

AFYON KOCATEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

UZAKTAN EĞİTİMDE VERİTABANI TASARIMI VE ÖRNEK MODEL

Yılmaz SARP KAYA

DANIŞMAN

Yrd. Doç. Dr. Ömer DEPERLİOĞLU

BİLGİSAYAR ANABİLİM DALI


EYLÜL – 2008

ONAY SAYFASI

Yrd. Doç. Dr. Ömer DEPERLİOĞLU danışmanlığında, Yılmaz SARP KAYA tarafından hazırlanan “Uzaktan Eğitimde Veritabanı Tasarımı ve Örnek Model” başlıklı bu çalışma, lisansüstü eğitim ve öğretim yönetmeliğinin ilgili maddeleri uyarınca

15 / 09 / 2008

tarhinde aşğıdaki jüri tarafından Bilgisayar Anabilim Dalında yüksek lisans tezi olarak oybirliğı ile kabul edilmiştir.

	Unvanı, Adı, SOYADI	İmza
Başkan	Yrd. Doç. Dr. Ömer DEPERLİOĞLU	
Üye	Doç. Dr. Hüseyin BAYRAKÇEKEN	
Üye	Yrd. Doç. Dr. Uçman ERGÜN	

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü Yönetin Kurulu'nun

19 / 09 / 2008 tarih ve

2008/22-32 sayılı kararıyla onaylanmıştır.



Doç. Dr. Zehra BOZKURT

Enstitü Müdürü

TEŐEKKÜR

Çalıőmalarım sürecinde düőünce, fikir ve görüşlerini esirgemeyen deęerli hocam
Yrd. Doç. Dr. Ömer DEPERLİOđLU na sonsuz teőekkürlerimi sunarım.

Yılmaz SARP KAYA
Afyonkarahisar Eylül 2008

ÖZET

Yüksek Lisans Tezi

Uzaktan Eğitimde Veritabanı Tasarımı ve Örnek Model

Yılmaz SARP KAYA

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Anabilim dalı

Danışman: Yrd. Doç. Dr. Ömer DEPERLİOĞLU

Türkiye uzaktan eğitim uygulamasına yeni adapte olmuş ve hızlı bir şekilde gündemine alarak kullanıma başlamıştır. Şimdiye kadar belirli bir ders yada uygulama modülü için birçok veritabanı tasarımı yapılmış ancak evrensel değerlerde bir eğitim kurumunu oluşturabilecek seviyede geniş çaplı bir çalışma henüz yayınlanmamıştır. Üniversiteler bu uygulamanın en çok yapıldığı eğitim kurumlarından biridir, ancak her bir kurum kendi uzaktan eğitim modeli için bir veritabanı hazırlamaktadır.

“Uzaktan Eğitimde Veritabanı Tasarımı ve Örnek Model” adlı çalışmada her kurum ve kuruluşa uyumlu, her ders için uyarlanabilir, gelişmiş, yönetimi kolay 153 tablo ve bu tabloların birbirleri ile tam ilişki içerisinde olduğu bir veritabanı tasarımı yapılmıştır. Veritabanı tasarımında Microsoft SQL Server 2005 veritabanı tasarım yazılımı kullanılmıştır. Veritabanı 38 ana modülden oluşmakta, farklı seviyelerde kullanıcılar tanımlanmakta ve bu kullanıcılara farklı erişim seviyeleri verilmektedir. Sistem bir bütün olarak farklı akademik ve idari personel tarafından yönetilmektedir.

Tasarlanan veritabanı ister Windows form ara-yüzüne istenirse web formu ara-yüzüne uyarlanabilmekte ve kullanıcı isteğine göre geliştirilebilmektedir.

2008 – 96

Anahtar kelimeler: Eğitim Öğretim Yönetim Sistemi (LMS), SQL Server, Veritabanı, İlişkisel Tasarım, Uzaktan Eğitim

ABSTRACT

Master of Science

Database Design for Distance Education and Sample Design

Yılmaz SARP KAYA

Afyon Kocatepe University

Graduate School of Natural and Applied Sciences

Computer

Supervisor: Asst. Prof. Dr. Ömer DEPERLİOĞLU

Turkey has been nearly adopted the application of distance education and started to use by taking its agenda. Up to now lots of database has been designed for specific courses or applications but a large-scale study which include establishing a universal education institute has not been released. Universities are the education institutions which use this system but each education institution designs own database model for their distance education system.

In this study a full relational global database for Learning Management System (LMS) has been designed in MS SQL Server. The database is compatible with every education institutions and could be adjustable every course also is easy to manage with 153 relational tables. Microsoft SQL Server 2005 software was used for this database design. The database consist of 38 modules, different users in different can be defined and different roles and levels are given to these users. Whole system is managed as entirely by academic and administrative staff.

The database is adopted to different forms such Windows or WEB, also can be developed depending on developers.

2008 – 96

Keywords: Learning Management System (LMS), SQL Server, Database, Relational Design, Distance Education

İÇİNDEKİLER

Onay Sayfası	ii
Teşekkür	iii
Özet.....	iv
Abstract.....	v
İçindekiler	vi
Simgeler ve Kısaltmalar	ix
Şekiller Dizini	x
1. GİRİŞ.....	1
2. VERİ TABANI YÖNETİMİ VE UZAKTAN ÖĞRETİM SİSTEMLERİ.....	4
2.1 SCORM Standartları	6
2.2 Uzaktan Eğitim.....	7
2.2.1 Dünyada Uzaktan Eğitimin Gelişimi.....	8
2.2.2 Türkiye’de Uzaktan Eğitimin Gelişimi	9
2.2.3 Bilgisayar Tabanlı Uzaktan Eğitim Modelleri	10
2.2.3.1 Çevrimiçi Uzaktan Eğitim.....	10
2.2.3.2 Çevrimiçi Uzaktan Eğitimde Modeli ve Teknolojileri.....	11
2.2.3.3 Çevrimdışı Enteraktif Eğitim	14
3. VERİTABANI.....	15
3.1 Veri (DATA)	15
3.2 Veri Tabanı (DATABASE).....	16
3.2.1 Üstünlükleri	16
3.3 Veritabanı Modelleri ve Gelişimi	17
3.3.1 Hiyerarşik Veritabanı Modeli.....	18
3.3.2 Ağ Veritabanı Modeli.....	19
3.3.3 İlişkisel Veritabanı Modeli	19
3.3.4 Nesne Yönelimli Veritabanı	21
3.4 Veritabanı Tasarımı	21
3.4.1 Temel Terimler	22
3.4.2 Doğru Planlama ve Tasarım Nedir?	22
3.4.3 Tasarıma giriş	23
3.4.4 Hedefin Belirlenmesi.....	23

3.4.5 Verileri Toplama ve Düzenleme.....	24
3.4.6 Nesneleri Tablolara Aktarma.....	24
3.4.7 Tablo Verilerini Sütunlara Aktarma.....	25
3.4.8 Birincil Anahtar Belirleme.....	26
4.4.9 Tablolar Arasında İlişki Kurma.....	27
3.4.9.1 Bire-Sonsuz (1-∞) ilişki,.....	27
3.4.9.2 Sonsuz-Sonsuz (∞-∞) ilişki,.....	28
3.4.9.3 Bire-Bir (1-1) ilişki,.....	29
3.4.10 Detaylandırma.....	29
3.4.11 Normalizasyon İlkeleri ve Uygunluğun Denetlenmesi.....	30
3.5 Veritabanı Yönetim Sistemleri.....	32
3.6 MS SQL Server 2005.....	33
3.6.1 Veritabanı diyagramları (Database Diagrams).....	35
3.6.2 Tablolar (Tables).....	36
3.6.3 Görünümler (Views).....	37
3.6.4 Senonimler (Synonyms).....	37
3.6.5 Programlanabilirlik (Programmability).....	38
3.6.6 Servis Habercisi (Service Broker).....	38
3.6.7 Saklama (Storage).....	39
3.6.8 Güvenlik (Security).....	39
4. EĞİTİM-ÖĞRETİM YÖNETİM SİSTEMİ İÇİN VERİTABANI TASARIMI.....	41
4.1 Online Başvuru Modülü.....	43
4.2 Kullanıcı Ve Kullanıcı Grupları Modülü.....	44
4.3 Rol Ve Yetkiler Modülü.....	46
4.4 Unvan Tanımlama Modülü.....	47
4.5 Okul Tanımlama Modülü.....	48
4.6 Bölüm Tanımlama Modülü.....	48
4.7 Eğitim Kataloğu Modülü.....	49
4.8 Dersler Modülü.....	50
4.9 Ders Programı Modülü.....	51
4.10 Sınıf İşlemleri Modülü.....	52
4.11 Danışman Atama İşlemleri Modülü.....	53

4.12 Ders Kayıt Ve Ders Onay İşlemleri Modülü	54
4.13 Soru Tanımlama Modülü.....	55
4.14 Sınav Modülü	56
4.15 Deneme Sınavı Modülü.....	57
4.16 Ödevler Modülü.....	58
4.17 Öğretim Elemanı Performans Takip.....	59
4.18 Öğrenci Performans Takip Modülü.....	60
4.19 Öğrenci Takip Raporu Modülü.....	61
4.20 Sözlük Modülü	62
4.21 Anket Modülü.....	62
4.22 Kişisel Notlar Modülü	63
4.23 Mesajlaşma Modülü	64
4.24 Dosyalar Modülü	65
4.25 Sohbet Modülü	66
4.26 Toplantı Modülü.....	67
4.27 Sıkça Sorulan Sorular Modülü	68
4.28 Görüş Ve Öneriler Modülü.....	69
4.29 Duyurular Modülü	69
4.30 Ajanda Modülü.....	70
4.31 Akademik Takvim Modülü	71
4.32 Transkript Görüntüleme Modülü.....	72
4.33 Raporlama Modülü.....	73
4.34 Mali İşlemler Modülü.....	74
4.35 Demo Kullanıcı Modülü.....	75
4.36 Workshop (Çalıştay) Modülü.....	76
4.37 Sanal Sınıf Modülü.....	77
4.38 Forum Modülü.....	78
4.39 Sisteme Erişim, Rol ve Yetkiler	79
5. SONUÇ VE ÖNERİLER	88
5.1 Yapılan Çalışmanın Eğitime ve Bilime Katkısı	89
6. KAYNAKLAR.....	93
ÖZGEÇMİŞ.....	97

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar

EÖYS	Eğitim Öğretim Yönetim Sistemi
LMS	Learning Management System
SQL	Structured Query Language
VTYS	Veritabanı Yönetim Sistemi

ŞEKİLLER DİZİNİ

Şekil 3.1 Veritabanı modellerinin örnek üzerinde karşılaştırılması	18
Şekil 3.2 Hiyerarşik veritabanı modeline bir örnek	19
Şekil 3.3 İlişkisel veritabanı modeli	20
Şekil 3.4 Veritabanı tablolarının oluşturulması	24
Şekil 3.5 İlişkilendirilmiş veritabanı modeli	27
Şekil 3.6 Bire-Bir (1-1) ilişkilendirilmiş tablolar	29
Şekil 3.7 MS SQL Server yönetim paneli görüntüsü	34
Şekil 3.8 MS SQL Serverda bir veritabanında bulunan temel nesnelere	35
Şekil 3.9 Yeni veritabanı diyagramı oluşturma penceresi	36
Şekil 3.10 MS SQL Server tablo oluşturma ara-yüzü	36
Şekil 3.11 MS SQL Serverda görünüm oluşturma	37
Şekil 3.12 Programlanabilirlik nesnesinin alt nesnelere	38
Şekil 3.13 MS SQL Server güvenlik nesnesi ve alt nesnelere	39
Şekil 4.1 Başvuran modülü tabloları ve ilişkileri	44
Şekil 4.2 Kullanıcılar modülü ve ilişki senaryosu	46
Şekil 4.3 Rol ve Yetki modülü tabloları ve ilişki senaryosu	47
Şekil 4.4 Unvan tanımlama modülü tabloları ve ilişki senaryosu	47
Şekil 4.5 Okula tanımlama modülü tabloları ve ilişki senaryosu	48
Şekil 4.6 Bölüm tanımlama tablosu	49
Şekil 4.7 Eğitim katalogu modülü için tablolar ve ilişkileri	50
Şekil 4.8 Dersler modülü tabloları ve ilişkileri	51
Şekil 4.9 Ders programı modülü tabloları ve ilişkileri	52
Şekil 4.10 Sınıf tanımlama modülü tabloları ve ilişkileri	53
Şekil 4.11 Danışman atama işlemleri modülü tablolar ve ilişkileri	54
Şekil 4.12 Ders kayıt ve ders onay işlemleri modülü tabloları ve ilişkileri	55
Şekil 4.13 Soru tanımlama modülü tabloları ve ilişkileri	56
Şekil 4.14 Sınav modülü tabloları ve ilişkileri	57
Şekil 4.15 Deneme sınavı modülü tabloları ve ilişkileri	58
Şekil 4.16 Ödevler modülü tablo ve ilişkisi	59
Şekil 4.17 Öğretim elemanı performans takip modülü tabloları ve ilişkileri	60

Şekil 4.18 Öğrenci performans takip modülü tabloları ve ilişkileri	61
Şekil 4.19 Öğrenci takip raporu modülü tabloları ve ilişkisi.....	62
Şekil 4.20 Sözlük modülü tablo ve ilişkisi	62
Şekil 4.21 Anket modülü tabloları ve ilişkileri	63
Şekil 4.22 Kişisel notlar modülü tabloları ve ilişkileri	64
Şekil 4.23 Mesajlaşma modülü tabloları ve ilişkileri	65
Şekil 4.24 Dosyalar modülü tabloları ve ilişkileri.....	66
Şekil 4.25 Sohbet modülü tabloları ve ilişkileri	67
Şekil 4.26 Toplantı modülü tabloları ve ilişkileri.....	68
Şekil 4.27 Sıkça sorulan sorular modülü tabloları ve ilişkileri	69
Şekil 4.28 Görüş ve öneriler modülü tabloları ve ilişkileri	69
Şekil 4.29 Duyurular modülü tabloları ve ilişkiler	70
Şekil 4.30 Ajanda modülü tabloları ve ilişkisi	71
Şekil 4.31 Akademik takvim modülü tabloları ve ilişkileri	72
Şekil 4.32 Transkript modülü tabloları ve ilişkisi	73
Şekil 4.33 Raporlama modülü tabloları ve ilişkileri.....	74
Şekil 4.34 Mali işler modülü tabloları ve ilişkileri.....	75
Şekil 4.35 Demo kullanıcı modülü tabloları ve ilişkileri.....	76
Şekil 4.36 Çalıştay (Workshop) tabloları ve ilişkileri	77
Şekil 4.37 Sanal sınıf modülü tabloları ve ilişkileri	78
Şekil 4.38 Forum modülü tabloları ve ilişkileri.....	79
Şekil 4.39 Sisteme erişim rolleri ve görevleri	81
Şekil 4.40 Öğrenci işleri rol ve yetkisinin erişebileceği modüller	82
Şekil 4.41 Öğrenci rol ve yetkisinin erişebileceği modüller	83
Şekil 4.42 Öğretim elamanı rol ve yetkisinin erişebileceği modüller	84
Şekil 4.43 Bölüm başkanı rol ve yetkisinin erişebileceği modüller	85
Şekil 4.44 Danışman ve yetkisinin erişebileceği modüller.....	86
Şekil 4.45 Misafirlerin erişebileceği modüller	87

1. GİRİŞ

Bilindiği gibi uzaktan eğitim, günümüz dünyasının vazgeçilmez bir unsuru olmuştur. “Beşikten mezara kadar eğitim” deyimini internetin yaygınlaşması ve günlük hayata girmesi ile gerçek anlamını bulmuştur. Eğitimin uzaktan yapılması insanların dersanelere yâda eğitim kurumlarına gitmeden evlerinde kendi kendilerine öğrenmelerini sağlamıştır. Dünya genelinde birçok üniversitede dersler çevrimiçi video konferansları ve benzetim teknikleri ile hazırlanmış sanal laboratuvarlar ile, elektronik kitaplar, çevrimiçi tartışma platformları ile verilmekte, böylece 30–40 kişilik bir sınıf yerine binlerce kişiye aynı dersi vererek daha geniş kitlelere ulaşabilmektedir.

“Uzaktan Eğitimde Veritabanı Tasarımı ve Örnek Model” adlı çalışmada İngilizce kısaltılması LMS (Learning Management Sysyem) olan ve Türkçe’ye Eğitim-Öğretim Yönetim Sistemi (EÖYS) olarak çevirdiğimiz her çeşit eğitim-öğretim kurumunun kullanabileceği, geliştirmeye açık, istenilen ders ve yönetim içeriğinin kolayca uygulanabileceği evrensel ve tam ilişkisel bir veritabanı tasarımı yapılmıştır. Veritabanının geliştirilmesinde MS SQL Server 2005 veritabanı yönetim sistemi yazılımı kullanılmıştır. Yönetimi kolay tam ilişkisel tablolar eklenmiş ve her tablo anlamlı bir şekilde birbirleri ile ilişkilendirilmiştir.

“Uzaktan Eğitimde Veritabanı Tasarımı ve Örnek Model” adlı çalışmada EÖYS, web tabanlı öğretim teknolojileri için geliştirilmiş ve 2008 yılında dördüncü sürümü yayınlanmış olan SCORM standartlarına uygun olarak tasarlanmıştır. Buna göre yapılan çalışma uzun süre kullanılabilir, taşınabilir, kolayca yönetilebilir, erişimi kolay olduğu gibi aynı zamanda da ortak geliştirmeye olanak sağlamaktadır.

Türkiye de şimdiye kadar belirli bir ders yada uygulama modülü için birçok veritabanı tasarımı yapılmış ancak evrensel değerde bir eğitim kurumunu oluşturabilecek seviyede geniş çaplı ve açık kaynak kodlu bir çalışma henüz yayınlanmamıştır. Bu çalışmada EÖYS’nin orijinalliği, sanal eğitim kurumunun veya kurumların temelini oluşturmaktadır. Yani bu çalışma ile sanal bir eğitim kurumu kurulabilir ve istenilen ara yüz çalışması ile yayınlanabilir. Oluşturulacak EÖYS’de bir fakülte veya enstitü

tanımlanabilir ve bu kurumların akademik ve idari işlemlerinin tamamı yapılan bu sistemde yönetilebilir. Örneğin;

- Fakültenin tanımlanması,
- Dekanın ve dekan yardımcılarının yönetici olarak tanımlanması,
- Bölümlerin ve bölüm başkanlarının tanımlanması,
- Bölüm öğretim elemanlarının ve açılacak derslerin yayınlanması,
- Eğitim – öğretim akademik takvimin oluşturulması,
- Öğrencilerin fakülteye kayıt işlemlerini yaptırması (Harç parası kimlik bilgileri vs.),
- Öğrencilere kayıt yılı, fakülte, bölüm ve programlarına göre otomatik numaranın verilmesi,
- Ders notlarının ve derslerin dersin öğretim elamanı tarafından görsel ve işitsel olarak yayınlanması,
- Dersin öğretim elemanları tarafından istenirse deneme sınavlarının yapılması,
- Tartışma guruplarının oluşturulması,
- Duyuru ve ilanların her kullanıcıya otomatik olarak duyurulması,
- Ara sınavların yapılması,
- Yıl sonu sınavlarının yapılması,
- Sınavların ilan edilmesi,

Ve bunun gibi daha birçok detay düşünülerek tasarlanmış bir EÖYS veritabanı sistemidir. Bu çalışmada tasarlanan veri tabanının bir özelliği de internet uygulamaları için hazır hale getirilmiş olmasıdır. Yani bir web formu ara-yüzü ile hemen bir sanal okul kurulabilir.

Bu tez çalışmasının ikinci bölümünde uzaktan eğitim ve eğitim öğretim sistemi için ulusal ve uluslar arası alanda yapılmış çalışmalar verilmiştir. Bu bölümde de görüleceği gibi bu çalışmaya içerik ve kapsam olarak benzer bir çalışma yapılmamıştır.

Üçüncü bölümde uzaktan eğitimin temel özellikleri ve gereksinimleri hakkında literatür bilgileri verilmiş ve uzaktan eğitimin günümüz modern dünyasında ne denli gerekli olduğu vurgulanmıştır.

Dördüncü bölümde bu çalışmanın temelini oluşturan veritabanı ve veritabanı yönetim sistemlerinin tanımı, özellikleri, sağladığı üstünlükler ve kullanım şekilleri, bir veri tabanı tasarımında dikkat edilmesi gereken unsurlar ve ilkeler vurgulanmış, aynı zamanda veritabanının temelini oluşturan yapısal sorgulama dilleri anlatılmıştır.

Beşinci bölümde ise bu tez çalışmasında yapılmış olan tam ilişkisel EÖYS veritabanı tasarımı anlatılmaktadır. Tasarlanan veritabanının ilişki senaryoları hem şekilsel olarak hem de tanımlama olarak anlatılmıştır. Sistemde kullanılan her bir modül genel olarak detaylandırılmıştır.

Sonuç ve Öneriler bölümünde yapılmış olan bu çalışmanın literatüre kazandırdıkları, geliştiricilere sağlayacağı olumlu etkileri ve en önemlisi sanal eğitim öğretim sistemi için bir çekirdek model olduğu ifade edilmiştir.

2. VERİ TABANI YÖNETİMİ VE UZAKTAN ÖĞRETİM SİSTEMLERİ

Veritabanları ve veritabanı yönetim sistemleri ile ilgili konularda hem ulusal hem de uluslararası çalışmalar yapılmıştır.

Ulusal çalışmalar arasında Sürgevil (2000) “İnternet Ortamında İlişkisel Veritabanı Uygulaması” adlı çalışmasında ORACLE veritabanı PL/SQL yapısal programlama dillerini kullanarak IP tabanlı ağlar ve internet üzerinden erişilebilecek örnek bir veritabanı uygulaması geliştirmiştir. Ancak çalışma kapsam açısından çok detaylandırılmış değildir.

Onay (1988) “Tam Bileşik Veritabanı Modeli” adlı çalışmasında bir birleşik veritabanı modeli tasarlamıştır. İlişki ve yelpaze kümesi gibi iki veri yapısına sahip bu veritabanı modeli ilişkisel, sıradüzensel ve ağ veritabanı modellerinin özelliklerine sahip olmuş ve COBOL programlama dili kullanılarak bir ara-yüzle kontrolü yapılmıştır.

Şahin (1989) “Veri Tabanı Kullanan Bir Bilgi Sisteminin Geliştirilmesi” adlı çalışmasında, deniz, kara ve hava taşımacılığı yapan ARKAS firması için ilişkisel veritabanı kullanarak bir veritabanı tasarımı ve kontrol ara-yüzü tasarlamıştır.

Bekatlı (1992) “Veritabanı Tasarım Yöntemlerinin Karşılaştırılması ve Bir Uygulama” adlı çalışmasında veritabanı tasarım yöntemlerini karşılaştırmış ve matematiksel yöntemlere dayalı bir tasarım yapmıştır.

Karacabey(1997) “Veritabanı Modelleri ve Hiyerarşik Veritabanından İlişkisel Veritabanına Dönüşüm” adlı çalışmasında veritabanı modellerini incelemiş, ilişkisel veritabanı modelinin hiyerarşik veritabanı modeline göre daha üstün olduğunu görmüş ve dönüşüm için örnek bir veritabanı modeli hazırlamıştır.

Yaşar (2000) “Veritabanı Ara-yüzü Tasarımı ve Veri Erişim Metotlarının İncelenmesi ve Uygulaması” adlı çalışmasında bir veritabanı tasarımı yapılmış ve veritabanını kontrol eden bir ara-yüz bir programlama dilinde tasarlanmıştır. Bu çalışma kapsamın

da okullardaki sınav notlarının tutulması ve hesaplanması, öğrencilerin endüstri ve eğitim stajlarının tutulması, öğretim elemanlarına ait haberleşme bilgilerinin tutulması, eklenmesi, değiştirilmesi ve silinmesi ve bu bilgilerin yazıcıdan rapor olarak alınmasını sağlayan bir yazılım geliştirmiştir.

Seral (2000) “Web Teknolojileri Kullanılarak Veritabanlarına Erişim Metotlarının İncelenmesi ve Uygulanması” adlı çalışmasında MySQL ile öğrencilere ait bir not sistemi ve adres kayıt sistemi veritabanı tasarlamış ve PHP Web programlama dili ile Web sayfası formu oluşturarak veritabanı kontrolünü sağlamıştır.

Yapıcıoğlu (2001) “Designing of a Database Management System for Course Design Process in Distance Education” adlı çalışmasında uzaktan eğitimde ders tasarım süreci için bir veritabanı yönetim sistemi tasarımı yapmıştır. Bu uygulama Anadolu Üniversitesi Açık Öğretim Fakültesi’nde uzaktan eğitim dersi hazırlanmasında maliyetlerin belirlenmesi için sistematik bir yaklaşım getirmiştir.

İlgin (2004) “Nesneye Yönelik Tasarım ile Nesne-İlişkisel Veritabanı Geliştirilmesi” adlı çalışmasında nesne yönelimli veritabanı modelleri üzerinde durmuş ve örnek bir veritabanı modeli geliştirmiştir. Bu modeli Microsoft Visio yazılımı ile nesnelendirerek görsel hale getirmiştir.

Xue and Xu (2003) Eşzamanlı (Koşut zamanlı) tasarım için web tabanlı yayınlanabilir sistem ve veritabanı modeli yapmışlardır. Bu çalışmada farklı mekanlarda olan ürün veritabanlarının, sistemin, ürün kütüphanelerinin birbirleri ile bağlantılarının birleştirildiği bir web tabanında yayınlanmasını sağlamışlardır.

McIntosh ve arkadaşları (2007) ekolojistlere gözlemlerle çekirdeğin varoluş bileşimi için bir veritabanı tasarımı yapmışlardır. Tasarımlarında ilişkisel veritabanı kullanmışlardır. Oluşturmuş oldukları CnopyView aracı ile data bankalarını ve yazma araçlarını kullanarak otomatik veritabanı oluşturmuşlardır. Bu çalışma sonucunda özellikle ekolojik olarak yardımcı servisler için olumlu sonuçlar elde etmişlerdir.

Lu and Cheng (2004) java tabanlı cep telefonları için veritabanı tasarımı ve uygulamasını yapmışlardır. Veritabanını tanımlamak için XML uyarlaması yapmışlardır. Tasarlanan veritabanı hem taşınabilir hem de ortak çalışılabilir ve aynı zamanda XML, J2ME ve XSLT bilinmeden kullanılabilir. Aynı zamanda istenildiğinde geliştirilebilir bir yapıya sahiptir.

2.1 SCORM Standartları

SCORM, “Shareable Content Object Referans Model” İngilizce tümcesinin kısaltılması olan “Paylaşılabilir İçerik Nesne Referans Modeli” olarak tanımlanan bir içerik yönetim sistemi standardıdır. SCORM içeriklerin çalışması için bir iskelet oluşturur. SCORM standartları içeriğin öğretilebilir özellikleri ile değil, yayımı ile ilgilenmektedir. SCORM standartları; e-öğrenme içeriklerinin birlikte çalışılabilirlik, yeniden kullanılabilirlik, yönetilebilirlik, ulaşılabilirlik, devamlılık, ölçeklenebilirlik, sıralama ve dolaşım özellikleri üzerine getirilen standartlardan oluşmaktadır (int 3).

Bir uzaktan öğretim için içerik yönetimi tasarımı yapılırken mutlaka SCORM standartları göz önüne alınmalı ve tasarımda bu standartlar kullanılmalıdır. Eğer SCORM standartları göz önüne alınırsa yapılan çalışma uzun süre kullanılabilir, aynı zamanda yer değiştirmesi kolay, istenilen yerden erişilebilir ve aynı içeriğe birçok kez erişilebilir hale gelecektir.

SCORM standartlarını uygularken, bu standartlara ait bir takım kriterler yerine getirilmelidir. Bu kriterler (int 2);

1. **Birlikte çalışabilirlik (Interoperability):** Farklı kaynaklardan alınan içeriklerin birleştirilmesi; farklı sistemlerde çalıştırılabilmesi; farklı sistemlerin birbirleri ile iletişim kurması ve etkileşimi.
2. **Yeniden kullanılabilirlik (Re-usability):** e-Öğrenme içeriğini oluşturan bilgi nesnelerinin (metin, grafik, ses, animasyon, video, kod...) yeniden kullanılabilir olması. Bu nesnelerin bir araya getirilerek farklı bir öğrenme nesnesine dönüşebilmesi.

3. **Yönetilebilirlik (Manageability):** Kullanıcıya ya da içeriğe ait bir bilginin eğitim yönetim sistemi tarafından izlenmesi.
4. **Ulaşılabilirlik (Accessibility):** Kullanıcının bir öğrenme nesnesine ne zaman isterse ulaşabilmesi.
5. **Devamlılık (Durability):** Teknolojik bir gelişmenin; örneğin içerik üretilirken kullanılan bir aracın yeni bir sürümünün çıkmasının, yeniden tasarım ya da kodlama gerektirmemesi.
6. **Ölçeklenebilirlik (Scalability):** Teknolojinin kullanıcı sayısında, ders sayısında ya da içerikte muhtemel bir artışı kaldırabilecek nitelikte olması.

Bir uzaktan eğitim için tasarlanmış bir içerik yönetim sisteminde mutlaka SCORM standartları uygulanmalıdır. Çünkü eğitim-öğretim yönetimi içeriği öncelikle geliştirmeye açık olmalı, sürekli olmalı, yönetimi kolay olmalı ve aynı zamanda ortak çalışmaya uygun olmalıdır. Bu standartlara uygun olarak yapılmış olan uzaktan eğitim öğretim yönetim sistemleri kısa ve uzun vadede etkili ve kullanışlı olabilir.

2.2 Uzaktan Eğitim

Farklı ortamlarda bulunan öğrenci ve öğretmenlerin, öğrenme ve öğretme faaliyetlerini, iletişim teknolojileri ve posta hizmetleri ile gerçekleştirdikleri bir eğitim sistemi modelini ifade eder (İşman 2005). Uzaktan Eğitim; özel organizasyonların ve uygulamaların yapılması yanında, ayrıca özel bir ders planı yapma tekniği ve özel öğretme teknikleri olan, elektronik olan veya olmayan sistemlerin kullanıldığı, özel iletişim metotları olan normal olarak öğretme faaliyetlerini farklı ortamlarda oluşturan planlı bir öğrenmedir (Gülner 2003).

Uzaktan eğitim, eğitimi alan kişilerin herhangi bir eğitim kurumuna gitmeden yazılı dokümanlarla CD, videokasetleri, İnternet ve televizyon aracılığı ile almış oldukları eğitim tipidir.

Uzaktan eğitim programı öğrenciyle eğitsel kaynaklar arasında bağlantı kurarak eğitimi gerçekleştiren bir sistemdir. Uzaktan eğitim programlarının herhangi bir eğitim kurumuna kayıtlı bulunmayan kimselere de eğitim imkânı sağlıyor olması, son

dönemde öğrenciye tanınan eğitim imkânlarının artmakta olduğunu göstermektedir (Yalçın 2005). Uzaktan eğitim, öğretmen ve öğrencilerin farklı mekânda buluşmaları durumunda, farklı teknolojilerden yararlanarak sürdürülen eğitim etkinlikleridir. Geleneksel eğitimden en temel farkı, uzaktan eğitimde iletişim araçlarının kullanılmasıdır. Bunlar da ses, video, veri ve baskı gibi gruplara ayrılabilir.

Uzaktan eğitim konusunda yapılan tanımlara bakıldığında aslında birbirinden çok da farklı olmayan tanımlamaların olduğu dikkat çekmektedir.

2.2.1 Dünyada Uzaktan Eğitimin Gelişimi

Dünyadaki uzaktan eğitim çalışmaları yaklaşık 200 yıl öncesine dayanmaktadır. Uzaktan eğitim alanındaki ilk girişimlerden biri, 1728 yılında ABD’de Boston Eyaletinde mektup ile stenografi dersleri verilmesi olmuştur. Avusturya’daki Queensland Üniversitesi 1890’lı yıllarda açık bir eğitim programı gerçekleştirmiştir. Benzer bir programı da 1920’lerde Columbia Üniversitesi gerçekleştirmiştir. İngiltere’de Londra Üniversitesi’ni dışardan bitirmek isteyenlere yönelik düzenlenen programlar da uzaktan eğitimin öncül uygulaması olarak kabul edilmektedir. National Extension College (NEC), 1974 yılında bugünkü anlamına yönelik önemli bir adım olan Açık Üniversitenin çekirdeğini oluşturur (İşman 2001).

Almanya’da 1856 yılında başlayan uzaktan eğitim girişimleri sonraları “Tele Kolleg”, “Schulfernsehen” , “Fern Universität” ve “Deutsch Institut Für Fernstudien” gibi günümüz uzaktan eğitim kurumlarına dönüşmüştür. Fransa’da 1907 yılında atılan adımlar ise 1939 yılında resmi Uzaktan Eğitim Merkezi’nin kuruluşu sağlamıştır. Yine bu dönemlerde uzaktan eğitimin halk eğitimi boyutundaki uygulamaları Rusya’da görülür. Japonya’da ise 1948 yılında temelleri atılıp 1986 yılında “University of The Air” kurulmuştur. Sonuç olarak uzaktan eğitim etkinliklerinin, özellikle II. Dünya Savaşı sonrasında ülkelerin eğitimine katkıda bulunmak üzere gündeme geldiği ve uygulamaya girdiği görülmektedir. 1900’lü yılların ilk yarısında radyo, teyp gibi araçlar pek çok okul tarafından bir uzaktan eğitim aracı olarak kullanılmaya başlanmıştır. 1950’lerde ise Amerika’da özellikle askeri amaçlı olarak kullanılan uzaktan eğitim için doküman tabanlı iletişim ortamı kullanılmıştır. Günümüzde, teknolojik gelişmeler

sayesinde disketler, videokasetler, CD-ROM'lar, uydu yayınları, video konferanslar ve İnternet, uzaktan eğitim çalışmalarında önemli bir yer edinmiştir (Aytaç 2002).

Avrupa Birliği, resmi politikası haline gelen e-öğrenme kavramını, birliğin hızlı ve etkili eğitim ihtiyacını karşılayacak, sosyal, kültürel yakınlaşmayı ve bütünleşmeyi sağlayacak, Avrupa'yı bir bütün halinde bilgi toplumuna dönüştürecek anahtar faktör olarak görmektedir. AB tarafından kabul edilen e-Öğrenme öncelikle dört eylem alanı kabul etmiştir (Çinici 2006):

- Altyapının ve araçların iyileştirilmesi (bütün sınıflara internet erişimi, her çoklu ortam bilgisayar başına 5–15 öğrenci),
- Her seviyede eğitim hamlesi (her mezuna sayısal okuryazarlık, öğretmenlerin eğitimde sayısal teknolojileri kullanmaya teşvik edilmesi, çevrimiçi eğitim platformlarının yaratılması, okul müfredat programlarının uyumlu hale getirilmesi, bütün işgücünün sayısal okuryazarlığa erişimi),
- Kaliteli içerik ve hizmetlerin geliştirilmesi,
- Avrupa'da okulların birbirine bağlanması.

2.2.2 Türkiye'de Uzaktan Eğitimin Gelişimi

Yapılan araştırmalar göstermektedir ki gelişmekte olan Türkiye, eğitim alanındaki çağdaşlık ve uygarlık sunan gelişmeleri eğitim sistemine uygulamaktan geri kalmamıştır. Türkiye'de uzaktan eğitim çalışmaları 1924 yılında Dewey'in sunduğu "öğretmen eğitimi raporu" ile gündeme gelmiş 1927 yılında kavram olarak oluşmaya başlamıştır. Daha sonra bu çalışmalar hızlanmış ve uygulama hazırlıkları başlamıştır (İşman 2001).

Uzaktan eğitimin Türkiye'deki gelişimi 3 ana evrede incelenmektedir (İşman 2001).

- 1- Kavramsallaşma Süreci
- 2- Mektupla Öğretim Süreci
- 3- İletişim Teknolojilerinin Kullanımı Süreci

Web tabanlı uzaktan eğitim uygulamaları geliştiren üniversitelerimizden; Bilgi Üniversitesi, Sakarya Üniversitesi, Boğaziçi Üniversitesi, İstanbul Teknik Üniversitesi, Doğu Akdeniz Üniversitesi, Mersin Üniversitesi ve Fırat Üniversitesi örnek olarak verilebilir. Bu yüksek öğretim kurumlarımız bünyelerinde çeşitli sertifika ve diplomaya yönelik eğitimlerini sürdürmektedirler (Coşkun 2007).

2.2.3 Bilgisayar Tabanlı Uzaktan Eğitim Modelleri

Bilgisayar tabanlı uzaktan eğitim modeli iki ana kategoride toplanmıştır. Bunlar;

1. Çevrimiçi uzaktan eğitim
2. Çevrimdışı interaktif eğitim,

Her bir model detaylı olarak açıklanmaktadır ancak bu iki modeli birbirinden ayıran en temel özelliğin çevrimiçi ve çevrimdışı olmasıdır.

2.2.3.1 Çevrimiçi Uzaktan Eğitim

İnternet tabanlı uzaktan eğitimin kullanım standartları yönüyle belirli kısıtlamaları olmasına rağmen, günümüz şartlarında farklı ülkelerde farklı şekillerde eğitim modelleri uygulanmaktadır. Gelişmekte olan kitle iletişim teknolojileri bağlamında dünyada uzaktan eğitim sistemleri incelendiğinde, aslında tüm modellerin temelinde eğitime ihtiyacı olan ancak ekonomik, fiziksel ve zaman yetersizliği vb. nedenlerle orta ya da yüksek öğrenim olanağı bulamayan bireylere, öğrenim görebilecekleri uygun koşulları sağlama ve eğitimde fırsat eşitliği yaratma amacının var olduğu görülmektedir (Çinici 2006).

İnternet tabanlı uzaktan eğitim olarak ta adlandırılan bu eğitim modelinde öğretmen ve öğrencilerin coğrafi olarak aynı ortamda bulunmaması durumunda, uzaktan eğitim vermek üzere her türlü İnternet (web, e-posta, tartışma grupları, mesaj panoları, sohbet programları, tele-konferans) olanaklarının kullanılmasıyla verilen eğitimidir (Şahin 2005).

İnternet Tabanlı Uzaktan Eğitim, İnternet altyapısını kullanan tüm eğitim modellerini kapsayan genel bir yaklaşımdır. İnternet ağını kullanan tele-konferans görüşmeleri, geleneksel postanın yerini alan elektronik postalar, basılı kaynaklara alternatif oluşturan elektronik kitap ve süreli yayınlar, İnternet Tabanlı Uzaktan Eğitimin birer parçası olarak kullanılmaktadır. İnternet Tabanlı Uzaktan Eğitim zamandan bağımsız olma, çoklu ortam adı verilen ses, görüntü ve yazılı verilerin birlikte kullanılabilmesi, kaynaklara hızlı erişim ve kolay depolama yapabilme gibi özellikleri ile uzaktan eğitim alanında önemli bir unsur olarak hızla yaygınlaşıp kabul görmüştür (Şahin 2005).

İnternet teknolojilerinin uzaktan eğitim aracı olarak düzenli ve sistemli olarak kullanılması ancak eğitim platformlarıyla mümkün olmuştur. İlk zamanlarda eldeki İnternet teknolojileri birbirinden ayrık ve düzensiz şekilde kullanılmaktayken, artık bu teknolojilerin uzaktan eğitimin gereksinimlerini giderecek şekilde birlikte kullanılmasını sağlayan yazılımlar üretilmiştir. Bir başka deyişle, İnternet Tabanlı uzaktan eğitim platformunu, uzaktan eğitim derslerinin verilmesi ve öğrencilerin değerlendirilmesi için gereken işlevleri paket halinde sunan ve İnternet üzerinden çalışan çok gelişmiş bir bilgisayar yazılımı olarak tanımlamak mümkündür (Şahin 2005).

2.2.3.2 Çevrimiçi Uzaktan Eğitimde Modeli ve Teknolojileri

İnternet Tabanlı Uzaktan Eğitimde kullanılan teknolojiler; www, elektronik posta, dosya transfer sistemi, tartışma-haber grupları ve posta listeleri, dinleme, ses akışı ve video konferans sistemi, ortak çalışma, yardımcı yazılım programları, sanal gerçeklik ile chat odaları ve paylaşılabilen beyaz tahtalar ele alınabilir İnternet teknolojileri bu etkileşimi sağlamak için önemli olanaklar sunmaktadır. Yüz yüze eğitimin temelini oluşturan, ve uzaktan eğitimin en önemli sorunlarından olan dönüt ve etkileşim sağlandığında, İnternet Tabanlı Uzaktan Eğitim en az yüz yüze eğitim kadar etkili olacağı iddia edilmektedir (Şahin 2005).

WWW (World Wide Web): WWW İnternet temelli bir gezinti, bir enformasyon dağıtım ve yönetim sistemi olup, kitlesel ya da kişisel iletişimde dinamik bir biçimde

işlemektedir. Her geçen gün katlanarak artan İnternet üzerindeki web siteleri, milyonların üzerine çıkmıştır. Web'in en önemli özelliği multimedya olmasıdır. Grafik ortamda ses, görüntü ve yazılı bilgi alışverişinde bulunulup, web sayfalarında bulunan ve hyperlink adı verilen bağlantılarla diğer hizmetlerden de yararlanılabilmektedir. Bu yönüyle kullanım bakımından diğer İnternet hizmetlerinden çok daha kolaydır. Web sayfaları, aşağıdaki özelliklere sahiptir (Şahin 2005):

- Basımı çok kolaydır.
- Birçok animasyon eklenerek dersin anlaşılabilirliği artırılabilir.
- İşitsel, görsel ve grafiksel öğelerle zenginleştirilebilir.

Elektronik Posta (E-mail): Elektronik posta, İnternet'in en yaygın ve en kullanışlı hizmeti olup, bu ortamda istenilen kullanıcıya elektronik mektup yollanabilmektedir. İnternet kullanıcılarının önemli bir kesimi yalnızca bu hizmetten yararlanmaktadır. New York Times tarafından yayınlanan bir araştırmada, ABD'deki İnternet kullanıcılarının %77 sinin sadece bu hizmeti kullandıkları belirlenmiştir. Elektronik posta, aşağıdaki yararları sağlamaktadır (Şahin 2005):

- İnteraktif çalışma hızını artırır.
- Yüz yüze iletişimden daha rahattır.
- İletişimdeki yanlış algılamaları ortadan kaldırır.
- Sözlü ifadesi iyi olmayanlar için daha etkindir.
- Grafik, resim, yazı vb. her şey iletilebilir.

FTP (File Transfer Protocol): İnternet üzerinden dosya transferini sağlayan bu erişim tipi, web erişimi kadar sık kullanılsa da aktarılan bilginin büyüklüğü göz önünde bulundurulduğunda, band genişliğinin kullanımında dikkate değer bir orana sahiptir. Gelişen web tarayıcılarının aynı zamanda FTP istemcisi görevini de gerçekleştirmeleri sayesinde, FTP kullanımını çoklukla kullanıcı tarafından web erişiminden farksız bir biçimde gerçekleşmektedir (Şahin 2005).

Tartışma-Haber Grupları ve Posta Listeleri: İnternet ortamı, benzer konuda çalışan çeşitli mekanlardaki insanları, aynı veya farklı zamanlarda bir araya getirerek fikirlerini birbirlerine aktarma olanağı yarattığı gibi, karşıt fikirlerin tartışma listelerine dahil

edilerek tartışılmasını da sağlayabilmektedir. E-posta kişiden kişiye iletişim sağlarken haber grupları tartışma forumlarında birçok insan birbiriyle iletişimde bulunabilir. Bir mesaj haber gruplarına gönderildiğinde herkes tarafından okunur hale gelir. E-postanın tersine kişiye özel değildir (Şahin 2005).

Posta listeleri, tartışma ve haber grupları ile e-postanın bir çeşit birleşimidir. Gönderilen bütün mesajlar otomatik olarak listedeki herkese postalanır. Tartışma ve haber grupları ile posta listeleri bireylerin birbirleri ile etkileşimlerini sağlamalarına rağmen, eşzamanlı iletişimler değildir.

Dinleme, Ses Akışı ve Video Konferans Sistemi: Bu sistemde herhangi bir kişi, seyahat gücünü çekmeden, mevcut görevini fazla aksatmadan kısa bir sürede herhangi bir toplantıda görüş bildirip, karşıdakilerin yorumunu alabilmektedir (Şahin 2005).

Ortak Çalışma: İlgili kişilerle eşzamanlı ya da eşzamansız ortak çalışmalar yapılabilir. Bununla ilgili İnternette kullanılabilecek çeşitli programlar mevcuttur (Şahin 2005).

Yardımcı Yazılımlar: İnternet olanaklarından faydalanılması yönünde piyasada, birçok yazılım mevcuttur. Bunlardan bir veya birkaçını kullanarak öğrenciyi takip etme sınavlarınızı yapma, dersin güvenliğini sağlama, ders hakkında istatistikleri anında alma vb. birçok çalışmayı düzenli olarak yapma olanağı vardır. Bu anlamda, TÜBİTAK Bilten Enstitüsü tarafından hazırlanan DANS (Duy-Anla-Sorgula) ve ILINC (Interactive Learning International Corporation) şirketi tarafından hazırlanan LearnLinc programları birer örnektir (Şahin 2005).

Sanal Gerçeklik (Virtual Reality): Gerçeklik üç boyutlu iken, web sayfaları iki boyutludur. Kullanıcı dünyanın içinde olup, görüntüyü önünde görebilmektedir. Dahası kullanıcı nesnelere ve insanlar ile sanki aynı mekanda gibi iletişim kurulmaktadır. Bu yöntem, teknik sorunlar giderildiğinde, eğitim açısından tatminkar sonuçlar doğurabilecektir (Şahin 2005).

Chat Odaları, Paylaşılabilen Beyaz Tahtalar: Aynı anda izin verildiği takdirde bilgisayar ekranı başkaları tarafından da kullanılabilir. Kullanıcılardan birinin yaptığı yanlış bir diğeri tarafından düzeltilebilir veya eksik kısımlar eklenebilir. İnternet ortamı yardımıyla açıklamalar karşılıklı çok daha kolay anlaşılır (Şahin 2005).

2.2.3.3 Çevrimdışı Enteraktif Eğitim

Çevrimdışı ya da asenkron olarak da bilinen bu model, bilginin önceden üretildiği ve depolandığı, daha sonra öğrencilerin dilediği zaman ve dilediği tekrarda erişebildiği bir uzaktan eğitim şeklidir. Bu modelde herhangi bir çevrimiçi bağlantı bulunmamakta sadece CD, DVD yada diğer medya ortam dosyalarından dersleri görsel olarak izleme imkanı bulunmaktadır. Bu tür bir modelde soru sorabileceğiniz, yada tartışma ortamı oluşturabileceğiniz bir ortam bulunmamaktadır.

3. VERİTABANI

3.1 Veri (DATA)

Veri (data) kavramı yeni biçimlenmeye başlayan bir kavramdır. Üzerindeki düşünce ekolleri henüz yeterince gelişmediğinden, İngilizcede bile, birçok anlama çekilmektedir. Bu kavram, daha önce de değinildiği gibi, çeşitli açılardan başlıca şu şekillerde açıklanmaktadır:

Bilgi, bir nesne veya olayda veya bunlara ilişkin raporlarda ortaya çıkan mesaj ile ilgilidir. Bu açıdan ele alındığında, sadece kaynağın bir fonksiyonu olma özelliği taşır ve bazen veri olarak da ifade edilir. Başka bir açıdan bilgi, mesajın iletilmesini açıklayan bir kavramdır. Bu açıdan ele alındığında bilgi, mesajın iletilmesi ile ilgili ihtimal hesaplarına dayanan, belirsizliğin azaltılması için gerekli olan bir kavramdır. Bu anlamda bilgi iletişim kanalının da bir fonksiyonudur. Diğer bir açıdan ise, bilgi bir alıcı tarafından kazanılan anlam ile ilgilidir. Bu anlamdaki bilgi, hem iletişim kaynağının hem de alıcının bir fonksiyonudur. Bu görüşlerin birincisi, edebiyatta fazla benimsenmeyen bir görüştür. Genellikle veri ile bilgi arasında farklılık olduğu ve verinin bilgi elde etmeye yarayan işlenmemiş ham malzeme olduğu kabul edilir. Kişi; bilmek, öğrenmek istedikten ve veriyi kullanmaya başladıktan sonra bilgi ortaya çıkar. İkinci görüş en yaygın olanıdır. Buna göre bilgi anlamlı biçimde derlenen ve birleştirilen veridir ve şimdiki zamanda ve gelecekte verilecek kararlar için var olan gerçek bir değerdir. Bir başka ifade ile bir kaynaktan, bir alıcıya iletilen mesajın içeriğidir. Bu anlamda bilgi, karar verme ile bağlantılıdır ve dolayısıyla veriye göre daha etkin bir kavramdır. Veri kelimesinin tekil hali (datum) Latince'den gelmektedir. Sözlük anlamı “gerçek” dir. Fakat veri her zaman somut gerçekleri göstermez. Bazen, kesin değildirler veya hiç olmamış şeyleri, örneğin bir fikri tarif etmek için kullanılırlar (Çokçetin 2006).

3.2 Veri Tabanı (DATABASE)

Veritabanı, verilerin organize bir şekilde tutulmasını sağlayarak, hızlı bir erişim, etkili bir yönetim ve güncelleme sunan sistemdir. Veritabanı, mantıksal olarak birbiri ile bağlantılı olan verileri paylaşımlı bir şekilde organize ederek, ilgili organizasyondaki birçok kullanıcının farklı bilgi gereksinimlerine yanıt verir (Sürgevil 2000).

Genel olarak iki tip veritabanı mimarisi vardır. Bunlar, "merkezi" ve "dağıtık" mimaridir. Merkezi veritabanında tüm veriler, tek bir yerde toplanmıştır. Bu verilere gereksinim duyan kullanıcılar, veri iletişim olanaklarını kullanarak, merkezi veritabanına erişirler. Merkezi veritabanları, erişim, güncelleme, bakım gibi işlemler yapılırken çok büyük bir kolaylık ve denetim sağlar. Fakat herhangi bir problem anında tüm uçlar, veritabanına erişimlerini kaybeder. Dağıtık veritabanları aslında mantıksal olarak tek bir veritabanıdır, fakat fiziksel olarak farklı birimlere veya bölümlere dağıtılmışlardır. Bu dağıtım işlemi sonucunda veritabanının güvenilirliği ve kullanılabilirliği artar, veriye erişim maliyeti düşer. Ancak veri bütünlüğünü sağlamak daha zordur (Sürgevil 2000).

3.2.1 Üstünlükleri

Veritabanının uygulamalara kazandırmış olduğu üstünlükleri şöyle sıralayabiliriz (Seral 2000, Sürgevil 2000).

1. Veritabanı sistemleri verileri organize bir şekilde tutarak, bu veriler üzerinde hızlı bir erişim ve güncelleme sunar.
2. Veritabanının en büyük avantajı bu veri organizasyonudur.
3. Veri tekrarının önlenmesi, verilerin merkezi denetiminin sağlanması
4. Veri paylaşımının sağlanması
5. Paylaşım sırasında veritabanı sistemi, kullanıcının veriye erişme isteğine hızlı bir şekilde cevap verir, dolayısıyla kullanıcılara hızlı veri erişimi sağlama gibi ek avantaj sağlar.

6. Veri tekrarının en alt seviyeye indirgenmesi sonucunda, elimizdeki verilerin tutarlılığı da sağlanmış olur. Çünkü verinin değiştirilmesi gerektiğinde, sadece bir yerde yapılan değişiklik yeterli olacaktır.
7. Veritabanında bulunan merkezi kontrol sistemi güvenlik, gizlilik gibi ek destekler sağlar.
8. Uygulama geliştirme maliyetlerini ve geliştirilmiş uygulamaların bakım maliyetlerini azaltır.

Veritabanı sistemlerinin üstünlüklerinin yanı sıra, bazı dezavantajları da bulunmaktadır. En büyük dezavantaj, veritabanı alanında yetişmiş personele gereksinim duyulmasıdır. Veritabanı yazılımların geliştirilmesi, geliştirilen uygulamaların bakımı, veritabanının yönetimi, yedekleme işlemleri, veriyi işleme süreci gibi ihtiyaçlar tamamen veritabanı konusunda yetişmiş personelce yapılabilmektedir.

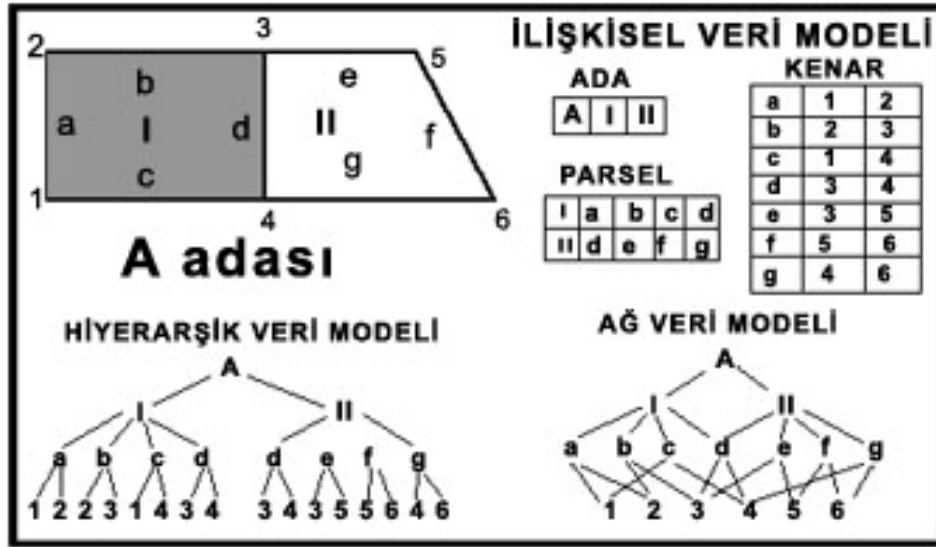
3.3 Veritabanı Modelleri ve Gelişimi

Veritabanı modellerine coğrafi veri depolama da denilmektedir. Veritabanı dünyasında, VTYS (**Veri Tabanı Yönetim Sistemi**) modelleri, verilerin nasıl saklandığını, kurulan ilişkileri, kullanılan ve yönetilen verileri temsil etmemizi olanaklı kılar. Belirli bir model üzerinde kurulu bir VTYS, verinin bu VTYS içinde organize edilme yolunu gösterir. Amaç model, tasarımcıların ve yöneticilerin bu verileri gözlemleyebilmelerine olanak sağlayacak şekilde, verileri temsil etmelidir. Daha önemlisi, veriye, performans sınırlamalarını ve bilgi gereksinimlerini karşılayacak şekilde ulaşılabilir ve ulaşılan veriler verimli bir şekilde yönetilebilmelidir (Karacabey 1997).

Dört tip veritabanı modeli bulunmaktadır. Bunlar:

1. Hiyerarşik veritabanı modeli
2. Ağ veritabanı modeli
3. İlişkisel veritabanı modeli
4. Nesne yönelimli veritabanı modeli

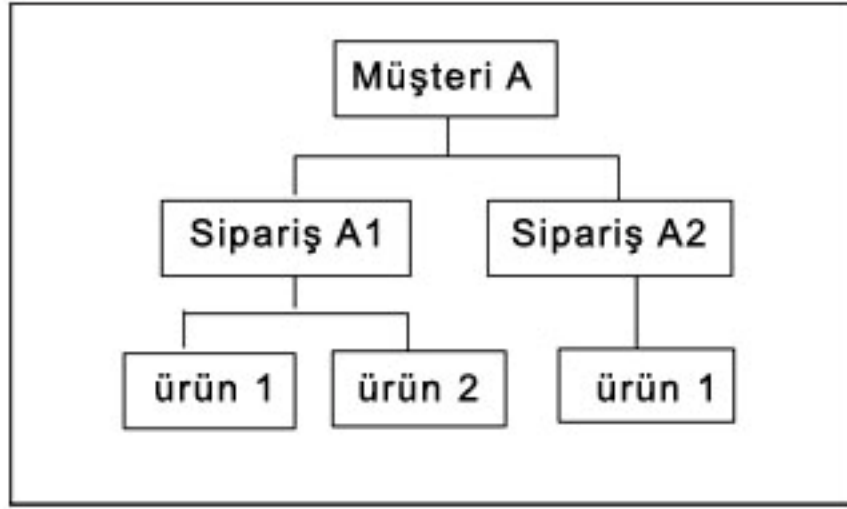
Şekil 3.1 de dört farklı veritabanı modellerinin şekilsel gösterimleri verilmiştir.



Şekil 3.1 Veritabanı modellerinin örnek üzerinde karşılaştırılması (İlgin 2004)

3.3.1 Hiyerarşik Veritabanı Modeli

Hiyerarşik veritabanı modelinde, veriler arasında bire-çok ve bire-bir ilişkiler vardır. Hiyerarşi içerisinde bir kayıt birden fazla ebeveyn kayda sahip olamaz. Aralarındaki benzerlik nedeniyle ağaç veri yapılarıyla ilgili kullanılan kök düzey ve yaprak terimleri hiyerarşik veri modelin incelenmesinde kullanılır. Hiyerarşik düzen içerisinde bir kök kayıt tipi vardır. Ona bağlı tüm düğümler ayrı kayıt tipleri olarak gösterilir. Her bir kayıt tipinin mutlaka bir ebeveyn kayıt tipi vardır. Bir ebeveyn kayıt tipinin birden fazla çocuk kaydı yâda yaprak kaydı olabilir. Aynı düzeydeki yaprak kayıtlar aynı kayıt tipi içerisinde kayıt tekrarı olarak oluşturulur. Örnek olarak müşteri sipariş ve ürün örneğini hiyerarşik model ile gösterilmek istenirse şekil 3.2 e benzer bir yapı ortaya çıkar. Burada müşteri kayıt tipi ile siparişler arasında bire sonsuz bir ilişki söz konusudur. Yani bir müşterinin sıfır veya sonsuz siparişi olabilirken herhangi bir sipariş sadece bir müşteriye ait olmaktadır. Buradan anlaşıldığı üzere verinin tekrarı söz konusudur. Örneğin ürün 1 iki ayrı siparişte yer aldığı için iki kere kaydedilmiştir (Bekathlı 1992).



Şekil 3.2 Hiyerarşik veritabanı modeline bir örnek (Bekatlı 1992)

Bir hiyerarşik model eğer bir ebeveyn düğüm birden fazla çocuk düğüme sahipse ortaya çıkar. Böyle bir yapı içerisindeki çocuk düğüm diğer herhangi bir ebeveyn düğüme bağlanabilir.

3.3.2 Ağ Veritabanı Modeli

Hiyerarşik veri tabanlarının yetersiz kalmasından dolayı, bilim adamlarının ortak çalışması sonucu, ortaya konulmuş bir veri tabanı türüdür. Ağ veri tabanları, verileri ağaçların daha da gelişmiş hali olan graflar (ağacın kendisinde özel bir graftır) şeklinde saklarlar. Bu yapı en karışık yapılardan biridir (Çokçetin 2006)

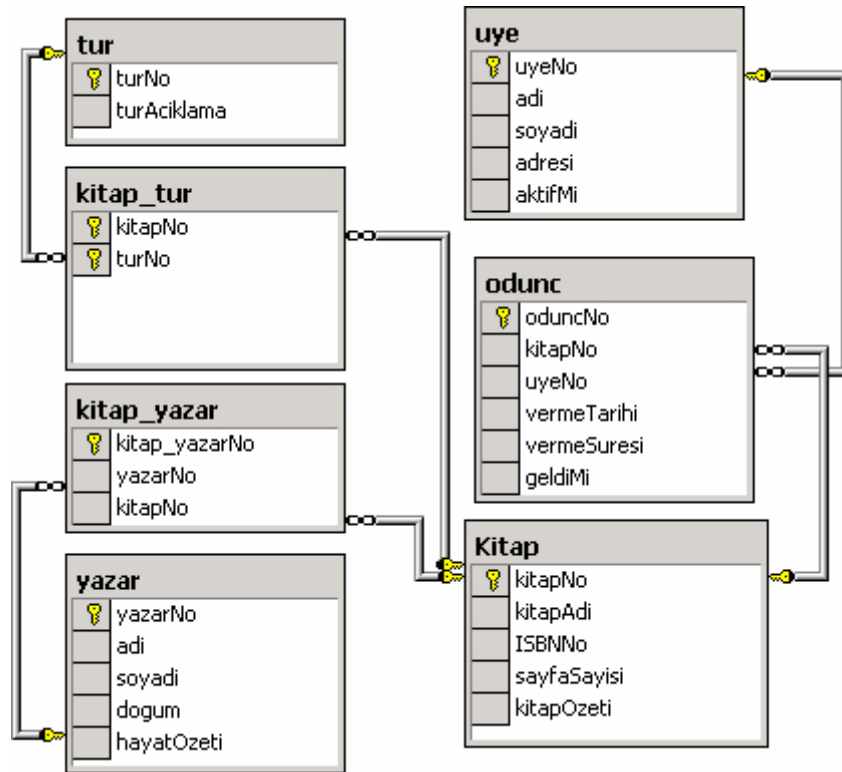
3.3.3 İlişkisel Veritabanı Modeli

İlişkisel veritabanı modeli 1970 yılında Dr. Codd tarafından ortaya atılmıştır. Bu model veritabanına kayıt edilen bilgilerin belirli kurallara uymasını sağlar. Bu modelde veriler iki boyutlu matris gibi düzenlenir. Buda satır ve sütun kavramını ortaya çıkarır (Çubukçu 1999).

Bu modelde veriler tablolar halinde depolanır. Her bir tabloya ilişki (relation) adı verilir. Tablolardaki her satır, bir detaya ait olup kayıt (record) olarak ifade edilir. Tablolardaki

sütunlar, alan (field) olarak adlandırılır. Bu sütunlarda detaylara ait grafik veriler (koordinat değerleri) ile grafik olmayan veriler (öznitelik değerleri) yer alır. Farklı tablolar arasındaki bağlantı, ortak alanlar (sütunlar) kullanılarak gerçekleştirilir. Bir tablodaki bir kayıta erişim için anahtar (key) oluşturulur ve kullanılır. Anahtar, tek bir alan olabileceği gibi, birden çok alanda da anahtar olabilir. Bir tabloda anahtar olmayan, fakat bağlantı yapılan başka bir tabloda anahtar olan bir alan yabancı anahtar (foreign key) olarak adlandırılır (İlgın 2004).

Bu veri depolama modelinde, bir ya da birden çok kayda erişim için önce tablolar birleştirilir, sonra bir yapısal sorgulama dili (SQL) ile sorgulama yapılır. Tabloların birleştirilmesi işlemi geçici olup, elde edilen bütünleşik tablo sadece sorgulama için kullanılır, yani bu tablo bilgisayar ortamında fiziksel olarak depolanmaz (Şekil 3.3). Bu modelde ilişkileri (tabloları) değiştirmek, yeni tablolar oluşturmak çok kolay olduğundan, modelin coğrafi veri depolamak için çok uygun olduğu söylenebilir (İlgın 2004).



Şekil 3.3 İlişkisel veritabanı modeli (Gözüdeli 2003)

3.3.4 Nesne Yönelimli Veritabanı

Günümüzde nesne kavramı her yerde kullanılmaktadır. Pek çok kelime işlemci ve hesap tablosu programlarının alıştığımız görünümüne artık bir de nesnelere eklenmiştir. Ancak bu gerçek anlamda bir nesneye yönelik yazılım demek değildir. Yüzde yüz nesneye yönelik bir yazılımın tamamen nesne temelli çalışması gerekir. Yazılımın mutlaka nesneye yönelik bir dilde yazılmış olması beklenir. Fakat Windows gibi işletim sistemi üzerinde çalışan yazılımlar bu özelliklere tümüyle sahip değildirler. Sadece nesne kavramını kullanarak bazı ek özellikler sunarlar. Nesneye yönelik veri tabanı da, C++ gibi nesneye dayalı bir programlama diliyle (OOPL) yazılmış olan ve yine C++ gibi nesneye dayalı (OOPL) programlama diliyle kullanılan bir veri tabanı anlamına gelir. Günümüz teknolojisinde yüzde yüz nesneye yönelik bir veri tabanı yaygın olarak kullanıma sunulmuş değildir. Ancak nesneye yönelik veri tabanlarının bazı üstünlükleri olacağından söz ediliyor. İlişkisel veri tabanları ile karşılaştırıldığında; nesneye yönelik veri tabanlarının sahip olması gereken üstünlükler şunlardır (Çokçetin 2006):

1. Nesnelere, bir tabloda yer alan bir kayıttan çok daha karmaşık yapıya sahiplerdir ve daha esnek bir yapıda çok daha kullanışlı düzenlenebilirler.
2. Nesneye dayalı bir veri tabanında, yapısı gereği arama işlemleri çok hızlı yapılabilir.
3. Özellikle büyük tablolarla uğraşırken ilişkisel veri tabanlarından çok daha hızlı sonuca ulaşırlar. Ancak çalışma mantığı tümüyle değişir.

Tüm bu özellikler, tamamen nesneye yönelik olan veri tabanları için geçerlidir. Bazı ilişkisel veri tabanları ile çalışan yazılımlarda, nesnelere bazı özelliklerini kullanırlar ama nesneye yönelik veri tabanı bunu kendini ilişkisel veri tabanı kurallarına uydurarak gerçekleştirebilir (Çokçetin 2006).

3.4 Veritabanı Tasarımı

Düzenli tasarlanmış bir veritabanı, güncel ve doğru bilgilere erişilmesini sağlar. Doğru tasarım, veritabanı çalışmalarına yönelik hedeflerin başarılmasını sağlaması açısından

son derece önemli olduğu için düzgün tasarım ilkelerini öğrenmek adına gerekli zaman yatırımının yapılması gerekmektedir. Bunun neticesinde gereksinimlerini karşılayan ve değişen koşullara kolaylıkla uyum sağlayan bir veritabanı elde edebilir. Hangi bilgilere gereksinim duyulacağına, bu bilgileri uygun tablo ve sütunlara nasıl bölüneceğine ve bu tabloların birbirleri ile nasıl ilişkilendirileceğini çok iyi bilinmesi gerekmektedir (int 3).

3.4.1 Temel Terimler

Veritabanı tasarımı yapmadan önce, veritabanı hakkında temel seviye bilgilerin olması gerekmektedir. Bu temel seviye bilgiler içerisinde veritabanı hakkında temel terimlerin de çok iyi bilinmesi gerekmektedir. Bu temel terimlerden bazıları şunlardır;

- Tablo
- Kayıt (Satır)
- Alan (Sütun)
- Birincil Anahtar
- Yabancı Anahtar (Foreign Key)
- İlişki Kurma (1-1, 1-∞)
- SQL (Yapısal Sorgulama Dili)
- Sorgu Oluşturma
- Değişken Tipleri (Int, Char, Varchar, Bool, Double, Float, Text, DateTime ve diğerleri)
- Ara-Yüz Formu

Veritabanı tasarımı temel seviye bilgisayar yapı bilgisi, dosyalama ve işletim sistemi bilgisi gerektirmektedir. Eğer bu temel seviye bilgiler yok ise tasarlanan veritabanı sağlıklı çalışmayabilir.

3.4.2 Doğru Planlama ve Tasarım Nedir?

Doğru planlanmış ve tasarlanmış her veritabanı belirli standart ilkelere ve kurallara dayanmaktadır. En basit şekli ile veritabanındaki bilgiler kesinlikle tekrar etmemesi gerekmektedir. Aynı zamanda veritabanına girilen bilgilerin türü de önemlidir ve tanımlanan alanda çok önemlidir. Uzman bir tasarımcı veritabanında kullanacağı yer

kadar alan ayırmalıdır. Tanımlanan alanlar girilen veri türü için ayrılmalı ve gerekli işlemler için ona göre kullanılmalıdır. Doğru planlanmış ve tasarlanmış bir veri tabanı hızlı, sorunsuz işlemesi kolay bir yapıda olur ve ayrıca zamandan kazandırır, yazılımcının işini kolaylaştırır. Aksi halde hatalar çok olur ve düzeltme için zaman harcanır, yazılımcının tasarımı zorlaşır.

3.4.3 Tasarıma giriş

Veritabanı tasarımına başlarken ilk işlem proje içerisindeki nesnelerin teker teker tanımlanması gerekmektedir. Örnek vermek gerekirse;

Kütüphane sistemi: Kitap, üyeler, türler, ödünç hareketleri

E-ticaret sistemi: Ürünler, müşteriler, siparişler, teslimat, fatura ilgileri, üreticiler, tedarikçiler, dağıtıcılar...

Futbol Ligi: Takımlar, sahalar, oyuncular, fikstür, hakemler, antrenörler

Okul Sistemi: Öğrenciler, öğretmenler, dersler, derslikler

Sözlük: Kelimeler, anlamlar, diller

Her bir nesne için bir tablo oluşturulmalı ve o nesneyi anlatan en iyi isim verilmelidir. Bu işlemler bilgisayar ortamına aktarılmadan kâğıt üzerinde olabilecek değişiklikler önceden görülebilir.

3.4.4 Hedefin Belirlenmesi

Veritabanı tasarımına başlamadan önce hedefin mutlaka belirlenmesi gerekmektedir. Hedef gerekirse bir kağıt üzerine aktarılmalı ve sesli düşünerek, amaçları ayrı ayrı listelenmelidir. Amaçlar belirlendikten sonra bu amaç içerisinde bulunması gereken tüm nesneler belirlenmeli ve ayrıntılara dikkat edilmelidir. Örneğin bir kütüphane yazılımı için veritabanı tasarımı yapılacak olsun.

Hedef: Kütüphane kitap kayıtların tutulacağı aynı zamanda ödünç verme sisteminin de takibinin yapılması istenen bir veritabanı tasarımı yapılacaktır.

Nesneler: Kitaplar, Yazarlar, Kitap Türleri, Üye kayıt, Ödünç Verme, vs.

3.4.5 Verileri Toplama ve Düzenleme

Veritabanının tasarımına geçmeden önce proje için gerekli olan bilgiler ve veriler toplanır yani önce nesnelere belirlenir ve daha sonra bu nesnelere ait bilgiler listelenir. Veritabanında nesnelere birer tablo olarak düşünülürse, nesnelere ait bilgiler ise o tablonun sütunları olarak düşünülebilir.

Verilerin toplanması aşamasında eksiksiz bir tasarım yapmak mümkün olmayabilir ancak belirlenen yol takip edildiği sürece en sonunda uzman bir veritabanı elde edilecektir.

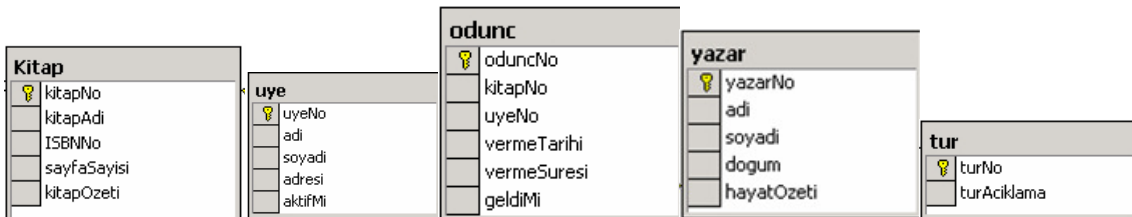
Verilerin toplanması esnasında çıktı olarak istenen raporların listelerinin yapılması ve raporlamada kullanılacak nesnelere ve nesne elemanlarının da belirlenmesi gerekmektedir.

3.4.6 Nesnelere Tablolara Aktarma

Tasarımın en önemli kısımlarından biride verilerin tablolara bölünmesidir. Burada veritabanı tasarımı için oluşturulan nesnelere her biri tablo olarak tanımlanır. Örnek vermek gerekirse;

Veritabanı adı : Kutuphane

Tabloları (Nesneler); Kitap, Üye, Ödünç, Yazar ve Tür (Şekil 3.4)



Şekil 3.4 Veritabanı tablolarının oluşturulması (Gözüdeli, 2003)

Şekil 3.4’de de görüldüğü gibi veritabanına ait her bir nesne tablo olarak tanımlanır ve veritabanına eklenir. Başlangıç için oluşturulan mevcut tablolar yeterlidir. Ancak ilerleyen durumda bir ilişki oluşturabilmek için bu tablolar yetersiz kalacaktır.

3.4.7 Tablo Verilerini Sütunlara Aktarma

Tasarımda ikinci önemli kısım ise nesnelere ait verilerin tablolara aktarılmasıdır. Verilerin tablolara aktarılmasında her bir alan ismi birbirinden farklı olmalı, verilen ad içeriğini doğru bir şekilde ifade etmeli ve o veri türü alana doğru tanımlanmalıdır. Şekil 3.4 deki kütüphane sistemine göre “Kitap” tablosu için düşünülürse “KitapNo” alanı “Metin” türünde bir alan türü olamaz yada “KitapOzeti” alanı “Varchar” olarak tanımlanan sınırlı sayıda karakter içeren bir veri türü olamaz. Veri türlerinin tanımlanmasına mutlaka özen gösterilmelidir.

Her bir tablodaki veriler detaylı bir şekilde listelendirilmeli ve her bir tablodaki bir alan diğerlerinden benzersiz sayısal ifade içermelidir. Örneğin TC kimlik numarası veya artan sıra numarası gibi.

Şekil 3.4’de verilen “Kütüphane” projesinin “Kitap” tablosu için olabilecek sütunsal veriler;

1. KitapNo
2. KitapAdi
3. ISBNNo
4. SayfaSayisi
5. Ozet

Olarak tanımlanabilir. Burada kütüphanedeki her bir kitap sıra ise sisteme eklenir ve her bir kitabın bir tane numarası olmaktadır. Burada ilk alan “Birincil Anahtar” olarak tanımlanmıştır. Türü int, özelliği her bir kayıt girildiğinde otomatik artan sayıdır. KitapAdı: “Varchar”, ISBNNo: “Vvarchar”, SayfaSayisi: “int”, Özet ise “Text” olarak

tanımlanmıştır. Örnek projenin diğer alanlarına ait tablolar ve verileri şekil 3.4’de verilmiştir.

3.4.8 Birincil Anahtar Belirleme

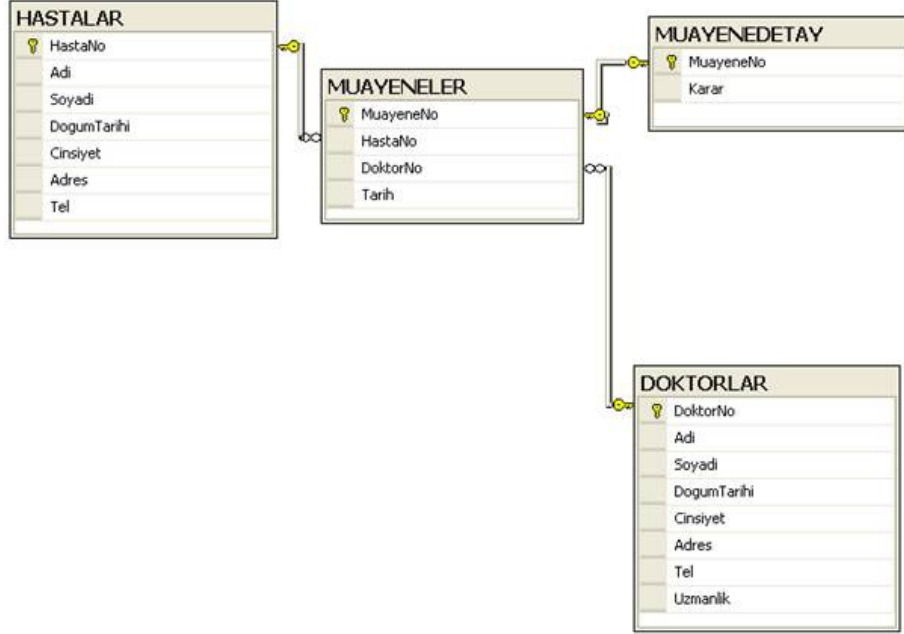
Veritabanındaki her bir tabloda saklanan her bir veri satırı, bir-biri ile aynı olmayan, benzersiz bir şekilde tanımlanmış bir sütun ile belirtilmelidir. Örneğin KitapNo, SeriNo, FiksturNo, SiparisNo gibi her bir satırdaki veri birbirinden farklıdır, genelde artan değer olarak tanımlanır. Veritabanı terminolojisinde bu tür alan özelliğine “Birincil Anahtar” denir. Birincil Anahtar alanları veritabanındaki tabloların ilişkilerinin oluşturulmasında kullanılır.

Kütüphanedeki her bir kitabı benzersiz olarak tanımlayan kitap numarası tabloda zaten varsa, bu tanımlayıcıyı numara tablonun birincil anahtarı olarak kullanılır. Ancak bu sütundaki değerlerin her zaman, her kayıt için farklı olması gerekir. Birincil anahtarda, yinelenen değerler olamaz. Örneğin, kitap adları benzersiz olmadığından kitap adları birincil anahtar olarak kullanılmaz. Aynı adda iki kitabın tabloda yer alma olasılığı vardır.

Birincil anahtarda her zaman bir değer olması gerekir. Bazı noktalarda sütun değeri atanmamış veya tanımsız ise (eksik değer), birincil anahtarda bileşen olarak kullanılamaz. Her zaman değeri değişmeyen bir birincil anahtarın seçilmesi gerekir. Birden fazla tablonun kullanıldığı veritabanında, bir tablonun birincil anahtarı diğer tablolarda başvuru olarak kullanılabilir. Birincil anahtar değişirse, anahtarın başvuru olarak alındığı her yerde de değişikliğin uygulanması gerekir. Değişmeyecek birincil anahtar kullanılması, birincil anahtarın, kendisine başvuran diğer tablolarla uyumsuzlaşma olasılığını azaltır. Çoğunlukla, rasgele benzersiz bir sayı birincil anahtar olarak kullanılır. Örneğin, her üyeye benzersiz bir üye numarası atanmalıdır. Üye numarasının tek amacı üyeyi tanımlamaktır. Üye numarası atandıktan sonra, bir daha değişmez. İyi bir birincil anahtar olarak genelde otomatik sayı veri türüne sahip sütun kullanılmalıdır. Otomatik Sayı veri türü kullanıldığında, veritabanı otomatik olarak yeni ve öncekinden bir artan değer atar (int 3).

4.4.9 Tablolar Arasında İlişki Kurma

Bilgiler tablolara aktarıldıktan sonra, amaç doğrultusunda anlamlı bir birleştirme yapmak için tabloların ilişkilendirilmesi gerekmektedir.



Şekil 3.5 İlişkilendirilmiş veritabanı modeli (int 4).

Veritabanı yönetim sistemlerinde farklı şekillerde veritabanı ilişkilendirme işlemleri bulunmaktadır. Bunlar;

1. Bire-Sonsuz (1-∞) ilişki,
2. Sonsuz-Sonsuz (∞-∞) ilişki,
3. Bire-Bir (1-1) ilişki

3.4.9.1 Bire-Sonsuz (1-∞) ilişki,

Şekil 3,5’de verilen ilişkilendirilmiş veritabanı modeline göre bir hastane ve muayene işlemi ele alındığında şu veriler bilinmeli ve ayırt edilmelidir;

1. Bir hastaneden her zaman için aynı ad ve soyada ait tek bir doktor olmalı;
2. Her zaman için bir aynı ada ve soyada ait tek bir hasta olmalı;

3. Muayene işlemi, bir hasta için birden fazla olabilir;
4. Ancak her muayene için bir karar olması gerekir.

Hasta bilgileri “Hastalar” adında bir tabloda toplanır ve her bir hasta kaydı ayrı ayrı tutulur, bir hastanın iki tane kaydı olamaz.

Doktor bilgileri “Doktorlar” adında bir tabloda toplanır ve her bir doktorun kaydı da ayrı ayrı tutulur. Aynı doktorun iki kaydı olamaz.

Ancak muayene işlemlerinde, her bir muayene ayrı ayrı kaydedilmeli ve her bir muayeneye bir numara verilerek kayıtları tutulmalıdır. Aynı hasta birkaç kez muayeneye gelebilir ancak her bir muayene işlemi için muayene numarası artmalı ve hastanın kişisel bilgileri her zaman aynı kalmalıdır. Aynı işlem doktorlar içinde geçerlidir.

Bu durumda “Muayeneler” tablosuna “HastaNo” diye bir alan tanımlanır ve Hastalar tablosu ile Bire-Sonsuz bir ilişki kurulur. Yani muayene sırasında hastanın sadece numarası çağrılır ve o hasta olduğu bilinir. Aynı şekilde “DoktorNo” diye bir sayısal alan eklenerek “Doktorlar” tablosu ile Bire-Sonsuz bir ilişkilendirilme kurulur. Muayenenin hangi doktor tarafından yapıldığı sadece doktor no alınarak bilinmiş olur.

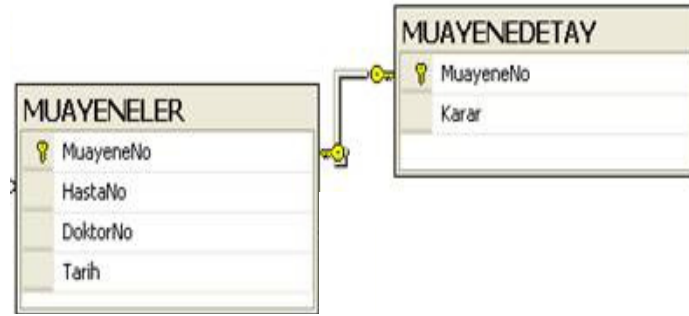
3.4.9.2 Sonsuz-Sonsuz ($\infty-\infty$) ilişki,

Şekil 3.5’de verilen ilişkilendirilmiş veritabanı modeline göre bir hastane ve muayene işlemi ele alındığında sonsuz-sonsuz ilişki için şu söylenebilir. Her zaman için hastalar ve doktorlar arasında sonsuz-sonsuz ilişki vardır. Yani bir doktor, bir hastayı birden fazla muayene edebilir yada bir hasta birden fazla doktora gidebilir, burada her bir muayene işlemi için doktor ve hasta bilgilerinin tekrar-tekrar girilmesi veritabanı tasarım mantığına terstir. Hastalar ve doktorlar arasındaki sonsuz-sonsuz ilişki ancak, araya yerleştirilebilecek ikinci bir tablo ile düzeltilebilir. Şekil 3.5’de de görüldüğü gibi aradaki “Muayeneler” tablosu ve bu tablo ile olan bire-sonsuz ilişki ortadaki problemi

çözmüştür. Veritabanı tasarımında pratikte sonsuz-sonsuz ilişki kurulumu mümkün değildir. Böyle sorunlar ancak üçüncü bir tablo ile çözülebilir.

3.4.9.3 Bire-Bir (1-1) ilişki,

Bu ilişki türünde bir tablodaki verilerin bir kısım detayının farklı bir tabloda tutulması istendiğinde kullanılan bir ilişki türüdür. Şekil 3.6’da muayene detayları farklı bir tabloya alınmıştır, sebebi ise bazı muayeneler karar gerektirebilir, eğer karar tablosu “Muayeneler” tablosu içerisinde olsa idi karar alınmayan bir muayene içinde veritabanında yer işgal edecekti. Ancak “MuayeneDetay” tablosu oluşturularak ve birincil anahtar olarak ta MuayeneNo tanımlandığı için Bire-Bir ilişki tanımlanmış olur. Böylece veritabanı için alan kazanılmış olur.



Şekil 3.6 Bire-Bir (1-1) ilişkilendirilmiş tablolar

3.4.10 Detaylandırma

Veritabanı için oluşturulan tablolar, alanlar ve ilişkiler tamamlandıktan sonra deneme aşamasına geçilmeli ve içeriği doldurulmalıdır. Şekil 3.5 için düşünülürse hasta kayıtları, doktor kayıtları girilmeli ve muayene işlemlerine başlanmalıdır. Daha sonra veritabanı yönetim sistemi içerisinde sorgulamalar yapılmalı, kayıt ekleme, silme işlemleri yapılmalıdır. Böylece unutulmuş bazı ayrıntıların olup olmadığına bakılmalıdır. Bazı durumlarda veri tekrarını önlemek için tabloları bölme zorunluluğu doğabilir, bu durum deneme aşamasında göze çaracaktır. Gerekli tablo ekleme, yeni ilişkilendirme işlemleri yapılarak alt detaylama işlemi de tamamlanmalıdır.

3.4.11 Normalizasyon İlkeleri ve Uygunluğun Denetlenmesi

İyi bir veritabanı tasarımı yapabilmek için yeteneğin, bilginin ve en önemlisi tecrübenin kullanılması gerekir. Bu iş için evrensel geçerliliği olan metotlar maalesef yoktur. Ancak iyi bir tasarım için belirlenmiş 5 normalizasyon kuralı vardır. Bunlar:

1. Bir satırdaki bir alan yalnızca bir tek bilgi içerebilir. Örneğin kitap tablosunda, birden fazla yazarı olan kitap için yazar1, yazar2, yazar3 diye alanlar açılıyorsa, bu kurala uyulmamış olurdu. Böyle bir durumda, ayrıca yazarlar tablosu da oluşturularak kural çiğnenmemiş olur. Verileri virgül veya bir başka karakter ile ayırıp aynı alana girmek, daha sonra program içerisinde split ile bu değerleri ayırmak ilişkisel veritabanının doğasına terstir (Gözüdeli 2003).
2. Bir tablo için, anahtar olmayan her alan, birincil anahtar olarak tanımlı tüm alanlara bağlı olmak zorundadır. Mesela, Ödünç tablosunda KitapAdi diye bir alan eklense idi, bu sadece ödünç verilen kitap ile ilgili bir bilgi olacaktı ve oduncNo'na bağlı bir nitelik olmayacaktı. Bunu çözmek için, kitap adlarını ayrı bir tabloda tutarak sorun çözülür. Ya da anahtar alanın birden fazla alandan oluştuğu tablolarda, anahtar alanlardan sadece birine bağlı veriler, tabloda yer almamalı, ayrı bir tabloya taşınmalıdır. Bunun tersi de geçerlidir. Yani iki ya da daha fazla tablonun birincil anahtarı aynı olamaz. Şayet böyle ise, bu iki tablo tek tabloya indirilmelidir (Gözüdeli 2003).
3. Bir tablo için, anahtarı olmayan bir alan, anahtarı olmayan başka hiç bir alana bağlı olamaz. Örneğin, kitaplar için cilt tipi adında bir alan ekleyip burada da karton kapak için K, deri cilt için D, spiral cilt için S yazılıyorsa, bu kodlama, kitap tablosunun birincil anahtarı olan “kitapNo” alanına bağlı bir kodlama olmayacaktı. Çünkü bu kodlama bir başka anahtarı olmayan alana bağlıdır. Bunun sonucunda da veritabanında, karşılığı olmayan bir kodlama yer almış olur. Cilt tipi bilgisini kodlu olarak tutan alan aslında cilt tipi açıklaması olan başka bir alana bağlıdır. Bu ilişki başka bir tabloda tutulmalıdır. Bu durumda, “ciltSekli” adında bir tablo açılması gerekir. Bu tablonun alanları da

“ciltTipKodu” ve “ciltSekli” olmalıdır. Ancak bundan sonra, kitaplar tablosunda ciltTipi adında bir sütun açıp buraya da D,S,K gibi kodları yazılabilir (Gözüdeli 2003).

4. Birincil anahtar alanlar ile anahtarı olmayan alanlar arasında, birden fazla bağımsız bire-çok ilişkisine izin verilmez. Örneğin, tabloda yer alan bir kitap hem hikaye kitabı hem de kişisel gelişim kitabı olabilir.(Bu durumda kitabın adı, kişisel gelişim hikayeleri olurdu her halde) Bu durum Kitap tablosunda nasıl ifade edilir? Dördüncü Normalizasyon kuralını sağlamak için, her bağımsız bire çok ilişki için ayrı bir tablo oluşturmak gerekir. Bu örnekte, türler diye bir tablo açılması gerekir. Daha sonra “kitapTurleri” diye bir başka tablo daha açılması gerekiyor. ‘Kişisel Gelişim Hikayeleri’ adlı kitap için, öncelikle kitap numarası, Hikaye bölümünün kodunun yer aldığı bir satır; ardından da yine kitap numarası, ardından da kişisel gelişim türünün kodunun aldığı yeni bir satır daha eklemek gerekir (Gözüdeli 2003).
5. Tekrarlamaları ortadan kaldırmak için her bir tabloyu mümkün olduğunca küçük parçalara bölmek gerekir. Aslında ilk dört kural sonuçta bu işe yarar ancak, bu kurallar kapsamında olmayan tekrarlamalar da beşinci normalizasyon kuralı ile giderilir (Gözüdeli 2003).

Örneğin, kitaplar için bir edinme şekli bilgisi girilecek sütun eklemek istensin: Bu bölüme girilebilecek bilgiler bellidir: Bağış veya satın alma. Bu bilgiler başka bir tabloda tutulabilir. Böylelikle, kullanıcıların bu alana gelişi güzel bilgiler girmesi engellenmiş olur. Bu da sorgulama esnasında veriler arasında bir tutarlılık sağlanır. Bu işlem sonucunda, tutarsızlıklara neden olabilecek ve sık tekrarlayan veriler başka bir tabloya taşınmış olur. Bu tablo için, veritabanı programlamada ‘look-up table’ terimi kullanılır (Gözüdeli 2003).

Ancak, veritabanı normalizasyon kuralları, bir ilişkisel veritabanının tasarlanma aşamalarını değil de ilişkisel veritabanında yer alacak kayıtların ilişkisel

veritabanı ile uyumlu olup olmadığını denetlemeye yöneliktir. Özetle ilişkisel bir veritabanı tasarımı şu dört öğeyi barındırmalıdır (Gözüdeli 2003).

- a. Veri tekrarı yapılmamalıdır.
- b. Boş yer mümkün olduğunca az olmalıdır.
- c. Veri bütünlüğü sağlanmalıdır.
- d. Veriler, aralarında bir ilişki tanımlanmaya müsait olmalıdır.

Veritabanındaki tablolarınızın doğru yapılandırılıp yapılandırılmadığını görmek için bu kurallar kullanır. Tüm bilgi öğelerini gösterdikten ve başlangıç tasarımına ulaştıktan sonra normalleştirme en kullanışlı öğedir. Buradaki yaklaşım, bilgileri uygun tablolara bölmek amacıyla tasarımcıya yardımcı olmaktır. Ancak normalleştirme, başlangıç için tüm doğru bilgilere sahip olmanızı sağlayamaz (int 3).

3.5 Veritabanı Yönetim Sistemleri

Veritabanı Yönetim sistemlerinden günümüzde kullanımı en yaygın olan ilişkisel veritabanıdır ve en yaygın veritabanı yönetim sistemleri, ilişkisel Veritabanı Yönetim Sistemleri ' (VTYS) dir. İlişkisel veritabanının en önemli yanı, tablolardan oluşmasıdır. Daha önemli yanı da bu tabloların birbiri ile ilişkilerinin olmasıdır. VTYS 'lere "ilişkisel" denmesinin anlamı budur (Burma 2005).

Bir İlişkisel Veritabanı Yönetim Sistemi tablolar üstünde şu üç işlevi yerine getirmek zorundadır (Burma 2005).

1. **Seçme:** Herhangi bir tabloda (listede) yer alan tüm bilgileri gösterebilmelidir. Örneğin, Kitap tablosunun bir dökümünü verebilmelidir ya da kitap listesinden bazı kitapların bilgilerini getirip diğer bir kısmını getirmeyebilmelidir.
2. **İzdüşürme:** Herhangi bir tablodan sadece belli sütunların yer aldığı seçme işlevlerini yerine getirebilmelidir. Örneğin, canı isteyen bir kullanıcı kitabın sadece adını ve kaç sayfa olduğunu seçebilmelidir.

- 3. Birleřtirme:** Birden fazla tabloda yer alan bilgiler, yeri geldiğinde tek bir tabloyuř gibi sunulabilmelidir. Örneđin, ödünç alınıp da geri getirilmeyen kitapların adları ve kimler tarafından alındığı bir tek tabloyuř gibi gösterilebilmelidir.

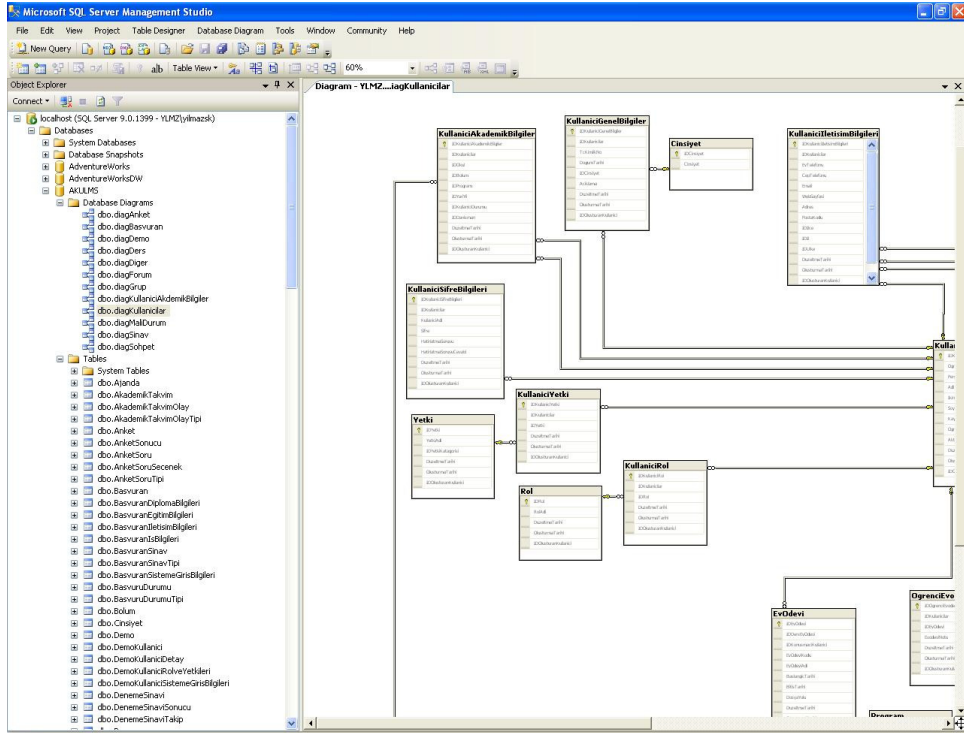
Günümüzde farklı niteliklerde iliřkisel veritabanı yönetim sistemleri bulunmaktadır. Bunlardan MYSQL ücretsiz olarak kullanıcılara sunulmakta ancak yönetimi komut satırları ile yapılmaktadır. Microsoft firmasının çıkarmıř olduđu Office gurubu içerisinde gelen Access veritabanı yönetim sistemi yazılımı ev, ofis ve orta ölçekli uygulamalar için ideal bir yazılımdır. Yine Microsoft firmasının çıkarmıř olduđu büyük ölçekli işletmelere uygun MS SQL Server veritabanı yönetim sistemi yazılımı bulunmaktadır. Yine MS SQL Server seviyesinde ve hatta daha profesyonel nitelikte olan ORACLE yine bir veritabanı ve veritabanı yönetim sistemidir.

Bütün bu ticari yazılımlarla istenilen şekilde veritabanı oluşturulabilir ve yönetimi yapılabilir. Ancak tam profesyonel veritabanı kontrolü için mutlaka Yapısal Sorgulama Dili (SQL) komutlarının iyi bilinmesi gerekmektedir.

3.6 MS SQL Server 2005

Microsoft SQL Server veritabanı yönetimi yazılımı veritabanı oluşturmak ve yönetmek için kullanılan bir yazılımdır. Bu yazılım orta ve büyük ölçekleri işletmelerin verilerinin saklanması ve yönetilmesinde kullanılmaktadır. MS SQL Server yazılımında veriler hafızada fiziksel dosyalar içerisinde tutulmaktadır. SQL Server verileri ve veriler ile ilgili günlükleri (log) tutan iki dosyadan oluşmaktadır. SQL Server'da verilerin tutulduđu dosyaların uzantısı *.mdf günlüklerin tutulduđu dosyaların uzantısı ise *.ldf dir. VTYS açısından SQL Server yazılımı incelendiğinde sadece iki adet fiziksel dosyadan oluşmadığı, yapısal olarak hafıza yönetim modülleri, tablolar, fonksiyonlar, kullanıcılar, prosedürler ve tetikleyiciler gibi birçok işleçleri bir arada bulunduđu görülmektedir. SQL Server veri yönetmede ve iliřkisel tasarım yapmada çok profesyonel çözümler sunmaktadır.

Şekil 3.7’de görüldüğü gibi veritabanı görsel olarak yönetilebilmektedir. Yani yeni bir veritabanı oluşturma, silme, veri yabanını yedekleme yada geri yükleme, veri tabanına kullanıcıların atanması, tabloların oluşturulması, ilişki diyagramlarının oluşturulması, değişkenlerin atanması ve görüntülerin oluşturulması gibi birçok işlem komut yerine görsel olarak yapılabilir. MS SQL Server yazılımının kendi veritabanı sistemi için geliştirmiş olduğu T-SQL dili ile de bütün görsel işlemler komutlar ile de yapılabilir.



Şekil 3.7 MS SQL Server yönetim paneli görüntüsü

Her bir veritabanında olması gereken birtakım temel veri tabanı nesnelere sahiptir. Bu veritabanı nesnelere verilerin saklanması ve işlenmesinde kullanılır. Bu nesnelere hem işlemleri hızlandırır hem de kullanıcının işini kolaylaştırır. Bütün veritabanları benzer nesnelere sahiptirler ancak gelişmiş yapıda olanlarda nesne ve nesnelere işlevleri daha fazla olmaktadır. Örneğin MS Access’te İmleç (Cursor), tetikleyiciler (trigger), ve rol tanımlama gibi birçok nesne bulunmamaktadır. MS SQL Server, ORACLE gibi geniş tabanlı VTYS’lerinde bu nesnelere birlikte birçok yeni nesnelere bulunmaktadır.

Şekil 3.8’de MS SQL Serverda bir veritabanında bulunan temel nesnelere ağaç görünümü şeklinde verilmiştir. Bunlar;

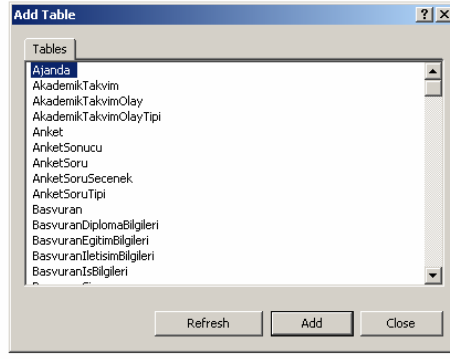
1. Veritabanı diyagramları (Database Diagrams)
2. Tablolar (Tables)
3. Görünümler (Views)
4. Senonimler (Synonyms)
5. Programlanabilirlik (Programmability)
6. Servis Habercisi (Service Broker)
7. Saklama (Storage)
8. Güvenlik (Security)



Şekil 3.8 MS SQL Serverda bir veritabanında bulunan temel nesnelere

3.6.1 Veritabanı diyagramları (Database Diagrams)

Oluşturulmuş olan veritabanında tablolar arasında bire-bir ve bire-sonsuz ilişkilerinin oluşturulduğu bölümdür. Bu nesne üzerine farenin sağ tuşuna tıklandıktan sonra yeni veritabanı diyagramı oluştur (New Database Diagram) denir. Daha sonra önceden hazırlanmış tablolar gelir (Şekil 3.9). Buradan ilişki kurulacak olan tablolara seçilir ve ilişkilendirilme oluşturulur.



Şekil 3.9 Yeni veritabanı diyagramı oluşturma penceresi

3.6.2 Tablolar (Tables)

Tablolar veritabanının temelini oluşturan nesnelerdir. Değişkenlerin tanımlandığı sütunlardan oluşmaktadır. Şekil 3.10'da oluşturulan kullanıcılar tablosunun değişken (sütun adları) adları ve değişken tiplerinin tanımları görülmektedir.

	Column Name	Data Type	Allow Nulls
?	IDKullanıcılar	int	<input type="checkbox"/>
	OgrenciNumarasi	nvarchar(11)	<input checked="" type="checkbox"/>
	PersonelNumarasi	nvarchar(6)	<input checked="" type="checkbox"/>
	Adi	nvarchar(25)	<input type="checkbox"/>
	IkinciAdi	nvarchar(25)	<input checked="" type="checkbox"/>
	Soyadi	nvarchar(25)	<input type="checkbox"/>
	KayitTarihi	datetime	<input checked="" type="checkbox"/>
	Ogrenci_Mi	bit	<input type="checkbox"/>
	Aktif_Mi	bit	<input type="checkbox"/>
	DuzeltmeTarihi	datetime	<input checked="" type="checkbox"/>
	OlusturmaTarihi	datetime	<input type="checkbox"/>
	IDOlusturanKullanici	int	<input type="checkbox"/>

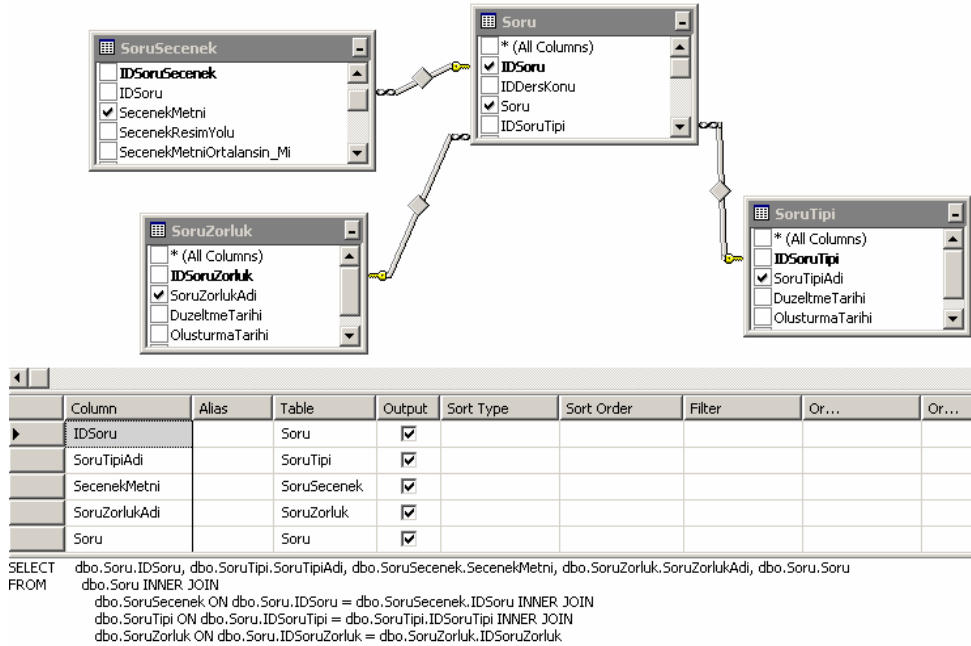
Şekil 3.10 MS SQL Server tablo oluşturma ara-yüzü

Tabloda değişkenler ifade edilirken Türkçe karakter ve boşluk karakteri kullanılmamalı, değişkenin türü mutlaka doğru belirtilmeli ve değişkenin alacağı alan kadar yer ayrılmalıdır. Tablo diğer tablolarla ilişkilendirecekse anahtar alan tanımlamasının

yapılmasına dikkat edilmelidir. Veri alanların boş (Null) değer alıp almayacağını da iyi belirlemek gerekmektedir. Örneğin anahtar alanlar kesinlikle boş alan olamazlar.

3.6.3 Görünümler (Views)

MS SQL Serverda görünüm, bir birleri ile ilişki içerisinde olan farklı tabloların verilerinin birleştirilerek kullanıcıya anlamlı bir bütün halinde sunulmasını sağlayan bir nesnedir. İlişkilendirilmiş çok sayıda tablodaki verilerin birleştirilerek kullanıcıya gösterilmesinde kullanılır. Şekil 3.11’de Soru tablosu “SoruSecenek”, “SoruZorluk” ve “SoruTipi” tabloları ile ilişkilendirilmiş olduğu görülmektedir. Soru tablosu diğer tablolar ile ilişkilendirilirken sadece sayısal kodlar ile ilişkilendirilmiş ancak kullanıcıya ilişkilendirilen sayı kodları yansıtılmamış kodun değeri olan metinsel ifade yansıtılmıştır.



Şekil 3.11 MS SQL Serverda görünüm oluşturma

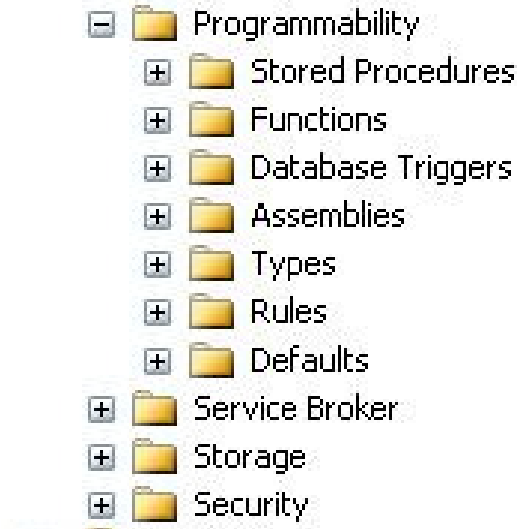
3.6.4 Senonimler (Synonyms)

Senonimler görünümlerden farklı olarak tablolar dışındaki saklı prosedürler (Stored Procedure) gibi nesnelere takma ad verilmesine olanak sağlayan bir nesnedir (Gözüdeli 2007).

3.6.5 Programlanabilirlik (Programmability)

MS SQL Server yazılımında veri tabanı nesnelere bir olan programlanabilirlik verilerin kaydedilmesi, silinmesi, güncelleştirilmesi, karşılaştırılması, matematiksel işlemlerin yapılması gibi daha birçok işlemlerin programatik olarak yapılmasını sağlayan bir nesnedir. Bu nesne altında alt nesnelere bulunmaktadır (Şekil 3.12). Bunlar;

1. Saklı prosedürler (Stored Procedure)
2. Fonksiyonlar (Functions)
3. Veritabanı tetikleyicileri (Database Triggers)
4. Assemblies
5. Tipler (Types)
6. Kurallar (Rules)
7. Varsayılanlar (Defaults)



Şekil 3.12 Programlanabilirlik nesnesinin alt nesnelere

3.6.6 Servis Habercisi (Service Broker)

MS SQL Server 2005 ile gelen bir özelliktir. “Service Broker” mimarisi SQL Server içerisinde servisler tanımlayabilmek ve bu servisler arasında mesaj temelli bir haberleşmeyi organize etmek için kullanılmaktadır. Mesaj tip tanımları, kontratlar,

kuyruklar, servisler, rotalar gibi bir dize nesne grubu ile bu türden bir mimari için alt yapı sağlamaktadır (Gözüdeli 2007)

3.6.7 Saklama (Storage)

MS SQL Server yazılımında “Storage” nesnesi, büyük boyutlu verilerle çalışıldığında, verilerin güvenliği sağlamak amacı ile veri dosyasını parçalara ayırarak farklı fiziksel disklerde saklanmasını ve “Storage Engine” nesnesi ile de tek sunucu platformunda kullanılmasını sağlamaktadır.

3.6.8 Güvenlik (Security)

MS SQL Serverda veri güvenliği ilk sırada gelmektedir. Sistem kurulum esnasında yada sonrasında Windows tabanlı yada hem Windows hem de sunucu tabanlı çalışması için güvenlik ayarlarının yapılandırılması gerekmektedir. Veri tabanına “Users” alt nesnesi ile yeni kullanıcılar tanımlanabilmekte, erişim şifreleri veya kullanıcı bilgileri güncellenebilmektedir. “Roles” nesnesi ile kullanıcılara roller tanımlanarak sistem içersindeki rolü belirlenmektedir. Örneğin bir kullanıcı veri tabanı ve tablo oluşturamaz sadece verileri görebilir nitelikte rol tanımlanabilir, yada başka bir kullanıcıya veritabanı oluşturma tablo oluşturma, fonksiyon oluşturma ve çalıştırma yetkileri verilebilir. MS SQL Server güvenlik (Security) nesnesinin alt nesnelere şekil 3.13’de verilmiştir.



Şekil 3.13 MS SQL Server güvenlik nesnesi ve alt nesnelere

Bütün veri tabanlarında olduđu gibi MS SQL Server veritabanında da yapısal sorgulama dili SQL kullanılmaktadır. SQL'nin bütün komutları tamamen SQL Server yazılımında geçerlidir. SQL Server yapısal sorgulama dilinin yanında kendisinin geliřtirmiş olduđu T-SQL dilini de kullanmaktadır.

4. EĞİTİM-ÖĞRETİM YÖNETİM SİSTEMİ İÇİN VERİTABANI TASARIMI

EÖYS tasarımında MS SQL Server veritabanı yönetim yazılımı kullanılmıştır. Veritabanı tasarımına geçilmeden önce eğitim öğretim sisteminin yapısı kurgulanmış, olması gereken özellikler belirlenmiş ve kâğıt üzerine aktarılmıştır. EÖYS 38 ana modül ve 153 tablodan oluşmaktadır. Her bir modül kendi içerisinde alt tablo yapısı ile bir bütündür. EÖYS'nin ana modülleri şunlardır;

1. Online başvuru modülü
2. Kullanıcı ve kullanıcı grupları modülü
3. Rol ve yetkiler modülü
4. Ünvan tanımlama modülü
5. Okul tanımlama modülü
6. Bölüm tanımlama modülü
7. Eğitim kataloğu modülü
8. Dersler modülü
9. Haftalık ders programı modülü
10. Şube işlemleri modülü
11. Danışman atama işlemleri modülü
12. Ders kayıt ve ders onay işlemleri modülü
13. Soru tanımlama modülü
14. Sınav modülü
15. Deneme sınavı modülü
16. Ödevler modülü
17. Öğretim elemanı performans takip
18. Öğrenci performans takip modülü
19. Öğrenci takip raporu modülü
20. Sözlük modülü
21. Anket modülü
22. Kişisel notlar modülü
23. Mesajlaşma modülü
24. Dosyalar modülü

25. Sohbet saatleri modülü
26. Toplantı modülü
27. Sıkça sorulan sorular modülü
28. Görüş ve öneriler modülü
29. Duyurular modülü
30. Ajanda modülü
31. Akademik takvim modülü
32. Transkript görüntüleme modülü
33. Raporlama modülü
34. Mali işlemler modülü
35. Demo kullanıcı modülü
36. Workshop modülü
37. Sanal sınıf modülü
38. Forum modülü

EÖYS'nin tabloları Ek'1 de listelenmiştir. Veritabanı tasarımında tam ilişkilendirme yöntemi kullanılmıştır. Bu ilişkilendirme işlemi yapılırken veritabanı tabloları arasında anlamsal bir senaryo oluşturulmuş ve bu senaryoya göre 14 farklı ilişkilendirme grubu oluşturulmuştur. Bu ilişkilendirme grupları şunlardır.

1. Çalıştay (Workshop)
2. Sohbet
3. Sınav
4. Menü
5. Mali Durum
6. Kullanıcılar
7. Akademik Bilgiler
8. Grup
9. Forum
10. Diğer
11. Ders
12. Demo

13. Başvuru

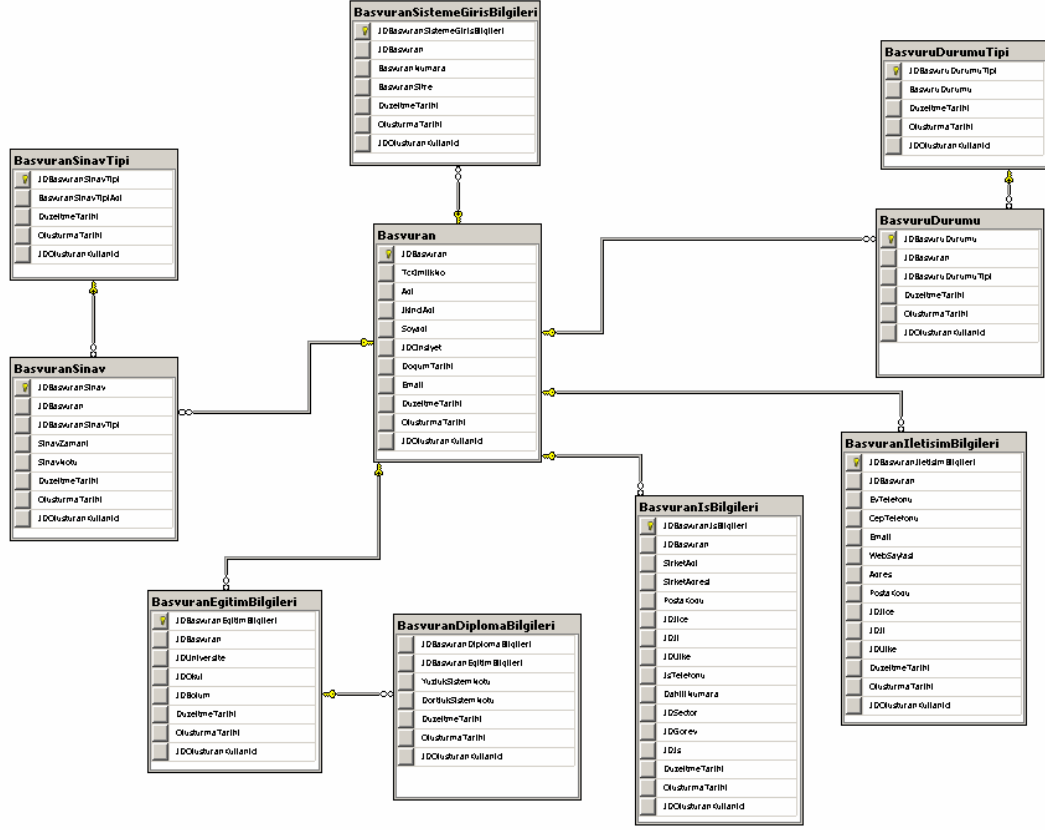
14. Anket

Sistem içerisindeki her bir modül kendi içerisinde bir bütündür. Bununla birlikte bir modül aynı zamanda bir ilişki senaryosu bütünüdür. Yani bir ilişki senaryosu içerisinde yalnızca bir modül olabildiği gibi birkaç kaç modülde bulunabilmektedir. Bu sebeple çalışmanın ilerleyen kısımlarında modüller tek-tek açıklanmıştır. Aynı zamanda ilişki senaryosu ile olan bağılılığı vurgulanmıştır.

4.1 Online Başvuru Modülü

EÖYS'nin ilk başlangıç aşamasını oluşturan bu modül aynı zamanda bir ilişki senaryosudur. Sistemden yararlanmak isteyen öğrencilerin ve aynı zamanda öğretim elemanlarının da kayıt oldukları giriş, yani başvuru kısmıdır. Bu sistem fen, sosyal ve sağlık bilimleri enstitüleri için de ekstra alanların tanımlandığı esnek bir EÖYS'dir. Enstitüler için öğrencilere ALES ve Dil sınavı sonuçlarının tanımlandığı alanlar eklenmiştir. Bu modülde başvuru için çevrimiçi başvuru işlemleri yapılmaktadır. Başvuran ilişki senaryosu ve modülü şekil 4.1 te görülmektedir. Modül içerisinde 10 adet tablo tanımlanmıştır. Bunlar;

1. BasvuranEgitimBilgileri
2. BasvuranDiplomaBilgileri
3. Basvuran
4. BasvuranSistemeGirisBilgileri
5. BasvuranİletisimBilgileri
6. BasvuruDurumu
7. BasvuruDurumuTipi
8. BasvuranSinav
9. BasvuranSinavTipi
10. BasvuranIsBilgileri



Şekil 4.1 Başvuran modülü tabloları ve ilişkileri

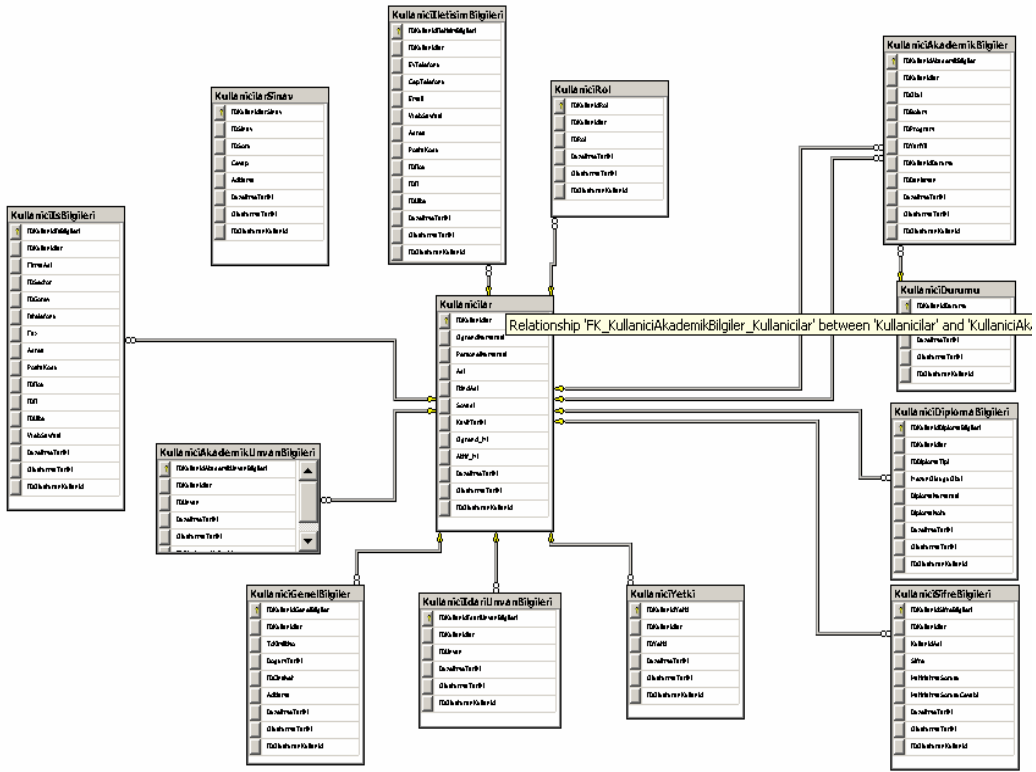
4.2 Kullanıcı Ve Kullanıcı Grupları Modülü

Kullanıcı modülü EÖYS'nin en önemli modüllerinden biridir. Burada kullanıcılar ve kullanıcı grupları üzerindeki işlemlerin yapıldığı modüldür. Burada belirli kullanıcı grupları tanımlanmıştır. Örneğin Admin seviyesinde tanımlanan kullanıcısı, sistem yöneticisi, bölüm sekreteri, bölüm başkanı ve öğrenci işleri yönetimin görebiliyor olmasıdır. Admin seviyesinde eklenen bir kullanıcının yapabileceği işlemler şunlardır;

- Eklenen kullanıcının sisteme erişebiliyor ve yetkisi olan alanları kullanabiliyor olması.
- Silinen kullanıcının sisteme erişemiyor olması.
- Güncellenen kullanıcı bilgilerinin tüm sistemde yeni halini alması.
- Eklenen kullanıcı grubunun sistemde olması ve bu gruba kullanıcılar eklenebiliyor olması.
- Güncellenen kullanıcı grubu bilgilerinin tüm sistemde yeni halini alması.

Şekil 4.2’de kullanıcı modülü ve bu modüle ait ilişki senaryosu verilmektedir. Burada bire-sonsuz ilişki “kullanıcı” tablosu üzerinedir. Bu modüle ait toplam 13 adet tablo kullanılmıştır. Bunlar;

1. KullaniciAkademikBilgiler
2. KullaniciAkademikUnvanBilgileri
3. KullaniciDiplomaBilgileri
4. KullaniciDurumu
5. KullaniciGenelBilgiler
6. KullaniciIdariUnvanBilgileri
7. KullaniciIletisimBilgileri
8. KullaniciIsBilgileri
9. Kullanicilar
10. KullanicilarSinav
11. KullaniciRol
12. KullaniciSifreBilgileri
13. KullaniciYetki

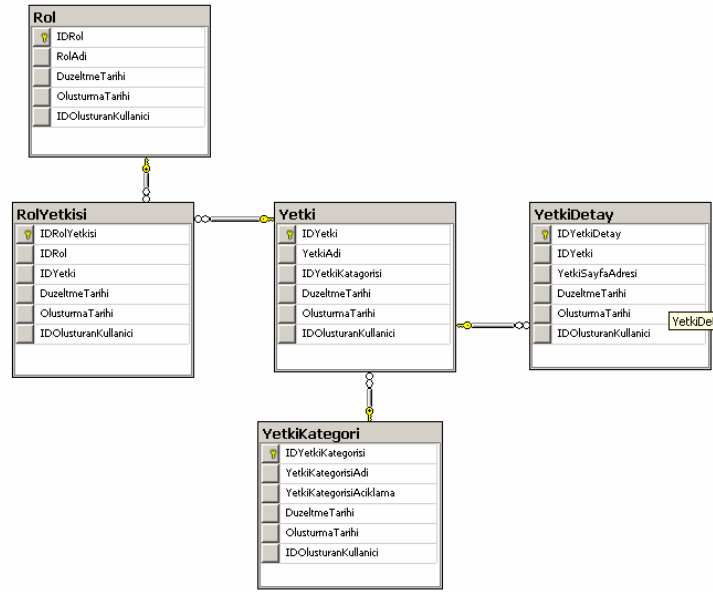


Şekil 4.2 Kullanıcılar modülü ve ilişki senaryosu

4.3 Rol Ve Yetkiler Modülü

Bu modülde sisteme belirli rol ve yetki tanımlamaları yapılmaktadır. Bu tanımlamalar içerisinde kayıtlı her bir kullanıcı için rol ve yetki ataması yapılmaktadır. Sisteme yeni roller ve yetkiler eklenebilmektedir. Sistem yöneticisi tarafından belirli alanlara erişilebilecek yeni yetki tanımlamaları yapılabilir veya kaldırılabilir. Eklenen veya güncellenen roller anında sistemde yer almaktadır. Şekil 4.3'te rol ve yetkiler modülüne ait tablo bir modülün ilişkisi yer almaktadır. Bu modülde yer alan tablolar şunlardır;

1. Rol
2. RolYetkisi
3. Yetki
4. YetkiDetay
5. YetkiKategori

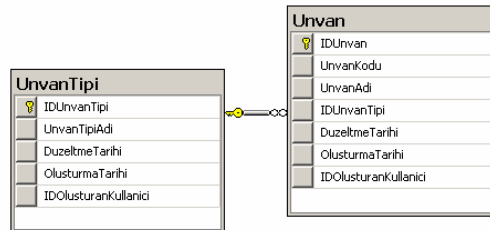


Şekil 4.3 Rol ve Yetki modülü tabloları ve ilişki senaryosu

4.4 Unvan Tanımlama Modülü

Unvan tanımlama modülü, sisteme kayıtlı akademik ve idari personelin unvan tanımlamaları yapılmaktadır. Akademik personel için Okt., Öğr.Grv., Yrd.Doç.Dr., Doç.Dr., Prof.Dr. kısaltmaları kullanılarak tanımlamalar yapılmakta, idari personel için ise Dekan, DekanYrd, Bölüm Başkanı ve bölüm sekreteri gibi unvanları tanımlanmaktadır. Unvan tanımlama modülünde iki adet tablo tanımlanmıştır. Bu tablo ve ikisi arasındaki ilişki şekil 4.4’de görülmektedir. Bu modül için tanımlanan tablolar şunlardır;

1. Unvan
2. UnvanTipi

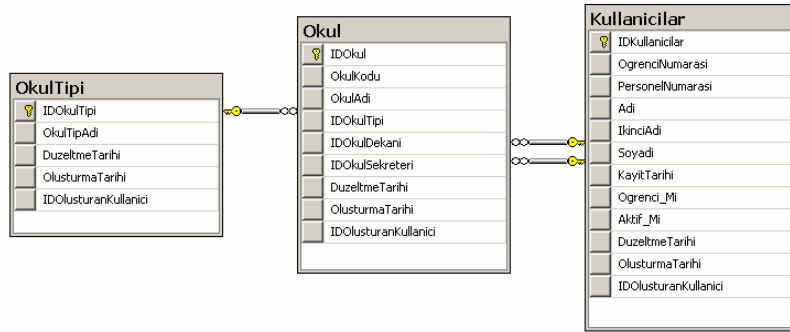


Şekil 4.4 Unvan tanımlama modülü tabloları ve ilişki senaryosu

4.5 Okul Tanımlama Modülü

Okul tanımlama modülü sistemin kurulacak ve yönetilecek olduğu okulun yada enstitünün tanımlandığı bölümdür. Burada fakülte, yüksek okul, meslek yüksek okulu yada enstitü tanımlanabilir. Okul tanımlaması yapıldıktan sonra okulun akademik ve idari personel atamaları yapılır. Okul tanımlama modülünde iki adet tablo kullanılmıştır. Bu tablolar ve ilişkileri şekil 4.5'te gösterilmektedir. Bu modül için tanımlanan tablolar şunlardır.

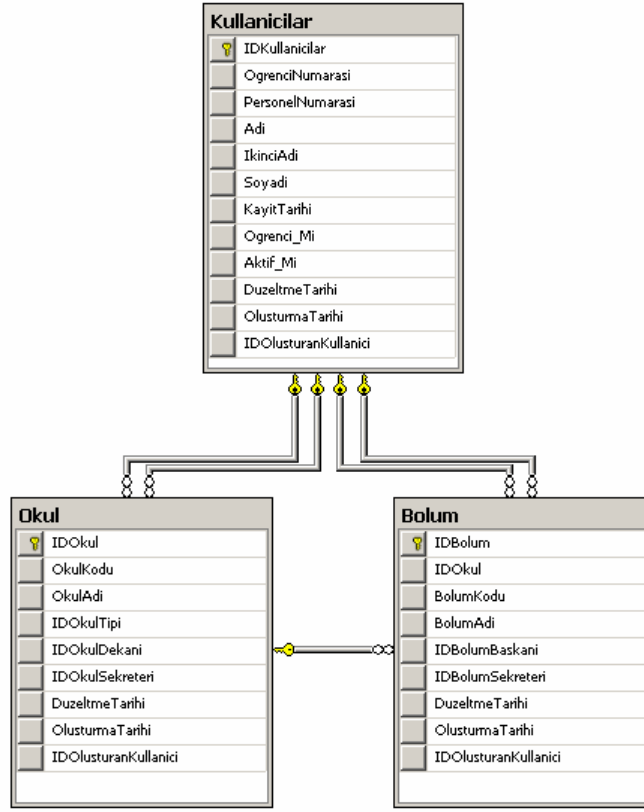
1. Okul
2. OkulTipi



Şekil 4.5 Okula tanımlama modülü tabloları ve ilişki senaryosu

4.6 Bölüm Tanımlama Modülü

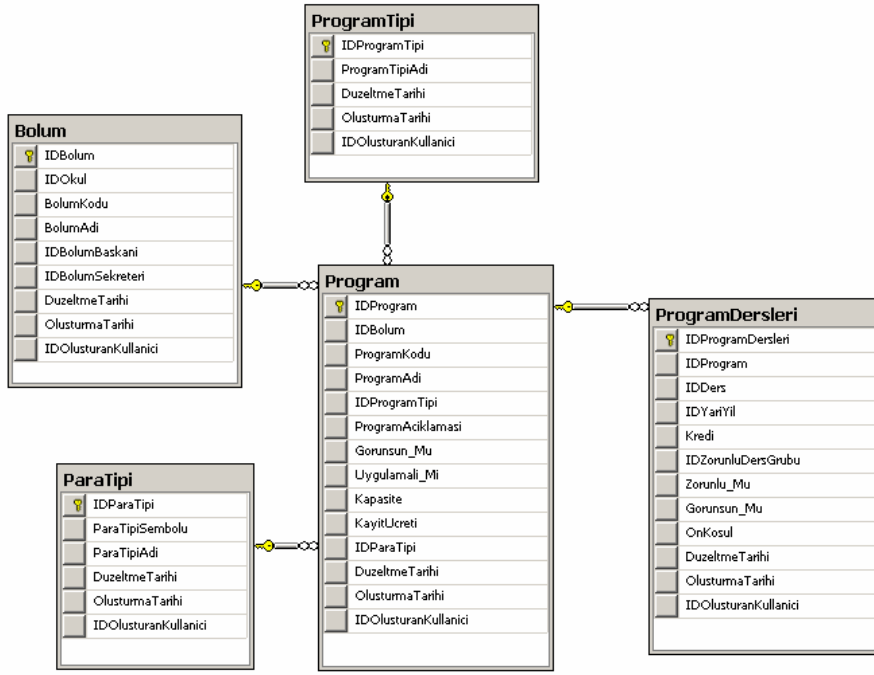
Sistem yöneticisi, okul tanımlama, idari ve akademik personeli tanımlama işlemini gerçekleştirdikten sonra bölüm kodu ile birlikte yeni bir bölüm açar bu bölüme bir bölüm başkanı ve bölüm sekreteri atar. Şekil 4.6'da görüldüğü gibi bu modül için sadece bir tablo kullanılmıştır. Bu tablo aynı zamanda okul tablosu ile ilişkilidir.



Şekil 4.6 Bölüm tanımlama tablosu

4.7 Eğitim Kataloğu Modülü

Eğitim katalogu modülü, sistem yöneticisinin veya bölüm sekreterinin tanımlama yapabileceği bölümler için alt programların tanımlandığı modüldür. Burada bir bölüm için yeni bir programın kodu ve adıyla detaylı bir şekilde tanımlanması yapılır. Daha sonra bu programa başvurular için başvuru kriterlerinin belirlenmesi, kayıt ücreti, sınıf kapasitesi dersin uygulamalı yada teorik olması ile ilgili tanımlamalar yapılır. Tanımlanmış olan programı tamamlamak için kredi yeter miktarını, programın derslerini, staj ve bunun gibi bir programın eğitim-öğretiminde gerekli olan tüm akademik ve idari işlemleri bu modülde tanımlanarak yapılır (Şekil 4.7).



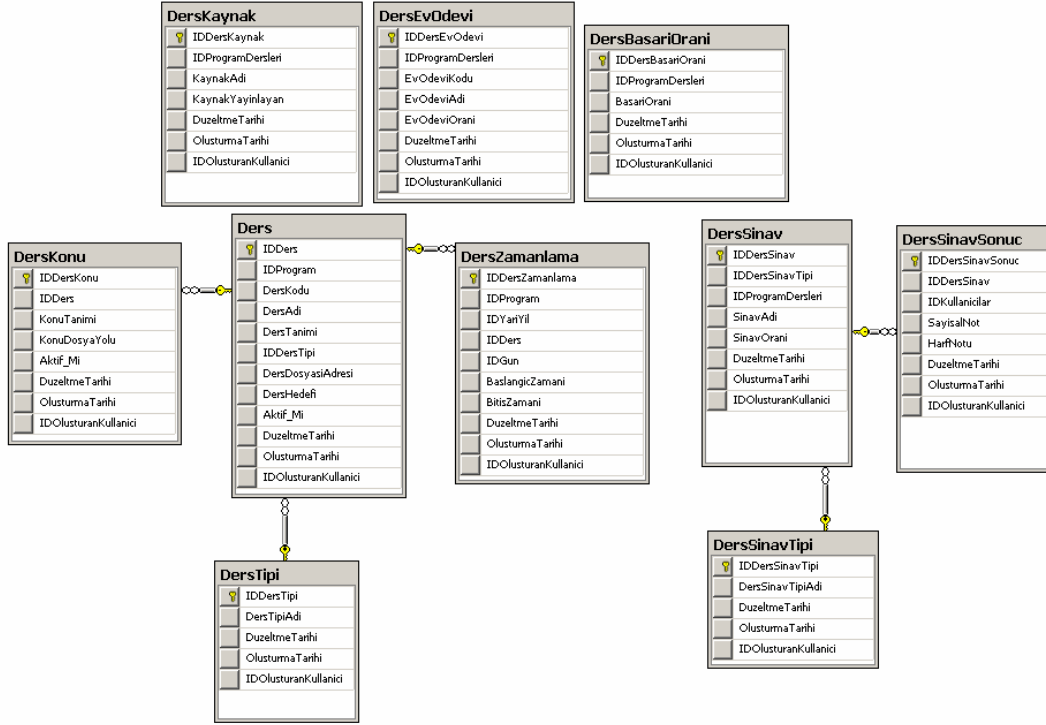
Şekil 4.7 Eğitim katalogu modülü için tablolar ve ilişkileri

4.8 Dersler Modülü

Dersler modülü adında anlaşılacağı üzere programlara ait derslerin tanımlandığı bir modüldür. Bu modülde bölüm ve programa ait dersin kodu, adı, tanımı, tipi, hedef bilgileri ve ders içeriklerinin yer alacağı adres tanımlanır. Derslerin ödevleri, içeriklerinin tanımlamaları, bu derse ait vize ve final sınavlarının belirlenmesi, sınav tiplerinin belirlenmesi gibi birçok işlemler bu modül içerisinde gerçekleştirilir. Şekil 4.8’de dersler modülünün tabloları ve ilişkileri verilmiştir. Bu modül içerisinde 10 adet tablo tanımlanmıştır. Bu tablolar şunlardır;

1. Ders
2. DersBasariOrani
3. DersEvOdevi
4. DersKaynak
5. DersKonu
6. DersSinav
7. DersSinavSonuc

8. DersSinavTipi
9. DersTipi
10. DersZamanlama

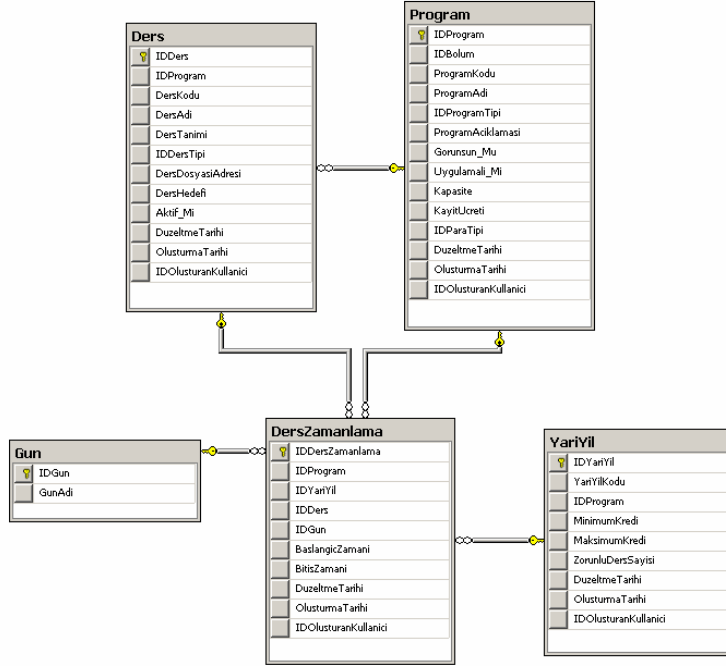


Şekil 4.8 Dersler modülü tabloları ve ilişkileri

4.9 Ders Programı Modülü

Ders programı modülünde eğitim öğretimin olduğu dönem içerisinde derslerin zamanlaması yapılmaktadır. Burada dersin başlama ve bitiş saatleri bölüm sekreteri tarafından tanımlanır yada güncellenir. Dersin zamanlaması dersin öğretim elemanının sanal sınıf ortamında öğrenciler ile görüşmesini sağlayan plandır. Bu plana göre öğrenciler dersler ile ilgili sormak istediklerini dersin öğretim elemanına sorabilirler veya sınıf arkadaşları ile bir tartışma ortamı oluşturabilirler. Burada bir dersin hangi yıl, yarıyıl, hangi programa ait olduğu ve başlangıç ve bitiş zamanlarının tanımlamaları yapılmaktadır. Şekil 4.9'da ders programı modülünün tabloları ve ilişkileri verilmiştir. Bu modül içerisinde 5 adet tablo tanımlanmıştır. Bunlar;

1. Ders
2. DersZamanlama
3. Gun
4. YariYil
5. Program



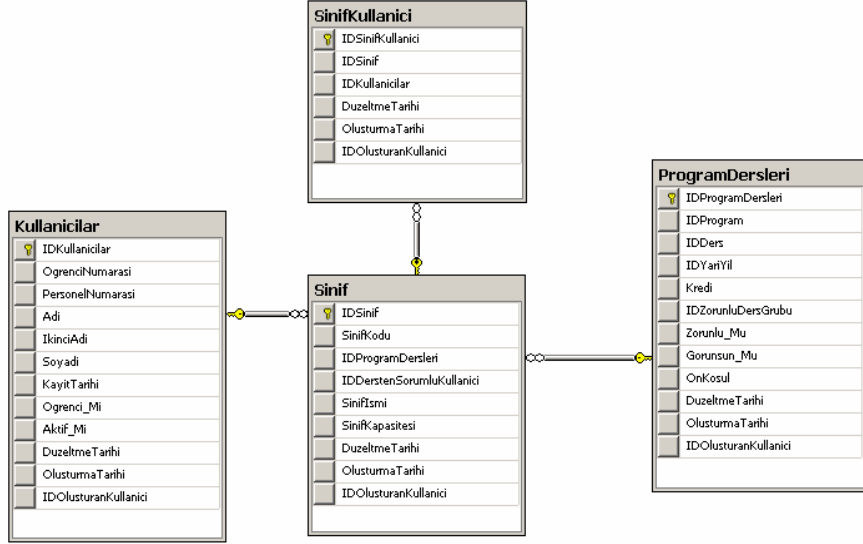
Şekil 4.9 Ders programı modülü tabloları ve ilişkileri.

4.10 Sınıf İşlemleri Modülü

Sınıf işlemleri modülü bir programa ve o programın o yılına ait birden fazla sınıf olduğu takdirde uygulamaya konulan bir modüldür. Örneğin Bilgisayar Öğretmenliği bölümü A ve B şubesi olarak düşünülür her iki sınıf (şube) içinde ders tanımlama, öğrenci tanımlama ve ders programı yapma gibi işlemler ayrı ayrı yapılacaktır. Bu modülde öğrenciler hangi sınıfa (şubeye) aitse o şubeye ataması yapılır ve tüm öğrencilik işlemleri o şube için gerçekleştirilir. Şekil 4.10 da sınıf işlemleri modülünün tabloları ve ilişkileri verilmiştir. Bu modülde 4 adet tablo tanımlanmıştır. Bu tablolar şunlardır;

1. Sınıf
2. SınıfKullanici

3. Kullanicilar
4. ProgramDersleri

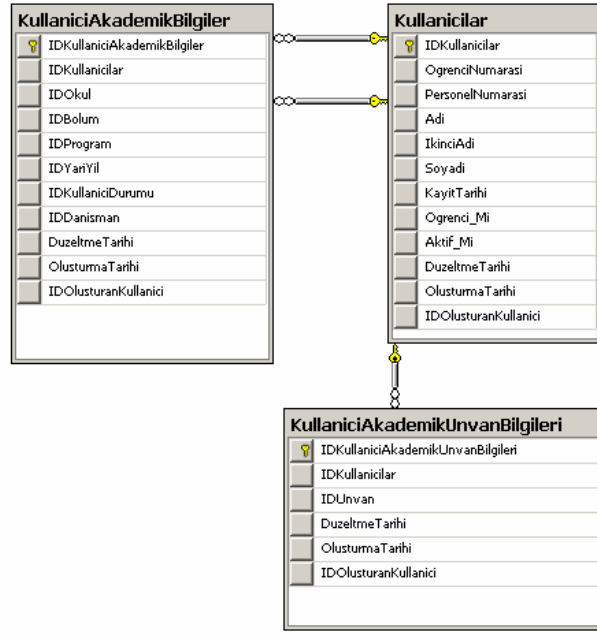


Şekil 4.10 Sınıf tanımlama modülü tabloları ve ilişkileri

4.11 Danışman Atama İşlemleri Modülü

Danışman atama işlemleri modülünde ilk defa kayıt yaptıran programın bir şube öğrencilerine bir danışman öğretim elemanı atama işlemi yapılmaktadır. Danışman, istenirse sistem yöneticisi yâda bölüm başkanı tarafından değiştirebilir. Atanan danışman öğretim elemanı öğrencilerin ders kayıt sürecini takip etmekte, ders onay ve kredi işlemlerini onaylamaktadır. Şekil 4.11’de danışman atama işlemleri modülünde kullanılan tablolar ve ilişkileri verilmiştir. Bu modülde 3 adet tablo kullanılmıştır. Bunlar;

1. Kullanicilar
2. KullaniciAkademikBilgiler
3. KullaniciAkademikUnvanBilgileri

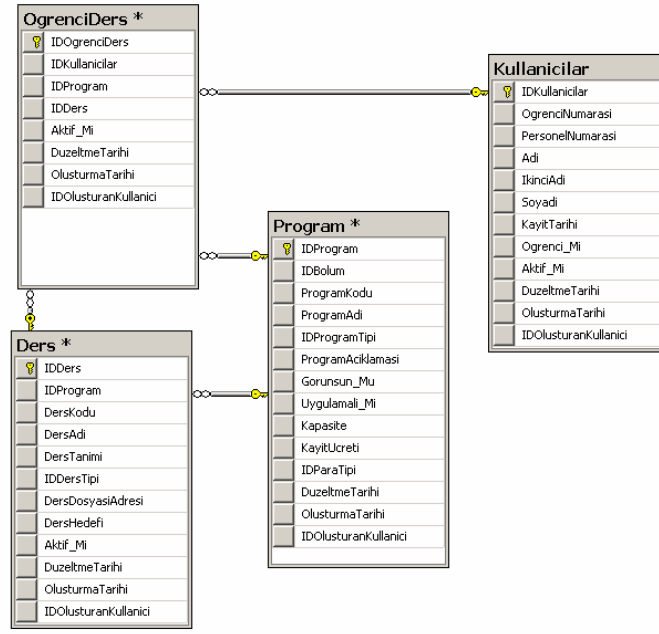


Şekil 4.11 Danışman atama işlemleri modülü tablolar ve ilişkileri

4.12 Ders Kayıt Ve Ders Onay İşlemleri Modülü

Ders kayıt ve onay işlemleri modülünde öğrencilerin okudukları program ve döneme ait ders seçme işlemleri ve seçilen derslerin aktivasyon işlemleri yapılmaktadır. Bu modülde her bir öğrenci bir kullanıcı adına ve şifreye sahiptir. Öğrenciler sisteme giriş yaptıktan sonra dönem başında kredi durumuna göre ders seçebilmektedir. Öğrencinin alt sınıftan dersi varsa o dersler zorunlu olarak seçilmekte ve kredi doluncaya kadar normal dönemden ders alabilmektedir. Ders seçme işlemleri tamamlandıktan sonra danışman dersleri inceler ve daha sonra onay verir ve böylece ders seçme işlemi kapanır. Artık alınan dersler öğrencinin o dönemine kaydedilmektedir. Bu modülün ilişkileri ve tabloları şekil 4.12’de verilmiştir. Bu modül için 4 adet tablo tanımlanmıştır. Bunlar;

1. Ders
2. ÖğrenciDers
3. Program
4. Kullanicilar

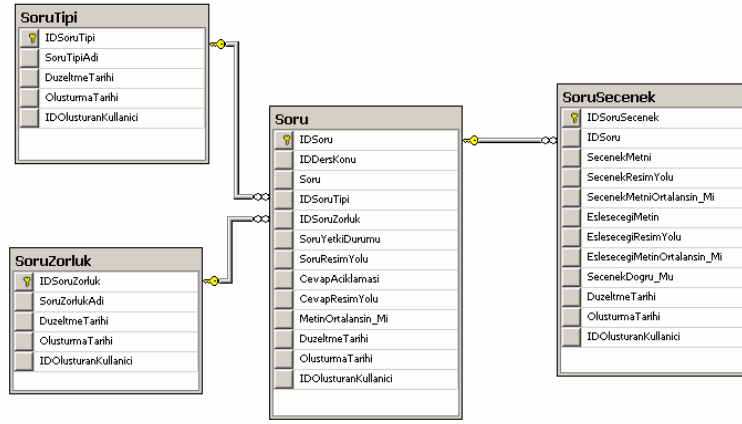


Şekil 4.12 Ders kayıt ve ders onay işlemleri modülü tabloları ve ilişkileri

4.13 Soru Tanımlama Modülü

Soru tanımlama modülünde derslerin vize ve final sınavları için soru tipleri ve sorular hazırlanmaktadır. Bu modülde derslerin vize ve final sınavları için çoktan seçmeli veya çoktan çok seçmeli gibi soru tipleri tanımlanır, birden beşe kadar zorluk seviyeleri belirlenir. Sorular için resimler eklenebilir, gerektiğinde güncellemeler yapılabilir. Sorulara yetkiler tanımlanabilir, yani bir soru grubu tüm sınıfa yönelik olabilir veya bir gruba veya bir kullanıcıya yönelik olabilir. Bu işlemler dersin öğretim elemanı tarafından tanımlanmaktadır. Soru tanımlama modülü tabloları ve ilişkileri şekil 4.13’de verilmiştir. Bu modülde 4 adet tablo tanımlanmıştır. Bunlar;

1. Soru
2. SoruSecenek
3. SoruTipi
4. SoruZorluk

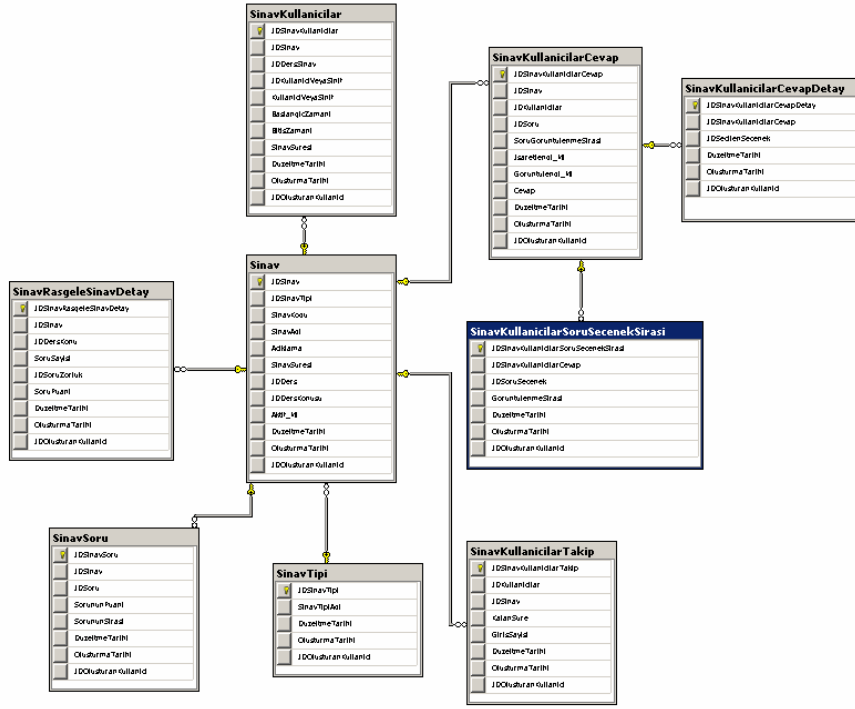


Şekil 4.13 Soru tanımlama modülü tabloları ve ilişkileri

4.14 Sınav Modülü

Sınav modülü dersin vize ve final sınavları olabileceği gibi özel sınavlarda olabilmektedir. Bunun için sınava tabi tutulacak şube yada sınıf tanımlanır, sınav tipi tanımlanır, sınav tarihi tanımlanır, sınavın başlangıç ve bitiş saati tanımlanır, önceden hazırlanmış sorular sınav için belirlenir ve puanlama belirlenerek sınavla birlikte öğrenciye sunulur. Öğrenciler sınava girer soruları cevaplar ve bu cevaplara göre değerlendirme ve puanlama eş zamanlı olarak yapılır ve öğrenciye ilan edilir. Sınav tipi ödev niteliğinde değerlendirme olursa puanlama ve ilan daha sonra yapılır. Şekil 4.14’de sınav modülü tabloları ve ilişkileri verilmiştir. Bu modülde 9 adet tablo kullanılmıştır. Bunlar;

1. Sınav
2. SınavKullanicilar
3. SınavKullanicilarCevap
4. SınavKullanicilarCevapDetay
5. SınavKullanicilarSoruSecenekSirasi
6. SınavKullanicilarTakip
7. SınavRasgeleSınavDetay
8. SınavSoru
9. SınavTipi

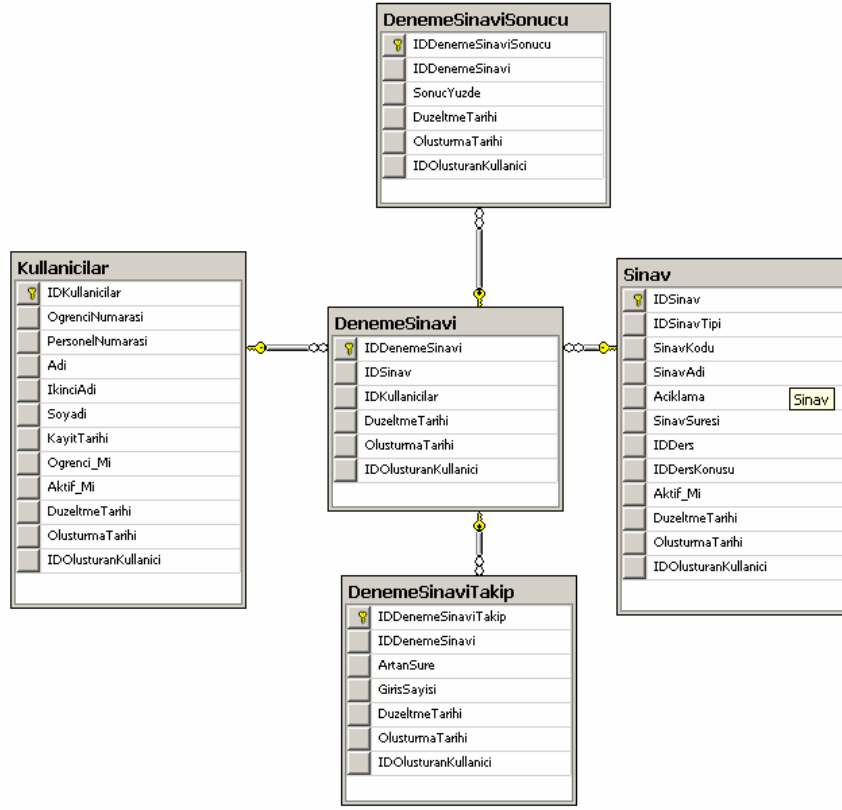


Şekil 4.14 Sınav modülü tabloları ve ilişkileri

4.15 Deneme Sınavı Modülü

Deneme sınavı modülünde öğrencinin almış olduğu modül listelenir ve deneme sınavı yapılacak olan ders seçilir ve varsa alt konuları tanımlanır. Deneme sınavında zorluk derecesi varsa zorluk dereceleri sıralanır. Sorular tablosundan deneme sınavı için sorular alınır ve deneme sınavı hazırlanmış olur. Öğrencilere uygulanan deneme sınavları normal sınav gibi uygulanır ve değerlendirilir. Deneme sınavı modülünün tablo ve ilişkiler şekil 4.15’de verilmiştir. Bu modül için 5 adet tablo tanımlanmıştır. Bunlar;

1. DenemeSinavi
2. DenemeSinaviSonucu
3. DenemeSinaviTakip
4. Kullanicilar
5. Sinav

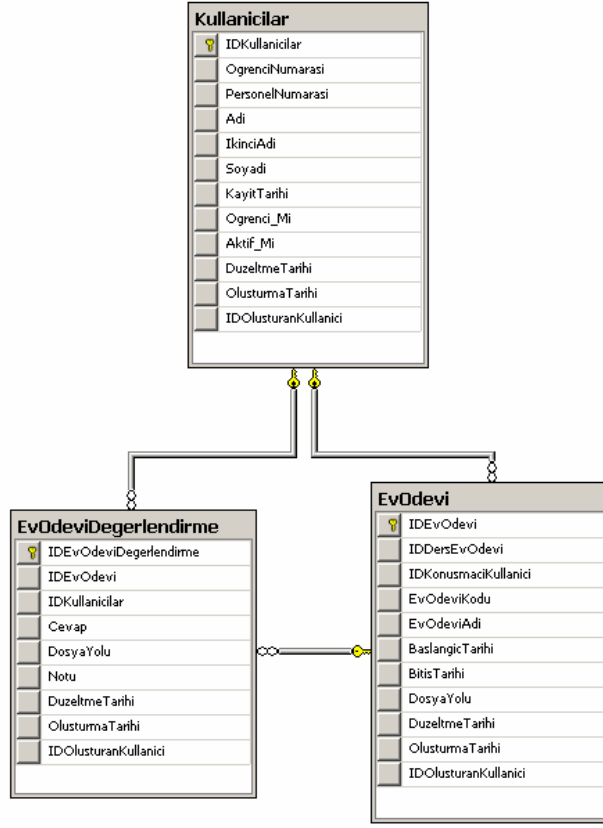


Şekil 4.15 Deneme sınavı modülü tabloları ve ilişkileri

4.16 Ödevler Modülü

Ödevler modülü öğrencilere verilen ev ödevlerinin öğrencilere atandığı ve öğrencilerin ev ödevlerinin sistem üzerinden dersin öğretim elemanına gönderdiği modüldür. Bu modülde. Dersin öğretim elemanı öğrencilerine ev ödevi tanımlar ve atar, öğrenci ödevi yaptıktan sonra sistem üzerinden elektronik olarak ödevi verilen tarihe kadar o ders altında dersin hocasına geri gönderir. Dersin öğretim elemanı ödevi değerlendirdikten sonra ödev notu olarak ilan eder. Bu modülün tabloları ve ilişkileri şekil 4.16'da verilmiştir. Ödevler modülünde 3 adet tablo tanımlanmıştır. Bunlar;

1. Kullanıcılar
2. EvOdevi
3. EvOdeviDeğerlendirme

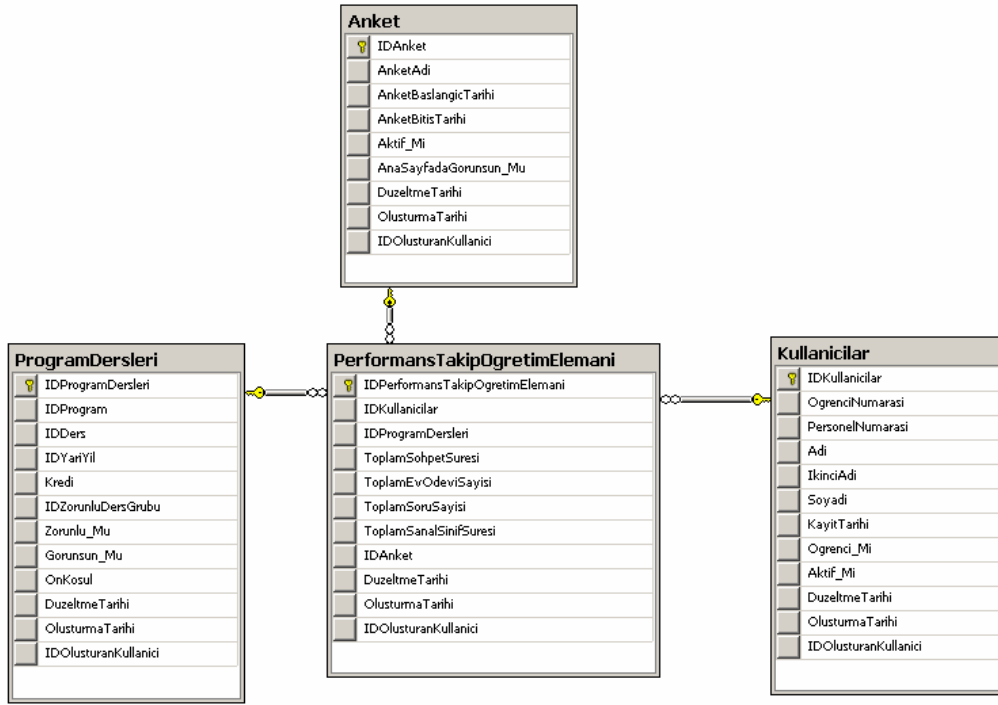


Şekil 4.16 Ödevler modülü tablo ve ilişkisi

4.17 Öğretim Elemanı Performans Takip

Öğretim elemanı performans takip modülünde ders veren öğretim elemanları listelenir. Buradan performansa bakılmak istenen yıl ve yarıyıl seçilir ve öğretim elemanının o döneme ait verdiği dersler listelenir. Burada verdiği ödevler, düzenlediği sanal sınıf toplantıları, dersleri, anket sonuçları, sisteme girdiği soru sayısı gibi kriterlerde öğretim elemanın performansı ölçülür. Bu modülü görmek ve yönetmek bölüm başkanına aittir. Diğer öğretim elemanları veya idari personel bunu göremez. Bu modülün tabloları ve ilişkileri şekil 4.17’de verilmiştir. Yine bu modül için 4 adet tablo tanımlanmıştır. Bunlar;

1. PerformansTakipOğretimElemanı
2. Anket
3. ProgramDersleri
4. Kullanicilar

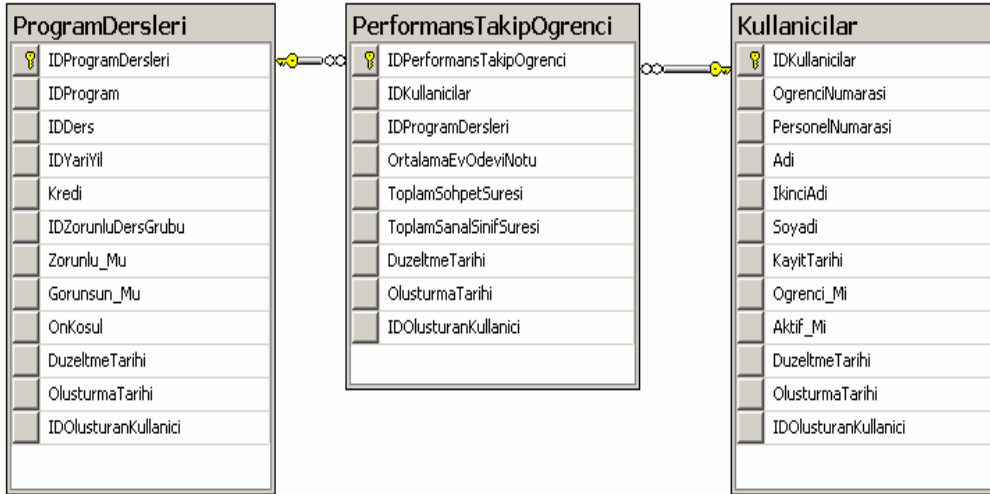


Şekil 4.17 Öğretim elemanı performans takip modülü tabloları ve ilişkileri

4.18 Öğrenci Performans Takip Modülü

Öğrenci performans takip modülünde, öğrencilerin eğitim öğretim dönemi içerisindeki performansları ölçülmektedir. Bu modülde öğrencinin aldığı dersler ve bu derslerin bilgileri listelenir. Burada vize, final ve ödev notları incelenir, sohbetlere katılıp katılmadıkları, katıldı iseler ne kadar sıklıkla katıldıkları, sanal derslere katılımları ve ders sorumlusunun vermiş olduğu performans notları ile öğrenci değerlendirilir. Burada öğrenci kendisinin performansını görebilir. Yani öğretim elemanlarının değerlendirmesi dışındadır. Yine öğrenciler de dersten sorumlu olan kişileri anketlerle değerlendirir. Bu modülün tablo ve ilişkileri şekil 4.18’de verilmiştir. Bu modüle 3 adet tablo tanımlanmıştır. Tanımlanan tablolar şunlardır;

1. ProgramDersleri
2. Kullanici
3. PerformansTakipOgrenci

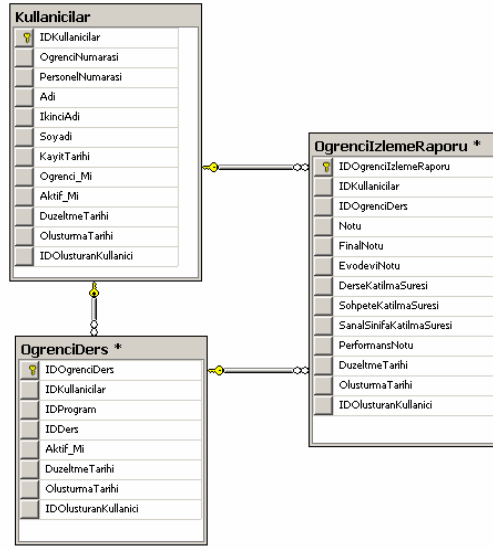


Şekil 4.18 Öğrenci performans takip modülü tabloları ve ilişkileri

4.19 Öğrenci Takip Raporu Modülü

Öğrenci takip raporu modülünün öğrenci performans takip modülünden farkı öğretim elemanlarının öğrenciyi değerlendirmesidir. Yine aynı şekilde bu modülde öğrencinin aldığı dersler ve bu derslerin bilgileri listelenir. Burada vize, final ve ödev notları incelenir, sohbetlere katılıp katılmadıkları, katıldı iseler ne kadar sıklıkla katıldıkları, sanal derslere katılımları ve ders sorumlusunun vermiş olduğu performans notları ile öğrenci değerlendirilir ve öğrencinin performans notu öğretim elemanı tarafından sisteme kaydedilir. Öğrenci takip raporu modülün tablo ve ilişkileri şekil 4.19’da verilmiştir. Bu modüle 3 adet tablo tanımlanmıştır. Bunlar;

1. Kullanicilar
2. OgrenciDers
3. OgrenciIzlemeRaporu

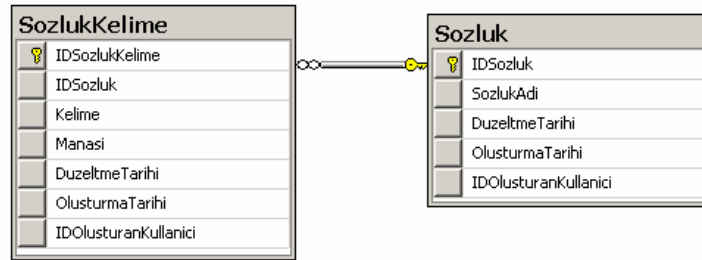


Şekil 4.19 Öğrenci takip raporu modülü tabloları ve ilişkisi

4.20 Sözlük Modülü

Öğrenci, öğretim elemanı, idari personel ve misafirlerinde kullanabileceği bir sözlük modüldür. Şekil 4.20’de sözlük modülünün tablosu ve ilişkisi verilmiştir. Bu modüde 2 adet tablo kullanılmıştır. Bunlar;

1. SozlukKelime
2. Sozluk



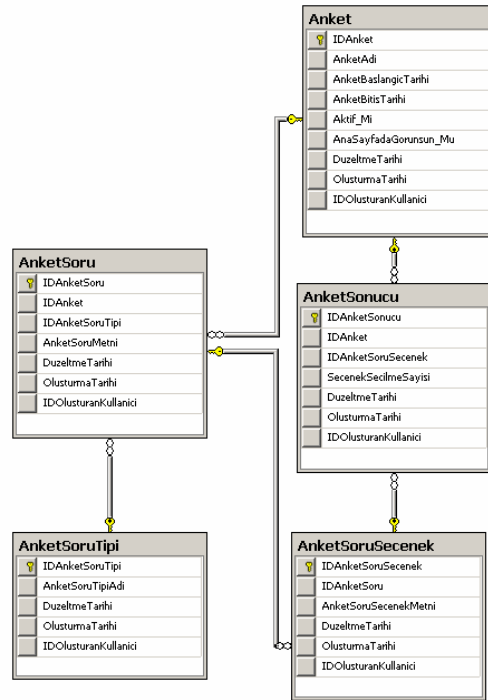
Şekil 4.20 Sözlük modülü tablo ve ilişkisi

4.21 Anket Modülü

Anket modülü adından da anlaşılacağı gibi öğrenciler ve öğretim elemanlarına yönelik anket çalışmaları için hazırlanmış bir modüldür. Kullanıcı tarafından anket tanımlanır

soruları hazırlanır ve arayüzün herhangi bir yerinde kullanıcıya gösterilir. Anket tayin edilen zamanda otomatik olarak sonlanır bu süreç içerisinde istatistiki bilgiler istenirse misafirlere gösterilebilir, yada gizli tutulabilir. Bu modüle ilişki senaryosu ve tabloları şekil 4.21’de verilmiştir. Anket modülü için 5 adet tablo tanımlanmıştır. Bunlar;

1. Anket
2. AnketSoru
3. AnketSoruTipi
4. AnketSoruSecenek
5. AnketSonucu

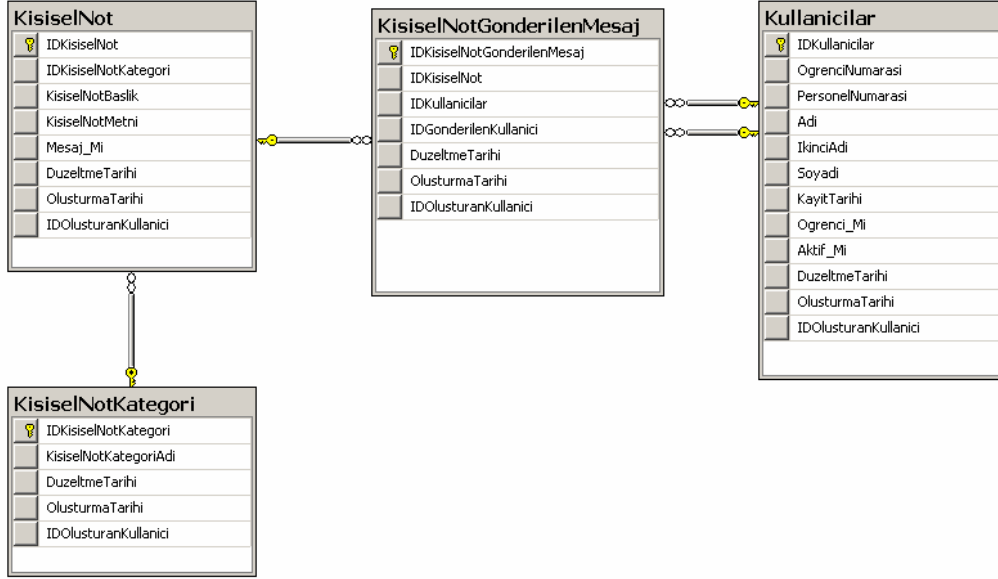


Şekil 4.21 Anket modülü tabloları ve ilişkileri

4.22 Kişisel Notlar Modülü

Kişisel notlar modülünde kullanıcılar öğrenciler, öğretim elemanları, sistem yöneticisi ve bölüm sekreterleridir. Kullanıcılar gerek ders gerekse kişisel konular hakkında kategori tanımlayarak notlar girebilir ve saklayabilir. Kişisel notlar modülünün tabloları ve ilişki senaryoları şekil 4.22’de verilmiştir. Bu modül için 4 adet tablo tanımlanmıştır. Bunlar;

1. KişiselNot
2. KişiselNotKategori
3. KişiselNotGonderilenMesaj
4. Kullanicilar

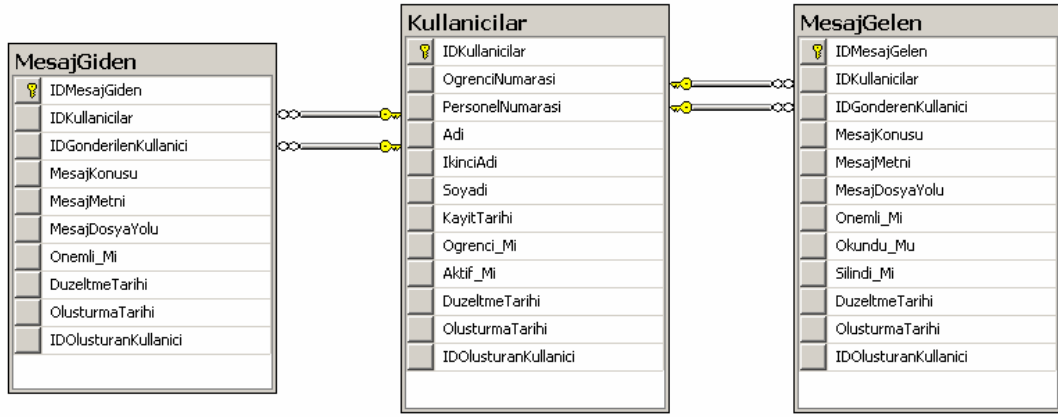


Şekil 4.22 Kişisel notlar modülü tabloları ve ilişkileri

4.23 Mesajlaşma Modülü

Mesajlaşma modülü sistem içerisindeki kullanıcıların birbirleri ile sistem içi haberleşmelerini ve mesajlaşmalarını sağlamaktadır. Öğrenciler, bölüm sekreterleri, öğretim elemanları, tüm akademik ve idari personel kendi aralarında kapalı sistem olarak haberleşmektedir. Mesajlaşma modülünün tabloları ve kurulan ilişki senaryosu şekil 4.23’de verilmiştir. Bu modül için 3 adet tablo tanımlanmıştır. Bunlar;

1. Kullanicilar
2. MesajGiden
3. MesajGelen

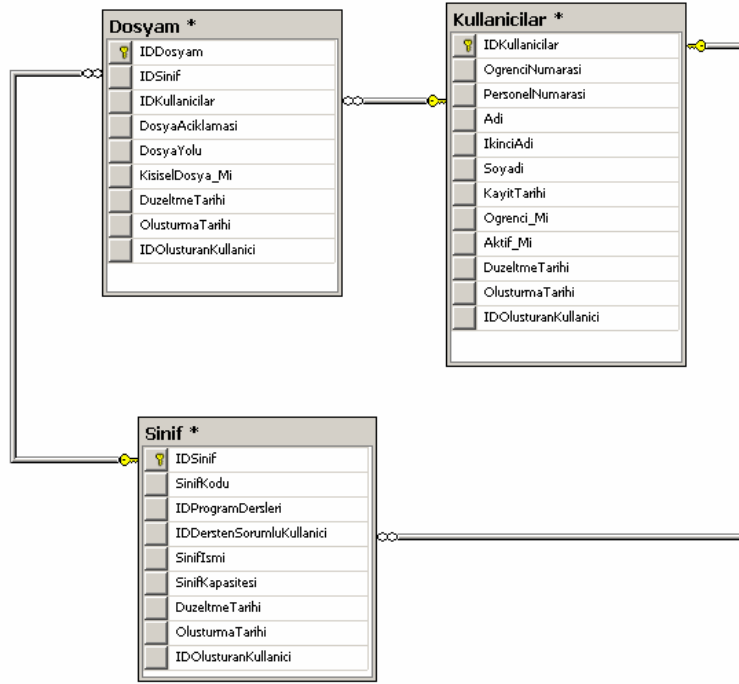


Şekil 4.23 Mesajlaşma modülü tabloları ve ilişkileri

4.24 Dosyalar Modülü

Dosyalar modülünde öğretim elemanlarının vermiş olduğu derslerle ilgili dokümanları sisteme yüklemesini ve dersi alan kişilerin bu dokümanlara erişebilmesini sağlamaktadır. Öğretim elemanlarına dersler ile ilgili dosyaları arşivlemeleri için belirli bir kota tayin edilmektedir. Bu kotayı aştıklarında sistem otomatik olarak kullanıcıyı uyarabilmektedir. Öğrenciler öğretim elemanlarının eklemiş oldukları bu dosyalara erişebilmektedir. Arşivdeki dosyalar silindiğinde sistem üzerinden tamamen silinmektedir. Öğrenciler arşivdeki dosyalar için kapsamlı arama yapabilmektedir. Dosyalar modülü için tablolar ve ilişki senaryoları şekil 4.24’de verilmiştir. Bu modül için 3 adet tablo tanımlanmıştır. Bunlar;;

1. Dosyam
2. Kullanicilar
3. Sinif

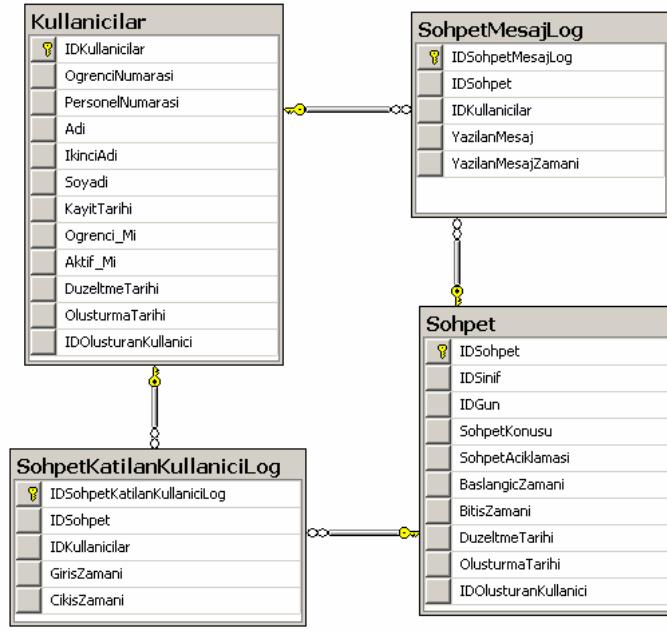


Şekil 4.24 Dosyalar modülü tabloları ve ilişkileri

4.25 Sohbet Modülü

Sohbet, EÖYS içerisinde tüm kullanıcıların birbirleri ile site içerisinde sohbet etmelerini sağlayan bir modüldür. Burada eğitim ile ilgili birçok konular açılabilir ve öğrencilerin bu konulara eşzamanlı yazarak sohbet ile katılmaları sağlanabilir. Sohbet modülü sadece kayıtlı kullanıcılara açık bir işlemdir. Yeni bir sohbet oluşturulacağı zaman, önce ders seçilir, ne zaman yapılacağı belirlenir ve kullanıcıların belirtilen zamanda sohbete katılmaları için çağrı yapılır. Şekil 4.25’de sohbet modülüne ait tablo ve ilişki senaryosu görülmektedir. Bu modül içerisinde 4 adet tablo birbirleri ile ilişkilendirilmiştir. Bunlar;

1. Sohbet
2. Kullanicilar
3. SohbetMesajLog
4. SohbetKatilanKullaniciLog

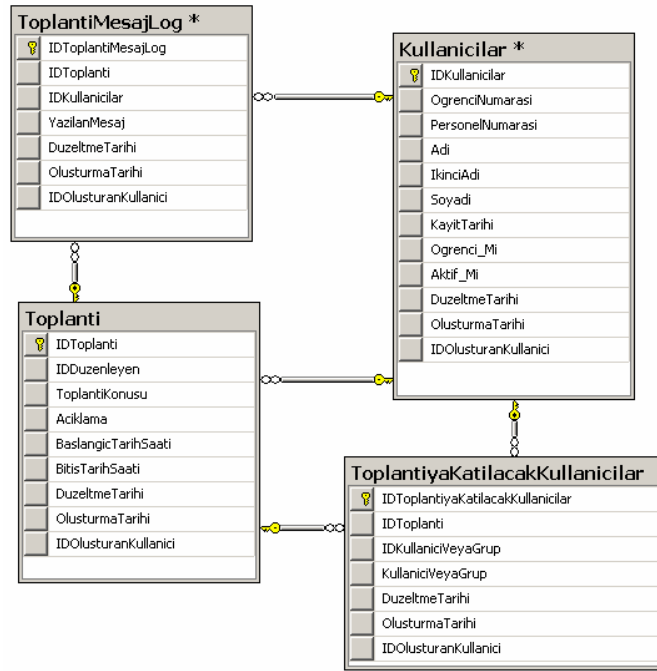


Şekil 4.25 Sohbet modülü tabloları ve ilişkileri

4.26 Toplantı Modülü

Toplantı modülü sistem yöneticisi, bölüm başkanı, danışman ve ders sorumlusunun sanal ortamda bir araya gelerek yapacakları görüşmeyi sağlamaktır. Sistem yöneticisi idari personeli yada bölüm başkanı bölümün öğretim elemanları ile görüşmek istediğinde, toplantı gününü ve saatini sistem aracılığı ile öğretim elemanlarına toplantı konusu ve başlığı ile duyurmaktadır. Toplantı sonuçları katılımcılara ayrıca gönderilmektedir. Toplantı kapsam olarak geniş ise diğer öğretim elemanları tarafından toplantı sonuçlarının görüntülenmesi sağlanabilir. Bunun için toplantılar konuları itibari ile listelenmektedir. Toplantı modülünde tanımlanan tablo ve ilişkiler şekil 2.26’da verilmiştir. Bu modül için 4 adet tablo tanımlanmıştır. Bunlar;

1. Kullanicilar
2. Toplantı
3. ToplantıMesajlog
4. ToplantıyaKatilacakKullanicilar

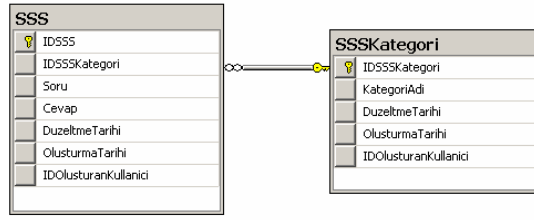


Şekil 4.26 Toplantı modülü tabloları ve ilişkileri

4.27 Sıkça Sorulan Sorular Modülü

Sıkça sorulan sorular modülü EÖYS hakkında ve çok sık sorulan diğer soruların cevapları kullanıcılara verilmesini sağlamaktadır. Bu modülde öğrencilerin sistemi nasıl kullanması gerektiğini ve kullanımı sırasında oluşabilecek aksamaların üstesinden gelmek için temel sorular ve bunların cevapları verilmiştir. Kullanıcılar belirli kategorilerde soruları seçebilmekte ve cevaplarını görebilmektedir. Sıkça sorulan sorular modülünün tablo ve ilişkisi şekil 4.27’de verilmiştir. Bu modülde 2 adet tablo tanımlanmıştır bunlar;

1. SSS
2. SSSKategori

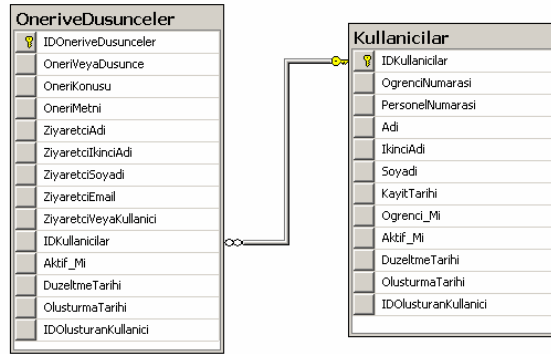


Şekil 4.27 Sıkça sorulan sorular modülü tabloları ve ilişkileri

4.28 Görüş Ve Öneriler Modülü

Görüş ve öneriler modülünde öğrenci, öğretim üyesi, sistem yöneticisi, bölüm sekreteri, ve misafirlerin her türlü görüş ve önerileri ifade edilmektedir. Kullanıcı sisteme giriş yaptıktan sonra görüş ve önerilerini sistem yöneticisine, ilgili idari yada akademik birime iletebilmektedir. Görüş ve öneriler modülünde “OneriveDusunceler” tablosu sadece kullanıcılar tablosu ile bire-sonsuz ilişki içerisindedir (Şekil 2.28). Bu modül için sadece 2 adet tablo tanımlanmıştır. Bunlar;

1. OneriveDusunceler
2. Kullanicilar



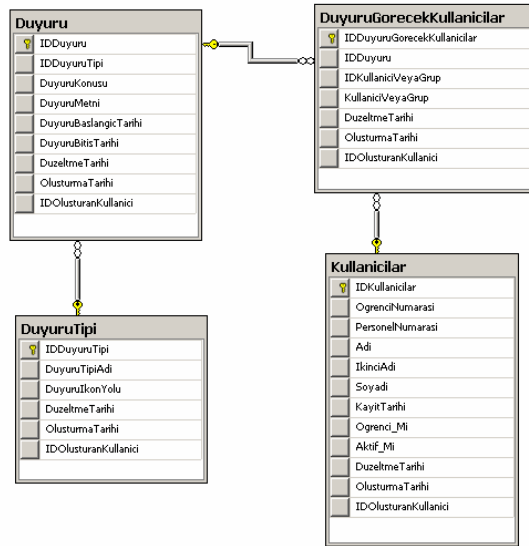
Şekil 4.28 Görüş ve öneriler modülü tabloları ve ilişkileri

4.29 Duyurular Modülü

Duyurular modülünde, eğitim öğretim dönemi içerisinde öğrencilere, öğretim elemanlarına yada belirlenen kullanıcı gruplarına duyurular yapılmaktadır. Dönem

sonlandırıldığında öğrencilere yapılan duyurular otomatik olarak sistemden silinmektedir. Duyuru kim tarafından yapılacaksa önce duyuru tipi, duyuru başlangıç ve bitiş tarihleri ve duyurunun yapılacak olduğu kişi yada guruplar seçilir daha sonra duyuru metni girilerek sisteme girilir. Duyurular otomatik olarak yada elle aktifleştirebilir. Duyuru, bitiş tarihinde otomatik olarak pasif olmaktadır. Duyurular modülü için tablolar ve ilişki senaryosu şekil 4.29’da verilmiştir. Bu modül için 4 adet tablo tanımlanmıştır. Bunlar;

1. Duyuru
2. DuyuruTipi
3. DuyuruyuGorecekKullanicilar
4. Kullanicilar

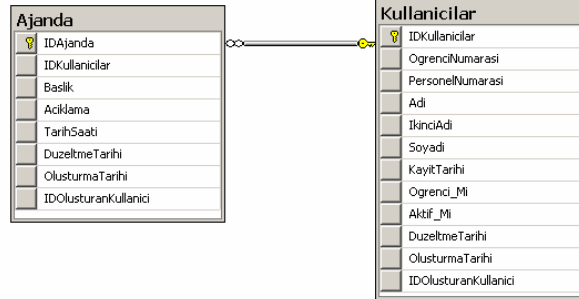


Şekil 4.29 Duyurular modülü tabloları ve ilişkiler

4.30 Ajanda Modülü

Ajanda modülü sisteme kayıtlı her kullanıcının tarih ve zamanı kullanarak kendine bir program yapmasını sağlayan modüldür. Kullanıcı randevularını notlarını kendi hesabında olan ajanda bölümüne kaydedebilir. Ajanda modülünde 2 adet tablo tanımlanmıştır (Şekil 4.30). Bunlar;

1. Kullanicilar
2. Ajanda

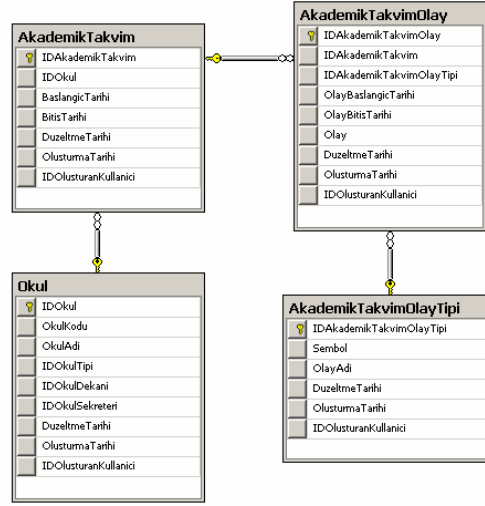


Şekil 4.30 Ajanda modülü tabloları ve ilişkisi

4.31 Akademik Takvim Modülü

Akademik takvim modülü yöneticilerin ve kullanıcıların görebileceği ve sistem yöneticisi tarafından kaydedilen eğitim öğretim takvimi modülüdür. Sistem yöneticisi okulların yeni kayıt ve ders kayıt dönemlerini, eğitimin öğretimin o dönem için başlangıç ve bitiş tarihlerini yılsonu sınav zamanlarını kaydettiği ve kullanıcılara duyurduğu takvimdir. Akademik takvim modülü tabloları ve ilişkileri şekil 4.31 de verilmiştir. Bu modül için 4 adet tablo tanımlanmıştır. Bunlar;

1. AkademikTakvim
2. AkademikTakvimOlay
3. AkademikTakvimOlayTipi
4. Okul

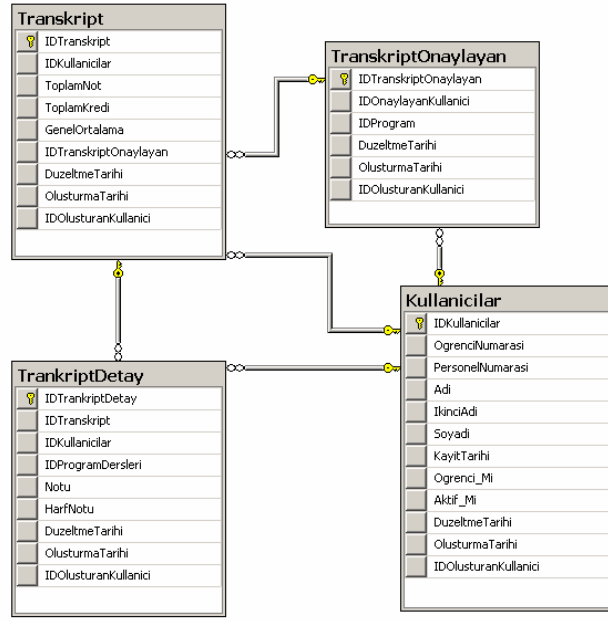


Şekil 4.31 Akademik takvim modülü tabloları ve ilişkileri

4.32 Transkript Görüntüleme Modülü

Transkript modülü eğitim öğretim dönemi içerisinde bir önceki döneme ait not, kredi ve disiplin durumlarının kullanıcıya yazıcı çıktısı olarak verildiği modüldür. Bu modülde öğrenci okuduğu dönemden önceki dönemlere ve hepsine ait toplam not durumunu talep ederek anında görebilir ve yazıcıdan çıktısını alabilir. Ancak transkript öğrenciye yazıcı çıktısı olarak verildiğinde mutlaka yetkililer tarafından onaylanması gerekmektedir. Transkript modülünün tablo ve ilişkileri şekil 4.32’de verilmiştir. Bu modül için toplam 4 adet tablo tanımlanmıştır. Bunlar;

1. Transkript
2. TranskriptOnaylayan
3. TranskriptDetay
4. Kullanıcılar

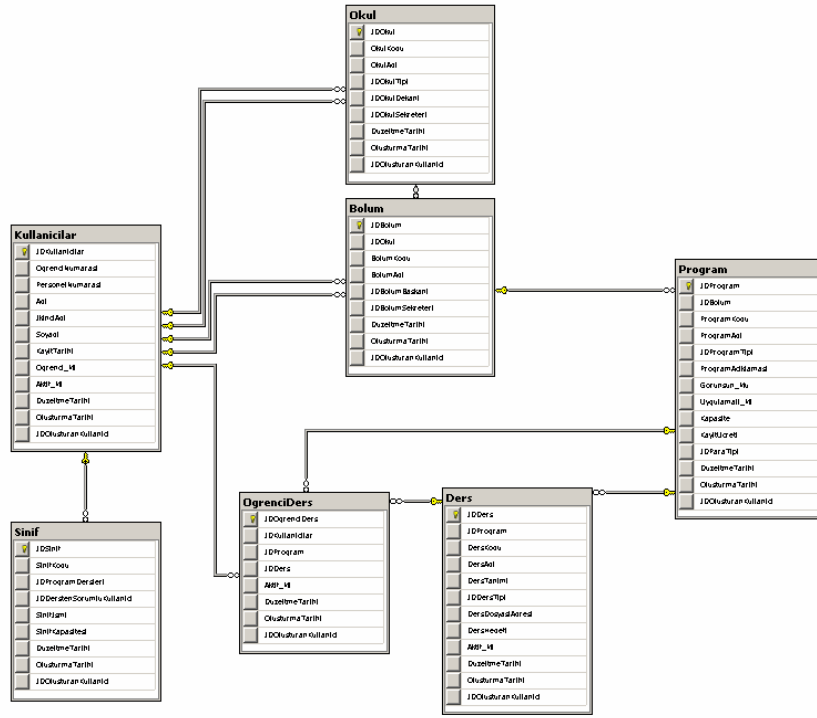


Şekil 4.32 Transkript modülü tabloları ve ilişkisi

4.33 Raporlama Modülü

Raporlama modülü sisteme kayıtlı her kullanıcının kişisel bilgilerini, transkript, okul, aldığı dersler, kredileri katıldığı toplantılar, sanal dersleri gibi sistem içerisinde yaptığı her bir olayın raporunu görüntüleyen modüldür. Bu modül için özel bir tablo tanımlanmamıştır, önceden tanımlı tabloların ilişki senaryolarından elde edilen ara-yüz ile yapılması için hazır hale getirilmiştir. Kullanıcı kendi kişisel bilgilerini ve sistem içerisinde yaptığı her işlemi veya işlemleri sonraki zamanlarda rapor olarak alabilmektedir. Raporlama modülü için kullanılan tablolar ve ilişki senaryosu şekil 4.33’de verilmiştir. Bu modül için 7 adet tablo tanımlanmıştır. Bunlar;

1. Okul
2. Bölüm
3. Program
4. Sınıf
5. Ders
6. ÖğrenciDers
7. Kullanicilar

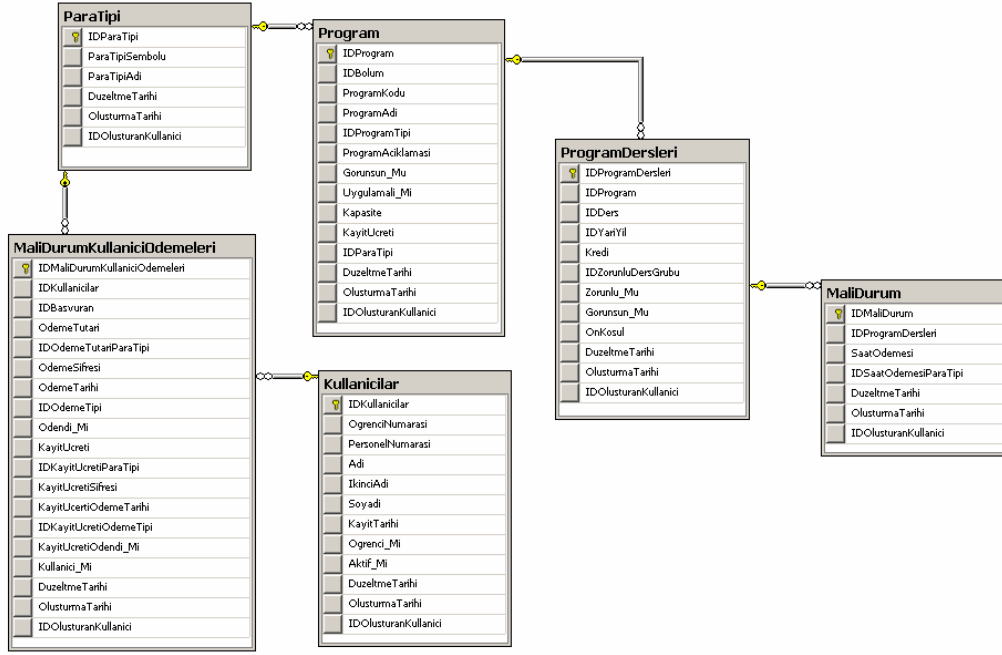


Şekil 4.33 Raporlama modülü tabloları ve ilişkileri

4.34 Mali İşlemler Modülü

EÖYS’inde, eğer bir eğitim kurumunda öğrenciler dönemlik harç yatırması gerektiği durumlarda, tek seferlik, dönemlik ya da yıllık harçların tutulduğu, ücretli bir kurs yada seminer düzenliğinde ücret bilgilerinin tutulduğu modül olarak işlev görmektedir. Mali durum modülü ilişki senaryosu şekil 4.34’de görülmektedir. Bu modülün tasarımında 6 adet tablo kullanılmıştır. Bunlar;

1. MaliDurum
2. ProgramDersleri
3. Kullanıcılar
4. Basvuran
5. MaliDurumKullaniciOdemeleri
6. Paratipi

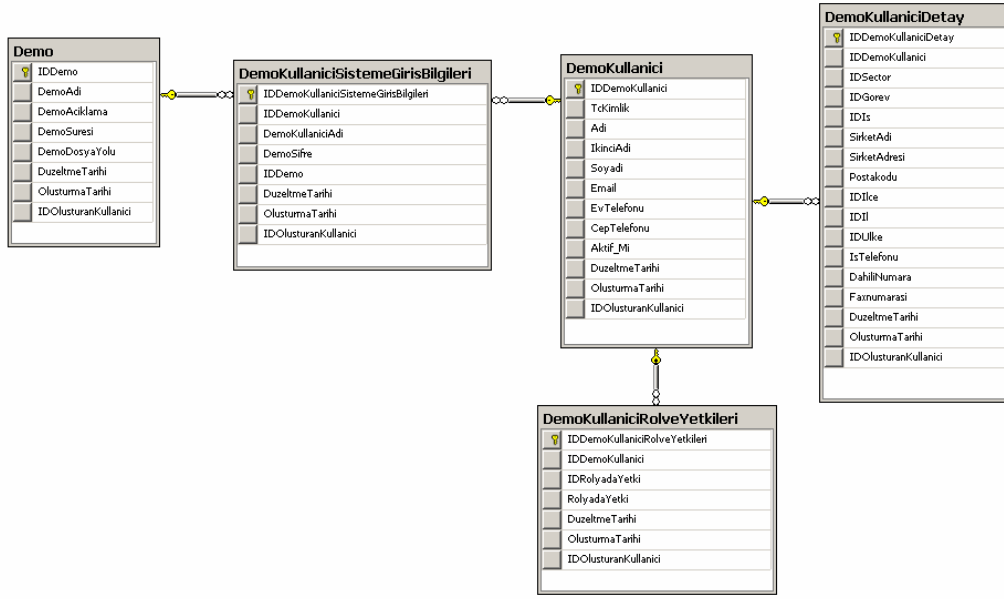


Şekil 4.34 Mali işler modülü tabloları ve ilişkileri

4.35 Demo Kullanıcı Modülü

Sisteme ait Demo modülü sistemi tam kayıtlı olmamış ve sistemi incelemek isteyen kişiler için tasarlanmış bir modüldür. Bu modüle göre kullanıcı sistemi sadece görebilmekte ve sistemde herhangi bir değişiklik yapamamaktadır. Demo kullanım için ilişki senaryosu şekil 4.35’de verilmiştir. Bu ilişki tasarımında 5 tane tablo tanımlanmıştır. Bunlar;

1. Demokullanici
2. DemoKullaniciDetay
3. DemoKullaniciSistemeGirisBilgileri
4. Demo
5. DemoKullaniciRolveYetkileri



Şekil 4.35 Demo kullanıcı modülü tabloları ve ilişkileri

4.36 Workshop (Çalıştay) Modülü

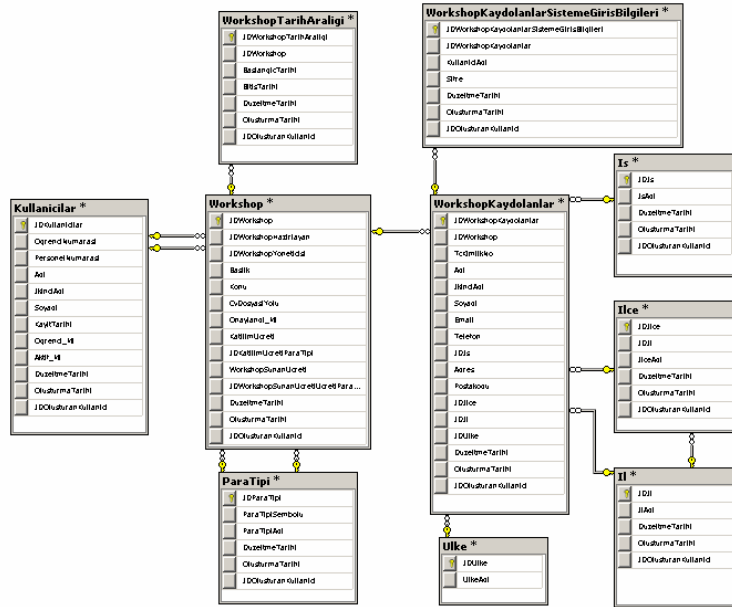
İngilizce Workshop olarak bilinen çalıştay EÖYS’ne alınmış bir modüldür. Bu ilişki senaryosunda 10 adet tablo kullanılmıştır. Bu modülün amacı; Eğitim öğretim sistemine kaydolmayan ziyaretçilerinde faydalanabileceği eğitim seminerlerinin düzenlendiği kısımdır. Bunun için bir örnek vermek gerekirse; genel bir Photoshop programı eğitimi düzenlenebilir bu eğitim normal eğitim sisteminden farklıdır. Bu sisteme kayıtlı öğrencilerin katılabileceği gibi aynı zamanda sistem dışındaki kullanıcıların katılımı da düşünülmüştür. Öğrenci olmayan ziyaretçiler sisteme “Seminer” ziyaretçisi olarak kaydolmakta ve bu imkanlardan yararlanabilmektedir.

Bu çalıştay senaryosu için kullanılan tabloların başlıkları şunlardır.

1. WorkshopKaydolanlar
2. WorkshopKaydolanlarSistemeGirisBilgileri
3. WorkshopTarihAraligi
4. Workshop
5. Kullanicilar
6. ParaTipi

7. Is
8. İl
9. İlçe
10. Ülke

Şekil 4.36’da Workshop (Çalıştay) ilişki senaryosu verilmektedir. Buna göre ilişki senaryosu “Workshop” ve “WorkshopKaydolanlar” tablosu üzerine kurulmuştur. Bu iki tablo ve gruba ait diğer tablolar arasında bire-sonsuz (1-∞) ilişki olduğu görülmektedir.

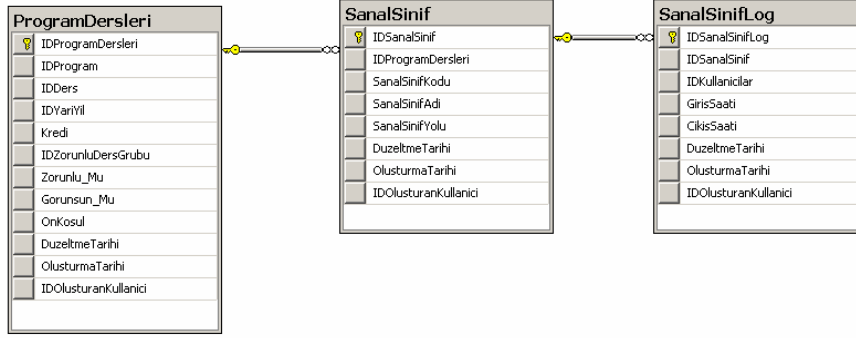


Şekil 4.36 Çalıştay (Workshop) tabloları ve ilişkileri

4.37 Sanal Sınıf Modülü

Sanal sınıf öğrencilerin sanal ortamda bir araya gelip dersin öğretim elemanı ile görüştükleri, dersin öğretim elemanına soru sordukları ve diğer öğrenci arkadaşları ile tartıştıkları bir sınıf platformudur. Sanal sınıf öğrencilerin almış oldukları dersler için oluşturulabilir, yani dersin öğretim elemanı sınıfı açar ve o dersin alan öğrencilerin sayfalarında sanal sınıfın açıldığı görülür ve belirtilen zamanda sanal sınıfa katılarak dersi dinlerler.

1. SanalSinif
2. SanalSinifLog
3. ProgramDersleri

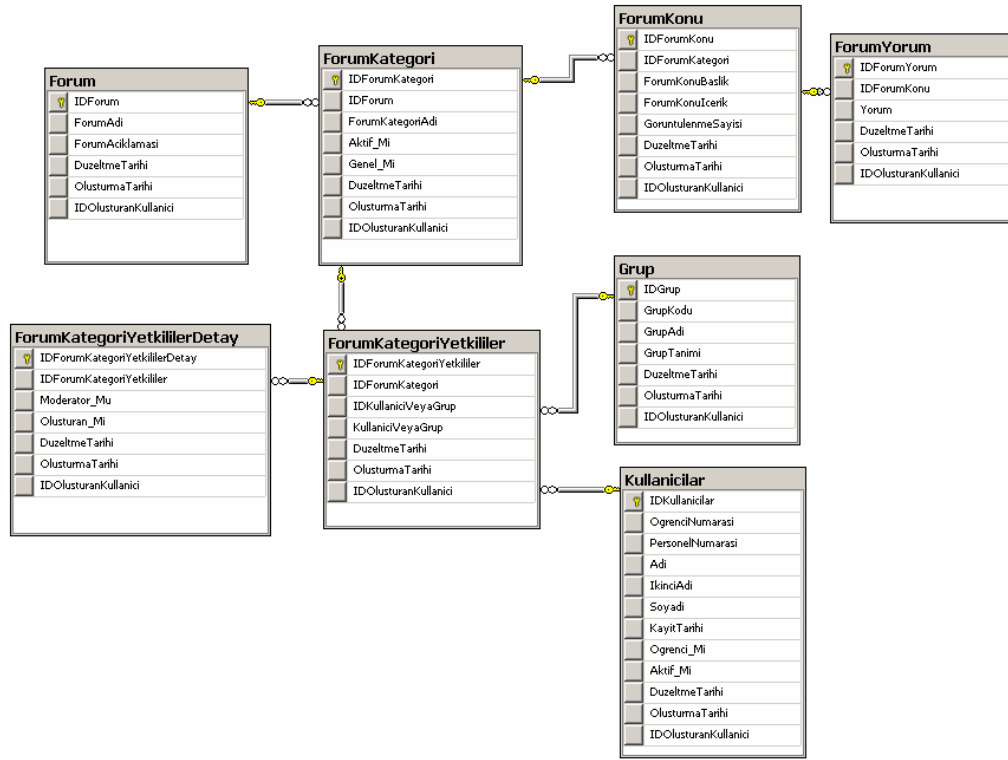


Şekil 4.37 Sanal sınıf modülü tabloları ve ilişkileri

4.38 Forum Modülü

EÖYS’de forum modülünde kullanıcıların bir tartışma ortamı oluşturulmuş ve bilgi paylaşımı sağlanmıştır. Bu modül sisteme kayıtlı kullanıcılar ile ilişkilendirilmiştir. Sisteme giriş yapan her kullanıcı aynı zamanda bilgi paylaşımına da katılabilmektedir. Forum modülü için ilişki senaryosu şekil 4.38’de verilmiştir. Aynı zamanda ilişki senaryosu için 8 adet tablo kullanılmıştır. Bunlar;

1. Forum
2. ForumKategori
3. ForumKodu
4. ForumKategoriYetkililerDetay
5. ForumKategoriYetkililler
6. Kullanicilar
7. Grup
8. ForumYorum



Şekil 4.38 Forum modülü tabloları ve ilişkileri

4.39 Sisteme Erişim, Rol ve Yetkiler

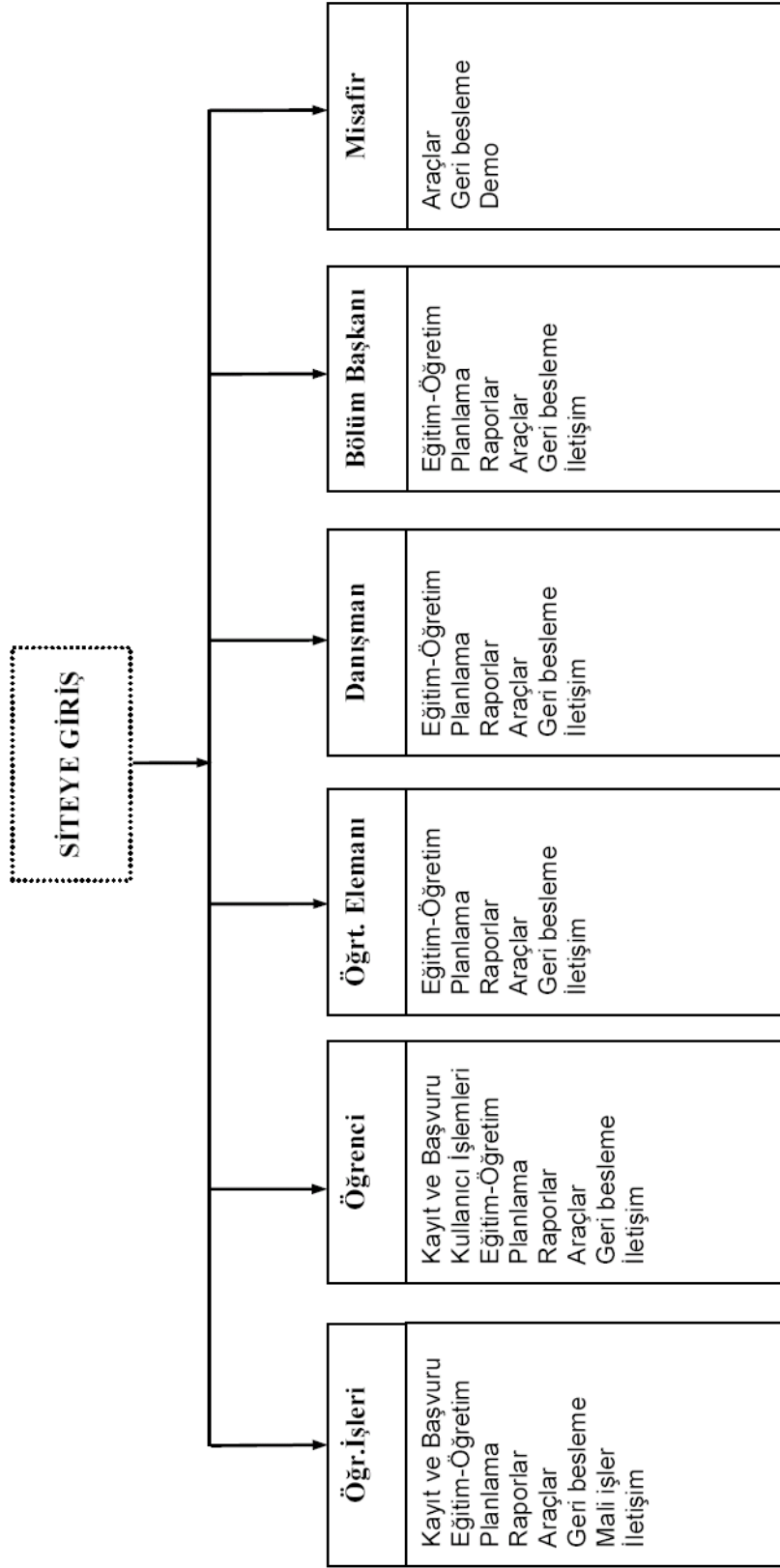
EÖYS için rol yetkiler site yöneticisi (Admin) tarafından tanımlanmaktadır. Bu tanımlama için herhangi bir sınırlama yoktur. Yönetici yetki tanımlamaları, sistem içinde tanımlanmış kişilere belirli modüllere erişmeleri için verilmiş izin ve rollerdir. Bir kişi kendi yetkileri ile yetki sınırları çerçevesinde işlem yapabilmektedir. Sistem içinde örnek bir rol ve yetki tanımlamaları yapılmıştır bunlar; Öğrenci işleri, bölüm başkanı, ders öğretim elamanı, öğrenci, danışman ve misafirdir. Sistem yöneticisi (Admin) sistemdeki tüm modüllere erişim (oluşturma, değiştirme ve silme) yetkilerine sahip olduğundan bu seviye için ekstra bir açıklamaya gerek yoktur. Diğer rol ve yetkilerin kısa açıklamaları şunlardır.

1. **Öğrenci İşleri:** Öğrencilerin kayıt ve başvuru işlemlerinin takibinin yapıldığı, eğitim ve öğretimin düzenlendiği, zaman ve takvim planlamalarının yapıldığı, öğrenci raporlarının alındığı, geri beslemelerin değerlendirildiği, mali durumun

takip edildiđi ve sanal iletiřimin sađlandığı bir rol ve yetki gurubudur. Adından da anlaşılacağı üzere bu yetki öğrenci işlerinde çalışan elemanlara verilmiştir.

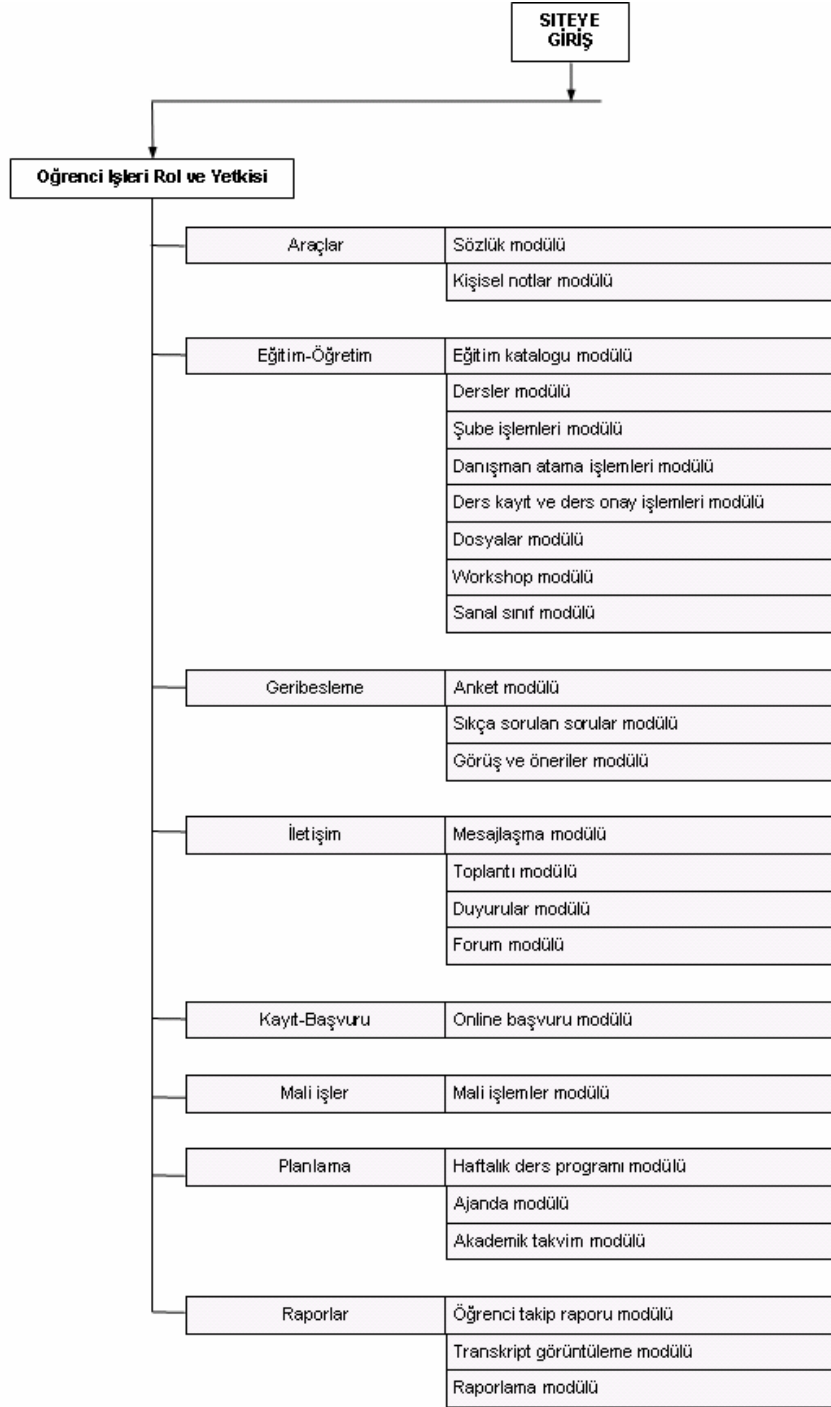
2. **Öğrenci:** Öğrenciler kendine ait modüllerle kayıt ve başvuruda bulunabilirler, kendilerine ait kullanıcı işlemlerini yapabilirler, eğitim öğretimi devam ettirirler ve eğitim öğretime ait modülleri kullanırlar. Kendilerine ait planlama yapabilirler örneğin ajanda oluşturabilirler. Geri besleme araçlarını kullanarak hocalarına bilgi aktarabilirler. Derslerinin ve sınav sonuçlarının raporlarını alabilirler.
3. **Öğretim Elemanı:** Kendine ait derslerin planını yapar ve eğitim-öğretim içerisinde sanal sınıfını kullanır, derslerini yapar, sınavlarını yapar, raporlarını alır ve geri beslemeyi kullanır. Öğrencilerle veya diđer öğretim elemanları ile iletişim kurar.
4. **Bölüm Başkanı:** Bölüm başkanı eğitim-öğretim içerisinde kendine ait modülleri kullanır. Bölümün alt programlarını oluşturur, dersleri tanımlar, sınıflara danışmanlar atar ve ayrıca planlama, raporlama iletişim ve araçları kullanır.
5. **Danışman:** Kendine ait sınıfın ders kayıt işlemlerini takip eder, gerekli raporları hazırlar. Planlama iletişim ve araçlar modüllerini de kullanabilir.
6. **Misafir:** Siteye kayıtlı olmayan kullanıcılar misafir olarak tanımlanır. Sistemi demo kullanıcı ile test eder, sitenin genel bilgiler kısmını görebilir, akademik takvim ve duyuruları görerek sistemi sınırlı olarak kullanır.

Şekil 4.39’da tanımlanan belirli roller için yaptıkları görevler kısa adları ile şematik gösterimi verilmektedir.



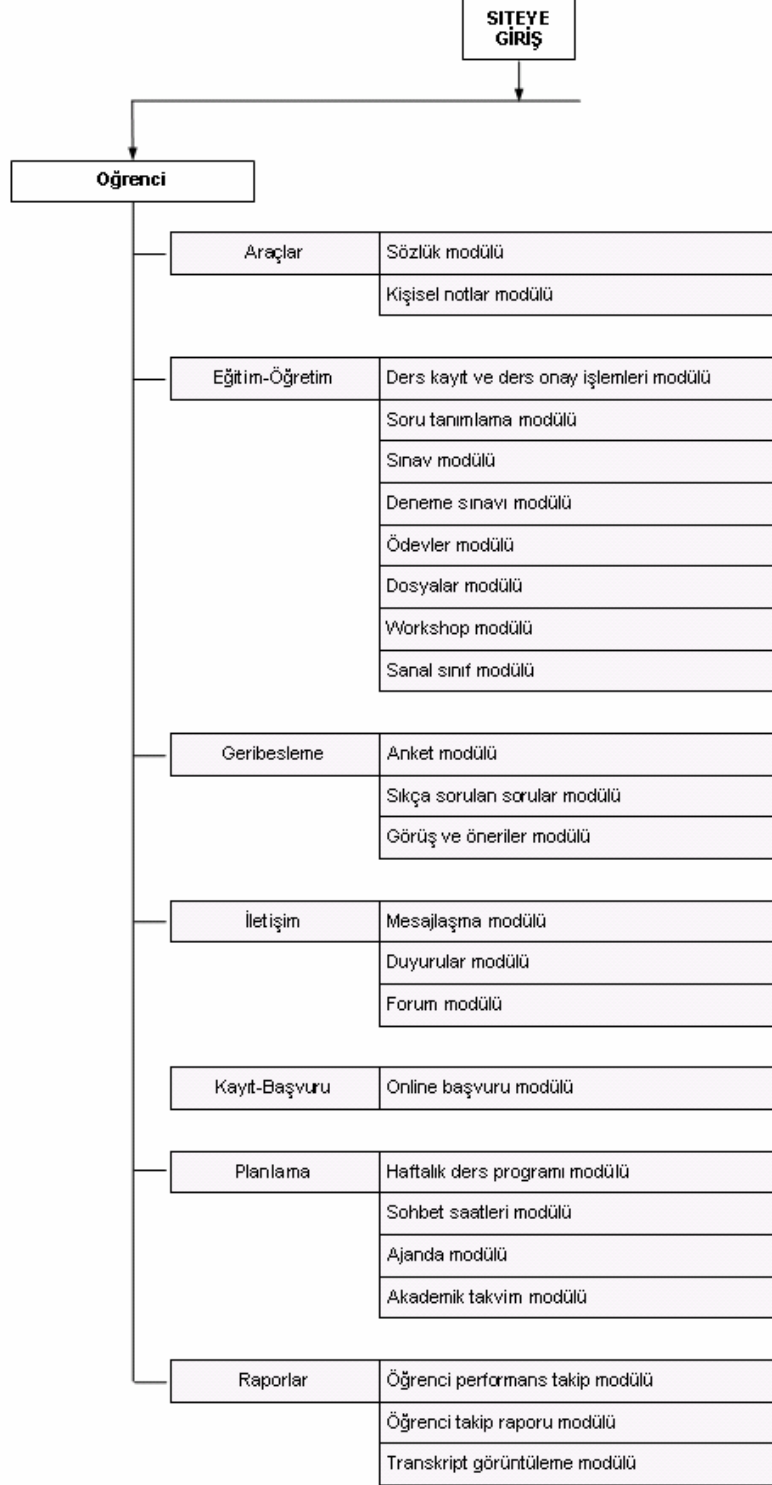
Şekil 4.39 Sisteme erişim rolleri ve görevleri

Şekil 4.40’da sisteme erişim rolleri içerisinde Öğrenci İşleri rol ve yetkisinin kullanabileceği modüller verilmiştir. 38 modül içerisinde 25 tanesini kullanmaya yetkilidir.



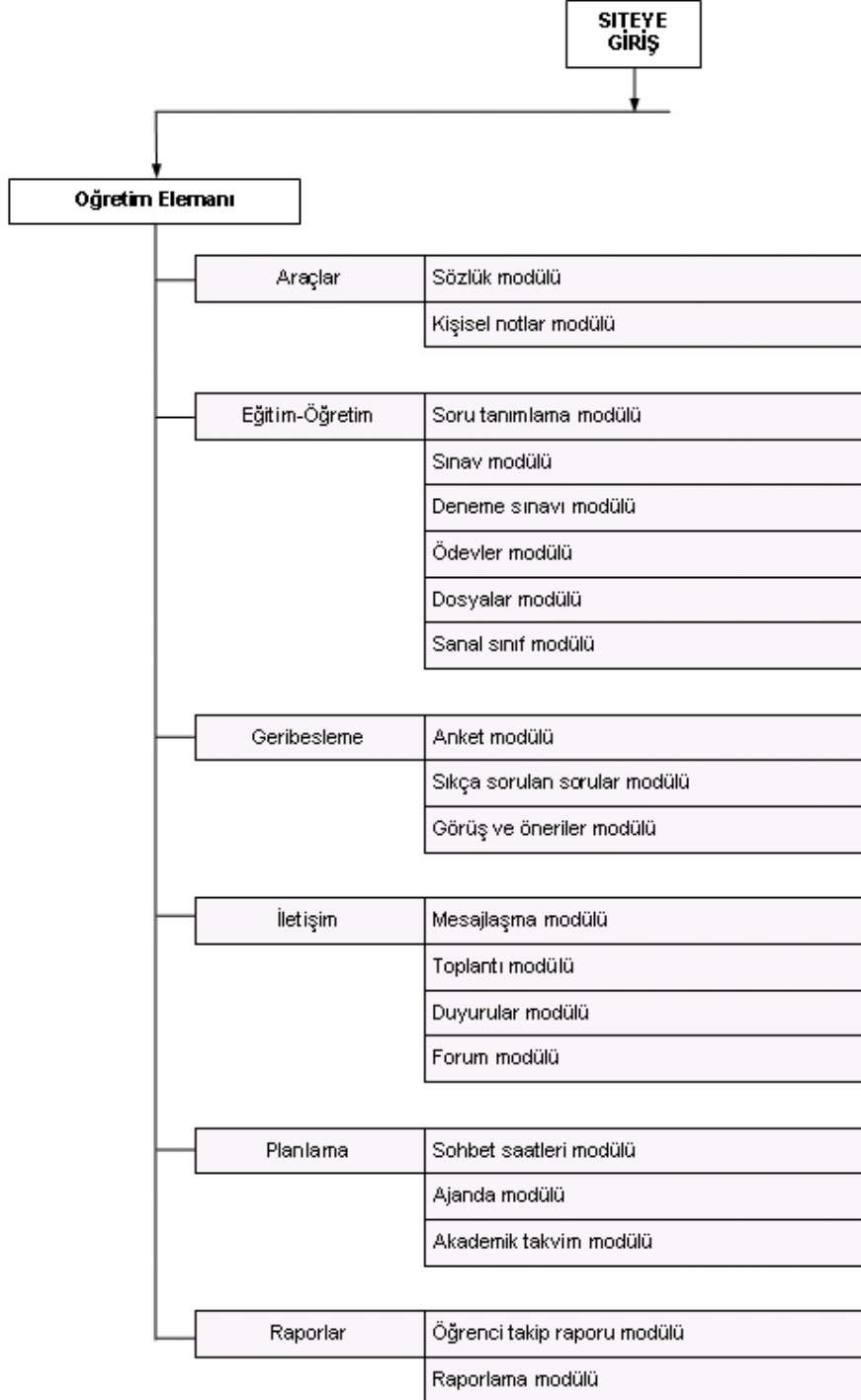
Şekil 4.40 Öğrenci işleri rol ve yetkisinin erişebileceği modüller

Şekil 4.41 da sisteme erişim rolleri içerisinde Öğrenci rol ve yetkisinin kullanabileceği modüller verilmiştir. 38 modül içerisinde 24 tanesini kullanmaya yetkilidir.



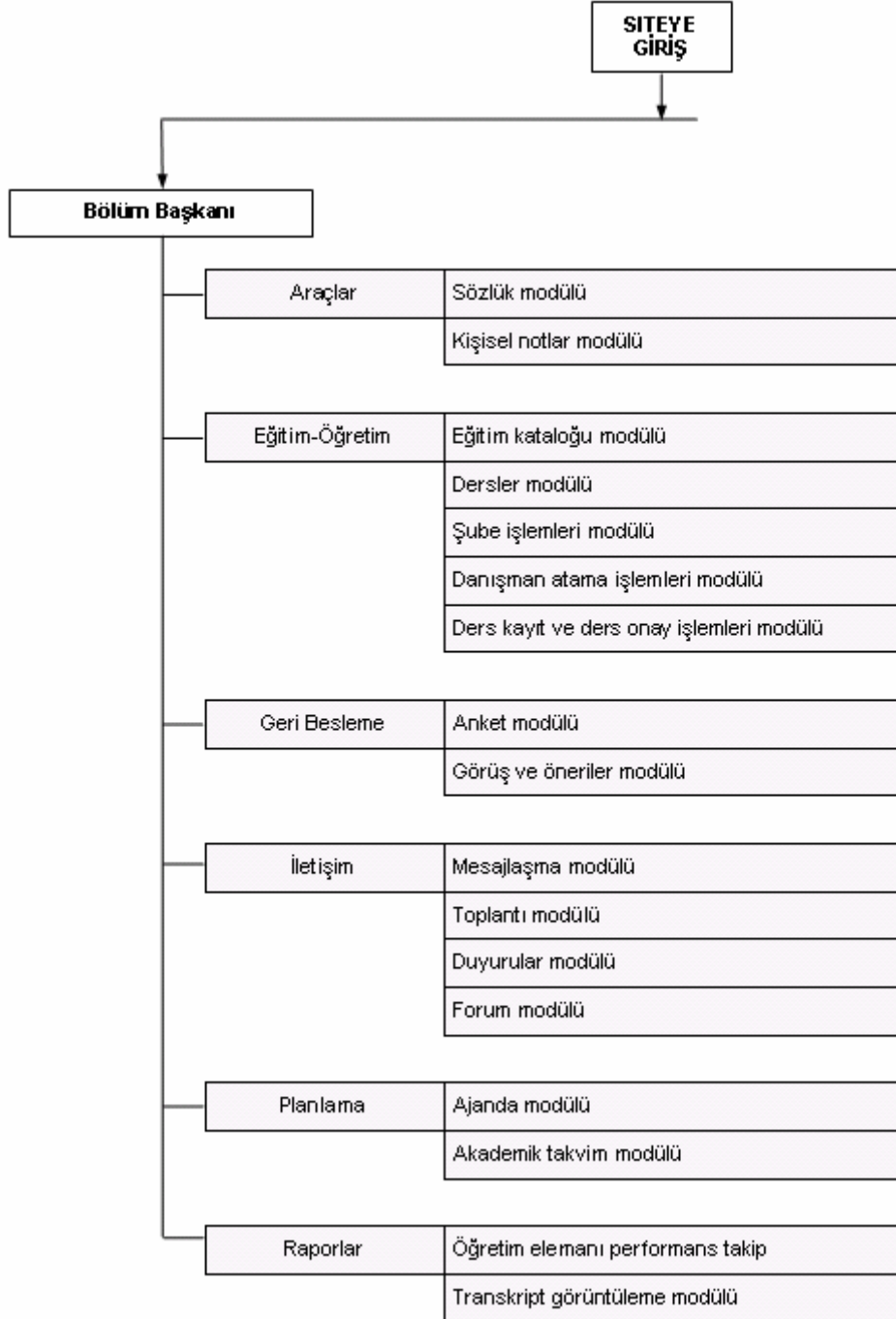
Şekil 4.41 Öğrenci rol ve yetkisinin erişebileceği modüller

Şekil 4.42’de sisteme erişim rolleri içerisinde dersin öğretim elamanı rol ve yetkisinin kullanabileceği modüller verilmiştir. 38 modül içerisinde 20 tanesini kullanmaya yetkilidir.



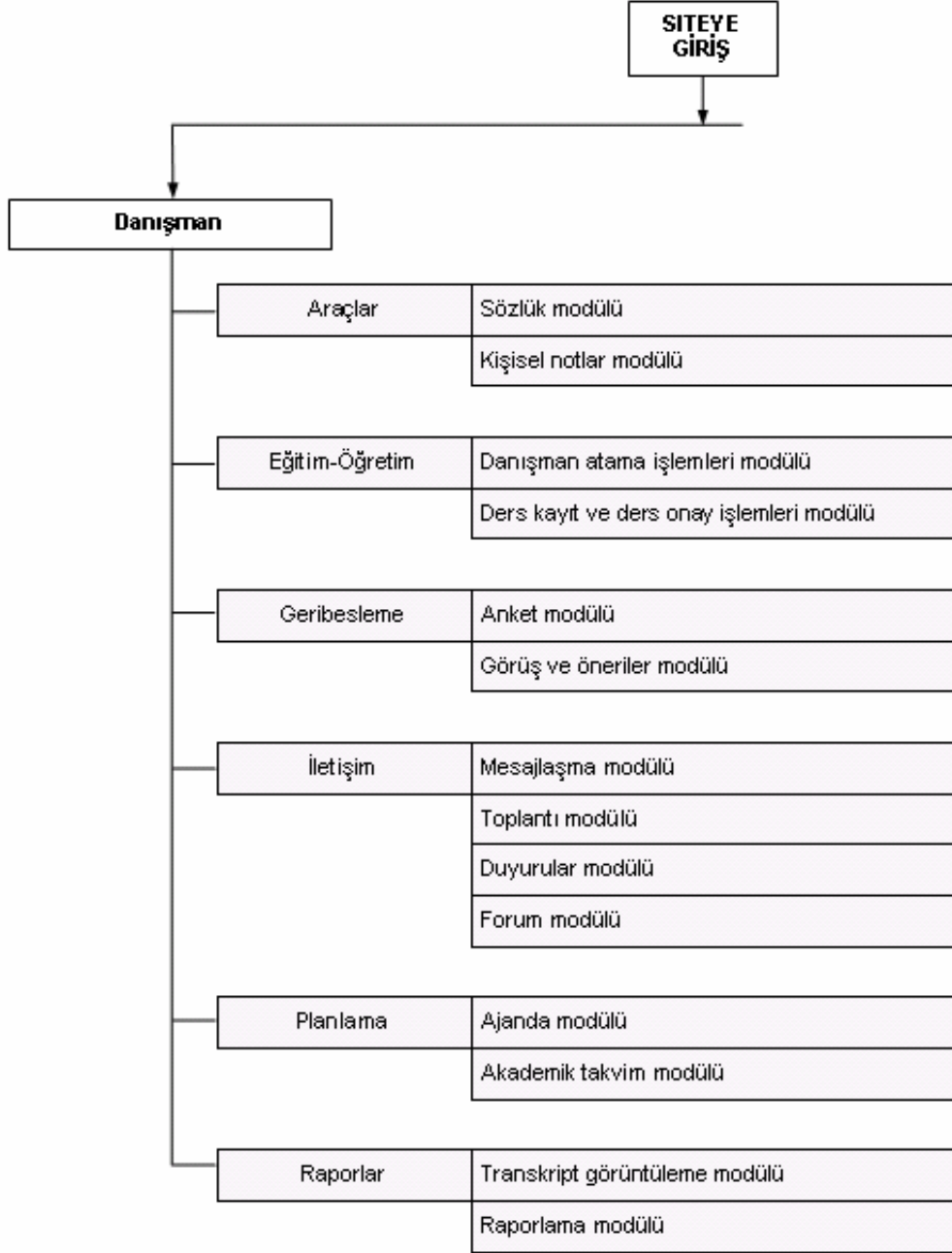
Şekil 4.42 Öğretim elamanı rol ve yetkisinin erişebileceği modüller

Şekil 4.43’de sisteme erişim rolleri içerisinde bölüm başkanı rol ve yetkisinin kullanabileceği modüller verilmiştir. 38 modül içerisinde 17 tanesini kullanmaya yetkilidir.



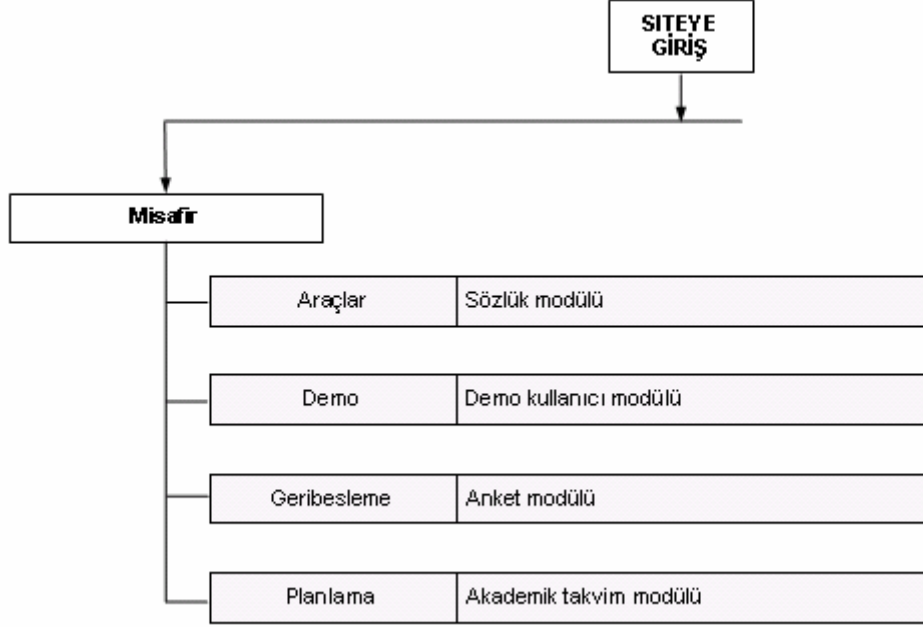
Şekil 4.43 Bölüm başkanı rol ve yetkisinin erişebileceği modüller

Şekil 4.44’de sisteme erişim rolleri içerisinde danışman rol ve yetkisinin kullanabileceği modüller verilmiştir. 38 modül içerisinde 14 tanesini kullanmaya yetkilidir.



Şekil 4.44 Danışman ve yetkisinin erişebileceği modüller

Şekil 4.45’de sisteme erişim rolleri içerisinde sisteme kayıtlı olmayan ziyaretçilerin (Misafir) kullanabileceği modül verilmiştir. 38 modül içerisinde 4 tanesini kullanmaya yetkilidir.



Şekil 4.45 Misafirlerin erişebileceği modüller

5. SONUÇ VE ÖNERİLER

Web tabanlı EÖYS'leri günümüz teknoloji çağında eğitimi kişilerin ayağına getirdiği sistemlerdir. Bilgiye ulaşmak kolaylaşmış aynı zamanda bilginin karşılıklı paylaşılması ve tartışılması sağlanmıştır. Kullanıcılar evlerinde oturarak sanal ortamda eş zamanlı görüşebilmekte, iletişim kurabilmekte ve bilgiye kolayca ulaşabilmektedir. Bilgiye ulaşmada uzaklık kavramı internet teknolojileri ile ortadan kalkmıştır.

EÖYS için tasarlanan veritabanı uzaktan eğitimin özelliklerini taşıyan aynı zamanda eğitim kurumu düzeyinde yönetilebilir yapıya sahip bir model olarak tasarlanmıştır. Veritabanı tasarımı yapılırken gerçek uygulamalarda tespit edilen, öğrenci ve öğretim elemanı performansı, anket, forum, sözlük gibi eksiklikler eklenmiş ve geliştirilmiştir. Her bir modül kendi içerisinde bir bütün olarak çalışmaktadır ve aynı zamanda diğer modüllere de doğrudan bağlantılıdır. Ek2 de verilen veritabanı diyagramında da görüldüğü gibi veritabanı sistemi kullanıcıları merkeze almıştır. Kullanıcılar öğrenciler, öğretim elemanları idareciler ve idari personeller olarak ayrı ayrı görünse de tek bir tabloda toplanmıştır. Ancak kullanıcılara verilen rol ve yetkiler farklıdır.

Yapılan çalışmada detaylarının verildiği tam ilişkisel bir EÖYS tasarlanmış ve bir ara-yüz için kullanıma hazır hale getirilmiştir. Yapılmış olan veritabanı tasarımı SCORM kriterlerine göre değerlendirildiğinde uyumu görülmektedir. Buna göre;

- 1. Birlikte çalışabilirlik** ilkesine uygunluğu; hazırlanan veritabanı modelinin istenilen ara-yüz yazılımına uygun olmasıdır. WEB formu kullanılmasına uygun olarak tasarlanmış model aynı zamanda Windows form ara-yüzü tasarımına da uygundur. Sistemin çıktıları farklı modüllere kolayca entegre edilebilmektedir.
- 2. Yeniden kullanılabilirlik** ilkesine uygunluğu; sistem içerisinde kullanılan öğrenim materyallerinin (metin, grafik, ses, animasyon, video, kod) nizami bir düzen içerisinde yapıldığından ve bu nesnelere kullanımının kolay olduğundan tekrar-tekrar kullanılabilmesi mümkün olabilmektedir.

3. Yönetilebilirlik ilkesine tam uygunluğu sağlanmıştır. Yapılan tasarım içerisinde farklı kullanıcı rollerinin tanımlanması sistemin esnek bir şekilde kullanılmasını sağlamaktadır. Örneğin, dersin öğretim elemanı ders notlarını hazırlayabilmekte, isterse deneme sınavı düzenleyebilmekte, vize ve final sınavlarını ve bu sınavların türlerini de belirleyebilmektedir. Bir idarecinin rolü, sistem içerisinde idari işleyişin kontrolünün yapılmasını sağlamaktır. Öğretim elemanlarının atanması, bu öğretim elemanlarına derslerin ataması ve öğrencilerin son notlarını alıncaya kadarki süreci kontrol etmesidir. Bir öğrencinin sistem içerisindeki rolü ders notlarını almak, sınavlara katılmak, notlarını takip etmek ve transkriptlerini çıkartmak vs gibi işlemleri yapmasıdır. Böylece sistemin farklı rollerle yönetilebilirliği tam olarak sağlanmış olmaktadır.

4. Ulaşılabilirlik ilkesine uygunluğu sistemin internet tabanlı olması nedeniyle kullanıcının her an ulaşması sağlanır. Bir web formu ara yüzü ile kullanıcıya sunulduğunda günün her saatinde ulaşılması mümkündür.

5. Devamlılık ilkesine uygunluğu tam olarak sağlanmıştır. Tasarlanan veritabanı tamamen yeniliğe ve geliştirilmeye açıktır. Veri tabanına yeni modül eklemeleri yapılabilir. Yapılan yenilikler kullanıcı ara yüzü tasarımı ile internet ortamından eş zamanlı olarak görülebilmektedir. Kullanıcının herhangi bir ek bileşen kurmasına gerek yoktur.

6. Ölçeklenebilirlik ilkesine sınırsız sayıda ders içeriği, dersin kendisi, öğrenci ve öğretim elemanı artışını kaldırabilecek kapasitede olması ile sağlanmaktadır. Sistem gerek içerik gerekse tablo ve modüllerin sınırsız sayıda eklenebilirliğini kaldırabilmektedir.

5.1 Yapılan Çalışmanın Eğitime ve Bilime Katkısı

Yapılmış olan tez çalışması ile şimdiye kadar tasarlanmamış EÖYS'inin tam ilişkisel veritabanı tasarımı yapılmıştır. Her eğitim kurumu bu modülünü kullanarak kendi sistemini bu tasarıma hemen entegre edebilir. Açık kaynak kodlu olduğu için kendi

amaçları doğrultusunda değiştirebilir, bileşenleri eklenebilmekte yada çıkartılabilmektedir.

Geniş ölçekli eğitim sistemleri için veri bütünlüğü ve veri güvenliği sağlanmış, tam ilişkisel tasarımı ile mükerrer kayıtların girişleri engellenmiş, gereksiz veri yığılmalarının önüne geçilmiştir. EÖYS için ilişki senaryoları oluşturulmuş ve herhangi bir ara-yüz formu için hazır hale getirilmiştir.

Tasarlanan veritabanı ASP.NET teknolojisi ile WEB ara-yüz kullanımı için uygun olmakla birlikte yazılımcının veritabanı tasarım işlemlerini ortadan kaldırmıştır. ASP.NET teknolojisi MS SQL server veritabanı yönetim sistemi ile uyumlu geliştirilmesi nispeten kolay olan esnek bir teknolojidir. Bu teknolojiyi kullanan yazılımcılar bu veritabanı üzerinde kolayca çalışma yapabilmektedir.

Yazılım geliştiriciler tasarlanan modeli test etmek, daha iyi anlayabilmek ve kullanıma hazır hale getirmek için tablolara veri girmeli ve yapısal sorgulama dili ile temel sorgulama işlemleri yapmalıdır. Bunun için öncelikli olarak kullanıcı tablosu öğrenci kayıtları ile doldurulur, daha sonra aynı tablo üzerine öğretim elemanları ve idari personel eklenir. Kullanıcılar tablosuna eklenen kişilere öğrenciler, öğrenci işleri, öğretim elemanı, bölüm başkanı gibi rol ve yetkiler atanır. İlk olarak kullanıcı ve kullanıcı grupları modülü ile rol ve yetkiler modülü tabloları örnek verilerle doldurulur.

Kullanıcılar tanımlandıktan sonra örnek bir okul tanımlanır, örneğin “Teknik Eğitim Fakültesi”, daha sonra bu okula ilişkili bölümler tanımlanır, örneğin “Bilgisayar Bölümü” aynı bölüm içerisine eğitim programları tanımlanır örneğin “Bilgisayar Öğretmenliği” programı. Eğitim katalogu modülü için dersler tanımlanarak ve dersler modülü tabloları örnek verilerle doldurulur. Bu derslere öğretim elemanları atama işlemleri yapılabilir. Daha sonra veriler sorgulama işlemleri için hazır hale gelir. Bu verilerden elde edilecek çıktılar önceden belirlenir ve bu çıktılara göre sorgulamalar yapılır. Bu sorgulama sırasında ortaya çıkabilecek sorunlar eş zamanlı olarak veritabanı üzerinde çözülebilir. Böylece hata ayıklama işlemi de yapılmış olunur.

Bu işlemler adım adım yapıldıktan ve ortaya çıkan hatalar çözümlendikten sonra, sırası ile diğer modüller ve bu modüllerin tabloları örnek verilerle doldurulur. Veritabanındaki her bir modül birbirlerinden bağımlı ve bağımsız olarak çalıştırılarak denenir ve sorgulanır.

Bu çalışma ile sadece uzaktan eğitim hazırlamanın yanında veritabanı tasarımcılarına yol gösterici bir rehber niteliğindedir. Tasarlanan veritabanına, tasarımcının 8 yıllık mesleki birikim ve tecrübeleri eklenmiştir. Eğitim öğretimin gerçek uygulamasında ortaya çıkan sorunlar ve bu sorunların çözümleri veritabanına yansıtılarak geliştirilmiş böylece veritabanının güvenilirliği daha da artmıştır. Bu çalışma normal ilişkisel veritabanı tasarımı içinde örnek bir çalışma modeli olarak kullanılabilir.

Yapılan uygulama bir uzaktan eğitimde olması gerekenler düşünülerek hazırlanmıştır. Örnek bir model olarak her eğitim kurumunda uzaktan eğitim yapacaklar için bir kaynak ve başlangıç noktası niteliğindedir. Bu şekliyle uzaktan eğitimi yüzeysel yapacaklar için çok karmaşık ve gereğinden fazla modül içeriyor olabileceği gibi, bazı kurumlar içinde yetersiz olabilecektir. Bu nedenle bu projeyi kaynak olarak kullanacak kurumlar kendilerine yetebilecek kısımlarını aynen alabileceği gibi, gereksiz gördükleri modülleride çıkarabileceklerdir.

Sonuçta bu şekliyle alıp üzerine kurum ihtiyacına göre gerekli eklemelerde yapıldıktan sonra proje geliştirme aşamasına çok hızlı bir şekilde gelinecektir. Örnek model uzaktan eğitim yapacak kurumların kendi projelerini hazırlamadan önce nelerin olması gerektiğini görmeyi sağlayacağı için hazırlamış olan bu projede harcanan zaman kadar zaman kazanmış olacaklardır.

Bu örnek modeli alıp proje hazırlayacak olan kurumlar öncelikle; herhangi bir proje hazırlamada olmazsa olmaz olan ekip çalışmasını benimsemeli ve projeyi birden fazla parçaya bölerek her bir parça için bir ekip oluşturmalıdır. Bu ekipler içerik ekibi, yazılım ekibi, görsel tasarım ekibi ve analiz ve veritabanı ekibi olabilir. Her bir ekip kendi alanında uzman ve uyumlu kişilerden seçilmelidir. Ekipler arasında koordinasyonu sağlayacak birde yönetici belirlenmelidir. Oluşturulan bu ekibin bilgi

eksiklikleri gerekli eğitimlerle giderilmelidir. Bütün bunlar yapıldıktan sonra her ekip için proje takvimi oluşturulmalı ve bu takvim çerçevesinde proje hazırlanmalıdır.

Bir uzaktan eğitim modeli hazırlayacak olan kuruluş bu modeli temel alarak kendi projesini oluşturabilir. Bu proje üzerine eğitim-öğretime başlamak için akıllı tahta, kamera, webcam, projeksiyon cihazları ve bunun gibi eğitim materyalleri ile sanal sınıflar kurularak proje hayata geçirebilmektedir.

6. KAYNAKLAR

Aytaç, V., 2002 “İnternet Destekli Uzaktan Eğitim Ve Öğretim Teknolojisi”, Doktora Tezi, Ege Üniversitesi Fen Bilimleri Enstitüsü, İzmir, 1-92.

Bekatlı, O., 1992, “Veritabanı Tasarım Yöntemlerinin Karşılaştırılması Ve Bir Örnek Uygulama” Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul

Burma, Z., A., 2004 “Veritabanı Yönetim Sistemleri I” Mersin Üniversitesi Yayınları, Mersin

Coşkun, C., 2007 “Uzaktan Eğitim İçin Web Tabanlı Bir Platform Geliştirilmesi Ve Mekanik Derslerine Uygulanması”Gazi Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, Ankara

Çetiner, M.H., Erten, Y.M., Gencel, Ç.,1999, “İnternete Dayalı Uzaktan Eğitim ve Çoklu Ortam Uygulamaları” V. Türkiye de İnternet Konferansı., Ankara. Erişim: “<http://inet-tr.org.tr/inetconf5/tammetin/gencel-egit.doc>” Erişim Tarihi: 05.07.2008.

Çinici, M., A., 2006 “Web Tabanlı Uzaktan Eğitimde Uyarlanı Değerlendirme Sistem Tasarımı ve Gerçekleştirimi” Hacettepe Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, Ankara

Çokçetin, B., 2006, “Php, Mysql Tabanlı Uzaktan Eğitim Modülü Tasarımı” Dumlupınar Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, Kütahya.

Çubukçu, F., 1999, “Microsoft SQL Server 7.0 Sistem Yönetimi ve Geliştirme”, Alfa Basın Yayın Dağıtım, 4-30

Ertem A.A., 2007 “Aile Hekimlerinin Eğitiminde Uzaktan Eğitim Yönteminin kullanılabilirliği (Düzce İli Örneği)” Ankara Üniversitesi Sağlık Bilimleri Enstitüsü Yüksek Lisans Tezi, Ankara

Gözüdeli, Y., 2004 “Veritabanı Programlama II, Byte Dergisi Eğitim Dizisi”, Acar Yayıncılık, İstanbul

Gözüdeli, Y., 2007 “Yazılımcılar için SQL Server 2005 Veritabanı Programlama”, Seçkin Yayıncılık, 2. Baskı 49-51 Ankara

Gözüdeli, Y., 2003 “Veritabanı Programlama I, Byte Dergisi Eğitim Dizisi”, Acar Yayıncılık, İstanbul

Gülner, B., 2003 “Bilgisayar ve İnternet Destekli Uzaktan Eğitim Programlarının Tasarım Geliştirme ve Değerlendirme Aşamaları (SUZEP ÖRNEĞİ)” Selçuk Üniversitesi Sosyal Bilimler Enstitüsü Yüksek Lisans Tezi, Konya

İlgin, D., E, 2004, “Nesneye Yönelik Tasarım ile Nesne-İlişkisel Veritabanı Geliştirilmesi” Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul

İşman A., 2001 “Sakarya Üniversitesinde Uzaktan Eğitimin Dünü Bugünü ve Geleceği” Sakarya Kongresi, <http://www.ef.sakarya.edu.tr/dergi/efdergisayi3-1f.pdf>, 2001 Sakarya

İşman A., 2005 “Uzaktan Eğitim”. Öğreti Yayınları, 2005, Ankara.

Karacabey, S. 1997, “Veritabanı Modelleri Ve Hiyerarşik Veri Tabanından İlişkisel Veri Tabanına Dönüşüm” Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul

Lu, E., J-L., Cheng, Y., Y., 2004 “Design and implementation of a mobile database for java phones” Computer Standarts & Interfaces 26, 401-410

McIntosh A.,C.,S., Cushing J.,B., Nadkarni, N.M., Zeman, L, 2007, “Database desing for ecologist: Composing core entities with observation” Ecological Informatics 2 224-236

Seral, D. 2000, “Web Teknolojileri Kullanılarak Veri Tabanlarına Erişim Metodlarının İncelenmesi ve Uygulaması” Yüksek Lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara

Sürgevil, C.,A. 2000, “İnternet Ortamında İlişkisel Veritabanı Uygulaması” Yüksek Lisans Tezi, Ege Üniversitesi Fen Bilimleri Enstitüsü, İzmir

Şahin, M.,C., 2005 “İnternet Tabanlı Uzaktan Eğitimin Etkililiği: Bir Meta Analiz Çalışması” Cukurova Üniversitesi Sosyal Bilimler Enstitüsü, Yüksek Lisans Tezi, Adana

Şahin, S., 1989, “Veritabanı Kullanan Bir Bilgi Sisteminin Geliştirilmesi” Yüksek Lisans Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir

Uşun S., 2006 “Uzaktan Eğitim” Nobel Yayın Dağıtım, Ankara.

Xue, D, 2003 “Web-based distributed system and database modelling for voncurrent desing ” Computer Aided Design 35, 433-452

Yalçın, Y., 2005 “Fırat Üniversitesi Öğretim Elemanları için Yeni Bir Uzaktan Eğitim Sisteminin Tasarımı ve Modellenmesi”, Fırat Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Tezi, Elazığ

Yapıcıoğlu, H. 2001 “Designing of a Database Management System for Course Design Process in Distance Education” Master Thesis, Middle East Technical University, The Graduate School of Natural and Applied Sciences, Ankara

İnternet

1. http://tr.wikipedia.org/wiki/SCORM_standartları Erişim tarihi: 24.07.2008
2. <http://e-learningtalks.com/index.php/2006/05/07/scormun-e-ogrenme-sektorundeki-rolu-ve-ilkeleri> : Erişim tarihi: 25.07.2008
3. <http://office.microsoft.com/tr-tr/access/HA012242471055.aspx#Normalization>
Erişim tarihi : 07.07.2008
4. <http://bilisimsozluk.com/index.php/2007/08/27/bastan-sona-sql-veritabani-olusturma/>
Erişim tarihi : 07.07.2008

ÖZGEÇMİŞ

Adı Soyadı	Yılmaz SARP KAYA
Doğum Yeri	Pozantı
Doğum Tarihi	07.01.1971
Medeni Hali	Evli
Yabancı Dili	İngilizce
	Eğitim Durumu (Kurum ve Yıl)
Lise	Kadirli Lisesi – Adana
Lisans	Anadolu Üniversitesi İktisadi ve İdari Bilimler Fakültesi
Yüksek Lisans	-
	Çalıştığı Kurum/Kurumlar ve Yıl aralığı
1990-1994	Fırat Üniversitesi Bilgi İşlem Daire Başkanlığı
1994-2008	Afyon Kocatepe Üniversitesi Bilgi İşlem Daire Başkanlığı

EK 1. Örnek Modeli Oluşturan Tablo İsimleri

1. Ajanda
2. AkademikTakvim
3. AkademikTakvimOlay
4. AkademikTakvimOlayTipi
5. AkademikTakvimOlayTipi
6. AnketSonucu
7. AnketSoru
8. AnketSoruSecenek
9. AnketSoruTipi
10. Basvuran
11. BasvuranDiplomaBilgileri
12. BasvuranEgitimBilgileri
13. BasvuranIletisimBilgileri
14. BasvuranIsBilgileri
15. BasvuranSinav
16. BasvuranSinavTipi
17. BasvuranSistemeGirisBilgileri
18. BasvuruDurumu
19. BasvuruDurumuTipi
20. Bolum
21. Cinsiyet
22. Demo
23. DemoKullanici
24. DemoKullaniciDetay
25. DemoKullaniciRolveYetkileri
26. DemoKullaniciSistemeGirisBilgileri
27. DenemeSinavi
28. DenemeSinaviSonucu
29. DenemeSinaviTakip
30. Ders

31. DersBasariOrani
32. DersEvOdevi
33. DersKaynak
34. DersKonu
35. DersSinav
36. DersSinavSonuc
37. DersSinavTipi
38. DersTipi
39. DersZamanlama
40. Dil
41. DilSinav
42. DilSinavTipi
43. DiplomaTipi
44. Dosyam
45. Duyuru
46. DuyuruGorecekKullanicilar
47. DuyuruTipi
48. EvOdevi
49. EvOdeviDegerlendirme
50. Forum
51. ForumKategori
52. ForumKategoriYetkililer
53. ForumKategoriYetkililerDetay
54. ForumKonu
55. ForumYorum
56. Gorev
57. Grup
58. GrupRolu
59. GrupUyesi
60. GrupYetkisi
61. Gun
62. Il

63. Ilce
64. Is
65. IzinTipi
66. KisiselNot
67. KisiselNotGonderilenMesaj
68. KisiselNotKategori
69. KullaniciAkademikBilgiler
70. KullaniciAkademikUnvanBilgileri
71. KullaniciDiplomaBilgileri
72. KullaniciDurumu
73. KullaniciGenelBilgiler
74. KullaniciIdariUnvanBilgileri
75. KullaniciIletisimBilgileri
76. KullaniciIsBilgileri
77. Kullanicilar
78. KullanicilarSinav
79. KullaniciRol
80. KullaniciSifreBilgileri
81. KullaniciYetki
82. LesSinav
83. LesSinavTipi
84. MaliDurum
85. MaliDurumKullaniciOdemeleri
86. MenuAlt
87. MenuAna
88. MenuYetki
89. MesajGelen
90. MesajGiden
91. OdemeTipi
92. OgrenciDers
93. OgrenciEvodeviNotu
94. OgrenciIzlemeRaporu

95. OgrenciProgramUcret
96. Okul
97. OkulTipi
98. OneriveDusunceler
99. ParaTipi
100. PerformansTakipOgrenci
101. PerformansTakipOgretimElemani
102. Program
103. ProgramDersleri
104. ProgramTipi
105. Rol
106. RolYetkisi
107. SanalSinif
108. SanalSinifLog
109. Sektor
110. ServisDersiKategori
111. ServisDersiKullanicilar
112. Sinav
113. SinavKullanicilar
114. SinavKullanicilarCevap
115. SinavKullanicilarCevapDetay
116. SinavKullanicilarSoruSecenekSirasi
117. SinavKullanicilarTakip
118. SinavRasgeleSinavDetay
119. SinavSoru
120. SinavTipi
121. Sinif
122. SinifKullanici
123. Sohpet
124. SohpetKatilanKullaniciLog
125. SohpetMesajLog
126. Soru

127. SoruSecenek
128. SoruTipi
129. SoruZorluk
130. Sozluk
131. SozlukKelime
132. SSS
133. SSSKategori
134. Toplantı
135. ToplantıMesajLog
136. ToplantıyaKatılacakKullanıcılar
137. TrankriptDetay
138. Transkript
139. TranskriptOnaylayan
140. Ülke
141. Üniversite
142. Unvan
143. UnvanTipi
144. Uyruk
145. Workshop
146. WorkshopKaydolanlar
147. WorkshopKaydolanlarSistemeGirisBilgileri
148. WorkshopTarihAraligi
149. YariYil
150. Yetki
151. YetkiDetay
152. YetkiKategori
153. ZorunluDersGrubu