

B.V.V. Sangha's
BASAVESHWARA ENGINEERING COLLEGE (AUTONOMOUS)
BAGALKOT - 587103

: Project Title :

KAKSHA [Android Based Application]

Self-Learning Application

As the number of smartphone owners grow, so does the popularity of mobile apps. We live in a fast-paced world where speed is key in every single thing we do. Whether good or bad, this makes us crave for any technology which can help us speed up on all levels, personal and professional. So this is the point of origin for our android based application "KAKSHA", which helps students and teachers by providing virtual class-room for carrying out basic activities in such pandemic.

BY : Kartik A Mashal
USN : 2BA18CS025

KAKSHA [Android Based Application]



B.V.V. Sangha's
BASAVESHWARA ENGINEERING COLLEGE (AUTONOMOUS)
BAGALKOT - 587103

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that **Kartik A Mashal** has satisfactorily completed the mini project on "(KAKSHA) Android based Application" prescribed by the university for 6th semester B.E class of this college during the year 2020-21.

Place : BAGALKOT

USN : 2BA18CS025

Date :

Supervisor

Prof. S. S. Yendigiri

Head of the department

Dr. V. B. Pagi

ACKNOWLEDGEMENT

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my project to complete the project successfully.

I would like to express my deep and sincere gratitude to my project supervisor, Mrs. Savitri Yendigeri and Mrs. Savita Hanji of Department of Computer Science and Engineering, Basaveshwar Engineering College, Bagalkot, for allowing me to do project and providing invaluable guidance throughout this project. Their dynamism, vision, sincerity and motivation have deeply inspired me. They taught me the methodology to carry out the project and to present the project works as clearly as possible. It was a great privilege and honor to work and study under their guidance. I would also like to thank them for their friendship, empathy, and great sense of humor.

I am extremely grateful to my parents for their love, prayers, caring and sacrifices for educating and preparing me for my future.

I express my special thanks Mr. V. B. Pagi, Professor and Head of Department of Computer Science and Engineering, Basaveshwar Engineering College, Bagalkot, for his genuine support throughout this project work.

I would like to say thanks to my friends, Mr. K. Adish, Mr. K. Anand, Mr. P. Rakshit and Mr. P. Rahul for their constant encouragement, help and support.

Finally, my thanks go to all the people who have supported me to complete the project directly or indirectly.

M. Kartik

ABSTRACT

Android, operating system for cellular telephones and tablet computers. Android began in 2003 as a project of the American technology company Android Inc., to develop an operating system for digital cameras. In 2004 the project changed to become an operating system for smartphones. Android Inc., was bought by the American search engine company Google Inc., in 2005. At Google, the Android team decided to base their project on Linux, an open source operating system for personal computers. The Android mobile platform has developed from its first phone in October 2008 to being the most popular smart phone operating system in the world by 2012. The explosive growth of the platform has been a significant win for consumers with respect to competition and features. The market has been booming in the past few years that, there are now over 3.14 million applications on the Android market.

Due to wide use of android applications most of the activities such as Learning, working, trading etc are done through the mobile applications.

As the number of smartphone owners grow, so does the popularity of mobile apps. We live in a fast-paced world where speed is key in every single thing we do. Whether good or bad, this makes us crave for any technology which can help us speed up on all levels, personal and professional.

So this is the point of origin for our android based application “KAKSHA”, which helps students and teachers by providing virtual class-room for caring out basic activities in such pandemic.

TABLE OF CONTENTS

| CHAPTER NO. | CHAPTER HEDDINGS | PAGE NO. |
|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|
| 1 | INTRODUCTION 1.1 PURPOSE 1.2 SCOPE 1.3 DEFINATION | 4-5 |
| 2 | LITERATURE SURVEY 2.1 PROBLEM DEFINATION 2.2 SYSTEM OBJECTIVES 2.3 SOFTWARE REQUIREMENTS 2.3.1 FUNCTIONAL SOFTWARE REQUIREMENTS 2.3.2 NON-FUNCTIONAL SOFTWARE REQUIREMENTS 2.4 SYSTEM REQUIREMENTS 2.4.1 SYSTEM SOFTWARE REQUIREMENTS 2.4.2 SYSTEM HARDWARE REQUIREMENTS 2.5 EXISTING SYSTEM 2.6 PROPOSED SYSTEM | 6-12 |
| 3 | IMPLEMENTATION 3.1 SOFTWARE TOOLS USED 3.2 DATA FLOW DIAGRAMS 3.2.1 FLOW DIAGRAMS SYMBOLS 3.2.2 SYSTEM ARCHITECTURE OVERVIEW 3.2.3 ADMIN MOULE OVERVIEW 3.2.4 STAFF MODULE OVERVIEW 3.2.5 STUDENT MODULE OVERVIEW 3.3 SYSTEM IMPLEMENTATIONS(MODULES) | 13-19 |
| 4 | RESULTS | 20-40 |
| 5 | CONCLUSION & FUTURE SCOPE | 41 |
| 6 | REFERENCES | 42 |

1. INTRODUCTION

This document provides a detailed description of the Self-study app (An android-based application). Which is used by the faculty and student to provide and access the study materials in the form of quiz and notes

1.1 Purpose

The purpose of the software requirement specification is to specify the functionality and performance (non-functionality) requirements of the software project. All of the required features of the software project expressed by the client and agreed upon by the development team shall be expressed in this document. Additional requirements of the client shall first be agreed upon by the development team. The formal release versions of the project shall meet every requirement listed in this document.

1.2 Scope

The self-learning app is an android based application for Basaveshwar engineering college. The application offers various operations like creating classrooms. Creating quizzes in classroom and providing the notes to the student. These services are conveniently grouped and developed specifically for the use on the android device. All operations are performed when an internet connection is available. Admin of the application can register prospective Students, faculty or staff to fully experience all services provided.

1.3 Definition, Acronyms, and abbreviations.

Android: Developed by google, a popular system for smartphones

CPU: Central processing unit. It is a hardware device in a computer that is responsible for most of the mathematical calculations required by the computer. All programs execution is performed by the CPU. The CPU includes all derived components including the ALU.

GUI: Graphical user interface. An interface that receives and reacts to the user inputs with a graphical display.



Disk Storage: May be any medium mapped to the root file system. Where the data is actually stored.

Data base: A cloud-based storage service where the shared data is stored.

XML: Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The World Wide Web Consortium's XML 1.0 Specification of 1998 and several other related specifications all of them free open standards define XML

KOTLIN: is a cross-platform, statically typed, general-purpose programming language with type inference. Kotlin is designed to interoperate fully with Java, and the JVM version of Kotlin's standard library depends on the Java Class Library, but type inference allows its syntax to be more concise. Kotlin mainly targets the JVM, but also compiles to JavaScript (e.g., for frontend web applications using React) or native code (via LLVM); e.g., for native iOS apps sharing business logic with Android apps. Language development costs are borne by Jet Brains, while the Kotlin Foundation protects the Kotlin trademark.

OS: Operating system

RAM: Volatile memory used by the CPU for sharing data too large to fit in a register along with execution of the running programs.

SRS: System requirements specification.

2. LITERATURE SURVEY

We will discuss and analyse about the developing process of Self Learning application including software requirement specification (SRS) and comparison between existing and proposed system. The functional and non-functional requirements are included in SRS part to provide complete description and overview of system requirement before the developing process is carried out. Besides that, existing vs proposed provides a view of how the proposed system will be more efficient than the existing one.

GENERAL DESCRIPTION: Self Learning application is an Android app which helps user (Staff & student) to manage the classroom daily activity in electronic format. It reduces the risk of paper work such as file lost, file damaged and time consuming. It can help user to reduce the Paper-waste that could have been generated.

2.1 PROBLEM DEFINATION

Being in 21st century, where automation plays a major role and even small things matter. Developing an android based application for class-room activities where classes are loaded into students account by auto-filtration feature. Since in other application we have entre the class code to join the class-room.

2.2 SYSTEM OBEJECTIVES

- To provide simple and user-friendly interface.
- To provide access to course related material any time.
- To provide self-assessment for students using quizzes.
- To provide unlimited creation of classes.
- To provide optimized code.
- To Design college specific.

2.3 SOFTWARE REQUIREMENTS

An android app has to be developed. Which will be helpful for staff and students which acts as a learning platform for students where the staff and student must be benefitted with the ease of use of the software and the platform it provides for studying.

2.3.1 FUNCTIONAL SOFTWARE REQUIREMENTS

- The software must consist of 3 modules
 - ADMIN
 - STAFF
 - STUDENT
- There must be intro page which allows the user to select their module

1) ADMIN :

Authentication: -The admin has to be authorized with their email and password.

TASKS :

- The admin can add another admin who have same authority of the pervious admin
- The admin can add a staff
- The admin can add a student
- The admin can delete the staff
- The admin can delete the student
- The admin also must authority to login as a staff as admin can also be a staff
- We must maintain a database where we store all the details of admin staff and student and other objects.
- The admin can be able to change his password if required
- The admin can be able to logout when required
- As of adding deleting staff and student admin will provide the proper information with are related to that object
- NOTE: - we store the data of admin staff student and others as json files or documents we refer them as objects in this context
 - OBJECT OF ADMIN
 - The admin object consists of his
 - email id
 - password
 - Admin id
 - Name

2) STAFF :

Authentication: -The staff have to be authorized with their email and password.

TASKS :

- The staff should be able to create classrooms (Creating classroom objects).
- The classrooms created by staff must be visible to the staff home page
- In each classroom the staff must be able to create quizzes, notes, announcements
- The Quizzes may contain any number of questions with 4 option per question with mention of correct answer.
- The Notes can be any image or Pdf documents with can be able to view in mobile using off the shelf available software's
- The announcements can be any text
- The staff must be able to view any previously created quizzes, notes and announcements.
- The staff can change his email and password if required
- The staff must be able to logout
- As of creating classroom, quizzes, notes, announcements the staff will provide the proper information with are related to that object
- NOTE: - we store the data of admin staff student and others as json files or documents we refer them as objects in this context
 - OBJECT OF STAFF
 - The staff object consists of
 - Staff Id
 - Staff Name
 - Staff Branch
 - Staff Email
 - Staff Password
 - OBJECT OF CLASSROOM
 - The classroom object consists of
 - Class Id
 - Class Name
 - Staff Id
 - Class Sem
 - Class Div
 - Class Subject Code
 - Class Branch
 - Class Students (Array of students usn)
 - Class PDF (Array of notes)



- Class Notices (Array of text)
- Class Quiz (array of quiz objects)
- OBJECT OF QUIZ
 - The QUIZ OBJECT consists of
 - Q Id
 - Q Name
 - Arr Questions: ArrayList<String>
 - Arr Options: ArrayList<String>
 - Arr Correct Option:ArrayList<String>

3) STUDENT :

Authentication: -The student has to be authorized with their email and password.

TASKS :

- The student can be able to view the classroom based on his branch and semester and division
- In each classroom the student must be able to view the announcements, notes, quizzes created by the Staff.
- The student must be able to take the quizzes and at the end he must be able to view his score
- The student must be able to change his email id and password
- The student must be able to update his semester
- The student must be able to logout
- NOTE: - we store the data of admin staff student and others as json files or documents we refer them as objects in this context
 - OBJECT OF STUDENT
 - The student object consists of
 - Stud Id
 - Stud Name
 - Stud Branch
 - Stud Sem
 - Stud USN
 - Stud Div
 - Stud Email
 - Stud Password

2.3.2 NON-FUNCTIONAL SOFTWARE REQUIREMENTS

1. **Performance and scalability.** The app should be light (consume less storage) and should perform well in all the possible device with good speed
2. **Portability and compatibility.** The app is designed for android based operating system. With the hardware constraints discussed in system requirements in detail.
3. **Reliability, availability, maintainability.** The app should be less prone to bugs and errors or failure. The app should be available when the user requests to use (proper internet access should be available).and easy to maintain (i.e., easy to add future upgrades and changes need by the end user)
4. **Security.** The system should be secure to use. It should allow illegal access to data. Proper authentication should be done before allowing to access the database
5. **Usability.** The user interface should be designed such that it is easy for the all the end user to use without any difficulties

2.4 SYSTEM REQUIREMENTS

To be used efficiently, all computer software needs certain hardware components or other software resources to be present on a computer. These prerequisites are known as (computer) **system requirements** and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements.

2.4.1 SYSTEM SOFTWARE REQUIREMENTS

I. ANDROID VERSION :

- Minimum version required = Android 5.0 (i.e., is Lollipop)
- Target version built = Android 11.0 (i.e., is Android R)

II. API LEVEL :

- Minimum API level supported = 21
- Target API level built = 30

2.4.2 SYSTEM HARDWARE REQUIREMENTS

I. PROCESSOR :

- Required quad core processor.
- Snap dragon series: Minimum processor = Snapdragon S4 Prime with 1.5 GHz.
- Exynos series: Minimum processor = Exynos 7 Quad with 1.4 GHz.
- Kirin Series: Minimum processor = K3V2 with 1.4 GHz.
- And all standard processors that can run Android 5.0 and above.

II. RAM :

- Minimum required = 512 MB.
- Recommended = 1.5 GB and above.

III. STORAGE REQUIREMENTS :

- Minimum required = 100 MB.
- Recommended = 500 MB and above.

2.5 EXISTING SYSTEM

- Recent days the classroom activities are gone from offline mode to online mode concerning the pandemic of Covid-19. The staff and students faced a lot of difficulties in conducting the classroom activities online. Due to less familiar software, the complex structure and non-user friendliness thus we come with the idea of building an app with user friendly and reduced complicity for both staff and students.
- In the offline time there was also a problem that the teachers find it difficult to prepare notes every year due to loss of old files either Electronic or normal hand written documents, thus every year they need to prepare it again.
- The students also use to face a problem i.e., they usually find it difficult to study from textbooks due to the restriction of only 2 books from the library and unavailability of old notes and quizzes that have been performed in the previous semester.
- Another concern is that the students usually neglect the importance of the notes that are provided to them in their previous semester and lose it or find it difficult when they are need in next semester or for the preparation for competitive exams.

- Maintenance of the Classroom notes and storage of the notes and quizzes is very complex task, and staff and students usually face difficulties more often.
- The less familiar software and their non-user friendliness also made it even difficult to manage classroom activities. Such as creating a classroom providing notes creating quizzes.
- Some concern about the student manually needs to get the classroom code or link in order to join the classroom, which is usually been shared through some messaging app and some students usually miss to join those classrooms and later find difficulties.
- There is storage constraint in the existing classroom apps. (Such as google classroom where only 15 GB of storage is provided which is also been share with Gmail, drive, photos etc.).

2.7 PROPOSED SYSTEM

To solve the inconveniences as mentioned in the existing system, a new android based **Self-Learning app** is proposed. The proposed system contains the following features:

- The teachers can create quizzes, provide notes and also make some announcements
- Individually each member will have his account through which he can access the information he needs.
- The students can access the notes, take quizzes and get the announcements made by the particular teacher
- The teacher is no longer need to create new notes every semester as their previous notes is saved and those classrooms can be accessed by the students directly filtering through the Department and their semester
- Maintenance will be easy as only required features are included make the app less complex and User-friendly
- No need of any classroom code or link to join the classroom as student can access all the classroom related to his Department and Semester directly and can update his semester whenever required either to access old semester classrooms or join next semester
- In order to manage the staff and students (i.e. the creation of account and deletion of account) we have added a new user that is admin who has to access to create staff and student accounts.
- We have tried to sub-divide the tasks of classroom activities into staff, student and admin modules making it less complex and more user friendly to operate
- There is no storage constraint in our app .so teaches are no longer have to be concerned with their cloud storage getting full.

3. IMPLEMENTATION

Implementation is the process of:

1. Defining how the information system should be built (i.e., physical system design),
2. Ensuring that the information system is operational and used,
3. Ensuring that the information system meets quality standard (i.e., quality assurance).

3.1) SOFTWARE TOOLS USED

3.1.1) XML :



Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The design goals of XML emphasize simplicity, generality, and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary data structures such as those used in web services.

A programming language consists of grammar rules and its own vocabulary which is used to create computer programs. These programs instruct the computer to perform specific tasks. XML does not qualify to be a programming language as it does not perform any computation or algorithms. It is usually stored in a simple text file and is processed by special software that is capable of interpreting XML.

3.1.2) Kotlin:



Kotlin is an open-source, statically-typed programming language that supports both object-oriented and functional programming. Kotlin provides similar syntax and concepts from other languages, including C#, Java, and Scala, among many others. Kotlin does not aim to be unique—instead, it draws inspiration from decades of language development.

In July 2011, JetBrains unveiled Project Kotlin, a new language for the JVM, which had been under development for a year. JetBrains lead Dmitry Jemerov said that most languages did not have the features they were looking for, with the exception of Scala. However, he cited the slow compilation time of Scala as a deficiency. One of the stated goals of Kotlin is to compile as quickly as Java. In February 2012, JetBrains open sourced the project under the Apache 2 license.

3.2.3) Fire-Base :



Firebase is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development.

Firebase evolved from Envolv, a prior startup founded by James Tamplin and Andrew Lee in 2011. Envolv provided developers an API that enables the integration of online chat functionality into their websites. After releasing the chat service, Tamplin and Lee found that it was being used to pass application data that were not chat messages. Developers were using Envolv to sync application data such as game state in real time across their users. Tamplin and Lee decided to separate the chat system and the real-time architecture that powered it. They founded Firebase as a separate company in September 2011 and it launched to the public in April 2012.

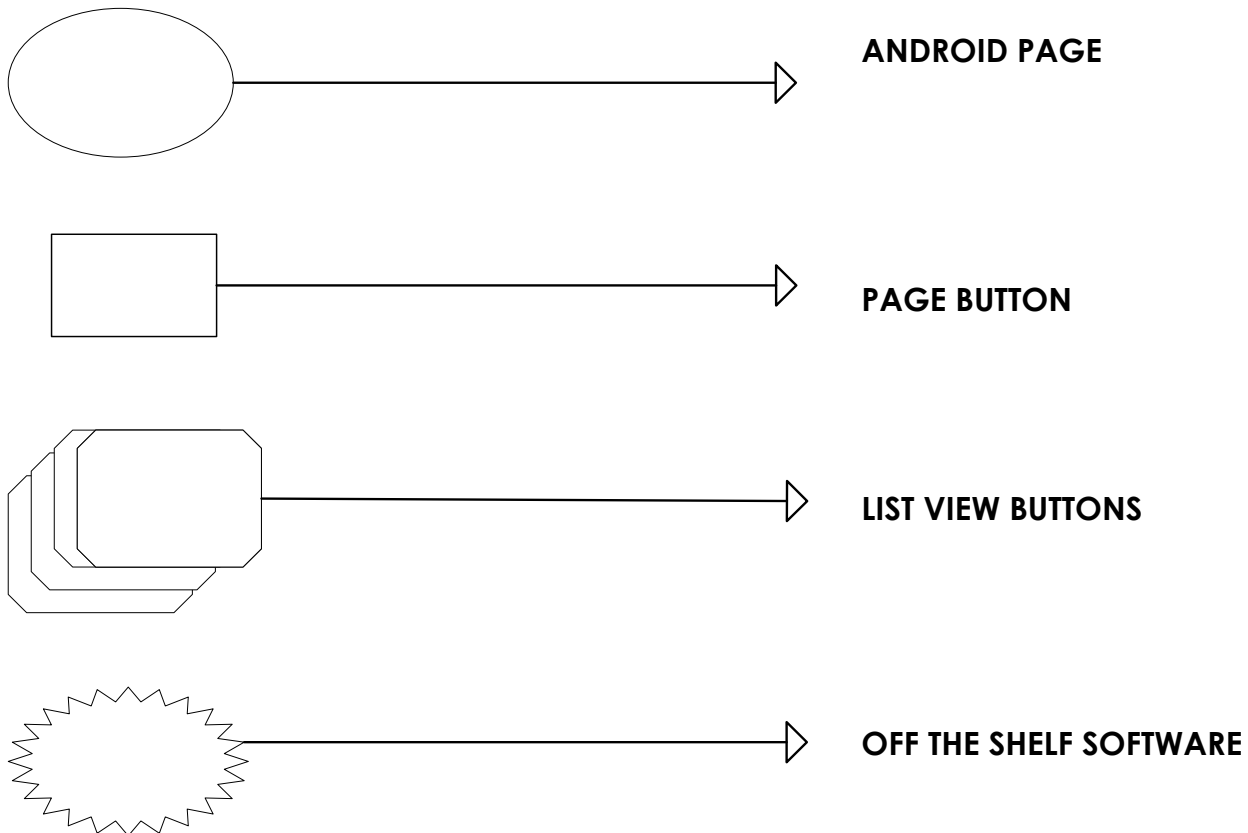
Firebase's first product was the Firebase Realtime Database, an API that synchronizes application data across iOS, Android, and Web devices, and stores it on Firebase's cloud. The product assists software developers in building real-time, collaborative applications.

3.2 FLOW DIAGRAMS

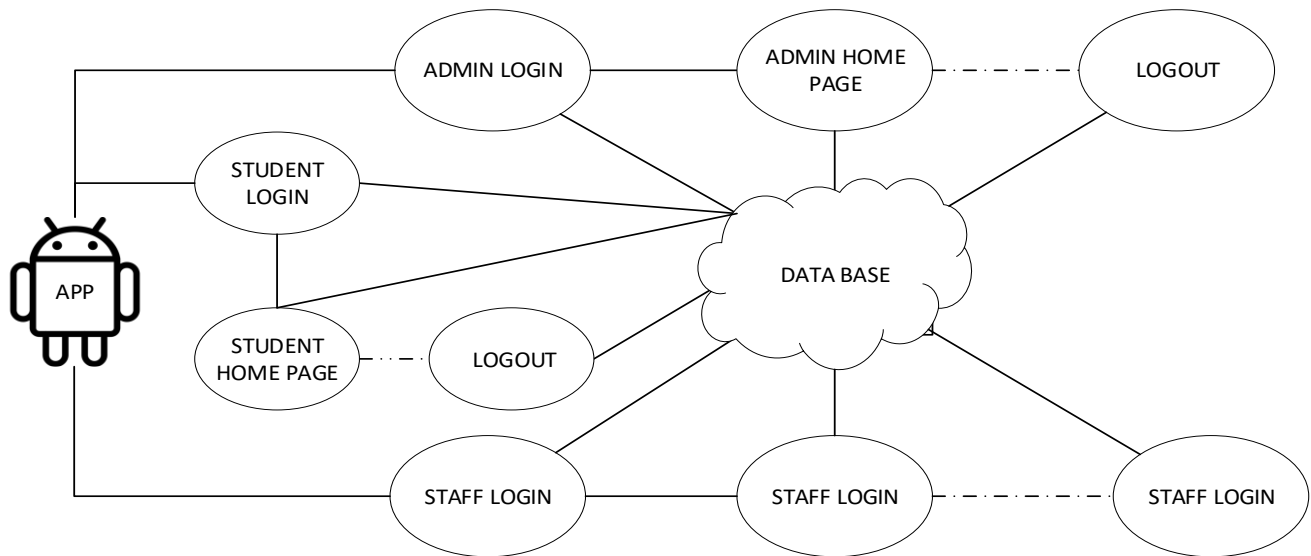
Flow diagram is a collective term for a diagram representing a flow or set of dynamic relationships in a system. The term flow diagram is also used as a synonym for flowchart, and sometimes as a counterpart of the flowchart.

Flow diagrams are used to structure and order a complex system, or to reveal the underlying structure of the elements and their interaction.

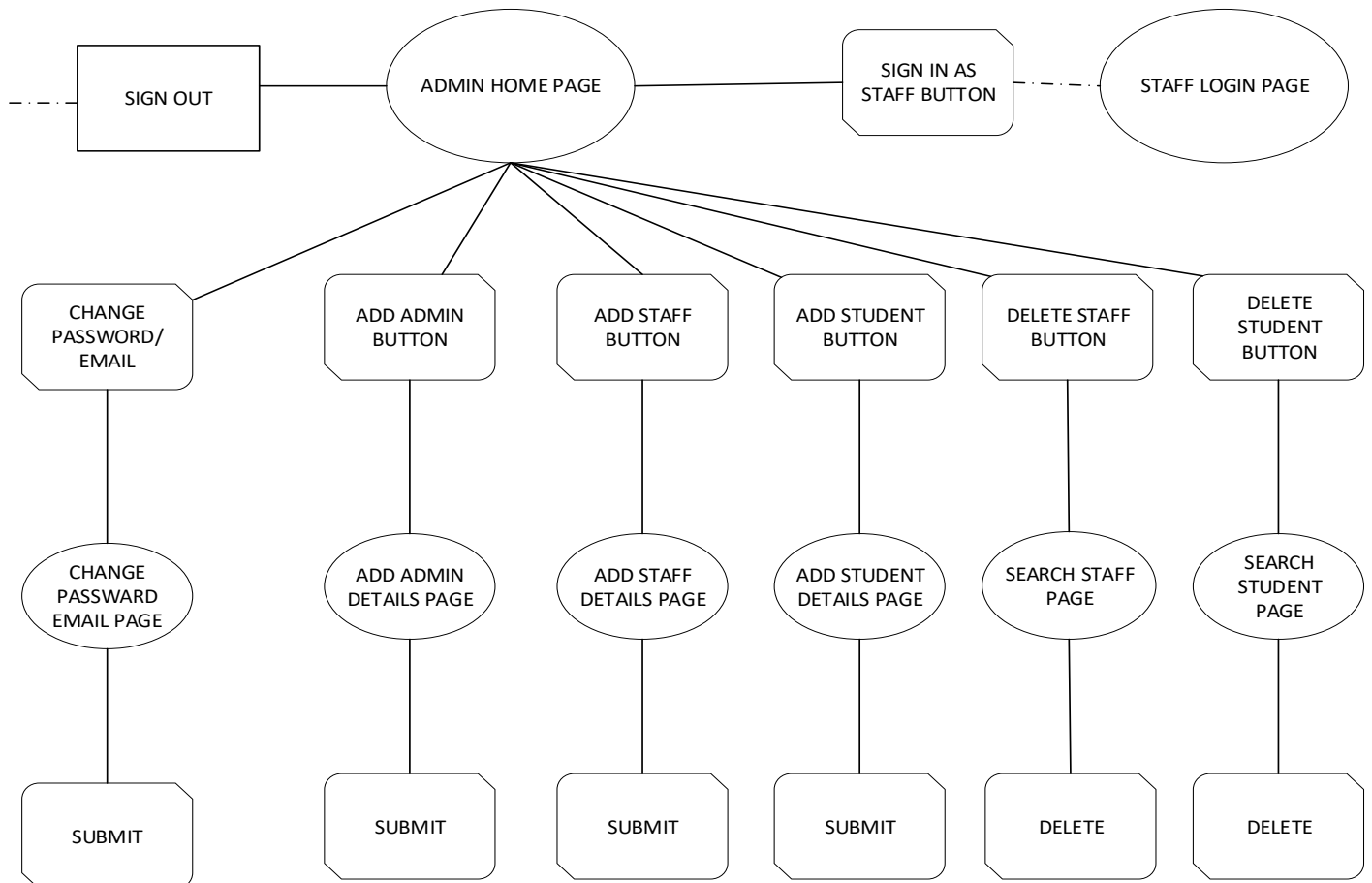
3.2.1 FLOW DIAGRAMS SYMBOLS



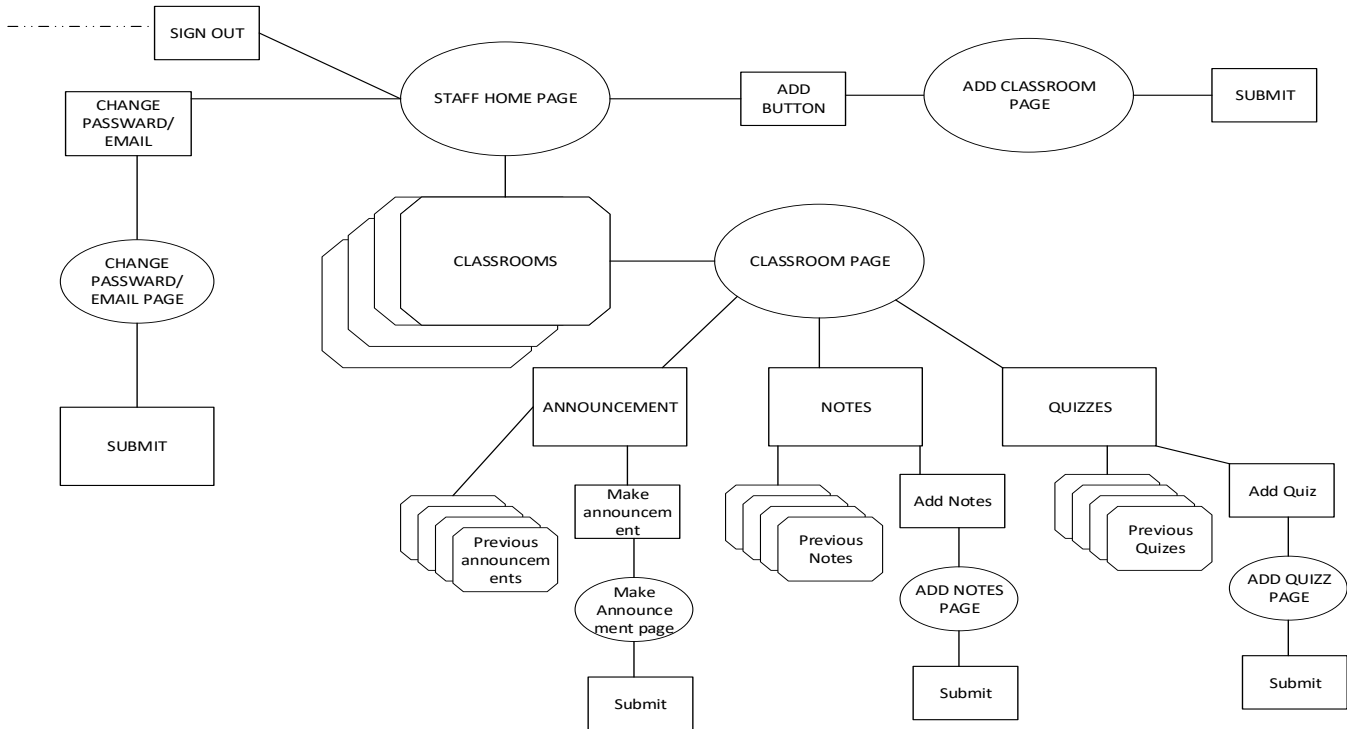
3.2.2 SYSTEM ARCHITECTURE OVERVIEW



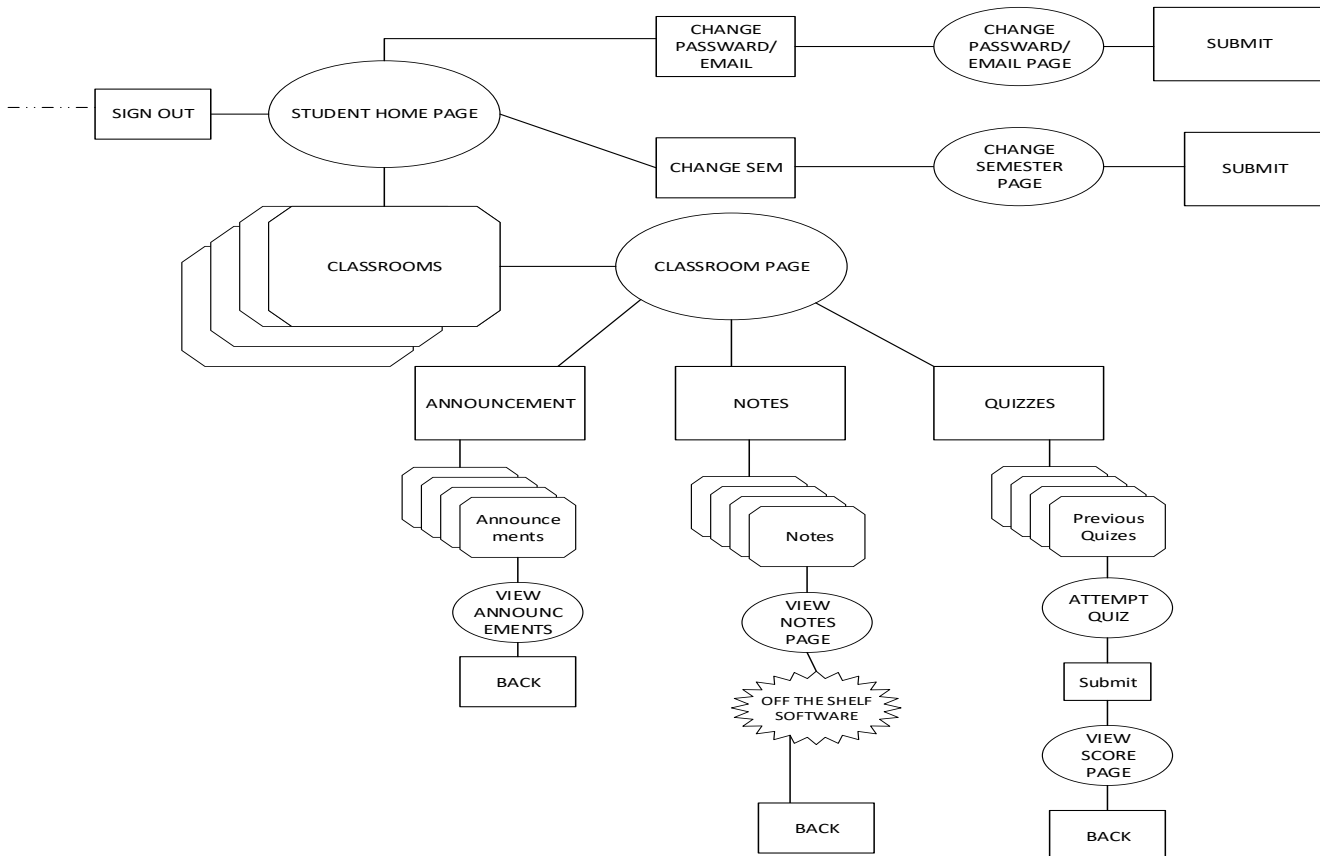
3.2.3 ADMIN MODULE FLOW DIAGRAM



3.2.4 STAFF MODULE FLOW DIAGRAM



3.2.5 STUDENT MODULE FLOW DIAGRAM



3.3 SYSTEM IMPLEMENTATION (Module's description)

I. ADMIN MODULE

- Admin is provided the access for the creation of accounts of new admin, staff and student
- Admin can also sign in as staff as the account of staff is automatically created when a new admin account is created (Note: - All admins are staff not all staff are admins)
- Other than creation of accounts admin has also given access to delete any students and staff's accounts
- Admin can change his email and password if required.

II. STAFF MODULE

- Staff is provided the access to create classrooms by giving details of classroom name, subject code, branch, semester etc.
- Staff can create any number of quizzes, provide notes or make announcements in his classroom
- Staff can change his email and password if required.

III. STUDENT MODULE

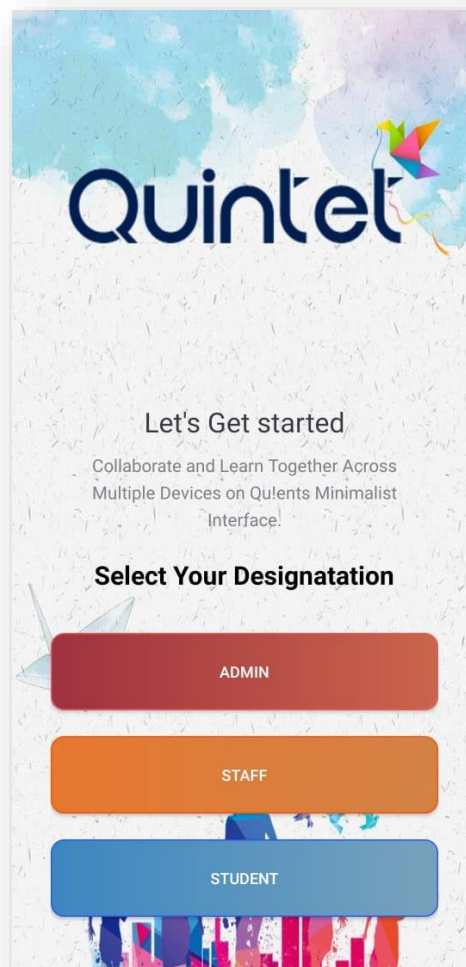
- Student is provided the access to update his semester to access the classrooms of that semester.
- Student can select a classroom and can access the notes quizzes and announcements made in that classroom
- Student can take the quiz and check his score at the end
- Student can change his email and password if required.

4. RESULTS

BEGNING PAGES SNAPSHOTS

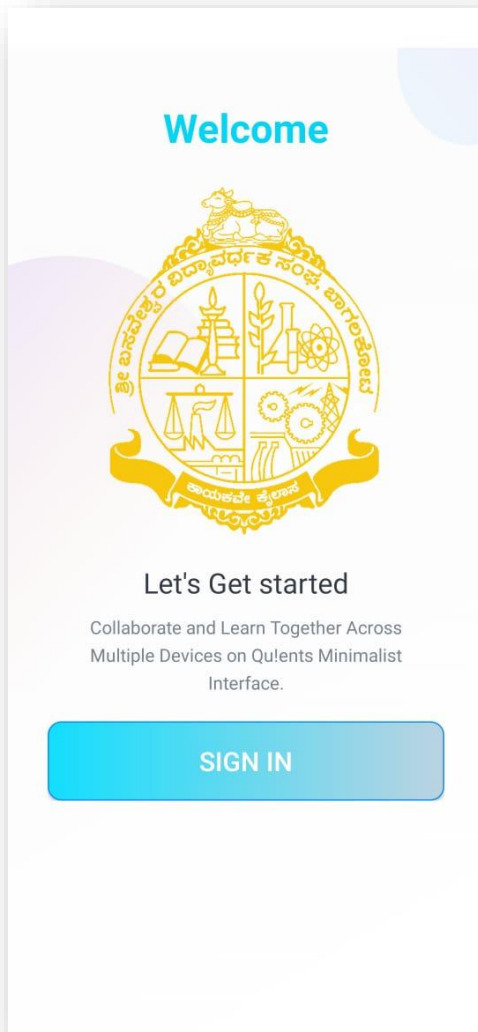


Intro Page

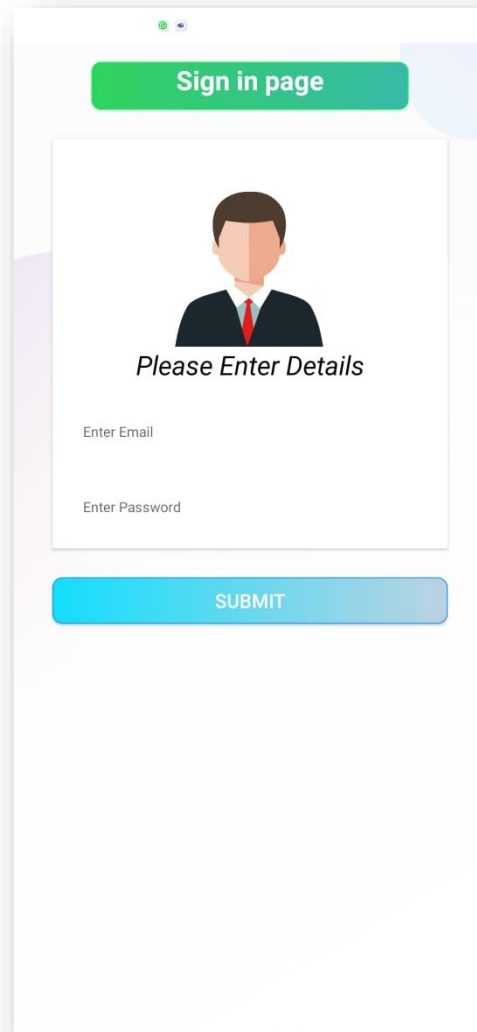


Option selection page

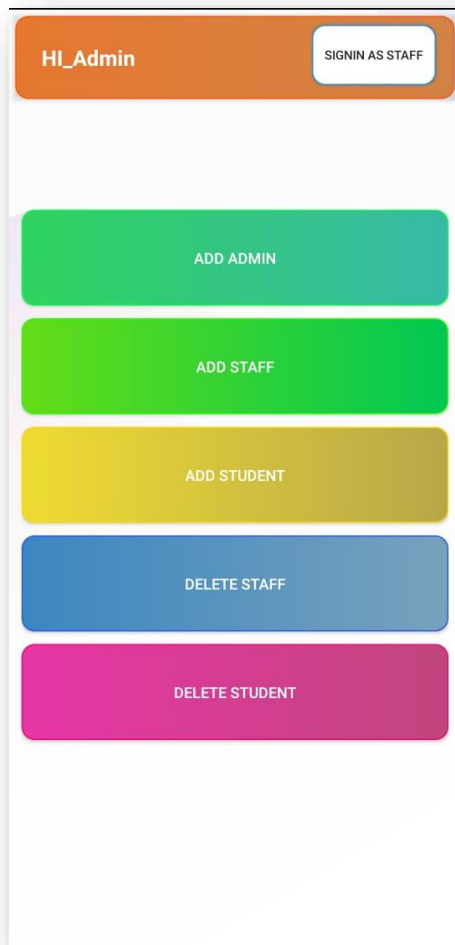
ADMIN MODULE PAGES SNAPSHOTS



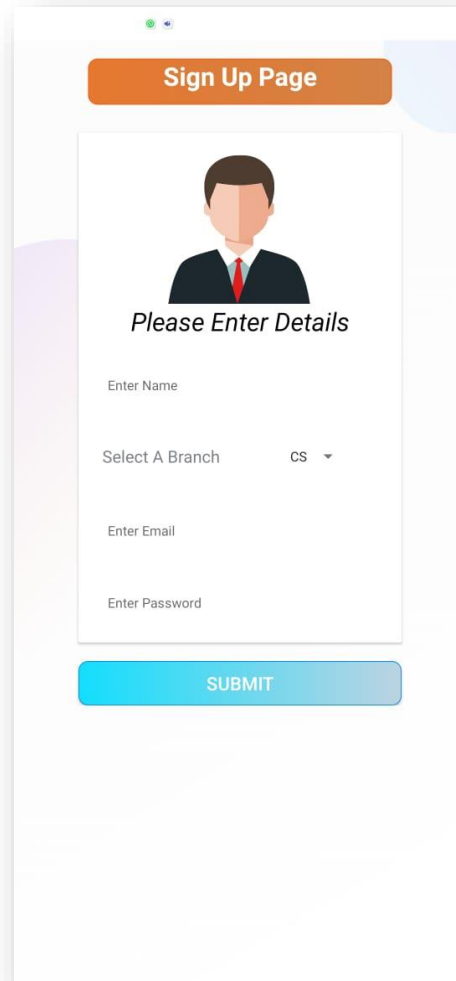
Admin Initial login page



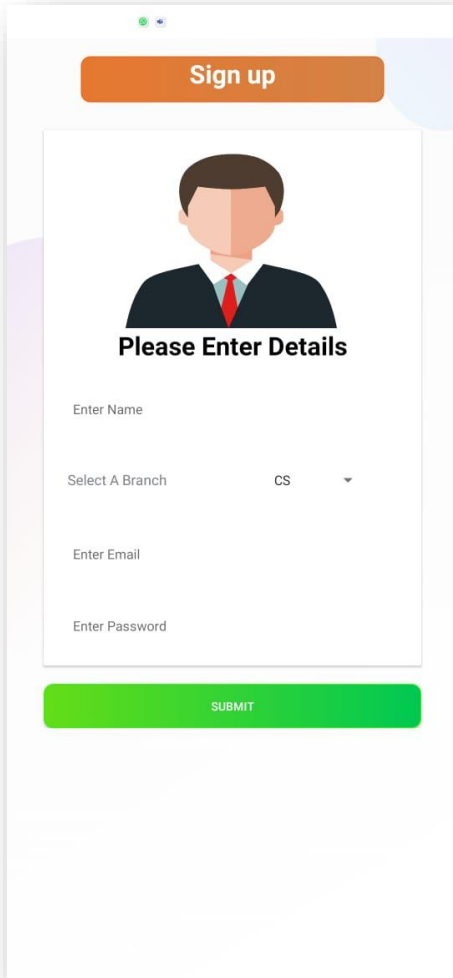
Admin login page



Admin main page

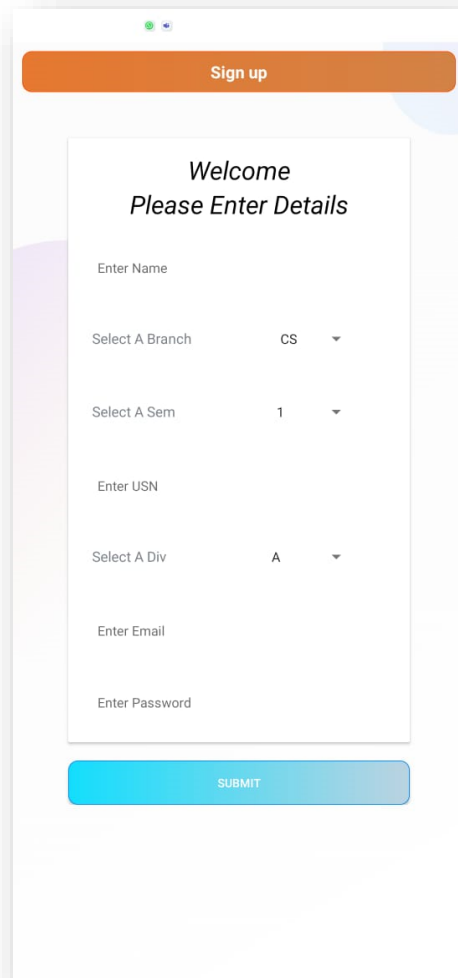


Add new Admin page



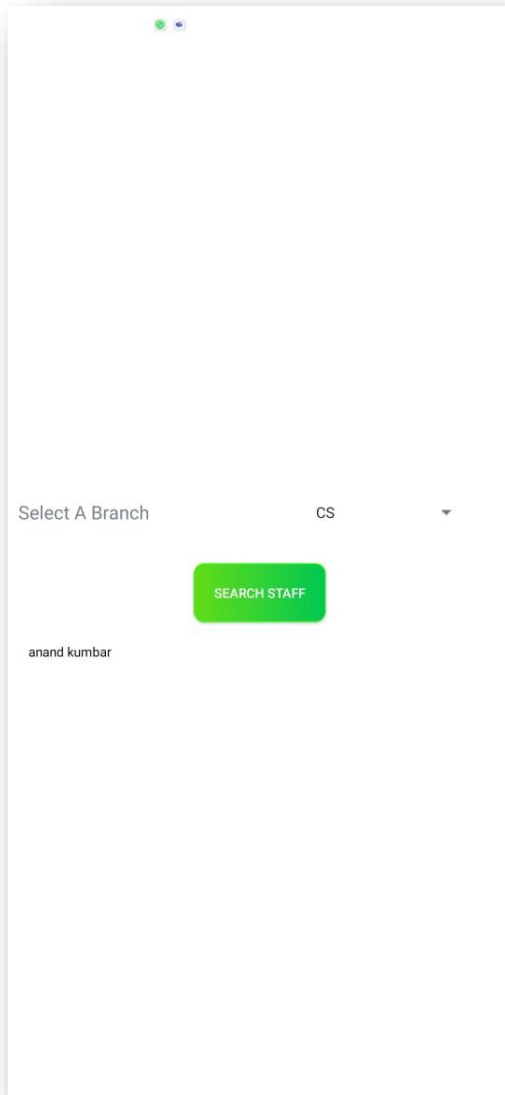
The image shows a mobile application interface for staff sign-up. At the top, there is an orange header with the text "Sign up". Below the header is a white box containing a stylized illustration of a man in a suit and tie. Under the illustration, the text "Please Enter Details" is displayed. Below this text are four input fields: "Enter Name", "Select A Branch" (with a dropdown menu showing "CS"), "Enter Email", and "Enter Password". At the bottom of the white box is a green button labeled "SUBMIT".

Add new staff page

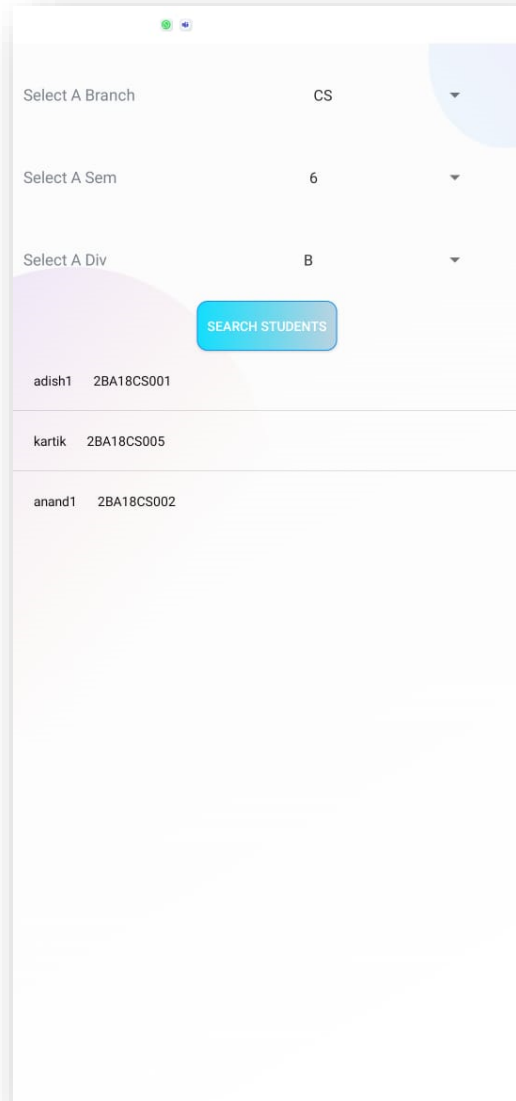


The image shows a mobile application interface for student sign-up. At the top, there is an orange header with the text "Sign up". Below the header is a white box containing the text "Welcome" and "Please Enter Details". Below this text are five input fields: "Enter Name", "Select A Branch" (with a dropdown menu showing "CS"), "Select A Sem" (with a dropdown menu showing "1"), "Enter USN", and "Select A Div" (with a dropdown menu showing "A"). Below these fields are two more input fields: "Enter Email" and "Enter Password". At the bottom of the white box is a blue button labeled "SUBMIT".

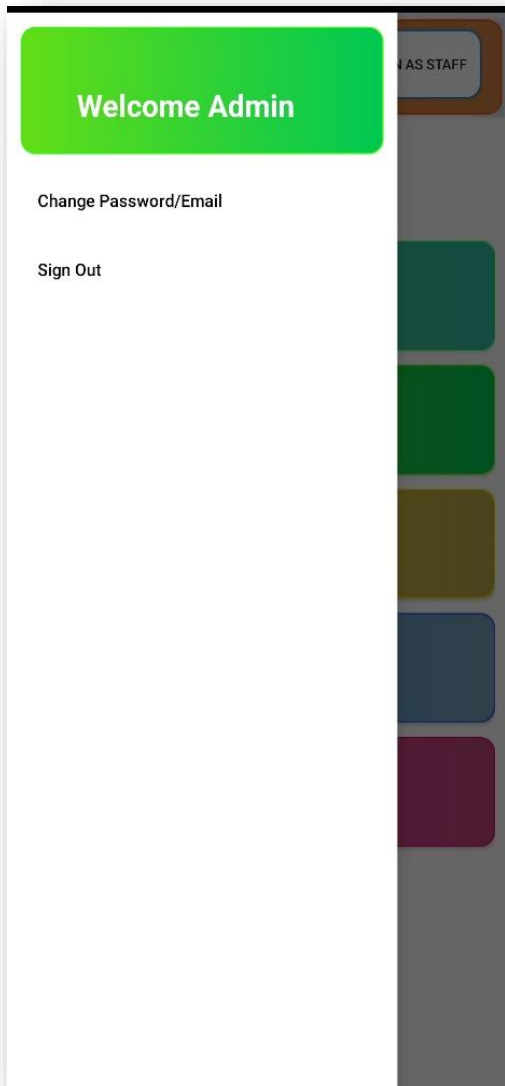
Add new student page



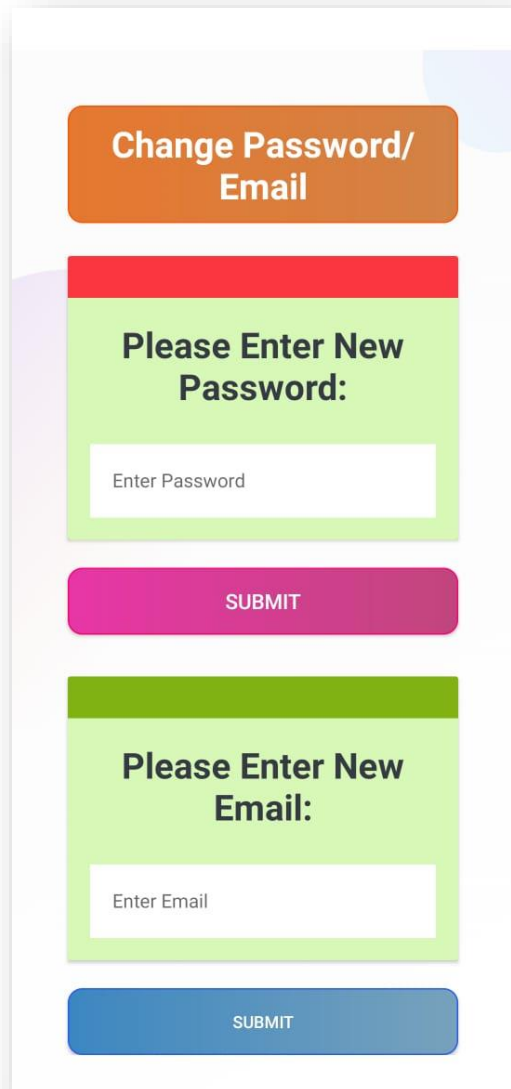
Delete staff page



Delete student page

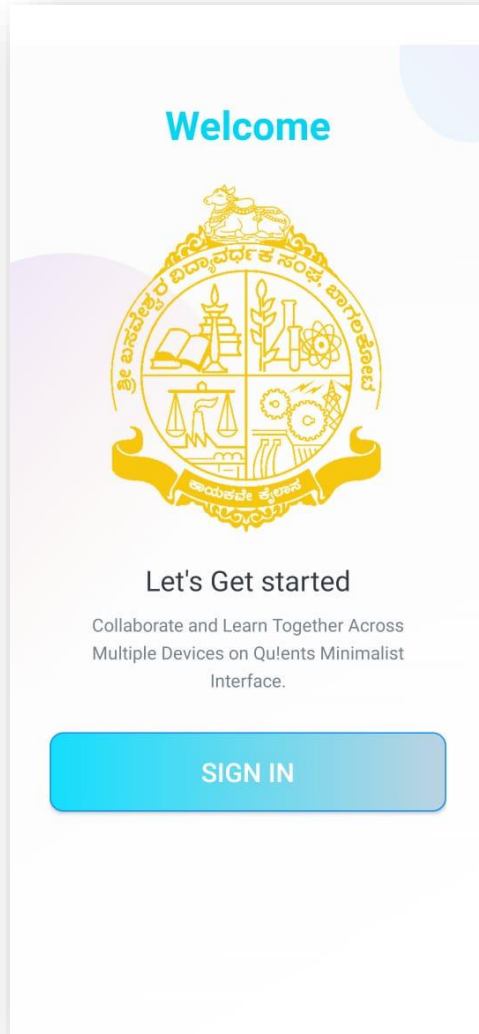


Admin navigation bar

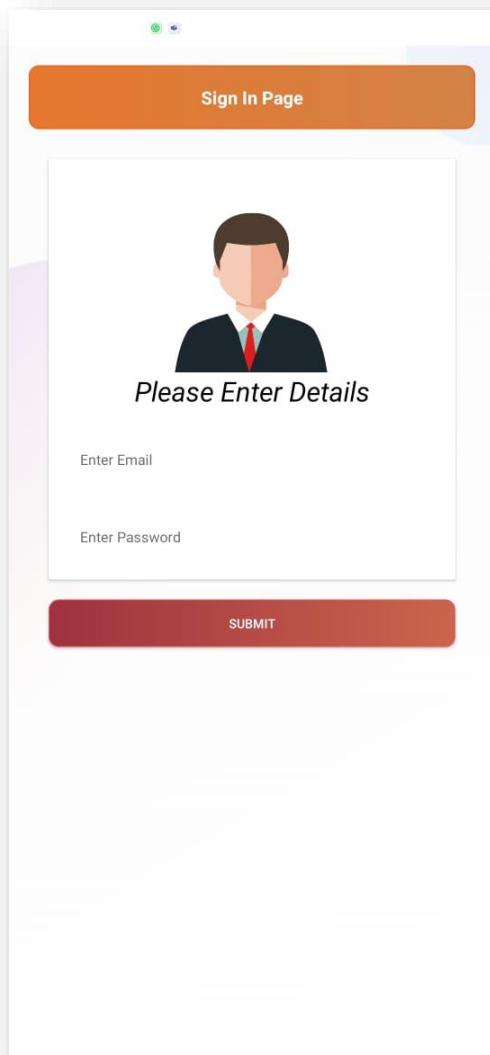


Admin email/password change page

STAFF MODULE PAGES SNAP SHORTS

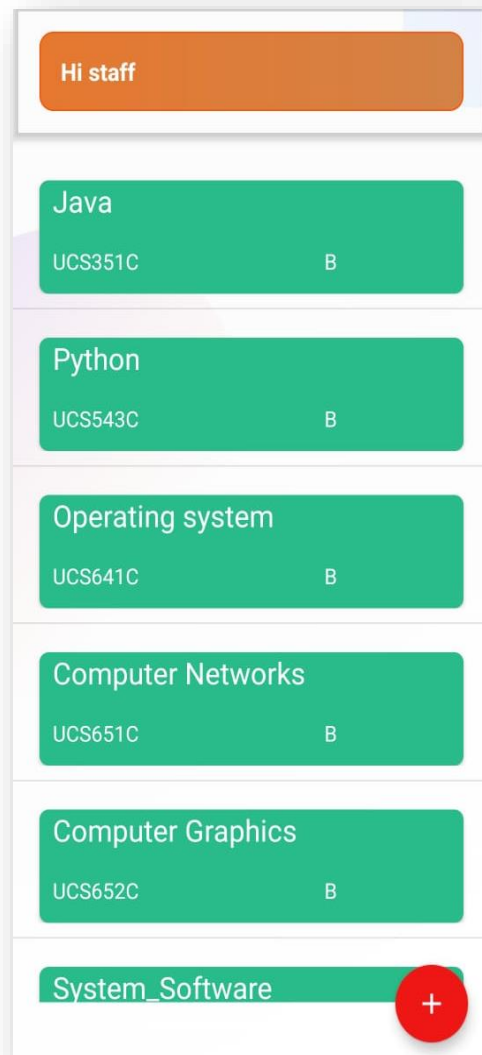


Staff sign in main page



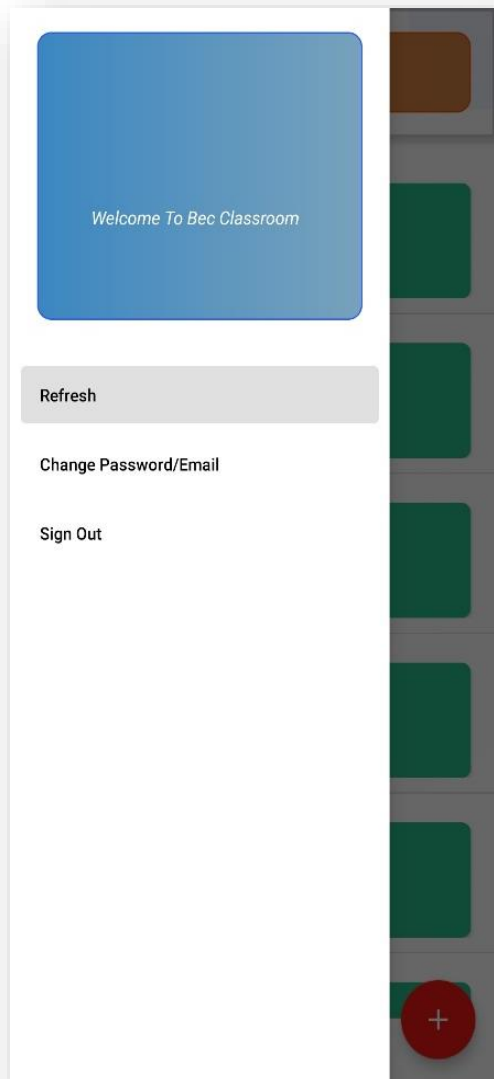
The sign-in page features a white background with a central white box. At the top of this box is a stylized illustration of a man in a suit. Below the illustration, the text "Please Enter Details" is displayed. Underneath, there are two input fields labeled "Enter Email" and "Enter Password". At the bottom of the white box is a red "SUBMIT" button. Above the white box, there is an orange header bar with the text "Sign In Page".

**Staff sign in page
(email and password)**

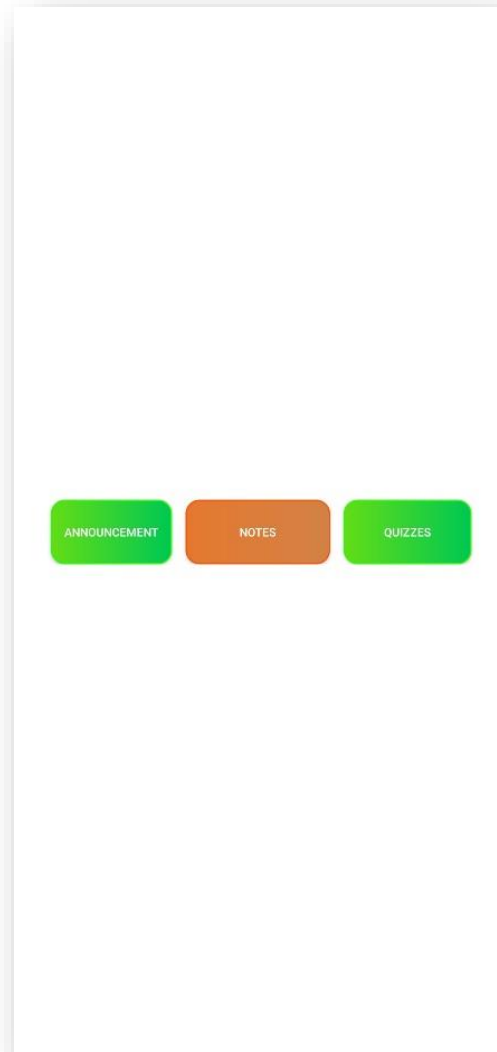


The main page has a white background. At the top is an orange header bar with the text "Hi staff". Below this is a list of subjects, each in a green box. The subjects are: Java (UCS351C, B), Python (UCS543C, B), Operating system (UCS641C, B), Computer Networks (UCS651C, B), and Computer Graphics (UCS652C, B). At the bottom is a green box labeled "System_Software" with a red circular button containing a white plus sign to its right.

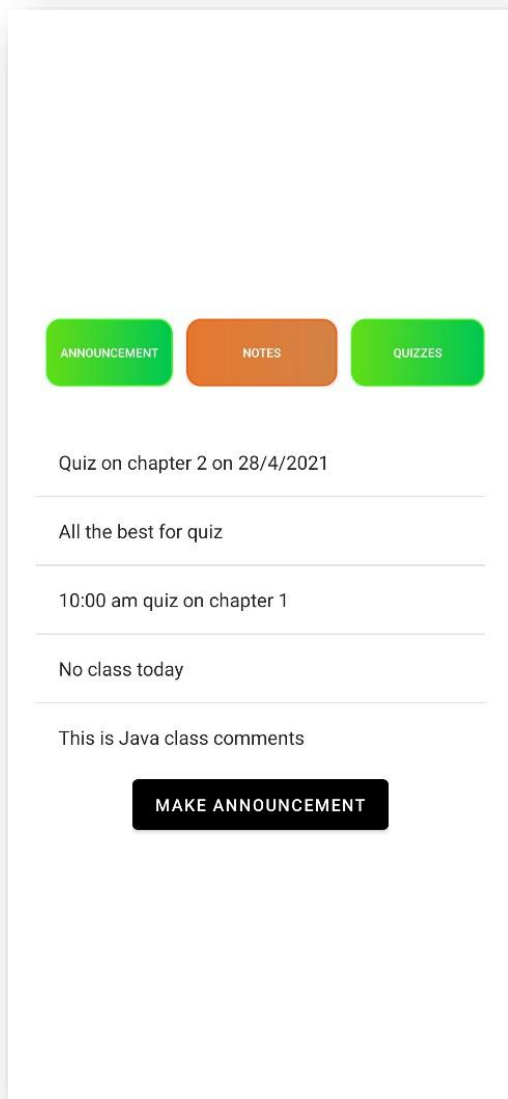
Staff main page



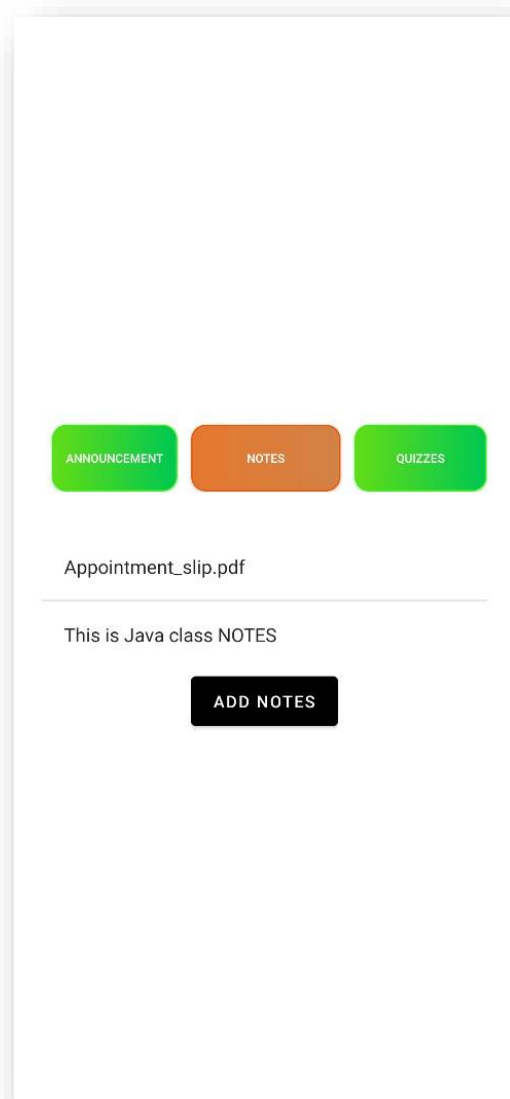
Staff navigation bar



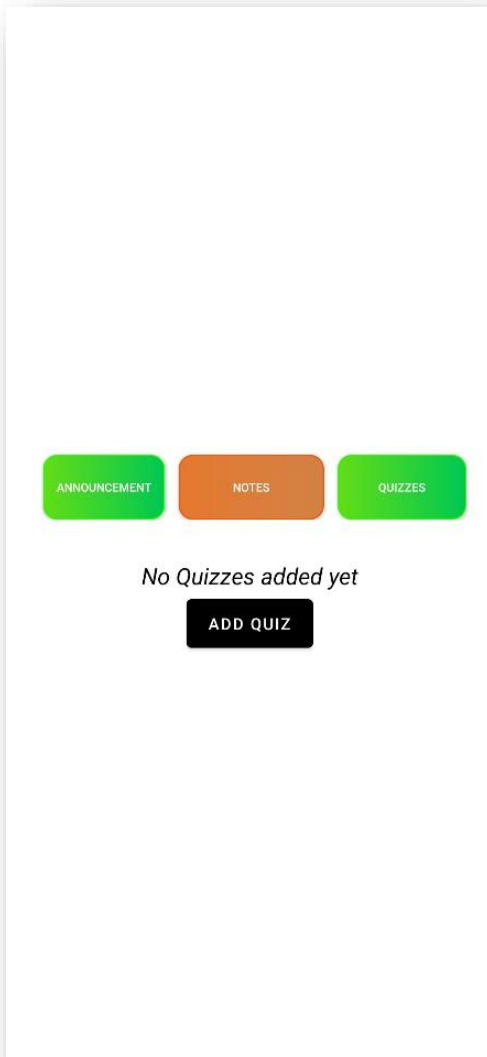
Classroom page



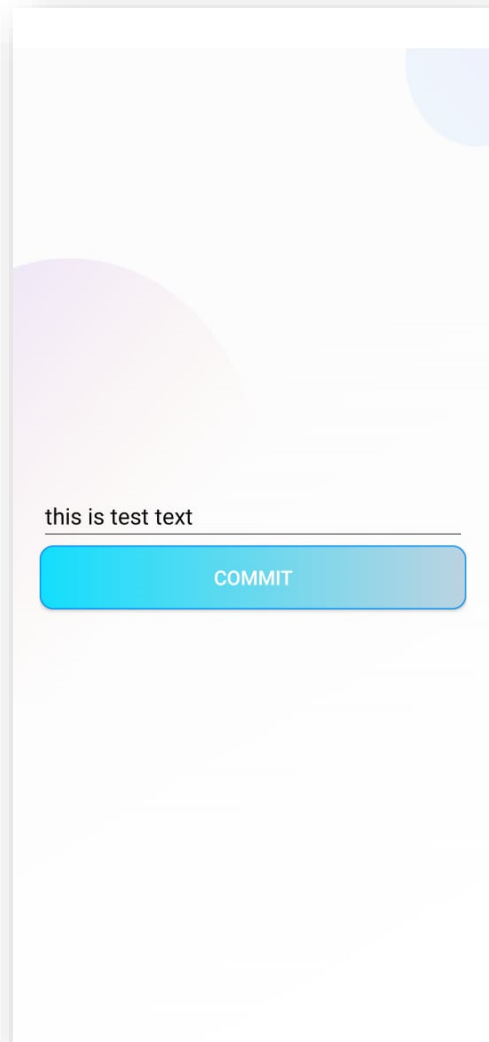
On announcement click



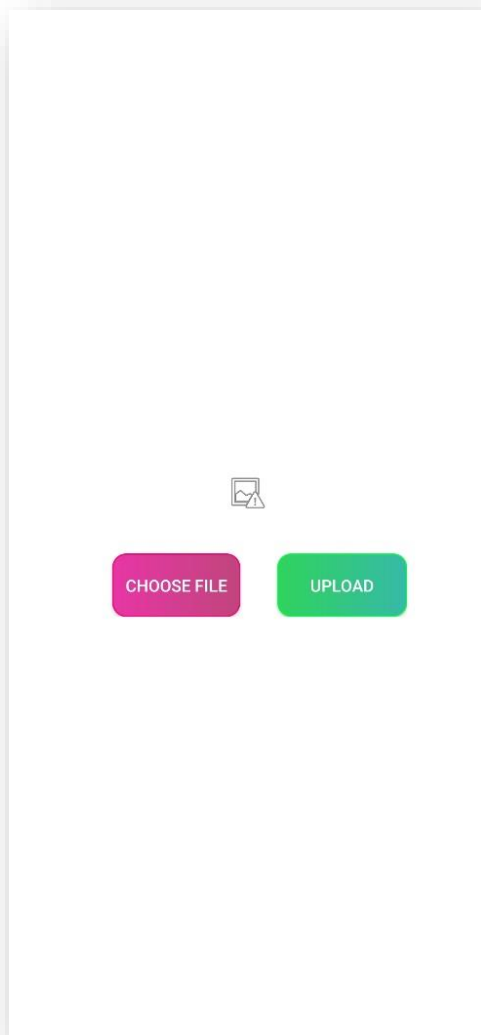
On notes click



On quiz click



Make announcement page



Selection /updating of notes

Quiz Model

Enter Quiz Name

Enter Question

Option1

Option2

Option3


Option4

Correct Option - (Enter Option Number)

ADD QUESTION TO LIST

SUBMIT THE QUIZ

Add quiz questions page



Class Model

Enter The Details

Enter Class Name

Enter Subject Code

Select A Sem 1 ▾

Select A Div A ▾

CREATE

**Create new classroom
Page**

Change Password/Email

Please Enter New Password:

Enter Password

SUBMIT

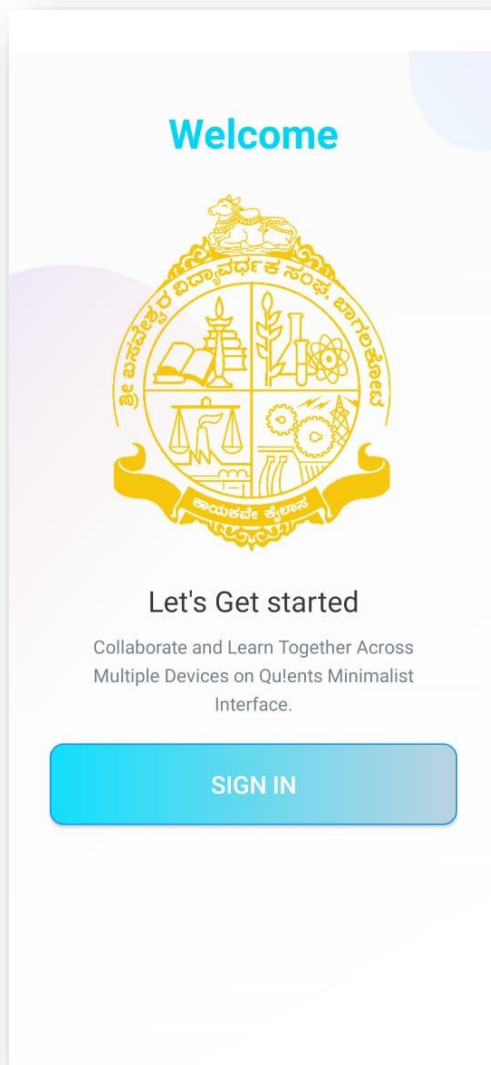
Please Enter New Email:

Enter Email

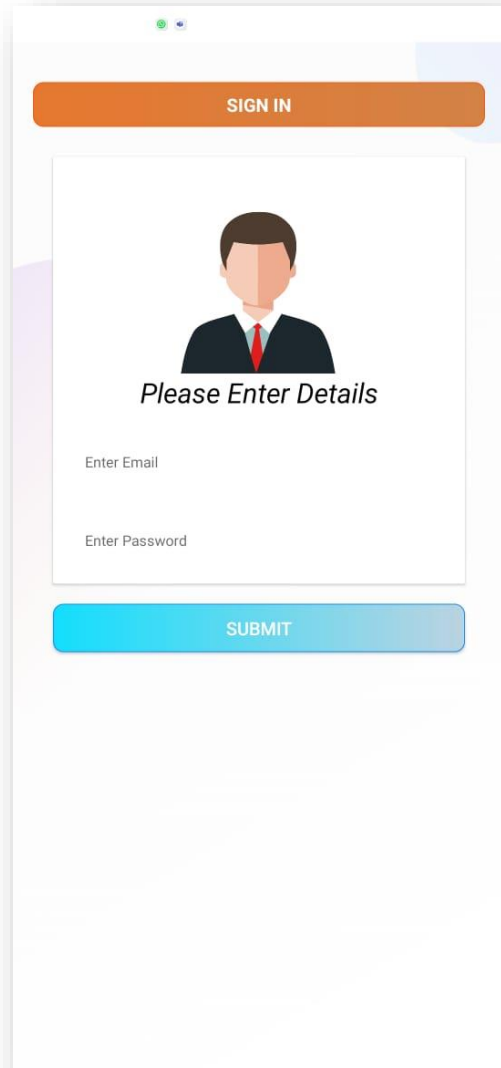
SUBMIT

**Change password/email
Page**

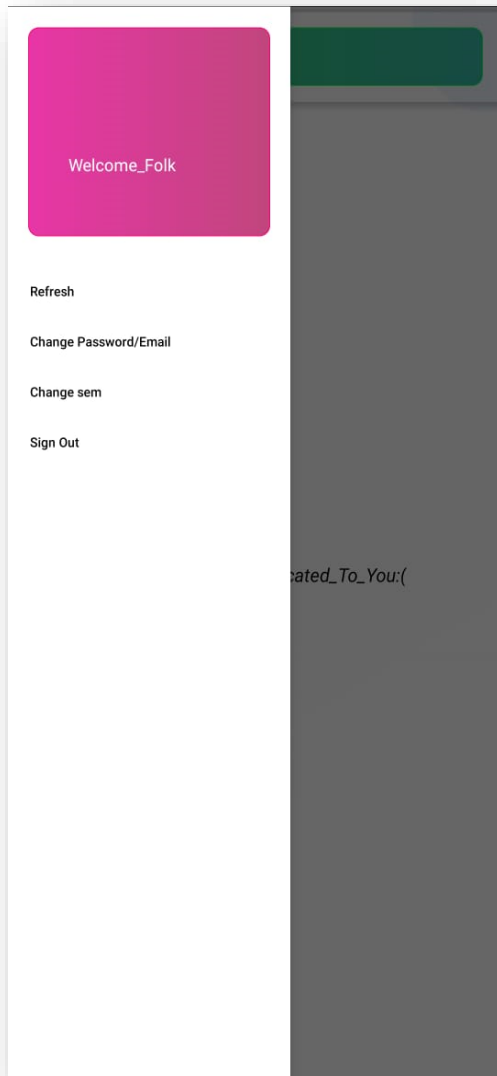
STUDENT MODULE PAGES SNAPSHOTS



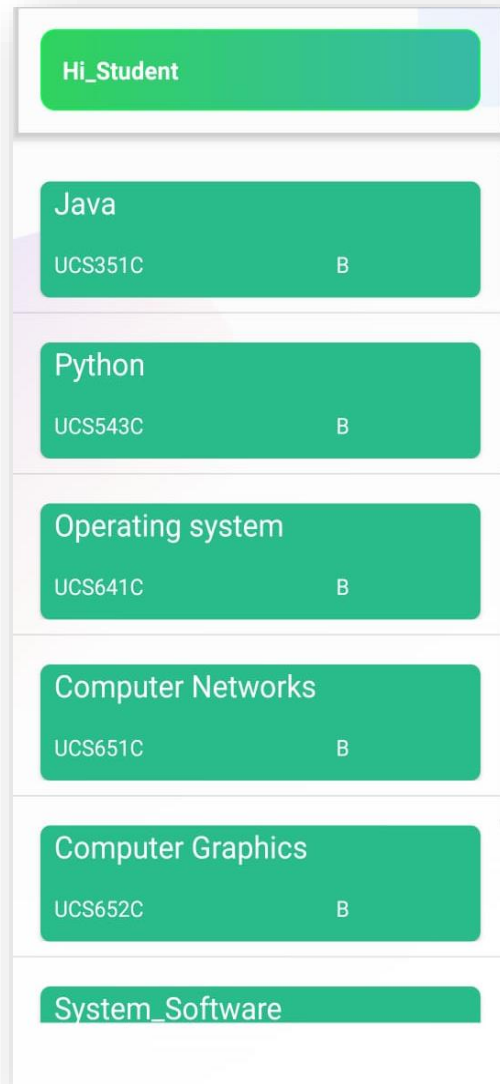
Student sign in main page



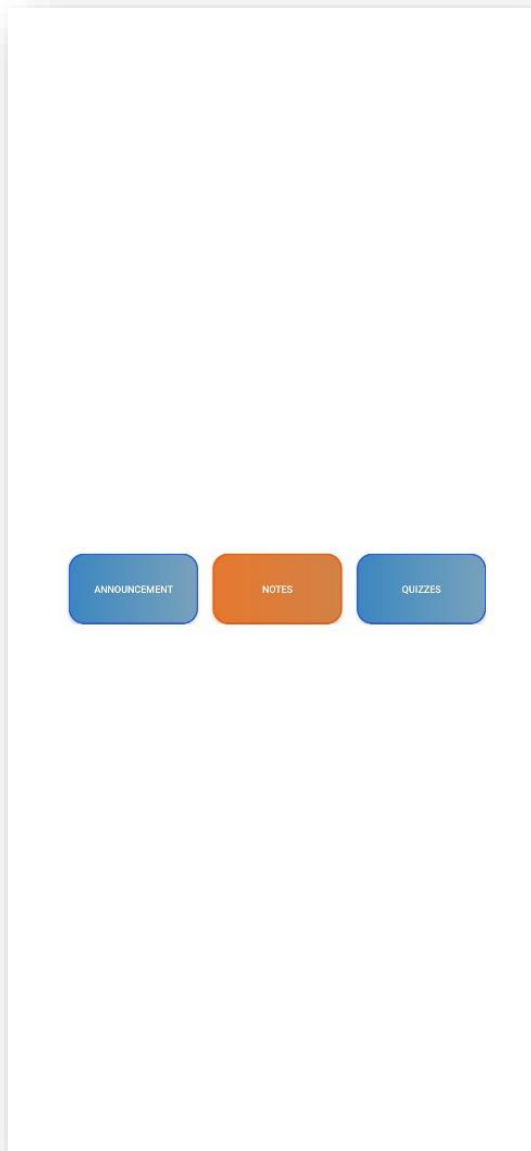
Student sign in page



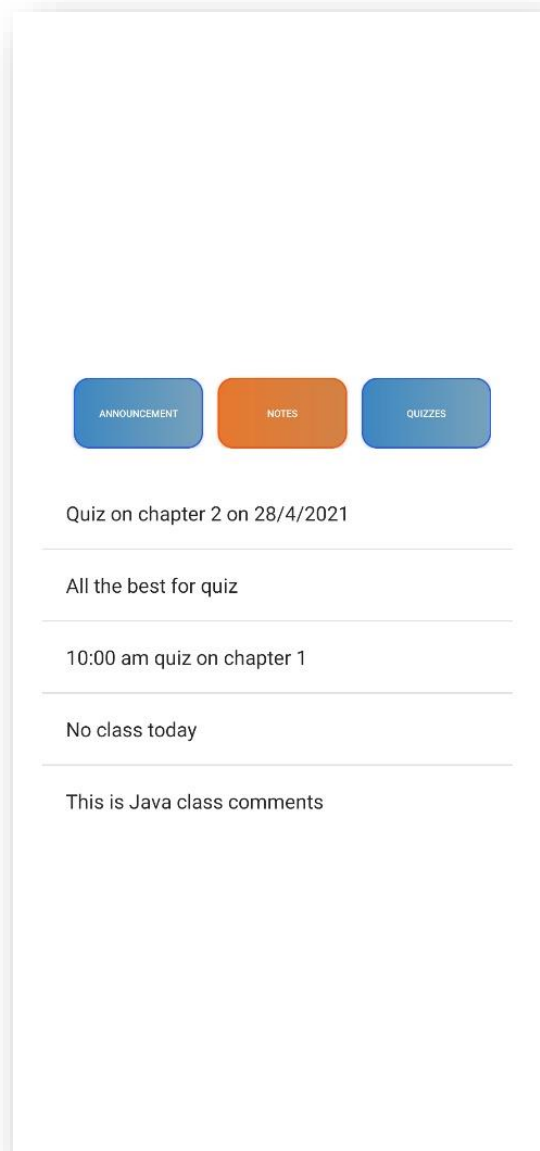
Student navigation bar



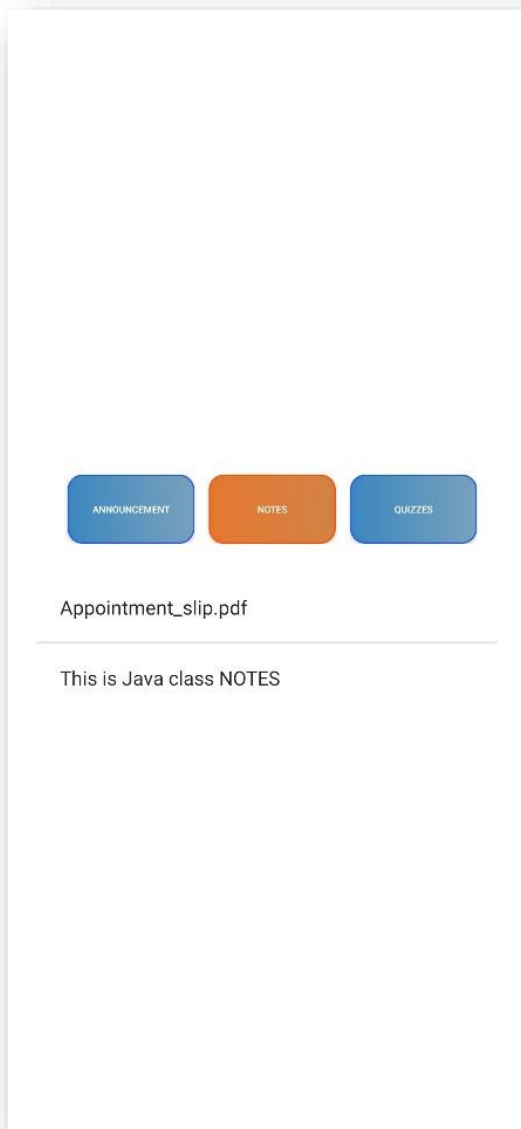
Student home page



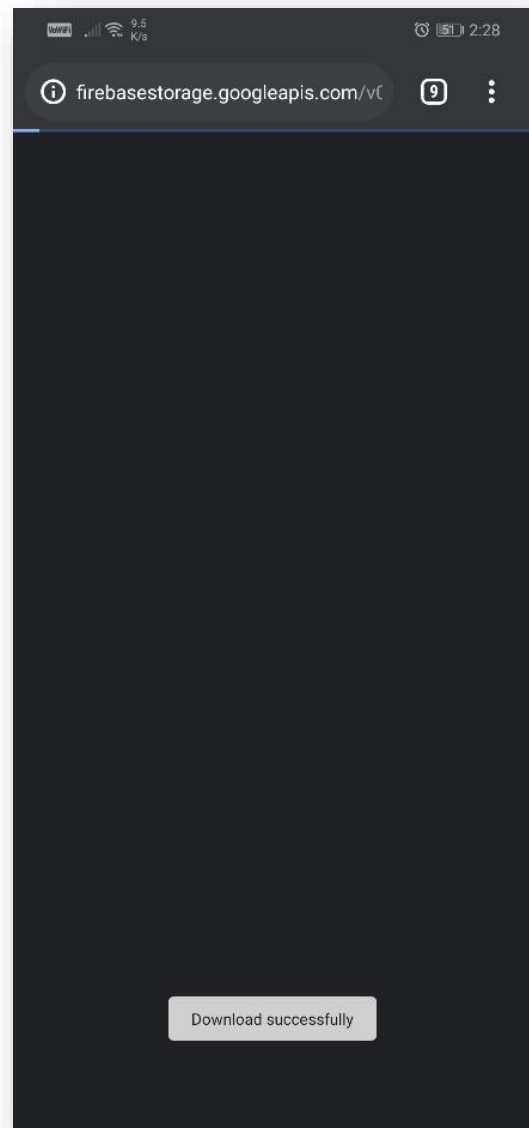
Classroom content page



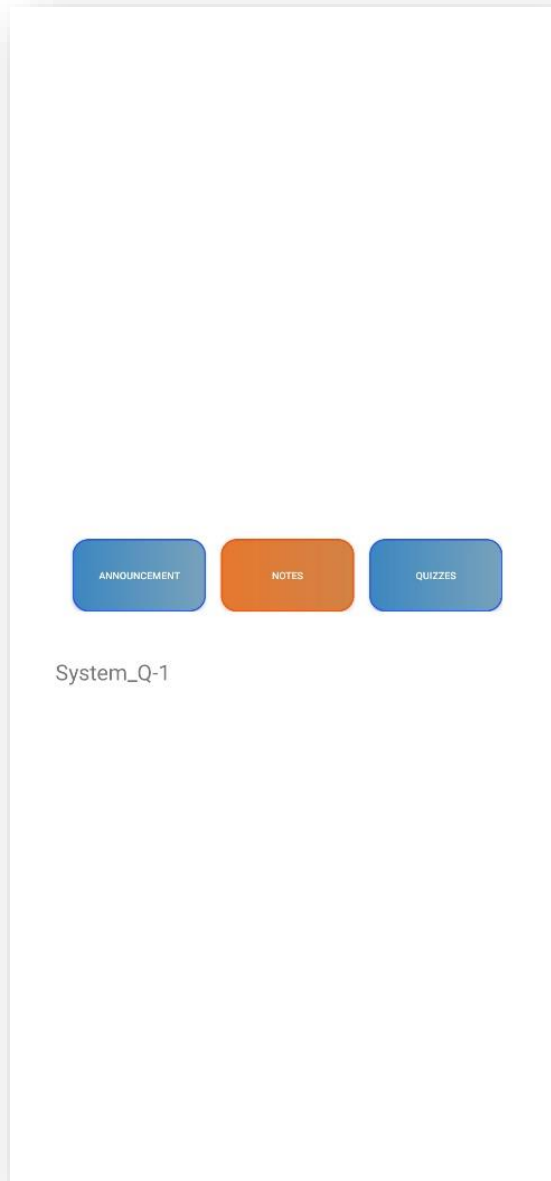
On announcement click



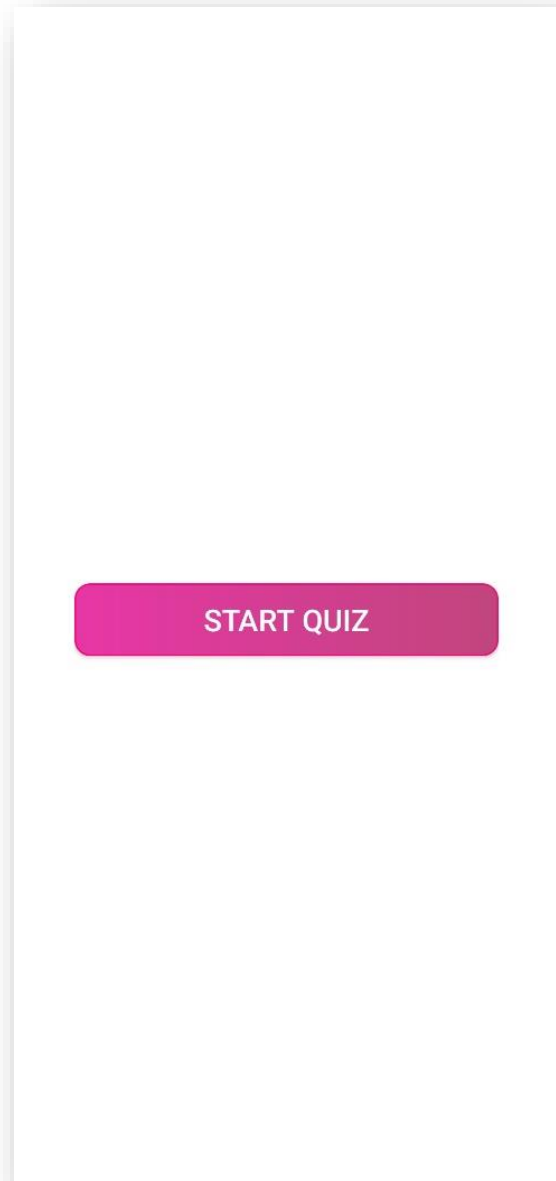
On notes click



Note's view using off the shelf software like browser



On Quiz click



Quiz start page

System_Q-1

What is system software?

Application software

software to run application software

Both A and B

None of these

NEXT



*Congratulations
Your score is :*

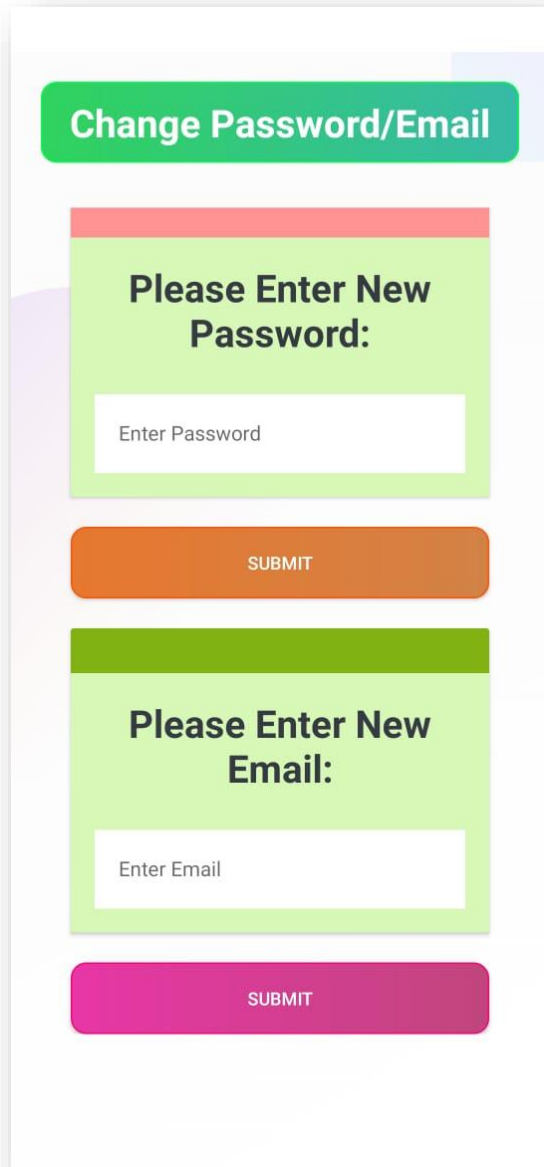
2/2

BACK

Message write success

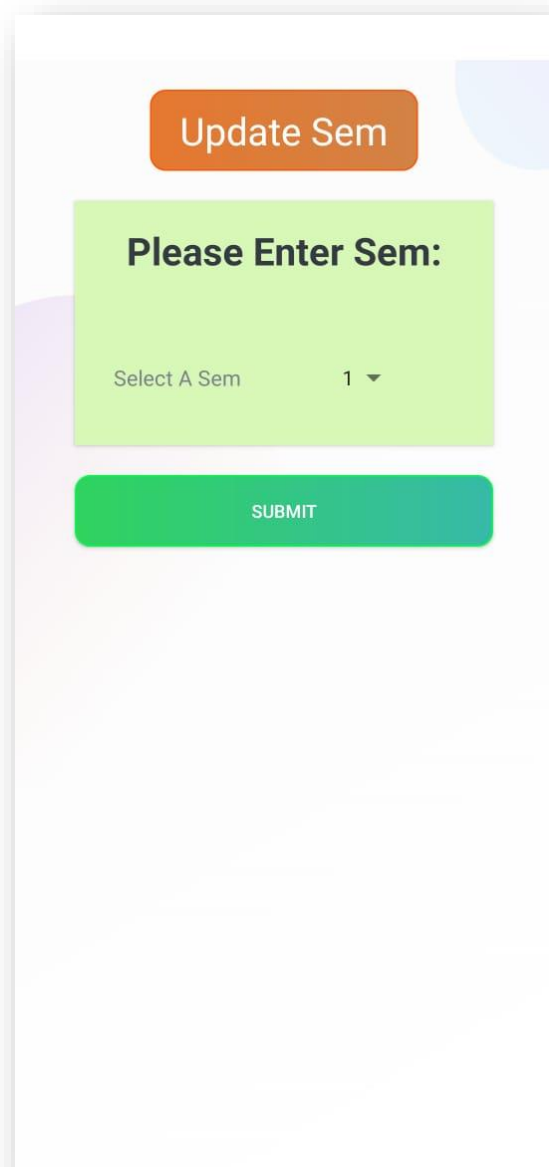
Quiz questions page

Quiz result page



The screen features a green header bar with the text "Change Password/Email". Below this, there are two distinct sections. The first section has a pink header bar, followed by a light green box containing the text "Please Enter New Password:" and a white input field with the placeholder "Enter Password". Below this is an orange "SUBMIT" button. The second section has a green header bar, followed by a light green box containing the text "Please Enter New Email:" and a white input field with the placeholder "Enter Email". Below this is a pink "SUBMIT" button.

Change password/ email page



The screen features an orange header bar with the text "Update Sem". Below this is a light green box containing the text "Please Enter Sem:" and a dropdown menu with the text "Select A Sem" and a value of "1" with a downward arrow. Below this is a green "SUBMIT" button.

Update semester page

5. CONCLUSION & FUTURE SCOPE

We have provided this android application with basic features that is required to carry-out basic activities of class-room with simple user-interface and other basic features. Due to its mobility and easy access it can help students in such pandemic.

As of future scope we will provide with new features such as resume-building, attendance tracker, to create assignment and marks assigner for both quiz and to assignment etc. Since it is college specific we can modify and provide the only features required which holds the college users from switching to other applications.



6. REFERENCES

1. <https://developer.android.com/training/basics/firstapp>
2. <https://developer.android.com/courses/pathways/android-basics-kotlin-one>
3. <https://developer.android.com/training/app-links>
4. <https://firebase.google.com/docs/android/setup>
5. <https://firebase.google.com/docs/projects/learn-more>
6. <https://firebase.google.com/docs/firestore/security/get-started>
7. <https://www.xmlfiles.com/xml/xml-intro/>
8. https://www.w3schools.com/xml/xml_what.asp
9. <https://stackoverflow.com/questions/2201917/how-can-i-open-a-url-in-androids-web-browser-from-my-application>
10. <https://medium.com/@harshithdwivedi/set-up-firebase-to-upload-user-files-165063cbeee1>