σ

CS 5551 – Advance Software Engineering Project

Application Name: "COURSE BUDDY "

PROJECT FINAL INCREMENT REPORT

TEAM NUMBER: 9

TEAM MEMBERS:

VAMSI KRISHNA CHALLA

SHANKAR PENTYALA

GIRISH KUMAR REDDY NAGELLA

1. INTRODUCTION:

COURSE BUDDY

Course buddy is an interactive application which acts as lifeline between students and the teaching fraternity. Student fraternity has a tendency to work along deadlines and many of the times, they tend to forget the deadlines and lose marks even having competency to complete on time. Not only deadlines, many of the student fraternity has a feeling of introvert of not being able to directly interact with the teaching fraternity.

Our COURSE BUDDY application tries to lessen the gap between teaching and student streams and also provide an interface for free flow of communication. Giving it a social media touch, would easily attract the students to make use of it.

Many times we face difficulty in finding out means to talk to our teaching faculty because of lack of availability of information. Myself, I am example how a student face difficulty in reaching faculty to break many barriers regarding the subject. As an international student, I was very much astonished at the etiquette of the college and I pulled myself backward from making a step ahead to clear my doubts regarding subjects. In the initial days, I was not able to find means of communication to reach out the faculty to be able to clarify my doubts regarding the subject and course curriculum.

So goes a saying, “Failure is the stepping stone of success.” As a blind follower of that, we tried to utilize this project opportunity to build an interface that would address the problems faced by us.

2. Project Goal and Objectives:

2.1. Overall goal:

Our primary goal is to develop a hybrid application “COURSE BUDDY” which is an interface between students and faculty. This interface is made using android software development kit and various other services which are needed for smooth running of applications.

2.2. Specific objectives:

Course buddy is used purely for educational purpose. Students login into the application to be able to talk with tutors, teaching assistants and also their course buddies without compromising their privacy

2.3. Specific Features:

a.) Not compromising student’s privacy:

Many doubts of students can actually be solved by talking with their friends. But in an international institution we may not be able to easily mingle with students. Our course buddy would create a discussion forum without disclosing student’s identity and it would facilitate the student to clarify their doubts easily.

b.) Talk with tutors and teaching assistants on a single platform:

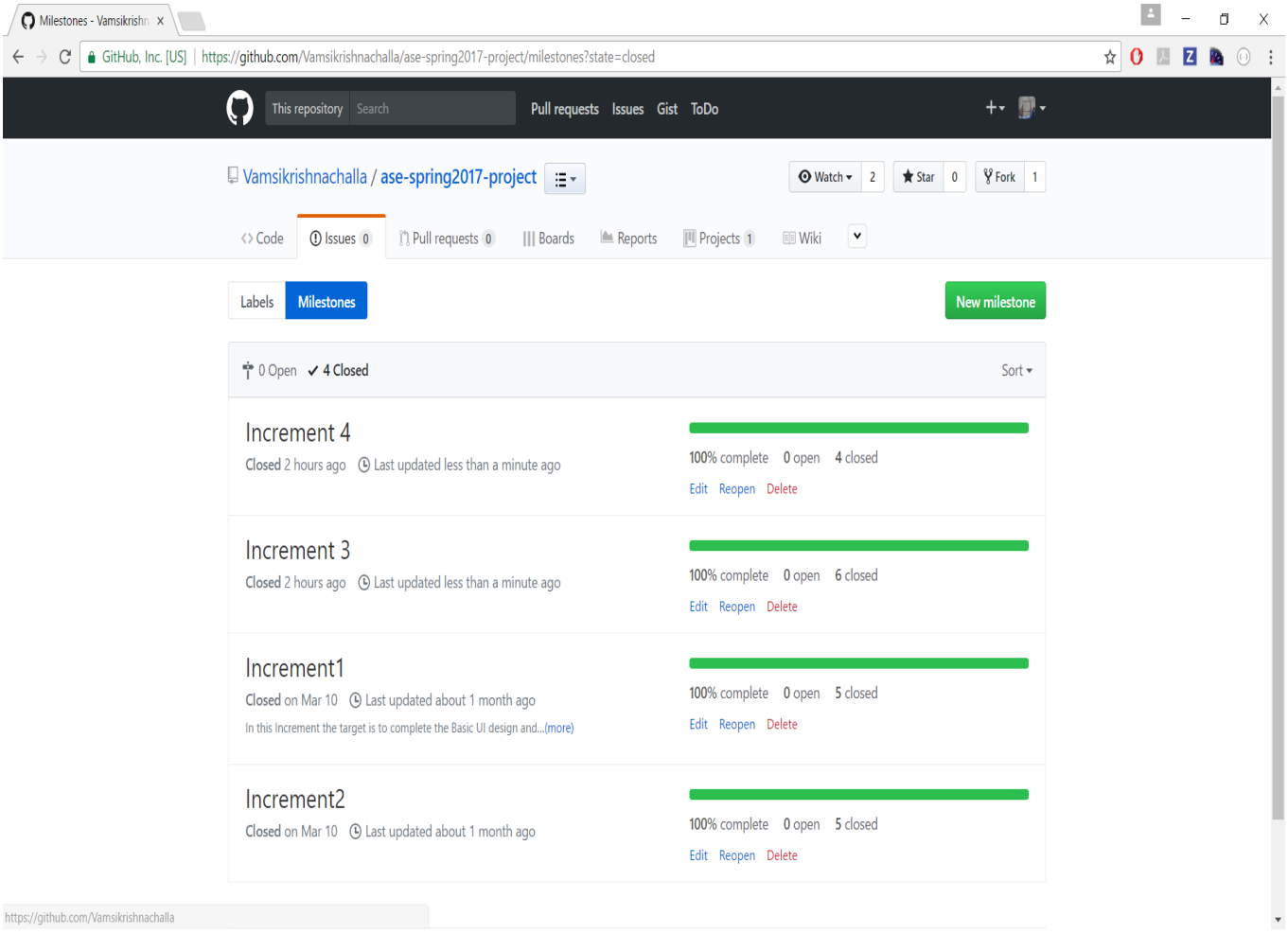
Instead of using mail based communication to reach out to professors and using mail based or spreadsheet based communication to reach out to teaching assistants, this interface provides a unique platform to reach out both at a single click.

2.4. Significance:

This application in creating a healthy environment where learning evolves from different aspiring minds and would help each other to excel themselves by taking help of their professors, teaching assistants and their friends.

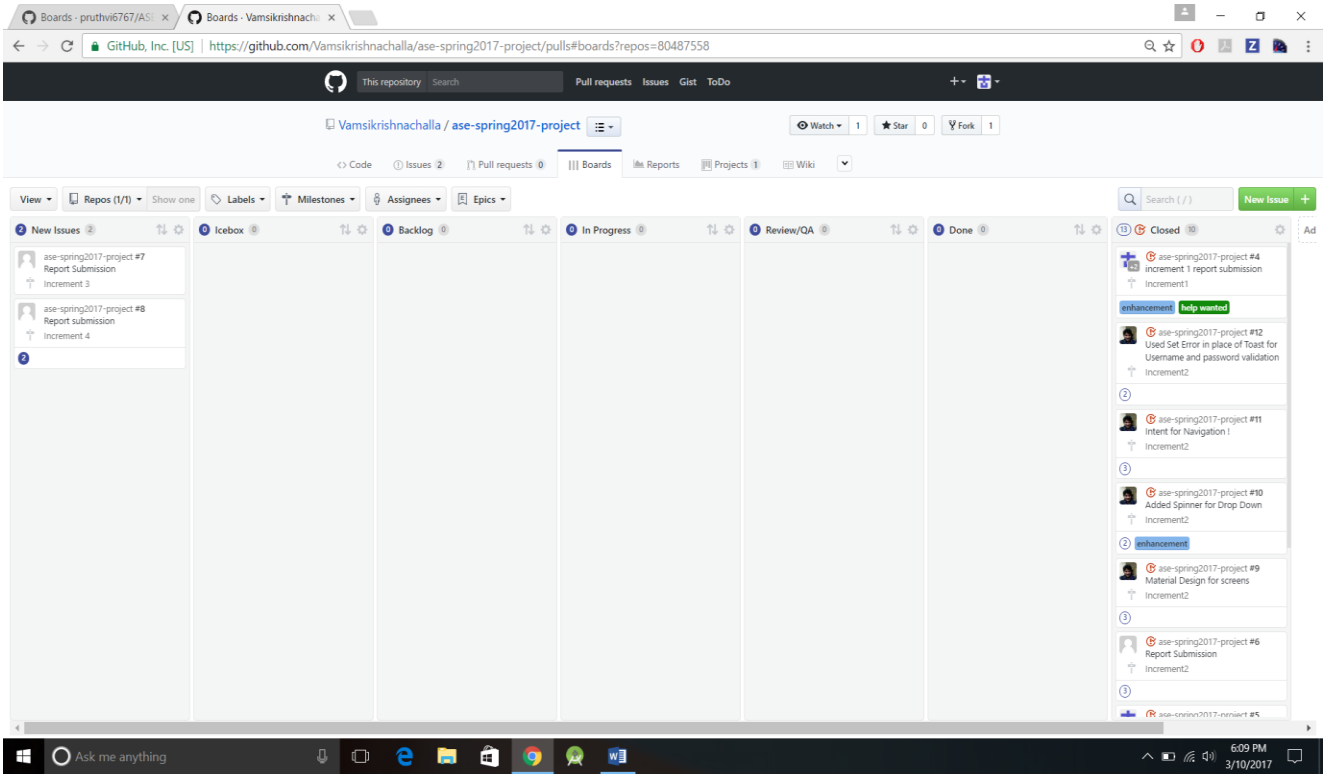
3. Project Lifecycle:

3.1. Schedule for different Increments:

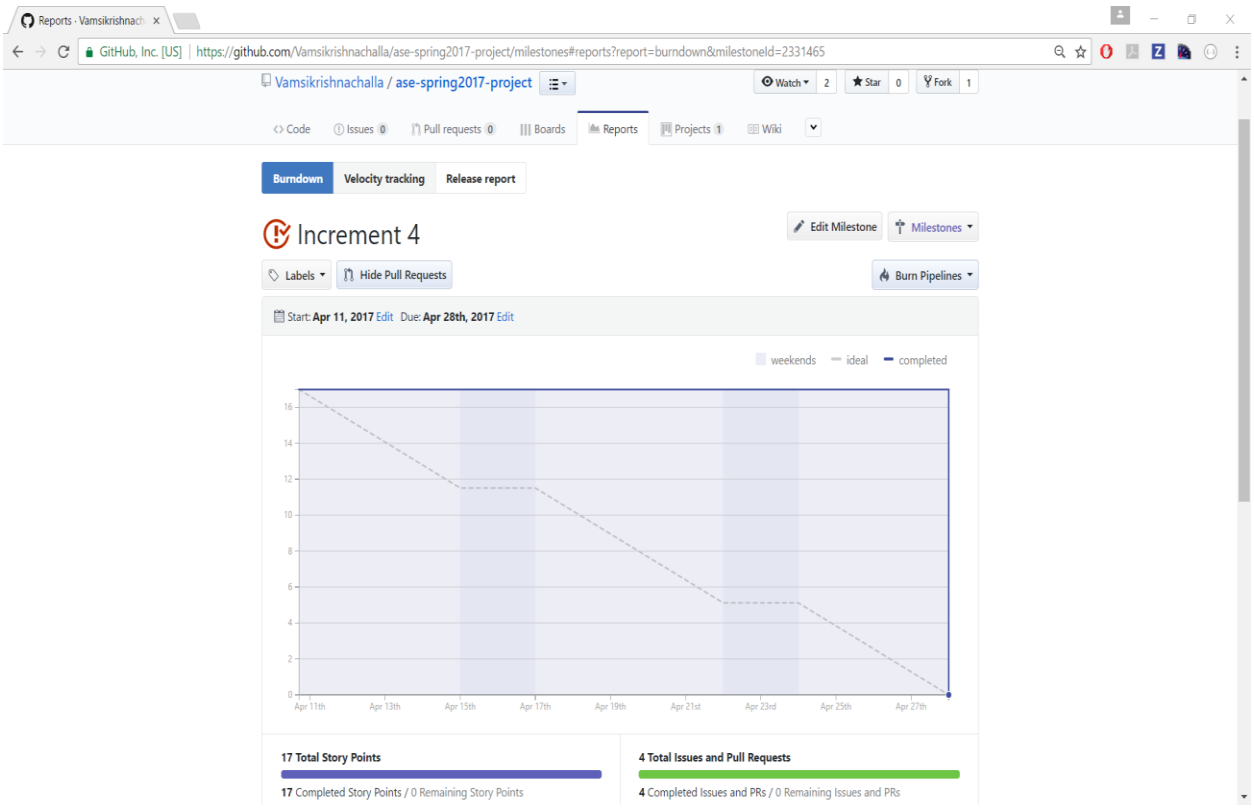


3.2. Project timelines, Members and responsibilities:

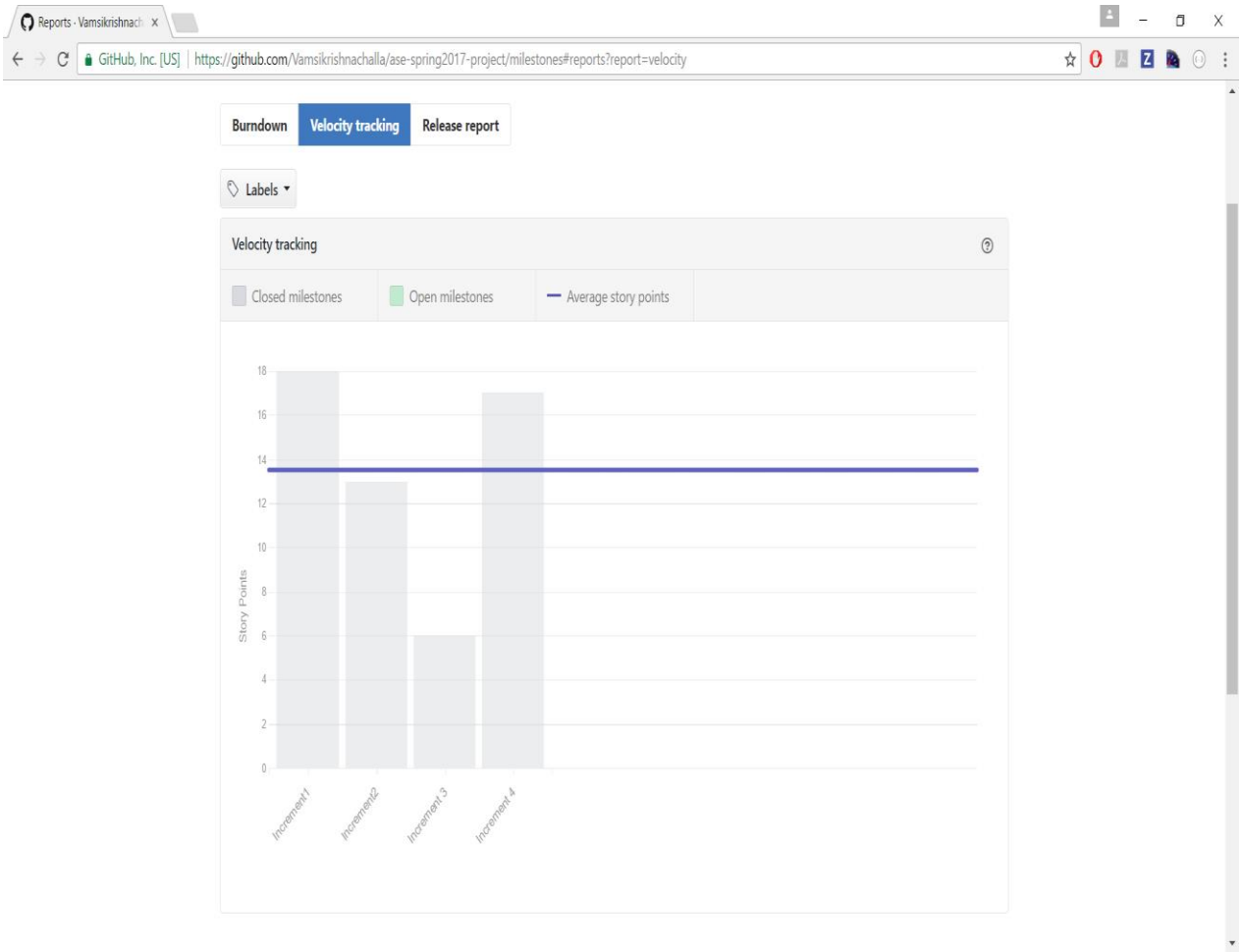
- Below are the issues that were created under each increment.
- These tasks are assigned to team members.
- Once the respective task is accomplished, the issue is moved to closed state.



3.3. Burndown chart:



3.4 Velocity Tracking:



4. Final Increment Report:

In this Final increment, we have designed app mockups by designing wireframes and by keeping them in mind, we have designed all our app pages login page, course selection page and choosing the option to talk with anyone of the professors or teaching assistants or course buddies etc.

4.1. Existing services / REST API:

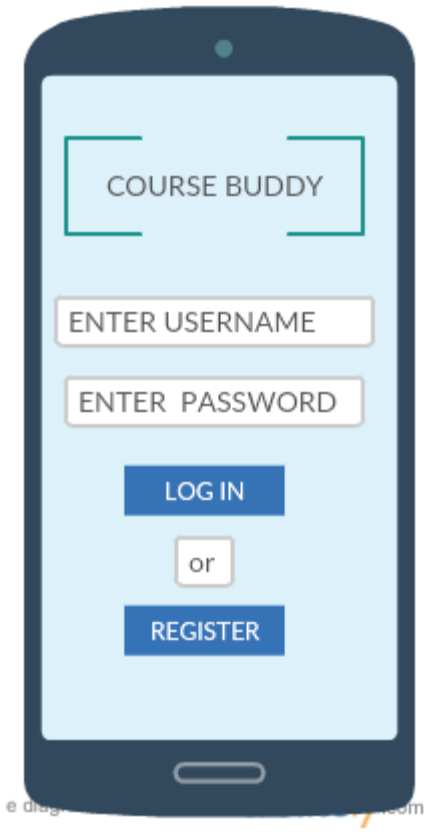
Throughout our project we used several services and frameworks including Firebase , Sinch , Volley and Picasso.

4.2. Detail design of features:

4.2.1 Wireframes:

The wireframes for our project are shown below:

Login page: after user opens the application, one would be asked to enter the credentials to enter into the application.



The user should have already be registered i.e., the details of the user must be already available with us as it is related to university and the details of student would be uploaded on a pre hand for authentication.

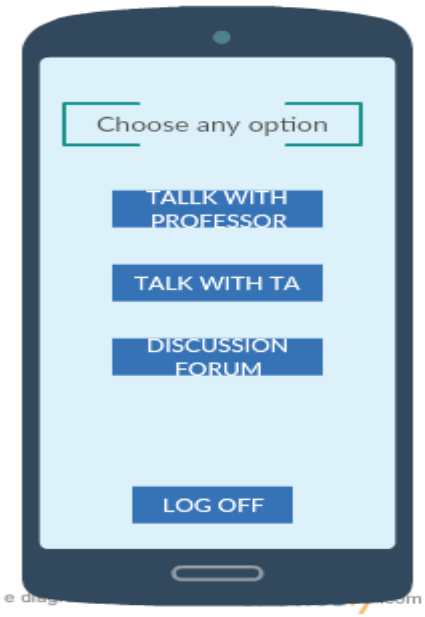
This wireframe comes once the login is successful.

In this wireframe we will be able to see the options to choose the courses In which we have enrolled.

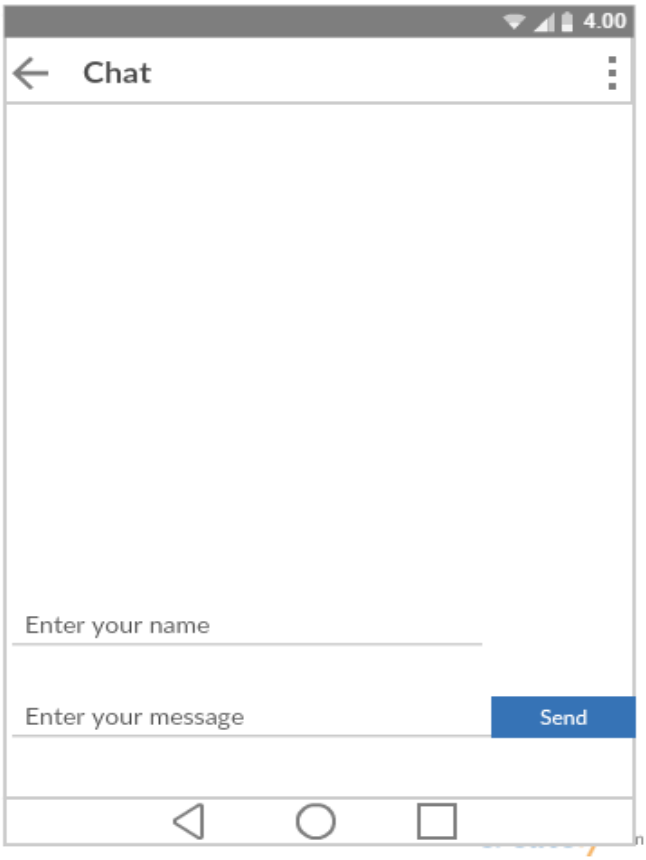
Upon selecting the subject, we would get the option to talk with one of the required persons which we are needed to talk to like:

1. Professor
2. Teaching assistants
3. Course buddies.

This is the wireframe for choosing the option to talk or else log off from the application.

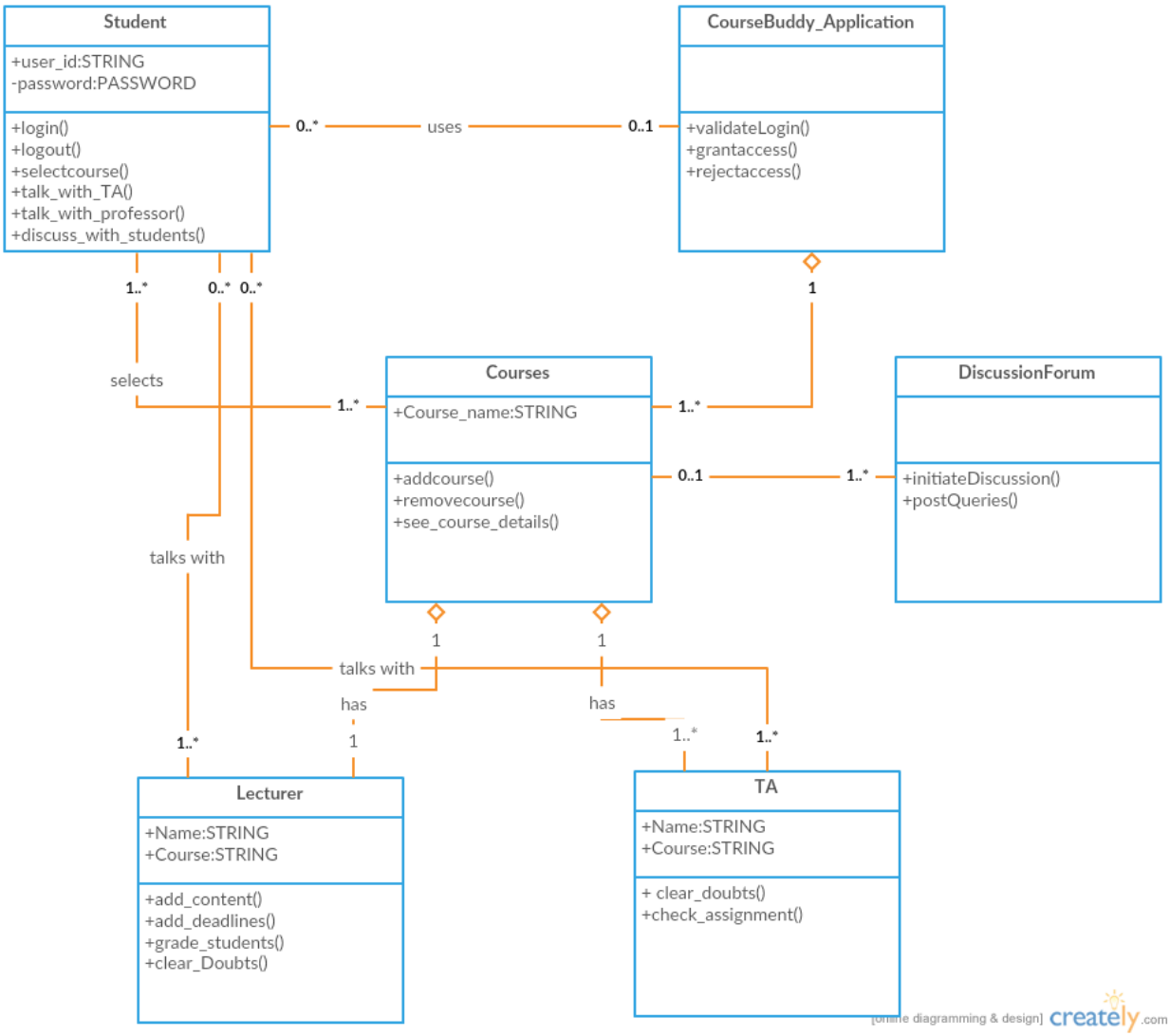


Upon Selecting an option, you will be redirected to chat screen,

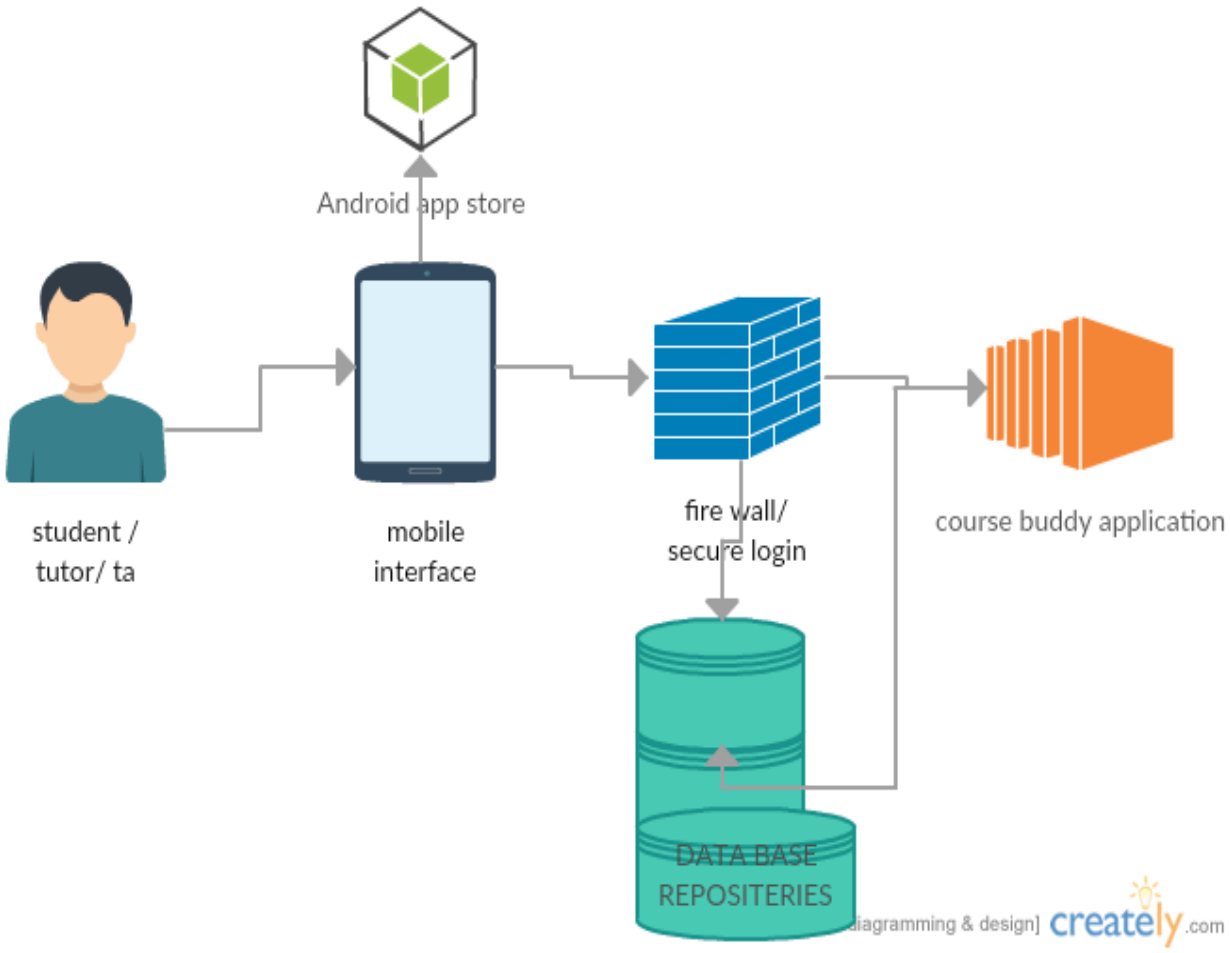


4.2.2 User Stories:

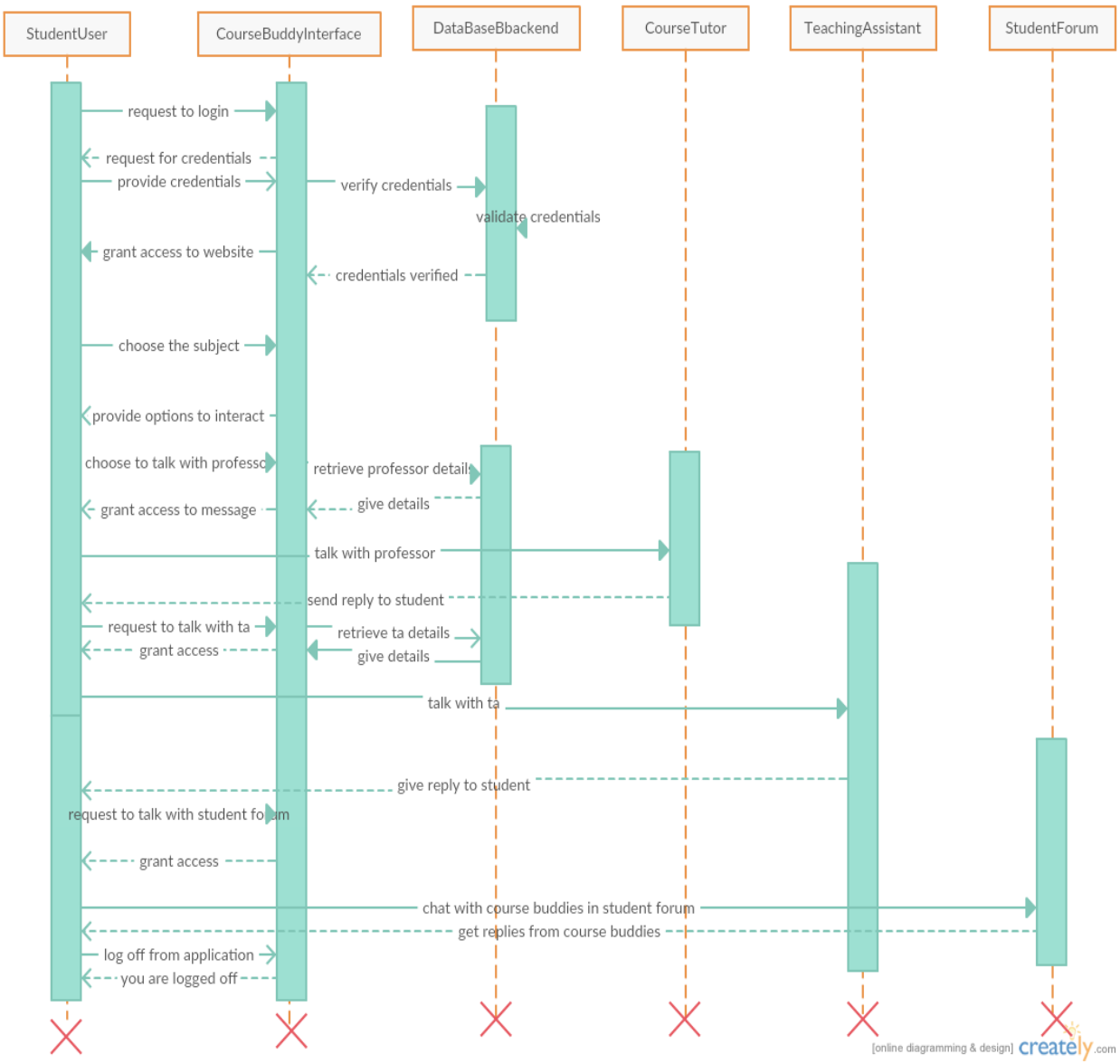
Class diagram:



Software architecture diagram:



Sequence diagram:


StudentUser
CourseBuddyInterface
DataBaseBbackend
CourseTutor
TeachingAssistant
StudentForum
request to login
request for credentials
provide credentials
verify credentials
validate credentials
grant access to website
credentials verified
choose the subject
provide options to interact
choose to talk with professo
retrieve professor details
grant access to message
give details
talk with professor
send reply to student
request to talk with ta
retrieve ta details
grant access
give details
talk with ta
give reply to student
request to talk with student fo um
grant access
chat with course buddies in student forum
get replies from course buddies
log off from application
you are logged off
[online diagramming & design] creately.com

Design Patterns Used:

1) Observer Pattern: In our code,

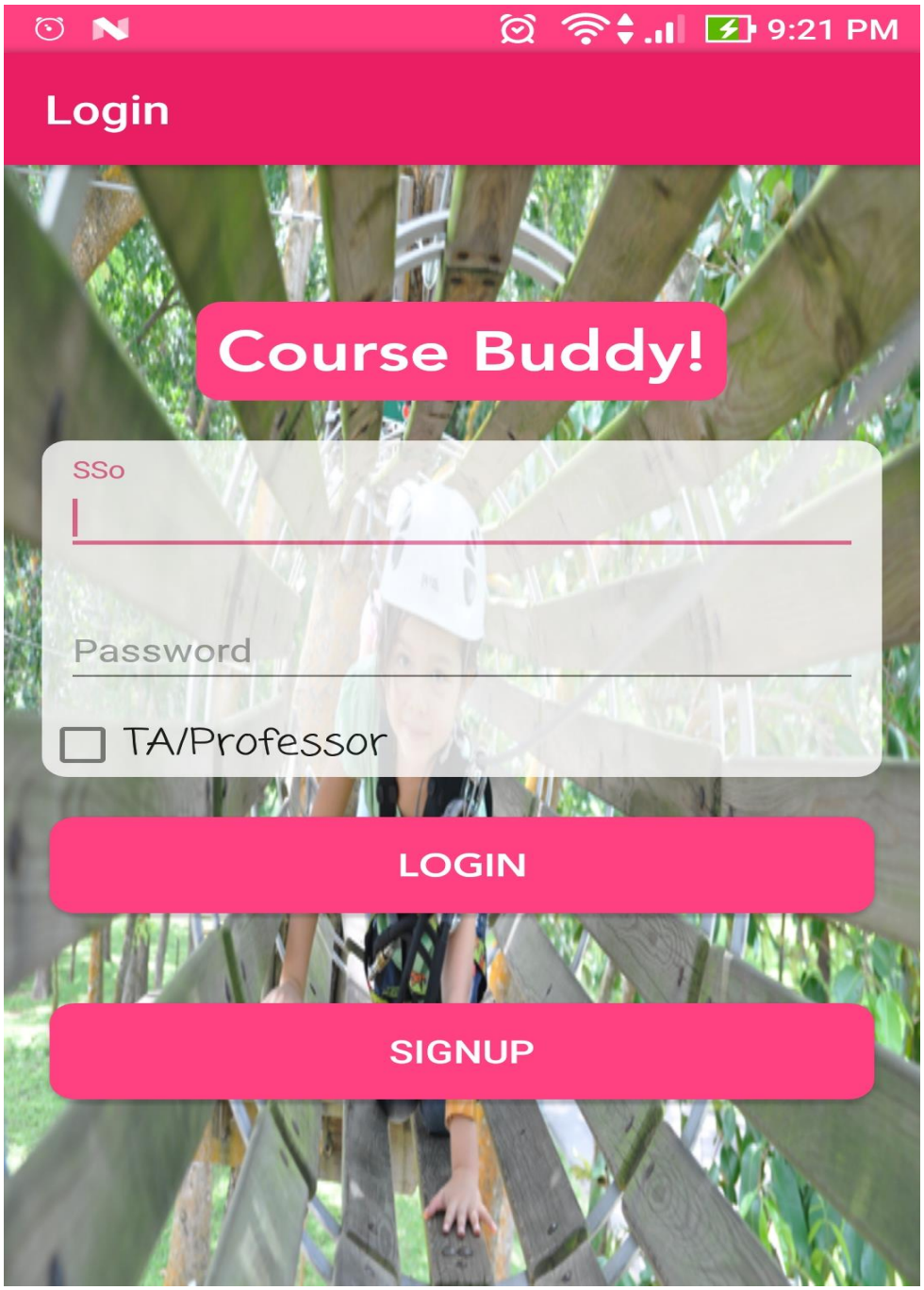
```
author = FirebaseAuth.getInstance();
listener = new FirebaseAuth.AuthStateListener() {
    @Override
    public void onAuthStateChanged(@NonNull FirebaseAuth firebaseAuth) {
        s[0] = firebaseAuth.toString();
       Toast.makeText(MainActivity.this,"State changed",Toast.LENGTH_LONG).show();
    }
};
author.addAuthStateListener(listener);
```

2) State Pattern : In our code,

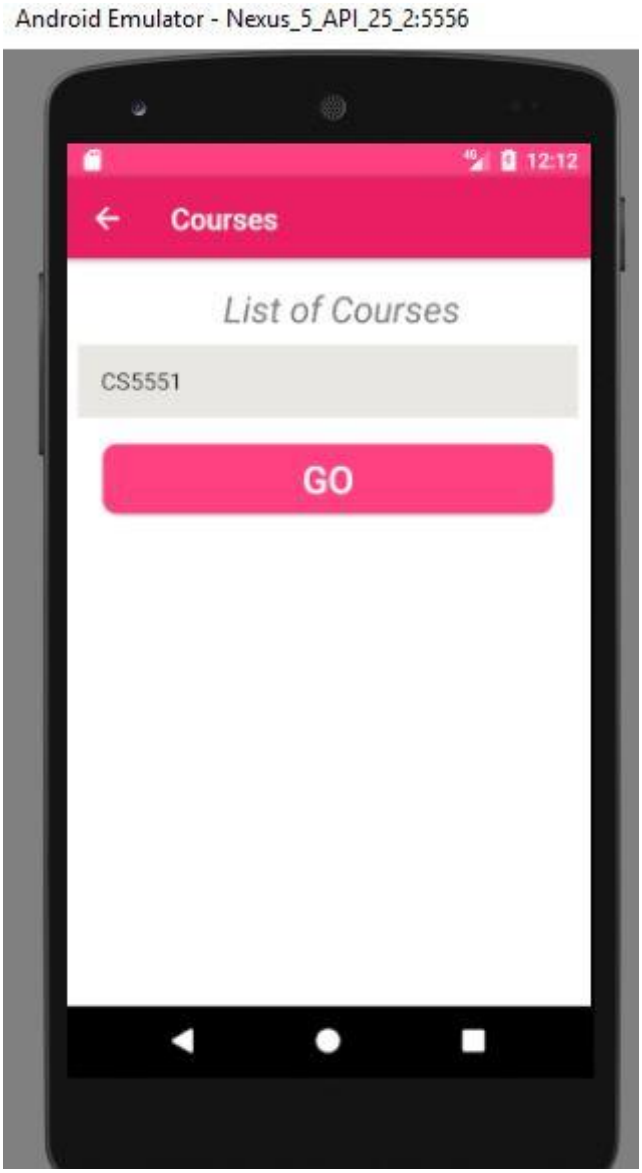
```
Login.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        sso = UserName.getText().toString();
        emailid = sso+"@mail.umkc.edu";
        Pwd = Password.getText().toString();
        if (emailid.isEmpty()) {
            UserName.setError("Enter Email id");
            //Toast.makeText(getApplicationContext(), "Enter User Name", Toast.LENGTH_SHORT).show();
        } else if (Pwd.isEmpty()) {
            Password.setError("Enter Password");
            //Toast.makeText(getApplicationContext(),"Enter Password",Toast.LENGTH_SHORT).show();
        }
```

4.3. Implementation:

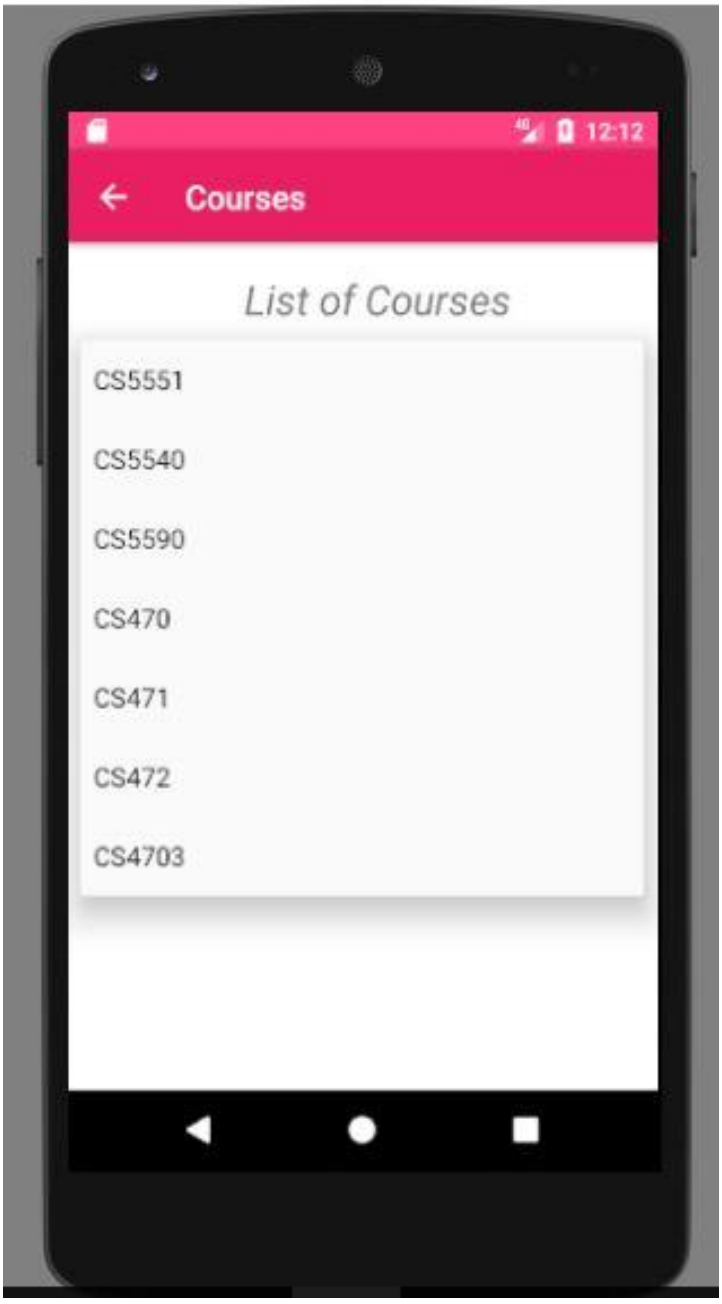
Home page for application:



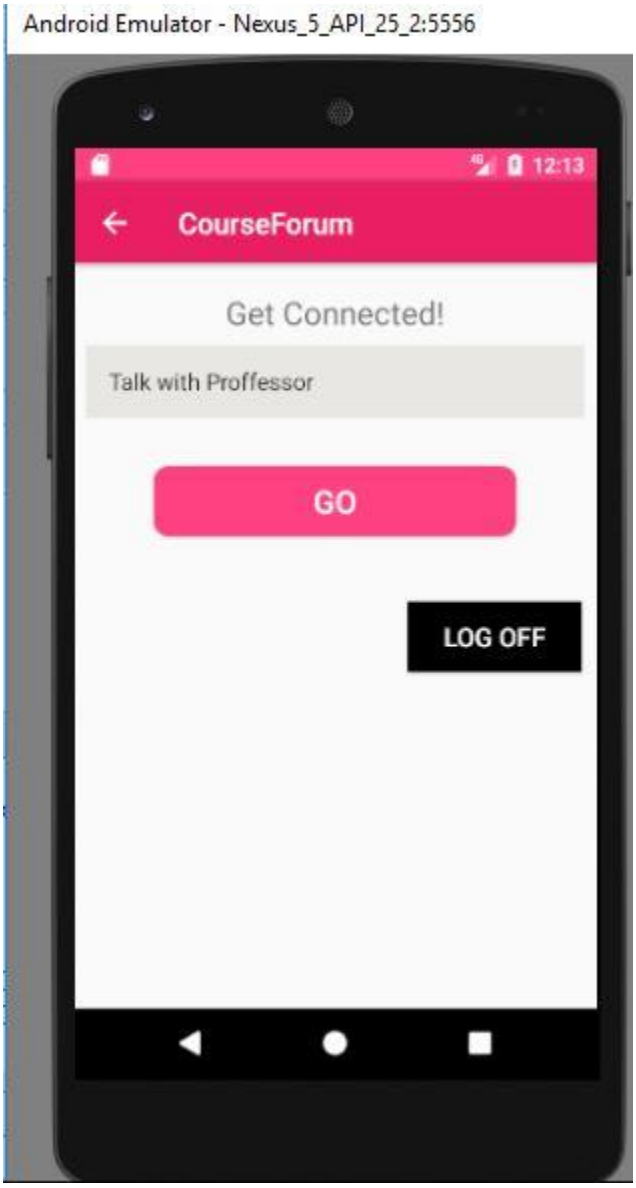
On successful login, we would get the option to choose the course:



Android Emulator - Nexus_5_API_25_2:5556

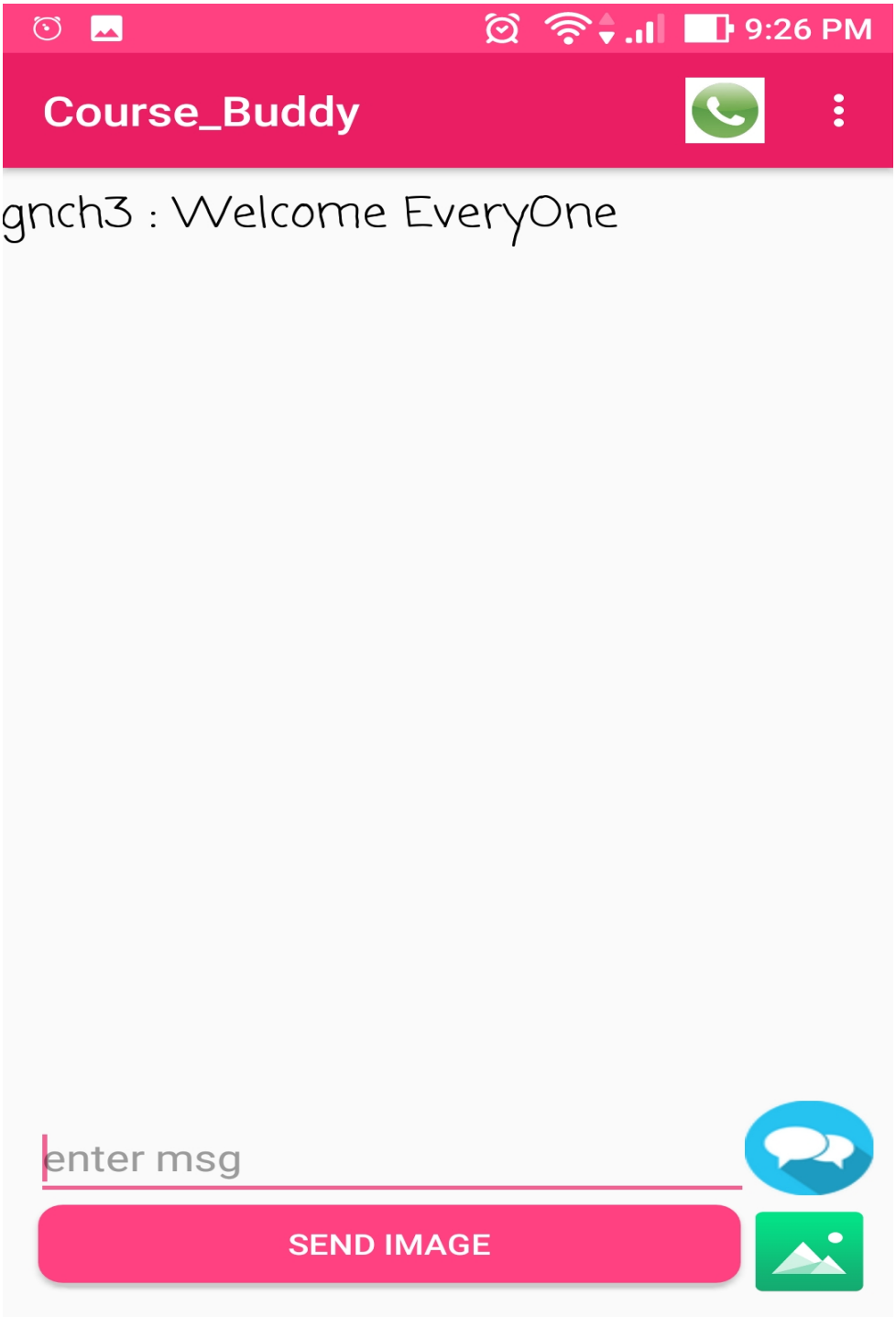


On selecting the course, we would get an option to talk with anyone:



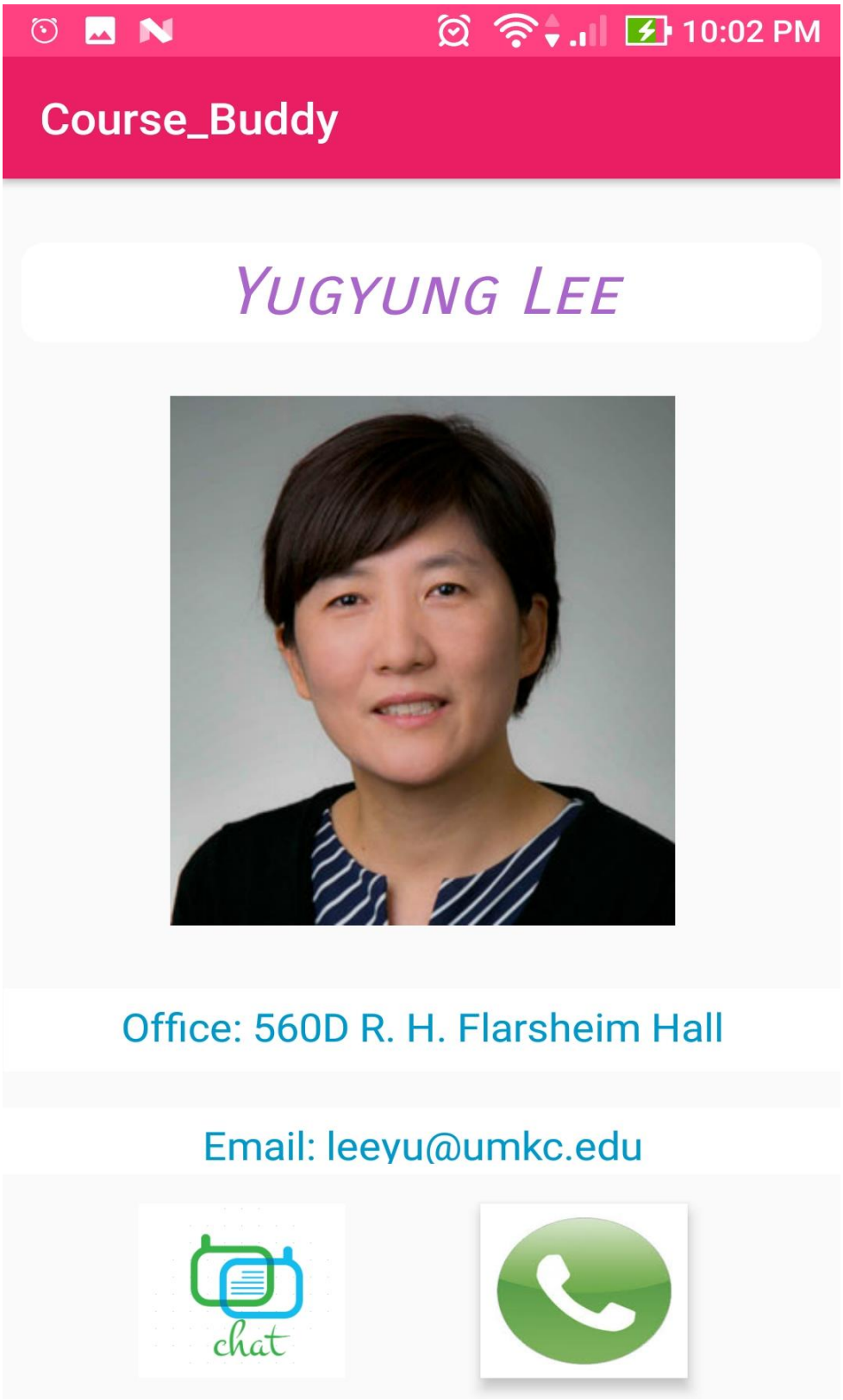
Android Emulator - Nexus_5_API_25_2:5556

On selecting the option Discussion Forum, A chat Screen opens.

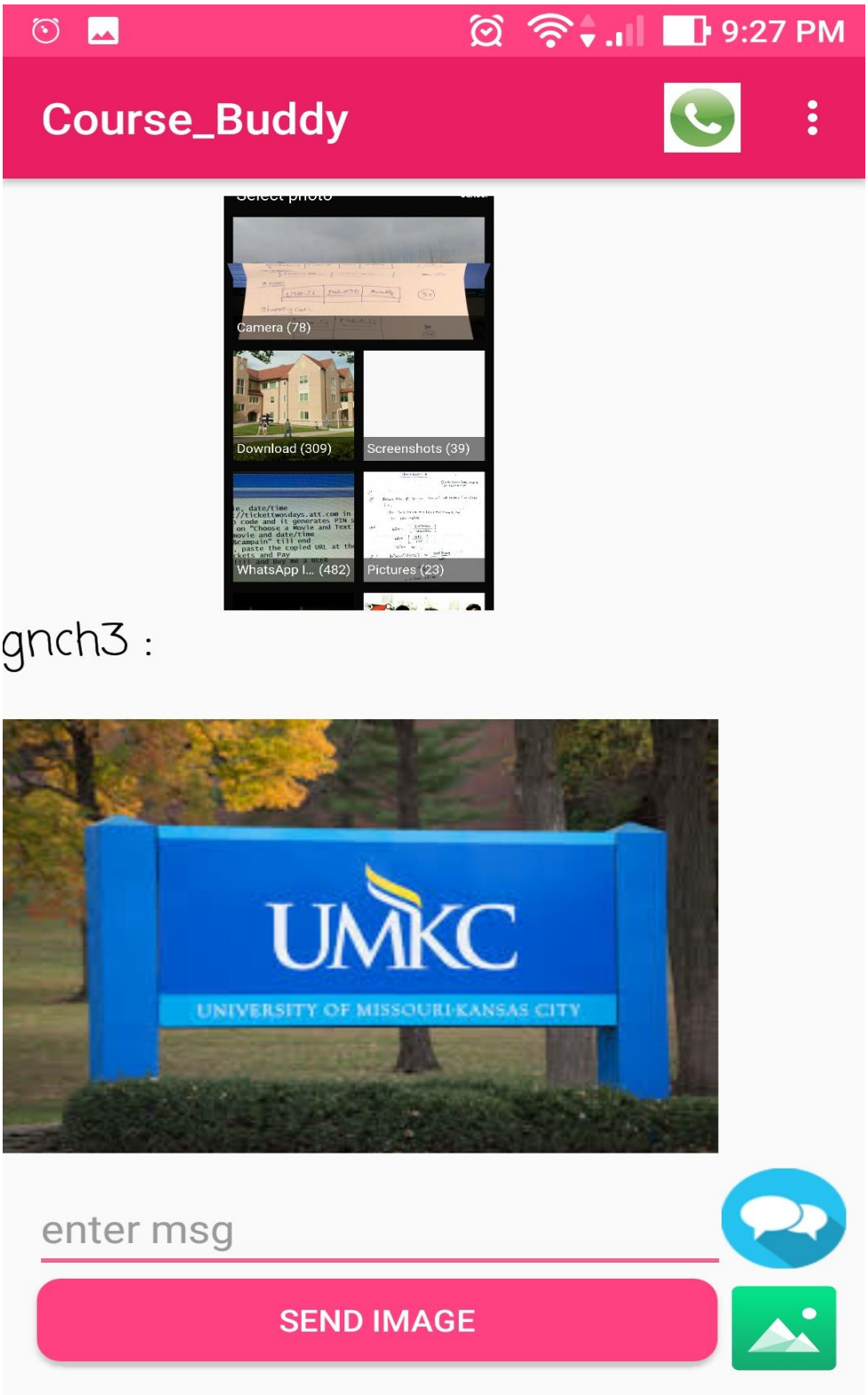


On selecting the option Talk with TA, another screen opens

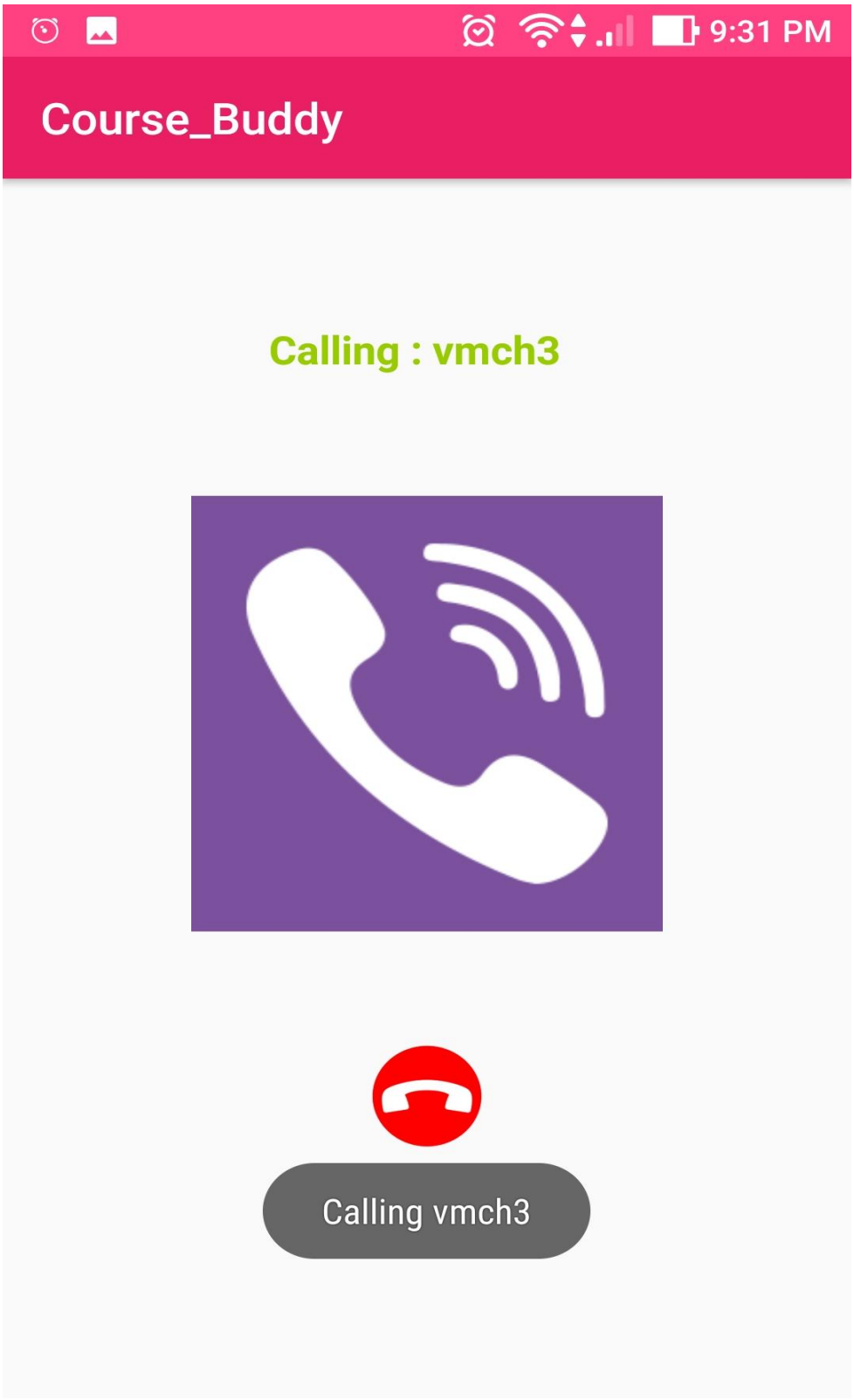
On selecting the option Talk with professor, professor screen opens



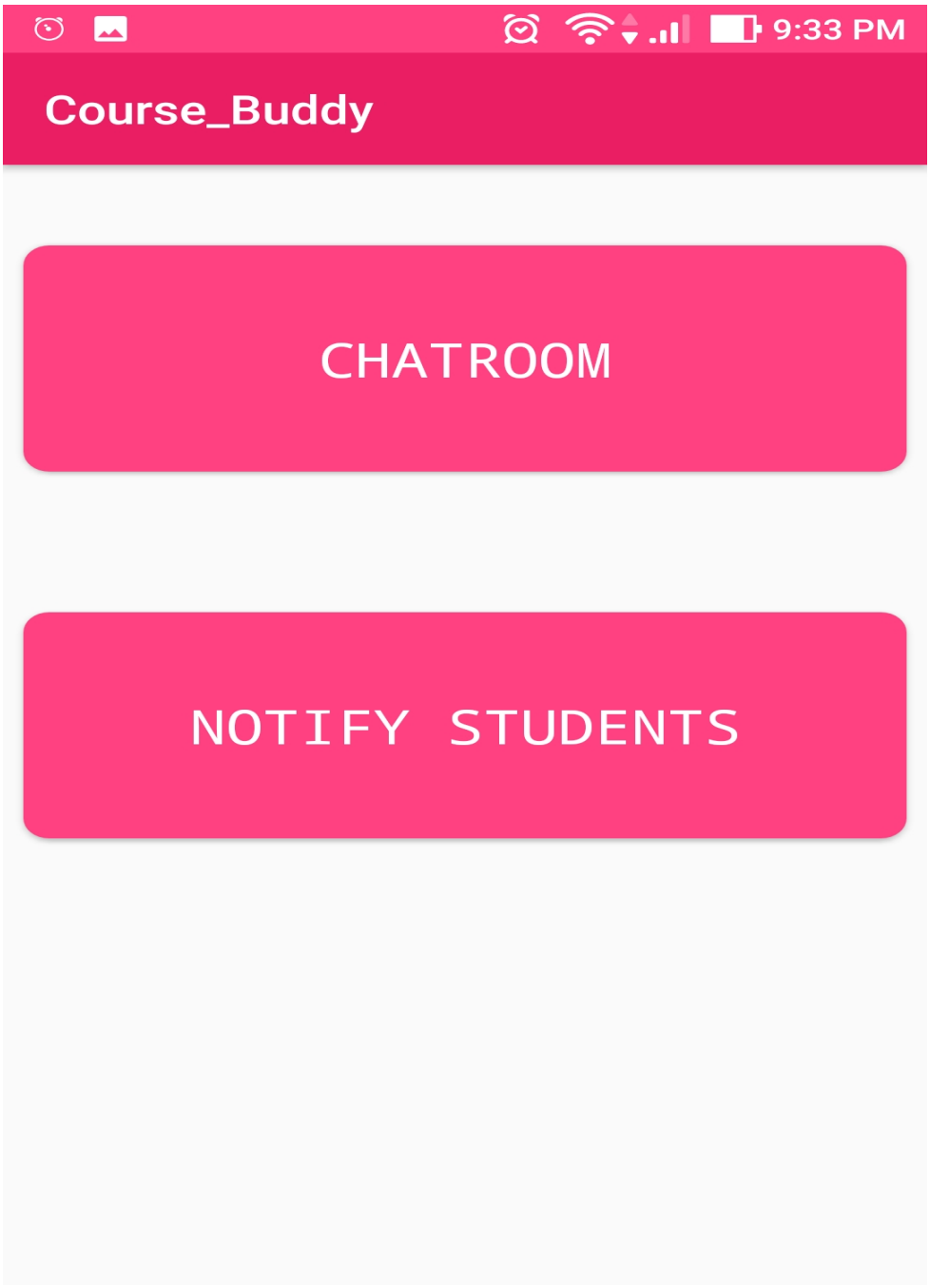
Also, we can perform App-App Image Sharing:



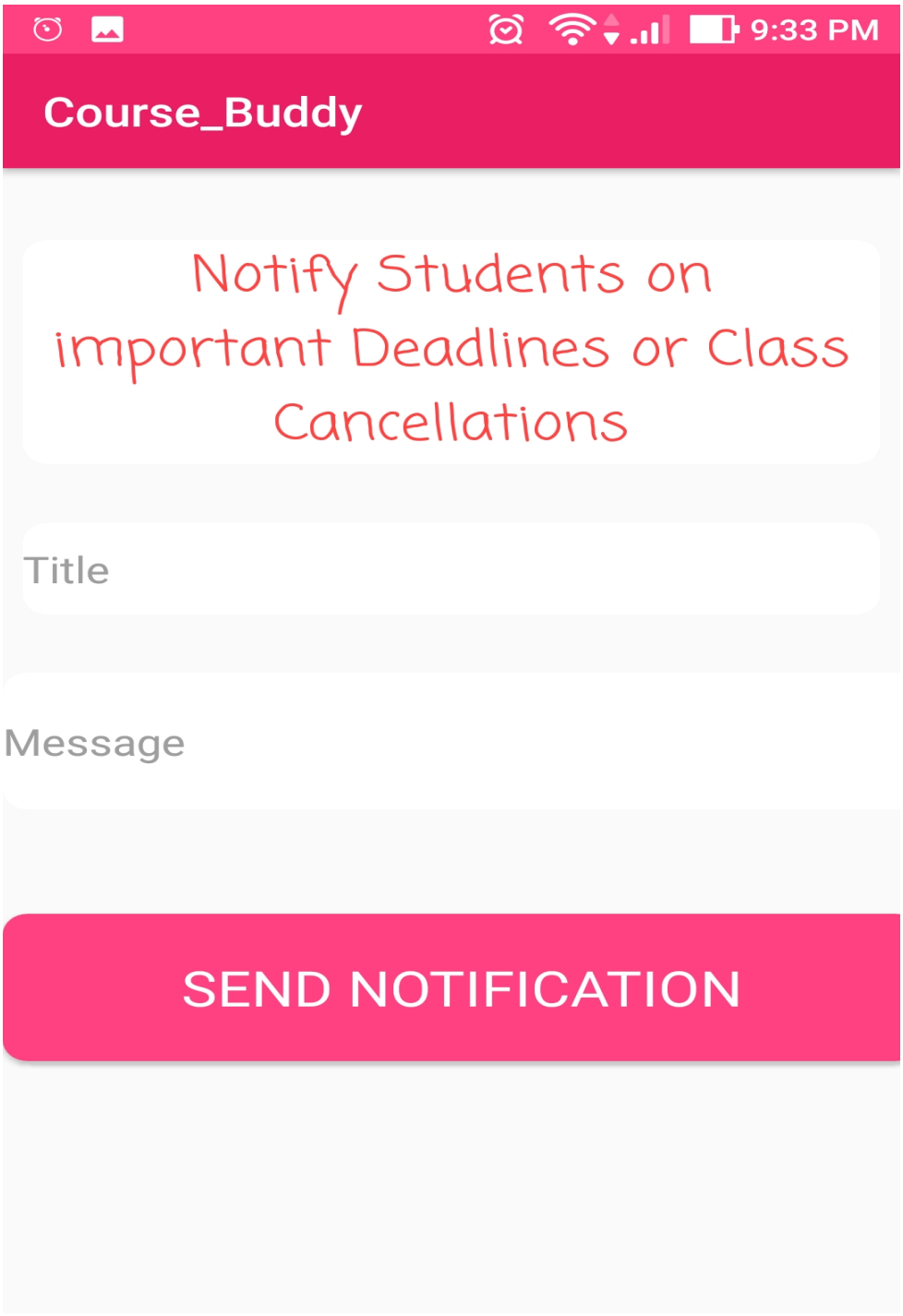
Finally, App-App calling is possible,



If we login as Tutor/TA:



Tutor/TA can send Important updates to students,



4.4. Unit Testing:

The below are the unit cases for final phase of the project:

s.no	Test case title	Description	Expected outcome	Result
1	User login verification	The user should provide valid email id and password to login.	Upon successful login, user needs to be moved to home page	Pass
2	User login validation with invalid credentials	User will enter wrong credentials	As the user have entered wrong credentials, access should be declined.	Pass
3	User login verification without details	User will not enter any credentials.	User should not be granted access as he had not entered any credentials.	Pass
4.	Correct navigation to choosen page	User will give login credentials	Upon successful login, the page should navigate to course selection page	Pass
5	Navigation after choosing the course	User will select the course he is enrolled in	Upon choosing the course, user needs to be navigated to talk with tutor, ta and discussion forum page	Pass
6	Navigation to Chat Screen page	User will select the option to engage in discussion forum.	Upon Choosing the Option, User needs to be navigated to Chat Screen page.	Pass
7	Navigation to Call Screen page	User will choose to call by clicking the call icon on top.	Upon Choosing to call, User needs to be navigated to Call Screen page	Pass
8	Navigation to Talk with Professor and Talk with TA pages.	User will select the option to either talk with tutor or TA.	Upon Choosing the Option, User needs to be navigated to respective pages.	Pass

4.5. Deployment:

- We have deployed our application on an emulator as well as on Device and have taken screenshots.
- We have explained about implementation in detailed in the above sections.
- GitHub link for source code and documentation:

4.6. Project Management:

Technologies used: Android SDK, JAVA SDK.

Software used: Android studio.

Services used: Firebase and Sinch.

By this increment we have completed all the work:

1. UML DIAGRAMS
2. Created increments in Zen Hub
3. Created wireframes using creately.
4. Login form modified with material design
5. Used spinners for drop down of courses and chat options.
6. Added linear layout with the material design of the screens login, courses, connect activities.
7. Added navigation.
8. Courses and Options page.
9. Implemented App-App Chatting.
10. Implemented App-App Image Sharing.
11. Implemented App-App Calling.
12. Implemented Secure authentication.
13. Implemented Notifications for important updates.
14. Different screens for different users.

Contributions:

Vamsi Krishna Challa: 33.33%

Shankar Pentyala: 33.3%

Girish Kumar Reddy Nagella: 33.3%