

Software Requirements Specification

for

KoiTui

Friend Finding Service System (Phone Application)

Version 1.0

Prepared By

Group 1

Chakraborty, Uchchhwash	15-28458-1
Ishtian, Rahim Md.	15-28490-1
Rahman, Alinoor	15-28700-1
Khan, Rifat Ahmed	15-29303-1



AMERICAN INTERNATIONAL UNIVERSITY - BANGLADESH

October 26, 2016

Table of Contents

Table of Contents	2
User Story	3
Process Model.....	4
1. Introduction.....	6
1.1 Purpose.....	6
1.2 Document Conventions.....	6
1.3 Intended Audience and Reading Suggestions	6
1.4 Project Scope	6
1.5 References.....	7
2. Overall Description	7
2.1 Product Perspective.....	7
2.2 Product Features	8
2.3 User Classes and Characteristics	9
2.4 Operating Environment.....	9
2.5 Design and Implementation Constraints	9
2.6 User Documentation	10
2.7 Assumptions and Dependencies	10
3. System Features	10
3.1 System Feature 1[Login].....	11
3.2 System Feature 2[Sign Up].....	11
3.3 System Feature 3[Create Profile].....	11
3.4 System Feature 4[Edit Profile].....	11
3.5 System Feature 5[Time Scheduling].....	11
3.6 System Feature 6[Change Class Status].....	12
3.7 System Feature 7[View Timeline]	12
3.8 System Feature 8[Search Friends]	12
3.9 System Feature 9[Send Friend Request].....	12
3.10 System Feature 10[Messaging]	12
3.11 System Feature 11[Phone Call]	13
3.12 System Feature 12[Notifications]	13
3.13 System Feature 13[Show Friend List].....	13
3.14 System Feature 14[Report]	13
3.15 System Feature 15[Settings]	13
4. External Interface Requirements	14
4.1 User Interfaces	14
4.2 User Interface Work Flow	19
5. Other Nonfunctional Requirements.....	20
5.1 Performance Requirements	20
5.2 Safety Requirements	20
5.3 Security Requirements	20
5.4 Software Quality Attributes	21
6. Other Requirements	22
Analysis Models.....	22

User Story

We are Software Engineering students in AIUB. We know that many private universities in Bangladesh provide open credit system. Therefore most of the students face lots of problems such as most often they can't register for courses properly, they can't even take the class times they wanted and so on. Among them I also have seen that students are facing tough conditions to look for their friends, seniors for certain helps and needs. It is tough to remember everyone's daily class schedules, their free times during university, their location. Maybe sometimes we need some important meetings or get in touch with classmates for some certain reasons such as collecting study materials or a hangout.

So we need a mobile based software so that students can use it to get connected with each other. They will be able to share their location and their daily class schedules through this application called '**Koitui**'.

This system will be online-based and it will use a database server to store information. The system will automatically add new registered member to the database. It will also keep information about students class schedules. Student will be able to maintain their privacy according to their choice. Also they will be able to share their location, contact numbers and schedule with whom they want to. And by doing so they will be able to know which of their friends are available at that current time.

There will be few tap search option to see which of your friends are free and closest to your location. All of the services will be free. Users will also be able to send friend request, call and message friends of their list. There will also be status indicator for giving your current status if you are free, in class or not available.

Process Model

To produce this system we need to follow a process model which could lead the development easy and efficiency. **Agile Development** strategy is good for this type of systems development. We know that agile is based on iterative enhancement and opportunity development process. Each iteration is a self-contained, mini-project with activities that span requirements analysis, design, implementation, and test. In agile there are small products and team also scalability is limited.

Agile is just a concept as well as a set of other methodologies. To develop this application system 'KoiTui' we are going to follow **Extreme Programming (XP)** methodology. Our team is small. So, it is perfect process model for the development of this type of application system.

The life cycle of XP consists of five phases:

Exploration
Planning
Iterations to Release
Productionizing
Maintenance and Death

Exploration Phase : In this phase the main task is to create stories. Here we are going to write out the story cards that we wish to be included in the first release. At the same time the team will familiarize themselves with the tools, technology and practices we will be using in this project. This phase will take few weeks to few months.

Planning Phase : This phase will require the stories for the next iteration according to priorities. Effort will be estimated here also.

Iterations to Release Phase : This is achieved by selecting the stories that will enforce building the structure for the whole system. Each iteration will take one to four weeks to implement.

Productionizing Phase : There will be a small release of our project. Requires extra testing and checking of the performance of the system before the system can be released

Maintenance and Death Phase : Here all stories are implemented and there are no more changes to the architecture we are going for the final release.

Now for this type of process model we need to follow some mandatory facts. These are -

Communication : We have to make sure that everyone in the team is communicating with each other.

Simplicity : We have to design a simple architecture of the system. System have to be very simple and environmental.

Feedback : The development team need to get the feedbacks when development is running.

Courage : The other three values allow the team to have courage in its actions and decision making.

Respect : Team members need to care about each other and about the project.

For the development we need to focus on some XP practices as well such as planning game , short releases , metaphor , simple design , testing , pair programming , continuous integration , coding standards ,open workspace and so on.

So, we think that we are able to develop our system by following this process model. This process model is best for this type of small system development and for the small team as well.

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 creating an android application '**KoiTui**' which is a Friend Finding Service System. It will illustrate the purpose and complete declaration for the development of system. It will also explain the purpose and features of the system, the 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 Document Conventions

This document follows MLA Format. Bold-faced text has been used to emphasize section and sub-section headings. Highlighting is to point out words in the glossary and italicized text is used to label and recognize diagrams.

1.3 Intended Audience and Reading Suggestions

The intended audience for this document is the project sponsor, the development team and the team advisor. Throughout the rest of this document the project will be broken up into sections for: Project Description, System Features, External Interface Requirements, and Non Functional Requirements. There is also a Glossary of common terms found throughout the document and some Analysis Models for this project.

1.4 Project Scope

This project '**KoiTui**' is a online-based mobile application which helps the students of private universities in Bangladesh to find their friends who are available during class breaks based on the other users current status like if they are free or in class or not available. The application should be free to download from either mobile phone application store or similar services.

Users can provide their daily class routine information to this application. This information will act as the bases for the search results displayed to the user. This is the main goal of this project to give the information of other students class schedules and break times to the user within few taps so that they can easily get what they are searching for. The benefit of this is that after

getting the information they can communicate with the other user with some internal communication services and can catch up and hang out together at that time.

Furthermore, the software needs Internet connections to fetch and display results. All system information is maintained in a database , which is located on a web-server. Also for communication it has messaging and phone call service. The application also has the capability of representing other users daily schedules and profile information.

1.5 References

[1] IEEE Software Engineering Standards Committee, "IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications", October 20, 1998.

[2] Feldt R, "re_lecture5b_100914", unpublished.

[3] Davis M A, "Just Enough Requirements Management: Where Software Development Meets Marketing", New York, Dorset House Publishing, 2005.

[4] Karlsson J, "A Cost-Value Approach for Prioritizing Requirements", Norges Teknisk Naturvitenskapelige Uni. 1997

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 systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

2.1 Product perspective

This system is a new and very user friendly system. It is totally different from some other finding and searching service systems. User friendly and simplified system environment is the major concern for this system. It is not complicated or complex to understand.

This system will consist of four parts :

- i. Mobile Application*
- ii. Database Server*
- iii. XMPP Server*
- iv. Phone Call Device*

Mobile Application : The mobile application will be used to get and modify information such as find and view information of other users. It will provide a nice and environmental user interface. This application will need to communicate with Messaging and Phone call applications system. As well as it will directly communicate with Database server from where it will collect the information to the user interface and show to the user.

Database Server : Since this is a data-centric product it will need somewhere to store the data. For that, a database will be used. Mobile Application will communicate with database. It will use the database in two different ways one is for store data which is provided by the users and one is to get the data.

XMPP Server : As this product is going to provide messaging services it will need Extensible Messaging and Presence Protocol (XMPP). It enables the near-real-time exchange of structured yet extensible data between any two or more network entities. Therefore two or more users can intercommunicate between themselves via messaging. Unlike most instant messaging protocols, XMPP is defined in an open standard and uses an open systems approach of development and application, by which anyone may implement an XMPP service and interoperate with other organizations' implementations.

Phone Call Device : This product will also include phone call service. It is not online phone call service. Rather it is normal phone call which is possible between two users if they have each other phone number.

The mobile application has some restrictions about the resource allocation. To avoid problems with overloading the operating system the application is only allowed to use 20 megabytes of memory while running the application. The maximum amount of hard drive space is also 20 megabytes.

2.2 Product Features

With the mobile application, the users will be able to see which of their friends or other users are available during class breaks. This feature is the major priority of this project. Even user can search for other users.

After signing up and creating user profile the users will be accessed directly to the home feed. From here they will be able to see their friends list with their current status according to the majority of the interactions. There is another interface which will show the daily class schedules of the user.

For communications, there are two functionality which are messaging and direct phone calls. Also one can send friend request and can accept or remove friend requests. All these will be notified in the notification interface.

Details will be provided later on in the System Features section.

2.3 User Classes and Characteristics

There will be only one user class who is basically the user of this mobile application system. There are no other user classes exists in this system.

The mobile application user can only use the application to find other users. We know that the major goal is to find other students who are available. So this user can find other users who are free, they can set their class schedules, modify information, search for friends, communicate with others and so on. So, we can see that there are only one user class who can access and use all the functionality and features of this system.

2.4 Operating Environment

This system will only run on Android operating system. Latest every version of Android OS can use this application. For better and efficient use and support we are omitting the older versions of Android.

OS	-	Version
Ice Cream Sandwich	-	4.0-4.0.4
Jelly Bean	-	4.1-4.3.1
Kitkat	-	4.4-4.4.4
Lollipop	-	5.0-5.1.1
Marshmallow	-	6.0-6.0.1
Nougat	-	7.0-7.1

2.5 Design and Implementation Constraints

Co-1: Only any Running Student of University , School or Colleges will be able to use the Application due to verification Process of administration.

Co-2: This Application will be firstly available for Android And later on IOS. Any android phone supporting Jellybean will be able to run the application.

Co-3: The database system of this application will be based on Oracle Database.

Co-4: This application will only available in English Language .

Co-5: Application itself will ensure the total security of user data .

Co-6: Coding of the application will have to main the java documentation standards and conventions.

2.6 User Documentation

UD-1: The first time a new user accesses the system and on user demand the system shall provide an demo of the application, to allow users easily get into the application.

UD-2: The application will also provide an user-manual and tutorial video in it.

UD-3: There will be option to send any complain or suggestion about user experience for improving the application.

2.7 Assumptions and Dependencies

DE-1: The application is online based as it will be connected to database for checking the friends available , checking schedules and their location .

DE-2: Sign Up of a user will depend on the administration panel of the application.

AS-1: This application will be available to user 24/7.

AS-2: It will be a social network for university students.

3. System Features

This section will give an overview of every features in this system. All the functionalities and every single details will be considered here.

3.1 System Feature 1

TITLE : Sign Up

DESC : A user need to sign up in the software to get all the facilities. So the user must give user name, email and password to create account.

RAT : In order to create own account.

3.2 System Feature 2

TITLE : Login

DESC : Given that a user is registered he/she can login or get access to the server using email and password or directly using Facebook, Google plus.

RAT : In order to get access in the software.

3.3 System Feature 3

TITLE : Create Profile

DESC : If anyone creates profile using Facebook or Google plus he/she would not need to set the name.

User can upload own photo. Also add his phone number, institution and city.

RAT : In order to give information to the profile.

3.4 System Feature 4

TITLE : Edit Profile

DESC : If the user want to make changes than the user can do so. Edit profile allows to change picture, number, name, institutions.

RAT : in order to update or just change.

3.5 System Feature 5

TITLE : Time Scheduling

DESC : This allows user to set weekdays, time for individual classes also user can set campus. Not only these but user can set course name and add extra description.

RAT: In order to find friends in their free time this is the most vital feature.

3.6 System Feature 6

TITLE : Change Class Status

DESC : User can update the status for the class, quiz and term exams.

RAT : To let other know what is going on at the time.

3.7 System Feature 7

TITLE : View Timeline

DESC : This feature will show the timeline of user. Activities will be shown according to the time on the timeline.

RAT : To see own activities or others' activities.

3.8 System Feature 8

TITLE : Search Friends

DESC : User can search friends using the name, phone number, or university name. This feature will help for collaborations.

RAT : To search for friends and communicate

3.9 System Feature 9

TITLE : Send Friend Request

DESC : After finding friend user can send add request to add him/her to own friend's list. This will allow many more features after added.

RAT : To set someone to own list.

3.10 System Feature 10

TITLE : Messaging

DESC : User can send messages to friends to communicate directly.

RAT : To communicate via text.

3.11 System Feature 11

TITLE : Phone Call

DESC : There is also special feature that is phone call. User can call friends.

RAT : To communicate via voice call.

3.12 System Feature 12

TITLE : Notifications

DESC : User will be getting notifications if he/she gets messages or friends request. This will let user know if someone is trying to reach.

RAT : To communicate

3.13 System Feature 13

TITLE : Show Friends List

DESC : In home feed this feature will show which friends are free at the time. Also whether they have class or not.

RAT : To see who is available.

3.14 System Feature 14

TITLE : Report

DESC : User will be able to report fake/abusive account, picture, mobile number, spam or any other invalid contents.

RAT : To validate functionalities and to keep good working environment.

3.15 System Feature 15

TITLE : Configure Settings

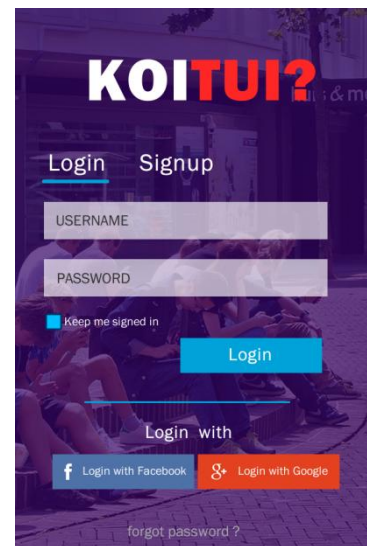
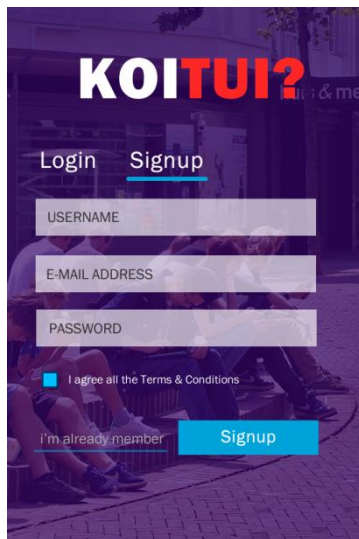
DESC : Setting will provide user various notification sounds to choose from. The availability of vibration and log out option.

RAT : To set the app to one's accord.

4. External Interface Requirements

User Interfaces will be shown and details will be discussed here. Also there will be a workflow attached later on.

4.1 User Interfaces



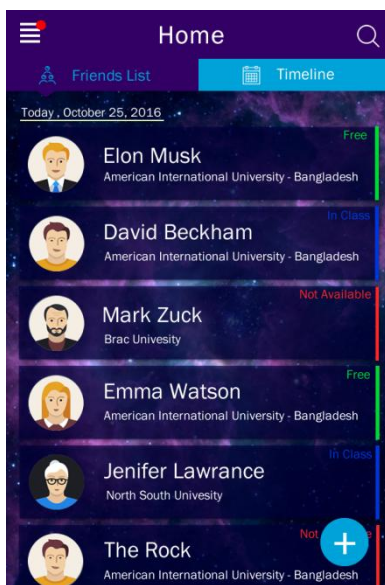
This is the starting page or the opening of the application. Here user will have option for login or signup as a new user.

Login: If user is not new user will have option for signing in. User may log into the app with KOITUI's Username and Password. Or else user may also login with Facebook or Google+ .

Signup: This option is for the new user. Here user will have to give a username , a mail address and a password. The id and mail will have to be unique . Multiple account with a mail will not be possible. User will have a option to agree with the Terms & Conditions. And at last a button for confirm the signup.

Forgot Password: Here user will have an option to recover the id in situation of forgetting the password.

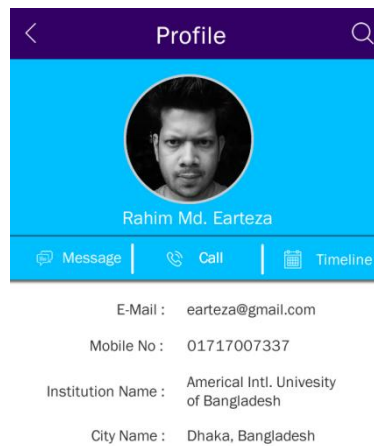
Create Profile: Here is the Profile Menu. Here user can change user's profile picture .Will also can edit certain information and change password from the menu.



Home: This is the home of the application. There is two tabs one is **Friends list** and one is **Timeline** .

Friend list : Here application will show all friend's of the user and their current status .Current status can be busy, in class , available , not available or free.

Timeline: Timeline will Show the users information. User's daily class routine. User will be able to add remove from here. User will also able to add new quiz , presentation and exam schedule's and set their timing's.



[Report user](#)

Profile Other user: This menu will show the Profile picture uploaded by the user. There is on screen tap option for message or calling the user. Also may tap Timeline for viewing other user's timeline. Underneath that will show the mail , mobile number , institute name , city name of the other user according to his privacy.



Add Class

Course Name

Course Description

Faculty Name

Week days

SAT SUN MON TUE WED THU FRI

Class Duration

Start

00.00 AM

to

END

00.00 AM

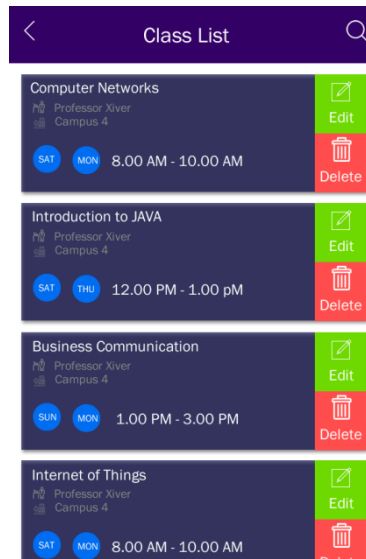
Campus Name

Room No.

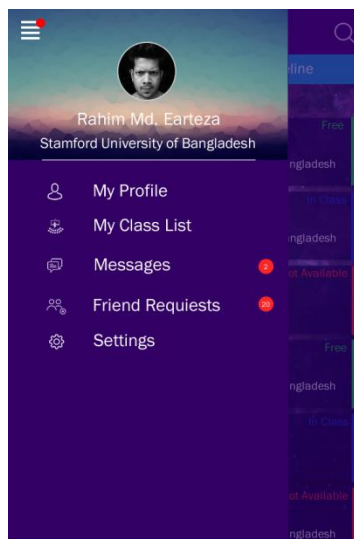
ADD ANOTHER

PUBLISH

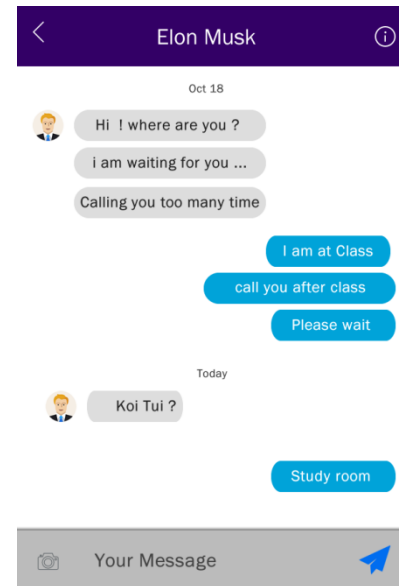
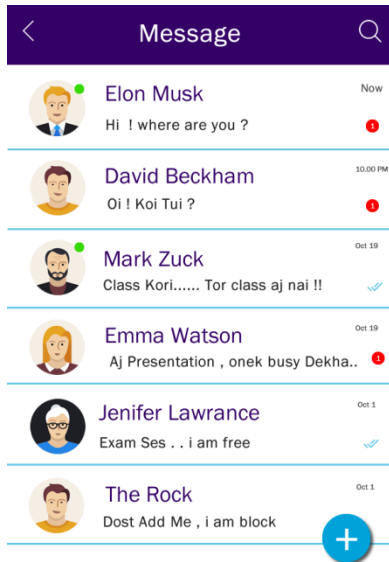
Add Class: Here user will be able to add new class schedules and information's . Class schedule will consists information about description ,faculty name , weekdays , class duration , building no , room no etc. User will have to write the data's on the following interface and press publish to save it.



Class list: Class list interface menu will show all class schedule's and class information's of a user. User will be able to add new class or delete previous class's from this menu. There is also a search option for searching for friend's.



Side Menu: Like all modern app menu's "KOITUI" also has a side menu. Here user will have a shortcut of all menus. Here notifications of friend requests and message will be shown. User can also get into settings.

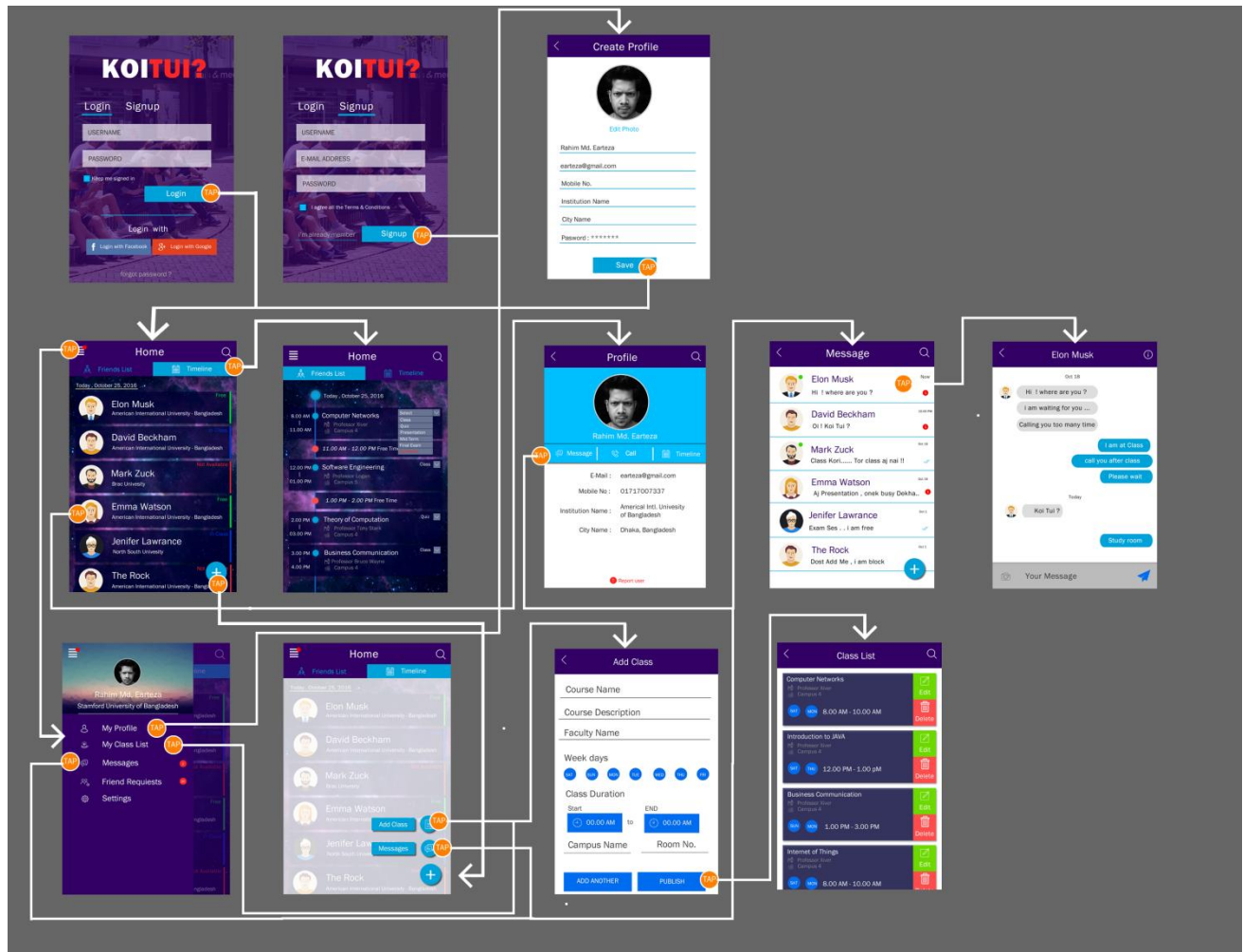


Message : In this menu user will be able to send a new message to a friend. Will also see all conversation with his friends. Unseen message will also show notification.

Message Chat Box: User will be able to chat in this menu. User will type messages and sending them .User will also have option to attach files and pictures. User can press the i(info) button to see profile information about other user.

4.2 User Interface Work Flow

Here is showing the work flow of the whole system. Showing the interactions between different interfaces according to the user activities.



5. Other Nonfunctional Requirements

Here all the other non-functional requirements performance , safety and security will be discussed. Also a short software quality description will be attached in the end.

5.1 Performance Requirements :

- Pe-1:** The system and server of the application will be able to maintain 50 thousand users after the launch of the application.
- Pe-2:** Responses to queries shall take no longer then 6-7 seconds on backhand.
- Pe-3:** The system will display the loading status of activity during action performed by the user.
- Pe-4:** Pages of the application shall be well designed and have to maintain the efficiency .
- Pe-5:** Primary Performance requirement of the application is speed of the Internet network of Mobile Phone .
- Pe-6:** Application may get a bit slow if many actions are performed by multiple users at the same time.

5.2 Safety Requirement

There are no safety requirements with this application except to create any fake id and not to disturb or harass any unknown person otherwise actions might be taken according to the policy and limitations of the application. other than any normal hazards of a mobile device. The only hazard is a user using the device when they should not be, such as while driving.

5.3 Security Requirements

1. All information's of a user data shall be encrypted and saved.
2. User will have to log in to application to perform any activity .
3. User will get the custom privacy experience same as "Facebook" like adding friend, blocking someone , selecting restricted friends.

4. Log in from multiple devices will not be possible. Users will only be able to use the application from a single device at a time.
5. All calls between users will be deleted from the server. Only log will be saved.

5.4 Software Quality Attributes

'**Koitui**' is a phone application, as it is online based , performance may fluctuate depending upon internet speed and mobile device. Portability on reliability of the application will be strongly maintained by the developer team. The application will not take much space or to make slow the phone system.

'**Koitui**' will be a media of communication for students .It will maintain social communication, information about friends . Finding someone will be real easy with the application.

This application is android based. This is in contrast to other phones .However, in the future, the ability to use this on other phones that support the goals of the project would be nice, but that is also outside of the scope of this project.

6. Other Requirements

Analysis Models

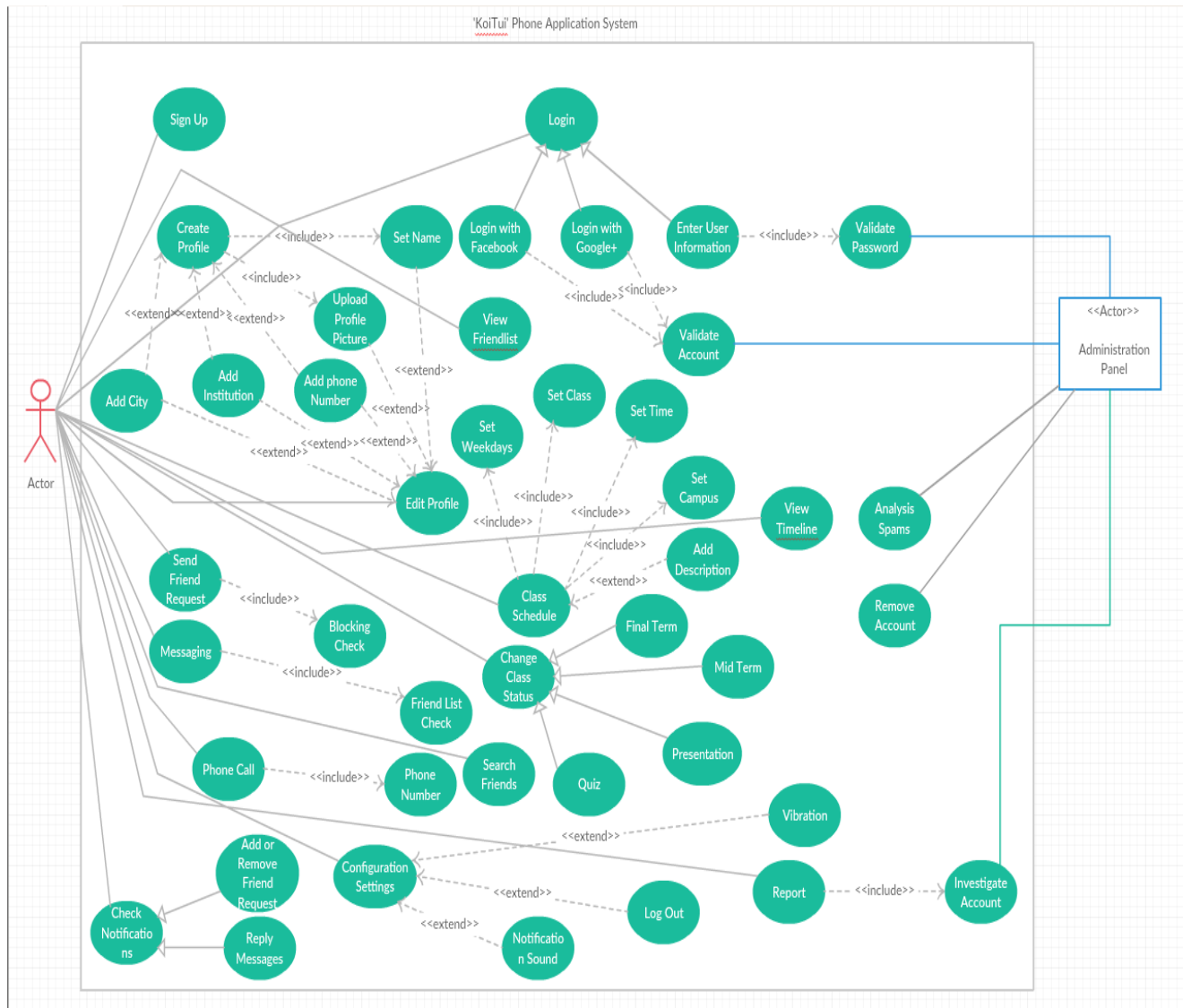


Figure : 'KoiTui' Phone Application System (Use Case Diagram)

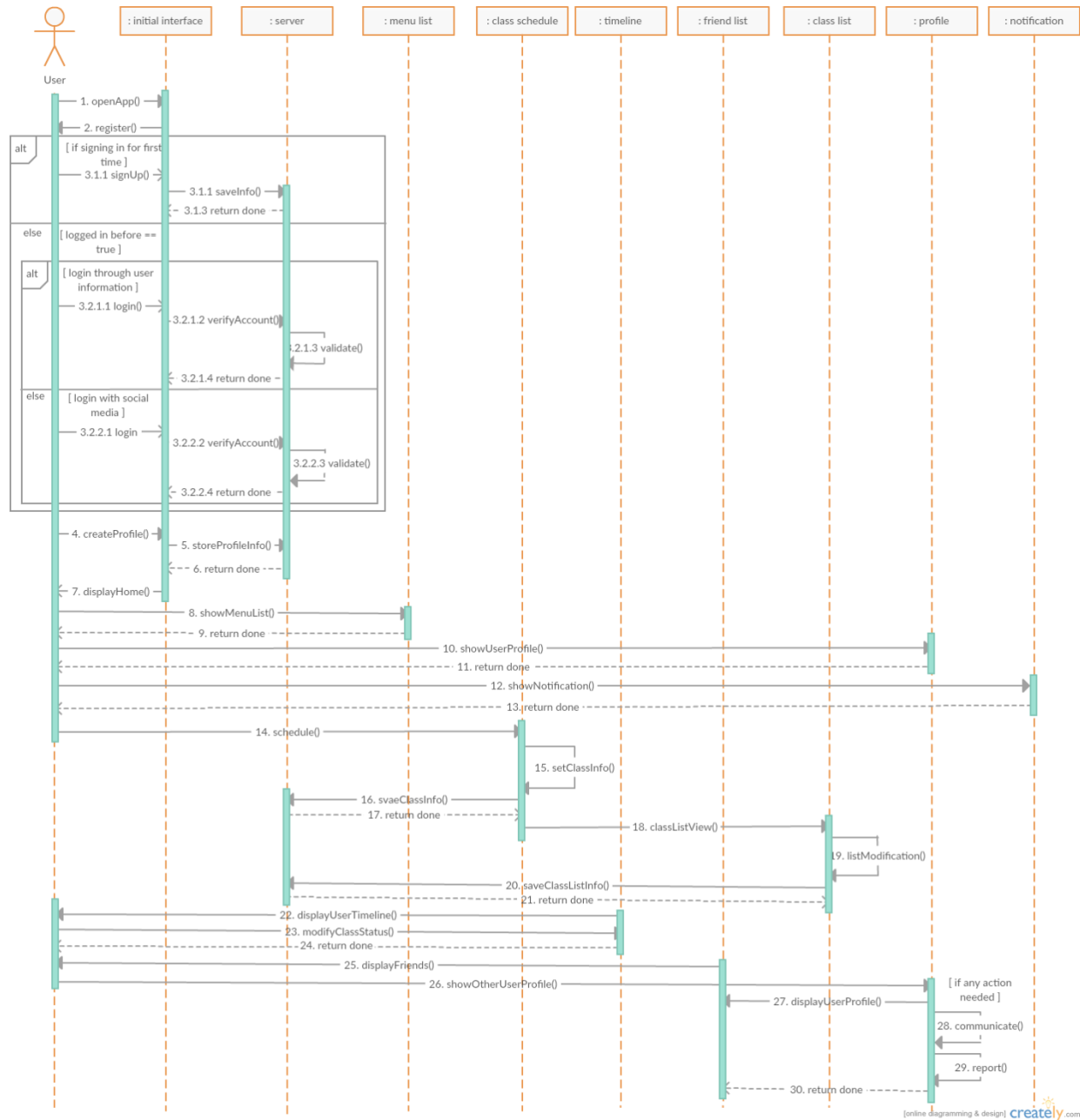


Figure : 'KoiTui' Phone Application System (Sequence Diagram)