AN ANDROID APP FOR HEALTH MANAGEMENT SYSTEM

SUBMITTED BY

MD.DELWAR HOSSAIN
ID: 131-15-2352

FAHMIDA KHANAM
ID: 131-15-2363

SANJIDA JAVEEN
ID: 131-15-2318

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering.

Supervised By

Nazmun Nessa Moon
Assistant Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2016

APPROVAL

This project titled **"An Android App for Health Management System"**, submitted by Md.Delwar Hossain, Fahmida Khanam, Sanjida Javeen to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on Monday 26 December 2016.

BOARD OF EXAMINERS

____

Dr. Syed Akhter Hossain **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Dr. Sheak Rashed Haider Noori **Internal Examiner**
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Narayan Ranjan Chakraborty **Internal Examiner**
Assistant Professor and Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Nazmun Nessa Moon **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Dr. Mohammad Shorif Uddin **External Examiner**
Professor and Chairman
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Nazmun Nessa Moon, Assistant Professor, Department of CSE,** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree and diploma.

Supervised by:

____

Nazmun Nessa Moon
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:

____________________	________________
Md. Delwar Hossain	**Fahmida Khanam**
ID: 131-15-2352	ID: 131-15-2363
Department of CSE	Department of CSE
Daffodil International University	Daffodil International University

____

Sanjida Javeen
ID: 131-15-2318
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty Allah for His divine blessing makes us possible to complete this project successfully.

We are grateful and wish our profound indebtedness to **Nazmun Nessa Moon**, Assistant Professor, Department of Computer Science and Engineering, Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “Android Application Development” influenced us to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Dr. Syed Akhter Hossain**,** Professor and Head, Department of CSE, CIS & CS Daffodil International University for his kind help and support to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of ours parents.

ABSTRACT

Due to revolution of Internet technology worldwide, the rapid use of mobile application regarding various subjects increasing day by day. A good mobile application on Health Care is user friendly to save the information of family members of current health condition. This project “An Android Apps for Health Management System” is based on this concept of development of a mobile application for health management. By using this application, it will be very easy to maintain healthcare. Its functionality is very easy and anyone can use it to manage family and personal health. Its functionality is designed according to the basic demands of user. It provides much functionality among them. In Diet Chart management, Vaccination management, Doctor Management, Medical History management and etc. The intended project “Design and Development of an Android Application for Advanced Health Management System” is targeted to facilitate users from home and abroad by giving information using mobile.

TABLE OF CONTENTS

CONTENTS **PAGE**

CHAPTER

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

In recent number of year’s instant use of communication tools like mobile devices have been successfully introduced to support communication and collaboration processes in work environments. Research suggests that the use of mobile applications also increasing rapidly day by day due to social presence and awareness within a collaborative group. Now a day’s all over the world people are more interested in various type of mobile application, and dynamical informative apps are one of the top most choice for its very significant matter. This kind of apps helps to learn easily and patiently. Android devices come in all kinds of sizes, with all sorts of features, and all sorts of prices. Each version of Android is named by dessert, and the most recent version of Android is Lollipop with Android in control of mobile experience. That’s why now a day’s android based application development is also one of the top most choice of the developers in many areas. Here we use android to develop an android health management app to take care of family health condition.

1.2 Objective with specific aims

Its functionality is designed according to the basic demands of user. It provides much functionality among them. There are Diet Chart management, Vaccination management, Doctor Management, Medical History management and etc. The project describes “Design and Development of an Android Application for Personal Health Management System” is an enhanced instant usable application tool which user can use through the help of android base mobile devices. By using this application every user can use it. It will be very easy to maintain family and personal healthcare and anyone.

1.3 Users of this application

The user of this application is everyone who has an android smart device. In this app the design and functionality is very user friendly. So anyone can use this application and record the family information for taking necessary step when they want.

1.4 About our app

“Health Management system” is an application which will help everyone/user to get information about family members. If user selects any specific patient listed in this app, he/she gets all information of that patient from the app.

This app has the information, Image, doctors list, diet list, emergency call, emergency note etc. By touching your android device you can get to know about your current state and calculate estimated time to reach your selected places.

Android is designed to run on many different types of devices, from phones to tablets and televisions. As a developer, the range of devices provides a huge potential audience for our app. This app made to be successful on all these devices; it should tolerate some feature variability and provide a flexible user interface that adapts to different screen configurations.

Our app's user interface is everything that the user can see and interact with. Android provides a variety of pre-build UI components such as structured layout objects and UI controls that allow us to build the graphical user interface for our app. Android also provides other UI modules for special interfaces such as menus.

1.5 Adaptability

Google android provides an application distribution platform named play store which is the android application market place and developers can post there application in here with following some rules and regulation. Also developers required to pay a small amount of fees annually. The user can download application from this play store mostly free of charge but some application is required to pay a small amount of fees. Whenever developers update of their application with new version, user will be informed immediately and most update version is guaranteed.

1.6 Conclusion

The remainder of the project paper is organized as follow. In this chapter, firstly we like to outline a typical use case scenario and application environment of the “Health Management System” prototype and the following section describes the theoretical importance of connectedness, social presence and awareness of smart phone supported communication and collaboration. We then presented a brief technical description of the prototype application development phases that brief overview of Android Development. The project paper concludes with a discussion of potentials, limitations and future upgrade prospects of the underlying concept.

CHAPTER 2

LITERATURE REVIEW

2.1 Basic Android Overview

Android is a comprehensive open source platform designed for mobile devices. It is championed by Google and Open Handset Alliance. The goal of the alliance is to "accelerate innovation in offer consumers a richer, less expensive, and better mobile experience." Android is the vehicle to do so. As such, Android is revolutionizing the mobile space. For the first time, it is a truly open platform that separates the hardware from the software that runs on it [2]. This allows for a much larger number of devices to run the same applications and creates a much richer ecosystem for developers and consumers. Fig 2.1 shows the relation between companies of open handset alliance.

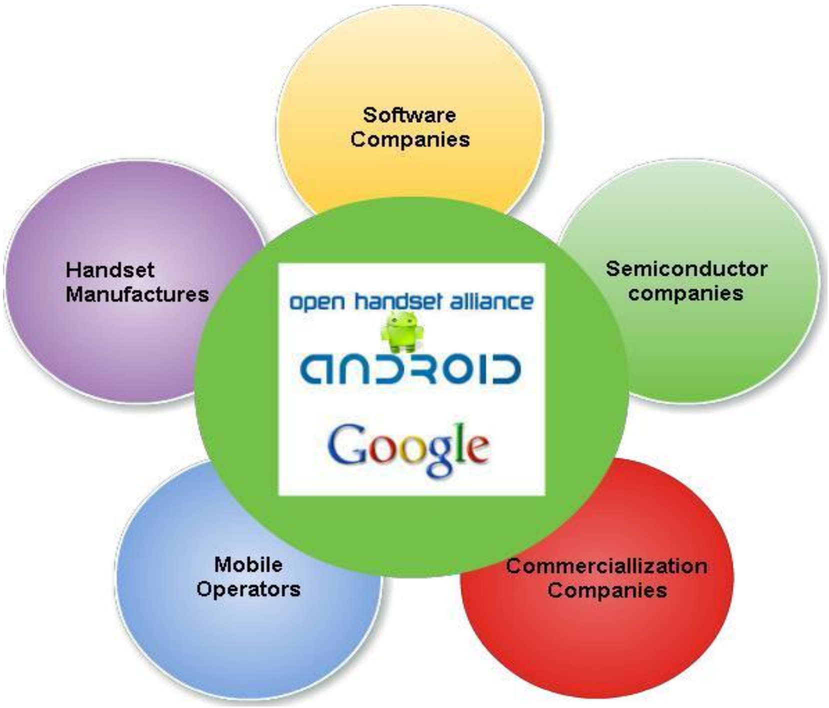


Fig 2.1: Open Handset Alliance (OHA) [2].

2.2 Android Versions

Since April 2009, Android versions have been developed under a confectionery-themed code name and released in alphabetical order; the exceptions are versions 1.0 and 1.1 as they were not released under specific code names:

- Alpha (1.0)
- Beta (1.1)
- Cupcake (1.5)
- Donut (1.6)
- Eclair (2.0–2.1)
- Froyo (2.2–2.2.3)

Gingerbread (2.3–2.3.7)
Honeycomb (3.0–3.2.6)
Ice Cream Sandwich (4.0–4.0.4)
Jelly Bean (4.1–4.3.1)
KitKat (4.4–4.4.4)
Lollipop (5.0–5.0.2)

Fig 2.2: Android Versions [3].

Fig 2.2 is the Android versions with codenames. The Android version number itself partly tells the story of the software platform's major and minor releases. What is most important is the AIP level. Version numbers change all the time, sometimes because the AIPs have changed, and other times because of minor bug fixes or performance improvements [3].

2.3 The Android Stack

The Android operating system is like a cake consisting of various layers. Each layer has its own characteristics and purpose. The layers are not cleanly separated but often seep into each other on. Fig 2.3 shows the part of the Android stack.

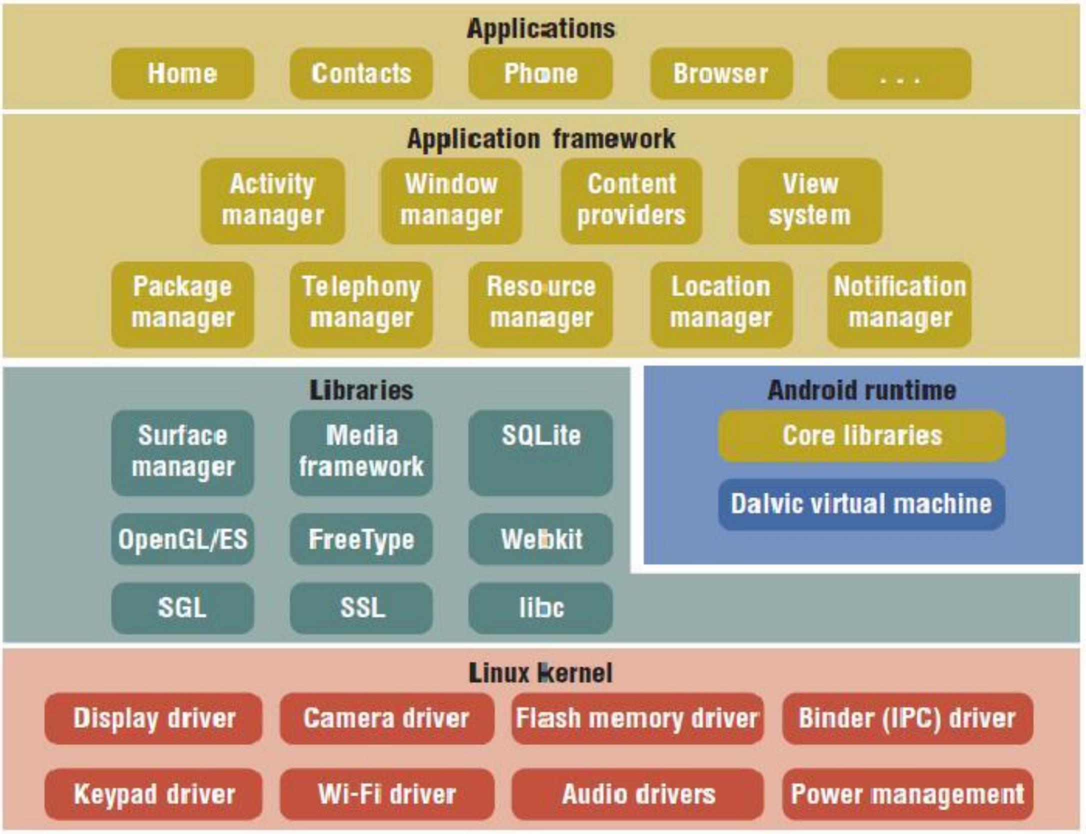


Fig 2.3: Android Stack [1].

2.4 Android and Java

In Java, we write our java source file, compile it into a java byte code using the Java compiler, and then run this byte code on the Java VM. In Android, things are different. We still write the Java source file, and we still compile it to Java byte code using the same Java compiler. But at that point, we recompile it once again using the Dalvik compiler to Dalvik byte code [3]. It is this Dalvik byte code that is than executed on the Dalvik VM.

Fig 2.4 illustrates this comparison between standard Java (on the list) in Android using Dalvik (on the right).

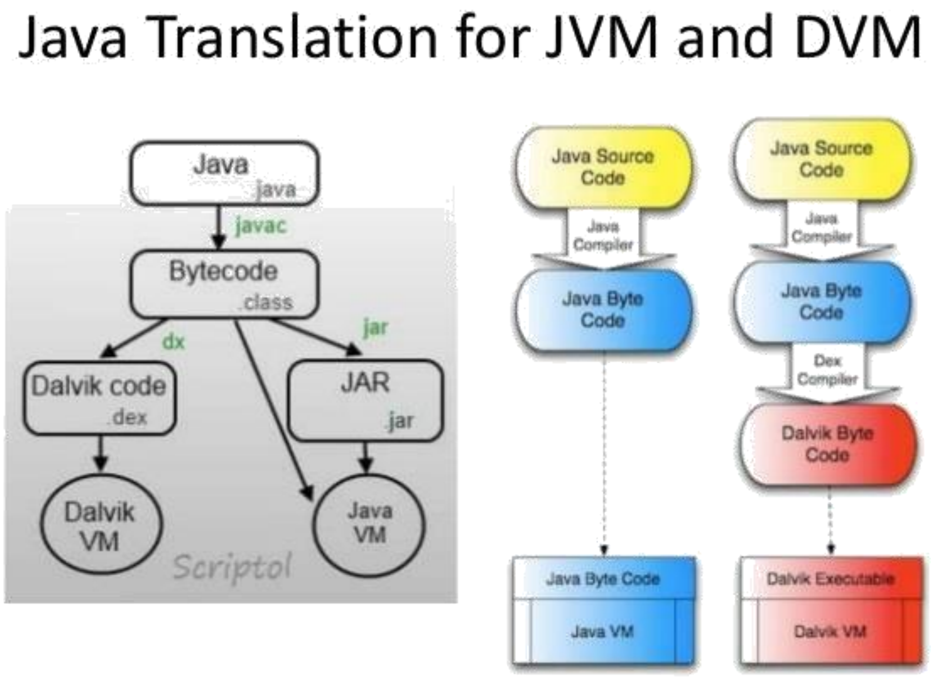


Fig 2.4: Java versus Dalvik [3].

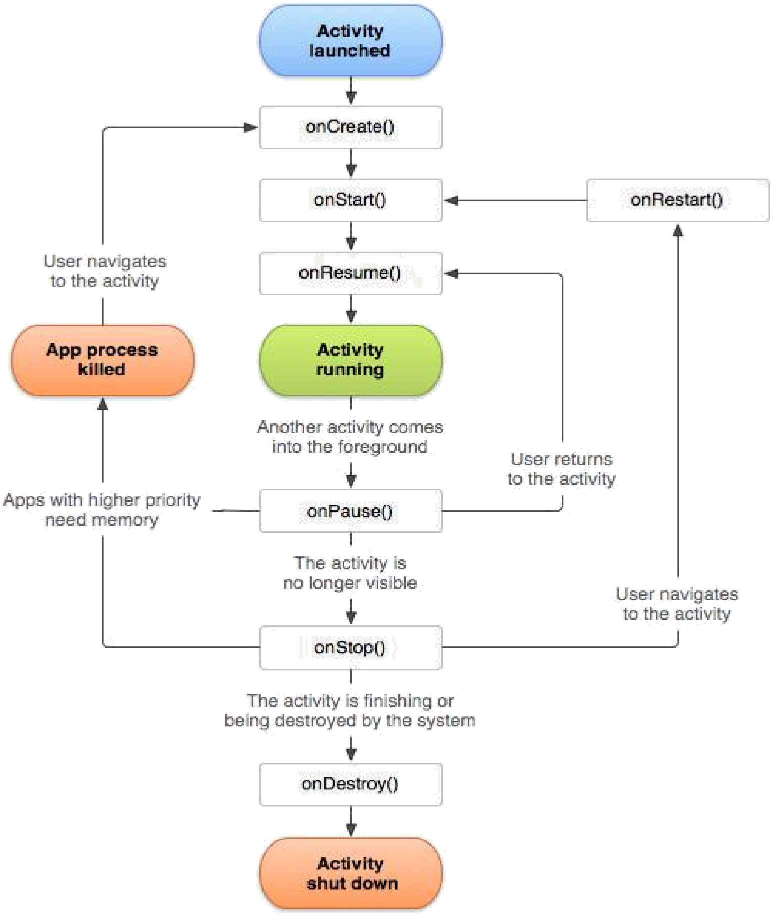
A side effect of this is that in theory we could write Android applications in any other language that compiles down to Java byte code.

2.5 Activities

An activity is usually a single screen that the user sees on the device at one time. An application typically has multiple activities, and the user flips back and forth among them. As such, activities are the most visible part of our application. We usually use an activity [4]. Just like a website has a “home page,” an android app has a main (full screen) activity, usually the one that is shown first when we launce the application. And just like a website has to provide some sort of navigation among various pages, an Android app should do to same. On the web, we can jump from a page on one website to a page on another. Similarly, in Android, we could be looking at an activity of one application, but shortly after we could start another activity in a completely separate application.

2.6 Activity Life Cycle

Launching an activity can be quite expensive. It may involve creating a new Linux process, allocating memory for all the UI objects, inflating all the objects from XML layouts, and setting up the whole screen. Since we are doing a lot of work to launch an activity, it would be a west to just toss it out once the user leaves that screen. To avoid this waste, the activity life cycle is managed via Activity Manager. Activity Manager is responsible for creating, destroying, and managing activities. For example, when the user starts an application for the first time, the Activity manager will create its activity and put it onto the screen. Later, when the user switches screens, the Activity Manager will move that previous activity to a holding place [4]. This way, if the user wants to go back to a previous activity, it can be started more quickly. Previous activities that the user hasn't used in a while will be destroyed in order to free more space for the currently active one. This mechanism is designed to help improve the speed of the user interface.


Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

` Fig 2.6: Activity Life Cycle [7].

Improve the overall user experience. Programming for Android is conceptually different than programming for some other environments. Fig 2.6 shows the states that activity [7] can go through.

a) Starting state

When an activity doesn't exist in memory, it is in a starting state. It is while starting up, the activity will go through a whole set of callback methods that we as a developer have an opportunity to fill out. Eventually, the activity will be in a running state.

b) Running state

The activity in a running state is the one that is currently on the screen and interacting with the user. We also say this activity is in focus, meaning that all user interactions such as touching the screen and clicking buttons are handled by this one activity. As such, there is only one running activity at any given time.

c) Paused state

When an activity is not in focus (i.e. not interacting with the user) but still visible on the screen, we say it's in a paused state. This is not a typical scenario, because the device's screen is usually small, and an activity is either taking up the whole screen or none all. We often see this case with dialog boxes that comes up in front of an activity, causing it to become paused. All activities go through a paused state en route to being stopped.

d) Stopped state

When an activity is not visible, but still in memory, we say it's in a stopped state. Stooped activity could be brought back to the front to become a running activity again. Or, it could be destroyed and removed from memory. The system keeps activities around in a stopped state because it is likely that the user will still want to get back to those activities sometime soon.

e) Destroyed state

A destroyed activity is no longer in memory. The Activity Manager decided that this activity is no longer needed and has removed it. Before the activity is destroyed, it can perform certain actions, such as save any unsaved information.

2.7 Intents

Intents are messages that are sent among the major building blocks [5]. They trigger an activity to start up, tell a service to start or stop, or are simply broadcasts. Intent could be explicit or implicit. In an explicit intent, the sender clearly spells out which specific component should be on the receiving end. In an implicit intent, the sender specifies the type of receiver. Fig 2.7 shows how Intent work.

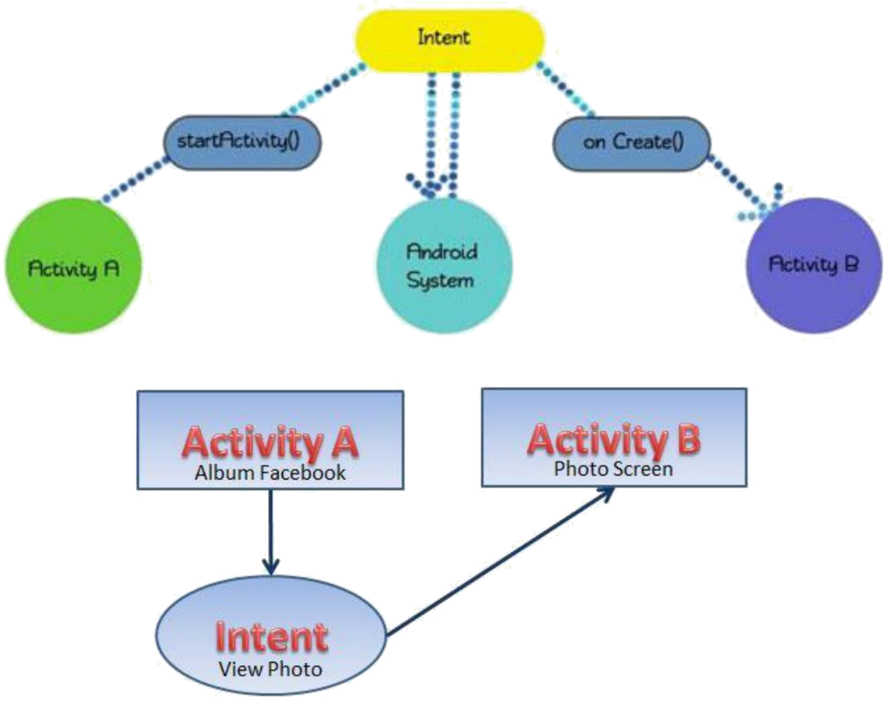


Fig 2.7: Intents.

2.8 Services

Services run in the background and don’t have any user interface components. They can perform the same actions as activities, but without any user interface. Services are useful for actions that we want to perform for a while, regardless of what is on the screen. For example, it may music player to play music even as you are flipping between other applications. Services have a much simpler life cycle than activities. Either start a service or stop it. Also, the service life cycle is more or less controlled by the developer, and not so much by the system.

Consequently, we as developers have to be mindful to run our services so that they don't consume shared resources unnecessarily, such as the CPU and battery. Fig 2.8 shows services life cycle and how it works.

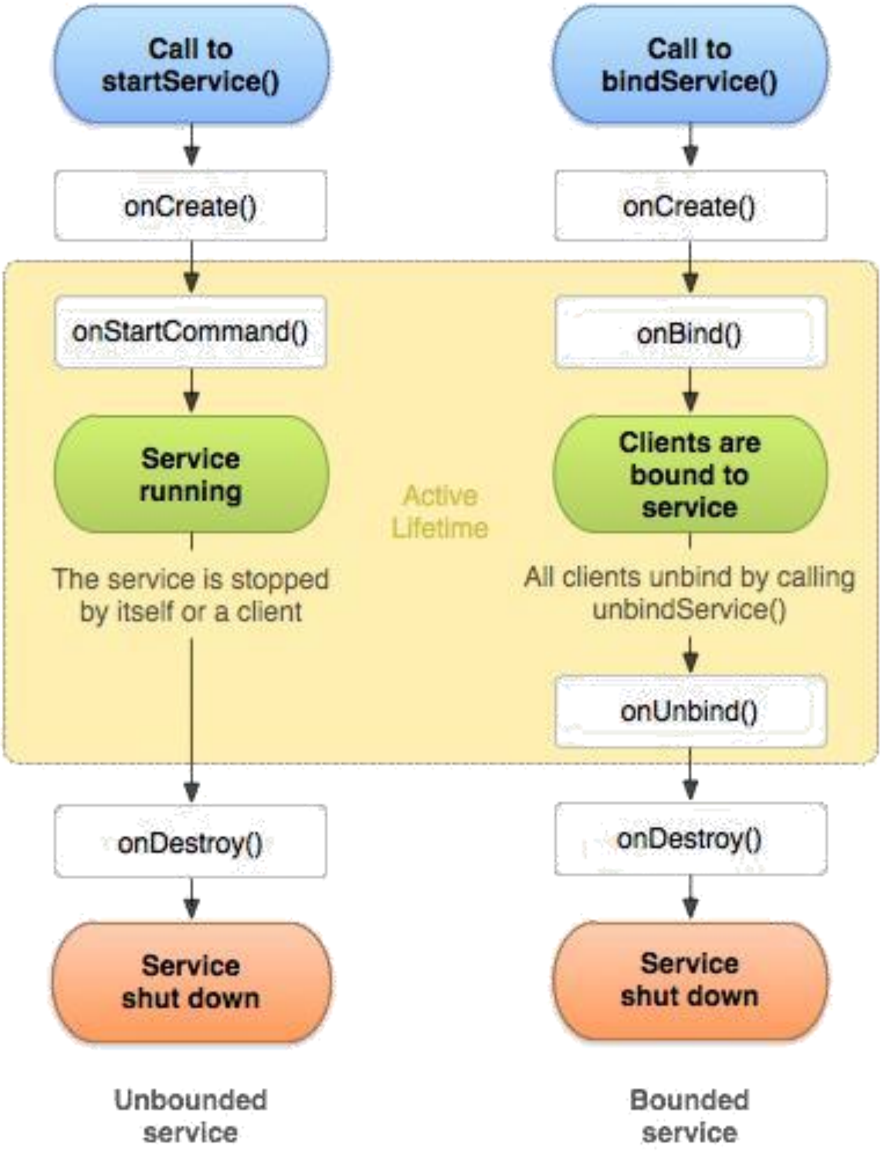


Fig 2.8: Services Life Cycle.

CHAPTER 3

SYSTEM DESIGN & ANALYSIS

3.1 Project Scenario

Due to revolution of internet technology worldwide, the rapid use of mobile application on regarding various subjects increasing day by day. A good mobile application on Health Care is user friendly to save the information of family members of current health condition. This project "**An Android App for Health Management System"** is based on this concept of development of a mobile application on caring health by mobile apps. By using this application, it will be very easy to maintain healthcare. Its functionality is very easy and anyone can use it to manage family and personal health. Its functionality is designed according to the basic demands of user. It provides much functionality among them. There are Diet Chart management, Vaccination management, Doctor Management, Medical History management and etc. The intended project "Design and Development of an Android Application for Health Management System" is targeted to facilitate users from home and abroad by giving information using mobile.

3.2 Requirement Analysis and Proposed Architecture

- As a user, we want to create icare profile, so that we can store data.
- To create multiple icare profile so that we can maintain that data of my family.
- To see profile list for further process.
- To get general health information/ vaccination information/ growth information/ diet and nutrition information etc, so that we can maintain mine and my family's health.
- To preview icare profile, so that we can see overall health condition.
- To edit profile information, so that it stays up-to-date.
- To remove profile data with and without all relevant content, if we don't need that.
- To prepare daily diet chart of each profile to maintain proper diet.
- To set reminder for diet chart so that we don't forget.
- To add weekly diet chart for each profile.
- To create important date list so that we can view them later.
- To create vaccination charts of child.

- To set reminder on vaccination date so that we don’t forget.
- To see remaining vaccination.
- To see given vaccination detail to know more about them.
- To know vaccination detail (reason for vaccination).
- To set doctors profiles so that we can differ them in time of need.
- To set doctor’s appointment information so that we don’t forget.
- To upload prescription so that doctors know about mu current or previous medication.
- To know the care center information.
- To the care center location using gps # Google Map so that we can find them.
- To save profile data into web server # server communication so that we don’t lose it.
- To capture photo so that we can send it to the doctor.
- To edit photo (crop, rotate, free hand drawing).
- To save edited photo into phone memory for future use.
- To see added photo gallery in time of need.
- To preview image with zoom operation, rotate operation, and scaling operation.
- To remove photo from gallery, if don’t need them anymore.
- To set an emergency number so that we can call on emergency time using sensor.
- To call/sms and send email with attachment to doctor.
- To see profiles medical history for further medication.
- To add important note.
- To get route on map from current location to doctor’s address.

3.3 Health Management System prototypes

The application aim is to provide detailed texts, picture and other guidance information of health so people can better understand of patient to make decision. Most of the time it’s seen that we don’t able to get health information timely when we are far away from home. Therefore, we intend to explore how to build a mobile apps that I saved all the information of my family members to get the current health condition.

Modules in this application

a) Creating patient profile.

b) Taking all the legal information of patient.

c) Add the doctors and diet routine of patient.

d) Add the Emergency Note and call for specific location.

e) Set the alarm of diet.

f) Knowing all the information of vaccination.

3.4 Use-Case model

A use case is an objective user's wants to achieve with a system. Use cases are named with verb or verb + noun phrase. The model in Fig 3.1 is usually short yet description enough to describe a user objective. User performs use case to yield observable goal. Take online hotel reservation system as an example. "Make reservation" is a use case as this is what user wants to achieve with the system. The function of looking up a hotel on an online map can also be what a user needs.

It is intended to provide an overview of what the user wants without knowing how to achieve the goal. In order to identify how to achieve a goal, we can also document its scenario and steps (i.e. interaction) involved between user and system, with main flow, exception flow, conditional flow, etc.

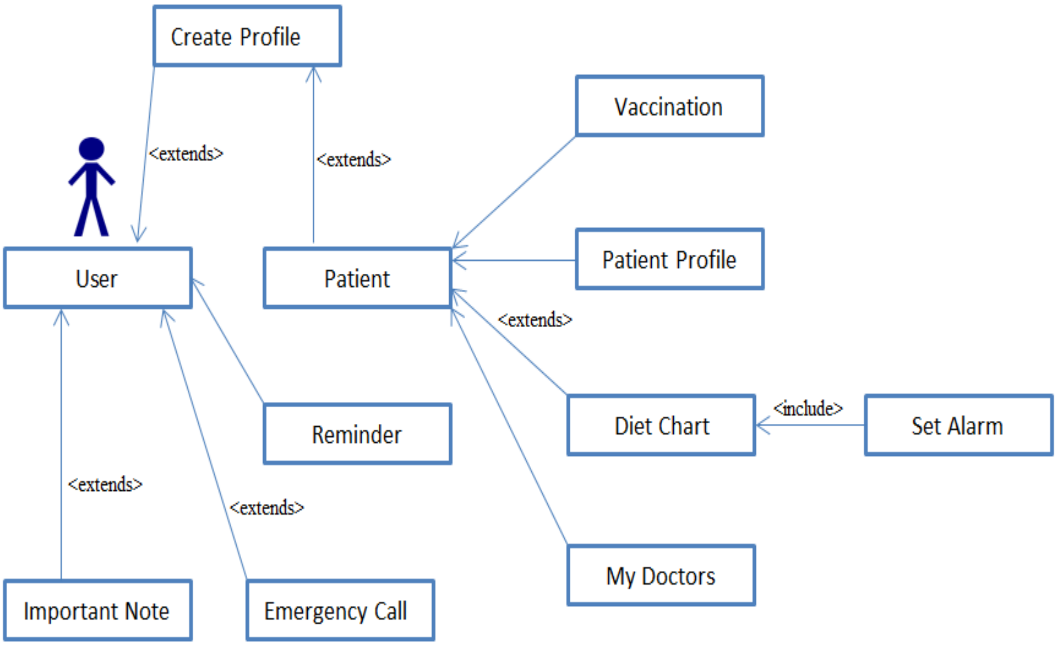


Fig 3.1: Use-Case Model.

3.5 Description of Use-Case Model:

In this use-case model human actor is patient/user.

User Use Case Details

Use case name : Create Profile

Precondition : Open Application

Actor : User

Primary Path : 1. Enter name
2. Enter Phone
3. Select profile type
4. Select current date
5. Explain Current Health Condition

Exceptional path : 1.1 No Profile Created.
2.1 Enter all the information
3.1 Enter valid patient type

Use case name : Important Note

Precondition : No Precondition

Actor : User

Primary Path : 1. Enter Note Title
2. Enter specific date
3. Click Save Button

Exceptional path : 3.1 Please Enter a Title
4.1 Must add issue date

Use case name	:	Emergency Call
Precondition	:	No Precondition
Actor	:	User
Primary Path	:	1. Enter Call Counter Name
	:	2. Enter Phone Number
Exceptional path	:	1.1 Invalid Call Settings

Use case name	:	All Service/Patient Profile
Precondition	:	Create Profile
Actor	:	User
Primary Path	:	1. View Patient Profile
		2. Add Diet Item
		3. Add Doctor Profile
		4. View Vaccination Info
Exceptional path	:	2.1 Must Add Patient information
		3.1 Check the patient type

3.6 Key features of the App

a) Simplicity and user friendly for the user

In this app, we have tried to design and develop a convenient user interface, whatever the functionality it will deliver simplicity and what is better for the users. All the content can be accessed in the simplest way possible. We let user perform every action with case, and they will not lose them. That is the things that keeps mobile apps going and makes them popular and user friendly.

b) Android vs. IOS

These are two platforms to be essentially encompassed. But we developed this prototype in Android platform which is open source and extremely popular and used worldwide. What’s worth mentioning, the former platform might cause troubles with its number of devices, and hence the app should be specifically designed and thoroughly tested on each device you choose.

C) Security

This issue is vital to many apps. It is one of the first topics of discussion between user and software developers. But in our project don’t content anything that related with security matter.

3.7 Steps of Creating an Android Projects

Step 01: Create Project

After lunching the application, a splash screen will be shown containing our logo, and all the platforms we supported. Then in android studio click on File->New->Android Project. You will get a dialog window asking you about your new Android project. Explanation of all the significant fields:

Step 02: Project name

The project’s specific name is most important for any project. It is a good idea not to use any spaces in our project name. This makes it easier to access from the command line later. We entered “Health Management System” here.

Step 03: Contents

Leave this as is set to creating a new project since that’s what we intend to do.

Step 04: Build Target

This field indicates the type of Android system we intend to run this application on. This could be any Android platform, either standard or proprietary. I assume we’re working with android 4.2 (API level 17) and thus will choose the Android 4.2 option.

Step 05: Application name

Application name is a name which is visible to our mobile screen. It can be any text. For our app, we gave “iCare” whose title is “Health Management System”.

Step 06: Package name:

This field designates a java package, and as such it need to adhere to Java package naming conventions. In a nutshell, you want to use the reverse of your domain name for your package. We are going to use “Health Management System” here.

Step 07: Create Activity:

For the activity name, we must adhere to Java class naming conventions. Doing that simply means using upper Camel Case. We are going to give name of the Java Class as “Splash Screen”. After a few second we can see our home screen.

3.8 A Splash Screen UI

It is the first opening screen of our app which will show an opening animation for first 2 second full green screen then after 2 second a Red Circle will show up middle center of the green screen and at 3 second App name will show up at top center Fig. 3.8. We actually created the flag of Bangladesh in our splash screen. Then in the center of the Red Circle a Text "Click to Explore" is used as a button, by touching it user will proceed for the next step.

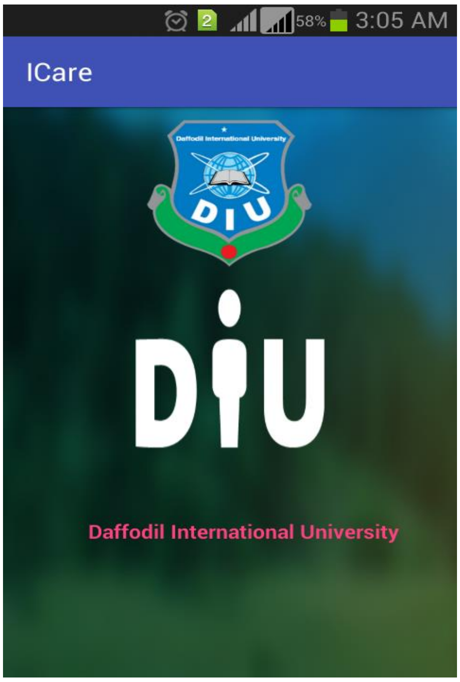

3:05 AM
58%
ICare
Daffodil International University
DIU
DiU
Daffodil International University

Fig 3.8: Splash Screen

CHAPTER 4

IMPLEMENTATION AND TESTING

In this chapter, we will discuss how to setup your environment for Android development. I'll go beyond just listing where we can download the software, and will cover some of the best practices in getting setup. I'll look at development operating system choices as well as the Android tools available.

4.1 Setting up a PATH to Tools

The Android SDK [2] has a folder that contains all its major tools. This will make it easier to access your tools without having to navigate to their specific location every single time. Details for setting up the PATH variable depend on the platform; see step 2 of the document "Installing Android SDK".

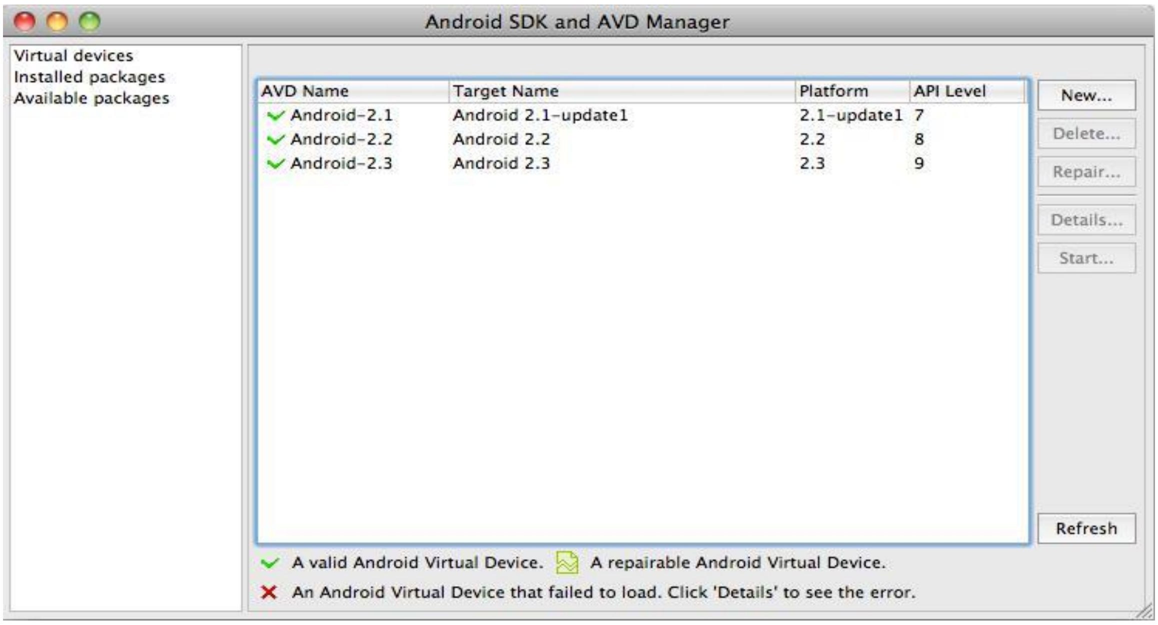
4.2 Installing Android Studio

Android Studio is an open source collection of programming tools originally created by IBM for Java. Nowadays, most developers in Java community favor Android Studio as their Integrated Development Environment (IDE) of choice. Android Studio has a lot of time-saving features, which I'll be pointing out as we continue. Keep in mind that, although powerful, Eclipse tends to be very resource hungry, and so you might want to restart it once a day if it starts running sluggishly. Although you can do Android development with any favorite text editor or integrated development environment (IDE), most developers seem to be using Eclipse, and thus that's what I use in this book.

Download Android recommend Android IDE for Java. User can install it in any directory.

4.3 Android Workspace

Android organizes all your work by projects. Projects are placed in a workspace; which is a location you choose. So, where you put your workspace is significant. I recommend ~/workspace as a simple place for your code. On Windows, however, I recommend storing your workspace in a directory that doesn't have spaces in it (they complicate anything you might do at the command line). C:\workspace is a good choice for Windows Users.



AVD Name	Target Name	Platform	API Level
✓ Android-2.1	Android 2.1-update1	2.1-update1	7
✓ Android-2.2	Android 2.2	2.2	8
✓ Android-2.3	Android 2.3	2.3	9

✓ A valid Android Virtual Device. A repairable Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.

Fig 4.3: Android SDK and AVD Manager.

4.4 Setting up Android Development Tools

We setup Android Tools [1] for Eclipse. The instructions are:

1. Start Eclipse, and then select Help->Install New Software.
2. In the Available Software dialog, click Add.
3. In the Add Site dialog that appears, enter a name for the remote site (for example, ''Android Plugging") in the "Name" field.
4. In the "Location" field, enter this URL: https://dl-ssl.google.com/android/eclipse/.
5. Click ok.
6. Back in the Available Software view, you should now see "Developer Tools" added to the list. Select the checkbox next to Developer Tools, which will automatically select the nested tools Android DDMS and Android Development Tools. Click Next.

7. In the resulting Install Details dialog, the Android DDMS and Android Development Tools features are listed. Click Next to read and accept the license agreement and install any dependencies then clicks Finish.
8. Restart Eclipse.

4.5 The Emulator

Running the application on a physical device versus an emulated device is pretty much the same thing. That is because the emulator is an actual code emulator, meaning it runs the same code base as the actual device, all the way down to the machine layer. A simulator and an emulator sound very similar, but are fundamentally different. To emulate means to imitate the machine executing the binary code. So, an emulator is sort of like a virtual machine. A simulator merely simulates the behavior of the code at a higher level. Android SDK [2] ships with a true emulator, based on QEMU. To use the emulator, we'll have to create an Android Virtual Device (AVD). The easiest way to do that is to start the android tool via Eclipse. To create a new AVD, start the tool called Android SDK [2] and AVD Manager Fig 4.1. You can start this tool from Eclipse by clicking on the icon or via the command line by starting the tools called android, which is located I your SDK/tools directory.

4.6 Project development and design tools

(A) Android [3.8.0v20120607] [1]

Eclipse is an open source community about software development. It is a very popular tool in the world. Also, there are many resources and tips on the internet so that it can enhance our work efficiency.

(B) Notepad

Notepad++ is another useful tool for us to deploy android, xml and java codes.

(C) Android SDK [v21.0.1-54035] [2]

To develop Android application, Android SDK is the essential tool. This SDK is very comprehensive tool that contains not only the library for development, but also includes the simulator to test the application. We use eclipse together to run and develop all the Android Application.

(D) Snipping Tools

(E) Adobe Photoshop CS6

(F) Adobe Illustrator CS6

(G) Cam studio utility

4.7 Hardware Implementation

When system is developed and tested it is ready for implementation and testing. This is the stage where system is going to be live. There should be an implementation and testing plan for this stage. When we were building this application we targeted a specific device so that the coding affect can easily be transferable or visible to the actual device also that can reflects our project concept. For the target device for testing purpose we choose Samsung Galaxy S Duos 2 Fig. 4.2 [4]. To support all the old and new android devices we selected minimum SDK Version as API level 8 for our project and Maximum SDK version as API level 21. Our device supports API level 17 and Operating System is v4.2.2 (Jelly Bean). Also it has all necessary A-GPS support, GPRSA, Wi-Fi 802.11b/g/n, WI-FI hotspot etc.

Fig 4.2: Target device of our application development [4].

4.8 Table Name: Patient Profile

The following table 1.1 represents the relation between patient profile and other services. Every patient has unique id to link other services like doctor profile, diet chart, vaccination, prescription etc.

Table 1.1: The relationship between patient profile and other services

Id	Pro.Type	Name	Image	Gender	BG	Date	Age	Height	Weight	Phone	Email	CDC
1	Own	Delwar	Png	M	A+	8.11.16	00	5.6’’	70kg	016	Abc	Good
2	Mother	Sahanara	Png	F	O+	8.11.16	00	5.6’’	50kg	017	Efg	Well
3	Father	Alam	Png	M	O+	8.11.16	00	5.6’’	75kg	017	Ijk	Good

Table Name: Doctor Profile

The following table 1.2 represents the relation between patient profile and doctors. Every patient has unique id to link with doctor.

Table 1.2: The relationship between doctor profile and other services

Id_Docto r	Id_Patient Doctor	Doctor Name	Doctor Number	Doctor Email	Doctor Address	Doctor About
1	1	Delwar	016	abc	Dhanmondi	FCPS
2	2	Sweety	016	abc	Mirpur	MBBS
3	3	Javeen	019	abc	Shukrabad	FRCS

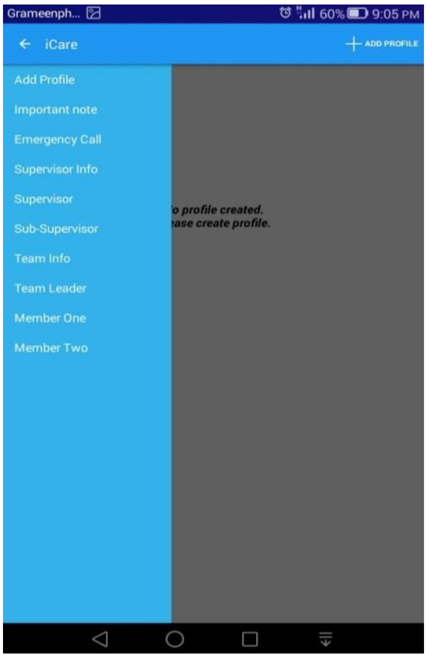
Table Name: Diet Profile

Table 1.3: The relationship between patient diet and other services

Id_Diet	Id_Patient_Diet	Day	Breakfast	Lunch	Dinner
1	1	Saturday	Bread	Rice	Tea
2	2	Monday	Bread	Rice	Tea
3	3	Friday	Bread	Rice	Tea

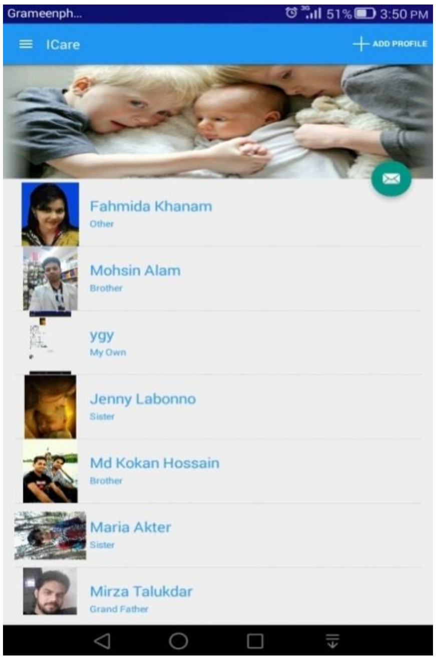
4.8.1 Home Screen

This is the home screen of our application. Here user can create there profile entering the details information of patient. Every patient have a unique id to identify there profile id and others information into database.



'o profile created.
'ase create profile.

Fig 4.8.2: Home Screen



Fahmida Khanam
Other

Mohsin Alam
Brother

ygy
My Own

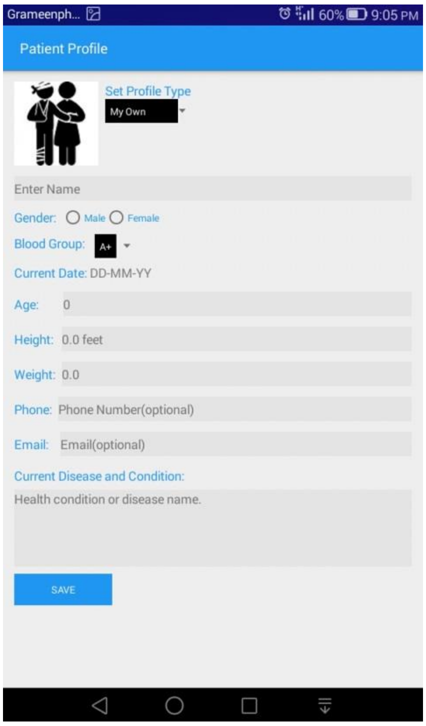
Jenny Labonno
Sister

Md Kokan Hossain
Brother

Maria Akter
Sister

Mirza Talukdar
Grand Father

Fig 4.8.3: All Patients



Patient Profile

Set Profile Type: My Own

Enter Name

Gender: ○ Male ○ Female

Blood Group: A+

Current Date: DD-MM-YY

Age: 0

Height: 0.0 feet

Weight: 0.0

Phone: Phone Number(optional)

Email: Email(optional)

Current Disease and Condition:

Health condition or disease name.

Fig 4.8.4: Patient Profile

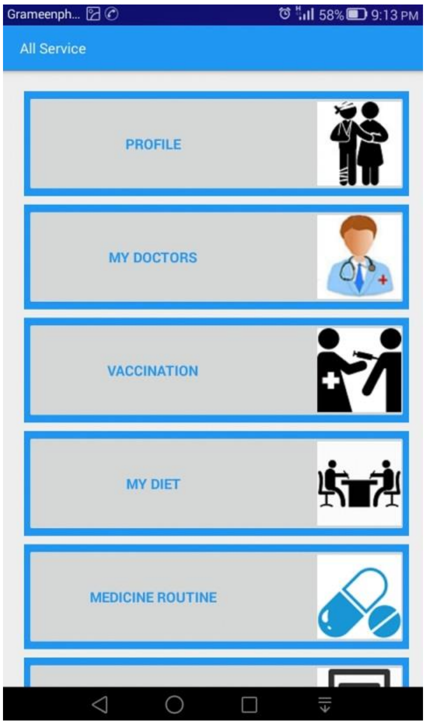
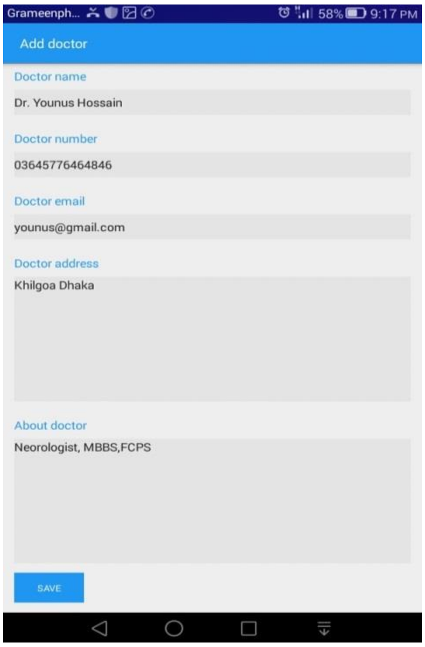


Fig 4.8.5: All Services

4.9 Doctor and Diet Profile

This is the doctor and diet management profile of apps. Here every user can add their doctor and diet information with specific patient such as: name, phone number, mail address, doctor address, expertise field of doctor’s etc. Every user can delete or update their information when needed. User can set alarm with the specific diet schedule which is shown in the following fig 4.9.1.



Doctor name
Dr. Younus Hossain

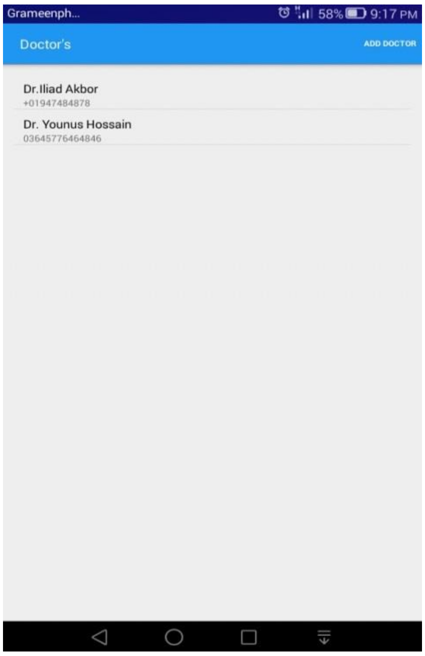
Doctor number
03645776464846

Doctor email
younus@gmail.com

Doctor address
Khilgoa Dhaka

About doctor
Neorologist, MBBS,FCPS

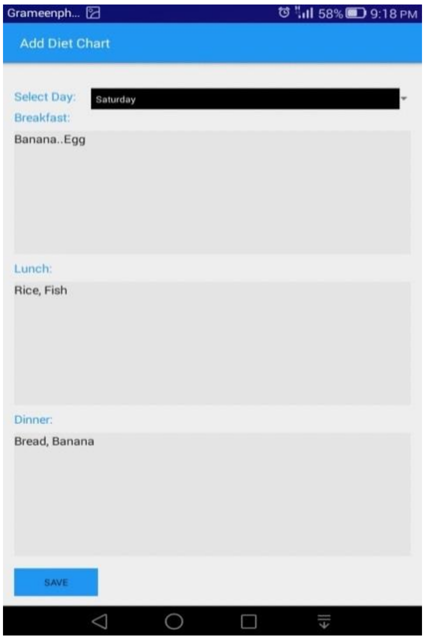
Fig 4.9.1: Doctor Profile



Dr.Iliad Akbor
+01947484878

Dr. Younus Hossain
03645776464846

Fig 4.9.2 All Doctors



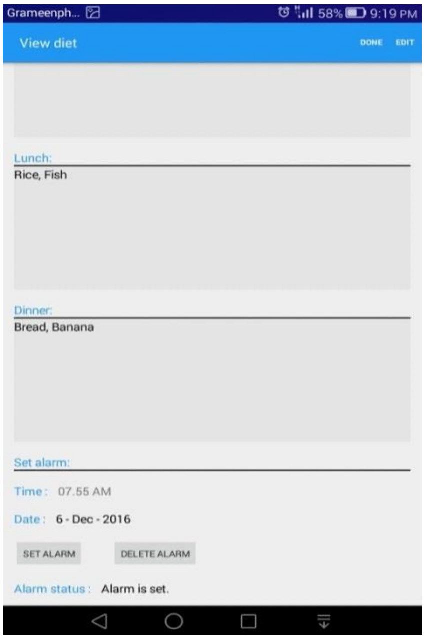
Select Day: Saturday

Breakfast:
Banana..Egg

Lunch:
Rice, Fish

Dinner:
Bread, Banana

Fig 4.9.3: Diet Profile



Lunch:
Rice, Fish

Dinner:
Bread, Banana

Set alarm:

Time : 07.55 AM

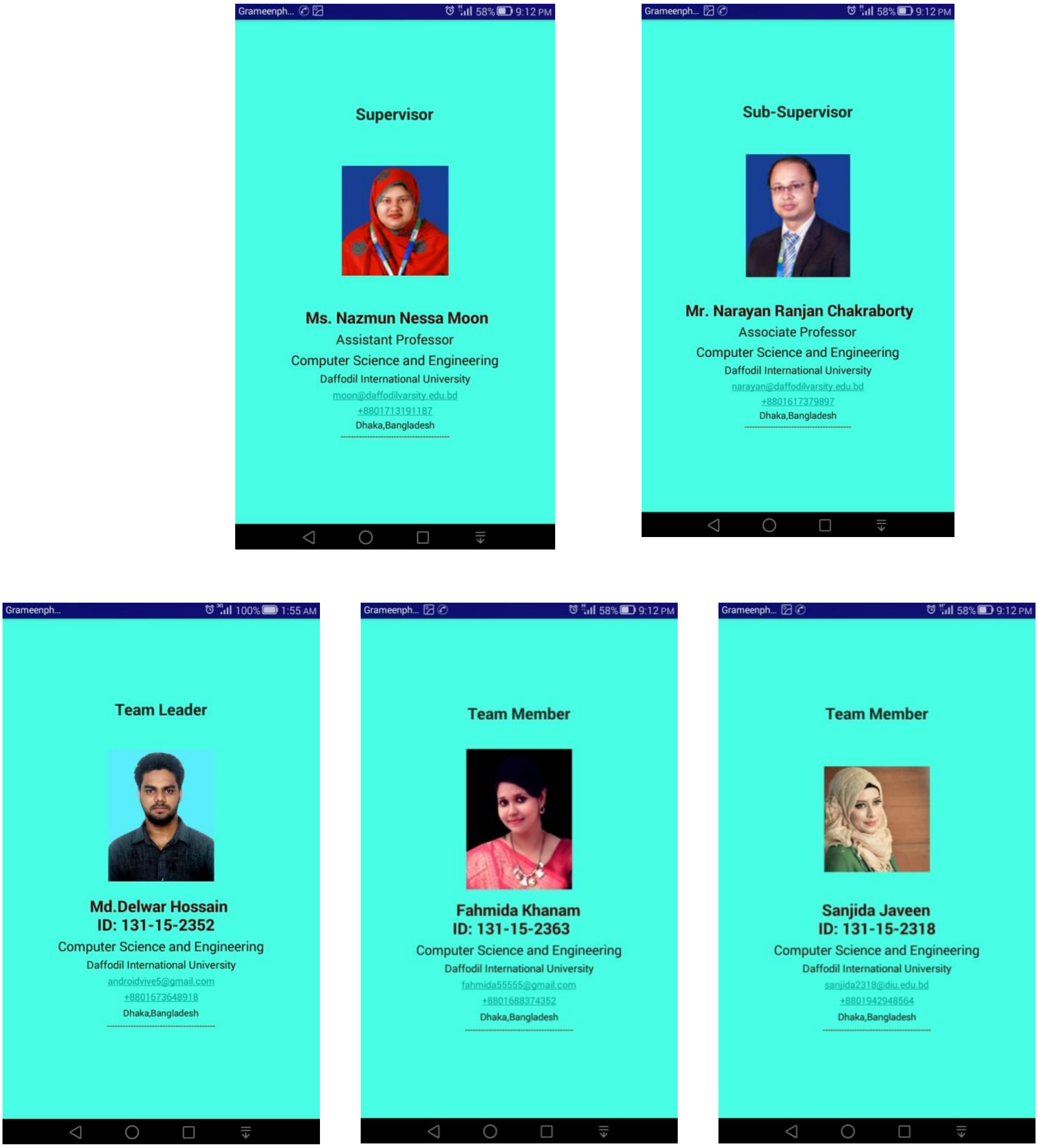
Date : 6 - Dec - 2016

Alarm status : Alarm is set.

Fig 4.9.4: Diet Alarm

4.10 About Us Page

In this page we have added detail about the Supervisor, Sub-Supervisor and Developer information of Personal Health Management System- iCare app shown in Fig: 4.10.Here all the information provides individually. Anyone can contact and send email by clicking number and email directly. It will choose the default call and mail automatically.



Supervisor

Ms. Nazmun Nessa Moon
Assistant Professor
Computer Science and Engineering
Daffodil International University
moon@daffodilvarsity.edu.bd
+8801713191187
Dhaka,Bangladesh

Sub-Supervisor

Mr. Narayan Ranjan Chakraborty
Associate Professor
Computer Science and Engineering
Daffodil International University
narayan@daffodilvarsity.edu.bd
+8801617379897
Dhaka,Bangladesh

Team Leader

Md.Delwar Hossain
ID: 131-15-2352
Computer Science and Engineering
Daffodil International University
androidvive5@gmail.com
+8801673648918
Dhaka,Bangladesh

Team Member

Fahmida Khanam
ID: 131-15-2363
Computer Science and Engineering
Daffodil International University
fahmida55555@gmail.com
+8801688374352
Dhaka,Bangladesh

Team Member

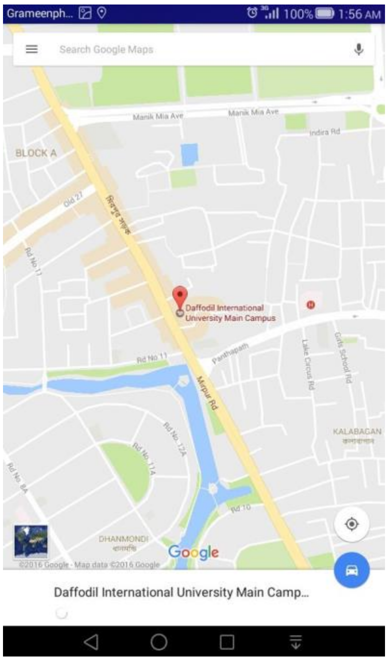
Sanjida Javeen
ID: 131-15-2318
Computer Science and Engineering
Daffodil International University
sanjida2318@diu.edu.bd
+8801942948564
Dhaka,Bangladesh

Fig 4.10: About Us Page

4.11 Check Location and Web Page

We have implemented dynamic app status detection methods in our coding. This dynamic status detection method will detect Internet Status and Back Button unexpected exit status and will show smart alert dialog shown in Fig: 4.11. For Example: Here user will get the maps of DIU’s “Permanent Campus”, “Dhanmondi Campus” and “Uttara Campus”. User will also get DIU website link in “View Website” button of this GUI. While the device is not connected with Internet or internet connects is disable it will pop up a smart Alert dialog which will alert user about the unavailability of internet connection and request user to connect with internet.

.



Daffodil International University Main Camp...

Fig 4.11: Map View

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Discussion

Personal Health Management System iCareis an application by which user can get guidance they need anytime anywhere by help of an android mobile phone. The application is ready to use now.

5.2 Future Scope

This application is developed only for English language now but in near future we will firstly add Bengali language support and step by step we will provide support for other most useable language of the world. It can be improved in many ways and can be extended to support more devices like the Windows Phone, iOS and Black Berry OS.

5.3 Conclusion

This project helps us to care all the family member health information anywhere I stayed. We try to develop a smart way to care a family by smart phone. Now everything is going on digital way. That’s why we try to develop this app to care health by digital way.

REFERENCES

[1] Developer Tools, http://developer.android.com/tools/index.html, Last Accessed on 6th Dec 2014

[2] Get the Android SDk, http://developer.android.com/sdk/index.html, Last Accessed on 13th Nov 2014

[3] Resources Overview, http://developer.android.com/guide/topics/resources/overview.html, Last Accessed on 2nd Jan 2015

[4] Activities, http://developer.android.com/guide/components/activities.html, Last Accessed on 14th Dec 2014

[5] Intents and Intent Filters, http://developer.android.com/guide/components/intents-filters.html, Last Accessed on 17th Dec 2014

[6] Media Playback, http://developer.android.com/guide/topics/media/mediaplayer.html, Last Accessed on 22nd Dec 2014

[7] Activity Codes,http://developer.android.com/guide/topics/manifest/activity-element.html, Last Accessed on 8th Jan 2014

[8] Reto Meier, Professional Android 2 Application Development, 2010 by Wiley Publishing, Inc., Indianapolis, Indiana ISBN: 978-0-470-56552-0, Last Accessed on 26th Nov 2014

[9] Wei Meng Lee, Beginning Android™ 4 Application Development, 2012 by John Wiley & Sons, Inc., Indianapolis, Indiana ISBN: 978-1-118-19954-1, Last Accessed on 7th Oct 2014