

STUDENT PORTAL

Mr. Harshavardhan J R¹, Mr. Suhaib Z Ahmed², Mr. Syed Mohammed Wasi³

¹Associate Professor, Department of Computer Science & Engineering, K. S. Institute of Technology, Bengaluru-62, India

^{2,3}Fourth Year B.E, Department of Computer Science & Engineering, K. S. Institute of Technology, Bengaluru-62, India

Abstract : *Social media is a platform where the general public communicates with each other. But there are no proper source or media which allows students/teachers to communicate with students/teachers of other colleges that are under VTU(Visvesvaraya Technological University). So, in this paper, an Integrated Android Application is described which allows students as well as teachers to communicate with students and teachers of other colleges under VTU and also provides them with additional features and functionality.*

Index Terms: *Integrated, Android Application, Social media.*

1. INTRODUCTION

Social media applications are used for posting and sharing information. It is used by a large number of students as well as teachers. But when it comes down to sharing information related to achievements in academics or other college related activities like Hackathons, Quiz competitions etc., social media applications like Facebook does not provide the right platform as they go unnoticed by other students/teachers. Hence, to provide a platform where students as well as teachers can interact with other students and teachers, share student related events, ask queries, share/upload notes etc., an integrated android application is built to provide all of these features and functionality.

2. LITERATURE SURVEY

There have been several applications and websites developed that are focused on student related activities :

[1] Smart Campus is a mobile as well as web application. It's main objective is to develop an application for the execution of several academic operation to provide students with information regarding complaints, any placement activities, general notices, and important notices regarding all departments. But it is applicable for only one University.

[2] University Portal portals integrate campus-specific information and activities which is stored in the campus electronic vaults such as databases, file systems and existing application systems, with unstructured data. But lack of or delayed communication can hamper progress and slow down the rate at which work is done.

[3] Student App for GDCST is another web application that provides students, teachers and administration to do all their study related activities without login to different websites. But it has limitations like students can only receive messages and not send them and the university has to create the student's account.

3. PROPOSED SYSTEM

3.1 WATERFALL MODEL :



Fig 3.1 - Waterfall Model

As shown in the figure 3.1, a simple Waterfall model is used for developing the application. A waterfall model is used as it is a simple and easy to use process model where the whole process is divided into separate phases. Each phase must be completed before the next phase can begin and the outcome of one phase serves as an input for the next phase. The various phases that are used for developing this application are as follows :

1. **Project Planning** : The first step of any project development is planning. The aim and objectives of the project were formed in order to get a clear picture of what is to be developed.
2. **Requirements** : All the requirements necessary for the creation of the android application were noted. The requirements include – system requirements, mobile phone requirements, technologies needed, software and hardware requirements.
3. **Analysis** : After project planning and requirements was known, they were further analysed and studied in order to understand their working and make it easier to develop the application.
4. **Design** : The design of the application is developed. The main objective of the design is to make the application user-friendly where the user can interact with each of its components with ease.
5. **Coding** : After the design is plotted, set of codes are written to develop each module of the application. Codes are written for both front-end as well as back-end of the application. Coding is done in Java and XML programming languages.
6. **Testing** : The codes are executed in order to run the application. Any bugs or errors that are discovered in the code are corrected in order to run the app smoothly and correctly. Each and every component is run and tested for errors and finally we obtain the proper and fully functional application.
7. **Deployment** : The application is finally deployed and used by different users.

3.2 CLASS DIAGRAM

Class diagrams are the main building blocks of object oriented modelling. It describes the structure of a system by showing their classes, attributes, operations (or methods), and the relationships among objects. In this application, various activities which are built are shown in the figure below :

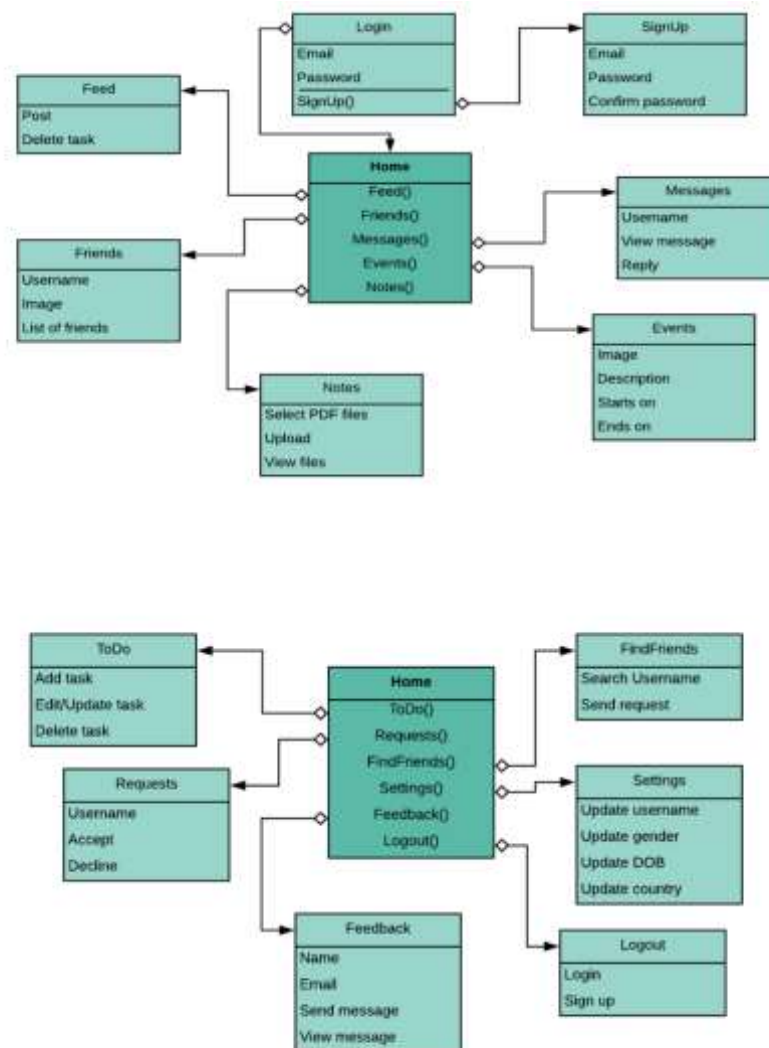


Fig 3.2 - Class diagram showing different activities

4. OVERVIEW

A brief description of all the activities provided by Student Portal application are as follows :

- i. **Login Activity** – Students as well as teachers can login directly to the app if they already have an account created and verified by entering their Email and password.
- ii. **Sign up Activity** - If the students and teachers don't have an account, they need to create on by using the sign up feature provided in the app. They are required to give their Email ID and a unique password and a verification mail will be sent to them to verify their account.
- iii. **Home Activity** - This is the home page of the app where the user will be taken after successfully logging in. It contains the post feed as well as a drawable menu bar which contains a list of other activities.
- iv. **Feed Activity** - This activity allows the user to post on the Home page of the application which can be viewed by the public. The user can also update and delete the post whenever required. The user can view the number of likes his/her post has got and also read and reply to comments.

- v. **Friends Activity** – This activity displays the list of friends of the user and shows if they are online or offline.
- vi. **Message Activity** – This activity displays all the chat messages that the user has sent to or received from his/her friend.
- vii. **Events Activity** – This is the activity where the user can post any events like college fests, quiz competitions, Hackathons etc. He/she can give a description about the even and also mention the start date and the end date of that event.
- viii. **Notes Activity** – The student can upload and share subject notes in this activity. Other users who require these notes, can view and download them easily.
- ix. **To-do list Activity** – The user can create a to-do list which is basically his/her list of activities that he/she wants to do in the near future.
- x. **Requests Activity** – The user can view any friend requests that he/she has received and can accept or reject them.
- xi. **Find Friends Activity** – This activity allows the user to find their friends by entering their username and can also send friend requests to them.
- xii. **Settings Activity** – The user can view and edit his/her personal details like username, status, country, date of birth. Once edited, it will be immediately updated.
- xiii. **Feedback Activity** – The user can send a feedback about the app using this activity. He/she can enter his/her mail and message and send it to the server.
- xiv. **Logout Activity** – This allows the user to logout of the app. All the user data is backed up and then the user is logged out of the app.

5. IMPLEMENTATION

The Student Portal app is written in Java using an IDE called Android Studio. **Firestore** which serves Backend-as-a-Service, provides us with a number of tools and services.

The Proposed System requires some of the following:

Firestore Cloud Messaging: A cross-platform solution for messages and notifications.

Firestore Auth: A service that authenticates users based on Email and Password login stored with Firestore.

Firestore database: A service that provides us an API allowing application data to be stored on the Firestore's cloud.

Firestore Storage: This service provides secure file uploads/downloads to our proposed system. We use it to store images and user uploaded files such as Notes.

The Gradle build system in Android Studio makes it easy to include these Firestore Services to our build as dependencies.

Dependencies also include other library modules such as Picasso, CircularImageView, and Android Image Cropper that help us display images in our app, enable us to have a circular image and also crop certain images straight from the app.

One of the main aspects of this proposed system is the Android Manifest File. **Android Manifest** describes essential information about our app to the Android build tools, and the Android operating system. Among many other things, the manifest file declares the following: **app's package name, components of the app**, such as activities and **Permissions** that our app needs in order to get access to the protected parts of the system.

Since we use Firebase (Backend-as-a-service), we require the following permissions.

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

The Users Data such as his profile information, posts or events updates can be retrieved on the app with help of a **Firestore RecyclerView Adapter** which is a generic way of backing an RecyclerView with a Firestore location. It basically assembles and arranges the received data in order, into the given class type. We pass in all the required parameters to its subclass and implement the populateViewHolder method.

```
public abstract class FirestoreRecyclerViewAdapter<T,VH extends
android.support.v7.widget.RecyclerView.ViewHolder>
extends android.support.v7.widget.RecyclerView.Adapter<VH>
```

We are also required to retrieve other users data on to our app that helps current user finding other existing Users With help of an Query. The Query class are quietly used for reading data. They get triggered when the corresponding data changes. Instances of an Query are obtained by calling startAt(), endAt(), or limit() on our Firestore Database Reference.

```
Query searchPeopleQuery =
allUsersDatabaseRef.orderByChild("fullname").startAt(searchBoxInput).endAt(searchBoxInput + "\uf8ff");
```

6. RESULTS

The snapshots of the Student Portal Application are shown below:

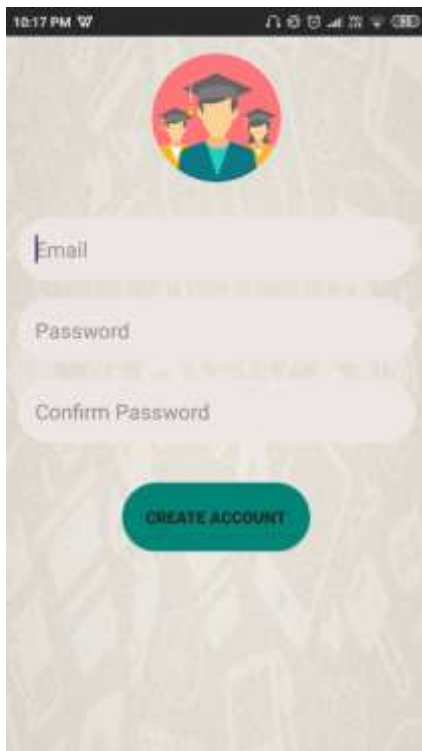


Fig 6.1 - Register/Signup Activity

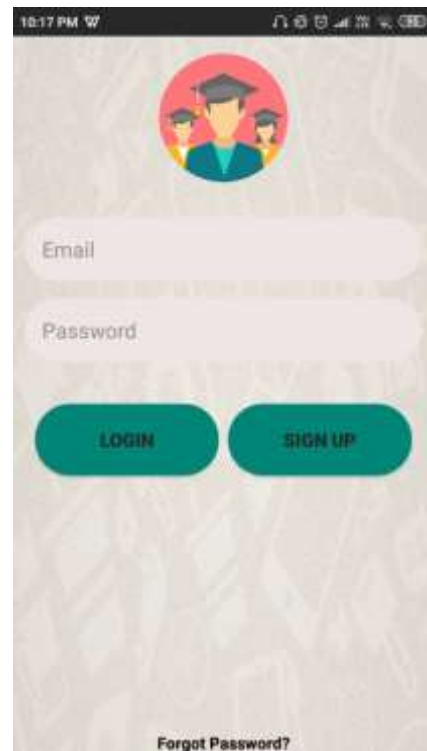


Fig 6.2 - Login Activity

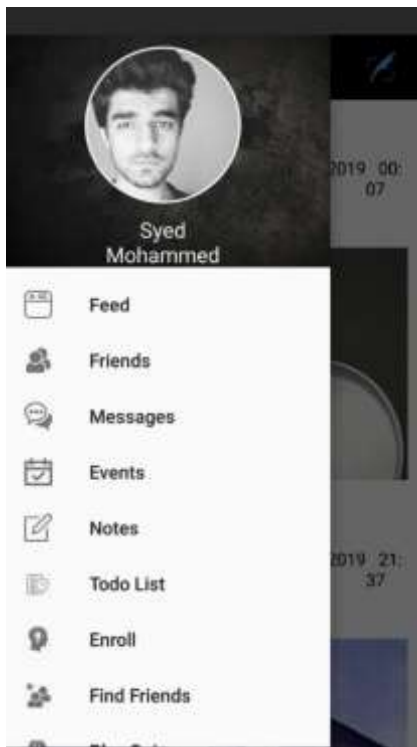


Fig 6.3 - Home/Drawer Menu

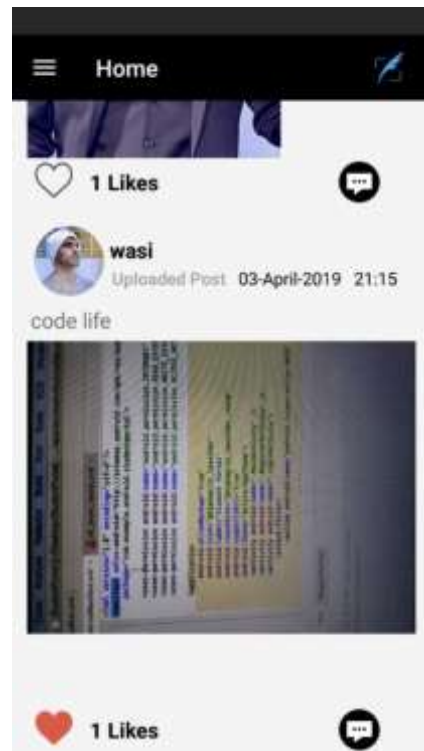


Fig 6.4 - Post Feed



Fig 6.5 - Events Activity



Fig 6.6 - Message Activity

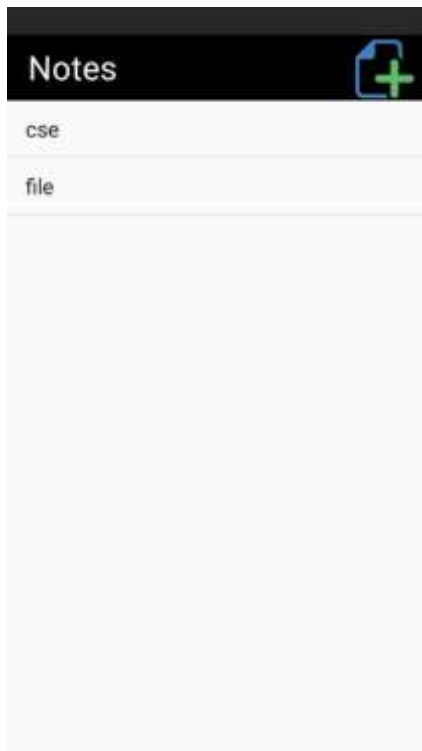


Fig 6.7 - Notes Activity

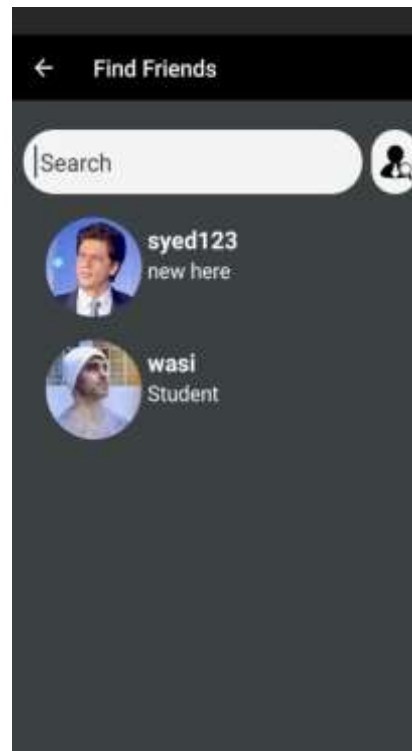


Fig 6.8 - Find Friends Activity



Fig 6.9 - Friend Requests Activity

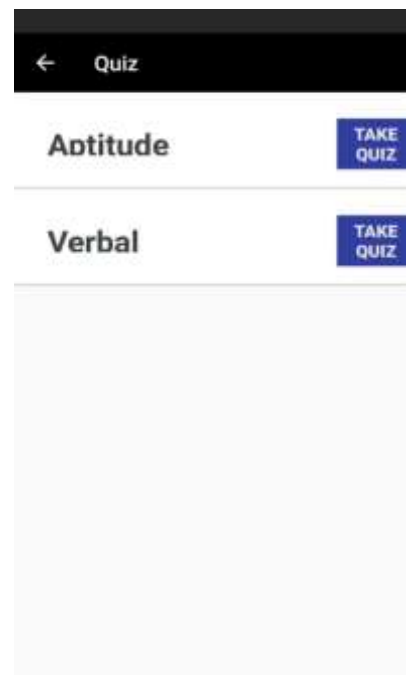
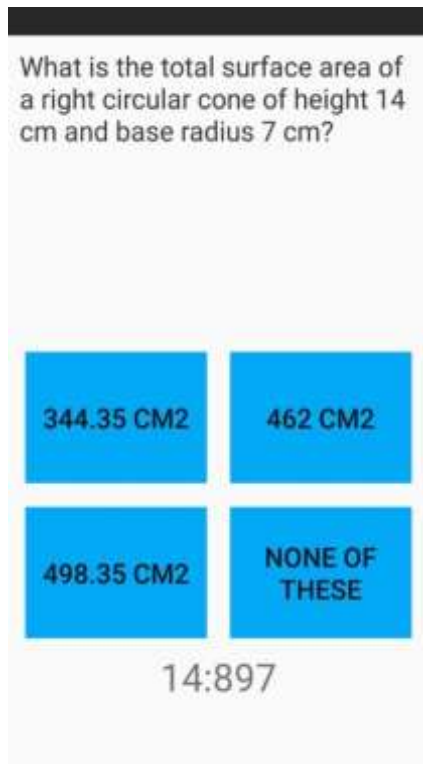
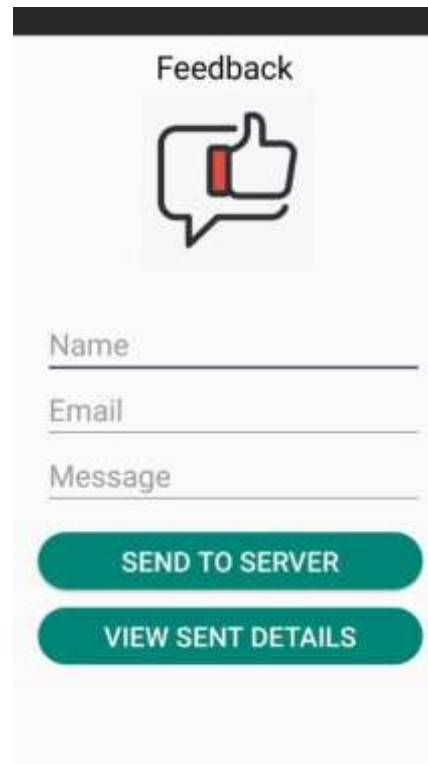


Fig 6.10 - Quiz Activity

**Fig 6.11 - Quiz Activity****Fig 6.12 - Feedback Activity**

7. CONCLUSION

In this project work, we reduce the physical efforts required to reach out to different Colleges and convey the information regarding events. We reduce the huddle by helping students download subject notes uploaded by teachers/students of other colleges. Making it easier for students of different colleges to share knowledge and communicate among one another. Help students to get their queries solved by the answers of other users (Students/Teachers) and also improve in their Verbal and Aptitude abilities by taking small built-in quiz. It helps to reach to every student and faculties in different colleges in very less time.

8. FUTURE SCOPE

The following are some of the future opportunities that would help sustain the portal :

- Cloud Vision API to detect and remove Inappropriate Post uploaded by Students.
- Online Registration and Payments of the Events apart from updates alone.
- Enable Teachers to Store Subject video lectures and help students from any colleges gain knowledge.
- Allow Two Students to compete against one another in Quiz

9. ACKNOWLEDGMENTS

We sincerely thank our management and institution K. S. Institute of Technology for providing us the resources and platform for showcasing our idea. Our deep sense of gratitude towards respected guide and Project coordinator Prof. Harshavardhan J.R for his valuable guidance, profound advice, persistent encouragement and help during the completion of this work. We extend our sincere thanks to our Prof and Head of Department Dr. Rekha B Venkatapur for providing all kinds of cooperation during the course. Finally we are thankful to the supporting staff of Computer Science Department and all those who directly or indirectly contributed to complete this work.

10. REFERENCES

- [1] Abdulkareem Ademola, Adeyinka Adewale, Dike U. Ike “Design and Development of a University Portal for the Management of Final Year Undergraduate Projects”, International Journal Of Engineering And Computer Science Volume 2 Issue 10 October, 2013 Page No. 2911-2920.
- [2] Pinay Patel “Student Portal for GDCST”, International Research Journal of Engineering and Technology (IRJET).
- [3] Published in: 2015 3rd International Conference on Control, Engineering & Information Technology (CEIT) “Student Portal architecture based on SOA”.