



**İSTANBUL ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**YÜKSEK LİSANS TEZİ**

**.NET TEKNOLOJİSİ KULLANILARAK  
İNTERNET TABANLI BİR DERS KAYIT SİSTEMİNİN  
GELİŞTİRİLMESİ**

**Burak ŞİŞMAN  
Enformatik Bölümü  
Enformatik Anabilim Dalı**

**Danışman  
Doç. Dr. Hülya ÇALIŞKAN**

**Aralık, 2008**

**İSTANBUL**

Bu çalışma 19/01/2009 tarihinde aşağıdaki jüri tarafından Enformatik Anabilim Dalı Enformatik programında Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Jürisi



Doç. Dr. Hülya ÇALIŞKAN (Danışman)  
İstanbul Üniversitesi  
Enformatik Bölümü



Prof. Dr. Selahattin DİLİDÜZGÜN  
İstanbul Üniversitesi  
Hasan Ali Yücel Eğitim Fakültesi



Yrd. Doç. Dr. Hulusi GÜLSEÇEN  
İstanbul Üniversitesi  
Enformatik Bölümü



Yrd. Doç. Dr. Ahmet AĞIR  
İstanbul Üniversitesi  
Hasan Ali Yücel Eğitim Fakültesi



Doç. Dr. Zuhale TANRIKULU  
Boğaziçi Üniversitesi  
Yönetim Bilişim Sistemleri Bölümü

Bu alıřma 2008 yılında İstanbul Üniversitesi Bilimsel Arařtırma Projeleri Yürütücü Sekreterliđinin 2415 numaralı projesi olarak desteklenmiřtir.

## ÖNSÖZ

Yüksek lisans öğrenimim ve tez hazırlama sürecimde her zaman yanımda olan danışman hocam sayın Doç. Dr. Hülya Çalışkan'a

Ayrıca;

Yüksek lisans eğitimim ve tez hazırlama sürecimde desteklerini esirgemeyen sevgili hocalarım Yrd. Doç. Dr. Zerrin A. Reis'e, Yrd. Doç. Dr. Ahmet Ağır'a ve Yrd. Doç. Dr. Serap Emir'e,

Tez kapsamında hazırladığım uygulamanın, geliştirilmesi sırasında yardımcı olan Hasan Ali Yücel Eğitim Fakültesi Dekanı, Sayın Prof. Dr. Selahattin Dilidüzgün'e, Öğrenci İşleri Bürosu Şefi, Kısmet Akın'a, büro çalışanları; Aygün Kalemci, Necdet Haklı, Emine Mehtap Koç ve Kenan Ağaldag'a,

Tez hazırlama sürecimde yardımlarını esirgemeyen Araş. Gör. İrfan Şimşek'e ve Araş. Gör. Yavuz Yaman'a,

Teşekkürü bir borç bilirim.

Tüm hayatım boyunca bana destek olan, kendi kararlarımın her zaman en iyisi olacağını düşünen, demokratik ve hür bir aile ortamı sağlayan ve bulunduğum yerde olmamda en büyük role sahip olan sevgili babam İrfan Şişman'a ve canım annem Firdevs Şişman'a, eğitim hayatım süresince maddi-manevi desteğini hiç esirgemeyen sevgili kardeşim Buğra Şişman'a da

Şükranlarımı sunarım.

**Aralık, 2008**

**Burak ŞİŞMAN**

# İÇİNDEKİLER

ÖNSÖZ .....	i
İÇİNDEKİLER .....	ii
ŞEKİL LİSTESİ .....	iv
TABLO LİSTESİ .....	vi
ÖZET .....	vii
SUMMARY .....	viii
1. GİRİŞ .....	1
2. DERS KAYIT SİSTEMİ .....	3
2.1. KLASİK DERS KAYIT SİSTEMİ .....	3
2.1.1. Klasik Ders Kayıt Sistemi Aşamalarının İncelenmesi .....	4
2.1.2. Klasik Ders Kayıt Sisteminde Karşılaşılan Problemler .....	7
2.2. İNTERNET TABANLI DERS KAYIT SİSTEMİ .....	8
2.2.1. İnternet Tabanlı Ders Kayıt Sistemi Kayıt Aşamaları .....	8
2.2.2. İnternet Tabanlı Ders Kayıt Sistemlerinde Karşılaşılan Problemler .....	10
2.2.3. Türkiye’deki İnternet Tabanlı Ders Kayıt Sistemi Uygulamaları .....	11
3. .NET TEKNOLOJİSİ KULLANILARAK İNTERNET TABANLI ÖĞRENCİ KAYIT SİSTEMİ GELİŞTİRİLMESİ .....	22
3.1. HASAN ALİ YÜCEL EĞİTİM FAKÜLTESİ’NDEKİ MEVCUT DERS KAYIT SİSTEMİ .....	22
3.2. GELİŞTİRİLEN İNTERNET TABANLI DERS KAYIT SİSTEMİ .....	27
3.2.1. Geliştirme Ortamının Seçilmesi .....	27
3.2.2. İTOKS Uygulaması .....	44
3.2.2.1 Sistem Yöneticisi (Rektörlük Öğrenci İşleri Daire Başkanlığı) Girişi .....	45
3.2.2.2 Fakülte Girişi .....	47
3.2.2.3 Öğrenci İşleri Bürosu Şefi Girişi .....	48
3.2.2.4 Öğrenci İşleri Bürosu Memuru Girişi .....	49
3.2.2.5 Personel İşleri Şefi Girişi .....	50
3.2.2.6 Personel İşleri Memuru Girişi .....	51
3.2.2.7 Öğretim Görevlisi Girişi .....	52
3.2.2.8 Öğrenci Girişi .....	54

<b>3.2.3. İTOKS'un Uygulanması</b> .....	59
3.2.3.1. <i>Evren ve Örneklem</i> .....	59
3.2.3.2. <i>Veriler ve Toplanması</i> .....	59
3.2.3.3. <i>Uygulama</i> .....	59
3.2.3.4. <i>Verilerin Analizi</i> .....	60
3.2.3.5. <i>Bulgular</i> .....	60
<b>4. SONUÇ VE TARTIŞMA</b> .....	63
<b>KAYNAKLAR</b> .....	66
<b>EKLER</b> .....	69
<b>ÖZGEÇMİŞ</b> .....	80

## ŞEKİL LİSTESİ

Şekil 2.1	: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 1. adım	4
Şekil 2.2	: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 2. adım	4
Şekil 2.3	: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 3. ve 4. adım	5
Şekil 2.4	: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 5. ve 6. adım	5
Şekil 2.5	: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 7. adım	5
Şekil 2.6	: Klasik ders kayıt sistemi aşamalarının şeması tüm adımlar	6
Şekil 2.7	: İnternet tabanlı ders kayıt sistemi aşamalarının incelenmesi şeması 1. adım	8
Şekil 2.8	: İnternet tabanlı ders kayıt sistemi aşamalarının incelenmesi şeması 2. adım	9
Şekil 2.9	: İnternet tabanlı ders kayıt sistemi aşamalarının incelenmesi şeması 3. adım	9
Şekil 2.10	: İnternet tabanlı ders kayıt sistemi aşamalarının incelenmesi şeması tüm adımlar	10
Şekil 3.1	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 1. adım	22
Şekil 3.2	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 2. adım	23
Şekil 3.3	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 3. adım	23
Şekil 3.4	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 4. adım	23
Şekil 3.5	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 5. adım	24
Şekil 3.6	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 6. ve 7. adım	24
Şekil 3.7	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 8. adım	25
Şekil 3.8	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 9. adım	25
Şekil 3.9	: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması tüm adımlar	26
Şekil 3.10	: Just In Time Compiler	33
Şekil 3.11	: CLR'nin bulunduğu konum	34
Şekil 3.12	: Örnek bir XML dosyası	36
Şekil 3.13	: Basit bir DTD örneği	37
Şekil 3.14	: XML dosyası element yapısı	38
Şekil 3.15	: Çok Katmanlı Mimari	40
Şekil 3.16	: Tablo kayıtlarının nesnelere yüklenmesi	41
Şekil 3.17	: İTOKS'daki Kullanıcı Rollerini	45
Şekil 3.18	: İTOKS - Sistem Yöneticisi (Rektörlük Öğrenci İşleri Daire Başkanlığı) Ekranı	46
Şekil 3.19	: İTOKS - Fakülte Ekranı	47
Şekil 3.20	: İTOKS - Öğrenci İşleri Bürosu Şefi Ekranı	48

Şekil 3.21	: İTOKS - Öğrenci İşleri Bürosu Memuru Ekranı .....	49
Şekil 3.22	: İTOKS - Personel İşleri Şefi Ekranı .....	51
Şekil 3.23	: İTOKS - Personel İşleri Memuru Ekranı .....	52
Şekil 3.24	: İTOKS - Öğretim Görevlisi Ekranı .....	53
Şekil 3.25	: İTOKS - Öğretim Görevlisi'nin Dersine Ait Sınıf Listesi Ekranı .....	53
Şekil 3.26	: İTOKS - Öğrenci Ekranı .....	54
Şekil 3.27	: İTOKS - Öğrenci Ders Kayıt Ekranı .....	55
Şekil 3.28	: İTOKS - Öğrenci Ders Kayıt Formu .....	56
Şekil 3.29	: İTOKS - Öğrenci Transkript Ekranı .....	56
Şekil 3.30	: İTOKS ile ders kayıt aşamalarının incelenmesi şeması 1. adım .....	57
Şekil 3.31	: İTOKS ile ders kayıt aşamalarının incelenmesi şeması 2. adım .....	57
Şekil 3.32	: İTOKS ile ders kayıt aşamalarının incelenmesi şeması 3. adım .....	58
Şekil 3.33	: İTOKS ile ders kayıt aşamalarının incelenmesi şeması tüm adımlar .....	58

## TABLO LİSTESİ

Tablo 2.1 : Türkiye’deki üniversitelerde internet tabanlı ders kaydı uygulamaları .....	12
Tablo 3.1 : XML de kullanılmayacak karakterlerin yerine kullanılacak ifadeler . .....	38
Tablo 3.2 : .NET için O/RM araçları .....	42
Tablo 3.3 : ”İnternet Tabanlı Ders Kayıt Sistemi, belirtilen kayıt tarihleri arasında istediğim gün ve saatte kayıt yapabilmemi sağlar.” .....	60
Tablo 3.4 : ”İnternet Tabanlı Ders Kayıt Sistemi, danışmanıma ulaşamama gibi olumsuz durumları ortadan kaldırır.” .....	60
Tablo 3.5 : ”İnternet Tabanlı Ders Kayıt Sistemi’ni kullanmanın kolay olduğu kanısındayım.” .....	61
Tablo 3.6 : ”İnternet Tabanlı ders kayıt klasik ders kaydına göre daha az zamanımı aldı.” .....	61
Tablo 3.7 : ”İnternet Tabanlı Ders Kayıt Sistemi sayesinde transkriptimi istediğim zamanda alabiliyorum.” .....	61
Tablo 3.8 : ”İnternet Tabanlı Ders Kayıt Sistemi sayesinde yardımsız kayıt yapabiliyorum.” .....	62
Tablo 3.9 : ”Kayıt yaparken hata yaptığımda anında düzeltebiliyorum.” .....	62

## ÖZET

### **.NET TEKNOLOJİSİ KULLANILARAK İNTERNET TABANLI BİR DERS KAYIT SİSTEMİNİN GELİŞTİRİLMESİ**

Bu tezin amacı üniversitemizde klasik yolla yapılan öğrenci ders kaydı işlemlerine alternatif olarak modern bir yöntemi tanıtmaktır. Bu yöntem, .NET Framework teknolojisi kullanılarak, ASP.NET C# programlama dili ve .NET O/RM araçlarından biri olan Nhibernate ile geliştirilmiştir. İnternet üzerinden ulaşılabilen bir yapıya sahiptir. Amacımız geliştirdiğimiz bu yazılımla üniversitemize katkıda bulunmaktadır.

Halen üniversitemizde yapılmakta olan ders kayıt işlemleri oldukça zorluklar ve karmaşa içerisinde ve hata oranı yüksek olarak gerçekleşmektedir. Öğrenciler ve kayıt yapan elemanlar klasik yöntemden kaynaklanan hatalardan dolayı oldukça sıkıntılı durumlar yaşamaktadırlar. Bu karmaşa ve zorlukları önlemek için teknolojiden faydalanarak daha iyi bir yöntemle kayıt işlemlerinin yapılması zorunlu hale gelmiştir. Bu yöntem sayesinde, zaman ve mekândan bağımsız olarak öğrenciyi ve kayıt yapan elemanları memnun edecek bir duruma gelinecektir.

Oluşturulan sistem üç aşamada gerçekleştirilmiştir;

Birinci aşamada, ilk olarak Hasan Ali Yücel Eğitim Fakültesi'nde bir çalışma yapılmıştır. Mevcut öğrenci kayıt sistemi incelenerek karşılaşılan sorunlar ve eksiklikler belirlenmiş, bunlara nasıl çözümler üretileceği kararlaştırılmıştır.

İkinci aşamada, üretilen çözümler doğrultusunda .Net Framework teknolojisinden yararlanarak, C# programlama dili ve Nhibernate kullanılarak söz konusu online kayıt sistemi geliştirilmiştir.

Üçüncü aşamada, sistem test edilmiş ve eksiklikler giderilmeye çalışılmıştır.

## **SUMMARY**

### **THE DEVELOPMENT OF AN INTERNET BASED COURSE REGISTRATION SYSTEM USING .NET TECHNOLOGY**

The aim of this thesis is to develop a modern and technological method for the student course registration process. At present, there is a classical course registration process at the İstanbul University and the students have to fill in the forms and take these forms to the faculty registration office.

Our method will have a structure which has been developed by NET Framework technology, ASP .NET C# computer language and Nhibernate which is one of the .NET O/RM tools. It will be accessible on the Internet. Thus the software will help our university in this subject.

At present, the student course registration process of the university is difficult, complex and has a high proportion of mistakes. Students and the registration staff members make a lot of mistakes with this classical method.

A smooth and technological registration method is required to prevent this confusion and difficulties. This course registration method will satisfy students and the staff since it will provide a lot of flexibility with its independent time and place feature.

The system has been completed in three stages;

In the first stage, the studies have been started at Hasan Ali Yücel Faculty of Education. We will examine the recent student course registration method at the faculty and notice the current problems and deficient points. Then we have tried to find solutions to these problems.

In the second stage, considering the solutions of the problems the online course registration code had been programmed by using .Net Framework technology, C# software language and Nhibernate.

In the third stage, the code has been tested and the deficiencies had tried to satisfied.

## 1. GİRİŞ

Sosyal, ekonomik, demografik ve teknolojik alanlardaki gelişmeler, yüksek öğretimdeki eğitimcileri, eğitimlerini veriş biçimlerini tekrar gözden geçirmeleri konusunda etkilemiştir. Schank'e [1] göre teknoloji, eğitsel yapıyı, bildiğimiz anlamdaki duvarları yıkarak ve bilgiye ulaşmaya, problemleri çözmeye ve birlikte çalışmayı sağlamaya yönelik yeni yollar geliştirmeye olanak tanıyarak değiştirmektedir. Teknolojinin eğitim alanında yaratabileceği devrim niteliğindeki değişiklikleri farketmenin bir sonucu olarak da, dünya çapındaki birçok yüksek öğretim kurumunda olduğu gibi Türkiye'de de birçok devlet üniversitesi ve özel üniversite, internet tabanlı teknolojileri kullanarak eğitim niteliklerini arttırmak üzere çeşitli çalışmalar yapmaya başlamışlardır.

Bu çalışmalar kağıt ortamında yapılan işlemlerin dijital ortama taşınması şeklinde başlamıştır. Öğrencilerin kampüse gelip öğrenci işleri ile ilgilenen birimlere uğrayarak işlemlerini halletmelerine gerek bırakmayacak şekilde dijital kayıt ve takip sistemleri kurmak, ders seçmek, ders kaydı yapmak, müfredat ve programlara erişmek gibi çevrim içi öğrenci işleri etkinliklerinin internet üzerinden yapılması, üniversite kütüphanesinde bulunan kaynaklara erişim ve listeleme için e-kütüphaneler oluşturulması ve eğitimcilerle ders programlarını ve materyallerini koyabilmeleri için basit HTML sayfaları oluşturabilecekleri alanlar sağlanması, dijital üniversiteler oluşturmaya yönelik hareketlerin temelinde yer almaktadır [2].

Bu çalışmada; ülkemizdeki bazı üniversitelerde kâğıt üzerinde klasik yollarla yapılan ders kayıtlarının problemleri ele alınmış ve bu problemleri, teknolojiyi kullanarak büyük ölçüde ortadan kaldırmayı amaçlayan internet tabanlı bir ders kayıt sistemi geliştirilmiştir. İnternet Tabanlı Öğrenci Kayıt Sistemi (İTOKS), öncelikle İstanbul Üniversitesi Hasan Ali Yücel Eğitim Fakültesi'nin öğrenci ders kayıt işlemlerini yapacak şekilde geliştirilmiştir ancak, istenildiğinde tüm üniversite bünyesinde kullanılabilir esneklikte dir.

Bu tezin daha sonraki bölümlerinde önce, klasik ders kayıt sisteminin tanımı ve yapısından bahsedilmiştir. Daha sonra ise Türkiye'deki üniversitelerin kullandıkları internet tabanlı öğrenci kayıt sistemleri ile ilgili bilgiler verilmiştir. Bir sonraki bölümde de internet tabanlı ders kaydı için önerilen yapının modelinden ve uygulanmasından bahsedilmiştir.

## 2. DERS KAYIT SİSTEMİ

Üniversitelerde öğrenci kayıt işlemleri, giriş kayıtları dışında, kayıt yenileme ve ders kaydı aşamalarından oluşmaktadır. Her üniversite öğrencisi eğitim hayatı boyunca güz, bahar ve varsa yaz olmak üzere yılda üç dönem kayıt yenilemekte ve o dönem alacağı dersleri seçmektedir. Klasik ders kayıt sistemi, öğrencilerin fakültelerine gelerek, kağıt üstünde ders seçip, kayıt yenilemeleri şeklinde olmaktadır. Kayıt yenileme işlemini klasik sistem ile yapan üniversitelerin ilgili birimleri ve öğrencileri çeşitli nedenlerden dolayı bu süreçle ilgili birçok sorunlar yaşamaktadırlar. Bu sorunlardan bazıları şunlardır:

- Takip edilmesi gereken öğrenci sayısının fazla oluşu.
- Öğrencilerin ders kayıt formlarında yaptıkları hatalar.
- Her öğrencinin doldurduğu kayıt evraklarının kontrolünün zorluğu ve zaman alması.
- Öğrencilerin ders kaydı için fakültelerine şahsen gelmek zorunda olmaları.
- Her öğrenci için transkript basılması zorunluluğunun olması.

Klasik ders kaydında yaşanan bu ve benzeri olumsuzluklar, sayısal ortamın kazanımları ile çözümlenebilmektedir. Bu nedenle, üniversitelerin büyük bir çoğunluğu kayıt sistemlerini internet ortamına taşımış bulunmaktadır. Bu da, internet aracılığı ile üniversitenin sağladığı yazılım üzerinden, İnternet Tabanlı Ders Kayıt Sistemi kavramını ortaya çıkarmıştır. Böylelikle ders kayıt sistemleri, klasik ve internet tabanlı olmak üzere incelenebilir. Türkiye'deki üniversitelerde uygulanan ders kayıt sistemleri hakkındaki bilgiler daha sonraki bölümlerde verilecektir.

### 2.1. KLASİK DERS KAYIT SİSTEMİ

Üniversitelerin Senatoları tarafından, ilan edilen akademik takvime göre belirli ders kayıt tarihlerinde öğrenciler üniversitelerine bizzat giderek, işlemlerini başlatır. Bu işlemler, danışmanlarından temin edecekleri formları doldurmak, almak istedikleri dersleri seçmek ve bu formları danışmanlarına ve öğrenci işleri birimine onaylatmak

şeklindedir. Daha sonra ilgili fakültenin öğrenci işleri birimi, bu bilgileri otomasyona kaydeder. Tüm bu işlem sürecini, klasik ders kayıt sistemi olarak tanımlayabiliriz.

### 2.1.1. Klasik Ders Kayıt Sistemi Aşamalarının İncelenmesi

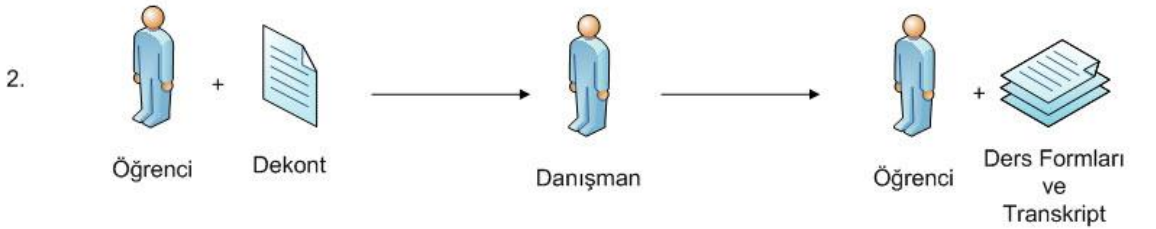
Klasik ders kayıt sistemi uygulanan üniversitelerde öğrencilerin dönem başlarında ders seçim işlemleri için genellikle izledikleri yol şu şekildedir;

1. Öğrenci dönem kaydı yaptırmak için yatırması gerekli olan harç miktarını ilgili banka şubesine yatırır ve dekontunu alır.



Şekil 2.1: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 1. adım

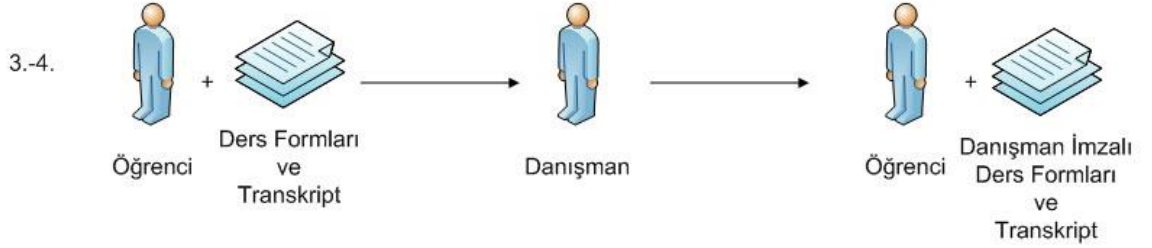
2. Öğrenci bankadan almış olduğu dekontla birlikte danışman öğretim elemanına gider ve önceki dönemlere ait transkriptini ve ders kayıt formunu alır.



Şekil 2.2: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 2. adım

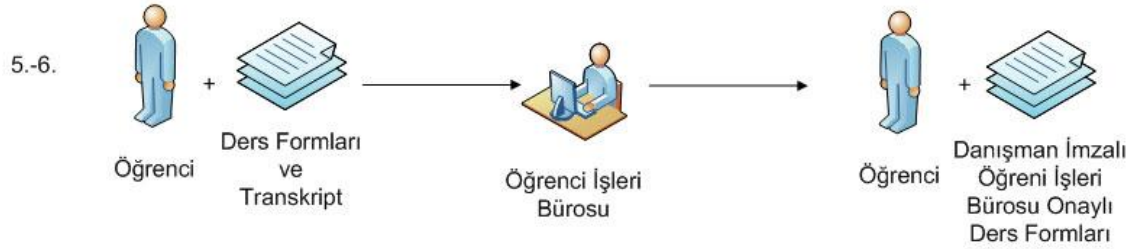
3. Öğrenci 3 nüsha şeklinde bu formu doldurur ve tekrar danışmanına giderek, kontrol etmesi için transkripti ile birlikte danışmanına verir.

4. Danışman öğrencinin seçtiği dersleri ve formları kontrol eder, bir problem yoksa üç nüshayı da imzalar.



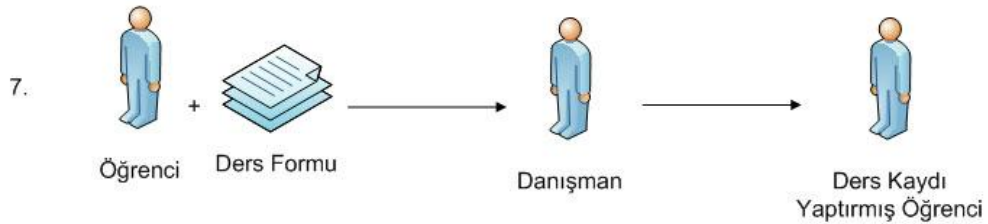
Şekil 2.3: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 3. ve 4. adım

5. Öğrenci danışmanın imzaladığı formları tekrar alır ve gerekli olan diğer belgelerle birlikte öğrenci işleri bürosuna götürür.
6. Öğrenci işleri bürosu, öğrencinin ders kayıt formlarını inceler ve bir problem yoksa mühürlü ve bir adet form kendisinde kalacak şekilde diğer formları öğrenciye teslim eder.



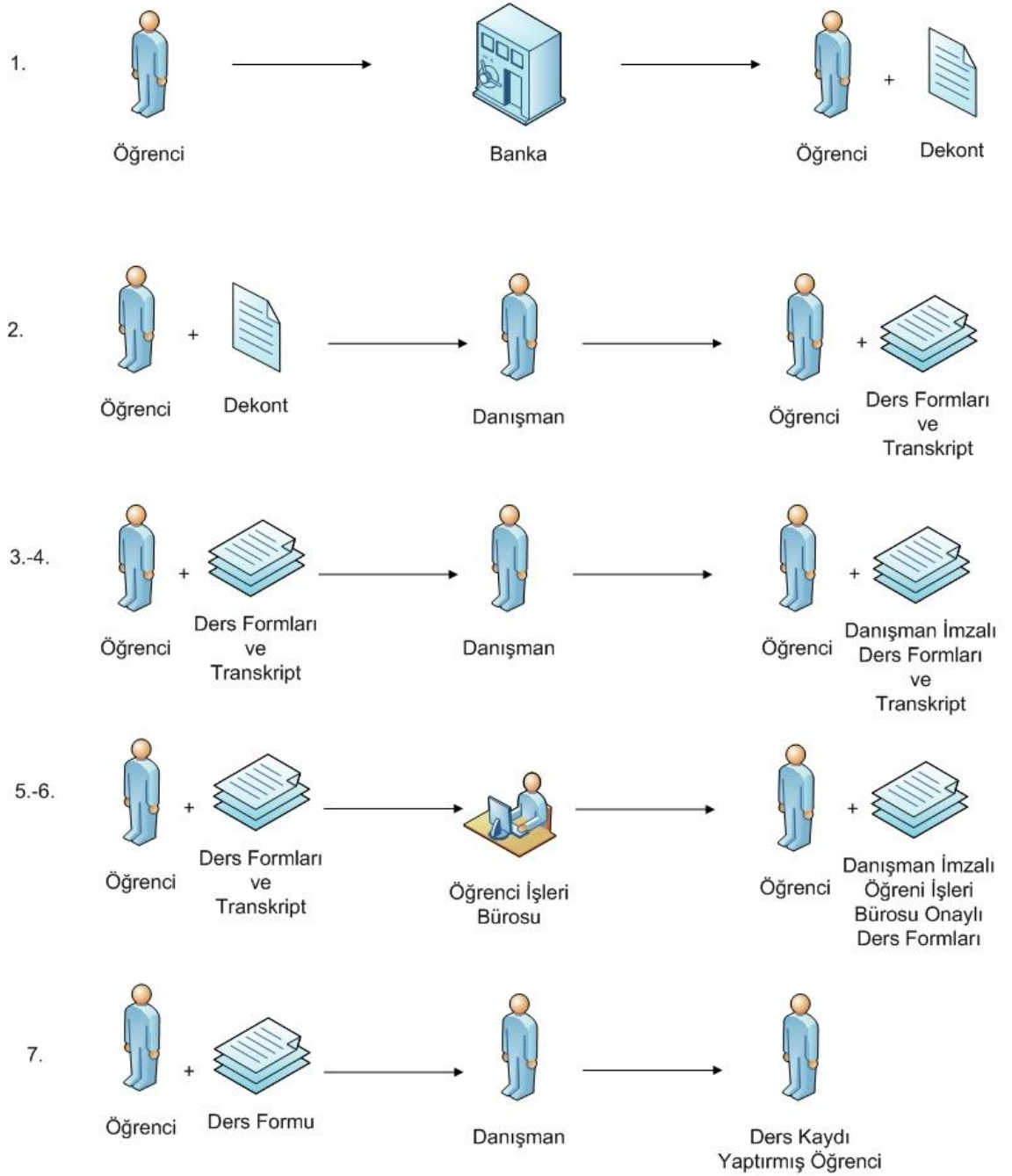
Şekil 2.4: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 5. ve 6. adım

7. Öğrenci elindeki iki formdan birini danışman öğretim elemanına iletir ve kayıt aşaması tamamlanmış olur.



Şekil 2.5: Klasik ders kayıt sistemi aşamalarının incelenmesi şeması 7. adım

### KLASİK DERS KAYIT AŞAMALARI



Şekil 2.6: Klasik ders kayıt sistemi aşamalarının şeması tüm adımlar

### **2.1.2. Klasik Ders Kayıt Sisteminde Karşılaşılan Problemler**

Bir önceki bölümde, klasik kayıt sistemi aşamalarının incelenmesinde görüldüğü gibi, bu kayıt süreci, hem öğrenciler hem de binlerce öğrencisi olan fakültelerin öğrenci işleri ve bu öğrencilere danışmanlık yapan öğretim elemanları için oldukça yorucu ve zaman alıcıdır.

Bu süreçte öğrenciler tarafından yanlış doldurulan formları ve ders seçiminde her öğrencinin geçmiş derslerinin incelenmesi amacıyla basılan, binlerce transkript belgesi düşünüldüğünde, çok sayıda kağıt harcandığı ve bu harcamanın devlet ve toplum bütçesine büyük zararlar verdiği görülmektedir.

Ayrıca, öğrenciler farklı bir bölümden ders seçmek istediklerinde, danışman öğretim elemanının bu bölümlerdeki dersler ve kontenjanlar hakkında bilgisinin yetersiz olmasından dolayı, öğrencileri ilgili bölümlere yönlendirmek zorunda kalmaktadır. Bu durum hem öğrenciler hem de danışmanlar için vakit kaybına ve ders kayıt sürecinin yavaşlamasına neden olmaktadır.

Görüleceği üzere böyle yoğun bir ortamda gözden kaçan hatalar (yanlış ders seçimi, öğrencinin alamayacağı dersi seçmesi, kredi aşımı v.b.) oldukça fazla olacaktır. Tüm bu hatalar sonradan farkedildiğinde, düzeltilebilmesi için fakülte yönetim kurulu kararları gerekmektedir. Bu da yine ek bir zaman kaybı demektir.

Öğrenci ve ailelerinin şikâyet ettikleri bir başka durum da; birçok üniversitede dönem dersleri genellikle ders kayıt işlemlerinden bir hafta sonra başlamaktadır. Ders kaydı yaptırmak için şehir dışından gelen öğrencilerin çoğu, işlemlerden sonra tekrar evlerine dönmeyi tercih etmektedir. Bu da ek bir yol ücreti ve masraf demektir.

Neticede klasik kayıt sisteminde taraflar için oldukça sıkıntılar yaşandığı görülmektedir. Şimdi de alternatif olarak internet tabanlı ders kayıt sistemini inceleyelim.

## 2.2. İNTERNET TABANLI DERS KAYIT SİSTEMİ

Bu kayıt sisteminde, yine, üniversitelerin Senatoları tarafından ilan edilen akademik takvim sürecinde, öğrenciler internet erişimi sağlayabilecekleri herhangi bir yerden, kendi üniversitelerinin sağladığı ders kayıt uygulamasını kullanarak, almak istedikleri dersleri seçme işlemini gerçekleştirirler. Bunu da internet tabanlı ders kayıt sistemi olarak tanımlayabiliriz.

Bu sistemde hazırlanan uygulamanın yönlendirmeleri, öğrenciye danışmanlık görevi yapmaktadır. Böylece danışman öğretim elemanlarının işi kolaylaşmış olmaktadır.

Bu sistem getirdiği avantajlar bakımından her ne kadar kusursuz gözükse de, bazı aksaklıkları vardır. Bunlar; üniversitelerimizin bilişim alt yapısındaki eksiklikler, farklı uygulama şekillerinden doğan problemler ve öğrencilerin bilgisayar kullanımında yetersiz olmaları gibi etkenlerden kaynaklanmaktadır. Bu aksaklıkları Sistemden Kaynaklanan ve Öğrenciden Kaynaklanan olmak üzere iki başlık halinde daha sonraki bölümlerde inceleyeceğiz. Şimdi Klasik Ders Kayıt Sistemi Aşamalarını incelediğimiz gibi İnternet Tabanlı Ders Kayıt Sisteminin aşamalarını inceleyelim;

### 2.2.1. İnternet Tabanlı Ders Kayıt Sistemi Kayıt Aşamaları

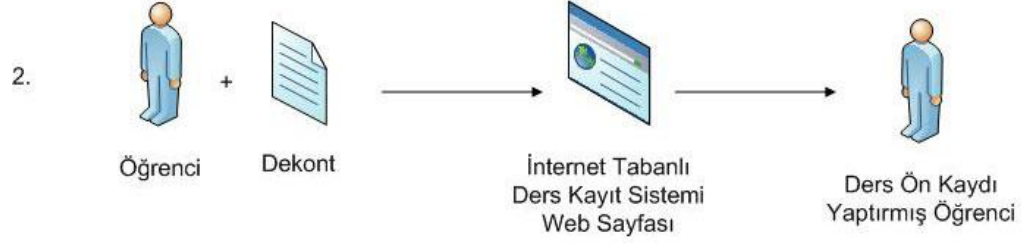
İnternet tabanlı ders kayıt sistemi kullanıldığında izlenen kayıt aşamaları genellikle şu şekildedir;

1. Öğrenci dönem kaydı yaptırabilmek için yatırması gerekli olan harç miktarını ilgili banka şubesine yatırır ve dekontunu alır.



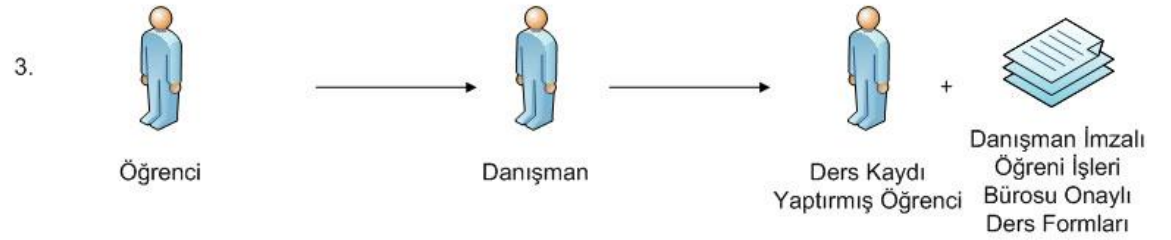
Şekil 2.7: İnternet tabanlı ders kayıt sistemi aşamalarının incelenmesi şeması 1. adım

2. Öğrenci internet bağlantısı olan herhangi bir yerdeki bir bilgisayar aracılığı ile üniversitesinin sağladığı İnternet Tabanlı Ders Kayıt Sistemi Uygulaması'na önceden almış olduğu giriş bilgileri ile giriş yapar, durumunu görüntüler, sistem aracılığıyla açılmış olan dersleri görüntüler ve alabileceği dersleri seçerek ön kaydını tamamlar.



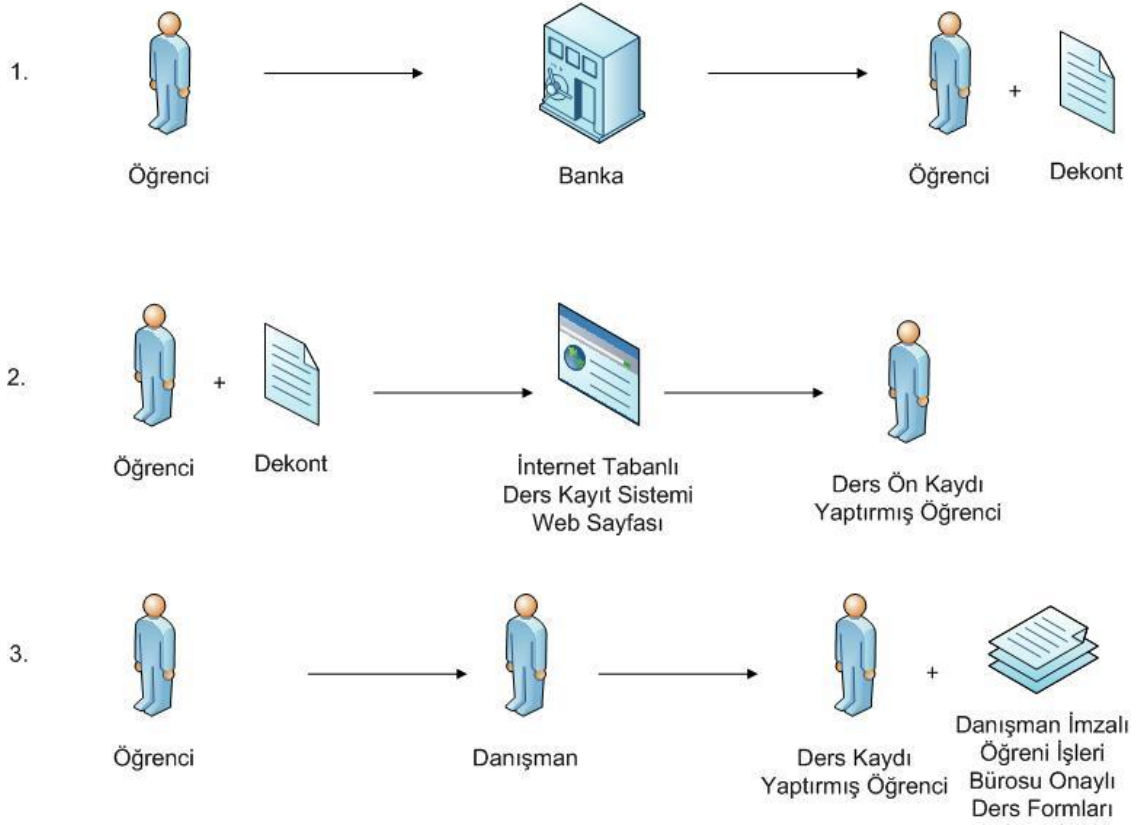
Şekil 2.8: İnternet tabanlı ders kayıt sistemi aşamalarının incelenmesi şeması 2. Adım

3. Öğrenci danışmanına gider, seçtiği derslerin dökümünü içeren imzalı ve onaylı ders kayıt formunu alır ve ders kaydını tamamlamış olur.



Şekil 2.9: İnternet tabanlı ders kayıt sistemi aşamalarının incelenmesi şeması 3. adım

**İNTERNET TABANLI DERS KAYIT SİSTEMİ  
KAYIT AŞAMALARI**



Şekil 2.10: İnternet tabanlı ders kayıt sistemi aşamalarının incelenmesi şeması tüm adımlar

### 2.2.2. İnternet Tabanlı Ders Kayıt Sistemlerinde Karşılaşılan Problemler

Bu kısımda, karşılaşılan problemleri: Öğrenciden Kaynaklanan ve Sistemden Kaynaklanan olarak iki şekilde inceleyebiliriz.

Öğrenciden Kaynaklanan Problemler:

- Bazı öğrencilerin, bilgisayar ve internet kullanım bilgilerinin yetersiz olmasından dolayı sistemin kullanımının zor gelmesi,
- Şifrelerini diğer arkadaşlarıyla paylaşmaları sonucu doğan problemler,
- İnternete güvenemediklerinden ötürü tedirgin olmaları

gibi problemlerdir.

Sistemden Kaynaklanan Problemler:

- Sistemi barındıran sunucu alt yapısının yetersizliği ve yoğunluktan dolayı sisteme ulaşamaması,
- Kullanıcı ara yüzünün iyi tasarlanmaması,
- Öğrencilerin şifrelerini nasıl temin edebilecekleri konusunda bilgilendirilmemeleri,
- Şifre unutulması durumunda sistemin şifre vermemesi ve öğrenciyi öğrenci işlerine yönlendirmesi,
- Şifre dağıtımının düzgün yapılmamasından doğan, öğrencilerin diğer öğrencilerin şifrelerini öğrenmesinden ve kolayca tahmin edebilmesinden kaynaklanan problemler,
- Çakışan dersler için sistemin öğrencilere çözüm üretememesi,
- Sistemi kullanmak isteyen kullanıcılara program yükletmek istenmesi aksi halde sisteme ulaşamaması,
- Banka harç sorgusu problemleri,

gibi problemlerdir.

### 2.2.3. Türkiye’deki İnternet Tabanlı Ders Kayıt Sistemi Uygulamaları

Bu kısımda, Yüksek Öğretim Kurulu’nun resmi web sitesinde [3] “Üniversiteler” başlığı altında bulunan 68 devlet üniversitesindeki öğrenci otomasyonlarına yönelik bilgiler verilecektir. Bu bilgilerde öğrenci otomasyonlarının;

- Hangi programlama dili ile geliştirildiğine,
- SSL<sup>1</sup> güvenlik protokolü kullanıp kullanmadığına,
- Öğrencilerine internetten ders kaydı hizmeti sunup sunmadığına

ilişkin yapılan inceleme sonuçlarından bahsedilecektir. Sonuçlar hem tablo hem açıklama şeklinde verilmiştir. Konu içerisinde geçecek olan programlama dillerinin isim ve kısaltmaları şöyledir;

---

<sup>1</sup> SSL: Secure Sockets Layer (SSL) şifrelenmiş bir bağlantının, bir web sunucu ve bir web görüntüleyici arasında tanımlanması için gereken standart bir güvenlik teknolojisidir [4].

- Microsoft-Active Server Pages (ASP),
- .NET dil ailesi (C# ve VB)'den Microsoft-Active Server Pages X (ASPX),
- Java Server Pages (JSP),
- Common Gateway Interface (CGI),
- ColdFusion Markup Language (CFML)
- Personel Home Page (PHP)

Tablo 2.1: Türkiye'deki üniversitelerde internet tabanlı ders kaydı uygulamaları

Üniversite	Programlama Dili	Ders Kaydı	SSL
ABANT İZZET BAYSAL ÜNİVERSİTESİ	ASPX	Evet	Hayır
ADİYAMAN ÜNİVERSİTESİ	PHP	Evet	Hayır
ADNAN MENDERES ÜNİVERSİTESİ	ASPX	Evet	Hayır
AFYON KOCATEPE ÜNİVERSİTESİ	ASPX	Evet	Hayır
AHI EVRAN ÜNİVERSİTESİ	PHP	Evet	Evet
AKDENİZ ÜNİVERSİTESİ	ASPX	Evet	Hayır
AKSARAY ÜNİVERSİTESİ	PHP	Evet	Evet
AMASYA ÜNİVERSİTESİ	ASP	Evet	Evet
ANADOLU ÜNİVERSİTESİ	ASP	Evet	Hayır
ANKARA ÜNİVERSİTESİ	PHP	Evet	Evet
ATATÜRK ÜNİVERSİTESİ	PHP	Evet	Hayır
BALIKESİR ÜNİVERSİTESİ	PHP	Evet	Hayır
BOĞAZİÇİ ÜNİVERSİTESİ	ASP	Evet	Evet
BOZOK ÜNİVERSİTESİ	ASPX	Evet	Hayır
CELAL BAYAR ÜNİVERSİTESİ	CGI	Evet	Hayır
CUMHURİYET ÜNİVERSİTESİ	JSP	Evet	Hayır
ÇANAKKALE ONSEKİZ MART ÜNİVERSİTESİ	ASP	Evet	Hayır
ÇUKUROVA ÜNİVERSİTESİ	ASP	Evet	Evet
DİCLE ÜNİVERSİTESİ	CGI	Evet	Hayır
DOKUZ EYLÜL ÜNİVERSİTESİ	PHP	Evet	Hayır
DUMLUPINAR ÜNİVERSİTESİ	JSP	Evet	Hayır
DÜZCE ÜNİVERSİTESİ	ASP	Evet	Hayır
EGE ÜNİVERSİTESİ	ASPX	Evet	Hayır
ERCİYES ÜNİVERSİTESİ	ASPX	Evet	Hayır
ERZİNCAN ÜNİVERSİTESİ	PHP	Evet	Hayır
ESKİŞEHİR OSMANGAZİ ÜNİVERSİTESİ	ASPX	Evet	Hayır
FIRAT ÜNİVERSİTESİ	ASP	Evet	Hayır
GALATASARAY ÜNİVERSİTESİ	YOK	Hayır	
GAZİ ÜNİVERSİTESİ	PHP	Evet	Evet

GAZİANTEP ÜNİVERSİTESİ	JAVA	Evet	Evet
GAZİOSMANPAŞA ÜNİVERSİTESİ	ASP	Evet	Hayır
GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ	ASPX	Evet	Hayır
GİRESUN ÜNİVERSİTESİ	ASPX	Evet	Hayır
HACETTEPE ÜNİVERSİTESİ	CGI	Evet	Hayır
HARRAN ÜNİVERSİTESİ	ASP	Evet	Hayır
HİTİT ÜNİVERSİTESİ	ASPX	Evet	Hayır
İNÖNÜ ÜNİVERSİTESİ	ASPX	Evet	Hayır
İSTANBUL ÜNİVERSİTESİ	YOK	Hayır	
İSTANBUL TEKNİK ÜNİVERSİTESİ	PHP	Evet	Hayır
İZMİR YÜKSEK TEKNOLOJİ ENSTİTÜSÜ	JSP	Evet	Hayır
KAFKAS ÜNİVERSİTESİ	ASPX	Evet	Hayır
KAHRAMANMARAŞ SÜTÇÜ İMAM ÜNİVERSİTESİ	JSP	Evet	Evet
KARADENİZ TEKNİK ÜNİVERSİTESİ	JSP	Evet	Evet
KASTAMONU ÜNİVERSİTESİ	PHP	Evet	Hayır
KIRIKKALE ÜNİVERSİTESİ	ASP	Evet	Hayır
KOCAELİ ÜNİVERSİTESİ	CFML	Evet	Hayır
MARMARA ÜNİVERSİTESİ	YOK	Hayır	
MEHMET AKİF ERSOY ÜNİVERSİTESİ	ASP	Evet	Hayır
MERSİN ÜNİVERSİTESİ	PHP	Evet	Hayır
MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ	JSP	Evet	Hayır
MUĞLA ÜNİVERSİTESİ	JSP	Evet	Hayır
MUSTAFA KEMAL ÜNİVERSİTESİ	ASP	Evet	Hayır
NAMIK KEMAL ÜNİVERSİTESİ	ASP	Evet	Hayır
NİĞDE ÜNİVERSİTESİ	ASP	Evet	Hayır
ONDOKUZ MAYIS ÜNİVERSİTESİ	ASP	Evet	Evet
ORDU ÜNİVERSİTESİ	ASPX	Evet	Hayır
ORTA DOĞU TEKNİK ÜNİVERSİTESİ	CGI	Evet	Evet
PAMUKKALE ÜNİVERSİTESİ	PHP	Evet	Hayır
SAKARYA ÜNİVERSİTESİ	PHP	Evet	Hayır
SELÇUK ÜNİVERSİTESİ	ASP	Evet	Hayır
SIİRT ÜNİVERSİTESİ	CGI	Evet	Hayır
SÜLEYMAN DEMİREL ÜNİVERSİTESİ	ASP	Evet	Hayır
TRAKYA ÜNİVERSİTESİ	ASP	Evet	Hayır
ULUDAĞ ÜNİVERSİTESİ	ASPX	Evet	Hayır
UŞAK ÜNİVERSİTESİ	ASPX	Evet	Hayır
YILDIZ TEKNİK ÜNİVERSİTESİ	JSP	Evet	Hayır
YÜZÜNCÜ YIL ÜNİVERSİTESİ	ASPX	Evet	Hayır
ZONGULDAK KARAEMLAS ÜNİVERSİTESİ	ASP	Evet	Hayır

Abant İzzet Baysal Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Adıyaman Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Adnan Menderes Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Afyon Kocatepe Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Ahi Evran Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Akdeniz Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Aksaray Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Amasya Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Anadolu Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Ankara Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Atatürk Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Balıkesir Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Boğaziçi Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Bozok Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Celal Bayar Üniversitesi; Programlama dili olarak CGI seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Cumhuriyet Üniversitesi; Programlama dili olarak JSP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Çanakkale On Sekiz Mart Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Çukurova Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Dicle Üniversitesi; Programlama dili olarak CGI seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Dokuz Eylül Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Dumlupınar Üniversitesi; Programlama dili olarak JSP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Düzce Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Ege Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Erciyes Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Erzincan Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Eskişehir Osmangazi Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Fırat Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Galatasaray Üniversitesi; İnternet tabanlı bir öğrenci otomasyonu yoktur. İnternet tabanlı ders kaydı hizmeti sağlamamaktadır.

Gazi Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Gaziantep Üniversitesi; Programlama dili olarak JAVA seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Gaziosmanpaşa Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Gebze Yüksek Teknoloji Enstitüsü Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Giresun Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Hacettepe Üniversitesi; Programlama dili olarak CGI seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Harran Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Hitit Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

İnönü Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

İstanbul Üniversitesi; İnternet tabanlı bir öğrenci otomasyonu yoktur. İnternet tabanlı ders kaydı hizmeti sağlamamaktadır.

İstanbul Teknik Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

İzmir Yüksek Teknoloji Enstitüsü; Programlama dili olarak JSP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Kafkas Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Kahramanmaraş Sütçü İmam Üniversitesi; Programlama dili olarak JSP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Karadeniz Teknik Üniversitesi; Programlama dili olarak JSP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Kastamonu Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Kırıkkale Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Kocaeli Üniversitesi; Programlama dili olarak CFML seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamamaktadır.

Marmara Üniversitesi; İnternet tabanlı bir öğrenci otomasyonu yoktur. İnternet tabanlı ders kaydı hizmeti sağlamamaktadır.

Mehmet Akif Ersoy Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Mersin Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Mimar Sinan Güzel Sanatlar Üniversitesi; Programlama dili olarak JSP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Muğla Üniversitesi; Programlama dili olarak JSP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Mustafa Kemal Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Namık Kemal Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Niğde Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Ondokuz Mayıs Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Ordu Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Orta Doğu Teknik Üniversitesi; Programlama dili olarak CGI seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Pamukkale Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Sakarya Üniversitesi; Programlama dili olarak PHP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Selçuk Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Siirt Üniversitesi; Programlama dili olarak CGI seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Süleyman Demirel Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Trakya Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Uludağ Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Uşak Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Yıldız Teknik Üniversitesi; Programlama dili olarak JSP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Yüzüncü Yıl Üniversitesi; Programlama dili olarak .NET dil ailesi (C#, VB.NET) seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Zonguldak Karaelmas Üniversitesi; Programlama dili olarak ASP seçilmiştir. SSL güvenlik protokolü kullanmamaktadır. İnternet tabanlı ders kaydı hizmeti sağlamaktadır.

Tablodan da görüleceđi gibi bu 68 üniversiteden 65'inin, öğrenci bilgi sistemi uygulamaları mevcut olup, öğrencilerine internet tabanlı ders kaydı sunmaktadır. 3 tanesinin ise öğrenci bilgi sistemi bulunmamaktadır ve halen klasik kayıt sistemi yürütmektedirler. Bu rakamlardan da anlaşılacağı üzere, Türkiye'deki üniversitelerin tamamına yakını, öğrenci kayıt işlemlerini dijital ortama taşımışlardır. İstanbul Üniversitesi'nin de içinde bulunduğu az sayıda üniversitenin bazısı, köklü ve dađınık yapısından ötürü bazısı da az sayıda öğrencisi olmasından dolayı kayıt işlemlerini halen dijital ortama taşıyamamıştır. Bu üniversitelerin öğrencilerinin de internette üye oldukları tartışma forumlarında, konudan şikâyetçi oldukları görülmektedir.

Tüm bunlar incelendikten sonra öncelikle fakültemiz daha sonra da üniversitemiz için böyle bir sistemin gerekliliđi düşünülerek, bu tez kapsamında. NET Teknolojisi Kullanılarak İnternet Tabanlı Bir Ders Kayıt Sistemi geliştirilmiştir. Bir sonraki bölümde bu sistem incelenecektir.

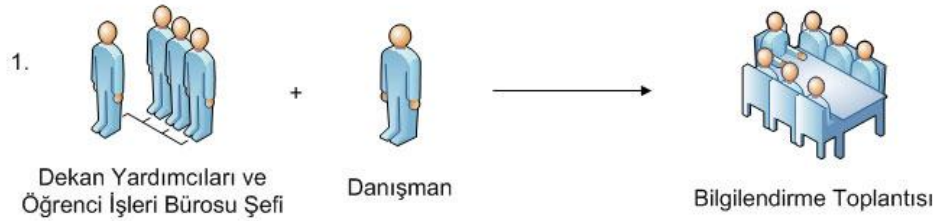
### 3. .NET TEKNOLOJİSİ KULLANILARAK İNTERNET TABANLI ÖĞRENCİ KAYIT SİSTEMİ GELİŞTİRİLMESİ

Bu bölümde önce Hasan Ali Yücel Eğitim Fakültesi'nde ders kaydı için yürütülen mevcut sistemden bahsedilecektir, daha sonra sırasıyla bu sisteme alternatif olarak tez çalışması kapsamında geliştirilen İTOKS'un geliştirme ortamının seçilmesi, uygulaması ve uygulanmasına değinilecektir.

#### 3.1. HASAN ALİ YÜCEL EĞİTİM FAKÜLTESİ'NDEKİ MEVCUT DERS KAYIT SİSTEMİ

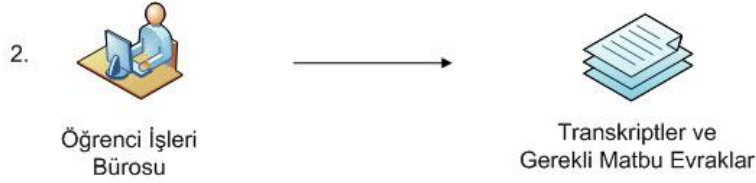
Hasan Ali Yücel Eğitim Fakültesi'nde, öğrencilerin ders kaydı için İstanbul Üniversitesi'nin genelinde olduğu gibi klasik ders kayıt sistemi uygulanmaktadır. Kullanılan sisteminin aşamaları şu şekildedir;

1. Fakülte dekan yardımcıları ve öğrenci işleri şefi başkanlığında, danışmanlık yapan tüm öğretim elemanları ile her dönem kayıt yenileme tarihinden önce kayıt yenileme sistemi hakkında bilgilendirme toplantısı yapmaktadırlar.



Şekil 3.1: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 1. adım

2. Kayıt yenileme tarihinden önce öğrenci işleri bürosu tarafından tüm öğrenciler için güncel transkriptler basılmaktadır ve diğer gerekli matbu evraklar hazırlanmaktadır.



Şekil 3.2: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 2. adım

3. Öğrenci işleri bürosu tarafından hazırlanan ders kayıt formları, danışman öğretim elemanları tarafından teslim alınmaktadır.



Şekil 3.3: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 3. adım

4. Kayıt yenileme tarihlerinde öğrenci işleri bürosu tarafından basılan güncel transkriptler, öğrencilere verilmektedir.



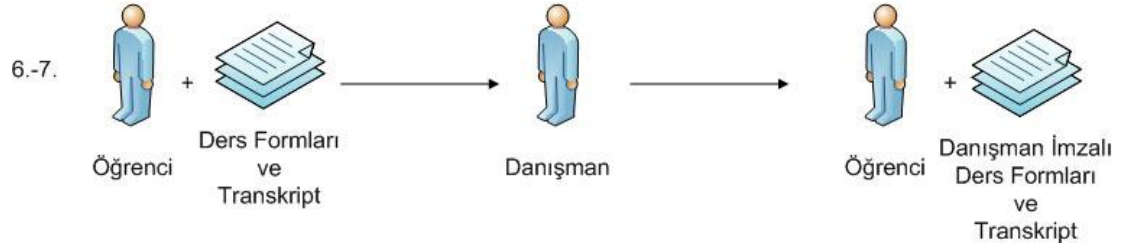
Şekil 3.4: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 4. adım

5. Öğrenciler ders kayıt formlarını doldurmak için önceden belirlenen yer ve zamanda danışman öğretim elemanları ile danışmanlık toplantısı için buluşmaktadırlar.



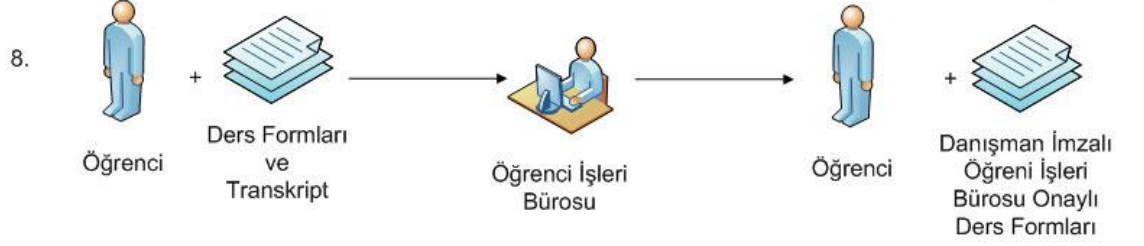
Şekil 3.5: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 5. adım

6. Danışmanlık toplantısında öğrenciler kendilerine üç nüsha şeklinde verilen ders kayıt formlarını doldurmakta ve danışman öğretim elemanının kontrolüne sunmaktadırlar. Verilen ders kayıt formlarına ait bir örnek form EK A'da verilmiştir.
7. Öğrencinin transkriptini ve ders kayıt formlarını alan danışman öğretim elemanı, her bir öğrenci için formları ve transkriptleri inceledikten sonra bir hata yapılmışsa öğrencinin tekrar doldurmasını istemektedir, hata yapılmamışsa formları imzalayıp onaylamaktadır.



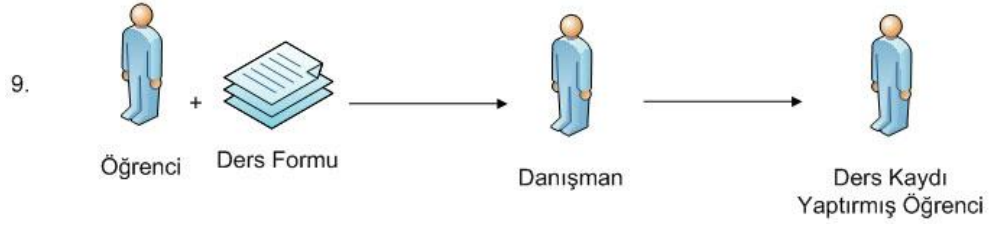
Şekil 3.6: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 6. Ve 7. adım

8. Öğrenci, her anabilim dalı öğrencileri için belirtilen zamanda danışman öğretim elemanı tarafından onaylanmış ders kayıt formlarını, öğrenci işleri bürosuna giderek onaylatmaktadır ve bir nüshasını öğrenci işleri bürosunu bırakmaktadır.



Şekil 3.7: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 8. adım

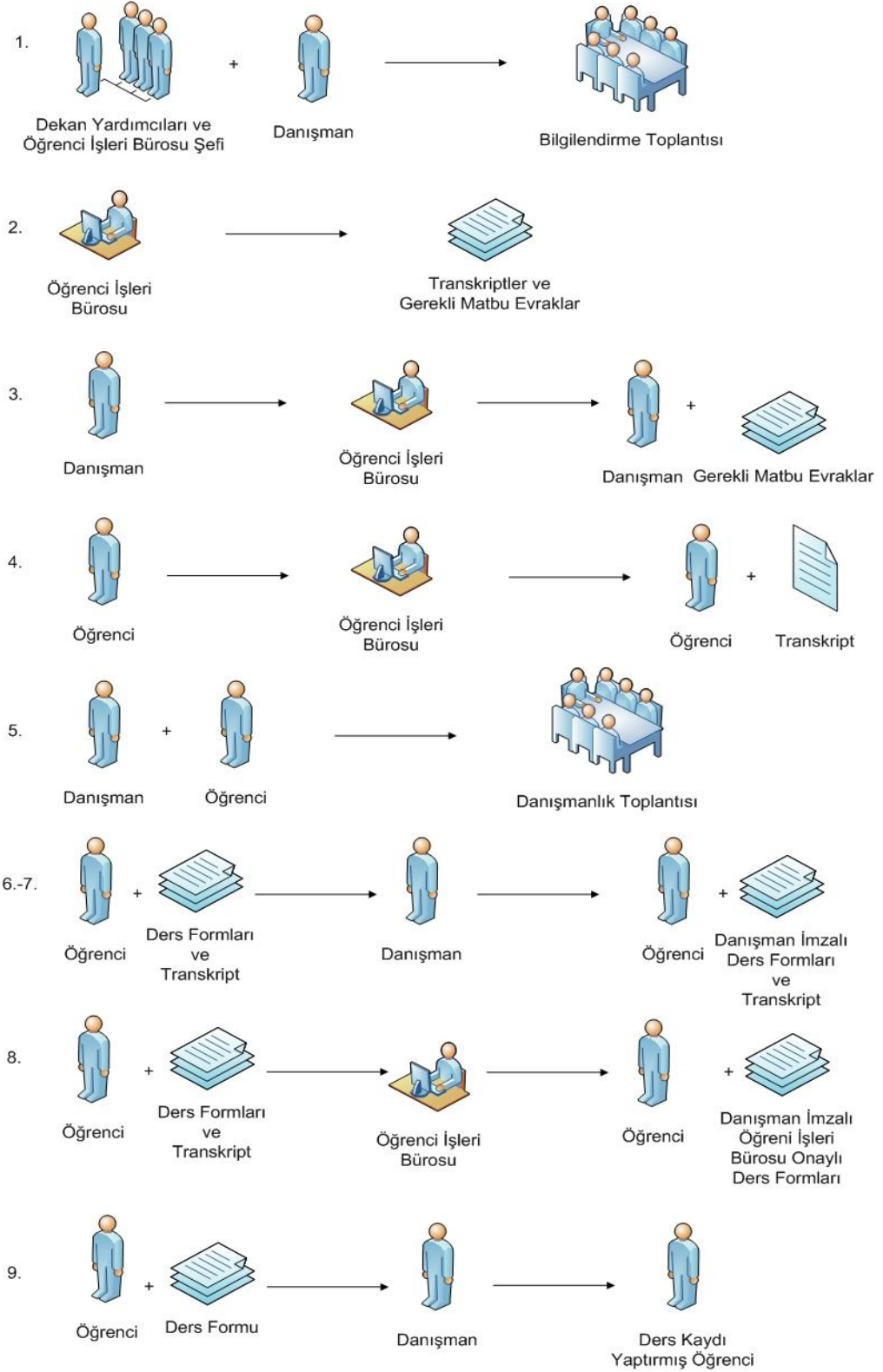
9. Üç nüshadan bir diğerini de danışman öğretim elemanına getirip bırakmakta ve diğer nüshayı da muhafaze etmektedir.



Şekil 3.8: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması 9. adım

Listelenen dokuz adım sonunda öğrenci ders kayıt yenileme işlemini gerçekleştirmiş olmaktadır.

HASAN ALİ YÜCEL EĞİTİM FAKÜLTESİ  
DERS KAYIT AŞAMALARI



Şekil 3.9: Hasan Ali Yücel Eğitim Fakültesi'ndeki mevcut ders kayıt sistemi aşamalarının incelenmesi şeması tüm adımlar

### 3.2. GELİŞTİRİLEN İNTERNET TABANLI DERS KAYIT SİSTEMİ

Tez çalışması kapsamında İstanbul Üniversitesi'nin genelinde olduğu gibi Hasan Ali Yücel Eğitim Fakültesi'nde de uygulanan klasik ders kayıt sisteminde ortaya çıkan birçok problemi ve şikâyeti ortadan kaldırmak için alternatif olarak İTOKS adı verilen uygulama geliştirilmiştir.

İTOKS'u geliştirirken aşağıdaki adımlar takip edilmiştir.

- **Ön Analiz:** Geliştirilecek olan İTOKS için, önce klasik ders kayıt aşamaları detaylı bir şekilde araştırılmış, sonra değişik internet tabanlı kayıt sistemleri incelenmiştir. Bir kayıt sisteminde bulunması gereken kısımlar analiz edilmiştir. Sistem analizi yapılırken temel yaklaşımlar oluşturulup sistem çözümlenmesi yapılmıştır.
- **Sistem Planlama:** Bu adımda geliştirme ortamı belirlenmesi için araştırma yapılmıştır. Bu konudaki araştırma bulguları sonraki bölümlerde anlatılmaktadır. Ayrıca hangi modüllerin kullanılacağı belirlenmiştir.
- **Sistemi Gerçekleştirme:** Bu adımda bir gerçekleştirme planı hazırlanmış ve geliştirilmiş olan yazılımın veri tabanı planı çıkartılmıştır. Bu diyagram ışığında prototip yazılım geliştirilmeye başlanmış ve üzerinde denemeler yapılmıştır. Bu denemeler vasıtası ile yazılımın, test edilmesi aşaması için gerekli olan kararlı sürümü hazırlanmıştır.
- **Sistemin Kullanıma Açılması:** Bu adımda geliştirilen İTOKS, test amaçlı olarak Hasan Ali Yücel Eğitim Fakültesi öğrencilerinin kullanımına açılmıştır.

#### 3.2.1. Geliştirme Ortamının Seçilmesi

Geliştirmede kullanılabilecek programlama dilleri ana hatlarıyla iki kısma ayrılabilir. Bunlardan ilki açık kaynak kodlu programlama dilleri, diğer ise yazılım şirketlerinin kapalı kaynak kodlu geliştirdikleri platformlardır. Bunlara aynı zamanda platform bağımlı ve platform bağımsız diller de denilmektedir. Çok yakın bir zamana kadar platform bağımsız dillerin kullanımı; çok ve çabuk taraftar toplayabilmesi, dolayısıyla hızlı gelişim gösterebilmesi, herhangi bir açık tespit edildiğinde hızlı bir şekilde kapatılabilmesinden dolayı yaygındı. Daha sonra, şirketler geliştirdikleri platformları yenilemişlerdir. Örneğin Microsoft tarafından geliştirilen ASP programlama dili, firmanın yeni geliştirdiği “.NET” platformuna aktarılmış ve platform bağımsız dillerin yükselen grafik eğrisi düşürülmediyse de kırılmıştır. Bu iki farklı dil yapısının birbirine

göre avantajları ve dezavantajları olduğundan dolayı dillerin seçimi, sistemin yapısına ve sistemin kullanıcılarına göre yapılmaktadır.

#### Platform Bağımsız Dillerin Avantajları

- Diğer şirketlerin önceki platformlarına göre daha hızlı olması.
- Herhangi bir işletim sistemine bağlı olmaması. Örnek olarak Apache, Microsoft IIS ve Microsoft PWS'in PHP dilini desteklemesi.
- Kodların açık olması dolayısıyla maliyetlerinin çok düşük olması ve uygulama geliştiricilerin istedikleri anda istedikleri değişiklikleri yapabilmeleri.

#### Platform Bağımsız Dillerin Dezavantajları

- Diğer dillere nazaran hata kontrollerinin daha zor olması.
- Yapılan bir hatanın kolaylıkla birden çok sayfada soruna neden olabilmesi.
- Cevap alma süresinin diğerine göre daha uzun olması.

#### Şirket Platformlarının Avantajları

- Kodların derlenmesinin ve işleme alınmasının hızlı olması.
- Yeni tamponlama sistemi (caching) sayesinde, sayfaların her kullanıldığında yeniden aratmak yerine tamponlama sistemi kullanılarak hızlı erişimin sağlanması.
- Daha güçlü ve daha geniş veri tabanı fonksiyonlarının olması.
- Geniş dil desteği sayesinde web uygulamalarının kolayca hizmete sunabilmesi.

#### Şirket Platformlarının Dezavantajları

- Tek bir işletim sistemine bağımlı olması.
- Platformun ücretli olması.
- Gerekli olabilecek eklentilerin ücretli olması.
- Hosting maliyetlerinin diğerlerine göre yüksek olması [5].

Bir programlama dilini öğrenme sırasında edindiğimiz deneyimler, bir yabancı dili öğrenme sırasında edindiğimiz deneyimlerle neredeyse aynıdır. İkisinin de belli kuralları vardır, kendilerini günceller, kişiden kişiye göre kullanımı değişir ve en

önemlisi zamanla daha iyi öğrenilir. Geliştirme ortamını seçerken bu deneyim ön plana çıkmıştır.

Web programlama, dünya çapında ağ standartlarını belirleyen örgüt olan The World Wide Web Consortium (W3C) tarafından standart haline getirilen HTML (Hyper Text Markup Language) ile başlamış ve Computer-generated imagery (CGI), JavaScript, Active Server Pages (ASP), Virtual Reality Modeling Language (VRML) gibi dillerle devam etmiştir. Bu dillerin bazıları sunucu tarafında bazıları ise istemci tarafında çalışmaktadır. İlk olarak HTML'i ele alalım; HTML dosyalarının içeriği bağlanılan sunucu tarafından, bağlanan bilgisayara yollanmaktadır ve bağlanan bilgisayarda bulunan web görüntüleyici (Internet Explorer, Mozilla Firefox gibi) tarafından sayfa anlamlandırılarak gösterilmektedir. Yani; bu yapıda tüm görüntüleme işlemini bağlanan bilgisayar yapmaktadır. HTML'den sonra geliştirilen Java Script teknolojisi de buna yakındır. Yani; dosyalar bağlanılan bilgisayar tarafından bağlanan bilgisayara yollanmaktadır ve bağlanan bilgisayar tarafından düzenlenip görüntülenmektedir. Javascript içeren sayfalar diğer sayfalara göre daha geç yüklenmektedir, çünkü tüm Javascript kodları bağlanan bilgisayarda çalıştırılmaktadır ve bu yüzden performans düşmektedir.

Sunucu Tarafı (server-side) çalışan programlama dillerinde ise tüm kodlar sunucu tarafında çalıştırılır ve bağlanan bilgisayara sadece HTML kodları gönderilir. Eğer web sunuculuğu yapan bilgisayar hızlı ise performans olarak büyük bir avantaj elde edilmiş olur.

Sunucu Tarafı programlamanın diğer bir avantajı ise güvenlidir. Örneğin Java Script kodları, web görüntüleyicide "Kaynağı Görüntüle" seçeneği seçildiğinde görülebilir, Ancak sunucu tarafı programlarda sadece bağlanan bilgisayara gönderilen içerik görülebilmektedir [6].

Yaygın olarak kullanılan sunucu tarafı programlardan kısaca bahsedecek olursak;

PHP, ilk olarak 1993 yılı ortalarında Rasmus Lerdorf tarafından geliştirilmeye başlanmıştır. Lerdorf'un amacı, kişisel bilgilerini internet üzerinden yayınlamaktı. Buradan yola çıkarak, kişisel web sayfası yapmak için Perl dilinden ödünç rutinler kullanarak bir yazılım hazırlamış ve buna Personal Home Pages (PHP) adını vermiştir.

PHP dili çok kısa bir süre sonra geniş anlamda yaygınlaşmıştır. Daha sonra bu dile, form tipi bilgileri işlemeyi sağlayan eklemeler yapılmış ve ismi PHP/FI (Form Interpreter/Form Yorumlayıcı) şeklini almıştır. (Bazı kaynaklarda dilin bu sürümü PHP 2 olarak adlandırılmaktadır). PHP dili, 1995 yılında Lerdorf'un kurmuş olduğu bir grup tarafından daha da geliştirilerek, ilk kez Perl dilindeki fonksiyonlardan tamamen arındırılmış ve Nesne Yönelimli (Object Oriented) bir dil haline getirilmiştir.

2005 yılı itibariyle PHP 5 sürümü geliştirilmiş durumdadır. PHP dili Linux gibi Açık Kaynak Kodlu bir dil olup ücretsiz olarak dağıtılmakta ve geliştirilmektedir. Linux, Unix, Windows tabanlı işletim sistemlerinde çalışabilen sürümleri mevcuttur [7].

JSP (Java Server Pages), teknolojisi basit ve hızlı bir şekilde dinamik olarak üretilen web sayfaları yapmak için kullanılan bir teknolojidir. JSP, Sun Microsystems'in öncülüğünde gerçekleştirilmiştir. JSP, Java Servlet API denilen modülün bir uzantısıdır. Servletler platform-bağımsız tamamen Java modülleri olup, bir web sunucusunun yeteneğini ve yapabileceği işleri arttırmaları. Servletler, diğer diller gibi, platforma özel durumlar ya da değişiklikler gerektirmezler. JSP teknolojisi ve Servletler birlikte, ASP ve Perl gibi dinamik web programlama araçlarına Java temelli bir alternatif teşkil ederler. JSP, XML'e ve scriptlere çok benzeyen etiketlemeler kullanır. Bu etiketlemeler Java programlama dilinde yazılıp sayfanın HTML içeriğini üretirler. Aynı zamanda uygulama mantığı sunucu tabanlı kaynaklarda mesela JavaBean<sup>2</sup> lerin içerisinde gömülü olabilir. Dökümanın içindeki bütün diğer HTML ya da XML etiketleri direkt olarak istemcinin önüne gelecek olan sayfaya gönderilir. Sayfanın akışını ve mantığını dizayn ve görüntüsünden ayırmak ve tekrar kullanılabilen bileşen-tabanlı teknolojiyi kullanabilmek, JSP teknolojisini diğer teknolojilerden daha hızlı ve kolay web uygulamaları yapar hale getirmiştir [9].

ASP (Active Server Pages), Microsoft firması tarafından klasik HTML sayfalarına dinamik bir yapı kazandırmak amacıyla ortaya çıkarılmış sunucu tarafı (server side) çalışan bir teknolojidir. Sunucunun sadece durağan sayfaları istemciye (ziyaretçi) göndermesinin yanı sıra, ziyaretçiden de veri kabul edilmesi ayrıcalığına sahiptir.

---

<sup>2</sup> JavaBean: Java programlama dilinde, belli bir standarta göre yazılmış sınıflardır [8].

ASP sayfaları çoğunlukla VBScript kullanılarak yazılır, ancak bünyesinde bulunan “@Language” komutuyla JScript, hatta Perl gibi başka bir dil de seçilebilir. Bir ASP dosyasının içinde, özel nesnelere ve VBS, JS, SQL adı verilen kodlar bulunur, bu sayfalar istemci tarafından istendiğinde, sunucu öncelikle ASP içindeki kodları icra ederek, istemciye göndereceği bilgiyi oluşturur ve gönderir. Gönderilen bilgi genellikle HTML şeklindedir. Fakat sadece bunlarla sınırlı değildir, aynı şekilde bir grafik dosyası da oluşturulup, istemciye gönderebilir. ASP sayfaları HTML kodlarının içine gömülü şekilde oluşturulduğu halde bir kez sunucu tarafından derlendiğinde tamamen HTML halinde döner. Kaynak kodlara bakıldığında ASP kodları görülmez. Bu durum kodlamacıların kaynaklarını saklamalarını kolaylaştırır [10].

ASP.Net yani Active Server Pages .Net; Microsoft’un yeni vizyonu olan .Net ile desteklenmiş internet uygulamaları ve web servisleri için sunucu taraflı yazılım geliştirmeyi kolay, güvenli ve genişleyebilir kılan bir teknolojidir. Asp.Net’i anlatabilmek için öncelikle .Net ve mimarisinden bahsedilmesi uygun olacaktır;

Yeni bir teknolojinin başarılı olması çok ciddi altyapı desteği ile birlikte sunulmasına bağlıdır. .Net’e bakıldığında, geniş bir kurumsal sunucu ailesi sayesinde destek sağladığı görülmektedir. Bu sunucu ailesine bakacak olursak; SQL Server ile veri tabanları en etkin biçimde yönetilirken, Biztalk Server ile firmalar arasında veri aktarımı XML teknolojisi kullanarak çok kolay gerçekleştirilebilmektedir. Internet Security and Acceleration Server ile güvenlik ve internet altyapılarında hızlanma sağlanırken, Content Management Server ile sitelerin içeriği kolayca yönetilebilmektedir. Mobile Information Server sayesinde tüm yapılan geliştirmeler günümüzün çok geniş teknoloji ürünleri yelpazesinden herhangi biri üzerinde kolayca çalıştırılabilmektedir. Microsoft tarafından geliştirilen sunucuların sayısı her geçen gün artmakta, .Net daha güçlü bir altyapıya taşınmaktadır. Sunucu çeşitliliği sayesinde işler daha modüler bir yapıda gerçekleşmekte ve bir anlamda bu sunucular ile birlikte .Net tamamlanmaktadır [11].

Microsoft, .NET platformunu geliştirirken platform ve programlama dilinden bağımsız, güvenli, tamamen nesne tabanlı bir sistem kurmayı hedeflemiş ve bunu da başarmıştır. Microsoft’un on üç yıllık vizyonu olan bu teknoloji, dünyada ve Türkiye’de giderek artan bir biçimde yazılım geliştiriciler tarafından kullanılmaktadır. .NET, Framework

denilen bir çekirdek sistem üzerinde çalışmaktadır. Framework içerisinde bulunan yapılar sayesinde geliştirilen uygulamalar, hızlı ve güvenilir olarak .NET platformunu destekleyen diğer programlama dilleri ile etkileşimli uygulamalar haline alırlar.

.NET Framework, yazılım geliştirme ve projenin çalışabilmesi için gerekli çekirdek sistemdir. .NET teknolojilerinin çalışabilmesi için uygulamanın çalışacağı sistemde mutlaka .NET Framework yüklü olmalıdır. .Net Framework, JAVA'daki JVM (Java Virtual Machine)'ye benzetilebilir.

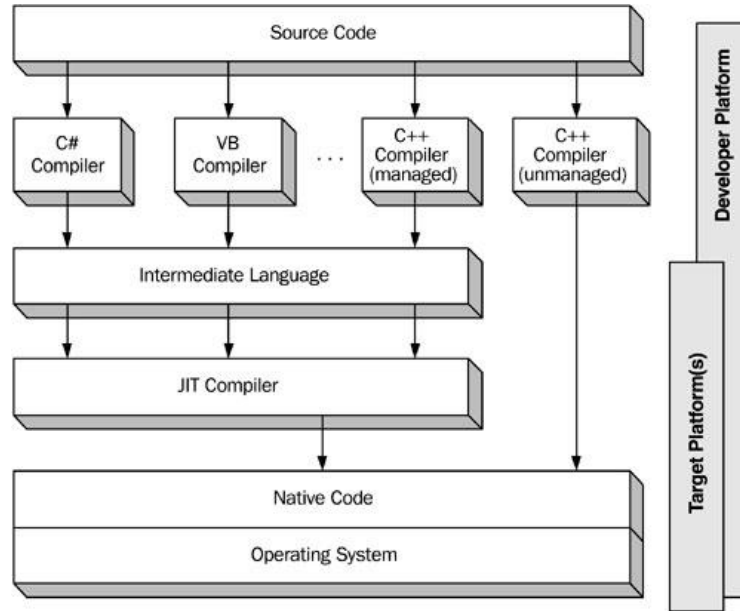
.NET ortamında, .NET programlama dillerinden biri ile yazılmış bir program (buna Assembly denmektedir) derlendiği zaman doğrudan ikili kodlara dönüştürülmez. Öncelikle IL (Intermediate Language - Ara dil) denilen bir yapıya çevrilir ve ardından .Net assembly'si ilk çağrıldığı esnada tekrar derlenir. Bu işleme, JIT (Just In Time - Anında Derleme) denir. Bu sayede assembly nin ihtiyaç duyulan kısımları belleğe taşınır ve assembly ikili koda dönüştürülür. .Net framework ile amaçlanan, projelerin işletim sistemi ve donanımsal ortamlardan soyutlanarak geliştirilmesini sağlamaktır.

.NET Framework çalışırken sistem üzerindeki temel sınıflar kullanılır. Bu sınıflar, tamamen nesne tabanlı, nesne yönelimli programlamanın temel işlevleri olan kalıtım (inheritance), kapsülleme (encapsulation) ve çok biçimlilik (polymorphism)'i destekler. FCL (Framework Class Library) de denilen bu temel sınıflar, programlama dilinden bağımsız oldukları için .NET platformunu destekleyen tüm programlama dilleri tarafından kullanılabilir. FCL'deki bu sınıflar windows ve web için çeşitlilik göstermekle birlikte, her iki grubun kullandığı ortak sınıflar olduğu gibi kendilerine özel sınıflar da bulunmaktadır [12].

.NET ile birlikte gelen CLR (Common Language Runtime) ise .NET altyapısında programların çalışmasını kontrol eden ve işletim sistemi ile program arasında yer alan arabirimdir. Normalde yazılan programlar derlenirken makine diline çevirilmektedir ve program bu şekilde işletim sistemi ile direkt bağlantı kurarak çalışmaktadır. Fakat platform bağımsız bir geliştirme ve yürütme ortamı istendiğinde CLR devreye girer ve .NET programlarını farklı platformlarda makineye ve işletim sistemine göre çalıştırır. Normalde bir Windows, Linux veya MACOS kurulu sistemler aynı programın kodunu

çalıştıramazlar. Bu platformlar için programın ayrı ayrı yazılıp, onlara göre hazırlanmış derleyicilerde derlenmesi gerekmektedir. Dünyada çok sayıda yaygın platform olduğunu düşünüldüğünde, bunların herbiri için ayrı ayrı derleme işlemini tek bir işletim sisteminde yapmak imkânsız gibidir. Bu durumda çözüm, ortak bir aradil kullanmak ve herbir platform için bu aradile çevrilmiş programın kodunu çalıştıracak altyapıları hazırlamaktır. .NET'in kullandığı bu aradil MSIL (Microsoft Intermediate Language) 'dir.

.NET platformunda hangi dili kullanırsak kullanalım yazdığımız programın kodu direkt olarak makine diline değil de MSIL'e çevrilir. Daha sonra, programı çalıştırdığımız sistemde kurulu olan CLR, MSIL kodlarını çevirerek programımızı çalıştırır, çalışma anında derleme işlemini ise JIT derleyicileri (Just In Time Compilers) üstlenir [13].



Şekil 3.10: Just In Time Compiler[14]



Şekil 3.11: CLR'nin bulunduğu konum

Yukarıdaki şekilde de görüldüğü gibi işletim sisteminin hemen üstünde .NET frameworkü kullanan tüm programlar CLR'nin denetimi altındadır. CLR'nin hemen üstünde FCL denilen temel sınıflar bulunmaktadır. En üstte de uygulama türleri görülmektedir. Gerek web uygulamaları gerekse windows uygulamaları temelde aynı temel nesnelere ihtiyaç duymaktadırlar [12].

Geliştirdiğimiz uygulamada sıkça kullanmış olduğumuz XML Web Servisleri'nden de bahsedecek olursak;

Etkin olarak kullanılmayan bilgi önemini kaybeder. Bilgi paylaşılmalı, yeniden kullanılabilir, sınıflandırılmalı, işlenebilir, depolanabilir, kısaca bilgi yönetilebilir. Ancak yazılı, sözlü ve görsel veriler (dökümanlar), aşırı miktarlarda olmaları ve birden fazla platformda kullanılmaları sebebiyle, hem aktarılma hem sorgulama hem de başka bir sisteme yerleştirilmelerinde maliyet, zaman ve işgücü kaybına yol açarlar. Öyleyse ihtiyaç duyulan, her ortamda donanım ve yazılımdan bağımsız veri transferini ve yönetimini kolaylaştıran bir standarttır.

XML (Extensible Markup Language); verilerin transferi, depolanması, sorgulanması ve yönetiminde; veriye içerik değeri katması, ihtiyaç duyduğumuz sistemi yaratabilme esnekliği sunması, dağınık verilerin kümelenmesi, karşılaştırma yapma kolaylığı, farklı veri formatlarını ve dilleri destekleyebiliyor olması ve tüm sistemlerle çalışabilme özelliğiyle bugün ve gelecekte ihtiyaç duyulan veri standardıdır. Veri öbekleri XML'le bilgiye dönüştürülür [15].

XML, deęişik veri, kavram ve içeriklerin tanımlanması ve temsil edilmesi için uygun bir ortam sunmaktadır. Bu nedenle XML farklı alanlarda uygulama verilerinin tanımlanması ve taşınması için üretici, dil ve platformdan bağımsız stratejik bir araç olarak hızla yaygınlaşmaktadır.

SGML (Standard Generalized Markup Language), ISO (International Standards Organization) tarafından 1986 yılında kabul edilmiş ve onaylanmış bilgi yönetimi standardıdır (ISO 8879:1986) [16]. SGML, platform-bağımsız ve uygulama-bağımsız dokümanlar yaratmak ve sağlamak için oluşturulmuştur. Dilbilgisi gibi bir mekanizma kullanılarak dokümanların yapısını, özel etiketler kullanarak tanımlamamıza yarar [17]. SGML, bir meta dildir. Yani yeni diller üretmeye müsait bir yapıya sahiptir.

Bağımsız bir kuruluş olan W3C (World Wide Web Consortium) organizasyonu tarafından tasarlanan ve herhangi bir kurumun tekelinde bulunmayan XML, Genişletilebilir İşaretleme Dili; SGML 'den türetilmiş, basit ve çok esnek bir işaretleme dildir. Başlangıçta büyük ölçekli elektronik yayınlardaki zorluklarla başa çıkmak için tasarlanan XML'in, daha sonraları geniş ölçekli verinin, web ve diğer ortamlar arasında deęiş tokuşunda önemli rol oynamasıyla popülaritesi artmıştır [18].

SGML'den türetilmiş işaretleme dilleri, uygulamalar içinde işlenecek ya da farklı ortamlarda bulunsalar bile uygulamalar arasında dağıtılabilecek verilerin tanımlanmasını ve dağıtımını kolaylaştırmaları bakımından önemlidirler. XML, HTTP (Hypertext Transfer Protocol)'yi kullanarak da, veri işleme konusunda ki güçlü yapısından dolayı kısa bir sürede evrensel bir protokol haline gelmiştir.

XML, karmaşık problemlere çok basit çözümler sunmaktadır. Veriyi ya da bilgiyi, kendi kendini tanımlayan doküman formatı içinde standart bir yapı sunar. XML, veri yapısını kendi içinde tanımladığından dolayı platform bağımsızdır. Ayrıca düz metin olarak tutulmasından dolayı HTTP protokolü ile bir yerden başka bir yere kolayca transfer edilebilir. Böylelikle XML, verinin transferi ve uygulamalar arasında alış verişinde diğer herhangi bir kısıtlamaya maruz kalmaz [19].

XML'in yapısının daha iyi anlaşılabilmesi için;

3214209323	Burak	Şişman	75	87
3214209324	İrfan	Şimşek	78	95

Yukarıdaki satırların gerçekten hangi tür bilgiyi gösterdiğini tahmin etmek çok kolay değildir. Ancak Şekil 3.12'deki XML dosyasının içeriğine bakıldığında, dosyada ne tür bir bilgi bulunduğu hakkında daha çok fikir sahibi olunmaktadır.

```

<?xml version="1.0" encoding="iso-8859-9" ?>
<!--Sınav Sonuçları-->
<dersler>
  <ogrenci no="3214209323">
    <adı>Burak</adı>
    <soyadı>Şişman</soyadı>
    <vize>75</vize>
    <final>87</final>
  </ogrenci>
  <ogrenci no="3214209324">
    <adı>İrfan</adı>
    <soyadı>Şimşek</soyadı>
    <vize>78</vize>
    <final>95</final>
  </ogrenci>
</dersler>

```

Şekil 3.12: Örnek bir XML dosyası

Şekil 3.12'deki XML dosyasında da olduğu gibi, ilk satır, dosyanın XML dosyası olduğunu ve burada bulunan XML'in sürüm bilgisi ile dosyadaki metnin karakter kodlama numarası bulunmaktadır.

Çok dilli belgeleri ve evrensel karakter kodlamasını destekleyen XML, uygulamaların uluslararası hale getirilmesinde önemli avantaj sağlar. Satırın kodlama parçası dosya içeriği işlenirken önem kazanır. Dosyada, belirtilen karakter kodlamasında bulunmayan bir karaktere ya da karakter dizisine rastlandığında hata üretilecektir. Burada belirtilen kodlama türü, ekran görüntüsünün kodlama türü değil, başlığı barındıran dosyanın karakter kodlamasıdır. Türkçe belgeler için "iso-8859-9" İngilizce belgeler için "iso-8859-1" ya da "utf-8" evrensel kodları kullanılmaktadır.

XML belgelerinde, HTML de olduğu gibi önceden belirlenmiş etiketler bulunmaz, etiketler kullanıcı tarafından esnek bir şekilde belirlenir. XML belgeleri, platformlar ve uygulamalar arası standart bir veri transfer yöntemi olması ve kullanıcı tanımlı genişleyebilir, evrensel bir yapıda olması dolayısıyla, paylaşılacak verinin dilbilgisi kuralının tanımlama bilgisine ihtiyaç vardır. Böylelikle uygulamalar, hangi tipte veri alış verişinde bulunacakları konusunda uyumlu olurlar.

DTD (Data Type Definition) tanımı ya da XSD (XML Schema Definition), XML çözümleyicisine gerekli etiketleri, isteğe bağlı olan etiketleri ve bir etiketin izin verilen değerlerini belirtilerek, belgenin dilbilgisini belirlerler. Her ne kadar XML esnek bir yapıya sahipse de, DTD de ya da şemada belirtilen kurallara, sıkı sıkıya uyulması gerekir. XML belgesinin bu kurallara uymayan kısmı olursa çözümleyici bir hata verir. Bu kurallara uygun olarak hazırlanmış XML belgesine, iyi biçimlenmiş (well-formed) belge denir. Şekil 3.13'deki örnek DTD'ye sahip XML belgesinin, *dersler* isimli kök elementin bulunduğu ve bu elementin *ogrenci* elementlerinden oluştuğunu, *ogrenci* elementinin ise içinde veri bulduran *adı*, *soyadı*, *vize* ve *final* isimli dört element ve *no* isimli zorunlu bir özniteliğe sahip olduğunu anlamaktayız.

```
<!ELEMENT dersler (ogrenci+)>
<!ELEMENT ogrenci (adı, soyadı, vize, final)>
<!ATTLIST ogrenci no CDATA #REQUIRED>
<!ELEMENT adı (#PCDATA)>
<!ELEMENT soyadı (#PCDATA)>
<!ELEMENT vize (#PCDATA)>
<!ELEMENT final (#PCDATA)>
```

Şekil 3.13: Basit bir DTD örneği

XML belgelerinde bilgi, etiketler arasında sunulur. Şekil 3.14'de gösterildiği gibi her bir belge bir adet kök elemente sahip olmalıdır. Bu kök içinde ağaç yapısı şeklinde diğer elementler sıralanabilirler.

```

<?xml version="1.0" encoding="iso-8859-9" ?>
<!-- Yorum Satırı -->
<Etiket1>
  <Etiket2>
    <Etiket3>
      ...
    </Etiket3>
  </Etiket2>
</Etiket1>

```

Şekil 3.14: XML dosyası element yapısı

Etiketlemede dikkat edilecek bazı kurallar vardır. Etiket isimleri daima < ve > karakterleri arasında olmalıdır. Dolayısıyla element içindeki bilgi içinde < ve > işaretleri gibi XML belgeleri için anlam ifade eden karakterler kullanılamazlar. Bunların yerine Tablo 3.1'deki gibi ifadeler kullanılmalıdır.

Tablo 3.1 : XML de kullanılmayacak karakterlerin yerine kullanılacak ifadeler

Karakter	Yerine
<	&lt
>	&gt
&	&amp
'	&apos
"	&quot

Etiket ve değişken isimleri daima bir harfle başlamalıdır. SGML ve HTML için etiket ismini oluşturan harflerin büyüklüğünün önemi yoksa da XML için küçük büyük harf duyarlılığı vardır. XML'deki bu özellik, temel aldığı kodlamadan kaynaklanmaktadır.

Bir etiket tanımında, içinde veri içermesi öngörülmüşse, her açılış etiketini bir kapanış etiketi izlemelidir. HTML ve SGML deki bazı durumlarda bitiş etiketi verilmeyebilirse de XML için bitiş etiketleri zorunludur.

Bir etiketin bitiş belirleyicisi “/” işaretidir. Bir veriyi tanımlayan etiketlerde, bu işaret, etiket isminin başına, işlem etiketlerinde ise sona konur.

XML belgesinin içine yazılacak olan yorum metinleri <!-- --> notasyonları arasına yazılmalıdır. Örnek,

<!-- Bu bir yorum satırıdır -->

Tüm bu bilgilerin ışığında ASP.NET; “.Net Framework” altında işletim sistemi ile bütünleşik çalışan bir mimaridir. Klasik uygulamaların kullanabildiği nesnelere aynı şekilde kullanabildiği gibi, .Net’in tüm nesnelere erişebilir ve türetebilir. ASP.NET sunucu ve istemciyi çok iyi bir şekilde birleştirir. İstemcinin kullandığı sistem özelliklerine özgü içeriği kullanabilir. Böylelikle istemci tarafında, bazı işlemlerin gerçekleşmesi, işi hızlandırıyor bu otomatik olarak tespit eder ve uygular. Bunu yaparken yazılım geliştiricinin ayrıca bir işlem yapmasına gerek yoktur.

ASP, sunucu tarafından yorumlanan bir teknoloji iken ASP.NET derlenmesi gereken bir teknolojidir. Derlenen kod, çok büyük bir performans artışı sağladığından uygulamalar çok daha hızlı çalışacaktır. ASP.NET bileşen mimarisine yeni bir boyut getirmektedir. Artık sisteme nesne yüklerken tanıtıcı kütüphane dosyası olan “regsvr32” kullanmak yerine .NET ile gelen bir metadata sayesinde dll dosyaları kendi açıklamalarını kendi üzerlerinde taşımaktadırlar. Bu yüzden ayrıca bir kayıt işlemine ve sunucuyu yeniden başlatmaya gerek yoktur.

Oturum yönetiminde ise, sunucu belleği üzerinde yoğunlaşan bir oturum mimarisi yerini dağıtık internet uygulamalarına bırakmıştır. Oturum bilgileri, SQL Server, State Server gibi yeni sunucular kullanılarak sunucunun belleğinden ayrılmıştır. Bu şekilde sunucular arası veri paylaşımı sağlanmış, güvenli ve dengeli bir yapıda sistem kurulmuş olmaktadır. ASP’de cookielere<sup>3</sup> dayanan mimari yüzünden yaşanan sorunlar aşılmış, istenirse “cookieless” yani cookie teknolojisi kullanılmadan da oturum yönetimi sağlanabilir hale gelmiştir.

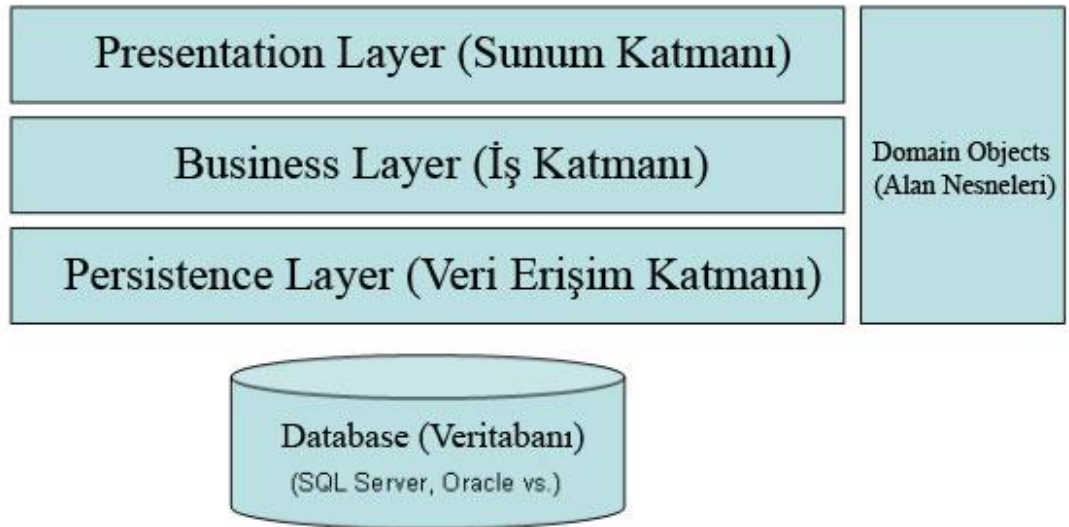
ASP.NET’de içerikler, program kodundan tamamen ayrılmıştır. Kullanıcı kontrolleri ve “codebehind” (arka planda kod) gibi özellikler sayesinde görsel içerik ile uğraşan kişiler sadece görsellikle, programlama kısmı ile ilgili kişilerin de sadece program kodu ile ilgilenmesine olanak tanımaktadır. İstemci-sunucu mimarisi oldukça başarılı bir şekilde

<sup>3</sup> Sunucu tarafından, ziyaretçinin daha sonraki ziyaretlerinde kullanılmak üzere oturum bilgilerini tutmak için sabit sürücüsüne yerleştirilen küçük metin dosyasıdır. Türkçe’ye çerez olarak çevrilmiştir

ASP.NET’de uygulanmış, yazılım geliştiricinin arka planda neler olduğunu bilmesine gerek kalmadan uygulamalarda önbellekleme ve performans düzenlenmiştir. ASP.NET tamamen nesneye yönelik programlamanın kullanılabilmesini sağlayan bir teknolojidir. ASP’de JavaScript veya VbScript kullanabiliyorken ASP.NET’de uygulama geliştirmede kullanılan diller olan Visual Basic, C# gibi dilleri kullanmak mümkündür. Böylelikle bir programlama dili kullanmanın tüm avantajları uygulamalarda kullanılabilir. Yani ASP.NET sunucu taraflı yazılım geliştirmeye farklı bir boyut getirmektedir [11].

Geliştirdiğimiz uygulamada kullanmış olduğumuz Nhibernate hakkında da bilgi verecek olursak;

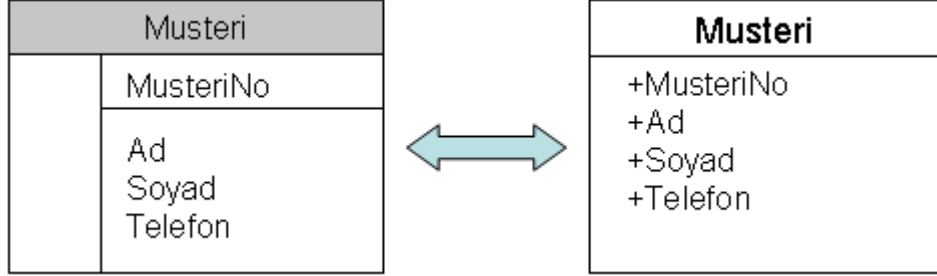
Nhibernate, Object/Relational Mapping (O/RM) ‘in bir aracıdır. O/RM ise, ilişkisel veritabanı sistemlerinde bulunan tablolar ile uygulama tarafında bulunan nesnelere ilişkilendirilmesi işlemine verilen genel bir addır. O/RM frameworkü bireysel olarak oluşturulabileceği gibi önceden hazırlanmış yaygın ve güvenilirliği ispatlanmış O/RM alt yapıları da kullanılabilir. O/RM kısaca Çok Katmanlı Mimarinin<sup>4</sup> Persistence Layer olarak da isimlendirilebilen Veri Erişim Katmanıdır.



Şekil 3.15: Çok Katmanlı Mimari

<sup>4</sup>Çok Katmanlı Mimari (N-Tier Architecture): Bir uygulamayı yeniden yazmak yerine çoklu katmanlara ayırarak bağımsız düzeltmeler yapılmasını sağlayan bir mimaridir [20].

O/RM araçları, kendi alt yapılarında tasarlandığı şekilde hazırlanmış olan mapping yapılandırmasına göre, ilişkisel veritabanı yönetim sisteminde yer alan tablolar ile bunlara karşılık gelen uygulama içindeki Domain Objectler arasında veri aktarımını otomatik olarak sağlar. Böylece uygulama geliştirici ayrıca kod yazmamış olur.



Şekil 3.16: O/RM ile Tablo kayıtlarının nesnelere yüklenmesi

Nesnelere saklanması, nesnelere ulaşılması, silinmesi, güncellenmesi gibi işlemler için alt yapıda tanımlanmış olan veritabanı yönetim sisteminin anlayacağı dilde SQL<sup>5</sup> sorgusunu oluşturulması ve işletilmesi tamamen otomatik olarak O/RM framework'ünün işidir. Birçok O/RM framework'ü farklı veritabanı yönetim sistemlerini desteklemektedir. Örneğin; Oracle ile çalışan bir uygulamayı SQL Server'a aktarmak (ya da tam tersi) gibi bir senaryo için genellikle yapılması gereken sadece bir tanımlamanın değiştirilmesinden ibarettir.

O/RM araçları nesnelere erişmek için kendi nesne modellerinde tanımlanmış metodlar sunmaktadır. Birçok O/RM aracı metodlar ile sunulan genel yapıların yanında daha özel ve karmaşık işlerde kullanmak için "Object Query" olarak tabir edilen kendi sorgulama dillerine de sahiptir.

O/RM araçlarından en bilinen ve başarılı olanları NHibernate, LLBLGen Pro gibi araçlardır. Mevcut .NET O/RM araçlarının listesi için Tablo 3.2 incelenebilir.

<sup>5</sup> SQL: İngilizce "Structured Query Language" kelimelerinin baş harfleri ile oluşturulmuş, Türkçesiyle Yapısal Sorgulama Dilinin kısa adıdır. SQL cümleleri kullanılarak veri tabanına kayıt eklenebilir, mevcut kayıtlar değiştirilebilir, silinebilir ve bu kayıtlardan listeler oluşturulabilir [21]

Tablo 3.2: .NET için O/RM araçları [22]

Adı	Son Versiyonu	Web Adresi
Opf3	3.0.4.341	<a href="http://www.opf3.com">http://www.opf3.com</a>
Retina.NET	2.0	<a href="http://www.codeplex.com/retina">http://www.codeplex.com/retina</a>
Dali	2.2	<a href="http://www.revtechnologies.com">http://www.revtechnologies.com</a>
ODX.NET	1.6.1	<a href="http://www.codeplex.com/odx">http://www.codeplex.com/odx</a>
LightSpeed	2.0	<a href="http://www.mindscape.co.nz">http://www.mindscape.co.nz</a>
Evaluant Universal Storage Services	0.9.1	<a href="http://www.codeplex.com/euss">http://www.codeplex.com/euss</a>
TierDeveloper	5.6	<a href="http://www.alachisoft.com">http://www.alachisoft.com</a>
WilsonORMapper	4.2.2.0	<a href="http://code.google.com/p/wilsonormapper">http://code.google.com/p/wilsonormapper</a>
Nolics.net 2005	4.2.2525.4	<a href="http://www.nolics.net">http://www.nolics.net</a>
Genome	3.3.7	<a href="http://www.genom-e.com">http://www.genom-e.com</a>
DevForce	3.5	<a href="http://www.ideablade.com">http://www.ideablade.com</a>
Elementary	0.8.1	<a href="http://www.adapdev.com/elementary">http://www.adapdev.com/elementary</a>
GURA Object Director	2.0	<a href="http://www.gura.com.au">http://www.gura.com.au</a>
Telerik OpenAccess	4.3.19	<a href="http://www.telerik.com">http://www.telerik.com</a>
Vanilla DAL	0.2.2	<a href="http://sourceforge.net/projects/vanilla-dal">http://sourceforge.net/projects/vanilla-dal</a>
Diamond Binding	1.2.1.3	<a href="http://dunnchurhill.com/products/diamondbinding">http://dunnchurhill.com/products/diamondbinding</a>
Neo	1.3.1	<a href="http://neo.sourceforge.net">http://neo.sourceforge.net</a>
CoolStorage.NET	1.2.0	<a href="http://www.codeplex.com/CoolStorage">http://www.codeplex.com/CoolStorage</a>
Grove	2.1	<a href="http://www.grovekit.com">http://www.grovekit.com</a>
MyGeneration	1.2.0.6	<a href="http://www.mygenerationsoftware.com">http://www.mygenerationsoftware.com</a>
DataBlock	1.3.1	<a href="http://www.voidsoft.ro">http://www.voidsoft.ro</a>
DomainObjects	1.1.1	<a href="http://domainobjects.sourceforge.net">http://domainobjects.sourceforge.net</a>
Habanero	2.0.0	<a href="http://www.chillisoft.co.za/habanero">http://www.chillisoft.co.za/habanero</a>
POTIS Object-Relational Toolkit	1.5	<a href="http://www.port.com.pl">http://www.port.com.pl</a>
Persistor.NET	1.5	<a href="http://www.persistor.net">http://www.persistor.net</a>
Sooda	1.0	<a href="http://sooda.sourceforge.net">http://sooda.sourceforge.net</a>
DataObjects.NET	3.8.5	<a href="http://www.x-tensive.com">http://www.x-tensive.com</a>
NJDX	1.5	<a href="http://www.softwaretree.com">http://www.softwaretree.com</a>
NObject	2.0.0.1087	<a href="http://www.macrobobject.com">http://www.macrobobject.com</a>
LLBLGen Pro	2.5	<a href="http://www.llblgen.com">http://www.llblgen.com</a>
Persistent Datasets	2.0.3	<a href="http://www.lastcomponent.com">http://www.lastcomponent.com</a>
eXpress Persistent Objects for .NET	v2008 vol 3.3	<a href="http://www.devexpress.com">http://www.devexpress.com</a>
Lattice.DataMapper	3.0	<a href="http://www.latticesoft.com">http://www.latticesoft.com</a>
DADO Solution	1.1	<a href="http://www.dadosolution.com">http://www.dadosolution.com</a>
SubSonic	2.0.3	<a href="http://subsonicproject.com">http://subsonicproject.com</a>

ObjectMapper .NET	1.50.1312.0	<a href="http://www.objectmapper.net">http://www.objectmapper.net</a>
NHibernate	2.0.1.GA	<a href="http://www.nhibernate.org">http://www.nhibernate.org</a>
RapTier	1.4.2	<a href="http://www.sharppower.com">http://www.sharppower.com</a>
OPF.Net	0.9.01.599	<a href="http://www.littleguru.net/opf">http://www.littleguru.net/opf</a>
.NetTiers	2.3.0 Beta 2	<a href="http://www.nettiers.com">http://www.nettiers.com</a>
NDO	1.2	<a href="http://www.netdataobjects.com">http://www.netdataobjects.com</a>
Gentle.NET	1.2.9	<a href="http://sourceforge.net/projects/gopf">http://sourceforge.net/projects/gopf</a>
GenWise Studio	1.09	<a href="http://www.genwise.com">http://www.genwise.com</a>
OR.NET	5.0	<a href="http://www.ornetmapper.com">http://www.ornetmapper.com</a>
Castle Project	1.0 RC 3	<a href="http://www.castleproject.org">http://www.castleproject.org</a>
BizBlox	1.6.0.3	<a href="http://www.pixolot.com">http://www.pixolot.com</a>
SqlTac	0.106.57	<a href="http://www.sqltac.com">http://www.sqltac.com</a>
ARBT ORM Framework	1.1	<a href="http://www.arbtinformatica.com">http://www.arbtinformatica.com</a>

Bu araçlardan NHibernate, açık kaynak kodludur. Java dünyasında kullanılan Hibernate'in, .NET için geliştirilmiş halidir. Güncel sürümü 2.0.1.GA'dır [23]. Hibernate'in 2.1 sürümünde olan özellikleri içerir. Hibernate'in 3.1 sürümündeki özelliklerin dönüştürülerek NHibernate'e kazandırılması için NHibernate'in geliştiricileri çalışmalarına devam etmektedir. Hibernate'i java dünyasında destekleyen firma olan JBoss firması, açık kaynak kodlu NHibernate'i geliştiren ekibin başındaki kişiyi kendi bünyesine katarak, NHibernate'i de resmi olarak desteklemeye başlamıştır.

NHibernate'in kaynak kodlarının veritabanı, internet üzerinden erişilebilir durumdadır. İstenirse kaynak kodlar CVS<sup>6</sup> aracılığıyla indirilebilmektedir [24].

Tüm bu kolaylık ve avantajlar göz önüne alınarak, İTOKS Uygulamasında .NET teknolojisinden yararlanılmıştır. Uygulamanın ASP.NET C# programlama dili ile geliştirilmesi ve Nhibernate'in kullanılması uygun görülmüştür. Uygulamada oluşturulan nesnelere gösteren bir grafik, EK C'de ve uygulamadan seçilen örnek kodlar, EK D'de verilmiştir.

<sup>6</sup> CVS (Concurrent Versions System) : Sürüm kontrol sistemidir. Bir projenin gelişim tarihçesini otomatik olarak tutar [25].

Uygulama, İstanbul Üniversitesi Araştırma Fonu tarafından sağlanan dört çekirdek ve 4 GB Ram'e sahip Windows 2003 Enterprise Server işletim sistemi kurulu bir bilgisayar üzerinden sunulmuştur. Veri tabanı olarak yine Microsoft firmasının geliştirdiği Microsoft SQL Server 2005 kullanılmıştır.

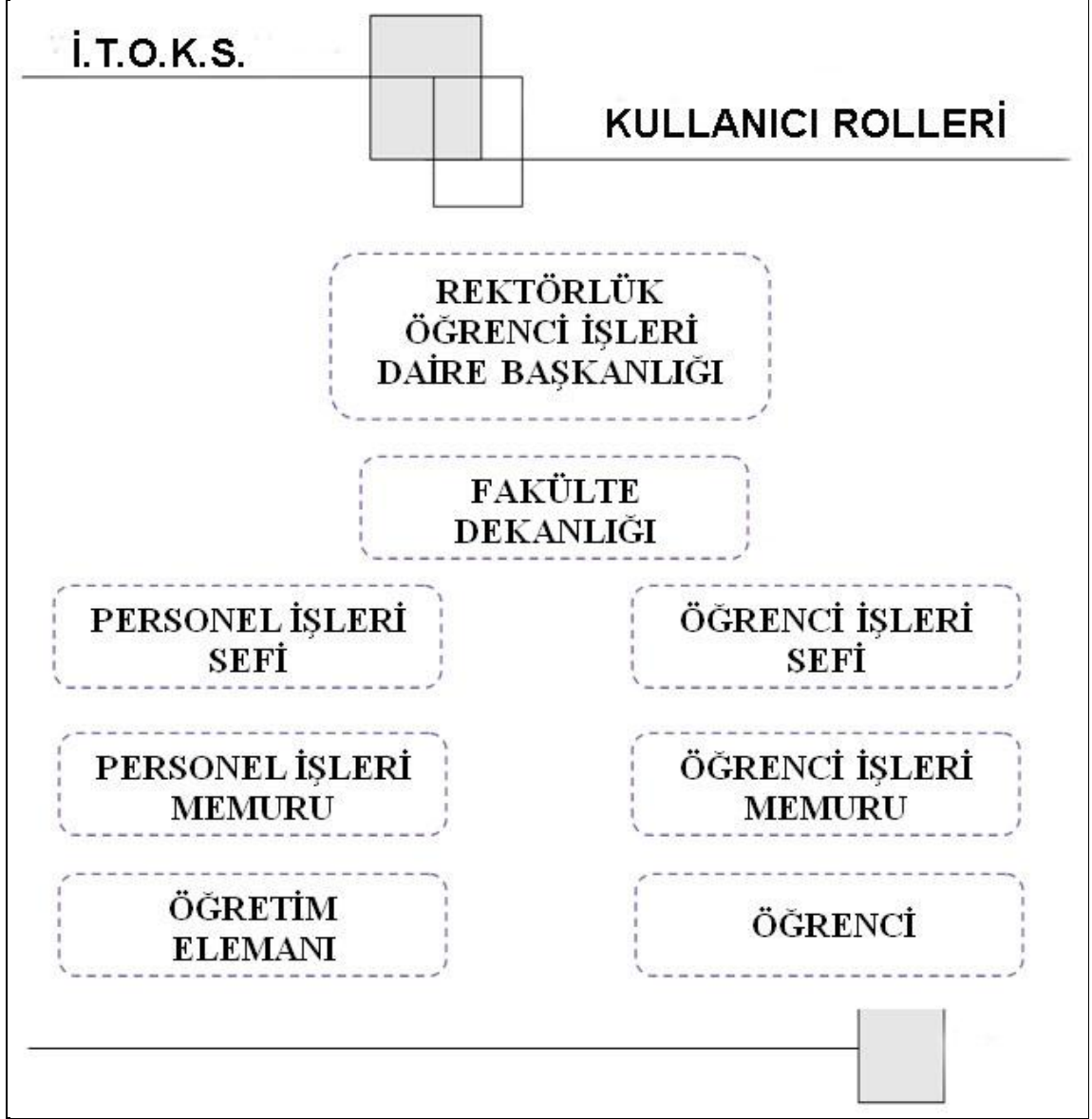
Uygulama sunucusuna, .NET teknolojilerini kullanabilmek için gerekli olan .NET Framework 3.5 versiyonu yüklenmiş ve uygulamanın sunulması için gerekli olan Internet Information Services (IIS) yüklenmiş ve ayarları yapılmıştır.

### **3.2.2. İTOKS Uygulaması**

İTOKS'da 8 tür kullanıcı rolü mevcut olup her kullanıcı rolünün farklı görev ve yetkileri vardır. Sistem dâhilindeki kullanıcı rolleri, genelde üniversitelerde işleyen hiyerarşi yapısına benzeştirilerek belirlenmiştir. Bununla amaçlanan her birimin sistemde kendi işini yapması ve böylece iş yükünün üniversite bünyesine yayılmasıdır. Genelde üniversitelerde işleyen yapı şu şekildedir; Rektörlük Öğrenci İşler Daire Başkanlığı tüm öğrencilerle ilgilidir ve bunu her fakültede oluşturulan Öğrenci İşleri Büroları ile birlikte gerçekleştirmektedir. Her fakültenin bir Öğrenci İşleri Birimi ve Personel İşleri (Özlük İşleri) birimi vardır. Ve bu birimlerin başında sırasıyla, Öğrenci İşleri Şefi ve Personel İşleri Şefi bulunmaktadır. Her fakültenin öğrencilerine, derslerine ve ders yönetmeliklerine ait bilgiler, Öğrenci İşleri Birimi tarafından işlenir, fakültenin akademik ve idari personel bilgileri ise Personel İşleri (Özlük İşleri) tarafından işlenmektedir.

İTOKS dâhilindeki bu kullanıcıların hiyerarşisi şu şekildedir;

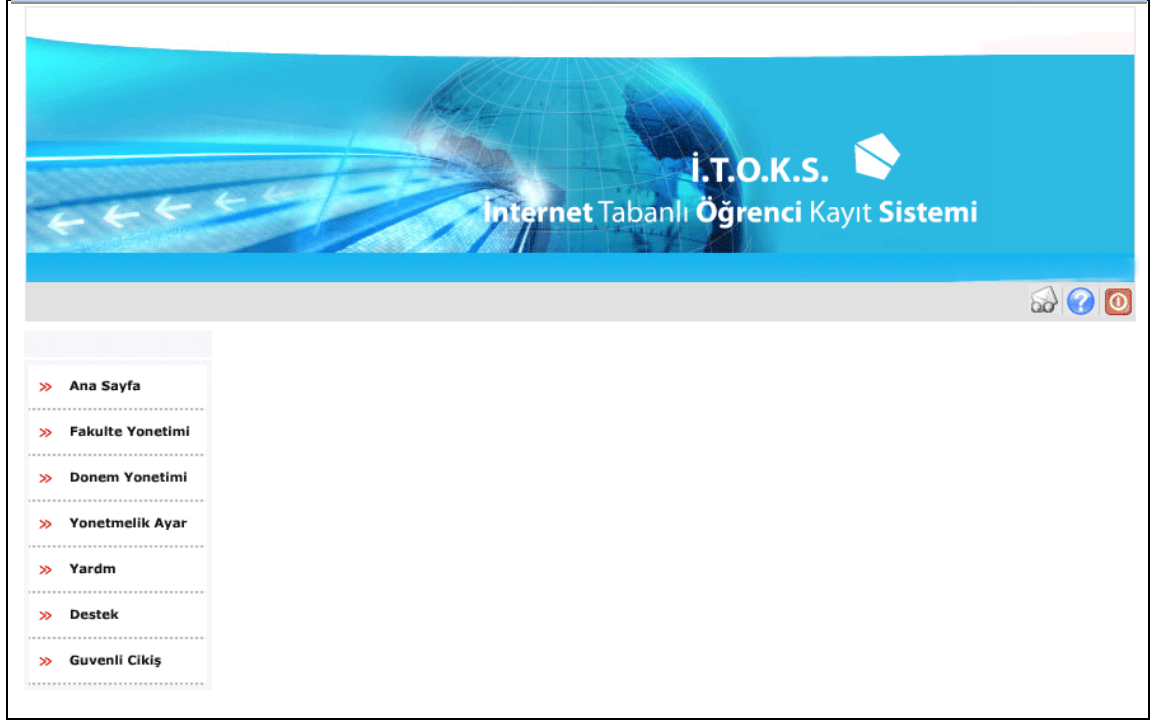
- Sistem Yöneticisi (Rektörlük Öğrenci İşleri Daire Başkanlığı Personeli),
- Fakülte (Fakülte Dekanlıkları),
- Öğrenci İşleri Bürosu Şefi,
- Öğrenci İşleri Bürosu Memuru,
- Personel İşleri Şefi,
- Personel İşleri Memuru,
- Öğretim Görevlisi
- Öğrenci



Şekil 3.17: İTOKS'daki Kullanıcı Roller

### 3.2.2.1 Sistem Yöneticisi (Rektörlük Öğrenci İşleri Daire Başkanlığı) Girişi

Bu giriş alanında, sistem yönetimi ile ilgili işlemlerin yapıldığı bir alana ulaşılmaktadır. Diğer kullanıcı tiplerinin bu alana ulaşma yetkisi bulunmamaktadır ve ayrıca sistem güvenliğine katkıda bulunmak için diğer kullanıcı tiplerinin girişleri ayrı bir kısımdan yapılmaktadır. Bu alanda altı ana başlık vardır: Fakülte Yönetimi, Dönem Yönetimi, Yönetmelik Ayar, Yardım, Destek ve Güvenli Çıkış.



Şekil 3.18: İTOKS - Sistem Yöneticisi (Rektörlük Öğrenci İşleri Daire Başkanlığı) Ekranı

**Fakülte Yönetimi** menüsü içerisinde; üniversite bünyesinde bulunan fakülteler tanımlanmakta ve bu fakültelerin sistemde ilgili modüllerine ulaşmaları için gerekli olan kullanıcı tanımlamaları yapılabilmektedir.

**Dönem Yönetimi** menüsü içerisinde; üniversitenin eğitim-öğretim dönemlerinin tanımlanması, düzenlenmesi ve aktif edilmesi gibi işlemler yapılabilmektedir.

**Yönetmelik Ayar** menüsü içerisinde; üniversitenin yönetmelik ayarları (öğrencinin ders alabilme şartları, geçme şartları v.b.) yapılabilmektedir.

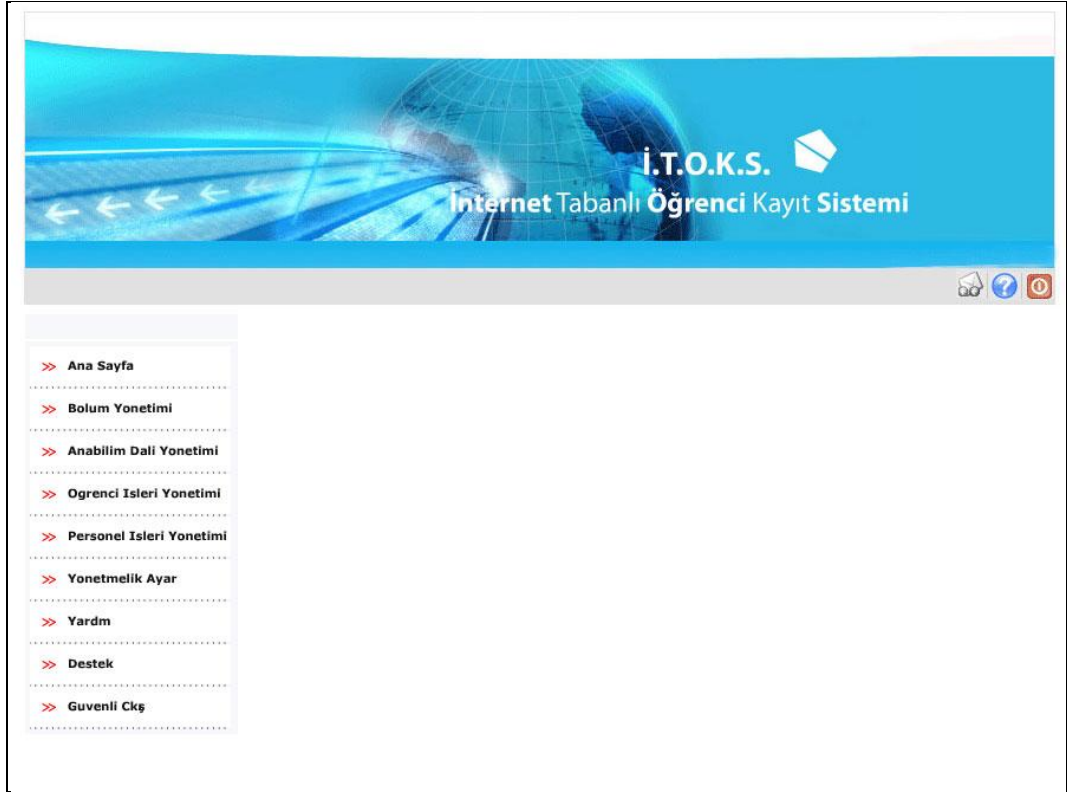
**Yardım** menüsü içerisinde; Sistem Yöneticisi rolüne sahip kullanıcının geliştirilen sistemi kolayca kullanabilmesi için gerekli dökümanlara ulaşabilmesi sağlanmaktadır.

**Destek** menüsü içerisinde; Sistem Yöneticisi rolüne sahip kullanıcı, sistemin geliştiricisiyle iletişim kurabilmektedir.

**Güvenli Çıkış** menüsü aracılığı ile Sistem Yöneticisi rolüne sahip kullanıcı güvenli bir şekilde sistemden çıkış yapabilmektedir.

### 3.2.2.2 Fakülte Girişi

Bu giriş alanında fakülte ile ilgili işlemlerin yapıldığı bir alana ulaşılmaktadır. Diğer kullanıcı tiplerinin bu alana ulaşma yetkisi bulunmamaktadır ve ayrıca sistem güvenliğine katkıda bulunmak için bu alana girişler ayrı bir kısımdan yapılmaktadır. Bu alanda altı ana başlık vardır: Öğrenci İşleri Yönetimi, Personel İşleri Yönetimi, Yönetmelik Ayar, Yardım, Destek ve Güvenli Çıkış.



Şekil 3.19: İTOKS - Fakülte Ekranı

**Öğrenci İşleri Yönetimi** menüsü içerisinde; fakülte bünyesinde bulunan Öğrenci İşleri Bürosu tanımlanmakta ve Öğrenci İşleri Bürosu Şefi'nin sistemde ilgili modüllerine ulaşması için gerekli olan kullanıcı tanımlamaları yapılabilmektedir.

**Personel İşleri Yönetimi** menüsü içerisinde; fakülte bünyesinde bulunan Personel İşleri tanımlanmakta ve Personel İşleri Şefi'nin sistemde ilgili modüllerine ulaşması için gerekli olan kullanıcı tanımlamaları yapılabilmektedir.

**Bölüm Yönetimi** menüsü içerisinde; fakülte bünyesinde bulunan Bölümler tanımlanmaktadır.

**Anabilim Dalı Yönetimi** menüsü içerisinde; fakülte bünyesinde bulunan Anabilim Dalları tanımlanmaktadır.

**Yönetmelik Ayar** menüsü içerisinde; fakültenin yönetmelik ayarları (öğrencinin ders alabilme şartları, geçme şartları v.b. gibi) yapılabilmektedir.

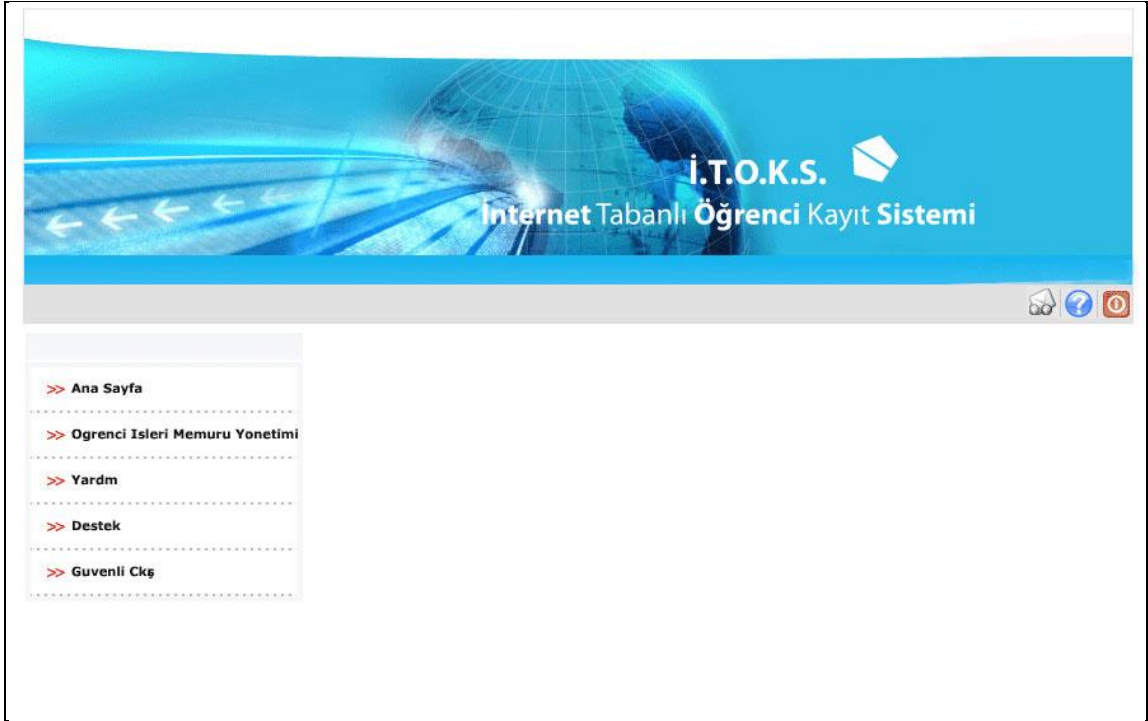
**Yardım** menüsü içerisinde; Fakülte rolüne sahip kullanıcının geliştirilen sistemi kolayca kullanabilmesi için gerekli dökümanlara ulaşabilmesi sağlanmaktadır.

**Destek** menüsü içerisinde; Fakülte rolüne sahip kullanıcı, sistemin geliştiricisiyle iletişim kurabilmektedir.

**Güvenli Çıkış** menüsü aracılığı ile Fakülte rolüne sahip kullanıcı güvenli bir şekilde sistemden çıkış yapabilmektedir.

### 3.2.2.3 Öğrenci İşleri Bürosu Şefi Girişi

Bu giriş alanında ilgili fakültenin öğrenci işleri bürosu ile ilgili işlemlerin yapıldığı bir alana ulaşılmaktadır. Diğer kullanıcı tiplerinin bu alana ulaşma yetkisi bulunmamaktadır ve ayrıca sistem güvenliğine katkıda bulunmak için bu alana girişler ayrı bir kısımdan yapılmaktadır. Bu alanda dört ana başlık vardır: Öğrenci İşleri Memur Yönetimi, Yardım, Destek ve Güvenli Çıkış.



Şekil 3.20: İTOKS – Öğrenci İşleri Bürosu Şefi Ekranı

**Öğrenci İşleri Memuru Yönetimi** menüsü içerisinde; fakülte bünyesinde bulunan Öğrenci İşleri Bürosu Memurları tanımlanmakta ve Öğrenci İşleri Bürosu Memurları'nın

sistemde ilgili modüllerine ulaşması için gerekli olan kullanıcı tanımlamaları yapılabilmektedir.

**Yardım** menüsü içerisinde; Öğrenci İşleri Bürosu Şefi rolüne sahip kullanıcının geliştirilen sistemi kolayca kullanabilmesi için gerekli dökümanlara ulaşabilmesi sağlanmaktadır.

**Destek** menüsü içerisinde; Öğrenci İşleri Bürosu Şefi rolüne sahip kullanıcı, sistemin geliştiricisiyle iletişim kurabilmektedir.

**Güvenli Çıkış** menüsü aracılığı ile Öğrenci İşleri Bürosu Şefi rolüne sahip kullanıcı güvenli bir şekilde sistemden çıkış yapabilmektedir.

#### 3.2.2.4 Öğrenci İşleri Bürosu Memuru Girişi

Bu giriş alanında öğrenci işleri bürosu memuru ilgili fakültenin ders, öğrenci, danışman atama gibi işlemlerin yapıldığı bir alana ulaşılmaktadır. Diğer kullanıcı tiplerinin bu alana ulaşma yetkisi bulunmamaktadır ve ayrıca sistem güvenliğine katkıda bulunmak için bu alana girişler ayrı bir kısımdan yapılmaktadır. Bu alanda altı ana başlık vardır: Ders Yönetimi, Öğrenci Yönetimi, Danışman Yönetimi, Yardım, Destek ve Güvenli Çıkış.



Şekil 3.21: İTOKS – Öğrenci İşleri Bürosu Memuru Ekranı

**Ders Yönetimi** menüsü içerisinde; fakülte bünyesinde bulunan dersler ve sistem yöneticisi tarafından açılan dönemlerde verilecek olan dersler tanımlanmaktadır.

**Öğrenci Yönetimi** menüsü içerisinde; fakülte öğrencileri tanımlanmakta ve öğrencilerin sistemde ilgili modüllerine ulaşmaları için gerekli olan kullanıcı tanımlamaları yapılabilmektedir.

**Danışman Yönetimi** menüsü içerisinde; fakülte öğrencilerine danışmanlık yapacak öğretim görevlilerinin tanımlanması yapılabilmektedir.

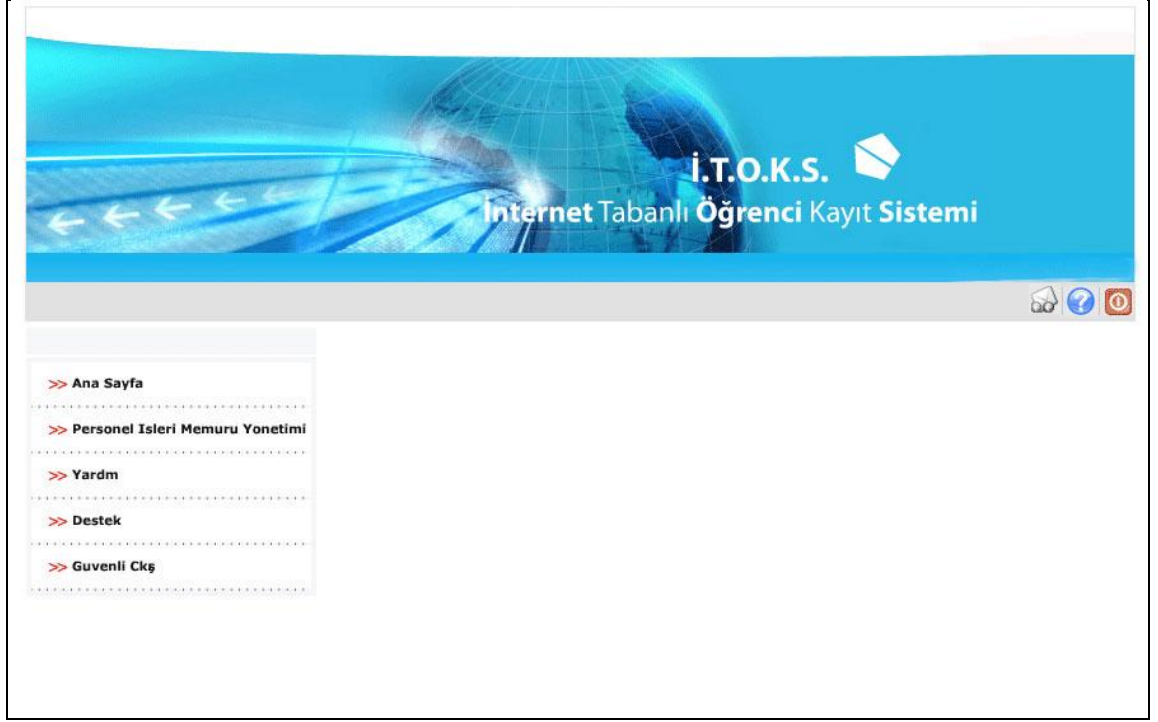
**Yardım** menüsü içerisinde; Öğrenci İşleri Bürosu Memuru rolüne sahip kullanıcının geliştirilen sistemi kolayca kullanabilmesi için gerekli dökümanlara ulaşabilmesi sağlanmaktadır.

**Destek** menüsü içerisinde; Öğrenci İşleri Bürosu Memuru rolüne sahip kullanıcı, sistemin geliştiricisiyle iletişim kurabilmektedir.

**Güvenli Çıkış** menüsü aracılığı ile Öğrenci İşleri Bürosu Memuru rolüne sahip kullanıcı güvenli bir şekilde sistemden çıkış yapabilmektedir.

#### *3.2.2.5 Personel İşleri Şefi Girişi*

Bu giriş alanında ilgili fakültenin personel işleri ile ilgili işlemlerin yapıldığı bir alana ulaşılmaktadır. Diğer kullanıcı tiplerinin bu alana ulaşma yetkisi bulunmamaktadır ve ayrıca sistem güvenliğine katkıda bulunmak için bu alana girişler ayrı bir kısımdan yapılmaktadır. Bu alanda dört ana başlık vardır: Personel İşleri Memur Yönetimi, Yardım, Destek ve Güvenli Çıkış.



Şekil 3.22: İTOKS – Personel İşleri Şefi Ekranı

**Personel İşleri Memuru Yönetimi** menüsü içerisinde; fakülte bünyesinde bulunan Personel İşleri Memurları tanımlanmakta ve Personel İşleri Memurları'nın sistemde ilgili modüllerine ulaşması için gerekli olan kullanıcı tanımlamaları yapılabilmektedir.

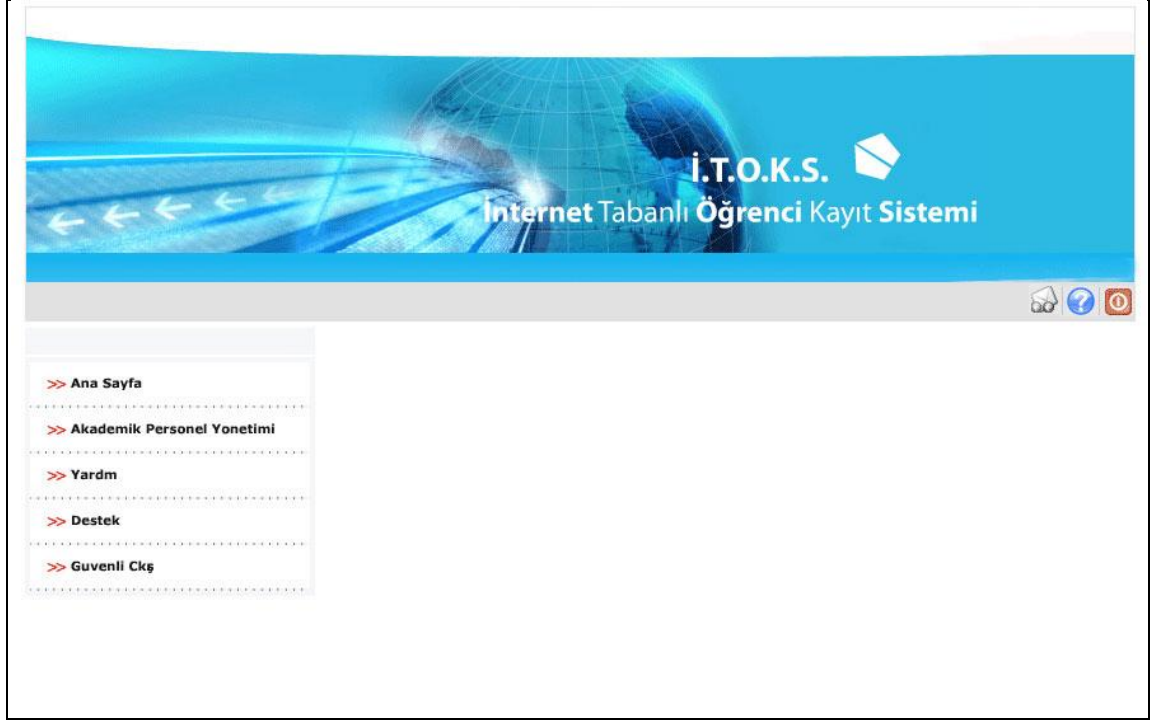
**Yardım** menüsü içerisinde; Personel İşleri Şefi rolüne sahip kullanıcının geliştirilen sistemi kolayca kullanabilmesi için gerekli dökümanlara ulaşabilmesi sağlanmaktadır.

**Destek** menüsü içerisinde; Personel İşleri Şefi rolüne sahip kullanıcı, sistemin geliştiricisiyle iletişim kurabilmektedir.

**Güvenli Çıkış** menüsü aracılığı ile Personel İşleri Şefi rolüne sahip kullanıcı güvenli bir şekilde sistemden çıkış yapabilmektedir.

### 3.2.2.6 Personel İşleri Memuru Girişi

Bu giriş alanında ilgili fakültenin personel işleri ile ilgili işlemlerin yapıldığı bir alana ulaşılmaktadır. Diğer kullanıcı tiplerinin bu alana ulaşma yetkisi bulunmamaktadır ve ayrıca sistem güvenliğine katkıda bulunmak için bu alana girişler ayrı bir kısımdan yapılmaktadır. Bu alanda dört ana başlık vardır: Öğretim Görevlisi Yönetimi, Yardım, Destek ve Güvenli Çıkış.



Şekil 3.23: İTOKS – Personel İşleri Memuru Ekranı

**Akademik Personel Yönetimi** menüsü içerisinde; fakülte bünyesinde bulunan Öğretim Görevlileri tanımlanmakta ve Öğretim Görevlileri'nin sistemde ilgili modüllerine ulaşması için gerekli olan kullanıcı tanımlamaları yapılabilmektedir.

**Yardı** menüsü içerisinde; Personel İşleri Memuru rolüne sahip kullanıcının geliştirilen sistemi kolayca kullanabilmesi için gerekli dökümanlara ulaşabilmesi sağlanmaktadır.

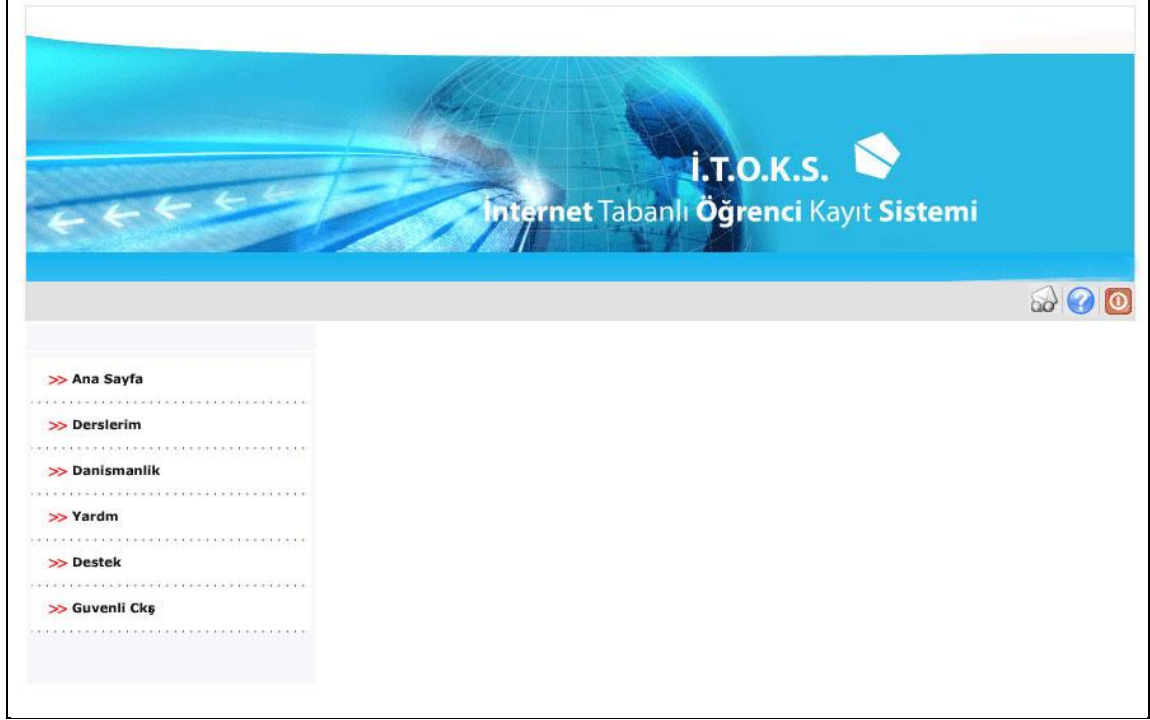
**Destek** menüsü içerisinde; Personel İşleri Memuru rolüne sahip kullanıcı, sistemin geliştiricisiyle iletişim kurabilmektedir.

**Güvenli Çıkış** menüsü aracılığı ile Personel İşleri Memuru rolüne sahip kullanıcı güvenli bir şekilde sistemden çıkış yapabilmektedir.

### 3.2.2.7 Öğretim Görevlisi Girişi

Bu giriş alanında öğretim görevlisi danışmanlığını yaptığı öğrencilerin ders seçim işlemlerini görüntülemekte, onay verebilmekte ve imzalamak üzere ders kayıt formunun çıktısını alabilmektedir. Diğer kullanıcı tiplerinin bu alana ulaşma yetkisi bulunmamaktadır ve ayrıca sistem güvenliğine katkıda bulunmak için bu alana girişler

ayrı bir kısımdan yapılmaktadır. Bu alanda dört ana başlık vardır: Danışmanlık, Yardım, Destek ve Güvenli Çıkış.



Şekil 3.24: İTOKS – Öğretim Görevlisi Ekranı

**Derslerim** menüsü içerisinde; öğretim görevlisi mevcut dönemde vermekte olduğu derslerin harf notu girişlerini yapabilmekte, öğrenci listelerini alabilmektedir.

2007-2008 Bahar 120212-Bilgisayar II Öğrenci Listesi				
Öğrenci No	Adı	Soyadı	AnaBilim Dalı	
3212070002	DURALI	KARAKAYA	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070004	NIĞMET ÇİĞDEM	BERBER	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070008	MÜRSEL	TÜRK	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070013	FATİH	YAZICI	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070014	FUAT	TÜZ	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070017	ELİF	İĞİRCİK	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070020	HASAN HÜSEYİN	ÇOK	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070021	ŞEYDA	DİNÇ	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070022	İSMAIL SEFA	KUBLAY	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070024	FATİMANUR	BAĞDATLI	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070024	FATİMANUR	BAĞDATLI	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070027	HAKAN	KARA	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070028	VAHİT REFİK	KURTAY	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070032	BAHADİR	YILDIZ	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070034	ÖZGE	SEZEN	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070036	DENİZ	CAN	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070038	DUYGU	SAYAR	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	
3212070041	ŞEVKET	İLGÜN	Sınıf Öğretmenliği Eğitimi Anabilim Dalı	

Şekil 3.25: İTOKS – Öğretim Görevlisi'nin Dersine Ait Sınıf Listesi Ekranı

**Danışmanlık** menüsü içerisinde; öğretim görevlisinin danışmanlığına atandığı öğrencilerin listelerini alabilmekte, ders kayıt formlarını görüntüleyerek inceleyebilmekte ve onaylayabilmektedir.

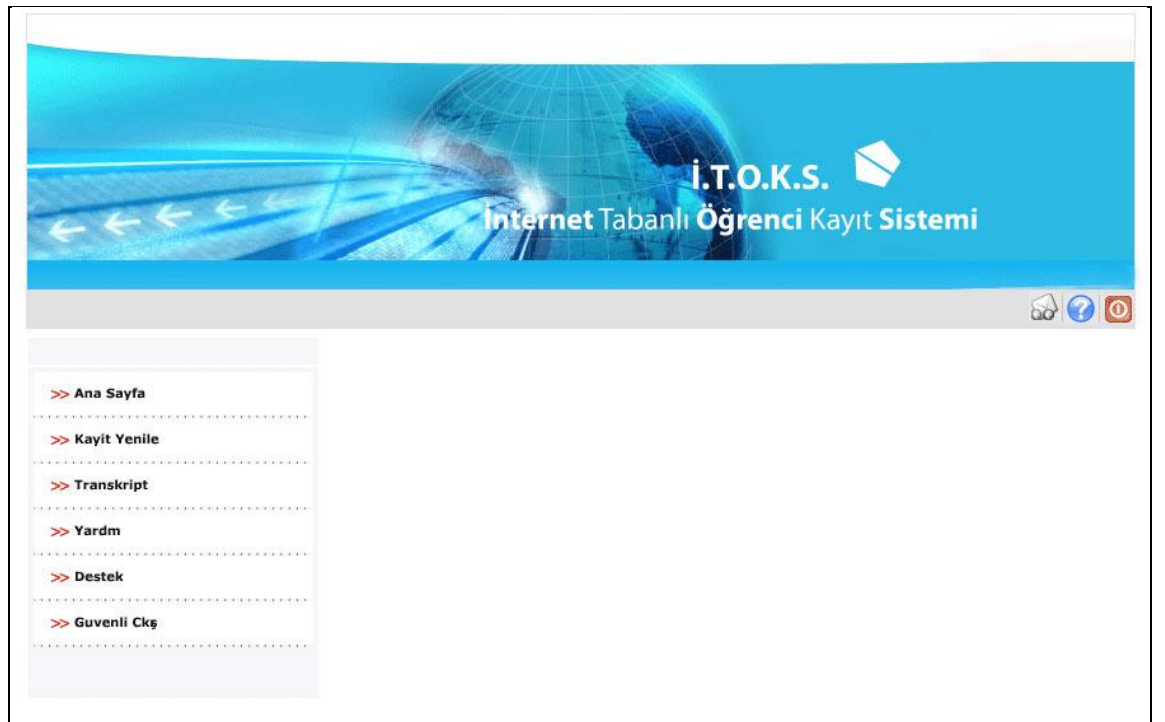
**Yardım** menüsü içerisinde; Öğretim Görevlisi rolüne sahip kullanıcının geliştirilen sistemi kolayca kullanabilmesi için gerekli dökümanlara ulaşabilmesi sağlanmaktadır.

**Destek** menüsü içerisinde; Öğretim Görevlisi rolüne sahip kullanıcı, sistemin geliştiricisiyle iletişim kurabilmektedir.

**Güvenli Çıkış** menüsü aracılığı ile Öğretim Görevlisi rolüne sahip kullanıcı güvenli bir şekilde sistemden çıkış yapabilmektedir.

### 3.2.2.8 Öğrenci Girişi

Bu giriş alanında öğrenci mevcut dönemde açılan dersleri, transkriptini ve durumunu görebilmekte, mevcut dönem için ders seçip kayıt yenileyebilmektedir. Diğer kullanıcı tiplerinin bu alana ulaşma yetkisi bulunmamaktadır ve ayrıca sistem güvenliğine katkıda bulunmak için bu alana girişler ayrı bir kısımdan yapılmaktadır. Bu alanda beş ana başlık vardır: Kayıt Yenile, Transkript, Yardım, Destek ve Güvenli Çıkış.



Şekil 3.26: İTOKS - Öğrenci Ekranı

**Kayıt Yenile** menüsü içerisinde; öğrenci mevcut dönemde açılmış olan dersleri görebilmekte ve seçmesi mümkün olan dersleri seçebilmektedir. İTOKS, bu süreçte öğrenciye danışmanlık etmektedir. Öğrencinin başarısız olduğu dersler varsa önce onları almalarını zorunlu tutar, sorumlu geçtiği dersleri de almasını önerir fakat not ortalaması şartına takılmamışsa zorunlu tutmaz. Aynı zamanda dersleri o ana kadar seçmiş kişi sayısını, kalan kontenjan sayısını, dersin kredisini ve dersi veren öğretim üyesi hakkında bilgi sahibi olabilir. Derslerini seçtikten sonra Kaydet butonuna tıklayarak, ders kayıt işlemini tamamlar ve yazıcıdan çıktısını alıp imzalanmak üzere seçtiği derslere dair bir form oluşturulur. Bu formu o anda yazdırabileceği gibi, daha sonra onay için danışmanına gittiğinde de yazdırabilir. Öğrencinin bu ders kayıt ekranını görüntüleyebilmesi ve ders seçim işlemini yapabilmesi için yatırması gerekli olan harç ücretini, ilgili bankaya yatırmış olmalıdır. Sistem bu kontrolü bankanın sistemi ile iletişime geçerek yapar ve ödeme yapılmamışsa, önce ödeme yapılması için uyarı verir.

Ders Kaydı			
Adı Soyadı	FERHAT SERBEST	Sınıfı	Bir
Genel Ortalama	3,06	Okuduğu Dönem	2
Sınamalı Öğrenci	Label	Max Alabileceği Dönem	2
Kayıt Yapabilir	KayıtYenileyebilir	Danışmanı	Araş.Gör. Burak Şişman
Kredi Toplamı	18	Danışman Onay Tarihi	Label
		Alabileceği Kredi	Label
Ders Dönemi :	2007-2008 Bahar		
<div style="border: 1px solid black; padding: 5px;">           140204-Atatürk İnkeleri ve İnkılap Tarihi II            140206-Yabancı Dil II            140209-Bilgisayar II         </div>			
Kaldır			
Seçilen derslerin kredi toplamı : 8			
<input type="checkbox"/>	140202 - Geometri	3	34/100 Araş.Gör. Burak Şişman
<input type="checkbox"/>	140204 - Atatürk İnkeleri ve İnkılap Tarihi II	2	Araş.Gör. Burak Şişman
<input type="checkbox"/>	140205 - Türkçe II: Sözlü Anlatım	2	Araş.Gör. Burak Şişman
<input type="checkbox"/>	140206 - Yabancı Dil II	3	Araş.Gör. Burak Şişman
<input type="checkbox"/>	140208 - Soyut Matematik	3	Araş.Gör. Burak Şişman
<input type="checkbox"/>	140209 - Bilgisayar II	3	Araş.Gör. Burak Şişman
<input type="checkbox"/>	140210 - Eğitim Psikolojisi	3	Araş.Gör. Burak Şişman
<input type="button" value="Vazgeç"/> <input type="button" value="Kaydet"/>			

Şekil 3.27: İTOKS - Öğrenci Ders Kayıt Ekranı

T.C. İSTANBUL ÜNİVERSİTESİ HASAN ALİ YÜCEL EĞİTİM FAKÜLTESİ DERS KAYIT FORMU			T.C. İSTANBUL ÜNİVERSİTESİ HASAN ALİ YÜCEL EĞİTİM FAKÜLTESİ DERS KAYIT FORMU			T.C. İSTANBUL ÜNİVERSİTESİ HASAN ALİ YÜCEL EĞİTİM FAKÜLTESİ DERS KAYIT FORMU																																
Öğrenci No :	3214070001		Öğrenci No :	3214070001		Öğrenci No :	3214070001																															
Adı Soyadı :	Ferhat Serbest		Adı Soyadı :	Ferhat Serbest		Adı Soyadı :	Ferhat Serbest																															
Bölüm :	İlköğretim Bölümü		Bölüm :	İlköğretim Bölümü		Bölüm :	İlköğretim Bölümü																															
Anabilim Dalı :	Matematik Eğitimi Anabilim Dalı		Anabilim Dalı :	Matematik Eğitimi Anabilim Dalı		Anabilim Dalı :	Matematik Eğitimi Anabilim Dalı																															
Dönemi :	2007-2008 Bahar Dönemi		Dönemi :	2007-2008 Bahar Dönemi		Dönemi :	2007-2008 Bahar Dönemi																															
Kayıt Tarihi :	10/04/2008		Kayıt Tarihi :	10/04/2008		Kayıt Tarihi :	10/04/2008																															
<table border="1"> <thead> <tr> <th>Dersler</th> <th>Kredi</th> </tr> </thead> <tbody> <tr> <td>123459 Türk Dili II</td> <td>3</td> </tr> <tr> <td>123458 Genel Matematik II</td> <td>3</td> </tr> <tr> <td>123456 Bilgisayar II</td> <td>3</td> </tr> <tr> <td><b>Kredi Toplamı :</b></td> <td><b>9</b></td> </tr> </tbody> </table>			Dersler	Kredi	123459 Türk Dili II	3	123458 Genel Matematik II	3	123456 Bilgisayar II	3	<b>Kredi Toplamı :</b>	<b>9</b>	<table border="1"> <thead> <tr> <th>Dersler</th> <th>Kredi</th> </tr> </thead> <tbody> <tr> <td>123459 Türk Dili II</td> <td>3</td> </tr> <tr> <td>123458 Genel Matematik II</td> <td>3</td> </tr> <tr> <td>123456 Bilgisayar II</td> <td>3</td> </tr> <tr> <td><b>Kredi Toplamı :</b></td> <td><b>9</b></td> </tr> </tbody> </table>			Dersler	Kredi	123459 Türk Dili II	3	123458 Genel Matematik II	3	123456 Bilgisayar II	3	<b>Kredi Toplamı :</b>	<b>9</b>	<table border="1"> <thead> <tr> <th>Dersler</th> <th>Kredi</th> </tr> </thead> <tbody> <tr> <td>123459 Türk Dili II</td> <td>3</td> </tr> <tr> <td>123458 Genel Matematik II</td> <td>3</td> </tr> <tr> <td>123456 Bilgisayar II</td> <td>3</td> </tr> <tr> <td><b>Kredi Toplamı :</b></td> <td><b>9</b></td> </tr> </tbody> </table>			Dersler	Kredi	123459 Türk Dili II	3	123458 Genel Matematik II	3	123456 Bilgisayar II	3	<b>Kredi Toplamı :</b>	<b>9</b>
Dersler	Kredi																																					
123459 Türk Dili II	3																																					
123458 Genel Matematik II	3																																					
123456 Bilgisayar II	3																																					
<b>Kredi Toplamı :</b>	<b>9</b>																																					
Dersler	Kredi																																					
123459 Türk Dili II	3																																					
123458 Genel Matematik II	3																																					
123456 Bilgisayar II	3																																					
<b>Kredi Toplamı :</b>	<b>9</b>																																					
Dersler	Kredi																																					
123459 Türk Dili II	3																																					
123458 Genel Matematik II	3																																					
123456 Bilgisayar II	3																																					
<b>Kredi Toplamı :</b>	<b>9</b>																																					
Öğrencinin,	Danışmanın,	Memurun,	Öğrencinin,	Danışmanın,	Memurun,	Öğrencinin,	Danışmanın,	Memurun,																														
Ferhat Serbest	Araş. Gör. Burak Şişman		Ferhat Serbest	Araş. Gör. Burak Şişman		Ferhat Serbest	Araş. Gör. Burak Şişman																															

Şekil 3.28: İTOKS - Öğrenci Ders Kayıt Formu

**Transkript** menüsü içerisinde; öğrenci geçmişte almış olduğu tüm dersleri ve notlarını gösteren transkriptini görüntüleyebilmektedir.

3214070001 lu FERHAT SERBEST için dönem notları

Dönemler	Ders Adı	Kodu	Kredi	Notu
2007-2008 Güz	Atatürk İnkılabı ve İnkılap Tarihi I	140104	2	AA
	Türkçe I : Yazılı Anlatım	140105	2	AA
	Yabancı Dil I	140106	3	M
	Genel Matematik	140108	5	CB
	Bilgisayar I	140109	3	AA
	Eğitim Bilimine Giriş	140110	3	AA
Dönem Kredi Toplamı : 18			Dönem Ort. : 3,08	

Şekil 3.29 İTOKS - Öğrenci Transkript Ekranı

**Yardım** menüsü içerisinde; Öğrenci rolüne sahip kullanıcının geliştirilen sistemi kolayca kullanabilmesi için gerekli dökümanlara ulaşabilmesi sağlanmaktadır.

**Destek** menüsü içerisinde; Öğrenci rolüne sahip kullanıcı, sistemin geliştiricisiyle iletişim kurabilmektedir.

**Güvenli Çıkış** menüsü aracılığı ile Öğrenci rolüne sahip kullanıcı güvenli bir şekilde sistemden çıkış yapabilmektedir.

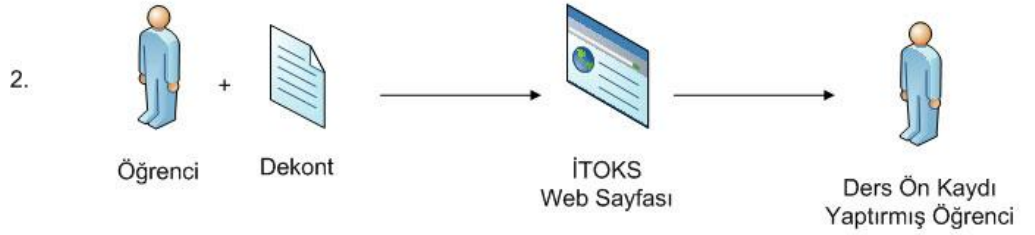
İTOKS sistemi dahilinde öğrencilerin dönem başlarında ders seçim işlemleri için izledikleri yol şu şekildedir;

1. Öğrenci dönem kaydı yaptırabilmek için yatırması gerekli olan harç miktarını ilgili banka şubesine yatırır ve dekontunu alır.



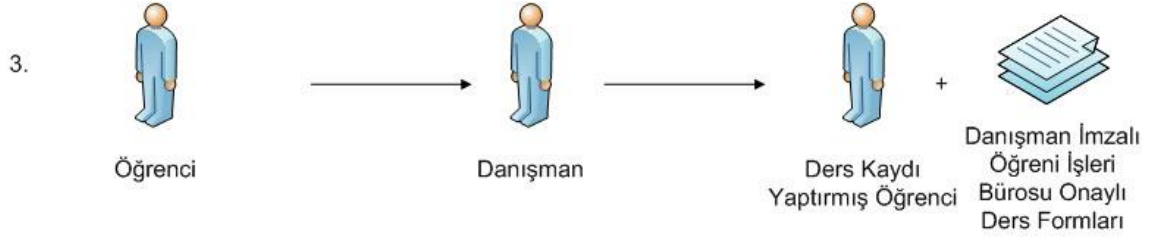
Şekil 3.30: İTOKS ile ders kayıt aşamalarının incelenmesi şeması 1. adım

2. Öğrenci internet bağlantısı olan herhangi bir yerdeki bir bilgisayar aracılığı ile İTOKS'a giriş bilgileri ile giriş yapar, durumunu görüntüler, sistem aracılığıyla açılmış olan dersleri görüntüler ve alabileceği dersleri seçerek ön kaydını tamamlar.



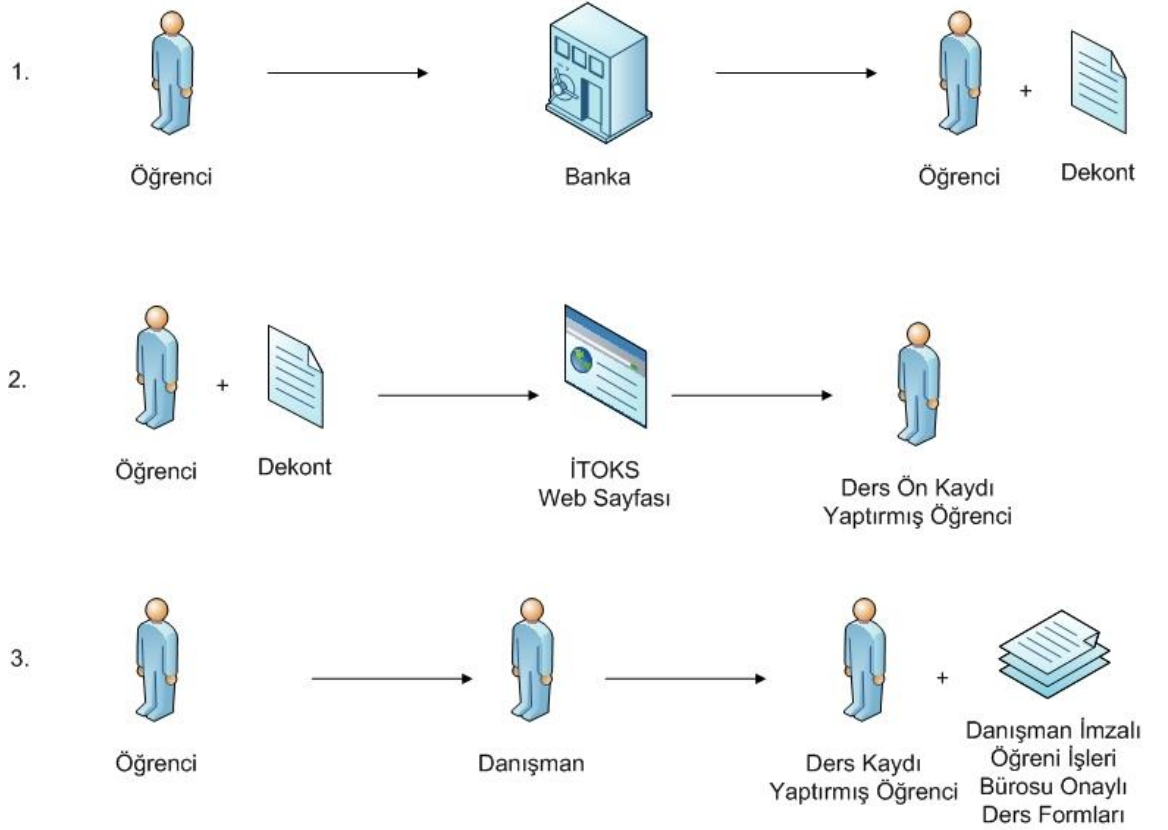
Şekil 3.31: İTOKS ile ders kayıt aşamalarının incelenmesi şeması 2. adım

3. Öğrenci danışmanına gider, seçtiği derslerin dökümünü içeren imzalı ve onaylı ders kayıt formunu alır ve ders kaydını tamamlamış olur.



Şekil 3.32: İTOKS ile ders kayıt aşamalarının incelenmesi şeması 3. adım

### İTOKS KAYIT AŞAMALARI



Şekil 3.33: İTOKS ile ders kayıt aşamalarının incelenmesi şeması tüm adımlar

### 3.2.3. İTOKS'un Uygulanması

Öncelikle, 2007-2008 bahar yarıyılında Hasan Ali Yücel Eğitim Fakültesi İlköğretim Bölümü Matematik Eğitimi Anabilim Dalı, Sınıf Öğretmenliği Eğitimi Anabilim Dalı, Fen Bilgisi Eğitimi Anabilim Dalı, Din Kültürü ve Ahlak Bilgisi Eğitimi Bölümü Din Kültürü ve Ahlak Bilgisi Eğitimi Anabilim Dalı ve Türkçe Eğitimi Bölümü Türkçe Eğitimi Anabilim Dalı öğrencilerine ait bilgiler ve güz döneminde aldıkları dersler ve notları İTOKS'a işlenmiştir.

Daha sonra, bu öğrenciler bilgisayar laboratuvarlarına alınarak, şifre bilgileri sağlandıktan sonra sistemi test etmeleri istenmiştir. Ve test etme işlemi bitiren öğrencilere sistem hakkındaki düşüncelerini öğrenebilmek amacıyla anket uygulanmıştır. Uygulanan anket Ek B'de verilmiştir.

#### 3.2.3.1. Evren ve Örneklem

Bu çalışmanın evreni, İstanbul Üniversitesi Hasan Ali Yücel Eğitim Fakültesi'nde lisans düzeyinde yüksek öğretimlerine devam eden öğrencileri kapsamaktadır. Çalışmanın örnekleme, 2007-2008 eğitim öğretim yılı birinci döneminde İstanbul Üniversitesi Hasan Ali Yücel Eğitim Fakültesi İlköğretim Bölümü Matematik Eğitimi Anabilim Dalı, Sınıf Öğretmenliği Eğitimi Anabilim Dalı, Fen Bilgisi Eğitimi Anabilim Dalı, Din Kültürü ve Ahlak Bilgisi Eğitimi Bölümü Din Kültürü ve Ahlak Bilgisi Eğitimi Anabilim Dalı ve Türkçe Eğitimi Bölümü Türkçe Eğitimi Anabilim Dalı'nda yükseköğretime lisans düzeyinde devam etmekte olan, birinci sınıf öğrencisi toplam 157 öğrenciden oluşmaktadır.

#### 3.2.3.2. Veriler ve Toplanması

Bu araştırma kapsamında, çalışmaya katılan öğrencilerden, deneme sonunda İnternet Tabanlı Ders Kayıt Sistemi'nin kullanılabilirliği hakkında görüşleri toplanmıştır.

#### 3.2.3.3. Uygulama

Veri toplama araçları, araştırmacı tarafından bir araya getirildikten sonra, oluşturulan bir uygulama programı çerçevesinde belirlenen zamanlarda, araştırmaya katılan bölümlerde öğrenimine devam eden öğrencilere bilgisayar laboratuvarında hazırlanan sistem kullanılarak, katılımcıların internet üzerinden sisteme dair anketi doldurmaları sağlanmıştır.

### 3.2.3.4. Verilerin Analizi

Araştırma kapsamında elde edilen verilerin istatistiksel çözümlemesinde, betimsel analiz teknikleri (yüzde ve frekans) kullanılmıştır. İstatistiksel çözümler, SPSS<sup>7</sup> 13.0 istatistiksel çözümler programı aracılığıyla yapılmıştır.

### 3.2.3.5. Bulgular

Çalışmanın bu bölümünde, katılımcılardan elde edilen verilerin araştırmanın amaçları çerçevesinde incelenmesi sonucunda ortaya konan bulgular sunulmaktadır.

Yapılan araştırmada İnternet Tabanlı Ders Kayıt Sistemi hakkında %95,5 oranında bilgi sahibi olduğu görülmüştür. Öğrencilerin %7,6'sı sistemi kullanmaktan hoşlanmadığını, %92,4'ü ise sistemi kullanmaktan hoşlandığını bildirmiştir. Bunun yanı sıra öğrencilerin %98,7'si bu Ders Kayıt Sistemi'nin uygulanmaya başlanması durumunda, okula sadece ders kaydı için geleceği için memnun olduğunu belirtmiştir.

Tablo 3.3: “İnternet Tabanlı Ders Kayıt Sistemi, belirtilen kayıt tarihleri arasında istediğim gün ve saatte kayıt yapabilmemi sağlar.”

	Frekans	Yüzde	Birikimli Yüzde
Evet	155	98,7	98,7
Hayır	2	1,3	100,0
Toplam	157	100,0	

Tablo 3.3'de görüldüğü üzere, öğrencilerin %98,7'si, İnternet Tabanlı Ders Kayıt Sistemi'nin, belirtilen kayıt tarihleri arasında istenilen gün ve saatte kayıt yapabilmemesini sağladığı düşüncesindedir.

Tablo 3.4: “İnternet Tabanlı Ders Kayıt Sistemi, danışmanıma ulaşamama gibi olumsuz durumları ortadan kaldırır.”

	Frekans	Yüzde	Birikimli Yüzde
Evet	150	95,5	95,5
Hayır	7	4,5	100,0
Toplam	157	100,0	

<sup>7</sup> SPSS: İngilizce açılımıyla, Statistical Package for the Social Sciences, istatistiksel analize yönelik bir bilgisayar yazılımıdır [26].

Tablo 3.4’de görüldüğü üzere, öğrencilerin %95,5’i, İnternet Tabanlı Ders Kayıt Sisteminin, danışmana ulaşamama gibi olumsuz durumları ortadan kaldıracacağı yönünde fikir belirtmiştir. Ayrıca öğrencilerin %99,4’ü, bu sistemi kullanmakla kâğıt israfının da ortadan kalkacağı kanısındadır.

Tablo 3.5: “İnternet Tabanlı Ders Kayıt Sistemi’ni kullanmanın kolay olduğu kanısındayım.”

	Frekans	Yüzde	Birikimli Yüzde
Evet	156	99,4	99,4
Hayır	1	,6	100,0
Toplam	157	100,0	

Tablo 3.5’de gösterildiği üzere, öğrencilerin %99,4’ü sistemi kullanmanın kolay olduğu kanısında iken % 0,6’sı kolay olmadığı kanısındadır.

Tablo 3.6: “İnternet Tabanlı ders kayıt klasik ders kaydına göre daha az zamanımı aldı.”

	Frekans	Yüzde	Birikimli Yüzde
Evet	155	98,7	98,7
Hayır	2	1,3	100,0
Toplam	157	100,0	

Tablo 3.6’da görüldüğü üzere, öğrencilerin %98,7’si İnternet Tabanlı ders kaydını klasik ders kaydına göre kıyasladıklarında, İnternet Tabanlı ders kaydının daha az zamanlarını aldığını söylemişlerdir. Bunun yanı sıra öğrencilerin %98,1’i İnternet Tabanlı Ders Kayıt Sistemi’ni kullanmanın kendilerine zaman kazandırdığını düşünmekte iken, %1,9 oranında öğrenci bunun tersini düşünmektedir.

Tablo 3.7: “İnternet Tabanlı Ders Kayıt Sistemi sayesinde transkriptimi istediğim zamanda alabiliyorum.”

	Frekans	Yüzde	Birikimli Yüzde
Evet	153	97,5	97,5
Hayır	4	2,5	100,0
Toplam	157	100,0	

Tablo 3.7’de görüldüğü üzere, öğrencilerin %97,5’i İnternet Tabanlı Ders Kayıt Sistemi sayesinde transkriptlerin istenen zamanda alınabildiği düşüncesinde iken %2,5’i bunun tersi kanısındadır.

Tablo 3.8: “İnternet Tabanlı Ders Kayıt Sistemi sayesinde yardımsız kayıt yapabiliyorum.”

	Frekans	Yüzde	Birikimli Yüzde
Evet	147	93,6	93,6
Hayır	10	6,4	100,0
Toplam	157	100,0	

Tablo 3.8’de görüldüğü üzere, öğrencilerin %93,6’sı bu sistem sayesinde yardımsız kayıt yapabildiklerini düşünmekte, %6,4’ü ise yardımsız kayıt yapamadıklarını bildirmektedir.

Tablo 3.9: “Kayıt yaparken hata yaptığımda anında düzeltebiliyorum.”

	Frekans	Yüzde	Birikimli Yüzde
Evet	143	91,1	91,1
Hayır	14	8,9	100,0
Toplam	157	100,0	

Tablo 3.9’da görüldüğü üzere, %91,1 oranında öğrenci kayıt yaparken hata yapıldığında anında düzeltebildiği kanısında iken; %8,9 oranında öğrenci bunun tersini düşünmektedir. Bunun yanı sıra maksimum ders alınabilecek dönemin sistemde görülmesinin hata yapma olasılığını azaltıp azaltmayacağı sorusuna öğrencilerin %96,8’i evet derken %3,2’si hayır demiştir.

#### 4. SONUÇ VE TARTIŞMA

Bu çalışmada; ülkemizdeki bazı üniversitelerde kâğıt üzerinde klasik yollarla yapılan ders kayıtlarının getirdiği iş yükü ve problemleri, teknolojiyi kullanarak büyük ölçüde ortadan kaldırmayı amaçlayan internet tabanlı bir ders kayıt sistemi geliştirilmiştir. Sistem, bu amaçla geliştirilmiş, test edilmiş, uygulaması yapılmış ve öğrencilerden görüşler alınmıştır.

Uygulanan anketten çıkan sonuçta; Öğrencilerin büyük bir kısmı, İTOKS için; belirli tarihler arasında istediği gün ve saatte kayıt yapabileceklerini, danışmanlarına ulaşamama gibi olumsuz durumlarla karşılaşmayacaklarını, sistemin kolay olduğunu, daha az zamanlarını aldığını belirtmişlerdir. Ayrıca, transkriptlerine istedikleri zaman ulaşabileceklerini, yardımsız kayıt yapabileceklerini ve hata yaptıklarında anında düzeltebileceklerini görerek memnuniyetlerini belirtmişlerdir.

İnternet tabanlı ders kayıt sistemi sayesinde öğrenciler danışmanın yardımına gerek duymadan kayıt yapabileceklerdir. Sistem öğrencilere alttan alması gereken zorunlu dersleri ve sorumlu geçtiği dersleri gösterdiği için bir nevi danışmanlık yapmaktadır. Bu sayede öğrenci, alttan hangi dersi almam gerekiyor kaygısına düşmeyecektir. Seçmeli dersler seçilirken derslerin kontenjan bilgileri anlık kontrol edilebilmektedir. Bu da öğrencilerin ders seçiminde kolaylık sağlamaktadır. Böylelikle hem öğrenciler hem de danışmanlar için büyük zaman kazancı sağlanacaktır. Bu sistem sayesinde klasik ders kaydı sırasındaki hataların azalacağı ve danışmanın iş yükünün hafifleyeceği beklenmektedir. Ayrıca her öğrenci için alınması gereken transkript belgesinin ve ders kayıt formlarının basılmasına gerek kalmadığından büyük oranda kâğıt tasarrufu sağlanacaktır.

Sistemde şimdilik eksik olan durum şudur; internetten ders kaydı yapılırsa dahi, resmîyet kazanması için ıslak imza gerektiğinden yapılan ders kaydı bir nevi ön kayıt şeklinde olmaktadır. Bu sistem ancak dijital imzanın resmi işlemlerde geçerli olması ile birlikte büyük kolaylık sağlayacak ve yine kâğıt tasarrufu elde edilmiş olacaktır.

Bazı öğrencilerin, internet üzerinden ders kayıt işlemine yönelik kaygıları tespit edilmiştir. Bunun sebebi, öğrencilerin bilgisayar kullanma oranlarının az olması ve sistemi nasıl kullanacaklarını bilmemeleridir. Öğrencilere bu amaca yönelik eğitimler verilerek, sistemin kullanımı teşvik edilip kaygıları yok edilecektir. Ayrıca, sistemin her zaman deneme amaçlı olarak kullanıma açık olması, bu kaygıları önlemede ikinci bir avantaj olacaktır.

Bu sistemin yürürlüğe sokulmasının ilk aşamalarında zorluklarla karşılaşılacağı düşünülmektedir. Bu zorluklar şu şekildedir;

Mevcut öğrenci otomasyonu, satın alınmış hazır bir paket program olduğundan bu yazılıma müdahale edilememektedir. Bu durumda İnternet Tabanlı Ders Kayıt Sistemi ve fakültenin kullandığı otomasyon ayrı çalışan sistemler olacaklardır. Bu nedenle;

- Mevcut otomasyonda kayıtlı tüm öğrenci bilgilerinin, derslerin ve her öğrencinin ders bilgilerinin dikkatli ve hatasız bir şekilde İTOKS'a aktarılması,
- Tüm öğrenciler için şifreler oluşturulması, gerekmektedir.
- Tüm işlemler için iki sistem birden kullanılmak zorunda olacaktır. Her işi iki kere yapmak zaman kaybı ve gereksiz olduğundan bunu önlemek için ara servisler geliştirilip işlemi bir kere yapıp sonuçların iki sisteme de işlenmesinin sağlanması gerekecektir.

Aynı şekilde halen kullanılan Personel Otomasyonu da üniversite bünyesinde geliştirilmemiştir. Bu yazılıma da müdahale edilememektedir. O yüzden;

- Mevcut otomasyonda kayıtlı akademik personelin bilgilerinin İTOKS'a hatasız aktarılması,
- Her fakültenin Personel İşleri Birimi tarafından öğretim elemanlarına ilk şifrelerinin oluşturulması,
- Akademik personel için bilgi güncellenmesi gerektiğinde iki sisteme de bu güncellemenin uygulanması gerekmektedir.

Ayrıca sistem dâhilindeki tüm kullanıcılara, güvenlik açısından çok önemli olan şifrelerini dağıtmak için izlenecek bir politika belirlenmelidir. Örneğin, her kişi için imza karşılığı kapalı zarf usulü ile şifreler dağıtılabılır.

Tüm bunlar göz önüne alınarak İTOKS'un öncelikle Hasan Ali Yücel Eğitim Fakültesi'nde, daha sonra düzenlemeler yapılarak üniversitemizde kullanılması uygun görülmektedir.

## KAYNAKLAR

1. SCHANK, R.C., 2000, A Vision of Education in the 21th Century, *Technology Horizons in Education*, 27 (6), 42-45.
2. ŞENGÜR, H., 2006, *Bilkent Üniversitesi 'nde Sanal Kampus Oluşturma Süresindeki Çalışmalar* [online], Akademik Bilişim 2006, <http://ab.org.tr/ab06/bildiri/135.doc> [Ziyaret Tarihi: 13 Kasım 2007].
3. Yükseköğretim Kurulu, 2008, *Üniversitelerin Web Siteleri* [online], Yükseköğretim Kurulu, [http://www.yok.gov.tr/universiteler/uni\\_web.htm](http://www.yok.gov.tr/universiteler/uni_web.htm) [Ziyaret Tarihi: 10 Haziran 2008].
4. SSL Digital Certificates, 2006, What is SSL? [online], SSL Digital Certificates, <http://info.ssl.com/article.aspx?id=10241> [Ziyaret Tarihi: 25 Ocak 2008].
5. SARP KAYA Y., KARASEKRETER N. ve DOĞAN M., 2007, *Uzaktan Eğitim Yazılım Altyapısının Bilginin Kalıcılığına ve geçerliliğine etkisi* [online], Akademik Bilişim 2007, <http://ab.org.tr/ab07/bildiri/224.doc> [Ziyaret Tarihi: 24 Aralık 2007].
6. ÇANKAYA, M.N., 2001, *ASP 'ye Giriş* [online], Microsoft Ürünleri Yazılım Geliştiriciler Grubu, <http://www.yazgelistir.com/Makaleler/MakaleDetay.aspx?MakaleId=100000034> [Ziyaret Tarihi: 27 Aralık 2007].
7. Wikipedi Özgür Ansiklopedi, 2007, *PHP* [online], Wikipedi Türkiye, <http://tr.wikipedia.org/wiki/PHP> [Ziyaret Tarihi: 27 Aralık 2007].
8. Wikipedia the free encyclopedia, 2007, *JavaBean* [online], Wikipedia the free encyclopedia, <http://en.wikipedia.org/wiki/JavaBean> [Ziyaret Tarihi: 27 Aralık 2007].
9. İNNER, B., 2004, *JSP ASP COLDFUSION ve PHP Karşılaştırması* [online], Godoro Eğitim Portalı, [http://www.godoro.com/divisions/ehil/mecmua/Magazines/Articles/txt/html/article\\_JSPAndASPAndPHP.html](http://www.godoro.com/divisions/ehil/mecmua/Magazines/Articles/txt/html/article_JSPAndASPAndPHP.html) [Ziyaret Tarihi: 04 Ocak 2008].
10. YAVUZ, Z. , 2006, *Dinamik Web Programlama* [online], Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü, [http://ceng.ktu.edu.tr/dsy/dinamik\\_web\\_programlama.pdf](http://ceng.ktu.edu.tr/dsy/dinamik_web_programlama.pdf) [Ziyaret Tarihi: 04 Ocak 2008].

11. ÇANKAYA, M.N., 2002, *ASP.NET Nedir?* [online], Microsoft Akademik Yazılım Geliştiriciler Grubu, [http://www.msakademik.net/makaleler\\_detay.aspx?id=1](http://www.msakademik.net/makaleler_detay.aspx?id=1) [Ziyaret Tarihi: 11 Şubat 2008].
12. ALTUNSÖĞÜT, Ö., 2005, *ASP.NET Teknolojilerini Kullanarak Bir Satın Alma Portalı Uygulamasunun Geliştirilmesi*, Yüksek Lisans, Trakya Üniversitesi Fen Bilimleri Enstitüsü.
13. NACAROĞLU, A.F., 2002, *.NET Teknolojilerine Giriş* [online], C# Nedir? Yazılım Merkezi, <http://www.csharpnedir.com/makalegoster.asp?MId=17> [Ziyaret Tarihi: 11 Şubat 2008].
14. LENZ G., MOELLER T., 2003, *.NET A Complete Development Cycle*, Addison-Wesley Professional, Boston, 0-321-16882-8
15. Software AG, 2005, *XML Nedir?* [online], <http://bilisim.milliyet.com.tr/detay.asp?id=123> [Ziyaret Tarihi: 13 Ocak 2008].
16. MASON J., 2006, *SGML and XML markup languages in vendor-neutral, machine-independent and human-readable format* [online], International Organization for Standardization , <http://www.iso.org/iso/en/commcentre/pdf/Sgml0002.pdf> [Ziyaret Tarihi: 13 Ocak 2008].
17. AYAN E., 2005, *XML Nedir?-Ne İşimize Yarar?* [online], MaxiASP, [http://www.maxiasp.net/d\\_xml\\_nedir\\_ne\\_isimize\\_yarar\\_.aspx](http://www.maxiasp.net/d_xml_nedir_ne_isimize_yarar_.aspx) [Ziyaret Tarihi: 15 Ocak 2008].
18. QUIN L., 2008, *Extensible Markup Language (XML)* [online], The World Wide Web Consortium, <http://www.w3.org/XML> [Ziyaret Tarihi: 15 Ocak 2008].
19. JORGENSEN D., 2002, *Developing .NET Web Services with XML*, Syngress, Rockland MA, 1-928994-81-4
20. ECKERSON, W.W., 1995, Three Tier Client/Server Architecture: Achieving Scalability, Performance, and Efficiency in Client Server Applications, *Open Information Systems*, 10 (1), 3-22.
21. Vikipedi Özgür Ansiklopedi, 2008, *SQL* [online], Vikipedi Türkiye, <http://tr.wikipedia.org/wiki/SQL> [Ziyaret Tarihi: 27 Haziran 2008].
22. SHARP TOOLBOX, 2008, *Object-Relational Mapping* [online], Sharp Toolbox, <http://sharptoolbox.com/categories/object-relational-mappers> [Ziyaret Tarihi: 28 Eylül 2008].

23. Hibernate, 2008, *NHibernate For .NET* [online], Hibernate, <http://www.hibernate.org/343.html> [Ziyaret Tarihi: 30 Eylül 2008].
24. Yazgelistir.com, 2008, *Object/Relational Mapping ve NHibernate* [online], Yazgelistir.com, <http://www.yazgelistir.com/Makaleler/1000000885.ygpx> [Ziyaret Tarihi: 30 Eylül 2008].
25. ŞİMŞEK, B., 2004, *CVS ile Açık Kod Yazılım Geliştirme* [online], Linux Belgelendirme Çalışma Grubu, <http://pdf.belgeler.org/howto/cvs-acikkod-nasil.pdf> [Ziyaret Tarihi: 30 Eylül 2008].
26. Vikipedi Özgür Ansiklopedi, 2008, *SPSS* [online], Vikipedi Türkiye, <http://tr.wikipedia.org/wiki/SPSS> [Ziyaret Tarihi: 20 Ekim 2008].

## EKLER

## EK A

## Ders kayıt formu

T.C. ISTANBUL UNIVERSİTESİ  
HASAN ALI YUCEL EGİTİM FAKÜLTESİ  
REHBERLİK VE PSİKOLOJİK DANIŞMANLIK LİSANS PROGRAMI  
2007 / 2008 Öğretim Yılı Ders Kayıt Çizelgesi

Öğrencinin;  
Adı,Soyadı:  
Fakülte No:  
Yarıyılı: GÜZ (1)  
Başvuru Tarihi:

Öğrenim Harcı  
Taksit Makbuz tarihi veya katkı kredisi no  
Makbuz no. Miktar TL (döviz)

SEÇTİĞİNİZ DERSLERİN YANINDA YER ALAN KUTULARI İŞARETLEYİNİZ.  
(Y =Yeni (ilk kez) alınan ders; Tek = Tekrarlanan ders; K = Kredi; T = Teorik; U = Uygulama)

1.yarıyıl						BİRİNCİ SINIF						2.yarıyıl								
T	Y	KODU	DERSİN ADI	T	U	K	T	Y	KODU	DERSİN ADI	T	U	K	T	Y	KODU	DERSİN ADI	T	U	K
		810101	Bilimsel Araştırma Teknikleri I	2	0	2			810201	Bilimsel Araştırma Teknikleri II	2	0	2			810201	Bilimsel Araştırma Teknikleri II	2	0	2
		810102	Eğitim Sosyolojisi ve Çağdaş Akımlar	2	0	2			810202	Eğitimde Program Uygulamaları (K)	2	0	2			810202	Eğitimde Program Uygulamaları (K)	2	0	2
		810103	Çocuk Gelişimi ve Eğitimi I	2	0	2			810203	Çocuk Gelişimi ve Eğitimi II	2	0	2			810203	Çocuk Gelişimi ve Eğitimi II	2	0	2
		810104	Psikolojiye Giriş I	2	0	2			810204	Psikolojiye Giriş II	2	0	2			810204	Psikolojiye Giriş II	2	0	2
		810105	Eğitim Prog. Ve Öğretim Yöntemleri (K)	3	0	3			810205	Rehberlik ve Psikolojik Danış. Giriş	2	0	2			810205	Rehberlik ve Psikolojik Danış. Giriş	2	0	2
		810106	Felsefeye Giriş	2	0	2			810206	Türkçe II: Sözlü Anlatım	2	0	2			810206	Türkçe II: Sözlü Anlatım	2	0	2
		810107	Türkçe I: Yazılı Anlatım	2	0	2			810207	Yabancı Dil II	3	0	3			810207	Yabancı Dil II	3	0	3
		810108	Yabancı Dil I	3	0	3			810208	Atatürk İlkeleri ve İnkılap Tarihi II	2	0	2			810208	Atatürk İlkeleri ve İnkılap Tarihi II	2	0	2
		810109	Atatürk İlkeleri ve İnkılap Tarihi I	2	0	2			810209	Okul Değişimi I	1	4	3			810209	Okul Değişimi I	1	4	3
		810110	Öğretmenlik Mesleğine Giriş	3	0	3			810210	Özel Eğitime Giriş	3	0	3			810210	Özel Eğitime Giriş	3	0	3
		810112	Güzel Sanatlar (Seçimlik)	3	0	3			810212	Güzel Sanatlar (Seçimlik)	3	0	3			810212	Güzel Sanatlar (Seçimlik)	3	0	3
		<b>Toplam</b>		<b>20</b>	<b>00</b>	<b>20</b>			<b>Toplam</b>		<b>19</b>	<b>04</b>	<b>21</b>			<b>Toplam</b>		<b>19</b>	<b>04</b>	<b>21</b>

1.yarıyıl						İKİNCİ SINIF						2.yarıyıl								
T	Y	KODU	DERSİN ADI	T	U	K	T	Y	KODU	DERSİN ADI	T	U	K	T	Y	KODU	DERSİN ADI	T	U	K
		810301	İstatistik I	2	0	2			810401	İstatistik II	2	0	2			810401	İstatistik II	2	0	2
		810302	Ergenlik Psikolojisi I	2	0	2			810402	Ergenlik Psikolojisi II	2	0	2			810402	Ergenlik Psikolojisi II	2	0	2
		810303	Temel Eğitim Teorileri ve Sorunlar	2	0	2			810403	Karşılaştırmalı Okul Sistemleri (K)	2	0	2			810403	Karşılaştırmalı Okul Sistemleri (K)	2	0	2
		810304	Çocuk Hakları ve Kor. Muh. Çocuklar	2	0	2			810404	Okul Öncesi Eğitim	3	0	3			810404	Okul Öncesi Eğitim	3	0	3
		810305	Birseyi Tanıma Teknikleri (K)	2	0	2			810405	Rehberlik ve Psikolojik Dan. Teknikleri (K)	2	0	2			810405	Rehberlik ve Psikolojik Dan. Teknikleri (K)	2	0	2
		810306	Fizyoloji (K)	3	0	3			810406	Reh. Ve Psik. Danışmanl. Etik Kuralları (K)	2	0	2			810406	Reh. Ve Psik. Danışmanl. Etik Kuralları (K)	2	0	2
		810307	Gelişim ve Öğrenme	3	0	3			810407	Psikofizyoloji (K)	2	0	2			810407	Psikofizyoloji (K)	2	0	2
		810308	Birseyi Tanıma Teknikleri	3	0	3			810408	Öğretimde Farklılaşma ve Değerlendirme	3	2	4			810408	Öğretimde Farklılaşma ve Değerlendirme	3	2	4
		810309	Fizyoloji I	2	0	2			810409	Eğitim Tarihi	2	0	2			810409	Eğitim Tarihi	2	0	2
		810310	Sağlık Bilgisi ve İlk Yardım (Seçimlik)	2	2	3			810410	Rehb. ve Psik. Dan.'da Çağdaş Akımlar	3	0	3			810410	Rehb. ve Psik. Dan.'da Çağdaş Akımlar	3	0	3
		810311	Bilgisayar	2	2	3			810411	Fizyoloji II	2	0	2			810411	Fizyoloji II	2	0	2
		<b>Toplam</b>		<b>18</b>	<b>02</b>	<b>19</b>			<b>Toplam</b>		<b>17</b>	<b>02</b>	<b>18</b>			<b>Toplam</b>		<b>17</b>	<b>02</b>	<b>18</b>

1.yarıyıl						ÜÇÜNCÜ SINIF						2.yarıyıl								
T	Y	KODU	DERSİN ADI	T	U	K	T	Y	KODU	DERSİN ADI	T	U	K	T	Y	KODU	DERSİN ADI	T	U	K
		810501	Davranış Bozuklukları ve Eğitimi I	2	0	2			810601	Okuma Bozuklukları (K)	3	0	3			810601	Okuma Bozuklukları (K)	3	0	3
		810502	Çocuk Suçluluğu ve Yeniden Eğitim	2	0	2			810602	Davranış Bozuklukları ve Eğitimi II	2	0	2			810602	Davranış Bozuklukları ve Eğitimi II	2	0	2
		810503	Rehberlik Semineri I (K)	1	0	1			810603	Yetişkin Psikolojisi ve Ana-Baba Eğitimi (K)	2	0	2			810603	Yetişkin Psikolojisi ve Ana-Baba Eğitimi (K)	2	0	2
		810504	Mesleğe Yönelme	2	0	2			810604	Rehberlik Semineri II (K)	1	0	1			810604	Rehberlik Semineri II (K)	1	0	1
		810505	Zeka Engelli Çocukların Psik. Ve Eğitimi (K)	2	0	2			810605	Ustun Çocukların Psikolojisi ve Eğitimi	2	0	2			810605	Ustun Çocukların Psikolojisi ve Eğitimi	2	0	2
		810506	Okul Yönetimi (K)	2	0	2			810606	Tem. İletişim Bec. ve İletişim Çalışmaları (K)	2	0	2			810606	Tem. İletişim Bec. ve İletişim Çalışmaları (K)	2	0	2
		810507	Resimleriyle Çocuk (K)	2	0	2			810607	Okul Psikolojisi (K)	2	0	2			810607	Okul Psikolojisi (K)	2	0	2
		810508	Aile Danışmanlığı (K)	2	0	2			810608	Sınıf Yönetimi (K)	3	0	3			810608	Sınıf Yönetimi (K)	3	0	3
		810509	Öğretim Tekn. Ve Materyal Geliştirme (K)	3	0	3			810609	Özel Öğretim Yöntemleri I (K)	3	0	3			810609	Özel Öğretim Yöntemleri I (K)	3	0	3
		810510	Öğretim Tekn. Ve Materyal Geliştirme	2	2	3			810610	Özel Öğretim Yöntemleri II	2	2	3			810610	Özel Öğretim Yöntemleri II	2	2	3
		810511	Zeka Engelli Çocukların Psik. ve Eğitimi (K)	3	0	3			810611	Özel Öğrenme Güçlükleri	3	0	3			810611	Özel Öğrenme Güçlükleri	3	0	3
		810512	Farklı Öğretim Kad. Reh. (K)	2	0	2			810612	Aile Danışmanlığı (K)	2	0	2			810612	Aile Danışmanlığı (K)	2	0	2
		810513	Yetişkin Psikolojisi ve Ana-Baba Eğitimi	2	0	2			810613	Çocuk Psikolojisi Uygulamaları	2	0	2			810613	Çocuk Psikolojisi Uygulamaları	2	0	2
		810514	Temel İletişim Becerileri	2	0	2			810614	Eğitimde Program Uygulamaları (K)	2	0	2			810614	Eğitimde Program Uygulamaları (K)	2	0	2
		810515	Okul Yönetimi	3	0	3			810615	Rehberlik ve Psik. Dan. Etik Kuralları	2	0	2			810615	Rehberlik ve Psik. Dan. Etik Kuralları	2	0	2
		810516	Zeka Engelli Çocukların Eğitimi ve Psikolojisi	2	0	2			810616	Sınıf Yönetimi	2	2	3			810616	Sınıf Yönetimi	2	2	3
		<b>Toplam</b>		<b>17</b>	<b>02</b>	<b>18</b>			<b>Toplam</b>		<b>15</b>	<b>04</b>	<b>17</b>			<b>Toplam</b>		<b>15</b>	<b>04</b>	<b>17</b>

/.

1.yarıyıl				DÖRDÜNCÜ SINIF				2.yarıyıl					
T	Y	KODU	DERSİN ADI	T	U	K	T	Y	KODU	DERSİN ADI	T	U	K
○	○	810701	Örgütsel Psikoloji	2	0	2	○	○	810801	İnsan Kaynakları	2	0	2
○	○	810702	Eğitim ve Öğretimde Testler I	2	0	2	○	○	810802	Liderlik (K)	2	0	2
○	○	810703	Rah. Ve Psik. Danış. Çağdaş Akımlar (K)	3	0	3	○	○	810803	Eğitim ve Öğretimde Testler II	2	0	2
○	○	810704	Grupla Psikolojik Danışma	2	0	2	○	○	810804	Psikiyatri (K)	2	0	2
○	○	810705	Kişiler Arası İlişkiler (K)	2	0	2	○	○	810805	Rehabilitasyon	3	0	3
○	○	810706	Okul Deneyimi II (K)	2	2	3	○	○	810806	Öğretmenlik Uygulamaları	2	6	6
○	○	810707	Özel Öğretim Yöntemleri II	2	2	3	○	○	810807	Seçmeli II	3	0	3
○	○	810708	Seçmeli I	3	0	3	○	○	810808	Karşılaştırmalı Okul Sistemleri	2	0	2
○	○	810709	Test Uygulama I (Seçimlik)	2	2	3	○	○	810809	Psikiyatri II	2	0	2
○	○	810710	Rah. Ve Psik. Dan. İlke ve Teknikleri	2	0	2	○	○	810810	Resimleriyle Çocuk	2	0	2
○	○	810711	Psikiyatri I	2	0	2	○	○	810811	Test Uygulama II (Seçimlik)	2	2	3
○	○	810712	Okul Deneyimi II	1	4	3	○	○	810812	*Seçimlik: Öğrenme Psikolojisi (D.B.Ö.)	3	0	3
○	○	810713	*Seçimlik: Öğrenme Psikolojisi (D.B.Ö.)	3	0	3							
		<b>Toplam</b>		<b>16</b>	<b>06</b>	<b>19</b>			<b>Toplam</b>		<b>18</b>	<b>06</b>	<b>21</b>

\* D.B.O.= Diğer bölümlerin öğrencileri için seçimlik ders. (Bu dersi, bu anabilim dalının öğrencileri alamaz.)

Yeni Aldığı Derslerin Toplam Kredisini:  
Tkrarlanan Derslerin Toplam Kredisini:  
Toplam Kredisi:

Yazışma adresi:

Telefon No:

Danışmanın,  
Adı Soyadı ve İmzası:

Tarih:

Programın Toplam Kredisini:

Bu dönem yukarıda işaretlediğim derslere devam edeceğimi bildiririm.

Öğrencinin imzası:

Tarih:

Kayıd Yapan Memurun,  
Adı Soyadı ve İmzası:

Tarih:

**EK B****İNTERNET TABANLI DERS KAYIT SİSTEMİ ANKETİ**

Fakültemiz’de uygulanmakta olan Klasik Ders Kayıt Sistemi’ne alternatif olarak geliştirdiğimiz İTOKS adını verdiğimiz internet tabanlı ders kayıt sistemini kullandınız.

Aşağıda bu sistem hakkında düşüncelerinizi ölçen sorular bulunmaktadır. Bu soruları dikkatle okuyarak, size en uygun gelen seçeneği işaretleyiniz.

Lütfen hiç bir soruyu boş bırakmayınız.

**1) İnternet Tabanlı Ders Kayıt Sistemi hakkında bilgi sahibi misiniz?**

- a) Evet
- b) Hayır

**2) İnternet Tabanlı Ders Kayıt Sistemi, belirtilen kayıt tarihleri arasında istediğim gün ve saatte kayıt yapabilmemi sağlar.**

- a) Evet
- b) Hayır

**3) İnternet Tabanlı Ders Kayıt Sistemi, danışmanıma ulaşamama gibi olumsuz durumları ortadan kaldırır.**

- a) Evet
- b) Hayır

**4) İnternet Tabanlı Ders Kayıt Sistemini kullanmak kâğıt israfını önler.**

- a) Evet
- b) Hayır

**5) Her zaman bilgisayara ulaşmam kolay olmadığı için bu sistemin kullanışlı olmadığını düşünüyorum.**

- a) Evet
- b) Hayır

- 6) **İnternet Tabanlı Ders Kayıt Sistemi'ni kullanmaktan hoşlanmadım.**
- a) Evet
  - b) Hayır
- 7) **İnternet Tabanlı Ders Kayıt Sistemini kullanmanın kolay olduğu kanısındayım.**
- a) Evet
  - b) Hayır
- 8) **İnternet Tabanlı Ders Kaydı, Klasik Ders Kaydı'na göre daha az zamanımı aldı.**
- a) Evet
  - b) Hayır
- 9) **Ders kaydı için okula gelmek zorunda olmayacağım için çok memnunum.**
- a) Evet
  - b) Hayır
- 10) **İnternet Tabanlı Ders Kayıt Sistemi'ni kullanmak bana zaman kazandırdı.**
- a) Evet
  - b) Hayır
- 11) **Bir problemle karşılaştığımda anında dönüt alamadığım için bu sistemin pek işler olmadığını düşünüyorum.**
- a) Evet
  - b) Hayır
- 12) **İnternet Tabanlı Ders Kayıt Sistemi sayesinde transkriptimi istediğim zamanda alabiliyorum.**
- a) Evet
  - b) Hayır

**13) İnternet Tabanlı Ders Kayıt Sistemi sayesinde yardımsız kayıt yapabiliyorum.**

- a) Evet
- b) Hayır

**14) İnternet Tabanlı Ders Kayıt Sistemi'nde alttan alacağım dersleri daha rahat görebiliyorum.**

- a) Evet
- b) Hayır

**15) İnternet Tabanlı Ders Kayıt Sistemi'nde anında yardım alabildiğim için çok mutluyum.**

- a) Evet
- b) Hayır

**16) İnternet Tabanlı Ders Kayıt Sistemi'nde Kayıt yaparken hata yaptığımda anında düzeltebiliyorum.**

- a) Evet
- b) Hayır

**17) İnternet Tabanlı kayıt yaptırmayı çok kullanışlı bulmuyorum.**

- a) Evet
- b) Hayır

**18) Tüm kayıt bilgilerimi bu sayede yedeklemiş oluyorum.**

- a) Evet
- b) Hayır

**19) Danışman ile kurulan birebir etkileşimi olumsuz etkilediğini düşünüyorum.**

- a) Evet
- b) Hayır

**20) Maksimum ders alabileceğim dönemi sistemde görmek hata yapmamı azaltıyor.**

- a) Evet
- b) Hayır

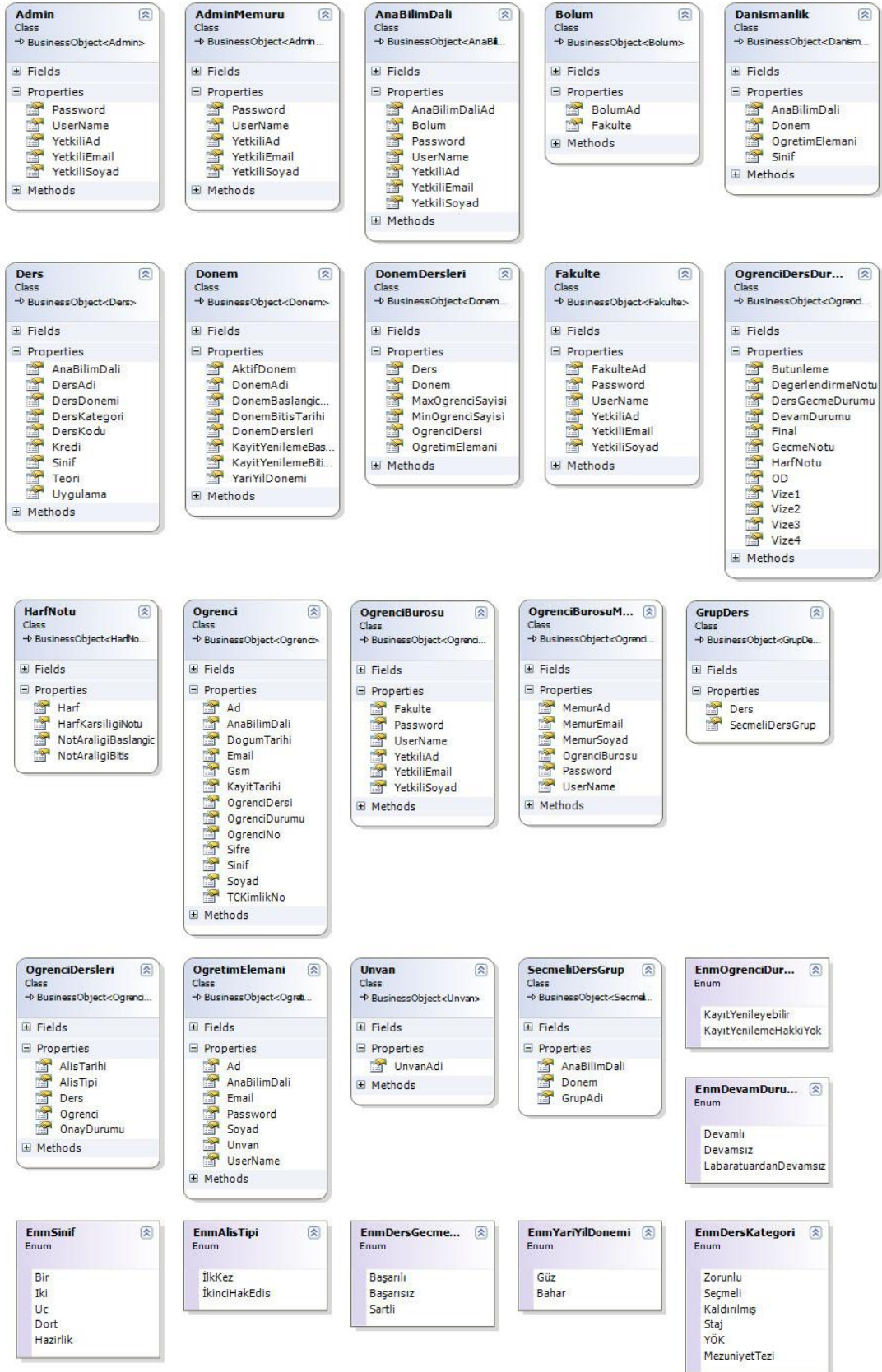
**21) İnternet Tabanlı Ders Kayıt Sisteminin anlaşılır bir şekilde hazırlanmış olduğunu düşünüyorum.**

- a) Evet
- b) Hayır

**Katılımınız için teşekkürler.**

## EK C

## İTOKS'DAKİ NESNELER



**EK D****İTOKS'DAN ÖRNEK KOD****Öğrenci Sınıfı (Class)**

```

using System;
using System.Collections.Generic;
using System.Text;

namespace ITOKS.Entity
{
    [Serializable]
    [NHibernate.Mapping.Attributes.Class(Lazy = false)]
    public class Öğrenci :
MH.NHibernateBase.Web.BusinessObject.BusinessObject<Öğrenci>
    {
        private string _ÖğrenciNo;
        [NHibernate.Mapping.Attributes.Property]
        public string ÖğrenciNo
        {
            get { return _ÖğrenciNo; }
            set { _ÖğrenciNo = value; }
        }

        private AnaBilimDali _AnaBilimDali;
        [NHibernate.Mapping.Attributes.ManyToOne(Column =
"AnaBilimDali", ClassType = typeof(AnaBilimDali), Cascade =
NHibernate.Mapping.Attributes.CascadeStyle.None)]
        public AnaBilimDali AnaBilimDali
        {
            get { return _AnaBilimDali; }
            set { _AnaBilimDali = value; }
        }

        private EnmÖğrenciDurumu _ÖğrenciDurumu;
        [NHibernate.Mapping.Attributes.Property]
        public EnmÖğrenciDurumu ÖğrenciDurumu
        {
            get { return _ÖğrenciDurumu; }
            set { _ÖğrenciDurumu = value; }
        }

        private string _TCKimlikNo;
        [NHibernate.Mapping.Attributes.Property]
        public string TCKimlikNo
        {
            get { return _TCKimlikNo; }
            set { _TCKimlikNo = value; }
        }

        private string _Ad;
        [NHibernate.Mapping.Attributes.Property]
        public string Ad
        {
            get { return _Ad; }
            set { _Ad = value; }
        }

        private string _Soyad;

```

```
[NHibernate.Mapping.Attributes.Property]
public string Soyad
{
    get { return _Soyad; }
    set { _Soyad = value; }
}

private string _Sifre;
[NHibernate.Mapping.Attributes.Property]
public string Sifre
{
    get { return _Sifre; }
    set { _Sifre = value; }
}

private EnumSinif _Sinif;
[NHibernate.Mapping.Attributes.Property]
public EnumSinif Sinif
{
    get { return _Sinif; }
    set { _Sinif = value; }
}

private DateTime _DogumTarihi;
[NHibernate.Mapping.Attributes.Property]
public DateTime DogumTarihi
{
    get { return _DogumTarihi; }
    set { _DogumTarihi = value; }
}

private string _Gsm;
[NHibernate.Mapping.Attributes.Property]
public string Gsm
{
    get { return _Gsm; }
    set { _Gsm = value; }
}

private string _Email;
[NHibernate.Mapping.Attributes.Property]
public string Email
{
    get { return _Email; }
    set { _Email = value; }
}

private DateTime _KayitTarihi;
[NHibernate.Mapping.Attributes.Property]
public DateTime KayitTarihi
{
    get { return _KayitTarihi; }
    set { _KayitTarihi = value; }
}
```

```

    /// <summary>
    /// Login Sorgusu
    /// </summary>
    public static Ogrenci Login(string UserName, string Password)
    {
        NHibernate.ICriteria criteria =
MH.NHibernateBase.Web.ProviderWeb.Session.CreateCriteria(typeof(Ogrenc
i));

        criteria.Add(NHibernate.Expression.Expression.Eq("OgrenciNo",
UserName));
        criteria.Add(NHibernate.Expression.Expression.Eq("Sifre",
Password));

        return criteria.UniqueResult<Ogrenci>();
    }

    private IList<OgrenciDersleri> _OgrenciDersi;

    [NHibernate.Mapping.Attributes.Bag(0)]
    [NHibernate.Mapping.Attributes.Key(1, Column = "Ogrenci")]
    [NHibernate.Mapping.Attributes.OneToMany(2, ClassType =
typeof(OgrenciDersleri))]
    public IList<OgrenciDersleri> OgrenciDersi
    {
        get { return _OgrenciDersi; }
        set { _OgrenciDersi = value; }
    }

    public static IList<Ogrenci> GetAllOgrenciByDonem(Donem
_Donem)
    {
        NHibernate.ICriteria criteria =
MH.NHibernateBase.Web.ProviderWeb.Session.CreateCriteria(typeof(Ogrenc
i));
        criteria.CreateCriteria("OgrenciDersleri",
"ad").Add(NHibernate.Expression.Expression.Eq("Donem", _Donem));
        return criteria.List<Ogrenci>();
    }

    public static List<Ogrenci>
GetALLOgrenciByABDandSinif(AnaBilimDali _ABD, EnmSinif _Sinif)
    {
        NHibernate.ICriteria criteria =
MH.NHibernateBase.Web.ProviderWeb.Session.CreateCriteria(typeof(Ogrenc
i));

        criteria.Add(NHibernate.Expression.Expression.Eq("AnaBilimDali",
_ABDD));
        criteria.Add(NHibernate.Expression.Expression.Eq("Sinif",
_Sinif));

        List<Ogrenci> _Ogrenci =
(List<Ogrenci>)criteria.List<Ogrenci>();
        return _Ogrenci;
    }

```

```
public static IList<Ogrenci> GetALLOgrenciByDers(Ders _Ders)
{
    NHibernate.ICriteria criteria =
MH.NHibernateBase.Web.ProviderWeb.Session.CreateCriteria(typeof(Ogrenc
i));
    criteria.CreateCriteria("AlinanDonemDersleri",
"ad").Add(NHibernate.Expression.Expression.Eq("Ders", _Ders));
    return criteria.List<Ogrenci>();
}

public static IList<Ogrenci>
GetALLOgrenciByDonemDersi(DonemDersleri _Ders)
{
    NHibernate.ICriteria criteria =
MH.NHibernateBase.Web.ProviderWeb.Session.CreateCriteria(typeof(Ogrenc
i));
    criteria.CreateCriteria("OgrenciDersleri",
"ad").Add(NHibernate.Expression.Expression.Eq("Ders", _Ders.Ders));
    return criteria.List<Ogrenci>();
}
}
```

## ÖZGEÇMİŞ

1984 yılında İstanbul'da doğan Burak ŞİŞMAN, ilköğretimini Fatma Süslügil İlköğretim Okulu'nda ve ortaöğretimini Eminönü Cibali Yabancı Dil Ağırlıklı Lisesi'nde tamamladı. 2002 yılında Dumlupınar Üniversitesi, Fen-Edebiyat Fakültesi Matematik Bölümü'nü kazandı. 2006 yılında bölüm üçüncüsü ve onur öğrencisi olarak mezun oldu. Yine 2006 yılında İstanbul Üniversitesi, Fen Bilimleri Enstitüsü Enformatik Bölümü'nde yüksek lisans eğitimine başladı.

Dumlupınar Üniversitesi'ndeki öğrenciliği süresince, Dumlupınar Üniversitesi Bilgi İşlem Dairesi'nde ve daha sonra Enformatik Bölümü'nde asistan öğrenci olarak part-time çalıştı. Bu süreçte, üniversite bünyesinde bazı bölümler ve projeler için web sayfaları tasarladı ve yazılımlar geliştirdi. Aynı zamanda sektöre birçok web sayfası tasarladı ve yazılımlar geliştirdi.

2006 yılının Aralık ayında İstanbul Üniversitesi, Hasan Ali Yücel Eğitim Fakültesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü'nde Araştırma Görevlisi ünvanı ile akademisyen oldu ve halen görevini daha çok severek devam ettirmektedir.

Yapay Zeka ve Yapay Sinir Ağları ilgisini çeken konulardır. Bunların dışında, e-öğrenme üzerine çalışmalar yapmaktadır.

İyi derecede İngilizce bilmekte, İspanyolca ve Fransızca öğrenmektedir.