PROFESSIONAL TRAINING REPORT

at

Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Technology Degree in

Information Technology

By

Md Arsalan Abid (Reg. No. 3612115)

DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMIL NADU.

JUNE 2018

_____

SCHOOL OF COMPUTING

<u>BONAFIDE CERTIFICATE</u>

This is to certify that this Professional Training Report is the bonafide work of Md Arsalan Abid (Reg. No. 3612115) who underwent the understanding of the back-end workings through "Cisco Networking Academy**"** under the supervision of Sathyabama Institute Of Science and Technology from June 2017 to May 2018.

Internal Guide

MS.Vimali.J.S, M.Tech.

Head of the Department

Dr. R.SUBHASHINI, M.E., Ph.D.

_____

Submitted for Viva voce Examination held on_____

Internal Examiner   **External Examiner**

DECLARATION

I, Md Arsalan Abid (Reg. No. 3612115) hereby declare that the Professional Training Report on "**Notification Chat App"** done by me under the guidance of <u>Ms.Vimali.J.S, M.Tech,</u> at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE: **SIGNATURE OF THE CANDIDATE**

<u>ABSTRACT</u>

Due to the absence of any efficient method to provide notices and circulars to the thousands of faculties; students and staffs alike, I propose a project which will send out such announcements without the need for manual labor, through android based application.

This application will work in two phases. The first through which the faculties would be able to issue the notice, this would be secured via an encrypted database so as to prevent any unauthorized misuse against the institution. The second phase will involve bringing the information sent by the institution to the recipients.

Through the use of applications such as these our institution can not only reduce the amount of manpower it has to spend on everyday activities like manually handling out notices, but also provide a quick, efficient and reliable way of dispatching news.

ACKNOWLEDGMENT

First and foremost, I would express our sincere gratitude to our beloved Founder Chancellor **Col. Dr. JEPPIAAR, M.A., B.L., Ph.D.,** and Chancellor **Tmt. Remibai Jeppiaar.** I extend my sincere thanks to our Pro Chancellor **Dr. Mariazeena Johnson, B.E., M.B.A., M.Phil., Ph.D.,** and the Vice President **Dr. Marie Johnson, B.E., M.B.A., M.Phil., Ph.D.,** and for providing me the necessary facilities for the completion of the professional training. I also acknowledge our Vice Chancellor **Dr. S. Sundar Manoharan Ph.D.,** and the Pro Vice Chancellor **Dr. T. Sasipraba, M.E., Ph.D.,** for their constant support and endorsement. I would like to express my gratitude to our Registrar **Dr. S. S. Rau, Ph.D.** and Controller of Examinations **Dr. Igni Sabasti Prabu M.E., Ph.D.,** for their valuable support offered to complete my professional training successfully.

I like to express my gratitude to **Dr. R. Subhashini**, **M.E., Ph.D.,** Head of the Department of Information Technology, Sathyabama Institute of technology for having been a constant source of support and encouragement for the completion of the professional training.

I would also like to express my sincere thanks to our internal guide **Ms.Vimali.J.S., M.Tech** who guided me in the preparation of the report.

Also, I thank the **Almighty** and my **Parents** for supporting me in the completion of the professional Training.

Contents

7. User Interface and Workings

8. Future Scope and Conclusion

9. References

CHAPTER 1

INTRODUCTION

This app helps you access online notices on your phone. It is an online board where a group of people can easily communicate with each other. The board is a place where staff/students gathers to get their latest release of corporate news. IT brings the notice board to a virtual location where staff/students can not only read notices, but immediately react and respond to them - from their own mobiles. With this electronic notice and announcement system, notification alerts may be sent out notifying staff and students that a new notice has been posted. This eliminates the need to keep a manual system which is hard to maintain due to the rapid movement of staff.

The interface of this application is straightforward and takes you roughly a minute to get started. Sending messages to particular person or a group is easy, just click on the tab and enter the text. Users can view the post on
the spot because of real time database management. Here registration is must for all the users having this application in order to send and receive posts.

CHAPTER 2

LITERATURE SURVEY

2.1. Android Web Apps

Developing applications for Android and other mobile devices using web technologies is now well within reach. When the capabilities of HTML5 are combined with CSS3 and JavaScript, web application developers have an opportunity to develop compelling mobile applications using familiar tools. Not only is it possible to build mobile web apps that feel as good as native apps, but to also write an application once and have it run a variety of different devices. While the HTML5 specification is still evolving, there is a lot that can be used right now to build mobile web apps. Mobile web apps are now starting to provide many of the features that were once only available to native-language-based apps in Java, Objective-C, etc.

2.2. Experience of using Chats for educational purposes

Chat is still very much in its infancy and it has been used mainly for social purposes by the young. However, there is a slowly growing literature reporting on teachers' experiences with CHAT for educational purposes and this section reports on some of these. Inevitably the teachers' experience is linked to very specific contexts, though this provides insights that may be generalized for other disciplines. Language teaching is an area where there has already been considerable interest in CHAT. Mark Freiermuth (2002) found it (perhaps surprisingly) useful in his work with students of English as a foreign language. Although he set up his lessons in a way that might seem rather artificial – students in a normal classroom 'talking' to each other through the medium of CHAT - he found that it had the following advantages:

- The teacher can monitor all the groups in a way that is not possible when students are discussing normally in small groups.

- The teacher can monitor and guide when appropriate in a relatively inconspicuous way.

- As a can be kept of all the conversations it is easy for the teacher to do follow-up work focusing on the language that was produced by the students.

- It is easier for all students to contribute to the conversation as they can go back over other's contributions if they do not understand them on first reading.

- The ability to see the conversation helps to keep students engaged, focused and contributing.

2.3. Problems in using such apps

The key problems referred to time and again in the literature relate to technology as such, especially the need for reliable technology. A CHAT session will easily be spoiled if the software does not work or if any of the participant has connectivity problems.

Some people also have problems with the abbreviated, highly colloquial kind of language used in CHAT sessions but this is not a universal concern. As a language teacher Freiermuth expressed concern about the quality of students' language and warns teachers to try to curb excessive use of abbreviations. (Freiermuth 2002:40)

Freirmuth's concerns about the abbreviated features of language used in CHAT sessions are not shared by Merchant (2001). He argues quite convincingly that new

and fast forms of written communication are being developed and that those who are comfortable with using these forms will be at an advantage. He writes: 'Derrida, in his deconstruciton of Plato's Phaedrus, uncovers a telling ambiguity in the myth of Thamus and Theuth, arguing that Plato sees the invention of writing, like the discovery of a new drug, as both 'poison' and 'medicine'. If we extend the analogy to the new forms of communication … we might ask whether these new electronic forms are really dangerously addictive and corrupting, or whether they are innovations which open new vistas of possibility.' (Merchant 2000: 305)

CHAPTER 3

SYSTEM ANALYSIS

3.1. Objective

The proposed system's objectives are to overcome all the limitations and drawbacks of the existing system. This application is user-friendly android application. The main objective of the application is its simplicity of design and ease of implementation that shows and helps to collect most of the information and relay it instantly. The interface will be very user-friendly.

The main objectives of the proposed system can be enumerated as follows:

• Faster dissemination of notices regarding education, technical events, cultural events.

• Easy way to broadcast your message.

• Helps you to be updated with whats going on in College.

• User can also create groups and interact with multiple persons at the same time.

3.2. Identification of Need

1. As we discussed earlier that manual maintenance of a notices is a tedious job. So to enhance the ease of working, we go for this package.

2. Giving the facility to convey messages to all students anytime and anywhere.

3. Making students updated about all the events and activities going on in the college.

4. The student will not require to stand in the crowd to see the notice. There will be no issue of fighting in order to see the notice first. Everyone will be able to see that notice inside their own mobile phone anywhere and anytime.

5. The least but most important it saves time.

6. Utilizing less man power. As there are many persons involved in circulating the message. With this application, only one person is required to post the notice. Rest of the man power is saved in the entire process.

3.3. Existing System

Currently our college has manual system of putting notices on notice board. Its outdated now. As nobody has a time to stand in rush in order to read the notices on noticeboard.

Limitations of Existing System:

1. *Order of Data:* Notice can get out of order in traditional notice board system. If someone accidentally puts some data in the wrong place, it can lead to lost data. Automated notice management systems allow users to quickly check whether information already exists somewhere in the system,

which helps avoid problems like redundant data.

2. *Complexity:* Automated system is less complex than manual system of handling notices, which can make it easier for untrained people to access and manipulate data. Anyone having the basic knowledge of mobiles can work on the automated system.

3. *Inconsistency of data:* There will be an unavailability for future user, since notice might get misplaced during manual notices management. So notice won't be preserved properly for future use.

4. *Damage:* Manual notices stack are vulnerable to damage, destruction and theft in ways that digital databases are not while a digital database will typically allow users to search the entire database for specific information in seconds, someone looking for information in a manual system may have to spend hours searching for a particular piece of data.

3.4. Proposed System

Proposed System will be able to do the following:

1. *To eliminate wastage of time and energy:* e-Notice app will be able to save lot of paper and time. It directs both teacher and pupils energy and attention to one thing at a time by placing proper persons at their proper places at the proper time. Everything will be instantaneous.

2. *To bring system into college life:* It would be dire need of all colleges as its easy and shortcut method to inform all the students. In the absence of proper notification system will make it very difficult to inform students at right time.

3. *Free Service:* It gives free service to notify all the students. There will be no cost of sending notification at all. Just have the good system implemented in college and that too free of cost.

4. *Prevent Crowd in College:* As you can see, there is always a crowd at notice board. As notice board is one, and people to see notice are more. With this application there will be no more crowd. Everyone will be well informed even at their homes. So they are free to focus on their other works.

5. *Automatically Updated Dashboard:* The dashboard of notice is automatically updated when a new message arrives. The user can himself refresh the dashboard to see any new notice.

6. *Anytime Anywhere Service:* With this application, notices will be delivered anytime and at any place. There is no restriction of time to send a notice.

7. *Reduce in Manpower:* With this application, huge resources will no longer be needed to manually distribute notice.

3.5. Unique Features of this System

The unique features of this application are as follow:

1. *Firebase database:* It will be used to broadcast notices to all the students who are registered with this application. Also, it allows you to attach up to 1,000 recipients to a single message, letting you easily contact large user bases quickly when appropriate, while minimizing the work load on your server.

2. *Battery Saving Application:* This application saves your battery too. Its because, the service implemented in application is not running all the time. Whenever the database pings the mobile, only then it makes a broadcast to phone that initiates the service. In this way, its saving your battery a lot.

3. *Automatically Updated Dashboard:* The dashboard of notice is automatically updated when a new message arrives. The user can himself refresh the dashboard to see any new notice.

4. *Anytime Anywhere Service:* With this application, notices will be delivered anytime and at any place without any restriction of time.

5. *Keeping Notices at one place:* This application allow you to have notices in one place only so there will be no here and there of notices.

6. *Free Service:* It gives free service to notify all the students. There will be no cost of sending notification to all. Just have the good system implemented in college and that too free of cost.

CHAPTER 4

SYSTEM REQUIREMENT SPECIFICATIONS

4.1. Data Requirements

Data requirement is meant to be the data that will be used in our application. Data required in this project is all notices, that need to be conveyed to the user. This application also require the username and passwords of persons in order to register them and sending notification about updates. So two main

requirements are:

• Notice Details

• User Details

4.2. Functional Requirements

In order to make this application functional, we require the following:

• *Download mobile application:* A user should be able to download the mobile an application through either an application store or

similar service on the mobile phone. The application should be free to download.

• *User registration:* 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 password and e-mail address.

• *User Login:* Given that a user has registered, then the user should be able to log in to the mobile application. The log-in information will be stored on the server and in the future the user should be logged in automatically.

• *Reset Password:* Given that a user has registered, then the user should be able to retrieve his/her password by e-mail.

4.3. 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.

• *Prominent search feature:* The search feature should be prominent and easy to find new user.

• *Response Time:* The response time should not be more than 5 seconds if user have a proper internet connection.

• *Fault Tolerance:* The fault tolerance of the system should be very good. If the system loses the connection to the Internet or the system gets some strange input, the user should be informed.

4.4. System Dependability

Following are the requirements that an application require from the device/mobile on which it is installed:

• *Internet Permission:* Application developed, require full internet permissions of mobile so that it can fetch notices from the server. At the same time.

• *System Tools:* This application require various system tools to be used. For example, it requires Gallery of mobile in order to get the image of your avatar.

• *Account Info:* It also fetches your registered account information in order to verify it and send temporary password in case of a forgotten password.

4.5. Maintainability Requirements

Following are the maintainability requirement of e-Notice mobile application:

• *Application extendibility:* The application should be easy to extend. The code should be written in a way that it favors implementation of new functions. It is requires in order for future functions to be integrated easily to the application.

• *Application testability:* Test environments should be built for the application to allow testing of the applications different functions.

4.6. Security Requirements

• *Communication Security:* There should be security of the communication between the system and server. The messages should be encrypted for log-in communications, so others cannot get user-name and password from those messages. Every exchanged of information between client and server should be encrypted so that no one can track it.

• *Account Security:* If an user tries to log in to the app with a non-existing account then the user should not be logged in and should be notified about log-in failure.

• *Account Security:* The security of account for users of the system should be maintained and the password encrypted at the back-end.

4.7. Look and Feel Requirements

Regarding look and feel, our client is straight forward. They believe in simplicity. So these are their requirements:

• *Simple and Light:* The user interface should be simple and lightly colored. It should give relaxing effect on looking at its GUI. No bright colors should be used while designing the UI of this application.

• *Easy to Use:* The application should be easy to use. If any user is doing something wrong, he/she should be informed correctly, what is going wrong behind the scene. There should be proper instructions for the user to use this application.

4.8. Validations

Any application is useless without validation. There should be a way to validate the user input first before sending the user request to the server. Following are the validations implemented in proposed system:

• *User Password Validation:* The application should check the user and password fields before sending any request to the server. It should check whether the fields are filled or not. if fields are not filled up, user should be instructed to fill up the fields before moving further. in this way, there will be less traffic on the server.

• *Validations During Registration:* There are a lot of validations that needs to be implemented in the application. They are as follow:

1. Email: The application must validate and email address entered by the user before sending request to the server.

2. Password: The password should contains one digit from 0-9, one   lowercase character,one uppercase character, one special symbols in the list "#$%" and length of password should be at least 6 characters and maximum of 20.

• *Validating During Posting:* The application should validate the fields before posting any notice. It should check whether fields are filled or not. if not, it should not sent anything.

• *Reset Password Validation:* The application should send the resetting password link on the registered email address.

4.9. Expected Hurdles

The main hurdles that can come in the notice are as follow:

1. *Post Notification:* Its possible that there may be a problem with receiving notifications from the database.

2. *Device Database:* Its very much difficult to see entries going inside the device database. You have to check through coding. You can't access the applications database easily. So its bit hard to debug the database errors.

4. *Google Play Services:* The application should use a concurrent version for all the dependencies used.

CHAPTER 5

SYSTEM DESIGN SPECIFICATIONS

5.1. Use Case Diagram
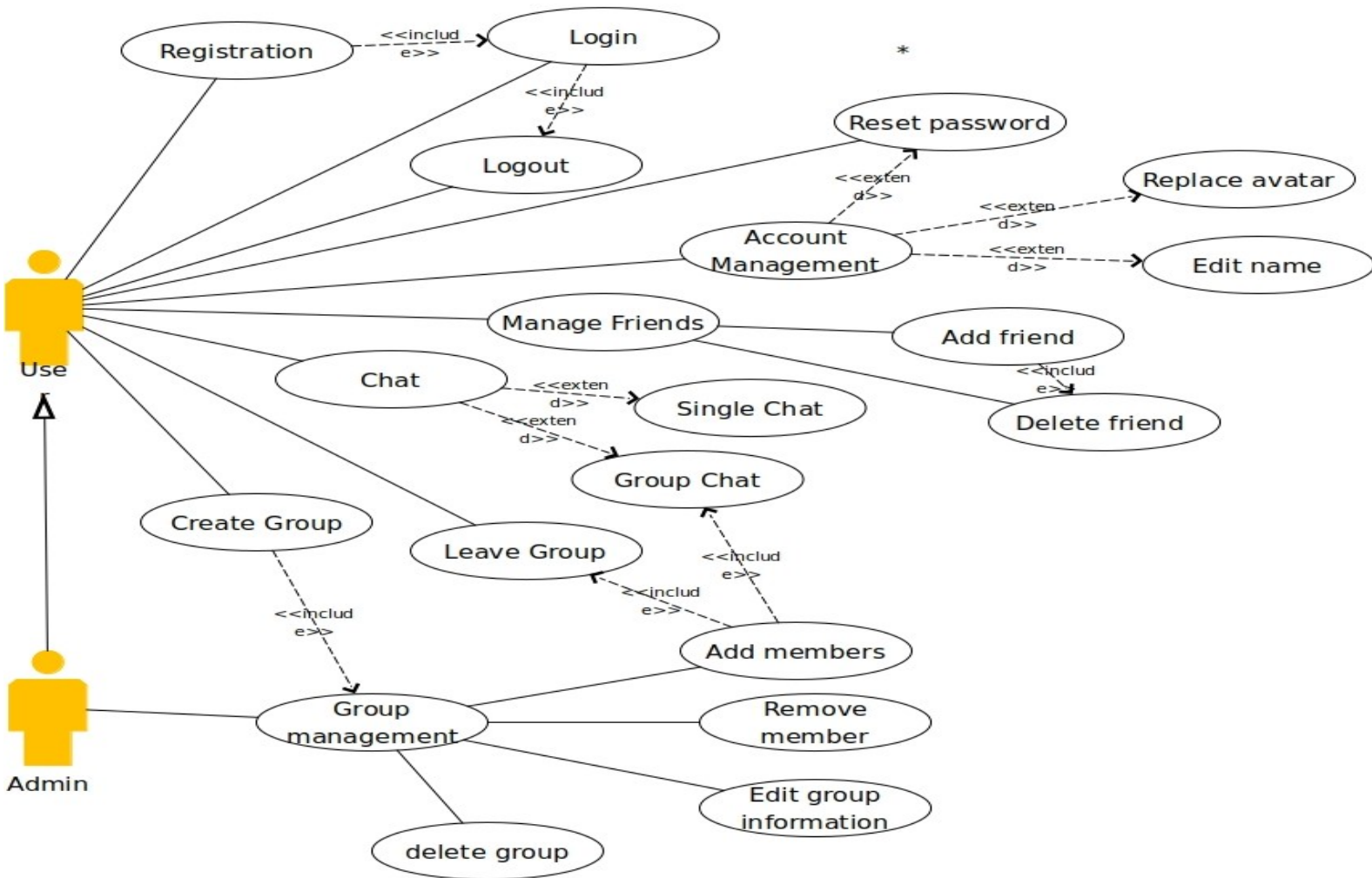
A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

5.2. Class Diagram

LoginActivity
-mAuth: FirebaseAuth
-user: FirebaseUser
-createUser()
-signIn()
-resetPassword()
-saveUserInfo()

Room
-member: List<String>
-groupInfo: Map<String, String>
+getter()
+setter()

ListFriend
-listFriend: List<Friend>
-memberName

Group
-id: String
-listFriend: ListFriend

FriendsFragment
-dataListFriend: ListFriend
-addFriend()
-getListFriend()

ListGroupAdapter
-listGroup: List<Group>
-listFriend: ListFriend
+setAdapter()
+notifyDataSetChanged()

ChatActivity
-messageAdapter: ListMessageAdapter
-sendMessage()
-receiverMessage()
-getLastMessage()

MainActivity
-mAuth: FirebaseAuth
-mAuthListener: AuthStateListener
-user: FirebaseUser
-initFirebase()
-initTab()
-setupViewPager()

GroupFragment
-listGroup: List<Group>
-adapter: ListGroupAdapter
-getListGroup()
-getGroupInfo()
-deleteGroup()

User
-name: String
- email: String
-avatar: String
-status: Status
-message: Message
+getter()
+setter()

ListMessageAdapter
-conversation: Conversation
-bitmapAvatar: Bitmap
+setAdapter()
+notifyDataSetChanged()
+onDataChange()

Configuration
-name: String
-email: String
- avatar: String
+getter()
+setter()

UserProfileFragment
-myAccount: User
-getUserInfo()

Friend
-id: String
-roomId: String
+getter()
+setter()

Conversation
-listMessateData: List<Message>
+getListMessageData()

Message
-idSender: String
-idReceiver: String
-text: String
-timestamp: String
+getter()
+setter()

UserInfoAdapter()
-listConfigInfo: Configuration
-setAdapter()
-notifyDataSetChanged()

A class diagram in the Unified Modeling Diagram(UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes their attributes, operations (or methods), and the relationships among objects.

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.
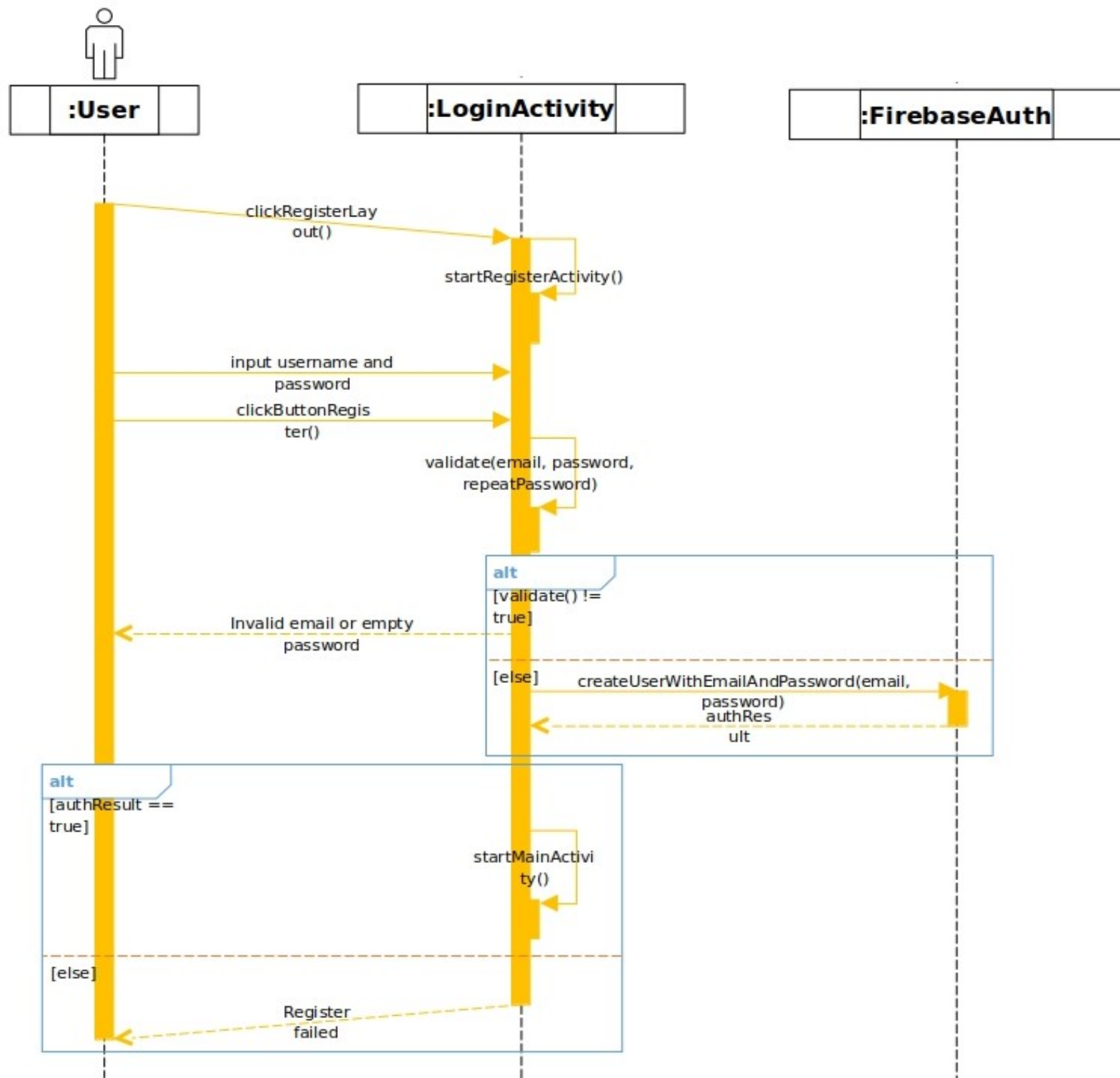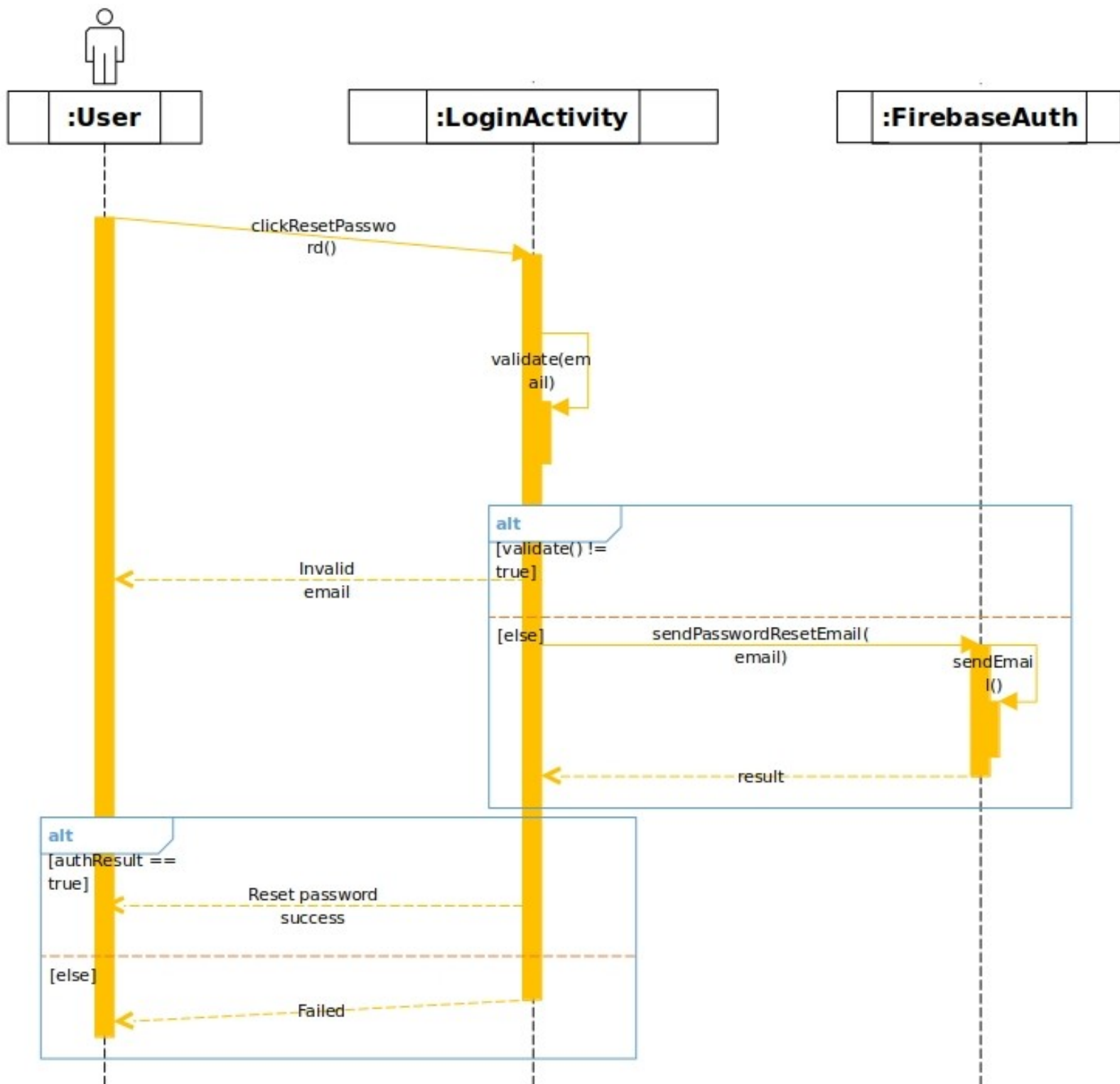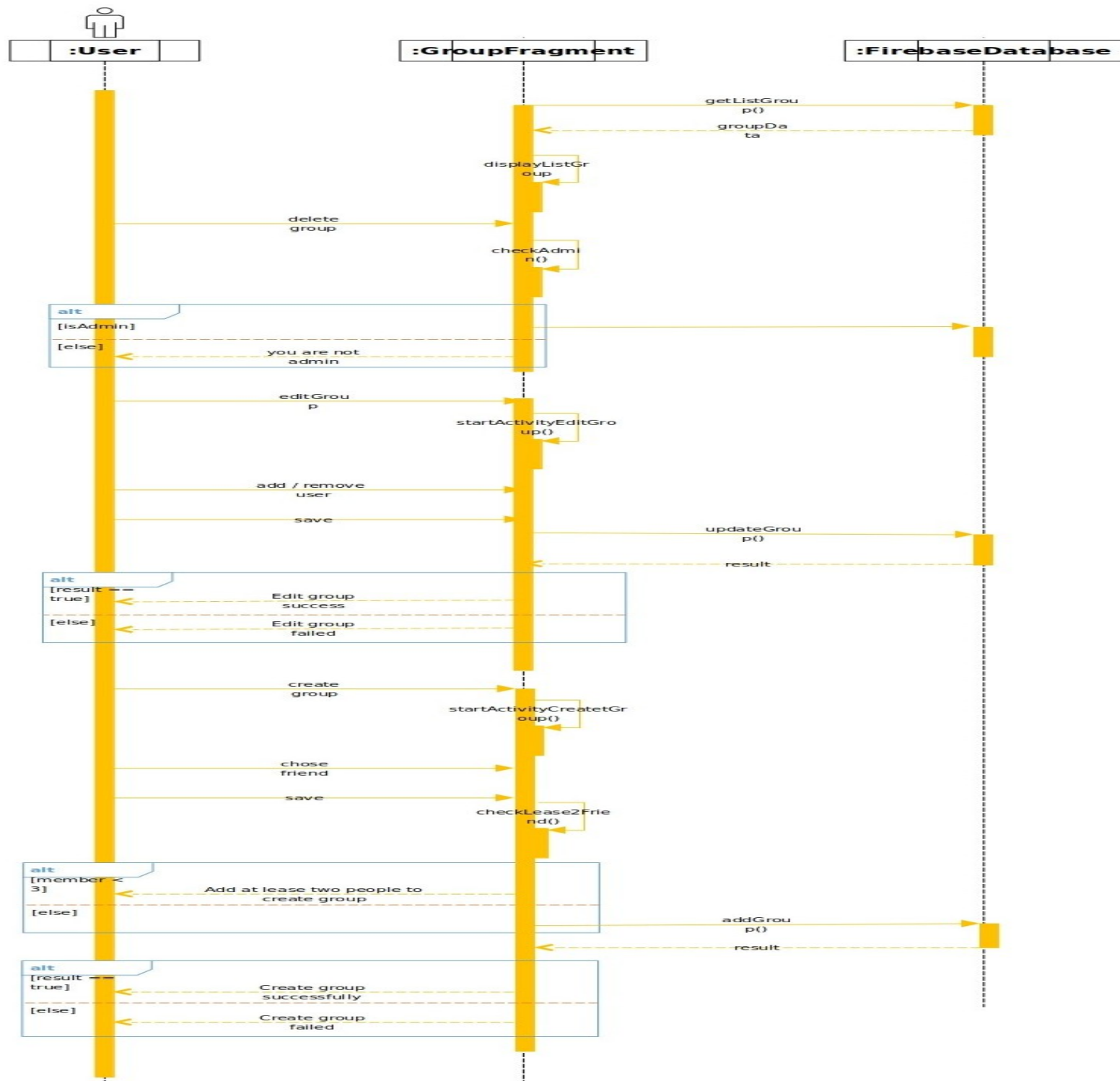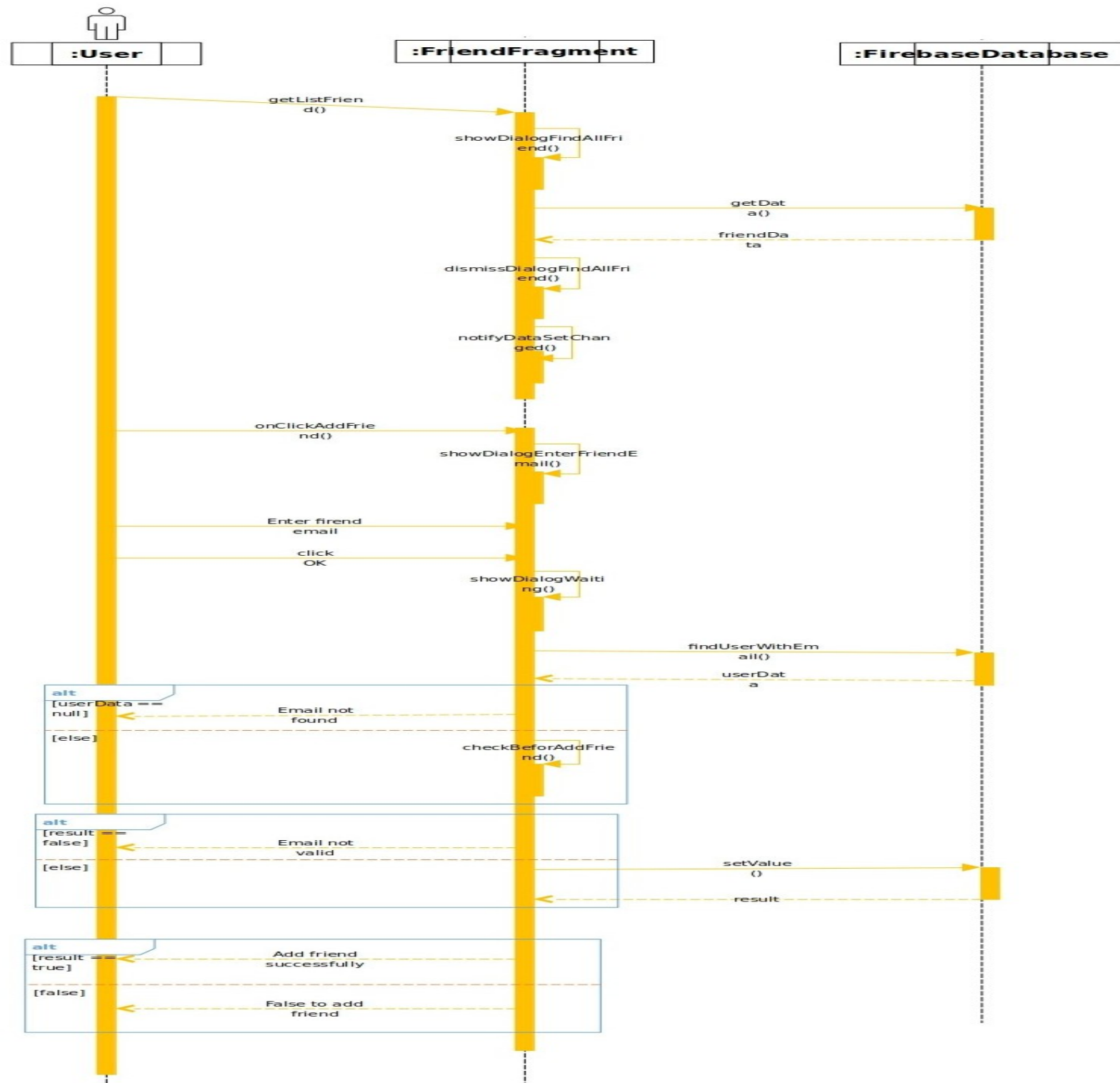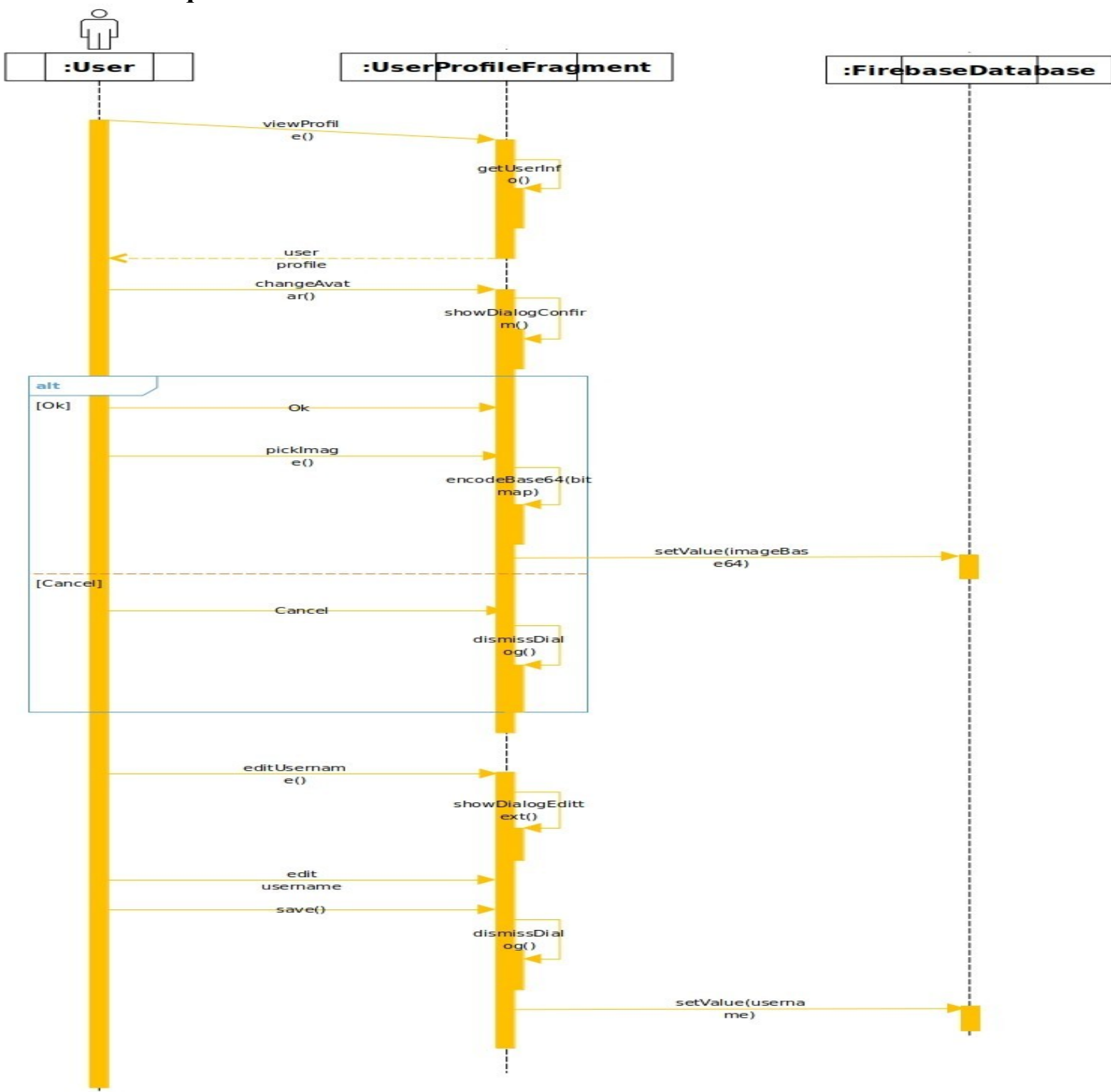
5.3. Sequence Chat
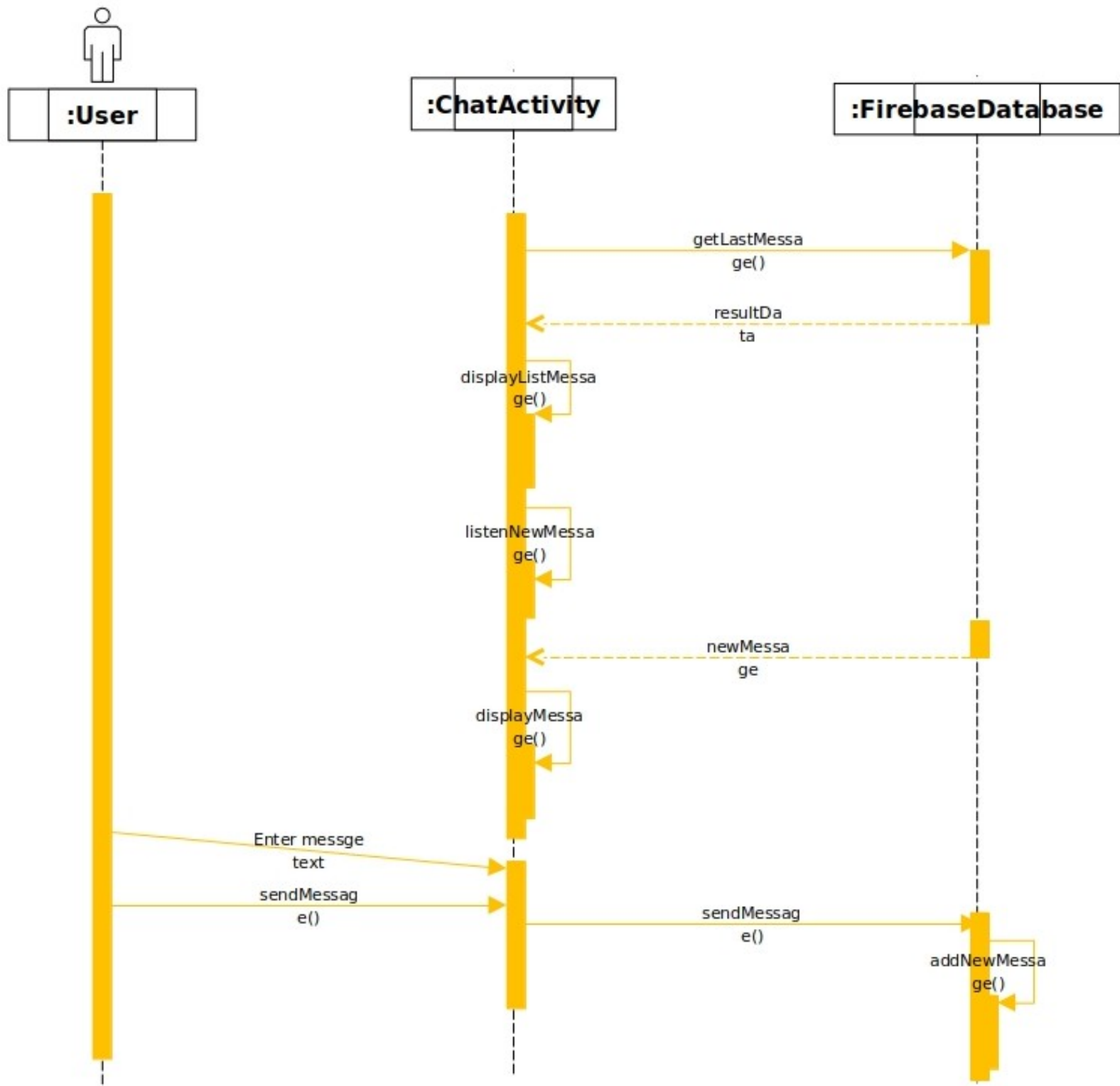
The above sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between two users during chat.

5.4. Sequence Login
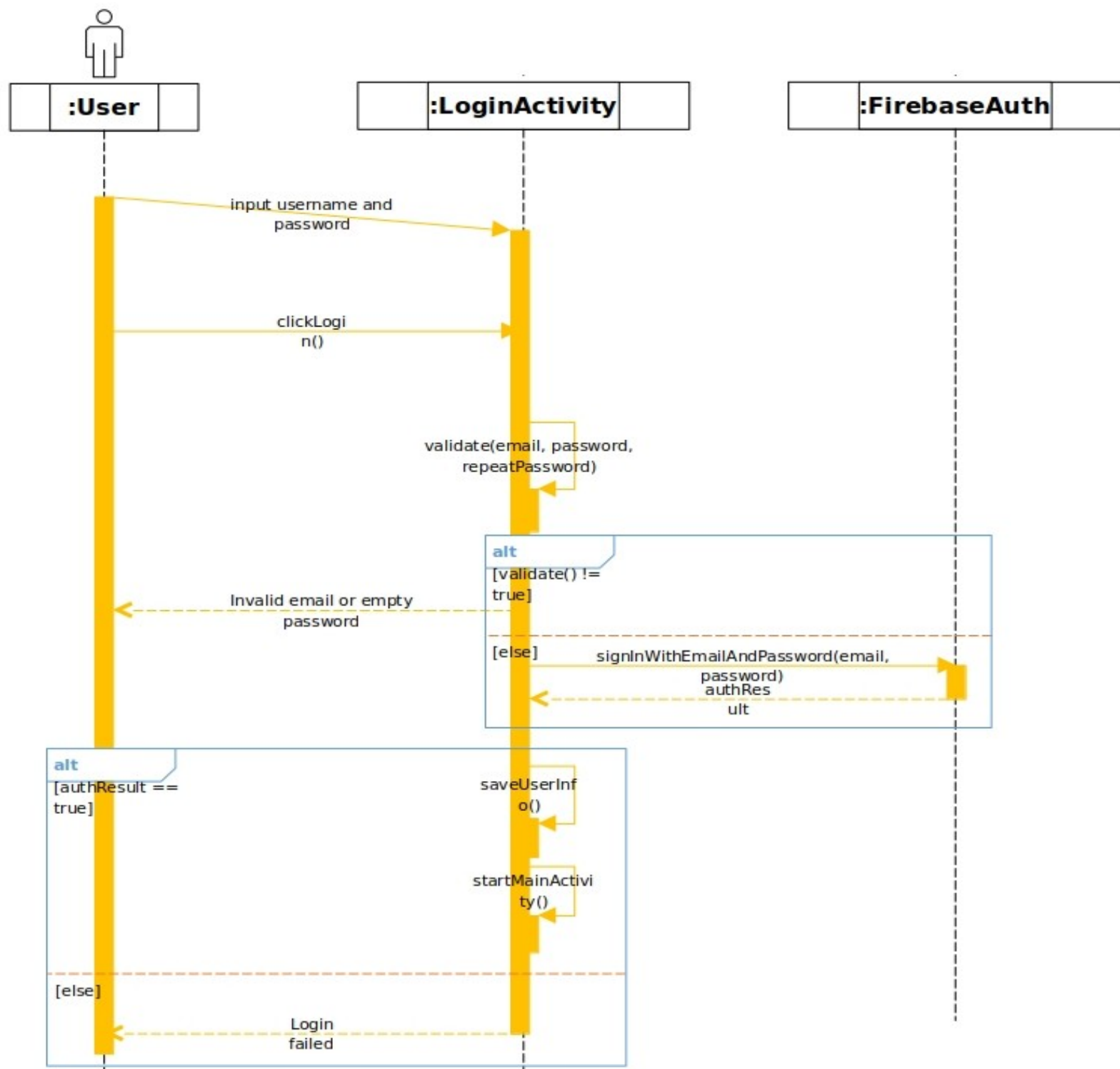
The above sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between two users during the log in of users.

5.5. Sequence Register

The above sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between two users during the registration of users.

5.6. Sequence Reset Password

The above sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between two users during the resetting of password done by the users.

5.7. Sequence Group

The above sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between two users during the construction and modifying of new or existing group.

5.8. Sequence List Friend

The above sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between two users during sequencing and listing of new and existing friends.

5.9. Sequence Profile

The above sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the user and the server to manipulate its own data.

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 Design Approach

A design approach is a general philosophy that may or may not include a guide for specific methods. Some are to guide the overall goal of the design. Other approaches are to guide the tendencies of the designer. A combination of approaches may be used if they don't conflict.

6.2 Function Oriented Design Approach

Function Oriented Design Approach is partitioning of a design into subsystems and modules, with each one handling one or more functions. Contrast with object-oriented design, data-structure-oriented design,

This application project uses function oriented design approach. Every module and sub modules are made, based on their functionality. These modules are designed and implemented separately and then they are integrated together to form the desired application.

6.3. Design Details

The detailed design of this application is as follow:

1. *Registering a User:* The first step in this application is to get the users registered to the back-end server. For this, user will provide all the necessary details and press the register button. The request will first go to Firebase database where the user gets registered.

2. *User Login:* After registering, the user is allowed to log in. Using the username and password which is sent at the server side for authentication. After authentication response is sent by the server to client, and then user gets logged in.

3. *Searching a friend/group:* The user is able to search a new friend which is added in the list view. Similarly new groups can be created with two or more users. It helps user to get the desired work done instantly.

4. *Deleting a friend/group:* If the user does not want to get contacted by someone they have the capability to delete the friend.

5. *Posting a message:* The user can send and receive messages both one on one as well as in a group.

6. *Resetting the Password:* This application also has the facility to reset the password. If one user has forgot his password, he/she can rest the password by going in the reset password option this will send the mail on a registered email address and going through the link the user can set a new password.

6.4 Modules Implemented

- Activity Friend Module

- Activity Group Manager Module

- Activity Login Module

- Activity Register Module

- Activity Reset Password Module

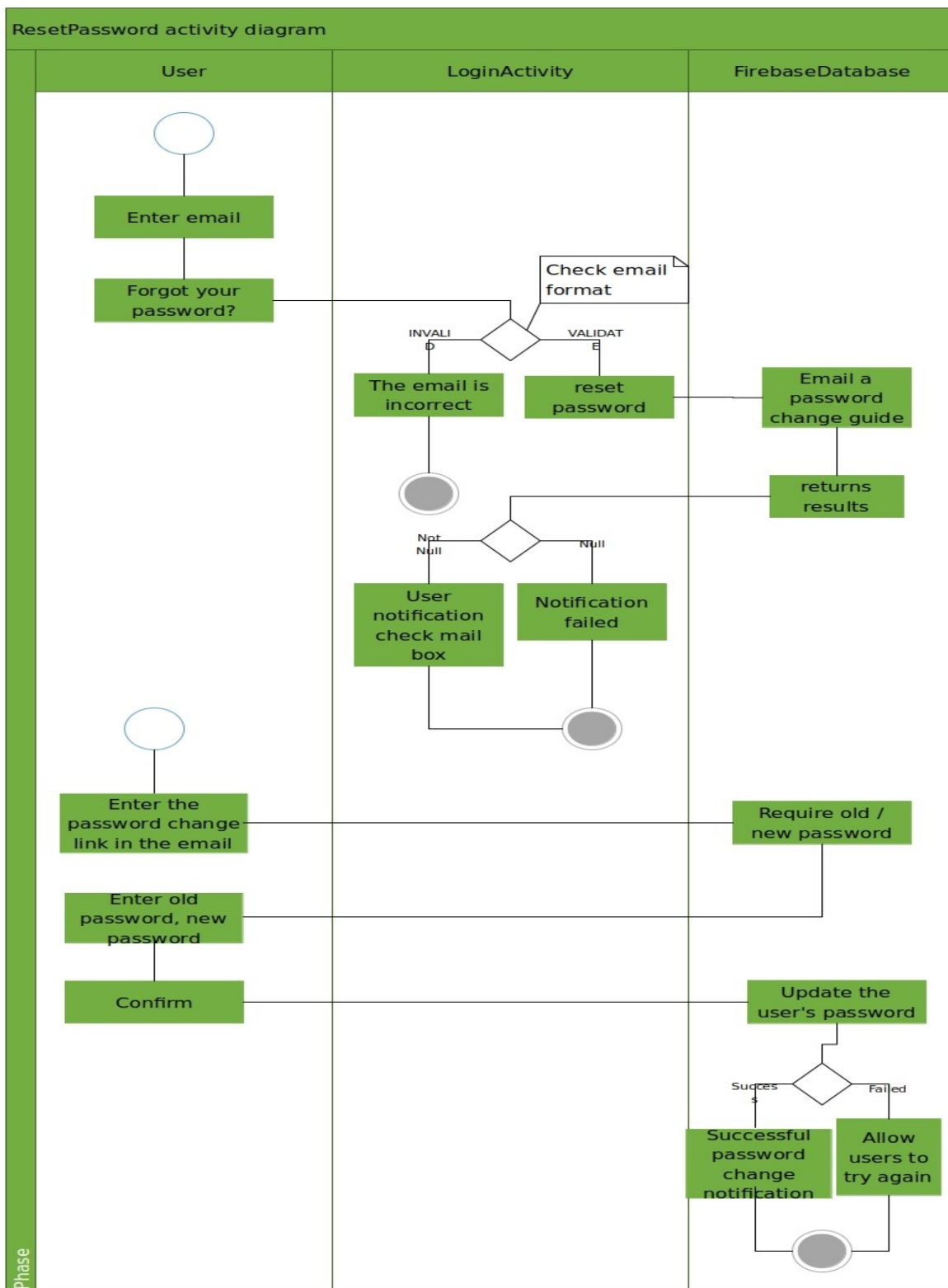- Activity View Profile Module

6.4.1. Activity Friend Module
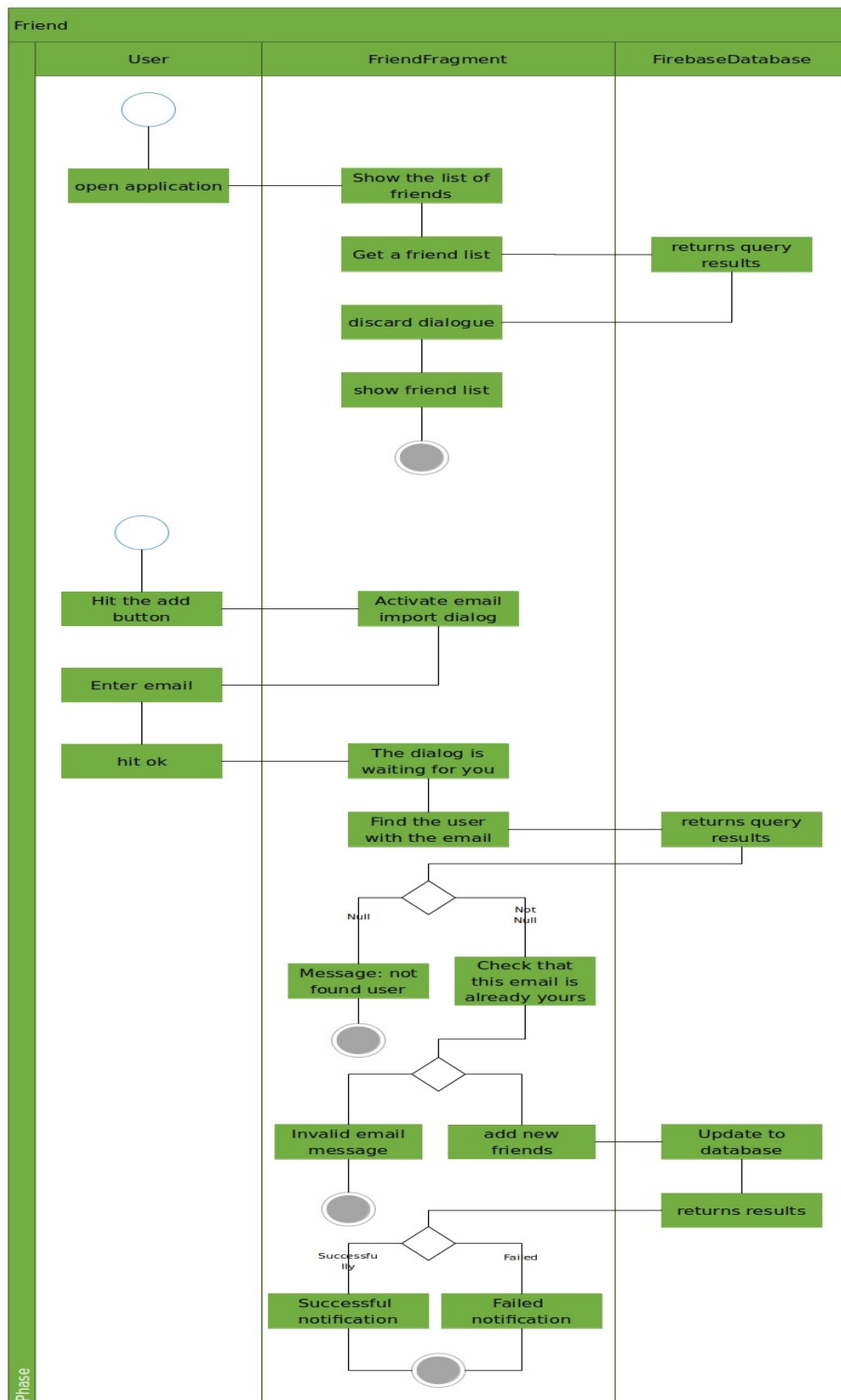
The above module diagram is used to show the working of friend module in the system, this module diagrams indicates the working of the system architecture on the client-side as well as the server-side.
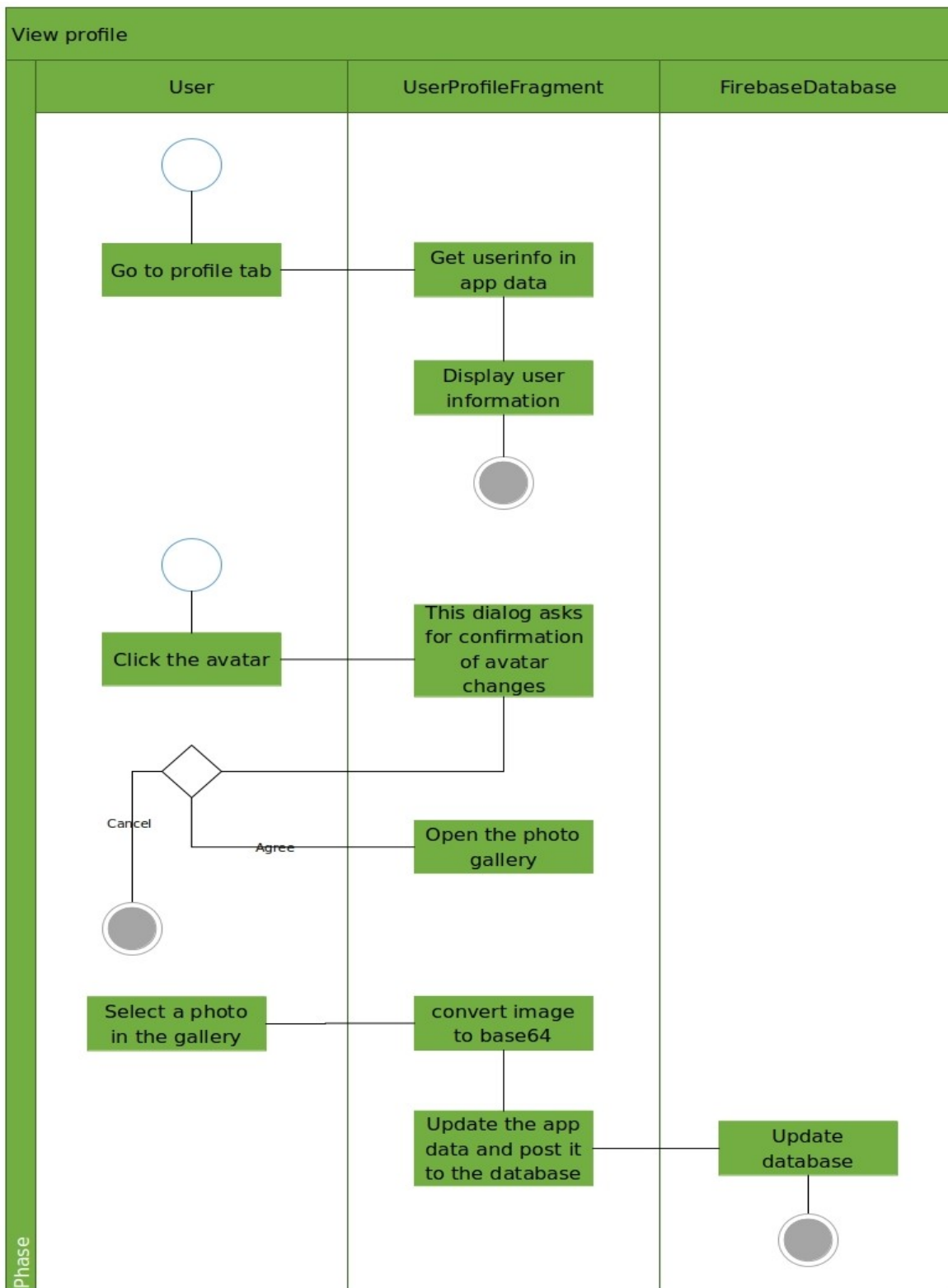
6.4.2. Activity Group Manager Module
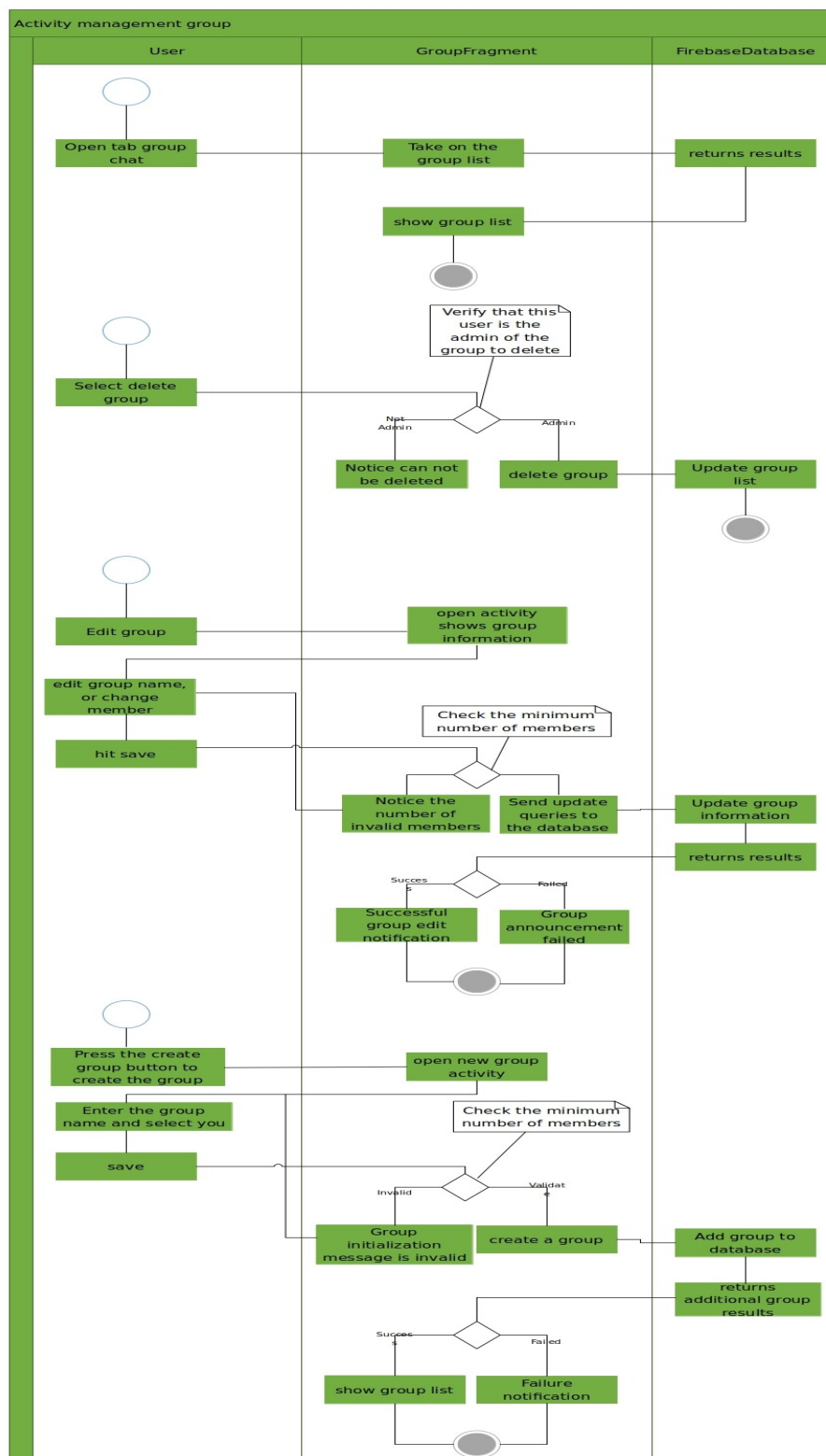
The above module diagram is used to show the working of group module in the system, this module diagrams indicates the working of the system architecture on the client-side as well as the server-side.

## 6.4.3. Activity Login Module
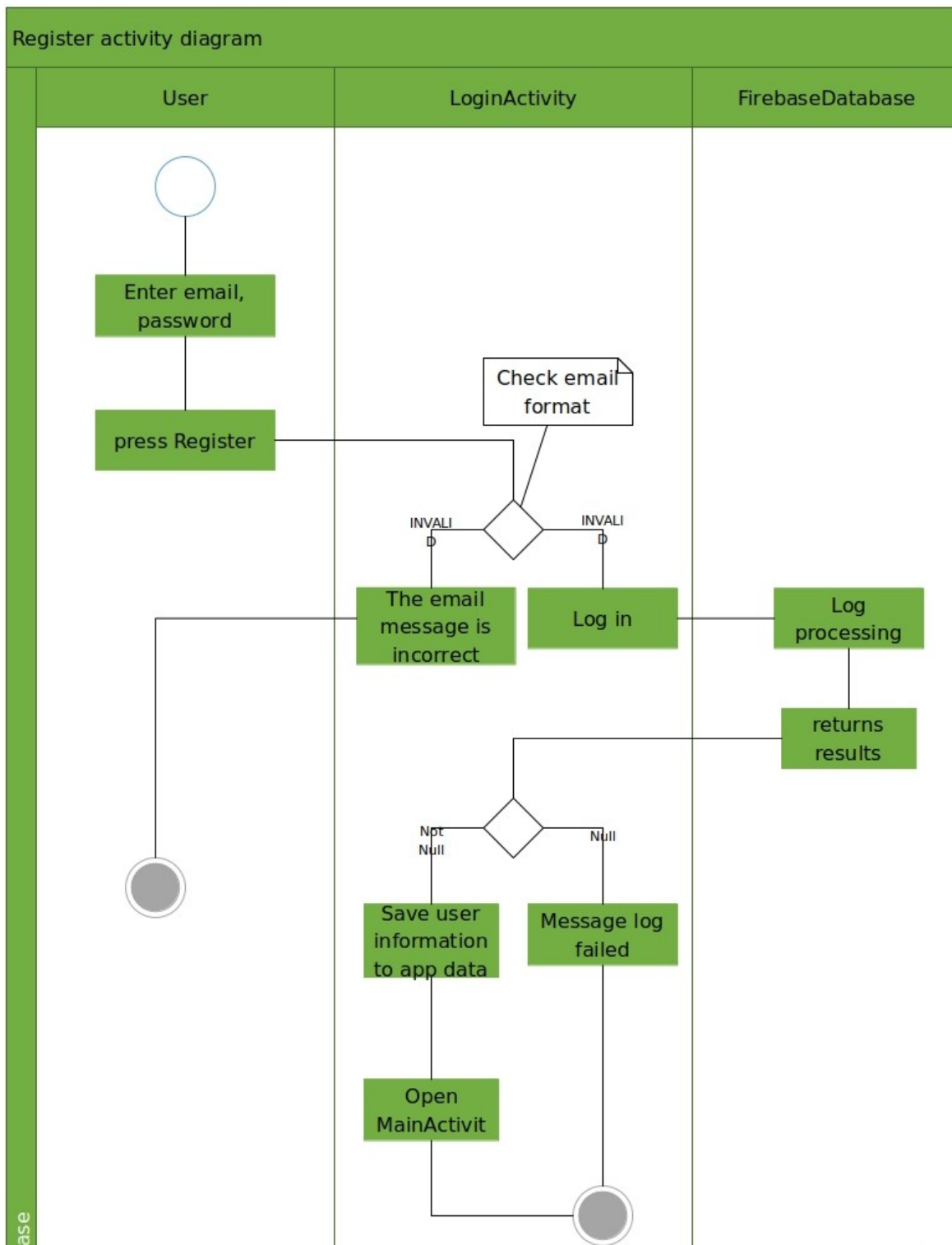
The above module diagram is used to show the working of the login module in the system, this module diagrams indicates the working of the system architecture on the client-side as well as the server-side.

6.4.4. Activity Register Module
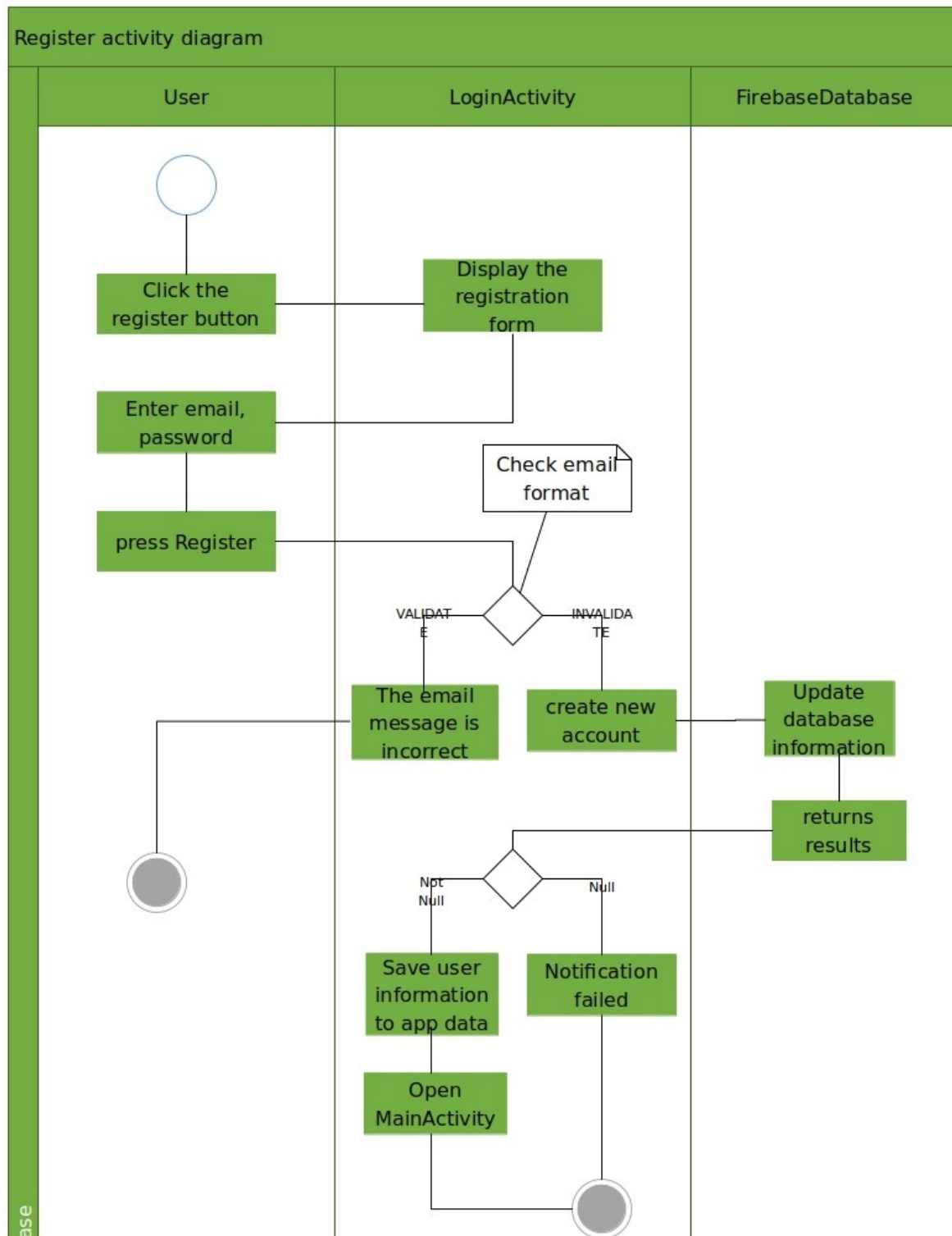
The above module diagram is used to show the working of the registration module in the system, this module diagrams indicates the working of the system architecture on the client-side as well as the server-side.

## 6.4.5. Activity Reset Password Module

The above module diagram is used to show the working of resetting of password module in the system, this module diagrams indicates the working of the system architecture on the client-side as well as the server-side.

6.4.6. Activity View Profile Module

The above module diagram is used to show the working of profile management module in the system, this module diagrams indicates the working of the system architecture on the client-side as well as the server-side.

CHAPTER 7

USER INTERFACE AND WORKINGS

7.1. Login and Registration
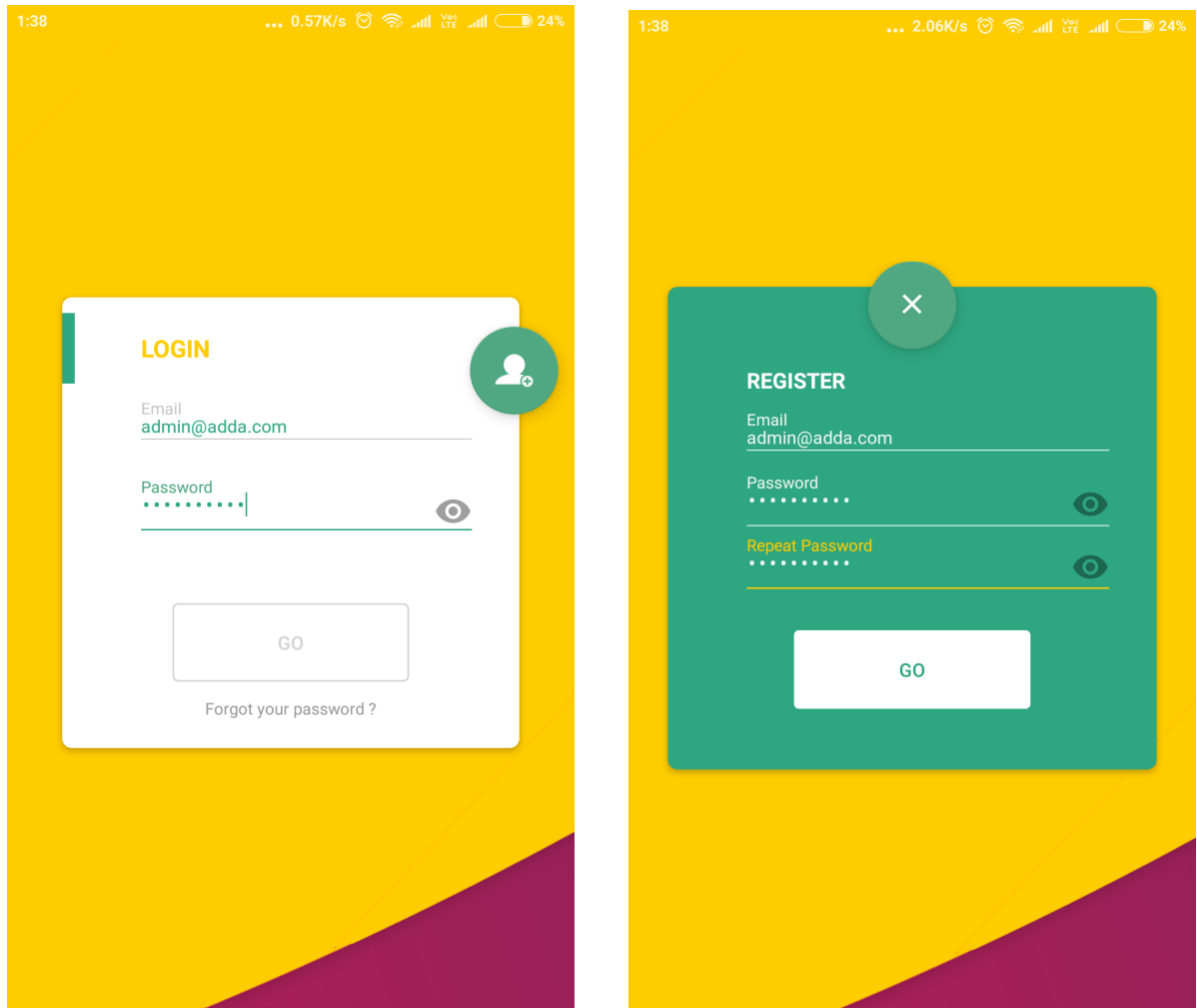
- The first page upon opening of the app is the login page, if the user is alreadys registered they can enter their details and login.

- If the user is new, they can press the green button and register themselves using an email id and password.

- If the user forgets his password, a mail will be sent with a link to give new password.

7.2. Friends and functions
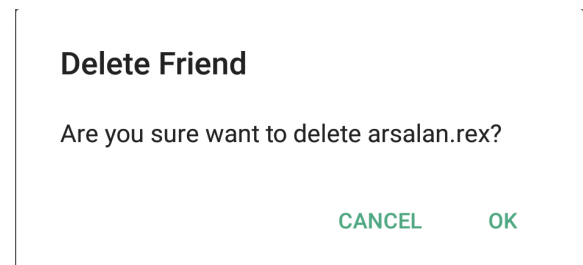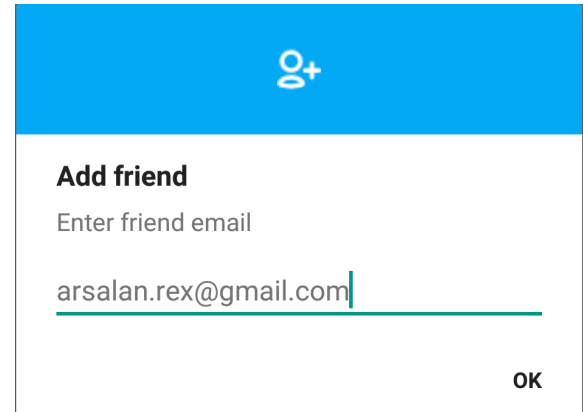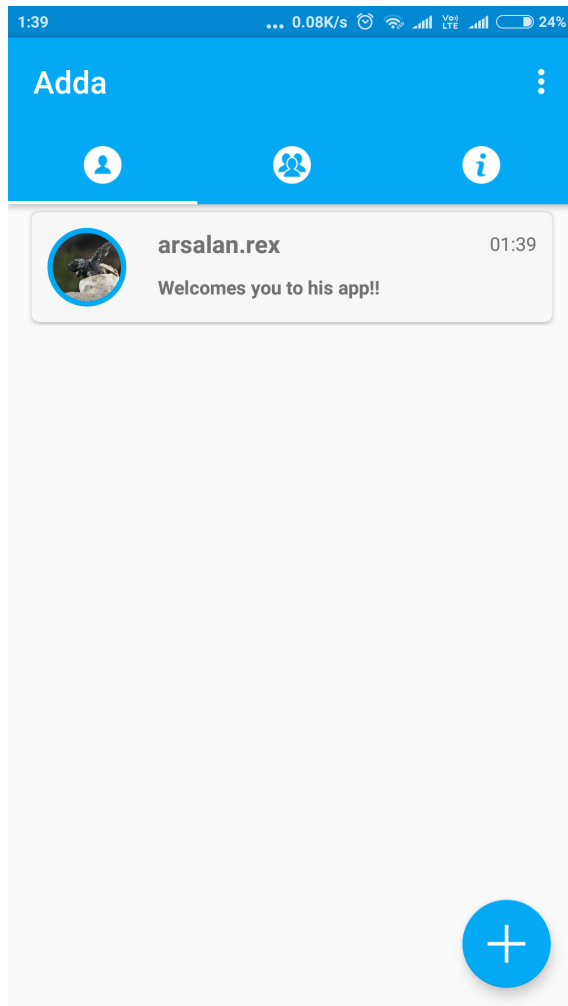
- Once logged in, the first view is of the friend list containing the last message and its time.

- New friends can be added using the + icon at the bottom right.

- New friends can be added using their email id, and not the mobile number.

- Long pressing on the friend's tab will give you the option to delete him/her from your list.

7.3. Groups and functions
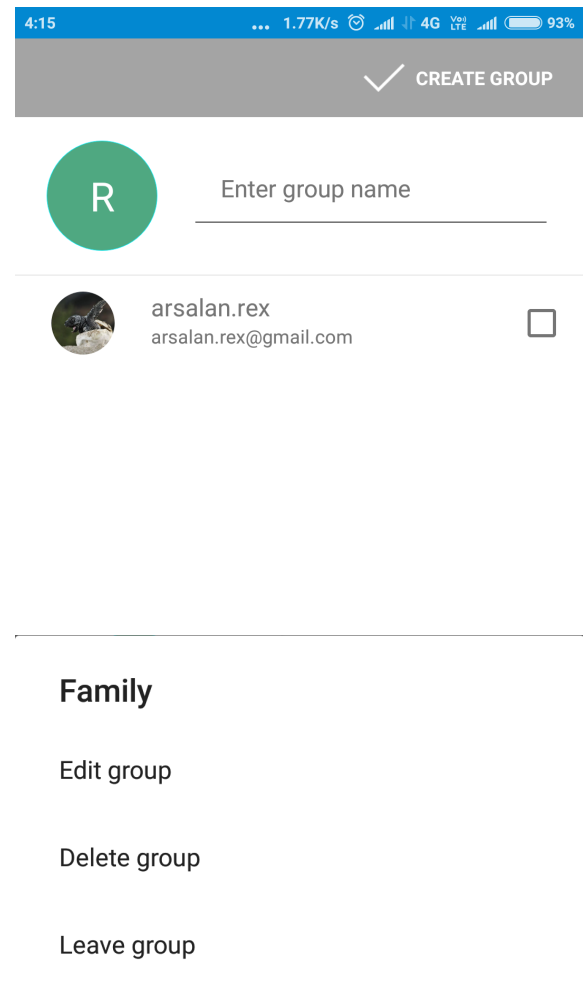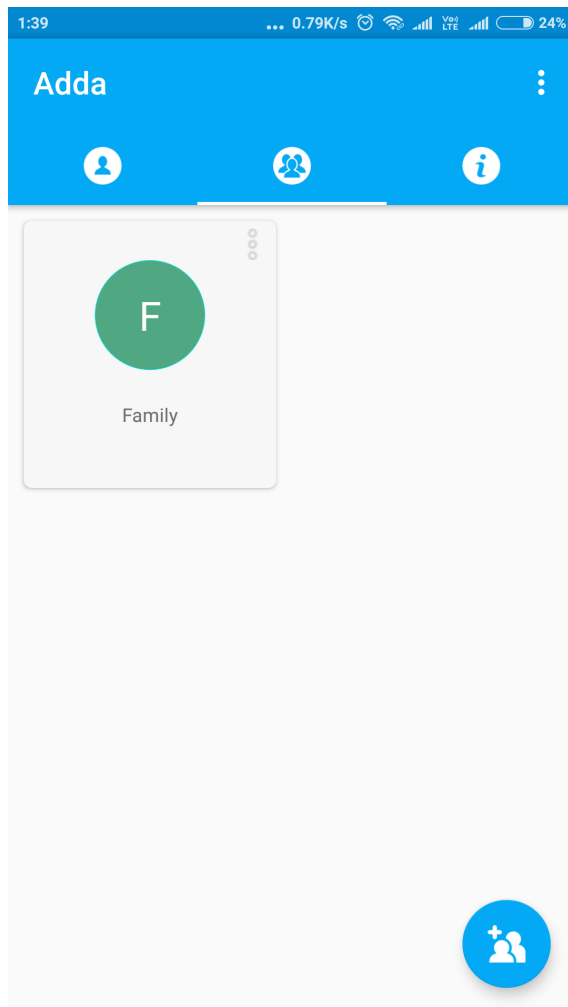
- Sliding to the right the user enters the tab of the different groups he/she is a member of.

- New goups can be added using the icon on the bottom right, a new group must have a minimum of two members.

- The three dots on the group gives you the functions of the group.

- 'Edit group' edits the members of a group and 'Delete group' deletes the group. These are only accessible to admin of the group.

- Users can leave the group using the third option anytime.

7.4. User Profile and functions
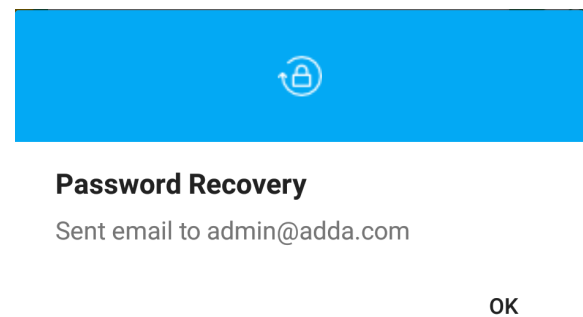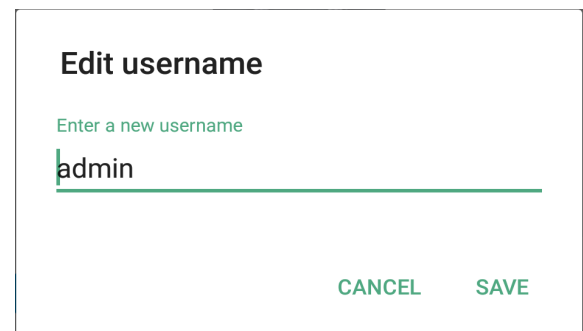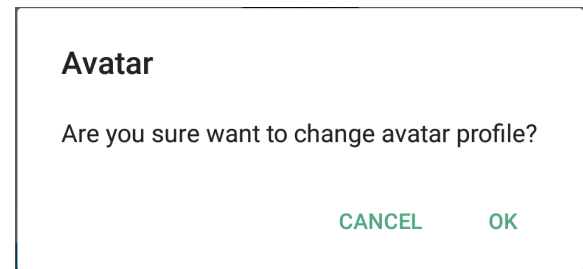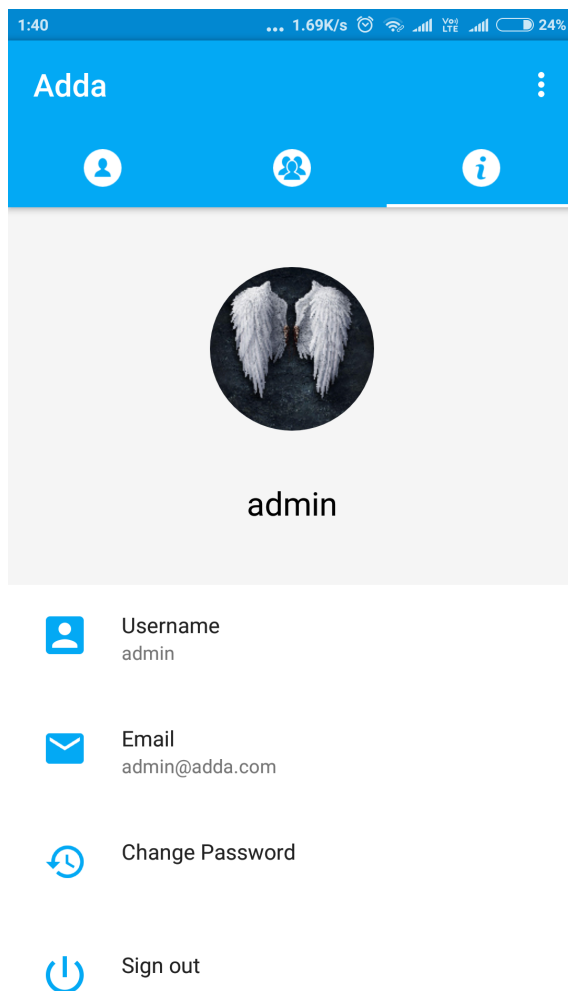
- The third tab on the right shows the user his/her user information.

- On the top is the user's display picture, upon clicking it you get the option to select an image from your galary.

- Below it is the username, upon clicking it you get the option to change your username. This is the name that'll appear on your friends list.

- Below it is the option to change the user's password, upon clicking it you get a dialog box stating that a mail has been sent to your email id.

- The lowest option is to Sign out, upon clicking it the user is redirected to the login page.

7.5. Password Recovery
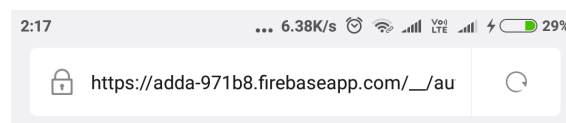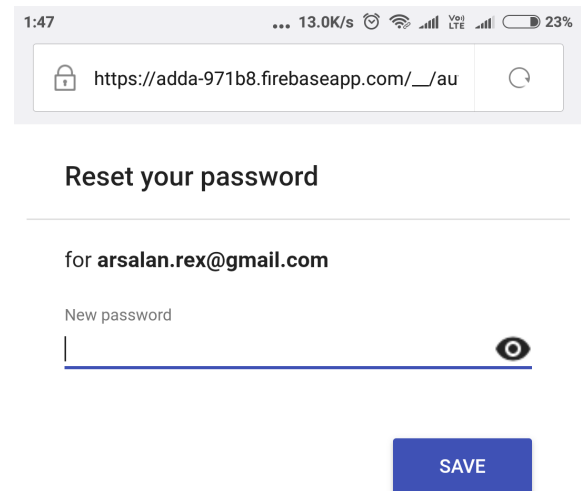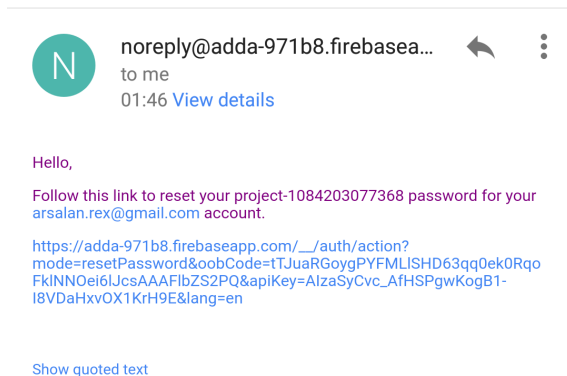
- The mail is recieved in two case, the first when the user presses 'Forgot your password ?' from the log in page and the second whe the user presses the 'Change Password' button from the User Profile page.

- The mail contains a link to reset the password.

- Opening the link in a browser we can reset the password.

- Opening the link a second time or after a long duration gives you an error message.

7.6. Developer toast and Simulation
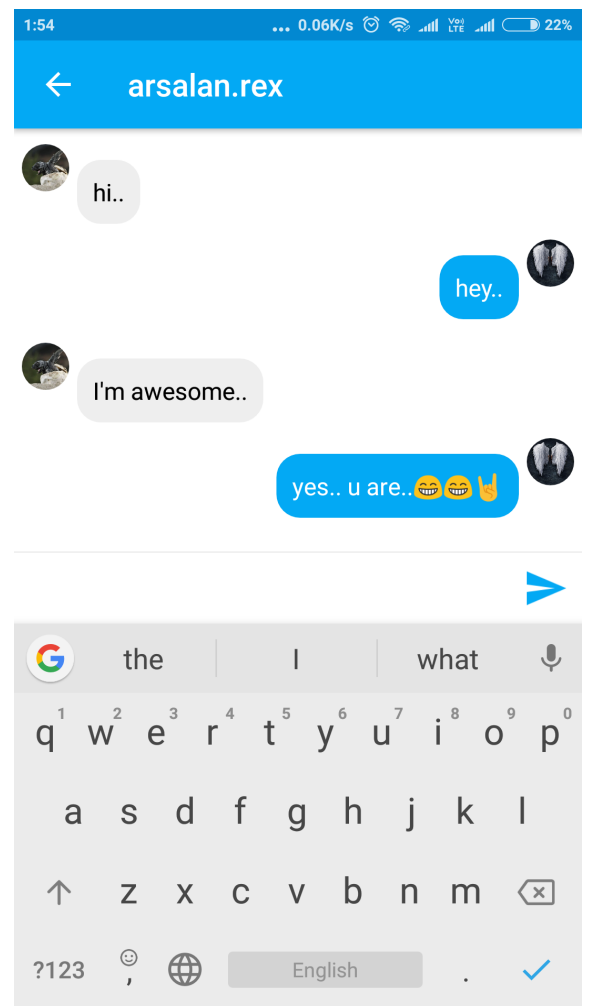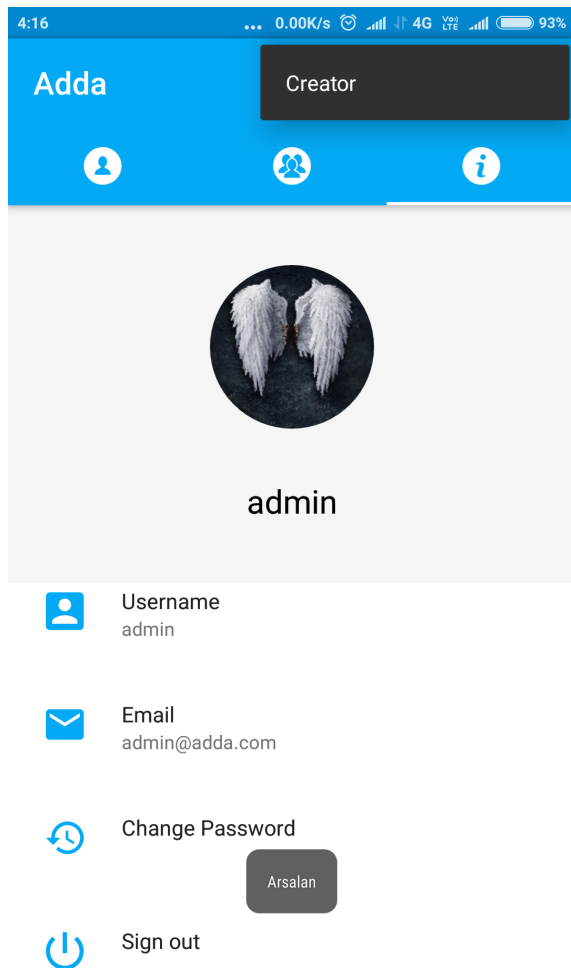
- The three dots on the top right whe pressed give the option to check for developer, upon clicking it you get a developer toast.

- A screenshot of a simulation text between two users is added.

7.7. Database Implementation

```
adda-971b8
  friend
      BbFLhv66SQebn1hV5PQAcpkslE82
      GbwcMsMrRfZoYMrw0cyB4r3TL5Z2
      qjsptz3R1tbcLRwCvZ8wzzUa2UG3
      toWeBwg1S8hiMP49Kf7iTzlUZci1
  group
      -279228668
  message
      -1881479658
      -1200706109
      -1168241182
      -279228668
      -213041710
      662778419
  user
      BbFLhv66SQebn1hV5PQAcpkslE82
      GbwcMsMrRfZoYMrw0cyB4r3TL5Z2
      qjsptz3R1tbcLRwCvZ8wzzUa2UG3
      toWeBwg1S8hiMP49Kf7iTzlUZci1
```

This application uses firebase database to store the information. Above is the stucture of the database implemented. The root gives branches as friend group messeges and user.

- Friend shows the linking of two people, it is modified whenever the users add or delete a friend.

- Group has the child 'groupInfo' and 'members', the first containing the admin and group's name and the second containing members of the group.

- Message records the sender, the reciever, the tect and the timestamp.

- User contains the username, the email address, group of which you are a part of and whether you are logged in or not.

CHAPTER 8

FUTURE SCOPE AND CONCLUSION

8.1. FUTURE SCOPE

This application of e-Notice can be further extended to include the following features:

1. *Categorization of Notice:* Notices can be categorized in different categories, so that its possible for user to easily manage the notices. Categorisation can also be done by making groups. Defining the notice to be circulated in a particular group can make it more secure.

2. *Documents and Images:* The attachments can be added to include PDF files, images or Doc files.

3. *Feedback:* Feedback on the notices can also be taken. It can increase communication among connected members and any issue can be easily sorted out on the spot.

4. *GCM:* Google cloud notifications can be added to the app for instant notification services.

8.2. CONCLUSION

I learned a lot by doing this project.

• Operating system: Ubuntu

• Languages used: Java, Android UI, PHP for backend, kotlin

• Servers Used: LAMP Server, Firebase Server

• Database: MySQL, SQLite

This project is used to reduce the time of the user by avoiding standing in the queue. There is no current systems for online automated notification services. This application has reduced lot of manual work. It has made notifying each and every user very easy and that too with no time and place restrictions

REFERENCES

[1] Basic Android Tutorials, available at http://www.tutorialspoint.com/android

[2] Android Development Tutorials, available at http://www.developer.android.com

[3] Android Book. Wei-Meng Lee, Android 4 Application Development, John Wiley & Sons, Inc., 2012

[4] L A TEXBook. Donald E. Knuth, The TeXbook, AddisonWesley, Boston, 1986