

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

(YÜKSEK LİSANS TEZİ)

**HEMATOLOJİK HASTALIKLARIN TEDAVİSİNDE
UZAKTAN HASTA TAKİBİ İÇİN ÇOKLU-ETMEN
SİSTEMİ GELİŞTİRİLMESİ**

Süleyman Kıvanç EKİCİ

Tez Danışmanı : Yrd. Doç. Dr. Özgür GÜMÜŞ

Bilgisayar Mühendisliği Anabilim Dalı

Bilim Dalı Kodu : 619.01.00

Sunuş Tarihi : 09.02.2015

Bornova-İZMİR

2015

Süleyman Kıvanç EKİCİ tarafından **YÜKSEK LİSANS** tezi olarak sunulan **“Hematolojik Hastalıkların Tedavisinde Uzaktan Hasta Takibi İçin Çoklu-Etmen Sistemi Geliştirilmesi”** başlıklı bu çalışma E.Ü. Lisansüstü Eğitim ve Öğretim Yönetmeliği ile E.Ü. Fen Bilimleri Enstitüsü Eğitim ve Öğretim Yönergesi’nin ilgili hükümleri uyarınca tarafımızdan değerlendirilerek savunmaya değer bulunmuş ve 09.02.2015 tarihinde yapılan tez savunma sınavında aday oybirliği/oyçokluğu ile başarılı bulunmuştur.

Jüri Üyeleri:

İmza

Jüri Başkanı : Yrd.Doç.Dr.Özgür GÜMÜŞ

.....

Raportör Üye : Yrd.Doç.Dr. Şebnem BORA

.....

Üye : Yrd.Doç.Dr Övünç ÖZTÜRK

.....

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

ETİK KURALLARA UYGUNLUK BEYANI

E.Ü. Lisansüstü Eğitim ve Öğretim Yönetmeliğinin ilgili hükümleri uyarınca Yüksek Lisans Tezi olarak sunduğum “Hematolojik Hastalıkların Tedavisinde Uzaktan Hasta Takibi İçin Çoklu-Etmen Sistemi Geliştirilmesi” başlıklı bu tezin kendi çalışmam olduğunu, sunduğum tüm sonuç, doküman, bilgi ve belgeleri bizzat ve bu tez çalışması kapsamında elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara atıf yaptığımı ve bunları kaynaklar listesinde usulüne uygun olarak verdiğimi, tez çalışması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını, bu tezin herhangi bir bölümünü bu üniversite veya diğer bir üniversitede başka bir tez çalışması içinde sunmadığımı, bu tezin planlanmasından yazımına kadar bütün safhalarda bilimsel etik kurallarına uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul edeceğimi beyan ederim.

09 / 02 / 2015

İmzası

Adı-Soyadı

ÖZET

**HEMATOLOJİK HASTALIKLARIN TEDAVİSİNDE UZAKTAN
HASTA TAKİBİ İÇİN ÇOKLU-ETMEN SİSTEMİ
GELİŞTİRİLMESİ**

EKİCİ, Süleyman Kıvanç

Yüksek Lisans Tezi, Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Yrd. Doç. Dr. Özgür Gümüş

2015, 103 sayfa

Sağlık bilişimi sistemleri çok çeşitli ve yüksek karmaşıklığın olduğu sistemlerdir ve bu alanda yapılan çalışmaların temel hedefi merkezi bir hasta kayıt sistemi oluşturmak ve oluşan kayıtları karar destek sistemlerinde kullanabilmektir. Ancak sağlık sürekli değişen ve gelişen bir alan olduğu için bu hedefe henüz ulaşamamıştır. 2015 itibariyle kurulu sistemler hastanede üretilen veriyi genel olarak kayıt etmekte ve kısmen merkezi sistemlere aktarmaktadır, ancak veri paylaşımı açısından sağlık kuruluşları birbirinin dilinden anlamayan iki ayrı devlet gibidir.

Bir başka önemli konu ise hastaların hastane dışında geçirdiği süre içinde hiçbir veri toplanmamaktadır. Bunun yerine birkaç ayda bir yapılan görüşmelerde hastanın 10 dakikalık sürede anlattığı yetersiz bilgiler kayıt edilmektedir. Hematolojik hastalıklarda, hastane dışı zamanlarda oluşan veriler hastanın yaşam kalitesini ve tedavinin yönünü etkileyecek kadar önemlidir. Bazı hematoloji hasta gruplarında ise sürekli tutulması gereken kayıtlar vardır ve olağandışlıkların doktor tarafından takip edilmesi gereklidir.

Bu tez çalışmasında bahsi geçen sorunlara Çoklu Etmek Sistemleri ile çözüm getirmek amaçlanmıştır. MASHemato adı verilen ÇES ile, mikro düzeyde her bir hastanın, doktorun ve sağlık biriminin ihtiyaçlarını karşılayan etmenlerin hareketleri ve kendi aralarında otomatik şekilde organize olmaları yoluyla; makro düzeyde ise veri paylaşımı, uzaktan takip ve hasta bilgilendirmeleri ile halk sağlığı konularında teknik çözüm üretilmiş ve ilerleme elde edilmiş olacaktır.

Bu çalışma kapsamında oluşturulan sistem, hasta, hekim, sağlık kuruluşu, ilk yardım ekibi, bakanlık ve Sosyal Güvenlik Kurumu gibi sağlıkla ilgili tüm birimler için alt sistemler içermektedir. Gereksinim analizleri ve mimari tasarım Tropos metodolojisi kullanılarak gerçekleştirilmiştir.

Sistemin veri analizi yönünden geniş imkânlarla sahip olması için ve yeni kod yazmadan sistemin geliştirilebilmesi amacıyla veriler OWL tabanlı ontolojiler ile kayıt edilmektedir.

Çalışma kapsamında ayrıca hasta takip sistemlerinde ve genel olarak sağlık bilişimi sistemlerinde kullanılması gereken standartların bir referans listesi oluşturulmuştur.

Anahtar Kelimeler: Çoklu Etmen Sistemleri, koordinasyon, Tropos metodolojisi, etmen tabanlı yazılım geliştirme, uzaktan hasta takibi, karar destek sistemleri, hematoloji

ABSTRACT**BUILDING A MULTIAGENT SYSTEM FOR REMOTE PATIENT SURVEILLANCE IN HEMATOLOGIC DISORDERS TREATMENT**

EKİCİ, Süleyman Kıvanç

MSc in Computer Science Eng.

Supervisor: Yrd. Doç. Dr. Özgür Gümüş

2015, 103 pages

Health information systems are very diverse and highly complex systems which aim to create a central health record system and use the information stored in decision support systems. However the goal is far from reach because of the changing and developing nature of the domain. As of 2015, the deployed systems store the data created at hospitals and send a relatively small part of the data to the central systems, yet in terms of data sharing the two healthcare institutions are like two different states that does not understand each others language.

Another issue about patient data is that, during the time outside the hospital, no data is collected and instead inadequate data from patient interviews, which lasts only ten minutes and held only every few months, is used and stored. In hematological diseases the data, created at times outside the hospital, can change the direction of the treatment and is critical to the patients quality of life. In some hematological disease groups there are records to be taken daily and any anomalies should be notified to the doctor.

MASHemato Project aims to suggest a solution to the addressed issues with Multi Agent Systems technology. The technical solution to data sharing, remote patient surveillance, patient instructing and public health improvement in macro level is achieved through micro level activities and coordinations of agents created patients, doctors, healthcare units and ministries.

The main system proposed in this work contains subsystems designed for patients, doctors, healthcare units, rescue ambulance units and ministries. The requirements analysis and architectural design is completed using Tropos methodology.

The data is stored as OWL ontologies in order to widen the capabilities of data analysis and to enhance the system without further coding.

A reference list to the standards, which needs to be complied by patient surveillance systems, is composed.

Keywords: Multi-Agent Systems, coordination, tropos methodology, agent oriented software engineering, remote patient surveillance, decision support systems, hematology

TEŞEKKÜR

Bu projeyi pek çok insanın desteğini ve görüşlerini almadan gerçekleştirmem mümkün olamazdı. Başta proje süresince sayısız güncellemelerimi okuyan, yorumlayan ve her konuda yakın desteğini esirgemeyen değerli danışmanım sayın Yrd. Doç. Dr. Özgür Gümüş’e çok teşekkür ederim.

Projenin fikrinin gelişimi ve tıbbi konulardaki katkıları nedeniyle Prof. Dr. Kaan Kavaklı’ya, Op. Dr. Tuncay Ekice’ye teşekkür ederim. Yazılım konusunda görüş ve katkıları nedeniyle arkadaşlarıma ve şirket çalışanlarıma teşekkür ederim.

Uzun çalışma sürecinde benimle birlikte olan sevgili eşime ve aileme teşekkürler ederim.

İÇİNDEKİLER

Sayfa

ÖZET	VII
ABSTRACT	IX
TEŞEKKÜR	XI
ŞEKİLLER DİZİNİ	XIX
ÇİZELGELER DİZİNİ	XXIII
SİMGELER VE KISALTMALAR DİZİNİ	XXV
1. GİRİŞ	1
1.1 Problemin Tanımı	2
1.2 Motivasyon	3
2. KAVRAMLAR	5
2.1 Etmen	5
2.2 BDI Etmenler	6
2.3 Çoklu Etmen Sistemleri	7
2.4 Çoklu Etmen Sistemleri İçin Kalite Ölçütleri	7
2.5 Etmen Platformu	8
2.6 Etmen Tabanlı Yazılım Geliştirme	9
2.7 Organizasyonel Desenler	12
2.7.1 Beşli yapı	12
2.7.2 Ortak girişim	13
2.7.3 Piramit stili	14

İÇİNDEKİLER (devam)

Sayfa

2.7.4	Değerler zinciri	15
2.7.5	Matris	15
2.7.6	Açık arttırma stili	15
2.7.7	Emsallere uygunluk stili.....	16
2.7.8	Hiyerarşik anlaşıma stili.....	16
2.8	TROPOS Metodolojisi.....	17
2.8.1	Neden TROPOS.....	17
2.8.2	Tropos kavramları.....	18
2.8.3	Modelleme işlemleri	19
2.8.4	Tropos adımları.....	21
2.9	Sağlık Bilişimi.....	24
2.9.1	Sağlık bilişiminde standart geliştiren kuruluşlar.....	25
2.9.2	Geliştirilen standartlar.....	25
2.9.3	HITSP Klinik Doküman ve Mesaj Terminolojisi bileşeni.....	26
2.9.4	ASC X12.....	26
2.9.5	DICOM	26
2.9.6	HL7 V3	27
2.9.7	NCPDP.....	29
2.9.8	Sağlık bilişiminde standart geliştirilen alanlar.....	30
2.9.9	Tanımlayıcılar	30
2.9.10	İçerik ve yapı standartları.....	31
2.9.11	Mesajlaşma standartları.....	31
2.9.12	Tıbbi terminolojiler ile ilgili standartlar.....	31
2.9.13	Sağlık bilişiminde standartların kullanıldığı uygulamalar	31
2.9.14	Veri gizliliği ve veri güvenliği	32
2.9.15	Veri kalitesi göstergeleri	32
2.9.16	SağlıkNet uygulaması	33
2.9.17	Sağlık bilişimi değerlendirme	35
2.10	Hematoloji.....	35
2.10.1	Hematoloji hastasının muayenesi.....	35
2.10.2	Hematoloji hastasının tıbbi geçmişi.....	36
2.10.3	Genel belirtiler	36

İÇİNDEKİLER (devam)

Sayfa

2.10.4	Spesifik belirtiler	37
2.10.5	İlaçlar ve kimyasallar.....	39
2.11	Servis Odaklı Mimari	40
2.12	Web Servisleri	40
2.13	Ontoloji	41
3.	TEKNİK ANALİZ.....	43
3.1	Sağlık Temalı Çoklu-Etmen-Sistemi Çalışmaları.....	43
3.1.1	Aingeru	44
3.1.2	Cervantes et al.....	45
3.1.3	Koutkias et al.	46
3.1.4	SAPHIRE.....	47
3.1.5	HeCaSe2	48
3.2	Çoklu Etmen Platformları Karşılaştırmalı Analizi.....	49
3.3	Etmen Platformu Seçimi	57
3.4	Ontoloji Altyapısı.....	57
3.4.1	Ontoloji dilleri	57
3.4.2	Ontoloji veritabanı	58
3.4.3	JENA.....	59
3.5	Web Servis Altyapısı	59
3.5.1	JAX WS	59
3.5.2	RESTful web servisleri.....	60
3.5.3	WCF servisleri	60
3.6	Mobil Uygulama Altyapıları.....	61
3.6.1	Android.....	61
3.6.2	iOS	62
3.6.3	Windows mobile.....	62

İÇİNDEKİLER (devam)

Sayfa

3.7	Sağlık Temalı Mobil Uygulamalar	63
3.8	Güvenlik.....	63
3.8.1	VPN.....	65
3.9	Genel Senaryo.....	65
3.9.1	Hasta senaryosu.....	66
3.9.2	Hasta yakını ve hasta bakıcısı senaryosu	67
3.9.3	Acil yardım servisi senaryosu	67
3.9.4	Uzman sağlık görevlisi senaryosu.....	67
3.9.5	Yardımcı sağlık görevlisi senaryosu	68
3.9.6	Sağlık kuruluşu departmanı senaryosu	68
3.9.7	Sağlık kuruluşu senaryosu	68
3.9.8	Rapor isteyen kuruluş senaryosu.....	69
4.	TASARIM VE MİMARİ.....	70
4.1	Gereksinimler	70
4.1.1	Alan gereksinimleri.....	70
4.1.2	Teknik gereksinimler	71
4.1.3	Erken gereksinim analizi.....	71
4.1.4	Geç gereksinim analizi.....	75
4.2	Sistem Mimarisi.....	79
4.2.1	Mimari tasarım.....	80
4.2.2	Detaylı tasarım	86
5.	GERÇEKLEŞTİRİM VE DEĞERLENDİRME	91
5.1	Gerçekleştirim	91
5.1.1	Etmenler	91
5.1.2	Web servisleri	92
5.1.3	Kullanıcı görsel arayüzleri	92
5.2	Değerlendirme	95

İÇİNDEKİLER (devam)

Sayfa

6. ÖZET VE SONUÇ.....97

KAYNAKLAR DİZİNİ99

EKLER

EK 1 ERKEN GEREKSİNİM ANALİZİ ŞEMALARI

EK 2 GEÇ GEREKSİNİM ANALİZİ ŞEMALARI.....

EK 3 MİMARİ TASARIM ŞEMALARI

EK 4 ETMEN GERÇEKLEŞTİRİM KODLARI.....

EK 5 RESTFUL WEB SERVİS GERÇEKLEŞTİRİM KODLARI.....

ŞEKİLLER DİZİNİ

<u>Şekil</u>	<u>Sayfa</u>
2-1 EŞLEŞTİRME DESENİ.....	10
2-2 GÖZLEMCİ DESENİ.....	11
2-3 ANLAŞMA DESENİ	11
2-4 ARACI DESENİ	12
2-5 BEŞLİ YAPI DESENİ	13
2-6 ORTAK GİRİŞİM DESENİ	14
2-7 PİRAMİT STİLİ DESENİ	14
2-8 DEĞERLER ZİNCİRİ DESENİ	15
2-9 AÇIK ARTTIRMA STİLİ.....	16
2-10 AKTÖRLERİN GÖSTERİMİ.....	18
2-11 HEDEFLERİN GÖSTERİMİ	18
2-12 PLANLARIN GÖSTERİMİ	19
2-13 KAYNAKLARIN GÖSTERİMİ.....	19
4-1 ALAN ORTAKLARI	72
4-2 TELETİP ALAN ORTAKLARI SADELEŞTİRİLMİŞ ANALİZ	73
4-3 HASTA ERKEN GEREKSİNİM ANALİZİ	74
4-4 UZMAN SAĞLIK GÖREVLİSİ ERKEN GEREKSİNİM ANALİZİ	75
4-5 MASHEMATO GEÇ GEREKSİNİM ANALİZİ	76
4-6 HASTA MASHEMATO GEÇ GEREKSİNİM ANALİZİ.....	77
4-7 MASHEMATO UZMAN SAĞLIK GÖREVLİSİ GEÇ GEREKSİNİM ANALİZİ	78
4-8 MASHEMATO SAĞLIK KURULUŞU GEÇ GEREKSİNİM ANALİZİ	78
4-9 MASHEMATO ACİL YARDIM EKİBİ GEÇ GEREKSİNİM ANALİZİ.....	79
4-10 HASTA MASHEMATO MİMARİ TASARIM	80
4-11 MİMARİ TASARIM: UZMAN SAĞLIK GÖREVLİSİ.....	81
4-12 MİMARİ TASARIM: SAĞLIK KURULUŞU	82
4-13 MİMARİ TASARIM: RAPOR İSTEMCİSİ KURUM	82
4-14 MİMARİ TASARIM: ACİL YARDIM EKİBİ SİSTEMİ	83
4-15 MİMARİ TASARIM: HASTA SİSTEMİ	84
4-16 HASTA SİSTEMİ ASİSTANI DETAYLI TASARIM.....	87
4-17 HASTA SİSTEMİ BİLGİ TOPLAMA ARACI DETAYLI TASARIM	87
4-18 HASTA SİSTEMİ HASTA TAKİP ARACI DETAYLI TASARIM	88
4-19 ACİL YARDIM EKİBİ SİSTEMİ ASİSTAN DETAYLI TASARIM	89
4-20 HASTA GÖZLEM İŞLEMİ SIRA DİYAGRAMI	90

ŞEKİLLER DİZİNİ (devam)

Şekil

Sayfa

5-1 KARŞILAMA GİRİŞ VE ANA EKRANLAR.....	93
5-2 BİLDİRİMLER, BİLGİLERİN VE İLAÇ KULLANIM İŞLEMLERİ EKRANI.....	93
5-3 UYGULAMA WEB ARAYÜZLERİ GİRİŞ.....	94
5-4 WEB ARAYÜZLERİ ANA SAYFA	94
6-1 ANA ŞEMA.....	3
6-2 UZMAN SAĞLIK GÖREVLİSİ	4
6-3 SAĞLIK KURULUŞU UZMAN SAĞLIK GÖREVLİSİ	4
6-4 SAĞLIK KURULUŞU VE DEPARTMAN.....	5
6-5 SAĞLIK KURULUŞU	6
6-6 SAĞLIK KURULUŞU DEPARTMANI-YARDIMCI SAĞLIK GÖREVLİSİ	7
6-7 SAĞLIK KURULUŞU DEPARTMANI-UZMAN SAĞLIK GÖREVLİSİ	7
6-8 SAĞLIK KURULUŞU DEPARTMANI.....	8
6-9 RAPOR İSTEMCİSİ-SAĞLIK KURULUŞU	8
6-10 HASTA	9
6-11 HASTA-SAĞLIK KURULUŞU	10
6-12 HASTA YAKINI BAKICI	10
6-13 ACİL YARDIM EKİBİ.....	11
6-14 MASHEMATO GEÇ GEREKSİNİM ANALİZİ.....	12
6-15 MASHEMATO SAĞLIK KURULUŞU DEPARTMANI	13
6-16 MASHEMATO SAĞLIK KURULUŞU	13
6-17 MASHEMATO RAPOR İSTEMCİSİ KURUM	14
6-18 MASHEMATO HASTA YAKINI BAKICI.....	14
6-19 MASHEMATO ACİL YARDIM EKİBİ	15
6-20 MASHEMATO HASTA.....	15
6-21 MASHEMATO ACİL YARDIM EKİBİ	16
6-22 MASHEMATO YARDIMCI SAĞLIK GÖREVLİSİ	16
6-23 MASHEMATO UZMAN SAĞLIK GÖREVLİSİ.....	17
6-24 MİMARİ TASARIM ACİL YARDIM EKİBİ	18
6-25 MİMARİ TASARIM YARDIMCI SAĞLIK GÖREVLİSİ.....	18
6-26 MİMARİ TASARIM YARDIMCI SAĞLIK GÖREVLİSİ SİSTEMİ	19
6-27 MİMARİ TASARIM YÖNETİM SİSTEMİ.....	19
6-28 VERİ YÖNETİMİ SİSTEMİ	20

ŞEKİLLER DİZİNİ (devam)

Şekil

Sayfa

6-29 UZMAN SAĞLIK GÖREVLİSİ SİSTEMİ	20
6-30 MİMARİ TASARIM UZMAN SAĞLIK GÖREVLİSİ.....	21
6-31 SİSTEM GİRİŞ YETKİLENDİRME SİSTEMİ	22
6-32 SAĞLIK KURULUŞU SİSTEMİ.....	23
6-33 SAĞLIK KURULUŞU DEPARTMANI SİSTEMİ.....	24
6-34 MİMARİ TASARIM SAĞLIK KURULUŞU DEPARTMANI	25
6-35 MİMARİ TASARIM SAĞLIK KURULUŞU	26
6-36 MİMARİ TASARIM RAPOR İSTEMCİSİ KURUM	27
6-37 RAPOR HAZIRLAMA SİSTEMİ	28
6-38 HASTA YAKINI BAKICI SİSTEMİ.....	29
6-39 MİMARİ TASARIM HASTA YAKINI BAKICI.....	29
6-40 HASTA SİSTEMİ	30
6-41 HASTA SİSTEMİ 2.....	31
6-42 MİMARİ TASARIM HASTA.....	32
6-43 EŞLEŞTİRME VE ARAMA SİSTEMİ.....	33
6-44 ACİL YARDIM EKİBİ SİSTEMİ.....	34

ÇİZELGELER DİZİNİ

<u>Çizelge</u>	<u>Sayfa</u>
3-1 ETMEN PLATFORMLARI VE İŞLETİM SİSTEMLERİ	52
3-2 ETMEN PLATFORMLARI VE METODOLOJİLER, GÜNCEL GELİŞTİRME ARAÇLARI	53
3-3 ETMEN PLATFORMLARI, TEKNİK DESTEK VE GELİŞTİRME	54
3-4 ETMEN PLATFORMLARI VE ARGE GRUBU ÇALIŞMALARI	55
3-5 ETMEN PLATFORMLARI VE KULLANIM LİSANSLARI	56

SİMGELER VE KISALTMALAR DİZİNİ

<u>Kısaltmalar</u>	<u>Açıklama</u>
AJAX	Asynchronous Javascript and XML
ASTM	American Society for Testing and Materials
BAN	Body Area Network
BDI	Belief Desire Intention
BT	Bilgi Teknolojileri
CDA	Clinical Document Architecture
CEN	European Committee for Standardization
CPRI	The Computer Based Patient Record Institute
ÇES	Çoklu Etmen Sistemi
DICOM	Digital Imaging and Communications in Medicine
DOM	Document Object Model
DoS	Denial of Service
ECRI	Emergency Care Research Institute
EHR	Electronic Health Records
EJB	Enterprise Java Beans
FIPA	The Foundation of Intelligent Physical Agents

SİMGELER VE KISALTMALAR DİZİNİ (devam)

<u>Kısaltmalar</u>	<u>Açıklama</u>
GUI	Graphical User Interface
HEDIS	Healthcare Effectiveness Data and Information Set
HITSP	Healthcare Information Technology Standards Panel
HL7	Health Level Seven
HL7 RIM	HL7 Belge Yönetimi Standartı
HTTP	Hyper-Text Transfer Protocol
IETF	Internet Engineering Task Force
IIS	Internet Information Server
ISO	International Organization for Standardization
JAX WS	Java API for XML Web Services
JMS	Java Message Service
JPA	Java Persistence API
LOINC	Logical Observation Identifier Names and Code
MAF	Multi Agent Facility
MCC	Medical Contact Center
MIME	Multipurpose Internet Mail Extensions
NCPDP	National Council for Prescription Drug Programs

SİMGELER VE KISALTMALAR DİZİNİ (devam)

<u>Kısaltmalar</u>	<u>Açıklama</u>
NDC	National Drug Code
OECD	The Organisation for Economic Cooperation & Development
OMG	Object Management Group
OWL	Web Ontology Language
PACS	Picture Archiving and Communication System
PDA	Personal Digital Assistant
RDF	Resource Definition Framework
RIM	Records Information Management
RMI	Remote Method Invocation
SKRS	Sağlık Kodlama Referans Sunucusu
SOA	Service Oriented Architecture
SNOMED	The Systematized Nomenclature of Human and Veterinary Medicine
SOAP	Simple Object Access Protocol
UDDI	Universal Description Discovery and Integration
UDP	User Datagram Protocol
UML	Unified Modelling Language

SİMGELER VE KISALTMALAR DİZİNİ (devam)

<u>Kısaltmalar</u>	<u>Açıklama</u>
UMDNS	Universal Medical Device Nomenclature System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USVS	Ulusal Sağlık Veri Sözlüğü
W3C	World Wide Web Consortium
WCF	Windows Communication Foundation
WHO	World Health Organization
WSDL	Web Service Description Language
WWW	World Wide Web
XML	eXtensible Markup Language

1. GİRİŞ

Bilgisayarlar, gelişen donanım teknolojileri sayesinde her geçen gün küçülmeye ve güçlenmeye devam etmektedirler. Bu gelişmeler mobil cihazları da küçültmüş ve yaygınlaştırmıştır, öyle ki artık cebe sığabilen akıllı telefonlar neredeyse her evde bulunmaktadır. Bunun yanı sıra Türkiye’de Milli Eğitim Bakanlığınca yürütülen Fatih Projesi ile de her öğrenciye bir tablet bilgisayar verilmekte ve bu cihazların kullanımı giderek artmaktadır. Bu kadar yaygın kullanılan ve giderek güçlenen bilgisayarlar biz bilişimcilere de çok geniş bir yelpazede mobil uygulamalar yapma imkânı sağlamaktadır.

Mobil uygulamalar her alanda hızla çoğalmaktadır ve bu alanlara sağlık da dâhildir. Kullanılan teknolojilerin altyapıları dünyanın en güçlü teknoloji firmaları tarafından üretilmekte ve desteklenmektedir. Sağlık alanında tele-tıp uygulamaları, gerek son kullanıcı gerekse araştırma geliştirme tarafından oldukça ilgi görmektedir ve bu ilgi zamanla artış göstermektedir.

Mobil cihazlar üzerinde geliştirilen uygulamalar kablosuz ağ, bluetooth gibi teknolojiler ile birçok cihaza bağlanmakta ve bunlar üzerinde işlem yapmaya olanak vermektedir. Hali hazırda yapılan bilimsel çalışmalarda yer alan BAN (Body Area Network), vücut hakkında bilgi toplayan sensörlerin bir bilgi toplama ve değerlendirme cihazına kablolu veya kablosuz olarak bağlanması ile oluşturulan bir ağıdır.

Tıbbi cihaz üreticileri de giderek artan bir ilgi ile taşınabilir cihaz projeleri geliştirmektedir. Bunlar arasında kalple ilgili ritim bilgilerini toplayarak kablosuz ağ üzerinden iletebilen Holter cihazı, bebeklerin nabız ve vücut sıcaklığını ölçebilen ve bu bilgileri kablosuz ağ üzerinden paylaşabilen cihazlar sayılabilir. Yeni üretilen cihazlar giderek artan biçimde iletişim teknolojilerini desteklemektedir.

Tüm bu bilgiler ışığında görülmektedir ki mobil cihazlar hayatımızın diğer alanlarında olduğu gibi sağlık alanında da çok etkin bir konuma gelecektir. Doktorlar hastalarının bilgilerini daha güvenilir yöntemlerle alabilecekler, bizler vücudumuzda oluşan anormal değişikliklerden çok geç olmadan haberdar olabileceğiz. Ayrıca yakınlarımızın sağlık durumları ve acil durumlarını da bu mobil cihazlar aracılığı ile yakından takip edebileceğiz.

1.1 Problemin Tanımı

Sağlık bilişimi alanında yapılan çalışmaların temel hedefi merkezi bir hasta kayıt sistemi oluşturmak ve oluşan kayıtları karar destek sistemlerinde kullanabilmektir. Bununla beraber sağlık alanındaki değişim ve gelişimin yarattığı zorluklar nedeniyle bu hedefe henüz ulaşılabilmiş değildir. Bugün itibarıyla geliştirilmiş olan sistemler, sağlık kuruluşu bünyesinde üretilen veriyi kayıt altına almakta ve kısmen merkezi sisteme aktarmaktadır. Ancak veri paylaşımı açısından iki farklı sağlık kuruluşu birbirinin dilinden anlamayan iki ayrı devlet gibidir. Bu durum aynı hasta iki farklı sağlık kuruluşunda tedavi olduğu takdirde yapılan işlemlerin bir diğeri tarafından görülememesine yol açmaktadır.

Bir diğerk problem ise hastaların hastane dışında geçirdikleri zamana ait verilerin toplanmasında yaşanmaktadır. Örneğin kronik bir hasta iki kontrol arasında geçen üç aylık bir dönemde önemli sorunlar yaşamış, bir başka hastanede tedavi görmüş veya deri döküntüsü, kızarıklık gibi sorunlarla karşılaşmış olabilir. Bu tür bilgiler genellikle telefon ile doktora anlatılsa dahi kayıt altına alınmamaktadır. Bu veriler hastanın belli bir zaman sonraki kontrol sırasında anlatacağı yetersiz bilgilere dayanarak kayıt edilmeye çalışılmaktadır.

Hematolojik hastalıklarda (kan hastalıkları) bu durum hastanın yaşam kalitesini ve hatta tedavinin yönünü belirleyebilecek kadar önemli bilgilerin kaybolmasına yol açmaktadır. Bazı hematolojik hastalık gruplarında ise sürekli tutulması gereken kayıtlar vardır ve bunlar üzerindeki değişikliklerin doktorlar tarafından anında takip edilmesi gerekebilir. Ek olarak oluşan bu verilerin de hastane sisteminde bulunan hastaya ait dosyaya kayıt edilebilmesi gereklidir.

Bu problemler dışında, tedavide yardımcı olabilecek muhtemel eksiklikler arasında hastanın doktora bir değişikliği haber verme ve bilgi isteme gereksinimi, doktorun hastalarına soru sorma gereksinimi ve bu sorulara verilecek cevapların hızlı bir şekilde değerlendirmeye hazır olması gereksinimi sayılabilir. Ayrıca verilecek cevaplara göre bir uyarı bildirimi yapılması ve farklı bir disiplinden doktorlara bu bilgilerin yönlendirilmesi tedaviyi daha kaliteli bir hale getirebilir.

1.2 Motivasyon

Hematolojik hastalıklar başta olmak üzere karşı karşıya kaldığımız birçok hastalık oldukça karmaşık sorunlar olmakla beraber evdeki yaşamımızı ve genel olarak hayat tarzımızı da etkilemektedir. Hastalıkların gidişatı ve belirtileri bir günde bile değişebilmektedir. Bu durumda doktorlar hastalarına birkaç ayda bir yapılan sadece onbeş dakika süren ziyaretler ile yeteri kadar faydalı olması mümkün olamamaktadır. Sonuç olarak hastanın uzaktan ve sürekli takibi, tedavi için asgari şartları sağlamak adına çok gereklidir.

Teknolojinin son yıllarda geldiği nokta sayesinde mobil cihazlar ve taşınabilir tıbbi ölçüm cihazlarının erişilebilirliği ileri düzeydedir. Akıllı telefonlar işlemci gücü ve işletim sistemi olarak birçok uygulamanın önünü açmaktadır. Ayrıca kullanım yaygınlığı açısından bakıldığında Nielsen araştırma şirketinin Şubat 2013 raporuna göre Türkiye’de akıllı telefon kullanım oranı % 19’a yani her beş kişiden birine ulaşmıştır. Buradan hasta takibi için kullanılacak en uygun altyapının mobil cihazlar olduğu görülmektedir.

Etmen teknolojileri giderek olgunlaşmakta ve sağlık alanında bilimsel çalışmalar ile uygulanmaktadır. Bu çalışma kapsamında da anlatılan projelerden birçoğu yerel ve uluslararası kuruluşlardan önemli maddi kaynak sağlamışlardır. Hasta takibi konusunda dikkat edilmesi gereken bir konu da doktorların bu işlemi nasıl yürüteceğidir. OECD 2011 verilerine göre Türkiye’de her 1000 hastaya sadece 1,6 doktor düşmektedir. Bu veriler göstermektedir ki doktorların hastalarını sürekli takip etmesi için gerekli zamanı bulunmamaktadır. Bu nedenle, bu çalışma kapsamında takip işlemini doktorlar adına otonom bir biçimde yürütecek olan etmen sistemleri kullanılmaktadır.

Sağlık alanı her yönü ile çok kapsamlı, karmaşık ve gereksinimleri sürekli değişim geçiren bir alandır. Sağlık bilişimi sektöründe üretilmesi gereken sistemler hangi büyüklükte olursa olsun bir tek bilişim kuruluşunun karşılayamayacağı kadar geniştir. Bu durum her biri farklı bir alanda uzmanlaşmış birçok sağlık bilişimi firmasının ortaya çıkmasına neden olmuştur. Beraberce işbirliği halinde çalışması gereken çok fazla kuruluşun ürettiği çok fazla sistem olması ise entegrasyon sorunlarını ortaya çıkarmaktadır. Bu durumda sağlık bilişimi alanında standartların öğrenilerek benimsenmesi gerektiği ve her projede uygulanması gerektiği sonucu çıkarılmaktadır.

Genel olarak tüm sađlık biliřimi sistemleri, özel olarak ta uzaktan hasta takibi sistemlerinin projelerine bakıldığında; geleneksel yapıdaki projelerde gereksinim tabanlı bir metodoloji kullanılmadığı, çoklu etmen sistemi tabanlı çalışmalarda ise ya hiçbir metodoloji izlenmediğı ya da gereksinimleri esas alan metodolojiler kullanılmadığı görölmektedir. Oysaki yukarıda bahsedildiğı gibi sađlık alanı sürekli gelişen ve gereksinimleri değıřen bir alan olmasından dolayı, gereksinim tabanlı bir metodoloji kullanılması sürdürülebilirlik açısından önemli görünmektedir.

Gereksinim yönelimli metodoloji kullanılarak geliştirilecek çoklu etmen sisteminin altyapı bakımından genişletilebilir bir çerçeve olarak tasarlanması hedeflenmektedir. Böylece benzer çalışmalarda tekrar kullanılabilecek ve bu çalışmada bahsettiğimiz gerekçelere dayanan bir çoklu etmen sistemi çerçevesi oluşturulmuş olacaktır.

Ayrıca bu çalışma kapsamında, hasta takibi sistemlerinde kullanılması gerekebilecek standartlar da belirlenerek daha sonraki benzer çalışmalarda kullanılabilecek bir referans listesi oluşturmak hedeflenmiştir.

2. KAVRAMLAR

2.1 Etmen

Yazılım etmenleri, karmaşık ve farklı yapılar içeren dağıtık sistemler geliştirmek amacıyla tasarlanmış yenilikçi bir teknolojidir. Etmen kavramı üzerinde uzlaşma olan bir tanım bulunmamakla beraber bazı tanımlar şöyledir:

“Etmen, bulunduğu ortamı sensörleri aracılığı ile algılayabilen ve bir takım organları ile bu ortam üzerinde etki yaratabilen her şeydir.” (Russel and Norvig 2009)

“Otonom etmenler, karmaşık ve dinamik bir ortamda bulunan, bu ortamı otonom bir şekilde algılayarak üzerinde etki yapabilen ve bu yolla tasarlandıkları bir takım hedefleri ve görevleri yerine getirebilen yazılım parçalarıdır.” (Maes 1993)

Bu çalışma kapsamında uygulandığı anlamı ile tanımlamak gerekirse etmen, hedef güdümlü, başka birey veya bireyler adına işlemler yapan yazılım parçalarıdır. Etmenler ve etmen sistemleri kendi içinde hâkim olan, alan hakkında bilgi içeren ve hedefleri ile görevlerini yerine getirmek için belirli bir dereceye kadar bağımsız hareket edebilen yazılımlardır. Bu sistemler sürekli değişen ve istikrarlı olmayan çevrelerde çalışmak üzere tasarlanmıştır.

Etmenlerin bazı özellikleri aşağıdaki gibi tanımlanmaktadır;

Otonomi: Etmenler, kendi adlarına veya başkaları adına yaptıkları işlemlerde belirli bir dereceye kadar bağımsız hareket edebilirler. İçsel durumları ve yapacakları hareketler konusunda kontrol sahibidirler.

Proaktiflik: Etmenler sadece çevresel değişkenliklere tepki vererek çalışmazlar, hedefleri doğrultusunda karar da vererek inisiyatif kullanabilirler.

Reaktiflik: Etmenler çevrelerinde oluşan değişikliklere tepki verebilir, bu değişiklikleri değerlendirebilir ve zamanla adapte olabilirler. Örneğin işe gitmek için evden çıkarak otobüse binmeniz gerekiyor ve işe ulaşıyorsunuz. Eğer belli bir süre otobüs gelmez ise yapılacak olan otobüsü beklemekten vazgeçerek bir taksi

çağırmaktır. Böylece çevresel değişikliklere göre işletilen planın beklenen sonucu üretmesinin mümkün olup olmayacağını hesaplayarak karar verir.

İletişim ve İşbirliği: Etmenler hedeflerine ulaşmak için diğer etmenlerle ve bireylerle iletişime geçebilir ve kendi hedeflerini gerçekleştirme için işbirliği yapabilirler. İşbirliği, bilgi isteme, talimat gönderme veya talimat alma şeklinde olabilir.

Müzakere: Etmenler diğer etmenler ile belirli bir derecede işbirliği sağlamak için organize görüşmeler yapabilirler.

Öğrenme: Etmenler çevreleri ile etkileşim içinde olarak zamanla performanslarını geliştirebilirler.

2.2 BDI Etmenler

Kanı İstek Niyet (Belief Desire Intention) mimarisi temel olarak insanların rasyonel karar verme biçimini örnek alır. Bu kararlar güdülen istek ve amaçlara göre ve karar anında çevresi ile kendisi hakkındaki bilgilerine dayanarak oluşturulur. Bu anlayış bilgisayar sistemlerine şu şekilde aktarılmıştır;

- i. Kanılar (Beliefs): Etmenin sahip olduğu güncel bilgiler
- ii. İstekler (Desires): Etmenin takip ettiği hedefleri (Goals)
- iii. Niyetler (Intentions): Etmenin güncel hareketlerini yöneten ve o anda aktif olan hedef.

Etmen bir karar alma (Deliberation) süreci işleterek hangi hedefi seçerek işleteceğini belirler. Bir hedef belirlendikten sonra o hedef artık niyet olur. Bu niyetin nasıl gerçekleşeceği ise Araç-Amaç Analizi (Means-Ends Analysis) denilen bir süreçle tespit edilir. Bu analiz sürecinde durumlara karşılık daha önceden belirlenmiş hareketler arasından seçim yapılarak hangi hareketin (Plan) çalıştırılacağı belirlenir.

BDI mimarisinde yapılması gereken karar verme ve çalışma süresinin dengelenmesidir. Çünkü çok fazla karar alma süresi etmenin harekete geçmesini

engeller ve aynı şekilde çok fazla hareket ile az karar alma süresi hedeflere ulaşılmasını imkânsızlaştırır.

2.3 Çoklu Etmen Sistemleri

Çoklu Etmen Sistemleri, otonom hareket edebilen, iletişim kurabilen homojen veya heterojen yapıda etmen denilen küçük yapılardan oluşan sistemlerdir (Wooldridge 2002). Çoklu etmen sistemleri karmaşık problemlere iletişim, koordinasyon ve yardımlaşma yapan etmenler ile çözüm bulmayı amaçlar. Bu sistemler öngörülebilirliği çok düşük ve karmaşıklığı yüksek olan yerlerde kullanılır (Weiss G 1999).

Bu çalışma kapsamında, her bir bireyin sağlık durumunu ve sağlık durumunda oluşan öngörülemez değişiklikleri takip etmek ve çok sayıda hastanın eş zamanlı olarak tek tek incelenmesine izin vermek için Çoklu Etmen Sistemleri kullanılması düşünülmüştür. Bu ve benzeri problemlerde çözüm görevleri farklı birçok etmenin beraber çalışması, iletişim kurması ve kendiliğinden organize olması yoluyla gelişmektedir.

Temel olarak ÇES'te etmenlerin mikro düzeydeki davranışlarından makro seviyede istenilen etkinin elde edilmesi amaçlanmaktadır (Michael et al. 1999).

2.4 Çoklu Etmen Sistemleri İçin Kalite Ölçütleri

Bir çoklu-etmen sistemi mimarisi tasarlanırken dikkate alınması gereken kalite ölçütleri bulunmaktadır. Bunları kısaca özetlemek gerekirse:

- i. Koordinasyon yeteneği: Sistemde bulunan aktörler gerektiği durumlarda yardımlaşabilmeli ve bir etmenin başarısı bir başka etmenin başarısızlığı anlamına gelmediği durumlarda rekabet edebilmeleri gerekmektedir.
- ii. Öngörülebilirlik: Sistem aktörlerinin yüksek derecelere kadar otonomi sahibi olmasından dolayı bu aktörlerin davranışlarını tahmin etmek zor olabilir ve bu nedenle de tüm sistemin davranışını belirlemek de zorlaşabilir.

- iii. Uyum yeteneđi: Sistemin evresel deęişimlere uyum saęlayabilmesi gereklidir. Ayrıca sistem alıřmaya devam ederken yeni etmenler eklenebilmeli ve var olanlar zerinde deęişiklik yapılabilmelidir.
- iv. Modlerlik: Sistemi işlevsel olarak daha esnek hale getirir.
- v. Birleşebilirlik: Daha etkin işlev yürütlmesi için bazı etmenlerin bileşenleri tmleşik bir başka etmene devredilebilir.
- vi. Gvenlik: Sistem aktrlerinin kullandığı verilerin ve aldıkları mesajların doęrulanması konusundaki strateji nemlidir.
- vii. Hazır bulunma: Sunulan hizmetlerin kesintiye uğramasının nlenmesine dair bir strateji olmalıdır.
- viii. Tolerans: Bir etmenin hatası btn sistemin kmesine sebep olmamalıdır. Sistemde aynı hizmeti sunan benzer veya kopya etmenlerin bulunması toleransa katkıda bulunur.

2.5 Etmen Platformu

Etmen platformu, zerinde etmenlerin aktif olarak var olabilmelerini ve fonksiyonlarını yerine getirmesini saęlayan, etmenler ve etmen sistemleri oluřturmak için gereken araları ieren bir yazılım altyapısıdır(FIPA 1997).

Etmen platformlarının saęladığı fonksiyonlar arasında etmenler arası iletişim kanalları, etmenlerin iletişim dili, etmenin platformlar ve cihazlar arasında taşınabilmesi imkânı saęlayan servisler ve sarı sayfalar servisi sayılabilir.

Etmen platformları farklı cihazlara kapsayıcılar (container) aracılığı ile kurulur. Kapsayıcılar birer sanal makine rneđidir ve oklu etmen ağıında bir dğm meydana getirirler. Her bir cihazda en az bir tane olmak zere yzlerce kapsayıcı bulunabilmektedir. Kapsayıcıların nemli bir fonksiyonu ise etmenleri altyapıda bulunan işletim sistemlerinden bağımsız hale getirmesidir. Etmenler bir kapsayıcıdan dięerine taşınabilirler (migration) ve aynı şekilde cihazlar arası geiş işlemi de yapılabilir.

2.6 Etmen Tabanlı Yazılım Geliştirme

Sağlık bilişimi alanında ortam ve gereksinimler sürekli değişmektedir ve tıp alanında yaşanan yeni gelişmeler ile birlikte eski sistemlere devamlı olarak yeni bileşenler eklenmesi gerekmektedir. Bunun gibi sürekli değişen ve gelişen ortamlarda çalışacak, devamlı olarak ortaya çıkan yeni gereksinimleri karşılayacak ve yeni bileşenlerin kolayca eklenmesine izin veren yazılım mimarilerine ihtiyaç vardır. Bu mimariler için başvurulabilecek yöntemlerin birisi etmen tabanlı yazılım geliştirme yöntemidir.

Sağlık alanında her geçen gün eklenen yeni cihazlar, kurallar, tedavi yöntemleri ve hatta kurumlar kullanılan yazılım sistemlerinin çok fazla bir müdahale olmadan farklı farklı ortamlarda çalışmasını, sağlam ve güvenilir yapıda olmasını, belli derecelere kadar otonom olmasını ve belirli derecede pro-aktif davranışlar göstermesini gerektirmektedir.

Etmen tabanlı yazılım geliştirmenin amacı, etmen sistemleri geliştirmede kullanılacak metodoloji ve araçları geliştirmektir. Ayrıca oluşturulan etmen sistemlerinin esnek, kullanışlı ve ölçeklenebilir olması hedeflenmektedir. Çoklu etmen sistemleri tasarlamak amacıyla birçok metodoloji ve araç geliştirilmiştir. Bunlardan en çok popüler olanları Tropos, O-MaSE ve Prometheus metodolojileri ve bu metodolojiler ile birlikte kullanmak amacıyla geliştirilmiş TAOM4E, PDT ve aT³ araçlarıdır.

O-MaSE, nesne tabanlı yazılım geliştirme yöntemini temel alan, kavramları ve modelleri etmen teknolojisine uyarlayan bir metodolojidir. Prometheus ise temelini BDI etmenleri geliştirme ve bu konularla ilgilenenlere akıllı etmen sistemlerinin geliştirilmesi ile ilgili yardımlardan kaynaklanan tecrübeler ve elde edilen bilgi birikiminden almaktadır. Bu çalışmalar sonucunda Prometheus adı altında çoklu etmen sistemlerinin analiz ve tasarımı için modeller ve süreçler belirlenmiştir.

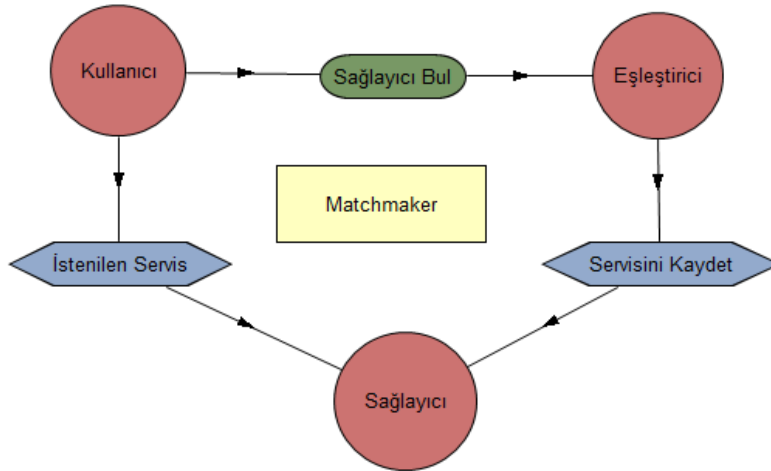
Tropos gereksinim tabanlı bir yol izlemektedir. Yöntemde alan ve gereksinim analizleri yapılmaktadır ve bu analizlerde oluşturulan modeller etmen tabanlı sistem geliştirmesine uyarlanmaktadır. Bu çalışma kapsamında bahsedildiği üzere gereksinimler ön planda olduğundan dolayı Tropos metodolojisi incelenecek ve uygulanacaktır.

Yazılım mimarileri makro seviyede sistem bileşenlerinin, alt sistemlerin ve bileşenler arası ilişkilerin tanımını yapmaktadır. Çoklu etmen sistemleri de tanımına bakıldığında bir organizasyon olarak görülebilir: “Organizasyon, tanımlanabilir sınırları olan, hedef ya da hedeflerine ulaşmak için görece devamlı olarak çalışan ve bilinçli olarak koordine edilen bir varlıktır.”(Morabito et.al. 2009)

Çoklu etmen sistemlerinde kullanılan ve Organizasyon Teorisi dâhilinde tanımlanmış modeller bulunmaktadır. Bu modeller Beşli Yapı (Structure in 5's), Piramit Stili (Pyramid Style), Değerler Zinciri (Chain of Values) ve Matris (Matrix) olarak sayılabilir. (Giorgini et al. 2003)

Çoklu etmen sistemlerinde kullanılan bazı tasarım desenleri şöyle sıralanabilir;

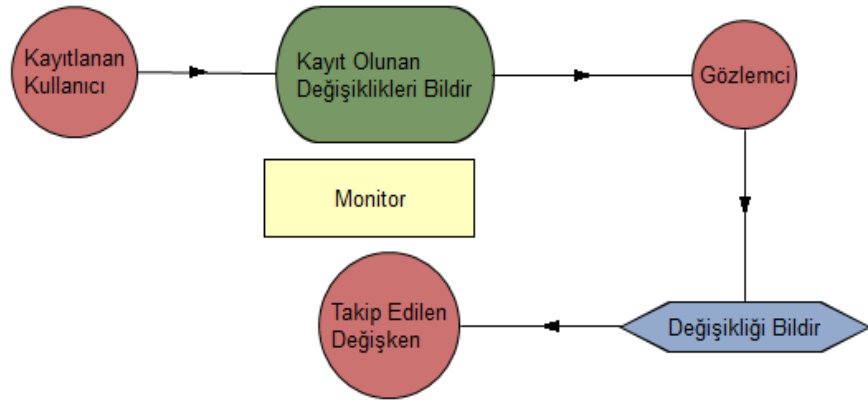
Matchmaker: Eşleştirme Deseni Şekil 2-1 de görülmektedir.



Şekil 2-1 Eşleştirme Deseni

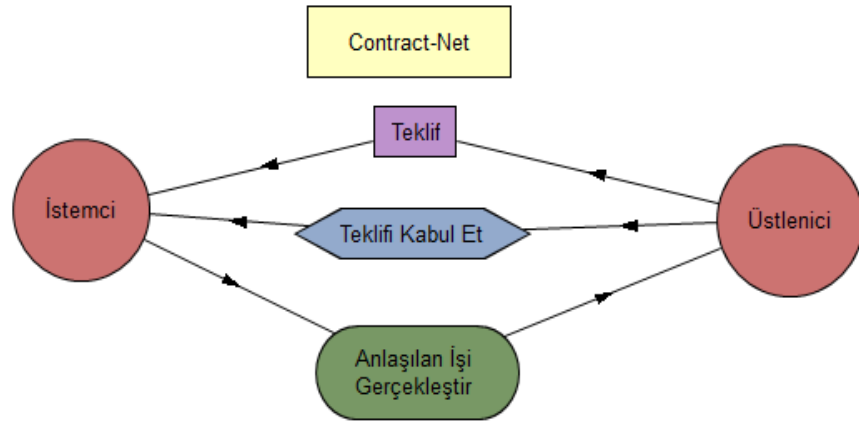
Şekillerin anlamı için Tropos Kavramları bölümüne bakınız.

Monitor: Gözlemci Deseni Şekil 2-2 te görülmektedir.



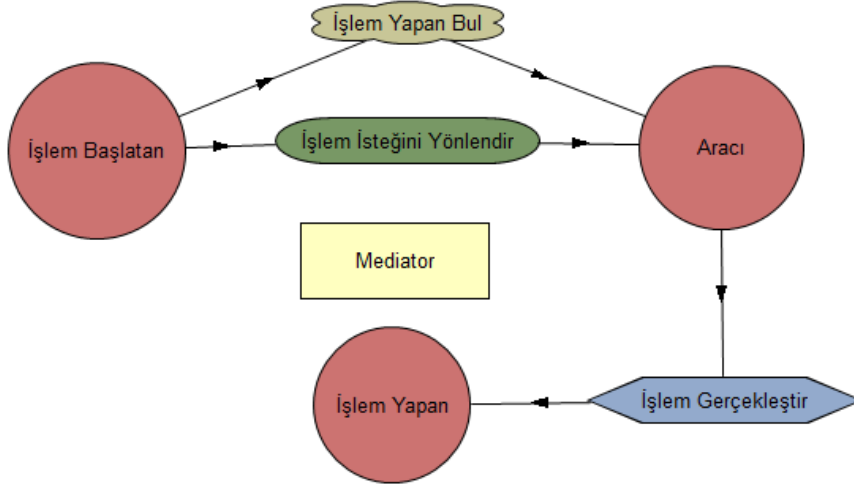
Şekil 2-2 Gözlemci Deseni

Contract net: Anlaşma Deseni Şekil 2-3 te görülmektedir.



Şekil 2-3 Anlaşma Deseni

Mediator: Aracı Deseni Şekil 2-4 te görülmektedir.



Şekil 2-4 Aracı Deseni

2.7 Organizasyonel Desenler

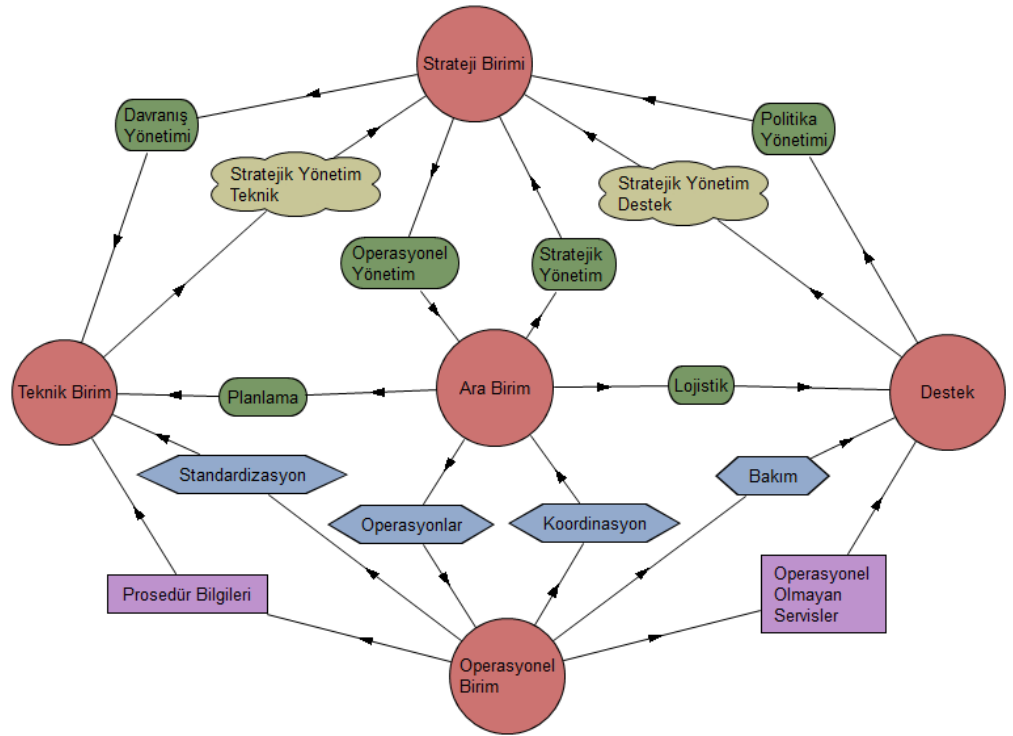
Çoklu etmen sistemleri otonom etmenlerin kendi arasında organize olması ile oluşmaktadır ve dolayısı ile bir organizasyondur. Etmen sistemleri üzerine yapılan çalışmalar incelendiğinde birçok çalışmada sistem mimarileri tasarlanırken organizasyon teorisi kapsamında önerilmiş olan modeller kullanılmaktadır. Bu modeller sadece yeni sistem mimarileri tasarlama esnasında değil aynı zamanda gereksinim analizlerinde alan ve katılımcıların yeni sistem öncesi değerlendirmeleri sırasında da modelleme sürecini kolaylaştırmaktadır.

Bu çalışma kapsamında ve benzer sağlık alanında yapılmış çoklu etmen sistemlerinde uygulanmış veya faydalanılmış bazı modeller incelenecektir.

2.7.1 Beşli Yapı

Beşli Yapı (Structure-in-5) organizasyon teorisi kapsamında tanımlanmış ve bir organizasyonun içyapısını detaylandıran bir modeldir. Organizasyonu beş alt birimden oluşacak biçimde önermektedir. Temel seviyede bütün temel görevlerin ve süreçlerin işletildiği operasyonel birim (operational core) orta seviyede üst ve alt seviyeler arasında hiyerarşik yetkilendirmeleri kontrol eden ve operasyonel birimin parçalarından sorumlu yöneticiler içeren ara birim (middle line) bulunmaktadır. Yine orta seviyede olan ancak ana yetkilendirmelerin dışında

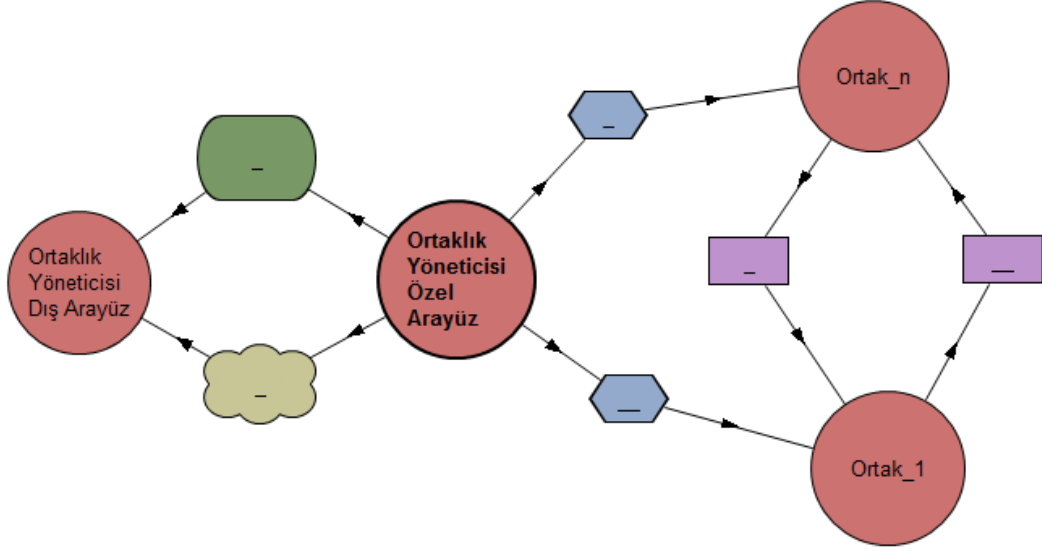
yer alan teknik birim (techno structure) ve destek birimi (support) bulunmaktadır. Teknik birim diğer birimlerin verimliliğini artırma ve adaptasyonu sağlama amacı taşır. Destek birimi ise normal iş akışının dışındaki özel hizmetleri sunmakla yükümlüdür. En üst seviyede ise stratejik birim bulunmaktadır ve amacı genel stratejiyi belirlemek ve yönetim kararları ile organizasyonun hedeflerine etkin bir yolla ulaşmasını sağlamaktır. Şekil 2-5 da detaylı olarak görülebilir.



Şekil 2-5 Beşli Yapı Deseni

2.7.2 Ortak Girişim

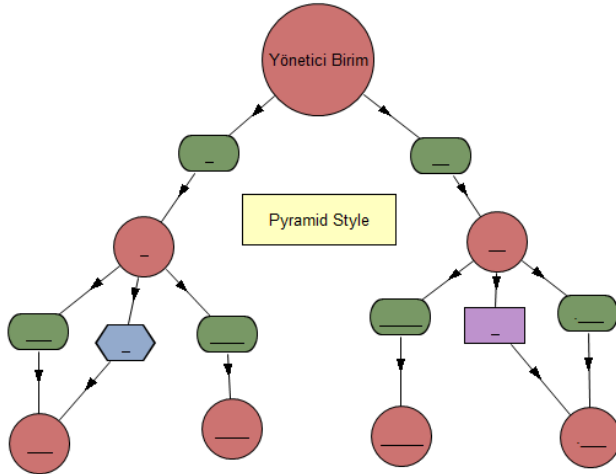
Yine organizasyon teorisi kapsamında önerilmiş bir model olan Ortak Girişim (Joint Venture) iki veya daha fazla aktörün bir anlaşma yaparak daha geniş çapta ve daha düşük maliyetle bir hedefi gerçekleştirmesi veya görevi yapmasını modellemektedir. Ortaklığın yönetimi için özel bir aktör kaynakların ve paylaşımların yönetimini yapar. Ortak girişim deseni Şekil 2-6 de detaylı olarak gösterilmiştir.



Şekil 2-6 Ortak Girişim Deseni

2.7.3 Piramit Stili

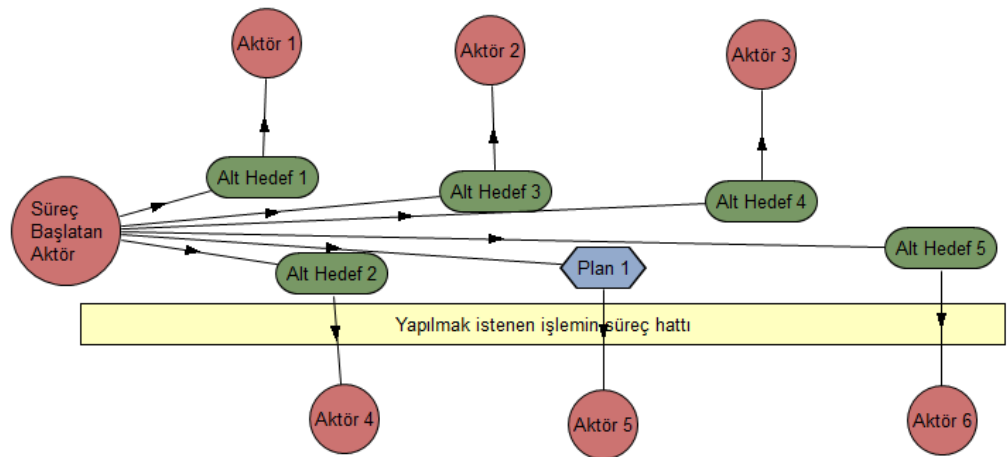
Piramit Stili (Pyramid Style) bir hiyerarşik yetki yapısıdır. Bütün organizasyon tepe noktasından yönetilir. Diğer seviyeler sadece aldığı emri bir alt seviyeye yönlendirmekle yükümlüdür ve ancak yerel düzeyde karar verir ve koordinasyon yapabilirler. Piramit deseni Şekil 2-7 de gösterilmiştir.



Şekil 2-7 Piramit Stili Deseni

2.7.4 Değerler Zinciri

Değerler Zinciri (The Chain of Values) modelinde bir süreç hattında yer yer devreye giren, ara hedefleri, planları gerçekleştiren farklı aktörlerin çalışması modellenmektedir. Katılımcı aktörler zincirin herbir adımında artı değer katan bir işlem yaparlar. Süreç hattı ve aktörlerin katılımı Şekil 2-8 da detaylı olarak gösterilmiştir.



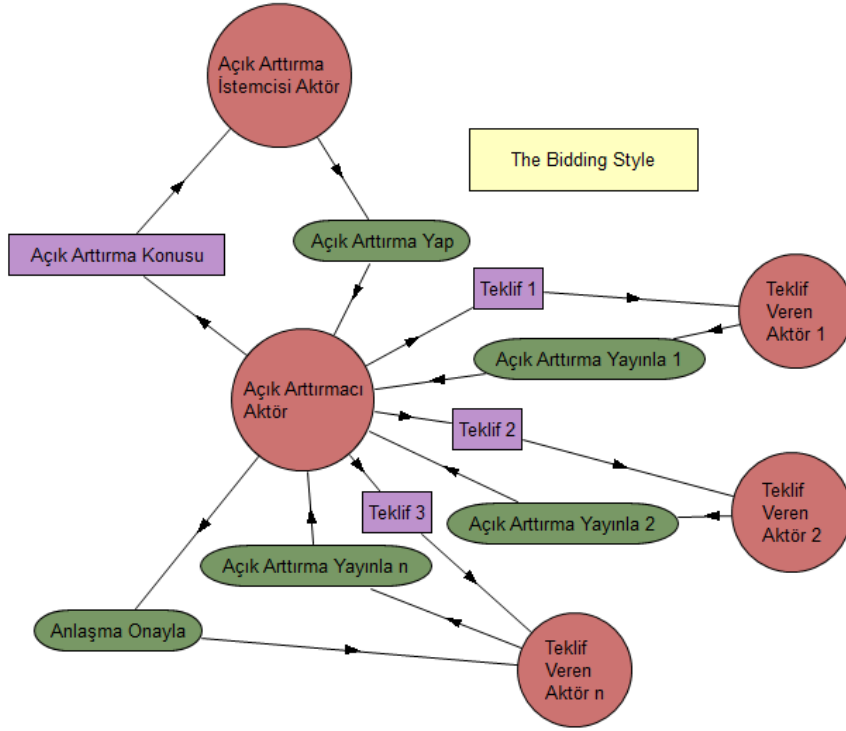
Şekil 2-8 Değerler Zinciri Deseni

2.7.5 Matris

Matris modelinde çoklu iletişim kanalları ve çoklu emir yapısı önerilmektedir ve bu kanalların eşzamanlı çalışması düşünülmektedir. Bu modelde iş akışı tek bir merkezden yönetilmek yerine farklı yetkilendirmeler arasında değerlendirme yapılarak yönlendirilmektedir. Dikey kanallar farklı grupları, yatay kanallar ise aynı grupta olan ve bir yerel yönetici tarafından yönetilen aktörleri temsil etmektedir.

2.7.6 Açık Arttırma Stili

Açık arttırma stili rekabet mekanizmaları ile ilgilidir. Aktörler bir açık arttırmadaymış gibi hareket ederler. Bir açık arttırmacı aktör, sunulacak özneyi (bir hedef, yapılacak bir görev veya kaynak olabilir) istek sahibi aktörden alır ve açık arttırmayı başlatır. Açık arttırmacı aktör teklifleri toplar ve istek sahibi aktöre iletir. Daha sonrasında seçilen bir teklif onaylanarak işlem tamamlanır. Desen Şekil 2-9 da detaylı olarak gösterilmiştir.



Şekil 2-9 Açık Artırma Stili

2.7.7 Emsallere Uygunluk Stili

Emsallere Uygunluk Stili (The Arms-Length Style) modelinde bağımsız ve rekabetçi aktörlerin bir anlaşma yaparak ortak bir işlevi yerine getirmesi söz konusudur. Aktörler kaynaklarını ve bilgilerini ortak hedef için ortaya koyarlar, bu esnada da bağımsızlıklarını ve otonomilerini korurlar. Bu modelde istenilen uyumu sağlamak için müzakereci, gözlemci ve aracılar gibi alt modeller kullanılmaktadır.

2.7.8 Hiyerarşik Anlaşma Stili

Bu modelde emsallere uygunluk ve pramit stilinin koordinasyon ve yetkilendirme mekanizmaları kullanılmaktadır. Buradaki piramit stili hiyerarşi aktörlerin otonomilerini kısıtlar ve ortaklar arasında bir tür kooperatif girişimi oluşturur.

2.8 TROPOS Metodolojisi

Tropos, erken gereksinim analizinden yazılımın gerçekleştirimine kadar tüm yazılım geliştirme adımlarını içeren, temel olarak etmen ve etmen ilişkilerini alan bir yazılım geliştirme metodolojisidir (Mylopoulos et al. 1999). Bu metodolojide ilgili alanın şu andaki durumunun modellendiği, alan katılımcılarının hedeflerinin ortaya çıkarıldığı ve incelendiği erken gereksinim analizi büyük önem taşımaktadır. Tropos adımları, aktörler, hedefler ve ilişkileri temel kavramlar olarak sunan i* çerçevesini benimser.

i* Modeli Eric Yu tarafından çok katılımcı içeren süreçlerin modellenmesi amacıyla önerilmiş bir gereksinim mühendisliği modelidir (Yu 1995). Çalışmanın ilk aşamasında hâlihazırda bulunan sistemin incelemesi yapılmaktadır. Bu aşamada sistem ile ilgili ne, nasıl ve neden sorularına cevaplar verilmektedir. Sistem sadece kendi içinde değil aynı zamanda insan aktörler ile ilgili olarak ta ilişkisel açıdan değerlendirmeye alınmaktadır. Model ilk aşamasında verdiği cevaplar ile sistem bağımlılıklarının detaylı analizini çıkarmakta ve işlevsel olan ve olmayan gereksinimleri belirlemekte tümleşik bir çözüm sunmayı amaçlamaktadır. Model kavramsal olarak aktör, hedef, aktör bağımlılıkları elemanlarını kullanmaktadır (Yu 1995).

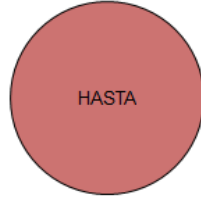
2.8.1 Neden TROPOS

Sağlık alanı bilişim açısından bakıldığında ihtiyaçları gün be gün değişen, karmaşıklaşan ve çok daha sofistike yaklaşımlar gerektiren bir alandır. Bu alanda geliştirilecek sistemlerin tamamen gereksinimlere dayalı olması ve bu gereksinimlerin kısa zaman dilimleri içinde değişmesine hazırlıklı olması gereklidir. Sistem gereksinimleri belirlenerek oluşturulan yapı gerekli görüldüğü anda geriye dönük herhangi bir tasarım aşamasında değişime gitmeye müsait olmalıdır.

TROPOS metodolojisi bu bağlamda i* gereksinim analizi modeli üzerine inşa edilmiş olması itibarı ile diğer etmen tabanlı yazılım geliştirme metodolojilerinden ayrılmaktadır. TROPOS metodolojisi tasarım amacı taşımakla beraber gereksinim analizinin erken aşamalarını da içerdiği için bilgi sisteminin çalışacağı ortamı da derinlemesine anlamaya yardımcı olmaktadır (Bresciani et.al. 2004).

2.8.2 Tropos Kavramları

Aktör (Actor) : Gerçekleştirilen sistem üzerinde veya organizasyonel kapsamda stratejik hedefleri veya niyetleri olan varlık. Bir aktör fiziksel, sosyal veya yazılımsal bir etmen olabileceği gibi bir rol veya pozisyon da olabilir (Bresciani et.al. 2004). Şekil 2-10 de bir alan ortağı aktörün gösterim biçimi verilmiştir.



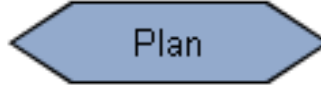
Şekil 2-10 Aktörlerin Gösterimi

Hedef (Goal) : Aktörlerin ilgi alanlarına ait stratejik amaçlardır. İki tür hedef vardır somut hedef (hardgoal) ve soyut hedef (softgoal), somut hedef alt hedefler ve planlar ile işletilerek bir iş yapan veya sonuç üreten hedeflerdir. Soyut hedefler ise net bir hedef ya da plan ile net bir sonuç üretmeyen beklentileri içerir ve herhangi bir fonksiyonu bulunmaz. Şekil 2-11 de hedeflerin gösterim biçimi verilmiştir.



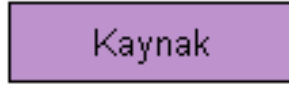
Şekil 2-11 Hedeflerin Gösterimi

Plan: Bir işi yapmak için gereken işlemler bütünüdür. Somut bir hedefi gerçekleştirmek veya soyut bir hedefe katkıda bulunmak için kullanılır. Şekil 2-12 de planların gösterim biçimi verilmiştir.



Şekil 2-12 Planların Gösterimi

Kaynak (Resource) : Bir bilgi parçası veya maddi bir varlık kaynak olarak adlandırılır. Şekil 2-13 te kaynakların gösterimi verilmiştir.



Şekil 2-13 Kaynakların Gösterimi

Bağımlılık (Dependency): Bir aktörün bir hedefi gerçekleştirmek, planı çalıştırmak veya bir kaynağa ulaşmak için bir başka aktöre bağımlı olması durumu. Bir sebepten bağımlı olan aktöre “bağımlı (dependor)” , bağımlılık sebebine “bağlantı (dependum)” ve işlemi yapan aktöre ise “sağlayan (dependee)” denilmektedir. Bir aktör bir başka aktöre bağmlanarak normal şartlarda kendisinin yapamayacağı veya daha iyi yapamayacağı bir işlemi yapmaya imkan bulur. Ancak sağlayan görevini yerine getiremez ise bu durumda bağımlı aktör de hedefini yerine getiremeyebilir.

Yetenek (Capability): Bir aktörün hedeflerine ulaşmak için uygulayabileceği planların tümüdür. Aktör çevresinden aldığı bilgiler ve kendi durumunu değerlendirdikten sonra hedefi için planlardan bir veya daha fazlasını tercih edebilir.

Kanı (Belief): Kanılar bir aktörün çevresi ile ilgili bilgilerinden oluşmaktadır.

2.8.3 Modelleme İşlemleri

Gereksinim modelinden başlayarak adım adım sistemin son ve detaylı modellemesine giden işlemlere modelleme işlemleri denilmektedir (Girogini et al. 2005).

Aktör Modelleme: Sistemin çalışacağı çevrede bulunan aktörlerin ve sistem aktörleri ile etmenlerinin belirlenmesi ve analizi işlemidir. İlk olarak organizasyondaki alan paydaşları belirlenerek aktör olarak modele eklenir. Belirlenen aktörlere daha sonra bir aktör halinde sistem aktörü modele dâhil edilir. Diğer modelleme işlemlerinden sonra sistem aktörü ayrıştırılarak alt aktörlere ayrılır. Bu yeni oluşturulan aktörler ilgili plan modellemesinden daha sonra yazılım etmenlerine dönüşecektir.

Bağımlılık Modelleme: Aktörler arasındaki bağlantıları (dependum) belirlemek için yapılan işlemdir. İlk olarak organizasyon içindeki sosyal aktörler (sistem dışındaki aktörler) arasındaki bağımlılıklar belirlenir. Bir sonraki aşamada sistem aktörünün eklenmesi ile birlikte sosyal aktörler ile sistem aktörü arasındaki bağımlılıklar belirlenir. Bu işlemde sosyal aktörler arasındaki bağımlılıklardan hangileri sistem aktörüne aktarılabilceği belirlenir.

Hedef Modelleme: Aktörlerin her biri ele alınarak, ilgili aktöre ait hedeflerin analizi yapılır. Analiz üç işlemde oluşur,

Kestirme yol analizi (means-end analysis): Bir hedefi gerçekleştirmek için gereken plan, hedef veya kaynakların belirlenmesi.

Katkı analizi (contribution analysis): Araçların hangi hedeflere katkı sağladığının belirlenmesi. Hedefler birbiri arasında olumlu veya olumsuz katkı yapabilirler.

Ve/Veya ayrıştırması (and/or decomposition): Hedefler bazı alt hedeflere bölünebilirler. Bu alt hedeflerin tamamı bir üst hedefin gerçekleşmesini sağlayabileceği gibi, bu üst hedefin gerçekleşmesini sağlayabilecek alternatif alt hedef grupları da bulunabilir.

Plan Modelleme: Hedef modelleme analizi işlemleri bu modelleme analizi için de geçerlidir.

Kestirme yol analizi: Bir planı gerçekleştirmek için gereken plan ve kaynakların belirlenmesi.

Katkı analizi: Araçların hangi planlara katkı sağladığının belirlenmesi işlemidir.

Ve/Veya ayrıştırması: Planlar bazı alt planlara bölünebilir. Alt planlar birbirinin tamamlayıcısı olabileceği gibi birbirinin alternatifi de olabilirler.

Yetenek Modelleme: Bu modelleme işlemi sistemin alt aktörleri de belirlendikten ve bu aktörlerin tüm diğer modellemeleri (bağımlılık modellemesi, hedef modellemesi, plan modellemesi) tamamlandıktan sonra yapılmaktadır. Aktörün hedeflerini gerçekleştirmesi ve planlarını uygulaması için bazı bireysel yeteneklerin belirlenmesi gereklidir. Ayrıca diğer aktörler ile bağımlı olduğu işlemler için iletişim kurabilmesi amacıyla bazı sosyal yeteneklerin de belirlenmesi gerekmektedir. Tüm tasarım işlemlerin sonunda bu yetenekler iyice detaylandırılarak gerçekleştirim esnasında kodlamaları yapılır.

2.8.4 Tropos Adımları

2.8.4.1 Erken Gereksinimler Analizi (Early Requirements Analysis):

Bu aşamada alan paydaşları belirlenerek işleyen sistemin mevcut durumu modellenir. Paydaşlar hedefler, işletilmesi gereken planlar ve kaynaklar için birbirine bağımlı sosyal aktörler olarak modellenir. Daha sonra bu aktörlerin birbirlerine olan bağımlılıkları belirlenir. Örneğin aktörler arasında hangi hedeflere ulaşılması, hangi planların işletilmesi ve hangi kaynakların işlenmesi için bir başka aktöre bağımlıdır, bunlar belirlenir. Aktörlerin niyetleri (intentions) hedefler (goal) olarak ele alınır. Belirlenen temel hedefler, hedef-odaklı bir analiz yapılarak daha detaylı alt hedeflere bölünür ve bu işlem aynı zamanda alternatiflerinde değerlendirmesini sağlar.

Bu aşamanın sonucu olarak aktörler ile ilgili hedefler, planlar ve kaynaklar listesi oluşmaktadır, bundan dolayı tüm hedeflerin ve planların analizlerinin tamamlanması gerekmektedir.

2.8.4.2 Geç Gereksinimler Analizi (Late Requirements Analysis):

Bu aşamanın amacı sistem aktörünü çalıştığı çevre ve işlevleri açısından analiz etmektir. Erken gereksinimler analizinde ortaya çıkan aktörlerin yanısıra bir adet sistem aktörü eklenir ve analize başlanır. Sistem aktörü ile diğer aktörler arasında bağımlılık analizi yapılarak daha önce diğer aktörler arasında tanımlanmış bazı bağımlılık ve hedefler bu andan itibaren sistem aktörüne tanımlanır. Belirlenen bağımlılıklar sistemin işlevsel olan ve işlevsel olmayan gereksinimlerini ortaya koymaktadır.

Belirlenen gereksinimler plan modelleme ve hedef modelleme yöntemleri ile analiz edilerek gerekli aktör diyagramı tamamlanmış olur. Burada yapılacak analizlerde hedefler alt hedeflere veya alternatif hedefler ayrılabilir ve planlara bağlanabilirler. Benzer şekilde planlar da alt planlara veya alternatif planlara ayrılabilirler.

2.8.4.3 Mimari Tasarım (Architectural Design):

Bu aşamada oluşturulacak sistemin genel mimarisi analiz edilir ve sistem birbirine veri ve iş akışları ile bağlı alt sistemler olarak incelenir. Burada geçen alt sistemler, yeni tanıtılacak sistem içi aktörleri ve veri ile iş akışları ise belirlenecek bağımlılıkları işaret etmektedir. Bu tasarım üç adımda oluşturulmaktadır;

İlk adımda sistemin genel mimari yapısı tanımlanmaktadır. Sisteme ait aktörler ve alt aktörler, aşağıdaki seviyelerde yapılan analizler ile mimari tasarım diyagramına eklenmektedir.

- a. Yeni aktörlerin tanımlanarak sistem hedef analizi ile alt hedeflerin bu aktörlere aktarılması.
- b. Mimari tarz (örn. Servis odaklı mimari) seçimine göre yeni aktörlerin tanımlanması.
- c. İşlevsel olmayan gereksinimlere katkıda bulunacak aktörlerin tanımlanması.

Bu aşamanın devamı olarak yeni tanımlanan aktörler arasındaki bağımlılıkların (hedefler, kaynaklar, planlar...) belirlenmesi ve gerekli analizler yapılmaktadır. Belirlenen aktörlerin detaylı hedef analizi, plan analizi ve yetenek

analizi yapılmaktadır ve gerek görülürse bu sistem aktörleri de alt aktörlere bölünebilir ve belirlenen bağımlılıklar bu alt aktörlere aktarılabilir.

İkinci adımda aktörlerin hedeflerini gerçekleştirmek için gereksinim duyduğu yeteneklerin belirlenmesi işlemi yapılmaktadır. Yetenekler bir aktörün detaylı analizinin sonucunda oluşan genişletilmiş aktör diyagramına bakılarak kolayca anlaşılabilir. Aktörün sahip olduğu bağımlılıkların her biri bir veya daha fazla yetenek eklenmesini gerektirebilir. Örneğin “hasta bilgi formu” adında bir kaynak bağımlılığı var ise bu durumda aktör “hasta bilgi formu al” şeklinde bir yeteneğe ihtiyacı var demektir.

Son adımda belirlenmiş olan yeteneklere göre etmen türleri tanımlanmaktadır. Bu türler kullanıcı arayüz etmeni, etmen kaynak aracı ve sorgu işleyen olarak örneklendirilebilir.

Tropos yukarıda bahsedilen adımlardan herhangi birinde geriye dönük tasarım değişimi yapılmasına müsaittir. Böylece oluşacak gereksinim değişimlerinde yapılan tasarıma istenilen aşamada sadece ilgili bölümlerde güncelleme yaparak adımları tekrar ileriye doğru izlemek ve sistemi hızlı bir şekilde gereksinimlere uygun hale getirmek mümkün olmaktadır.

2.8.4.4 Detaylı Tasarım (Detailed Design):

Detaylı tasarım işleminde daha önceki aşamalarda belirlenmiş olan etmenlerin detaylı özellikleri belirlenir. Etmenlerin hedefleri, planları ve birbiri arasındaki iletişimleri detaylı olarak belirlenir. Bu aşamada uygulanacak yöntem seçilen platform ve geliştirme araçları ile doğrudan bağlantılıdır. Genel olarak OMG (Object Management Group) ve FIPA (Foundation for Intelligent Physical Agents) organizasyonları UML (Unified Modelling Language) kullanılmasını desteklemektedir.

Mimari tasarım aşamasında ortaya çıkan etmenler birer birer ele alınarak tasarım işlemine tabi tutulurlar. Belirlenmiş olan yetenekler ve planlar UML aktivite diyagramları ile gösterilirler. Etmenler arasındaki protokolleri tasarımılamak için ise Agent UML kullanılabilir.

2.9 Sağlık Bilişimi

Güvenilir ve güncel sağlık bilgisi, sağlık sistemlerinin temeli olmasına rağmen veri toplanması, analizi, dağıtımı ve kullanımı sistemleri ile ilgili yetersiz yatırımlar nedeniyle, genellikle sağlanamamaktadır. Bununla beraber karar vericiler sorunları ve ihtiyaçları tanımlayamamakta, süreci takip edememekte, değişimin etkilerini değerlendirememekte ve buna uygun kararları alamamaktadır.

Elektronik hasta kayıtları, kurum içi ve kurumları arası iletişim, farklı sistemlerin entegrasyonu ve karar destek sistemlerini geliştirilmesi için sağlık bilişimi alanında standartlara ihtiyaç vardır. Sağlık sistemlerinde standartların kullanım zorunluluğu öncelikle entegrasyon ihtiyacından kaynaklanmaktadır. Sağlık sistemlerinde tam bir çözümün tek bir firma veya kurum tarafından yapılabilmesi mümkün olamamaktadır. Çünkü sağlık alanında oluşan ve sürekli değişen ihtiyaçlar birçok çeşitli ve büyük sistemlerin farklı uzmanlık alanlarında oluşturulmasını zorunlu kılmaktadır. Dolayısı ile tek elden tam bir çözüm elde edilemeyeceğinden bu sistemlerin bir standart dahilinde entegre edilmesi ve yönetiminin sağlanması gerekmektedir. Ayrıca kullanıcı sayısı arttıkça kişi başına düşen maliyeti azaltmaya yardımcı olmaktadır.

Standartlar, alanında uzman ve tecrübeli kişilerin biraraya gelerek ortaya koyduğu, temel olarak üzerinde anlaşılmış ve tekrar uygulanabilir iş yapış yollarını belirten dokümanlardır. Standartlar kural, yönerge veya tanım olarak kullanılmak üzere tasarlanmış teknik şartnameler ve diğer açık kriterleri içerirler. Standartların kullanılması ürünlerin, sistemlerin ve hizmetlerin güvenilirliğini ve etkinliğini artırır.

Bilgilerin doğru değerlendirilmesi ve risk yönetimi standart kullanımını zorunlu kılmaktadır. Örneğin her yıl önemli bir miktar insan hayatını reçete hataları yüzünden kaybetmektedir. Sağlık verilerinin standartlaştırılmış bir biçimde toplanması ve analiz edilebilir olması gerekmektedir. Bu alanda büyük miktarlarda veri üretimi gerçekleştirilmektedir ancak birçoğu analiz ve değerlendirmeye gerek görülmeden sadece kayıt edilmektedir. Standardizasyon ile bu verilerin analizi mümkün olabilecektir.

2.9.1 Sağlık Bilişiminde Standart Geliştiren Kuruluşlar

Sağlık bilişimi alanında standart geliştiren kuruluşların başında Healthcare Information Technology Standards Panel (HITSP) kuruluşu gelmektedir ve kamu ve özel sektörler arasında bir yardımcı işlevi görerek genel kabul gören ve işe yarar standartlara ulaşılmasını amaçlamaktadır. Bu sayede de sağlık bilişimi uygulamalarının birbiri arasında yerel, bölgesel ve ulusal sağlık ağında geniş bir şekilde birlikte çalışabilirliğini sağlamayı amaçlamaktadır.

Sağlık bilişiminde standartların oluşması şu şekilde olmaktadır:

- Amaca özel (Ad hoc) : DICOM gibi standartlar medikal görüntü iletişimi için kullanılan yazılımlar tarafından ortaya koyulmuş standartlardır.
- De facto: Kullanılan altyapı sistemleri nedeniyle oluşan standartlar. Örnek olarak, Microsoft işletim sistemlerini kullanan sistemlerin Microsoft ürünlerinin getirdiği standartları kullanması verilebilir.
- Kamu kurumlarının yönlendirmesi ile standartlar oluşabilmektedir.
- Uzlaşma: Sistem üreticilerinin arasında yapılan uzlaşma ile de standartlar oluşturulabilmektedir.

Standart geliştiren uluslararası kuruluşlardan bazıları ISO (International Organization for Standardization), CEN (European Committee for Standardization), WHO (World Health Organization) olarak sayılabilir. Daha önce bahsi geçen HITSP kuruluşu ANSI sponsorluğunda bir ulusal kuruluştur. Diğer bir kuruluş olan OASIS (Organization for the Advancement of Structured Information Standards) ise kar amacı gütmeyen bir konsorsiyumdur.

2.9.2 Geliştirilen Standartlar

HITSP standartlar için üst veri modelleri oluşturmuştur ve standartlara uyan sistemlerin test edilmesi ve gerçekleştirimi amacıyla üst veri kayıtçıları (metadata registries) kullanmaktadır.

2.9.3 HITSP Klinik Doküman ve Mesaj Terminolojisi Bileşeni

Bu bileşen dokümanının amacı doküman tabanlı veya mesaj tabanlı yapılar için bir sözlük oluşturmaktadır. Bu yapılara örnek CDA (Clinical Document Architecture) dokümanları ve HL7 mesajları verilebilir.

2.9.4 ASC X12

Bu standart sağlık hizmeti veren kurumlar ile ödeme yapan kurumlar arasındaki idari işlemler ile ilgili standartları içerir.. ASC X12 standartının temel mesajlaşma birimi bölümlerden ve döngülerden oluşan bir işlem setidir. İşlem setleri kontrol bölümleri tarafından ayrılırlar. Bir bölüm ise bir veya daha fazla birleşik veri yapısı veya basit veri elemanları içerir. Bölümler tekrarlanabilir ve bilgi aralara eklenebilir.

Örnek bir ASC X12 mesajı.

```
ISA*00* *00* *ZZ*4137147 *ZZ*PLANA
*060119*1545*U*00401*999999999*0*P*:~
GS*HS*4137147*PLANA*20060119*1545*999999999*X*004010X092A1~
ST*270*0001~
BHT*0022*13*999999999*20060119*1545~
HL*1**20*1~
NM1*PR*2*****PI*00999~
HL*2*1*21*1~
NM1*IP*1*ALLISON*JASON*H***SV*ABC123DEF~
HL*3*2*22*0~
NM1*IL*1*ARROJO*ROLANDO*A***MI*5643296~
DMG*D8*19710102~
EQ*30~
SE*11*0001~
GE*1*999999999~
IEA*1*999999999~
```

*(<http://wiki.hitsp.org/docs> (Erişim tarihi: Ağustos 2014))

İşlemler bir numara ile belirtilirler ve yukarıdaki örnek 270 numaralı işleme ait bir örnektir. Burada bölümler iki veya üç karakterli bir başlangıç ile tanımlayıcı belirtilen ve ~ karakteri ile biten satırlarda gözlenmektedir. Birleşik veya basit veri elemanları yıldız (*) işaretleri arasındaki bölümlere konulmaktadır.

2.9.5 DICOM

Digital Imaging and Communications in Medicine (DICOM), tıbbi görüntülerin işlenmesi, saklanması, basılması ve iletilmesi ile ilgili standartları içermektedir. Farklı üreticileri tarayıcıları, sunucuları, yazıcıları ve ağ donanımları

DICOM standartı sayesinde bir resim işleme ve iletişim sistemine (PACS) entegrasyon yapabilirler. Bu standart ayrıca ISO 12052 olarakta bilinir.

2.9.6 HL7 V3

HL7 klinik ve yönetimsel verilere yönelik standartları içermektedir.

Yönetimsel verilere yönelik standartları :

- i. Muhasebe ve Faturalama (Accounting and Billing)
- ii. Talep ve Harcamalar(Claims and Reimbursement)
- iii. Veri Erişim İçeriği (Data Access Content)
- iv. İlaç Tutarlılığı Raporu (Drug Stability Reporting)
- v. Ana Dosya/Kayıtçı Yapısı (Master File/Registry Infrastructure)
- vi. Tıbbi Kayıtlar (Medical Records)
- vii. Personel Yönetimi (Personnel Management)
- viii. Program (Scheduling)

Klinik verilere yönelik standartları :

- i. Care Provision; Professional Services
- ii. Care Record Topic
- iii. Care Structures Topic
- iv. Care Transfer Topic
- v. Clinical Genomics, Pedigree
- vi. Implantable Device Cardiac – Follow-up Device Summary
- vii. Individual Case Safety Report
- viii. Notifiable Condition Report
- ix. Queries Care Record Topic
- x. Regulated Product Submission
- xi. Regulated Studies Annotated ECG
- xii. Regulated Studies, Periodic Reporting of Clinical Trials Lab Results
- xiii. The Annotated ECG Implementation Guide
- xiv. V3 Clinical Trials Lab Results Implementation Guide
- xv. V3 Individual Case Safety Reporting
- xvi. V3 Care Provision: Chronic Disease Management, Implementation Guide

Diğer HL7 Standartları:

Version 3 Messaging Standard: İşlem mesajları için birlikte çalışabilirlik yönergeleri içermektedir. V3 Mesajları mesaj çerçeveleri, sıralı etkileşimler ve model tabanlı mesaj içerikleri kavramlarını kapsar.

Version 3 Rules/GELLO: GELLO karar destek sistemleri için bir kodlama dili standarttır. GELLO, UML sınıf modelleri üzerinde kısıt veya sorgu yazmak için geliştirilmiş OCL (Object Constraint Language) dili temel alınarak geliştirilmiştir. HL7V3 RIM (HL7 Version 3 Reference Information Model) ve R-MIM (Refined Message Information Models) UML tabanlıdır ve GELLO HL7 veri sözlüğünü ve veri tiplerini kullanarak HL7 modellerinin anlamsallığını pekiştirmek için kullanılmaktadır.

Arden Syntax: Arden bir kurallar söz dizimidir. Böylece kurallar bilgisayar sistemlerinden bağımsız olarak alınıp kullanılabilir.

CCOW/Visual Integration: Görsel entegrasyon mesajları, uygulamalarının görsel entegrasyonu ile ilgili standartları içerir. Tek giriş ve tek hasta gözleme ile tek oturum ile birden fazla uygulamaya erişim veya tek hasta bilgisi ile farklı uygulamalar arasında geçişler sağlanmaktadır. Mesajlar bir kullanıcı tanımlayıcı, hasta tanımlayıcı veya gözlem tanımlayıcı ile uygulamalar arasında geçiş yapmaya olanak verir.

Claims Attachments: Ambulans, rehabilitasyon veya acil servis gibi taleplerin yapılabilmesi için gereken ek klinik veya yönetsel bilgileri içerirler.

Clinical Document Architecture (a V3-based standard) : CDA klinik dokümanları için iletişim modellerini içerir ve tıbbi kayıtların elektronik ortamda takibini hedefler. XML, RIM ve veri sözlükleri yardımıyla dokümanları hem makine tarafından anlaşılabilir hem de ihtiyaç duyan insanlar tarafından okunabilir hale getirmektedir. CDA dokümanları web tarayıcıdan gösterilebileceği gibi cep telefonu gibi cihazlarda da gösterilebilir.

Electronic Health Record / Personal Health Record: HL7 bir elektronik sağlık kayıt sisteminde bulunabilecek fonksiyonların bir referans listesini içerir.

Bu liste kullanılarak sistem fonksiyonlarının tutarlı bir biçimde tanımlanmasını sağlar.

Structured Product Labeling (a V3-based standard) : SPL yönergeleri bir ilaca ait reçetelendirme bilgilerini içeren dokümanların yazım standartını belirler. Bu dokümanlar düzenleyici makamlar veya ürünleri hakkında resmi bilgi sağlamak durumunda olan organizasyonlar için gereklidir.

Örnek HL7 V3 mesajı

```
<ClinicalDocument>
<realmCode code='US'/>
<typeId extension='POCD_HD000040' root='2.16.840.1.113883.1.3'/>
<templateId root='1.3.6.1.4.1.19376.1.5.3.1.1.1'/>
<templateId root='2.16.840.1.113883.10.20.3'/>
:
```

* <http://wiki.hitsp.org/docs> (Erişim tarihi: Ağustos 2014)

2.9.7 NCPDP

NCPDP eczane hizmet işlemlerinde kullanmak üzere oluşturulmuş mesajlaşma standartları içerir. NCPDP kuruluşu ilaç üreticileri, ilaç dağıtım şirketleri, sigorta kuruluşları, eczaneler, danışmanlar ve yazılım firmaları gibi kuruluşlardan oluşmaktadır.

Mesaj düzeni:

- Aktarılabacak mesaj
 - Başlık bölümü (Header Segment)
 - Hasta bölümü (Patient Segment)
 - Sağlık sigortası bölümü (Insurance Segment)
 - İşlem Bölümü (Transaction 4 adete kadar)
 - Talep Bölümü (Claim Segment)
 - İlaç sağlayıcı bölümü (Pharmacy Provider Segment)
 - İlaç yazan bölümü (Prescriber Segment)
 - Tarafların koordinasyonu bölümüCoordination of Benefits Segment

- İşçi ödemesi bölümü (Worker's Compensation Segment)
- DUR/PPS Segment
- Fiyat bölümü (Pricing Segment)
- Kupon bölümü (Coupon Segment)
- Yetkilendirme bölümü (Prior Authorization Segment)
- Klinik bölümü (Clinical Segment)

2.9.8 Sağlık Bilişiminde Standart Geliştirilen Alanlar

Tanımlayıcı standartları, içerik ve yapı standartları, mesajlaşma standartları, tıbbi terminolojiler ile ilgili standartlar, veri güvenliği ve veri gizliliği standartları, veri setleri ve kalite göstergeleri alanları sağlık bilişiminin standart gerektirdiği alanlardır.

2.9.9 Tanımlayıcılar

Tedavi gören hastayı, hizmet veren sağlık kurumunu, hizmet veren sağlık personelinin benzersiz biçimde tanımlamaya yarayan araçlardır. Halihazırda ki sistemlerde sağlık sigortasının bütün bireyleri içermemesi ve çoğu bireyin aynı numarayı kullanmaları sosyal güvenlik numarasını güvenli ve tutarlı olmaktan çıkarmaktadır.

Benzersiz Hasta Tanımlayıcısı (Unique Patient Identifier) : Farklı kurumlardaki hasta kayıtlarının birleştirilmesi amacıyla her hastanın benzersiz bir tanımlayıcı numarası olması gerekmektedir.

Sağlık Bakım Hizmeti Sunanlara Ait Tanımlayıcılar (Provider Identifiers) : Sağlık hizmetini veren kişi ve kurumlara bir tanımlayıcı verilerek işlemlerin takip edilmesi amaçlamaktadır.

Ürün ve Malzeme Etiketlendirme Tanımlayıcıları: İlaçların ve malzemelerin etiketlenmesinde kullanılmaktadır.

2.9.10 İçerik ve Yapı Standartları

Hasta kayıtlarında bulunacak veri elemanlarının belirlenmesi, bulunacak temel veri elemanlarının tanımlanmasını ve bu elemanların veri tipi, uzunluğu ve bu alan için geçerli olabilecek verilerin belirlendiği standartlardır. Bu alanda standart geliştiren kuruluşlar ASTM ve HL7 dir.

ASTM Committee E31, HL7, ADA, CEN TC251 WG1 bu konu ile ilgili standartlardır.

2.9.11 Mesajlaşma Standartları

Daha önce anlatılmış olan HL7, DICOM, ASC X12, NCPDP gibi standartlar mesajlaşma standartlarıdır. Mesajların kimler arasında gidip geleceğine ve hangi tür bilgilerin aktarılacağına göre değişik bir standart seçilebilir.

2.9.12 Tıbbi Terminolojiler ile ilgili Standartlar

Tıp alanında kullanılan terimler ve isimler üzerinde de standartlaşma gereklidir. Bu standartlar şöyle örneklenebilir:

- i. SNOMED (The Systematized Nomenclature of Human and Veterinary Medicine) : Her türlü medikal kavramın kodlanmasını hedefleyen SNOMED standartı hastalık etkeni organizmalar, topografi, morfoloji, kimyasallar, belirtiler, tanı ve tedavi amaçlı girişimler, hastalık ile ilgili araçlar ve aktiviteler, sosyal koşullar ve genel terimler ile beraber 11 farklı alana göre kodlama yapmaktadır.
- ii. LOINC (Logical Observation Identifier Names and Code) : LOINC laboratuvar test sonuçları, klinik gözlemler ve tanısal gözlemlerin tam ve doğru biçimde kayıt edilmesi amacıyla oluşturulan bir standarttır. HL7 tarafından içerilmektedir.
- iii. NDC (The National Drug Code) : Bir ilaç kodlama sistemidir. Kod içinde üretici kodu, paket tipi kodu ve ürün kodu bulunmaktadır.
- iv. UMDNS (Universal Medical Device Nomenclature System) : ECRI tarafından desteklenen bu standart tıbbi araç ve sarf malzemelerinin isimlendirilmesinde kullanılır.

2.9.13 Sağlık Bilişiminde Standartların Kullanıldığı Uygulamalar

Genel olarak tekrar edecek olursak X12N sağlık kuruluşları ile para ödeyenler arasında gerçekleşen idari işlemler için kullanılmaktadır. DICOM tıbbi resim, video gibi görüntüleri saklamak, aktarmak, işlemek gibi konularda

kullanılmaktadır. HL7 ise bir mesajlaşma standartıdır. Ancak HL7 içerik yapıları hakkında ve LOINC ile birlikte terminolojiler hakkında standartlar içermektedir.

- i. Malzeme Yönetimi: X12N
- ii. Faturalama, Hakediş Ödemeleri: X12N
- iii. Bağışıklık veritabanı: HL7
- iv. Kurum Raporları: HL7, X12N
- v. Hizmet Sağlayıcı: HL7, X12N
- vi. Vaka Raporlama: HL7, X12N
- vii. Tıbbi Görüntüler: DICOM
- viii. Dalga Formları: HL7
- ix. Eczane Ödemeleri: NCPDP
- x. Klinik Sistem: HL7

2.9.14 Veri Gizliliği ve Veri Güvenliği

Hasta kayıtlarında verilerin güvenliğinin sağlanması ve gizliliğin korunması çok önemlidir. Özellikle verilerin bir araya getirilerek paylaşılması düşünüldüğünden bu konu ile ilgili çalışmaların yapılması gerekmektedir. Bu nedenle iyi tanımlanmış veri doğrulama, veri güvenliği ve mahremiyet alanında standartlara gereksinim vardır. Bununla ilgili bazı standartlar ve çalışmalar:

- i. ASTM Committee E31 : Bu komite ve alt komiteleri mahremiyet ve güvenlik konularında çalışmaktadırlar.
 - a. E31.17 – “Privacy, Confidentiality and Access”
 - b. E31.20 – “Data and System Security for Health Information
- ii. The Computer-based Patient Record Institute (CPRI): Hasta verilerine ulaşımı kontrol etmeyi amaçlayan bir dokümandır.
- iii. CEN TC 251 WG III: Koruma prosedürleri, dijital imza, güvenlik kategorizasyonu, güvenilirlik ve denetleme mekanizmaları ile ilgili çalışmalar yapılmaktadır.
- iv. ISO TC 215 – WG4: Veri güvenliği, mahremiyeti, ulaşılabilirliği ve tutarlılığı ile ilgili çalışmalar yürütülmektedir.

2.9.15 Veri Kalitesi Göstergeleri

Veri kalitesini etkileyen faktörler aşağıdaki gibidir:

- Sağlık sistemlerinde veri değil bilgi istenmesi ve dolayısı ile hesaplama hatalarının oluşabilmesi.

- Verinin format ve içerik olarak tanımlı olmaması
- Verinin yetersiz zamanda hazırlanması veya talep edilmesi
- Bilgi sisteminde gönderilen verinin kağıt ortamında tekrar istenmesi
- Aynı verinin kısa sürede tekrar istenmesi
- Verinin birden fazla sistem veya birim tarafından istenmesi
- Tekrar kullanılmayacak olan verilerin istenmesi

HEDIS sağlık alanında kalite ölçüsü geliştirme amacıyla uygulanan bir süreçtir. Değerlendirilecek klinik alanın belirlenerek tanımlanması, ilgili literatürün geniş bir biçimde taranması, araçlar kullanılarak ölçünün oluşturulması, bazı paydaşlarla güvenlik çalışması yapılması ve son olarakta alan testi ile faydası güvenilirliği ve geçerliliği doğrulanması adımları ile gerçekleştirilmektedir.

2.9.16 SağlıkNet Uygulaması

Sağlık-NET projesi, sağlık kuruluşlarında üretilen her türlü veriyi standartlara uygun bir şekilde, doğrudan üretildikleri yerden toplamayı ve toplanan bu verilerden tüm ilgilenenler için uygun bilgiler üreterek sağlık alanında verim ve kaliteyi arttırmayı hedefleyen entegre ve güvenilir bir bilgi sistemidir.

E-Sağlık daha kaliteli hizmet beklentilerini karşılamak ve maliyetleri düşürmek amacıyla ortaya konan bir projedir.

Sağlık-NET projesi, Minimum Veri Seti, Ulusal Sağlık Veri Sözlüğü ve Sağlık Kodlama Referans Sunucusu üzerinde standartlara dayalı bir Elektronik Sağlık Kaydı sistemi ile birlikte Karar Destek Sisteminden oluşmaktadır.

- **Ulusal Sağlık Veri Sözlüğü:** Türkiye'deki sağlık kurumlarında kullanılan sistemlerin uymaları gereken terminolojileri içeren bir sözlük uygulamasıdır. Diğer uygulamalar için önemli bir alt yapı sağlamaktadır. Bütün sistemlerde bu sözlük ile bir terminoloji birliği sağlanacaktır.
- **Minimum Veri Seti :** Bu setler sağlık bakanlığının toplayacağı minimum içeriğe sahip veri setlerini ifade etmektedir. Sağlık Kurumları, veri setleri içerisinde yer alan veri elemanlarını kullandıkları bilgi sistemleri aracılığıyla veya Sağlık-NET portalı üzerindeki veri seti bildirim ekranları yardımıyla Sağlık Bakanlığına iletebilecektir.
- **Sağlık Kodlama Referans Sunucusu:** SKRS İhtiyaç duyulan Sağlık Bilgi Sistemi standartlarının mevcutları ve bundan sonra geliştirilecek olan

standartları içerecektir ve bu standartları ilgili tüm kullanıcıların kolay erişebilmesi için açık teknoloji standartları(XML) ile paylaşacak bir platformdur. Kapsadığı alanlar;

- Tanı Sınıflama Sistemi (ICD–10)
 - Tanı Sınıflama Sistemi (ICPC2)
 - İlaç ve İlaç Sınıfları Kodlama Sistemi (ATC)
 - Sağlık Uygulama Tebliği (SUT) Kodları
 - Klinik Kodları
 - Branş Kodları
 - Sağlık Kurumu Kodları
 - Adres Kodları
 - Aşı Listesi
 - Aşı Takvimi Değerleri Listesi
 - Meslek Grupları Listesi
 - Parametreler Listesi
 - Bebek İzlem Listesi
 - Gebe İzlem Listesi
 - Çocuk İzlem Listesi
 - Persentil Değerleri Listesi
 - Olası Tanı Kriterleri
 - Enfeksiyon Etkenleri Tanı Kriterleri
 - Tümör Yerleri
 - Kesin Tanı Kriterleri
 - Histoloji Kodları
 - Doktor Bilgi Bankası
 - Sözlük Veri Kapsamı Alan Kodları
- Sağlık Kodlama Referans Sunucusu (SKRS) ve Ulusal Sağlık Veri Sözlüğü (USVS) ilişkisi: Ulusal Veri Sözlüğünde tanımlanan veriler için bir kodlama veya sınıflama yapılacaksa bu SKRSden sorgulanarak elde edilebilir.
 - SKRS Sistem Kodunun, parametrik değerleri işaret etmesi durumu (Cinsiyet, Medeni hal, Rapor Türü... vb.) Örneğin “Reçete Türü” veri elemanının, alabileceği değerler Ulusal Sağlık Veri Sözlüğünde listelenmiştir

Sağlık-NET Veb Servisleri: Sahadaki verilerin toplanması için 25 adet HL7 veb servisi bulunmaktadır.

Sağlık-Net projesinin hedefleri, tüm sağlık kurumlarının enterasyonunun tamamlanması, sağlık veri setleri ile birlikte mali ve idari veri setlerinin eklenmesi, toplanan verilerin veri gizliliği kurallarına uyarak kurumlar arasında paylaşılması ve son olarakta bakanlık tarafından sağlıkta kaliteyi arttırmak için karar destek sistemlerinin kurulabilmesi bulunmaktadır.

2.9.17 Sağlık Bilişimi Değerlendirme

Sağlık bilişimi kendine hedef olarak merkezi bir hasta kayıt sistemi oluşturmayı belirlemiştir. Ancak sağlık sistemlerinin doğasında varolan fazla heterojen yapı çok çeşitli çözümleri bir arada kullanmayı zorunlu kılmaktadır. Çalışan sistemlerdeki birbirinden bağımsız ve kaotik yapı bu hedefi uzaklaştırmakla beraber önlenabilir sebeplerden dolayı oluşan can kayıpları ve maliyetler çalışmaların hızlandırılarak sürmesi yönünde baskı yaratmaktadır. Standartların kullanılması ile süreç hızlandırılabilen olsada hâlihazırda kullanılan sistemlerin değişmesi gerektiğinden sürecin uzun olacağı anlaşılmaktadır. Türkiye HL7 ile bu alanda standartları kullanmaya başlamıştır. Hedefte Türkiye'nin merkezi sağlık sistemini oluşturmak vardır. Bu ana kadar sadece minimum veri setleri ile kısıtlı bir bölümü gerçekleştirmiş olsada makul bir gelecekte bu hedefi yakalayabilecektir.

Bu çalışma kapsamında önerilen Çoklu Etmen Sistemi sağlık bilişimi kapsamına giren bir sistem olması nedeniyle ilgili standartlara uymalıdır. Türkiye ilgili standartları sağlık kuruluşu dâhilindeki sistemler için uygulamaya koymuştur ve kuruluşlar dışındaki uzaktan hasta takibi gibi alanlarda herhangi bir standart uygulaması koymamıştır. Ancak bu tür sistemler yaygınlaştıkça ve geliştikçe ileride bu konularda da ulusal bazda kullanılacak standartlar yönetmelikler ile belirlenecektir.

2.10 Hematoloji

2.10.1 Hematoloji Hastasının Muayenesi

Hematolojik hastalık şüphesi taşıyan hastaların değerlendirilmesi sistematik olarak derinlemesine bir tıbbi geçmiş analizi ve fiziksel muayene ile yapılmaktadır ve hastalığın türü belirlenmeye çalışılmaktadır (Kaushansky et al. 2010). Doktor, hastanın belirtilerini sistematik bir şekilde belirlemeli, belirtilerin gelişimi ve sebepleri hakkında ve yakın ve uzak zamanlarda oluşan gelişmeleri anlamaya yönelik özel olarak hazırlanmış sorular ile hastanın genel sağlık durumu hakkında olabildiğince detaylı bilgiler edinmelidir. Daha önce yapılmış kayıtların incelenmesi hastalığın başlangıcı ve gelişimi ile ilgili önemli bilgilerin tespit edilmesini sağlar. İrsi ve çevresel faktörler ile ilgili bilgilerde elde edilmeli ve dikkatle incelenmelidir. İlaç kullanımı, yeme alışkanlıkları ve cinsel yaşam da hesaba katılmalıdır. Doktor, tıbbi geçmişi inceleyerek ve fiziksel muayene ile doku ve organ anormalliklerini tespit etmeye ve tıbbi geçmişin işaret ettiği hastalıkların

işaretlerini bulmaya çalışır. Derideki değişimler, karaciğer, dalak ve lenf bezlerindeki genişleme teşhis için önemli bilgiler verebilir. Tıbbi geçmiş ve fiziksel muayene daha sonra kan ve ilik testleri, tıbbi görüntüleme testleri ve biyopsiler ile devam edecek muayene süreci için gereken temel bilgileri sağlamaktadır.

Birincil hematolojik hastalıklar görülme, sıklığı açısından yaygındır ancak diğer hastalıkların yanı sıra ortaya çıkan hematolojik belirtiler daha sıklıkla izlenmektedir. Örneğin kansızlık ve lenf düğümü büyümesi sıklıkla hematolojik bir hastalık sebebiyle ortaya çıkabileceği gibi, hematolojik olarak değerlendirilmeyen başka tür sağlık sorunlarında da benzer sıklıkla görülmektedir.

Çok çeşitli hastalıklar da hematolojik belirtilere sebebiyet verebilir. Örneğin bağ doku hastalığı olan bir hastada, anemi hastalığının bütün belirtileri ve işaretleri gözlemlenebilir ve lenfadenopati izlenebilir, ancak gözlenen ek belirtiler birincil olarak başka bir sorunun varlığına işaret eder.

2.10.2 Hematoloji Hastasının Tıbbi Geçmişi

Tıbbi alanda, teknoloji ve süreç yönelimli yaklaşım izlenmektedir ve giderek hasta sorgusu ve muayenesinde dikkatlice veri toplamak önceliğini yitirmektedir. Hastanın geçmişi ve fiziksel muayenesi hematolojik problemlerin değerlendirmesinde çok önemli bir yer tutmaktadır.

2.10.3 Genel Belirtiler

Kilo Kaybı: Bu belirti birçok önemli hastalığa eşlik etmektedir ve kansızlık (anemi) belirtisi olarak görülebilmektedir.

Ateş: Bu belirti agresif lenfoma ve akut lösemi hastalığının ön işaretlerindendir. Ayrıca kemoterapi sonrası enfeksiyonlarda ateş görülmektedir. Sebebi bilinmeyen ateş görülüyor ise öncelikle lenfoma değerlendirmesi yapılmalıdır. Üşümeler vücutta hemolitik bir süreç olduğunu göstermektedir. Ayrıca gece terlemeleri de düşük dereceli ateş işaretidir ve lenfoma ile lösemi hastalarında görülmektedir.

Bitkinlik, Keyifsizlik ve Halsizlik: Bu belirtiler hematolojik bir soruna işaret edebilir. Ayrıca orta ve ağır dereceli anemi hastalarında görülmektedir. Eğer

anemi bulgusu yok ise demir yetersizliği sebep olabilir. Yavaş gelişen kronik anemilerde hastalar zaman içinde hareket etmeye karşı toleranslarını giderek yitirmektedirler veya başka fiziksel kayıplar da yaşarlar.

Güçsüzlük: Bu belirti anemi sonucunda oluşabildiği gibi diğer hematolojik sorunların nörolojik etkilerinden dolayı bölgesel olarakta görülebilir. Vitamin B12 eksikliğinde (örn. pernisiyöz anemi) yürüme bozuklukları ve bacaklarda kaşıntı ve uyuşma görülebilir. Lösemi, myeloma ve lenfoma hastalarında bir veya daha fazla uzuvda görülen güçsüzlük bu hastalıkların etkisi ile sinir sistemine bir yayılma veya bir basınç olduğunu gösterebilir.

2.10.4 Spesifik Belirtiler

Baş Ağrısı: Anemi ve polisitemi hastalığı hafif ve çok ağır baş ağrılarına sebep olmaktadır. Lösemi veya lenfoma hastalığında da beyne yayılma veya kafa içi basınç artışı sebebi ile baş ağrısı oluşturabilir. Trombositopeni veya diğer kanama bozukluğu hastalarında beyinde kanama olması ani baş ağrılarına sebebiyet vermektedir.

Şaşkınlık ve Kafa karışıklığı: Ağır anemi ve hiperkalsemi hastalıklarında görülür. Ayrıca yüksek dozlu glukokortikoid tedavisinin sonucu olarak görülür. Şaşkınlık veya bariz bunaklık pernisiyöz aneminin işareti olabilir.

Gözler: Gözlerde anemi ve trombositopeni eşliğinde retinal kanama nedeniyle bazen körlük olmaktadır. Ayrıca makroglobulinemi veya lösemi hastalığında aşırı hiperlökositozundan kaynaklanan hiperviskozite de (kanın akışkanlığının azalması) görüşün bulanıklaşmasına neden olmaktadır. Kısmi veya tamamen körlük retinal damarlar veya arterlerde tromboz (kanın pıhtılaşarak tıkanıklığa sebep olması) olmasından kaynaklanmaktadır. Diplopi veya oküler hareket bozuklukları lenfoma, myeloma ve myeloid sarkoma hastalığından kaynaklanan tümörlerin sinirlere yaptığı baskı sonucunda meydana gelen felçlerden kaynaklanmaktadır.

Kulaklar: Baş dönmesi, çınlama ve uğuldama, belirli anemi, lösemi, polisitemi, makroglobulinemi kaynaklı hiperviskozite eşliğinde görülmektedir.

Burun, Boğaz ve Ağız boşluğu: Burun kanaması trombositopeni, pıhtılaşma bozuklukları nedeniyle oluşabilir. Koku alamama ve koku sanrıları pernisiyöz anemi sonucunda oluşabilir. Ekstranodal lenfoma ve granulositik sarkoma burun yollarına yayılabilir ve burada çeşitli tıkanmalara ve başka belirtilere yol açabilir. Dil üzerinde kaşıntı ve ağrı pernisiyöz anemide ve ağır demir ve vitamin eksikliğinde görülür. Kanama bozukluklarında damaklarda kanama görülür. Hiperkalsemi ve myelom hastalığında ağız kuruluğu görülmektedir. Kronik demir yetersizliği olan hastalarda mukoza körelmesine bağlı olarak yutma güçlüğü görülmektedir.

Boyun: Ağrısız boyun şişlikleri lenfoma işareti olarak görülmektedir. Eğer ağrı veya hassasiyet eşlik ediyor ise lenfoma yanı sıra bir enfeksiyon olduğuna işaret etmektedir. Şişlikler giderek yüze de yayılabilir.

Göğüs ve Kalp: Efor harcama sonrası ve bazen de dinlenme esnasında ortaya çıkan nefes darlığı ve çarpıntı anemiye işaret eder. Nodüllerin bronşlara baskı yapması sonucu öksürük oluşabilir. Lenfoma veya multipl myelom hastalığının kaburga ve göğüs kemiklerine yayılması sonucunda göğüs ağrısı oluşabilir. Nefes almayla beraber oluşan ağrı pulmoner enfarktüse (akciğer krizi) işaret eder. Göğüs kemiği hassasiyeti lösemide, miyelofibrosis (ilik problemi) ve şiddetli lenfoma, myeloma yayılmasında görülür.

Mide, Bağırsak: Hiperkalsemi ve azotemi anoreksia, bulantı ve kusma belirtileri gösterir. Birçok hastalığa eşlik eden doluluk hissi, erken doyma, bulantı ve karında huzursuzluk hematolojik hastalıklarda da çok büyümüş dalak nedeniyle görülmektedir. Karın ağrısı, lenfoma yayılması, kanama, hemoliz (alyuvarların büyük miktarlarda yıkımı) ve orak hücre hastalığı karın krizleri nedeniyle görülebilmektedir. Pernisiyöz anemide diyare belirtisi de görülmektedir. Trombositopeni ve kanama bozukluklarında iç kanamadan kaynaklı ağrılar olabilmektedir. Gaitada kan görülmesi kanama bozukluğu hastalıkları ile beraber görülüyorsa bağırsak lezyonuna işaret etmektedir.

Sırt, Kollar ve Bacaklar: Sırt ağrısı şiddetli lenfoma ve akut lösemi nedeniyle görülebilir ve myelom hastalığının en çok görülen belirtisidir. Hematolojik hastalıklar sebep oldukları ürik asit artışı nedeniyle eklem ağrısı ve eklem iltihabına neden olurlar. Ciddi kanama bozukluğu olan hastalarda eklem içi kanamalar belirgin eklem ağrılarını neden olur. Sol tarafta omuz ağrısı dalak

krizine işaret eder ve sağ tarafta ağrı irsi anemi hastalığından kaynaklanan safra kesesi rahatsızlığına işaret eder. Kemik ağrısı hematolojik hastalıklarda sıklıkla görülür. Bu hastalarda alkol alımından sonra kemiklerde dâhil olmak üzere lezyon bölgelerinde ağrı görülür. Bazen tek bazen iki bacakta da görülen ödem lenfomatöz kitlelerin damarlara baskı yapmasından veya damar trombozundan kaynaklanabilir. Damar trombozu ayrıca kollarda da ödem yapabilir.

Deri: Hematolojik hastalıklarda deri ile ilgili belirtiler çok önemlidir. Demir eksikliği hastalarında deri kurudur, saçlar zayıf ve kurudur, tırnaklar ise kırılığandır. Pernisiyöz anemi ve hemolitik anemide sarılık bariz bir şekilde görülmektedir. Lenfoma karaciğere yayıldığında da sarılık görülmektedir. Hemokromatoz hastalarının cildi grimsi veya bronz görünümündedir. Methemoglobinemi, düşük oksijen kapasiteli anormal hemoglobinler ve polisitemi hastalığında derinin çeşitli yerlerinde morarma görülür. Lenfoma hastalığında görülebilir bir lezyon olmadığı halde aşırı kaşıntılar görülebilir. Polisitemi vera hastalığında da banyo sonrası kaşıntı görülmektedir. Peteşi (derinin nokta şeklinde kanaması) ve ekimozlar (deri altına kılcak damarlardan kan sızması nedeniyle morarma) pıhtılaşma bozuklukları ve trombositopeni hastalıklarında görülmektedir. Ekimozlar eğer herhangi bir darbe nedeniyle oluşmadıysa ağrısızdır ve zamanla iyileşir. Kolayca morarma çoğunlukla bir sorun teşkil etmez ancak pıhtılaşma bozukluklarına işaret ediyor olabilir.

2.10.5 İlaçlar ve Kimyasallar

İlaçlar: Toplumda ilaç kullanımı oldukça yaygındır. İlaç kullanımı hematolojik hastalıkları tetikleyebilir veya ilerlemesini hızlandırabilir. Bu nedenle hastaların kullandığı ilaçlar çok dikkatli bir şekilde takip edilmelidir. İlaçların faydalı etkileri, yan etkileri ve ters etkileri mutlaka takip edilmelidir. Bazı ilaçlar hayatın bir parçası haline gelebilir ve ilaç olduğu hasta tarafından unutulabilir veya ilaç olarak görülmeyebilir. Aspirin, laksatifler (müshil), sakinleştiriciler, demir takviyesi, vitaminler ve diğer besin takviyeleri bu gruba girerler. Birde bilinçli olmadan alınan ilaçlar vardır yemeklerin içindeki antibiyotikler ve toniklerin içindeki kinin maddesi gibi. Bu nedenle hastalara sık aralıklarla ve tekrarlayıcı bir şekilde sorgu yapmak tam bir ilaç kullanım geçmişi almak için çok önemlidir. Alkol kullanımı ve eğlence amaçlı ilaç kullanımı bilgisi de tüm hematoloji hastalarından öğrenilmesi gerekmektedir. Ayrıca bitkisel ilaçlar ve katkı maddelerinin kullanımı bilgisi de yargılayıcı olmadan öğrenilmelidir.

Kimyasallar: Hastalar ilaçların dışında bazı kimyasallara maruz kalmaktadır ve bu kimyasallar hematolojik hastalıkları tetikleyebilir ve şiddetlendirebilir. Ayrıca iş ortamında maruz kalınan kimyasallarda dikkate alınmalıdır. Eğer bir toksinden şüpheleniliyor ise hastanın günlük aktiviteleri ve bulunduğu ortam dikkatlice incelenmelidir.

Aşılar: Aşılarda trombositopeni hastalığını şiddetlendirebilir.

Besinler: Emzirilen bebekler eğer demir takviyesi kullanılmıyor ise demir eksikliği anemisi gelişebilir. Beslenme ile ilgili bilgilendirme yapılması yetersizlik kaynaklı anemi oluşmasını engelleyebilir. Vejetaryenler gibi bazı besinlerin tamamen alınmaması veya pişmemiş balık yenmesi megaloblastik anemi hastalığının gelişmesine dair ipucu olabilir.

2.11 Servis Odaklı Mimari

SOA (Service Oriented Architecture) dağıtık BT sistemlerinde farklı sistemlerin işlev ve hizmetlerinin nasıl yapılandırılacağına ve kullanılacağına dair bir mimari modeldir. Başka bir deyişle sistemler arası entegrasyonla ilgilenir. SOA mimarisinde işlevler ve hizmetler web servisleri olarak sunulur. Web servisleri birbirleri ile iletişim kurabilir ve istenilen iş mantığını gerçekleştirirler. Web servisleri tekrar kullanılabilir ve gevşek bağlıdırlar.

2.12 Web Servisleri

Web servisleri SOA mimarisinde hizmetlerin teknik bir gerçekleştirimidir. Web servisleri tekil bir adresten erişilebilir durumdadır ve bilgi alışverişi için kullanılır. Web servislerinin en önemli bileşenleri şöyledir;

WSDL (Web Service Definition Language): XML tabanlı web servis tanımlama dilidir.

SOAP (Simple Object Access Protocol): Web servis çağrılarını yapmak için XML tabanlı bir RPC protokolüdür.

UDDI (Universal Description, Discovery and Integration): Web servislerinin aranıp bulunmasını ve yönetilmesini sağlayan hizmet.

2.13 Ontoloji

Ontoloji kavramı temel olarak varlık bilimi felsefesinden gelir. Bu felsefe varlıkların araştırılması, gerçekliği ve mümkün olup olmadığını sorgular ve varlık biliminin ana öğretilerini oluşturur.

Bilgisayar bilimindeki ontoloji kavramı ise verilerin makineler tarafından anlaşılabilir bir form halinde kayıt edilmesini sağlar (BernersLee et al. 2001). Bu amaçla ontoloji ilk olarak, içerdiği verilerin tanımlamalarını gösteren bir veri bölümü içerir. İkinci olarak ta yine içeriğinde tutacağı verilerin arasındaki ilişkilerin kurallara uygun bir modelini içerir. Bu tanımlamalar ve ilişki modeli kullanılarak eldeki veriden yeni bilgiler türetilmesini sağlar. Bir örnek verecek olunursa, ontoloji de “A B’nin babasıdır” ve “B A’nın oğludur” üçlemeleri kayıt edilmiş olsun. Bu durumda eğer elimizde “Yaman Özgür’ün oğludur” bilgisi bulunduğunda “Özgür Yaman’ın babasıdır.” bilgisini ontolojide açıkça yazılmamış olmasına rağmen elde edebiliriz.

3. TEKNİK ANALİZ

Bu bölümde hedeflenen projede kullanılacak teknikler, yöntemler ve benzer çalışmalar incelenecektir. Projeyi mümkün kılan altyapılar ve çerçeve sistemler incelenerek yöntemsel tercihler belirlenecektir. Son olarakta sistemin kullanımı ile ilgili senaryolar değerlendirilecektir.

3.1 Sağlık Temalı Çoklu-Etmen-Sistemi Çalışmaları

Sağlık alanının tıbbi süreçlerinin doğasından kaynaklanan nedenlerle karmaşık ve çeşitli ölçekli olması nedeniyle oluşturulan ve oluşturulmak istenen sistemleri ve çevreyi modellemekte geleneksel yazılım teknikleri yetersiz kalmaktadır. Yazılım etmenleri ise ölçeklenebilirlik, beraber çalışabilme ve yeniden yapılandırılabilme özellikleri ile bu türden karmaşık ve değişken ortamları modellemek için yeni bir paradigma olarak ortaya çıkmaktadır.

Bu bağlamda yapılan çalışmalar karar destek sistemlerinden, hasta takip sistemlerine, dağıtık veri kaynaklarının yönetilmesinden, sağlık süreçlerinin koordinasyonuna kadar geniş bir aralıkta çözümler önerilmiştir. Bu çözümlerin genel olarak konuları şöyledir:

- i. Kurum içi dağıtık hasta randevu sistemi
- ii. Uzaktan hasta takibi sağlayan sistemler
- iii. Yaşlı hasta takibi sağlayan sistemler
- iv. Hasta gözlem ve muayene amaçlı sistemler
- v. Etmen tabanlı tıbbi karar destek sistemleri
- vi. İnternette tıbbi bilgileri toplayarak organize eden ve bu bilgileri sağlayan etmen sistemleri
- vii. Organ nakli yönetimi için hastaneler arası işbirliği sağlayan çoklu etmen sistemleri.

Bu çalışmada bir hasta takip sistemi önerilmesi nedeniyle hasta takibi konulu çalışmalar detaylandırılacaktır.

3.1.1 Aingeru

Aingeru yaşlı insanlar için tele yardım hizmeti sağlayan bir sistem olarak ortaya konmuştur ve hâlihazırda bulunan tele yardım hizmetlerinin eksiklerini gidermeyi amaçlamaktadır (Tablado et al. 2004). Tele yardım hizmetleri genel olarak kullanıcı talebi ile harekete geçer ve beklenmedik dış etkilere karşı pasiftir. Ayrıca otomatik olarak önemli biyo sinyalleri izlemediği için hizmetleri oldukça sınırlıdır. Buna karşın yalnız yaşayan yaşlı insan sayısı giderek artmakta ve yardım servislerine ihtiyaç duymaktadır. Aingeru bu fikirden yola çıkarak PDA'lar (Personal Digital Assistants), kablosuz iletişim, sematik web ve etmen teknolojileri kullanan ve aşağıdaki özellikleri içeren bir sistem olarak tasarlanmaktadır:

- i. Aktif olma, anomalileri tespit ederek işlem yapabilem yeteneği
- ii. Her yerde ve her zaman çalışabilme
- iii. Yaşamsal değerleri gözlemleyerek gerektiği durumda alarm vermesi
- iv. Son olarakta doktorların bu verileri bir web arayüzü ile görebilmesi.

Aingeru mimarisinin temel yapı taşları aşağıdaki gibi özetlenebilir:

PDA: Aingeru tarafından takip edilecek olan her kullanıcı yanında bir el bilgisayarı (PDA) taşır. El bilgisayarının iki görevi bulunmaktadır. Birincisi kullanıcıyı izlemek ve ikincisi de özel bir durum oluştuğunda kişi ve bakım merkezi arasında bağlantı kurmaktır. Bu bağlantı üç şekilde sağlanmaktadır. Kullanıcı tarafından bir sorun olduğunda alarm verilmesi, sensörlerden gelen verilerin değerlendirilerek bir anomali tespit edildiğinde alarm verilmesi ve doktor randevuları gibi hatırlatmaları yapan ajanda servisi. Bu işlemler için PDA üzerinde etmenler ve iki adet ontoloji bulunmaktadır. Bunlar OperOnt ve MedOnt ontolojileridir.

Kontrol Merkezi: Kontrol merkezleri insanları takip etmek ve verileri izlemek ile görevlidir. Ana fonksiyonu ise bir alarm verilmesi durumunda harekete geçmek ve gerekli işlemleri yapmaktır. Bu amaçla kontrol merkezindeki bir bilgisayarda etmenler, web servisleri, veritabanı ve iletişimleri anlamsal tabanlı gerçekleştirmek için OperOnt ontolojisi bulunmaktadır.

Bakım Merkezleri: Bakım merkezleri birincil seviye sağlık hizmeti veren kuruluşlardır ve Aingeru'nun bir parçasıdır burada çalışmaktadır. Buradaki sistem parçasının amacı doktorların PDA içinde biriken verilere ulaşımını ve doğrudan randevu girilebilmesini sağlamaktır. Bu amaçla etmenler, web servisleri, veritabanı ve OperOnt ontolojileri bulunmaktadır.

Sağlık Merkezleri: Bu merkezler hastanelere karşılık gelmektedir. Bu parçada hastane ile Aingeru entegrasyonu için web servisleri ve OperOnt ontolojisi bulunmaktadır.

Teknik Merkez: Bu merkezdeki parçanın amacı sistemin bakımını ve geliştirilmesini sağlamaktır. Bu amaçla bir etmen güncellemeleri yayınlar ve uygulanmasını takip eder.

Ayrıca iletişim açısından bakılacak olursa etmenler arası iletişimde FIPA ve web servisleri için ise SOAP kullanılmaktadır. Etmenler ve web servislerin iletişiminde kullanılan ontolojiler ise OWL ile yazılmıştır.

3.1.2 Cervantes et al.

Bu çalışma çoklu etmen platformu ve yapay sinir ağlarını biraraya getiren bir karar destek sistemi olarak tasarlanmıştır (Cervantes et al. 2007). Yapay sinir ağları, kronik hastaların takibi esnasında toplanan verilerin analiz edilmesi ve bu yolla hasta durumunda ortaya çıkan olumsuz gelişmeleri tespit edilmesi amacıyla kullanılmaktadır. Çalışmanın temel olarak aldığı çoklu etmen sistemi verilerin toplanması ve yönetilmesi amacıyla kullanılmaktadır.

Çalışma çoklu etmen sistemi tabanlı yaklaşımı ile hastanın tedavisi sırasında ortaya çıkan kronik problemlerin gözlemlenen belirtilerin desenlerinden tespit edilmesi ve uygun tedavinin yönetilebilmesini sağlama amaçındadır. Çalışma mimari açıdan incelendiğinde şu temel yapılar gözlemlenmektedir:

Sensör Etmenleri: Amacı sensörlerden gelen bilgileri toplamak ve ön analizini yapmaktır. Belirtileri Hasta Etmenine gönderir.

Hasta Etmeni: Bu bileşen sensör etmenlerinden gelen belirtileri yapay sinir ağları yardımıyla analiz ederek hastanın o anki durumunu belirler. Hasta durumu tanımlandıktan sonra belirlenen duruma göre Doktor Etmeni ile iletişime geçer.

Doktor Etmeni: Herbir doktorun kendine ait etmeni vardır ve Hasta Etmeninden gelen bilgiler ve etmen yardımıyla bir sonraki adıma karar verir (Alarm, randdevu, yönlendirme vb.).

3.1.3 Koutkias et al.

Bu çalışma, genel olarak Sağlık Merkezlerinin (Medical Contact Center) gözlem, takip ve eğitim hizmetlerini geliştirmek amacıyla bir çoklu etmen sistemi önermektedir (Koutkias et al. 2005). Hastaların ev ortamında geçirdikleri zamanda üretilen çok miktarda veriler ve sağlık kuruluşları ile etkileşimlerini verimli bir şekilde yönetmek ihtiyacını esas almaktadır. Yazılım etmenleri bu tür durumlarda gerçek zamanlı veri işleme görevlerine uygunluğu ve değişken ortamlarda gösterdikleri işbirliği yetenekleri nedeniyle tercih edilmiştir.

Sistem özet olarak tıp merkezlerinin çalışma ortamını takip eder ve önemli durumları tespit ederek ilgili yönetici personeli alarmlar, bildirim mesajları ve öneriler ile yönlendirme işlevini görmektedir. Bu amaçla sistem sürekli olarak hastayı ve tıp merkezinde yapılan işlemleri takip etmektedir. Etmenlerin ana görevi gelen verileri değerlendirmek ve kural tabanlı bilgiye dayanarak proaktif bir biçimde olağandışlıkları tespit etmektir. Sistemde aşağıdaki diyagramda görüldüğü gibi on farklı etmen bulunmaktadır. Bu etmenler izleme ve gösterim işlevlerini yürütmektedirler.

Problem Tespit Etmeni (Problem Indication Agent): Bu etmen ontolojiler yardımıyla sınır değerleri aşan tıbbi verileri ve diğer belirtileri tanımlamaktadır.

Genel Değerlendirme Etmeni (Overall Assessment Agent): Diğer etmenlerden gelen bilgileri değerlendirerek hastanın genel durumunu tanımlar.

Eğitim Etmeni (Educational Agent): Hastanın o anki durumu ve eğitim planına göre hastaya tavsiyeler ve bilgilendirmelerde bulunur.

Planlama Etmeni (Scheduler Agent): Hastaların randevu zamanlarını ve tıp merkezi ile iletişimi yönetmektedir.

Yönetici Etmen (Administrator Agent): Kullanıcıların sistemde toplanan verilere erişimini kontrol etmektedir.

Erişim Etmeni (Contact Agent): Kullanıcılar ve sistem arasında aracı olarak işlev görmektedir.

Arayüz Etmenleri: Görsel arayüzleri yönetmektedir.

Doktor Etmenleri: Disease Agent ve Disease GUI Agent etmenleri doktorların ilgili belirtilen hastalık ile ilgili neler yapıldığını takip etmesini sağlamaktadır.

3.1.4 SAPHIRE

SAPHIRE sağlık kuruluşları ve sağlık çalışanlarının yükünü azaltmak amacıyla tasarlanmış bir klinik karar destek sistemidir (Doğaç et al. 2008). Gereksinim duyulan uzman bilgisi için Klinik Kurallarına başvurulmuştur. Hedef alınan problemin doğası heterojen ve dağıtık kaynaklarla ilgilenen otonom varlıklar gerektirdiği için çözüm olarak bir çoklu etmen sistemi önerilmiştir. Çeşitlilik gösteren tedavi ortamlarının kaynaklarını kullanabilmek için semantik bir altyapı düşünülmüştür.

SAPHIRE sisteminin bileşenleri:

Agent Factory Agent: Bu etmenin temel görevi hastaya uygun Klinik Kurallarını belirlemek ve hastaya özel bir Guideline Agent başlatmaktır. Görevini yerine getirmek için sensör, medikal web servisleri ve hasta kayıtlarını bir araya getirerek Klinik Kurallarının uygulamaya hazır olmasını sağlar.

EHR Agent: Sistemin elektronik hasta kayıtlarına erişimini sağlayan bir arayüz olarak çalışmaktadır. Bu arayüz aracılığı ile diğer etmenler ve hasta kayıtlarına erişim işlemleri birbirinden soyutlanmaktadır. SAPHIRE mimarisinde temel hasta versisi paylaşımı IHE XDS Registry/Repository mimarisi ile gerçekleştirilmektedir. EHR etmeni bu tip veri depoları ile iletişim yapabilme imkanlarına sahiptir.

Ontology Agent: SAPHIRE mimari tasarımı farklı sağlık kuruluşlarının kaynaklarına erişim sırasında ortaya çıkan semantik uyum problemlerini aşmaya yönelik olarak planlanmıştır. SAPHIRE sisteminde hasta verilerinin referansları HL7 RIM modeli esas alınarak modellenmiştir. Tıbbi web servislerinin ve sensör verilerinin değişik veri yapıları ve terminolojiler kullanması mümkündür. Ontoloji etmeni ile bu birlikte çalışabilirlik sorunu çözümlenmiştir.

Guideline Agent: Hastaya özel Klinik Kurallarını işleten bileşendir. İlgili kural tanımlamalarını işleyerek kurallarda belirtilen gerekli görevleri çalıştırır. Kural etmeni birkaç modüler araç kullanarak görevini yerine getirir.

Monitoring Agent: Klinik Kuralları çalıştırılırken o anki işlem durumu belirli bir Monitoring Agent etmenine bildirilir. Bu etmen sağlık çalışanlarına işlemleri görüntüleme ve durumu izleme imkanı vermektedir.

Alarm Distribution Agent: Klinik Kurallarının işletilmesi esnasında bildirim, alarm ve hatırlatmalar yapma gereksinimi ortaya çıkmaktadır. Bu tür durumlarda ilgili mesaj alarm dağıtım etmenine gönderilir. Bu etmen ise mesajların ilgili yerlere güvenilir ve etkin bir biçimde aktarılmasını sağlar.

3.1.5 HeCaSe2

HeCASE, bir şehrin vatandaşlarına veya ziyaretçilerine kişiselleştirilmiş tıbbi hizmetler sağlamak için tasarlanmış bir etmen tabanlı sistemdir (Isern et al. 2007).Sistemde tıbbi merkezlerin ve doktorların bilgilerini içeren ve yöneten etmenler bulunmaktadır. Bu etmenler, tıbbi kayıtlarına ulaşmak isteyen, bir doktor veya hastane arayan ya da randevu almak isteyen sistem kullanıcılarına bir takım hizmetler sunmak için kendi aralarında koordine olarak çalışırlar.

Sistemin mimari bileşenleri incelendiğinde şu yapılar görülmektedir:

- i. Kullanıcının kişisel etmeni, kullanıcının sistemi kullanabilmesi için gerekli arayüzleri sağlar.
- ii. Aracı etmen, kullanıcı etmeni ile diğer sistem etmenleri arasındaki iletişim için aracı olur.
- iii. Tıbbi merkezler için üç seviyeli bir alt sistem mevcuttur. Birinci seviyede Tıbbi Merkez Etmeni (MCA) bulunur. İkinci seviyede Departman Etmenleri (DEPs) ve son seviyede ise Doktor Etmenleri (DA) bulunur. MCA merkeze ait genel bilgileri içermektedir. DEPs ilgili bölümün bilgilerini içermektedir ve DA'lar ise doktorların randevu saatlerini ve kesinleşmiş hasta ziyaretlerinin bilgilerini içermektedir. Ayrıca doktorlar için jandırlanmış bir web arayüzü ile doktor hasta bilgilerini görebilir ve takip edebilir.
- iv. Veritabanı yönetici etmeni, tıbbi kayıtların yer aldığı ve veritabanı erişimini kontrol eden etmendir.

3.2 Çoklu Etmen Platformları Karşılaştırmalı Analizi

Kullanılacak platformun standartlara uygun olması gereklidir. Standartlar ile kalite, güvenilirlik, verimlilik ve maliyet gibi gereksinimleri sağlamak mümkün olacaktır. Etmen teknolojilerinde standart üreten ana iki kurum bulunmaktadır: FIPA (Foundation for Physical Agents) ve OMG (Object Management Group). Etmen teknolojilerinin yeni bir alan olduğu düşünüldüğünde daha değişken alanlarda uygulanan eski standartın daha çok kabul görmesi daha olasıdır.

FIPA IEEE Computer Society standartlar komitesinin bir üyesidir. FIPA standartlarının amacı etmen teknolojisinin kullanımını arttırmak ve diğer teknolojiler ile entegrasyonunu ilerletmektir. FIPA spesifikasyonlarını standart haline getiren bir süreci 2002 yılında tamamlamıştır. Bu spesifikasyonlar etmen teknolojisinin farklı özelliklerini tanımlamaktadır;

- i. Etmen İletişimi
- ii. Etmen Yönetimi

- iii. Etmen Mesajı Taşıma
- iv. Soyut Etmen Mimarisi
- v. Etmen Uygulamaları

OMG 2000 yılında yayınladığı MAF (Multi Agent Facility) dokümanında FIPA tarafından belirtilen etmen yönetimi, soyut mimari konularını içermektedir. OMG FIPA yaklaşımından farklı olarak daha somut tasarlanmıştır, örneğin CORBA (Common Object Request Broker) tekniğinin kullanılmasını önermektedir. OMG MAF standardı 2000 yılından sonra herhangi bir gelişme göstermemiştir ve görece eski kalmıştır. Ancak etmen platformlarını karşılaştırırken MAF uyumunu bir artı değer olarak görmek mümkündür.

Bu çalışma kapsamında FIPA standardı temel alınacaktır. FIPA resmi sitesinde uyumlu platformların bir listesini vermektedir (www.fipa.org/resources/livesystems.html (Erişim tarihi: Eylül 2014));

- i. Agent Development Kit
- ii. April Agent Platform
- iii. Comtec Agent Platform
- iv. FIPA-OS
- v. Grasshopper
- vi. JACK Intelligent Agents (Howden et al. 2001)
- vii. JADE
- viii. JAS Java Agent Services API
- ix. LEAP
- x. Zeus

FIPA resmi sitesinde yayınlanmayan ancak FIPA standartını destekleyen diğer bazı platformlar;

- i. JADEX Agents (Pokahr et al. 2003)
- ii. The SPADE Multiagent and Organizations Platform

Kullanılacak etmen altyapılarının platform bağımsız olması dağıtık sistemlerin pratik ve mimari gereksinimleri açısından önemlidir. Etmenlerin ve dolayısı ile etmen platformlarının çeşitli işletim sistemleri ve mobil cihazlarda çalışabilmesi gereklidir. Bu çalışma kapsamında mobil cihazlar, tıbbi cihazlar ve kurum sunucularında çalışabilmesi özellikle önemlidir. Çizelge 3-1 den Çizelge 3-5 e kadar platformların çeşitli kriterlere göre karşılaştırmalı incelenmesi görülmektedir.

Çizelge 3-1 Etmen Platformları ve İşletim Sistemleri

Etmen Platformu	Windows	Linux	Android	iOS	OSX
JACK Intelligent Agents	Evet	Evet	Evet	—	Evet
JADEX Agents	Evet	Evet	Evet	—	Evet
JADE	Evet	Evet	Evet	—	Evet
Agent Development Kit	Evet	Evet	Evet	—	Evet
April Agent Platform	—	Evet	—	—	—
Comtec Agent Platform	—	—	—	—	—
FIPA OS	Evet	Evet	—	—	Evet
The SPADE Multiagent and Organizations Platform	Evet	Evet	—	—	—
Grasshopper	—	—	—	—	—
Zeus	Evet	Evet	—	—	Evet
LEAP	—	—	—	—	—
JAS Java Agent Services API	Evet	Evet	—	—	Evet

Sağlık alanında geliştirilen çözümlerin sürekli bir değişim içinde olacağı düşünüldüğünde kullanılacak platformun hızlı ve etkin geliştirmeye imkan vermesi gereklidir. Bu amaçla platformu hedefleyen geliştirme araçlarının mevcut olması ve yeterli olması gerekmektedir.

Çizelge 3-2 Etmen Platformları ve Metodolojiler, Güncel Geliştirme Araçları

Etmen Platformu	Ana Metodoloji ve Geliştirme Aracı
JACK Intelligent Agents	Prometheus, PDT Prometheus Design Tool
JADEX Agents	Tropos, TAOM4E
JADE	Gaia, MaDKit / O-MaSE, agentTool III
Agent Development Kit	—
April Agent Platform	—
Comtec Agent Platform	—
FIPA OS	—
The SPADE Multiagent and Organizations Platform	—
Grasshopper	—
Zeus	—
LEAP	—
JAS Java Agent Services API	—

Platformun teknik desteğinin ve gelişiminin devam etmesi bu çalışma konusunda olduğu gibi ciddi uygulamaların en önemli önşartlarından biridir.

Çizelge 3-3 Etmen Platformları, Teknik Destek ve Geliştirme

Etmen Platformu	Teknik Destek ve Geliştirme
JACK Intelligent Agents	Ticari olarak teknik destek var ve güncelleştirmeler devam ediyor.
JADEx Agents	Teknik destek var ve güncelleştirmeler devam ediyor.
JADE	Teknik destek var ve güncelleştirmeler devam ediyor.
Agent Development Kit	Teknik destek var ve güncelleştirmeler devam ediyor.
April Agent Platform	Teknik desteğe erişim yok, son güncelleme 2013 yılında.
Comtec Agent Platform	Erişim yok.
FIPA OS	Erişim yok, son güncelleme 2001 yılında.
The SPADE Multiagent and Organizations Platform	Teknik yardım sadece dokümanlarla sağlanıyor, son güncelleme 2012 yılında.
Grasshopper	Erişim yok.
Zeus	Erişim yok, son güncelleme 2013 yılında.
LEAP	Erişim yok.
JAS Java Agent Services API	Proje 2011 de sonlandırılmış.

Platformun araştırma grubu aktif olarak çalışmalarına devam ediyor olması oluşturulacak çözümlerin geleceği için önemlidir.

Çizelge 3-4 Etmen Platformları ve ArGe Grubu Çalışmaları

Etmen Platformu	ArGe Çalışmaları
JACK Intelligent Agents	Evet ve ticari olarak geliştirilmeye devam ediliyor. Son çalışma 2014 yılında.
JADEX Agents	Evet ve açık kaynak projesi olarak geliştirilmeye devam ediliyor. Son çalışma 2014 yılında.
JADE	Evet ve açık kaynak projesi olarak geliştirilmeye devam ediliyor. Son çalışma 2014 yılında.
Agent Development Kit	Ticari olarak geliştirilmeye devam ediliyor.
April Agent Platform	Erişim yok.
Comtec Agent Platform	Erişim yok.
FIPA OS	Son çalışma 2002 yılında.
The SPADE Multiagent and Organizations Platform	Evet ve açık kaynak projesi olarak geliştirilmeye devam ediliyor. Son çalışma 2013 yılında.
Grasshopper	Son çalışma 2000 yılında.
Zeus	Son çalışma 2011 yılında.
LEAP	Son çalışma 2002 yılında.
JAS Java Agent Services API	Erişim yok.

Son olarak platformun lisans açısından değerlendirilmesi gereklidir. Açık kaynak kodlu sistemler sağlık alanı gibi çok katılımcının olduğu alanlarda daha önemlidir.

Çizelge 3-5 Etmen Platformları ve Kullanım Lisansları

Etmen Platformu	Lisans
JACK Intelligent Agents	Ticari ve Akademik Lisans
JADEx Agents	GNU Genel Kamu Lisansı
JADE	GNU Kısıtlı Genel Kamu Lisansı
Agent Development Kit	Ticari Lisans
April Agent Platform	GNU Genel Kamu Lisansı
Comtec Agent Platform	GNU Genel Kamu Lisansı
FIPA OS	Sonera Kamu Lisansı, Nortel Kamu Lisansı, Emorphia Kamu Lisansı
The SPADE Multiagent and Organizations Platform	GNU Kısıtlı Genel Kamu Lisansı
Grasshopper	Erişim yok.
Zeus	Zeus Açık Kaynak Lisansı
LEAP	GNU Kısıtlı Genel Kamu Lisansı
JAS Java Agent Services API	Erişim yok.

3.3 Etmen Platformu Seçimi

Yapılan incelemeler ve karşılaştırmalardan sonra sağlık bilişimi alanında uygulanabilir nitelikte altyapılardan JACK Intelligent Agents ve JADEX Agents platformları ön plana çıkmaktadır. JACK Intelligent Agents ticari uygulama alanı bulmuş olması ve yine ticari ve kaliteli teknik destek imkanı ile öne çıkmaktadır.

JADEX Agents platformu açık kaynak olması, sürekli geliştiriliyor olması ve araştırmacılarının aktifliği ile kayda değer bir imkan sunmaktadır.

Yapılan incelemede JACK Intelligent Agents gerçek ortamda çalışacak bir sağlık sistemi için daha uygun görülse de, ticari olması nedeniyle erişimi kısıtlıdır. Eğitimi için gerekli kaynaklara ulaşım JADEX ile karşılaştırıldığında daha maliyetli ulaşılması zordur. Bu bilimsel çalışmada altyapı dosyalarına erişim kolaylığı, dokümantasyon ve eğtici materyale erişim kolaylığı nedeniyle JADEX Agents altyapısı kullanılacaktır.

3.4 Ontoloji Altyapısı

Bu bölümde kullanılacak ontoloji dilleri ve ontolojik bilgileri saklamak amacıyla kullanılacak veritabanları incelenecektir.

3.4.1 Ontoloji Dilleri

Ontoloji oluşturmak için birçok farklı dil ve sentaks bulunmaktadır. Bu çalışma kapsamında RDF (Resource Definition Framework) sentaksı tabanlı OWL (Web Ontology Language) kullanılacaktır. RDF'in amacı kaynaklar (resources) denilen tanımlanmış varlıklar için bir çerçeve (Framework) denilen bir yapı sağlamaktır.

RDF'te anlamlı bir işlem yapabilmek için konu edilen varlık tekil bir biçimde bir evrensel tekil tanımlayıcı (URI - Universal Resource Identifier) ile tanımlanmalıdır. Örneğin “<http://www.roatis.net/Projects#CureMap>” gibi. RDF üç temel elemandan oluşmaktadır (Klyne et al. 2004). Bunlar;

Kaynaklar (Resources): Tanımlanan varlıklar

Özellikler (Properties): Varlıklar arasındaki ilişkiler

Sınıflar (Classes): Varlıkları gruplayan tanımlar.

Bu elemanlar birleştirilerek üçlü denilen önermeler oluşturulur. Örneğin;

<EUTFHastanesi> <varBolum> <AcilServis>

<<http://www.roatis.net/Projects#EUTFHastanesi>>

<dc:varBolum>

<<http://www.roatis.net/Projects#AcilServis>>

RDF’te hazır halde bulunan bazı özellikler, tür (type), alt sınıf (subClassOf), alt özellik (subPropertyOf), etkilenen tür (range), tanım alanı (domain), insanlar tarafından okunabilir isim (label), okunabilir yorum (comment) olarak sıralanabilir (Prud’hommeaux et al. 2007).

OWL dili varlıkların tanımlanarak sınıflandırılması dışında bunların arasındaki ilişkilerin de tanımlanması ve sınıflandırılmasını içerir. OWL RDF’te bahsi geçen temel tanımların yanısıra fazladan temel elemanlara sahiptir;

Sınıflar arasındaki ilişkiler. Örn: farklıdır (disJointWith)

Eşitlik. Örn: aynısıdır (sameAs)

Daha zengin özellikler. Örn: geçişli (transitive)

Kısıtlamalar. Örn: bu değerden itibaren tüm değerler (allValuesFrom)

Buradan da görüldüğü gibi OWL daha geniş kapsamlıdır ve mantıksal çıkarsamalar yapmaya izin veren bir yapıdadır.

3.4.2 Ontoloji Veritabanı

Ontoloji veritabanı olarak birçok seçenek bulunmaktadır. Bunlardan öne çıkanlar StarDog ve Oracle Spatial and Graph yazılımlarıdır. StarDog saf Java uygulaması olması itibarıyla mobil platformlarda kullanım için daha uygun

görülmektedir ancak her platformda ve büyüklükte kullanılabilir. Merkezi kayıt için ise kendini kanıtlamış bir veritabanı yazılımı olan Oracle üzerinde çalışan ve JENA apisi ile uyumlu olan Oracle Spatial and Graph daha elverişli görülmektedir.

3.4.3 JENA

Ontolojiler üzerinde çalışmak amacıyla JENA uygulama arabirimi kullanılacaktır. Semantik web uygulamaları geliştirme sırasında kullanılmak üzere tasarlanmış Java tabanlı bir program çerçevesidir ve ontolojiler üzerinde SPARQL sorgularını çalıştırmaya izin verir.

3.5 Web Servis Altyapısı

Bu çalışma kapsamında kullanılacak teknoloji ve altyapılar belirlenirken birinci öncelik birlikte çalışabilirlik özelliğidir. Web servislerin kullanılacağı noktalar ise mobil cihazlar ile merkezi sistemler ve kuruluşlarda halihazırda bulunan kurulu sistemlere entegrasyon olacaktır.

Bu durumda mobil cihazlar ve işletim sistemlerinin kısıtlarına en uygun, kurumsal kullanıcıların sistemleri ile veri alış verişi için etmen sistemleri ile en sorunsuz çalışacak web servis teknolojisinin seçilmesi gereklidir.

Bu kapsamda JAX-WS, RESTful web servisleri ve .Net tabanlı WCF servisleri incelenecektir.

3.5.1 JAX WS

Java EE teknolojisinde JAX WS, SOAP tabanlı web servisleri için gerekli fonksiyonları sağlamaktadır. Bu web servisleri SOAP standartına uyumlu XML mesajları kullanırlar. SOAP mesaj yapılarını ve mesaj mimarilerini belirleyen XML tabanlı bir dildir. Bu servisler WSDL denilen makine tarafından anlaşılabilir ve servisin sunduğu işlevlerin tanımlandığı dosyalar içerir.

JAX WS ile bir servis yazılacağı zaman öncelikle operasyonları, mesajları ve servisin lokasyonunu belirten bir WSDL dokümanı hazırlanmalıdır. Ancak bu doküman olmadan da bir JAX WS servisi SOAP mesajlarını işleyebilir.

JAX WS kullanmanın diğerk bir avantajı ise kendini kanıtlamış diğerk Java teknolojileri ile birlikte kullanılabilmesidir. Örneğink EJB ile iş zekası ayrılabilir ve ölçeklenebilir servisler geliştirilebilir, JMS ile daha esnek iletişim kurabilen servis gerisi iş akışı düzenlenebilir, JPA ve Hibernate ile veritabanı işlemleri kolayca yapılabilir. Bunların dışında bu servislerin üzerinde çalışacağı sunucular anlamında da çok geniş seçenekler mevcuttur. Ücretsiz ve açık kaynak kodlu Apache Tomcat gibi geniş kullanımlı, olgunlaşmış bir sunucu yazılımından Glassfish, JBoss ve Oracle Application Server kurumsal anlamda desteklenen orta ve büyük projelerde kullanılabilecek sunucu yazılımlarında bulunmaktadır.

3.5.2 RESTful Web Servisleri

Representatioanl State Transfer web servisleri temel veya amaca yönelik entegrasyon gereksinimleri için daha uygun olan bir servis çeşididir. RESTful servisler SOAP standartından ziyade HTTP standartına uygundur ve WSDL tanımlamaları ile XML mesajları gerektirmez.

RESTful web servisleri W3C ve IETF standartlarını (HTTP, XML, URI, MIME vs) kullanması ve yazımı ve istemci tarafından kullanılması için çok az düzeyde araç gereksinimi olması nedeniyle çok kullanışlıdır. Bu özellik ve SOAP gibi XML kısımları olmadığı için bant genişliği konusunda da iyi olması bu servisleri mobil cihazlar için iyi bir seçenek haline getirmektedir.

Hazırda bulunan servisler kolayca RESTful servislere taşınabilirler ve böylece uygulamalar yeniden yazılmadan AJAX ve http ile bu servisleri kullanabilir.

RESTful web servisleri birçok farklı dil ve teknoloji ile yazılabilmektedir. Öne çıkan teknolojilerden Java EE içeriğinde JAX-RS kütüphanesi ile gerekli fonksiyonları sağlamaktadır.

3.5.3 WCF Servisleri

WCF (Windows Communication Foundation) Microsoft Windows için geliştirilmiş bir servis odaklı iletişim platformudur. SOAP kullandığı için diğerk teknolojiler ile hazırlanan web servisleri ile uyumlu çalışabilmektedir.

WCF de üç temel kavram bulunmaktadır;

Address: Web servisin URL adresi

Binding: İletişim şeklinin tanımı, kullanılacak protokoller (HTTP, TCP, UDP) ve mesaj biçimi (Binary, SOAP, özel format)

Contract: Web servis ile ilgili bilgilerin bir Interface olarak hazırlanması. Bu Interface de servisin sunduğu operasyonlar bulunmaktadır.

WCF yine Microsoft firmasına ait olan IIS web sunucusunda çalışabileceği gibi bir .Net uygulaması tarafından da yayınlanabilmektedir.

3.6 Mobil Uygulama Altyapıları

Projeyi gerçekleştirilebilir kılan en önemli unsur mobil cihazlarda yaşanan gelişmelerdir. 2000'li yılların başında PDA lar ile uygulanan projeler cihazlardaki teknik kısıtlardan ve altyapı yetersizliğinden dolayı uygulanabilirliği çok sınırlı kalmıştır. Bugünlerde ise cihazlar ve işletim altyapıları performans ve yeni özellikler ile hemen hemen her yıl güncellenerek oldukça iyi bir seviyeye gelmiştir. Bu bölümde projenin hastalar ve doktorlar tarafından kullanılan bileşenlerinin çalışacağı yaygın olarak kullanılan mobil akıllı telefonlar ile işletim sistemleri incelenecektir.

3.6.1 Android

Android bir işletim sistemi, akıllı telefonlar ve tablet bilgisayarlar için geliştirilmiş bir yazılım platformudur. Google firması tarafından kurulmuş olan Open Handset Alliance (87 firmanın dahil olduğu bir konsorsiyum HTC, Motorola vb.) tarafından telif hakkı olmayan bir işletim sistemi fikriyle meydana getirilmiştir.

Android Linux tabanlı ve açık kaynak kodludur. 2013 rakamlarına göre kullanım oranı yaklaşık 187 milyon kullanıcı ile %79 civarındadır. Üzerinde Java tabanlı Dalvik sanal makinesi çalışmaktadır ve android için özel Java kütüphaneleri mevcuttur.

Android uygulamaları genellikle Java dili ile yazılmakla birlikte C ve C++ dilleri ile yerli uygulamalar yazmak mümkündür ve bu diller içinde uygulama kütüphaneleri vardır.

Güçlü bir web tarayıcısı, SQLite gibi bir veritabanı ve OpenGL desteği ile birçok uygulama geliştirmenin zeminini oluşturmaktadır. Geliştirme araçları ve kütüphanelerin ücretsiz olması ve iyi dokümente edilmiş olduğu için yazılım geliştirmek oldukça kolaylaşmıştır.

Android işletim sistemi kurulu cihazlara erişim ise benzerlerine göre birçok fiyatta ve segmentte bulunabildiğinden dolayı daha kolaydır. Gelir oranı nispeten daha az olan kullanıcılara yönelik yazılımlarda öne çıkmaktadır.

3.6.2 iOS

Apple firması tarafından yine kendi ürettiği iPhone, iPad ve diğer cihazlarında kullanmak için geliştirdiği MacOSX tabanlı bir işletim sistemidir. Geliştirilmeye 2005 yılında başlanmıştır ve Apple dışında hiçbir firmaya kullanım lisansı verilmemektedir.

Yazılım geliştirmek için iOS SDK adında bir geliştirme platformu bulunmaktadır ve uygulama geliştirmek için Objective C dili kullanılmaktadır. Geliştirme araçları MacOS bilgisayarlarda ücretsiz olmakla beraber diğer işletim sistemlerinde kullanım için belirli bir ücret talep edilmektedir.

Bahsedilen özellikler bazı kısıtlamalar getirmesine karşın Apple cihazlarının ileri teknoloji güçlü özellikleri ve cihazların az sayıda ve iyi tanımlanmış olması nedeniyle yüzlerce farklı teknoloji ve ekran çözünürlüğünde yapılmış olan Android işletim sistemleri cihazlara göre daha etkin kullanıcı arayüzleri ve uygulamalar yazmaya imkan vermektedir.

iOS pazardaki %14 pay ile ikinci en yaygın işletim sistemidir dolayısıyla gözardı edilemez kullanıcı sayısı nedeniyle mobil projelerde dikkate alınması gerekmektedir.

3.6.3 Windows Mobile

Microsoft firması tarafından akıllı telefonlar için geliştirilmiştir ve piyasadaki önemli akıllı telefon üreticileri tarafından kullanılmaktadır. Kullanımı son yıllarda büyük bir artış göstermektedir ve 2013 rakamları ile %3,6' ya ulaşarak üçüncü en yaygın mobil işletim sistemi olmuştur.

Uygulama geliştirme işlemi Visual Studio geliştirme aracı üzerinden yapılmaktadır ve bu kullanılan araçların ücretsiz versiyonları da mevcuttur. Platform olarak .Net kullanması nedeniyle .Net tabanlı diller (C#, Visual Basic, Visual C++, F# vb.) ile uygulama geliştirilebilmektedir. Platformun birçok özelliğini de kullanabiliyor olması nedeniyle uygulama geliştirme sırasında çok geniş seçenekler sunabilmektedir.

3.7 Sağlık Temalı Mobil Uygulamalar

Sağlık alanında geliştirilen birçok mobil uygulamaya her geçen gün yenileri eklenmektedir. Genel olarak bakıldığında zaman uygulamalar beslenme ve diyet, egzersiz, diyabet, süte veren annelere ilaçların emzirmeye etkilerini gösteren uygulama benzeri bilgilendirme uygulamaları ön plana çıkmaktadır. Hematoloji alanında geliştirilen uygulamalara bakıldığında ise hemofili takibi amacıyla oluşturulan uygulamalar ön plana çıkmaktadır.

Oluşturulan uygulamalar çok büyük oranda sadece kurulu oldukları mobil cihaz üzerine kayıt tutan ve herhangi bir merkezi sistem ile ilgisi bulunmayan uygulamalardır. Birçok uygulama kayıtlı verilerin mail gönderimi yoluyla paylaşılmasına imkân vermektedir. Ayrıca verilerin detaylı incelenmesi için çeşitli grafik gösterimler ve öneri sistemleri de mevcuttur.

3.8 Güvenlik

Sağlık bilişimi sistemleri başta olmak üzere kapsamlı ve sofistike dağıtık yazılım sistemlerinde güvenlik en önemli konulardan biridir. Yazılım sistemlerinin güvenliği tanımlanırken bazı özelliklerin olup olmadığına bakılmaktadır, bunlar:

- Gizlilik
- Kimlik denetimi ve doğrulama
- Sistem bütünlüğü
- Çevrimiçi olma oranı
- Erişim kontrolü

- Veri doğrulanabilirliği, (verinin dijital olarak imzalanması gibi)
- Muhtemel tehditler ile başa çıkabilme kabiliyeti

Güvenlik, performans ve sağlamlık gibi sistemin fonksiyonel olmayan hedeflerinden biridir. Daha önce de bahsedildiği üzere fonksiyonel olmayan hedefler sistem için bir kalite özelliği getirmekler beraber sistemin geneli için de bazı kısıtları ortaya koyar.

Bu nedenlerden dolayı bu çalışma kapsamında güvenlik iki ayrı kategoride ele alınmaktadır. Bu kategorilerin ilki erişim kontrolüdür ve son kullanıcıların sisteme erişiminin kontrol altında tutulmasını hedeflemektedir. Bu amaçla bir sanal özel ağ önerilmektedir. Bu öneride son kullanıcılar sisteme bağlanmadan önce bu SSL şifrelemesi ile korunan sanal özel ağa bağlanması gerekecektir. Böylece kullanıcıların ağ ile iletişimi güvenli hale getirilecek ve de erişim kontrolü sağlanmış olacaktır.

Diğer güvenlik kategorisinde ise MASHemato sistemi içinde tanımlanacak alt sistemler tarafından yönetilen gizlilik, kimlik doğrulama, sistem bütünlüğü ve veri doğrulaması gereksinimleri karşılanmış olacaktır.

Sistem mimarisi içinde tanımlanacak olan Giriş ve Doğrulama Sistemi, kurum ve hasta sistemlerinin kimlik doğrulama ve veri doğrulama işlemlerini yürütecektir. Bir başka alt sistem olan Yönetim Sistemi, sistemin bütünlüğünü takip ederek yeni kullanıcı sistemlerinin oluşturulması, yeniden başlatılması ve sonlandırılması işlemlerini üstlenecektir. Kullanıcı ve kurum sistemleri oluşturulur iken sağlık bilişimi başlığı altında bahsedilen güvenlik standartlarına uyum sağlamak amaçlanacaktır.

Bu çalışma kapsamında ele alınmamış olan ancak güvenlik gereksinimlerinin tamamen karşılanması için gerekli olan diğer bir konu ise muhtemel saldırılara karşı savunma mekanizmaları gerekliliğidir. Sisteme karşı yapılması muhtemel saldırılardan bazıları virüsler, DoS (Denial of Service), kriptografik saldırılar ve dinleme saldırılarıdır.

3.8.1 VPN

Sanal özel ağlar (VPN), hâlihazırda var olan fiziksel ağ altyapısını kullanan, veri ve IP erişimi için güvenli bir iletişim mekanizması sağlayan özel bir ağıdır. İnternet gibi var olan ağlar üzerine kurulabilmesi ile kamuya açık ağlar üzerinden önemli ve hassas verilerin güvenli olarak taşınmasını sağlar.

VPN, kurumlar ve kullanıcı bölümleri arasında ayrılmış özel iletişim hatları gibi alternatiflerine göre daha az maliyetlidir. VPN istemcilerin nerede olduğundan bağımsız olarak sunucular ile güvenli iletişim imkanı sağlamaktadır. Bu çalışmada da sistem kullanıcıları arasından hastalar ev, iş, hastane, şehirdışı gibi herhangi bir yerde olabilir ve benzer şekilde ilk yardım ekipleri, doktorlar ve sağlık kuruluşları da herhangi bir yere bağlı kalmaksızın çalışabilirler. Bu durumda hasta verilerinin transferi ve sisteme erişim güvenliği bir sanal özel ağ yardımıyla sağlanabileceği anlaşılmaktadır.

Sanal özel ağlarda güvenlik SSL, IPSec ve PPTP yöntemleri ile sağlanmaktadır. IPSec protokolü işletim sistemi çekirdeğinde bulunan IP yığnında değişiklik öngördüğünden her işletim sistemi için ayrı bir gerçekleştirim gerektirebilir. PPTP protokolü Windows işletim sistemlerinde hazır olarak gelmektedir. Ancak bu çalışma hem çeşitli işletim sistemleri üzerinde hem de mobil olarak çalışmasını gerektirdiği için önemli bir avantaj sağlamamaktadır. SSL, internet üzerinden güvenli veri transferi için 1994 yılında Netscape firması tarafından geliştirilmiş bir protokoldür. Günümüze kadar geliştirilmiş ve e-ticaret, kurumlar arası özel ağlar ve askeri alanda yaygın olarak kullanılmaktadır. SSL çok sayıda ve çeşitli konumlardaki kullanıcılar için en yaygın olarak kullanılan çözümdür ve bu yönüyle çalışma gereksinimini karşılamaktadır.

SSL VPN veri güvenliğini büyük ölçüde sağlamakla beraber riski tamamen ortadan kaldırmamaktadır. Ancak bu çalışmada önerilen sistemde hedeflenen kullanıcı sayısı, bağlantı konumlarının çeşitliliği ve işletim sistemi altyapılarının çeşitliliği göz önüne alındığında ve uygulama yaygınlığıyla sistemin güvenlik gereksinimini karşılamaya en uygun yöntemlerden biridir.

3.9 Genel Senaryo

Bu bölümde çalışma kapsamında önerilen sistemin uygulamaya girmesi durumunda gerçekleşecek senaryolar anlatılmaktadır. Bu senaryolar sistemin

gereksinimlerinin belirlenmesi ve sistemin mimari tasarımı esnasında göz önüne alınmıştır.

Bu çalışma kapsamında önerilen sistem bir hasta takibi sistemi olduğundan sistemin katılımcıları hasta, hasta yakını, hasta bakıcısı, tüm sağlık kuruluşları, hasta ile temas kuran tüm sağlık çalışanları ve dolaylı olarak katılımcı olan sosyal güvenlik kurumu ve sağlık bakanlığı olarak listelenebilir. Sağlık kuruluşları olarak sağlık ocağı, hastaneler, klinikler, diş klinikleri, göz klinikleri, fizik tedavi merkezleri, onkoloji merkezleri ve özel muayenehaneler sayılabilir. Uzman sağlık çalışanları çeşitli uzmanlık alanlarında doktorlar, fizik tedavi uzmanları olarak sıralanabilir. Yardımcı sağlık çalışanları ise gerek hastane içinde gerekse hastaya evinde bakım sağlayan hemşireler, doktor asistanları olarak belirtilebilir. Bu çalışmada sağlık bakanlığı, sosyal güvenlik kurumu, ilaç üreten firmalar, sağlık alanında arge yapan ve diğer kamu sağlığı ile ilgili olan ve gerçek zamanlı rapor talebinde bulunabilecek kuruluşlar Rapor İstemcisi Kuruluşlar olarak adlandırılmıştır.

3.9.1 Hasta Senaryosu

Hastanın yapması gerekenler, belirtilere dikkat etmek, verilen ilaç, egzersiz ve perhizleri dikkatle uygulamak, doktor tarafından talep edilen bilgileri kayıt etmek ve acil bir durum olduğunda önemli bilgileri paylaşmaktır. Bu amaçla senaryo hastanın mobil cihazında sistemi aktive etmesi ile başlar. Mobil cihaz eğer varsa tıbbi cihazlar ile bağlantılarını kurar ve merkez ile iletişime geçer. Hasta mobil cihazında bildirimleri, hatırlatmaları ve uyarıları takip eder. Ana ekranında randevuları, alması gereken ilaçları, bildirimleri, alarmları, doktordan gelen formları, mesajları, son yapılan işlemleri, tetkik sonuçları gibi önemli bilgileri listeler.

Hasta günlük yaşamında doktoru tarafından kendisine atanan günlük görevleri yapar, bunlar günlük tutmak, aldığı ilaçları kayıt etmek, beklenmedik değişimleri kayıt etmek olabilir. Ayrıca eğer var ise hasta eğitimi için gerekli bilgileri, oluşması muhtemel yan etkileri, dikkat edilmesi gereken konuları görür.

Bunların dışında hasta acil durumlarda yakınlarına, bakıcısına, doktoruna yardım çağrısında bulunabilir veya durumunu paylaşabilir. Acil yardım ekibinin

görebilmesi için acil durum bilgileri özel bir ekranda bulunabilir veya cihazdan cihaza paylaşılabılır.

3.9.2 Hasta Yakını ve Hasta Bakıcısı Senaryosu

Hasta yakını veya bakıcısı, hastanın çocuk, yaşlı veya kendi işlerini göremez durumda olabileceği düşünülerek hastanın yapacağı tüm işlemleri yapabilmelidir. Ek olarak hastadan veya doktorundan gelen bildirimleri görecektir ve hastanın durumunu ve ölçümelerini takip edebilecektir. Bu senaryoda bir kullanıcıya birden fazla hasta kayıtlı olabilir ve bildirimleri, uyarıları, alarmları buna göre alabilmelidir.

3.9.3 Acil Yardım Servisi Senaryosu

Acil yardım servisinde senaryo yardım çağrısı alınmasıyla başlar. Alınan çağrı hastanın bulunduğu adresi ve durumu hakkında ön bilgileri içerir. Bir ilk yardım ekibi araçla hastanın bulunduğu yere ulaşır ve işlemlerini başlatır. Bu çalışma kapsamında önerilen sistem için ekibe verilen mobil cihaz, hastanın mobil cihazı ile iletişime geçerek veya hasta kimlik bilgilerinden merkeze bağlanarak giriş yapar. Bu aşamadan sonra hasta hakkında acil durumlarda gerekli bilgiler, örneğin hemofili hastası ise en son ilacı ne zaman aldığı ve kanamanın durdurulabilmesi için hangi tür faktör ilaç kullanılması gerektiği, yakınları ile iletişim bilgileri, doktoru ve allerjileri gibi bilgileri ekranda görülür ve müdahale yapılarak işlem sonuçlandırılır.

3.9.4 Uzman Sağlık Görevlisi Senaryosu

Doktor tarafında muayene ve tedavi senaryosu hastanın ziyaret etmesi veya doktorun o hasta ile ilgili çalışma yapması ile başlar. Hastanın mobil telefonu yanında ise cihazlar arasında otomatik iletişim kurularak doktorun cihazından hasta bilgilerine erişim gerçekleştirilir. Giriş gerçekleştirildikten sonra sistemden hasta ile ilgili bilgilere ulaşılır. Bu bilgiler hasta tarafından gerçekleştirilen kayıtlar, hastanın etkileşimde olduğu sağlık kuruluşlarının listesi, sağlık kuruluşunda bulunan kayıtlar, raporlar, tetkik sonuçları ve doktor tarafından daha önce verilmiş formlar yardımıyla tutulmuş günlük bilgileridir.

Muayene işleminden sonra tedavi aşamasına geçildiğinde doktor hastaya eğitim için verilmesi gereken ilaç kullanımı, perhiz, egzersiz gibi bilgileri sistemde hazır bulunan veya doktor tarafından özelleştirilmiş bir profili atayarak

hastanın cihazına iletir. Doktor bunların dışında hastaya özel olarak bir günlük tutulmasını gerek görürse sistem üzerinden bir form oluşturarak belirli periyotlarla hastaya gösterilmesini sağlayabilir. Ayrıca ölçüm alınacak bir cihaz seçerek sürekli veya periyodik olarak değerlerin kayıt edilmesini isteyebilir. Bu bilgiler dışında doktor belirli bir hastalık için veya özel bir hasta için uyarı ve bildirimlerde gönderilme imkanına sahip olacaktır.

Formlar vasıtasıyla tutulan günlüklerden, tıbbi cihazlar vasıtasıyla alınan ölçümlerden ve hasta tarafından girilen semptom, ilaç kullanım raporları, egzersiz raporları incelenerek tedavinin özelleştirilmesi sağlanmış olur.

3.9.5 Yardımcı Sağlık Görevlisi Senaryosu

Hemşirelerin ve doktor asistanların birçok önemli faaliyeti olmakla beraber birçok görevi hastane yönetim sistemi ile ilgilidir. Bu çalışmadaki asıl amaç doktora yardımcı işlevlerin yerine getirilmesine imkan vermektir. Bu nedenle yardımcı sağlık görevlisi hastaların doldurması için özel soru formları hazırlamak, hastanın hastane sisteminden alınamayan bilgilerini ve doktor notlarını kayıt etmek, yine aynı amaçla dokümanların resimlerini çekerek sisteme yüklemek şeklinde işlevleri olacaktır.

3.9.6 Sağlık Kuruluşu Departmanı Senaryosu

Hastane bölümleri başta raporlama olmak üzere oluşabilecek gereksinimlere cevap vermek üzere doktorların, diğer çalışanlarının, hala aktif olanlar ile tedavisi sonuçlandırılmış olan hastaların listesini güncel bir şekilde hazır tutar. Başka bölümler ile bilgi paylaşımı yapabilir ve gelen rapor isteklerine cevap verebilir.

3.9.7 Sağlık Kuruluşu Senaryosu

Hastaneler ve diğer sağlık kuruluşları, sistemde aktive olduktan sonra bünyesindeki bölümleri ve dolayısıyla hastaları ve doktorların bilgisini tutar. Bu bilgiler yardımıyla rapor isteklerine cevap verir.

Sağlık kuruluşlarında sistemin en önemli bölümü halihazırda kurulu bulunan HBYS (Hastane Bilgi Yönetim Sistemi) sistemine entegrasyondur. Bu yolla sistem kuruluşlar arasında ve doktorlar arasına bilgi paylaşımını mümkün kılacak ve birçok önemli ve işlevsel rapor oluşturulmasını mümkün kılacaktır.

Bu görevlerin dışında sistem yönetimi açısından veritabanı yönetmek, ontolojileri yönetmek, hastalık türlerine göre profiller oluşturmak ve bu profillere göre formları ve kuralları belirlemek ve bir iletişim noktası işlevi görmek görevleri vardır.

3.9.8 Rapor İsteyen Kuruluş Senaryosu

Rapor istemcisi kuruluş sistemine giriş yapar ve yetkilendirme tanımlanır. Yetkili istemci bir dizi sorgu parametresi belirler ve bunu ilgili kurumlara ve hastalara gönderir. Hastalar parametrelere ve istenen verilere göre raporları hazırlayarak istemci kuruluşa geri gönderirler. Son olarak istemci kuruluş gelen raporları birleştirerek tek bir rapor haline getirir. Örneğin bir rapor İzmir ilindeki kronik anemi hastalarının ilaç kullanım miktarları olabilir. Bir diğer rapor ise sistemin GPS yetenekli mobil cihazlar aracılığı ile temin edebileceği konum bilgisi sayesinde belirli bir hastalığın veya salgın şüphesinin gerçek zamanlı coğrafi dağılımının ve bu dağılımın zamana göre gelişiminin görülebilmesi sağlanabilir.

4. TASARIM VE MIMARI

Bu bölümde hedeflenen sistemin tasarımı, Tropos metodolojisinin adımları izlenerek, gerçekleştirilecektir. Sistem tasarımı uzaktan hasta takibi bakış açısıyla oluşturulacak ve gerçekleştirim işlemine hazırlık yapılacaktır

4.1 Gereksinimler

Sistem gereksinimlerini ortaya koymak amacıyla ilk olarak hedeflenen alanın şu andaki işleyişi incelenecektir. Bu inceleme sonucunda alan ortakları ve aralarındaki ilişkiler ve bağımlılıklar belirlenecektir.

İkinci aşamada sistem ile alan ortakları arasındaki ilişkiler belirlenecek ve görevlerden hangileri sistem tarafından yerine getirilebilir ortaya konulacaktır. Ayrıca sistem ek olarak ne katkılar getirebilir sorusunda cevap aranacak ve varsa yeni özellikler tanımlanacak. Bu aşamanın sonunda sistemin hedef analizi detaylı olarak yapılacak ve görevleri netleştirilecektir.

Üçüncü aşamada sistemin bileşenleri ve varsa alt sistemler tanımlanacaktır. Oluşturulan bileşenler ve alt sistemlerin detaylı analizlerinden sonra sistemin gereksinim analizi işlemleri tamamlanmış olacaktır.

4.1.1 Alan Gereksinimleri

Uzaktan hasta takibi bakış açısıyla sağlık alanına baktığımızda alan ortakları olarak Hasta ile doğrudan temasta olan kişi ve kuruluşlar ön plana çıkmaktadır. Bunlar doktorlar, hemşireler, hasta yakını, hasta bakıcısı, sağlık ocakları, hastaneler, fizik tedavi merkezleri, onkoloji merkezleri, diş muayenahaneleri, eczaneler, acil yardım ekipleri olarak sıralanmaktadır. Dolaylı olarak hasta ile temasta olan kurumlar Sağlık Bakanlığı ve Sosyal Güvenlik Kurumu olarak görülmektedir.

Sağlık kuruluşlarının hedefleri hemen hemen aynıdır, bu nedenle bu çalışmada bu kurumlar Sağlık Kuruluşu adı altında değerlendirilmektedir. Doktorlar branşlarından bağımsız olarak aynı muayene ve tedavi sürecinin uyguladıklarından bu çalışma kapsamında Uzman Sağlık Görevlisi adı altında işleme alınmaktadır. Devlet kurumları ve diğer bazı sağlık kuruluşlarının amacı ise verileri analiz ederek daha doğru sağlık politikaları oluşturmak olduğundan bu

kuruluşlarda çalışma kapsamında Rapor İstemcisi Kurum olarak değerlendirilmektedir.

4.1.2 Teknik Gereksinimler

Kavramlar başlığı altında sağlık bilişiminin ihtiyaçlarından detaylı olarak bahsedilmiştir. Burada ön plana çıkan ilk gereksinim sürekli değişen ortama hızlı bir adaptasyon sağlanması gereksinimidir. Oluşturulacak sağlık sistemleri yeni ve ağır bir iş yüküne sebep olmadan değiştirilebilmelidir.

Bir diğer teknik gereksinim erişilebilirlik ve ölçeklenebilirliktir. Sağlık verisi şu anda binlerce farklı kurumda dağınık vaziyettedir. Ayrıca sistemin kullanıcı sayısının birkaç bin den başlayarak milyonlarca kullanıcıya ulaşması söz konusu olabilir. Dolayısı ile sistem en düşük maliyet ile süratli bir biçimde ölçeklenebilir olmalıdır.

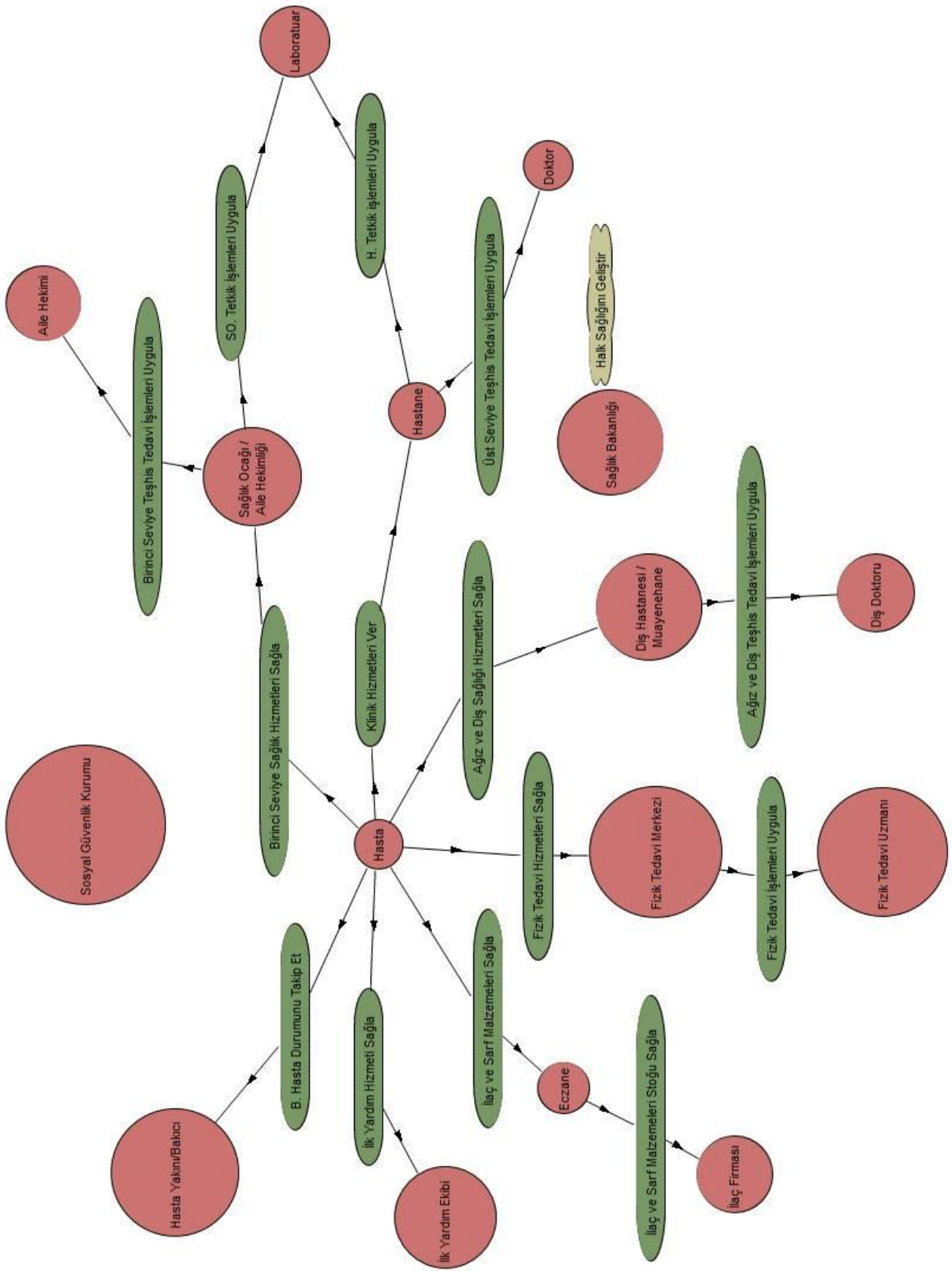
Teletıp alanında çalışan sistemlerin mobil altyapılar başta olmak üzere internet bağlantı altyapısı gibi altyapı kaynaklı kısıtlamalara cevap verebilecek kapasitede olması gerekmektedir. Ayrıca ölçüm amacıyla kullanılacak tıbbi cihazların bağlantı türlerine ve teknolojilerine uygun olması gerekmektedir.

Son olarak güvenlik gereksinimi mutlaka entegre bir biçimde dikkate alınmalıdır. Hasta verilerinin güvenliği hem yasal hemde kullanıcıların gizliliği açısından çok önemlidir.

4.1.3 Erken Gereksinim Analizi

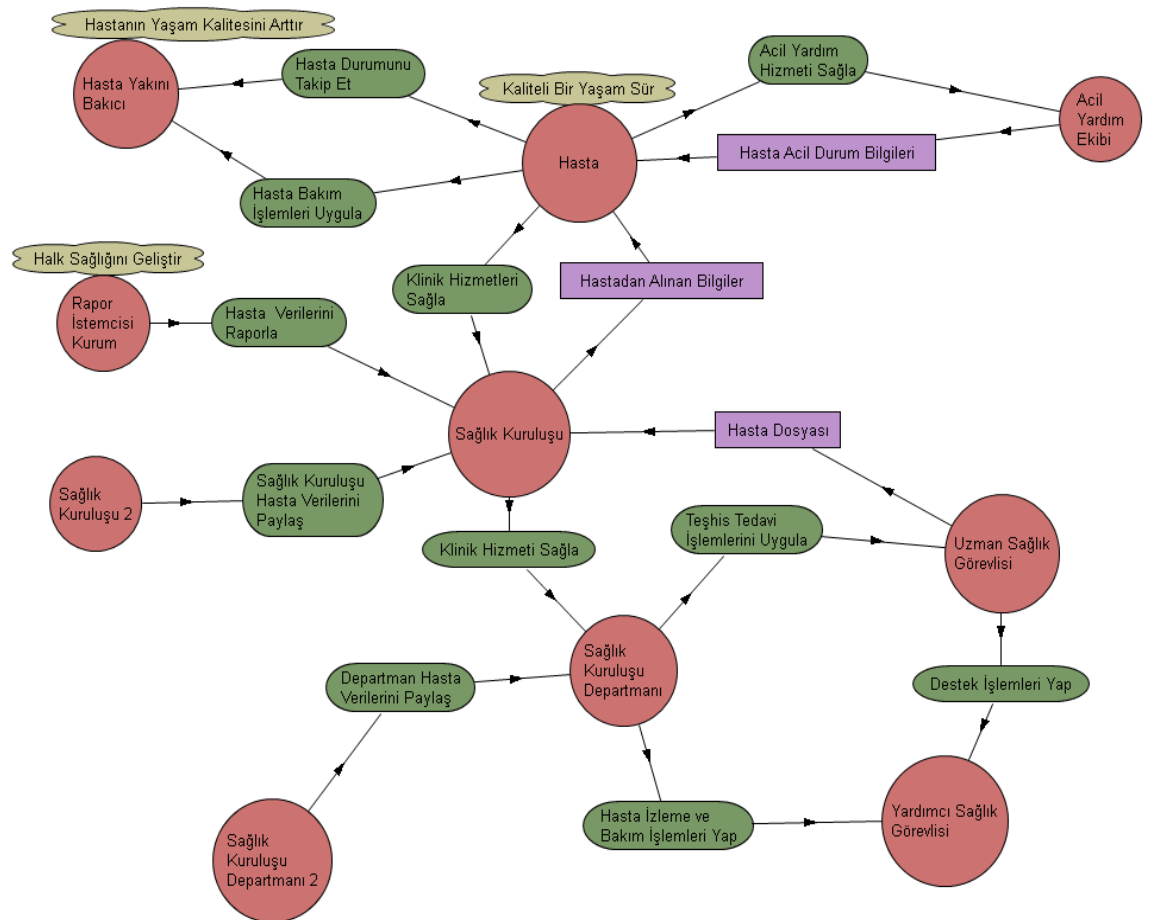
Bu aşamada hedeflenen alanın sistem olmadan önceki durumu ve işleyişi incelenerek amaçları ve birbirine bağımlılıkları belirlenecektir.

Şekil 4-1 de görüleceği üzere Hastayı merkeze alan bir gereksinimler dizisi ortaya çıkmıştır. Buradan görüleceği üzere Hasta tüm sağlık kuruluşlarından teşhis ve tedavi hizmetleri almaktadır ve devlet düzenleyici kurumları ise Halk Sağlığını Geliştirmek gibi işlevsel olmayan hedefler için verileri analiz etme ve raporlama işlemlerini yapmaktadır.



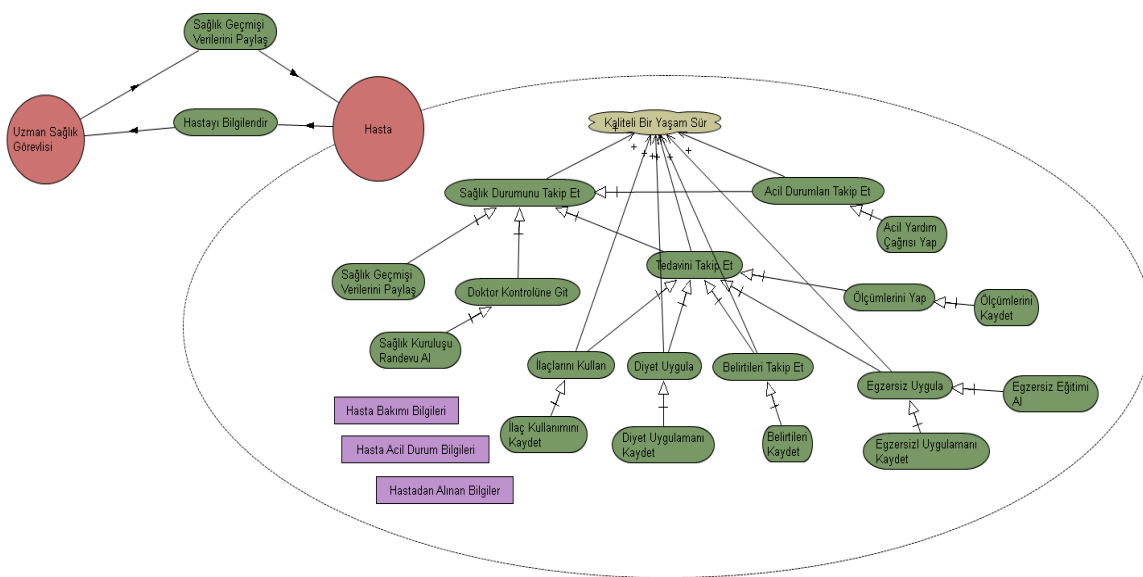
Şekil 4-1 Alan Ortakları

Alan ortaklarının analizi benzer yapılar birleştirilerek sadeleştirilmiş ve daha detaylı bir analiz yapılmıştır. Burada Hasta Bakıcı/Yakını, Hastanın çocuk olması veya yardıma ihtiyaç duyması halinde Hastanın yapması gerekenleri onun yerine yapabilecek bir görev üstlenmiştir. Bunun dışında Hastanın durumunda oluşan gelişmeler Hasta Yakınına bildirilmekte ve takip etmektedir. Acil Yardım Ekibi acil çağrılarını almakta ve belirtilen adreste Hastaya müdahale etmektedir. Burada yazılan Hasta Acil Durum Bilgileri, eğer Hasta bilinci açıksa, Hastanın yanında bilgilere sahip bir kişi var ise geçerlidir. Aksi durumlarda bu bilgi sadece Acil Yardım Ekibinin genel muayenesi ve kanaati sonucu elde edilmektedir. Sağlık Kuruluşu klinik hizmetleri sağlamakta, hastane bilgi yönetim sistemi aracılığı ile Hasta dosyasını muhafaza etmekte ve çeşitli devlet arayüzleri ile hasta verilerini ham olarak göndermektedir. Sağlık kuruluşu burada bir başka kuruluş ile veri alışverişi yapmaktadır ancak bu alışveriş günümüz itibarıyla elde götürülen kağıt dosyalardan ibarettir ve her zaman erişilebilir değildir.



Şekil 4-2 Teletıp Alan Ortakları Sadeleştirilmiş Analiz

Şekil 4-2 de görülen Sağlık Kuruluşu Departmanı sadece bölümler arası veri paylaşımını göstermek ve sağlık çalışanlarını kategorize etmek amacıyla hesaba katılmıştır. Uzman Sağlık Görevlisi bu analizde doktor, diş doktoru, onkolog, fizik tedavi uzmanı gibi uzmanları temsil etmektedir. Uzman sağlık Görevlisi hasta verilerine hastane sisteminden ve hasta ziyaretlerinden elde edilen mülakatlardan temin etmektedir. Yardımcı Sağlık Görevlisi hemşire, laborant ve asistan doktorları temsil etmektedir ve gözlem yapmak ile Uzman Sağlık Görevlisini tamamlayıcı görevler üstlenmektedir.

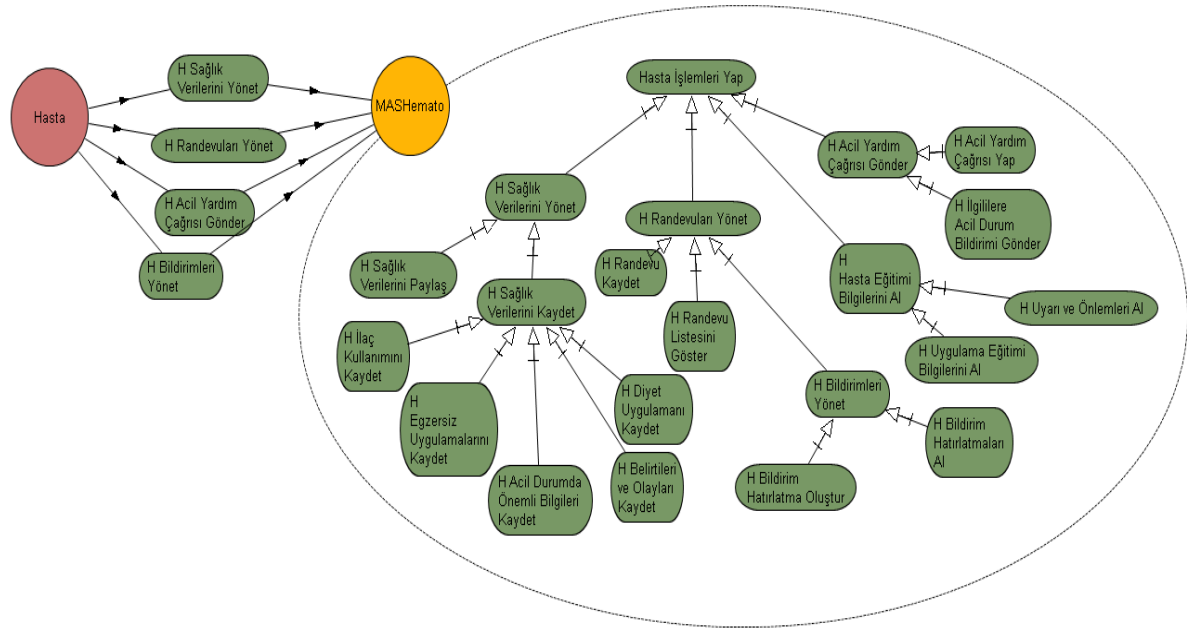


Şekil 4-3 Hasta Erken Gereksinim Analizi

Şekil 4-3 te Hasta için yapılan erken gereksinim analizi görülmektedir. Hasta kaliteli bir hayat sürebilmek için sağlığını takip etmeli ve acil durumlarda hemen bir sağlık kuruluşuna başvurmalıdır.

İyi bir muayene hizmeti alabilmek için hasta verilerini en iyi şekilde paylaşmalıdır, ancak bu aşamada hasta bunu ya kağıt günlüklerde kayıt ederek yapmaktadır ya da sadee aklına geldiği kadarıyla sözlü olarak aktarmaktadır.

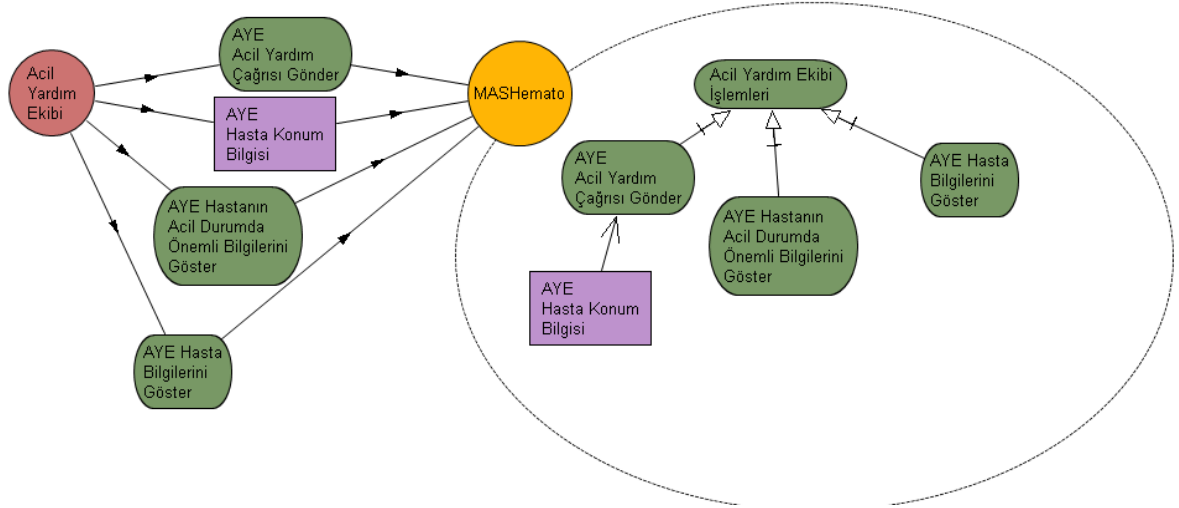
Hasta iyi bir tedavi süreci için ise ilaç kullanımına, diyetine, egzersizlerin uygulanmasına ve ortaya çıkan belirtilere dikkat etmelidir. Hastaya egzersiz eğitimi günümüz itibarı ile doktorlar tarafından elden dağıtılan broşürlerden ibarettir.



Şekil 4-6 Hasta MASHemato Geç Gereksinim Analizi

Şekil 4-6 da Hasta ile Sistem aktörü arasında bir analiz gerçekleştirilmektedir. Bu diyagramda görülen analiz ile birlikte sistemin Hasta ile ilgili yapacağı işlemler belirlenmiş olacaktır. Hastanın sisteme atanabilir her hedefine karşılık sistemin gerekli iş aktarımını gerçekleştirmek üzere hedefi bulunur. Şekilde sistem hastanın, sağlık verilerinin paylaşılmasını, randevuların yönetilmesi, bildirimlerin yönetilmesi işlemini yapmaktadır. Hasta doktordan alacağı eğitici bilgilendirmeleri, acil yardım çağrılarını, kayıt altına alacağı günlük verilerini sistem aracılığı ile gerçekleştirecektir.

öngörülmüştür ancak bunun yanısıra rapor taleplerini gerçekleştirme ve dışarıdan doğrulanmış verilerin hastane bilgi sistemine kayıt edilmesi işlemlerinin de sistem üzerinden gerçekleştirilecektir.



Şekil 4-9 MASHemato Acil Yardım Ekibi Geç Gereksinim Analizi

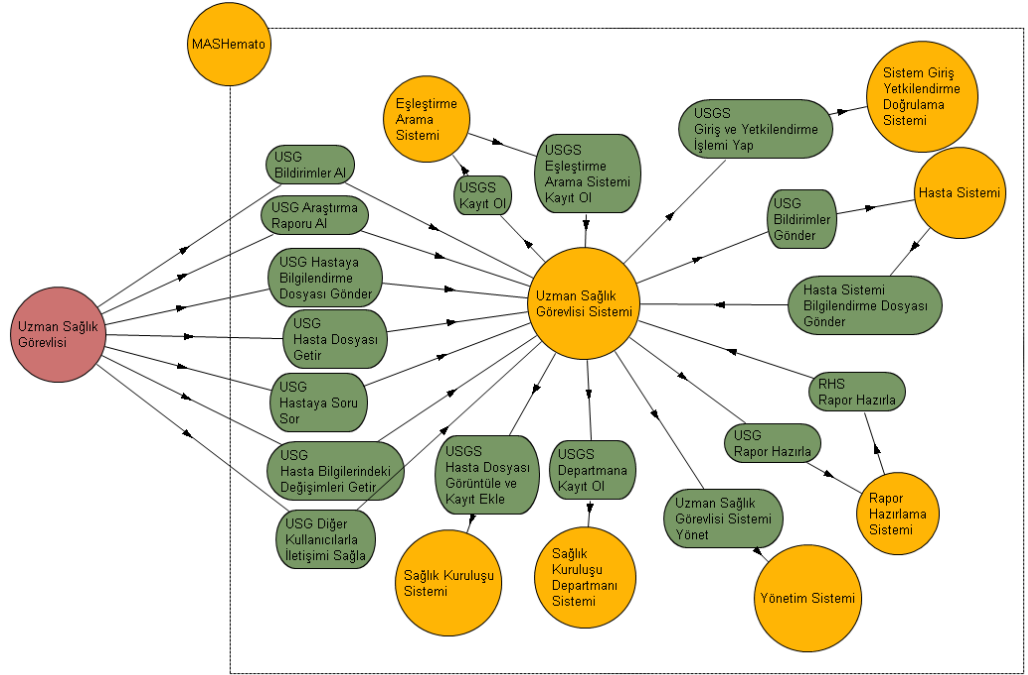
Şekil 4-9 da gösterilen diyagram ile acil yardım ekibi ile sistem arasındaki ilişki incelenmiştir. Acil yardım ekibinin amacı sistem üzerinden yardım çağrılarını alabilmek ve ilgili hastanın öncelikle acil durumda gerekli bilgileri olmak üzere tüm tıbbi kayıtlarının görüntülenebilmesidir. Sistem ayrıca mobil cihazlardan elde ettiği küresel konum belirleme sistemi koordinat bilgisi ile hastanın net olarak konumunu alabilecektir.

Geç gereksinim analizi aşamasının sonucunda oluşturulan şemaları Ek-2 bölümünde bulabilirsiniz.

4.2 Sistem Mimarisi

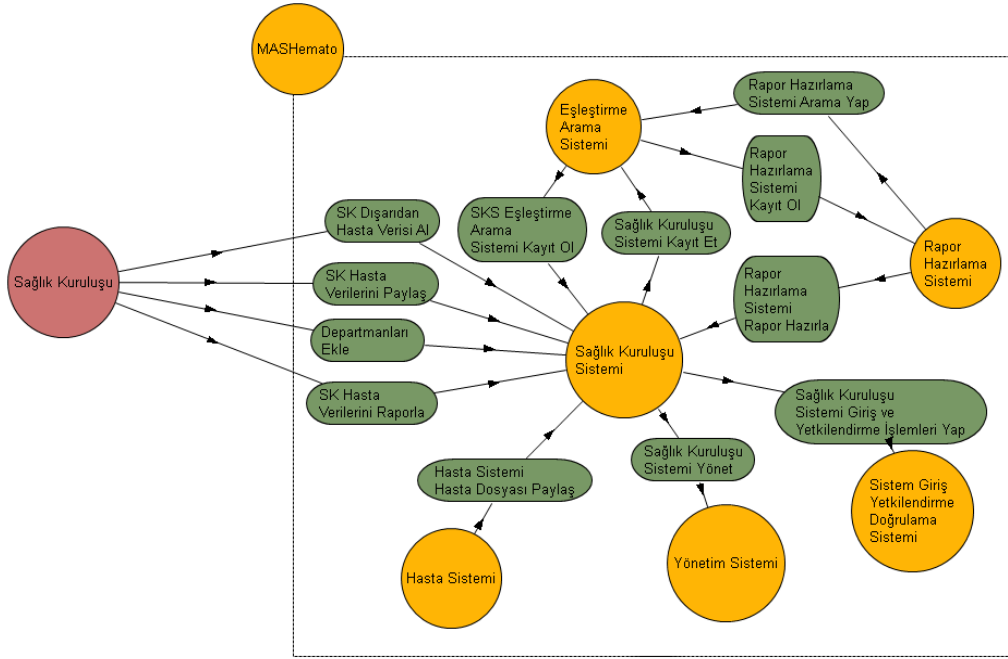
Sistem mimarisi teknik ve alan gereksinimlerine uygun bir şekilde Tropos metodolojisi yardımıyla gerçekleştirilecektir. Öncelikle Geç Gereksinim Analizi sırasında tanımlanmış hedefler dikkate alınarak gereken alt sistemler önerilir. Alt sistemler ve etmenler tanımlandıktan sonra detaylı tasarım aşamasında tüm hedefler somut planlara bağlanarak devam edilir. Planların oluşması ile birlikte tasarımın son aşamasına geçilerek UML tasarımları ve sıra diyagramları oluşturulur.

Yönetim Sistemi hem diğer sistemlerin başlatılmasını ve sonra da çalışır olarak devam etmesini sağlar. Ayrıca Hasta sisteminin gerekli durumlarda başka kurum sunucusuna taşınmasını sağlayarak iletişimi daha verimli hale getirmeyi amaçlar.



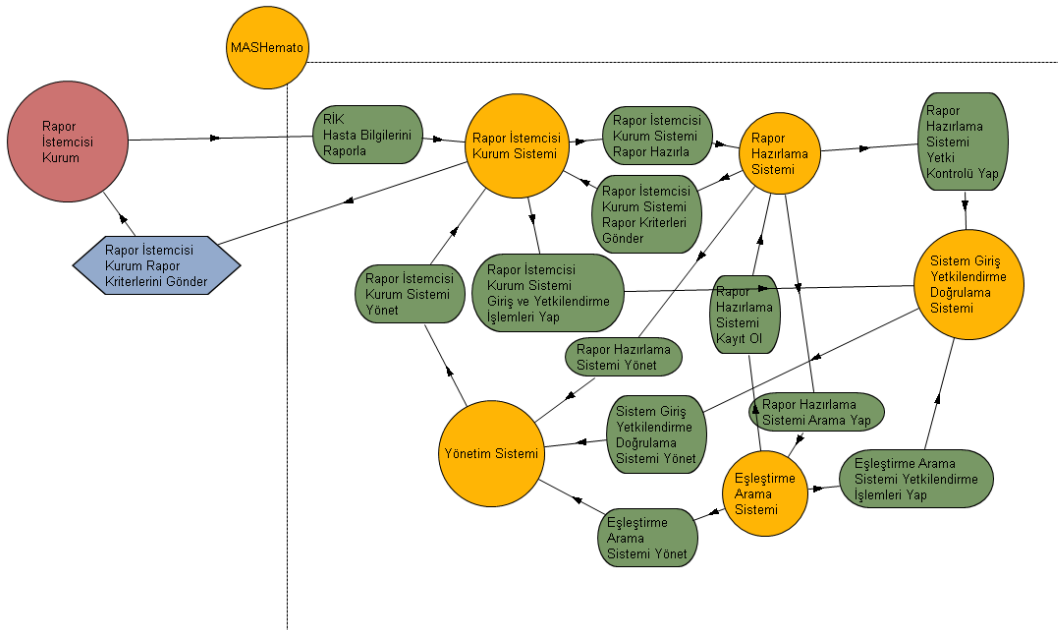
Şekil 4-11 Mimari Tasarım: Uzman Sağlık Görevlisi

Şekil 4-11 de görüldüğü gibi doktorlar Uzman Sağlık Görevlisi Sistemi alt sistemi üzerinde tüm görüntüleme ve takip işlemlerini gerçekleştirebilecektir. Sistem doktorun görevli olduğu sağlık kuruluşundan ve hastanın etkileşimde olduğu diğer sağlık kuruluşlarından gerekli tüm hasta dosyalarına erişim sağlayabilmektedir. Rapor Hazırlama sistemi aracılığıyla bir hasta hakkında veya genel olarak bir rapor isteminde bulunabilmektedir. Ayrıca sistem üzerinden hasta ile iletişime geçerek gerekli eğitim ve bildirim işlemleri yapılabilmektedir.



Şekil 4-12 Mimari Tasarım: Sağlık Kuruluşu

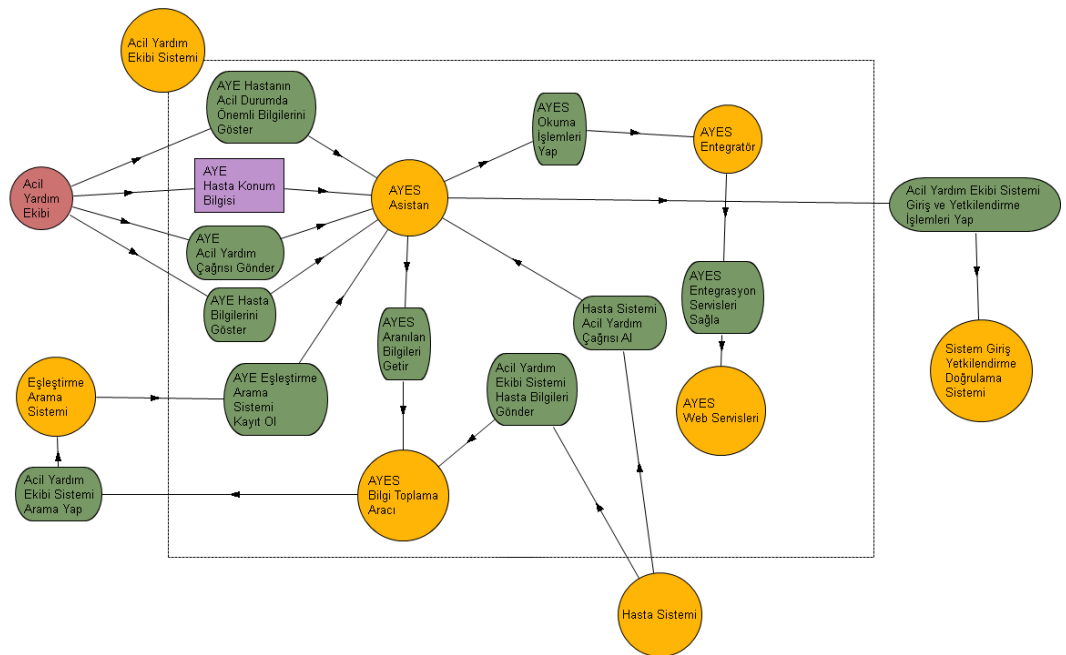
Şekil 4-12 de tanımlanan Sağlık Kuruluşu Sistemi alt sisteminin esas amacı yerel hastane bilgi yönetimi sisteminde tutulan verilerin diğer kuruluş ve uzman sağlık görevlileriyle paylaşılmasıdır. Ayrıca Rapor Hazırlama Sistemi ile beraber çalıştığından rapor istekleri otomatik olarak karşılanabilmektedir.



Şekil 4-13 Mimari Tasarım: Rapor İstemcisi Kurum

Şekil 4-13 te Rapor İstemcisi Kurum için gerekli olan tasarım görülmektedir. Burada tanımlanan alt sistem ile kurumun görsel arayüz ve rapor talepleri yönetilmektedir. Alt sistem rapor taleplerini oluşturmakta ve Rapor Hazırlama Sistemine yönlendirmektedir.

Rapor Hazırlama sistemi gerekli bağlantıları kurarak diğer etmenlerden bilgileri toplamakta ve rapor halinde istemciye geri göndermektedir.



Şekil 4-14 Mimari Tasarım: Acil Yardım Ekibi Sistemi

Şekil 4-14 te acil yardım ekibinin kullanacağı Acil Yardım Ekibi Sistemi görülmektedir. Tasarımda AYES Web Servisleri ile acil çağrı hizmeti sistemine entegrasyon yapılmaktadır. AYES entegratör Asistan etmeninden gelen istekleri sisteme uyarlayarak kayıt etmektedir. Bilgi Toplama Aracı etmeni ise Asistan etmenine gerekli bilgileri temin ederek sağlamaktadır. Asistan etmeni hem arayüz işlemlerini hemde Hasta Sisteminden gelen acil yardım çağrılarını takip etmektedir.

- iv. Hasta Yakını Bakıcı Sistemi
- v. Acil Yardım Ekibi Sistemi
- vi. Rapor İstemcisi Kurum Sistemi

ve MASHemato sistemi için gereken alt sistemler:

- i. Eşleştirme ve Arama Sistemi
- ii. Veri Yönetimi Sistemi
- iii. Sağlık Kuruluşu Sistemi
- iv. Rapor Hazırlama Sistemi
- v. Yönetim Sistemi
- vi. Giriş Yetki ve Doğrulama Sistemi

olarak belirlenmiştir. Sonrasında bu alt sistemler detaylı olarak tasarlanmış ve gerekli etmenler belirlenerek analiz edilmiştir. Oluşturulan etmenler görevlerine göre gruplandırılırsa:

- i. Asistan Etmenler, ilgili sistemde bir erişim noktası olarak arayüz görevleri ve bazı ana görevleri yerine getirirler. Ayrıca yapılacak işleri yardımcı etmenlere paylaştırarak gerçekleşmesini sağlar.
- ii. Bilgi Toplama Etmenleri, Aracı Desenine uygun olarak tasarlanan bu etmenler arama sistemine bağlanarak ilgili etmenlerin adresini alır bilgileri hazırlar ve Asistan etmenlere geri gönderirler.
- iii. Hasta Takip Etmenleri, Gözlemci Desenine uygun olarak tasarlanan bu etmenler, Asistan etmenine gelen değişiklikleri takip ederek duruma göre alarm ve bildirim üretirler.

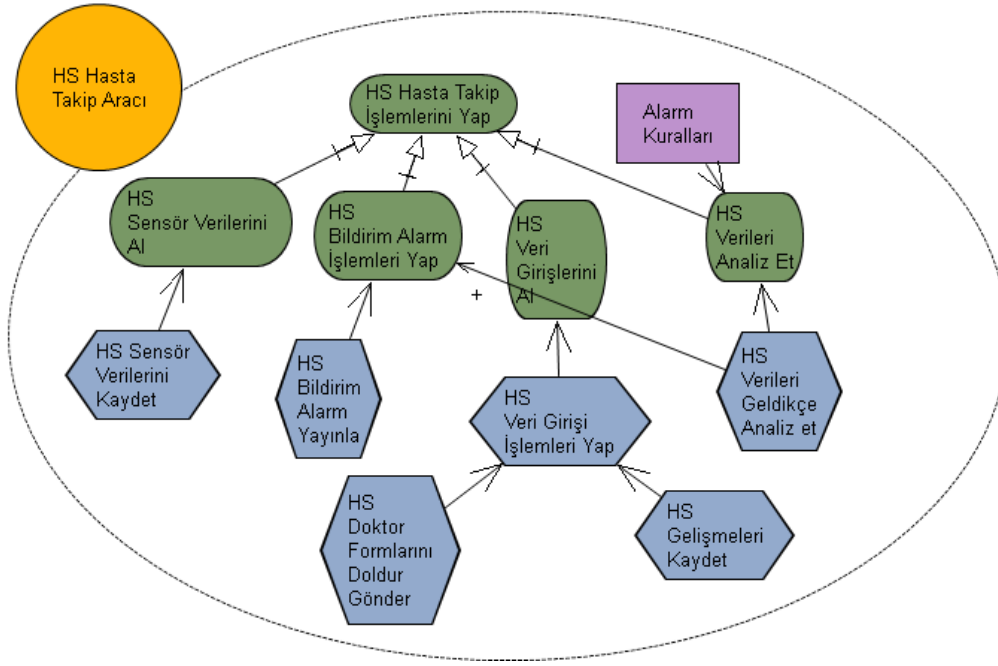
- iv.Rapor Hazırlama Etmenleri, Anlaşma Desenine uygun olarak tasarlanmıştır. Rapor için gerekli bilgileri toplar ve rapor haline getirirler.
- v.Entegrasyon Etmenleri, Aracı Desenine uygun olan bu tasarımda diğer etmenlerden gelen istekler ilgili standartlara uyumlandırılarak web servislerine gönderilir.
- vi.Yönetim Etmenleri, Gözlemci Deseninden faydalanılan bu etmenler, tüm sistemdeki etmenlerin çalışmaya başlaması ve devam etmesinden sorumludurlar. Etmenleri gözlemleyerek herhangi bir nedenden durmaları halinde yeniden başlatırlar. Ayrıca alt sistemlerin gerekli durumlarda göç etmelerini sağlayarak sistemin verimliliğine katkıda bulunurlar.

Mimari tasarım aşamasının sonucunda ortaya çıkan şemaları Ek-3 te bulabilirsiniz.

4.2.2 Detaylı Tasarım

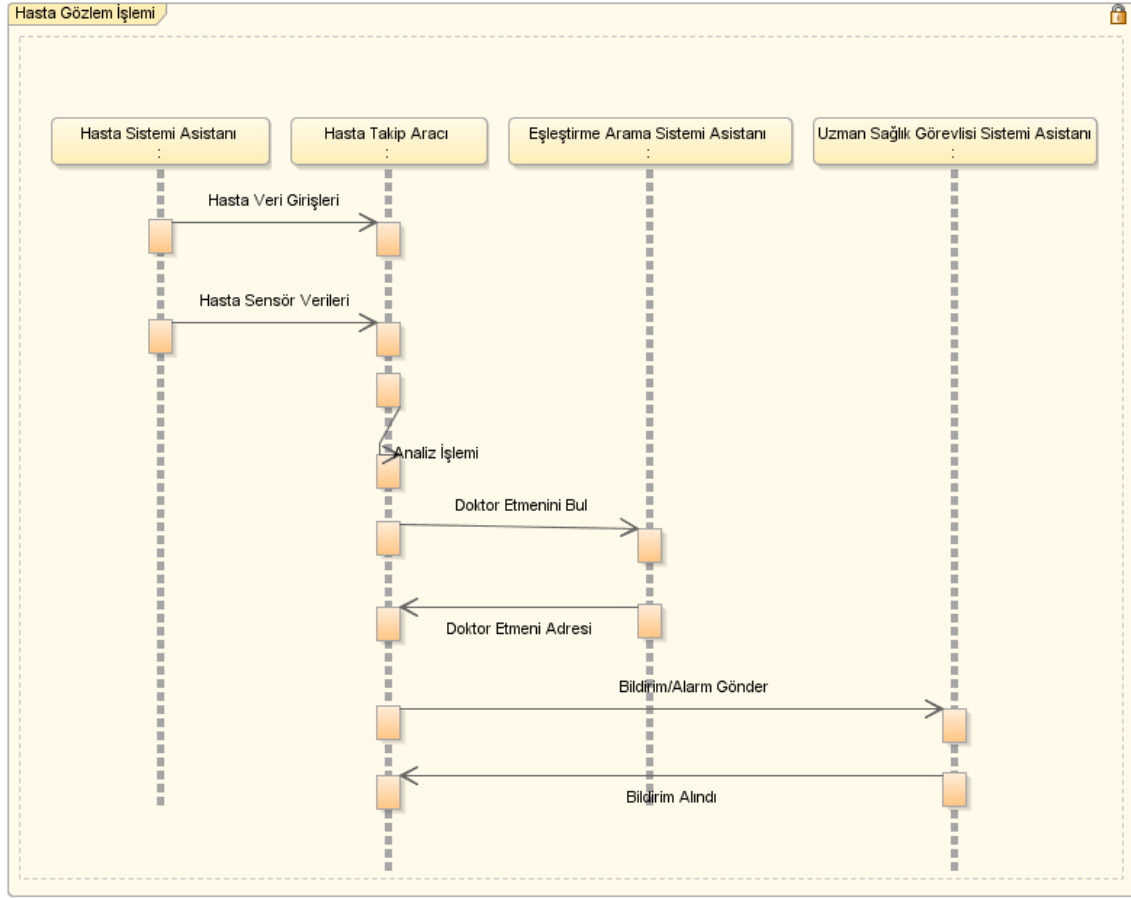
Bu aşamada daha önce tamamlanan mimari tasarım aşamasında belirlenmiş olan etmenlerin detaylı hedef ve amaç araç analizleri yapılmaktadır ve etmenlerin planları belirlenmektedir. Etmenlerin analizleri tamamlandıktan sonra ortaya çıkan planların gerçekleştirime hazırlanması için gerekli UML tasarımlar yapılmaktadır.

Şekil 4-17 de görülen etmen Hasta Sistemi için hasta hakkındaki tüm bilgileri toplayan bir etmendir. İşlemlere önce hastanın etkileşim yaptığı tüm sağlık kuruluşlarının listesini getirerek başlamakta ve sonra kuruluşlardan verileri toplamaktadır. Kuruluşlardan sırayla verileri toplama işleminde de önce ilgili kuruluşu aratma yaparak etmenin adresi temin edilmekte ve bilgiler talep edilmektedir. Son olarakta gelen belgelerin yazarı ve bütünlüğü kontrol edilmektedir.



Şekil 4-18 Hasta Sistemi Hasta Takip Aracı Detaylı Tasarım

Şekil 4-18 de tasarımı görülen Hasta Takip Aracı etmeni, hastanın genel durumunu takip etmekte ve gelen verileri analiz etmektedir. Doktor tarafından atanan alarm kurallarına göre gerekli durumlarda bildirim ve gerekli durumlarda acil yardım çağrısı gönderebilmektedir. Bunun dışında bu etmen sensör verilerini ve hasta günlüklerini almakla da yükümlüdür.



Şekil 4-20 Hasta Gözlem İşlemi Sıra Diyagramı

Detaylı tasarımın son aşaması planların gerçekleştirime hazırlanması için UML diyagramlarının hazırlanmasıdır. Bunun bir örneği olarak hasta verilerinin Takip etmeni tarafından işlenmesi ve diğer etmenler ile etkileşimi üzerine analiz edilmiş bir UML Sıra Diyagramı Şekil 4-20 de gösterilmiştir.

5. GERÇEKLEŞTİRİM VE DEĞERLENDİRME

5.1 Gerçekleştirim

Gerçekleştirim işlemi Tropos aracı yardımıyla etmen kodlarının oluşturulmasıyla başlamaktadır. Etmenlerin kodlanması için gerekli dosyalar ve tanımlamaların bir kısmı hazır olarak elde edilir. Oluşturulan dosyalardan Plan kodları tamamlanarak etmen geliştirme işlemi sonuçlandırılır. Etmen kodlarının gerçekleştirimi Java dilinde ve seçilen Jadex platformunun kütüphaneleri kullanılarak Eclipse bütünleşik geliştirme ortamı yardımıyla yapılır.

İkinci aşamada başta uyum amaçlı web servisleri, kullanıcı arayüzlerini etmen sistemine bağlayan web servisleri oluşturulmaktadır. Uyum servisleri JAX WS ve tıbbi bilişimde kullanılan standartların (DICOM, HL7) yazılım kütüphaneleri kullanılarak gerçekleştirilmektedir. Kullanıcıların cihazlarının bağlantı yapacağı web servisleride platform bağımsızlığı amacıyla HTML5 ile en uyumlu Restful web servisleri yöntemiyle gerçekleştirilmektedir. Web servislerinin gerçekleştirimi bu amaçla birçok araç sağlayan Netbeans bütünleşik geliştirme ortamı ile yapılmaktadır.

Son olarak kullanıcı arayüzleri HTML5 dili kullanılarak melez uygulama geliştirme aracı olan Intel XDK ortamı kullanılarak gerçekleştirilecektir. HTML5 uygulamaları yaygın olarak akıllı web sayfaları olarak kullanılsada uygulama paketlenerek mobil ve masaüstü cihazlara melez uygulama olarak kurulabilmektedir. Ayrıca bu ortam aracılığıyla gerçekleştirilen uygulamalar mevcut yerli uygulama dağıtım kanallarıyla dağıtılabilen platform bağımsız uygulamalar geliştirilebilmektedir. Ortam proje oluşturma, ekran tasarımı, kodlama, test ve paketleme için gerekli araçlara sahiptir.

5.1.1 Etmenler

Etmenler Jadex platformunda bir tanımlayıcı dosya yardımıyla XML biçiminde tanımlanır. Hedefler, planlar ve koşullar bu dosya içerisinde belirtilir. Planlar için ayrı Java kod dosyaları vardır ve gerçekleştirim için plan kodları tamamlanarak derlenmelidir.

Ek-4'te Hasta Sistemi Asistan etmeninin XML tanımlaması verilmiştir.

5.1.2 Web Servisleri

Kullanıcı arayüzlerinin çoklu-etmen sistemine bağlanması, etmen sisteminin entegrasyon amaçlı olarak HBYS sistemlerine bağlanması gereksinimini karşılamak için web servisleri kullanılmaktadır. Arayüzler ve etmenler arasında Restful Web Servisleri ve entegrasyon amacıyla JAX WS web servisleri kullanılmıştır.

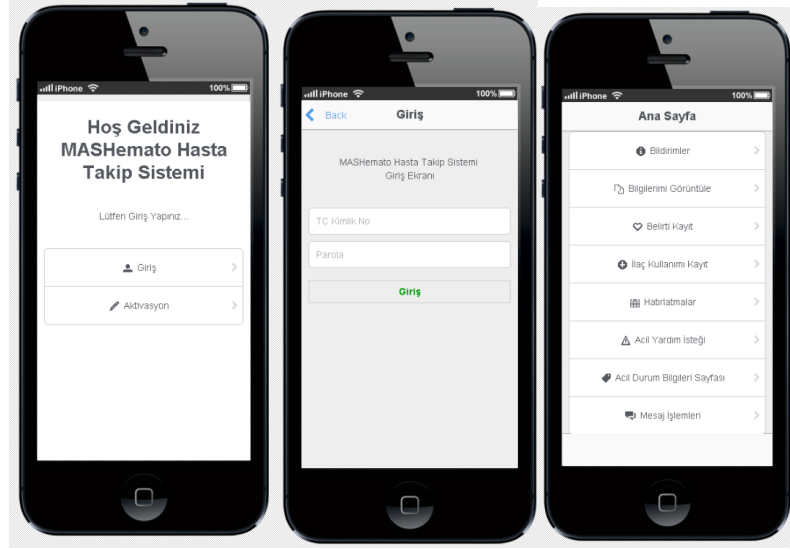
Hasta sistemine bağlanmak için kullanılacak Restful web servis örneği Ek-5'te verilmiştir.

Diğer kullanıcıların sistemlerine bağlanmak için kullanacakları servisler büyük oranda örnekte verilen web servisine benzeyecektir. Entegrasyon amaçlı web servisleri hâlihazırda sağlık kuruluşlarında kullanılmakta olan kurulu sağlık sistemleri ile beraber çalışması nedeniyle ilgili ticari yazılıma özgü olacaktır. Bu nedenle bu çalışma kapsamında örneklenmemiştir.

5.1.3 Kullanıcı Görsel Arayüzleri

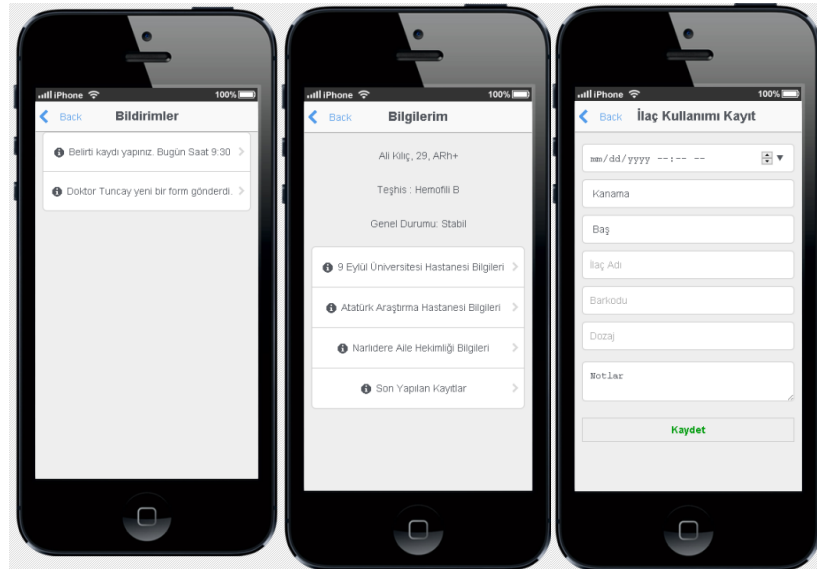
Kullanıcı arayüzlerinin gerçekleştirimi HTML5 kullanılarak gerçekleştirilmiştir. Bu sayede hem birçok mobil cihaz için gerekli uygulamalar hem de web arayüzleri tek bir proje üzerinden ve tek bir kaynak kod çalışması ile tamamlanabilmiştir.

Hasta tarafından kullanılacak arayüz Giriş ekranıyla başlamaktadır. Eğer kullanıcı ilk kez giriş yapacak ise bir aktivasyon işlemi gerçekleştirecektir. Giriş işlemi başarılı bir şekilde tamamlandığında Ana sayfa ekranını görecektir ve buradan bütün işlevlerin listelendiği menüden işlemini seçecektir.



Şekil 5-1 Karşılama Giriş ve Ana Ekranlar

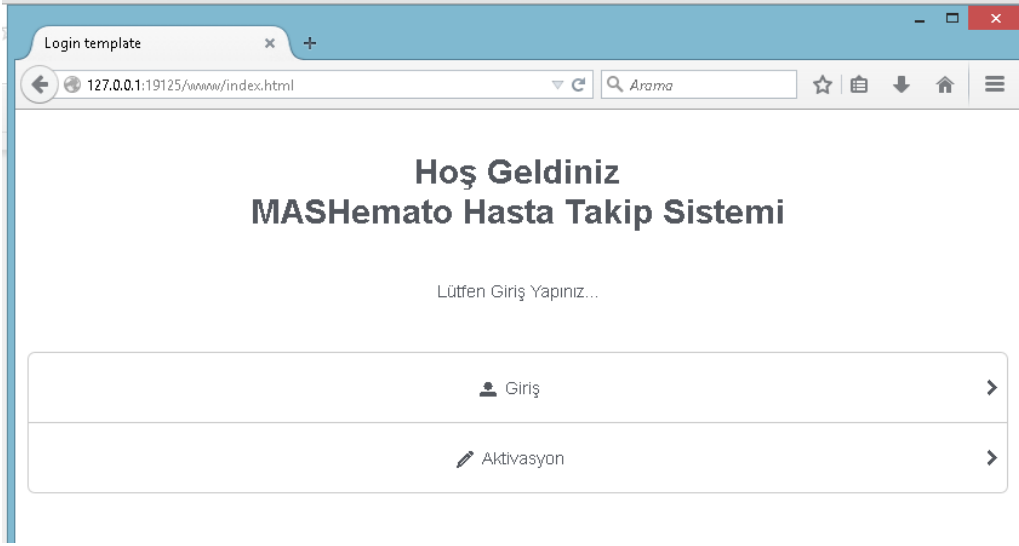
Şekil 5-1 de görülen ekranlar Restful web servisleri ile çoklu-etmen sistemine bağlanarak işlemleri gerçekleştirmektedir. Uygulama pluginler aracılığı ile var ise tıbbi cihazlara bağlanarak gerekli verileri otomatik olarak Hasta Asistanı etmenine göndermektedir.



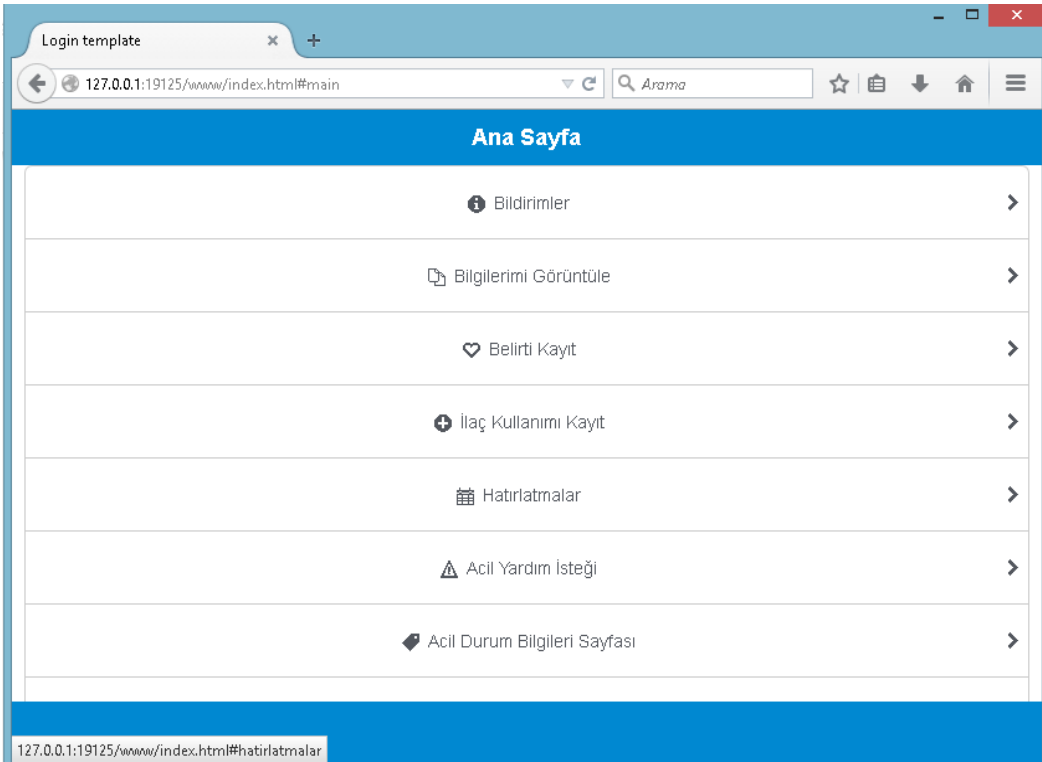
Şekil 5-2 Bildirimler, Bilgilerin ve İlaç Kullanım İşlemleri Ekranı

Şekil 5-2 de görülen Bilgilerim ekranı Hasta Sistemi Asistanı etmenine yapılan istek sonucu Bilgi Toplama Aracı yardımıyla toplanan bilgiler ve Hastanın daha önce kayıt etmiş olduğu bilgilerin bir araya getirilmesi ile

listelenmektedir. Listedenden bir sağlık kuruluşu seçilerek bu sağlık kuruluşunda bulunan hasta dosyasıda incelenebilmektedir.



Şekil 5-3 Uygulama Web Arayüzleri Giriş



Şekil 5-4 Web Arayüzleri Ana Sayfa

Şekil 5-3 ve 5-4 te aynı uygulamanın web sayfası olarak kullanımı görülmektedir. Buradan da anlaşılacağı gibi uygulama kolayca internette yayınlabilecek biçimde olup dağıtım ve erişim açısından büyük avantajlar sağlamaktadır.

5.2 Değerlendirme

Gerçekleştirim işlemi tamamlandıktan sonra sistem güvenlik açısından iki konuda değerlendirilebilir. Birinci konu olarak sistemin sağlamlığı temel olarak MASHemato Yönetim Sistemi ile sağlanmaktadır. Yönetim sistemi kayıtlı olan sistemleri kontrol etmekte ve sonlanan sistemleri yeniden başlatmaktadır. Bir diğer fonksiyonu olan göç ettirme ile yoğunluğun fazla olduğu makineden bazı alt sistemlerin taşınmasını sağlayarak yük dengesine katkıda bulunmaktadır. İkinci konu ise yetkilendirme ve doğrulama işlemidir. Yetkilendirme ve doğrulama sistemi ile MASHemato içerisinde başlatılan etmenlerin doğrulanması sağlanmış ve bu yolla güvenliğin artırılması hedeflenmiştir.

Ölçeklenebilirlik konusunda sistem veri kayıt işlemleri dışında minimum kaynak kullanması ile bir tek sunucuda onbinlerce etmenin rahatlıkla çalışabildiği gözlemlenmiştir. Ayrıca kullanılan platformun sağladığı imkanlar ve yönetim sisteminin göç ettirme kapasitesi sayesinde çalışan sisteme yeni sunucular eklemek ve sunuculara bakım yapmak çok kolay olmuştur.

Sistem hastaların sadece birkaç tuşla gelen sorulara yanıt vererek çok önemli miktarda veri girişi yapabilmesi, doktorların hastalık türüne göre hazır profillerden seçebilmesi sayesinde doktora ek yük getirmemesi, otomatik rapor hazırlayabilme kapasitesi ile raporları otomatik olarak oluşturması, Hasta mobil telefonu ile sağlık çalışanlarının mobil cihazlarının yakınlığında bilgi aktarabilmesi sayesinde gayet kullanışlı bir sistem olarak sonuçlanmıştır.

Sistemin gerçekleştirimi sırasında bazı zorluklarla karşılaşmıştır. Bunların ilki sağlık kuruluşlarında kurulu bulunan hastane bilgi yönetimi sistemlerine entegrasyon işlemidir. Kullanılan sistemler çok çeşitli ve yazılımlar ticari olduğundan entegrasyon işlemine başlanması zorlaşmış ve sonrasında ilgili HBYS yazılımının entegrasyon kapasitesinin az olması sorunlar yaratmıştır. Bir diğer zorluk birçok mobil işletim sistemine uygulama hazırlanması olarak ortaya çıkmıştır. İlgili mobil işletim sistemleri birbirinden çok farklı olmakla birlikte

uygulama geliştirme araçları ve kullandıkları dillerde önemli farklılıklar göstermektedir.

Çizelge 5-1 de bu tez çalışması kapsamında oluşturulan MASHemato sisteminin daha önce incelenmiş olan benzer çoklu-etmen sistemleri ile karşılaştırması görülmektedir. En önemli farklar sistemin gereksinim tabanlı bir çoklu-etmen sistemi geliştirme metodolojisi ile yapılmış olması ve hem sistem içi hemde kurumlar arası tüm verilerin paylaşılmasına izin vermesidir. Ayrıca MASHemato sistemi için güvenlik konusunda çalışma yapılmış olup somut öneriler getirilmiştir ancak diğer sistem önerilerinde güvenlik konusuna değinilmediği görülmektedir.

Çizelge 5-1 Tez Çalışması ve Benzer Sistemleri Karşılaştırması

Çoklu-Etmen Sistemi	Metodoloji Kullanımı	Ontoloji	Güvenlik	Mobil	Veri Paylaşımı
MASHemato	Tropos	RDF +OWL	SSL VPN +Sistem İçi Doğrulama	Evet	Evet
SAPHIRE	—	RDF +OWL	—	—	Kısmi
Aingeru	—	+OWL	—	Evet	Kısmi
HeCaSe2	—	—	—	—	Kısmi
Cervantes et al.	—	—	—	—	Kısmi
Koutkias et al.	—	—	—	Evet	Kısmi

6. ÖZET VE SONUÇ

Bu çalışma kapsamında ilk olarak uzaktan hasta takip sisteminin çözüm önerdiği konular derinlemesine incelenmiştir. İncelemeler sonrasında çözüm ortaya konmuş ve çözümün gerçekleştirilmesi için anlaşılması gereken kavramlar üzerinde durulmuş ve bu kapsamda etmen teknolojisi, etmen tabanlı yazılım geliştirme, sağlık bilişimi, hematoloji, web servisleri ve ontoloji kavramları detaylı olarak tanımlanmıştır.

Kavramların tanımlanmasının ardından çözümün teknik analizine geçilmiş ve başta bu çalışmaya benzer projeler olmak üzere, altyapı seçenekleri, platform seçenekleri karşılaştırmalı analizler ile ayrıntılandırılmıştır. Güvenlik kavramı üzerinde durulmuş ve altyapısal güvenliğin nasıl sağlanacağı araştırılmış ve tespit edilmiştir. Tüm kavramsal ve teknik analizlerin tamamlanmasının ardından sistemin kullanım senaryoları oluşturularak bir sonraki aşama olan tasarım için bir kaynak hazırlanmıştır.

Tropos metodolojisinin aşamaları takip edilerek sistem adım adım oluşturulmuş, test edilmiş ve tasarım gerçekleştirime hazır hale getirilmiştir. Gerçekleştirim aşamasında Tropos tasarımlarından t2x aracı yardımıyla otomatik olarak oluşturulan Jadex etmen kodlarıyla işleme devam edilmiştir ve gerçekleştirim sonuçlandırılmıştır. Gerçekleştirim Eclipse IDEsi aracılığıyla Java dilinde tamamlanmıştır.

MASHemato sistemi, doktorların hastaları daha yakından ve yüksek doğruluk oranıyla devamlı takip etmesini sağlayan, acil yardım gerektiren durumlarda hastanın bilgi vermesine gerek kalmadan hastalığını, allerjilerini ve diğer çok önemli bilgilerini otomatik olarak görülebilmesine imkan veren, hasta yakınlarının tedavi sürecine müdahil olarak katkı sağlamasına yardımcı olan, gerçek zamanlı rapor sağlamasıyla bir çok araştırma ve kaliteli raporun oluşturulabilmesini sağlayan, bir çoklu-etmen sistemi olarak ortaya konulmuştur.

Bu çalışma ile Türkiye koşullarına uygun ve daha sonraki çalışmalara referans olabilecek bir çoklu-etmen sağlık sistemi uygulaması yapılmıştır. Yine bu çalışma çıktılarından sağlık bilişimi standartları referans listesi önemlidir. Ayrıca sağlık bilişiminin karakterine uygun bir tasarım yöntemi olan Tropos

metodolojisinin detaylı uygulaması daha sonraki alıřmalarada katkı saęlayacaktır.

Gelecekte bu alıřma devamında yapılabilecek projelerden bazıları gerek zamanlı salgın hastalıkların takibi, hematoloji dıřında saęlık alanlarının takibi, saęlıklı insanların sisteme dahil edilmesini öngören bir alıřma olabilir. Sistemin fiziki altyapı ihtiyacını göz önüne alırsak bir ulusal saęlık bulutu alıřması ilgi görecek alanlardan biridir.

KAYNAKLAR DİZİNİ

Berners-Lee, T., Hendler, J., and Lassila, O., 2001, The Semantic Web: Scientific American. Scientific American.

Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., and Mylopoulos, J., 2004, Tropos: An agent-oriented software development methodology, Autonomous Agents and Multi-Agent Systems, 8, 203–236.

Cervantes, L., Lee, Y., Yang, H., Ko, S., Lee, J., 2007, Agent-based iIntelligent decision support for the home healthcare environment, Advances in Hybrid Information Technology Lecture Notes, pp 414-424

Doğaç., A., Laleci, G.B., Olduz, M., Tasyurt, İ., Yuksel, M., Okcan, A., 2008, SAPHIRE: A multi-agent system for remote healthcare monitoring through computerized clinical guidelines, Whitestein Series in Software Agent Technologies and Autonomic Computing, pp 25-44

FIPA , 1997, FIPA 97 Specification, Part 1: “Agent Management. Foundation for Intelligent Physical Agents”, Version 1.2, October 10

FIPA , 1997, FIPA 97 Specification, Part 2: “Agent Communication Language. Foundation for Intelligent Physical Agents”, Version 1.2, October 10

Giorgini, P., Mylopoulos, J., and Sebastiani, R., 2005, Goal-oriented requirements analysis and reasoning in the tropos methodology, Engineering Applications of Artificial Intelligence, Agent-oriented Software Development, 18(2), 159 – 171.

Giorgini, P., Kolp, M., Mylopoulos, J., 2003, Organizational patterns for early requirements analysis, CAiSE03, p 17

Howden, N., Ronnquist, R., Hodgson, A., and Lucas, A., 2001, JACK intelligent agents - summary of an agent infrastructure, Proceedings of the 5th ACM International Conference on Autonomous Agents.

Isern, D., Sanchez, D., Moreno, A., 2007, HeCaSe2: A Multi-agent Ontology-Driven Guideline Enactment Engine, Springer Lecture Notes in Computer Science Volume 4696, pp 322-324

Kaushansky, K., Lichtman, M., Beutler, E., Kipps, T., Prchal, J., Seligsohn, U., 2010, Williams Hematology 8e, McGrawhill, p 2460

Klyne, G. and Carroll, J. J., 2004, Resource Description Framework (RDF): Concepts and Abstract Syntax, W3C Recommendation, WorldWideWeb Consortium.

KAYNAKLAR DİZİNİ (devam)

Koutkias, V.G. and Maglaveras,N. and Chouvarda, I. and Gogou G, Lekka I and Goulis D and Avramidis A and Karvounis ,C and Louridas G and Balas EA., 2005, The citizen health system (CHS): a modular medical contact center providing quality telemedicine services.

Maes, P.,1993, Modeling adaptive autonomous agents, Artificial Life 1993 Vol.1 MIT Press p135-162

Michael N. H. and Larry M. S. , 1999, Multiagent systems: a modern approach to distributed artificial intelligence, MIT Pres, Cambridge, MA, 79-120

Mylopoulos, J., Chung, L., and Yu, E., 1999, From object-oriented to goaloriented requirements analysis. Commun. ACM, 42, 31–37.

Morabito, J., Sack,I, Stohr,E. A., & Bhate, A., 2009 Designing flexible organizations, Global Journal of Flexible Systems Management 10.2

Pokahr, A. and Braubach, L., and Lamersdorf, W., 2003, Jadex: Implementing a BDI infrastructure for jade agents, EXP - in search of innovation (Special Issue on JADE), 3(3), 76–85.

Pokahr, A. and Braubach, L., and Lamersdorf, W., 2005, Jadex: A BDI reasoning engine, R. Bordini, M. Dastani, J. D. and Seghrouchni, A. E. F., editors, Multi-Agent Programming, Springer Science+Business Media Inc., USA. Book chapter, 149–174.

Prud'hommeaux, E. and Seaborne, A., 2007, SPARQL Query Language for RDF (Working Draft). Technical report, W3C.

Russel,S.J., and Norvig, P., 2009, Artificial Intelligence: a modern approach 3rd Edition,Prentice Hall, 1152 p

Tablado, A. and Illarramendi, A. and Bagües, M.I. and Bermudez, J. and Goni, A.,2004, Aingeru: an innovating system for tele assistance of elderly people Wooldridge, M., 2002, Introduction to MultiAgent Systems, John Wiley & Sons.

Wooldridge, M. and Jennings, N. R., 1995, Intelligent agents: Theory and practice, Knowledge Engineering Review, 10, 115–152.

Weiss G., 1999, Multiagents Systems: A moder approach to distributed artificial intelligence.”, MIT press

Wooldridge M. J. , 2000, Reasoning about Rational Agents, Intelligent Robots and Autonomous Agents Series, Cambridge, MA: The MIT Press

KAYNAKLAR DİZİNİ (devam)

Yu, E., 1995, Modelling strategic relationships for process reengineering, PhD thesis, University of Toronto, Department of Computer Science

ÖZGEÇMİŞ**KİŞİSEL BİLGİLER**

Adı Soyadı : Süleyman Kıvanç Ekici

Doğum Tarihi / Yeri: 20.03.1984 / Elbistan

EĞİTİM BİLGİLERİ

2010 – ... : **Yüksek Lisans**
Ege Üniversitesi, Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği A. D.

2005 – 2006 : **Lisans Öğrenci Değişim Programı**
Technische Hochschule Mittelhessen
Informatik

2003 – 2008 : **Lisans**
Ege Üniversitesi, Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü

KARİYER BİLGİLERİ

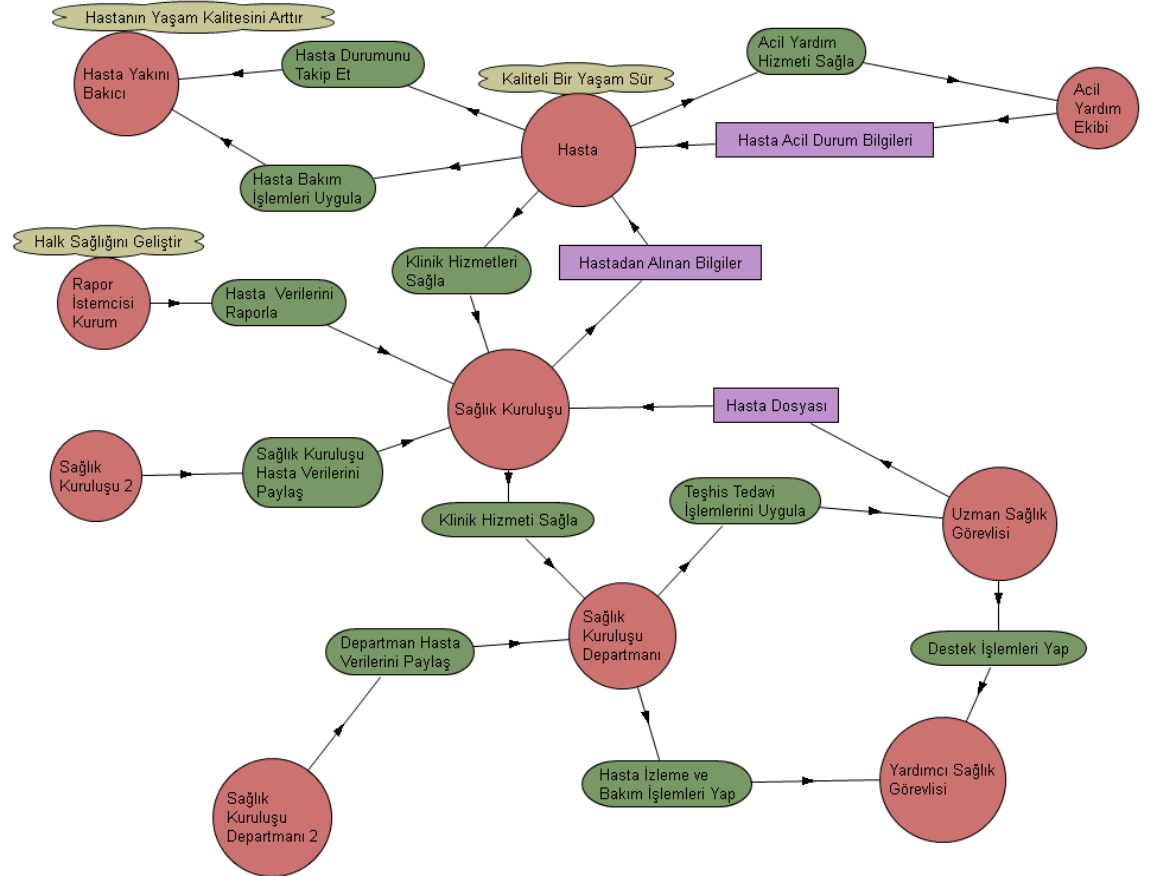
2008 – ... : **Genel Müdür**
ROATIS Teknoloji, İzmir

2012 – 2013 : **Öğretim Görevlisi**
Ege Üniversitesi, Mühendislik Fakültesi, Bilgisayar
Mühendisliği Bölümü, İzmir

EKLER

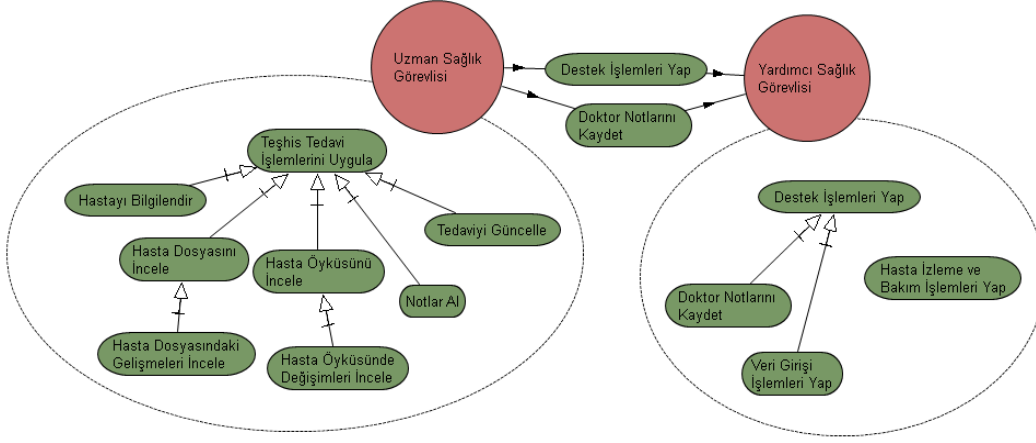
- Ek 1 Erken Gereksinim Analizi Şemaları
- Ek 2 Geç Gereksinim Analizi Şemaları
- Ek 3 Mimari Tasarım Şemaları
- Ek 4 Etmen Gerçekleştirim Kodları
- Ek 5 Restful Web Servis Gerçekleştirim Kodları

Ek 1 Erken Gereksinim Analizi Şemaları

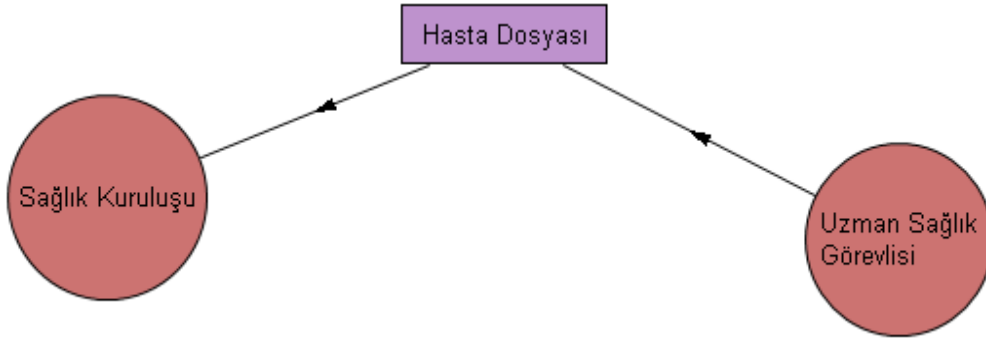


Şekil 6-1Ana Şema

Ek 1 Erken Gereksinim Analizi Şemaları (devam)

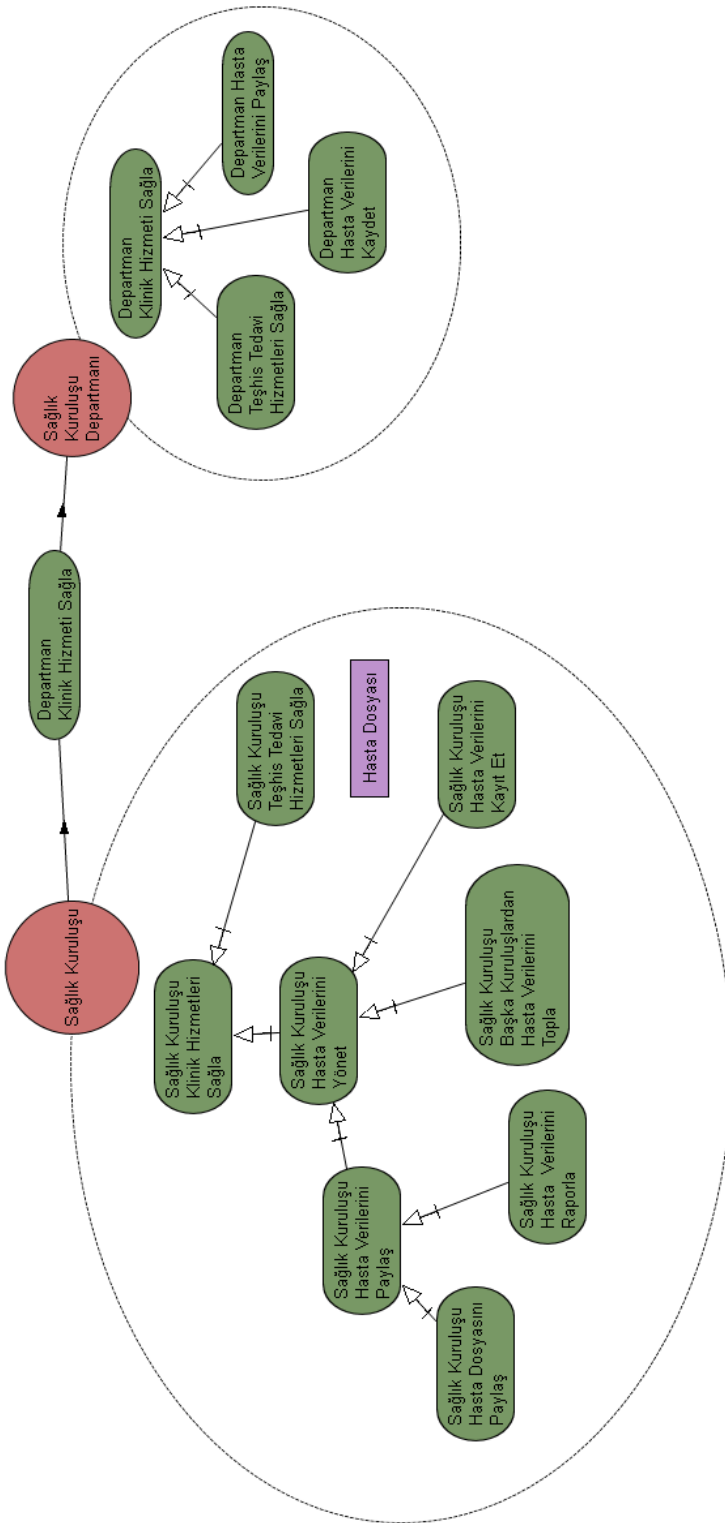


Şekil 6-2 Uzman Sağlık Görevlisi



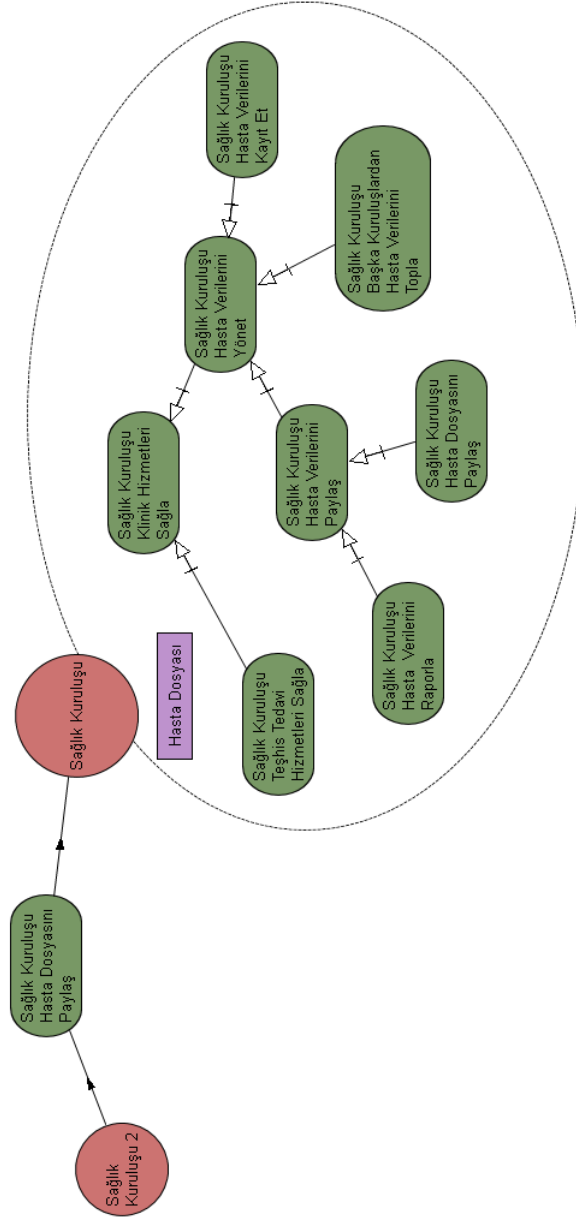
Şekil 6-3 Sağlık Kuruluşu Uzman Sağlık Görevlisi

Ek 1 Erken Gereksinim Analizi Şemaları Devamı



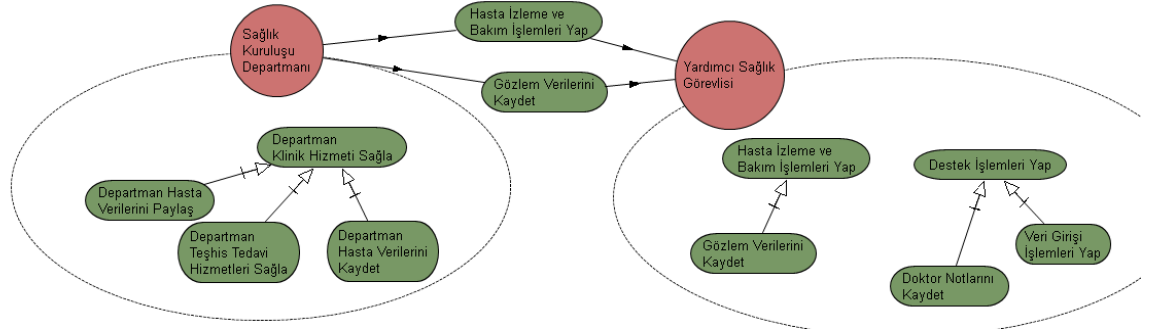
Şekil 6-4 Sağlık Kuruluşu ve Departman

Ek 1 Erken Gereksinim Analizi Şemaları (devam)

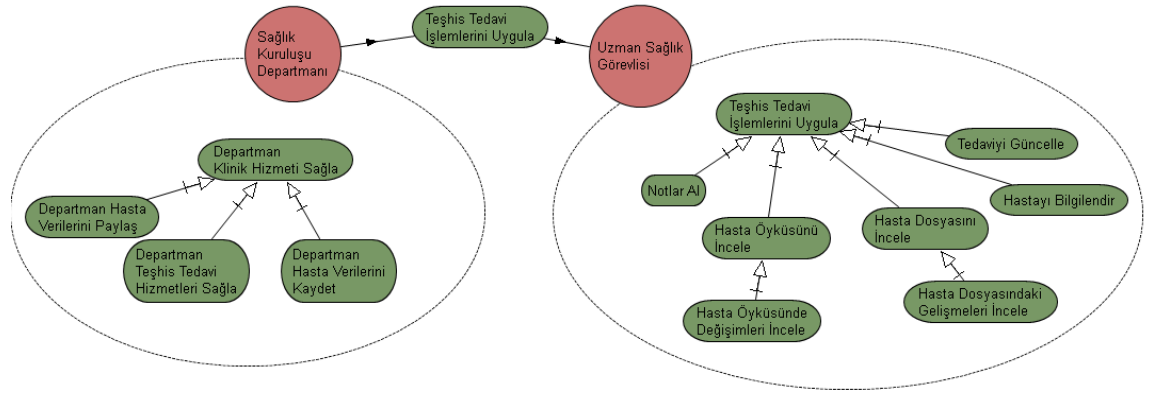


Şekil 6-5 Sağlık Kuruluşu

Ek 1 Erken Gereksinim Analizi Şemaları (devam)

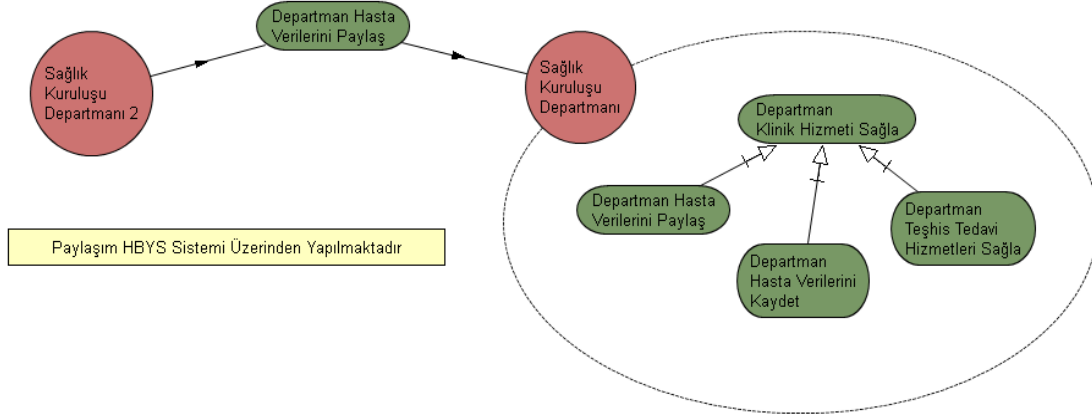


Şekil 6-6 Sağlık Kuruluşu Departmanı-Yardımcı Sağlık Görevlisi

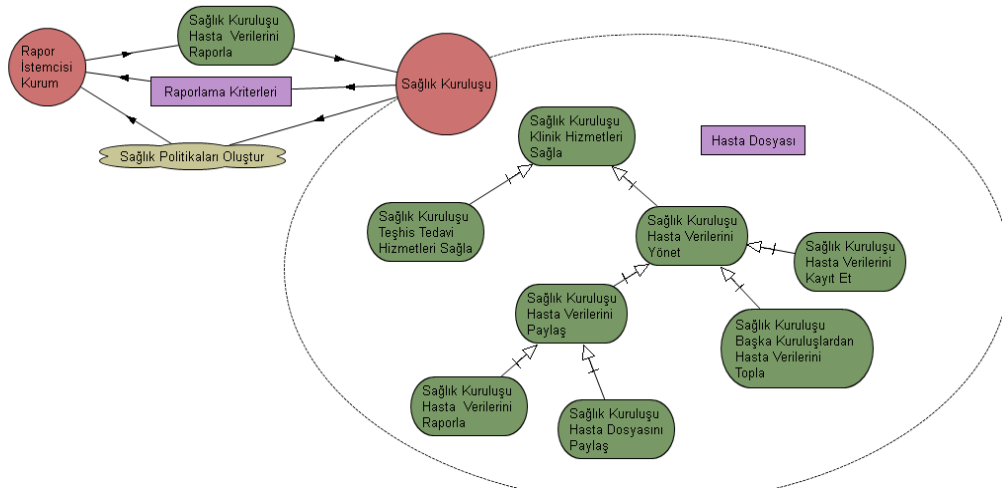


Şekil 6-7 Sağlık Kuruluşu Departmanı-Uzman Sağlık Görevlisi

Ek 1 Erken Gereksinim Analizi Şemaları (devam)

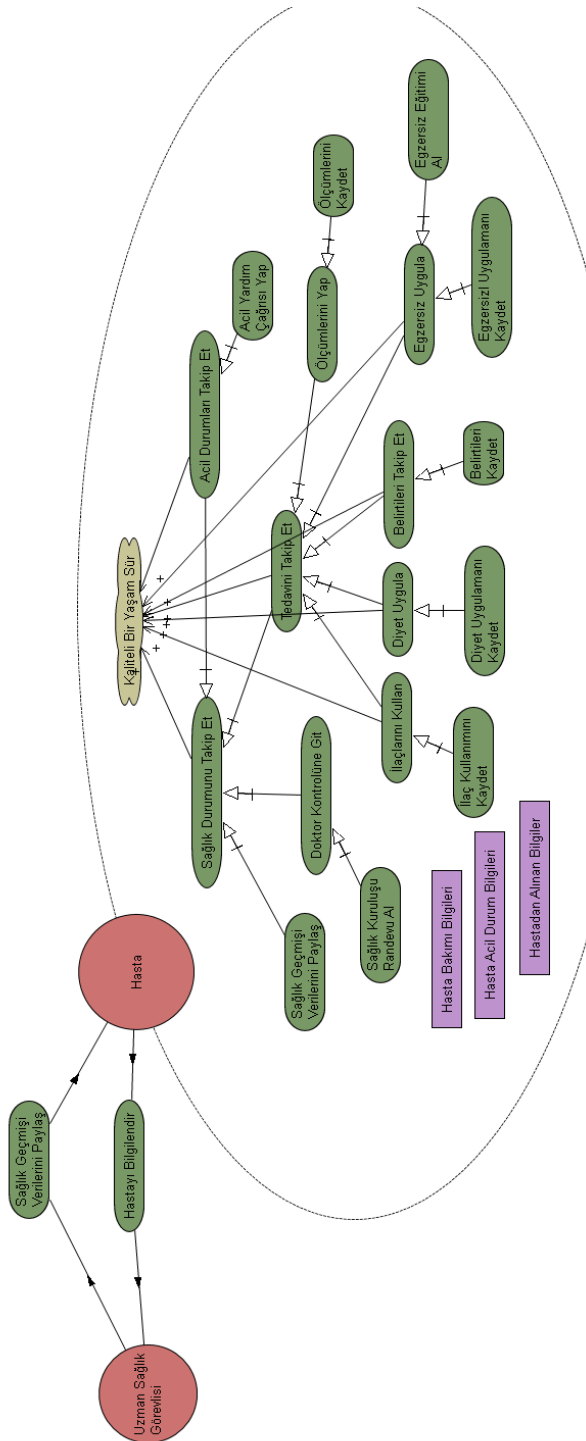


Şekil 6-8 Sağlık Kuruluşu Departmanı



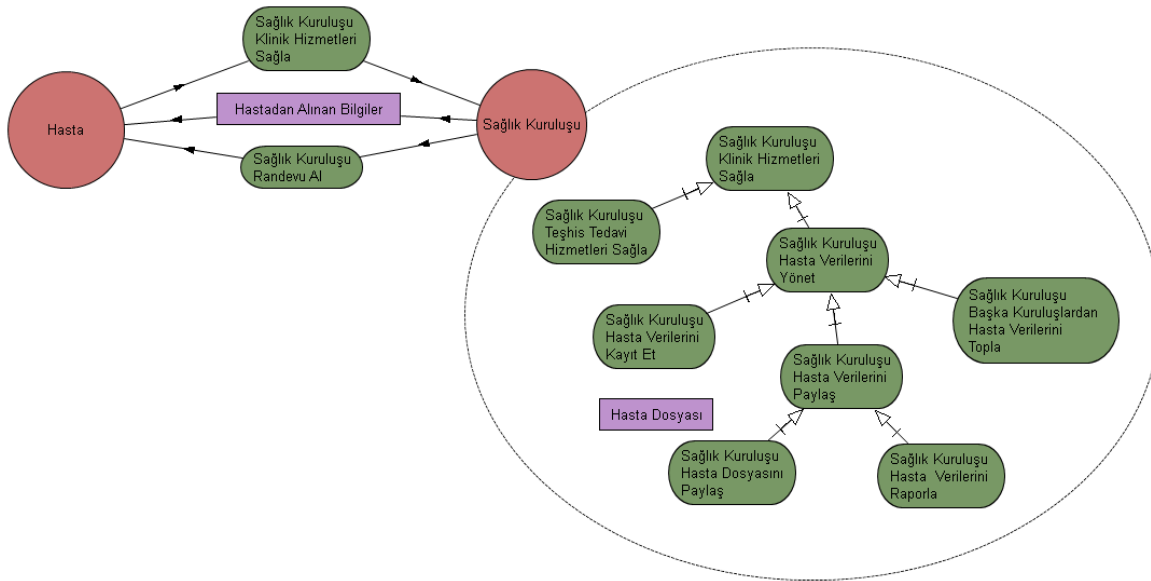
Şekil 6-9 Rapor İstemcisi-Sağlık Kuruluşu

Ek 1 Erken Gereksinim Analizi Şemaları (devam)

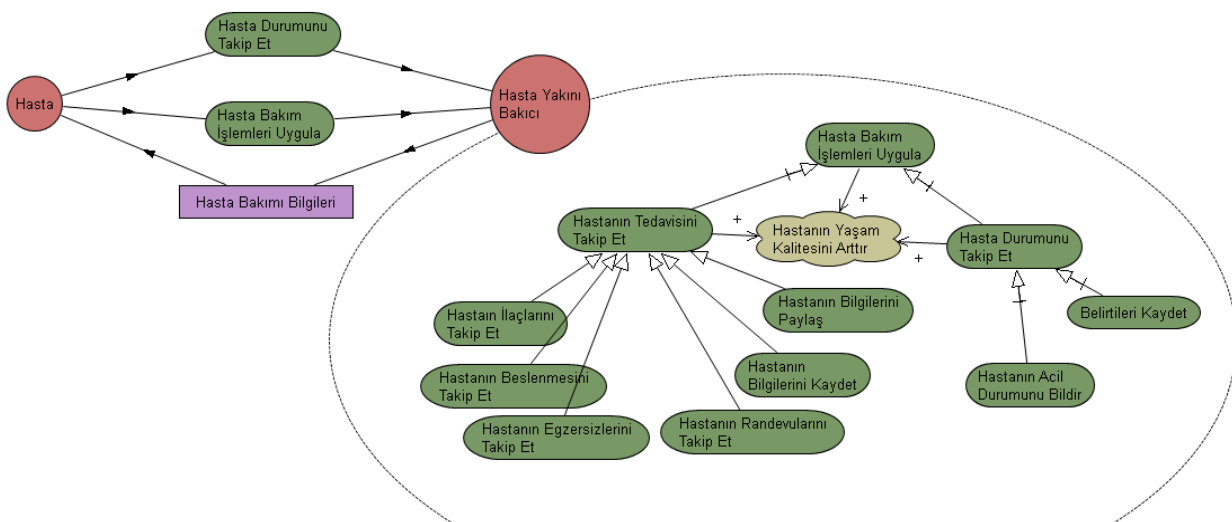


Şekil 6-10 Hasta

Ek 1 Erken Gereksinim Analizi Şemaları (devam)

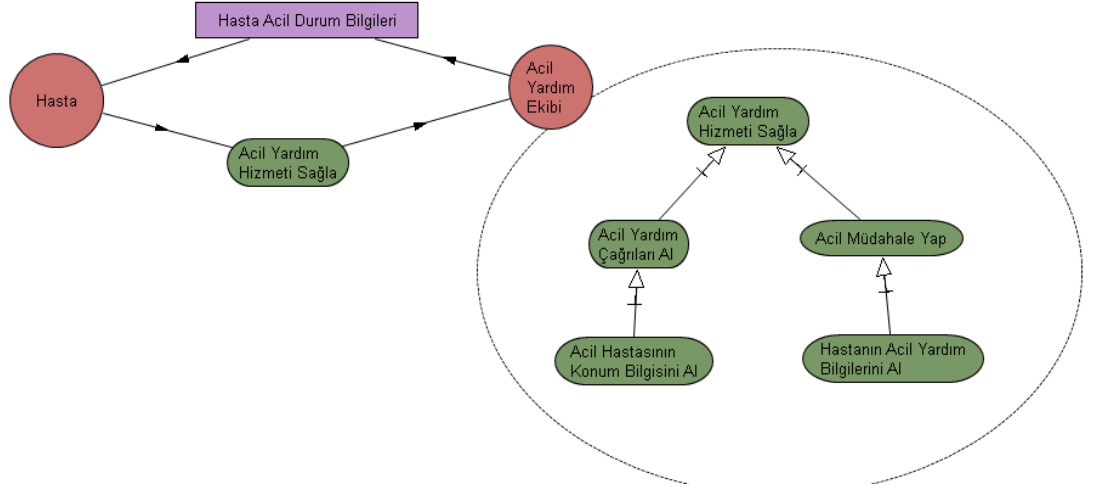


Şekil 6-11 Hasta-Sağlık Kuruluşu



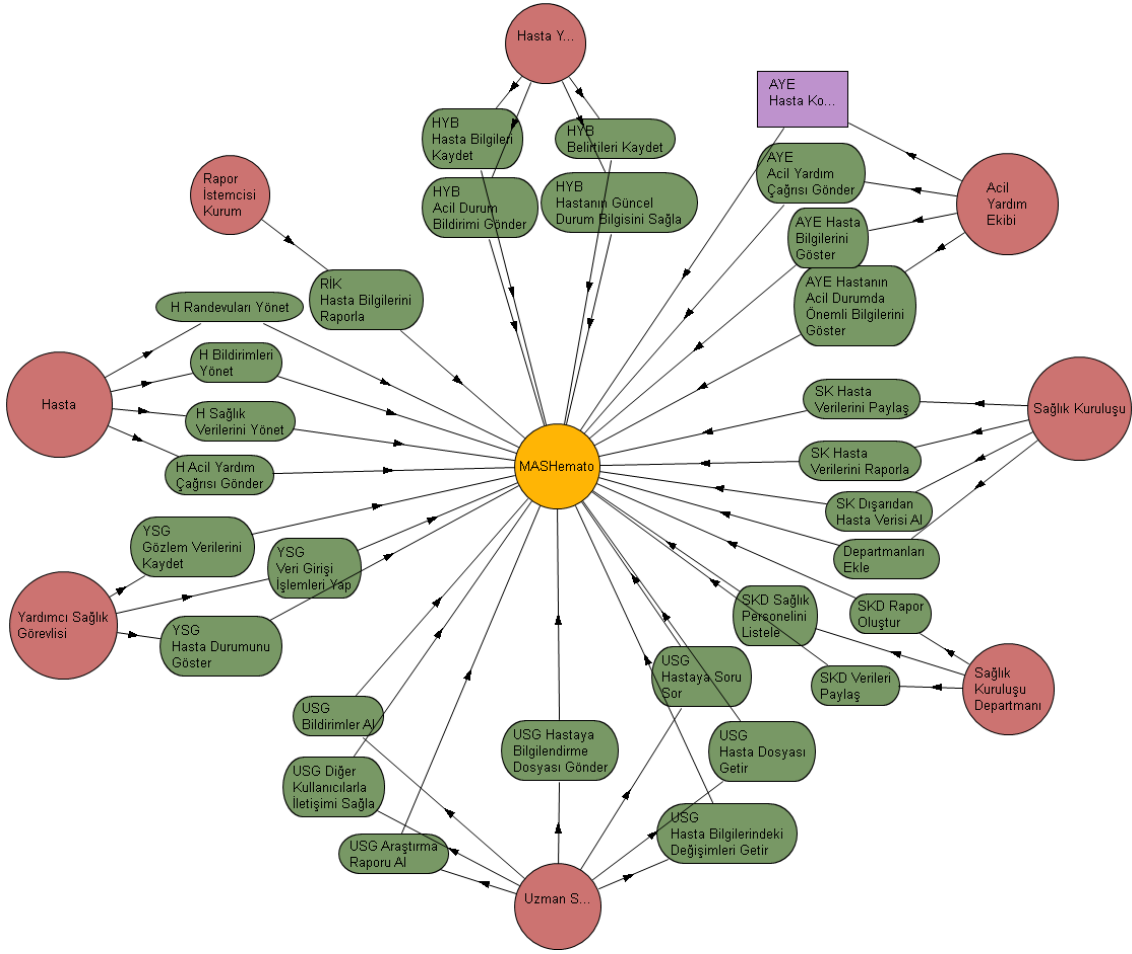
Şekil 6-12 Hasta Yakını Bakıcı

Ek 1 Erken Gereksinim Analizi Şemaları (devam)



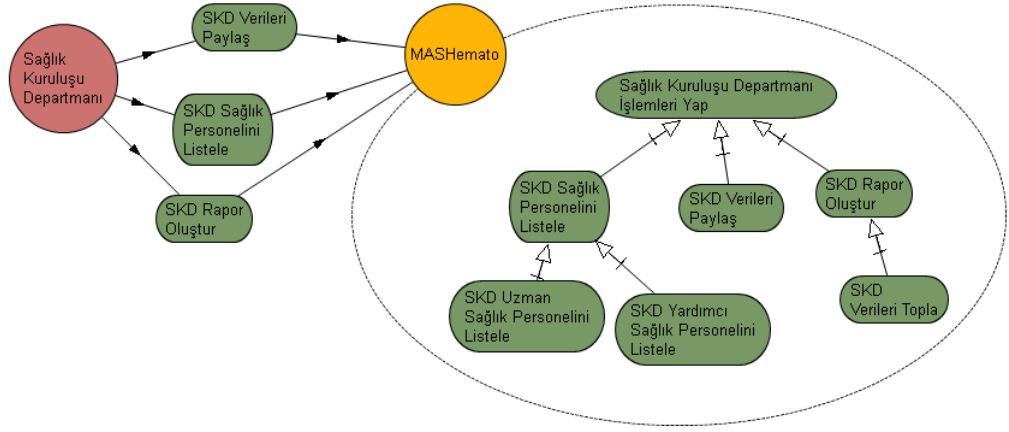
Şekil 6-13 Acil Yardım Ekibi

Ek 2 Geç Gereksinim Analizi Şemaları

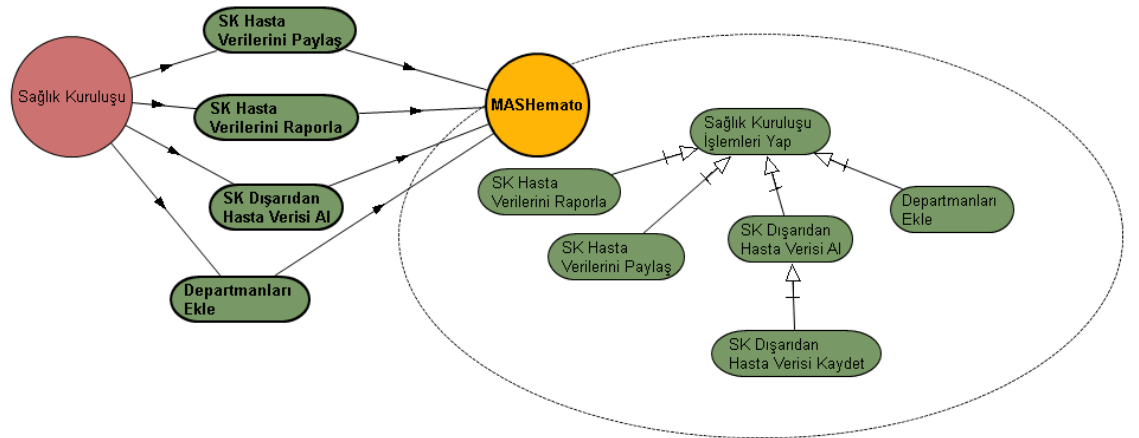


Şekil 6-14 MASHemato Geç Gereksinim Analizi

Ek 2 Geç Gereksinim Analizi Şemaları (devam)

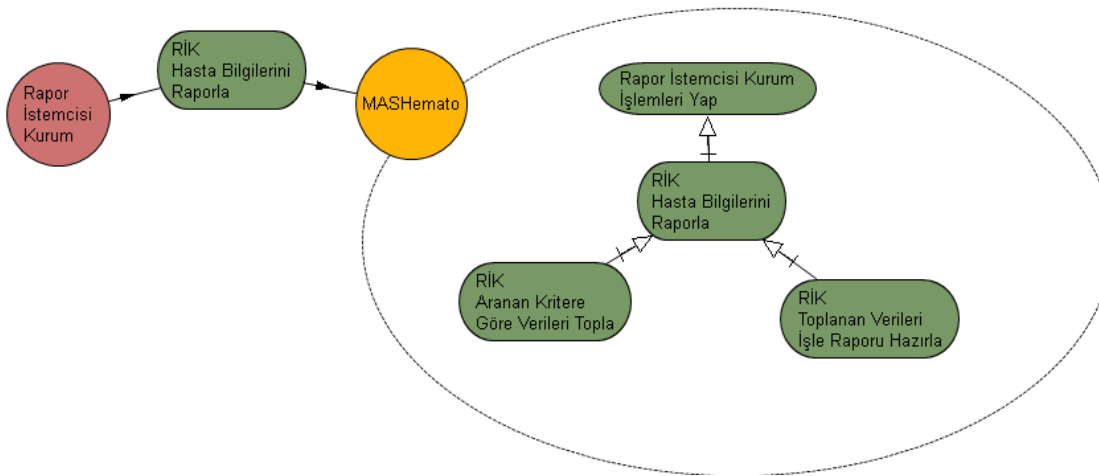


Şekil 6-15 MASHemato Sağlık Kuruluşu Departmanı

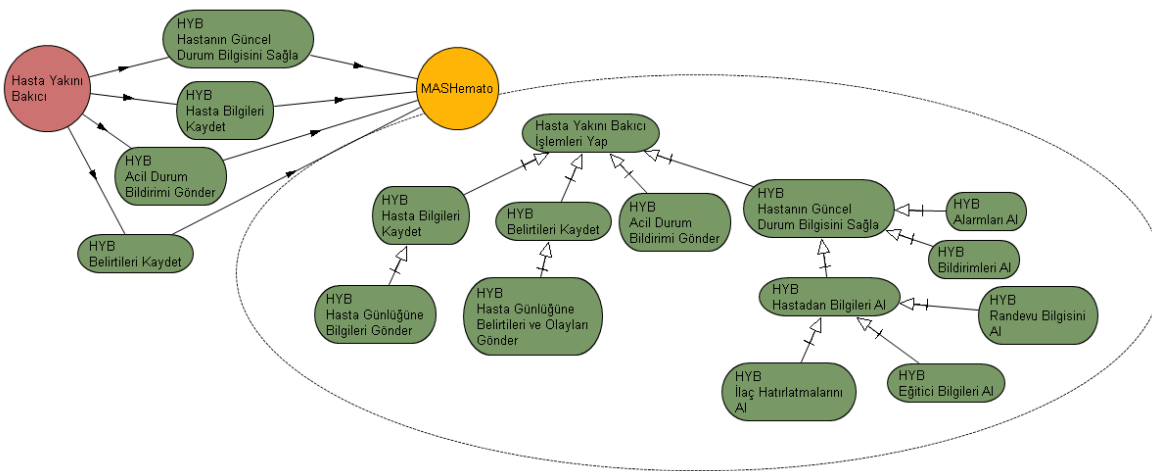


Şekil 6-16 MASHemato Sağlık Kuruluşu

Ek 2 Geç Gereksinim Analizi Şemaları (devam)

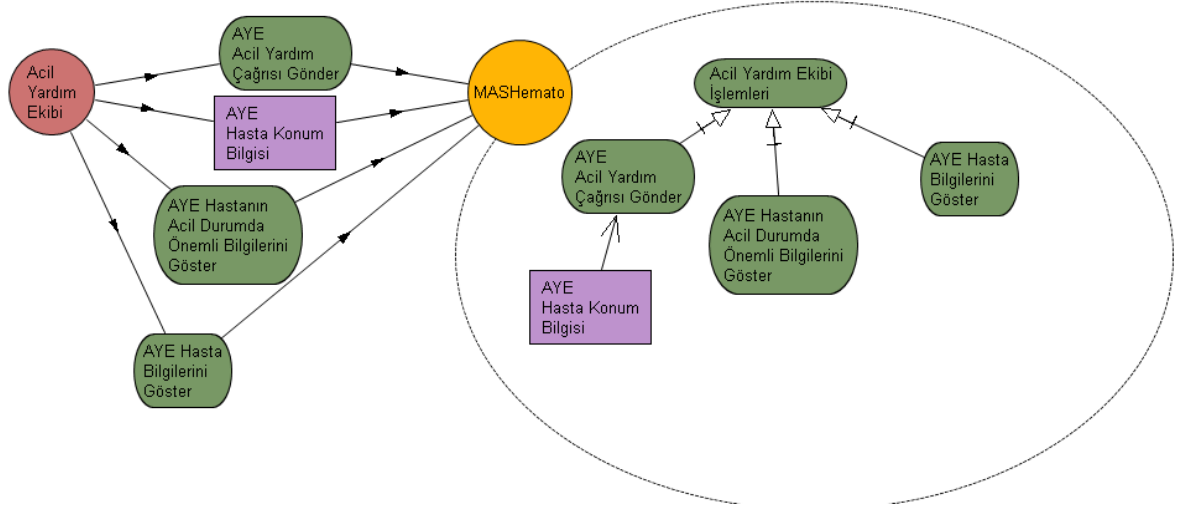


Şekil 6-17 MASHemato Rapor İstemcisi Kurum

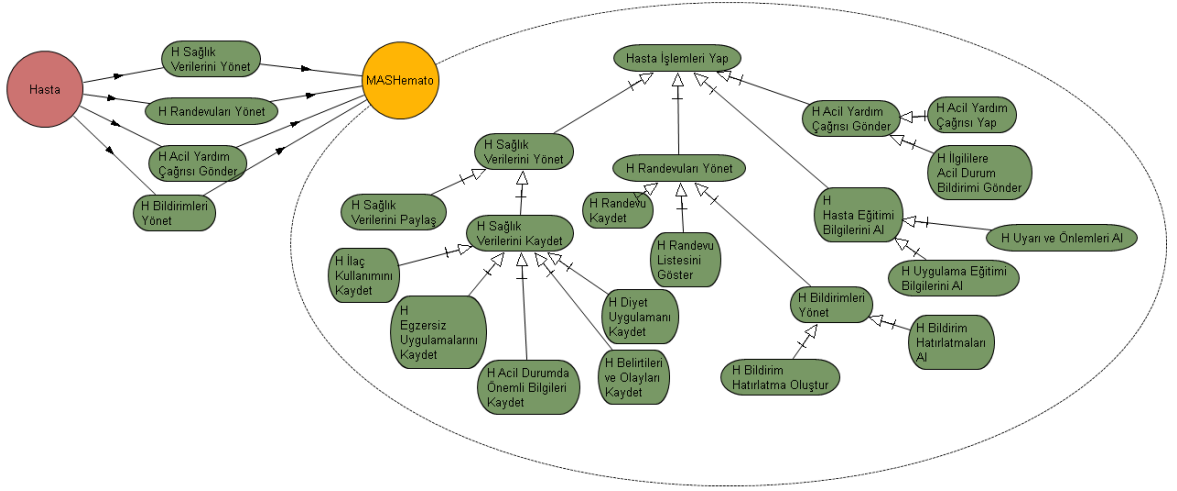


Şekil 6-18 MASHemato Hasta Yakını Bakıcı

Ek 2 Geç Gereksinim Analizi Şemaları (devam)

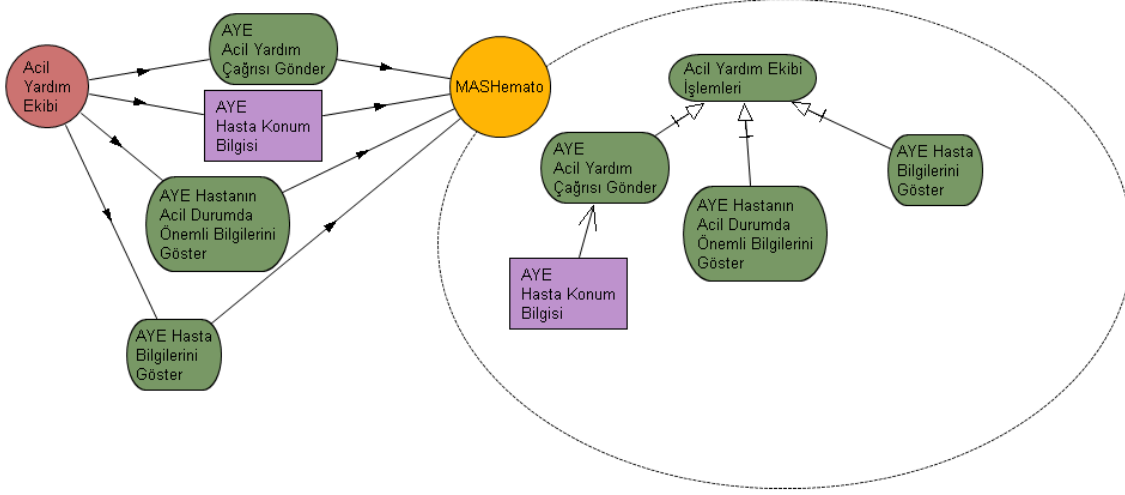


Şekil 6-19 MASHemato Acil Yardım Ekibi

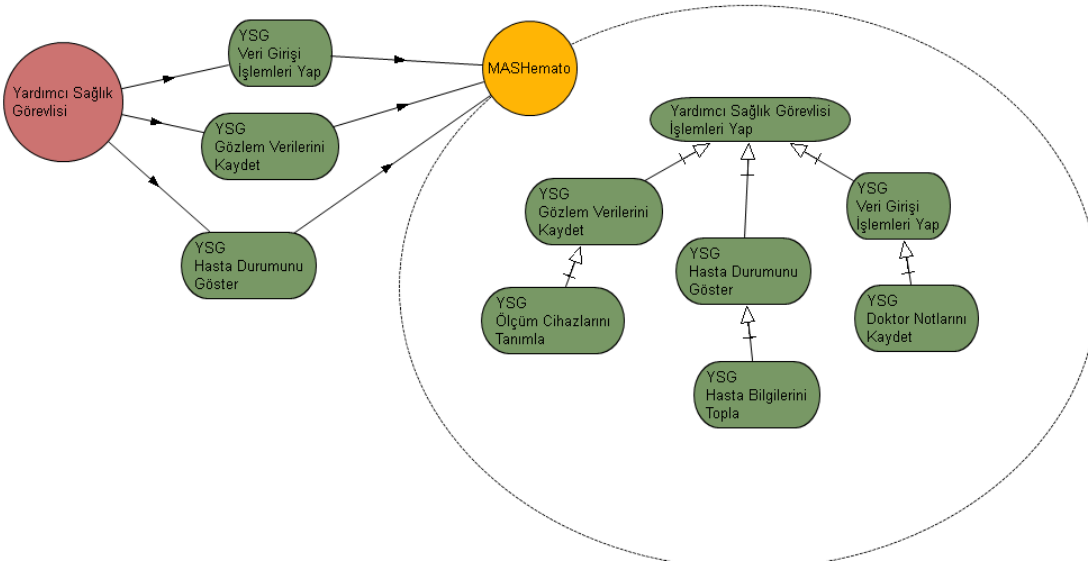


Şekil 6-20 MASHemato Hasta

Ek 2 Geç Gereksinim Analizi Şemaları (devam)

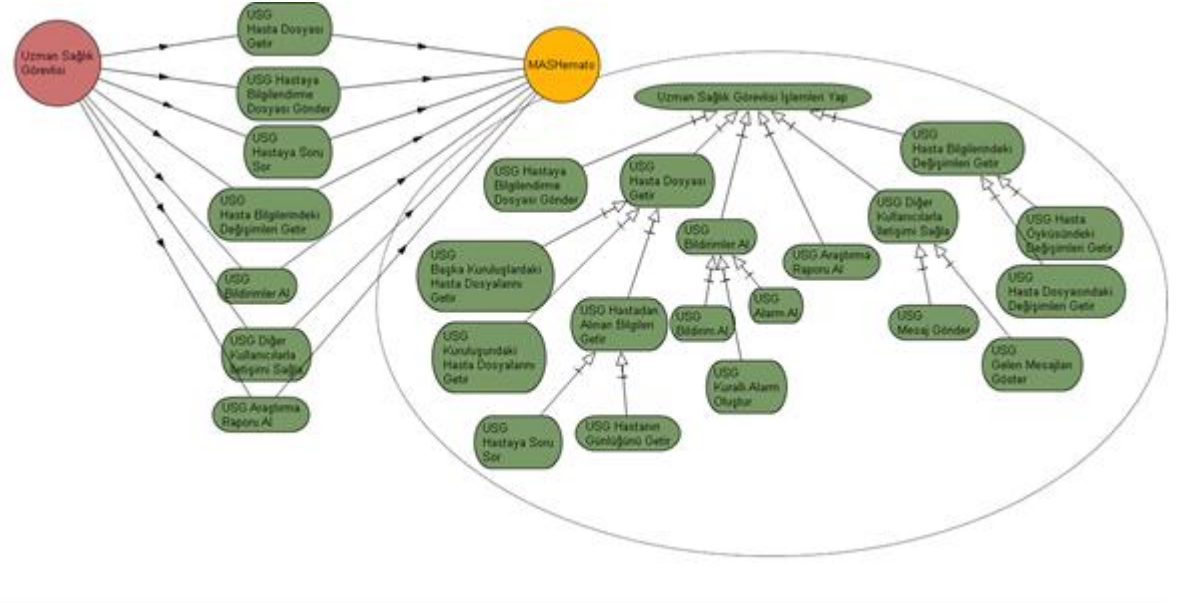


Şekil 6-21 MASHemato Acil Yardım Ekibi



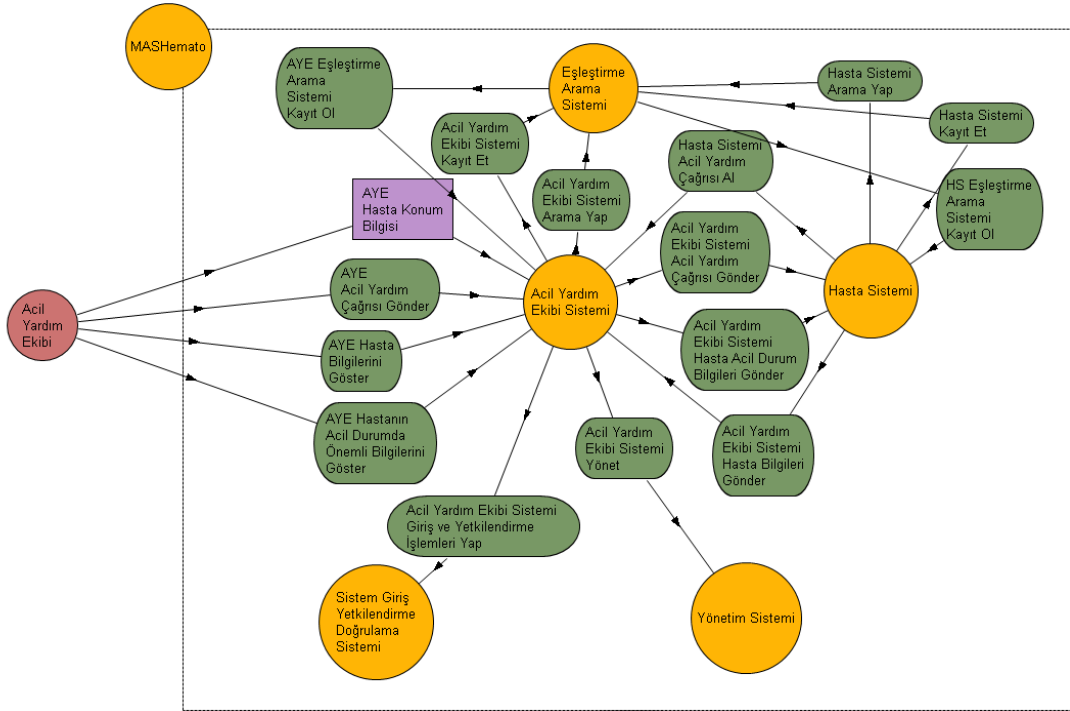
Şekil 6-22 MASHemato Yardımcı Sağlık Görevlisi

Ek 2 Geç Gereksinim Analizi Şemaları (devam)

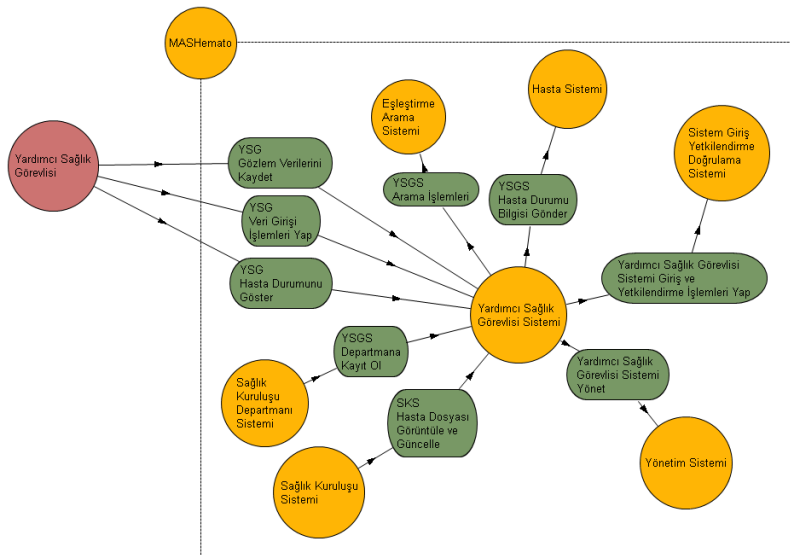


Şekil 6-23 MASHemato Uzman Sağlık Görevlisi

Ek 3 Mimari Tasarım Şemaları

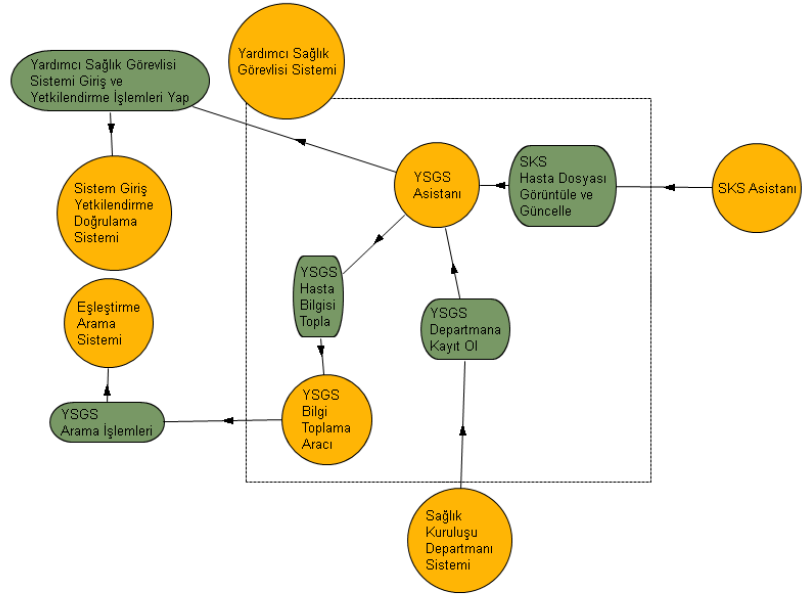


Şekil 6-24 Mimari Tasarım Acil Yardım Ekibi

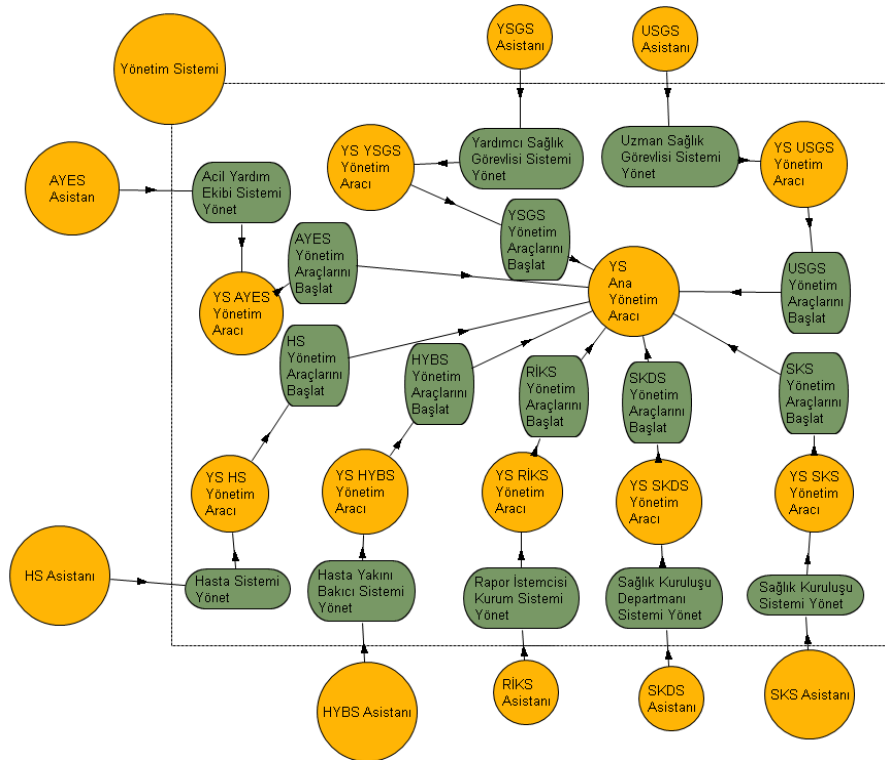


Şekil 6-25 Mimari Tasarım Yardımcı Sağlık Görevlisi

Ek 3 Mimari Tasarım Şemaları (devam)

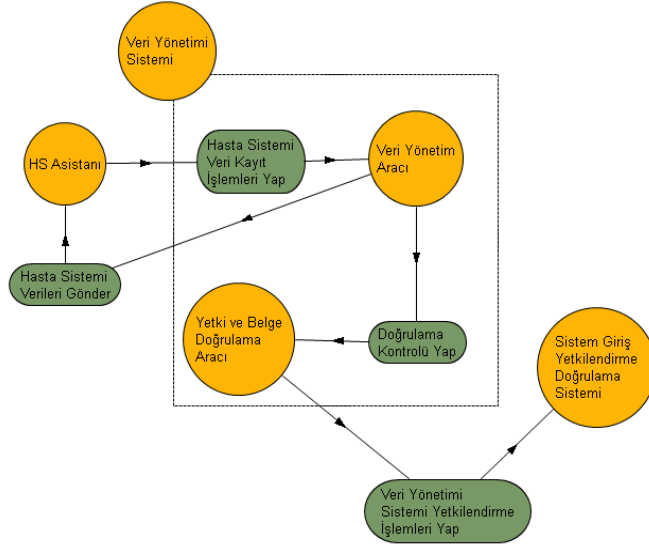


Şekil 6-26 Mimari Tasarım Yardımcı Sağlık Görevlisi Sistemi

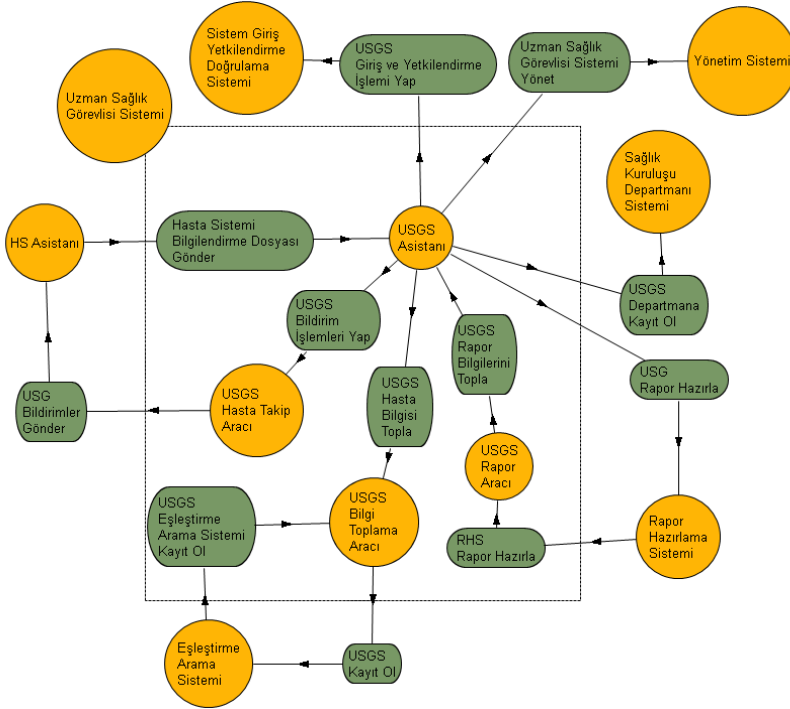


Şekil 6-27 Mimari Tasarım Yönetim Sistemi

Ek 3 Mimari Tasarım Şemaları (devam)

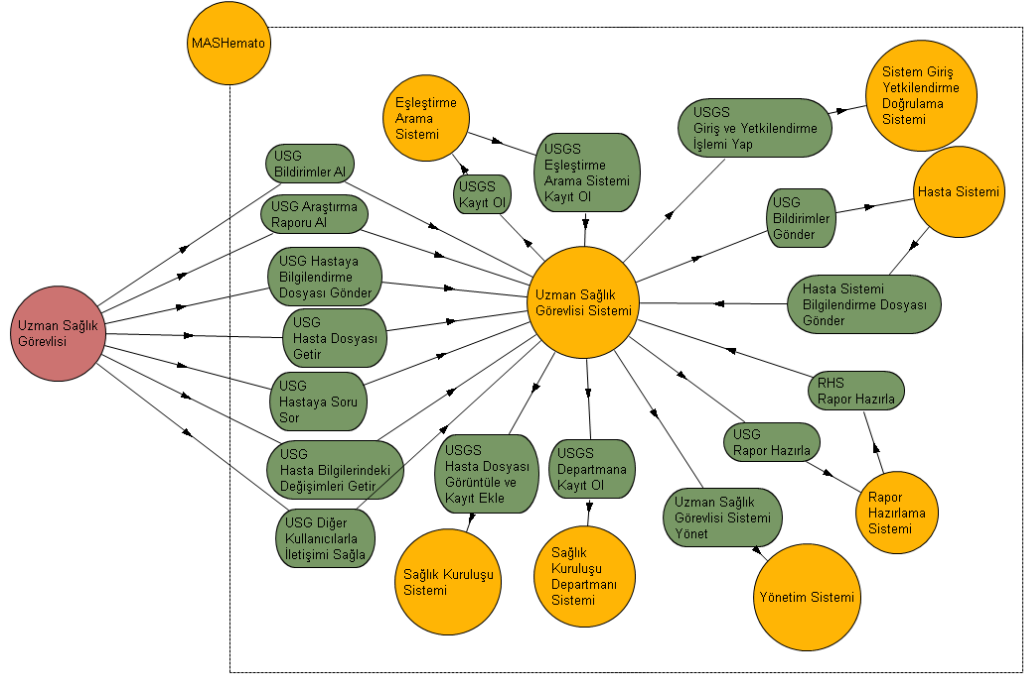


Şekil 6-28 Veri Yönetimi Sistemi



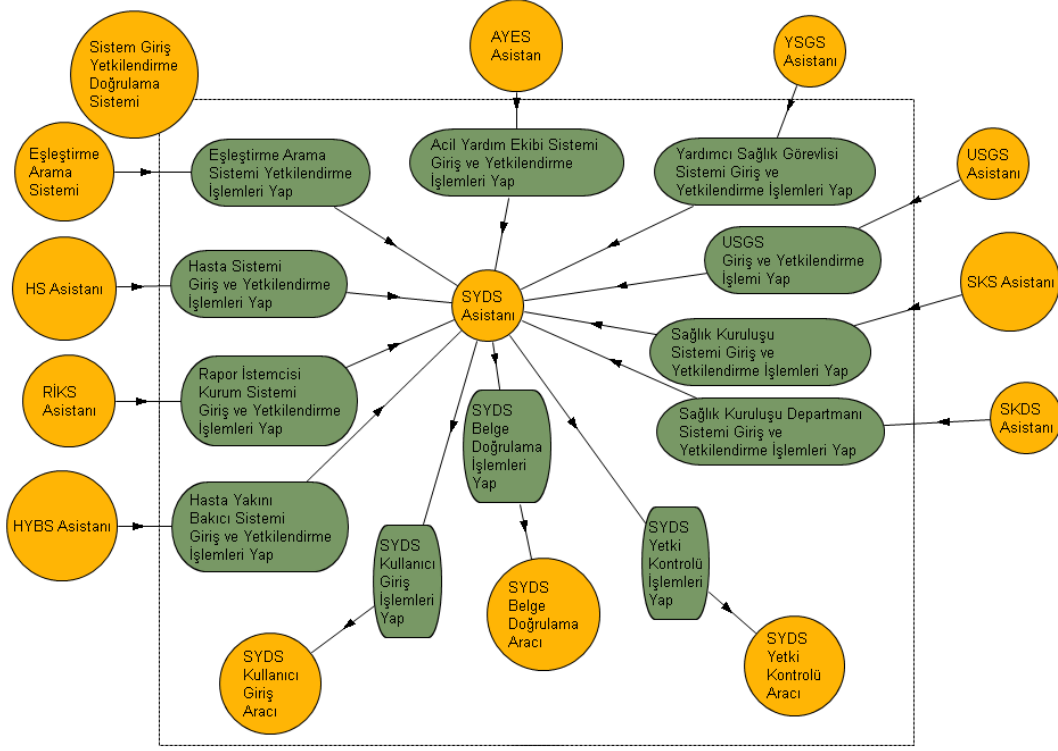
Şekil 6-29 Uzman Sağlık Görevlisi Sistemi

Ek 3 Mimari Tasarım Şemaları (devam)



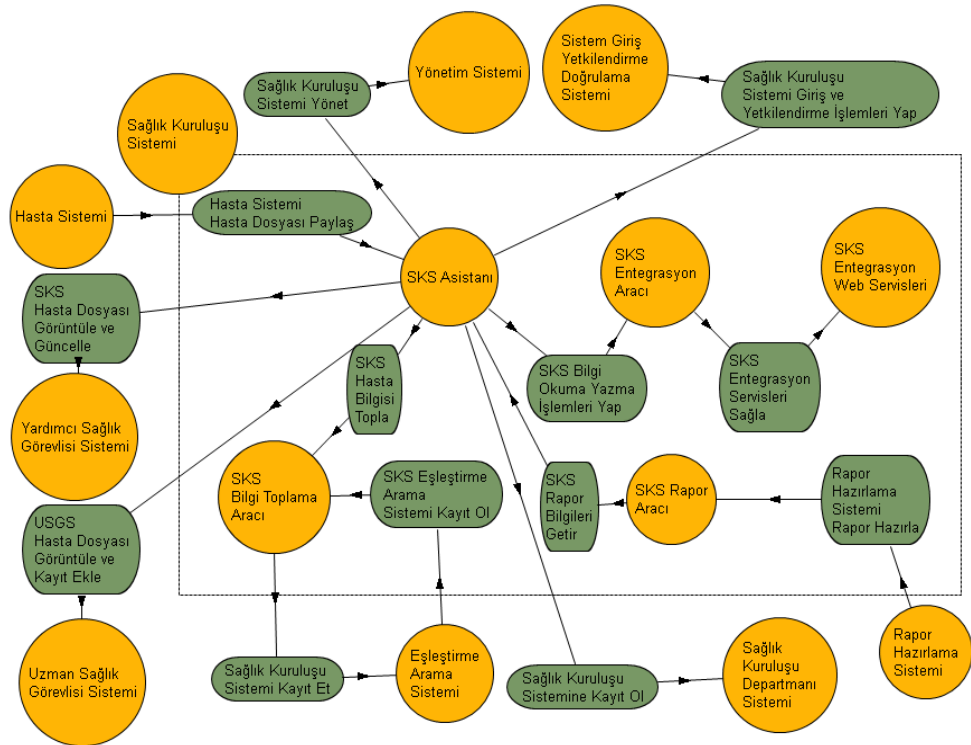
Şekil 6-30 Mimari Tasarım Uzman Sağlık Görevlisi

Ek 3 Mimari Tasarım Şemaları (devam)



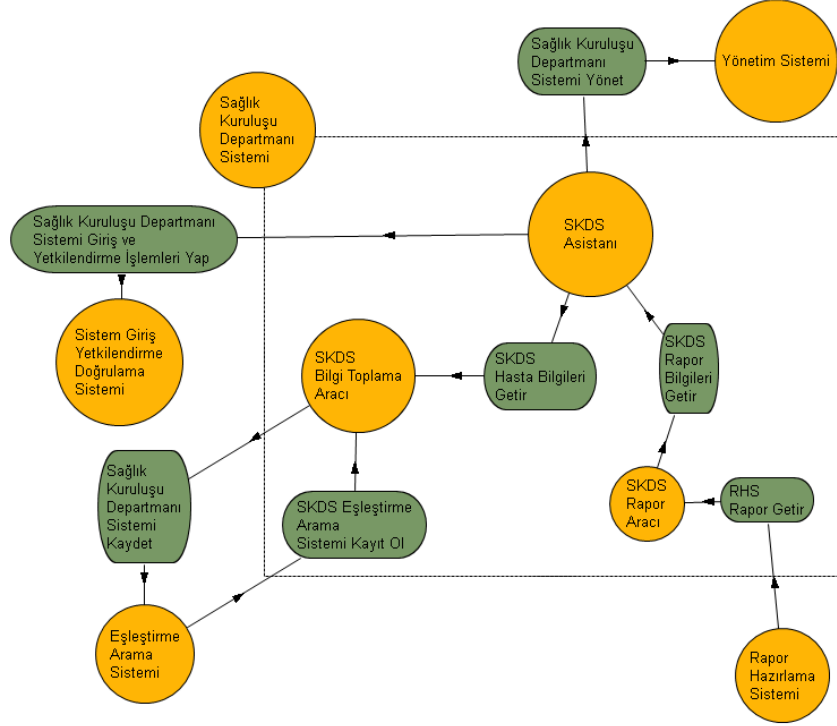
Şekil 6-31 Sistem Giriş Yetkilendirme Sistemi

Ek 3 Mimari Tasarım Şemaları (devam)



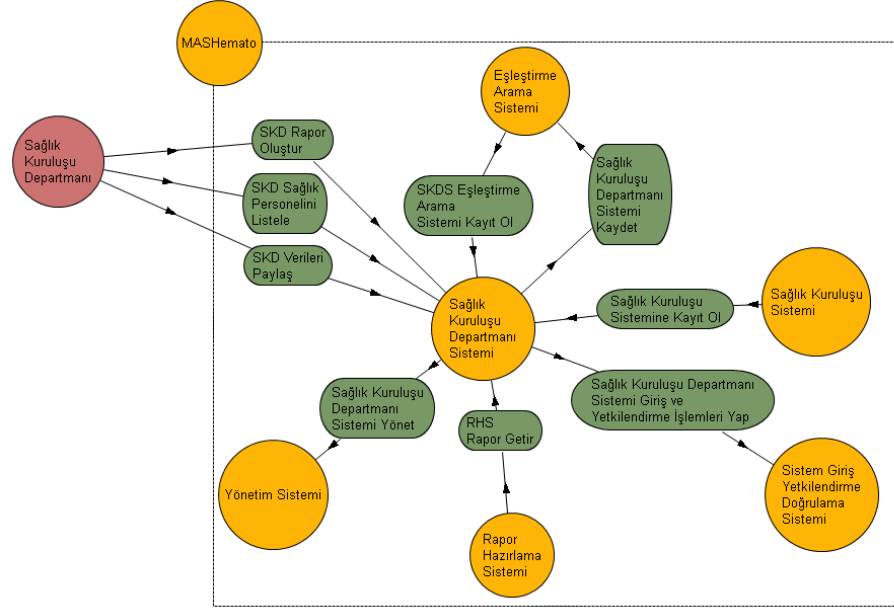
Şekil 6-32 Sağlık Kuruluşu Sistemi

Ek 3 Mimari Tasarım Şemaları (devam)



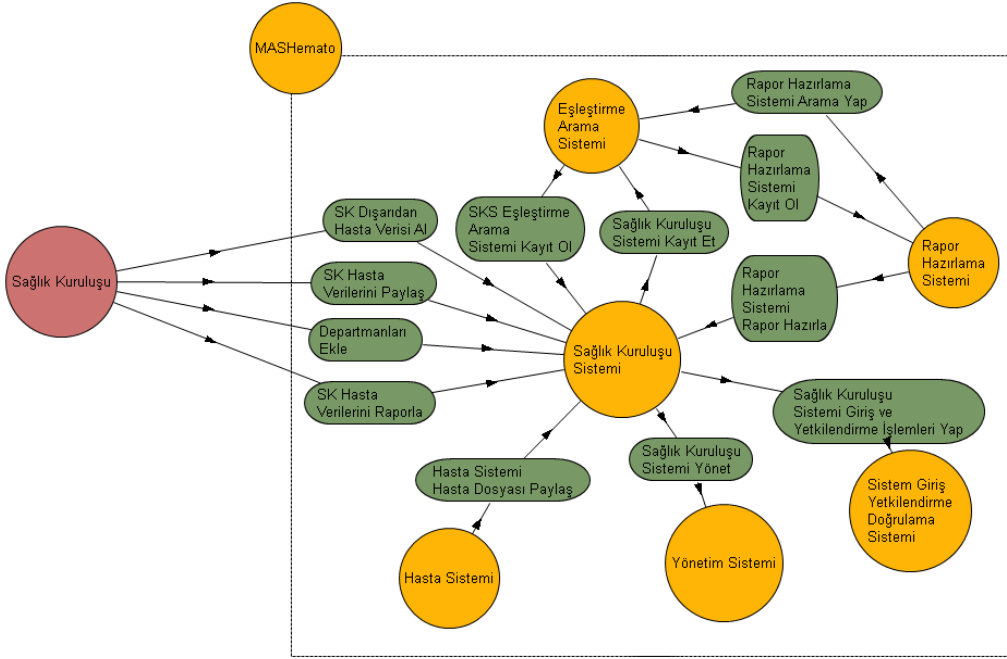
Şekil 6-33 Sağlık Kuruluşu Departmanı Sistemi

Ek 3 Mimari Tasarım Şemaları (devam)



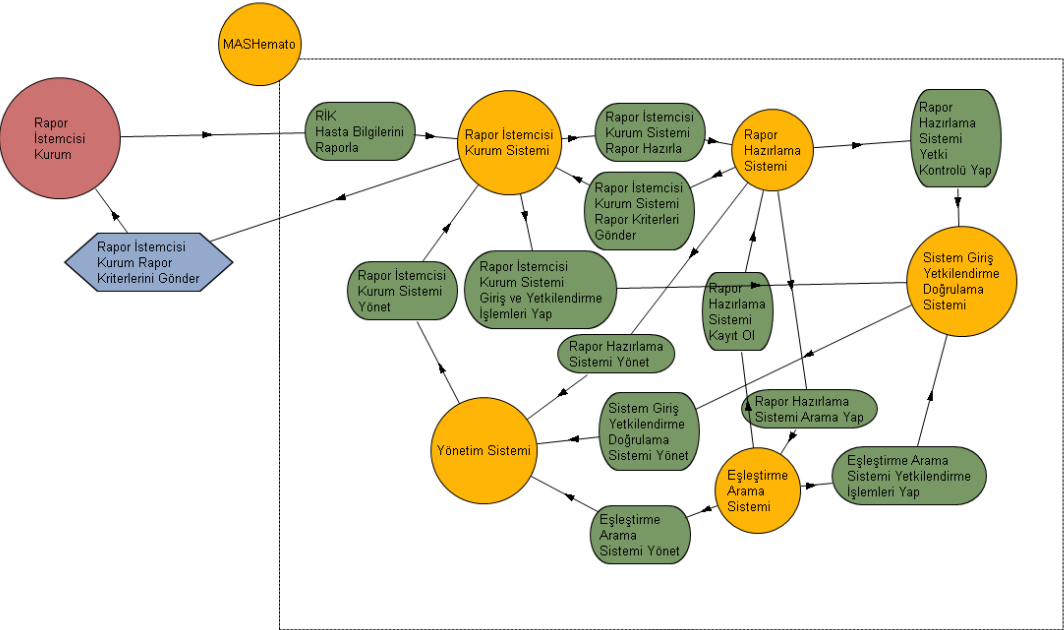
Şekil 6-34 Mimari Tasarım Sağlık Kuruluşu Departmanı

Ek 3 Mimari Tasarım Şemaları (devam)



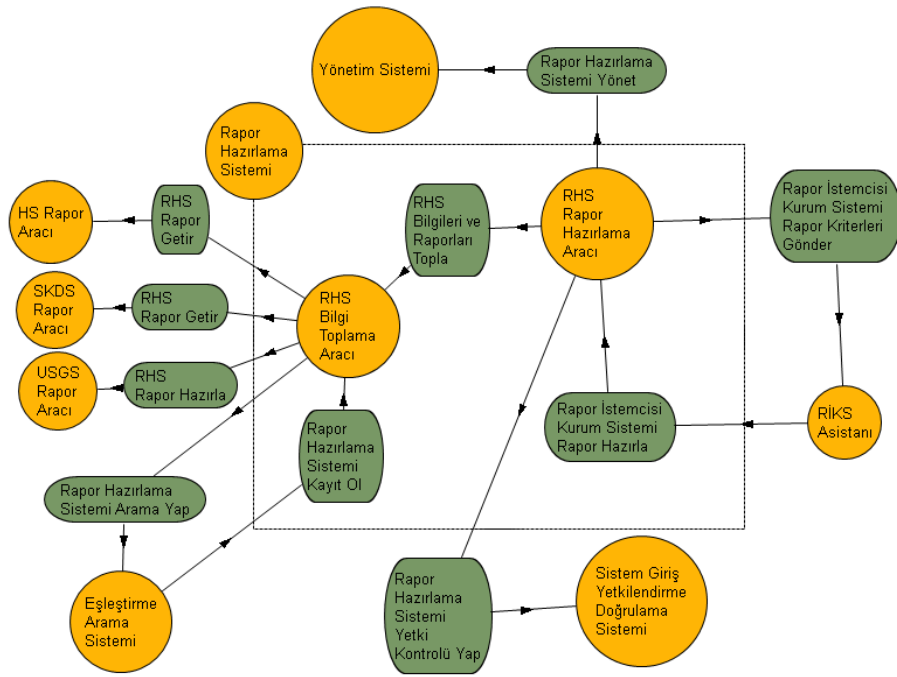
Şekil 6-35 Mimari Tasarım Sağlık Kuruluşu

Ek 3 Mimari Tasarım Şemaları (devam)



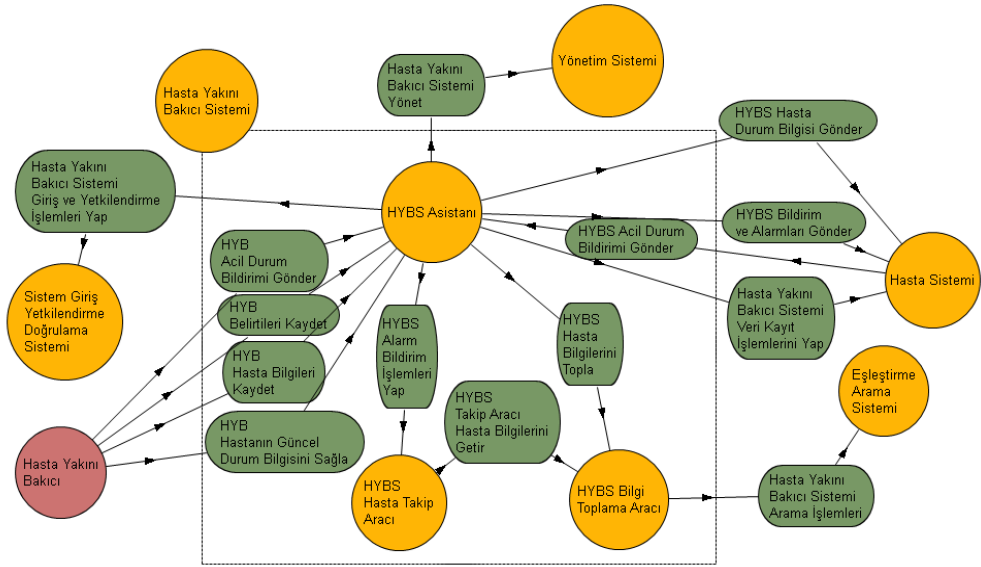
Şekil 6-36 Mimari Tasarım Rapor İstemcisi Kurum

Ek 3 Mimari Tasarım Şemaları (devam)

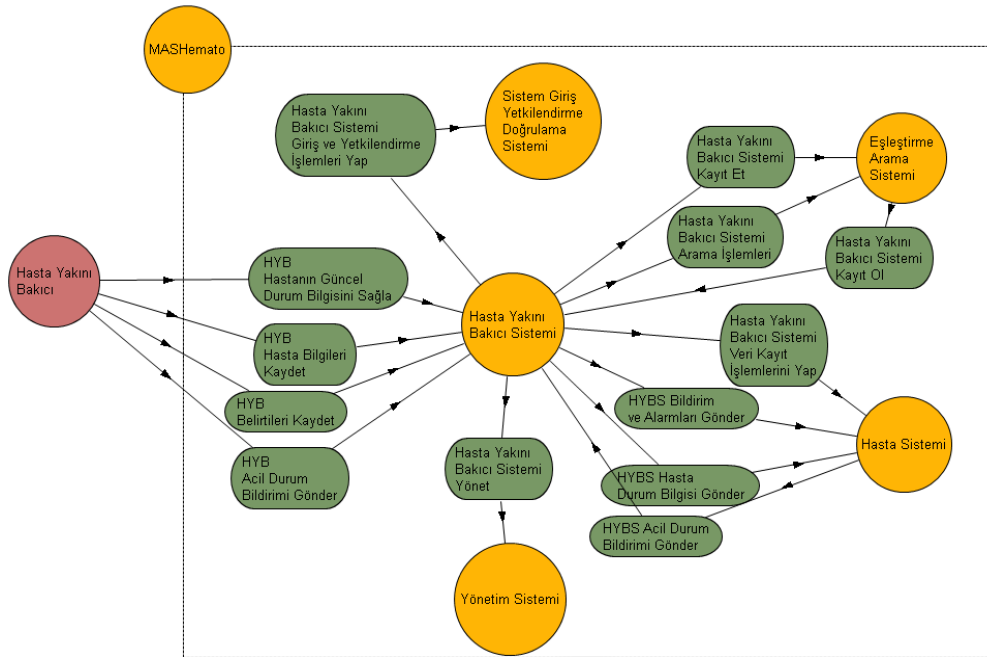


Şekil 6-37 Rapor Hazırlama Sistemi

Ek 3 Mimari Tasarım Şemaları (devam)

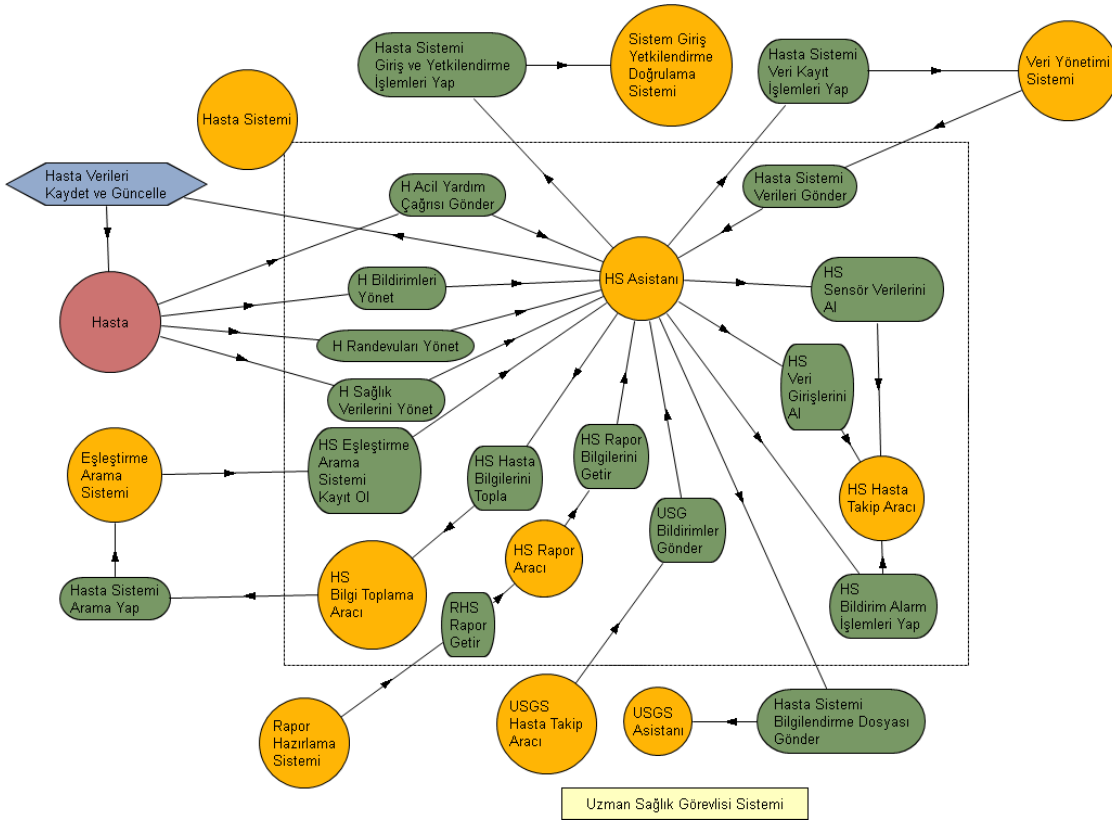


Şekil 6-38 Hasta Yakını Bakıcı Sistemi



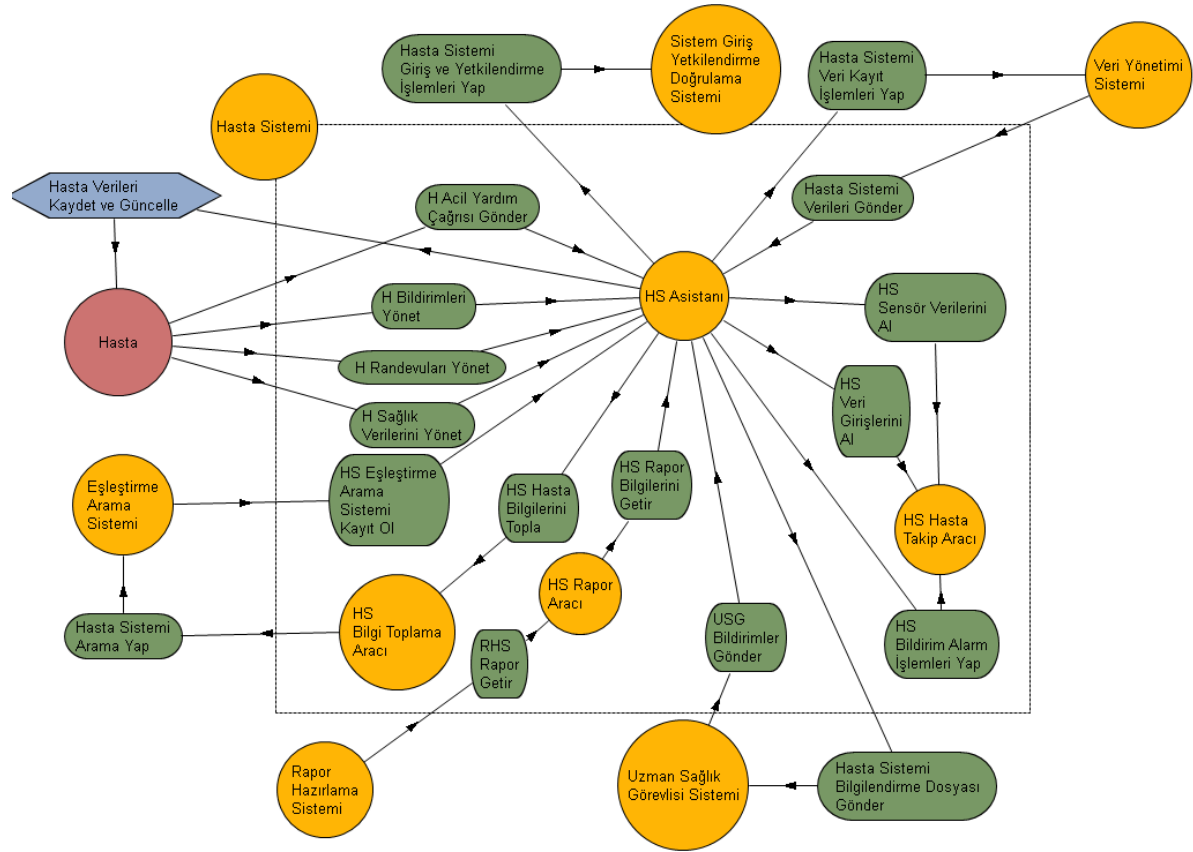
Şekil 6-39 Mimari Tasarım Hasta Yakını Bakıcı

Ek 3 Mimari Tasarım Şemaları (devam)



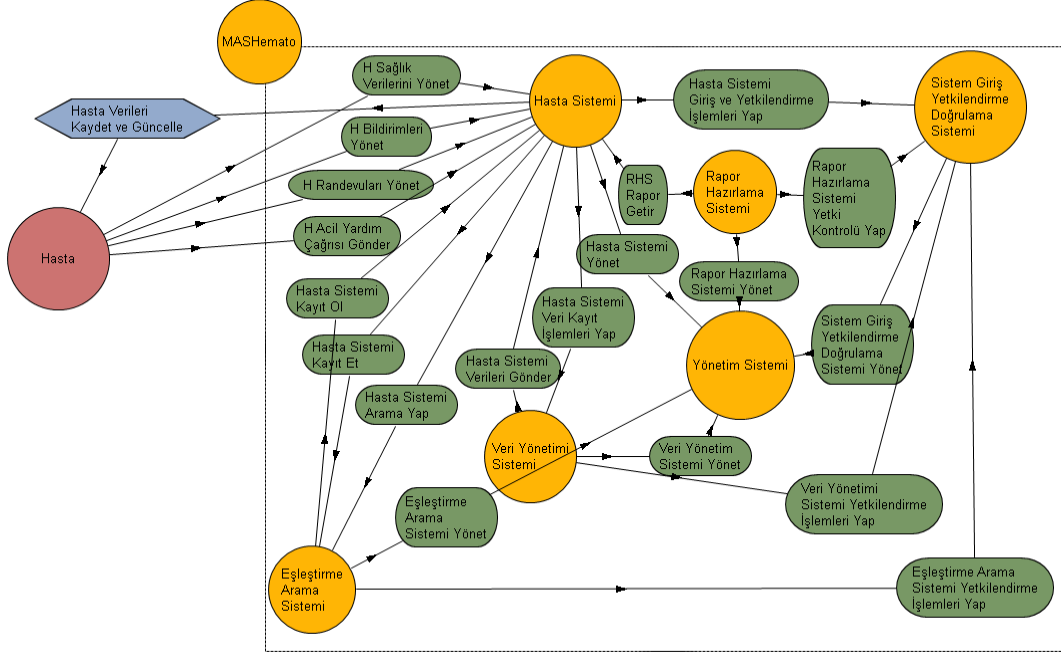
Şekil 6-40 Hasta Sistemi

Ek 3 Mimari Tasarım Şemaları (devam)



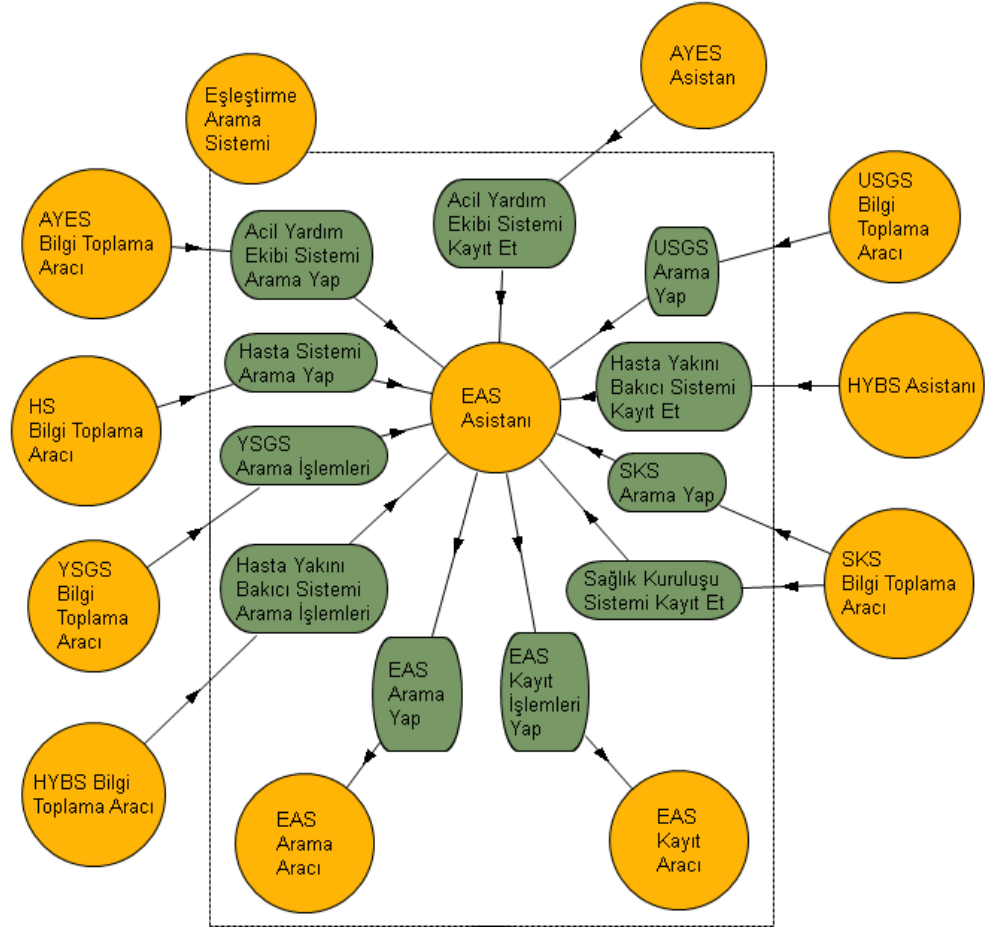
Şekil 6-41 Hasta Sistemi 2

Ek 3 Mimari Tasarım Şemaları (devam)



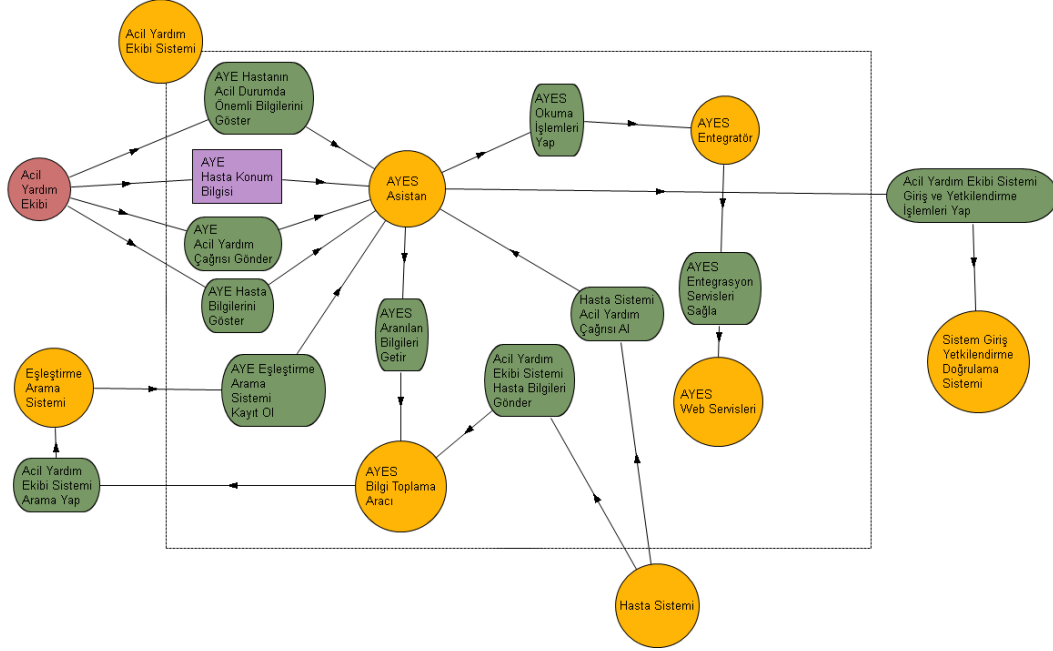
Şekil 6-42 Mimari Tasarım Hasta

Ek 3 Mimari Tasarım Şemaları (devam)



Şekil 6-43 Eşleştirme ve Arama Sistemi

Ek 3 Mimari Tasarım Şemaları (devam)



Şekil 6-44 Acil Yardım Ekibi Sistemi

Ek4 Etmen Gerçekleştirim Kodları

```
<agent xmlns="http://jadex.sourceforge.net/jadex"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://jadex.sourceforge.net/jadex
    http://jadex.sourceforge.net/jadex-0.96.xsd"
  name="HS_Asistanı"
  package="AgentRole_HS_Asistanı">

  <imports>
    <import>jadex.util.*</import>
    <import>jadex.adapter.fipa.*</import>
    <import>jadex.runtime.*</import>
    <import>java.util.logging.*</import>
    <import>AgentRole_HS_Asistanı.plans.*</import>
    <import>AgentRole_HS_Asistanı.util.kl.plans.*</import>
    <import>AgentRole_HS_Asistanı.util.kl.components.*</import>
  </imports>

  <beliefs>
    <!-- The belief contains the tropos hierarchy as facts. -->
    <beliefset name="goals" class="TGoal">

      <fact>Components.createGoal("HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol","ME")</fact>
    >

      <fact>Components.createGoal("HS_Rapor_Bilgilerini_Getir","ME")</fact>

      <fact>Components.createGoal("H_Sağlık__Verilerini_Yönet","AND")</fact>

      <fact>Components.createGoal("H_Acil_Yardım__Çağrısı_Gönder","ME")</fact>

      <fact>Components.createGoal("USG_Bildirimler_Gönder","ME")</fact>

      <fact>Components.createGoal("Hasta_Sistemi_Verileri_Gönder","ME")</fact>
      <fact>Components.createGoal("H_Bildirimleri_Yönet","ME")</fact>
      <fact>Components.createGoal("H_Randevuları_Yönet","ME")</fact>
    </beliefset>
    <beliefset name="softgoals" class="TSoftgoal">
    </beliefset>
    <beliefset name="decomp" class="TLink">
      <fact>new
TLink("H_Sağlık__Verilerini_Yönet","Hasta_Sistemi_Verileri_Gönder",1)</fact>
      <fact>new
TLink("H_Sağlık__Verilerini_Yönet","H_Randevuları_Yönet",1)</fact>
      <fact>new
TLink("H_Sağlık__Verilerini_Yönet","H_Bildirimleri_Yönet",1)</fact>
      <fact>new
TLink("H_Sağlık__Verilerini_Yönet","HS_Rapor_Bilgilerini_Getir",1)</fact>
    </beliefset>
    <beliefset name="meansend" class="TLink">
      <fact>new
TLink("H_Randevuları_Yönet","HSA_Randevuları_Görüntüle")</fact>
      <fact>new
TLink("HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol","HSA_Arama_Sistemine_Kayıt_Ol")</fact>
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
<fact>new
TLink("H_Acil_Yardim__Çağrısı_Gönder","HSA_Acil_Yardim_Çağrısı_Yap")</fact>
<fact>new
TLink("H_Bildirimleri_Yönet","H_Bildirimleri_Alarmaları_Görüntüle")</fact>
<fact>new
TLink("HS_Rapor_Bilgilerini_Getir","HSA_Kayıtları_Kriterlere_Göre_Getir")</fact>
<fact>new
TLink("H_Randevuları_Yönet","HSA_Randevu_Kaydet")</fact>
<fact>new
TLink("H_Acil_Yardim__Çağrısı_Gönder","HSA_Acil__Yardim_Sistemi_Bul")</fact>
<fact>new
TLink("H_Bildirimleri_Yönet","H_Bildirim_Gönder")</fact>
<fact>new
TLink("USG_Bildirimler_Gönder","HSA_Doktora_Bildirim_Gönder")</fact>
<fact>new
TLink("H_Randevuları_Yönet","HSA_Randevu_Güncelle_Sil")</fact>
<fact>new
TLink("Hasta_Sistemi_Verileri_Gönder","HSA_Kayıtları_Veritabanına_Gönder")</fact>
<fact>new
TLink("H_Bildirimleri_Yönet","H_Bildirim_İşlemleri_Yap")</fact>
</beliefset>
<beliefset name="contributions" class="TContrib">
</beliefset>
<beliefset name="dependencies" class="TDependency">
<!--TDependency: 'why'-Goal(must be AND-decomposed), Dependum,
Dependee -->
</beliefset>
<!--$ENTITIES-->

<!--$RESOURCES-->

<!--To force reevaluation of all conditions-->
<belief name="counter" class="Integer">
<fact evaluationmode="dynamic"> 0 </fact>
</belief>
</beliefs>

<goals>
<!-- default values: metalevelreasoning enabled, exclude when tried. -->
<achievegoal name="HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol"
exclude="when_failed">
<parameter name="param" class="String">
<value>""</value>
</parameter>
<parameter name="result" class="String" direction="out"/>
</achievegoal>

<achievegoal name="HS_Rapor_Bilgilerini_Getir" exclude="when_failed">
<parameter name="param" class="String">
<value>""</value>
</parameter>
<parameter name="result" class="String" direction="out"/>
</achievegoal>

<achievegoal name="H_Sağlık_Verilerini_Yönet" exclude="when_failed">
<parameter name="param" class="String">
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
<value>""</value>
</parameter>
<parameter name="result" class="String" direction="out"/>
</achievegoal>

<achievegoal name="H_Acil_Yardim__Çağrısı_Gönder"
exclude="when_failed">
  <parameter name="param" class="String">
    <value>""</value>
  </parameter>
  <parameter name="result" class="String" direction="out"/>
</achievegoal>

<achievegoal name="USG_Bildirimler_Gönder" exclude="when_failed">
  <parameter name="param" class="String">
    <value>""</value>
  </parameter>
  <parameter name="result" class="String" direction="out"/>
</achievegoal>

<achievegoal name="Hasta_Sistemi_Verileri_Gönder" exclude="when_failed">
  <parameter name="param" class="String">
    <value>""</value>
  </parameter>
  <parameter name="result" class="String" direction="out"/>
</achievegoal>

<achievegoal name="H_Bildirimleri_Yönet" exclude="when_failed">
  <parameter name="param" class="String">
    <value>""</value>
  </parameter>
  <parameter name="result" class="String" direction="out"/>
</achievegoal>

<achievegoal name="H_Randevuları_Yönet" exclude="when_failed">
  <parameter name="param" class="String">
    <value>""</value>
  </parameter>
  <parameter name="result" class="String" direction="out"/>
</achievegoal>

<!-- The meta-level goals for choosing between plans (and goals). -->
<metagoal name="meta_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol">
  <parameterset name="applicables" class="ICandidateInfo"/>
  <parameterset name="result" class="ICandidateInfo" direction="out"/>
  <trigger>
    <goal ref="HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol"/>
  </trigger>
</metagoal>
<metagoal name="meta_HS_Rapor_Bilgilerini_Getir">
  <parameterset name="applicables" class="ICandidateInfo"/>
  <parameterset name="result" class="ICandidateInfo" direction="out"/>
  <trigger>
    <goal ref="HS_Rapor_Bilgilerini_Getir"/>
  </trigger>
</metagoal>
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
</metagoal>
<metagoal name="meta_H_Acil_Yardim__Çağrısı_Gönder">
  <parameterset name="applicables" class="ICandidateInfo"/>
  <parameterset name="result" class="ICandidateInfo" direction="out"/>
  <trigger>
    <goal ref="H_Acil_Yardim__Çağrısı_Gönder"/>
  </trigger>
</metagoal>
<metagoal name="meta_USG_Bildirimler_Gönder">
  <parameterset name="applicables" class="ICandidateInfo"/>
  <parameterset name="result" class="ICandidateInfo" direction="out"/>
  <trigger>
    <goal ref="USG_Bildirimler_Gönder"/>
  </trigger>
</metagoal>
<metagoal name="meta_Hasta_Sistemi_Verileri_Gönder">
  <parameterset name="applicables" class="ICandidateInfo"/>
  <parameterset name="result" class="ICandidateInfo" direction="out"/>
  <trigger>
    <goal ref="Hasta_Sistemi_Verileri_Gönder"/>
  </trigger>
</metagoal>
<metagoal name="meta_H_Bildirimleri_Yönet">
  <parameterset name="applicables" class="ICandidateInfo"/>
  <parameterset name="result" class="ICandidateInfo" direction="out"/>
  <trigger>
    <goal ref="H_Bildirimleri_Yönet"/>
  </trigger>
</metagoal>
<metagoal name="meta_H_Randevuları_Yönet">
  <parameterset name="applicables" class="ICandidateInfo"/>
  <parameterset name="result" class="ICandidateInfo" direction="out"/>
  <trigger>
    <goal ref="H_Randevuları_Yönet"/>
  </trigger>
</metagoal>

</goals>

<plans>
  <!-- Initial plans for handling message requests. They have the task to
    create appropriate subgoals. They use waitqueues to store
    message events that arrived during the plan was busy. -->
  <plan name="requestPlan_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol">
    <body>new
GoalRequestPlan("HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol")</body>
    <trigger>
      <messageevent
ref="request_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol"/>
    </trigger>
  </plan>
  <plan name="requestPlan_HS_Rapor_Bilgilerini_Getir">
    <body>new GoalRequestPlan("HS_Rapor_Bilgilerini_Getir")</body>
    <trigger>
      <messageevent ref="request_HS_Rapor_Bilgilerini_Getir"/>
    </trigger>
  </plan>
</plans>
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
</plan>
<plan name="requestPlan_H_Sağlık_Verilerini_Yönet">
  <body>new GoalRequestPlan("H_Sağlık_Verilerini_Yönet")</body>
  <trigger>
    <messageevent ref="request_H_Sağlık_Verilerini_Yönet"/>
  </trigger>
</plan>
<plan name="requestPlan_H_Acil_Yardım_Çağrısı_Gönder">
  <body>new
GoalRequestPlan("H_Acil_Yardım_Çağrısı_Gönder")</body>
  <trigger>
    <messageevent
ref="request_H_Acil_Yardım_Çağrısı_Gönder"/>
  </trigger>
</plan>
<plan name="requestPlan_USG_Bildirimler_Gönder">
  <body>new GoalRequestPlan("USG_Bildirimler_Gönder")</body>
  <trigger>
    <messageevent ref="request_USG_Bildirimler_Gönder"/>
  </trigger>
</plan>
<plan name="requestPlan_Hasta_Sistemi_Verileri_Gönder">
  <body>new
GoalRequestPlan("Hasta_Sistemi_Verileri_Gönder")</body>
  <trigger>
    <messageevent
ref="request_Hasta_Sistemi_Verileri_Gönder"/>
  </trigger>
</plan>
<plan name="requestPlan_H_Bildirimleri_Yönet">
  <body>new GoalRequestPlan("H_Bildirimleri_Yönet")</body>
  <trigger>
    <messageevent ref="request_H_Bildirimleri_Yönet"/>
  </trigger>
</plan>
<plan name="requestPlan_H_Randevuları_Yönet">
  <body>new GoalRequestPlan("H_Randevuları_Yönet")</body>
  <trigger>
    <messageevent ref="request_H_Randevuları_Yönet"/>
  </trigger>
</plan>

<plan name="informChangePlan">
  <body>new InformChangePlan()</body>
  <trigger>
    <messageevent ref="inform_ChangeEnv"/>
  </trigger>
</plan>

<!-- Plans triggered by a parent goal, used to dispatch a child goal. -->

<!-- Plans associated to an AND-decomposed goal, used to dispatch all subgoals
in sequence. -->
<plan name="dispatchANDGoalPlan_H_Sağlık_Verilerini_Yönet">
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
<parameter name="param" class="String">
  <goalmapping ref="H_Sağlık_Verilerini_Yönet.param"/>
</parameter>
<parameter name="result" class="String">
  <goalmapping ref="H_Sağlık_Verilerini_Yönet.result"/>
</parameter>
<body>new ANDGoalPlan_H_Sağlık_Verilerini_Yönet()</body>
<trigger>
  <goal ref="H_Sağlık_Verilerini_Yönet"/>
</trigger>
</plan>
```

<!-- Meta-Plans associated to their Metagoals, used to chose between alternative

Plans

```
(and so between goals)-->
<plan name="metaPlan_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol">
  <parameterset name="applicables" class="ICandidateInfo">
    <goalmapping
ref="meta_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol.applicables"/>
  </parameterset>
  <parameterset name="result" class="ICandidateInfo" direction="out">
    <goalmapping
ref="meta_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol.result"/>
  </parameterset>
  <body>new
MetaPlan_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol()</body>
  <trigger>
    <goal ref="meta_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol"/>
  </trigger>
</plan>
<plan name="metaPlan_HS_Rapor_Bilgilerini_Getir">
  <parameterset name="applicables" class="ICandidateInfo">
    <goalmapping
ref="meta_HS_Rapor_Bilgilerini_Getir.applicables"/>
  </parameterset>
  <parameterset name="result" class="ICandidateInfo" direction="out">
    <goalmapping
ref="meta_HS_Rapor_Bilgilerini_Getir.result"/>
  </parameterset>
  <body>new MetaPlan_HS_Rapor_Bilgilerini_Getir()</body>
  <trigger>
    <goal ref="meta_HS_Rapor_Bilgilerini_Getir"/>
  </trigger>
</plan>
<plan name="metaPlan_H_Acil_Yardım_Çağrısı_Gönder">
  <parameterset name="applicables" class="ICandidateInfo">
    <goalmapping
ref="meta_H_Acil_Yardım_Çağrısı_Gönder.applicables"/>
  </parameterset>
  <parameterset name="result" class="ICandidateInfo" direction="out">
    <goalmapping
ref="meta_H_Acil_Yardım_Çağrısı_Gönder.result"/>
  </parameterset>
  <body>new MetaPlan_H_Acil_Yardım_Çağrısı_Gönder()</body>
  <trigger>
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```

        <goal ref="meta_H_Acil_Yardim__Çağrısı_Gönder"/>
    </trigger>
</plan>
<plan name="metaPlan_USG_Bildirimler_Gönder">
    <parameterset name="applicables" class="ICandidateInfo">
        <goalmapping
ref="meta_USG_Bildirimler_Gönder.applicables"/>
    </parameterset>
    <parameterset name="result" class="ICandidateInfo" direction="out">
        <goalmapping ref="meta_USG_Bildirimler_Gönder.result"/>
    </parameterset>
    <body>new MetaPlan_USG_Bildirimler_Gönder()</body>
    <trigger>
        <goal ref="meta_USG_Bildirimler_Gönder"/>
    </trigger>
</plan>
<plan name="metaPlan_Hasta_Sistemi_Verileri_Gönder">
    <parameterset name="applicables" class="ICandidateInfo">
        <goalmapping
ref="meta_Hasta_Sistemi_Verileri_Gönder.applicables"/>
    </parameterset>
    <parameterset name="result" class="ICandidateInfo" direction="out">
        <goalmapping
ref="meta_Hasta_Sistemi_Verileri_Gönder.result"/>
    </parameterset>
    <body>new MetaPlan_Hasta_Sistemi_Verileri_Gönder()</body>
    <trigger>
        <goal ref="meta_Hasta_Sistemi_Verileri_Gönder"/>
    </trigger>
</plan>
<plan name="metaPlan_H_Bildirimleri_Yönet">
    <parameterset name="applicables" class="ICandidateInfo">
        <goalmapping ref="meta_H_Bildirimleri_Yönet.applicables"/>
    </parameterset>
    <parameterset name="result" class="ICandidateInfo" direction="out">
        <goalmapping ref="meta_H_Bildirimleri_Yönet.result"/>
    </parameterset>
    <body>new MetaPlan_H_Bildirimleri_Yönet()</body>
    <trigger>
        <goal ref="meta_H_Bildirimleri_Yönet"/>
    </trigger>
</plan>
<plan name="metaPlan_H_Randevuları_Yönet">
    <parameterset name="applicables" class="ICandidateInfo">
        <goalmapping
ref="meta_H_Randevuları_Yönet.applicables"/>
    </parameterset>
    <parameterset name="result" class="ICandidateInfo" direction="out">
        <goalmapping ref="meta_H_Randevuları_Yönet.result"/>
    </parameterset>
    <body>new MetaPlan_H_Randevuları_Yönet()</body>
    <trigger>
        <goal ref="meta_H_Randevuları_Yönet"/>
    </trigger>
</plan>

```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
things. -->
<!-- Real Plans that hold the activity part of a capability and "do" the requested
things. -->
<plan name="realPlan_HSA_Randevuları_Görüntüle">
  <parameter name="param" class="String">
    <goalmapping ref="H_Randevuları_Yönet.param"/>
  </parameter>
  <parameter name="result" class="String">
    <goalmapping ref="H_Randevuları_Yönet.result"/>
  </parameter>
  <body>new RealPlan_HSA_Randevuları_Görüntüle()</body>
  <trigger>
    <goal ref="H_Randevuları_Yönet"/>
  </trigger>
</plan>
<plan name="realPlan_HSA_Arama_Sistemine_Kayıt_Ol">
  <parameter name="param" class="String">
    <goalmapping
ref="HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol.param"/>
  </parameter>
  <parameter name="result" class="String">
    <goalmapping
ref="HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol.result"/>
  </parameter>
  <body>new RealPlan_HSA_Arama_Sistemine_Kayıt_Ol()</body>
  <trigger>
    <goal ref="HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol"/>
  </trigger>
</plan>
<plan name="realPlan_HSA_Acil_Yardım_Çağrısı_Yap">
  <parameter name="param" class="String">
    <goalmapping
ref="H_Acil_Yardım_Çağrısı_Gönder.param"/>
  </parameter>
  <parameter name="result" class="String">
    <goalmapping
ref="H_Acil_Yardım_Çağrısı_Gönder.result"/>
  </parameter>
  <body>new RealPlan_HSA_Acil_Yardım_Çağrısı_Yap()</body>
  <trigger>
    <goal ref="H_Acil_Yardım_Çağrısı_Gönder"/>
  </trigger>
</plan>
<plan name="realPlan_H_Bildirimleri_Alarmları_Görüntüle">
  <parameter name="param" class="String">
    <goalmapping ref="H_Bildirimleri_Yönet.param"/>
  </parameter>
  <parameter name="result" class="String">
    <goalmapping ref="H_Bildirimleri_Yönet.result"/>
  </parameter>
  <body>new RealPlan_H_Bildirimleri_Alarmları_Görüntüle()</body>
  <trigger>
    <goal ref="H_Bildirimleri_Yönet"/>
  </trigger>
</plan>
<plan name="realPlan_HSA_Kayıtları_Kriterlere_Göre_Getir">
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
<parameter name="param" class="String">
    <goalmapping ref="HS_Rapor_Bilgilerini_Getir.param"/>
</parameter>
<parameter name="result" class="String">
    <goalmapping ref="HS_Rapor_Bilgilerini_Getir.result"/>
</parameter>
<body>new RealPlan_HSA_Kayıtları_Kriterlere_Göre_Getir()</body>
<trigger>
    <goal ref="HS_Rapor_Bilgilerini_Getir"/>
</trigger>
</plan>
<plan name="realPlan_HSA_Randevu_Kaydet">
    <parameter name="param" class="String">
        <goalmapping ref="H_Randevuları_Yönet.param"/>
    </parameter>
    <parameter name="result" class="String">
        <goalmapping ref="H_Randevuları_Yönet.result"/>
    </parameter>
    <body>new RealPlan_HSA_Randevu_Kaydet()</body>
    <trigger>
        <goal ref="H_Randevuları_Yönet"/>
    </trigger>
</plan>
<plan name="realPlan_HSA_Acil_Yardım_Sistemi_Bul">
    <parameter name="param" class="String">
        <goalmapping
ref="H_Acil_Yardım_Çağrısı_Gönder.param"/>
    </parameter>
    <parameter name="result" class="String">
        <goalmapping
ref="H_Acil_Yardım_Çağrısı_Gönder.result"/>
    </parameter>
    <body>new RealPlan_HSA_Acil_Yardım_Sistemi_Bul()</body>
    <trigger>
        <goal ref="H_Acil_Yardım_Çağrısı_Gönder"/>
    </trigger>
</plan>
<plan name="realPlan_H_Bildirim_Gönder">
    <parameter name="param" class="String">
        <goalmapping ref="H_Bildirimleri_Yönet.param"/>
    </parameter>
    <parameter name="result" class="String">
        <goalmapping ref="H_Bildirimleri_Yönet.result"/>
    </parameter>
    <body>new RealPlan_H_Bildirim_Gönder()</body>
    <trigger>
        <goal ref="H_Bildirimleri_Yönet"/>
    </trigger>
</plan>
<plan name="realPlan_HSA_Doktora_Bildirim_Gönder">
    <parameter name="param" class="String">
        <goalmapping ref="USG_Bildirimler_Gönder.param"/>
    </parameter>
    <parameter name="result" class="String">
        <goalmapping ref="USG_Bildirimler_Gönder.result"/>
    </parameter>
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
<body>new RealPlan_HSA_Doktora_Bildirim_Gönder()</body>
<trigger>
  <goal ref="USG_Bildirimler_Gönder"/>
</trigger>
</plan>
<plan name="realPlan_HSA_Randevu_Güncelle_Sil">
  <parameter name="param" class="String">
    <goalmapping ref="H_Randevuları_Yönet.param"/>
  </parameter>
  <parameter name="result" class="String">
    <goalmapping ref="H_Randevuları_Yönet.result"/>
  </parameter>
  <body>new RealPlan_HSA_Randevu_Güncelle_Sil()</body>
  <trigger>
    <goal ref="H_Randevuları_Yönet"/>
  </trigger>
</plan>
<plan name="realPlan_HSA_Kayıtları_Veritabanına_Gönder">
  <parameter name="param" class="String">
    <goalmapping ref="Hasta_Sistemi_Verileri_Gönder.param"/>
  </parameter>
  <parameter name="result" class="String">
    <goalmapping ref="Hasta_Sistemi_Verileri_Gönder.result"/>
  </parameter>
  <body>new RealPlan_HSA_Kayıtları_Veritabanına_Gönder()</body>
  <trigger>
    <goal ref="Hasta_Sistemi_Verileri_Gönder"/>
  </trigger>
</plan>
<plan name="realPlan_H_Bildirim_İşlemleri_Yap">
  <parameter name="param" class="String">
    <goalmapping ref="H_Bildirimleri_Yönet.param"/>
  </parameter>
  <parameter name="result" class="String">
    <goalmapping ref="H_Bildirimleri_Yönet.result"/>
  </parameter>
  <body>new RealPlan_H_Bildirim_İşlemleri_Yap()</body>
  <trigger>
    <goal ref="H_Bildirimleri_Yönet"/>
  </trigger>
</plan>
</plans>

<events>
  <!-- Specifies a request to achieve a goal (one per goal). -->
  <messageevent name="request_HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol"
direction="receive" type="fipa">
    <parameter name="performative" class="String" direction="fixed">
      <value>SFipa.REQUEST</value>
    </parameter>
    <parameter name="content-start" class="String" direction="fixed">
      <value>"HS_Eşleştirme_Arama_Sistemi_Kayıt_Ol"</value>
    </parameter>
  </messageevent>
</events>
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
</messageevent>
<messageevent name="request_HS_Rapor_Bilgilerini_Getir"
direction="receive" type="fipa">
  <parameter name="performative" class="String" direction="fixed">
    <value>SFipa.REQUEST</value>
  </parameter>
  <parameter name="content-start" class="String" direction="fixed">
    <value>"HS_Rapor_Bilgilerini_Getir"</value>
  </parameter>
</messageevent>
<messageevent name="request_H_Sağlık_Verilerini_Yönet"
direction="receive" type="fipa">
  <parameter name="performative" class="String" direction="fixed">
    <value>SFipa.REQUEST</value>
  </parameter>
  <parameter name="content-start" class="String" direction="fixed">
    <value>"H_Sağlık_Verilerini_Yönet"</value>
  </parameter>
</messageevent>
<messageevent name="request_H_Acil_Yardım_Çağrısı_Gönder"
direction="receive" type="fipa">
  <parameter name="performative" class="String" direction="fixed">
    <value>SFipa.REQUEST</value>
  </parameter>
  <parameter name="content-start" class="String" direction="fixed">
    <value>"H_Acil_Yardım_Çağrısı_Gönder"</value>
  </parameter>
</messageevent>
<messageevent name="request_USG_Bildirimler_Gönder" direction="receive"
type="fipa">
  <parameter name="performative" class="String" direction="fixed">
    <value>SFipa.REQUEST</value>
  </parameter>
  <parameter name="content-start" class="String" direction="fixed">
    <value>"USG_Bildirimler_Gönder"</value>
  </parameter>
</messageevent>
<messageevent name="request_Hasta_Sistemi_Verileri_Gönder"
direction="receive" type="fipa">
  <parameter name="performative" class="String" direction="fixed">
    <value>SFipa.REQUEST</value>
  </parameter>
  <parameter name="content-start" class="String" direction="fixed">
    <value>"Hasta_Sistemi_Verileri_Gönder"</value>
  </parameter>
</messageevent>
<messageevent name="request_H_Bildirimleri_Yönet" direction="receive"
type="fipa">
  <parameter name="performative" class="String" direction="fixed">
    <value>SFipa.REQUEST</value>
  </parameter>
  <parameter name="content-start" class="String" direction="fixed">
    <value>"H_Bildirimleri_Yönet"</value>
  </parameter>
</messageevent>
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
type="fipa">
<messageevent name="request_H_Randevuları_Yönet" direction="receive"

    <parameter name="performative" class="String" direction="fixed">
        <value>SFipa.REQUEST</value>
    </parameter>
    <parameter name="content-start" class="String" direction="fixed">
        <value>"H_Randevuları_Yönet"</value>
    </parameter>
</messageevent>

<!--messages for informs on environment changes-->
<messageevent name="inform_ChangeEnv" direction="receive"
    type="fipa">
    <parameter name="performative" class="String" direction="fixed">
        <value>SFipa.INFORM</value>
    </parameter>
    <parameter name="content-start" class="String" direction="fixed">
        <value>"change "</value>
    </parameter>
</messageevent>

<!--FIPA-messages needed to communicate with other agents-->
<messageevent name="request" direction="send" type="fipa">
    <parameter name="performative" class="String" direction="fixed">
        <value>SFipa.REQUEST</value>
    </parameter>
    <parameter name="reply-with" class="String">

<value>SFipa.createUniqueId($scope.getAgentName())</value>
    </parameter>
</messageevent>

<!--default FIPA-messages needed to communicate with other agents-->
<messageevent name="inform" direction="send_receive" type="fipa">
    <parameter name="performative" class="String" direction="fixed">
        <value>SFipa.INFORM</value>
    </parameter>
</messageevent>
<messageevent name="agree" direction="send_receive" type="fipa">
    <parameter name="performative" class="String" direction="fixed">
        <value>SFipa.AGREE</value>
    </parameter>
</messageevent>
<messageevent name="failure" direction="send_receive" type="fipa">
    <parameter name="performative" class="String" direction="fixed">
        <value>SFipa.FAILURE</value>
    </parameter>
</messageevent>
<messageevent name="n_u" direction="send_receive" type="fipa">
    <parameter name="performative" class="String" direction="fixed">
        <value>SFipa.NOT_UNDERSTOOD</value>
    </parameter>
</messageevent>
<messageevent name="refuse" direction="send_receive" type="fipa">
    <parameter name="performative" class="String" direction="fixed">
```

Ek4 Etmen Gerçekleştirim Kodları (devam)

```
<value>SFipa.REFUSE</value>
</parameter>
</messageevent>

</events>

<expressions>
  <!--All expressions are not changed during the automatic code generation-->
  <!-- This query selects the first matching entry from the dictionary,
        whereby the parameter $eword is compared to the first element of
        a belief set tuple. -->
  <expression name="query_link">
    select $link.getGoal(1)
    from TLink $link in $beliefbase.decomp
    where $link.get(0).equals($component)
    order by $link.getPriority() desc
    <parameter name="$component" class="String"/>
  </expression>
  <expression name="query_ME_link">
    select $link.get(1)
    from TLink $link in $beliefbase.meansend
    where $link.get(0).equals($component)
    <parameter name="$component" class="String"/>
  </expression>
  <expression name="query_contributions">
    select $link
    from TContrib $link in $beliefbase.contributions
    where $link.get(0).equals($component)
    <parameter name="$component" class="String"/>
  </expression>
  <expression name="query_dependencies">
    select $link
    from TDependency $link in $beliefbase.dependencies
    where $link.getWhyGoal().equals($component)
    <parameter name="$component" class="String"/>
  </expression>
</expressions>

<properties>
  <!-- Only log outputs >= level are printed. -->
  <property name="logging.level">Level.INFO</property>
  <!-- The default parent handler prints out log messages on the console. -->
  <property name="logging.useParentHandlers">true</property>
  <!--<property name="debugging">true</property>-->
</properties>

  <!-- initialstates changed to configurations in Jadex 0.96 -->
  <configurations>
    <configuration name="default">
    </configuration>
  </configurations>

</agent>
```

package org.mashemato.ws;

Ek5 Restful Web Servis Gerçekleştirim Kodları

```
import javax.ws.rs.core.Context;
import javax.ws.rs.core.UriInfo;
import javax.ws.rs.PathParam;
import javax.ws.rs.Consumes;
import javax.ws.rs.PUT;
import javax.ws.rs.Path;
import javax.ws.rs.GET;
import javax.ws.rs.HeaderParam;
import javax.ws.rs.Produces;

// Hasta Arayüzünün etmen sistemine bağlanması amacıyla kullanılmaktadır.
@Path("/HastaWS")
public class HastaRestfulWS {

    @Context
    private UriInfo context;

    public HastaRestfulWS() {

    }

    /**
     * İstenilen belgeyi geri döndürür, Login, bildirimler, randevular gibi bilgileri talep edebilir.
     * * @param talepMesaji gönderilen mesaj
     * * @return an instance of java.lang.String
     */
    @GET
    @Produces("application/xml")
    @Path("/getirBelge/{kullanici kodu}")
    public String getirBelge(@PathParam("kullanici kodu")String
kullanici kodu, @HeaderParam("talepMesaji") String talepMesaji) {
        return Yardimci.getInstance().getirBelge(kullanici kodu, talepMesaji);
    }

    /**
     * Etmenlere mesaj göndermek, kayıt yapmak veya bildirimde bulunmak amacıyla kullanılır
     * * @param belge gönderilen mesaj
     * * @return
     */
    @PUT
    @Consumes("application/xml")
    public void belgeGonder(String belge) {
        Yardimci.getInstance().belgeGonder(belge);
    }
}
```