

S: / H

**İSTANBUL TEKNİK ÜNİVERSİTESİ**

**Bilgisayar ve Bilişim Fakültesi**

**Bilgisayar Mühendisliği Bölümü**

**MÜŞTERİ DESTEK SİSTEMİ İÇİN WEB  
SİTESİNİN YAZILIMI**

**STAJ**

**Abdullah AYDEĞER**

**YAZ / 2011**

## **Stajların**

**İTÜ Bilgisayar ve Bilişim Fakültesi**

**Staj Kuralları ve Yapılması Gerekenler**

***uyarınca yapılması gereklidir.***

***aksi durumda stajlar geçerli sayılmayacaktır.***

Staj Raporunun Hazırlanmasında Dikkat Edilecek Noktalar:

1. Yapılan iş ile ilgili açıklama.
  - 1.1. Her sayfanın başlık kısmı doldurulacak.
  - 1.2. İş konusu hakkında özet olarak bir ön bilgi verilecektir.
  - 1.3. Yapılan iş teknik bir rapor şeklinde yazılacaktır.
2. Yapılan şekil ve yazılan yazılar ile ilgili açıklamalar.
  - 2.1. Yazılar yalnızca yapılan iş ile ilgili olacaktır.
3. Staj defterlerinin kabul edilmesiyle ilgili açıklama.
  - 3.1. Staj Raporunun ön sayfalarında gösterilen yer ile son sayfa çalışılan kurumun resmi mührü ile mühürlenmiş, yetkili teknik sorumlusu tarafından da ilgili yerler doldurulmuş ve imzalanmış olacaktır. Aksi durumlarda yapılan çalışmalar staj komisyonunca geçerli sayılmayacaktır.
4. Daha fazla bilgi İTÜ Bilgisayar ve Bilişim Fakültesi Staj Kuralları <http://www.bb.itu.edu.tr> sayfasındadır.

**İstanbul Teknik Üniversitesi**  
**Bilgisayar Mühendisliği Bölümü**  
**İTÜ-SUNY Bilişim Sistemleri Mühendisliği Program**  
**STAJ RAPORU**

Akademik Yıl: 2011

Staj yapılan dönem (Güz (G) / Bahar (B) / Yaz (Y)): Y

**Öğrenci ile ilgili bilgiler**

Adı ve Soyadı: Abdullah Aydeğer

Öğrenci Numarası: 040090533

Bölüm: Bilgisayar Mühendisliği

Program: Bilgisayar Mühendisliği

e-posta adresi: aydeger@itu.edu.tr

(Cep) Tel No: 0538 448 99 08

ÇAP öğrencisi misiniz? (Hayır (H) / ÇAP yaptığınız Fakülte – Bölüm): H

Staj yaptığınız dönem mezun olacaksınız? (Evet (E) / Hayır (H)): H

(Transkriptinizi gösterilen yere koyun.)

Sonsuz sınav hakkınız varmı? (Evet (E) / Hayır (H)): H

(Orijinal belgeyi gösterilen yere koyun.)

SUNY öğrenci misiniz? (Evet (E) / Hayır (H)): H

**Öğrencinin çalıştığı kurum ile ilgili bilgiler**

İsmi: Etiya Bilgi Teknolojileri

Birimi:

Web Adresi: <http://www.etiya.com/>

Kısa adresi: Yıldız Teknik Üniversitesi Davutpaşa Kampüsü Teknopark Faz 2 B1 Blok k:1 D:12-13-14-15

Esenler-İstanbul

**Yetkili kiři ile ilgili bilgiler**

Ciltlenmemiř Staj Raporunu imzalayıp ve damgalamayın!

(İTÜ Bilgisayar ve Biliřim Fakóltesi Staj Kuralları <http://www.bb.itu.edu.tr>).

Bólümü:

Mühür / İmza

Ünvanı:

Adı ve Soyadı:

e-posta:

(Cep) Tel No:

**Yapılan iş ile ilgili bilgiler**

Konu:

Grubu (Yazılım (Y) / Donanım (D)): Y

Stajın başlangıç tarihi: 15.08.2011

Stajın bitiş tarihi: 16.09.2011

Stajda net çalışılan süre sayısı (Hafta / Gün): 22

NOT: Net iş günü hafta sonlarını, bayram günlerini, resmi tatil günlerini ve herhangi bir nedenle alınan izinleri içermez.

Stajınızı Staj Kurallarında belirtildiđi gibi stajınızı süreklimi yaptınız? (Evet (E) / Hayır (H)): E

Staj yaptığınız süre içinde İTÜ tarafından sigortalandınız mı (Kurum tarafından yapılan sigorta değil) ? (Evet (E) / Hayır (H)): E

**NOT: Türkiye’de yapılan a) İTÜ tarafından yapılan sigorta kapsamında olmayan b) sürekli olmayan stajlar kabul edilmez.**

Yukarıda belirttiđim bilgilerin dođruluđunu, getirdiđi yükümlölükleri kabul ederim.

Tarih ve öğrencinin İmzası

**Bu (iv.) dördüncü sayfadır. Bu sayfaya kurum tarafından doldurulmuş olan (ıslak damgalı) “STAJ BİTİŞ BELGESİ” ni ([staj.itu.edu.tr](http://staj.itu.edu.tr) web sayfasındaki D4 Formu (2)’yi yerleştirin.**

**Kağıtların arkasını BOŞ bırakın.**

**Bu (v.) beşinci sayfadır. Bu sayfadan başlamak üzere stajı yaptığınız dönemi içeren transkripti yerleştirin.**

**Bu bir nottur, çıktı almadan önce siliniz.**

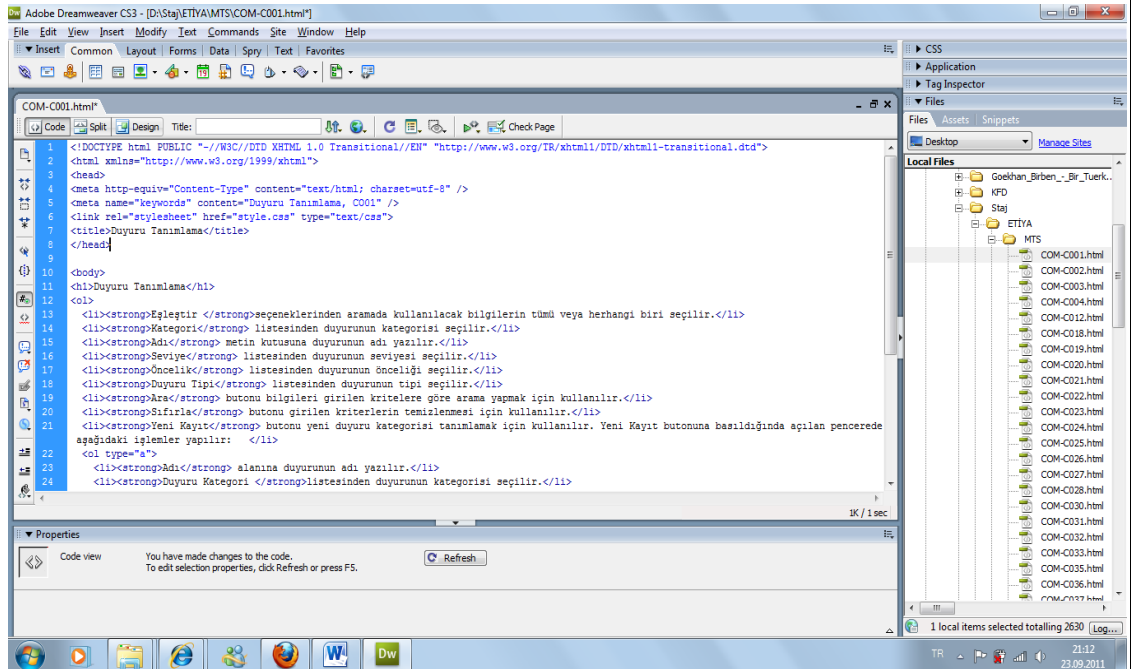
## Yapılan iş:

### Şirkete Ait Bir Projenin Yardım Dosyalarının Hazırlanması

Yardım dosyalarının hazırlandığı ortam Adobe Dreamweaver CS3, kullanılan dil ise HTML'dir. Öncelikle bu ortam ve dil hakkında genel bilgiler aşağıda anlatılan gibidir.

## Adobe Dreamweaver CS3

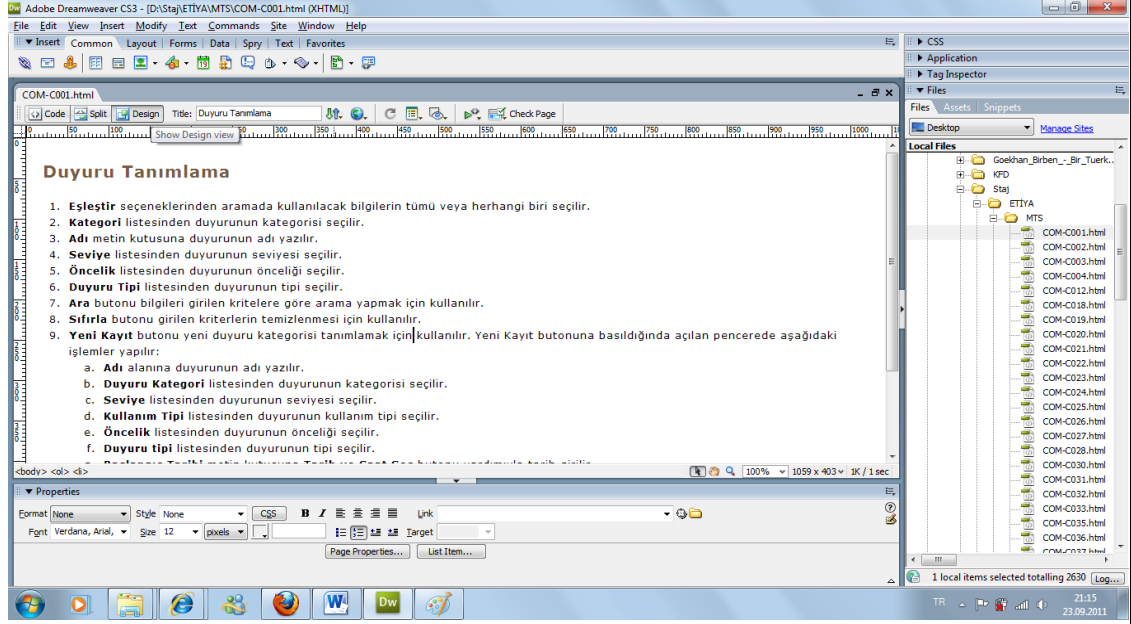
Adobe Dreamweaver HTML ve/veya PHP ile çalışmak isteyen web programcılarının programlama konusunda büyük ölçüde kolaylık sağlayan, ayrıca web sitelerindeki görselliği sağlayabilmek konusunda da tasarımcılara oldukça fazla seçeneği tek bir butonla sunan bir programdır. İçerisinde HTML, PHP ve JavaScript gibi web programlama dillerinin derlenebileceği gibi CSS gibi web sitelerine görsellik kazandıran dosyalar da hazırlanabilir. Bu program programcılara yazdıkları web programını farklı tarayıcılarda çalıştırabilme imkanını sunar. Dreamweaverda çalışırken aşağıda gösterildiği gibi HTML kodunu aynı zamanda hem kod, hem dizayn, hem de ikisini birden görebilme ve değiştirebilme imkanına sahip oluruz.



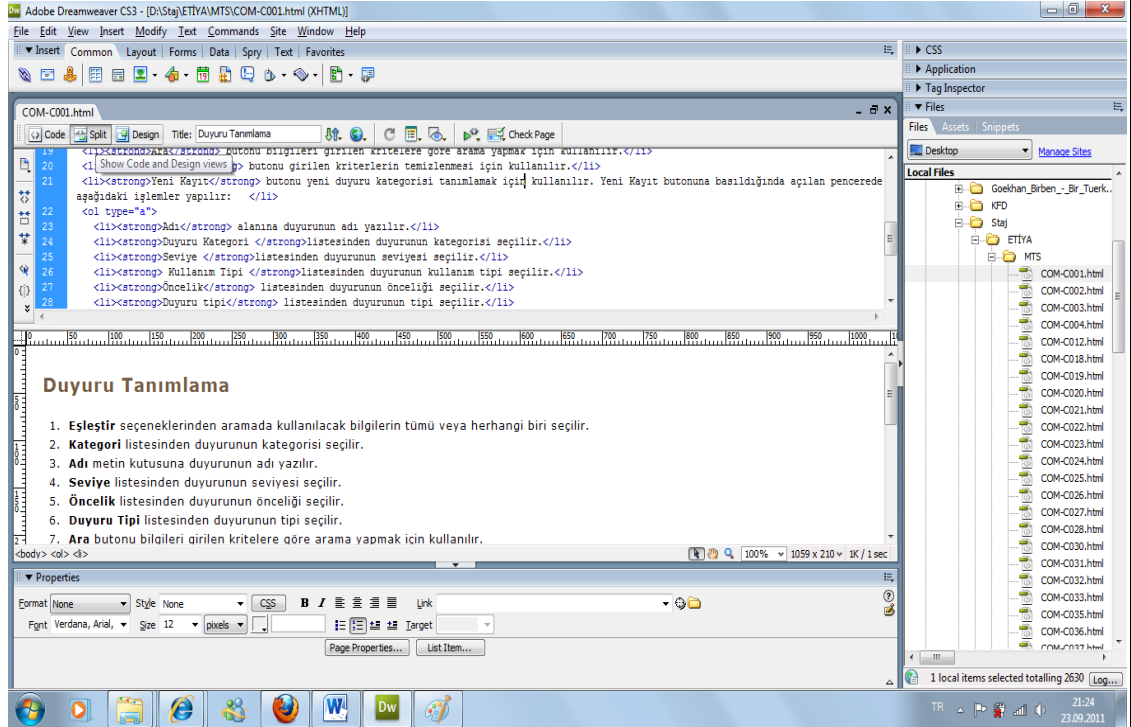
Şekil 1. Kod Kısmı

## Yapılan iş:

### Şirkete Ait Bir Projenin Yardım Dosyalarının Hazırlanması



Şekil 2. Dizayn Kısmı



Şekil 3. Kod ve Dizayn Kısmı



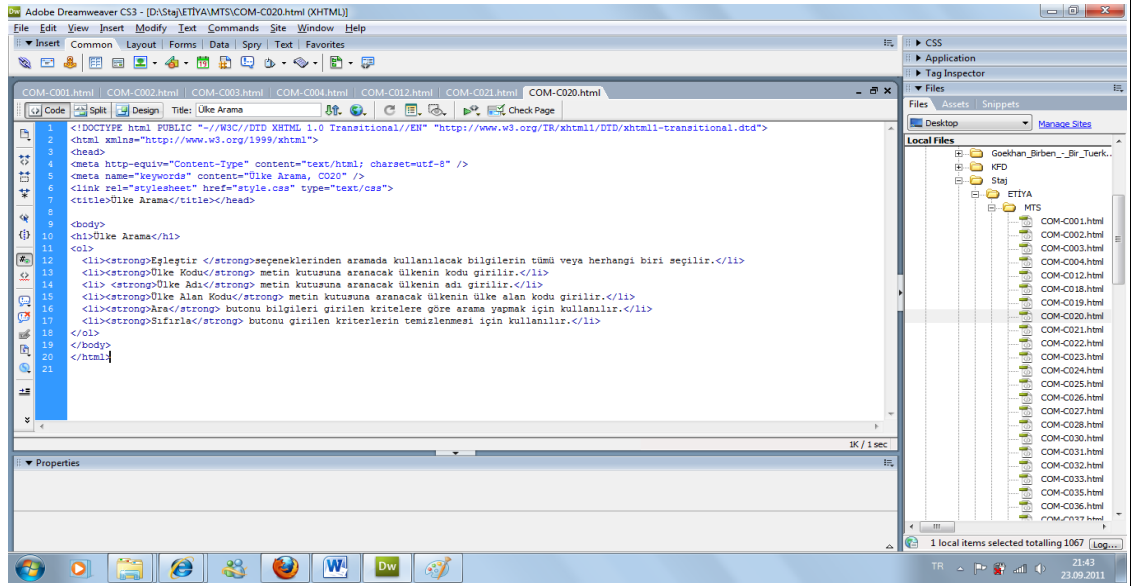
## Yapılan iş:

Şirkete Ait Bir Projenin Yardım Dosyalarının Hazırlanması

### HTML (HyperText Markup Language)

HTML web sitelerinin hazırlanması için kullanılan en temel web programlama dilidir. HTML ile statik web siteleri tasarlanabilir. Ancak dinamik web siteleri için yeterli bir dil değildir. Oldukça eski bir teknolojidir.

Stajda hazırladığım help (yardım) dosyalarını HTML dilinde hazırladım. Bunlardan bir örnek aşağıdaki ekran görüntüsüyle birlikte görülmektedir.



Şekil 4. Örnek HTML

Aynı zamanda HTML kodları (tagleri vs..) JSP dosyalarının içinde de kullanılabilir.

**Yapılan iş:**

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

Etiya Bilgi Teknolojileri şirketinin müşteri destek sayfasını hazırlama projesinde JSP web programlama dili kullanılmıştır. Projede MySQL veritabanı kullanılmış olup, proje NetBeans ortamında oluşturulup geliştirilmiştir. Öncelikle bu teknolojiler hakkında genel bilgiler verilip daha sonra sitenin sayfaları kısaca açıklanacaktır.

## JSP (Java Server Pages)

JSP, Javanın internet sayfalarını programlayabilmek için kullanılan bir alt dilidir. Nesneye dayalı (object oriented) özelliği olup, programcılara profesyonel kodlama imkanı sağlamakla beraber, Java'da olduğu gibi programcıyı nesneye dayalı programlar yazmaya zorlamaz. Yani programcı isterse C dilinde olduğu gibi nesne kullanmadan da program yazabilir.

JSP çok yeni bir teknoloji olmamakla beraber günümüzde halen oldukça yaygın olarak kullanılmaktadır. JSP diline ait temel birkaç konulara göz atacak olursak;

**JSP'de değişkenler:** Tanımlanmış bir değişkene aynı jsp dosyası içerisinde erişim sağlanabilir, bu değişken üzerinde değişiklik yapılabilir. Aynı zamanda tanımlandığı jsp içerisinde farklı bir jspdeki fonksiyona değişken olarak yollanıp burada kullanılabilir ve/veya değiştirilebilir. Aşağıda birkaç JSP değişken tipleri gösterilmiştir.

**int** - tamsayı tipinden değişkenleri belirtmekte kullanılır.

**String** - katar (kelime, cümle vb.) tipinden değişkenleri belirtmekte kullanılır.

**boolean** - bool değerlerini (lojik 1 veya 0) belirtmekte kullanılır.

**JSP'de fonksiyonlar:** JSP'de fonksiyonların aynı JSP dosyasının içinden çağırılabilceği gibi farklı JSP'ler arası fonksiyon çağrımları da söz konusudur. Fonksiyonlar kodun herhangi bir kısmında yer alabilirler. Geri dönüş değeri ve aldığı parametreler fonksiyonun tanımlandığı yerlerde açıkça belirtilmelidir. Birkaç örnek aşağıda gösterilmiştir.

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

```
<%!  
public boolean giris_Yapildimi (Object id) {  
    if (id==null)  
        return false;  
    else  
        return true;  
    }  
%>
```

### Şekil 5. Fonksiyon Örneği

Bu örnekte geri dönüş değeri boolean olan ve aldığı parametre Object tipinde olan bir fonksiyon tanımlanmıştır. Bu fonksiyon eğer aldığı id null ise false, aksi halde true değer döndürür. Bu benim hazırladığım sayfalarda kullanıcının giriş yapıp yapmadığını kontrol amaçlı kullanılabilir olan bir fonksiyondur.

```
<%@ page import="java.sql.*" %>  
<%@ include file="anasayfa.jsp" %>
```

### Şekil 6. JSP dosyası içermek

Burada da farklı bir JSP dosyasındaki fonksiyonu kullanabilmek için gerekli kod gözükmektedir. Görüldüğü gibi anasayfa.jsp include komutuyla bulunulan sayfaya değişkenleri ve fonksiyonlarıyla birlikte eklenmiştir. Veritabanı bağlantısı için kullanılacak olan fonksiyonlar ise page import komutu ile eklenebilmiştir.

```
out.print("<br>Sorunuza henüz cevap verilmemiştir.");
```

### Şekil 7. Print fonksiyonu

JSP’de fonksiyonlardan bahsedince en temel fonksiyonlarından olan print fonksiyonuna da bir örnek vermeden geçilemez. JSP’de out.print fonksiyonu içerisine yazılan değerleri veya değişkenleri ekrana yazar. Aynı zamanda içerisine HTML taglerini de yazabiliriz. Burada <br> bir HTML tagidir ve satır atlama işlemini gerçekleştirir.

**JSP’de kontrol yapıları:** çoğu programlama dilinde olduğu gibi **for**, **while**, **if** gibi komutlarla sağlanır.

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

**JSP’de Session ve Request:** JSP dilinin web programcılarına sunduğu bir özellik olan session kavramı, web sitesindeki erişim haklarını belirlemeyi sağlar. Yani bir ziyaretçi web sitesindeki bazı sayfaları açıp bazılarını açamazken, ziyaretçinin açmadığı bazı sayfaları web sitesine kayıtlı olan bir kullanıcı açabilir. Ayrıca sayfalar arasında dolaşırken herhangi bir kullanıcının oturum bilgileri de aktarılır ve bu bilgiler vasıtasıyla kimin siteye ne zaman giriş yaptığı, hangi sayfaları dolaştığı gibi bilgiler de kolaylıkla elde edilebilir. Aşağıda session kullanımına ait bir örnek gösterilmektedir.

```
if(Sifree.equals(kullanici_sifre)){
    session.setAttribute("ID", rs.getInt("Id"));
    session.setAttribute("AD", rs.getString("Ad"));
    session.setAttribute("SOYAD", rs.getString("Soyad"));
    response.sendRedirect("anasayfa.jsp");
}
```

Şekil 8. Session oturumu açma

```
<% if(session.getAttribute("ID")==null)
    response.sendRedirect("index.jsp");
%>
```

Şekil 9. Session oturum kontrolü

```
<%
    session.removeAttribute("AD");
    session.removeAttribute("SOYAD");
    session.removeAttribute("ID");
%>
```

Şekil 10. Session oturum kapatma

Şekil 8’de üye girişi yapan bir kullanıcının session oturumunu başlatan kodlar bulunurken, şekil 9’da dolaştığı sayfalarda üye girişinin yapıp yapılmadığı ve eğer yapılmadıysa giriş sayfasına yönlendirildiği kodlar bulunmaktadır. Ayrıca şekil 10’da kullanıcının çıkış yaptığı zaman oturumunun kapatılması için gerekli kodlar görülmektedir.

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

Request kullanımına bir örnek olarak;

```
<form name="input" method="post" action="uyeol.jsp">
  <div class="holder" align="center">
    Adi      :<input type="text" name="adi" title="Adini Yaz" maxlength="40" width="150" height="20" /> <br><br>
    Soyadi   :<input type="text" name="soyadi" title="Soyadini Yaz" maxlength="40" width="150" height="20" /> <br><br>
    Sifre    :<input type="password" name="sifre" title="Sifreni Yaz" maxlength="16" width="150" height="20" /> <br><br>
    Kisi-Kurum:<input type="text" value="kişi için 1; kurum için 0 giriniz" name="kisi-kurum" title="Kurum adini Yaz" maxlength="70"
    E-mail   : <input type="text" name="e-mail" title="Mail adresini Yaz" maxlength="60" width="150" height="20" /><br><br>
    Adres    : <input type="text" name="adres" title="Adresini Yaz" maxlength="60" width="150" height="20" /> <br><br>
    Üye adi  :<input type="text" name="uyeadı" title="Üye Adini Yaz" maxlength="40" width="150" height="20" /> <br><br>
    <br> <input type="submit" value="Kayıt Ol" /> <br><br>
  </div>
</form>
```

Şekil 11. Form örneği

```
String ad = request.getParameter("adi");
String soyad = request.getParameter("soyadi");
String kul_Ad = request.getParameter("uyeadı");
String sifre = request.getParameter("sifre");
String email = request.getParameter("e-mail");
String kurum_Kisi = request.getParameter("kisi-kurum");
String adres = request.getParameter("adres");
```

Şekil 12. Request sorgusu (uyeol.jsp)

Formdan name parametresinde belirtildiği adıyla gelen değişkenlerin gönderildiği JSP dosyasında request'ten alınması yukarıdaki kodlarda görüldüğü gibidir. Request'ten sorgulama request.getParameter fonksiyonu yardımıyla yapılabilmektedir.

JSP'de request ve session kavramları oldukça geniş olmasına rağmen, projemde kullandığım kısmı az olduğu için detaya inmemeyi tercih ediyorum. Yoksa request'in kendi içinde tanımlı birçok özel fonksiyonu bulunmaktadır. Ayrıca programcılar bu fonksiyonlardan yararlanırken JSP'nin nesneye dayalı (object oriented) olmasından faydalanmaktadır.

### Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

**JSP – Veritabanı Bağlantısı:** JSP veritabanına bağlanmayı JDBC Driver ile sağlamaktadır. JSP kodları vasıtasıyla veritabanında yeni kayıtlar eklenebilir, varolanlar güncellenebilir veya silinebilir. JSP dosyalarında veritabanı bağlantısını gerçekleştirebilmek için java.sql sayfası bağlantısının gerçekleşmek istendiği tüm sayfalarda import komutuyla eklenmiş olmalıdır. JSP ile programlama yaparken veritabanı kullanımına ilişkin birkaç örnek aşağıda verilmiştir.

```
String connectionURL = "jdbc:mysql://localhost:3306/etiya";
Connection connection = null;
Statement statement = null;
ResultSet rs1 = null;
```

#### Şekil 13. Veritabanı için temel tanımlamalar

Burada connectionURL olarak görülen string tipinden değişken: kullanılan driver (sürücü), kullanılan sql tipi ve veritabanının bulunduğu yer ve adını içermektedir. Ayrıca veritabanı bağlantısını gerçekleyebilmek için gerekli olan Connection, Statement ve ResultSet nesnelere yaratılmış olup başlangıçta null değere atanmaktadır.

```
Class.forName("com.mysql.jdbc.Driver").newInstance();
connection = DriverManager.getConnection(connectionURL, "root", "");
statement = connection.createStatement();
```

#### Şekil 14. Veritabanı bağlantısı oluşturma

Buradaki ilk satır ile yeni bir bağlantı oluşturulmuştur. Ayrıca connection değişkenine yukarıda belirtilen connectionURL, veritabanı kullanıcı adı ve şifresinden elde edilen bir bağlantı atanmaktadır. Daha sonra ise bu connection değişkeni bir statement oluşturmuş olup bu statement değişkenine atanmaktadır. Bundan sonra veritabanı bağlantısı gerçekleşmiştir. Veritabanından sorguları statement değişkeni vasıtasıyla yapabilmekteyiz.

**Yapılan iş:**

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

```
rs1 = statement.executeQuery("SELECT * FROM musteri_tablosu");
```

**Şekil 15. Veritabanı sorgusu**

Burada önceden tanımlanmış olan ResultSet nesnesine veritabanından yapılan sorgu aktarılmaktadır. Veritabanındaki tablonun ismi bu örnekte müşteri\_tablosu'dur. Eğer farklı tablodan sorgu yapılacaksa bu isim değiştirilmelidir.

```
rs = statement.executeQuery("SELECT * FROM musteri_tablosu WHERE Id='"+session.getAttribute("ID")+"'");
```

**Şekil 16. Veritabanından değışkene göre sorgulama**

Burada da yine önceden tanımlanmış bir ResultSet nesnesine veritabanı sorgusu atanmaktadır. Farklı olarak burada musteri\_tablosundaki sorgulamada bir de şart belirtilmektedir. Tablodaki Id değerleri oturumu açan kullanıcının ID'si ile aynı olması gerekmektedir. Bunu yukarıda görülen kod sayesinde yapabilmekteyiz. "WHERE" komutu veritabanından koşullu aramayı programcılara sunmaktadır. Burada "session.getAttribute("ID")" komutundan önceki "+" lar aranacak kriterin yine bir değışkene bağlı olduğunu göstermektedir. Sadece bir sabit bir stringe göre arama yapacak olsaydık bunlar kullanılmayacaktı.

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

## MySQL

MySQL sql tipi bir veritabanıdır. MySQL'de veritabanı oluşturabilmek için öncelikle MySQL'in bilgisayara kurulmuş olması gereklidir. Kurulum işlemi gerçekleştirilmiş ise <http://localhost/phpmyadmin/> adresinden kendi bilgisayarımıza veritabanı kurabilir, bunlarla çalışabiliriz. Bu açılan sayfada SQL sekmesini seçersek, "localhost" sunucusu üzerinde SQL sorgusunu/sorgularını çalıştır gibi bir yazı çıkacak ve buraya gerekli SQL kodunu yazacağız. Projede kullandığım veritabanını "create database etiya" koduyla oluşturulmuştur. Daha sonra bu veritabanının altında değişik tablolar oluşturulmuştur. Bunların oluşturulma kodlarından birkaç örnek aşağıda gösterilmiştir.

```
etiya veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ?  
CREATE TABLE IF NOT EXISTS `musteri_tablosu` (  
  `Id` int(10) NOT NULL AUTO_INCREMENT,  
  `Ad` varchar(15) NOT NULL,  
  `Soyad` varchar(15) NOT NULL,  
  `Kul_Ad` varchar(25) NOT NULL,  
  `Sifre` varchar(21) NOT NULL,  
  `Email` varchar(1000) NOT NULL,  
  `Statu` int(10) NOT NULL,  
  `Kurum_kisi` int(10) NOT NULL,  
  `Adres` varchar(1000) NOT NULL,  
  PRIMARY KEY (`Id`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=3300 ;
```

Şekil 17. Müşteri tablosunun oluşturulması

```
etiya veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ?  
CREATE TABLE IF NOT EXISTS `konu_tablosu` (  
  `id` int(11) NOT NULL DEFAULT '0',  
  `aciklama` varchar(200) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Şekil 18. Konu tablosunun oluşturulması



### Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

```
etiya veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ⓘ —  
CREATE TABLE IF NOT EXISTS `problem_tablosu` (  
  `Id` int(11) DEFAULT NULL,  
  `Musteri_Id` int(11) DEFAULT NULL,  
  `Statu` int(11) DEFAULT NULL,  
  `Aciliyet_durumu` int(11) DEFAULT NULL,  
  `Konu_Id` int(11) DEFAULT NULL,  
  `Konu` varchar(1000) DEFAULT NULL,  
  `Soru` varchar(1000) DEFAULT NULL,  
  `Cevap` varchar(1000) DEFAULT NULL,  
  `GirisTarihi` date DEFAULT NULL,  
  `CevapTarihi` date DEFAULT NULL  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Şekil 19. Problem tablosunun oluşturulması

```
etiya veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ⓘ —  
CREATE TABLE IF NOT EXISTS `duyurular_tablosu` (  
  `id` int(11) NOT NULL DEFAULT '0',  
  `baslik` varchar(20) NOT NULL,  
  `aciklama` varchar(1000) NOT NULL,  
  `statu` int(2) DEFAULT NULL,  
  `tarih` date DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Şekil 20. Duyuru tablosunun oluşturulması

Tablolara doğrudan localhosttan ekleme yapılabileceği gibi önceki konularda değinildiği gibi JSP kodlarıyla da ekleme düzenleme vb. işlemler gerçekleştirilebilir. Localhosttan ekleme işlemi için SQL sorgusu da çalıştırılabilir veya doğrudan tablodaki bilgiler metin kutularına bir bir elle girilebilir. Aşağıda bunlara ait örnekler ekran görüntüleriyle verilmiştir. Bunlara ek olarak oluşturulmuş olan tablolardan da örnekler gösterilmektedir.

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

```
etiya veritabanı üzerinde SQL sorgusunu/sorgularını çalıştır: ?  
INSERT INTO 'musteri_tablosu' ('Id', 'Ad', 'Soyad', 'Kul_Ad', 'Sifre', 'Email', 'Kurum_kisi', 'Adres') VALUES  
(3287, 'Fatih', 'Ata', 'fata', '432', 'fatih.ata@gmail.com', 0, 'Şişli/İstanbul');
```

Şekil 21. Veritabanına ekleme-1

localhost ▶ etiya ▶ muster\_i\_tablosu

Gözet Yapı SQL Ara Ekle Dışa Aktar İçe Aktar İşlemler Boşalt Kaldır

Alan	Türü	İşlev	Boş	Değer
Id	int(10)			3553
Ad	varchar(15)			Hasan
Soyad	varchar(15)			Yıldırım
Kul_Ad	varchar(25)			yildirim
Sifre	varchar(21)			3553
Email	varchar(1000)			h.yildirim@itu.edu.tr
Kurum_kisi	int(10)			1
Adres	varchar(1000)			itü ayazağa

Git

Şekil 22. Veritabanına ekleme-2

←T→	Id	Ad	Soyad	Kul_Ad	Sifre	Email	Kurum_kisi	Adres
<input type="checkbox"/>	3553	Hasan	Yıldırım	yildirim	3553	h.yildirim@itu.edu.tr	1	itü ayazağa
<input type="checkbox"/>	3287	Fatih	Ata	fata	432	fatih.ata@gmail.com	0	Şişli/İstanbul
<input type="checkbox"/>	3303	Abdullah	Aydeger	aydeger	1234	aydeger@itu.edu.tr	1	Fatih/Ist.
<input type="checkbox"/>	3304	Mehmet	Ercomert	erco	4321	ercomert@hotmail.com	0	Besiktas/Ist.
<input type="checkbox"/>	3292	Ediz	Göksu	goksu	asdf	ediz.goksu@etiya.com	1	ytü davutpaşa

Şekil 23. Tablolardan örnek-1

←T→	Id	Musteri_Id	Statu	Aciliyet_durumu	Konu_Id	Konu	Soru	Cevap	GirisTarihi	CevapTarihi
<input type="checkbox"/>	0	3292	1	1	1	Teno	CEVAP	cevap	2011-09-14	2011-09-30
<input type="checkbox"/>	1	3292	1	1	1	ETIYA	CEVAP	cevap	2011-09-14	2011-09-30

Şekil 24. Tablolardan örnek-2

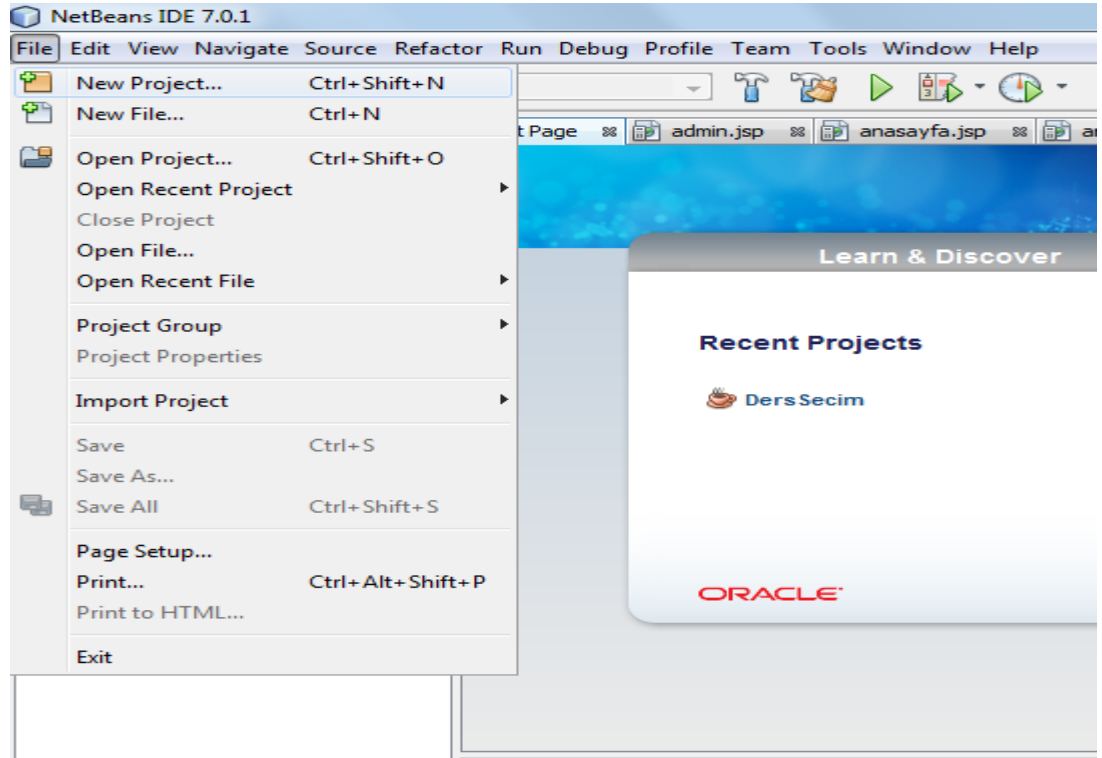
## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

## NetBeans

NetBeans C, C++ ve Java gibi programlama dillerinde yazılan kodların yazılıp derlenebileceği bir platformdur. Masaüstü uygulamalarının yazılabileceği gibi web uygulamalarının da yazılabileceği ve programcıya birçok kolaylık sağlayan bir IDE'dir.

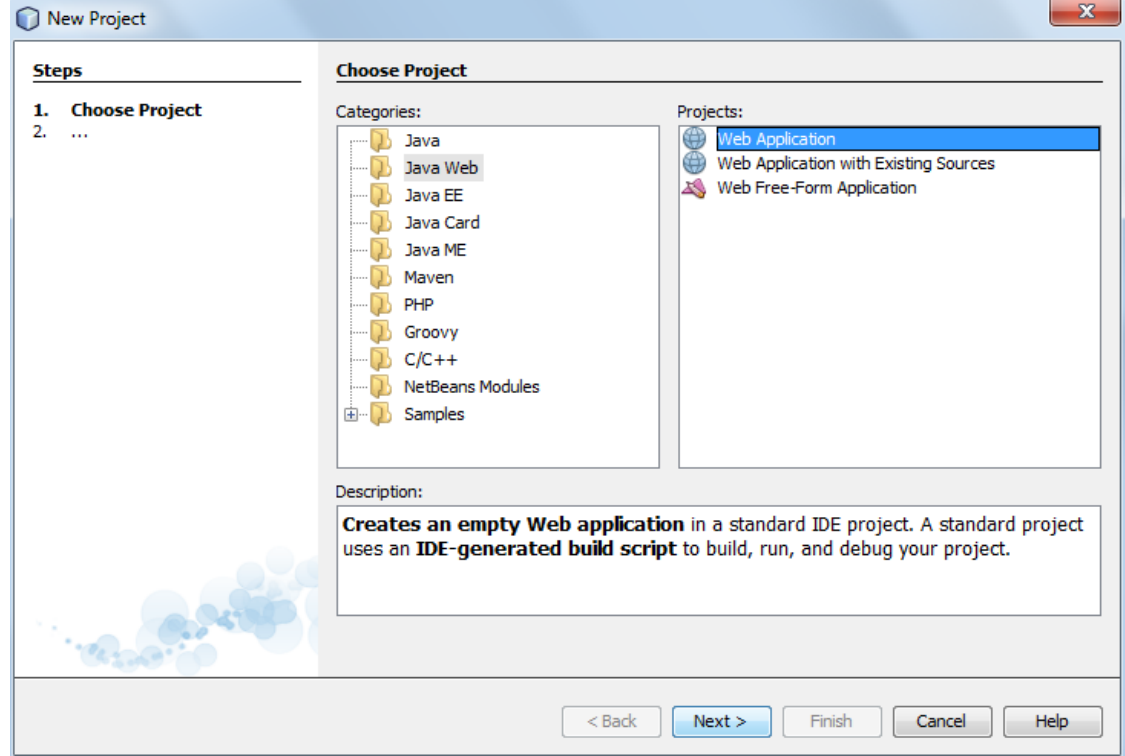
Projemi geliştirirken JSP dosyalarını NetBeans'te oluşturup çalıştırdım. Öncelikle NetBeans'te bir web uygulamasının nasıl açılacağını aşağıdaki ekran görüntüleriyle anlatacağım. Görüntülerdeki adımlar sırasıyla uygulanmalıdır.



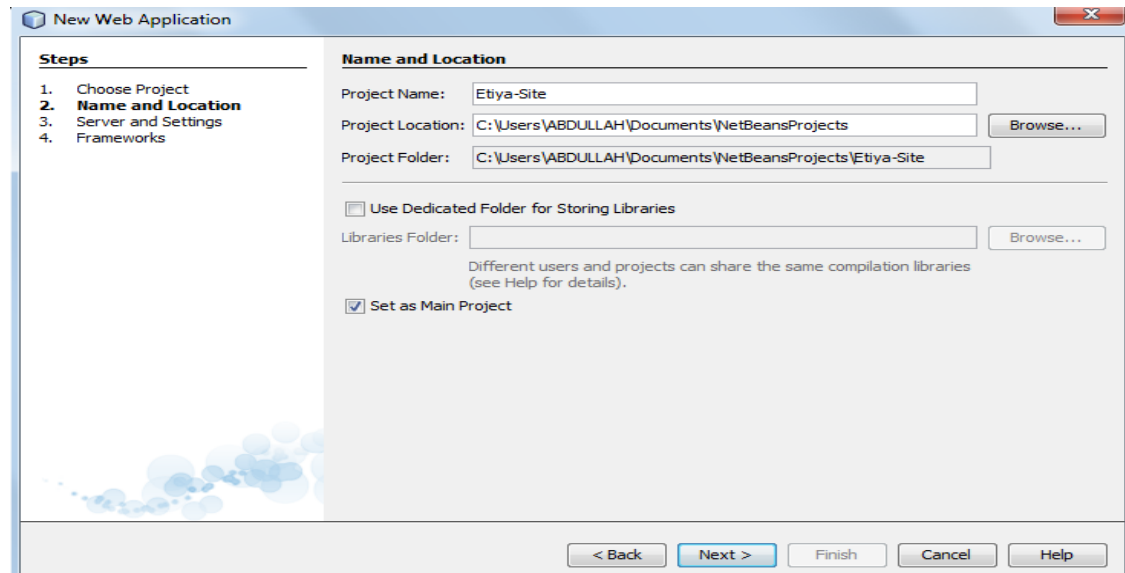
Şekil 25. NetBeans web projesi oluşturma-1

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması



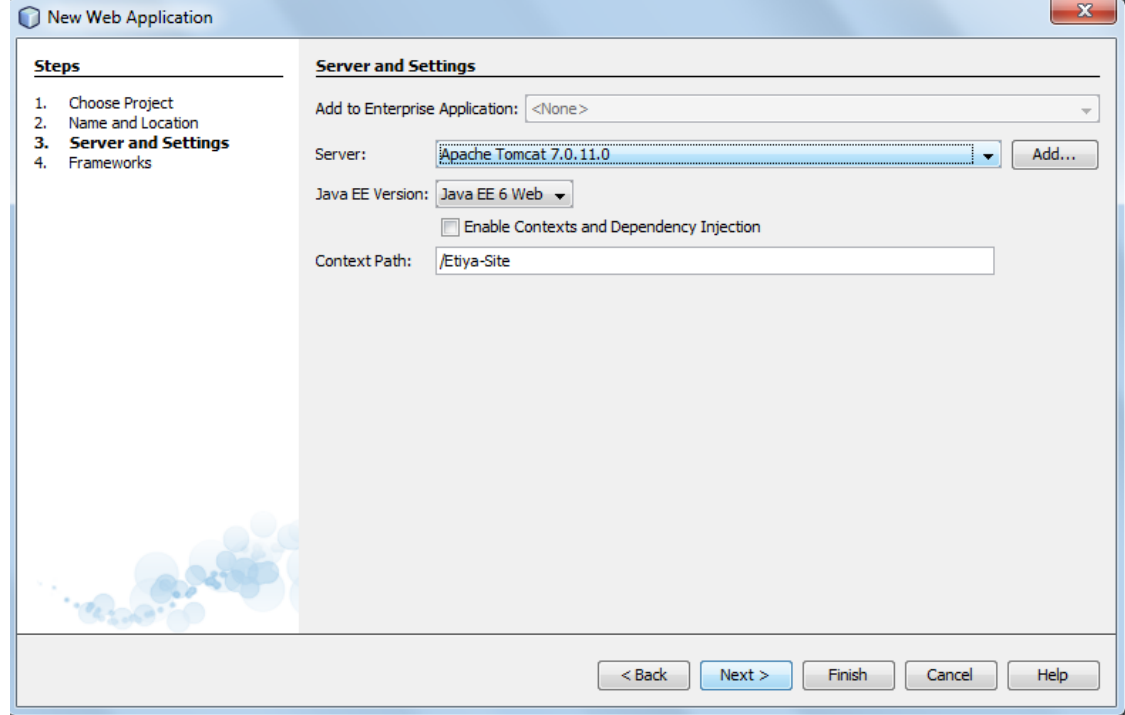
Şekil 26. NetBeans web projesi oluşturma-2



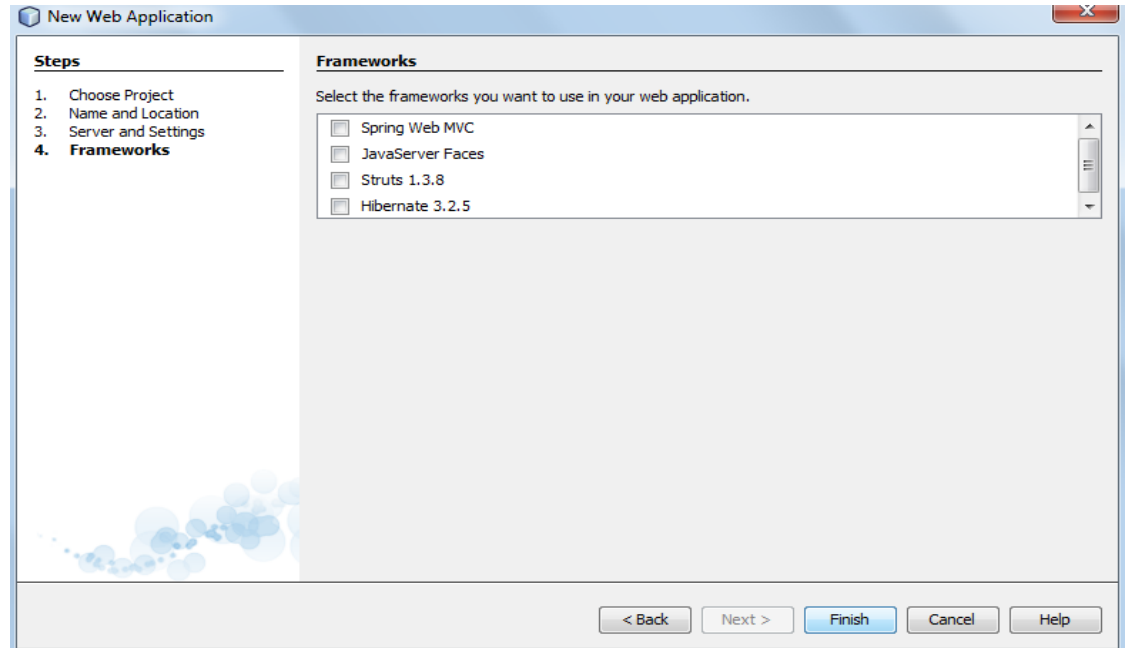
Şekil 27. Netbeans web projesi oluşturma-3

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması



Şekil 28. Netbeans web projesi oluşturma-4



Şekil 29. NetBeans web projesi oluşturma-5

**Yapılan iş:**

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

Ekran görüntülerini açıklamak gerekirse; Şekil .25’de yeni bir proje oluşturmaya başlıyoruz. Şekil.26’da projeyi web application şeklinde açıyoruz. Daha sonra şekil.27’de projeye bir isim verip bulundurmak istediğimiz klasörü yazıyoruz. Şekil.28’de web projesinin çalıştırılacağı server seçiliyor. Burada Tomcat önceden yüklü olduğu için onu seçtik, ancak NetBeans’in kendi server’ı olan GlassFish’de seçilebilirdi. Şekil.29’da ise eğer herhangi bir FrameWork kullanılacaksa o seçilir, aksi halde hiçbiri seçilmeyip proje oluşturulur.

## Müşteri Destek Sistemi

### *Kullanıcı Kısmı*

Stajda tasarladığım müşteri destek sisteminde yukarıda anlattığım gibi JSP dilini MySQL veritabanıyla birlikte kullandım. JSP kodlarını NetBeans ortamında derledim. Şimdi hazırladığım siteye ait birkaç ekran görüntüsüyle sitede yapılabileceklerle değinelim.

ETİYA  
- Müşteri Destek Sayfası -

Anasayfa Bilgilerim Taleplerim Çıkış

Kullanıcı Giriş Sayfası

Kullanıcı Adı :  Şifre :  Giriş Yap

[Üye Ol](#)

Şekil 30. MDS - Giriş Sayfası

**Yapılan iş:**

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

Projeyi çalıştırdığımızda ilk olarak index.jsp sayfasıyla karşılaşırız. Bu sayfada kullanıcı giriş yapması beklenmekte ve eğer üye değilse üye ol linkinden siteye kayıt olması gerekmektedir. Yukarıda görülen Anasayfa – Bilgilerim – Taleplerim – Çıkış sayfalarına da eğer oturum açmadan gitmeye çalışırsa bu sayfalar herhangi bir oturumun açık olup olmadığını kontrol edip eğer oturum açılmamışsa tekrar index.jsp sayfasını açacaktır. Bu sayede kullanıcı giriş yapmadan veya üye olmadan siteyi dolaşamayacaktır.

Kullanıcının kayıtlı olmadığı varsayıp üye ol'a tıklamasıyla aşağıda görülen sayfa açılacaktır. Bu sayfada kullanıcı gerekli bilgileri girip üye ol'a tıkladığında veritabanında yeni bir kayıt oluşturulacaktır.

ETİYA

- Müşteri Destek Sayfası -

Anasayfa

Bilgilerim

Taleplerim

Çıkış

Üye Kayıt Sayfası

Adi :

Soyadi :

Sifre :

Kisi-Kurum: kişi için 1; kurum için 0 gi

E-mail :

Adres :

Üye adı :

Kayıt Ol

Şekil 31. MDS - Üye olma sayfası

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

The screenshot shows the 'etiya' logo and navigation menu with options: Anasayfa, Bilgilerim, Taleplerim, Çıkış. Below the logo, it says '- Müşteri Destek Sayfası -'. The main content area is titled 'Merhaba Ediz Göksu' and contains a support request form with the following fields: 'Durum' (Status) set to 'Önemsiz', 'Konu' (Topic) set to 'Internet', and an empty 'Açıklama' (Description) field. A 'Gönder' (Send) button is present. Below the form, there are two columns: 'Son 10 Talebiniz' (Your last 10 requests) with links for 'Talep 1 Teno' and 'Talep 2 ETIYA', and 'Duyurular' (Announcements) with links for 'Duyuru 1: Cep Tel', 'Duyuru 2: BBB', and 'Duyuru 3: cCC'.

Şekil 32. MDS - Üye Ana Sayfası

Kullanıcı giriş yaptıktan sonra anasayfaya gidilecek ve bu sayfada destek talebi gönderebileceği bir form olacak. Aynı zamanda site yöneticilerinin yaptığı duyurular ve kullanıcının son 10 talebi bu anasayfada kayan yazı şeklinde görülecektir. Ayrıca bu kayan yazılara tıklanırsa duyuru veya taleplerin ayrıntısı görülebilecektir.

The screenshot shows the details of an announcement. On the left, under 'Son 10 Talebiniz', there are links for 'Talep 1 Teno' and 'Talep 2 ETIYA'. On the right, under 'Duyurular', the details for 'Duyuru 1: Cep Tel' are shown: 'Duyurunun başlığı: Cep Tel', 'Duyuruya ait açıklama: AAAA', and 'Duyurunun statusu: 1'.

Şekil 33. MDS - Duyuru Detayı Gösterme



### Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

ETİYA  
- Müşteri Destek Sayfası -

Anasayfa Bilgilerim Taleplerim Çıkış

Ediz Göksu

Adınız: Ediz

Soyadınız: Göksu

Kullanıcı Adınız: goksu

Şifreniz: asdf

Kurum-kisi: 1

E-mail: ediz.goksu@etiya.com

Adres: ytü davutpaşa

Güncelle

Şekil 34. MDS – Bilgilerim

Kullanıcı eğer Bilgilerim sekmesine tıklarsa kendi özel bilgilerini görebilecek ve eğer isterse bunları güncelle butonu vasıtasıyla veritabanında anında güncelleyebilecek.

ETİYA  
- Müşteri Destek Sayfası -

Anasayfa Bilgilerim Taleplerim Çıkış

Ediz Göksu

Açık Talepleriniz

Kapalı Talepleriniz

Açık talebiniz bulunmamaktadır.

Talep 1: Teno

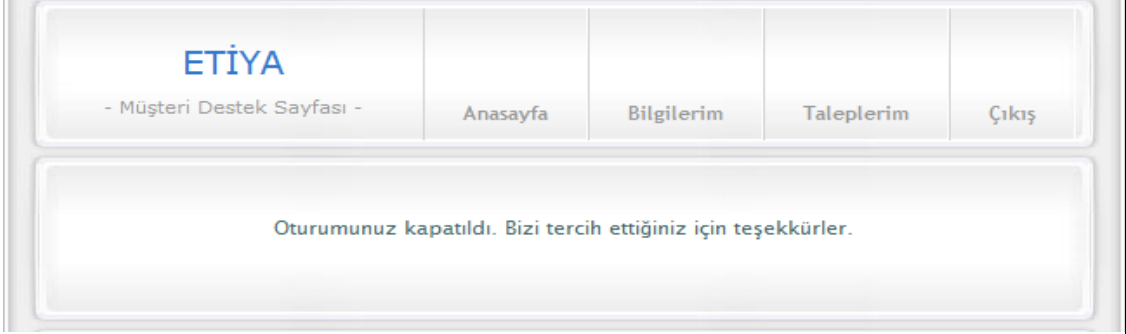
Talep 2: ETİYA

Şekil 35. MDS - Taleplerim

### Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

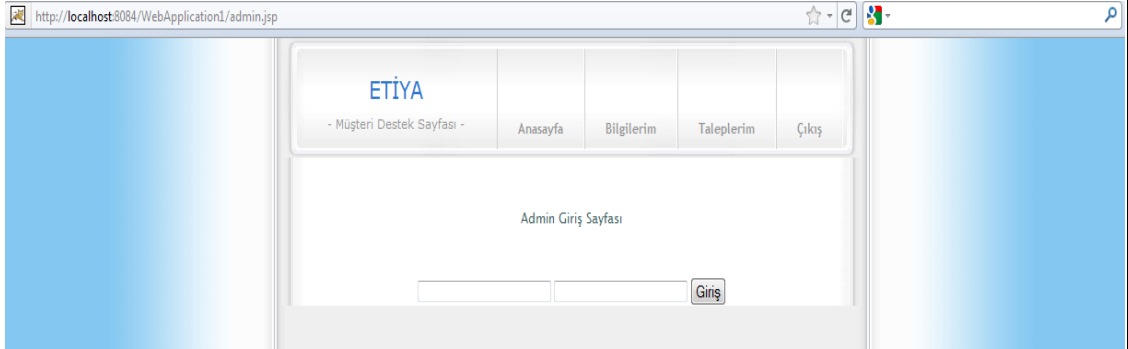
Kullanıcı eğer taleplerim sekmesine tıklarsa kendi açık ve kapalı taleplerini kayan yazı şeklinde görecektir. Bu taleplere tıklarsa tıklanan talebin detayını görebilecektir.



### Şekil 36. MDS - Kullanıcı Çıkış

Kullanıcı eğer çıkış sekmesine tıklarsa oturumu kapatılacak ve bir daha yeniden giriş yapana dek sitede dolaşamayacak.

### *Admin (Yönetici) Kısmı*



### Şekil 37. MDS - Admin Girişi

Sitenin bir de admin girişi için ayrı bir kısmı var. Burada admin.jsp yi özel olarak elle çalıştırmamız gerekiyor. Bu sitenin güvenliği için tercih edildi. Site yöneticisi bunu bilecek ancak diğer kullanıcılar bundan habersiz olacaktır. Burada açılan sayfadaki metin kutulardan ilkinde ETİYA, ikincisine TEKNOPARK yazılırsa admin girişi başarıyla yapılmış olacak. Bunlar benim atadığım değerler, değiştirilebilir.

### Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

Admin girişi yapılıncı tüm müşterilerin gözükeceği bir sayfaya yönlendirileceğiz.

ETİYA

- Müşteri Destek Sayfası -

Talepler

Müşteriler

Çıkış

Toplam Müşteri Sayısı: 5

Toplam Açık Talep Sayısı: 0

Müşteri

Ara

Hasan Yıldırım itü ayazağa [Detay](#)

Fatih Ata Şişli/İstanbul [Detay](#)

Abdullah Aydeger Fatih/Ist. [Detay](#)

Mehmet Ercomert Besiktas/Ist. [Detay](#)

Ediz Göksu ytü davutpaşa [Detay](#)

### Şekil 38. MDS – Müşteriler

Açılan sayfada en başta toplam müşteri sayısı ve toplam açık talep sayısı gösterilecek. Açık talep henüz cevap verilmemiş olan talep anlamında olduğu için yönetici'nin bunu görüp bunlara cevap vermesi önemlidir.

Bu sayfada admin müşteri arayabilecek ve detay linkine tıkladığı zaman müşteriye ait bilgileri görebilecek. Ayrıca burada müşterinin bilgilerini de değiştirebilecek.

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması

The screenshot shows a web application interface for 'ETİYA'. The header includes the logo 'ETİYA' and the text '- Müşteri Destek Sayfası -'. Below the header, there are three navigation tabs: 'Talepler' (highlighted in blue), 'Müşteriler', and 'Çıkış'. The main content area displays the following information:

- Toplam Müşteri Sayısı: 5
- Toplam Açık Talep Sayısı: 0
- Müşteri:
- Durum:
- Konu:
- Talep Durum: Açık talepler için -1 Kap
- Ara:

**Şekil 39. MDS – Talepler**

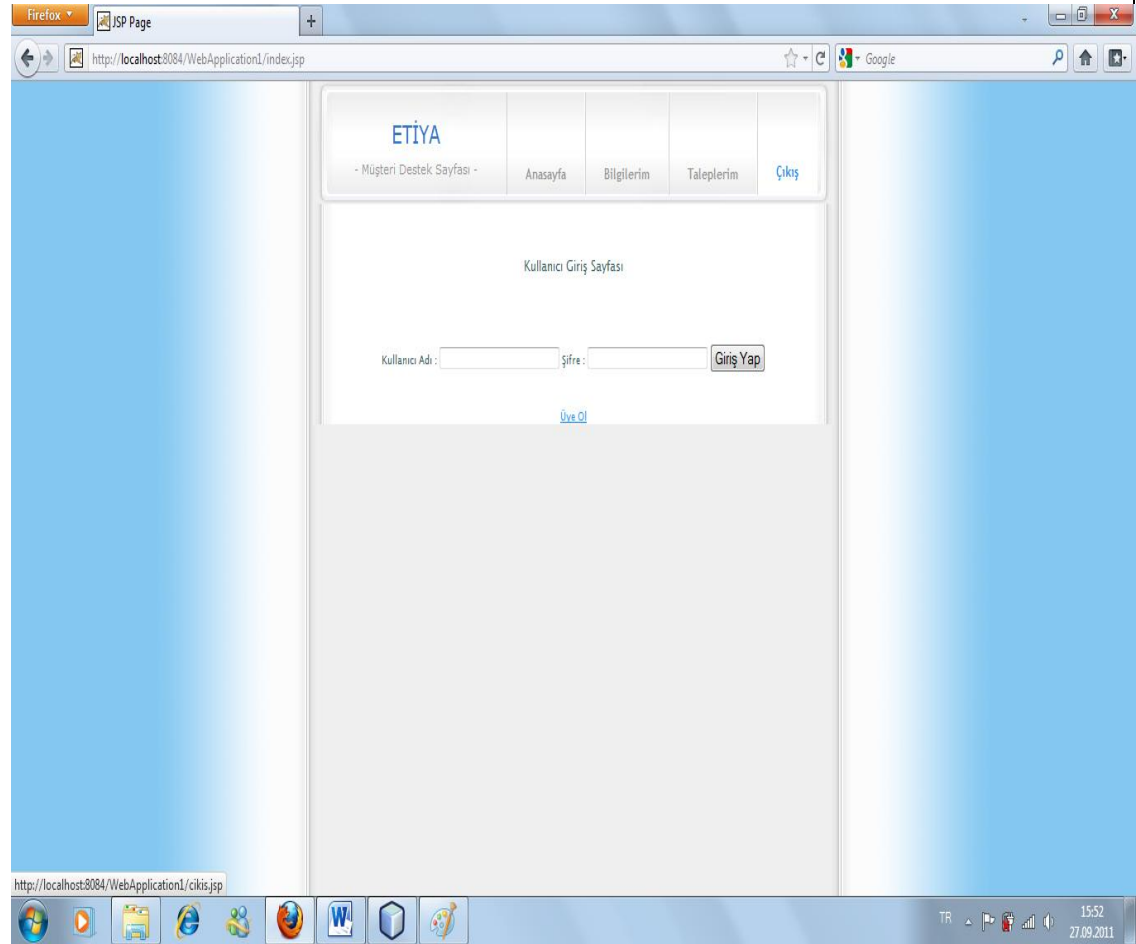
Yönetici talepler sekmesine tıkladığında talepleri arayabileceği bir formla karşılaşacak. Burada aramak istediği kriterleri girip Ara butonuna tıkladığında veya metin kutularının tamamını boş bırakıp Talep Durum metin kutusuna da 0 girerse tüm talepleri görebilecektir.

Burada arama yaparken girilen konu ve durum bilgileri de admin tarafından daha önceden oluşturulmuş veritabanı tablosundan seçilecektir. Çünkü müşteri talep gönderirken bu tablodaki verilere uygun olarak gönderebilmektedir.

Yönetici çıkış sekmesine tıkladığında ise normal kullanıcı çıkışı gibi oturum kapatılacak ve normal kullanıcı giriş sayfası açılacaktır.

## Yapılan iş:

JSP (Java Server Pages) ile müşteri destek sitesini tasarlamak ve gerekli veritabanlarının oluşturulması ve kullanılması



Şekil 40. MDS - Admin Çıkış

İş Yeri

Kurum:

Bölüm:

Stajyer Öğrenci

Adı:

Soyadı:

Yukarıda adı geçen öğrenci \_\_\_\_\_ tarihinden  
\_\_\_\_\_ tarihine kadar \_\_\_\_\_ net iş günü kurumumuzda İTÜ  
tarafından sigortalı olarak (Kurum tarafından yapılan sigorta değil) staj yapmıştır.

NOT I: Net iş günü hafta sonlarını, bayram günlerini, resmi tatil günlerini ve herhangi  
bir nedenle alınan izinleri içermez.

NOT II: Lütfen ciltlenmemiş Staj Raporunu imzalayıp ve damgalamayın.

(İTÜ Bilgisayar ve Bilişim Fakültesi Staj Kuralları: <http://www.bb.itu.edu.tr>).

Diğer konular :

\_\_\_\_ / \_\_\_\_ / 20\_\_

Bölüm Müdürü \_\_\_\_\_

İmza & Mühür

Kurum veya Bölüm Müdürünün:

Ünvanı

Adı & Soyadı