

Virtual Classroom Platform (Java Web Application)

Project Overview

This project is an Online Virtual Classroom platform designed to facilitate interaction between teachers and students remotely. It supports essential e-learning features such as user authentication, course management, and content sharing.

The original project implements a traditional 3-tier Java Web Application architecture using **JSP** (JavaServer Pages) for the presentation layer, **Java Servlets** for the controller and handling requests, and **MySQL** for persistent data storage.

Key Features

- **User Authentication:** Separate login for **Students** and **Teachers**.
- **Role-Based Access:** Teachers can create and manage courses; Students can enroll and view content.
- **Course Creation:** Teachers can add new subjects/courses with descriptions.
- **Content Sharing:** Upload and share lecture materials (video links, PPTs, documents).
- **Interactive Platform:** Facilitates various forms of communication (simulated through static links and message boards).

Technology Stack

Layer	Technology	Version / Tool	Purpose
Backend	Java	JDK 8+	Core business logic and server-side processing.
Data Access	JDBC	MySQL Connector	Connects Java logic to the database.
Presentation	JSP (JavaServer Pages) & HTML/CSS/JS	Servlet Container (e.g., Apache Tomcat)	Handles dynamic content generation and user interface.
Database	MySQL		Stores all user, course, and

			content data.
Build/Deployment	Maven / WAR file		Packaging and dependency management.

Project Structure (Simplified)

Virtual-Classroom/

— src/	
— main/	
— java/	# Java Servlets, Models, and Service logic (e.g., User.java, ClassroomService.java)
— webapp/	# JSP and Web Content Root
— WEB-INF/	# Configuration files (web.xml)
— css/	# All CSS files (including SCSS if processed)
— js/	# All JavaScript files
— *.jsp	# Core JSP files for Login, Student/Teacher Dashboard, etc.
— pom.xml	# Maven build configuration
— README.md	# This file

Setup and Running Locally

To run this project, you need to set up a MySQL database and a Java Servlet Container (like Tomcat).

Prerequisites

1. Java Development Kit (JDK 8 or higher)
2. Apache Maven (to build the project)
3. MySQL Server
4. Apache Tomcat (Servlet Container) or equivalent IDE integration.

Steps

1. Clone the Repository:

```
git clone
```

```
[https://github.com/nirmalnishant645/Virtual-Classroom.git](https://github.com/nirmalnishant645/Virtual-Classroom.git)
```

```
cd Virtual-Classroom
```

2. Database Setup:

- Create a new MySQL database (e.g., virtual_classroom).

- Execute the SQL script (often found in the repository root or a dedicated sql/ folder) to create the necessary tables (user, course, etc.).

3. Update Configuration:

- Locate the database connection settings (often in a Java utility class or a properties file).
- Update the JDBC URL, username, and password to match your local MySQL configuration.

4. Build the Project:

- Use Maven to compile the Java code and package the application into a WAR file.

<!-- end list -->mvn clean install

5. Deployment:

- Deploy the generated virtual-classroom.war file to your Tomcat server's webapps directory.

6. Access the Application:

- Start Tomcat and navigate to the application URL in your browser (e.g., <http://localhost:8080/virtual-classroom>).