MEDI APP

By

ASAD ATIQUE
(01-134102-013)

SYED MUHAMMAD HAMZA
(01-134102-093)

Supervised By

MR.ADEEL M SYED

2014-2015

A Report is submitted to the Department of Computer Science,
Bahria University, Islamabad.
In partial fulfillment of requirement for the degree of BS (CS)

Certificate

We accept the work contained in this report as a confirmation to the required standard for the partial fulfillment of the degree of BS (CS).

Head of Department

Supervisor

Internal Examiner

External Examiner

Dedication

This work is dedicated to our parents and teachers without their complete support we would not be able to complete this project.

Acknowledgements

First of all we thank to ALLAH who gave us that much strength to full fill this project. After that we are thankful to our family who supported us throughout our whole studies career. Without their support we cannot complete our study career up to here. We would like to thank MR.ADEEL M SYED for his support and supervision. He helped us a lot without his help and guidance we would not be able to complete this project. He was always present for our help and support, despite his busy schedule..We are also thankful to Mrs.SUMAIRA KAUSAR for her support throughout the project.

Abstract

Medical Application is a mobile application developed for the users who want to know about diseases which they have. It will result in precautions from the deadly diseases or it also work like life savior. Project is divided into two major parts which are server side and a client side.

Most of the things such as Add family diseases, view family diseases, diagnose or search, add contacts or user information, send emergency message. Getting nearby hospitals location will do on client side whereas the user's information stored on server. In addition to these features, contact as soon as possible, chats on different forums are also included for the users.

Google Maps has been used for showing the nearby hospitals to the users in the case of any emergency. Data from the clients is encrypted and then stored into the server for further enhancement of the security, so that if data cannot be used by any unauthorized personnel.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Project Overview

Peoples on the earth are divided in to different families. Family's members biologically linked with each other. So family's member's shares some features like, chronic diseases, life styles culture etc. There are some disease that runs from parents to child and their child so on. There should be an application which would help and educate people about such diseases and provide them symptoms and preventive measures.

A mobile application will be developed for the users who would select a disease and after selecting it they would be asked to add family members or even the people who might be suffering or interested in getting to know about the disease from their cellular contact list. A special panic button (Emergency Button) feature is also introduced which will send an SMS to certain people who you would select from your contact list in case of pressing the panic button (Emergency Button).

Another important feature in our application would be that the user will provide the medical history of his family and the symptoms they are suffering from and the symptoms the user suffering from. This application will automatically diagnose the disease after matching the symptom plus it will generate the precautionary measure regarding that disease.

1.2 Problem Description

It was felt that an application was needed which would cater to the users in a personal way by acquiring some basic information of the user's personal and their family diseases helping the user to acquire sufficient preventive and symptomatic measures they will be asked to add family or other members from their contact list. There would be two methods used for sending messages (Web and Sms Bundles).A panic (Emergency button) will be added in the app which will trigger a series of (Help Me) messages. This panic button can also be used in case of any emergency for example (accident, theft, stuck in elevator etc) situation. A web forum will also be made where people can ask questions and share their expertise.

1.3 Project Objective

To develop a mobile based application that would cater as a medical Aid and informative application with an emergency button using location API.

1.4 Project Scope

Project will focus on the people who are having family diseases or may be interested in getting to know about certain diseases and their preventive measures a large number of individuals are interested which has been confirmed by a survey carried within Islamabad. Information regarding diseases, preventive measures and contacting doctors will be focused whereas Google maps and GPS will be integrated in the applications feature. Application can evolve later on into a strong information source and social networking app for concerned individuals including feature like trending diseases in blog or forum area of particular diseases for keen users. This application will be able to inform the user about the nearby hospitals and clinics In case of emergency using Google ma

Chapter 2

Literature Review

2.1 Overview

Most people have their family diseases transform from genes to genes like cancer, diabetes, some other weaknesses like breathing problems etc. So there should be some methodology or technique to aware the peoples about these diseases so they can get cure at right time. Our application educates people about such diseases and provides them symptoms and preventive measures. This mobile application developed for the users who would select a disease and after selecting it they would be asked to add family members or even the people who might be suffering or interested in getting to know about the disease from their cellular contact list.

2.2Chronic Diseases APP Examples

2.2.1 My Plate Calorie Tracker

There's no denying that people with Cohn's disease have dietary challenges. Live strong's My Plate Calorie Tracker is designed to keep track of calories and exercise. Making sensible decisions is easy with My Plate. A vast database of foods and restaurant items allows you to easily learn more about what you're about to eat. The app also calculates the number of calories you're likely to burn and helps you stay on track with weight and fitness goals. You can even access support groups and forums to help you in your quest for better health.

2.2.2 GI Monitor

When it comes to autoimmune diseases, providing accurate personal information to your doctor is vital. One patient went searching for an app that would help him do just that. When he couldn't find one, he created GI Monitor.

With real-time symptom tracking, this app can help you find the links between dietary or lifestyle choices and symptoms, or between symptoms and treatments that work. You can even sync your data across multiple platforms. Printed reports will provide your doctor with more details about your life with Cohn's, including what's working for you and what's not.

2.2.3 MyIBD

Managing a chronic disease is easier when you can chart your progress and make evidence-based decisions. The Hospital for Sick Children created myIBD for patients and doctors to get a better handle on treatment for inflammatory bowel diseases, including Cohn's.

With this app, it's fast and easy to record details about symptoms, appetite, and trips to the bathroom. You can view your information in text or graph form at any time. Additionally, the app keeps track of all of your medications, including alternative meds to help you avoid possible drug interactions.

2.2.4 GI Buddy

Chronic illnesses require lots of cooperation between patients and their doctors. Having the information you need at your fingertips puts you in a better position to manage your Cohn's. That's why the Cohn's and Colitis Foundation of America came up with GI Buddy. With this app, people with IBD can easily share information with doctors.

The app is loaded with features to help you track symptoms, treatments, and diet. Check your data by week, month, or year to detect possible patterns, and have your reports emailed to you.

2.2.5 I Triage Health

This application is created by two ER doctors. This application provides users with a large database having information about the diseases. More than 12million peoples have already downloaded this application that has a 4.5 star rating from over 100K reviews.

2.3 Drawbacks

Some of the drawbacks of existing systems are as follows:

- Not provide medical history of patient
- No information about diseases
- No provided information of non-smart phone user (only smart phone user)
- No Location API
- No emergency scenario

2.4 Our Approach

Domain of the application is the medically concerned people. They will have access to the knowledge of their particular interest of disease, also allow you to add family members and friends. Application will also be helpful for any kind of emergency one could fall into and contact concerned through panic button feature. Forums will provide and answers and allows users to form social interaction among each other's.

Chapter 3

Requirement Specification

3.1 Existing System

There are different applications like this such as Diseases Dictionary Medical, Handbook of diseases. But these applications have limited resources like Diseases Dictionary Medical only shows information about the diseases. Handbook of diseases will only show the list of diseases and precautions.

3.2 Proposed System

A mobile application will be developed for the users who would select a disease and after selecting it they would be asked to add family members. A special panic button (Emergency Button) feature is also introduced which will send an SMS to certain people who you would select from your contact list in case of pressing the panic button (Emergency Button).

Another important feature in this application would be that the user will provide the medical history of his family and the symptoms they are suffering from and the symptoms the user suffering from. This application will automatically diagnose the disease after matching the symptom.

3.3 Functional Requirements

Functional requirements are used to define the functions of the system and have information about its all components. With the help of functional requirement user can get the goal of the whole system. Functional requirement is always leading with the non-functional requirements. Functional requirements have their own unique functional numbers.

3.3.1 Login Module

The Login management modules to secure the application from the unauthorized user or enhance the security level. First of all user should login to the system so he can use all the services provided by the system. Unauthorized users will not get access to the account. User should register himself in order to login to the account.

3.3.2 Sign up Module

Signup will be necessary for every user. Then steps like verification and authorization will be done from the server side. Signup will be the first step for every user. System will signup user from the server and then authorize it.

3.3.3 User's Information Module

Information about user's disease/people added to the group. All the information of the peoples who will be contacted will save in the local database.

3.3.4 GPS and Location Module

GPS and location management will be handled in this phase. Using Location API application will be able to show the locations of the nearby hospitals by using Google maps.

3.3.5 Searching Module

In this module such algorithms will be implemented that will list diseases.

3.3.6 Contacts Retrieving Module

This module will help user to retrieve contacts from internal directory. After retrieval contacts will be saved on local database. So at the time of need it would be used to send emergency message.

3.3.7 SMS Sending Module

This module will help user to send sms to the specific user. User must have link to the same application.

3.4 Non-Functional Requirements

Some Non-Functional Requirements are as follows:

3.4.1 Functionality

All the functionalities that have been mentioned in this system will be provided to the users.

3.4.2 Security

Login and signup processes will help in order to gain high level security. Users have been authorized from the web server that is developed with high security protocols so no one can breach it. Android provides high level security so no one can access it directory structure. Android based on Linux kernel so it is even impossible to access Linux directory structure. So android provide high level security by their own.

3.4.3 Consistency

The system has consistent interface, which has been designed user friendly with proper guide manual. All functionalities performed by the system are consistent.

3.4.4 Efficiency

The time required by this system is far less than as compared to other apps like that; friendly interface results in easy understanding of the system resulting in time saving.

3.4.5 Usability

System's interface is very easy to understand and consistent, while designing it has been specially kept in mind to remain it simple to provide conformity to the user.

3.5 Dependencies

The dependencies of the system are as below:

3.5.1 Wi-Fi Connectivity

System would not be able to login or signup without Wi-Fi connectivity. Wi-Fi will be required to login form the server. Wi-Fi is also required to show the Google Maps on the layout. Without Wi-Fi connectivity Google Maps will not load on the user interface.

3.5.2 Android OS

All the users or family member must be using android operating system only in this way they can communicate through this app. This app is android base so it will not run on any other operating system.

3.6 Use Case

The Generic Diagram of use case is as follows:

Figure 1 Use Case : Entire System

3.6.1 Use Case Login

Table 1 Use Case 1: Login

Title	Login	
Version No.	1.0	
Actors	Admin, User	
Description	1	This Use Case describes the entry of an Admin in admin panel.
	2	This Use Case describes the entry of User in client panel.
Trigger	The system will login the user and display the specific panel on the basis of username and password.	
Main success scenario	Step	Action
	1	First enter user name
	2	Second enter password
	3	Press Submit button
Alternative Flow	Step	Action
	1	In case of incorrect password the user will not be logged in.
Assumption	The user is already registered.	
Pre-Condition	User credentials are already present on the server.	
Post-Condition	Main screen of the respective panel will appear depending upon the username and password	

User Interface	Layout with Email and password having fields.

3.6.2 Use Case Logout

Table 2 Use Case 2: Logout

Title	Logout	
Version No.	1.0	
Actors	Admin, User	
Description	1	This Use Case describes the logout of admin in admin panel.
	2	This Use Case describes the logout of User in client panel.
Trigger	Login Page will be displayed by the system.	
Main success scenario	Step	Action
	1	Press logout
Alternative Flow	Step	Action
	1	None
Assumptio n	User will be logged out.	
Pre-Conditi on	User was logged in.	
Post-Condi tion	Login page will be displayed	
User Interface	Same layout will be displayed through which user logged in.	
Issues	None	

3.6.3 Use Case Sign Up

Table 3 Use Case 3: Sign up

Title	Sign Up	
Version No.	1.0	
Actors	Admin, User	
Description	1. This Use Case describes the registration of an Admin in admin panel. 2. This Use Case describes the entry of User in client panel.	
Trigger	The system will display the Registration page carrying fields for the registration.	
Main success scenario	Step	Action
	1	Enter values in the fields
	2	Fill all fields
	3	Sign up button will be pressed for the information to be saved in the server.
	4	App will automatically take control to the login screen.
Alternative Flow	Step	Action
	1	If the entered data is not according to the format described, the system will ask the user to reenter the values.
Assumption	Entries will be saved on the server.	
Pre-Condition	Admin must be logged in.	
Post-Condition	Message box will appear acknowledging that the user have registered successfully on the server.	
User Interface	Page with multiple fields having specific formats.	
Issues	Invalid values.	

3.6.4 Use Case User’s Information

Table 4 Use Case 4: User’s Information

Title	Enter User’s contact number to local database	
Version No.	1.0	
Actors	User	
Description	This Use Case is showing how contact will be added by the user.	
Trigger	The system will display the Contacts detail page carrying fields for the contacts.	
Main success scenario	Step	Action
	1	Enter values in the required fields
	2	Fill all fields
	3	Save button will be pressed for the values to be saved in the local database.
	4	Menu button will take admin back to the main menu.
Alternative Flow	Step	Action
	1	If the entered data is not according to the format described, the system will ask the user to reenter the values.
Assumption	Contacts will be saved on the server.	
Pre-Condition	User must be logged in.	
Post-Condition	Message box will appear acknowledging that the entered data has been saved on the local database.	
User Interface	Page with contact field having a proper format.	
Issues	Invalid values.	

3.6.5 Use Case Add Disease

Table 5 Use Case 5: Add disease

Title	Add Disease	
Version No.	1.0	
Actors	User	
Description	This Use Case is showing how to add list of diseases to the database.	
Trigger	System will show name of the diseases with the checkboxes in front of them. User can check the checkbox so the selected disease will be transfer to the further diagnose.	
Main success scenario	Step	Action
	1	Get list of the diseases.
	2	Click on checkbox.
	3	Save button will be pressed to save the selected data.
	4	Back button will be pressed in order to go back to the main menu.
Alternative Flow	Step	Action
	1	If user will not check any box no data will be transfer to further diagnose.
Assumption	It will show nearby hospitals names and tags on Google Maps.	
Pre-Condition	Select any area.	
Post-Condition	Web View will show you the hospitals locations	
User Interface	Page having one field and one web view.	
Issues	Invalid area name.	

3.6.6 Use Case GPS Location

Table 6 Use Case 6: GPS Location

<table>
<tr><td>Title</td><td colspan="2">Find GPS Location</td></tr>
<tr><td>Version No.</td><td colspan="2">1.0</td></tr>
<tr><td>Actors</td><td colspan="2">Users</td></tr>
<tr><td>Description</td><td colspan="2">This Use Case is showing how user will get the location of the nearby hospitals by using Google map API.</td></tr>
<tr><td>Trigger</td><td colspan="2">The system will show a field that require name of the specific area. Web View on the same page will load the map and show nearby hospitals on the Map with tags.</td></tr>
<tr><td rowspan="5">Main success scenario</td><td>Step</td><td>Action</td></tr>
<tr><td>1</td><td>Enter name of the area.</td></tr>
<tr><td>2</td><td>Fill field.</td></tr>
<tr><td>3</td><td>Go button will be pressed in order to get the specific map of the area.</td></tr>
<tr><td>4</td><td>Back button will be pressed in order to go back to the main menu.</td></tr>
<tr><td rowspan="2">Alternative Flow</td><td>Step</td><td>Action</td></tr>
<tr><td>1</td><td>If the name of the area is incorrect it will show to a message to enter correct name.</td></tr>
<tr><td>Assumption</td><td colspan="2">It will show nearby hospitals names and tags on Google Maps.</td></tr>
<tr><td>Pre-Condition</td><td colspan="2">Select any area.</td></tr>
<tr><td>Post-Condition</td><td colspan="2">Web View will show you the hospitals locations</td></tr>
<tr><td>User Interface</td><td colspan="2">Page having one field and one web view.</td></tr>
<tr><td>Issues</td><td colspan="2">Invalid area name.</td></tr>
</table>

3.6.7 Use Case Search Disease

Table 7 Use Case 7: Search Disease

Title	Search Disease	
Version No.	1.0	
Actors	User	
Description	This Use Case is showing how user will search the disease.	
Trigger	The system will display the different diseases that will match the symptoms.	
Main success scenario	Step	Action
	1	Enter in the specific layout.
	2	Match Diseases
	3	Search for diseases.
	4	Back button to get back to the main menu.
Alternative Flow	Step	Action
	1	It will only let user know about the disease that already present in the web server.
Assumption	It will get diseases from the database.	
Pre-Condition	User must be logged in.	
Post-Condition	Message will be shown having list of diseases.	
User Interface	Page having interface for disease search.	
Issues	Invalid values.	

3.6.8 Use Case Sending Information

Table 8 Use Case 8: Sending Information

Title	Sending SMS	
Version No.	1.0	
Actors	Users	
Description	This Use Case is showing how user sends sms to the relatives having information about the diseases.	
Trigger	The system will display the page having scenario how to send sms to the required members.	
Main success scenario	Step	Action
	1	Enter contact number or select from the existing contacts number.
	2	Send sms.
Alternative Flow	Step	Action
	1	System will also be able to send the sms to the specific contacts that already stored in the database.
Assumption	Sms will send to the specific users.	
Pre-Condition	Users contact number should be in the database.	
Post-Condition	Sms sent successful message box will be shown.	
User Interface	Page with layout that is able to send sms.	
Issues	Invalid contact number.	

Chapter 4

System Design

4.1 System Architecture

System functionality of the system is given below in diagrammatical form. In this project android phone will be work as client and user can get detail about disease by using his phone and share those details with other fellows. System will send messages like i.e. Panic messages, contact ASAP etc. After sending the specific data the user can add (Family diseases, View Family diseases, add Contact etc.) in to local data base. Other user that wills currently using this application can get shared data on the click of a button. The entire scenario is discussed below in Figure 4.1 Architecture Diagram.

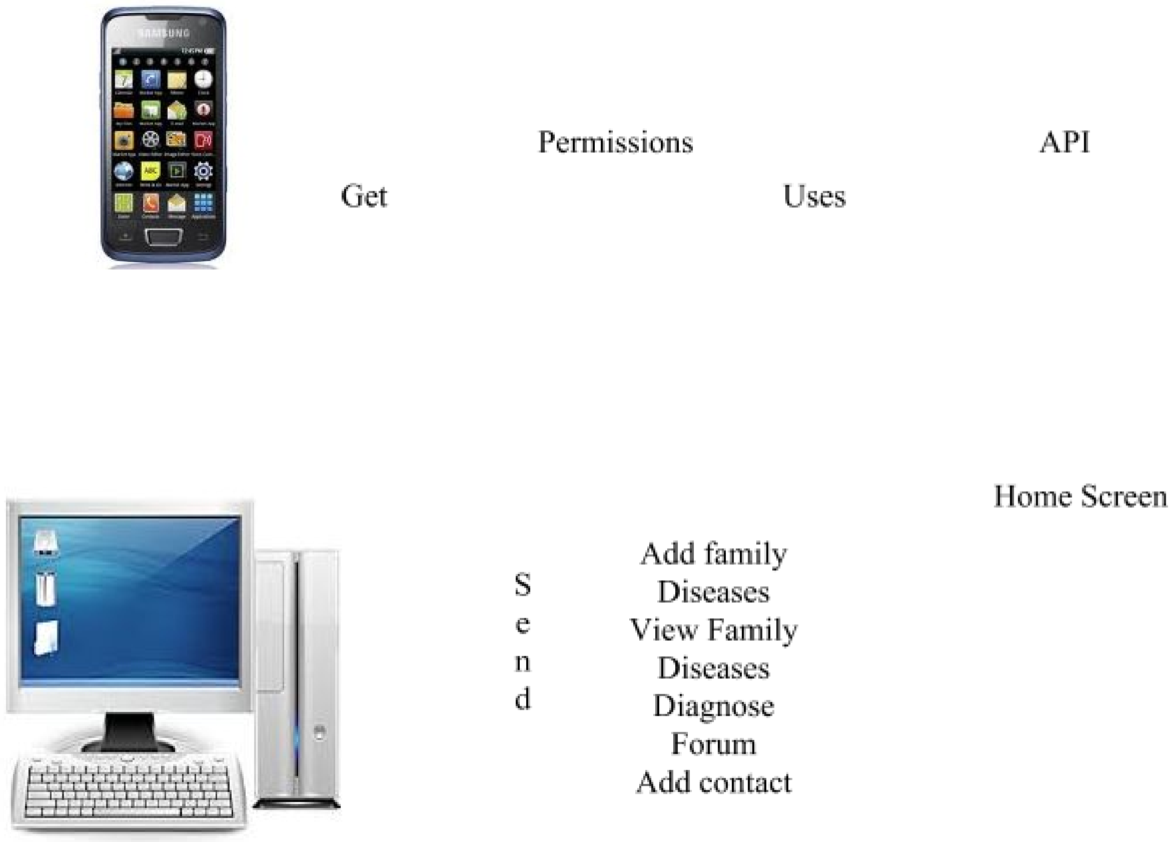


Figure 2 Architecture Diagram

4.1.1 Functionality

First of all user will sign up and login from remote server. After Login user will use all the functionalities of the application. Like user can add diseases to the database. User can Add members to the database user can diagnose the disease user can go to different forums. User can also send emergency sms to the members. User can also get the location of the nearby hospitals on the map.

Server

Server will run for establishing the connection with the client. Wi-Fi is necessary for this whole process. After establishing connection with the client server will be ready to get credentials of the user and save it to database.

Client

Client will also establish connection with the server. Wi-Fi is necessary for connectivity. After establishing connection with the server. Application will allow user to login by using his correct email and password. Instead of login all the working will be done on the server.

4.2 Database Design

Database design of the system is as follows:

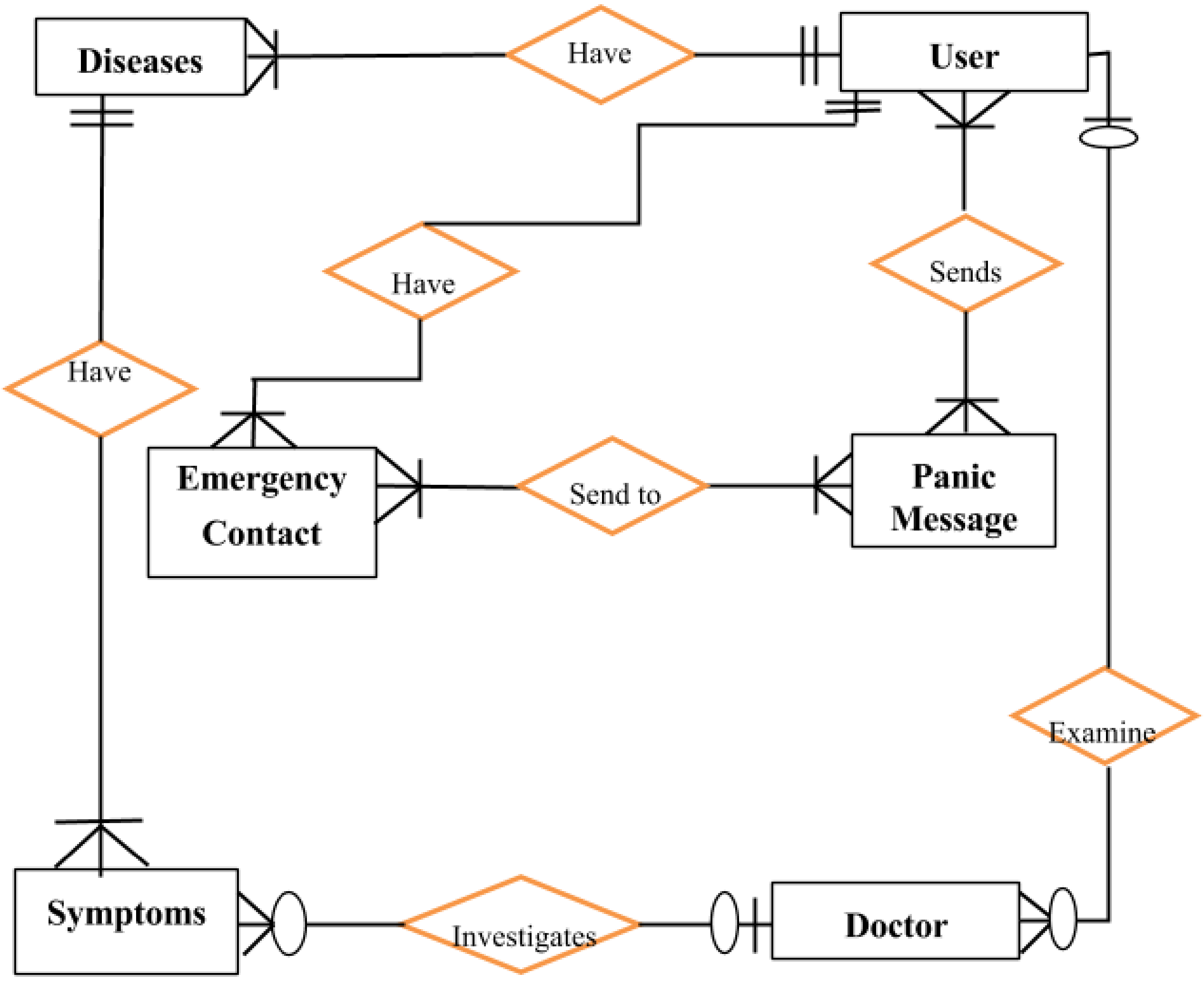


Figure 3 Database Design

4.3 ER Diagram

ER Diagram of the system is as follows:

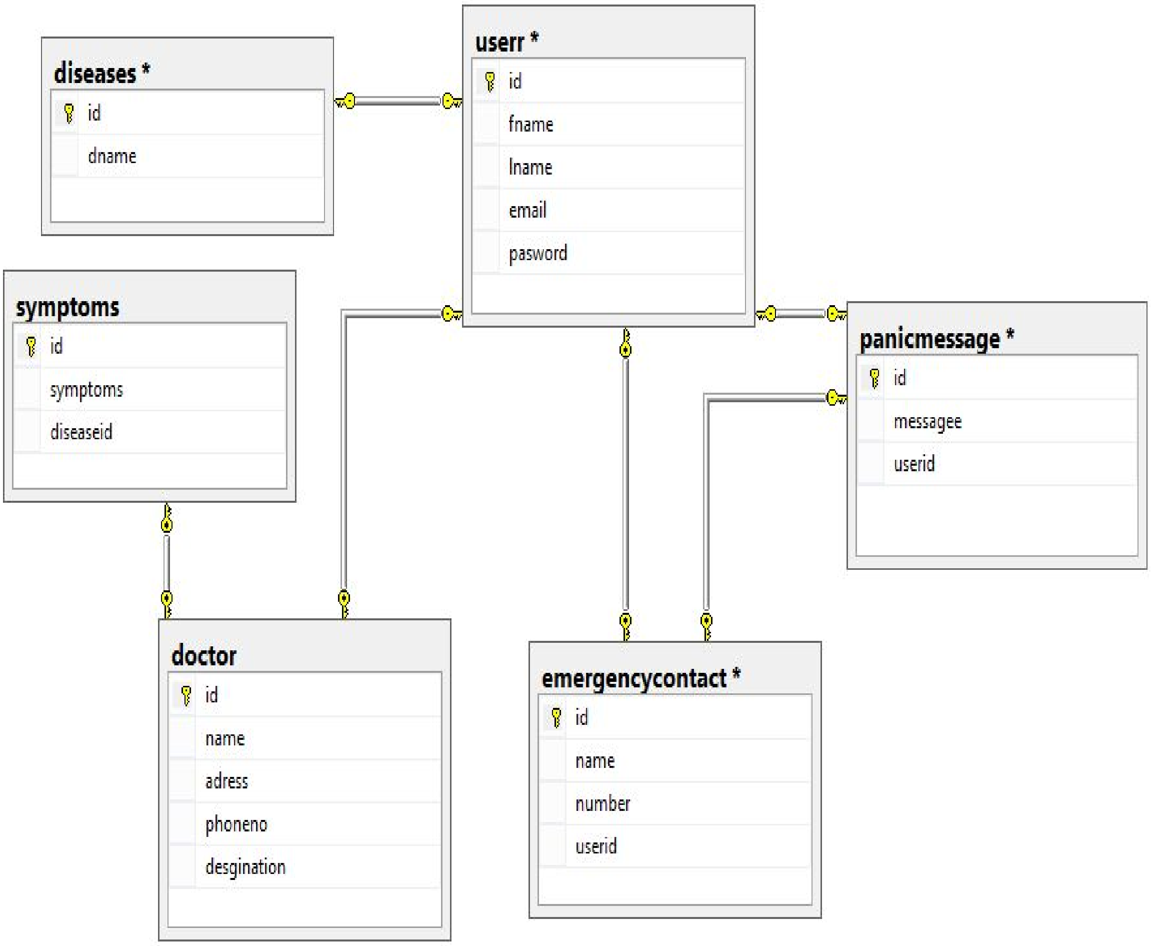


Figure 4ER-Diagrams

4.4 Sequence Diagram

4.4.1 Login

This sequence diagram shows the process of authentication. There is security reason due to which only authorized members should be allowed to enter in the system. For that one has to be implement login scenario. User must enter their valid email address and id in order to get authorized. Sequence diagram will show the sequence of the foreground and background tasks. Login Sequence diagram is described and briefly discussed in the figure no. 4.3 given below

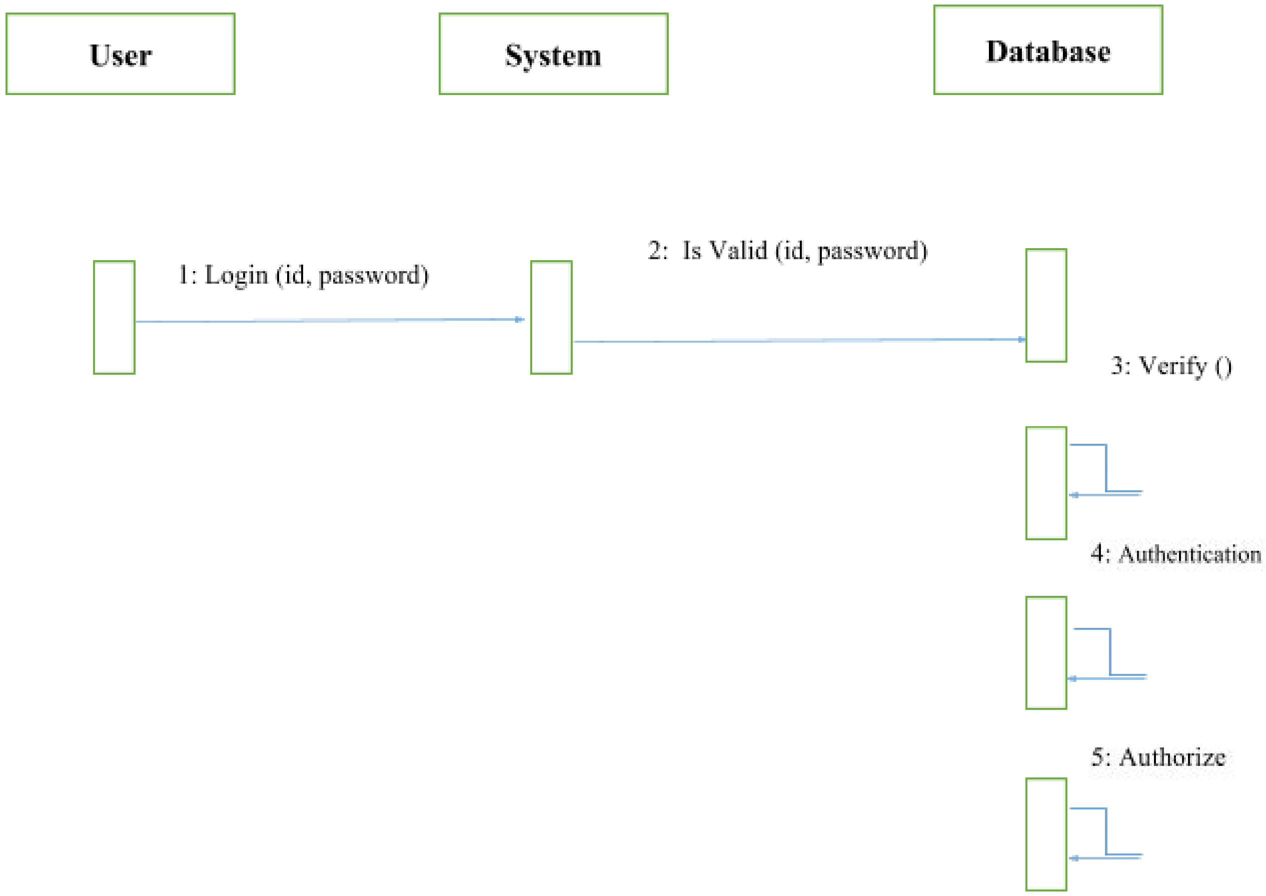
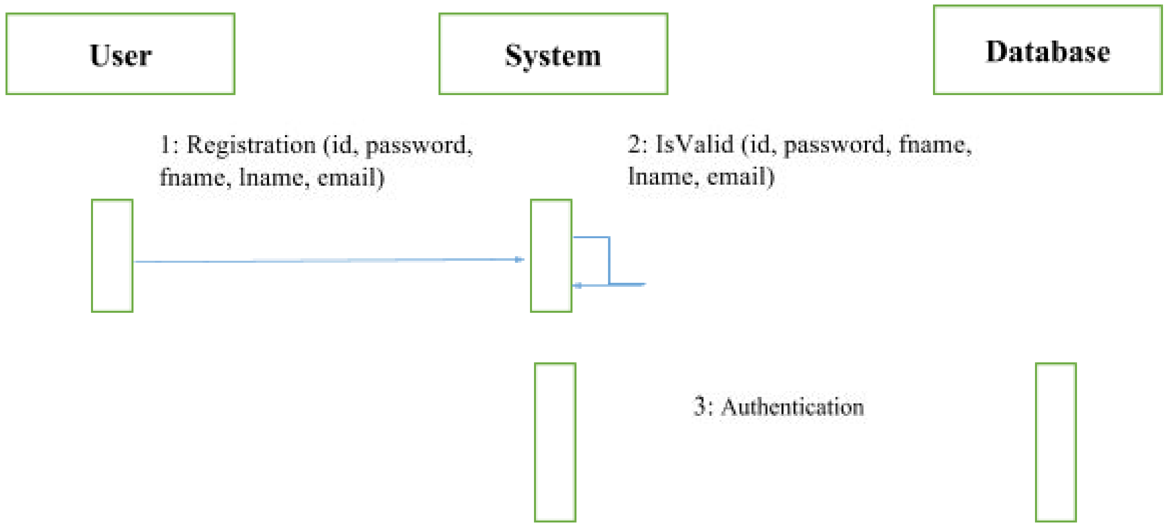


Figure 5 Sequence Diagram: Login

4.4.2Sign Up


User
System
Database
1: Registration (id, password, fname, lname, email)
2: IsValid (id, password, fname, lname, email)
3: Authentication

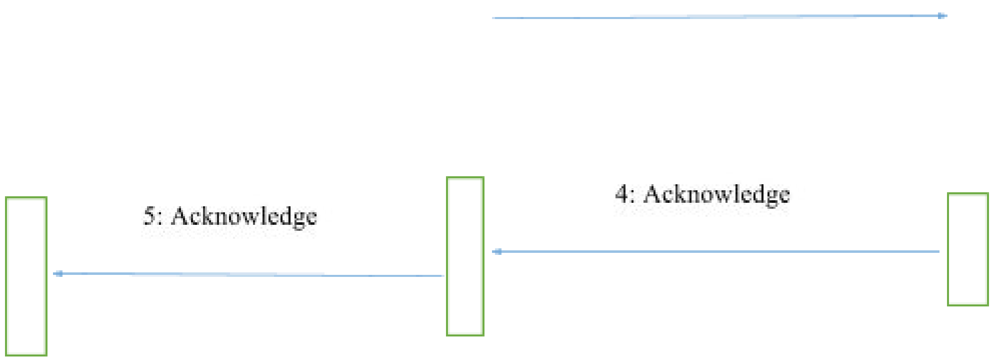


Figure 6 Sequence Diagram: Sign Up

4.4.3 User Information

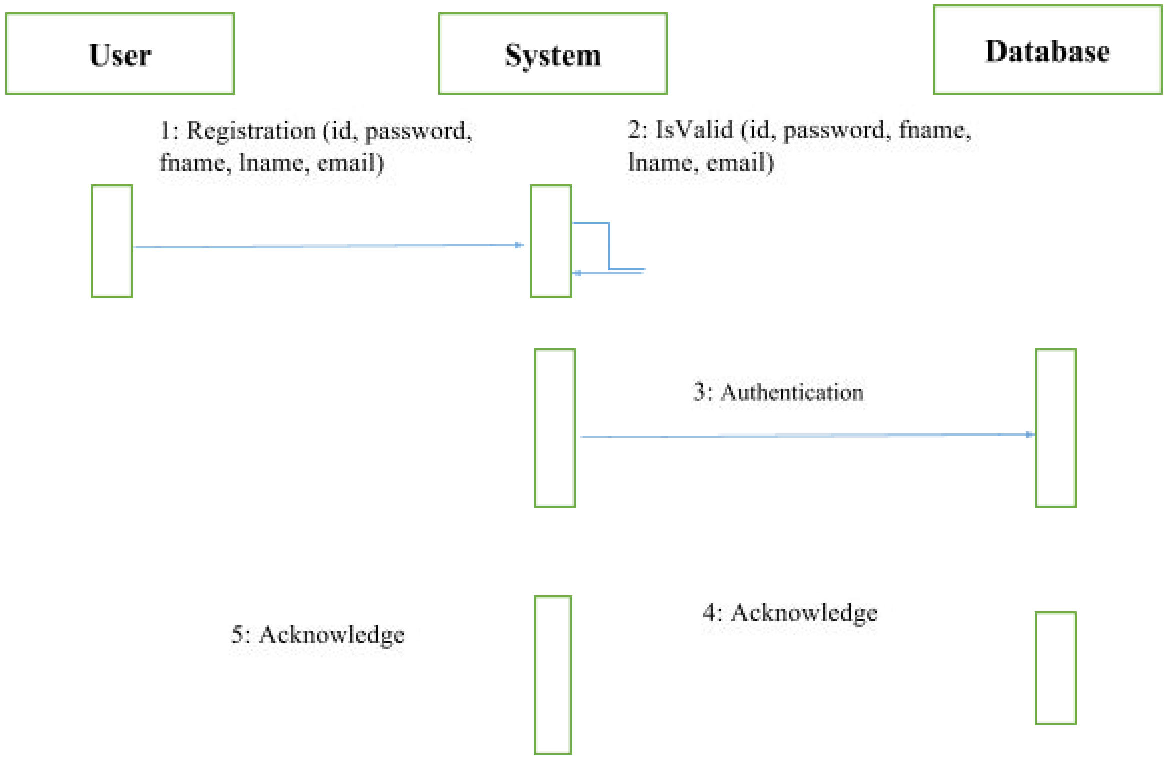

User
System
Database
1: Registration (id, password, fname, lname, email)
2: IsValid (id, password, fname, lname, email)
3: Authentication
5: Acknowledge
4: Acknowledge

Figure 7 Sequence Diagram: User Information

4.4.4 Searching

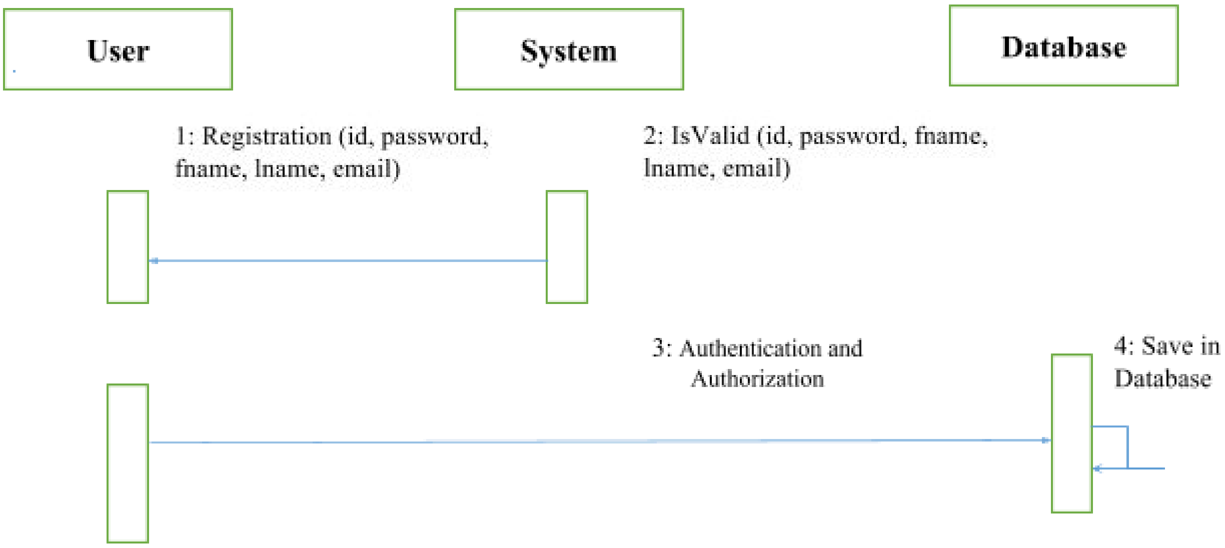

User
System
Database
1: Registration (id, password, fname, lname, email)
2: IsValid (id, password, fname, lname, email)
3: Authentication and Authorization
4: Save in Database

Figure 8 Sequence Diagram: Searching

4.4.5 Contract Retrieve and Sms Sending

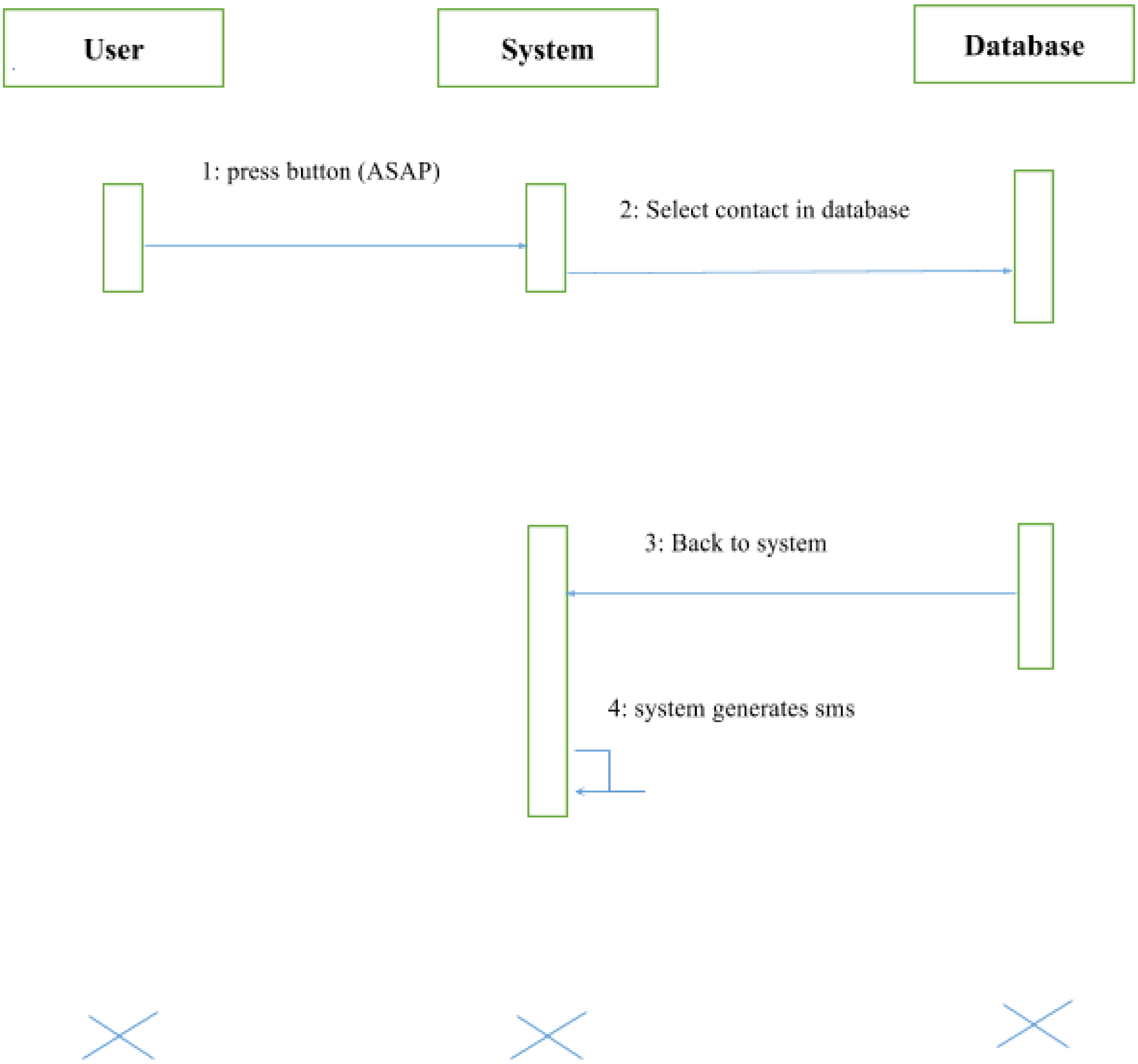


Figure 9 Sequence Diagram: Contact Retrieve and sms sending

4.4.6 Searching or Diagnose

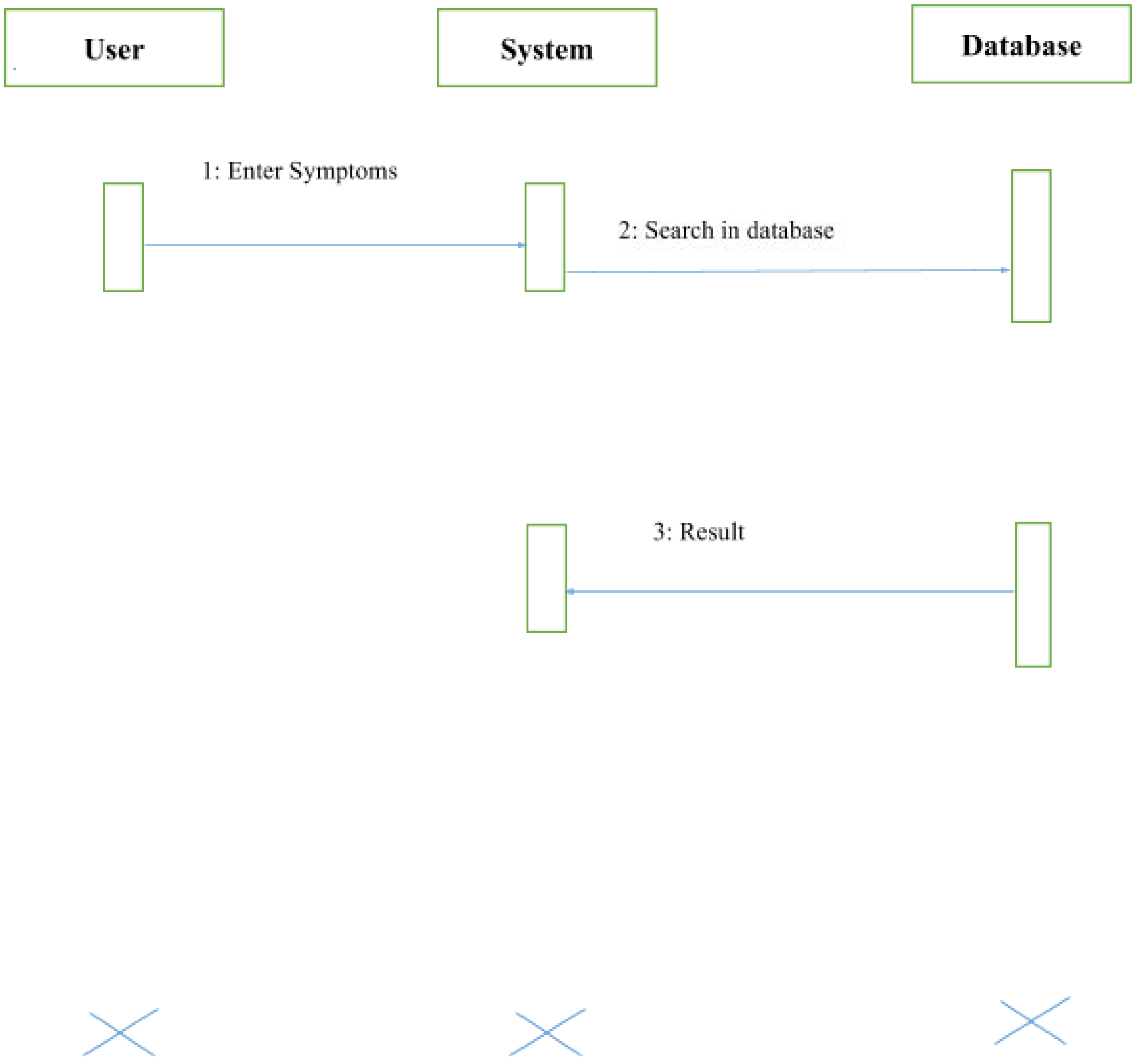


Figure 10 Sequence Diagram: Sms Sending

Chapter 5

System Implementation

5.1 System Architecture

There are two main components of this system:

5.1.1 Client Side

Client side is basically responsible for the sending request to the server. After establishing connection to the server client starts requesting to the server and its server responsibility to fulfill the clients request. Client send request for login authentication to the server and server will authenticate user from the database and send acknowledgement to the client.

5.1.2 Server Side

Server will receive request from the client and fulfill their request. Server can also receive multiple requests from the client and have to fulfill these requests. In this case server will receive information of the user's and store it into database.

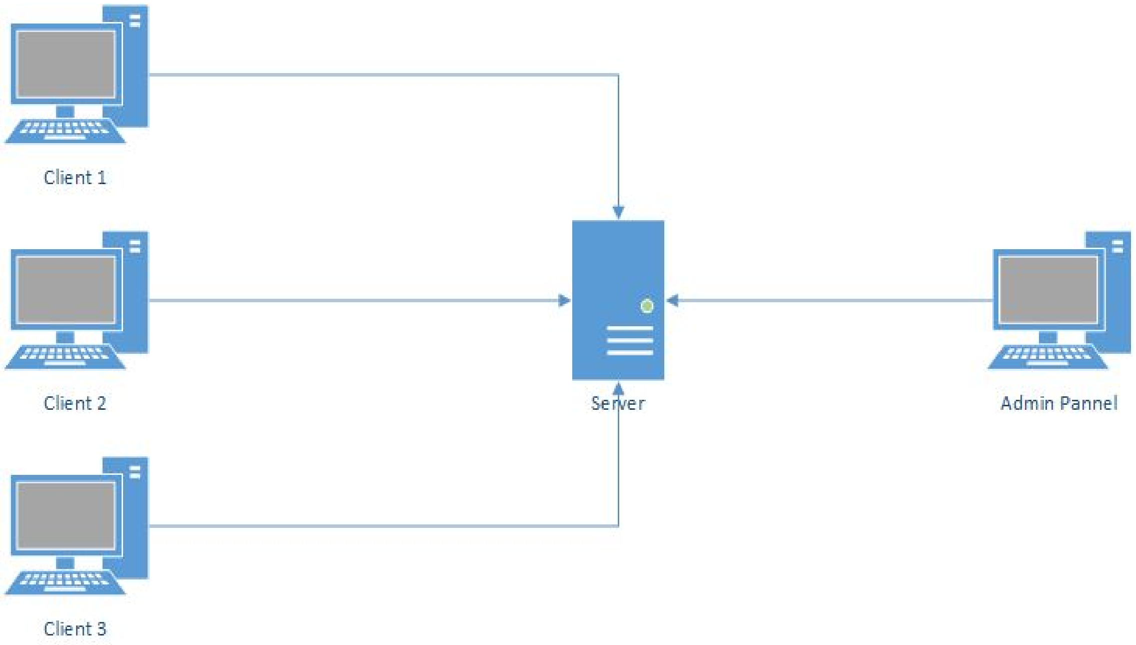


Figure 112 System Architecture

5.2 Tools and Technologies used

5.2.1 Tools

5.2.1.1Android SDK

5.2.1.1.1 Android version history

Android is an open source operating system. Its first version is developed in November 2007. Google owns android. Android releases many versions like cup cake, froyo jelly beans, kit Kat etc.

5.2.1.1.2 Android Software Development

Android software development is the process of creating applications that runs on android platform. Basically eclipse and android studio is used for android development. Android studio is basically developed for android development but eclipse is for java. Eclipse provides plug-in for the android development.

5.2.2 TECHNOLOGIES

5.2.2.1 Android (Operating System)

Android based on Linux Kernel. Android is an open source operating system and developed by Open Handset alliances (47 technological companies). Open handset alliance is basically a group of 47 technology companies who thinks that there should be an operating system that should be less expensive and have high user experience.

5.2.2.2 Android Open Source Project

Firstly open hand set alliance develops android and releases it as open source. Android based on Linux kernel. Linux is also open source operating system. Then Google owns android and start building its versions. First android version was released in 2007.

5.2.3 Features

Some features and specifications of android operating system.

Storage

This project will use window storage for storing large amount of data.

Connectivity

Android supports different connectivity technologies like Bluetooth and Wi-Fi.

Java Support

Android uses java for coding. Eclipse is used for development of android. So java is used for android development.

Java

Java is an object oriented language. It is a high level language developed for high level development. Java is used for both types of development desktop as well as web development. Eclipse and net beans are used for java development.

Chapter 6

System Testing and Evaluation

6.1 System Testing

6.1 Objective

Testing is the next phase of the software development process after coding. Testing and debugging is the main part of the software development. Testing will be done by different approaches. Debugging is the process of finding and removing error. Whole system will be dividing into different modules and each module will be tested independently. In testing process results of the different things will be find out like, errors, response time, ambiguities etc.

6.2 Test Cases

Test cases check the functionality of the each use case. Test cases are executed according to the sequence of use cases defined earlier in the report. The results taken after execution compared to the required result and present it as pass/fail or not executed. The test cases are given below:

6.2.1 Component of Testing

Different modules will be tested in this phase. Bugs will be finding out and resolved.

6.2.2 Integration of Testing

Whole system is divided in to different modules or sub systems. And these modules will be tested separately to generate results. All modules will be combined to make sure that whole system will work fine.

6.2.3 User Testing

User will involve in this testing stage. User will provide some known input to the system and system will provide required output. If the output will be the same that is requires then its mean system is free of errors. If output didn't match the required output then system needs to be debugged. This is called acceptance testing.

6.1.1 Login

Table 9 Test Case 1: Login

Test Case ID		TC_FUNCT_01	
Description		Tests the Login Screen	
Applicable for		Disease Identifier system	
Requirements		User should already registered and have username and password.	
Initial Conditions		Android phone should be working properly.	
Step	**Full / Regr**	**Task & Expected Result**	
1		Open the login screen.	
2		Verify that the login screen is displayed on client side.	**Pass / Fail**
3		Enter Email and Password.	
4		Verify that the Email can be entered.	**Pass / Fail**
5		Verify that the password is masked and can be entered.	**Pass / Fail**
6		Verify that submit button is displayed.	**Pass / Fail**

6.1.2 Sign Up

Table 10 Test Case 2: Sign Up

Test Case ID	TC_FUNCT_02
Description	Tests the Registration process.

Applicable for		Disease identifier system.	
Requirements		User did not have registered already.	
Initial Conditions		Nil	
Step	**Full / Regr**	**Task & Expected Result**	
1		Open the Registration screen.	
2		Verify that the Signup screen will show to client.	**Pass / Fail**
3		Enter First name, last name, email and password and etc.	**Pass / Fail**
4		Verify that all the data in the fields can be entered.	**Pass / Fail**
5		Verify that the password is masked and can be entered.	**Pass / Fail**
6		Verify that submit button is displayed.	**Pass / Fail**

6.1.3 Logout

Table 11 Test Case 3: Logout

Test Case ID	TC_FUNCT_3
Description	Logout form the application.
Applicable for	Successfully logout user from the application.
Requirements	User has logged in.
Initial Conditions	Member record is present in the database from where user logged in.

Step	**Full / Regr**	**Task & Expected Result**	
1		Click the logout button.	
2		Verify from the dialog box that you really want to logout.	**Pass / Fail**
3		User wills successfully logout by saving all his credentials.	**Pass / Fail**
4		User is redirected to the login screen.	**Pass / Fail**

6.1.4 User’s Information

Table 12 Test Case 4: User’s Information

Test Case ID	TC_FUNCT_4
Description	Enter user’s information.
Applicable for	Entering user’s information such as contact number to the local database.

Requirements	User has logged in. User has privileges. User has filled all the fields that are given on page		
Initial Conditions	User must be logged in and have privileges.		
Step	**Full / Regr**	**Task & Expected Result**	
1		Click the add contact button.	
2		Verify the contact number.	**Pass / Fail**
3		Verify that the data field entries working.	**Pass / Fail**
4		Verify that the contact is saved on local database.	**Pass / Fail**
5		Verify that it will take back to main menu.	**Pass / Fail**

6.1.5 GPS LOCATION

Table 13 Test Case 5: GPS Location

Test Case ID	TC_FUNCT_05
Description	Get the location.
Applicable for	All the users.
Requirements	User has logged in. User has administrator privileges.
Initial Conditions	User has logged in.

Step	**Full / Regr**	**Task & Expected Result**	
1		Click the Announced posts icon.	
2		Verify that the Google map screen is displayed.	**Pass / Fail**
3		Verify that the dates can be entered in Edit Text.	**Pass / Fail**
4		Verify that Go button is working.	**Pass / Fail**
5		Verify that the location is displayed according to the area.	**Pass / Fail**
6		Verify that the Back button is working.	**Pass / Fail**

6.1.6 Search Disease

Table 14 Test Case 6: Search Disease

Test Case ID	TC_FUNCT_06
Description	Search disease
Applicable for	Search disease and contact as soon as possible.
Requirements	User has logged in. User has administrator privileges.
Initial Conditions	Users contact number must be in local database.

Step	Full / Regr	Task & Expected Result	
1		Click the Search disease button.	
2		Click the panic button	**Pass / Fail**
3		Verify that the disease has been searched.	**Pass / Fail**
4		Verify that the detail about the disease is ready to send to persons in contact.	**Pass / Fail**
5		Send sms to all the contacts and some required contacts.	**Pass / Fail**

Chapter 7
Conclusion

7.1 Conclusion

This is a mobile based application that would cater as a medical Aid and informative application with an emergency button using location API. This Application will work as a medical guide for every person. It will help users to share symptoms and diseases with other peoples. We have tried to make a very user friendly system and we hope it will provide great experience to every user. This system has given us deep insight into this area and we strongly feel that our work is just like a drop in an ocean. There is a lot to be done in this area especially efficient data searching etc.

7.2 Future Enhancements

The future enhancements that can be done by making more interactive user interface. In future there would be some additional changes like automatic response system. System will be able to response automatic. All the information about the user's disease will send automatically to all the users.

References

[1]: ECLIPSE
Date of page: 4/08/2015
URL: https://eclipse.org/home/index.php

[2]: Java SE Downloads
Date of page: 7/08/2015
URL: http://www.oracle.com/technetwork/java/javase/downloads/index.html

[3] Chronic Disease Management (formerly Enhanced Primary Care or EPC) — GP services
Date of page: 12/08/2015
URL:
http://www.health.gov.au/internet/main/publishing.nsf/content/mbsprimarycare-chronicdiseasemanagement

[4]: Chronic Disease Management Patient Information
Date of page: 20/08/2015
URL:
http://www.health.gov.au/internet/main/publishing.nsf/content/mbsprimarycare-chronicdisease-pdf-infosheet

[5]: Chronic Disease Management Plan
Date of page: 28/08/2015
URL:
http://www.humanservices.gov.au/customer/services/medicare/chronic-disease-management-plan

[6]: Medisave for Chronic Disease Management Programme (CDMP) and vaccinations
Date of page: 10/09/2015
URL:
https://www.moh.gov.sg/content/moh_web/home/policies-and-issues/elderly_healthcare.html

[7]: Chronic disease management programme in people with severe knee osteoarthritis: efficacy and moderators of response.
Date of page: 12/09/2015
URL: http://www.ncbi.nlm.nih.gov/pubmed/18212037

[8]: 7 mobile apps for chronic condition management
Date of page: 18/09/2015
URL:
http://www.fiercemobilehealthcare.com/slideshows/7-mobile-apps-chronic-condition-management

Appendix

User Manual

Splash Screen

The splash Screen Logo of the Application it's normally appears on the start of the application.

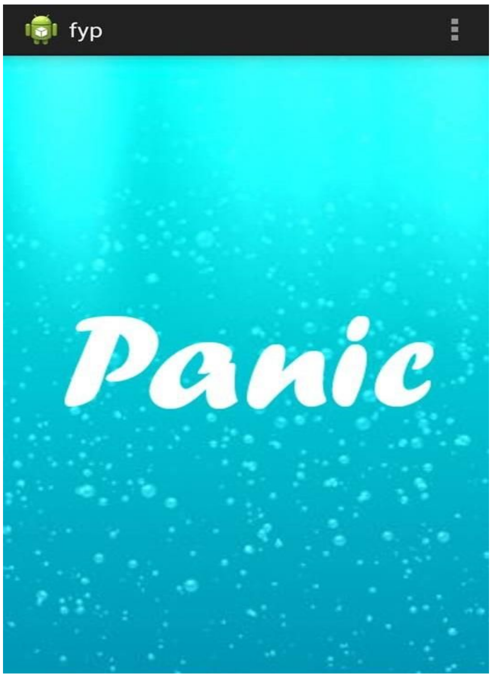


Figure 12Splash Screen

Login and Signup

This is the login screen where the User enters the registered id and password for "Signup" and if the user is not registered then go to "Signup" and create user account.

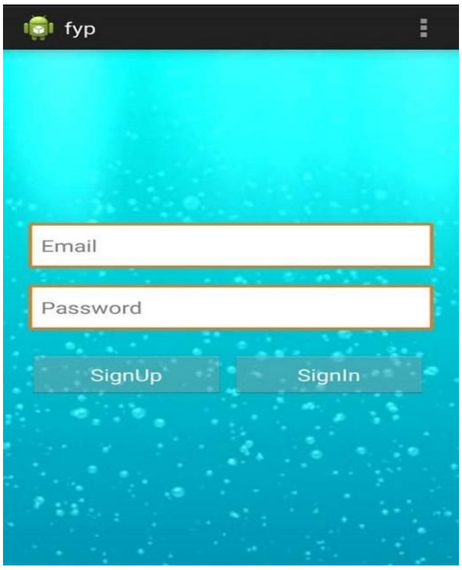


Figure 13 Login and Signup

Sign Up

The user can registered him through this screen.

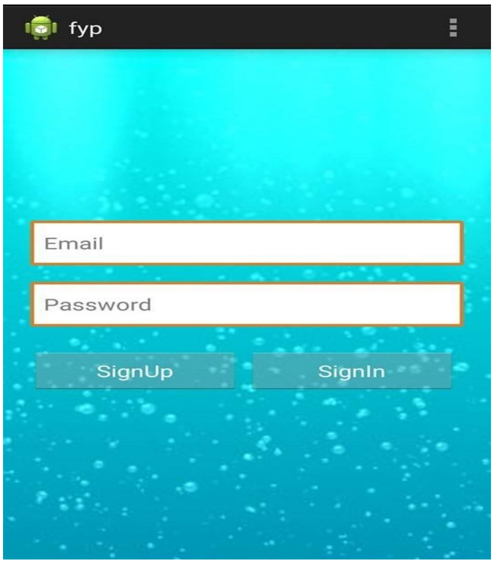


Figure 14 Login and Signup

Main Form

This is the main primarily used for navigation for the user.

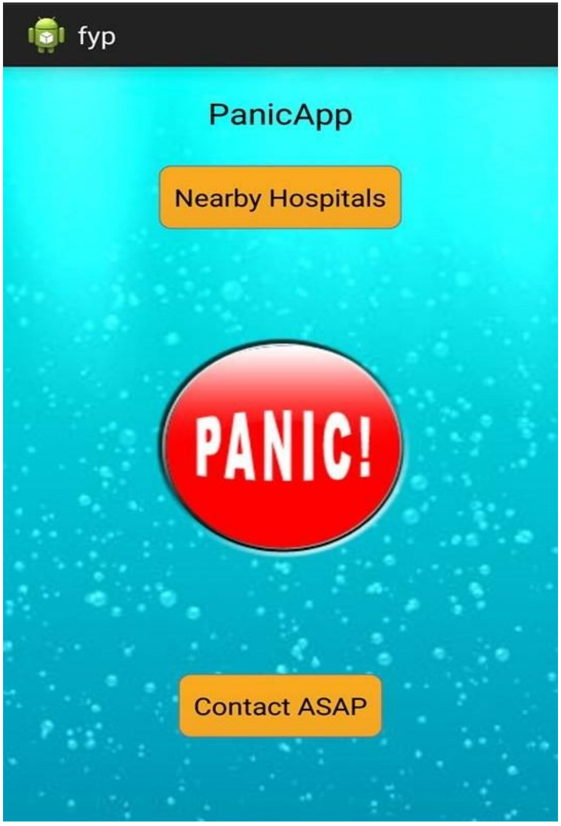


Figure 15 Main Form

Panic Message

The user can enter the place or select the location through Google Map and send the panic message.

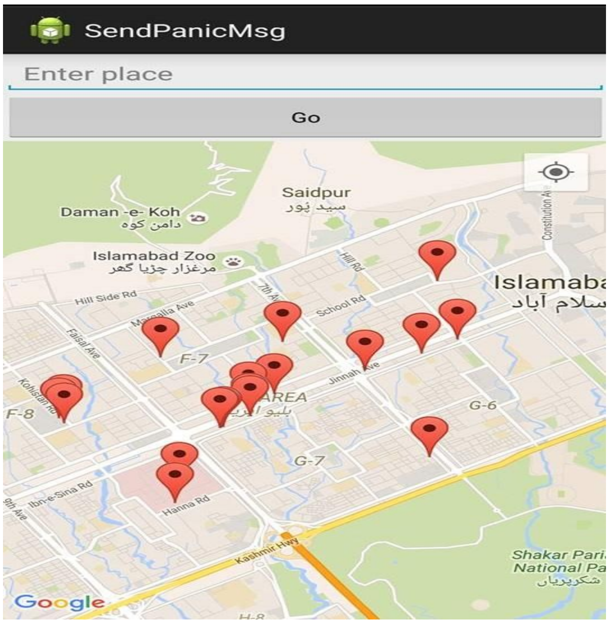


Figure 16 Panic Message

Menu Screen

In this screen the user can add family diseases view family diseases etc.

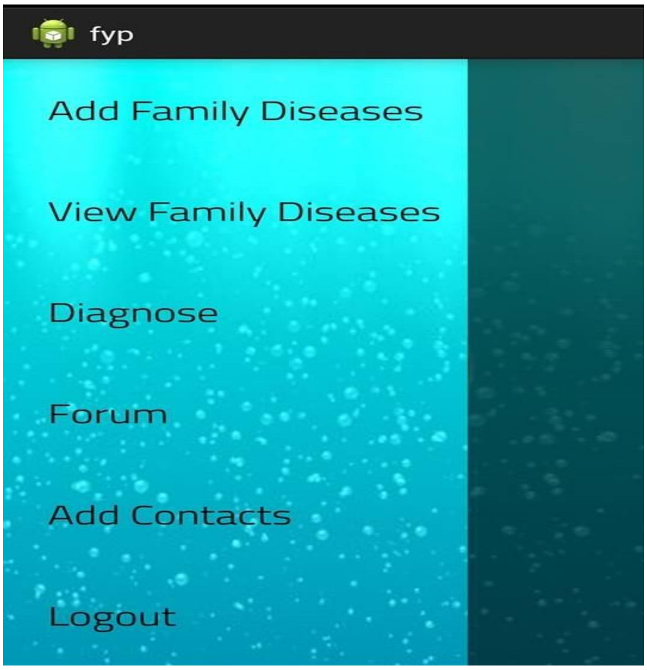
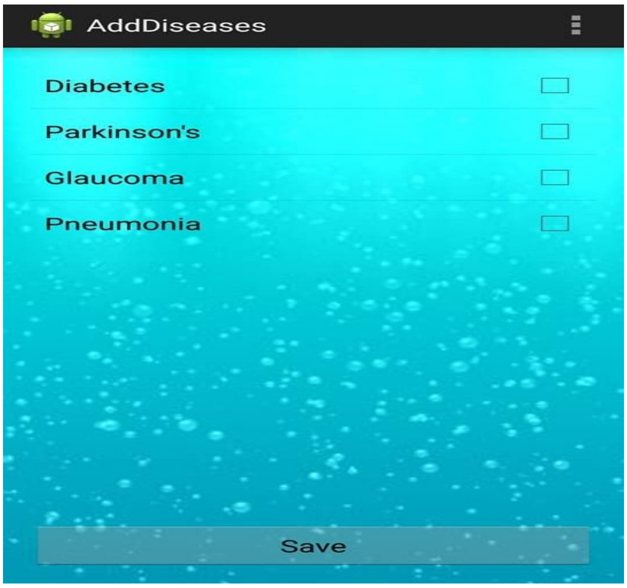


Figure 17 Menu Screen

Add diseases

The user can add their family diseases into database at this screen.



Diabetes □

Parkinson's □

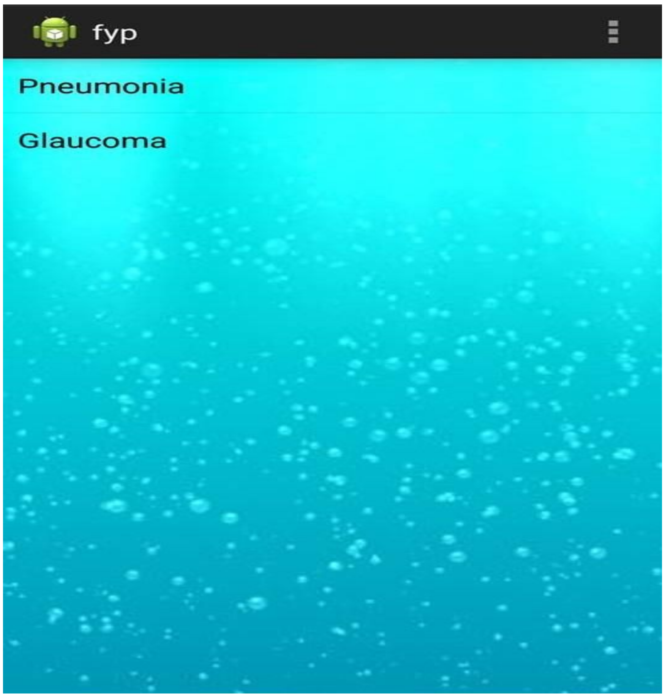
Glaucoma □

Pneumonia □

Figure 18 Add Diseases

View Family Diseases

The user can view their family Diseases at this screen.



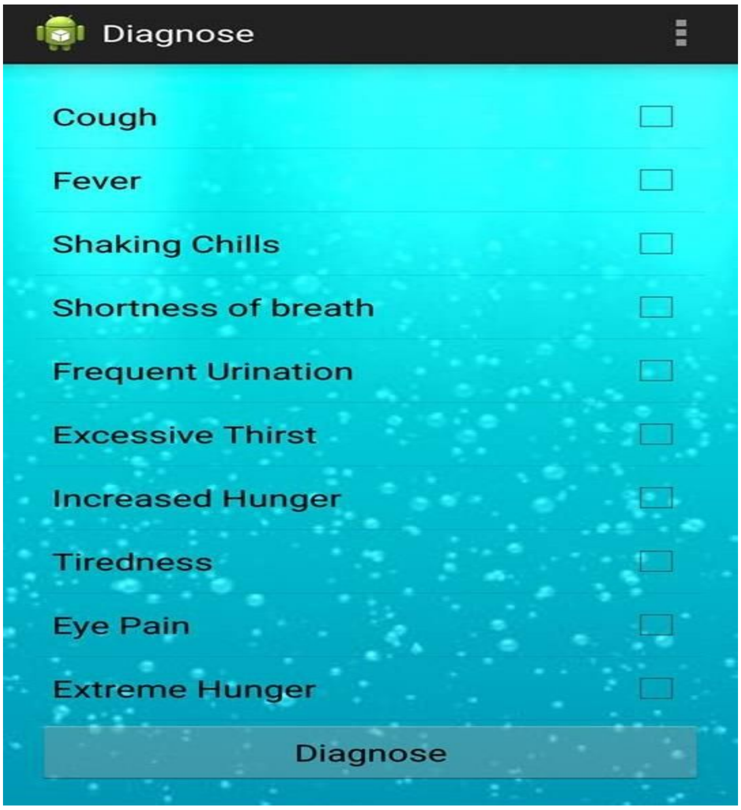
Pneumonia

Glaucoma

Figure 19View Family Diseases

Diagnose

At this screen the user can gives the symptoms of the disease the app will find out what disease the user might have.



Cough □

Fever □

Shaking Chills □

Shortness of breath □

Frequent Urination □

Excessive Thirst □

Increased Hunger □

Tiredness □

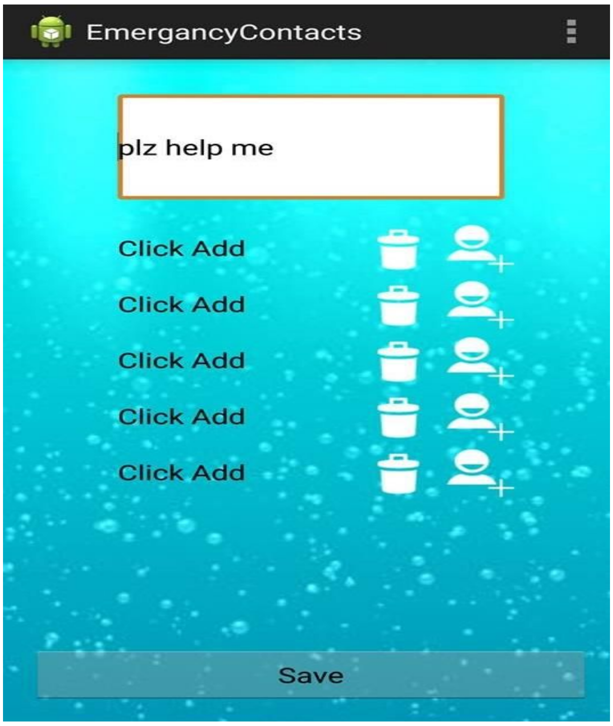
Eye Pain □

Extreme Hunger □

Figure 20 Diagnose

Add Contacts

At this screen the user select the contact and send the sms.



EmergancyContacts

plz help me

Click Add

Click Add

Click Add

Click Add

Click Add

Figure 21 Add Contacts