**Campus Recruitment System**

**TECHNICAL REPORT**



**SUBMITTED BY**

Syed Ahmad Areeb Zaidi (2018-ag-8343)

Mubeen Khan (2018-ag-8317)

**ADVISED BY**

Dr. Hassan Tariq

**A TECHNICAL REPORT SUBMITTED IN PARTIAL FULFILLMENT OF REQUIREMENT FOR THE DEGREE OF**

*Bachelors of Sciences*

*IN*

*Software Engineering*

**DEPARTMENT OF COMPUTER SCIENCE**

**FACULTY OF SCIENCES**

**UNIVERSITY OF AGRICULTURE FAISALABAD**

**DECLARATION**

I hereby declare that the contents of the report **Campus Recruitment System** are project of my own research and no part has been copied from any published source (except the references). I further declare that this work has not been submitted for award of any other diploma/degree. The university may take action if the information provided is found false at any stage. In case of any default the scholar will be proceeded against as per UAF policy.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Syed Ahmad Areeb Zaidi

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Mubeen Khan

**CERTIFICATE**

To,

The Controller of Examinations,

University of Agriculture,

Faisalabad.

The supervisory committee certify that **Syed Ahmad Areeb (2018-ag-8343)** and **Mubeen Khan (2018-ag-8317)** has successfully completed their project in partial fulfillment of requirement for the degree of BS. Software Engineeringunder our guidance and supervision.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr. Hassan Tariq

Supervisor

**ACKNOWLEDGEMENT**

I thank all who in one way or another contributed in the completion of this report. First, I thank to ALLAH ALMIGHTY, most magnificent and most merciful, for all his blessings. Then I am so grateful to the Department of Computer Science for making it possible for me to study here. My special and heartily thanks to my supervisor, *Dr. Hassan Tariq* who encouraged and directed us. His challenges brought this work towards a completion. It is with his supervision that this work came into existence. For any faults I take full responsibility. I am also deeply thankful to my informants. I want to acknowledge and appreciate their help and transparency during my research. I am also so thankful to my fellow students whose challenges and productive critics have provided new ideas to the work. Furthermore, I also thank my family who encouraged me and prayed for me throughout the time of my research. May the Almighty God richly bless all of you

**ABSTRACT**

Campus Recruitment System is a mobile application developed using state of the art technology, React Native, a strong, powerful framework designed by Facebook.

Mostly in Pakistan, graduates have to wander right after their graduation for job hunting. This process is very time consuming. It takes the real exam of man’s patience. Mostly graduates lose their hope and make them busy in other fields rather than their own study of interest.

Campus Recruitment System provides facility to the students to find job remotely, without facing hardships of the society. This will help students to secure their future even during their degree by finding a job. Job fare trend is not very common in Pakistani Universities. So this application will play a great role by providing job opportunity to students. This product will bring ease in student’s life to find a better job. The product is very easy to understand and easy to use.

This application is designed on Figma, developed using React Native and it has firebase as its backend service provider. Firebase is developed by Google which provides many cloud services, databases, social authentications etc. This project is developed using rapid application development model which provides flexibility to the developers to build and run the increments rapidly. User can see their output very soon if the project in being developed using this development process model.

Finally, this project is completed with all its predefined requirements and provides facility to students and company to apply and post jobs.

Table of Contents

[Chapter 1 - INTRODUCTION 1](#_Toc536663558)

[1.1 Background: 1](#_Toc536663559)

[1.2 Description: 1](#_Toc536663560)

[1.3 Problem Statement: 2](#_Toc536663561)

[1.4 Scope: 2](#_Toc536663562)

[1.5 Objectives: 2](#_Toc536663563)

[1.6 Feasibility: 3](#_Toc536663564)

[1.7 Requirements: 4](#_Toc536663565)

[1.7.1 Functional Requirements 4](#_Toc536663566)

[1.7.2 Non- Functional Requirements 5](#_Toc536663567)

[1.7.3 Hardware Requirements 5](#_Toc536663568)

[1.7.4 Software Requirements 5](#_Toc536663569)

[1.8 Stakeholders: 5](#_Toc536663570)

[Chapter 2 – MATERIALS & METHODS 9](#_Toc536663571)

[2.1 Process Model: 9](#_Toc536663572)

[2.2 Tools & Technologies 9](#_Toc536663573)

[2.3 Design: 9](#_Toc536663574)

[2.3.1 Use Case Diagrams: 10](#_Toc536663575)

[2.3.3 Sequence Diagram: 16](#_Toc536663577)

[2.3.4 Class Diagram: 19](#_Toc536663578)

[2.3.5 Data Flow Diagram: 20](#_Toc536663579)

[2.3.6 ER Diagram: 24](#_Toc536663580)

[2.3.7 Database Model: 24](#_Toc536663581)

[2.3.8 Architecture: 25](#_Toc536663582)

[Chapter 3 - RESULTS & DISCUSSION 27](#_Toc536663583)

[3.1 Testing: 27](#_Toc536663584)

[3.2 Test Cases: 27](#_Toc536663585)

[3.3 Conclusion: 29](#_Toc536663587)

[Chapter 4 - USER MANUAL 30](#_Toc536663588)

[References 31](#_Toc536663589)

List of Figures

[Figure ‎1.1 Stakeholders 8](#_Toc536625484)

[Figure ‎2.1 Agile Activities 9](#_Toc536625485)

[Figure 2.‎2 Use Case Diagram 12](#_Toc536625486)

[Figure 2.3 Sequence Diagram 19](#_Toc536625487)

[Figure 2.4 Class Diagram 20](#_Toc536625488)

[Figure ‎2.5 Context Diagram 22](#_Toc536625489)

[Figure ‎2.6 Level 0 DFD 22](#_Toc536625490)

[Figure 2.7 Level 1 DFD 23](#_Toc536625491)

[Figure ‎2.8 Entity Relationship Diagram 24](#_Toc536625492)

[Figure 2.9 Database Model 25](#_Toc536625493)

[Figure ‎2.10 Applications's Architecture 26](#_Toc536625494)

[Figure ‎4.1 Signing in 30](#_Toc536625495)

List of Tables

[Table 2. 1: Add User 15](#_Toc536658797)

[Table 3. 1: User login Test Case 29](#_Toc536658805)

# Chapter 1 - INTRODUCTION

## 1.1 Background:

During a degree every student thinks that they will get a job after completing their degree and most of the students enroll in many certifications to gain skills and want to work during their degree period. It is a big task to find a job in a well-known well reputed company or institute because everyone who secures a degree from a university have no experience in the beginning so they are worried about their jobs and career. After graduating from the university, students face too much trouble in terms of finding a well-reputed organization to start their career. This is because there is a huge gap between students and the companies. Most of the companies hire their within the company because of experience concerns they are worried about new employees work and skills. So it is the need of the hour to fill up the gap. We need to create a platform for this concern so it can help to those company and students. We are designing this application for both company and students. We put all the possibilities that can make the process easier for the new students to search suitable jobs for them and for companies to hire the right and suitable talent. Application covers all the features for searching and posting jobs and places for the people. The application will fill the gap by acting as a bridge between these two entities.

## 1.2 Description:

The system will provide facility to the students for job hunting as they can search jobs for themselves according to their requirements like degree they get and skills they develop during the degree and from extra certifications as well as to the companies looking for the talented employees as it is a big task to hire some new and right talent for their business because in this era we have countless options for us that makes the process of hiring and searching good talent difficult. In this application we added features according to the requirements of companies and students. The students who wants to search for their job can make a profile and will put their qualifications and certification so they can get the right and exact one for the as well as the human resource managers or talent acquisition persons can also make profiles for their companies and can put the job requirements they are searching employees for. The details will be according to the position that will help them to hire right person for their desire position. They can modify their data as they can add and delete things in their profiles according to the information they required. Both of the entities will create and manage their profiles within the system. This will all help them both companies and students for securing the right target according to their need and requirement.

## 1.3 Problem Statement:

Joblessness is a big problem in under developed countries like Pakistan. Graduates suffer very much even after years of their degrees. Some have to work in other fields which are not even their area of knowledge. Many graduates like start giving tuitions to earn their living hood.

This project is a solution in this regard to provide a platform in a university. University will invite companies to build their profiles on this application and post jobs on this application. Students of the university will find an opportunity to find jobs and internships even during their degrees. This application can also be very beneficial to the students to start freelancing on this application. Outsourcing and all the opportunities to earn some living hood during study can be carried out using this application.

## 1.4 Scope:

Campus recruitment system is a mobile application (for both android and iOS) designed for the students of the universities who are graduated from the university or about to graduate for job purposes. Universities can purchase this application to give an opportunity to their students to get jobs according to the skillset and qualification.

This will create a link between University Students and recruiting companies. Both of the entities will maintain their profiles according to their need. Companies will post their jobs and the student with relevant expertise will be able to apply for those jobs. Companies will receive a collection of CVs of all the applicants after posting jobs. At the end, company can contact shortlisted students.

Only University Student will be allowed to join as Student using their registration number provided them at the time of admission. Campus will have an admin authority to manage the system. Any student providing wrong information or any suspected activity by the company will be restricted by the admin. In this case it is also secure and valid platform for the both companies and students.

## 1.5 Objectives:

The main goal of this application is to facilitate students to find a better job for them even during their graduation and after the completion of graduation. The application will also provide the functionality to modify data according to their need and to maintain their CVs in a professional way and apply for relevant jobs posted by the different human resource managers of the different companies. Companies registered on this portal will have a direct access to the newly graduates. They can see the profiles of the relevant students and approach them to apply for the vacancy they posted. They will receive the big amount of CVs from the relevant students and they can have shortlisted the exact one’s matches to their requirements and positions. The main objective of this application to provide an ease to the youngsters, new generation and new talent to fulfil their dreams towards securing a best job for them as in this time it is a big issue to search out something best for your position. On other hand it is also a facility for the companies to grow their business by hiring such fresh and new talent who are familiar to the technology as it is an era of technology and virtual world and it is a big need to hire the persons with the knowledge of new technology.

## 1.6 Feasibility:

A feasibility study is performed by a company when they want to know whether a project is possible given certain circumstances. Feasibility studies are undertaken under many circumstances – to find out whether a company has enough money for a project, to find out whether the product being created will sell, or to see if there are enough human resources for the project. A good feasibility study will show the strengths and deficits before the project is planned or budgeted for. By doing the research beforehand, companies can save money and resources in the long run by avoiding projects that are not feasible. There are many different types of feasibility studies; here is a list of some of the most common:

**1.6.1 Technical Feasibility** – All the technological requirements for this project are easily available in market. This can be done using various tools and technologies.

**1.6.2 Schedule Feasibility** – The given time span is far enough for current requirements of this project. This project can easily be done in within time.

**1.6.3 Economic Feasibility** – This project is very economical and can be performed using various free tools, technologies and free cloud and database services. This product will cost nothing from the users expect internet resources and a normal mobile phone.

**1.6.4 Cultural Feasibility** – Now in the modern era, everything is being digitalized and human acceptable. This product will easily be digestible by every part of the world. Now this is in trend in bloom in the whole world to find online applications like this to be a part of Global World.

**1.6.5 Legal/Ethical Feasibility** – Nowadays, every country is promoting digital products and making laws to bring ease for public and stop harmful or unethical cyber services. This product is totally ethical and acceptable in every country of the world.

**1.6.6 Resource Feasibility** – All the required resources for this product to be built are easily accessible like mobile devices, laptop, internet, required knowledge and helping resources.

**1.6.7 Operational Feasibility** – This product will provide a central place to companies and students to hire and get hired. This problem will successfully be resolved by this product.

## Requirements:

Both functional and nonfunctional requirements are detailed below.

### 1.7.1 Functional Requirements

Functional requirements of the project are described below.

**FR01:** Provide email and password to log in

|  |  |
| --- | --- |
| FR01-01 | System shall get Email and Password from user |
| FR01-02 | System should authenticate Email and password |
| FR01-03 | System shall let the user to log in if information is valid |
| FR01-04 | If information is not valid then system will display message to create an account. |

**FR02:** Create user account

|  |  |
| --- | --- |
| FR02-01 | System shall get Email and Password from user |
| FR02-02 | System shall send a confirmation email to the user to continue. |
| FR02-03 | If email is not verified by the user, system shall not allow user to log in. |

**FR03:** Enter Details

|  |  |
| --- | --- |
| FR03-01 | System shall get basic information from the user to continue. |
| FR03-02 | If user does not provide details, he shall not be allowed to move further until he fills all the required fields. |

**FR04:** Uploading CV/ Resume

|  |  |
| --- | --- |
| FR04-01 | System shall get CV or resume. |
| FR04-02 | Format of the file should be pdf or docs. System shall not accept any other file format. |
| FR04-03 | If file is not uploaded, user will be restricted from moving forward. |

**FR05:** Display Jobs

|  |  |
| --- | --- |
| FR05-01 | System shall display jobs available to the student as well as those on which student has applied |

**FR06:** Display Applicants

|  |  |
| --- | --- |
| FR06-01 | System shall display all the applicants to the company. |

**FR07:** Display Posted Jobs

|  |  |
| --- | --- |
| FR07-01 | System shall display all the posted jobs by the company. |

**FR08:** Report as spam

|  |  |
| --- | --- |
| FR08-01 | System shall allow all the authenticated users to mark any profile or job as spam if they find anything against rules and ethics. |

**FR09:** Display Spams

|  |  |
| --- | --- |
| FR09-01 | System shall display all profiles and jobs reported by company and students. |

**FR010:** Display Applicants

|  |  |
| --- | --- |
| FR10-01 | System shall allow admin to remove profiles of spam companies and students. |

### 1.7.2 Non- Functional Requirements

**NFR01:** System shall remain available 24/7 to its users.

**NFR02:** System shall have three types of users i.e., admin, student and company.

**NFR03:** System shall display a warning message if any required data is not provided by the user.

### 1.7.3 Hardware Requirements

Minimum Android SDK version: Android Marshmallow (Android 6)

RAM: 2GB or more

### 1.7.4 Software Requirements

Operating System: Android

## Stakeholders:

This project is dependent on basically three stakeholders, student, company and admin. Here admin can be defined as University administration. University is the core stakeholder in this regard to provide a central space to both companies and students. The other two stakeholders are dependent on the first.

# Chapter 2 – MATERIALS & METHODS

## 2.1 Process Model:

Campus Recruitment System is developed using Agile methodology. Agile model is Rapid Application Development model. It is a type of incremental model. In Agile model the components or functions are developed in parallel as if they were mini projects. The developments are time boxed, delivered and then assembled into a working prototype. This can quickly give the customer something to see and use and to provide feedback regarding the delivery and their requirements.

*Figure 2.1 Agile Activities*

## 2.2 Tools & Technologies

This project is developed using React Native as front end firebase as its backed server provider. Firebase cloud firestore is the database providing database service in this regard.

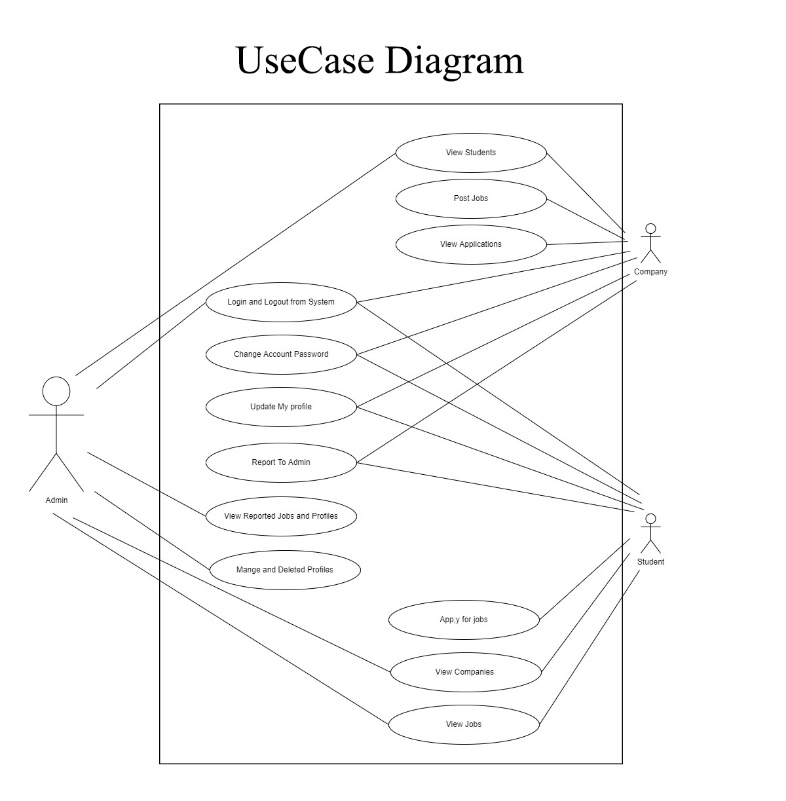
React Native is state of the art technology developed by Facebook. It is a hybrid application development tool used for developing Android and IOS applications with minimal code. No separate development is required in the modern era. React Native runs on both operating systems with single code.

Firebase is developed by Google. It provides many services like storage and cloud services. Here in this project. Firebase authentication is used for authenticating users and Cloud Firestore is used as database. Cloud storage is used to store files.

Visual Studio Code is the integrated development environment used for developing the whole project.

## 2.3 Design:

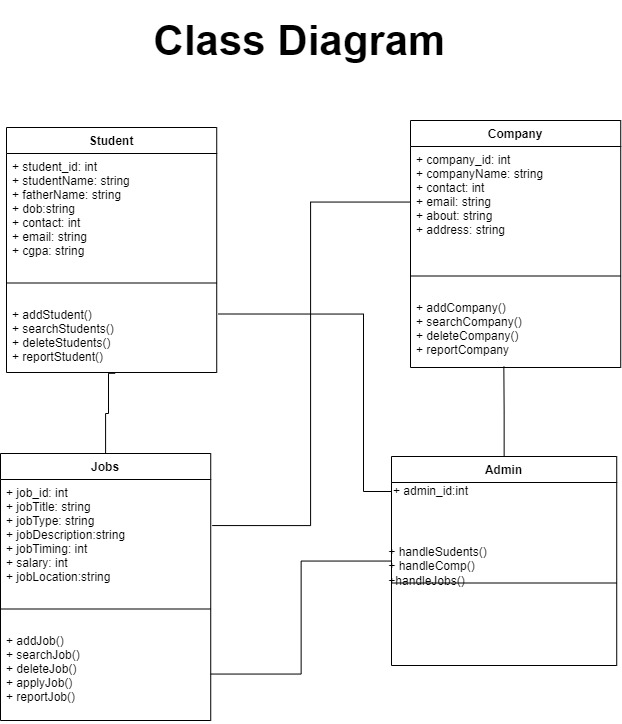
### 2.3.1 Use Case Diagram:





*Figure 2.2 Use Case Diagram*

### 2.3.4 Class Diagram:

****

*Figure 2.4 Class Diagram*

### 2.3.5 Data Flow Diagram:

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. A Data Flow Diagram (DFD) is traditional visual representation of the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or combination of both.

It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system. [3]

It is usually beginning with a context diagram as the level 0 of DFD diagram, a simple representation of the whole system. To elaborate further from that, we drill down to a level 1 diagram with lower level functions decomposed from the major functions of the system. This could continue to evolve to become a level 2 diagram when further analysis is required. Progression to level 3, 4 and so on is possible but anything beyond level 3 is not very common. Please bear in mind that the level of details for decomposing particular function really depending on the complexity that function. For further reading use the link given below:

<https://www.visual-paradigm.com/guide/data-flow-diagram/what-is-data-flow-diagram/>

#### DFD Diagram Notations

#### External Entity

An external entity can represent a human, system or subsystem. It is where certain data comes from or goes to. It is external to the system we study, in terms of the business process. For this reason, people used to draw external entities on the edge of a diagram.

cust

#### Process

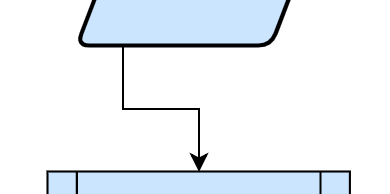
A process is a business activity or function where the manipulation and transformation of data takes place. A process can be decomposed to finer level of details, for representing how data is being processed within the process.   
process

#### Data Store

A data store represents the storage of persistent data required and/or produced by the process. Here are some examples of data stores: membership forms, database table, etc.   


#### Data Flow

A data flow represents the flow of information, with its direction represented by an arrow head that shows at the end(s) of flow connector.



**Context Diagram:**

****

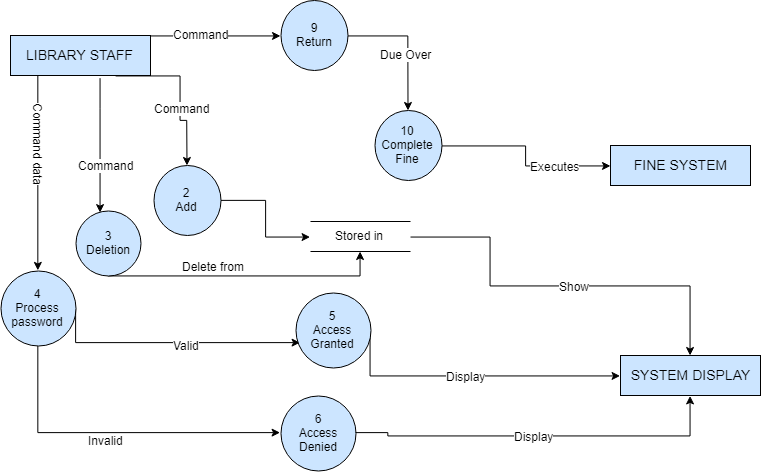
*Figure 2.5 Context Diagram*

**Level 0:**

****

*Figure 2.6 Level 0 DFD*

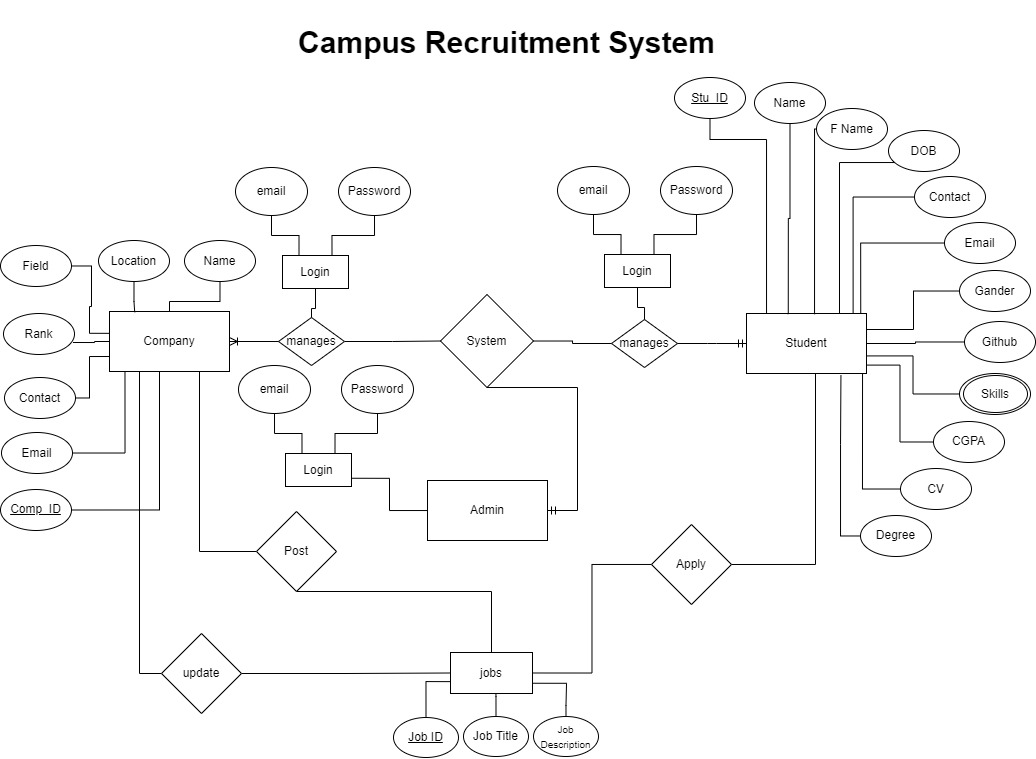
**Level 1:**

****

*Figure 2.7 Level 1 DFD*

**Also include Data Dictionary in this section.**

### 2.3.6 ER Diagram:

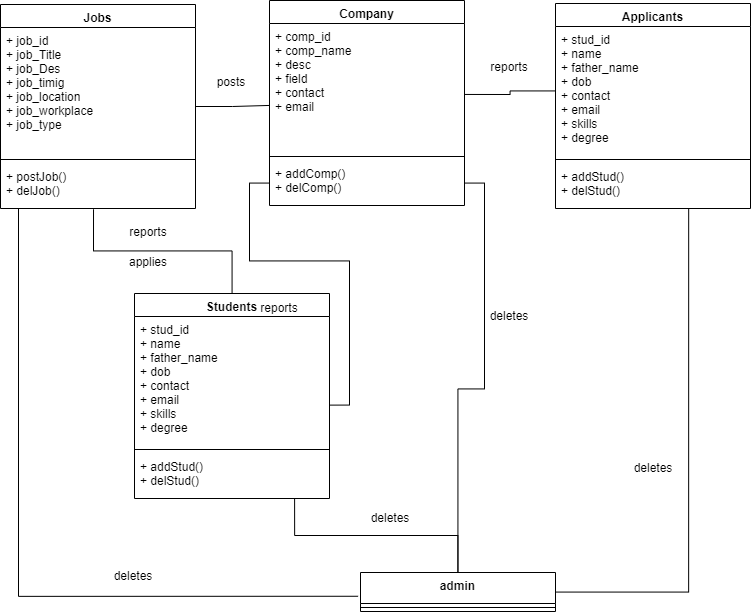


*Figure 2.8 Entity Relationship Diagram*

### 

### 2.3.7 Database Model:

Firebase is a no SQL database used for this project. It is more easy to handle and fast to use. It stores data in JavaScript Object Notation. Structure of database is shown below.



*Figure 2.9 Database Model*

### 2.3.8 Architecture:

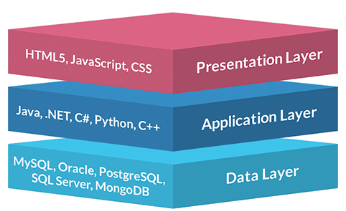
**3-Tier:**

This project is developed on 3-Tier architecture. First layer is presentation layer, second layer is all about logical portion and third layer is data storage layer. Their explanation is given below.

**Presentation Tier-** Front end of the Application is built using react native. React native uses Jsx format to build front end. Its logic is quite similar to that of HTML but Jsx is the extension of Javascript. React native provide within file styling format to give styling to the front end.

**Application Tier-** Core logic of the application is written in Javascript and Node js which is a strong framework of javascript. It makes the application to work smoothly and data to flow through the application properly.

**Data Tier-** This layer is all about data and database. It has firebase as its database which allows node js to send and get requests of data to the database Cloud Firestore. It also authenticates user before login.

****

*Figure 2.10 Application Architecture*

# Chapter 3 - RESULTS & DISCUSSION

In this chapter discuss overall performance or all functional and non-functional requirements you listed in chapter no. 1 as this section will verify the performance measures proposed for this project. For this software testing plays a vital role.

## 3.1 Testing:

Software testing is a process, to evaluate the functionality of a software application with an intent to find whether the developed software met the specified requirements or not and to identify the defects to ensure that the product is defect free in order to produce the quality product. In this regard, Test case writing is a major activity and considered as one of the most important parts of software testing. It is used by the testing team, development team as well as the management. If there is no documentation for an application, we can use test case as a baseline document. Below are some suggestions for writing good test cases:

## 3.2 Test Cases:

**Test Case: User Login:**

Table 3. 1: User login Test Case

|  |  |
| --- | --- |
| Test Case ID: | TC-1 |
| Test Case Title: | To verify the Login functionality of the application |
| Test Case Priority: | High |
| Requirement: | User Login |
| Test Description: | This test will verify the user login process. |
| Test Date: | 06/02/2022 |
| Pre-Conditions: | 1. Run the application.  2. Click Sign in button. |
| Dependencies: | Internet Availability |
| Test Steps: | 1. Enter Valid user name and password and click Login  2. Click Sign Out  3. Without entering user name click sign in  4. Without entering password click sign in  5. Enter wrong password or user name and click sign in |
| Test Data | Email id and password of user |
| Expected Results: | 1. System should open home page.  2. Login page should be displayed.  3. An error message should be shown to enter user name  4. An error message should be shown to enter password  5. Error message should be shown to enter correct password and user id |
| Actual Results: | As above |
| Post Conditions: | System shows Dash board page of signed in user. In case of unauthorized sign in attempt system shows the message “Invalid username/password”. |
| Status: (Pass/Fail) | Pass |
| Other Comments: | None |

Test Case: User Data Sending:

Table 3.2: User Data Sending Test Case

|  |  |
| --- | --- |
| Test Case ID: | TC-2 |
| Test Case Title: | To verify the functionality of sending user’s data to the data base |
| Test Case Priority: | High |
| Requirement: | User Data Sending |
| Test Description: | This test will verify that the user can send data to the database right after signing up.. |
| Test Date: | 06/02/2022 |
| Pre-Conditions: | 1. Run the application.  2. Sign Up successfully. |
| Dependencies: | Internet Availability |
| Test Steps: | 1. Enter all the required data.  2. Click next button  3. Without entering data, click next button  4. Without entering a few fields, click next button. |
| Test Data | Email id and password of user |
| Expected Results: | 1. System should enable next button.  2. Home page should be displayed.  3. An error message should be shown to fill the fields and then disable next button.  4. System should disable next button and show error message. |
| Actual Results: | As above |
| Post Conditions: | System shows Dash board page of signed in user. |
| Status: (Pass/Fail) | Pass |
| Other Comments: | None |

**Test Case: Job Apply:**

Table 3.3: User Job Apply Test Case

|  |  |
| --- | --- |
| Test Case ID: | TC-3 |
| Test Case Title: | To verify the functionality of Applying for a job |
| Test Case Priority: | High |
| Requirement: | Job Apply |
| Test Description: | This test will verify that the user can apply on available jobs. |
| Test Date: | 06/02/2022 |
| Pre-Conditions: | 1. Run the application.  2. Sign in successfully. |
| Dependencies: | Internet Availability |
| Test Steps: | 1. Click on an available job.  2. Click on apply job. |
| Test Data | Email id and password of user |
| Expected Results: | 1. System should open detailed page to show more about that job.  2. A success massage should be displayed to user.  3. That particular job should be displayed in applied job tab in user’s dashboard. |
| Actual Results: | As above |
| Post Conditions: | System shows all the available jobs and applied jobs to the user in his dashboard. |
| Status: (Pass/Fail) | Pass |
| Other Comments: | None |

**Test Case: Marking as Spam:**

Table 3.4: Marking as spam Test Case

|  |  |
| --- | --- |
| Test Case ID: | TC-4 |
| Test Case Title: | To verify the functionality reporting a job as spam. |
| Test Case Priority: | High |
| Requirement: | Marking as spam |
| Test Description: | This test will verify that the user can mark a job as spam. |
| Test Date: | 06/02/2022 |
| Pre-Conditions: | 1. Run the application.  2. Sign in successfully.  3. Open a job details page. |
| Dependencies: | Internet Availability |
| Test Steps: | 1. Click on the report button. |
| Test Data | Email id and password of user |
| Expected Results: | 1. System should display a massage and send that job to the admin’s dashboard. |
| Actual Results: | As above |
| Post Conditions: | System should display that job in admin’s panel.. |
| Status: (Pass/Fail) | Pass |

**Test Case: Report a Student:**

Table 3.5: Report Student Test Case

|  |  |
| --- | --- |
| Test Case ID: | TC-5 |
| Test Case Title: | To verify the functionality of reporting a student. |
| Test Case Priority: | High |
| Requirement: | Report a student. |
| Test Description: | This test will verify that the user can mark a student as spam. |
| Test Date: | 06/02/2022 |
| Pre-Conditions: | 1. Run the application.  2. Sign in successfully.  3. Open a applicants page. |
| Dependencies: | Internet Availability |
| Test Steps: | 1. Click on the report button. |
| Test Data | Email id and password of user |
| Expected Results: | 1. System should display a massage and send that job to the admin’s dashboard. |
| Actual Results: | As above |
| Post Conditions: | System should display that student in admin’s panel.. |
| Status: (Pass/Fail) | Pass |
| Other Comments: | None |

**Test Case: Data Delete:**

Table 3.6: Data delete Test Case

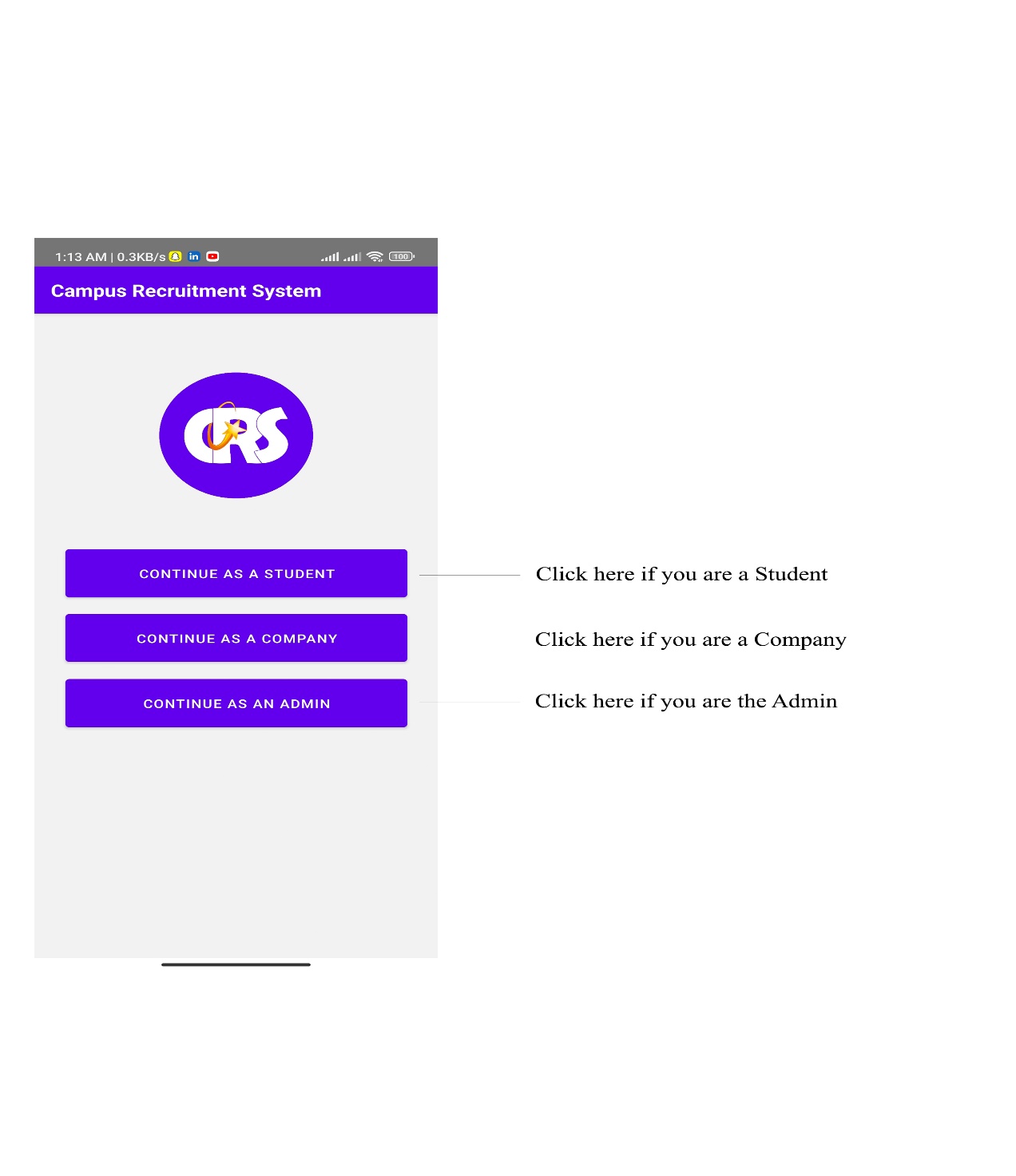
|  |  |
| --- | --- |
| Test Case ID: | TC-6 |
| Test Case Title: | To verify the functionality deleting job and students. |
| Test Case Priority: | High |
| Requirement: | Delete data |
| Test Description: | This test will verify that the admin can delete data from database using dashboard. |
| Test Date: | 06/02/2022 |
| Pre-Conditions: | 1. Run the application.  2. Sign in successfully.  3. Open a dashboard.  4. Delete a reported job.  5. Delete a reported student. |
| Dependencies: | Internet Availability |
| Test Steps: | 1. Click on delete button. |
| Test Data | Email id and password of user |
| Expected Results: | 1. System should display a massage that data has deleted. |
| Actual Results: | As above |
| Post Conditions: | System should display that student in admin’s panel.. |
| Status: (Pass/Fail) | Pass |
| Other Comments: | None |

## 3.3 Conclusion:

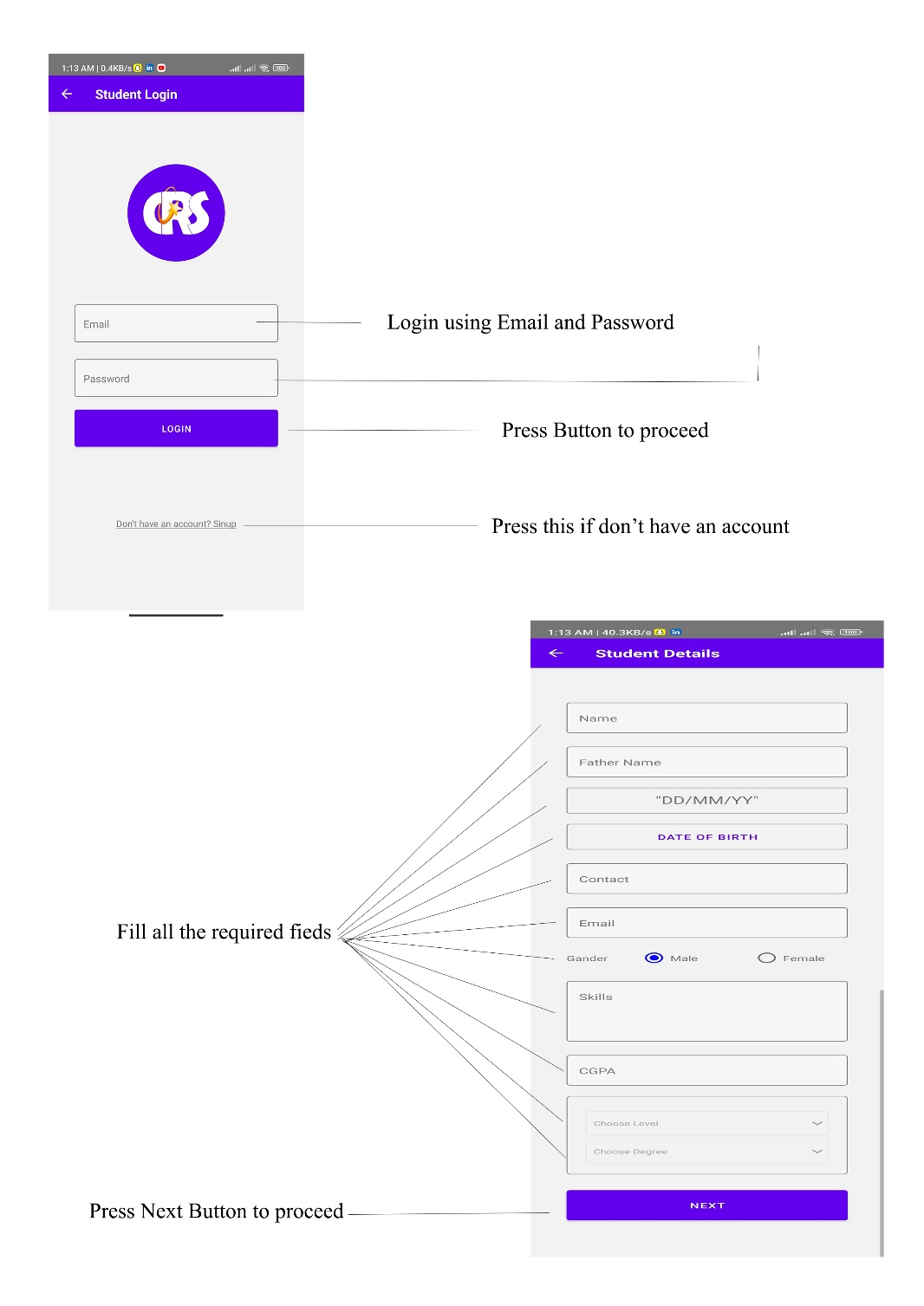
This project is completely tested against all the possible conditions. None of the test failed during testing. The overall performance of this application is tremendous. The project works smooth and fast. This is designed in such a way that even a person who have very less interaction will mobile or internet, can use and understand this application easily. Every end point is well defined will clear and understandable display messages against every action.

# Chapter 4 - USER MANUAL

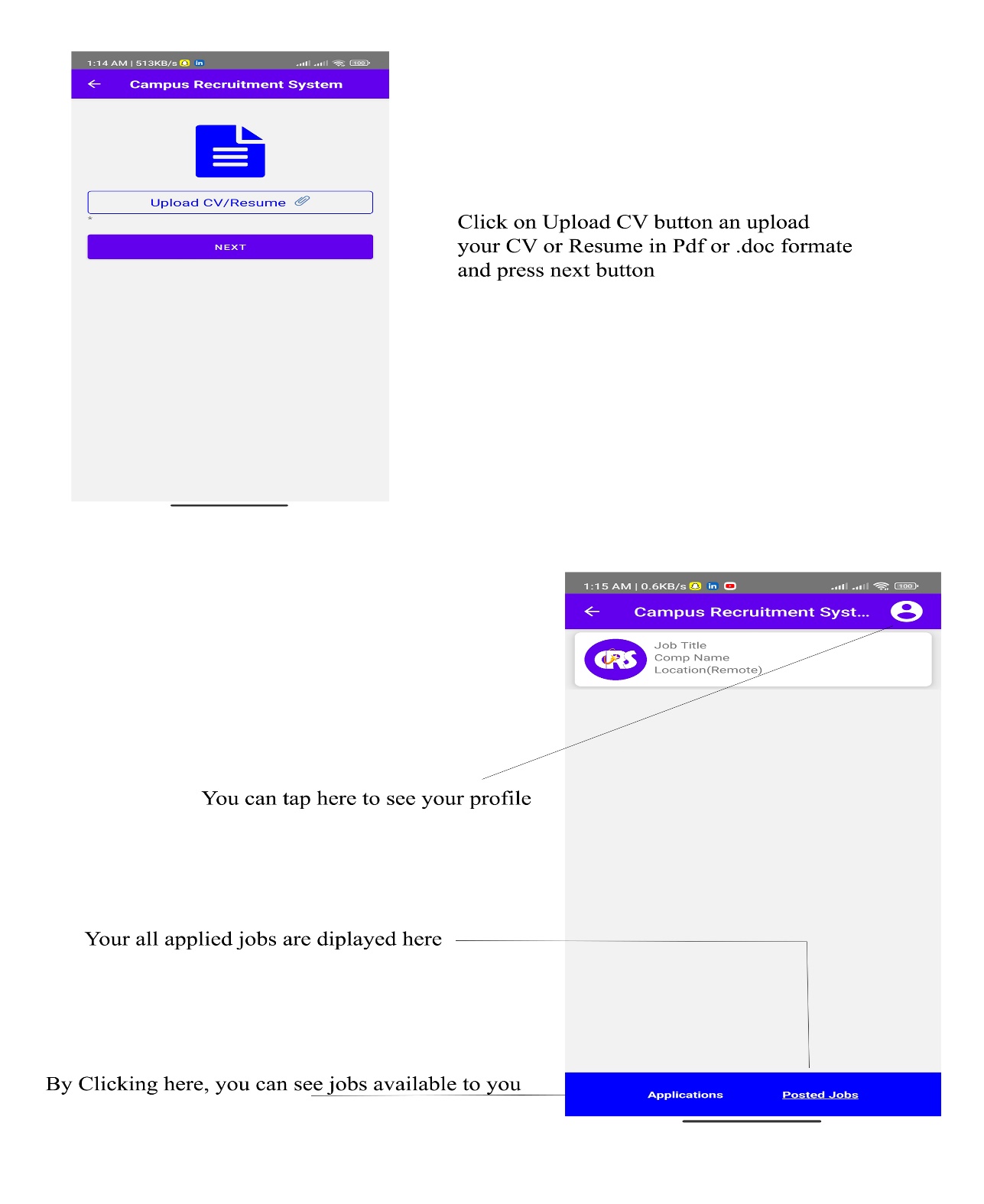
User manual of Campus Recruitment System is attached below.



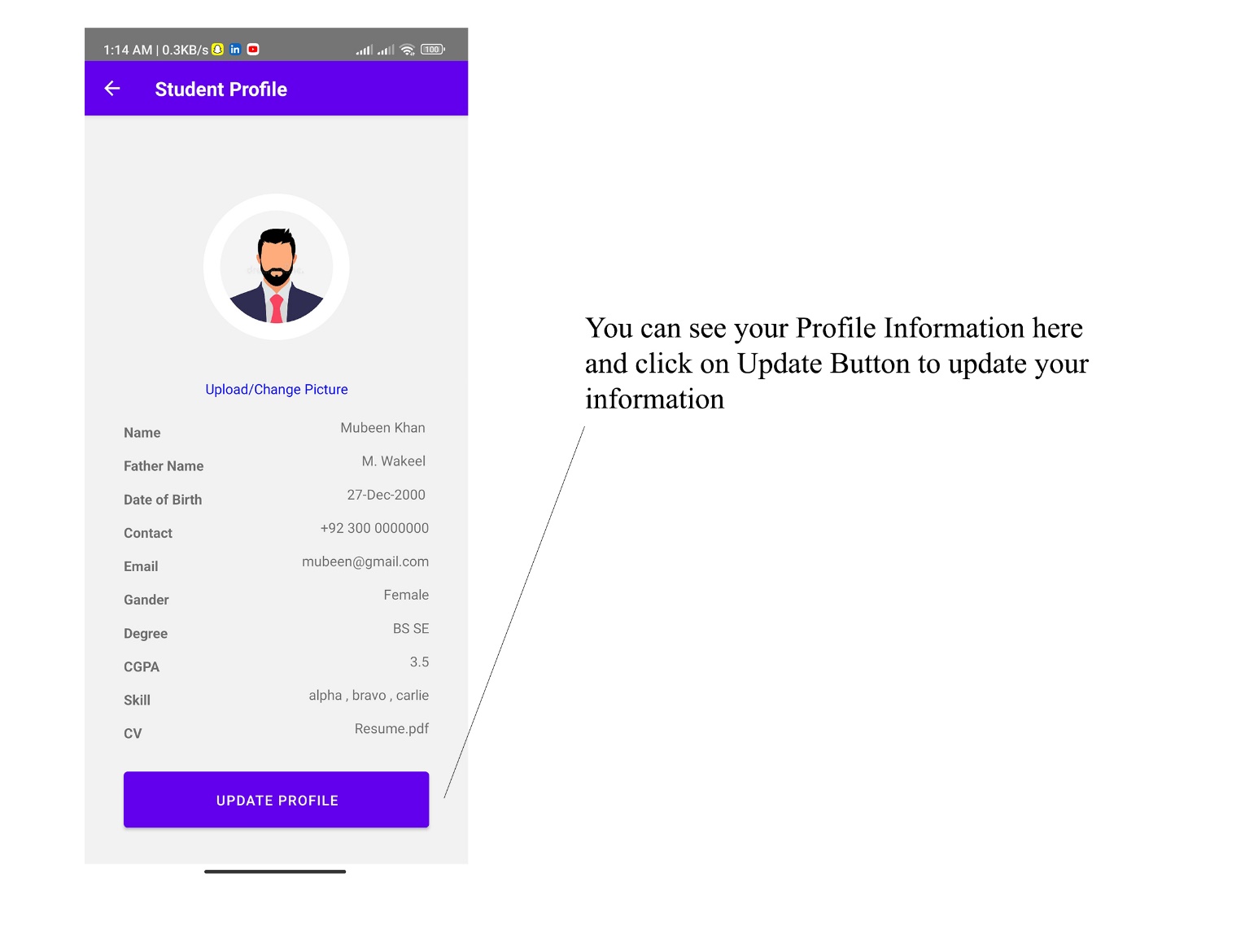
*Figure 3.1 Welcome Screen*



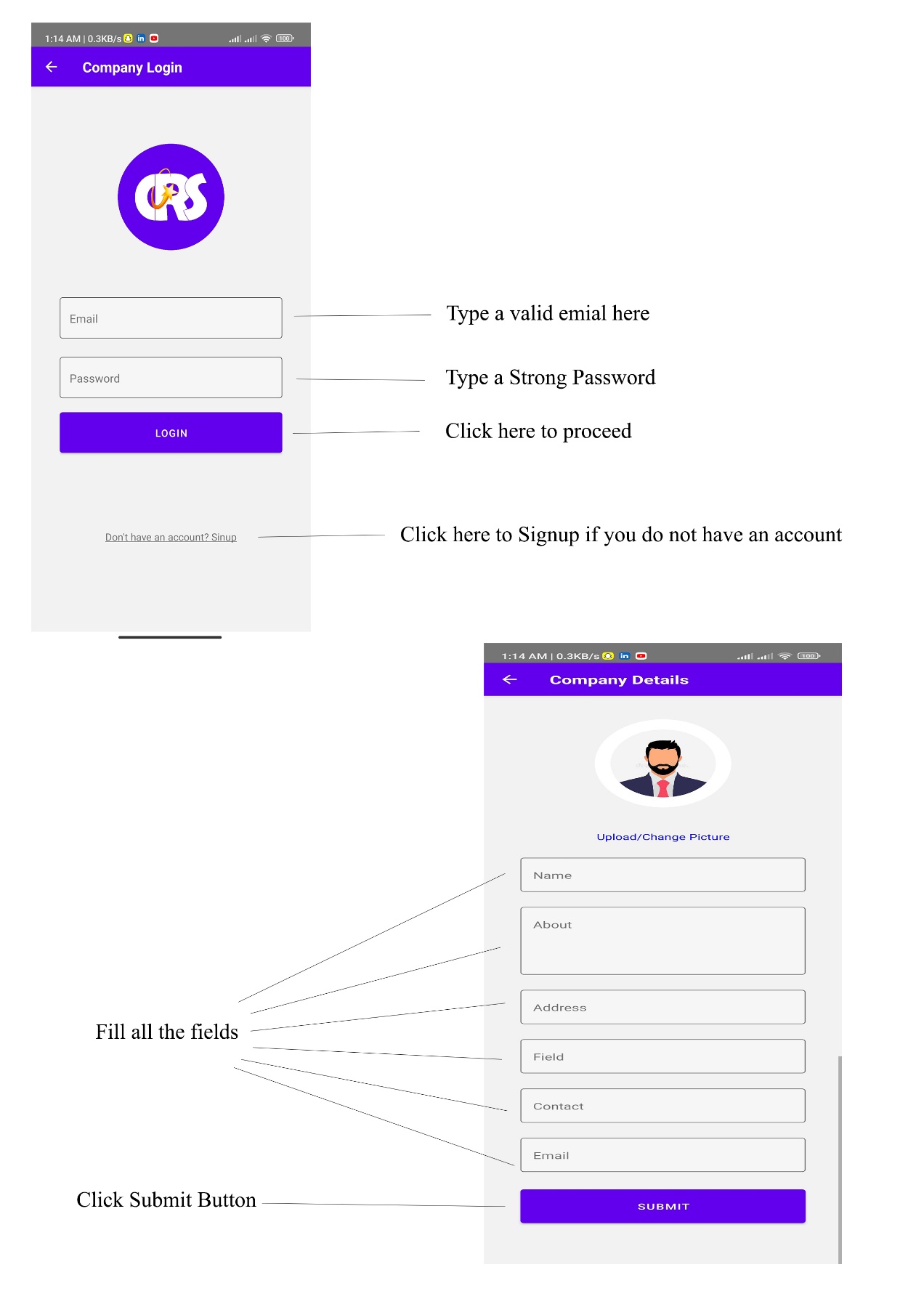
*Figure 3.2 Login and fill details as student*



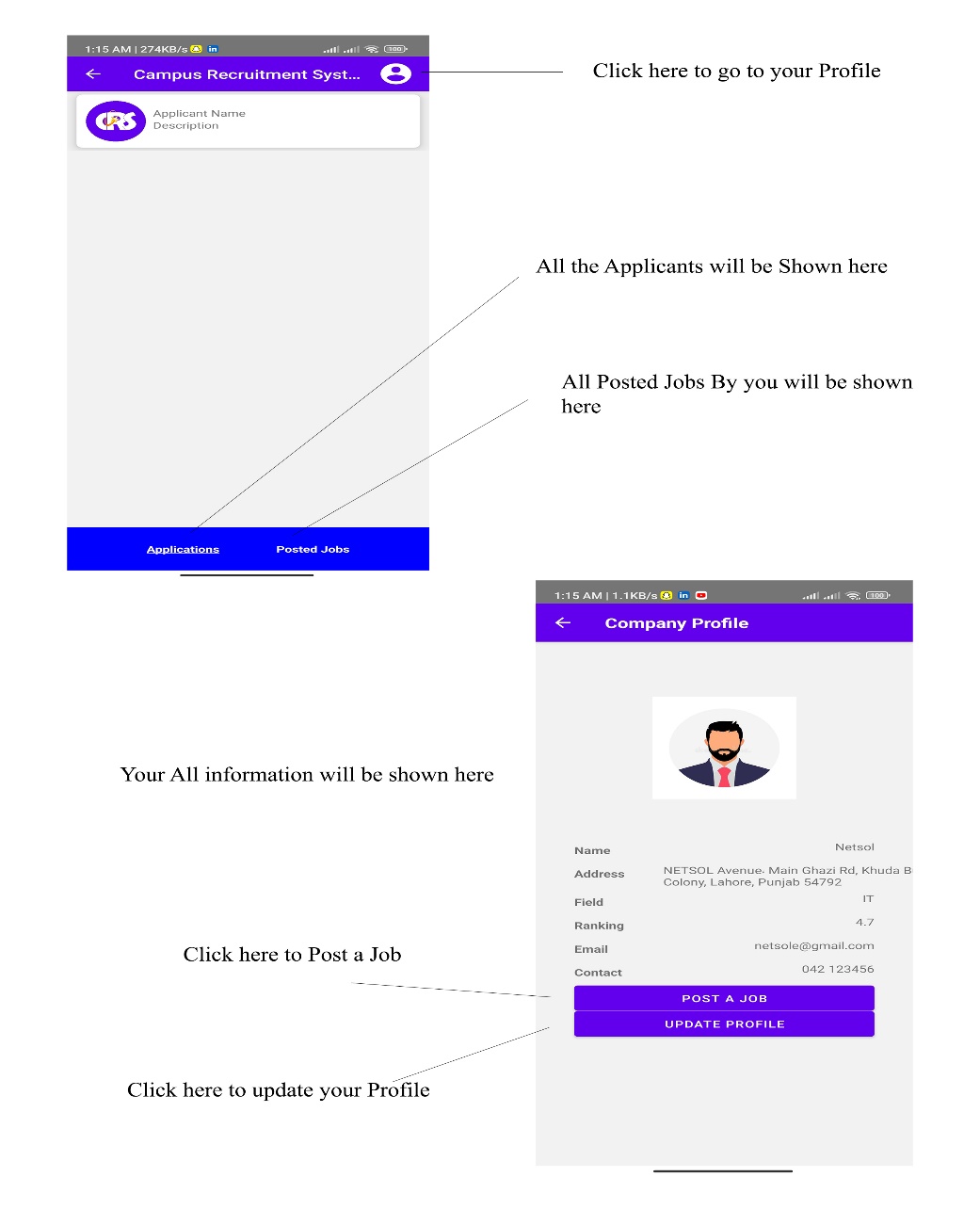
*Figure 3.3 Upload resume and home Screen*



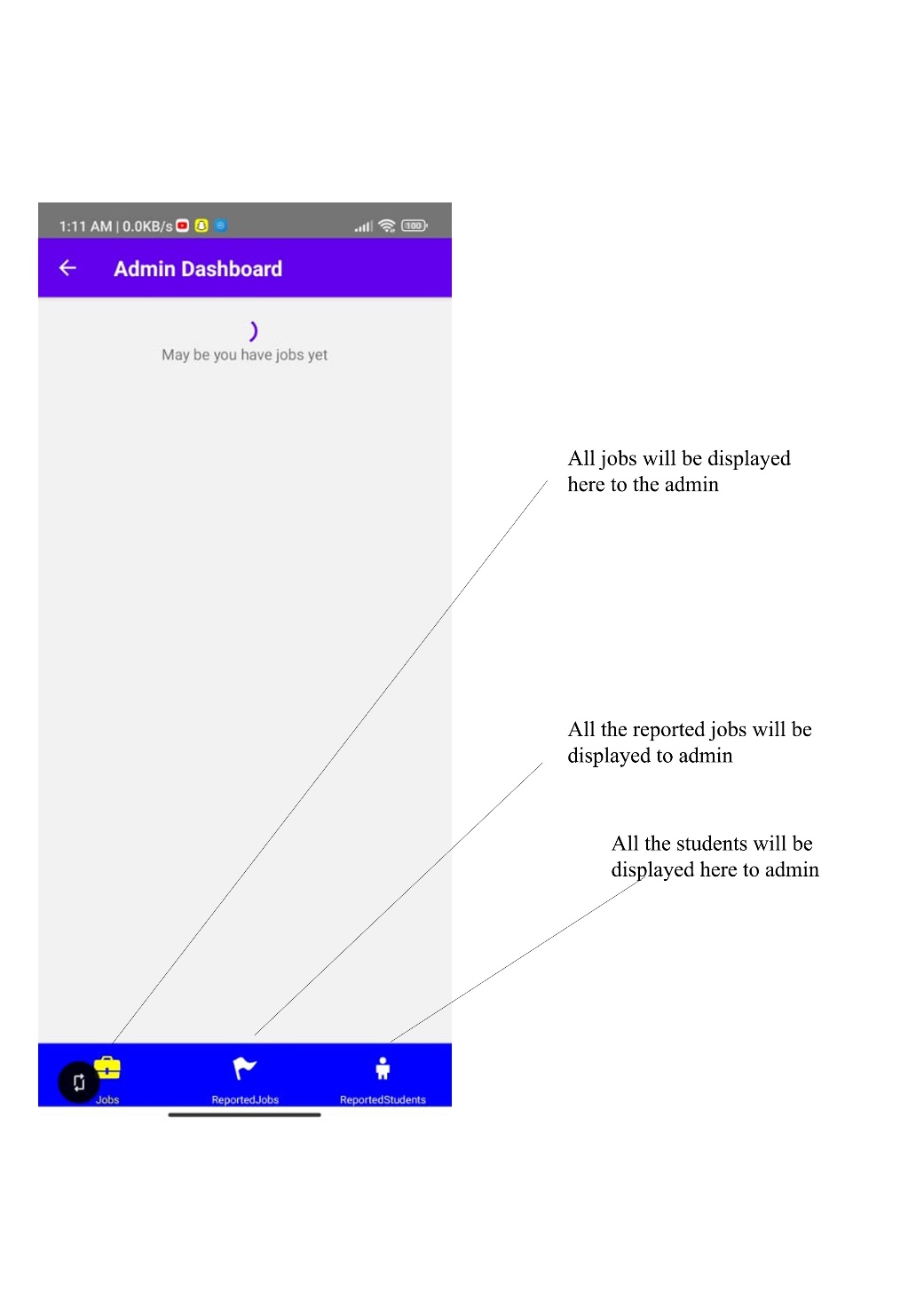
*Figure 3.4 Student Profile*



*Figure 3.5 Login as Company and fill details*



*Figure 3.6 Home Screen and Profile Screen of Company*



*Figure 3.7 Admin Home Screen*

**End**