

BUTA

HEALTH@HAND

INITIAL DESIGN REPORT

Hilal BAYDAROV - 1474022

Hüseyin DİRİCAN -1500016

Uğur CEBECİ- 1502236

Yusuf DÖNMEZ -1502293

BUTA	1
1 Introduction	3
1.1 Purpose of the Document	3
1.2 Project Definition	3
1.3 Motivation	4
2 Design Strategy and Development Methodology	4
3 Data Design	6
3.1 ER Design	6
3.2 Database Schemas	7
3.2.1 Patient Table	7
3.2.2 Hospital Table	7
3.2.3 Clinic Table	8
3.2.4 Doctor Table	8
3.2.5 Appointmet Table	8
3.2.6 Comment Table	9
3.2.7 Prescription Table	9
4 Architectural Design	10
4.1 Design Overview	10
4.2 Data Flow Diagrams	11
4.2.1 Level 0 DFD	11
4.2.2 Level 1: Health@Hand System	12
4.2.3 DFD Level 2 Process Command	13
4.2.4 DFD Level 3: Get Patient Place Info	14
4.3 Class Diagrams	15
4.3.1 View Package	15
4.3.2 Controller Package	15
4.3.3 Model Package	18
4.4 Activity Diagram	19
4.4.1 Login Diagram	19
4.4.2 Edit Profile Diagram	20
4.4.2 Doctor Access	21
4.4.3 General Activity Diagram	22
4.5 Use Case Diagrams	23
4.5.1 Patient Use Case Diagram	23
4.5.2 Doctor Use Case Diagram	23
5 Interface Design	24
5.1 Main Screen	24
5.2 Login Screen	25
5.3 Sign-up Screen	26
5.4 Patient Profile	27
5.5 Doctor Profile Screen	28
5.6 Hospital Profile Screen	29
5.7 Search Hospital Screen	30
6 Project Schedule	31
7 Conclusion	32
7.1 Previous Work	32
7.2 Future Work	32
8 References	33

1 Introduction

The aim of this report is to present initial design plan of the Health@Hand project that is conducted by Buta Inc. In this report general architecture of project will be explained by using the requirements of the Health@Hand project of group “BUTA” . In this report, the field research regarding the project is explained, detailed use cases and requirements are described and the road map of the project is presented.

1.1 Purpose of the Document

This report is written to show initial design of Health@Hand project by BUTA project team. The headings and parts of this document are divided appropriately by grouping relevant subjects of whole system. Project group has tried to give the definition and design of the software system by dividing subsystems and showing the relations with each other. In addition, the other necessary issues like implementation requirements, specifications are handled. In this report, the field research regarding the project is explained, detailed use cases and requirements are described and the road map of the project is presented.

1.2 Project Definition

Nowadays, mobile phones are the most widely used communication tools. Almost everyone has mobile phone or even phones. Therefore, using mobile phones just for communication is a kind of narrow usage. Together with the rapidly developing mobile industry, mobile device producers offer lots of beneficial alternatives to their customers. Therefore, today’s mobile phones are much more than just a communication device. They are now used as entertainment centers (music players, TVs, game players, web surfers, etc.), communications hubs (email, text, MMS, the web), general purpose computers (calculators, organizers, spreadsheets, etc.). Even they can analyze blood to detect HIV, Malaria and more.

Mobile phones become a part of our life. Most people cannot go somewhere without taking their phones with them. They almost live together with their phones. Therefore we could take advantage of this situation to ease our life more. Especially it can help people living in rural areas where it is harder to meet basic human needs such as health care.

1.3 Motivation

Health@Hand will be Java mobile application and to develop it, Turkcell API, Google Map API, Java mobile SDK will be used. Health@Hand will enable users to get health care services easier. Health@Hand takes the doctor or hospital information from its database and enable user to see doctor or hospital schedule. Moreover, people can see the places of doctor or hospital from the Google Map view. Therefore using Health@Hand application, people can find the closest available doctor by saving their time. They no more need to go to hospital to see whether the doctor is available or not. Furthermore, Health@Hand database keeps the users health log also. Hence users or doctors can know patient's earlier health information. Besides on these feature, Health@Hand enables electronic prescription (e-prescription). Hence, users no need to go to doctor and wait in the patient queue to take their prescription and then go to pharmacy. Doctor sends prescription by examining the patient health logs and user can take the prescription via Health@Hand. Moreover, user can confirm to buy medicine electronically.

2 Design Strategy and Development Methodology

Since none of our BUTA group member is experienced in mobile application development, the details will be learned while developing the project. Therefore, we could make some mistake during the development process. For this reason, using a development model like "Waterfall" will be unreasonable, because making a mistake at the beginning of the development process is inevitable by an inexperienced team. For this reason, we need a more flexible development methodology like Spiral model. We decide to use flexible development methodology like "Spiral". The idea behind this model is that at each choice point in the software development process, one assesses the risk that the project could fail to meet its goals. Based on that analysis the next step is to mitigate that risk. It might mean doing a prototype, refining the

requirements, or doing more testing. Therefore, it lets the development team learning by making mistakes and even an inexperienced team can form a solid structure after enough number of iterations.

However, making an adequate number of iterations, could take lots of times. Hence, taking an advantage of Agile Methodology can really speed up our development process. For this reason, BUTA will mainly follow Spiral methodology during development process. However, for the parts that needs special attention Agile methodology can be feasible.

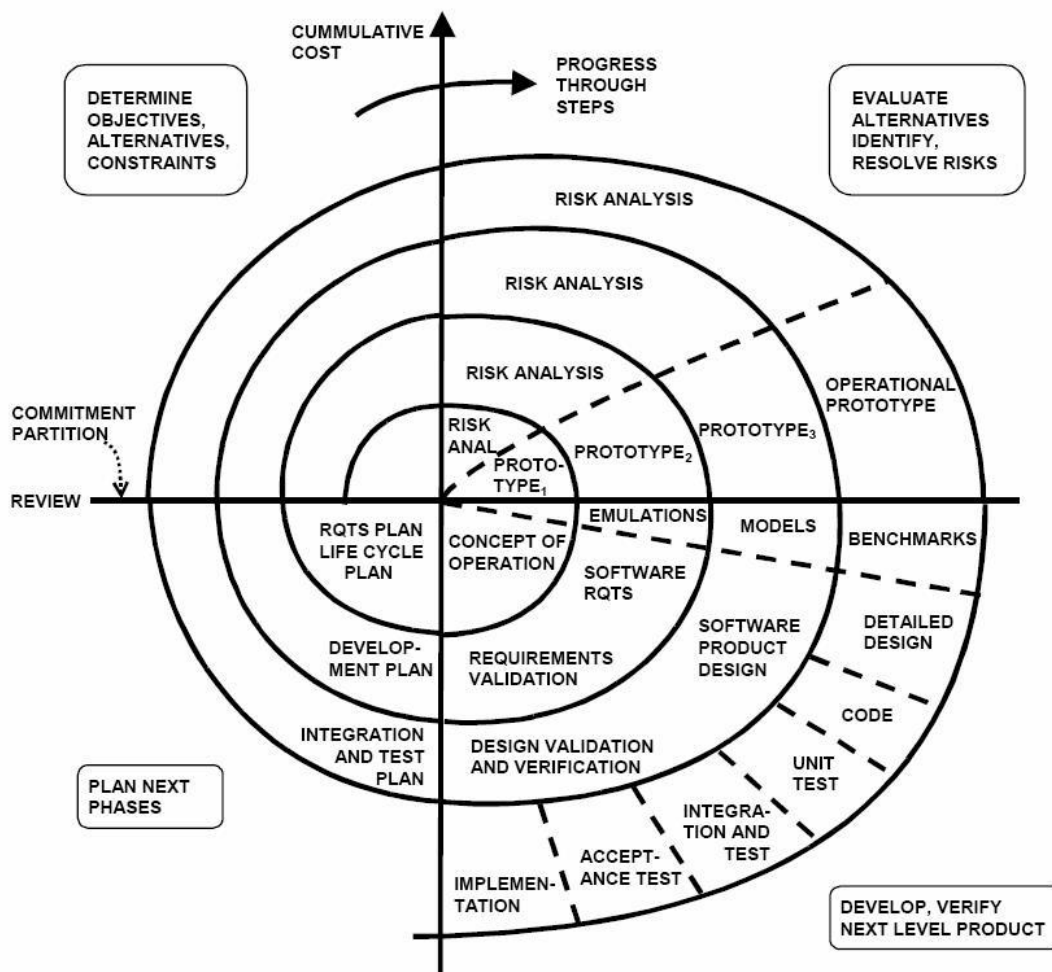


Figure 1 - http://www.reliablesoftware.com/weblog/uploaded_images/spiral-12085.bmp

3 Data Design

3.1 ER Design

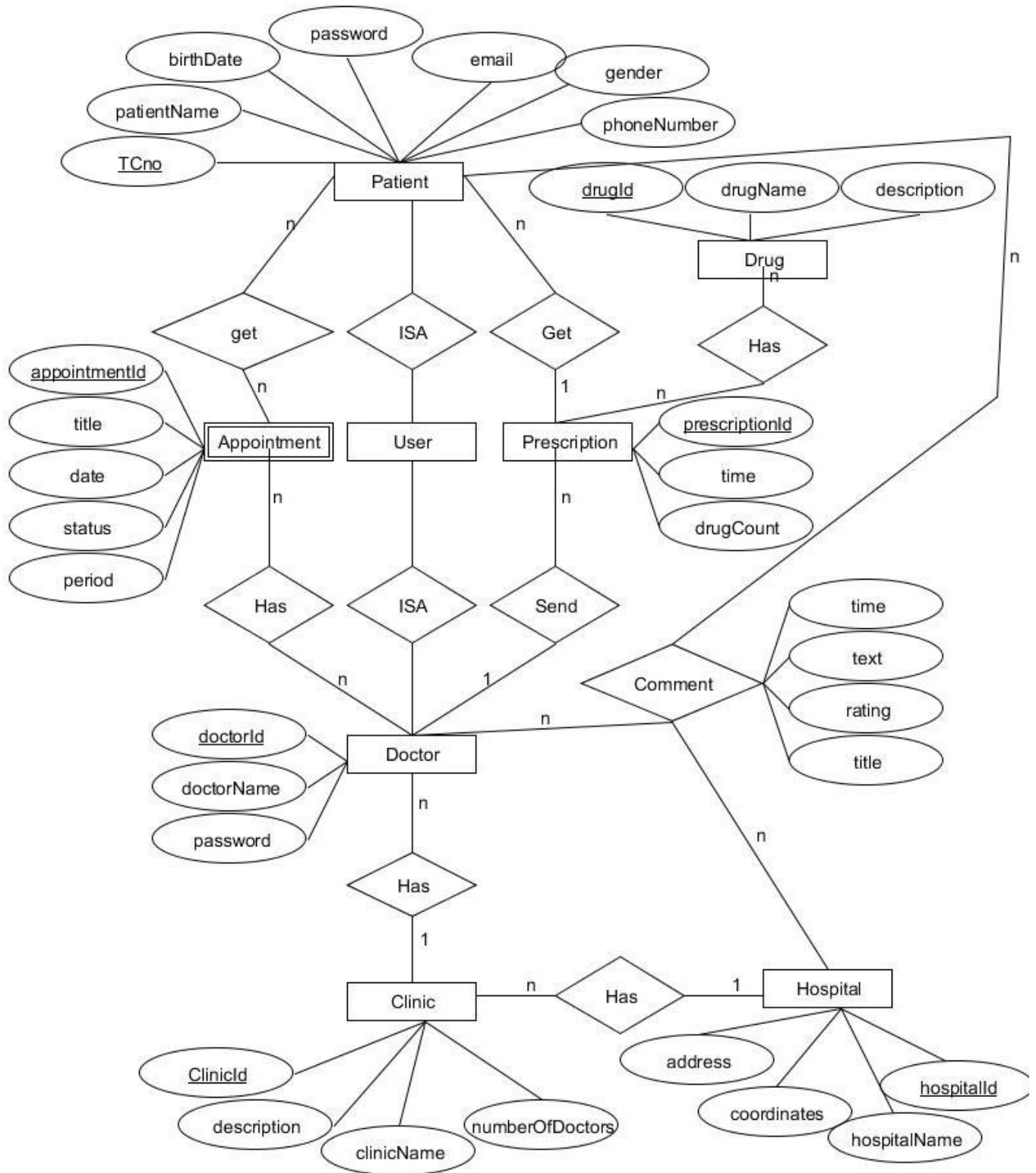


Figure 2 - System ER Diagram

3.2 Database Schemas

3.2.1 Patient Table

Field	Type	Null	Foreign Key	References
<u>Tcno</u>	integer	No	No	-
email	Varchar(50)	No	No	-
patientName	Varchar(50)	No	No	-
password	Varchar(50)	No	No	-
gender	Varchar(10)	Yes	No	-
phoneNumber	integer	No	No	-
bloodType	Varchar(10)	No	No	-

Table 1 – Patient Table

This is table that holds Health@Hand 's registered users data. Since their "Tcno"s are unique that field is accepted as primary key of this table.

3.2.2 Hospital Table

Field	Type	Null	Foreign Key	References
<u>hospitalId</u>	integer	No	No	-
hospitalName	Varchar(50)	No	No	-
address	Varchar(300)	No	No	-
longitude	float	No	No	-
latitude	float	No	No	-
altitude	float	No	No	-

Table 2 – Hospital Table

This table holds information related to hospitals that registered to our system. There are regular information like address and location information of hospitals. This table can be updated only by administrators of hospitals except "hospitalId" field. They will be shown in Google Map through their location information hold in this table.

3.2.3 Clinic Table

Field	Type	Null	Foreign Key	References
<u>clinicId</u>	integer	No	No	-
clinicName	Varchar(50)	No	No	-
description	Varchar(300)	No	No	-
numberOfDoctors	integer	No	No	-

Table 3 – Clinic Table

In that table database holds clinics data like number of doctor that hospital has for that clinic, and general information about clinic.

3.2.4 Doctor Table

Field	Type	Null	Foreign Key	References
<u>doctorId</u>	integer	No	No	-
doctorName	Varchar(50)	No	No	-
password	Varchar(50)	No	No	-

Table 4 – Doctor Table

Doctor table holds information related to doctors, which can do changes on appointment schedule, that registered by hospital admins to their hospitals' clinics.

3.2.5 Appointment Table

Field	Type	Null	Foreign Key	References
<u>appointmentId</u>	integer	No	No	-
<u>doctorId</u>	integer	No	Yes	Doctor
<u>patientId</u>	integer	No	Yes	Patient
status	Varchar(15)	No	No	-
period	Varchar(50)	No	No	-
title	text	No	No	-
date	integer	Yes	No	-

Table 5 – Appointment Table

This table holds appointment related data such as which patient and doctor are included for specific appointment. Period of appointment that decided by doctors are included in this table. Those period's status and date are also hold in Appointment table.

3.2.6 Comment Table

Field	Type	Null	Foreign Key	References
<u>patientId</u>	integer	No	Yes	Patient
<u>doctorId</u>	integer	No	Yes	Doctor
rating	integer	Yes	No	-
time	Varchar(15)	No	No	-
Title	Varchar(50)	No	No	-
text	text	No	No	-
rating	integer	Yes	No	-

Table 6 – Comment Table

Health@Hand system give the right to users making comment on both doctor and hospitals which they visited before. And they can rate about this issues.

3.2.7 Prescription Table

Field	Type	Null	Foreign Key	References
<u>prescriptionId</u>	integer	No	No	-
<u>doctorId</u>	integer	No	Yes	Doctor
<u>patientId</u>	integer	No	Yes	Patient
drugCount	integer	No	No	-
date	integer	No	No	-

Table 7 – Prescription Table

Doctors can write their patient's prescriptions by online and send it both patient and pharmacy. This table holds owner and writer info of this prescription and date it written.

4 Architectural Design

4.1 Design Overview

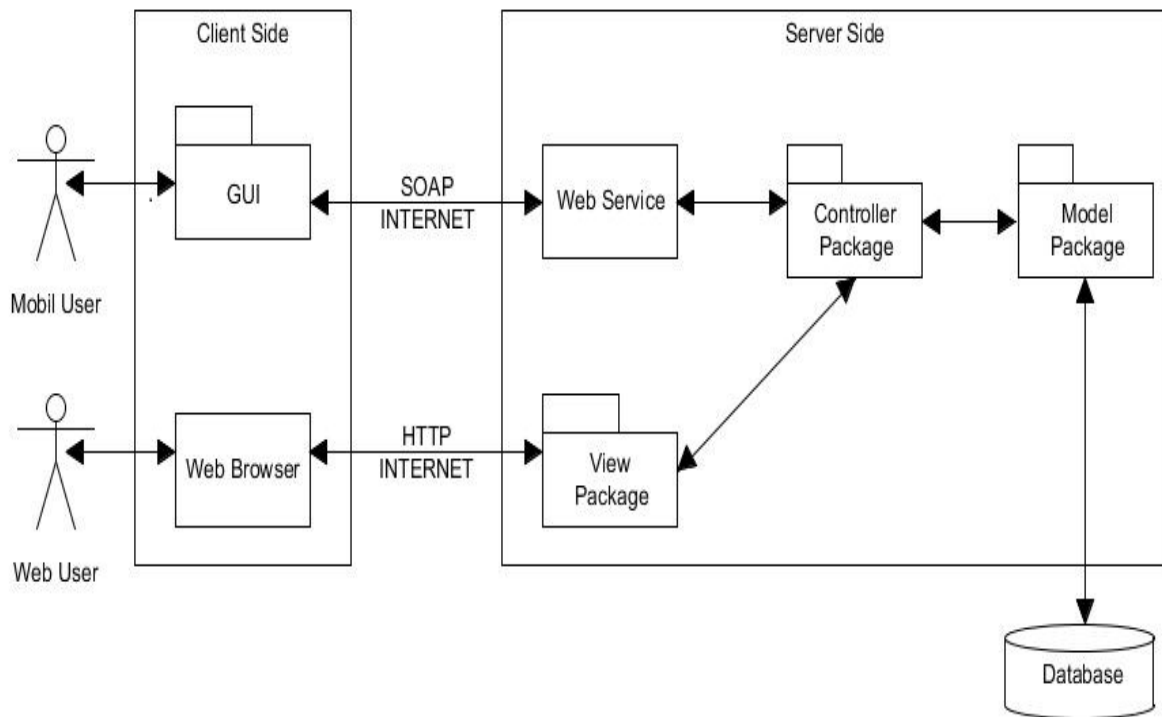


Figure 3 - Design Overview

As it is explained throughout the report, Health@Hand system needs a complex architecture because lots of modules will work cooperatively. Furthermore, together with the mobile and internet technology also user interests are changing rapidly. Therefore, our architecture should be easily modifiable according to these changes and it should allow developers for developing new modules. Moreover, it should make this complex system's development phase less difficult with good separation of layers.

4.2 Data Flow Diagrams

4.2.1 Level 0 DFD

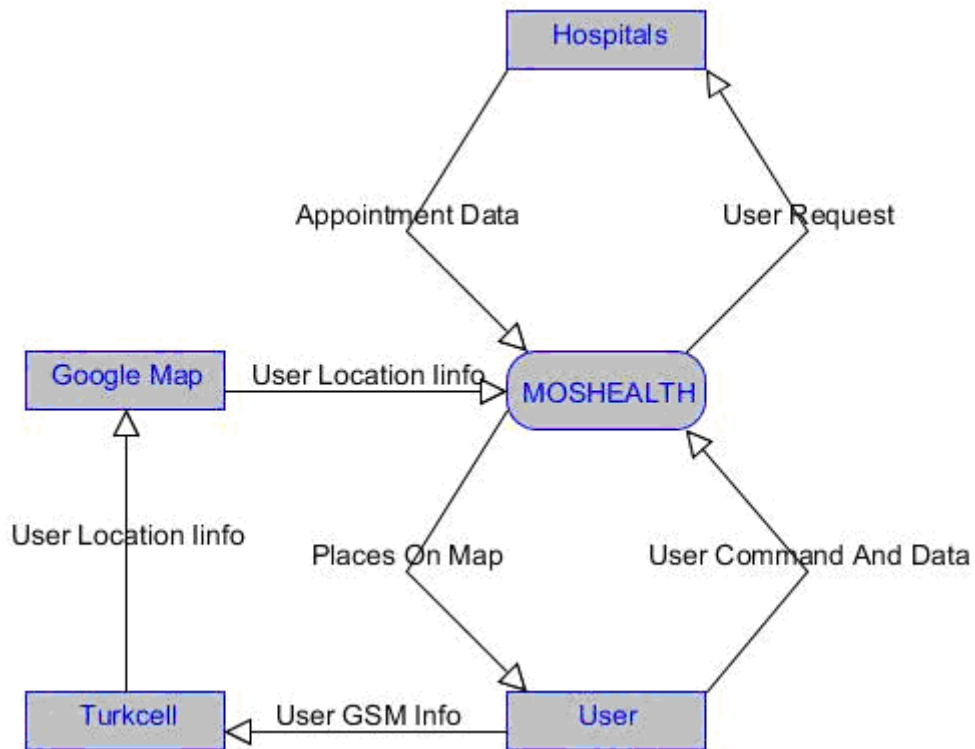


Figure 4 Context Level DFD

4.2.2 Level 1: Health@Hand System

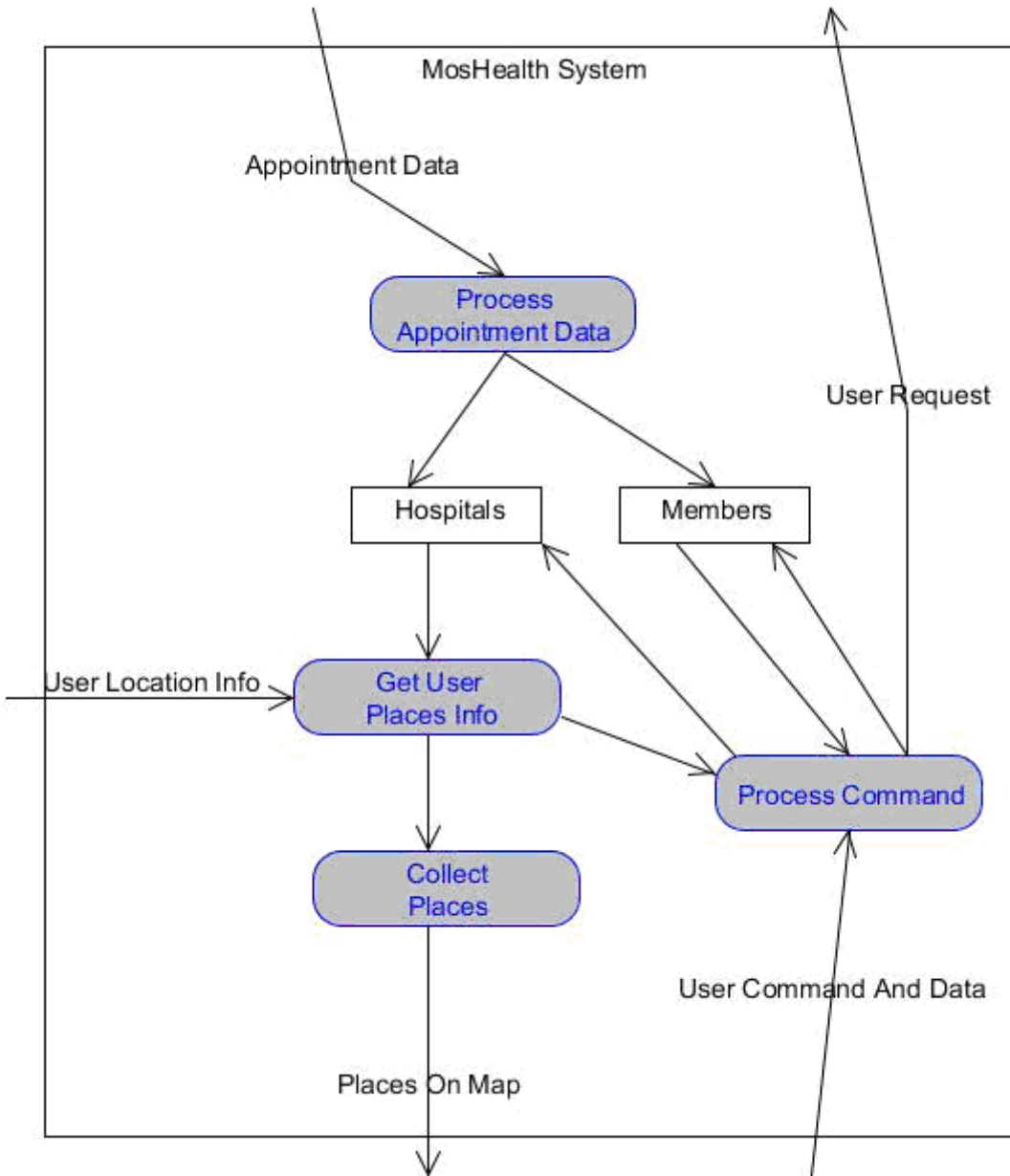


Figure 5 Health@Hand System Level 1 DFD

4.2.3 DFD Level 2 Process Command

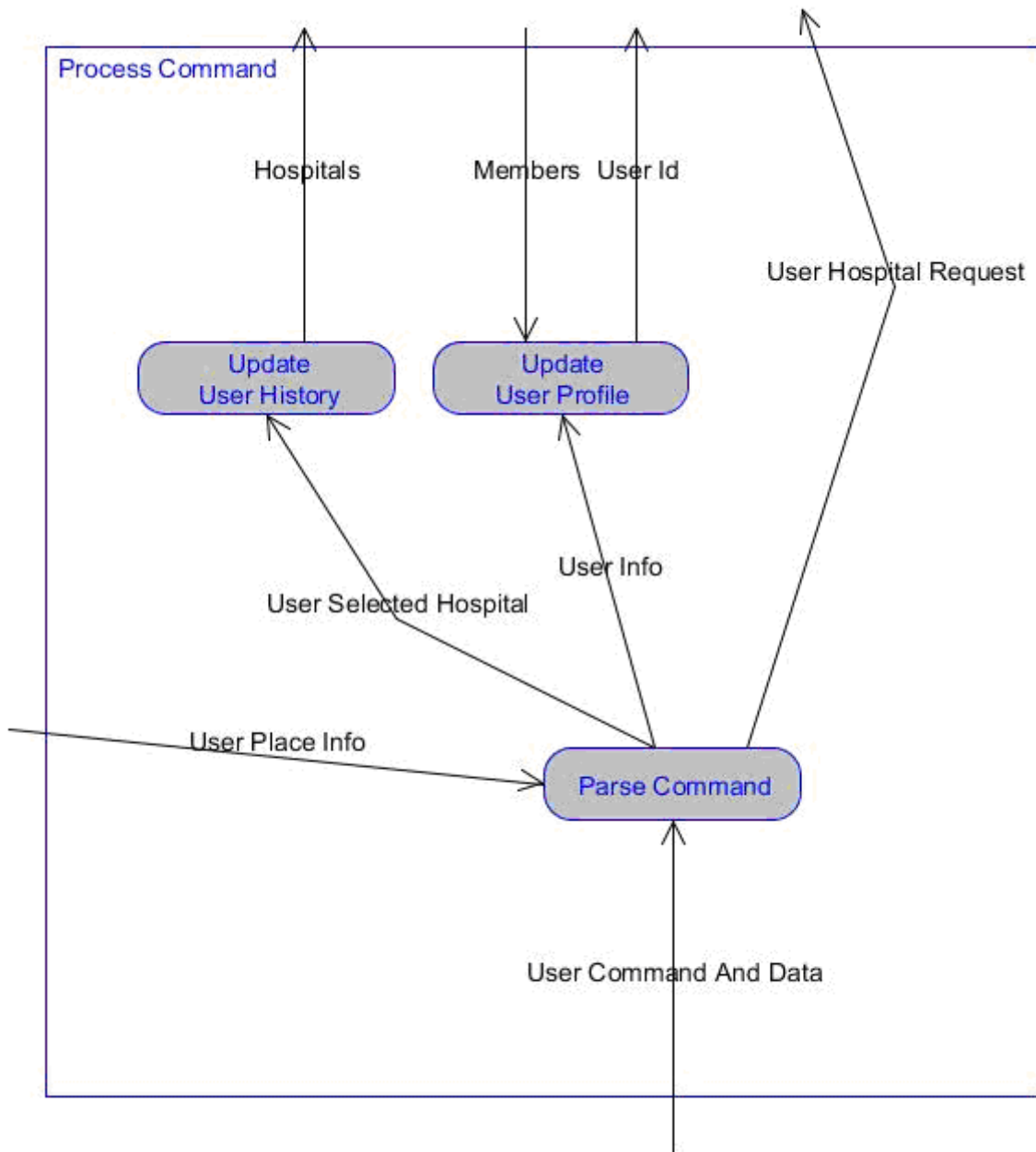


Figure 6 Process Command Level 2 DFD

4.2.4 DFD Level 3: Get Patient Place Info

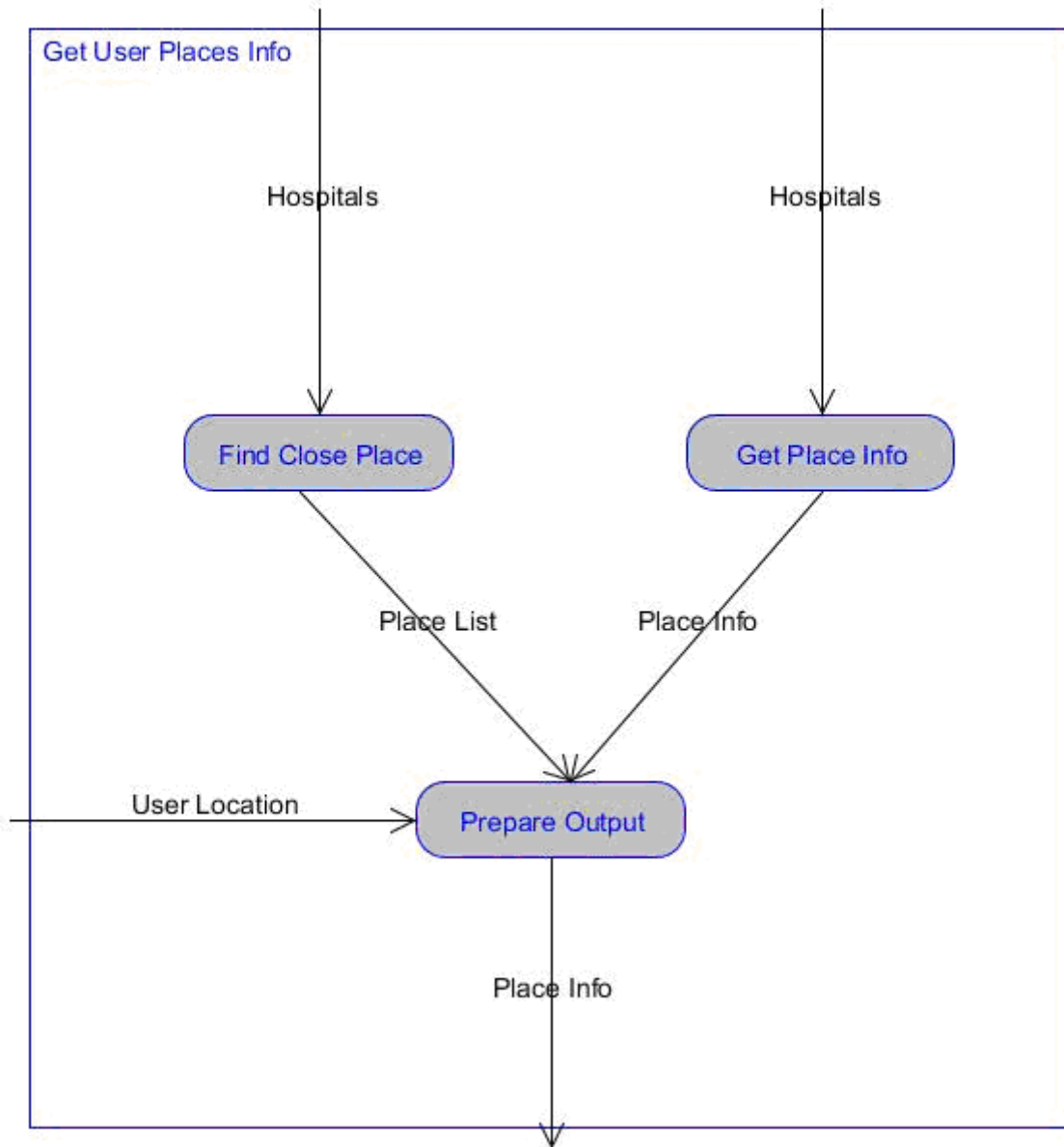


Figure 7 Get Patient Place Info Level 3 DFD

4.3 Class Diagrams

4.3.1 View Package

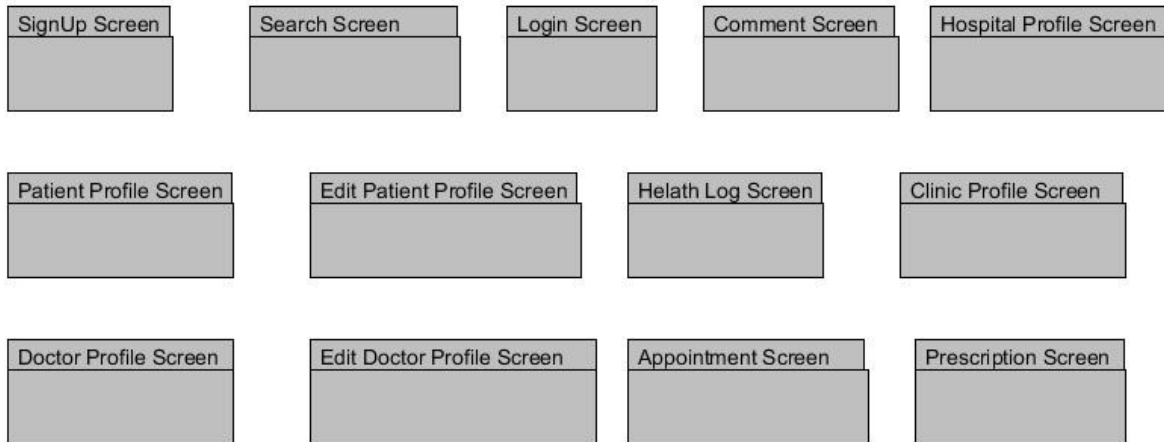


Figure 8 - View Package

View packages hold the classes which implement user interfaces. Each screen class processes user request and calls related controller functions with appropriate arguments. This classes also renders user interfaces using outputs retrieved from controller functions.

4.3.2 Controller Package

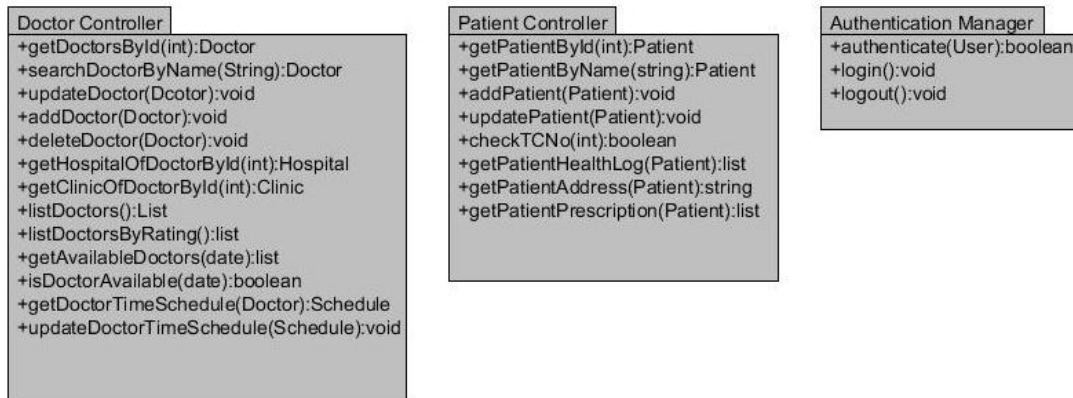


Figure 9 - Controller Packages1

Activity Diagrams In user Controller part we have two classes PatientController and DoctorController class which are the main part of this project. There many different functions between them.

When user log in to our system Controller starts to work and handle some function. For example, in PatientController class we check the correctness of TCno of patient

whether it is valid or not before log in to our system. In addition, we store the patient address, healthLog and prescriptions he got in our database. We can add and remove Patient with the help of this class also we will update patient profile with this class. In patient table we store patients name and other specific information and it will easy to us to reach patient information at any time.

Similarly, In DoctorController class we can add and remove Doctor and update doctor’s profile. In this class we will also store the doctor’s time schedule to check if he/she is available when the patient want to get an appointment. For example, isDoctorAvailable(Date) method check whether doctor is available or not. We can also search doctor, update doctor’s time schedule and get the available doctor with the help of this class. In addition, we can list doctors by their rating. The rating check the patient was satisfied or not.

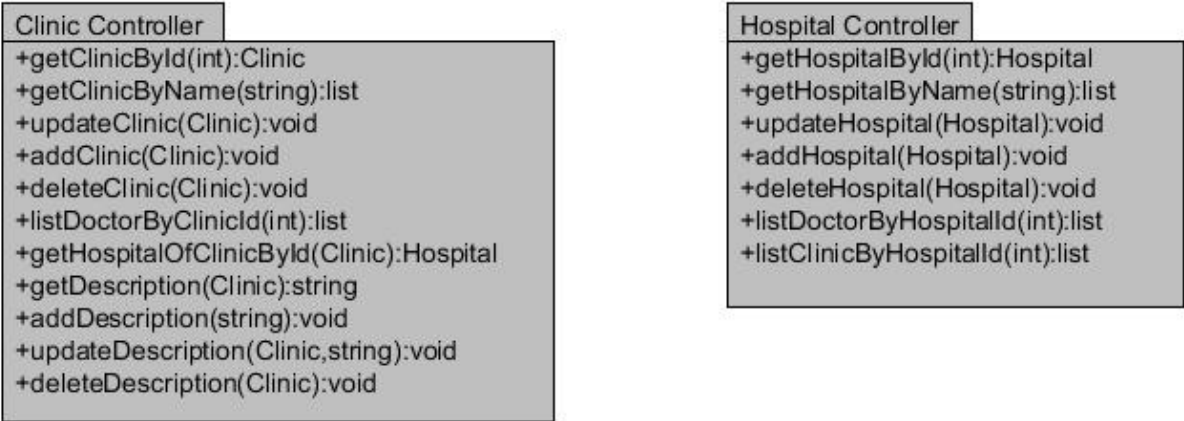


Figure 10 - Controller Packages2

In HospitalController part we also have two main controller which are ClinicController and Hospital Controller. The HospitalController class also has basic functions that are add hospital and remove hospital and update hospital’s information. Hospital can contains many clinic and we can list them with the help of this class. We can also the doctors who are work in this hospital and we can list them and can get the comment and rating which are commented to them.

In Clinic Controller class we can add and update the clinic’s description. We can get the list of doctor’s who are work under any department. In addition, we can list the clinics of any hospital by the functions which this class contain.

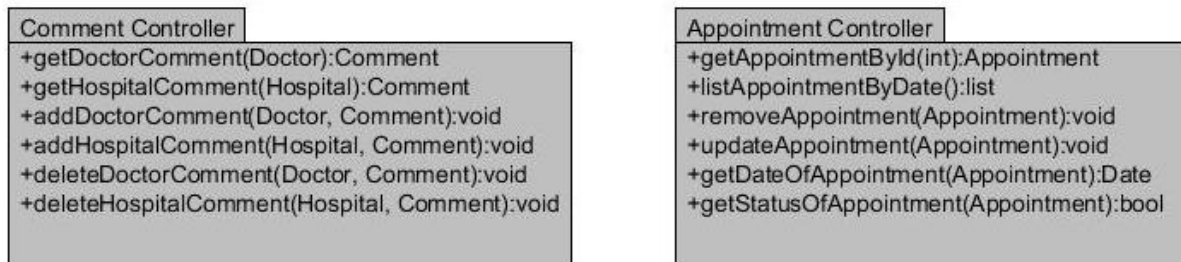


Figure 11 -Controller Packages3

In CommentController class patient(user) can comment and rate the Doctor and the Hospital and we will organize their comments with the help of this class. We can get their comment, its title, its date and so on.. Also we can delete it and get or list the comment by doctor or hospital. For example, addDoctorComment and addHospitalComment are the two methods of them which do the adding comment to doctor's or hospital's profile. The rating help other patients to select best for them.

In AppointmentController class we store the date of the appointment that the patient got the rendezvous. After the date of appointment the patient got we give a priority for patient to change it. In this duration we check the doctor's time schedule if he/she appropriate or not. In other words, if the doctor is busy the system will show a message that the doctor has a rendezvous at that time. The system also has a function to list the appointment in given date in order to help hospital to do their business in detail.

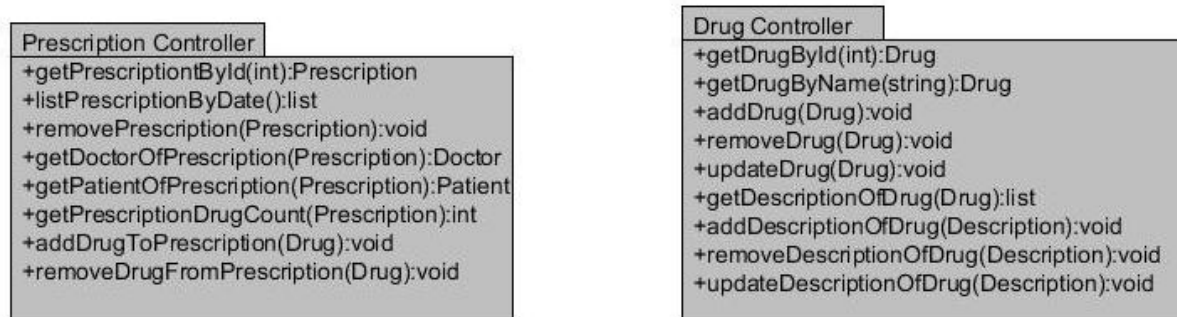


Figure 12 - Controller packages4

The last part of this project contains two class which names are the PrescriptionController and DrugController. It is obviously known that in this project doctor has a authority to add and remove a drug also doctor has a authority remove the prescription of the patient.

The system also shows how many drug the prescription contains and their specifications or description. In DrugController class also has a basic function add drug and remove drug and also update drug description or other specification. In addition, the patient can see the drug's description and the doctor can add and update this drug's descriptions. DrugController class has a strict relation with PrescriptionController class

4.3.3 Model Package

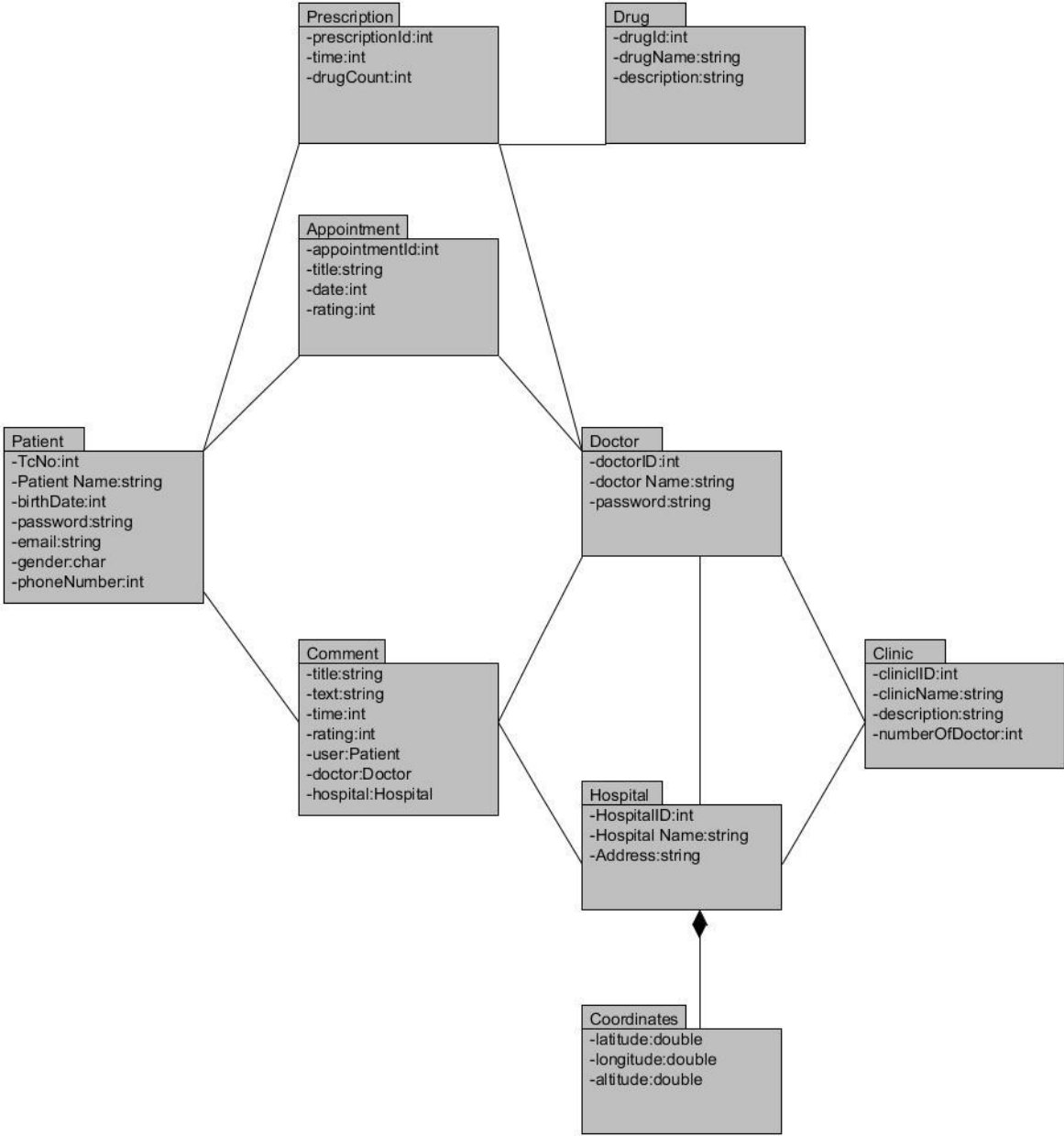


Figure 13 - Model Package

4.4 Activity Diagram

In this part of the report, some features of Health@Hand System are explained as activity diagrams below.

4.4.1 Login Diagram

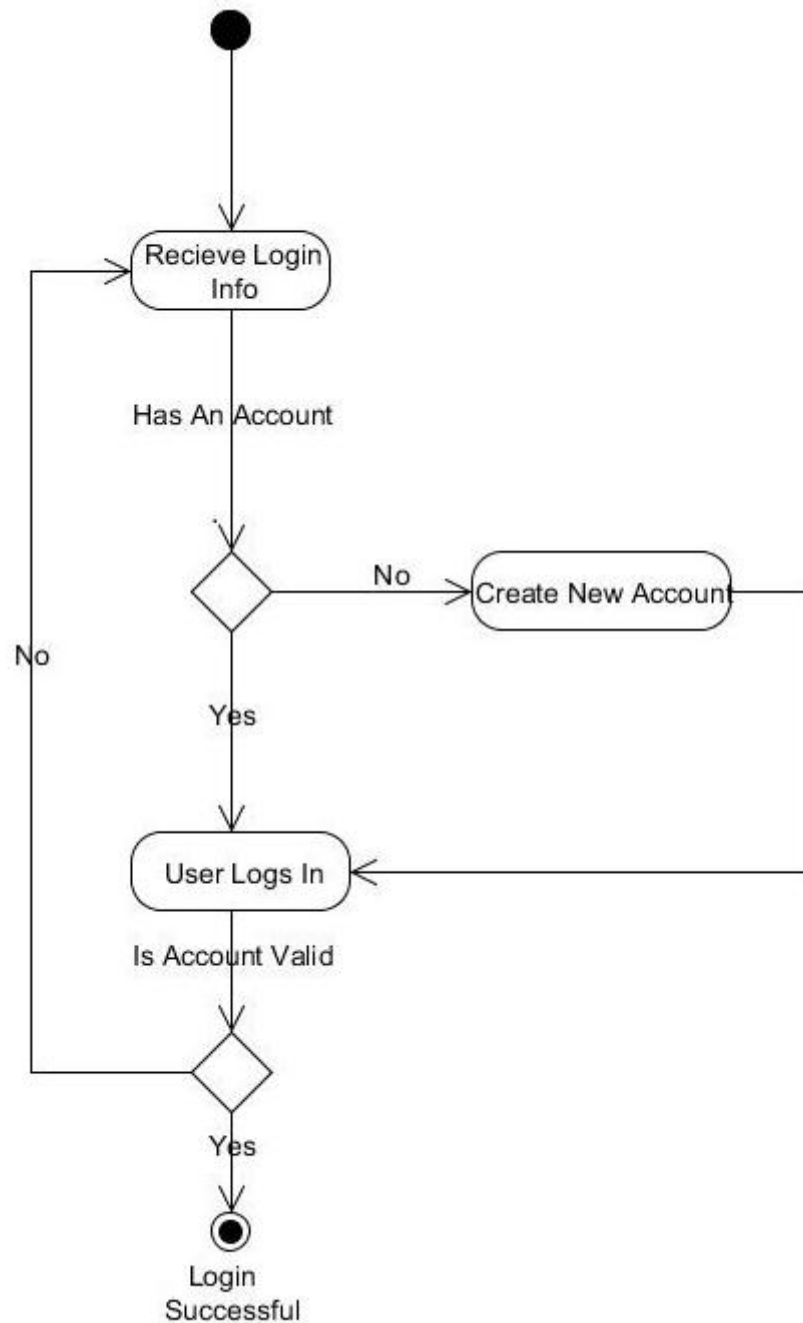


Figure 14 - Login Activity Diagram

4.4.2 Edit Profile Diagram

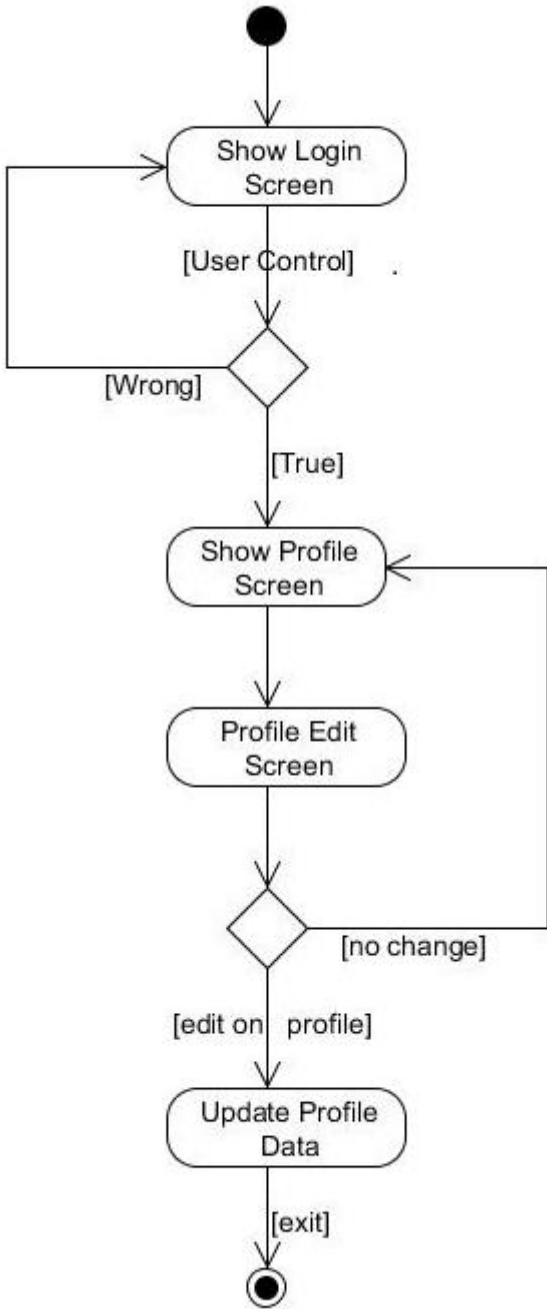


Figure 15 - Edit Profile Diagram

4.4.2 Doctor Access

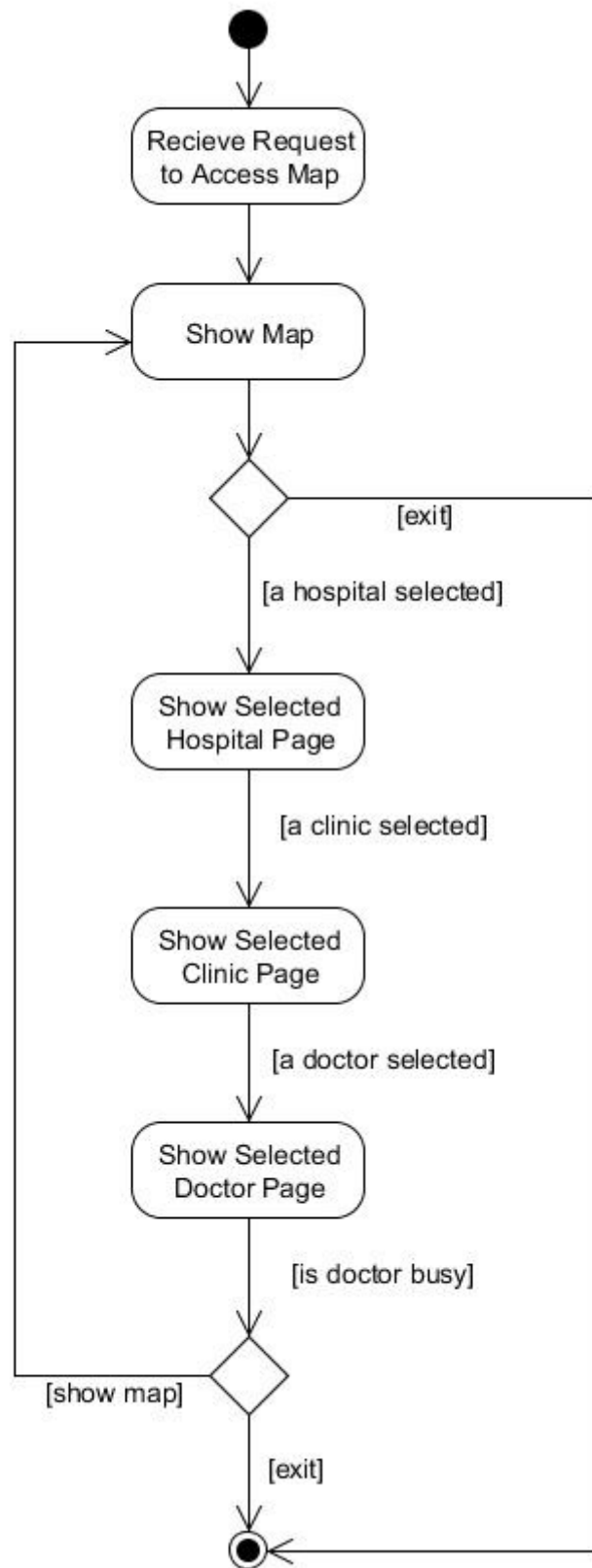


Figure 16- Doctor Access Diagram

4.4.3 General Activity Diagram

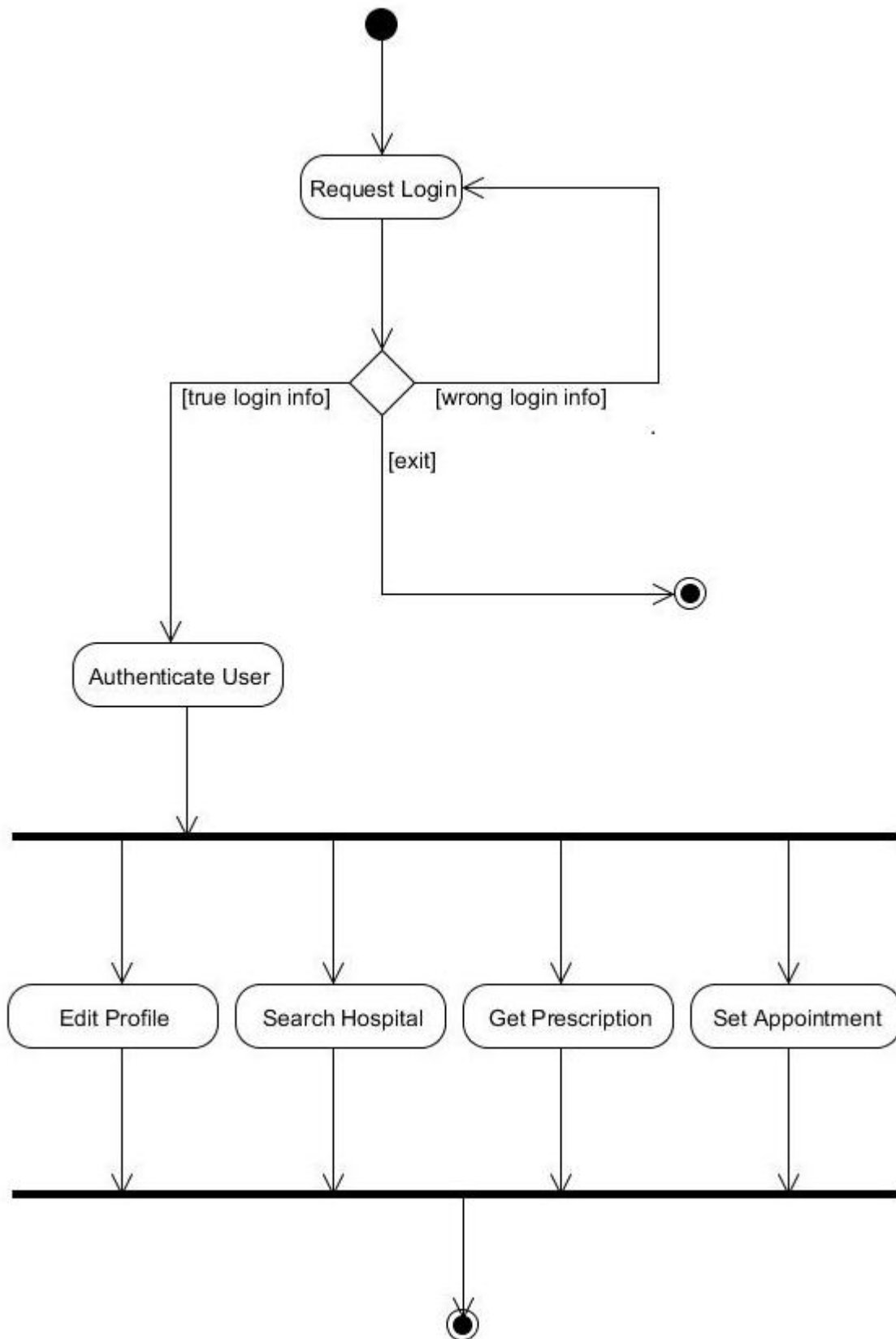


Figure 17 - General Activity Diagram

4.5 Use Case Diagrams

4.5.1 Patient Use Case Diagram

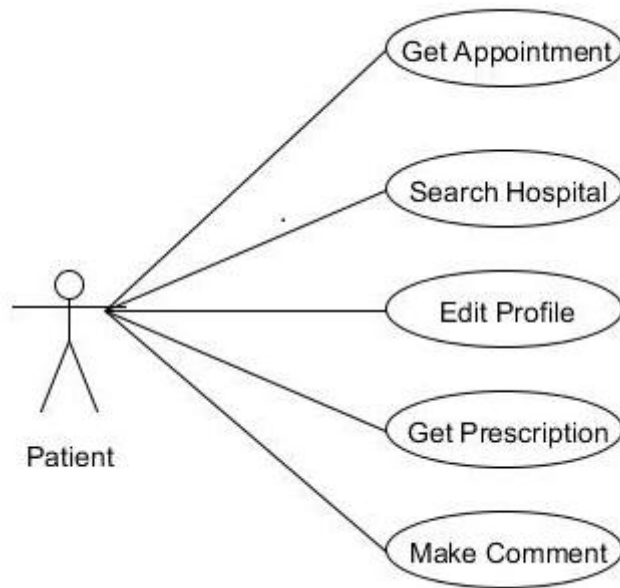


Figure 18 - Patient Use Case Diagram

4.5.2 Doctor Use Case Diagram

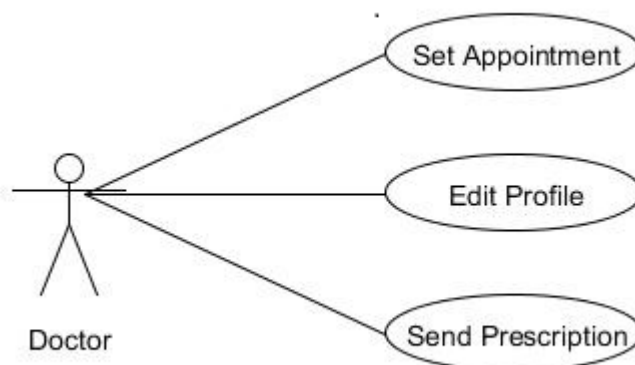
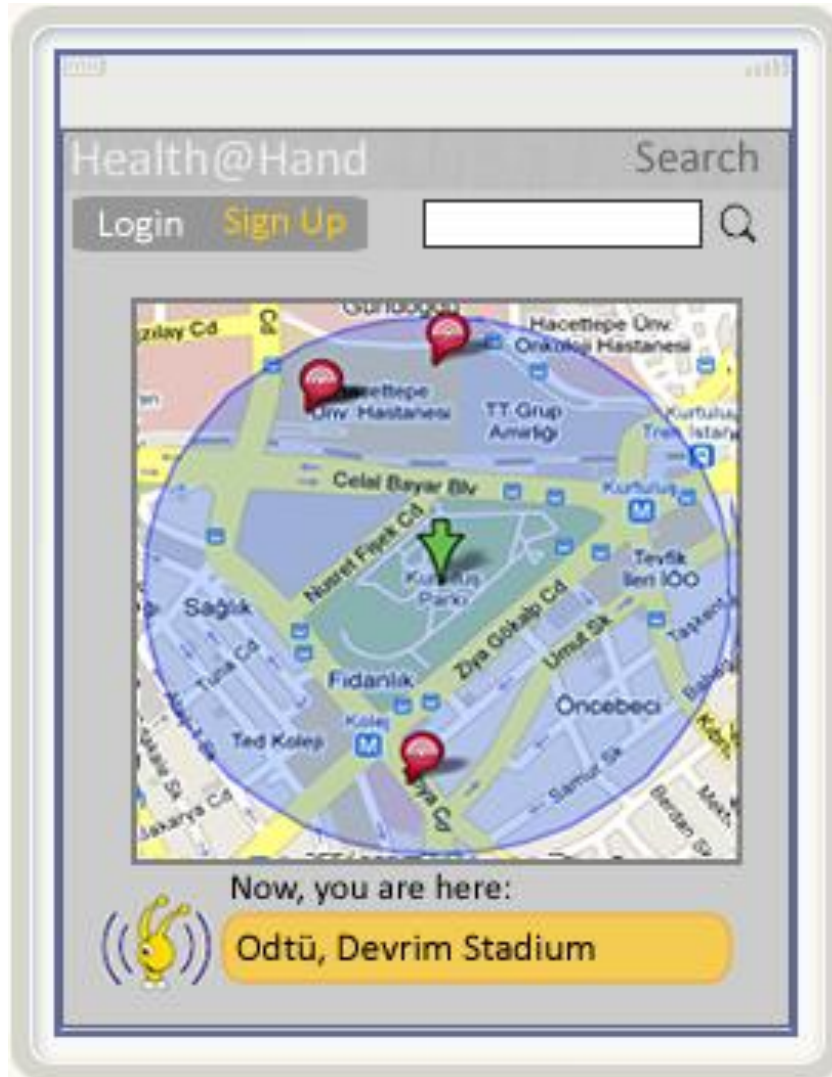


Figure 19 - Doctor Use Case Diagram

5 Interface Design

5.1 Main Screen



Interface 1 - Main Screen

Users are going to see this page at the beginning. There is no need to login to see this page.

Main screen includes four main modules.

- Login and Sign Up links-path to login and sign up screens
- Search input-for searching hospital in the wanted location
- Google-Maps API-shows nearest hospital at the specific location
- Turkcell location module-show where the user is

5.2 Login Screen



Interface 2 - Login Screen

When a user want to login to the health@hand system, this screen will come.

. Login screen includes

- Username
- Password
- Login button-to end login operation
- Sign up link- if not an user, this link give a chance to sign up

5.3 Sign-up Screen

Healt@Hand

Sign Up

TC. Kimlik No

Name

Username

Password

Gender female male

Blood Type

Birth date

Telephone

E-mail



BUTA Inc.

Interface 3 - Sign-up Screen

This screen will be appeared while creating a new account.

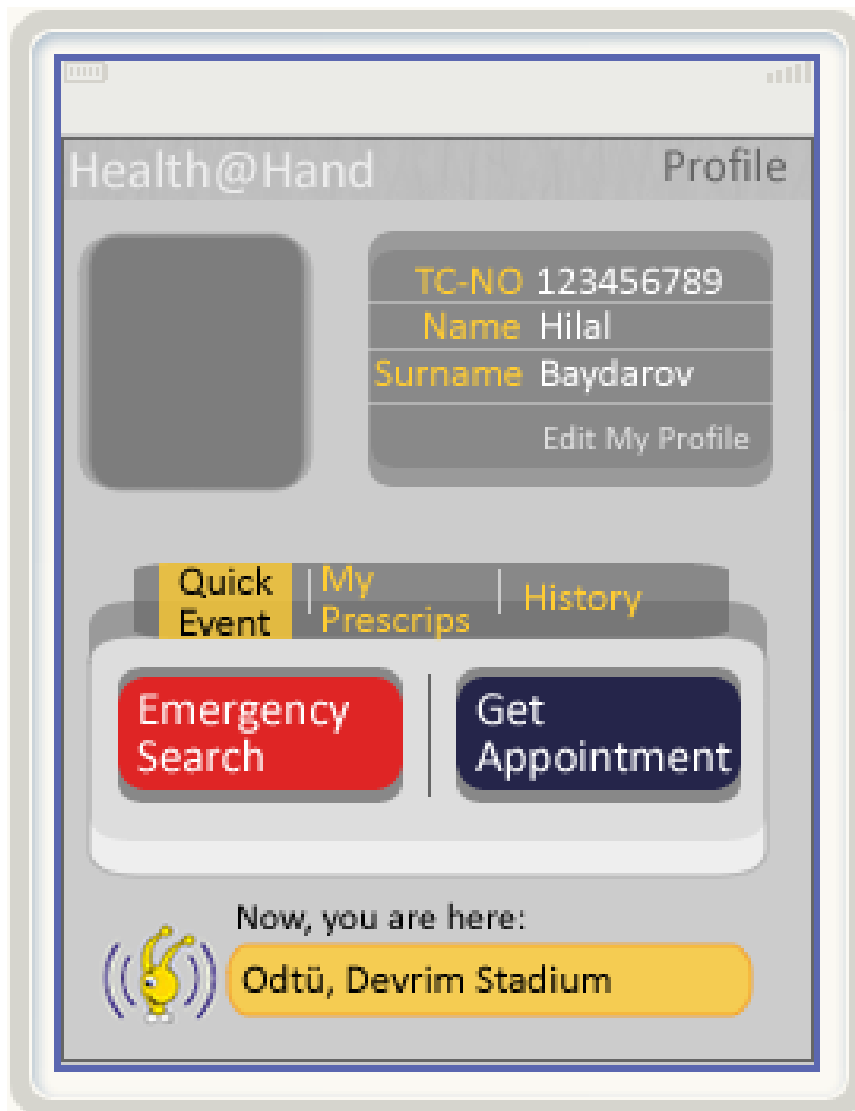
Sign Up Screen

Includes

- TC Kimlik No-this will be used to guarantee uniqueness of new user
- Name
- Username-unique username
- Password
- Gender
- Blood Type-will be used for searching needed blood type

- Birth Date
- Telephone
- E-mail
- Sign-up button-to ending sign up operation.

5.4 Patient Profile



Interface 4 - Patient Profile Screen

When an patient logged in to the health@hand, patient profile screen will be appeared.

Patient Profile Screen

Includes

- Profile picture-includes user's picture

- Profile in formations-includes some important information about User
- Profile quick menu
 - Quick Event- user can go to search and get appointment screen.
 - My Prescriptions-user can see old prescriptions.
 - History- user can see old health@hand logs.
- Now, you are here- shows location with the help of turkcell.

5.5 Doctor Profile Screen



Interface 5 - Doctor Profile Screen

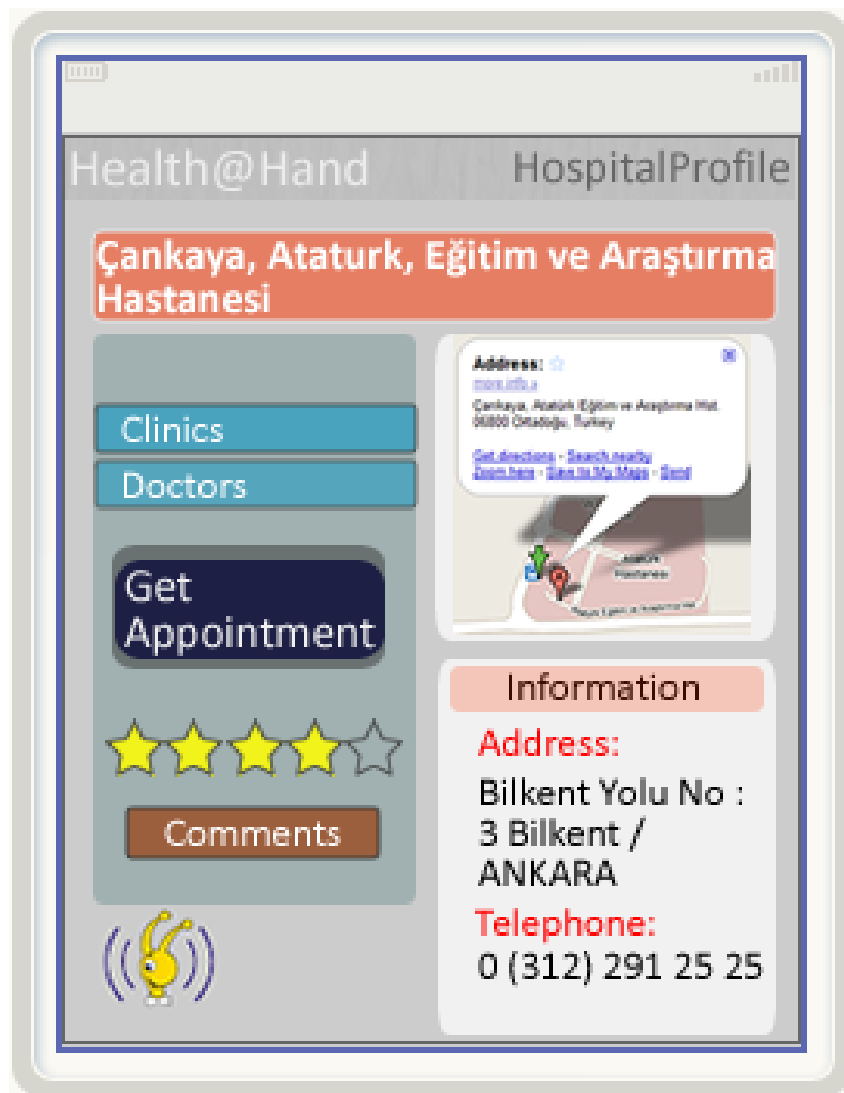
After searching of doctor and clicking doctor profile, this page will come.

Doctor Profile

Includes:

- Doctor picture
- Doctor rating-users can rate doctor
- Get appointment- users can get appointment from the doctor
- Comments-users can see comments written by other users
- Doctor information - show some special and academic information about the doctor.

5.6 Hospital Profile Screen



Interface 6 - Hospital Profile Screen

After search hospital operation and selection of it, this screen will come.
An user can do so many operation on this mode

Like

- Getting appointment-to get appointment from hospital
- Rating-to rate hospital services
- See comments-to see hospital comment written by other users
- Map of hospital
- Hospital information-some crucial information of the hospital.

5.7 Search Hospital Screen



Interface 7 - Search Hospital Screen

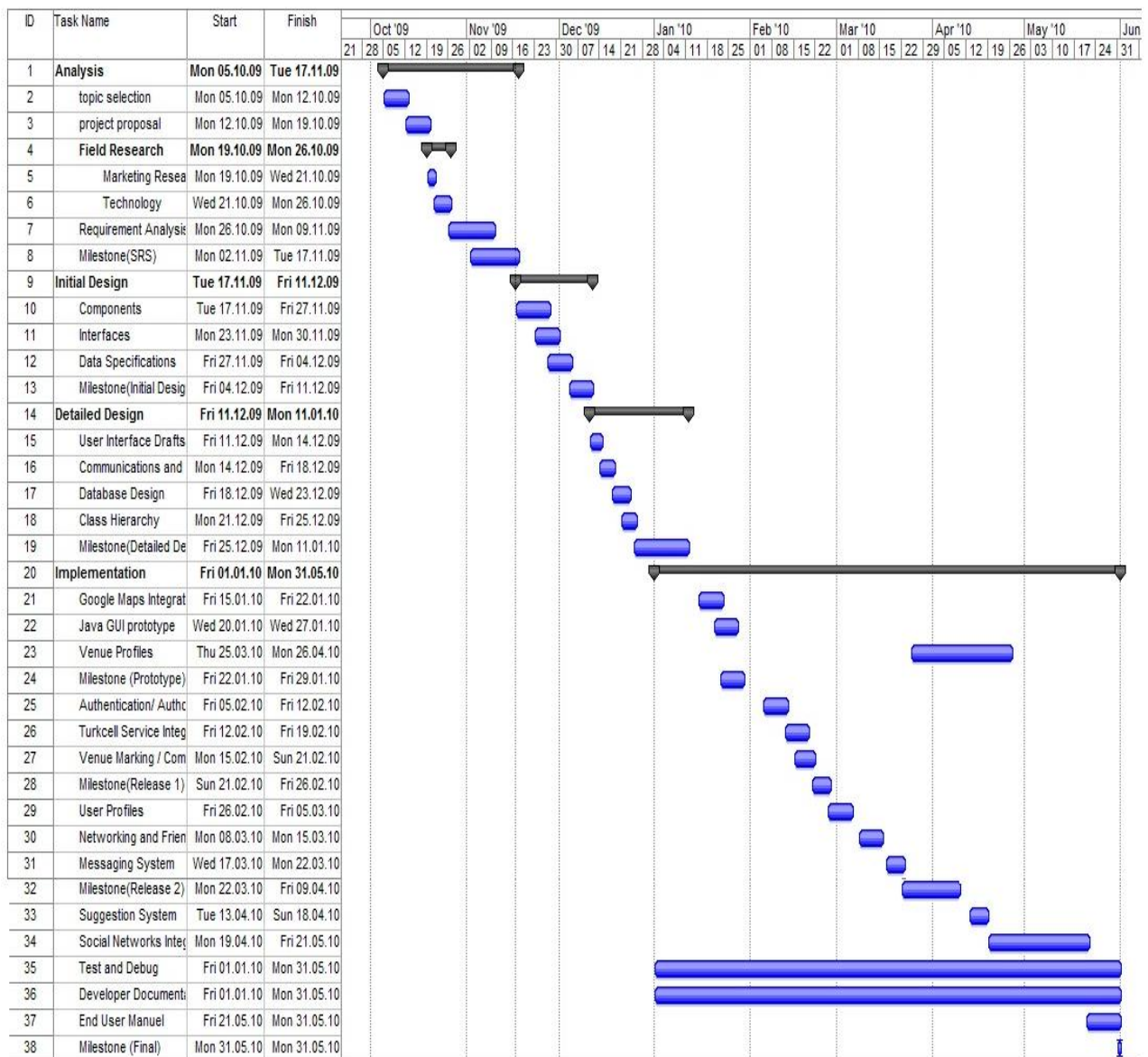
People can search specific data at the screen by choosing one option from the search menu.

Search Screen

Includes:

- Menu - defines search option
- Search text box – the text written to searched
- Google maps API- show nearest hospitals at the specific location
- Now, you are here- shows location with the help of turkcell.

6 Project Schedule



7 Conclusion

7.1 Previous Work

At the beginning of the semester we don't have any idea what would be project topic. After we saw the options that given us mos-health project was seen attractive to us. Then we develop our own idea and make project more detailed. Before SRS, marketing and technology researches were done. Requirements such as functional and non-functional ones, which platform will be used to develop etc. now we just finished initial design report and did our presentation of project to class. GUI design of application was almost finished while writing design report. Designing of our database have been finished early beginning of report. Discussions on functions like logining in, registering, searching and making appointments methods have been finished. With this report details of project design have been composed in our minds.

7.2 Future Work

We are decided to begin to develop our demo of our project. In the demo version of our project we will prepare a login screen, log out screen, registration screen and also we will build a small database that connects through Turkcell API. We will also use Google Maps API to show the position of hospital. In addition, the construction of drug controller, e-prescription controller and the blood type controller will make a different point to our project and it will increase the functionality and efficiency of our project. Because of the one of the main purpose of the project that it will be used by in rural areas mainly, we give a specific attention in the simplicity of this project.

8 References

- Software Engineering A Practitioner's Approach, 5th Edition , Roger S. Pressman
- Component Oriented Software Engineering, Ali H. Doğru
- <http://code.google.com/intl/tr-TR/apis/maps/>
- <http://java.sun.com/javame/reference/apis.jsp>
- Database Programming with JDBC & Java, Second Edition, George Reese