

**SCORM UYUMLU GENİŞLETİLEBİLİR WEB TABANLI  
UZAKTAN EĞİTİM YÖNETİM SİSTEMİNİN  
TASARLANMASI VE GERÇEKLEŞTİRİLMESİ**

**YÜKSEK LİSANS TEZİ**

**Emin İBİLİ**

**DANIŞMAN  
Prof. Dr. Mevlüt DOĞAN**

**BİLGİSAYAR ANA BİLİM DALI**

**TEMMUZ 2009**

Bu tez çalışması 08-FENED-01 numaralı proje ile BAPK tarafından desteklenmiştir.

**AFYON KOCATEPE ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**YÜKSEK LİSANS TEZİ**

**SCORM UYUMLU GENİŞLETİLEBİLİR WEB**  
**TABANLI UZAKTAN EĞİTİM YÖNETİM**  
**SİSTEMİNİN TASARLANMASI VE**  
**GERÇEKLEŞTİRİLMESİ**

**EMİN İBİLİ**

**DANIŞMAN**

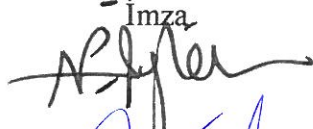


**Prof. Dr. Mevlüt DOĞAN**

**BİLGİSAYAR ANABİLİM DALI**

**Temmuz 2009**

## ONAY SAYFASI

Prof. Dr. Mevlüt DOĞAN danışmanlığında, Emin İBİLİ tarafından hazırlanan “Scorm Uyumlu Genişletilebilir Web Tabanlı Uzaktan Eğitim Yönetim Sisteminin Tasarlanması ve Gerçekleştirilmesi” başlıklı bu çalışma, lisansüstü eğitim ve öğretim yönetmeliğinin ilgili maddeleri uyarınca 27/07/2009 tarihinde aşağıdaki jüri tarafından Bilgisayar Anabilim Dalında Yüksek Lisans tezi olarak oybirliği/oy çokluğu ile kabul edilmiştir.

	Ünvanı, Adı, SOYADI	İmza
Başkan	Prof.Dr. Necdet SAĞLAM	
Üye	Prof.Dr. Süleyman Taşgetiren	
Üye(Danışman)	Prof.Dr. Mevlüt DOĞAN	

Afyon Kocatepe Üniversitesi  
Fen Bilimleri Enstitüsü Yönetim Kurulu'nun  
.../.../...tarih ve  
... sayılı kararıyla onaylanmıştır.

Doç. Dr. Rıdvan ÜNAL

Enstitü Müdürü

## ÖZET

Yüksek Lisans Tezi

SCORM Uyumlu Genişletilebilir Web Tabanlı Uzaktan Eğitim Yönetim Sisteminin  
Tasarlanması ve Gerçekleştirilmesi  
Emin İBİLİ

**Afyon Kocatepe Üniversitesi**  
**Fen Bilimleri Enstitüsü**  
Bilgisayar Anabilim Dalı

**Danışman:** Prof. Dr. Mevlüt DOĞAN

Bilgi teknolojilerindeki önemli gelişmeler ile birlikte hızla artan bilgi yoğunluğu doğru bilgiyi bulma, bu bilgiyi yönetme gibi zorlukları da beraberinde getirmiştir. Bu zorluklarla birlikte hızla artan nüfus oranı ve bu nüfusa yetişemeyen okullaşma ve öğretmen ihtiyacı toplumları geleneksel eğitim anlayışından farklı alternatifler bulmaya yöneltmiştir. İnternet ve Web Teknolojilerinde gelişmeler sayesinde bilgi zaman ve mekân sınırı tanımadan kolayca paylaşılabilir, yönetilebilir ve değerlendirilebilir hale gelmiştir. Web teknolojilerinin eğitim alanında kullanılması ile birlikte Web Tabanlı Öğrenim Yönetim Sistemleri ortaya çıkmıştır. Bu öğrenim yönetim sistemlerini dayanıklı, yeniden kullanılabilir, diğer yazılımlarla birlikte çalışabilir, ulaşılabilir, taşınabilir, olmasını sağlamak için SCORM standartları geliştirilmiştir. Bu doğrultuda eğitim sistemine katkıda bulunmak amacı ile web tabanlı modüler bir uzaktan öğrenim yönetim sistemi tasarlanmıştır. Yüksek öğretimde kullanılmak üzere geliştirilen bu sistem 36 modülden oluşmuş ve SCORM standartlarına uygun olarak tasarlanmıştır. Bu çalışmada SCORM uyumlu bir uzaktan eğitim sisteminin nasıl hazırlanması gerektiği hakkında bilgiler verilmiş, analiz raporları hazırlanmış ve hazırlanan modüller ayrıntılı olarak anlatılmıştır.

2009, 311 sayfa

**Anahtar Kelimeler:** Uzaktan eğitim, SCORM, Öğrenim Yönetim Sistemi, Web Tabanlı Eğitim, e-eğitim, LMS

## **ABSTRACT**

M. Sc. Thesis

Designing and Application of SCORM Consistent and Web Based Distance Learning  
Management System Which Can Be Expanded

Emin İBİLİ

**Afyon Kocatepe University**  
**Graduate School of Natural and Applied Sciences**  
Department of Computer

**Supervisor:** Prof. Dr. Mevlüt DOGAN

Important developments in information technologies and increasing information consistency, bring forward some difficulties like finding out the right information and managing this information. These difficulties as well as increasing population and limited schools and teachers focused the societies on new alternatives. Thanks to the developments in web technologies information can be shared without limits of time or location, and also it has become manageable and evalunble. With the use of web technologies in education, web based teaching and managing systems have come forward. These teaching systems are durable and reusable, they can be used with other softwares, and they're attainable and portable. In order to reach these standarts SCORM systems have been developed. In this way, so as to support education, a web based modular distance education managing system has been designed. The system developed for high education consists of modular structure and it was developed in accordance with SCORM standarts. In this study, some information was given on the preporation of SCORM consistent web based learning management system, analysis reports were prepared and the modules were explained in detail.

2009, 311 pages

**Key Words:** Distant education, SCORM, Learning Management System, Web based education, E-learning, LMS

## TEŐEKKÜR

Yüksek lisans süresi boyunca, yardımını, ilgisini, anlayışını ve bana olan güvenini eksik etmeyen danışmanım Prof. Dr. Mevlüt DOĞANA'a, manevi desteęi ile çalışma azmime her daim ivme kazandıran değerli hocam Prof. Dr. Necdet SAĞLAM'a, çalışmamın her aşamasında bana destek olan Öğr. Gör. Mahmut KANTAR'a, LMS'nin hazırlanmasında yardımcı olan arkadaşım Fatih BAYRAM'a, İçerikleri hazırlayan Fidan HAKKARİ'ye uzaktan eğitim konusunda gerekli eğitimleri almamızı sağlayan, tecrübelerini esirgemeyen aynı zamanda analiz raporlarının hazırlanmasında büyük emeęi gecen BilgeAdam firmasına, SCORM API nin geliştirilmesi ve LMS e uyarlanması konusunda yardımcı olan Advancity firmasına, çalışmalarım boyunca bana yol gösteren tüm hocalarıma, bugüne kadar her konuda beni destekleyen aileme, ayrıca “*e-fizik uygulaması için alt yapının oluşturulması ve uygulanması*” adlı ve 08-FENED-01 nolu proje paralelinde bu tez projemi destekleyen AKÜ, BAKB'ye teşekkür ederim.

Emin İBİLİ

Afyonkarahisar, 2009

## İÇİNDEKİLER

<b>ÖZET</b>	<b>i</b>
<b>ABSTRACT</b>	<b>ii</b>
<b>TEŞEKKÜR</b>	<b>iii</b>
<b>İÇİNDEKİLER</b>	<b>iv</b>
<b>ŞEKİLLER DİZİNİ</b>	<b>ix</b>
<b>TABLolar DİZİNİ</b>	<b>xiii</b>
<b>SİMGELER VE KISALTMALAR DİZİNİ</b>	<b>xv</b>
<b>1. GİRİŞ</b>	<b>1</b>
<b>2. GENEL KAVRAMLAR</b>	<b>5</b>
2.1 Eğitim	5
2.2 Uzaktan Eğitim	7
2.2.1 Uzaktan Eğitimin Avantajları	9
2.2.2 Uzaktan Eğitimin Dezavantajları	10
2.3 Uzaktan Eğitimin Tasarlanması Ve Geliştirilmesi Süreci	11
2.4 SCORM (Sharable Courseware Object Reference Model)	12
2.5 Modüler Yapı	14
<b>3. SCORM</b>	<b>15</b>
3.1 İçerik Kümesi Modeli (Content Aggregation Model-CAM)	15
3.1.1 İçerik Modeli (Content Model)	15
3.1.2 İçerik Paketi	16
3.1.3 Tanımlama XML Dosyası (Manifest)	16
3.1.4 Üst-veri (Meta-data)	16
3.2 Çalışma Ortamı (Runtime Environment-RTE)	17
3.3 Sıralama ve Dolaşım (Sequencing and Navigation – SN)	18
3.4 SCORM 1.2 API Özellikleri	18

3.4.1	SCORM 1.2 API Veri Tabanı Yapısı	20
3.5	SCORM 2004 API'nın Özellikleri	26
3.5.1	SCORM 2004 API Çalışma Şekli	26
3.5.2	SCORM_2004 VERİTABANI TASARIMI	28
3.5.3	SCORM 2004' de Veri Tabanı Tabloları	28
3.5.3.1	SCORM_APPLICATIONDATA Tablosu	29
3.5.3.2	SCORM_COURSEINFO Tablosu	29
3.5.3.3	SCORM_COURSESTATUS Tablosu	30
3.5.3.4	SCORM_ITEMINFO Tablosu	31
3.5.3.5	SCORM_SCOCOMMENTS Tablosu	33
3.5.3.6	SCORM_USERCOURSEINFO Tablosu	34
3.5.4	SCORM 2004'TE OBJECT TABLOLARI	35
3.5.4.1	SCORM_OBJ_ADVSCORM_2004_DMELEMENT Tablosu	35
3.5.4.2	SCORM_OBJ_ADVDMELEMENTDESCRIPTOR Tablosu	37
3.5.4.3	SCORM_OBJ_ADLTOC tablosu	39
3.5.4.4	SCORM_OBJ_ADLVALIDREQUESTS Tablosu	40
3.5.4.5	SCORM_OBJ_ADVADLTRACKING Tablosu	41
3.5.4.6	SCORM_OBJ_ADVSEQACTIVITYTREE Tablosu	42
3.5.4.7	SCORM_OBJ_ADVSEQOBJECTIVETRACKING Tablosu	43
3.5.4.8	SCORM_OBJ_ADVVERSION Tablosu	44
3.5.4.9	SCORM_OBJ_SEQOBJECTIVE Tablosu	45
3.5.4.10	SCORM_OBJ_SEQOBJECTIVEMAP Tablosu	46
3.5.4.11	SCORM_OBJ_SEQACTIVITY Tablosu	47
3.5.6	SCORM Uyumlu İçerik Geliştirilmesi	59
3.5.6.1	Macromedia Flash ile SCORM Uyumlu Örnek İçerik Geliştirilmesi	59
3.5.6.2	Skor ve Sayfa Bilgilerinin Tutulması	63
3.5.6.3	Bilgilerin Gönderilmesi	64
<b>4.</b>	<b>HAZIRLANAN LMS'İN TEKNİK YAPISI</b>	<b>65</b>
4.1	Hazırlanan LMS de Kullanılan Teknolojiler.	65



4.2	Mimari Tasarımı Oluşturan Katmanlar	65
4.2.1	Data Access Layer Katmanı	66
4.2.2	Data Layer Katmanı	67
4.2.3	Entity Layer Katmanı	67
4.2.4	Business Layer Katmanı	67
4.2.5	Presentation Layer Katmanı	68
4.3	SQL Server 2005	68
4.4	Visual Studio .NET 2005	72
4.5	Ajax	74
4.5.1	Ajax Kullanımının Avantajları	75
4.5.2	Ajax Kullanımının Dezavantajları	76
4.6	Uzaktan Öğrenim Yönetim Sistemleri	78
4.7	Modüler Sistem	79
4.8	SCORM Yapısının LMS'ye Uyarlanması	79
4.8.1	SCORM 1.2 API nin LMS'ye Uyarlanması	79
4.8.2	SCORM 2004 API'nin LMS'ye Uyarlanması	85
4.9	Hazırlanan Eğitim Yönetim Sisteminin Temel Modülleri	88
<b>5.</b>	<b>SCORM UYUMLU MODÜLER LMS'İN TASARLANMASI VE GERÇEKLEŞTİRİLMESİ</b>	<b>93</b>
5.1	Online Başvuru Modülü	94
5.1.1	Online Başvuru Ekranları	94
5.1.1.1	Kişisel Bilgiler Ekranı	95
5.1.1.2	İletişim Bilgileri Ekranı	98
5.1.1.3	İş Bilgileri Ekranı	100
5.1.1.4	Eğitim Bilgileri Ekranı	103
5.1.1.5	Sınav Bilgileri Ekranı	107
5.1.1.6	Başvurulacak Program Seçimi	109
5.1.2	Başvurusu Onaylanan Öğrencilere E-posta Gönderimi	111
5.1.3	Bölüm Sekreteri Evrak Onay	113

5.1.4	Bölüm Başkanı, Mülakat Notu Girişi	114
5.1.5	Öğrenci İşleri Onay	115
5.2	Kullanıcı ve Kullanıcı Gurupları Modülü	121
5.2.1	Personel Girişi	121
5.2.1.1	Kişisel Bilgiler Ekranı	121
5.2.1.2	İletişim Bilgileri	124
5.2.2	Kayıdı Yapılan Öğretim Üyesini Bölüme Atama	127
5.3	Rol ve Yetkiler Modülü	131
5.3.1	Rol Tabanlı Güvenlik	131
5.3.1.1	Projede Rol Tabanlı Güvenlik Oluşturulması	134
5.3.1.2	Stored Procedure'nin Anlamı	137
5.3.2	Rol Tabanlı Sistemin Dinamik Hale Getirilmesi	146
5.3.2.1	Klasör Yetkileri	146
5.3.2.2	Web Sitesi Erişim Kuralları	147
5.3.2.3	Rollerin Belirlenmesi	147
5.3.2.4	Kullanıcı Profilleri	148
5.3.2.5	Aktif/Pasif Kullanıcılar	149
5.3.2.6	Kullanıcı Ekleme	150
5.3.2.7	Kullanıcı Bilgilerini Düzenleme ve Onaylama	152
5.3.3	Aynı Role Sahip Kullanıcılar Arasındaki Yetkilendirme	154
5.3.3.1	Uygulama	156
5.4	Ünvan Tanımlama Modülü	163
5.5	Okul Tanımlama Modülü	166
5.6	Bölüm Tanımlama Modülü	172
5.7	Eğitim Kataloğu Modülü	176
5.8	Dersler Modülü	188
5.8.1	Sisteme Yeni Ders Ekleme	188
5.8.2	Dersin İçerik Ağacının Oluşturulması	191
5.8.3	Dersin Sınavlarını belirlemek	192
5.8.4	Ödev eklemek	193

5.8.5	Ders Kaynak Bilgilerini Ekleme	194
5.9	Haftalık Ders Programı Modülü	197
5.10	Şube İşlemleri Modülü	204
5.11	Danışman Atama İşlemleri Modülü	208
5.12	Ders Kayıt ve Onay İşlemleri Modülü	211
5.12.1	Ders Kayıt İşlemleri	211
5.12.2	Ders Kayıt	213
5.12.3	Danışman Ders Kayıt Onayı	214
5.13	Soru Tanımlama Modülü	218
5.14	Sınav Modülü	222
5.15	Deneme Sınavı Modülü	225
5.16	Ödevler Modülü	227
5.16.1	Öğretim Elemanı Ödev Gönderme İşlemi	227
5.16.2	Öğrenci Ödev Gönderme İşlemi	230
5.16.3	Ödev Değerlendirme İşlemi	231
5.17	Öğretim Elemanı Performans Takip Modülü	234
5.18	Öğrenci Performans Takip Modülü	235
5.19	Öğrenci Takip Raporu	238
<b>6.</b>	<b>SONUÇ</b>	<b>241</b>
<b>7.</b>	<b>KAYNAKÇA</b>	<b>244</b>
<b>8.</b>	<b>EK-1</b>	<b>247</b>
<b>9.</b>	<b>EK-2</b>	<b>261</b>
<b>10.</b>	<b>ÖZGEÇMİŞ</b>	<b>310</b>

## ŞEKİLLER DİZİNİ

	<b>Sayfa No</b>
Şekil 2. 1 Eğitimsel daire modeli (Codone 2001).	6
Şekil 2. 2 SCORM sürümlerindeki gelişim evreleri.	13
Şekil 3. 1 Öğrenim Yönetim Sistemleri ve API arasındaki iletişim.	17
Şekil 3. 2 SCORM 1.2 API' nin işleyişi.	18
Şekil 3. 3 API ve İçerik İletişimi	20
Şekil 3. 4 SCORM 2004 API' nin işleyişi.	26
Şekil 3. 5 Macromedia da Flash içeriğin publish edilerek SCORM uyumlu hale getirilmesi.	61
Şekil 4. 1 Mimari tasarımı oluşturan katmanlar.	66
Şekil 4. 2 SCORM 1.2 tabloları.	80
Şekil 4. 3 Programa yeni ders ekleme.	81
Şekil 4. 4 Sisteme SCORM 1.2 uyumlu içerik paketi ekleme.	82
Şekil 4. 5 SCORM1_2_IMP_PaketBilgisi tablosu.	83
Şekil 4. 6 SCORM 1.2 dosyaları.	84
Şekil 4. 7 wwwroot dizini.	86
Şekil 4. 8 SCORM 2004 dosyalarının sanal dizin içinde çalıştırılması.	86
Şekil 4. 9 Alias isminin ve Physical path'in belirlenmesi.	87
Şekil 4. 10 AkuElearning Projesi içinde SCORM_2004 projesinin çalıştırılması.	88
Şekil 5. 1 Uyruk Türkiye değilse kişisel bilgiler ekranında doldurulacak alanlar.	97
Şekil 5. 2 Uyruk Türkiye ise doldurulacak bilgiler.	98
Şekil 5. 3 Bulunduğu ülke Türkiye ise doldurulacak alanlar.	99
Şekil 5. 4 Bulunduğu ülke Türkiye değil ise doldurulacak alanlar.	100
Şekil 5. 5 İş Bilgileri ekranındaki ilk soru.	101
Şekil 5. 6 Aday çalışıyorsa ve ülke Türkiye ise doldurması gereken bilgiler.	102
Şekil 5. 7 Aday çalışıyorsa ve ülke Türkiye değilse doldurması gereken bilgiler.	103
Şekil 5. 8 Ülke Türkiye ise eğitim bilgileri ekranındaki doldurulması gereken alanlar.	104
Şekil 5. 9 Eğitim bilgileri ekranı, seçilen ülke türkiye değilse doldurulacak alanlar.	105

Şekil 5. 10 Excel dönüşümleri sayfası.	107
Şekil 5. 11 Sınav bilgileri ekranı.	108
Şekil 5. 12 Başvurulacak programın seçimi.	109
Şekil 5. 13 Rapor sayfası.	110
Şekil 5. 14 Bölüm sekreteri giriş ekranı.	111
Şekil 5. 15 Mülakat bilgileri E-posta sayfası.	112
Şekil 5. 16 Gönderilen mülakat bilgilerinin E-posta adresine ulaşması.	113
Şekil 5. 17 Evrak onay sayfası.	114
Şekil 5. 18 Mülakat notu girişi sayfası.	114
Şekil 5. 19 Öğrenci işleri onay sayfası.	115
Şekil 5. 20 Peronel TC vatandaşı, erkek ve askerliğini ertelemişse doldurulacak alanlar.	125
Şekil 5. 21 Eklelen peronel TC vatandaşı değilse doldurulacak alanlar.	126
Şekil 5. 22 Öğretim üyelerinin bölümlere atanması.	127
Şekil 5. 23 Uzaktan öğrenim yönetim sisteminde oluşturulan önemli roller.	132
Şekil 5. 24 Online başvuru modülündeki klasörler.	133
Şekil 5. 25 aspnet_regsql.exe' nin çalıştırılması.	134
Şekil 5. 26 Aspnet Procedure' lerinin ve tablolarının Oluşturulması.	135
Şekil 5. 27 Database seçimi.	136
Şekil 5. 28 Aspnet procedure tablosu	136
Şekil 5. 29 Saklı yordamların faydaları.	137
Şekil 5. 30 SQL Server erişim bilgilerinin girilmesi.	138
Şekil 5. 31 Peronel TC vatandaşı, erkek ve askerliği ertelemişse doldurulacak alanlar.	139
Şekil 5. 32 Login kontrolünün sayfaya eklenmesi.	139
Şekil 5. 33 Administration Tool ekranı.	140
Şekil 5. 34 Application setting ekranı.	141
Şekil 5. 35 Erişim metodu.	142
Şekil 5. 36 Rollerin aktifleştirilmesi.	143
Şekil 5. 37 Sisteme yeni rolün eklenmesi.	143
Şekil 5. 38 Yeni kullanıcı kaydı.	144
Şekil 5. 39 Erişim kuralları.	145

Şekil 5. 40 Klasör yetkileri.	146
Şekil 5. 41 Web sitesi erişim kuralları.	147
Şekil 5. 42 Rol ekleme ve silme.	148
Şekil 5. 43 Roller ve kullanıcıları.	149
Şekil 5. 44 Aktif/Pasif kullanıcılar.	150
Şekil 5. 45 Kullanıcı ekleme işlemleri.	151
Şekil 5. 46 Kullanıcı bilgilerini düzenleme.	152
Şekil 5. 47 Kullanıcı bilgilerinin belirtilmesi.	153
Şekil 5. 48 Sistemdeki kayıtlı kullanıcılar.	154
Şekil 5. 49 Sistemdeki kayıtlı kullanıcılar.	156
Şekil 5. 50 Bölüm başkanı rolünün verilmesi.	157
Şekil 5. 51 Kullanıcıya yeni bir rolün verilmesi.	158
Şekil 5. 52 Verilen yeni rolün görevlendirileceği fakülte.	159
Şekil 5. 53 Yeni üvan ekleme sayfası.	164
Şekil 5. 54 İşletme organizasyonu	167
Şekil 5. 55 Okul ekleme işlemi.	168
Şekil 5. 56 Sisteme eklenen okullara Dekan yada Müdür ekleme işlemi.	169
Şekil 5. 57 Okul organizasyon şeması.	172
Şekil 5. 58 Bölüm tanımlama sayfası.	173
Şekil 5. 59 Bölüm Başkanı atama.	174
Şekil 5. 60 Örnek Okul, Bölüm ve Program organizasyonu.	176
Şekil 5. 61 Program ekleme işlemi.	178
Şekil 5. 62 Programa dönem ekleme işlemi.	180
Şekil 5. 63 Döneme ders ekleme.	182
Şekil 5. 64 Döneme zorunlu seçmeli ders gurubu ekleme.	184
Şekil 5. 65 Programa yeni ders ekleme.	189
Şekil 5. 66 SCORM 2004 paketinin sisteme eklenmesi.	190
Şekil 5. 67 SCORM 2004 paketin öğrenci ekranında görüntülenmesi (Hakkari,2009).	190
Şekil 5. 68 Alt konu ekleme işlemi.	191
Şekil 5. 69 Sınav ekleme işlemi.	192

Şekil 5. 70 Ödev ekleme işlemi.	193
Şekil 5. 71 Ders kaynak bilgileri ekleme işlemi.	194
Şekil 5. 72 Ders girebilecek öğretim elemanlarının belirlenmesi.	198
Şekil 5. 73 Bölüm Sekreterinin bildirilen programı düzenlemesi.	200
Şekil 5. 74 Ders sorumlusunun ders istek saatini belirtmesi.	201
Şekil 5. 75 Şube ekleme işlemi.	204
Şekil 5. 76 Ders şubesi tanımlama.	205
Şekil 5. 77 Danışman atama işlemi.	209
Şekil 5. 78 Ders kayıt işlemleri için bölüm sekreterinin onay vermesi.	212
Şekil 5. 79 Öğrenci ders kayıt işlemi.	214
Şekil 5. 80 Danışman ders onay işlemi	215
Şekil 5. 81 Articulate ile sınav hazırlamak.	219
Şekil 5. 82 Soruların SCORM uyumlu hale getirilmesi.	220
Şekil 5. 83 Articulate programının hazırlanan soruları SCORM paketine dönüştürmesi.	220
Şekil 5. 84 Articulate programı ile SCORM 2004 paketinin oluşturulması.	221
Şekil 5. 85 Sınav paketi ekleme	223
Şekil 5. 86 Sınav paketini seçme	224
Şekil 5. 87 Online sınav ekranı.	226
Şekil 5. 88 Ödev ekleme işlemi.	228
Şekil 5. 89 Ödev düzenleme ve silme işlemi	229
Şekil 5. 90 Öğrenci ödev gönderme işlemi.	230
Şekil 5. 91 Ödev değerlendirme işlemi.	231
Şekil 5. 92 Öğretim elemanı performans takip.	234
Şekil 5. 93 Öğrenci performans raporu.	236
Şekil 5. 94 Öğrenci performans notu.	239
Şekil 5. 95 Test Sonuçları	242

## TABLULAR DİZİNİ

Sayfa No

Çizelge 2. 1 Uzaktan Eğitim Projesindeki Personel Gereksinimleri (Codone 2001).	11
Çizelge 3. 1 SCORM API 1.2 Veri Tabanı Yapısı.	21
Çizelge 3. 2 SCORM API 1.2 veri modeli.	23
Çizelge 3. 3 SCORM_APPLICATIONDATA tablosu.	29
Çizelge 3. 4 SCORM_COURSEINFO Tablosu.	30
Çizelge 3. 5 SCORM_COURSESTATUS tablosu.	31
Çizelge 3. 6 SCORM_ITEMINFO tablosu.	32
Çizelge 3. 7 SCORM_SCOCOMMENTS tablosu.	34
Çizelge 3. 8 SCORM_USERCOURSEINFO tablosu.	35
Çizelge 3. 9 AdvSCORM_2004_DMELEMENT tablosu.	36
Çizelge 3. 10 SCORM_OBJ_ADVDMELEMENTDESCRIPTOR tablosu.	37
Çizelge 3. 11 SCORM_OBJ_ADLTOC tablosu.	39
Çizelge 3. 12 SCORM_OBJ_ADLVALIDREQUESTS tablosu.	40
Çizelge 3. 13 SCORM_OBJ_ADVADLTARACKING tablosu.	41
Çizelge 3. 14 SCORM_OBJ_ADLTOC tablosu.	42
Çizelge 3. 15 SCORM_OBJ_ADVSEQOBJECTIVETRACKING tablosu.	43
Çizelge 3. 16 SCORM_OBJ_ADVVERSION tablosu.	44
Çizelge 3. 17 SCORM_OBJ_SEQOBJECTIVE tablosu.	45
Çizelge 3. 18 SCORM_OBJ_SEQOBJECTIVEMAP Tablosu	46
Çizelge 3. 19 SCORM_OBJ_SEQACTIVITY tablosu.	48
Çizelge 3. 20 SCORM 2004 Runtime CMI Veri Modeli.	53
Çizelge 4. 1 İyileştirilmiş güvenlik özelliklerinin özeti(İnt. Kyn. 7)	69
Çizelge 5. 1 Kişisel Bilgiler Ekranında doldurulacak bilgiler.	95
Çizelge 5. 2 Online kayıt modülündeki bilgilerin veri tabanını ilişkisi.	116
Çizelge 5. 3 Personel kayıt işlemleri Kişisel Bilgiler diyagramı.	123
Çizelge 5. 4 Kullanıcı ve Kullanıcı Grupları modlü ile ilişkili veri tabloları.	128
Çizelge 5. 5 aspnet_users tablosu.	160



Çizelge 5. 6 aspnet_Membership tablosu.	160
Çizelge 5. 7 aspnet_Roles tablosu.	160
Çizelge 5. 8 aspnet_UsersInRoles tablosu.	161
Çizelge 5.9 aspnet_Membership, aspnet_users, aspnet_UserInRoles, aspnet_Roles tabloları arasındaki ilişki.	161
Çizelge 5. 10 aspnet_Membership, aspnet_users, aspnet_UserInRoles, aspnet_Roles tabloları arasındaki ilişkinin basite indirgenmiş hali.	162
Çizelge 5. 11 Ünvan ve bağlı olduğu birimler.	163
Çizelge 5. 12 Ünvan tablosu.	165
Çizelge 5. 13 Okul Tablosu ve bu tablo ile ilişkili olan Okul_Tip_Tbl tablosu.	170
Çizelge 5. 14 Bolum_Tbl tablosu ve Okul_Tbl tablosu arasındaki ilişki.	175
Çizelge 5. 15 Eğitim Kataloğu verilerinin tutulduğu ve bu tabloların ilişkili olduğu tablolar.	185
Çizelge 5. 16 Dersler tablosu ve ilişkili veri tabloları.	195
Çizelge 5. 17 Haftalık ders programı modülünde kullanılan tablolar ve bu tabloların ilişkili olduğu tablolar.	202
Çizelge 5. 18 Şube işlemlerinde kullanılan tablolar ve bunların ilişkili olduğu tablolar.	206
Çizelge 5. 19 Danışman işlemlerinde kullanılan veri tabloları ve bu tabloların ilişkili olduğu tablolar.	210
Çizelge 5. 20 Ders kayıt işlemlerinde kullanılan tablolar ve bu tabloların ilişkili olduğu tablolar.	216
Çizelge 5. 21 Odevler Modülünde kullanılan tablolar ve bu tablolar ile ilişkili tablolar	232

## SİMGELER VE KISALTMALAR DİZİNİ

### 1. Kısaltmalar

ADL	Advanced Distrubuted Learning
AICC	The Aviation Industry CBT Commitee
AÖF	Açık Öğretim Fakültesi
API	Uygulama Programlama Arayüzü (Aplication Programming Interface)
ATBİT	Araştırma Teknikleri Ve Bilimsel İletişimin Temelleri
CMI	Bilgisayar Denetimli Eğitim(Computer Managerd Instructrion)
IEEE	Elektrik ve Elektronik Mühendisleri Enstitüsü(Institute of Electrical and Electrincs Engineers)
IMS	Öğretimsel Yönetim Sistemi (International Project Manager)
LMS	Öğrenim Yönetim Sistemi (Learning Managment Systems)
ODTÜ	Orta Doğu Teknik Üniversitesi
ÖYS	Öğrenme Yönetim Sistemi
SCORM	Paylaşılabilir İçerik Nesnesi Referans Modeli (Sharable Content Object Reference Model)
SQL	Yapılandırılmış Sorgulama Dili (Structured Query Language)
YL	Yüksek Lisans
YÖK	Yüksek Öğretim Kurulu
WWW	World Wide Web
XML	Extensible Markup Language

## 1. GİRİŞ

Uzaktan öğretim, kablolu, kablosuz ve posta yoluyla haberleşmeyi sağlayan kanallar üzerinden; internet, intranet, video, telefon, televizyon, bilgisayar, radyo, mektup vb. araçlarla sunulan; öğrenci ile öğretmenin birbirlerinden fiziksel olarak ayrı olmalarına rağmen eş zamanlı ve eş zamansız bir araya gelerek öğrenme ve öğretmeyi kısmen ya da tamamen gerçekleştirdikleri öğretim faaliyetlerinin tümüdür (Toprakçı 2008).

Başlangıcı yazının icadına kadar dayanmakta olan uzaktan eğitim, Türkiye’de ilk olarak 1927 yılında eğitim sorunlarının görüşüldüğü bir toplantıda dile getirilmiştir. 1950’li yıllara kadar devam eden tartışmaların ardından, Türkiye’de ilk uzaktan eğitim uygulaması 1956 yılında Ankara Üniversitesi Hukuk Fakültesi, Banka ve Ticaret Hukuku Araştırma Enstitüsünde başlatılmıştır. Bankada çalışanların hizmet içinde yetiştirilmesini amaçlayan bu çalışmada banka çalışanları mektupla öğrenim görmüşlerdir. 1958-1974 yılları arasında ise Milli Eğitim Bakanlığı Mesleki ve Teknik Öğretim Müsteşarlığı tarafından çalışmalarda bulunulmuş ve bazı kurum ve kişilerin yabancı dil ve meslek öğretimi alanlarında bu yöntem kullanılmıştır. 1958 yılında Mektupla Öğretim Merkezi kurularak, okul bitirmek isteyenlere hazırlık programı uygulanmıştır. 1974 yılında Mektupla Yüksek Öğretim Merkezi kurulmuştur. Bu merkez 1983 yılında Mesleki ve Teknik Açık Öğretim Okulu adını almıştır. 15 ay sonra Yaygın Yüksek Öğretim Kurumu (YAYKUR) kurulmuştur. 1981 yılında 2547 sayılı Yüksek Öğretim yasası ile birlikte Anadolu Üniversitesi bünyesinde Açık Öğretim Fakültesi kurulmuştur. 1992 yılında uzaktan eğitim orta öğretimde de yerini almıştır (İnt. Kyn. 1).

İnternet üzerinden eğitim ile sanal sınıflar oluşturularak klasik sınıf ortamlarından uzaklaşmıştır. Öğretmen-öğrenci iletişiminin geleneksel işleyişi terk edilmiş, “eş zamanlı” ve “eş zamansız” olmak üzere farklılaşmış, ders içerikleri; içeriğin sunuluş şekline uygun olarak elektronik ortamlarda geliştirilmeye başlanmış, her öğrenciye öğrenimin gerçekleşmesi için yeterli zaman ve tekrar etme imkânı sağlanarak günümüz modern eğitim modellerinden tam öğrenme modeli hedef alınmıştır.

Ülkemiz nüfusu hızlı bir şekilde artmaktadır. Artan bu nüfus oranı ile paralel bir şekilde artması gereken okullaşma ve öğretmen ihtiyacı ne yazık ki aynı paralelliği gösterememektedir. Bu olumsuzluk ülkemiz eğitim sistemini alternatif eğitim yöntemlerine yönlendirmektedir. Bu sistemlerin başında Web Tabanlı Eğitim (WTE) sistemleri başı çekmektedir. Ülkemizde bu uygulamayı Yüksek Öğretim düzeyinde ilk olarak Anadolu Üniversitesi ile başlatılmış, şu an ise Sakarya, Ahmet Yesevi, ODTÜ ve bazı özel kurum ve devlet kuruluşları kendi Web tabanlı eğitim sistemlerini oluşturmaktadırlar. Ayrıca bilgi ve bilgisayar teknolojisinin hızlı bir şekilde gelişme göstermesiyle var olan bilgi her gün artış göstermektedir. 21. yüzyılın en önemli özelliği bilgiyi tek bir kaynaktan öğrenme yerine, çoklu ortamlardan bilgiye ulaşma ve ulaşılan bu bilgiyi değerlendirme, zaman ve mekân sınırı tanımadan diğerleri ile paylaşma, tartışma ve gerekli durumlarda bu bilgiyi gerçek yaşamda kullanabilmeyi gerektirmektedir. Bu nedenle, sanal üniversiteler, sözü edilen tüm bu yaşantıları kullanıcılara sunabilen sanal iletişim ve öğrenme ortamı haline gelebilmelidir. Bu bağlamda, en önemli konu, sanal eğitim ortamlarının yönetimi ve değerlendirilmesi olmaktadır (İbili ve ark. 2008).

Sanal eğitim ortamlarının yönetimi için Öğrenim Yönetim Sistemleri (Learning Management System-LMS) tasarlanmaktadır. Bu sistemler kullanıcı ile öğretmen materyalleri arasındaki aktiviteleri izleyen, yöneten ve raporlayan yazılımlardır. Günümüzde e-öğrenme olarak da adlandırılan bu yazılımları üreten şirket ve üniversitelerce geliştirilen LMS'lerin yanı sıra, kaynak kodları açık pek çok LMS örnekleri de bulunmaktadır. LMS'ler arasında farklı standartlar olması eğitim materyallerinin paylaşımını, yeniden kullanılabilirliğini güçleştirmektedir. Bu güçlükler LMS tasarımcılarını ortak standartlar geliştirmeye yöneltmiş ve ADL adlı şirket tarafından halen geliştirilmekte olan SCORM standartları ortaya çıkmıştır. ADL, Ocak 1999'da, A.B.D.'de Savunma Departmanı (Department of Defense - DoD) kamu ile özel sektör için teknoloji tabanlı eğitim standartları ve kuralları geliştirilmesi işini üstlenmiştir (Ostyn 2007). SCORM uzaktan eğitim içeriklerinin, ortak bir teknik çerçeve içinde tekrar kullanılabilir eğitim nesneleri şeklinde tasarlanmasını ve hazırlanmasını sağlayan bir kurallar ve standartlar paketidir SCORM (ADL 2004).

- IEEE Data Model For Content Object Communication,
- IEEE ECMAScript Application Programming Interface for Content to Runtime Services Communication,
- IEEE Learning Object Metadata (LOM) ,
- IEEE Extensible Markup Language,
- (XML) Schema Binding for Learning Object Metadata Data Model,
- IMS Content Packaging,
- IMS Simple Sequencing,

standartlarına dayanmaktadır. Sırasıyla SCORM 1.0 (Ocak 2000), SCORM 1.1 (Ocak 2001), SCORM 1.2 (Ekim 2001), SCORM 2004 1st Edition (Ocak 2004), SCORM 2004 2nd Edition (Temmuz 2004), SCORM 2004 3rd Edition (Ekim 2004 - Son Versiyon) yayınlanmıştır. Bununla birlikte ADL aşağıdaki kitapları çıkarmıştır ([www.adl.org](http://www.adl.org) adresinden bu kitapların elektronik formatları indirilebilir).

- SCORM 2004 3rd Edition Overview
- SCORM 2004 3rd Edition Content Aggregation Model (CAM)
- SCORM 2004 3rd Edition Run-Time Environment (RTE)
- SCORM 2004 3rd Edition Sequencing and Navigation (SN)

Günümüzde SCORM Standartlarına uygun olarak hazırlanmış çok sayıda LMS bulunmaktadır. Bunların başında MOODLE gelmektedir. MOODLE toplam 75 ülkede, 77 ayrı dil desteği ve toplam 75000 kullanıcısı ile en yaygın Eğitim Öğretim Yönetim Sistemidir (Elmas ve ark. 2008). MOODLE ile birlikte Enocta şirketinin geliştirmiş olduğu uzaktan eğitim yönetim sistemi, ODTÜ'nün geliştirdiği IDEA, Sakarya Üniversitesinin geliştirdiği LMS'ler de mevcuttur.

SCORM, bir Eğitim Öğretim Yönetim Sisteminin (ADL 2004).

- Dayanıklı,
- Yeniden kullanılabilir,
- Diğer yazılımlarla birlikte çalışabilir,

- Ulaşılabilir,
- Taşınabilir,

olmasını sağlamaktadır. Günümüz Öğrenim Yönetim Sistemleri hazırlanırken SCORM standartları ile birlikte Modülerlik ön plana çıkmıştır. Modüler Öğrenim Yönetim Sistemlerinin en önemli avantajı, kendi içerisinde belirli iş yüküne sahip parçalara (modüllere) ayrılmasıdır. Bu sayede her bir iş yükü kendi içerisinde genişletilebilir, esnetilebilir, yönetilebilir hale gelmektedir.

Bu çalışmada Yüksek Öğretim Kurumlarının alt yapısına uygun olarak, SCORM uyumlu Modüler Eğitim Yönetim Sistemi tasarlanmış ve tasarlanan bu LMS tamamlanmıştır. Toplam 36 modülden oluşan eğitim yönetim sistemi oldukça basite indirgenmektedir. SCORM1.2 ve SCORM 2004 standartlarına uygun olarak hazırlanmıştır. Hazırlanan LMS' in tüm modülleri test edilmiştir. Testler sonucunda LMS'in başarılı bir şekilde çalıştığı izlenmiştir.

## 2. GENEL KAVRAMLAR

### 2.1 Eğitim

Eğitim bir ülkenin kalkınması, o ülkede yaşayan insanların çağdaş ve yeni teknolojilerin kullanıldığı, planlı, bilinçli ve çok seçenekli eğitim süreçlerinden yeterince yararlanılması ile mümkün olabilen, toplumun gelişmesini sağlayan önemli süreçlerdendir. Hızlı bir değişim ve gelişim yaşadığımız bugünlerde, çalışan kesimin, çalıştıkları alanlarda en yüksek standartlara ulaşmaları hedeflenmekte, bu hedefe ulaşmanın ilk basamağını da nitelikli insan gücü yetiştirmek oluşturmaktadır.

Eğitime etki eden faktörler sıralamak gerekirse bunların en merkezinde öğrenme performansı gelmektedir. Öğrenme Performansına Etki eden 3 önemli nokta bulunmaktadır (Codone 2001).

#### 1) Aile

- Aile beklentisi
- Aile yapısı
- Ailenin eğitimsel iletişimi
- Aile katılımı

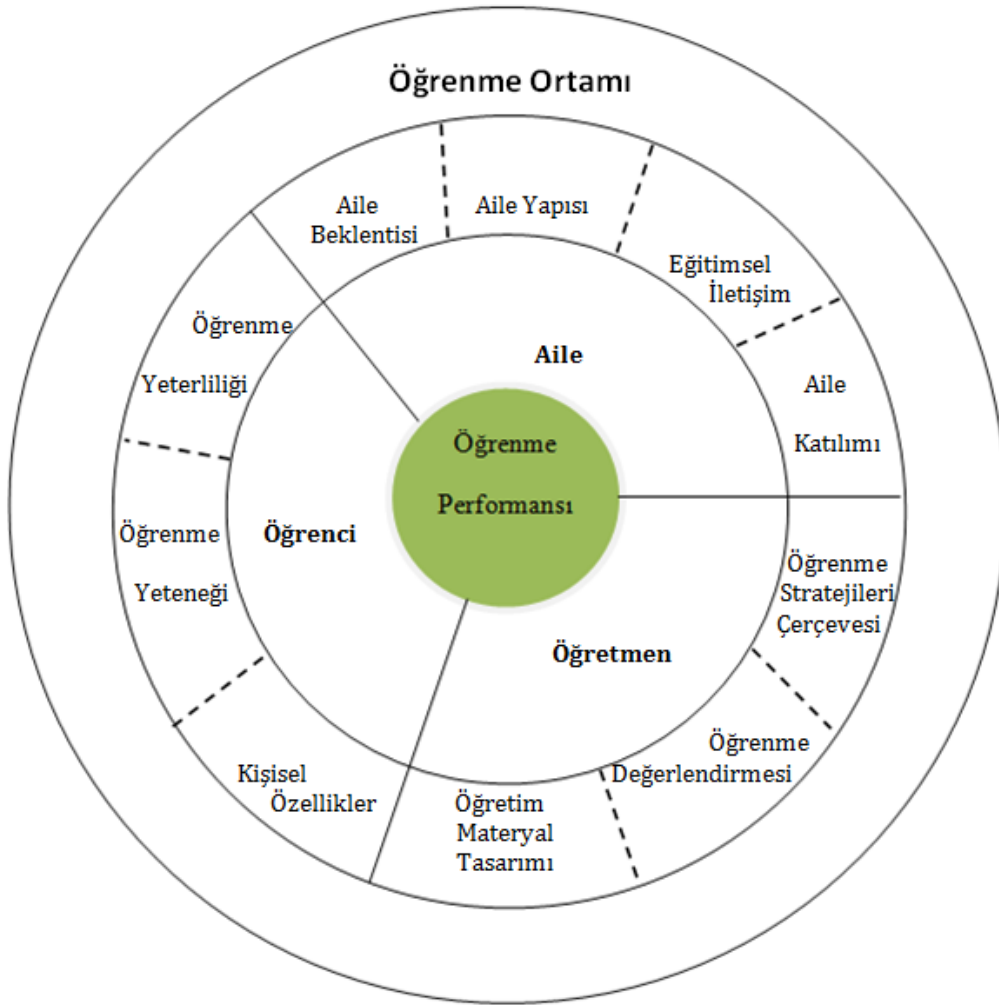
#### 2) Öğretmen

- Öğretmen stratejileri
- Öğretmen değerlendirmesi
- Öğretimde materyal tasarımı

#### 3) Öğrenci

- Öğrenci yeteneği
- Kişisel özellikler
- Öğrenme şekilleri

Uzaktan Eğitim Sistemleri, aileyi, öğrenci performanslarının takibi ve değerlendirmesi gibi raporlar ile öğrenciyi yönlendirmelerini sağlayarak eğitime katabilmektedir. Eğitimdeki en önemli faktörlerden bir tanesi de öğretmendir. Uzman öğretimsel tasarımcılar ile hazırlanan eğitim içerikleri; Öğretmen stratejileri, Öğretmen yeterliliği, Öğretimde materyal kullanımı gibi sıkıntıları ortadan kaldırmaktadır. Öğrencilere, öğrenmeleri için ihtiyaç duydukları kadar zaman ve tekrar etme olanağı sağlayarak, öğrencilerin sahip oldukları öğrenme düzeyi gibi bireysel farklılıkları ortadan kaldırarak tam öğrenme gerçekleşmesi sağlanabilir.



Şekil 2. 1 Eğitimsel daire modeli (Codone 2001).



## 2.2 Uzaktan Eğitim

Uzaktan Eğitim birçok şekilde isimlendirilse de en en yaygın olanları; Online Eğitim (online Learning), Sanal Eğitim (virtual learning) , Dağınmık Eğitim (distributed learning), Ağ ve web tabanlı Eğitim (network and web based learning) dir. “Farklı mekânlardaki öğrenci, öğretmen ve öğretim materyallerinin iletişim teknolojileri aracılığıyla bir araya getirildiği kurumsal bir eğitim faaliyetidir” (Işık ve ark. 2008).

Örgün eğitim sisteminin en belirgin özelliği eğitmen ve öğrencinin yüz yüze iletişim kurmasıdır. Bu tip eğitim genel olarak öğrencilerin değişik zeka ve öğrenme yetenekleri yerine grubun genel seviyesine göre düzenlenmektedir. Ancak ideal olan her öğrencinin kişisel özellikleri dikkate alınarak hazırlanmış ders içerikli eğitim sistemidir. Örgün eğitim anlayışı ile gerçekleştirilmesi çok zor olan bu sistem, uzaktan eğitim ve kapsamlı Öğrenim Yönetim Sistemleri sayesinde hayata geçirilmeye başlanmıştır. Yapılan yatırım maliyetini karşılayacak sayıda öğrencinin olması durumunda uzaktan eğitim, diğer eğitim sistemleri arasında en ucuz olanıdır (İnt. Kyn. 2).

Uzaktan Eğitimin gereksinimi başlıca iki kavram veya inanıştan kaynaklanmaktadır. Bunlardan birincisi, fırsat eşitliği kavramı ya da toplumdaki tüm bireylerin, eğitim yaşantı ve olanaklarından yararlanmaya hakkı olduğu inancı; diğeri ise, eğitimin bir dizi bireysel ve toplumsal amaçların gerçekleşmesine katkıda bulunabilecek başlıca etmenlerden biri olduğu hakkında beslenen inançtır. Bugün bile eğitimde fırsat ve olanak eşitsizliklerinin giderilememiş olması nedeniyle kalite ve standart düşüklüğü sorunlarıyla karşılaşmaktadır. Bu nedenle eğitimde verim ve etkinliliği arttırmak, bireylerin değişen eğitim ihtiyaçlarına cevap vermek, eğitim teknolojisinin kitle eğitim boyutunu oluşturan uzaktan eğitimin gelişmesine yol açmıştır. Uzaktan eğitim sayesinde herkese yaşam boyu eğitim imkânı sağlanmıştır (İnt. Kyn. 3). Uzaktan Eğitimin Genel Özellikleri aşağıda verilmiştir (Işık ve ark. 2008).

- **Küreselleşme:** Uzaktan eğitim hizmeti veren kurumlar genel olarak küresel düzeyde eğitim verebilecek niteliğe sahip olmalıdır. 100 yılı aşkın bir süredir Avrupa'da (özellikle İngiltere ve Fransa) bu tür faaliyetlerde bulunan eğitim kurumları deniz

aşırı ülkelerdeki çalışan resmi görevli ya da ticaretle uğraşan vatandaşlarına düzenli olarak uzaktan eğitim hizmeti vermektedir.

- ***Kişiselleştirme:*** Doğu ve Batıdaki Geleneksel Eğitim sisteminin en belirgin özelliği eğitmen ve öğrencinin yüz yüze iletişim kurması olup bu tip eğitim öğrencilerin değişik zekâ ve öğrenme yetenekleri yerine grubun genel seviyesine göre düzenlenmektedir.
- ***Özelleştirme:*** Uzaktan eğitim sistemi, öğrenciyi sınıf ortamından alarak bireysel olarak eğitilebileceği bir konuma taşır. Bu sistemde, kurumsal öğrenmenin yerini bireyselleştirilmiş öğrenme alır.
- ***Endüstrileşme:*** Büyük bir kitlenin bir ürüne olan artan ihtiyacını karşılayabilmek için ilgili endüstrilerin kurulmasına benzer bir şekilde, insanların artan eğitim taleplerine hızlı ve etkin bir şekilde cevap verebilmek için uzaktan eğitim kurumlarının açılması kaçınılmaz duruma gelmiştir.
- ***Geleneksel eğitime uygun olmayan öğrencilere hizmet verme:*** Uzaktan eğitim, dünya çapında her yıl milyonlarca kişi tarafından tercih edilen bir eğitim sistemidir. Bununla beraber eğitim almak istedikleri eğitim kurumundaki ders saatlerine yer / zaman açısından katılma imkânı olmayanlar (tam zamanlı olarak çalışanlar - memurlar, askerler - farklı şehir / ülkede yaşayanlar) için uygun bir eğitim sistemidir.
- ***Hareket kabiliyeti:***1980'li yıllarda, uzak mesafeler arasında iletişimi kurmak amacıyla sabit bilgisayar konferans sistemleri ve iki yönlü video konferans sistemleri kullanılmakta iken, günümüzde bunlara ilave olarak portatif bilgisayarlar ve cep telefonları önemli düzeyde kullanılmaya başlanmıştır.
- ***Hızlı geri besleme:*** Günümüzde uzaktan eğitim sayesinde öğrenciler, e-posta yolu ile dünyanın herhangi bir yerinden günün herhangi bir saatinde ödevlerini

gönderebilmekte ve bu çalışmalarının değerlendirme sonuçlarını hemen aynı şekilde internet üzerinden alabilmektedirler.

- ***Diğer eğitim sistemlerine göre ucuz olması:*** Altyapıya yönelik yatırımın çok yüksek düzeyde olması ya da öğrenci başına düşen ücretlendirmenin geleneksel eğitime göre daha yüksek olması veya yapılan yatırım maliyetini karşılayacak sayıda öğrencinin bulunamaması durumları hariç, uzaktan eğitim, eğitim sistemleri arasında en ucuzudur.
- ***Teknoloji ve Eğitim:*** Sanal sınıflar uydu veya sıkıştırılmış video kodlama ya da tam bant genişliği kullanılarak birbirlerine bağlanabilmekte ve bu sayede kişiler uzak yerlerde olsalar bile yüz yüze eğitim alabilmektedirler.

### **2.2.1 Uzaktan Eğitimin Avantajları**

Uzaktan eğitimin genel olarak sayılabilecek avantajları aşağıda verilmiştir (Çağiltay 2001).

- Zamandan ve mekândan bağımsız olması,
- Evinden dışarı çıkamayan (engellilerin) eğitim alabilmesi,
- Eğitim verimliliğinde artış sağlanması,
- Bilgi ve birikimlerin paylaşılmasına olanak sağlanması,
- Kesintisiz ve sınırsız eğitim imkânı vermesi,
- Eğitim merkezlerinden dinamik ve geniş kitlelere eğitim imkânı sunması,
- Yalnızca ulusal değil, uluslararası standartlar da göz önüne alınarak yapılandırılması,
- Bir servisten tüm bilgi düzeltilip daha kolay güncellenerek kitlelere aktarım imkânı sunması,
- Öğrencilerin metin, grafik, animasyon gibi görsel-işitsel teknikler kullanılarak motivasyon ve bilginin kalıcılığının sağlanması,

- Çalışan insanlara buldukları yerden eğitim hayatlarına devam etmelerine olanak sağlaması,
- Kişilerin gelişimlerinin takip edilip, geri bildirimlerini kolayca yapılabilmesi,
- Öğrenciler açısından, eğitim giderlerini (barınma, yol masrafları ve genel eğitim giderleri) aza indirmesi,
- Postalama masrafının olmaması bilgi dağıtım baskı ve taşıma masrafları sıfırlanmış olarak internet üzerinden yapılabilir olması,
- Eğitimci ve öğrencinin farklı mekânda olmasına rağmen İnternet aracılığıyla tek veya çift yönlü iletişim kurulabilmesi.

## **2.2.2 Uzaktan Eğitimin Dezavantajları**

Uzaktan eğitimin avantajlarının yanında dezavantajları da vardır. Bunlar aşağıda verilmiştir (Çağiltay 2001).

- Öğrenme ortamlarında önemli görülen yüz yüze etkileşim ortam ve olanakları, öğrenme sürecinde karşılaşılan öğrenme güçlüklerinin anında çözülememesi ve bu durumun ardından gelişebilecek sıkıntılar,
- Anında yardım görememe ve sorunun giderilmemesinden kaynaklanan davranışların gelişimi,
- Kendi kendine çalışma alışkanlığı olmayan ve bu yeteneğini geliştirmemiş bireyler için planlama zorluğu,
- Çalışan bireylerin kendine ayıracakları vakitte ders çalışma zorunluluğu,
- Laboratuvar, atölye gibi uygulama ağırlıklı konuların işlenmesindeki sınırlılıklar,
- Öğrenci sayısındaki fazlalık nedeni ile iletişimdeki sınırlılıklar.

### 2.3 Uzaktan Eğitimin Tasarlanması Ve Geliştirilmesi Süreci

Uzaktan eğitim sistemlerinin tasarlanması ve geliştirilmesi süreci; kurslar, seminerler, workshoplar, online öğrenme portalları, chat ve tartışma gurupları, yazılım ve donanım, çeşitli insan kaynakları, standartlar, öğrenci gereksinimlerine uygun tasarım geliştirilmesi, ve uygulamalardır (Naidu 2006). Bu sürecin başarılı bir şekilde yürütülmesi için, iyi bir takım çalışması gerekmektedir. Takım çalışmasının başarılı olması ise, takımdaki görev ve yetkilerin iyi şekilde belirlenmesi ve buna uygun bir ekibin oluşturulmasına bağlıdır. Uzaktan Eğitim sürecinde oluşabilecek rol ve yetkiler Çizelge 2.1 de verilmiştir.

**Çizelge 2. 1** Uzaktan Eğitim Projesindeki Personel Gereksinimleri (Codone 2001).

<b>Personel</b>	<b>Meslek Bilgisi</b>	<b>Görev</b>
<b>Öğretimsel Tasarımcı</b>	Öğretimsel Tasarım yada öğretim teknolojileri alanında üniversite mezunu olmalıdır.	Öğretim Meteryallerini planlama ve tasarlama yetkisine sahiptir.
<b>Grafik Tasarımcısı</b>	Üniversite eğitimi her zaman gerekli değildir. Genelde iş yerinde edinilen bir meslektir.	Çizim, Animasyon, web yada medya alanlarında görevlidir.
<b>Programcı veya Yazılım Uzmanı</b>	Üniversitelerin Bilgisayar bilimleri, yazılım mühendisliği alanlarından yada bu alanlarda iş tecrübesi olan.	Web yada diğer uygulama yazılımlarında programlama dilleri ile script yada kod yazmakla görevlidir.
<b>İçerik Konu Uzmanı</b>	İçerikler hazırlanırken önemlidir. Çok fazla eğitim tecrübesi gerektirmez.	Kurslara doğru probleblemler ve seneryolar ekleyen, gerçek yaşamla ilgili örnekler gerçekleştiren, içerik aydınlatıcısıdır.
<b>Kalite Kontrol teknisyeni</b>	Kalite Yönetimi alanında tecrübeli olması önemlidir.	Dilbilgisine uygunluk, estetik ve kalite açısından dikkatli bir şekilde takip eder ve değerlendirir

<b>Site Yöneticisi Veritabanı uzmanı</b>	Web yada bilgisayar bilimleri alanında tecrübeli veya bu alanlarda eğitim almış.	Kursları intranet üzerinden aktaran, yöneten web siteleri, öğrenme portalları ve LMS leri hazırlamakla görevlidir.
<b>Proje Yöneticisi</b>	Yönetim ya da teknoloji Mezunu olmalıdır.	Proje Planını, görevli Personelleri, müşteri ilişkilerini yönetir.

#### **2.4 SCORM (Sharable Courseware Object Reference Model)**

ADL (Advanced Distributed Learning Initiative) tarafından yönetilen ve yayınlanan bir standarttır. Uzaktan eğitim içeriklerinin, ortak bir teknik çerçeve içinde tekrar kullanılabilir eğitim nesneleri şeklinde tasarlanmasını ve hazırlanmasını sağlayan bir kurallar ve standartlar paketidir. SCORM, bir Eğitim Öğretim Yönetim Sisteminin; dayanıklı, yeniden kullanılabilir, diğer yazılımlarla birlikte çalışabilir, ulaşılabilir, taşınabilir, olmasını sağlamaktadır (ADL 2004).



Şekil 2. 2 SCORM sürümlerindeki gelişim evreleri.

SCORM sürümleri arasında bazı farklılıklar vardır. Bu farklılıklar aşağıda verilmiştir (Doruk 2006).

**SCORM 1.0:** SCORM' un ilk sürümü (SCORM 1.0) genel olarak, öğrenme yazılımının birlikte işlerliğine ve yeniden kullanımına yönelik standartlardan oluşmaktadır. İlk sürümde SCORM; “Sharable Courseware (Eğitim Yazılımı) Object Reference Model” şeklinde telaffuz ediliyordu. İkinci sürümle yapılan en önemli değişiklik Courseware ifadesinin Content'e (İçerik) dönüştürülmesi oldu. Böylece SCORM e-öğrenme içeriğinin geliştirilmesinde her türlü seviyeye uyarlanabilecekti. Bu sürümde, daha çok ilk sürümün kullanıcılarından toplanan geri bildirimler doğrultusunda yapılan düzeltmeler ve geliştirmeler yer almaktadır.

**SCORM 1.2:** Ekim 2001'de sürülen SCORM 1.2 ile e-öğrenme yazılımlarının birlikte işlerliğine yönelik önemli bir adım atılmıştır. Bu amaçla, SCORM'a, IMS'in içerik paketleme standartları dahil edilmiş ve metadata standartları geliştirilmiştir. Bu XML (eXtensible Markup Language) tabanlı paketleme ve meta data (üst veri) standartları sayesinde içerik ambarları ile LMS (Learning Management Systems) arasındaki bağlantı da kurulmuştur.

**SCORM 2004:** Ocak 2004'te ADL, SCORM' un son sürümünü, SCORM 2004 olarak sunmuştur. İsim değişikliğine nedeni, SCORM 2004' e Sıralama ve Dolaşım Standartları (Sequencing and Navigation – SN) eklenmiştir. Sıralama ve Dolaşım Standartları, kullanıcının içerikle ilgili deneyimlerinden yola çıkarak, içerik sunumunun kişiselleştirilebilmesine olanak sağlamıştır.

## **2.5 Modüler Yapı**

Modüler Sistem, bütünü en yararlı olabilecek parçalara bölerek tümevarım yöntemiyle bütünü tamamlamaktır. Modüler sistemde her bir modül belirli bir iş yüküne sahiptir. Modülün iş yükü programın tamamı düşünülerek ve ilgili modül için beklenen görevler göz önüne alınarak hesaplanır. Modüler yapı, puzzle parçalarının bir araya getirilmesi ile oluşan bir yapıya benzetilebilir. Bu yapıda oluşan Öğrenim Yönetim Sistemleri nde, LMS modüllere ayrılmıştır. Her bir modül, LMS'in tamamı düşünülerek oluşturulmaktadır. Modüler sistemin diğer bir avantajı ise, LMS dinamik hale dönüşmektedir. Bu sayede, gerekli modüller sisteme entegre edilebildiği gibi, gereksiz modüller de sistemden çıkartılabilmektedir. Böylece LMS ile kolayca etkileşebilen ve hızlı çalışabilen modüler bir yapı ortaya konmaktadır.



### 3. SCORM

SCORM 1997 yılında Amerika Birleşik Devletleri ordusu tarafından geliştirilmiştir. Çeşitli ülkelerdeki ve farklı platformlardaki Amerikan Ordularının eğitimi için ortaya atılmıştır. Başta İngiltere olmak üzere çeşitli ülkelere ve akademisyenlerden gelen desteğin ardından ADL (Advanced Distributed Learning) adlı şirket ortaya çıkmıştır. ADL'nin çalışmaları doğrultusunda, e-öğrenme alanında geliştirilmiş birçok standart ve tanımlamalar tek bir başvuru modeli çatısı altında toplanmaya başlanmış ve oluşturulan bu modele ise SCORM (Sharable Courseware Object Reference Model) adı verilmiştir. SCORM, bir e-öğrenme yazılımının, dayanıklı, yeniden kullanılabilir, diğer yazılımlarla birlikte çalışabilir, ulaşılabilir ve taşınabilir olması için geliştirilen standartlardan uyarlanarak oluşturulmuş bir başvuru modelidir. SCORM; İçerik Kümesi Modeli, Çalışma Ortamı ve Sıralama ve Dolaşım bölümlerinden oluşmaktadır.

#### 3.1 İçerik Kümesi Modeli (Content Aggregation Model-CAM)

İçerik Kümesi Modeli içerisinde belirtilen her bir öge, öğrenme aktivitesine karşılık gelmektedir. CAM dört bölümden oluşmaktadır. (ADL 2004).

##### 3.1.1 İçerik Modeli (Content Model)

SCORM bileşenlerini tanımlamaktadır. Bunlar; Varlıklar (Asset), Paylaşılabilir İçerik Nesneleri (Sharable Content Object-SCO), İçerik Organizasyonlarıdır (Content Organization). Varlıklar, dijital olarak sunulan her türlü medya bileşenini, örneğin grafik, Flash dosyası, html dosyası, XML dosyası, ses ya da video dosyası gibi kullanıcıya sunulmak üzere hazırlanmış, web'te sunulacak her türlü nesneye verilen genel bir addır. Paylaşılabilir İçerik Nesneleri (SCO), LMS tarafından çalıştırılan, izlenen ve bir ya da birden fazla varlık ile oluşturulan yapıya verilen addır. Paylaşılabilir içerik nesneleri, LMS ile standartlar çerçevesinde iletişime geçen en küçük yapıdır. İçerik organizasyonu ise SCO'ların ve varlıkların bir araya getirilmesi ile

oluşturulmuş yapılara verilen isimdir. Yapılandırılmış bir e-öğrenme içeriğinin nasıl akacağını gösteren harita gibidir. İçerik organizasyonu, bir dersin bir bölümünü, bir dersi ya da komple bir eğitim paketini tanımlayabilir.

### **3.1.2 İçerik Paketi**

SCORM uyumlu bir içeriğin SCORM uyumlu bir LMS üzerinde çalışabilmesi için gerekli olan bütün dosyaları içerir. İçerik paketi içerisinde, “**manifest**” dosyası ve öğrenme kaynağını oluşturan bütün fiziksel dosyaların bulunması gerekir.

### **3.1.3 Tanımlama XML Dosyası (Manifest)**

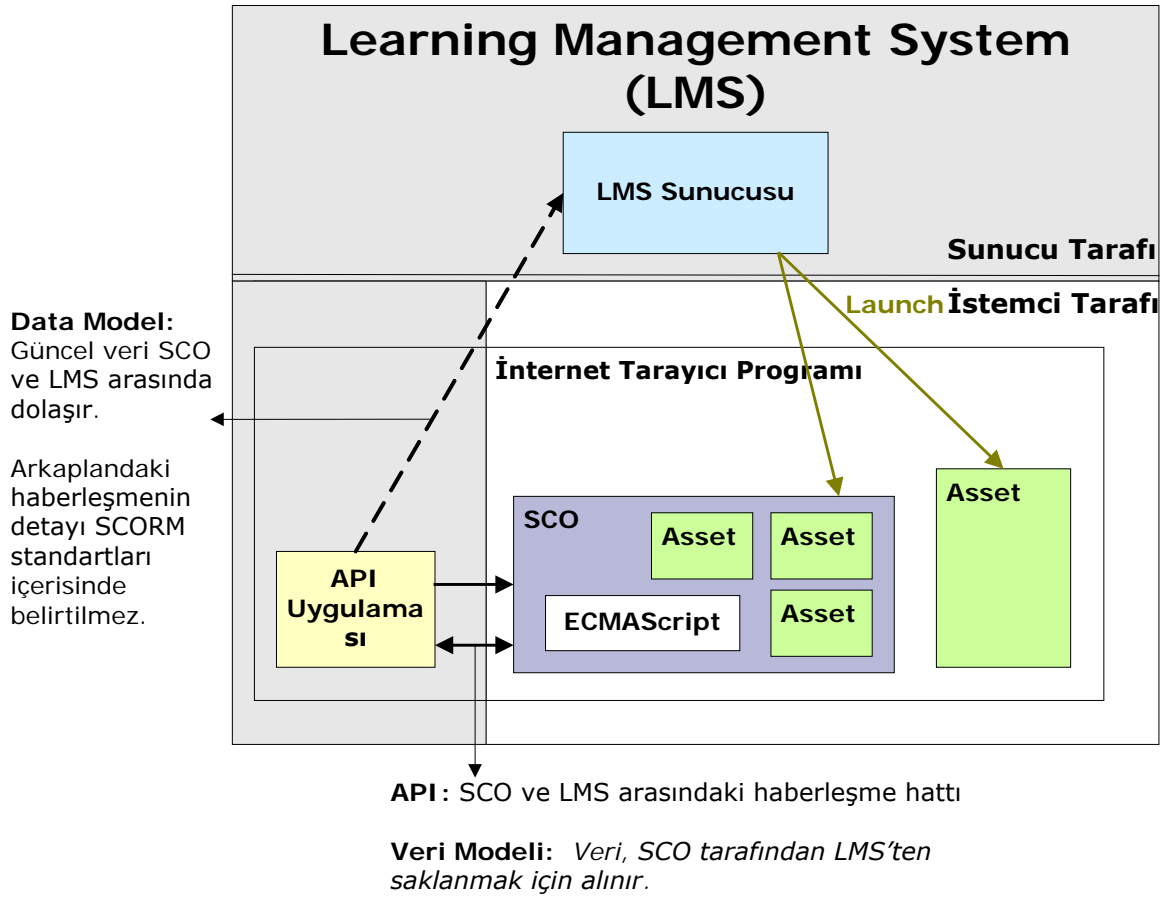
Oluşturulan SCORM Paketleri bir adet manifest dosyası ile içerik nesnelere dayanmaktadır. İçerisinde; içerik nesnelere tanımlayan Metadatalar, Aktiviteler ve Organizasyonlar ile varlıkları tanımlayan meta-datalar bulunmaktadır.

### **3.1.4 Üst-veri (Meta-data)**

Her bir içerik nesnesi için kimlik belgesi gibi düşünülebilir. Meta-data'lar, bir varlığın öğrenme içeriği içerisinde nerede ve nasıl kullanıldığına/kullanılacağına bakmaksızın, tanımlayıcı bilgileri taşır. Bu da, varlığın, yeniden kullanılabilir, aranınca kolay ulaşılabilir olmasını sağlayacak; içerik geliştiricilerinin işini ciddi boyutta kolaylaştıracak bir yöntemdir.

### 3.2 Çalışma Ortamı (Runtime Environment-RTE)

Kullanıcı ile öğretim materyalleri arasındaki haberleşmeyi çalışma ortamları sağlamaktadır. Bu çalışma ortamı içerisinde bulunan API (Application Programming Interface) dosyaları SCORM paketleri ile LMS arasındaki iletişimi sağlayan ECMAScript standartları ile yazılmış olan JavaScript dosyalarıdır. Core, student\_data, interactions, objectives, student\_preference gibi parametrelerden oluşmaktadır. RTE bu verilere nasıl ulaşabileceğini ve nasıl çalıştırabileceğini belirler. RTE ve LMS arasındaki iletişim Şekil 3.1 de verilmiştir.



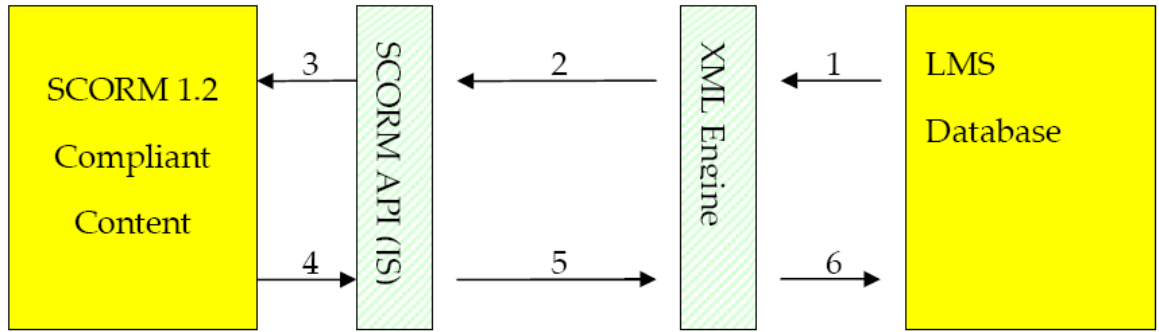
Şekil 3. 1 Öğrenim Yönetim Sistemleri ve API arasındaki iletişim(ADL 2004).

### 3.3 Sıralama ve Dolaşım (Sequencing and Navigation – SN)

Sıralama ve Dolaşım (Sequencing and Navigation – SN), öğrenme ihtiyaçlarına göre şekillendirilebilir dinamik e-öğrenme sunumlarının, öğrenci ya da sistem tarafından oluşturulabilir olması gerekliliğinden hareketle, içerik geliştiricileri tarafından tanımlanan sıralama kurallarının LMS tarafından nasıl gerçekleştirileceği tarif edilir. İçerik organizasyonları içerik modeli, içerik paketi ve üst-veri (Meta-Data) bölümlerinden oluşmaktadır. Verileri belirli bir düzene göre kümeleyerek kullanıcılara hangi sıra ile aktarılacağını gösteren aktivite ağaçlarının temelini oluştururlar.

### 3.4 SCORM 1.2 API Özellikleri

Scorm API, kullanıcı ile öğretim materyalleri arasındaki etkileşimi sağlayan ECMAScript standartları ile yazılmış olan JavaScript dosyalarıdır. Şekil 3.2 de LMS' e entegre edilen SCORM 1.2 API'nin çalışma yapısı gösterilmektedir (İnt. Kyn. 3).



Şekil 3. 2 SCORM 1.2 API' nin işleyişi.

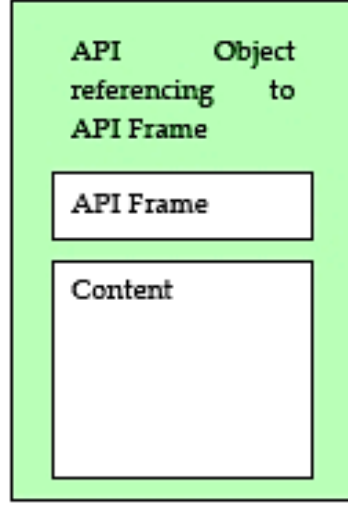
Şekil 3.2'de verilen SCORM 1.2 API' nin LMS ile arasındaki iletişim basamakları aşağıda verilmiştir.

1. **LMS – XML Motor İletişimi:** XML motoru LMS veritabanını belirli bir öğrencinin belirli bir dersi için sorgulayıp 2.adımda SCORM API' ye geçirecek olan XML

dokümanını yaratır. Bu bilgi SCORM 1.2 RTE içerisindeki tüm gerekli ve opsiyonel parametreleri içerir.

2. **XML Motoru – SCORM API İletişimi:** SCORM API XML motorunu belirli bir öğrencinin belirli bir dersi için sorgulayıp, birinci adımda bahsedilen iletişim vasıtasıyla anında kullanılabilecek olan XML dokümanını yaratır.
3. **SCORM API – İçerik İletişimi:** Dersin başlatılmasını takiben LMS API’yi de başka bir pencerede başlatır. İçerik API’yi ana çerçevelerde arayıp bulmaya çalışır. API çerçeveye ilk yerleştirildiğinde LMS tarafından ona bir öğrenci ID’si ve bir ders ID’si atanır. İçerik API’yi bulunca get istekleri (1, 2, 3) ve post istekleri (4, 5, 6) yapar.
4. **İçerik – SCORM API İletişimi:** 3. bölümde bahsedilenler geçerlidir. 3. bölümdeki iletişim çift yönlüdür. Kullanıcı içerikle etkileşime geçtiğinde, içerik veriyi API’ye geçirir. API bu veriyi içerikten Commit komutu alıncaya kadar saklar. Commit komutu alındığında API XML dokümanı hazırlayıp XML motoruna geçirir.
5. **SCORM API – XML Motoru İletişimi:** SCORM API 4. bölümde oluşturulan XML dokümanını LMS veritabanına yazar. XML dokümanı ID’lerle beraber yazılır.
6. **XML Engine – LMS İletişimi:** XML motoru 5. adımdaki XML dokümanını alır ve işler. Daha sonra veritabanındaki gerekli alanları günceller veya yeni kayıtlar yaratır.

LMS de API’ nin kullanıcı ile etkileşimini oluşturmak için Şekil 3.3’de de görüldüğü gibi iki çerçeveden oluşan bir ana çerçeve oluşturmuştur. İçeriği barındıran çerçeve %100 yükseklikteyken API’ yi barındıran çerçeve %0 yüksekliklidir. Bu şekilde kullanıcı çerçeve kullanıldığının farkına varmamaktadır. Ana çerçeve sayfası API isimli bir değişken barındırıp bunu API çerçevesindeki API objesine yönlendirir. API başlatılırken LMS öğrenci ID’ sini, ders ID’ sini ve LMS’in ihtiyaç duyacağı diğer ID leri API’ye atar. Bu değişiklikler LMS’e gönderilen XML dokümanına yansıtılır.



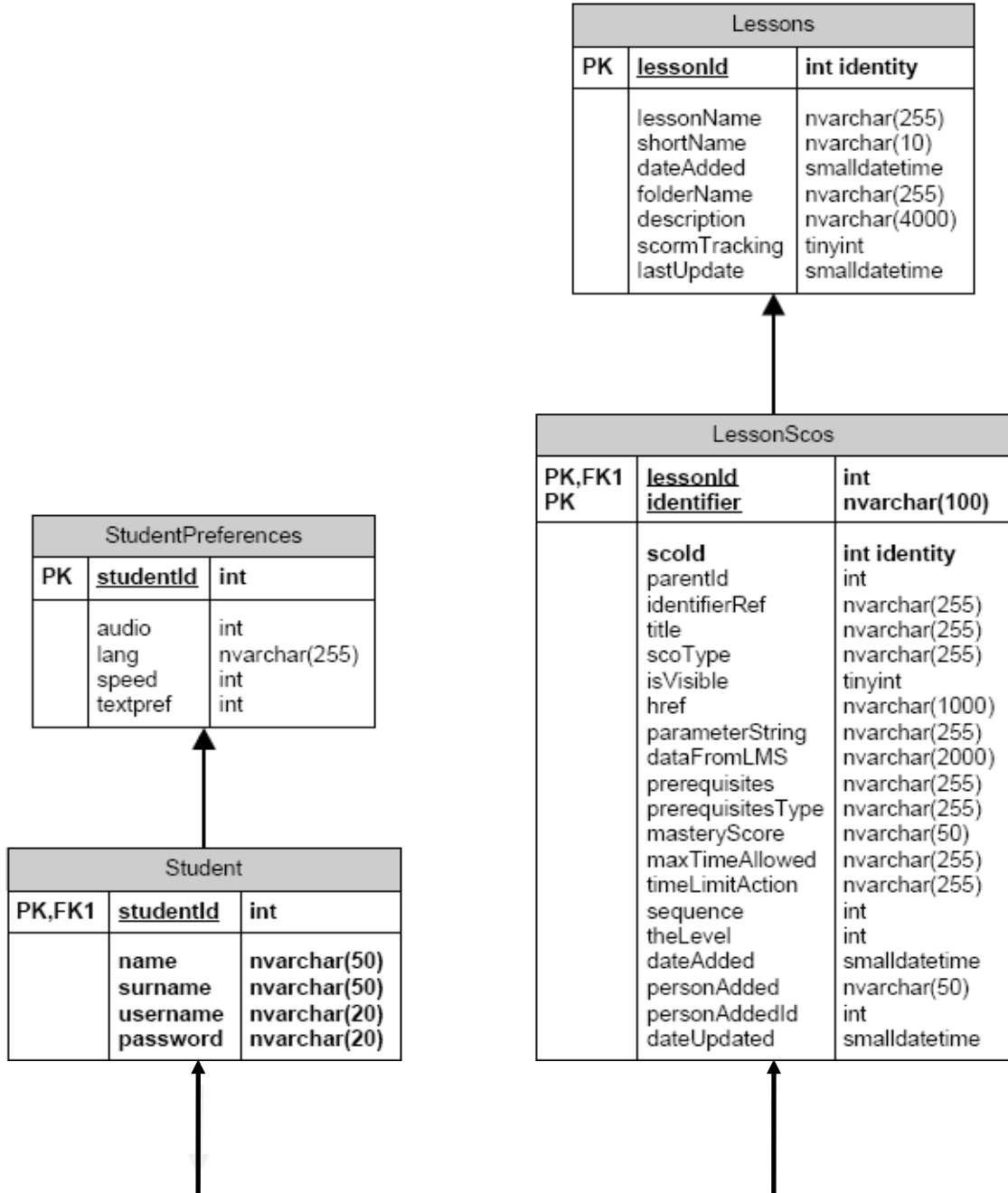
**Şekil 3. 3** API ve İçerik İletişimi.

#### **3.4.1 SCORM 1.2 API Veri Tabanı Yapısı**

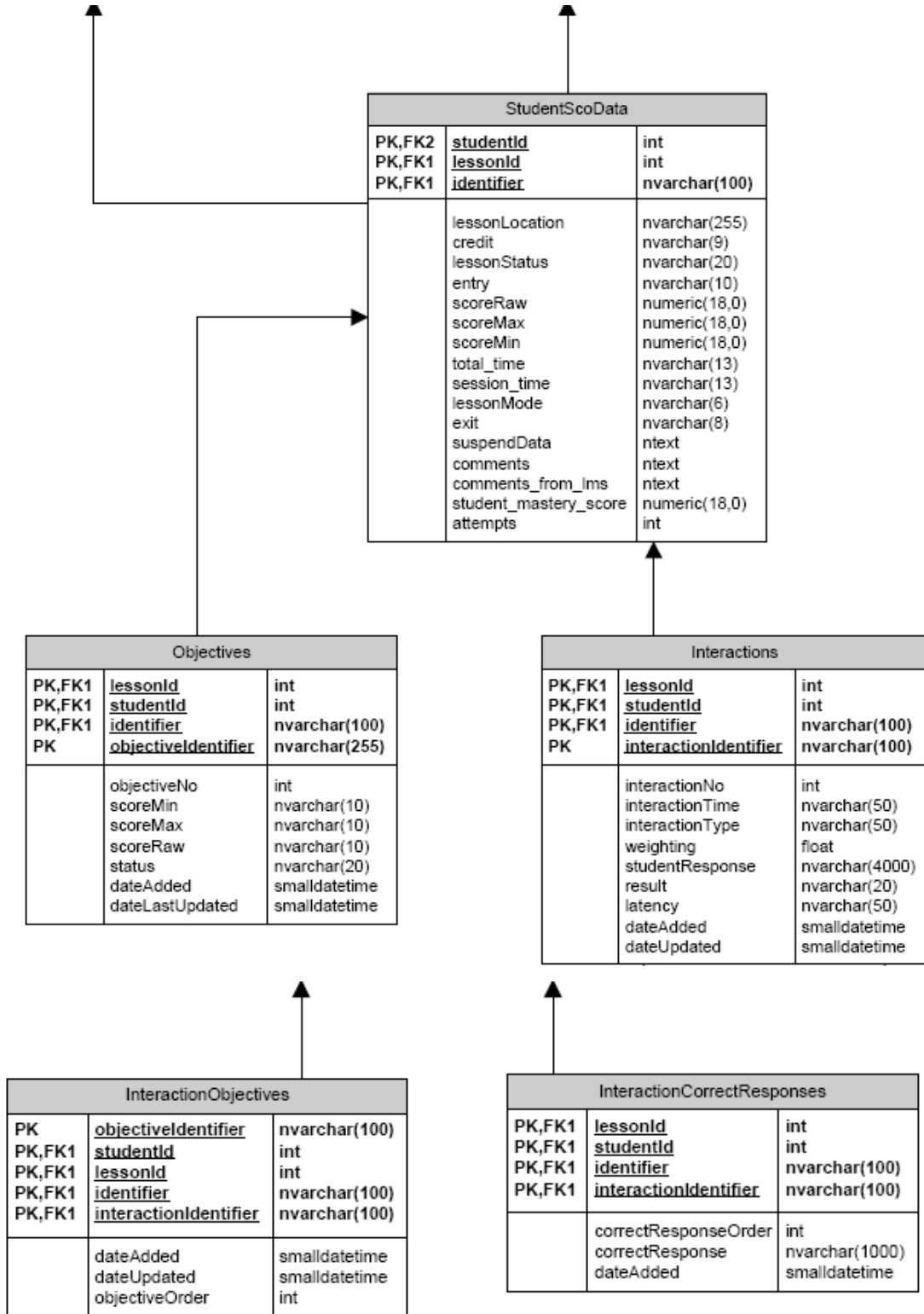
Çizelge 3.1 ve 3.2’ de SCORM 1.2 API için kullanılan SQL veri tabloları ve bu tablolarda tutulan değerlerle birlikte, ilişkili oldukları SCORM tabloları verilmiştir. Bu tablolar API’den gelen öğrenci, içerik, ders bilgileri gibi bilgileri kalıcı olarak saklamaktadır. Yani öğrenci, bir SCO’nun içeriğini görüntülemek için API’den istekte bulunur. API içeriği görüntülenmek istenen dersi veritabanında tutulan bilgilerden bulur ve LMS’ye iletir. LMS, SCO bilgilerini API aracılığı ile alır ve kullanıcı ekranına getirir. Kullanıcı, bu bilgileri başlatmak için API’ye iletişimi başlat mesajını gönderir. Bu mesajı alan API, veri modelini oluşturur. Öğrenci bilgileri, API’nin sunucu ile bağlantısını sağlayan kavramdan istenmektedir. Bu bilgiler sunucuda bulunan veri modeli bağlantı kavramından istenmektedir. Bu kavram veritabanı ile iletişime geçerek ara kavram olan API sunucu bağlantı kavramına iletmektedir. API bu kavramdan bilgileri alarak veri modelinde saklamaktadır. SCO ve API arasındaki iletişim için veri modeli kavramı kullanılmaktadır. SCO ve öğrenci ile ilgili bilgiler, “Derse ne zaman başlandı?”, “Ne zaman bitti?” gibi bilgiler veritabanında saklanmak yerine, geçici olarak veri modeli kavramında saklanmaktadır. SCO, API’ ye LMS ile iletişimi bitir mesajı gönderir, bu mesajı alan API, LMS ile iletişimi bitirir. Veri modelindeki bilgilerin veritabanına kayıt edilmesi için, LMS’ ye veri modelini gönderip, LMS’de

sunucudaki veri tabanına iletmekte ve iletişim sonlanmaktadır. LMS ile veri modelindeki bu bilgiler Çizelge 3.2’ de verilen tablolarda kalıcı olarak kayıt edilmektedir.

**Çizelge 3. 1** SCORM API 1.2 Veri Tabanı Yapısı.



Çizelge 3. 1 (Devam) SCORM API 1.2 Veri Tabanı Yapısı.





Scorm 1.2 API Scorm 1.2 versiyonuna uygun olarak bazı deęişkenleri tutmamızı istemektedir. Bu deęerler SCORM alıřma ortamında yer alan veri modelinin gelerini, izleme verileri olarak kullanılmaktadır. İzleme verileri sayesinde, bir đrenci derse ne zaman bařladı, ne zaman bitirdi, nerede kaldı gibi bilgiler elde edilebilmektedir. Bu bilgiler API tarafından izelge 3,2’de verilen veri modelinde tutulmaktadır. rneđin bu veri modelinden đrenci ID’ sini ađırmak iin “**CMI.core.student\_id**” parametresi kullanılır.

**izelge 3. 2 SCORM API 1.2 veri modeli.**

CMI Model	XML Model
<b>CORE</b>	
CMI.core.student_id	core\student_id
CMI.core.student_name	core\student_name
CMI.core.lesson_location	core\lesson_location
CMI.core.credit	core\credit
CMI.core.lesson_status	core\lesson_status
CMI.core.lesson_mode	core\lesson_mode
CMI.core.entry	core\entry
CMI.core.total_time	core\total_time
CMI.core.session_time	core\session_time
CMI.core.exit	core\exit
CMI.core.score.min	core\score\min
CMI.core.score.max	core\score\max
CMI.core.score.raw	core\score\raw
<b>OTHER</b>	
CMI.suspend_data	suspend_data
CMI.launch_data	launch_data
CMI.comments	comments
CMI.comments_from_lms	comments_from_lms
<b>STUDENT DATA</b>	
CMI.student_data.mastery_score	student_data\mastery_score
CMI.student_data.max_time_allowed	student_data\max_time_allowed
CMI.student_data.time limit action	student_data\time limit action
<b>STUDENT PREFERENCE</b>	
CMI.student_preference.audio	student_preference\audio
CMI.student_preference.language	student_preference\language
CMI.student_preference.speed	student_preference\speed
CMI.student_preference.text	student_preference\text
<b>OBJECTIVES</b>	
CMI.objectives._count	\objectives\count
CMI.objectives.n.id	\objectives\n[index]\id
CMI.objectives.n.score.raw	\objectives\n[index]\score\raw
CMI.objectives.n.score.max	\objectives\n[index]\score\max
CMI.objectives.n.score.min	\objectives\n[index]\score\min
CMI.objectives.n.status	\objectives\n[index]\status
<b>INTERACTIONS</b>	
CMI.interactions._count	\interactions\count
CMI.interactions.n.id	\interactions\n[index]\id
CMI.interactions.n.objectives._count	\interactions\n[index]\objectives\count
CMI.interactions.n.objectives.n.id	\interactions\n[index]\objectives\n[index]\id
CMI.interactions.n.time	\interactions\n[index]\time
CMI.interactions.n.type	\interactions\n[index]\type
CMI.interactions.n.correct_responses._count	\interactions\n[index]\correct_responses\count
CMI.interactions.n.correct_responses.n.pattern	\interactions\n[index]\correct_responses\n[index]\pattern
CMI.interactions.n.weighting	\interactions\n[index]\weighting
CMI.interactions.n.student_response	\interactions\n[index]\student_response
CMI.interactions.n.result	\interactions\n[index]\result
CMI.interactions.n.latency	\interactions\n[index]\latency

**Çizelge 3.2(Devam) SCORM API 1.2 veri modeli.**

CMI Model	XML Model
<b>CORE</b>	
CMI.core.student_id	core\student_id
CMI.core.student_name	core\student_name
CMI.core.lesson_location	core\lesson_location
CMI.core.credit	core\credit
CMI.core.lesson_status	core\lesson_status
CMI.core.lesson_mode	core\lesson_mode
CMI.core.entry	core\entry
CMI.core.total_time	core\total_time
CMI.core.session_time	core\session_time
CMI.core.exit	core\exit
CMI.core.score.min	core\score\min
CMI.core.score.max	core\score\max
CMI.core.score.raw	core\score\raw
<b>OTHER</b>	
CMI.suspend_data	suspend_data
CMI.launch_data	launch_data
CMI.comments	comments
CMI.comments_from_lms	comments from lms
<b>STUDENT DATA</b>	
CMI.student_data.mastery_score	student_data\mastery_score
CMI.student_data.max_time_allowed	student_data\max_time_allowed
CMI.student_data.time limit action	student_data\time limit action
<b>STUDENT PREFERENCE</b>	
CMI.student_preference.audio	student_preference\audio
CMI.student_preference.language	student_preference\language
CMI.student_preference.speed	student_preference\speed
CMI.student_preference.text	student_preference\text
<b>OBJECTIVES</b>	
CMI.objectives._count	\objectives\count
CMI.objectives.n.id	\objectives\n[index]\id
CMI.objectives.n.score.raw	\objectives\n[index]\score\raw
CMI.objectives.n.score.max	\objectives\n[index]\score\max
CMI.objectives.n.score.min	\objectives\n[index]\score\min
CMI.objectives.n.status	\objectives\n[index]\status
<b>INTERACTIONS</b>	
CMI.interactions._count	\interactions\count
CMI.interactions.n.id	\interactions\n[index]\id
CMI.interactions.n.objectives._count	\interactions\n[index]\objectives\count
CMI.interactions.n.objectives.n.id	\interactions\n[index]\objectives\n[index]\id
CMI.interactions.n.time	\interactions\n[index]\time
CMI.interactions.n.type	\interactions\n[index]\type
CMI.interactions.n.correct_responses._count	\interactions\n[index]\correct_responses\count
CMI.interactions.n.correct_responses.n.pattern	\interactions\n[index]\correct_responses\n[index]\pattern
CMI.interactions.n.weighting	\interactions\n[index]\weighting
CMI.interactions.n.student_response	\interactions\n[index]\student_response
CMI.interactions.n.result	\interactions\n[index]\result
CMI.interactions.n.latency	\interactions\n[index]\latency

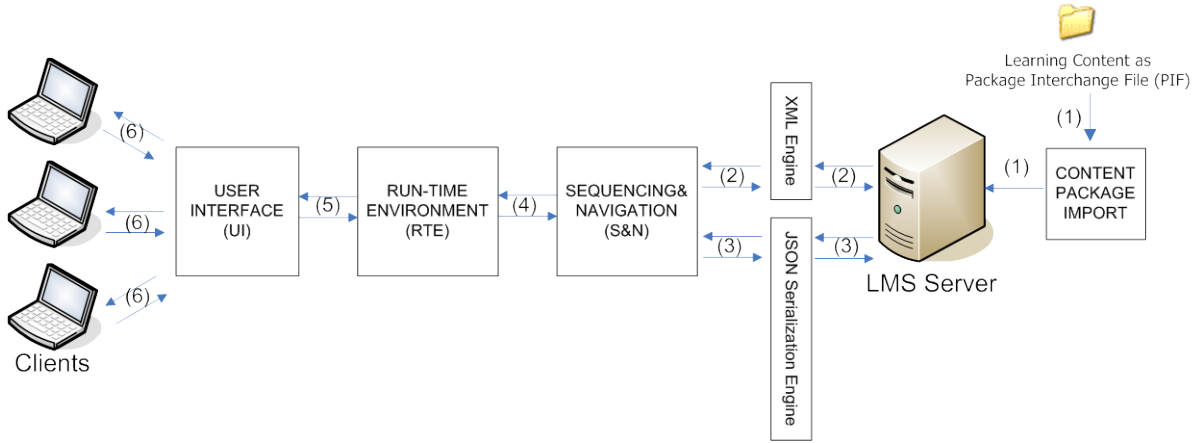
Çizelge 3.2(Devam) SCORM API 1.2 Veri Modeli.

Database Table	Database Field
Student Student StudentScoData StudentScoData StudentScoData StudentScoData StudentScoData StudentScoData StudentScoData StudentScoData StudentScoData StudentScoData	id name lesson_location credit lessonStatus lessonMode entry total_time session_time exit scoreMin scoreMax scoreRaw
StudentScoData LessonScos StudentScoData LMS Specific!	suspend_Data dataFromLMS comments
LessonScos LessonScos LessonScos	masteryScore maxTimeAllowed timeLimitAction
StudentPreference StudentPreference StudentPreference StudentPreference	audio language speed text
Calculated on the fly Objectives Objectives Objectives Objectives Objectives	objectiveIdentifier scoreRaw scoreMax scoreMin status
Calculated on the fly Interactions Calculated on the fly InteractionObjectives Interactions Interactions Calculated on the fly InteractionCorrectResponses Interactions Interactions Interactions Interactions	interactionIdentifier  objectiveIdentifier interactionTime interactionType  correctResponse weighting studentResponse result latency

### 3.5 SCORM 2004 API'nin Özellikleri

#### 3.5.1 SCORM 2004 API Çalışma Şekli

Şekil 3.4'de LMS'ye entegre edilen SCORM 1.2 API'nin çalışması basitçe gösterilmektedir.



Şekil 3. 4 SCORM 2004 API' nin işleyişi.

Şekil 3.4'de verilen SCORM 2004 ile LMS arasındaki iletişim basamakları aşağıda ayrıntılı olarak verilmiştir (İnt. Kyn. 3).

1. **İçerik Paketi Alma:** PIF formatında hazırlanmış bir içerik paketi LMS sunucusuna İçerik Paketi Alma Modülü tarafından alınır. Bu andan itibaren içerik kullanım için yayınlanmaya hazırdır.
2. **XML Motoru:** İstemciler ve LMS sunucusu arasındaki veritabanı sorguları gönderme ve sonuçları alma iletişimi XML Motoru üzerinden yapılır.
3. **JSON Serileştirme Motoru:** İstemciler ve LMS sunucusu arasındaki karmaşık veri yapıları (eğitim içeriği aktivite ağacı ve öğrencinin veri takibi gibi) JSON serileştirme motoru üzerinden yapılır. Buna göre serileştirilmiş aktivite ağacı ve veri takibi LMS sunucusunda JSON dizgileri halinde tutulur.

4. **RTE-S&N Etkileşimi:** RTE ve S&N modülleri öğrencinin öğrenim tecrübesi üzerinden uygun içeriği sunmak için SCORM® 2004 3rd Edition Sequencing and Navigation Version 1.0'e uygun olarak etkileşirler.
5. **UI-RTE Etkileşimi:** Öğrenim içeriği içerisindeki navigasyon istekleri Kullanıcı Arayüzü (UI) tarafından RTE'ye gönderilip, RTE tarafından SCORM® 2004 3rd Edition Run-Time Environment Version 1.0'e uygun biçimde işlenip uygun takip bilgisi yaratılır.
6. **İstemci-UI Etkileşimi:** Her istemci öğrenim içeriğini başlatıp içerisinde UI üzerinden hareket edebilir.

### 3.5.2 SCORM\_2004 VERİTABANI TASARIMI

SCORM 2004 veri tabanı yapısı SCORM 1.2 veri tabanı yapısından biraz farklıdır. Bu veri tablolarında tüm SCORM tabloları “SCORM\_” ön eki kullanılarak isimlendirilmiştir. Bazı tablolar ise “SCORM\_OBJ\_” ön eki kullanılarak isimlendirilmiştir. “SCORM\_OBJ\_” ön ekine sahip tabloların hepsi, API tarafından EYS’ye (LMS) gönderilen run-time ve sequencing verilerini tutmak üzere tasarlanmıştır. Bu veriler esasen “/SCORM\_sf” dizini altında tutulmaktadır. Dosyadaki veriler de veritabanına eş zamanlı olarak kaydedilir.

Tabloların çoğunda bulunan COURSEID alanı, SCORM\_COURSEINFO tablosunun Primary Key alanı olan COURSEID gibi SCORM 2004 paketi olarak kurulan bir dersi tanımlamaktadır. Yine tabloların çoğunda bulunan LEARNERID alanı ise EYS tarafında bir kullanıcıyı tanımlayan eşsiz bir değerdir. Örnek vermek gerekirse EYS veritabanındaki Student tablosunun StudentID alanı LEARNERID olabilir.

Bu tablolar kullanılarak yazılacak SQL sorgularında, COURSEID ve LEARNERID alanları tabloları birleştirmede kullanılmalıdır. Örneğin aşağıdaki örnek sorgu hangi öğrencinin hangi SCORM derslerini bitirdiğini gösterir.

```
SELECT U.LEARNERID, C.COURSEID FROM SCORM_COURSEINFO C INNER  
JOIN SCORM_USERINFO U ON C.COURSEID = U.COURSEID INNER JOIN  
SCORM_COURSESTATUS S ON S.LEARNERID = U.LEARNERID AND  
S.COURSEID = U.COURSEID WHERE S.COMPLETED = 'completed'
```

### 3.5.3 SCORM 2004’ de Veri Tabanı Tabloları

Aşağıda veri tabanı tabloları ayrıntılı olarak açıklanmıştır.

### 3.5.3.1 SCORM\_APPLICATIONDATA Tablosu

SCORM\_APPLICATIONDATA tablosu çeşitli metin ve sayı değerlerini tutmak için oluşturulmuştur. SCORM\_APPLICATIONDATA tablosunda şimdilik sadece, yeni kurulacak dersin ID'si tutulmaktadır. Dataname alanında nextCourseId, NUMBERVALUE kısmında ise ilk kurulumda 1 yazar. Her SCORM 2004 paketi kurulmasında nextCourseId değeri 1 artırılır. Bu değer elle değiştirilmemesi gerekmektedir. Oluşturulan bu tablo Çizelge 3.3 de verilmiştir.

**Çizelge 3. 3** SCORM\_APPLICATIONDATA tablosu.

SCORM_APPLICATIONDATA	
🔑	DATANAME
	TEXTVALUE
	NUMBERVALUE

Çizelge 3.3'teki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- DATANAME: Uygulama ayar anahtarıdır.
- TEXTVALUE: Uygulama ayar anahtarı veri tipi metin ise değeri burada tutulur.
- NUMBERVALUE: Uygulama ayar anahtarı veri tipi sayı (int) ise değeri burada tutulur.

### 3.5.3.2 SCORM\_COURSEINFO Tablosu

Kurs ID'si, kurs başlığı, kursun aktif olup olmadığı, kurs paketinin sisteme kurulma tarihi bilgisini tutan tablo, Çizelge 3.4'de verilmiştir. Bu tabloda TOC alanının değerinin 1 olması içindekiler tablosunun gösterilmesi, içindekiler tablosunu göster butonunun olması gerektiğini, START değerinin 1 olması ise kursa başla butonunun

konabileceğini gösterir. Başla butonuna basıldığında kişi içindekiler tablosunda bir aktivite seçmeye başlamadan bir ders sayfası açılarak kurs başlar.

**Çizelge 3. 4 SCORM\_COURSEINFO Tablosu.**

SCORM_COURSEINFO	
?	COURSEID
	COURSETITLE
	ACTIVE
	IMPORTDATETIME
	START
	TOC

Çizelge 3.4'teki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- COURSEID: SCORM 2004 kurs tanımlayıcısıdır. Course-### formatındadır.
- COURSETITLE: Kursun adıdır.
- ACTIVE: Kursun aktif/pasif olma durumudur.
- IMPORTDATETIME: SCORM 2004 paketinin sisteme aktarıldığı tarihtir.
- START: Bu alanda kurs başlangıç sayfasının gösterilip gösterilmeyeceği bilgisi "true" ya da "false" olarak tutulur.
- TOC: Bu alanda içindekiler menüsünün gösterilip gösterilmeyeceği bilgisi "true" ya da "false" olarak tutulur.

### 3.5.3.3 SCORM\_COURSESTATUS Tablosu

Öğrencinin kurstaki durumunun tutulduğu SCORM\_COURSESTATUS veri tablosu, Çizelge 3.5 de verilmiştir. SCORM\_COURSESTATUS tablosu öğrenci dersi ilk almaya başladığında SATISFIED, COMPLETED ve MEASURE alanlarına 'unknown' değeri yazılır. Dersteki ilerleme durumuna göre bu alanların alabileceği değerler aşağıdaki gibidir.



SATISFIED = {unknown, satisfied, notSatisfied}

COMPLETED = {unknown, completed, incomplete}

-1.0 <= MEASURE <= 1.0

**Çizelge 3.5** SCORM\_COURSESTATUS tablosu.

SCORM_COURSESTATUS	
🔑	COURSEID
🔑	LEARNERID
	SATISFIED
	MEASURE
	COMPLETED






Çizelge 3.5'teki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- COURSEID: Kurs tanımlayıcısıdır.
- LEARNERID: Kullanıcı tanımlayıcısıdır.
- SATISFIED: Kurstaki hedefler (objectives) gerçekleştirildiyse "true" değilse "false" değerini alır.
- MEASURE: Kurstan alınan nottur. 0 ile 1 arasında rasyonel bir sayıdır.
- COMPLETED: Kurs tamamlanmışsa "true" tamamlanmamışsa "false" değerini alır.

### 3.5.3.4 SCORM\_ITEMINFO Tablosu

SCORM paketinde bulunan imsmanifest.xml dosyasındaki her SCO ile ilgili bilgi, SCORM paketi sisteme kurulduğunda bu tabloya kaydedilir. Bu tablo Çizelge 3.6'da verilmiştir.

**Çizelge 3. 6** SCORM\_ITEMINFO tablosu.

SCORM_ITEMINFO	
	COURSEID
	ORGANIZATIONIDENTIFIER
	ITEMIDENTIFIER
	RESOURCEIDENTIFIER
	LAUNCH
	TYPE
	TITLE
	PARAMETERSTRING
	PERSISTSTATE
	DATAFROMLMS
	MINNORMALIZEDMEASURE
	ATTEMPTABSOLUTEDURATIONLIMIT
	TIMELIMITACTION
	COMPLETIONTHRESHOLD
	NEXT
	PREVIOUS
	[EXIT]
	EXITALL
	ABANDON
	ACTIVITYID
	SUSPEND

Çizelge 3.6'daki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- COURSEID: Kurs tanımlayıcısıdır.
- ORGANIZATIONIDENTIFIER: İmsmanifest dosyasında kursun organizationIdentifier verisidir.
- ITEMIDENTIFIER: imsmanifest dosyasında SCO'ya ait olan ItemIdentifier verisidir.
- RESOURCEIDENTIFIER: SCO nesnesine ait kaynak dosyasının tanımlayıcısıdır (imsmanifest dosyasında <resources> bölümü altındadır).
- LAUNCH: SCO'nun başlatılması için açılacak olan web sayfası adresidir.
- TYPE: Kayıt tipidir (sco, asset).
- TITLE: SCO'nun başlığıdır ve içindekiler ağacında gösterilir.
- PARAMETERSTRING: SCO açılış sayfası açılırken sayfa adresinin sonuna eklenen parametre dizisidir.

- PERSISTSTATE: SCORM 2004 3rd Ed. ile beraber kaldırılmıştır ve kullanılmamaktadır.
- DATAFROMLMS: SCO tarafından ihtiyaç duyulan bir veridir. ParameterString içinde gönderilmez ve API aracılığıyla (cmi.launch\_data) okunmak istenir.
- MINNORMALIZEDMEASURE: Eğer SCO için satisfiedByMeasure değeri "true" olarak belirlenmişse SCO ile ilgili hedefe ulaşıldığını (objective) belirlemede kullanılan ölçüt değeridir.
- ATTEMPTABSOLUTEDURATIONLIMIT: Bir kullanıcının bir aktivite (SCO) üzerinde harcayabileceği maksimum süredir. Runtime tarafında cmi.max\_time\_allowed değişkeninin ilklendirilmesinde kullanılır.
- TIMELIMITACTION: İzin verilen maksimum süre dolduğunda LMS tarafından yapılması gereken işlemi bildiren değer bu alanda tutulur. Olası değerler: "exit, message", "exit, no message", "continue, message" ve "continue, no message" gibidir.
- COMPLETIONTHRESHOLD: SCO'nun tamamlanma bilgisinin belirlenmesini sağlayan eşik değeridir, 0 ile 1 arasında rasyonel bir sayıdır.
- NEXT: "İleri butonu gösterilsin mi?" true ya da false değerini alır.
- PREVIOUS: "Geri butonu gösterilsin mi?" true ya da false değerini alır.
- [EXIT]: "Çıkış butonu gösterilsin mi?" true ya da false değerini alır.
- EXITALL: "Tüm kurstan çıkış butonu gösterilsin mi?" true ya da false değerini alır.
- ABANDON: "İptal butonu gösterilsin mi?" true ya da false değerini alır.
- ACTIVITYID: Aktivite tanımlayıcı sayısıdır.
- SUSPEND: Derse ara verildiyse true, ilk defa giriliyor ya da çıkış yapıp giriliyorsa false değerini taşır.

### 3.5.3.5 SCORM\_SCOCOMMENTS Tablosu

SCO ile ilgili not tutma için tasarlanmış bir tablodur. Ancak mevcut yapıda kullanılmamaktadır. İleriki aşamalar için saklanmaktadır. Bu tablo, Çizelge 3.7'de verilmiştir.

**Çizelge 3. 7 SCORM\_SCOCOMMENTS tablosu.**

SCORM_SCOCOMMENTS	
🔑	ACTIVITYID
🔑	COMMENTID
	COMMENT
	COMMENTDATETIME
	COMMENTLOCATION

Çizelge 3.7'deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- **ACTIVITYID:** SCORM\_ItemInfo tablosundaki ActivityId ile aynıdır. Aktivite tanımlayıcı sayısıdır.
- **COMMENT:** SCO (aktivite) ile ilgili yorumların tutulduğu alandır.
- **COMMENTDATETIME:** Yorumun yazılma tarihidir.
- **COMMENTID:** Yorum tanımlayıcı sayısıdır.
- **COMMENTLOCATION:** Yorumun yazıldığı yer ile ilgili bilgi amaçlı tutulan alandır.

### **3.5.3.6 SCORM\_USERCOURSEINFO Tablosu**

Hangi kullanıcının hangi SCORM 2004 kursuna kayıtlı olduğunu gösteren tablodur. Bu tablo Çizelge 3.8'de verilmiştir. Bu tablodaki USERID, sistemin entegre edildiği LMS'teki kullanıcı numarasıdır. SUSPENDALL değeri true ya da false olarak girilir. Ara ver (Suspend) butonuna basılarak ara verilen dersler için bu alanda true yazar. Bu sayede LMS, dersin kaldığı yerden başlatılacağını anlayarak Derse Devam Et şeklinde buton gösterir.

**Çizelge 3. 8** SCORM\_USERCOURSEINFO tablosu.

SCORM_USERCOURSEINFO	
USERID	
COURSEID	
SUSPENDALL	

Çizelge 3.8'deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- USERID: Kullanıcı tanımlayıcısı değeridir (diğer tablolardaki LEARNERID ile aynıdır).
- COURSEID: Kurs tanımlayıcısı değeridir (USERID ile beraber PRIMARY KEY oluştururlar).
- SUSPENDALL: Kullanıcı eğer Ara ver butonuna tıklayarak kursa ara verdiyse “true” değilse “false” değerini taşır.

### 3.5.4 SCORM 2004'TE OBJECT TABLOLARI

SCORM 2004 Runtime API, sıralama (sequencing), navigasyon (navigation) ve çalışma zamanı (Runtime) verilerini JSON nesnelere olarak eğitim yönetim sistemine gönderir. EYS tarafından bu veriler okunarak aşağıda bahsedilen tablolara girilir.

#### 3.5.4.1 SCORM\_OBJ\_ADVSCORM\_2004\_DMELEMENT Tablosu

API'de AdvSCORM\_2004\_DMELEMENT isimli sınıftan türeyen her veri modeli nesnesi ile ilgili bilgiler bu tabloda her kullanıcı ve her kurs için ayrı ayrı tutulur. AdvSCORM\_2004\_DMELEMENT tablosu, Çizelge 3.9'da verilmiştir.

**Çizelge 3. 9** AdvSCORM\_2004\_DMELEMENT tablosu.

SCORM_OBJ_ADVSCORM_2004_DMELEMENT	
	DM
	ID
	PARENT
	INITIALIZED
	VALUE
🔑	KEYNAME
🔑	COURSEID
🔑	LEARNERID
🔑	FILENAME
🔑	HASHKEYNAME

Çizelge 3.9'daki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- DM: Veri modeli nesnesi adıdır.
- ID: Veri modeli nesnesinin ID'sidir.
- PARENT: Elamanın bağlı olduğu üst seviye veri modeli nesnesi ID'sidir.
- INITIALIZED: Veri modeli nesnesinin ilklendirilip ilklendirilmediğini belirtir. True/false değerini alır.
- VALUE: Veri modeli nesnesinin değeridir.
- KEYNAME: *Hashtable* veri yapısı içinde veri modeli nesnesine ulaşmak için kullanılan anahtarın ismidir. JSON nesnesi olarak gönderilen veriler, *hashtable* olarak serialize edilerek gönderilir.
- COURSEID: SCORM kurs ID'sidir.
- LEARNERID: Kullanıcı ID'sidir.
- FILENAME: SCO ID'sidir.
- HASHKEYNAME: JSON verisi içinde veri modeli nesnesinin değerini almak için kullanılan anahtarların ismidir. KEYNAME alanı ile ilişkilidir.

### 3.5.4.2 SCORM\_OBJ\_ADVDMELEMENTDESCRIPTOR Tablosu

API'den gelen advDMElementDescriptor isimli sınıftan türeyen nesnelerin bilgilerinin tutulduğu tablodur. Her veri modeli (advSCORM\_2004\_DMElement) için bir açıklayıcı nesne (advDMElementDescriptor) bulunmaktadır. Dolayısıyla bir kullanıcının bir SCORM 2004 kursundaki bir aktiviteyi izlemesi sırasında tutulmuş veriler bu tablo ve “SCORM\_OBJ\_ADVSCORM\_2004\_DMELEMENT” tablosu kullanılarak yazılan sorgularla raporlanabilir. Bu tablo, Çizelge 3.10’da verilmiştir.

**Çizelge 3. 10** SCORM\_OBJ\_ADVDMELEMENTDESCRIPTOR tablosu.

SCORM_OBJ_ADVDMELEMENTDESCRIPTOR	
	BINDING
	ISREADABLE
	ISWRITABLE
	DELIMITERS
	INITIAL
	DEPENDENTON
	ISUNIQUE
	WRITEONCE
	VALUESPM
	SPM
	OLDSPM
	MAXIMUM
	SHOWCHILDREN
	VALIDATOR
🔑	KEYNAME
🔑	COURSEID
🔑	LEARNERID
🔑	FILENAME
🔑	HASHKEYNAME

Çizelge 3.10’daki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- **BINDING:** Veri modeli nesnesinin “.” (nokta) ile ayrılmış elemanlarının sonuncusudur. cmi.suspend\_data için suspend\_data verisidir.

- ISREADABLE: Veri modeli nesnesinin okunabilir olup olmadığını belirtir. “True” yada “False” değerini alır.
- ISWRITABLE: Veri modeli nesnesinin yazılabilir olup olmadığını belirtir. “True” ya da “False” değerini alır.
- DELIMITERS: Veri modeli nesnesinin sınırlayıcı bilgisidir.
- INITIAL: Veri modeli nesnesinin ilk değeridir.
- DEPENDENTON: Veri modeli nesnesinin bağlı olduğu nesnedir.
- ISUNIQUE: Veri modeli nesnesinin eşsiz olup olmadığını bilgisidir.
- WRITEONCE: Veri modeli nesnesinin bir kere yazılıp yazılamayacağı bilgisidir.
- VALUESPM: Karakter tipi için en küçük izin verilen maksimum değeridir (Smallest permitted maximum).
- SPM: Sayılar için en küçük izin verilen maksimum değeridir (Smallest permitted maximum).
- OLDSPM: Önceki SPM değeridir.
- MAXIMUM: Veri modeli nesnesinin maksimum değeridir.
- SHOWCHILDREN: Alt bağlı nesnelerin gösterilip gösterilmemesi ile ilgilidir.
- VALIDATOR: Veri modeli nesnesinin validasyon tanımıdır.
- KEYNAME: Bkz. SCORM\_OBJ\_ADVSCORM\_2004\_DMELEMENT
- COURSEID: Kurs ID değeridir.
- LEARNERID: Kullanıcı ID değeridir.
- FILENAME: SCO nesnesi tanımlayıcısı (SCO Item identifier).
- HASHKEYNAME:  
Bkz. SCORM\_OBJ\_ADVSCORM\_2004\_DMELEMENT.HASHKEYNAME



### 3.5.4.3 SCORM\_OBJ\_ADLTOC tablosu

Sisteme eklenen SCORM 2004 İçerik paketindeki dersin, içindekiler kısmının tutulduğu tablodur. Bu tablo, tablo Çizelge 3.11’de verilmiştir.

**Çizelge 3. 11** SCORM\_OBJ\_ADLTOC tablosu.

SCORM_OBJ_ADLTOC	
	TITLE
	DEPTH
	COUNT
	LEAF
	PARENT
	INCHOICE
	ISENABLED
	ISVISIBLE
	ISCURRENT
	ISSELECTABLE
	ID
🔑	KEYNAME
🔑	COURSEID
🔑	LEARNERID
🔑	FILENAME
🔑	HASHKEYNAME
	DATAKEYNAME

Çizelge 3.11’deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- TITLE: SCO nesnesi başlığıdır.
- DEPTH: İçindekiler ağacındaki seviyesidir.
- COUNT: Tıklanma sayısıdır.
- LEAF: Alt aktivite ise TRUE, değilse FALSE değeri alır.
- PARENT: Başlık ise -1, alt aktivite ise 0 değeri alır.
- INCHOICE: Seçilebilir olup olmadığı bilgisidir.
- ISENABLED: Etkin olup olmadığı bilgisidir.
- ISVISIBLE: Görünür olup olmadığı bilgisidir.

- ISCURRENT: Son seçili aktivite ise True, değilse False değerini alır.
- ISSELECTABLE: Seçilebilir olup olmadığını belirtir.
- ID: Aktivite SCO ID' sidir.
- KEYNAME: Bkz. SCORM\_OBJ\_ADVSCORM\_2004\_DMELEMENT.KEYNAME
- COURSEID: Kurs ID' sidir.
- LEARNERID: Kullanıcı ID' sidir.
- FILENAME: Obj dosyasının ismidir (Örn: serialize).
- HASHKEYNAME: Aktivite ID değerini alır.
- DATAKEYNAME: Veri modeli anahtarı ismidir.

#### 3.5.4.4 SCORM\_OBJ\_ADLVALIDREQUESTS Tablosu

Bu tablo, API tarafından gönderilen veri nesnesinin veritabanında düzgün tutulması için gerekli bir tablodur. Bu tablo ile UI den gelen başla, devam, ilerle, çıkış, önceki, sonlandır gibi komutları tutan tablodur. Kurs ile ilgili yapılabilecek navigasyon isteklerini tutar SCORM\_OBJ\_ADLVALIDREQUESTS tablosu, Çizelge 3.12'de verilmiştir.

**Çizelge 3. 12** SCORM\_OBJ\_ADLVALIDREQUESTS tablosu.

SCORM_OBJ_ADLVALIDREQUESTS	
	START
	RESUME
	[CONTINUE]
	CONTINUEEXIT
	PREVIOUS
	SUSPEND
🔑	KEYNAME
🔑	COURSEID
🔑	LEARNERID
🔑	FILENAME

Çizelge 3.12'deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- START: Kursa başlama komutu verilip verilemeyeceği bilgisidir.
- RESUME: Kursa kaldığı yerden devam edip edememe bilgisidir.
- CONTINUE: Kursta SCO'lar arasında devam ederek gezinebilme bilgisidir.
- CONTINUEEXIT: Kursa devam ederek son SCO'da ayrıca çıkış yapabilme bilgisidir.
- PREVIOUS: Önceki SCO'yu açıp açamama bilgisidir.
- SUSPEND: Kurstan sonra kaldığı yerden devam edecek şekilde çıkıp çıkamama bilgisidir.
- KEYNAME: İlgili JSON nesnesinin adıdır.
- COURSEID: Kurs ID' sidir.
- LEARNERID: Kullanıcı ID' sidir.
- FILENAME: Obj dosyasının ismidir (serialize).

### 3.5.4.5 SCORM\_OBJ\_ADVADLTRACKING Tablosu

Kullanıcı takip verileri bu tabloda tutulur. Bu tablo, API tarafından gönderilen veri nesnesinin veritabanında düzgün tutulması için gerekli bir tablodur. Kullanıcı takip verileri bu tabloda tutulur. Bu tablo, Çizelge 3.13'de verilmiştir.

**Çizelge 3. 13** SCORM\_OBJ\_ADVADLTARACKING tablosu.

SCORM_OBJ_ADVADLTRACKING	
<input type="checkbox"/>	TRACK_UNKNOWN
<input type="checkbox"/>	TRACK_SATISFIED
<input type="checkbox"/>	TRACK_NOTSATISFIED
<input type="checkbox"/>	TRACK_COMPLETED
<input type="checkbox"/>	TRACK_INCOMPLETE
<input type="checkbox"/>	DIRTYPRO
<input type="checkbox"/>	PRIMARYOBJ
<input type="checkbox"/>	PROGRESS
<input type="checkbox"/>	ATTEMPTABDUR
<input type="checkbox"/>	ATTEMPTXDUR
<input type="checkbox"/>	ATTEMPT
<input checked="" type="checkbox"/>	COURSEID
<input checked="" type="checkbox"/>	LEARNERID
<input checked="" type="checkbox"/>	FILENAME
<input checked="" type="checkbox"/>	KEYNAME

Çizelge 3.13'daki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- TRACK\_UNKNOWN: “unknown” sabit değeridir.
- TRACK\_SATISFIED: “satisfied” sabit değeridir.
- TRACK\_NOTSATISFIED: “notsatisfied” sabit değeridir.
- TRACK\_COMPLETED: “completed” sabit değeridir.
- TRACK\_INCOMPLETE: “incomplete” sabit değeridir.
- DIRTYPRO: JSON nesnesi içinde geçirilen işleyiş ile ilgili bir verisidir.
- PRIMARYOBJ: “\_primary\_” sabit değeridir.
- PROGRESS: Kurstaki ilerleme bilgisidir (Örnek: completed, incomplete, unknown).
- ATTEMPTABDUR: Bir SCO ziyaretinde harcanabilecek maksimum süredir.
- ATTEMPTEXDUR: Önceki süredir.
- ATTEMPT: Toplam kursa giriş sayısıdır.
- COURSEID: Kurs ID' sidir.
- LEARNERID: Kullanıcı ID' sidir.
- FILENAME: Javascript Object dosyası ismidir.
- KEYNAME: JSON nesnesinde ilgili anahtarın ismidir.

### 3.5.4.6 SCORM\_OBJ\_ADVSEQACTIVITYTREE Tablosu

Bu tablo, API tarafından gönderilen veri nesnesinin veritabanında düzgün tutulması için gerekli bir tablodur. Çizelge 3.14'de verilmiştir.

**Çizelge 3. 14** SCORM OBJ ADLTOC tablosu.

SCORM_OBJ_ADVSEQACTIVITYTREE	
	LASTLEAF
	SCOPEID
🔑	COURSEID
🔑	LEARNERID
	OBJSET
	OBJMAP
	OBJSCAN
🔑	FILENAME
🔑	KEYNAME

Çizelge 3.14'deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- LASTLEAF: Son açılan SCO'nun tanımlayıcı kodudur (identifier ).
- SCOPEID: Kapsam tanımlayıcısıdır.
- COURSEID: Kursun ID' sidir.
- LEARNERID: Kullanıcı ID' sidir.
- OBJSET: Objective set bilgisidir.
- OBJMAP: Objective map bilgisidir.
- FILENAME: JAvascript object dosyasının ismidir.
- KEYNAME: İlgili JSON Hashtable anahtarının ismidir.

### 3.5.4.7 SCORM\_OBJ\_ADVSEQOBJECTIVETRACKING Tablosu

API tarafından gönderilen, öğrenme hedefleri (objectives) ile ilgili verilerin tutulduğu tablodur. SCORM\_OBJ\_ADVSEQOBJECTIVETRACKING tablosu, Çizelge 3.15'de verilmiştir.

**Çizelge 3. 15** SCORM\_OBJ\_ ADVSEQOBJECTIVETRACKING tablosu.

SCORM_OBJ_ADVSEQOBJECTIVETRACKING	
LEARNERID	
SCOPEID	
DIRTYOBJ	
SETOK	
HASSATISFIED	
SATISFIED	
HASMEASURE	
MEASURE	
READSTATUS	
READMEASURE	
WRITESTATUS	
WRITEMEASURE	
COURSEID	
FILENAME	
KEYNAME	
HASHKEYNAME	

Çizelge 3.15’deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- LEARNERID: Kullanıcı ID’ sidir.
- SCOPEID: Kapsam tanımlayıcısıdır.
- HASSATISFIED: Önceden tatminkar bir sonuç alınıp alınmadığı bilgisidir.
- SATISFIED: Tamamlama bilgisidir.
- HASMEASURE: Not alınıp alınmadığıdır.
- MEASURE: Alınan nottur.
- READSTATUS: Okuma durumudur.
- READMEASURE: Okuma ölçüsüdür.
- WRITESTATUS: Yazma durumudur.
- WRITEMEASURE: Yazma ölçüsüdür.
- COURSEID: Kurs ID’ sidir.
- FILENAME: JSON dosya ismidir.
- KEYNAME: JSON dosyasındaki *Hashtable* anahtar ismidir.
- HASHKEYNAME: JSON dosyasındaki *Hashtable* anahtar ismidir.

#### 3.5.4.8 SCORM\_OBJ\_ADVVERSION Tablosu

Veri modelinin versiyonunu tutmaya yarayan tablodur. API tarafından gönderilen veri nesnesi içinde yer alan versiyon bilgisi her kurs için ayrı ayrı bu tabloda tutulur. Bu yapıda bu bilgi daima “1.0” olarak girilir. Çizelge 3.16’da bu tablo verilmiştir.

**Çizelge 3. 16** SCORM\_OBJ\_ADVVERSION tablosu.

SCORM_OBJ_ADVVERSION	
	BINDING
	VERSION
🔑	KEYNAME
🔑	COURSEID
🔑	LEARNERID
🔑	FILENAME
🔑	HASHKEYNAME

Çizelge 3.16'daki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- BINDING: Veri modeli nesnesinin ismidir (\_version).
- VERSION: 1.0 olarak girilen versiyon değeridir.
- KEYNAME: JSON nesnesindeki ilgili Hashtable anahtarının ismidir.
- COURSEID: Kurs ID' sidir.
- LEARNERID: Kullanıcı ID' sidir.
- FILENAME: SCO tanımlayıcı kodudur (identifier).
- HASHKEYNAME: JSON verisi içinde veri modeli nesnesinin değerini almak için kullanılan anahtarların ismidir.

#### 3.5.4.9 SCORM\_OBJ\_SEQOBJECTIVE Tablosu

Bu tablo, API tarafından gönderilen veri nesnesinin veritabanında düzgün tutulması için gerekli bir tablodur. Bu tablo, Çizelge 3.17 de verilmiştir.

**Çizelge 3. 17** SCORM\_OBJ\_SEQOBJECTIVE tablosu.

SCORM_OBJ_SEQOBJECTIVE	
	OBJID
	SATISFIEDBYMEASURE
	ACTIVEMEASURE
	MINMEASURE
	CONTRIBUTESTOROLLUP
	MAPS
🔑	COURSEID
🔑	LEARNERID
🔑	FILENAME
🔑	KEYNAME

Çizelge 3.17'deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- OBJID: Objective tanımlayıcı kodudur.
- SATISFIEDBYMEASURE: Skorla satisfied durumunun belirlenip belirlenmeyeceği bilgisidir.
- MINMEASURE: En az ölçüt değeridir.
- CONTRIBUTESTOROLLUP: Rollup işlevinde bağlı bulunan üst aktivitenin skorunun hesaplanmasına katkısı bulunup bulunmayacağı bilgisidir.
- COURSEID: Kurs tanımlayıcı kodudur.
- LEARNERID: Kullanıcı tanımlayıcı kodudur.
- FILENAME: JSON dosyasının ismidir.
- KEYNAME: Json dosyasında ilgili Hashtable anahtarı ismidir.

### 3.5.4.10 SCORM\_OBJ\_SEQOBJECTIVEMAP Tablosu

Bu tablo, API tarafından gönderilen veri nesnesinin veritabanında düzgün tutulması için gerekli bir tablodur. Objectivelerin hangi global objectivelerle ilgili olduğunu gösteren tablodur. Aşağıdaki alanların detaylı anlatımı SCORM 2004 Sequencing and Navigation kitabındaki “3.10.3. Objective Map” kısmında mevcuttur. Bu tabloda tutulan veriler Çizelge 3.18 de verilmiştir.

**Çizelge 3. 18** SCORM\_OBJ\_SEQOBJECTIVEMAP Tablosu

SCORM_OBJ_SEQOBJECTIVEMAP	
	GLOBALOBJID
	READSTATUS
	READMEASURE
	WRITESTATUS
	WRITEMEASURE
🔑	FILENAME
🔑	COURSEID
🔑	LEARNERID
🔑	KEYNAME
🔑	HASHKEYNAME



**Çizelge 3.18'**deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- GLOBALOBJID: Global objective'in tanımlayıcı kodudur.
- READSTATUS: Okuma durumudur.
- READMEASURE: Okuma ölçüsüdür.
- WRITESTATUS: Yazma durumudur.
- WRITEMEASURE: Yazma ölçüsüdür.
- FILENAME: SCO tanımlayıcı kodudur (identifier).
- COURSEID: Kurs ID' sidir.
- LEARNERID: Kullanıcı ID' sidir.
- KEYNAME: JSON nesnesindeki ilgili Hashtable anahtarının ismidir.
- HASHKEYNAME: JSON verisi içinde veri modeli nesnesinin değerini almak için kullanılan anahtarların ismidir.

#### **3.5.4.11 SCORM\_OBJ\_SEQACTIVITY Tablosu**

Bu tablo, API tarafından gönderilen veri nesnesinin veritabanında düzgün tutulması için gerekli bir tablodur. Buradaki alanların çok detaylı anlatımı, SCORM 2004 kitapları arasında bulunan Sequencing and Navigation başlıklı kitapta bulunmaktadır. Çizelge 3.19 da bu tablo verilmiştir.

Çizelge 3.19 SCORM OBJ SEQACTIVITY tablosu.

SCORM_OBJ_SEQACTIVITY	
	TIMING_NEVER
	TIMING_ONCE
	TIMING_EACHNEW
	TER_EXITALL
	PRECONDITIONRULES
	POSTCONDITIONRULES
	EXITACTIONRULES
	XML
	DEPTH
	COUNT
🔑	LEARNERID
	SCOPEID
	ACTIVITYID
	RESOURCEID
	STATEID
	TITLE
	ISVISIBLE
	[ORDER]
	ACTIVEORDER
	SELECTED
	PARENT
	ISACTIVE
	ISSUSPENDED
	DELIVERYMODE
	CONTROL_CHOICE
	CONTROL_CHOICEEXIT
	CONTROL_FLOW
	CONTROL_FORWARDONLY
	CONSTRAINCHOICE
	PREVENTACTIVATION
	USECROBJ
	USECURPRO
	MAXATTEMPTCONTROL
	MAXATTEMPT
	ATTEMPTABDURCONTROL
	ATTEMPTABDUR
	ATTEMPTXDURCONTROL
	ATTEMPTXDUR
	ACTIVITYABDURCONTROL
	ACTIVITYABDUR
	ACTIVITYXDURCONTROL
	ACTIVITYXDUR
	BEGINTIMECONTROL
	BEGINTIME
	ENDTIMECONTROL
	ENDTIME
	AUXRESOURCES
	ROLLUPRULES
	ACTIVITYMEASURE
	REQUIREDFORSAISFIED
	REQUIREDFORNOTSAISFIED
	REQUIREDFORCOMPLETED
	REQUIREDFORINCOMPLETE
	OBJECTIVES
	OBJMAPS
	ISOBJECTIVEROLLEDUP
	OBJMEASUREWEIGHT
	ISPROGRESSROLLEDUP
	SELECTTIMING
	SELECTSTATUS
	SELECTCOUNT
	SELECTION
	RANDOMTIMING
	REORDER
	RANDOMIZED
	ISTRACKED
	CONTENTSETSCOMPLETION
	CONTENTSETSOBJ
	NUMATTEMPT
	NUMSCOATTEMPT
	ACTIVITYABDUR_TRACK
	ACTIVITYXDUR_TRACK
🔑	COURSEID
🔑	KEYNAME
🔑	HASHKEYNAME
🔑	FILENAME

Çizelge 3.19'deki tabloda geçen alanların açıklaması aşağıda verilmiştir.

- TIMING\_NEVER: “never” sabit deęeridir.
- TIMING\_ONCE: “once” sabit deęeridir.
- TIMING\_EACHNEW: “onEachNewAttempt” sabit deęeridir.
- TER\_EXITALL: “\_EXITALL\_” sabit deęeridir.
- PRECONDITIONRULES: SCO aktivitesinin öncelik kurallarıdır.
- POSTCONDITIONRULES: SCO aktivitesinin sonraki kurallarıdır.
- EXITACTIONRULES: SCO aktivitesinin çıkış kurallarıdır.
- XML: İlgili XML verisidir.
- DEPTH: Aktivitenin SCORM ağaç yapısındaki derinlięidir.
- COUNT: Aktivitenin açılma sayısıdır.
- LEARNERID: Kullanıcı ID’ sidir.
- ACTIVITYID: SCO tanımlayıcı kodudur (identifier).
- RESOURCEID: Imsmanifest.xml dosyasında da bulunan aktivitenin kaynak tanımlayıcı kodudur (resource identifier).
- TITLE: SCO aktivitesinin başlıęıdır.
- ISVISIBLE: SCO aktivitesinin içindekiler ağacında gösterilip gösterilmeyeceęidir.
- ORDER: Bir kursun içindekiler ağacında DEPTH deęeri aynı olan dięer SCO’lara göre dizilmesini saęlayan sıralama sayısıdır.
- ACTIVEORDER: SCO’nun aktif sırasıdır.
- SELECTED: Aktivitenin seçilip seçilmedięi bilgisidir.
- PARENT: Aktivitenin baęlı olduęu üst aktiviteye ait tanımlayıcı kodudur.
- ISACTIVE: Aktivitenin aktif olup olmadıęı bilgisidir.
- ISSUSPENDED: Aktiviteden tekrar kalındıęı yerden devam etmek üzere çıkılıp çıkılmadıęının bilgisidir.
- DELIVERYMODE: İçerięin verilme şeklidir.
- CONTROL\_CHOICE: Alt aktivitelerin seçilip seçilemeyeceęi bilgisidir.
- CONTROL\_CHOICEEXIT: Aktivitenin alt aktivitesi olmayan dięer aktiviteleri seçerek aktiviteden çıkma izninin olup olmama bilgisidir.
- CONTROL\_FLOW: Aktivite kümesi içindeki aktiviteler arasında ileri geri butonlarını kullanarak gezinebilmeye olanak tanınıp tanınmayacaęı bilgisidir.

- CONTROL\_FORWARDONLY: Aktiviteler arası sadece ileri butonu kullanılarak dolaşılıp dolaşılmayacağı bilgisidir.
- CONSTRAINCHOICE: Aktivitenin sadece bir ileri ya da gerisinde bulunan aktivitelerin seçilip seçilemeyeceği bilgisidir.
- PREVENTACTIVATION: Alt aktivitenin eğer ilgili üst aktiviteye henüz ulaşılmadıysa seçilmesine izin verilir verilmeyeceği bilgisidir.
- USECUROBJ: O anki objective bilgisinin (current objective information) kullanıp kullanılmayacağı bilgisidir.
- USECURPRO: O anki ilerleme bilgisinin (current attempt progress information) kullanılıp kullanılmayacağı bilgisidir.
- MAXATTEMPTCONTROL: Bir aktiviteye maksimum kaç defa girilebileceği kontrolünün olup olmadığı bilgisidir.
- MAXATTEMPT: Bir aktiviteye maksimum kaç defa girilebileceği bilgisidir.
- ATTEMPTABDURCONTROL: Bir aktivitede bir girişte maksimum harcanacak süre kontrolünün olup olmayacağı bilgisidir.
- ATTEMPTABDUR: Bir girişte aktiviteye harcanabilecek maksimum süredir.
- ACTIVITYABDURCONTROL: Bir aktivitede harcanabilecek toplam maksimum süre kontrolünün olup olmayacağı bilgisidir.
- ACTIVITYABDUR: Bir aktivitede harcanabilecek toplam maksimum süredir.
- ACTIVITYEXDURCONTROL: “Activity Experienced Duration” kontrolünün olup olmayacağı bilgisidir.
- ACTIVITYEXDUR: Activity Experienced Duration değeridir.
- BEGINTIMECONTROL: Başlangıç zamanı kontrolü olup olmayacağı bilgisidir.
- BEGINTIME: Başlangıç zamanıdır.
- ENDTIMECONTROL: Bitiş zamanı kontrolü olup olmayacağı bilgisidir.
- ENDTIME: Bitiş zamanıdır.
- AUXRESOURCES: Aktivite ile ilişkili yardımcı kaynaklardır (auxiliary resources).
- ROLLUPRULES: Sıralama ile ilgili roll-up kurallarıdır.
- ACTIVITYMEASURE: Aktivitenin en az normalleştirilmiş ölçeğine göre (minimum normalized measure) karşılaştırılıp değerlendirilip değerlendirilmeyeceği bilgisidir (Bkz. 3.9. Rollup Consideration Controls).

- **REQUIREDFORSAISFIED:** Aktivitenin bađlı olduđu üst aktivitenin Satisfied durumuna aktivitenin katkısının olup olmadığı bilgisidir (Bkz. 3.9. Rollup Consideration Controls).
- **REQUEIREDFORNOTSAISFIED:** Aktivitenin bađlı olduđu üst aktivitenin Not Satisfied durumuna aktivitenin katkısının olup olmadığı bilgisidir (Bkz. 3.9. Rollup Consideration Controls).
- **REQUIREDFORCOMPLETED:** Aktivitenin bađlı olduđu üst aktivitenin Completed durumuna aktivitenin katkısının olup olmadığı bilgisidir (Bkz. 3.9. Rollup Consideration Controls).
- **REQUIREDFORINCOMPLETE:** Aktivitenin bađlı olduđu üst aktivitenin Incomplete durumuna aktivitenin katkısının olup olmadığı bilgisidir (Bkz. 3.9. Rollup Consideration Controls).
- **OBJECTIVES:** Aktivite ile ilişkili hedeflerdir.
- **OBJMAPS:** Bir lokal objective'in global objectivelere olan ilişkisidir.
- **ISOBJECTIVEROLLEDUP:** Objective'in roll-up işleminde kullanılıp kullanılmayacağı bilgisidir.
- **OBJMEASUREWEIGHT:** Rollup Objective Measure Weight değeridir (Bkz. Scorm 2004 Seq. & Nav. Kitabı 4.6.3 Measure Rollup Process).
- **ISPROGRESSROLLEDUP:** İlerleme bilgisinin Rollup işleminde kullanılıp kullanılmayacağı bilgisidir.
- **SELECTTIMING:** Aktivite seçiminin zamanlamasıdır (Bkz. 3.11.Selection Controls).
- **SELECTSTATUS:** Aktivite seçilme sayısının anlamlı olup olmadığı bilgisidir.
- **SELECTCOUNT:** Bir dizi aktivite arasından kaç tanesinin seçileceđi bilgisini içerir.
- **RANDOMTIMING:** Alt aktivitelerin sıralamasının rasgele ne zaman deđiştirileceđi bilgisidir (Bkz. 3.12. Randomization Controls).
- **REORDER:** Alt aktivitelerin rasgele gösterilip gösterilmediđi bilgisidir.
- **RANDOMIZED:** Alt aktivitelerin rasgele gösterilip gösterilmeyeceđi bilgisidir.
- **ISTRACKED:** Aktivitenin takip verilerinin API aracılıđıyla gönderilip gönderilmediđi bilgisidir.

- CONTENTSETSCOMPLETION: Aktivitenin tamamlanıp tamamlanmadığı bilgisinin içerik tarafından belirlenip belirlenmediği bilgisidir (Bkz. 3.13.2. Completion Status Set By Content).
- CONTENTSETSOBJ: Aktivitenin tanımlanmış amacına ulaşıp ulaşılmadığı bilgisinin içerik tarafından belirlenip belirlenmemesi bilgisidir (3.12.3. Objective Set By Content).
- NUMATTEMPT: Aktiviteye yapılan giriş sayısıdır.
- NUMSCOATTEMPT: Aktiviteden her çıkış olduğunda artırılan, toplam giriş sayısıdır.
- COURSEID: Kurs ID' sidir.
- KEYNAME: JSON dosyasındaki ilgili ana Hashtable anahtarı ismidir.
- HASHKEYNAME: JSON dosyasındaki ilgili Hashtable anahtarı ismidir.
- FILENAME: JSON dosyası ismidir.

### **3.5.5 SCORM 2004 Runtime CMI Veri Modeli**

SCORM 2004 AAPI, Scorm 1.2 Api'de olduğu gibi bazı değişkenleri tutmamızı istemektedir. Daha önce veri modeli hakkında bilgi verilmiştir (Bkz. 3.4.1 SCORM 1.2 API Veri Tabanı Yapısı). Scorm 2004 API'de kullanılan veri modeli Çizelge 3.20'de verilmiştir.

### Çizelge 3. 20 SCORM 2004 Runtime CMI Veri Modeli.

- cmi.\_version (characterstring, RO) Represents the version of the data model
- cmi.comments\_from\_learner.\_children (comment,location.timestamp, RO)  
Listing of supported data model elements
- cmi.comments\_from\_learner.\_count (non-negative integer, RO) Current number of learner comments
- cmi.comments\_from\_learner.n.comment (localized\_string\_type (SPM: 4000), RW) Textual input
- cmi.comments\_from\_learner.n.location (characterstring (SPM: 250), RW) Point in the SCO to which the comment applies
- cmi.comments\_from\_learner.n.timestamp (time (second,10,0), RW) Point in time at which the comment was created or most recently changed
- cmi.comments\_from\_lms.\_children (comment,location.timestamp, RO) Listing of supported data model elements
- cmi.comments\_from\_lms.\_count (non-negative integer, RO) Current number of comments from the LMS
- cmi.comments\_from\_lms.n.comment (localized\_string\_type (SPM: 4000), RO) Comments or annotations associated with a SCO
- cmi.comments\_from\_lms.n.location (characterstring (SPM: 250), RO) Point in the SCO to which the comment applies
- cmi.comments\_from\_lms.n.timestamp (time(second,10,0), RO) Point in time at which the comment was created or most recently changed
- cmi.completion\_status (state (completed, incomplete, not attempted, unknown), RW) Indicates whether the learner has completed the SCO
- cmi.completion\_threshold (real(10,7) range (0..1), RO) Used to determine whether the SCO should be considered complete
- cmi.credit (state (credit, no\_credit), RO) Indicates whether the learner will be credited for performance in the SCO
- cmi.entry (state (ab\_initio, resume, ""), RO) Asserts whether the learner has previously accessed the SCO

- cmi.exit (state (timeout, suspend, logout, normal, ""), W) Indicates how or why the learner left the SCO
- cmi.interactions.\_children (id,type,objectives,timestamp,correct\_responses,weighting,learner\_resposne, result,latency,description, RO) Listing of supported data model elements
- cmi.interactions.\_count (non-negative integer, RO) Current number of interactions being stored by the LMS
- cmi.interactions.n.id (long\_identifier\_type (SPM: 4000), RW) Unique label for the interaction
- cmi.interactions.n.type (state (true\_false, multiple\_choice, fill\_in, long\_fill\_in, matching, performance, sequencing, likert, numeric, other), RW) Which type of interaction is recorded
- cmi.interactions.n.objectives.\_count (non-negative integer, RO) Current number of objectives (i.e., objective identifiers) being stored by the LMS for this interaction
- cmi.interactions.n.objectives.n.id (long\_identifier\_type (SPM: 4000), RW) Label for objectives associated with the interaction
- cmi.interactions.n.timestamp (time(second,10,0), RW) Point in time at which the interaction was first made available to the learner for learner interaction and response
- cmi.interactions.n.correct\_responses.\_count (non-negative integer, RO) Current number of correct responses being stored by the LMS for this interaction
- cmi.interactions.n.correct\_responses.n.pattern (format depends on interaction type, RW) One correct response pattern for the interaction
- cmi.interactions.n.weighting (real (10,7), RW) Weight given to the interaction relative to other interactions
- cmi.interactions.n.learner\_response (format depends on interaction type, RW) Data generated when a learner responds to an interaction
- cmi.interactions.n.result (state (correct, incorrect, unanticipated, neutral, real (10,7) ), RW) Judgment of the correctness of the learner response
- cmi.interactions.n.latency (timeinterval (second,10,2), RW) Time elapsed



between the time the interaction was made available to the learner for response and the time of the first response

- `cmi.interactions.n.description` (`localized_string_type` (SPM: 250), RW) Brief informative description of the interaction
- `cmi.launch_data` (`characterstring` (SPM: 4000), RO) Data provided to a SCO after launch, initialized from the `dataFromLMS` manifest element
- `cmi.learner_id` (`long_identifier_type` (SPM: 4000), RO) Identifies the learner on behalf of whom the SCO was launched
- `cmi.learner_name` (`localized_string_type` (SPM: 250), RO) Name provided for the learner by the LMS
- `cmi.learner_preference._children`  
(`audio_level,language,delivery_speed,audio_captioning`, RO) Listing of supported data model elements
- `cmi.learner_preference.audio_level` (`real(10,7)`, range (0..\*), RW) Specifies an intended change in perceived audio level
- `cmi.learner_preference.language` (`language_type` (SPM 250), RW) The learner's preferred language for SCOs with multilingual capability
- `cmi.learner_preference.delivery_speed` (`real(10,7)`, range (0..\*), RW) The learner's preferred relative speed of content delivery
- `cmi.learner_preference.audio_captioning` (`state (-1,0,1)`, RW) Specifies whether captioning text corresponding to audio is displayed
- `cmi.location` (`characterstring` (SPM: 1000), RW) The learner's current location in the SCO
- `cmi.max_time_allowed` (`timeinterval (second,10,2)`, RO) Amount of accumulated time the learner is allowed to use a SCO
- `cmi.mode` (`state (browse, normal, review)`, RO) Identifies one of three possible modes in which the SCO may be presented to the learner
- `cmi.objectives._children` (`id,score,success_status,completion_status,description`, RO) Listing of supported data model elements
- `cmi.objectives._count` (`non-negative integer`, RO) Current number of objectives being stored by the LMS

- cmi.objectives.n.id (long\_identifier\_type (SPM: 4000), RW) Unique label for the objective
- cmi.objectives.n.score.\_children (scaled,raw,min,max, RO) Listing of supported data model elements
- cmi.objectives.n.score.scaled (real (10,7) range (-1..1), RW) Number that reflects the performance of the learner for the objective
- cmi.objectives.n.score.raw (real (10,7), RW) Number that reflects the performance of the learner, for the objective, relative to the range bounded by the values of min and max
- cmi.objectives.n.score.min (real (10,7), RW) Minimum value, for the objective, in the range for the raw score
- cmi.objectives.n.score.max (real (10,7), RW) Maximum value, for the objective, in the range for the raw score
- cmi.objectives.n.success\_status (state (passed, failed, unknown), RW) Indicates whether the learner has mastered the objective
- cmi.objectives.n.completion\_status (state (completed, incomplete, not attempted, unknown), RW) Indicates whether the learner has completed the associated objective
- cmi.objectives.n.progress\_measure (real (10,7) range (0..1), RW) Measure of the progress the learner has made toward completing the objective
- cmi.objectives.n.description (localized\_string\_type (SPM: 250), RW) Provides a brief informative description of the objective
- cmi.progress\_measure (real (10,7) range (0..1), RW) Measure of the progress the learner has made toward completing the SCO
- cmi.scaled\_passing\_score (real(10,7) range (-1 .. 1), RO) Scaled passing score required to master the SCO
- cmi.score.\_children (scaled,raw,min,max, RO) Listing of supported data model elements
- cmi.score.scaled (real (10,7) range (-1..1), RW) Number that reflects the performance of the learner
- cmi.score.raw (real (10,7), RW) Number that reflects the performance of the

learner relative to the range bounded by the values of min and max

- cmi.score.min (real (10,7), RW) Minimum value in the range for the raw score
- cmi.score.max (real (10,7), RW) Maximum value in the range for the raw score
- cmi.session\_time (timeinterval (second,10,2), WO) Amount of time that the learner has spent in the current learner session for this SCO
- cmi.success\_status (state (passed, failed, unknown), RW) Indicates whether the learner has mastered the SCO
- cmi.suspend\_data (characterstring (SPM: 64000), RW) Provides space to store and retrieve data between learner sessions
- cmi.time\_limit\_action (state (exit,message, continue,message, exit,no message, continue,no message), RO) Indicates what the SCO should do when cmi.max\_time\_allowed is exceeded
- cmi.total\_time (timeinterval (second,10,2), RO) Sum of all of the learner's session times accumulated in the current learner attempt
- adl.nav.request (request(continue, previous, choice, exit, exitAll, abandon, abandonAll, \_none\_), RW) Navigation request to be processed immediately following Terminate()
- adl.nav.request\_valid.continue (state (true, false, unknown), RO) Used by a SCO to determine if a Continue navigation request will succeed.
- adl.nav.request\_valid.previous (state (true, false, unknown), RO) Used by a SCO to determine if a Previous navigation request will succeed.
- adl.nav.request\_valid.choice (state (true, false, unknown), RO) Used by a SCO to determine if a Choice navigation request for a particular activity will succeed.

#### **API\_1484\_11**

- Initialize( "" ) : bool
- Terminate( "" ) : bool
- GetValue( element : CMIElement ) : string

- SetValue( element : CMIElement, value : string ) : bool
- Commit( "" ) : bool
- GetLastError() : CMIErrCode
- GetErrorString( errorCode : CMIErrCode ) : string
- GetDiagnostic (errorCode : CMIErrCode ) :string

Sequencing Activity State Data	
Activity Information	Objective Information
<ul style="list-style-type: none"> <li>• Activity Progress Status controls:</li> <li>• Activity Attempt Count – which is the number of attempts the learner has made on this activity</li> <li>• Attempt Progress Status controls:</li> <li>• Attempt Completion Amount – which is the measure of the learner’s progress towards completion and is related to cmi.progress_status</li> <li>• Attempt Completion Status – which indicates whether an attempt is complete and is related to cmi.completion_status</li> </ul>	<ul style="list-style-type: none"> <li>• Objective Progress Status controls:</li> <li>• Objective Satisfied Status – which indicates whether an objective is satisfied and is related to cmi.success_status</li> <li>• Objective Measure Status controls:</li> <li>• Objective Normalized Measure – which is the measure of the score for an objective</li> </ul>

### 3.5.6 SCORM Uyumlu İçerik Geliştirilmesi

SCORM uyumluluk yalnız başına bir LMS'e herhangi bir avantaj sağlamaz. SCORM uyumlu bir LMS, bu özelliğini ancak, SCORM uyumlu bir içeriği sunarken ve izlerken gösterir. Dijital ortama taşınan içerikler, uzaktan eğitim standardı olan SCORM'a göre paketlenir ve takip sistemi geliştirilerek öğrencilerin ders ile ilgili takip bilgileri kayıt altına alınır (İnt. Kyn. 6).

SCORM uyumlu içerikler dijital ortama taşınırken, uzaktan eğitim standardı olan SCORM'a göre paketlenir ve takip sistemi geliştirilerek öğrencilerin ders ile ilgili takip bilgileri kayıt altına alınır. SCORM uyumlu içeriğin avantajları aşağıda verilmiştir.

- Kalıcılık (Durability): İçerik, tutarlı olduğu sürece kalıcı olmaktadır.
- Taşınabilirlik (Portability): İçerik bir sistemden başka bir SCORM destekli sisteme taşınabilmektedir.
- Tekrar Kullanılabilirlik (Reusability): SCORM uyumlu içerik tekrar tekrar kullanılabilir.
- Çok yönlülük (Interoperability): İçerik bir sistemden başka bir sisteme taşındığında da çalışabilmektedir.
- Erişilebilirlik (Accessibility): Belirli metadataların içerik paketine eklenmesi ile içeriğe, içerik kütüphanesinden erişilebilmektedir.

#### 3.5.6.1 Macromedia Flash ile SCORM Uyumlu Örnek İçerik Geliştirilmesi

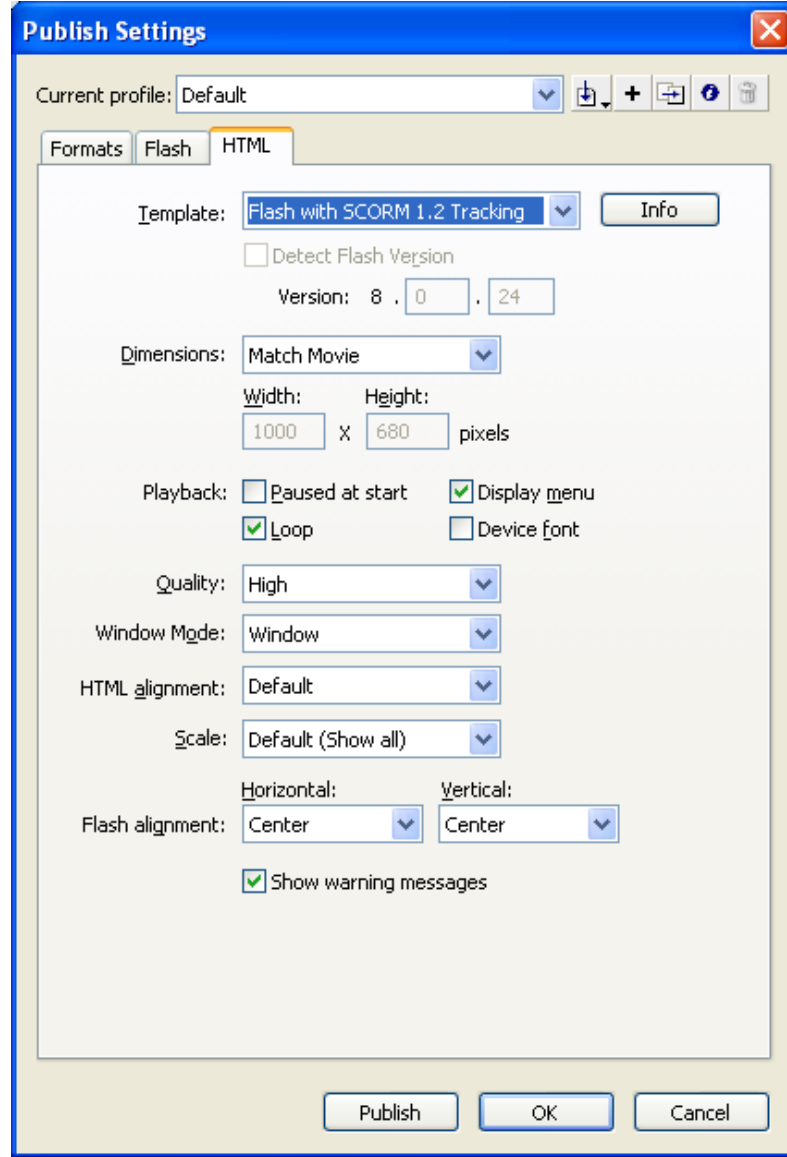
Flash öğrenme etkileşimleri ve test şablonları, AICC ve SCORM uyumlu LMS'lerle kolay iletişim imkanı yaratırlar. Flash belgelerinin içine yerleştirilen kod ve ona karşılık gelen HTML/JavaScript dosyaları LMS'ye doğru bir şekilde formatlanmış veriler

gönderir. Test şablonları puanı ve harcanan toplam zamanı takip ederken, tek başına çalışan etkileşimler soru verileri gönderir.

SCORM uyumlu olması için içerik ilk olarak başlatıldığında veya diğer izleme komutları LMS' ye gönderilmeden önce bir başlatma komutu çağırmalıdır. SCORM HTML şablonu içeren Flash, dosya yüklendiğinde SCORM uyumlu bir LMS ile iletişim başlatmak için tasarlanmıştır. Ayrıca dosya boşaltıldığında, bitirme komutu önceden açık bir şekilde gönderilmemişse, LMS'ye bir bitirme iletişimi gönderir.

Aşağıda SCORM 1.2 Uyumlu örnek flash içerik hazırlanması anlatılmıştır.

1. **Giris fla**, **Şekil 3.6**'daki ayarlar kullanılarak publish edilmelidir. Bu sayede oluşturulan **giris.html** dosyasına SCORM API'yi kullanacak javascript metotları eklenmiş olur.



Şekil 3. 5 Macromedia da Flash içeriğın publish edilerek SCORM uyumlu hale getirilmesi.

2. **ScormHandler.as** dosyasında bulunan scormHandler sınıfı geliştirilmiştir. Bu sınıfta bulunan statik metotlar aracılığıyla (ExternalInterface sınıfı kullanılarak) giriş.html dosyasında bulunan ve scorm api metotlarını çağıran metotların kullanılmasına olanak sağlanmıştır. ScormHandler.as kodları aşağıda verilmiştir.

### ScormHandler.as Kodları:

```
import flash.external.*;
class scormHandler extends Object {
public static function SCOGetValue(nam)      {
    trace(nam);
    var rVal = null;
    if(ExternalInterface.available) rVal =
    ExternalInterface.call("SCOGetValue",nam);
    return rVal && rVal!="null" ? String(rVal) : "";
}

public static function SCOCommit() {
    var rVal = null;
    if(ExternalInterface.available) rVal = ExternalInterface.call("SCOCommit");
    return rVal && rVal!="null" ? String(rVal) : "";
}

public static function SCOGetLastError() {
    var rVal = null;
    if(ExternalInterface.available) rVal =
    ExternalInterface.call("SCOGetLastError");
    return rVal && rVal!="null" ? String(rVal) : "";
}

public static function SCOGetErrorString(n) {
    var rVal = null;
    if(ExternalInterface.available) rVal =
    ExternalInterface.call("SCOGetErrorString",n);
    return rVal && rVal!="null" ? String(rVal) : "";
}
```



```
public static function SCOGetDiagnostic(p)    {  
var rVal = null;  
if(ExternalInterface.available) rVal =  
ExternalInterface.call("SCOGetDiagnostic",p);  
return rVal && rVal!="null" ? String(rVal) : "";  
}  
  
public static function SCOSetValue(nam:String,val:String){  
var rVal = null;  
if(ExternalInterface.available) rVal =  
ExternalInterface.call("SCOSetValue",nam,val);  
return rVal && rVal!="null" ? String(rVal) : "";  
}  
}
```

3. Giriş fla'ya "scorm test" isimli katman eklenmiştir ve burada deneme yapılabilmesi için iki tane metin kutusu ve buton eklenmelidir. Bu katmandaki script incelendiğinde değerlerin nasıl gönderileceği ve okunacağı, hata yakalamanın nasıl yapılacağı konusunda fikir elde edebilir. scormHandler sınıfı, fla dosyası ile aynı dizinde olacak şekilde import deyiimiyle alınıp, içindeki metotlar çağrılabilir.

### 3.5.6.2 Skor ve Sayfa Bilgilerinin Tutulması

Flash öğrenme etkileşimleri ve test şablonları kullanılarak oluşturulan dosyalar, SCORM uyumlu LMS'ye izleme verileri gönderebilir. Tek etkileşimler toplam puan ve izleme verileri göndermezler, ama etkileşim veya soru verisi gönderebilirler. Aşağıda bu kodlamaya örnek verilmiştir.

- Skoru set etmek için aşağıdaki kod yazılabilir.

```
scormHandler.SCOSetValue("cmi.core.score.raw", "50");
```

- Kişinin dolaştığı sayfaları ve son kaldığı yeri de aşağıdaki gibi tutmak mümkündür.

```
scormHandler.SCOSetValue("cmi.suspend_data", "1,2,3,4|4");
```

- Virgüllü liste ile dolaştığı sayfaları yazıp bu stringin sonuna | işareti ile de son kaldığı yer eklenebilir. Kişi derse girdiği zaman da aşağıdaki şekilde kişinin gezdiği sayfalar ve kaldığı yer bilgisi alınabilir:

```
var stateData = scormHandler.SCOGetValue("cmi.suspend_data");  
var score = scormHandler.SCOGetValue("cmi.core.score.raw");
```

### 3.5.6.3 Bilgilerin Gönderilmesi

SCOSetValue çalıştırdıktan sonra belli aralıklarla (ya da hemen) bilgilerin sunucuya gönderilip kaydedilmesi için **scoHandler.SCOCommit()** fonksiyonunu çağırmak gerekmektedir. Ayrıca gönderilen bilgilerde bir hata olup olmadığını anlamak için de aşağıdaki gibi bir satır eklenebilir:

```
if(error && error.length > 0 ) Alert.show(error);
```

SCORM API de toplam 8 fonksiyon kullanılmaktadır. Bunlar aşağıda verilmiştir.

- Initialize()
- Terminate()
- GetValue()
- SetValue, Commit()
- GetLatError()
- GeterrorString()
- GetDiagnostic()

Bunların kullanımı ile ilgili detaylı bilgiye ADL Runtime API kitabından ulaşılabilir.

## **4. HAZIRLANAN LMS'İN TEKNİK YAPISI**

### **4.1 Hazırlanan LMS de Kullanılan Teknolojiler.**

Hazırlanan LMS, Yüksek öğretim programları hedef alınarak hazırlanmıştır. Toplam 37 Modülden oluşan, SCORM 1.2 ve SCORM 2004 uyumlu olan LMS, hazırlanırken aşağıdaki teknolojilerden yararlanılmıştır.

- Microsoft SQL Server 2005
- Net Framework 2.0
- Microsoft SQL Server 2005 Reporting Services
- Microsoft Visual Studio 2005
- Microsoft Visual Studio Team System
- XML Web Servisleri
- AJAX Teknolojisi

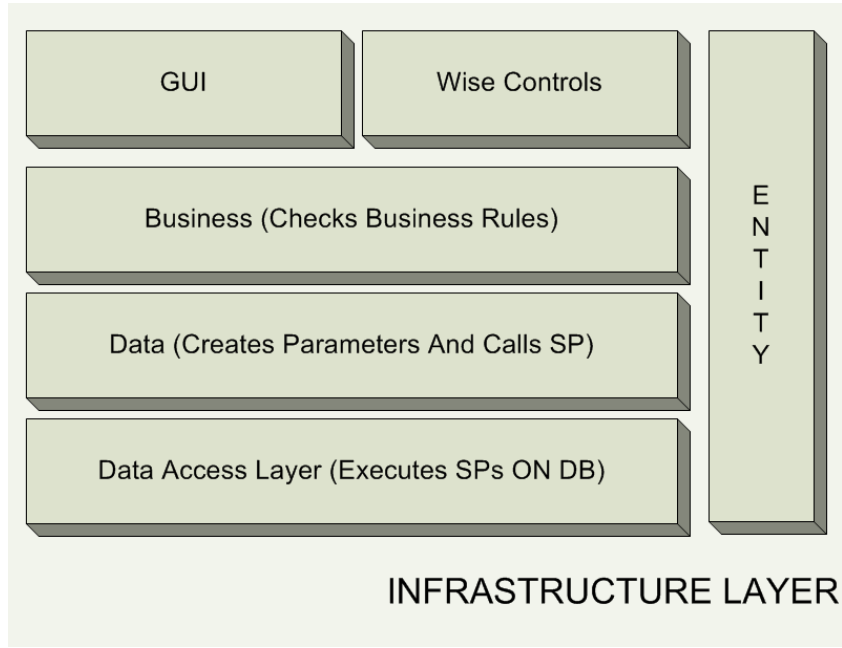
Kullanılan teknolojiler sayesinde elde edilecek kazanımlar ise aşağıda verilmiştir (Althoff, K et al. 2005).

- Wise Web kontroller ile kullanıcı dostu bir arabirim sağlanacaktır.
- .Net 2.0 ve Ajax Teknolojisinin güncel ve yeni bir teknoloji olması sayesinde elde edilecek performans getirisinin yanında geleceğe yönelik yeni versiyon geliştirme işlemleri mevcut alt yapı üzerinden devam edebilecektir.
- Web Partlar kullanımı sayesinde kullanıcılar ekranlarını kişiselleştirebilecektir.
- Reporting Services kullanımı sayesinde sistem üzerinde kolaylıkla yeni raporlar oluşturulabilecektir.

### **4.2 Mimari Tasarımı Oluşturan Katmanlar**

Mimari tasarım katmanlarının en önemli avantajı parçala ve yönet olayıdır. Hangi işin nerede yapıldığı bilindiği için, bir problem durumunda sadece oradaki bölüm değiştirilip

program, çalışmasına kaldığı yerden devam etmesi sağlanır. Bir diğer avantajı ise, uygulamanın genişlemesi durumudur. Eğer uygulamanın başka bir veritabanında çalışması yada uygulamanın başka bir istemci sistemde çalışması durumunda, yapılması gereken işlemi kolaylaştırmaktadır. Mimari tasarım katmanlarını oluşturmak için sistemi temel modüllere bölmek, en uygun stratejidir. Şekil 4.1 de Mimari tasarımı oluşturan katmanlar verilmiştir.



Şekil 4. 1 Mimari tasarımı oluşturan katmanlar.

Bu katmanların açıklaması aşağıda verilmiştir.

#### 4.2.1 Data Access Layer Katmanı

*Data Layer*'da yer alan *RDBMS* sistemine göre hazırlanmış bir katmandır. Bu katman, *Data Layer* katmanında kullanılan veritabanı yönetim sistemi ile veri iletiminin nasıl yapılacağını belirler ve bu görevi üstlenir. Sistemin farklı veritabanı yönetim sistemlerine taşınması için *Data Access Layer* üzerinde küçük düzenlemeler yapılması yeterlidir. Bu katmanda generic database sürücülerini (*OLEDB sürücülerini*) kullanmak

yerine alt katmanda bulunan veritabanı yönetim sistemi üzerinde en iyi performansı verecek *ADO.NET* veri sağlayıcıları kullanılmaktadır.

#### **4.2.2 Data Layer Katmanı**

*Business Layer* ve *Data Access Layer* arasında olan bu katmanın görevi *Data Access Layer* aracılığı ile alınan ham verinin *Entity*'lere yüklenmesini veya *Entity*'de bulunan verinin veritabanı kullanımını için hazır hale getirilmesini sağlayan katmandır.

#### **4.2.3 Entity Layer Katmanı**

Analiz çalışmaları neticesinde ortaya çıkan uygulama nesnelерinin tanımlandığı katmandır. *Entity* terimine örnek vermek gerekir ise, *Personel*, *Job Family Factors*, *Education Faktors* gibi terimler sistemdeki bazı *Entity*'leri tanımlamaktadır. *Entity*'ler tanımlanırken gerçek yaşamdaki nesne ilişkileri düşünülerek gerçek bir model tasarlanmaktadır.

#### **4.2.4 Business Layer Katmanı**

Projede yer alan iş kurallarının uygulamasının gerçekleştirildiği katmandır. Bir iş kuralı için yapılacak sına ve karar verme işlemleri bu katman içerisinde yapılır ve buna göre *presentation layer*'a bilgi aktarımı yada *data mapper layer*'a veri aktarımı gibi işlemler gerçekleştirilir. *Presentation layer* diğer katmanlar ile direkt iletişim kurmaz mutlaka *business layer* üzerinden geçerek işlemleri gerçekleştirmektedir. Böylece iş kuralları, hata yakalama blokları tek katman üzerinden bulunmaktadır ve kodlama kolaylığı sağlamaktadır.

#### 4.2.5 Presentation Layer Katmanı

Uygulamanın gereksinimlerine göre Windows Application, Web Application ya da Mobile Application'lar olabilmektedir. Kullanıcının göreceği ekranları oluşturan *presentation layer* kullanıcı dostu, şık ve gerekli durumlarda özelleştirilebilir kullanıcı arabirimi sunmaktadır.

#### 4.3 SQL Server 2005

SQL Server 2005, entegre iş zekası (BI) araçlarıyla kurumsal sınıf veri yönetimi sağlayan kapsamlı bir veritabanı platformudur. SQL Server 2005 veritabanı motoru, işletmelerde kullanılması için yüksek kullanılabilirliğe ve performansa sahip veri uygulamalarını oluşturup yönetilmesine olanak vererek, ilişkisel ve yapılandırılmış veri için güvenli ve güvenilir depolama sağlar.

SQL Server 2005 veri motoru, kurumsal veri yönetimi çözümünün temelini oluşturur. Ayrıca SQL Server 2005, analiz, raporlama, entegrasyon ve uyarı servislerinin en iyi özelliklerini birleştirir. Skor kartlar, paneller, Web servisleri ve mobil aygıtlar yoluyla işletmenin her köşesinde verinin yönetilmesine yardım eder. Microsoft Visual Studio, Microsoft Office System ve Business Intelligence Development Studio gibi yeni geliştirme araçları ile sıkı entegrasyonu, SQL Server 2005'i diğerlerinden farklı kılar. SQL Server 2005, geliştirici, veritabanı yöneticisi, bilgi çalışanı veya karar alıcıların verilerden daha fazla değer almasına yardım eden yenilikçi çözümler sağlamaktadır.

Çizelge 4.1'deki diyagram, SQL Server 2005'in temel bileşenlerini göstermektedir. İşletmenin her köşesine verileri iletmek için çözümler sunmak üzere SQL Server 2005'in Microsoft Windows platformuyla entegre olarak Windows Server System'in ne kadar önemli bir parçası olduğunu göstermektedir. Bu özellikler, yapılacak işin üç ana alanda geliştirmeye yardımcı olmaktadır.

- **Kurumsal Veri Yönetimi:** SQL Server 2005, iş alanı ve analitik uygulamalar için daha güvenilir, güvenli ve verimli bir veri platformu sunar. SQL Server'ın bu sürümü, SQL Server'ın bugüne kadarki en büyük, en güvenli ve en bağımlı sürümüdür.
- **Geliştirici Verimliliği:** SQL Server 2005, geliştirici verimliliğini önemli ölçüde artıran ve geliştiricileri güçlendiren birçok yeni teknolojileri içeren uçtan uca bir geliştirme ortamı sağlar.
- **İş Zekası:** SQL Server 2005'in kapsamlı analitik, entegrasyon ve veri geçişi becerileri, şirketlerin mevcut uygulamalarının değerini temel platformdan bağımsız olarak genişletmelerine olanak verir. SQL Server 2005'den yararlanan iş zekâsı çözümleri, çalışanlara kritik ve zamanında bilgiler sunarak, daha doğru kararlar almaları konusunda onları güçlendirir. Çizelge 4.1' de, SQL Server 2005'in yeni ve iyileştirilmiş güvenlik özelliklerinin bir özetini sunmaktadır.

**Çizelge 4. 1** İyileştirilmiş güvenlik özelliklerinin özeti (İnt. Kyn. 7).

Özellik	Açıklama
<b>Varsayılan Olarak Kapalı</b>	Varsayılan olarak kapalı SQL Server 2005, sadece sınırlı sayıdaki temel özellik ve servisleri varsayılan olarak sağlar, böylece, sunucunun açığa çıkan görünen alanını sınırlar ve yöneticilerin sadece kendi ortamları için gereken servis ve özellikleri etkinleştirmesine olanak verir. SQL Server 2005'de varsayılan olarak devre dışı olan servisler ve bileşenler şunlardır: Microsoft .NET Framework, SQL Service Broker Ağ Bağlanılabilirliği ve Analiz Servisleri'ndeki HTTP bağlanılabilirliği.
<b>Detaylı izin kontrolü</b>	SQL Server 2005'teki yeni bir güvenlik modeli, izinlerin yönetimini kolaylaştırmak ve en düşük ayrıcalıkların desteklenmesi prensibini sağlamak için, yöneticilerin belirtilen bir kapsamda ve bölümlenmiş bir düzeyde izinleri yönetmelerine olanak verir.

<b>Kullanıcıların ve şemanın ayrılması</b>	SQL Server 2005, kullanıcılar ve sahip oldukları veritabanı nesneleri arasındaki bağlantıyı ayırarak, güvenlik yönetimini kolaylaştırır.
<b>Standart oturumlar için uygulanan şifre ilkesi</b>	Yöneticiler, standart oturum açmalarda Microsoft Windows stilindeki ilkeleri belirlenmesine olanak verir. böylece etki alanındaki tüm hesaplar için tutarlı bir ilke uygulanır.
<b>Modüller üzerinde içerik çalıştırma</b>	SQL Server 2005, bir modüldeki ifadelerin çalıştığı bir bağlamı belirlenmesine olanak verir. Bu özellik, bölümlenmiş izin yönetimi için mükemmel bir mekanizma olarak da çalışır.
<b>Veri Tanımlama Dili (DDL) tetikleyicileri</b>	SQL Server 2005 ile DDL eylemlerini denetleme için tamamlayıcı bir mekanizma sağlayan DDL operasyonları üzerinde tetikleyici belirlenebilir.
<b>Veritabanı içinde veri şifreleme</b>	SQL Server 2005, bir anahtar yönetim altyapısı ile tam entegre olan veritabanının kendi içindeki şifreleme özelliklerini destekler.
<b>Kümeleme kimlik doğrulaması</b>	SQL Server 2005 kümelemesi, SQL Server 2005 sanal sunucusuna karşı Kerberos kimlik doğrulamasını destekler.
<b>Çoklu proxy hesapları</b>	SQL Server Agent, birden çok proxy hesabını destekler (her iş alt sistemi için bir adet).
<b>Local Security Authority (LSA) veritabanından bağımsız</b>	SQL Server Agent, yerel yönetici olarak çalışmak için servisin etkinleştirilmesini gerektirmez.
<b>SQL Profiler, artık sistem yöneticisi haklarını gerektirmez.</b>	SQL Server 2005'de bulunan yeni bir izin sayesinde, sistem yöneticisi haklarına sahip olmayan kullanıcılar, SQL Profiler'ı çalıştırabilir.



<b>Sunucu tanımlı ilkeler yoluyla Analysis server iletişim şifrelemesi</b>	Varsayılan olarak, istemci/sunucu iletişimleri şifrelenmiştir. Güvenlik güvencesini merkezileştirmek için, sunucu ilkesi, şifrelenmemiş iletişimlerini reddetmek için tanımlanabilir.
<b>Analysis server için detaylı yönetim rolleri</b>	SQL Server 2005 içinde daha fazla yönetimsel izinler mevcuttur. Çevrimiçi analitik işleme (OLAP) yöneticilerine ek olarak, veritabanı yöneticileri de, tek bir veritabanı içeriği içindeki yönetimsel izinlere sahip olabilir. Nesnelere üzerindeki yeni izinler, kullanıcıların nesne tanımlamasını görmelerine ve bir nesneyi işlemelerine olanak verir.
<b>SQL Server Agent iş rolleri</b>	SQL Server Agent, işler üzerindeki hakları atamayı detaylı bir şekilde desteklemek üzere iyileştirilmiştir.
<b>Yeni araçlar ve Yardım dosyaları</b>	Yeni kurma araçları ve belgeleri, SQL Server 2005'in mevcut bir SQL Server topolojisine veya yeni bir yüklemeye güvenli bir şekilde kurulmasına yardım eder. Bu araçlar, detaylı bilgi sağlayarak, mevcut topolojiyi analiz ederek, ön gereksinimleri kontrol ederek, bir yapılandırma ayarı önererek ve her adımı doğrulayarak adım adım gerçekleşen bir yaklaşım sağlar.
<b>Analysis Services için gelişmiş denetleme becerileri</b>	SQL Server 2005 Analiz Servisleri, SQL Profiler ile entegre olan yeni denetim becerilerini içerir.
<b>Güvenlik bültenleri</b>	Güvenlik bültenleri Microsoft, gerek gördükçe SQL Server 2005 için güvenlik bültenleri ve yamalar yayımlamaktadır. Bu bültenler, mevcut ortamlarımıza gelebilecek olası tehditleri anlamamıza ve saptamamıza yardım eder, ayrıca bunları nasıl etkisiz hale getireceğimiz hakkında bilgiler sunar.
<b>Microsoft Internet</b>	SQL Server 2005'i bir Windows 2000 Server platformu

<b>Information Services (IIS) Sihirbazı</b>	<b>Kilitleme</b>	üzerinde kurmayı planlıyorsak, IIS Kilitleme Sihirbazı Web sunucusu ortamımızı güvene almak için çok güçlü bir araçtır. ISS Kilitleme Sihirbazı, ortamımız için gereksiz olan özellikleri kapatır, böylece saldırıya uğrayabilecek güvenlik açıklarını azaltır.
---	------------------	---

#### 4.4 Visual Studio .NET 2005

Web'in programlanabilmesi demek, şu anda en zor işlemlerden biri olan programların entegrasyonunun kolaylaşması anlamına gelmektedir. Web servislerinin bize bu noktada sağladığı ise belirli standartlar üzerinde ortak bir platformu kullanan ve birbirleri ile konuşabilen uygulamalardır.

Web servisleri daha önce kullanılan program modülleri gibi tek başlarına belirli bir işi yerine getirmek üzere yapılmış programlardır. Bu servisleri kullanan kişilerin programın detaylarını bilmelerine gerek yoktur. Bu servisler kontrat olarak adlandırılan belirli arayüzler içermektedir. Bu kontratlar, sağlanan servislerin mesajlar şeklinde nasıl çağrılacağını ve geri dönen sonuçların tanımlarını içermektedir. Kontrat içerisinde servisin yapabildiği ölçüde Web servisi kullanıcıları, farklı şekillerde kendi uygulamaları için kullanabilmektedirler. Kontrat değişse dahi kullanım bilgileri içerisinde olduğu için, eğer geriye doğru uyumlu ise kullanıcı programları problemsiz çalışmaya devam etmektedir. Tabi ki bütün bunları yapabilmek için kullanılan teknoloji XML'dir. Kontratları tanımlayan (SCL) Service Contract Language tamamen XML grameri üzerine oturtulmuş bir yapıdır. DISCO (Discovery) tanımı web servisleri kontratlarını yayınlatabilmek ve kullanıcıların bunları bulabilmesi için geliştirilmiş XML tabanlı bir protokoldür. Neticede SOAP, SCL, Disco gibi protokoller uygulama geliştiricilere web servisini kullanmak için yardımcı olmak üzere tasarlanmış protokollerdir. Yazılımcılar bu protokolleri ve syntax'lerini bilmek zorunda değiller, çünkü geliştirme araçları (vs.Net) zaten bu yapıları tanımlamaktadır ve interface' i yazılımcıya göstermektedir.

.NET Framework içerisinde servisler önemli bir yer tutmaktadır. Bunlar sistem üzerinde çalışan Common Language Runtime (CLR) üzerinde temel class'ları kullanan ve Winforms ve Webforms ile direkt haberleşen bir yapıda bulunmaktadır.

Web Servisleri ise yalnızca Web için tasarlanmış olduğundan Winforms'un yanısıra ASP+'ın bir parçası gibi bağımsız çalışan ve iyi tanımlanmış programlar olarak bulunmaktadır.

Servis altyapısı herhangi bir .NET dili için temel class'ları ve programlama araçlarını sağlar, böylece farklı dillerde farklı class'lar kullanıldığı için her ortamda aynı hata ayıklama, kurulum ve analiz araçları kullanıldığından .NET ortamında öğrenme zamanı çok daha kısalmaktadır.

.Net Framework'ün içinde bulunan class kütüphaneleri standart olarak tüm dillerde kullanılmaktadır. Böylece farklı dillerde farklı integer, string vs gibi yapılarla karşılaşp bunları birbirine çevirmek zorunda kalınmamaktadır. Neticede .NET Framework Web servisleri için olması gereken standart bir yapıyı getirmektedir. Web servisleri de ihtiyaçları olan diğer standartları bunların üzerine kurarak gereken altyapıyı tamamlamaktadır.

Web formları ASP ve HTML kullanarak kendi kontrolleri ile web uygulamalarını masa üstü uygulamaları gibi geliştirilmesini sağlar. Web Formaları üzerinde kullanılan kontroller standart HTML olarak yazılmış kontrollerdir. Böylelikle veri bağlama, listeleme, görüntüleme ve veri girişi tasarımları daha hızlı yapılabilir.

ASP+ aynı zamanda web servislerini programlayabilmek için de gelişmiş bir programlama modeli sunmaktadır. ASP ve VB (veya C# veya C++ veya COBOL ..... ) kullanarak HTML, SOAP, SCL, XML vb protokolleri hiç öğrenilmeden web servisleri kullanılabilir.

Bir web servisi yaratmak için ASP+'ta bir ASMX dosyası oluşturulması gerekmektedir. Bu dosyadaki public class içine WebMethod attribute ile tanımlanacak Web Servisi

metodu eklenir. Bu metodlar ise ASMX dosyasına gelen HTTP istekleri ile harekete geçmektedir. ASP+'ın önemli bir özelliği de, bu class'ın metadatasını inceleyip SCL dosyasını kullanıcı tarafından istendiğinde otomatik olarak üretilebilmesidir. Web servisi kullanıcıları ise SOAP, HTTP GET, HTTP POST gibi yöntemlerle isteklerini gönderir.

Bu arada kullanıcı için bir proxy yaratılır ve bu proxy web servisine giden istekler ve gelen cevapları kendi üzerine alarak uygulama ile haberleşir. Böylelikle uygulama servisi ölçeklenebilir bir şekilde kullanabilmektedir. Bu model aynı zamanda stateless bir yapı üzerine oturtularak ölçeklenebilirliği arttırmaktadır.

Sonuçta bütün bu protokoller, standartlar ve programlama disiplinlerinin getirdiği basit ve esnek bir Web servisi geliştirme ortamı yaratır. Geliştirme ortamında protokollerin bilinmesine gerek yoktur. Sadece normal bir ASP sayfası yazarmış gibi servisi ve yine aynı şekilde bunu kullanan sayfayı yazar. Bu da Web servisi geliştirme konusunda büyük bir rahatlık sağlamaktadır. Web servislerine gelince, şu anda görülen tüm uygulamaların internet üzerine taşınacağı ve birbirleri ile konuşmak zorunda olacağı, Web servisleri ise iki probleme de çözüm getirerek geleceğin uygulama standardı olmayı hedeflemektedir.

#### **4.5 Ajax**

AJAX (Asynchronous JavaScript and XML), İnternet sayfalarında JavaScript ve XMLHttpRequest kullanımı ile etkileşimli uygulamalar yaratan tekniğin adıdır. En yaygın kullanım alanı, sayfayı yeniden yüklemeye gerek kalmaksızın, sayfada görünür değişiklikler yapmaktır. XMLHttpRequest kullanılarak birden fazla bağımsız işlem yapılabilir. Bazı bilişim uzmanları, AJAX'ın HTML ve XML'den sonra en yenilikçi İnternet yazılımı olduğunu ve Web 2.0.'ı sonlandırıp, 3. evrenin kapısını açtığını öne sürmüşlerdir. Asynchronous JavaScript and XML sözcüklerinin kısaltması olan Ajax, etkileşimli (interaktif) web uygulamaları yaratmak için kullanılan bir web programlama tekniğidir. Temel amacı arka planda sunucuyla ufak miktarda veri değişimi sayesinde

sayfayı daha hızlı güncellenebilen web sayfaları yapmak, dolayısıyla kullanıcının istediği her anda bütün web sayfasını güncellemek derdinden kurtulmaktır. Bu da web sayfasının etkileşimini, hızını ve kullanılabilirliğini artırmak demektir. Ajax tekniği aşağıdaki teknolojileri kullanmaktadır:

- Bilgiyi biçimlendirmek ve görüntüsünü değiştirmek için XHTML (ya da HTML) ve CSS Görüntülenecek bilgiyi dinamik olarak göstermek ve onunla etkileşimli çalışmak için özellikle JavaScript ve JScript gibi ECMAScript olan kullanıcı tarafındaki bir scripting dili yardımıyla erişilebilen DOM.
- Web sunucusu ile bilgi değiş tokuşunda kullanılan XMLHttpRequest objesi. Bazı Ajax frameworklerde ve bazı durumlarda web sunucusuyla bilgi alışverişi için XMLHttpRequest yerine IFrame'de kullanılır; diğer uygulamalarda ise dinamik olarak eklenen JavaScript TAG'leri de kullanılabilir.
- Kullanıcıyla sunucu arasındaki bilgi alışverişlerinde genelde XML formatı kullanılır. Önceden hazırlanmış HTML, düz metin, JSON ve hatta EBML olmak üzere herhangi bir format da kullanılabilir. Bu tip dosyalar sunucu tarafında herhangi bir script (PHP, ASP vs.) ile bile üretilmiş olabilir.
- DHTML, LAMP ve SPA gibilerin aksine Ajax kendi başına bir teknoloji değildir, ancak terim birçok teknolojinin bir bütün olarak kullanıldığını ifade eder.

#### 4.5.1 Ajax Kullanımının Avantajları

Bantgenişliği ayarlama : HTML'i **Ağ tarayıcısı** (browser) programında oluştururken, JavaScript işlemlerini ve işlenecek verileri kullanırken, sunucudan gelen sayfa bilgisi gerektiğinden daha az yer tuttuğundan dolayı Ajax web sayfalarının görece olarak daha hızlı yüklendiği gözlenebilir. Bu tekniğe örnek olarak bilginin birden çok sayfadan oluştuğu büyük miktarda veriyi ele alalım. Ajax ile sayfanın HTML kodu, örneğin bir tablo kontrolü ya da ilgili TD, TR etiketleri (tag) **ağ tarayıcısı** (browser) içinde

oluşturulabilir, dolayısıyla dökümanın ilk sayfasını indirmeye gerek kalmaz. İçeriğin "isteğe bađlı olarak yüklenmesine" ek olarak, bazı web uygulamaları ilk önce Olay Yöneticilerini (event handler), ardından ilgili fonksiyonları indirir. Bu teknik, karmaşık mekanizması ve fonksiyonu olan web uygulamalarının gereksinim duyduğu yüksek miktarda bantgenişliđi tüketimini önemli ölçüde hafifletmektedir.

**Kullanıcı arabirimi :** Ajax kullanmak için en önemli neden kullanıcı arabirimindeki iyileştirme dir. Ajax kullanan sayfalar tipik bir web sayfasından çok, tarayıcının çalıştığı bilgisayarda bir stand-alone uygulama gibidir. Sayfanın tamamının güncellenmesinin gerektiđi linklere tıklanđıđı zaman işlemin çođunluđu "ađır (hantal)" olduđu hissedilir, ancak Ajax ile sayfa çođu kez dinamik olarak güncellenebilir.

#### 4.5.2 Ajax Kullanımının Dezavantajları

**İndeksleme :** Google ve benzeri site indeksleyen robotlar tarafından henüz ikinci plandadır ve en büyük dezavantajı budur. Ancak içeriđe uygun site haritaları ile içeriğin arama motorları tarafından indexlenmesi sağlanabilir. Bu konuda en iyi yöntem sitenin arama motorları için tasarlanmış ayrı bir sürümünü oluşturmaktır. Herhangi bir server-side programlama dili ile senkronize edildiğinde genelde yazılması gereken kod miktarını neredeyse iki katına çıkarır. Bunun sebebi ajax ile yazılmış fonksiyon ya da class'ın talep cevap sistemi ile çalışması gösterilebilir. Ajax uygulamasının gönderdiđi her bir talep için server tarafında cevap verecek bir dosya olması gerekmektedir.

**Tarayıcı Uyumluluđu:** Dinamik olarak oluşturulan sayfalar tarayıcının önceden işlediđi sayfaları yöneten mekanizmalarına kendini otomatik olarak kaydetmez, bu yüzden tarayıcının "Geri" tuşuna basıldıđında istenen sayfa elde edilemeyebilir. Geliştiriciler bu problemi çözmek için deđişik çözümler geliştirmiştir. Bu tip çözümler genelde tarayıcının Geri tuşu kullanıldıđında geçmiş listesiyle ilgili işlemleri çalıştıran görünmez IFRAME'leri kullanır. Örneğin Google Maps, arama suretiyle istenen bilgileri önce görünmez bir IFRAME içinde işler, sonra sayfanın görünen kısmında

ilgili elemanın içine yerleştirir. World Wide Web Consortium (W3C) iframe elemanını XHTML 1.1 Çalışmasına eklememiştir, bunun yerine object elemanını önermiştir.

**Yanıt Verme Süresiyle İlgili Kısıtlamalar :** Ajax'la geliştirme esnasında network gecikmesi ya da kullanıcı isteğine sunucunun verdiği yanıt arasında geçen süre özellikle dikkate alınmalıdır. Gecikme denen şey, kullanıcıya bilginin düzgün gönderilmemesi ya da XMLHttpRequest objesinin doğru bir şekilde kullanılmaması durumunda kullanıcının beklemediği ya da anlamadığı bir olaydır. Buna ek olarak, bütün bir sayfa render edilirken sayfanın içeriği değişimi esnasında insan gözünün adaptasyonu için geçen belli bir süre vardır. Ekranın değişiminin daha ufak adımlarla ayarlanması yapılmazsa bu gecikme daha çok hissedilir. Kullanıcıya arka planda bir işlemin yapıldığı ya da içeriğin yüklenmeye çalışıldığını söylemek için kullanılan görsel efektler bu tip gecikme problemlerine çözüm olarak önerilmektedir. Genel olarak network gecikmesinin zamana bağlı olarak değişmesi gibi gecikmenin potansiyel etkileri, hâlihazırda bulunan açık kaynak Ajax araçlarının ve framework'lerin hiçbirince henüz çözümlenememiştir.

**Arama Motorlarıyla İlgili İyileştirmeler :** Arama motorlarınca indekslenmesi gereken bilgileri yüklemek için Ajax kullanılan web sistemleri, Ajax işlevselliği için gerekli olan JavaScript kodunu arama motorlarının genelde çalıştırmamasından dolayı, arama motorlarının okuyabileceği bir formatta ve bir URL linki şeklinde bilgi koymalıdır. Bu problem Ajax'a özgü değildir. Örneğin form gönderme (form submit) sonrası gereken tam sayfa güncellemesi gibi dinamik sayfalar üreten sitelerde de bu problem vardır (bu gibi problemler genelde hidden web olarak adlandırılır).

**Erişilebilirlik :** Web uygulamalarında Ajax teknolojilerini kullanmak, Bilgisayar erişilebilirliği konusunda araçlar geliştiren tasarımcılar için birçok sorunu beraberinde getirmektedir. Buna ek olarak Section 508 gibi Amerikan Uyum Standartlarına sıkı bir şekilde uyulması gerektiği ABD devlet destekli birçok geliştirme projeleri bulunmaktadır. Bu standartlara uyulmaması, bazı durumlarda bilgisayar uyumluluğuyla ilgili birçok iş anlaşmalarının iptaline ve hukuki problemlerin doğmasına neden olmaktadır. Bu yüzden Ajax'la yapılmış çözümlerin büyük bir kısmı grafik tabanlı

tarayıcılara hitap ettiğinden dolayı, geliştiriciler başka platform ve tarayıcı kullanıcılarının da gereksinimlerini karşılamak zorundadır. Web tasarımcıları, web tarayıcısının bütün sayfayı renderlemesine gerek kalmaksızın veri işlemeye olanak tanındığı durumlarda bazen web sayfasının sadece belli kısımları için Ajax'ı kullanabilir. Ajax kullanamayanlar ise bütün bir sayfayı yenilemeye ve yüklemeye devam edebilmeli; geliştiriciler ise Ajax'ı destekleyen taracılarla daha teknik olanaklar sunsa da -bütün erişilebilirlik konuları dahil olmak üzere- Ajax içermeyen ortamlardaki kullanıcıların yeteneklerini kısıtlamamalıdır.

#### **4.6 Uzaktan Öğrenim Yönetim Sistemleri**

Öğrenim Yönetim Sistemleri (Learning Management System-LMS); kullanıcı ile öğretim materyalleri arasındaki aktiviteleri izleyen, yöneten ve raporlayan yazılımlardır. Günümüzde uzaktan eğitimin yaygınlaşmasında en büyük etken çeşitli Öğrenim Yönetim Sistemleri nin yaygınlaşmasıdır. Bunların başında MOODLE gelmektedir. MOODLE Toplam 75 ülkede, 77 ayrı dil desteği ve Toplam 75000 kullanıcısı ile en yaygın Eğitim Öğretim Yönetim Sistemidir. Bu eğitim Yönetim Sisteminin Yaygınlaşmasının en büyük nedenlerinden bir tanesi Açık Kaynak Kodlu Olmasıdır. Bu özellik moodle sisteminin ihtiyaçlara göre geliştirilmesini sağlamaktadır. Moodle sisteminin bir diğer avantajı ise modüler yapıda olmasıdır. Ancak moodle üniversitelerin ihtiyacına tam olarak cevap vermemektedir. Daha çok kursların uzaktan eğitimle verilmesine uygun bir eğitim yönetim sistemidir. Diğer dezavantajı ise açık kaynak kodlu olması bir dizi güvenlik açıklarını da beraberinde getirmektedir.

Günümüz Uzaktan Öğrenim Yönetim Sistemlerinde 2 önemli nokta ön plana çıkmaktadır. Bunlar; Modüler yapıda olması ve SCORM uyumlu olmasıdır.



## **4.7 Modüler Sistem**

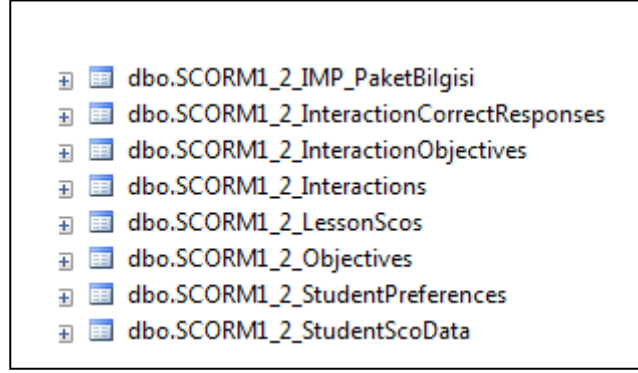
Bilgisayar teknolojilerinin geçmişine bakılırsa; bir zamanlar programlama dilleri ile bir yazılım üretmek, kitap yazmaktan farksızdı. Günümüzde bu tür problemleri ortadan kaldırmak için küçük parçalardan oluşan modüler yapıdaki programlar üretilmeye başlanmıştır. Daha karmaşık yapılar, daha hızlı ve kolay üretilir olmuştur. Modüler yapıyı baz alan programlama dilleri, nesne tabanlı programlama dilleri olarak adlandırılmaktadır. Modüler Sistem, bütünü en yararlı olabilecek parçalara bölerek tümevarım yöntemiyle bütünü tamamlamaktır. Modüler sistemde her modül belli bir iş yüküne sahiptir. Modülün iş yükü programın tamamı düşünülerek ve ilgili modül için beklenen görevler göz önüne alınarak hesaplanır. Nesne tabanlı programlama dillerinde yaşanan bu gelişmeler kitap içeriği gibi hazırlanan, puzzle parçaları gibi sadece bulunduğu yerde anlamlı içerik parçalarından oluşan ve bu şekilde programlanan e-öğrenme yazılımlarını da olumlu etkilemiştir. İşte burada birbiri ile kolayca birleşebilen, LMS ile kolayca etkileşebilen ve hızlı çalışabilen modüler bir yapı ortaya konmaktadır.

## **4.8 SCORM Yapısının LMS'ye Uyarlanması**

Scorm 1.2 ve Scorm 2004 api'nin LMS'ye uygulanması için bazı değişiklikler yapılması gerekmektedir. Bu değişiklikler SQL veri tabanında ve IIS üzerinde yapılması gereken değişikliklerdir. Bunlar aşağıda verilmiştir.

### **4.8.1 SCORM 1.2 API nin LMS'ye Uyarlanması**

Daha önceki konularda bahsedildiği gibi SCORM 1.2 için kullanılan toplam 8 tablo bulunmaktadır. Bu tablolar Şekil 4.2 de verilmiştir.



**Şekil 4. 2** SCORM 1.2 tabloları.

SCORM 1.2 tabloları LMS için kullanılan aynı veri tabanına import edilmiştir. Sisteme SCORM 1.2 uyumlu içerik eklenmesi 8. Modül olan dersler modülünde anlatılmaktadır. Bölüm Sekreteri ilgili bölüme yeni ders ekleme işlemi yaptığında sistem bu ders için eklenmek istenen ders paketinin türünü sormaktadır. Şekil 4.3’de sisteme yeni ders ekleme işlemi örnek olarak verilmiştir.

Bolum Seç Bilgisayar Bolumu ▼

  
Yeni Ders  
Ekle

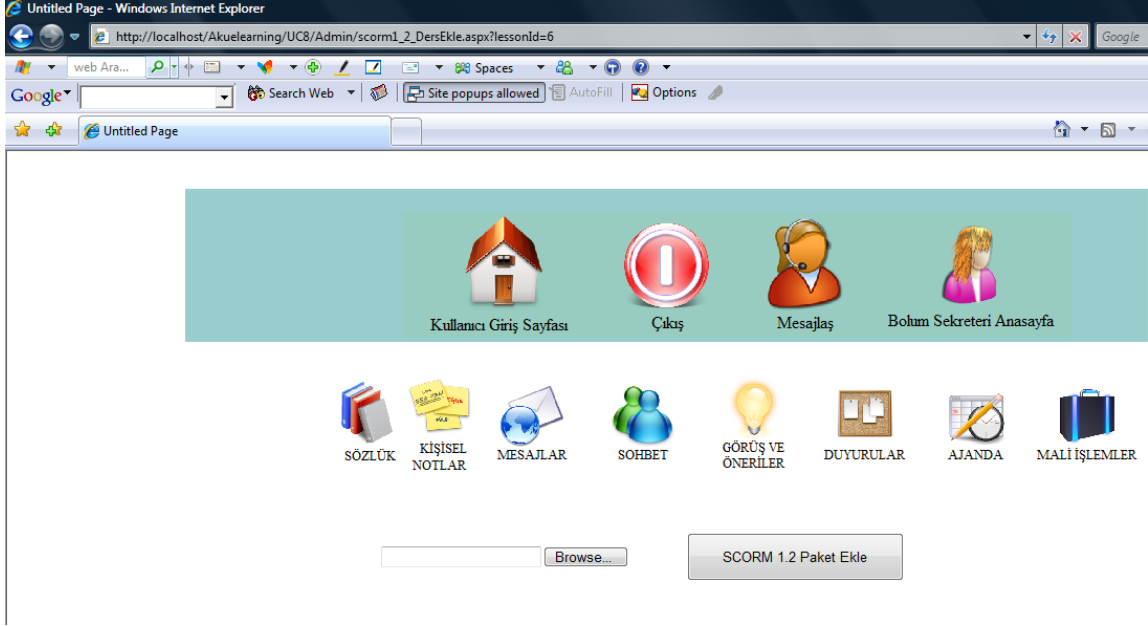
<b>Bolum Adi</b>	
<b>Ders Adi</b>	Araştırma Teknikleri 2
<b>Ders Kod</b>	6
<b>Ders Detay</b>	Bilgisayar Kontrol Sistemleri Yüksek lisans dersidir. Bu dersi alabilmek için Arastirma Teknikleri 1 dersinden başarılı olmak gerekir.
<b>Ders Tipi</b>	LMS
<b>Ders icerik Adres</b>	-
<b>Ders Hedef</b>	Arastirma Becerisi Kazandırmak

Ekle Cancel

	<u>Ders No</u>	<u>Ders Adi</u>	<u>Bolum Adi</u>	Scorm 1.2 İçerik Ekle	Scorm 2004 İçerik Ekle	
<u>Seç</u>	6	Arastirma Teknikleri	Bilgisayar Bolumu	<a href="#">Ekle</a>	<a href="#">Ekle</a>	<a href="#">Sil</a>
<u>Seç</u>	7	Fizik	Bilgisayar Bolumu	<a href="#">Ekle</a>	<a href="#">Ekle</a>	<a href="#">Sil</a>
<u>Seç</u>	8	Algoritma Tasarımı	Bilgisayar Bolumu	<a href="#">Ekle</a>	<a href="#">Ekle</a>	<a href="#">Sil</a>

Şekil 4. 3 Programa yeni ders ekleme.

Eklenen paket SCORM 1.2 uyumlu içerik paketi ise seçilen dersin ID'sini alarak Şekil 4.4'de verilen SCORM 1.2 içerik paketi ekleme sayfasına gitmektedir. Eklenen paketin ait olduğu ders sistemde bu ID ile tanımlanmaktadır.



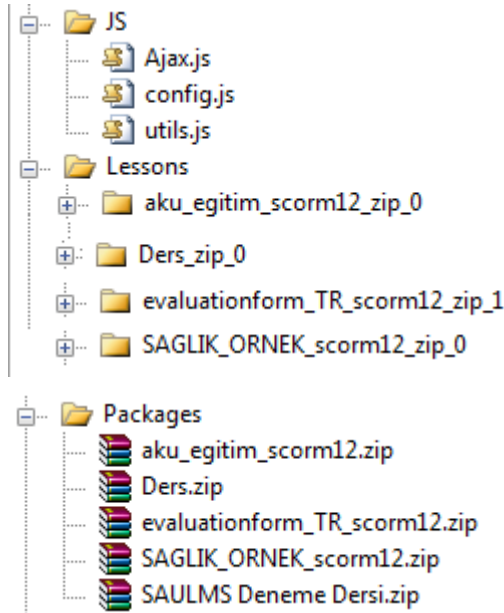
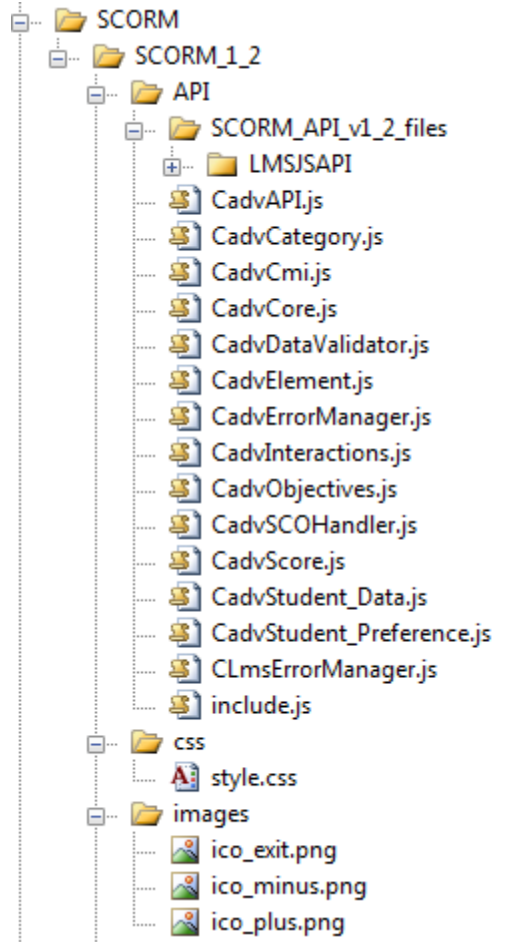
Şekil 4. 4 Sisteme SCORM 1.2 uyumlu içerik paketi ekleme.

Sisteme eklenen paket ile ilgili Activite ID ve Ders ID'si öncelikle veritabanında bulunan SCORM1\_2\_IMP\_PaketBilgisi tablosuna Şekil 4.5'deki gibi kaydedilmektedir. Bu tablodaki lessonId alanı eklenene paketin ait olduğu dersi belirten ID, foldername alanı ders paketinin ismini belirtmektedir.

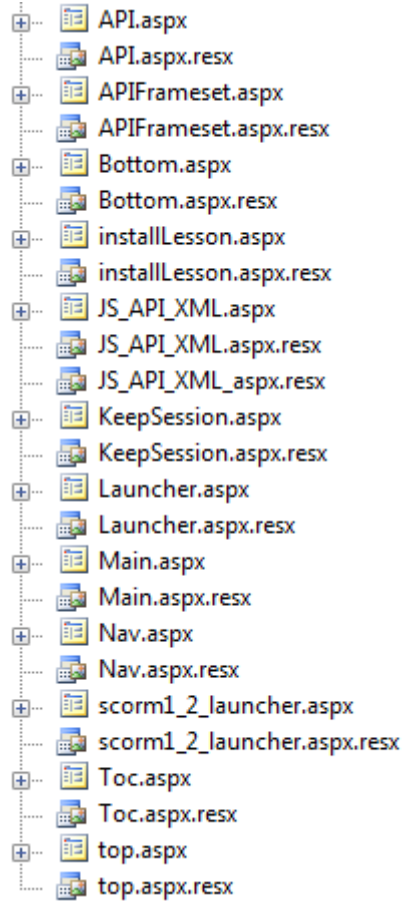
Table - dbo.SCORM1_2_IMP_PaketBilgisi		Summary		
	scormPaketId	folderName	lessonId	activiteId
▶	14	SAULMS Deneme Dersi_zip_0	5	5
	15	aku_egitim_scorm12_zip_0	5	5
	16	SAGLIK_ORNEK_scorm12_zip_1	5	5
	17	SAGLIK_ORNEK_scorm12_zip_2	5	5
	18	evaluationform_TR_scorm12_zip_1	12	5
	19	evaluationform_TR_scorm12_zip_2	12	25
	20	evaluationform_TR_scorm12_zip_3	11	5
	21	aku_egitim_scorm12_zip_1	11	25
	22	aku_egitim_scorm12_zip_2	11	25
	23	aku_egitim_scorm12_zip_3	12	25
	24	aku_egitim_scorm12_zip_4	11	25
	25	SAGLIK_ORNEK_scorm12_zip_3	6	25

**Şekil 4. 5** SCORM1\_2\_IMP\_PaketBilgisi tablosu.

Scorm 1.2 için oluşturulan bütün dosyalar Akuelearning projesi içindeki SCORM klasöründe toplanmaktadır. SCORM 1.2 API dosyaları bu klasör içindeki SCORM\_v1\_2\_files klasöründe yer almaktadır. Sisteme eklenen SCORM 1.2 ders içerik paketleri Package klasöründe tutulmaktadır. Ayrıca eklenen ders içerik paketlerinin içindeki dosyalar Lessons klasörüne çıkartılmaktadır. AkuElearning projesi içinde yeralan Scorm 1.2 dosyaları Şekil 4.6’da verilmiştir.



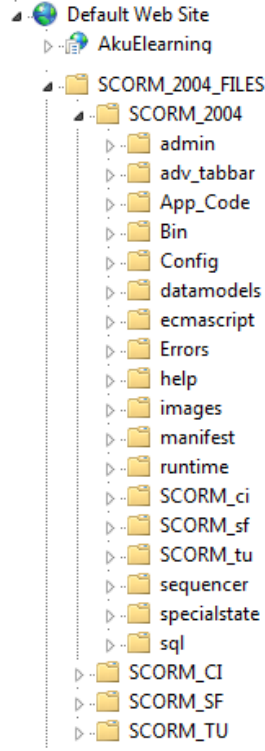
Şekil 4. 6 SCORM 1.2 dosyaları.



Şekil 4.6(Devam) SCORM 1.2 dosyaları.

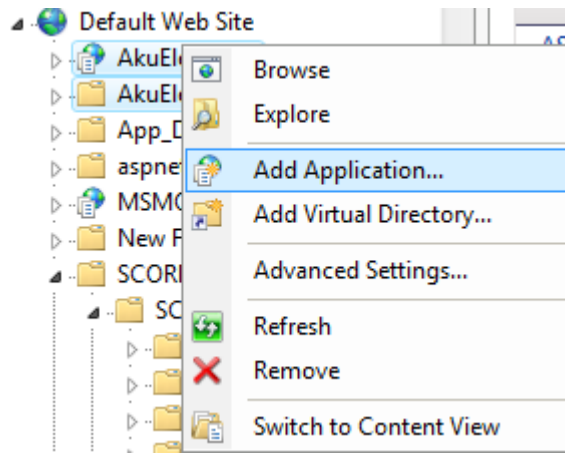
#### 4.8.2 SCORM 2004 API'nın LMS'ye Uyarlanması

SCORM 2004 ün LMS üzerinde çalıştırılması SCORM 1.2'ye göre biraz farklılık göstermektedir. SCORM 1.2 dosyalarını AkuElearning projesine eklerken bu dosyaları projenin içerisine kopyalamak yeterli olmaktadır. Ancak SCORM 2004 ayrı bir proje ve ürün olarak tasarlanmıştır. Bu dosyaları sisteme eklemek için IIS üzerinde bazı değişiklikler yapılması gerekmektedir. Şekil 4,7'de de görüldüğü gibi bu dosyalar wwwroot dizini içinde ayrı bir proje olduğu için Local klasör olan AkuElearning klasöründen dış klasöre erişilmek istendiğinde sorun olmaktadır. Aşağıda Vista üzerindeki IIS tarafından bu işlemlerin nasıl gerçekleştirilmesi anlatılmaktadır.



**Şekil 4. 7** wwwroot dizini.

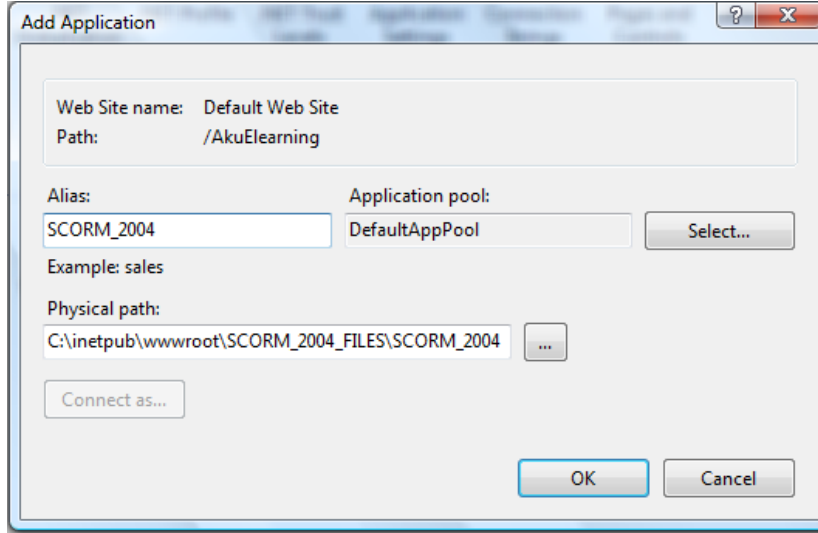
SCORM\_2004\_FILES klasörü içerisinde yer alan SCORM\_2004 dosyalarını sanal dizin olan AkuElearning projesi içine ayrı bir proje olarak eklenmesi gerekmektedir. Bunun için AkuElearning projesi üzerine mouse imlecini getirdikten sonra mouse'un sağ tuşuna tıklanması gerekmektedir. Şekil 4.8'de de verilen pencereden ise Add Application seçeneğinin seçilmesi gerekmektedir.



**Şekil 4. 8** SCORM 2004 dosyalarının sanal dizin içinde çalıştırılması.

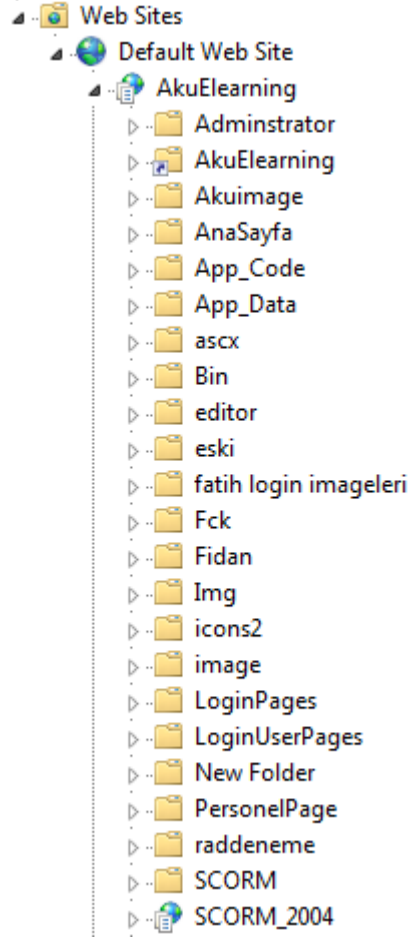


Ekrana gelen Add Application penceresindeki Physical Path bölümünde SCORM\_2004 klasörünün seçilmesi ve Alias ismine de SCORM\_2004 adının verilmesi gerekmektedir. Bu işlemler Şekil 4.9’da verilmiştir.



**Şekil 4. 9** Alias isminin ve Physical path'in belirlenmesi.

Yapılan işlemlerden sonra SCORM\_2004 Files klasörünü IIS local dizini içine katacağıdır. SCORM\_2004 klasörü Şekil 4.10' daki gibi AkuElearning projesi içinde ayrı bir proje olarak çalıştıracaktır.



**Şekil 4. 10** AkuElearning Projesi içinde SCORM\_2004 projesinin çalıştırılması.

Sisteme SCORM 2004 ders paketinin eklenmesi, 8. Modülde anlatılmıştır. SCORM 2004 tabloları ise LMS için kullanılan veri tabanına taşınmıştır. Bu tablolar ile ilgili açıklamalar SCORM konusu içinde ayrıntılı olarak anlatılmıştır.

#### **4.9 Hazırlanan Eğitim Yönetim Sisteminin Temel Modülleri**

Aşağıda hazırlanan modüller kısa açıklamalarıyla beraber verilmiştir.

1. **Online başvuru Modülü**; Kullanıcıların sorunsuz şekilde online başvuru işlemini gerçekleştirebilmesi ve takip edebilmesini sağlamaktadır.
2. **Kullanıcılar ve kullanıcı grupları Modülü**; Eklenen kullanıcının sisteme erişebiliyor ve yetkisi olan alanları kullanabiliyor olmasını sağlamaktadır. Silinen kullanıcının sisteme erişemiyor olmasını, güncellenen kullanıcının bilgilerinin tüm

sistemde etkin hale gelmesini, eklenen kullanıcı grubunun sistemde tanımlı olmasını ve bu gruba kullanıcılar eklenebiliyor olmasını sağlamaktadır. Güncellenen kullanıcı grubu bilgilerinin ise tüm sistemde yeni halini almasını sağlamaktadır.

3. **Rol ve yetkiler modülü;** Oluşturulan rolün sistemde tanımlı ve erişilebiliyor olması, kullanıcılara bu rolün atanabilmesini ve güncellenen rolün sistemde yeni halini almasını sağlamaktadır.
4. **Ünvan tanımlama modülü;** Eklenen ünvanların sistemde tanımlı ve erişilebilir olmasını sağlamaktadır.
5. **Okul Tanımlama Modülü;** Eklenen okulların sistemde tanımlı ve erişilebilir olmasını sağlamaktadır.
6. **Bölüm Tanımlama Modülü;** Eklenen bölümlerin sistemde tanımlanması ve erişilebilir olmasını sağlamaktadır.
7. **Eğitim Kataloğu Modülü;** Yeni programların sisteme başarılı bir şekilde kaydedilmesini ve kullanıcıların sistemdeki tüm programları görebilmesini sağlamaktadır.
8. **Dersler Modülü;** Ders içeriklerinin sisteme başarılı bir şekilde kaydedilmesini ve kullanıcıların sistemdeki derslerini görebilmesini sağlamaktadır.
9. **Haftalık Ders Programı Modülü;** Sisteme girilen saatlerin kullanıcılar tarafından erişilebiliyor olmasını ve Sohbet Saatinin haftalık ders programında görülebilir olmasını sağlamaktadır.
10. **Şube İşlemleri Modülü;** Derslerin şube düzenlemesini, şubelere ders sorumlusu atama işlemlerini ve şubeye öğrenci ekleme ve şube bilgilerini düzenleme işlemlerinin yapılmasını sağlamaktadır.
11. **Danışman Atama İşlemleri Modülü;** Danışmanların bölümlere atanabiliyor olmasını, sistemde tanımlanmış ve erişilebilir olmasını, kendilerine atanan programları görebilmesini sağlar. Ayrıca danışmanı oldukları öğrencileri görebilmesini, ders kayıt işlemlerini yapabilmesini sağlamaktadır.
12. **Ders Kayıt ve Ders Onay İşlemleri Modülü;** Dersleri seçen ve seçmeyen öğrencilerin takip edilmesini, ders seçimini yapan öğrencilerin, ders onaylama işlemini yapmasını ve dönemine ait dersleri seçmesini sağlamaktadır.
13. **Soru Tanımlama Modülü;** Eklenen soruların sistemde tanımlı ve erişilebilir olmasını sağlamaktadır.

14. **Sınav Modülü;** Ders sorumlusunun ilgili dersin sınavını hazırlayabilmesini, sistemde tanımlı soruları kullanarak sınav hazırlayabilmesini ve öğrencinin ilgili dersine atanmış sınavları yanıtlayabilmesini sağlamaktadır.
15. **Deneme Sınavı Modülü;** Öğrencinin deneme sınavını hazırlayıp kullanabilmesini sağlamaktadır.
16. **Ödevler Modülü;** Yeni ödevin sisteme eklenmesi ve ilgili kullanıcılar tarafından görülüyor olmasını sağlamaktadır. Ayrıca girilen ödevin sildiğinde programda bir daha gözükmeyecek şekilde sistemden kaldırılmasını sağlamaktadır. Kullanıcıların ise kendisine ait ödevleri görebiliyor olmasını, yapmış oldukları ödevleri gönderebilmesi ve ödev veren kişiye ulaşmasını sağlamaktadır.
17. **Öğretim Elemanı Performans Takip Modülü;** Öğretim elemanını, vermiş olduğu dersin bilgilerine göre değerlendirebilmesini sağlamaktadır.
18. **Öğrenci Performans Takip Modülü;** Öğrencinin kendi notları ile birlikte performans notlarını listeleyebilmesini ve Öğrencinin ders sorumlusu hakkında değerlendirmelerini yapabilmesini sağlamaktadır.
19. **Öğrenci Takip Raporu Modülü;** Öğretim üyesinin şubelerdeki öğrencilerini ilgili derslerinde, performans notu ile değerlendirmelerini yapabilmesini sağlamaktadır.
20. **Sözlük Modülü;** Yeni oluşturulan sözlüğün ve kayıt yapılan sözcüğün sisteme eklenebilir ve kullanıcılar tarafından erişilebiliyor olmasını sağlamaktadır. Ayrıca kullanıcıların aradığı sözcük sistemde mevcutsa, bilgilerine ulaşabiliyor olmasını sağlamaktadır. Bilgileri güncellenmek istenen sözcüğün yeni girilen değerler ile sisteme kaydedilmesini, sistemden silinmek istenen sözcüğün programda bir daha gözükmeyecek şekilde sistemden kaldırmasını, sisteme import edilen verilerin ise sisteme eklenmesini sağlamaktadır.
21. **Anket Modülü;** Eklenen anketin sistemde tanımlı ve erişilebilir olmasını ve kullanıcı aktif olan anketleri görebilmesi ve anketi kullanabilmesini sağlamaktadır.
22. **Kişisel Notlar Modülü;** Eklenen notun sistemde tanımlı ve ekleyen kullanıcı tarafından erişilebilir olmasını sağlamaktadır.
23. **Mesajlaşma Modülü;** Gönderilen mesajları ilgili kişinin görebilmesini ve kullanıcının gelen mesajlarını görebilmesini sağlamaktadır.
24. **Dosyalar Modülü;** Öğretim görevlilerinin vermiş olduğu derslerle ilgili dokümanları sisteme ekleyebilmesini sağlamaktadır. Dersi alan kişilerin ise bu

dokümanlara erişebiliyor olmasını, arşivdeki dosyaları sildiğinde sistem üzerinden tamamen silinmesini ve aldığı derslerle ilgili dokümanlara ulaşabiliyor olmasını sağlamaktadır.

25. **Sohbet Saatleri Modülü;** Sisteme girilen saatlerin kullanıcılar tarafından erişilebiliyor olmasını ve Sohbet Saatinin haftalık ders programında görülebilir olmasını sağlamaktadır.
26. **Toplantı Modülü;** Oluşturulan toplantıların davet edilen kullanıcılar tarafından görülebilir olmasını, silinen toplantıların davet edilen kullanıcılara iptalinin bildirilmesini ve davetli olduğu toplantıları görüp, katılabiliyor olmasını sağlamaktadır.
27. **Sıkça Sorulan Sorular Modülü;** Kategori işleminde, kategorilerin sisteme eklenebilir ve erişilebilir olmasını, kategori silme işleminde ise sistemdeki o kategori ve altındaki soruların tümüyle silinebilmesini sağlamaktadır. Ayrıca kategoriler halinde soruların düzenlenebilmesini, sisteme girilen bilgilerin kullanıcılar tarafından erişilebiliyor olmasını ve kategoriler halinde sorulara ulaşabilmeyi sağlamaktadır.
28. **Görüş ve Öneriler Modülü;** Kullanıcılar tarafından gönderilen görüş ve önerileri onayladığında sistemde yayınlanmasını, kullanıcıların, görüş veya önerilerini gönderebilmesini ve onaylanan görüş ve önerileri görebilmesini sağlamaktadır.
29. **Duyurular Modülü;** Yeni oluşturulan duyurunun ve duyuru tiplerinin sisteme eklenmesini ve kullanıcılar tarafından görülüyor olmasını sağlamaktadır. Dönem sonunda ise o dönemle ilgili tüm duyuruların otomatik olarak, programda bir daha gözükmeyecek şekilde sistemden kaldırmasını ve kullanıcıların duyuruları görebiliyor olmasını sağlamaktadır.
30. **Ajanda Modülü;** Eklenen hatırlatma notunun sistemde tanımlı ve erişilebilir olmasını sağlamaktadır.
31. **Akademik Takvim Modülü;** Akademik takvimin sisteme başarılı bir şekilde kaydedilmesini ve kullanıcıların akademik takvimi görebilmesini sağlamaktadır.
32. **Trasnkript Görüntüleme Modülü;** Sistemde kayıtlı öğrencilerin transkript bilgilerini görebilmesini ve yetkisi dahilinde sistemde mevcut transkriptleri görebilmesini sağlamaktadır.

33. **Kullanıcıların yetkileri kapsamında hazır raporları alabilmesi;** Öğrenci Belgesi, Öğrenci transkripti, Öğrenci Takip raporu, Öğr. Üyesi takip raporunu hazırlamayı sağlamaktadır.
34. **Mali İşlemler Modülü;** Kullanıcıların sorunsuz şekilde ödemelerini yapabilmesini sağlamaktadır.
35. **Demo Kullanıcı Modülü;** Demo kullanıcıların sistemde yetkisi olan bölümlere erişebiliyor olmasını sağlamaktadır.
36. **Workshop Modülü;** Workshopların sistem üzerinden sorunsuz gerçekleştirilebilmesini sağlamaktadır.
37. **Sanal Sınıf Modülü;** Workshopların sistem üzerinden sorunsuz gerçekleştirilebilmesini sağlamaktadır.

## 5. SCORM UYUMLU MODÜLER LMS'İN TASARLANMASI VE GERÇEKLEŞTİRİLMESİ

Hazırlanan uzaktan eğitim projesine, ilk olarak üniversite geneli bir proje kapsamında başlanılmıştır. Yazılım, içerik ve tasarım olmak üzere 3 gruba oluşturulmuştur. Oluşturulan gruplar mesleki bilgi ve deneyimlere göre oluşturulmuştur. Grupların yapması gereken görev ve yetkilere göre BilgeAdam firması tarafından ileri düzeyde kurslar verilmiştir. Bu kurslar içerik grubu için Öğretimsel tasarım, Photoshop, SCORM, Flash, Illustrator ve Fireworks, yazılım grubu için yazılım geliştirme kursu, tasarım grubu içinse Photoshop ve flash kurslarıdır. Yaklaşık 5 haftalık bir eğitim sürecinin sonunda yazılım grubu analiz raporunu hazırlamış, İçerik grubu Araştırma Teknikleri ve Bilimsel İletişimin Temelleri dersinin senaryosuna başlamış ve flash ile bu senaryonun içeriklerini geliştirmiştir. Tasarım grubu ise LMS'in ve Player ekranının tasarımını gerçekleştirmiştir. Ancak daha sonra projenin kapsamı üniversite geneli bir proje olmaktan çıkarılıp tez projesi haline dönüştürülmüştür. Yazılımın ilk 19 modülü bu tez paralelinde, diğer 18 modül ise Fatih BAYRAM (Bayram 2009) tarafından hazırlanan yüksek lisans tezi paralelinde kodlanmış ve ayrıntılarıyla anlatılmıştır.

Ayrıca Fidan HAKKARİ tarafından hazırlanarak yüksek lisans tezi paralelinde "Araştırma Teknikleri ve Bilimsel İletişimin Temelleri" dersinin senaryosu hazırlanarak Flashla animasyonlaştırılmıştır (Hakkari 2009). Hazırlanan bu içerikler de SCORM standartlarına uygun hale getirilerek LMS'e entegre edilmiştir. Bir projenin parçaları olarak hazırlanan bu üç tez bir biri ile entegre olarak çalıştırılıp sonuçlandırılmıştır. Aşağıda ilk 19 modül ayrıntılı bir şekilde açıklanmıştır. Diğer 18 modül ise EK-1'de özet olarak verilmiş, detaylı anlatımı ise Fatih BAYRAM'ın hazırladığı (Bayram 2009) yüksek lisans tezinde yer almaktadır.

## 5.1 Online Başvuru Modülü

Bu modül, lisans yada yüksek lisans öğrenimi görmek isteyen öğrencilerin internet üzerinden başvurularını yaptığı modüldür. Modüller hazırlanırken üzerinde en çok durulan modüllerden birisidir. Bu modülde yapılabilecek hataları minimum seviyeye indirmek ve daha hızlı kayıt imkânı sağlamak için Ajax teknolojisi kullanılmıştır. Online kayıt modülü başlıca 5 aşamadan oluşmaktadır.

- 1) Öğrenci online başvuru ekranı yardımı ile başvurusunu tamamlar.



- 2) Bölüm Sekreteri, başvurusu uygun olan öğrencilere mülakat bilgilerini E-posta ile gönderir.



- 3) Mülakata gelen öğrencilerin yanında getirmesi gereken evrakları, mülakattan önce alır ve eksik evrakı bulunmayan öğrencileri, mülakat için onaylar.



- 4) Bölüm Başkanı, mülakata katılan öğrencilerin mülakat notlarını girer.



- 5) Öğrenci işleri kazanan öğrencileri onaylar.

### 5.1.1 Online Başvuru Ekranları

Online başvuru ekranları toplam 6 aşamadan oluşmaktadır.

1. Kişisel Bilgiler Ekranı
2. İletişim Bilgileri Ekranı
3. İş Bilgileri Ekranı
4. Eğitim Bilgileri Ekranı



5. Sınav Bilgileri Ekranı
6. Başvuru Yapılacak Program Seçimi Ekranı

### 5.1.1.1 Kişisel Bilgiler Ekranı

Bu ekranda doldurulacak kısımlar Çizelge 5.1’de verilmiştir.

**Çizelge 5. 1** Kişisel Bilgiler Ekranında doldurulacak bilgiler.



Çizelge 5.1 de gösterildiği gibi başvuru ekranında aşağıdaki bilgiler doldurulmaktadır.

1. Ad
2. Soyad
3. Uyruk

Bu 3 alan bütün kullanıcılar için ortaktır.

**Uyruk Türkiye ise nüfusa kayıtlı olduğu il ve ilçe seçimi de ekranda yer alır.**

4. Nüfusa Kayıtlı Olduğu İl
5. Nüfusa Kayıtlı Olduğu İlçe
6. Vatandaşlık Numarası
7. Cinsiyet
8. Doğum Tarihi

**Uyruk Türkiye ve cinsiyet erkek ise, askerlik bilgileri ekranda yer alır.**

9. Kayıtlı Olduğu Askerlik Şubesi Seçilir.
10. Askerlik Durumu Belirtilir. 3 Adet Askerlik Durumu bilgisi vardır.

- Yaptım
- Erteledim
- Muafim

**Askerlik durumu erteledim ise ekranda erteleme bitiş tarihi görünür.**

11. Erteleme Bitiş Tarihi Girilir.

Kişisel Bilgiler ekranındaki seçeneklerde uyruk Türkiye seçilmemişse Şekil 5.1'deki bilgileri doldurması gerekmektedir.



A - KİŞİSEL BİLGİLERİNİZ (Detayları Gizle...)

<b>Ad (*) :</b>	<input type="text" value="Ayşe"/>
<b>Soyad(*) :</b>	<input type="text" value="Demir"/>
<b>Uyruđu(*) :</b>	<input type="text" value="Almanya"/>
<b>Vatandaşlık No(*) :</b>	<input type="text" value="12123354433"/>
<b>Doğum Tarihi</b>	<input type="text" value="09.03.1985"/>
<b>Cinsiyet(*):</b>	<input type="radio"/> Erkek <input checked="" type="radio"/> Bayan

Şekil 5. 1 Uyrık Türkiye değılse kişisel bilgiler ekranında doldurulacak alanlar.

Kişisel bilgiler ekranındaki seçeneklerde uyrık Türkiye, cinsiyet erkek, askerlik durumu erteledim ise Şekil 5.2’de verilen bilgileri doldurması gerekmektedir.

☑ A - KİŞİSEL BİLGİLERİNİZ (Detayları Gizle...)

<b>Ad (*) :</b>	<input type="text" value="Emin"/>
<b>Soyad(*) :</b>	<input type="text" value="İbili"/>
<b>Uyruğu(*) :</b>	<input type="text" value="Türkiye"/>
<b>Nüfusa Kayıtlı Olduğu İl(*) :</b>	<input type="text" value="AFYON"/>
<b>Nüfusa Kayıtlı Olduğu İlçe (*) :</b>	<input type="text" value="MERKEZ"/>
<b>Vatandaşlık No(*) :</b>	<input type="text" value="16934125798"/>
<b>Doğum Tarihi</b>	<input type="text" value="24.03.1982"/>
<b>Cinsiyet(*) :</b>	<input checked="" type="radio"/> Erkek <input type="radio"/> Bayan
	<input type="text" value="AFYON"/>
<b>Kayıtlı Olduğu Askerlik Şubesi :</b>	<input type="text" value="MERKEZ ASKERLİK Ş"/>
<b>Askerlik Durumu:</b>	<input type="radio"/> Yaptım <input checked="" type="radio"/> Erteledim <input type="radio"/> Muafım
<b>Erteleme Bitiş Tarihi</b>	<input type="text" value="01.07.2010"/> 

Şekil 5. 2 Uyrak Türkiye ise doldurulacak bilgiler.

### 5.1.1.2 İletişim Bilgileri Ekranı

İletişim bilgileri kayıt ekranında doldurulması gereken bilgiler;

1. Bulduğunuz Ülke
2. Bulduğunuz İl
3. Bulduğunuz İlçe
4. E-posta adresi

5. İkinci e-posta adresi
6. Ev Telefonu
7. Cep telefonu
8. Açık Adres.

Şayet seçilen ülke Türkiye ise bulunduğunuz il, bulunduğunuz ilçe seçenekleri ekranda görüntülenmektedir. Aksi takdirde bu bölümler görüntülenmemektedir. Telefon Alan Kodları ise ülke seçiminin ardından otomatik olarak ev telefonu ve cep telefonu kısımlarının yanına eklenmektedir. Şekil 5.3 ve Şekil 5.4'te iletişim bilgileri formunun örnek ekran görüntüsü verilmiştir.



🔒 **B-İLETİŞİM BİLGİLERİNİZ** (Detayları Gizle...)

<b>Bulduğunuz Ülke (*) :</b>	<input type="text" value="Türkiye"/>
<b>Bulduğunuz İl (*) :</b>	<input type="text" value="AFYON"/>
<b>Bulduğunuz İlçe (*) :</b>	<input type="text" value="MERKEZ"/>
<b>E-Posta1(*):</b>	<input type="text" value="emin_ibili@hotmail.com"/>
<b>E-Posta2(*):</b>	<input type="text" value="akulms@aku.edu.tr"/>
<b>Tel Ev(*) :</b>	<b>Örn(272) 254-61xx</b> <input type="text" value="+90 (272) 25461**"/>
<b>Tel Cep(*) :</b>	<input type="text" value="+90 (555) 21729**"/>
<b>Açık Adres (*) :</b>	<input style="height: 40px;" type="text" value="Susuz Kasabası Sakaraya Mahallesi Milli Birlikte Caddesi Gümüş Apartmanı No:01 Afyonkarahisar"/>

Şekil 5. 3 Bulduğu ülke Türkiye ise doldurulacak alanlar.



**B-İLETİŞİM BİLGİLERİNİZ** (Detayları Gizle...)

**Bulduğunuz Ülke (\*) :** Avusturya

**E-Posta1(\*) :** emin\_ibili@hotmail.com

**E-Posta2(\*) :** akulms@aku.edu.tr

**Tel Ev(\*) :** **Örn(272) 254-61xx**  
+43 (484) 070-09\*\*

**Tel Cep(\*) :** +43 (484) 289-65\*\*

**Açık Adres (\*) :** Krottendorf 8 Kapfenberg  
Hause Number :2  
Postcode:21  
Austria

**Yanlış Telefon Numarası Girişi**  
Telefon Numaranızı Standartlara Uygun Olarak Giriniz (###) ###-####

**Şekil 5. 4** Bulduğu ülke Türkiye değil ise doldurulacak alanlar.

Şekil 5.4’de de görüldüğü gibi Online Başvuru Modülünde öğrencinin hatasız bir şekilde kaydolması için AjaxControlToolkit kullanılmıştır. Bu teknoloji ile öğrencinin yapmış olduğu işlemler adım adım kontrol edilmektedir. Öğrenci gerekli verileri girmedeği takdirde yada yanlış formatta girdiğinde AjaxControlToolkit ile bu hatalar kontrol edilmekte ve kullanıcıya hata mesajı gönderilmektedir.

### 5.1.1.3 İş Bilgileri Ekranı

Öncelikle kayıt yaptırmak isteyen adayın her hangi bir iş yerinde çalışıp çalışmadığı öğrenilir. Şayet aday çalışmıyorsa Şekil 5.5’deki gibi “Hayır Çalışmıyorum” seçeneğini seçmesi gerekmektedir.



☑ C-İŞ BİLGİLERİ(Detayları Gizle...)

**Herhangi Bir İşyerinde Çalışıyormusunuz(\*)**

- Evet Çalışıyorum  
 Hayır Çalışmıyorum

Şekil 5. 5 İş Bilgileri ekranındaki ilk soru.

Seçim Evet Çalışıyorum ise aşağıdaki kısımları doldurması gerekecektir.

**Çalıştığı ülke Türkiye ise il ve ilçe kayıt kısımları ekranda görüntülenecektir.**

1. Çalıştığı İl
2. Çalıştığınız İlçe
3. İş yeri Adı
4. İş Telefonu
5. Dahili
6. İş Alanı
7. Departmanı
8. Pozisyonu
9. İş Yeri Açık Adresi

Çalıştığı ülkenin Türkiye olması durumunda doldurması gereken formun ekran görüntüsü Şekil 5,6'da verilmiştir.



C-İŞ BİLGİLERİ(Detayları Gizle...)

**Herhangi Bir İşyerinde Çalışıyormusunuz(\*)**

Evet Çalışıyorum

Hayır Çalışmıyorum

**Çalıştığı Ülke(\*) :**

Türkiye

**Çalıştığı İl(\*):**

AFYON

**Çalıştığı İlçe(\*):**

MERKEZ

**İş Yeri Adı:**

Afyon Kocatepe Üniversitesi

**İş Telefonu :**

Örn(272) 254-61xx  
(272) 444-0303

**Dahili:**

219

**İş Alanı**

Akademik

**Departman**

Bilgi işlem

**Pozisyonu(\*):**

Analist Programcı

**Açık Adres(\*):**

Gazlıgöl yolu  
10. KM Afyonkarahisar

Şekil 5. 6 Aday çalışıyorsa ve ülke Türkiye ise doldurması gereken bilgiler.

**Çalıştığı Ülke Türkiye Değilse aşağıdaki bilgiler görüntülenecektir.**

1. İş yeri adı
2. İş Telefonu
3. Dahili
4. İş Alanı
5. Departmanı
6. Pozisyonu
7. İş Yeri Açık Adresi



Çalıştığı ülkenin Türkiye olmaması durumunda doldurması gereken formun ekran görüntüsü Şekil 5.7’de verilmiştir.



C-İŞ BİLGİLERİ(Detayları Gizle...)

**Herhangi Bir İşyerinde Çalışıyormusunuz(\*)**

Evet Çalışıyorum

Hayır Çalışmıyorum

**Çalıştığı Ülke(\*) :** Avusturya

**İş Yeri Adı:** FH-JOANNEUM UNIVERSITY

**İş Telefonu :** Örñ(272) 254-61xx  
(484) 107-09\*\*

**Dahili:** 219

**İş Alanı** Akademik

**Depertman** Bilgi işlem

**Pozisyonu(\*):** Analist Programcı

**Açık Adres(\*):** Krottendorf 6, Kapfenberg  
Austria.

Şekil 5. 7 Aday çalışıyorsa ve ülke Türkiye değilse doldurması gereken bilgiler.

#### 5.1.1.4 Eğitim Bilgileri Ekranı

Eğitim bilgileri kısmında yüksek lisansa kayıt yaptırmak isteyen adayların aşağıdaki bilgileri doldurması gerekmektedir.

1. Mezun olduğu üniversitenin bulunduğu ülke

**Seçilen ülke Türkiye ise bulunduğu il, üniversite, fakülte, program/bölüm kısımları seçilebilir olacaktır.**

2. Mezun olduğu üniversitenin bulunduğu il
3. Mezun olduğu üniversite
4. Mezun olduğu Fakülte
5. Mezun olduğu program/bölüm
6. Mezuniyet tarihi
7. Mezuniyet derecesi
8. Diğer eğitimler

Başvuruyu yapan lisans mezunu adayın, bitirdiği üniversite Türkiye’de ise Şekil 5.8’de, örnek ekran görüntüsü verilen bilgileri doldurması gerekmektedir.



**D-EĞİTİM BİLGİLERİ(ÜNİVERSİTE)(Detayları Gizle...)**

<b>Bulunduğu Ülke :</b>	<input type="text" value="Türkiye"/>
<b>Bulunduğu İl :</b>	<input type="text" value="Ankara"/>
<b>Üniversite :</b>	<input type="text" value="Ankara Üniversitesi"/>
<b>Mezun Olduğu Fakülte / Yüksekokul</b>	<input type="text" value="Eğitim Fakültesi"/>
<b>Mezun Olduğu Program / Bölüm :</b>	<input type="text" value="Bilgisayar Öğretmenliği"/>
<b>Mezuniyet Tarihi</b>	<input type="text" value="05.09.2008"/>
<b>Mezuniyet Derecesi / Notu :</b>	<input type="text" value="90,000"/>
<b>Diğer Eğitimler :</b>	<input style="height: 40px;" type="text" value="Yüksek lisans Afyon Kocatepe Üniversitesi"/>

**Şekil 5. 8** Ülke Türkiye ise eğitim bilgileri ekranındaki doldurulması gereken alanlar.

**Seçilen ülke Türkiye değilse bulunduğu üniversite, fakülte, program/bölüm kısımları text içerisinde doldurulabilir formatta olacaktır.**

1. Okuduğu üniversitenin Bulunduğu İl
2. Okuduğu üniversite
3. Mezun Olduğu Fakülte
4. Mezun Olduğu Program/Bölüm
5. Mezuniyet Tarihi
6. Mezuniyet Derecesi
7. Diğer eğitimler

Başvuruyu yapan lisans mezunu adayın, bitirdiği üniversite Türkiye’de değil ise Şekil 5.9’da, örnek ekran görüntüsü verilen bilgileri doldurması gerekmektedir.



The screenshot shows a form titled "D-EĞİTİM BİLGİLERİ(ÜNİVERSİTE)(Detayları Gizle...)" with a navigation bar at the top containing six numbered circles (1-6). The form fields are as follows:

<b>Bulunduğu Ülke :</b>	Avusturya
<b>Mezun Olduğunuz Üniversite</b>	FH-Joanneum
<b>Mezun Olduğu Fakülte / Yüksekokul</b>	İnternet Teknolojileri
<b>Mezun Olduğu Program / Bölüm :</b>	Grafik Tasarımı
<b>Mezuniyet Tarihi</b>	09.07.2009
<b>Mezuniyet Derecesi / Notu :</b>	85,000
<b>Diğer Eğitimler :</b>	Micrasoft Uygulama Geliştirme Sertifika Eğitimi (MSCD:Microsoft Certified Solution Developer)

**Şekil 5. 9** Eğitim bilgileri ekranı, seçilen ülke Türkiye değilse doldurulacak alanlar.

- Kişisel bilgiler ekranında yer alan askerlik şubeleri,
- İletişim bilgileri ekranında yer alan ülke il ve ilçe seçenekleri,

- İş bilgileri ekranında yer alan ülke il ve ilçe seçenekleri,
- Eğitim bilgileri ekranında yer alan ülke, il, üniversite, fakülte ve bölüm seçenekleri

Bu seçenekler, SQL veri tablolarından çekildiği takdirde sistemi çok yavaşlatmaktadır. Sistemin performansını düşürmemek için en uygun yöntem bu verileri Ajax tekniğini kullanarak XML'den çekmektir. Bu işlemi yapabilmek için öncelikle üniversite, fakülte, bölüm, askerlik şubesi gibi bilgiler Excel'e aktarılmıştır. Ayrıca üniversite bilgilerinin ÖSYM tarafından Excel şeklinde gönderilmesi nedeniyle, bu Excel tablolarını ve diğer bilgileri XML tablosuna dönüştüren ve veritabanına yazan bir yazılım yapılmıştır. Bu işlemin tek tek yapılması belki de haftalar almaktadır. Ayrıca çok sayıda üniversite kodu, fakülte kodu veya bölüm kodunu sisteme hatasız girmek neredeyse imkânsızdır. Şekil 5.10'da bu işlemlerin yapıldığı sayfanın örnek ekran görüntüsü verilmiştir.



Verileri XML e Dönüştür



Üniversiteleri DataBase Aktar



Fakülteleri DataBase Aktar



Programları DataBase Aktar

**Şekil 5. 10** Excel dönüşümleri sayfası.

### 5.1.1.5 Sınav Bilgileri Ekranı

Sınav bilgileri kısmında 9 adet doldurulabilir kayıt alanı yer almaktadır.

1. ALES Eşit Ağırlık

2. ALES Sayısal
3. ALES Sözel
4. YDGYS
5. ÜDS
6. KPDS
7. TOEFL
8. ILTS
9. ÜDYS

Bu kısımlardan ALES Eşit Ağırlık, ALES Sayısal, ALES Sözel ile birlikte en az bir dil sınav notunun girilmesi gereklidir. Aksi takdirde sistem kabul etmemektedir. Şekil 5.11’de sınav bilgileri ekranının örnek çıktısı verilmektedir.



E-SINAV BİLGİLERİ(Detayları Gizle...)	
<b>Not LES PUANLARININ TAMAMI VE EN AZ BİR DİL PUANI GİRİLMELİ</b>	
<b>ALES(EA) :</b>	87,980
<b>ALES(SAY) :</b>	89,680
<b>ALES (SÖZ) :</b>	84,610
<b>YDGYS :</b>	
<b>ÜDS:</b>	91,000
<b>KPDS</b>	
<b>TOEFL</b>	
<b>ILTS</b>	
<b>ÜDYS</b>	

Şekil 5. 11 Sınav bilgileri ekranı.

### 5.1.1.6 Başvurulacak Program Seçimi

Başvurulacak program seçimi kayıt bölümünde ise aday yüksek lisans için hangi programa kayıt yaptırmak istediğini seçmesi gereklidir. Bu bölümde seçilmesi gereken alanlar;

1. Enstitü Seçimi,
2. Bölüm Seçimi,
3. Program Seçimidir.

Başvuru sayfasının ekran görüntüsü Şekil 5.12’de verilmiştir.



F-BAŞVURACAĞINIZ PROGRAMI SEÇİNİZ(Detayları Gizle...)

**Enstitü Seçiniz**

Fen Bilimleri Enstitüsü

**Bölüm Seçiniz**

Fizik Anabilim Dalı

**Tercih (\*) :**

Tezsiz Fizik Yüksek Lisans



İLERİ



İPTAL

Şekil 5. 12 Başvurulacak programın seçimi.

Bu alanlar seçildikten sonra adayın doldurmuş olduğu kayıt bilgileri rapor edilmektedir. Şayet herhangi bir yanlışlık yok ise aday kabul ediyorum butonuna tıklayarak kayıt

işlemini bitirebilir. Başvuru yapıldıktan sonra başvuru için doldurulan bilgiler Şekil 5.13'deki gibi rapor halinde verilmektedir.

☑ A - KİŞİSEL BİLGİLERİNİZ (Detayları Gizle...)



<b>Ad (*) :</b>	Emin
<b>Soyad(*) :</b>	İbili
<b>Uyruğu(*) :</b>	Türkiye
<b>Nüfusa Kayıtlı Olduğu İl(*) :</b>	AFYON
<b>Nüfusa Kayıtlı Olduğu İlçe (*) :</b>	MERKEZ
<b>Vatandaşlık No(*) :</b>	12486644343
<b>Doğum Tarihi :</b>	24.03.1982
<b>Cinsiyet(*) :</b>	Erkek
<b>Kayıtlı Olduğu Askerlik Şubesi :</b>	MERKEZ ASKERLİK ŞUBESİ
<b>Askerlik Durumu :</b>	Erteledim
<b>Erteleme Bitis Tarihi</b>	16.11.2009 00:00:00



KABUL  
EDİYORUM



İPTAL

Şekil 5. 13 Rapor sayfası.






## 5.1.2 Başvurusu Onaylanan Öğrencilere E-posta Gönderimi

Öğrenci başvuru işlemleri sona erdikten sonra, her bölümün sekreteri, programlara başvuru yapan öğrencilere mülakat ile ilgili e-posta gönderir. E-posta öğrencinin kayıt esnasında belirtmiş olduğu her iki e-posta adresine de ulaştırılmaktadır. E-posta gönderim esnasında e-posta kimden geldiğini, konusunu, içerik tipini ve mesaj içeriğini yazması gerekmektedir. Göndereceği kişileri ise listeden seçebilir. E-posta gönderme işleminin yapıldığı sayfanın örnek ekran çıktısı Şekil 5.15’de verilmektedir. Bölüm sekreteri uzaktan öğrenim yönetim sistemimizde en aktif olan ve an çok görevi olan kullanıcıdır. Şekil 5.14’de bölüm sekreterinin sahip olduğu görevler gösterilmektedir.

 Danisman Ekle	 Danisman Ders Kayıt	 Ders Kayıt Onay	 Öğretim Elemanı Ders Listesi	 Evrak Kontrol	 Mail Gönder	 Program İşlemleri Dönem İşlemleri Ders İşlemleri	 Öğrenci Başvuru İşlemleri Danisman İşlemleri	 Şube İşlemleri Ders Programı	 Sınav İşlemleri Ödev İşlemleri Kaynakça Tanımlamaları Konu Tanımlamaları
--	--	--	---	--	--	--	--	--	--

Şekil 5. 14 Bölüm sekreteri giriş ekranı.

Sıra No	Adi	Soyadi	Email	Email2	Seç
☆ 	1	Emin Ibili	emin_ibili@hotmail.com	eminnet03@yahoo.com	<input checked="" type="checkbox"/>
☆ 	3	Fatih Bayram	fatihbayr12@hotmail.com	fatihbayr122@hotmail.com	<input checked="" type="checkbox"/>
☆ 	4	Fidan Hakkari	fidan123@hotmail.com	fidan1234@hotmail.com	<input checked="" type="checkbox"/>

Kimden: Fen Bilimleri Enstitüsü Bilgisayar Anabilim Dalı


Konu: Yüksek Lisans Başvurusu

İçerik Tipi: HTML ▾

Mesaj:

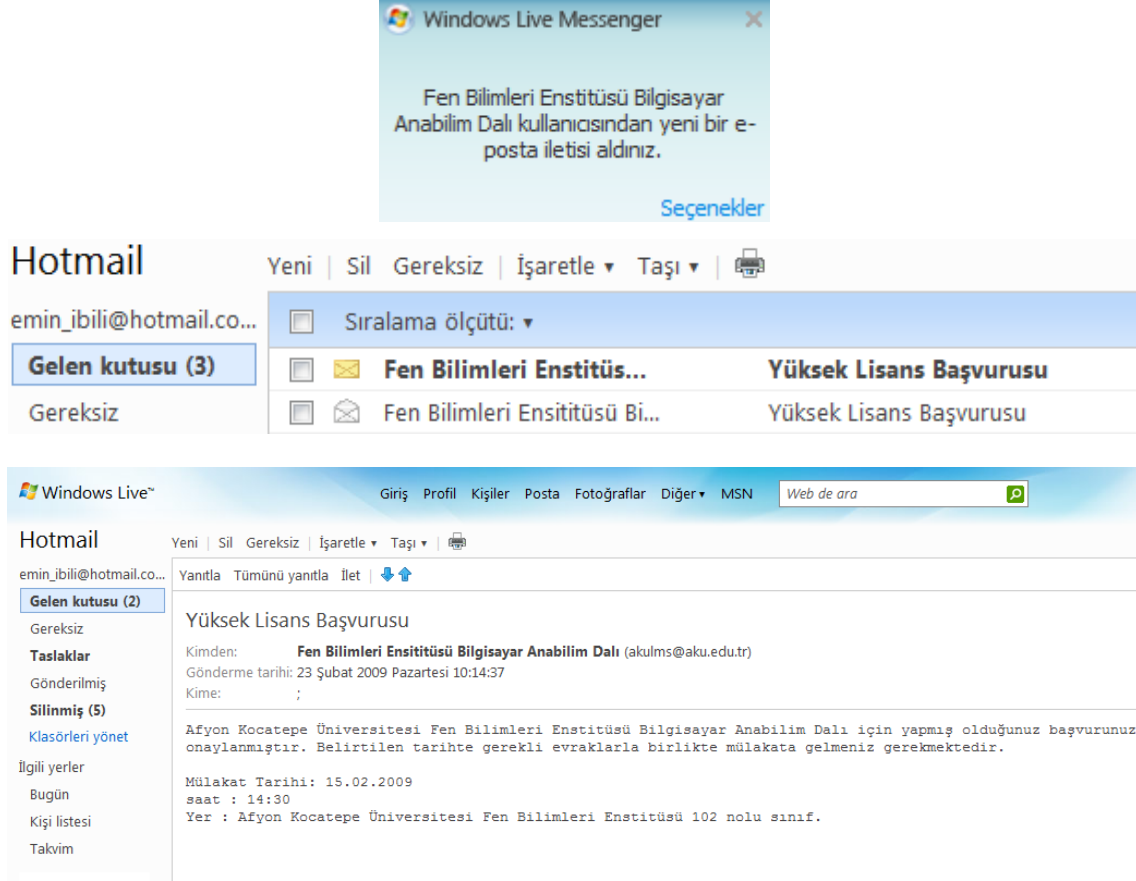
Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Anabilim Dalı için yapmış olduğunuz başvurunuz onaylanmıştır. Belirtilen tarihte gerekli evraklarla birlikte mülakata gelmeniz gerekmektedir.

Mülakat Tarihi: 15.02.2009  
Saat : 13:30  
Yer : Afyon Kocatepe Üniversitesi Ahmet Necdet Sezer Kampüsü Fen Bilimleri Enstitüsü 102 nolu sınıf

  
Gonder

Şekil 5. 15 Mülakat bilgileri E-posta sayfası.

Gönderilen bu e-posta Şekil 5.16'da da gösterildiği gibi kullanıcının yazmış olduğu e-posta adresine ulaşmaktadır.



Şekil 5. 16 Gönderilen mülakat bilgilerinin E-posta adresine ulaşması.

### 5.1.3 Bölüm Sekreteri Evrak Onay

Bölüm sekreteri daha önce mülakatla ilgili öğrencilere e-posta göndermişti. Mülakat tarihinde ise, aday mülakata girmeden önce evrakları kontrol edilmektedir. Şayet evraklarda her hangi bir eksiklik yada yanlışlık yoksa adayın mülakata girebilmesi için onay verilir. Şekil 5.17’de evrak onay sayfasının görüntüsü verilmektedir.

	Sıra No	Bolum No	Ogrenci Adı	Ogrenci Soyadı	Ales Say	Ales Uds	Seç
<a href="#">Detay Göster</a>	1		Emin	Ibili	85	90	<input checked="" type="checkbox"/>
<a href="#">Detay Göster</a>	3		Fatih	Bayram	90	90	<input checked="" type="checkbox"/>
<a href="#">Detay Göster</a>	4		Fidan	Hakkari	90	90	<input checked="" type="checkbox"/>

[Seçilenleri Onayla](#)

**Şekil 5. 17** Evrak onay sayfası.

#### 5.1.4 Bölüm Başkanı, Mülakat Notu Girişi

Evrakları onaylanan öğrenciler, bölüm başkanı'nın ekranında görüntülenmektedir. Bölüm başkanı adayların mülakattan aldığı notları sisteme girmektedir. İsterse adayların detaylı bilgilerini görüntüleyebilmektedir. Şekil 5.18'de mülakat notu girişinin yapıldığı sayfanın ekran görüntüsü verilmektedir.

İlk Öğretim Fizik Öğretmenliği ▾

Onaylanan Başvuruları Listele
  Onaylanmayan Başvuruları Listele

Sıra No	Vatandaslik No	Ylot Adı	Ylot Soyadı	Asil Seç	
1	12121212121	Emin	Ibili	<input type="text" value="90"/>	<a href="#">Select</a>
3	12312121231	Fatih	Bayram	<input type="text" value="90"/>	<a href="#">Select</a>
4	12121212122	Fidan	Hakkari	<input type="text" value="90"/>	<a href="#">Select</a>

[Seçilenleri Onayla](#)

**Şekil 5. 18** Mülakat notu girişi sayfası.

### 5.1.5 Öğrenci İşleri Onay

Bölüm başkanının, mülakat sonuçlarını sisteme eklemesinden sonra, öğrenci işleri, kazanan öğrencileri ilan etmektedir. Ayrıca öğrencinin interaktif eğitim için kullanabileceği şifre ve kullanıcı adı, öğrenciye e-posta yoluyla gönderilmektedir. Onaylanan öğrenci, Student rolü ile sisteme kaydolur ve öğrenci, yetkisi dâhilindeki sayfalar arasında geçiş yapabilir. Şekil 5.19’da öğrenci işleri onay sayfası verilmiştir.

	Oğrenci Sıra No	Ylot Adı	Ylot Soyadı	Email	Vatandaşlık No	Asil Seç
<a href="#">Select</a>	1	Emin	Ibili	emin_ibili@hotmail.com	12121212121	90 <input checked="" type="checkbox"/>
<a href="#">Select</a>	3	Fatih	Bayram	fatihbayr12@hotmail.com	12312121231	90 <input checked="" type="checkbox"/>
<a href="#">Select</a>	4	Fidan	Hakkari	fidan123@hotmail.com	12121212122	90 <input checked="" type="checkbox"/>

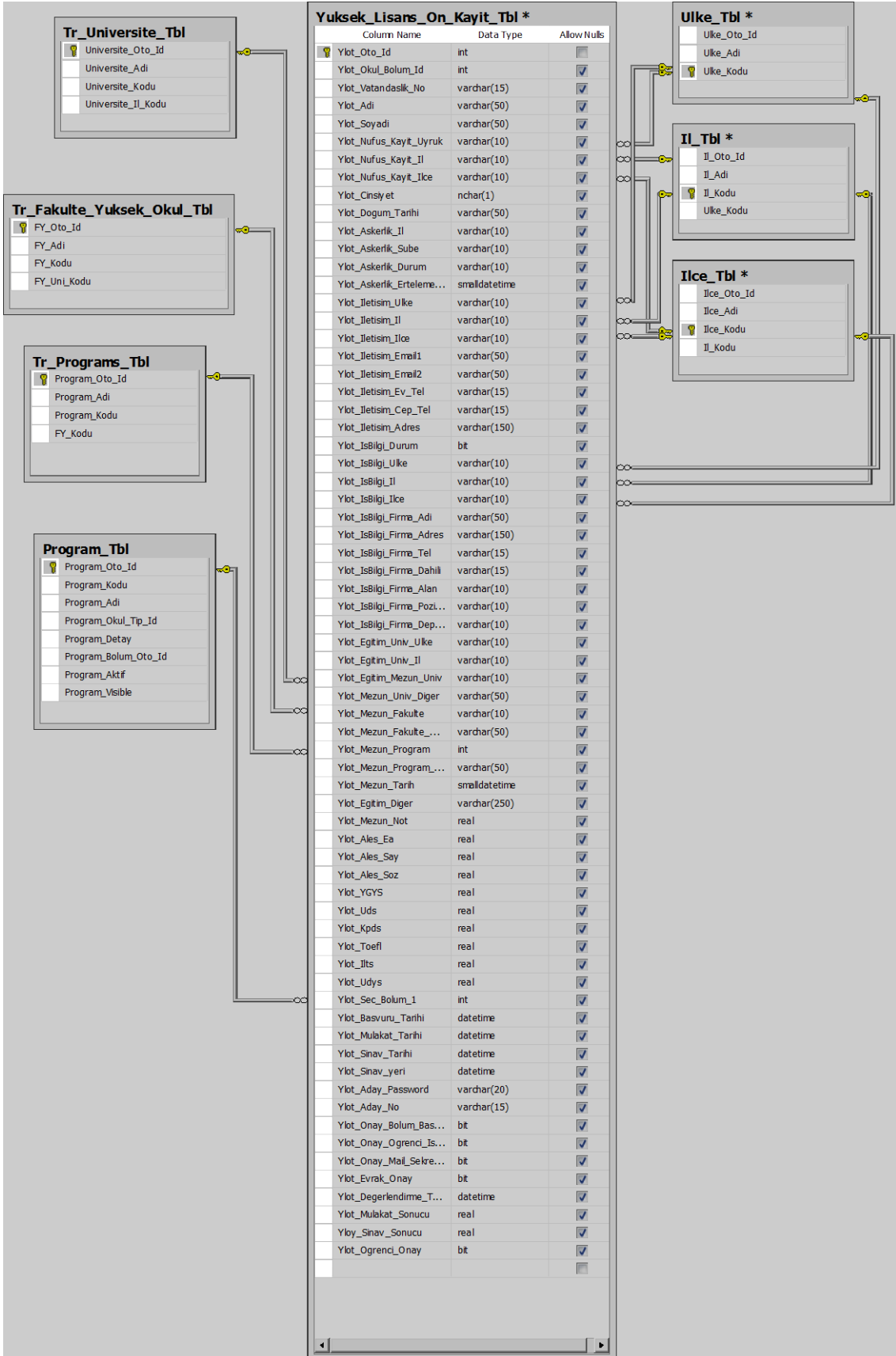
Kazanan Öğrencileri Listele  Diğer Öğrencileri Listele

[Seçilenleri Onayla](#)

Şekil 5. 19 Öğrenci işleri onay sayfası.

Veri tabanında online kayıt modülü ile ilgili Çizelge 5.2’deki ilişki mevcuttur.

Çizelge 5. 2 Online kayıt modülündeki bilgilerin veri tabanını ilişkisi.



Çizelge 5,2'deki tablolardan da anlaşılacağı üzere yüksek lisansa ön kayıt için başvuru yapan öğrencilerin kişisel bilgileri Yuksek\_Lisans\_On\_Kayit\_Tbl tablosunda tutulmaktadır. Bu tablodaki alanların açıklaması aşağıda verilmiştir.

- **Ylot\_Oto\_Id:** Her öğrencinin kaydında otomatik artan sıralama ID'sidir.
- **Ylot\_Okul\_Bolum\_Id:** Öğrencinin, başvuru yaptığı programın kayıtlı olduğu bölümün kodudur. Program\_Tbl deki Program\_Bolum\_Oto\_Id ile ilişkilidir.
- **Ylot\_Vatandaslik\_No:** Öğrencinin vatandaşlık numarasıdır.
- **Ylot\_Adi:** Öğrencinin adıdır.
- **Ylot\_Soyadi:** Öğrencinin soyadıdır.
- **Ylot\_Nufus\_Kayit\_Uyruk:** Öğrencinin vatandaşı olduğu ülkenin kodudur (Örn. Türkiye için 90). Bu alan Ulke\_Tbl deki Ulke\_Kodu ile ilişkilidir.
- **Ylot\_Nufus\_Kayit\_Il:** Nüfusa kayıtlı il kodudur (Örn. Afyonkarahisar için 03). IL\_Tbl deki Il\_Kodu ile ilişkilidir.
- **Ylot\_Nufus\_Kayit\_Ilce:** Nüfusa kayıtlı olduğu ilçe kodudur (Örn. İncehisar ilçesi 13). Ilce\_Tbl deki Ilce\_Kodu ile ilişkilidir.
- **Ylot\_Cinsiyet:** Öğrencinin cinsiyetidir. 2 değer alır. Erkek adaylar için "1" Bayan adaylar için "0" değerini alır.
- **Ylot\_Dogum\_Tarihi:** Doğum tarihidir.
- **Ylot\_Askerlik\_Il:** Türk vatandaşı olan erkek adaylar için kayıtlı olduğu askerlik şubesinin bulunduğu il kodudur. IL\_Tbl deki Il\_Kodu ile ilişkilidir.
- **Ylot\_Askerlik\_Sube:** Türk vatandaşı olan erkek adaylar için kayıtlı olduğu askerlik şubesinin kodudur. Ilce\_Tbl deki Ilce\_Kodu ile ilişkilidir. Her ilçede bir askerlik şubesi olduğundan bu alanla ilişkilendirilmiştir.
- **Ylot\_Askerlik\_Durum:** : Türk vatandaşı olan erkek adaylar için askerlik durumunu belirten alandır.
- **Ylot\_Askerlik\_Erteleme\_Tarih:** Türk vatandaşı olan erkek adayın askerlik durumu her hangi bir tarihe kadar ertelenmiş ise erteleme bitiş tarihi bilgisi tutulur.
- **Ylot\_Iletisim\_Ulke:** Yaşadığı ülkenin kodudur. Bu alan Ulke\_Tbl deki Ulke\_Kodu ile ilişkilidir.
- **Ylot\_Iletisim\_Il:** Yaşadığı il kodudur. IL\_Tbl deki Il\_Kodu ile ilişkilidir.

- **Ylot\_Iletisim\_Ilce:** Yaşadığı ilçe kodudur. Ilce\_Tbl deki Ilce\_Kodu ile ilişkilidir.
- **Ylot\_Iletisim\_Email1:** Ulaşılabilir 1.E-posta adresidir.
- **Ylot\_Iletisim\_Email2:** Ulaşılabilir 2. E-posta adresidir.
- **Ylot\_Iletisim\_Ev\_Tel:** Ev telefonu numarasıdır.
- **Ylot\_Iletisim\_Cep\_Tel:** Cep telefonu numarasıdır.
- **Ylot\_Iletisim\_Adres:** İletişim kurulacak açık adrestir.
- **Ylot\_IsBilgi\_Durum:** Her hangi bir işte çalışıp çalışmadığı durumunu belirten alandır. Çalışıyorsa “1” çalışmıyorsa “0” değerini alır.
- **Ylot\_IsBilgi\_Ulke:** Çalışıyorsa, çalıştığı ülkenin kodudur. Ulke\_Tbl deki Ulke\_Kodu ile ilişkilidir.
- **Ylot\_IsBilgi\_Il:** Çalıştığı ülke Türkiye ise çalıştığı il kodudur. Il\_Tbl deki Il\_Kodu ile ilişkilidir.
- **Ylot\_IsBilgi\_Ilce:** Çalıştığı ülke Türkiye ise ilçe kodudur. Ilce\_Tbl deki Ilce\_Kodu ile ilişkilidir.
- **Ylot\_IsBilgi\_Firma\_Adi:** Çalıştığı firmanın ismidir.
- **Ylot\_IsBilgi\_Firma\_Adres:** Çalıştığı firmanın açık adresidir.
- **Ylot\_IsBilgi\_Firma\_Tel:** İş yerinin telefon numarasıdır.
- **Ylot\_IsBilgi\_Firma\_Dahili:** Çalıştığı firmanın dahili numarasıdır.
- **Ylot\_IsBilgi\_Firma\_Alan:** Çalıştığı sektörün kodudur.
- **Ylot\_IsBilgi\_Firma\_Pozisyon:** Çalıştığı sektördeki pozisyonudur.
- **Ylot\_IsBilgi\_Firma\_Depertman:** Çalıştığı departmandır.
- **Ylot\_Egitim\_Univ\_Ulke:** Mezun olduğu üniversitenin bulunduğu ülke kodudur. Ulke\_Tbl deki Ulke\_Kodu ile ilişkilidir.
- **Ylot\_Egitim\_Univ\_Il:** Mezun olduğu üniversitenin bulunduğu il kodudur. Il\_Tbl deki Il\_Kodu ile ilişkilidir.
- **Ylot\_Egitim\_Mezun\_Univ:** Mezun olduğu üniversitenin kodudur (lisans). Tr\_Universite\_Tbl'deki Universite\_Kodu ile ilişkilidir.
- **Ylot\_Mezun\_Univ\_Diger:** Şayet mezun olduğu üniversite Türkiye’de değilse, üniversite isminin tutulduğu tekst alanıdır.



- **Ylot\_Mezun\_Fakulte:** Mezun olduđu üniversite Türkiyede ise mezun olduđu Fakültenin kodu. Tr\_Fakulte\_Yukse\_Okul\_Tbl alanındaki Fy\_Kodu ile ilişkilidir.
- **Ylot\_Mezun\_Fakulte\_Diger:** Mezun olduđu üniversite Türkiye’de değil ise Mezun olduđu fakültenin isminin tutulduđu tekst alanıdır.
- **Ylot\_Mezun\_Program:** Mezun olduđu üniversite Türkiye’de ise mezun olduđu programın kodudur. Tr\_Programs\_Tbl’deki Program\_Kodu ile ilişkilidir.
- **Ylot\_Mezun\_Program\_Diger:** Mezun olduđu üniversite Türkiyede değil ise mezun olduđu programın isminin tutulduđu text alanıdır.
- **Ylot\_Mezun\_Tarih:** Üniversiteden mezun olduđu tarihtir.
- **Ylot\_Egitim\_Diger:** Almış olduđu diğer eğitimlerdir.
- **Ylot\_Mezun\_Not:** Mezuniyet ortalamasıdır.
- **Ylot\_Ales\_Ea:** ALES sınavından aldığı eşit ağırlık puanıdır.
- **Ylot\_Ales\_Say:** ALES sınavından aldığı sayısal puanıdır.
- **Ylot\_Ales\_Soz:** ALES sınavından aldığı Sözel puanıdır.
- **Ylot\_YGYS:** YGYS puanıdır.
- **Ylot\_Uds:** UDS puanıdır.
- **Ylot\_Kpds:** KPDS puanıdır.
- **Ylot\_Toefl:** TOEFL puanıdır.
- **Ylot\_Ilts:** ILTS puanıdır.
- **Ylot\_Udys:** UDYS puanıdır.
- **Ylot\_Sec\_Bolum\_1:** Başvurmak istediği yüksek lisans programının kodudur. Program\_Tbl’deki Program\_Kodu ile ilişkilidir.
- **Ylot\_Basvuru\_Tarihi:** Başvuru yapıldığı tarihtir.
- **Ylot\_Mulakat\_Tarihi:** Mülakat tarihtir.
- **Ylot\_Sinav\_Tarihi:** Mülakat sınavının yapıldığı tarihtir.
- **Ylot\_Sinav\_yeri:** Mülakat sınavını yapıldığı yerdir.
- **Ylot\_Aday\_Password:** İlk aday şifresidir. Başvuru durumunu kontrol etmesi için gereklidir.
- **Ylot\_Aday\_No:** İlk aday numarasıdır.

- **Ylot\_Onay\_Bolum\_Baskani:** Bölüm başkanının onay verip vermediği bilgisidir. "1" ve "0" değeri alır.
- **Ylot\_Onay\_Ogrenci\_Isleri:** Öğrenci işlerinin onay verip vermediği bilgisidir. "1" ve "0" değeri alır.
- **Ylot\_Onay\_Mail\_Sekreter:** Bölüm sekreterinin mülakat için e-posta gönderip göndermediği bilgisidir. Mail gönderilmişse "1" değerini alır. E-posta gönderilmemişse yada gönderme işleminde hata varsa "0" değerini alır.
- **Ylot\_Evrak\_Onay:** Mülakat öncesinde evrak kontrolünde eksik evrak bulunup bulunmadığı bilgisidir. Eksik evrak varsa "0" değerini alır ve aday mülakata giremez. Evrak tam ise "1" değerini alır ve aday mülakata girebilir.
- **Ylot\_Degerlendirme\_Tarihi:** Mülakat notunun verildiği tarihtir.
- **Ylot\_Mulakat\_Sonucu:** Mülakat puanıdır.
- **Yloy\_Sinav\_Sonucu:** ALES notunun %60'ı, dil puanın %15'i, Mezuniyet ortalamasının %15'i, mülakat notunun %10' udur.
- **Ylot\_Ogrenci\_Onay:** Öğrenci başvuru işleminin başarılı olup olmadığıdır. Başarılı ise "1" başarısız ise "0" değerini alır.

## 5.2 Kullanıcı ve Kullanıcı Gurupları Modülü

Bu modül ile personel hareketliliği sağlanmaktadır. Sisteme yeni eklenecek öğretim üyesi bu bölümden eklenmektedir. Bölümle ilgili işlem yapma yetkisi yine personel daire başkanlığı rolüne sahip kullanıcılarda bulunmaktadır. Personel daire başkanlığında üç adet yetki bulunmaktadır.

- 1) **Unvan Tanımlama:** Bu konu ünvan tanımlama modülünde anlatılacaktır.
- 2) **Personel Girişi:** Akademik ve idari personel girişi yapılmaktadır.
- 3) **Öğretim Üyesi Bölüm:** Sisteme kaydedilen öğretim üyesi görev yapacağı bölüme atanmaktadır.

### 5.2.1 Personel Girişi

Uzaktan eğitim yönetim sisteminde ders vermekle görevli olan öğretim üyeleri personel daire başkanlığının yetkisinde sisteme eklenmektedir. Personel ekleme işlemi 2 aşamada gerçekleşmektedir.

1. Kişisel bilgiler
2. İletişim bilgileri

#### 5.2.1.1 Kişisel Bilgiler Ekranı

Kişisel bilgiler ekranında aşağıdaki bilgilerin girilmesi gerekmektedir.

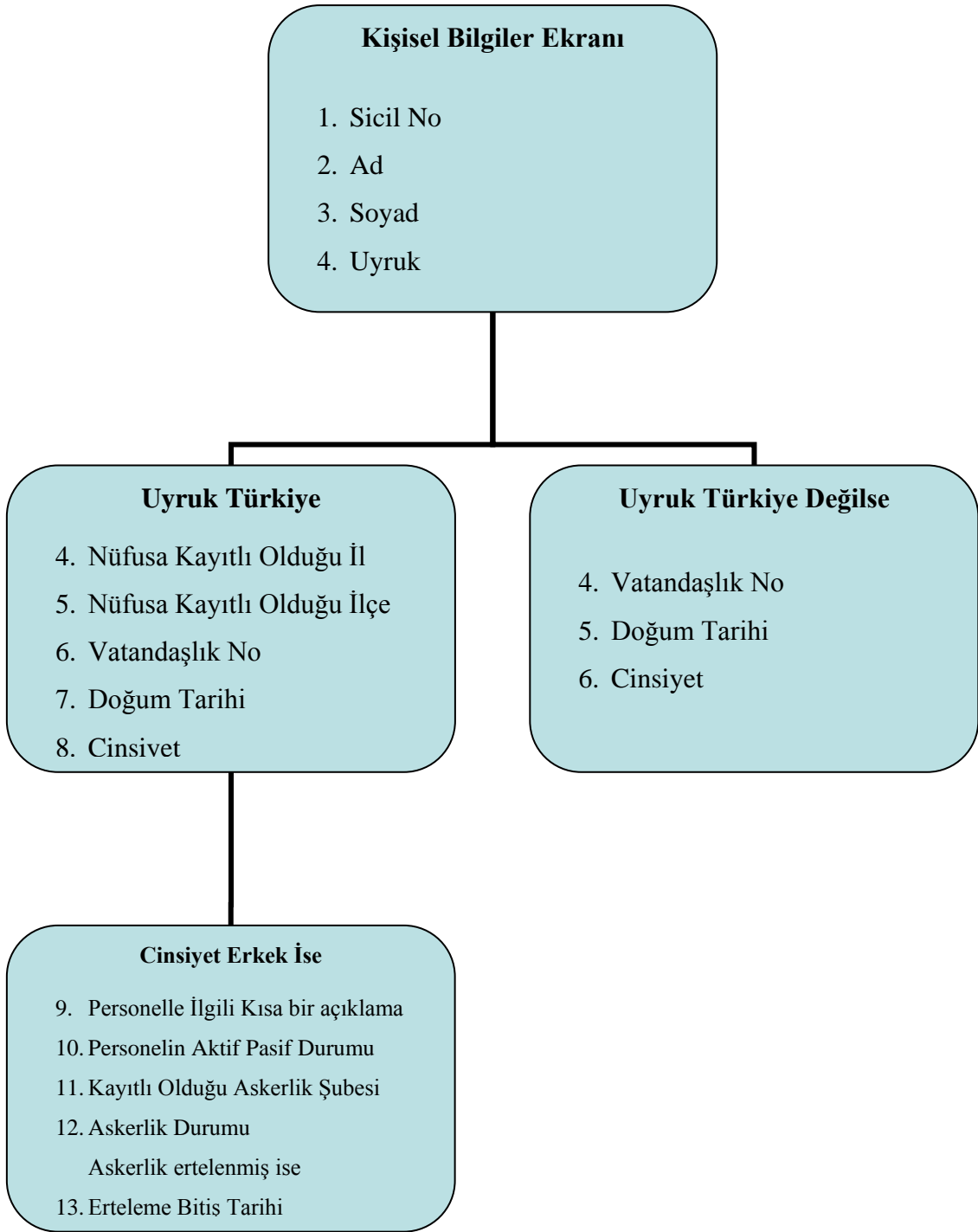
1. Sicil No
2. Ad
3. Soyad
4. Uyruk

Şayet uyruk Türkiye ise kullanıcıdan nüfusa kayıtlı olduğu il, nüfusa kayıtlı olduğu ilçe bilgileri de istenmektedir.

4. Nüfusa Kayıtlı Olduğu İl
5. Nüfusa Kayıtlı Olduğu İlçe
6. Vatandaşlık No
7. Doğum Tarihi
8. Cinsiyet
9. Kayıtlı Olduğu Askerlik Şubesi
10. Askerlik Durumu
11. Personelle İlgili Kısa bir açıklama
12. Personelin Aktif Pasif Durumu

Kişi askerliğini ertelemişse ne zamana kadar ertelediği belirtilmelidir. Kişisel bilgiler ekranını basite indirgemek gerekirse Çizelge 5.3'deki akış diyagramının takip edilmesi gerekmektedir.

**Çizelge 5. 3** Personel kayıt işlemleri Kişisel Bilgiler diyagramı.









### 5.2.1.2 İletişim Bilgileri

İletişim bilgileri alanında aşağıdaki bilgilerin girilmesi gerekmektedir.


- 1) İlk E-posta adresi
- 2) İkinci E-posta Adresi
- 3) Web Sitesi
- 4) Ev Tel
- 5) Cep Tel
- 6) Dahili
- 7) Açık Adres
- 8) Posta Kodu


İleri düğmesi tıklandıktan sonra girmiş olduğu bilgiler raporlanır ve yetkili kişiden bu bilgileri onaylaması istenmektedir. Onaylanması durumunda personel Lecture adlı role atanır ve e-mail adresine online işlemler için kullanacağı kullanıcı adı ve şifresi gönderilir. Eklenen personelin Türk vatandaşı olması, erkek olması ve askerliği herhangi bir tarihe kadar erteleme durumunda Şekil 5.20' deki örnek personel ekleme işlemi verilmiştir.


  																																															
Kullanıcı Giriş Sayfası      Çıkış      Mesajlaş																																															
 <p>UNVAN TANIMLAMA</p>  <p>PERSONEL GİRİŞİ</p>  <p>ÖĞRETİM ÜYESİ BÖLÜM</p>	<div style="background-color: #4a69bd; color: white; padding: 5px; text-align: center;">       A - KİŞİSEL BİLGİLERİNİZ (Detayları Gizle...)     </div> <div style="text-align: center; margin-top: 10px;"> <b>YENİ PERSONEL</b> </div> <div style="background-color: #4a69bd; color: white; padding: 5px; text-align: center; margin-top: 10px;">       A - KİŞİSEL BİLGİLERİNİZ (Detayları Gizle...)     </div> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;"><b>SicilNo</b></td> <td><input type="text" value="A_123456"/></td> </tr> <tr> <td><b>Ad (*) :</b></td> <td><input type="text" value="Mevlüt"/></td> </tr> <tr> <td><b>Soyad(*) :</b></td> <td><input type="text" value="Doğan"/></td> </tr> <tr> <td><b>Uyruğu(*) :</b></td> <td><input type="text" value="Türkiye"/></td> </tr> <tr> <td><b>Nüfusa Kayıtlı Olduğu İl(*) :</b></td> <td><input type="text" value="AFYON"/></td> </tr> <tr> <td><b>Nüfusa Kayıtlı Olduğu İlçe (*) :</b></td> <td><input type="text" value="MERKEZ"/></td> </tr> <tr> <td><b>Vatandaşlık No(*) :</b></td> <td><input type="text" value="22123344342"/></td> </tr> <tr> <td><b>Doğum Tarihi</b></td> <td><input type="text" value="03.03.1968"/></td> </tr> <tr> <td><b>Ünvanı</b></td> <td><input type="text" value="Prof.Dr"/></td> </tr> <tr> <td><b>Cinsiyet(*) :</b></td> <td> <input checked="" type="radio"/> Erkek    <input type="radio"/> Bayan         </td> </tr> <tr> <td><b>Kayıtlı Olduğu Askerlik şubesi :</b></td> <td><input type="text" value="AFYON"/></td> </tr> <tr> <td><b>Askerlik Durumu:</b></td> <td> <input type="text" value="MERKEZ ASKERLIK ŞUBESİ"/>   <input checked="" type="radio"/> Yaptım    <input type="radio"/> Ertele dim    <input type="radio"/> Muafım         </td> </tr> <tr> <td><b>Açıklama</b></td> <td><input type="text" value="Fizik Bölümü Öğretim Üyesi"/></td> </tr> <tr> <td><b>Aktif / Pasif</b></td> <td> <input type="radio"/> Pasif    <input checked="" type="radio"/> Aktif         </td> </tr> </table> <div style="background-color: #4a69bd; color: white; padding: 5px; text-align: center; margin-top: 10px;">       B-İLETİŞİM BİLGİLERİNİZ (Detayları Gizle...)     </div> <p>Örnek Telefon Numarası : (272) 228-1213</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;"><b>E-Posta1(*) :</b></td> <td><input type="text" value="mevlutdogan@hotmail.com"/></td> </tr> <tr> <td><b>E-Posta2(*) :</b></td> <td><input type="text" value="mdogan@aku.edu.tr"/></td> </tr> <tr> <td><b>Web Sitesi</b></td> <td><input type="text" value="www.mevlutdogan.com"/></td> </tr> <tr> <td><b>Ev Tel(*) :</b></td> <td><input type="text" value="(272) 312-2222"/></td> </tr> <tr> <td><b>Cep Tel(*) :</b></td> <td><input type="text" value="(535) 738-8203"/></td> </tr> <tr> <td><b>İş Tel(*)</b></td> <td><input type="text" value="(272) 228-1213"/></td> </tr> <tr> <td><b>Dahili</b></td> <td><input type="text" value="1132"/></td> </tr> <tr> <td><b>Açık Adres (*) :</b></td> <td> <input type="text" value="Dumlupınar Mahallesi"/>   <input type="text" value="İstiklal Caddesi"/>   <input type="text" value="No:101"/>   <input type="text" value="Afyonkarahisar"/> </td> </tr> <tr> <td><b>Posta Kodu(*)</b></td> <td><input type="text" value="03107"/></td> </tr> </table>	<b>SicilNo</b>	<input type="text" value="A_123456"/>	<b>Ad (*) :</b>	<input type="text" value="Mevlüt"/>	<b>Soyad(*) :</b>	<input type="text" value="Doğan"/>	<b>Uyruğu(*) :</b>	<input type="text" value="Türkiye"/>	<b>Nüfusa Kayıtlı Olduğu İl(*) :</b>	<input type="text" value="AFYON"/>	<b>Nüfusa Kayıtlı Olduğu İlçe (*) :</b>	<input type="text" value="MERKEZ"/>	<b>Vatandaşlık No(*) :</b>	<input type="text" value="22123344342"/>	<b>Doğum Tarihi</b>	<input type="text" value="03.03.1968"/>	<b>Ünvanı</b>	<input type="text" value="Prof.Dr"/>	<b>Cinsiyet(*) :</b>	<input checked="" type="radio"/> Erkek <input type="radio"/> Bayan	<b>Kayıtlı Olduğu Askerlik şubesi :</b>	<input type="text" value="AFYON"/>	<b>Askerlik Durumu:</b>	<input type="text" value="MERKEZ ASKERLIK ŞUBESİ"/> <input checked="" type="radio"/> Yaptım <input type="radio"/> Ertele dim <input type="radio"/> Muafım	<b>Açıklama</b>	<input type="text" value="Fizik Bölümü Öğretim Üyesi"/>	<b>Aktif / Pasif</b>	<input type="radio"/> Pasif <input checked="" type="radio"/> Aktif	<b>E-Posta1(*) :</b>	<input type="text" value="mevlutdogan@hotmail.com"/>	<b>E-Posta2(*) :</b>	<input type="text" value="mdogan@aku.edu.tr"/>	<b>Web Sitesi</b>	<input type="text" value="www.mevlutdogan.com"/>	<b>Ev Tel(*) :</b>	<input type="text" value="(272) 312-2222"/>	<b>Cep Tel(*) :</b>	<input type="text" value="(535) 738-8203"/>	<b>İş Tel(*)</b>	<input type="text" value="(272) 228-1213"/>	<b>Dahili</b>	<input type="text" value="1132"/>	<b>Açık Adres (*) :</b>	<input type="text" value="Dumlupınar Mahallesi"/> <input type="text" value="İstiklal Caddesi"/> <input type="text" value="No:101"/> <input type="text" value="Afyonkarahisar"/>	<b>Posta Kodu(*)</b>	<input type="text" value="03107"/>
<b>SicilNo</b>	<input type="text" value="A_123456"/>																																														
<b>Ad (*) :</b>	<input type="text" value="Mevlüt"/>																																														
<b>Soyad(*) :</b>	<input type="text" value="Doğan"/>																																														
<b>Uyruğu(*) :</b>	<input type="text" value="Türkiye"/>																																														
<b>Nüfusa Kayıtlı Olduğu İl(*) :</b>	<input type="text" value="AFYON"/>																																														
<b>Nüfusa Kayıtlı Olduğu İlçe (*) :</b>	<input type="text" value="MERKEZ"/>																																														
<b>Vatandaşlık No(*) :</b>	<input type="text" value="22123344342"/>																																														
<b>Doğum Tarihi</b>	<input type="text" value="03.03.1968"/>																																														
<b>Ünvanı</b>	<input type="text" value="Prof.Dr"/>																																														
<b>Cinsiyet(*) :</b>	<input checked="" type="radio"/> Erkek <input type="radio"/> Bayan																																														
<b>Kayıtlı Olduğu Askerlik şubesi :</b>	<input type="text" value="AFYON"/>																																														
<b>Askerlik Durumu:</b>	<input type="text" value="MERKEZ ASKERLIK ŞUBESİ"/> <input checked="" type="radio"/> Yaptım <input type="radio"/> Ertele dim <input type="radio"/> Muafım																																														
<b>Açıklama</b>	<input type="text" value="Fizik Bölümü Öğretim Üyesi"/>																																														
<b>Aktif / Pasif</b>	<input type="radio"/> Pasif <input checked="" type="radio"/> Aktif																																														
<b>E-Posta1(*) :</b>	<input type="text" value="mevlutdogan@hotmail.com"/>																																														
<b>E-Posta2(*) :</b>	<input type="text" value="mdogan@aku.edu.tr"/>																																														
<b>Web Sitesi</b>	<input type="text" value="www.mevlutdogan.com"/>																																														
<b>Ev Tel(*) :</b>	<input type="text" value="(272) 312-2222"/>																																														
<b>Cep Tel(*) :</b>	<input type="text" value="(535) 738-8203"/>																																														
<b>İş Tel(*)</b>	<input type="text" value="(272) 228-1213"/>																																														
<b>Dahili</b>	<input type="text" value="1132"/>																																														
<b>Açık Adres (*) :</b>	<input type="text" value="Dumlupınar Mahallesi"/> <input type="text" value="İstiklal Caddesi"/> <input type="text" value="No:101"/> <input type="text" value="Afyonkarahisar"/>																																														
<b>Posta Kodu(*)</b>	<input type="text" value="03107"/>																																														

Şekil 5. 20 Peronel Türk vatandaşı, erkek ve askerliğini ertelemişse doldurulacak alanlar.

Eklenen personel Türk vatandaşı değil ise Şekil 5.21’de verilen alanları doldurması gerekmektedir.

  
Kullanıcı Giriş Sayfası

  
Çıkış

  
Mesajlaş

**A - KİŞİSEL BİLGİLERİNİZ** (Detayları Gizle...)

**YENİ PERSONEL**

**A - KİŞİSEL BİLGİLERİNİZ** (Detayları Gizle...)

<b>SicilNo</b>	<input type="text" value="A_213412"/>
<b>Ad (*)</b>	<input type="text" value="Mahmut"/>
<b>Soyad(*)</b>	<input type="text" value="Kantar"/>
<b>Uyruğu(*)</b>	<input type="text" value="Avusturya"/>
<b>Vatandaşlık No(*)</b>	<input type="text" value="21454323221"/>
<b>Doğum Tarihi</b>	<input type="text" value="03.03.1975"/>
<b>Ünvanı</b>	<input type="text" value="Öğretim Görevlisi"/>
<b>Cinsiyet(*)</b>	<input checked="" type="radio"/> Erkek <input type="radio"/> Bayan

**B - İLETİŞİM BİLGİLERİNİZ** (Detayları Gizle...)

**Örnek Telefon Numarası : (272) 228-1213**

<b>E-Posta1(*)</b>	<input type="text" value="m_kantar@aku.edu.tr"/>
<b>E-Posta2(*) :</b>	<input type="text" value="m_kantar@yahoo.com"/>
<b>Web Sitesi</b>	<input type="text" value="-"/>
<b>Ev Tel(*) :</b>	<input type="text" value="(272) 122-1222"/>
<b>Cep Tel(*) :</b>	<input type="text" value="(555) 122-1222"/>
<b>İş Tel(*)</b>	<input type="text" value="(272) 122-1222"/>
<b>Dahili</b>	<input type="text" value="1117"/>
<b>Açık Adres (*) :</b>	<input type="text" value="Cumhuriyet Mahallesi&lt;br/&gt;Ordu Bulvarı&lt;br/&gt;No:91&lt;br/&gt;Afyonkarahisar"/>
<b>Posta Kodu(*)</b>	<input type="text" value="03106"/>

Şekil 5. 21 Eklelen peronel Türk vatandaşı değilse doldurulacak alanlar.



## 5.2.2 Kayıdı Yapılan Öğretim Üyesini Bölüme Atama

Öğretim üyesinin kayıdı tamamlandıktan sonra, görevli olduğu bölüme atanmaktadır. Bu işlemde yetkili birim yine personel daire başkanlığı rolüne sahip kullanıcılardır. Bu kullanıcılar öğretim üyelerini bölümlere atamak için öncelikle işlem yapmak istediği bölümün öğretim üyelerini listelemesi gerekmektedir. Listelenen öğretim üyelerinden seçilenler istenilen bölüme atanabilir. Şekil 5.22’de öğretim üyelerini bölümlere atama işlemi gösterilmektedir.

**Listelemek İstedığınız Öğretim Üyeleri**

Bütün Öğretim Üyeleri

Bölümü Belirtilmemiş Öğretim Üyeleri



Bölümü Belirtilen Öğretim Üyeleri

Sicil No	Adı	Soyadı	Aktif	Seç
121	Mevhut	Dogan	True	<input checked="" type="checkbox"/>

**Öğretim Üyesini Kaydetmek İstedığınız Bölüm**

Okulları Seç Fen Bilimleri Enstitüsü ▼

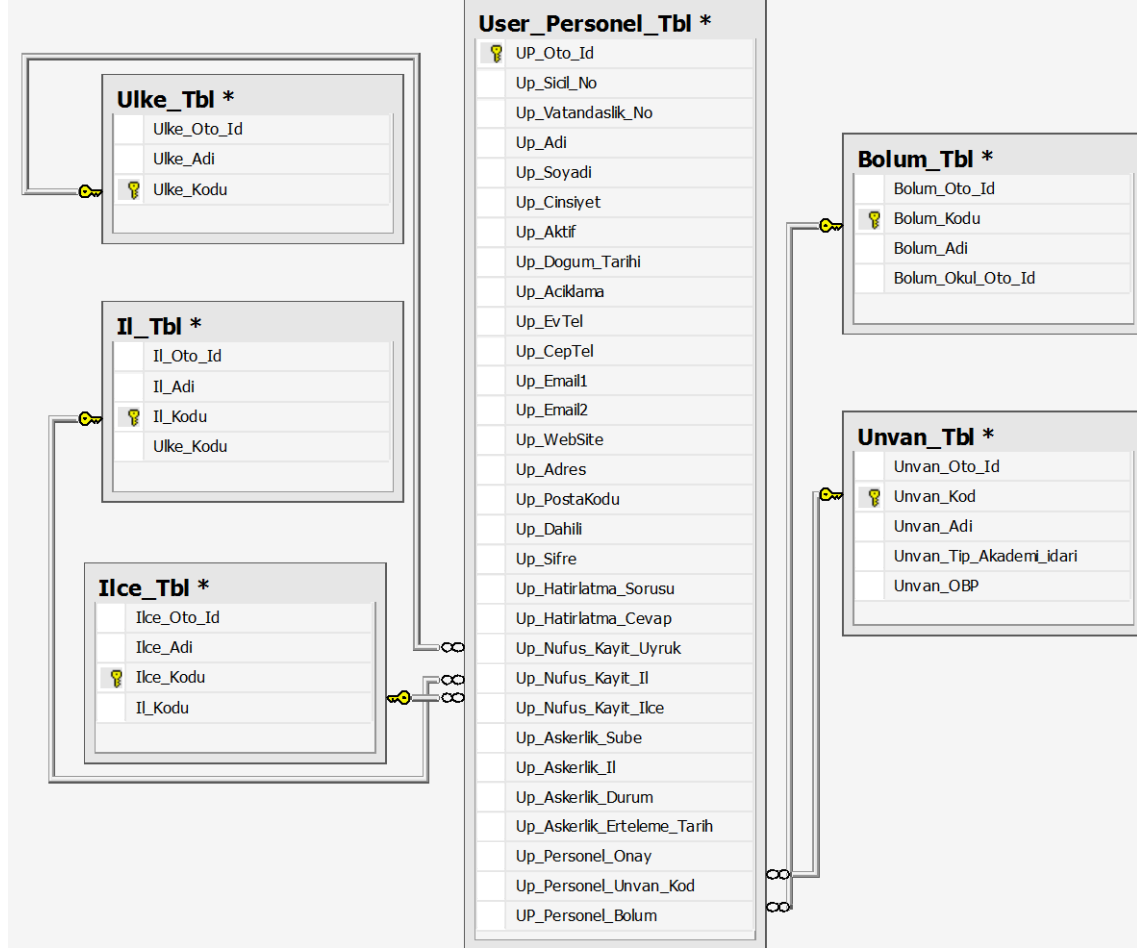
Bölüm Seç Bilgisayar Bölümü ▼

Şekil 5. 22Öğretim üyelerinin bölümlere atanması.

Kullanıcı ve kullanıcı grupları modülünde yapılan işlemler Çizelge 5.4'deki veri tablolarında tutulmaktadır.

**Çizelge 5. 4** Kullanıcı ve Kullanıcı Gurupları modülü ile ilişkili veri tabloları.



Personel kayıt işlemleri Çizelge 5.4'de verilen User\_Personel\_Tbl tablosunda tutulmaktadır. Bu tablodaki alanların açıklaması ise aşağıda verilmiştir.

- **UP\_Oto\_Id:** Personel kayıt işleminde otomatik olarak artan sıralama ID'sidir.
- **Up\_Sicil\_No:** Personel sicil numarasıdır.
- **Up\_Vatandaslik\_No:** Vatandaşlık numarasıdır.
- **Up\_Adi:** Eklenen kişinin ismidir.
- **Up\_Soyadi:** Eklenen kişinin soyismidir.
- **Up\_Cinsiyet:** Cinsiyet bilgisidir. Erkek ise "1" bayan ise "0" bilgisini alır.

- **Up\_Aktif:** Sistemdeki Aktif/Pasif durumudur. Aktif ise “1” değerini alır. Pasif ise “0” değerini alır.
- **Up\_Dogum\_Tarihi:** Doğum tarihidir.
- **Up\_Aciklama:** Ekelen öğretim elemanı ile ilgili kısa bir açıklamadır.
- **Up\_EvTel:** Ev telefon numarasıdır.
- **Up\_CepTel:** Cep telefonu numarasıdır.
- **Up\_Email1:** 1. E-posta adresidir.
- **Up\_Email2:** 2. E-posta adresidir.
- **Up\_WebSite:** Kişisel web sitesi adresidir.
- **Up\_Adres:** İletişim adresidir.
- **Up\_PostaKodu:** Posta kodu bilgisidir.
- **Up\_Dahili:** Dahili numarasıdır.
- **Up\_Nufus\_Kayit\_Uyruk:** Nufusa kayıtlı olduğu ülke kodudur. Ulke\_Tbl deki Ulke\_Kodu ile ilişkilidir.
- **Up\_Nufus\_Kayit\_II:** Nufusa kayıtlı olduğu il kodudur. Il\_Tbl deki Il\_Kodu ile ilişkilidir.
- **Up\_Nufus\_Kayit\_Ilce:** Nufusa kayıtlı olduğu ilçe kodudur. Ilce\_Tbl deki Ilce\_Kodu ile ilişkilidir.
- **Up\_Askerlik\_Sube:** Türk vatandaşı olan erkek personel için kayıtlı olduğu askerlik şubesinin kodudur. Ilce\_Tbl deki Ilce\_Kodu ile ilişkilidir. Her ilçede bir askerlik şubesi olduğundan bu alanla ilişkilendirilmiştir.
- **Up\_Askerlik\_II:** Kayıtlı olduğu askerlik şubesinin bulunduğu il kodudur. Il\_Tbl deki Il\_Kodu ile ilişkilidir.
- **Up\_Askerlik\_Durum:** Türk vatandaşı olan erkek personel için askerlik durumunu belirten alandır.
- **Up\_Askerlik\_Erteleme\_Tarih:** Türk vatandaşı olan erkek personelin askerlik durumu her hangi bir tarihe kadar ertelenmiş ise erteleme bitiş tarihi bilgisi tutulur.
- **Up\_Personel\_Onay:** Kayıt işleminin başarılı olup olmadığı bilgisidir. Kayıt işlemi başarılı ise “1” değerini alır değil ise “0” değerini alır.

- **Up\_Personel\_Unvan\_Kod:** Sahip olduđu unvan kodudur. Unvan\_Tbl'deki Unvan\_kodu ile ilişkilidir.
- **UP\_Personel\_Bolum:** Görevlendirildiđi bölüm kodudur. Bolum\_Tbl deki Bolum\_Kodu ile ilişkilidir.

## 5.3 Rol ve Yetkiler Modülü

### 5.3.1 Rol Tabanlı Güvenlik

Son yıllarda internet hızı olabildiğince gelişmiş, teknoloji yüksek seviyeye ulaşmış, internet kullanıcı sayısı artmıştır. Ancak bunlara paralel olarak internet casusluğunda da artış kendini göstermiştir. Özellikle bir eğitim kurumunun uzaktan öğrenim yönetim sistemi hazırlanacaksa güvenliğe daha fazla dikkat edilmesi gerekmektedir. Microsoft Framework2.0 teknolojisi Asp.net Projelerinde Rol Tabanlı Güvenlik sistemini sunmaktadır. Rol Tabanlı güvenliğe güncel hayattaki fabrika işleyişi örnek olarak verilebilir. Örneğin bir fabrikanın değişik departmanlarında çeşitli roller ve rollere sahip işçiler bulunmaktadır. Fabrika müdürü her işçiye bir görev vermektedir. Yetkiler verilen rollere göre dağıtılmıştır. Örneğin bir pazarlama müdürü bilgi işlem elemanının yapması gereken bir işi yapamaz yada bilgi işlem odasına giremez. Üretim müdürü üretimden sorumludur. Bu bölümde çalışan elemanları yönetebildiği gibi bu bölümdeki yetkili kişide odur. Genel müdürde bütün yetkiler toplanmıştır. İsteddiği kişiyi istediği bölümde müdür yada şef olarak atayabilir. Ama çok gerekli olmadıkça o bölümdeki detaylarla fazla uğraşmaz. Rol Tabanlı güvenlikte ise fabrika örneğindeki gibi görevler ve yetkiler dağıtılmıştır. Her kullanıcı Rolünün izin verdiği alanlara ulaşabilmektedir. Uzaktan öğrenim yönetim sisteminde kullanılan önemli roller Şekil 5.23'de verilmiştir. Bunlar Rektör, Dekan yada Müdür, Öğrenci işleri, Personel başkanlığı, Bölüm Başkanı, Bölüm Sekreteri, Öğretim Elemanı, Administrator ve öğrencilerdir.



REKTÖR



DEKAN/MÜDÜR



ÖĞRENCİ İŞLERİ



PERSONEL BAŞKANLIĞI



BÖLÜM BAŞKANI



BÖLÜM SEKRETERİ



ÖĞRETİM ELEMANI



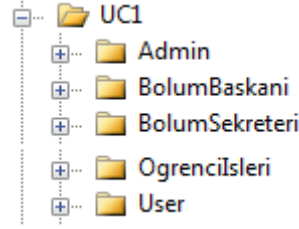
ADMİNİSTRATOR



ÖĞRENCİ SAYFASI

Şekil 5. 23 Uzaktan öğrenim yönetim sisteminde oluşturulan önemli roller.

5.1 bölümünde açıklanan 1. Modül örnek verilerek bu görev ve yetkilerin nasıl dağıtıldığı aşağıda anlatılmaktadır. Online Başvuru Modülündeki Klasörler Şekil 5.24’ de verilmiştir.

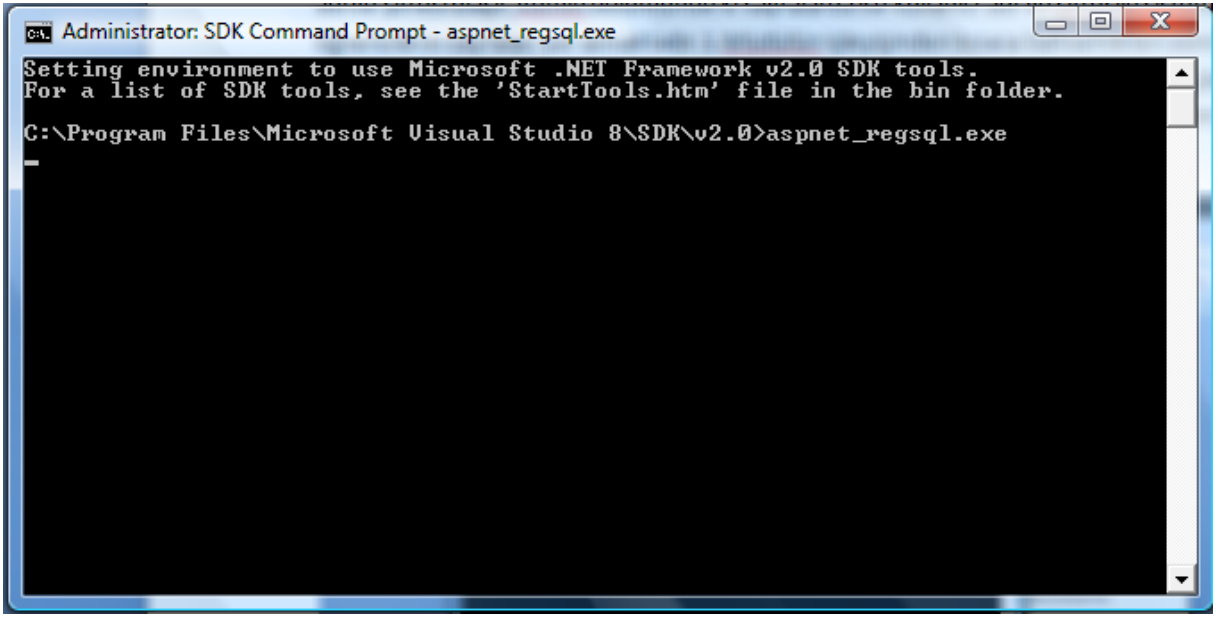


Şekil 5. 24 Online başvuru modülündeki klasörler.

UC1 Modülü, yani Online Başvuru Modülü için tanımlanan kısaltmadır. Sistemde bu modülle ilgili sayfalar ilişkilendirildiği rolle ilgili klasörlerde tutulmaktadır. Online Başvuru Modülü için yaklaşık 5 farklı admin seviyesi bulunmaktadır. Öncelikle **Öğrenciler** User sayfasından “YukseklisansBasvuru.aspx” sayfasına ulaşır. Buradaki başvuru ekranındaki bilgileri doldurarak yüksek lisans için başvurusunu tamamlar. Daha sonra **bölüm sekreteri** bu öğrencilerin doldurduğu bilgileri inceleyerek başvurusunda sorun olmayan öğrencilere mülakat için gerekli bilgileri ve getirmesi gereken evrakların listesini gönderir. Bu süre içerisinde bölüm başkanı başvuru yapan öğrencileri görebilmektedir. Ancak her hangi bir işlem yapamamaktadır. Verilen Mülakat tarihinde öğrenciler geldiğinde mülakata girmeden bölüm sekreteri yanlarında getirdikleri evrakları ve online başvuruda verdiği bilgilerin doğruluğunu kontrol eder. Öğrencinin bilgilerinde ve evraklarında her hangi bir eksiklik yoksa öğrencinin mülakata girmesini onaylar. Onaylanan Öğrenci, **bölüm başkanın** sistemine geçer ve burada öğrencinin mülakattan aldığı notları sisteme girer. Bölüm başkanın onayının ardından bu bilgiler **öğrenci işlerinin** sistemine geçer ve öğrenci işleri kontenjana ve puan sıralamasına göre kazanan öğrencileri ve yedek öğrencileri belirler. Kazanan öğrencilerin mail adreslerine kullanıcı adı ve şifresi gönderilir. Administrator yetkilisi ise Öğrencinin, Bölüm sekreterinin, Bölüm başkanın ve Öğrenci İşlerinin yetki ve sorumluluğunu belirler.

### 5.3.1.1 Projede Rol Tabanlı Güvenlik Oluřturulması

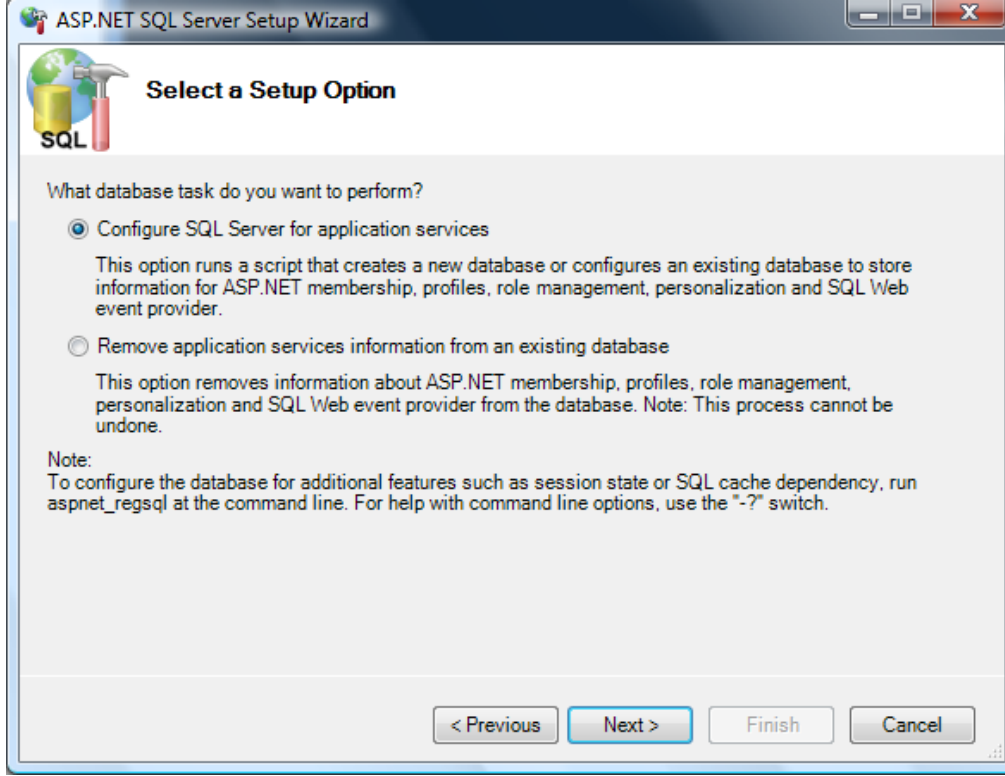
İlk olarak Masaüstü-> Başlat->All Programlar-> Microsoft .NET Framework SDK v2.0 -> SDK Command Prompt seçtikten sonra karşımıza **Dos** ekranı çıkmaktadır. Bu penceredeki aktif satıra `aspnet_regsql.exe` yazılırsa Şekil 5.25’de verilen ekran karşımıza çıkar.



Şekil 5. 25 aspnet\_regsql.exe’ nin çalıştırılması.

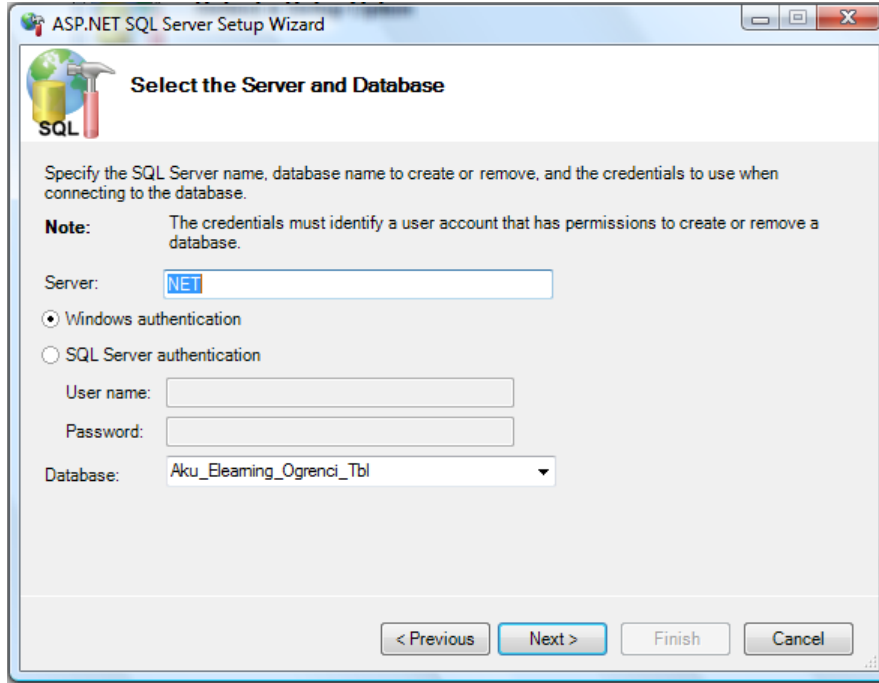
Enter tuşuna basıldıktan sonra Server Setup Wizard penceresi açılmaktadır. İlk pencerede Next ile ilerledikten sonra ikinci pencereden Configure Sql Server for application services seçeneğinin seçilmesi gerekmektedir. Bu ekran Şekil 5.26’da verilmiştir.



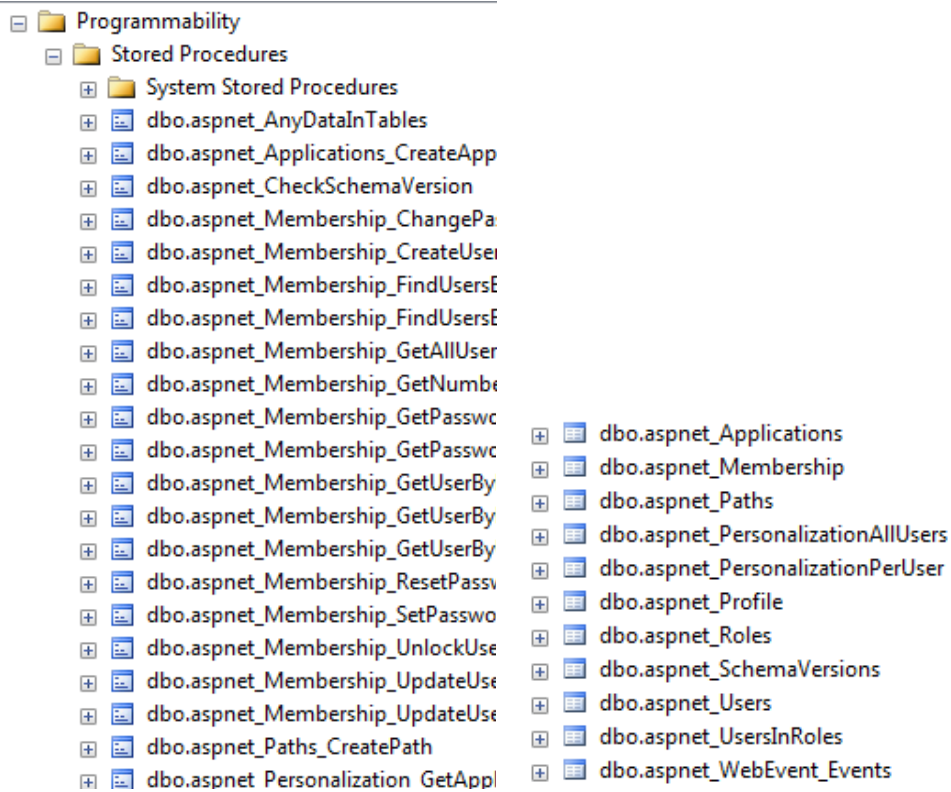


**Şekil 5. 26** Aspnet Procedure' lerinin ve tablolarının Oluşturulması.

Şekil 5.27'de ise kullanılan sunucu ve dataBase seçilmesi gerekmektedir. Next ile işlemi sonlandırdıktan sonra SQL Server'ı başlatıp, tablolar görüntülediğinde bazı yeni tabloların oluştuğu görülmektedir. Programmability klasöründe ise çok fazla sayıda Stored Procedures'ler oluşmaktadır. Bu procedürlerden bazıları Şekil 5.28'de verilmiştir.



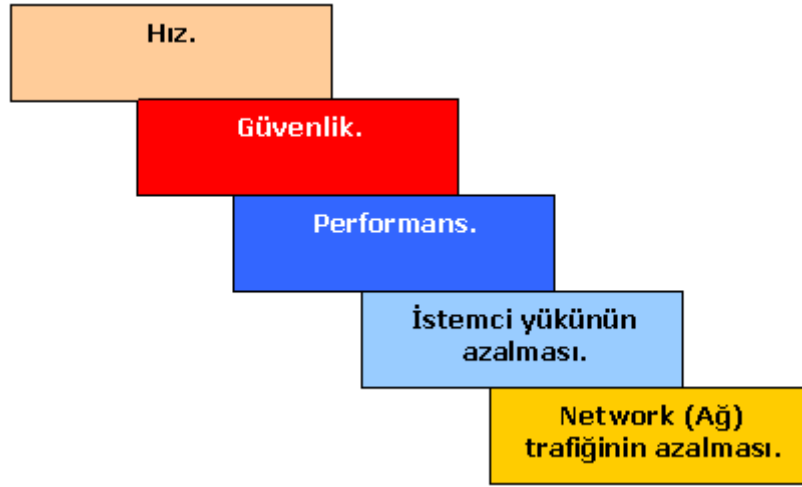
Şekil 5. 27 Database seçimi.



Şekil 5. 28 AspNet procedure tablosu.

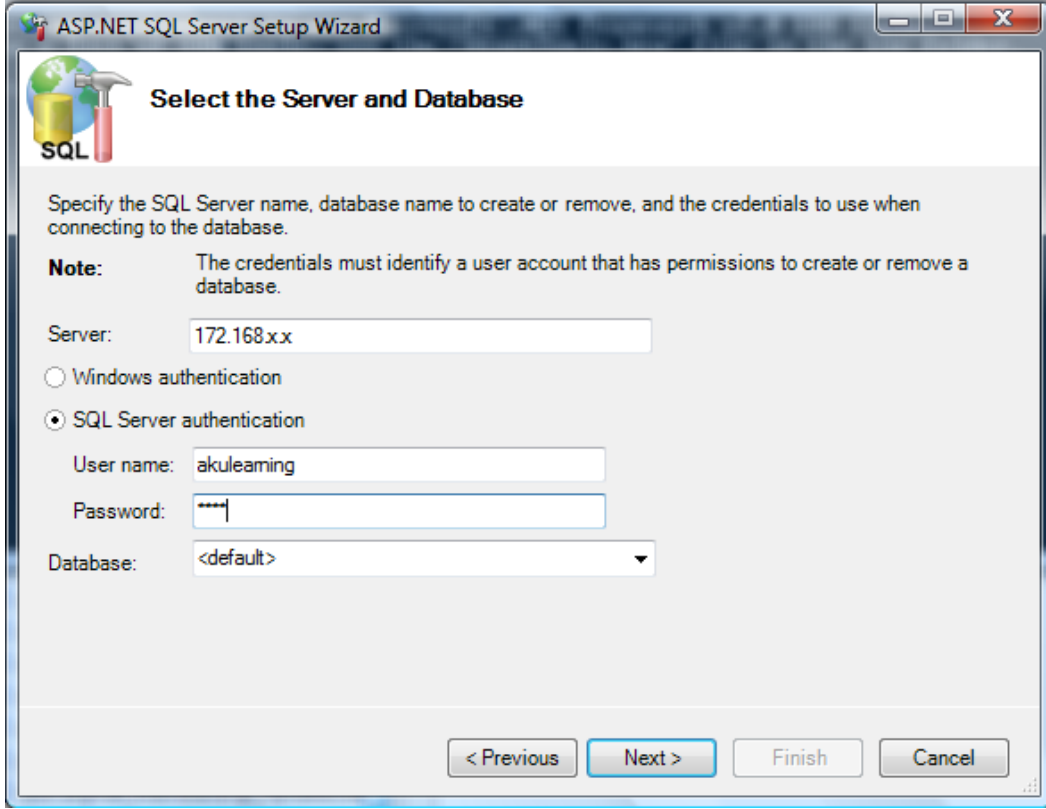
### 5.3.1.2 Stored Procedure'nin Anlamı

Saklı yordamlar derlenmiş sql cümlecikleridir. Bunlar birer veritabanı nesnesi oldukları için, doğrudan veritabanı yöneticisi olan programda (örneğin Sql Server da) yer almaktadırlar. Bu nedenle veritabanı bir yere taşındığında otomatik olarak, saklı yordamlarda taşınmış olur. Bu saklı yordamların tercih edilme nedenlerinden sadece birisidir. Diğer yandan, derlenmiş olmaları aslında bu sql cümleciklerinin doğrudan makine diline dönüştürüldüğü anlamına gelmemektedir. Aslında, çalıştırmak istenilen sql cümlecikleri bir saklı yordam içine yerleştirilerek, bunun bir veritabanı nesnesi haline gelmesini ve çalıştırıldığında doğrudan, veritabanı yöneticisini üzerinde barındıran sunucu makinede işlemlerini sağlar. Bu doğal olarak, istemci makinelerdeki iş yükünü azaltır ve performansı artırır. Nitekim bir program içinde çalıştırılan SQL cümlecikleri, Saklı Yordam' dan çok daha yavaş sonuç döndürür. Dolayısıyla Saklı Yordamlar özellikle çok katlı mimarinin uygulamak istendiği projelerde faydalıdır. Saklı yordamların faydaları Şekil 5.29'da verilmiştir.



Şekil 5. 29 Saklı yordamların faydaları.

Şekil 5.30'da aspnet tablolarının ve aspnet procedurlerinin oluşturulacağı database için gerekli güvenlik ayarlarının girilmesi verilmiştir.



Şekil 5. 30 SQL Server erişim bilgilerinin girilmesi.

Login Kontrolleri LocalSqlServer değilde uzaktaki bir sqlserver da oluşturulması gerekiyorsa durum biraz farklılaşmaktadır. Böyle bir durumda Windows authentication yerine SQL Server authentication seçeneğinin seçilmesi ve karşı taraftaki server'ın Username, Pasword ve ulaşmak istenilen database seçilmesi gerekmektedir. Ancak yinede güvenlik ayarları yüksek tutulan server'lar bu işlemi yapmamıza izin vermemektedir. Bu durumda Başlat->All Programlar-> Microsoft .NET Framework SDK v2.0 -> SDK Command Prompt adımlarından sonra karşımıza gelen **Dos** ekranına aşağıdaki bilgiler yazılmalıdır. Şekil 5.31'de bu komuta örnek verilmiştir.

```
aspnet_regsql -S <ServerName> -U <UserName> -P <Password> d-<DataBaseName>  
-A all
```

```
Administrator: SDK Command Prompt
Setting environment to use Microsoft .NET Framework v2.0 SDK tools.
For a list of SDK tools, see the 'StartTools.htm' file in the bin folder.

C:\Program Files\Microsoft Visual Studio 8\SDK\v2.0>aspnet_regsql -S 216.119.1.72 -U eminibi_akulearning -P ■■■■■■ -d eminibi_logindb -A all

Start adding the following features:
Membership
Profile
RoleManager
Personalization
SqlWebEventProvider

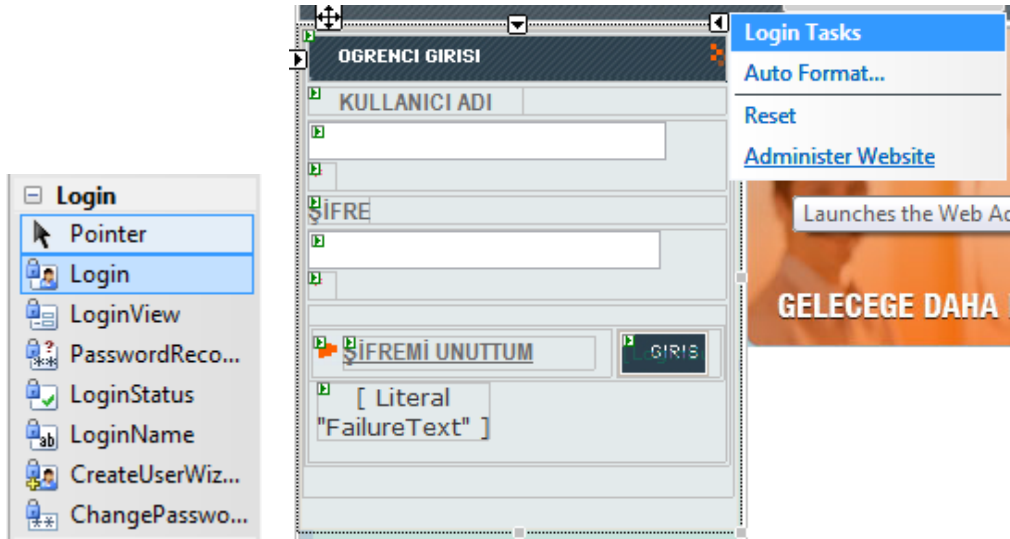
.....

Finished.

C:\Program Files\Microsoft Visual Studio 8\SDK\v2.0>
```

Şekil 5. 31 Peronel TC vatandaşı, erkek ve askerliği ertelemişse doldurulacak alanlar.

Bu işlemler bittikten sonra sisteme Visual Studio penceresindeki ToolBox menüsünde bulunan Login kontrollerinden login elemanını yani kullanıcı girişi kontrolü sayfaya eklenmelidir. Daha sonra properties penceresinden gerekli ayarlamalar yapılarak Şekil 5.32’de de görüldüğü gibi bu pencerenin sağ üst köşesinde bulunan Administer WebSite seçildiği takdirde Şekil 5.33’deki Administer sayfası açılmaktadır.

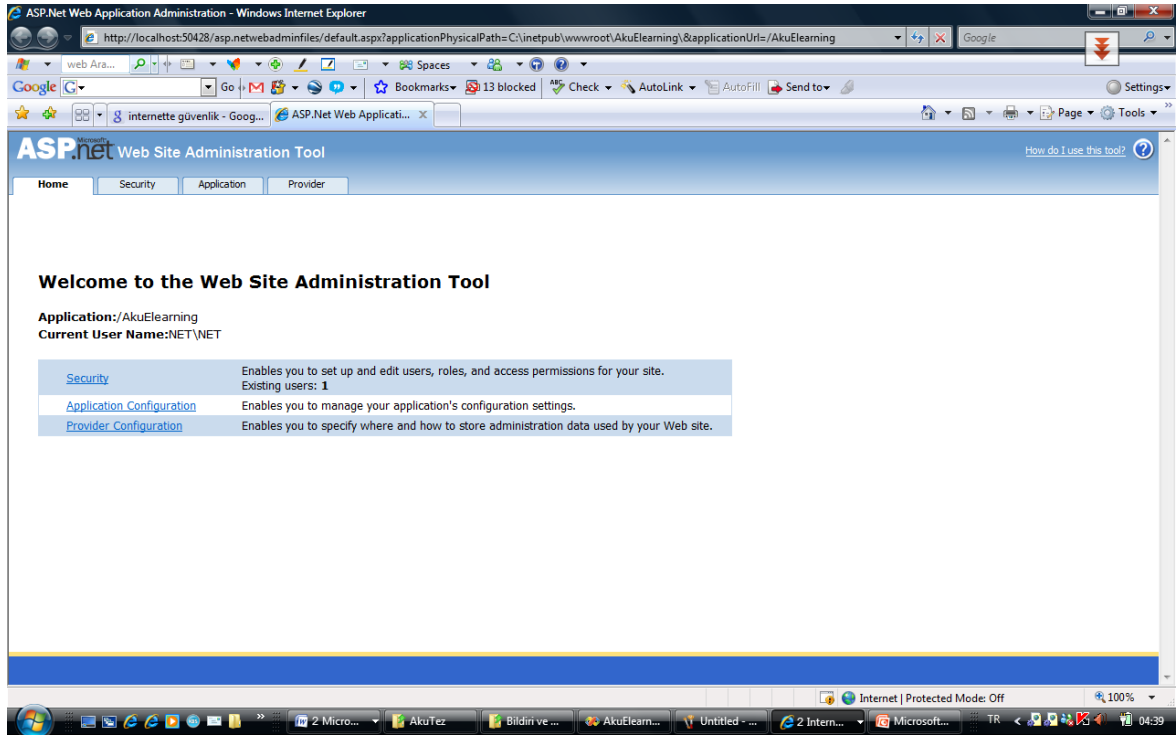


Şekil 5. 32 Login kontrolünün sayfaya eklenmesi.

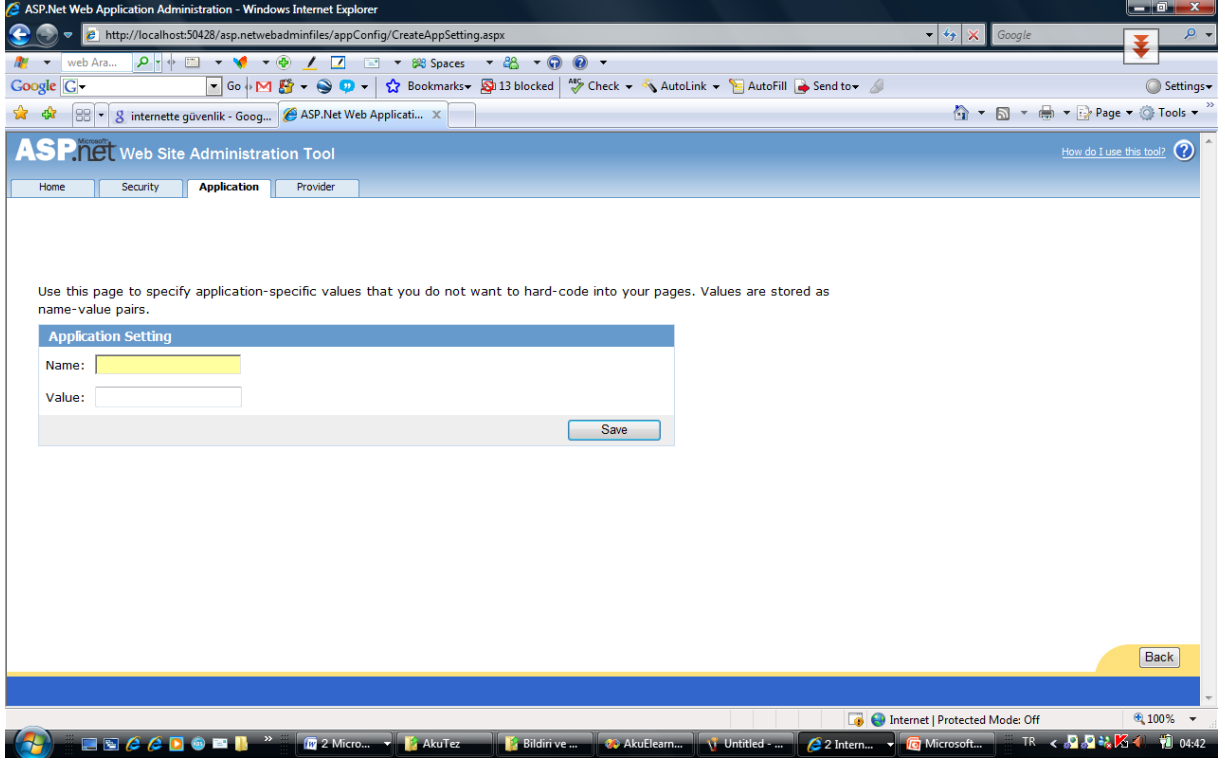
Şekil 5.33’deki Administration Tool ekranında Application sekmesine tıkladıktan sonra Create Application Settings linkinin tıklanması ve açılan sayfada bir adet bağlantı

nesnesinin oluşturulması gerekmektedir. Yada Şekil 5.34'deki gibi bu satıra `ConnectionString` ismini ve value değerlerini ayrı ayrı girmek yerine aşağıdaki kodu `Web.Config` dosyasındaki gereken yere eklenebilir.

```
<connectionStrings>
  <remove name="LocalSqlServer" />
  <add name="LocalSqlServer" connectionString="Data Source=NET;user id=sa;
  pwd=1234;Initial Catalog=Aku_Elearning_Ogrenci_Tbl;Persist Security Info=True" >
</connectionStrings>
```

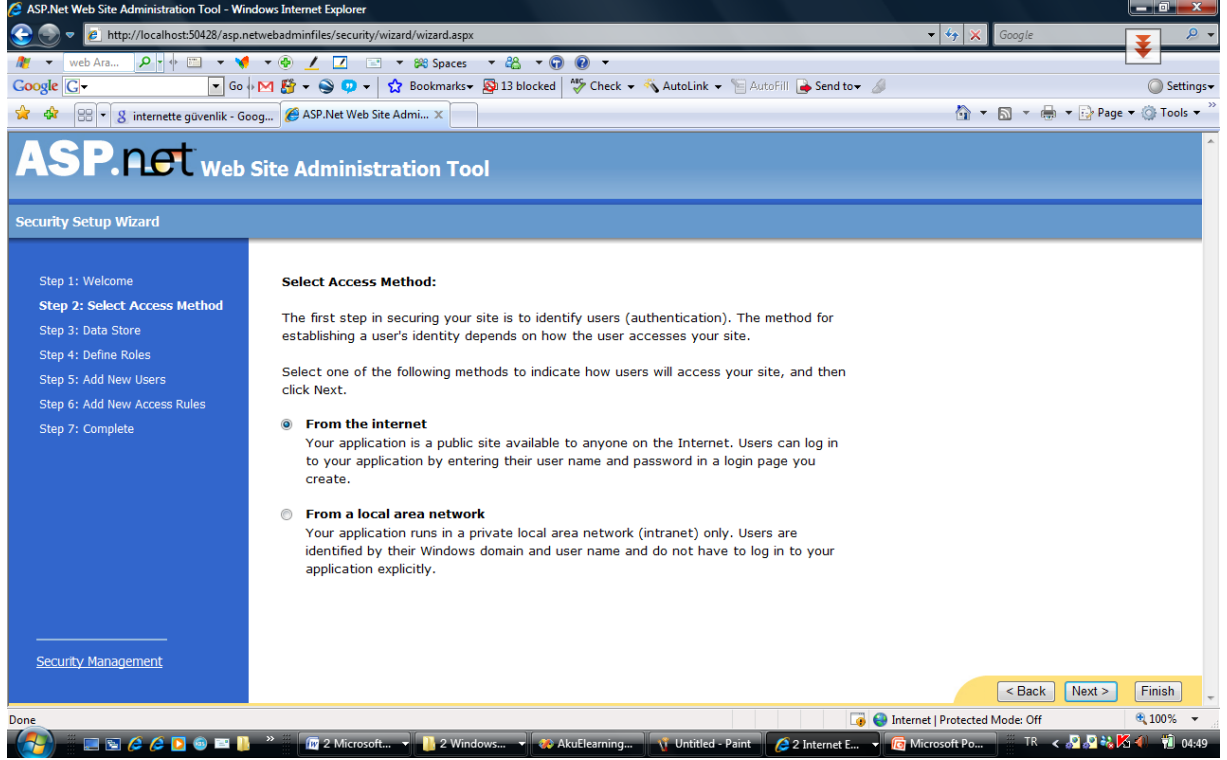


Şekil 5. 33 Administration Tool ekranı.



Şekil 5. 34 Application setting ekranı.

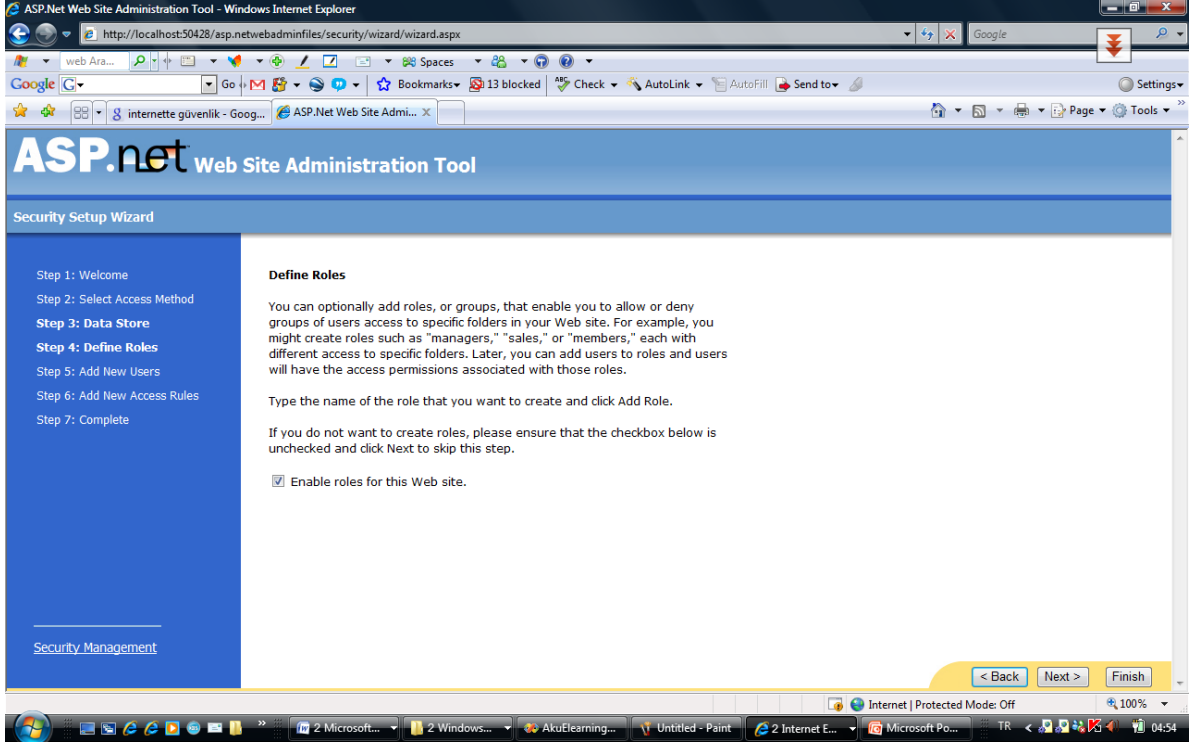
Bu işlem bittikten sonra Home sekmesinin seçilmesi gerekmektedir. Çıkan pencereden önce “Security” daha sonra “Use the security Setup Wizard to configure security step by step.” linkini kullanarak **Security Setup Wizard** yardımı ile gerekli işlemler yapılmalıdır. Security Setup Wizard seçeneği seçildikten sonra Şekil 5.35’deki erişim metodu ekranı karşımıza çıkmaktadır.



Şekil 5. 35 Erişim metodu.

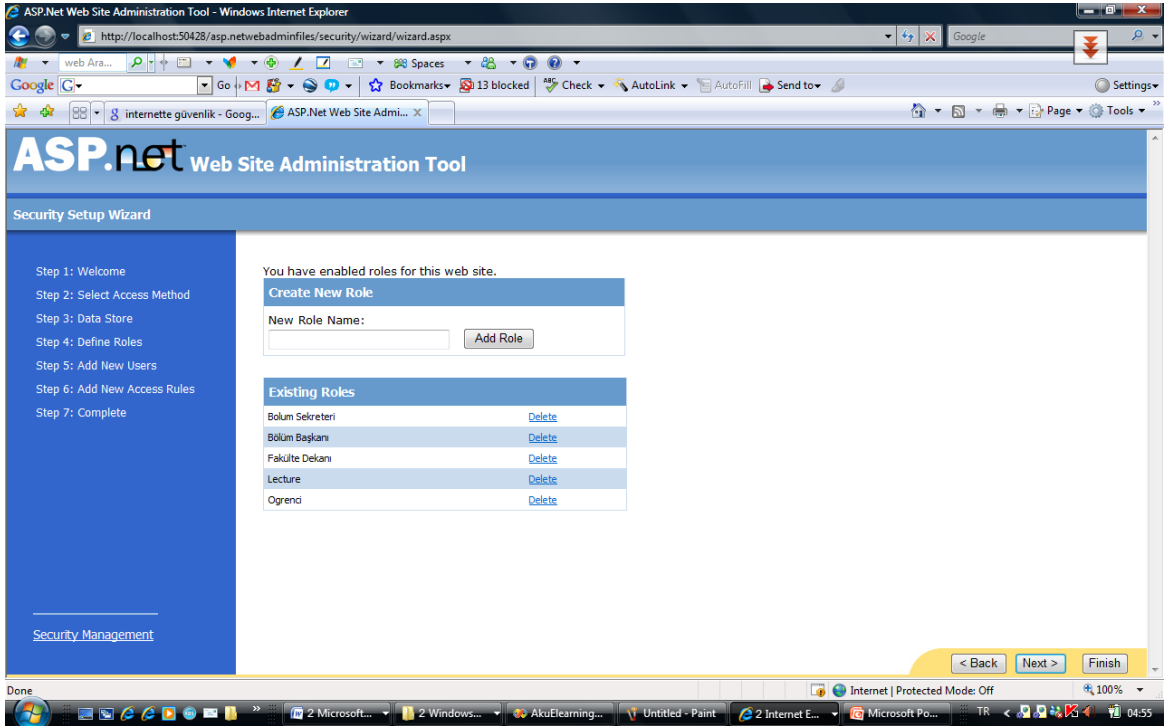
Ekrana 7 adımdan oluşan bir **Security Setup Wizard** penceresi çıkacaktır. 1. adımda wizard penceresi ile ilgili kısa bir bilgi vermektedir. 2. Adımda erişim metodunun belirtilmesi gerekmektedir. Şayet sadece local'de kullanılacaksa “from a local area” network seçimi yapılabilir. 3. Sekmede kısa bir bilgi verilmektedir. 5. adımda ise Enable roles for his website kutucuğu aktif hale getirilmelidir. Bu ekran Şekil 5.36'da verilmektedir.





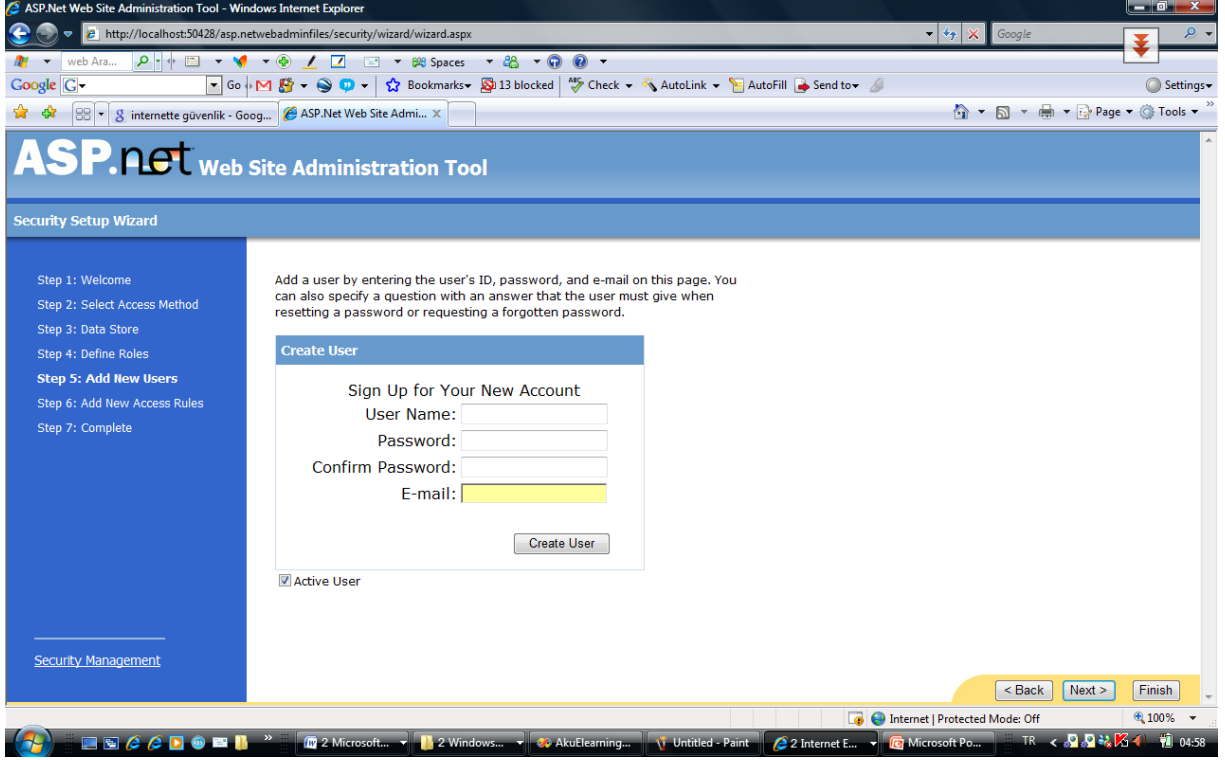
Şekil 5. 36 Rollerin aktifleştirilmesi.

Sisteme eklenen roller Şekil 5.37’de verilmektedir.



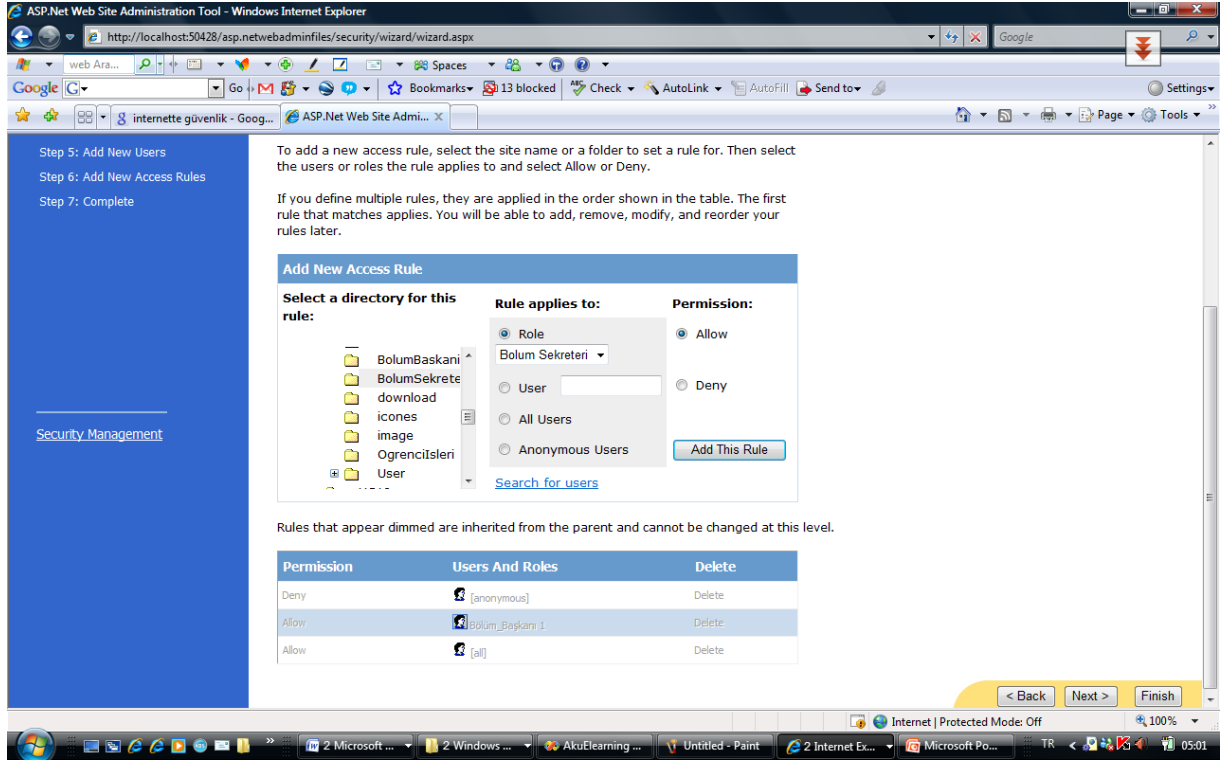
Şekil 5. 37 Sisteme yeni rolün eklenmesi.

5.adım ise yeni kullanıcı ekleme işlemlerinin yapıldığı adımdır. Şekil 5.38’de bu ekran verilmektedir.



Şekil 5. 38 Yeni kullanıcı kaydı.

Kullanıcı kayıt bölümü kullanılarak yeni kullanıcılar eklenebilir. 6. Adımda ise Rollerin yetkileri belirtilmelidir.




Şekil 5. 39 Erişim kuralları.

Örneğin BolumSekreteri klasörüne Bölüm Sekreteri Rolüne Sahip Kullanıcılar Erişebilsin veya Öğrenci işleri Rolündeki kişiler bu klasöre erişemesin gibi. 7 adım sonunda ise artık Rol Tabanlı sistem aktif hale gelmiş olur. Şayet eklenen kullanıcılarla ilgili yeni işlemler yapılması gerekiyorsa “Security Management” linkini tıkladıktan sonra karşımıza çıkan pencereden Manage Users Seçeneği ile kullanıcılarla ilgili işlemler yapılabilir. Ancak bu noktadan sonra şu soru akla gelmektedir. Bahsedilen bu uygulamaları yapmak için bilgisayarda sqlserver ve framework kurulu olmalıdır. Bu işlemleri yapacak kişinin server bilgilerini bilmesi gerekir. Böyle bir durumda yukarıda açıkladığımız işlemleri yapmak pek mantıklı olamamaktadır. Bunun yerine bu işlemleri aspx sayfalarına dönüştürerek kodlar ile bu işlemlerin yapılması gerekmektedir. Üzerinde çalışılan uzaktan yönetim sistemi projesinde bu işlemler aspx sayfalarına dönüştürülerek dinamik hale getirilmiştir. İlk olarak bir kullanıcı adı ve şifre oluşturulmuştur. Bu şifre ve kullanıcı adı administrator şifresi ve kullanıcı ismidir. Sisteme administaror yetkisine sahip kullanıcı adı ve şifresi ile giriş yapan kullanıcı yukarıda bahsedilen işlemleri online olarak yapabilmektedir. Yazılımı tamamlanan uzaktan eğitim projesinde 12 aspx sayfası ile yukarıdaki işlemler yapılabilir.


## 5.3.2 Rol Tabanlı Sistemin Dinamik Hale Getirilmesi


### 5.3.2.1 Klasör Yetkileri


Şekil 5.40'da emin\_ibili adlı kullanıcının erişebileceği klasörler ve erişemeyeceği klasörler gösterilmektedir. Yeşil olan klasörler bu kullanıcının erişim izni bulunduğu klasörler, kırmızı olanlar ise erişim izni olmadığı klasörlerdir. Klasörlerdeki yetkilendirme işlemleri kullanıcı adına göre yapabildiği gibi rollere göre de yapılabilmektedir. Örneğin BolumBaskani klasörüne Bolum Sekreteri kullanıcılarını engelle gibi.





Aktif Kullanıcılar    Kullanıcı Ekle    Kullanıcı Düzenle    Kullanıcı Kilitle    Kullanıcılar    Online Kullanıcılar

  
KLASÖR YETKİLERİ

  
ERİŞİM KURALLARI

  
ROLLER

  
KULLANICI PROFİLLERİ

  
ROLLERİ DÜZENLE

### 1) Klasör Yetkileri

**Website Erişim Güvenliği Özeti**

Rol Seç  emin\_ibili

- AkuElearning (ALLOW: All Users)
- Administrator (ALLOW: All Users)
- access (ALLOW: All Users)
- activity (ALLOW: All Users)
- (ALLOW: All Users)
- icons (ALLOW: All Users)
- management (ALLOW: All Users)
- Akuimage (ALLOW: All Users)
- App\_Code (ALLOW: All Users)
- Bin (ALLOW: All Users)
- editor (DENY: Role=Bölüm Başkanı)
- css (DENY: Role=Bölüm Başkanı)
- behaviors (DENY: Role=Bölüm Başkanı)
- images (DENY: Role=Bölüm Başkanı)
- dialog (DENY: Role=Bölüm Başkanı)
- common (DENY: Role=Bölüm Başkanı)
- images (DENY: Role=Bölüm Başkanı)
- fck\_about (DENY: Role=Bölüm Başkanı)
- fck\_docprops (DENY: Role=Bölüm Başkanı)
- fck\_flash (DENY: Role=Bölüm Başkanı)
- fck\_image (DENY: Role=Bölüm Başkanı)
- fck\_link (DENY: Role=Bölüm Başkanı)
- fck\_select (DENY: Role=Bölüm Başkanı)
- fck\_spellerpages (DENY: Role=Bölüm Başkanı)
- spellerpages (DENY: Role=Bölüm Başkanı)
- spellerpages (DENY: Role=Bölüm Başkanı)

Şekil 5. 40 Klasör yetkileri.

### 5.3.2.2 Web Sitesi Erişim Kuralları

Şekil 5.41’de gösterildiği gibi emin\_ibili kullanıcısı için BoluBaskani klasörüne tam yetki verilmiştir. Bu klasör için diğer tam yetki erişimine sahip kullanıcıları seçebildiğimiz gibi kısıtlı kullanıcıları, kısıtlı rolleri yada rollerin bu klasöre olan erişim izinlerin belirlenebilmektedir.

**Web Site Erişim Kuralları**  
~/UC1/BoluBaskani/'a Kurallar Oluştur

Olay	Roller	Kullanıcılar	Kullanıcı Sil	Kural Taşı(Sil)
Allow		emin_ibili	Delete Rule	Up Down
Deny		?	Daha yüksek düzeyden alınan	Daha yüksek düzeyden alınan
Allow	Bölüm_Başkanı 1		Daha yüksek düzeyden alınan	Daha yüksek düzeyden alınan
Allow		*	Daha yüksek düzeyden alınan	Daha yüksek düzeyden alınan

---

**~/UC1/BoluBaskani/İçin Yeni Kural Oluştur**

Olay:  Kısıtlı  Tam Yetki

**Kural Uygula:**

Bu Rol: Bölüm Başkanı

Bu Kullanıcı: emin\_ibili

Bütün Kullanıcılar (\*)


Giriş Yapmayan Kullanıcılar (?)

Create Rule


Şekil 5. 41 Web sitesi erişim kuralları.

### 5.3.2.3 Rollerin Belirlenmesi


Şekil 5.42’de sisteme rollerin eklenmesi ve silinmesi gösterilmektedir. Ayrıca roller ile ilgili kullanıcı sayıları verilmektedir.




KLASÖR YETKİLERİ




ERİŞİM KURALLARI



ROLLER



KULLANICI PROFİLLERİ



ROLLERİ DÜZENLE

### Roller

Rol Adı	Kullanıcı Sayısı	Kullanıcı Sil
Bölüm Sekreteri	1	<input type="button" value="Delete"/>
Bölüm Başkanı	1	<input type="button" value="Delete"/>
Fakülte Dekanı	1	<input type="button" value="Delete"/>
Lecture	0	<input type="button" value="Delete"/>
Oğrenci	0	<input type="button" value="Delete"/>

New Role:

Şekil 5. 42 Rol ekleme ve silme.

#### 5.3.2.4 Kullanıcı Profilleri

Şekil 5.43'deki raporla birlikte kullanıcıların sistem içerisindeki hareketliliği gösterilmektedir. Ayrıca kullanıcı ismi, e-posta, oluşturulma tarihi, son girişi tarihi, son aktivite tarihi, aktif mi, online mı, çıkış yapmış mı gibi bilgiler bu sayfadan öğrenilebilmektedir.



KLASÖR YETKİLERİ



ERİŞİM KURALLARI



ROLLER



KULLANICI PROFİLLERİ



ROLLERİ DÜZENLE

Role filter:

Kullanıcı Adı	Email	Bilgiler	Oluşturma Tarihi	Son Giriş Tarihi	Son Aktivite Tarihi	Aktif mi	Online mı?	Çıkış Yapmış mı?
<a href="#">emin_ibilı</a>	emin_ibilı4@hotmail.com	123	18.04.2008 02:20:28	28.08.2008 22:34:13	28.08.2008 22:34:13	True	False	False

Şekil 5. 43 Roller ve kullanıcıları.

### 5.3.2.5 Aktif/Pasif Kullanıcılar

Aktif/Pasif kullanıcılar sayfasında, sistemdeki aktif kullanıcılar yada pasif kullanıcılar görüntülenmektedir. Bu sayfa Şekil 5.44’de görüntülenmektedir.



Şekil 5. 44 Aktif/Pasif kullanıcılar.

### 5.3.2.6 Kullanıcı Ekleme

Şekil 5.45'deki kullanıcı ekleme işleminde doldurulması gereken alanlar;

1. Rol Seçimi
2. Aktif/Pasif
3. Kullanıcı Adı
4. Password
5. E-posta
6. Sicil No

Rol Seçiminde aynı kullanıcıya birden fazla rol verilebilmektedir. Ancak burada dikkat edilmesi gereken nokta aynı kullanıcı adı ve e-posta adresi ile sisteme kullanıcı kaydedilememektedir. Kesinlikle bu bilgilerin farklı olması gerekmektedir. Lecture rolüne sahip kullanıcı personel daire başkanlığı sayfasında bulunan personel ekleme işlemi ile de yapılabilmektedir. Öğrenci ise sisteme, öğrenci işlerinin öğrenciyi kazandı olarak onaylamasının ardından, sistemde otomatik olarak AN + OkulKodu + BölümKodu + ProgramKodu + SıraNo ile başlayan bir kullanıcı adı oluşmaktadır. Şifresi ise Random olarak üretilir ve bu iki bilgi kullanıcının mail adresine gönderilir.



Lecture ve öğrenci adlı bu iki rol online başvuru modülünde ve kullanıcı ve kullanıcı gurupları modülünde detaylı olarak anlatılmıştır.



**Kullanıcı Ekle**

**Roller:**

Bolum Sekreteri

Bolum Başkanı

Fakülte Dekanı

Lecture

Ogrenci

**Ana Bilgi:**

Aktif User

Kullanıcı Adı

Password

Email

Sicil No

Şekil 5. 45 Kullanıcı ekleme işlemleri.

### 5.3.2.7 Kullanıcı Bilgilerini Düzenleme ve Onaylama

Kullanıcı bilgilerini düzenlemesi için öncelikle kullanıcıyı seçmek gerekmektedir. Kullanıcıların listelendiği sayfa Şekil 5.46’da gösterilmektedir.




The screenshot displays a user management interface. At the top, there are six navigation buttons with icons: 'Aktif Kullanıcılar' (Active Users), 'Kullanıcı Ekle' (Add User), 'Kullanıcı Düzenle' (Edit User), 'Kullanıcı Kilitle' (Lock User), 'Kullanıcılar' (Users), and 'Online Kullanıcılar' (Online Users). Below these buttons is a search filter labeled 'Kullanıcı Süzgeci' with a dropdown menu set to 'All' and a list of letters from A to Z. The main part of the interface is a table with the following columns: 'Kullanıcı Adı', 'Email', 'Bilgiler', 'Oluşturma Tarihi', 'Son Giriş Tarihi', 'Son Aktivite Tarihi', 'Aktif mi', 'Online mı?', and 'Sistem Dışı mı?'. Two users are listed in the table.

Kullanıcı Adı	Email	Bilgiler	Oluşturma Tarihi	Son Giriş Tarihi	Son Aktivite Tarihi	Aktif mi	Online mı?	Sistem Dışı mı?
<a href="#">emin ibili</a>	emin_ibili4@hotmail.com	123	18.04.2008 02:20:28	28.08.2008 22:34:13	28.08.2008 22:34:13	True	False	False
<a href="#">Mevlut Dogan</a>	dogan_m@hotmail.com	A32145	28.08.2008 23:44:00	28.08.2008 23:44:00	28.08.2008 23:44:00	True	False	False

Şekil 5. 46 Kullanıcı bilgilerini düzenleme.

İşlem yapılacak kullanıcı seçildikten sonra sayfa da kullanıcı bilgileri yer almaktadır. Gerekli düzenlemeler yapıldıktan sonra Kullanıcı Bilgilerini Düzenle butonuna basılarak onaylanması gerekmektedir. Kullanıcı bilgilerinin düzenlendiği sayfa Şekil 5.47’de verilmiştir.



Aktif Kullanıcılar   Kullanıcı Ekle   Kullanıcı Düzenle   Kullanıcı Kilitle   Kullanıcılar   Online Kullanıcılar

**Kullanıcı Bilgisi**

**Eoller:**

Bölüm Sekreteri

Bölüm Başkanı

Fakülte Dekanı

Lecture

Öğrenci

**Ana Bilgi:**

Kullanıcı Adı	Mevlut_Dogan
Email	dogan_m@hotmail.com
Bilgi	A32145
Aktif User	<input checked="" type="checkbox"/>
Çıkmış mı	<input type="checkbox"/>
Online mı?	<input type="checkbox"/>
CreationDate	28.08.2008 23:44:00
Son Aktivite Tarihi	28.08.2008 23:44:00
Son Giriş Tarihi	28.08.2008 23:44:00
Son Çıkış Tarihi	01.01.1754 02:00:00
Son Şifre Değiştirme Tarihi	28.08.2008 23:44:00

Şekil 5. 47 Kullanıcı bilgilerinin belirtilmesi.

Kullanıcı Kilitle, Kullanıcılar ve Online Kullanıcılar, bu 3 işlemde de kullanıcılar listelenmektedir. Online kullanıcılar butonu ile o an sistemde olan kullanıcılar, Kullanıcılar butonu ile bütün kullanıcılar, Kullanıcı Kilitle butonu ile de yine sistemde

olan bütün kullanıcılar listelenir. İşlem yapılmak istenilen yâda pasif hale getirilmek istenilen kullanıcı sistemden seçilebilmektedir. Sistem de kayıtlı olan kullanıcılar Şekil 5.48’de gösterilmektedir.



The screenshot shows a user management interface. At the top, there are six navigation icons: a single green person (Aktif Kullanıcılar), a blue person with a green plus sign (Kullanıcı Ekle), a group of blue and red people (Kullanıcı Düzenle), a blue person with a red minus sign (Kullanıcı Kilitle), a group of blue people (Kullanıcılar), and a group of red and orange people on a red platform (Online Kullanıcılar). Below the icons is a search bar labeled 'Kullanıcı Süzgeçi:' with a dropdown menu set to 'All' and a list of letters from A to Z. Below the search bar is a table with the following columns: 'Kullanıcı Adı', 'Email', 'Bilgiler', 'Oluşturma Tarihi', 'Son Giriş Tarihi', 'Son Aktivite Tarihi', 'Aktif mi?', 'Online mı?', and 'Sistem Dışı mı?'. The table contains two rows of user data.

Kullanıcı Adı	Email	Bilgiler	Oluşturma Tarihi	Son Giriş Tarihi	Son Aktivite Tarihi	Aktif mi?	Online mı?	Sistem Dışı mı?
<a href="#">emin ibili</a>	emin_ibili4@hotmail.com	123	18.04.2008 02:20:28	28.08.2008 22:34:13	28.08.2008 22:34:13	True	False	False
<a href="#">Mevhut Dogan</a>	dogan_m@hotmail.com	A32145	28.08.2008 23:44:00	28.08.2008 23:44:00	28.08.2008 23:44:00	True	False	False

Şekil 5. 48 Sistemdeki kayıtlı kullanıcılar.

### 5.3.3 Aynı Role Sahip Kullanıcılar Arasındaki Yetkilendirme

Bölüm başkan, fakülte dekanı, bölüm sekreteri olarak kaydedilen bir kullanıcı, bölüm başkanı klasörüne erişebilmektedir. Ancak 2 yada daha fazla sayıda bu rollere sahip kullanıcı varsa bu kullanıcıların yetkileri belirlenmesi, örneğin hangi bölüm başkanının hangi bölümlerde yetki sahibi olabileceği, yada Fakülte Dekanı hangi fakültelerde, Bölüm Sekreteri hangi bölümlerde yetki sahibi olacağı gibi işlemler RollerDuzenle sayfasından yapılmaktadır. Ancak bu işlemlerin nasıl yapıldığına geçmeden bazı kavramların netlik bulması gerekmektedir.

Sistemde 3 adet yetki seviyesi bulunmaktadır.

**1. Seviye Okul:** Okul kavramı Enstitü, Fakülte, Yüksek Okul yada Meslek Yüksek okulu nu kapsamaktadır.

**Örnek:** Fakülte Dekanını 1. seviyeden yetkilidir ve okulla ilişkilidir. Bir fakülte dekanı belirtilirken sadece okulu seçmek yeterlidir. Yada bir Enstitü Müdürü belirtilmek isteniyorsa sadece Enstitüyü seçmek yeterlidir. Örnek olarak Sosyal Bilimler Enstitüsü Müdürü. Bu da 1. seviyeden bir yetkidir.

2. **Seviye Bölüm:** Bölüm kavramı örnek olarak, Fen Edebiyat Fakültesi Fizik Bölümü, Fen Bilimleri Enstitüsü Bilgisayar Bölümü, Yada Meslek Yüksek Okulu Mobilya Bölümü gibi bölüm kavramlarını kapsamaktadır. Ancak dikkat edildiği gibi bir bölümü tarif edebilmek için önce okul adı seçilmeli sonra bölüm ismi seçilmelidir.

**Örnek:** Bölüm Başkanı, Bölüm sekreteri gibi atamalar 2. seviyedendir. Yani bu rollere kullanıcı atarken öncelikli olarak Okul seçilmesi gerekmektedir, sonra ise bölümün seçilmesi gerekmektedir.

Fen Bilimleri Enstitüsü, Matematik Bölümü, Bölüm Başkanı  
Teknik Eğitim Fakültesi, Bilgisayar Bölümü, Bölüm Sekreteri

3. **Seviye Program:** Program ise bölüm altında açılan programdır. Örneğin;

Okul: Eğitim Fakültesi  
Bölüm: İlk Öğretim Bölümü  
Program: Sınıf Öğretmenliği Programı

**Örnek:** Bu seviye çok sık kullanılmamaktadır. Örnek olarak aşağıdaki gibi durumlarda kullanılmaktadır. Fen Bilimleri Enstitüsü, Fizik Bölümü yüksek lisansın programına başvuru yapan öğrencilerin işlemlerini takip edecek sorumlu atamak istendiğinde kullanılmaktadır. Örneğin; Fen Bilimleri Enstitüsü, Bilgisayar Bölümü, Bilgisayar Öğretim Teknolojileri programına başvuru yapan öğrencilerin takibini yapmak için emin\_ibili kullanıcısı ders sorumlusu yapılabilmektedir.

### 5.3.3.1 Uygulama

Bölüm Başkanı rolüne sahip olan Mevlut\_Dogan kullanıcısının Fen Bilimleri Enstitüsü, Fizik Bölümü, Bölüm Başkanı olarak atanması. İlk olarak Şekil 5.49'da gösterildiği gibi kullanıcılar listesinden Mevlut\_Dogan kullanıcısının seçilmesi gerekmektedir.



**Roller ve Yetkiler**

Rol Seçiniz

	Kullanıcı Adı	Email	Sicil No	Aktif mi
<a href="#">Select</a>	emin_ibili	emin_ibili4@hotmail.com	123	True
<a href="#">Select</a>	Mevlut_Dogan	dogan_m@hotmail.com	A32145	True






Şekil 5. 49 Sistemdeki kayıtlı kullanıcılar.

İşlem yapılacak kullanıcı seçildikten sonra açılan pencereden Bölüm Başkanı Rolünün sahip olacağı görev ve yetkiler belirlenmelidir. Bu görev ve yetkiler;

- 1) Okul Tipi
- 2) Akademik/idari
- 3) Görev Alanı
- 4) Okulu


5) Bölümüdür.


Şekil 5. 50'de örnek ekran görüntüsü verilmiştir.


 KLASÖR YETKİLERİ	<b>Roller ve Yetkiler</b>
 ERİŞİM KURALLARI	Seçili Kullanıcı: <b>Mevlut_Dogan</b>
 ROLLER	Okul Tipini Belirtiniz: <input checked="" type="radio"/> Enstitü <input type="radio"/> Fakülte <input type="radio"/> Yüksek Okul <input type="radio"/> Meslek Yüksek Okul
 KULLANICI PROFİLLERİ	Akademik/İdari: <input type="radio"/> Akademik <input checked="" type="radio"/> İdari
 ROLLERİ DÜZENLE	Görev Alanı: Bolum İcin Görev Tanımla Seçiniz Okul İçin Görev Tanımla Bolum İcin Görev Tanımla Program İcin Görev Tanımla
	Okulları Seç: Fen Bilimleri Enstitüsü
	Bolum Seç: Fizik Bolumu
	Program Seç:
	<b>Mevlut_Dogan Kullanıcısına Fizik Bolumu İçin Bölüm Başkanı Görevi Verilsin mi?</b>
	<input checked="" type="checkbox"/> <input type="checkbox"/>


Şekil 5. 50 Bölüm başkanı rolünün verilmesi.


Şayet Mevlut\_Dogan kullanıcısının fakülte dekanı olarak atanması gerekiyorsa bunun için öncelikle Mevlut\_Dogan kullanıcısına Fakülte Dekanı rolünün verilmesi gerekmektedir. Bu işlem Şekil 5.51'de gösterilen Rollerü Düzenle sayfasından yapılmaktadır.

  
 KLASÖR YETKİLERİ

  
 ERİŞİM KURALLARI

  
 ROLLER

  
 KULLANICI PROFİLLERİ

  
 ROLLERİ DÜZENLE

### Kullanıcı Bilgisi

#### Eoller:








Bölüm Sekreteri  
 Bölüm Başkanı  
 Fakülte Dekanı  
 Lecture  
 Öğrenci

#### Ana Bilgi:

Kullanıcı Adı	Mevlut_Dogan
Email	dogan_m@hotmail.com
Bilgi	A32145
Aktif User	<input checked="" type="checkbox"/>
Çıkış mı	<input type="checkbox"/>
Online mı?	<input type="checkbox"/>
CreationDate	28.08.2008 23:44:00
Son Aktivite Tarihi	28.08.2008 23:44:00
Son Giriş Tarihi	28.08.2008 23:44:00
Son Çıkış Tarihi	01.01.1754 02:00:00
Son Şifre Değişirme Tarihi	28.08.2008 23:44:00

**Şekil 5. 51** Kullanıcıya yeni bir rolün verilmesi.



 KLASÖR YETKİLERİ  ERİŞİM KURALLARI  ROLLER  KULLANICI PROFİLLERİ  ROLLERİ DÜZENLE	Seçili Kullanıcı	Mevlut_Dogan
	Okul Tipini Belirtiniz	<input type="radio"/> Enstitü <input checked="" type="radio"/> Fakülte <input type="radio"/> Yüksek Okul <input type="radio"/> Meslek Yüksek Okul
	Akademik/İdari	<input type="radio"/> Akademik <input checked="" type="radio"/> İdari
	Gorev Alanı	Okul İçin Görev Tanımla ▼
	Okulları Seç	Fen Edebiyat Fakültesi ▼
	Bölüm Seç Program Seç	Seçiniz Teknik Eğitim Fakültesi Fen Edebiyat Fakültesi
	Mevlut_Dogan Kullanıcısına Fen Edebiyat Fakültesi İçin Fakülte Dekanı Görevi Verilsin mi?  	

Şekil 5. 52 Verilen yeni rolün görevlendirileceği fakülte.

Bu yapılan işlemler aspnet tablolarında aşağıdaki gibi tutulmaktadır.

- **dbo.aspnet\_Users**

Kullanıcı adı, UserId arasındaki ilişki Çizelge 5.5’de gösterilen sql tablosunda tutulmaktadır.

**Çizelge 5. 5** aspnet\_users tablosu.

Table - dbo.aspnet_Users Summary							
ApplicationId	UserId	UserName	LoweredUserN...	MobileAlias	IsAnonymous	LastActivityDate	
97a4-44119f9ff9b9	11393256-e442-4335-8ff2-6d128d006706	emin_ibili	emin_ibili	NULL	False	28.08.2008 23:16:00	
ced92acc-bff8-42b2-97a4-44119f9ff9b9	c12f4fd0-a79b-444d-9f33-436cfd5c8fe1	Mevlut_Dogan	mevlut_dogan	NULL	False	28.08.2008 20:44:00	
97981709-7ac0-467f-b67e-47187d457504	dbdaa08c-c4ea-4b50-b3ac-0ef5108e6945	emin_ibili	emin_ibili	NULL	False	25.08.2008 06:56:00	
97981709-7ac0-467f-b67e-47187d457504	468df3c3-01e6-4ff8-acdc-b07df2c9032e	Mevlut_Dogan	mevlut_dogan	NULL	False	28.08.2008 20:44:00	
NULL	NULL	NULL	NULL	NULL	NULL	NULL	

- **aspnet\_Membership tablosu**

Çizelge 5.6’da gösterilen aspnet\_Membership tablosunda ise kullanıcıların diğer tüm bilgileri bu tabloda tutulmaktadır. Şifresi, e-posta adresi, son giriş tarihi, aktivite tarihi vs. dikkat edilirse 1. Tablodaki emin\_ibili adlı kullanıcının UserId’si ile bu tablodaki UserId ilişkilendirilmiştir.

**Çizelge 5. 6** aspnet\_Membership tablosu.

Table - dbo.aspnet_Membership Summary								
ApplicationId	UserId	Password	Passw...	PasswordSalt	MobilePIN	Email	LoweredEmail	
97a4-44119f9ff9b9	c12f4fd0-a79b-444d-9f33-436cfd5c8fe1	12345	0	Pbrf3cZOEmVYF...	NULL	dogan_m@hotmail.com	dogan_m@hotmail.com	
ced92acc-bff8-42b2-97a4-44119f9ff9b9	11393256-e442-4335-8ff2-6d128d006706	1234	0	luQo4gUBheD+e...	NULL	emin_ibili@hotmail.com	emin_ibili@hotmail.com	
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

- **aspnet\_Roles tablosu**

Rollerin isimleri ve Rol id leri Çizelge 5.7’deki aspnet\_Roles tablosunda tutulmaktadır.

**Çizelge 5. 7** aspnet\_Roles tablosu.

Table - dbo.aspnet_Roles Summary					
ApplicationId	RoleId	RoleName	LoweredRoleNa...	Description	
67e-47187d457504	081c4e2b-00ff-4dc4-01a-03ff89515fc9	Bolum Sekreteri	bolum sekreteri	NULL	
97981709-7ac0-467f-b67e-47187d457504	65f6f930-4299-42ef-9d67-c4447b9cc157	Bölüm Başkanı	bölüm başkanı	NULL	
97981709-7ac0-467f-b67e-47187d457504	2ef5ffc1-d9fc-499e-839e-9908d783ed2e	Fakülte Dekanı	fakülte dekanı	NULL	
97981709-7ac0-467f-b67e-47187d457504	80eea32e-78cf-4ac6-9c7e-250e1d7c6fa0	Lecture	lecture	NULL	
97981709-7ac0-467f-b67e-47187d457504	3cf03420-0333-494e-a38c-f55e8ec19b5b	Ogrenci	ogrenci	NULL	
NULL	NULL	NULL	NULL	NULL	

- **aspnet\_UsersInRoles tablosu**

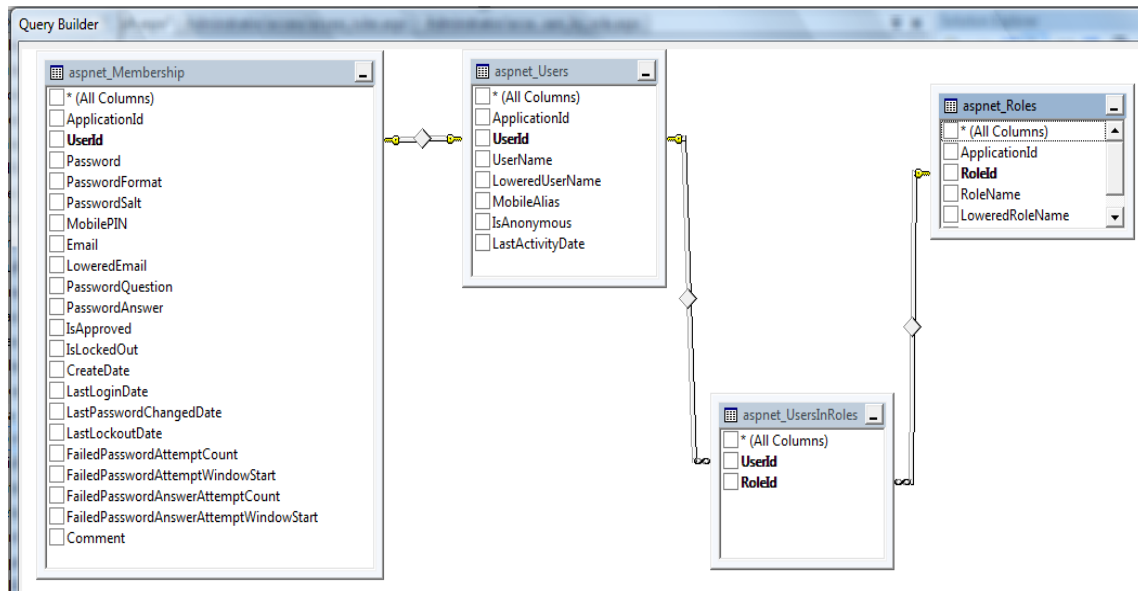
Çizelge 5.8’de verilen aspnet\_UsersInRoles tablosunda kullanıcının UserId’si ve sahip olduğu rollerin ID’leri bu tabloda tutulmaktadır.

**Çizelge 5. 8** aspnet\_UsersInRoles tablosu.

Table - dbo.aspnet_UsersInRoles Summary	
Userid	RoleId
b3ac-0ef5108e6945	081c4e2b-00ff-4dc4-a01a-03ff89515fc9
468df3c3-01e6-4ff8-acdc-b07df2c9032e	80eea32e-78cf-4ac6-9c7e-250e1d7c6fa0
dbdaa08c-c4ea-4b50-b3ac-0ef5108e6945	2ef5ffc1-d9fc-499e-839e-9908d783ed2e
468df3c3-01e6-4ff8-acdc-b07df2c9032e	2ef5ffc1-d9fc-499e-839e-9908d783ed2e
dbdaa08c-c4ea-4b50-b3ac-0ef5108e6945	65f6f930-4299-42ef-9d67-c4447b9cc157
468df3c3-01e6-4ff8-acdc-b07df2c9032e	65f6f930-4299-42ef-9d67-c4447b9cc157
* NULL	NULL

Bu 4 tablo arasındaki ilişkilendirme ise Çizelge 5.9’da verilmiştir.

Çizelge 5.9 aspnet\_Membership, aspnet\_users, aspnet\_UserInRoles, aspnet\_Roles tabloları arasındaki ilişki.



Örneği basite indirgeyerek anlatmak gerekirse Çizelge 5.10’deki ilişki mevcuttur.

Çizelge 5. 10 aspnet\_Membership, aspnet\_users, aspnet\_UserInRoles, aspnet\_Roles tabloları arasındaki ilişkinin basite indirgenmiş hali.

<b>RolTablosu(aspnet_Roles tablosu)</b>				
<b>RoleId</b>		<b>RolAdı</b>		
Rol1		BolumBaskani		
Rol2		Fakülte Dekanı		

aspnet\_Roles tablosundaki RolId ve RolName alanıdır.

<b>Kullanıcı Tablosu(aspnet_Users tablosu)</b>	
<b>UserId</b>	<b>UserAdi</b>
User123	emin_ibili

Aspnet\_Users tablosundaki UserId ve UserName alanıdır.

<b>KullanıcıRolTablosu(aspnet_UserInRoles tablosu)</b>	
<b>UserId</b>	<b>RoleId</b>
User123	Rol1
User123	Rol2

aspnet\_UserInRoles tablosundaki UserId ve RolId alanıdır.

<b>KullanıcıBilgileri Tablosu (aspnet_Membership tablosu)</b>				
<b>UserId</b>	<b>Pasword</b>	<b>Email</b>	<b>Son Giriş Tarihi</b>	<b>Aktivasyon Tarihi</b>
User123	12345	<u>emin_ibili@hotmail.com</u>	30.03.2009	25.03.2009

Aspnet\_Membership tablosundaki UserId, Pasword, Email, LastLoginDate, CreateDate alanlarıdır.

5.10'daki tabloda bu 4 tablo daki önemli alanlar arasındaki ilişkilendirme basite indirgenerek verilmiştir.

#### 5.4 Ünvan Tanımlama Modülü

Bu modülde akademik ve idari unvanlar tanımlanmaktadır. Sisteme kaydı yapılan öğretim üyesinin, yada öğretim elemanın sahip olduğu unvanlar bu modülde tanımlanmaktadır. Akademik ünvanlar Öğr.Grv., Yrd.Doç.Dr., Doç.Dr., Prof.Dr. vs. dir. İdari ünvanlar ise Bölüm Sekreteri, Bölüm Başkanı, Dekan, Dekan Yrd. vs. dir. Unvan tanımlama işlemini Personel Daire Başkanlığı yapmaktadır. Her unvan bir unvan koduna sahiptir. Bu unvan kodları devlet personel daire başkanlığından alınmaktadır. Örneğin Profösör ünvanın bu birimde kullandığı kod A\_1212 gibidir. Sistemde aynı isimde birden fazla unvan tanımlanamamaktadır. Aksi takdirde kullanıcıya hata bilgisi göndermektedir. Unvan tanımlama işleminde unvan idari bir unvan ise hangi birimle ilgili bir unvan oluşu belirtilmelidir. Örneğin; Sisteme dekan olarak kaydedilen bir idari unvan Okul ile ilişkilidir. Eğer akademik bir unvan ise bütün akademik ünvanlar okul biriminin altına kaydedilmektedir. Çizelge 5.12’de bu örnek çoğaltılarak verilmiştir.

**Çizelge 5. 11 Ünvan ve bağlı olduğu birimler.**

	Okul	Bölüm	Program
Dekan	*		
Enstitü Müdürü	*		
Yüksek Okul Müdürü	*		
Bölüm Sekreteri		*	
Bölüm Başkanı		*	
Danışman			*
Enstitü Sekreteri	*		

Tanımlanan ünvanlar, öğretim elemanlarını yada öğretim üyelerini sisteme kaydederken kullanılmaktadır. Öğretim elemanını sisteme kaydetme işlemi bölüm

5.2’de verilmiştir. Sisteme ünvan ekleme işlemine örnek Şekil 5.53’de verilmiştir.

Unvan Kodu	<input type="text" value="C_103"/>
Unvan Adı	<input type="text" value="Dekan"/>
Unvan Tipi	<input checked="" type="radio"/> Akademik <input type="radio"/> İdari
Unvan OBP	<input type="radio"/> Okul <input checked="" type="radio"/> Bölüm <input type="radio"/> Prgram
<a href="#">Ekle</a> <a href="#">İptal</a>	



Yeni Ünvan

Unvan No	Unvan Kodu	Unvan Adı	Unvan Tipi	Unvan OBP	
6	C_103	Dekan	<input type="radio"/> Akademik <input checked="" type="radio"/> İdari	<input checked="" type="radio"/> Okul <input type="radio"/> Bölüm <input type="radio"/> Prgram	<a href="#">Düzenle</a> <a href="#">Yeni Ekle</a> <a href="#">Delete</a>
5	B_102	Doc.Dr	<input checked="" type="radio"/> Akademik <input type="radio"/> İdari	<input checked="" type="radio"/> Okul <input type="radio"/> Bölüm <input type="radio"/> Prgram	<a href="#">Düzenle</a> <a href="#">Yeni Ekle</a> <a href="#">Delete</a>
4	A_101	Prof.Dr	<input checked="" type="radio"/> Akademik <input type="radio"/> İdari	<input checked="" type="radio"/> Okul <input type="radio"/> Bölüm <input type="radio"/> Prgram	<a href="#">Düzenle</a> <a href="#">Yeni Ekle</a> <a href="#">Delete</a>
1	A_901	Bohum Baskani	<input type="radio"/> Akademik <input checked="" type="radio"/> İdari	<input type="radio"/> Okul <input checked="" type="radio"/> Bölüm <input type="radio"/> Prgram	<a href="#">Düzenle</a> <a href="#">Yeni Ekle</a> <a href="#">Delete</a>

Şekil 5. 53 Yeni ünvan ekleme sayfası.

Şekil 5.53’den de anlaşılacağı üzere bu veriler üzerinde düzenleme ve silme işlemi yapılabilmektedir. Ancak silme işlemi yapabilmek için, silinecek ünvana bağlı bir personelin bulunmaması gerekmektedir. Aksi takdirde bu işlem gerçekleştirilememektedir. Ünvan olarak tanımlanan veriler Çizelge 5.13’deki SQL tablosunda tutulmaktadır.

**Çizelge 5. 12** Ünvan tablosu.

Unvan_Tbl	
Unvan_Kod	
Unvan_Adi	
Unvan_Tip_Akademi_idari	
Unvan_OBP	

Çizelge 5.13 de belirtilen alanların açıklaması aşağıda verilmiştir.

- **Unvan\_Oto\_Id:** Sisteme her ünvan eklendiğinde otomatik olarak artan sıralama ID'sidir.
- **Unvan\_Kod:** Personel daire başkanlığının belirlediği ünvan kodudur.
- **Unvan\_Adi:** Tanımlanan ünvanın ismidir.
- **Unvan\_Tip\_Akademik\_İdari:** Tanımlanan ünvanın tipidir. Akademik mi, idari mi olduğunu tanımlamaktadır. Akademi ise sistem de "0" değerini alır, idari ise "1" değerini alır.
- **Unvan\_OBP:** Tanımlanan ünvanın kayıtlı olduğu birimi tanımlar. Okul mu, Birim mi yada Program mı? olduğunu belirtmektedir. Okul ise "1", Bölüm ise "2", Program ise "3" değerini alır.

## 5.5 Okul Tanımlama Modülü

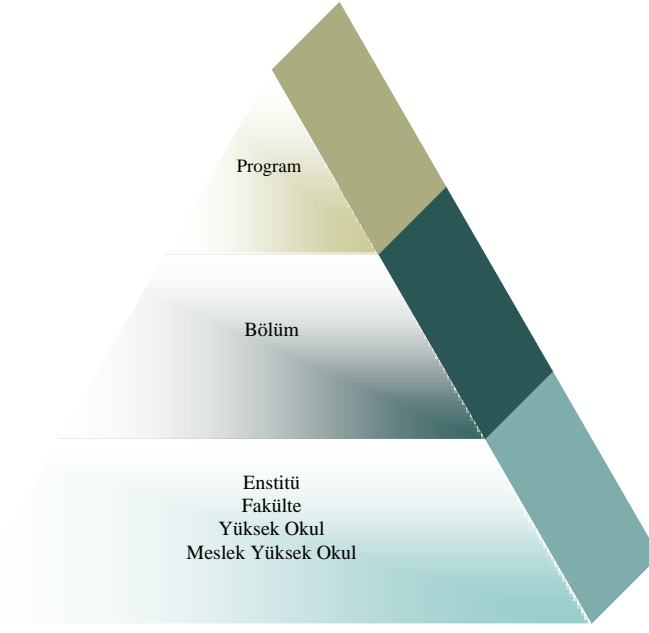
Sistemde 4 tane okul tipi bulunmaktadır. Bunlar Enstitü, Fakülte, Yüksek Okul ve Meslek Yüksek Okuludur. Bu okulları sisteme ekleme işlemini Rektör yada Rektör tarafından yetkili olan birim yapmaktadır. Sistemde açılacak okullar OSYM tarafından ilan edilen okullar olması gerekmektedir. Aksi takdirde 1. modülde anlatılan Online Başvuru Modülündeki 5.1.1.4 Eğitim Bilgileri Ekranının bölümünde de anlatıldığı üzere, XML verilerinin sisteme aktarıldığı tablolar ile uyumsuzluk olacağı için sistem kabul etmeyecektir. Üniversite, Fakülte, Program ve bunların kodları OSYM tarafından gelen Excel sayfalarından XML dönüşümleri sayfası ile hem veri tabanına aktarılır, hem de XML verilerine dönüştürülür. Bu bölümde açılması gereken okullar ve bunların kodları sisteme rastgele olarak kesinlikle girilemez.

Oluşturulan sistemin ilk aşamada yüksek lisans olarak kullanılması hedeflenmiştir. Ancak ileriki zamanlarda lisans içinde kullanılabilir olması için okul tipi sayısı çoğaltılmıştır. Bu sebeple oluşturulan analiz raporları ve gerçekleştirilen uzaktan öğrenim yönetim sistemi bir üniversite alt yapısının bütününe hedef olarak tasarlanmıştır. Bu bağlamda üniversitemizde 4 farklı okul tipinin olduğu tespit edilmiştir. Ancak bu okulların dördü de ortak bir işletme organizasyonu çerçevesinde tasarlanmıştır. Örneğin;

- Enstitünün altında Bölüm, Bölümün altında Program  
Fen Bilimleri Enstitüsü, Bilgisayar Anabilim dalı, Bilgisayar Yüksek Lisans Programı.
- Fakültenin altında bir bölüm, bölümün altında Program  
Teknik Eğitim Fakültesi, Bilgisayar ve Elektronik Eğitimi Bölümü, Bilgisayar Öğretmenliği gibi.

Bu işletme orgnizasyon yapısı Şekil 5.54'de gösterilmektedir.





**Şekil 5. 54** İşletme organizasyonu.


Bu işletme organizasyonu düşünülerek Okul Tanımlama Modülü oluşturulmuştur. Yeni okul tanımlamak için öncelikle yetkili kullanıcının sisteme giriş yapması gerekmektedir. Yetkili kişi rektör yada rektör tarafından görevlendirilen birimdeki bir kişi olmalıdır. Yetkili kullanıcı önce okul tanımlama ile ilgili bağlantı aracılığı ile okul tanımlama sayfasına ulaşır ve aşağıdaki adımları takip eder.

1. Okul kodu verisini girer.
2. Okul adını girer.
3. Okul tipini seçer.

Okul ekle işlemini tetikler. Şayet sistemde bu okul kodu ile bir okul girişi yapılmamış ise sistem başarılı sonuç verecektir. Aksi takdirde kullanıcıya hata mesajı döndürecektir.

Okul ekleme işlemine örnek Şekil 5.55’de verilmiştir.

Okul Kodu	901
Okul Adı	Fen Bilimleri Enstitüsü
Okul Tipi	<input checked="" type="radio"/> Enstitü <input type="radio"/> Fakülte <input type="radio"/> Yüksek Okul <input type="radio"/> Meslek Yüksek Okul
<a href="#">Ekle</a> <a href="#">Cancel</a>	








[Okulla Göre Listele](#)

<u>Okul No</u>	<u>Okul Kodu</u>	<u>Okul Adı</u>	<u>Okul Tipi</u>	
<a href="#">Edit</a> <a href="#">Yeni</a> 11	901	Fen Bilimleri Enstitüsü	<input checked="" type="radio"/> Enstitü <input type="radio"/> Fakülte <input type="radio"/> Yüksek Okul <input type="radio"/> Meslek Yüksek Okul	<a href="#">Delete</a>

Şekil 5. 55 Okul ekleme işlemi.

Şekil 5.55’de görüldüğü gibi bu sayfada yeni okul tanımlama işlemi ile birlikte okul silme ve güncelleme işlemi de yapılabilmektedir. Ancak okul güncelleme işleminde doğru bölüm kodunun girilmesine dikkat edilmesi gerekmektedir. Okul silmek için, silenecek okula bağlı hiçbir bölümün bağlı olmaması gerekmektedir. Ayrıca Okul Tipi, Okul Kodu, Okul Adına göre listeleme de yapılabilmektedir.

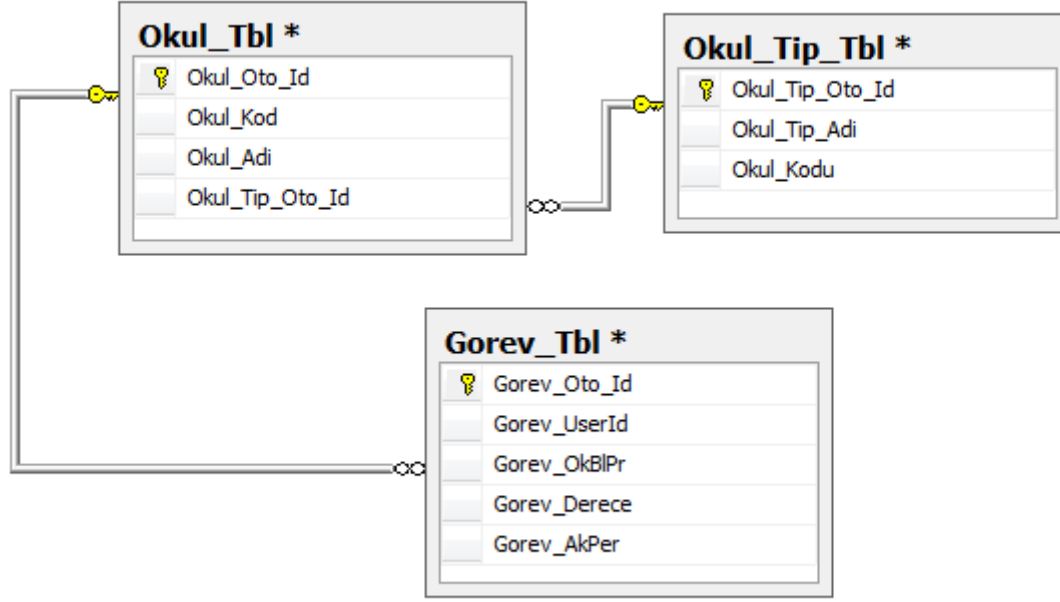
Sisteme eklenen okullara Dekan yada Müdür ekleme işlemi Rol ve Yetkiler modülünde yapılmaktadır. Bu konu 5.2 Rol ve Yetkiler Modülü konusunda ayrıntılı olarak anlatılmıştır. Ayrıca Şekil 5.56’da örnek bir dekan atama işlemi verilmiştir.

 KLASÖR YETKİLERİ  ERİŞİM KURALLARI  ROLLER  KULLANICI PROFİLLERİ  ROLLERİ DÜZENLE	Seçili Kullanıcı	Mevlut_Dogan
	Okul Tipini Belirtiniz	<input type="radio"/> Enstitü <input checked="" type="radio"/> Fakülte <input type="radio"/> Yüksek Okul <input type="radio"/> Meslek Yüksek Okul
	Akademik/İdari	<input type="radio"/> Akademik <input checked="" type="radio"/> İdari
	Gorev Alanı	Okul İçin Görev Tanımla ▼
	Okulları Seç	Fen Edebiyat Fakültesi ▼
Bohum Seç	Seçiniz	
Program Seç	Teknik Eğitim Fakültesi Fen Edebiyat Fakültesi	
	<b>Mevlut_Dogan Kullanıcısına Fen Edebiyat Fakültesi İçin Fakülte Dekanı Gorevi Verilsin mi?</b>	
		

**Şekil 5. 56** Sisteme eklenen okullara Dekan yada Müdür ekleme işlemi.

Okul modülünde yapılan işlemler Çizelge 5.14'te verilen Okul\_Tbl tablosunda tutulmaktadır.

**Çizelge 5. 13** Okul tablosu ve bu tablo ile ilişkili olan Okul\_Tip\_Tbl tablosu.



Bu modülde yapılan işlemler Çizelge 5.14 deki 3 tablo ile ilişkilendirilebilir. Okul\_Tbl tablosunda bulunan alanlar aşağıda anlatılmıştır.

**Okul\_Oto\_Id:** Sisteme her okul eklendiğinde otomatik olarak artan sıralama ID'sidir.

**Okul\_Kod:** ÖSYM tarafından tanımlanan okul kodudur.

**Okul\_Adi:** Tanımlanan okul adıdır.

**Okul\_Tip\_Oto\_Id:** Tanımlanan okul tipidir. Enstitü ise "1", Fakülte ise "2", Yüksek Okul ise "3", Meslek Yüksek Okulu ise "4" değerini alır. Bu alan Okul\_Tip\_Tbl tablosu ile Okul\_Tip\_Oto\_Id değeri ile ilişkilendirilir.

Gorev\_Tbl tablosunda ise görevlendirme bilgileri tutulmaktadır. Örneğin, Mevlut\_Dogan kullanıcısı Dekan olarak belirlendiğinde bu bilgiler Gorev\_Tbl tablosunda tutulmaktadır. Bu tablodaki alanlar aşağıda anlatılmıştır.

**Gorev\_Oto\_Id:** Her göreve verilen otomatik değerdir.

**Gorev\_UserId:** Görev verilen kullanıcının UserID'sidir.

**Gorev\_OkBlPr:** Verilen görev Okul, Bölüm, Program alanlarından hangisidir (Bkz. 5.3.3 Aynı Role Sahip Kullanıcılar Arasındaki Yetkilendirme).

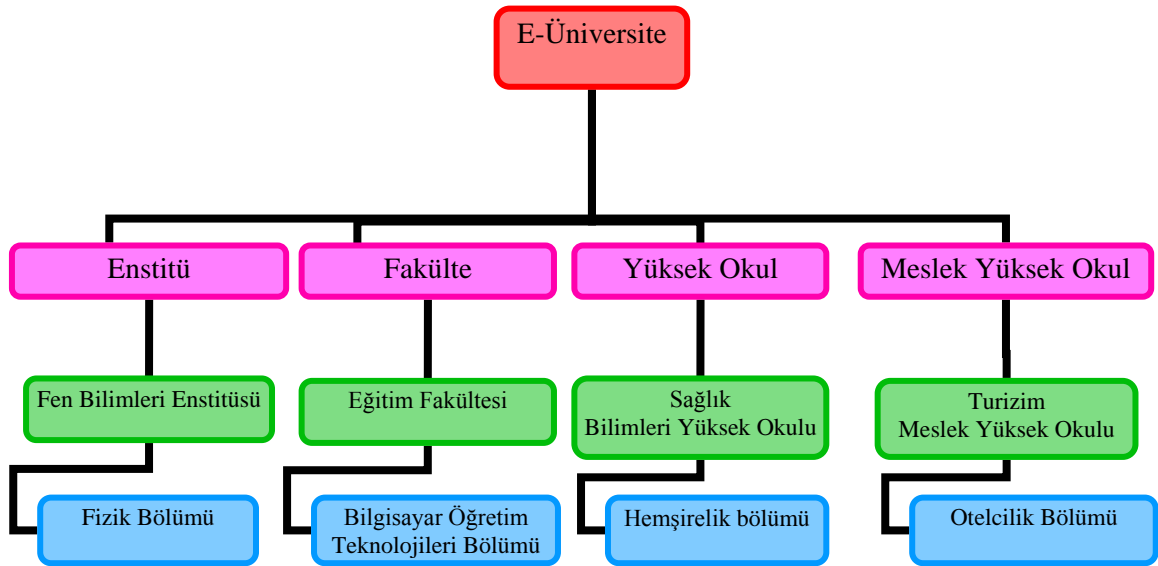
**Gorev\_Derece:** Verilen görev Okul için tanımlanmışsa “1”, Bölüm için tanımlanmışsa “2”, Program için tanımlanmışsa “3” değerini alır.

**Gorev\_AkPer:** Verilen görevin akademik mi, idari mi ? olduğunu tanımlar.

## 5.6 Bölüm Tanımlama Modülü

Bölüm Tanımlama Modülündeki yetki Dekan yada Müdür'e aittir. Tanımlanacak Bölüm Enstitü, Yüksek Okul, Meslek Yüksek Okuluna ait bir bölümse Müdür rolündeki kullanıcılar yapabilmektedir. Fakültede ise Dekan yetkilidir. Bir kullanıcının sadece dekan yada müdür rolüne sahip olması bu modülde bir işe yaramaz. Aynı zamanda kullanıcının hangi okula Müdür yada dekan olarak atandığı belirtilmelidir. Örneğin; **Fen Bilimleri Enstitüsü** Müdürü. Çünkü dekan yada müdürler yalnızca bağlı buldukları okula bölüm ekleyebilmektedirler. Okul tanımlama modülünde belirtildiği gibi Bölüm koduna rast gele bir değer verilemez.

Sistemdeki 4 okul tipinin altında da Bölüm oluşturulabilir. Şekil 5.57' de örnek bölümler tanımlanmıştır.



Şekil 5. 57 Okul organizasyon şeması.

Sistemde bölüm tanımlamak için yetkili bölümün admininin sisteme giriş yapması gerekmektedir. Daha sonra bölüm tanımlamayla ilgili bağlantı aracılığı ile bölüm tanımlama sayfasına ulaşır. Sırasıyla Bölüm kodunu girer, Bölüm adını girer ve ekle işlemini tetikler. Bölüm kodu sistemde daha önceden tanımlanmış ise kullanıcıya hata mesajı döndürülür, aksi halde yeni bölüm sisteme eklenir.

Şekil 5.58’da bölüm tanımlama sayfası verilmiştir.

Bolum Adı	<input type="text" value="Fizik Bolumu"/>
Bolum Kodu	<input type="text" value="1234"/>
<a href="#">Insert</a> <a href="#">Cancel</a>	








	<u>Bolum No</u>	<u>Bolum Kodu</u>	<u>Bolum Adı</u>	<u>Okul Adı</u>	
<a href="#">Select</a> <a href="#">Edit</a>	24	1234	Fizik Bolumu	Fen Bilimleri Enstitüsü	<a href="#">Delete</a>
<a href="#">Select</a> <a href="#">Edit</a>	25	12345	Bilgisayar Bolumu	Fen Bilimleri Enstitüsü	<a href="#">Delete</a>



Yeni Bolum

**Şekil 5. 58** Bölüm tanımlama sayfası.

Şekil 5.58’de de görüldüğü gibi sisteme eklenen bölüm tekrar düzenlenebilir, yada silinebilir. Silme işleminin gerçekleşmesi için silinecek bölümün altında bir program kayıtlı olmaması gerekmektedir. Sisteme eklenen bölüme Bölüm Başkanı yada Bölüm Sekreteri atama için öncelikle Bölüm Başkanı olacak kullanıcıyı sisteme eklemek gerekmektedir. Daha sonra ise listeden bölüm başkanı olacak kullanıcı seçilir ve Şekil 5.59’da verilen işlemler yapılır.

 KLASÖR YETKİLERİ  ERİŞİM KURALLARI  ROLLER  KULLANICI PROFİLLERİ  ROLLERİ DÜZENLE	<b>Roller ve Yetkiler</b>	
	Seçili Kullanıcı	Mevlut_Dogan
	Okul Tipini Belirtiniz	<input checked="" type="radio"/> Enstitü <input type="radio"/> Fakülte <input type="radio"/> Yüksek Okul <input type="radio"/> Meslek Yüksek Okul
	Akademik/İdari	<input type="radio"/> Akademik <input checked="" type="radio"/> İdari
	Gorev Alanı	Bolum İcin Gorev Tanımla ▼ Seçiniz Okul İcin Görev Tanımla Bolum İcin Görev Tanımla Program İcin Gorev Tanımla
	Okulları Seç	Fen Bilimleri Enstitüsü ▼
Bolum Seç	Fizik Bolumu ▼	
Program Seç		
	<b>Mevlut_Dogan Kullanıcısına Fizik Bolumu İcin Bölüm Başkanı Gorevi Verilsin mi?</b>	
		

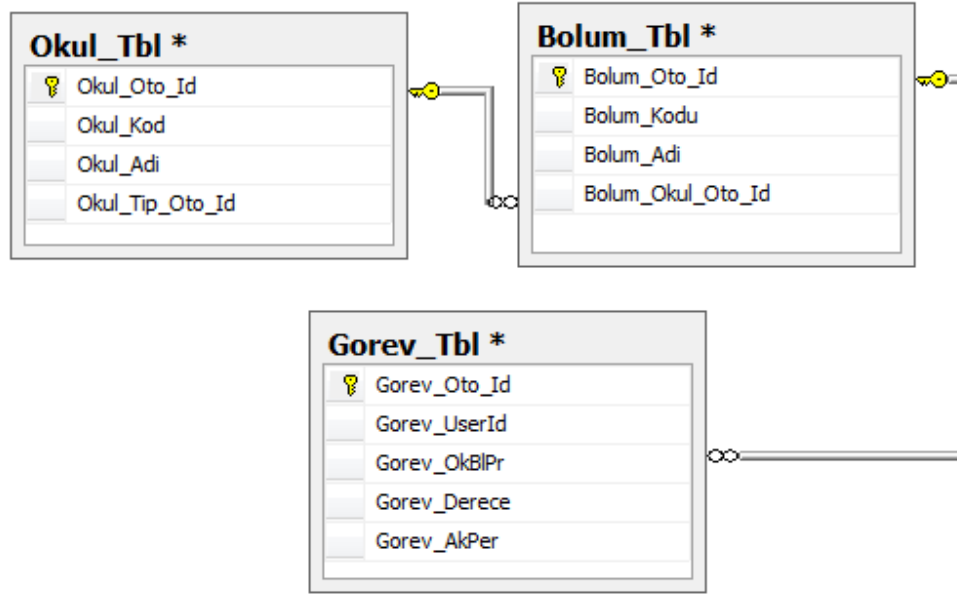
Şekil 5. 59 Bölüm Başkanı atama.

Şekil 5.59'daki işlemde sistemdeki bölüm başkanı kullanıcılarından Mevlut\_Dogan adlı kullanıcı seçilmiş ve idari görev olan Fizik Bölümü Bölüm Başkanı görevine atanmıştır.

Bu modülde yapılan işlemler Bolum\_Tbl tablosunda kayıt altına alınmaktadır. Bolum\_Tbl tablosu, Okul\_Tbl tablosu ve Gorev\_Tbl tablosunda Çizelge 5.15'deki gibi bir ilişkilendirme mevcuttur.



Çizelge 5. 14 Bolum\_Tbl tablosu ve Okul\_Tbl tablosu arasındaki ilişki.



Çizelge 5. 15' de verilen Bolum\_Tbl tablosundaki alanlar aşağıda açıklanmıştır. Gorev\_Tbl ve Okul\_Tbl tablosu, Çizelge 5.13 ile aynı olduğundan burada açıklanmamıştır.

**Bolum\_Oto\_Id:** Sisteme her yeni bölüm eklenmesinde otomatik olarak artan değerdir.

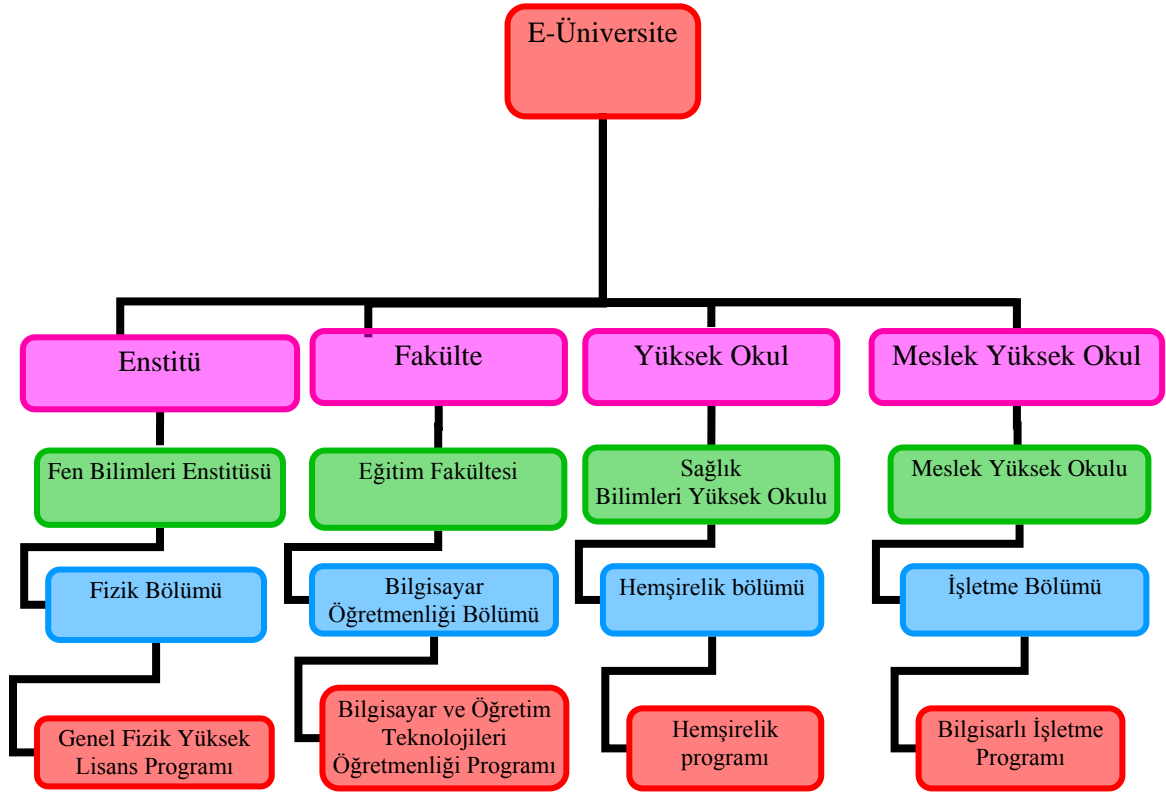
**Bolum\_Kodu:** Tanımlanan bölümün kodudur.

**Bolum\_Adi:** Tanımlanan bölüm adıdır.

**Bolum\_Okul\_Oto\_Id:** Bölümün kayıtlı olduğu okuldur. Bu alan Okul\_Tbl tablosu ile ilişkilendirilmiştir.

## 5.7 Eğitim Kataloğu Modülü

Eğitim Kataloğu Modülünde bölümlerin altına yeni programlar eklenmektedir. Program açma ve kapama yetkisi bölüm sekreterine aittir. Bölüm sekreteri Program eklerken kendisinden 5 bilginin girilmesi istenmektedir. Program Kodu, Program Adı, Programla ilgili Detay, Programın Aktif mi/ Pasif mi olacağı ve son olarak da programın katalogta görünüp görünmeyeceğini belirtmesi gerekmektedir. Hangi bölüm sekreterinin hangi bölüme program ekleme çıkarma işlemi yapabileceği rol ve yetkiler bölümünde ayrıntılı olarak anlatılmıştır. Şekil 5.60'da ise örnek bir Okul, Bölüm ve Program organizasyon şeması verilmiştir.



Şekil 5. 60 Örnek Okul, Bölüm ve Program organizasyonu.

Sistemde Toplam 12 adet dönem bulunmaktadır ve bu dönemler sabittir. Açılan program kaç dönemden oluşacaksa okadar dönem sisteme eklenebilir. Örneğin tezli bir yüksek lisans programı toplam 4 dönemden oluşmaktadır. O halde sisteme 1.Sınıf Güz dönemi, 1. Sınıf Bahar Dönemi, 2.Sınıf Güz Dönemi, 2. Sınıf Bahar Dönemi programa

eklenmelidir. 4 yıllık programlar için sisteme 8 dönem, 4+Hazırlık programları için 10 ve Tıp fakülteleri için toplam 12 dönem sisteme eklenmelidir. Bu 12 dönem ise aşağıda verilmiştir.

- 1) Hazırlık Güz Dönemi
- 2) Hazırlık Bahar Dönemi
- 3) 1.Sınıf Bahar Dönemi
- 4) 1.Sınıf Güz Dönemi
- 5) 2.Sınıf Bahar Dönemi
- 6) 2.Sınıf Güz Dönemi
- 7) 3.Sınıf Bahar Dönemi
- 8) 3.Sınıf Güz Dönemi
- 9) 4.Sınıf Bahar Dönemi
- 10) 4.Sınıf Güz Dönemi
- 11) 5.Sınıf Bahar Dönemi
- 12) 5.Sınıf Güz Dönemi

Sisteme 12 adet dönem eklenmesi ile tüm ihtimaller ve olabilecek durumlar sisteme dahil edilebilecektir. dönemlik bir tıp fakültesinin neden eklenebilir. Bu sistemin oluşturulmasının nedeni uzaktan eğitimin üniversite içerisinde de local olarak verilebilmesi içindir. Örneğin Tıp Fakültesinde derslere giren bazı öğretim üyeleri dersi uzaktan eğitimle vermek isteyebilirler. Araştırma Teknikleri dersi gibi veya seçmeli Bilgisayar dersi gibi. Böyle bir durumda Fakülte, Yüksek Okul veya Meslek Yüksek Okul gibi birimlerinde sistemde bulunması gerekmektedir. Sisteme yeni bir bölüm tanımlanması gerekiyorsa ilgili okuldaki dekan yada müdür bu işlemi yapabilecektir veya program eklenecekse programın bağlı olacağı bölümdeki Bölüm Sekreterliği tarafından programın açılması gerekmektedir.

Sisteme yeni program eklemek için aşağıdaki işlemlerin takip edilmesi gerekmektedir.

Öncelikle ilgili bölümün sekreteri sisteme giriş yapar. Programlar ile ilgili bağlantıyı kullanarak program tanımlama ile ilgili sayfaya ulaşır.

1. Program kodu bilgisini girer.
2. Program adı bilgisini girer.
3. Programın açıklama bilgisini girer.
4. Programın açılacağı bölüm bilgisini seçer.
5. Programın başvurulabilir olup/olmama bilgisini belirler.
6. Programın katalogda görünüp/görünmeme bilgisini belirler.

Program ekleme işlemini tetiklendikten sonra program kodu bilgileri sistemde daha önceden varsa hata bilgisi döndürülür, yoksa eklenir. Bu işlem Şekil 5.61’de verilmiştir.

**Bolum Seç** Bilgisayar Bolumu

Bolum No	23
Program Kodu	19
Program Adı	Bilgisayar Kontrol Sistemi
Program Detay	Bilgisayar Bolumu Yuksek Lisans Programlari Arasinda Yer Almaktadır.
Program Aktif	<input type="radio"/> Aktif Yap <input checked="" type="radio"/> Pasif Yap
Program Katalog Durum	<input type="radio"/> Sistemde Göster <input checked="" type="radio"/> Sistemde Gizle
<a href="#">Insert</a> <a href="#">Cancel</a>	
<a href="#">Delete</a>	

	Program No	Program Adı	Program Kodu	Visible	Aktif	
<a href="#">Select</a>	19	Bilgisayar Kontrol Sistemleri	BK100	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="#">Delete</a>

Şekil 5. 61 Program ekleme işlemi.

Şekil 5.61’de de görüldüğü üzere tanımlanan program tekrar düzenlenebilmekte yada silinebilmektedir. Silme işlemi için bu programa kayıtlı öğrencinin olmaması gerekmektedir. Programın katalogda görünüp görünmeme özelliği ile aktif olan bir program başvurulacak programlar arasında gizlenebilir yada program yeni eklenmişse şimdilik başvurulabilir özelliği gizle moduna getirilebilir. Böylece bu programla ilgili ders ekleme ve dönem ekleme işlemleri ve ders paketleri yüklendikten sonra gerektiği zaman aktif hale getirebilmek için bekletilir. Programa daha önceki dönemlerde öğrenci alınmış ve bu dönemde öğrenci alınmaması gerekiyorsa katalogda görünme özelliği yine gizlenebilir. Program sistemde hiç kullanılmıyorsa ve bu programa kayıtlı hiç öğrenci yoksa program tamamen pasif hale getirilebilir. Sistemdeki programa dönem eklemek için ise önce dönem eklenecek program seçilir. Sonra dönem no, o dönemde alabileceği minimum kredi ve maksimum kredi bilgisini Zorunlu/Seçmeli ders sayısını girer ve dönem ekle işlemini tetikler. Dönem no bilgileri sistemde daha önce tanımlı ise hata bilgisi döndürülür, değilse dönem programa eklenir. Programa dönem ekleme işlemi Şekil 5.62’de verilmiştir.

**Bolum** Bilgisayar Bolumu

**Program** Bilgisayar Kontrol Sistemleri

**DONEMLER**

Donem Oto Id	Donem Adi	Min Kredi	Max Kredi	Ders Saiyisi	Ekle
15	5. Sınıf Guz Dönemi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
16	5. Sınıf Bahar Dönemi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	Hazırlık Guz Donemi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	Hazırlık Bahar Donemi	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>



### PROGRAMA KAYITLI DONEMLER

No	Program Adı	Donem Adı	Min Kredi	Max Kredi	Ders Sayısı	Çıkar
64	Bilgisayar Kontrol Sistemleri	1.Sınıf Guz Donemi	2	2	2	<input type="checkbox"/>
65	Bilgisayar Kontrol Sistemleri	1.Sınıf Bahar Dönemi	2	2	2	<input type="checkbox"/>
66	Bilgisayar Kontrol Sistemleri	2. Sınıf Guz Dönemi	2	2	2	<input type="checkbox"/>
67	Bilgisayar Kontrol Sistemleri	2. Sınıf Bahar Dönemi	2	2	2	<input type="checkbox"/>
69	Bilgisayar Kontrol Sistemleri	3. Sınıf Bahar Dönemi	2	2	2	<input type="checkbox"/>
70	Bilgisayar Kontrol Sistemleri	4. Sınıf Guz Dönemi	2	2	2	<input type="checkbox"/>
71	Bilgisayar Kontrol Sistemleri	4. Sınıf Bahar Dönemi	2	2	2	<input type="checkbox"/>
68	Bilgisayar Kontrol Sistemleri	3. Sınıf Guz Dönemi	2	2	2	<input type="checkbox"/>



Şekil 5. 62 Programa dönem ekleme işlemi.

Sistemdeki döneme ders ekleme işlemini yine bölüm sekreteri yapabilmektedir. Ders ekleme işlemini yapabilmek için önce işlem yapılacak program seçilir. Sonra dersin ekleneceği dönem seçilir ve programın kayıtlı olduğu bölümde var olan dersler görüntülenir (Bölüme ders ekleme işlemi 5.8 Dersler Modülünde anlatılmıştır). Bu derslerden hangileri işlem yapılacak dönemde alınması gerekiyorsa, seçilir ve o dönemde alınmak üzere ders kaydedilir. Ancak seçili döneme aktarılacak ders bilgilerinin girilmesi gerekmektedir. Bunlar;

- **Kredi bilgisi:** Dersin etki yapacağı kredi miktarının girilmesi gerekmektedir.
- **Açık/Kapalı:** Dersin aktif yada pasif mi olacağı durumunun belirlenmesi gerekir.
- **Zorunlu/Seçmeli/Genel durumu:** Zorunlu ders mi, seçmeli dersmi yada genel ders mi olduğu belirtilmelidir. Zorunlu yada seçmeli bir ders ise dersin ekleneceği gurup belirtilmelidir. Bu işlemin amacı öğrencinin ilişkili olduğu ders gurubu içerisinde yalnızca birini seçmesi gerekiyorsa yada en az ikisi gibi belirli bir ders seçmesi gerekiyorsa, bu guruptaki derslerden istediğini yada istediği iki tanesini seçebilmesine fırsat vermek ve bu işlemi takip edebilmektir. Örneğin seçmeli ders olan bilgisayar yada resim derslerinin ikisinde aynı gurup içinde olsun. Bu derslerden yalnızca birinin seçilmesi gerekiyorsa sistem bu guruptaki ikinci bir dersi almasına imkan sağlamayacaktır. Yada bilgisayar dersleri gibi seçmesi

zorunlu olan bir guruptan öğrencinin en az 3 tane ders seçmesi gerekiyorsa ve 5 tane ders varsa bu işlemin doğru yapıp yapılmadığının takibini yapabilmek için ders gurubu bilgisi girilmesi gerekmektedir.

- **Görünüp/Görünmeme durumu:** Dersin katalogta görülüp görülmeyeceği durumu belirlenmelidir.
- **Ön koşulu bilgisini:** Bu dersin alınması için daha önce bazı derslerden başarılı olma şartı varsa o dersler belirtilmelidir.

Bu bilgiler girildikten sonra ders ekle işlemini tetiklenir ve ders belirtilen döneme eklenir. Programa ders ekleme işlemi Şekil 5.63’de verilmiştir.

Bolum	Bilgisayar Bolumu
Program	Bilgisayar Kontrol Sistemleri
Dönem	1.Sınıf Guz Donemi

### BOLUM DERSLERI

Ders Oto Id	Ders Adı	Ders Kod	
8	Algoritma Tasarımı	2113	Select

### BOLUM DERSLERI

Kredi Notu	2
Aktif/Pasif	<input checked="" type="radio"/> Aktif <input type="radio"/> Pasif
Zorunlu/Secmeli	<input type="radio"/> Zorunlu <input type="radio"/> Seçmeli <input checked="" type="radio"/> <b>Bilgisayar Gurubu</b> <input type="radio"/> Genel
Visible	<input checked="" type="radio"/> True <input type="radio"/> False
On Kosul	Algoritmaya Gris Dersinden Basarılı olmak gerekir
Gurup	15
Insert Cancel	



### PROGRAMA KAYITLI DONEMLER

PDDT Oto Id	Donem Adı	Ders Adı	Ders Kod	Çıkar
6	1.Smf Guz Donemi	Arastirma Teknikleri	12	<input type="checkbox"/>
7	1.Smf Guz Donemi	Fizik	123	<input type="checkbox"/>



Şekil 5. 63 Döneme ders ekleme.



Döneme Zorunlu/Seçmeli ders grubu ekleme işleminin amacı öğrencinin ilişkili olduğu ders gurubu içerisinde yalnızca birini seçmesi gerekiyorsa yada en az ikisi gibi belirli bir ders seçmesi gerekiyorsa bu gruptaki derslerden istediğini yada istediği iki tanesini seçebilmesine fırsat vermek ve bu işlemi takip etmek gerekir. Bu işlemi yapabilmek için aşağıdaki işlemlerin sırasıyla yapılması gerekmektedir.

1. Program seçilir.
2. Dönem seçilir.
3. Ders grubunun ekleneceği dönem seçilir.
4. Grup kodu bilgisi girilir.
5. Grup adı bilgisi girilir.
6. Grup açıklama bilgisi girilir.
7. Grupta seçilecek ders sayısı bilgisi girilir.
8. Grup ekleme işlemi girilir.

Bu işlemler yapıldıktan sonra Zorunlu/Seçmeli ders grubu döneme eklenir. Döneme Zorunlu ve Seçmeli Ders Gurubu Ekleme İşlemi Şekil 5.64’de verilmiştir.

Bohum Seç

Program

Donem



Gurup Kodu

Gurup Adi

Gurup Detay

Gurup Adeti

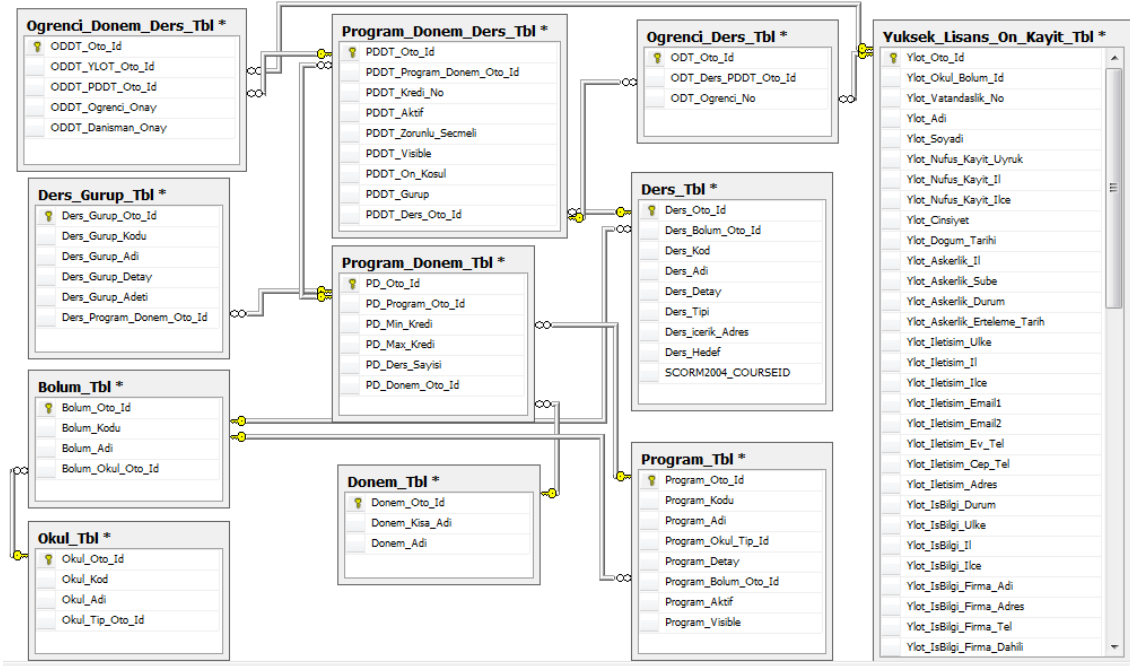
[Insert](#) [Cancel](#)

	<u>Ders Gurup Oto Id</u>	<u>Ders Gurup Kodu</u>	<u>Ders Gurup Adi</u>	<u>Ders Gurup Adeti</u>	
<u>Düzenle</u>	15	123	Bilgisayar Gurubu	12	<u>Delete</u>

Şekil 5. 64 Döneme zorunlu seçmeli ders gurubu ekleme.

Bu modüldeki verilerin kayıtlı olduğu ve ilişkili olduğu alanlar Çizelge 5. 17’de verilmiştir.

Çizelge 5. 15 Eğitim Kataloğu verilerinin tutulduğu ve bu tabloların ilişkili olduğu tablolar.



Sisteme eklenen programlar Program\_Tbl tablosunda tutulmaktadır. Daha önce belirtilen sabit 12 dönem Donem\_Tbl tablosunda, programlara eklenen dönemler ise Program\_Donem\_Tbl tablosunda tutulmaktadır. Dönemlere eklenen dersler Program\_Donem\_Ders\_Tbl tablosunda, Zorunlu ve seçmeli ders gurubları ise Ders\_Gurup\_Tbl tablosunda tutulmaktadır. Aşağıda bu tablolarda yer alan alanların açıklaması verilmiştir.

### Program\_Tbl Tablosu

- **Program\_Oto\_Id:** Her yeni eklenen programa veri tabanı tarafından verilen otomatik ID'dir.
- **Program\_Kodu:** Tanımlanan programın kodudur.
- **Program\_Adi:** Tanımlanan programın adıdır.
- **Program\_Detay:** Programla ilgili açıklama bilgisidir.
- **Program\_Bolum\_Oto\_Id:** Programın eklendiği bölümün ID'sidir. Bolun\_Tbl tablosu ile ilişkilidir.

- **Program\_Aktif:** Programın Aktif yada pasiflik durumudur. Aktifse “1”, Pasifse “0” deęerini alır.
- **Program\_Visible:** Programın katalogda grlp grlmeyeceęi bilgisidir. Visible zellięi true ise “1”, False ise “0” deęerini alır.

#### **Donem\_Tbl Tablosu**

- **Donem\_Oto\_Id:** Dnem ID’sidir.
- **Donem\_Kisa\_Adi:** Dnemin kısa adıdır (rn. Arařtırma Teknikleri ve Bilimsel İletiřimin Temelleri dersi iin ‘ATBİT’).
- **Donem\_Adi:** Dnemin Adıdır.

#### **Program\_Donem\_Tbl Tablosu**

- **PD\_Oto\_Id:** Otomatik artan ID’dir.
- **PD\_Program\_Oto\_Id:** Donemin eklendięi programın ID’sidir.
- **PD\_Min\_Kredi:** Donemde alınabilecek minimum kredidir.
- **PD\_Max\_Kredi:** Donemde alınabilecek maksimum kredidir.
- **PD\_Ders\_Sayisi:** Donemde alınabilecek maximum ders sayısıdır.
- **PD\_Donem\_Oto\_Id:** Donem\_Id sidir. Donem\_Tbl tablosu ile iliřkilidir.

#### **Program\_Donem\_Ders\_Tbl**

- **PDDT\_Oto\_Id:** Otomatik ID’dir.
- **PDDT\_Program\_Donem\_Oto\_Id:** dersin eklendięi dnemim ID’sidir. Program\_Donem\_Tbl ile iliřkilidir.
- **PDDT\_Kredi\_No:** Dersin kredisidir.
- **PDDT\_Aktif:** Dersin Aktif/Pasif durumudur.
- **PDDT\_Zorunlu\_Secmeli:** Zorunlu/Secmeli ders gurubu ID’sidir.
- **PDDT\_Visible:** Programın katalogda grlp grlmeyeceęi durumudur.

- **PDDT\_On\_Kosul:** Dersin alınabilmesi için gerekli ön şart bilgisidir.
- **PDDT\_Gurup:** Dersin gurub ID'sidir. Gurup\_Tbl ile ilişkilidir.
- **PDDT\_Ders\_Oto\_Id:** Ders ID'sidir. Ders\_Tbl tablosu ile ilişkilidir.

### **Ders\_Gurup\_Tbl**

- **Ders\_Gurup\_Oto\_Id:** Otomatik verilen sıralama ID'sidir.
- **Ders\_Gurup\_Kodu:** Gurup kodudur.
- **Ders\_Gurup\_Adi:** Gurup adıdır.
- **Ders\_Gurup\_Detay:** Gurupla ilgili detaylı açıklama bilgisidir.
- **Ders\_Gurup\_Âdeti:** Gurup adetidir.
- **Ders\_Program\_Donem\_Oto\_Id:** Gurubun eklendiği dönem ID'sidir. Program\_Donem\_Tbl ile ilişkilidir.

## 5.8 Dersler Modülü

Dersler modülünde sistemdeki bölümlere yeni ders ekleme işlemi yapılmaktadır. Bu işlem bölüm sekreteri tarafından yapılmaktadır. Bölüm sekreterinin bu işlemde aktif olmasının sebebi, her bölüm, bölümünde okutacağı derslerin havuzunu oluşturur. Programlar bölüme, dönemler programa, dersler ise döneme eklenir. Dönemlerde okutulacak dersler bölüm havuzunda oluşturulan derslerden alınır. Örneğin; Bilgisayar I dersi bu bölümde okutulacak bütün programlarda ortaktır. Sisteme eklenen derslerin içeriği olan SCORM ders paketleri yine bu modülden sisteme eklenmektedir. Ayrıca derslere alt konu ekleme, kaynakça ekleme, sınav ekleme işlemleri de yapılabilmektedir.


### 5.8.1 Sisteme Yeni Ders Ekleme

Sisteme yeni bir ders eklemek için öncelikle Bölüm Sekreterinin sisteme giriş yapması gerekir. Daha sonra dersler ile ilgili bağlantıyı kullanarak dersler sayfasına ulaşılır. Ders ekleyeceği bölümü belirtmesi gerekmektedir. Bölüm seçildikten sonra bölüm içerisindeki varolan dersler listelenir. Ekeleyeceği dersle ilgili aşağıdaki bilgileri girmesi gerekmektedir.

- Ders adı bilgisini girer.
- Ders kodu bilgisini girer.
- Ders tanımını bilgisini girer.
- Ders tipi bilgisini girer (SCORM).
- Ders hedef bilgisini girer.

Kaydet işlemini tetiklendikten sonra ders kodu daha önceden tanımlı ise kullanıcıya hata bilgisi döndürülür, değilse ders tanımlanır. Ders tipi bilgisi olarak şimdilik LMS yada SCORM girilmesi gerekmektedir. Ancak ileride AICC Standartlarımda kabul eden bir sistem düşünüldüğü için bu alanın belirtilmesi gerekmektedir. Örnek ders ekleme işlemi Şekil 5.65' de verilmiştir.

Bolum Seç Bilgisayar Bolumu

  
Yeni Ders  
Ekle

Bolum Adi	
Ders Adi	Araştırma Teknikleri 2
Ders Kod	6
Ders Detay	Bilgisayar Kontrol Sistemleri Yüksek lisans dersidir. Bu dersi alabilmek için Arastirma Teknikleri 1 dersinden başarılı olmak gerekir.
Ders Tipi	LMS
Ders icerik Adres	-
Ders Hedef	Arastirma Becerisi Kazandırmak
Ekle Cancel	

	<u>Ders No</u>	<u>Ders Adi</u>	<u>Bolum Adi</u>	Scorm 1.2 İçerik Ekle	Scorm 2004 İçerik Ekle	
<u>Seç</u>	6	Arastirma Teknikleri	Bilgisayar Bohumu	<a href="#">Ekle</a>	<a href="#">Ekle</a>	<a href="#">Sil</a>
<u>Seç</u>	7	Fizik	Bilgisayar Bohumu	<a href="#">Ekle</a>	<a href="#">Ekle</a>	<a href="#">Sil</a>
<u>Seç</u>	8	Algoritma Tasarımı	Bilgisayar Bohumu	<a href="#">Ekle</a>	<a href="#">Ekle</a>	<a href="#">Sil</a>

Şekil 5. 65 Programa yeni ders ekleme.

Ders ekleme işlemi tamamlandıktan sonra derse ait SCORM paketinin eklenmesi gerekmektedir. Hazırlanan Uzaktan Öğrenim Yönetim Sistemi hem SCOM 1.2 Uyumlu içeriği, hem de SCORM 2004 uyumlu içeriği kabul etmektedir. Eklenecek SCORM paketinin türü hangisi ise ilgili linke tıklanılır ve o derse ait SCORM paketi eklenir. Örnek olarak SCORM 2004 paketinin eklenmesi Şekil 5.66’da verilmiştir.

### Ders Paketi Yükle

#### Ders dosyası yüklendi

Ders ile ilişkilendirmek istediğiniz SCORM paketini (.zip, .pif) seçip GÖNDER düğmesine tıklayınız.

SCORM Paketi:  Gözet...

GÖNDER

Şekil 5. 66 SCORM 2004 paketinin sisteme eklenmesi.

SCORM paketinin LMS’ e eklendikten ve öğrencinin ders kayıt işlemleri yapıldıktan sonra öğrenci ekranında görüntülenmesi Şekil 5.67’de verilmiştir.



Şekil 5. 67 SCORM 2004 paketin öğrenci ekranında görüntülenmesi (Hakkari 2009).



## 5.8.2 Dersin İçerik Ağacının Oluşturulması

Hazırlanan öğrenim yönetim sistemi şimdilik ders paketlerini bütün halinde kabul etmektedir. Ancak ileriki aşamalarda alt konular içinde ders paketlerinin ayrı şekilde eklenebilmesi hedeflenmektedir. Bu amaçla her dersin içerik ağacını oluşturmak için bir aspx sayfası oluşturulmuştur. Bu sayfa ile derslerin alt ana başlıkları ve dersin sırası belirlenebilmektedir. Böylece ileriki aşamalarda dersi veren öğretim üyelerinin isteği doğrultusunda alt konulardaki SCORM paketleri dinamik hale gelebilecek ve bir ders için tek tip içerik yerine ders içeriği farklılaşabilen bir yapı oluşturulacaktır. Şekil 5.68’de alt konu ekleme işlemi verilmiştir.

Bölüm Seç	Bilgisayar Bölümü
Ders Seç	Arastirma Teknikleri

Sıra No	5
Konu Adı	Hipotez
Konu Adres	http://localhost/Akuelearning/SCORM_;
Konu Aktif	<input checked="" type="radio"/> Aktif <input type="radio"/> Pasif
<a href="#">Insert</a> <a href="#">Cancel</a>	



Konu Ekle

<u>Konu No</u>	<u>Sıra No</u>	<u>Ders Adı</u>	<u>Konu Adı</u>	<u>Aktif</u>	<u>Select</u>
6	5	Arastirma Teknikleri	Hipotez	True	<a href="#">Select</a>

Şekil 5. 68 Alt konu ekleme işlemi.

### 5.8.3 Dersin Sınavlarını belirlemek

Eklenen her ders için kaç sınav olacağı bölüm sekreteri tarafından belirlenmektedir. Hazırlanan uzaktan öğrenim yönetim sisteminde sınavlar SCORM uyumlu programlar aracılığı ile hazırlanmakta ve sisteme sınav modülünden eklenmektedir. Bu işlemler 5.13 ve 5.14’de ayrıntılı olarak anlatılmıştır. Derslere sınav eklemek için aşağıdaki işlemlerin takip edilmesi gerekmektedir.

- Öncelikle sınavın tipi seçilir. Sınavın vize mi yada final sınavı mı olduğu belirtilmelidir.
- Sınavın yıl sonu notunun yüzde kaçını oluşturacağı girilir.
- Sınav tipi final ise DERS ADI + final sınavı şeklinde, sınav tipi vize ise Vize + Numarası şeklinde otomatik olarak verilir.

Bu işlemlerin ardından sınav ekle işlemini tetiklenir. Sınav tipi final ise daha önce final eklenmemişse sınav ilgili derse eklenir, eklenmişse hata mesajı döndürülür. Şekil 5.69’da sınav ekleme işlemi verilmiştir.

Bölüm Seç	Bilgisayar Bölümü
Ders Seç	Araştırma Teknikleri
Sınav Tipi	<input checked="" type="radio"/> Vize <input type="radio"/> Final Vize No 1
Sınav Adı	Vize1
Sınav Yüzdeleri	10
<u>Insert</u> <u>Cancel</u>	

Şekil 5. 69 Sınav ekleme işlemi.

#### 5.8.4 Ödev eklemek

Hazırlanan uzaktan öğrenim yönetim sistemi aracılığı derslerle ilgili dönem ödevleri verilebilmektedir. Bu ödevler yılsonu notuna katkı yapacak ödevlerdir. Dönem ödevi şeklindedir. Bu bölümdeki admin ise bölüm sekreteridir. Bölüm sekreterinin siteme yeni bir ödev ekleyebilmesi için aşağıdaki işlemleri takip etmesi gerekmektedir.

- Ödevin yıl sonu notunun yüzde kaçını oluşturacağını girmesi gerekmektedir.
- Ödev adını girmesi gerekmektedir.

Bu bilgileri girildikten sonra ödev ekle işlemini tetiklenmesi gerekmektedir. Böylece ödev, ilgili dersin ödevleri arasına eklenmektedir. Veilen ödevler dönem içerisinde silinebilmekte yada düzenlenebilmektedir. Ancak silme işlemi için ödevin kesinlikle yayında olmaması gerekmektedir. Şekil 5.70’de ödev ekleme işlemi verilmiştir.

Bölüm Seç

Ders Seç



Odev Ekle

<u>Odev No</u>	<u>Odev Adı</u>	<u>Odev Yuzdelik</u>	
118	Araştırmada Bilimsel Yöntem Nedir?	10	<a href="#">Seç</a>
119	Bilimsel Yöntem Nedir	5	<a href="#">Seç</a>
120	Nicel Gözlem Nedir	5	<a href="#">Seç</a>

Şekil 5. 70 Odev ekleme işlemi.

### 5.8.5 Ders Kaynak Bilgilerini Ekleme

Öğrencilere derslerinde yardımcı olabilmesi için dersin hocaları tarafından bazı kaynakların verilmesi öğrencilere faydalı olacaktır. Bunun için önemli kaynakların paylaşılmasını sağlayan ders kaynak sayfası oluşturulmuştur. Dersin hocaları tarafından bölüm sekreterine bildirilen dersle ilgili yararlı kaynaklar, bölüm sekreteri tarafından sisteme girilmektedir. Bu işlemi gerçekleştirebilmek için bölüm sekreterinin öncelikle sisteme giriş yapması gerekmektedir. Daha sonra ilgili bağlantı aracılığı ile ders kaynak ekleme sayfasına ulaşır. Önce eklemek istediği kaynak kitap adını girer. Daha sonra kaynak yayınevi bilgisini girer. Kaynak bilgileri ekle işlemini tetikler. Belirtilen ada sahip kaynak daha önce eklenmemişse ders kaynakları arasına eklenir, zaten mevcutsa hata mesajı döndürülür. Şekil 5.71’de örnek ders kaynak ekleme işlemi verilmiştir.

Bölüm Seç

Ders Seç

Kitap Adı	<input type="text" value="Nicel - Nitel Araştırma Teknikleri"/>
Yayın Evi	<input type="text" value="Elif Kuş/ Anı Yayıncılık..."/>
<a href="#">Insert</a> <a href="#">Cancel</a>	



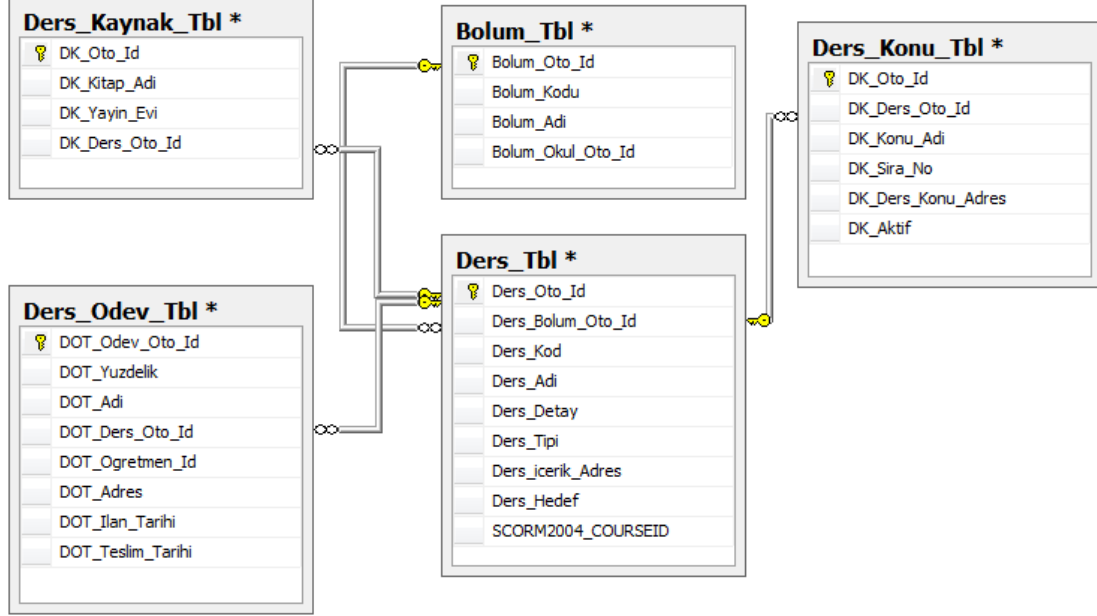
Kaynak Ekle

Kaynak No	Kitap Adı	Yayın Evi	Seç
1	Nicel - Nitel Araştırma Teknikleri	Elif Kus/ Anı Yayıncılık	<a href="#">Select</a>

Şekil 5. 71 Ders kaynak bilgileri ekleme işlemi.

Dersler modülünde yapılan işlemler Çizelge 5.18’de verilen tablolarda tutulmaktadır.

**Çizelge 5. 16** Dersler tablosu ve ilişkili veri tabloları.



Çizelge 5.18’de verilen dersler tablosu ve ilişkili veri tabloları aşağıda verilmiştir.

### Ders\_Tbl Tablosu

- **Ders\_Oto\_Id:** Eklenen dersin ID’sidir.
- **Ders\_Bolum\_Oto\_Id:** Dersin eklendiği bölümün ID’sidir. Bolum\_Tbl tablosu ile ilişkilidir.
- **Ders\_Kod:** Tanımlanan dersin kodudur.
- **Ders\_Adi:** Tanımlanan dersin adıdır.
- **Ders\_Detay:** Ders ile ilgili detaylı bilgidir.
- **Ders\_Tipi:** Dersin tipidir (SCORM, AICC vb.)
- **Ders\_icerik\_Adres:** Dersin eklendiği içeriğin adres bilgisidir.
- **Ders\_Hedef:** Dersin hedefidir.
- **SCORM2004\_COURSEID:** Eklenen SCORM 2004 paketinin ID’sidir.

### **Ders\_Konu\_Tbl Tablosu**

- **DK\_Oto\_Id:** Tanımlanan konun ID'sidir.
- **DK\_Ders\_Oto\_Id:** Tanımlanan konunun ders ID'sidir. Ders\_Tbl tablosu ile ilişkilidir.
- **DK\_Konu\_Adi:** Tanımlanan konunun adıdır.
- **DK\_Sira\_No:** Dersin içerik sırasıdır.
- **DK\_Ders\_Konu\_Adres:** Konu ile ilgili eklenen ders paketinin adresidir.
- **DK\_Aktif:** Dersin aktif yada pasif durumunu bildirir. Aktifse "1", Pasifse "0" dır.

### **Ders\_Odev\_Tbl Tablosu**

- **DOT\_Odev\_Oto\_Id:** Eklenen ödevin ID'sidir.
- **DOT\_Yuzdelik:** Ödevin başarı notuna yaptığı yüzdelik katkısıdır.
- **DOT\_Adi:** Ödevin adıdır.
- **DOT\_Ders\_Oto\_Id:** Verilen ödevin ait olduğu dersdir.
- **DOT\_Ogretmen\_Id:** Ödevi veren hocanın ID'sidir (Ödevler modülünde ödevi veren öğretim elemanının UserID'sidir).
- **DOT\_Adres:** Ödev word dosyası yada zip dosyası gibi dosyalarla birlikte verildiyse bu ek dosyanın adresidir.
- **DOT\_Ilan\_Tarihi:** Ödevin ilan tarihidir (Ödevler modülünde öğretim elemanı belirleyecektir).
- **DOT\_Teslim\_Tarihi:** Ödevin teslim tarihidir (Ödevler modülünde öğretim elemanı belirleyecektir).

### **Ders\_Kaynak\_Tbl Tablosu**

- **DK\_Oto\_Id:** Tanımlanan kaynağın ID'sidir.
- **DK\_Kitap\_Adi:** Tanımlanan kitabın adresidir.
- **DK\_Yayin\_Evi:** Tanımlanan kitabın yayın evidir.
- **DK\_Ders\_Oto\_Id:** Kaynak kitabın ders ID'sidir. Ders\_Tbl tablosu ile ilişkilidir.

## 5.9 Haftalık Ders Programı Modülü

1. modül olan kullanıcı ve kullanıcı gurupları modülünde öğretim üyeleri bölümlere kaydedilmişti. Bu modülde ise derse girecek öğretim elemanları ve ders programı belirlenmektedir. Ders programını öğretim elemanlarının kendisi belirlemektedir. Ancak bölüm sekreteri gerektiği takdirde bu program saatini ve gününü değiştirebilmektedir. Öğretim elemanları ders programını belirleyebilmesi için o dersin öğretim görevlisi olarak atanmış olması gerekmektedir. Bir ders için yalnızca tek bir öğretim elemanının görevli olması şart değildir. Bir dersi verebilecek birden fazla öğretim elemanı belirlenebilir. Öğretim elemanları kendi ders programının yoğunluğuna göre diğer derslerin programını belirleyebilir. Yalnız belirlenen program bölüm sekreterinden onay almadan aktif olamaz. Dersler için öğretim elemanları tanımladıktan sonra, öğretim elemanları sisteme giriş yaptıklarında verebilecekleri dersler ekranda listelenir. Öğretim elemanı bu derslerden vermek istediği dersi seçer ve o dersi verebileceği uygun gün ve tarihi belirler. Öğretim elemanlarının hepsi istediği kadar program tarihi ve saati verebilir. Bölüm sekreteri, genel programın uygunluğuna göre o derse girecek öğretim elemanını belirler.

Öğretim elemanları için böyle bir program belirlenmesinin amacı interaktif bir uzaktan eğitim hizmeti vermektir. Bu sayede öğrenciler derslerde takıldıkları noktaları video konferans yada mesajlaşma yoluyla giderilebilirler. Bu işlemler sanal sınıf modülünde uygulanabilmektedir. Öğretim elemanlarının gireceği derslerin belirlenmesine örnek Şekil 5.72’de verilmiştir.

## İŞLEM YAPMAK İSTEDİĞİNİZ DERS

Bohum Seç

Ders Seç

## Seçilen Öğretim Üyeleri Bu Dersi Verebilir.

Okulları Seç

Bohum Seç

- Seçilen Bölümdeki Öğretim Üyelerini Listele
- Dersi Veren Diğer Öğretim Üyelerini Görüntüle

Personel No	Sicil No	Adı	Soyadı	Aktif	Seç
10	121	Mevlut	Dogan	True	<input checked="" type="checkbox"/>
11	1q2w	Omer	Sise	True	<input type="checkbox"/>



Şekil 5. 72 Derse girebilecek öğretim elemanlarının belirlenmesi.

Şekil 5.72’de de görüldüğü üzere bilgisayar bölümünde verilen araştırma teknikleri dersini verebilecek öğretim elemanları belirlenmiştir. Burada örnek olarak Mevlut Dogan ve Omer Sise adlı iki öğretim elemanının bu dersi verebilmesine onay verilmiştir. Ancak dikkat edilirse ders için sadece dersin bağlı bulunduğu bölümdeki bir öğretim elemanının bu dersi vermesi zorunlu değildir. Diğer Enstitü yada Fakültelerde görevli öğretim elemanları da bu dersi verebilir. Bu işleyişe örnek olarak Bilgisayar dersi verilebilir. Örneğin Fen Edebiyat Fakültesi Fizik Bölümündeki öğrencilerin alabildiği bu dersi, Eğitim Fakültesi Bilgisayar Öğretim Teknolojileri Bölümündeki bir öğretim elemanı da verebilir.



Öğretim elemanlarının istekde bulunduğu ders programı bölüm sekreterinin ekranına gelmektedir. Bölüm sekreteri Şekil 5.72'deki gibi önce görüntülemek istediği dersin bağlı bulunduğu bölümü daha sonra ise sırasıyla programı, yarıyılı ve dersi seçer. Bu işlemlerin ardından ekranda o dersle ilgili program istekleri varsa görüntülenir. Uygun bulunan program seçilir ve onaylama işlemi tetiklenir. Yada gerekiyorsa bazı düzenlemeler yapılabilir. Şayet herhangi bir program bildirimini yoksa bölüm sekreteri o dersi verebilecek öğretim elemanını belirler. Şekil 5.72'de de görüldüğü üzere Mevlüt Dogan adlı kullanıcı araştırma teknikleri dersini Pazartesi ve Çarşamba günü verebileceğini belirtmiş aynı zamanda o günler içinde saat kaçta uygun olduğunu belirtmiştir.

Ders programının manual olarak bildirilmesinden kaynaklanabilecek hataları ortadan kaldırmak için sisteme “+” ve “-“ tuşları konulmuştur. Bu tuşlar ile saat kısmında 1 er saat artırım yapılmaktadır. Dakikada ise 10 ar dakikalık artırım yapılmaktadır. Bu sayede onliene interaktif eğitim için ders programlarının belirlenmesi olabildiğince hassaslaşmıştır. Şekil 5.73'de bölüm sekreterinin bildirilen programları onaylama işlemine örnek verilmiştir.

Bolum	Bilgisayar Bolumu
Program	Bilgisayar Kontrol Sistemleri
Dönem	1.Sınıf Guz Donemi
Ders	Arastirma Teknikleri

Program No	7			
Donem No	6			
Ogretmen No	10			
Baslangic Saati	Saat	8	+	-
	Dakika	0	+	-
	Saati Onayla			
	15 : 30			
Bitis Saati	Saat	8	+	-
	Dakika	0	+	-
	Saati Onayla			
	17 : 0			
Gün	Carsamba			
<a href="#">Onayla</a> <a href="#">Cancel</a>				

### DERSİ VEREBİLECEK DİĞER ÖĞRETİM GÖREVLİLERİ

NO	Ders	Ogretmen No	Adi	Soyadi	Baslangic	Bitis	Gun	
7	Arastirma Teknikleri	10	Mevlut	Dogan	15 : 30	17 : 0	Carsamba	<a href="#">Select</a>

### DERSİ VERMESİ İÇİN ONAYLANAN ÖĞRETİM GÖREVLİLERİ

DIOT	Oto Id	Up Adı	Up Soyadı	DIOT Baslangic	DIOT Bitis	Gun	Adi	
5		Mevlut	Dogan	10 : 30	12 : 30	Pazartesi		<a href="#">Delete</a>
6		Mevlut	Dogan	8 : 00	10 : 00	Pazartesi		<a href="#">Delete</a>

Şekil 5. 73 Bölüm sekreterinin bildirilen programı düzenlemesi.

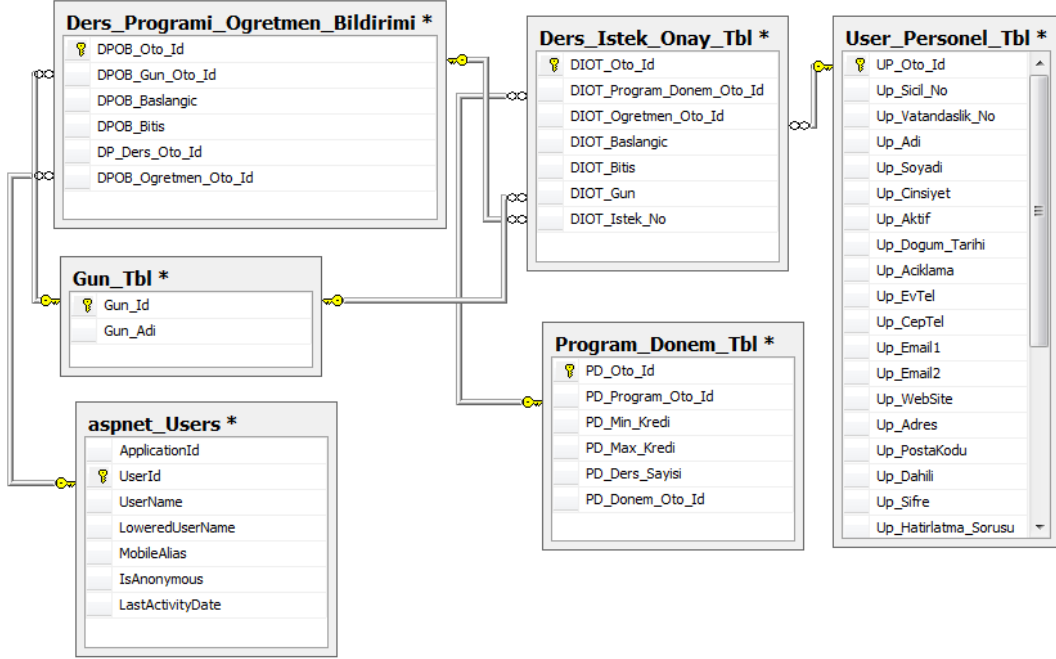
Ders sorumlusu ise ders programı bildirmek için ilk önce sisteme giriş yapar ve ilgili bağlantı aracılığı ile ders program işlemleri sayfasına ulaşır. Ancak öğretim elemanı bu işlemleri bahar/güz yarıyılları başlamadan yapabilmektedir. Verdiği dersler arasından ders saatini belirleyeceği dersi seçer. Haftanın hangi günü yapmak istediğini belirler. Başlangıç saatini ve bitiş saatini belirledikten sonra ders saati bildir işlemini tetikler. Bildirim ilgili bölüm sekreterine iletilir. Şekil 5.74’ de ders sorumlusunun ders istek saati uygulaması verilmiştir.

Bildir No	Baslangic Saati	Bitis Saati	Ders Adi	Gun	
5	10 : 30	12 : 30	Arastirma Teknikleri	Pazartesi	<a href="#">Select</a>
7	15 : 30	17 : 0	Arastirma Teknikleri	Carsamba	<a href="#">Select</a>

Şekil 5. 74 Ders sorumlusunun ders istek saatini belirtmesi.

Haftalık ders programı modülünde kullanılan tablolar ve bu tabloların ilişkili olduğu tablolar Çizelge 5,19’da verilmiştir.

**Çizelge 5. 17** Haftalık ders programı modülünde kullanılan tablolar ve bu tabloların ilişkili olduğu tablolar.



Haftalık ders programı bildirimleri Ders\_Programi\_Ogretmen\_Bildirimi tablosunda tutulmaktadır. Bu bildirimlerden onaylananlar ise Ders\_Istek\_Onay\_Tbl tablosunda tutulmaktadır. Bu iki tablonun ilişkili olduğu tablolar Çizelge 5.21’de verilmiştir. Ders\_Programi\_Ogretmen\_Bildirimi tablosu ve Ders\_Istek\_Onay\_Tbl tablosunun açıklaması ise aşağıda verilmiştir.

#### Ders\_Programi\_Ogretmen\_Bildirimi tablosu

- **DPOB\_Oto\_Id:** SQL tarafından verilen otomatik sıralama ID’sidir.
- **DPOB\_Gun\_Oto\_Id:** Bildirilen programdaki gün ID’sidir. Gun\_Tbl tablosu ile ilişkilidir.
- **DPOB\_Baslangic:** Bildirilen programın başlangıç saatidir.

- **DPOB\_Bitis:** Bildirilen programın bitiş saatidir.
- **DP\_Ders\_Oto\_Id:** Bildirilen programın ait olduğu Donemdeki dersin ID'sidir.
- **DPOB\_Ogretmen\_Oto\_Id:** Programı bildiren Öğretmenin ID'sidir.

### **Ders\_Istek\_Onay\_Tbl**

- **DIOT\_Oto\_Id:** Kabul edilen program için otomatik verilen ID'dir.
- **DIOT\_Program\_Donem\_Oto\_Id:** Kabul edilen programın ait olduğu Donemdeki dersin ID'sidir.
- **DIOT\_Ogretmen\_Oto\_Id:** Programı bildiren Öğretmenin ID'sidir.
- **DIOT\_Baslangic:** Kabul edilen programın başlangıç saatidir.
- **DIOT\_Bitis:** Kabul edilen programın bitiş saatidir.
- **DIOT\_Gun:** Kabul edilen programın günüdür. Gun\_Tbl tablosu ile ilişkilidir.
- **DIOT\_Istek\_No:** Bildirimi yapılan programın ID'sidir. Ders\_Programi\_Ogretmen\_Bildirimi tablosu ile ilişkilidir.

Ders\_Istek\_Onay\_Tbl tablosunda gün, başlangıç saati ve bitiş saati gibi verilerin tekrar tutulmasının sebebi, bölüm sekreterinin bildirim yapılan programda değişiklik yapabilme yetkisine sahip olmasındandır. Yani öğretim elemanının yapmış olduğu bildirim ile bölüm sekreterinin değişiklik yaptıktan sonra onayladığı programda farklılık oluşabilmektedir. Bu sebeple bu veriler her iki tablodada tutulmuştur.

## 5.10 Şube İşlemleri Modülü

Şube işlemleri modülü ile öğrenciler belirli guruplara ayrılabilir. Böylece sanal sınıf uygulaması gibi uygulamalarda internet yoğunluğu hafifletilerek daha hızlı internet erişimi sağlanabilir. Şube işlemleri modülü iki kısma ayrılmaktadır. Önce bölümlere şubeler eklenir. Daha sonra ders programı belirlenirken, uygun şube seçilir. Derse şube eklemek için önce bölüm içerisindeki programlar listelenir. Listelenen programlardan işlem yapılacak program seçilir. Şube işlemi yapacağı ders seçilir ve seçilen derse ait şubeler listelenir. Daha sonra şube ekle bağlantısı tetiklenerek ekrana çıkan paneldeki şube bilgileri doldurulur. Doldurulması gereken bu bilgilerde şube kodu bilgisi, şube adı bilgisi ve şube kapasitesi bilgisi yer almaktadır. Bu bilgiler doldurulduktan sonra şube kaydet işlemi tetiklenir ve girilen şube kodu ders şubeleri arasında varsa kullanıcıya hata mesajı döndürülür, yoksa yeni şube bölümdeki şubelerin arasına eklenir. Şube ekleme işlemi Şekil 5.75’de verilmiştir.

Bölüm Seç		Bilgisayar Bölümü
Sube Kodu	101	
Sube Adı	ATBIT_100	
Sube Kapasitesi	20	
<a href="#">Ekle</a> <a href="#">İptal</a>		



Sube No	Sube Kodu	Sube Adı	Sube Kapasitesi	
5	101	ATBIT_100	20	<a href="#">Seç</a>

Şekil 5. 75 Şube ekleme işlemi.

Ekleme işlemi gerçekleştirildikten sonra, liste güncellenir ve bölümde yer alan şubeler tekrar listelenir. Bölüm sekreteri listeden işlem yapmak istediği dersi seçer. Ders seçildikten sonra uygun şube sistemden seçilir ve o derse eklenir. Bu işlem Şekil 5.76'da verilmiştir.

Bölüm	Bilgisayar Bölümü
Program	Bilgisayar Kontrol Sistemleri
Dönem	1.Sınıf Güz Donemi
Dersler	Araştırma Teknikleri

SEÇİLMEYEN SUBELER				
No	Kodu	Adı	Kapasitesi	Seç
6	102	LAB_102	20	<input checked="" type="checkbox"/>
7	103	Fizik_103	50	<input type="checkbox"/>

SEÇİLEN SUBELER				
Seç	No	Kodu	Adı	
<input type="checkbox"/>	5	101	ATBIT_100	

➔

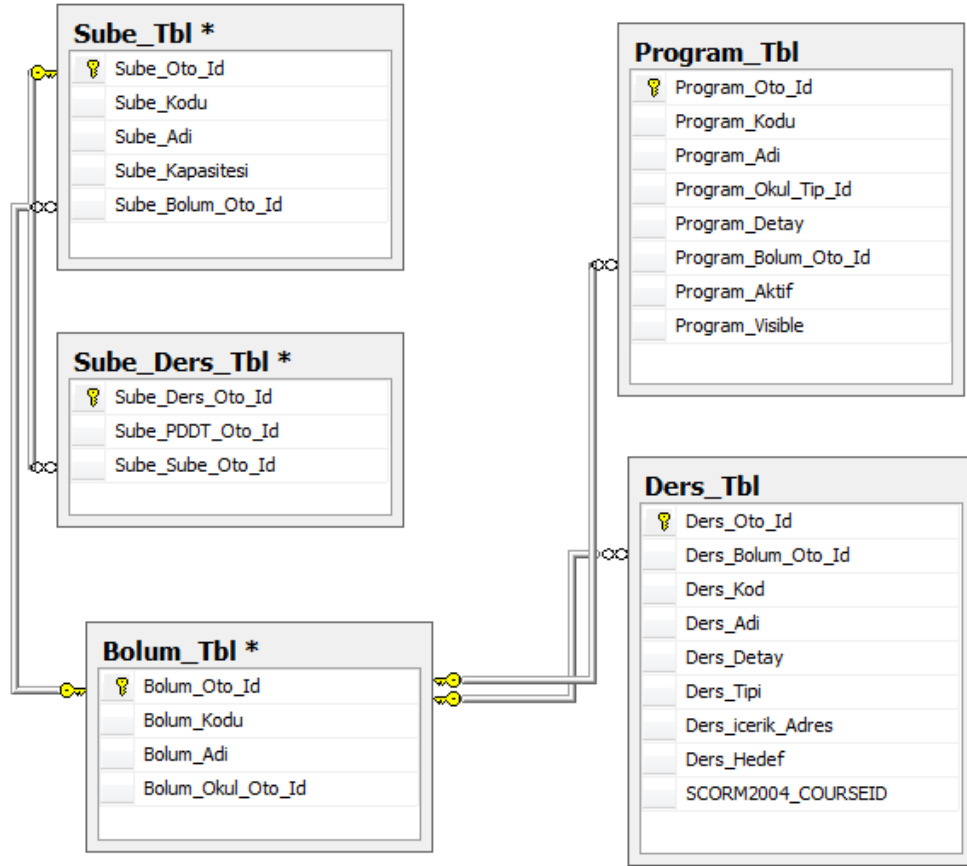
Şekil 5. 76 Ders şubesi tanımlama.

Şekil 5.76'da da görüldüğü gibi önce bölüme LAB\_102, Fizik\_103 ve ATBIT\_100 adlı 3 adet sanal sınıf eklenmiştir. Eklenen sanal sınıflardan ATBIT\_100 sınıfı araştırma teknikleri dersinin sanal sınıfı olarak kabul edilmiştir. Programlardaki öğrenci kapasitesine göre bir den fazla sanal sınıf oluşturulabilmektedir. Böylece ders anındaki bağlantı hızı artmış olur.

Sanal sınıflar oluşturulmasının ikinci sebebi ise dersi verecek öğretim elemanının öğrenciler ile olan etkileşim zamanını artırmaktır. Örneğin sanal sınıf uygulaması başlatıldığında sistemde 50 öğrencinin olması öğretim elemanının verimini düşürmektedir. Çünkü aynı anda daha çok öğrenci ile etkileşim içerisinde olması gerekmektedir. Sanal sınıf uygulaması ihtiyaç görüldüğü takdirde birebir özel ders te verilebilmektedir. Bu uygulama ile sınıf genelinde tam öğrenme hedefine ulaşılması sağlanabilmektedir.

Şube işlemleri modülü için kullanılan tablolar ve bu tabloların ilişkili olduğu tablolar Çizelge 5,20’de verilmiştir.

**Çizelge 5. 18** Şube işlemlerinde kullanılan tablolar ve bunların ilişkili olduğu tablolar.



Çizelge 5.22’den de anlaşılacağı gibi sisteme yeni eklenen şubeler Sube\_Tbl tablosuna eklenmektedir. Sube\_Ders\_Tbl tablosunda ise dersler için eklenen sube bilgileri tutulmaktadır. Bu tablolarda tutulan alanlar ise aşağıda verilmiştir.

#### Sube\_Tbl tablosu

- **Sube\_Oto\_Id:** Tanımlanan şubenin ID’sidir.
- **Sube\_Kodu:** Tanımlanan şubenin kodudur.



- **Sube\_Adi:** Tanımlanan şubenin adıdır.
- **Sube\_Kapasitesi:** Tanımlanan şubenin kapasitesidir.
- **Sube\_Bolum\_Oto\_Id:** Subenin eklendiği bölümün ID'sidir.

#### **Sube\_Ders\_Tbl**

- **Sube\_Ders\_Oto\_Id:** Ders için eklenen subenin ID'sidir.
- **Sube\_PDDT\_Oto\_Id:** Eklenen şubenin bağlı bulunduğu dönemdeki dersin ID'sidir. Program\_Donem\_Ders\_Tbl tablosu ile ilişkilidir.
- **Sube\_Subes\_Oto\_Id:** Şubenin ID'sidir. Sube\_Tbl ile ilişkilidir.

## 5.11 Danışman Atama İşlemleri Modülü

Danışman atama işlemleri modülünde ders kayıt işlemleri için onay verilen her öğrenciye danışman atanır. Bu onaylanma süreci öğrencinin dönem taksitini yatırıp yatırmadığına, okulla ilişkisinin kesilip kesilmediği gibi durumlara bağlıdır. Geçerli dönem için öğrencinin dönem kayıt işlemi onaylandıktan sonra, bölüm sekreteri bu öğrencilerin her birine danışman atar. Danışmanlar, öğrencinin kayıtlı olduğu bölümde görevli öğretim elemanlarıdır. Öğrenci ekranında ders kayıt işlemi onayının alınmasının ardından ilgili bağlantı linki belirir. Öğrenci bu bağlantı aracılığıyla ders kayıt işlemleri sayfasına ulaşır ve buradan ders kayıt işlemlerini tamamlar. Ders kayıt işlemini öğrencinin kendisi yapabildiği gibi danışmanı da yapabilmektedir. Ancak her durumda danışmanın ders kayıt işlemine onay vermesi gerekmektedir. Öğrenci ders kayıt işlemini yapmasının ardından ikinci bir değişikliği kendisi yapamaz. Bu değişikliği ancak danışmanı yapabilmektedir. Danışman da onayladıktan sonra düzenleme yapılacaksa artık gerekli düzenlemeler sistem yöneticisi tarafından yapılabilir. Öğrencinin ders alma işlemlerinde dikkat etmesi gereken nokta dönem içerisinde alabileceği maksimum kredi, minimum kredi, maksimum ders sayısı yada dersi alabilmesi için yeterli ön koşula sahip olup olmadığıdır. Örneğin, eğer öğrenci Bilgisayar 2 gibi bir dersi almak istiyorsa, Bilgisayar 1 dersinden başarılı olması gibi ön koşul olabilir. Böyle bir durumda bu dersi alamaz.

Öğrencilere danışman atayabilme işlemleri dönem başında gerçekleşmektedir. Bölüm sekreteri ders kayıt işlemleri için onay verdikten sonra, onaylanan her öğrenciye o dönem için bir danışman belirler. Bu işlemleri yapabilmek için bölüm sekreteri öncelikle danışman atama ile ilgili bağlantı aracılığı ile danışman atama işlemlerini yapabileceği sayfaya ulaşır ve aşağıdaki adımları sırasıyla gerçekleştirir.

1. Bölüm seçer.
2. Program seçer.
3. Yarıyıl seçer.
4. Onay alan öğrenciler listeler.

Danışman atanacak öğrenciler belirlendikten sonra danışman olarak atanacak öğretim elemanı seçilir. Bu işlem için önce danışmanın bulunduğu bölüm seçilir. Seçilen bölümdeki öğretim üyeleri listelenir ve buradan da öğrencilere danışman olarak atayacağı öğretim üyesini seçerek danışman atama işlemini tetikler. Bu işlemlerden sonra seçilen öğrencilere belirlenen öğretim üyesi danışman olarak atanır. Danışman atama işlemlerine örnek Şekil 5.77’de verilmiştir.

### ÖĞRENCİLERİ LİSTELE

Bölüm

Program

Dönem

Sıra No	Öğrenci Adı	Öğrenci Soyadı	Vatandaşlık No	Seç
225	Mahmu	Kantar	12131231231	<input type="checkbox"/>

### DANIŞMAN SEÇ

Okul

Bölüm

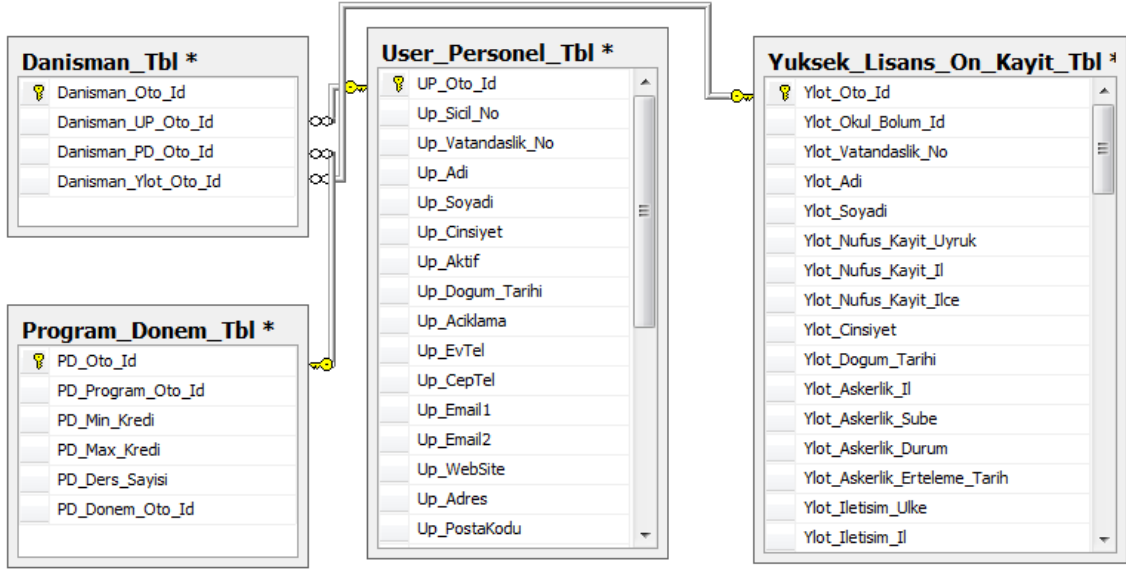
Öğretim Görevlisi



Şekil 5. 77 Danışman atama işlemi.

Danışman işlemlerinde kullanılan veri tabloları ve bu tabloların ilişkili olduğu tablolar Çizelge 5.21’de verilmiştir.

**Çizelge 5. 19** Danışman işlemlerinde kullanılan veri tabloları ve bu tabloların ilişkili olduğu tablolar.



Çizelge 5.21’den de anlaşılacağı gibi öğrenciler için atanan danışmanlar Danisman\_Tbl tablosunda tutulmaktadır. Bu tablodaki alanların açıklaması aşağıda verilmiştir.

#### Danisman\_Tbl Tablosu

- **Danisman\_Oto\_Id:** Danışman ID’sidir.
- **Danisman\_UP\_Oto\_Id:** Danışmanın sistemdeki personel ID’sidir. Bu alan User\_Personel\_Tbl tablosu ile ilişkilidir.
- **Danisman\_PD\_Oto\_Id:** Danışmanlık yapacağı dönemin numarasıdır.
- **Danisman\_Ylot\_Oto\_Id:** Danışmanlık yapacağı öğrencinin numarasıdır.

## 5.12 Ders Kayıt ve Onay İşlemleri Modülü

Öğrencinin ders kayıt işlemi yapabilmesi için bölüm sekreterinin kayıt yapılmak istenen dönem için onay vermesi gerekmektedir. Bunun için öğrencinin kayıt yapmak istediği dönemle ilgili gerekli yeterliliğe ulaşması gerekmektedir. Öğrencinin daha önceki dönemlerden başarısız olduğu bir ders varsa öncelikle o dersi seçmesi gerekmektedir. Ayrıca seçmek istediği dersle ilgili ön koşula, yeterliliğine sahip olup olmadığına dikkat etmesi gerekmektedir. Öğrenci kayıt yapılacak dönem için belirlenen maksimum kredi miktarı ve maksimum ders sayısı kadar ders alabilir. Bu yüzden bir üst dönemden de ders alabilir. Ancak öğrenci bir önceki dönemdeki dersleri ve kayıt yapılacak dönemdeki dersleri önce almalıdır. Öğrencinin dönem kayıt işlemi onaylandıktan sonra, bölüm sekreteri bu öğrencilerin her birine danışman ataması yapmaktadır. Öğrencilere danışman atayabilme işlemleri ve ders kayıt işlemleri dönem başında gerçekleşmektedir. Öğrencilere danışman atamak için ilk önce danışmanlığını yapacağı öğrenceleri seçer, sonra bu öğrencilere atayacağı danışmanı belirler ve onaylama işlemini tetikler. Danışman atama işlemleri 5.11 Danışman atama işlemleri modülünde ayrıntılı olarak anlatılmıştır.

### 5.12.1 Ders Kayıt İşlemleri

Ders kayıt dönemi başladığında, bölüm sekreteri dönem taksitini yatıran ve kaydında sorun olmayan öğrencilere kayıt yapabilmeleri için onay vermektedir. Bunun için önce işlem yapmak istediği programa kayıt yapabilecek öğrenciler listelenir ve seçilen öğrencilerin kayıt yapabileceği dönem belirlenir. Ders kayıt işlemleri için bölüm sekreterinin onay verme işlemi Şekil 5.78’de verilmiştir.

## Seçilen Öğrencilerin Kayıt Edileceği Dönem

- 1. Sınıf Guz Donemi
- 1. Sınıf Bahar Dönemi
- 2. Sınıf Guz Dönemi
- 2. Sınıf Bahar Dönemi
- 3. Sınıf Guz Dönemi
- 3. Sınıf Bahar Dönemi
- 4. Sınıf Guz Dönemi
- 4. Sınıf Bahar Dönemi

Onayla

Bölüm Fizik Ana Bilim Dalı

Program İlk Öğretim Fizik Öğretmenliği

### PROGRAMA KAYITLI ÖĞRENCİLER

Kayıt No	Adı	Soy Adı	Vatandaşlık No	Seç
1	Emin	İbili	12121212121	<input checked="" type="checkbox"/>
3	Fatih	Bayram	12312121231	<input checked="" type="checkbox"/>
4	Fidan	Hakkari	12121212122	<input type="checkbox"/>

Şekil 5. 78 Ders kayıt işlemleri için bölüm sekreterinin onay vermesi.

Şekil 5.78’de de görüldüğü gibi önce öğrencilerin kayıt yaptırabileceği dönem seçilmelidir. Daha sonra işlem yapmak istediği bölümün ve programın seçilmesi gerekmektedir. Program seçme işleminin ardından o programa kayıtlı öğrenciler listelenmektedir. Bölüm sekreteri listedeki öğrenciler arasından kayıt yapmasına izin vereceği öğrencileri belirlemektedir.

### 5.12.2 Ders Kayıt

Ders kayıt dönemi başladığında öğrenci sisteme giriş yapar ve ilgili bağlantı aracılığıyla ders kayıt sayfasına ulaşır. Şekil 5.79 'da da görüldüğü gibi öğrenci ders kaydı yapmak istediği dönemi seçer. Daha sonra almak istediği dersleri seçer. Seçeceği derslerin, dönemde seçilebilecek, minimum kredi, maksimum kredi, maksimum ders ve zorunlu seçmeli, ders gurupları dikkate alınarak kayıt işlemi gerçekleştirilir. 1. Dönem derslerini aldıktan sonra hala ders alabileceği kadar kredisi varsa ve aldığı ders sayısı maximum ders sayısının altındaysa, 1. Sınıf güz döneminin derslerini listeler ve istiyorsa bu dönemden de ders seçebilir. Dersler seçildikten sonra, listeye ekleme butonu olan aşağı ok işaretli butona tıklanır. Seçmiş olduğu dersi listeden çıkarmak istiyorsa yukarı ok tuşuna basması gerekmektedir. Bu işlemlerin en sonunda öğrencinin seçilen derslere onay vermesi gerekmektedir. Onaylama işlemini tetikledikten sonra şayet ders kayıt işleminde hata varsa öğrenciye hata mesajı döndürülür ve öğrenci hata mesajına göre ders kayıt işleminde değişiklikler yapar. Dikkat edilmesi gereken diğer bir noktada öğrenci ders kayıt işlemini yaptıktan sonra ikinci bir değişikliği kendisi yapamamaktadır. Ancak danışmanı bu değişiklikleri yapabilir. Danışmanda onay vermişse ve yapılması gereken değişiklik zorunlu bir değişiklikle bu sefer sistem yöneticisi bu değişikliği yapmaktadır. Şekil 5.79'da öğrencinin ders kayıt işlemine örnek verilmiştir.

Kayıt Yapmak İstedığınız Dönem

- 1. Sınıf Guz Donemi
- 1. Sınıf Bahar Dönemi
- 2. Sınıf Guz Dönemi
- 2. Sınıf Bahar Dönemi
- 3. Sınıf Guz Dönemi
- 3. Sınıf Bahar Dönemi
- 4. Sınıf Guz Dönemi
- 4. Sınıf Bahar Dönemi

Listelemek İstedığınız Derslerin Dönemi

1. Sınıf Guz Donemi

No	Basvuru No	Ders Adı	Ders Kodu	Kredi No	Zorunlu Secmeli	Gurup No	Gurup Adeti	Seç
8	6	Biyoloji	2113	2	1	15	12	<input type="checkbox"/>



No	Basvuru No	Ders Adı	Ders Kodu	Kredi No	Zorunlu Secmeli	Onay	Seç
7	6	Arastirma Teknikleri	12	1	1		<input type="checkbox"/>
8	6	Fizik	123	1	1		<input type="checkbox"/>



Maximum Kredi **2** Minimum Kredi **2** Maximum Ders Sayisi **2**  
Aldığımız Kredi **2** Aldığımız Ders **2**

ONAYLIYORUM

Şekil 5. 79 Öğrenci ders kayıt işlemi.

### 5.12.3 Danışman Ders Kayıt Onayı

Danışman, sisteme giriş yaptığında ve ilgili sayfaya ulaştığında, danışmanı olduğu öğrenciler listelenir yada programlara göre danışmanı olduğu öğrencileri listeleyebilir. Bu öğrencilerden ders kayıt işlemlerini onaylamak istediği öğrenciyi seçer. Öğrenci ders kayıt işlemini onaylamış ise, ders kayıt işlemini kontrol eder. Onaylama işlemi tamamlandıktan sonra öğrenci almış olduğu derslere öğrenci ekranından ulaşabilir ve dersleri takip edebilir. Bu işlemler Şekil 5.80'de verilmiştir.



Okul	Fen Bilimleri Enstitüsü
Bölüm	Bilgisayar Bölümü
Program	Bilgisayar Kontrol Sistemleri
Dönem	1.Sınıf Güz Donemi

### DANIŞMANI OLDUĞUNUZ ÖĞRENCİLER

Sıra No	Program No	Adı	Soyadı	Vatandaşlık_No	
6	64	Mahmut	Kantar	11111111111	Select

LİSTELEYECEĞİNİZ DERSLERİN DÖNEMİNİ SEÇİNİZ

1.Sınıf Güz Donemi

### SEÇİLMEYEN DERSLER

No	Ders Kodu	Ders Adı	Kredi No	Zorunlu Secmeli	Gurup No	Gurup Adeti	Seç
8	2113	Biyoloji	2	1	15	12	<input type="checkbox"/>

### SEÇİLEN DERSLER

ID	No	Ders Kodu	Ders Adı	Kredi No	Zorunlu Secmeli	Gurup No	Gurup Adeti	Seç
7	6	12	Arastirma Teknikleri	1	1	15	12	<input type="checkbox"/>
8	7	123	Fizik	1	1	15	12	<input type="checkbox"/>

### DERS SAYISI VE KREDİ ORANLARI

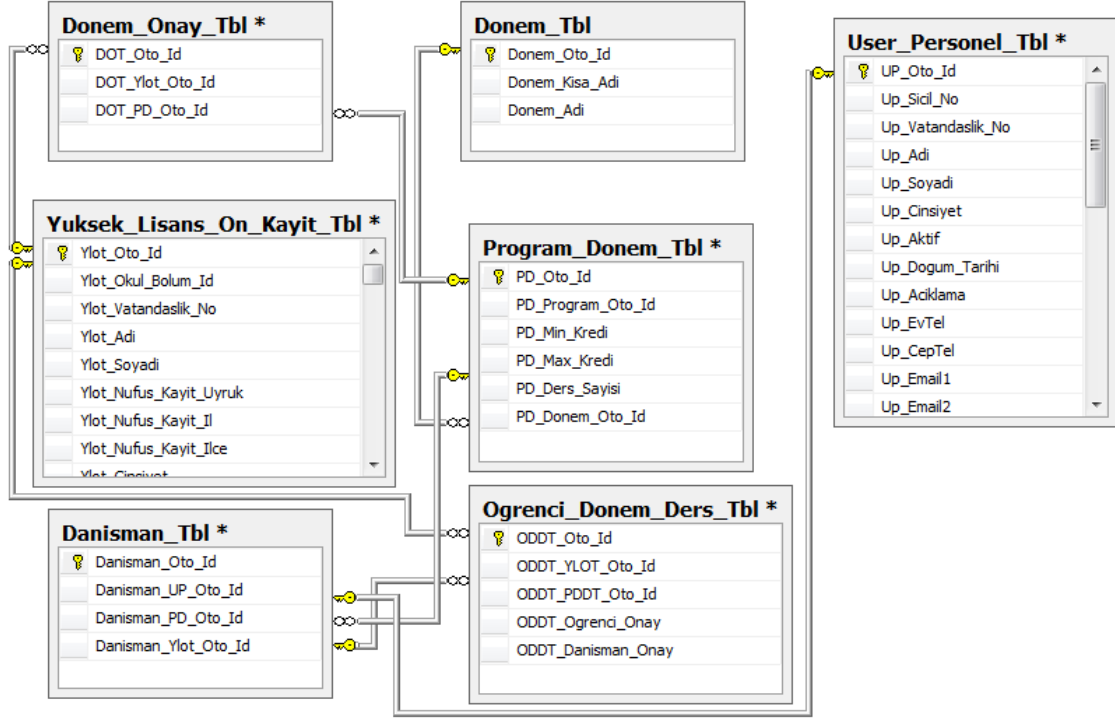
Maximum Kredi **2** Minimum Kredi **2** Maximum Ders Sayisi **2**  
 Aldığımız Kredi **2** Aldığımız Ders **2**

ONAYLIYORUM

Şekil 5. 80 Danışman ders onay işlemi.

Ders kayıt işlemleri modülünde kullanılan tablolar ve bunların ilişkili olduğu tablolar Çizelge 5.22'de verilmiştir.

**Çizelge 5. 20** Ders kayıt işlemlerinde kullanılan tablolar ve bu tabloların ilişkili olduğu tablolar.



Öğrenciler için belirlenen danışmanlar Danisman\_Tbl tablosunda tutulmaktadır. Dönem için ders kayıt onayı alan öğrenciler Donem\_Onay\_Tbl tablosunda ve alınan dersler ise Ogrenci\_Donem\_Ders\_Tbl tablosunda tutulmaktadır. Bu üç tablonun ilişkili olduğu tablolar Çizelge 5.22’de, tabloadaki alanların açıklamaları ise aşağıda verilmiştir.

### Donem\_Onay\_Tbl

- **DOT\_Oto\_Id:** Her öğrenciye dönem ders onayı verildiğinde otomatik verilen ID’dir.
- **DOT\_Ylot\_Oto\_Id:** Ders kayıt onayı verilen öğrenci numarasıdır. Ylot\_Ogrenci\_Tbl ile ilişkilidir.
- **DOT\_PD\_Oto\_Id:** Ders kayıt onayı verilen dönemdir. Donem\_Tbl ile ilişkilidir.

## **Danisman\_Tbl**

- **Danisman\_Oto\_Id:** Her danışmanın kayıt ID'sidir.
- **Danisman\_UP\_Oto\_Id:** Danışman olarak atanan personel numarasıdır. User\_Personel\_Tbl ile ilişkilidir.
- **Danisman\_PD\_Oto\_Id:** Danışmanlık yapacağı dönemdir. Program\_Dönem\_Tbl ile ilişkilidir.
- **Danisman\_Ylot\_Oto\_Id:** Danışmanlık yapacağı öğrenci numarasıdır.

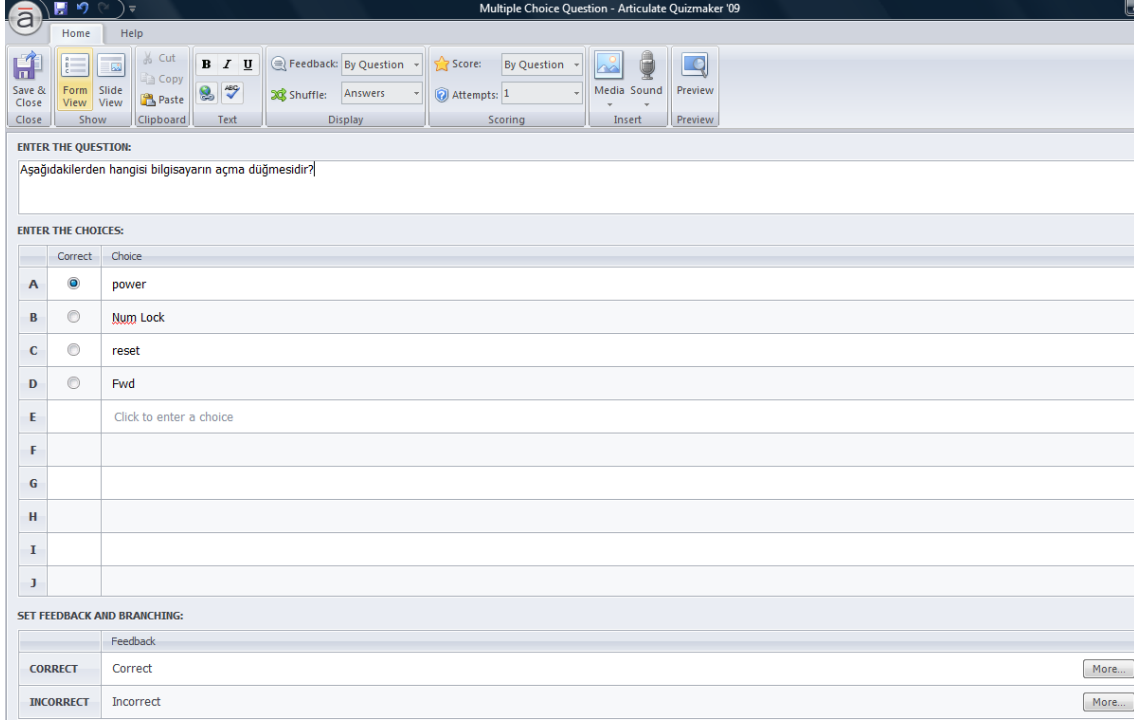
## **Ogrenci\_Donem\_Ders\_Tbl**

- **ODDT\_Oto\_Id:** Alınan ders ID'sidir.
- **ODDT\_YLOT\_Oto\_Id:** Dersi alan öğrencinin numarasıdır.
- **ODDT\_PDDT\_Oto\_Id:** Alınan dersin ID'sidir. Program\_Donem\_Ders\_Tbl ile ilişkilidir.
- **ODDT\_Ogrenci\_Onay:** Öğrencinin dersi kayıt işlemine onay verip vermediği bilgisidir. Verdi ise "1" vermedi ise "0" değerini alır.
- **ODDT\_Danisman\_Onay:** Danışmanın öğrencisinin dersi kayıt işlemine onay verip vermediği bilgisidir. Verdi ise "1" vermedi ise "0" değerini alır.

### 5.13 Soru Tanımlama Modülü

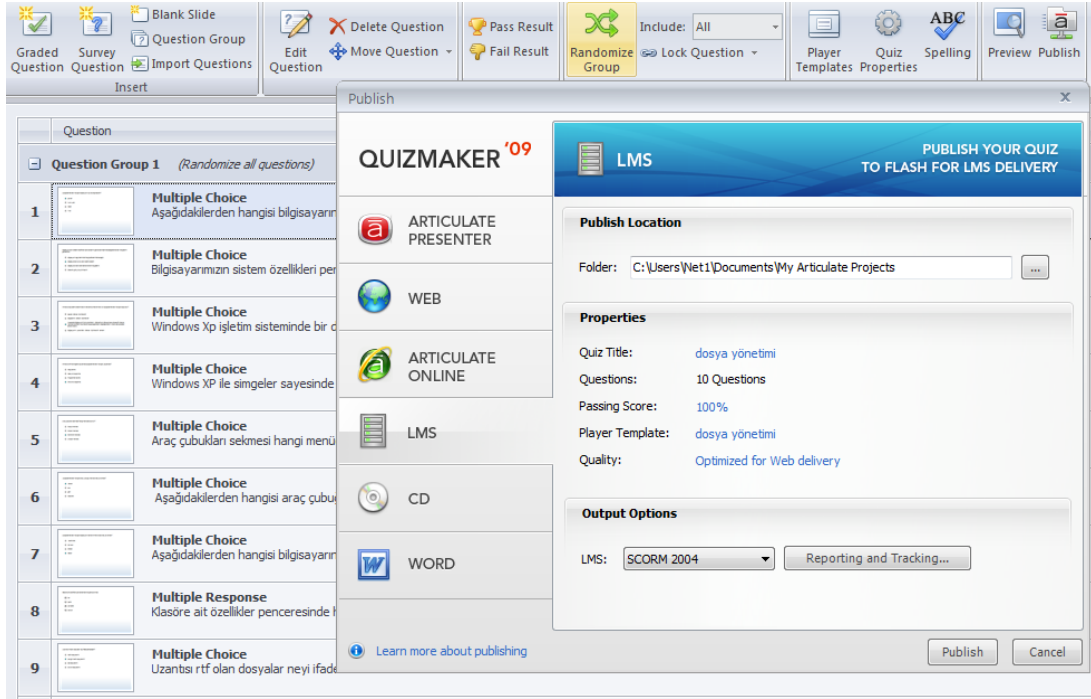
Uzaktan eğitim uygulamalarında dönem içerisinde alından dersin öğrenci tarafından ne derecede etkili olduğunun ölçülmesi gerekmektedir. Böylece içeriklerin öğrenci ihtiyaçlarına göre yenilenmesinde ve daha etkin hale getirilmesinde büyük yardım sağlayacaktır. Bu sebeple diğer uzaktan eğitim sistemlerinde olduğu gibi geliştirilen uzaktan eğitim sisteminde de bu çalışma sınav ve testler yoluyla yapılmaktadır. Genelde bu tip sistemlerde kullanılan iki farklı yöntem genel olarak tercih edilmektedir. Bunlardan biri dönem sonunda öğrencilerin bir merkezde toplanarak sınava tabi tutulmalarıdır. Bu sistem farklı ülkelerden sisteme dahil olan kullanıcılar için uygun bir yöntem değildir. Bu durumda çevrimiçi sınavlar devreye girmektedir. Bunlara en güzel örnek TOEFL sınavları ve Microsoft sınavlarıdır. Öğrenciler terminaller yardımıyla merkezden gelen soruları yanıtlamaktadırlar. Genel değerlendirmede kullanılacak testlerin yanı sıra, sadece deneme amaçlı olarak testlerin oluşturulabilmesi ve bu testlerin eğitim sistemi üzerinden öğrenciye sunulabilmesi de gerekmektedir.

Sistemde sınav işlemleri SCORM uyumlu sorular ile gerçekleşmektedir. Soru Tanımlama Modülü aracılığı ile sisteme SCORM uyumlu quiz yada diğer sınavlar eklenerek öğrenciler sınav yapılmaktadır. SCORM soru paketi hazırlamak için Articulate Quizmaker' 09 Programından yararlanılmıştır. Articulate 2009 tercihi edilmesinin diğer bir sebebi de bu program ile birçok farklı tipte soru tanımlanabilmektedir. Örneğin soru tipi doğru/ yanlış olan soru ekleme, eşleştirmeli soru ekleme, çoktan tek seçmeli soru ekleme, çoktan çok seçmeli soru ekleme gibi birçok farklı tipte soru tanımlanabilmektedir. Ayrıca Articulate Quizmaker' 09 ile soruların zorluk derecesi, sorular için gerektiğinde resim ekleme gibi ek özelliklerde ayarlanabilmektedir. Ancak ileriki zamanlarda EK-2'de verilen soru tanımlama modülündeki analiz raporu dikkate alınarak yeni bir soru tanımlama modülünün oluşturulması hedeflenmektedir. Aşağıda Şekil 5.81'de verilen örnek sınavın Articulate Quizmaker' 09 ile SCORM paketlerine nasıl dönüştürüldüğü anlatılmaktadır.



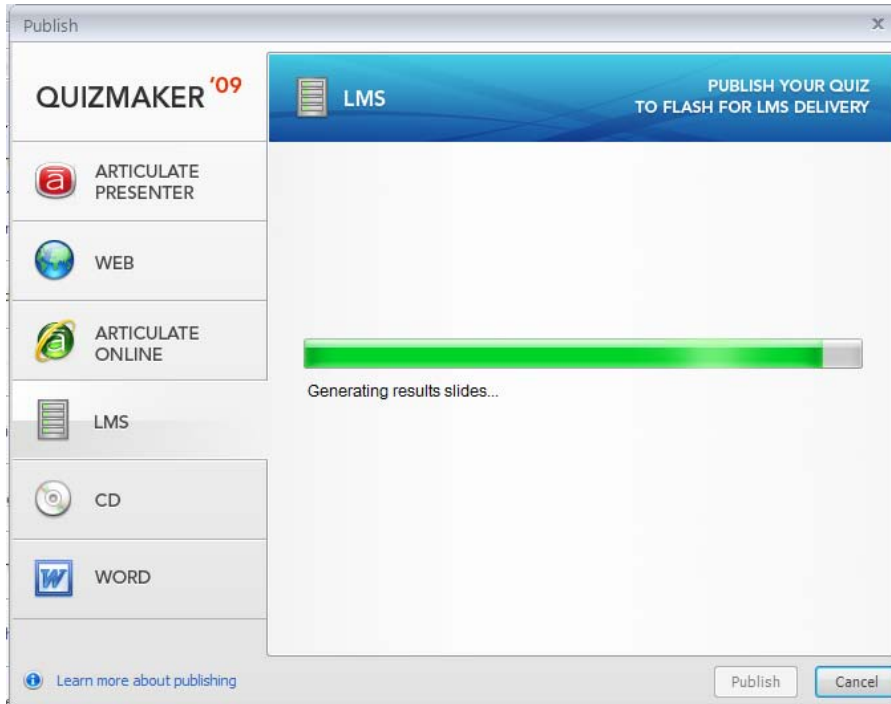
**Şekil 5. 81** Articulate ile sınav hazırlamak.

Sınav soruları hazırlama işlemi bittikten sonra Scorm 2004 paketi almak için ekranın sağ üst köşesinde bulunan Publish tuşuna tıklanır ve çıkan pencereden önce LMS seçeneği seçilir, sonra oluşturulmak istenen paket tipi seçilir (SCORM 1.2 yada SCORM 2004). Şekil 5.82’de de görüldüğü gibi Articulate Programı aracılığı ile “DosyaYonet” adında SCORM 2004 standartlarına uygun olarak bir sınav paketi oluşturulmuştur.



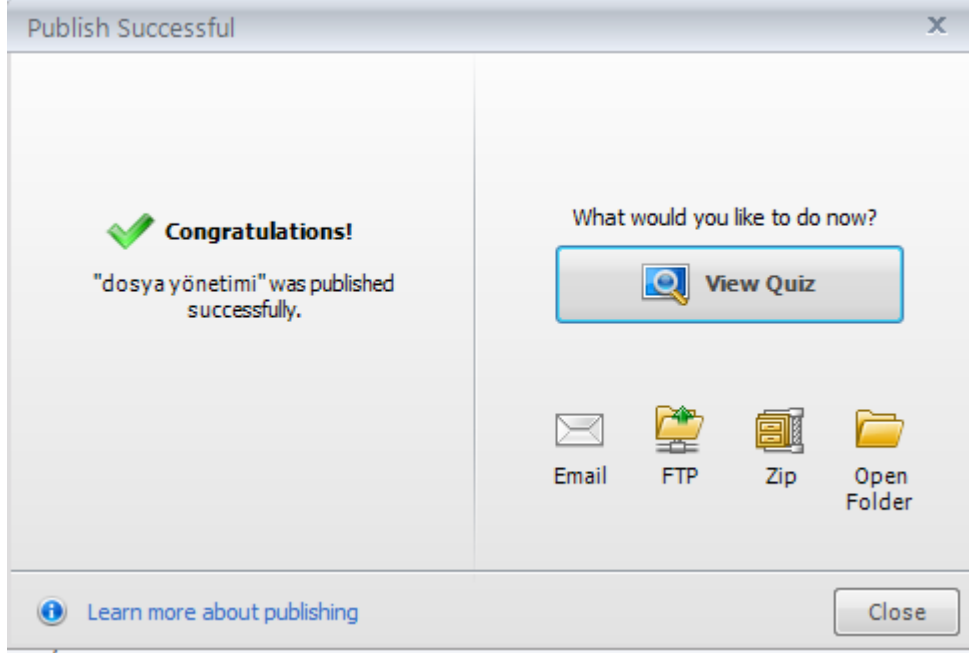
Şekil 5. 82 Soruların SCORM uyumlu hale getirilmesi.

Publis Tuşuna basıldıktan sonra Articulate Şekil 5.83'deki gibi Scorm uyumlu paketi hazırlanmaktadır.



Şekil 5. 83 Articulate programının hazırlanan soruları SCORM paketine dönüştürmesi.

Hazırlama işleminin başarılı bitmesinin ardından “.zip” paketi olarak alınıp sisteme sınav olarak eklenir.



Şekil 5. 84 Articulate programı ile SCORM 2004 paketinin oluşturulması.

## 5.14 Sınav Modülü

Geliştirilen Web-Tabanlı Sınav Sistemi, teknoloji destekli eğitimin bir parçası olarak üniversiteler için bir alt yapı oluşturacak önemli bir adımdır. Öğretmenlerin soru bankası hazırlaması bir önceki modül olan soru tanımlama modülünde anlatılmıştır. Bu modülde ise soru tanımlama modülünde Articulate yada diğer SCORM paketi verebilen programlarda hazırlanmış ve SCORM paketi haline getirilmiş sınav paketleri sisteme sınav olarak eklenmektedir.

Sistem öğretim elemanı, öğrenci ve bölüm sekreteri olmak üzere üç kullanıcı grubu tarafından kullanılmaktadır. Öğretim elemanları Articulate Quizmaker' 09 gibi programlar kullanarak SCORM uyumlu sınav paketleri oluşturmaktadırlar ve hazırlanan bu paketleri bölüm sekreterine göndermektedirler. Bölüm sekreteri ise öğretim elemanının belirtmiş olduğu dersin sınavına bu paketi vize, yada final sınavı olarak eklemektedir.

Bölüm sekreteri, bölümlerde okutulan derslerin değerlendirmesinin hangi sınavlardan oluşacağını Şekil 5.85'de de görüldüğü gibi tanımlamaktadır. Önce eklemek istediği sınavın hangi derse ait olacağını belirler. Bunun için önce bölüm seçer. Bölüm seçiminin ardından o bölümde okutulan dersler listelenir ve listedeki bu derslerden hangisine sınav ekleyecekse o dersi seçer. Daha önce bu derse eklenmiş sınavlar varsa bunlar görüntülenir. Önce sınavın vize sınavı mı yoksa final sınavı mı olduğu belirtilir. Eğer sınav vize ise vize numarasını girer. Sınav adı kısmında Vize+Sınav no şeklinde sınavın ismi listelenir. Sınav yüzdeleri kısmında ise sınavın öğrencinin not ortalamasına yapacağı yüzdeleri katkıyı belirlemelidir. Sınav bilgilerinin tanımlanması bittikten sonra bu sınav Şekil 5.85'de de görülen SCORM 2004 soru paketi ekle linkine tıklanır ve açılan SCORM paketi yükleme sayfasından ise paket seçilir ve sisteme yüklenir.



Bölüm Seç

Ders Seç

Sınav Tipi	<input checked="" type="radio"/> Vize <input type="radio"/> Final	<b>Vize No</b> <input type="text" value="1"/>
Sınav Adı	<input type="text" value="Vize1"/>	
Sınav Yüzdeleri	<input type="text" value="10"/>	
<a href="#">Insert</a> <a href="#">Cancel</a>		

Sınav No	Sınav Adı	Sınav Tipi	Sınav Yüzdeleri	
4	Vize1	True	10	<a href="#">Scorm 2004 Soru Paketi Ekle</a>

**Şekil 5. 85** Sınav paketi ekleme.

Articulate programında hazırlanmış ve SCORM Paketi haline getirilmiş olan sınavlar ise Şekil 5.86'daki gibi şekilde sisteme eklenmektedir.

## Ders Paketi Yükle

Ders ile ilişkilendirmek istediğiniz SCORM paketini (.zip, .pif) seçip GÖNDER düğmesine tıklayınız:

SCORM Paketi: QUIZLER\dosyaYonet.zip

### Şekil 5. 86 Sınav paketini seçme

Bir önceki modülde Articulate Quizmaker' 09 ile hazırlanan DosyaYonet.zip adındaki sınav paketi Şekil 5.86'da da görüldüğü gibi seçilen sınava eklenmiştir. Eklenen sınav bilgileri Donem\_Sınav\_Tbl tablosunda tutulmaktadır. Bu tablodaki alanlar ve bu alanların açıklamaları aşağıda verilmiştir.

- **DST\_Oto\_Id:** Eklenen sınavın ID'sidir.
- **DST\_Sınav\_Adi:** Eklenen sınavın adıdır.
- **DST\_Sınav\_Tip:** Eklenen sınavın tipidir.
- **DST\_Yuzdelik:** Sınavın dersin ortalamasına yapacağı yüzdellik katkısıdır.
- **DST\_Ders\_Oto\_Id:** Eklenen sınavın ait olduğu derstir.
- **DST\_Scorm\_Adres:** Sınava ait eklenen SCORM paketinin sistemdeki adresidir.

## 5.15 Deneme Sınavı Modülü

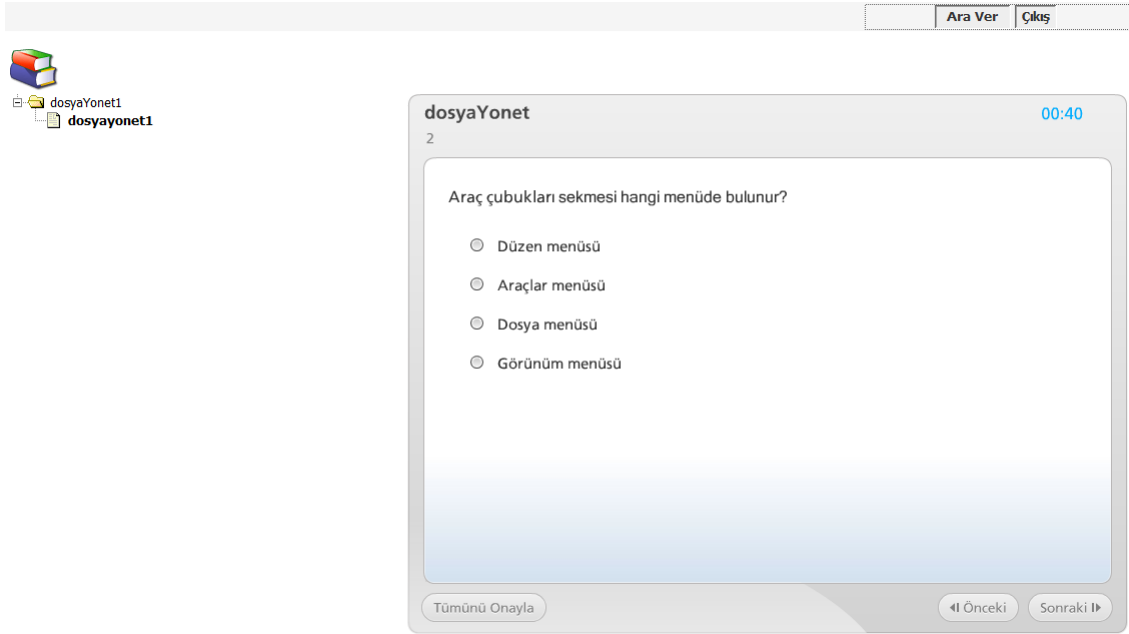
Eğitimin konularından biri de, öğrencilerin edindikleri bilgi ve becerilerinin ölçülmesidir. Klasik okul/sınıf ortamlarında, öğrencilerin aynı zamanda aynı mekana toplanmasıyla yapılan sınavlar, e-öğrenme veya uzaktan eğitim söz konusu olunca geçerliliğini ve uygulanabilirliğini yitirmektedir. İnternet destekli eğitim alan öğrencilerin klasik sınavlara sokulmasının zorlukları ortadadır. Bu nedenle uzaktan eğitimi destekleyecek bir uzaktan sınav, veya bir başka deyişle, e-öğrenmeyi destekleyecek bir e-sınav modeline gerek vardır (Özmen 2006). Hazırlan öğrenim yönetim sistemindeki deneme sınavı modülünde sadece ders saatleri içinde sınav uygulaması yapılabildiği gibi ders saatleri dışında da sınavlar yapılabilmesi mümkündür. Bu modül ile birlikte öğrenciler sınavları internet üzerinden evlerinden olabilmektedir.

Öğrencinin sınav anında verileri SCORM API tarafından izlenmektedir. SCORM API'nin öğrenci verilerini izlemesi ve kullanıcı ile etkileşimini oluşturmak için LMS'de iki çerçeveden oluşan bir ana çerçeve oluşturulmuştur. İçeriği barındıran çerçeve %100 yükseklikteyken API'yi barındıran çerçeve %0 yüksekliklidir. Bu şekilde kullanıcı çerçeve kullanıldığının farkına varmamaktadır. Ana çerçeve sayfası API isimli bir değişken barındırıp bunu API çerçevesindeki API objesine yönlendirir. API başlatılırken LMS öğrenci ID'sini, ders ID'sini ve LMS'in ihtiyaç duyacağı diğer ID'leri API'ye atar. Bu değişiklikler LMS'e gönderilen XML dokümanına yansıtılır. Daha sonrada veri tabanına kaydedilir (Bkz. 3.4 SCORM 1.2 API özellikleri ve ve 3.5 SCORM 2004 API özellikleri).

Deneme sınavı olabilmek için öğrenci kendi kullanıcı sayfasını açtığı anda sınav zamanı ise ekranın da sınav linki belirir ve bu linke tıklayarak vize, final yada quiz olabilir. Öğrenci sınav işlemini bitirdikten sonra sınav ekranının üst kısmındaki çıkış butonuna basması gerekmektedir. Böylece almış olduğu puan hanesine yazılmaktadır. Öğrenci sınav seçeneğini tıkladığında, karşısına o anda alabileceği sınavların listesi gelir. Öğrenci bu sınavlardan birini seçer. Soru ekranları görüntülendiğinde, seçtiği sınavın ilk sorusu karşısına gelir. Öğrenci sorunun cevabını girebilir veya başka bir soruya geçebilir. Sorunun cevabını girdiğinde, otomatik olarak bir sonraki soru karşısına gelir.

Öğrenci, sorular içinde dolaşabilir, yani ekranına sınavın herhangi bir sorusunu getirebilir. Daha önce cevapladığı bir soruyu getirdiğinde, önceki girdiği cevapta ekranında belirir, cevabında değişiklik yapabilir. Öğrenci artık yeni cevap girmeyecekse veya cevaplarında değişiklik yapmayacaksa, sınavı bitirdiyse, ekranında "Bitir" butonunu tıklayarak sınavını tamamlar. Sınav bitiminin ardından öğrenci çıkış butonuna basarak SCORM API tarafından tutulan gerekli veriler SCORM tablolarına kaydedilir.

Şekil 5.87’de bir önceki modülde eklenen “dosyayonet1” adındaki sınavın online ekran görüntüsü verilmektedir. Ekranın sol tarafında ise o pakette bulunan sınavlar görüntülenmektedir. Eklenen pakette tek bir sınav olduğundan sadece bir tane görüntülenmektedir. Sınavın sonunda ise öğrenci çıkış butonuna basarak, verilerini kaydetmektedir. Aksi takdirde bir sonraki girişinde düzgün çıkış yapmadığını belirten bir uyarı penceresi ile karşılaşır. Ancak ekrana gelen hata penceresindeki çıkış butonuna basarak saklanan verileri gerekli tablolara kaydedilir. Öğrenci düzgün çıkış yapmadan ikinci bir sınavı asla başlatamaz. Sınav anında tutulan öğrenci verileri SCORM tablolarında tutulmaktadır. Bu tablolar ise 3.4 ve 3.5 bölümlerinde anlatılmıştır.



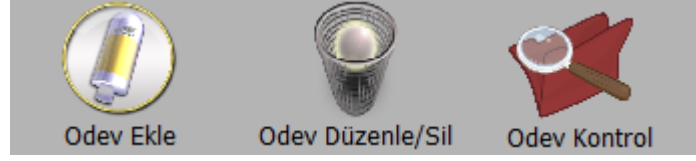
Şekil 5. 87 Online sınav ekranı.

## 5.16 Ödevler Modülü

Uzaktan eğitim sistemlerinde öğretimin etkinliğini artırmak için gerekli olan modüllerden birisidir. Öğretim elemanı öğrencileri araştırmaya teşvik etmek amacıyla hazırlanan sistem üzerinden ödevler göndermektedir. Ödev öğrencilere bir word dosyası, text dosyası yada zip dosyası olarak gönderilebilmektedir. Öğrenciler ise verilen ödevi açarak görüntüleyebilmektedir. Ancak ödev öğrenci ekranlarında, öğretim elemanının sisteme girerken belirtmiş olduğu ilan tarihinde görüntülenmektedir. Eğer ödevi gönderme süresi dolmuş ise ödev yayından otomatik olarak kalkmaktadır. Ancak isterse öğretim elemanı ödevin süresini uzatabilmektedir. Öğrenciler ödevlerini hazırladıktan sonra yine sistem üzerinden gönderebilirler. Gönderilecek ödev, word PowerPoint, Text veya Zip formatında olmak zorundadır. Öğretim elemanı gönderilen ödevleri bilgisayarına yükleyerek değerlendirebilmektedir. Ödevlerin öğrencinin not ortalamasına etkilediği gibi öğretim elemanının performans raporunu da etkilemektedir. Bölüm başkanlarının, bölümünde bulunan öğretim elemanlarının performansını takip etmek istediklerinde, öğretim elemanının yıl içinde kaç defa ödev gönderdiği gibi istatistikler de bu raporda yer almaktadır.

### 5.16.1 Öğretim Elemanı Ödev Gönderme İşlemi

Ödev gönderme işlemi yapmak için, öncelikle kullanıcının sisteme giriş yaparak ilgili bağlantı aracılığı ile ödev gönderme sayfasına ulaşması gerekmektedir. Öğretim elemanının vermiş olduğu dersler listelenir. Öğretim elemanı ödev göndermek istediği dersi seçer. Ödev adını, ödevin ilan tarihini, ödev teslim tarihini girer ve varsa ilgili ödev dosyasını ekler. Daha sonra Ödev vereceği öğrencileri seçer ve ödev gönderme işlemini tetikler. Ödev başarılı ise seçilen öğrencilerin ajandalarına olay olarak eklenir. Eklenen olayın başlığı ise ödevdir. Şekil 5.88’de ödev gönderme işlemine örnek verilmiştir.



Dosya Ekleme İstedığınız Dersi Seçiniz:

Arastirma Teknikleri  
Arastirma Teknikleri  
Fizik

Odev Ekle

Odev No	Odev Adı	Odev Teslim Tarihi	Odev İlan Tarihi
118	Arastirmada Bilimsel Yöntem Nedir?	2/1/2009	2/4/2009
119	Bilimsel Yöntem Nedir	2/2/2009	2/2/2009
120	Nicel Gözlem Nedir	2/1/2009	2/1/2009

**Odev Ekleneyecek Ders: Arastirma Teknikleri**

[Dersi Değiştir](#)

Dosya Açıklama Bilgisi (Max. 50) :

Odev Adı:

Odev İlan Tarihi:

Odev Teslim Tarihi:

Göndermek İstedığınız Dosyayı Seçiniz (Max. 2 Mb) :

**Dosya Başarıyla Sisteme Yüklenmiştir.**

**Odevi Göndermek İstedığınız Öğrencileri Seçiniz**

Eklenecek Öğrenciler  Eklenecek Öğrenciler

Yıl Ot Id	Yıl Adı	Yıl Soyadı	Program Adı	Seç
6	Mahmut	Kantar	Bilgisayar Kontrol Sistemleri	<input type="checkbox"/>

Şekil 5. 88 Ödev ekleme işlemi.

Şekil 5.88’de görüldüğü gibi sistemdeki öğretim elemanı araştırma teknikleri dersi ve fizik dersi olmak üzere iki ders vermektedir. Öğretim elemanının ödev göndermek istediği ders seçildikten sonra aşağıda o dersi alan öğrenciler listelenmektedir. Programa kayıtlı bütün öğrencilere aynı ödevi göndermek zorunda değildir. Göndermek istediği öğrencileri seçtikten sonra onay butonuna basması gerekmektedir. Bu aşamadan sonra ödev, ilan tarihi geldiğinde, öğrenci ekranındaki yaklaşan olaylar alanına ödev başlığı altında eklenir. Öğretim elemanı, ödev gönderdiği öğrencilerden iptal etmek istediği varsa, Eklenen Öğrenciler seçeneğini seçtikten sonra çıkan listeden iptal etmek istediği öğrencileri seçmesi ve iptal butonuna basması gerekmektedir. Ayrıca gönderilen ödevde değişiklikler yada ödevi silme işlemi yapılabilmektedir. Bunun Şekil 5.88’de verilen ekranın en üst kısmındaki düzenle butonu tıklandıktan sonra sistem düzenleme moduna geçmektedir. Düzenleme ve silme işlemi şekil 5.89’da verilmiştir.

**Dosya Ekleme İstediyiniz Dersi Seçiniz:** Arastirma Teknikleri ▼

	Odev No	Odev Adı	Odev İlan Tarihi	Odev Teslim Tarihi	
<a href="#">Düzenle</a>	118	Arastirmada Bilimsel Yöntem Nedir?	2/4/2009	2/1/2009	<a href="#">Sil</a>
<a href="#">Düzenle</a>	119	Bilimsel Yöntem Nedir	2/2/2009	2/2/2009	<a href="#">Sil</a>
<a href="#">Düzenle</a>	120	Nicel Gözlem Nedir	2/1/2009	2/1/2009	<a href="#">Sil</a>

Ders :

**Odev Eklenecek Ders:** Dersi Degistir

**Dosya Açıklama Bilgisi (Max. 50) :**

Odev Adı

Odev İlan Tarihi

Odev Teslim Tarihi

**Göndermek İstediyiniz Dosyayı Seçiniz (Max. 2 Mb) :**

C:\Users\Net1\Desktop\Düzeltilmis\_AnalizDokuman

**Şekil 5. 89** Ödev düzenleme ve silme işlemi.

### 5.16.2 Öğrenci Ödev Gönderme İşlemi

Öğrenci ödev ilan tarihi geldiğinde gönderilen ödevi görüntüleyebilmektedir ve ödevi hazırlamış ise ilgili öğretim elemanına hazırlanan ödevi gönderebilmektedir. Bu işlem için ödevler bölümüne erişebileceği ilgili bağlantıyı seçer ve kullanıcı ekranına aldığı derslerle ilgili ödevler listelenir. Hangi ödev gönderilecekse o ödev seçilir. Sistem tarihi ile ödevin teslim tarihi karşılaştırılır. Eğer ödevin teslim tarihi geçtiyse ödevi gönder bağlantısı kaybolur. Eğer teslim tarihi geçmediyse "ödevi gönder" bağlantısını tetikler. Ödev dosyasını ekledikten sonra gönder işlemini tetikler ve durum alanı ödev gönderildi olarak değiştirilir. Öğrenci ödevin son teslim tarihine kadar istediği kadar ödev gönderebilir. Gönderdiği son dosya sistem tarafından kabul edilir. Ancak gönderilecek ödev maksimum 2 Mb olmalıdır ve .doc, .ppt ve .zip uzantılı dosyalardan birisi olmalıdır. Öğrencinin hazırlamış olduğu ödevi ilgili öğretim elemanına gönderme işlemine Şekil 5.90'da örnek verilmiştir.

OId	OdevNo	Odev Adı	Ders Adı	İlan Tarihi	Teslim Tarihi	Gönderildimi	Ödevi Gönder	Odevi Görüntüle
6	46	Nicel Gözlem Nedir	Arastırma Teknikleri	2/1/2009	2/1/2009	■	<a href="#">Seç</a>	<a href="#">Ac</a>

**Gönderilecek Ödev : Arastırma Teknikleri Nicel Gözlem Nedir**  
**Göndermek İstedığınız Dosyayı Seçiniz (Max. 2 Mb) :**

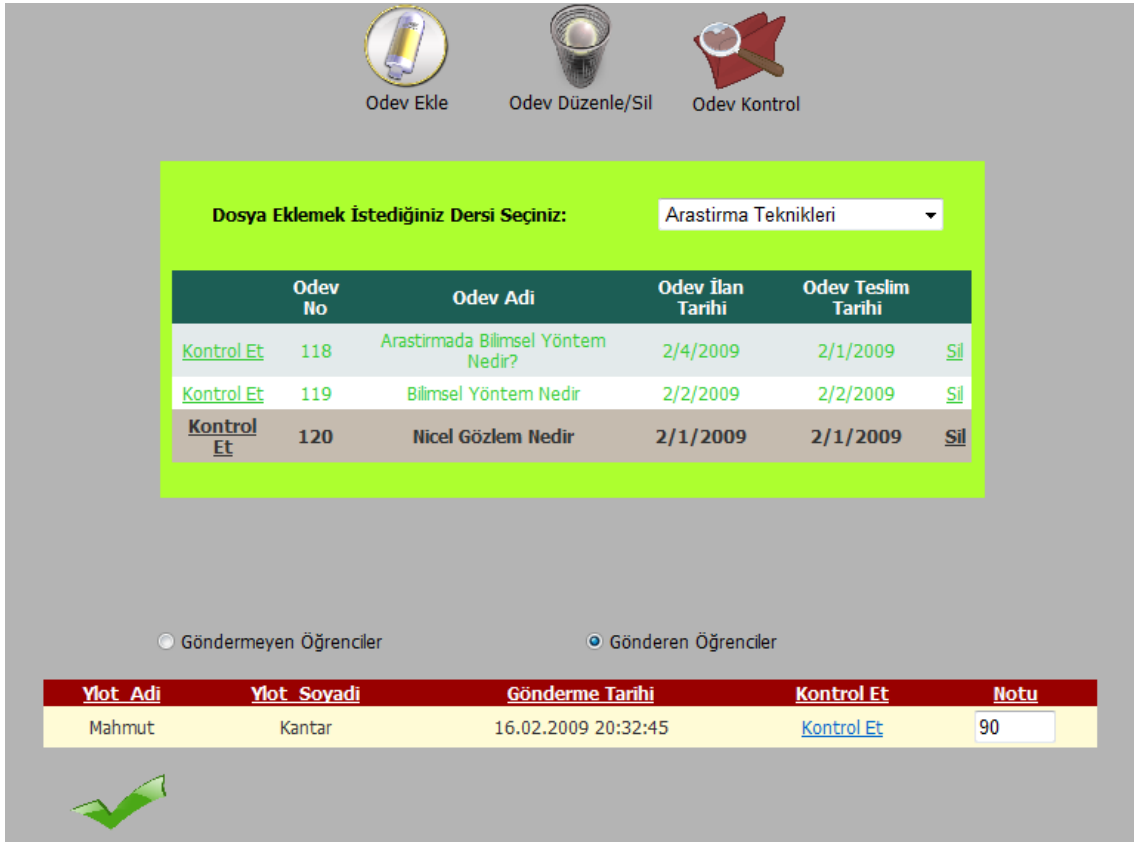
C:\Users\Net1\Desktop\Düzeltilmis\_AnalizDokuman

Şekil 5. 90 Öğrenci ödev gönderme işlemi.



### 5.16.3 Ödev Değerlendirme İşlemi

Ödev gönderme işleminin sona ermesiyle birlikte, gönderilen ödevler öğretim elemanının ekranında listelenmektedir. Öğretim elemanı ödev ilan tarihi sona ermeden ödev değerlendirme işlemi yapamamaktadır. Bunun sebebi ödevin son teslim tarihine kadar öğrencilerin ödevlerinde değişiklikler yapabilmesidir. Ödev değerlendirme işlemi yapabilmek için öncelikle öğretim elemanı ödev verdiği derslerden hangisinin kontrolünü yapacaksa o dersi seçmesi gerekmektedir. Ders seçildikten sonra derse ait son teslim tarihi dolmuş ödevler ekranda listelenir. Kontrol etmek istediği ödevi seçtikten sonra ilgili ödevi gönderen öğrenci ve ödev cevapları listelenir. Listeden ödevini kontrol etmek istediği öğrenciyi seçer ve ödevini görüntüler. Ödev kontrolünden sonra öğrencinin ödev notunu girer ve onay butonuna basar. İlgili öğrencinin ödev notu bilgilerine işlenir. Ödev değerlendirme işlemine Şekil 5.91’de örnek verilmiştir.



Odev Ekle Odev Düzenle/Sil Odev Kontrol

Dosya Ekleme İstedığınız Dersi Seçiniz: Arastirma Teknikleri

	Odev No	Odev Adi	Odev İlan Tarihi	Odev Teslim Tarihi	
<a href="#">Kontrol Et</a>	118	Arastirmada Bilimsel Yöntem Nedir?	2/4/2009	2/1/2009	<a href="#">Sil</a>
<a href="#">Kontrol Et</a>	119	Bilimsel Yöntem Nedir	2/2/2009	2/2/2009	<a href="#">Sil</a>
<a href="#">Kontrol Et</a>	120	Nicel Gözlem Nedir	2/1/2009	2/1/2009	<a href="#">Sil</a>

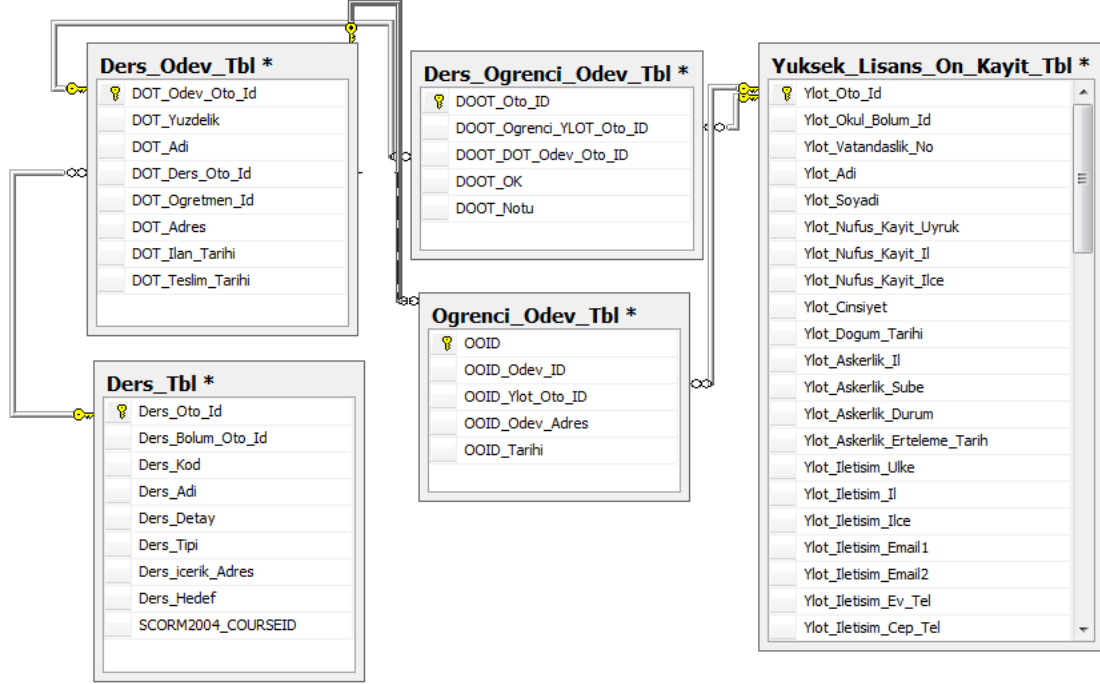
Göndermeyen Öğrenciler  Gönderen Öğrenciler

Yol Adı	Yol Soyadı	Gönderme Tarihi	Kontrol Et	Notu
Mahmut	Kantar	16.02.2009 20:32:45	<a href="#">Kontrol Et</a>	90

Şekil 5. 91 Ödev değerlendirme işlemi.

Ödevler Modülünde kullanılan tablolar ve bu tablolar ile ilişkili tablolar Çizelge 5.23’de verilmiştir.

**Çizelge 5. 21** Odevler Modülünde kullanılan tablolar ve bu tablolar ile ilişkili tablolar



Öğretim elemanının vermiş olduğu ödevlerin bilgileri Çizelge 5.23’deki Ders\_Odev\_Tbl tablosunda ödev verilen öğrenciler ve bu öğrencilerin ödev bilgileri Ogrenci\_Odev\_Tbl tablosunda, ödev notları ise Ders\_Ogrenci\_Odev\_Tbl tablosunda tutulur. Bu tablolardaki alanların açıklaması aşağıda verilmiştir.

### Ders\_Odev\_Tbl

- **DOT\_Odev\_Oto\_Id:** Ödev ID’sidir.
- **DOT\_Yuzdelik:** Ödevin not ortalamasına yapacağı yüzdelik etkisidir.
- **DOT\_Adi:** Ödevin adıdır.
- **DOT\_Ders\_Oto\_Id:** Ödevin ait olduğu ders ID’sidir. Des\_Tbl ile ilişkilidir.
- **DOT\_Ogretmen\_Id:** Ödevi veren öğretmen ID’sidir.

- **DOT\_Adres:** .doc, .ppt, .zip uzantılı dosyanın sistemdeki adresidir.  
**DOT\_Ilan\_Tarihi:** Ödevin ilan tarihidir.
- **DOT\_Teslim\_Tarihi:** Ödevin son teslim tarihidir.

#### **Ogrenci\_Odev\_Tbl**

- **OOID:** Ödev verilen her öğrenciye sistem tarafından verilen otomatik ID'dir.
- **OOID\_Odev\_ID:** Ödevin numarasıdır. Ders\_Odev\_Tbl ile ilişkilidir.
- **OOID\_Ylot\_Oto\_ID:** Öğrenci numarasıdır.
- **OOID\_Odev\_Adres:** Öğrencinin göndermiş olduğu ödev dosyasının sistemdeki adresidir.
- **OOID\_Tarihi:** Öğrencinin ödevini gönderdiği tarihidir.

#### **Ders\_Ogrenci\_Odev\_Tbl**

- **DOOT\_Oto\_ID:** Ödev kontrol numarasıdır.
- **DOOT\_Ogrenci\_YLOT\_Oto\_ID:** Kontrol edilen öğrenci ID'sidir.
- **DOOT\_DOT\_Odev\_Oto\_ID:** Ödev numarasıdır. Ders\_Odev\_Tbl ile ilişkilidir.
- **DOOT\_OK:** Kontrol işleminin tamamlanıp tamamlanmadığıdır. Kontrol edildiyse "1", Kontrol edilmediyse "0" değeri alır.
- **DOOT\_Notu:** Verilen ödev notudur.

## 5.17 Öğretim Elemanı Performans Takip Modülü

Öğretim elemanı performans takip modülü ile birlikte bölüm başkanları, bölümlerinde görevli olan öğretim elemanlarının performansını raporlayabilmektedirler. Bu raporun içerisinde öğretim elemanın düzenlediği toplam sohbet saati, sanal sınıfta düzenlediği toplam ders sayısı, verdiği ödev sayısı ve değerlendirme anketi sonucu yer almaktadır. Bu işlemler için bölüm başkanının aşağıdaki adımları takip etmesi gerekmektedir.

- Öğretim elemanı performans takibi ile ilgili bağlantı aracılığı ile sayfaya ulaşır.
- Bölümde ders veren öğretim elemanları listelenir.
- Görüntülemek istediği öğretim elemanını seçer.
- Görüntüleme işlemini tetikler.
- Öğretim elemanının yapmış olduğu faaliyetleri listeler.

Öğretim elemanı performans takip raporuna örnek Şekil 5.92’de verilmiştir.

### Mevlut Dogan Adlı Öğretim Elemanın Performans Sonuçları

ID	Adi	Soyadi	Odev Sayisi	Sohbet Saati Sayisi	Sanal Sınıf Sayisi	Değerlendirme anketi
10	Mevlut	Dogan	3	3	2	4

### Bölümünüzdeki Öğretim Üyeleri

ID	Sicil No	Adi	Soyadi	Performans
10	121	Mevlut	Dogan	<a href="#">Görüntüle</a>
11	1q2w	Omer	Sise	<a href="#">Görüntüle</a>

Şekil 5. 92 Öğretim elemanı performans takip.

Şekil 5.92’de de görüldüğü üzere sistemdeki bölüm başkanının idaresindeki bölümde görev yapan öğretim elemanları listelenmiştir. Listedeki öğretim elemanlarından Mevlut Dogan adlı kullanıcının performansı raporlanmıştır. Bu rapor içerisinde vermiş olduğu ödev sayısı, sohbet saati sayısı, sanal sınıf sayısı ve değerlendirme anketi yer almaktadır. Değerlendirme anketi 5’lik sisteme göre yapılmaktadır.

## 5.18 Öğrenci Performans Takip Modülü

Öğrenci performans takip modülü ile öğrenci bilgileri raporlanmaktadır. Böylece öğrenciler dönem içerisindeki başarı durumunu değerlendirebilmektedirler. Raporlarda bulunan veriler ise şunlardır.

- Öğrenci vize notları
- Final notları
- Ödev notları
- Sohbet katılım saatleri
- Derse katılım saatleri ve ders sorumlusunun vermiş olduğu performans notları

Bu başarı notlarının bazıları SCORM tablolarından çekilmektedir. Bunlar Vize, Final notları ve derse katılım saatleridir. Hazırlanan öğrenim yönetim sistemi SCORM uyumlu olmasından dolayı ileriye dönük bir sistemdir. İleriki aşamalarda bu raporlar daha geniş bir biçimde geliştirilebilir. Örneğin şimdilik öğrencinin kaç defa ders içeriklerini izlediğinin sayısı verilmiştir. Bu bilgiler daha da detaylandırılarak öğrencinin hangi dersi ne kadar süre takip ettiği, en çok hangi ekranları izlediği, ekranlar arasındaki geçiş hızı vb. gibi bilgilerde raporlanabilir. Bu bilgiler sistemdeki SCORM API tarafından detaylı bir şekilde kaydedilmektedir. Öğrenci verilerinin raporlanmasına örnek Şekil 5.93’de verilmiştir.

## Performansınız

### Final Smavı Performansı

Ders	Final	Notu
Arastırma Teknikleri	Final Notu	95
Fizik	Final Notu	85

### Vize Smavları Performansı

Ders	Vize	Notu
Arastırma Teknikleri	Vize1	85
Arastırma Teknikleri	Vize2	90
Fizik	Vize1	100
Fizik	Vize2	50

### Derse Katılım Performansı

Ders	Katılım Tarihi
Arastırma Teknikleri 3	25.02.2009
Arastırma Teknikleri 2	31.01.2009
Arastırma Teknikleri 2	15.01.2009
Fizik 1	31.01.2009
Fizik 2	31.02.2009

### Odev Performansı

Ders	Odev	Notu
Arastırma Teknikleri	Arastırmada Bilimsel Yöntem Nedir?	75
Arastırma Teknikleri	Bilimsel Yöntem Nedir	50
Fizik	Kimyasal Tepkime Nedir?	65
Fizik	Ataomun Yapısı	75

Şekil 5. 93 Öğrenci performans raporu.

### Ders Performans Notu

Adi	Ogretmen	Performans Notu
Arastirma Teknikleri	Mevlut DOGAN	85
Fizik	Omer Sise	80

### Sohbet Katılım Performansı

Sohbet	Tarih
E-fizik konulu konferans	24.03.2009
Materyal Geliştirme Teknikleri	20.02.2009

**Şekil 5.93(devam)** Öğrenci performans raporu.

Öğrenci performans notunu görüntülemek istediğinde Şekil 5.93'te görüldüğü gibi rapor sayfası çıkmaktadır. Bu rapor sayfasında öğrenci vize notları, final notları, ödev notları, sohbet katılım saatleri, derse katılım saatleri ve ders sorumlusunun vermiş olduğu performans notları görüntülenmektedir.

## 5.19 Öğrenci Takip Raporu

Uzaktan eğitimin başarı ölçüsü öğrenciyi takipteki başarıyla ölçülür. Öğrencinin derslerde ne kadar başarılı olduğu, dersleri ne kadar iyi anladıkları, kendilerini ne kadar geliştirdikleri, içeriklerin yeterli olup olmadığını ancak takip araçlarıyla ölçülebilir. Öğrenci çok iyi takip edilmeli ve öğrencinin nerde takıldığı nereyi anlamadığı tespit edilmelidir. Tespit edilen eksiklikler hemen giderilmeye çalışılmalı ve yeni teknikler geliştirilmelidir. Bilgisayar teknolojisi her türlü yeniliğe açık olduğu için bu eğitimde bu teknolojinin tüm imkanlarını kullanmak çok yerinde olacaktır (Bayram ve ark. 2002).

Bu modülde öğretim elemanı ders verdiği şubelerdeki öğrencileri performans notu ile değerlendirebilmektedir. Performans notu öğrencinin not ortalamasını etkilemektedir. Ancak bu başarı notunu ne kadar etkileyeceği bölüm sekreteri tarafından belirlenmektedir. Bu işlemi yapabilmek için kullanıcı sisteme giriş yapar. İlgili bağlantı aracılığı ile öğrenci değerlendirme sayfasına ulaşır. Öğretim üyesinin verdiği dersler listelenir. Listesinden dersi seçer ve dersin şubeleri listelenir. Önce şubeyi seçer sonra seçilen şube içerisindeki öğrenciler listelenir. Listedeki performansını görüntülemek istediği öğrenciyi seçer ve öğrenci bilgileri ekranda raprolanır. Ropradaki bilgiler şunlardır;

- Vize notu
- Final notu
- Ödev notları
- Derse katılım saati sayısı
- Sohbet saati sayısı

Öğretim elemanı rapordaki performansa göre öğrencinin performans notunu girerek değerlendirmeyi kaydet işlemini tetikler. Öğrenci performans raporu dönem sonunda verilebilmektedir. İlan edilen sınav tarihleri sonunda öğretim elemanı sisteme giriş yaklaşan olaylar menüsünden bu bölüme ulaşabilmektedir. Öğrenci performans raporuna örnek şekil 5.95’de verilmiştir.



## Seçilen Öğrenci : Emin İBİLİ

### Final Sınavı Performansı

Ders	Final	Notu
Arastirma Teknikleri	Final Notu	95
Fizik	Final Notu	85

### Vize Sınavları Performansı

Ders	Vize	Notu
Arastirma Teknikleri	Vize1	85
Arastirma Teknikleri	Vize2	90
Fizik	Vize1	100
Fizik	Vize2	50

### Derse Katılım Performansı

Ders	Katılım Tarihi
Arastirma Teknikleri 3	25.02.2009
Arastirma Teknikleri 2	31.01.2009
Arastirma Teknikleri 2	15.01.2009
Fizik 1	31.01.2009
Fizik 2	31.02.2009

### Odev Performansı

Ders	Odev	Notu
Arastirma Teknikleri	Arastirmada Bilimsel Yöntem Nedir?	75
Arastirma Teknikleri	Bilimsel Yöntem Nedir	50
Fizik	Kimyasal Tepkime Nedir?	65
Fizik	Ataomun Yapısı	75

### Sohbet Katılım Performansı

Sohbet	Tarih
E-fizik konulu konferans	24.03.2009
Materyal Geliştirme Teknikleri	20.02.2009

### Ders Performans Notu

Öğrenci Adı	Basarı Notu
Emin İBİLİ	85

Şekil 5. 94 Öğrenci performans notu.

## 20.-37. Modüller

Bir birine bağımlı olarak hazırlanan 3 tez çalışmasından oluşan bu çalışma, uygulanabilir bir uzaktan öğretim yönetim sistemi (LMS) ve bu LMS içerisinde uygulanabilir olan bir dersin senaryosunu ve ders materyallerini içermektedir. Bahsedilen modüller öncelikle uzun süren bir analiz aşamasından geçirilerek üniversitenin işleyiş sistemi öğrenilerek kullanılacak veritabanları oluşturulmuştur. Analiz aşamasında tüm detaylar en ince ayrıntısıyla irdelenmiştir. Bu nedenle ortaya çıkarılan analiz metni kodları yazılacak modüllerin algoritması olarak kullanılmıştır. Analiz raporlarının hazırlanması aşamasında modüler bir sistemin tasarlanması hedeflenmiş ve bu doğrultuda analiz raporları hazırlanmıştır. Bu süreç sonucunda toplam 37 modülden oluşan bir analiz raporu oluşturulmuş ve Öğretim Yönetim Sisteminin yazılması ise bu analiz raporuna göre hazırlanmıştır. Oluşturulan modüllerin ilk 19 modülünün kodlaması bu tez paralelinde yapılmış ve bu çalışmada detaylı bir şekilde anlatılmıştır. Bu modüllerin analiz raporu ise EK-2’de verilmiştir. Diğer modüllerin kodlaması ise yüksek lisans tez Fatih BAYRAM tarafından yapılmıştır (Bayram 2009). EK-1’de Fatih Bayram tarafından gerçekleştirilen modüller özet olarak anlatılmıştır. Bu modüller ile ilgili detaylı bilgi ise Bayram (2009) yüksek lisans tezinde yer almaktadır.

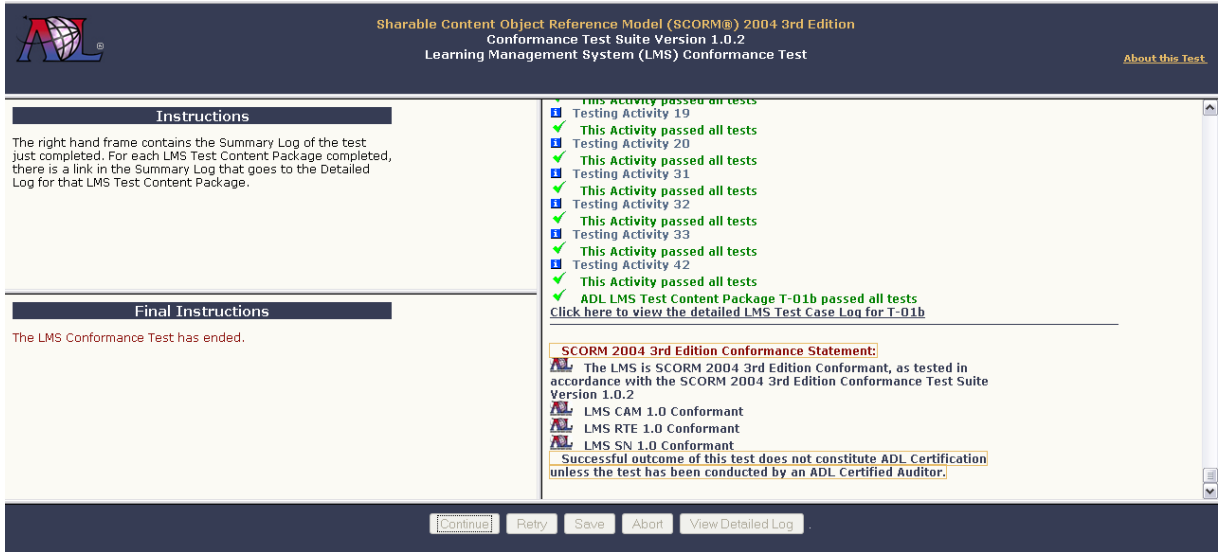
## 6. SONUÇ

21. yüzyılın en önemli özelliği bilgiyi tek bir kaynaktan öğrenme yerine, çoklu ortamlardan bilgiye ulaşma ve ulaşılan bu bilgiyi değerlendirme, zaman ve mekân sınırı tanımadan diğerleri ile paylaşma, tartışma ve gerekli durumlarda bu bilgiyi gerçek yaşamda kullanabilmeyi gerektirmektedir. Bu nedenle, eğitim yönetim sistemleri, sözü edilen tüm bu yaşantıları kullanıcılara sunabilen sanal iletişim ve öğrenme ortamı haline gelebilmelidir. Bu bağlamda, en önemli konu, sanal eğitim ortamlarının yönetimi ve değerlendirilmesi olmaktadır.

Uzaktan eğitim yönetim sistemleri, eğitim seviyesinin yükselebilmesi ve bu hedefe ulaşabilmek için çok elverişli, ekonomik ve uygulanabilirlik açısından en mümkün olan seçeneklerden birisidir. Fakat yapılan her türlü eğitim toplumun çeşitli kesimleri tarafından, iş çevrelerinden ve diğer kurumlar tarafından kabul edilebilir olması için bazı standartlara uygun olması gerekmektedir. Bunun sağlanması için verilen eğitimin ve eğitim platformunun gerek tasarım, gerek içerik ve gerekse yazılımın kalitesi açısından tatmin edici düzeyde ve genel kabul görmüş bir eğitimi bize sağlıyor olması gerekmektedir. Bu bağlamda, en önemli konu, sanal eğitim ortamlarının yönetimi ve değerlendirilmesi olmaktadır. Bu eğitim ortamlarının yönetimi kullanılan yazılım alt yapısı ve işlerliği ile paralel bir şekilde ilerlemektedir. Bununla birlikte SCORM API yazılımı ile LMS son derece uyumlu bir şekilde çalışması gerekmektedir.

Uzaktan eğitim sistemlerinde başarıyı yakalamak, sadece bu sistemlerin yazılım ve teknoloji alt yapısına bağlı değildir. Uzaktan eğitimde, eğitimin kalitesini artırmak aynı zamanda öğretim sistemleri tasarımı yaklaşımı ile planlandığı, kullanıldığı, geliştirildiği ve değerlendirildiği sürece başarılı olacaktır. Uzaktan eğitim sistemlerinin eğitim kalitesi aynı zamanda kaliteli öğretmenlerin, mühendislerin, öğretim tasarımcısı ve teknologlarının ve konu uzmanı bireylerin meydana getirdiği öğretim tasarımı gruplarının birlikte çalışmalarına bağlıdır. Bu çalışmalar gücünü, öğrenme ve öğretim teori ve modelleri ile teknolojik gelişmelerden almak durumundadır.

Yazılımı tamamlanan Uzaktan eğitim yönetim sistemi için bazı derslerin SCORM paketleri hazırlanmıştır. Ayrıca LMS tasarımı yapılmış ve toplam 36 modülden oluşan LMS'in analiz dokümanları ve SQL veri tabloları olusturulmuştur. Geliştirilen LMS modüler bir sistem üzerine kurulmuştur. Yaklaşık 200 öğrenci aynı anda sisteme kaydedilmiş ve bu öğrencilere Araştırma Teknikleri dersini sistem üzerinden ek olarak alma imkanı sunulmuştur. LMS'nin SCORM 2004 3<sup>rd</sup> Ed. uyumluluk testi için SCORM standartlarını belirleyen ADL tarafından sağlanan "SCORM 2004 3rd Edition Conformance Test Suite Version 1.0.2 Self Test" kullanılmıştır. SCORM 1.2 için ise SCORM 1.2 Edition Conformance Test Suite Version 1.0.2 Self Test kullanılmış ve her iki testte başarılı bir şekilde sonuçlanarak Şekil 5.95'deki sonuç elde edilmiştir. Testlere başlamadan önce ADL tarafından sağlanan 103 adet SCORM 2004 test paketi sisteme kurulmuştur. Test sırasında ders paketlerini kullanmak üzere iki tane kullanıcı hesabı (Test1 ve Test2) oluşturulmuştur.



Sharable Content Object Reference Model (SCORM®) 2004 3rd Edition  
Conformance Test Suite Version 1.0.2  
Learning Management System (LMS) Conformance Test

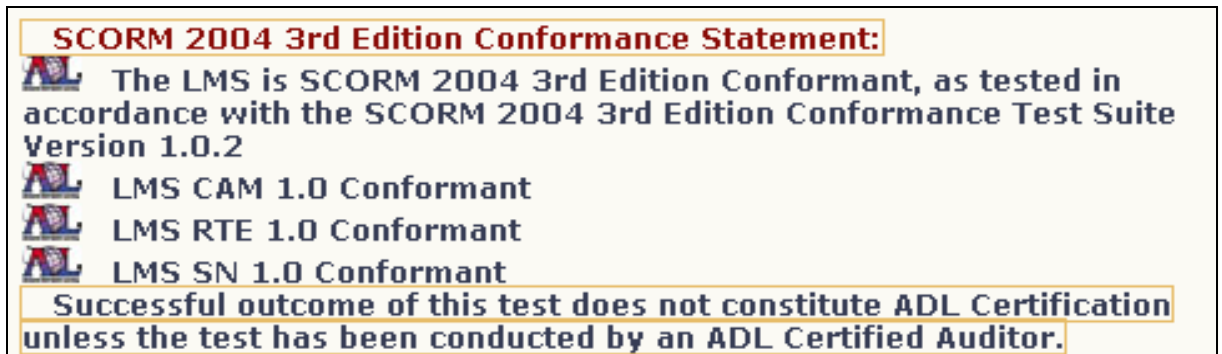
**Instructions**  
The right hand frame contains the Summary Log of the test just completed. For each LMS Test Content Package completed, there is a link in the Summary Log that goes to the Detailed Log for that LMS Test Content Package.

**Final Instructions**  
The LMS Conformance Test has ended.

This Activity passed all tests  
Testing Activity 19  
This Activity passed all tests  
Testing Activity 20  
This Activity passed all tests  
Testing Activity 31  
This Activity passed all tests  
Testing Activity 32  
This Activity passed all tests  
Testing Activity 33  
This Activity passed all tests  
Testing Activity 42  
This Activity passed all tests  
Testing Activity 42  
This Activity passed all tests  
ADL LMS Test Content Package T-01b passed all tests  
Click here to view the detailed LMS Test Case Log for T-01b

**SCORM 2004 3rd Edition Conformance Statement:**  
The LMS is SCORM 2004 3rd Edition Conformant, as tested in accordance with the SCORM 2004 3rd Edition Conformance Test Suite Version 1.0.2  
LMS CAM 1.0 Conformant  
LMS RTE 1.0 Conformant  
LMS SN 1.0 Conformant  
Successful outcome of this test does not constitute ADL Certification unless the test has been conducted by an ADL Certified Auditor.

Continue Retry Save Abort View Detailed Log



**SCORM 2004 3rd Edition Conformance Statement:**  
The LMS is SCORM 2004 3rd Edition Conformant, as tested in accordance with the SCORM 2004 3rd Edition Conformance Test Suite Version 1.0.2  
LMS CAM 1.0 Conformant  
LMS RTE 1.0 Conformant  
LMS SN 1.0 Conformant  
Successful outcome of this test does not constitute ADL Certification unless the test has been conducted by an ADL Certified Auditor.

Şekil 5. 95 Test Sonuçları.

Şekil 5.95’de de görüldüğü üzere test başarılı sonuç vermiştir. Test sırasında denenen tüm paketler başarıyla tamamlanmıştır. Test sonuçlarını içeren XML dosyaları, bir zip dosyası olarak ekte verilmiştir.

SCORM test uyumluluğu için kullanılan bilgisayar konfigürasyonu şunlardır;

- Internet Explorer 7.0
- Java RTE 1.6
- Windows XP Service Pack 2
- Intel Core2 Duo 2.2 GHZ
- 2 GB RAM
- 17” monitör

Sistemin ADL sertifikasyonunu alabilmesi için ADL tarafından yetkilendirilmiş bir kurum tarafından test edilmesi gerekmektedir. Şekil 5.95’de de görüldüğü gibi yetkili bir kurumun da yapacağı test aynı testtir ve gerekli ücret ödenmesi durumunda yapılacak bu test sonucu sistemin ADL sertifikası alınabilmektedir.

## 7. KAYNAKÇA

- Althoff, K., Althoff ,D., Dengel, A., Bergmann, R., Nick, M., Roth Berghofer, T., 2005, "Professional Knowledge Management", Springer, 209-214
- Bayram, F., 2009, "Scorm Uyumlu Modüler Öğretim Yönetim Sisteminin Üniversitede Kullanımı", Afyon Kocatepe Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Afyonkarahisar.
- Bayam, Y., Urin M. 2002, "Uzaktan Eğitimde Öğrenci Takibi ve Değerlendirmesi", Açık ve Uzaktan Eğitim Sempozyumu, Eskişehir.
- Codone, S., 2001, "E-Learning An Primer", Raytheon Interactive Pensacola, Florida
- Çağiltay, K., 2001, "Uzaktan Eğitim: Başarıya Giden Yol Teknolojide mi Yoksa Pedagojide mi?"; <http://www.teknoturk.org/docking/yazilar/tt000037-yazi.htm> (12/07/2009).
- Çetiner, M., Gencil, Ç., Erten, Y., 2004, "ODTÜ Enformatik Enstitüsü, Bilişim Sistemleri İnternete Dayalı Uzaktan Eğitim ve Çoklu Ortam Uygulamaları" <http://inet-tr.org.tr/inetconf5/bildiri/Cetiner.html>(12/07/2009).
- Doruk, Z. 2006, "Nesne Tabanlı e-Öğrenme Yazılımları için Bir Basvuru Modeli:SCORM". Erisim : <http://e-learningtalks.com/2006/05/07/nesnetabanli-e-ogrenme-yazilimleri-icin-bir-basvuru-modeli-scorm/> (02.6.2009)
- Downing F.K., Holtz K.J, 2008, "Online Science Learning: Best Practices and Technologies", DePaul University, USA
- Elmas, Ç., Doğan, N., Biroğul, S., Koç, M.S., 2008, "Moodle Eğitim Yönetim Sistemi İle Örnek Bir Dersin Uzaktan Eğitim Uygulaması", İnternet Teknolojileri ile Uzaktan Eğitim Sempozyumu, Gazi Üniversitesi, Ankara
- Fer, S., 2000, "Modüler Program Yaklaşımı ve Bir Öneri", Milli Eğitim Dergisi, 147, 21- 37.
- Ford N., 2008, "Web-based Learning Through Educational Informatics: Information Science Meets Educational Computing", IGI Publishing, 216-262
- Hakkari, F., 2009, "Uzaktan Eğitimde Materyal Hazırlamada İçerik Tasarımı, Senaryo Hazırlama Ve Uygulaması", Afyon Kocatepe Üniversitesi, Fen Bilimleri Enstitüsü, , Yüksek Lisans Tezi, Afyonkarahisar.

- Iskander M., 2007, "Innovations in E-learning, Instruction Technology, Assessment, and Engineering Education", Spring, 347-349
- İşık, İ., Hakan, A., Güler, İ., 2008, "Uzaktan Eğitimde Üç Boyutlu Web Teknolojilerinin Kullanılması", Bilişim Yeknolojileri Dergisi, 75-77
- İbili, E., Bayram, F., AKBAŞ, Ü., Orhan, Z., Kantar, M., Hakkari, F., Doğan, M., 2008, Uzaktan Eğitim Sistemlerinde Yazılım Ve İçerik Oluşturma, 2. Uluslar arası Gelecek İçin Öğrenme Alanında Yenilikler Konferansı, İstanbul
- Ostyn, C., 2007, "The Eye of the SCORM", Erişim: [http:// ww.ostyn.com/resources.htm](http://ww.ostyn.com/resources.htm) (06.05.2008)
- O'Neil H. F., Perez R.S., 2008 "Advances in E-Learning: Experiences and Methodologies", IGI Publishing, 150-166
- Özmen H., 2006, "E-sınav Sistemi", Akademik Bilişim-2006, Denizli
- Kantar, M., Hakkari, F., Boy, Y., Bayram, F., İbili, E., Doğan, M., 2008, "Uzaktan Eğitim Sistemlerinde Yazılım Ve İçerik Oluşturma", 2. Uluslar arası Gelecek İçin Öğrenme Alanında Yenilikler Konferansı, İstanbul
- McClure W., Glavich P., Cate S., Orr C.S., Shoemaker C., Smith A.S., Zimmerman J., 2007, "ASP.NET 2.0 AJAX: Written and Tested with the Final 1.0 Release Version of ASP.NET AJAX for ASP:NET 2.0", Wrox Publishing, 115-153
- Nagel C., Evjen B., Glynn j., 2008, Watson K., Skinner M., 2008, "Professional C# 2008", Wiley Publishing, 1017-1385
- Satzinger W.J, Jackson Robert B., Burd D.S., 2009, "Systems Analysis and Design in a Changing World", Cengage Learning, 314-568
- Schaefer K., Cochran J., Forsyth S., Baugh R., Everest M., Glendenning D., 2008, "Professional IIS 7", Wiley Publishing, India
- Schaefer K., Cochran J., Forsyth S., Baugh R., Everest M., Glendenning D., 2008, "Professional SQL Server 2005 Performance Tuning", Wiley Publishing, India
- MacDonald M., Szpuszta M., 2008, "Pro ASP.NET 3.5: In C# 2008", Apress Publishing, 259-587
- Tai L., 2008, "Corporate E-Learning", Oxford University Press, USA
- Toprakçı, E., 2008, "Uzaktan Öğretimde Öğretmen Rollerini", II. Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu, Ege Üniversitesi Eğitim Fakültesi, İzmir

## İnternet Kaynakları

## Erişim Tarihi

- 1) <http://uzaktanegitim.blogspot.com/> 21.07.2009
- 2) [www.moodle.org](http://www.moodle.org) 21.07.2009
- 3) <http://www.advancity.net> 21.07.2009
- 4) [www.json.com](http://www.json.com) 21.07.2009
- 5) [www.adl.org](http://www.adl.org) 21.07.2009
- 6) <http://www.mmistanbul.com/makale/title/scorm-2004un-bolumlerine-genel-bir-bakis> 21.07.2009
- 7) [www.microsoft.com](http://www.microsoft.com) 21.07.2009
- 8) [www.csharpnedir.com](http://www.csharpnedir.com) 21.07.2009
- 9) <http://www.yazgelistir.com> 21.07.2009



## 8. EK-1

### 20.-37. MODÜLLER

Hazırlanan Öğrenim Yönetim Sisteminin ilk 19 modülü tez raporunda ayrıntılı olarak anlatılmıştır. 20. modül ile 37. modüller ise EK-2 olarak sunulan bu raporda özet bilgileri ile birlikte verilmiştir. Detaylı bilgiler ise Fatih BAYRAM(Bayram 2009)'ın hazırlamış olduğu yüksek lisans tezinde yer almaktadır.

#### 20. Sözlük Modülü

Sözlük modülü ile sistemdeki kullanıcılar (öğrenci, öğretim üyesi yada diğer kullanıcılar) sözlük oluşturma, sözlük ekleme yada var olan sözlüğe sözcük ekleme gibi işlemler yapmaktadır. Sözlük modülü ile öğrencilerin bilmediği terim yada sözcüklere kısa sürede ulaşması sağlanarak eğitim süresi daha verimli kullanılmaktadır. Bu modülde sisteme giriş yapan admin seviyesindeki kullanıcılar sözlük modülü ile ilgili aşağıdaki işlemleri yapabilmektedirler (Bayram 2009).

- Yeni oluşturulan sözlük sisteme eklenmekte ve diğer kullanıcılar tarafından erişilebilir hale getirilmektedir.
- Yeni kayıt yapılan sözcüğün sisteme eklenmesi sağlanmaktadır ve diğer kullanıcılar bu sözcüğe erişebilmektedir.
- Bilgileri güncellenmek istenen sözcüğün yeni girilen değerler ile sisteme kaydedilmesi sağlanmaktadır.
- Sistemden silinmek istenen sözcüğün programda bir daha gözükmeyecek şekilde sistemden kaldırması sağlanmaktadır.
- Sisteme import edilen verilerin sisteme eklenmesi ve kullanıcılar tarafından erişilebilir olması sağlanmaktadır.

Yeni sözlük oluşturmak için ilk olarak sistemde sözlük kategorisinin oluşturulması gerekmektedir. Örneğin Almanca-İngilizce, İngilizce-Türkçe, bilgisayar terimleri sözlüğü vs. Bu işlemleri yapabilmek için kullanıcı Sözlük menüsü aracılığı ile sözlük yapısında yapabileceği işlemler sayfasına ulaşır. Yeni sözlük oluştur butonuna tıklar. Açılan pencereden oluşturacağı sözlük kategorisinin adını girerek ekle butonuna tıklar. Aynı ada sahip sözlük bulunmuyorsa sözlük sistemdeki sözlükler arasına eklenir. Aksi takdirde kullanıcıya hata mesajı döndürülür.

Sözcük import alanı ile sistemde var olan sözlükteki sözcüklerin arasına yeni sözcükler eklenmektedir. Bu işlem iki farklı şekilde yapılmaktadır.

1. Sözcük Excel yada XML dosyasından eklenecekse, önce eklemek istediği sözlüğün adını veya adın bir bölümünü ilgili alana girerek arama yapar. Arama sonucunda bulunan sözlük adları listelenir ve listeden sözcük Import edeceği sözlüğü seçer. Import edeceği dosya Excel dosyası ise, bu dosya iki kolondan oluşmalıdır. İlk kolon sözcük isimlerini, ikinci kolon anlamlarını içermelidir. Import edeceği dosya XML dosyası ise her düğüm iki alt düğümden oluşmalıdır. İlk düğüm sözcük ismini, ikinci düğüm sözcük anlamını içermelidir. Dosya seçiminin ardından import işlemini tetikler ve dosya belirtilen formata uygunsa sözlüğe eklenir, uygun değilse kullanıcıya hata mesajı döndürür.
2. Sisteme girilecek sözcükler Excel yada xml dosyasından girilmeyecekse, sözcüğün adını ve anlamını girerek yeni sözcük ekle işlemini tetikler.

## **21. Anket Modülü**

Uzaktan eğitim sistemlerinde, hızlı dönüt alabilmenin en iyi yollarından birisi de anket formlarıdır. Bu formlar ile öğrencilerden, öğretim elemanlarından yada diğer kullanıcılardan yanıtlar alarak, ankette çıkan sonuçlara göre AR-GE faaliyetlerinin yürütülmesi amaçlanmaktadır. Bu amaçla uzaktan eğitim sisteminde anket modülü oluşturulmuştur. Anket modülünde admin ve user olmak üzere iki farklı seviye

bulunmaktadır. Sistemdeki admin kullanıcıları öğretim üyesi, sistem yöneticisi ve bölüm sekreteridir. User seviyesindeki kullanıcılar ise sistemdeki öğrenciler yada misafir ziyaretçilerdir. Sisteme yeni bir anket eklemek için admin seviyesindeki kullanıcısının anket bölümü ile ilgili bağlantı aracılığı ile anket tanımlama sayfasına ulaşması gerekmektedir. Anket sayfasındaki anket oluştur linkine tıklayarak aşağıdaki anket bilgilerini tanımlaması gerekmektedir (Bayram 2009).

1. Anket Adı verisini doldurur.
2. Anketin başlangıç tarihi doldurur.
3. Anketin bitiş tarihi doldurur.
4. Anketin aktif/pasif durumunu seçer.
5. Arama sonuçları arasından anketi cevaplamasını istediği kullanıcı ve kullanıcı gruplarını seçebilir.
6. Anket oluştur işlemini tetikler.
7. Aynı ada sahip anket sistemde varsa kullanıcıya hata mesajı verilir, yoksa anket tanımlanıp soru ekleme bölümü gösterilir.

Anketin sistemde başarılı bir şekilde oluşturulmasının ardından ankette yer alacak sorular eklenir. Anket soru ekleme bölümünde iki farklı tipte soru tanımlanabilmektedir.

1. Çoktan tek seçimli soru
2. Çoktan çok seçimli soru

Eklenecek soru metni girildikten sonra anket tipi seçilir ve seçenek ekleme işlemine geçilir. Bu aşamada kullanıcı, seçenek metnini girer ve seçenek ekle işlemini tetikler. Seçenek metni soruda varsa kullanıcıya hata mesajı döndürülür. Yoksa seçenek soruya eklenir. Seçenek sayısı 5'e kadar ise kullanıcı standart sekmesini tıklayarak en fazla 5 adet standart seçenek ekleyebilir. Kullanıcının eklemek istediği seçenek 5'in üzerinde ise gelişmiş kısmını tıklayarak istediği kadar seçeneği sisteme ekleyebilir.

## 22. Kişisel Notlar Modülü

Kişisel notlar modülü öğrencilerin yada sistemdeki diğer kullanıcıların ders anında yada sonrasında önemli noktaları not edebileceği bir modüldür. Öğrenciler kişisel notlar modülüyle notlarını sisteme girebilmektedir ve gerektiğinde bu notları değiştirebilmektedir. İstendiği takdirde notlar diğer kullanıcılara mesaj yoluyla iletilebilmektedir. Kişisel notlar modülünde not ekleme işlemi 2 aşamada gerçekleşmektedir. Öncelikle oluşturulacak notlar kategoriler halinde tutulmalıdır. Örneğin bilgisayar dersinde tutulacak notlar bilgisayar ders notları kategorisi gibi bir başlık altında toplanmalıdır. Bu işlemi gerçekleştirebilmek için not bölümü ile ilgili bağlantı aracılığı ile notlar sayfasına ulaşılır. Kategori adı ekle işlemini tetikler. Aynı anda ait kategori yoksa kategori sisteme eklenir. 2. Aşamada eklemek istediği notu girer. Bu işlem için yeni not ekleme linkini tıklar. Açılan sayfadan notun ekleneceği kategoriyi seçer. Not başlığını ve not metnini girerek not ekleme işlemini tetikler.

İstenildiği takdirde sisteme eklenen bu not mesaj olarak gönderilebilir. Bunun için önce gönderilecek not seçilir, daha sonra gönderilecek kullanıcı ve kullanıcı grubu seçildikten sonra mesaj gönderme işlemi tetiklenir. Kaydedilen bu notlar sistemde "LMS Ders Notlarım" adında varsayılan olarak açılan bir kategori altında tutulur (Bayram 2009).

## 23. Mesajlaşma Modülü

Mesajlaşma modülü, kullanıcılar arası mesajlaşma işlemlerin yapıldığı modüldür. öğrenci, öğretim üyesi, sistem yöneticisi, bölüm sekreteri ve bölüm başkanı bu modüldeki aktif kullanıcılarıdır. Mesajlaşma işlemi öğrenci-öğrenci, öğrenci-öğretim üyesi, bölüm başkanı-bölüm sekreteri arasında gerçekleşmektedir. Sistem yöneticisinde gereken zamanlarda, bazı duyuruları mesaj yoluyla ilgili kullanıcılara gönderebilmektedir. Mesaj gönderme işlemi aşağıdaki şekilde yapılmaktadır (Bayram 2009).

- Kullanıcı veya kullanıcı grubu arar.

- Arama sonuçlarından mesaj göndereceği kullanıcıları veya kullanıcı gruplarını seçer.
- Mesajın konusunu girer.
- Mesajla beraber dosya gönderecek ise dosyayı ekler.
- Gönderilecek dosya boyutu maximum 1.5 MB' dır.
- Gerekli ise "Okundu bilgisi al" seçeneğini seçer.
- Mesajın tam metnini girer.
- Mesajı Gönder i tetikler.
- Kullanıcı mesaj ve kapasite kotalarını aşmadı ise mesaj gönderilen mesajlara ve gönderilen kişinin gelen mesajlar kısmına kaydedilir.

Kullanıcı gelen mesajları okuyabilmek için Gelen Mesajlarım bölümüne girer. Gelen mesajlar tarih sırasıyla ve konuları ile listelenir. Kullanıcının isteğine göre mesajlar içerisinde belirli tarihe yada belirli konu başlığına göre arama yapılabilir. Görüntülemek istediği mesajı seçer ve mesaj görüntülenir. Ayrıca kullanıcı Gönderdiğim Mesajlarım kısmına girerek daha önce göndermiş olduğu mesajları görüntüleyebilir. Kullanıcı mesaja cevap vermek istiyorsa mevcut mesajlar arasından cevap yazılacak mesajı seçer. "Kime" hücresinde hangi kullanıcı bu mesajı göndermiş ise o kullanıcı adı yazılmış olarak gelir. Mesaj metni doldurulduktan sonra mesaj gönder işlemi tetikler ve mesajı gönderen kişiye cevap ulaşır.

## **24. Dosyalar Modülü**

Kullanıcılar arası dosya transferinin yapıldığı modüldür. Bu modüldeki aktif kullanıcılar öğrenci ve öğretim üyesidir. Öğretim üyesi dosyalar modülü aracılığıyla vermiş olduğu derslerle ilgili dokümanları sisteme ekleyerek dersi alan kişilerin bu dokümanlara erişebilmesini sağlamaktadır. Öğretim üyesi dosya transferini aşağıdaki şekilde yapmaktadır (Bayram 2009).

- Dosyanın açıklama bilgisini girer.
- Ekleyeceği dosyayı seçer (Dosya boyutu max. 2 MB).

- Dosya boyutu fazla ise kullanıcıya hata uyarısı verilir.
- Gönder işlemini tetikler.
- Kullanıcı arşiv kotasını(100 MB) aşıyorsa uyarı mesajı verilir aşmıyorsa gönderilen bilgiyle birlikte dosya boyutu ve gönderim tarihi sistemden alınır ve kayıt bilgileri tutulur.

Gönderilen dosya, ders dosyaları arasına eklenerek kullanıcının kota bilgisi güncellenir. Kullanıcılar kendi kişisel notlarını da bu modülde tutabilmektedir. Yaklaşık 50 Mb' a kadar dosyayı sistemde tutabilirler.

Kullanıcılar sistemdeki dosyalar arasından arama işlemi yapabilmektedirler. Arama işlemi için aramak istediği kriterleri belirleyerek dosya açıklamasına göre yada dosya ismine göre arama yapabilirler. Arama işleminde kullanılacak kriterler ise şunlardır.

- Girilen metin ile başlayanları,
- İçerisinde girilen metin geçenleri,
- Girilen metin ile bitenleri arayabilir.

## **25. Sohbet Saatleri Modülü**

Öğrencilerin belirli zamanlarda öğretim elemanları ile bir araya gelerek, sormak istedikleri soruları iletebildiği ve öğretim elemanının cevapladığı modüldür. Sohbet saatleri öğretim elemanı tarafından tanımlanmaktadır. Sohbet saati tanımlanırken, belirtilen saatin boş olmasına dikkat edilmelidir. Aksi takdirde sohbet saatinin diğer sohbet saatleri ile çakışması bazı öğrencilerin yada öğretim elemanının sohbe katılmasına engel olabilir. Sohbet saati tanımlamak için aşağıdaki işlemler takip edilir (Bayram 2009).

- Hangi programdaki öğrenciler için sohbet oluşturulmak isteniyorsa o program seçilir.
- Verdiği dersler arasından dersi seçilir.

- Programa ait haftalık ders programı görüntülenir.
- Sohbetin haftanın hangi günü olacağını belirlenir (Günün o saatinin boş olmasına dikkat etmelidir).
- Sohbet saati girilir.
- Sohbet konusu girilir.
- Sohbeti bildirmek için kullanıcılara gönderilecek mesajın içeriği belirlenir.
- Sohbet oluşturma işlemi tetiklenir.

Dersi alan kullanıcılara mesaj gönderilir ve kullanıcıların ajandalarına olay olarak eklenir. Gönderilen mesajın konusu “DERS ADI + sohbeti” olarak tanımlanmaktadır. Kullanıcı eklemiş olduğu sohbeti silmek istiyorsa önce vermiş olduğu sohbetler içerisinde silinecek sohbeti seçer ve silme işlemini tetikler. Şayet sohbet tarihi o anki tarihle aynı değilse sohbet silinir, aynı ise kullanıcıya hata döndürülür. Sohbet olayı kullanıcıların ajandalarından silinir ve kullanıcılara “DERS ADI + sohbeti + İptali” başlığı ile sohbetin iptal edildiğini belirten mesaj gönderilir.

## **26. Toplantı Modülü**

Bu modülde sistem yöneticisi, bölüm başkanı, danışman ve ders sorumlusu belirli zamanlarda ihtiyaç duyulan konularda toplantılar oluşturabilmektedir. Yeni bir toplantı oluşturmak için kullanıcı aşağıdaki bilgileri girerek toplantı ekleyebilir (Bayram 2009).

- Toplantı tarihi belirlenir.
- Toplantı saati girilir.
- Toplantı konusu girilir.
- Arama sonuçları arasından toplantıya katılmasını istediği kullanıcı ve kullanıcı grupları seçilir.
- Toplantı bildirmek için kullanıcılara yollanacak mesajın içeriği girilir ve Toplantı oluşturma işlemi tetikler.

Toplantı ekleme işlemi başarılı ise toplantıya katılacak kullanıcılara “Toplantı” başlığı altında mesaj gönderilir ve katılacak kullanıcıların ajandalarına olay eklenir. Toplantı tarihi geldiğinde toplantı sayfasına girerek davetli olduğu toplantılar listelenir ve katılmak istediği toplantıyı seçerek toplantıya katıl işlemini tetikler. Toplantı silinmek istendiğinde ise toplantı tarihi o anki tarihle aynı değilse toplantı silinir ve kullanıcılara toplantının iptal edildiğini belirten “Toplantı İptali” başlığı altında mesaj gönderilir. Toplantıya katılacakların ajandalarından ise toplantı iptal edilir (Bayram 2009).

## **27. Sıkça Sorulan Sorular Modülü**

Sıkça sorulan sorular modülü ile öğretim elemanları, dersle ilgili sıkça sorulan soruları ve bu soruların cevaplarını belirli kategoriler altında toplamaktadırlar. Öğrenciler öğrenmek istediği sorunun cevabına bu kategori içerisinde ulaşmaktadır. Sistemde yeni bir kategori oluşturmak ve bu kategori içerisine sıkça sorulan bir soru eklemek için SSS sayfasında aşağıdaki işlemler takip edilir.

1. Sistemde eklemek istediği soru daha önce oluşturulan kategorilerden farklı ise ilk olarak yeni kategori oluşturulur. Bunun için tanımlanacak kategori adı girilir ve kategori oluştur işlemi tetiklenir. Sistemde aynı ada sahip kategori adı varsa kullanıcıya hata mesajı döndürülür, yoksa yeni kategori sistemdeki kategoriler arasına eklenir.
2. Soruyu sisteme eklemek için ise, öncelikle eklenecek kategori seçilir. Daha sonra soru metnini ve soru cevabı girilerek soru oluştur işlemi tetiklenir. Soru bu kategoride varsa kullanıcıya hata bilgisi gösterilir, yoksa soru kategori altına eklenir.

## **28. Görüş ve Öneriler Modülü**

Kullanıcıların görüş bildirmeleri veya öneride bulunmaları için oluşturulmuş bir modüldür. Bu modülde öğrenci, öğretim üyesi, sistem yöneticisi, bölüm sekreteri ve



bunlara ek olarak bir de misafir kullanıcısı aktiftir. Bu modül üç aşamadan gerçekleşmektedir (Bayram 2009).

1. Kullanıcıların, görüş veya önerilerini gönderebilmesi,
2. Kullanıcılar tarafından gönderilen görüş ve önerilerin onaylanması,
3. Onaylanan görüş ve önerilerin kullanıcılar tarafından görülebilmesidir.

Görüş yada öneri bilgilerinin girilmesinin ardından görüş/öneri gönder işlemini tetiklenerek bu bilgiler sisteme kaydedilir. Bildirilen görüş yada öneri sistemde görülmeye uygunsuz onaylama işlemi tetiklenir ve onaylanan görüş/öneriler kaydedilerek sistemde görünebilir hale getirilir.

## **29. Duyurular Modülü**

Okul, bölüm, program yada dersler için duyuruların eklendiği modüldür. Öğrenci, öğretim üyesi, sistem yöneticisi, bölüm sekreteri ve öğrenci işleri bu modüldeki aktif kullanıcılarıdır. Bu modül ile sistemdeki yeni duyurular duyuru tipleri altında sisteme eklenmekte ve yayınlanmaktadır. Sisteme bir dönem boyunca eklenen tüm duyurular dönem sonunda otomatik olarak, programda bir daha gözükmeyecek şekilde sistemden kaldırılmaktadır. Sisteme yeni bir duyuru oluşturma işlemi aşağıdaki şekilde yapılmaktadır (Bayram 2009).

- Eklenecek duyurunun tipi daha önce tanımlanmamış ise kullanıcı öncelikle yeni bir duyuru tipi tanımlar. Bunun için duyuru tipinin adını ve simgesini girerek sisteme ekleme işlemini gerçekleştirir.
- Duyuru eklemek için ise duyuru tipi verisini opsiyona bağlı olarak seçer ve aşağıdaki bilgileri girer.

- ✓ Duyuru ilan tarihi
- ✓ Duyuru başlangıç tarihi
- ✓ Duyuru bitiş tarihi

- ✓ Duyuru başlığı
- ✓ Duyuru metnini

Bu bilgiler girildikten sonra duyuru oluřtur iřlemi tetiklenir ve duyuruyu hazırlayan kiřinin bilgisi alınarak sisteme kaydedilir.

### **30. Ajanda Modülü**

Ajanda modülü ile kullanıcılar yaklaşan olayları takip edebilmektedirler. Ajanda alanındaki olaylar diđer kullanıcılar tarafından belirtilen toplantı, sohbet, sanal sınıf uygulaması gibi yaklaşan olaylar olduđu gibi ayrıca kullanıcının kendisi tarafından eklenen kiřisel olaylarda olabilmektedir. Diđer kullanıcılar tarafından eklenen onaylar ajanda alanına otomatik olarak düşmektedir. Kiřisel olay notu girmek için ise kullanıcı not başlığını, not metnini, not tarih ve saatini girerek notu kaydet iřlemini tetikler ve girilen not sisteme kaydedilir. Yaklaşan olaylar Ajandam alanında bir takvim üzerinde seçilen zaman diliminin notları iřaretli olarak gelir. Kullanıcı hangi günü tıklarsa o günün notları ekranda gösterilir (Bayram 2009).

### **31. Akademik Takvim Modülü**

Dönem içerisindeki önemli tarihler akademik takvim modülü ile belirlenir. Akademik takvim içerisinde;

1. Öğrenci online başvuru dönemi başlangıç ve bitiş tarihleri,
2. Öğrenci kayıt dönemi başlangıç ve bitiş tarihleri,
3. Ders kayıt dönemi ders kayıt ve yenileme dönemi tarihleri,
4. Okul Güz dönemi başlangıç ve bitiş tarihleri,
5. Okul Bahar dönemi başlangıç ve bitiş tarihleri,
6. Okul Yazokulu dönemi başlangıç ve bitiş tarihleri,
7. Ayrıca eklenmesi gereken başka bir olay varsa olayın başlangıç ve bitiş tarihleri yer almaktadır.

### **32. Transkript Modülü**

Transkript modülü ile sistemde kayıtlı olan öğrencilerin transkript bilgilerini danışmanlarının, öğrenci işlerinin, bölüm sekreterinin, sistem yöneticilerinin ve öğrencilerin görebilmesi sağlanmaktadır. Öğrenciler girdikleri sınavların sonuçlarına buradan erişebilmektedirler. Transkriptlerde bulunan bilgiler şunlardır (Bayram 2009).

- Öğrenci numarası bilgisi.
- Öğrenci adı ve soyadı bilgisi.
- Öğrenci okul bilgisi.
- Öğrenci bölüm bilgisi.
- Öğrenci durum bilgisi(Mezun/Okuyor).
- Toplam puan bilgisi
- Toplam kredi
- Genel ortalama bilgisi
- Görüntülenen günün tarih bilgisi
- Ders bilgileri
- Öğrencinin disiplin durumu bilgisi.
- Kişisel bilgilerinden kontrol edilerek “disiplin cezası vardır/yoktur” biçiminde yazdırılır.
- Transkript onama/imza yetkilisi bilgileri,

### **33. Raporlama Modülü**

Sistem yöneticisinin, öğretim üyesinin, danışmanının ve öğrenci işlerinin ihtiyaç duyduğu bütün raporlar bu modül ile görüntülenmektedir. Raporlama modülü ile aşağıdaki raporlar alınabilmektedir (Bayram 2009).

- Öğrenci Belgesi

- Öğrenci transkripti
- Öğrenci takip raporu- Ders listesi ve her derse giriş sayısı dersin tamamladığı yüzdesi
- Öğrenciye göre ders ve sınav listesi
- Raporun tarihi
- Zamanında harç yatırmayanların listesi
- Öğr. Üyesi takip raporu
- Verdigi ders ve şubeler
- Danışmanı olduğu öğrencilerin listesi
- Bölümlerin listesi
- Fakültedeki öğrenciler
- Programların listesi
- Bölümdeki öğrencilerin listesi
- Programdaki öğrencilerin listesi
- Programa ait derslerin listesi
- Programlara göre ders sorumluları listesi
- Derslere göre kişi listesi
- Sınavlara göre kişi listesi
- Derslere göre kişinin bitirme durumu
- Derse atanamayan kişi listesi
- Tüm derslerin listesi
- Ders bilgileri (içerik bilgieri)
- Ünvanlara göre kullanıcı listesi
- Gruplara göre kişi listesi
- Yetki seviyesine göre kullanıcı listesi
- Sohbet Sanal sınıf kayıtları
- Şubelere göre sınav not listesi
- Ders onay belgesi (Danışmanın imzalayacağı belge formatı)

### **34. Mali İşlemler Modülü**

Ders saati ücretleri belirleme, ders saati ücretleri güncelleme, öğretim üyesi ücretleri belirleme işlemleri bu modülde gerçekleştirilmektedir. Ayrıca öğrencilerin alacakları ders sayısına göre harç ödemelerini ve öğretim elemanlarının da girdikleri ders sayısına göre alacakları ücretleri görebilmelerini sağlamaktadır. Ders saati ücreti programlara göre değişebilmektedir. Ders saati ücreti belirlemek için mali işlemlerle yetkili kullanıcı ilgili sayfa üzerinden önce sistemdeki tüm programları listeler ve bu listeden ders ücretlerini belirleyeceği programı seçerek ders saat ücretlerini girer. Öğretim üyesi ücretlerini belirlemek için ise bir web servis aracılığı ile personel sicil numaraları üzerinden sistemde ders veren öğretim üyelerinin ders saat bilgisi varolan sisteme gönderilerek belirlenir. Öğrenciler ise ilgili bağlantı aracılığı ile ders kayıt ücreti yatırma işlemleri sayfasına ulaşarak danışman tarafından onaylanan ders listesine göre yatırması gereken tutarı görüntüler. Tutar, program ücreti yada kayıt ücretidir. Havale yada kredi kartı ile bu tutar ödenebilir. Yatırma işlemi sonrasında kullanıcı bankadan aldığı şifreleri girer. Şayet yatırma işlemleri için onay alınırsa kullanıcıya kullanıcı adı ve şifresi mail olarak gönderilir. Gönderilen kullanıcı adı öğrenci numarasıdır. Şifre ise otomatik verilir ve ilk girişinde kullanıcının değiştirmesi istenir. Kullanıcının başvuru takip bilgileri güncellenir ve kullanıcı bu bilgilerle giriş yaptığında sistem aktif hale getirilir (Bayram 2009).

### **35. Demo Kullanıcı Modülü**

Demo kullanıcılarının normal kayıt yaptıran öğrenciler gibi kayıt işlemleri, dersler gibi modülleri sınırlı ölçüde kullanmalarını sağlayarak, sistemi tanımalarını sağlayan modüldür. Demo kullanıcısı olarak sisteme giriş yapmak için kayıt olmak şarttır. Kayıt işleminde demo kullanıcısının temel bilgileri doldurması gerekmektedir (Bayram 2009).

Kayıt işleminin sonunda kullanıcı adı ve şifre sistem tarafından üretilerek kullanıcının belirtmiş olduğu mail adresine gönderilir. Demo kullanıcısı gönderilen kullanıcı adı ve şifreyle sisteme giriş yaparak izin verilen alanlara ulaşabilmektedir.

### **36. Workshop Modülü**

Akademisyenlerin belirledikleri bir konu üzerinde tartişmalarını, bilgilerini paylaşmalarını ya da bilgi alışverişı yapmalarını sađlayan modüldür. Örneđin bulunduđu üniversiteden yurtdışında yada yurt içinde bulunan alanındaki öğretim elemanlarıyla bir konu hakkında konuşmak üzere online olarak toplanmalarını sağlamaktadır. Açılacak workshoplar akademisyenler tarafından önerilir. Workshop talebinde bulunacak kiři gerekli temel bilgilerini girerek sisteme kaydolur.

Workshop için onay alınmasının ardından vermek istediđi workshop için gerekli bilgileri tanımlar. Bu bilgileri girdikten sonra workshop yapacak kiři CV dosyasını sisteme ekler. Kredi kartı yada havale ile gerekli ücreti ödemesinin ardından sisteme giriş yapacağı kullanıcı adı ve şifre otomatik olarak üretilerek kullanıcının belirtmiş olduđu mail adresine gönderilir.

### **37. Sanal Sınıf Modülü**

Öğretim elemanı ile öğrencilerin sanal bir sınıfta (online olarak) istedikleri bir dersin anlatılmasında kullanacakları bir modüldür. Burada öğretim elemanı sanal bir tahta kullanarak dersi anlatır. Öğrenciler de el kaldırma sistemi ile öğretim elemanının da müsaade vermesi ile sorularını sorabilir yada derse katılabilirler. Sistemde sanal sınıf oluşturmak için ders sorumlusu sanal sınıf kısmında yapabileceđi işlemler sayfasına ulaştıktan sonra, oluşturmak istediđi sanal sınıfın hangi programa açık olacağını belirledikten sonra sanal sınıf tanımla işlemini tetikler. Belirtilen sanal sınıf FMS üzerinde açılır. Sanal sınıfa yalnızca o dersi alanlar erişebilir.

EK-2

# AFYONKARAHİSAR KOCATEPE ÜNİVERSİTESİ



## UZAKTAN ÖĞRETİM SİSTEMİ PROJESİ ANALİZ DÖKÜMANI

Version	Date	Description	Author
1.1	17.02.2007	Analiz Dökümanı	Ayhan KOÇ ArkadaşKILIÇ

Proje Yürütücüsü Kurumları	Proje Yürütücüleri	Yürütme
Afyon Kocatepe Üniversitesi	Doç.Dr. Mevlut DOĞAN	Mahmut KANTAR
BilgeAdam	Ümit AKBAŞ	Ayhan KOÇ Ümit AKBAŞ

# GİRİŞ

## MEVCUT SİSTEM

Mevcut bir sistem bulunmamaktadır.

## YENİ SİSTEM HEDEFLERİ VE GETİRİLERİ

- Afyonkarahisar Kocatepe Üniversitesinde Uzaktan Öğretim Yöntemini kullanan Yüksek Lisans, Lisans, Ön Lisans Programları açmak.
- Türkiye’de ilk defa senkron sınıf uygulaması kurularak, görsel bir şekilde, kişilerin buldukları yerden hocalarını görerek ve soru sorarak eğitim yapabilmeleri sağlanacaktır.
- Uzaktan Öğretim Yöntemini kullanarak (Workshop) çalışmalar düzenlemek.
- Öğretimde coğrafi sınırları ortadan kaldırmak.
- Kullanıcılara sağlanan esnek yetkilendirme sistemi ile kullanıcının hangi uygulama ekranlarını görebileceği, görebildiği ekranlarda hangi verilere erişebileceği tanımlanabilecektir.
- Kullanılacak teknolojiler sayesinde elde edilecek kazanımlar:
  - Wise Web kontroller ile kullanıcı dostu bir arabirim sağlanacaktır.
  - .Net 2.0 ve Ajax Teknolojisinin güncel ve yeni bir teknoloji olması sayesinde elde edilecek performans getirisinin yanında geleceğe yönelik yeni versiyon geliştirme işlemleri mevcut alt yapı üzerinden devam edebilecektir.
  - Web Partlar kullanımı sayesinde kullanıcılar ekranlarını kişiselleştirebilecektir.
  - Reporting Services kullanımı sayesinde sistem üzerinde kolaylıkla yeni raporlar oluşturulabilecektir.



## GENEL TERİMLER SÖZLÜĞÜ

### GENEL TERİMLER SÖZLÜĞÜ - DÖKÜMAN VERSİYONLARI

Version	Date	Description	Author
1.0	17.02.2007	Analiz toplantıları sonucunda oluşturulan sözlük.	Ayhan KOÇ Arkadaş KILIÇ

### TERİMLER

**OKUL:** Fakülte, Enstitü ve Yüksek Okul terimlerine karşılık sistemde kullanılan ifadedir.

**BÖLÜM:** Üniversitelerdeki Fakülteler altında yer alan bölümlere karşılık gelecek şekilde sistemde kullanılan ifadedir.

**PROGRAM:** Sistemde bölümler altında programlar vardır.Bu programlar Bilgisayar Mühendisliği Yüksek Lisans Programı vs. gibi olacaktır.

**EĞİTİM KATALOĞU:** Sistemde bulunan tüm programları içeren yapıdır.

**ŞUBE:** Derslerin işleneceği sınıfa karşılık gelmektedir.Derslerin şubesi varsa programa dahil kullanıcılar derse erişebilecektir.

**KULLANICI GRUBU:** Sistemde kayıtlı kullanıcılardan oluşan ve mesajlaşma, listeleme işlemlerinde kolaylık sağlayan ifadedir.

**YÖNETİM ÜNVANI:** Yönetimde üst düzeyde söz sahibi olan kişilerin sahip oldukları sıfattır.(Dekan,Müdür ,Bölüm Başkanı,Danışman vs.)

**AKADEMİK ÜNVAN:** Kişilerin akademik dünyada sahip oldukları sıfattır.(Profesör, Doçent,Yrd Doçent,Öğretim Görevlisi vs.)

**İDARİ ÜNVAN:** Kişilerin üniversitede yönetim aşamalarında üstlenebildikleri sıfatlardır. (Bölüm Sekreteri, Fakülte Sekreteri, Daire Başkanı vs.)

## SİSTEMDE YER ALAN SEVİYELER

**ADMİN:** Sistemde genellikle ekleme, silme, güncelleme yetkilerine sahip olan kullanıcıların bulunduğu seviyedir.

**USER:** Sistemde genellikle görüntüleme ve kullanma yetkilerine sahip kullanıcıların bulunduğu seviyedir.

## SİSTEMDE YER ALAN AKTÖRLER

Aşağıdaki aktörler aslında sistemde varsayılan olarak yer alması düşünülen kullanıcı rolleridir. Bu rollere yetkiler eklenip çıkarılabilmektedir. Bu sayede daha sonradan bu aktörlerin yetkileri belirlenebilir, yeni roller oluşturulabilir.

**SİSTEM YÖNETİCİSİ:** Sistemde yeralan tüm yetkilere sahip kullanıcıdır.

**BÖLÜM BAŞKANI:** Sistemde bölümler üzerinde tüm yetkilere sahip kullanıcıdır.

**BÖLÜM SEKRETERİ:** Sekreteri olduğu bölüm üzerinde belirli görevleri gerçekleştirebilecek kullanıcıdır.

**DERS SORUMLUSU:** Dersi veren öğretim elemanını ifade eder.

**ÖĞRENCİ İŞLERİ:** Üniversitelerde yeralan öğrenci işleri biriminin yetkilerine sahip olacak kullanıcılarıdır.

**MİSAFİR:** Sisteme kayıtlı olmayan sistem yöneticisi tarafından belirlenen bölümleri kullanabilen kullanıcılarıdır.

**DEMO KULLANICI:** Sistem üzerinde tanıtım amaçlı oluşturulmuş programlara erişim izni olan kullanıcıdır.

**DANIŞMAN:** Üniversite yapısında her öğrencinin bir danışmanı bulunmaktadır. Sistemde bu kullanıcının en önemli görevi ders kayıtlarını inceleyip onaylamaktır.

**SANAL SINIF:** Asenkron eğitimlerin gerçekleştirileceği online ortamdır.

**FMS (Flash Media Server):** Sanal sınıf uygulamalarının üzerinde çalışacağı medya sunucusudur.

**ÖĞRENCİ NUMARASI:** Sistem tarafından öğrencilere verilecek olan tekil numaradır.

Formatı şu şekildedir;

- 11 karakter ve sayısal bir ifade olacaktır.
  - İlk 2 hane: Öğrenci giriş yılı
  - Sonraki 2 hane: Okul Kodu
  - Sonraki 2 hane: Bölüm Kodu
  - Sonraki 2 hane: Program kodu
  - Sonraki 3 hane: Öğrencinin programdaki sıra numarası olacaktır.

## KULLANILAN TEKNOLOJİLER

- .Net Framework 2.0
- Microsoft SQL Server 2005
- Microsoft SQL Server 2005 Reporting Services
- Microsoft Visual Studio 2005
- Microsoft Visual Studio Team System
- XML Web Servisleri
- AJAX Teknolojisi

## MİMARİ TASARIM

### MİMARİ TASARIM - DÖKÜMAN VERSİYONLARI

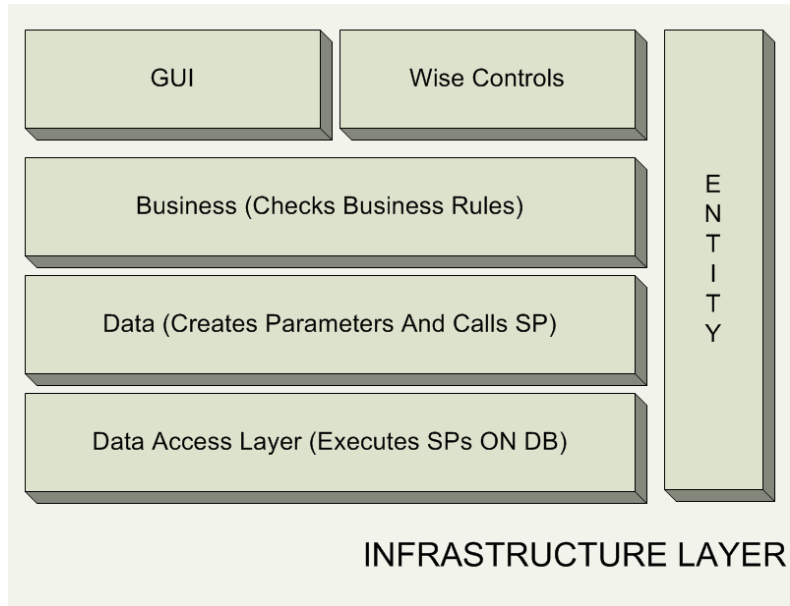
Version	Date	Description	Author
1.1	17.02.2007	Gerçekleştirilecek uygulamanın mimari tasarımı	Ayhan KOÇ Arkadaş KILIÇ

## MIMARI TASARIM DETAYLARI

Mimari tasarımı oluşturan katmanlar:

- GUI Layer
- Wise Controls
- Business Layer
- Data Layer
- Data Access Layer
- Entity Layer

olarak ayrılmaktadır.



Şekil Ek.1 Mimari tasarım katmanları.

## DATA ACCESS LAYER

- *Data Access Layer*, *Data Layer*'da yer alan *RDBMS* sistemine göre hazırlanmış bir katmandır. Bu katman *Data Layer* katmanında kullanılan veritabanı yönetim sistemi ile veri iletiminin nasıl yapılacağını bilen ve bu görevi üstlenen katmandır.

- Sistemin farklı veritabanı yönetim sistemlerine taşınması için *Data Access Layer* üzerinde küçük düzenlemeler yapılması yeterlidir.
- Bu katmanda generic database sürücülerini (*OleDb sürücülerini*) kullanmak yerine alt katmanda bulunan veritabanı yönetim sistemi üzerinde en iyi performansı verecek *ADO.NET* veri sağlayıcıları kullanılmaktadır.

## DATA LAYER

- *Business Layer* ve *Data Access Layer* arasında olan bu katmanın görevi *Data Access Layer* aracılığı ile alınan ham verinin *Entity*'lere yüklenmesini, veya *Entity*'de bulunan verinin veritabanı kullanımı için hazır hale getirilmesini sağlayan katmandır.

## ENTITY

- Analiz çalışmaları neticesinde ortaya çıkan uygulama nesnelere tanıtıldığı katmandır.
- *Entity* terimine örnek vermek gerekir ise, *Personel*, *Job Family Factors*, *Education Factors* gibi terimler sistemdeki bazı *Entity*'leri tanımlamaktadır.
- *Entity*'ler tanımlanırken gerçek yaşamdaki nesne ilişkileri düşünülerek gerçek bir model tasarlanmaktadır.

## BUSINESS LAYER

- Projede yer alan iş kurallarının uygulamasının gerçekleştirildiği katmandır.
- Bir iş kuralı için yapılacak sınıma ve karar verme işlemleri bu katman içerisinde yapılır ve buna göre presentation layer'a bilgi aktarımı yada data mapper layer'a veri aktarımı gibi işlemler gerçekleştirilir.
- Presentation layer diğer katmanlar ile direkt iletişim kurmaz mutlaka business layer üzerinden geçerek işlemleri gerçekleştirmektedir. Böylece iş kuralları, hata yakalama blokları tek katman üzerinden bulunmaktadır ve kodlama kolaylığı sağlamaktadır.

## PRESENTATION LAYER

- Presentation Layer, uygulamanın gereksinimlerine göre Windows Application, Web Application ya da Mobile Application'lar olabilmektedir.
- Kullanıcının göreceđi ekranları oluřturan presentation layer kullanıcı dostu, řık ve gerekli durumlarda özelleřtirilebilir kullanıcı arabirimi sunmaktadır.
- Presentation Layer'da kullanmak üzere Windows Application ve Web Application'lar için dünya çapında kalitesini kanıtlamıř ve bilinen üçüncü parti bileřen sađlayıcılarından satın alınmıř olan bileřen kütüphanemiz kullanıcılara zengin bir arayüz sunmaktadır.

## UC1: ONLİNE BAŞVURU

### KAPSAM

Online başvuru işlemleri

### SEVIYE

Admin, User

### AKTÖR

Sistem Yöneticisi, Bölüm Başkanı, Bölüm Sekreteri, Öğrenci İşleri, User

### ÖN ŞARTLAR

### BAŞARILI SENARYO SONUCU

Kullanıcıların sorunsuz şekilde online başvuru işlemini gerçekleştirebiliyor ve takip edebiliyor olması

### TETIKLEYICI

### BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar

1. Admin ise;
  - a. Online başvuru işlemleriyle ilgili bağlantı aracılığı ile online başvuru sayfasına ulaşır.
  - b. Kullanıcı yapmak istediği işlemi belirler.
  - c. Program tipi seçer.
    - i. Yüksek Lisans ise;
      1. Online başvuru onaylama
        - a. Kullanıcının yetkisi dahilindeki programı seçer.
        - b. İlgili programa başvuran kullanıcılar listelenir.

- c. Kullanıcılar ALES puanlarına göre sıralanır.
  - d. Listedeki uygun başvuruları seçer.
    - i. Kullanıcılar programın kontenjan sınırına göre
      - 1. Asil
      - 2. Yedek olarak belirlenir.
  - e. Online başvuru onayla işlemini tetikler.
  - f. Online başvuru onaylanır.
  - g. Kullanıcının başvuru takip bilgileri güncellenir.
  - h. Kullanıcılara davet maili gönderme işlemi beklenir.
2. Başvuru bilgilerini gönderme(Bölüm sekreteri)
- a. Kullanıcı yetkisi dahilindeki programı seçer.
  - b. İlgili programda online başvurusu onaylanan kullanıcılar listelenir.
  - c. Kullanıcıya gönderilecek maili hazırlar.
    - i. Başvuruları onaylanan kullanıcıların email adreslerine başvuru bilgileri ile birlikte ,
      - 1. Kabul edildiği program bilgisi
      - 2. Getirmesi gereken evrak listesi
        - a. Bu alan html formatını destekleyecektir.
      - 3. Ek Dosya
        - a. Gerekli bilgilerin tamamının bir dosyada bulunduğu durumlarda kullanıcı dosyayı ekleyebilecektir.
      - 4. Mülakat bilgileri
        - a. Mülakat yeri girer.
        - b. Mülakat tarihini seçer.
        - c. Mülakat saatini girer.



## 5. Sınav bilgileri

- a. Sınav yeri girer.
- b. Sınav tarihini seçer.
- c. Sınav saatini girer.
- d. Davet mailini gönder işlemini tetikler.
- e. Kullanıcının başvuru takip bilgileri güncellenir.

## 3. Kullanıcı kabul (öğrenci işleri)

- a. Program seçer.
- b. İlgili programda online başvurusu onaylanan kullanıcılar listelenir.
- c. Kullanıcıdan mail ile istenen bilgiler listelenir.
- d. Bilgi kontrolü yapılır. Tüm belgeler istenilen formatta ise kullanıcı kabul görür, mülakat ve sınav aşamasına yönlendirilir.
- e. Kullanıcının başvuru takip bilgileri güncellenir.

## 4. Mülakat ve sınav sonuc girişi işlemi(bölüm sekreteri)

- a. Kullanıcı yetkisi dahilindeki programı seçer.
- b. İlgili programa, sınav ve mülakat için yönlendirilmiş kullanıcılar listelenir.
- c. Kullanıcının bilgilerine ,
  - i. Mülakat sonucu
  - ii. Sınav sonucu
- d. Giriş işlemini tetikler.

## ii. Lisans / Ön Lisans ise;

### 1. Kullanıcı ÖSYM den gelen dosyayı seçer.

- a. Gelen dosya .txt formatında olup alanlar şu şekildedir;

Sıra No	Aralık			Uzunluk	Alan Tipi	Alan Adı
1	1	-	11	11	Sayısal	TC KİMLİK NUMARASI
2	12	-	41	30	Alfabetik	ADI
3	42	-	71	30	Alfabetik	SOYADI
4	72	-	101	30	Alfabetik	BABA ADI
5	102	-	131	30	Alfabetik	ANA ADI
6	132	-	133	2	Sayısal	DOĞUM GÜNÜ
7	134	-	135	2	Sayısal	DOĞUM AYI
8	136	-	137	2	Sayısal	DOĞUM YILI
9	138	-	138	1	Sayısal	CİNSİYETİ
						1-ERKEK, 2-KIZ, 9-BELİRSİZ
10	139	-	139	1	Sayısal	UYRUĞU
						1-TC, 2-KKTC, 3-YABANCI
11	140	-	169	30	Alfabetik	DOĞUM YERİ
12	170	-	171	2	Sayısal	NUFÜSA KAYITLI OLDUĞU İL
						NUFÜSA KAYITLI OLDUĞU
13	172	-	175	4	Sayısal	İLÇE
14	176	-	181	6	Sayısal	OKUL KODU
15	182	-	186	5	Sayısal	OKUL TÜRÜ
16	187	-	190	4	Sayısal	OKUL KOLU
17	191	-	191	1	Sayısal	ÖĞRENİM DURUMU
						0-BELİRSİZ
						1-SON SINIFTA
						2-BEKLEMELİ
						3-OKUL DIŞI
						4-YÜKSEK ÖĞR.DEVAM ETMİYOR

						5-KAYIT OLMADI
						6-YÜKSEK ÖĞR.DEVAM EDİYOR
						7-YÜKSEK ÖĞR.MEZUNU
						8-KAYDI SİLİNDİ
18	192	-	195	4	Sayısal	ORTAÖĞRENİM MEZUNİYET YILI
19	196	-	196	1	Sayısal	M.LİSESİ ADAY.TÜR KODLARI
						4-ANADOLU TEKNİK LİSESİ
						3-A.MESLEK L.,TEKNİK LİSE
						2-MESLEK LİSESİ
						1-ENSTİTÜLER
20	197	-	197	1	Sayısal	MESLEK LİSESİ MEZUNLARI
						2-KENDİ METEBİNDE
						1-KENDİ METEBİ DIŞINDA
21	198	-	204	7	Sayısal	MSG BAŞARI PUANI
22	205	-	205	1	Sayısal	YABANCI DİLİ
						1-ALMANCA
						2-FRANSIZCA
						3-İNGİLİZCE
						9-BELİRSİZ
23	206	-	206	1	Sayısal	KATKI KREDİSİ İSTEĞİ
						0-İSTEMİYORUM
						1-İSTİYORUM
24	207	-	207	1	Sayısal	ÖĞRENİM KREDİSİ İSTEĞİ
						0-İSTEMİYORUM

						1-İSTİYORUM
25	208	-	257	50	Alfabetik	ADRES SATIRI-1
26	258	-	307	50	Alfabetik	ADRES SATIRI-2
27	308	-	337	30	Alfabetik	SEMT-İLÇE
28	338	-	342	5	Sayısal	POSTA KODU
29	343	-	344	2	Sayısal	ADRES İLİ
						01-81 İL TRAFİK KODLARI
						90-KIBRIS
						91-YURT DIŞI
						99-BELİRSİZ
30	345	-	347	3	Sayısal	TELEFON ALAN KODU
31	348	-	354	7	Sayısal	TELEFON NUMARASI
32	355	-	355	1	Sayısal	OKUL BİRİNCİSİ
						1-OKUL BİRİNCİSİ
33	356	-	361	6	Sayısal	SÖZEL-1 ÖSS PUANI
34	362	-	367	6	Sayısal	SAYISAL-1 ÖSS PUANI
35	368	-	373	6	Sayısal	EA -1 ÖSS PUANI
36	374	-	379	6	Sayısal	DİL ÖSS PUANI
37	380	-	385	6	Sayısal	SÖZEL-2 ÖSS PUANI
38	386	-	391	6	Sayısal	SAYISAL-2 ÖSS PUANI
39	392	-	397	6	Sayısal	EA -2 ÖSS PUANI
40	398	-	403	6	Sayısal	Y-SÖZ-1 ÖSS PUANI(0.3)
41	404	-	409	6	Sayısal	Y-SAY-1 ÖSS PUANI(0.3)
42	410	-	415	6	Sayısal	Y-EA-1 ÖSS PUANI(0.3)
43	416	-	421	6	Sayısal	Y-DİL ÖSS PUANI(0.3)
44	422	-	427	6	Sayısal	Y-SÖZ- 2 ÖSS PUANI(0.3)
45	428	-	433	6	Sayısal	Y-SAY-2 ÖSS PUANI(0.3)
46	434	-	439	6	Sayısal	Y-EA-2 ÖSS PUANI(0.3)

47	440	-	445	6	Sayısal	Y-SÖZ-1 ÖSS PUANI(0.8)
48	446	-	451	6	Sayısal	Y-SAY-1 ÖSS PUANI(0.8)
49	452	-	457	6	Sayısal	Y-EA-1 ÖSS PUANI(0.8)
50	458	-	463	6	Sayısal	Y-DİL ÖSS PUANI(0.8)
51	464	-	469	6	Sayısal	Y-SÖZ- 2 ÖSS PUANI(0.8)
52	470	-	475	6	Sayısal	Y-SAY-2 ÖSS PUANI(0.8)
53	476	-	481	6	Sayısal	Y-EA-2 ÖSS PUANI(0.8)
54	482	-	487	6	Sayısal	Y-SÖZ-1 ÖSS PUANI(0.8+0.24)
55	488	-	493	6	Sayısal	Y-SAY-1 ÖSS PUANI(0.8+0.24)
56	494	-	499	6	Sayısal	Y-EA-1 ÖSS PUANI(0.8+0.24)
57	500	-	505	6	Sayısal	Y-DİL ÖSS PUANI(0.8+0.24)
58	506	-	511	6	Sayısal	Y-SÖZ-2 ÖSS PUANI(0.8+0.24)
59	512	-	517	6	Sayısal	Y-SAY-2 ÖSS PUANI(0.8+0.24)
60	518	-	523	6	Sayısal	Y-EA-2 ÖSS PUANI(0.8+0.24)
61	524	-	530	7	Sayısal	Y-SÖZ-1 (0.3) BAŞARI SIRASI
62	531	-	537	7	Sayısal	Y-SAY-1 (0.3) BAŞARI SIRASI
63	538	-	544	7	Sayısal	Y-EA-1 (0.3) BAŞARI SIRASI
64	545	-	551	7	Sayısal	Y-DİL (0.3) BAŞARI SIRASI
65	552	-	558	7	Sayısal	Y-SÖZ-2 (0.3) BAŞARI SIRASI
66	559	-	565	7	Sayısal	Y-SAY-2 (0.3) BAŞARI SIRASI
67	566	-	572	7	Sayısal	Y-EA-2 (0.3) BAŞARI SIRASI
68	573	-	579	7	Sayısal	Y-SÖZ-1 (0.8) BAŞARI SIRASI
69	580	-	586	7	Sayısal	Y-SAY-1 (0.8) BAŞARI SIRASI
70	587	-	593	7	Sayısal	Y-EA-1 (0.8) BAŞARI SIRASI
71	594	-	600	7	Sayısal	Y-DİL (0.8) BAŞARI SIRASI
72	601	-	607	7	Sayısal	Y-SÖZ-2 (0.8) BAŞARI SIRASI
73	608	-	614	7	Sayısal	Y-SAY-2 (0.8) BAŞARI SIRASI

74	615	-	621	7	Sayısal	Y-EA-2 (0.8) BAŞARI SIRASI
75	622	-	628	7	Sayısal	Y-SÖZ-1 (0.8+0.24) BAŞARI SIRASI
76	629	-	635	7	Sayısal	Y-SAY-1 (0.8+0.24) BAŞARI SIRASI
77	636	-	642	7	Sayısal	Y-EA-1 (0.8+0.24) BAŞARI SIRASI
78	643	-	649	7	Sayısal	Y-DİL (0.8+0.24) BAŞARI SIRASI
79	650	-	656	7	Sayısal	Y-SÖZ-2 (0.8+0.24) BAŞARI SIRASI
80	657	-	663	7	Sayısal	Y-SAY-2 (0.8+0.24) BAŞARI SIRASI
81	664	-	670	7	Sayısal	Y-EA-2 (0.8+0.24) BAŞARI SIRASI
82	671	-	676	6	Sayısal	Y-SÖZ-1 (0.3) ÖSS PUAN YÜZDELİĞİ
83	677	-	682	6	Sayısal	Y-SAY-1 (0.3) ÖSS PUAN YÜZDELİĞİ
84	683	-	688	6	Sayısal	Y-EA-1 (0.3) ÖSS PUAN YÜZDELİĞİ
85	689	-	694	6	Sayısal	Y-DİL (0.3) ÖSS PUAN YÜZDELİĞİ
86	695	-	700	6	Sayısal	Y-SÖZ-2 (0.3) ÖSS PUAN YÜZDELİĞİ
87	701	-	706	6	Sayısal	Y-SAY-2 (0.3) ÖSS PUAN YÜZDELİĞİ
88	707	-	712	6	Sayısal	Y-EA-2 (0.3) ÖSS PUAN

						YÜZDELİĞİ
89	713	-	718	6	Sayısal	Y-SÖZ-1 (0.8) ÖSS PUAN YÜZDELİĞİ
90	719	-	724	6	Sayısal	Y-SAY-1 (0.8) ÖSS PUAN YÜZDELİĞİ
91	725	-	730	6	Sayısal	Y-EA-1 (0.8) ÖSS PUAN YÜZDELİĞİ
92	731	-	736	6	Sayısal	Y-DİL (0.8) ÖSS PUAN YÜZDELİĞİ
93	737	-	742	6	Sayısal	Y-SÖZ-2 (0.8) ÖSS PUAN YÜZDELİĞİ
94	743	-	748	6	Sayısal	Y-SAY-2 (0.8) ÖSS PUAN YÜZDELİĞİ
95	749	-	754	6	Sayısal	Y-EA-2 (0.8) ÖSS PUAN YÜZDELİĞİ
96	755	-	760	6	Sayısal	Y-SÖZ-1 (0.8+0.24) ÖSS PUAN YÜZDELİĞİ
97	761	-	766	6	Sayısal	Y-SAY-1 (0.8+0.24) ÖSS PUAN YÜZDELİĞİ
98	767	-	772	6	Sayısal	Y-EA-1 (0.8+0.24) ÖSS PUAN YÜZDELİĞİ
99	773	-	778	6	Sayısal	Y-DİL (0.8+0.24) ÖSS PUAN YÜZDELİĞİ
100	779	-	784	6	Sayısal	Y-SÖZ-2 (0.8+0.24) ÖSS PUAN YÜZDELİĞİ
101	785	-	790	6	Sayısal	Y-SAY-2 (0.8+0.24) ÖSS PUAN YÜZDELİĞİ

102	791	-	796	6	Sayısal	Y-EA-2 (0.8+0.24) ÖSS PUAN YÜZDELİĞİ
103	797	-	803	7	Sayısal	YERLEŐTİRMEDE KUL.PUANI
104	804	-	810	7	Sayısal	YERLEŐTİRMEDE KUL.PUAN BAŐ.SIR
105	811	-	816	6	Sayısal	YER.KULLANILAN PUAN YÜZDESİ
106	817	-	823	7	Sayısal	GİRDİĞİ BÖLÜM KODU
						0000000-HİÇ BİR BÖLÜME GİREMEDİ
107	824	-	825	2	Sayısal	GİRDİĞİ BÖLÜM TERCİH SIRASI
						99-HİÇ BİR TERCİHİNE GİREMEDİ
108	826	-	827	2	Sayısal	YERLEŐTİĞİ PUAN TÜRÜ
						7-SÖZ,8-SAY,9-EA,10-DİL,11-SÖZ-2,12-SAY-2,13-EA-2
109	828	-	841	14	Alfabetik	GENEL,OKUL BİR.ÖZÜRLÜ,TUBİTAK

2. Kullanıcıları sisteme ekle işlemini tetikler.
3. Belirtilen kullanıcılar Lisans/ÖnLisans öğrencileri arasına eklenir.

2. User ise;

- a. Online başvuru işlemleriyle ilgili bağlantı aracılığı ile online başvuru sayfasına ulaşır.
- b. Kullanıcı yapmak istediği işlemi belirler.
  - i. Program tipi seçer.



1. Ön Lisans
  2. Yüksek Lisans
  3. Lisans
- ii. Yüksek Lisans ise;
1. Online başvuru yapmak
    - a. Online başvuru için gerekli alanları doldurur.
      - i. Kişisel bilgiler
        1. TC kimlik numarası
        2. Adı
        3. Soyadı
        4. Cinsiyeti
        5. Doğum tarihi
      - ii. İletişim bilgileri
        1. Email
        2. Ev telefonu
        3. Cep telefonu
        4. Adres
      - iii. İş Bilgileri
        1. Firma Adı
        2. Firma Adresi
        3. İlçe
        4. İl
        5. Posta Kodu
        6. İş Telefonu
        7. Dahili
        8. Pozisyonu
        9. Departmanı
      - iv. Eğitim bilgileri

1. Mezun olduđu üniversiteyi seçer.
2. Fakülte seçer.
3. Bölüm seçer
4. Diploma notu girer.
  - a. Gireceđi diploma notu türünü seçer
  - b. Seçebileceđi türler
    - i. 100' lük diploma notu
    - ii. 4'lük diploma notu
    - iii. ALES puanları
5. Başvuruda deđerlendirilecek ALES puanlarını girer.
  - a. ALES EA Puanı
  - b. ALES SAY Puanı
  - c. ALES SOZ Puanı
6. Varsa yurtdışı genel yetenek sınavı sonucunu girer.
7. Dil puanlarını girer.
  - a. Varsa ÜDS puanını girer.
  - b. Varsa KPDS puanını girer.
  - c. Varsa TOEFL puanını girer.
  - d. Varsa ILTS puanını girer.
  - e. Varsa Üniversite Dil Yeterlilik Sınavı puanını girer.

- b. Başvurabileceđi programlar listelenir.
- c. Kullanıcı başvurmak istediđi programları seçer.
- d. Programa başvur işlemini tetikler.
- e. Kullanıcıya girmiş olduđu bilgiler gösterilerek, onay alınır.

- i. Kullanıcı başvuru bilgilerini deęiřtirmek  
isterse başvuru ekranına dön iřlemine tetikler.
  1. Başvuru ekranı görüntülenir.
  2. Gerekli düzenlemeleri yapar.
  3. Programa başvur iřlemine tetikler.
- f. Kullanıcı bilgileri onaylarsa onayla iřlemine tetikler.
- g. Sisteme giriř için kullanıcıya kullanıcı adı olarak aday numarası ve řifre otomatik olarak belirlenir.
  - i. Aday Numarası
    1. AN ile başlayacaktır.
    2. Okul kodu, bölüm kodu, program kodu, sıra no bilgilerini içerecektir.
  - h. İletişim bilgilerinde belirttięi mail adresine bu bilgiler yollanır.
  - i. İlgili programa girdięi bilgilerle başvurusu yapılır.
2. Online başvuru takibi
  - a. İlgili bağlantı aracılıęı ile online başvuru takip ekranına ulaşır.
  - b. İlgili alanlara kullanıcı adı ve řifre bilgilerini girer.
    - i. Aday numarasını girer.
    - ii. Sifresini girer.
    - iii. Kullanıcının başvurduęu programlar ve başvuru durumu listelenir.
      1. Listeleme ekranı
        - a. Program adı
        - b. Başvuru tarihi
        - c. Deęerlendirme tarihi
        - d. Durumu

- i. Deęerlendirme ařaması devam ediyor.
- ii. Kabul edilmediniz.
- iii. Davet mailiniz gönderildi.
- iv. Belgeleriniz kontrol edildi.
- v. Kabul edildiniz.
- vi. Tüm işlemleriniz tamamlandı, Öğrenci numaranız ve şifreniz mail adresinize gönderildi.

iii. Lisans/Ön Lisans ise;

1. Online başvuru işlemi olmayacaktır. ÖSYM den gelen veriler sisteme eklendiği zaman kullanıcılar ilk girişini aşağıdaki bilgilerle yapacaktır.
  - a. Kullanıcı adı TC Kimlik No
  - b. Şifre TC Kimlik No olacaktır.
2. Ders kayıt işlemini gerçekleştirir.
3. Danışmandan onay aldığı anda ödemesini yapabilir.
4. Ödemeyi yaptığında Öğrenci No ve şifre oluşturulacaktır.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

## UC2: KULLANICI VE KULLANICI GRUPLARI MODÜLÜ

### KAPSAM

Kullanıcılar ve kullanıcı grupları üzerindeki işlemler

### SEVIYE

Admin

### AKTÖR

Sistem Yöneticisi, Bölüm Sekreteri, Bölüm Başkanı, Öğrenci İşleri

### ÖN ŞARTLAR

### BAŞARILI SENARYO SONUCU

Admin işlemleri,

- 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.

### TETİKLEYİCİ

### BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar,

1. Admin ise;

a. Kullanıcılar bölümü ile ilgili bağlantı aracılığı ile kullanıcı işlemleri sayfasına ulaşır.

b. Yapmak istediği işlemi belirler.

i. Kullanıcı arama

1. Arama kriterleri

a. Kullanıcı adı

i. Arama yaparken kullanacağı opsiyonlar

ii. Girilen metin ile başlayanları

iii. İçerisinde girilen metin geçenleri

iv. Girilen metin ile bitenleri arayabilir.

b. Kullanıcı grubu

i. Belli bir gruba göre arama yapabilecektir.

2. Arama işlemi tetikler.

3. Sistem yöneticisi ise bütün kullanıcıları, değilse sadece aktif durumda olan kullanıcıları listeler.

ii. Yeni kullanıcı oluşturmak.

1. Sisteme giriş bilgileri olan

a. Personel ise sicil numarası, Öğrenci ise öğrenci numarası alınacaktır. (Opsiyona bağlı)

i. Sicilno ilk ikisi alfabetik( kg0001)

ii. Öğrenci numarası 9 karakter(nümerik)

b. Kullanıcı adı

c. Ad

d. Soyad

e. Akademik ünvanı (öğrenci, Bölüm Sekreteri vs.)

f. Idari ünvanı(bölüm sekreteri vs.)

g. Kayıt tarihi (otomatik olarak işlem yapılan tarih alınacaktır.)

h. Aktif/Pasif

2. Kullanıcı genel bilgileri olan
  - a. TC Kimlik No
  - b. Doğum Tarihi
  - c. Cinsiyet
  - d. Açıklama
3. Öğrenci ise akademik bilgileri olan
  - a. Öğrenci no
  - b. Okul kodu
  - c. Bölüm kodu
  - d. Program kodu
  - e. Yarıyılı
  - f. Durumu(mezun, okuyor vs.)
  - g. Danışmanı
4. İletişim bilgileri olan
  - a. Ev telefonu
  - b. Cep telefonu
  - c. Email
  - d. Web Sayfası
  - e. İl
  - f. İlçe
  - g. Posta kodu
  - h. Adres
5. İş bilgileri olan
  - a. Firma Adı
  - b. Sektörü
  - c. Görevi
  - d. İş telefonu
  - e. Faks

- f. İş adresi
  - g. Posta kodu
  - h. İl
  - i. İlçe
  - j. Web Sayfası
6. Şifre ve şifre hatırlatma bilgileri olan
    - a. Şifre
    - b. Şifre tekrar
    - c. Hatırlatma Sorusu
    - d. Cevabı, bilgilerini girer.
  7. Kullanıcı oluştur işlemini tetikler.
  8. Girilen bilgiler istenilen formata uygunsa kullanıcı sisteme eklenmiş olur.
- iii. Kullanıcı silmek.
1. Kullanıcı arar.
  2. Silinecek kullanıcıyı seçer.
  3. Kullanıcı sil işlemini tetikler.
  4. Kullanıcı aktif bilgisi pasif hale getirilerek sisteme erişimi engellenir.
- iv. Kullanıcı güncellemek.
1. Kullanıcı arar.
  2. Bilgilerini güncellemek istediği kullanıcıyı seçer.
  3. Kullanıcı adı bilgisi haricindeki tüm verileri güncelleyebilir.
  4. Kullanıcı güncelle işlemini tetikler.
  5. Girilen bilgiler istenilen formata uygunsa kullanıcı bilgileri sistemde güncellenmiş olur.
- v. Kullanıcı grubu arama
1. Arama kriterleri
    - a. Grup adı



- i. Arama yaparken kullanacağı opsiyonlar
      1. Girilen metin ile başlayanları
      2. İçerisinde girilen metin geçenleri
      3. Girilen metin ile bitenleri, arayabilir.
    2. Arama işlemini tetikler.
    3. Arama sonucu bulunan gruplar listelenir.
  - vi. Kullanıcı grubu eklemek.
    1. Kullanıcı grubu arar
    2. Opsiyona bağlı olarak alt grup ekleyeceği grubu seçer.
    3. Gruba ait bilgileri girer
    4. Grup kodu
      - a. Grup kodu 5 karakter olacaktır.
    5. Grup adı
      - a. Grup adı 50 karakter olacaktır.
    6. Grup tanımı
      - a. Grup tanımı 250 karakter olacaktır. Bilgilerini girer.
    7. Kullanıcı grubu ekle işlemini tetikler.
    8. Girilen grup kodu ve adı varsa kullanıcıya hata döndürülür, yoksa kullanıcı grubu sisteme eklenir.
  - vii. Kullanıcı grubu bilgileri güncellemek.
    1. Grubu arar.
    2. Bilgileri güncellenecek grubu seçer.
    3. Yeni grup kodunu girer.
    4. Yeni grup adını girer.
    5. Yeni grup açıklamasını girer.
    6. Grubu güncelle işlemini tetikler.

7. Girilen grup kodu ve adı varsa kullanıcıya hata döndürülür, yoksa kullanıcı grubu yeni bilgiler ile güncellenir.
- viii. Kullanıcı grubu silmek.
1. Grubu arar.
  2. Silinecek grubu seçer.
  3. Grubu sil işlemini tetikler.
  4. Grup altında alt gruplar ve/veya kullanıcılar varsa hata döndürülür, yoksa grup sistemden silinir.
- ix. Kullanıcıyı gruba eklemek.
1. Kullanıcıyı arar.
  2. Gruba ekleyeceği kullanıcıyı seçer.
  3. Kullanıcıyı gruba ekle işlemini tetikler.
  4. Kayıtlı olduğu kullanıcı grupları listelenir.
    - a. Kullanıcı grubu arar.
    - b. Kullanıcıyı dahil etmek istediği grupları seçer.
  5. Gruba ekle işlemini tetikler.
  6. Kullanıcı seçilen gruplara eklenir.
- x. Kullanıcıyı gruptan çıkarmak
1. Kullanıcıyı arar.
  2. Grup çıkaracağı kullanıcıyı seçer.
  3. Kayıtlı olduğu kullanıcı grupları listelenir.
  4. Kullanıcıdan grup çıkar işlemini tetikler.
  5. Grup çıkar işlemini tetikler.
  6. Kullanıcı o gruptan çıkarılır.
- xi. Gruba kullanıcı eklemek.
1. Grubu arar.
  2. Kullanıcı ekleyeceği grubu seçer.
  3. Gruba kullanıcı ekle işlemini tetikler.
  4. Gruba kayıtlı kullanıcılar listelenir.

- a. Kullanıcıyı arar.
  - b. Gruba dahil etmek istediđi kullanıcıları seçer.
5. Kullanıcıyı gruba ekle işlemini tetikler.
  6. Gruba seçilen kullanıcı eklenir.
- xii. Gruptan kullanıcı çıkarmak.
1. Grubu arar.
  2. Kullanıcı çıkaracağı grubu seçer.
  3. Gruba kayıtlı kullanıcılar listelenir.
  4. Gruptan kullanıcı çıkar işlemini tetikler.
  5. Kullanıcı o gruptan çıkarılır.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC3: ROL VE YETKİLER MODÜLÜ**

### **KAPSAM**

Rol ve yetkiler ile ilgili işlemler

### **SEVIYE**

Admin

### **AKTÖR**

Sistem Yöneticisi

## ÖN ŞARTLAR

### BAŞARILI SENARYO SONUCU

- Oluşturulan rolün sistemde tanımlı ve erişilebiliyor olması, kullanıcılara bu rolün atanabilmesi.
- Güncellenen rolün sistemde yeni halini alması.

### TETİKLEYİCİ

### BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar,

- a. Roller ve yetkiler ile ilgili bağlantı aracılığı ile sayfaya ulaşır.
  - i. Kullanıcı yapmak istediği işlemi belirler.
    1. Rol tanımlamak.
      - a. Rol numarası bilgisini girer.
      - b. Rol adı bilgisini girer.
      - c. Sistemde varolan yetkilerden istediklerini seçer.
      - d. Rol tanımla işlemini tetikler.
      - e. Girilen rol sistemde daha önceden tanımlı ise kullanıcıya hata döndürülür, değilse rol sistem üzerinde tanımlanır.
    2. Rol güncellemek.
      - a. Sistemde varolan roller listelenir
      - b. Güncelleme işlemi yapacak rolü seçer.
      - c. Rolün içerdiği yetkiler listelenir.
      - d. Kullanıcı yapacağı işlemi belirler
        - i. Role adını güncellemek

1. Yeni rol adını girer.
  - ii. Role yetki eklemek
    1. Sistemde varolan yetkiler listelenir.
    2. Kullanıcı eklemek istediđi yetkileri seęer.
    3. Role yetki ekle işlemini tetikler.
    4. Seęilen yetkiler daha önceden varsa kullanıcıya hata döndürölür, yoksa seęilen yetkiler role eklenir.
  - iii. Rolden yetki çıkarmak
    1. Roldeki yetkiler listelenir.
    2. Kullanıcı çıkartmak istediđi yetkileri seęer.
    3. Rolden yetki çıkar işlemini tetikler.
    4. Yetki çıkarma işlemleri için kullanıcıdan onay alınır, kullanıcı işlemleri onaylarsa yetkiler rolden çıkarılır.
  - e. Rol güncelle işlemleri tetikler.
  - f. Rol sistemde güncellenir.
3. Rol silmek
    - a. Sistemde varolan roller listelenir
    - b. Silinecek rolü seęer.
    - c. Rol silme işlemleri tetikler.
    - d. Sistemde o rolün atandıđı kullanıcı grubu ve/veya kullanıcı varsa kullanıcıya hata mesajı döndürölür.
    - e. Rolün atandıđı kullanıcı grubu ve/veya kullanıcı yoksa rol silinir.
  4. Grup arama

a. Arama kriterleri

i. Grup adı aramak

1. Arama yaparken kullanacağı opsiyonlar

- a. Girilen metin ile başlayanları
- b. İçerisinde girilen metin geçenleri
- c. Girilen metin ile bitenleri,arayabilir.

b. Arama işlemini tetikler.

c. Arama sonucu bulunan gruplar listelenir.

5. Gruba rol/yetki atamak

a. Grup arar.

b. Rol/yetki atayacağı grubu seçer.

c. Grubun varolan roller ve yetkileri listelenir.

d. Kullanıcı yapacağı işlemi belirler.

i. Rol ekle

1. Sistemde varolan roller listelenir.
2. Ekleyeceği roller seçer
3. Rol ekle işlemini tetikler.
4. Eklenen roller grupta varsa hata dönderilir, yoksa seçilen roller eklenir.

ii. Yeti ekle

1. Sistemde varolan yetkiler listelenir
2. Ekleyeceği yetkileri seçer.
3. Yetki ekle işlemini tetikler.
4. Eklenen yetkiler grupta varsa hata dönderillir, yoksa seçilen yetkiler eklenir.

6. Grubun rol/yetkilerini çıkarmak

- a. Grup arar.
- b. Rol/yetki çıkaracağı grubu seçer.
- c. Grubun varolan roller ve yetkileri listelenir.
- d. Kullanıcı yapacağı işlemi belirler.
  - i. Rol silme
    1. Grupta varolan roller listelenir.
    2. Sileceği rolleri seçer.
    3. Rol sil işlemini tetikler.
    4. Rolü silme için kullanıcıdan onay alınır, onaylarsa roller gruptan silinir.
  - ii. Yetki silme
    1. Grupta varolan yetkiler listelenir.
    2. Sileceği yetkileri seçer.
    3. Yetki sil işlemini tetikler.
    4. Yetki silme için kullanıcıdan onay alınır, onaylarsa yetkiler gruptan silinir.

7. Kullanıcı arama

- a. Arayacağı kullanıcı adını girer.
  - i. Arama yaparken kullanacağı opsiyonlar
    1. Girilen metin ile başlayanları
    2. İçerisinde girilen metin geçenleri
    3. Girilen metin ile bitenleri arayabilir.
- b. Kullanıcı arayacağı grubu opsiyona bağlı olarak seçer.
- c. Arama işlemini tetikler.
- d. Arama sonucu bulunan kullanıcılar listelenir.

8. Kullanıcıya rol/yetki atamak

- a. Kullanıcı arar.
- b. Rol/yetki atayacağı kullanıcıyı seçer.
- c. Kullanıcının varolan rol ve yetkileri listelenir.
- d. Kullanıcı yapacağı işlemi belirler.
  - i. Rol ekle
    1. Sistemde varolan roller listelenir.
    2. Ekleyeceği rolleri seçer
    3. Rol ekle işlemini tetikler.
    4. Eklenen roller kullanıcıda varsa hata dönderilir, yoksa seçilen roller eklenir
  - ii. Yetki ekle
    1. Sistemde varolan yetkiler listelenir
    2. Ekleyeceği yetkileri seçer.
    3. Yetki ekle işlemini tetikler.
    4. Eklenen yetkiler kullanıcıda varsa hata dönderillir, yoksa seçilen yetkiler eklenir.

9. Kullanıcının rol/yetkilerini çıkarmak

- a. Kullanıcı arar.
- b. Rol/yetki çıkaracağı kullanıcıyı seçer.
- c. Kullanıcının varolan rol ve yetkileri listelenir.
- d. Kullanıcı yapacağı işlemi belirler.
  - i. Rol silme
    1. Kullanıcıda varolan roller listelenir.
    2. Sileceği rolleri seçer.
    3. Rol sil işlemini tetikler.
    4. Rolü silme için kullanıcıdan onay alınır, onaylarsa roller gruptan silinir.
  - ii. Yetki silme



1. Kullanıcıda varolan yetkiler listelenir.
2. Sileceđi yetkileri seçer.
3. Yetki sil işlemini tetikler.
4. Yetki silme için kullanıcıdan onay alınır, onaylarsa yetkiler gruptan silinir.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC4: ÜNVAN TANIMLAMA MODÜLÜ**

### **KAPSAM**

Ünvan tanımlama işlemleri.

### **SEVIYE**

Admin

### **AKTÖR**

Sistem Yöneticisi

### **ÖN ŞARTLAR**

### **BAŞARILI SENARYO SONUCU**

- Eklenen ünvanların sistemde tanımlı ve erişilebilir olması

## TETİKLEYİCİ

## BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar,

1. Admin ise;
  - a. Ünvan tanımı ile ilgili bağlantı aracılığı ile ünvan tanımlama sayfasına ulaşır.
  - b. Kullanıcı yapmak istediği işlemi belirler.
    - i. Yeni ünvan tanımlamak.
      1. Ünvan kodu bilgisini girer.
      2. Ünvan kodu bilgi max 5 karakter olmalıdır.
      3. Ünvan tipi bilgisini girer.
      4. Akademik (Öğr.Grv., Yrd.Doç.Dr., Doç.Dr., Prof.Dr. vs.)
      5. İdari (Bölüm Sekreteri, Bölüm Başkanı, Dekan, Dekan Yrd. Vs.)
      6. Ünvan adı bilgisini girer.
      7. Ünvan adı max. 50 karakter olmalıdır.
      8. Ünvan ekle işlemini tetikler.
      9. Eklenen ünvan kodu mevcut ünvanlar kodları arasında varsa kullanıcıya hata bilgisi verilir, yoksa yeni ünvan kaydedilir.
    - ii. Ünvanı güncellemek.
      1. Sistemde varolan ünvalar listelenir.
      2. Güncelleme işlemi yapılacak ünvanı seçer.
      3. İlgili ünvan bilgilerini günceller.
      4. Ünvan kodu
      5. Ünvan adı
      6. Ünvan güncelle işlemini tetikler.

7. Ünvan kodu bilgisi sistemde daha önceden tanımlı ise hata bilgisi döndürülür, değilse ünvan bilgisi yeni bilgiler ile güncellenir

iii. Ünvan silmek

1. Silinecek ünvanı seçer.
2. Silme işlemi tetikler.
3. Sistemde silinecek ünvanı taşıyan kişiler varsa kullanıcıya hata bilgisi verilir.
4. Yoksa ünvan silinir.
5. Silme işlemi ile ilgili log tutulur.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC5: OKUL TANIMLAMA MODÜLÜ**

### **KAPSAM**

Okul tanımlama işlemleri.

### **SEVIYE**

Admin

### **AKTÖR**

Sistem Yöneticisi

## ÖN ŞARTLAR

## BAŞARILI SENARYO SONUCU

- Eklenen okulların sistemde tanımlı ve erişilebilir olması.

## TETİKLEYİCİ

## BAŞARILI SENARYO ADIMLARI

Admin sisteme giriş yapar,

1. Okul tanımlama ile ilgili bağlantı aracılığı ile okul tanımlama sayfasına ulaşır.
2. Yapmak istediği işlemi belirler.
  - a. Yeni okul tanımlamak.
    - i. Okul kodu verisini girer
      1. Okul kodu 2 karakter olmalıdır.
    - ii. Okul adı verisini girer.
      1. Okul adı max 250 karakter olmalıdır.
    - iii. Okul tipini girer.
      1. Okul tipi fakülte, enstitü ve yüksek okul olabilir.
    - iv. Okul dekanını seçer.
    - v. Okul ekle işlemini tetikler.
    - vi. Mevcut okul kodu daha önceden oluşturulmuş ise kullanıcıya hata bilgisi döndürülür, değilse okul sisteme eklenir.
  - b. Okul güncellemek.
    - i. Varolan okullar listelenir.
    - ii. Kullanıcı yapacağı işlemi belirler.
      1. Okul bilgilerini güncellemek
        - a. Güncelleme işlemi yapılacak okulu seçer.

- b. Okul Kodu bilgisi
  - c. Okul Adı bilgisi
  - d. Okul Tipi bilgisi
  - e. Okul Dekanı, bilgilerini girer
  - f. Okul güncelle işlemini tetikler.
  - g. Okul kodu ve adı bilgileri sistemde daha önceden tanımlı ise hata bilgisi gönderilir, değilse okul bilgisi yeni bilgiler ile güncellenir.
- c. Okulu silmek
- i. Silme işlemi yapılacak okulu seçer.
  - ii. Silme işlemini tetikler.
  - iii. Okulda kayıtlı bölümler varsa kullanıcıya hata döndürülür.
  - iv. Yoksa okul sistemden silinir.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC6: BÖLÜM TANIMLAMA MODÜLÜ**

### **KAPSAM**

Bölüm tanımlama işlemleri.

## SEVIYE

Admin

## AKTÖR

Sistem Yöneticisi

## ÖN ŞARTLAR

## BAŞARILI SENARYO SONUCU

- Eklenen bölümlerin sistemde tanımlanması ve erişilebilir olması.

## TETİKLEYİCİ

## BAŞARILI SENARYO ADIMLARI

Admin sisteme giriş yapar,

1. Bölüm tanımlamayla ilgili bağlantı aracılığı ile bölüm tanımlama sayfasına ulaşır.
2. Yapmak istediği işlemi belirler.
  - a. Yeni bölüm tanımlamak.
    - i. Bölüm kodu verisini girer.
      1. Bölüm kodu 2 karakter olmalıdır.
    - ii. Bölüm adı verisini girer.
      1. Bölüm adı max. 250 karakter olmalıdır.
    - iii. Bölüm açacağı okulu seçer.
    - iv. Bölüm başkanını seçer.
    - v. Bölüm sekreterini seçer.
    - vi. Bölüm ekle işlemini tetikler.
    - vii. Bölüm kodu sistemde daha önceden tanımlanmış ise kullanıcıya hata döndürülür, aksi halde yeni bölüm eklenir.
  - b. Bölüm güncellemek.
    - i. Bölümler listelenir.

- ii. Güncelleme işlemi yapılacak bölümü seçer.
  - iii. Bölüm adı girer
  - iv. Okul seçer
  - v. Bölüm başkanı seçer
  - vi. Bölüm sekreteri seçer
  - vii. Bölüm güncelle işlemini tetikler.
  - viii. Bölüm bilgisi yeni bilgiler ile güncellenir
- c. Bölüm silmek
- i. Bölümler listelenir.
  - ii. Sileceği bölümü seçer.
  - iii. Silme işlemini tetikler.
  - iv. Bölümde kayıtlı program varsa kullanıcıya hata döndürülür.
  - v. Yoksa bölüm silinir.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC7: EĞİTİM KATALOĞU MODÜLÜ**

### **KAPSAM**

Eğitim kataloğu (Program) ekleme modülündeki işlemler

## SEVIYE

Admin

## AKTÖR

Sistem Yöneticisi, Bölüm Sekreteri

## ÖN ŞARTLAR

## BAŞARILI SENARYO SONUCU

- Yeni programların sisteme başarılı bir şekilde kaydedilmesi.
- Kullanıcıların sistemdeki tüm programları görebilmesi.

## TETİKLEYİCİ

## BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar.

1. Sistem Yöneticisi ise,
  - a. Programlar ile ilgili bağlantıyı kullanarak program tanımlama ile ilgili sayfaya ulaşır.
  - b. Yapacağı işlemi belirler.
    - i. Program arama
      1. Arayacağı program adını girer
        - a. Arama yaparken kullanacağı opsiyonlar
          - i. Girilen metin ile başlayanları
          - ii. İçerisinde girilen metin geçenleri,
          - iii. Girilen metin ile bitenleri arayabilir.
      2. Arama işlemini tetikler.
      3. Programlar listelenir.
    - ii. Yeni program eklemek.
      1. Program kodu bilgisini girer.



- a. Program kodu max. 2 karakter olmalıdır.
  2. Program adı bilgisini girer.
    - a. Program adı max. 250 karakter olmalıdır.
  3. Program tipi bilgisini girer.
    - a. Ön lisans
    - b. Lisans
    - c. Yüksek lisans
  4. Programın açıklama bilgisini girer.
    - a. Program açıklama bilgisi max. 250 karakter olmalıdır.
  5. Programın açılacağı bölüm bilgisini seçer.
  6. Programın başvurulabilir olup/olmama bilgisini belirler.
  7. Programın katalogda görünüp/görünmeme bilgisini belirler.
  8. Program ekleme işlemini tetikler.
  9. Program kodu bilgileri sistemde daha önceden varsa kullanıcıya hata bilgisi döndürülür, yoksa program eklenir.
- iii. Programı güncellemek.
1. Program arar.
  2. Güncelleme işlemi yapılacak program seçer.
  3. İlgili program bilgilerini günceller
    - i. Program adı
    - ii. Program tipi
    - iii. Program açıklaması
    - iv. Program açacağı bölüm
    - v. Programın başvurulabilirliği
    - vi. Programın katalogda görünüp/görünmeme bilgisi
  4. Program güncelle işlemini tetikler

5. Program kodu bilgileri sistemde daha önceden tanımlı ise hata bilgisi gönderilir, değilse program bilgisi yeni bilgiler ile güncellenir

iv. Program silmek

1. Program arar.
2. Silinecek program seçer.
3. Silme işlemini tetikler.
4. Programa bağlı dersler varsa kullanıcıya hata bilgisi döndürülür.
5. Yoksa program silinir.

v. Programa dönem eklemek

1. Program arar.
2. Dönem eklenecek program seçer.
3. Eklenecek dönem bilgilerini girer
  - a. Dönem no girer
  - b. Minimum krediyi girer.
  - c. Maksimum krediyi girer.
  - d. Zorunlu/Seçmeli ders sayısını girer.
4. Dönem ekle işlemini tetikler.
5. Dönem no bilgileri sistemde daha önce tanımlı ise hata bilgisi döndürülür, değilse dönem programa eklenir.

vi. Programdan dönem silmek.

1. Program arar.
2. Dönemi silinecek program seçer.
3. Dönemler listelenir.
4. Silinecek dönemi seçer.
5. Dönem sil işlemini tetikler.
6. Dönem aktif bir programda yer alıyorsa kullanıcıya hata bilgisi döndürülür, yer almıyorsa dönem programdan silinir.

vii. Döneme Ders ekleme.

1. Program arar
2. Program seçer.
3. Dönemler listelenir.
4. Ders ekleyeceği dönemi seçer
5. Döneme ders ekle işlemini tetikler
  - a. Okul seçer.
  - b. Blüm seçer.
  - c. Bölüm altındaki dersler listelenir.
6. Eklemek istediği dersi seçer
7. Kredi bilgisini girer.
8. Açık/Kapalı durumunu belirler.
9. Zorunlu/Seçmeli durumunu belirler
10. Görünüp/Görünmeme durumunu belirler.
11. ön koşulu bilgisini girer.(metin alan)
12. Ders zorunlu/seçmeli ise grubu bilgisini girer.
13. Ders ekle işlemini tetikler.
14. Ders belirtilen döneme eklenir.

viii. Dönemden ders silme

1. Program arar
2. Program seçer.
3. Dönemler listelenir.
4. Ders sileceği dönemi seçer
5. Dönemde varolan dersler listelenir.
6. Ders sil işlemini tetikler.
7. Ders belirtilen dönemden silinir.

ix. Döneme Zorunlu/Seçmeli ders grubu eklemek.

1. Program seçer.
2. Dönemler listelenir.

3. Ders grubu ekleyeceđi dönemi seçer
  4. Grup kodu bilgisini girer.
  5. Grup adı bilgisini girer.
  6. Grup açıklama bilgisini girer.
  7. Grupta seçilecek ders sayısı bilgisini girer.
  8. Grup ekleme işlemini tetikler.
  9. Zorunlu/Seçmeli ders grup döneme eklenir.
  10. Bu gruba döneme ders ekleme işlemini kullanarak zorunlu/seçmeli dersleri ekler.
- x. Dönemden zorunlu/Seçmeli ders grubunu silmek
1. Program seçer.
  2. Dönemler listelenir.
  3. Ders grubu sileneđi dönemi seçer
  4. Zorunlu/Seçmeli ders grupları listelenir.
  5. Varolan gruplardan silinecek grubu seçer.
  6. Silme işlemini tetikler.
  7. Kullanıcıya grup ve altındaki derslerin silineceđi bildirilerek onay alınır
  8. Onaylarsa zorunlu/Seçmeli ders grubu ve altındaki dersler dönemden silinir.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## UC8: DERSLER MODÜLÜ

### KAPSAM

Ders içeriđi ekleme işlemleri, ders sınavlarının deđerlendirmeye etkisinin belirlenmesi, Ödevlerin deđerlendirmeye etkisinin belirlenmesi

### SEVIYE

Admin

### AKTÖR

Sistem Yöneticisi, Bölüm sekreteri

### ÖN ŞARTLAR

### BAŞARILI SENARYO SONUCU

- Ders içeriklerinin sisteme başarılı bir şekilde kaydedilmesi.
- Kullanıcıların sistemdeki derslerini görebilmesi.

### TETİKLEYİCİ

### BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar,

Admin ise;

1. Dersler ile ilgili bağlantıyı kullanarak dersler sayfasına ulaşılır.
2. Yapmak istediđi işlemi belirler.
  - a. Ders ekleme.
    - i. Okul seçer
    - ii. Ders içeriđi ekleyeceđi bölümü seçer. (ders izin kodu)
    - iii. Seçilen bölüm içerisindeki varolan dersler listelenir.
    - iv. Ders kodu bilgisini girer.

1. Ders kodu 5 karakter
  - v. Ders adı bilgisini girer.
    1. Ders adı bilgisi max 100 karakter.
  - vi. Ders tanımını bilgisini girer.
    1. Ders tanımı max. 1000 karakter olacaktır.
  - vii. Ders tipi bilgisini girer.(LMS)
  - viii. Ders içeriklerinin yer alacağı adres bilgisini girer.
    1. Derse ait alt konuların scorm paketlerinin yer alacağı adres alınacaktır.
  - ix. Ders hedef bilgisini girer.
    1. Metin alanıdır, varsa dersin hedefleri hakkında kısa açıklama verir.
  - x. Kaydet işlemini tetikler.
  - xi. Ders kodu daha önceden tanımlı ise kullanıcıya hata bilgisi döndürülür, değilse ders tanımlanır.
  - xii. Ders alt konularını ekleme
    1. Alt konu bilgilerini girer.
      - a. Konu Sırası
      - b. Konu metni
      - c. Konu Scrom paketi adresi
      - d. Konu aktif/pasif
    2. Alt konu ekle işlemini tetikler.
    3. Aynı sıra numarasına sahip konu yoksa alt konu mevcut konular arasına eklenir.
- b. Dersin içeriğini güncellemek.
- i. Okul seçer
  - ii. Ders içeriği güncelleyeceği bölümü seçer.(ders dizin kodu)
  - iii. Seçilen bölüm içerisinde varolan dersler listelenir
  - iv. İçeriğini güncelleyeceği dersi seçer.
  - v. Ders bilgileri ve alt programlar listelenir.

- vi. İlgili ders bilgilerini girer.
  - 1. Ders adı
  - 2. Ders tanımı
  - 3. Ders tipi
  - 4. Ders içeriklerinin yer alacağı adres
  - 5. Ders hedef bilgilerini girer
- vii. Alt Program ekleyip, silebilir.
- viii. Güncelle işlemini tetikler.
  - ix. Mevcut ders bilgileri yeni bilgiler ile güncellenir.
- c. Ders içeriğini silmek
  - i. Okul seçer
  - ii. Ders içeriği sileceği bölümü seçer.(ders dizin kodu)
  - iii. Seçilen bölüm içerisindeki varolan dersler listelenir
  - iv. Dersler arasından silinecek dersi seçer.
  - v. Silme işlemini tetikler.
  - vi. Silme işlemi için kullanıcıdan onay alınır. Kullanıcı işlemi onaylarsa ders bölüm derslerinde disable edilir.
- d. Ders Sınavlarını belirlemek
  - i. Sınav eklemek
    - 1. Sınavın tipini seçer.
      - a. Vize
      - b. Final
    - 2. Sınavın yıl sonu notunun yüzde kaçını oluşturacağını girer.
    - 3. Sınav tipi adı otomatik olarak verilir.
      - a. Sınav tipi vize ise Vize + Numarası şeklinde olacaktır.
      - b. Sınav tipi final ise DERS ADI + final sınavı şeklinde olacaktır.
    - 4. Sınav ekle işlemini tetikler.

- a. Sınav tipi final ise daha önce final eklenmemişse sınav ilgili derse eklenir, eklenmişse hata mesajı döndürülür.
5. Sınav ilgili derse eklenir.
- ii. Ödev eklemek
  1. Ödevin yıl sonu notunun yüzde kaçını oluşturacağını girer.
  2. Ödev adını girer.
    - a. Ödev adı max. 50 karakter olacaktır.
  3. Ödev ekle işlemini tetikler.
  4. Ödev ilgili dersin ödevleri arasına eklenir.
- iii. Ödev Silmek
  1. Ödevler arasından silinecek ödevi seçer.
  2. Ödev sil işlemini tetikler.
  3. İlgili ödev yayında değilse silinir, yayında ise hata mesajı döndürülür.
- iv. Sınav Silmek
  1. Sınavlar arasından silinecek sınavı seçer.
  2. Sınav aktif değil ise silinir, aktif ise hata mesajı döndürülür.
- e. Ders kaynak bilgileri eklemek
  - i. Kaynak kitap adını girer.
    1. Kitap adı max. 250 karakter olacaktır.
  - ii. Kaynak yayınevi bilgisini girer.
    1. Yayınevi bilgisi max. 100 karakter olacaktır.
  - iii. Kaynak bilgileri ekle işlemini tetikler.
  - iv. Belirtilen ada sahip kaynak daha önce eklenmemişse ders kaynakları arasına eklenir, zaten mevcutsa hata mesajı döndürülür.
- f. Ders kaynak bilgileri silmek
  - i. Kaynak bilgileri arasından silmek istediği kaynağı seçer.
  - ii. Kaynak sil işlemini tetikler.
  - iii. Kaynak mevcut kaynaklar arasından silinir.



- g. Öğrenci performans notunun deęerlendirmeye etkisini belirlemek.
- i. Öğrenci performans notunun yıl sonu notunu yüzde kaç etkileyeceğini girer.
  - ii. Kaydet işlemini tetikler.
  - iii. Belirlenen oran kaydedilir.

User ise;

- 1) Dersler ile ilgili bağlantıyı kullanarak dersler sayfasına ulaşılır.
- 2) Yapmak istedięi işlemi belirler.
  - a) Ders bilgileri görüntülemek.
    - i) Aldığı dersler arasından ders seçer.
    - ii) Derse ait bulgiler güncellenir.
      - (1) Dersin sorumlusu
      - (2) Dersin tanımı
      - (3) Dersin alt konuları
      - (4) Kaynaklar
      - (5) Sınavlar
        - (a) Sınavların yıl sonu notuna katkı payları
      - (6) Ödevler
        - (a) Ödevlerin yıl sonu notuna katkı payları
      - (7) Öğrenci performansının yıl sonu notuna katkı payı
    - b) Ders içerięi görüntülemek
      - i) Aldığı dersler arasından içerięini görüntülemek istedięi dersi seçer.
      - ii) Dersin o an aktif olan alt konuları listelenir.
      - iii) Alt konuyu seçer.
      - iv) İçerięi görüntüle işlemini tetikler.

## **EKLENTİLER**

-

## ÖZEL GEREKSİNİMLER

-

## KESİNLİK KAZANMAYAN NOKTALAR

-

## UC9: HAFTALIK DERS PROGRAMI MODÜLÜ

### KAPSAM

Haftalık ders program modülündeki işlemler

### SEVIYE

Admin, User

### AKTÖR

Bölüm Sekreteri

### ÖN ŞARTLAR

### BAŞARILI SENARYO SONUCU

Admin işlemleri,

- Sisteme girilen saatlerin kullanıcılar tarafından erişilebiliyor olması

User işlemleri,

- Sohbet Saatinin haftalık ders programında görülebilir olması

### TETİKLEYİCİ

### BAŞARILI SENARYO ADIMLARI

1. Bölüm Sekreteri ise ;
  - a. Haftalık ders program işlemleri sayfasına ulaşır.

b. Gerçekleştireceği işlemi belirler.

i. Haftalık ders programı hazırlamak.

1. Program seçer.
2. Yarıyıllara göre gruplanmış şekilde ders sorumlularının derslerini hangi gün ve saatlerde yapabileceği bilgilerini görür.
3. Bu bilgilere haftalık ders programı girişlerini yapar.
  - a. Program seçer.
  - b. Yarıyıl seçer.
  - c. O yarıyıldaki dersler arasından ders seçer.
  - d. Haftanın gününü seçer.
  - e. Başlangıç Saatini girer.
  - f. Bitiş saatini girer.
  - g. Seçtiği programın haftalık ders programına belirtilen ders eklenir.
  - h. Bu işlemi tekrarlayarak dersin diğer günlerini de belirler.

ii. Haftalık ders programı güncellemek.

1. Program seçer.
2. O programın haftalık ders programı görüntülenir.
3. Gün ve saatlerini güncellemek için ders seçer.
  - a. Program seçer.
  - b. Yarıyıl seçer.
  - c. Ders seçer.
  - d. Dersin gün ve saatleri listelenir.
  - e. Güncellemek istediği gün ve saati seçer.
    - i. Yeni gün bilgisini seçer.
    - ii. Yeni başlangıç saatini girer.
    - iii. Yeni bitiş saatini girer.

- f. Güncelle işlemini tetikler.
    4. Haftalık ders programında seçilen ders gün ve saati güncellenir.
  2. Ders sorumlusu ise;
    - a. Haftalık ders program işlemleri sayfasına ulaşır.
    - b. Gerçekleştireceği işlemi belirler.
      - i. Ders saatlerini bildirmek.
        1. Bu bildirimleri bahar/güz yarıyılları başlamadan yapabilecektir.
        2. Verdiği dersler arasından ders saatini belirleyeceği dersi seçer.
        3. Haftanın hangi günü yapmak istedigini seçer.
        4. Başlangıç saati girer.
        5. Bitiş saati girer.
        6. Ders saati bildir işlemini tetikler.
        7. Bildirim ilgili bölüm sekreterine iletilir.
      - ii. Ders saatlerini görüntülemek
        1. Verdiği dersler ve haftalık ders bilgileri görüntülenir.
          - a. Ders Adı
          - b. Haftanın hangi günü olduğu bilgisi
          - c. Saat bilgisi
        2. Ders sorumlusu bu bölümü kullanarak kendi ders saatlerini görüntüleyebilmektedir.
3. Öğrenci ise;
  - a. Haftalık ders program ile ilgili bağlantıyı seçerek haftalık ders programına ulaşır.
  - b. Bulunduğu programın haftalık ders program görüntülenir.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC10: ŞUBE İŞLEMLERİ MODÜLÜ**

### **KAPSAM**

Şube tanımlama işlemleri.

### **SEVİYE**

Admin

### **AKTÖR**

Sistem Yöneticisi, Bölüm Sekreteri

### **ÖN ŞARTLAR**

### **BAŞARILI SENARYO SONUCU**

Admin işlemleri,

- Derslerin şube düzenlemesini yapabilir.
- Şubelere ders sorumlusu atama işlemlerini yapabilir.
- Şubeye öğrenci ekleme ve şube bilgilerini düzenleme işlemlerini yapabilir.

### **TETİKLEYİCİ**

## BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar.

i) Admin ise;

a. Şube işlemleri ile ilgili bağlantı aracılığı ile Şube işlemleri sayfasına ulaşır.

b. Kullanıcı yapmak istediği işlemi belirler.

1. Şube arama

a. Arama kriterleri

i. Şube kodu arama

1) Arama yaparken kullanacağı opsiyonlar

a. Girilen metin ile başlayanları

b. İçerisinde girilen metin geçenleri

c. Girilen metin ile bitenleri, arayabilir.

ii. Arama yapacağı programı opsiyona bağlı olarak seçer.

iii. Arama yapacağı dersi opsiyona bağlı olarak seçer.

b. Arama işlemi tetikler.

c. Arama sonucu bulunan şubeler listelenir.

2. Şube eklemek.

i. Bölüm içerisindeki programlar listelenir.

ii. İşlem yapacağı program seçer.

iii. Programta tanımlı dersler listelenir

iv. Şube işlemi yapacağı dersi seçer.

v. Seçilen derse ait şubeler listelenir.

vi. Şube ekle bağlantısını tetikler.

vii. Şube bilgilerinden,

a. Şube kodu bilgisini girer.

a. Max 10 karakter

b. Şube adı bilgisini girer.

a. Max 25 karakter

- c. Ders sorumlusu seçer.
    - a. Seçilen dersin sorumlusunu öğretim üyeleri listesinden seçer.
    - d. Şube kapasitesi bilgisini girer.
      - a. Max 3 haneli nümerik değer olmalıdır.
  - viii. Şube kaydet işlemini tetikler.
  - ix. Girilen şube kodu ders şubeleri arasında varsa kullanıcıya hata döndürülür, yoksa yeni şube dersin şubelerine eklenir.
3. Şube güncelleme.
- i. Bölüm içerisindeki programlar listelenir
  - ii. İşlem yapacağı program seçer.
  - iii. Programta tanımlı dersler listelenir.
  - iv. Şube işlemi yapacağı dersi seçer.
  - v. Seçilen derse ait şubeler listelenir.
  - vi. Güncelleme yapacağı şubeyi seçer.
  - vii. İlgili şube bilgilerini girer.
    - 1) Şube adı girer.
    - 2) Ders sorumlusu seçer.
    - 3) Şube kapasitesi belirler.
  - viii. Şube güncelle işlemi tetiklenir.
  - ix. Mevcut şube, girilen güncel şube bilgileri ile değiştirilerek seçilen derse ait şubeler arasında listelenir
4. Şube silme.
- i. Bölüm içerisindeki programlar listelenir
  - ii. İşlem yapacağı program seçer.
  - iii. Programta tanımlı dersler listelenir.
  - iv. Şube işlemi yapacağı dersi seçer.
  - v. Seçilen derse ait şubeler listelenir.

- vi. Sileceđi Őubeyi belirler.
- vii. Őube sil iŐlemine tetikler.
- viii. Őubeye eklenmiŐ Őđrenciler varsa kullanıcıya bu bildirilerek hata dŐndürölür, yoksa Őube sistemde disable edilir.

5. Őube Őđrencilerini belirleme

- i. Bölüm iŐerisindeki programlar listelenir.
- ii. IŐlem yapacađı program seŐer.
- iii. Programa bađlı dersler listelenir.
- iv. IŐlem yapacađı dersi seŐer.
- v. Dersi aldıđı halde hiŐbir Őubeye atanmamıŐ Őđrenciler listelenir.
- vi. Őubeye ekleyeceđi Őđrencileri seŐer.
- vii. Őube listesinden Őđrencileri ekleyeceđi Őubeyi belirler.
- viii. Őube belirleme iŐlemini tetikler.
- ix. SeŐilen Őđrencilerin ders Őubesi bilgisini belirler.

6. Őube Őđrencilerini deđiŐtirme

- i. Bölüm iŐerisindeki programlar listelenir.
- ii. IŐlem yapacađı program seŐer.
- iii. Programa bađlı dersler listelenir.
- iv. IŐlem yapacađı dersi seŐer.
- v. Derse ait Őubeler listelenir.
- vi. DeđiŐtirme iŐlemi yapacađı Őubeyi seŐer.
- vii. Őubeye ait Őđrenciler listelenir.
- viii. Őubesi deđiŐtirilecek Őđrenci yada Őđrencileri seŐer.
- ix. Őube listesinden yeni Őubeyi seŐer.
- x. Őube deđiŐtir iŐlemini tetikler.
- xi. Őđrenciler yeni Őubeye atanır, diđer Őubedeki kayıt bilgileri silinir.



## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC11: DANIŞMAN ATAMA İŞLEMLERİ MODÜLÜ**

### **KAPSAM**

Danışman atama işlemleri.

### **SEVIYE**

Admin

### **AKTÖR**

Sistem Yöneticisi, Bölüm Sekreteri, Ders Sorumlusu

### **ÖN ŞARTLAR**

### **BAŞARILI SENARYO SONUCU**

Admin işlemleri,

- Danışmanların bölümlere atanabiliyor olması.
- Danışmanların sistemde tanımlanmış ve erişilebilir olması.

User işlemleri,

- Danışmanların kendilerine atanan programları görebilmesi.
- Danışmanların, danışmanı oldukları öğrencileri görebilmesi.

- Danışmanların, danışmanı oldukları öğrencilerin ders kayıt işlemlerini yapabilmesi.

## **TETİKLEYİCİ**

## **BAŞARILI SENARYO ADIMLARI**

Kullanıcı sisteme giriş yapar,

1. Admin ise;

- a. Danışman atama ile ilgili bağlantı aracılığı ile danışman atama yapısında yapabileceği işlemler sayfasına ulaşır.
- b. Kullanıcı yapmak istediği işlemi belirler.
  - i. Danışman atamak.
    1. Öğrencileri listeler.
      - a. Okul seçer
      - b. Bölüm seçer
      - c. Program seçer.
      - d. Yarıyıl seçer.
      - e. Seçilen yarıyıldaki öğrenciler listelenir.
    2. Öğretim üyelerini listeler
      - a. Okul seçer
      - b. Bölüm seçer
      - c. Seçilen bölümdeki öğretim üyeleri listeler.
    3. Danışmanını tanımlayacağı öğrencileri seçer.
    4. öğrencilere danışman olarak atayacağı öğretim üyesini seçer
    5. Danışman ata işlemini tetikler.
    6. Seçilen öğrencilere girilen öğretim üyesi danışman atanır.
  - ii. Öğrenci danışmanı değiştirme
    1. Öğrencileri listeler.
      - a. Okul seçer
      - b. Bölüm seçer

- c. Program seçer.
    - d. Yarıyıl seçer.
    - e. Seçilen yarıyıldaki öğrenciler listelenir.
  2. Öğretim elemanı listeler
    - a. Okul seçer
    - b. Bölüm seçer
    - c. Seçilen bölümdeki öğretim elemanını listeler.
  3. Danışmanını değiştireceği öğrenci yada öğrencileri seçer.
  4. öğrencilere danışman olarak atayacağı yeni öğretim elemanını seçer
  5. Danışman değiştir işlemini tetikler.
  6. Öğretim üyesi seçilen öğrencilere danışman olarak atanır.
- iii. Danışmanlık iptali
  1. Okul seçer
  2. Bölüm seçer.
  3. Danışmanlık görevi verilmiş öğretim üyeleri listelenir.
  4. Danışmanlık görevi iptal edilecek öğretim üyesini seçer.
  5. Danışmanı olduğu öğrenciler listelenir.
  6. Bölümdeki öğretim üyeleri listelenir.
  7. Danışman olarak atayacağı öğretim üyesini seçer.
  8. Danışman ata işlemini tetikler.
  9. Öğrencilere yeni danışman atanır.
  10. Öğretim elemanını danışmanlık görevi iptal edilmiş olur.

2. User ise;

- a. Danışman işlemleri ilgili bağlantı aracılığı ile danışman işlemleri sayfasına ulaşır.
- b. Kullanıcı yapmak istediği işlemi belirler.
  - i. Danışmanı olduğu öğrencileri görmek.

1. Danışmanı olduğu programı seçer.
2. Seçilen programın bilgilerini listeler.
3. Görüntülemek istediği dönemi seçer.
4. Danışmanı olduğu öğrenciler listelenir.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC12: DERS KAYIT VE DERS ONAY İŞLEMLERİ MODÜLÜ**

### **KAPSAM**

Ders kayıt ve ders onay işlemleri.

### **SEVİYE**

Admin, User

### **AKTÖR**

Sistem Yöneticisi, Ders Sorumlusu, Öğrenci.

### **ÖN ŞARTLAR**

### **BAŞARILI SENARYO SONUCU**

Admin işlemleri,

- Derslerini seçen ve seçmeyen öğrencileri takip edebilir.
- Ders seçimini yapan öğrencilerin, ders onaylama işlemini yapabilir.

User işlemleri,

- Dönemine ait dersleri seçebilir.

## TETIKLEYİCİ

## BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar.

1. Admin ise;
  - a. Ders onayı ile ilgili bağlantı aracılığı ile ders onay sayfasına ulaşır.
  - b. Kullanıcı yapmak istediği işlemi belirler.
    - i. Ders onaylamak.
      1. Danışmanı olduğu öğrenciler listelenir.
      2. Ders kaydını onaylayacağı öğrenciyi seçer.
      3. Öğrencinin ders kayıt ekranı listelenir.
      4. Öğrencinin ilgili programda alması gereken ders listesinden gerekli kontrolü yapar.
      5. Düzeltme işlemi gerekiyorsa, ilgili düzeltmeyi yapar.
      6. Onaylama işlemi tetikler.
      7. Öğrencinin seçili dersleri onaylanır.
      8. Öğrencinin seçtiği dersleri aktif edilir.
2. User ise;
  - a. Ders kayıt işlemleri ile ilgili bağlantı aracılığıyla ders kayıt işlemi sayfasına ulaşır.
  - b. Kullanıcı yapmak istediği işlemi belirler.
    - i. Ders kaydı yapmak.
      1. Öğrencinin o dönemde seçebileceği ders kayıt ekranı karşısına gelir.
      2. Belirlenen kriterler doğrultusunda derslerini seçer.

- a. Dönemde alabileceği kredi oranını geçmemelidir.
  - b. Ders seçerken ön koşulu olan derslerin ilgili bilgileri yerine getirilmiş olmalıdır
  - c. Dönemde alabileceği maksimum minimum kredi kurallarına uymalıdır.
3. Ders kayıt işlemini tetikler.
  4. Dersler sisteme kaydedilir.

#### **EKLENTİLER**

-

#### **ÖZEL GEREKSİNİMLER**

-

#### **KESİNLİK KAZANMAYAN NOKTALAR**

-

### **UC13: SORU TANIMLAMA MODÜLÜ**

#### **KAPSAM**

Soru tanımlama işlemleri

#### **SEVIYE**

Admin

#### **AKTÖR**

Ders Sorumlusu

#### **ÖN ŞARTLAR**

#### **BAŞARILI SENARYO SONUCU**

- Eklenen soruların sistemde tanımlı ve erişilebilir olması

## TETIKLEYICI

## BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar,

1) Admin ise;

i) Servis dersleri kategorisi eklemek

(1) Servis derslerinin altında yer alacağı kategori adını girer.

(a) Kategori adı max 100 karakter olacaktır.

(2) Kategori ekle işlemini tetikler.

(3) Aynı ada sahip kategori sistemde yer almıyorsa kategori sisteme eklenir, yer alıyorsa kullanıcıya hata mesajı döndürülür.

ii) Servis Dersi eklemek

(1) Servis dersi adını girer.

(2) Bu derse soru ekleyebilecek kullanıcıları seçer.

(a) Ders sorumlusu görevinde olan kullanıcılar listelenir.

(b) Yetki vermek istediği kullanıcıları seçer.

(3) Servis dersi ekle işlemini tetikler.

(4) Servis dersi adına sahip başka bir ders yoksa sisteme eklenir, varsa kullanıcıya hata mesajı döndürülür.

2) User ise;

i) Soru tanımlama işlemleriyle ilgili bağlantı aracılığı ile ilgili sayfasına ulaşır.

ii) Kullanıcının verdiği dersler ve soru ekleme yetkisine sahip olduğu servis dersleri listelenir.

iii) İşlem yapacağı dersi seçer.

iv) Soru ekleyeceği dersin alt kategorisini seçer.

v) Kullanıcı yapacağı işlemi belirler.

(1) Soru arama

- (a) Arayacağı soru metnini girer
  - (i) Arama yaparken kullanacağı opsiyonlar
    1. Girilen metin ile başlayanları
    2. İçerisinde girilen metin geçenleri
    3. Girilen metin ile bitenleri, arayabilir
  - (b) Arayacağı soru tipini opsiyona bağlı olarak belirler.
  - (c) Arayacağı soru zorluk derecesini opsiyona bağlı olarak belirler.
  - (d) Arama işlemini tetikler
  - (e) Arama sonucu bulunan sorular listelenir.

(2) Soru ekleme

- (a) Soru tipi seçer(Çoktan tek seçmeli,çoktan çok seçmeli vs.)
- (b) Seçilen tipteki sorular listelenir.
  - (i) Soru tipi Doğru/yanlış olan soru eklemek;
    1. Soru metnini girer.
      - a. Soru metni html formatını destekleyecektir.
    2. Soru zorluk derecesini seçer(1 – Çok Kolay/ 5- Çok Zor gibi)
    3. Soru yetki durumunu seçer.
      - a. Özel
        - i. Sadece kullanıcının kendisinin göreceği sorular
      - b. Genel
        - i. Tüm kullanıcıların görebileceği sorular
      - c. Ders sorumlularına özel
        - i. Tüm ders sorumlularının görebileceği sorular
    4. Varsa soru resmini seçer.
    5. Seçenek ekleme
      - a. Seçenek metnini girer.
        - i. Seçenek metni max 50 karakter.
      - b. Varsa seçenek resmini seçer.



- c. Seçenek ekle işlemini tetikler.
  - d. Seçenek metni soruda varsa kullanıcıya hata mesajı döndürülür.
  - e. Yoksa seçenek soruya eklenir.
6. Soru ekle işlemini tetikler.
  7. Soru metni daha önceden tanımlı ise kullanıcıya hata mesajı verilir, değilse yeni soru sistemde kayıtlı sorular arasına eklenir.
- (ii) Soru tipi Doğru/yanlış olan soru güncellemek;
1. Soru arar
  2. Varolan sorular arasından güncellenecek soruyu seçer.
  3. Yeni soru metnini girer.
  4. Soru zorluk derecesini girer.
  5. Soru yetki durumunu girer.
  6. Varsa soru resmini seçer.
  7. Seçenek güncelleme
    - a. Seçenek ekleyebilir
    - b. Seçenek sil
      - i. Sileceği seçeneği veya resmini seçip sil işlemini tetikler
      - ii. Seçenek sorudan silinir.
    - c. Seçenek güncelle
      - i. Güncelleyeceği seçeneği seçer.
      - ii. Seçenek metnini veya resmini girer
      - iii. Seçenek güncelle işlemini tetikler.
      - iv. Secenek metni soruda varsa kullanıcıya hata döndürülür.
      - v. Yoksa seçenek soruya eklenir.
  8. Soru güncelleme işlemini tetikler.
  9. Mevcut soru bilgileri yeni bilgiler ile güncellenir.
- (iii) Soru tipi Eşleştirmeli olan soru eklemek

1. Soru metnini girer.
  2. Soru zorluk derecesini seçer.
  3. Soru yetki durumunu seçer.
    - a. Özel
      - i. Sadece kullanıcının kendisinin göreceği sorular
    - b. Genel
      - i. Tüm kullanıcıların görebileceği sorular
    - c. Ders sorumlularına özel
      - i. Tüm Ders sorumlularının görebileceği sorular
  4. Varsa soru resmini seçer.
  5. Seçenek ekleme
    - a. Seçenek metnini girer.
    - b. Varsa seçenek resmini seçer.
    - c. Eşleşeceği değer metnini girer.
    - d. Varsa eşleşeceği değer resmini seçer.
    - e. Seçenek ekle işlemini tetikler.
    - f. Seçenek metni soruda varsa kullanıcıya hata mesajı döndürülür.
    - g. Yoksa seçenek soruya eklenir.
  6. Soru ekle işlemini tetikler.
  7. Soru metni daha önceden tanımlı ise kullanıcıya hata mesajı verilir, değilse yeni soru sistemde kayıtlı sorular arasına eklenir.
- (iv) Soru tipi Eşleştirmeli olan soru güncellemek.

1. Soru arar
2. Varolan sorular arasından güncellenecek soruyu seçer.
3. Soru metnini girer.
4. Soru zorluk derecesini seçer.
5. Soru yetki durumunu seçer.
  - a. Özel

- i. Sadece kullanıcının kendisinin göreceği sorular
  - b. Genel
    - i. Tüm kullanıcıların görebileceği sorular
  - c. Ders sorumlularına özel
    - i. Tüm Ders sorumlularının görebileceği sorular
- 6. Varsa soru resmini seçer.
- 7. Seçenek güncelleme
  - a. Seçenek ekleyebilir
  - b. Seçenek sil
    - i. Sileceği seçeneği veya resmini seçip sil işlemi tetikler
    - ii. Seçenek sorudan silinir.
  - c. Seçenek güncelle
    - i. Güncelleyeceği seçeneği seçer.
    - ii. Seçenek metnini veya resmini girer
    - iii. Seçenek güncelle işlemi tetikler.
    - iv. Seçenek metni soruda varsa kullanıcıya hata döndürülür.
    - v. Yoksa seçenek soruya eklenir.
- 8. Soru güncelleme işlemi tetikler.
- 9. Mevcut soru bilgileri yeni bilgiler ile güncellenir.
- (v) Soru tipi çoktan tek seçmeli olan soru eklemek
  - 1. Soru metnini girer.
  - 2. Zorluk derecesini seçer.
  - 3. Soru yetki durumunu seçer.
    - a. Özel
      - i. Sadece kullanıcının kendisinin göreceği sorular
    - b. Genel
      - i. Tüm kullanıcıların görebileceği sorular
    - c. Ders sorumlularına özel

i. Tüm Ders sorumlularının görebileceği sorular

4. Varsa soru resmini seçer.
5. Doğru cevabı girer.
6. Seçenek ekleme
  - a. Seçenek metnini girer.
  - b. Varsa seçenek resmini seçer.
  - c. Seçenek ekle işlemini tetikler.
  - d. Seçenek metni soruda varsa kullanıcıya hata mesajı döndürülür
  - e. Yoksa seçenek soruya eklenir.
7. Soru ekle işlemini tetikler.
8. Soru metni daha önceden tanımlı ise kullanıcıya hata mesajı verilir, değilse yeni soru sistemde kayıtlı sorular arasına eklenir

(vi) Soru tipi çoktan tek seçmeli olan soru güncellemek

1. Soru arar.
2. Varolan sorular arasından güncellenecek soruyu seçer.
3. Soru metnini girer.
4. Zorluk derecesini seçer.
5. Soru yetki durumunu seçer.
6. Varsa soru resmini seçer.
7. Doğru cevabı girer.
8. Seçenek güncelleme
  - a. Seçenek ekleyebilir
  - b. Seçenek sil
    - i. Sileceği seçeneği veya resmini seçip sil işlemini tetikler
    - ii. Seçenek sorudan silinir.
  - c. Seçenek güncelle
    - i. Güncelleyeceği seçeneği seçer.
    - ii. Seçenek metnini veya resmini girer

- iii. Seçenek güncelle işlemini tetikler.
  - iv. Secenek metni soruda varsa kullanıcıya hata döndürülür.
  - v. Yoksa seçenek soruya eklenir.
9. Soru güncelleme işlemini tetikler.
  10. Mevcut soru bilgileri yeni bilgiler ile güncellenir.
- (vii) Soru tipi çoktan çok seçmeli olan soru eklemek
1. Soru metnini girer.
  2. Zorluk derecesini seçer.
  3. Soru yetki durumunu seçer.
    - a. Özel
      - i. Sadece kullanıcının kendisinin göreceği sorular
    - b. Genel
      - i. Tüm kullanıcıların görebileceği sorular
    - c. Ders sorumlularına özel
      - i. Tüm Ders sorumlularının görebileceği sorular
  4. Soru resmini seçer.
  5. Sorunun doğru cevap sayısını girer.
  6. Seçenek ekleme
    - a. Seçenek metnini girer.
    - b. Seçenek resmini seçer.
    - c. Seçeneğin doğru olup olmadığını girer.
    - d. Seçenek ekle işlemini tetikler.
    - e. Seçenek metni soruda varsa kullanıcıya hata mesajı döndürülür.
    - f. Yoksa seçenek soruya eklenir.
  7. Soru ekle işlemini tetikler.
  8. Soru metni daha önceden tanımlı ise kullanıcıya hata mesajı verilir, değilse yeni soru sistemde kayıtlı sorular arasına eklenir

- (viii) Soru tipi çoktan çok seçmeli olan soru güncelleme
1. Soru arar.
  2. Varolan sorular arasından güncellenecek soruyu seçer.
  3. Soru metnini girer
  4. Zorluk derecesini seçer
  5. Soru yetki durumunu belirler
    - a. Özel
      - i. Sadece kullanıcının kendisinin göreceği sorular
    - b. Genel
      - i. Tüm kullanıcıların görebileceği sorular
    - c. Ders sorumlularına özel
      - i. Tüm Ders sorumlularının görebileceği sorular
  6. Soru resmini belirler
  7. Sorunun doğru cevap sayısını girer
  8. Seçenek güncelleme
    - a. Seçenek ekleyebilir
    - b. Seçenek sil
      - i. Sileceği seçeneği veya resmini seçip sil işlemini tetikler
      - ii. Seçenek sorudan silinir.
    - c. Seçenek güncelle
      - i. Güncelleyeceği seçeneği seçer
      - ii. Seçenek metni veya resmini girer.
      - iii. Seçenek güncelle işlemini tetikler.
      - iv. Seçenek metni soruda varsa kullanıcıya hata döndürülür.
      - v. Yoksa seçenek soruya eklenir.
  9. Soru güncelleme işlemini tetikler.
  10. Mevcut soru bilgileri yeni bilgiler ile güncellenir.
- (ix) Soru tipi klasik olan soru eklemek
1. Soru metnini girer.

2. Soru zorluk derecesini seçer.
  3. Soru yetki durumunu olarak seçer.
    - a. Özel
      - i. Sadece kullanıcının kendisinin göreceği sorular
    - b. Genel
      - i. Tüm kullanıcıların görebileceği sorular
    - c. Ders sorumlularına özel
      - i. Tüm Ders sorumlularının görebileceği sorular
  4. Varsa soru resmini seçer.
  5. Soru ekle işlemini tetikler.
  6. Soru metni daha önceden tanımlı ise kullanıcıya hata mesajı verilir, değilse yeni soru sistemde kayıtlı sorular arasına eklenir.
- (x) Soru tipi klasik olan soru güncelleme
1. Soru arar.
  2. Varolan sorular arasından güncellenecek soruyu seçer.
  3. Soru metnini girer.
  4. Zorluk derecesini seçer.
  5. Soru yetki durumunu seçer.
  6. Varsa soru resmini seçer.
  7. Soru güncelleme işlemini tetikler.
  8. Mevcut soru bilgileri yeni bilgiler ile güncellenir.
- (3) Soru silme
- (a) Soru arar.
  - (b) Varolan sorulardan silinecek soruyu seçer.
  - (c) Soru sil işlemini tetikler.
  - (d) Soru aktif bir sınavda kullanılıyorsa kullanıcıya hata döndürülür.
  - (e) Kullanılmıyorsa soru ve soruya ait seçenekler silinir.

## EKLENTİLER

-

## ÖZEL GEREKSİNİMLER

-

## KESİNLİK KAZANMAYAN NOKTALAR

-

## UC14: SINAV MODÜLÜ

### KAPSAM

Sınav modülü ile ilgili işlemler

### SEVIYE

Admin, user

### AKTÖR

Ders Sorumlusu, Öğrenci

### ÖN ŞARTLAR

### BAŞARILI SENARYO SONUCU

Admin İşlemleri

- Ders sorumlusunun ilgili dersin sınavını hazırlayabilmesi.
- Sistemde tanımlı soruları kullanarak sınav azırlayabilmesi.

User İşlemleri

- Öğrencinin ilgili dersine atanmış sınavları yanıtlayabilmesi.

### TETİKLEYİCİ



## BAŞARILI SENARYO ADIMLARI

1. Admin ise;
  - a. Sınav oluşturma ile ilgili bağlantı aracılığıyla sınav tanımlama ekranına ulaşır.
  - b. Yapmak istediği işlemi belirler.
  - c. Yeni sınav hazırlamak
    - i. Ders sorumlusunun verdiği dersler listelenir.
    - ii. Sınav hazırlayacağı dersi belirler.
    - iii. Ders için belirlediği sınavlardan oluşturmak istediğini seçer.
      1. Oluşturduğu vize ve finaller arasından seçer.
    - iv. İlgili şube bilgisini belirler.
    - v. Hazırlayacağı sınav tipini belirler.
      1. Soruları rastgele getirilecek sınav hazırlamak.
      2. Sınavın başlangıç tarihi bilgisini belirler.
      3. Sınavın bitiş tarihi bilgisini belirler.
      4. Sınav süresi bilgisini belirler.
      5. Sınava soru ekler
        - a. Seçili dersle ilgili alt konular listelenir.
        - b. Soru eklenecek alt konuyu seçer.
        - c. Alt konudan istenilen soru sayısını belirler.
        - d. Soruların zorluk derecesini belirler.
        - e. Sorunun sınav içerisindeki puanını belirler.
        - f. Sınav şablonuna soru ekleyi tetikler.
        - g. Belirlenen kriterde sorular rastgele soru bankasından seçilerek sınav şablonuna eklenir
  6. Ders sorumlusu (D.) maddesinde anlatılan adımları tekrarlayarak istediği alt konudan ilgili bilgilerini girmek koşuluyla sınava rastgele soru atama işlemini tekrarlayabilir.

7. Sınavı kaydet işlemini tetikler.
8. Sınava eklenen sorulara verilen puanlar 100 üzerinden olacaktır.bu sebeple sınav soruları puan toplamı bu değeri aşmamalıdır.Aıması durumunda kullanıcıya hata bilgisi döndürülür.
9. Sınav belirtilen şablona uygun olarak sistem üzerinde tanımlanır.
10. Soruları seçilen sınav hazırlamak.
11. Sınavın başlangıç tarihi bilgisini girer.
12. Sınavın bitiş tarihi bilgisini girer.
13. Sınav süresi bilgisini girer.
14. Sınava soru ekler
  - a. Seçilen dersle ilgili alt konular listelenir.
  - b. Soru eklenecek alt konuyu seçer.
  - c. Alt konuda tanımlanmış sorular listelenir.
  - d. Sınav şablonuna eklenecek soruları seçer.
  - e. Sorunun sınav içerisindeki puanını belirler.
  - f. Sınav şablonuna soru ekleyi tetikler.
  - g. Seçili sorular sınav şablonuna eklenir.
15. Ders sorumlusu (D.) maddesinde anlatılan adımları tekrarlayarak istediği alt konudan sınava soru ekleyebilir.
16. Sınavı kaydet işlemini tetikler.
17. Sınava eklenen sorulara verilen puanlar 100 üzerinden olacaktır.bu sebeple sınav soruları puan toplamı bu değeri aşmamalıdır.Aşması durumunda kullanıcıya hata bilgisi döndürülür.
18. Sınav başlangıç tarihi seçilen kişilerin ajandasına olay olarak eklenir.
  - a. Not başlığı “sınav” olacaktır.

b. Not metni belirlenen formatta olacaktır.

19. Seçilen sorulardan oluşturulan sınav sistem üzerinde tanımlanır.

d. Sınav güncellemek

i. Sınav başlama tarihi gelmemiş olan sınavlar listelenir.

ii. Düzenleme işlemi yapılacak sınavı seçer.

1. Soruları rastgele getirilen sınavlar.

2. Sınavın başlangıç tarihi bilgisini girer.

3. Sınavın bitiş tarihi bilgisini girer.

4. Sınav süresi bilgisini girer.

5. Sınavı oluşturan alt konu ve bilgileri listelenir.

a. Düzenle işlemi tetikler

b. Alt konu bilgisini belirler.

c. Soru sayısı bilgisini belirler.

d. Soru derecesi bilgisini belirler

e. Soru puanı bilgisini belirler.

f. Alt konuyu güncelle işlemi tetikler.

g. Alt konudan oluşturulmuş soru bilgileri yeni bilgiler ile güncellenir.

6. Ders sorumlusu (D.) de belirtilen maddeleri tekrarlayarak diğer alt konuda hazırlanmış soruları günceller.

7. Sınavı güncelle işlemi tetikler.

8. Sınava eklenen sorulara verilen puanlar 100 üzerinden olacaktır.bu sebeple sınav soruları puan toplamı bu değeri aşmamalıdır.Aşması durumunda kullanıcıya hata bilgisi döndürülür.

9. Mevcut sınav bilgileri yeni bilgiler ile güncellenir.

10. Soruları seçilerek hazırlanan sınavlar.

11. Sınavın başlangıç tarihi bilgisini girer.
  12. Sınavın bitiş tarihi bilgisini girer.
  13. Sınav süresi bilgisini girer.
  14. Sınavı oluşturan alt konu ve o konuda seçilmiş sorular listelenir.
    - a. Düzenle işlemini tetikler.
    - b. Alt konu bilgisini belirler.
    - c. Alt konudan sınava eklenecek soruları seçer.
    - d. Sorunun sınav içerisindeki puanını belirler.
    - e. Alt konuyu güncelle işlemini tetikler.
    - f. Alt konudan seçilmiş soru bilgileri yeni bilgiler ile güncellenir.
  15. Ders sorumlusu (D.) de belirtilen maddeleri tekrarlayarak diğer alt konuda hazırlanmış soruları günceller
  16. Sınavı güncelle işlemini tetikler.
  17. Sınava eklenen sorulara verilen puanlar 100 üzerinden olacaktır.bu sebeple sınav soruları puan toplamı bu değeri aşmamalıdır.Aşması durumunda kullanıcıya hata bilgisi döndürülür.
  18. Mevcut sınav bilgileri yeni bilgiler ile güncellenir.
- e. Sınavı silmek
- i. Tanımlı sınavlar listelenir.
  - ii. Silme işlemi yapılacak sınavı belirler.
  - iii. Sınavı sil işlemini tetikler.
  - iv. Sınav silinmeden yapılacak kontroller
    1. Sınav başlama tarihi gelmemiş olan bir sınav siliniyorsa kullanıcıdan silme işlemi için onay alınır.

2. Sınav tarihi gemiř bir sınav ise sınavı kullananlar varsa bu bilgi ile birlikte kullanııcıdan onay alınır.Sınav disable edilerek kullanııcılara gsterilmez.
  - v. Seili sınav ve sınava baėlı btn bilgiler disable edilir.
2. User ise;
- a. Dersi seer.
  - b. Sınavlarla ilgili baėlantı aracılıėı ile sınav sayfasına ulařır.
  - c. İlgili sınavı seer.
  - d. Yapmak istediėi iřlemi belirler.
  - e. Sınavı bařlatmak.
    - i. Sınav soruları karřısına getirilir.
    - ii. Sınav sresi ierisinde soruları yanıtlar.
    - iii. Her soru yanıtladıėında ilgili sınavın ilgili soru cevabıda iřlenir.
    - iv. Sınavı tamamlama iřlemini tetikler.
    - v. İlgili cevaplar sisteme kaydedilir.
    - vi. ėrencinin sınavdan aldıėı puan hesaplanarak sisteme kaydedilir.

## **EKLENTİLER**

-

## **ZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC15: DENEME SINAVI MODL**

## **KAPSAM**

Deneme sınavı işlemleri.

## SEVİYE

User

## AKTÖR

Öğrenci

## ÖN ŞARTLAR

## BAŞARILI SENARYO SONUCU

- Öğrencinin deneme sınavını hazırlayıp kullanabilmesi.

## TETİKLEYİCİ

## BAŞARILI SENARYO ADIMLARI

Öğrenci sisteme giriş yapar.

1. Deneme sınavı bağlantısı ile deneme sınavı düzenleme sayfasına ulaşır.
  - a. Yeni deneme sınavı işlemi.
    - i. Öğrencinin almış olduğu dersler listelenir.
    - ii. Deneme sınavı hazırlanacak ders seçer.
    - iii. Dersin alt konusunu seçer.
    - iv. Deneme sınavına eklenecek sorunun zorluk derecesini seçer.
    - v. Zorluk derecesinde sınava eklenecek sorusu sayısını seçer.
      1. Zorluk derecesi boş geçilmesi halinde rastgele zorluk derecesinde belirtilen sayıda soru getirilir.
    - vi. Sınava soru ekle işlemi tetiklenir.
    - vii. Sorular deneme sınavı şablonuna eklenir.
      1. Öğrenci istediği kadar (iii ile vii adımları arasındaki işlemleri tekrarlayarak ) sınav şablonuna soru ekler.
    - viii. Sınavı oluştur işlemi tetiklenir.

- ix. Seçilen kriterlere göre hazırlanan deneme sınavı oluşturulur.
- x. Deneme sınavı başlat işlemini tetikler.
- xi. Soruları yanıtlayıp, sınavı tamamlama işlemini tetikler.
- xii. Deneme sınavı sonuçları,
  1. Doğru sayısı.
  2. Yanlış sayısı.
  3. Boş bırakılan sayısı.
  4. Alınan puan, olarak kullanıcıya yansıtılır.

#### **EKLENTİLER**

-

#### **ÖZEL GEREKSİNİMLER**

-

#### **KESİNLİK KAZANMAYAN NOKTALAR**

-

### **UC16: ÖDEVLER MODÜLÜ**

#### **KAPSAM**

Ödevler modülündeki işlemler

#### **SEVIYE**

Admin, User

#### **AKTÖR**

Öğrenci, Öğretim Üyesi

#### **ÖN ŞARTLAR**

## BAŞARILI SENARYO SONUCU

Admin işlemleri,

- Yeni ödevin sisteme eklenmesi ve ilgili kullanıcılar tarafından görülüyor olmasıdır.
- Girdiği ödevi sildiğinde programda bir daha gözükmeyecek şekilde sistemden o ödevin kaldırılması.

User işlemleri,

- Kullanıcıların kendisine ait ödevleri görebiliyor olmasıdır.
- Kullanıcıların yapmış oldukları ödevleri gönderebilmesi ve ödev veren kişiye ulaşması.

## TETİKLEYİCİ

## BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar,

i) Admin ise;

- a. İlgili bağlantıyı kullanarak Ödev işlemi sayfasına ulaşır.
- b. Kullanıcının verdiği dersler listelenir.
- c. Ödev işlemi yapılacak dersi listeden seçer.
- d. Kullanıcı yapmak istediği işlemi belirler.
  1. Yeni ödev eklemek
    - i. Ödev adını girer.
      - a) Ödev adı max 50 karakter olmalıdır.
    - ii. Ödev ilan tarihini girer.
    - iii. Ödev teslim tarihini girer.
    - iv. İlgili ödev dosyasını ekler.
      - a) İlgili ödev dosyası 1.5 MB'dan fazla olmamalıdır.
    - v. Dersi alan öğrenciler listelenir.
    - vi. Ödev vereceği öğrencileri seçer.



- vii. Ödev ekle işlemini tetikler.
  - viii. Seçilen öğrencilerin ödevinin son teslim tarihi ajandalarına olay olarak eklenir.
    - a. Not başlığı “Ödev” olacaktır
    - b. Not metni belirlenen formatta olacaktır.
  - ix. Yeni ödev belirtilen kullanıcılara atanır.
2. Ödev silmek
- a. Dersleri listelenir.
  - b. Ödev verdiği dersi seçer.
  - c. Verilen ödevler listelenir.
  - d. Sileceği ödevi seçer
  - e. Ödev sil işlemini tetikler.
  - f. Ödev yayında değilse silinecektir.
3. Ödev güncellemek
- a. Dersleri listelenir.
  - b. Ödev verdiği dersi seçer.
  - c. Verilen ödevler listelenir.
  - d. Güncelleyeceği ödevi seçer.
  - e. İlgili ödev bilgilerini girer.
    - i. Ödev adını girer
    - ii. Ödev ilan tarihini girer.
    - iii. Ödev teslim tarihini girer.
    - iv. İlgili ödev dosyasını seçer.
    - v. Dersi alan ve ödev atanmış öğrenciler listelenir.
    - vi. Ödev vereceği öğrencileri seçer.
    - vii. Ödev güncelle işlemini tetikler.
    - viii. Ödev adı sistemde kayıtlı ise kullanıcıya hata döndürülür, kayıtlı değilse ödev güncellenir.

#### 4. Ödev değerlendirmek

- a. Dersleri listelenir.
- b. Ödev kontrolü yapacağı dersi seçer.
- c. Son teslim tarihi dolmuş ödevler listelenir.
- d. Dersle ilgili ödevi gönderen öğrenci ve ödev cevapları listelenir.
- e. İlgili öğrenciyi seçer ve ödevini görüntüler.
- f. Ödev notunu girer.
- g. İlgili öğrencinin ödev notu bilgilerine işlenir.

#### ii) User ise;

- a) Ödevler bölümüne erişebileceği ilgili bağlantıyı seçer.
- b) Kullanıcı ekranına aldığı derslerle ilgili ödevler listelenir.
- c) Kullanıcı yapmak istediği işlemi belirler.

#### 1. Ödevi Göndermek

- i. Hangi ödev gönderilecekse o ödev seçilir.
- ii. Sistem tarihi ile ödevin teslim tarihi karşılaştırılır.
- iii. Eğer ödevin teslim tarihi geçtiyse ödevi gönder bağlantısı kaybolur.
- iv. Eğer teslim tarihi geçmediyse "ödevi gönder" bağlantısını tetikler.
- v. Ödev dosyasını ekler.
- vi. Durum alanı ödev gönderildi olarak değiştirilir.
- vii. Son teslim tarihine kadar istediği kadar ödev gönderebilir.
- viii. Gönderdiği son dosya kabul edilir.

#### **EKLENTİLER**

-

#### **ÖZEL GEREKSİNİMLER**

-

#### **KESİNLİK KAZANMAYAN NOKTALAR**

## UC17: ÖĞRETİM ELEMANI PERFORMANS TAKİP

### KAPSAM

Öğretim elemanı performans takibi

### SEVİYE

Admin

### AKTÖR

Sistem yöneticisi, Bölüm başkanı

### ÖN ŞARTLAR

Kullanıcı sisteme giriş yapmış olmalı.

### BAŞARILI SENARYO SONUCU

- Öğretim elemanını, vermiş olduğu dersin bilgilerine göre değerlendirebilmesi.

### TETİKLEYİCİ

### BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar.

- 1) Öğretim elemanı performans takibi ile ilgili bağlantı aracılığı ile sayfaya ulaşır.
- 2) Yapmak istediği işlemi belirler.
  1. Öğretim elemanı performans takibi
    - i. Bölümde ders veren öğretim elemanları listelenir.
    - ii. Görüntülemek istediği öğretim elemanını seçer.
    - iii. Kullanıcı tarih aralığını belirler.
      - 1) Yıl seçer.
      - 2) Yarıyıl seçer.
    - iv. Öğretim elemanının vermiş olduğu dersler listelenir.

- v. Görüntülemek istediđi dersi seçer.
- vi. Görüntüleme işlemini tetikler.
- vii. Öğretim elamanının seçilen derslerle ilgili yapmış olduđu faaliyetleri listeler.
  - a. Düzenlediđi toplam sohbet saati,
  - b. Sanal sınıfta düzenlediđi toplam ders sayısı
  - c. Verdiđi ödev sayısı
  - d. Sisteme eklediđi toplam soru sayısı
  - e. Dersle ilgili öğrencilerin yapmış oldukları değerlendirme anketi sonuçlarını görebilir.
    - a. Deđerlendirme anketi Bölüm Sekreteri tarafından hazırlanıp ilgili derslere atanır.
    - b. Anket sonuçları;
      - i. Soru gösterilir
      - ii. Cevaplar grafiksel ve sayısal olarak gösterilir.

## **EKLENTİLER**

-

## **ÖZEL GEREKSİNİMLER**

-

## **KESİNLİK KAZANMAYAN NOKTALAR**

-

## **UC18: ÖĞRENCİ PERFORMANS TAKİP MODÜLÜ**

### **KAPSAM**

Öğrencinin kendi değerlendirmeleri.

## SEVİYE

User

## AKTÖR

Öğrenci

## ÖN ŞARTLAR

## BAŞARILI SENARYO SONUCU

- Öğrencinin kendi notları ile birlikte performans notlarını listeleyebilmesi,
- Öğrencinin ders sorumlusu hakkında değerlendirmelerini yapabilmesi.

## TETİKLEYİCİ

## BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar.

i) User ise;

- (a) Öğrenci değerlendirme ile ilgili bağlantı aracılığı ile sayfaya ulaşır.
- (b) Kullanıcı yapmak istediği işlemi belirler.

1. Öğrencinin kendini değerlendirmesi

i. Öğrencinin aldığı dersler ve ilgili ders bilgileri listelenir,

- a. Vize notları
- b. Final notları
- c. Ödev notları
- d. Sohbet katılım saatleri
- e. Derse katılım saatleri
- f. Ders sorumlusunun vermiş olduğu performans notları, görüntülenir.

2. Öğrencinin ders sorumlusu değerlendirmesi

- i. Öğrencinin aldığı dersler listelenir.
- ii. Dersle ilgili değerlendirme anketini görüntüler.
- iii. Anketi doldur işlemini tetikler.
- iv. Anket sorularını görüntüler ve ders sorumlusu ile ilgili soruları yanıtlar.
  - 1) Öğrenci anketi belirlenen tarihlerde ve bir defaya mahsup doldurur.
- v. Değerlendirme anketini kaydeti tetikler.
- vi. Anket bilgileri sisteme kaydedilir.

#### **EKLENTİLER**

-

#### **ÖZEL GEREKSİNİMLER**

-

#### **KESİNLİK KAZANMAYAN NOKTALAR**

-

### **UC19: ÖĞRENCİ TAKİP RAPORU**

#### **KAPSAM**

Öğretim üyesinin öğrenciyi değerlendirme modülü

#### **SEVİYE**

Admin

#### **AKTÖR**

Sistem yönetici, Öğretim Üyesi

#### **ÖN ŞARTLAR**

## BAŞARILI SENARYO SONUCU

- Öğretim üyesinin şubelerdeki öğrencilerini ilgili derslerinde, performans notu ile değerlendirmelerini yapabilmesi

## TETIKLEYICI

## BAŞARILI SENARYO ADIMLARI

Kullanıcı sisteme giriş yapar.

1) Admin ise;

(a) Öğrenci değerlendirme ile ilgili bağlantı aracılığı ile sayfaya ulaşır.

(b) Kullanıcı yapmak istediği işlemi belirler.

1. Öğrenci performans notu girişi

i. Öğretim üyesinin verdiği dersler listelenir

ii. Ders listesinden dersi seçer.

iii. Dersin şubeleri listelenir.

iv. Şubeyi seçer.

v. Seçilen şube içerisindeki öğrenciler listelenir.

1) Vize notu

2) Final notu

3) Ödev notları

4) Derse katılım

5) Sohbet saatleri

6) Sanal sınıf saatleri, bilgileri listelenir.

vi. Performans notunu girer.

vii. Değerlendirmeyi kaydet işlemini tetikler.

viii. Öğrencilerin performans notları kaydedilir.

## EKLENTİLER

## ÖZGEÇMİŞ

Adı Soyadı	Emin İBİLİ
Doğum Yeri	Susuz/Afyonkarahisar
Doğum Tarihi	24.03.1982
Medeni Hali	Evli
Yabancı Dili	İngilizce

### Eğitim Durumu (Kurum ve Yılı)

Lise	Afyonkarahisar Gazi Anadolu Teknik Lisesi, 2001
Lisans Bölümü	Mersin Üniversitesi Tarsus TEF Elektronik Bilgisayar Eğitimi Bilgisayar Öğretmenliği, 2006
Yüksek Lisans	Afyon Kocatepe Üniversitesi Bilgisayar Ana Bilim Dalı, 2009

### Yayımları (SCI ve diğer)

1. **Emin İBİLİ**, Fatih BAYRAM, Ümit AKBAŞ, Zihni ORHAN, Mahmut KANTAR, Fidan HAKKARİ ve Mevlüt DOĞAN, 2008, “Scorm Uyumlu Modüler Eğitim Yönetim Sistemlerinin Tasarlanması”, Future-Learning 2. Uluslararası Gelecek İçin Öğrenme Alanında Yenilikler Konferansı , İstanbul.
2. Fidan HAKKARİ, **Emin İBİLİ**, Mahmut KANTAR, Yasemin BOY, Fatih BAYRAM ve Mevlüt DOĞAN, 2008, “Uzaktan Eğitimde Ders Materyallerinin Hazırlanmasında Ders İçeriklerinin Tasarımı ve Senaryolaştırılması”, Future-Learning 2. Uluslararası Gelecek İçin Öğrenme Alanında Yenilikler Konferansı , İstanbul.
3. Mahmut KANTAR, **Emin İBİLİ**, Yasemin BOY, Fatih BAYRAM, Fidan HAKKARİ ve Mevlüt DOĞAN, 2008, “Uzaktan Eğitim Sistemlerinde Yazılım Ve İçerik Oluşturma”, Future-Learning 2. Uluslararası Gelecek İçin Öğrenme Alanında Yenilikler Konferansı , İstanbul.



4. Fatih BAYRAM, **Emin İBİLİ**, Ümit AKBAŞ, Zihni ORHAN, Mahmut KANTAR, Fidan HAKKARİ, Mevlüt DOĞAN, 2008, Takashi LINZBICHLER ve Franz NIEDERL, “Farklı LMS Mimarilerinde E-Eğitim İçeriklerini Paylaşmak İçin Paketleme Standartlarının Kullanımı”, Future-Learning 2. Uluslararası Gelecek İçin Öğrenme Alanında Yenilikler Konferansı, İstanbul.

6. **Emin İBİLİ**, Fatih BAYRAM, Fidan HAKKARİ, Mahmut KANTAR ve Mevlüt DOĞAN, 2009, “Scorm Uyumlu Eğitim Yönetim Sisteminin Tasarlanması Ve Üniversite Bazında Uygulanması”, Akademik Bilişim 2009 Konferansı, Harran Üniv., Şanlıurfa.

7. Fidan HAKKARİ, **Emin İBİLİ**, Mahmut KANTAR, Yasemin BOY, Fatih BAYRAM ve Mevlüt DOĞAN, 2009, “Uzaktan Öğretimde Ders Materyallerinin Scorm Uyumlu Hazırlanması Ve Uygulanması”, Akademik Bilişim 2009 Konferansı, Harran Üniv., Şanlıurfa.

8. Fatih BAYRAM, **Emin İBİLİ**, Fidan HAKKARİ, Mahmut KANTAR ve Mevlüt DOĞAN, 2009, “E-Üniversite: Scorm Uyumlu Modüler Öğrenim Yönetim Sistemlerinin Yükseköğretimde Kullanımı”, Akademik Bilişim 2009 Konferansı, Harran Üniv., Şanlıurfa.

### **Kazandığı Ödüller**

**MEB** : III.Web Tabanlı Eğitim İçeriği Geliştirme Yarışması Türkiye 3.lüğü

### **Yurtdışı Tecrübesi**

1. FH-JOANNEUM UNIVERSTY Degree of Internet Technic -Erasmus Student  
GRAZ/ AVUSTURYA 30.08.2007-11.02.2008
2. OptimalSoft Campany- Erasmus Training  
VİYANA/AVUSTURYA 16.02.2008-30.09.2008