DEPARTMENT OF COMPUTER ENGINEERING
GOVERNMENT POLYTECHNIC, AWASARI (KH)
SEMESTER VI [2018-2019]

A
PROJECT REPORT
ON

"AwareBot to enhance students' productivity"

'(Awarer + Robot)'

Submitted by:

Sagar Ravindra Labade	271213
Naikade Rohit Anand	271210
Inamdar Yasar Atharabbas	18CO311

UNDER THE GUIDANCE OF

Mrs. Dandale J. P.

DEPARTMENT OF COMPUTER ENGINEERING
GOVERNMENT POLYTECHNIC, AWASARI (KH)
SEMESTER VI [2018-2019]

CERTIFICATE

This is to certify that project work entitled

Android app and website "AwareBot" for College students to enhance productivity

Submitted by:

1. Sagar Ravindra Labade 271213
2. Naikade Rohit Anand 271210
3. Inamdar Yasar Atharabbas 18CO337

is a bonafide work carried out under the supervision of Mrs. Dandale J. P. and it is submitted towards the partial fulfillment of the requirement of MSBTE, for the award of the Diploma in Engineering (Computer Engineering).

(Mrs. Dandale J. P.)
Guide (Internal Examiner)

(Mrs. Sant S.S.)
Head of Computer Department

Prof. Kerkal S. R.
(Principal)

Place: Awasari (Kh), Pune **Date:**

DEPARTMENT OF COMPUTER ENGINEERING
GOVERNMENT POLYTECHNIC, AWASARI (KH)
SEMESTER VI [2018-2019]

CERTIFICATE

This is to certify that project work entitled

Android app and website "AwareBot" for College students to enhance productivity

Submitted by:

Sagar Ravindra Labade	271213
Naikade Rohit Anand	271210
Inamdar Yasar Atharabbas	18CO337

Have completed successfully project for the academic year 2018-19.

……..	…….....
(Mrs. Dandale J. P.)	……..
(Internal Examiner)	**(External Examiner)**

ABSTRACT

Project Topic: 'AwareBot' to enhance students' productivity

We present a system that enhances productivity of college students. In this system, website and android application is used. Our system is used to give updates to students directly from teachers in their hands anytime, anywhere. Here, when teacher's login takes place successfully he/she is directed to activity to enter some data about lectures, notices and so on. When students get logged into the system they can see the posts posted by their corresponding teachers. By this mechanism both teachers and students save lot of time and efforts.

For logging in, we provide permissions to teachers to assign different passwords for different classes. If students do not have android, then they can use our website to see posts. This website is developed for apple iPhone or desktop users.

To implement this, we have used realtime database which allows to store and retrieve data in milliseconds. This makes our system more responsive and reliable.

ACKNOWLEDGMENT

I have great pleasure and sense of satisfaction in presenting this project report on "AwareBot to enhance students' productivity".

As part of the curriculum of diploma in Computer Engineering. I am very fortunate to be guided by people with vast and resourceful experience in their respective field of work

I express my sincere gratitude to my guide **Mrs. Dandale J. P.** (Lecturer in computer Engineering department) for her timely guidance, support and suggestions. I am also thankful for her sincere help and for making me available all the faculty of the department. Without their efforts and constants monitoring the project and documentation would not have been duly completed. Also, I express my sincere thanks to **Mrs. Sant S. S.** (HOD of computer department), besides, I take this opportunity to express my sincere gratitude to the principal **Dr. Kerkal S. R.**, G.P. Awasari for providing a good environment and facilities to complete this project. I would also like to thanks all my colleagues who have directly or indirectly guided and helped me in the preparation of this project.

……..……
Sagar Ravindra Labade (271213)

……
Rohit Anand Naikade (271210)

……..
Yasar Inamdar Atharabbas (18CO337)

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1: INTRODUCTION

1.1 Introduction of 'AwareBot'

AwareBot is website and android application. It is used to give updates to students directly from teachers in their hands anytime, anywhere. Here, when teacher's login takes place successfully he/she is directed to activity to enter some data about lectures, notices and so on. When students get logged into the system they can see the posts posted by their corresponding teachers. By this mechanism both teachers and students save lot of time and efforts.

For logging in, we provide permissions to teachers to assign different passwords for different classes. If students do not have android, then they can use our website to see posts. This website is developed for apple iPhone or desktop users.

To implement this, we have used realtime database which allows to store and retrieve data in milliseconds. This makes our system more responsive and reliable.

In previous systems, we used to create timetable of college lectures. Students have to see it every time to know time of lectures. Also, some lectures of them were not conducted every day. It used to waste lot of time of students. Also, in previous systems, only notice boards were used to share important notices with students. Students miss many notices of them and that becomes problematic for teachers or college staff.

In our application, this problem is overcome. Students are able to see their upcoming lectures and practicals every day. Problem of off-lectures will not occur. It will save students' time. And teachers are also allowed to share important notices anytime. So, notices will deliver to each student.

Sometimes, WhatsApp groups were used to inform students by teachers about lectures, notices and checking/ submission alerts. Students did lot of discussions about that topic. It used to create miscommunication.

In our system, only teachers are granted to update information and alerts. Students don't have permission to reply on anything. It makes system more reliable.

Advantages:

1) Saves time
2) Saves efforts of both teachers and students
3) Increase productivity
4) Increase awareness among students
5) Learn before attending lecture

1.2 Problem Statement

1) Students waste time on attending to off lectures.
2) Students don't know 'what's going on in their college'.
3) They don't attend every lecture.
4) They don't know syllabus of their subjects.
5) They don't do regular study.
6) They don't take benefits of e-learning and smart study.

1.3 Project Objectives

1) Save time.
2) Save efforts.
3) Increase productivity
4) Smart study
5) Learn before attend
6) Build interest in learning

Chapter 2: FEASIBILITY STUDY

2.1 TECHNICAL FEASIBILITY STUDY

The technical feasibility assessment is focused on gaining an understanding of the present technical resources of the organization and their applicability to the expected needs of the proposed system. It is an evaluation of the hardware and software and how it meets the need of the proposed system.

Feasibility study of 'Awarebot':

1) Familiarity with Application:
   We have decided to use easy components in our application and website to interact with users. It is more familiar to users which generates less risks.
2) Familiarity with Technology:
   We would implement android apps and website to satisfy requirements. So, our team members completed courses and became familiar with existing technology which would also generate less risks.
3) Project size:
   We have decided to keep project size as less as possible. So, it generates low risk to project.

2.2 TIME FEASIBILITY

A time feasibility study will take into account the period in which the project is going to take up to its completion. A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Time feasibility is a measure of how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some projects are initiated with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable.

Time Feasibility of 'AwareBot':

We estimated time our system will take to develop. We got results of 3-4 months. We found it reasonable. Also we found our project's deadlines reasonable as project has to be delivered up to March 2019. We determined whether deadlines are mandatory or not.

CHAPTER 3: REQUIREMENT ANALYSIS

3.1 FUNCTIONAL REQUIREMENTS:

3.1.1. Observation

"AwareBot" is an application which allows both students and teachers to be aware about everything about their need. It allows teachers to share urgent notifications to their students. Data shared at each minute by teachers could become very helpful to students as it contains information which make students productive and very smart.

3.2 SOFTWARE REQUIREMENT

a) **Technologies** : Java, XML

b) **Database** : Firebase Realtime Database

c) **Operating System** : Microsoft® Windows® 7/8/10 (32 or 64 bit)

d) **Application Software** : Android studio

e) **Tools** : SDK

f) **Drivers** : ADB, SDK

3.3 SOFTWARE REQUIREMENT FOR WEBSITE

a) **Technologies** : HTML, CSS, JAVASCRIPT

b) **Database** : Firebase Real-time Database

c) **Operating System** : Microsoft® Windows® 7/8/10 (32 or 64 bit)

d) **Application Software** : Visual Studio, Google chrome

3.4 HARDWARE REQUIREMENT

a) **RAM** : 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator

b) **Hard disk** : 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)

c) **Resolution** : 1280 x 800 minimum screen resolution

CHAPTER 4: PROJECT PLANNING

4.1 EFFORTS

We listed down following efforts to successfully complete our project:

1. Finding Project Topic
2. Finding Existing System and disadvantages
3. Finding solution
4. Collecting information
5. Implementing Proposed System
6. Developing GUI
7. Implementing database
8. Testing of Project

4.2 TIME ESTIMATION

Tasks	**Period**
Requirement analysis	September 3rd week-October 2nd week
Project Planning	October 3rd week to November 2nd week
Design	November 3rd week to November 4th week
Development	December 1st week to February 4th week

Integration and testing	March 1st week to March 2nd week
Execution and Deployment	March 3rd week

Table 4.1 Time Estimation

4.3 COST ESTIMATION OF THE PROJECT

To complete this project, before starting the planning phase, we studied about our project requirements. Then, we planned about what we would need to develop app and website. For completing it, we thought that we would not need any other resources to spend money other than our laptops and programming knowledge.

Sr. No.	**Description**	**Cost**
1	2 laptops*	0 ₹
2	2 chargers*	0 ₹
3	Internet charge	149 ₹ × 7 months
4	1 Android Mobile*	0 ₹
	Total:	1050 ₹

* - Recourses we already have
₹ - Indian Currency (Rupee)

Table 4.2 Cost Estimation

So, our estimated cost was up to 1050 rupees.

4.4 WORK STRUCTURE:

We estimated time to complete project. So, for avoid late delivery of our project each phase had to be completed in corresponding period.

As per time estimation, we developed bar graph to understand our work structure easily.

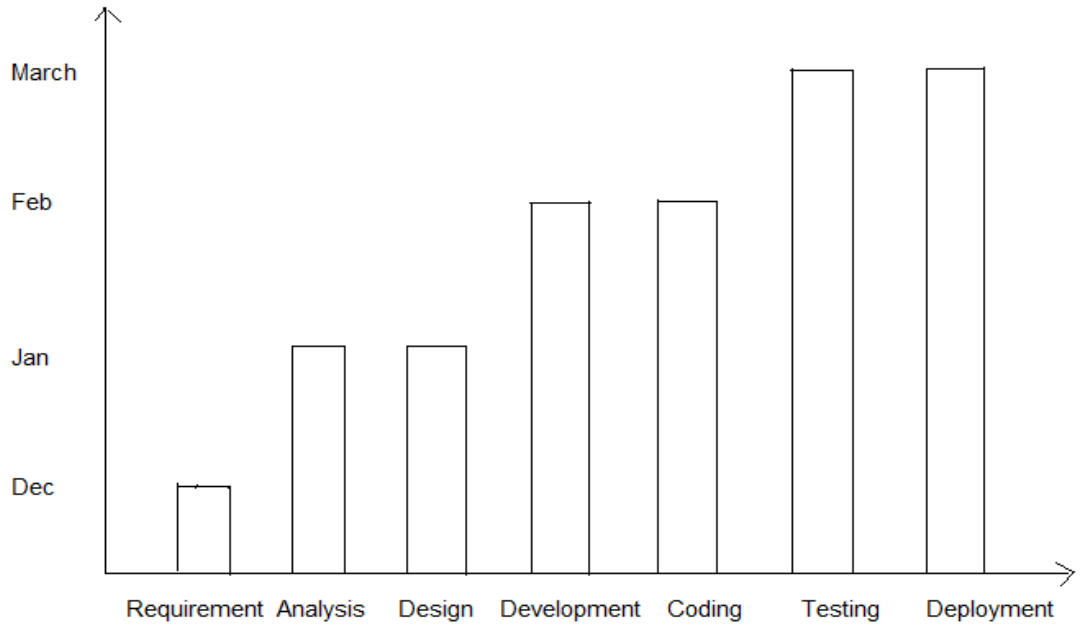


Diagram No. 4.1 Work Structure

4.4.1 Architecture

After requirements gathering and calculating estimations, we need to develop "system architecture". We knew that, Architecture diagram would help us visualize the high-level, overall structure of our system or application, in order to ensure the system meets our users' needs. Using architecture diagram, we could also describe patterns that are used throughout the design.

Some of the benefits of developing 'system architecture':

1) It gives a basis for analysis of software systems' behavior before the system has been built. The ability to verify that a future software system fulfills its users' needs without actually having to build it represents substantial cost-saving and risk-mitigation. A number of techniques have been developed to perform such analyses, such as ATAM.
2) It provides a basis for re-use of elements and decisions. A complete software architecture or parts of it, like individual architectural strategies and decisions, can be re-used across multiple systems whose stakeholders require similar quality attributes or functionality, saving design costs and mitigating the risk of design mistakes.
3) It supports early design decisions that impact a system's development, deployment, and maintenance life. Getting the early, high-impact decisions right is important to prevent schedule and budget overruns.
4) It helps in risk management. Software architecture helps to reduce risks and chance of failure.
5) It enables cost reduction. Software architecture is a means to manage risk and costs in complex IT projects.
6) For that, we studied about users who are involved

Efforts needs to be done:

1) Understand users in system.
2) Understand requirements
3) Understand Flow of data in system
4) Recognize technical resources

4.4.2 'AwareBot' architecture:

We successfully developed system architecture as below:

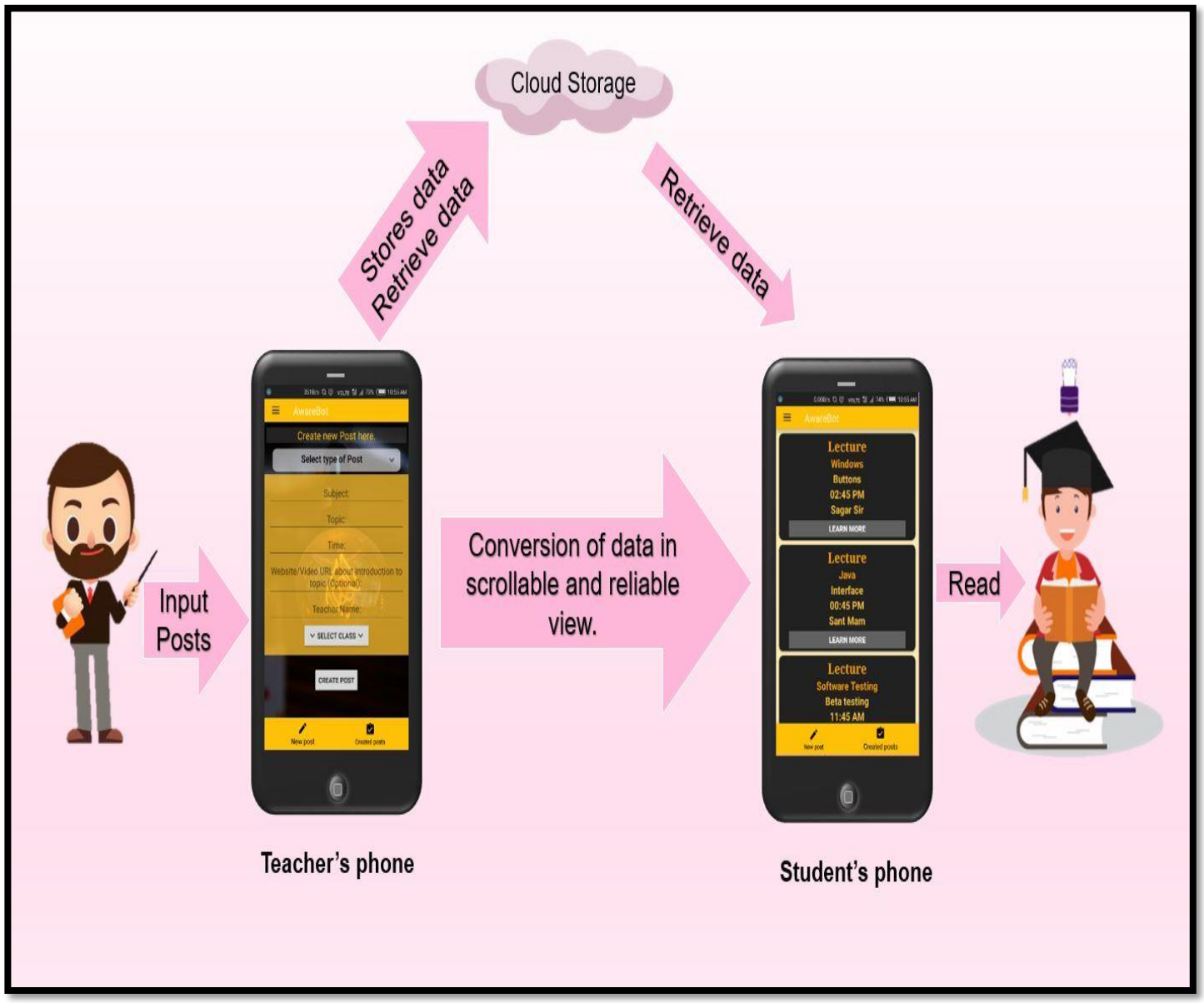


Diagram 4.2: 'AwareBot' architecture

CHAPTER 5: PROJECT SCHEDULE

5.1 PROJECT WORK BREAK DOWN STRUCTURE

Dividing complex projects to simpler and manageable tasks is the process identified as Work Breakdown Structure (WBS).

Usually, the project managers use this method for simplifying the project execution. In WBS, much larger tasks are broken down to manageable chunks of work. These chunks can be easily supervised and estimated.

WBS is not restricted to a specific field when it comes to application. This methodology can be used for any type of project management.

Following are a few reasons for creating a WBS in a project:

1) Accurate and readable project organization.
2) Accurate assignment of responsibilities to the project team.
3) Indicates the project milestones and control points.
4) Helps to estimate the cost, time and risk.
5) Illustrate the project scope, so the stakeholders can have a better understanding of the same.

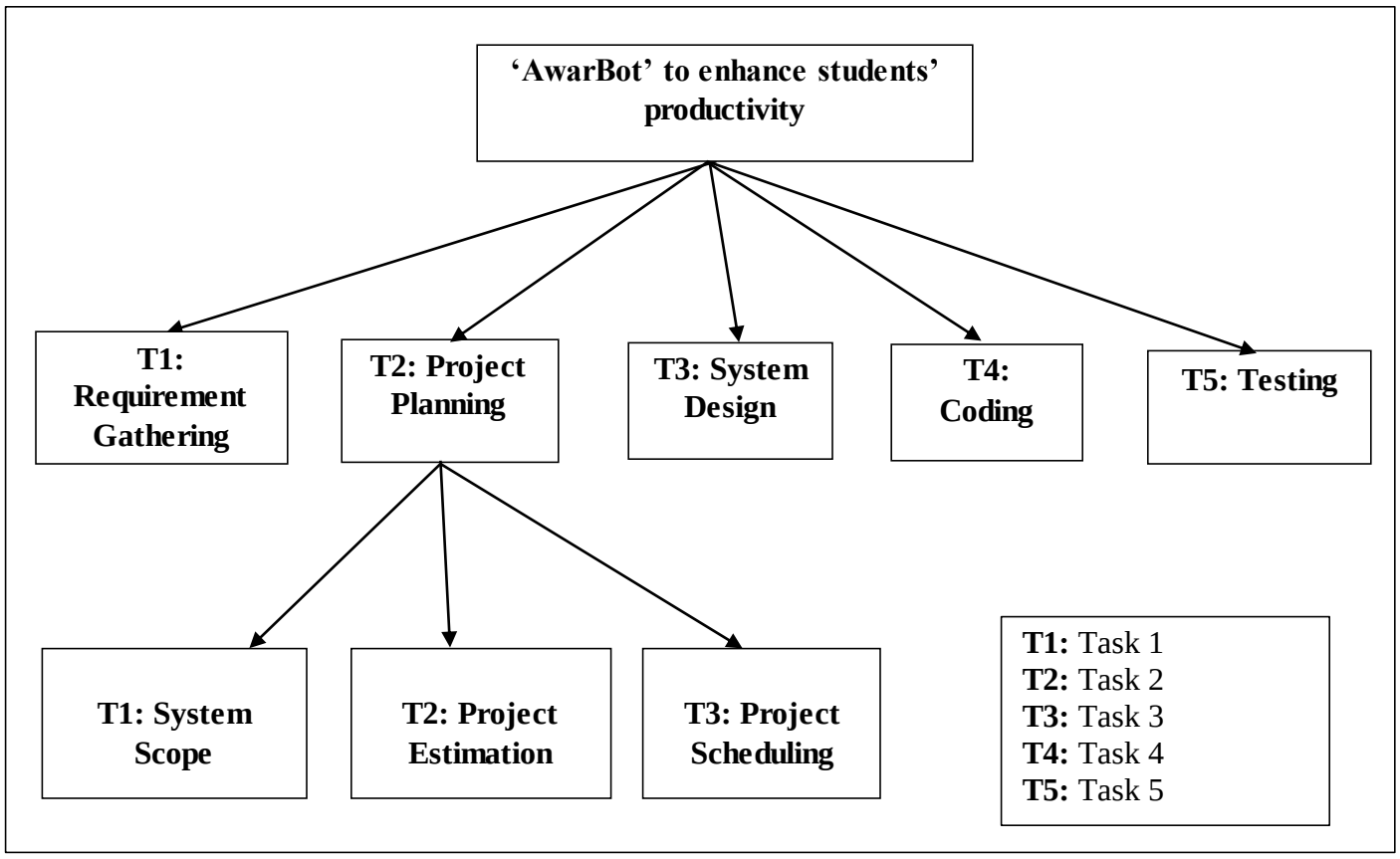


Diagram 5.1. Project Work Break Down Structure.

5.2 Team Structure:

5.3 Project Scheduling:

Project scheduling is a mechanism to communicate what tasks need to get done and which organizational resources will be allocated to complete those tasks in what timeframe. A project schedule is a document collecting all the work needed to deliver the project on time.

Before going deeper into project scheduling, let's review the fundamentals to project scheduling. Project scheduling occurs during the planning phase of the project. You have to ask yourself three questions to start:

1. What needs to be done?
2. When will it be done?

3. Who will do it?

Our project schedule was as below:

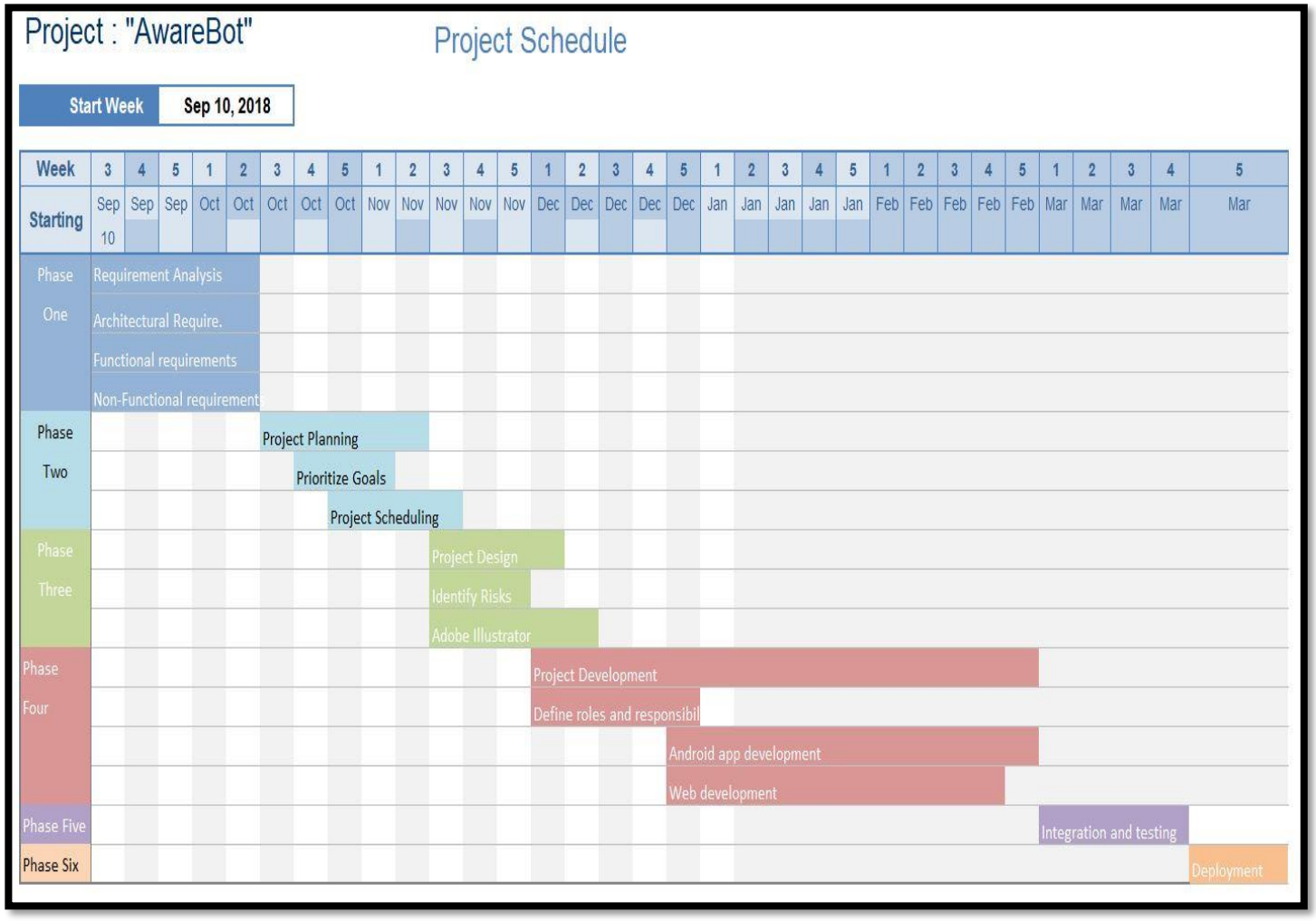


Diagram No. 5.1

5.4 TASKS AND MILESTONE:

T1: Project topic searching

T2: Project Topic Selected

T3: Preparing and submitting synopsis

T4: Preparing presentation for selected topic

M1: Project Finalized

M2: Requirement analysis for the project

T5: Deciding Modules, model for implementation

M3: Requirement Analysis and Functional Requirement

T6: Preparing Report of project

M4: Modelling and Project Report Complete

T7: Implementation of project begins

CHAPTER 6: PROJECT IMPLEMENTATION

6.1 DETAILED WORKING OF PROJECT

The project illustrated in this paper is entirely based on the idea of increasing productivity of college's students and teachers. Teachers send posts after entering some data into in. Posts are classified as per their audience such as first year, second year and third year (students). That means, if teachers create post for first year students then application provides option to select post audience, and only first year students would be able to read the posts.

Our application provides security to both teachers and students. Which enables teachers to enter password to use application features. This will stop data temper. Also these passwords will be checked through database. Even someone knows password, it will be changed from database. Also students need passwords to enter into system. These passwords are managed by teachers. They can change it or see it.

We implemented some types of posts as result of teachers' requirements. These types help teachers to easily fill data and submit into database.

They are as below:

6.1.1 LECTURE

Whenever teachers want to conduct their lectures, they have to make post related to it for specific audience. Due to this student get alert. Sometimes they don't know which lecture is going to take place. So, they bunk lectures. And they miss important lectures. So, this section will help students to easily decide whether to attend it or not. Also we provided feature to "Learn More" option which will redirect student to website or video where he can learn on internet and easily generate interest in it. This will reduce his efforts for studying later. Also, many students don't study until exams come near. But they can watch at least two to three videos per day and learn something. So, this feature will generate interest in learning. "Teaching" becomes easy for teachers and "learning" becomes easy for students.

Fields related to Lecture:

1) **Subject**: Subject which teacher is going to teach

2) **Topic**: Topic in that subject which teacher going to teach
3) **Time**: Time on which lecture would start
4) **Website/Video URL**: Website or YouTube video link of that topic
5) **Teacher Name**: Name of teacher who is sending that post.

6.1.2 OFF-PERIOD

If teachers are not conducting lecture, they can use this option to enter data in application. This will save students' lot of time, and efforts. Some students come to college from long distances for attending specific lecture or difficult lectures. Sometimes, teachers have some other important works on same lecture time. So, it could become annoying for any student. To overcome this problem, which we also faced lot of time, our application helps a lot.

Teacher just has to create post of 'Off-period', and done. Also, teachers can suggest students, what they should do at such periods.

This has following fields:

1) **Subject**: Subject which teacher is not going to teach
2) **Suggestion**: Suggestion of tasks to do
3) **Time**: Time on which there is off-period
4) **Suggest Website/Video URL**: Suggestion of website or YouTube video link to watch and learn something
5) **Teacher Name**: Name of teacher who is sending that post.

6.1.3 NOTICE

Sometimes, teachers need to tell something very urgently to students. It can be of anything. Creating hard copies of notices and stick them on notice board takes time and also there is very less guarantee that it reaches up to everyone.

So, our app easily sends those notices in students' phone in very less time. This becomes it more smart and productive.

Fields related to notices:

1) **About**: Topic about which notice has to be sent
2) **Description**: Description of notice in short.
3) **Teacher Name**: Name of teacher who is sending that post.

6.1.4 CHECKING/SUBMISSION

Teachers can send alert of checking or submission to students. Students can easily understand which assignments/ practicals to complete and which study should be done to give oral to their teachers.

This includes following fields:

1) **Subject and experiment**: Subject and experiment to be checked
2) **Place**: Place of checking/submission
3) **Time**: Time of checking
4) **Teacher's Name**: Name of teacher who is sending that post

6.1.5 OTHER

There could be any other posts those teachers need to create. But if it's option is not available in our app, then they can create own custom post as per their requirement.

This includes following fields:

1) **Title**: Title of custom post
2) **Description**: Description of custom post
3) **Teacher**: Name of teacher who is sending that post
4) **Website/Video URL**: Optional website or video link related to that topic

Ex. If teacher wants to notify their students about some company's recruitment they can Title as "Recruitment Notification", description as 'vacancies and salary' and in website/video URL section they can enter company's official registration form's link.

So, after clicking on "Learn More" button application will redirect user directly to that website.

CHAPTER 7: ACTIVITY FLOW – USE CASES, DFD'S, CLASS DIAGRAM

7.1 DATA FLOW DIAGRAM

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system) The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

Android Application
DFD Level 0

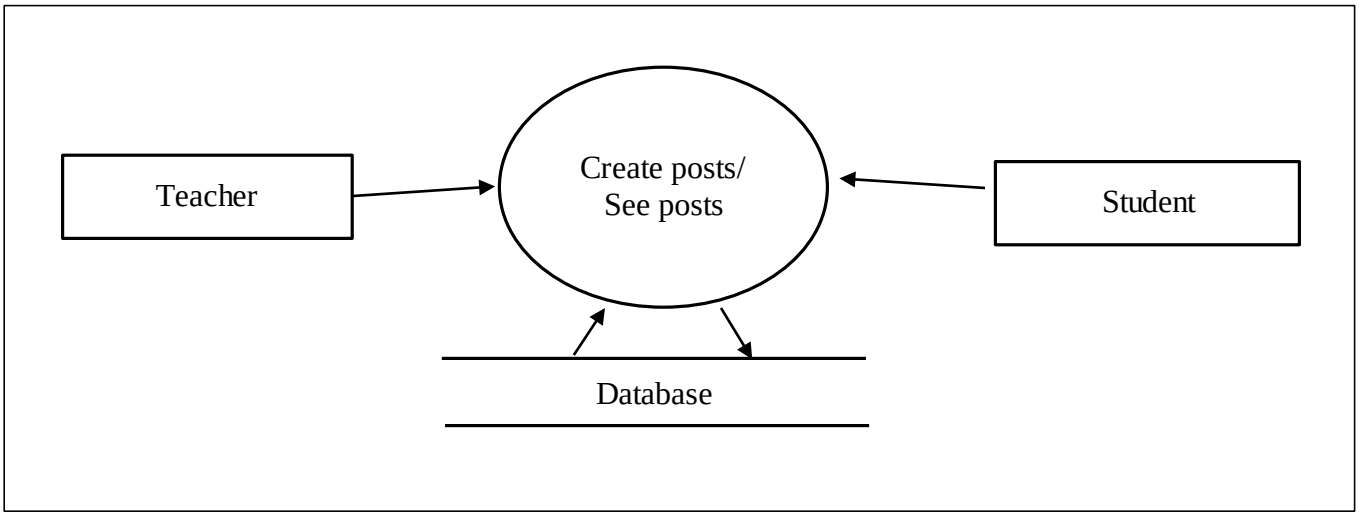


Fig 7.1.1 DFD Level 0

DFD Level 1

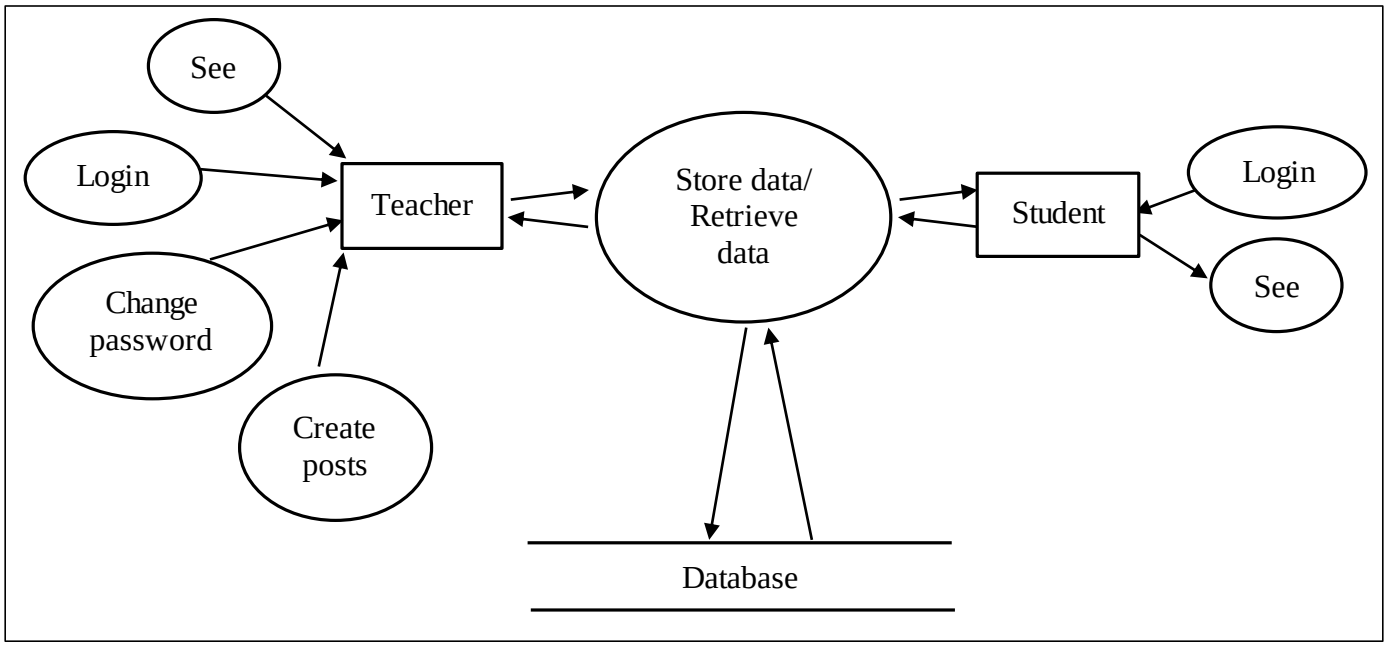


Fig 7.1.2 DFD Level 1

Website
DFD Level 0

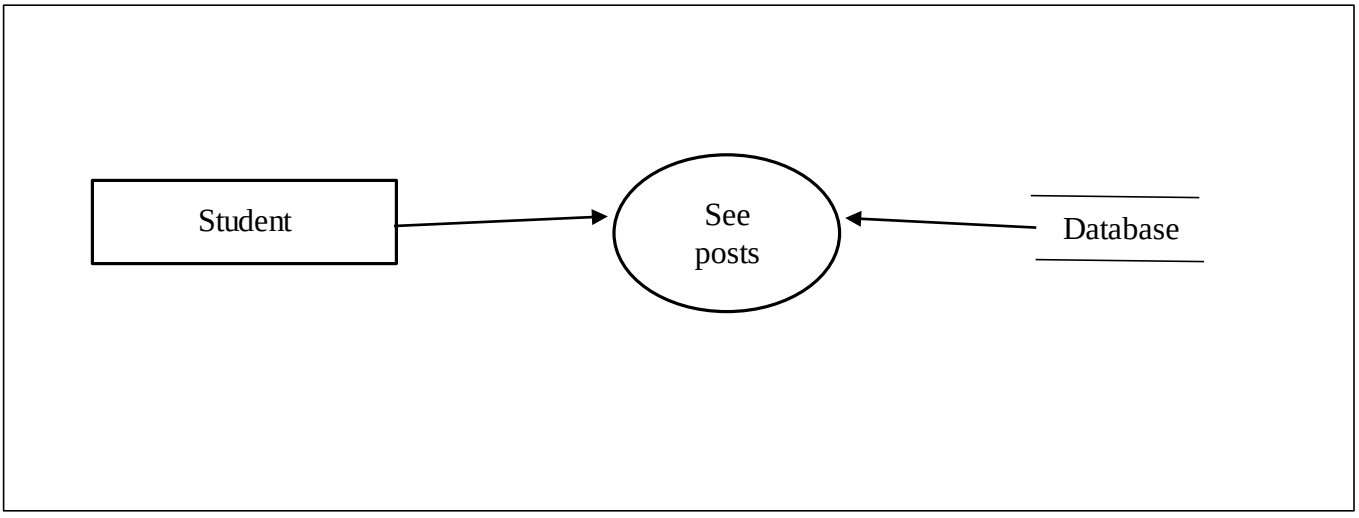


Fig 7.1.3 DFD Level 0 (Website)

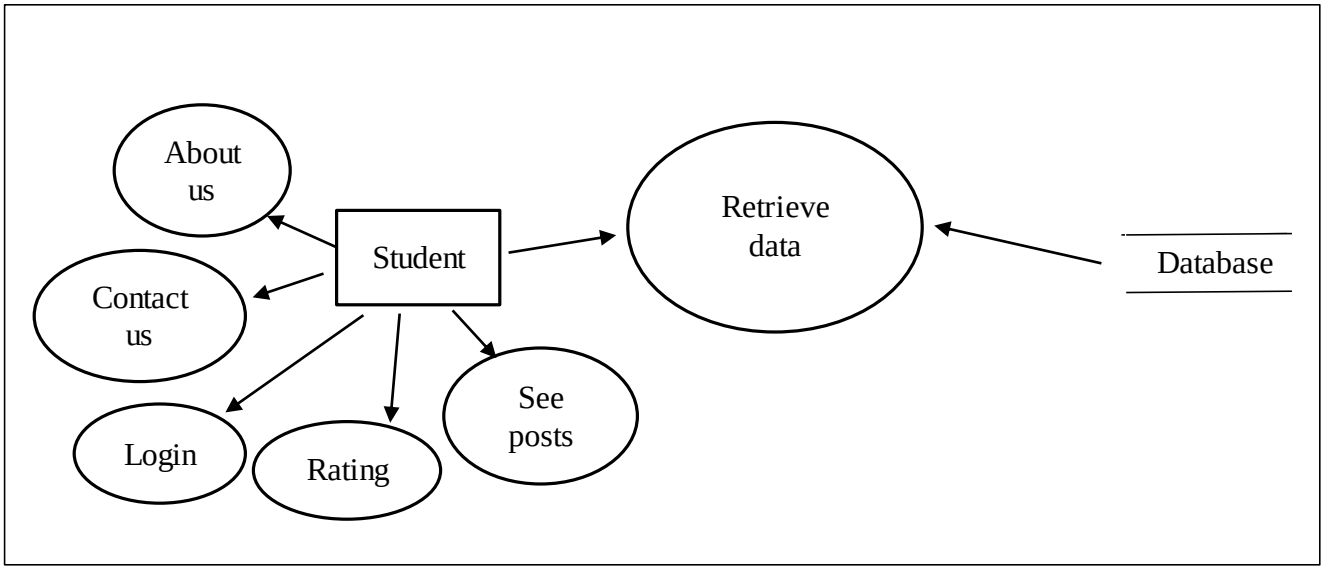


Fig 7.1.4 DFD Level 1(Website)

7.2 USE CASE DIAGRAM

A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform. In this context, a "system" is something being developed or operated, such as a web site. The "actors" are people or entities operating under defined roles within the system.

Some Benefits:

1) Use case diagrams are valuable for visualizing the functional requirements of a system that will translate into design choices and development priorities.
2) They also help identify any internal or external factors that may influence the system and should be taken into consideration.
3) They provide a good high level analysis from outside the system. Use case diagrams specify how the system interacts with actors without worrying about the details of how that functionality is implemented.

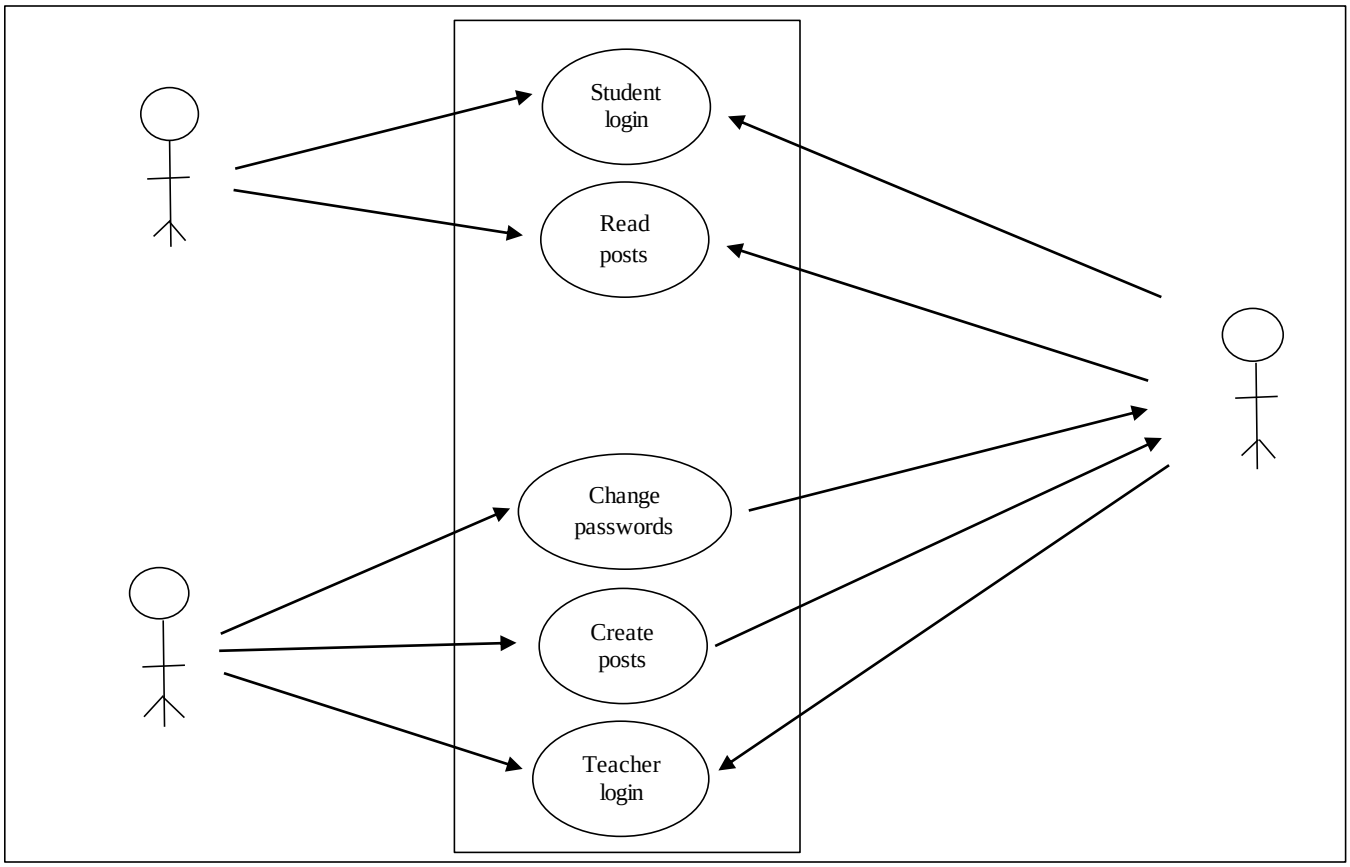


Fig 7.2.1 Use case diagram

CHAPTER 8: PROJECT IMPLEMENTATION TESTING

8.1 TEST CASES

8.1.1 TEST CASES FOR TEACHER SECTION

Test case Id	Test Objective	Prerequisites	Steps	I/P data	Expected Result	Actual Result	Status
TC_1	Login	Login activity opened	1.Enter incorrect password 2.Enter correct password	1. null value. 2. admin 3.admin123 4.admin@123 5.Correct password	If password is correct, teacher should login into system.	Teacher logged in, after entering correct password	PASS
TC_2	"Lecture" post creation	Teacher logged into system	1.Select post type - "Lecture" 2.Fill into subject field 3.Fill into topic field 4.Select time 5. Enter URL 6. Enter Teacher Name 7. Select class	1. Empty fields 2. "Math" 3. "Integration" 4. Select time 5. "Shinde sir" 6. Select "First year"	1. If data is not filled, it must display error message. 2. It should display "Post Created" after all data entered.	1. If data is not filled, it displays error message. 2. It displays "Post Created" after all data entered.	PASS
TC_3	"Off Period" post creation	Teacher logged into system	1.Select post type -" Off Period" 2.Fill into Subject field 3.Fill into suggestion field 4.Select time 5. Enter URL 6. Enter Teacher	1. Empty fields 2. "Math" 3. "Complete assignments" 4. Select time 5. "Shinde sir" 6. Select "First year"	1. If data is not filled, it must display error message. 2. It should display "Post Created" after all data entered.	1. If data is not filled, it displays error message. 2. It displays "Post Created" after all data entered.	PASS

			Name 7. Select class				
TC_4	"Notice" post creation	Teacher logged into system	1.Select post type -" Notice" 2.Fill into "about" field 3.Fill into "description" field 4. Enter Teacher Name 5. Select "First year"	1. Empty fields 2. "Seminar" 3. "Come at Seminar hall on 11AM" 4. "Shinde sir" 5. Select "First year"	1. If data is not filled, it must display error message. 2. It should display "Post Created" after all data entered.	1. If data is not filled, it displays error message. 2. It displays "Post Created" after all data entered.	PASS
TC_5	"Alert" post creation	Teacher logged into system	1.Select post type -"Alert -> Checking/Submission" 2.Fill into "Subject& Experiment" field 3.Fill into "Place" field 4. Enter into "time" field 5.Enter into Teacher name field	1. Empty fields 2."Linux-grep command" 3. "Lab 2" 4. Select time 5. "Rasane Mam 6. "Third Year"	1. If data is not filled, it must display error message. 2. It should display "Post Created" after all data entered.	1. If data is not filled, it displays error message. 2. It displays "Post Created" after all data entered.	PASS
TC_6	"Other" post creation	Teacher logged into system	1.Select post type -"Other" 2.Fill into title field 3.Fill into description	1. Empty fields 2. "Seminar" 3. "Campus Recruitment" 4. "Fill form	1. If data is not filled, it must display error message.	1. If data is not filled, it displays error message. 2. It	PASS

			field 4. Enter Teacher Name 5. Select class	of Infosys company recruitment before 3 april." 5. "Shinde sir" 6. "www.infosys.com/careers"	2. It should display "Post Created" after all data entered.	displays "Post Created" after all data entered.	

Table 8.1 Cases of test

8.1.2 TEST CASES FOR STUDENT SECTION

Test case Id	Test Objective	Prerequisites	Steps	I/P data	Expected Result	Actual Result	Status
TC_1	Login	Student Login activity opened	1.Enter incorrect password 2.Enter correct password	1. null value. 2. admin 3.admin123 4.admin@123 5.Correct password	If password is correct, teacher should login into system.	Teacher logged in, after entering correct password	PASS
TC_2	Learn More	Student logged in successfully	1.Click on "learn on" button of different posts	N/A	Web view should open related to corresponding post	Web view opened related to corresponding post	PASS

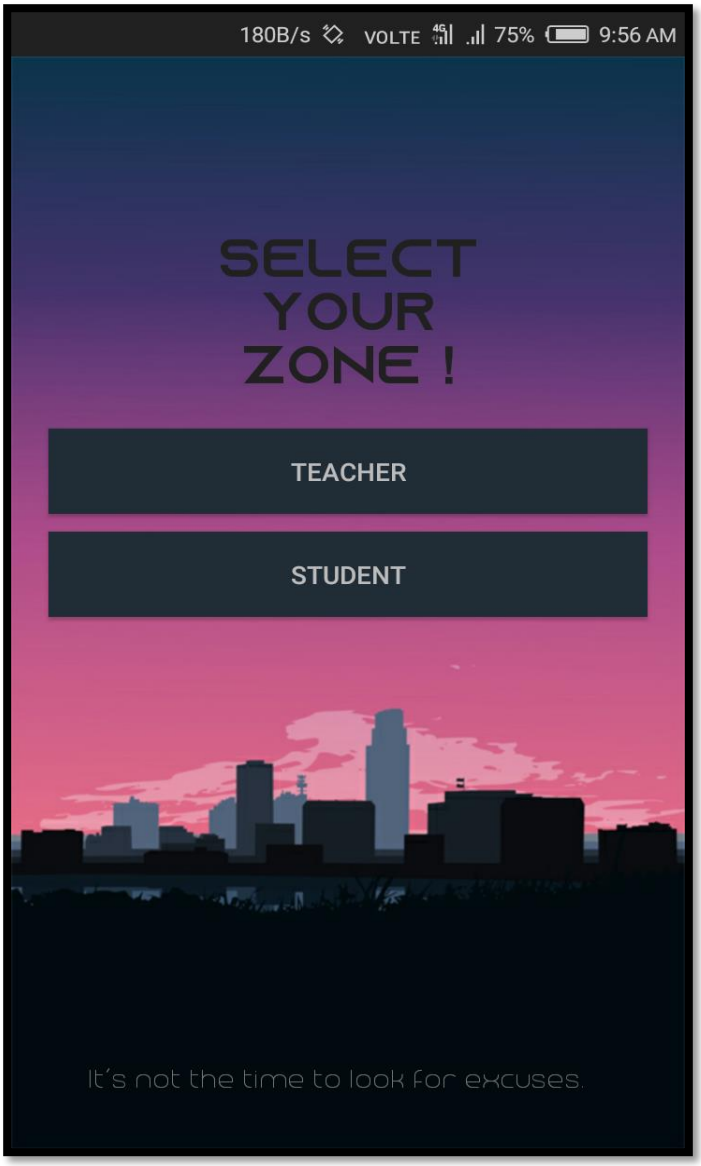
8.2 TYPES OF TEST

The purpose of testing to discover the error. Testing is the process of discover every conceivable fault or weakness in a work product.

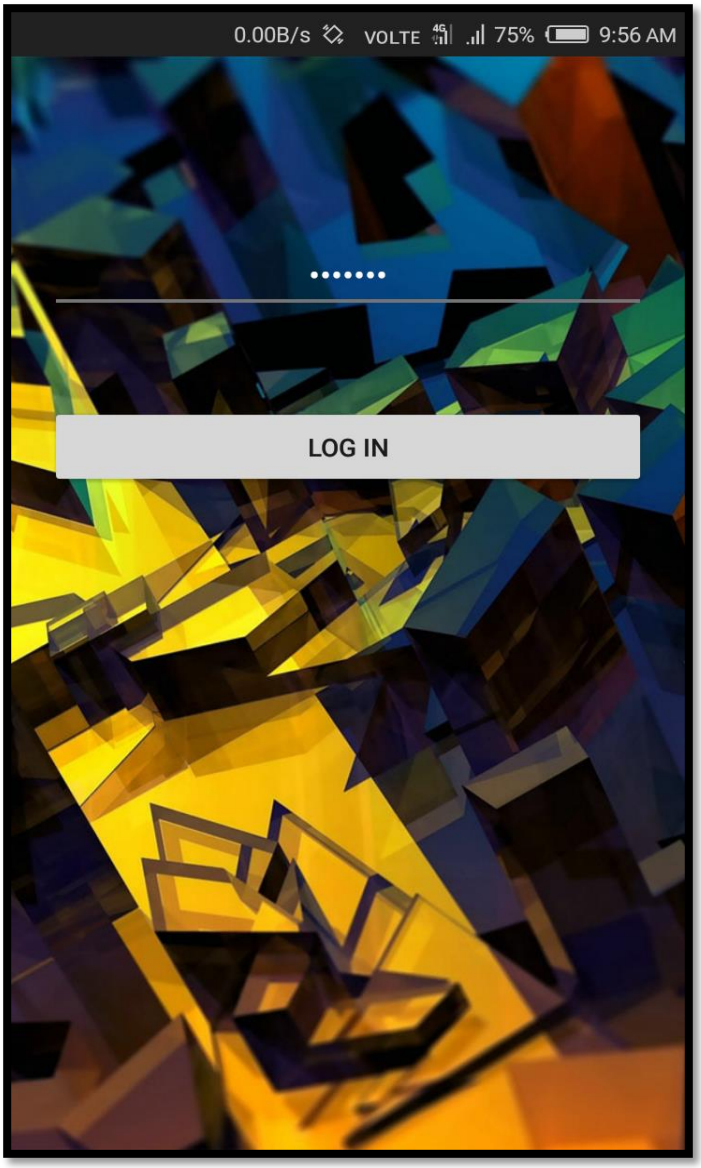
It provides way to check functionality of the component, sub-assemblies, assemblies & a product. It is the process of exercising software with the intent of ensuring that the software system meets the requirement & user expectation & does not that the unacceptable manner.

1) Unit Testing
2) Integration Testing
3) Functional testing
4) System testing
5) White Box Testing
6) Black Box Testing

CHAPTER 9: GUI IMPLEMENTATION(SNAPSHOTS):

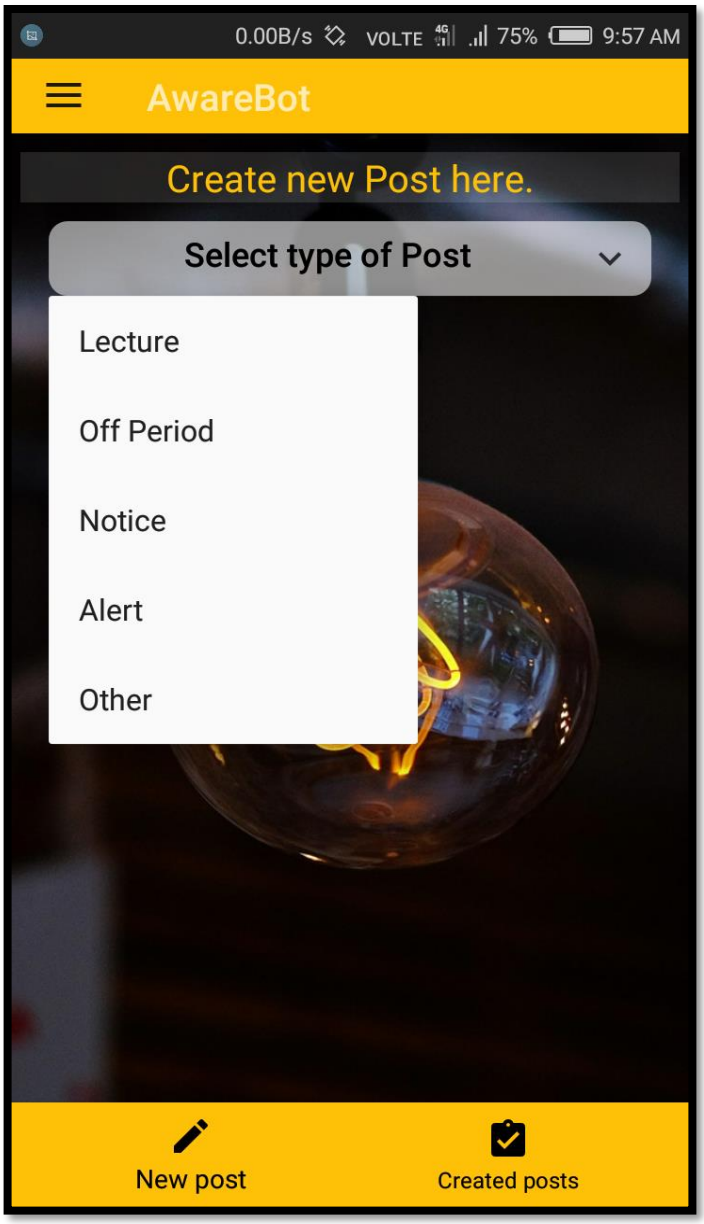


Main Home activity

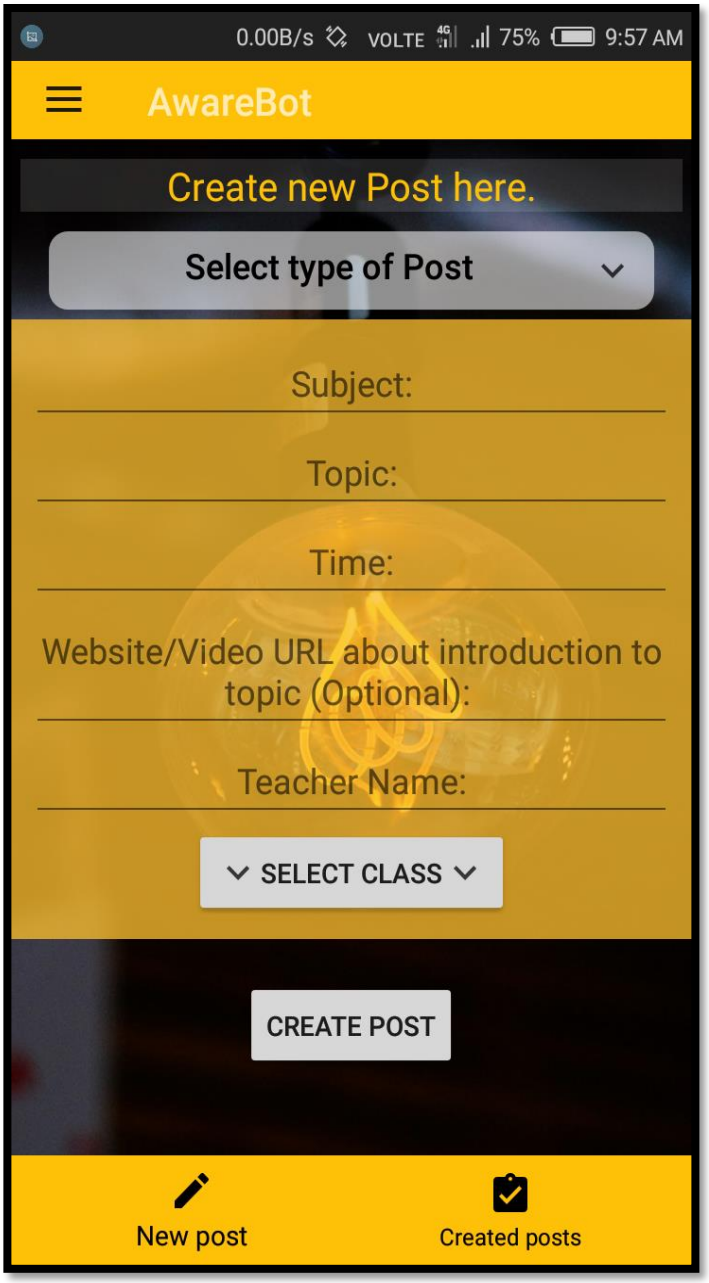


Teacher Login

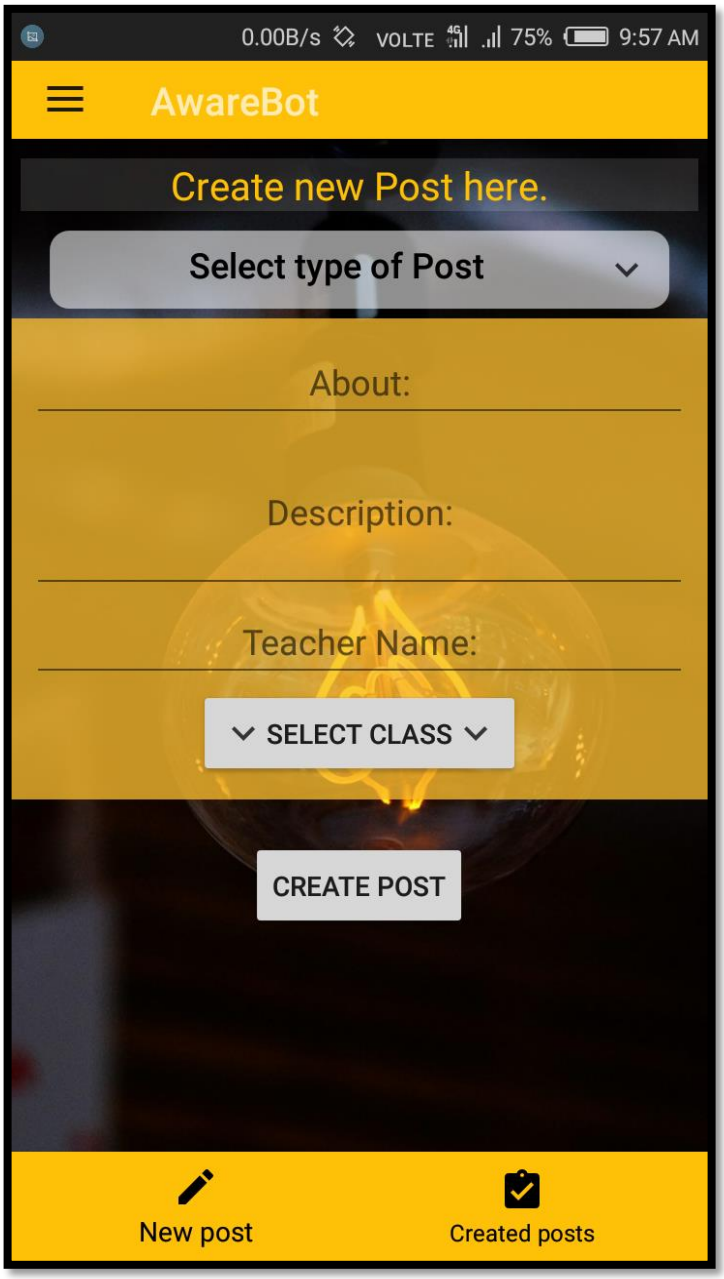
Teacher Home Activity



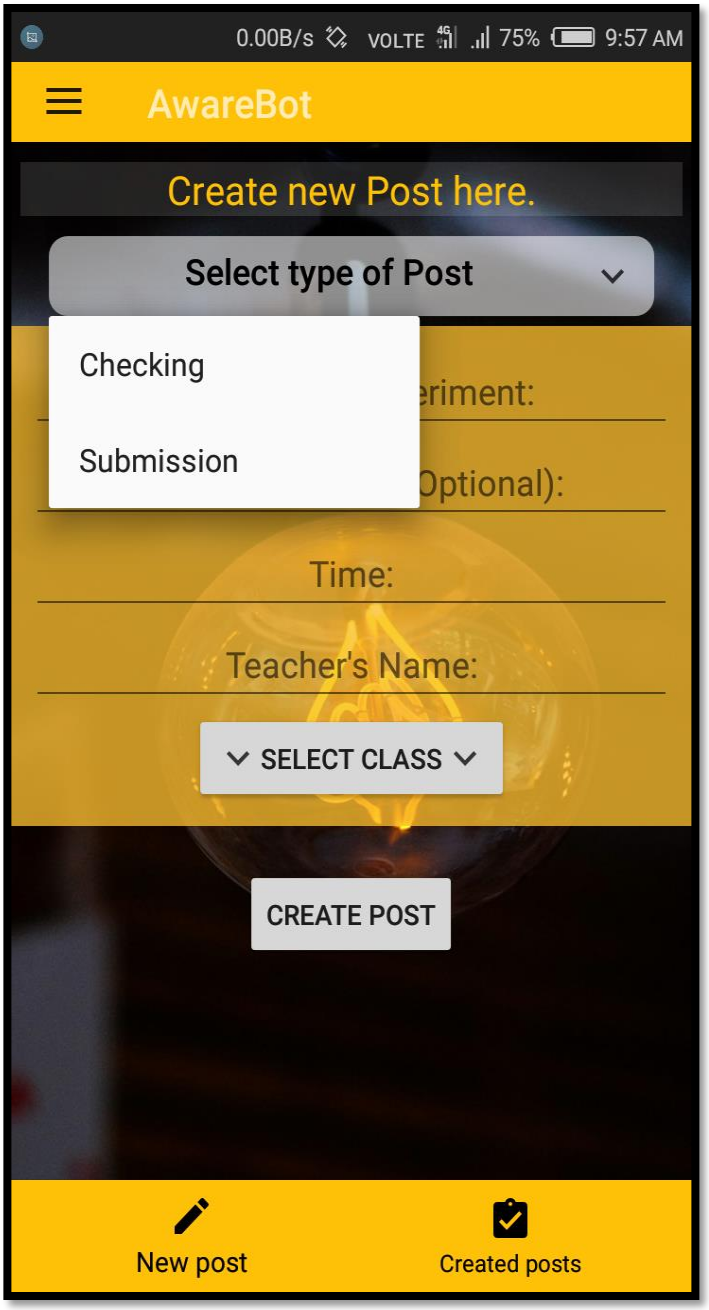
Post Options



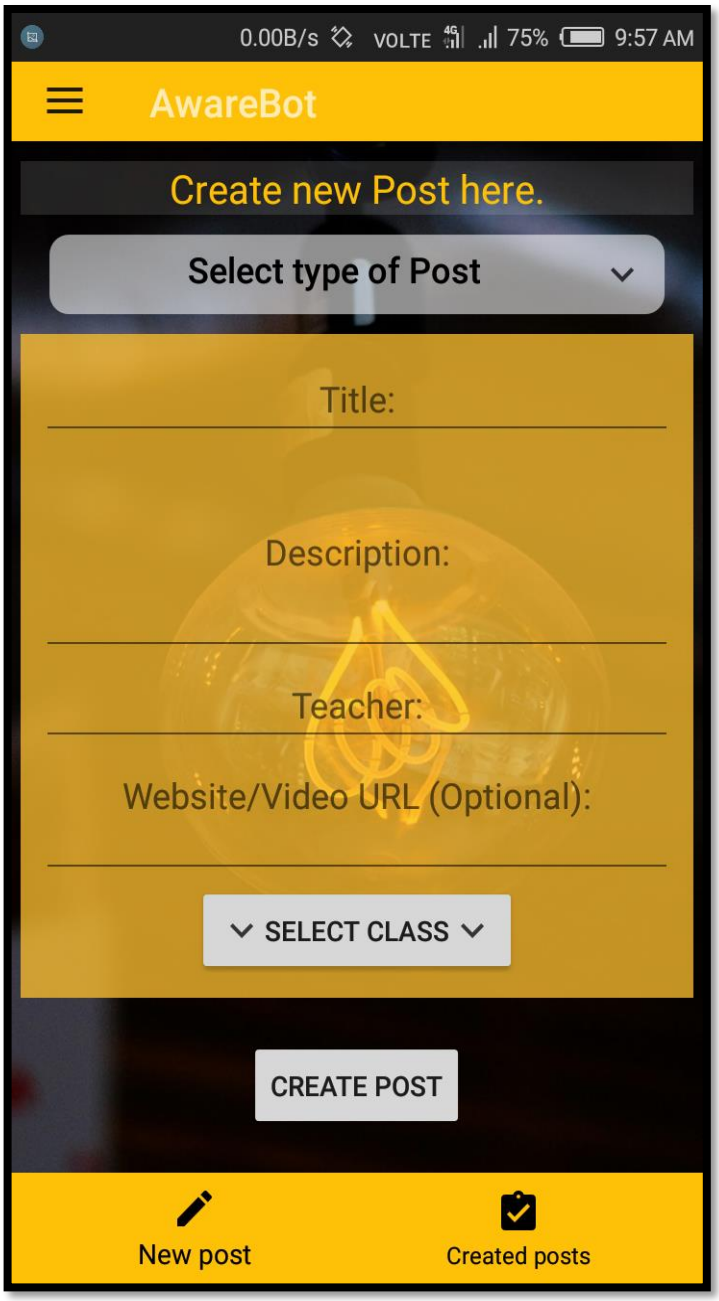
"Lecture" Option



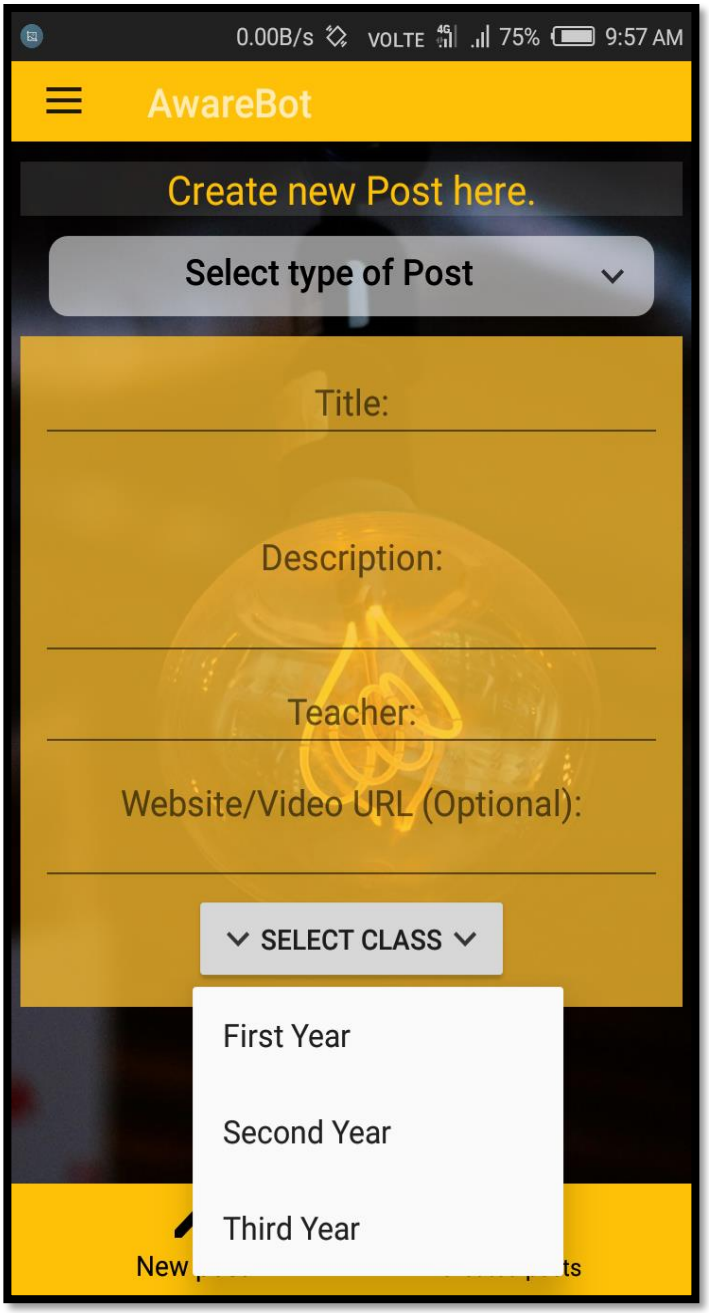
"Notice" option



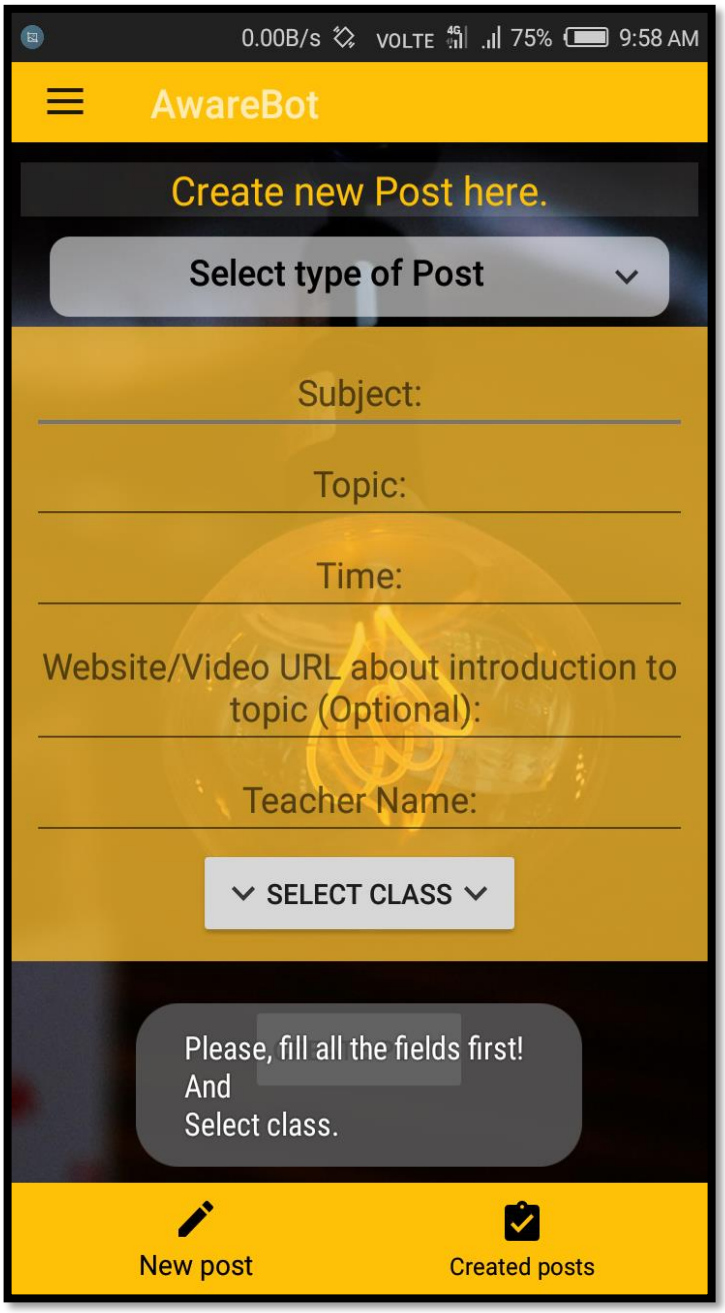
"Alert" option



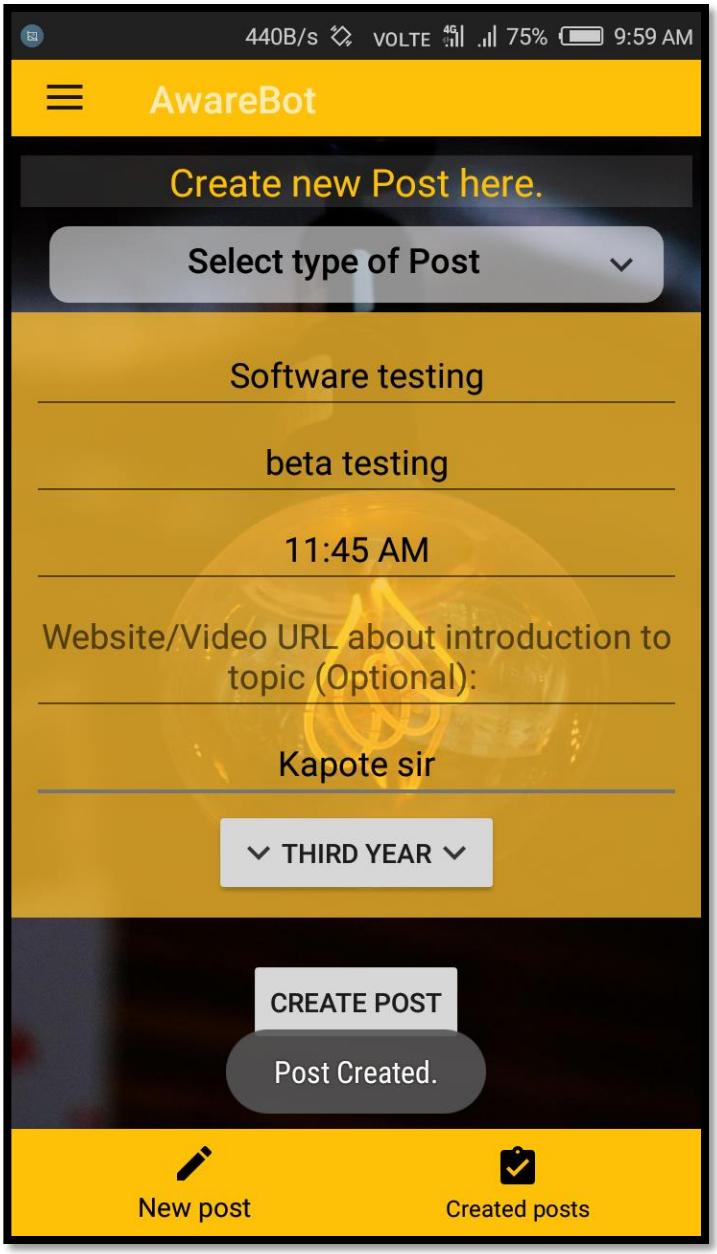
"Other" option



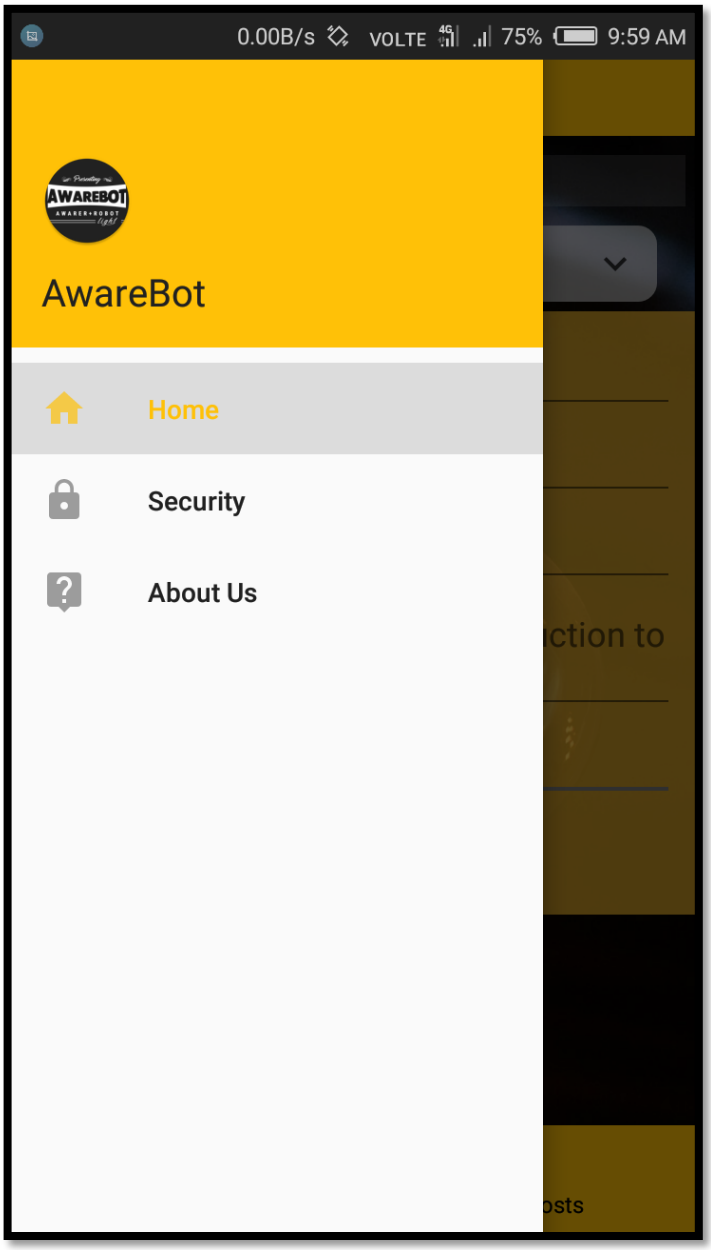
Post Audience Selection



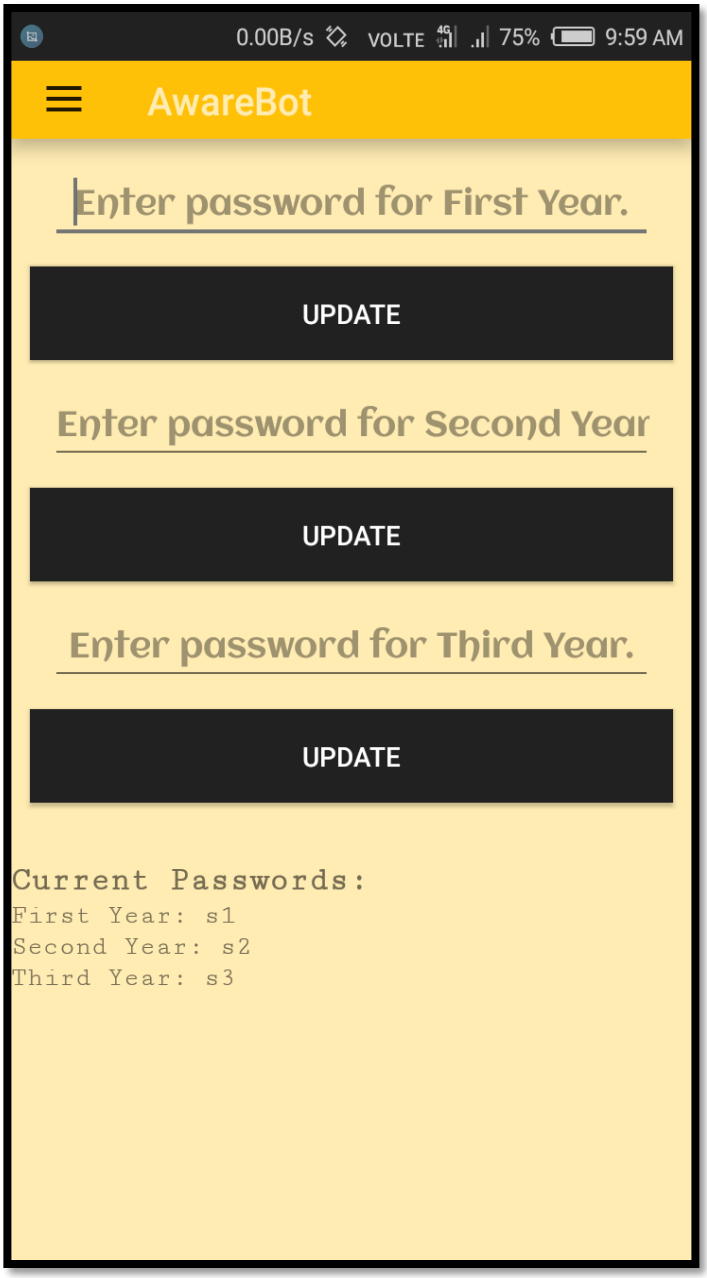
Data entry errors management



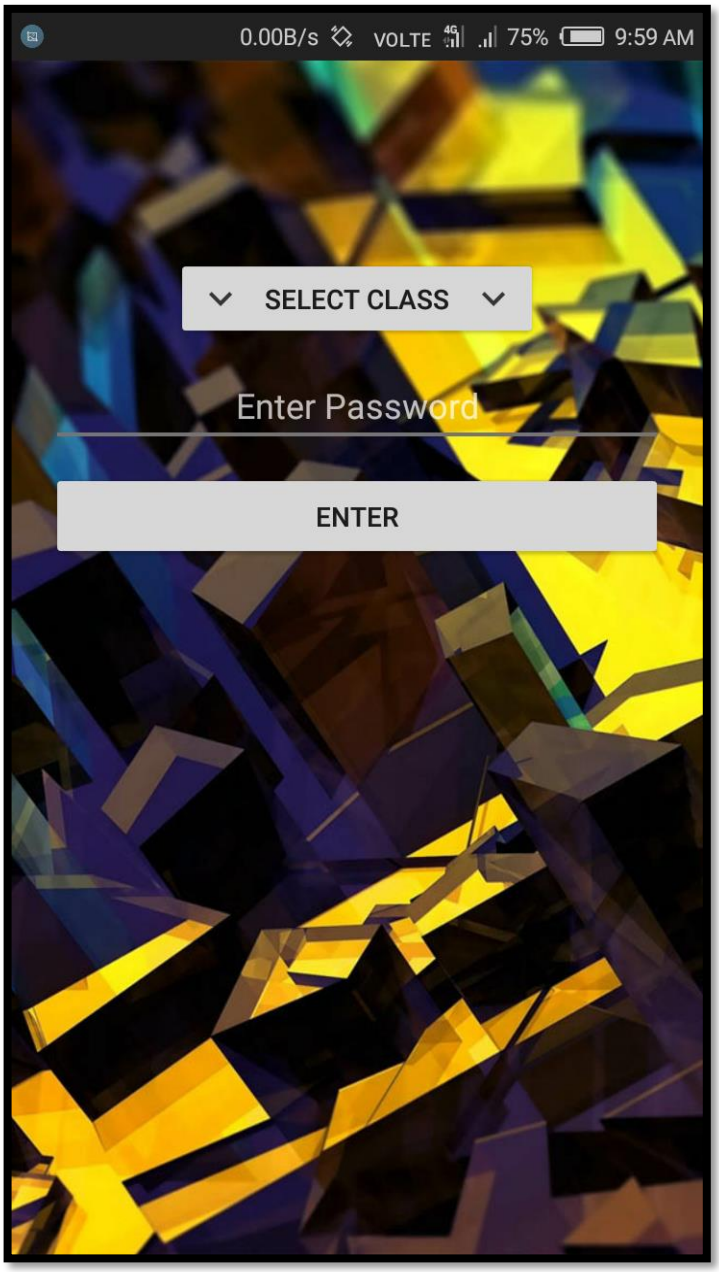
Post Creation



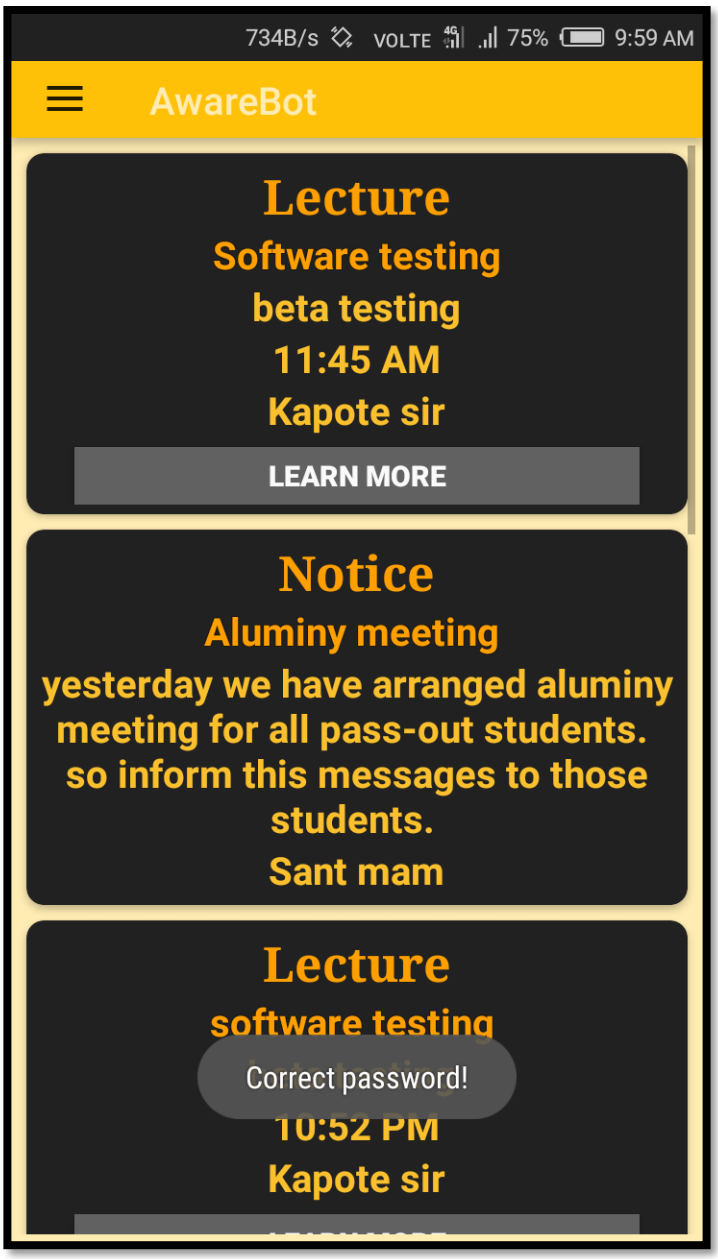
Application Drawer



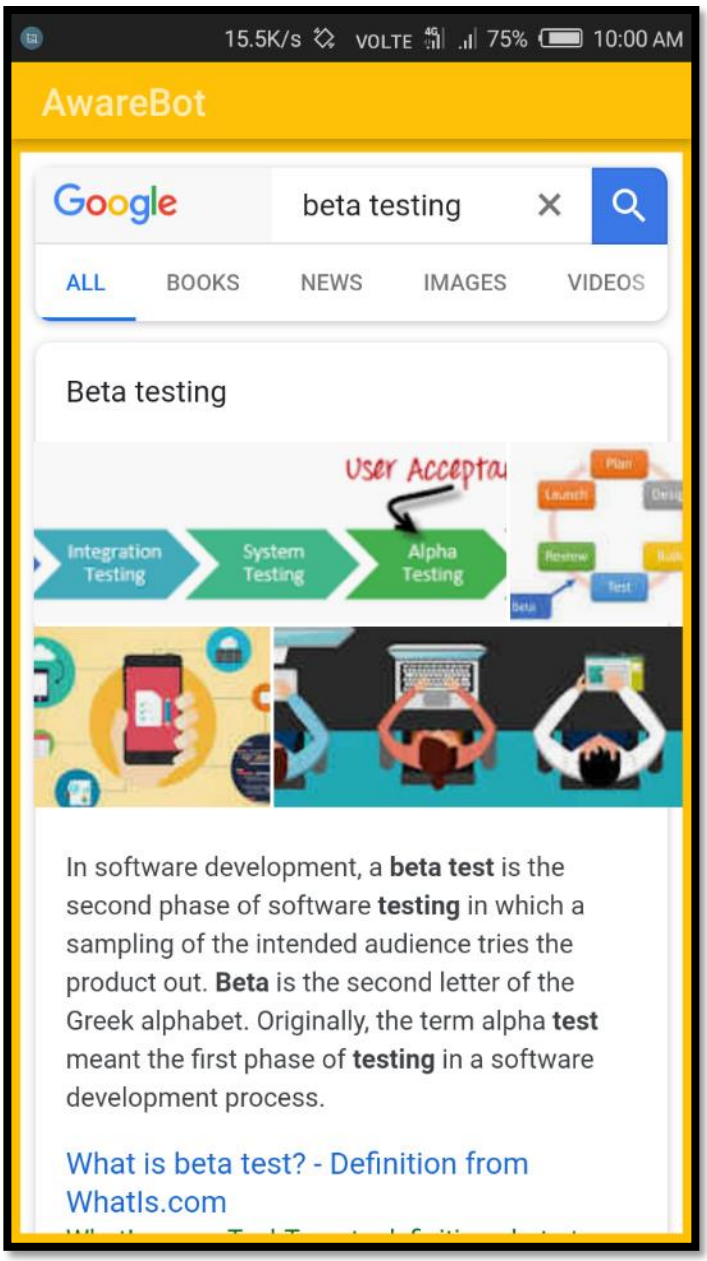
Security Fragment



Student Login

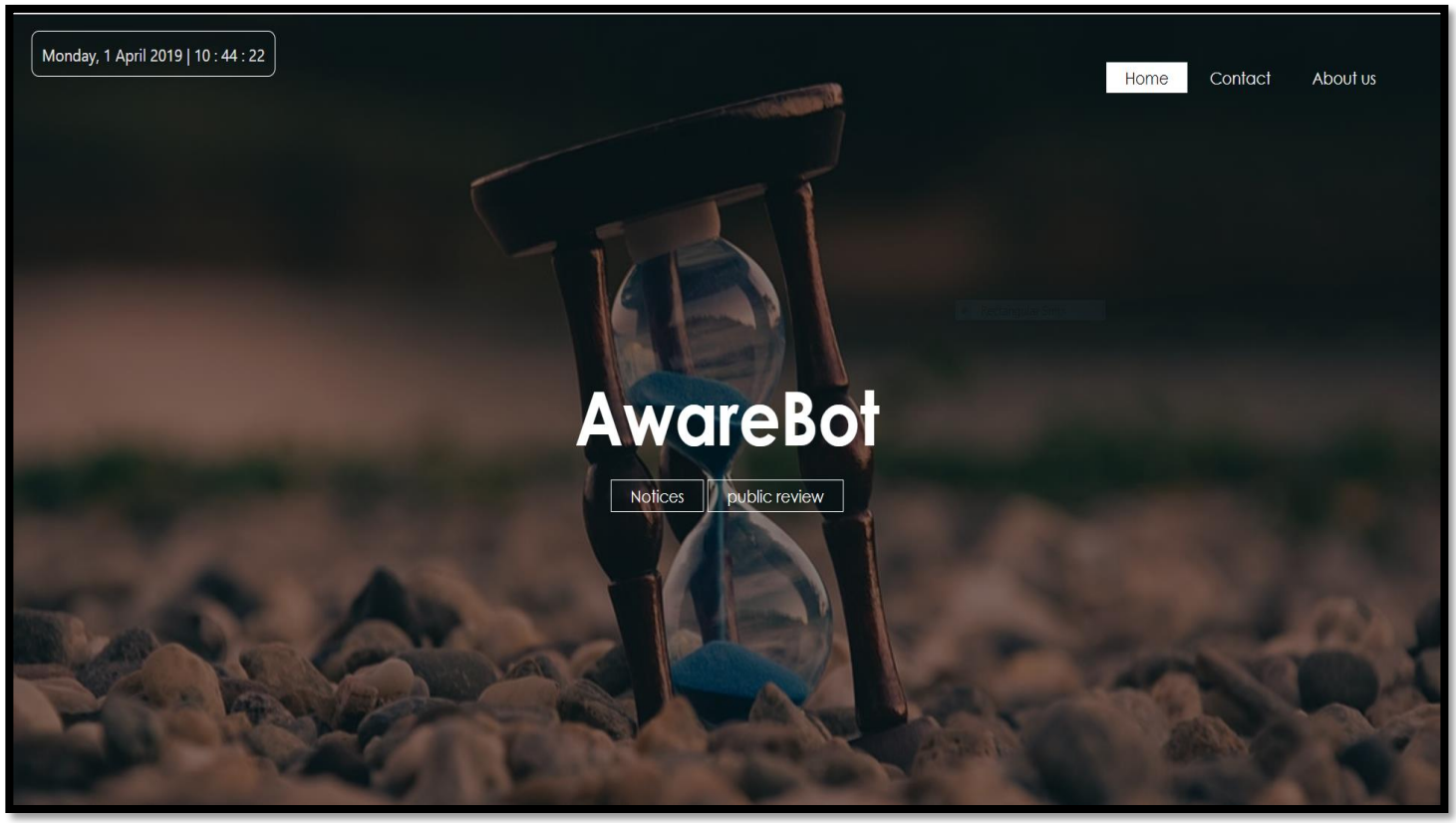


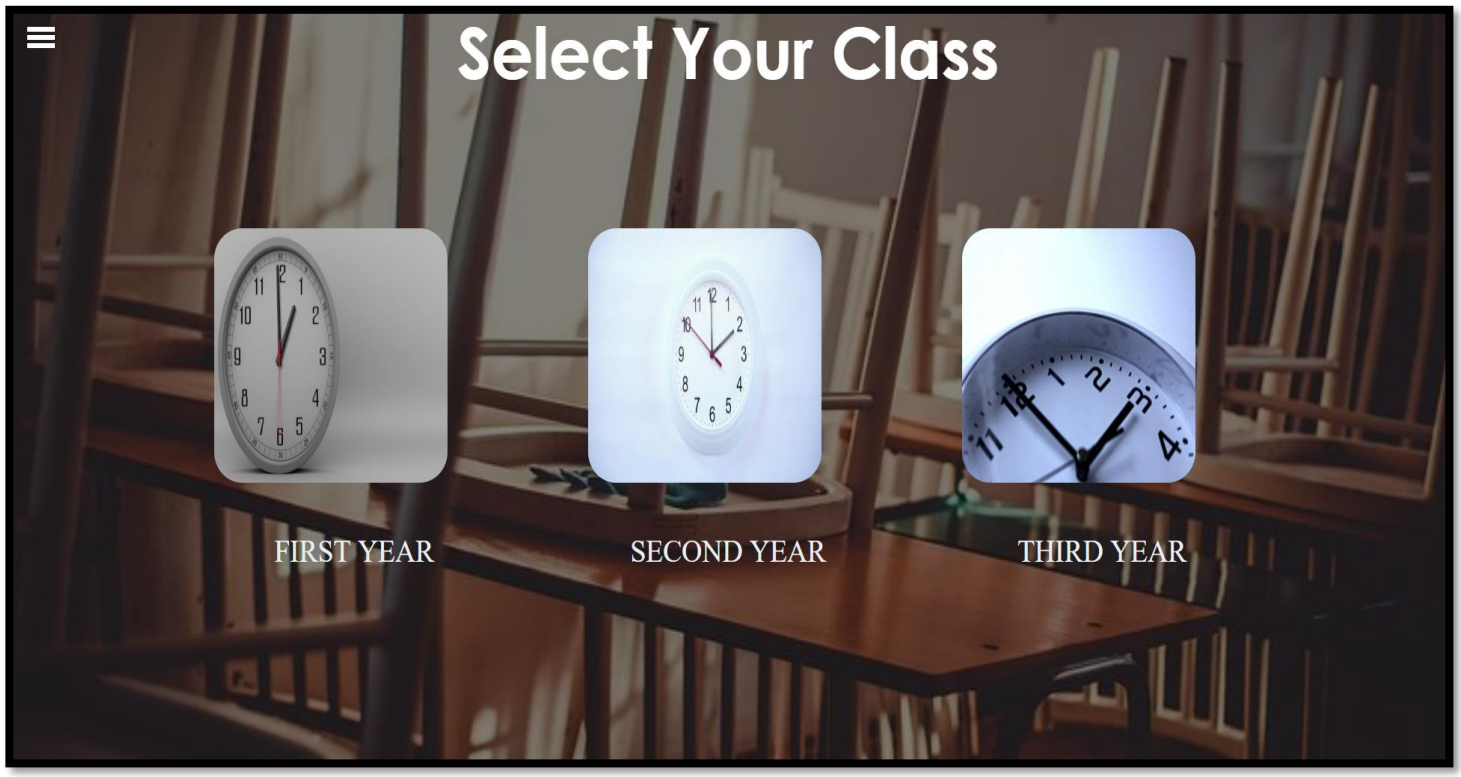
Student Home activity

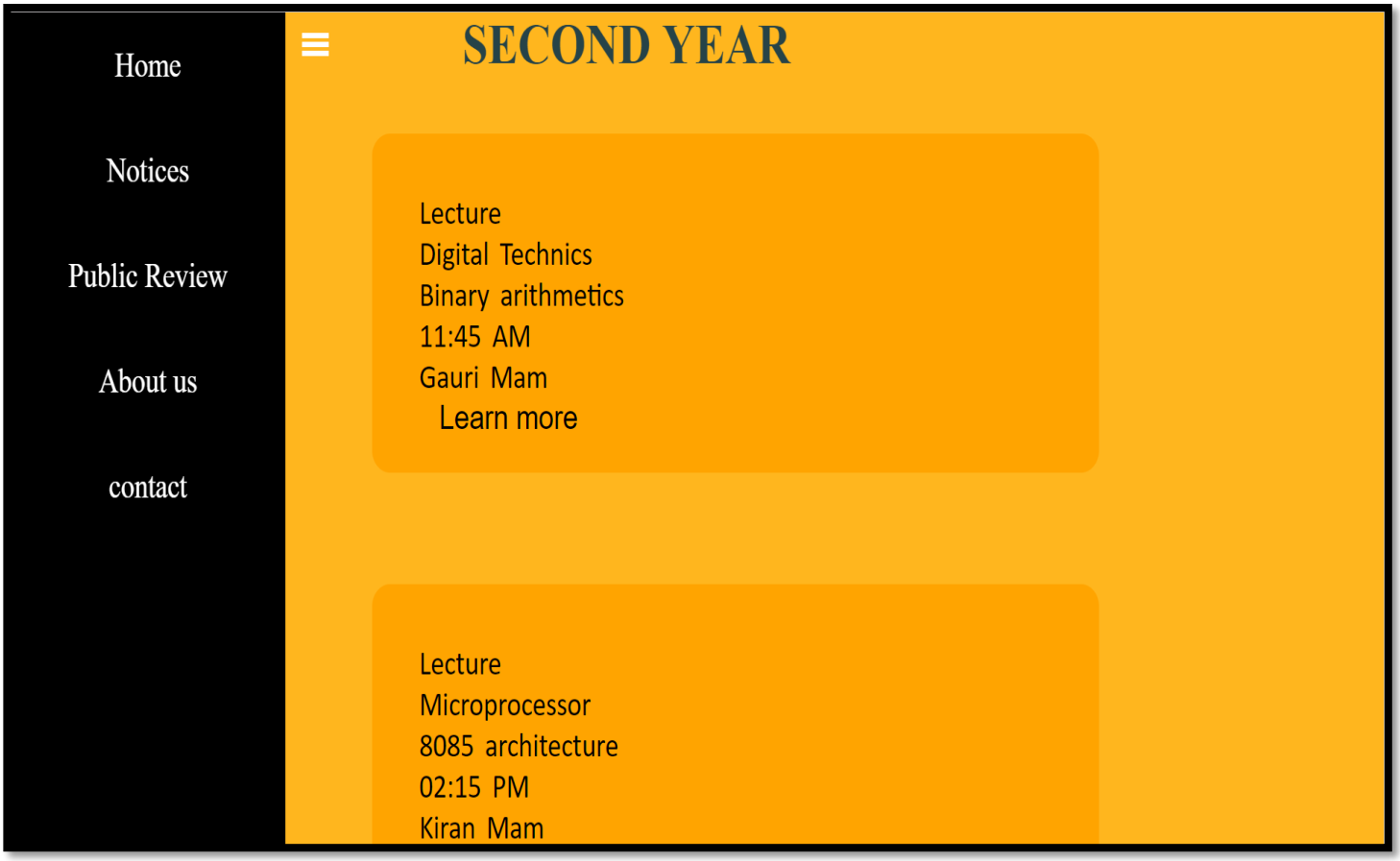


After clicking "Learn More"

Website Snapshots:

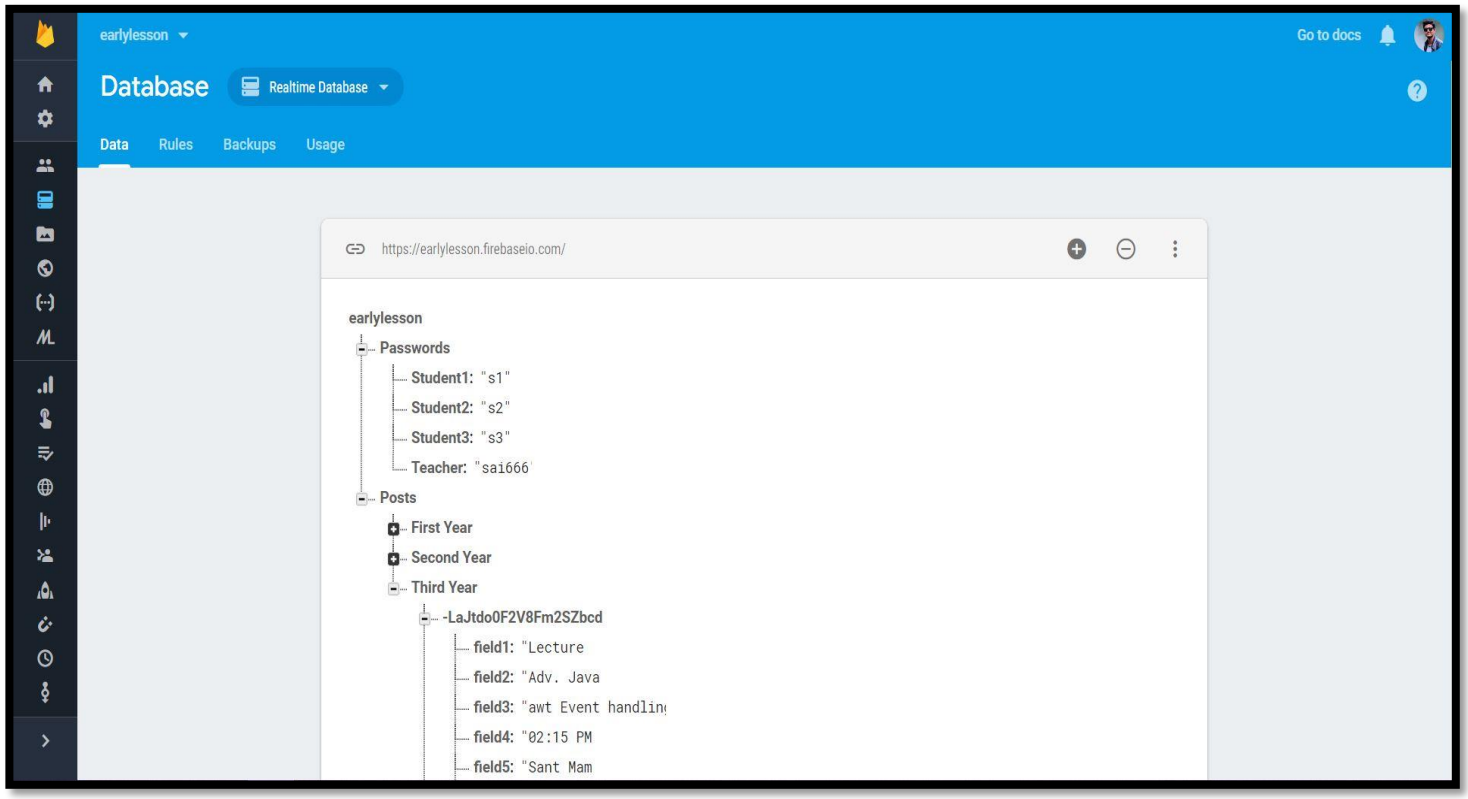


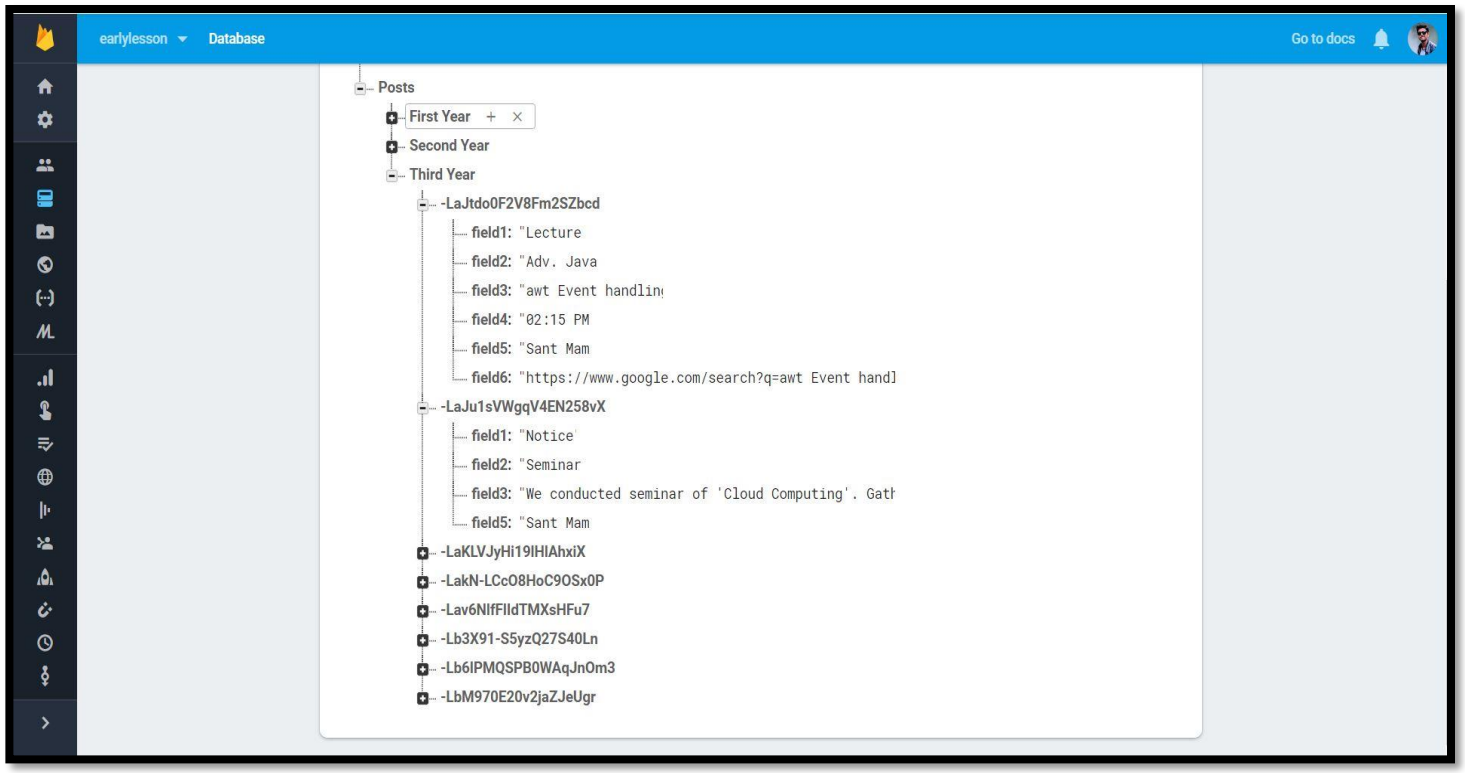




Student Home page

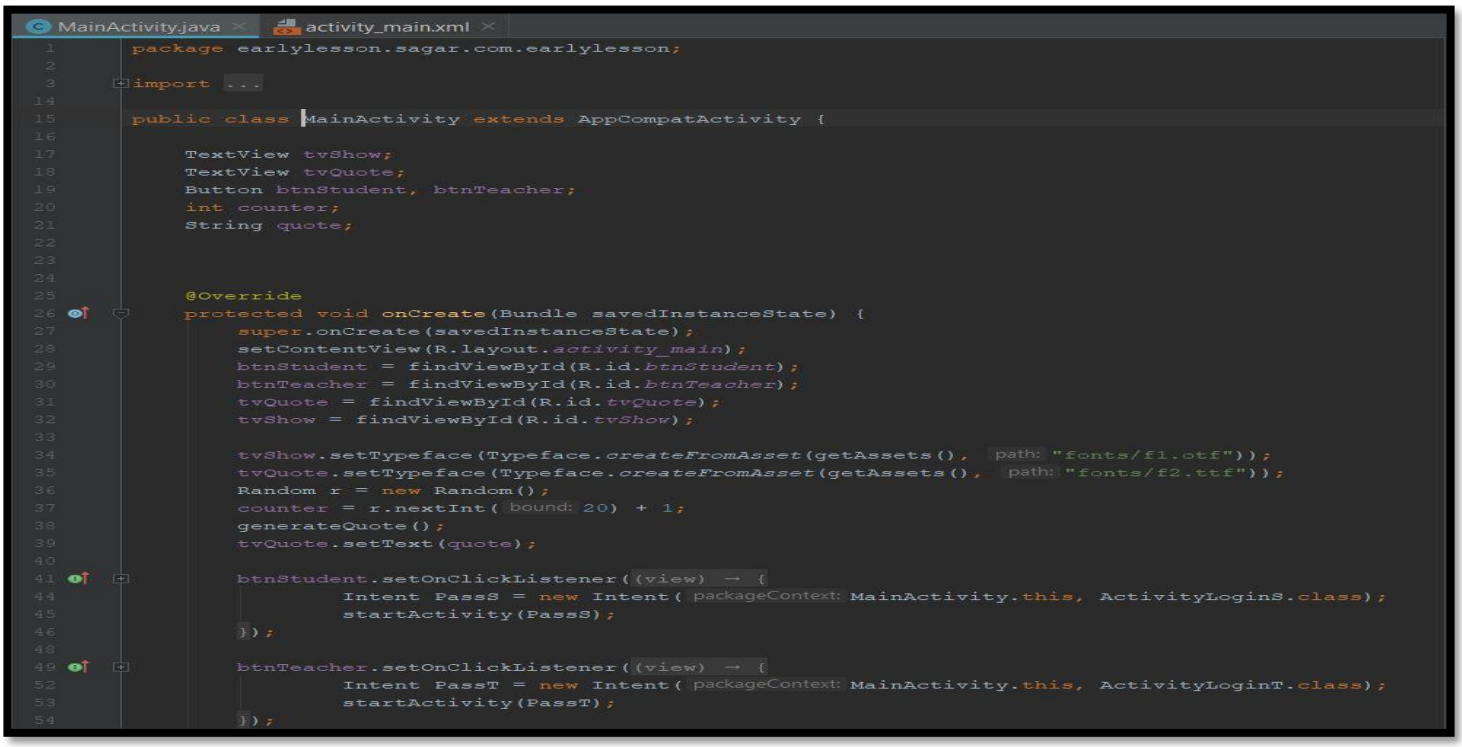
Application database- Firebase database:





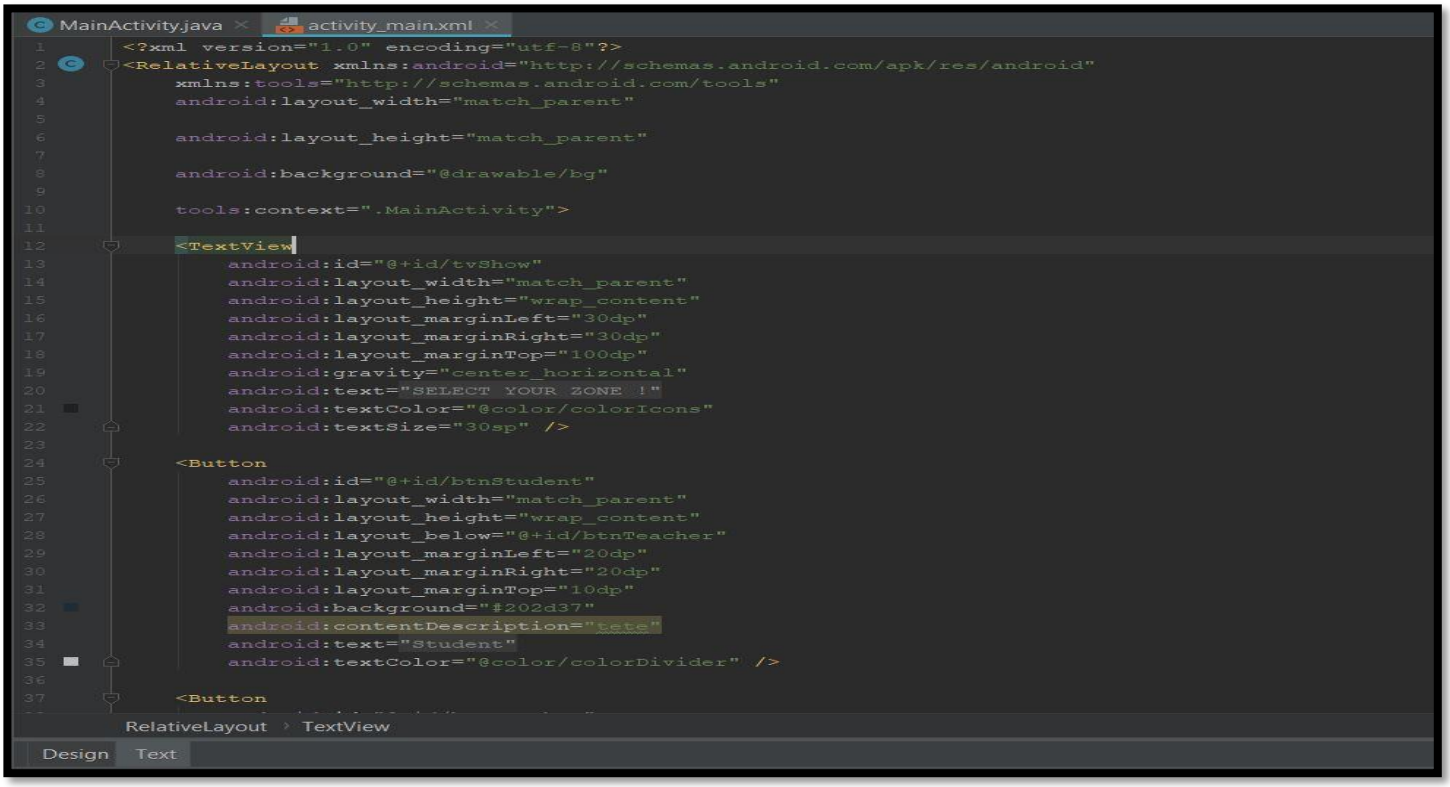
Code Snapshots:

Main activity java file:



```
package earlylesson.sagar.com.earlylesson;

import ...

public class MainActivity extends AppCompatActivity {

    TextView tvShow;
    TextView tvQuote;
    Button btnStudent, btnTeacher;
    int counter;
    String quote;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btnStudent = findViewById(R.id.btnStudent);
        btnTeacher = findViewById(R.id.btnTeacher);
        tvQuote = findViewById(R.id.tvQuote);
        tvShow = findViewById(R.id.tvShow);

        tvShow.setTypeface(Typeface.createFromAsset(getAssets(), path: "fonts/f1.otf"));
        tvQuote.setTypeface(Typeface.createFromAsset(getAssets(), path: "fonts/f2.ttf"));
        Random r = new Random();
        counter = r.nextInt( bound: 20) + 1;
        generateQuote();
        tvQuote.setText(quote);

        btnStudent.setOnClickListener((view) -> {
                Intent PassS = new Intent( packageContext: MainActivity.this, ActivityLoginS.class);
                startActivity(PassS);
        });

        btnTeacher.setOnClickListener((view) -> {
                Intent PassT = new Intent( packageContext: MainActivity.this, ActivityLoginT.class);
                startActivity(PassT);
        });
```

Respective xml file:



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:background="@drawable/bg"

    tools:context=".MainActivity">

    <TextView
        android:id="@+id/tvShow"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginRight="30dp"
        android:layout_marginTop="100dp"
        android:gravity="center_horizontal"
        android:text="SELECT YOUR ZONE !"
        android:textColor="@color/colorIcons"
        android:textSize="30sp" />

    <Button
        android:id="@+id/btnStudent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/btnTeacher"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:layout_marginTop="10dp"
        android:background="#202d37"
        android:contentDescription="tete"
        android:text="Student"
        android:textColor="@color/colorDivider" />

    <Button
```

Teacher Home activity java file:

THomeFragment.java | thome_fragment.xml

```
package earlylesson.sagar.com.earlylesson;


import ...

public class THomeFragment extends Fragment {
    Button btn_alert_type;

    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
        return inflater.inflate(R.layout.thome_fragment, container, attachToRoot: false);

    }

    @Override
    public void onViewCreated(@NonNull View view, @Nullable Bundle savedInstanceState) {
        btn_alert_type = view.findViewById(R.id.btn_alert_type);
        btn_alert_type.setOnClickListener((view) -> {
                PopupMenu popup = new PopupMenu(getActivity(), btn_alert_type);

                popup.getMenuInflater().inflate(R.menu.type_popup, popup.getMenu());

                popup.setOnMenuItemClickListener((item) -> {
                        switch (item.getItemId()) {
                            case R.id.menu_lecture:
                                getChildFragmentManager().beginTransaction().replace(R.id.fragment_container_child, new PostTypeLecture()).commit();
                                break;
                            case R.id.menu_off_period:
                                getChildFragmentManager().beginTransaction().replace(R.id.fragment_container_child, new PostTypeOffPeriod()).commit();
                                break;
                            case R.id.menu_notice:
                                getChildFragmentManager().beginTransaction().replace(R.id.fragment_container_child, new PostTypeNotice()).commit();
                                break;
                            case R.id.menu_checking:
                                getChildFragmentManager().beginTransaction().replace(R.id.fragment_container_child, new PostTypeChecking()).commit();
                                break;
                            case R.id.menu_submission:
                                getChildFragmentManager().beginTransaction().replace(R.id.fragment_container_child, new PostTypeSubmission()).commit();
```

THomeFragment

Respective xml file:

THomeFragment.java | thome_fragment.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/bg3">


    <TextView
        android:id="@+id/tv_tp1"
        android:layout_width="match_parent"
        android:layout_height="27dp"
        android:layout_centerInParent="false"
        android:layout_marginLeft="5dp"
        android:layout_marginRight="5dp"
        android:layout_marginTop="10dp"
        android:background="#20FFFFFF"
        android:gravity="center_vertical|center_horizontal"
        android:text="Create new Post here."
        android:textColor="@color/colorPrimary"
        android:textSize="20sp" />
    <Button
        android:id="@+id/btn_alert_type"
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:layout_below="@id/tv_tp1"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:textColorHint="#000000"
        android:layout_marginTop="10dp"
        android:background="@drawable/et_bg"
        android:drawableEnd="@drawable/more"
        android:gravity="center_horizontal"
        android:hint="Select type of Post"
        android:paddingBottom="5dp"
        android:paddingLeft="10dp"
        android:paddingRight="10dp"
        android:paddingTop="5dp"
```

RelativeLayout › TextView

Student Home activity java file:

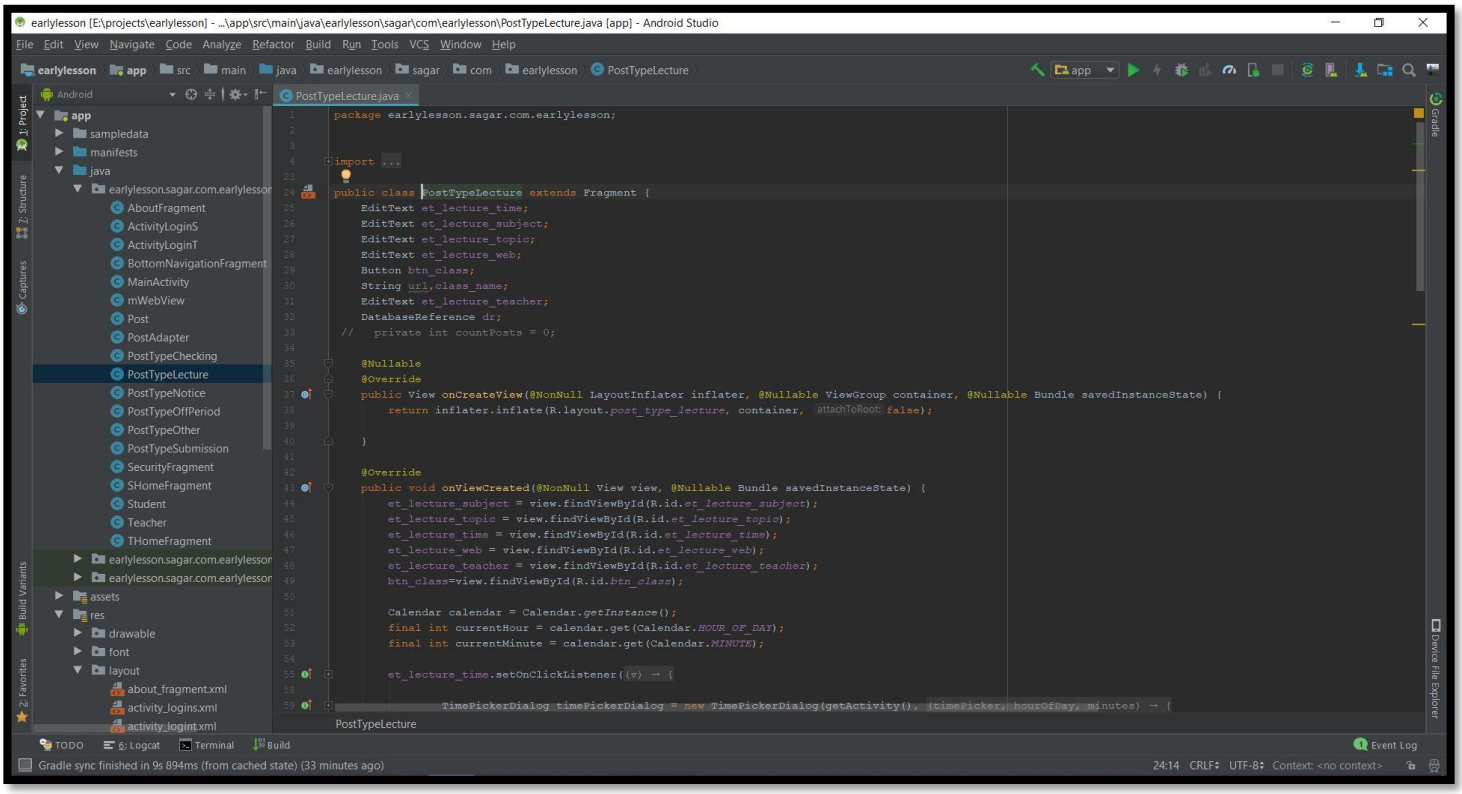
```java
package earlylesson.sagar.com.earlylesson;

import ...

public class SHomeFragment extends Fragment {

    private RecyclerView recyclerView;
    private RecyclerView.LayoutManager layoutManager;
    private PostAdapter postAdapter;

    FirebaseDatabase fd;
    DatabaseReference dr;
    ChildEventListener childEventListener;
    List<Post> datalist;
    public String mclass;

    public SHomeFragment() {}

    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nulla

        View v = inflater.inflate(R.layout.shome_fragment, container, attachToRoot: false);
        mclass = this.getArguments().getString( key: "data");

        return v;


    }

    @Override
    public void onViewCreated(@NonNull View view, @Nullable Bundle savedInstanceState) {
        datalist = new ArrayList<>();
        fd = FirebaseDatabase.getInstance();
        switch (mclass) {
            case "First Year":
                dr = fd.getReference( path: "Posts/First Year");
                break;
            case "Second Year":
```

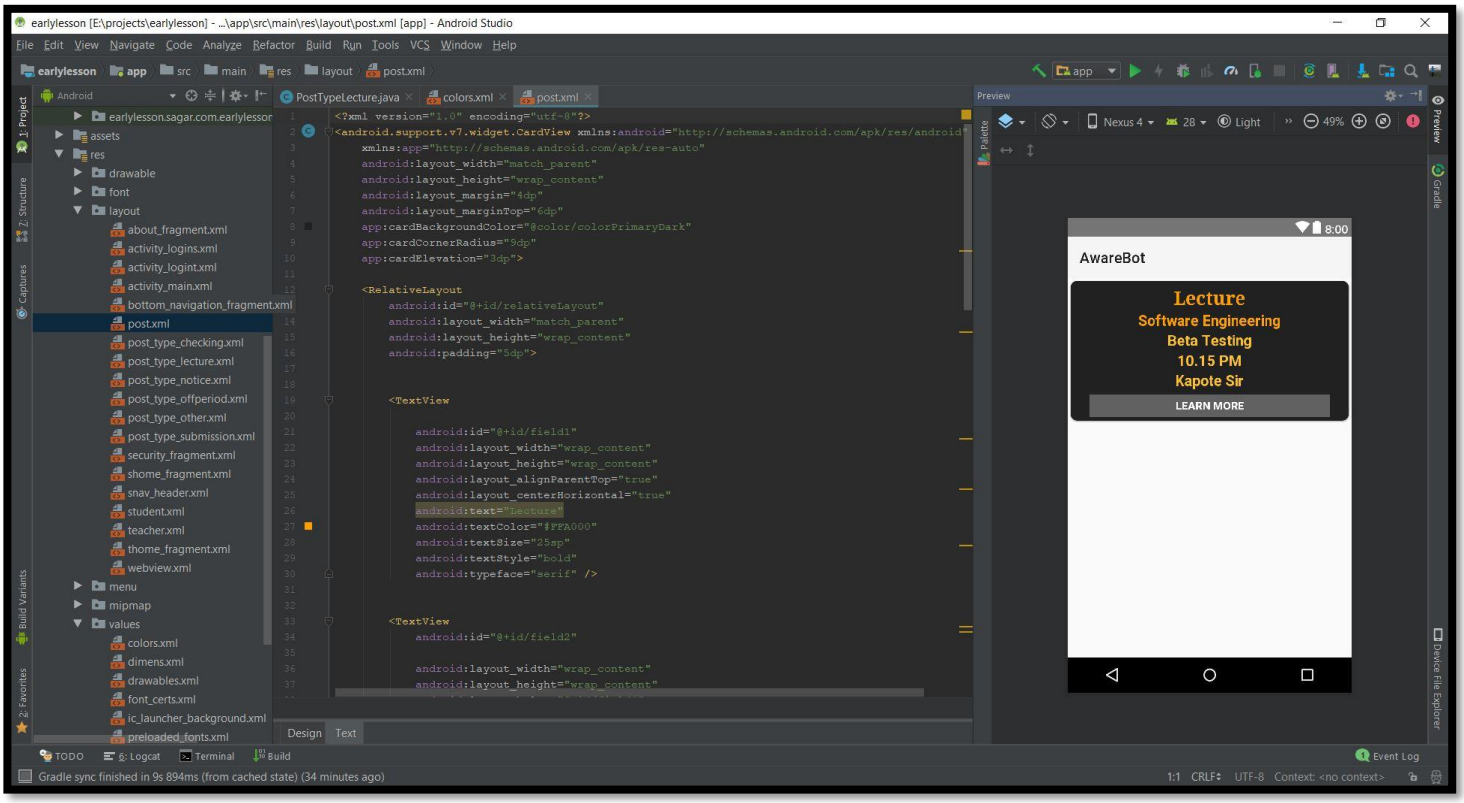
SHomeFragment

Respective xml file:

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">


    <android.support.v7.widget.RecyclerView

        android:id="@+id/recyclerview"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@color/colorPrimaryLight"
        android:padding="4dp"
        android:scrollbars="vertical">

    </android.support.v7.widget.RecyclerView>


</RelativeLayout>
```

App development were done using IDE -Android Studio:





CHAPTER 10: CONCLUSION AND FUTURE SCOPE

Conclusion:

So, we successfully developed our project "AwareBot" which can be used to increase productivity of students. Application and website is in completely usable form. We learned lot of things in development phase. We learned time management. We evaluated various risks and managed them. Our efforts got best results.

Future scope:

1) This application will be used in colleges.
2) Teachers will use this system to save their time and efforts.
3) Students will use this system to be aware at every minute.
4) Minor improvements such as push up notifications, posts deletion etc. will be available in later updates.

CHAPTER 11: REFERENCES

1) https://en.wikipedia.org/wiki/Feasibility_study
2) https://www.costengineering.eu/back-to-the-basics-what-is-project-cost-estimation
3) https://www.mindtools.com/pages/article/newPPM_01.htm
4) https://en.wikipedia.org/wiki/Project_architect
5) https://en.wikipedia.org/wiki/Work_breakdown_structure
6) https://www.projectmanager.com/blog/what-is-project-scheduling
7) https://en.wikipedia.org/wiki/Schedule_(project_management)
8) https://en.wikipedia.org/wiki/Data-flow_diagram
9) https://www.tutorialspoint.com/uml/uml_use_case_diagram.htm
10) http://softwaretestingfundamentals.com/test-case/
11) https://www.developers.android.com
12) https://www.wikipedia.com

CHAPTER 11: APPENDIX

1) **Android:** It is a mobile operating system developed by Google. It is based on a modified version of the Linux kernel and other open source software, and is designed primarily for touchscreen mobile devices such as smartphones and tablets.
2) **Firebase:** The Firebase Realtime Database is a cloud-hosted NoSQL database that lets you store and sync data between your users in realtime.
3) **ADB:** ADB, Android Debug Bridge, is a command-line utility included with Google's Android SDK.
4) **SDK:** it is a set of tools that can be used to develop software applications targeting a specific platform.
5) **XML:** Extensible Markup Language is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.