SHIVAJI UNIVERSITY

VirtualClassroom

by

Author Name

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Department of Computer Science and Engineering
Sharad Institute of Technology College of Engineering,Yadrav-Ichalkaranji

April 2015

SHIVAJI UNIVERSITY

Abstract

Department of Computer Science and Engineering
Sharad Institute of Technology College of Engineering,Yadrav-Ichalkaranji

Doctor of Philosophy

by Author Name

In the present 21st Century every person needs a viable and easy access to undertake any task. The ease that the present technologies provide saves our lots of time and effort. So adhering to the modern world we have developed Virtual Classroom an Android Application that establishes a network between the mobile device and server. Using this application one can access the on demand video lecture that is stored at server without downloading onto device. This would create a new revolution in the Indian society and would provide a feasible solution to education system problem.

Acknowledgements

We take this opportunity to express our profound gratitude and deep regards to our guide Mrs.M.N.Jadhav for her exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to our Principal, Dr. S. A. Khot and our H.O.D., Prof. O.D.Joshi for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are obliged to all faculty and staff members of all departments of our Sharad Institute of Technology College of Engineering, Yadrav for the valuable information provided by them in their respective fields. We are grateful for their cooperation during the period of our assignment.

Lastly, we thank almighty, our parents, brothers, sisters, friends and our colleagues for their constant encouragement without which this assignment would not be possible.

Miss.Shinde Swatika Raghunath

Miss. Nandgave Akshata Appasaheb

Miss. Patil Komal Subhash

Miss.Sutar Madhuri Manohar

Contents

List of Figures

Chapter 1

INTRODUCTION

1.1 Introduction of project

The Virtual Classroom is a collaborative teaching application to assist the students to leharn in an interactive manner. It aims to complement the efforts of teachers to integrate technology into their classrooms and link the students to the Internet in educationally productive ways and provide them a stimulating, positive and enjoyable environment to study.

Today, most of all colleges use the standard technique for teaching i.e. teachers distributes the knowledge to the students by standing in front of them. This technique requires more time. So this drawback will overcome in this project. Our project is one of the education based tool developed on Android platform. It helps in enhancing the level as well as provides ease in spreading knowledge based information. The mode of retrieving information is Wi-Fi i.e. the connection with the established server.

Using the android application a user can stream the video lectures present in the server with which it is connected. One needs to enter the desired IP address of the server in order to gain access to its content. The only criteria are that the given server is in the range of WiFi network. After establishing the secure connection user can see the list of the video lectures currently present in the server. When a new student or staff register himself for this application he have to register by clicking on new user link. After registration the new user is not activated. For security purpose the student or staff have to be authorized by admin. Authorized staff will able to upload the video lectures onto the server. The another facility provided to staff is record video. While giving lecture he/she is able to start the recording. After recording completed, the staff uploads that video on server. When video is uploaded then that particular video is saved in folder according to standard and subject.

The additional feature of this project is bookmarking a video, by using this feature student can bookmark the video & when next time student logged in, he/she can able to play the bookmarked videos. The videos which are not recorded by staff such videos can also be uploaded by staff. When student log into the application he have to choose particular subject and class. After logging, student is able to see all the video lectures present on server according to subject chosen by him/her. When student select a video then that video starts. The videos which are not used many times can be deleted. The authority of deleting video is only given to admin. The student account as well as staff account can be deleted by admin.

Chapter 2

LITERATURE REVIEW

2.1 Overview of the H.264 /AVC Video Coding Standard in IEEE Transactions on Circuits and Systems for Video Technology

H.264/AVC is newest video coding standard of the ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group. The main goals of the H.264/AVC standardization effort have been enhanced compression performance and provision of a network-friendly video representation addressing conversational (video telephony) and non conversational (storage, broadcast or streaming) applications. H.264/AVC has achieved a significant improvement in rate-distortion efficiency relative to existing standards. This article provides an overview of the technical features of H.264/AVC, describes profiles and applications for the standard, and outlines the history of the standardization process.

2.1.1 The Architecture Design of Streaming Media Applications for Android OS

The imperfection of the Android operating system multimedia function and the complexity of streaming media system of the development cycle are long, and efficiency is low. In order to solve those problems, the thesis designed the streaming engine layer between Linux kernel layer and application framework layer of Android platform, and constructed fast and convenient streaming media application development framework. First of all, Android operating system (OS) was analyzed, and then the media engine was added into the Android OS architecture. Based on the characteristic of the streaming media system, the steaming media engine is divided into five layers, including the user interface, data capture, and data output, codec, and network transmission layer. Through an instance of video data transmission client, the feasibility of the architecture is confirmed.

2.1.2 XML Document Parsing: Operational and Performance Characteristics

XML Parsing starts with character conversion followed by lexical analysis that is invariant among different parsing models with syntactic analysis creates data representation based on parsing model used. Using xml parsing in Virtual Classroom we have included the streaming capability feature. It requires low latency and memory usage and usually parses a small portion of document sequentially without having fully fledged information of entire document structure. [**?**]

2.1.3 Android XML Processing with the Xml Pull Parser Version 1.3

XML stands for Extensible Markup Language and was defined 1998 by the World Wide Web Consortium (W3C). An XML document consists of elements, each element has a start tag, content and an end tag. An XML document must have exactly one root element (i.e. one tag which encloses the remaining tags). XML differentiates between capital and non-capital letters. An XML file is called valid if it is well-formed and if it is contains a link to an XML schema and is valid according to the schema. The Java programming language provides several standard libraries for processing XML files. The SAX and the DOM XML parsers are also available on Android. The SAX and DOM parser API is on Android the same as in standard Java. SAX and DOM have their limitations, therefore it is not recommended to use them on Android. The Java standard provides also the Stax parser. This parser is not part of the Android platform. Android provides for XML parsing and writing the Xml Pull Parser class. This parser is not available in standard Java but is similar to the Stax parser.

Chapter 3

OBJECTIVE AND SCOPE

3.1 Objectives

3.1.1 Providing access to different video lecture that is stored at server without downloading onto device.

Virtual classroom project provides the facility to access different video lectures present on server without downloading it. The video lecture list can be displayed according to branch as well as year of student.

3.1.2 Project provides facility of bookmarking.

The bookmarking facility is nothing but the student is able to see video lectures which he had already seen to certain time interval. After that student may continue to saw same video lecture.

3.1.3 Admin can delete the video lectures and also user accounts.

Admin have authority to delete the video lectures present on server to reduce the load on server. Admin can also delete the user accounts that are passed out.

3.1.4 Providing one option to store the important lectures in the SD card in case there is loss of Wi-Fi connectivity.

If there is loss of Wi-Fi connection in that case user can save the video lectures into the SD card.

3.2 Scope

- User can view the currently active video lectures present in server without downloading it.
- The additional feature of this project is bookmarking a video.
- Staff can upload the video and also record the video.

3.3 Out of Scope

- Support of all video formats on android mobile client.
- Adjusting the video quality and aspect ratio according to network strength.
- User can adjust the speed of the buffered video lecture according to the need.
- Shows the list of related video lecture at the end of current video lecture.

Chapter 4

REQUIREMENT ANALYSIS

4.1 Software Requirements

- JAVA
- Windows OS
- XAMPP
- PHP Script
- MySQL
- Apache Tomcat
- Eclipse

4.2 Hardware Requirements

- Smart Phone
- External Storage
- Wi-Fi Route

JAVA:

1. Description:

   A high-level programming language developed by Sun Microsystems. Java was originally called OAK, and was designed for handheld devices and set-top boxes.

Oak was unsuccessful so in 1995 Sun changed the name to Java and modified the language to take advantage of the burgeoning World Wide Web.

It is a computer programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible.It is intended to let application developers write once, run anywhere (WORA), meaning that code that runs on one platform does not need to be recompiled to run on another. Java applications are typically compiled to byte code that can run on any Java virtual machine (JVM) regardless of computer architecture.

2. JAVA platfrom:

   It is the name given to the computing platform from Oracle that helps users to run and develop Java applications. The platform does not just enable a user to run and develop a Java application, but also features a wide variety of tools that can help developers work efficiently with the Java programming language.

   The platform consists of two essential softwares:

   (a) Java Runtime Environment (JRE)

   (b) Java Development Kit (JDK)

Eclipse:

1. Description:

   Eclipse is an integrated development environment (IDE). It contains a base workspace and an extensible plug-in system for customizing the environment. Written mostly in Java, Eclipse can be used to develop applications. By means of various plug-ins, Eclipse may also be used to develop applications in other programming languages: C, C++, COBOL, Fortran, JavaScript, Natural, Perl and PHP. It can also be used to develop packages for the software Mathematical.

2. Android Development Tools for Eclipse:

   ADT (Android Developer Tools) is a plug-in for Eclipse that provides a suite of tools that are integrated with the Eclipse IDE. It offers you access many features that help you develop Android applications quickly. ADT provides GUI access to many of the command line SDK tools as well as a UI design tool for rapid prototyping, designing, and building of your application's user interface. Because ADT is a plug-in for Eclipse, you get the functionality of a well-established IDE, along with Android-specific features that are bundled with ADT. Developing in Eclipse with ADT is highly recommended and is the fastest way to get started. With the guided project setup it provides, as well as tools integration, custom XML editors, and debug output pane, ADT gives you an incredible boost in developing Android applications.

MYSQL:

1. Description:

   MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed, and supported by MySQL AB, which is a Swedish company. MySQL isbecoming so popular because of many good reasons. MySQL is released under an open-source license. So you have nothing to pay to use it. MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages. MySQL uses a standard form of the well-known SQL data language. MySQL works on many operating systems and with many laguages including PHP, PERL, C, C++, JAVA, etc. MySQL works very quickly and works well even with large data sets. MySQL is very friendly to PHP, the most appreciated language for web development. MySQL supports large databases, up to 50 million rows or more in a table. The default size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB). MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to their own specific environments.

PHP Script:

1. Description:

   PHP is a program that gets installed on top of your web server software. It works with versions of Apache , Microsoft IIS and other server software packages. We use PHP by inserting PHP code inside the HTML that makes up your website. When a client visits a web page that contains this code, your server executes it. Thats why you need to install your own server in order to test PHP locally the server is the brain here, not your browser. Users dont need any special plug-ins or anything to see your PHP in action it gets to the end user as regular old-fashioned HTML. PHP is a scripting language, like HTML. That means that code does not need to be compiled before it gets used it gets processed on the fly as necessary. PHP is an open-source language, and PHP.net is its control center, with extensive reference material about the language and tips sent in by users across the globe. PHP.net has exceptional, deep information about the language, but it can be a little cryptic for the newcomer.

XAMPP:

1. Description:

   XAMPP requires only one zip, tar, 7z, or exe file to be downloaded and run, and little or no configuration of the various components that make up the web server is required. XAMPP is regularly updated to incorporate the latest releases of Apache, MySQL, PHP and Perl. It also comes with a number of other modules including OpenSSL and phpMyAdmin.Self-contained, multiple instances of XAMPP can exist on a single computer, and any given instance can be copied from one computer to another. Officially, XAMPP's designers intended it for use only as a development tool, to allow website designers and programmers to test their work on their own computers without any access to the Internet. To make this as easy as possible, many important security features are disabled by default. In practice, however, XAMPP is sometimes used to actually serve web pages on the World Wide Web. A special tool is provided to password-protect the most important parts of the package. XAMPP also provides support for creating and manipulating databases in MySQL and SQLite among others. Once XAMPP is installed, it is possible to treat a localhost like a remote host by connecting using an FTP client. Using a program like FileZilla has many advantages when installing a content management system (CMS) like Joomla or WordPress. It is also possible to connect to localhost via FTP with an HTML editor.

Chapter 5

SYSTEM DESIGN

5.1 Use Case Diagram

Description:

A use case diagram is the simplest representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can define the different types of users of a system and the various ways that they interact with the system. They provide the simplified and graphical representation of what the system must actually do.The purpose of the use case diagrams is simply to provide the high level view of the system and convey the requirements.

5.1.1 Use-Case Diagram

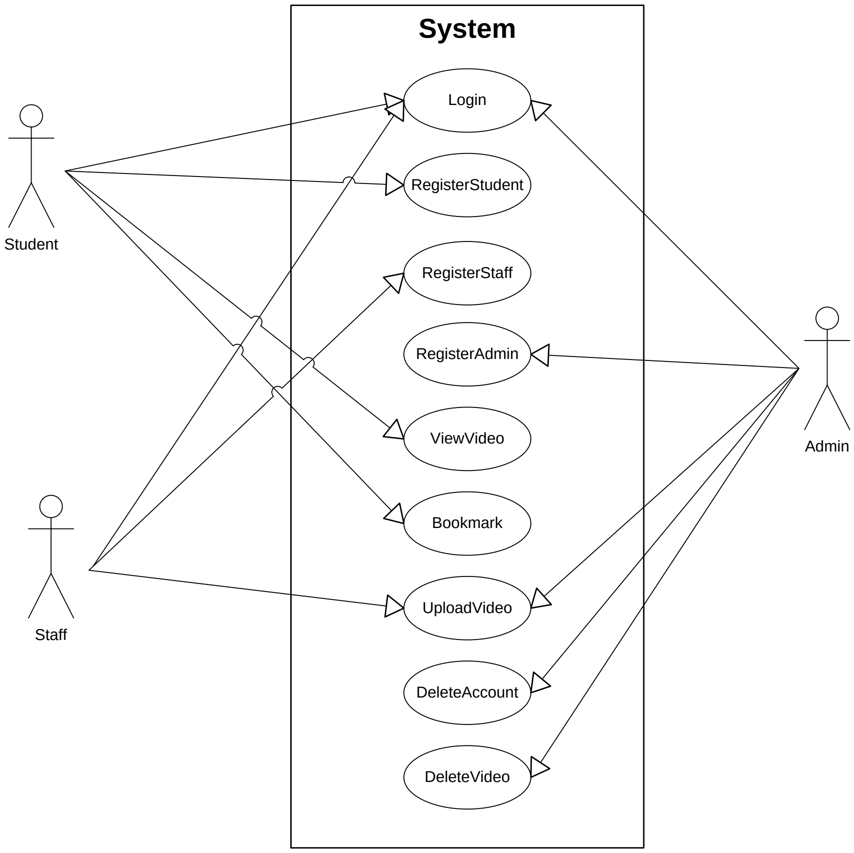
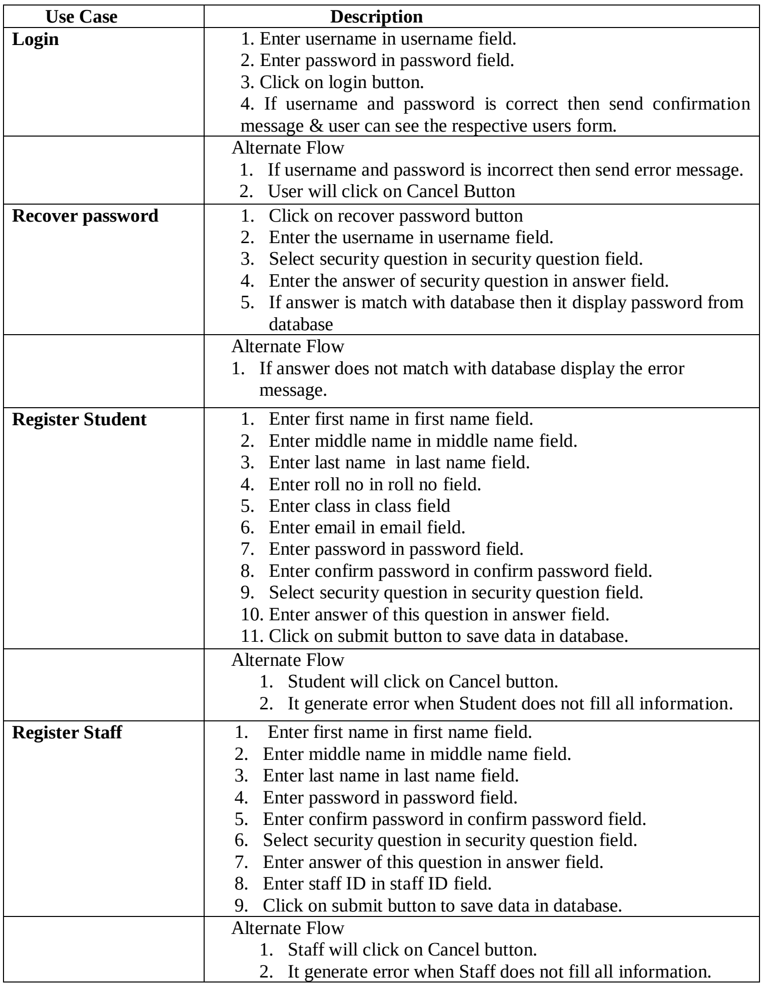


FIGURE 5.1: Use case diagram for Virtual Classroom.

5.1.2 Use-Case Scenario



Use Case	Description
Login	1. Enter username in username field. 2. Enter password in password field. 3. Click on login button. 4. If username and password is correct then send confirmation message & user can see the respective users form.
	Alternate Flow 1. If username and password is incorrect then send error message. 2. User will click on Cancel Button
Recover password	1. Click on recover password button 2. Enter the username in username field. 3. Select security question in security question field. 4. Enter the answer of security question in answer field. 5. If answer is match with database then it display password from database
	Alternate Flow 1. If answer does not match with database display the error message.
Register Student	1. Enter first name in first name field. 2. Enter middle name in middle name field. 3. Enter last name in last name field. 4. Enter roll no in roll no field. 5. Enter class in class field 6. Enter email in email field. 7. Enter password in password field. 8. Enter confirm password in confirm password field. 9. Select security question in security question field. 10. Enter answer of this question in answer field. 11. Click on submit button to save data in database.
	Alternate Flow 1. Student will click on Cancel button. 2. It generate error when Student does not fill all information.
Register Staff	1. Enter first name in first name field. 2. Enter middle name in middle name field. 3. Enter last name in last name field. 4. Enter password in password field. 5. Enter confirm password in confirm password field. 6. Select security question in security question field. 7. Enter answer of this question in answer field. 8. Enter staff ID in staff ID field. 9. Click on submit button to save data in database.
	Alternate Flow 1. Staff will click on Cancel button. 2. It generate error when Staff does not fill all information.

Figure 5.2: Use Case Scenario Table.

5.2 Sequence Diagram

A Sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

5.2.1 Sequence Diagrams for Login User:

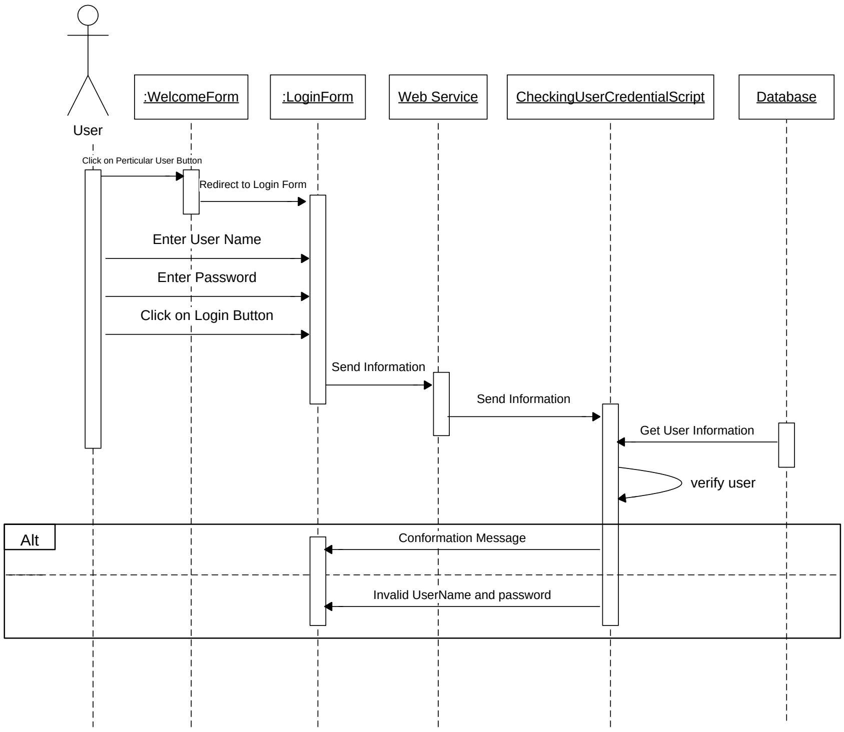


Figure 5.3: sequence diagram for User Login

Above diagram shows the sequence of activities carried out when login is done. Using the above form student, staff as well as admin is able to login.

5.2.2 Sequence Diagrams for Recover Password:

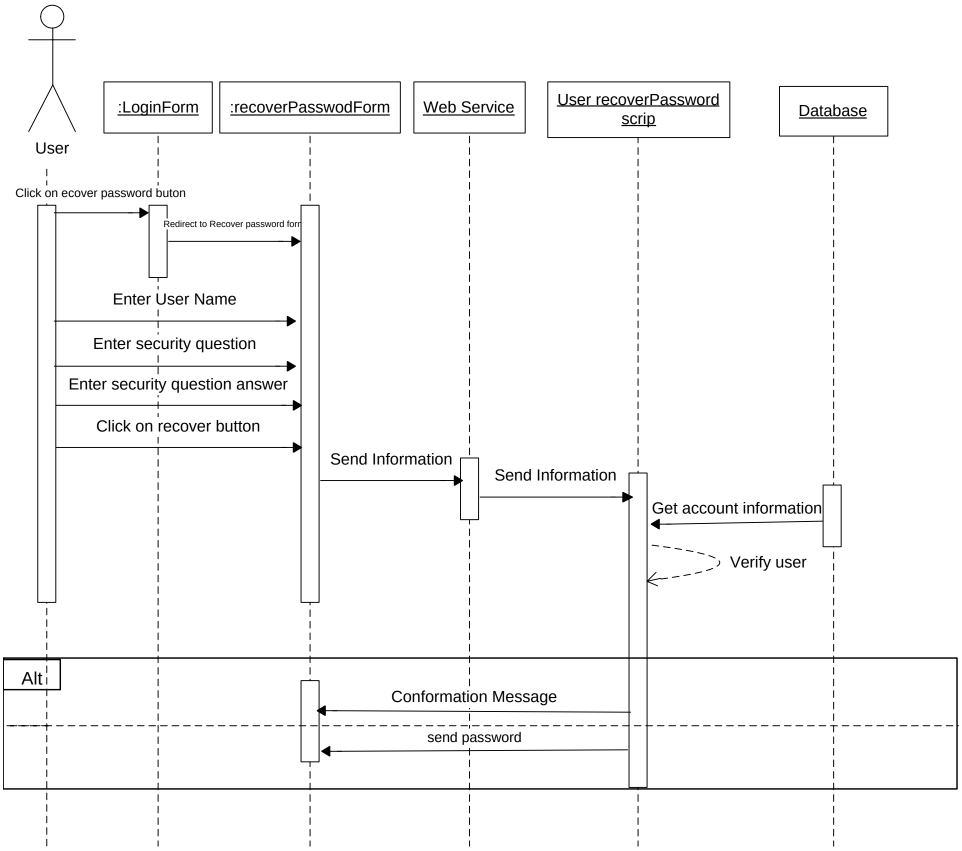


Figure 5.4: sequence diagram for Recover Password

Above diagram shows sequence diagram for recover password. To recover password user choose security and answer. The answer given by user is compared with database. If answer matches then password is sent to user.

5.2.3 Sequence Diagrams for Register Admin:

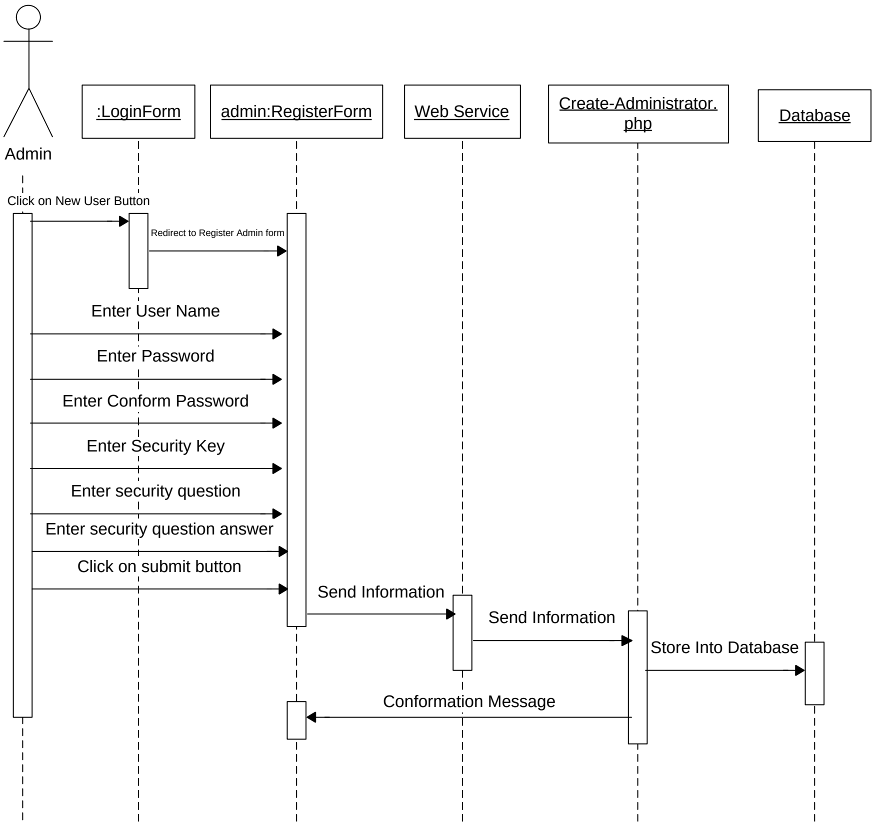


FIGURE 5.5: sequence diagram for Admin Registration.

Above diagram shows sequence diagram for admin registration. Admin gives required information as well as key. The username and password is used for further application. The whole data is saved into database.

5.2.4 Sequence Diagrams for Register Staff:

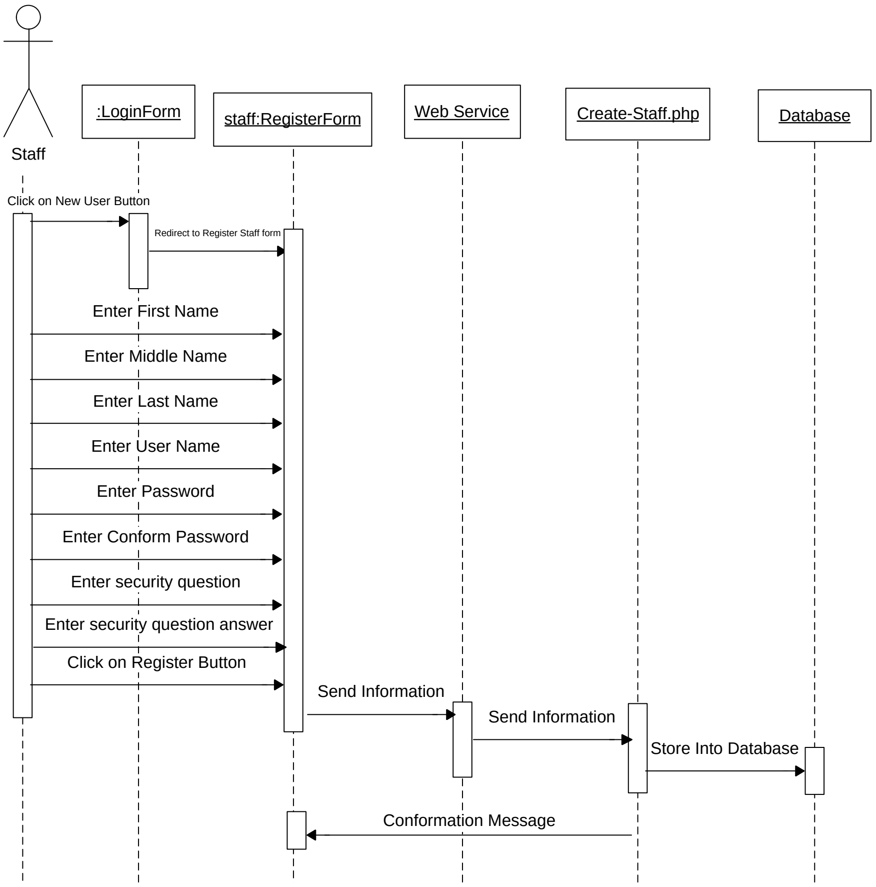


FIGURE 5.6: sequence diagram for Staff Registration.

Above diagram shows sequence diagram for staff registration. Staff gives required information in form. The username and password is used for further application. The whole data is saved into database.

5.2.5 Sequence Diagrams for Register Student:

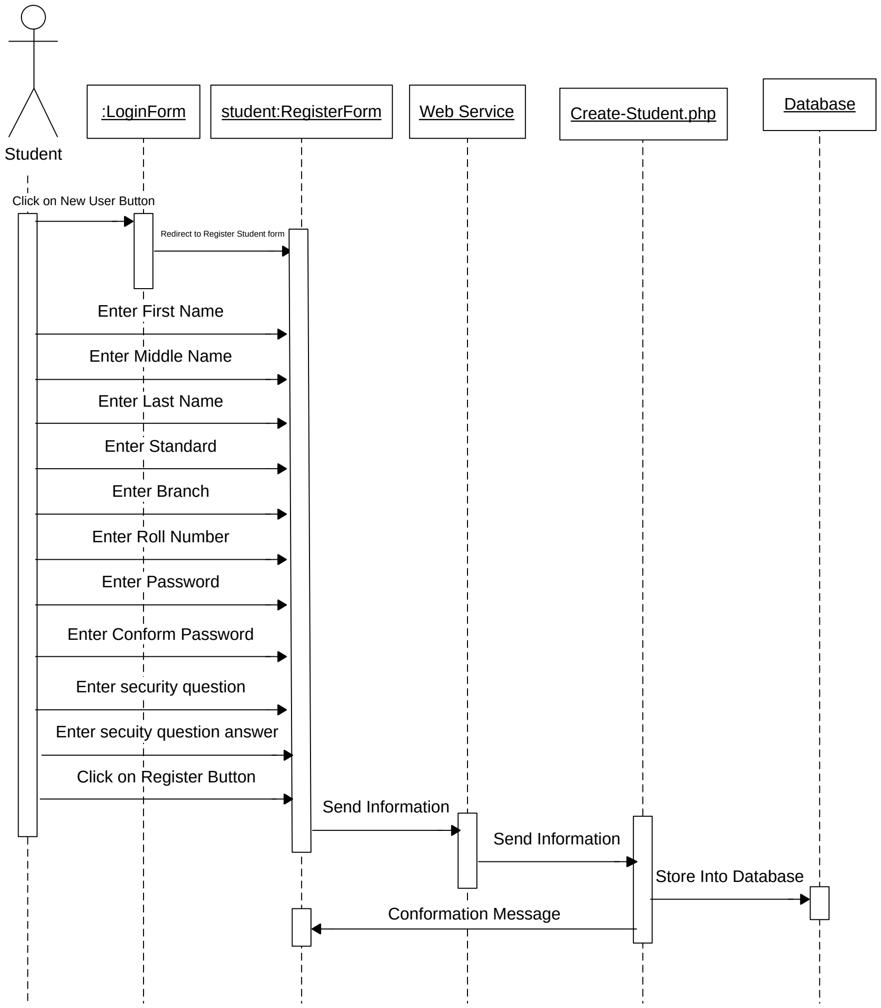


FIGURE 5.7: sequence diagram for Student Registration.

Above diagram shows sequence diagram for student registration. Student gives required information in form. The username and password is used for further application. The whole data is saved into database.

5.2.6 Sequence Diagrams for View Video:

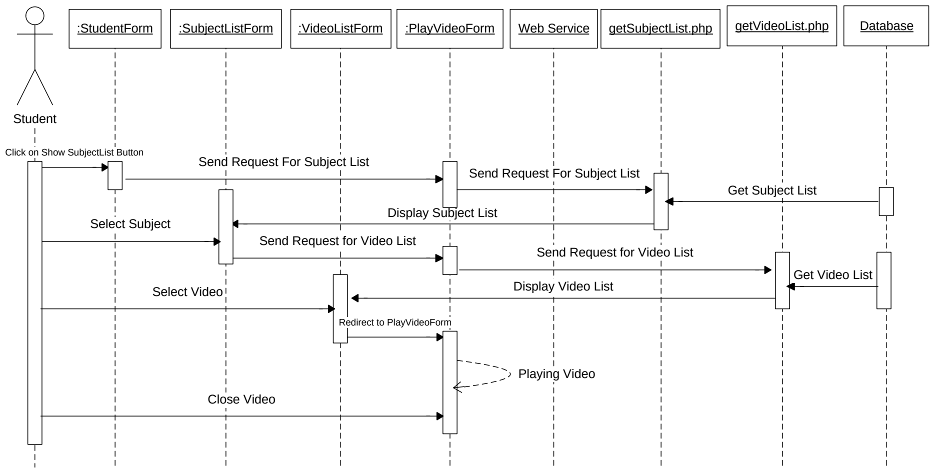


Figure 5.8: sequence diagram for View Video.

The above figure indicates the sequence diagram for view video. When authorized student wants videos then first he/she get list of videos. By choosing the particular video, he/she can able to do the pause video, resume video or closing video.

5.2.7 Sequence Diagrams for Bookmark Video:

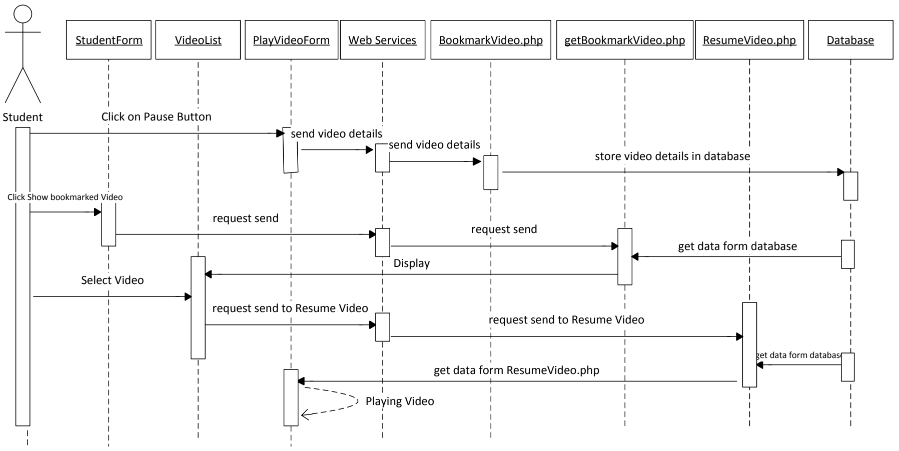


Figure 5.9: sequence diagram for Bookmark Video.

The above figure indicates the sequence diagram for Bookmark video. When authorized student pause the video then remaining time and pause time is stored into the bookmark history database. When next time student login, he/she can able to resume previously paused video.

5.2.8 Sequence Diagrams for Upload Video:

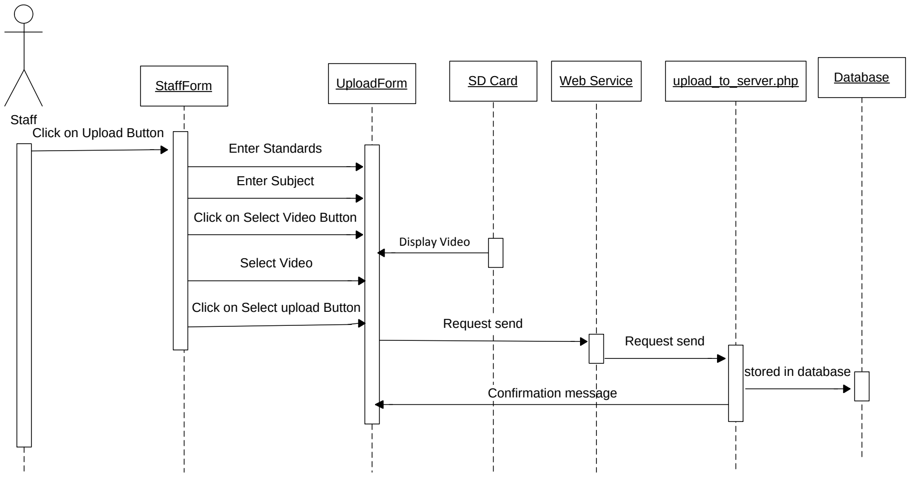


FIGURE 5.10: sequence diagram for Upload Video.

The above figure indicates the sequence diagram for Upload video. When staff get login the system he can upload the video. The uploaded video is saved into database.

5.2.9 Sequence Diagrams for Record Video:

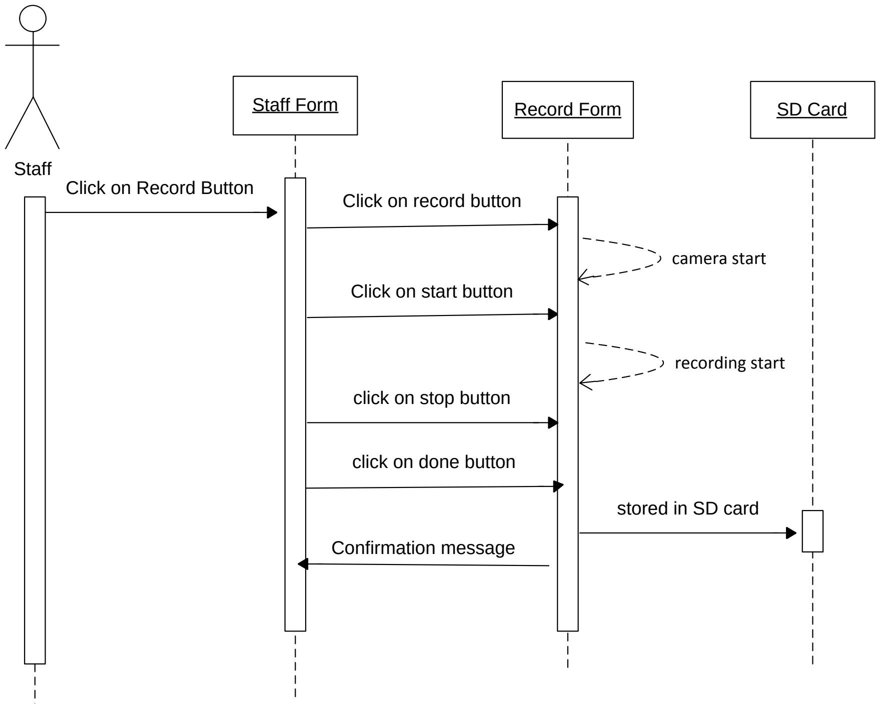


FIGURE 5.11: sequence diagram for Record Video.

The above figure indicates the sequence diagram for record video.When staff get login the system he can record the video.The staff can also upload the recorded video.The recorded video is saved into database.

5.2.10 Sequence Diagrams for Delete Video:

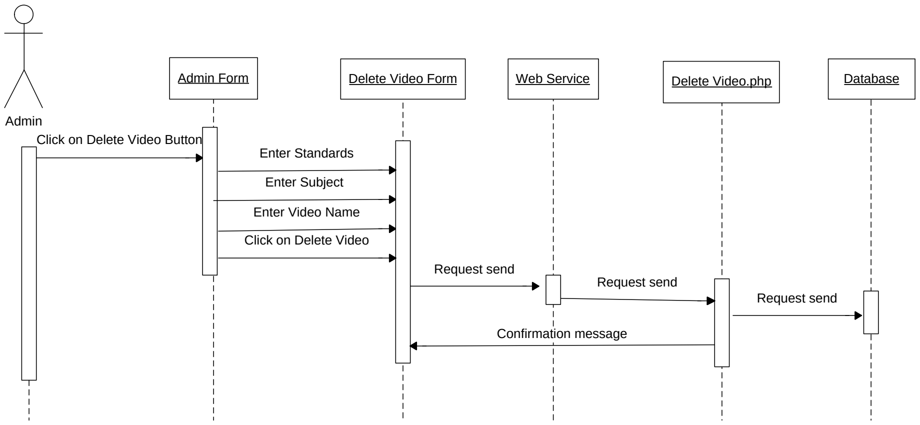


Figure 5.12: sequence diagram for Delete Video.

The above figure indicates the sequence diagram for delete video. Admin has authority to delete video from database.

5.3 Class Diagram

A UML class diagram describes the object and information structures used by your application, both internally and in communication with its users. It describes the information without reference to any particular implementation. Its classes and relationships can be implemented in many ways, such as database tables, XML nodes, or compositions of software objects.

Components of class diagram:

1. Class:

   A definition of objects that share given structural or behavioral characteristics.

2. Attribute:

   A typed value attached to each instance of a classifier.

3. Operation:

   A method or function that can be performed by instances of a classifier.

5.3.1 Class Diagram:

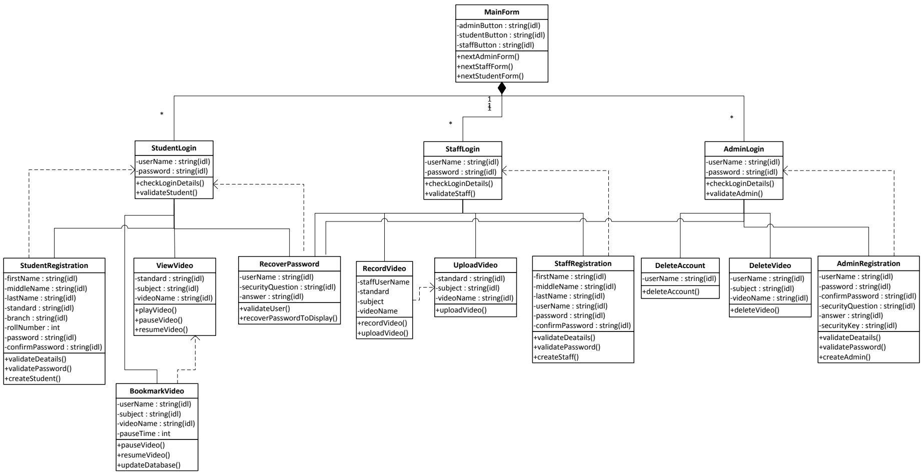


FIGURE 5.13: Class diagram for Virtual Classroom.

5.4 Deployment Diagram:

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.To describe a web site, for example, a deployment diagram would show what hardware components (nodes) exist (e.g., a web server, an application server, and a database server), what software components (artifacts) run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI). The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

There are two types of Nodes:

1. Device Node.
2. Execution Environment Node.

Device nodes are physical computing resources with processing memory and services to execute software, such as typical computers or mobile phones. An execution environment node (EEN) is a software computing resource that runs within an outer node and which itself provides a service to host and execute other executable software elements

5.4.1 Deployment Diagram:

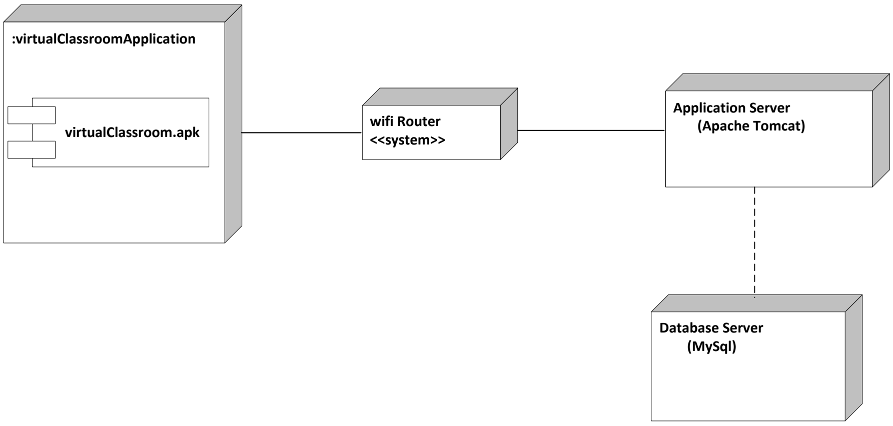


Figure 5.14: Deployment diagram for Virtual Classroom.

Chapter 6

Coading

6.1 Introduction of tools and Installation

6.1.1 Android

There's no other software quite like Android. Google engineered Android, and Google own apps run better on it. And with millions of apps, games, songs, and videos on Google Play, Android is great for fun, and for getting things done.

Android devices come in all kinds of sizes, with all sorts of features, and in all sorts of prices. Each version of Android is named after a dessert, and the most recent version of Android is lolipop. With Android, youre in control of your mobile experience.

The world is contracting with the growth of mobile phone technology. As the number of users is increasing day by day, facilities are also increasing. Starting with simple regular handsets which were used just for making phone calls, mobiles have changed our lives and have become part of it. Now they are not used just for making calls but they have innumerable uses and can be used as a Camera , Music player, Tablet PC, T.V. , Web browser etc. . And with the new technologies, new software and operating systems are required.

- What is android

Operating Systems have developed a lot in last 15 years. Starting from black and white phones to recent smart phones or mini computers, mobile OS has come far away. Especially for smart phones, Mobile OS has greatly evolved from Palm OS in 1996 to Windows pocket PC in 2000 then to Blackberry OS and Android.

- ADT Bundle

The Android SDK is a software development kit which provides API libraries and necessary developer tools necessary for building Android applications. Android SDK is officially provided by android developers.

steps for the installation and set-up of Android development environment:

1. Download Eclipse
2. Download JDK and install it, set the environment path.
3. Download ADT plugin inside Eclipse.
4. Set the Preference with Android-SDK path.
5. Download the latest platform-tools and everything.

The ADT Bundle includes everything you need to begin developing apps:

1. Eclipse + ADT plugin
2. Android SDK Tools
3. Android Platform-tools
4. The latest Android platform
5. The latest Android system image for the emulator

Yes there are also possible ways if you want to use existing version of Eclipse or any other IDE.

- **Setting Up the ADT Bundle:**

As you have downloaded ADT bundle, follow below steps to setup it:

1. Unpack the ZIP file named “adt bundle osplatform.zip ” and save it to an appropriate location such as a “Development” directory in your home directory.
2. Open the adt bundle osplatform goto eclipse and next directory and launch eclipse.

6.1.2 XAMPP Installation Steps

1. Download the software from apachefriend website.

   Select the Installer option under the Basic Package. You may be taken to a page that presents you with a bunch of different download locations. Just click one of the download buttons, and then save the file to your desktop. Once downloaded, the installer works like most Windows installers.

2. In Internet Explorer, you may get a warning about downloading the file. Click the yellow information bar that appears above the Web page in IE, and choose Download File

3. Double-click the .exe file you downloaded.

   A window opens, asking you to select the language youd like to use.

   If a warning dialog appears click the "Allow" option to install XAMPP.

4. Choose a language from the menu, and then click OK.

   A Setup Wizard window appears, ready to step you through the setup process.

5. Click the Next button.

   The installer suggests putting the application on your main drive at C, You can pretty much install it anywhere.

6. Click the Next button once again.

   The XAMPP Options window appears. In most cases, its fine to leave all the windows checkboxes just as you see.

7. Click Install.

   The installer places all the files onto your system. This process takes a while, since a lot of programs and files are being installed.

8. Finally, click the Finish button.

   A window appears "congratulating" you way to double-click the installer program and asking whether you wish to start the XAMPP Control panel.

9. Click Yes, to open the XAMPP Control Panel .

   The XAMPP Control Panel lets you start and stop the Apache Web server and MySQL database server.

10. If the buttons to the right of Apache and MySQL say Start, click them to start the Web server and the MySQL database server.

11. To do so, launch a Web browser, and, in the Location bar and type localhost.

6.1.3 PHP

PHP (recursive acronym for **PHP:** *Hypertext Preprocessor*) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.

PHP is probably the most popular scripting language on the web. It is used to enhance web pages. With PHP, you can do things like create a username and password login pages, check details from a form, create forums, picture galleries, surveys, and a whole lot more.

PHP is known as a server-sided language. That's because the PHP doesn't get executed on your computer, but on the computer you requested the page from. The results are then handed over to you.

The most popular explanation of just what PHP stands for is "Hypertext Pre-processor". But that would make it HPP, surely? An alternative explanation is that the initials come from the earliest version of the program, which was called Personal Home Page Tools. At least you get the letters "PHP" in the right order!

- **What is a PHP File**
  - PHP files can contain text, HTML, CSS, JavaScript, and PHP code
  - PHP code is executed on the server, and the result is returned to the browser as plain HTML
  - PHP files have extension ". PHP"
- **What Can PHP Do**
  - PHP can generate dynamic page content
  - PHP can create, open, read, write, and close files on the server
  - PHP can send and receive cookies
  - PHP can add, delete, modify data in your database.
  - With PHP you are not limited to output HTML. You can output images, PDF files, and even flash movies. You can also output any text, such as XHTML and XML.
- **Why PHP**
  - PHP runs on various platforms like Windows, Linux, Unix, Mac OS X, etc.
  - PHP is compatible with almost all servers used today Apache, Apache Tomcat,IIS, etc.

  - PHP supports a wide range of databases
  - PHP is free. Download it from the official PHP resource website.

6.1.4 MySQL

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. The MySQL official web site www.mysql.com provides the latest information about MySQL software.

- MySQL is a database management system.

  A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- MySQL databases are relational.

  A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and pointers between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data. The SQL part of “MySQL” stands for “Structured Query Language”. SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly for example, to generate reports, embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax.

- MySQL software is Open Source.

  Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL for GNU General Public License, to define what you may and may not do with the software in different situations.

- The MySQL Database Server is fast, reliable, scalable and easy to use.

  MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and IO capacity available. MySQL can also scale up to clusters of machines, networked together. MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

6.2 Snippets

Login Form:

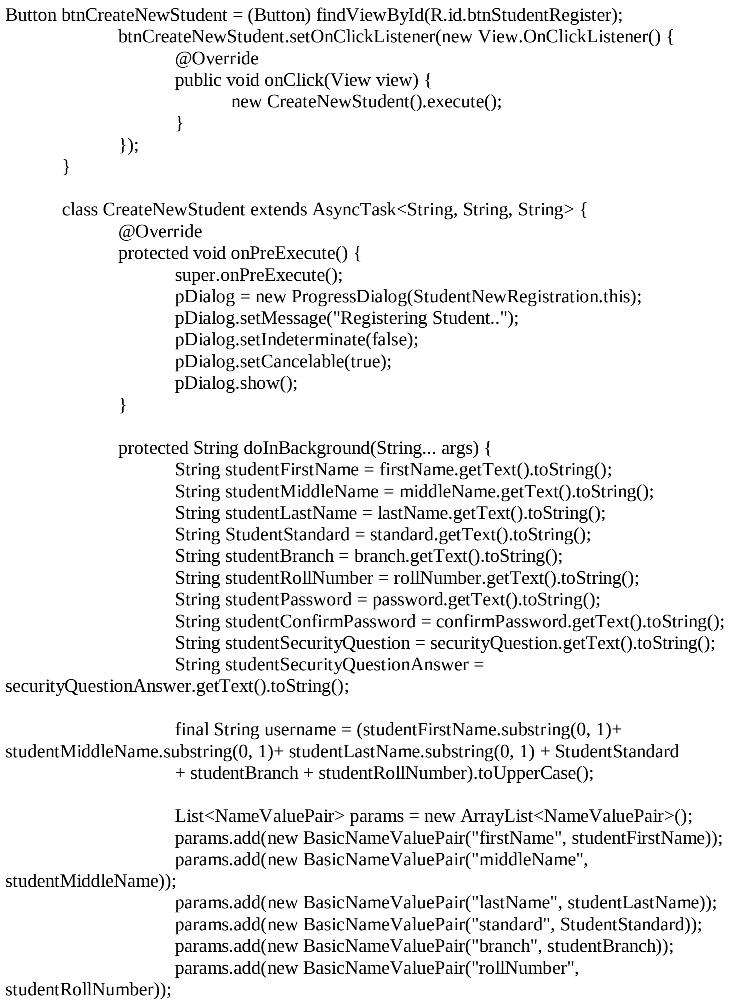
```
import java.util.ArrayList;
import java.util.List;
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONException;
import org.json.JSONObject;
import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.AsyncTask;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import com.example.virtualclassroomproject.R;

public class StudentLogin extends Activity {
	public static final String STUDENT_PREFERENCE="student";
	private ProgressDialog pDialog;
	JSONParser jsonParser = new JSONParser();
	EditText username;
	EditText password;
	private static String url_checkValidStudent = 
"http://10.0.2.2/VirtualClassroom/check_valid_student.php";
	private static final String TAG_SUCCESS = "success";

	@Override
	public void onCreate(Bundle savedInstanceState) {
		super.onCreate(savedInstanceState);
		setContentView(R.layout.student_login);
		username = (EditText) findViewById(R.id.studentLoginUsername);
		password = (EditText) findViewById(R.id.studentLoginPassword);

		Button btnNewStudentRegistration = (Button) 
findViewById(R.id.btnLinkToNewStudentRegistration);
		btnNewStudentRegistration.setOnClickListener(new View.OnClickListener() {

					@Override
					public void onClick(View v) {
						Intent i = new 
Intent(getApplicationContext(),StudentNewRegistration.class);
```

```
    startActivity(i);
                    }
                });

        Button btnCheckStudent = (Button) findViewById(R.id.btnStudentLogin);
        btnCheckStudent.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                new CheckValidStudent().execute();
            }
        });

        Button
btnStudentForgetPassword=(Button)findViewById(R.id.btnLinkToStudentForgetPassword);
        btnStudentForgetPassword.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                Intent i = new
Intent(getApplicationContext(),StudentForgetPassword.class);
                startActivity(i);

            }
        });
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (resultCode == 100) {
            Intent intent = getIntent();
            finish();
            startActivity(intent);
        }
    }

    class CheckValidStudent extends AsyncTask<String, String, String> {
        @Override
        protected void onPreExecute() {
            super.onPreExecute();
            pDialog = new ProgressDialog(StudentLogin.this);
            pDialog.setMessage("Checking credentials. Please wait...");
            pDialog.setIndeterminate(false);
            pDialog.setCancelable(false);
            pDialog.show();
        }
```

```
protected String doInBackground(String... args) {
            String adminUsername = username.getText().toString();
            String adminPassword = password.getText().toString();
            List<NameValuePair> params = new ArrayList<NameValuePair>();
            params.add(new BasicNameValuePair("username", adminUsername))
            params.add(new BasicNameValuePair("password", adminPassword));
            JSONObject json = jsonParser.makeHttpRequest(url_checkValidStude
    "POST", params);
            Log.d("Create Response", json.toString());

            try {
                int success = json.getInt(TAG_SUCCESS);
                if (success == 1) {

            Intent i = new
Intent(getApplicationContext(),StudentRegistrationSuccessful.class);

            SharedPreferences settings=
getSharedPreferences(STUDENT_PREFERENCE, 0);
    SharedPreferences.Editor editor= settings.edit();

    String editedStandard= username.getText().toString();
    String standard= editedStandard.substring(3, 5);

    String editedBranch= username.getText().toString();
    String branch= editedBranch.substring(5, 8);

    editor.putString("standard", standard);
    editor.putString("branch", branch);
    editor.putString("username", username.getText().toString());
    editor.commit();
            startActivity(i);
            finish();
            }
            else {
                    runOnUiThread(new Runnable() {
                        public void run() {
                            Toast.makeText(getApplicationContext(
                                    "Invalid Student. Please
check details",Toast.LENGTH_LONG).show();
                        }
                    });

                }
            } catch (JSONException e) {
                e.printStackTrace();
}
            return null;
        }
        protected void onPostExecute(String file_url) {
            pDialog.dismiss();

        }
    }
}
```

Registration Form:

```
public class StudentNewRegistration extends Activity {

    private ProgressDialog pDialog;
    JSONParser jsonParser = new JSONParser();
    EditText firstName;
    EditText middleName;
    EditText lastName;
    EditText standard;
    EditText branch;
    EditText rollNumber;
    EditText password;
    EditText confirmPassword;
    EditText securityQuestion;
    EditText securityQuestionAnswer;
    private static String url_create_student = 
"http://10.0.2.2/virtualClassroom/create_student.php";
    private static final String TAG_SUCCESS = "success";

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.student_registration);

        firstName = (EditText) findViewById(R.id.studentFirstName);
        middleName = (EditText) findViewById(R.id.studentMiddleName);
        lastName = (EditText) findViewById(R.id.studentLastName);
        standard = (EditText) findViewById(R.id.studentStandard);
        branch = (EditText) findViewById(R.id.studentBranch);
        rollNumber = (EditText) findViewById(R.id.studentRollNumber);
        password = (EditText) findViewById(R.id.studentPassword);
        confirmPassword = (EditText) findViewById(R.id.StudentConfirmPassword)
        securityQuestion = (EditText) findViewById(R.id.studentSecurityQuestion);
        securityQuestionAnswer = (EditText) findViewById(R.id.studentAnswer);
```



```
Button btnCreateNewStudent = (Button) findViewById(R.id.btnStudentRegister);
                btnCreateNewStudent.setOnClickListener(new View.OnClickListener() {
                        @Override
                        public void onClick(View view) {
                                new CreateNewStudent().execute();
                        }
                });
        }

        class CreateNewStudent extends AsyncTask<String, String, String> {
                @Override
                protected void onPreExecute() {
                        super.onPreExecute();
                        pDialog = new ProgressDialog(StudentNewRegistration.this);
                        pDialog.setMessage("Registering Student..");
                        pDialog.setIndeterminate(false);
                        pDialog.setCancelable(true);
                        pDialog.show();
                }

                protected String doInBackground(String... args) {
                        String studentFirstName = firstName.getText().toString();
                        String studentMiddleName = middleName.getText().toString();
                        String studentLastName = lastName.getText().toString();
                        String StudentStandard = standard.getText().toString();
                        String studentBranch = branch.getText().toString();
                        String studentRollNumber = rollNumber.getText().toString();
                        String studentPassword = password.getText().toString();
                        String studentConfirmPassword = confirmPassword.getText().toString();
                        String studentSecurityQuestion = securityQuestion.getText().toString();
                        String studentSecurityQuestionAnswer = 
securityQuestionAnswer.getText().toString();

                        final String username = (studentFirstName.substring(0, 1)+ 
studentMiddleName.substring(0, 1)+ studentLastName.substring(0, 1) + StudentStandard
                        + studentBranch + studentRollNumber).toUpperCase();

                        List<NameValuePair> params = new ArrayList<NameValuePair>();
                        params.add(new BasicNameValuePair("firstName", studentFirstName));
                        params.add(new BasicNameValuePair("middleName", 
studentMiddleName));
                        params.add(new BasicNameValuePair("lastName", studentLastName));
                        params.add(new BasicNameValuePair("standard", StudentStandard));
                        params.add(new BasicNameValuePair("branch", studentBranch));
                        params.add(new BasicNameValuePair("rollNumber", 
studentRollNumber));
```

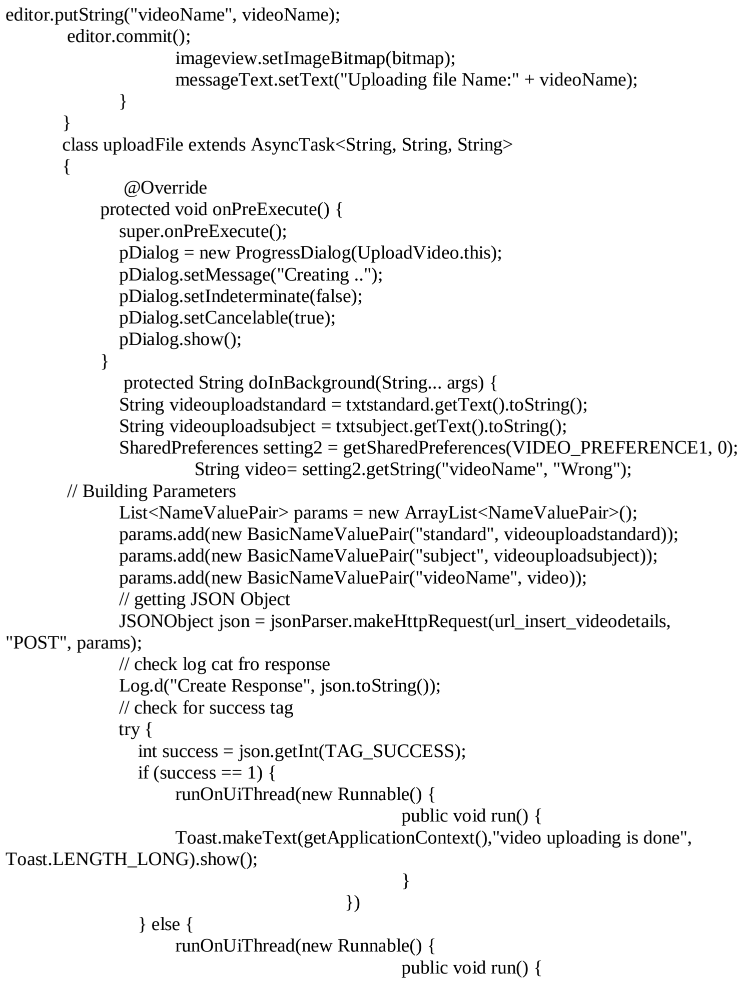
```
        params.add(new BasicNameValuePair("password", studentPassword));
                        params.add(new
BasicNameValuePair("confirmPassword",studentConfirmPassword));
                        params.add(new
BasicNameValuePair("securityQuestion",studentSecurityQuestion));
                        params.add(new BasicNameValuePair("securityQuestionAnswer",
        studentSecurityQuestionAnswer));
                        params.add(new BasicNameValuePair("username", username));
                        params.add(new BasicNameValuePair("status", "Disabled"));

                        JSONObject json =
jsonParser.makeHttpRequest(url_create_student,"POST", params);
                        Log.d("Create Response", json.toString());
                        try {
                                int success = json.getInt(TAG_SUCCESS);
                                if (success == 1) {

                                        runOnUiThread(new Runnable() {
                                                public void run() {
                                                        Toast.makeText(getApplicationContext(),
                                                                        "Student Registration is done
Successfully", Toast.LENGTH_LONG).show();
                                                        Toast.makeText(getApplicationContext(),
                                                                        "Your Username
is:"+username.toString(), Toast.LENGTH_LONG).show();
                                                        Intent i = new
Intent(getApplicationContext(),StudentLogin.class);
                                                        startActivity(i);
                                                        finish();
                                                }
                                        });
                                } else {
                                        runOnUiThread(new Runnable() {
                                                public void run() {
                                                        Toast.makeText(getApplicationContext(),
                                                                        "Erro to Register Student",
Toast.LENGTH_LONG).show();
                                        }
                                        });
                                }
                        } catch (JSONException e) {
                                e.printStackTrace();
                        }
                        return null;
                }
                protected void onPostExecute(String file_url) {
```

```
pDialog.dismiss();
                }
        }
}
```

Upload Video:

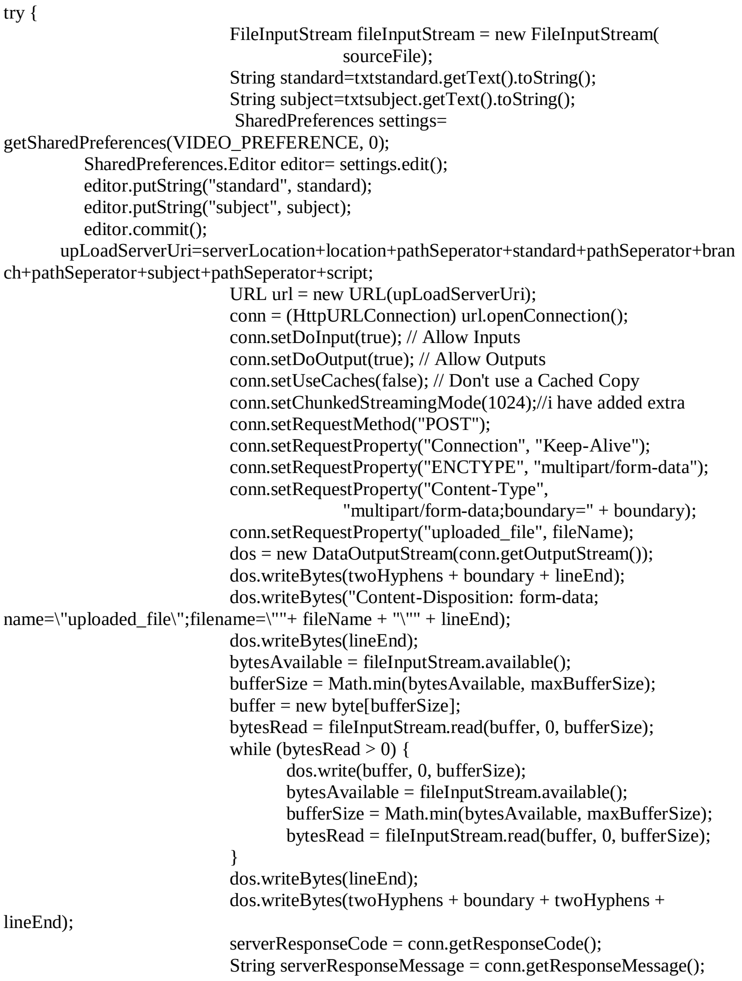
```
public class UploadVideo extends Activity implements OnClickListener{
        public static final String VIDEO_PREFERENCE="video";
        public static final String VIDEO_PREFERENCE1="video1";
        String fileName=null;
        private ProgressDialog pDialog;
        JSONParser jsonParser = new JSONParser();
        private TextView messageText;
        private Button uploadButton, btnselectvideo;
        private ImageView imageview;
        private int serverResponseCode = 0;
        private ProgressDialog dialog = null;
        private String upLoadServerUri = null;
        private String filepath = null;
        int FLAG = 0;
private static final String TAG_SUCCESS = "success";
        //uploading details
        String serverLocation="http://10.0.2.2/";
        String pathSeperator="/";
        String folder="VirtualClassroom";
        private static String url_insert_videodetails = 
"http://10.0.2.2/virtualClassroom/create_video_entry.php";
        EditText txtstandard;
        EditText txtsubject;
        String script="upload_to_server.php";
        String location="VirtualClassroom";
        String branch="CSE";

@Override
        public void onCreate(Bundle savedInstanceState) {
                super.onCreate(savedInstanceState);
                setContentView(R.layout.uploadvideo);
                txtstandard=(EditText)findViewById(R.id.videouploadStandard);
                txtsubject=(EditText)findViewById(R.id.videouploadSubject);
                uploadButton = (Button) findViewById(R.id.uploadButton);
                messageText = (TextView) findViewById(R.id.messageText);
                btnselectvideo = (Button) findViewById(R.id.button_selectvideo);
                imageview = (ImageView) findViewById(R.id.imageView_pic);
                btnselectvideo.setOnClickListener(this);
                uploadButton.setOnClickListener(this);
```

```
}
    @Override
    public void onClick(View arg0) {
        if (arg0 == btnselectvideo) {
            FLAG = 1;
            Intent intent = new Intent();
            intent.setType("video/*");
            intent.setAction(Intent.ACTION_GET_CONTENT);
            startActivityForResult(
                    Intent.createChooser(intent, "Complete action using"), 1);
            if (btnselectvideo.isPressed()) {
                Drawable bitmap =
getResources().getDrawable(R.drawable.video);
                imageview.setImageDrawable(bitmap);
            }
        } else if (arg0 == uploadButton) {
            if (FLAG == 0) {
                Toast.makeText(UploadVideo.this,"Please select video
!!!",Toast.LENGTH_LONG).show();
            } else {
                if (filepath != null) {
                    dialog = ProgressDialog.show(UploadVideo.this,
"","Uploading file...", true);
                    messageText.setText("uploading started.....");
                    new Thread(new Runnable() {
                        public void run() {
                            uploadFile(filepath);
                        }
                    }).start();
                } else {
                    Toast.makeText(UploadVideo.this, "Please try again !!!",
                            Toast.LENGTH_LONG).show();
                }
            }
        }
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        if (requestCode == 1 && resultCode == RESULT_OK) {
            Uri selectedImageUri = data.getData();
            filepath = getPath(selectedImageUri);
            Bitmap bitmap = BitmapFactory.decodeFile(filepath);
            String videoName=filepath.substring(12);
             SharedPreferences settings=
getSharedPreferences(VIDEO_PREFERENCE1, 0);
     SharedPreferences.Editor editor= settings.edit();
```



```
editor.putString("videoName", videoName);
         editor.commit();
                         imageview.setImageBitmap(bitmap);
                         messageText.setText("Uploading file Name:" + videoName);
                 }
        }
        class uploadFile extends AsyncTask<String, String, String>
        {
                 @Override
              protected void onPreExecute() {
                 super.onPreExecute();
                 pDialog = new ProgressDialog(UploadVideo.this);
                 pDialog.setMessage("Creating ..");
                 pDialog.setIndeterminate(false);
                 pDialog.setCancelable(true);
                 pDialog.show();
              }
                 protected String doInBackground(String... args) {
                 String videouploadstandard = txtstandard.getText().toString();
                 String videouploadsubject = txtsubject.getText().toString();
                 SharedPreferences setting2 = getSharedPreferences(VIDEO_PREFERENCE1, 0);
                            String video= setting2.getString("videoName", "Wrong");
         // Building Parameters
                 List<NameValuePair> params = new ArrayList<NameValuePair>();
                 params.add(new BasicNameValuePair("standard", videouploadstandard));
                 params.add(new BasicNameValuePair("subject", videouploadsubject));
                 params.add(new BasicNameValuePair("videoName", video));
                 // getting JSON Object
                 JSONObject json = jsonParser.makeHttpRequest(url_insert_videodetails,
"POST", params);
                 // check log cat fro response
                 Log.d("Create Response", json.toString());
                 // check for success tag
                 try {
                    int success = json.getInt(TAG_SUCCESS);
                    if (success == 1) {
                         runOnUiThread(new Runnable() {
                                                         public void run() {
                         Toast.makeText(getApplicationContext(),"video uploading is done",
Toast.LENGTH_LONG).show();
                                                         }
                                                 })
                    } else {
                         runOnUiThread(new Runnable() {
                                                         public void run() {
```

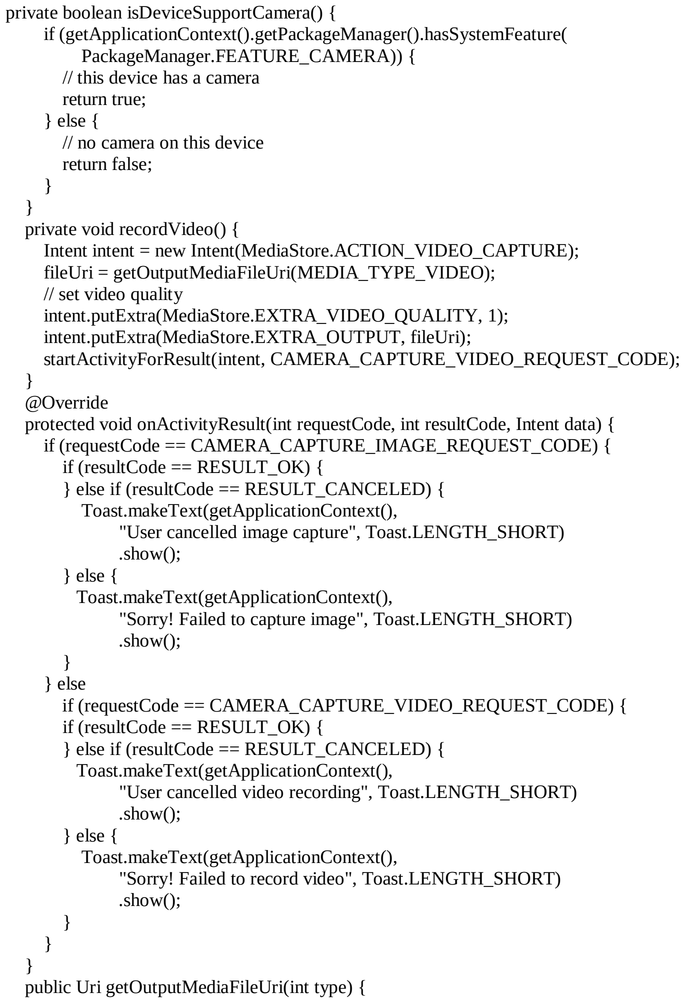
```
Toast.makeText(getApplicationContext(),      "Error to upload video",
Toast.LENGTH_LONG).show();
                                                                }
                                                        });
                          }
                        } catch (JSONException e) {
                          e.printStackTrace();
                        }
                        return null;
                    }
protected void onPostExecute(String file_url) {
                        // dismiss the dialog once done
                        pDialog.dismiss();
                    }
            }
            @SuppressWarnings("deprecation")
            public String getPath(Uri uri) {
                        String[] projection = { MediaStore.Images.Media.DATA };
                        Cursor cursor = managedQuery(uri, projection, null, null, null);
                        int column_index = cursor
                                                .getColumnIndexOrThrow(MediaStore.Images.Media.DATA);
                        cursor.moveToFirst();
                        return cursor.getString(column_index);
            }
            public int uploadFile(final String sourceFileUri) {
                        //String fileName = sourceFileUri;
                        fileName=sourceFileUri;//this is added new
                        HttpURLConnection conn = null;
                        DataOutputStream dos = null;
                        String lineEnd = "\r\n";
                        String twoHyphens = "--";
                        String boundary = "*****";
                        int bytesRead, bytesAvailable, bufferSize;
                        byte[] buffer;
                        int maxBufferSize = 1 * 1024;
                        File sourceFile = new File(sourceFileUri);
                        if (!sourceFile.isFile()) {
                                    dialog.dismiss();
                                    Log.e("uploadFile", "Source File not exist :" + filepath);
                                    runOnUiThread(new Runnable() {
                                                public void run() {
                                                            messageText.setText("Source File not exist :" + filepath);
                                                }
                                    });
                                    return 0;
                        } else {
```



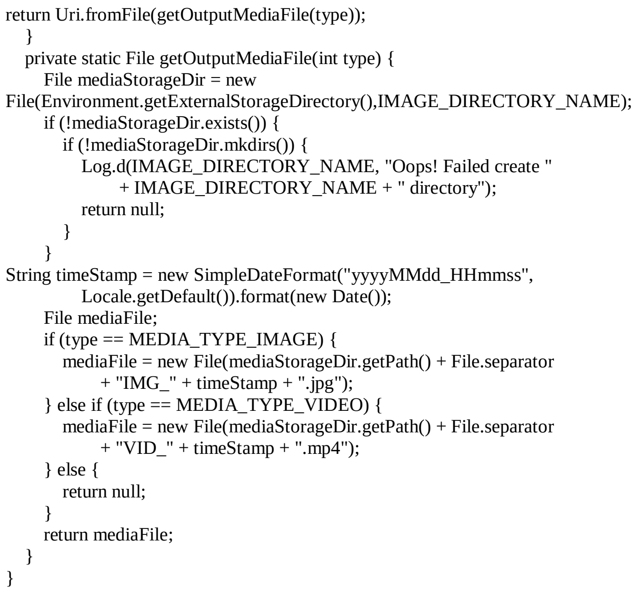
```
try {
                                        FileInputStream fileInputStream = new FileInputStream(
                                                            sourceFile);
                                        String standard=txtstandard.getText().toString();
                                        String subject=txtsubject.getText().toString();
                                         SharedPreferences settings=
getSharedPreferences(VIDEO_PREFERENCE, 0);
              SharedPreferences.Editor editor= settings.edit();
              editor.putString("standard", standard);
              editor.putString("subject", subject);
              editor.commit();
          upLoadServerUri=serverLocation+location+pathSeperator+standard+pathSeperator+bran
ch+pathSeperator+subject+pathSeperator+script;
                                        URL url = new URL(upLoadServerUri);
                                        conn = (HttpURLConnection) url.openConnection();
                                        conn.setDoInput(true); // Allow Inputs
                                        conn.setDoOutput(true); // Allow Outputs
                                        conn.setUseCaches(false); // Don't use a Cached Copy
                                        conn.setChunkedStreamingMode(1024);//i have added extra
                                        conn.setRequestMethod("POST");
                                        conn.setRequestProperty("Connection", "Keep-Alive");
                                        conn.setRequestProperty("ENCTYPE", "multipart/form-data");
                                        conn.setRequestProperty("Content-Type",
                                                            "multipart/form-data;boundary=" + boundary);
                                        conn.setRequestProperty("uploaded_file", fileName);
                                        dos = new DataOutputStream(conn.getOutputStream());
                                        dos.writeBytes(twoHyphens + boundary + lineEnd);
                                        dos.writeBytes("Content-Disposition: form-data;
name=\"uploaded_file\";filename=\""+ fileName + "\"" + lineEnd);
                                        dos.writeBytes(lineEnd);
                                        bytesAvailable = fileInputStream.available();
                                        bufferSize = Math.min(bytesAvailable, maxBufferSize);
                                        buffer = new byte[bufferSize];
                                        bytesRead = fileInputStream.read(buffer, 0, bufferSize);
                                        while (bytesRead > 0) {
                                                  dos.write(buffer, 0, bufferSize);
                                                  bytesAvailable = fileInputStream.available();
                                                  bufferSize = Math.min(bytesAvailable, maxBufferSize);
                                                  bytesRead = fileInputStream.read(buffer, 0, bufferSize);
                                        }
                                        dos.writeBytes(lineEnd);
                                        dos.writeBytes(twoHyphens + boundary + twoHyphens +
lineEnd);
                                        serverResponseCode = conn.getResponseCode();
                                        String serverResponseMessage = conn.getResponseMessage();
Log.i("uploadFile", "HTTP Response is : "+ serverResponseMessage + ": " +
serverResponseCode);
                if (serverResponseCode == 200) {
                    runOnUiThread(new Runnable() {
                        public void run() {
                            new uploadFile().execute();
    Toast.makeText(UploadVideo.this,"File Upload Complete.",
Toast.LENGTH_SHORT).show();
                            finish();
                        }
                    });
                }
                fileInputStream.close();
                dos.flush();
                dos.close();
            } catch (MalformedURLException ex) {
                dialog.dismiss();
                ex.printStackTrace();
                runOnUiThread(new Runnable() {
                    public void run() {
    messageText.setText("MalformedURLException Exception : check script url.");
    Toast.makeText(UploadVideo.this,"MalformedURLException",
Toast.LENGTH_SHORT).show();
                    }
                });
                Log.e("Upload file to server", "error: " + ex.getMessage(), ex);
            } catch (Exception e) {
                dialog.dismiss();
                e.printStackTrace();
                runOnUiThread(new Runnable() {
                    public void run() {
                        messageText.setText("Got Exception : see logcat ");
                        Toast.makeText(UploadVideo.this,
                                "Got Exception : see logcat ",
                                Toast.LENGTH_SHORT).show();
                    }
                });
                Log.e("Upload file to server Exception",
                        "Exception : " + e.getMessage(), e);
            }
            dialog.dismiss();
            return serverResponseCode;
        }
    }
}
```

Record Video:

```
public class RecordVideo extends Activity {
            // Activity request codes
    private static final int CAMERA_CAPTURE_IMAGE_REQUEST_CODE = 100;
    private static final int CAMERA_CAPTURE_VIDEO_REQUEST_CODE = 200;
    public static final int MEDIA_TYPE_IMAGE = 1;
    public static final int MEDIA_TYPE_VIDEO = 2;
    // directory name to store captured images and videos
    private static final String IMAGE_DIRECTORY_NAME = "";
    private Uri fileUri; // file url to store image/video
    //private ImageView imgPreview;
    private VideoView videoPreview;
    private Button btnRecordVideo,btnUploadRecordVideo;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.record_video);
        videoPreview = (VideoView) findViewById(R.id.videoPreview);
        btnRecordVideo = (Button) findViewById(R.id.btnRecordVideo);
        btnUploadRecordVideo =  (Button) findViewById(R.id.btnUploadRecordVideo);
        btnUploadRecordVideo.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // capture picture
                        Intent i = new Intent(getApplicationContext(),UploadVideo.class);
                                                startActivity(i);
            }
        });
btnRecordVideo.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // record video
                recordVideo();
            }
        });
        // Checking camera availability
        if (!isDeviceSupportCamera()) {
            Toast.makeText(getApplicationContext(),
                    "Sorry! Your device doesn't support camera",
                    Toast.LENGTH_LONG).show();
            // will close the app if the device does't have camera
            finish();
        }
    }
```



```
private boolean isDeviceSupportCamera() {
        if (getApplicationContext().getPackageManager().hasSystemFeature(
                PackageManager.FEATURE_CAMERA)) {
            // this device has a camera
            return true;
        } else {
            // no camera on this device
            return false;
        }
    }
    private void recordVideo() {
        Intent intent = new Intent(MediaStore.ACTION_VIDEO_CAPTURE);
        fileUri = getOutputMediaFileUri(MEDIA_TYPE_VIDEO);
        // set video quality
        intent.putExtra(MediaStore.EXTRA_VIDEO_QUALITY, 1);
        intent.putExtra(MediaStore.EXTRA_OUTPUT, fileUri);
        startActivityForResult(intent, CAMERA_CAPTURE_VIDEO_REQUEST_CODE);
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        if (requestCode == CAMERA_CAPTURE_IMAGE_REQUEST_CODE) {
            if (resultCode == RESULT_OK) {
            } else if (resultCode == RESULT_CANCELED) {
                Toast.makeText(getApplicationContext(),
                        "User cancelled image capture", Toast.LENGTH_SHORT)
                        .show();
            } else {
               Toast.makeText(getApplicationContext(),
                        "Sorry! Failed to capture image", Toast.LENGTH_SHORT)
                        .show();
            }
        } else
            if (requestCode == CAMERA_CAPTURE_VIDEO_REQUEST_CODE) {
            if (resultCode == RESULT_OK) {
            } else if (resultCode == RESULT_CANCELED) {
               Toast.makeText(getApplicationContext(),
                        "User cancelled video recording", Toast.LENGTH_SHORT)
                        .show();
            } else {
                Toast.makeText(getApplicationContext(),
                        "Sorry! Failed to record video", Toast.LENGTH_SHORT)
                        .show();
            }
        }
    }
    public Uri getOutputMediaFileUri(int type) {
```



```
return Uri.fromFile(getOutputMediaFile(type));
    }
    private static File getOutputMediaFile(int type) {
        File mediaStorageDir = new
File(Environment.getExternalStorageDirectory(),IMAGE_DIRECTORY_NAME);
        if (!mediaStorageDir.exists()) {
            if (!mediaStorageDir.mkdirs()) {
                Log.d(IMAGE_DIRECTORY_NAME, "Oops! Failed create "
                        + IMAGE_DIRECTORY_NAME + " directory");
                return null;
            }
        }
String timeStamp = new SimpleDateFormat("yyyyMMdd_HHmmss",
                Locale.getDefault()).format(new Date());
        File mediaFile;
        if (type == MEDIA_TYPE_IMAGE) {
            mediaFile = new File(mediaStorageDir.getPath() + File.separator
                    + "IMG_" + timeStamp + ".jpg");
        } else if (type == MEDIA_TYPE_VIDEO) {
            mediaFile = new File(mediaStorageDir.getPath() + File.separator
                    + "VID_" + timeStamp + ".mp4");
        } else {
            return null;
        }
        return mediaFile;
    }
}
```

Chapter 7

TESTING

7.1 What is Software Testing

Software testing is the process of analyzing or operating software for the purpose of finding bugs. Testing can be described as a process used for reveling defects in software, and for establishing that the software has attained a specified degree of quality with respect to selected attribute.The fundamental objective of testing is to find defects ,as early as possible and get them fixed.

Software Testing Process

- Test Planning high level plans which list test objectives, test approach, measurement criteria along with test schedule and resources.
- Test Design create test cases, identify test cases for automation(if applicable),prioritize test cases and finalize test iterations.
- Test Implementation Create test scripts using automated testing tools.
- Test Execution Execute the test cases on the test environment and test reports.
- Test analysis Use test and project metrics to calculate key indicators. The data usually will be obtained from your defect tracking system.
- Postmortem reviews Discuss lessons learnt and identify strategies which will prevent such problems in future.

7.2 Test methods

7.2.1 Black box testing

It is also called as functional testing, it is the process of giving the input to the system and checking the output of the system.Without bothering about the system that how the system generates the output. It is also called as Behavior testing.

- Approach to testing where the program is considered as a Black Box.
- Testing based solely on analysis of requirements user specification, user documentation etc.
- The test cases are based on the specifications.
- Black box testing techniques apply to all levels of testing.
- Test planning and design can begin early in the software process.
- Tests are done from a users point of view.

7.2.2 White Box Testing

White box testing or structural testing considers facets like programming style, control method, source language, database design. A test for remote monitoring routine can be an example of structural test. This type of testing helps to uncover defects at structural level. The tests go below the top or functional layer to uncover the defects.

- Testing that takes into account internal structure and flow of a system or component.
- The testing is based on code structure or the algorithm.
- White box testing assumes that the procedural design and code is known to the tester.
- Obliviously test design can be done only after coding is complete.
- White box tests are inherently finite.

7.3 Test cases and test data

- Test data are inputs that have been devised to test the system.
- Test cases are inputs and output specification plus a statement of the function under test.
- Test data can be generated automatically or real.

7.3.1 Test Cases

TC ID	OBJECTIVE	PREREQUISITES	STEPS TO BE FOLLOWED	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Login into the application	User Must be login	1.User Should be enter the correct user name. 2.User shold be fill the password in the password field. 3.click on Login button	The Application will give an authorized access.	Granted the permissions and redirect to next page	Pass
2	Register student into the application	To perform it's operation user must be click on New User button	1.Enter First Name in FirstName TextField 2.Enter Middle Name in MiddleName TextField 3.Enter Last Name in LastName TextField 4.Enter Standard in Standard TextField 5.Enter Brach in Branch TextField 6.Enter Roll No. in RollNo. TextField 7.Enter Password in Password TextField	It will create student account successfully	Give Confirmation message to student about registration	pass

Continued on next page

Table 7.1 – continued from previous page

TC ID	OBJECTIVE	PREREQUISITES	STEPS TO BE FOLLOWED	EXPECTED RESULT	ACTUAL RESULT	REMARK
			8.Re-enter Password in ConfirmPassword TextField 9.Enter Security Question in SecurityQuestion TextField 10.Enter Answer of Security Question in Answer TextField 11.Click on Submit Button			
3	Register staff in application	To perform it's operation staff must be click on New User button	1. Enter First Name in FirstName TextField 2.Enter Middle Name in MiddleName TextField 3.Enter Last Name in LastName TextField 4.Enter Username in UserName TextField 5.Enter Password in Password TextField 6.Re-enter password in ConfirmPassword TextField	It will create staff account successfully	Give Confirmation message to staff about registration	pass

Continued on next page

Table 7.1 – continued from previous page

TC ID	OBJECTIVE	PREREQUISITES	STEPS TO BE FOLLOWED	EXPECTED RESULT	ACTUAL RESULT	REMARK
			7.Enter Security Question in SecurityQuestion TextField 8.Enter Answer of Security Question in Answer TextField 9.Click on Submit button			
4	Register Admin in application	To perform it's operation admin must be click on New User button	1.Enter username in UserName TextField 2.Enter password in Password TextField 3.Re-enter pasword in ConfirmPassword TextField 4.Enter Security Question in SecurityQuestion TextField 5.Enter Answer of security question in Answer TextField 6.Enter Security Key in SecurityKey TextField 6.Click on Submit Button	It will create admin account successfully	Give Confirmation message to admin about registration	pass

Continued on next page

Table 7.1 – continued from previous page

TC ID	OBJECTIVE	PREREQUISITES	STEPS TO BE FOLLOWED	EXPECTED RESULT	ACTUAL RESULT	REMARK
5	View Video	To perform it's operation user must be click on Show Subjectwise Videobutton	1.User should click on Show Subjectwise video Button 2.User should Click on particular subject in the list 3.User should select particular video for playing video	It will show video list & play particular video	It will show video list & play particular video	pass
6	Upload Video	To perform it's operation staff must be click Upload-Video button	1. Enter standard in Standard TextField 2.Enter sunject in Subject TextField 3.click on Select Video button 4.Select Video from sdcard 5.Click on Upload button	It will upload video on server	After uploading video on server it will confirmation message to staff	pass

Continued on next page

Table 7.1 – continued from previous page

TC ID	OBJECTIVE	PREREQUISITES	STEPS TO BE FOLLOWED	EXPECTED RESULT	ACTUAL RESULT	REMARK
7	Record Video	To perform it's operation staff must be click on Record Video button	1.Click on Record Video button 2.Click on start button 3. Click on stop button 4.Click on done button 5.Click on Upload button	Recording complete & upload that video on server	Recording complete & upload that video on server	pass
8	Bookmark video	To perform it's operation user must be click on pause button	1. Click on show subjectwise videolist button 2.Select particular subject from subject list 3.Select video from video list for playing that video 4.click on Bookmsrk Video button	It will store details of video into the database	It will store details of video into the database	pass

Continued on next page

Table 7.1 – continued from previous page

TC ID	OBJECTIVE	PREREQUISITES	STEPS TO BE FOLLOWED	EXPECTED RESULT	ACTUAL RESULT	REMARK
9	Delete Video	To perform it's operation admin must be click on Delete Video button	1. Click on delete video button 2.Enter standard in Standard TextField 3.Enter subject in Subject TextField 4.Enter video name in VideoName TextField 5.Click on Delete Video button	It will delete video from database	It will delete video from database successfully	pass
10	Delete Account	To perform it's operation admin must be click on Delete Account button	1.Click on Delete account button 2.Enter username in UserName TextField 3.Clicks on Delete Account button	It will delete account from database	It will delete account from database successfully	pass

Table 7.1: Test Cases

Chapter 8

CONCLUSION

8.1 Conclusion

Our project would definitely provide new revolution in our education system scenario. The portability it provides is beneficial from students point of view. The feature it has makes it a better education tool for students. Todays education is the most expensive and proposed portal in the current scenario. It makes a person literate and gives him/her knowledge to learn the basic needs of life. So taking all these things in mind an android application Virtual Classroom is developed to dynamically load the video lectures just by making a secure Wi-Fi connection with the available local server. In our project we have stated the utility of the application Virtual Classroom that would certainly define the future perspective of the education in society. The methods that are being implicated are useful for learning and saving time and effort. The application provides a real time streaming of video and is portable so the range within which one can operate is reasonable. Finally we can conclude that the use of this application will embark a true mark in the education society and then learning would be affordable to distances.

Appendix A

Appendix Title Here

Write your Appendix content here.

Bibliography

[] A. S. Arnold, J. S. Wilson, and M. G. Boshier. A simple extended-cavity diode laser. *Review of Scientific Instruments*, 69(3):1236–1239, March 1998. URL http://link.aip.org/link/?RSI/69/1236/1.

[] Carl E. Wieman and Leo Hollberg. Using diode lasers for atomic physics. *Review of Scientific Instruments*, 62(1):1–20, January 1991. URL http://link.aip.org/link/?RSI/62/1/1.

[] C. J. Hawthorn, K. P. Weber, and R. E. Scholten. Littrow configuration tunable external cavity diode laser with fixed direction output beam. *Review of Scientific Instruments*, 72(12):4477–4479, December 2001. URL http://link.aip.org/link/?RSI/72/4477/1.