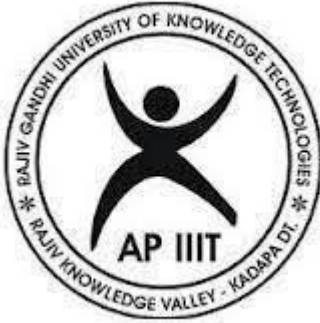


“STUDENT APP”
BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING



RGUKT
Rajiv Gandhi University of Knowledge Technologies
R.K.VALLEY

Submitted by
PALLELA JAGAN MOHAN -- R171149

Under the Esteemed guidance
of Mr. Satyanandaram N
RGUKT RK Valley.

DECLARATION

We hereby declare that the report of the B.Tech Mini Project Work entitled

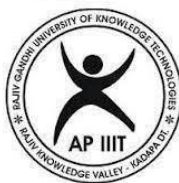
“STUDENT APP” which is being submitted to Rajiv Gandhi

University of Knowledge Technologies, RK Valley, in partial fulfillment of the requirements for the award of Degree of Bachelor of Technology in Computer Science and Engineering, is a bonafide report of the work carried out by us. The material contained in this report has not been submitted to any university or institution for award of any degree.

P JAGAN MOHAN – R171149

Dept. Of Computer Science and Engineering.

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES



RGUKT

(A.P.Government Act 18 of 2008)

RGUKT, RK VALLEY

Department of Computer Science and Engineering

CERTIFICATE FOR PROJECT COMPLETION

This is certify that the project entitled “**STUDENT APP**” submitted by
P JAGAN MOHAN—R171149, under our guidance and supervision for the partial fulfillment for the degree Bachelor of Technology in Computer Science and Engineering during the academic semester -2 2021-2022 at RGUKT, RK VALLEY. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any University or Institute for the award of any degree or diploma.

Project Internal Guide

Mr.N.Satya Nandaram

Assistant Professor

RGUKT, RK Valley

Head of the Department

Mr. Harinadha P

HOD Of CSE

RGUKT, RK Valley

Abstract

Student App [Study + Entertainment] which focus on the student education along with Entertainment. In this Mini project module, We are focusing on the student study materials , Question & Answers and Unit wise Test. Our application helps the student to maintain their time in studies and have fun while learning.

The app contain entire semester syllabus material and each subject includes content , Q & A and Unit Test modules. Not only for reference, this app will help the student recall and know how much they have learned by mock test. And so many features are awaiting to develop more accuracy for the student.

Index

1. Abstraction	4
2. Introduction	6
2.1 Purpose	6
2.2 Intended Audience	6
2.3 Product Vision	6
2.4 Technologies	6
3. React Native	7
4. JavaScript Runtime	8
5. development environment	10
6. Creating New Application	12
7. Running New Application	12
8. System in Context	15
7.1 Context Diagram	15
9. System-Wide Requirements	16
10. Function Requirements	16
9.1 Use Case Diagram	17
11. ER Diagram	19
12. Image References	21
13. Conclusion	28
14. Future Enhancements	28
15. References	28

Student_App SRS Document

Introduction:

This document has the requirements of providing simplified study material and simplify their doubts by giving questions and answers as well as they can take unit test by completion of the unit.

Purpose

The purpose of this document is focusing on the student study materials , Question & Answers and Unit wise Test. Our application helps the student to maintain their time in studies and have fun while learning. The app contain entire semester syllabus material and each subject includes content , Q & A and Unit Test modules. Not only for reference, this app will help the student recall and know how much they have learned by mock test.

Intended Audience:

The intended audience will be the students who can access the platform to get all the simplified study content and refers the questions and answers.

Product Vision:

The product vision is to providing simplified study content, which is user friendly and easily accessible. This Student app helps to provide the study material for all the students.

Technologies:

- React Native
- Java Script
- Visual Studio Code
- Local data base

REACT NATIVE

React Native is an open source JavaScript Mobile framework from Facebook specially designed to build native mobile apps for iOS and Android. React Native is based on ReactJS JavaScript library that helps to build the user interface for mobile platforms.

React Native can be directly used inside an existing iOS or android app or you can build a native app right from scratch. At present React Native is used with some popular apps like Facebook mobile app, Instagram, Pinterest, Skype, etc. Some important features of React Native that makes it a very popular mobile development app today are –

Cross Platform Support – To develop mobile apps you don't need a team expert in iOS and Android programming instead JavaScript developers who are enthusiastic to build apps can use React Native to build native apps without having to learn Kotlin or Java for Android and Swift or Objective-C for iOS apps. You can write one common code and React Native will take care of displaying it in iOS and Android.

React Native Components – React Native offers native components like View, Text, and Image that are changed to the iOS or Android native UI.



React Native Release – The first version of React Native was released by Facebook in the year 2015 and from there onwards they are updating and maintaining it. It has gained a lot of popularity post the release as it is one of the top most frameworks used for mobile apps development. As per React Native *official website*, in 2018 React Native has been the 2nd highest number of contributors for any repository in GitHub. Today, React Native is supported by contributions from individuals and companies around the world including Callstack, Expo, Infinite Red, Microsoft and Software Mansion. The Facebook community has been very active and updates the projects regularly with new updates across platforms.

JAVA SCRIPT

JavaScript Runtime

When using React Native, you're going to be running your JavaScript code in two environments:

- In most cases, React Native will use [JavaScriptCore](#), the JavaScript engine that powers Safari. Note that on iOS, JavaScriptCore does not use JIT due to the absence of writable executable memory in iOS apps.
- When using Chrome debugging, all JavaScript code runs within Chrome itself, communicating with native code via WebSockets. Chrome uses [V8](#) as its JavaScript engine.

While both environments are very similar, you may end up hitting some inconsistencies. We're likely going to experiment with other JavaScript engines in the future, so it's best to avoid relying on specifics of any runtime.

JavaScript Syntax Transformers

Syntax transformers make writing code more enjoyable by allowing you to use new JavaScript syntax without having to wait for support on all interpreters.

React Native ships with the [Babel JavaScript compiler](#). Check [Babel documentation](#) on its supported transformations for more details.

A full list of React Native's enabled transformations can be found in [metro-react-native-babel-preset](#).

React Native Project development environment

1. Expo Go:- The easiest way to get started is with Expo Go. Expo is a set of tools and services built around React Native and, while it has many [features](#), the most relevant feature for us right now is that it can get you writing a React Native app within minutes. You will only need a recent version of Node.js and a phone or emulator. If you'd like to try out React Native directly in your web browser before installing any tools, you can try out [Snack](#).

2.Native CLI :- You may want to use React Native CLI. It requires Xcode or Android Studio to get started. If you already have one of these tools installed, you should be able to get up and running within a few minutes. If they are not installed, you should expect to spend about an hour installing and configuring them.

Installing dependencies

You will need Node, the React Native command line interface, a JDK, and Android Studio. While you can use any editor of your choice to develop your app, you will need to install Android Studio in order to set up the necessary tooling to build your React Native app for Android.

Node, JDK

We recommend installing Node via [Chocolatey](#), a popular package manager for Windows. It is recommended to use an LTS version of Node. If you want to be able to switch between different versions, you might want to install Node via [nvm-windows](#), a Node version manager for Windows. React Native also requires [Java SE Development Kit \(JDK\)](#), which can be installed using Chocolatey as well. Open an Administrator Command Prompt (right click Command Prompt and select "Run as Administrator"), then run the following command:

```
choco install -y nodejs-lts openjdk11
```

Android development environment

Setting up your development environment can be somewhat tedious if you're new to Android development. If you're already familiar with Android development, there are a few things you may need to configure. In either case, please make sure to carefully follow the next few steps.

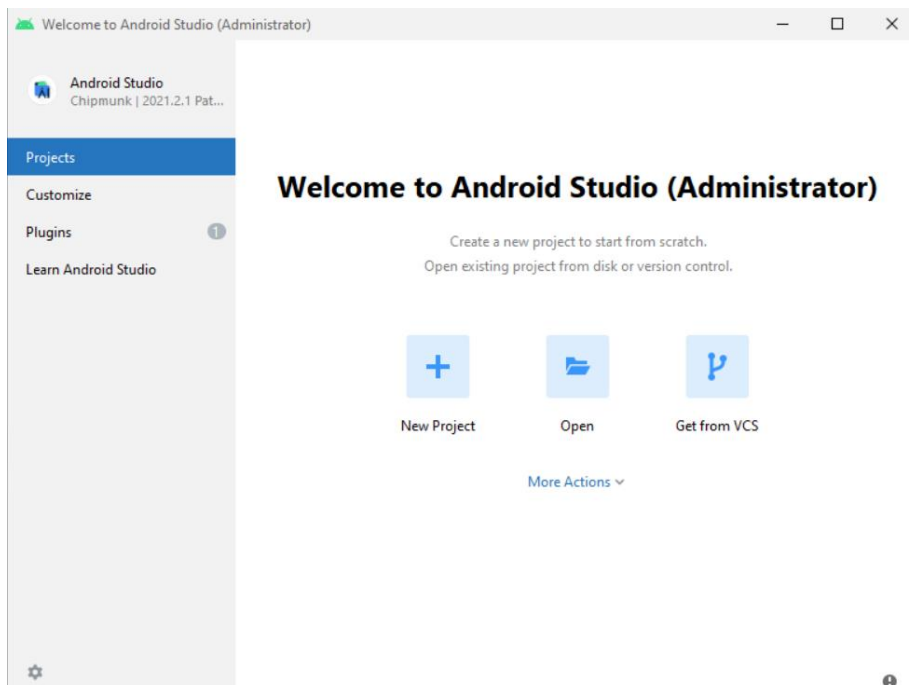
1. Install Android Studio

[Download and install Android Studio](#). While on Android Studio installation wizard, make sure the boxes next to all of the following items are checked:

- Android SDK
- Android SDK Platform
- Android Virtual Device
- If you are not already using Hyper-V: Performance (Intel ® HAXM) ([See here for AMD or Hyper-V](#))

2. Install the Android SDK

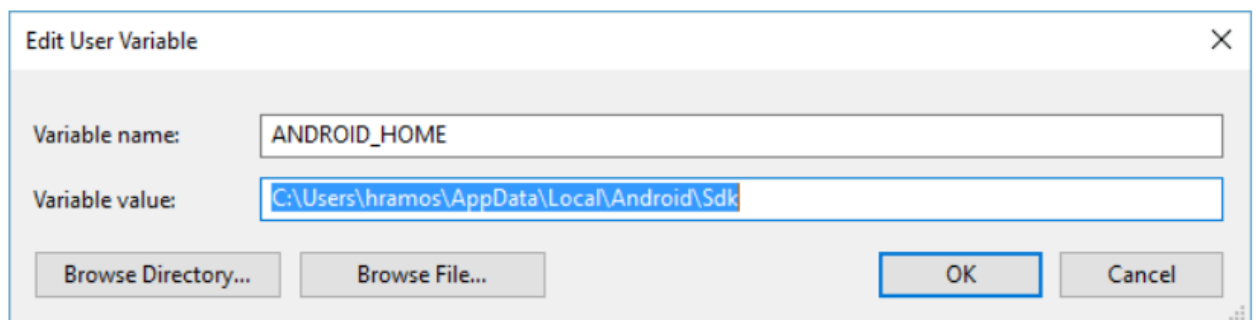
Android Studio installs the latest Android SDK by default. Building a React Native app with native code, however, requires the Android 12 (S) SDK in particular. Additional Android SDKs can be installed through the SDK Manager in Android Studio. To do that, open Android Studio, click on "More Actions" button and select "SDK Manager".



3. Configure the ANDROID_HOME environment variable

The React Native tools require some environment variables to be set up in order to build apps with native code.

1. Open the **Windows Control Panel**.
2. Click on **User Accounts**, then click **User Accounts** again
3. Click on **Change my environment variables**
4. Click on **New...** to create a new `ANDROID_HOME` user variable that points to the path to your Android SDK:



4. Add platform-tools to Path

1. Open the **Windows Control Panel**.
2. Click on **User Accounts**, then click **User Accounts** again
3. Click on **Change my environment variables**
4. Select the **Path** variable.
5. Click **Edit**.
6. Click **New** and add the path to platform-tools to the list.

React Native Command Line Interface

React Native has a built-in command line interface. Rather than install and manage a specific version of the CLI globally, we recommend you access the current version at runtime using `npx`, which ships with Node.js. With `npx react-native <command>`, the current stable version of the CLI will be downloaded and executed at the time the command is run.

Creating a new application

React Native has a built-in command line interface, which you can use to generate a new project. You can access it without installing anything globally using `npx`, which ships with Node.js. Let's create a new React Native project called "AwesomeProject":

```
npx react-native init AwesomeProject
```

This is not necessary if you are integrating React Native into an existing application, if you "ejected" from Expo, or if you're adding Android support to an existing React Native project (see [Integration with Existing Apps](#)). You can also use a third-party CLI to init your React Native app, such as [Ignite CLI](#).

[Optional] Using a specific version or template

If you want to start a new project with a specific React Native version, you can use the `--version` argument:

```
npx react-native init AwesomeProject --version X.XX.X
```

Running your React Native application

Step 1: Start Metro

First, you will need to start Metro, the JavaScript bundler that ships with React Native. Metro "takes in an entry file and various options, and returns a single JavaScript file that includes all your code and its dependencies."—[Metro Docs](#)

To start Metro, run `npx react-native start` inside your React Native project folder:

```
npx react-native start
```

Step 2: Start your application

Let Metro Bundler run in its own terminal. Open a new terminal inside your React Native project folder. Run the following:

```
npx react-native run-android
```


Local Data Base

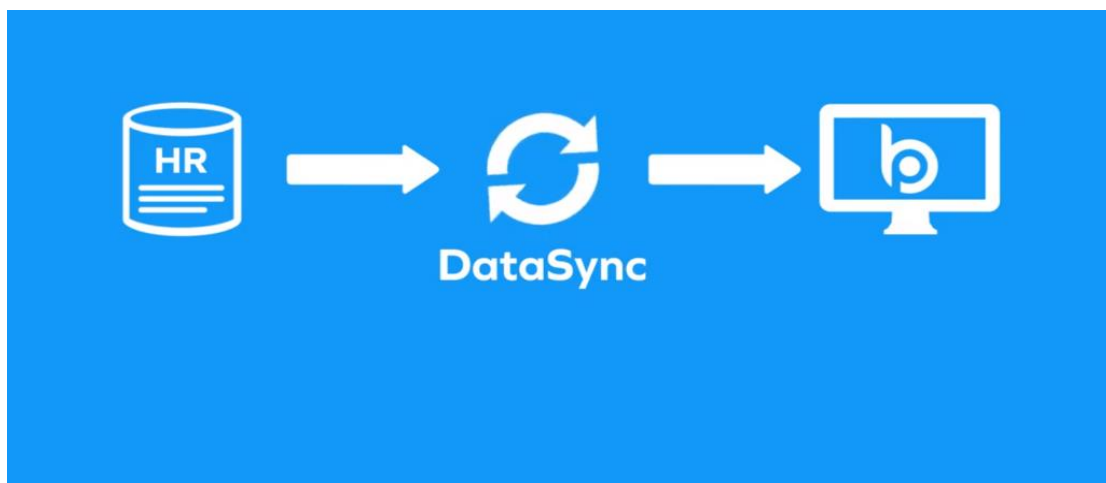
Overview

Every `mongod` instance has its own `local` database, which stores data used in the replication process, and other instance-specific data. The `local` database is invisible to replication: collections in the `local` database are not replicated.

Collections on all `mongod` Instances

`local.startup_log`

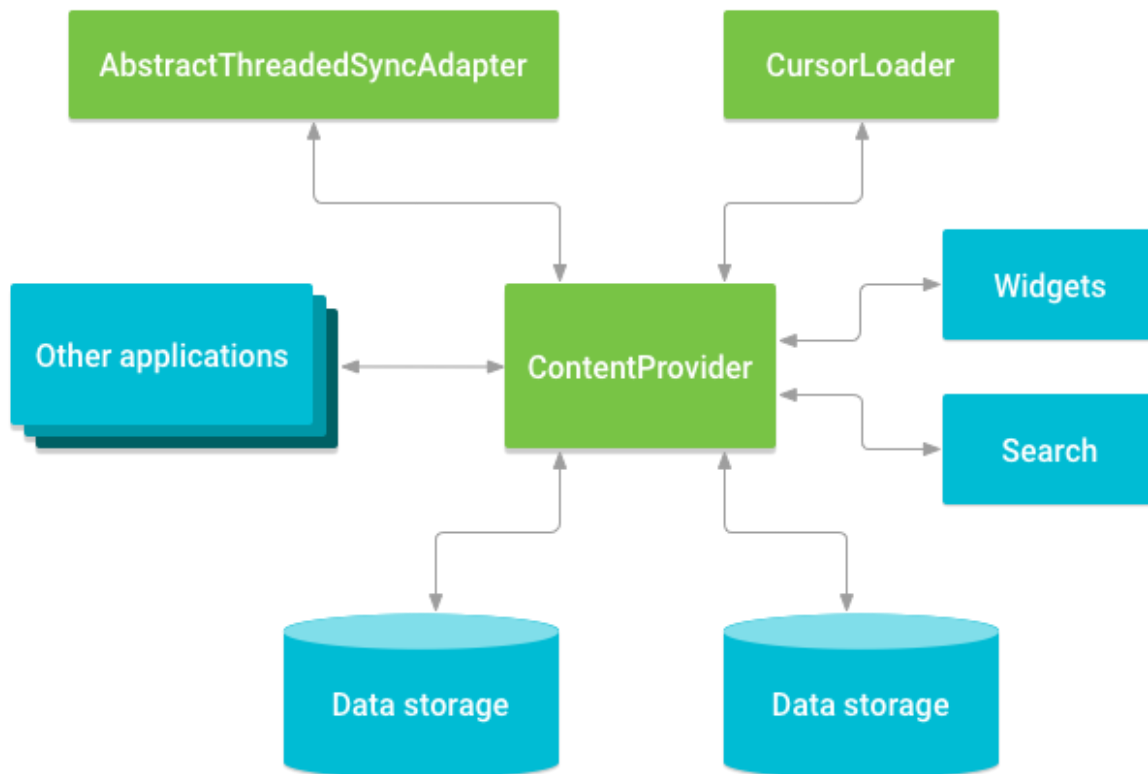
On startup, each `mongod` instance inserts a document into `startup_log` with diagnostic information about the `mongod` instance itself and host information. `startup_log` is a capped collection. This information is primarily useful for diagnostic purposes.



System in Context:

The student app provides the users to punctual time period to guide the students in a proper way. This app helps the student to maintain their time in studies and have fun while learning. And focus on mock test for each unit.

Context Diagram:



System-wide Requirements(Received):

Actors:

The system interact with many users. Each user has its own functions to access with the system. The functionalities of users are dependent on each other.

Events:

Student is multiuser system which provides user to study with each other for day to day operations.

The most critical events are:

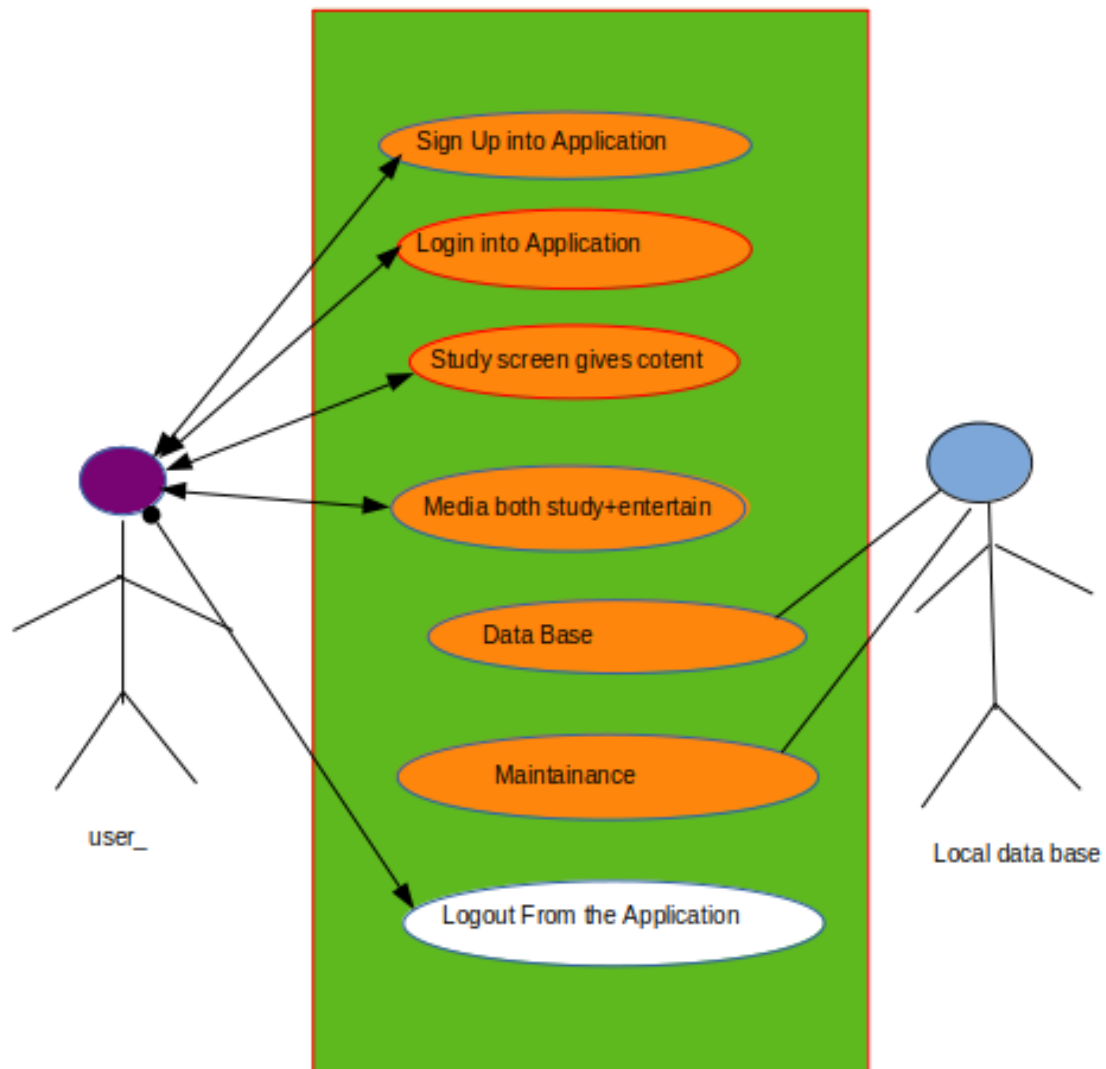
- 1.Gets register first using the college ID, Email address, password.
- 2.Users login using the email and password can select another user.
- 3.After login/signup to the app, then student can access all the features.
- 4.Student can take exam/unit test after completion of a unit.
5. User can log out from the app

The below table provides a set of user visible events that define the functionalities that are in Student App.

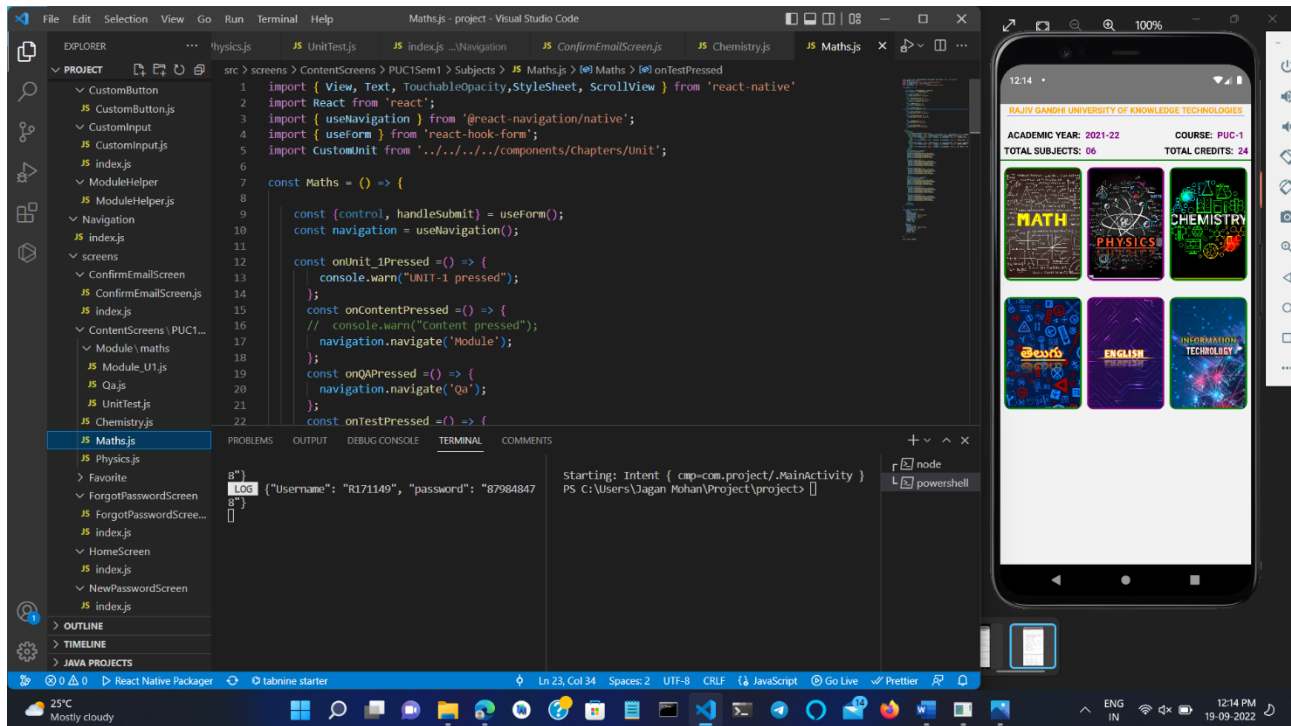
	Actor	Action	Object	Frequency	Arrival Pattern	Response
1.	user	onclick	Splash screen	1/day	Episodic	Show the app icon for a while.
2.	user	Onclick	Login	1/day	Episodic	It redirects to the home screen

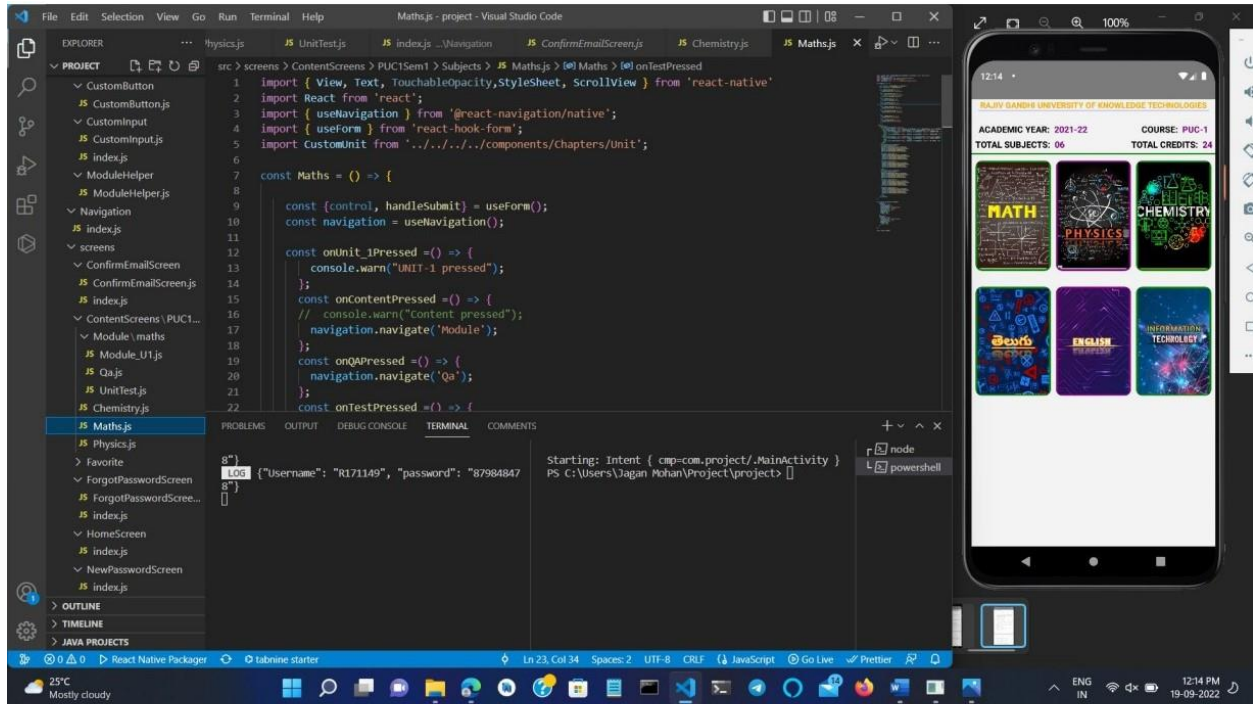
3.	user	Onclick	Register	1/day	Episodic	It redirects to the sign up screen
4.	user	onclick	Forgot password	1/day	Episodic	Send to change password screen
5.	user	onclick	subject	1/day	Episodic	It shows units of the unit
6.	user	onclick	content	1/day	Episodic	It displays content of the unit
7.	user	onclick	Q&A	1/day	Episodic	It displays the question and answers
8.	user	Onclick	Unit Test	1/day	Episodic	It displays questions to write test

Use case Diagram:



Output(Image References)





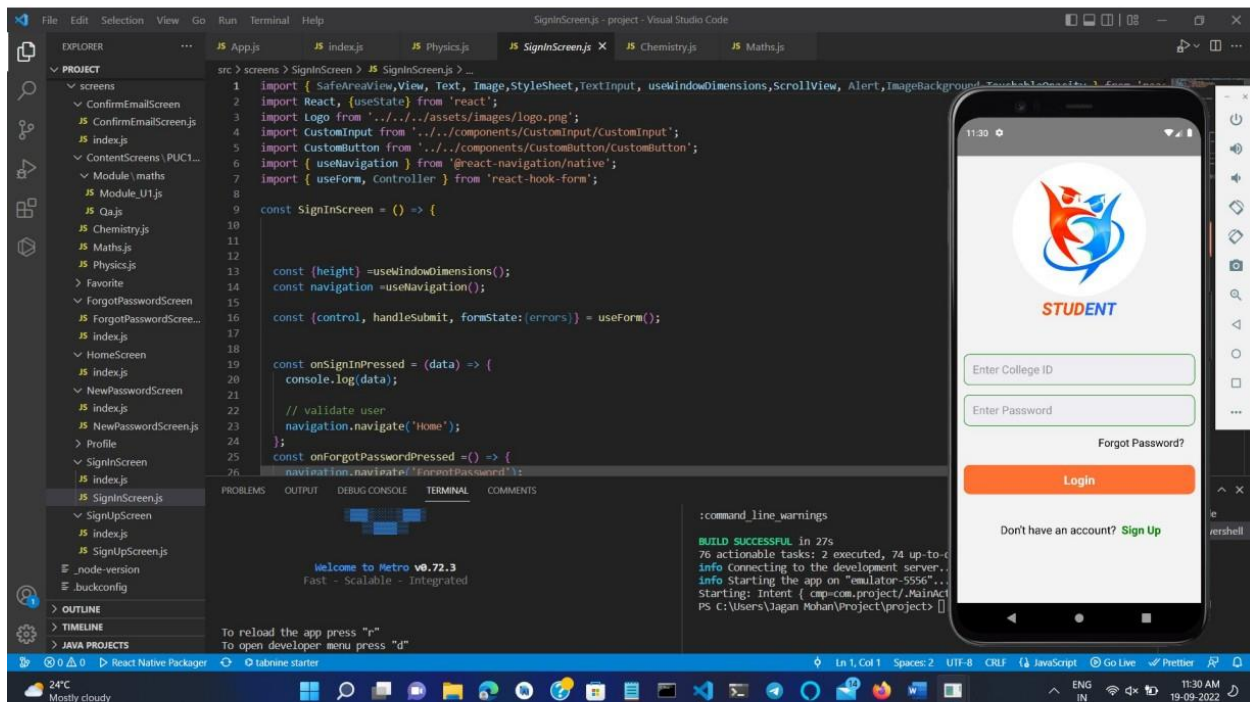
FLOW OF ARHENTICATION [LOCAL DATA BASE]

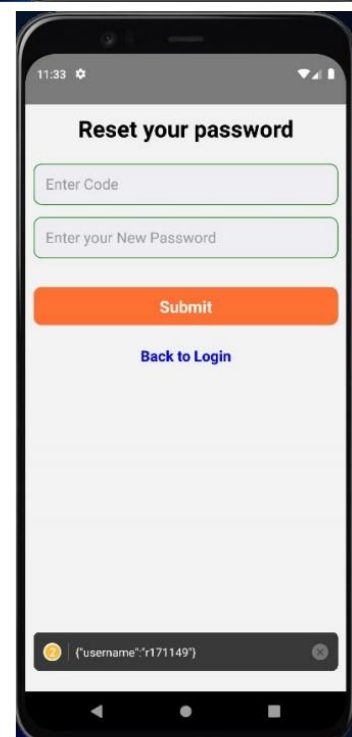
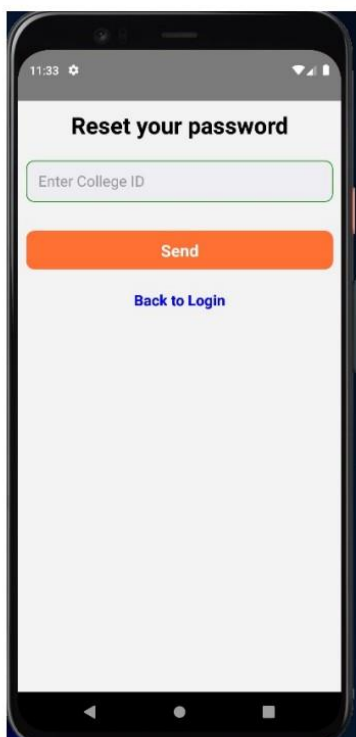
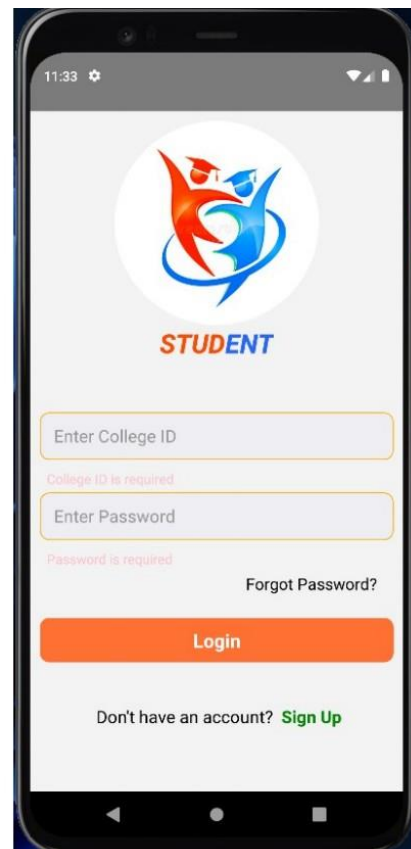
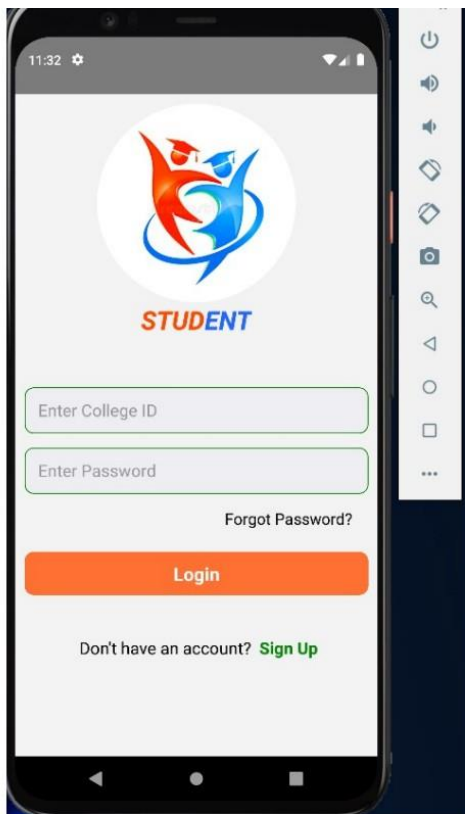
```
npm install react-native@latest
Jagan Mohan@Jagan MINGW64 ~
$ cd Project
Jagan Mohan@Jagan MINGW64 ~/Project
$ npx react-native init StudEnt

#####
###           ###
###   ###   ###
###   ###   ###
###   ###   ###
###   ###   ###
#####
###           ###
###   ###   ###
###   ###   ###
###   ###   ###
###   ###   ###
#####
###           ###
###   ###   ###
###   ###   ###
###   ###   ###
###   ###   ###
#####
###           ###
###   ###   ###
###   ###   ###
###   ###   ###
###   ###   ###
#####
###           ###
###   ###   ###
###   ###   ###
###   ###   ###
###   ###   ###
#####
#####

Welcome to React Native!
Learn once, write anywhere

/ Downloading template
```





11:35

Create Account

R171149

r171149@gruktrkv.ac.in

.....

12345

Register

By registering, you confirm that you accept our [Terms of Use](#) and [Privacy Policy](#)

> Suggest contacts? Tap for info.

q w e r t y u i o p
a s d f g h j k l
z x c v b n m

7123 , ☺ . ✓

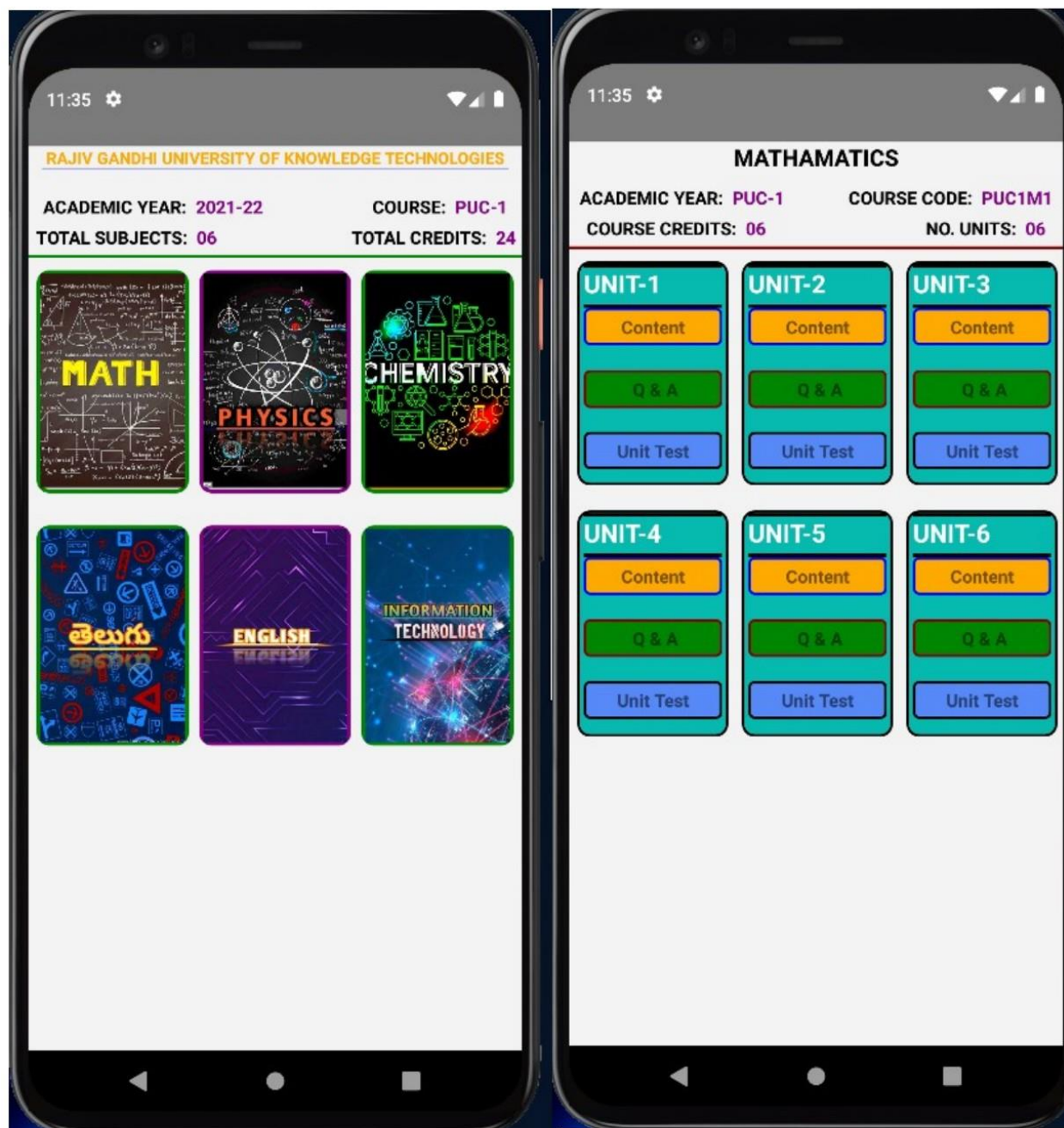
11:35

Confirm your Email

Enter your confirmation code

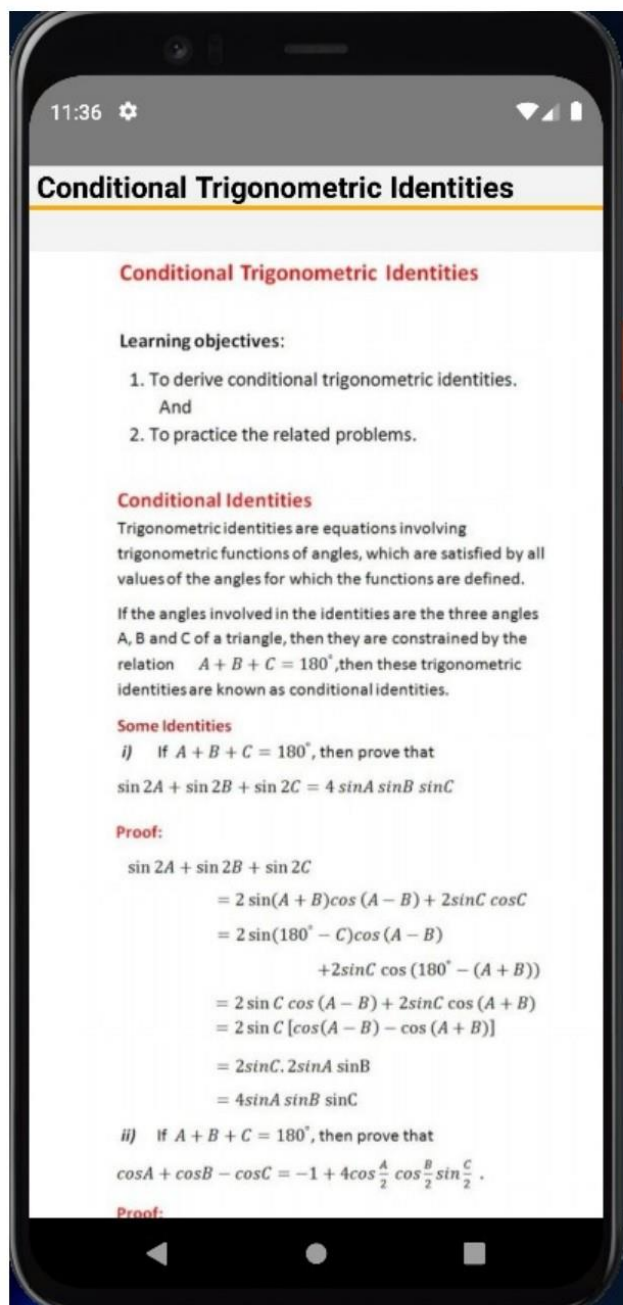
Confirm

[Back to SignUp](#) [Resend Code](#)

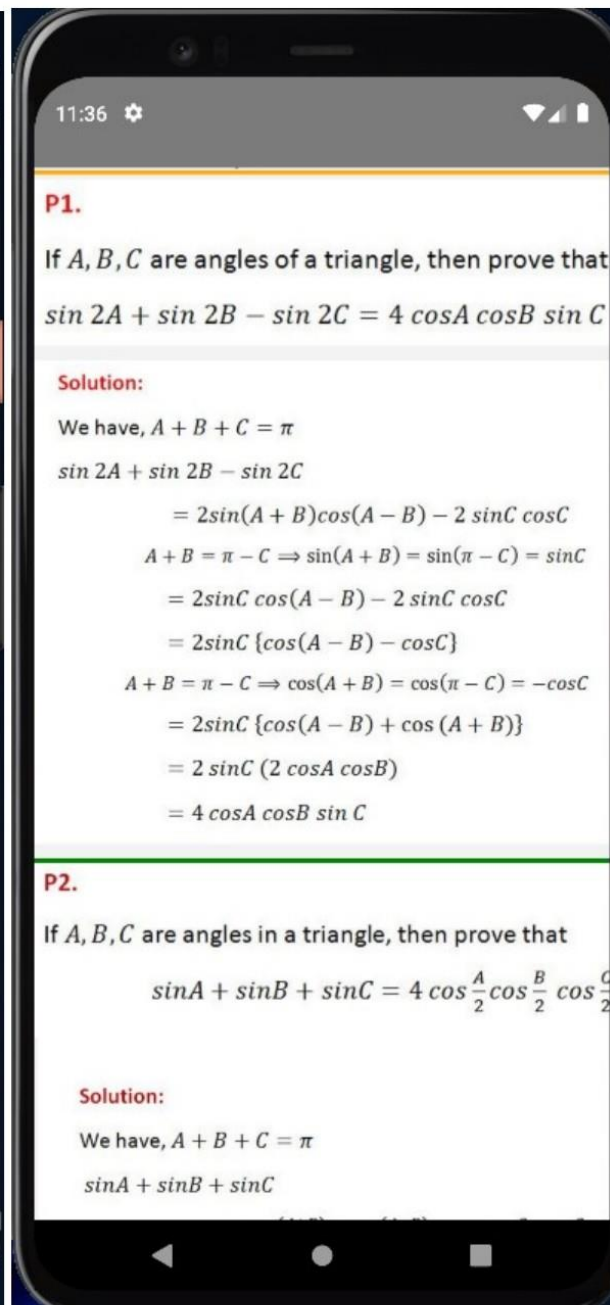


Academic Details

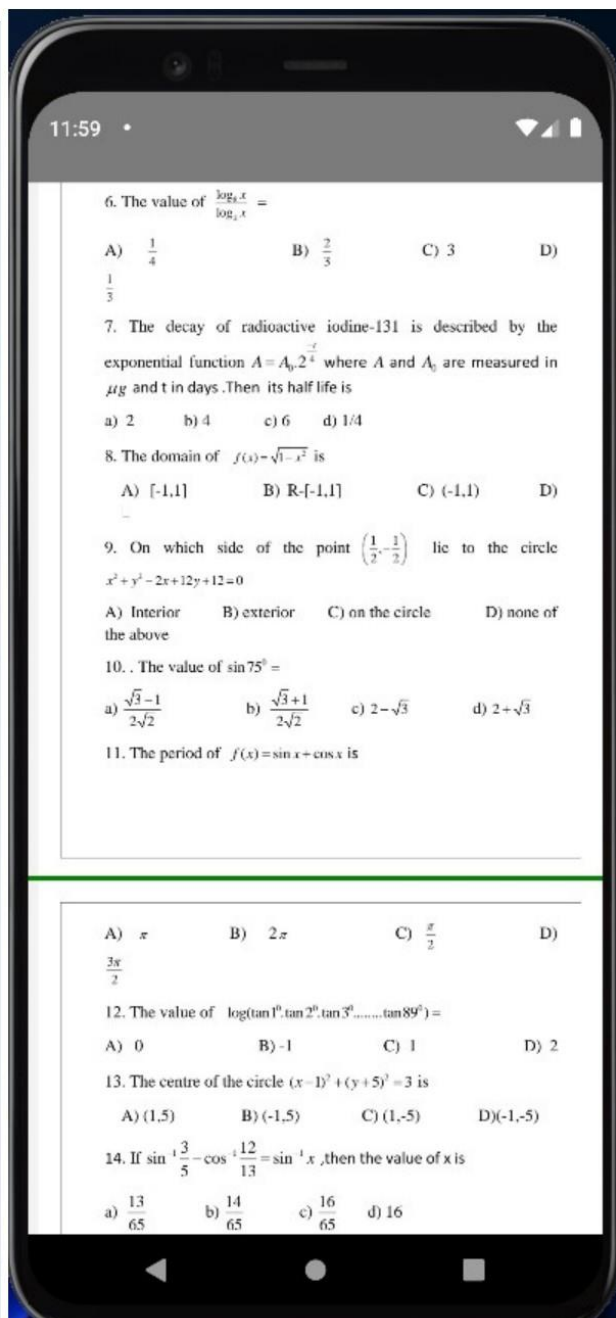
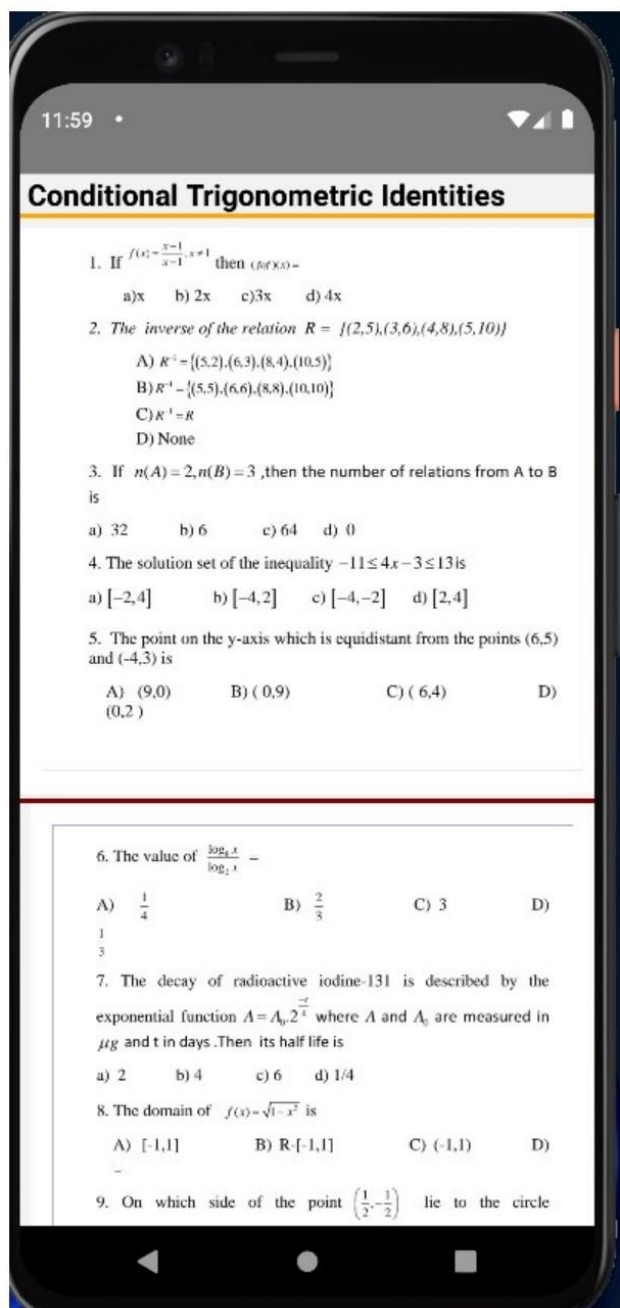
Subject Details



College Content



Questions & Answers



Unit Test (self Test)

CONCLUSION

This Student Application provides user study material with Rich User Interface and UX.

Student App Covers the simplified content various Categories and displays the document.

The Application simplifies the student burden with task by task understanding and units test to know them self.

FUTURE ENHANCEMENT

User can communicate all over the students. User will be provided with certain Username and Password for the User Authentication. And they edit their profile them self.

Students can question and write answers over there by the student rating suggestions will be done. Social media with new implementation with rich UI and best Logic implementation.

Notifications about the News will received by the User from the Next Version.

REFERENCES

<https://reactnative.dev/docs>

<https://localdatabase.google.com/docs/auth>

<https://git/reactnative/navigation-screens>