SOFTWARE DESIGN SPECIFICATION

Student Internship Management System

Team Members: -

Nischal Gudehindler Lingeswara (2825960)

Savan Yeshwanth Rao (2784780)

Vadde Krishnasai (2805416)

Gangula Harshini (2845690)

1.0 Introduction

This section provides an overview of the entire design of the document. This document describes all data, functional and behavioral requirements for Student Internship Management System. The student Internship management system is an Android app that is an online internship management application in which a student can be able to apply for an internship by submitting his application form online through the application portal to the concerned roles. On the other end the admin could be able to go through the application form that is submitted by the student, review the details of the student. The admin can examine both the overall number of students and the number of students who applied for internships in the meanwhile.

The design contains 2 major parts: 1 The Student portal and 2: The Admin portal. Where users can interact according to the portal they would have logged in to. The student portal would generally have all his login credentials, forgot password, apply option, search option and results options. Coming to the Admin portal it is designed similar to the student portal, but here with extra benefits such as: the user can add new openings, can review the applicants, and can approve or deny the application.

1.1 Goals and objectives

The goal of the project remains same, the impact of using computer applications to the teaching and learning processes is becoming important issues and this project attempts to provide a solution to these problems. Thus, we aim to develop a prototype of an online internship management system that works by allowing the university to deploy a portal, to collect applicant's data, review that data, make a decision, and then continue to interact with applicants and reviewers. The goal of the system is to enable online interaction with applicants and their supervisors and automate the entire process; thereby eliminating a huge amount of paperwork and manual tasks.

1.2 Statement of scope

The scope of the system is to provide a friendly Android Application which is easy to navigate and at the same time provides sufficient depth and information about the system and how it works.

The scopes of the project are:

- Android application

- Users:
    - Students of any department.
    - Faculty Supervisors.
    - System Administrators.

1.3 Software context

The software used to design this entire project is Android Studio, which is an enterprise level application, that is used to develop mobile applications using programing languages like Java, Kotlin, C++ and JavaScript's. Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It will help you develop your app in a more productive way at scale. Everything you need to build on Android. Code and iterate faster than ever. Optimize code workflow. Build rich experiences. Code with confidence. Creating the best code.

1.4 Major constraints

The major constraint that would need to be focused on in this software is that we need to create 2 portal or 2 user interfaces separately, keeping in mind both the students and the admins requirements.

2.0 Data design

For any application the data design plays a key role in the backend that should able to store the data locally and private the data from using by other applications and also the cloud data base that will be helpful to access the localized data of the users when ever the admin or the admin permitted users can access the data for the development and growth of the users data in this model me will be using SQLite and Firebase to store the data locally and also into the cloud.

2.1 Data structures

In the project we are using two types of data structures Linear data structures and non-linear data structures. According to the application to store the user registration form and the also the login page we use the data query code to get the data and insert the data into the data table and also for the uploading of the resumes we use the non-linear data structure that will help us to store the data into the cloud that way we can access the data by the administrator and also by the companies.

2.2 Database description

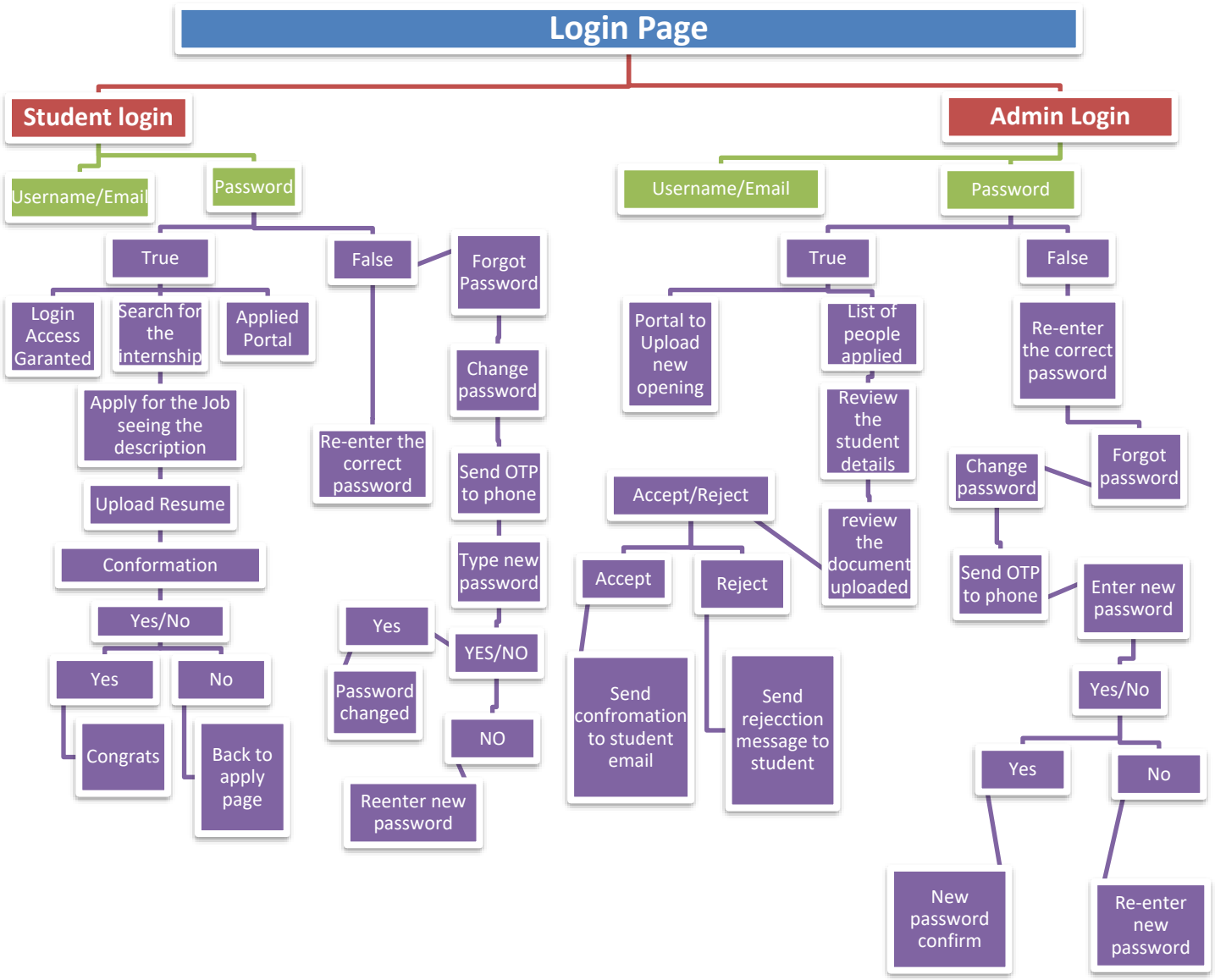
For now, we have two options for the database depending on the design that we are still planning on which is most flexible and feasible to use.

1. SQLite Database: which is an offline database, in which we can store the records in forms of tables and can be accessed depending upon the user requirements once we add all the required field such as first name, last name, email, phone etc.
2. Firebase Database: It is a real-time database which makes use of cloud-based NoSQL database, which will allow us to add all the information and store it and sync the database in real-time, unlike using the SQLite database.

3.0 Architectural and component-level design

A description of the software architecture is presented below, providing each tree and its branch and working of them is shown below in abstract. Each block has a specific function to execute and provide an output, depending on which the next step is and the flow is presented or displayed on the UI of the virtual phone or the phone connected though the WIFI and can be demonstrated and checked for the output of each input to the component that is been designed for both the admin and the student portal respectively.

3.1 Architecture diagrams.



3.2 Description for Components

The major components of the system are login page, student login, password , login access granted, Search for the Internship , Apply for the job seeing the description, Decision Yes/No , Applied Portal , False (Password Validation), Forgot Password, Password Validation, Admin Login, State Yes, Portal to upload new openings, List of people Applied, Review the student Details, Decision,

3.2.1 Component n description

1. **Login Page.**

   The login page contains two elements Student login and the Admin Login.

2. **Student Login.**

   When the User is student, he/she will select the Student login and get login into the application.

3. **Password**

   The password has a password validation function when the password is right then the user will go to the Login Access Granted portal if not he/she will use the Forgot password tab.

4. **Login Access Granted**

   This will allow the user to get access for the full end application to submit the resume in the application. This will give other options like Search for the internship and Applied Portal to get the Acceptance or rejection reply from the admin. At the same time that will show the updated application query also.

5. **Search for the Internship**

   This will allow the user to go to the application for the job by seeing the Description form where the user can see the listed jobs in the application.

6. **Apply for the Job Seeing the Description**

   In this page the user can see all the listed jobs by the administrator time to time and can apply to the jobs when the user applies for a job that will take him to the conformation process where the data is validated and processed for the Decision yes or no.

7. **Decision Yes/No**

   Here there will be two different steps will be performed to validate the conformation when the Decision if No that will take the user back to the Apply to the job seeing the Description that mean the User, student didn’t selected appropriate job requirements according to him that’s why the application automatically rejects the application if everything is okay it will proceeds to the Yes function where the system receive signal of conformation and prints a congrats message and take him/she to the Applied Portal.

8. Applied Portal

Where the students/users can see the applications status and review and edit and attach additional evidence according to the requirements of the administrative, that way users can keep track of their applications.

9. False (Password Validation)

When the user trying to get login with the wrong password then the system prompts he/she in to this page where he need to reset the password by going to forgot password but before doing it the system also lets give the user to re-enter the correct password option the user by himself need to come out of that loop and try to go to the Forgot password form.

10. Forgot password

When the user gets into this page that will prompt to change the password when the user tries to give the new password that will send an OTP to phone or email such that user can get access to type new password.

11. Password Validation

There are two stages where if the entered password is new the system takes the user directly to the Login Access Granted page if not that will be prompting a new page for the Re-enter new password message and takes the user to Type new password page.

12. Admin Login

Admin will be also having the same options in relation to the Validating the password the same loop will be applied for the admin when there is a wrong password loop discussed from 9 to 11. But when the admin gives a new password to the username email page.

13. State Yes

When the Admin password is written that will bring him to this page where the admin goes through the List of People Applied and Portal to Upload new Openings

14. Portal to Upload new Openings

In this the Admin can upload the new openings where the rest of the users/ students can see the listings and apply for the new openings accordingly.

15. List of people Applied

Where the admin can only see the listings of the applied peoples when the systems send an alert all the time for the successful completion of the application by the user/student.

16. Review the student details

Where the administrator can review the personal details and the skills that will be matching accordingly to the field of interest and the skills that they have accordingly to the job description.

17.Decision

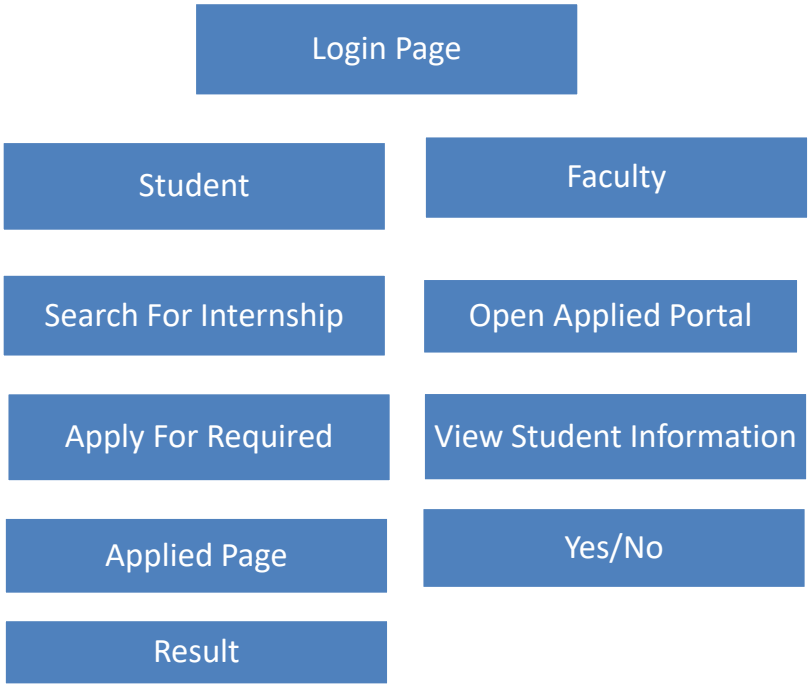
There will be a system performance program will be happening where the application is accepted that will send messages to students about the Acceptancy/Rejections. That will be full filling the list of people applied Inbox applications that way this is a set of look program.

3.2.1.1 Interface description

The Interface description is like the 4.1 Description of the user interface & 4.2 Interface design rules. In the Charts I explained each Input and Output functionality of the modules with the descriptions that can be followed in this Interface description (How a user can be input and the Output as the application mechanism is explained).

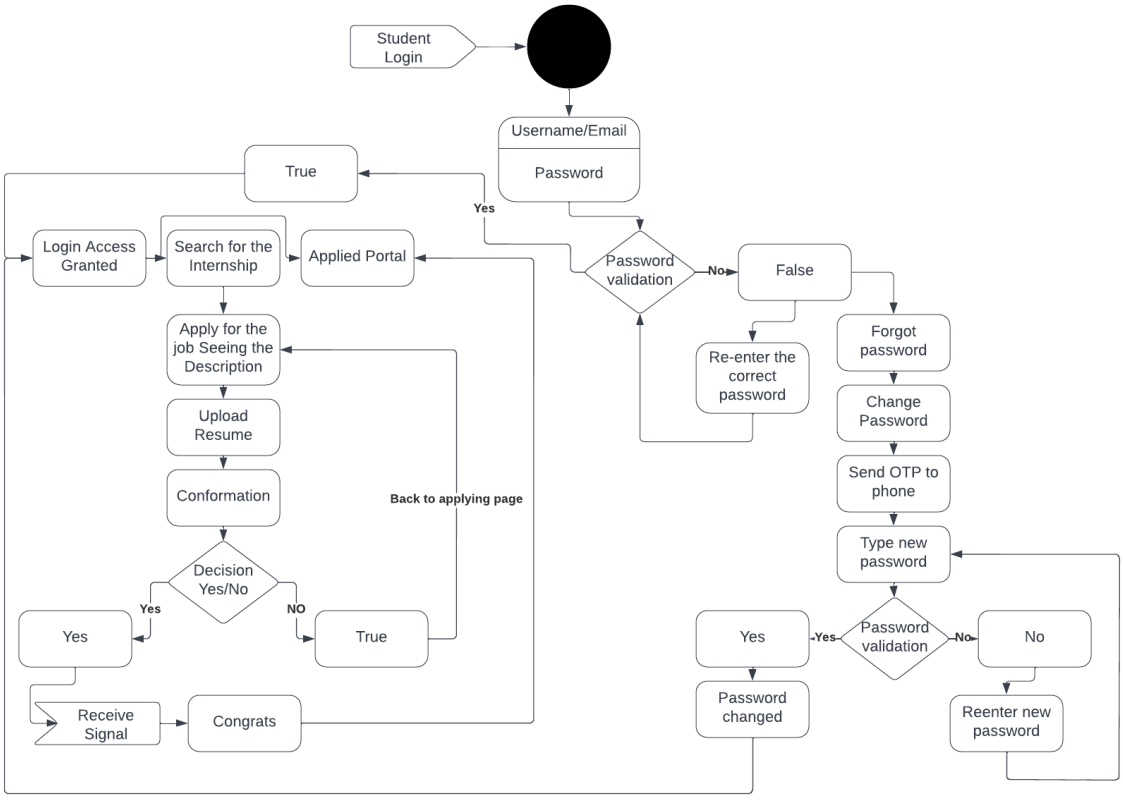
3.2.3.2 Static models

A reference to the data dictionary is provided. The dictionary is maintained in electronic

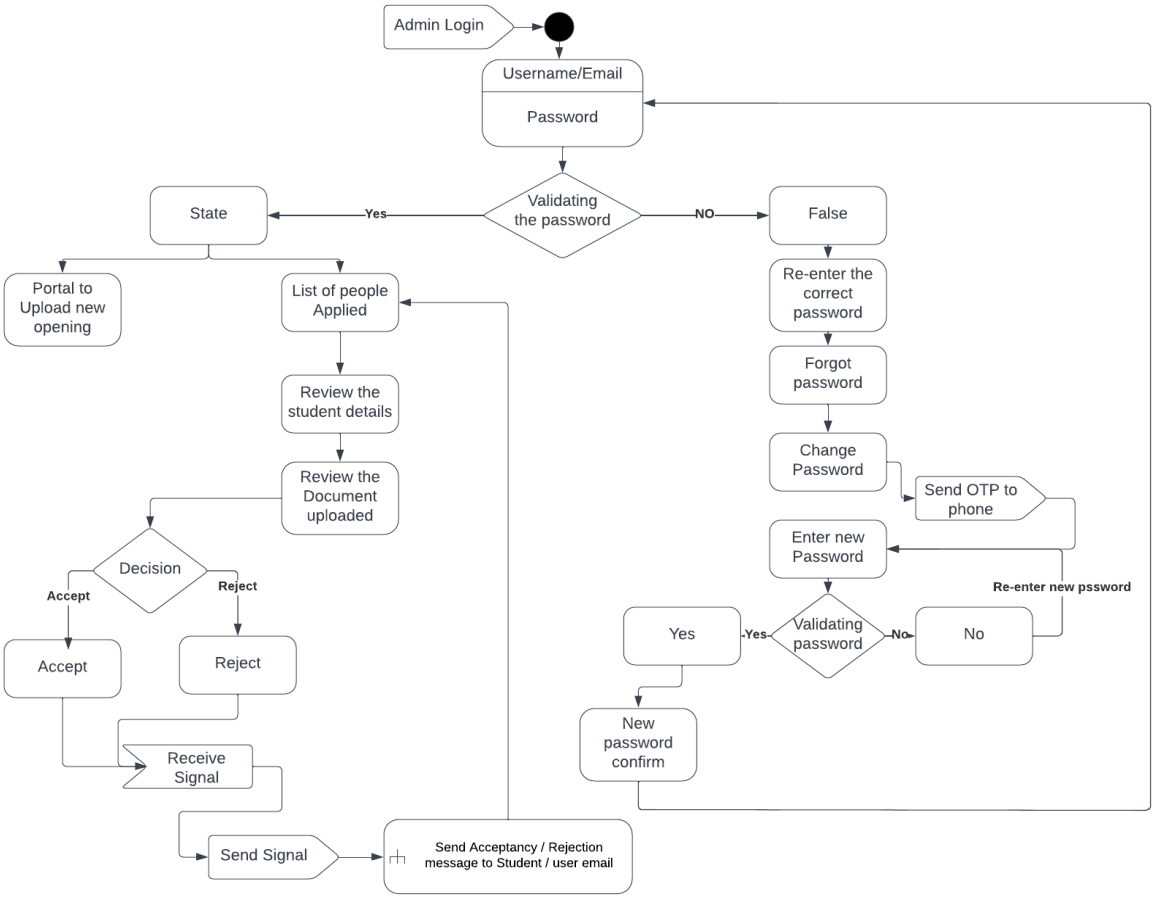


3.2.3.3 Dynamic models

Activity diagrams for the student login and the admin login page.



The above figure shows when the user tries to login through the Login page and submit the application.



The above figure shows the Activity diagram when the admin login and make a decision on the application that was received by the student/users

3.3 External Interface Description

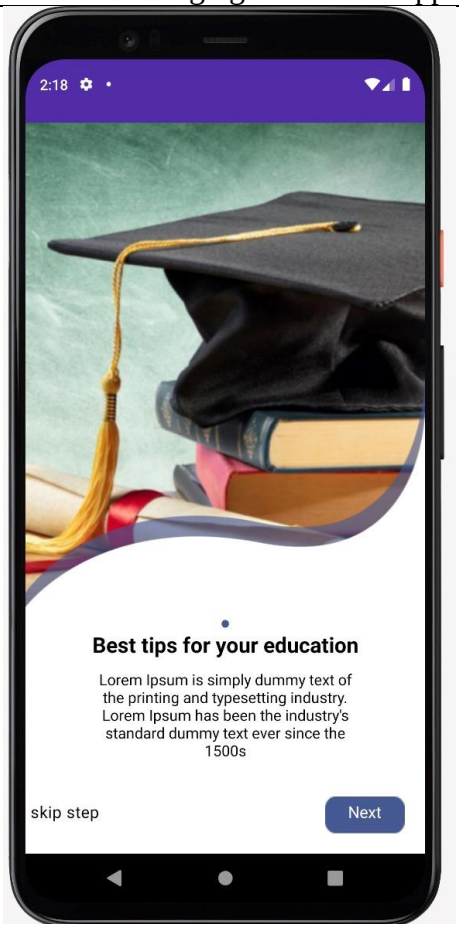
This application is only designed to be used within the college so we will be leaving a link for the git hub repository.

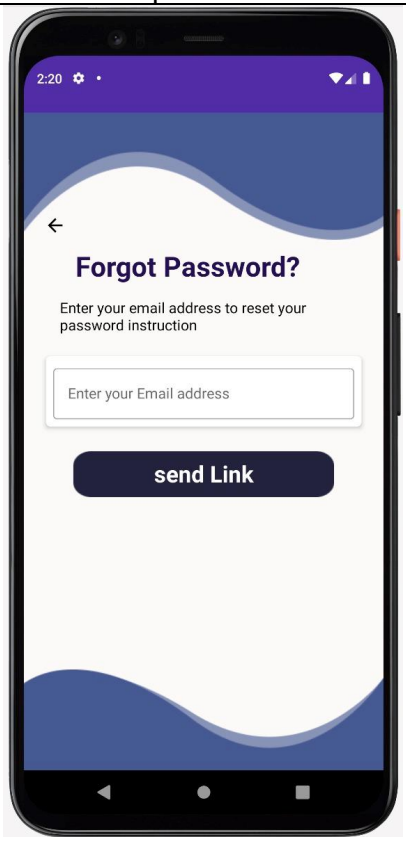
https://github.com/vaddekrishn/CIS-634/tree/Develop

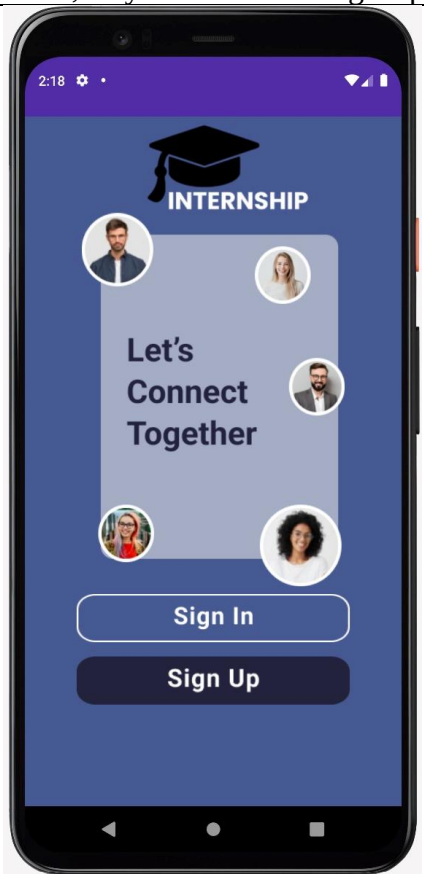
4.0 User interface design

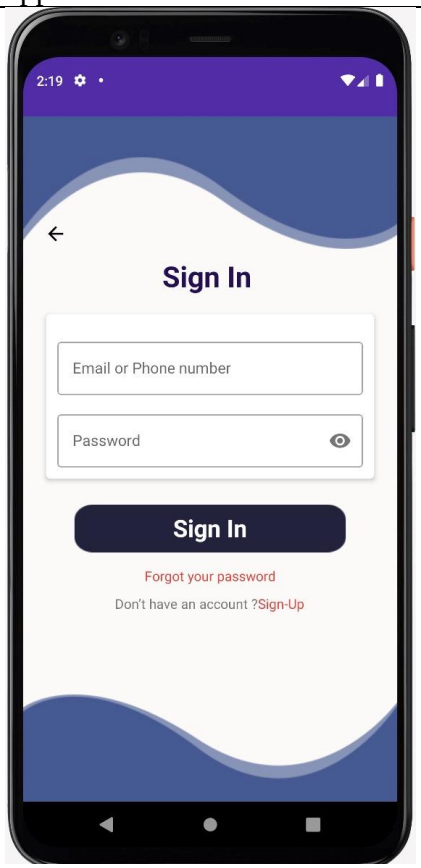
In this project there are 7 xml layouts which are main these will help the users to navigate through the application. We created this application user friendly such that the user can easily operate this application without any issues.

4.1 Description of the user interface & 4.2 Interface design rules

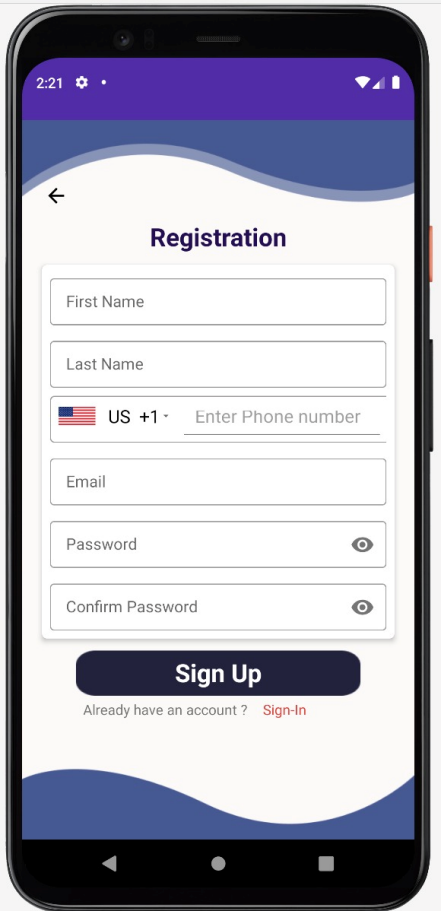
Design Rules	ID, User Input, Expected User Output, Description
ID	Home Activity
User Input	Display
Expected User Output	Displaying the slider view
Description	It shows us description of the app and provides best tips for education, like walkthrough guide for the application.
User interface	

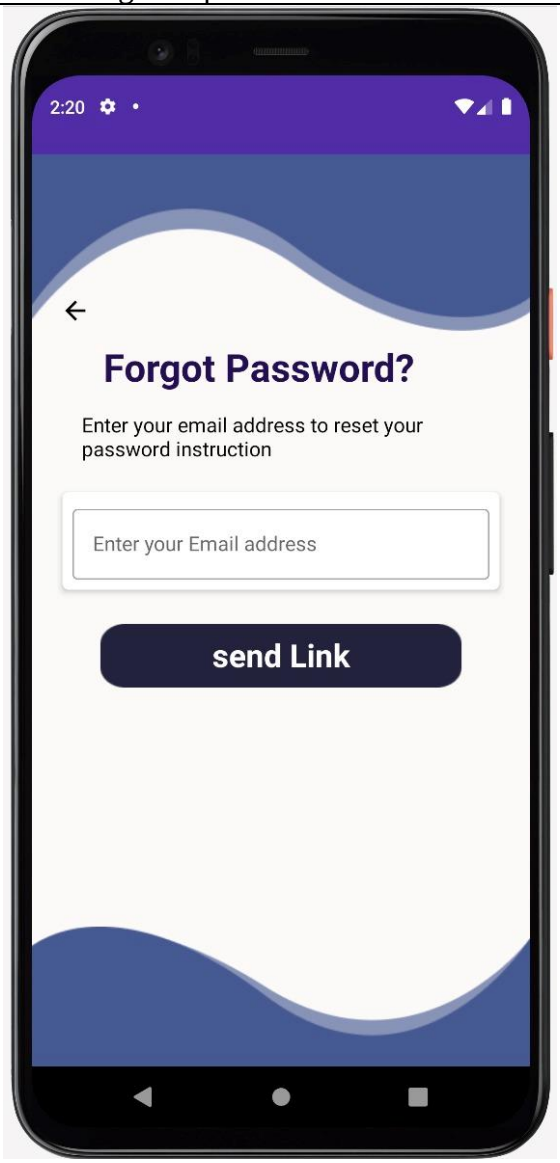
Design Rules	ID, User Input, Expected User Output, Description
ID	Forgot password
User Input	Click on forgot password in sign in menu
Expected User Output	User needs to enter email id to change his/her password
Description	In case if the user forget his/her password he or she can use this option to reset the password with the help of the register email id.
User interface	

Design Rules	ID, User Input, Expected User Output, Description
ID	Home Fragment
User Input	It as ON click functions for sign in or sign up
Expected User Output	Sign in goes to login page, sign up goes to registration page.
Description	When the user clicks on the sign in page, he/she can enter their email and password and login to the application if they are already a member. If not, they can select the Sign Up option and register as a new member.
User interface	

Design Rules	ID, User Input, Expected User Output, Description
ID	Login Activity
User Input	Email, password, sign in button, forgot password, sign up options.
Expected User Output	Enter into the application, Change password, Sign up
Description	From this page the user can start his/her search for the internship, can even change the password, or even can sign up if they are new to the application.
User interface	

Design Rules	ID, User Input, Expected User Output, Description
ID	Welcome Activity
User Input	Display
Expected User Output	Shows all the list of availability
Description	User can select which ever desired for his/her listings
User interface	

Design Rules	ID, User Input, Expected User Output, Description
ID	Register Activity
User Input	First name, last name, phone number, email, password, conform password, sign up/in
Expected User Output	New Account is created or can jump to login.
Description	By providing all the details he/she can register for using the application.
User interface	

Design Rules	ID, User Input, Expected User Output, Description
ID	Reset Password Activity
User Input	Email id
Expected User Output	OTP to registered email
Description	If the user forgets the password, he/she can send an OTP to email/phone and change the password.
User interface	

5.0 Restrictions, limitations, and constraints

Restrictions: -

- It is only for University Use only.

Limitations: -

- Only nearby Companies can be in the app.

6.0 Appendices

A more comprehensive understanding of the information that supplements the design specification.

6.1 Requirements traceability matrix

Student Panel:

- **Login/Registration**
- **Registration**
    - First name
    - Last Name
    - Mobile number
    - Email (Auto Filled)
    - Password
    - Confirm password
- **Login**
    - Email
    - Password
    - Forget Password
        - User can change the password by sending OTP to the email ID
- On the home screen, the user would be able to see the list of the company that provides the internship
- Users would be able to search for the internship using Search Bar
- Users would be able to filter the list of companies by filters such as
    - Mechanical Engineering
    - Civil Engineering
    - Computer and Science Engineering
    - Electrical engineering
    - Etc.
    - Other filters
        - On Campus

        - Off Campus
        - Full time
        - Part Time
- Following internship Details will be there:
    - Title of the Internship
    - Type of internship
        - On Campus
        - Off Campus
        - Full time
        - Part Time
    - Name of company
    - Company address
    - Internship Brief Detail
- Users would be able to see details of internships and apply for particular companies by submitting an application
    - Following application detail will be there that the students must fill
        - First name
        - Last Name
        - Email Id
        - Phone Number
        - Education Stream Such as
            - Mechanical Engineering
            - Civil Engineering
            - Computer and Science Engineering
            - Electrical engineering
        - Percentage
        - Resume
- Users would be able to see the list of companies where he has already applied
- Users also would be able to check the status of their application and whether it is approved by the faculty OR not.

Faculty (Admin) Panel:

- **Login**
    - Email
    - Password
    - Forget Password
        - User can change the password by sending OTP to the email ID
- On the home, screen Faculty would be able to see the list of Students who applied for a particular company
- Faculty will check the student list by applying the following filters
    - Mechanical Engineering
    - Civil Engineering
    - Computer and Science Engineering
    - Electrical engineering
    - Etc.

- Other filters
  - On Campus
  - Off Campus
  - Full time
  - Part Time

- The faculty will check the student's application detail and based on that, the application will be approved by the faculty
- Once the faculty approve OR reject a student's application the status of the application at the student end will be changed accordingly
- Faculty would be able to add the following things
  - Filter Category
  - Company Name
  - Company Location
  - Internship Title
  - Internship Brief Information
- All above things directly will be shown to the student's home screen

Providing Services:

- App Designing
- API development
- App Development
- Testing

6.2 Implementation issues

While developing the application we considered the following Implementation issues.

1. Software Fragmentation.

   Google releases several Android applications every year with varied new features and enhancements. These most recent iterations of Android, however, are not compatible with all Android devices and are still lagging in terms of acceptability. As an illustration, Android 6.0 Marshmallow currently has a bigger market share than Android 7.0 Nougat and Android 8.0 Oreo. As a result, when creating mobile apps, developers must consider Android apps that function with all Android operating system versions. Making the app utilize the features provided by both the most recent and older versions of the Android operating system is sometimes a challenge for them.

2. Device Fragmentation.

   There are a variety of Android smartphones on the market, each with a unique ecosystem and screen resolution. Making an app that works seamlessly on all devices is one of the main challenges faced by us

3. Testing Fragmentation.

   While testing considerations include device computability, user interface performance, and code efficiency. It requires in-depth expertise with testing and debugging tools.

4. User Interface Design Rules.

   While developing the application we customized many times to get an easy understandability for the user to access the application navigating them into right directions with a minimum amount of accessibility buttons and interferences that helps the user to keep in track what's happening throughout the application.

5. Programming Language.

   Even though there any many programing languages available we kept writing in java programming language that caused us many errors and more understanding of the programming language that made more lines of code that mad our application rapid development of app and works well with both native and cross platform apps. Many issues adopting the libraries while creating the application.

6. Security Issues

   Nevertheless, due to its openness and significant market dominance, Android is frequently the target of cyberattacks. Despite its security precautions, there is still a chance that virus will infiltrate programs and reroute users to other URLs without the designers' consent.