# PROJECT REPORT ON
# E-SLATE COLLABORATIVE APPLICATION

**Submitted by**


| | |
|---|---|
| **SAILAKSHMI PISUPATI** | **BE 5 11** |
| **MANASI S. AGTE** | **BE 5 13** |
| **SIDDHARTH BIDWALKAR** | **BE 5 22** |


**Under The Guidance of**


**Prof VAISHALI GHATE**
**SHAH AHND ANCHOR KUTCCHI ENGINEERING COLLEGE**

**Prof J.A. GOKHALE**
**INTELLYSYS APPLICATIONSSOLUTIONS PVT. LTD.**


**(INFORMATION TECHNOLOGY)**



**Shah & Anchor Kutchhi Engineering College,**
**Chembur,Mumbai-400 088,India**

# Acknowledgement

Mahavir Education Trust's

## Shah &Anchor Kutchhi Engineering College

No project is ever completed without the guidance of the experts who have already traced this path before and have become masters of it and as a result, our teachers. So, we would like to take this opportunity to thank all those individuals who have helped us in this project.

We thank the management staff for providing the necessary resources and helping us out whenever required. We are very grateful to our Principal **B. K. Lande** and our Head of Department **Prof. Swati Deshpande** for extending their help directly and indirectly through various channels in our project.

We express our deep gratitude to our internal project guide **Prof. Vaishali Ghate** for providing timely assistance to our queries and the guidance that they gave us owning to their experience in their field for many years.

A project will be irrelevant if it is built on outdated technologies and hence we would like to thank our external guide **Mr. Jayanthi Gokhale** for sharing with us his vast knowledge of industry trends and help us identify the daring technologies of the future.

We extend our sincere appreciation to all our professors from Shah and Anchor Kutchhi Engineering College for their valuable insights during the project.

Thanking you,

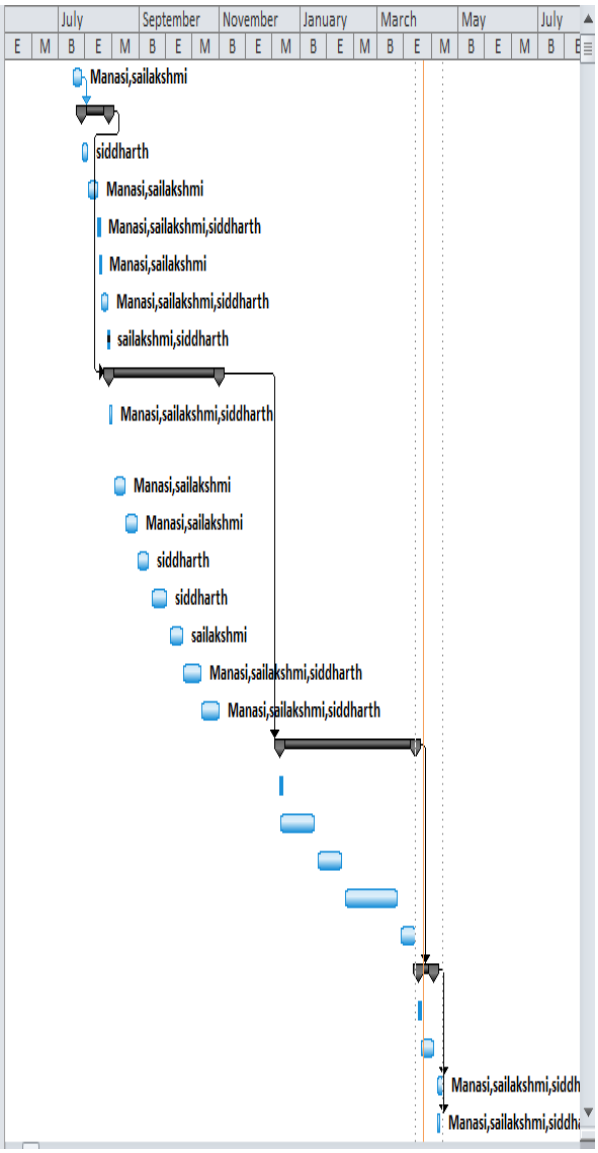
| | |
|---|---|
| Ms. Sailakshmi Pisupati | BE 5-11 |
| Ms. Manasi Agte | BE 5-13 |
| Mr. Siddharth Bidwalkar | BE 5-22 |

# Time & Task Management Plan

| Task Id | Milestone | Duration | Deliverable | Team members |
|---|---|---|---|---|
| **1** | Making of basic Project Abstract. | 1 week | Initial abstract report of project | Sailakshmi Pisupati<br>Manasi Agte<br>Siddharth Bidwalkar |
| **2** | Documentation of Project requirements and plan. | 2 weeks | Software Requirement specification | Sailakshmi Pisupati<br>Manasi Agte<br>Siddharth Bidwalkar |
| **3** | Whole designing of actual working of project using UML approach. | 2 months | UML diagrams including use case specifications, Activity ,class and sequence diagram | Sailakshmi Pisupati<br>Manasi Agte<br>Siddharth Bidwalkar |
| **4** | Actual project implementation. | 1 quarter | Basic working product with all functionalities | Sailakshmi Pisupati<br>Manasi Agte<br>Siddharth Bidwalkar |
| **5** | Test implemented product. | 10 days | Test plan and test results documents | Sailakshmi Pisupati<br>Manasi Agte<br>Siddharth Bidwalkar |
| **6** | Evaluation of product performance and delivery of final product. | 1 week | Final product | Sailakshmi Pisupati<br>Manasi Agte<br>Siddharth Bidwalkar |

# Gantt Chart for Time Management of the Project

| | Task Name | Duration | Start | Finish | Resource Names |
|---|---|---|---|---|---|
| 1 | PROJECT ABSTRACT | 5 days | Thu 12-07-12 | Wed 18-07-12 | Manasi,sailakshmi |
| 2 | **PROJECT PLANNING** | **15 days** | **Thu 19-07-12** | **Wed 08-08-12** | |
| 3 | Literature Survey | 3 days | Thu 19-07-12 | Mon 23-07-12 | siddharth |
| 4 | Requirement Gathering | 5 days | Tue 24-07-12 | Mon 30-07-12 | Manasi,sailakshmi |
| 5 | Requirement Analysis | 2 days | Tue 31-07-12 | Wed 01-08-12 | Manasi,sailakshmi, |
| 6 | Scheduling | 1 day | Thu 02-08-12 | Thu 02-08-12 | Manasi,sailakshmi |
| 7 | SRS documentation | 3 days | Fri 03-08-12 | Tue 07-08-12 | Manasi,sailakshmi, |
| 8 | Hardware and software I/F | 1 day | Wed 08-08-12 | Wed 08-08-12 | sailakshmi,siddhar |
| 9 | **PROJECT DESIGN** | **60 days** | **Thu 09-08-12** | **Wed 31-10-12** | |
| 10 | Obejct oriented approach ,Design Environment setup | 2 days | Thu 09-08-12 | Fri 10-08-12 | Manasi,sailakshmi, |
| 11 | Usecase Diagram design | 6 days | Mon 13-08-12 | Mon 20-08-12 | Manasi,sailakshmi |
| 12 | Activity Diagram design | 7 days | Tue 21-08-12 | Wed 29-08-12 | Manasi,sailakshmi |
| 13 | Usecase Realization | 7 days | Thu 30-08-12 | Fri 07-09-12 | siddharth |
| 14 | Sequence Diagram design | 10 days | Mon 10-09-12 | Fri 21-09-12 | siddharth |
| 15 | Class Diagram design | 8 days | Mon 24-09-12 | Wed 03-10-12 | sailakshmi |
| 16 | Gui and Database Design | 10 days | Thu 04-10-12 | Wed 17-10-12 | Manasi,sailakshmi, |
| 17 | System Interface Design | 10 days | Thu 18-10-12 | Wed 31-10-12 | Manasi,sailakshmi, |
| 18 | **PROJECT IMPLEMENTATION** | **75 days** | **Mon 17-12-12** | **Fri 29-03-13** | **Manasi,sailakshmi,** |
| 19 | Coding Environment setup | 1 day | Mon 17-12-12 | Mon 17-12-12 | |
| 20 | Gui Coding | 20 days | Mon 17-12-12 | Fri 11-01-13 | |
| 21 | Database Coding | 15 days | Mon 14-01-13 | Fri 01-02-13 | |
| 22 | Server side coding | 30 days | Mon 04-02-13 | Fri 15-03-13 | |
| 23 | Final Integration | 10 days | Mon 18-03-13 | Fri 29-03-13 | |
| 24 | **PROJECT TESTING** | **10 days** | **Mon 01-04-13** | **Fri 12-04-13** | **Manasi,sailakshmi,** |
| 25 | Black Box Testing | 2 days | Mon 01-04-13 | Tue 02-04-13 | |
| 26 | White Box testing | 8 days | Wed 03-04-13 | Fri 12-04-13 | |
| 27 | Performance Evaluation | 5 days | Mon 15-04-13 | Fri 19-04-13 | Manasi,sailakshmi, |
| 28 | Release | 2 days | Mon 15-04-13 | Tue 16-04-13 | Manasi,sailakshmi, |



# Abstract of project

This project is an attempt to use the Android Platform for Virtual Classroom application as it is needed by various teaching and coaching institutes to make learning faster, better & easier. This application is a collaborative tool which is modeled on MVC (Model – View – Controller) architecture. It has a Content Delivery Engine, a Backend Database and a User Interface for different clients. There are different roles in user that are student, teacher administrator, content provider and head of department.

It is real time application for the classroom sessions where in the classroom the teachers and students get connected to the wireless network through their devices and can make use of this application for learning purpose. Each user is provided with different features depending on his/her role in the system. The E-Slate Collaborative Application works in the wireless environment. It is designed to connect multiple tablets to a Content Delivery Engine via a host network through an Access Point provided in every classroom. The application will help to deliver the teaching lessons to the student's tablets and also help in creating an interactive classroom session that includes Quiz, Ask-a-Query. This application helps in maintaining attendance records of students and teachers and the academic performance of the students. This application also helps in constructing, conducting and reviewing the tests and assignments. This tool does not aim to use or allow internet facilities in classroom, in short no web access during the classroom sessions. This E-Slate Collaborative Tool is a self-contained, independent software product.

## Table of Contents

**LIST OF FIGURES**

# CHAPTER 1

# INTRODUCTION

# 1. Introduction

The motivation for the development of this software is that to improve the methods adopted in teaching environment to make classroom sessions more interactive and user-friendly.

## 1.1 Objective

This product is intended for users of the teaching and coaching environment. This product can be used in a larger classroom environment where the traditional blackboard teaching cannot suffice the large audience

## 1.2 Methodology Used

E-slate Collaborative Application is developed using Object Oriented Approach in Software Engineering. According to this approach the initial software requirement documents and the project scheduling is carried out as a part of planning with the use of Microsoft Office and Microsoft Project for the purpose. The application framework design phase of development is designing of the software application, where the UML (Unified Modelling Language) is used for developing the UML diagrams with the use of IBM Rational Rose.

In Coding phase Android operating system was used for Graphical User Interface development. Tablets with android platforms, Eclipse – Software development tool with Android SDK (Software Development Kit) and AVD (Android Virtual Device) emulators have been used to build, and run the application.

The development of the Content Delivery Engine has been done with the use of XAMPP server, which stores the databases and also for the purpose of php scripting for running the application.

Routers (Wi-Fi hotspot), tablets and laptops are hardware used.

## 1.3 Problem Statement

This project is an attempt to use the Android Platform for Virtual Classroom application as it is needed by various teaching and coaching institutes to make learning faster, better & easier. This application is a collaborative tool which is modeled on MVC (Model – View – Controller) architecture. It has a Content Delivery Engine, a Backend Database and a User Interface for different clients. All interactions are controlled by the Content Delivery Engine depending on the role of the user (Student, Teacher, Admin, Content Provider, and HOD). The backend database contains all the needed data. The content is delivered using the content delivery engine, forming the middle layer.

Each user has to login to the system using his/her username and password. Each user will be classified based on the Roles he/she belongs to. Each role has certain common features and special features.

Admin has the authority to handle all the other clients, it is the admin who will manage the system and bring about the needed changes.

Class Mode is main Classroom Environment for student and teacher.

The Teachers can opt for his/her subject, lesson. They can construct, conduct and review test. They can synchronize and view the activities of all the other students in their classroom. They can share their notes and subject related data with the clients. They can create group of students who can work upon on a given assignment or project together.

Students can give test and view its results. Students cannot share data at the time of test and on timings restricted by the teacher. The Content Provider is responsible for adding notes and other resources of project.

# CHAPTER 2

# LITERATURE SURVEY

# 2. Literature Survey

## 2.1 Tablet in classroom

With different types of technologies kids are exposed to today, teachers are trying to keep up with their classrooms. Many teachers have used laptops, iPods, iPads and numerous software applications to keep students engaged in their lessons.

A few of the newest additions to this technological line-up are Android devices. These devices, like tablets and e-Readers, are available for use in education, offering great benefits for teachers as a way to better educate their students.

Android tablets can be used to give students the web at their fingertips. Imagine pulling out the tablets and showing students a multidimensional look at the human skeleton during a science lesson. Not only are teachers explaining lessons now, but they can literally show students examples with these tablets.

Teachers will spend less time lecturing, which will give students more time to be actively engaged in their learning experience. One of the greatest features of these tablets is the capability to download educational apps to use on them. These apps are many and diverse, ranging from math and reading games to an on-hand dictionary. Apps are a great way to expand the learning experience for students.

With Android tablets, students can research right at their desk or even watch a relevant curriculum video. The best part is that students can do this at their own pace. Teachers will find themselves with fewer students falling behind and fewer students bored as they wait for the lesson to progress. The benefits of having web-accessible tablets in the classroom are endless.

Android e- readers would be advantageous in any curriculum. The days of crowding a classroom with bulky books are over with these inexpensive e-Readers. Students can use them for in-class reading assignments or just for something for them to do if they finish their work early.

Android e-Readers give teachers a great sponge activity that requires very little fuss; plus, students will love working with the new technology, giving them another incentive to read in

the classroom. Many Android e-Readers also have text-to-speech capability, giving those auditory learners a boost in classrooms that cater to the many visual learners.

For many students, school is the only place where they can access such technology. It opens up a whole new world for these kids. Allowing students to access different technologies in the classroom give them skills they will take out in the real world.

They learn how to interact with the different devices, how to successfully explore the Internet and how to use the new applications that are available on these devices. Android devices can enhance learning experiences because students are given an in-depth look into their subject area, not just a day or two into textbook reading and worksheets.

The utilization of Android devices in the classroom is just the sort of change to current lesson plans that teachers need to take the education of their students to the next level. Additionally, Android devices are some of the least expensive products of their kind on the market today. Their use can stretch both inside and outside the classroom, providing great help and motivation when revision is needed. (Classroom Management Software)

## 2.2 Interactive wireless tablets

An increasingly popular accessory or alternative to the interactive classroom blackboard is the interactive wireless tablet. These tablets generally connect via wireless Bluetooth technology. These pads can act as an extension to the board or act independently of them.

When an interactive wireless tablet is used with an interactive whiteboard:

- The teacher can have the tablet in hand while student uses the interactive whiteboard
- Allows the teacher to make corrections and control the student's activity on the interactive whiteboard.
- Excellent tool for special needs students

When an interactive wireless tablet is used alone:

- Functions are the same as a whiteboard
- Portable size
- Allows added flexibility in the size of projected image
- No shadows cast on screen by user
- Less expensive alternative to blackboard

Student response systems have taken the world of quizzing, reviewing and test taking to an entirely new level and have strengthened the impact of the interactive classroom. No longer will one have one or two students be the only students who answer questions during a review session.

## 2.3 Goclass application

A similar classroom application existing today is GoClass application on iPad.

GoClass is a cloud enabled teaching application for tablet devices that redefines the boundaries of computing in the classroom. Connect with your students like never before, customize and fine-tune your lesson plans on the fly, engage students in new ways and continuously evaluate their understanding while you are in class.

**Features for teachers**

**Lesson plan**

Instructors can string together resources in a lesson plan format along with lecture notes and questions for formative assessment through a single app, and then deliver it through the same app. What's more, these lesson plans can be created and modified during class session. The Lesson Plan follows a simple syntax of SHOW-EXPLAIN-ASK. The SHOW aspect allows instructor to add media resources required for the lesson. EXPLAIN allows the instructor to create lecture notes in relation to the key concepts of the lesson. It enables the instructor to summarize the concept in simple bulleted list. EXPLAIN can also be used to is to list out and project homework assignments or give topics and directions for in-class pair or group activities that students can refer to while working on the activity. ASK allows the instructor to create questions that can be administered in class while explaining the lesson concepts. The app gives instructor instant results so that instructor can adjust his/her teaching focus in real time.

**Session settings**

When using the lesson for conducting a classroom session, the instructor can define the settings for that particular session.

- Flipback - Allows students to browse teacher broadcasts while in the session
- Bookmark - Bookmarked items may be viewed after session
- Sleep Mode - Instructor can control app screen for both instructor and student iPads

- Communication Mode- Instructor can switch from default cloud communication to WiFi mode

**Project**

Project feature can be used when device is connected to an external projector. It allows instructor to project videos, images, documents, questions, lecture notes. It also allows realtime projection of scribbles, annotations and results of formative assessments.

**Broadcast**

During a classroom session this feature allows instructor to share resources (all resources except Video) and ask questions which are pre-authored in the lesson plans with students. The answers that the students submit can be viewed in real time.

**Scribble**

An instructor can now annotate on images and explanations in free hand form (or using a capacitive stylus). Additional templates are available for a variety of scenarios. Annotations reflect across projected media as well as broadcasted media. Instructor can use this feature in SHOW to annotate on images and in EXPLAIN to scribble, annotate and draw.

**Polls**

Instructor can create an instant poll to get students opinion and get them involved in active learning rather than passively listening to the lecture. This feature can also be used for formative assessment and use the results to pace instruction accordingly. Poll results along with results of any questions asked in class can be reviewed in the Reports and Logs section of the web app.

**Student list**

This feature allows instructor to keep a tab on which students are present in the classroom session and whether they are logged in. This indicates whether students are cued in to what the instructor is sharing with them or teaching them.

**Features for students**

**Join session**

Once the instructor starts a class session using GoClass and selects the relevant student roster, students in that class can join the class session through the App on their individual devices.

**View broadcasts**

Students can view all the resources that the instructor broadcasts. They can also view the annotations and made on a resource in real time when the instructor has already broadcasted the resource.

**Bookmark**

Students can bookmark resources shared(broadcast) by the instructor. These bookmarked items will be available for students to view during and after session.

**Interact & respond**

Once the instructor shares reading or viewing materials with the students, the students can interact with the content using all the touch interactions such as zoom, pinch, pan. Students can also respond to questions asked during class through the GoClass App on their device.

The idea to use the tools to develop our application.

During the requirement analysis phase, we searched various forums online to find the best tool and IDE to develop our application. During our study work, we came across Eclipse IDE to be the best to develop Android application, as it contains the Android SDK and AVD (to be downloaded separately or from the IDE.).

Also the idea to use the XAMPP server came across when we attended an Apache-MySQL-PHP workshop, as we were supposed to implement a client server model and processing of Java request (i.e. Android-client side) is easier and friendlier in PHP environment.

# CHAPTER 3

# PROBLEM STATEMENT

# 3. Problem Statement

This project is an attempt to use the Android Platform for Virtual Classroom application as it is needed by various teaching and coaching institutes to make learning faster, better & easier. This application is a collaborative tool which is modeled on MVC (Model – View – Controller) architecture. It has a Content Delivery Engine, a Backend Database and a User Interface for different clients. All interactions are controlled by the Content Delivery Engine depending on the role of the user (Student, Teacher, Admin, Content Provider, and HOD). The backend database contains all the needed data. The content is delivered using the content delivery engine, forming the middle layer.

Each user has to login to the system using his/her username and password. Each user will be classified based on the Roles he/she belongs to. Each role has certain common features and special features.

Admin has the authority to handle all the other clients, it is the admin who will manage the system and bring about the needed changes.

Class Mode is main Classroom Environment for student and teacher.

The Teachers can opt for his/her subject, lesson. They can construct, conduct and review test. They can synchronize and view the activities of all the other students in their classroom. They can share their notes and subject related data with the clients. They can create group of students who can work upon on a given assignment or project together.

Students can give test and view its results. Students cannot share data at the time of test and on timings restricted by the teacher. The Content Provider is responsible for adding notes and other resources of project.

# CHAPTER 4

# REQUIREMENT ANALYSIS

# 4. Requirement Analysis

## 4.1 SRS Document

The Software Requirements Specification (SRS) is a formal statement of the application functional and operational requirements. It serves as a contract between the developer and the customer for whom the system is being developed. The developers agree to provide the capabilities specified. The client agrees to find the product satisfactory if it provides the capabilities specified in the SRS.

### 4.1.1 Introduction

#### 4.1.1.1 Purpose

The purpose of Software Requirement Specification (SRS) is to present detailed description of E-slate collaborative application on which the project team is going to work. It also states the various required constraints by which the system will abide.

#### 4.1.1.2 Scope

The E-Slate Collaborative Application works in the wireless environment. It is designed to connect multiple tablets to a Content Delivery Engine via a host network through an Access Point provided in every classroom. This system can be employed in schools and coaching classes.

#### 4.1.1.3 Intended Audience

This product is intended for users of the teaching and coaching environment. This product can be used in a larger classroom environment where the traditional blackboard teaching cannot suffice the large audience.

#### 4.1.1.4 Definition, Acronyms & Abbreviation

#### 4.1.1.5 References


i) http://en.wikipedia.org/wiki/Software_Requirements_Specification

ii) www.w3schools.com

iii) www.android.com

iv) http://php.net/manual/en/book.mysql.php

## 4.1.2 Overall Description

### 4.1.2.1 Product Perspective

This E-Slate Collaborative Tool is a self-contained, independent software product. It is designed to work in a wireless environment, where the tablets get connected to the Access Points provided in each classroom. The Access Points are in turn connected to the main Content Delivery Engine that manages this system. Content is retrieved from Content Delivery Engine and displayed on screens of tablets connected in network.

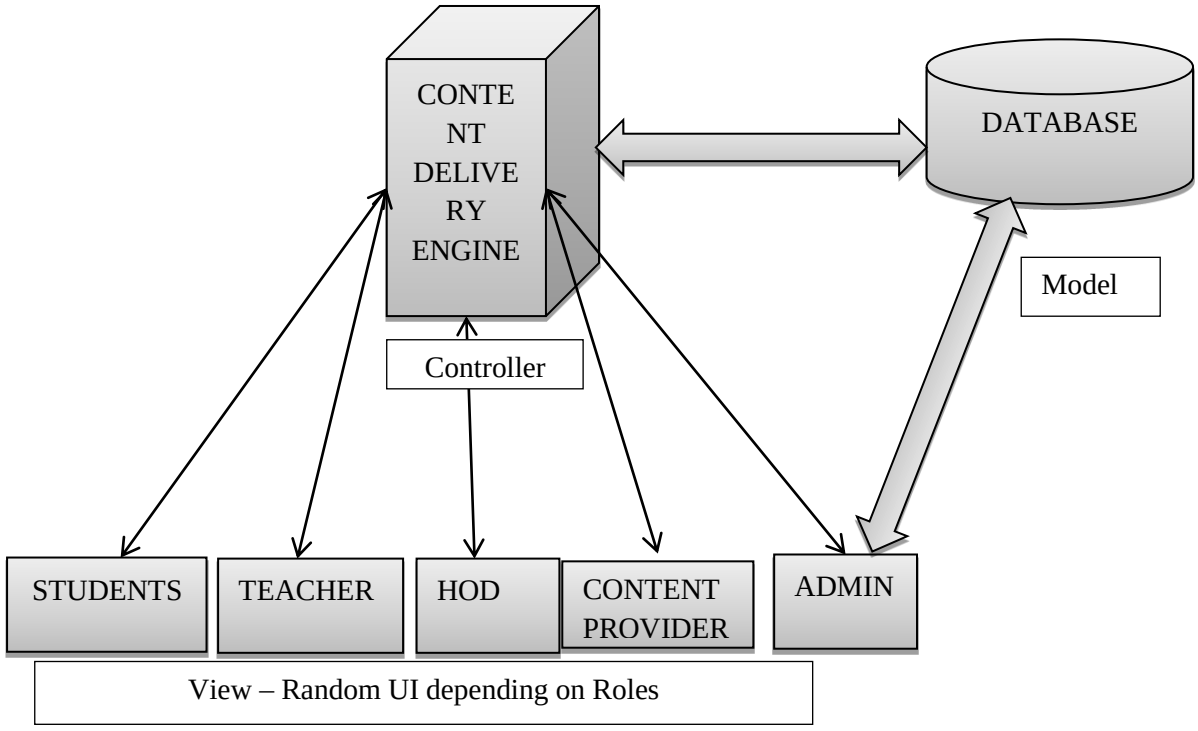



**Figure 1: System Block Diagram**

### 4.1.2.2 Product Functions

This E-Slate Tool helps connect the tablets of teachers (the host) and the students (the clients) to content delivery engine placed in the institution via access points provided in each classroom.

A login facility is provided for enabling only authorized users to access the system.

The users have to start the application installed in their tablets and login to the system. Once logged in, the system will automatically make a record of the user, thereby handling the attendance of the teacher or student and also creating a session between the host and the

clients connected to an Access Point. Each session can have three inter-related modes of classroom teaching which includes Lecture hour, Quiz session and Ask-a-Query.

During the lecture hour the host (teacher) can retrieve the required teaching material from the Content Delivery Engine and load the same to the host as well as client's tablets. The host handles the client's tablets and all the contents that need to be delivered to their tablets.

During the Quiz session, the host can dynamically ask questions to the students to get a better idea of their understanding; the host will get the replies of each student in a personalized thumbnail view of each student.

During the Ask-a-Query session, the client can ask their queries to the teachers via message box provided in application. The host is alerted by blinking of thumbnail of student.

This tool also provides the function of Conducting Test and assigning various Assignments to the users. The Test will be time-based tests and the Assignments will have submission dates when the students will have to load it to the Content Delivery Engine, which will also make a note of the upload date and time.

The Administrator will handle the authorization of the Various Roles in this system. The Content Provider will be handling the content in the content delivery engine. The role of Content Provider can be taken by teachers also.

#### 4.1.2.3 User Classes and Characteristics

The Intended user of the product can be a student or a learner with knowledge of e-learning using his Tablets. Such user will find the application useful.

User can be a teacher or a tutor. She/he can use the product for taking classes in the particular network with number of clients. This application can be used in a larger classroom environment where the traditional blackboard teaching cannot suffice the large audience. The Tutor should be comfortable in handling the tablet devices. Users can also access the saved files (documents and media files) in an offline mode.

#### 4.1.2.4 Operating Environment

This system is designed to work in classrooms in a wireless environment. The users can also uses the application in an offline mode when they can access the files saved during sessions.

#### 4.1.2.5 Design and Implementation Constraints

The application will have to implement a security policy to safeguard the information on administrator's System from being modified by unauthorized users.

Connecting the users i.e., clients tablet and host tablet with the administrator's module is a very difficult job which must be done accurately.

Each and every System on the network must be assigned a unique identifier so as to avoid any ambiguity.

#### 4.1.2.6 Assumptions and Dependencies

It is assumed that for the application to work,

- The users are having devices with specified hardware and software requirements for the use of application.
- Wireless network with the required protocols is required for connectivity between the application clients. All the clients are within the range of network.
- Every class must have access point which is connected to Content Delivery Engine.

### 4.1.3 External Interface Requirement

#### 4.1.3.1 User Interface

The users will have to use this tool using the Android Tablets. The host and the client will be having different user interfaces.
The following screens will be provided.

- Login Screen:

This will be the first screen that will be displayed. It will allow user to access different screens based upon the user's role.
Various fields available in this screen will be,
User ID: Alphanumeric up to 8 characters
Password: Alphanumeric up to 8 characters
Role: Will have the following values
Administrator, Head of Department, Content Provider, Teacher, Student.

- Teaching scenario screen:

Here, the large part of screen is dedicated to display educational content from Content Delivery Engine. In teacher's tablet, functions are provided to retrieve the data from Content Delivery Engine. Data can be text or multimedia file. When teacher selects file from server, the file is simultaneously loaded to host as well as all client tablets. Above procedure will again be repeated if teacher wants to change the file.

- Ask-a-Query screen:

Here, a message box is provided on client screens. When student wants to ask a doubt to teacher, he simply types his doubt and sends it to teacher. Whenever student asks a doubt, thumbnail of that student will change color as sign of notification. Then, teacher can answer doubt of that particular student using message box. Teacher can also use message box to send message to more than one student at a time.

- Quiz screen:

This scenario is designed to ensure that students have understood the topic that has been taught to them. Here, teacher sends a question to all students at once. Students answer that question and teacher gets notification on screen about all student answers. Teacher then evaluates the answer.

- Assignment Screen:

Here, teacher loads the assignment document from server to all client tablets. Students download the document to local memory of their respective tablets. They solve assignment at home and load answer document on server on the day of submission. Assignment is evaluated in server.

- Test Screen:

Here, teacher loads the test document from server to all client tablets. Teacher starts timer for test. Students solve test in stipulated time and load answer document on server. Test is evaluated in server.

### 4.1.3.2 Hardware Interfaces

- Tablet with the following features: ARM cortex based processor, Android OS, min 16 GB of secondary memory.
- The product can even work on the laptops having android tablet emulator.
- Wi-Fi Router

### 4.1.3.3 Software Interfaces

- **Eclipse – Software development tool to build the application.**

In computer programming, **Eclipse** is a multi-language software development environment comprising a base workspace and an extensible plug-in system for customizing the environment. It is written mostly in Java. It can be used to develop applications in Java and, by means of various plug-ins, other programming languages including Ada, C, C++, COBOL, Fortran, Haskell, JavaScript, Perl, PHP, Python, R, Ruby (including Ruby on Rails framework), Scala, Clojure, Groovy, Scheme, and Erlang. It can also be used to develop packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++ and Eclipse PDT for PHP, among others. (Wikipedia/Eclipse)

The installation of Eclipse IDE is above the scope of this document. For the purpose of reference please refer the following link. (Installation)

- **ADT Plugins**

**Download the ADT Plugin**

---

1. Start Eclipse, then select **Help** > **Install New Software**.
2. Click **Add**, in the top-right corner.

   In the Add Repository dialog that appears, enter "ADT Plugin" for the *Name* and the following URL for the *Location*:

   https://dl-ssl.google.com/android/eclipse/

3. Click **OK**.

   If you have trouble acquiring the plugin, try using "http" in the Location URL, instead of "https" (https is preferred for security reasons).

4. In the Available Software dialog, select the checkbox next to Developer Tools and click **Next**.

5. In the next window, you'll see a list of the tools to be downloaded. Click **Next**.
6. Read and accept the license agreements, then click **Finish**.

   If you get a security warning saying that the authenticity or validity of the software can't be established, click **OK**.

7. When the installation completes, restart Eclipse.

   For errors please refer. (Download the ADT Plugin)

- **Android SDK and AVD (Android Virtual Device) to run and test the application.**

**Adding Platforms and Packages**

The Android SDK separates tools, platforms, and other components into packages you can download using the Android SDK Manager. The original SDK package you've downloaded includes only the SDK Tools. To develop an Android app, you also need to download at least one Android platform and the latest SDK Platform-tools.

**Launch the SDK Manager.**

If you've used the Windows installer to install the SDK tools, you should already have the Android SDK Manager open. Otherwise, you can launch the Android SDK Manager in one of the following ways:

- On Windows, double-click the SDK Manager.exe file at the root of the Android SDK directory.
- On Mac or Linux, open a terminal and navigate to the tools/ directory in the Android SDK, then execute android sdk.

The SDK Manager shows all the SDK packages available for you to add to your Android SDK. As a minimum configuration for your SDK, we recommend you install the following:

- The latest Tools packages (check the **Tools** folder).
- The latest version of Android (check the first **Android** folder).
- The Android Support Library (open the **Extras** folder and check **Android Support Library**).

Once you've chosen your packages, click **Install**. The Android SDK Manager installs the selected packages into your Android SDK environment.

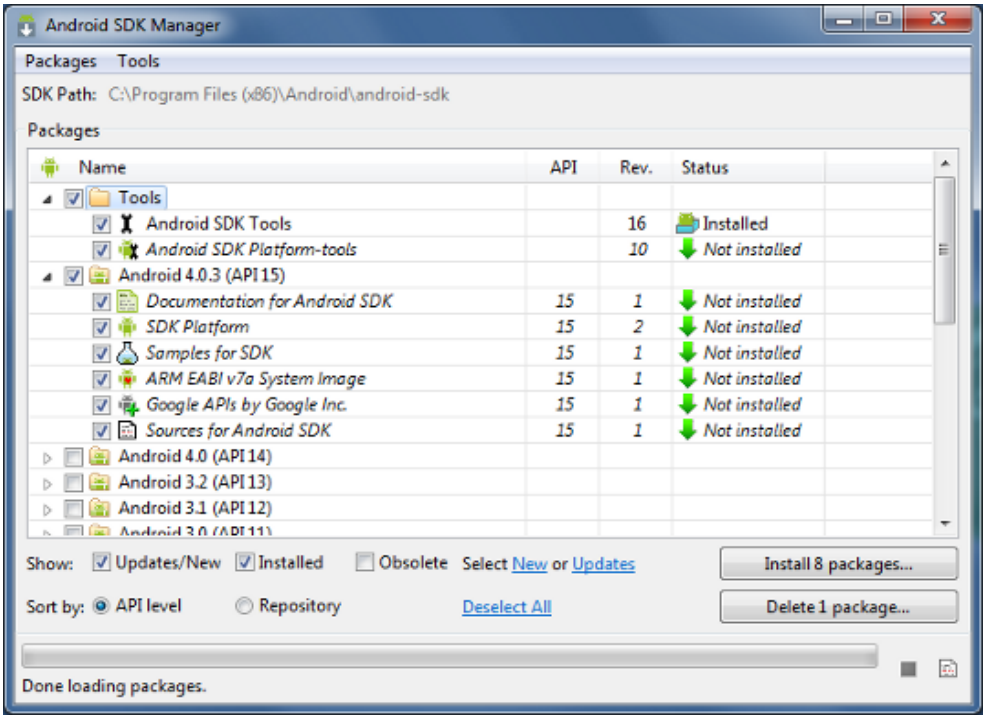



**Figure 2: The Android SDK Manager shows the SDK packages that are available, already installed, or for which an update is available.**

- XAMPP server for the content delivery engine and backend database. (apache friends - xampp for windows)
    - XAMPP is a compilation of free software (comparable to a Linux distribution), it's free of charge and it's free to copy under the terms of the GNU General Public License. But it is only the compilation of XAMPP that is published under GPL. Please check every single license of the contained products to get an overview of what is, and what isn't, allowed.
    - In the case of commercial use please take a look at the product licenses (especially MySQL), from the XAMPP point of view commercial use is also free.

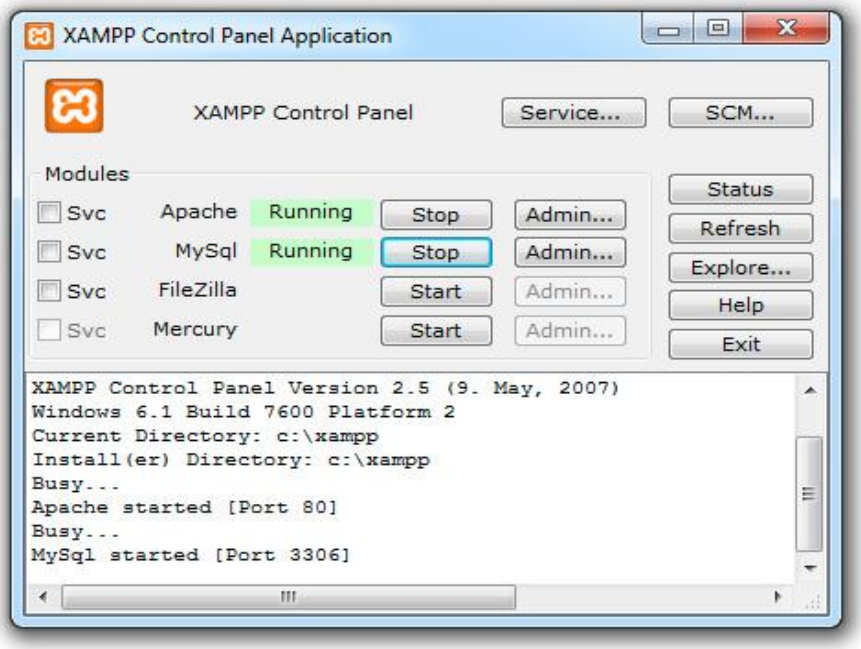



**Figure 3: The XAMPP control panel for start/stop Apache, MySQL, FileZilla & Mercury.**

### 4.1.4 System Features

#### 4.1.4.1 Login

4.1.4.1.1 Brief Description

This use case describes how a user logs into the e-slate application.

4.1.4.1.2 Flow of events

This use case states when the user wishes to login to the application.

- System requests user to enter username, role and password.
- The user enters his/her username, role and password.
- The system validates the entered username and password and logs the user into the system.

4.1.4.1.3 Alternative flow

Invalid username or password

In the basic flow if the user enters an invalid name and password, the system displays an error message.

The user can either choose to return to the beginning of the basic flow or cancel login.

4.1.4.1.4 Special Requirement

None

4.1.4.1.5 Preconditions

None

4.1.4.1.6 Post Conditions

If the use case is successfully executed the user logs into the system and if not the system state is unchanged.

4.1.4.1.7 Extension Points
None

### 4.1.4.2 Retrieve File

4.1.4.2.1 Brief Description
This use case describes how a teacher can retrieve files that are preloaded from the content delivery engine.

4.1.4.2.2 Flow of events

4.1.4.2.2.1 Basic Flow
This use case states when teacher wants to retrieve from Content Delivery Engine,

- Teacher searches the file in the CDE from which she wants to teach
- When teacher finds that file, then she loads it in teacher's as well as all student tablets.

4.1.4.2.2.2 Alternative flow
None

4.1.4.2.3 Special Requirement
None

4.1.4.2.4 Pre-condition
Teacher and students should be logged in the system. The file which is to be retrieved must be present in the system.

4.1.4.2.5 Post-condition
File can be simultaneously viewed on teacher's as well as all student tablets. Hence, lecture session starts.

4.1.4.2.6 Extension points
None

### 4.1.4.3 Conduct test

4.1.4.3.1 Brief Description
This use case describes how teacher conducts a test in the class

4.1.4.3.2 Flow of Events

4.1.4.3.2.1 Basic flow
When teacher wishes to conduct a test,

- She retrieves test papers from Content Delivery Engine
- When teacher finds a test paper for particular topic, she load test paper on student tablets and starts test timer.
- When timer runs out, student load marked test papers back to CDE.

4.1.4.3.2.2 Alternative Flow
None

4.1.4.3.3 Special Requirements
None

4.1.4.3.4 Pre-condition
Teacher and students should be logged in the system. The test paper(file) which is to be retrieved must be present in the system.

4.1.4.3.5 Post-Condition
Test paper can be simultaneously viewed on teacher's as well as all student tablets. Hence, students can begin the test.

4.1.4.3.6 Extension Points
None

### 4.1.4.4 Ask Quiz

4.1.4.4.1 Brief Description
When teacher finishes teaching a topic to class, she can conduct a quiz to see if students have understood the topic. This use case describes how teacher conducts a random quiz in the class

4.1.4.4.2 Flow of Events

4.1.4.4.2.1 Basic flow
When teacher wishes to conduct a quiz,

- Teacher transmits a question to all student tablets and waits for an answer.
- The thumbnail of student selecting correct answer will turn green.

4.1.4.4.2.2 Alternative Flow
None

4.1.4.4.3 Special Requirements
None

4.1.4.4.4 Pre-condition
Teacher and students should be logged in the system.

4.1.4.4.5 Post-Condition
Question can be simultaneously viewed on teachers as well as all student tablets. Hence, students can answer the question.

4.1.4.4.6 Extension Points
None

### 4.1.4.5 Assign Assignment

4.1.4.5.1 Brief Description
When teacher finishes teaching a topic to class, he/she can assign assignments to the student on that particular topic.

4.1.4.5.2 Flow of Events

4.1.4.5.2.1 Basic flow
When teacher wishes to assign an assignment,

- Teacher transmits an assignment to all student tablets or to particular group of students.

4.1.4.5.2.2 Alternative Flow
None

4.1.4.5.3 Special Requirements
None

4.1.4.5.4 Pre-condition
Teacher and students should be logged in the system. The assignment has to be present in the content delivery engine.

4.1.4.5.5 Post-Condition
None

4.1.4.5.6 Extension Points
None

### 4.1.4.6 Answer Query

4.1.4.6.1 Brief Description
When a student asks a query in the classroom session, the teacher can reply to it.

4.1.4.6.2 Flow of Events

4.1.4.6.2.1 Basic flow
When teacher wishes to answer a query,

- Teacher can answer this query to that particular student or can collectively all students.

4.1.4.6.2.2 Alternative Flow
None

4.1.4.6.3 Special Requirements
None

4.1.4.6.4 Pre-condition
Teacher and students should be logged in the system. The student must have asked a query to the teacher.

4.1.4.6.5 Post-Condition

4.1.4.6.6 Extension Points
None

### 4.1.4.7 Ask Query

4.1.4.7.1 Brief Description
When the teacher is conducting the classroom sessions, the student can ask queries related to the topic being though.

4.1.4.7.2 Flow of Events

4.1.4.7.2.1 Basic flow
When student wants to ask a query,

- He types his query in message box of his tablet
- He sends query to teacher and awaits for answer.

4.1.4.7.2.2 Alternative Flow
None

4.1.4.7.3 Special Requirements
None

4.1.4.7.4 Pre-condition
Teacher and students should be logged in the system.

4.1.4.7.5 Post-Condition
Question can be viewed on teacher's tablet. Hence, teacher can answer the query.

4.1.4.7.6 Extension Points
None

### 4.1.4.8 Submit Assignment

4.1.4.8.1 Brief Description
When student finishes answering questions in assignment, he can submit it back to system. This use case describes the procedure to do so.

4.1.4.8.2 Flow of Events

4.1.4.8.2.1 Basic flow
When student wants to submit an assignment,

- He selects completed assignment file in his tablet.
- He loads assignment file in Content Delivery Engine.

4.1.4.8.2.2 Alternative Flow
None

4.1.4.8.3 Special Requirements
None

4.1.4.8.4 Pre-condition
Student should be logged in the system.

4.1.4.8.5 Post-Condition
Assignment is evaluated in Content Delivery Engine.

4.1.4.8.6 Extension Points
None

### 4.1.4.9 Answer Test

4.1.4.9.1 Brief Description
This use case describes how student can answer the test and submit it to the system.

4.1.4.9.2 Flow of Events

4.1.4.9.2.1 Basic flow
When student receives a test,

- He answers all the questions before timer runs out.
- When student finishes the test, he can load the file in Content Delivery Engine.

4.1.4.9.2.2 Alternative Flow
None

4.1.4.9.3 Special Requirements
None

4.1.4.9.4 Pre-condition
Teacher and Students should be logged in the system.

4.1.4.9.5 Post-Condition
Test is evaluated in Content Delivery Engine.

4.1.4.9.6 Extension Points
None

### 4.1.4.10 Answer Quiz

4.1.4.10.1 Brief Description
This use case describes how student can answer the quiz question.

4.1.4.10.2 Flow of Events

4.1.4.10.2.1 Basic flow
When student receives a quiz question, he/she marks right option among given options.

4.1.4.10.2.2 Alternative Flow
None

4.1.4.10.3 Special Requirements
None

4.1.4.10.4 Pre-condition
Student and teacher must be logged in system.

4.1.4.10.5 Post-Condition
Teacher can wish to ask another question or stop the session.

4.1.4.10.6 Extension Points
None

### 4.1.4.11 Check Statistics

4.1.4.11.1 Brief Description
This use case describes how Head of Department can check daily/weekly statistics about lectures held, topics taught, students present, etc.

4.1.4.11.2 Flow of Events

4.1.4.11.2.1 Basic flow
When Head of Department wishes to check statistics,

- She can check table about lectures conducted on day or week.
- She can check table about topics taught in week.
- She can check student attendance for week.

4.1.4.11.2.2 Alternative Flow
None

4.1.4.11.3 Special Requirements
None

4.1.4.11.4 Pre-condition
Head of Department must be logged in system.

4.1.4.11.5 Post-Condition
None

4.1.4.11.6 Extension Points
None

### 4.1.4.12 Add User

4.1.4.12.1 Brief Description
This Use case describes how System Administrator adds users in the system.

4.1.4.12.2 Flow of Events

4.1.4.12.2.1 Basic flow
When administrator wishes to add users in the existing system,

- He fills personal details of user in system database.
- He grants role (teacher/student/HOD/Content Provider) as per requested by user and gives rights to him accordingly.

4.1.4.12.2.2 Alternative Flow
None

4.1.4.12.3 Special Requirements
None

4.1.4.12.4 Pre-condition
Person must submit personal details to administrator.

4.1.4.12.5 Post-Condition
None

4.1.4.12.6 Extension Points
None

### 4.1.4.13 Delete User

4.1.4.13.1 Brief Description
This Use case describes how System Administrator deletes users in the system.

4.1.4.13.2 Flow of Events

4.1.4.13.2.1 Basic flow
When administrator wishes to delete users from the existing system,

- He revokes rights granted to user
- He erases personal data of user from database.

4.1.4.13.2.2 Alternative Flow
None

4.1.4.13.3 Special Requirements
None

4.1.4.13.4 Pre-condition
User must be registered in system.

4.1.4.13.5 Post-Condition
None

4.1.4.13.6 Extension Points
None

### 4.1.4.14 Add Files

4.1.4.14.1 Brief Description
This Use case describes how Content Provider adds files in the Content Delivery Engine.

4.1.4.14.2 Flow of Events

4.1.4.14.2.1 Basic flow
When administrator wishes to add files in the existing system,

- He checks if content in files is in reference with topics to be taught
- He adds the file in database and makes changes in other database file according to new file.

4.1.4.14.2.2 Alternative Flow
None

4.1.4.14.3 Special Requirements
None

4.1.4.14.4 Pre-condition
Content Provider must be logged in the system.

4.1.4.14.5 Post-Condition
None

4.1.4.14.6 Extension Points
None

### 4.1.4.15 Delete Files

4.1.4.15.1 Brief Description
This Use case describes how Content Provider deletes files from Content Delivery Engine.

4.1.4.15.2 Flow of Events

4.1.4.15.2.1 Basic flow
When Content Provider wishes to delete files from Content Delivery Engine,

- He erases file from database.
- He removes reference of that file from other files in database.

4.1.4.15.2.2 Alternative Flow
None

4.1.4.15.3 Special Requirements
None

4.1.4.15.4 Pre-condition
Content Provider must be logged in the system.

4.1.4.15.5 Post-Condition
None

4.1.4.15.6 Extension Points
None

# 4.2 System Design

## 4.2.1 Use Case Diagram

Use case diagrams are used to gather the requirements of a system including internal and external influences.

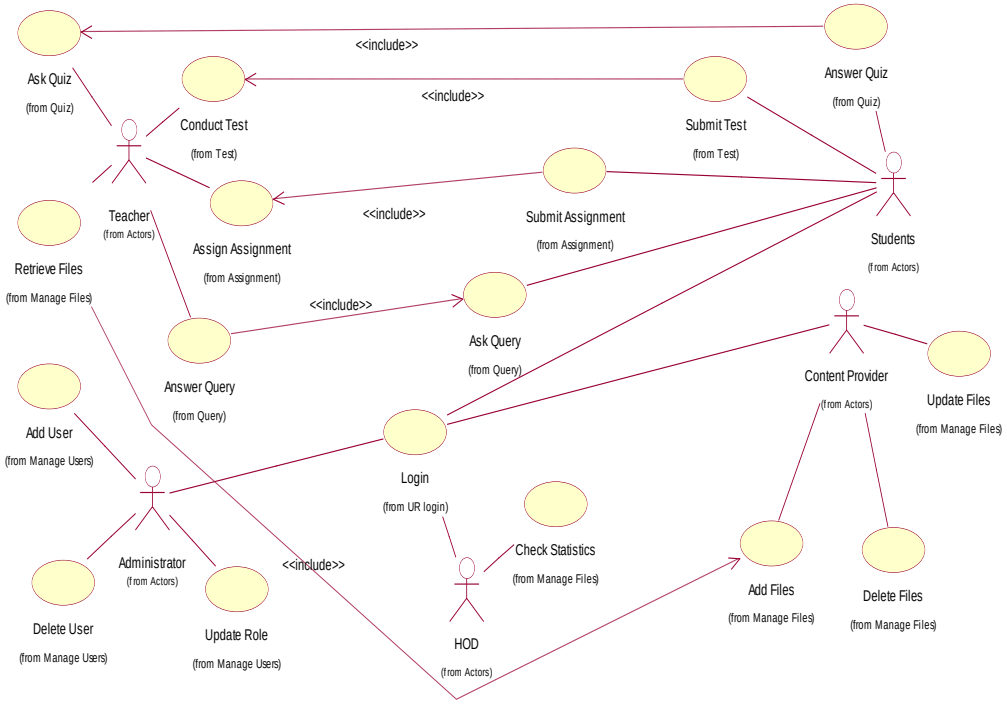



**Figure 4: Use Case Diagram**

### ❖ Use-Case Realization:

- **Actors: Administrator, Teachers, Students ,Head of the Departments , Content providers**
- An actor is an idealization of an external person, process, or thing interacting with a system, subsystem, or class. An actor characterizes the interactions that outside
- Users may have with the system.
- A student or a learner with knowledge of e-learning using his electronic devices like tabs or laptops. Such user will find the application useful.
- A teacher or a tutor who can use the product for taking e-classes digitally in the particular network with number of clients. He should be comfortable in handling the tablet devices.

- Any user can also use this application when in offline environment to access his documents as well as media files.
- An administrator defines different roles and takes care of usernames and passwords of application users.
- Head of Departments are just the readers who use the system to check the statistics about overall performances.
- Content providers just do the work of updating the contents of the application databases.

Use Case Realization for the use cases.

### 4.2.2 Login :

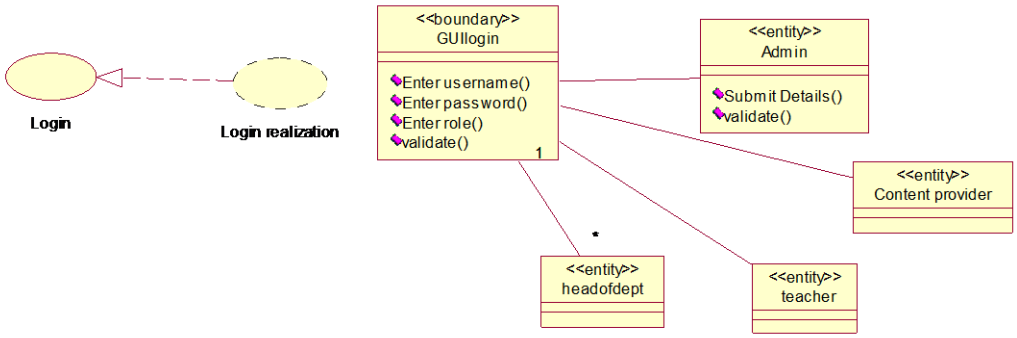



**Figure 5: Use Case Realization of Login Use case**

### 4.2.3 Add User

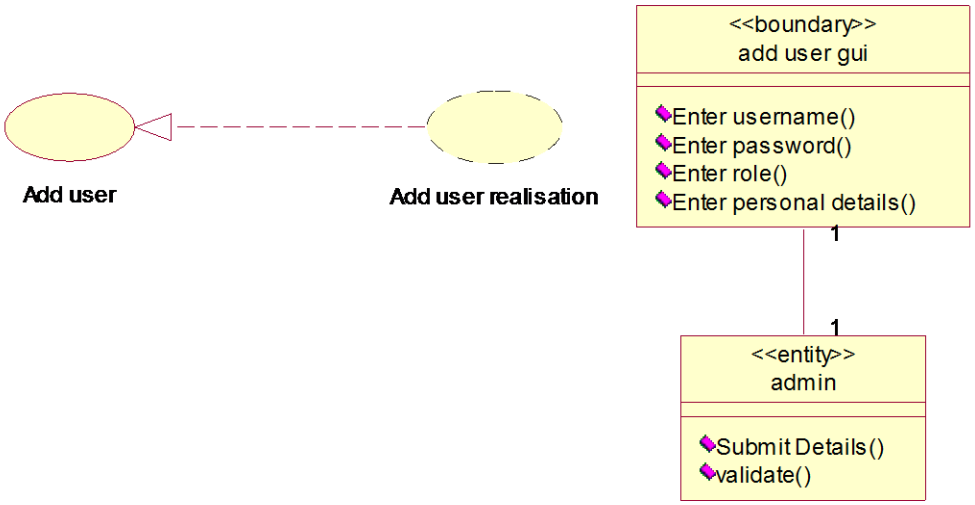



**Figure 6: Use Case Realization of Add User Usecase**

### 4.2.4 Check_Statistics

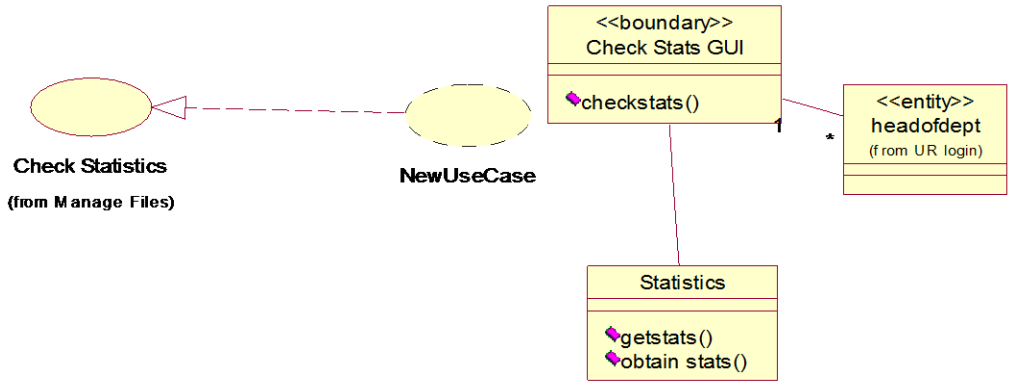



**Figure 7: Use Case Realization of Check Statistics Usecase**

### 4.2.5 Add files

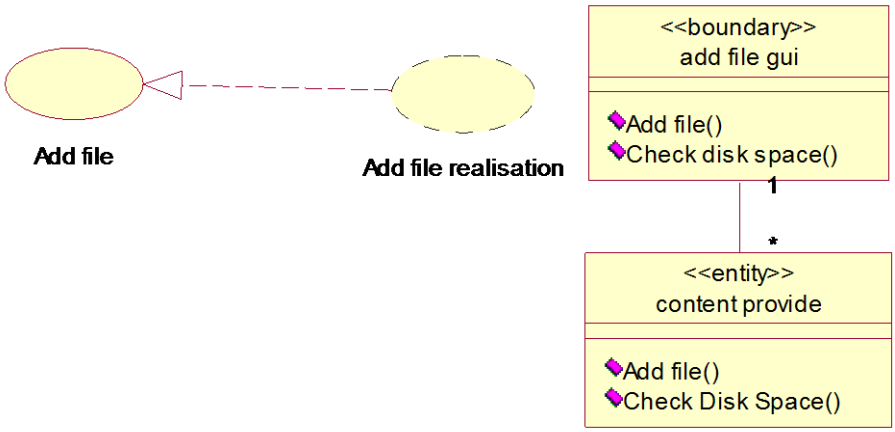



**Figure 8: Use Case Realization of Add Files Usecase**

### 4.2.6 Delete files and Update files (same as Add file)

### 4.2.7 Assign Assignment

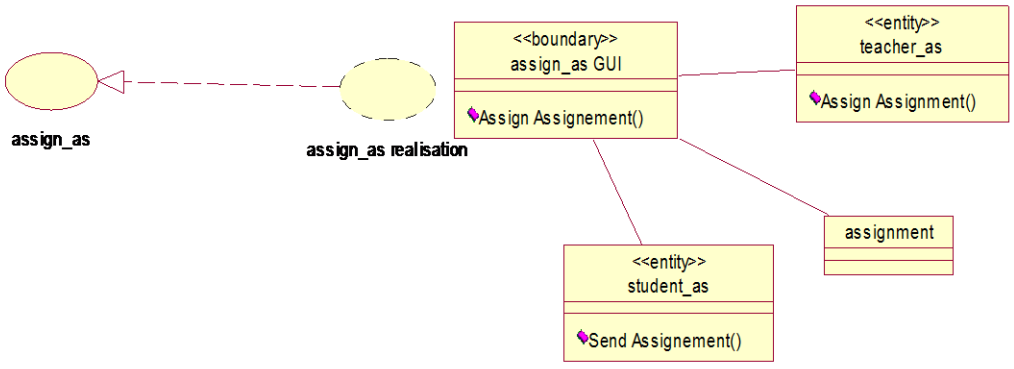



**Figure 9: Use Case Realization of Assign Assignment Usecase**

### 4.2.8 Answer Query

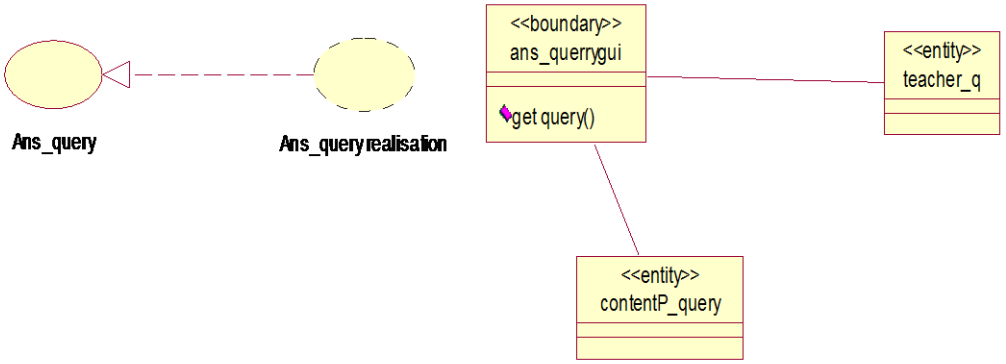



**Figure 10: Use Case Realization of Answer Query Usecase**

### 4.2.9 Answer Quiz

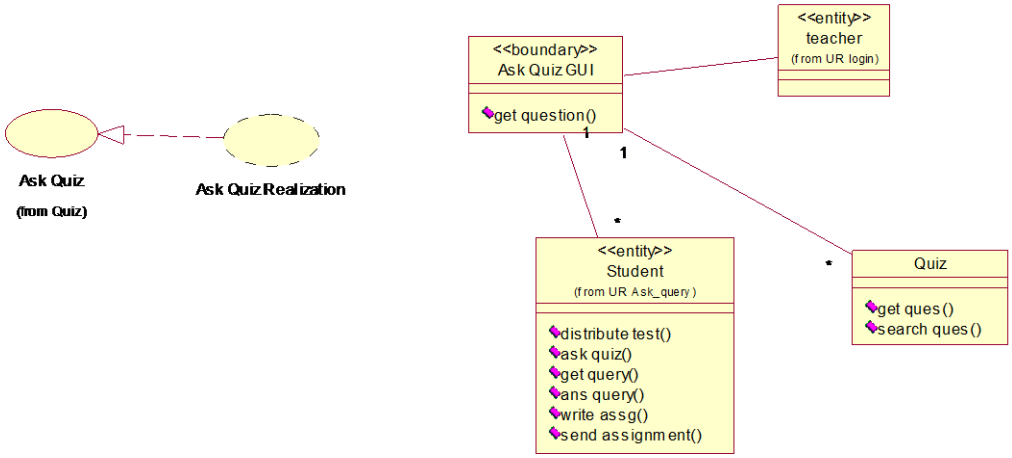



**Figure 11: Use Case Realization of Answer Quiz Usecase**

### 4.2.10 Ask Query

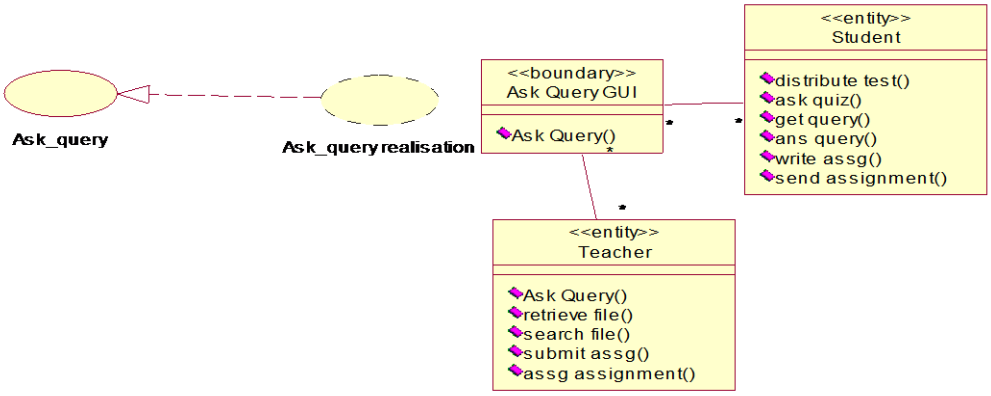



**Figure 12: Use Case Realization of Ask Query Usecase**

### 4.2.11 Conduct Test

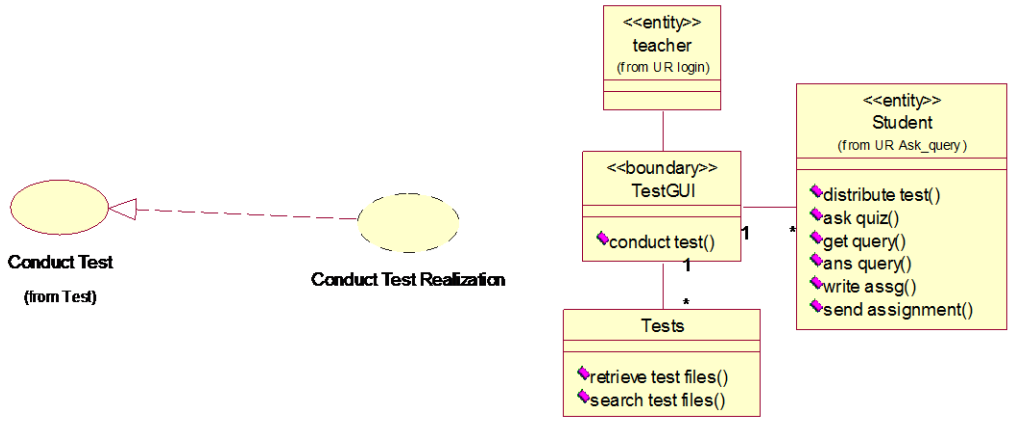



**Figure 13: Use Case Realization of Conduct Test Usecase**

### 4.2.12 Retrieve files

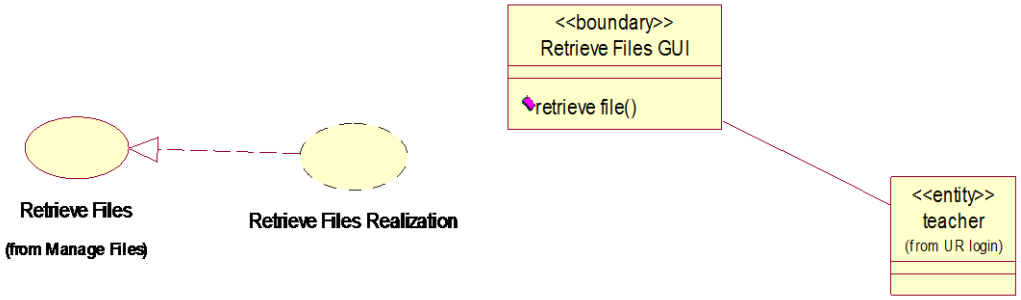



**Figure 14: Use Case Realization of Retrieve files Usecase**

## 4.3 Activity Diagram

- **A form of "OO flowcharts".**
- Advantages:
    - Greater flexibility to model different types of flow
    - Clear distinction between activity diagrams and state machines
- These are attached to modeling elements to describe behavior.
- an activity diagram the purpose of which is to model the real-world workflows of a human organization
- Typically attached to use cases, classes, components, interfaces, collaborations, and operations
- Activity Diagram Can be used also to model business processes and workflows and gives the sequence in which particular actions take place in the application.

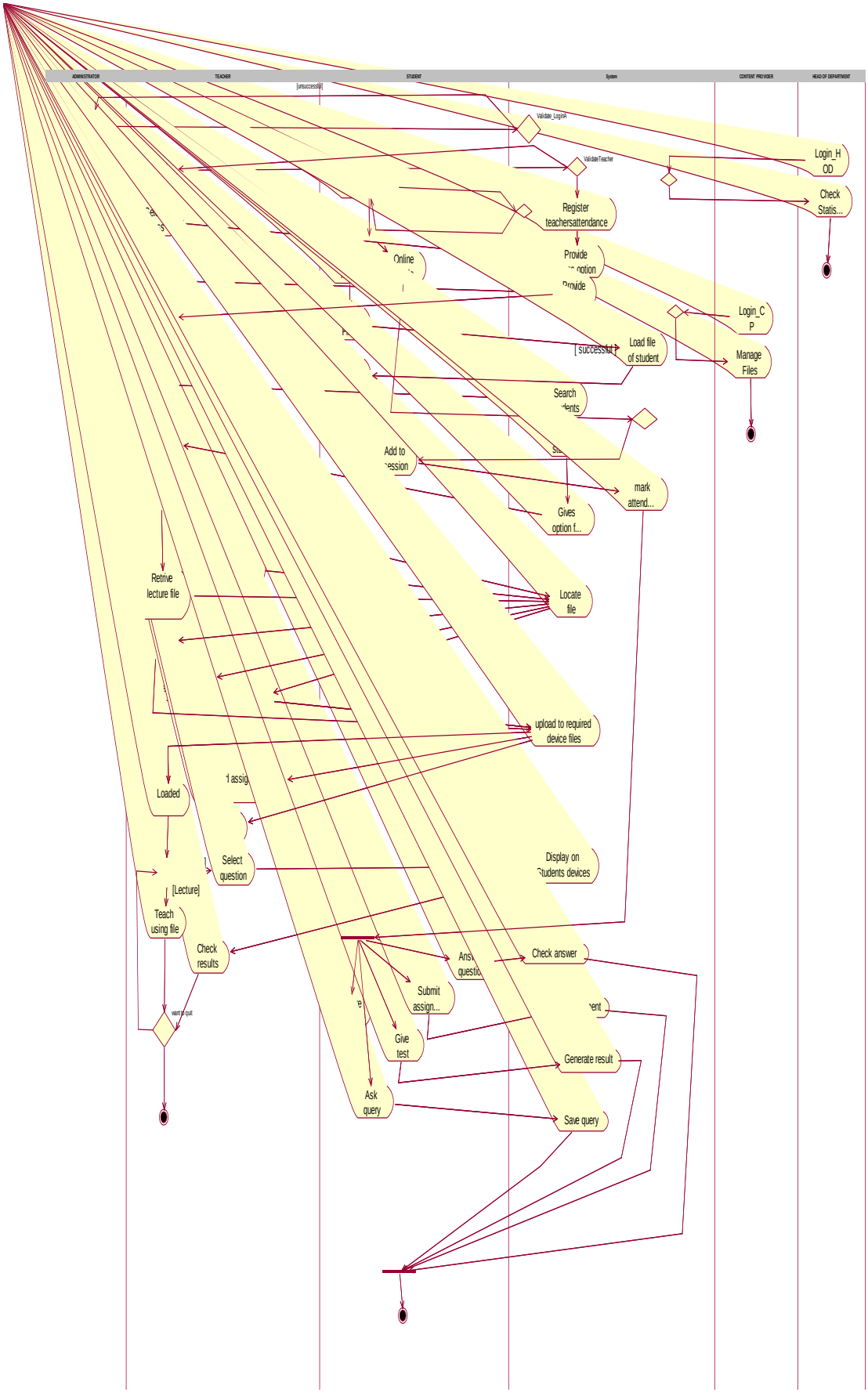



**Figure 15: Activity Diagram**

# CHAPTER 5

# PROJECT DESIGN

# 5. Project Design

## 5.1 Sequence Diagrams

- Sequence diagrams specify the flow of actions in the particular event of the systems.
- A sequence diagram displays an interaction as a two-dimensional chart.
- The vertical Dimension is the time axis; time proceeds down the page.
- The horizontal dimension shows the classifier roles that represent individual objects in the Collaboration. Each classifier role is represented by a vertical column—the lifeline. During the time an object exists, the role is shown by a dashed line. During the time an activation of a procedure on the object is active, the lifeline is drawn as a double line.
- A message is shown as an arrow from the lifeline of one object to that of another.

  Each sequence diagram

  The sequence diagram of each use case is shown below

### 5.1.1 Login

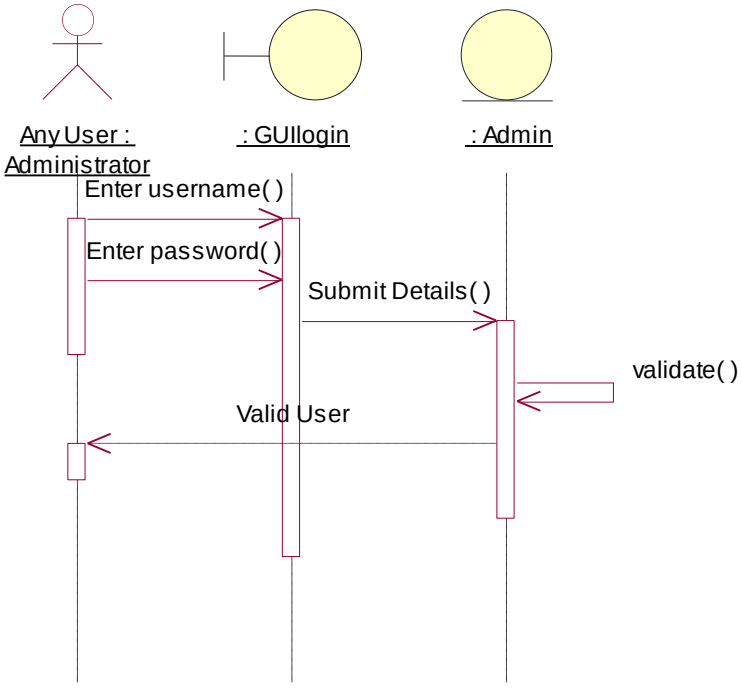



**Figure 16: Sequence Diagram for Login**

### 5.1.2 Add User

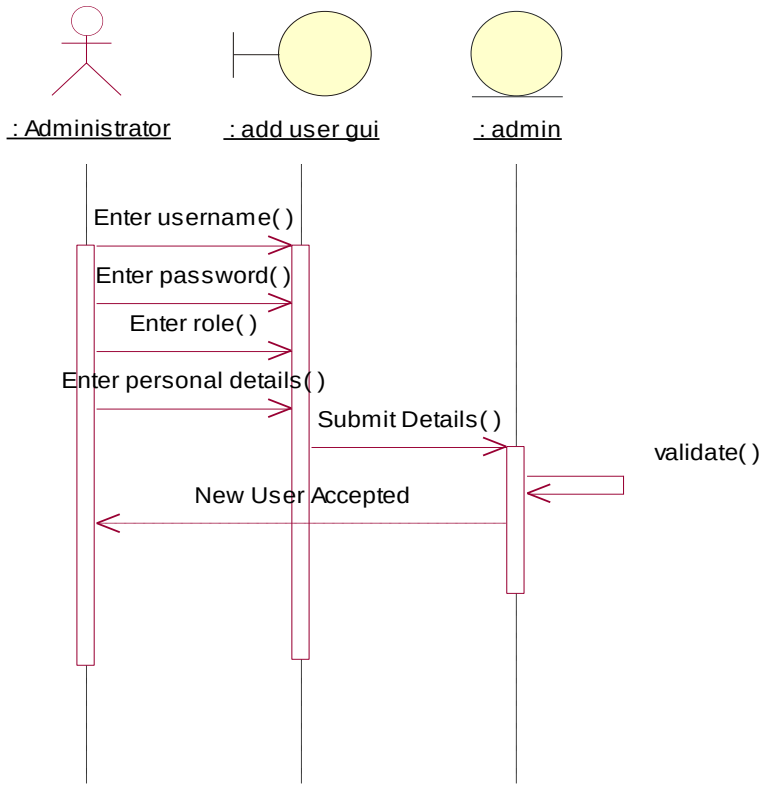



**Figure 17: Sequence Diagram for Add User**

### 5.1.3 Retrieve file

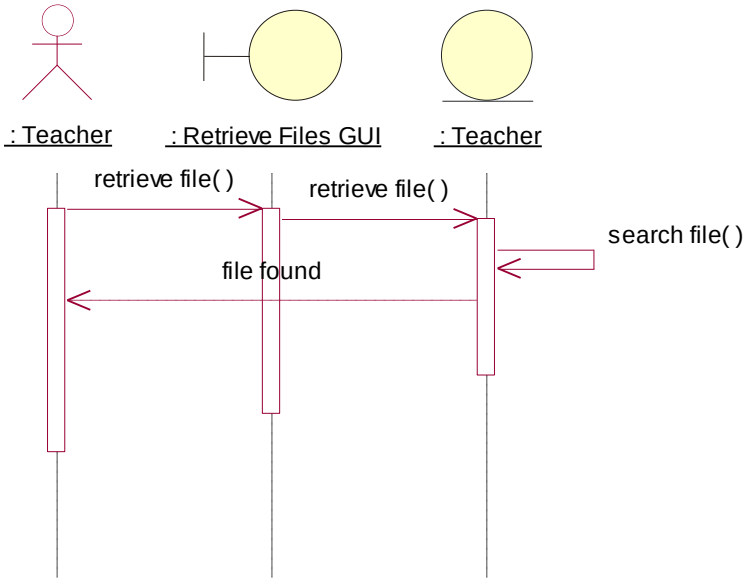



**Figure 18: Sequence Diagram for Retrieve File**

### 5.1.4 Conduct Test

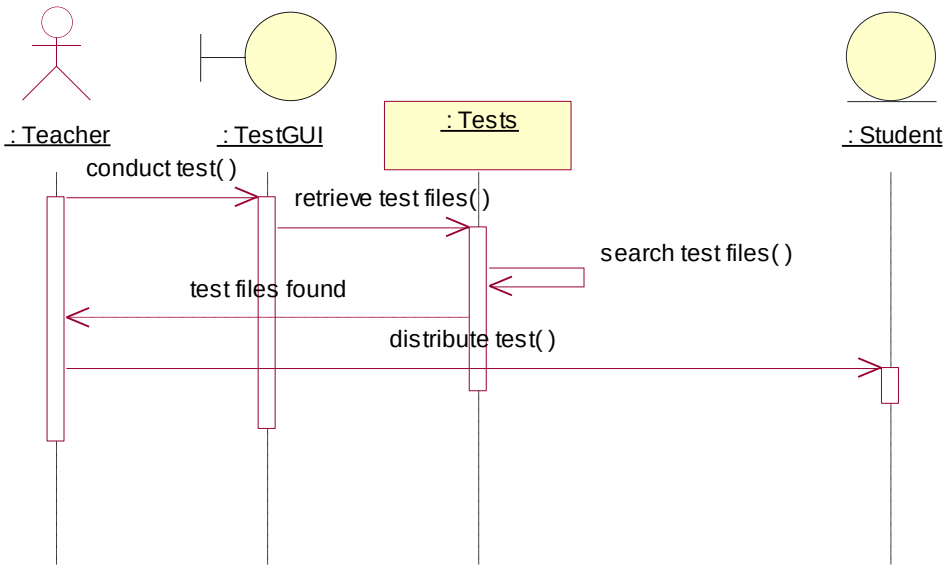



**Figure 19: Sequence Diagram for Conduct Test**

### 5.1.5 Check Statistics

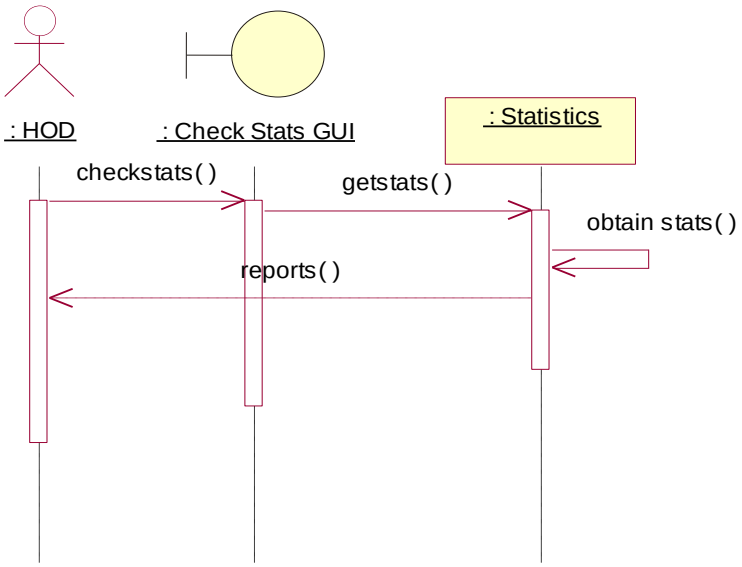



**Figure 20: Sequence Diagram for Check Statistics**

### 5.1.6 Assign Assignment

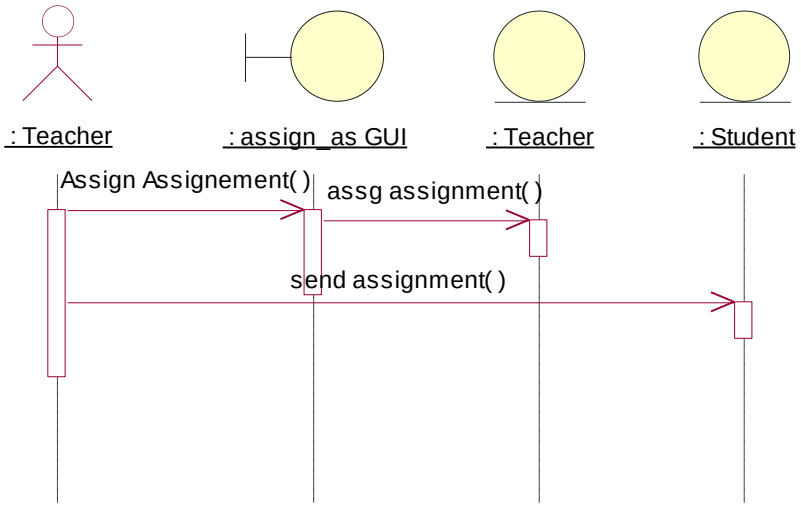



**Figure 21: Sequence Diagram for Assign Assignment**

### 5.1.7 Ask Query

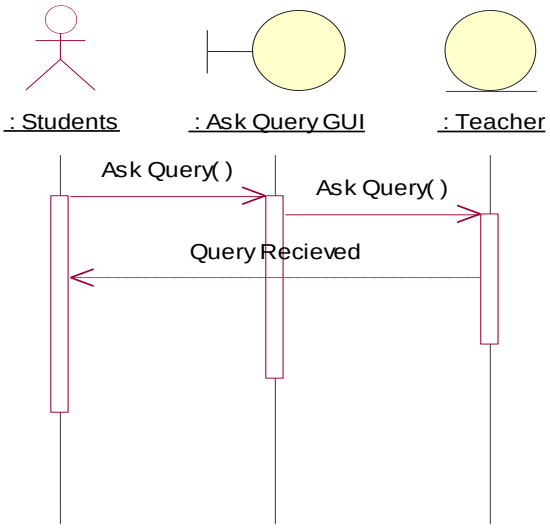



**Figure 22: Sequence Diagram for Ask Query**

### 5.1.8 Ask Quiz

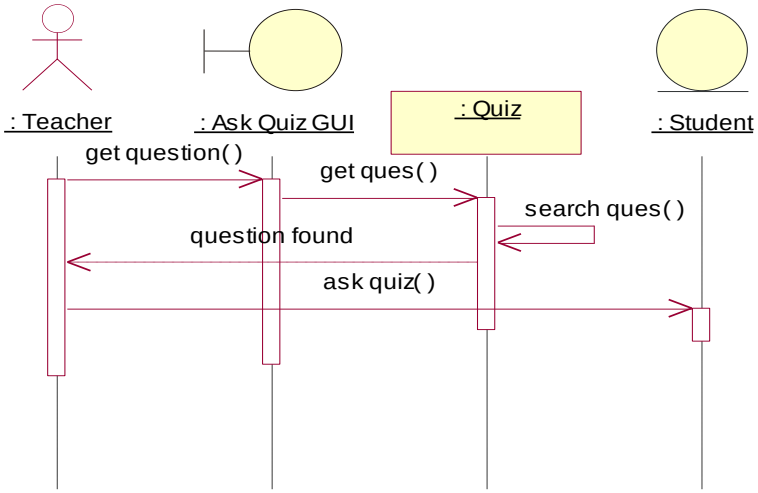



**Figure 23: Sequence Diagram for Ask Quiz**

### 5.1.9 Answer Query

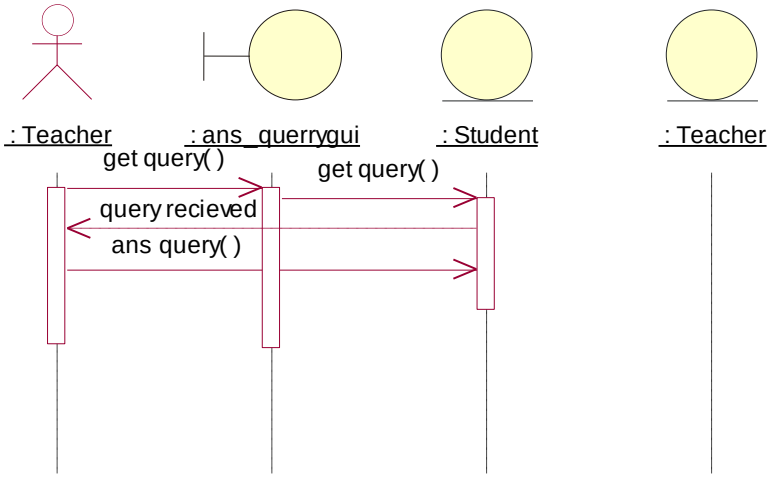



**Figure 24: Sequence Diagram for Answer Query**

### 5.1.10 Assign Assignment

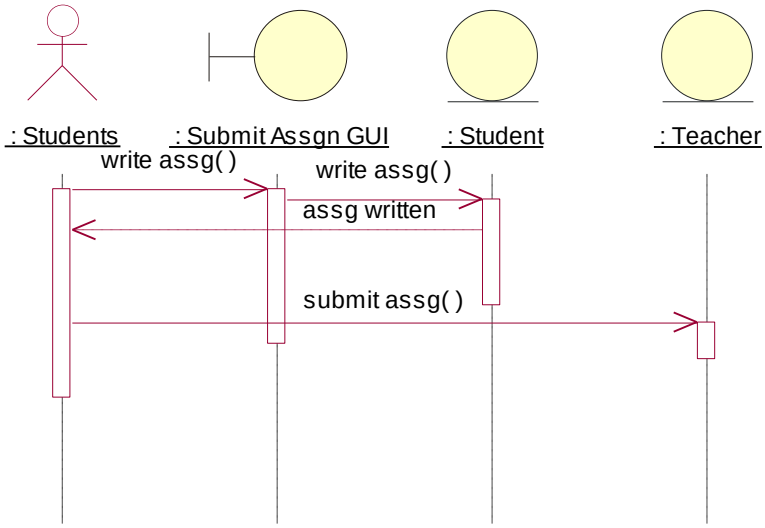



**Figure 25: Sequence Diagram for Assign Assignment**

### 5.1.11 Add Files

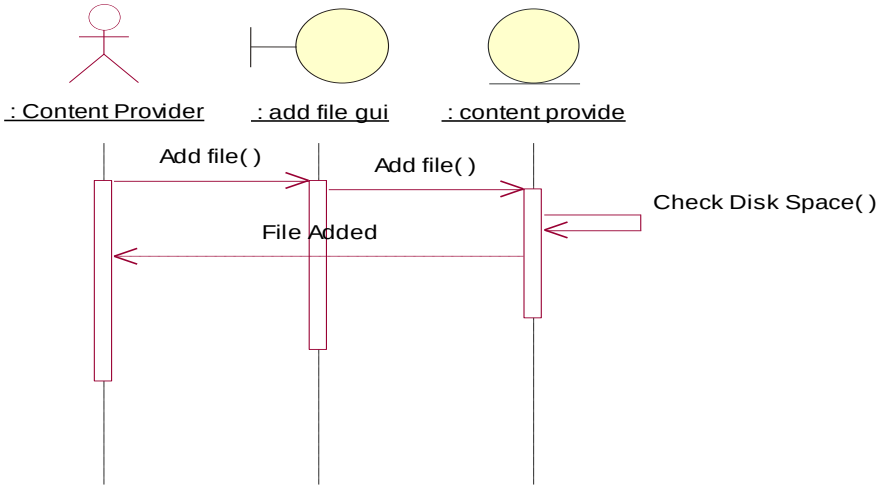



**Figure 26: Sequence Diagram for Add Files**

## 5.2 Class diagram

- UML provides class diagrams for illustrating classes, interfaces and their associations in system.
- Class diagrams are used for static object modeling.
- Note that class diagrams are also used for visualizing domain models.
- The term design class diagram (DCD) will be used to refer to the class diagram used in the design perspective.

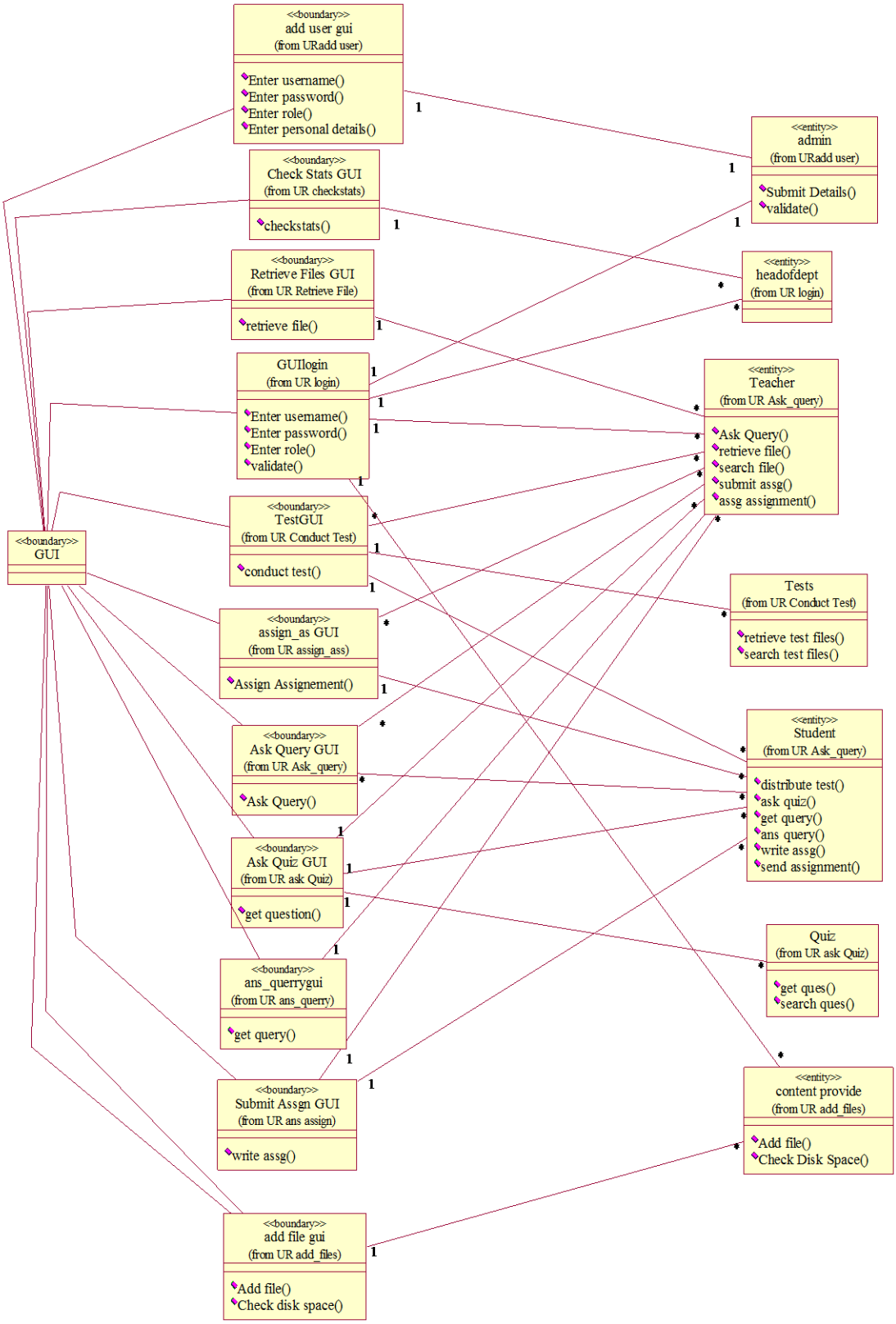



**Figure 27: Class Diagram**

# CHAPTER 6

# IMPLEMENTATION DESIGN

# 6. Implementation Design

## 6.1 Module and Description

Client Side: All the client side modules have been built using the Eclipse IDE. The GUI has been made in the XML format with Java being used for Query processing, GUI handling and other important features as described below.

Server Side: All the server side scripting is being done by the XAMPP (X-Apache-MySQL-Perl-PHP) server, where query processing is handled by the php files and database in SQL language.

### 6.1.1 Automatic Attendance

Client Side: The user (teacher or student) will login to the system with their respective username, password and role.

Server Side: The php scripting with validate the credentials of the user and on successful login the attendance of the user will be stored in the database for the date and time.

### 6.1.2 Manage Users (Admin)

Client Side: The Admin can add or Delete users (teacher/student) to the system by filling up the details of the users and submit the form.

Server Side: The php scripting will check if such user exits and take upon the necessary steps (add/delete/notify accordingly).

### 6.1.3 Manage Self Account (All Users)

Client Side: The users can edit their Username or Password with the help of the given form. At the client side itself certain validations like valid username, valid re-entry of username/password will take place.

Server Side: The php scripting will edit the changes in the database and notify the same to the user.

### 6.1.4 Assignment

Client Side 1(Teacher): This module contains two parameters, one being check uploaded answers and the other upload new set of questions. The teacher can upload

the assignment questions to the server. The teacher can also check the answers uploaded by the students.

Client Side 2 (Student): This module contains two parameters, one being check uploaded questions and the other upload answers of questions. The student can check the questions uploaded by the teachers. The student can upload the assignment answers to the server, here the student need to note that he/she has to upload the file with name as 'Std-Div-Roll no' format only.

Server Side: The selected file will be type converted and added to the server in the specified folder, (Answers for Students and Questions for Teachers).

### 6.1.5 Notes

Client Side 1(Teacher): The teacher can upload the notes to the server by selecting the file from the device storage. Notes of any file format can be uploaded. Files that need to be accessed later by the student in the offline made need to be in the .doc or .pdf file format while those only for online purpose will have to be in the .txt file format.

Client Side 2 (Student): The student can check the notes uploaded by the teachers. Notes with downloadable formats like (.doc and .pdf) will have to be saved into student's device for viewing them; these files can also be accessed in the offline mode with the help of the inbuilt reader. Notes with file format .txt cannot be downloaded and can be accessed only during the online mode and need net require a reader for the purpose.

Server Side: The selected file will be type converted and added to the server in the specified folder, (Notes).

### 6.1.6 Query

Client Side: The teacher and student share a common screen here. They can view the queries by the students and responses given by either other students or the teacher. Here the teacher has an added feature of clearing all the Queries of the students when required or at the end of the classroom.

Server Side: The messages (queries or responses) get added to the database and list only the last 20 messages on the screen. It also ensures that the screen is refreshed every 10 secs to display the new message.

### 6.1.7 Quiz

Client Side 1 (Teacher): This module contains two parameters. The teacher can add a quiz question during the class session and also check the responses given by each student. The teacher can also erase all the previous questions and answers using the 'Clear Questions' and 'Clear Answers'.

Client Side 2 (Students): The students can only view the question by the teacher and respond accordingly; on successful entry, they will be notified. The students cannot view responses from other students.

Server Side: The messages (questions or responses) get added to the database and list only the last 20 messages on the screen. It also ensures that the screen is refreshed every 10 secs to display the new message.

### 6.1.8 Test

Client Side 1 (Teacher): This module provides the three functionalities to the teacher, create a new test, conduct test and check results. The create new test allows to create new test, add questions and answers. The conduct test displays the credentials of the test (testid, subject and password) which will be announced to the students during the test. The check result provides the results of the test given by the students.

Client Side 2 (Student): The student can give test by adding the test credentials. If the student has already given the test then appropriate message will be displayed, else the student can give the test. During the test session, the student can submit and resubmit answer and on final test submit no changes can be future made.

Server Side 1(Teacher): The php scripting will add the questions and answers to the database on creation of new test; retrieve the test credentials and results for conduct test and check results options respectively.

Server Side 2 (Student): The php scripting will add the answers of the students to the database also update it on changing the answer; on test completion i.e. final submission of test the students result will be calculated.

## 6.2 Snapshots

**Front End GUI**


**Figure 28: Splash Screen Snapshot**

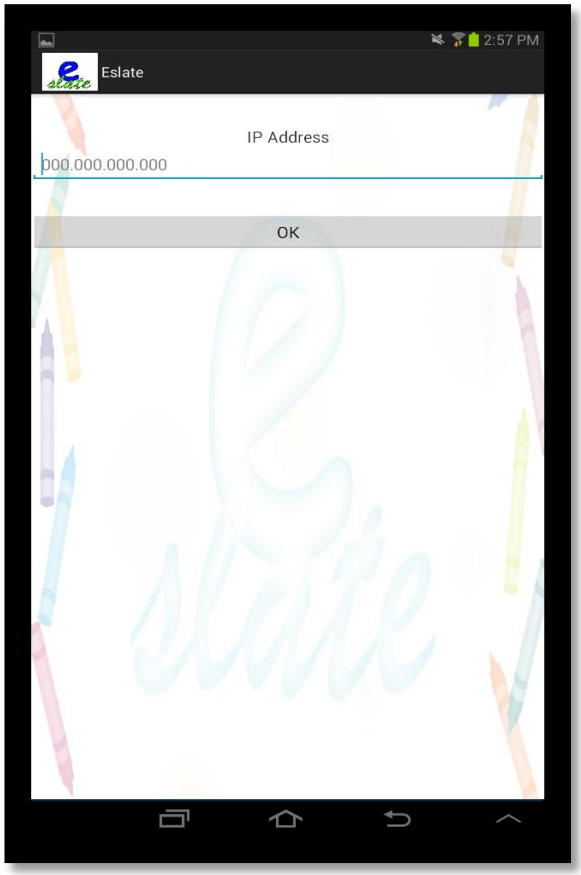



**Figure 29: IP Address Snapshot**

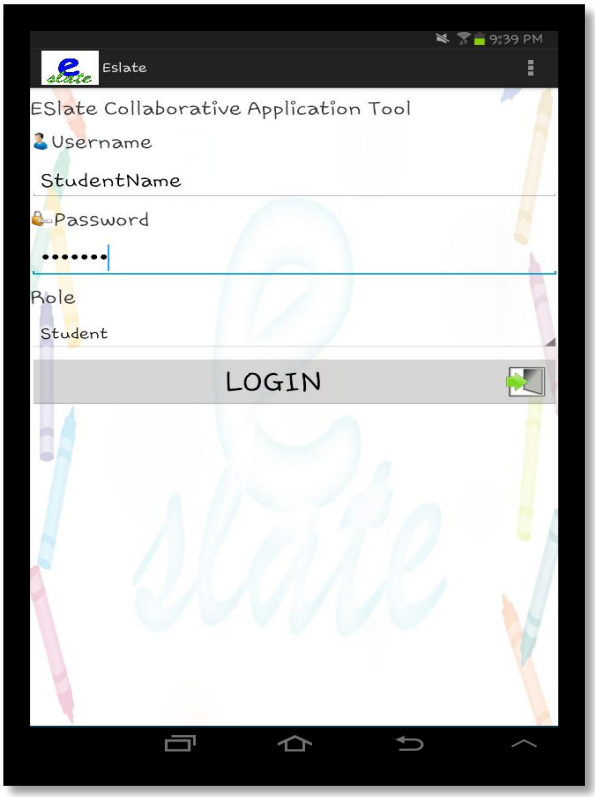



**Figure 30: Login Screen Snapshot**

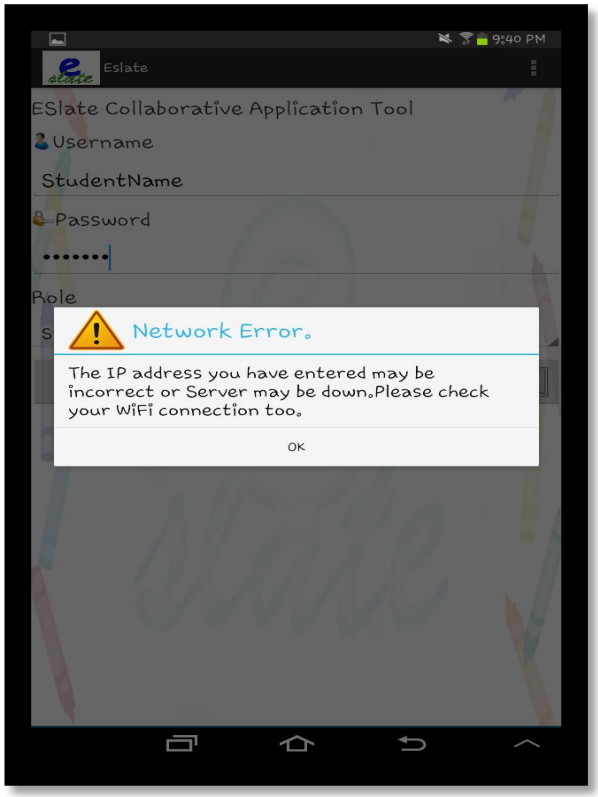



**Figure 31: Connection Failed Snapshot**

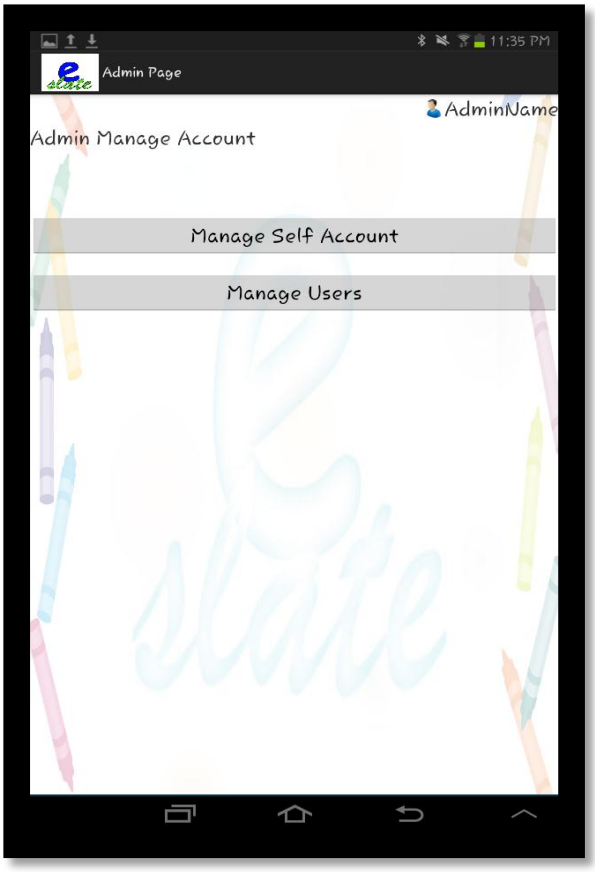



**Figure 32: Admin Page Snapshot**

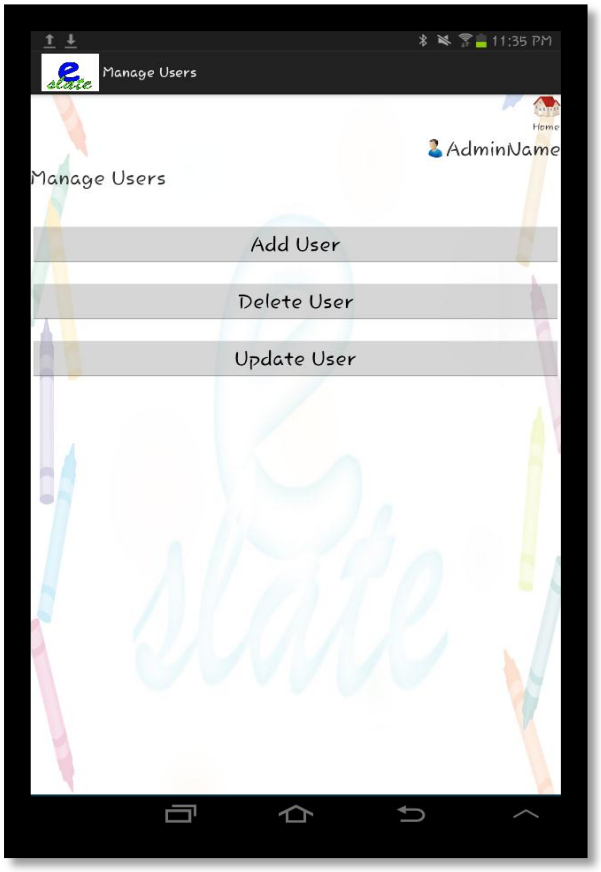



**Figure 33: Admin Manage User Snapshot**

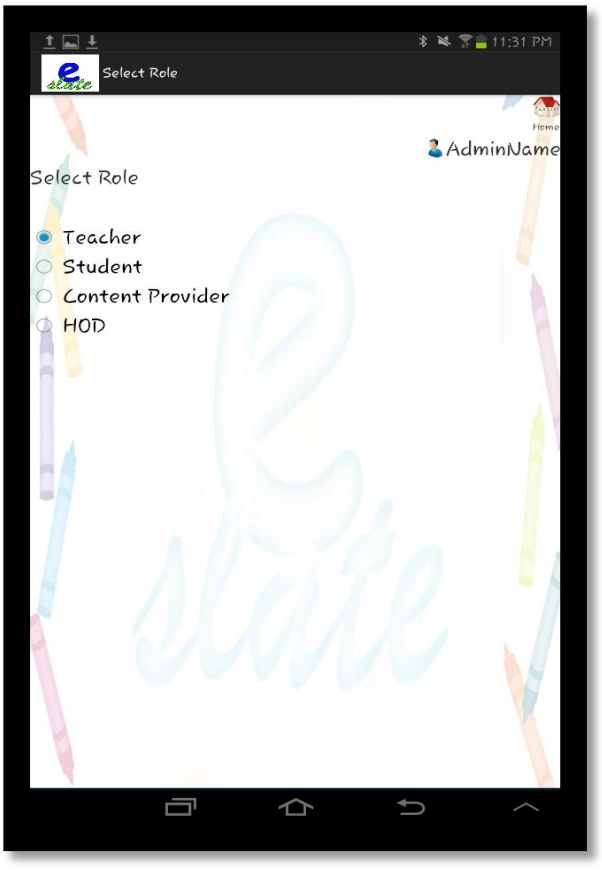



**Figure 34: Select Role Snapshot**

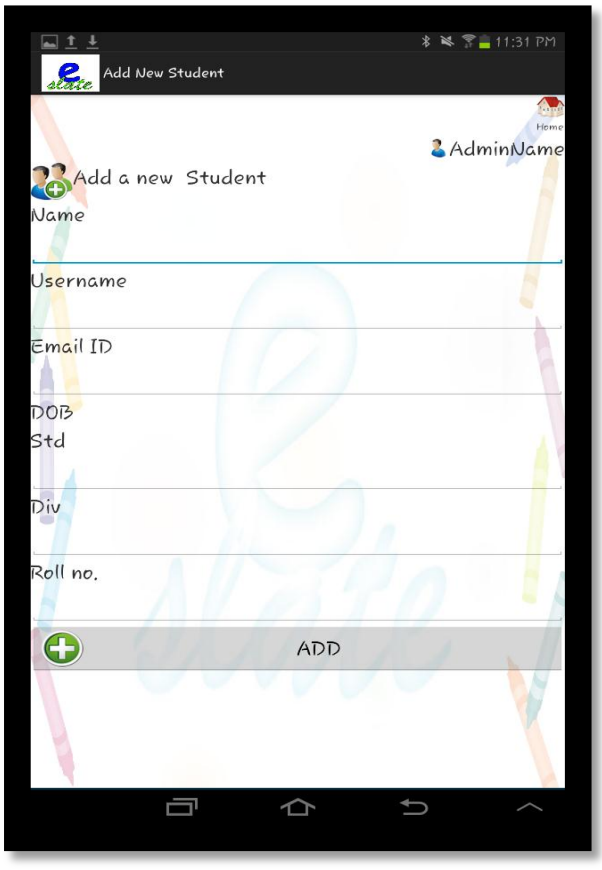



**Figure 35: Add New User Snapshot**

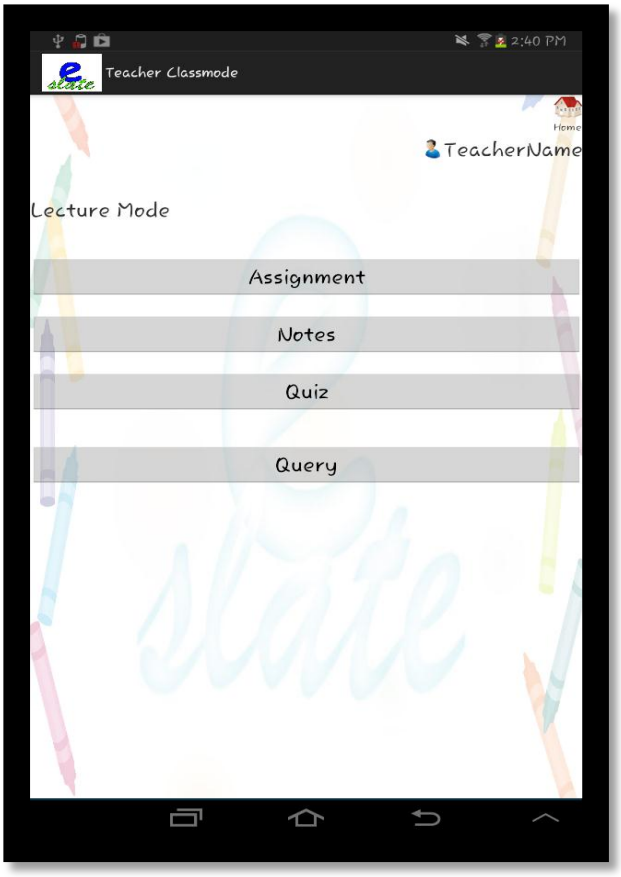



**Figure 36: Classroom options Snapshot**

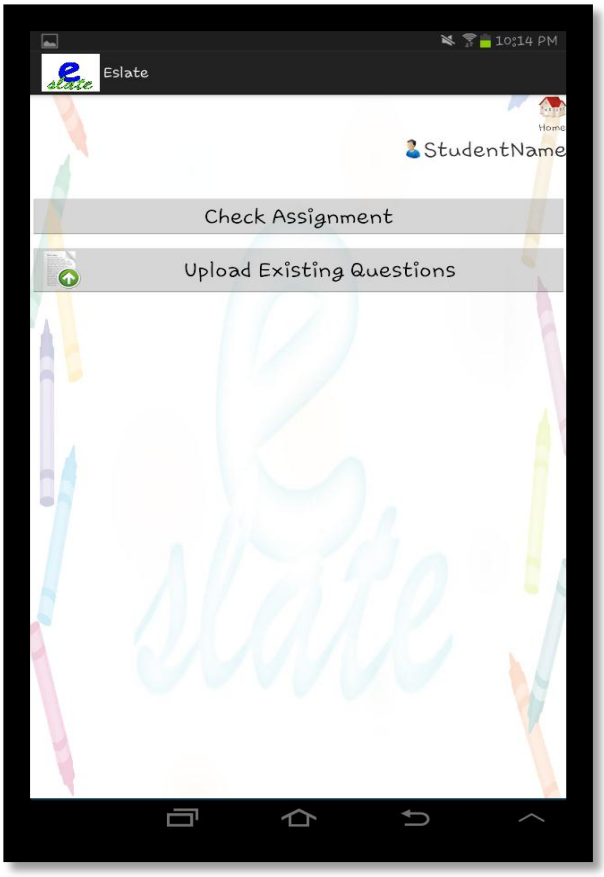



**Figure 37: Assignment**

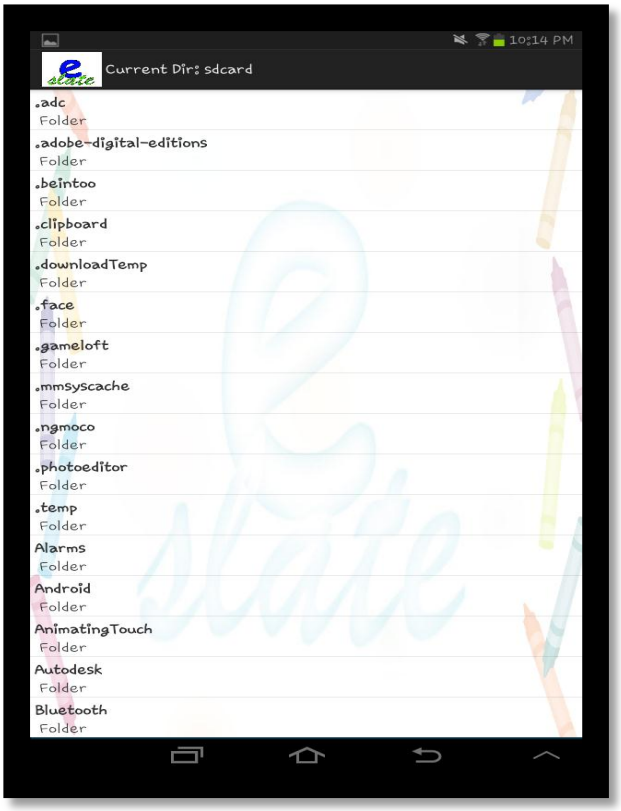



**Figure 38: Directory View of External Storage**

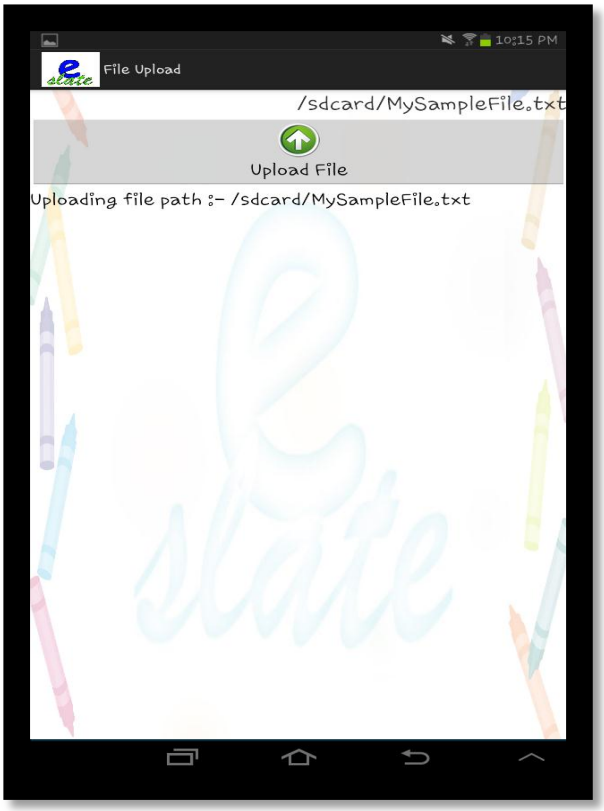



**Figure 39:Upload Assignment to Server**


**Figure 40: List of Uploaded Notes Snapshot**


**Figure 41: Assignment Teacher View Snapshot**

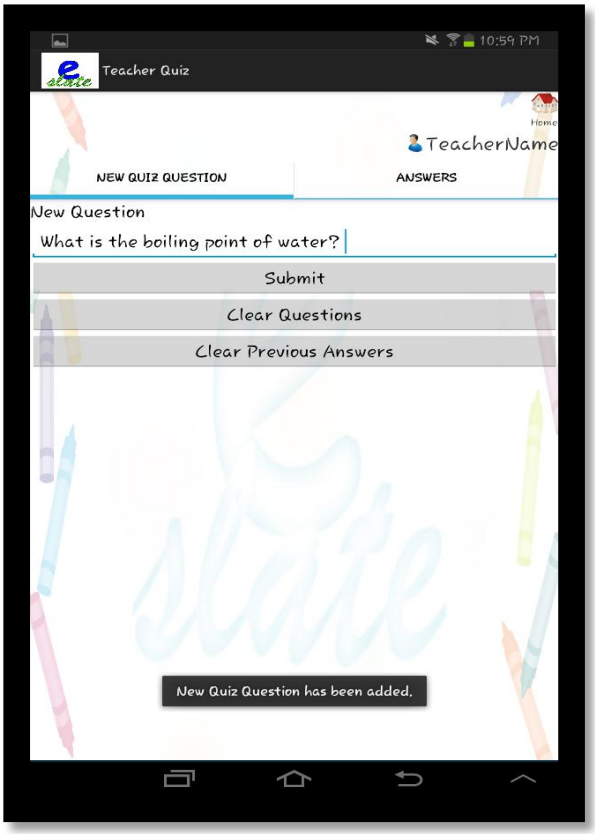



**Figure 42: Quiz View of Teacher Snapshot**

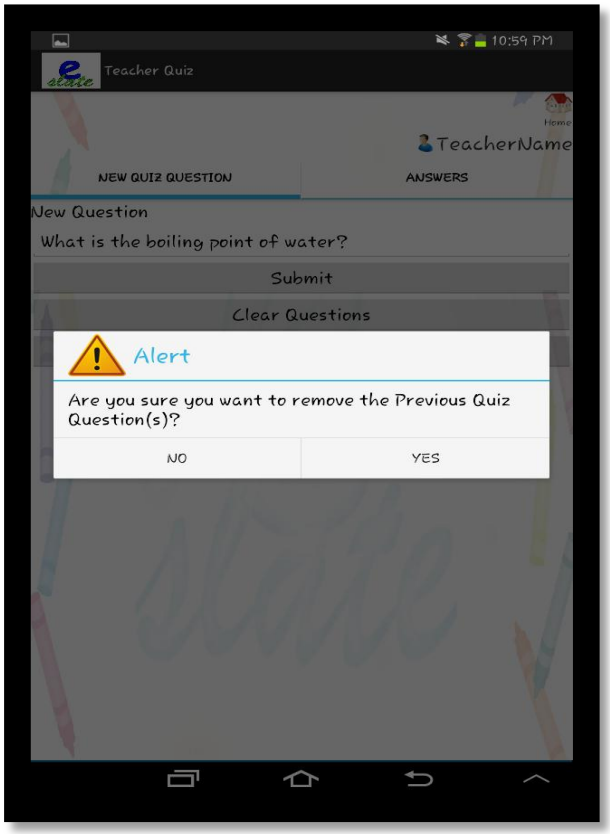



**Figure 43:Delete Previous Question Snapshot**

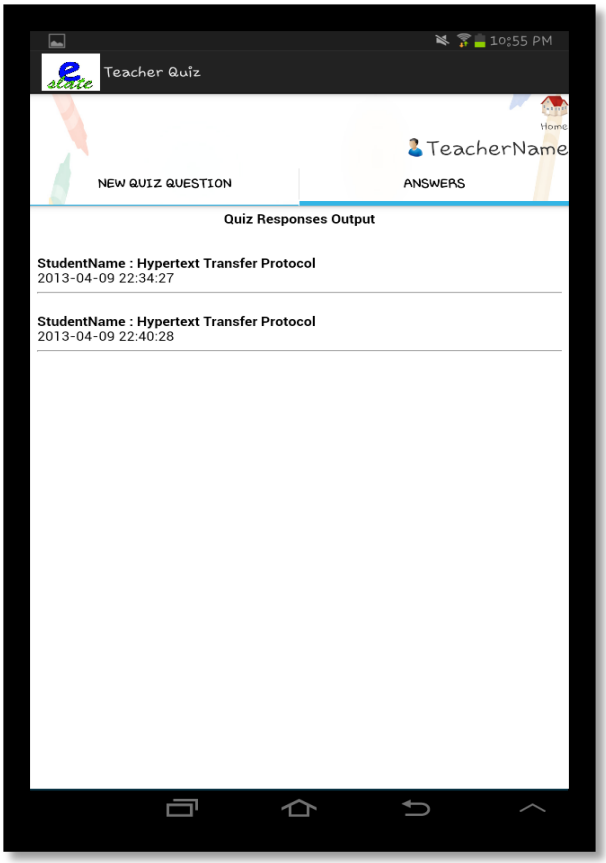



**Figure 44: Teacher View of Quiz responses Snapshot**

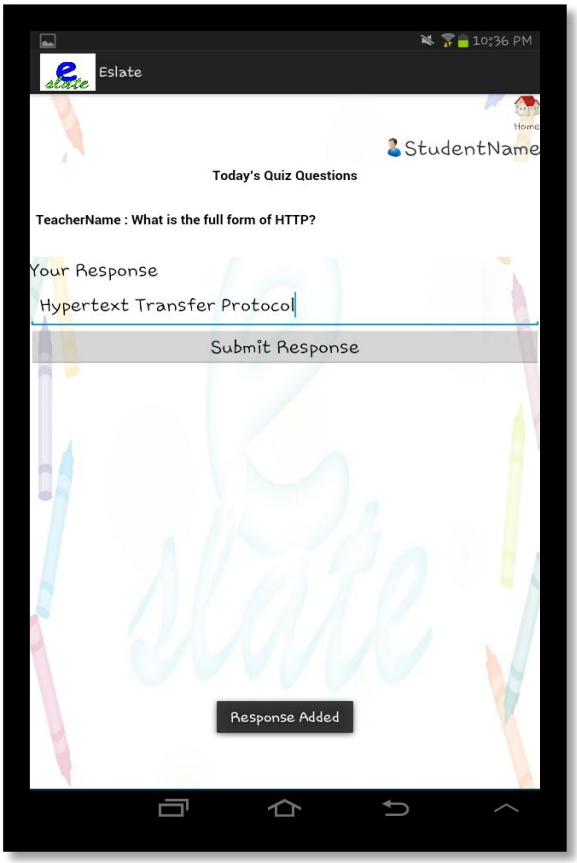



**Figure 45:Student View of Quiz Snapshot**

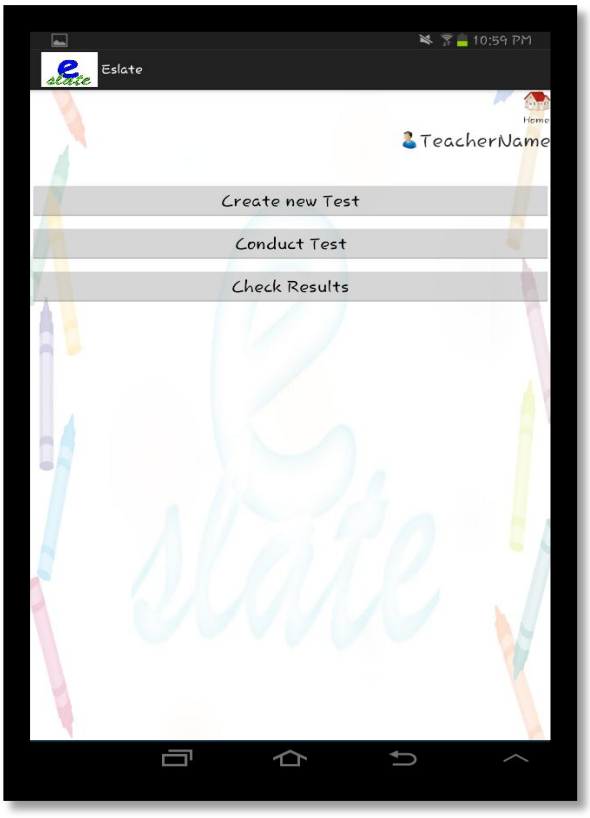



**Figure 46: Testmode Teacher View Snapshot**

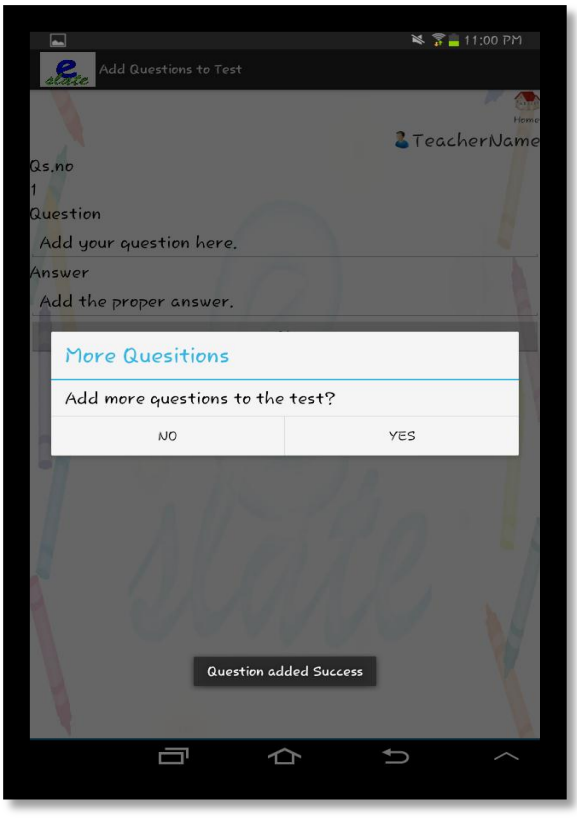



**Figure 47: Create Test Questions-Teacher View Snapshot**

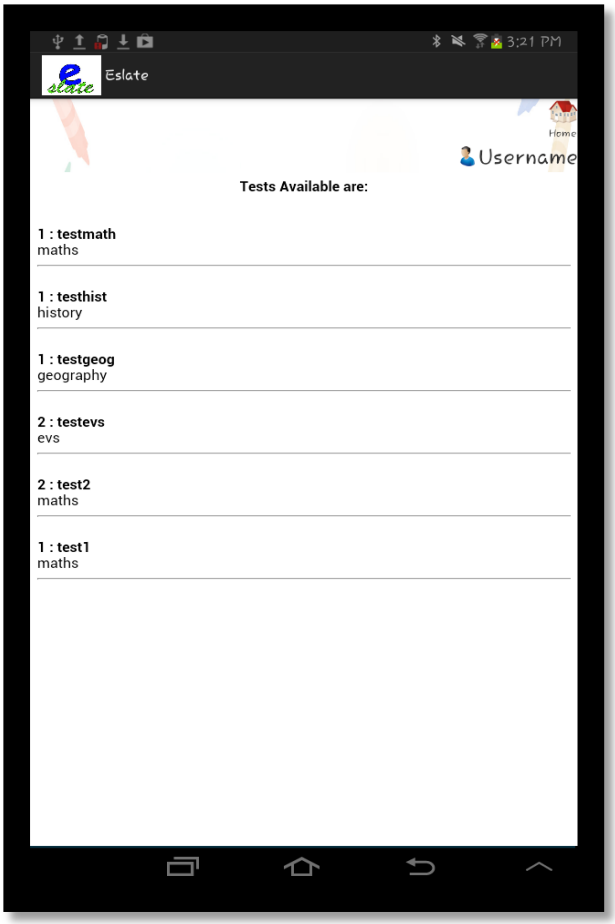



**Figure 48: Conduct Test Teacher View Snapshot**

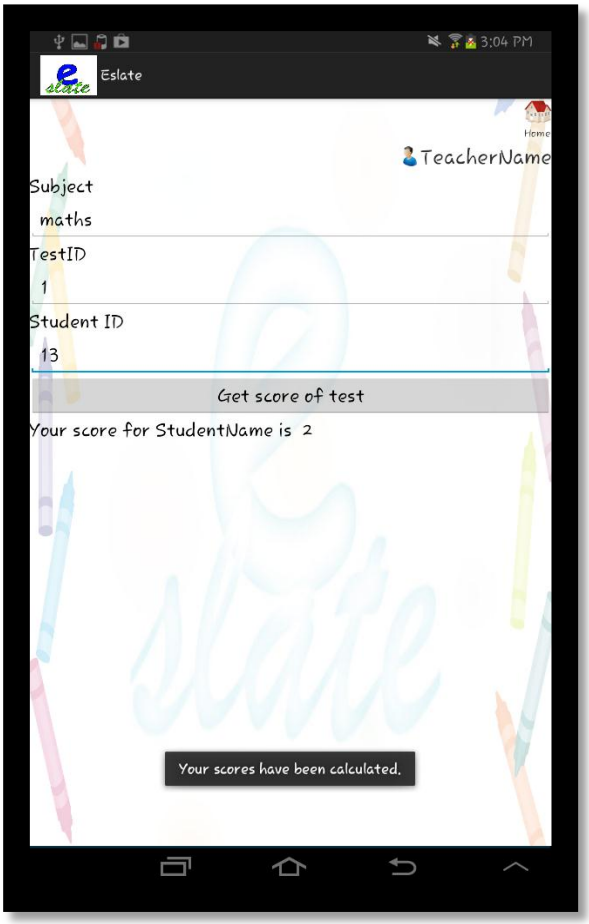



**Figure 49: Check Results Snapshot**

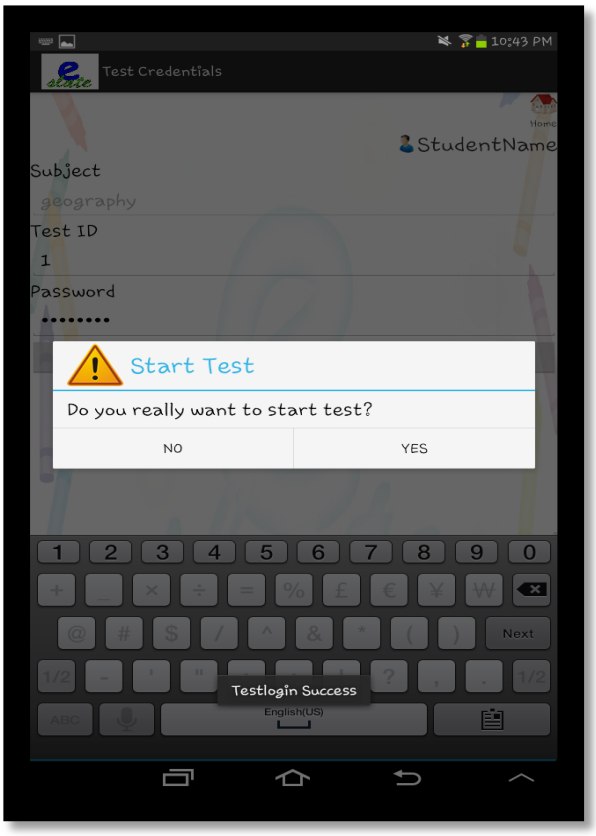



**Figure 50: Test Credentials Student View Snapshot**

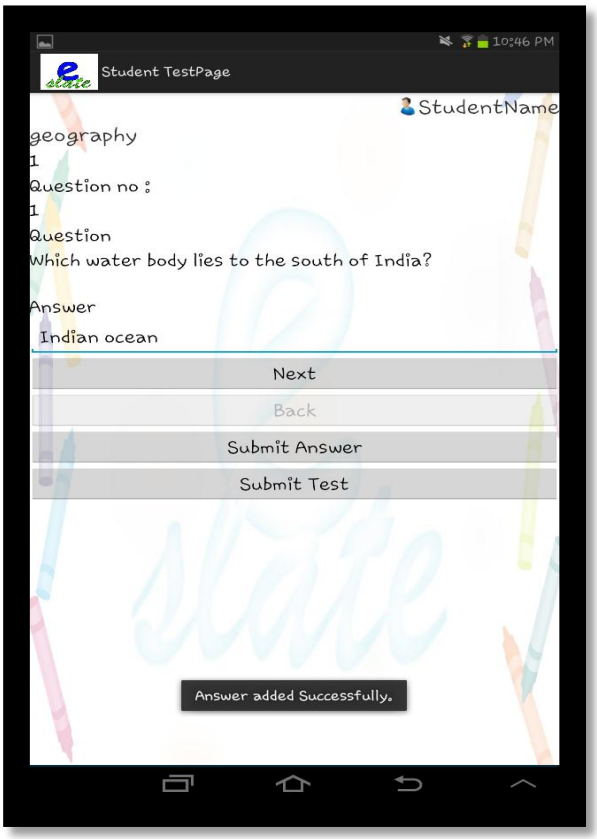



**Figure 51: Take Test Student View Snapshot**

## Server Side and Back End (Apache Server and MySQL DATABASE)

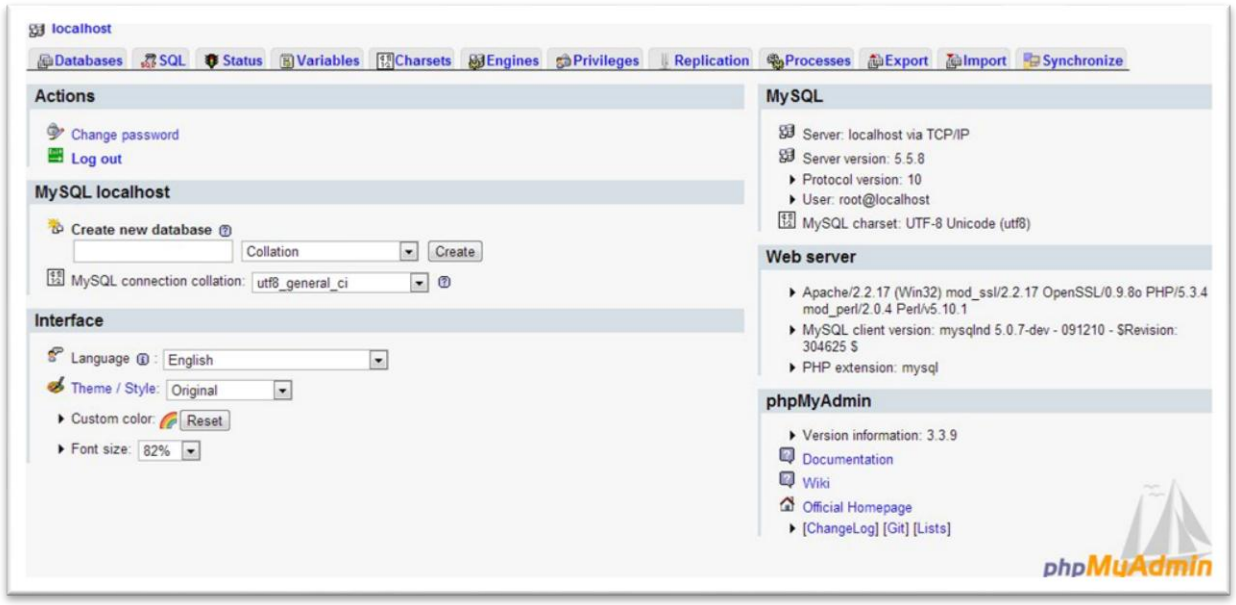



**Figure 52: PHPMyAdmin of XAMPP Server**

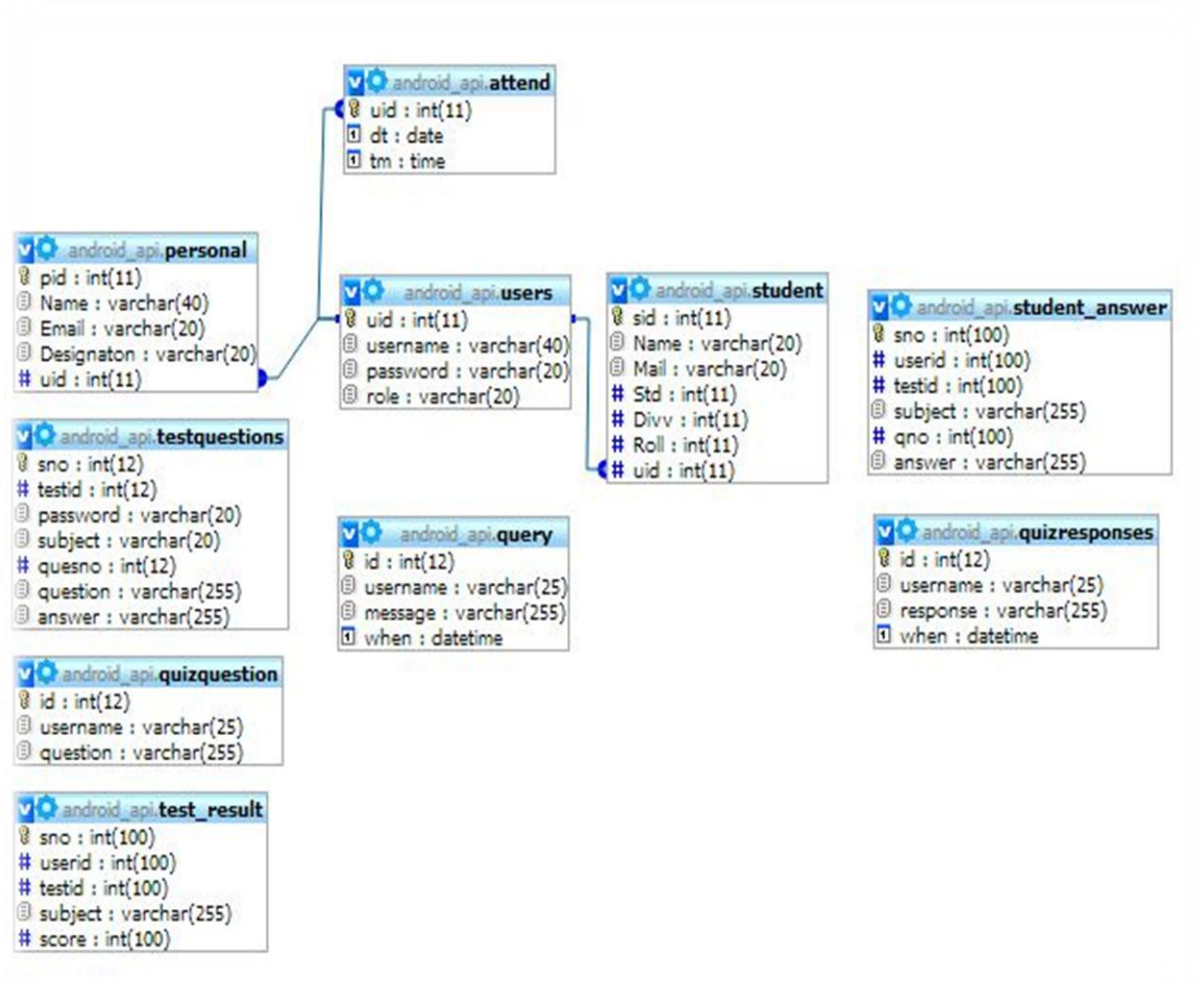



**Figure 53: Database Schema Implementation**

# CHAPTER 7

# TESTING

# 7. Testing

Development of test cases (functional)

| Test Case ID | Objective | Steps/Description | Input | Expected Output | Actual Output | Result |
|---|---|---|---|---|---|---|
| 1 | Correct Validation | Enter correct validation details | Correct username, Password, Role | User validation message. Go to next page | User validated | Pass |
| 2 | Wrong validation | Enter wrong validation details | Wrong username, password | Alert message about wrong inputs | Alert message received | Pass |
| 3 | Change username | Enter old and new username | Old username, password, new username | Success message. Username changed | Success message received | Pass |
| 4 | Incorrect username while changing username | Enter incorrect old username | Old username, password, new username | Alert message about wrong username | Alert message received | Pass |
| 5 | Incorrect new username | Don't enter new username correct twice | Old username, password, new username | Alert message about new username entered wrong | Alert message received | Pass |
| 6 | Change password | Enter old and new password | Old username, password, new password | Success message. Password changed | Success message received | Pass |
| 7 | Incorrect password while changing password | Enter incorrect old password | Old username, password, new password | Alert message about wrong password | Alert message received | Pass |
| 8 | Incorrect new password | Don't enter new password correct twice | Old username, password, new password | Alert message about new password entered wrong | Alert message received | Pass |
| 9 | Add teacher | Enter personal details of teacher and get success message | Name, Username, Email-ID, Designation, Role | Success message about teacher added | Success message received | Pass |

| | | | | | | |
|---|---|---|---|---|---|---|
| 10 | Add student | Enter personal details of teacher and get success message | Name, Username, Email-ID, Std, Div | Success message about student added | Success message received | Pass |
| 11 | Delete user | Delete user by entering his login details | Username, Password, Role | Success message about user deleted | Success message received | Pass |
| 12 | Deleting user by entering wrong details | Add incorrect login details of user. | Username, Password, Role | Alert message about user not present | Alert message received | Pass |
| 13 | Correct Validation of test details | Enter Test validation details. | Enter Test id , subject and password | Test should be validated Test questions are retrieved | Test is validated. | Pass |
| 14 | Wrong test validation details | Enter wrong test validation details. | Enter wrong Test id , subject and password | Wrong validation alert | Alert message is retrieved. | Pass |
| 15 | Working of next/back button | Click next and back button | Next and back button is clicked | Next or back question should get retrieved. | Next or back question is retrieved. | Pass |
| 16 | Empty Field Alert | Leave all or any one of the textboxes empty | Don't fill any details in the textbox | Empty field alert should be called. | Empty field alert is triggered | Pass |
| 17 | Create new test | Enter the required test details | Enter Test id, subject and password | New test should be created | New test id created. | Pass |
| 18 | Adding questions to test | Enter questions and answers to be added | Enter question and answer to textbox | Test question should get added in the created test | New question should be inserted and new question alert is retrieved. | Pass |
| 19 | Check test scores | Enter the required field details | Enter test id, user id, subject | Scores of required student should be retrieved and stored | Scores of required student is showed. | Pass |
| 20 | No score alert | Enter the required field details | Enter wrong details | Score should not be retrieved | No score found in database ,alert is | Pass |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | triggered. | |
| 21 | Upload notes/ Assignment | Select the required notes/assignment details | Select the file or document | Notes and assignments are uploaded to server. | Notes and assignments are uploaded to server. | Pass |
| 22 | Download notes/assignment | Select the required notes/assignment details | Select the file or document | Notes and assignments are downloaded from server. | Notes and assignments are downloaded from server. | Pass |
| 23 | Firing new query | Enter the query | Enter the query in message box | Query is shown on all connected tablet device | Query is retrieved on each tablet and queries are stored in database | Pass |
| 24 | Taking new quiz | Enter quiz question | Enter the question | Quiz question is shown on students tablet | Quiz question is stored in database ,and shown on students tablets | Pass |
| 25 | Answering quiz | Enter quiz answer | Enter quiz Answer in textbox | Quiz answers are stored in database with each user id | Quiz answers are shown on each connected tablet and stored in database. | Pass |

# CHAPTER 8

# RESULT AND ANALYSIS

# 8. Results & Analysis

The user (teacher or student) will login to the system with their respective username, password and role. After proper validation and login the software with the help of its MVC architecture provides the particular interface to the respective user.

For the classroom mode the query and quiz being uploaded to the server will be made available to the students. For the test mode, the test questions uploaded by the teacher will be made available to the students while the results of the answers submitted by the students are available to the teacher.

# CHAPTER 9

# CONCLUSION AND FUTURE SCOPE

# 9. Conclusion and Future Scope

Thus, we have successfully implemented the software that helps in making the classroom environment friendlier. This software helps in managing the classroom sessions, conducting test, uploading and accessing the files to and from the server.

In the future, this project can be implemented over the entire school thus serving the entire school.

# References

*apache friends - xampp for windows*. (n.d.). Retrieved July 2012, from Apachefriends: http://www.apachefriends.org/en/xampp-windows.html

*Classroom Management Software*. (n.d.). Retrieved from DyKnow: http://www.dyknow.com/company-news/classroom-management-software-whats-under-the-hood/

*Download the ADT Plugin*. (n.d.). Retrieved June 2012, from Developer Android Web Site: http://developer.android.com/sdk/installing/installing-adt.html#Configure

*Installation*. (n.d.). Retrieved March 2012, from Eclipse IDE Web Site: http://wiki.eclipse.org/PDT/Installation

*Wikipedia/Eclipse*. (n.d.). Retrieved March 2012, from Wikipedia: http://en.wikipedia.org/wiki/Eclipse_(software)