



**e-HASTANE SİSTEMLERİNİN İNCELENMESİ VE JAVA  
TEKNOLOJİLERİ İLE e-HASTANE UYGULAMASI  
GELİŞTİRİLMESİ**

**ANALYSIS OF e-HOSPITAL SYSTEMS AND DEVELOPING  
AN e-HOSPITAL APPLICATION USING JAVA  
TECHNOLOGIES**

**M. ÇAĞRI ELİBOL**

Başkent Üniversitesi  
Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin  
İSTATİSTİK ve BİLGİSAYAR Bilimleri Anabilim Dalı İçin Öngördüğü  
YÜKSEK LİSANS TEZİ  
olarak hazırlanmıştır.

2008

## KABUL VE ONAY SAYFASI

Fen Bilimleri Enstitüsü Müdürlüğü'ne,

Bu çalışma, jürimiz tarafından **İSTATİSTİK ve BİLGİSAYAR BİLİMLERİ ANABİLİM DALI** 'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Başkan :

Üye (Danışman) :

Üye :

## ONAY

Bu tez 18 / 01 / 2008 tarihinde, Enstitü Yönetim Kurulunca belirlenen yukarıdaki jüri üyeleri tarafından kabul edilmiştir.

/ / 2008

Prof.Dr. Emin AKATA

FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRÜ

## TEŐEKKÜR

Çalıőmalarım boyunca göstermiő olduđu sabır, anlayıő,yardımlarından ötürü ve ayrıca bilgi iletiőim alanında sahip olduđu tecrübeleri benimle paylaőtıđı için, danıőmanım olan Yrd. Doç. Dr. Güvenç Arslan'a teőekkürü bir borç bilirim.

Hayatım boyunca, içinde bulunduđum çalıőmalarda göstermiő olduđu destek ve ilgiden ötürü, annem H. Reyhan Elibol'a teőekkür ederim.

Ankara, Ocak, 2008

M. Çađrı ELİBOL

## ÖZ

### **e-HASTANE SİSTEMLERİNİN İNCELENMESİ VE JAVA TEKNOLOJİLERİ İLE e-HASTANE UYGULAMASI GELİŞTİRİLMESİ**

Mehmet Çağrı ELİBOL

Başkent Üniversitesi Fen Bilimleri Enstitüsü

İstatistik ve Bilgisayar Bilimleri Anabilim Dalı

Bir hastanenin amacı; muayene, teşhis ve tedavi için hastaneye başvuran hastaların ayakta veya yatırılarak tedavisi ile sağlık hizmetleriyle ilgili tüm alanlarda eğitim-öğretim, araştırma ve uygulama yapmak, her düzeyde yetkili tıp ve sağlık personeli yetiştirerek ve diğer kurumlarla işbirliği yaparak sağlık hizmetlerinin tam, kaliteli ve verimli düzeyde yürütülmesini sağlamaktır. Bu amacın işlemesi için en önemli aşama, hastanın hastaneye geldiği anda müracaat kaydının yapılmasıdır.

Bu çalışmada, mevcut hastane sistemleri incelenerek, Türkiye'nin ihtiyacı olan, hastaların, doktorların ve hastanede çalışan görevlilerinin beklentilerini sağlayacak, hızlı ve kolay erişilebilir, hizmet standartları yüksek, yeni teknolojilerden faydalanmış bir uygulama yazılımı geliştirmek hedeflenmiştir.

**Anahtar Kelimeler** : e-Hastane, e-Sağlık, Java, JPA, Bilişim, Teknoloji, Hastane

**Danışman** : Yrd. Doç. Dr. Güvenç ARSLAN, Başkent Üniversitesi, İstatistik ve Bilgisayar Bilimleri Bölümü

## **ABSTRACT**

### **ANALYSIS OF e-HOSPITAL SYSTEMS AND DEVELOPING AN e-HOSPITAL APPLICATION USING JAVA TECHNOLOGIES**

M. Cagri ELIBOL

Baskent University Institute of Science

Statistics and Computer Science Department

Purpose of a hospital is, to give a successful medical treatment to patients who visits the hospital for the purpose of a medical exam or to get a diagnosis, to research all the new aspects of health, to train the new personnel, to cooperate with other health institutions, to get the health services work better with more quality. The most important stage of this purpose to work is the registration process of a patient into the hospital's system.

In this study, with the help of analyzing the available hospital systems, developing a fast, high standard and easily accessible software application which Turkey needs and expectations of patients, doctors and the employee in the hospital are fulfilled, is aimed.

**Keywords:** e-Hospital, e-Health, Java, JPA, IT, Technology, Hospital.

**Supervisor:** Asst. Associate Prof. Guvenc ARSLAN, Baskent University, Statistics and Computer Science Department

## İÇİNDEKİLER DİZİNİ

TEŞEKKÜR .....	i
ÖZ .....	ii
ABSTRACT .....	iii
İÇİNDEKİLER DİZİNİ .....	iv
ŞEKİLLER DİZİNİ .....	vi
TABLolar DİZİNİ .....	viii
SİMGELER VE KISALTMALAR .....	ix
<b>1 GİRİŞ .....</b>	<b>1</b>
1.1 Genel Bakış .....	2
<b>2 SAĞLIKTA BİLİŞİM SİSTEMLERİ .....</b>	<b>3</b>
2.1 Sağlık Bilişiminde Kaliteli Veri .....	3
2.2 Sağlık Bilişiminde Hedefler .....	6
2.3 Sağlık Bilişiminde Mevcut Durum .....	8
2.4 Hastane İşleyiş Yapısı .....	11
2.5 Sağlık Bilişiminin Sorunları .....	14
2.6 Sağlık Bilişiminde Yapılan Çalışmalar .....	15
<b>3 JAVA PROGRAMLAMA DİLİ .....</b>	<b>18</b>
3.1 JPA .....	20
3.2 JPA Özellikleri .....	21
3.3 JPA Açıklayıcı Notları .....	22
<b>4 e-HASTANE SİSTEMİ .....</b>	<b>24</b>
4.1 Genel Yapı .....	25
4.2 Proje Dosyaları ve Paketleri .....	27
4.3 e-Hastane Yetki Sistemi .....	30
4.3.1. Yönetim .....	31

4.3.2.	Kullanıcı .....	31
4.3.3.	Doktor .....	31
4.4	e-Hastane Müracaat Kayıt Kabul Modülü.....	31
4.4.1.	Hasta Arama .....	32
4.4.2.	Hasta Kayıt .....	33
4.4.3.	Müracaat Kayıt.....	34
4.5	Yönetim Modülü .....	36
4.5.1.	Bölüm Ekranı .....	37
4.5.2.	Alt bölüm Ekranı.....	38
4.5.3.	Kullanıcı Ekranı.....	38
4.5.4.	Meslek Ekranı .....	39
4.5.5.	Personel Ekranı .....	40
4.5.6.	İl Ekranı.....	41
4.6	Poliklinik Modülü .....	42
4.6.1.	Müracaat Listeleme Ekranı .....	42
<b>5</b>	<b>SONUÇ VE ÖNERİLER.....</b>	<b>44</b>
5.1	Sonuçlar .....	44
5.2	Öneriler .....	46
<b>6</b>	<b>KAYNAKLAR LİSTESİ.....</b>	<b>48</b>



## ŞEKİLLER DİZİNİ

Şekil 1 e-Sağlık Sistemi Bilgi Sıralaması.....	4
Şekil 2 Veri Kalitesi Yönetim Modeli .....	5
Şekil 3 Hedeflenen Hastane Sistemi .....	7
Şekil 4 Sağlık Bakanlığı Hastane Basamakları.....	9
Şekil 5 Hastaneleri Elektronikleşme Seviyeleri.....	10
Şekil 6 Hastane Bilgi Sistemi Modülleri .....	12
Şekil 7 Hastane Sistemi İşleyiş Şeması .....	13
Şekil 8 Hastaların İnternet Kullanım Oranları .....	16
Şekil 9 JPA Metadata Açıklayıcı Notları .....	20
Şekil 10 JPA Kayıt Tutma Özelliği.....	22
Şekil 11 e-Hastane Sistemi Kullanım Durumları Diyagramı .....	24
Şekil 12 e-Hastane Sistemi Proje Yapısı.....	26
Şekil 13 e-Hastane Sistemi Paket Yapısı .....	28
Şekil 14 e-Hastane Sistemi Proje Dosyaları.....	28
Şekil 15 e-Hastane Sistemi Giriş Ekranı.....	30
Şekil 16 e-Hastane Sistemi Giriş Kullanıcı Mesajları .....	30
Şekil 17 e-Hastane Sistemi Hasta Arama Aktivite Diyagramı.....	32
Şekil 18 e-Hastane Sistemi Hasta Kayıt Aktivite Diyagramı .....	34
Şekil 19 e-Hastane Sistemi Müracaat Ekranı .....	35
Şekil 20 e-Hastane Sistemi Müracaat Kayıt Aktivite Diyagramı.....	36

Şekil 21 e-Hastane Sistemi Düğme Paneli.....	37
Şekil 22 e-Hastane Sistemi Bölüm Ekranı.....	37
Şekil 23 e-Hastane Sistemi Alt Bölüm Ekranı.....	38
Şekil 24 e-Hastane Sistemi Alt Bölüm Ekranı.....	39
Şekil 25 e-Hastane Sistemi Meslek Ekranı.....	40
Şekil 26 e-Hastane Sistemi Personel Ekranı.....	41
Şekil 27 e-Hastane Sistemi İl Ekranı .....	42
Şekil 28 e-Hastane Sistemi Müracaat Listeleme Ekranı.....	43

## **TABLULAR DİZİNİ**

Tablo 1 Birinci Basamak Sağlık Kuruluş Sayıları..... 8

Tablo 2 JPA Açıklayıcı Not Örnekleri ve Açıklamaları ..... 23

## SİMGELER VE KISALTMALAR

JPA	Java Persistence API
SAĞLIK BAKANLIĞI	T.C. Sağlık Bakanlığı
SAĞLIK SİSTEMİ	T. C. Sağlık Sistemi
WEB BROWSER	İnternet tarayıcı
SSK	Sosyal Sigortalar Kurumu
XML	Extensible Markup Language
SUN	Sun Microsystems
JDBC	Java Database Connectivity
EJB	Enterprise Java Beans
GUI	Graphical User Interface
ORM	Object Relational Mapping
API	Application User Interface
SPGK	Sağlık Bakanlığı Sağlık Projeleri Genel Koordinatörlüğü

# 1 GİRİŞ

Sağlık hizmetlerinin elektronik ortamda, ülke çapında, sürdürülebilir olarak, güvenilir bir biçimde hizmete konulup, yönetilmesine e-Sağlık\* denir (Türkiye II. Bilişim Şurası, 2004). e-Sağlık, hızla gelişen bilişim teknolojileriyle birlikte Avrupa Birliği ve diğer gelişmiş ülkeler gibi Türkiye’de de, uygulanması kaçınılmaz olan diğer elektronik devlet hizmetlerinden sadece bir tanesidir. e-Sağlık hizmetinin sadece bir kolu olan e-Hastane (e-Hospital) ise, hastanedeki bilgilerin sayısal ortama aktarılmasıyla oluşan sistemdir.

Bir ülkenin e-Hastane sistemi, kendi yapısı içinde birçok sistemi barındıran bir çatı görevi görür. Bu çatı altında bulunan temel bileşenler, Hasta Kayıt / Kabul, Poliklinik, Yatan Hasta, Ameliyathane, Eczane, Vezne, İstatistik, Döner Sermaye / Faturalama / Muhasebe modülleri tarafından oluşturulmaktadır (T.C. Sağlık Bakanlığı, I. Sağlık Bilişim Kongresi, İzmir, 2006).

Bu çalışmada, yetkili kişiler tarafından ulaşılabilir, hasta bilgilerinin güvenliğinin sağlandığı, hasta tedavisinin tamamının izlenebildiği, hastanın hastaneye girişinden, taburcu olmasına kadar süren akışının sağlanabileceği bir uygulama programı geliştirilmiştir. Uygulama programı, bir başlangıç olarak modellenmiş ve ileriye yönelik çalışmalara temel oluşturabilecek şekilde analiz edilip, tasarlanmıştır.

Mevcut sistemlerin işleyişi incelendiğinde bir çok sorunla karşılaşılmıştır. Bu sorunlar içerisinde, sağlık sektöründe de enformasyona dayalı bir yönetimin varlığından söz etmenin olası olmadığı, kurum içi, kurumlar arası ve yurt dışı kurumlarla yeterli düzeyde bilgi alış verişi bulunmaması, verilerin kalitesinin düşük olduğu bilindiğinden, yöneticilerce kullanılmadığı gözlemlenmiştir.

Bu sorunların ortadan kaldırılması, ancak bilgilerin, tamamıyla sayısal ortama taşınması ile gerçekleşebileceği için, öncelikle mevcut sistemlerin analizi yapıldıktan sonra bu yapıya uygun bir yazılım geliştirilmesiyle mümkün olacaktır.

\* T.C. Sağlık Bakanlığı tarafından, kayıtların dijital ortama taşınmasını öngören proje.

Bu çalışmada, bilgilerin sayısal ortama taşınmasıyla, hasta kayıtlarının el ile girilen bilgilerden daha hızlı, daha kolay ulaşılabilir, daha güvenli ve daha kaliteli olarak tutulmuştur. Böylece sistem, hastane personeline daha sonraki işlemler için iyi bir alt yapı sağlamış ve iyi bir model oluşturmuştur.

## **1.1 Genel Bakış**

Bu bölüm teze bir genel bakış sunmaktadır.

Bölüm 2, mevcut hastane sistemlerinin incelenmesi, sorunları, çözümleri ve hedefleri gibi konulara deyinmektedir.

Bölüm 3 ise, Java teknolojileri hakkında detaylı bilgi vermektedir. Bu bölümde Java teknolojilerinin yapısı, kodların çalışma şekli, Java veritabanı bağı ve Java Persistent API (JPA) hakkında detaylı bilgi verilmektedir.

Bölüm 4, bu çalışma sonucunda ortaya çıkarılmış olan e-Hastane sistemini tanıtmaktadır. e-Hastane sisteminin mevcut sistemlerde nasıl ele alındığı, bir müracaat kaydının nasıl yapıldığı, sistemin yapısı ve tanımı ve aşamalar bu bölümde tartışılmıştır.

Son olarak bölüm 5'te ise, bu çalışma sonunda ortaya çıkan sistemin kazandırdığı yenilikler, karşılaşılan zorluklar ve çözümler, ve son olarak da ileriye yönelik yapılması gereken çalışmalara değinilmiştir.

## 2 SAĞLIKTA BİLİŞİM SİSTEMLERİ

2006 yılında T.C. Sağlık Bakanlığı tarafından, İzmir’de yapılan, I. Sağlık Bilişim Kongresinde de bahsedildiği gibi, sağlık sektörüne genel olarak bakıldığında diğer sektörlerde de olduğu gibi hizmet sunum ve planlamalarında bilgiye dayalı yönetimin giderek önem kazandığı bir çağı yaşıyoruz, “Bilgi Toplumu Çağı”. Bilindiği gibi bilgi toplumu çağı diye nitelendirdiğimiz günümüzde, en büyük yatırım insana, dolayısıyla tıp bilişimine – bilgiye yapılmaktadır. Ekonomik gücün en önemli kaynağı da bilginin yönetilmesi olarak kabul edilmektedir. Söz konusu Bilgi Yönetimi ise Tıp Bilişimi ile birlikte bilişim teknolojilerini doğru kullanmaktan geçmektedir.

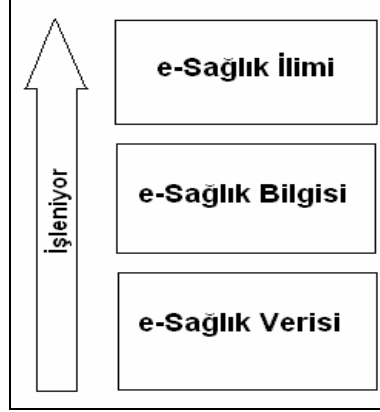
Avrupa Birliği üye ülkeleri, gelecek 10 yılda dünyadaki en rekabetçi ve en dinamik bilgi tabanlı ekonomisi haline gelmek üzere hedeflerini ortaya koymuşlar ve e-Avrupa Eylem Planını oluşturmuşlardır. Aday ülkeler de aynı stratejiyi benimsemişler ve benzer biçimde e-Avrupa+ girişimini başlatmışlardır. Her iki girişimin eylem planlarında e-Sağlık alanına geniş yer verilmiş ve Türkiye Cumhuriyeti Sağlık Bakanlığı tarafından da e-Avrupa+’daki e-Sağlık hedefleri gerçekleştirilmek üzere çalışmalar başlatılmıştır (T.C. Sağlık Bakanlığı, I. Sağlık Bilişim Kongresi, İzmir, 2006).

T.C. Sağlık Bakanlığının, Türkiye Sağlık Bilgi Sistemi adlı sunumuna göre e-Sağlık, bilgi ve iletişim teknolojilerinin tüm fonksiyonlarının, vatandaşların, hastaların sağlığını iyileştirmesinde sağlık hizmetlerine ulaşılabilirliğini arttırmak ve sağlık sektöründe yer alan tüm paydaşlara kaliteli, verimli ve etkin hizmetlerin sunumu için kullanılması olarak yorumlanmıştır. Burada dikkatlerin toplanması gereken kelime, e-Sağlık açısından da çok önemli olan kalitedir.

### 2.1 Sağlık Bilişiminde Kaliteli Veri

Bu hedeflere ulaşabilmek için belirli sayıda veriye, bilgiye sahip olmak gerekmektedir. Öncelikle bilgi ve veri kavramlarını birbirinden ayırt etmek gerekmektedir. Bilgi, işlenmiş veri topluluğudur. İlim ise bilginin belirli kurallar,

deneyimler ve ilişkiler üstünde uygulanmasıdır. Aşağıdaki şekilde bilgi, veri ve ilimin sağlık sistemleri içindeki sıralaması görülmektedir.



Şekil 1 e-Sağlık Sistemi Bilgi Sıralaması

Yapılan bütün çalışmalar, başka ülkelerde dünyanın değişik yerlerinden katılımcıların katıldığı konferanslar ve T.C. Sağlık Bakanlığı'nın koymuş olduğu hedefler, hep kaliteli verilerin toplanmasıyla, işlenip bilgi haline gelmesiyle ve arkasından ilim olarak kullanılmasıyla oluşmaktadır.

Genel olarak güvenilir ve kaliteli veriler hemen hemen her türlü çalışma ve özellikle de bilimsel araştırmaların temelini ve ilk aşamasını oluşturmaktadır. Elimizde sağlıklı veriler olmadığı sürece sağlıklı bir sisteme sahip olmamız düşünülemez. Bu sebepten dolayı, e-Hastane projesi içinde kullanıcı, hasta ile ilgili bilgileri sisteme kaydederken ne kadar titiz davranırsa, bilgiler ne kadar sağlıklı olursa sistemin değişik bölümlerinde kullanıcılara o kadar yardımcı olacaktır. Bilgi, e-Hastane projesinde iki yerde işimize yaramaktadır. Bunlardan birincisi, hastaların kayıtları ve tedavileri sürecinde, diğeri ise birçok amaç için kullanılacak olan raporlar kısmındadır.

Wager ve ark. 2005 yılında yapmış olduğu çalışmalar sonucunda, sağlık sistemi'nde kaliteli verinin önemi bir kaç başlık altında toplamak mümkündür.

**Hasta Güvenliği :** Hasta bilgilerinin hastadan yanlış alınması, hastanın tedavisi ve sonrasındaki safhalarında hastanın güvenliğini kötü yönde etkileyebilmektedir. Bu



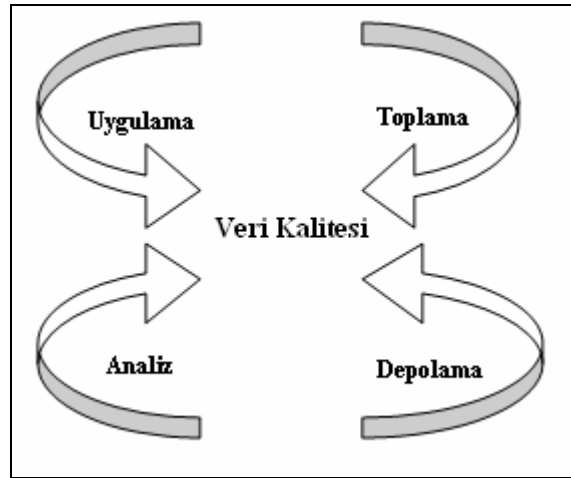
konu başlığı altında verinin yanlış alınması, yanlış tercüme edilmesi, doğru olmayan veri alınması örnek olarak gösterilebilir.

**Toplum Güvenliği :** Toplum sağlığını yakından ilgilendiren bir konudur. Verinin zamana karşı, koordineli, eğitilmiş kişiler tarafından doğru bir biçimde toplanabilmesi toplumun güvenliğini de yakından ilgilendirmektedir.

**Hasta Tedavisinin Devamı :** Hastanın tedavisinin sürdürülebilmesi toplanan bilginin sistem içinde gerekli yerler tarafından paylaşılması sonucunda sağlanabilir.

**Sağlık Sistemi Ekonomisi :** Yukarıdaki bölümde de belirtildiği gibi verinin lazım olduğu raporlama sisteminde eksik olan veri, doğru olmayan bir veri topluluğu, örnek olarak bir sistem için gerekli olan bütçenin yanlış çıkmasına sebep olabilir.

e-Hastane sSistemi için önemli olan veri kalitesi yönetimi modeli aşağıdaki şekilde gösterilmiştir.



**Şekil 2 Veri Kalitesi Yönetim Modeli**

Şekil 2’de belirtilen veri kalitesi modeli belirli özelliklere sahiptir, bunlardan bazıları;

- Erişilebilirlik
- Tutarlılık
- Birim
- Sayılabilir olması

- Duyarlılık
- Doğruluk
- Tanımlanabilir olması
- İlişkisel olması
- Zamanlılık

olarak sıralanabilir. Yukarıda sayılan maddelerden her biri, e-Sağlık ve e-Hastane gibi sistemlerde kaliteli veriyi kullanmamızı gerektiren sebeplerdendir.

## 2.2 Sağlık Bilişiminde Hedefler

Bilindiği gibi, tüm dünyada sağlık sektörü, çok çeşitli seviyelerdeki sağlık kurumları ve hastanelerle beraber, hekimlerin, laboratuvar ve görüntüleme merkezlerinin, eczane ve ilaç depolarının, sigorta kuruluşlarının, bankaların ve hasta olarak tüm bireylerin katılımcı olarak bulunduğu karmaşık bir ilişkiler ağını içinde barındırır. (Türkiye II. Bilişim Şurası, 2004)

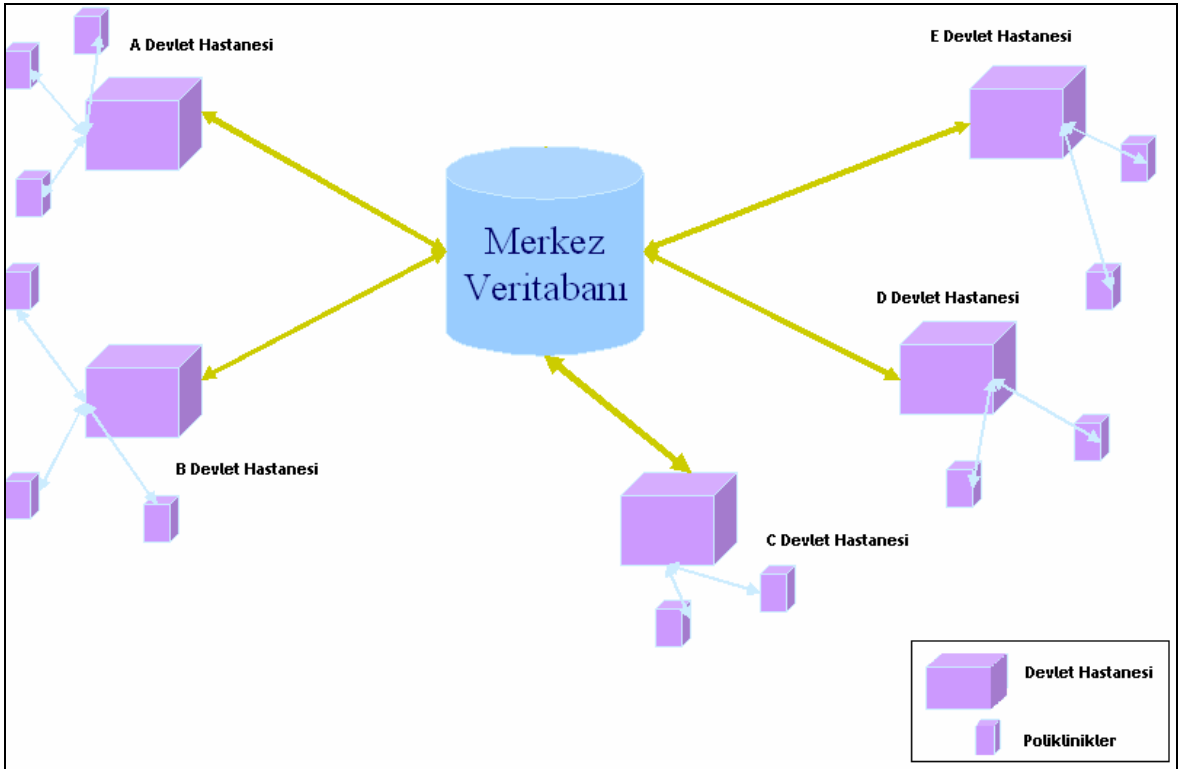
Sağlık sektöründe bilişim teknolojilerinden faydalanmamız bizi “e-Hastane” projesinde bir takım hedeflere daha çok yaklaştırmaktadır. Bu hedefler;

- Ülke genelinde hizmet verilmesi,
- Hızlı ve kolay erişilebilir olması,
- Her yerde yüksek standartlara uygunluğu,
- Kurum ihtiyaçlarına değil, hasta ihtiyaçlarına odaklı olması,
- Hastaya verilen hizmetin kalitesini en üst düzeye çıkarmaya yönelik verimli çalışması,
- Yeni teknolojilerden ve bilgi birikiminden faydalanması,
- Hastalıkların tedavisinin yanı sıra sebeplerinin de ortaya çıkmasına yardımcı olmasıdır,
- Araştırma ve geliştirme için gerekli güvenilir ve kaliteli verilerin sağlanmasıdır.

2003 yılında T.C. Sağlık Bakanlığı tarafından bir hedef olarak belirlenen, yaşam boyu sağlık kaydının oluşturulabilmesi amacıyla, bireylere doğum ile birlikte hiç değişmemek üzere, kişiyi tanıttıcı bir numara verilmesi hakkında çalışmalar başlamış ve belirli aşamalara gelinmiştir (Türkiye Sağlık Bilgi Sistemi, 2003).

Reinhold Haux'un 2006 da yayınlanan Health Information Systems – Past, Present, Future (Sağlık Sistemlerinin Dünü Bugünü Yarını) adlı makalesinde de, hastane bilgi sistemlerinin hedefleri olarak belirttiği yüksek kalitede, etkin bir hasta takip sistemi, bize, bir sağlık sisteminin hedeflerinin dünyanın her yerinde aynı olması gerektiğini gösteriyor.

Bu çalışma sonucunda, bir adım olarak ortaya çıkan e-Hastane sisteminin, ileriki aşamalarda olması istenen hali aşağıdaki şekilde gösterilmiştir.



**Şekil 3 Hedeflenen Hastane Sistemi**

Bu sistemde olması istenen, Türkiye çapında büyük bir veritabanına sahip, bütün hastanelerin ortak bilgi alışverişinde bulunabileceği, hasta bilgilerinin güvenli olarak saklanabilmesidir. Bu sağlık sisteminde, bütün devlet hastaneleri birbirine entegre olmuş ve doktorlar, hastaların tedavilerini kolaylıkla gerçekleştirecek ve

takip edilmiş teknolojilerin getirdiklerini kullanarak, muayeneleri kolaylaştıracak servislerden faydalanacaklardır.

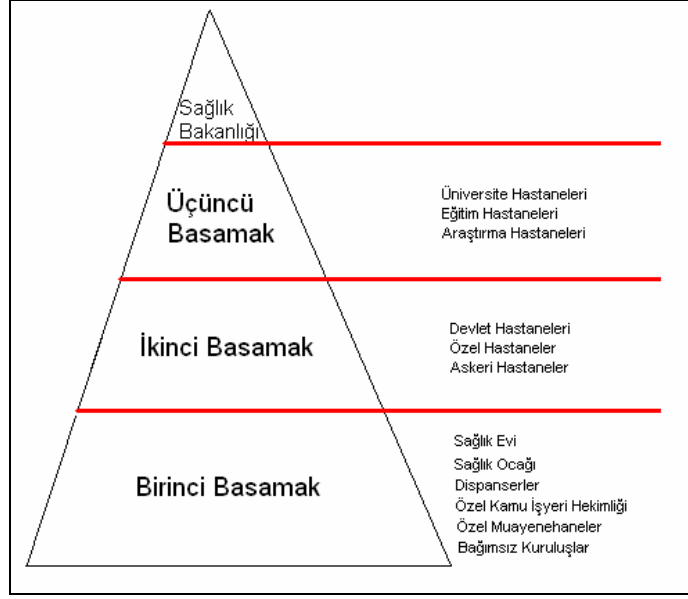
### 2.3 Sağlık Bilişiminde Mevcut Durum

Türkiye'nin şu anda içinde bulunduğu durum incelendiğinde, karmaşık bir yapıyla karşı karşıya kalınır. Bunun başlıca sebeplerinden bir tanesi, karmaşık bir biçimde yapılanmış olan sağlık birimleridir. Bunlar; birinci basamakta sağlık ocakları, ana çocuk sağlığı ve aile planlaması merkezleri, verem savaş dispanserleri, kamu hastanesi poliklinikleri, SSK sağlık istasyonları ve dispanserleri, belediyeler, işyeri hekimleri, özel teşhis ve tedavi poliklinikleri, özel muayenehaneler ve farklı büyüklükte diğer kamu dispanserleri ile vakıf poliklinikleri olarak sayılabilir.

Kurum Tipi	Sayısı
Sağlık Evi	11,731
Sağlık Ocağı	5, 793
Verem Savaş Dispanseri	274

**Tablo 1 Birinci Basamak Sağlık Kuruluş Sayıları**

İkinci ve üçüncü basamakta yataklı tedavi hizmeti veren 1226 kurumun %60.7'si Sağlık Bakanlığı, %21.3'ü özel hastaneler, %9.6'sı SSK hastaneleri, % 9.2'si Milli Savunma Bakanlığı ve %3.4'ü üniversite hastanelerine aittir. (Türkiye II. Bilişim Şurası, 2004).



**Şekil 4 Sağlık Bakanlığı Hastane Basamakları**

Türkiye’de yapılan sağlık harcamalarının kaynağı üç yerden sağlanmaktadır. Bunlar sırasıyla, devlet bütçesi, sosyal güvenlik kurumları ve bireylerin cepten yaptığı harcamalardır. Türkiye’de e-Sağlık gibi bir ulusal projede mevcut durumda olmamızın sebeplerinden en büyüğü ve temel olanı, ülkemizde bir “Ulusal Sağlık Hesabı Sistemi” olmamasıdır. Bu tarz bir sistemin olmayışı, bizi bu konuda elimizde tutmanın çok büyük faydalar getireceği verilere sağlıklı bir şekilde ulaşmamızı engellemektedir.

Ülkemizde diğer sektörlerde olduğu gibi, sağlık sektöründe de enformasyona dayalı bir yönetimin varlığından söz etmek olası değildir. Bunun yanı sıra toplanan verilerin kalitesi düşüktür ve bu veriler çeşit ve sayısal olarak eksik, kapsamı yetersiz ve doğruluğu kuşkuludur. Ayrıca veri kalitesinin kontrolünü yapacak bir mekanizma bulunmamaktadır (Türkiye II. Bilişim Şurası, 2004).

Türkiye’nin içinde bulunduğu bu durum, değişik açılardan bakıldığında çeşitli sonuçlar doğurmaktadır. Fakat bunlardan değişmeyecek olanlarından bir tanesi, yukarıdaki bölümlerde de belirtildiği gibi veri kalitesidir. Gerekli olan verinin toplanabilmesi, verileri toplayabilecek olan personelin gerekli bilgisayar eğitiminin almış olması, gerektiğinde sistemin problemleri yerlerine müdahale edebilmesi gerekmektedir. Dolayısıyla, e-Sağlık sisteminde eğitim önemli bir yer almaktadır.

Aşağıdaki şekil hastanelerin elektronikleşmesi konusunda, 5 değişik seviye olduğunu göstermektedir. Türkiye'nin e-Sağlık konusundaki genel durumu şu an itibariyle birinci seviyedir (Tan, 2005).



**Şekil 5 Hastaneleri Elektronikleşme Seviyeleri**

Bu seviyelerin detaylandırılmış açıklamaları aşağıda verilmiştir.

**Seviye 1 :** Bu seviyede, hasta kayıtları, hasta hizmetleri sisteminde hala kağıt üstünde tutulmaktadır. Hasta bilgilerinin bir kısmı bilgisayara geçirilmiş olabilmektedir. Bu seviyede sağlık hizmetleri sisteminin gereksinimi olan bir kısım işlemlerin (hasta kayıt, randevu, sonuçlar vb.) otomasyonu yapılabilir.

**Seviye 2 :** Bu seviyede de, hasta hizmetleri sisteminde, hasta kayıtları sayısallaştırılarak (dijital ortama geçirilerek) doküman görüntüleme sisteminde yerini alır. Hastanın tümüne yakın bilgileri alınarak taranır ve optik imaj olarak sisteme aktarılır (Wager ve ark., 2005). Fakat bu bilgilerin sisteme aktarılması, kullanıcıya bilgileri güncelleme, analiz etme vb. imkanları sunmamaktadır.

**Seviye 3 :** Wager ve arkadaşlarına göre, bu seviyeye kadar tıp kayıtları pasif bir depolama aygıtı görevi görmüştür fakat bu seviyede tıp kayıtlarının kullanıcı hekimlere bir karar verme destek aracı olarak hizmet vermekte olduğunu görmekteyiz. Aynı zamanda kayıtlar kullanıcılara uyarı verebilmektedir. Bir hastanın herhangi bir ilaca karşı alerjisi olup olmadığını daha önceden girilen

bilgilere göre kullanıcı hekime uyarı vermesi buna bir örnek olarak gösterilmektedir.

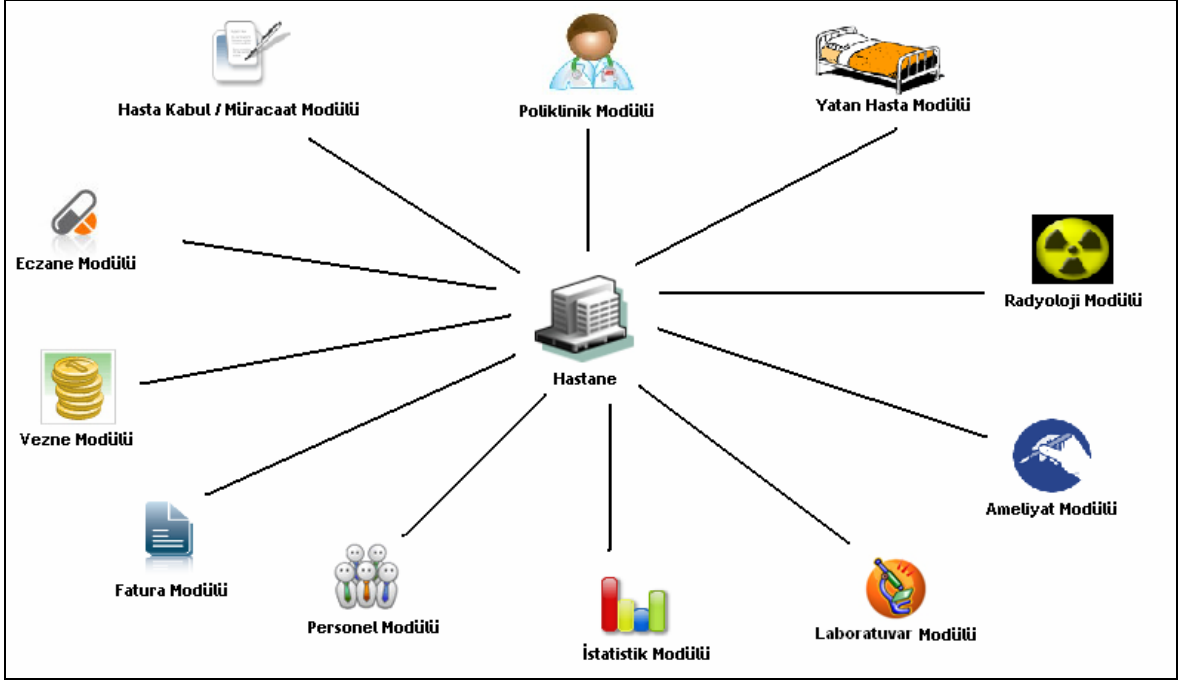
**Seviye 4** : Elektronik tıp kayıtları sadece hastanın bilgilerini tutmasına rağmen, Elektronik hasta hizmeti kayıtları, hastanın bilgilerinin yanı sıra bütün hasta hizmetlerini ilgilendiren bilgileri ve hatta, eğer bağlıysa, bağlı olduğu diğer kamu kuruluşlarındaki bilgilerini de tutmaktadır. Hastanın geçmişte müracaatta bulunduğu hastanelerdeki bilgilerini, hangi tedavilerin uygulandığını, ne sonuçlar verdiğini bu seviyeden görebilmemiz mümkündür (Wager ve ark., 2005).

**Seviye 5** : Bu seviyede daha çok geniş kitlelerden bilgiler alınıp, genel sonuçlar doğuracak bilgiler ortaya çıkarılmaktadır. Örnek olarak, bireylerden beslenme alışkanlıkları, sigara ve alkol kullanma sıklıkları hakkında bilgiler toplanır ve bu sonuçlara göre raporlar çıkartılır. Wager ve ark. (2005) göre bu bilgiler zamanı geldiğinde çıkarılır ve eğer tedavisinde yardımcı olacaksa kullanılmaktadır. Bu seviyede kayıtlar daha çok tıbbi değil refah kayıtları olarak adlandırılır.

Bu çalışma ile beraber Türkiye Sağlık Portal'ında, mevcut hasta kayıtlarının kağıt üstünde tutulması yerine, sayısal ortamlara aktarılması ve bundan sonra girilecek olan hasta kayıtlarının analiz edilmesi, bilgilerinin güncellenmesi, kısacası sağlık sistemini bir üst seviyeye taşımak adına bir yazılım geliştirilmiştir.

## **2.4 Hastane İşleyiş Yapısı**

Çalışma dahilinde, geliştirilen e-Hastane yazılımının, analizi sırasında incelenen hastanelerden birisi olan 100. Yıl Hastanesi'nin ve T.C. Sağlık Bakanlığı'na bağlı bir diğer hastane olan Balıkesir Devlet Hastanesi'nin işleyiş yapısı, modülleri ve hatta tutulan hasta kayıtları verileri birbirine benzerlik gösterir. Bu sonuca göre, T.C Sağlık Bakanlığının yayınladığı I. Sağlık Bilişimi kongresinin Sonuç raporunda da belirtilen, bir hastane bilgi sisteminin işleyişi sırasında kullanılan modüller Şekil 6'da gösterilmiştir.



**Şekil 6 Hastane Bilgi Sistemi Modülleri**

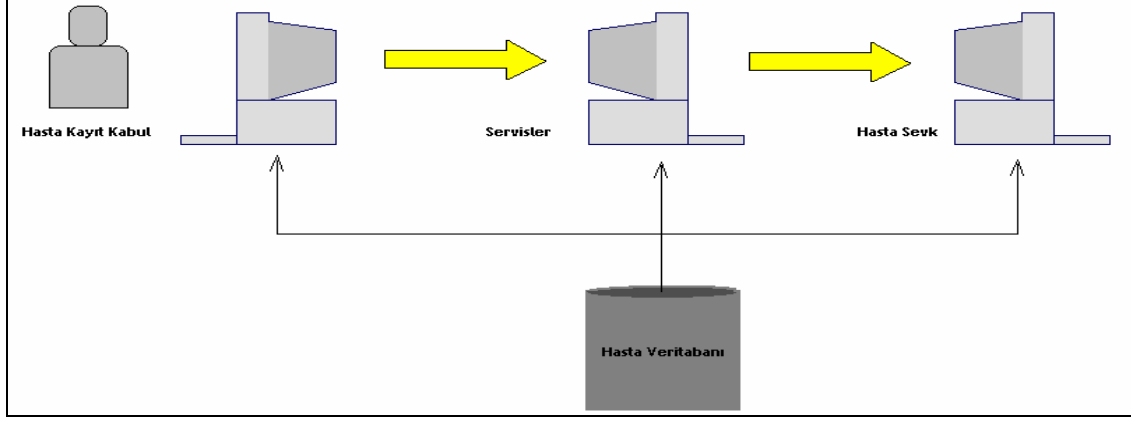
2006 yılında yayınlanan Richard E. Scott'ın e-Records in Health (Sağlıkta Elektronik Kayıtlar) adlı makalesinde , bir hastane bilgi sisteminde, hastanın hastaneye attığı ilk adımdan itibaren kayıt altına alınması, bilgi sisteminin Müracaat / Kayıt Kabul Modülü ile gerçekleşeceği ve bu bölümün hastane sistemi için hayati bir önem taşıdığı belirtilmektedir. Ayrıca makalede, kayıt altına alınacak olan kaliteli (doğru, hatasız, gerekli) bilgilerin, geliştirilecek olan hastane sistemlerinin diğer modüllerinde de, kullanıcılara kolaylık sağlayacağına değinilmiştir.

Gerçek hayatta bir hastanın randevu alabilmesi ve muayene olabilmesi için belirli işlemleri sırasıyla tamamlamış olması gerekmektedir. Bu işlemlerden ilki, hastanın öncelikle, başvuracağı hastaneye sağlık karnesi ile gelmesi ve müracaatı sayısal ortama geçirecek olan sağlık görevlisine sağlık karnesini teslim etmesidir.

Bu ve bundan sonraki aşamalar, araştırma sırasında incelenen hastanelerden Ankara 100. Yıl Hastanesi, Bayındır Hastanesi ve Başkent Üniversitesi Hastanesi için de benzerlik göstermektedir. Sağlık görevlisi bir sonraki aşamada hastanenin hastalardan istemiş olduğu bilgileri, kimi zaman bilgisayar kullanarak bir veritabanına, kimi zaman ise bir form doldurularak bir dosyaya işler. Girilen bu



bilgiler sonrasında hastanın muayene olmak istediği alt bölümün müsait olması durumunda hastaya sıra numarası (bar kod)\* verilerek, ilgili bölüme yönlendirilir. Bu işlemlerin tamamlanmasından sonra hasta muayene sıra numarasını almış ve numaratornden\*\* sırasını beklemektedir.



**Şekil 7 Hastane Sistemi İşleyiş Şeması**

Bir hastanenin işleyişi, incelenen hastaneler ve yapılan araştırmalar sonucu yukarıda Hastane İşleyiş Şeması, Şekil 7’de görüldüğü gibidir. Buna göre önce hasta hastaneye gelir, bilgileri kayıt altına alınır ve arkasından gerekli servislerde gerekli işlemler yapılır ve hasta gerek duyulduğu takdirde sevk edilir yada muayenesi sonlandırılır.

Bu şekilde bir çalışmanın gerçek hayatta uygulanmasının sonucunda kazanılacak yenilik ise, kağıtsız hastane ve işlem hızı olarak değerlendirilebilir.

“Kağıtsız Hastane” terimini açıklayacak olursak, kağıdın ortadan kalktığı, hasta, müracaat, muayene ve laboratuvar işlemlerinin tamamen sayısallaştırılmış olduğu bir e-Hastane sistemi olduğunu görürüz. Bu işlem gerçekleşirse aynı zamanda, Türkiye’de bulunan hastane sistemi birinci seviyeden ikinci seviye çıkartılmış olacaktır. Bu ise diğer seviyelere çıkmak için gerekli olan ilk adımdır.

\* Hastaya müracaatı sonrası sırasını bildiren etiket

\*\* Hastaya müracaat numarası, etiket, barkod, veren aygıt.

## 2.5 Sağlık Bilişiminin Sorunları

Sağlık sektöründe bulunan temel sorunlar, yaşlanmakta olan nüfus, her geçen gün değişen hasta beklentileri, sağlık tehditlerine alınması gereken tedbirler, teknolojiye yatırım ve organizasyon değişimi, bilgilerin güvenli bir şekilde alınması ve saklanması, kısıtlı bütçe imkanlarıyla en iyi sağlık hizmeti sunumu gereksinimi olarak isimlendirilebilir.

Teknolojik olarak gelişen toplumlara bakıldığında dikkati çeken tek şeyin “e-Sağlık” programı olmadığı görülmektedir. “e-Sağlık”ın yanı sıra e-Devlet, e-İş, e-Eğitim ve bunların hepsinin alt yapısını sağlamak adına oluşturulmuş geniş bant İnternet erişimi ve çok sağlam bir güvenlik altyapısı, tam anlamıyla gelişmiş teknolojik bir ülkede bulunması gerekli olan e-uygulamalardır.

Sağlık hizmetlerinin bilişim ile buluşmasının hastalara ve kullanıcılara kazandırdığı zamanın yanında, verilerin ve yapılan işlemlerin aynı zamanda güvenli bir şekilde saklanmasına imkan sağlanması önemli bir özelliktir. “e-Sağlık” hizmetlerinde, bilgi ve iletişim destekli sağlık bakım süreçleri, elektronik hasta kayıtları ve eğitim, günümüz sağlık hizmetlerimizde işimizi kolaylaştıran temel başlıklardır. Yukarıda mevcut durumdan bahsedilirken görüldüğü üzere, Türkiye’de bir “Ulusal Sağlık Hesabı Sistemi” olmayışı, bize sağlıklı verilerin gelişini engellemektedir. Türkiye II. Bilişim Şurası’nda da anlatıldığı gibi bu bilgilerin sağlıklı olarak elimize geçmemesi bize bir takım sorunları da beraberinde getirmektedir.

Prof. Dr. Recep Akdur’un, Türkiye’nin Geleceği için Sağlık Sektöründe Yeni Arayışlar adlı sunumunda belirtildiği üzere:

- Ülkemizde diğer sektörlerde olduğu gibi, sağlık sektöründe de enformasyona dayalı bir yönetimin varlığından söz etmenin olası olmadığı,
- Verilerin kalitesinin düşük olduğu bilindiğinden, yöneticilerce kullanılmadığı,
- Sağlık Bakanlığınca ve diğer kuruluşlarca mevcut sistem içinde toplanan verilerin çeşit ve sayısal olarak eksik, kapsamının yetersiz ve doğruluğunun kuşkulu olduğu,

- Kalite kontrolü yapacak bir mekanizmanın bulunmadığı,
- Kurum içi, kurumlar arası ve yurt dışı kurumlarla yeterli düzeyde bilgi alış verişi bulunmaması,
- Yaşam boyu sağlık kayıt sisteminin oluşturulmaması, tıbbi süreçlerde hatalı kararlar verilmesine ve bu şekilde sağlık hizmetinin kalitesinin düşmesinin yanı sıra, yüksek maliyetli sağlık hizmeti sunumuna, hatta istismar boyutlarına varan ciddi ekonomik kayıplara yol açması,
- Sağlık Bakanlığı ve diğer kurumlara bağlı sağlık kurumlarında sağlıklı bir enformasyon sisteminin olmaması,
- Sektörde fonksiyonel bilgi sistemi olmaması nedeniyle sağlık finansmanı kurumları bütçeleme bağlamında etkin ve verimli çalışmaması,

bize sağlık sektöründe karşılaşılan sorunları açıkça göstermektedir. Bu sebeplerin hepsi bize, aynı zamanda kaliteli bilgiler ile çalışabilecek bir e-Hastane projesinin ne kadar gerekli olduğunu göstermektedir (I. Sağlık Bilişim Kongresi, 2006).

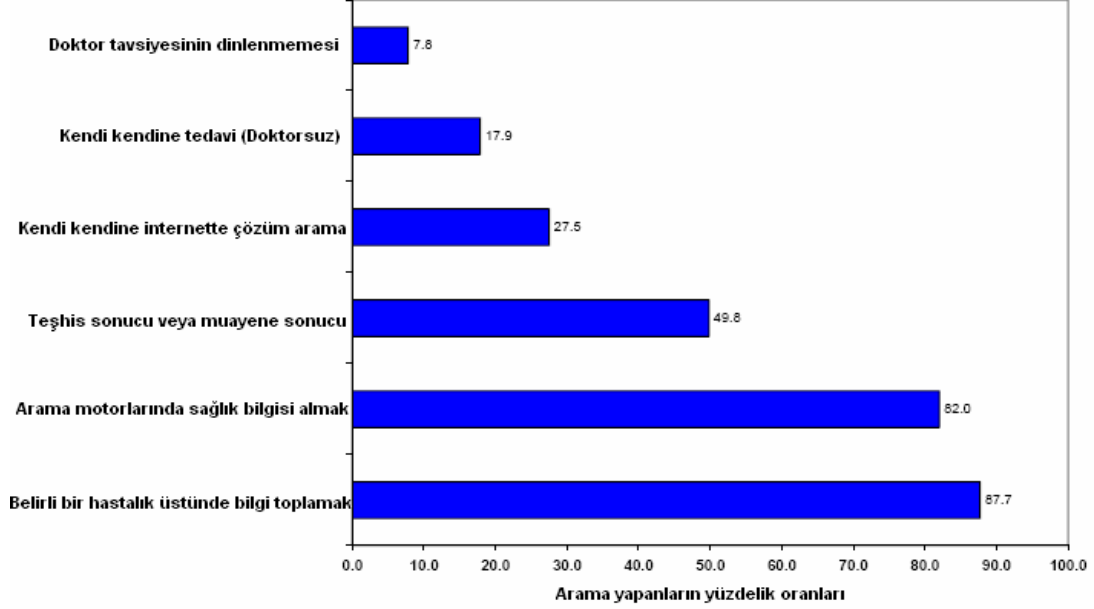
Bunların hep bir sorun olduğu göz önünde tutulduğunda, Türkiye’de bir e-Hastane Projesinin acil olarak hayata geçirilmesi gerektiği ve bunun için ne gerekiyorsa yapılması gerektiği kolayca anlaşılmaktadır.

## **2.6 Sağlık Bilişiminde Yapılan Çalışmalar**

Önceki bölümlerde de bahsedildiği gibi, sağlıkta bilişim hızla gelişmekte ve bu gelişmeleri takip eden ülkeler, hastalarının tedavilerini daha iyi şekillerde hizmet vererek sürdürmektedirler. Bu zamana kadarda dünyanın her yanından ülkeler sağlık hizmetlerinin gelişmesi adına değişik teknolojileri takip ederek çeşitli araştırmalar yapmışlar ve çeşitli sonuçlar elde etmişlerdir.

Tamamlanan çalışmaların birinde, 2005 yılında Venedik’te yapılan konferansa katılan Ivica Milicevic, insanların hastalanmadan, hastalandıktan sonra ve tedavi yöntemleri için İnternet’ e ne sıklıkla başvurduklarını yüzdeli oranlarla bir grafik

içinde göstermektedir. Şekil 6'da gösterilen bu oranlar bize internet gibi çok güvenilir olmayan bir kaynaktan insanların hastalıkları hakkında bilgi sahibi olmaya çalıştıklarını fakat kaliteli veri sunan, etkin kullanılan bir hastane bilgi sistemi olması dahilinde İnternet' e bu kadar çok başvurulmayacağını, dolayısıyla da insanların yanlış bilgilere sahip olmayacağını göstermektedir.



**Şekil 8 Hastaların İnternet Kullanım Oranları**

İncelemelerde bulunulan, Bandırma Kapıdağ Devlet Hastanesi'nde ise hastaların rahatlığı için geliştirilmiş bir müracaat kayıt sistemi geliştirilmiş olduğu gözlemlenmiş, ve bu sistemde, hastalar bir defaya mahsus olarak sağlık karneleriyle ilgili hastaneye başvurularını yapmakta, ve bir müracaat görevlisi hastaların bilgilerini sisteme girmektedir. Girilen bu bilgiler sonucunda yine, her hastaya bir tane olmak kaydı ile, sosyal güvenlik kurumu tarafından bir muayene başvuru kartı verilmektedir. Bu kartlar ile hastalar, muayene olmak istedikleri doktorun bölümünde bulunan, bar kod makinelerinden kartlarını geçirerek bir sonraki müracaat numarasını almaktadırlar. Doktorlara ise, sistem tarafından sırası gelen hastanın bilgileri gösterilmektedir.

Bu tarz bir sistemin etkin bir şekilde kullanılması, öncelikle seçilecek olan, bir ilçenin, arkasından ise bir il ve daha sonra Türkiye çapında uygulanması, çok kayıt

tutabilen ve ortak kullanılabilen bir veritabanı, ve bu veritabanını etkin kullanabilecek bir programlama dili ile yazılması ile gerekleŒebilir.

Haux'un 2006 yılında yayınlanan makalesinde kendisinin de belirttiđi gibi, sađlık sektöründe bir adım bile atmadan önce, öncelikli olarak yapılması gereken Œey, bir sađlık stratejisi belirlemek. Belirlenecek olan bu sađlık stratejisinin uygulanması sırasında ıkacak olan sorunların, engellerin, problemin aŒılması zaman alacađı için, bir an önce bu tarz bir planı eyleme geirmek gerekmektedir.

Avrupa Birliđi üye ülkeleri, gelecek 10 yılda dünyadaki en rekabeti ve en dinamik bilgi tabanlı ekonomisi haline gelmek üzere hedeflerini ortaya koymuŒlar ve e-Avrupa Eylem Planını oluŒturmuŒlardır. Aday ülkeler de aynı stratejiyi benimsemiŒler ve benzer biimde e-Avrupa+ giriŒimini baŒlatmıŒlardır. Her iki giriŒimin eylem planlarında e-Sađlık alanına geniŒ yer verilmiŒ ve Türkiye Cumhuriyeti Sađlık Bakanlıđı tarafından da e-Avrupa+'daki e-Sađlık hedefleri gerekleŒtirilmek üzere alıŒmalar baŒlatılmıŒtır (Œahin, 2004).

### 3 JAVA PROGRAMLAMA DİLİ

Java programlama dili, Sun Microsystem's tarafından, Java teknolojilerinin bir parçası olarak 1995 yılında geliştirilmiştir. Bu dil C ve C++ programlama dillerinin yazım kurallarının türevlerini kullanır fakat bu dillerle kıyaslandığında daha basit bir nesne modeline ve daha üstün yeteneklere sahiptir (Wikipedia, 2007) .

Java dizaynı, uygulanabilirliği ve kullanım alanları itibariyle, bilişim endüstrisi içinde dünyanın en hızlı gelişen ve en yaygın kullanılan programlama dili olmayı başarmıştır. Aynı zamanda Java'nın bir diğer özelliği ise, yer yüzünde bir çok aygıt üzerinde kullanılan programların bu programlama dili kullanılarak yazılmış olması ve tarihte bulunan en popüler açık kaynak kod (open source) yazılım ürünü olmasıdır.

Sun Microsystems'ın kendi İnternet sitesinde yapmış olduğu açıklamaya göre Java teknolojilerinin iş hayatında getirdiği yararlardan bazıları; çok seçenek sunması, performans artımı, verimlilik ve esnekliktir. Java platformu enformasyon teknolojileri yöneticilerine oldukça fazla iş mantığı uygulamaları ve kullandıkları ürünlerden bağımsız çalışan bir teknoloji sunar.

Bunlarla birlikte Java iş hayatında olduğu kadar, program geliştiriciler için de bir çok yararlı imkan sağlamaktadır. Bunlardan en önemlisi, java programlama dilini kullanarak, bir yazılımı bir platform üzerinde yazarak, bir çok platform üzerinde çalıştırma imkanı sunmasıdır. Aynı zamanda kullanıcılara yazdıkları programları Web Service\* olarak yada Web Browser içinde gösterme olanağını da beraberinde getirir.

Java'nın kurucusu olan Sun Microsystems tarafından yapılan açıklamaya göre Java'nın en önemli üç özelliği;

- Daha Zengin Kullanıcı Deneyimi

\* Bir ağ üzerinde makine-makine birlikteliğini destekleyebilen yazılım sistemi

- Web Servisleri İçin Daha Uygun Uygulama Ortamı
- İş Birleştirme Özelliğidir

Yukarıda belirtilen ilk özelliğin detayına inilecek olunursa, Java teknolojisi tarafından desteklenen bir telefon kullanılması yada iş yerinize ait bir ağa erişiminizde, Java platformunun yaşattığı gerçek hareket kabiliyeti, bu teknolojinin getirdiği en büyük özelliklerdendir.

Ele alınan ikinci madde de ise, dünyaca çok tercih edilen ve genişletilebilir programla dili olan Java ve XML programlama dillerinin çok kişiye hizmet verebilmesi için, her zaman, her yerde, her aygıtta maksimum erişim sağlayabilmesidir.

Son madde ise, Java'nın tek, birleştirici programlama modeli olduğu ve bu özelliği ile iş dünyası alt yapı sistemini oluşturabilecek her parçayı birleştirebilecek olmasıdır. Böylelikle Java kullanarak iş dünyasında geliştirilebilecek her uygulama yazılımı arkasına Java teknolojisi desteğini alarak, her durumu göz önünde bulundurur ve sadece günü değil, ilerisi içinde bir yatırım yapmış olur.

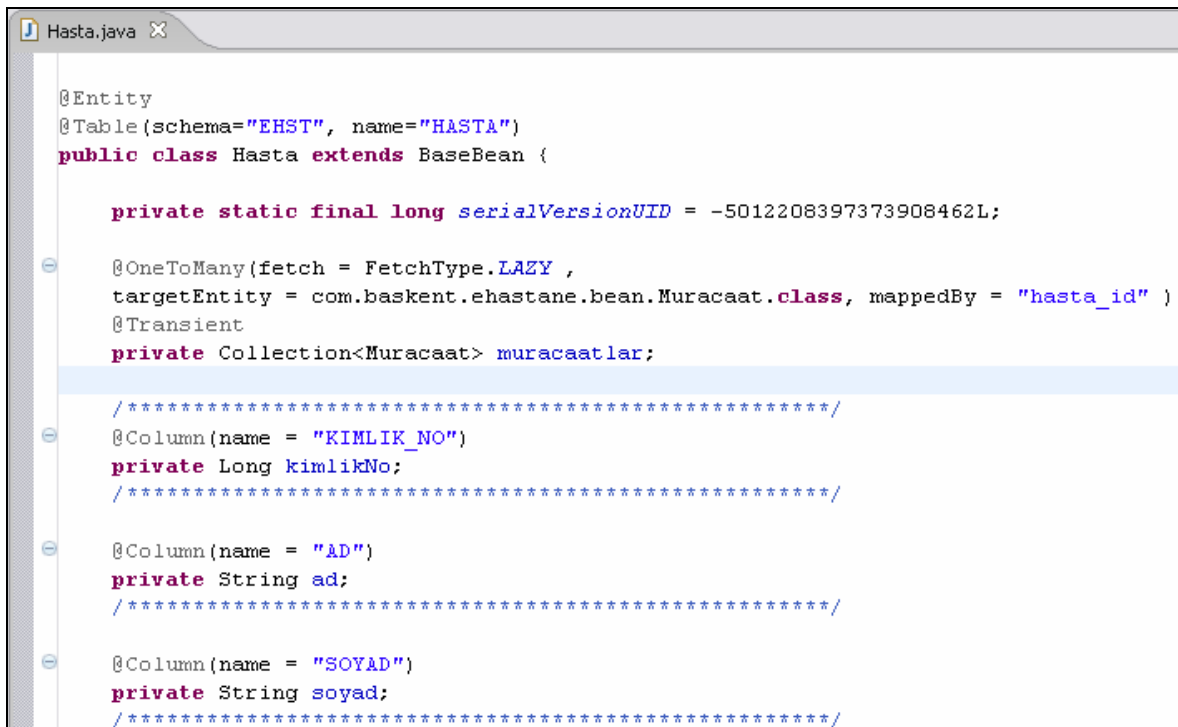
1995 yılında geliştirilen Java programlama dilinin hedeflenen özellikleri ise;

- Nesne tabanlı bir metodoloji kullanması
- Aynı programın değişik işletim sistemleri üzerinde çalışabilmesi
- Diğer nesne tabanlı programlama dillerinin güzel olan yanlarını kolaylıkla kullanılabilir olması
- Bilgisayar ağlarını kendi içinde desteklemesi
- Veritabanı işlemlerini kolaylıkla gerçekleştirebilmesidir (Sun Microsystems, 2007).

### 3.1 JPA

Java programlama dilinin taslağı olan Java Persistence API(Application Programmer Interface) (JPA), programcılara, Java platformu üzerinde ilişkisel verileri yönetme olanağı sağlar. JPA'nın ortaya çıkmasıyla, programcıların, nesnel ve ilişkisel eşlemeler için artık uyumsuz ve standart olmayan modelleri kullanmak zorunda kalmamaları JPA'nın programcılar için getirdiği en büyük faydalardandır (Sun Microsystems, 2007).

Sun şirketinin kendi İnternet sitesinde yayımladığı yazıya göre, JPA'nın programlama dünyasına getirdiği yeniliklerden biri, veritabanları ile Java nesnelere Java programlama dilini destekleyen nesnel/ilişkisel eşleme detayları ve kullanılan metadata açıklayıcı notlarının kullanımıdır. Bunun sayesinde, Java nesnelere veritabanlarıyla Hibernate yada diğer ORMLer'deki (Object/Relation Mapping) gibi ayrı bir XML sınıfına ihtiyaç duymamakta, Java nesne sınıfları içinde tanımlanan metadata açıklayıcı notları ile ilgili bağlantıyı kolayca kurabilmektedir.



```
@Entity
@Table(schema="EHST", name="HASTA")
public class Hasta extends BaseBean {

    private static final long serialVersionUID = -5012208397373908462L;

    @OneToMany(fetch = FetchType.LAZY ,
targetEntity = com.baskent.ehastane.bean.Muracaat.class, mappedBy = "hasta_id" )
    @Transient
    private Collection<Muracaat> muracaatlar;

    /*****/
    @Column(name = "KIMLIK_NO")
    private Long kimlikNo;
    /*****/

    @Column(name = "AD")
    private String ad;
    /*****/

    @Column(name = "SOYAD")
    private String soyad;
    /*****/
```

Şekil 9 JPA Metadata Açıklayıcı Notları



Yapılan arařtırmalar sonucunda, Trkiye’de kullanılan sađlık sistemlerinde Java programlama dilinin yaygın olarak kullanılmaması, kullanılan yerlerde de JPA’nın hi kullanılmamıř olması, getirdiđi faydalar da gz nnde bulundurulunca, e-Hastane sisteminin iinde JPA kullanımını dođal hale getirmiřtir.

### 3.2 JPA zellikleri

JPA’nın bir uygulama iinde getirdiđi faydaları aıklamak gerekirse, JPA’nın Java nesnelere, geliřmiř nesne tabanlı (Object Oriented) kavramları, iř btnlđn, uyumluluđu, byk lekli veri setlerini, diđer sistemlerden farklı olarak mevcut řemaları desteklemesi gibi zellikler JPA’nın diđer ORMler’ le karřılařtırılabilecek zelliklerindendir (Linskey, 2006).

e-Hastane sisteminde JPA’nın kullanılması, veritabanı bađlantısı iin gerekli olan bađlantının XML sınıfları yada bařka bir yol ile yapılmayıřı, Java nesnel sınıfları iinde deđiřkenleri tanımlarken yapılması programcıya, yksek performans kazandırıp, zaman olarak da programcı zerindeki yk azaltmıřtır. Aynı zamanda JPA’nın desteklemiř olduđu JPQL (Java Persistence Query Language) sayesinde yapılması istenen sorgular kolaylıkla dinamik olarak yapılmıřtır. Bu da uygulama iinde herhangi bir parametrenin sorgu cmlenciđine gnderilmesini kolaylařtırmıř ve uygulamaya hız kazandırmıřtır (Keith, 2007).

JPA’nın java.util.logging, log4J, vb. eklentilerle, desteklediđi kayıt tutma(*logging*) zelliđi ile, uygulama alıřırken arka planda yapılan iřlerin, alıřan sorguların kayıtları tutulmaktadır. Bu da ileri vadede yapılabilecek olan bir yanlıřı tespit etmede iře yarayacak olan bir zelliktir. Bu zellik řekil 10’da detaylı olarak gsterilmiřtir.

```
Console X
WebSphere Application Server v6.1 [WebSphere v6.1 Server] WebSphere Application Server v6.1 (WebSphere v6.1)
O INFO [24 12 2007 00:28] org.hibernate.cfg.SettingsFactory.buildSettings(SettingsFactory.
O INFO [24 12 2007 00:28] org.hibernate.cfg.SettingsFactory.buildSettings(SettingsFactory.
O INFO [24 12 2007 00:28] org.hibernate.cfg.SettingsFactory.buildSettings(SettingsFactory.
O INFO [24 12 2007 00:28] org.hibernate.impl.SessionFactoryImpl.<init>(SessionFactoryImpl.
O INFO [24 12 2007 00:28] org.hibernate.impl.SessionFactoryObjectFactory.addInstance(Sessi
O WARN [24 12 2007 00:28] org.hibernate.impl.SessionFactoryImpl.buildCurrentSessionContext
O Hibernate: select meslekO_.ID as ID6_, meslekO_.AD as AD6_ from EHST.MESLEK meslekO_
O Hibernate: select personelO_.ID as ID1_, personelO_.AD as AD1_, personelO_.bolum_ID as bo
O Hibernate: select bolumO_.ID as ID7_0_, bolumO_.AD as AD7_0_ from EHST.BOLUM bolumO_ wher
O Hibernate: select 110_.ID as ID8_0_, 110_.AD as AD8_0_ from EHST.IL 110_ where 110_.ID=?
O Hibernate: select bolumO_.ID as ID7_0_, bolumO_.AD as AD7_0_ from EHST.BOLUM bolumO_ wher
O Hibernate: select 110_.ID as ID8_0_, 110_.AD as AD8_0_ from EHST.IL 110_ where 110_.ID=?
O Hibernate: select bolumO_.ID as ID7_0_, bolumO_.AD as AD7_0_ from EHST.BOLUM bolumO_ wher
O Hibernate: select 110_.ID as ID8_0_, 110_.AD as AD8_0_ from EHST.IL 110_ where 110_.ID=?
O Hibernate: select bolumO_.ID as ID7_0_, bolumO_.AD as AD7_0_ from EHST.BOLUM bolumO_ wher
O Hibernate: select 110_.ID as ID8_0_, 110_.AD as AD8_0_ from EHST.IL 110_ where 110_.ID=?
O Hibernate: select bolumO_.ID as ID7_0_, bolumO_.AD as AD7_0_ from EHST.BOLUM bolumO_ wher
O Hibernate: select 110_.ID as ID8_0_, 110_.AD as AD8_0_ from EHST.IL 110_ where 110_.ID=?
O Hibernate: select altbolumO_.ID as ID0_, altbolumO_.AD as AD0_, altbolumO_.BOLUM as BOLUM
O Hibernate: select kullanicio_.ID as ID4_, kullanicio_.AD as AD4_, kullanicio_.personel_ID
O Hibernate: select hastaO_.ID as ID5_, hastaO_.AD as AD5_, hastaO_.ADRES as ADRES5_, hasta
O Hibernate: SELECT * FROM EHST.KULLANICI WHERE AD = 'doktor2'
```

Şekil 10 JPA Kayıt Tutma Özelliği

### 3.3 JPA Açıklayıcı Notları

JPA'nın açıklayıcı notları sayesinde, veritabanına bağlantı kolayca kurulabilmekte ve bu özelliği ile JPA, mevcut sistemlerle arasında bir fark yaratmaktadır. Bu açıklayıcı notlar, arka tarafta programcıya, Java nesneleriyle ilişkisel veritabanlarının tabloları arasında açıklayıcı bir şekilde, standartlara uygun olarak eşleştirme yapmayı sağlar.

Bu açıklayıcı notlar temel özelliklere sahiptirler. Bu özellikler ;

- javax.persistence.XXX (Kullanılan JPA özelliği)'nin kullanılan Java sınıfına eklenmesi
- Açıklayıcı notlar hem Java sınıflarına, hem de sınıfların içinde bulunan değişkenlere eklenmesi,
- Değişkenlere açıklayıcı not eklenebildiği gibi, bu değişkenleri çağıran getiricilere (getters) de açıklayıcı not eklenebilmesi,

- Açıklayıcı notların “@” öneki ile başlaması

olarak maddelenebilir.

Açıklayıcı notlar kendi aralarında gruplara ayrılırlar. Bu gruplar, bağımsız varlıklar (Entity), veritabanı şema özellikleri, benzerlik (Identity), doğrusal eşleştirme, ilişkisel eşleştirme, bileşim(Composition) vb. Bu gruplara ait açıklayıcı notlardan bazıları aşağıdaki Tablo 2’de gösterilmiştir.

<u>@Entity</u>	Bir sınıfın kalıcı olduğunu belirtir.
<u>@Inheritance</u>	İlgili sınıfı kalıcı yapıcak kalıtım modelini belirtir.
<u>@IdClass</u>	İlgili sınıf için belirleyici (Id) özelliği gösterir.
<u>@NamedQuery</u>	İlgili kalıcı birim üzerinde belirlenen bir JPQL’i tanımlamak için kullanılır.
<u>@Table</u>	İlgili sınıfın eşleştireceği veritabanı tablosunu tanımlar.
<u>@Transient</u>	Kalıcı olmayacak alanları tanımlama için kullanılır.
<u>@OneToOne</u>	Başka bir kalıcı alanla 1-1 ilişkisi olacak alanı tanımlar.
<u>@OneToMany</u>	Başka bir kalıcı alanla 1-N ilişkisi olacak alanı tanımlar.
<u>@ManyToMany</u>	Başka bir kalıcı alanla M-M ilişkisi olacak alanı tanımlar.
<u>@ManyToOne</u>	Başka bir kalıcı alanla N-1 ilişkisi olacak alanı tanımlar.
<u>@GeneratedValue</u>	İlgili alanın bir yaratıcı kullanarak tanımlanacağını belirtir.
<u>@OrderBy</u>	İlgili alanın belirtilen özelliğe göre sıralanması gerektiğini belirtir.

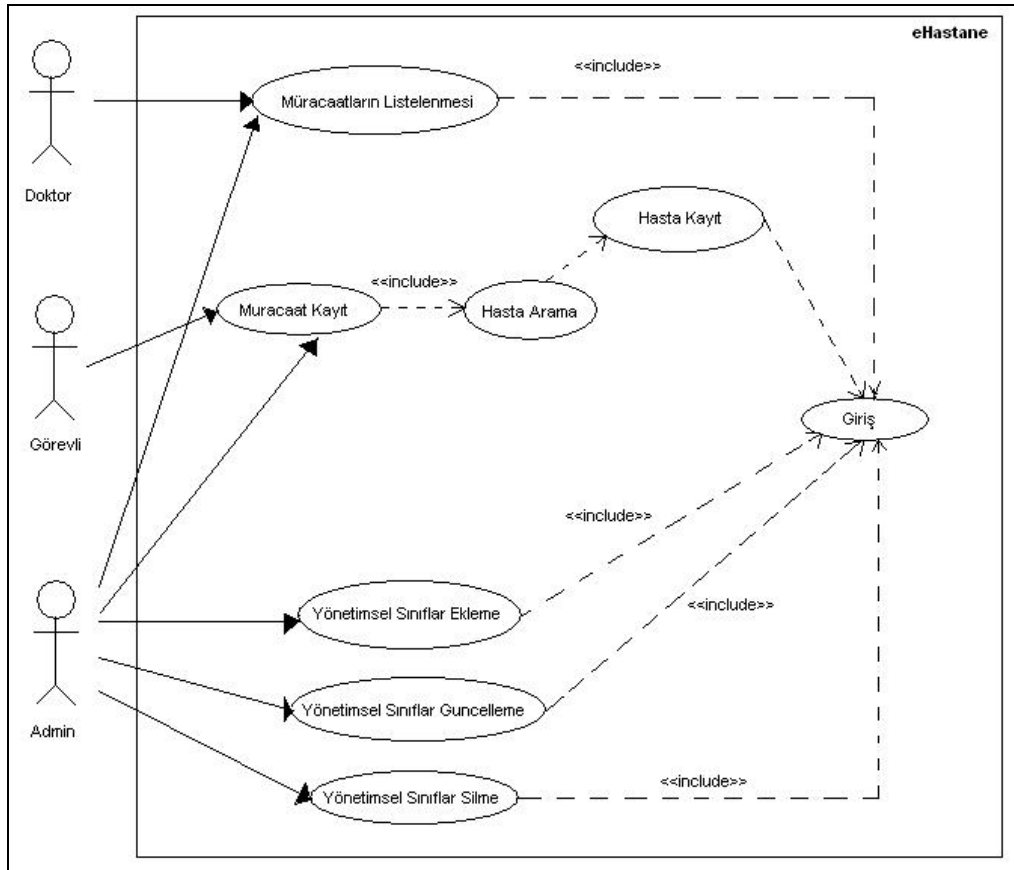
**Tablo 2 JPA Açıklayıcı Not Örnekleri ve Açıklamaları**

#### 4 e-HASTANE SİSTEMİ

Türkiye'nin içinde bulunduğu bu mevcut durum, önceki bölümlerde sayılan sebepler ve hala günümüzde devam eden sorunlar, e-Hastane gibi bir sistemi kullanmayı zorunlu kılmıştır. Bu çalışma sonucunda ortaya çıkan e-Hastane sistemi, hastaların hastaneye kabulünde kullanıcılara kolaylık sağlaması amacıyla geliştirilmiştir.

Bu çalışma süresince 100. Yıl Hastanesi ve Bandırma Kapıdağ Devlet Hastaneleri'nin hasta kayıt kabul modülleri incelenmiştir. Yapılan incelemeler sonucunda ortaya çıkan sonuçlar ele alınarak gerçeğe uygun, ileride geliştirilerek kullanılacak bir yazılım geliştirilmiştir.

Bu yazılımın kabiliyetlerini gösteren kullanım durumları diyagramı aşağıdaki Şekil 11'de gösterilmiştir.



Şekil 11 e-Hastane Sistemi Kullanım Durumları Diyagramı

#### 4.1 Genel Yapı

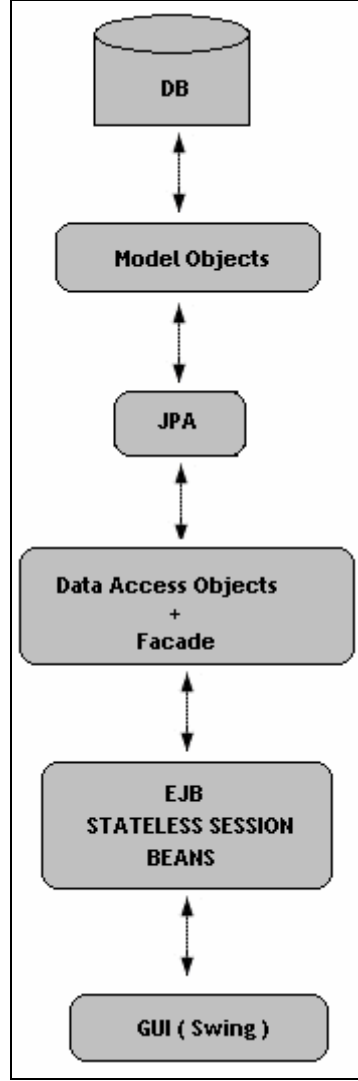
e-Hastane Proje yapısı J2EE mimarisi üzerine kurulmuştur. Programlama dili olarak Java(Swing) ve ORM aracı olarakda JPA (Hibernate) seçilmiştir. Bu sayede geliştiriciler veritabanı üzerindeki ilişkisel yapı üzerinde fazla düşünmeden Java sınıfları üzerinden çalışabilmektedir. Ayrıca CRUD (Create, Update, Delete) işlemleri için SQL yazmak zorunda kalınmaz. JPA ve Java seçiminin e-Hastane sistemine getirdiği faydalara Bölüm 3'te değinilmiştir.

Uygulamanın geliştirilmesi için IBM Rational Application Developer (RAD) kullanılmıştır. IBM' in bu ürünü kullanıcılarına bir çok servisi kolaylıkla kullanma özelliği sunmuş ve yardımcı dokümanlarıyla programcılara maksimum fayda sağlamayı hedeflemiştir. İnternet üzerinden eklenen ekstra paketler yardımıyla programcıya modelleme, görsel düzenleme ve EJB oluşturma imkanı vermiştir.

Veritabanı olarak, teknolojinin de uyumlu çalışması için IBM DB2 Versiyon 8.1 kullanılmıştır. IBM DB2 yardımı ile tablolar kullanıcı ara yüzleriyle kolaylıkla yaratılmış ve bakımı her zaman kolaylıkla yapılmıştır. Tablolar yaratılırken isimlendirmelere önem verilmiş, tablo ve kolon isimlerinin gerçek dünya isimlerine yakın olmasına dikkat edilmiştir.

Ayrıca proje süresince, girilen verileri kontrol etmek, güncellemek veya silmek için Synametrics WinSQL adlı program kullanılmıştır. Bu program sayesinde daha önceden girilmiş olan veriler düzenli bir şekilde kullanıcıya gösterilebilmekte, kullanıcıya ayrıca sorgu yapabilme imkanı sunmakta ve son olarakda kullanıcıya daha önceden çalıştırmış olduğu sorguları arşivleme imkanı sunmaktadır.

Bu yazılımla kullanılarak ortaya çıkan yazılımın mimarisi aşağıdaki Şekil 12'de gösterilmektedir.



**Şekil 12 e-Hastane Sistemi Proje Yapısı**

e-Hastane sistemi proje yapısını anlatan Şekil 12’de bulunan veritabanı (DB), uygulama boyunca sistem tarafından kullanılır ve hasta kayıtları, kullanıcı şifreleri ve müracaat vb. bilgileri tutar.

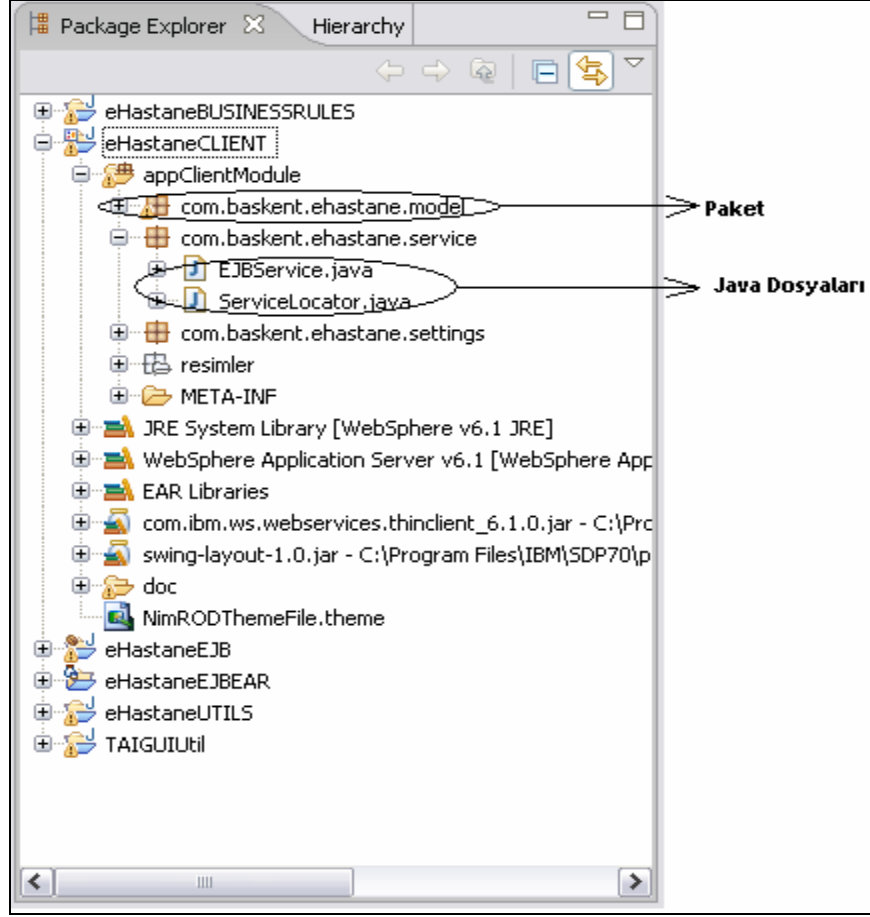
Proje içinde yaratılmış olan Java nesne sınıflarıyla ve aynı nesne sınıfları içinde tanımlanan JPA tanımlamalarıyla, ilgili sınıflar veritabanı üzerinde ilgili tablolara tanıtılır. Bu tanımlama sınıf içerisinde yapıldığı için JPA’nın kullanılmadığı Hibernate uygulamalarında olduğu gibi her sınıf için ayrı XML dosyalarına ihtiyaç olmamıştır. Bu da performans ve kullanım açısından programcıya büyük zaman kazandırmaktadır.

Program içinde kullanılan Stateless Enterprise Java Beanler'e (EJB) kullanıcı tarafından erişmek için Facade\* sınıfı yazılmıştır. Program içinde bulunan her Facade sınıfının metodu, bean içinde bir yönteme denk gelmektedir. Bu Facade sınıfının yöntemleriyle, sistem EJB' ye gider ve gerekli bilgileri veritabanından alır ve kullanıcıya göstermek üzere Kullanıcı Ara Yüzü'ne (GUI) gönderir ve kullanıcının istediği bilgiyi görmesini sağlar.

#### **4.2 Proje Dosyaları ve Paketleri**

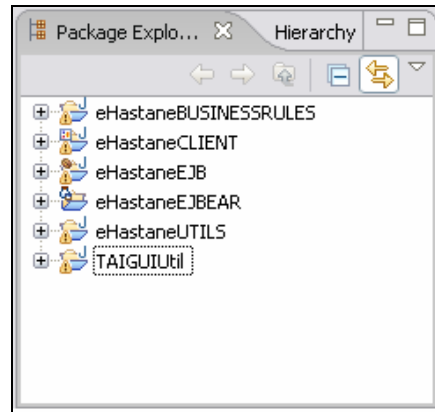
e-Hastane uygulaması geliştirilirken IBM RAD üzerinde yedi adet proje yaratılmıştır. Bu projelerin her biri programcıya, programlama sırasında olduğu kadar programlama sonrasında da (bakım esnasında), kolaylık sağlar. Bu proje dosyaları içinde bir çok paket bulunmaktadır. Bu paketler kendi aralarında da gruplara ayrılmışlardır. Bu gruplar, içinde barındırdıkları Java sınıflarına göre gruplandırılmaktadırlar. Paketlerin isimleri Java standartlarına uygun bir biçimde verilmiş olup "com.baskent.ehastane.xxxxx" olacak şekilde isimlendirilmiştir. Bu paketlerden bir kaçını aşağıdaki şekilde görebilirsiniz.

\* Nesne tabanlı sistemlerle beraber kullanılan yazılım mühendisliğinin bir tasarım örüntüsü.



**Şekil 13 e-Hastane Sistemi Paket Yapısı**

Yukarıdaki paketlerin ait oldukları proje dosyalarının kullanımı projenin geliştirme esnasında programcıya aradığını bulma konusunda da çok yardımcı olmaktadır. Bu proje dosyalarının isimlerini içeren şekil aşağıda gösterilmektedir.



**Şekil 14 e-Hastane Sistemi Proje Dosyaları**



**eHastaneBUSINESSRULES** : Bu proje dosyası içinde bir çok paketi barındırmaktadır. Java sınıflarının ve dolayısıyla JPA tanımlamalarının ve Facade'ın yani kısaca iş tanımlamalarının bulunduğu bir proje dosyasıdır. Bu dosya içinde iş ile alakalı bütün kurallar tanımlanır ve proje işleyişinde takip edilmesi gereken bütün kurallar burada tutulur. Bu sınıf aynı zamanda birde sınıf diyagramını kendi içinde barındırmaktadır. Bu sınıf diyagramı sistem içinde kullanılan bütün sınıfları ve bağlantılarını göstermektedir.

**eHastaneCLIENT** : Bu proje dosyası kullanıcıya gösterilecek olan görsel sınıfları kapsar. Aynı zamanda programın çalışması esnasında gerekli ayarların bulunduğu paketler ve dosyalar da yine bu sınıf içinde tanımlanmıştır. Kullanıcıya ekranda gösterilecek olan ikonlar yine bir klasör yapısı içinde bu proje dosyasında gösterilmektedir. Kısacası, ara yüz ve kontrol sınıflarını yani kullanıcıyla alakalı olan birçok dosyanın tutulduğu proje dosyasıdır.

**eHastaneEJB** : Sunucu üzerinde deploy edilecek EJBler' i içeren projedir. Bu proje içinde yaratılmış olan EJB Beanleri tutulmaktadır.

**eHastaneEJBEAR** : eHastaneEJB projesi yaratılırken otomatik olarak yaratılan bir projedir. Bütün gerekli kodlar yazıldıktan sonra, bu proje sunucu üzerine yüklenir ve EJB projesi deploy edildikten sonra kullanıcı, istemci tarafından projeyi çalıştırır.

**eHastaneUTILS** : Metin ve Swing\* ile ilgili bir çok yardımcı sınıf bu proje içinde tutulmaktadır. Diğer projelerle beraber yer almayışının tek sebebi, burada yer alan dosyaların, sınıfların, her projede ortak olarak kullanılabilir olmasıdır. Böylelikle internalization özelliği de projeye kazandırılmış olunur.

**TAIGUIUtil** : eHastaneUTILS projesi ile aynı göreve sahip olan bu proje daha zengin sınıflara sahip olmakla beraber, çoğunlukla tablo yapılarıyla ilgili sınıflara ve yöntemlere yer verir.

\* Java tabanlı ekranları yapmak için kullanılan bir araç takımı.

### 4.3 e-Hastane Yetki Sistemi

e-Hastane sisteminin amaçlarından, hedeflerinden biri olan hasta bilgilerinin güvenliğinin sağlanması adına e-Hastane sistemi içinde üç ayrı kullanıcı tipi tanımlanmıştır. Bu üç değişik kullanıcı tipleri sayesinde, hastane içinde herkes sadece üstüne düşen görevi yapacak, hasta bilgileri gizli kalacak ve zamandan da tasarruf sağlanacaktır. Bu kullanıcı çeşitleri; kullanıcı, yönetim ve doktor'dur.

e-Hastane sistemine ilk giriş yapılırken kullanıcıya, kullanıcı adı ve şifre sorulur, eğer kullanıcı her iki bilgiyi de doğru girerse bir bilgi mesajı ile sisteme girmesine izin verilir, eğer iki bilgiden birini yanlış girdiyse kullanıcıya bir hata mesajı verilerek ilgili bilgileri tekrar girmesi istenir.



Şekil 15 e-Hastane Sistemi Giriş Ekranı



Şekil 16 e-Hastane Sistemi Giriş Kullanıcı Mesajları

#### **4.3.1. Yönetim**

Yönetim tipinde olan bir kullanıcı, e-Hastane sisteminde kullanılacak olan bir çok veriyi girebilir, güncelleyebilir ve hatta silebilir. Yönetim pozisyonundaki bir kullanıcının bölüm, alt bölüm\*, personel, il, kullanıcı, meslek vb. ekranlarda veri girme, güncelleme ve silme yetkisine sahip tek kullanıcıdır.

#### **4.3.2. Kullanıcı**

Kullanıcı tipinde olan bir kullanıcının yapabileceği işlemler sistem tarafından kısıtlanmıştır. Buna göre kullanıcı sadece hasta arama, hasta kayıt ve muracaat alma işlemlerini yapabilir. Aynı zamanda bir muayene günü sonunda girilmiş olan bütün müracaatları kapatmak için yetkilendirilmiş olan kullanıcı tek bir komutla sisteme bütün müracaatları kapattırabilir.

#### **4.3.3. Doktor**

Doktor tipinde bir kullanıcı, kullanıcı tarafından, kendi alt bölümü için girilmiş bütün müracaatları sırasıyla görür. Her bir müracaatı ayrı ayrı kapatma yetkisine sahiptir. Doktor, isterse kendisine ait müracaatların hepsini bir seferde kapatabilir yada daha önceden kapanmış olan müracaatları görüntüleyebilir.

### **4.4 e-Hastane Müracaat Kayıt Kabul Modülü**

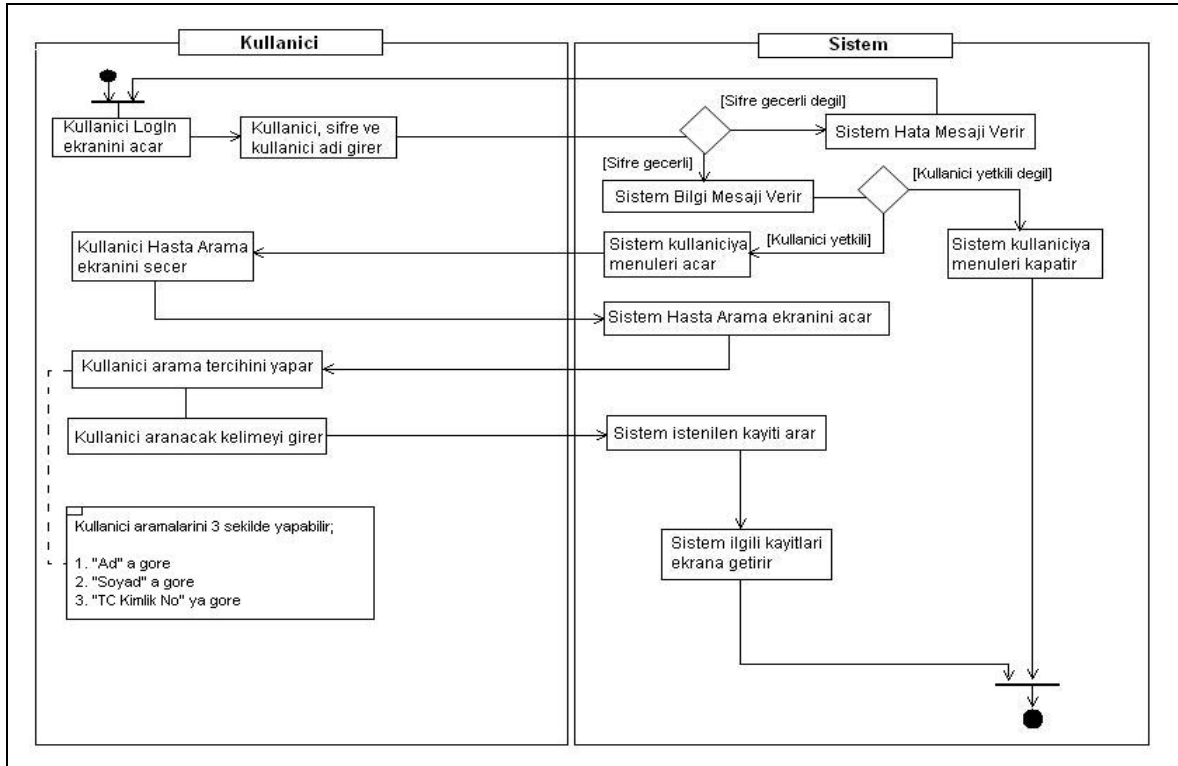
e-Hastane sisteminin temel adımlarından olan Müracaat Kayıt Kabul Modülü, üzerinde titiz bir çalışma gerektiren modüldür. Bu modül içinde, hastanın hastaneye, ilk sefere mahsus olmak kaydıyla sağlık karnesiyle beraber, gelip Hasta Kabul bölümünden giriş kaydını yaptırması beklenmektedir. Bu bölümde

\* e-Hastane sistemi içerisinde bir bölüme ait olan alt bölüm

görevli personel tarafından alınan kayıt, hastanın bundan sonraki hastane ziyaretlerinde de kullanılacak olan kayıttır. Bu aşamada alınacak olan bilginin kalitesinin önemi önceki bölümlerde anlatılmıştır. Bu aşamada görevlinin (kullanıcının) yapabileceği üç temel işlem vardır; Hasta Arama, Hasta Kayıt, Müracaat Kayıt.

#### 4.4.1. Hasta Arama

Bu aşamada görevli personel (kullanıcı), gelen hastanın bilgilerinin daha önceden sisteme girilip girilmediğini kontrol eder. Bu kontrol etme işlemi için, kullanıcının doğru kullanıcı adı ve şifreyi girmiş olması beklenmektedir. Uygun giriş yapıldıktan sonra, kullanıcı hastayı üç kriterle göre sistem içinde arayabilir. Bu kriterler sırasıyla; Ad, Soyad ve T.C. Kimlik Numarasıdır. Bu kriterlerden birini seçmek için kullanıcının ilgili radyo düğmesini seçmesi yeterli olacaktır. Seçilen kriterin ardından, aranmak istenen metin, ilgili metin kutusuna yazılır ve metin yazıldıkça sonuçlara arasından eleme yapılarak istenen kayıt ekran üzerinde bir tabloda kullanıcıya gösterilir. Bu bölüm ile ilgili aktivite diyagramı aşağıdaki şekilde daha detaylı olarak gözükmemektedir.

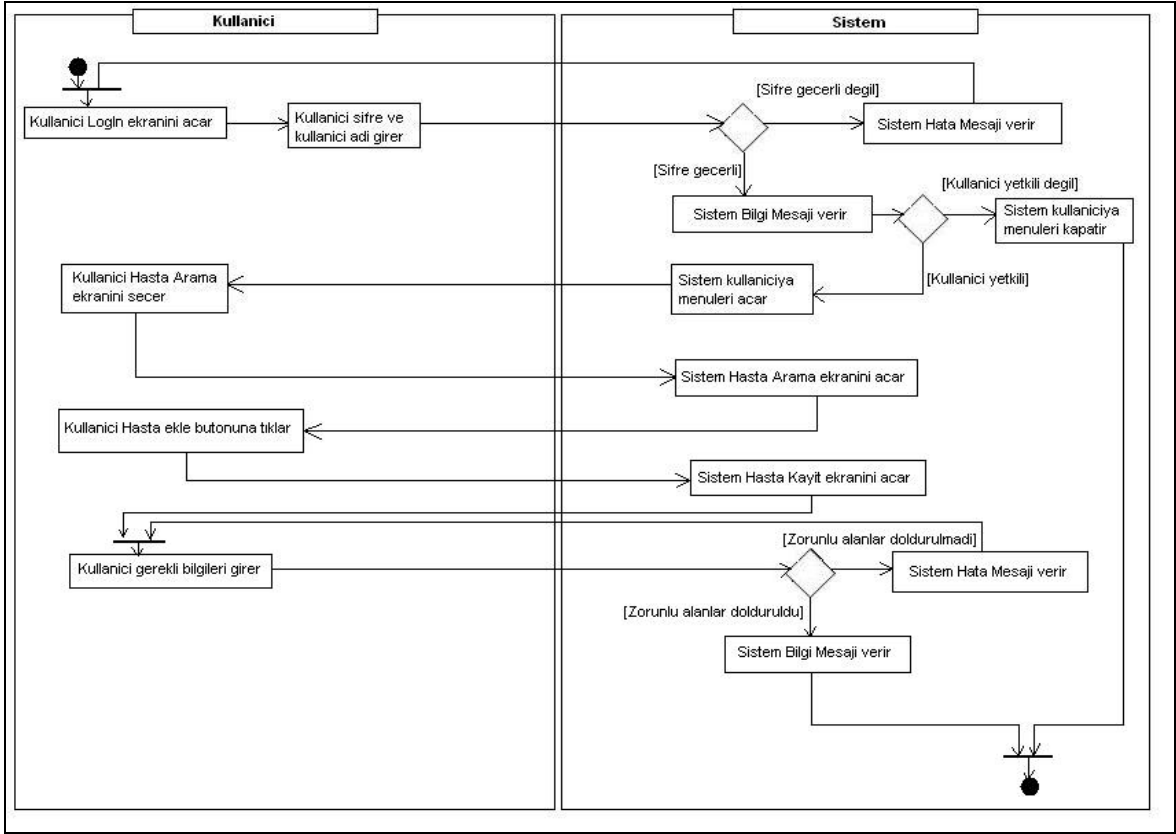


Şekil 17 e-Hastane Sistemi Hasta Arama Aktivite Diyagramı

#### **4.4.2. Hasta Kayıt**

Bu bölüm, hasta arama bölümü ile bağlantılı bir bölümdür. Görevli personelin yaptığı arama sonucunda sistemde bilgilerini bulamadığı hasta, bu bölümde sisteme tanıtılır. Tanıtma işlemi için, ilgili kullanıcı, sisteme doğru kullanıcı adı ve şifreyle giriş yaptıktan sonra hasta arama ekranını seçer. Bu ekran üzerinde bulunan düğmeleri kullanarak, kullanıcı yeni hasta eklemek için *Hasta Kayıt* ekranını açar. Bu ekranda kullanıcıdan müracaat eden hastanın bilgilerini girmesi istenir. Bu bölümde zorunlu doldurulması gereken alanlar mevcuttur. Bunlardan bir tanesi de, T.C. kimlik numarasıdır. Bu numaranın ayrıca 11 karakterden fazla girilmemesi istenir. Kullanıcının hastadan istediği bilgiler içinde, hastanın adı, soyadı, doğum tarihi, uyruğu, doğduğu il, sicil numarası, cinsiyeti, adresi, kan grubu vb. gibi bilgiler bulunmaktadır. Ayrıca bu ekran üzerinden görevli personel (kullanıcı) istenildiği takdirde gerekli hasta bilgisi güncellemelerini de yapabilir.

Bu tarz bilgilerin, bu safhada hastadan doğru olarak alınması çok önemlidir. Bunun sebebi, bir sonraki adımlarda, e-Hastane sisteminin doktorlar için olan kısmında, işlemlerin değişik kan grupları, cinsiyetler, yaşlar vb. gibi bilgiler için değişik sonuçlar doğurabilecek olmasıdır. Bu bölümde alınacak bir yanlış bilgi, hastanın tanısının koyulmasından, yanlış tedavisine kadar hastayı kötü yönde etkileyebilir. Bu ekran ile ilgili detaylı bilgi aşağıda bulunan aktivite diyagramında gösterilmektedir.



**Şekil 18 e-Hastane Sistemi Hasta Kayıt Aktivite Diyagramı**

#### 4.4.3. Müracaat Kayıt

Bu çalışmanın ana işlevlerinden biri olan müracaat kayıt, bu bölümde ince ayrıntılarıyla, gerçek hayatta bir hastaneye uyum sağlayacak şekilde analiz edilmiş ve uygulamaya koyulmuştur.

Görevli personel kendisine sistemin yöneticisi tarafından verilen kullanıcı adı ve şifreyi doğru olarak girdikten sonra sisteme giriş yapar. Bu giriş aşamasından sonra kullanıcıya yetkisine göre aktif olan İşletim menüsünden Hasta Arama / Kayıt'ı seçer ve kullanıcıya sistem tarafından hasta arama ekranı gösterilir. Kullanıcı burada müracaat kaydı yapılması istenen hastayı bulur ve arkasından bu hastanın gösterildiği satır üzerine iki kere tıklayarak yada yukarıda bulunan Müracaat Ekle düğmesini tıklayarak, müracaat ekranını açar.

Açılan müracaat ekranının üst kısmında, müracaatı yapılacak olan hastanın bilgileri kullanıcıya gösterilmektedir, hemen alt kısmında ise, yapılacak olan müracaatın detayları yer almaktadır.

Hasta Bilgileri		
TC Kimlik No :	99283282233	Kan Grubu : A Rh +
Ad :	TARKAN	Sicil No : 2323552
Soyad :	TEVETOĞLU	DogumYeri : Giresun
Meslek :	REKLAMCI	Cinsiyet : ERKEK
		Dogum Tarihi : 09/12/2007
		Calisma Durumu : EMEKLI

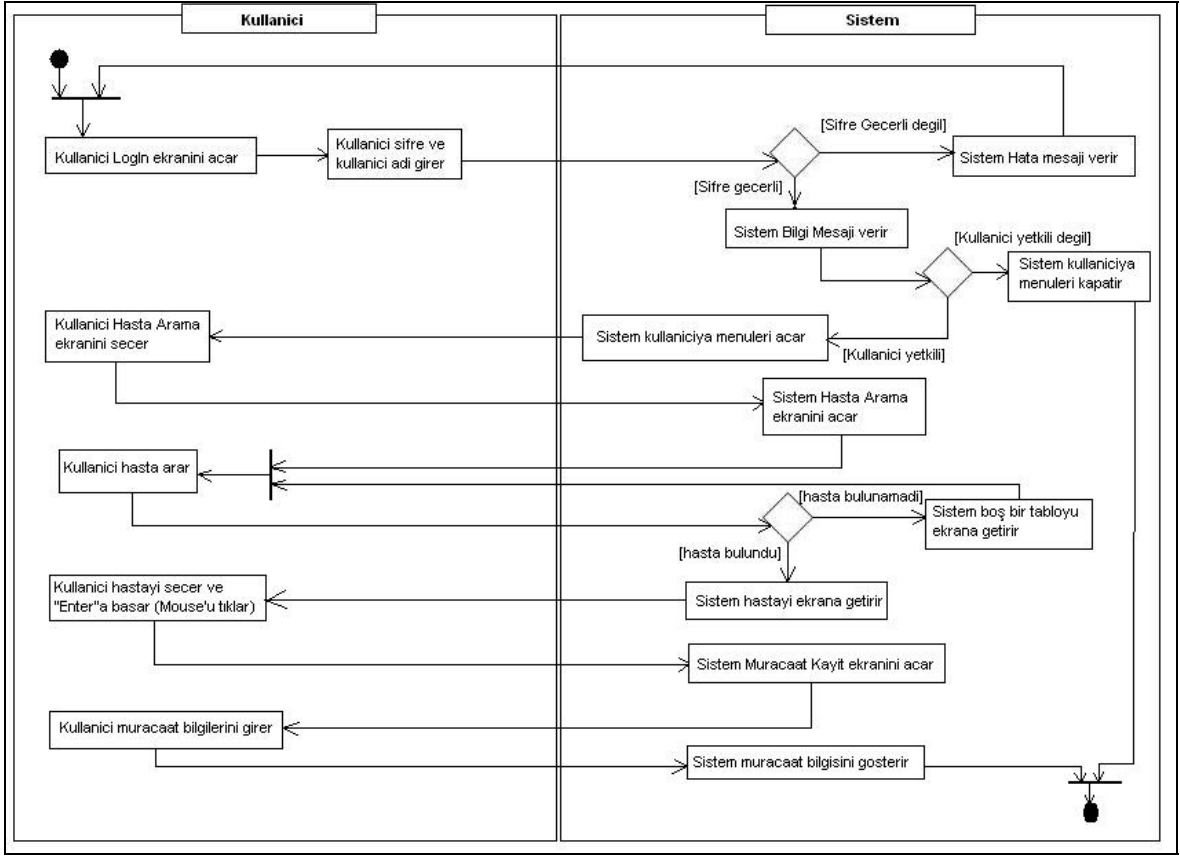
  

Muracaat Bilgileri	
Bolum :	ORTOPEDI
Alt Bolum :	ORTOPEDI1
Doktor :	HAKAN UYDUR
Bakilabilecek Hasta Sayisi :	5
Yatak Sayisi :	5
Basvuru Tarihi :	Aralık 22, 2007

Muracaat

**Şekil 19 e-Hastane Sistemi Müracaat Ekranı**

Kullanıcı, hastanın istediği bölüme müracaat yapmak için ilgili bölümü seçer ve bunun karşılığında sistem otomatik olarak seçilen bölüme ait, alt bölümleri ve aynı zamanda alt bölümlere ait doktorları ilgili çoktan seçmeli kutular içine listeler. Kullanıcı, hastanın istediği alt bölümün müsait olması durumunda o alt bölümü veya en uygun olan alt bölüme hastanın kaydını yapar. Bu kayıt işleminden sonra kullanıcıya bir etiket bastırılır. Bu etiket, ileriki safhalarda geliştirilerek bir bar kod olarak bastırılabilir. Bu akış ile ilgili aktivite diyagramı aşağıdaki şekilde detaylı olarak gösterilmektedir.



**Şekil 20 e-Hastane Sistemi Müracaat Kayıt Aktivite Diyagramı**

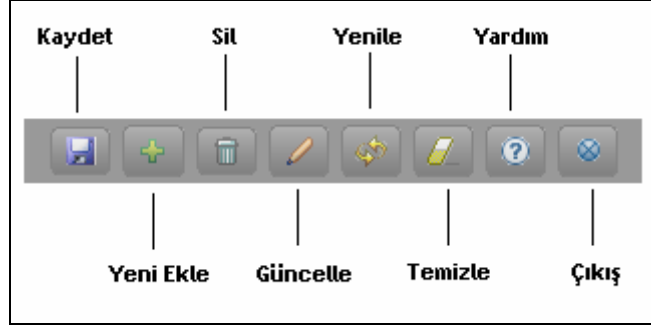
#### 4.5 Yönetim Modülü

Bu kısım, sadece sistemin yönetici olarak tanımladığı kişilere açıktır. Bu kısımda yönetici olarak tanımlanan kişiler, e-Hastane sistemi içerisinde kullanılacak olan ortak verilerin girişi bu modülden yapılmaktadır. Kullanıcı, bu modülün değişik ekranlarından giriş yaparken, her ekranda bulunan düğme panelini kullanır.

Bu panel üzerinde, yönetim modülünün fonksiyonlarını gerçekleştirecek olan çeşitli düğmeler yer almaktadır.

Örnek vermek gerekirse bu düğmeler ; Ekle, sil, güncelle, yenile vb. Bu düğmelerin bir kısmı herhangi bir kayıt seçili olduğu zaman aktif olmaktadır.



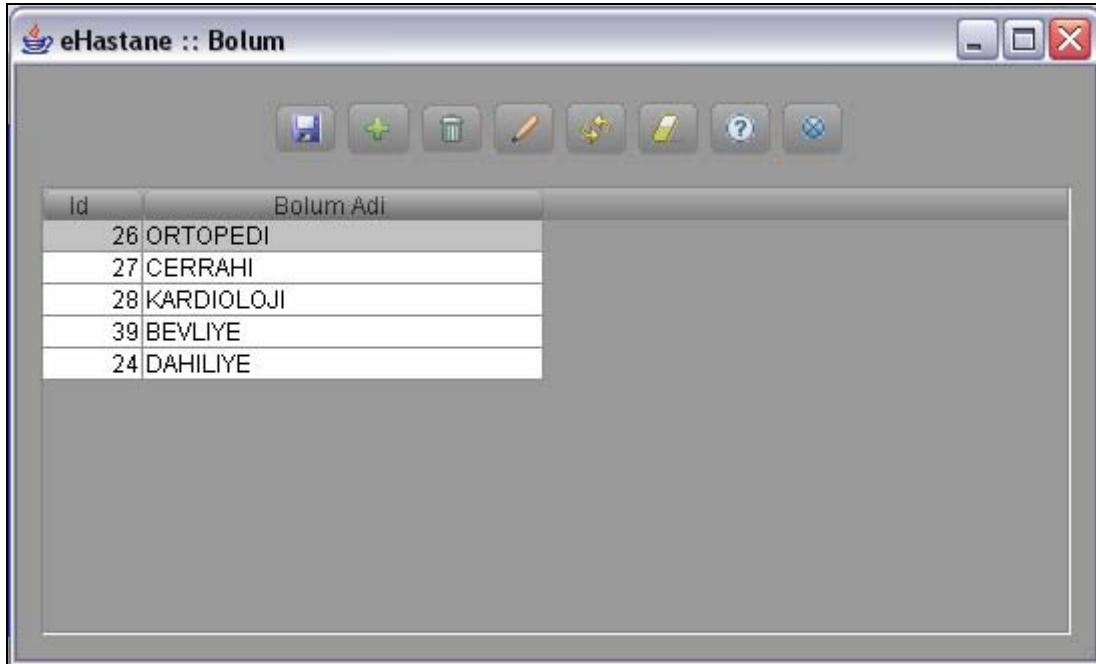


Şekil 21 e-Hastane Sistemi Düğme Paneli

Yönetim modülü kendi içinde bir çok ekranı barındırmaktadır. Bu ekranı kullanarak kullanıcılar, sistemde kullanılacak olan ortak verilerin girişlerini yapabilirler.

#### 4.5.1. Bölüm Ekranı

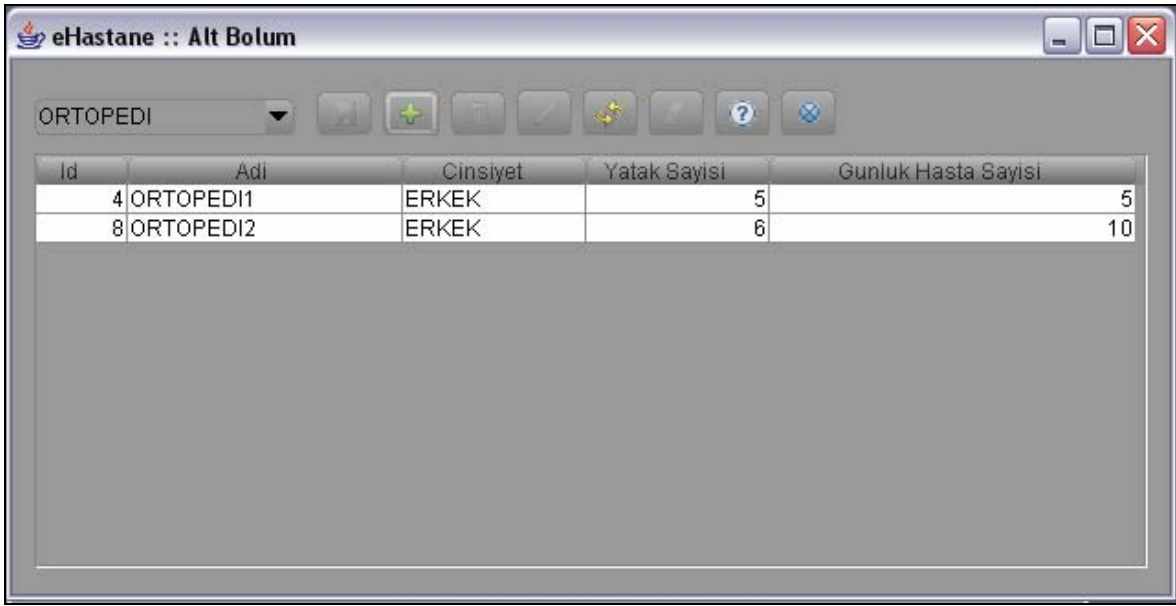
Bu ekranda kullanıcı, e-Hastane sistemi içinde müracaatların başvurusunun yapılacağı bölümlerin bilgilerini girerler. Kullanıcıdan, bu ekranda sadece girilecek olan bölümlerin isimleri istenir. Girilmiş olan bilgiler sistem tarafından büyük harfe çevrilir ve veritabanında ilgili tabloya kaydedilir.



Şekil 22 e-Hastane Sistemi Bölüm Ekranı

#### 4.5.2. Alt bölüm Ekranı

Bu ekranda kullanıcı, öncelikle girmek istediği alt bölümün bölümünü seçer. Daha sonra sistem veritabanından ilgili bölüme ait alt bölümleri bir tablo içine listeler. Bu ekranda kullanıcı, dönen liste üzerinde istediği değişikliği yapma hakkına sahiptir. Kullanıcı isterse, dilediği bölüme altbölüm girebilir, güncelleme yapabilir yada istediği kaydı veritabanından sistemi kullanarak silme hakkına sahiptir. Ekranda kullanıcıya sadece alt bölümlerin veritabanındaki numaraları (id) ve isimleri gösterilir.



Id	Adı	Cinsiyet	Yatak Sayısı	Günlük Hasta Sayısı
4	ORTOPEDI1	ERKEK	5	5
8	ORTOPEDI2	ERKEK	6	10

Şekil 23 e-Hastane Sistemi Alt Bölüm Ekranı

#### 4.5.3. Kullanıcı Ekranı

Kullanıcı ekranı, sisteme yönetici olarak giren kullanıcı tarafından kullanılır. İlgili kullanıcı, sisteme giriş yapacak olan kullanıcıların kullanıcı adlarını ve şifrelerini tanımlar. Böylelikle sisteme giriş yapabilecek kullanıcılar tanımlanır. Sisteme giriş yapacak kullanıcıların tanımlandığı bu ekranın detaylı görüntüsü aşağıda gösterilmektedir.

Id	Kullanici Adı	Sifre	Personel	Yetki
7	kullanici	kullanici	KEMAL OLTU	KULLANICI
6	yonetim	yonetim	CAGRI ELIBOL	ADMIN
8	doktor1	doktor1	CAGRI ELIBOL	DOKTOR
9	doktor2	doktor2	KEMAL OLTU	DOKTOR
10	doktor3	doktor3	HAKAN UYDUR	DOKTOR
11	doktor4	doktor4	MURAT TOK	DOKTOR
12	doktor5	doktor5	CAN KANAT	DOKTOR
13	doktor6	doktor6	SAFFET SUNA	DOKTOR

**Şekil 24 e-Hastane Sistemi Alt Bölüm Ekranı**

Kullanıcı bu ekranda, istediği kullanıcı adı ve şifresini 20 karaktere kadar girebilmekte, sisteme daha önceden girdiği personellerden birini seçerek, üç yetki tipinden birini seçerek kullanıcı tanımlama işlemini tamamlar.

#### **4.5.4. Meslek Ekranı**

Bu ekran da, bölüm 4.5.1’de tanımlanan Bölüm Ekranı gibi, e-Hastane sistemi içinde kullanılacak olan mesleklerin girişini sağlamakta kullanılır. Meslekler için herhangi bir kısıt getirilmemiştir. Tek dikkat edilmesi gereken, isimlerin 20 karakteri geçmemesidir. Bu ekranda kullanıcıya düğme panelinde bulunan düğmelerin fonksiyonları haricinde bir işlem yaptırılmaz. Girilen meslek isimleri büyük harfe çevrilerek veritabanında ilgili tabloya kaydedilir. Bu meslekler sistem içinde Personel Ekranı, Hasta Ekranı gibi ekranlarda kullanılır.



**Şekil 25 e-Hastane Sistemi Meslek Ekranı**

#### **4.5.5. Personel Ekranı**

Personel Ekranında, yetkili kullanıcı (Yönetici) doğru şifre ve kullanıcı adıyla giriş yaptıktan sonra, sistem içinde kullanılmak üzere, ilgili hastanede çalışacak personeli tanımlar. Personel tanımlarken verilerin dikkatli ve doğru olarak girilmesi dikkat edilmesi gereken en önemli noktadır. Bu ekranın detaylı bir görüntüsü aşağıdaki şekilde gösterilmektedir

**eHastane :: Personel**

Yeni Personel

TC Kimlik No :  Cinsiyeti : ERKEK

Adi :  Sicil No :

Soyadi :  Bolumu : ORTOPEDI

Dogum Tarihi : Aralık 22, 2007 Meslek : BILG. MUHENDISI

Dogum Yeri : Adana Uzmanlik : Uzmanlik Yok

Personel Listesi

Id	Adi	Soyadi	Sicil No	TC Kimlik No
9	CAGRI	ELIBOL	1234	12345678901
10	KEMAL	OLTU	3242	23213123123
11	HAKAN	UYDUR	34222	43434231131
12	MURAT	TOK	31212	23123123123
13	CAN	KANAT	52112	12312312222
14	SAFFET	SUNA	83422	53223432343

**Şekil 26 e-Hastane Sistemi Personel Ekranı**

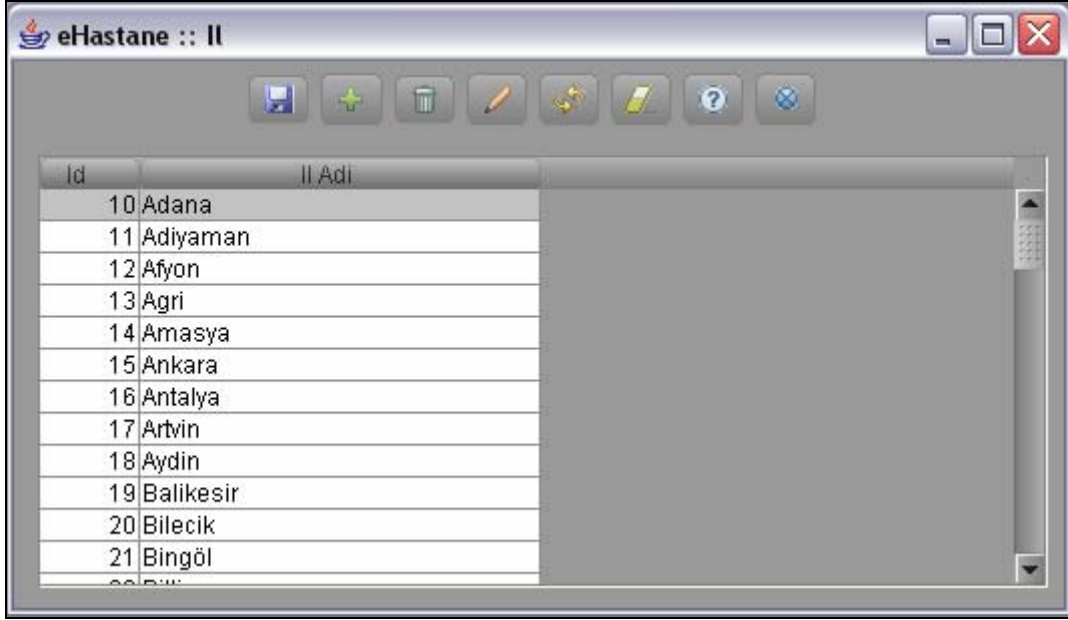
Bu ekranda daha önceden girilen bilgilerin üzerine tıklandığında, ilgili kaydın bilgileri sistem tarafından veritabanından alınarak, ekranın üst bölümünde bulunan ilgili metin kutularına ve alanlara yerleştirilir.

Doktor olarak girişi yapılacak personelin uzmanlığı, bu ekranda girilmek zorundadır. Girilmediği takdirde bu doktor (personel), Müracaat Ekranında, gözükmeyecektir.

#### 4.5.6. İl Ekranı

Bu ekranda, bölüm 4.5.1'de bulunan Bölüm Ekranı gibi, e-Hastane sistemi içinde kullanılacak olan illeri belirlemek için kullanılır. Burada girilecek olan iller, Hasta

Kayıt ve Personel Ekranlarında, hastanın doğum yeri ve adres bilgisi olarak kullanıcının karşısına çıkacaktır. Kullanıcının girmesi ve dikkat etmesi gereken tek nokta, il ismi girerken 20 karakteri geçmemektir. Bu ekranın sistem içindeki görüntüsü aşağıdaki şekilde gösterilmektedir.



Şekil 27 e-Hastane Sistemi İl Ekranı

#### 4.6 Poliklinik Modülü

Bu modül, mevcut e-Hastane sisteminin içinde mutlak suretle bulunması gereken modüllerden biridir. Mevcut sistemde, bu modül içerisine sadece doktorun kendi alt bölümüne yapılan müracaatları görme ve ilgili müracaatları kapatma işlemi bulunmaktadır. İleride bu modüle eklenebilecek olan servisler arasında sevk, laboratuvar, hasta işlem, hasta tanı vb. yer almaktadır. Bu modül kullanıcının hastayı sisteme kaydetmesinden, doktorun hastayı, işlemlerini yapmak üzere muayene etmesine kadar olan zaman diliminde olacak işlemleri kapsamaktadır.

##### 4.6.1. Müracaat Listeleme Ekranı

Bu ekranda sisteme doğru kullanıcı adı ve şifreyle giriş yapan kullanıcı doktor, ekranın üst bölümünde kendisine ait bilgileri görmektedir. Kullanıcı kendi bilgilerinin yanı sıra, ekranın alt bölümünde, Açık Müracaatlar adlı sekmede de, o

gün içerisinde kendi bölümüne müracaatta bulunmuş hastaları sırasıyla görmektedir. Kullanıcıya (doktora), hastaları muayene ettikten sonra, ilgili tablo satırında gösterilmiş “Kapat” isimli kolondaki, tik işaretini tıklayarak, ilgili müracaatı kapatma şansı verilmektedir. Kapanan müracaatlar ekranda bir sekme içinde tutulan “Kapanan Müracaatlar” isimli bölümde kullanıcıya gösterilmektedir.

The screenshot shows a window titled "eHastane :: Muracaat Listele". It contains a form for doctor information and a table of applications.

**Doktor Bilgileri**

Ad :	KEMAL	TC Kimlik No :	23213123123
Soyad :	OLTU	Cinsiyet :	ERKEK
Sicil No :	3242	Dogum Tarihi :	Aralık 5, 1975
Bolum :	KARDIOLOJİ	Alt Bolum :	KARDIOLOJİ1

**Muracaatlar**

Acık Muracaatlar | Kapanan Muracaatlar

Muracaat Kapat

Sıra No	Hasta	Basvuru Tarihi	Ana Doktor	Kapat
1	TARKAN TEVETOĞLU	23.Ara.2007	KEMAL OLTU	<input checked="" type="checkbox"/>

Şekil 28 e-Hastane Sistemi Müracaat Listeleme Ekranı

## 5 SONUÇ VE ÖNERİLER

### 5.1 Sonuçlar

Bu çalışmada hastane sistemleri incelenmiş ve ihtiyaç duyulan, hastaların, doktorların ve hastanede çalışan görevlilerinin beklentilerini sağlayacak, hızlı ve kolay erişilebilir, hizmet standartları yüksek, yeni Java teknolojilerini kullanan bir uygulama yazılımı geliştirmek hedeflenmiştir. Bu durumda Türkiye’de halen geçerli olan, daha çok sağlık kuruluşu ve daha çok doktorla hizmet verme eğilimi yerine, daha kaliteli ve iyi hedeflenmiş bir sağlık hizmeti sunumu esas amaçtır.

Ermütlu ve ark. yapmış oldukları “Ayaktan Hasta Bakımında bilgi Teknolojisi Kullanımı” adlı çalışmada belirttikleri gibi, Türkiye hızlı bir şekilde bilişim teknolojilerini kullanma sürecine girmiştir. Kamu kuruluşları ve özel kuruluşlar bilişim teknolojilerinin kullanımıyla geliştirilmiş sistemlerin kullanıma geçirilmesini için yoğun talep içindedirler. Yoğun bir talep olmasına rağmen ilgili sistemler için ortak bir görüş bulunmamaktadır. Sağlık Bakanlığı Sağlık Projeleri Genel Koordinatörlüğünün (SPGK) bu konudaki değişik proje ve araştırmaları yeteri kadar tanıtılmamıştır. Türkiye günden güne, birbiriyle iletişim kuramayan, sağlık hizmeti sistemleriyle kirlenmektedir.

Her gün gelişen bilişim teknolojileri, hastanelerinde yazılım kullanan kurumları, hasta beklentilerini karşılama açısından çok yakından ilgilendirir. Çalışma sonunda, bu beklentileri karşılamak adına, JPA kullanılarak bir uygulama geliştirilmiştir. Bu uygulamada, bir hastanede bulunması gereken modüllere (Poliklinik, laboratuvar, eczane, fatura vb.) adım ve zemin oluşturması için, hastanın kabul ve müracaat safhalarını içeren Müracaat Modülü analiz edilmiştir. İlgili uygulama, Bölüm 3’te detaylı olarak bahsedilen, Java programlama dili ile geliştirilmiştir.

Mevcut hastanelerde kullanılan yazılımların Java dışında Smalltalk, C, C++ vb. nesne tabanlı dillerle yazılmış olması ve ORM olarak JPA kullanımına rastlanmamış olması, JPA’nın bu uygulamada seçilmiş olmasının sebeplerindedir. JPA seçiminin bir hastane sistemine kazandıracığı fayda ise,



Bölüm 3.1’de de bahsedildiği üzere, hasta beklentilerinden biri olan, müracaatta hızı (minimum zaman kaybını), beraberinde getirmesidir. JPA’nın sahip olduğu metadata açıklayıcı notlarının kullanımıyla e-Hastane sistemi, Java nesnelерinin veritabanlarıyla Hibernate yada diğer ORMLer’deki (Object/Relation Mapping) gibi ayrı bir XML sınıfına ihtiyaç duymadan ilgili bağlantıyı kolayca kurabilmiş olması ile hastaların beklentisi olan hızı bünyesine katmıştır.

Bu tez çalışmasında, ortaya çıkan e-Hastane sistemi üzerinde bir durum çalışması uygulanarak, müracaat işleminin ne kadar sürdüğü gözlemlenmiştir. Bu araştırmaya göre, hastanın daha önceden e-Hastane sistemine kayıtlı olmaması durumunda, müracaat görevlisi, hastanın bilgilerini sağlık karnesinden, hastanenin veritabanına e-Hastane sistemini kullanarak girmesi ve istenilen poliklinikten bir sıra numarası alması 33.09 saniye ve e-Hastane sistemine daha önceden kayıtlı bir hastanın(Hasta Tanıma Özelliği ile) müracaat süresi ise sadece 09.79 saniye sürmüştür. Bu süre, hastane yazılımına sahip olmayan bir hastanede, kullanıcıya (müracaat görevlisi) bağlı olarak 2-5 dakika arasında sürmektedir. Bayındır Hastanesi ve Başkent Üniversitesi Hastanesi Diş Hekimliği Polikliniği gibi yazılım kullanan hastanelerle bulunulan temaslarda, kayıt kabul görevlilerinin vermiş oldukları bilgiler doğrultusunda, aynı tarz bilgilerin alındığı bir müracaat işleminin en az 1.5’-2.0’ dakika sürdüğü öğrenilmiştir.

Yukarıda sonuçları açıklanan araştırmanın sonucu olarak söylememiz gereken, e-Hastane sistemi gibi bir sistemin, Türkiye’nin nüfus bakımından yoğun illerinden birinde kurulu olması durumunda, bir hastanın hastaneye attığı ilk adımı, müracaat işlemini, kullanılan JPA teknolojisi ve Hasta Tanıma Özelliği ile hızlandırması ve bunun da, çalışmanın temel amaçlarından biri olan hasta memnuniyetini arttıracaktır.

e-Hastane sisteminin varlığı, yazılım kullanmayan hastaneler için Bölüm 2.3’te bahsedilen Seviye 1’e bir adım, yazılım kullanan hastaneler için ise, daha önceden gelen hastaları veritabanında tutma özelliği (Hasta Tanıma Özelliği) ile, müracaat esnasında, sıra numarası almak için beklenen zamanı azaltması olacaktır. Böylelikle hastalar için bekleme zamanı, daha önceden hastaneye gelen bir hasta için büyük oranda azalmış olacak ve bu sistemin diğer modüllerinin analizi ve

kodlanması durumunda da, ileriye yönelik, hızlı çalışan, hasta beklentilerini karşılayan, veri kaybının az olarak yaşandığı, standartlara uygun bir sistem geliştirilmiş olacaktır.

e-Hastane sisteminin kazandırdığı bir diğer özellik ise, hastalardan alınan, önceden belirlenmiş, bilgilerin belirli standartlara göre olması ve bu yüzden de elde edilen bilgilerin daha sonradan kullanılabilirliğinin yüksek olmasıdır. Bu sayede, ileride kodlaması yapılacak olan Poliklinik, Laboratuvar vb. modüllerinde, doktor daha önceden girilmiş yüksek standartlı, kaliteli bilgilerle sadece tedavi değil, aynı zamanda, ileriye yönelik araştırma da yapabilir duruma gelebilecektir. Bu veriler aynı zamanda, Richard E. Scott'ın (2006) Sağlıkta e-Kayıt (e-Record in Health) adlı çalışmasında da bahsedildiği üzere, merkezi bir veritabanında uzun süre saklanabilir ve kullanıma her zaman açık olma özelliğine de sahiptirler. İlgili verilerin sadece yetkili kişiler tarafından görüntülenmesi, e-Hastane uygulaması dahilinde sağlanmış, böylelikle bilgilerin güvenlik sorunu da ortadan kalkmıştır.

## **5.2 Öneriler**

Bu çalışmada yapılan araştırmalar ve incelemeler sonunda varılan sonuçlardan birisi de Türkiye'deki, bütün sağlık kuruluşlarının teker teker farklı sistem ve yaklaşımlarla bilişim teknolojileri uygulamalarına geçmesini önlemek olmalıdır. Sağlık Bakanlığı her kuruluşun kendi yazılım ve donanımını seçmesini serbest bırakmalı ancak, Türkiye yakın bir gelecekte evrensel olacak, Sağlık Verileri Bilişim Standartlarını belirleyip, ilan etmelidir. Bunun sonucunda elde edilen veriler daha kaliteli ve güvenilir olurken hekimlerin hastalardan toplanan verileri yine hasta tedavilerinde kullanarak daha sağlıklı bilimsel çalışmalar yapabilir, devlet kuruluşları bu bilgiler ile daha planlı çalışabilir ve hastalık öncesi eğitimler verebilirler.

Yapılması mümkün olan bir diğer yenilik ise, e-Hastane sistemi ile mevcut hastane sistemlerinin kullandığı akıllı kart uygulamalarını birleştirmektir. Şimdilik sadece müracaat işleminde kullanılan bu kartlar üzerindeki yazılımlara e-Hastane sistemi gibi bir yazılım yüklenerek, kartların sadece müracaat değil, diğer modüllerde de

alıřabilir hale gelmesi saęlanabilir. Fakat bu, akıllı kartların ne kadar akıllı olduęu sorusunu gündeme getirir. Bunun bařlıca sebebi ise, akıllı kartların bilinen bařlıca dezavantajlarıdır. Bunları sıralamak gerekirse; akıllı kartların maliyetinin oldukça fazla olması, bu kartları okuyacak olan aygıtların oldukça pahalı olması ve kullanılan her yeni teknoloji için güncellenmesi gerektięi ve en önemlisi güvenlik konusunun nasıl saęlandıęıdır.

Mevcut e-Hastane sisteminin alt yapısı (Veritabanı, programlaması vb.) bakımıyla, ileriye yönelik adımlara ve teknolojilere açık bir sistemdir. Mevcut sisteme yapılabilecek olan eklemelerle e-Hastane sisteminde maksimum fayda saęlanabilir, hastaların beklentileri karşılanabilir. Örnek vermek gerekirse, bu sistemde yapılmıř olan analiz güncellenerek, belirlenmesini istenilen Saęlık Verileri Biliřim Standartlarına uygun bir řekilde, Bölüm 2.4'te açıklanan hastanenin dięer modülleri için de kodlama yapılması e-Hastane sistemini sadece bir modül için deęil, bir hastanede olabilecek bütün modüller için aynı özelliklere sahip olmasını saęlayacaktır. Böylece güncellenecek olan ilgili e-Hastane sistemi, önce bir poliklinikte, daha sonra ise belirlenecek olan bir pilot hastane de uygulamaya geçirilip getireceęi faydalardan yararlanılması, esas ama olan hasta beklentilerini karşılanmasında, ülke genelinde hizmet verebilmesinde, hızlı ve kolay erişilebilir olmasında, kurum yerine hasta ihtiyaçlarına odaklı olmasında, tedavilerin yanı sıra sebeplerinin de ortaya çıkmasında, bize ileri vade de büyük kazanç saęlayacaktır.

## 6 KAYNAKLAR LİSTESİ

- [1] T.C Sağlık Bakanlığı (2006), *I. Sağlık Bilişim Kongresi Sonuç Raporu*, 1-8.
- [2] Wager, K., Lee, F. Ve Glaser, J. (2005), *Managing Health Care Information Systems*(1), A.B.D : Jossey-Bass.
- [3] Türkiye II. Bilişim Şurası. (11.05.2004). *Sonuç Bildirgesi*.14.09.2007, <http://www.bilisimsurasi.org.tr/SonucBildirgesi/>.
- [4] T.C. Sağlık Bakanlığı Bilgi İşlem Daire Başkanlığı, *Türkiye Sağlık Bilgi Sistemi*, 30.01.2003.
- [5] Haux, R. (2006). Health Information Systems – past, present future, *International Journal of Medical Informatics* .75, 268—281.
- [6] Tan, J. (2005), *E-Health Care Information Systems*(1), A.B.D : Jossey - Bass.
- [7] Richard E. Scott (2006), e-Records in Health-Preserving our Future, *International Journal Of Medical Informatics*. 76, 427–431.
- [8] Milicevic, I. (2005). *Quality Of Online Health Information*, IPSI Conference.
- [9] Akdur, Recep, (2007), *Türkiye'nin Geleceği İçin Sağlık Sektöründe Yeni Arayışlar*, 14.09.2007, [www.recepakdur.com](http://www.recepakdur.com).
- [10] Şahin, Feyzullah, T.C Sağlık Bakanlığı *Sağlıkta Dönüşüm Programı ve Sağlık Eğitimi*, 2004.
- [11] Wikipedia (2007), *Java Programming Language*, 10.12.2007, [http://en.wikipedia.org/wiki/Java\\_\(programming\\_language\)](http://en.wikipedia.org/wiki/Java_(programming_language)).
- [12] Sun Microsystems Inc. (2007), *Java*, 14.11. 2007, <http://java.sun.com>.

- [13] Linskey, P. (2006), *JPA: The Goo, The Bad and the Ugly*, 08.10.2007, <http://bea.com/kodo>.
- [14] Keith, M. (2007), *Java Persistence API: Portability Do's and Don'ts*, 11.09.2007, <http://otn.oracle.com/jpa>.
- [15] Ermutlu, B., Karavuş M.,Çalı Ş., Gürbüz, Y. ve Akdaş A. (2000), *Ayaktan Hasta Bakımında Bilgi Teknolojisi Kullanımı*.487-503.
- [16] Enünlü, T. (2007), *Elektronik Hasta Kayıtları*, 01.11.2007, [www.turkmia.org/eski/file/331esk\\_musoglu\\_enunlu.ppt](http://www.turkmia.org/eski/file/331esk_musoglu_enunlu.ppt).