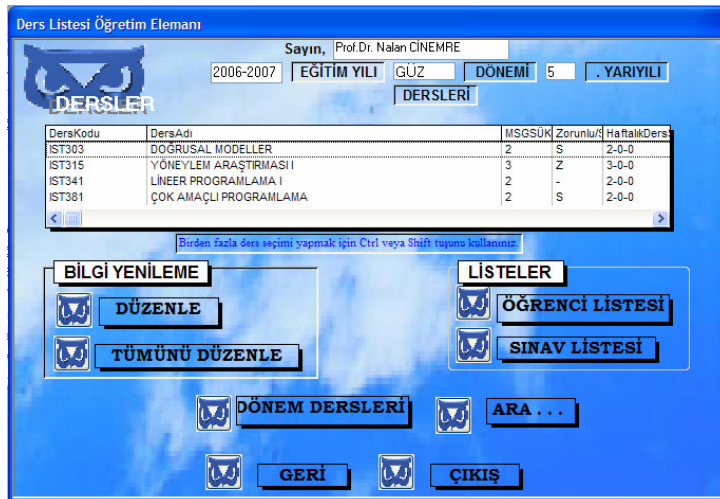
Aşağıdaki ekran görüntüsü öğrenci no alanına “200614” değeri girildikten sonra alınmıştır.



Ekran 8.27. “Lisans Öğrenci Arama” ekranı.

Dersler Düğmesi: Bu düğme, öğretim elemanını, bölümde verdiği derslerin listesini gösteren “Ders Listesi” ekranına ulaştırır (Ekran 8.28). Ama öncesinde kullanıcı “Dönem Seçimi Dersler” ekranıyla karşı karşıya kalır. Bu ekranda eğitim yılı, dönem ve yarıyıl alanları seçilip “Göster” düğmesine tıklaması gerekmektedir.

Aşağıdaki ekran görüntüsü “Dönem Seçimi Dersler” ekranında eğitim yılı olarak 2006-2007, dönem olarak “Güz”, yarıyıl olarak ise “5” seçilip “Göster” düğmesine tıklandıktan sonra alınmıştır.



Ekran 8.28. “Ders Listesi” ekranı.

“Ders Listesi” ekranında yer alan “Düzenle”, “Tümünü Düzenle” düğmeleri “Bilgi Yenileme” ayırımında, “Öğrenci Listesi”, “Sınav Listesi” düğmeleri ise “Listeler” ayırımında yer almaktadır. Ekranda ayrıca “Dönem Dersleri” ve “Ara” düğmeleri de

bulunmaktadır. “Dönem Dersleri” ve “Ara” düğmeleri dışındaki düğmelerin işlevlerini gerçekleştirebilmeleri için listeden bir dersin seçilmiş olması gerekmektedir.

Düzenle Düğmesi: Bu düğme, listede seçilen derslerin kod, ad, zorunlu/seçmeli, kredi, vb. gibi bilgilerinin değiştirilmesini sağlayan “Ders Düzenle” ekranını açar. Bu ekran Kesim 8.4.2.4.’de sözü edilen “Ders Düzenle” ekranıyla benzer kullanım ve işleve sahiptir.

Tümünü Düzenle Düğmesi: Bu düğme “Dersler Listesi” ekranında listelenen tüm derslerin düzenlenmek üzere “Ders Düzenle” ekranında bulunmasını sağlar.

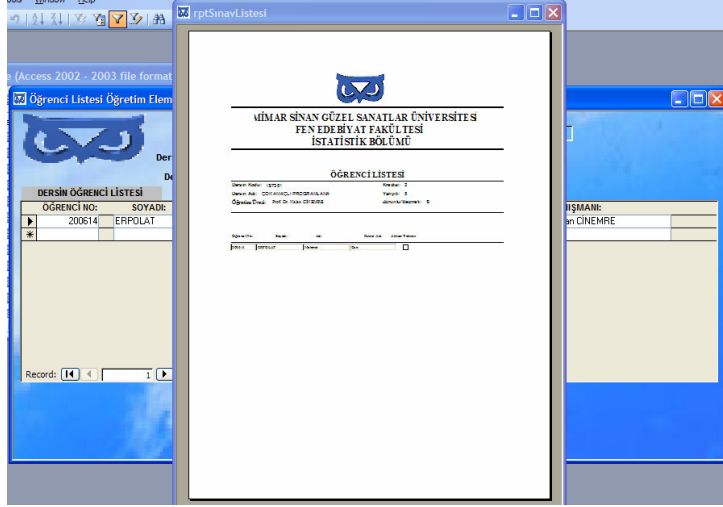
Öğrenci Listesi Düğmesi: Bu düğme, listedeki derslerin kaç öğrenci tarafından seçildiğini gösteren listeyi veren ekranı açar.

Aşağıdaki ekran görüntüsü “Dersler Listesi” ekranındaki derslerden “Çok Amaçlı Programlama” dersi seçilip “Öğrenci Listesi” düğmesine tıklandıktan sonra alınmıştır.

ÖĞRENCİ NO	SOYAD	ADI	İKİNCİ ADI	ALT TAN TEKRAR	DANIŞMANI
200614	ERPOLAT	Mehmet	Can	<input type="checkbox"/>	Prof. Dr. Nalan CİNEMRE

Ekran 8.29 . “Öğrenci Listesi” ekranı.

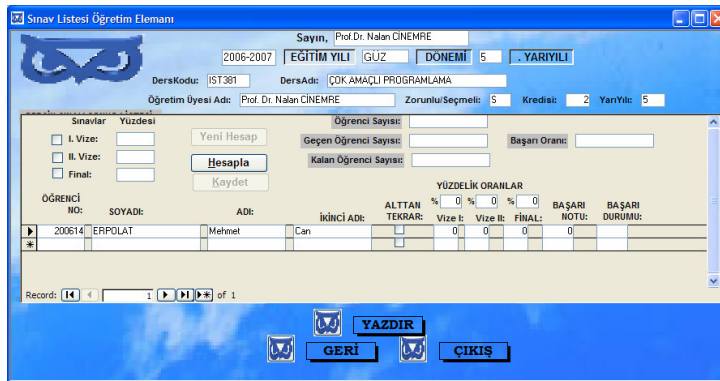
Ekran üzerinde bulunan “Yazdır” düğmesine tıklandıktan sonra Ekran 8.50’de gösterilen “Öğrenci Raporları” print ekranına benzer bir ekran açılır. Bu ekranda yer alan “Print” düğmesine tıklandığında ise Ekran 8.30’da gösterilen rapora ulaşılır.



Ekran 8.30. “Öğrenci Listesi” raporu.

Sınav Listesi Düğmesi: Bu düğme, listedeki derslerin sınav sonuçlarının programa girilmesini sağlayan “Sınav Listesi” ekranını açar. Öğretim elemanı bu ekran üzerinde dersi alan öğrencilerin sınav sonuçlarını girip belirlediği yüzdelerle başarı notu ve durumlarını hesaplar.

Aşağıdaki ekran görüntüsü “Dersler Listesi” ekranında yer alan derslerden “Çok Amaçlı Programlama” dersi seçilip “Sınav Listesi” düğmesine tıklandıktan sonra alınmıştır.



Ekran 8.31. “Sınav Listesi” ekranı.

Öğretim elemanı bu ekranda not girişlerini gerçekleştirdikten sonra “Kaydet” düğmesine basar. Böylece notların kayıt işlemi gerçekleşmiş olur. Ardından sınavların yüzdelerini belirleme işlemine geçer. Ekranın sol tarafında yer alan yüzdeler ve sınav türü belirleme kısmında yapılan sınavlara ilişkin çek kutuları seçilip sınav yüzdeler oranları girildikten sonra “Hesapla” Düğmesine basılır. Burada

dikkat edilmesi gereken nokta sınav yüzdelik toplamının “100” olmasıdır. “Hesapla” düğmesinin VB kodları aşağıdaki gibidir.

```
Private Sub cmdHesapla_Click()
Dim cnThisConnect As ADODB.Connection
Dim rcdSınavlar As New ADODB.Recordset
Dim rcdTümDersler As New ADODB.Recordset
Dim i As Integer
Dim j As Integer
Dim a As String
Dim VI As Integer
Dim VII As Integer
Dim f As Integer
Dim ÖğrenciSayısı As Integer
Dim GeçenSayısı As Integer
Dim KalanSayısı As Integer
Dim BaşarıOranı As Integer
Set cnThisConnect = CurrentProject.Connection
rcdSınavlar.Open "tblSınavlar", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdTümDersler.Open "tblÖğrencilerinTümAlmasıGerekenDersleri", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
'Vize I chkbox
If (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = False) And (Me.txtFinal = "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = False) And (Me.txtFinal = "") Then
    Me.chkVizeII.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal = "") Then
    Me.chkFinal.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal = "") Then
    Me.chkVizeII.Value = False
    Me.chkFinal.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'Vize II chkbox
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal = "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal = "") Then
    Me.chkVizeI.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
End Sub
```



```

ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal = "") Then
    Me.chkFinal.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal = "") Then
    Me.chkVizeI.Value = False
    Me.chkFinal.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'Final chkbox
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.chkVizeI.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.chkVizeII.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.chkVizeII.Value = False
    Me.chkVizeI.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'VizeI ve VizeII chkbox
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal = "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal = "") Then
    Me.chkFinal.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'VizeI ve Final chkbox
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.chkVizeII.Value = False
    Me.cmdYeniHesap.Enabled = True

```

```

Me.cmdYeniHesap.SetFocus
Me.cmdHesapla.Enabled = False
'VizeII ve Final checkbox
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.chkVizeI.Value = False
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'VizeI, VizeII ve Final checkbox
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'Vize I txtbox
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal = "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = False) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'Vize II txtbox
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal = "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'Final txtbox
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII = "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI = "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then

```

```

Me.cmdYeniHesap.Enabled = True
Me.cmdYeniHesap.SetFocus
Me.cmdHesapla.Enabled = False
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'VizeI ve Vize II txtbox
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = False) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'VizeI ve Final txtbox
ElseIf (Me.chkVizeI.Value = True) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = False) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
'VizeII ve Final txtbox
ElseIf (Me.chkVizeI.Value = False) And (Me.txtVizeI <> "") And (Me.chkVizeII.Value = True) And
(Me.txtVizeII <> "") And (Me.chkFinal.Value = True) And (Me.txtFinal <> "") Then
    Me.cmdYeniHesap.Enabled = True
    Me.cmdYeniHesap.SetFocus
    Me.cmdHesapla.Enabled = False
Else
    Me.chkVizeII.Value = False
    Me.chkVizeI.Value = False
    Me.chkFinal.Value = False
    Me.txtFinal = ""
    Me.txtVizeI = ""
    Me.txtVizeII = ""
    Me.cmdYeniHesap.Enabled = False
    Me.cmdHesapla.Enabled = True
    Me.txtVizeI.SetFocus
    MsgBox "Lütfen sınavları ve yüzdeler oranlarını seçiniz.", vbCritical, gstrAppTitle
    Exit Sub
End If
If (Me.chkVizeI.Value = True) And (Me.chkVizeII.Value = True) And (Me.chkFinal.Value = True)
Then
    If ((Me.txtVizeI) + (Me.txtVizeII) + (Me.txtFinal)) <> 100 Then
        MsgBox "Lütfen seçtiğiniz sınavlara ilişkin yüzdeler oran toplamlarının '100' " & _
            "olmasına dikkat ediniz.", vbCritical, gstrAppTitle
        Me.txtVizeI = ""
        Me.txtVizeII = ""
        Me.txtFinal = ""
        Me.chkVizeI = False
        Me.chkVizeII = False
        Me.chkFinal = False
        Me.txtVizeI.SetFocus
        Me.cmdYeniHesap.Enabled = False
        Me.cmdHesapla.Enabled = True
        Exit Sub
    End If
ElseIf (Me.chkVizeII.Value = True) And (Me.chkFinal.Value = True) Then
    If ((Me.txtVizeII) + (Me.txtFinal)) <> 100 Then
        MsgBox "Lütfen seçtiğiniz sınavlara ilişkin yüzdeler oran toplamlarının '100' " & _
            "olmasına dikkat ediniz.", vbCritical, gstrAppTitle
        Me.txtVizeI = ""

```

```

Me.txtVizeII = ""
Me.txtFinal = ""
Me.chkVizeI = False
Me.chkVizeII = False
Me.chkFinal = False
Me.txtVizeI.SetFocus
Me.cmdYeniHesap.Enabled = False
Me.cmdHesapla.Enabled = True
Exit Sub
End If
Elseif (Me.chkVizeI.Value = True) And (Me.chkFinal.Value = True) Then
If ((Me.txtVizeI) + (Me.txtFinal)) <> 100) Then
MsgBox "Lütfen seçtiğiniz sınavlara ilişkin yüzdeler oran toplamlarının '100' " & _
"olmasına dikkat ediniz.", vbCritical, gstrAppTitle
Me.txtVizeI = ""
Me.txtVizeII = ""
Me.txtFinal = ""
Me.chkVizeI = False
Me.chkVizeII = False
Me.chkFinal = False
Me.txtVizeI.SetFocus
Me.cmdYeniHesap.Enabled = False
Me.cmdHesapla.Enabled = True
Exit Sub
End If
Elseif (Me.chkVizeI.Value = True) And (Me.chkVizeII.Value = True) Then
If (((Me.txtVizeI) + (Me.txtVizeII)) <> 100) Then
MsgBox "Lütfen seçtiğiniz sınavlara ilişkin yüzdeler oran toplamlarının '100' " & _
"olmasına dikkat ediniz.", vbCritical, gstrAppTitle
Me.txtVizeI = ""
Me.txtVizeII = ""
Me.txtFinal = ""
Me.chkVizeI = False
Me.chkVizeII = False
Me.chkFinal = False
Me.txtVizeI.SetFocus
Me.cmdYeniHesap.Enabled = False
Me.cmdHesapla.Enabled = True
Exit Sub
End If
Elseif (Me.chkFinal.Value = True) Then
If (Me.txtFinal) <> 100) Then
MsgBox "Lütfen seçtiğiniz sınavlara ilişkin yüzdeler oran toplamlarının '100' " & _
"olmasına dikkat ediniz.", vbCritical, gstrAppTitle
Me.txtVizeI = ""
Me.txtVizeII = ""
Me.txtFinal = ""
Me.chkVizeI = False
Me.chkVizeII = False
Me.chkFinal = False
Me.txtVizeI.SetFocus
Me.cmdYeniHesap.Enabled = False
Me.cmdHesapla.Enabled = True
Exit Sub
End If
Elseif (Me.chkVizeII.Value = True) Then
If (Me.txtVizeII) <> 100) Then
MsgBox "Lütfen seçtiğiniz sınavlara ilişkin yüzdeler oran toplamlarının '100' " & _
"olmasına dikkat ediniz.", vbCritical, gstrAppTitle
Me.txtVizeI = ""

```

```

Me.txtVizeII = ""
Me.txtFinal = ""
Me.chkVizeI = False
Me.chkVizeII = False
Me.chkFinal = False
Me.txtVizeI.SetFocus
Me.cmdYeniHesap.Enabled = False
Me.cmdHesapla.Enabled = True
Exit Sub
End If
ElseIf (Me.chkVizeI.Value = True) Then
If (Me.txtVizeI) <> 100 Then
MsgBox "Lütfen seçtiğiniz sınavlara ilişkin yüzdeler oran toplamının '100' " & _
"olmasına dikkat ediniz.", vbCritical, gstrAppTitle
Me.txtVizeI = ""
Me.txtVizeII = ""
Me.txtFinal = ""
Me.chkVizeI = False
Me.chkVizeII = False
Me.chkFinal = False
Me.txtVizeI.SetFocus
Me.cmdYeniHesap.Enabled = False
Me.cmdHesapla.Enabled = True
Exit Sub
End If
End If
If (Me.chkFinal = True) And (Me.chkVizeII = True) And (Me.chkVizeI = True) Then
For i = 1 To rcdSınavlar.RecordCount
If rcdSınavlar![DersID] = Forms!frmSınavListesiÖğretimElemanı.DersID Then
rcdSınavlar![YüzdeVizeI] = Me.txtVizeI
rcdSınavlar![YüzdeVizeII] = Me.txtVizeII
rcdSınavlar![YüzdeFinal] = Me.txtFinal
rcdSınavlar![BaşarıNotu] = (((Me.txtFinal) / 100) * rcdSınavlar![Final]) + (((Me.txtVizeII) / 100)
* rcdSınavlar![Vize II]) + (((Me.txtVizeI) / 100) * rcdSınavlar![Vize I])
If (89 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 101) Then
rcdSınavlar![BaşarıDurumu] = "A"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (79 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 90) Then
rcdSınavlar![BaşarıDurumu] = "B"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (69 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 80) Then
rcdSınavlar![BaşarıDurumu] = "C"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (59 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 70) Then
rcdSınavlar![BaşarıDurumu] = "D"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1

ElseIf (49 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 60) Then
rcdSınavlar![BaşarıDurumu] = "E"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (0 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 50) Then
rcdSınavlar![BaşarıDurumu] = "F"
KalanSayısı = KalanSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
End If

```

```

End If
rcdSınavlar.MoveNext
Next i
ElseIf (Me.chkVizeII = True) And (Me.chkVizeI = True) Then
    Me.txtFinal = ""
    f = 0
    Me.chkFinal = False
For i = 1 To rcdSınavlar.RecordCount
If rcdSınavlar![DersID] = Forms!frmSınavListesiÖğretimElemanı.DersID Then
    rcdSınavlar![YüzdeVizeI] = Me.txtVizeI
    rcdSınavlar![YüzdeVizeII] = Me.txtVizeII
    rcdSınavlar![YüzdeFinal] = f
    rcdSınavlar![BaşarıNotu] = (((Me.txtVizeII) / 100) * rcdSınavlar![Vize II]) + (((Me.txtVizeI) / 100)
* rcdSınavlar![Vize I])
    If (89 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 101) Then
        rcdSınavlar![BaşarıDurumu] = "A"
        GeçenSayısı = GeçenSayısı + 1
        ÖğrenciSayısı = ÖğrenciSayısı + 1
    ElseIf (79 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 90) Then
        rcdSınavlar![BaşarıDurumu] = "B"
        GeçenSayısı = GeçenSayısı + 1
        ÖğrenciSayısı = ÖğrenciSayısı + 1
    ElseIf (69 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 80) Then
        rcdSınavlar![BaşarıDurumu] = "C"
        GeçenSayısı = GeçenSayısı + 1
        ÖğrenciSayısı = ÖğrenciSayısı + 1
    ElseIf (59 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 70) Then
        rcdSınavlar![BaşarıDurumu] = "D"
        GeçenSayısı = GeçenSayısı + 1
        ÖğrenciSayısı = ÖğrenciSayısı + 1
    ElseIf (49 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 60) Then
        rcdSınavlar![BaşarıDurumu] = "E"
        GeçenSayısı = GeçenSayısı + 1
        ÖğrenciSayısı = ÖğrenciSayısı + 1
    ElseIf (0 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 50) Then
        rcdSınavlar![BaşarıDurumu] = "F"
        KalanSayısı = KalanSayısı + 1
        ÖğrenciSayısı = ÖğrenciSayısı + 1
    End If
End If
rcdSınavlar.MoveNext
Next i
ElseIf (Me.chkFinal = True) And (Me.chkVizeI = True) Then
Me.txtVizeI = ""
    Me.txtFinal = ""
    f = 0
    Me.chkVizeI = False
    Me.chkFinal = False
For i = 1 To rcdSınavlar.RecordCount
If rcdSınavlar![DersID] = Forms!frmSınavListesiÖğretimElemanı.DersID Then
    rcdSınavlar![YüzdeVizeI] = Me.txtVizeI
    rcdSınavlar![YüzdeVizeII] = Me.txtVizeII
    rcdSınavlar![YüzdeFinal] = f
    rcdSınavlar![BaşarıNotu] = (((Me.txtFinal) / 100) * rcdSınavlar![Final]) + (((Me.txtVizeI) /
100) * rcdSınavlar![Vize I])
    If (89 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 101) Then
        rcdSınavlar![BaşarıDurumu] = "A"
        GeçenSayısı = GeçenSayısı + 1
        ÖğrenciSayısı = ÖğrenciSayısı + 1
    ElseIf (79 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 90) Then

```

```

rcdSnavlar![BaşarıDurumu] = "B"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (69 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 80) Then
rcdSnavlar![BaşarıDurumu] = "C"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (59 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 70) Then
rcdSnavlar![BaşarıDurumu] = "D"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (49 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 60) Then
rcdSnavlar![BaşarıDurumu] = "E"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (0 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 50) Then
rcdSnavlar![BaşarıDurumu] = "F"
KalanSayısı = KalanSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
End If
End If
rcdSnavlar.MoveNext
Next i
ElseIf (Me.chkFinal = True) And (Me.chkVizeII = True) Then
Me.txtVizeII = ""
Me.txtFinal = ""
VII = 0
f = 0
Me.chkVizeII = False
Me.chkFinal = False
For i = 1 To rcdSnavlar.RecordCount
If rcdSnavlar![DersID] = Forms!frmSnavListesiÖğretimElemanı.DersID Then
rcdSnavlar![YüzdeVizeI] = Me.txtVizeI
rcdSnavlar![YüzdeVizeII] = VII
rcdSnavlar![YüzdeFinal] = f
rcdSnavlar![BaşarıNotu] = (((Me.txtFinal) / 100) * rcdSnavlar![Final])) + (((Me.txtVizeII) / 100)
* rcdSnavlar![Vize II])
If (89 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 101) Then
rcdSnavlar![BaşarıDurumu] = "A"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (79 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 90) Then
rcdSnavlar![BaşarıDurumu] = "B"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (69 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 80) Then
rcdSnavlar![BaşarıDurumu] = "C"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (59 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 70) Then
rcdSnavlar![BaşarıDurumu] = "D"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (49 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 60) Then
rcdSnavlar![BaşarıDurumu] = "E"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (0 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 50) Then
rcdSnavlar![BaşarıDurumu] = "F"
KalanSayısı = KalanSayısı + 1

```

```

ÖğrenciSayısı = ÖğrenciSayısı + 1
End If
End If
rcdSınavlar.MoveNext
Next i
ElseIf (Me.chkVizeI = True) Then
    Me.txtVizeII = ""
    Me.txtFinal = ""
    VII = 0
    f = 0
    Me.chkVizeII = False
    Me.chkFinal = False
For i = 1 To rcdSınavlar.RecordCount
If rcdSınavlar![DersID] = Forms!frmSınavListesiÖğretimElemanı.DersID Then
rcdSınavlar![YüzdeVizeI] = Me.txtVizeI
rcdSınavlar![YüzdeVizeII] = VII
rcdSınavlar![YüzdeFinal] = f
rcdSınavlar![BaşarıNotu] = (((Me.txtVizeI) / 100) * rcdSınavlar![Vize I])
If (89 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 101) Then
    rcdSınavlar![BaşarıDurumu] = "A"
    GeçenSayısı = GeçenSayısı + 1
    ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (79 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 90) Then
    rcdSınavlar![BaşarıDurumu] = "B"
    GeçenSayısı = GeçenSayısı + 1
    ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (69 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 80) Then
    rcdSınavlar![BaşarıDurumu] = "C"
    GeçenSayısı = GeçenSayısı + 1
    ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (59 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 70) Then
    rcdSınavlar![BaşarıDurumu] = "D"
    GeçenSayısı = GeçenSayısı + 1
    ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (49 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 60) Then
    rcdSınavlar![BaşarıDurumu] = "E"
    GeçenSayısı = GeçenSayısı + 1
    ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (0 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 50) Then
    rcdSınavlar![BaşarıDurumu] = "F"
    KalanSayısı = KalanSayısı + 1
    ÖğrenciSayısı = ÖğrenciSayısı + 1
End If
End If
rcdSınavlar.MoveNext
Next i
ElseIf (Me.chkVizeII = True) Then
    Me.txtVizeI = ""
    Me.txtFinal = ""
    VI = 0
    f = 0
    Me.chkVizeI = False
    Me.chkFinal = False
For i = 1 To rcdSınavlar.RecordCount
If rcdSınavlar![DersID] = Forms!frmSınavListesiÖğretimElemanı.DersID Then
rcdSınavlar![YüzdeVizeI] = VI
rcdSınavlar![YüzdeVizeII] = Me.txtVizeII
rcdSınavlar![YüzdeFinal] = f
rcdSınavlar![BaşarıNotu] = (((Me.txtVizeII) / 100) * rcdSınavlar![Vize II])
If (89 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 101) Then

```



```

rcdSnavlar![BaşarıDurumu] = "A"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (79 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 90) Then
rcdSnavlar![BaşarıDurumu] = "B"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (69 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 80) Then
rcdSnavlar![BaşarıDurumu] = "C"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (59 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 70) Then
rcdSnavlar![BaşarıDurumu] = "D"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (49 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 60) Then
rcdSnavlar![BaşarıDurumu] = "E"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (0 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 50) Then
rcdSnavlar![BaşarıDurumu] = "F"
KalanSayısı = KalanSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
End If
End If
rcdSnavlar.MoveNext
Next i
ElseIf (Me.chkFinal = True) Then
Me.txtVizeI = ""
Me.txtVizeII = ""
VI = 0
VII = 0
Me.chkVizeI = False
Me.chkVizeII = False
For i = 1 To rcdSnavlar.RecordCount
If rcdSnavlar![DersID] = Forms!frmSnavListesiÖğretimElemanı.DersID Then
rcdSnavlar![YüzdeVizeI] = VI
rcdSnavlar![YüzdeVizeII] = VII
rcdSnavlar![YüzdeFinal] = Me.txtFinal
rcdSnavlar![BaşarıNotu] = (((Me.txtFinal) / 100) * rcdSnavlar![Final])
If (89 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 101) Then
rcdSnavlar![BaşarıDurumu] = "A"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (79 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 90) Then
rcdSnavlar![BaşarıDurumu] = "B"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (69 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 80) Then
rcdSnavlar![BaşarıDurumu] = "C"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (59 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 70) Then
rcdSnavlar![BaşarıDurumu] = "D"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1
ElseIf (49 < rcdSnavlar![BaşarıNotu]) And (rcdSnavlar![BaşarıNotu] < 60) Then
rcdSnavlar![BaşarıDurumu] = "E"
GeçenSayısı = GeçenSayısı + 1
ÖğrenciSayısı = ÖğrenciSayısı + 1

```

```

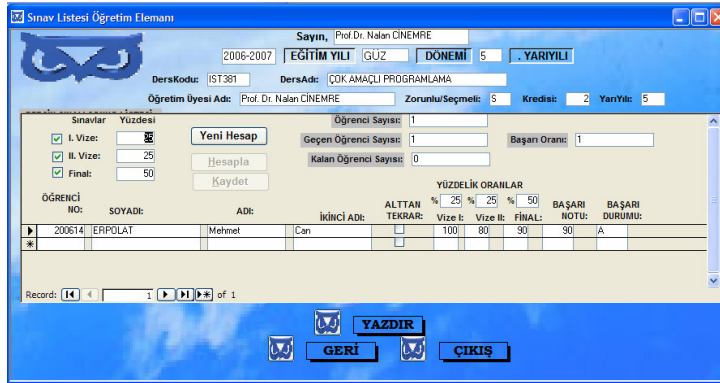
Elseif (0 < rcdSınavlar![BaşarıNotu]) And (rcdSınavlar![BaşarıNotu] < 50) Then
    rcdSınavlar![BaşarıDurumu] = "F"
    KalanSayısı = KalanSayısı + 1
    ÖğrenciSayısı = ÖğrenciSayısı + 1
End If
End If
rcdSınavlar.MoveNext
Next i
End If
DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
    rcdSınavlar.MoveFirst
For i = 1 To rcdSınavlar.RecordCount
    rcdTümDersler.MoveFirst
    For j = 1 To rcdTümDersler.RecordCount
        If (rcdTümDersler![ÖğrenciID] = rcdSınavlar![ÖğrenciID]) And (rcdTümDersler![DersID] =
rcdSınavlar![DersID]) And (rcdTümDersler![EğitimYılı] = rcdSınavlar![EğitimYılı]) And
(rcdTümDersler![Dönem] = rcdSınavlar![Dönem]) And (rcdTümDersler![Yarıyıl] =
rcdSınavlar![Yarıyıl]) Then
            rcdTümDersler![BaşarıDurumu] = rcdSınavlar![BaşarıDurumu]
            If rcdSınavlar![BaşarıDurumu] = "F" Then
                rcdTümDersler![Seç/Bırak] = False
                rcdTümDersler![Onay] = False
                rcdTümDersler![EğitimYılı] = ""
                rcdTümDersler![Dönem] = ""
                rcdTümDersler![Yarıyıl] = ""
            End If
        End If
    Next j
    rcdTümDersler.MoveNext
Next j
    rcdSınavlar.MoveNext
Next i
DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
Me.RecordSource = "qrySınavListesiÖğretimElemanı1"
Me.txtÖğrenciSayısı = ÖğrenciSayısı
Me.txtGeçenSayısı = GeçenSayısı
Me.txtKalanSayısı = KalanSayısı
Me.txtBaşarıOranı = (GeçenSayısı / ÖğrenciSayısı)
Me.txtVizeI.SetFocus
Me.cmdHesapla.Enabled = False
Me.cmdYeniHesap.Enabled = True
End Sub

```

“Hesapla” düğmesi, belirlenen sınav ve yüzdelerine göre öğrencilerin notlarını toplayıp ortalama olarak başarı notunu bulur ve bulduğu bu notun başarı durumunu, başarı notu 90-100 olanlar için “A”, 80-89 olanlar için “B”, 70-79 olanlar için “C”, 60-69 olanlar için “D”, 50-59 olanlar için “E” ve 0-49 olanlar için ise “F” olarak atar. Bu değerler “tblSınavlar” tablosuna işlenir.

Aşağıdaki ekran görüntüsü “Çok Amaçlı Programlama” dersini alan öğrencilerin sınav sonuçları girilip “Kaydet” düğmesine basıldıktan sonra yapılan sınavlar ile bu sınavların “Başarı Notu”nu belirlemedeki yüzdeleri belirlenip “Hesapla” düğmesine

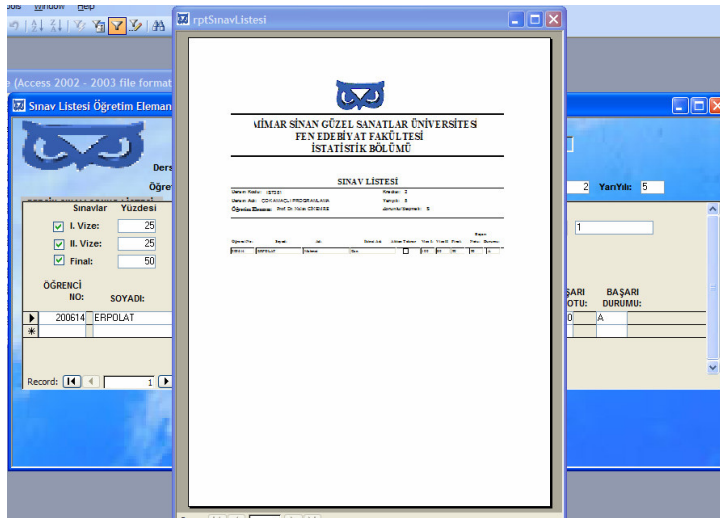
basıldıktan sonra alınmıştır. Sınav türleri olarak “Vize I, Vize II ve Final” sınavları seçilmiş olup bu sınavların ağırlıkları sırasıyla “25, 25 ve 50” olarak alınmıştır.



Ekran 8.32. Not hesaplaması yapıldıktan sonraki “Sınav Listesi” ekranı.

Ekranında ayrıca, dersi alan toplam öğrenci sayısı, dersten geçer not alan toplam öğrenci sayısı, başarısız olan öğrenci sayısı ve başarı oranları da hesaplanıp kullanıcıya sunulmaktadır.

“Sınav Listesi” ekranında “Yazdır” düğmesine tıklandığında açılan print ekranındaki “Print” düğmesinin tıklanmasıyla Ekran 8.33’de gösterilen raporla karşılaşılır.



Ekran 8.33. “Sınav Listesi” raporu.

8.4.3.3. Lisans Üstü Öğrencileri Ayırımı

“Lisans Üstü Öğrencileri” ayırımında yeralan düğme ve bu düğmelerin işleyişleri Kesim 8.4.3.2’de sözü edilen düğmelerle aynı olduğundan ayrıca değinilmeyecektir.

Tek farklılık, yapılan işlemlerin Baykuş programında lisans üstü öğrencileri ilgilendiren tablolar kullanılarak gerçekleştiriliyor olmasıdır.

8.4.4. BÖLÜM BAŞKANI GİRİŞİ

Bölüm Başkanı girişi, sadece bölüm başkanı tarafından kullanılabilen giriştir. Bu girişte bölüm başkanı bölümdeki bütün öğretim elemanlarının kişisel bilgilerine; lisans, bilimsel hazırlık, yüksek lisans ve doktora öğrencilerinin kişisel bilgileri ve bölüm bilgilerine, bölümde okutulan bütün lisans, bilimsel hazırlık, yüksek lisans ve doktora derslerine ulaşma, değiştirme ve silme yetkilerine sahiptir.

Bu giriş Baykuş programının güncellemelerinin, düzeltmelerinin veya eklemelerinin yapılmasını sağlayan ekranları içermektedir.

Bölüm Başkanı Girişi düğmesine tıklandığından diğer düğmelerde olduğu gibi bir kullanıcı adı ve şifre doğrulama ekranı ile karşılaşılır. “Bölüm Başkanı Sign On” adı verilen bu ekranın görünüşü ve dayandığı VB kodları diğer “Sign On” ekranların VB kodlarının benzeri olup tek farkı girilen kullanıcı adı ve şifrenin “tblBölümBaşkanıKullanıcıları” tablosunda aranıyor olmasıdır.

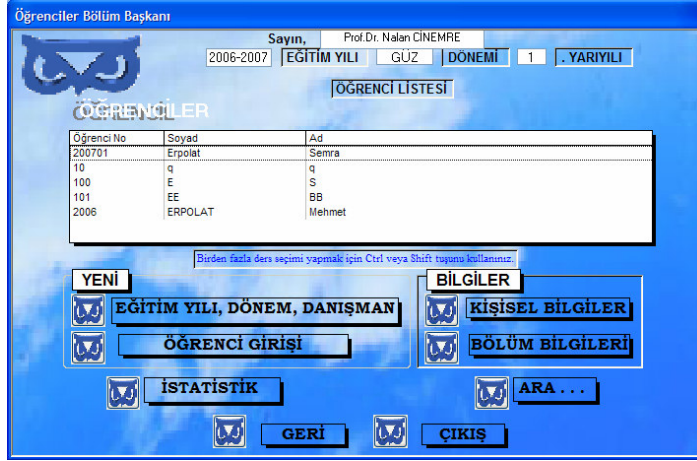
Açılan “Sign On” penceresinde doğru kullanıcı ve şifre girildiği takdirde Ekran 8.34’de gösterilen “Bölüm Başkanı Main” ekranı açılır.

Aşağıdaki ekran görüntüsü, İstatistik bölüm başkanı “Prof. Dr. Nalan CİNEMRE”nin kullanıcı adı ve şifresi girilerek alınmıştır.



Ekran 8.34 . “Bölüm Başkanı Main” ekranı.

Aşağıdaki ekran görüntüsü, “Dönem Seçimi” ekranında eğitim yıllı olarak “2006-2007”, dönem olarak “GÜZ”, yarıyıl olarak “1” seçilip “Göster” düğmesine basıldıktan sonra görüntülenen “Lisans Öğrencileri” ekranıdır.



Ekran 8.36 . “Lisans Öğrencileri” ekranı.

Lisans Öğrencileri ekranı “Yeni” ve “Bilgiler” olmak üzere iki ayrıma sahiptir. “Yeni” ayırımında ekranda listelenen lisans öğrencilerine ilişkin yeni eğitim yılı, dönem, yarıyıl ve danışman ataması yapmayı sağlayacak ekrana ulaştıran düğme ile yeni öğrenci girişini sağlayan düğme, “Bilgiler” ayırımında ise listede yer alan lisans öğrencilerine ilişkin kişisel bilgileri ve bölüm bilgilerini gösterecek olan ekranlara ulaştıran düğmeler yer almaktadır. Ekranda ayrıca “İstatistik” ve “Ara” düğmeleri de yer almaktadır.

“Yeni Eğitim Yılı, Dönem, Yarıyıl ve Danışman”, “Kişisel Bilgiler” ve “Bölüm Bilgileri” düğmelerinin işlevlerini gerçekleştirebilmeleri için listede yer alan lisans öğrencilerinden birinin seçilmiş olması gerekir.

Yeni Ayırımı: Listedenden seçilen öğrenciye ilişkin yeni eğitim yılı, dönem, yarıyıl ve danışman girişinin ve ilk kez kadı yapılacak yeni öğrenci girişlerinin yapılmasını sağlayan düğmeleri içinde bulunduran ayırımdır.

Yeni Eğitim Yılı, Dönem, Yarıyıl Ve Danışman Düğmesi: “Yeni Eğitim Yılı, Dönem, Yarıyıl ve Danışman” düğmesi, tıklandığında kullanıcıyı “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu” adındaki forma ulaştıran düğmedir.

Ekran 8.37’de gösterilen ekran görüntüsü, Lisans Öğrencileri ekranındaki listeden “Mehmet Can ERPOLAT” adlı öğrenci seçilip “Yeni Eğitim Yılı, Dönem, Yarıyıl ve Danışman” düğmesine tıklandıktan sonra alınmıştır.

Öğrencilerin Eğitim Yılı Dönem Yarıyıl Durumu Bölüm Başkanı

Sayın, Prof. Dr. Nalan CINEMRE

2006-2007 EĞİTİM YILI GÜZ DÖNEM 1 YARIYILI

Öğrenci No: 2006

Soyad: ERPOLAT

Ad: Mehmet İkinci Ad: Can

EĞİTİM YILI DÖNEM YARIYIL

Eğitim Yılı	Dönem	Yarıyıl	Expr1004
2006-2007	GÜZ	1	Prof. Dr. Nalan CINEMRE

Bir den fazla ders seçimi yapmak için Ctrl veya Shift tuşunu kullanınız

BİLGİ YENİLEME

DÜZENLE YENİ

TÜMÜNÜ DÜZENLE

GERİ ÇIKIŞ

Ekran 8.37. “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu” ekranı.

Bu form, listeden seçilen öğrencinin İstatistik Bölümüne kayıt olduğu eğitim yılından o andaki döneme kadar tamamladığı bütün eğitim yıllarını, dönemlerini, yarıyıllarını ve danışmanlarını listeler. Form üzerinde “Düzenle”, “Tümünü Düzenle” ve “Yeni” olmak üzere üç düğme bulunmaktadır.

Düzenle Düğmesi: Düzenle düğmesi, kullanıcıyı “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu Düzenle” formuna ulaştırır (Ekran 8.38). Düğmenin işlevini gerçekleştirebilmesi, kullanıcının düzenlemek istediği bilgiyi listeden seçmiş olmasına bağlıdır. Formun kayıt kaynağının dayandığı SQL ifadesi aşağıdaki gibidir.

```
SELECT tblÖğrenciSınıf.ÖğrenciSınıfID, tblÖğrenciSınıf.EğitimYılı, tblÖğrenciSınıf.Dönem,
tblÖğrenciSınıf.Yarıyıl, tblÖğrenciSınıf.Danışman, tblÖğrenciSınıf.ÖğrenciID,
tblÖğrenciSınıf.ÖğrenciID, tblÖğrenci.ÖğrenciNo, tblÖğrenci.Soyad, tblÖğrenci.Ad,
tblÖğrenci.İkinciAd, tblÖğrenci.Danışman, tblÖğrenci.İkinciAd, tblÖğrenciSınıf.Danışman
FROM tblÖğrenci INNER JOIN tblÖğrenciSınıf ON tblÖğrenci.ÖğrenciID =
tblÖğrenciSınıf.ÖğrenciID
WHERE
(((tblÖğrenci.ÖğrenciNo)=[Forms]![frmÖğrencilerinEğitimYılıDönemYarıyılDurumuBölümBaşkanı]
![ÖğrenciNo]))
ORDER BY tblÖğrenciSınıf.EğitimYılı, tblÖğrenciSınıf.Dönem, tblÖğrenciSınıf.Yarıyıl;
```

Yukarıdaki sorguda “tblÖğrenci” tablosu ile “tblÖğrenciSınıf” tablosu “ÖğrenciID” alanına göre ilişkilendirilmiştir.

Aşağıdaki ekran görüntüsü, ilgili öğrencinin “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu” ekranında listelenen satır seçilip “Düzenle” düğmesine tıklandıktan sonra alınmıştır.

Ekran 8.38 . “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu Düzenle” ekranı.

Seçim gerçekleştirilip düğme tıklandığında, formun “Eski” olarak adlandırılan ayırımında düzeltilmek istenen bilgiler görüntülenir. Kullanıcı yapmak istediği değişiklikleri formun “Yeni” ayırımında seçip “Değiştir” düğmesine basarak gerçekleştirir. “Değiştir” düğmesinin işlettiği VB kodları aşağıdaki gibidir.

```
Private Sub cmdDeğiştir_Click()  
If IsNothing(Me.cmbEğitimYılı) Then  
    MsgBox "Lütfen Eğitim Yılını seçiniz.", vbCritical, gstrAppTitle  
    Exit Sub  
End If  
If IsNothing(Me.CmbDönem) Then  
    MsgBox "Lütfen Dönemi seçiniz.", vbCritical, gstrAppTitle  
    Exit Sub  
End If  
If IsNothing(Me.cmbYarıyıl) Then  
    MsgBox "Lütfen Yarıyılı seçiniz.", vbCritical, gstrAppTitle  
    Exit Sub  
End If  
If IsNothing(Me.cmbDanışman) Then  
    MsgBox "Lütfen Danışmanı seçiniz.", vbCritical, gstrAppTitle  
    Exit Sub  
End If  
Me.cmbEğitimYılı.SetFocus  
Me.EğitimYılı = Me.cmbEğitimYılı.Text  
Me.CmbDönem.SetFocus  
Me.Dönem = Me.CmbDönem.Text  
Me.cmbYarıyıl.SetFocus  
Me.Yarıyıl = Me.cmbYarıyıl.Text  
Me.cmbDanışman.SetFocus
```



```

Me.tblÖğrenciSınıf_Danışman = Me.cmbDanışman
Me.tblÖğrenci_Danışman = Me.cmbDanışman
Me.cmbEğitimYılı.SetFocus
On Error Resume Next
Screen.PreviousControl.SetFocus
If Err <> 0 Then
    Me.cmbEğitimYılı.SetFocus
    Err.Clear
End If
If Not SaveIt() Then Exit Sub
End Sub

```

Yukarıdaki kodlar ile kullanıcının belirlediği yeni değerler “tblÖğrenci” ve “tblÖğrenciSınıf” tablolarının “EğitimYılı”, “Dönem”, “Yarıyıl” ve “Danışman” alanlarındaki eski değerleri ile değiştirilecektir.

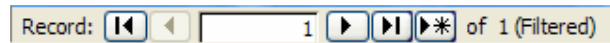
Kullanıcının listeden seçtiği bilginin düzeltilmesini veya değiştirilmesini değil de silinmesini istemesi durumunda “Sil” düğmesine basması yeterlidir. Bu düğme öğrenci için daha önce girilmiş olan eğitim yılı, dönem, yarıyıl ve danışman bilgilerinin silinmesini sağlar. Düğmenin işlettiği VB kodları aşağıdaki gibidir.

```

Private Sub cmdSil_Click()
On Error GoTo Err_cmdSil_Click
    DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
    DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
Exit_cmdSil_Click:
    Exit Sub
Err_cmdSil_Click:
    MsgBox Err.Description
    Resume Exit_cmdSil_Click
On Error Resume Next
Screen.PreviousControl.SetFocus
If Err <> 0 Then
    Me.cmbEğitimYılı.SetFocus
    Err.Clear
End If
If Not SaveIt() Then Exit Sub
End Sub

```

Tümünü Düzenle Düğmesi: Tümünü Düzenle düğmesi “Düzenle” düğmesi ile aynı işleve sahip olup tek farkı “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu Düzenle” ekranını listeden seçilen tek kayıt için değil de listedeki bütün kayıtlar için açmasıdır. Bu ekranda kayıtlara ulaşmak için ekranın altında bulunan düğmelerden yararlanılır (Ekran 8.39).



Ekran 8.39. Kayıt gezinti düğmeleri.

Yeni Düğmesi: Yeni düğmesi tıklandığında Ekran 8.40’da gösterilen “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu Yeni” formu açılacaktır.

Ekran 8.40. “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu Yeni” ekranı.

Bu formda girilmesi istenen bilgiler seçilip “Ekle” düğmesine basılır. Formun kayıt kaynağını oluşturan SQL, “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu Düzenle” formunun kayıt kaynağı SQL’i ile aynıdır. “Ekle” düğmesinin VB kodları ise aşağıdaki gibidir.

```

Private Sub cmdDeğiştir_Click()
Dim cnThisConnect As ADODB.Connection
Dim rcdÖğrenciSınıf As New ADODB.Recordset
Dim i As Integer
Set cnThisConnect = CurrentProject.Connection
rcdÖğrenciSınıf.Open "tblÖğrenciSınıf", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
If IsNothing(Me.cmbEğitimYılı) Then
    MsgBox "Lütfen Eğitim Yılımı seçiniz.", vbCritical, gstrAppTitle
    Exit Sub
End If
If IsNothing(Me.CmbDönem) Then
    MsgBox "Lütfen Dönemi seçiniz.", vbCritical, gstrAppTitle
    Exit Sub
End If
If IsNothing(Me.cmbYarıyıl) Then
    MsgBox "Lütfen Yarıyılı seçiniz.", vbCritical, gstrAppTitle
    Exit Sub
End If
If IsNothing(Me.cmbDanışman) Then
    MsgBox "Lütfen Danışmanı seçiniz.", vbCritical, gstrAppTitle
    Exit Sub
End If
For i = 1 To rcdÖğrenciSınıf.RecordCount
If (rcdÖğrenciSınıf![ÖğrenciID] = Me.ÖğrenciID) And (rcdÖğrenciSınıf![EğitimYılı] =
Me.cmbEğitimYılı) And (rcdÖğrenciSınıf![Dönem] = Me.CmbDönem) And
(rcdÖğrenciSınıf![Yarıyıl] = Me.cmbYarıyıl) Then
    MsgBox "Zaten mevcut olan bir kaydı girmeye çalışıyorsunuz .", vbCritical, gstrAppTitle
    Me.CmbDönem = ""

```

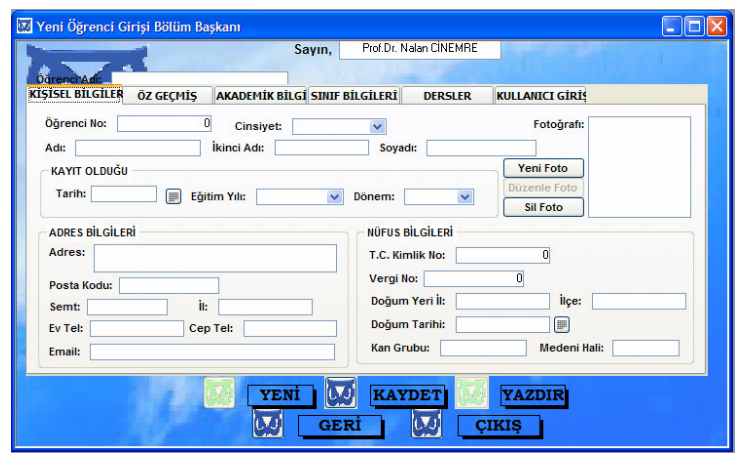
```

Me.cmbEğitimYılı = ""
Me.cmbDanışman = ""
Me.cmbYarıyıl = ""
Exit Sub
Else
Me.EğitimYılı = Me.cmbEğitimYılı
Me.Dönem = Me.CmbDönem
Me.Yarıyıl = Me.cmbYarıyıl
Me.tblÖğrenci_Danışman = Me.cmbDanışman
Me.tblÖğrenciSınıf_Danışman = Me.cmbDanışman
End If
rcdÖğrenciSınıf.MoveNext
Next i
Me.cmdCancel.SetFocus
Me.cmdDeğiştir.Enabled = False
Me.cmdYeni.Enabled = True
On Error Resume Next
Screen.PreviousControl.SetFocus
If Err <> 0 Then
Me.cmbEğitimYılı.SetFocus
Err.Clear
End If
If Not SaveIt() Then Exit Sub
End Sub

```


Bu kodlar ile öğrencilerin daha önceki eğitim yılı, dönem, yarıyıl ve danışmanlık bilgilerinin programda yer aldığı “tblÖğrenciSınıf” tablosunda, kullanıcının eklemek istediği yeni değerler aranır. Eğer girilen değerler daha önceden girilmiş ise kullanıcı "Zaten mevcut olan bir kaydı girmeye çalışıyorsunuz" mesajıyla uyarılır, yoksa ekleme işlemi gerçekleştirilir.

Öğrenci Girişi: Yeni ayırımında yeraln “Öğrenci Girişi” düğmesi, ilk kez kaydı yapılacak lisans öğrencilerinin Baykuş programına girişini sağlamak için kullanılan “Yeni Lisans Öğrenci Girişi” formunu açan düğmedir (Ekran 8.41). Form ilk açıldığında “Yeni” ve “Yazdır” düğmeleri kullanılamaz durumdadırlar.



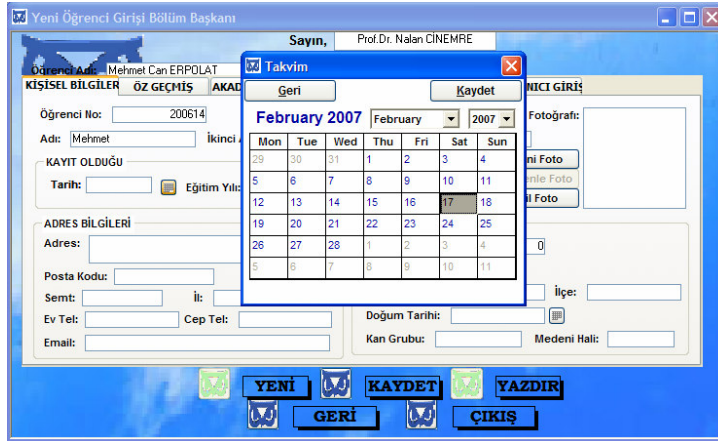
Ekran 8.41. “Yeni Lisans Öğrenci Girişi” ekranı.

Bu form daha önce “Öğretim Elemanı” girişinde anlatılan “Öğrencilerin Kişisel Bilgileri” formunun benzeri olup “Dersler” ve “Kullanıcı Girişi” sekmelerinde farklılık göstermektedir.

Kişisel Bilgiler Sekmesi: Yeni kayıt olan öğrencilerin kişisel bilgilerinin girişinin yapıldığı sekmedir. Bu sekmede “Kayıt Tarihi” ve “Doğum Tarihi” alanlarının yanlarında bulunan  düğmesi takvim düğmesidir. Takvim düğmesine basılınca işletilen VB kodları aşağıdaki gibidir.

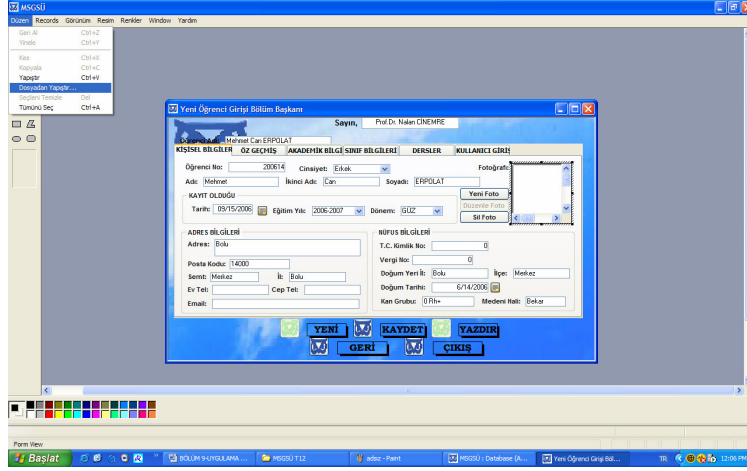
```
Private Sub cmdKayıtTarihi_Click()  
Dim varReturn As Variant  
    Me.KayıtTarihi.SetFocus  
    varReturn = GetDateOCX(Me.KayıtTarihi, True)  
End Sub
```

Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adındaki öğrencinin Baykuş programına ilk kez kaydedilmesi sırasında kişisel bilgileri girilirken takvim düğmesine basıldıktan sonra alınmıştır. Takvimde tarih bilgisi seçilip “Kaydet” düğmesine basılınca takvim kapanır ve seçilen tarih alana yerleşir.



Ekran 8.42. “Kişisel Bilgiler” sekmesi ve takvimin görünüşü.

“Kişisel Bilgiler” sekmesine öğrencinin resmini “Fotoğrafi” alanına yerleştirilebilmek için sekme üzerindeki “Yeni Foto” düğmesi tıklanır. Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adındaki öğrencinin kişisel bilgilerinin girilip “Yeni Foto” düğmesine basılmasından sonra alınmıştır.



Ekran 8.43. “Kişisel Bilgiler” sekmesine fotoğraf ekleyen ekran.

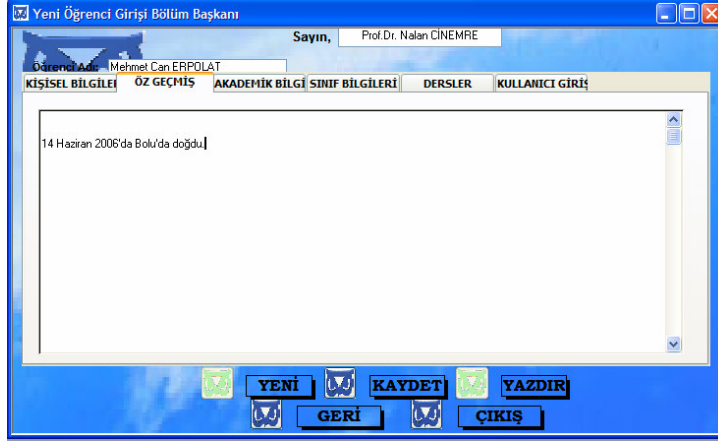
Ekran 8.43’den de görüldüğü gibi, “Yeni Lisans Öğrenci Girişi” formu farklı bir pencerede açılır. Bu pencerede bulunan “Düzen” menüsünden “Dosyadan Yapıştır” seçeneği seçilir. Böylece daha önceden bilgisayara kaydedilmiş olan resmin bulunduğu dosyaya ulaşılır ve bu dosyadan öğrencinin resmi seçilerek forma eklenmesi sağlanır.

Aşağıdaki ekran görüntüsü öğrenciye ait resmin form üzerine yerleştirilmesinden sonra alınmıştır.



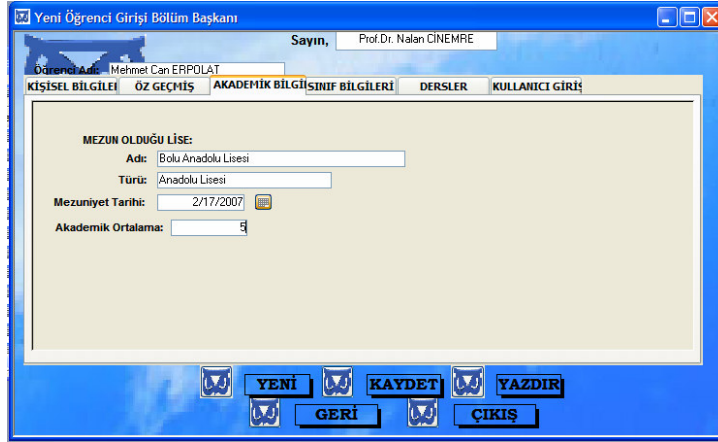
Ekran 8.44. “Kişisel Bilgiler” sekmesine fotoğrafın eklenmiş hali.

Özgeçmiş Sekmesi: Lisans öğrencisinin özgeçmişinin yazıldığı metin niteliğindeki sekmedir (Ekran 8.45). Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adlı öğrencinin özgeçmiş yazıldıktan sonra alınmıştır.



Ekran 8.45. “Özgeçmiş” sekmesi

Akademik Bilgiler Sekmesi: Öğrencinin daha önceki akademik bilgilerinin kayıt edileceği sekmedir. Ekran 8.46’da gösterilen “Akademik Bilgiler” sekmesi, “Mehmet Can ERPOLAT” adlı öğrencinin akademik bilgileri girildikten sonra alınmıştır.



Ekran 8.46. “Akademik Bilgiler” sekmesi

Sınıf Bilgisi Sekmesi: Öğrencinin kayıt olduğu eğitim yılı, dönem, yarıyıl bilgilerinin işlendiği ve kendisine bir danışman hocaların belirlendiği sekmedir (Ekran 8.47).

Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adlı öğrencinin eğitim yılı, dönem, yarıyıl bilgileri girilip kendisine bir danışman hocaların atanmasından sonra alınmıştır.

Yeni Öğrenci Girişi Bölüm Başkanı

Sayın, Prof. Dr. Nalan CİNEMRE

Öğrenci Adı: Mehmet Can ERPOLAT

KİŞİSEL BİLGİLERİ ÖZ GEÇMİŞ AKADEMİK BİLGİ SINIF BİLGİLERİ DERSLER KULLANICI GİRİŞİ

EĞİTİM YILI, DÖNEM, YARIYIL, DANIŞMAN BELİRLEME:

Eğitim Yılı: 2006-2007 Danışman: Prof. Dr. Nalan CİNEMRE

Dönem: GÜZ

Yarıyıl: 1

YENİ KAYDET YAZDIR GERİ ÇIKIŞ

Ekran 8.47. “Akademik Bilgiler” sekmesi

Dersler Sekmesi: Bu sekmede, İstatistik lisans programında okutulan derslerin listesi yer almaktadır. Kullanıcı, kaydı yapılan öğrencinin o dönem hangi dersleri seçeceğini listedeki derslerin karşısındaki “Seç” onay kutucuğunu tıklayıp doldurmak suretiyle gerçekleştirir. Sekme Ekran 8.48’de gösterildiği gibidir.

Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adlı öğrencinin 2006-2007 eğitim yılı güz dönemi 1’inci yarıyılında seçebileceği derslerin “Seç” sütunundaki kutucuğa çek atılmak suretiyle belirlenmesinden sonra alınmıştır.

Yeni Öğrenci Girişi Bölüm Başkanı

Sayın, Prof. Dr. Nalan CİNEMRE

Öğrenci Adı: Mehmet Can ERPOLAT

KİŞİSEL BİLGİLERİ ÖZ GEÇMİŞ AKADEMİK BİLGİ SINIF BİLGİLERİ DERSLER KULLANICI GİRİŞİ

DersID:	DersKodu:	DersAdı:	ÖğrÜyesiAdı:	Seç
1	IST101	İSTATİSTİĞE GİRİŞ I	Yrd. Doç. Dr. Meral YAY	<input checked="" type="checkbox"/>
2	IST103	OLASILIK I	Prof. Dr. Gülay KIROĞLU	<input checked="" type="checkbox"/>
3	IST105	BİLGİ İŞLEM I	Prof. Dr. Aydın ERAR	<input checked="" type="checkbox"/>
4	IST107	İŞLETME I	Yrd. Doç. Dr. Füsün DERİŞ	<input checked="" type="checkbox"/>
5	IST109	İKTİSAT I	Yrd. Doç. Dr. Yeşim REEL	<input checked="" type="checkbox"/>
6	MAT181	ANALİZ I	Yrd. Doç. Dr. Nil KOFDOĞLU	<input checked="" type="checkbox"/>
7	SOS123	SOSYOLOJİ I	Yrd. Doç. Dr. Firdevs GÜMÜŞOĞLU	<input checked="" type="checkbox"/>
8	ING111	TEMEL İNGİLİZCE I	Dkt. Aynur DURANSOY	<input checked="" type="checkbox"/>
9	ALM111	TEMEL ALMANCA I		<input type="checkbox"/>

Record: 14 of 102

YENİ KAYDET YAZDIR GERİ ÇIKIŞ

Ekran 8.48. “Dersler” sekmesi.

“Seç” düğmesinin seçlip doldurulmasıyla işleme giren VB kodları aşağıdaki gibidir.

```
Private Sub Seç_Click()
Dim cnThisConnect As ADODB.Connection
Dim rcdTümDersler As New ADODB.Recordset
If Me.Seç = True Then
Set cnThisConnect = CurrentProject.Connection
```

```

rcdTümDersler.Open "tblÖğrencilerinTümAlmasıGerekenDersleri", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdTümDersler.AddNew
rcdTümDersler![DersID] = Me.DersID
rcdTümDersler![ÖğrenciID] = Forms!frmYeniÖğrenciGirişiBölümBaşkani.ÖğrenciID
rcdTümDersler.Update
End If
End Sub

```

Kodlardan da anlaşılacağı gibi, seçimin gerçekleştirilmesiyle Baykuş programında öğrencilerin aldıkları dersleri içeren "tblÖğrencilerinTümAlmasıGerekenDersleri" tablosuna yeni öğrencinin bilgileri işlenmiş olur.

Kullanıcı Girişi Sekmesi: Bu sekme Ekran 8.49'daki gibidir. Kaydı yapılan öğrenciye Baykuş programını kullanabilmesi için bir kullanıcı adı ile şifrenin verilmesini sağlar.

Ekran 8.49. "Kullanıcı Girişi" sekmesi.

Kullanıcı adı ve şifre girildikten sonra "Onay" düğmesine tıklanır. Böylece çalıştırılan VB kodları aşağıda gösterildiği gibidir.

```

Private Sub cmdOnay_Click()
Dim cnThisConnect As ADODB.Connection
Dim rcdÖğrenci As New ADODB.Recordset
Dim rcdKullanıcılarÖğrenciler As New ADODB.Recordset
Dim i As Integer
Dim j As Integer
Dim k As Integer
Dim L As Integer
Dim m As Integer
Set cnThisConnect = CurrentProject.Connection
rcdKullanıcılarÖğrenciler.Open "tblKullanıcılarÖğrenciler", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdÖğrenci.Open "tblÖğrenci", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable

```



```

If Me.txtKullanıcıAdı0 = "" Then
    Me.txtKullanıcıAdı0 = ""
    Me.txtŞifre0 = ""
    Me.txtKullanıcıAdı0.SetFocus
    MsgBox "Lütfen bir Kullanıcı Adı giriniz. ", vbCritical, gstrAppTitle
    Exit Sub
End If
If Me.txtŞifre0 = "" Then
    Me.txtKullanıcıAdı0 = ""
    Me.txtŞifre0 = ""
    Me.txtKullanıcıAdı0.SetFocus
    MsgBox "Lütfen bir Şifre giriniz. ", vbCritical, gstrAppTitle
    Exit Sub
End If
If Me.txtKullanıcıAdı0 = "" And Me.txtŞifre0 = "" Then
    Me.txtKullanıcıAdı0 = ""
    Me.txtŞifre0 = ""
    Me.txtKullanıcıAdı0.SetFocus
    MsgBox "Lütfen bir Kullanıcı Adı ve Şifre giriniz. ", vbCritical, gstrAppTitle
    Exit Sub
End If
For ii = 1 To rcdKullanıcılarÖğrenciler.RecordCount
    If rcdKullanıcılarÖğrenciler![KullanıcıAdı] = Me.txtKullanıcıAdı0 Then
        Me.txtKullanıcıAdı0 = ""
        Me.txtŞifre0 = ""
        Me.txtKullanıcıAdı0.SetFocus
        MsgBox "Lütfen farklı bir Kullanıcı Adı giriniz. ", vbCritical, gstrAppTitle
        Exit Sub
    End If
    rcdKullanıcılarÖğrenciler.MoveNext
Next ii
rcdKullanıcılarÖğrenciler.MoveFirst
For ii = 1 To rcdKullanıcılarÖğrenciler.RecordCount
    If rcdKullanıcılarÖğrenciler![Şifre] = Me.txtŞifre0 Then
        Me.txtKullanıcıAdı0 = ""
        Me.txtŞifre0 = ""
        Me.txtKullanıcıAdı0.SetFocus
        MsgBox "Lütfen farklı bir Şifre giriniz. ", vbCritical, gstrAppTitle
        Exit Sub
    End If
    rcdKullanıcılarÖğrenciler.MoveNext
Next ii
rcdKullanıcılarÖğrenciler.MoveFirst
For ii = 1 To rcdKullanıcılarÖğrenciler.RecordCount
    If (rcdKullanıcılarÖğrenciler![KullanıcıAdı] = Me.txtKullanıcıAdı0) And
(rcdKullanıcılarÖğrenciler![Şifre] = Me.txtŞifre0) Then
        Me.txtKullanıcıAdı0 = ""
        Me.txtŞifre0 = ""
        Me.txtKullanıcıAdı0.SetFocus
        MsgBox "Lütfen farklı bir Kullanıcı Adı ve Şifre giriniz. ", vbCritical, gstrAppTitle
        Exit Sub
    End If
    rcdKullanıcılarÖğrenciler.MoveNext
Next ii
rcdKullanıcılarÖğrenciler.MoveFirst
rcdKullanıcılarÖğrenciler.AddNew
rcdKullanıcılarÖğrenciler![ÖğrenciID] = Forms!frmYeniÖğrenciGirişiBölümBaşkanı.ÖğrenciID
rcdKullanıcılarÖğrenciler![KullanıcıAdı] = Me.txtKullanıcıAdı0
rcdKullanıcılarÖğrenciler![Şifre] = Me.txtŞifre0
rcdKullanıcılarÖğrenciler.Update

```

```
rcdÖğrenci.MoveLast  
rcdÖğrenci![KullanıcıAdı] = Me.txtKullanıcıAdı0  
rcdÖğrenci.Update  
Me.txtKullanıcıAdı0.SetFocus  
Me.cmdOnay.Enabled = False  
End Sub
```

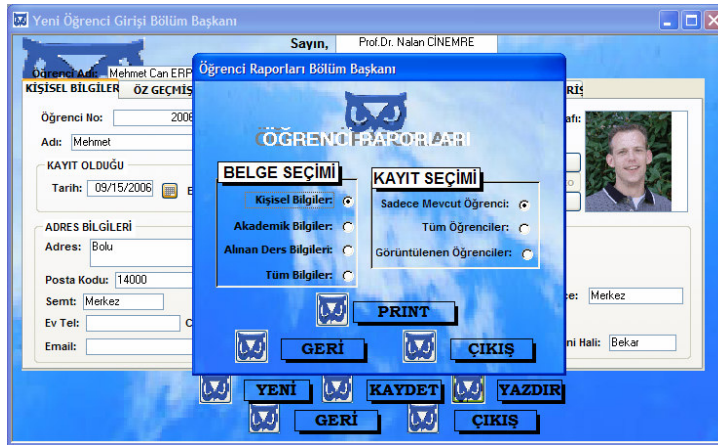
Bu kodlar yardımıyla öğrenci için belirlenen kullanıcı adı ve şifrenin daha önce başka bir öğrenciye verilip verilmediği sınıranır. Bu sınıma “tblKullanıcılarÖğrenciler” tablosu üzerinde yapılır. Eğer benzer bir kayda rastlanmazsa ekleme işlemi gerçekleştirilir.

Kaydet Düğmesi: “Yeni Öğrenci Girişi” formunda tüm bilgiler girildikten sonra “Kaydet” düğmesine basılmalıdır. Aksi takdirde girilen bilgiler programa kaydedilmez, geçici olur.

Yeni Düğmesi: Yeni düğmesi, “Yeni Öğrenci Girişi” formunu yeni bir öğrencinin daha girişinin gerçekleştirilmesi için temizlenmesini sağlar.

Yazdır Düğmesi: “Yazdır” düğmesi girişi tamamlanan öğrenciye ilişkin sekmelerdeki tüm bilgilerin çıktılarını almak için Ekran 8.50’de gösterilen “Öğrenci Raporları” print ekranını açar.

Aşağıdaki ekran görüntüsü “Mehmet Can ERPOLAT” adlı öğrencinin tüm bilgilerinin girilmesi, kaydının yapılmasının ardından “Yazdır” düğmesine basıldıktan sonra alınmıştır.



Ekran 8.50. “Öğrenci Raporları” print ekranı.

“Öğrenci Raporları” print ekranı “Belge Seçimi” ve “Kayıt Seçimi” olmak üzere iki ayırımdan oluşmaktadır. “Belge Seçimi” ayırımında “Kişisel Bilgiler”, “Akademik Bilgiler”, “Alınan Ders Bilgileri” ve “Tüm Bilgiler” Seçenekleri yer almaktadır. “Kayıt Seçimi” ayırımında ise “Sadece Mevcut Öğrenci”, “Tüm Öğrenciler” ve “Görüntülenen Öğrenciler” seçenekleri bulunmaktadır. Bu ayırımdaki “Tüm Öğrenciler” ve “Görüntülenen Öğrenciler” seçenekleri daha çok öğrencilerin kişisel bilgilerinin görüntülediği diğer formlarda kullanılmaktadır.

Aşağıdaki ekran görüntüsü belge seçimi olarak “Kişisel Bilgiler” kayıt seçimi olarak da “Sadece Mevcut Öğrenci” seçenekleri seçilip “Print” düğmesine tıklandıktan sonra alınmıştır.

Öğrencilerin Kişisel Bilgileri

İMAM SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ
FEN EDEBİYAT FAKÜLTESİ
İSTATİSTİK BÖLÜMÜ

ÖĞRENCİNİN KİŞİSEL BİLGİLERİ

Sıradan Adı: BEPOLAT
Adı: Mehmet
İkinci Adı: Can
Öğrenci No: 200614
Kayıt Tarihi: 09/18/2006
Kayıt Eğitim Yılı: 2005-2007
Kayıt Durumu: GİZ

ADRES BİLGİLERİ

Adres: Bolu
Posta Kodu: 14000 Semt: Merkez İl: Bolu
Ev Tel: Cep Tel:

Diğer Bilgiler

T.C. Kimlik No: Vergi No: 0
Doğum Tarihi: 6/14/2006
Doğum Yeri: İl: Bolu İlçe: Merkez
Medeni Hali: Bekar
Kan Grubu: 0 Rh+

İstanbul, 7 Ekim 2007 Page 1 of 1

Ekran 8.51. “Kişisel Bilgiler” raporu.

Aşağıdaki ekran görüntüsü belge seçimi olarak “Akademik Bilgiler” kayıt seçimi olarak da “Sadece Mevcut Öğrenci” seçenekleri seçilip “Print” düğmesine tıklandıktan sonra alınmıştır.

Öğrenci Raporları

MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ
FEN EDEBİYAT FAKÜLTESİ
İSTATİSTİK BÖLÜMÜ

ÖĞRENCİNİN AKADEMİK BİLGİLERİ

Soyadı: EŞPOLAT
Adı: Mehmet
Sınıfı/Adı: Can
Öğrenci No: 200614

LİSE

Ad:
Yarıyıl:
Yarıyıl Başlangıç Tarihi:

KAYDET

Ekran 8.52. “Akademik Bilgiler” raporu.

Aşağıdaki ekran görüntüsü belge seçimi olarak “Alınan Ders Bilgileri” kayıt seçimi olarak da “Sadece Mevcut Öğrenci” seçenekleri seçilip “Print” düğmesine tıklandıktan sonra alınmıştır.

Öğrenci Raporları

MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ
FEN EDEBİYAT FAKÜLTESİ
İSTATİSTİK BÖLÜMÜ

ÖĞRENCİNİN ALMASI GEREKEN DERSLERİ

Soyadı: EŞPOLAT
Adı: Mehmet
Sınıfı/Adı: Can
Öğrenci No: 200614

Ders No	Ders Adı	Ders Türü	Ders Saati	Ders Öğretmeni	Ders Durumu
000111	FENOLİT	2	2	Öz	1
000112	STATİSTİK GİRİŞİ	2	2	Öz	1
000113	STATİSTİK	2	2	Öz	1
000114	STATİSTİK	2	2	Öz	1
000115	STATİSTİK	2	2	Öz	1
000116	STATİSTİK	2	2	Öz	1
000117	STATİSTİK	2	2	Öz	1
000118	STATİSTİK	2	2	Öz	1
000119	STATİSTİK	2	2	Öz	1
000120	STATİSTİK	2	2	Öz	1
000121	STATİSTİK	2	2	Öz	1
000122	STATİSTİK	2	2	Öz	1
000123	STATİSTİK	2	2	Öz	1
000124	STATİSTİK	2	2	Öz	1
000125	STATİSTİK	2	2	Öz	1
000126	STATİSTİK	2	2	Öz	1
000127	STATİSTİK	2	2	Öz	1
000128	STATİSTİK	2	2	Öz	1
000129	STATİSTİK	2	2	Öz	1
000130	STATİSTİK	2	2	Öz	1
000131	STATİSTİK	2	2	Öz	1
000132	STATİSTİK	2	2	Öz	1
000133	STATİSTİK	2	2	Öz	1
000134	STATİSTİK	2	2	Öz	1
000135	STATİSTİK	2	2	Öz	1
000136	STATİSTİK	2	2	Öz	1
000137	STATİSTİK	2	2	Öz	1
000138	STATİSTİK	2	2	Öz	1
000139	STATİSTİK	2	2	Öz	1
000140	STATİSTİK	2	2	Öz	1
000141	STATİSTİK	2	2	Öz	1
000142	STATİSTİK	2	2	Öz	1
000143	STATİSTİK	2	2	Öz	1
000144	STATİSTİK	2	2	Öz	1
000145	STATİSTİK	2	2	Öz	1
000146	STATİSTİK	2	2	Öz	1
000147	STATİSTİK	2	2	Öz	1
000148	STATİSTİK	2	2	Öz	1
000149	STATİSTİK	2	2	Öz	1
000150	STATİSTİK	2	2	Öz	1
000151	STATİSTİK	2	2	Öz	1
000152	STATİSTİK	2	2	Öz	1
000153	STATİSTİK	2	2	Öz	1
000154	STATİSTİK	2	2	Öz	1
000155	STATİSTİK	2	2	Öz	1
000156	STATİSTİK	2	2	Öz	1
000157	STATİSTİK	2	2	Öz	1
000158	STATİSTİK	2	2	Öz	1
000159	STATİSTİK	2	2	Öz	1
000160	STATİSTİK	2	2	Öz	1
000161	STATİSTİK	2	2	Öz	1
000162	STATİSTİK	2	2	Öz	1
000163	STATİSTİK	2	2	Öz	1
000164	STATİSTİK	2	2	Öz	1
000165	STATİSTİK	2	2	Öz	1
000166	STATİSTİK	2	2	Öz	1
000167	STATİSTİK	2	2	Öz	1
000168	STATİSTİK	2	2	Öz	1
000169	STATİSTİK	2	2	Öz	1
000170	STATİSTİK	2	2	Öz	1
000171	STATİSTİK	2	2	Öz	1
000172	STATİSTİK	2	2	Öz	1
000173	STATİSTİK	2	2	Öz	1
000174	STATİSTİK	2	2	Öz	1
000175	STATİSTİK	2	2	Öz	1
000176	STATİSTİK	2	2	Öz	1
000177	STATİSTİK	2	2	Öz	1
000178	STATİSTİK	2	2	Öz	1
000179	STATİSTİK	2	2	Öz	1
000180	STATİSTİK	2	2	Öz	1
000181	STATİSTİK	2	2	Öz	1
000182	STATİSTİK	2	2	Öz	1
000183	STATİSTİK	2	2	Öz	1
000184	STATİSTİK	2	2	Öz	1
000185	STATİSTİK	2	2	Öz	1
000186	STATİSTİK	2	2	Öz	1
000187	STATİSTİK	2	2	Öz	1
000188	STATİSTİK	2	2	Öz	1
000189	STATİSTİK	2	2	Öz	1
000190	STATİSTİK	2	2	Öz	1
000191	STATİSTİK	2	2	Öz	1
000192	STATİSTİK	2	2	Öz	1
000193	STATİSTİK	2	2	Öz	1
000194	STATİSTİK	2	2	Öz	1
000195	STATİSTİK	2	2	Öz	1
000196	STATİSTİK	2	2	Öz	1
000197	STATİSTİK	2	2	Öz	1
000198	STATİSTİK	2	2	Öz	1
000199	STATİSTİK	2	2	Öz	1
000200	STATİSTİK	2	2	Öz	1

KAYDET

Ekran 8.53. “Akademik Bilgiler” raporu.

Bilgiler Ayırımı: Bilgiler ayırımı, “Kişisel Bilgiler” ve “Bölüm Bilgileri” düğmelerinden oluşmaktadır. Bu düğmeler, “Öğretim Elemanı Girişi”ndeki

“Danışmanlık” düğmesi altındaki öğrencilere ilişkin “Kişisel Bilgiler” ve “Bölüm Bilgileri” düğmeleriyle aynı işleve sahip olduklarından burada sözü edilmeyecektir.

İstatistik Düğmesi: “İstatistik” düğmesi, Ekran 8.26’daki İstatistik düğmesi ile aynı işleve sahiptir.

Ara Düğmesi: “Ara” düğmesi Ekran 8.27’de değinilen “Ara” düğmesiyle benzer işleve sahiptir.

8.4.4.1.2. Lisans Üstü Öğrencileri Ayırımı

“Lisans ve Lisans Üstü Öğrencileri Main” ekranında yer alan “Lisans Üstü Öğrencileri” ayırımı, “Lisans Öğrencileri” ayırımında olduğu gibi benzer işlevleri olan ekran ve düğmelere sahiptir. Farklı olarak bu ayırımı yapılan işlemlerin tümünün Baykuş programında lisans üstü öğrencilerin bilgilerini saklamak için kullanılan “tblYLOğrencileri”, “tblYLOğrenciSınıf”, “tblYLOğrencilerinAlması GerekenTümdersler” adlarındaki tablolar üzerinde gerçekleştiriliyor olmasıdır.

8.4.4.2. Dersler Düğmesi

“Bölüm Başkanı Main” formundaki “Dersler” düğmesine basıldığında Ekran 8.54’deki “Lisans ve Lisans Üstü Dersleri Main” adındaki form açılır.



Ekran 8.54. “Lisans ve Lisans Üstü Dersleri Main” ekranı.

Bu formda “Lisans Dersleri” ve “Lisans Üstü Dersleri” olmak üzere iki ayırım ve her iki ayırımın da sahip olduğu “Dersler” düğmesi yer almaktadır.

Lisans Dersleri ayırımındaki “Dersler” düğmesine tıkladığında “Dönem Seçimi Dersler” adındaki görüntülenmesi istenen derslere ilişkin eğitim yılı, dönem ve yarıyıl değerlerinin belirlendiği “Dönem Seçimi” ekranı açılır. Kullanımı, daha önce açıklanan dönem seçimi ekranlarıyla aynı olan bu ekranda “Göster” düğmesine basıldığında Ekran 8.55’de gösterilen “Ders Listesi” ekranına ulaşılır.

Aşağıdaki ekran görüntüsü “Dönem Seçimi” ekranında eğitim yılı olarak 2006-2007, dönem olarak “GÜZ” ve yarıyıl olarak ise “1” değeri seçilip “Göster” düğmesine tıklandıktan sonra alınmıştır.



Ekran 8.55. “Ders Listesi” ekranı.

“Ders Listesi” ekranında “Yeni” ayırımında “Yeni Ders Ekle” ve “Yeni Dönem Ekle” düğmeleri, “Listeler” ayırımında “Öğrenci Listesi” ve “Sınav Listesi” düğmeleri, “Bilgi Yenileme” ayırımında ise “Düzenle” ve “Tümünü Düzenle” düğmeleri yer almaktadır. “Tümünü Düzenle” düğmesi dışındaki düğmelerin işlevlerini yerine getirebilmesi için yukarıdaki listede bulunan derslerden birinin seçilmiş olması gerekir. Ekranda ayrıca “Ara” ve “Dönem Dersleri” düğmeleri de yer almaktadır.

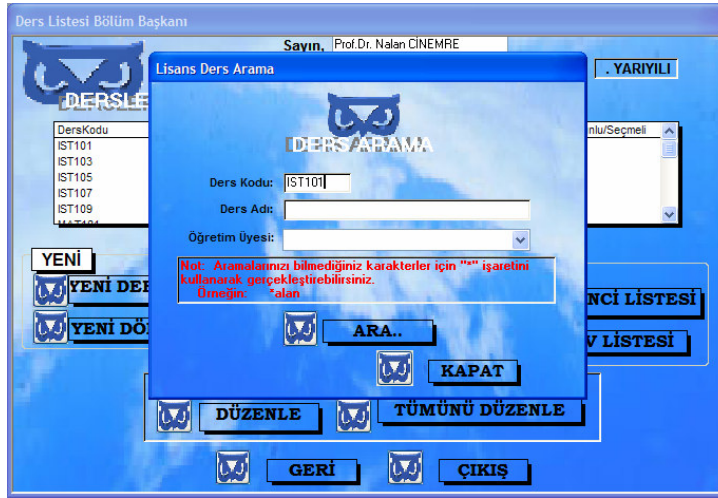
8.4.4.2.1. Dönem Dersleri Düğmesi

“Ders Listesi” ekranında, “Dönem Seçimi Dersler” ekranı ile belirlenen eğitim yılı, dönem, yarıyıl kriterlerini sağlayan derslerin listesi bulunur. Bu listede geçerli eğitim yılı ve döneme ilişkin bütün derslerin listelenmesi isteniyorsa “Dönem Dersleri” düğmesine tıklanır.

8.4.4.2.2. Ara Düğmesi

Bu düğme Ekran 8.56’da gösterilen “Ders Arama” ekranını açar. Bu arama ekranı kullanıcının girdiği ders kodu veya ders adı veya öğretim üyesi adı kriterlerine göre ders arama işlemlerini gerçekleştirilir. Arama “tblDersler” tablosu üzerinde gerçekleşir.

Aşağıdaki ekran görüntüsü, “Ders Arama” ekranında “Ders Kodu” alanına “IST101” değeri girildikten sonra alınmıştır.



Ekran 8.56. “Lisans Ders Arama” ekranı.

“Ara” düğmesine ilişkin VB kodları aşağıdaki gibidir.

```
Private Sub cmdSearch_Click()  
Dim varWhere As Variant  
varWhere = Null  
If Not IsNothing(Me.txtDersKodu) Then  
varWhere = "[DersKodu] LIKE '" & Me.txtDersKodu & "*"'  
End If  
If Not IsNothing(Me.txtDersAdı) Then  
varWhere = (varWhere + " AND ") & "[DersAdı] LIKE '" & Me.txtDersAdı & "*"'  
End If  
If Not IsNothing(Me.CmbÖğretimÜyesi) Then  
varWhere = (varWhere + " AND ") & "[ÖğrÜyesiAdı]= " & Me.CmbÖğretimÜyesi  
End If  
If IsNothing(varWhere) Then  
MsgBox "Aramanın gerçekleşebilmesi için en az bir alana değer girmelisiniz.", vbInformation,  
gstrAppTitle  
Exit Sub  
End If  
If IsNothing(DLookup("DersKodu", "tblDersler", varWhere)) Then  
MsgBox "Girdiğiniz kriterlere uyan bir ders bulunamadı.", vbInformation, gstrAppTitle  
Exit Sub  
End If
```

```
DoCmd.OpenForm f, WhereCondition:=varWhere
DoCmd.Close acForm, ff
DoCmd.Close acForm, Me.Name
End Sub
```

Arama işlemi sonucunda eğer girilen kriterlere uyan ders bulunursa, bu derse ilişkin bilgiler Kesim 8.4.4.2.5.'de değinilen “Ders Düzenle” ekranında görüntülenir (Ekran 8.59).

8.4.4.2.3. Yeni Ayırımı

Yeni Ders Ekle: Bu düğme “Ders Ekle” formunu açar (Ekran 8.57). Bu form “tblDersler” tablosuna, dolayısıyla Baykuş programına yeni bir lisans dersinin girilmesini sağlayan formdur. Form üzerinde yeni giriş yapıldıktan sonra “Kaydet” düğmesine tıklanıp kayıt işlemi gerçekleştirilir.

Ekran 8.57. “Ders Ekle” ekranı.

Yeni Dönem Ekle: Bölüm başkanı “Ders Listesi” ekranında listelenen derslerden birini seçip “Yeni Dönem” düğmesine tıkladığında Ekran 8.58’de gösterilen “Derslerin Eğitim Yılı, Dönem, Yarıyıl Durumu” ekranına ulaşır.

Aşağıdaki ekran görüntüsü “Ders Listesi” ekranında bulunan listeden “İSTATİSTİĞE GİRİŞ I” dersi seçilip “Yeni Dönem” düğmesine basıldıktan sonra alınmıştır.

Derslerin Dönem Durumları Bölüm Başkanı

Sayın, Prof.Dr. Nalan CİNEMRE

2006-2007 EĞİTİM YILI GÜZ DÖNEMİ 1 . YARIYILI

Ders Kodu: IST101

Ders Adı: İSTATİSTİĞE GİRİŞ I

Öğretim Üyesi Adı: Yrd. Doç. Dr. Meral YAY

EĞİTİM YILI, DÖNEM, YARIYIL

EğitimYılı	Dönem	YarıYıl	tblDerslerSınıf.Öğr.ÜyesiAdı
2006-2007	Güz	1	Yrd. Doç. Dr. Meral YAY

Birden fazla ders seçimi yapmak için Ctrl veya Shift tuşunu kullanınız.

DÜZENLE TÜMÜNÜ DÜZENLE YENİ

GERİ ÇIKIŞ

Ekran 8.58. “Derslerin Eğitim Yılı, Dönem, Yarıyıl Durumu” ekranı.

“Derslerin Eğitim Yılı, Dönem, Yarıyıl Durumu” adındaki bu ekranda İstatistik Bölümü’nde okutulan derslerin mevcut döneme kadar olan tüm eğitim yılları, dönemleri ve yarıyıllarının listesi yer alır. Ekran üzerinde “Düzenle”, “Tümünü Düzenle” ve “Yeni” olmak üzere üç düğme bulunmaktadır. Bu düğmelerin işlevleri daha önce sözü edilen “Öğrencilerin Eğitim Yılı, Dönem, Yarıyıl ve Danışmanlık Durumu” adındaki formda değinilen düğmelerle aynıdır. Farklı olarak işlemler dersleri ilgilendirdiğinden “tblÖğrenci” tablosu yerine “tblDersler” ve “tblÖğrenciSınıf” tablosu yerine ise “tblDerslerSınıf” tablosunun kullanılmasıdır.

8.4.4.2.4. Listeler Ayırımı

Öğrenci Listesi Düğmesi: Bu düğme seçilen derse ait öğrencilerin listesini veren “Öğrenci Listesi” formunu açar. Formun kullanımı “Öğretim Elemanı” girişindeki “Ders Listesi” ekranında yer alan “Öğrenci Listesi” düğmesiyle aynıdır.

Sınav Listesi Düğmesi: Bu düğme seçilen derse ait öğrencilerin sınav sonuçlarının girilmesini ve hesaplanmasını sağlayan “Sınav Listesi” formunu açar. Formun kullanımı daha önce değinilen “Öğretim Elemanı” girişindeki “Sınav Listesi” ekranında yer alan “Sınav Listesi” düğmesiyle aynıdır.

8.4.4.2.5. Bilgi Yenileme Ayırımı

Düzenle Düğmesi: Bu düğme listedeki derslerden seçili bir tanesine ait bilgilerin düzeltilmesini veya tamamen silinmesini sağlayan Ekran 8.59'daki “Ders Düzenle” formunu açar.

Aşağıdaki ekran görüntüsü “Ders Listesi” ekranındaki listeden “İSTATİSTİĞE GİRİŞ I” dersinin seçilip “Düzenle” düğmesine basılmasından sonra alınmıştır.

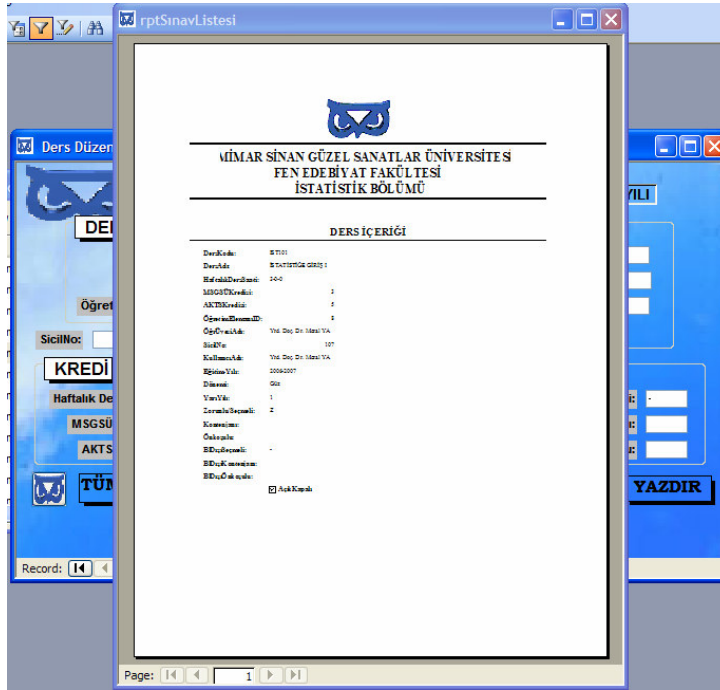
Ekran 8.59. “Ders Düzenle” ekranı.

“Ders Listesi” ekranındaki “Tümünü Düzenle” düğmesi ile aynı işleve sahip olan “Tümünü Düzenle” düğmesi, kullanıcının bu ekranda listelenmekte olan tüm kayıtlara bir seferde ulaşabilmesini sağlar. “Ekle Yeni” düğmesi de “Ders Listesi” ekranındaki “Yeni Ders Ekle” düğmesi ile aynı işleve sahip olup, Baykuş programına yeni bir dersin eklenmesini sağlayan düğmedir. Ekranda görülen alanlarda gerekli değişiklikler yapıldıktan sonra “Kaydet” düğmesine tıklanır.

“Yazdır” düğmesi Ekran 8.60'daki “Ders İçeriği Print” ekranını açar. Bu ekranda “Ders İçeriği” adlı tek bir seçenek olduğundan bu seçenek seçilip “Print” düğmesine basıldığında Ekran 8.61'da gösterilen “Ders İçeriği Raporu” ile karşılaşılır.



Ekran 8.60. “Ders İçeriği Print” ekranı.



Ekran 8.61. “Ders İçeriği Raporu” ekranı.

Tümünü Düzenle Düğmesi: Bu düğme “Ders Listesi” ekranındaki listede sıralanan tüm derslerin bilgilerini düzeltmek veya tamamen silmek amacıyla Ekran 8.59’da gösterilen “Ders Düzenle” formunda açar. Formun kullanımı “Düzenle” düğmesiyle aynıdır.

8.4.4.3. Öğretim Elemanı Düğmesi

Öğretim Elemanı düğmesi, Ekran 8.62’de gösterilen “Öğretim Elemanları” ekranını açar.

Formun kayıt kaynağını oluşturan SQL ifadesi “tblÖğretimElemanları” üzerinde işlem görüp aşağıdaki gibidir.

```
SELECT tblÖğretimElemanları.SicilNo, tblÖğretimElemanları.Soyad, tblÖğretimElemanları.Ad,
tblÖğretimElemanları.İkinciAd, tblÖğretimElemanları.KullanıcıAdı, tblÖğretimElemanları.Ünvanı,
tblÖğretimElemanları.AnabilimDalı, tblÖğretimElemanları.GöreveBaşlamaTarihi,
tblÖğretimElemanları.ÖğretimElemanıID, tblÖğretimElemanları.Adres,
tblÖğretimElemanları.PostaKodu, tblÖğretimElemanları.Semt, tblÖğretimElemanları.İl,
tblÖğretimElemanları.EvTel, tblÖğretimElemanları.CepTel, tblÖğretimElemanları.Email,
tblÖğretimElemanları.TCKimlikNo, tblÖğretimElemanları.VergiNo,
tblÖğretimElemanları.DoğumTarihi, tblÖğretimElemanları.DoğumYeriİl,
tblÖğretimElemanları.DoğumYeriİlçe, tblÖğretimElemanları.MedeniHali,
tblÖğretimElemanları.KanGrubu, tblÖğretimElemanları.Foto,
tblÖğretimElemanları.ÖğretimElemanıID
FROM tblÖğretimElemanları
ORDER BY tblÖğretimElemanları.Soyad, tblÖğretimElemanları.Ad;
```

“Öğretim Elemanları” ekranı Baykuş programında kayıtlı olan bütün öğretim elemanlarının listesini verir.



Ekran 8.62. “Öğretim Elemanları” ekranı.

“Öğretim Elemanları” formu üzerinde “Yeni” ve “Bilgiler” ayırımı bulunmaktadır. Yeni ayırımı “Yeni Öğretim Elemanı Girişi” düğmesini, bilgiler düğmesi ise “Öğretim Elemanlarının Kişisel Bilgileri” düğmesini içermektedir.

8.4.4.3.1. Yeni Öğretim Elemanı Girişi Düğmesi

“Yeni Öğretim Elemanı Girişi” formunu açan düğme olup İstatistik Bölümü’ne yeni alınan veya bölüme dışarıdan ders vermeye gelen öğretim elemanların girişini yapmak amacıyla kullanılmaktadır. Ekran 8.63’de gösterilen “Öğretim Elemanlarının Kişisel Bilgileri” ekranıyla benzer görüntüye sahip olup, farklı olarak “Yeni Öğretim

Elemanı Girişi” formunda “Lisans Dersleri” ve “Lisans Üstü Dersleri” sekmelerinin bulunmamalarıdır.

8.4.4.3.2. Kişisel Bilgiler Düğmesi

Kişisel Bilgiler Düğmesi, “Öğretim Elemanları” ekranında yeralan listeden bir öğretim elemanı seçimi yapıldıktan sonra işlevlik kazanır. Seçimi gerçekleştirilip düğmenin tıklanmasının ardından “Öğretim Elemanının Kişisel Bilgileri” ekranıyla karşılaşılır (Ekran 8.63).

Aşağıdaki ekran görüntüsü “Öğretim Elemanları” ekranındaki listeden “Prof. Dr. Nalan CİNEMRE” seçilip “Kişisel Bilgiler Düğmesi” tıklandıktan sonra alınmıştır.

Ekran 8.63. “Öğretim Elemanının Kişisel Bilgileri” ekranı.

Ekran üzerinde “Kişisel Bilgiler”, “Özgeçmiş”, Akademik Bilgiler”, “Yayınlar”, “Lisans Dersleri”, “Lisans Üstü Dersleri” ve “Kullanıcı Girişi” sekmeleri yer almaktadır.

Kişisel Bilgiler Sekmesi: Bu sekme öğretim elemanının kişisel ve nüfus bilgilerinin görüntülenip üzerinde değişiklik yapılmasına olanak verdiği sekmedir.

Özgeçmiş Sekmesi: Öğretim elemanının özgeçmişini içeren sekmedir.

Akademik Bilgiler Sekmesi: Öğretim elemanının akademik bilgilerinin yer aldığı sekmedir.

Lisans Dersleri Sekmesi: Öğretim elemanının İstatistik Bölümü’nde verdiği lisans derslerinin listelendiği sekmedir. Aşağıdaki ekran görüntüsü seçilen öğretim elemanının İstatistik Bölümü’nde verdiği lisans derslerini sergileyen ekran görüntüsüdür.

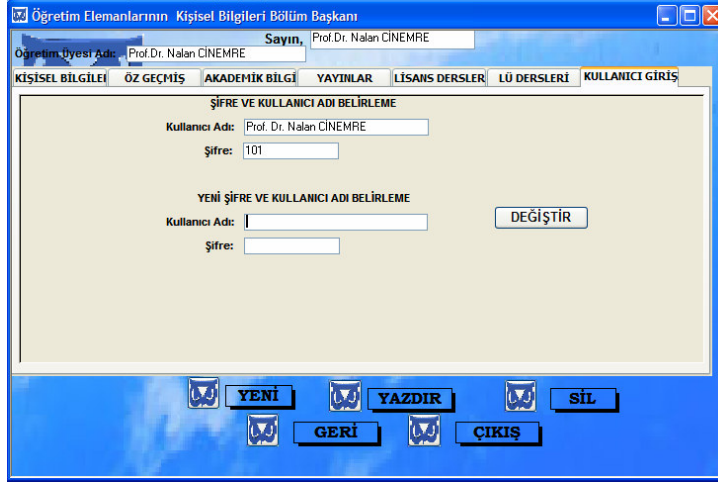
KODU	ADI	KREDİSİ	YARIYILI	DÖNEMİ	Z / S
IST203	DOĞRUSAL PROGRAMLAMA I	2	3	Güz	Z
IST204	DOĞRUSAL PROGRAMLAMA II	2	4	Bahar	Z
IST303	DOĞRUSAL MODELLER	2	5	Güz	S
IST315	YÖNEYLEM ARAŞTIRMASI I	3	5	Güz	Z
IST341	LINEER PROGRAMLAMA I	2	5	Güz	-
IST381	ÇOK AMAÇLI PROGRAMLAMA	2	5	Güz	S
IST318	YÖNEYLEM ARAŞTIRMASI II	3	6	Bahar	Z
IST342	LINEER PROGRAMLAMA II	2	6	Bahar	-
IST405	DENEY TASARIMI I	2	7	Güz	Z
IST441	YÖNEYLEM ARAŞTIRMASI I	2	7	Güz	-
IST475	KARAR KURAMI	2	7	Güz	S
IST477	SİSTEM ANALİZİ	2	7	Güz	S
IST481	BULANIK PROGRAMLAMA	2	7	Güz	S
IST406	DENEY TASARIMI II	2	8	Bahar	Z
IST418	PAKET PROGRAMLAMA	2	8	Bahar	Z

Ekran 8.64. “Lisans Desleri” sekmesi.

Lisans Üstü Dersleri Sekmesi: Öğretim elemanının İstatistik lisans üstü programında verdiği derslerinin listelendiği sekmedir. Görüntü olarak Ekran 8.64’de gösterilen “Lisans Dersleri” sekmesine benzemektedir.

Kullanıcı Girişi Sekmesi: Öğretim elemanının Baykuş programına ilk kaydı yapılırken kendisine verilen “Kullanıcı Adı” ve “Şifre” bilgilerinin yer aldığı, bu bilgileri yenileriyle değiştirmeye olanak veren sekmedir.

Aşağıdaki ekran görüntüsü “Prof. Dr. Nalan CİNEMRE”nin kişisel bilgileri görüntülenirken “Kullanıcı Girişi” sekmesine tıklandıktan sonra alınmıştır.



Ekran 8.65. “Kullanıcı Girişi” sekmesi.

Eğer kullanılmakta olan kullanıcı adı ve şifre bilgileri yenileriyle değiştirilmek isteniyorsa “Yeni Şifre ve Kullanıcı Adı Belirleme” kısmında yeni değerler girilip “Değiştir” düğmesine basılır. Böylece öğretim elemanının Baykuş programını kullanabilmesi için kendisine verilmiş olan “Kullanıcı Adı” ve “Şifre” bilgileri yenileriyle değiştirilmiş olur.

“Değiştir” düğmesinin işlettiği VB kodları aşağıdaki gibidir.

```

Private Sub cmdDeğiştir_Click()
Dim cnThisConnect As ADODB.Connection
Dim rcdÖğretimElemanları As New ADODB.Recordset
Dim rcdDersler As New ADODB.Recordset
Dim rcdYLDersleri As New ADODB.Recordset
Dim rcdKullanıcılarÖğretimElemanları As New ADODB.Recordset
Dim ii As Integer
Dim i As Integer
Dim j As Integer
Dim k As Integer
Dim L As Integer
Dim m As Integer
DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
Set cnThisConnect = CurrentProject.Connection
rcdKullanıcılarÖğretimElemanları.Open "tblKullanıcılarÖğretimElemanları", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdÖğretimElemanları.Open "tblÖğretimElemanları", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdDersler.Open "tblDersler", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
rcdYLDersleri.Open "tblYLDersler", cnThisConnect, _
    adOpenKeyset, adLockOptimistic, adCmdTable
If Me.txtKullanıcıAdı = "" Then
    Me.txtKullanıcıAdı = ""
    Me.txtŞifre = ""
    Me.txtKullanıcıAdı.SetFocus
    MsgBox "Lütfen bir Kullanıcı Adı giriniz.", vbCritical, gstrAppTitle

```

```

Exit Sub
End If
If Me.txtŞifre = "" Then
    Me.txtKullanıcıAdı = ""
    Me.txtŞifre = ""
    Me.txtKullanıcıAdı.SetFocus
    MsgBox "Lütfen bir Şifre giriniz. ", vbCritical, gstrAppTitle
    Exit Sub
End If
If Me.txtKullanıcıAdı = "" And Me.txtŞifre = "" Then
    Me.txtKullanıcıAdı = ""
    Me.txtŞifre = ""
    Me.txtKullanıcıAdı.SetFocus
    MsgBox "Lütfen bir Kullanıcı Adı ve Şifre giriniz. ", vbCritical, gstrAppTitle
    Exit Sub
End If
For ii = 1 To rcdKullanıcılarÖğretimElemanları.RecordCount
If rcdKullanıcılarÖğretimElemanları![KullanıcıAdı] = Me.txtKullanıcıAdı Then
    Me.txtKullanıcıAdı = ""
    Me.txtŞifre = ""
    Me.txtKullanıcıAdı.SetFocus
    MsgBox "Lütfen farklı bir Kullanıcı Adı giriniz. ", vbCritical, gstrAppTitle
    Exit Sub
End If
rcdKullanıcılarÖğretimElemanları.MoveNext
Next ii
rcdKullanıcılarÖğretimElemanları.MoveFirst
For ii = 1 To rcdKullanıcılarÖğretimElemanları.RecordCount
If rcdKullanıcılarÖğretimElemanları![Şifre] = Me.txtŞifre Then
    Me.txtKullanıcıAdı = ""
    Me.txtŞifre = ""
    Me.txtKullanıcıAdı.SetFocus
    MsgBox "Lütfen farklı bir Şifre giriniz. ", vbCritical, gstrAppTitle
    Exit Sub
End If
rcdKullanıcılarÖğretimElemanları.MoveNext
Next ii
rcdKullanıcılarÖğretimElemanları.MoveFirst
For ii = 1 To rcdKullanıcılarÖğretimElemanları.RecordCount
If (rcdKullanıcılarÖğretimElemanları![KullanıcıAdı] = Me.txtKullanıcıAdı) And
(rcdKullanıcılarÖğretimElemanları![Şifre] = Me.txtŞifre) Then
    Me.txtKullanıcıAdı = ""
    Me.txtŞifre = ""
    Me.txtKullanıcıAdı.SetFocus
    MsgBox "Lütfen farklı bir Kullanıcı Adı ve Şifre giriniz. ", vbCritical, gstrAppTitle
    Exit Sub
End If
rcdKullanıcılarÖğretimElemanları.MoveNext
Next ii
rcdKullanıcılarÖğretimElemanları.MoveFirst
For i = 1 To rcdKullanıcılarÖğretimElemanları.RecordCount
If rcdKullanıcılarÖğretimElemanları![ÖğretimElemanıID] = Me.ÖğretimElemanıID Then
    rcdKullanıcılarÖğretimElemanları![KullanıcıAdı] = Me.txtKullanıcıAdı
    rcdKullanıcılarÖğretimElemanları![Şifre] = Me.txtŞifre
End If
rcdKullanıcılarÖğretimElemanları.MoveNext
Next i
For j = 1 To rcdÖğretimElemanları.RecordCount
If rcdÖğretimElemanları![ÖğretimElemanıID] = Me.ÖğretimElemanıID Then
    rcdÖğretimElemanları![KullanıcıAdı] = Me.txtKullanıcıAdı

```



```

End If
rcdÖğretimElemanları.MoveNext
Next j
For k = 1 To rcdDersler.RecordCount
    If rcdDersler![ÖğretimElemanıID] = Me.ÖğretimElemanıID Then
        rcdDersler![KullanıcıAdı] = Me.txtKullanıcıAdı
    End If
    rcdDersler.MoveNext
Next k
For L = 1 To rcdYLDersleri.RecordCount
    If rcdYLDersleri![ÖğretimElemanıID] = Me.ÖğretimElemanıID Then
        rcdYLDersleri![KullanıcıAdı] = Me.txtKullanıcıAdı
    End If
    rcdYLDersleri.MoveNext
Next L
Me.RecordSource = "qryKullanıcılarÖğretimElemanları"
Me.txtKullanıcıAdı = ""
Me.txtŞifre = ""
Me.txtKullanıcıAdı.SetFocus
End Sub

```

Kodlarda ilk başta yeni kullanıcı adı ve şifre alanlarının doldurulmuş olması sağlanır. Daha sonra girilen bu değerlerin bir başka kullanıcı tarafından kullanılıp kullanılmadığının araması yapılır. Bunun için “tblKullanıcılarÖğretimElemanları” tablosunun tüm kayıtları tek tek incelenir. Eğer eskisiyle değiştirilmesi istenen yeni değerler tabloda yer almıyorsa yani başka bir kullanıcı tarafından kullanılmıyorsa istenen değişiklik yapılır.

Yeni Düğmesi: Yeni bir öğretim elemanı girişini sağlayan “Yeni Öğretim Elemanı Girişi” ekranını açar.

Yazdır Düğmesi: Bilgileri görüntülenmekte olan öğretim elemanına ait raporların alınmasını sağlayan “Öğretim Elemanı Raporları” print ekranının açılmasını sağlar. Ekranın görüntüsü Ekran 8.66’da gösterildiği gibidir.



Ekran 8.66. “Öğretim Elemanı Raporları” print ekranı.

Ekranın işleyişi ve verdiği raporlar daha önce Kesim 8.4.4.1.1.’de sözü edilen “Öğrenci Raporları” print ekranıyla (Ekran 8.50) benzer olduğundan burada ayrıca ayrıntısına girilmeyecektir.

Sil Düğmesi: Görüntülenmekte olan öğretim elemanına ait tüm bilgilerin silinmesini sağlayan düğmedir. Bu işlem aynı zamanda öğretim elemanının Baykuş programının kullanıcılarından biri olmaktan çıkmasını da sağlamaktadır.

8.4.5. ÇIKIŞ DÜĞMESİ

Baykuş programının ana menüsünde yer alan “Çıkış” düğmesi, programdan çıkışı sağlayan düğme olup aynı zamanda program verilerinin yedeklenmesi işlemi de gerçekleştirmektedir. Düğmenin işlettiği VB kodları aşağıdaki gibidir (Viescas, 2005).

```
Private Sub cmdÇıkış_Click()
Dim intErr As Integer, frm As Form, intI As Integer, strData As String, strDir As String
Dim lngOpen As Long, datBackup As Date
Dim strLowBkp As String, strBkp As String, intBkp As Integer
Dim db As DAO.Database, rst As DAO.Recordset
If vbNo = MsgBox("Uygulamadan çıkmak istediğinizden emin misiniz?", vbYesNo + vbQuestion
+ vbDefaultButton2, _
gstrAppTitle) Then
Exit Sub
End If
On Error Resume Next
For intI = (Forms.Count - 1) To 0 Step -1
Set frm = Forms(intI)
If frm.Name <> "frmMain" Then
frm.cmdCancel_Click
DoEvents

```

```

End If
    If Err <> 0 Then intErr = -1
Next intI
On Error GoTo frmMain_Error
If intErr = 0 Then
    Set db = CurrentDb
    Set rst = db.OpenRecordset("ztblVersion", dbOpenDynaset)
    rst.MoveFirst
    lngOpen = rst!OpenCount
    datBackup = rst!LastBackup
    rst.Close
    Set rst = Nothing
    If (lngOpen Mod 10 = 0) Or ((Date - datBackup) > 14) Then
        If vbYes = MsgBox("MSGSU uygulaması, verilerinizde herhangi bir kaybın olmaması " & _
            "için verilerin yedeklenmesini önermektedir. Yedekleme işleminin şimdi
gerçekleştirilmesini ister misiniz?", _
            vbYesNo + vbQuestion, gstrAppTitle) Then
            strData = Mid(db.TableDefs("ztblVersion").Connect, 11)
            strDir = Left(strData, InStrRev(strData, "\"))
            If Len(Dir(strDir & "BackupData", vbDirectory)) = 0 Then
                MkDir strDir & "BackupData"
            End If
            strBkp = Dir(strDir & "BackupData\MSGSÜBkp*.mdb")
            Do While Len(strBkp) > 0
                intBkp = intBkp + 1
                If (strBkp < strLowBkp) Or (Len(strLowBkp) = 0) Then
                    strLowBkp = strBkp
                End If
                strBkp = Dir
            Loop
            If intBkp > 2 Then
                Kill strDir & "BackupData\" & strLowBkp
            End If
            strBkp = strDir & "BackupData\MSGSÜBkp" & Format(Date, "yymmdd") & ".mdb"
            If Len(Dir(strBkp)) > 0 Then Kill strBkp
            DBEngine.CompactDatabase strData, strBkp
            db.Execute "UPDATE ztblVersion SET LastBackup = #" & Date & "#", dbFailOnError
            MsgBox "Verilerin yedeklenmesi başarıyla gerçekleştirildi!", vbInformation, gstrAppTitle
        End If
        If db.TableDefs("ErrorLog").RecordCount > 20 Then
            If Not (DLookup("DontSendError", "tblUsers", _
                "UserName = " & gstrThisUser & "")) Then
                DoCmd.OpenForm "fdlgErrorSend", WindowMode:=acDialog
            Else
                db.Execute "DELETE * FROM ErrorLog", dbFailOnError
            End If
        End If
    End If
    Set db = Nothing
End If
frmMain_Exit:
    On Error GoTo 0
    DoCmd.Close acForm, Me.Name
    DoCmd.SelectObject acForm, "frmMain", True
    Exit Sub
frmMain_Error:
    ErrorLog "frmMain", Err, Error
    Resume frmMain_Exit
End Sub

```

Yedekleme işlemi Baykuş programında bulunan “ztblVersion” tablosuyla işlem yapmaktadır. Tabloda versiyonu gösteren “Versyon”, son yedekleme tarihini gösteren “SonYedekleme” ve programın kaç kez açıldığını gösteren “AçılışSayısı” alanları yer almaktadır.

Yukarıdaki VB kodlarıyla, program ilk olarak yedekleme işlemine gerek olup olmadığına karar verir. Bunun için “ztblVersion” tablosunu açar ve bu tabloda yer alan verileri inceler. Eğer kullanıcılar programı 10 kez açmışlar veya yapılan en son yedeklemenin tarihi 2 hafta öncesinin tarihi ise bu durumda program, "MSGSÜ uygulaması, verilerinizde herhangi bir kaybın olmaması için verilerin yedeklenmesini önermektedir. Yedekleme işleminin şimdi gerçekleştirilmesini ister misiniz?" şeklinde bir mesaj verir. Eğer mesaja olumlu cevap verilirse yedekleme işlemi gerçekleşir.

Yedekleme daha önce sözü edilen MSGSÜData dosyasının Baykuş programının yer aldığı klasör içinde bulunan “BacupData” adındaki klasöre “MSGSÜBkp” adıyla kaydedilmesini sağlar.

BÖLÜM 9

SONUÇLAR VE TARTIŞMA

Baykuş programı, MSGSÜ Fen Edebiyat Fakültesi İstatistik Bölümü lisans, bilimsel hazırlık, yüksek lisans ve doktora programlarına kayıtlı öğrenciler ile bölüm öğretim elemanlarına, yetkileri dahilinde yardımcı olmak amacıyla tasarlanmış bir KD Sistemi'dir. Program; (1) lisans öğrencisi, (2) lisans üstü öğrencisi, (3) öğretim elemanı ve (4) bölüm başkanı olmak üzere dört farklı kullanıcı türüne hizmet vermektedir. Bu kullanıcı türlerinin her biri farklı özelliklere sahip olduğundan, doğal olarak gerçekleştirmeleri gereken ve gerçekleştirmek istedikleri işlemler de farklılık gösterir.

Farklı türdeki kullanıcıların her birinin Baykuş programını kullanarak gerçekleştirebileceği işlemler ve bu işlemlerin ardından ulaşabileceği sonuçlar aşağıdaki alt başlıklarda sıralanmıştır.

Lisans ve Lisansüstü Öğrencisi Kullanıcıları: Baykuş programı aracılığıyla lisans ve lisans üstü programlarına kayıtlı öğrencilerin gerçekleştirebilecekleri işlemler ve elde ettikleri sonuçlar aşağıdaki gibidir.

- Öğrencinin bulunduğu döneme ilişkin seçebileceği derslerin listesini görüp bu derslerden istediğini seçmesi. Böylelikle danışmanının öğrencinin ders seçim tercihleri hakkında bilgi sahibi olması sağlanmaktadır.
- Öğrencinin yaptığı ders seçimlerinden danışmanı tarafından onay görmüş olanlarının öğrenciye gösterilmesi. Böylelikle öğrenci ve danışman iletişimi sağlanarak ilgili dönemde öğrencinin alacağı derslere ortak bir karar verilmesi sağlanmaktadır.
- Öğrenciye dönem derslerinin sınav sonuçlarını gösterme. Böylelikle öğrenciye aldığı dersler hakkında bilgi verilerek, önemsemesi gereken dersleri hakkında karar vermesi sağlanmaktadır.

- Öğrencinin ilgili döneme kadar aldığı tüm dersleri ve başarı durumunu gösterme. Böylelikle, akademik ortalaması ve kredi toplamları hakkında bilgilendirilen öğrencinin sonraki dönemlerde alması gereken ders sayısına karar vermesine yardımcı olunmaktadır.

- Kişisel bilgileri gösterme. Böylelikle öğrencinin kişisel bilgilerinin doğru olup olmadığını kontrol etmesine yardımcı olunmaktadır.

Öğretim Elemanı Kullanıcısı: Baykuş programı aracılığıyla öğretim elemanları aşağıdaki işlemleri gerçekleştirebilirler.

- Kendilerine ait bilgilerin doğruluğunu kontrol edebilmek.
- Danışmanı oldukları öğrencilerin durumlarını izlemek ve ders seçimlerine yardımcı olup seçimlerini onaylamak.
- Danışmanı oldukları öğrencilerle ilgili karar almak gerektiğinde bu öğrencilere ilişkin gerekli bilgilere ulaşabilmek.
- Yürüttükleri derslerin sınav sonuçlarını girmek.
- İstatistik Bölümü'nün tüm programlarında kayıtlı olan öğrencilerin kişisel bilgilerine ulaşmak.

Bölüm Başkanı Kullanıcısı: Baykuş programı aracılığıyla bölüm başkanı aşağıda sıralanan işlemleri gerçekleştirebilmektedir.

- Bölüm öğretim elemanlarının kişisel bilgilerine ulaşabilme ve bu bilgileri değiştirebilme.
- Lisans, bilimsel hazırlık, yüksek lisans ve doktora programındaki öğrencilerin kişisel bilgileri ve bölüm bilgilerine ulaşabilme, değiştirebilme ve silebilme.
- Bölümde okutulan bütün lisans, bilimsel hazırlık, yüksek lisans ve doktora derslerine ulaşabilme, değiştirebilme ve silebilme.
- Baykuş programının güncellemelerinin, düzeltmelerinin veya eklemelerinin yapılmasını sağlayan ekranları kullanabilme.

KAYNAKLAR

- Alter, S., 1977, A Taxonomy of Decision Support Systems, Sloan Management Review, Vol.19, No.1, Fall.
- Alter, S., 1992, Information Systems A Management Perspective, Addison-Wesley Publishing Company, USA.
- Alter, S., 1980, Decision Support Systems: Current Practice and Continuing Challenges, Addison-Wesley Publishing Company; USA.
- Andriole, S.J., 1986, Microcomputer Decision Support Systems: Design, Implementation, and Evaluation, QED Information Sciences.
- Anthony, R.N., 1965, Planning and Control Systems: A Framework for Analysis, Division of Research, Harvard Graduate School of Research, Boston.
- Baker, D., Bridges, D., Hunter, R., Johnson, G., Krupa, J., Murphy, J. and Sorenson, K., 2001, Guidebook to Decision-Making Methods, <http://emi-web.inel.gov/Nissmg/Guidebook-2002.pdf>, erişim tarihi: 05.09.2006.
- Barquin, R., 1996, On the First Issue of the Journal of Data Warehousing, The Journal of Data Warehousing, 1 (July): 2-6.
- Beynon-Davies, P., 1991, Expert Database Systems a Gentle Introduction, UK: McGraw-Hill Book Company (UK) Limited.
- Bielawski, L. and Lewand, R., 1991, Intelligent Systems Design: Integrating Expert Systems, Hypermedia, and Database Technologies, United States America: John Wiley & Sons, Inc.
- Bhargava, H.K., Sridhar, S. and Herrick, C., 1999, Beyond Spreadsheets: Tools for building Decision Support Systems, IEEE, sy. 31-39.
- Bohanec, M., 2006, What is Decision Support?, Department of Intelligent Systems, Jozef Stefan Institute, Slovenia.
- Bui, T. and Lee, J., 1999, An Agent-Based Framework for Building Decision Support Systems, Decision Support Systems, 25(1999): 225-237.
- Cinemre, N., 2004, Yöneylem Araştırması, İkinci Baskı, Beta Basım Yayım Dağıtım A.Ş., Çağaloğlu-İstanbul.
- Courtney, J.F., 2001, Decision Making and Knowledge Management in Inquiring Organizations: Toward a New Decision-Making Paradigm for DSS, Decision Support Systems, 31 (2001): 17-38.
- Covaliu, Z., 2001, Decision Analysis: Concepts, Tools and Promise, A Fair Isaac White Paper, sy.17.

- Codd, E.F., Codd, S.B. and Salley, C.T., Providing OLAP to User-Analysts: An IT Mandate, Eriřim: [<http://dev.hyperion.com/resource-library/white-papers/providing-olap-to-user-analysts.pdf>]. Eriřim: 03.02.2006.
- Davis, M.W., 1988, Applied Decision Support, Prentice-Hall.
- Donovan, J.J., and Madnick, S.E., 1977, Institutional and Ad-Hoc DSS and Their Effective Use, Data Base, Vol.8, No.3.
- Eom, S.B., 2001, Decision Support Systems, International Encyclopedia of Business and Management, 2nd Edition, Edited by Malcolm Warner, International Thomson Publishing Co., London
- Erpolat, S. ve Cinemre, N., 2006, Analitik Hiyerarři Yöntemi'yle İř Sektörü Seçimi, MÜ Sos. Bil. Enst. Hakemli Dergisi Öneri, sayı: 25, cilt: 7, sy. 231-241.
- Finlay, P., 1994, Introducing Decision Support Systems, Great Britain: Blackwell Publishers.
- Goodall, A., 1985, The Guide to Expert Systems, Oxford and New Jersey: Learned Information.
- Gorry, G.A. and Scott-Morton, M.S., 1971, A Framework for Management Information Systems, Slogan Management Review, 13 (1), Fall 1971: 55-70.
- Hackathorn, R. D., and Keen, P. G. W., 1981, Organizational Strategies for Personal Computing in Decision Support Systems, MIS Quarterly, Vol. 5, No. 3.
- Hayes-Roth, F., 1983, Building Expert Systems, Addison-Wesley Publishing.
- Hoffer, J.A., Prescott, M.B. and McFadden, F.R., 2002, Modern Database Management, 6th Edition, New Jersey: Pearson Education, Inc.
- Holsapple, C.W., 2001, Editorial: Knowledge Management Support of Decision-Making, Decision Support Systems, 31 (2001): 1-3.
- Holsapple, C.W. and A.B. Winston, 1996, Decision Support Systems: A Knowledge-Based Approach St. Paul: West Publishing.
- Howard, R.A., 1988, Decision Analysis: Practice and Promise, Management Science, Vol.34, No.6, sy.679.
- Inmon, W.H. and Hackathorn, R.D., 1994, Using the Data Warehouse, New York: John Wiley & Sons.
- Kuruüzüm, A., 1998, Karar Destek Sistemlerinde Çok Amaçlı Yöntemler, Akdeniz Üniv. Basım evi, Antalya.
- Keen, P.G.W. and Scott-Morton, M.S. ,1978, "Decision Support Systems, An Organizational Perspective".

- Khoong, C.M., 1995, "DSS: An Extended Research", Omega, Int., J. Mgmt. Sci. Vol.23, pp.221-229.
- Kleijnen, P.C.J., 1993, "Simulation and Optimization in Production Planning." Decision Support Systems. 269-280.
- Klein, M.R. and Methlie, L.B., 1995, Knowledge-Based Decision Support Systems with Applications in Business, 2nd Edition, England: John Wiley & Sons.
- Lawrence, J.A. (jr.) and Pasternack, B.A., 2002, Applied Management Science: Modelling Spreadsheet Analysis, and Communication for Decision Making, 2nd Edition, United States of America: John Wiley & Sons, Inc.
- Mallach, E.G., 1994, Understanding Decision Support Systems and Expert Systems, United States of America: John Wiley & Sons, Inc.
- Mallach, E.G., 2000, Decision Support and Data Warehouse Systems, Irwin/McGraw-Hill, Boston.
- Nabiyev, V.V., 2003, Yapay Zeka, 1. Basım, Ankara: Seçkin Yayıncılık.
- Nicerson, R.C., 1998, Business And Information systems, Addison-Wesley Educational Publishers, USA.
- O'Keefe, R.M. and McEachern, T., 1998, Web-Based Customer Decision Support Systems, Communications of the ACM, 41 (3), March: 71-80.
- Olson, D.L. and Courney, J.F. (jr.), 1992, Decision Support Models and Expert Systems, United States of America: Macmillan Publishing Company.
- Pal, K. and Palmer, O., 2000, A Decision Support System for Business Acquisitions, Decision Support Systems, 27 (2000): 411-429.
- Poe, V., 1996, Building A Data Warehouse For Decision Support, Upper Saddle River, NJ: Prentice Hall PIR.
- Raggad, B.G. and Gargano, M.L., 1999, Expert System: Defection and Perfection, Logistics Information Management, 12 (5): 395-406.
- Reynolds, G.W., 1995, Information Systems For Managers, West Publishing
- Rhia, K.C., 2000, Preparing for a Decision Support System, Topics in Health Information Managemant, 21-1, ABI/INFOR Global.
- Sauter, V.L., 1999, Intuitive Decision Making, Communications of the ACM, 42 (6) June: 109-122.
- Scott-Morton, M.S., 1971, Management Decision Systems: Comuter-Based Support for Decision Making, Division of Research, Harvard University, Cambridge, Mass.

- Sharifi, M.A. and Keulen, H.V., 1994, "A Decision Support System for Land Use Planning at Farm Enterprise Level." *Agricultural Systems*, 239-257.
- Shim, J.P., Warketin, M., Courtney, J.F., Power, D.J., Sharda, R. and Carlsson, C., 2002, Past Present and Future of Decision Support Technology, *Decision Support Systems*, 33 (2002), February: 111-126.
- Shoshani, A., 1997, OLAP and Statistical Databases: Similarities and Differences, In *Proceedings of the 16th ACM SIGACT-SIGMOD-SIGART Symposium on Principles and Database Systems (PODS'97)*, Tucson, USA, 185-196.
- Simmons, L.F. and Poulos, L., 1988, "DSS: The Successful Implementation of A Mathematical Programming Model For Strategic Planning", *Comput. Opns. Res.*, Vol. 15, No.1, pp. 1-5.
- Simon, H.A., 1960, *The New Science of Management Decision*, New York: Harper & Brothers Publishers.
- Simon, H.A., 1977, *The New Science of Management Decision*, United States of America: Prentice-Hall International, Inc.
- Sprague, R.H. (jr.) and Carlson, E.D., 1982, *Building Effective Decision Support Systems*, Second Reprinting 1986, Grolier Incorporated, New Jersey: Prentice-Hall.
- Sprague, R.H. (jr) and Watson, H.J., 1996, *Decision Support For Management*, Prentice-Hall, USA.
- Sprague, R.H., 1980, A Framework for the Development of Decision Support Systems, *MIS Quarterly*.
- Stair, R.M., 1992, *Principles of Information Systems A Managerial Approach*.
- Taylor, B. W. III, 1999, *Introduction to Management Science*, 6th Edition, United States of America: Prentice Hall.
- Turban, E., 1993, *Decision Support and Expert Systems: Management Support Systems*, 3rd Edition, New York: Macmillian Publishing Company.
- Turban, E., 1995, *Decision Support and Expert Systems: Management Support Systems*, 4th Edition, New York: Macmillian Publishing Company.
- Turban, E., Mclean, E. and Wetherbe, J., 1996, *Information Technology for Management*.
- Turban Efraim, Lee J., King D. & Chung H. M., 2000, *Electronic Commerce: A Managerial Perspective*, International Edition, United States of America: Prentice Hall.
- Turban, E., Mclean, E. and Wetherbe, J., 2001, *Information Technology for Strategic Advantage*, 2nd Edition. New York, Chichester, Weinheim, Brisbane, Singapore, Toronto: John Wiley & Sons, Inc.

- Turban, E., Aronson, J.E. and Liang, T-P, 2005, Decision Support Systems and Intelligent Systems, Pearson Prentice Hall.
- Vassiliadis, P. and Sellis T.K., 1999, A Survey of Logical Models for OLAP Databases, ACM SIGMOD Record, vol.28, no.4, pp.64-69.
- Viescas, J.L., 2005, Enine Boyuna Microsoft Office Access 2003, Arkadaş Yayınevi, Ankara.
- Watson, H.J., 1998, Decision Suport in the Data Warehouse, Prentice-Hall.
- Weigel, S.H. and Wilcox P.S., 1993, "The Army's Personnel Decision Support System." Decision support systems. 281-306.
- Weldon, J.L., 1996, Data Mining and Visualization, Database Programming and Design, 9 (May): 21-24.
- Zwass, V., 1998, Foundation of Informatian Systems, McGraw-Hill Companies Inc., Singapore.
- www.doc.ic.ac.uk/~frk/frank/kmt/DSS%20handout.pdf, Decision Support Systems, Erişim Tarihi: 17.03.2006.

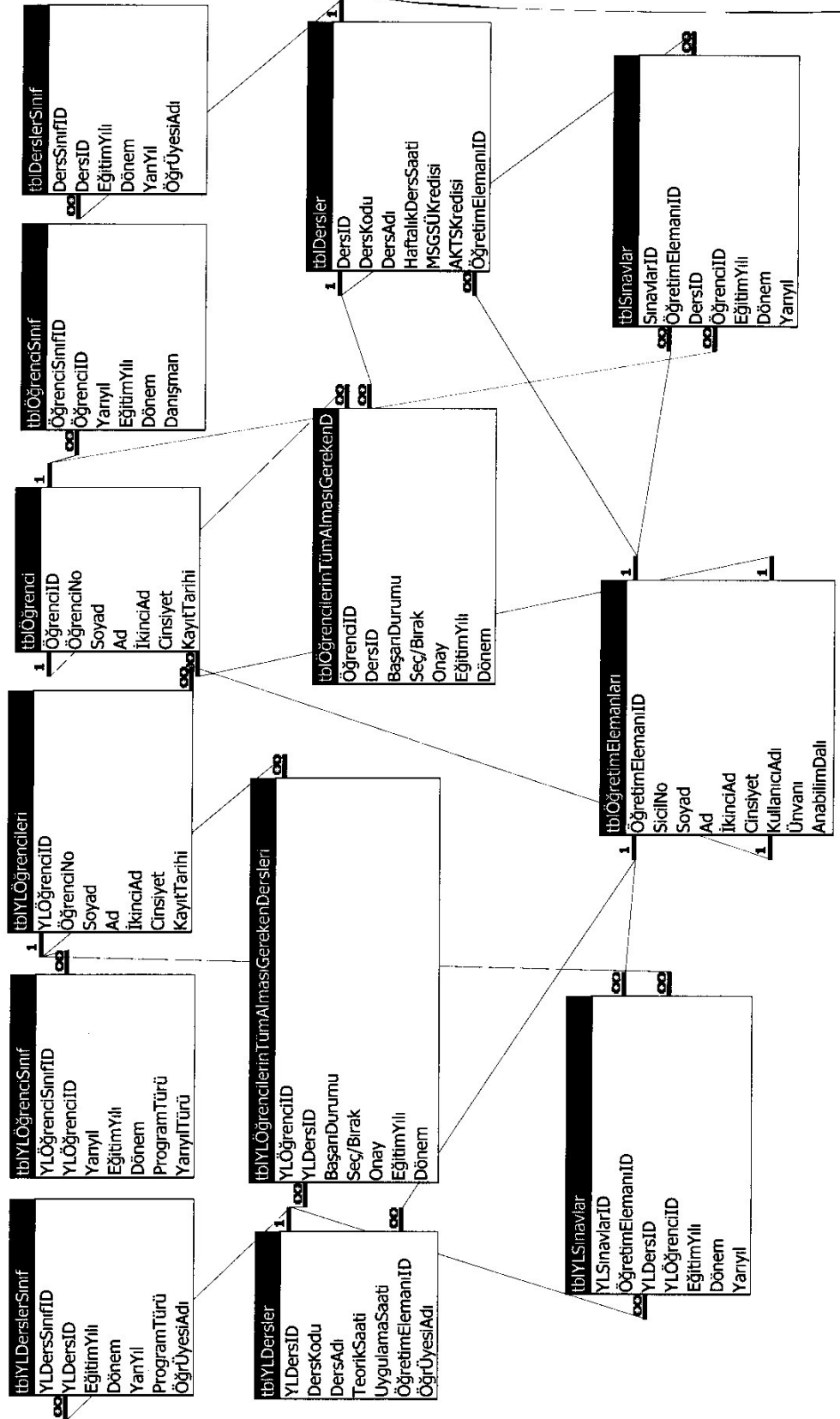
EKLER DİZİNİ

EK 1. TABLO LİSTESİ.....	247
EK 2. TABLO İLİŞKİLERİ.....	Error! Bookmark not defined.

EK 1. TABLO LİSTESİ

1. tblAkademikBilgilerÖğrenciler
2. tblAkademikBilgilerÖğretimElemanları
3. tblAkademikBilgilerYLOğrencileri
4. tblAnabilimDalı
5. tblCinsiyet
6. tblDanışman
7. tblDersler
8. tblDerslerSınıf
9. tblDönem
10. tblEğitimYılı
12. tblKullanıcılarBölümBaşkanı
13. tblKullanıcılarÖğrenciler
14. tblKullanıcılarÖğretimElemanları
15. tblKullanıcılarYLOğrencileri
16. tblÖğrenci
17. tblÖğrencilerinTümAlmasıGerekenDersleri
18. tblÖğrenciSınıf
19. tblÖğretimElemanları
20. tblSınavlar
21. tblTümÖğretimÜyeleri
22. tblYarıyıl
23. tblYayımlarÖğretimElemanları
24. tblYLDersler
25. tblYLDerslerSınıf
26. tblYLOğrencileri
27. tblYLOğrencilerinTümAlmasıGerekenDersleri
28. tblYLOğrenciSınıf
29. tblYLProgram
30. tblYLSınavlar
31. ztblVersion

EK 2. TABLO İLİŞKİLERİ



ÖZGEÇMİŞ

Adı Soyadı : Semra ERPOLAT

Doğum Yeri : Gercüş

Doğum Yılı : 16.09.1977

Medeni Hali : Bekar

Eğitim ve Akademik Durumu:

Lise : 1992-1995 Z.H. Anadolu Meslek Lisesi Bilgisayar Bölümü

Lisans : 1995-1999 Hacettepe Üniversitesi Fen Fakültesi İstatistik Bölümü

Yüksek Lisans: 1999-2002 Hacettepe Üniversitesi Fen Fakültesi İstatistik Bölümü

Yabancı Dil : İngilizce

Çalışmalar:

1."Karar Destek Sistemleri: Akademik Başarının Kişisel Çabaya Etkisi", 1999, Hacettepe Üniversitesi, Fen Edebiyat Fakültesi, İstatistik Bölümü, Yüksek Lisans Tezi.

2. Erpolat S., Cinemre N., 2006, "Analitik Hiyerarşi Yöntemi'yle İş Sektörü Seçimi", MÜ Sos. Bil. Enst. Hakemli Dergisi Öneri, sayı: 25, cilt: 7, sy. 231-241.

3. Erpolat S., Cinemre N., "Bilgisayar Yazılımları Öğreniminde Kişisel Başarının Akademik Başarıya Etkisi, Öneri dergisi", Öneri dergisi Marmara Ün.v.(baskıda).

4. Erpolat S., Yay M., 2005, "Çokdeğişkenli Yardımcı Bilginin Kullanıldığı İki Aşamalı Örneklemede Sonlu Yığın Ortalaması İçin Bazı Tahmin Edicilerin Görelî Etkinliklerinin İncelenmesi", 4. İstatistik kongresi.

5. Erpolat S., 2006, "İstatistiksel Veri Tabanları ve OLAP", 15. İstatistik Araştırma Sempozyumu 2006.