



INDORE

Medi-Caps Institute of Technology & Management, Indore

Department of Computer Science & Engineering

April, 2017

Software Requirement Specification

ON

AutoMode

Submitted To

Mr.Hitesh Kag

Submitted By

Swapnil Kuwar (0812CS141103)

Yash Gupta (0812CS141118)

Suyash Jain (0812CS141102)

Department of Computer Science and Engineering

June, 2017

Table of Contents

1. Introduction

- 1.1. Purpose
- 1.2. Scope
- 1.3. Definitions, acronyms & abbreviations
- 1.4. Overview

2. Overall description

- 2.1. Product perspective
- 2.2. Product functions
- 2.3. User characteristics
- 2.4. Constraints
- 2.5. Assumptions and dependencies
- 2.6. Apportioning of requirements

3. Specific Requirements

- 3.1 External interface requirements
- 3.2 Specific requirements
- 3.3 Performance requirements
- 3.4 Design constraints
- 3.5 Software system attributes
- 3.6 Other requirements

4. UML Diagrams

5. Test Cases

1. Introduction

This section gives a scope description and overview of everything included in this SRS document. Also, the purpose for this document is described and a list of abbreviations and definitions is provided.

1.1. Purpose

The purpose of this document is to give a detailed description of the requirements for the “AutoMode” software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

1.2. Scope

This “AutoMode” software is an android based mobile application which is a GPS service based application which would help us in locating the exact geo-position of people depending upon their current location/whereabouts and updates the profile to a desired mode based on entering to a restricted zone already saved. “AutoMode” app feature allows user such as student in a college to get details of all events happening in college and go through timetable of every day in a single swipe rather than going through everywhere around campus. This app can be used at any location which is available on google map. The administrator shall register the users on the app for use. The users shall set the location via input location where they want a desired profile. The details of login will be stored on a database which the admin can access after correct login and rest information would be stored on mobile system database.

The database is stored on a server for shared access by user and admin. The application should be provided permissions to use the internet connectivity and to access your device location using GPS turned ON.

1.3 Definitions acronyms, and abbreviations

Term	Definition
User	Someone who interacts with the mobile phone application
Admin/Administrator	System administrator who is given specific permission for managing and controlling the application
Server	A server is a computer program or a device that provides functionality for other programs or devices, called "clients".
GPS	Global Positioning System
SQLite	SQLite is a open source SQL database that stores data to a text file on a device
GMS	Google map service
XML	Extensible Markup Language
INP_LOC	Input Location (where user enter its location details using google map)
CURR_MOD	Current Mode (where user can set its current profile for its mobile phone)
SAV_LOC	Saved Location (where user can view and edit location details saved and set profile)
LBS	Location Based Service
API	Application Programming Interfaces
SDK	Android Software Development Kit

1.4. Overview

The remainder of this document contains details of the system. The document has been classified in terms of overall description, specific requirements and appendix.

Overall description includes product perspective, product functions, characteristics, assumptions on the following section 2 of this document.

Next, the document addresses the specific requirements of the system, which covers the external interface requirements, requirements of the system, performance requirements, and other requirements in section 3.

1.5. References

- <http://developer.android.com/guide/basics/what-is-android.html>
- Android Architecture 2010[R/OL].
- Android Official Website (2008)—“Android | Official Website”,
<<http://www.android.com/>>
- Suryabhan Gupta, Amol Joglekar, "A CHANGE OF PROFILE BASED ON LOCATION" in
International Journal of Research in Engineering and Technology Volume: 04 Issue: 05 | May-2015.

2. Overall Description

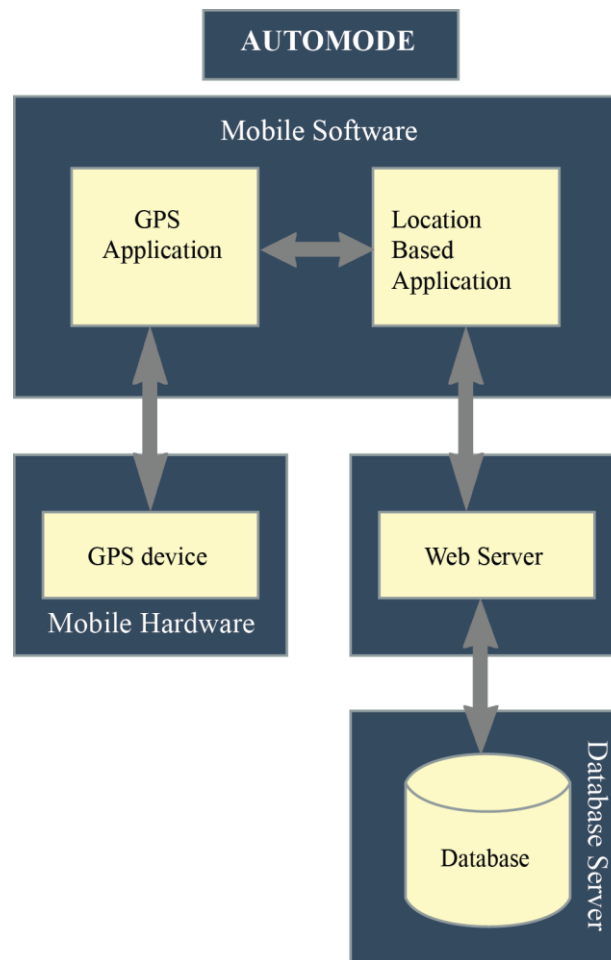
This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other and introduce the basic functionality of it. It will also describe about the functionality that is available for user using the application. At last, the constraints and assumptions for the system will be presented.

2.1. Product Perspective

System will consist of mobile application and server. The first part of the project involves development of a GPS service based application which would help us in locating the exact geo-position of people depending upon their current location/whereabouts and updates the profile to a desired mode based on location. The second part of the project involves College data management onto Android enabled devices.

The server will be used for managing the information about the user credentials and also data related to college.

In our application AutoMode we are going to implement automatic profile changing facility means using this feature of our android app, the profile of user's mobile device will automatically change according to places where person goes. The user needs to register the particular places/location for which he wants to change the profile.

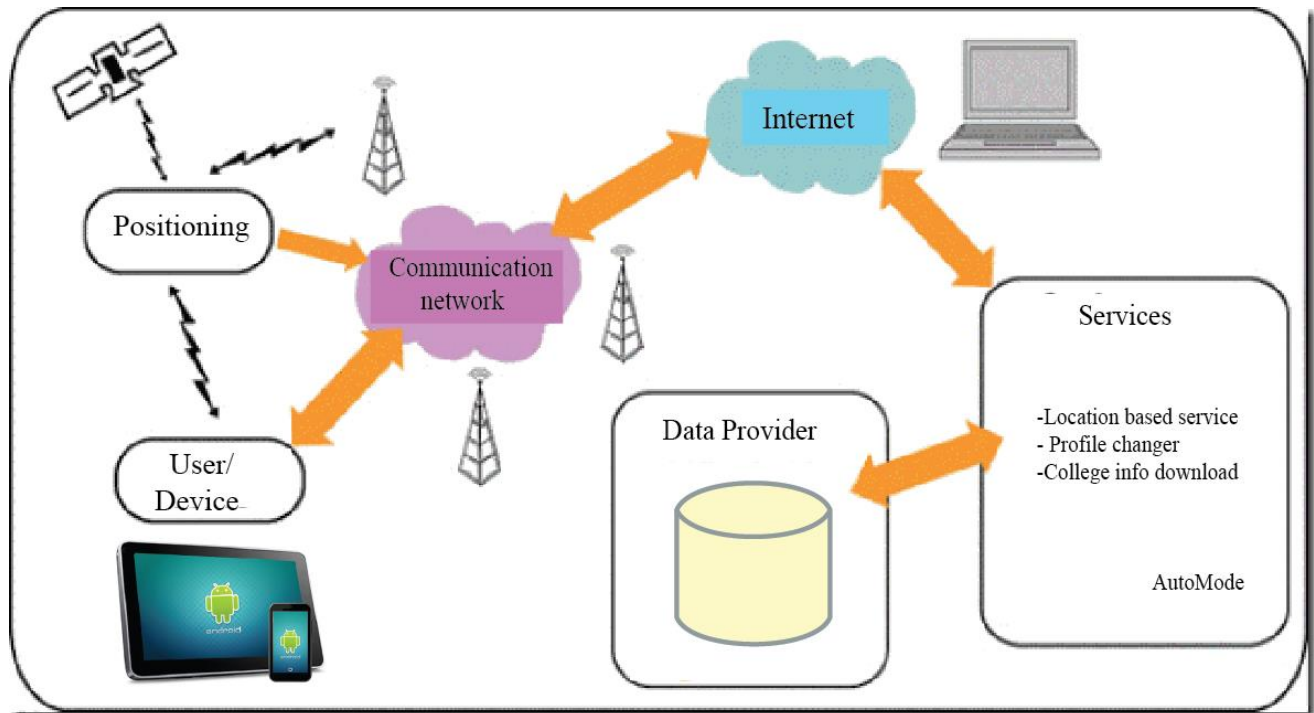


And accordingly the profile changer will work in that particular registered perimeter only. Here first the user's mobile device will locate using GPS technology then according to place the profile of mobile will change. Sometimes the person forgets to change the profile of mobile phone at certain places, so this app will help which automatically change profile.

The GPS will provide the mobile application with locations of user. The functionality provided by the GPS will be embedded into the application in order for the user to be able to use the functions in the application in a seamlessly manner.

Since this is a data-centric product it will need somewhere to store the data. For that, a database will be used.

Both the mobile application and server will communicate with the database, however in slightly different ways. The mobile application will only use the database to get data while the server (admin) will also add and modify data.



2.2. Product Functions

The application is built to provide a systematic way of profile changing according to the GPS coordinates. And also provide student best way to go through information related to college. With the mobile application, the users will be able to change profile of mobile, based on location set by user and when user entered the restricted location.

The product functions include:

- **Account Registration:-**

The registration function shall allow users to create accounts. The account will take the user's name, email address, password and confirm password.

- **Account Login:-**
The account login function shall allow account members to enter their username and password. Once verified, users will be able to input location, set current mode, view saved locations and also college information.
- **Input Location:-**
This function shall offer users the ability to input the location address in search bar to get geo position of location (viewed by a marker set on map) on Google map (using Google API) having internet and GPS enabled and then finally saving the location simply clicking on save button on screen.
- **View Saved Location:-**
The result of the saved input location in input location activity will be viewed here in a list view. The list view for each saved location will contain location id, complete address, latitude and longitude of location and mode set at that location (which is Silent by default and can be modified).
- **Set Current Mode:-**
Here user will able to set profile of mobile for current usage of mobile. Profile can be set as one of the three i.e. Sound, Vibrate, Silent.
- **View College Information:-**
Here user can get access college information such as timetable of the particular day. And one can also view the recent event happening in the college. These would be downloaded from internet as pdf files.
- **About Us:-**
The about us function shall give users the information about the application and for query developers contact details such as email and phone number.
- **Account Logout:-**
The account logout function shall allow account members to exit their account for security purposes.

2.3. User Constraints

The users do not need to have any special technical skills to use the product. However, since the application is android based, a fair experience with android devices is appreciated and required.

There are two kind of user which uses our app, User (student) & admin.

User will use the application to input location via g-map to set the location and save it. And further set the profile which the mobile should be set when user entered the saved location.

The admin manages the server and handle the information about the college.. It analysis the daily registration and check the performance of software.

2.4. Constraints

The Internet connection is a constraint for the application. Since the application fetches data from the database over the Internet, it is crucial that there is an Internet connection for the application to function. The application will be constrained by the capacity of the database. Since the database is shared between both user and admin applications it may be forced to queue incoming requests and therefore increase the time it takes to fetch data.

Also the app shall be constrained by the number of users connected simultaneously. The scalability shall not introduce any issues in operation of user applications.

The mobile application is constrained by the system interface to the GPS navigation system within the mobile phone. Since there are multiple system and multiple GPS manufacturers, so if there arises any difference the application will use GMS (Google map service), so it is same for every one of them. Therefore application needs a GPS and GOOGLE PLAY SERVICE on your device to run effectively.

2.5. Assumptions and dependencies

One assumption about the product is that it will always be used on mobile phones that have enough performance and are connected to the network. If the phone does not have enough hardware resources available for the application, the app may lag on certain devices though will still work.

Minimum Android OS version on android enabled devices should be Kitkat or above to use this application. Another assumption is that database is always up to date and accurate so the details presented by app won't be false.

The following dependencies are required by the app:

- com.android.support:appcompat-v7:22.1.0
- com.android.support:support-v4:22.0.1
- com.google.android.gms:play-services-maps:8.4.0
- com.google.android.gms:play-services-location:8.4.0
- com.google.android.gms:play-services-gcm:8.4.0
- com.squareup.okhttp3:okhttp:3.6.0

Another assumption is that the device running the application has Google Play Services installed in it which helps to run Google map service. Another assumption is that device support GPS on it and GPS should be enabled for use of the application.

2.6. Apportioning of requirements

In the case that the project is delayed, there are some requirements that could be transferred to the next version of the application. Those requirements are to be developed in the second release.

These may include:

- Support for setting up reminders on location based.
- Notification facility for the student about the upcoming events and also about location saved.
- Message alert by admin for any changes in college information.

3. Specific Requirements

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all its features.

3.1. External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

3.1.1 User Interface

The system offers interaction facility to the user via graphical user interface. The application's GUI contains numerous activities which undergo transition to provide required output. The activity is a single, focused thing that the user can do. Almost all activities interact with the user, creating a user interactive window.

3.1.2 Hardware Interface

For AutoMode application to run, a mobile device with android as operating system is required. The app will interact with the network component of the device to establish internet connectivity and local storage for cache memory.

Hardware-

- Any Android Enabled Handheld
- Android OS Version : Kitkat & Above
- Google Maps API Compatible

Since the mobile application does not have any designated hardware, it does not have any direct hardware interfaces. The physical GPS is managed by the GPS application in the mobile phone and the hardware connection to the database server is managed by the underlying operating system on the mobile phone.

3.1.3 Software Interface

As it has been stated earlier, AutoMode runs on any android device so it does not need any specific software from the user end. But it does make use of native android libraries which are included in the app package itself.

Software-

- Server Side : Web Server – Apache Tomcat software
- Server Side : Database Server – MySQL 4.1 or higher
- Client End : Android
- Client End Database : SQLite Database

The mobile application communicates with the GPS application in order to get geographical information about where the user is located. The communication between the database and the mobile application consists of both reading and modifying the data.

3.1.4 Communication Interface

The communication between the database and the app at admin side consists of operation concerning both reading and modifying the data, while the communication between the database and the app at client side consists of only reading operations.

3.2. Functional Requirements

These are the statements of services the system should provide, how the system should react to a particular inputs and how the system should behave in a particular situation.

3.2.1 Functional Requirement 1.1

ID: FR1

TITLE: Download mobile application

DESC: A user should be able to download the mobile application through a link on internet. The application should be free to download.

DEP: None.

3.2.2 Functional Requirement 1.2

ID: FR2

TITLE: User registration

DESC: Given that a user has downloaded the mobile application, then the user should be able to register through the mobile application. The user must provide required details such as name, email, password and confirm password.

DEP: FR1

3.2.3 Functional Requirement 1.3

ID: FR3

TITLE: User Log-in

DESC: Given that a user has registered, then the user should be able to login in to the mobile application. The login information will be stored on the phone and in the future the user should be logged in automatically.

DEP: FR1, FR2

3.2.4 Functional Requirement 1.4

ID: FR4

TITLE: Location Tab

DESC: Given that a user downloaded the mobile application, then the first page that is shown should be the Location tab. The user should be able to use the main services of app (related to changing profile based on location) from this page.

DEP: FR1, FR2, FR3

3.2.5 Functional Requirement 1.5

ID: FR5

TITLE: Places Tab

DESC: Given that a user has log in successfully, then this page is obtain by swiping right to main page which results in Places tab. The user should be able to use the services related to retrieving college information from this page.

DEP: FR1, FR2, FR3

3.2.6 Functional Requirement 1.6

ID: FR6

TITLE: Current Mode

DESC: The user once on location tab, can set Current Mode of mobile using this page. Profile can be set as Sound, Vibrate, Silent.

DEP: FR1, FR2, FR3, FR4

3.2.7 Functional Requirement 1.7

ID: FR7

TITLE: Input Location

DESC: User can input desired location in search bar (using text or voice) and would be navigated to that location on map using Google map service. Further entered input location will direct to specified location on map and would be set by a marker which can be saved.

DEP: FR1, FR2, FR3, FR4

3.2.8 Functional Requirement 1.8

ID: FR8

TITLE: Location match found

DESC: If no match is found for entered location, the user should be informed but kept on the search page in order to get the possibility of new search.

DEP: FR6

3.2.9 Functional Requirement 1.9

ID: FR9

TITLE: Saved Location

DESC: User can view the saved location (saved previously in input location). The list view for each saved location will contain location id, complete address, latitude and longitude of location and mode set at that location. User here will be able to change mode of particular saved location and delete that location also from the list view.

DEP: FR1, FR2, FR3, FR4, FR6

3.2.10 Functional Requirement 1.10

ID: FR10

TITLE: College Information

DESC: User can view college information like daily timetable and events which can be downloaded from internet in pdf format.

DEP: FR1, FR2, FR3, FR5

3.2.11 Functional Requirement 1.11

ID: FR11

TITLE: GPS check

DESC: When user logs in successfully, on the first page user will be asked to enable mobile GPS so as to use Google map services.

DEP: FR1, FR2, FR3

3.2.12 Functional Requirement 1.12

ID: FR12

TITLE: Quick access

DESC: User can have quick navigation to many pages from navigation bar at bottom on main page. Navigations are for home page, input location, saved locations and about us.

DEP: FR1, FR2, FR3

3.3 Performance requirements

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance.

The performance requirements are as follows:

- System login/logout shall take less than 5 seconds.
- Searches shall return results within 10 seconds.

- **Non Functional Requirements**

In requirement engineering, a non functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system rather than specific behaviours.

3.3.1 Non-Functional Requirement 1.1

ID: NR1

TITLE: Usage of the result in the saved location.

DESC: The results displayed in the list view should be user friendly and easy to understand. Selecting a location in the result list should only take one click.

DEP: None

3.3.2 Non-Functional Requirement 1.2

ID: NR2

TITLE: System Dependability

DESC: The system should be able to perform fault tolerance.

DEP: None

3.3.3 Non-Functional Requirement 1.3

ID: NR3

TITLE: Efficient Query Management and Database

DESC: The database should consume minimal space and time for query optimization.

DEP: None

3.3.4 Non-Functional Requirement 1.4

ID: NR4

TITLE: Easy user interface

DESC: To give user an easier interface so that human efforts are reduced with minimal touch.

DEP: None

3.3.5 Non-Functional Requirement 1.5

ID: NR5

TITLE: Faster retrieval of data.

DESC: The data should be retrieved fast and should not make app and waiting time of user slow.

DEP: None

3.4. Logical Structure of the Data

The types of information used by various functions of the application:

Functions	Type of Information Used
Account Registration	User information (username, email address, password and confirm password)
Account Login	User information (username and password)
View Saved Location	Location Details(location id, location full address, latitude and longitude of location, mode of location)
View College Info.	Timetable and Event(day name, event name)

3.5. Design constraints

This section includes the design constraints on the software caused by the hardware.

- **Disk Space Requirement**

Description: Application's need of memory on secondary storage

Minimum disk space required: 25 Mb

Recommended disk space: 25 Mb

Unit: Mb (Megabyte)

- **Application memory usage**

Description: The amount of Operate System memory occupied by the application

Minimum disk space required: 200 Mb

Recommended disk space: 200 Mb

Unit: Mb (Megabyte)

3.6. Software system attributes

The requirements in this section specify the required reliability, availability, security and maintainability of the software system.

3.6.1 Reliability

The reliability of the system will ensure that the system gives the right result on a search. In testing more than 98% of the searches must be reliable.

3.6.2. Availability:

Application will be available to user 24x7. This will provide flexibility to the user to use services of app at any time.

- **GPS Connection:**

The app should be connected to the GPS device. In order for the app to get the user's location, the map and to calculate the distance

- **Internet Connection:**

The app should be connected to the Internet. In order for the app to communicate with the database.

3.6.3. Security:

We will provide well equipped security features so that authorized user can only login and also our application will be safe from malicious users.

3.6.4. Maintainability:

- **Application extendibility:**

Our software can easily extended in near future. The code should be written in a way that it favors implementation of new functions. Any updates or defect fixes shall be able to be made on server-side computers only without any patches required by the user.

- **Application testability:**

Test environments should be built for the application to allow testing of the application's different functions.

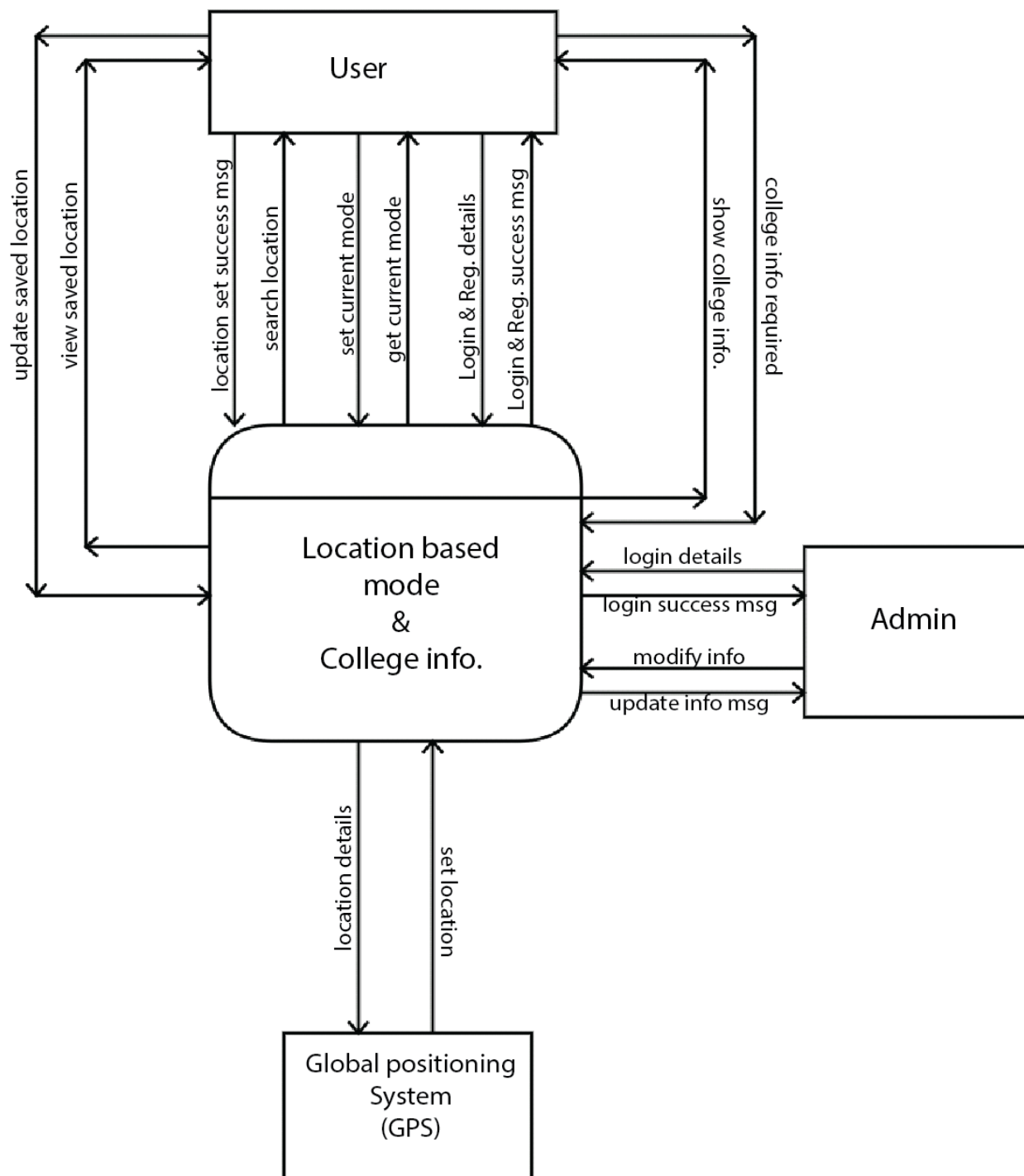
3.6.5. Portability:

Our application can be run on android enabled phones having android OS (SDK 17 to SDK 23)

4. UML Diagrams

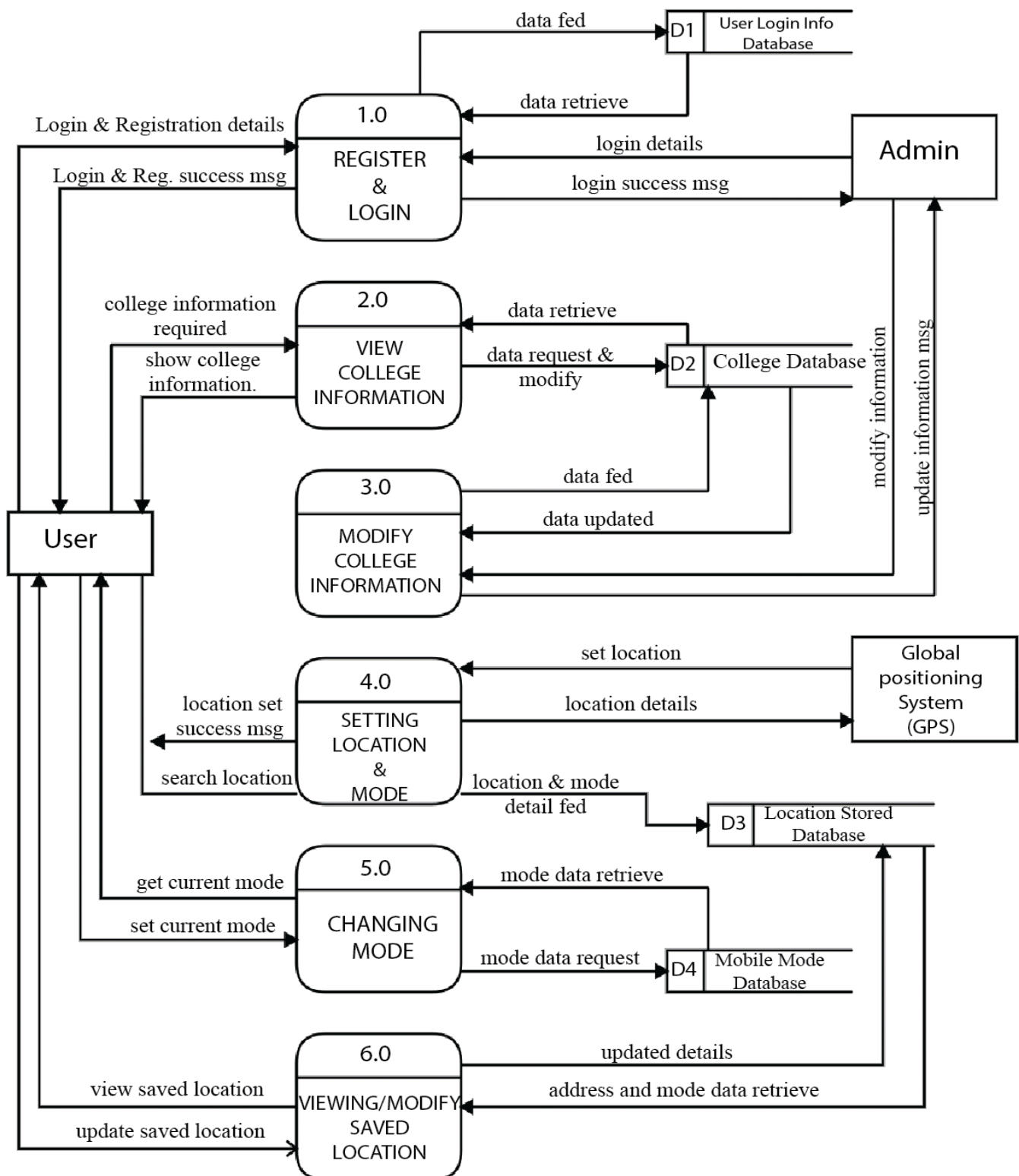
Submission Date: 20/02/2017

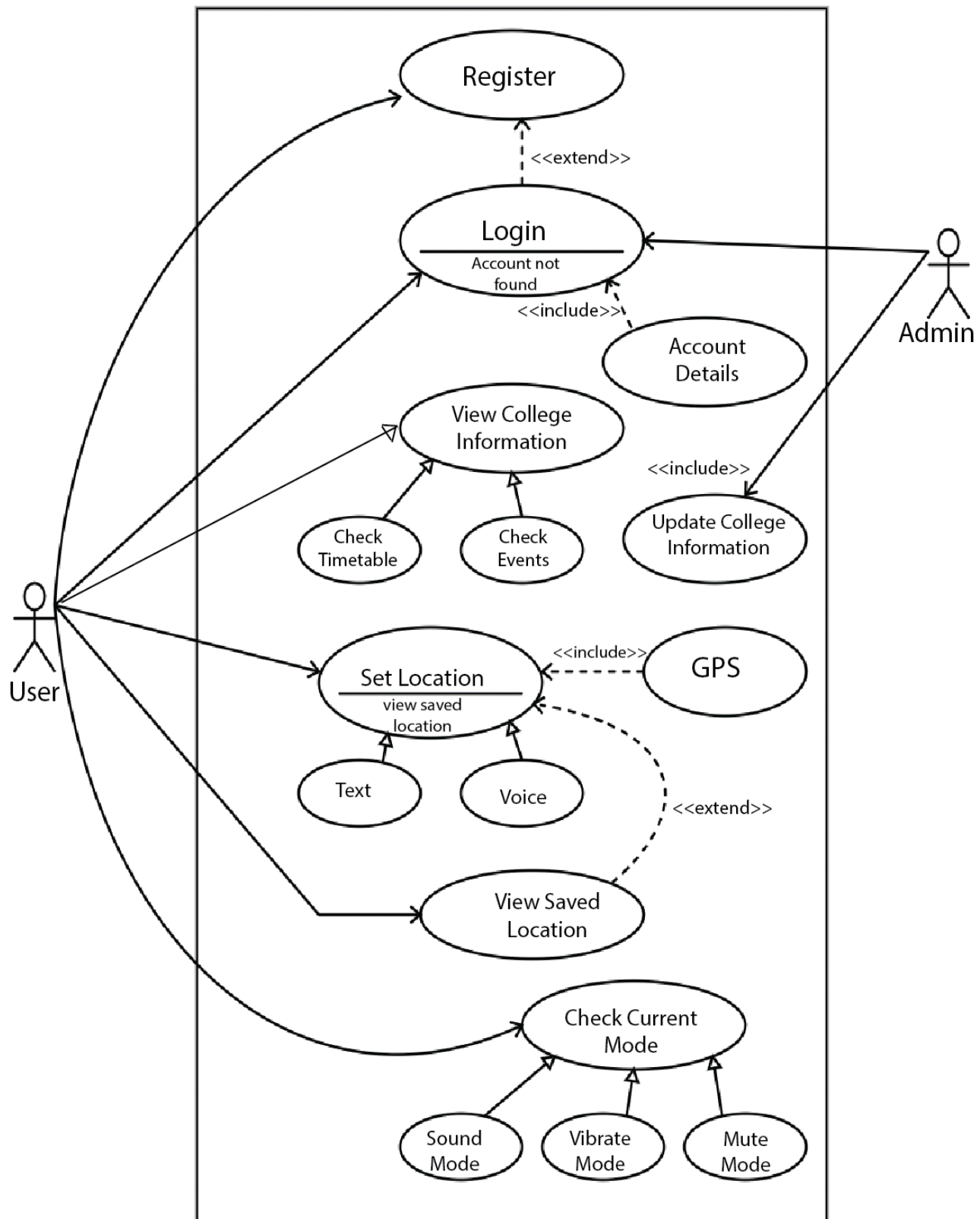
4.1 Context Diagram of AutoMode



4.2 Data Flow Diagram (Level 0) of AutoMode

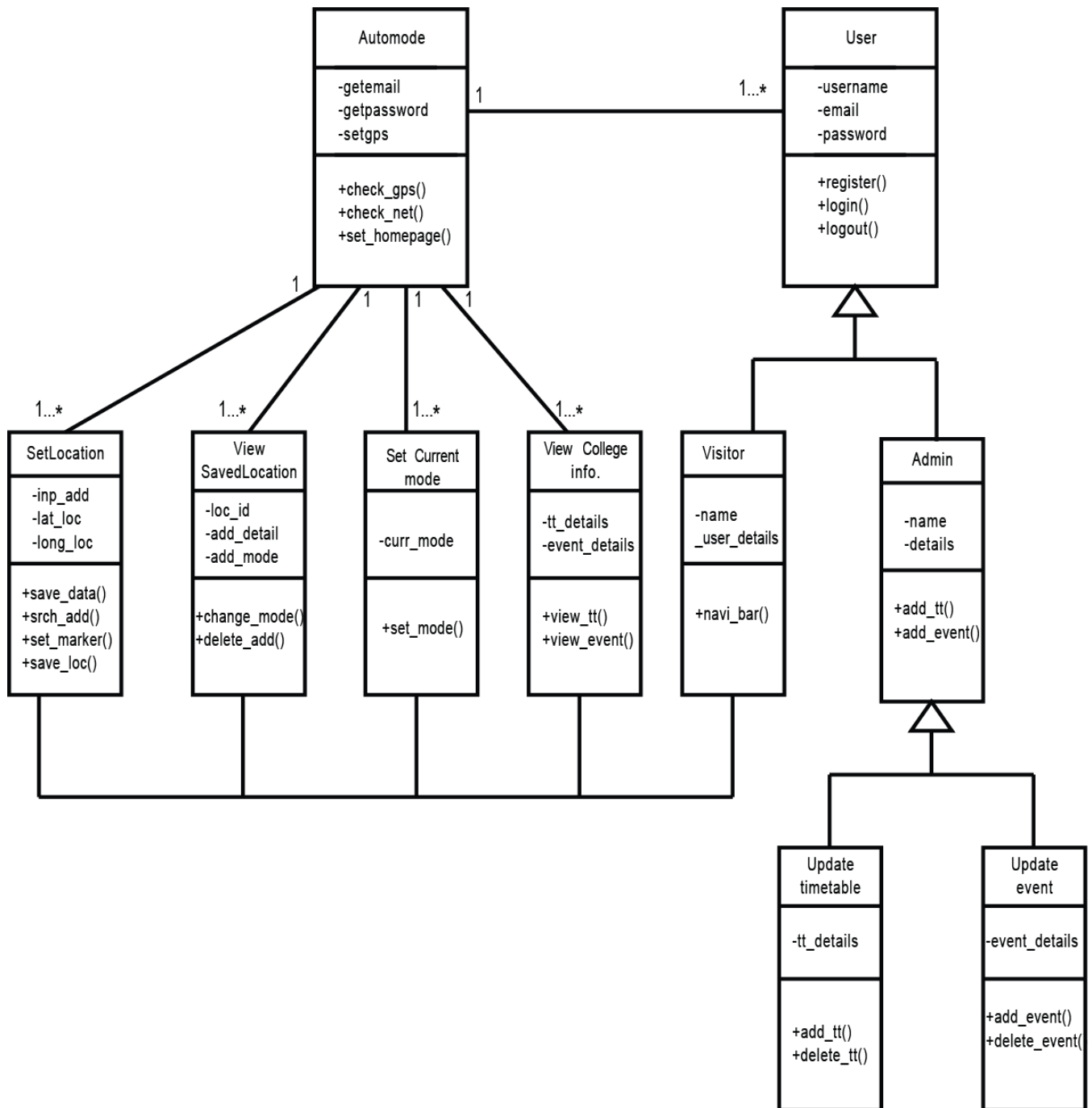
Submission Date: 20/02/2017





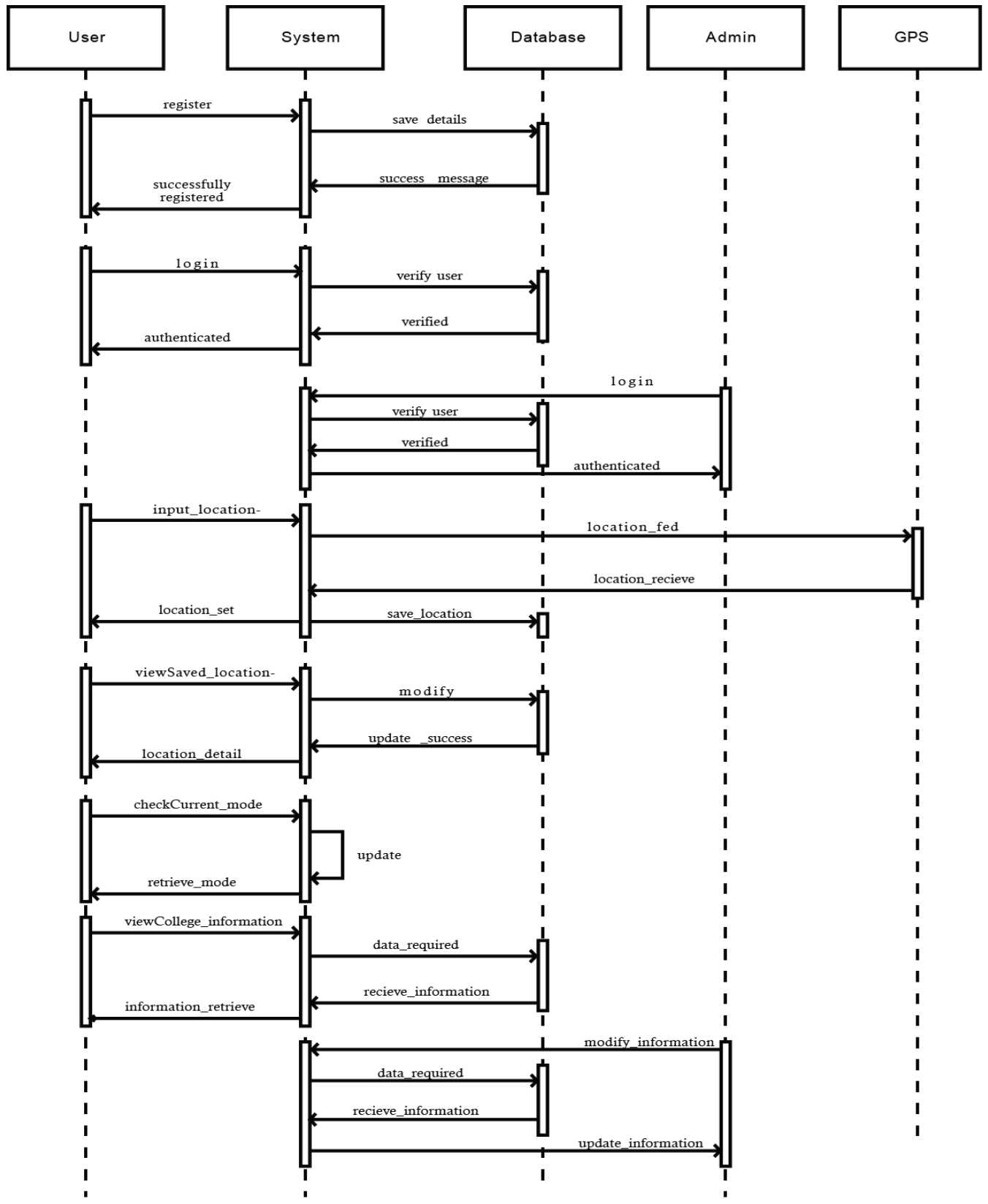
4.4 Class Diagram of AutoMode

Submission Date: 06/03/2017



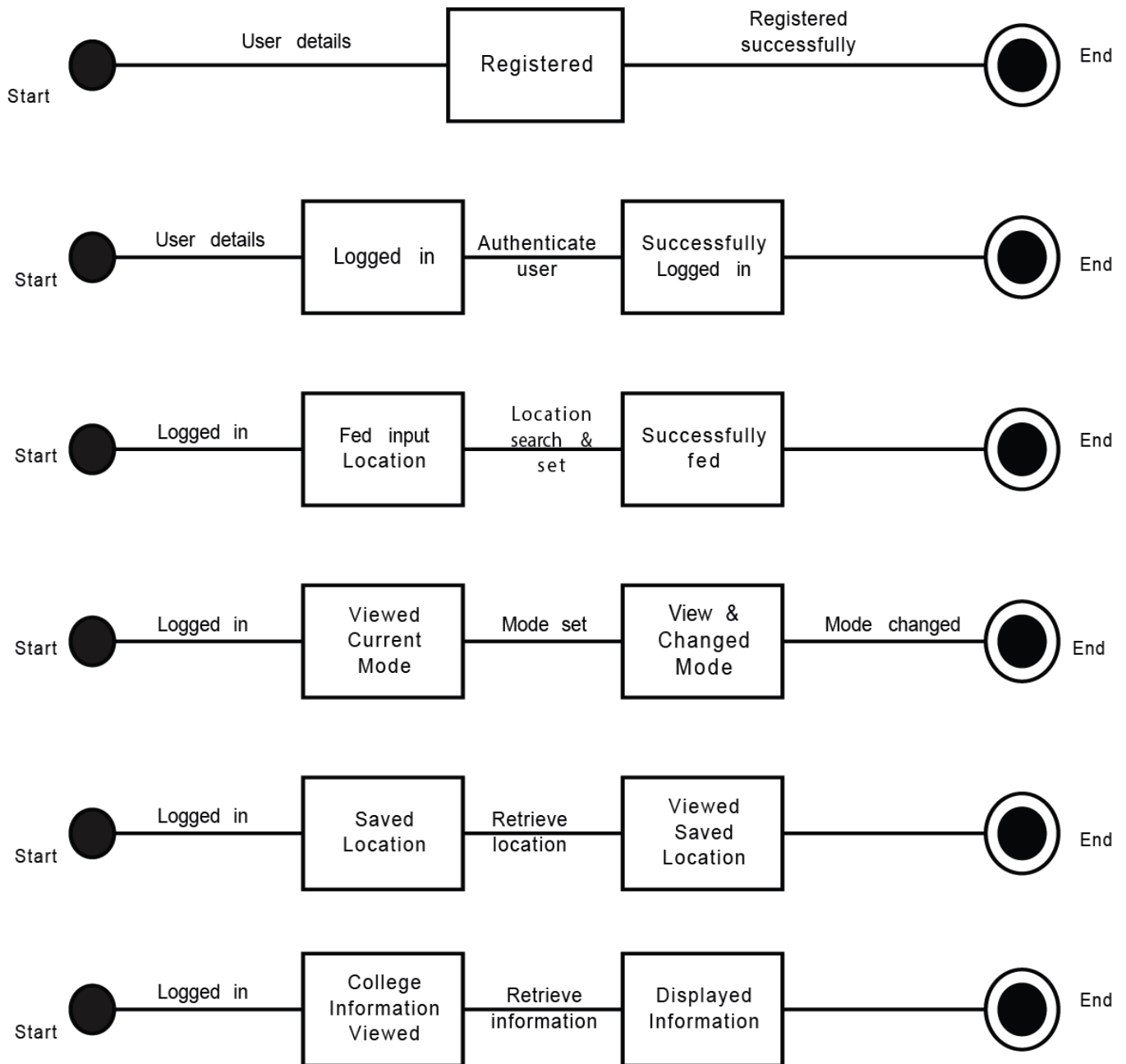
4.5 Sequence Diagram of AutoMode

Submission Date: 20/03/2017



4.6 State Diagram of AutoMode

Submission Date: 10/04/2017



4. Test Cases

4.1 Black Box Testing

BLACK BOX						
S.No.	Test Name	Test Objective	Test State	Test Data	Expected Output	Output Result
1	Registration	To check whether the user has registered with unique username.	Registration has already been done.	Clicked on Register button.	Unique Registration Message.	Register with not used credentials.
2	GPS Enable	To check the GPS is on	GPS must be enabled	To find current location	GPS enabled	PASS
3	Validations	To check whether the user has entered correct details.	Registration Page with all the required fields is available.	Clicked on Register button.	Validation message.	Registered Successfully.
4	Login	To check whether the user has logged in with valid credentials.	Login Page with all the required fields is available.	Clicked on Login button.	Successful Login.	Logged In.
5	College updates	User must update app	Check for updates	College information and event details	App not updated message	App update window open
6	No internet access	To check whether the user has searched with network on.	user has logged in and is on main page	Clicked on input location.	Network not available message.	No location searched
						FAIL

4.2 White Box Testing

WHITE BOX							
S.No.	Test Name	Test Objective	Test State	Test Data	Expected Output	Output Result	Remarks
1	Input Location	To Input Location	Map opens to search location	Click on search bar. Location entered	Location found	Location not found	FAILED
2	Current mode	To check current mode	Application shows current mode tab	Click on change modes	Can't changed the mode	Mode not changed as already set for phone	FAILED
3	Edit saved location	To edit saved location	Page with saved location appears	Click on delete location	Successfully delete	Location deleted successfully	PASS
4	Add location	Add location and mode	G-map locating the location	Location set and mode saved	Location added	Location added with saved mode	PASS
5	Map load	To load the g-map	Loading of g-map	Search location on google map	Map loaded	Successfully loaded	PASS

4.3 Regression Testing

REGRESSION TESTING							
S.No.	Test Name	Test Objective	Test State	Test Data	Expected Output	Output Result	Remarks
1	Input Location	To Input Location	Map opens to search location	Click on search bar Location entered.	Location found message	Location viewed on map	PASS
2	Validations	To check whether the user has entered correct details. The changes made are that validations has been applied to all the fields on the page.	Registration Page with all the required fields is available.	Clicked on Register button.	Validation message.	Proper validation messages for each incorrect detail appeared.	PASS
3	No internet access	To check whether the change in code during the regression testing solves the error or not. The check for Internet connection has been applied.	Search engine is available with search button.	Clicked on Search button.	Network not available message.	Network not available message appeared.	PASS
4	Current mode	To check current mode	App shows the current mode tab	Click on change mode	Successfully mode changed	Mode set for phone	PASS