

T.C.  
BEYKENT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI  
BİLGİSAYAR MÜHENDİSLİĞİ BİLİM DALI

**AKILLI e-SORU SINAV SİSTEMİ TASARIMI VE  
UYGULANMASI**

(Yüksek Lisans Tezi)

Tezi Hazırlayan:

**Şerafettin KARATAŞ**

İSTANBUL, 2009

T.C.  
BEYKENT ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI  
BİLGİSAYAR MÜHENDİSLİĞİ BİLİM DALI

**AKILLI e-SORU SINAV SİSTEMİ TASARIMI VE  
UYGULANMASI**

(Yüksek Lisans Tezi)

Tezi Hazırlayan:

**Şerafettin KARATAŞ**

Danışman:

**Dr. Rifat ÇÖLKESEN**

EYLÜL, 2009

İSTANBUL

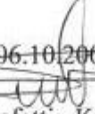
## TEŐEKKÜR

Her zaman bana destek olup alıőmalarım sırasında da hoőgörülu olan aileme, yardımlarını esirgemeyen öđretmen arkadaşlarıma ve tez alıőmam süresince deđerli vaktini ayırarak bana yol gösteren ve yardımcı olan danışman hocam Dr. Rifat ÖLKESEN'e teőekkür ederim.

## YEMİN METNİ

Sunduđum yüksek lisans tezini akademik etik ilkelere bađlı kalarak, hi kimseden akademik ilkelere aykırı bir yardım almaksızın bizzat kendimin hazırladıđıma and ierim.

06.10.2009

  
Şerafettin Karataş

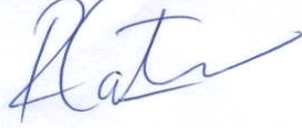
**BEYKENT ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜ**  
**TEZLİ YÜKSEK LİSANS TEZ SINAV TUTANAĞI**

06/10/2009

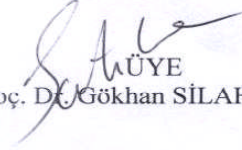
Enstitümüz Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Bilim Dalı yüksek lisans öğrencilerinden 060820001 numaralı **Şerafettin KARATAŞ**' a "*Beykent Üniversitesi Lisansüstü Eğitim - Öğretim Yönetmeliği*"nin ilgili maddesine göre hazırlayarak, Enstitümüze teslim ettiği "*Akıllı e-Soru Sınav Sistemi Tasarımı ve Uygulaması*" tezini, Yönetim Kurulumuzun 16.09.2009 tarih ve 2009/11 sayılı toplantısında seçilen ve Fakülte binasında toplanan biz jüri üyeleri huzurunda, ilgili yönetmeliğin (c) bendi gereğince aday tarafından savunulmuş ve sonuçta adayın tezi hakkında *oybirliği* ile **Kabul** kararı verilmiştir.

İşbu tutanak Enstitü Müdürlüğü'ne sunulmak üzere tarafımızdan düzenlenmiştir.


DANIŞMAN  
Yrd. Doç. Dr. Rifat ÇÖLKESEN



ÜYE  
Yrd. Doç. Dr. Gökhan SİLAHTAROĞLU



ÜYE  
Yrd. Doç. Dr. Haluk KUL



## ÖZET

### AKILLI e-SORU SINAV SİSTEMİ TASARIMI VE UYGULAMASI

Şerafettin KARATAŞ

2009

Son zamanlarda eğitim alanında yaşanan gelişmelerin motoru uzaktan eğitim teknolojileriyle üretilen uygulamalar olmaktadır. Bu uygulamalar vasıtasıyla eğitim faaliyetleri her zaman istenildiği gibi yaygınlık ve hız kazanmıştır. Bu gelişmelere eğitimde ölçme ve değerlendirme yöntemlerini uygulayanlarda ayak uydurmalıdır.

Bu tez çalışmasında öğretmene kapsamlı ve hızlı soru sorma imkânı sağlayan yardımcı kaynak olarak kullanabileceği bir araç sunulmuştur. Bununla birlikte öğrenciye kendisini ders konusunun tümünden değerlendirip, hızla sonuç alabileceği ve eksiklerini fark edebileceği bir yazılım aracı oluşturulmuştur.

Yapılan çalışmada ders konusunu içeren metinlerin çözümlenmesiyle dinamik olarak üretilen sorular öğretmen denetiminden geçtikten sonra sınav sorularına dönüştürülür. Sunulan bu yaklaşımın eğitime ve akademik hayata katkısı ile üretilen çok sayıda zengin içerikli sorularla kolay hazırlanabilen, adil, hızlı sonuç üreten değerlendirmeler yapılacaktır.

**Anahtar Kelimeler:** Dinamik Soru Üretilmesi, Uzaktan Eğitim, Akıllı Soru Sorma sistemi, Doğal Dil İşleme, Cümlelerin Kelimelerini Çözümleme

**Tez Danışmanı:** Dr. Rifat ÇÖLKESEN

## ABSTRACT

# INTELLIGENT e-QUESTION EXAM SYSTEM DESIGN AND APPLICATION

Şerafettin KARATAŞ

2009

Nowadays, the applications by remote education technology are the engine of the developments in education. Thanks to these applications, education activities have been more widespread and speedy. People, applying evaluation and measurement methods on education, should also keep in step with these developments.

By this thesis, an appliance, that makes an extensive and quick way of preparing questions, is presented for the teachers. This appliance can also be used as a helper source. In addition to this, students can use this. Students evaluate the understandings of the whole topic and notice their missing quickly by using this appliance.

This working first makes analyzing of the lesson text. After this, questions related to text are prepared dynamically. Finally, teacher checks the questions and make them exam questions. When this presented approach is used in education and academic life, easy-prepared, fair and quick measurements and evaluations are made by so many wealthy content questions.

**Key Words:** Making Dynamic Question, Remote Education, Intelligent Asking Question System, Natural Language Processing, Analyzing the Words of the Sentence.

**Advisor:** Dr. Rifat ÇÖLKESEN

## İÇİNDEKİLER

TEŞEKKÜR.....	I
YEMİN METNİ.....	II
TEZ SINAV TUTANAĞI.....	III
ÖZET.....	IV
ABSTRACT.....	V
İÇİNDEKİLER.....	VI
KISALTMALAR LISTESİ.....	VIII
ŞEKİL LISTESİ.....	IX
TABLO LISTESİ.....	X
1. GİRİŞ.....	1
2. AKILLI e-SORU SİSTEMİNİN ANALİZİ ve TASARIMI.....	2
2.1. Tasarlanan Yazılımın Metodolojisi ve Standartları.....	2
2.2. Bilgisayar Sistemi Mühendisliği.....	4
2.2.1. Proje Yaşam Çevrimi.....	4
2.2.2. Edinme Süreci.....	4
2.2.3. Sağlama Süreci.....	8
2.2.4. Geliştirme Süreci.....	11
2.2.5. İşletme ve Bakım Süreci.....	19
2.3. Risk Yönetimi Risklerin Tanımlanması ve Risklerin Çözümlemesi .....	20
2.3.1. Kullanıcı Hatalarından Kaynaklanan Riskler.....	20
2.3.2. Doğal Dilin Yapısından Kaynaklanan Riskler.....	20
3. AKILLI e-SORU SINAV SİSTEMİNİN TEMELLERİ ve KULLANILAN METİN ÇÖZÜMLEME YAPISI.....	22
3.1. Sistemin Genel Mimarisi.....	22
3.2. Sistemin Girdileri Çıktıları ve Gerçekleşen Süreçler.....	23
3.3. Metin Çözümleme Yapısı.....	25
3.3.1. Türkçe Kelimelerin Ek Yapısı.....	25



3.3.2. Doğal Dil Çözümleyicilerin Genel Yapısı.....	26
3.3.3. Biçimbilim Analiz.....	28
3.4. Cümle Öğelerini Çözümleme Algoritması.....	28
3.4.1. Kaba Kod Biçiminde Çözümleme Algoritması.....	30
3.4.2. Kelime Veri Modelleri.....	31
<b>4. SİSTEMİN YAZILIM TEMELLERİ</b>	
4.1. Yazılımda Kullanılan Temel Elemanlar .....	35
4.2. Kullanılan Veri Tipleri ve Veri Yapıları.....	36
4.3. Soru Üretme Algoritması.....	37
4.4. Kullanılan Sınıflar.....	38
4.4.1. Sınıf Diyagramı .....	39
<b>5. AKILLI e-SORU SINAV SİSTEMİNİN UYGULAMASI.....</b>	<b>44</b>
5.1. Uygulamaya Erişim ve Yetkilerin Yapılandırılması.....	44
5.2. Soru Üretilmesi ve Sınav Hazırlanması.....	45
5.3. Sınavın Aktifleştirilmesi Sonuçların İzlenmesi.....	48
5.4. Üretilen Örnek Soruların İncelenmesi ve Değerlendirilmesi.....	50
<b>6. SONUÇ.....</b>	<b>53</b>
Kaynakça.....	55

## KISALTMALAR LİSTESİ

ÇLE	Çoğul Eki
DDİ	Doğal Dil İşleme
EKE	Ek Fiil Eki
FÇE	Fiil Çekim Eki
FİYE	Fiilden İsim Yapım Eki
FYE	Fiil Yapım Eki
HLE	Hal Eki
İÇE	İsim Çekim Eki
İFYE	İsimden Fiil Yapım Eki
İYE	İyelik Eki
İYE	İsim Yapım Eki
KOK	Kelime Kökünün Türü
NLP	Natural Language Processing (Doğal Dil İşleme)
OZE	Olumsuzluk Eki
SRE	Soru Eki
ŞHE	Şahıs Eki
TDK	Türk Dil Kurumu
ZŞE	Zaman Şekil Eki

## ŞEKİL LİSTESİ

Şekil 1 - Spiral Model .....	3
Şekil 2 - Klasik Çevrim .....	4
Şekil 3 - Bilgisayar Destekli Türkçe Öğretimi Modelinin Mimarisi.....	5
Şekil 4 - Metinlerin Matematiksel Modellenmesi.....	6
Şekil 5 - Klasik Proje Yaşam Çevrimi.....	9
Şekil 6 – Zaman Yapılabilirliği.....	12
Şekil 7 – Kullanım Senaryosu.....	14
Şekil 8 – Varlık İlişki Diyagramı .....	15
Şekil 9 – Sınav Veritabanı Diyagramı.....	16
Şekil 10 – Veri Akışı Modeli .....	17
Şekil 11 – Merkezi Tümlleştirme Mimarisine Göre sistemin Genel Yapısı.....	18
Şekil 12 – Uçtan Uca Akıllı e-Soru Sınav Siteminin Genel Mimarisi .....	22
Şekil 13 – Sistem Unsurlarının Giriş Çıkış ve İşleme Süreci.....	24
Şekil 14 – Türkçe Kelime ve Ekler Veri Modeli.....	25
Şekil 15 – Doğal Dil Çözümleyici Yapısı.....	27
Şekil 16 – Cümle Öğelerini Çözümleme Algoritması .....	29
Şekil 17 – Kaba Kod Çözümleme Algoritması.....	30
Şekil 18 – İsim Modeli UML Etkinlik Diyagramı.....	33
Şekil 19 – Fiil Modeli UML Etkinlik Diyagramı.....	34
Şekil 20 – Uygulamanın Dilbilgisi İsim Alanı Sınıf Yapıları.....	36
Şekil 21 – Soru Üretme Algoritması.....	37
Şekil 22 – Uygulamanın Sınıf Diyagramı.....	43
Şekil 23– Roller.....	44
Şekil 24– Güvenlik ve Üyeler Veritabanı.....	44
Şekil 25 – Kelime Çözümleme Ekranı.....	45
Şekil 26 – Soru Hazırlama Ekranı.....	46
Şekil 27 – Sınav Soruları Tablo Yapısı. ....	46
Şekil 28 – Terim Sorusu Üretim Ekranı.....	47
Şekil 29 – Terim Sorusu Üretim Sonuçları Ekranı.....	47
Şekil 30– Sınav Hazırlama Ekranı.....	48
Şekil 31– Sınav Ekranı.....	49
Şekil 32 – Sınav Sonuçları Ekranı.....	49
Şekil 33 – Klasik Sınav Değerlendirme Ekranı.....	50
Şekil 34 – Örnek Soru Üretimi Ekranı.....	51
Şekil 35 – Örnek Soru Tipleri Ekranı.....	52

## TABLO LİSTESİ

Tablo 1 - En Az ve Tavsiye Edilen Donanım Gereksinimi.....	8
Tablo 2 – İsim-Fiil Türü Kelimeler Veri Modeli Örneği.....	31
Tablo 3 – İsim Türü Kelimeler Veri Modeli Örneği.....	31
Tablo 4 - Cümlelerin Kelimeleri Birleşik Veri Modeli Örneği.....	31
Tablo 5 – Fiil Türü Kelimeler (basit veya bileşik çekimli fiiller) Veri Modeli Örneği.....	32
Tablo 6 – Kelime Modelleri ve Alfabe.....	39
Tablo 7 – Dilbilgisi Kuralları.....	40
Tablo 8 – Cümlelerin Öğeleri ve Gövde Bilgisi.....	41
Tablo 9 – Sınav Bölümü Sınıfları .....	42

## 1. GİRİŞ

Hızla gelişen bilişim teknolojileri sayesinde insanlara her alanda kolaylık sağlayan sistemler sunulmaktadır. Teknoloji ve bilimin insan hayatına etkilerinin en önemli sonuçlarından biri eski alışkanlıkların değişmesidir. Dönüşüm süreçlerini bu sistemlerin tetiklediği somut olarak görülmektedir. Bu çerçevede geliştirilen ve insanların hizmetine sunulan birçok teknolojik ürün insanların hem günlük hayatında hem de iş hayatında bilgiyi kullanmasını etkin hâle getirmektedir. Bununla birlikte amaçlanan işlevlerin başarıyla gerçekleştirilmesi sebebiyle özellikle akıllı elektronik sistemler çok yaygın olarak kullanılmaktadır. Klasik yöntemlerle gerçekleştirilen işler bilgisayar destekli sistemlerle yapılmakta ve karmaşıklık aşğılara indirgenerek, iş süreci sonucunda alınması beklenen çıktılarının niceliğini ve niteliğini yükseltmektedir.

Teknoloji birçok alanda olduğu gibi eğitim sürecinde de çoğu iş yapma modelini değiştirmektedir. Uygulanan eğitimin sonuçlarının tespit edilebilmesi ölçme değerlendirme metodlarıyla olacaktır. Bundan dolayı öğrenciye çeşitli sınav uygulamaları yapılır. Öğretmenin asıl yükünün ders ve konunun kavratılması yönlerine doğru çekilmesi, öğretmene hız ve zaman kazandırması, öğrencinin mekâna zamana bağlı kalmaksızın, öğretmene ihtiyaç olmadan kendini ölçebilmesi ve eksik olduğu konuları ortaya çıkararak ilgili noktalara odaklanabilmesi ihtiyaçları mevcuttur. Bunun gibi nedenlerle otomatik olarak, belirlenen metinlerden dinamik soru üreten sınav yazılımı bu alandaki gereksinimleri karşılayacak bir araç olmaktadır.

Tasarlanan mekanizmanın uygulanmasıyla ders konusundan çok fazla sayıda birbirine benzeyen sorular üretilen, aynı sınıfta sınava giren öğrencilerin birbirinden kopya çekmesi de önlenecektir. Ayrıca uygulamanın Web tabanlı olması uzaktan eğitim sistemlerinde tercih edilmesini sağlayacaktır. Uzaktan eğitim alan, çeşitli sebeplerle zaman kısıtlaması olan öğrencilerin, istedikleri zaman sınava girerek kendilerini değerlendirmelerine olanak tanınacaktır. Derslere ve konulara göre soru dağılımı yapıldığından, öğrencinin eksik olduğu taraflar ortaya çıkacaktır. İnternet altyapısından kaynaklanabilecek kesinti, darboğaz, yoğun trafik gibi aksaklıklara karşı kullanıcının sınav akışını engellemeyecek özellikte geliştirilen uygulamada ile esnek çalışma koşulları mevcuttur.

Türkçe dilbilgisi geliştirilen yazılım içinde dilbilgisi kurallarını içeren bir sınıf oluşturularak modellenmiştir. Bu çalışmadaki isterlere uygun olarak tez çalışmasında tasarım ve uygulama işlemlerinde Microsoft C# programlama dili, Microsoft. NET ortamı, Web uygulama tarafı için ASP.NET Web geliştirme platformu, veritabanı işlemlerinde bu ortama en uyumlu olan MS SQL tercih edilmiştir.

Bu tez çalışması altı bölümden oluşmuştur. İkinci bölümde sistem analizi ve tasarımı yapılmış, üçüncü bölümde akıllı e-soru sınav sisteminin temelleri ve kullanılan metin çözümleme yapısı anlatılmıştır. Dördüncü bölümde sistemin yazılımsal temelleri açıklanmıştır. Beşinci bölümde tanıtılan akıllı e-soru sınav sisteminin uygulamasının ardından altıncı bölümde değerlendirmeler yapılarak tez çalışması sonuçlandırılmıştır.

## **2. SİSTEMİNİN ANALİZİ ve TASARIMI**

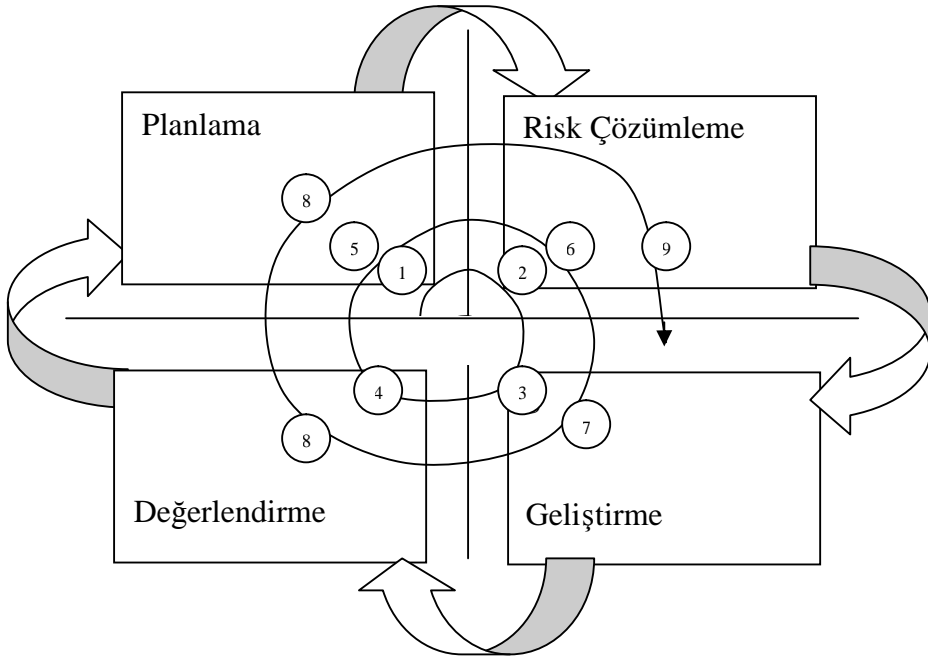
Bu bölümde akıllı soru üretimi sisteminin yapısı, tez çalışması sırasında oluşturulacak sistemin analizi ve Türkçe metin tabanlı dışarıdan girilen kaynak ders konuları bilgilerinden faydalanarak gerçekleştirilen otomatik soru üretiminin tasarımının açıklanması yer almaktadır.

### **2.1. Tasarlanan Yazılımın Metodolojisi ve Standartları**

Yapılan tezde yazılım geliştirme standartlara uygunluk, karmaşıklıkların en aza indirilmesi, sistem yapısı, kullanılan nesnelere, nesnelere arası veri akışı, veri modeli ve yapılarının açıklanması, kullanılan veritabanının ve ara yüzlerin projedeki yeri ve diğer tüm geliştirme aşamalarında uluslararası IEEE/EIA 12207 yazılım geliştirme süreçlerine başvurularak standartlar uygulanmıştır. Ayrıca sistem yaşam döngü süreçleri ile ISO/IEC 15288 standartları, bakım sürecinde IEEE 1219-1998 standartları temel alınmıştır.

Yazılım geliştirme sürecini açıklayan modeller olarak şelale modeli ve spiral model seçilmiştir. Şelale modeli, büyük tasarım modeli adıyla veya geleneksel model adıyla da adlandırılır. Şelale modeliyle tüm sistemin, sistemdeki gereksinimler, kullanıcıların istekleri, çözümlenen yapıların isteklere uygun tasarlanması gerçekleştirilip test edilmesi ve bakımı mümkün olmaktadır. Bu modele göre birbirini takip eden sistem mühendisliği, çözümleme, tasarım, gerçekleştirim, test, bakım adımlarıyla sistematik olarak yazılım geliştirilmiştir. Ayrıca yazılım geliştirilirken faydalanılan Spiral model karma bir yöntem olarak ele alınmıştır.

Klasik yöntem ve prototipleme yöntemlerinin bir araya gelmesiyle oluşturulan bu modelde ürünün ilk tesliminden sonraki zamanlarda geliştirmeler yapılarak yeni sürümler teslim edilir. Çıktılar yeni sürümlerin girdileri olurlar. Evrimsel bir yaklaşımla müşteri ve geliştiricinin her evrede riskleri birlikte anlaması ve çözümler bulmasını, önlemler almasını sağlar. [1]



Şekil 1 – Spiral Model

Spiral model temel alınarak yukarıdaki şekilde görülen sırayla numaralandırılmış ilk dört aşama; gereksinimlerin elde edilmesi ve gereksinimler dikkate alınarak projenin planlanması, gereksinimlere göre risk analizi yapılması, risk analizine göre çeşitli yöntemlerle geliştirme, müşterinin ürünü incelemesi ve değerlendirmesi diğer aşamalarda sırayla tekrar eder.[2]

Proje yazılım yöntembilimi modelleri olarak seçilen Spiral model ve klasik çevrim modeline göre gerçekleştirilmiştir.

## 2.2. Bilgisayar Sistemi Mühendisliği

Tezin bu aşamasında yazılım, donanım, İnternet, gibi fiziksel unsurlar ile birlikte kullanıcılara yazılım hizmet sunulurken kullanılan öğeler standartlara göre açıklanmıştır.

### 2.2.1. Proje Yaşam Çevrimi

En çok kullanılan ve en son belirlen yazılım geliştirme standardı IEEE/EIA 12207'ye göre tez çalışmasının proje yaşam çevrimi etkinlikleri edinme, sağlama, geliştirme, işletme ve bakım süreçleri aşağıda verilen şemaya uygun olarak anlatılmıştır.



Şekil 2 – Proje Yaşam Çevrimi

### 2.2.2. Edinme Süreci

Akıllı e-Soru Sınav Sisteminin kavramsal tanımlamaları, projenin amacı, işletim sistemi, donanım, yazılım gibi temel gereksinimlerinin açıklanmıştır.

- **Referans Taranması İncelenmesi ve Değerlendirilmesi**

Çalışmanın bu aşamasında referans olabilecek kaynaklara yönelik literatür taraması yapılmış tez konusu ile ilgili çalışma yapıp yapılmadığı araştırılmıştır. Tarama sonucunda projeye tam olarak örnek olabilecek bir çalışmaya rastlanmamıştır. Bununla birlikte doğal dil işleme ve Türkçe dilbilgisi ile ilgili tez ve makalelerden yararlanılmıştır.

İncelenen makalelerden Kemal Oflazer, H. Cem Bozşahin, (1994) ,Turkish Natural Language Processing Initiative [3] çalışmalarında bilgisayar destekli Türkçe dil işleme uygulamalarına



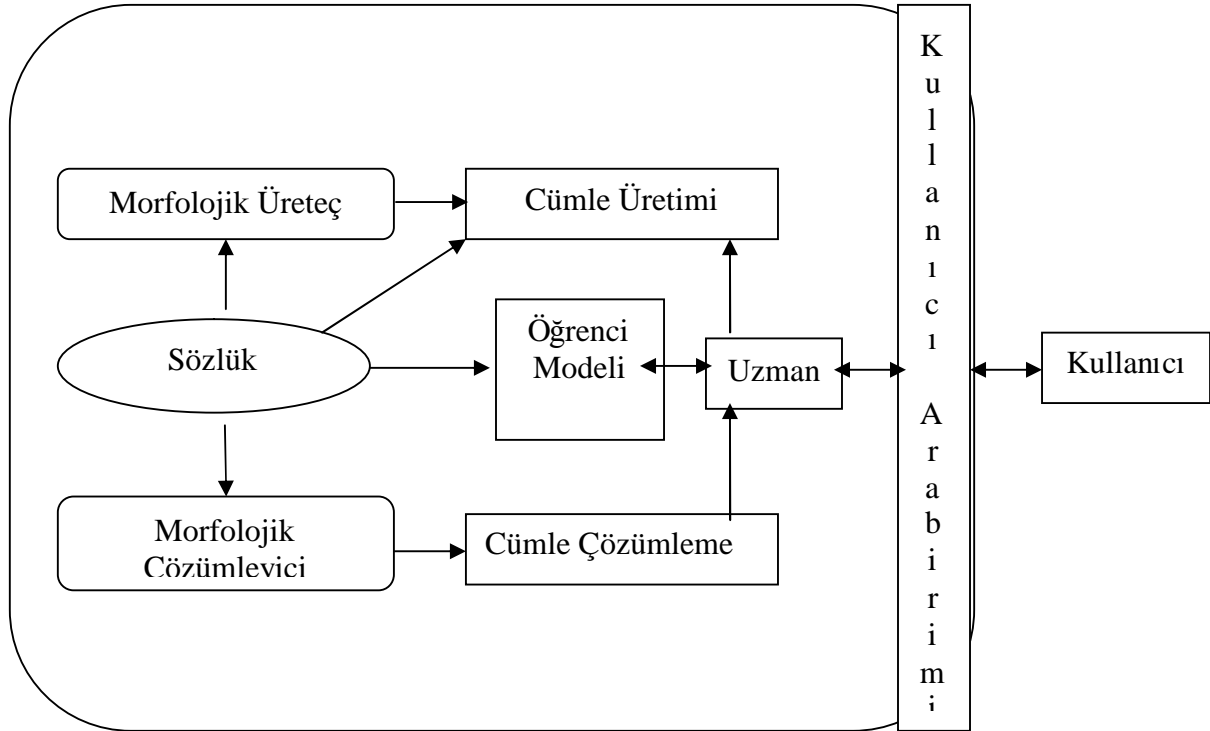
yararlı olabilecek bir yapı sunmuşlardır.

Türkçe'nin sondan eklemeli dil yapısındadır ve kelimenin anlamlı en temel parçası olan kökünün sonuna eklerin eklenmesiyle yeni kelimelerin üretilmesi yapılmaktadır. Bu durum değerlendirilerek Türkçe dilinin İngilizce gibi farklı yapıdaki dillere göre biçimsel olarak analizini kolaylaştırmakta olduğu esasıyla bilgisayar destekli dil çalışmalarının sürdürülebilmesi amaçlanmıştır.

Türkçe biçimbilimsel (morfolojik) çözümlerinde kullanılabilecek bir yapı ortaya konulmuştur.[3] Türkçe doğal dil işlemede morfolojik analiz üç bölüme ayrılır.

1. Kök tespiti
2. Morfolojik sınıama
3. Morfolojik inceleme [3]

Morfolojik analizi kullanan bir uygulama olan, bilgisayar destekli Türkçe öğretimde kullanılabilen, sözlük kullanarak cümle üreten ve çözümleneyen yapının şeması aşağıdadır.

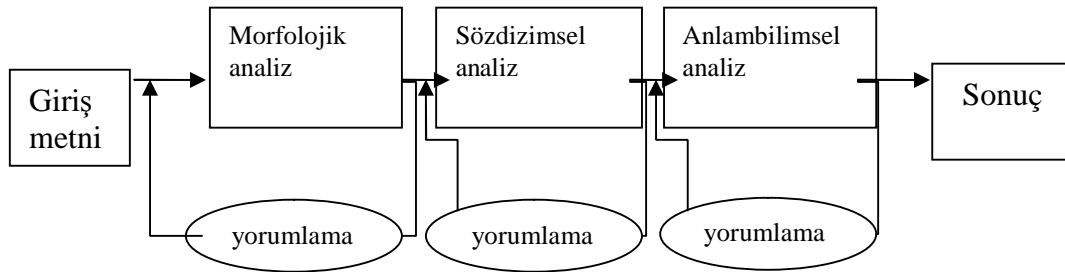


Şekil 3 - Bilgisayar Destekli Türkçe Öğretimi Modelinin Mimarisi [9]

Marvin Minsky, (1982), Why People Think Computers Can't [4] makalesinde uzman sistemler

aracılığıyla daha kesin sonuçlar üretebilmenin cümlelerin anlamını en ince detayına kadar çıkartabilmekten geçtiğini bildirmektedir. Kavramların anlamlarını detaylandırarak modeller çıkartmanın gerekliliğini vurgulamaktadır.

Ünal Çakıroğlu, Türkçe Aritmetik Problemlerin Bilgisayarla Çözümü [5] Türkçe problemlerin metinlerini çözümlerken, çok geniş sözcük dağarcığının gerekliliğini belirtmektedir. Morfolojik ve sözdizimsel analizle birlikte bilgi tabanı yardımıyla elde edilen ifadelerin formülleştirilmesi ve matematiksel modelin ortaya konulmasıyla çözümlenmeye yaklaşmaktadır. Aşağıdaki şekilde bu model gösterilmiştir.



Şekil 4 - Metinlerin Matematiksel Modellenmesi [8]

Zeynep Altan, Dil Modellemede Belirsizlik Probleminin Etmenlenmiş Dilbilgisi İle Giderilmesi çalışmasında, tümcelerin sözdizimsel çözümlemesi diğer dillerin çoğunda olduğu gibi Chomsky sıra düzenindeki bağlamdan bağımsız dilbilgisine (context-free-grammar, CFG) göre gerçekleştirmektedir. Metinleri çözümlemede morfolojik seviyede oluşan karmaşıklığı yeni bir dilbilgisi olarak etmenlenmiş dilbilgisi tanımlayarak ve bu dilbilgisine göre geliştirilen çözümleme algoritmasıyla çözmeyi önermektedir. [6]

Muhammed Oguzhan Külekci, An Intelligent Diagnostic System From The Clinical Narratives In Turkish [7] morfolojik analizi ekleri grup bazında ele alarak yapar, dışarıdan verilen metin içerisindeki kelimelerin köklerini tespit ederek bilgi bankasındaki anahtar sözcüklerle karşılaştırarak sonuca ulaşır. Bu işlem sırasında soldan sağa doğru tarama yapan bir algoritmayla kökleri bulup, eklerin grup seviyesinde analiz eder. [7]

- **Akıllı e-Soru Sistemi Tasarımının Faydaları ve Amacı**

Eđitim ölçme-deęerlendirme yöntemlerinin konun tümünü deęerlendiren, konu dıřına çıkmadan objektif soru nitelięi taşıyan metinler içermesi bilgisi ölçülen öęrenci kesin doęru olarak deęerlendirilmesi ihtiyaçlarına yönelik geliştirilmiştir. Ders içerięi ile ilgili metinlerinin kullanılmasıyla o metinlerle ilgili sorular oluşturulur. Üretilen sorular konu metinlerini içerdiiğinden konu ile ilgili öęrenci bilgisi ölçülür. Böylece öęrencinin kendi eksiklerini fark etmesini sağlar. Sistemi kullanarak soru hazırlayan öęretmenin vaktini verimli kullanmasını ve Web tabanlı sınav uygulaması sayesinde hızlı bir şekilde sınav deęerlendirmesi olanaęı amaçlanmıştır.

- **Sistemin İşletim Sistemi Gereksinimi**

Geliştirilen mekanizma sunucu kullanıcı mimarisini kullanacaktır. Web hizmeti olarak çalışacak sistemin sunucu tarafı veritabanı sunucusu barındırarak Web sunucu hizmeti sunacaktır. Bu hizmetler her ne kadar Windows XP veya Windows Vista işletim sistemleri üzerinde çalışıyor olsa da başarımın yüksek olması için Windows Server ailesinden Windows Server 2003 veya Windows Server 2008 tercih edilmelidir.[8][9]

- **Sistemin Yazılım Gereksinimi**

Web tabanlı çalışmayı ve uzaktan eğitim uygulamalarını destekleyen sistem çok sıralı istemci sunucu yapısıyla bir taraftan kullanıcı etkileşimli olarak veritabanı hizmetleri verecek dięer taraftan Web kullanıcılarına hizmet sunacaktır. Sistemi kullananların sayısı arttıkça çok sıralı kullanıcı sunucu mimarilerinde yük artışından dolayı başarım düşmeleri ve tıkanıklar görülebilir.[10]

İçerdiği programlama unsurları, günümüzdeki yaygın kullanımı ve hedeflenen isteklere çözüm olacağı düşünülerek programlama dili olarak C# programlama dili kullanılmıştır. C# için doğal geliştirme ortamı olan Microsoft .Net platformu aracı olan Visual Studio 2008 , veritabanı uygulamaları geliştirilirken bu ortam için en yüksek verimi ve uygulama başarımını sağlayacak

Microsoft SQL server 2005 yönetim aracı ile birlikte Microsoft SQL veritabanı dili kullanılmıştır.

- **Sistemin Donanım Gereksinimi**

Çalışan sistemin sunucuları aşağıda belirtilen gereksinimler ile çalışır. Bu özellikler 32 bitlik işlemci yeterli olduğu için 32 bit işlemci esas alınarak aşağıdaki tabloda belirtilmiştir. Bu tablodaki değerler işletim sistemi ile birlikte çalışan yazılımların toplam gereksinimleridir.[11][12][13]

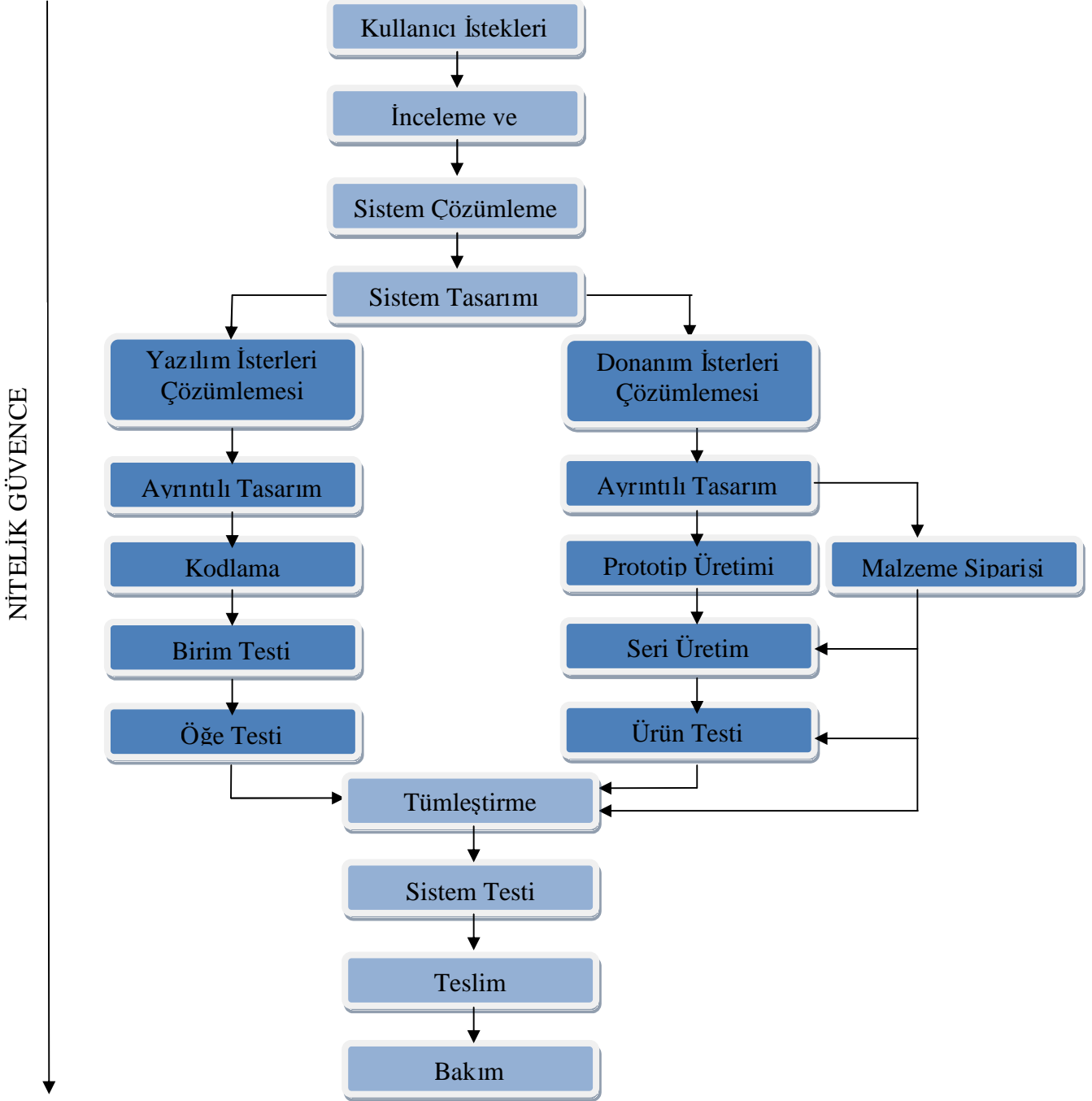
**Tablo 1 - En Az ve Tavsiye Edilen Donanım Gereksinimi**

	En az	Tavsiye edilen
Bellek	1 GB	2 GB
İşlemci	2.0 GHZ	2.6 GHZ
Disk Alanı	8 GB	20 GB

### **2.2.3. Sağlama Süreci**

Bu aşamada iş tanımı yapılarak yaşam çevrim modelinin seçimi ve planlamalar yapılmıştır.

Projede sistem yaşam döngü süreçleri ile ilgili ISO/IEC 15288 standartlarına uygun tanımlanan yaşam çevrim modeli olarak klasik proje yaşam çevrimi seçilmiştir. Projenin esas alınarak kullanılan Klasik Proje Yaşam Çevrimi modeli bir sonraki sayfada yer almaktadır.



Şekil 5 - Klasik Proje Yaşam Çevrimi [1]

- **Kullanıcı İstekleri**

Proje amacına bakıldığında daha çok bir araştırma geliştirme projesi olduğu için öncelikli olarak ilerideki çalışmalara başvuru kaynağı olabilecek temel gereksinimlerin elde edilmesi hedeflenmiştir. Uygulamanın çalışma süreci metin tabanlı akıllı soru sorma sisteminin mimarisini oluşturan “Uzaktan Eğitimde Akıllı Soru Sorma Mekanizması”[14] yaklaşımının yapılabirliği kanıtlanmak için yürütülmüştür. Çalışmanın ilk ayağında beklenen soru üreten bir mekanizmanın oluşturulmasıdır. Soru üretimi için kullanılması planlanan girdiler, ilgili derse ait konunun anlatıldığı metin tabanlı bilgisayar ortamına aktarılmış kaynaklardır. Ortaya çıkartılacak soruların sorulabilecek nitelikte olup olmadığı öğretmen tarafından belirlenecektir. Anlamsız, basit veya gereksiz görülen soruların öğretmen tarafından elemeye tabi tutulacağı bir bölüm geliştirilecektir. Tasarlanan sistem sonrasında uygulamaya dönüştürülerek kullanıcılara sunulacaktır. Sistem tümüyle Web tabanlı çalışacaktır. Hem soruların üretilmesi hem de sınav uygulaması ile ilgili işlemler Web üzerinden yapılacaktır. Sınav hazırlama işlemleri öğretmen tarafından yapılarak ders, konu soru tipi seçilecek, istenen öğrenciler için sınav aktif hâle getirilecektir. Sınavı aktif olan öğrenci verilen süre içinde sınavını tamamlamalıdır. Sınav bitiminde eğer klasik ise öğretmen, test ise sistem otomatik olarak sınav değerlendirmesi yapacaktır. Sınav sonuçları öğrenci veya öğretmen tarafından sınavdan sonraki zamanlarda geçmişe doğru izlenebilir yapıda olacaktır. Öğrencinin sınav geçmişi izlenerek zayıf olduğu konular tespit edilebilecektir.

- **İnceleme ve Planlama**

Yazılımı kullanacak kullanıcı profili öğrenci ve öğretmen olmak üzere iki temel kullanıcı tipinden oluşur. Öğretmen kullanıcısı Web ara yüzünü kullanarak sistemden sorular ürettirip sınav hazırlayacaktır. Öğrenci kullanıcısı ise sistemde güvenli yöntemlerle Web üzerinden dâhil olup sınav alacaktır. Ayrıca kullanıcı işlemlerini yöneten yönetici kullanıcısı mevcuttur.

Sistem sayesinde öğretmen kullanıcısı daha kapsamlı, hızlı ve etkin sınav hazırlama aracına sahip olacaktır. Öğrenci kullanıcısının ise hangi konudan bilgi eksiği olduğu tespit edebileceği hızlı sonuç veren yapan bir değerlendirme olanağı olacaktır.

#### **2.2.4. Geliştirme Süreci**

- **Sistemin Çözümlemesi**

Sistemin analizi veya çözümlemesi aşamasında kullanıcı ortamı, geliştirme ve davranış modelleri belirlenmiştir.

- **Yapılabilirlik Raporları**

- **❖ Zaman Fizibilitesi**

Çalışmanın zaman yapılabilirliği değerlendirilerek zamanlanması yapılmıştır. Projenin zaman fizibilitesi zaman yapılabilirliği diyagramı olarak hazırlanmıştır. Yapılan zamanlamaya göre gerçekleştirilen projenin adımları ve bu adımlara ayrılan süreler ile ilgili bilgiler bir sonraki sayfada sunulmuştur.





### ❖ Teknik Fizibilitesi

Sistemin teknik olarak yapılabilirliği yazılım donanım ve işletim sistemi gereksinimleri kısmında araştırılarak belirtilmiştir. Bunlara ek olarak yapılan incelemelerde kullanılan yazılım ortamının planlanan hedefleri desteklediği görülmüştür. Microsoft .Net platformu ve yazılım geliştirme aracı olan Visual Studio 2008, işletim sisteminden ve donanımdan daha üst seviyede taşınabilir olarak tasarlanmıştır.

İncelenen .NET Framework yapısı:

- ✓ İnternet’i hedef alan bileşen setiyle
- ✓ Programcılarını esnek çalışma imkânını sunan, diller arası etkileşime olanak tanıyan bağımsız dil mimarisıyla
- ✓ Donanım-bağımsız ara seviye derleme ve güvenli talimat icraatını güvence altına alan çalışma zamanı ortamıyla
- ✓ Web uygulamalarına getirdiği özgün yaklaşımıyla
- ✓ Zengin içerikle organize edilmiş sınıf kütüphanesi ve dokümantasyonu ile
- ✓ Sınırları ortadan kaldıran Web servisi desteğiyle [15]

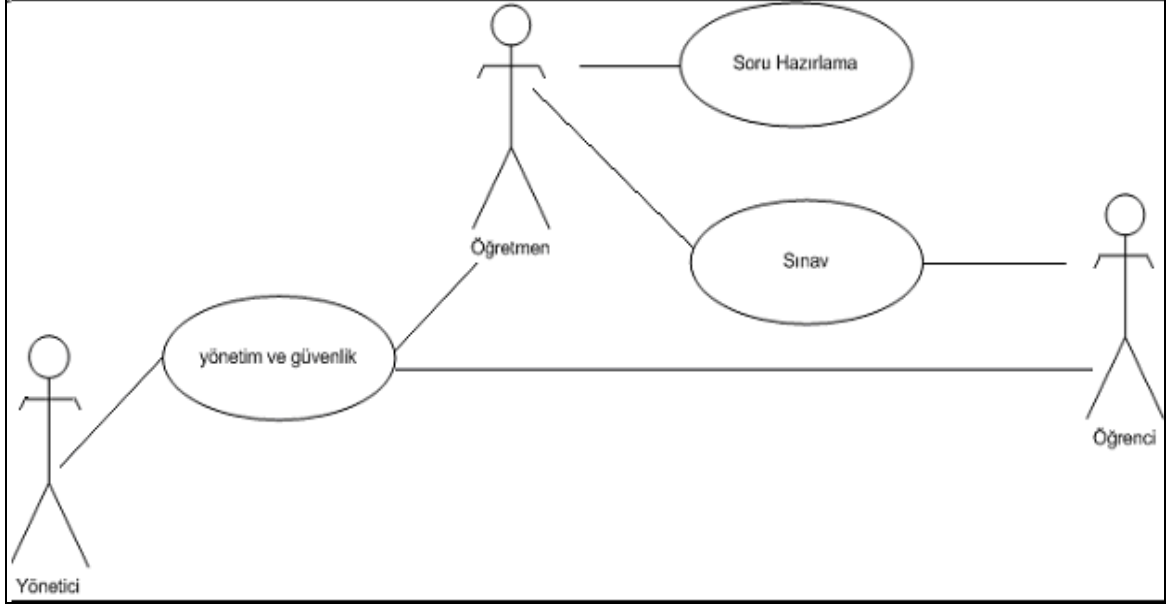
Proje hedefleri olan Web üzerinden Türkçe metin işleme ve sonuçlarla Web tabanlı sınavlar uygulayabilme özellikleri değerlendirildiğinde projede kullanılacak teknik özellikleri olan bir yazılım ortamıdır.

### ❖ Yasal Fizibilitesi

Uygulamada Web üzerinden ders konularını oluşturan Türkçe metin girdileri kullanılacağından bu bilgilerin alınacağı kaynakların yasal izinlerinin alınmış olmasına dikkat edilmelidir. Bununla birlikte sistemde kullanılan yazılımlar olan Visual Studio 2008, Microsoft SQL Server 2005 lisansları ve sunucu işletim sistemlerinin kullanıcı sayısı göz önünde bulundurularak yapılandırılan lisanslama işlemleri yasal olarak yapılmış olmalıdır.

- **Kullanım Senaryosu**

Kullanım senaryosu diyagramı kullanılarak aşağıdaki modelde kullanıcıların sistemi kullanma senaryoları çizilmiştir.

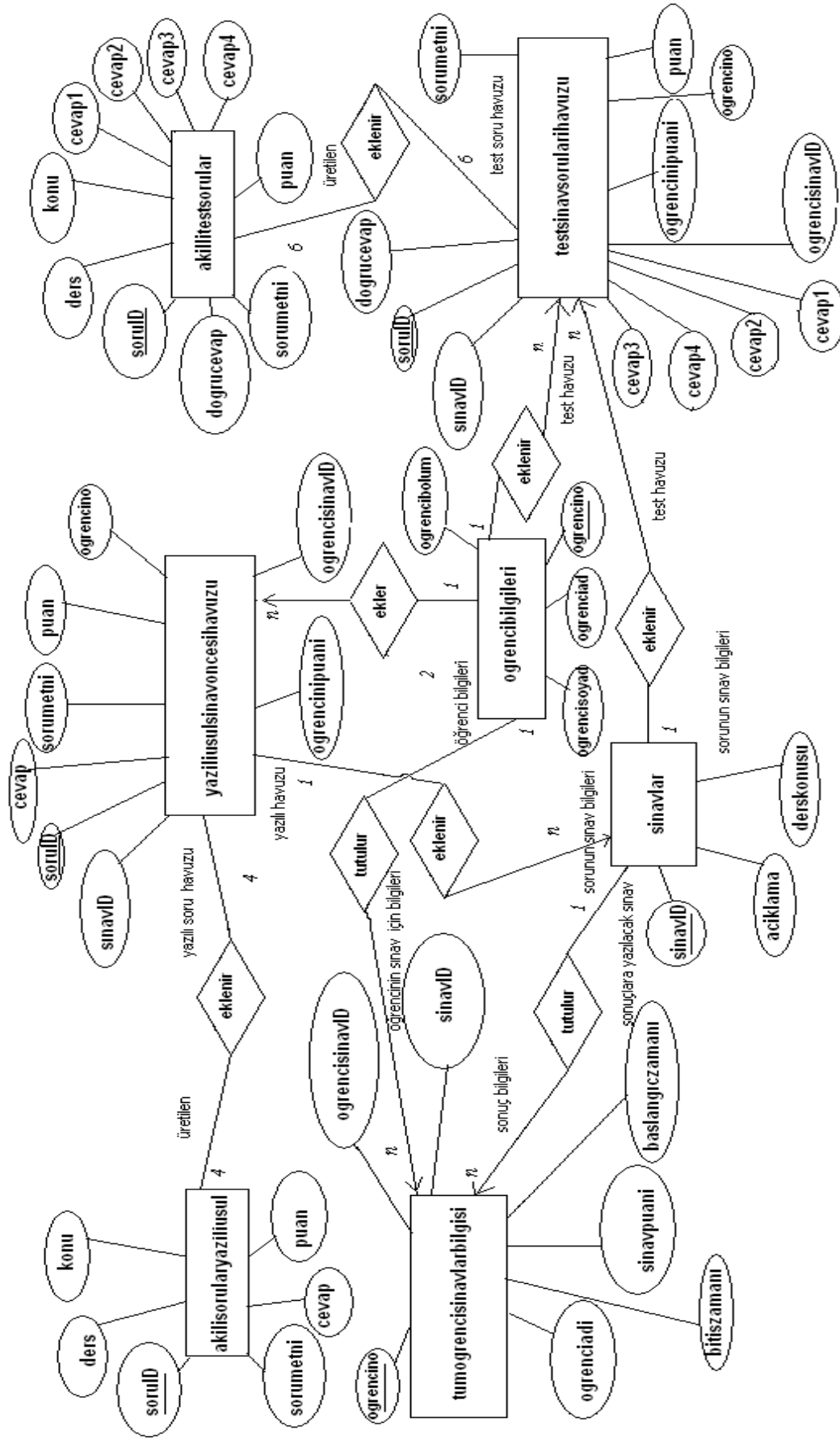


Şekil 7 – Kullanım Senaryosu

Bu senaryo gereğince öğretmen kullanıcısı dinamik üretilen soruları düzenleyerek sınavı hazırlayacak, öğrenci kullanıcısı sınav olarak sistemi kullanacaktır. Yönetici kullanıcısının ise Web tabanlı sistemin güvenlik ve üye yönetimi hakkı bulunmaktadır.

- **Varlık İlişki Diyagramı**

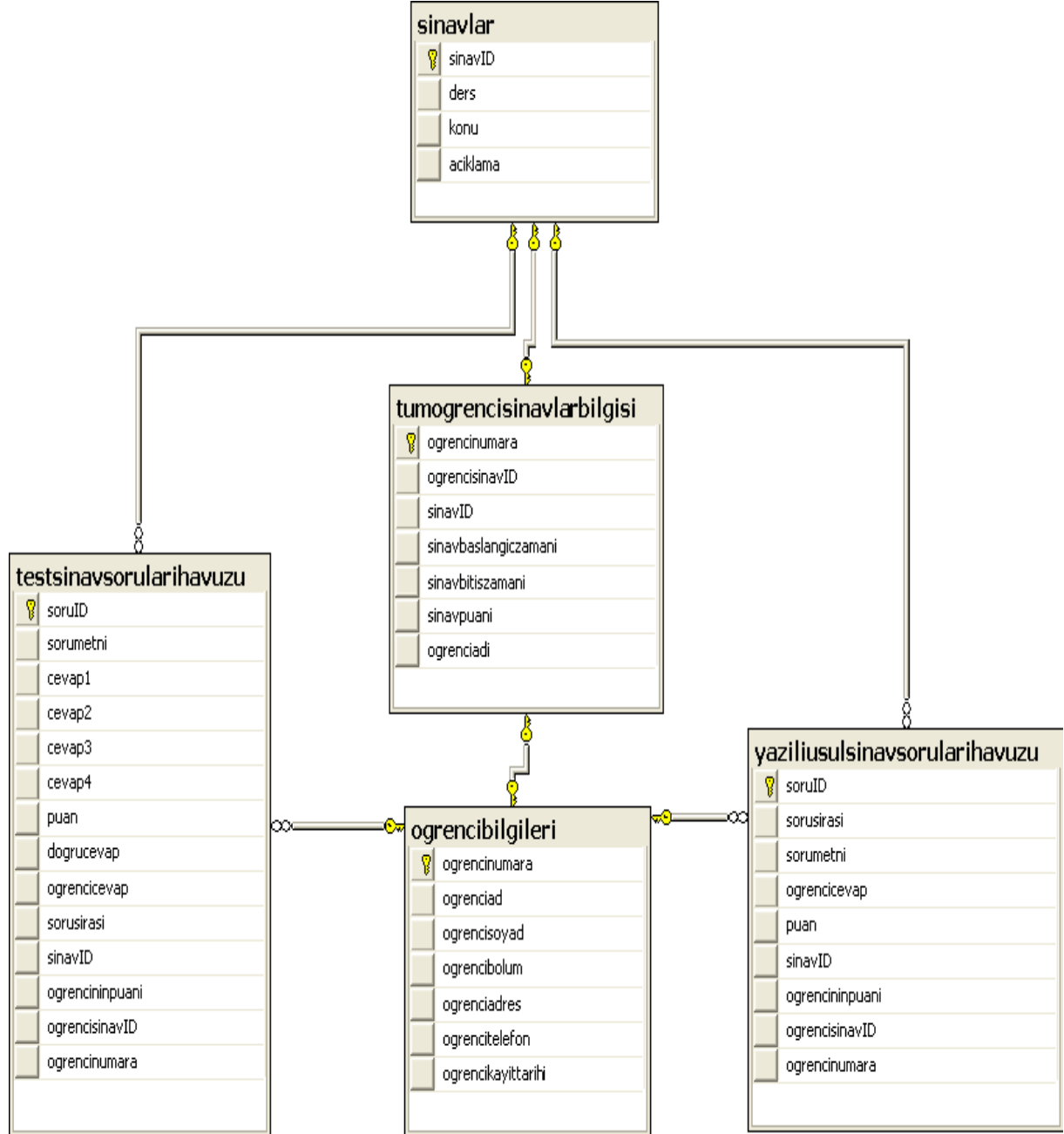
Sistemin büyüklüğü ve ileriye doğru büyüebilirliği göz önüne alınarak sistemin veritabanlarında bulunan veri düzeni yapısı ve bunlar arasındaki ilişkiler aşağıdaki varlık ilişki diyagramında gösterilmiştir.



Şekil 8 – Varlık İlişki Diyagramı

- **Sınav Veritabanı Diyagramı**

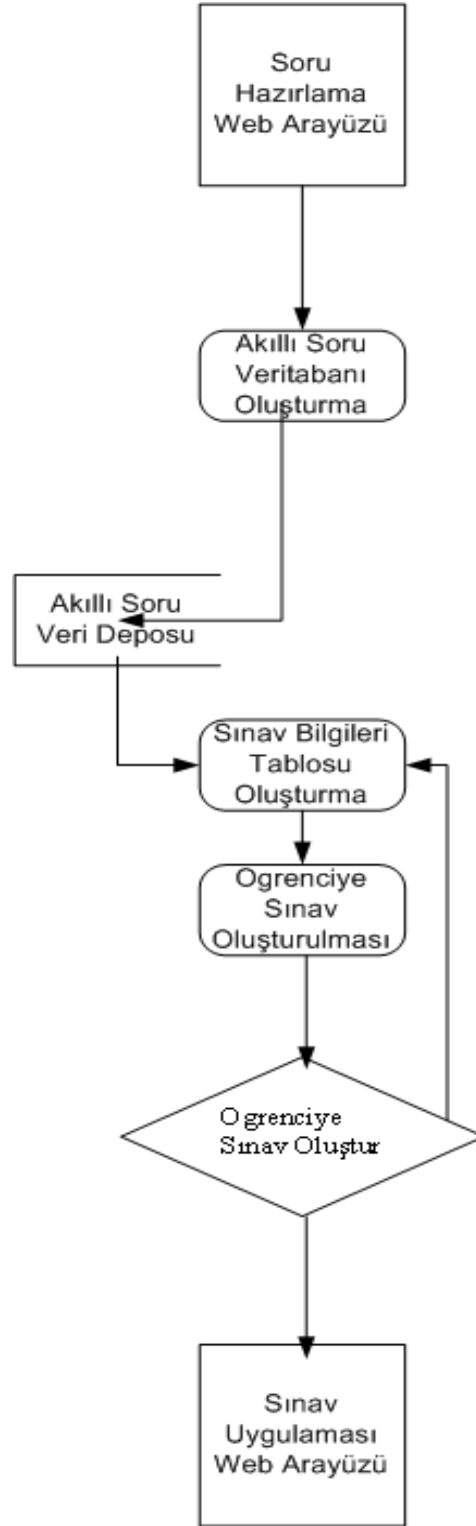
Sınav veritabanında mevcut bulunan tablolar, veri alanlarını oluşturan kolon adları ve tablolar arası ilişkilerin gösterildiği şema kullanılarak sınav veritabanı geliştirilmiştir.



Şekil 9 – Sınav Veritabanı Diyagramı

- **Veri Akış Modeli**

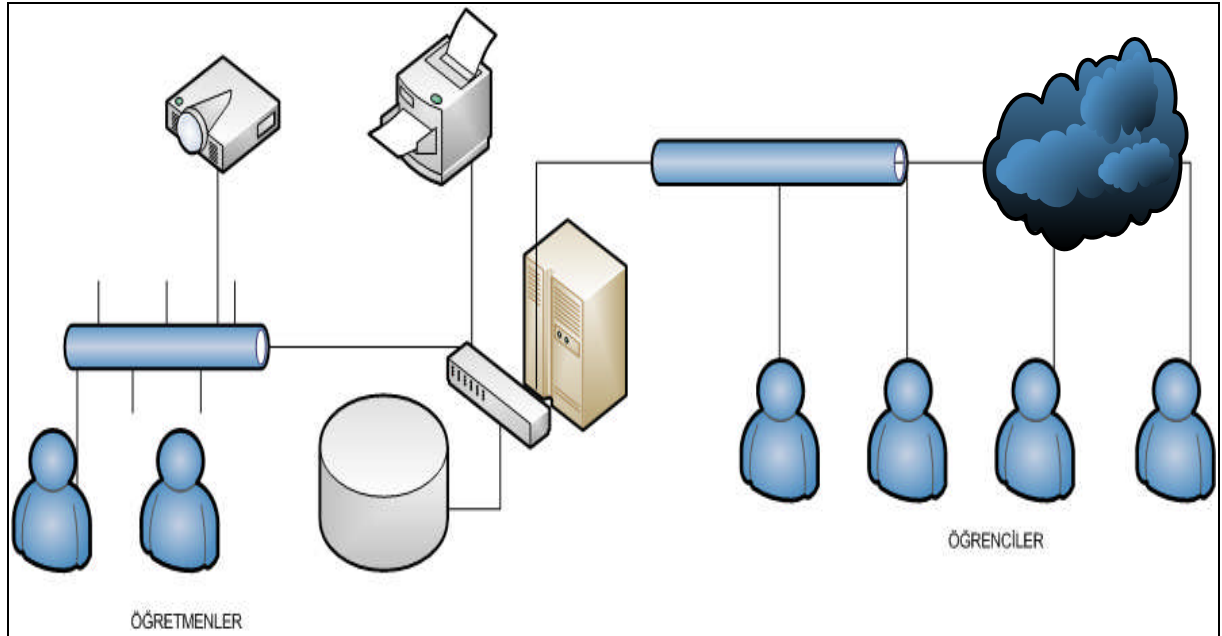
Yazılımın çalışma zamanında sistemin ögeleri arasında kurulan veri iletişimi aşağıda gösterilmiştir.



Şekil 10 – Veri Akışı Modeli

- **Sistemin Tasarımı**

Yazılımın asıl tasarımı klasik çevrim modeline göre tasarım aşamasında yapılmıştır. Bu aşamada ise tasarıma hazırlık işlemleri ve ön tasarım yapılmıştır. Çözümleme sonucu oluşturulan veri modellerine göre sistemin genel mimarisi belirlenmiştir. Sistem araç gereçlerinin donanım ve yazılım unsurlarının sistemi kullanan öğeler arasında dağılımı yapılmıştır. Merkezi tümleştirme mimarisine göre gerçekleştirilecek sistemin mimarisinde akıllı soru hazırlama ve sınav hizmeti veren ana bilgisayar Web ve veritabanı sunucusu rolüyle merkezde bulunur. Ayrıca Web tabanlı olarak sistemin üyelik ve güvenlik işlemlerini de yönetir. İster uzaktan İnternet bağlantısıyla, istenirse de yerel ağ içerisinde sistem kullanılabilir. Sistem bilgileri merkezi bir veritabanında saklanmaktadır. Sistemin Merkezi Tümleştirme Mimarisine göre sistemin genel yapısı aşağıda verilmiştir.



**Şekil 11 – Merkezi Tümleştirme Mimarisine Göre Sistemin Genel Yapısı**

Gerek İnternet üzerinden kullanıcı erişimleri gerekse İntranet yapısında bulunan kullanıcıların sistemi kullanmaları bilgisayar sistemlerinin kullandıkları yazılım ve donanım ortamlarının oluşturduğu platformdan bağımsız olarak gerçekleşir. TCP/IP protokol kümesi ve yapısı ağ içinde ve ağlar arası iletişim işlemlerini kotarır.[16]

- **Sistemin Gerçekleştirilmesi**

Seçilen klasik çevrim modeline göre yazılım gerçekleştirilmiştir. Tanımlanan veri modellerine göre oluşturulan yazılım ögeleri her aşamada kendi içinde test edilerek kodlanmıştır. Kodlama aşamasında yazılım niteliklerine dikkat edilmiş, kodlama uygulanan testlerle en iyi getirilmiştir. Yazılımın uygulama aşamasında gerçekleştirme işlemlerinin açıklanmasına daha çok yer verilmiştir.

- **Sistemin Testi ve Teslimi**

Tamamlanan ögelerin, kodlanması ve her seferinde test edilmesi sonucu elde edilen yazılım sistemin tümleştirilmesiyle tekrar test edilmektedir. Daha iyi sonuçların elde edilmesi için test işlemleri her aşamanın içerisinde tekrarlanmıştır. Karşılaşılan sorunlar çözülerek test işlemlerine devam edilmiştir.

Sistemin klasik proje yaşam çevrimine göre teslim aşamasında sunumu gerçekleştirilmesidir. Tezin konusunu oluşturan proje tamamlandıktan sonra sunumu yapılacaktır.

### **2.2.5. İşletme ve Bakım Süreci**

Sistemin işleme testi yazılım ve veritabanı kod testi belirli bir süre daha yapılmasıdır.

Bakım süreci ise işletme sürecinde çıkan veya olabileceği varsayılan sorunlar raporlanarak düzeltilmesidir.

- Projede IEEE/EIA 12207 bakım süreci standartları ve IEEE 1219 standartları temel alınarak bakım süreci gerçekleştirilmiştir.
- Yazılım bakım sürecinde IEEE 1219-1998 standartları ( IEEE Std 1219-1998 IEEE Standard for Software Maintenance ) içerisinde tanımlanan aşağıdaki aşamalar temel alınarak yazılım bakım işlemleri yapılmıştır.
  - ✓ Problemi tanımlama, sınıflandırma, yeniden şekillendirme, öncelikleri belirleme
  - ✓ Çözümleme
  - ✓ Tasarım
  - ✓ Gerçekleştirme
  - ✓ Sistem testi
  - ✓ Kabul edilebilirlik testi
  - ✓ Teslim [17]

## 2.3. Risk Yönetimi Risklerin Tanımlanması ve Risklerin Çözümlemesi

### 2.3.1. Kullanıcı Hatalarından Kaynaklanan Riskler

Kullanıcı soru üreticini kullanırken Türkçe yazım kurallarına veya dilbilgisi kurallarına uymayan metinler girebilir. Bu durum yazılım doğru sonuç üretmemesine yol açacaktır. Kullanıcılar dilbilgisi kurallarına uygun metin kullanmaları konusunda bilgilendirilmelidir. Bunun yanı sıra çok sayıda kullanıcının sistemi kullanması durumunda donanım ihtiyaçları kullanıcı sayısına bağlı olarak artacaktır. Sistemde oluşabilecek darboğazlara karşı ağ denetim ve gözetim araçları kullanarak sonuçlara göre tedbirler alınmalıdır.

### 2.3.2. Doğal Dilin Yapısından Kaynaklanan Riskler

Karşılaşılma ihtimali olan riskler çözümleme yapılarak en aza indirilmeye hafifletilmeye çalışılmıştır. Normal dilbilgisi kuralları dışında istisnalar ile ilgili çok sayıda kural yazılmıştır.

Kelime modeline göre kelime türü bulma modülünde uç değerlerler belirlenerek yapılan test çalışmaları sırasında beklendiği gibi çalışmayan durumlara rastlanmıştır. Dilbilgisi isim alanı içinde yazılan normal dilbilgisi kurallarını uygulayan sınıfların içindeki fonksiyonlardaki kurallara uymayan istisna durumlar yakalanıp, bu sorunlara karşı yeni kurallar üretilerek çözümler sağlanmıştır. Örnek vermek gerekirse:

İsmin hâlleri göre çözümleme yapılırken bir kelime birden fazla kök ve ekin çözümü olabilir.

Bu ihtimaller göze alınarak anlam seviyesinde ikinci bir değerlendirme yapılır. Anlamsız sözcükler elenir. Hamile kelimesi ismin hâllerine göre çözümlenirse çözüm olasılığı olarak sadece 'e' ek hâlinde değerlendirilir. Bu örnekte olduğu gibi Türkçe dilbilgisi kurallarına uymayan durumlar çözüm dışı tutulur.

Örnek: hami + le ihtimali elenir. Çünkü hami kökü eğer çekim eki alsaydı 'y' kaynaştırma harfi olarak hamiyle olmalıydı!

Hamile + le = hamileyle örneği kurallara uygun ve anlamlıdır. Programda doğru olarak çözümlenir.

Yumuşamış kökler düzeltilerek değerlendirmeye alınmaktadır. Bu sırada sertleşme yumuşama kuralına uymayan 3 harfli ğ ile biten sözcüklerde kontrol edilerek doğru değerlendirilir.

Örnek: dađı = dađ + ı, ocađı= ocak+ı, kitaba = kitap+a, havucu=havuç+u

Bazı kelimelerin aldığı eklerde dilbilgisi kurallarının dışına çıkılıyor. Örnek:



Ünlü uyumu kurallarına göre Kemalın, Cemalın, Meralın, Celalın, olması gerekirken Türkçe kullanımları

Kemal + in, Cemal + e, Şevval + de, Meral + in, Celal + in, helal + in olmaktadır.

Ünlü uyumuna uymayan Cemale, Kemalın, Şevvalden gibi kelimelerin kök ve eklerini bulmak için uygulanan ara denetim ile bu kelimelerinde çözümü sağlanmıştır.

(Emrahın, Serapın, terası, gibi örneklerde görünen bulunma durumları bozulmadan çalışan kurallar yazılmıştır. )

İyelik ekleri için iki ihtimalde bulunur.

Örnek: seri+m=serim(şimdi dağıttığım benim serim) ser+im=serim(başım anlamında)

Kelime 3.tekil iyelik alırken de iki ihtimal bulunabilir.

Örnek: klas+1 oda sı odas+1 parisi = Paris+i pari+si anlamsız olsada iki ihtimalde çalışır.

“Doluyor” kelimesi cümle içerisinde yüklem görevi aldığında iki farklı kök olabilir.

1.örnek: dol + uyor= kova suyla doluyor.

2. örnek: dola + uyor=şişin etrafını iple doluyor.

Yüklem olan kelimenin kökü dola, ek a düşerek kökün farklı bir eylem kökü ile aynı olmasına yol açıyor.

Gelecek zamanda kaynaştırma harfi olan durumlarda da farklı iki kök olma ihtimali vardır.

Örnek: arayacak=ara +y+acak, kay+acak,

Tüm ihtimaller gelmiş farz edilerek anlam seviyesinde cümle anlamına bakılarak eleme yapılmalı ve doğru olan ihtimal öğretmen tarafından seçilmelidir.

Geniş zamanda uyum kuralı farklılaşıyor. Örnek: gelir, gezer, koşar, çözer, kalır, dalar, bilir, bulur, üremek: ürer, ürür, döker, söker, kalır, dalar, sürer, sürür (her iki durumda göz önünde bulundurulur)

Şimdiki zamanda ara+ (ı)yor = arıyor, tarıyor, topluyor örneklerinde a sesi düşüyor. Kalın sesliyse;

ele+(i)yor=eliyor, özlüyor, deniyor, çözümlü+(ü)yor = çözümlüyor e sesi düşüyor.

İnce sesliyse iki ihtimalli olarak gelmeli, anlamsızlar atılmalıdır.

Yukarıdaki bütün durumlar için kurallar oluşturularak programın doğru çalışacağı çözümler elde

edilmiştir. Ancak Türkçe dilbilgisi yapısında istisnalar çok sayıda olduğu için bazı durumlar istenildiği gibi çalışmayacaktır. Burada çalışılan doğal dil olduğu için buradaki zorlukları hafifletmek amacıyla örneklem almak ve belirli kurallara dikkat edilerek oluşturulmuş metin kullanmak daha kesin sonuçlar verecektir. [18]

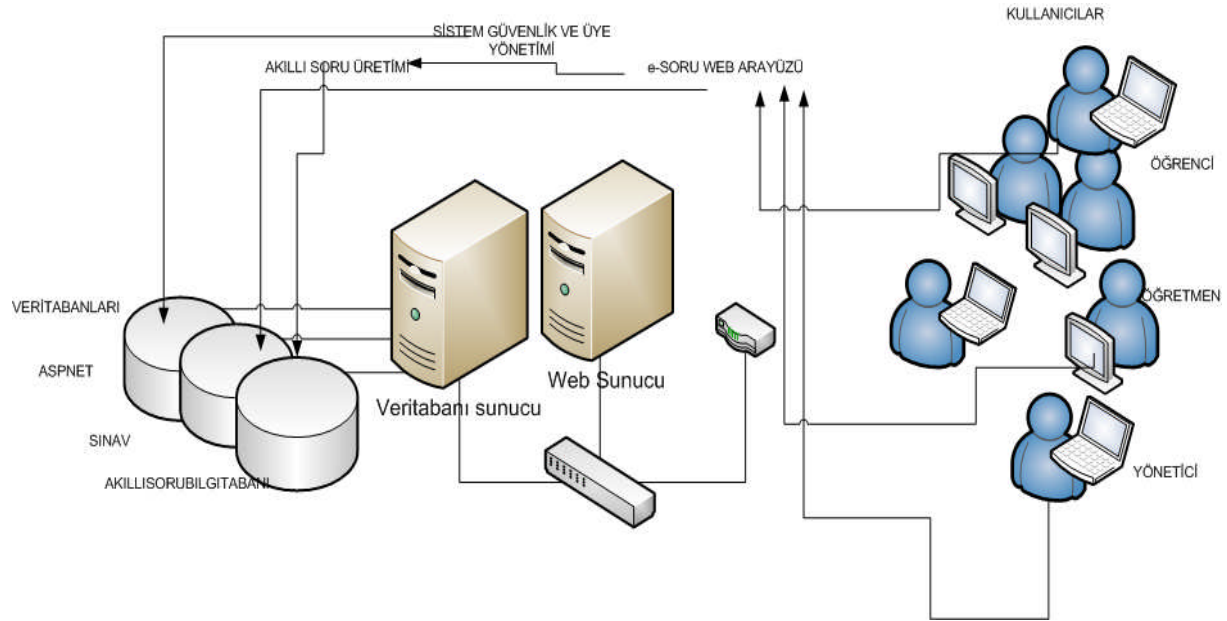
Ayrıca kelimenin cümle içindeki anlamı İstek kipinde kaynaştırma ihtimali (ara+ya ): Hele bir araya. Kuzuları çimene yaya iki anlamada gelebilir. Bu gibi durumlarda anlamsız üretilen sorular olacaktır. Bu gibi soruları öğretmen sınav sorusu hazırlama aşamasında eleyecektir.

### 3. AKILLI e-SORU SINAV SİSTEMİNİN TEMELLERİ ve KULLANILAN METİN ÇÖZÜMLEME YAPISI

Bu bölümde metin tabanlı çözümleme işlemin çalışma prensibi ve sistemin parçalarını oluşturan yazılımın temel elemanları anlatılacaktır.

#### 3.1. Sistemin Genel Mimarisi

Sisteme genel olarak bakıldığında sistemin işlevi dışarıdan gelen doğal dil metinlerini işleyen, cümlelerin türlerine göre kelimelerini çözümleyerek eklerine ayıran ve bu çözümlemeden yola çıkarak sorular üreten bir mekanizma yardımıyla veritabanına kaydedilen bu soruların sına giren öğrencilere Web ara yüzü aracılığıyla sorulmasıdır. Uçtan uca iletişim için sistemin genel yapısını gösteren şema aşağıda yer almaktadır.



Şekil 12 – Uçtan Uca Akıllı e-Soru Sınav Sisteminin Genel Mimarisi

#### 3.2. Sistemin Girdileri Çıktıları ve Gerçekleşen Süreçler

Soru oluřturulması srecinden đrenciye sınav uygulanıp deđerlendirilmesine kadar geen srete programdaki rollerin basit olarak dađılımı ařađıdaki řekilde verilmiřtir. Bu sre gerekleřirken ilk blmde yetkili kullanıcının belirlediđi izinler dâhilinde đretmen kullanıcısı sisteme giriř yaparak sınav oluřturulması iřlemlerini kurar. Bu iřlemi ders ve konu seimi yaparak, dersin konularını ieren metinleri yazılma gndererek bařlatır. retilen ok sayıda soru arasından istemediđi soruları eleyerek kendisi iin sorulabilir nitelikte grdđ soruları veritabanına isterse kendi vereceđi puan sistemi ile kaydeder. Bu kayıt iřlemi sırasında ders ve konuda belirlendiđi iin sorular sınıflandırılmıř olmaktadır.

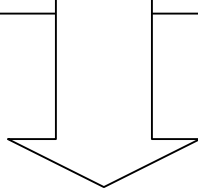
Soru hazırlama iřlemlerinden sonra istediđi bir zamanda đretmen kullanıcısı sınav hazırlama iřlemlerini yapar. Sınav hazırlarken ders, konu, sınavla girecek đrenciler, sınavın bařlama ve bitiř zamanı belirlenir ve sınav aktif hâle getirilir. Aktif hâle gelen sınavları ilgili đrencilerin alabilmesi iin đrencinin kendi hesabıyla sisteme gvenli giriř yapması ve kendisi iin aktif olan sınavı bařlatması yeterlidir.

Sınavı tamamlayan đrenci sınavın tipi test ise anında deđerlendirme sonucunu grr. Eđer sınav klasik usulde hazırlanmıřsa sınav sonularını đretmenin deđerlendirmesinin ardından grr. Sınav sonuları sistem veritabanına kaydedilerek saklanır. İstenildiđi zaman gemiř tarihteki sonular raporlanabilir.

Bu sreleri temsil eden řekil devam eden sayfada verilmiřtir.

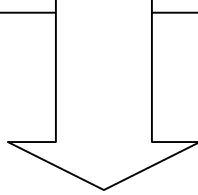
### **Giriş Birimleri**

2. Metin Bilgileri (Ders konusunu oluşturan metin içerikleri)
3. Öğrenci Bilgileri (Örnek: Öğrenci kimlik bilgileri)
4. Öğretmen Bilgileri (Örnek: Öğretmen adı, Yetkisi )
5. Soru/Sınav Bilgileri (Örnek: SınavID, Puan, Doğru cevap )



### **İşlem Süreci**

- Veritabanından Giriş Onayı
- Soruların Oluşturulması
- Uygun Sınavın Oluşturulması
- Sınav Güvenliğinin Sağlanması
- Üyelik Yönetiminin sağlanması
- Sistemin Gereksinimleri Karşılıyarak Devam Ettirilmesi



### **Çıkış Birimleri**

- Sınav Sisteminin Ara yüzlerinin Kullanıcıya Sunulması
- Sonuçların Değerlendirilmesi ve Görüntülenmesi
- Sınav Bilgilerinin ve Soruların Saklanması

**Şekil 13 – Sistem Unsurlarının Giriş Çıkış ve İşleme Süreci**



Türkçe sondan eklemeli dillerdendir. Bundan dolayı kök sözcükten yeni kelime türetmek için kelimenin sonuna yapım ekleri eklenir. Bu özelliğinden dolayı Türkçe'deki az sayıdaki kökten çok sayıda kelime türetilebilir.

Kelime gövdelerinin sonuna eklenen ve yapım eklerinden sonra gelen çekim ekleri vasıtasıyla cümlede kelimenin görevi diğer kelimelerle ilişkisi ve cümle içindeki anlamını belirlenir. [22]

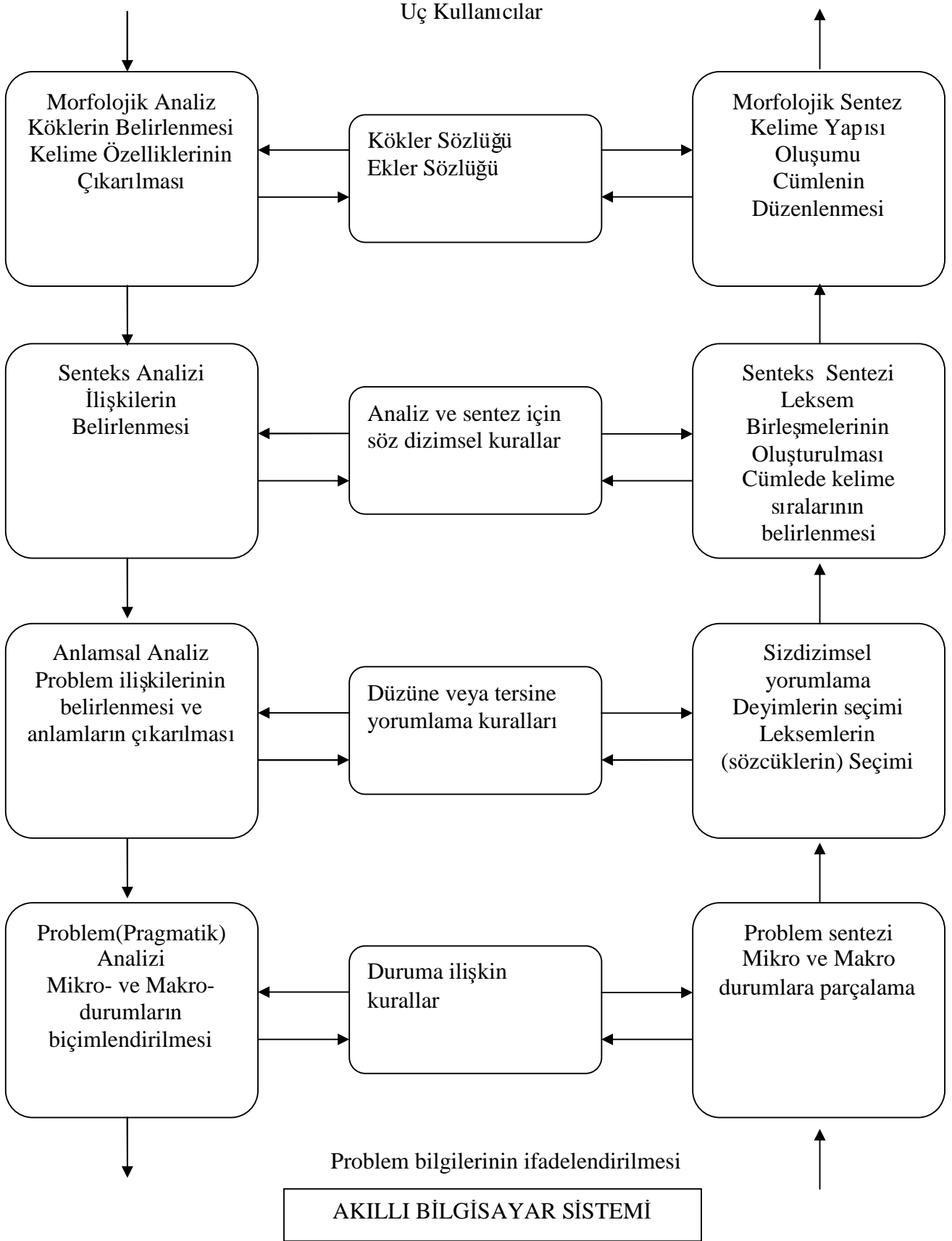
Türkçe kelimelerin ve eklerin yapısını aşağıda bir formülle ifade edilmiştir.

Kelime = kök + yapım ek(ler)i [ - ] + çekim ek(ler)i [ - ]
--

### 3.3.2. Doğal Dil Çözümleyicilerin Genel Yapısı

Doğal dilde incelenen metinlerin yazılım tarafından kullanılabilmesi, gelen metnin öğelerinden sistemde yararlanılabilecek biçimlerin elde edilmesi doğal dil çözümleme yöntemleriyle gerçekleşmektedir. DDİ (NLP) metotları uygulanarak yapılan çözümlenmelerle amaçlanan programda veri olarak kullanılacak sözdizimsel yapıların ve anlam ilişkilerinin elde edilmesidir.

Takip eden sayfada doğal dil çözümleyici yapılarının genel şeması verilmiştir.



Şekil 15 – Doğal Dil Çözümleyici Yapısı [17]

Yapılan alıřmalar sırasında sistemdeki iř akıřına girdi olarak verilen metinlerin özümlemesine cümlelerden bařlamıřtır. Cümlelerin Türke dilbilgisi sözdizimi kuralları modellere dönüřtürülmüřtür. Bu modellerdeki kurallara göre cümle öęeleri elde edilmesi saęlanmaktadır.

### **3.3.3. Biibilim Analiz**

“Morfoloji kelimesi řekillerin incelenmesi anlamına gelmektedir.” [23]

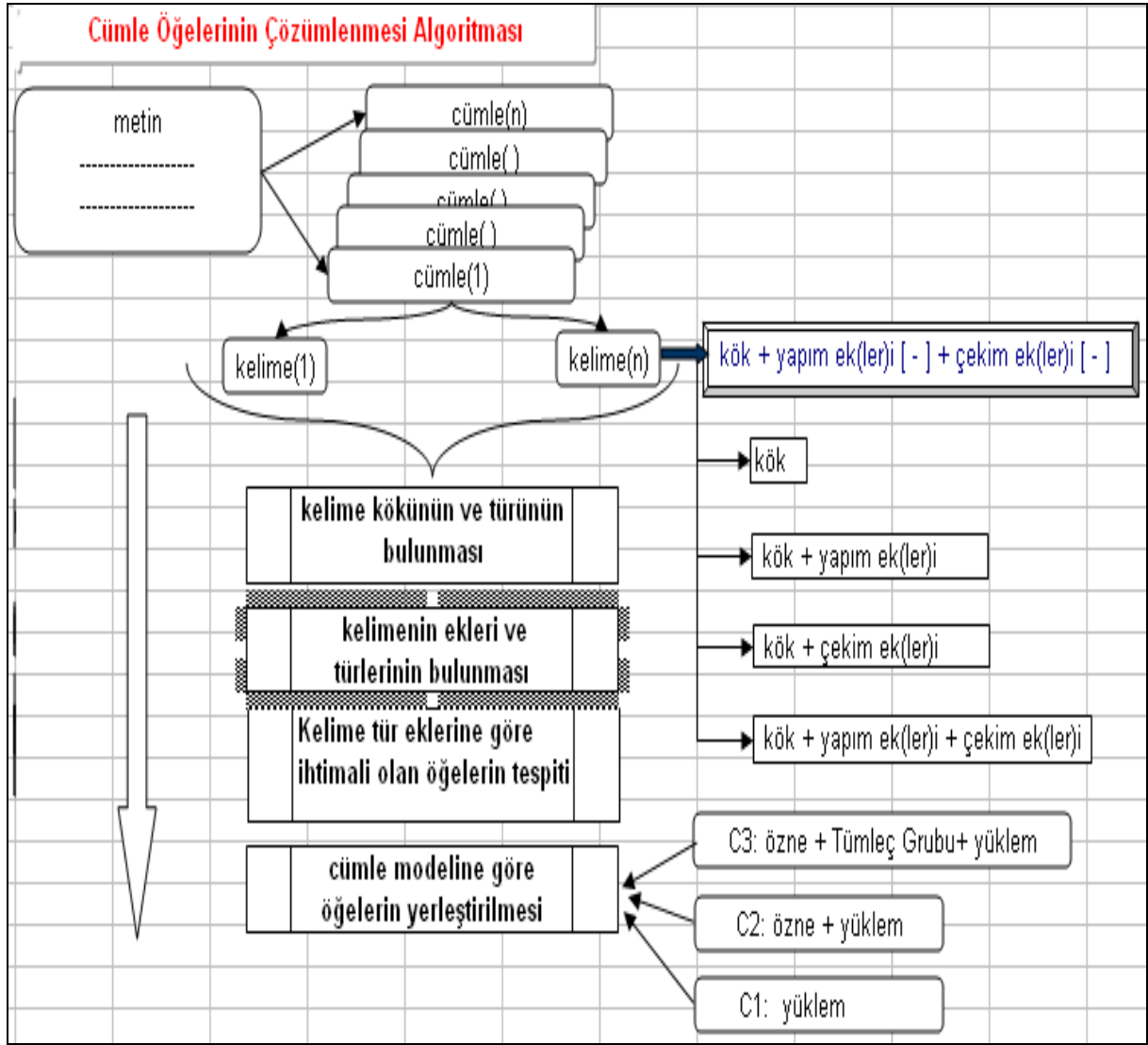
Kelimelerin eklerini ve köklerini ararken kelime köklerini hazırlanan dilin kelimelerini ieren sözlükler kullanılır. Bu tez alıřmasında morfolojik özümlemede kullanılan iki sözlük oluřturma yönteminden deklaratif yöntem seçilmiřtir.

Deklaratif yöntemle Sözlüęe eklenen kelimelerin türleri ile birlikte tutulduęu veritabanı tablosu oluřturulmuřtur. Biibilimsel özümleme yapılırken kök belirleme, biibilimsel testler ve biibilimsel ayrıřtırma süreçleri uygulanır.

### **3.4. Cümle Öęelerini özümleme Algoritması**

Takip eden sayfada uygulamada kullanılan cümle öęelerini özümleme algoritmasının řeması verilmiřtir.





**Şekil 16 – Cümle Öğelerini Çözümleme Algoritması**

”Gramer çözümü, bir dile dayandırılarak ortaya koyulan metnin veya programın yazım kurallarını çözümlmek, ayırtmaktır. ” [24]

Yapılan çalışmada cümlelerin öğelerine ayrılmasının olabirliği öğelerin bulunması için kelime türlerinin dilbilgisi kurallarına göre modellendiği kelime veri modellerinin çözümlendiği algoritmalar sayesinde gerçekleşmektedir.

### 3.4.1. Kaba Kod Biçiminde Çözümleme Algoritması

Genel olarak cümle çözümleme algoritmanın akış sırası kaba kod biçiminde takip eden satırlarda yazılmıştır.

1. Programın başlatılarak, değişkenlerle ilgili işlemler yapılır.
  - 1.1. Değişkenlerin tanımı, başlatılması ve ilk değer atamaları yapılır.
  - 1.2. Cümlelerin tutulacağı dinamik dizi tanımlanır.
  - 1.3. Kelimelerin tutulacağı yığın tanımlanır.
  - 1.4. Yığından çekilecek kelimelerin atılacağı yeni bir dizi oluşturulur.
  - 1.5. Yığından çekilecek kelimelerin atılacağı yeni bir dizi oluşturulur.
  - 1.6. Soruyu oluşturan cümle ile ilgili değişkenler tanımlanır.
2. Bir döngü içinde metnin başından sonuna kadar çözümleme yapılır.
  - 2.1. İşleminin ilk alt adımında metin içinde ilk cümle bulunur, son cümle bulunana kadar alt adımlardaki işlemler tekrarlanır.
    - 2.1.1. Cümlede kelime modelleri aranmadan önce cümle dışı unsur olarak adlandırılan semboller ve ünlem, bağlaç, edat gibi kelimeler çıkarılarak cümlenin kalan öğeleri için inceleme yapılır.
    - 2.1.2. Cümlenin sonundan başına doğru her bir kelime için sırayla basit çekimli fiil türü kelime modeli, birleşik çekimli fiil türü kelime modeli, isim-fiil türü kelime modeli, isim türü kelime modeli taraması yapılır.
    - 2.1.3. Tarama işlemi yapılırken her kelime ihtimal olarak o modelin tüm varyasyonlarıyla eşleştirilerek taranır, olası tüm model desenleri taranır.
    - 2.1.4. Eğer kelime herhangi bir desenle eşleşirse yüksek olasılıkta kelimenin ekleri ve türü tespit edilmiştir. Düşük olasılıklarda olsa da eklerin desene uymasına rağmen kok olarak tespit edilen değerler anlamsız olabilir.
    - 2.1.5. Kesin sonuç elde etmek için veritabanında TDK sözlükten yararlanılarak oluşturulmuş olan kök türleri sözlüğü kullanılır. Olası kök sözlükle karşılaştırılarak bir anlamı olup olmadığı tespit edilir.
      - 2.1.5.1. Fiil kökü sözlükte varsa onaylanır.
      - 2.1.5.2. İsim kökü sözlükte varsa onaylanır.
    - 2.1.6. Onaylanan kelimelerden alınan veriler çeşitli soru üretme yöntemleri kullanan fonksiyonlara dinamik olarak sorular üretilir.
  - 2.2. Sırayla tüm kelimeler tarandıktan sonra deyimlerin, özel ifadelerin ve eğer oluşturulmuşsa ilgili dersin kavramlarını içeren sözcüklerin ait olan veritabanından taranması gerçekleştirilir.

Bu tarama işlemi cümlenin başından sonuna doğru bir döngü içinde yapılır. Döngü içinde taranan karakterlerdir.

    - 2.2.1. İlk karakterden cümlenin son karakterine kadar tarama yapılır.
    - 2.2.2. Döngü içinde yeni bir döngü çalıştır. Bu döngüde ilk karakterden son karaktere kadar her seferinde karakterler başlangıcını bir artırarak belirle.
    - 2.2.3. Karakter boşluk değilse ve cümle sonu değilse çalış.
    - 2.2.4. İç içe üçüncü seviyedeki döngüyü çalıştır.
    - 2.2.5. Bu döngü içinde her seferinde cümlenin karakterlerini tarayarak ara kelimeler oluştur.
    - 2.2.6. Ara kelimeleri veritabanındaki bilgilerle karşılaştırmak için parametre olarak kullanılacak metin kutusuna atılır.
    - 2.2.7. Veritabanı bağlantısını sağlayarak karşılaştırmayı gerçekleştir.
    - 2.2.8. Tespit edilen kavram veya özel ifadelerin verileri ilgili soru üretme fonksiyonlarına iletilerek çeşitli sorular üret.
3. Programın çıktıları sayfadaki liste kutusuna yazdırılarak program sonlandırılır.

Şekil 17 – Kaba Kod Çözümleme Algoritması

### 3.4.2. Kelime Veri Modelleri

Aşağıdaki tablolarda fiil ve isim için hazırlanan veri modellerine örnekler verilmiştir.

ÇLE	Çoğul Eki
EKE	Ek Fiil Eki
HLE	Hal Eki
İÇE	İsim Çekim Eki
İYE	İyelik Eki
KOK	Kelime Kökünün Türü
OZE	Olumsuzluk Eki
SRE	Soru Eki
ŞHE	Şahıs Eki
ZŞE	Zaman Şekil Eki

Tablo 2 – İsim-Fiil Türü Kelimeler Veri Modeli Örneği

İsim-Fiil Türü Kelimeler Veri Modeli	Cümledeki sırası ve görevi	KOK	ÇLE	İYE	HLE	EKE	ŞHE	SRE
<b>Ek</b>	2	Kırmızı	-	-	-	-	dırlar	mı
<b>Türü</b>	Dolaylı tümleş	Sıfat	-	-	-	-	3.çoğul şahıs	soru eki

Tablo 3 – İsim Türü Kelimeler Veri Modeli Örneği

İsim Türü Kelimeler Veri Modeli	Cümledeki sırası ve görevi	KOK	EKE	ŞHE	SRE
<b>Ek</b>	4	Değil	miş	Sin	-
<b>Türü</b>	Olumsuz isim fiil cümlesi	olumsuzluk anlamı veren kelime	mişli geçmiş zaman	2.tekil şahıs	-

Tablo 4 - Cümlelerin Kelimeleri Birleşik Veri Modeli Örneği

Cümlelerin Kelimeleri Birleşik Veri Modeli	Cümledeki sırası ve görevi	KOK	OZE	ÇLE	İYE	HLE	ZŞE	EKE	ŞHE	SRE
<b>ek</b>	3	Gel	me	-	-	-	di	-	m	-
<b>türü</b>	Yüklem	Fiil	olumsuzluk eki	-	-	-	geçmiş zaman	-	1.tekil şahıs	-

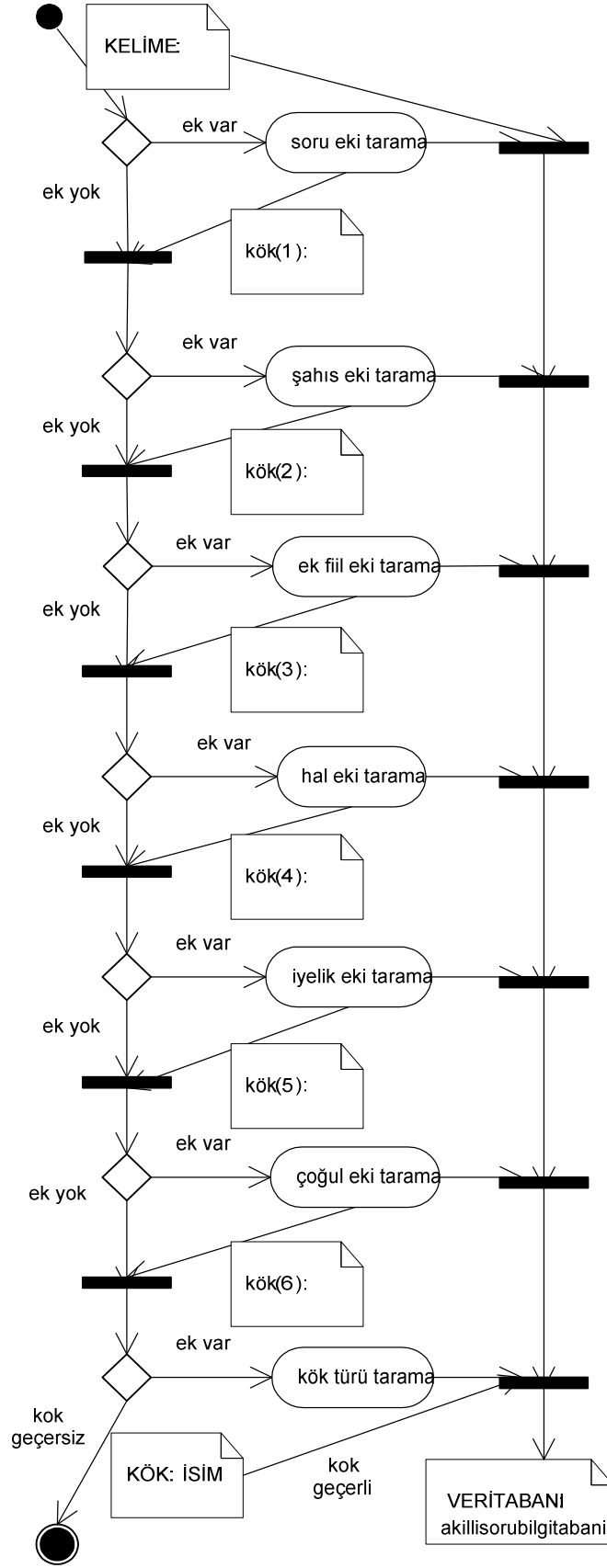
**Tablo 5 – Fiil Türü Kelimeler (basit veya bileşik çekimli fiiller) Veri Modeli Örneği**

<b>Fiil Türü Kelimeler (çekimli fiil) Veri Modeli</b>	<b>Cümledeki sırası ve görevi</b>	<b>KOK</b>	<b>OE</b>	<b>ZŞE</b>	<b>Y kaynaştırma harfi durumu</b>	<b>EKE</b>	<b>ŞE</b>	<b>Boş luk</b>	<b>SRE</b>
	<i>0</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>6</i>	<i>7</i>	<i>8</i>
<b>ek</b>	5	Çalış		tı	-	-	n	-	mı
<b>türü</b>	yüklem	Fiil	olumsuzluk eki	geçmiş zaman	-	-	2.teki l şahıs	-	soru eki

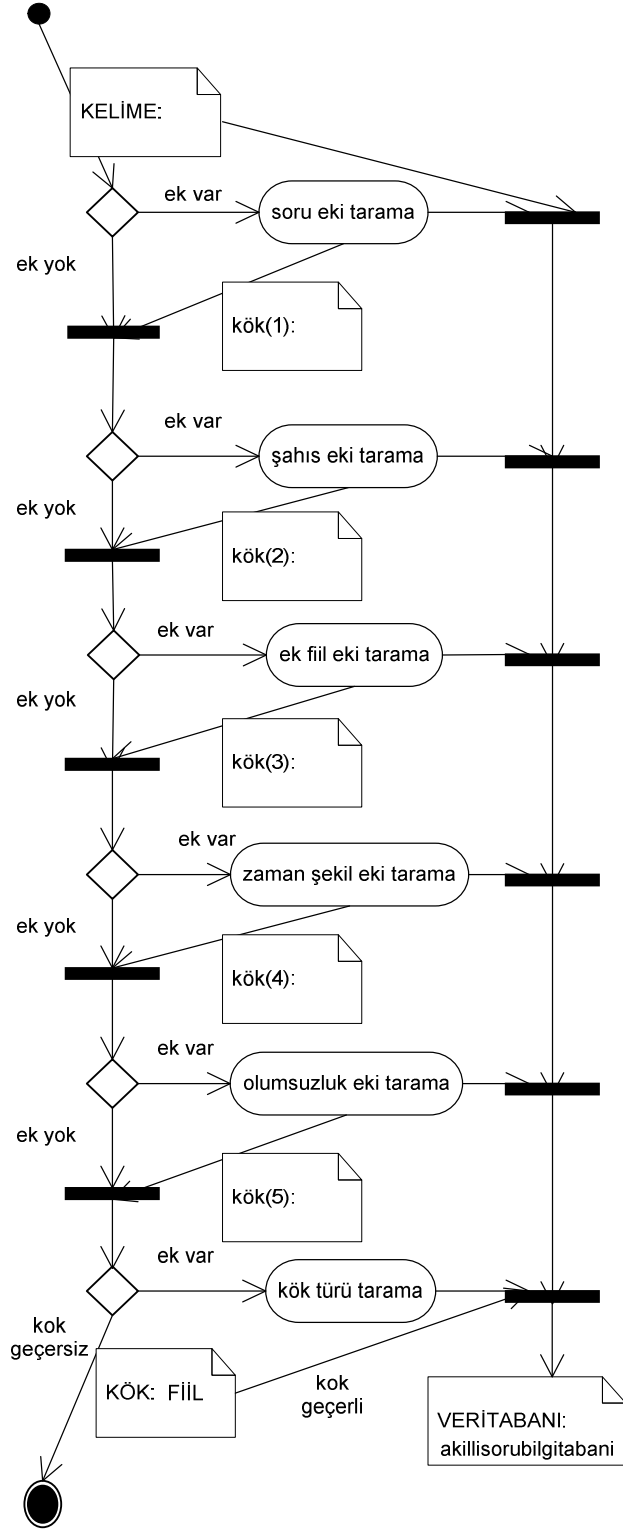
Bir kelimenin fiil olup olmadığı araştırılırken yukarıdaki tabloda verilen örnekte olduğu gibi olabilecek tüm biçimler sırayla sınırlanır. Bu işlem diğer kelime veri modelleri içinde geçerlidir. Kelimenin sonundan başına doğru, sağdan sola yönlü tarama yapılır. Karşılaştırmada eğer aranan ek aday kelime varsa bu ek ve türü oluşturulan veri modeli içindeki bölüme atılır. Karşılaştırma sonucunda değer uyumlu bir ek değeri elde edildiği sürece tarama devam eder. Son duruma ulaşırsa, fiil ekleri uyduğu için bu kelimenin fiil olduğu varsayılır. Böylece bir sözlüğe başvurmadan kelimenin kök ve ek türleri tespit edilir. Tüm ek bölümleri için aynı işlem tekrarlanır. Bu işlemler tasarlanan kelime modellerine göre uygulama içerisinde gerçekleştirilmektedir.

Çözümlemelerde ögenin ne olabileceğine dair farklı ihtimaller olabilmektedir. Bu ihtimaller yine de değerlendirilerek anlam aşamasında veritabanında TDK sözlüğünden faydalanılarak oluşturulan kök kelimeler ve türlerinin yapılandırıldığı kök tablosundan anlamı olup olmadığı sınırlanır.

Tasarlanan isim ve fiil veri modellerinin işlevlik diyagramları takip eden sayfalarda verilmiştir.



Şekil 18 - İsim Modeli UML Etkinlik Diyagramı



Şekil 19 - Fiil Modeli UML Etkinlik Diyagramı

## 4. SİSTEMİN YAZILIM TEMELLERİ

### 4.1. Yazılımda Kullanılan Temel Elemanlar

Yazılım geliştirilirken C# dili kullanılmıştır. C# programlama dilinin nesne tabanlı dil olması sayesinde sınıflar yazılarak, gerektiğinde bu sınıflardan nesnelere oluşturulmuştur. Bu bölümde yazılımda kullanılan başlıca C# unsurları ve görevlerine değinilmiştir.

Yazılımda oluşturulan dilbilgisi isim alanı sayesinde birbirleriyle ilişkili, karışıklığa yol açmadan veri iletişimi yapabilen dilbilgisi kurallarını oluşturan sınıflar etkin olarak çalışmaktadır.

“System.Collections.Generic” isim alanında bulunan “dictionary” içerisinde anahtar, anahtar değeri çiftleri tutularak programın daha verimli bellek kullanımını sağlanmıştır. Anahtar kullanımdan dolayı içinde tutulan elemanlara çok hızlı erişilmektedir.[25]

```
static Dictionary<iyelikler, List<IHarf>> iyelik_ekleri = new  
Dictionary<iyelikler, List<IHarf>>();
```

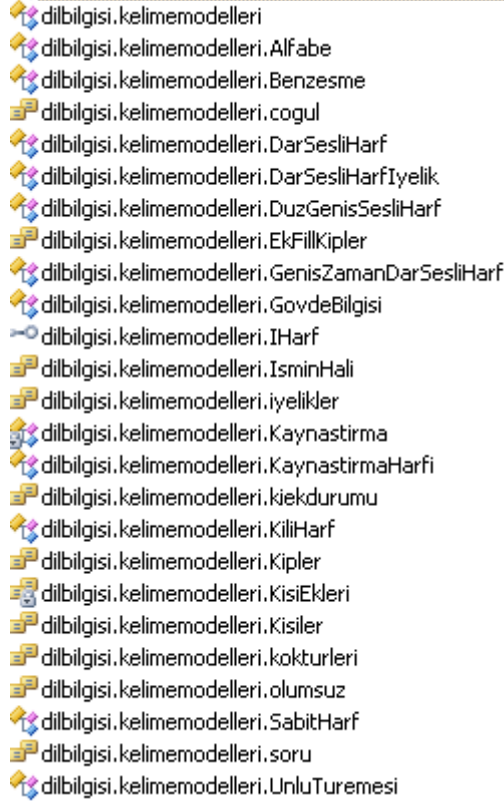
Yukarıdaki örnekteki gibi iyelikler, ismin hâlleri, şahıs ekleri, zaman ekleri, gibi dilbilgisi yapıları oluşturulan yeni veri tipleri içinde Enum olarak tanımlanmıştır. Elemanlar verilen eklere bağlı olarak yeni üretilecek olan liste içindeki değerle eşleştirilerek sözlüğe eklenecektir.

Programda kullanılan örnek bir liste aşağıda verilmiştir. İyelikler Enum tipi içindeki elemanlar verilen iyelik ekine bağlı olarak yeni üretilecek olan liste içindeki değerle eşleştirilerek sözlüğe eklenecektir.

```
new List<IHarf>  
(new IHarf[] { new DarSesliHarfIyelik(), new SabitHarf(Alfabe.M) })
```

Bu liste dinamik olarak programdaki IHarf arayüzünü kullanan DarSesliHarfIyelik ve SabitHarf sınıflarından iki yeni eleman üretip bu elemanları listeye eklemektedir. Ek desenleri oluşturulurken bu listeler kullanılmaktadır.

Yazılımda kullanılan IHarf arayüzü bu arayüzü kullanan dilbilgisi isim alanındaki sınıflarda tanımlanmıştır. Bu sınıflar dilbilgisi yapısını kuran kuralların yazıldığı sınıflar oluşturulmuştur.



Şekil 20 – Uygulamanın Dilbilgisi İsim Alanı Sınıf Yapıları

Aşağıdaki örnek Benzesme sınıfında sert sessiz benzeşmesi kuralını gerçekleştiren bir fonksiyon olmaktadır. Bu kurala göre bir kelimenin son harfi eğer alfabenin sert sessiz harflerinden biriye ve bu kelimeye sesli harf ile başlayan bir ek gelirse bu harf yumuşamaktadır.

```
public char Bas(GovdeBilgisi govde)
{
    if (Alfabe.SertSessizler.Contains(govde.SonHarf))
    {
        switch (benzes)
        {
            case Alfabe.B: benzes = Alfabe.P; break;
            case Alfabe.C: benzes = Alfabe.Ç; break;
            case Alfabe.D: benzes = Alfabe.T; break;
            case Alfabe.G: benzes = Alfabe.K; break;
            case Alfabe.Ğ: benzes = Alfabe.K; break;
            default: break;
        }
    }
}
```

#### 4.2. Kullanılan Veri Tipleri ve Veri Yapıları

Program geliştirilirken standart olarak kullanılan integer, string, boolean, char gibi veri tiplerinin yanısıra .Net framework yapısında sınıf olarak oluşturulmuş olan StringBuilder, Hashset, Arraylist, Stack gibi veri sınıfları da kullanılmıştır. Alfabe tanımlarında const ile sabit tanımları yapılmıştır ve alfabenin tüm harfleri tanımlanmıştır.



Uygulamada cümle çözümlenirken algoritma sağdan sola doğru çalıştığı için ters sıralanan ve sayısı bilinmeyen kelimelerin, sıralanması için cümlelerin kelime sayısının tespit eden kelime dizilişini çeviren yığın gösterilmiştir. System.Text sınıfına ait bir nesne olan “String Builder” veri türü uygulamada başlangıç kapasitesi belirtilmeden sonradan değiştirilebilen kapasiteli String değişken kullanmayı sağladığı için ve string sınıfına göre bellek performansı çok iyi olduğu için kullanılmıştır.[22]

```
Stack cumledekikelimeler = new Stack();
public static string[] CumleninKelimeleriniYaz(Stack yigin, int ks)//cümle yığın
içine atılarak ters sırada yapılan cümle kelime analizi ters çevrilerek düzeltiliyor
ve tüm kelimeler sırayla str[] dizisi içine atılıyor.
    {   int i=0;
        string[] str = new string[ks];
        object obj = new Object();
        Stack yeniYigin = (Stack)yigin.Clone();

        if(yigin.Count!=0)
        {
            while(yeniYigin.Count>0)
            {
                obj = yeniYigin.Pop(); str[i]=("\t"+ obj.ToString());i++; } }   return str;}
```

### 4.3. Soru Üretme Algoritması

Veri modellerine göre oluşturulan sınıf yapılarıyla soru üretimini sağlayan algoritma aşağıdadır.

1. Tanımlanan değişkenlere ilk değer atamaları yapılır.
2. Veritabanında cümle bilgilerinin geçici olarak tutulacağı tablo temizlenir.
3. Cümle metinden alınarak sağdan sola doğru taranmaya başlanır. Her durumda Geçici tablodaki elde edilmiş olan işlenmiş kok ve ek bilgiler dataset'e alınarak tekrar işlenir.
  - 3.1. Sağdan sola doğru cümlelerin kelimelere ayrılır. stack'e atılan kelimelerin sırası düzeltilerek str adlı diziye atılır
  - 3.2. Kelimeler sırayla modeller temel alınarak değerlendirilmeye tutulur.
  - 3.3. Her kelime isim ve isim-fiil modeli, isim modeli fiil modeli taramasından geçirilir. Sondan başa doğru tüm ek olasılıkları araştırılır.
  - 3.4. ilk olarak şahıs ekleri araştırılır.
  - 3.5. İkinci durumda ek fiil eki taraması yapılır.
  - 3.6. Üçüncü durumda hal eki taraması yapılır.
  - 3.7. Dördüncü durumda iyelik eki taraması yapılır.
  - 3.8. Beşinci durumda çoğul eki taraması yapılır.
  - 3.9. Altıncı durumda kokün anlamlı olup olmadığı sınanır. Cümle çözümleme algoritması doğru sonuçla birlikte düşük olasılıkta olsada yanlış (anlamsız) sonuçlar üretebilir.
  - 3.10. Eğer sözlüğe başvurulması istenirse, morfolojik analiz sonucu çözümleme algoritmasından elde edilen sonucun anlamlı olup olmadığının tespiti için kökler sözlüğüne bakılır.
  - 3.11. Sözlükteki bir kelimenin birkaç türde olabilir. İlk bulunan türler sırayla değerlendirilir.
  - 3.12. Kelimenin isim olmaması, şahıs eki almaması, 3.tekil şahıs olma ihtimali değerlendirilir.
4. Tarama işlemini sona erdikten sonra çıkan ekler değerlendirilerek sorular üretilir.
  - 4.1. İsmi hal eklerine bakılır, hal ekine göre farklı sorular üretilebilir. İhtimallere bağlı sorular aşağıda verilmiştir.

"Yonelme": soruifadesi " nereye/kime "
"Gosterme": soruifadesi " neyi/kimi "
"Cikma": soruifadesi " nereden/kimden "
"Bulunma": soruifadesi " nerede/kimde "
"Tamlama": soruifadesi " neyin/kimin "
"Vasita": soruifadesi " neyle/kimle "
"Esitlik": soruifadesi " nasıl "

- 4.2. İsim fiil 3. tekil şahıs için tarama dan sonra çıkan ekler değerlendirilerek “nedir” ile sorular üretilir.
- 4.3. Özneyi ve nesneyi soran soruları ek almayan isimleri kullanarak soru oluşturulur.
5. Üretilen sorular gecici tabloya atılarak aynı soru varsa elenir ve liste kutusuna sorular yazdırılır.

Şekil 21 – Soru Üretme Algoritması

Metinden cümle olarak alınır, cümlenin kelimeleri sırayla sağdan sola doğru incelenir. Örnek: bilgisayarın kelimesinin aldığı ekler tespit edilmek amacıyla sırasıyla isim, isim-fiil ve fiil veri modellerinin ek durumları taranarak eklerin olup olmadığı anlaşılır. Modellere göre ek taraması sonucu “ın” tamlama eki olarak bulunur ve tabloya bu bilgi atılır. Bilgisayar kök kelime olarak bulunur. Soru üretim safhasında kelimenin aldığı tamlama eki bilgisine göre “İşlemci neyin/kimin işleme süreçlerinde ana bellekten gelen bilgilerle hızlıca sonucu üreten birimdir.” sorusu üretilir.

Üretilen sorular:

- ✓ İşlemci nedir?
- ✓ İşlemci bilgisayarın işleme süreçlerinde ana bellekten gelen bilgilerle nasıl sonucu üreten birimdir.
- ✓ Ne/kim bilgisayarın işleme süreçlerinde ana bellekten gelen bilgilerle hızlıca sonucu üreten birimdir.
- ✓ İşlemci neyin/kimin işleme süreçlerinde ana bellekten gelen bilgilerle hızlıca sonucu üreten birimdir.
- ✓ İşlemci bilgisayarın nerede/kimde ana bellekten gelen bilgilerle hızlıca sonucu üreten birimdir.
- ✓ İşlemci bilgisayarın işleme süreçlerinde nereden/kimden gelen bilgilerle hızlıca sonucu üreten birimdir.
- ✓ İşlemci bilgisayarın işleme süreçlerinde ana bellekten gelen neyle/kimle hızlıca sonucu üreten birimdir.
- ✓ İşlemci bilgisayarın işleme süreçlerinde ana bellekten gelen bilgilerle hızlıca neyi/kimi üreten birimdir.

#### 4.4. Kullanılan Sınıflar

Uygulamada dilbilgisi isim yapısı altındaki sınıflardan metin çözümleme işlemini yapan sınıflar, Web arayüzü ve veritabanı işlemlerini oluşturan sınıflarla veri alışverişi yaparak süreçleri gerçekleştirmektedir.

#### 4.4.1. Sınıf Diyagramı

Uygulamada geliştirilen sınıf yapıları aşağıda gösterilmiştir. Bu sınıf yapılarını oluşturan bölümler ile uygulamadaki işlevler gerçekleştirilmektedir.

Kelimemodelleri	Alfabe
<pre>string[, ] : isimharfdeseni { get; set; } string : sertkelime { get; set; } cogul : enum soru : enum olumsuz : enum IsminHali : enum İyelikler : enum EkFillKipler : enum Kipler : enum Kisiler : enum KisiEkleri : enum kiekdurumu : enum kokturleri : enum</pre>	<pre>Sesliler : HashSet&lt;char&gt; DuzSesliler : HashSet&lt;char&gt; YuvarlakSesliler : HashSet&lt;char&gt; DarSesliler : HashSet&lt;char&gt; GenisSesliler : HashSet&lt;char&gt; KalinSesliler : HashSet&lt;char&gt; InceSesliler : HashSet&lt;char&gt; SertSessizler : HashSet&lt;char&gt; YumusakSessizler : HashSet&lt;char&gt;  A = 'a' : const char B = 'b' : const char C = 'c' : const char Ç = 'ç' : const char D = 'd' : const char E = 'e' : const char F = 'f' : const char G = 'g' : const char Ğ = 'ğ' : const char H = 'h' : const char I = 'ı' : const char İ = 'i' : const char J = 'j' : const char K = 'k' : const char L = 'l' : const char M = 'm' : const char N = 'n' : const char O = 'o' : const char Ö = 'ö' : const char P = 'p' : const char R = 'r' : const char S = 's' : const char Ş = 'ş' : const char T = 't' : const char U = 'u' : const char Ü = 'ü' : const char V = 'v' : const char Y = 'y' : const char Z = 'z' : const char</pre>
<pre>kelimemodelleri()  isminhaliuret(govde:string, hhhaller: string) basitfiilkipiuret(Kok:string, hhhkipler: string) basiyuklemuret(Kok:string, hhhkipler:string, kkkisiler:string) ekfiilliyuklemuret(fiilKoku:string, kkkip1:string, kkip2:string, kkkisi:string) KisiyeGoreCekimle(fiilGovdesi:string, kisi: Kisiler)  SahisEkiBul(govde:string) EkfiilKipBul(govde:string) fiilKipBul(govde:string) fiilonay(govde:string) Kiekbul(govde:string) olumsuzlukekbul(govde:string) soruekbul(govde:string) cogulekbul(govde:string) isimhalekbul(govde:string) isimiyelikbul(govde:string) KisiEkiDegistir(ek:KisiEkleri)</pre>	<pre>sertyumusak(sertklm: string, kek: string) yumusaksert(yumusakkelime: string, yumusakgovde: string)</pre>

Tablo 6 – Kelime Modelleri ve Alfabe

Yukarıdaki tabloda kelimelerin türünü bildiren kelime modelleri sınıfı cümle çözümü sırasında incelenen bir kelimenin dilbilgisine göre türünü sınıflandırmak için kullanılır.

Aşağıdaki tabloda yer alan sınıfların her biri Türkçe imla kılavuzu dilbilgisi kuralları temel alınarak oluşturulmuştur. Bu sınıflar bir kelimenin ilgili kurala göre işlenmesini sağlarlar.

<code>KaynastirmaHarfi</code>	<code>DuzGenisSesliHarf</code>
<code>kaynas : char</code>	
<code>Bas(govde:GovdeBilgisi)</code> <code>KaynastirmaHarfi(k:char)</code>	<code>Bas(govde:GovdeBilgisi)</code>
<code>Kaynastirma</code>	<code>GenisZamanDarSesliHarf</code>
<code>Y = Alfabe.Y: const char</code> <code>N = Alfabe.N: const char</code> <code>S = Alfabe.S: const char</code> <code>Ş = Alfabe.Ş: const char</code>	
	<code>Bas(govde:GovdeBilgisi)</code>
<code>KiliHarf</code>	<code>Benzesme</code>
	<code>benzes : char</code>
<code>Bas(govde:GovdeBilgisi)</code>	<code>Bas(govde:GovdeBilgisi)</code>
<code>DarSesliHarf</code>	<code>UnluTuremesi</code>
<code>Bas(govde:GovdeBilgisi)</code>	<code>Bas(govde:GovdeBilgisi)</code>
<code>DarSesliHarfIyelik</code>	<code>SabitHarf</code>
	<code>Harf { get; set; }benzes : char</code>
<code>Bas(govde:GovdeBilgisi)</code>	<code>Bas(govde:GovdeBilgisi)</code>

**Tablo 7 – Dilbilgisi Kuralları**

Aşağıdaki tabloda sunulan sınıflar kelimenin gövdesi ile ilgili inceleme yapmak ve bu kelimeyi dilbilgisi kuralları sınıflarına göre değerlendirmek için oluşturulmuştur. Cümlelerin öğelerini bulmak için oluşturulan sınıf her kelime için gövde bilgisi sınıfına başvurur. Sonuçta kelime ile ilgili cümle öğelerini tespit için kullanılacak bilgiler elde edilir.

GovdeBilgisi
<pre> SonSesli { get; set; }: char SondanikiSesli { get; set; }: char SonHarf { get; set; }: char SessizIleBitiyor { get; set; }: bool SertSessizIleBitiyor { get; set; }: bool Sonikises : char[] </pre>
<pre> GovdeBilgisi(ncelenecekkelime: string) SonikiSesliyiVer(kelimesonikisesli: string) SonSesliyiVer(string kelimesonsesli) </pre>
cumleogeleri
<pre> incelenecekcumlekelimedizisi : ArrayList sahislarkokliste : ArrayList sahislarekliste : ArrayList sahislarekturliste : ArrayList fiilkipkokliste : ArrayList fiilkipekliste : ArrayList fiilkipekturliste : ArrayList isminhalikokliste : ArrayList isminhaliekliste : ArrayList isminhaliekturliste : ArrayList </pre>
<pre> Button1_Click(sender:object, e:EventArgs) Button2_Click(sender:object, e:EventArgs) Button3_Click(sender:object, e:EventArgs) Button4_Click(sender:object, e:EventArgs) Button5_Click(sender:object, e:EventArgs) Button6_Click(sender:object, e:EventArgs) Button7_Click(sender:object, e:EventArgs) Button8_Click(sender:object, e:EventArgs) Button9_Click(sender:object, e:EventArgs) Button10_Click(sender:object, e:EventArgs) Button12_1Click(sender:object, e:EventArgs) Button13_Click(sender:object, e:EventArgs) Button14_Click(sender:object, e:EventArgs) Button15_Click(sender:object, e:EventArgs) Button16_Click(sender:object, e:EventArgs) </pre>

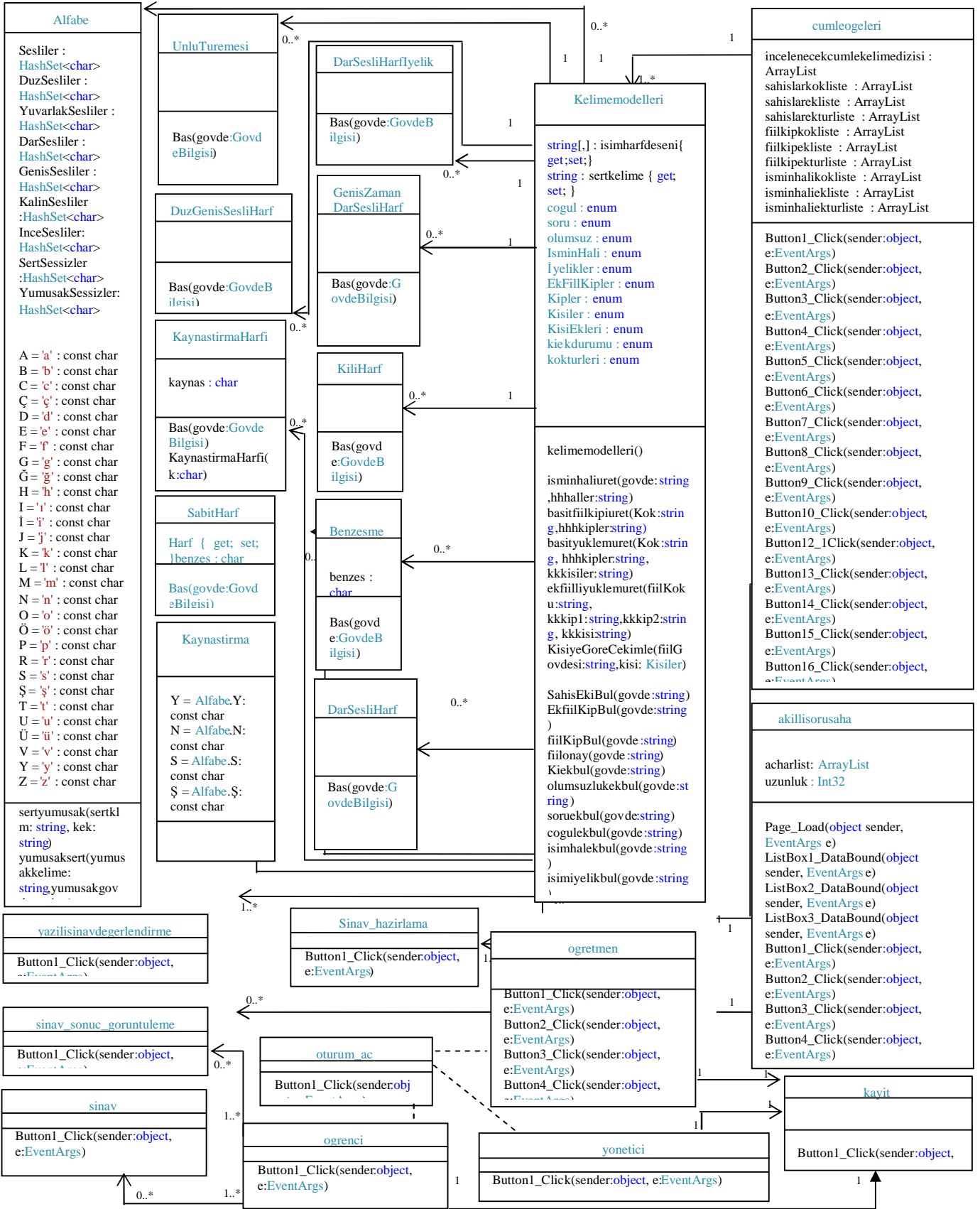
**Tablo 8 – Cümlelerin Öğeleri ve Gövde Bilgisi**

Aşağıda Akıllı e-Soru Sınav Sisteminin sınav bölümü ile ilgili yetkilendirme, kayıt, oturum açma, sınav hazırlama, sınav olma, raporlama işlemleri için kullanılan sınıf yapıları gösterilmiştir.

yazilisinavdegerlendirme	Sinav_hazirlama
Button1_Click(sender:object , e:EventArgs )	Button1_Click(sender:object , e:EventArgs )
kayit	sinav_sonuc_goruntuleme
Button1_Click(sender:object , e:EventArgs )	Button1_Click(sender:object , e:EventArgs )
oturum_ac	yonetici
Button1_Click(sender:object , e:EventArgs )	Button1_Click(sender:object , e:EventArgs )
ogretmen	ogrenci
Button1_Click(sender:object , e:EventArgs ) Button2_Click(sender:object , e:EventArgs ) Button3_Click(sender:object , e:EventArgs ) Button4_Click(sender:object ,	Button1_Click(sender:object , e:EventArgs )

**Tablo 9 – Sınav Bölümü Sınıfları**

Akıllı e-Soru Sınav Sisteminin yazılımı geliştirilirken Türkçe dilbilgisi kurallarına uygun çözümler yapabilmek amacıyla dilbilgisi sınıf yapısı oluşturulmuş ve Türkçe dilbilgisi kurallarının yapılandırıldığı kurallar ile ilgili sınıflar oluşturulmuştur. Geliştirilen uygulamanın sınıf diyagramı bir sonraki sayfada çizilmiştir.



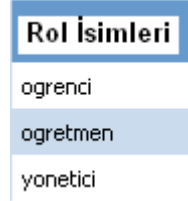
Şekil 22 – Uygulamanın Sınıf Diyagramı

## 5. AKILLI e-SORU SINAV SİSTEMİNİN UYGULAMASI

Bu bölümde uygulamanın bölümleri ve kullanımını anlatılacaktır.

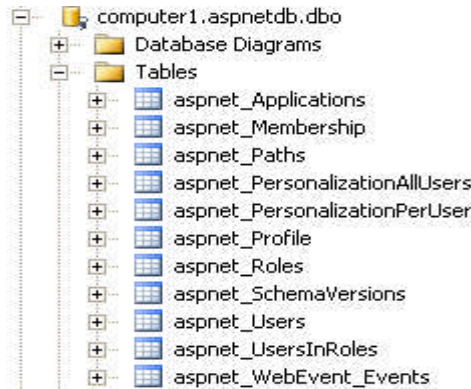
### 5.1. Uygulamaya Erişim ve Yetkilerin Yapılandırılması

Sistemin kullanımına açıldığı 3 tür kullanıcı rolü vardır. Bu roller öğretmen, öğrenci ve yöneticiler için ayrı seviyelerde yetki ve erişim izinlerini sunarlar.



Şekil 23 – Roller

Uygulamayı kullanmak isteyen bir kullanıcının mutlaka bu yetkilerden birini alan ve kimlik tanımlanması onaylanan bir kullanıcı olması şarttır. Kullanıcı mevcut olan hesabıyla yetkilendirilerek ve onaylanarak sisteme giriş yapar. Eğer aday kullanıcı sisteme erişmek istiyorsa kayıt işlemlerini gerçekleştirir. Kayıt işlemleri tamamlandıktan sonra yönetici kullanıcısı eğer uygunsa kullanıcının uygulamayı kullanmasına onay verir. Asp.net platformunda uygulama geliştirirken sistemin kendisi ile ilgili bilgiler ve güvenlik ve üye bilgilerinin tutulabileceği tablolar vardır. Bu tabloların bulunduğu veritabanı framework yapısında mevcuttur.[23] Uygulamada kullanıcı güvenlik ve üye yönetiminin yapılması için aspnetdb veritabanını kullanılmaktadır.



Şekil 24 – Güvenlik ve Üyeler Veritabanı

Yönetici uygulama üzerinden aspnetdb veritabanına ulaşarak işlemleri gerçekleştirir.



## 5.2. Soru Üretilmesi ve Sınav Hazırlanması

Yetkili öğretmen kullanıcısı Soru üretim bölümüne erişerek cümle analizi ve soru üretimi yapabilir.

KÖKLER	EKLER	EK TÜRLERİ
ev	den	Çıkma

olumsuzluk eki bul	fiil kipi bul	şahıs eki bul	soru eki bul	ek fiil kipi bul	liste temizle
cogul eki ara	iyelik ara	hal eki ara	ki eki ara		

**cümle ara (model1)** yüklem > özne > belirteli/belirtisiz nesne > dolaylı tümleş > zarf / edat tümleşci

Nereden/Kimden geldiler ?

Şekil 25 – Kelime Çözümleme Ekranı

Cümledeki kelimeler tek tek çözümlenerek ekleri, ek türleri, kökleri ortaya çıkarılır. Sonra bu eklerden çözümleme yapılır.

**ÜRETİLEN SORULAR**

*Soruya özel işlemler*

**Ders:**  *Aşağıdaki bölümü kullanarak soruyu test sorusuna dönüştürebilirsiniz*

**Konu:**

**Puan:**

**İşlemci nedir?**


Şekil 26 – Soru Hazırlama Ekranı

Öğretmen ders konusu olan metinleri girerek sistemde soru üretilmesini sağlar. Bu sorular öğretmenin elemesinden geçebilir. Ayrıca öğretmen soruların konusunu, dersini ve puanını girerek soruların veritabanında sınıflandırılarak kullanılmasını tesis eder.

Column Name	Data Type	Allow Nulls
soruID	int	<input type="checkbox"/>
ders	varchar(50)	<input type="checkbox"/>
konu	varchar(100)	<input type="checkbox"/>
sorumetni	varchar(MAX)	<input type="checkbox"/>
cevap	nvarchar(MAX)	<input checked="" type="checkbox"/>
puan	int	<input type="checkbox"/>

Şekil 27 – Sınav Soruları Tablo Yapısı

## AKILLI SORU OLUŐTURMA MEKANİZMASI ÜRETİM SAHAŐI



banner  
Akıllı soru üretim mekanizmasına hoşgeldiniz... ı

**tera**

**çevir**

**temizle**

terim sorusu üret

**SORU OLUŐTURULACAK METNİN GİRİLECEĐİ ALAN**

İşlemciye ne yapmasını istediđimizi söyleyen programlar olmadıđı sürece işlemci bir işe yaramaz. Programlar bellekte tutulur, çalıştırılır, bu programların içerdiđi komutlar işlemciye gönderilir. Bilgisayarda tüm programlar sabit diskte (hard disk) tutulur. Sabit diskte tutulan bu programların çalıştırılması aşamasında programlar işlemciye gönderilirler.

**TERİMLER**

Şekil 28 – Terim Sorusu Üretim Ekranı

Öğretmen kullanıcısı eđer ders uygunsa konu ile ilgili bilimsel kavramları soran soru üretebilir.

içer	1	eylem
gerçek	3	isim
Bilgisayar	4	bilişim terimi
bir	6	sıfat
o	6	sıfat
nerede	7	zarf
sabit disk	4	bilişim terimi
Bilgi	3	isim
komut	4	bilişim terimi
süre	4	bilişim terimi

Şekil 29 – Terim Sorusu Üretim Sonuçları Ekranı

### 5.3. Sınavın Aktifleştirilmesi ve Sonuçların İzlenmesi

Öğretmen kullanıcısı ders konusu ile ilgili üretilen soruları veritabanından çekerek sınav hazırlar ve seçtiği öğrenciler için sınavı aktif hâle getirir.

Adres [http://localhost:3021/WebTEST/sinav\\_islemleri/sinav\\_hazirlama.aspx](http://localhost:3021/WebTEST/sinav_islemleri/sinav_hazirlama.aspx)

Sınav hazırlamak için aşağıdaki bilgileri girmelisiniz.

Ders Seç:  
Bilişim Teknolojileri

Konu Seç:  
Anakartlar

Soru Sayısı belirle:

Öğrenci Sayısı belirle:

Sınav Süresi Belirle:

Sınavın açılacağı öğrencileri seç Öğrenci Seç: ogrenci1

Aktifleştir

Şekil 30 – Sınav Hazırlama Ekranı

Aktif sınavı olan öğrenci sınavını belirlenen zaman diliminde alabilir.

Soru3: İşlemci nedir?

ileri geri

sınavı bitir

Şekil 31 – Sınav Ekranı

Öğrenci veya öğretmen sınav sonuçları geçmişini izleyebilir.

Sınav sonuçlarını görüntülemek için öğrencinin ve girdiği sınavın bilgilerini yazarak sorgulayınız.

P

Öğrenci Adı:

Öğrenci Sınav ID:

ogrencisnavID	snavID	snavbaslangiczamani	snavbitiszamani	snavpuani	ogrenciadi
0	0	20.08.2009 00:00:00	20.08.2009 00:00:00	0	abc
1	1	20.08.2009 00:00:00	20.08.2009 00:00:00	1	abc
2	2	20.08.2009 00:00:00	20.08.2009 00:00:00	0	abc
3	3	20.08.2009 00:00:00	20.08.2009 00:00:00	1	abc
4	4	20.08.2009 00:00:00	20.08.2009 00:00:00	0	abc

SqlDataSource - SqlDataSource1

Şekil 32 – Sınav Sonuçları Ekranı

Klasik usulde yapılan sınavları öğretmen öğrenci ve sınavını seçerek değerlendirir.

Klasik usulde uygulanan sınavların değerlendirmesini yapmak için aşağıdan öğrenci adı seçiniz.

Değerlendirme yapacağınız öğrenciyi seçip yazısını puanlayabilirsiniz.

ogrenci1  öğrencinin girdiği sınavları görüntüle Öğrencinin puanlacak sınavını seç:  Databound

SqlDataSource - SqlDataSource2

soruID	sorusirasi	sorumetni	cevap	puan	snavID	ogrencinipuanı	ogrencisnavID
0	0	abc	abc	0	0	0	0
1	1	abc	abc	1	1	1	1
2	2	abc	abc	2	2	2	2
3	3	abc	abc	3	3	3	3
4	4	abc	abc	4	4	4	4

SqlDataSource - SqlDataSource1

Şekil 33 – Klasik Sınav Değerlendirme Ekranı

#### 5.4. Üretilen Örnek Soruların İncelenmesi ve Değerlendirilmesi

Öğretmen kullanıcısı sisteme erişim yaptıktan sonra ders konusunun kaynaklarını metin olarak soru üretim sayfasında sisteme girer. Bu aşamada çok farklı türde soru üretme imkânı oluşturulmuştur. Bunları sıralamak gerekirse:

- Hâl eklerinden faydalanılarak üretilen sorular (bulunma, çıkma, yönelme, vasıta, eşitlik hâlleri için)
- İyelik eki varsa üretilen sorular
- Ek fiil ekinden üretilen sorular
- Boşluk doldurma soruları
- Fiil kipine ve zamanına göre üretilen sorular
- Kavramların bulunması ile üretilen kavram soruları
- Sıfat veya zarf köklerinin tespiti ile oluşturulan sorular
- Doğru yanlış soruları

Soru türleri olarak sıralanabilir.

Soru üretimi sırasında üretilen yukarıdaki türdeki sorulardan bazıları anlamsız olabilmektedir. Anlamsız sorular cümlenin kelimelerinin farklı ek bulundurma ihtimalide dikkate alındığı için oluşmaktadır. Bu tip sorular öğretmen tarafından elenerek değerlendirme dışı tutulmalıdır. Ayrıca öğretmene beğenmediği soruları çıkartma seçeneği de sunulmuştur. Örnek çalışmaların ekran görüntüleri aşağıda sunulmuştur.

Girilen metnin her bir cümlesi cümle bölümüne gönderilir. Sırayla soru türlerine göre çıkabilecek olan sorular üretilir. Metinden cümle bazında soru üretme işlemi cümlenin kelimeleri çözümlenerek ve eklere göre soru türleri değerlendirilerek ortaya çıkartılır.

KÖKLER	BAZILAR	BAZILAR	ÜRETİLEN SORULAR
kaynakt gele kaynak kaynak kaynak kaynak kaynak kaynak gel gel	n n tan tan ta a ta a ta a e e	kiniciTekil kiniciTekil Çıkma Çıkma Bulunma Bulunma Bulunma Yonelme Yonelme Yonelme Gosteme Gosteme	nerede/kimde gelen bilgi işlenir ? nereden/kimden gelen bilgi işlenir ? nereseykime ali aysşe geldi ? nereseykime ali geldi ? nereseykime Bir ağ tasanımı yaparken ağın hızı, maliyeti ve nereseykime Bir ağ tasanımı yaparken ağın hızı, maliyeti ve nereseykime Bir ağ tasanımı yaparken ağın hızı, maliyeti ve nereseykime kaynaktan bilgi işlenir ? neyükimi babası Ali'dir. ? neyükimi bir donanım birimidir ? neyükimi kalemi mavidir ? neyükimi rengi kırmızıdır ? neyükimin babası Ali'dir. ? neyükimin kalemi mavidir ? neyükimin rengi kırmızıdır ? ali aysşe eve neyükimi ? ali eve neyükimi ? Bir ağ neyükimi yaparken ağın hızı, maliyeti ve kablolama Bir ağ tasanımı yaparken neyükimi hızı, maliyeti ve kablo Bir ağ tasanımı yaparken neyükimin hızı, maliyeti Bir ağ tasanımı yaparken ağın hızı, neyükimi ve kablolama Bir ağ tasanımı yaparken ağın hızı, maliyeti ve kablolama Bir ağ tasanımı yaparken ağın hızı, maliyeti ve kablolama işlemci bir neyükimi birimidir ?
<p>Soruyu özel işlemler</p> <p>Ders:</p> <p>Konu:</p> <p>Özellik:</p> <p>Soru at</p> <p>timunu sil</p> <p>yazılı sorusu ekle</p> <p>boşluk sorusu ekle</p>			<p>Doğru soru tipini seçiniz</p> <p>Grup1</p> <p>Grup2</p> <p>Grup3</p> <p>Grup4</p> <p>Doğru cevap</p> <p>Cevap1</p> <p>test sorusu ekle</p>
<p>Kelime / Kelime Grup</p> <p>Özellik</p>			<p>model1 isim isim fiil soru oluştur</p> <p>model2 fiil soru oluştur</p>

Şekil 34 – Örnek Soru Üretimi Ekranı

Üretilen sorular öğretmen tarafından gözden geçirilir. İstenmeyen veya anlamsız bulunan sorular elenir. Seçilen sorular istenilen soru tipi biçimine dönüştürülerek akıllı soruların bilgilerini depolayan veritabanına gönderilir. Beş farklı tipte soru biçimi hazırlanabilir. Hazırlanabilecek soru tipleri şunlardır:

- Klasik usul
- Test soruları
- Boşluk doldurma
- Doğru yanlış soruları
- Terim Soruları

Soru tipleri ile ilgili çalışma aşağıda verilmiştir. Sayfada üretilen bir sorunun soru tipi olarak biçimlendirilmesi test, yazılı usul, boşluk doldurma veya doğru yanlış biçimleri olabilmektedir.

*Soruya özel işlemler*

*Ders:*

*Konu:*

*Puan:*

soru at

tümünü sil

yazılı sorusu ekle

boşluk sorusu ekle

*Aşağıdaki bölümlü  
kullanarak soruyu test  
sorusuna  
dönüştürebilirsiniz*

*Cevap1*

*Cevap2*

*Cevap3*

*Cevap4*

*Doğru cevap*

Cevap1 ▾

test sorusu ekle

### ÜRETİLEN SORULAR

kaynaktan gelen..... işlenir  
..... gelen bilgi işlenir  
kaynaktan..... bilgi işlenir  
metin kutusu bir..... kutusu bileşenidir.  
..... kutusu bir araç kutusu bileşenidir.  
metin kutusu bir araç kutusu.....

Şekil 35 – Örnek Soru Tipleri Ekran



## 6. SONUÇ

Yapılan bu tez çalışmayla bir akıllı e-soru sınav sisteminin tasarlanması ve uygulaması gerçekleştirilmiştir. Ders konusunu içeren metinlerin çözümlenmesiyle dinamik olarak üretilen sorular öğretmen denetiminden geçtikten sonra sınav sorularına dönüştürülmüştür. Sunulan bu yaklaşımın eğitime ve akademik hayata katkısı ile üretilen çok sayıda zengin içerikli sorularla kolay hazırlanabilen, adil, hızlı sonuç üreten değerlendirmeler yapılacaktır.

Akıllı soru sorma mekanizması ile planlanan hedeflere ulaşılmıştır. Sistem öğrencinin eksik ve zayıf olduğu konuları tespit edip, öğrencinin çalışması gereken yerleri bilerek hız kazanmasına olanak sağlayacak özelliklere sahiptir. Öğrenci kendi değerlendirmesini sınav sonuçlarını izleyerek yapabilir. Böylece öğrencinin eksik olduğu modülleri fark edeceği düşünülmüştür. Konu ile ilgili zengin içerikli bilgiye dayalı soruların sorulmasına imkân tanınmıştır. Web üzerinden soru, sınav hazırlama ve uygulama özelliklerinin yanında öğrencinin sınav sonuçlarının ve geçmiş sınav raporlarının öğrencinin kendisi ve öğretmen tarafından izlenebileceği sistemin Web tabanlı olması uzaktan eğitim aracı olarak kullanılabilirliğini sağlamıştır. Çalışma koşulların zaman mekân gibi kısıtlamaları aşması öğrenciye sağlanan faydalardandır. Diğer soru çeşitlerinin yanında klasik soru üretecek uygulama geliştirilmiştir. Yapılan araştırmalar sonucu sistemin öğrenciye kolaylık sağlayan, hız kazandıran yapısıyla eğitim sisteminin pedagojik yaklaşımlarına uygun olduğu saptanmıştır.

Soru üretimi için kullanılan girdiler, ilgili derse ait konunun anlatıldığı metin tabanlı bilgisayar ortamına aktarılmış kaynaklardır. Ortaya çıkartılacak soruların sorulabilecek nitelikte olup olmadığı öğretmen tarafından üretimden sonra elenerek belirlenmektedir. Anlamsız, basit veya gereksiz görülen soruların öğretmen tarafından elenir. Öğretmen istediği soruları seçerek veritabanında depolar. Uygulamanın yapılandırıldığı sistem tümüyle Web tabanlı çalışır. Hem soruların üretilmesi hem de sınav uygulaması ile ilgili işlemler Web üzerinden yapılmaktadır. Sınav hazırlama işlemleri öğretmen tarafından yapılarak ders, konu soru tipi seçilecek, istenen öğrenciler için sınav aktif hâle getirilir. Sınavı aktif olan öğrenci verilen süre içinde sınavını tamamlamalıdır. Sınav bitiminde eğer klasik ise öğretmen, test ise sistem otomatik olarak sınav değerlendirmesi yapacaktır. Sınav sonuçları öğrenci veya öğretmen tarafından sınavdan sonraki zamanlarda geçmişe doğru izlenebilir yapıdadır. Öğrencinin sınav geçmişini öğrenci veya öğretmen izleyerek zayıf olduğu konular tespit edebilir.

Yapılan çalışmada cümlelerin öğelerine ayrılması ve öğelerin bulunması için kelime türlerinin dilbilgisi kurallarına göre modellendiği kelime veri modelleri ile gerçekleştirilir. Bu yapıların kullanıldığı cümlelerin çözümlendiği algoritma ile kelime ve cümle ile ilgili bilgiler elde edilmektedir. Elde edilen bu bilgiler ışığında sorular üretilmektedir.

Uygulama girilen metin tabanlı bilgilerin işlenerek doğal dil çözümleme yöntemlerine başvurulması sonucu soru üretmektedir. Metinler önce cümle seviyesinde ele alınır, kelime tarama aşamasında ise oluşturulan veri modelleri rehberliğinde uyuşan kelimelerin kök, kelime türü, aldığı ekler ve türleri gibi bilgileri elde edilir. Örneğin bir kelimenin basit çekimli fiil türünde olduğu varsayılırsa bu kelime için soru eki, şahıs eki, zaman şekil eki, olumsuzluk eki, ek fiil eki olmak üzere tüm olasılıkları sınanmaktadır. Bu bilgiler ışığında çok sayıda, farklı kombinasyonlarda soru tipleri seçilerek üretilir. Özellikle cümlenin öğelerini bulmaya yönelik sorular hazırlanmaktadır. Bilgi kaynağını oluşturan ders konusu metinlerinin zengin içerikli, düzgün hazırlanmış metinler olması ile çıktılarının işe yarar olmasının doğru orantılı olduğu görülmüştür.

Sistemin soru üretim kısmı uyarlanarak farklı uygulamaları oluşturan yazılımlar geliştirilebilir. Örneğin Türkçe dilbilgisi öğretimi için öğrenciye daha kolay ve eğlenceli çalışmayı sağlayabilecek bir araca dönüştürülebilir. Bir öğretmenle karşılaştırıldığında, uygulamada şimdiye kadar yapılan geliştirmeler her zaman ilgili dersin öğretmeni kadar nitelikli sorular soramamıştır. Buna rağmen öğretmenin işini kolaylaştıran yardımcı bir araç olarak değerlendirilebilir. Cümle analizi yapıldığı çatı bir motor işlevi görerek dilbilgisi kurallarını kullanarak cümle ile ilgili yakaladığı tüm bilgileri veritabanına göndermektedir. Bu özelliğinden dolayı cümleden anlam çıkarma, çeviri sistemleri, yazıdan sese dönüşüm yazılımları gibi doğal dil işleme projelerinde bir iskelet yapı olarak esas alınabileceği düşünülmektedir.

Uygulama geliştirmelere açıktır. Yapay zekâ uygulamaları, doğal dil işleme süreçleri, dili bilgisayar destekli anlamaya yönelik çalışmalar geliştikçe uygulamanın yeni versiyonlarında soru sorma kabiliyetinin daha nitelikli olacağı tasarımlar geliştirilebilir. Böylece ileriki çalışmalarda özellikle bir öğretmenin sorabileceği sorulara yakın sorular üretilebileceği öngörülmektedir.

## Kaynakça

### Kitap

- [1] Sarıdoğan, M.Erhan, (2004), Yazılım Mühendisliği, Papatya Yayıncılık Eğitim, İstanbul
- [2] Pressman, R.S., (2005), Software Engineering: A Practitioner's Approach, The McGraw-Hill, London
- [3] Kemal Oflazer, H. Cem Bozşahin, Turkish Natural Language Processing Initiative, An Overview in Proc. Of the 3. Turkish Symposium, on AI&NN, Ankara, 1994
- [5] Ünal Çakıroğlu, Türkçe Aritmetik Problemlerin Bilgisayarla Çözümü, Elektrik-Elektronik-Bilgisayar Mühendisliği Sempozyumu Ve Fuarı (Eleco 2006), Bursa, 2006
- [7] Muhammed Oguzhan Külekci , An Intelligent Diagnostic System From The Clinical Narratives In Turkish, B.S. in Computer Engineering, Bogaziçi University, Yüksek Lisans Tezi, 1998
- [10] Haluk GÜMÜŞKAYA - Ömer BOYACI, (2003), Java Ağ Programcılığı, Alfa Basım Yayın, İstanbul
- [14] Rifat Çölkesen, Turhan Karagüler, Gökhan Silahtaroglu, Uzaktan Eğitimde Akıllı Soru Sorma Mekanizması, Akademik Bilişim 2001, Samsun, 1-2 Şubat 2001
- [16] Rifat Çölkesen, (2003), Bilgisayar Haberleşmesi ve Ağ Teknolojileri, Papatya Yayıncılık Eğitim, İstanbul
- [17] Mike Snell, Glenn Johnson, Tony Northrup, (2009), Microsoft .NET Framework 3.5 - ASP.NET, Microsoft Pres, Redmond, Washington
- [18] Niyazi Karasar, (1998), Bilimsel Araştırma Yöntemi, Nobel Yayın Dağıtım, Ankara
- [21] Vasif V. Nabyev, (2005), Yapay Zeka, Seçkin Yayıncılık, Ankara
- [22] Mehmet Hengirmen, (1997) Türkçe Dilbilgisi, Engin Yayınevi, Ankara
- [23] Türk Dil Kurumu, (1992), Türkçe Sözlük, Milliyet Tesisleri, İstanbul
- [24] Rifat Çölkesen, (2007), Veri Yapıları ve Algoritmalar, Papatya Yayıncılık Eğitim, İstanbul
- [25] Donis Marshall, (2008), Programming Microsoft Visual C# 2008: The Language, Microsoft Pres, Redmond, Washington

### İnternet Kaynakçası

- [4] <http://web.media.mit.edu/~minsky/papers/ComputersCantThink.txt>, erişim tarihi, 21.03.2009
- [6] [turkoloji.cu.edu.tr/DILBILIM/zeynep\\_altan\\_dil\\_modelleme\\_belirsizlik.pdf](http://turkoloji.cu.edu.tr/DILBILIM/zeynep_altan_dil_modelleme_belirsizlik.pdf), erişim tarihi, 03.11.2008
- [8] <http://msdn.microsoft.com/en-us/asp.net/aa336598.aspx>, erişim tarihi, 04.09.2009
- [9] <http://msdn.microsoft.com/en-us/library/cc645993.aspx>, erişim tarihi, 04.09.2009
- [11] <http://msdn.microsoft.com/en-us/library/4c26cc39.aspx>, erişim tarihi, 10.09.2009
- [12] <http://msdn.microsoft.com/en-us/library/ms143506.aspx#EE32>, erişim tarihi,10.09.2009
- [13] <http://www.microsoft.com/visualstudio/products/teamsystem/default.aspx>, erişim tarihi, 10.09.2009
- [19] <http://www.tdk.gov.tr>, erişim tarihi, 16.01.2009
- [20] <http://www.turkceciler.com>, erişim tarihi,24.01.2009