FIELD RING APPLICATION FOR RESEARCHERS

GRADUATE PROJECT

Submitted to the Faculty of
the Department of Computing Sciences
Texas A&M University-Corpus Christi
Corpus Christi, Texas

In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Science

By
Mayuri Gujja
Spring 2017

Committee Members

Dr. AHMED MAHDY
Committee Chairperson

Dr. DAVID THOMAS
Committee Member
ABSTRACT

Collaboration among researchers from across the globe has increasingly become a key to successful research efforts. Such collaboration allows for more holistic visions and a farther impact and outreach. Finding the right collaborators from beyond an individual’s circle or institution can be challenging. A social networking site such as Facebook can be of use to connect with colleagues, however, researchers need a professional and a trusted application that is optimized for such collaboration purposes. A dedicated application would be a better idea so that the analysts can easily communicate with the experts and the fellow researchers to connect, and exchange the information.

The usage of current social networking sites in the field of research is not that effective as the users of the social networking sites are not restricted to research in a specific field. This causes many unwanted posts from various users to show up where getting our required content is like groping in the dark. This clearly establishes the need to throw light on those specified articles we are looking for, on an independent platform. The objective of this project is to create a network for researchers allowing them to connect with potential collaborators based on areas of interest. This application is tentatively named as “Field Ring”.

Field Ring is a platform for the posts and information only related to research area of user’s interest. It is a dedicated and secure android application for the investigators. This application allows the user to subscribe themselves to get recommendations to connect with the researchers who match their interest.
# TABLE OF CONTENTS

Abstract .......................................................................................................................... ii

Table of Contents ........................................................................................................... iii

List of Figures ................................................................................................................ iv

List of Tables ................................................................................................................... v

1. Background and Rationale ......................................................................................... 1
   1.1. Related Work ........................................................................................................ 1

2. Narrative ..................................................................................................................... 3
   2.1 Problem Statement ............................................................................................... 3
   2.2 Motivation ............................................................................................................. 3
   2.3 Product Description ............................................................................................. 4
   2.4 Product Scope ...................................................................................................... 5
   2.5 System Requirements .......................................................................................... 5

3. Proposed System ........................................................................................................ 7
   3.1 Android Studio .................................................................................................... 7
   3.2 Amazon Web Services ......................................................................................... 7
   3.3 Groovy ............................................................................................................... 8
   3.4 SQL Lite ............................................................................................................. 8
3.5 Play Framework ................................................................. 8
3.6 JPA ................................................................................. 9
3.7 Hibernate ........................................................................ 9
3.8 Java ................................................................................ 9
3.9 Design Flow Diagram ...................................................... 10
3.10 Use Case Diagram .......................................................... 12
3.11 User Interface Screens .................................................. 13
   3.11.1 Register and Login .................................................... 13
   3.11.2 Profile Page ............................................................... 15
   3.11.3 Public Forum ............................................................. 16
   3.11.4 Comments, Interested, Share .................................... 17
   3.11.5 Search .................................................................... 18
   3.11.6 Search by Public Posts and Users ............................. 19
   3.11.7 Notifications ........................................................... 20
4. Implementation of Application Modules ................................. 22
   4.1 Manifest ....................................................................... 22
   4.2 Permissions ................................................................... 22
   4.3 Web Apis ...................................................................... 23
4.4 Session Manager .................................................................................................................. 24

4.5 Notifications .......................................................................................................................... 25

4.6 Public Forum ........................................................................................................................... 26

4.7 Log Out .................................................................................................................................. 27

5. System Architecture .............................................................................................................. 27

5.1 System Design ....................................................................................................................... 28

5.2 Architecture of Play Framework ............................................................................................ 29

6. Testing and Evaluation ............................................................................................................ 30

6.1 Testing and Evaluation .......................................................................................................... 31

6.2 Login .................................................................................................................................... 32

6.3 Profile, Project and Organization .......................................................................................... 33

6.4 Image .................................................................................................................................... 35

7. Conclusion and Future Work ................................................................................................... 38

Bibliography and References ...................................................................................................... 40

LIST OF FIGURES

Figure 3.1 Design Flow Diagram ............................................................................................... 10

Figure 3.2 Use Case Diagram .................................................................................................... 12

Figure 3.3 Splash Screen ............................................................................................................. 13
Figure 3.4 Register .................................................................................................................................................. 13
Figure 3.5 Floating Button .................................................................................................................................... 15
Figure 3.6 Profile Page .......................................................................................................................................... 15
Figure 3.7 Public Forum ....................................................................................................................................... 16
Figure 3.8 Comment ............................................................................................................................................... 17
Figure 3.9 Share .................................................................................................................................................... 17
Figure 3.10 Share opens Gmail ........................................................................................................................... 18
Figure 3.11 Search Section ................................................................................................................................... 18
Figure 3.12 Search by Post .................................................................................................................................. 19
Figure 3.13 Search by People ............................................................................................................................... 19
Figure 3.14 Notifications ...................................................................................................................................... 20
Figure 4.1 Classes in Manifest ............................................................................................................................. 22
Figure 4.2 Permissions in Manifest .................................................................................................................... 22
Figure 4.3 Web Apis .............................................................................................................................................. 23
Figure 4.4 Instance in Amazon Web Services ..................................................................................................... 24
Figure 4.5 Session Manager Class ...................................................................................................................... 25
Figure 4.6 Notifications Class ............................................................................................................................. 26
Figure 4.7 Public Forum ....................................................................................................................................... 27
Figure 4.8 Log Out.................................................................27

Figure 5.1 System Design..........................................................28

Figure 5.2 Architecture of Play Framework .......................................29

Figure 6.1 (a) Registration Positive Test Case......................................31

Figure 6.1 (b) Registration Negative Test Case....................................31

Figure 6.2 Login Test Case................................................................32

Figure 6.3 (a) Project Positive Test Case............................................33

Figure 6.3 (b) Project Negative Test Case...........................................34

Figure 6.3 (c) Paper Test Case..........................................................35

Figure 6.3 (d) Research Interest Test Case ..........................................35

Figure 6.3 (e) Organization Test Case................................................35

Figure 6.4 (a) Image Positive Test Case.............................................35

Figure 6.4 (b) Image Negative Test Case............................................35

Figure 6.5 Feedbacks Chart.............................................................37
1. BACKGROUND AND RATIONALE

Researchers spend countless hours working on their projects, writing groundwork papers and publishing the articles. They often look to seek some help from the people who work in similar areas. Also, savvy investigators are finding other ways to ensure that their work is noticed. Although the analysts have a good friend circle, they might not find a perfect friend or colleague who has the similar interest to share the research. In such cases, the researcher typically reaches out to the social networking site. But, it is hard to find a professional networking application as the analyst generally prefers to be away from all other areas in the social site that deviate from the research. The social networking sites expose the user to all the areas that might not be of use to an investigator. Moreover, it would be difficult to find the person who has similar research interests, and hard to see the research background to be able to connect with. This problem needs a specialized application that can help the experimenters not only search for the connections, but also get recommendations of the people who share the similar interest so that they don’t miss out on any important ideas. In addition, a public forum can help the researchers ease their work in reaching out to everyone to seek help. It can directly showcase the user all the posts when he opens the public forum.

1.1. Related Work:

Currently, there is no dedicated mobile application for the researchers to connect with the fellow researchers. In fact, there are some social networking websites to connect with people, but the users get diverted because of the entertainment content in such sites like Facebook, Twitter. The lack of professionalism in such sites isn’t a prerequisite for the
researchers who use them. A dedicated app is usually preferred to be used as it doesn’t deviate from the content that is looked for. The direct approach is encouraged to get the right information at the right time. A group in a social networking website may not serve the whole purpose of what a dedicated mobile application does. A group could be stipulated to a set of users, and might not reach out to the outer world. A dedicated application can be reached out to everyone, and recommendations make life easier.

All the current applications like Facebook, Twitter, LinkedIn focus on just communicating with the peers. Sharing the research documents must be given high priority. It is important for the researchers not only to communicate, but also to be able to share the documents with each other. An email or phone call between two peers, makes the conversation more elementary.
2. NARRATIVE

2.1. Problem Statement

Most researchers find it difficult to obtain data or information from their study population or area of study. Despite adequate planning, the practical part of the research is unknown. The available resources might not be sufficient to start or continue research. Especially, for the beginners, it’s a difficult task to start the research without consulting an expert. Guidance makes it simple to achieve the goal. Research is an area where the information must be gathered from different perspectives. Team work yields the best results compared to individual efforts. Hence, researchers often seek help from their peers to get things done right. An expert’s advice would be of more use, and a reference of their documents would give a better idea of the goals to be achieved in research. Without networking with different people, research is difficult to be conducted. Thoughts from a wide range of people helps the research to be conducted in an extensive way.

2.2. Motivation

To help researchers find an easy way to connect with fellow researchers, an android application can cater to the researcher’s needs effectively. The Field Ring application provides a networking platform to the researchers from different subject areas, and it ensures security and privacy. A researcher need not search for researchers in a big ocean of users, when there is a dedicated application exclusively for the researchers themselves.
2.3. Product Description

The Field Ring application allows the user to create an account to login to the application. On the first screen, the application asks for the field of study, for example arts, computers, communications etc. so that it starts sending recommendations once the user starts using the application. Besides this, while creating the profile it also asks for the research interests, papers and projects one is working on. This could help the fellow researchers view the profile and send a connection request if the content in the profile is of any use. The recommendations are based on the research interests and field of study. The research interests are the topics from a specific field of study. It could be “Machine Learning” research interest from the Computer Sciences, or another research interest like “Microbiology” from the field of Chemistry. The search bar in the application acts as an important element as it eases the work of the user in finding the appropriate person, paper or project in the application. A public forum in this application displays the posts of all the users. The main advantage of this forum is that it allows the users to enter the forum and discuss the views, without disturbing the rest of the audience. There is no notification spam as there are limited notifications sent to the users of the app that do not include the notifications of everyone’s posts. Requests from the connections will be notified, but the posts would not be notified. Hence the user’s privacy is preserved. A user can check the public forum for information/postings about interest.
2.4. Product Scope

This application can be accessible by any user who has an Android mobile with the application installed and proper network connection. The application is designed to meet the following needs of the users as described below:

1. Create Profile
2. Add research interests, add papers, add projects
3. Search for people, interested projects
4. Discuss on the public forum
5. Get notified when you get a new connection
6. Send private messages to the peers
7. Share documents via emails
8. Make phone call to the interested peers

Admin will be able to explicitly notify all the users about any event that could be useful to all the researchers. All the users of the application are targeted with the same notification.

2.5 System Requirements

The following requirements are required to develop this application

- Operating System:
  - Windows 10
- Database:
  - Amazon Web Services
  - SQLite
The Amazon Web Services is used because of the following features:

- Easy to Use
- Flexible
- Cost Effective
- Reliable
- Scalable and high performance
- Secure
- 10k – 20k users

- Integrated Develop Environment:
  - Android Studio
  - Play Framework

- Software:
  - Android Software Development Kit
  - JAVA SE 8

- Hardware:
  - A Personal Computer and an Android Mobile Device
3. PROPOSED SYSTEM

The proposed system uses the following to develop *Field Ring Android Application*.

3.1. **Android Studio**

Android Studio is the official integrated development environment (IDE) for the Android platform[12]. It was announced on May 16, 2013 at the Google I/O conference. Android Studio is freely available under the Apache License 2.0. Android Studio was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. Based on JetBrains' IntelliJ IDEA software, Android Studio is designed specifically for Android development. It is available for download on Windows, macOS and Linux, and replaced Eclipse Android Development Tools (ADT) as Google's primary IDE for native Android application development.

3.2. **Amazon Web Services**

Amazon Web Services (AWS) is a secure cloud services platform, offering compute power, database storage, content delivery and other functionality to help businesses scale and grow.[9] Explore how millions of customers are currently leveraging AWS cloud products and solutions to build sophisticated applications with increased flexibility, scalability and reliability. The AWS Cloud provides a broad set of infrastructure services, such as computing power, storage options, networking and databases, delivered as a utility: on-demand, available in seconds, with pay-as-you-go pricing.
3.3. Groovy

Apache Groovy[6] is a powerful, optionally typed and dynamic language, with static-typing and static compilation capabilities, for the Java platform aimed at improving developer productivity thanks to a concise, familiar and easy to learn syntax. It integrates smoothly with any Java program, and immediately delivers to your application powerful features, including scripting capabilities, Domain-Specific Language authoring, runtime and compile-time meta-programming and functional programming.

3.4. SQL Lite

SQLite is a popular choice as embedded database software for local/client storage in application software such as web browsers.[5] It is arguably the most widely deployed database engine, as it is used today by several widespread browsers, operating systems, and embedded systems (such as mobile phones), among others. SQLite has bindings to many programming languages.

3.5. Play Framework

Play is an open source web application framework, written in Scala and also usable from e.g. Java (Play includes a Java API in more recent versions), which follows the model–view–controller (MVC) architectural pattern.[8] It aims to optimize developer productivity by using convention over configuration, hot code reloading and display of errors in the browser.
3.6. JPA


Persistence in this context covers three areas:

- the API itself, defined in the javax.persistence package
- the Java Persistence Query Language (JPQL)
- object/relational metadata

3.7. Hibernate

Hibernate ORM (Hibernate in short) is an object-relational mapping tool for the Java programming language. It provides a framework for mapping an object-oriented domain model to a relational database. Hibernate solves object-relational impedance mismatch problems by replacing direct, persistent database accesses with high-level object handling functions.

3.8. JAVA

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture.
3.9. DESIGN FLOW DIAGRAM

The following design flow diagram explains the flow of the data and the sequence in which they are performed. When the application is started, the user can either log in or register. After successful login only, the app allows the user enter the application. Four tabs namely “Profile”, “Public Forum”, ”Notifications”, ”Search” are seen in the first page, and when clicked on each tab, the sub modules explained below will be visible.

![Design Flow Diagram]

Figure 3.1 Design Flow Diagram
Any user who has an interest to work in a research are open to create an account in the *Field Ring* app. The user must log in if an account already exists, otherwise the registration process must be completed. The registration process contains the following details:

- Username
- Email
- Password
- Academic Status
- Field of study

The app utilizes this information for recommending the users with the people sharing similar profile status. The similarity is based on the academic status, and the field of study. After registration, the user can login with the email and password he had used while creating the account.

Profile page, Public forum, Search and Notifications are accessible after successful login into the app. The profile page allows the user to add research interests, projects, papers and the organization. It displays everything that has been entered in this page. The public forum page is used to post research posts which are public. The posts that are posted in the page are visible to everyone who has an account in the *Field Ring*. The posts can be shared through different other applications present on phone. If there is a need to contact the person who had posted in the forum, “interested” button can be clicked, and the gmail app opens up to contact the fellow researcher directly. As there are huge number of posts,
to get the appropriate post, there is a search bar that filters the posts with the name entered in the search bar. This can also be used to search for the users in the search page.

The suggested users section is provided in the search page, which shows the connections whose profile status is similar with the user who logged in.

3.10. Use Case Diagram:

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions that some system can perform in collaboration with one or more external users of the system. The actors are the users who login to the Field Ring account.

![Use Case Diagram](image)

**Figure 3.2 Use Case Diagram**

The above use case diagram indicates the actor who is the end user. The actions of the end user would be registering in the application, and have permission to access the
profile, public forum, search and notifications page. The rest of the three actors on the right side of the figure 3.2, are the connections who receive request from the first actor on the left side. Any number of users can exchange connections between each other. Everybody is notified with the connection requests they receive from the other users.

3.11. User Interface Screens

The following user interface screens are the screens that are captured in the android tablet device RCA with the version 6.0.

3.11.1 Register and Login

Figure 3.3. and 3.4 show the initial screen, and the register screens.
Splash screen is the first screen that appears to the user. It keeps the user busy with the spinner, while it fetches all the necessary information from the server. Once all the details are ready, it opens up the login screen.

The registration screen is used to register into the Field Ring app. The details that are taken into account are the full name, email address, password, academic status, and the field of study. All these details are stored in the database table “user”. After registration, the user is directly taken to the login page, where the same details email and password are used to log in the application.
The email and password entered while creating the account are stored in the database, and checked again when the user logs in. If the details don’t match, the app doesn’t allow the user log in, instead it throws an error message saying that the email or the password entered is incorrect.

3.11.2 Profile page

Figures 3.5 and 3.6 show the screenshots of the profile page.

![Figure 3.5 Floating button](image1.png) ![Figure 3.6 Profile Page](image2.png)

The above figures show the profile page features. A floating button is applied on the page to facilitate different operations in the same page. The “Add Organization”, “Add Projects”, “Add Papers”, “Add Image”, and “Log out” options are included in the floating
button. When the user clicks on each button, the respective page appears, and allows to enter the information. The log out button logs off the user from the application.

All the information that has been entered through the floating button are being displayed on this page in their respective sections.

3.11.3 Public Forum

Public forum page is the place where the user can write the posts. These posts are displayed to all the users when logged in. The below figure shows the profile page.

![Figure 3.7 Public Forum](image-url)
The posts are public so that all the users can view everyone’s posts irrespective of the subject area. This is to give the users a broader scope to exchange information with each other. The most recent posts are displayed on top of the list. The floating button at the bottom of the page is used to write a post. It takes the user to the next page, to enter the content, and add it. The page then returns to the public forum page to show the post that has just been entered.

3.11.4 Comments, Interested and Share

Three buttons are included under each post, “Comment”, “Interested”, and “Share”. Out of many posts, only a few of them might be of real use to the researchers, to which comments can be posted. There is no limit for the audience that can post the comments, share and use “interested” feature. It depends on the server used for the application. If it’s a premier server, then it can exceed the limit of free tier services.

Figure 3.8 Comments

Figure 3.9 Share
As in fig 3.8, the comments are visible in the page after the post is clicked. The most recent comments are displayed on the top of the list. Everybody is open to comment on any post in the public forum.

The “Interested” button can be used to contact the user directly. This button takes the user to the gmail application in the phone that can be used to send messages, and share documents privately via emails as shown in fig 3.10.

The “share” button is used to share the posts via any other application like hangouts, drives, Bluetooth etc. as shown in the fig 3.9.

3.11.5 Search

Search is one of the tabs of the main tool bar. When the tab is switched to the “Search” section, it displays all the suggested users in the first section of the page. These suggestions pop up here based on the field of study selected while creating the account. There is no limit for the suggested users. It displays all the users that share the same profile. The “connect” button enables the researcher to send a connection request to the selected person.

The current system just sends a request to the person, and notifies the other user about the request. Though the user had already sent a request, the suggestion still appears, but when the connect is clicked again, it pops up a message that the request had already been sent.
3.11.6 Search by public posts and people:

As mentioned earlier, the search is facilitated with two options, search by public posts and search by users. As shown in the fig 3.12, the “test” search shows the results with the posts that have the keyword “test”. Similarly, in the fig 3.13, the search is filtered by person “test”. It shows the users that have the name “test” in the page. It also shows the connect button to send the connection request.

The search bar plays an important role as it makes it easy to search for the required post or user, rather than scrolling the app throughout the lists.
3.11.7 Notifications

The notification screen just displays the connection requests from different users across the app. The notifications aren’t enabled for the public forum posts, to avoid disturbances to the researchers. As the posts are from different varieties of fields, and are plenty in number, the researchers need not know every post posted in the public forum. Hence the notifications are disabled for the posts, and just enabled for the connection requests.
Figure 3.14 Notification

The most recent notifications are appeared on the top of the list, along with the rest of the notifications below. In each notification, an “accept” button is enabled to accept the friend request. If the request had already been accepted, it displays the message saying that the request had already been accepted in the current system.
4. IMPLEMENTATION OF APPLICATION MODULES

The important code snippets are shown in this section. All the code of the application is programmed in the Android Studio.

4.1. Manifest

The manifest file comprises of all the activities included in the application. The permissions like internet permissions etc. are included in the manifest file.

![Figure 4.1 Classes in the Manifest file](image)

4.2. Permissions

The permissions shown in the fig 4.2 are the internet permissions provided to the app.

![Figure 4.2 Permissions in the Manifest file](image)
This android app is connected to the amazon web services [9]. The code in Figure 4.1 is to declare all the java classes in the manifest file. All these declarations are made in the manifest file in Android Studio.

4.3. Web apis

The WebApis class is used for declaring the ip address that was generated in the amazon web services. An instance is created in the amazon web services to get the IP address. This IP address is fetched in the WebApis android code to make it run on the server.

```
public class WebApis {
    static String ip = "52.35.119.217";

    public static String BASE_URL = "http://"+ip+"/AndroidWebServices/saveFCMToken";
    public static String URL_LOGIN = "http://"+ip+"/AndroidWebServices/androidlogin";
    public static String URL_REGISTER = "http://"+ip+"/AndroidWebServices/androidregister";

    public static String getFieldofstudy(){
        return "http://"+ip+"/AndroidWebServices/getFieldofstudy";
    }

    public static String getAcademicStatus(){
        return "http://"+ip+"/AndroidWebServices/getAcademicStatus";
    }

    public static String ADD_RESEARCH = "http://"+ip+"/AndroidWebServices/addResearch";
    public static String ADD_ORG = "http://"+ip+"/AndroidWebServices/addOrganisation";
    public static String ADD_PROJECT = "http://"+ip+"/AndroidWebServices/addProject";
    public static String ADD_PAPER = "http://"+ip+"/AndroidWebServices/addPaper";
    public static String PHOTO_URL = "http://"+ip+"/AndroidWebServices/addphoto";
}
```

Figure 4.3 Web Apis in Android Studio

Each page is provided with it’s name at the end of the link that contains the IP address, so that the server goes to the page when it sees such URL in the code. The instance creation
is shown in the fig 4.4. This IP address is used to connect as virtual server in the system, and connect to the server.

![Instance in Amazon Web Services](image)

**Figure 4.4 Instance in Amazon Web Services**

### 4.4 Session Manager

The session manager class stores the user sessions, i.e, the user session ID. It keeps track of the session ID till the user logs out of the application. Unless the user logs out, the session stays in the application, and it remembers the user every time the app is opened.

The session manage class is shown in the figure 4.5.
4.5 Notifications

In the notifications class, NotificationCardAdapter is kept running in the doInBackground method so that it works in background and pops up the notifications to the user.
Figure 4.6 Notifications Class

All these notifications, once fetched from the adapter in the background, are displayed on the notifications page through recycler view. The recycler view makes the notifications display in a list, pushing the most recent notifications on to the top of the list.

4.6. Public Forum

The public forum displays the posts that are published in the “Add Post” page. The “posts” method gets the posts from the database using the query “order by postedAt desc”. The public forum class is shown in the figure 4.7.
This query is the short form of the select query that displays the posts as per the order.

4.7 Log Out

The log out class is shown in the figure 4.8.
5. SYSTEM ARCHITECTURE

The architecture of the system is shown in this section. The system design and the architecture of the play framework are shown in the figures below.

5.1. System Design:

The Android code is deployed on the Amazon Web Services server, which includes model and controller. The model interacts with the MySQL database. The whole business logic of the application lies in the controller of the server.

![Figure 5.1 System Design](image-url)
5.2. Architecture of Play framework:

The architecture of the play framework is shown in the figure 5.2.

Figure 5.2 Architecture of Play Framework
6. TESTING AND EVALUATION

Verification and validation (V&V) is the process of checking that a software system meets specifications and that it fulfills its intended purpose. It may also be referred to as software quality control [10]. It is normally the responsibility of software testers as part of the software development lifecycle. In simple terms, software verification is: "Assuming we should build X, does our software achieve its goals without any bugs or gaps?" On the other hand, software validation is: "Was X what we should have built? Does X meet the high level requirements?"

The Field Ring Application is developed bug free, handling most of the exceptions, the application might throw. It is tested by fetching wrong inputs, duplicate data, etc. Assuming that the errors crash the app, all of them are handled by prompting suitable warning alerts. The app is validated by checking if all the requirements mentioned below, that it is supposed to satisfy are met.

1. The app should check for the existing users when trying to create a new account
2. It should be able to send connection requests.
3. It must allow exchanging of messages.
4. Should be able to recommend users based on the profile information

The current application is ready to be uploaded in the Google Play Store. However, it works well only for a limited number of users as the AWS free tier service serves 10k – 20k users. If there is no internet connection, the application just doesn’t get updated with the new posts, but it doesn’t get crashed.
This application is tested on RCA Android tablet with Android version 6.0.1 and other devices of the friends for feedbacks. But the screenshots are grabbed from the RCA tablet.

The following are the test cases that were tested:

### 6.1. Registration

The user needs to fill all the fields in the form provided while registering for the application.

![Registration Positive Case](image1)

**Figure 6.1 (a) Registration Positive Case**

![Registration Negative Case](image2)

**Figure 6.1 (b) Registration Negative Case**

### Positive Test Case:

If all the fields are filled out in the form, the student is logged into the app and opens up the login page from registration page. After successful registration, it prompts the message “User successfully registered. Try login now!” as shown in the above figure 6.1.
Negative Test Case:

If the user misses out on any fields an error message is displayed asking to enter all the fields as shown in Figure 6.1(b). Only after the details are filled, the error message disappears, and takes the user to the login page. However, the two drop down boxes for the academic status and the field of study need not be selected. If they aren’t selected, they consider the first fields as the input, and thus continues to the next login page provided the full name, email and password are filled.

6.2. Login

The login page is displayed once the user enters the application. Login can be successful only with the correct email and password.

Positive Test Case:

After a successful registration in the app, the user is taken to the next page, which is the main activity containing all the tabs “profile”, “public forum”, “search” and, “notifications”.

Negative Test Case:

If the email address or password doesn’t match with the one in the database, then the application throws an error “Incorrect email or password” as shown in the figure 6.2.
6.3. Project, Paper, Organization

The project, paper and organization pages are shown in the below figures:

Positive Test Case:

When the user fills the field in “Add Project”, the successful entry prompts a message “Project successfully added”.

Negative Test Case:

If the page is submitted without entering the project details, then the page throws the error “Please enter your details”. Empty field will not be submitted.
The following are the figures that show the negative test cases while adding paper, organization and research interest.
6.4. Image
**Positive Test Case:**

To notify the user that the image is successfully being updated into the database, a spinner is enabled which states that the image is being uploaded as shown in the figure 6.4. After it has been uploaded, the page takes the user back to the profile page. This indicates the successful upload of the image.

**Negative Test Case:**

If the image is too big to be uploaded into the database, or if it’s empty, then the error message prompts stating that “Image is too big”. When the correct size is uploaded, this error message disappears.

**6.5. User Testing and Feedbacks:**

The APK of the *Field Ring* app has been sent to a set of people in the course work, and asked to review the application on the rating of 1 – 5. The following are a sample of ratings and feedbacks:

1. **Reviewer 1:**
   
   “Good app. Best platform for users interested in research work. Has good scope”
   
   Rating: 4/5

2. **Reviewer 2:**
   
   “The UI looks nice. Performance is better with good internet connectivity, however, it doesn’t work when there is no internet.”
   
   Rating: 3.8/5
3. **Reviewer 3:**

“Well, the idea of a dedicated app for the researchers is interesting. Unique idea. Good UI. Brilliant features”

Rating: 4.5/5

4. **Reviewer 4:**

“Privacy is maintained well. Good UI, looks like a professional app. Fast performance. But better with a chat option”

Rating: 4/5

5. **Reviewer 5:**

“Good app. Some sections look blank, may be because it is the new app with limited users. Easy navigation with swipe option”.

Rating: 3.7/5

Below is the chart that shows the graph of the overall ratings:

![Figure 6.5 Feedbacks chart](image-url)
7. CONCLUSION AND FUTURE WORK

The purpose of the Field Ring application to provide a platform for all the people who are interested in research or learning new things about the various technologies in their field. Any user can install the field ring app, create an account, select the interested field of study, and proceed logging in. The app enables the user create their profile, and connect with the fellow researchers. Users can click on “Interested” button if the public posts interests, and can share the documents through emails. Posts can even be commented and shared through different applications like Hangouts. Notifications are facilitated to get notified when there are any connection requests.

The initial goals of the application that are met now are:

1. It provides an easy way to connect to the researchers, and the users who are interested in the research work. There is no need of searching for such people in different social networking sites.

2. It directly recommends the users who share the similar profile, so that it eases the work of searching for specific people.

3. Comments and Interested options play a key role as they enable the peers interact with each other, and share the documents privately. This ensures the security of the users.

4. There is no disturbance of notifications as it is enabled only for the connection requests. This way, it maintains the privacy of the users.
The future enhancements for this app can include the following:

1. Provide deleting option for the posts, research interests, papers, organizations, projects, profile images.
2. Enable viewing of everyone’s profile when clicked on their names in public forum.
3. Able to create a group to share information in specific field.
4. Remove the connections that have already sent connection requests in suggested users section.
5. Provide notification that the connection has been accepted.
6. Forgot Password isn’t given any code in the current system. Provide resetting password option.
7. Provide chat option for the users.
BIBLIOGRAPHY AND REFERENCES

https://www.researchgate.net/

https://www.mendeley.com/

https://www.waldenu.edu/connect/newsroom/publications/articles/2010/01-
research-challenges

https://www.twitter.com

https://en.wikipedia.org/wiki/SQLite

https://en.wikipedia.org/wiki/Groovy_(programming_language)


https://www.playframework.com/

https://aws.amazon.com/

https://en.wikipedia.org/wiki/Software_verification_and_validation

https://en.wikipedia.org/wiki/Android_Studio