Fall 2022

Online Learning Management System

Sai Karan Reddy Kankunta

Online Learning Management System

By

Sai Karan Reddy Kankunta
B.Tech, JNTU, 2015

GRADUATE CAPSTONE SEMINAR PROJECT

Submitted in partial fulfillment of the requirements

For the Degree of Master of Science,

With a Major in Computer Science

Governors State University
University Park, IL 60484

2022

ABSTRACT

Online Learning websites have completely altered the learning, course search, or learning process. Instructors and Students can now search and enroll in learning courses from the convenience of their own homes. The student/instructor can use an online learning website to find courses that align with their interests and career path. An Online Learning Management System may significantly improve a student's course application process and an instructor's enrolling process for their courses by utilizing the internet (*Jeremy Roschelle*, 1992). Seeking and enrolling in a course has grown simpler, more accessible, and faster in recent years.

One of the primary features of enrolling in a course online is the availability of a wide range of categories of subjects for the courses. Course seekers may simply enroll in a broad range of courses by logging in and browsing the courses in the dashboard. Apart from that students can view their courses any time by logging in to the web application and entering the student homepage.

Similarly, internet course enrollment services may assist instructors in expediting their enrollment process. Instructors can easily create, update, or disable their courses in the web application with a few clicks. As a result, Online Learning Management Systems save them time and effort.

The cost of walk-in enrollments would be too high for the instructor. Aside from transportation and meals, the instructor must also invest capital in professional clothing and cannot present multiple courses at a single stretch. The Online Learning Management System allows the instructor to host his courses in the web application without breaking the bank.

In the proposed Online Learning Management System, the students and instructors can register themselves in the application. Students and instructors are assigned different roles. The student can see a list of courses and can apply to any course that matches his/her interest or career path. Similarly, instructors can add new courses and view their profits in the dashboard when students purchase his/her course.

To implement the frontend module of the project Bootstrap, jQuery, JavaScript and HTML5 is being used. Java and Spring boot MVC is being used for the backend module of the web application. MySQL is used to store the user and course data.

The web application consists of two different roles as follows: the users with student roles will be able to view a course title, instructor name, and brief description of the course and purchase the course by providing their credit/debit card details. The instructor role user can create, edit, and disable their course in the web application after registration.

Table of Content

Contents

1 Project Description

Online Learning Management System is a web application, the platform provides an instructor with a facility to upload his/course in the portal and for students to enroll/purchase the course he/she desires after the registration. The web application will be a platform for students to view or purchase a variety of courses from a wide range of instructors. The application UI/UX is user-friendly, and the application consists of a login screen where the user can also register if not already registered.

Roles and Features:

The following are the roles supported in the application.

1. Instructor
2. Student

Instructor

Any user can register and can become an instructor. The instructor role would be able to create a course and attach a file of their course. The instructor can later update the course details and can disable the course if required.

Student

Students can sign up for the web application and after logging into the application, he/she will be provided with a homepage with the courses he/she purchased. The student can direct to the courses page and browse through a variety of courses by all instructors.

1.1 Competitive Information

Various online learning management portals are like our platform. The following are a few of the competitors of this application.

- Udemy
- Coursera
- Khan academy

1.2 Relationship to Other Applications/Projects

This is an online learning management web application, which helps students and instructors to enroll or post courses on a web-based platform. The web application uses Spring boot and web technologies like HTML, CSS3, jQuery, and JavaScript.

1.3 Assumptions and Dependencies

Only authenticated users who have registered can access this website, and if they have not already, they can do so right away. Instructors will be able to create and sell their courses. Students will be able to enroll in the courses using any credit and debit card.

1.4 Future Enhancements

Currently, the application is not horizontally scalable, eventually, when the application gains more traffic, the architecture should be able to scale horizontally. CDN can be integrated to render pages faster regardless of the location of the user. Multi Factor authentication can be integrated to increase login security.

1.5 Definitions and Acronyms

Instructor: The one who creates and uploads the course.
CDN: Content Delivery Network
Student: The One who purchases and views courses.
Course: An extended period of scheduled study that usually results in a qualification

2 *Project Technical Description*

This project is built on the Java Spring boot framework. The following is the software stack used for development and deployment

Software Stack

- **Backend**
  - Java – 11.0
  - Spring Boot framework – 2.7.4
  - IntelliJ by JetBrains 2022
  - SQL server (local DB)
- **Frontend**
  - HTML
  - Css3
  - JavaScript
  - Bootstrap
  - jQuery
  - Google Chrome

2.1 Application Architecture

The design pattern used for the application is MVC (Model, View, and Controller) in the Spring MVC architecture. (Abdul Majeed & Ibtisam Rauf, 2018, 7)

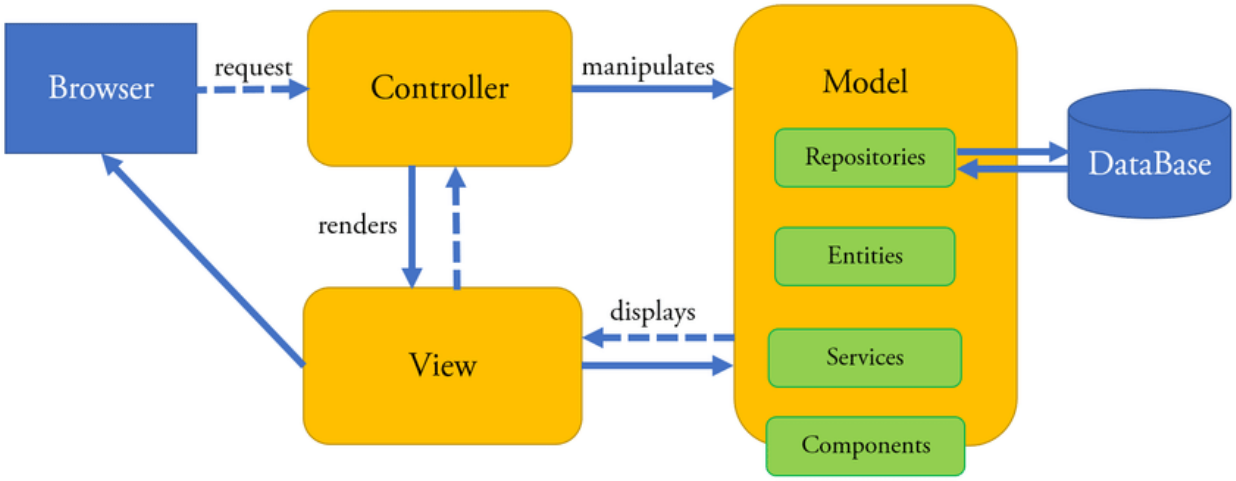

Figure 1: MVC Architecture

2.2 Application Information flows

The below illustration (figure 2) demonstrates how the flow of the web application works. The user will be able to login through the login page or register if not already registered. The user will be then redirected to their respective screen/views based on their role.

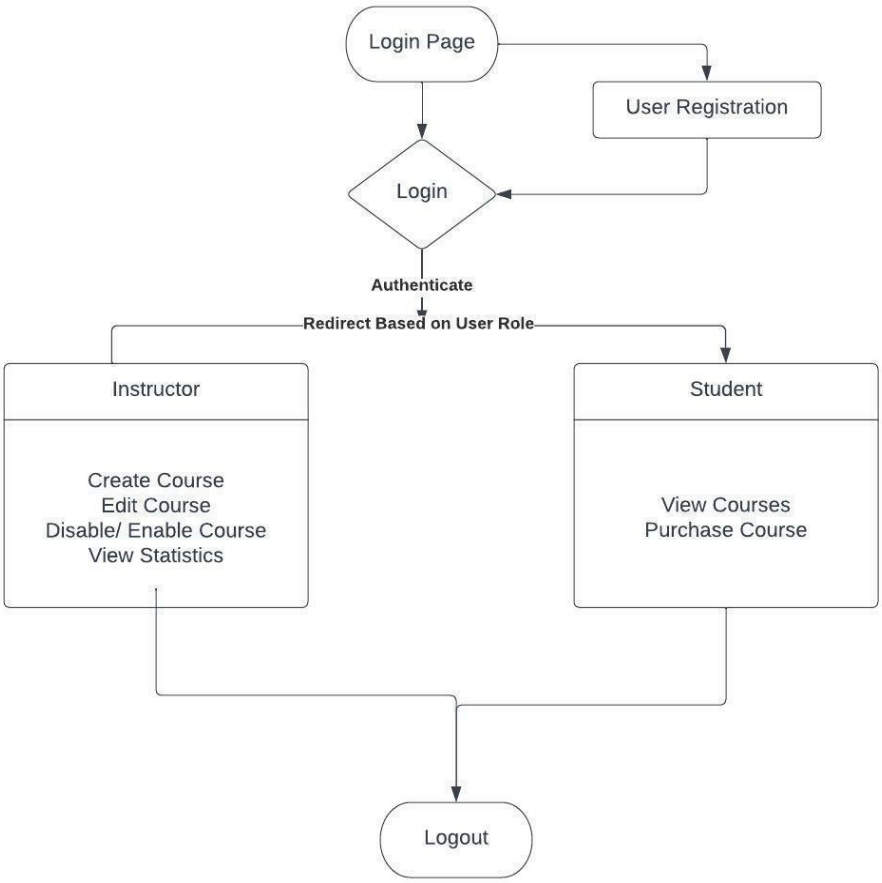

Figure 2: Application Architecture of Job Engine

Sequence Diagrams

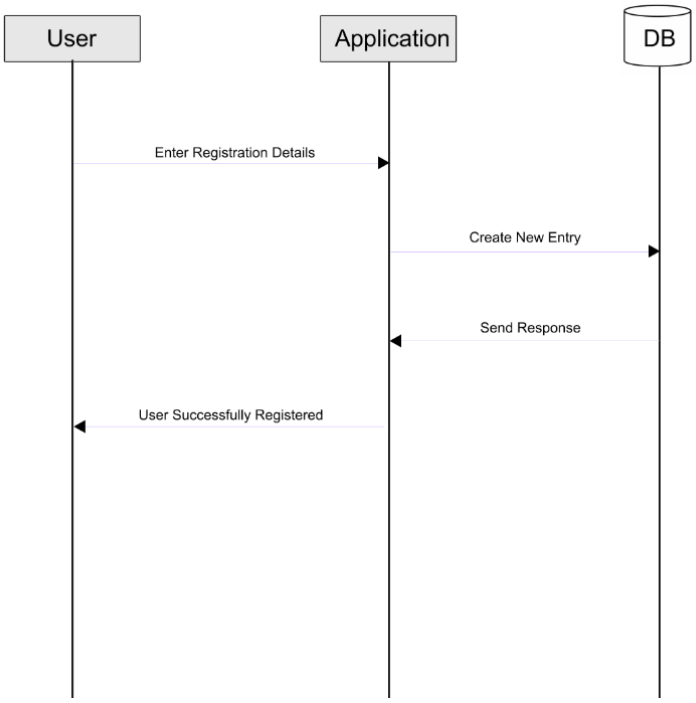

Figure 3: Sequence diagram for Registration

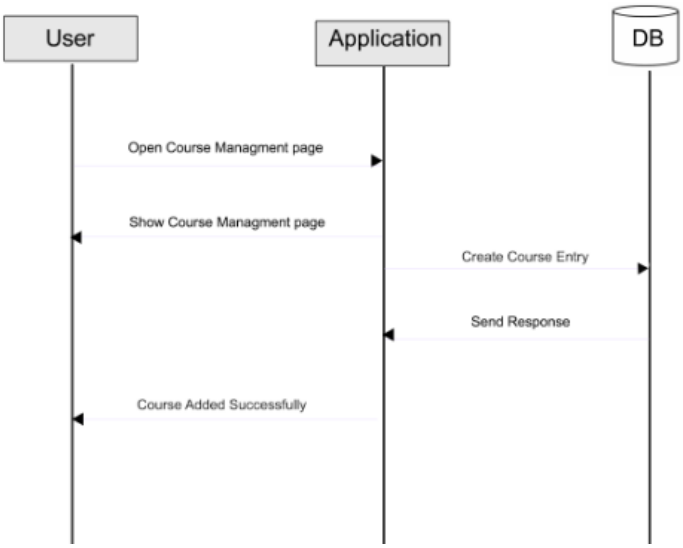

Figure 4: Sequence diagram for Course Entry

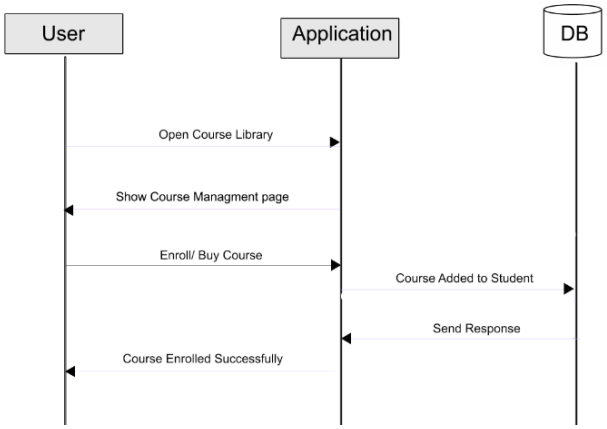

Figure 5: Sequence diagram for Course Enrollment

2.3 Interactions with other Applications

None (Currently this is a standalone application and does not interact with other Projects.)

2.4 Capabilities

- Database user must have a write access
- Data backups must be done periodically to avoid data loss
- The availability of the server should be high without any downtimes for compatible vertical scaling.

2.5 Risk Assessment and Management

The possibility of abusing and misusing the application is high, people might create fake accounts and create false courses to sell courses of no value. The following are the possible solutions.

- Add captcha verification to avoid automated scripts to create fake data/ users.
- Limit users to create courses in a specific range of time.
- Add two-factor authentication for logging, purchasing, and creating the course.

3 Project Requirements

3.1 Identification of Requirements

The use of named and enumerated requirements to identify and number requirements are briefly explained in this section. Please utilize the following SMART criteria as a guide for each requirement (Guide) :

• Specific - concentrate on a single area that needs work.

• Measurable - quantify or at the very least, suggest a progress indicator.

• Achievable - state what will be achieved.

• Realistic - Describe the outcomes that can be attained realistically with the resources at hand.

• Time-related - indicate the earliest possible time for the result(s).

User Capability:
<GSU-GS_FA2022-1 User-Capability- 000100>
The application should be capable of adding, removing, and updating users.
Implementation: Mandatory – 2 user roles can be created in this application (Instructor and Student)

<GSU-GS_FA2022-1 Course-Capability- 000200>
The project must allow new courses to be posted, updated, or disabled by the users with instructor roles.
Implementation: Mandatory – Courses should be managed by the users with the role of Instructor.

<GSU-GS_ FA2022-1 Course-Capability- 000300>
The project must allow users with the role of students to apply for adding new courses to his/her profile.
Implementation: Mandatory – For all courses created by the instructor the student should be able to purchase it to their profile.

<GSU-GS_ FA2022-1 Course-Capability- 000300>
The project must allow users with roles as instructors to view his/her statistics of purchases made for their courses.
Implementation: Mandatory – For all courses created by the instructor should be able to view the profits obtained by selling his/her course.

User Experience:
: **<GSU-GS_SP2022-1 Node-00003>**

The web application consists of all the useful elements that an enterprise application has. The application can handle huge traffic, the UI is responsive, Frontend and backend validation are enabled, and uses spring security.

3.2 Operations, Administration, Maintenance, and Provisioning (OAM&P)

Operations: Operation actions are performed to keep the network (and the services it provides) operating. It requires monitoring the network and locating problems. These vulnerabilities should ideally be identified before users are impacted. Daily, the application's problem tracking can be done utilizing the JIRA dashboard, which allows developers to keep track of jobs or issues. JIRA allows you to create a personal dashboard and add gadgets to keep track of your tasks and problems. Dashboards are designed to display gadgets that help you organize your chores, assignments, and successes in various charts.

Administration: Network resource use must be monitored as part of the administrative duties. All the accounting data needed to monitor and manage a network's hardware and software components are included. A company's Master Admin can be set up in the system and can be given control over all the recruiters working for that company. The SQL administrator can access the Users table to see every user who has signed up for the system, including the company's administrative user, and can manage all of them directly from the SQL entity table.

Maintenance: Repairs and upgrades can be made possible by performing maintenance tasks, such as installing a new switch or updating the operating system image on a router. Modifying the configuration and settings of devices falls under the remit of preventative and corrective maintenance that can increase the managed network's performance. To ensure that you always have the most up-to-date copy of your application, the database administrator may set up a cron job to automatically back up the SQL Server database and the program's source code, along with all the relevant version histories, at regular intervals.

Provisioning: Provisioning is used to build up the network so that it can supply new services, whereas operations and availability management (OAM) is used to maintain the network so that it can continue to support current services. Modifications to the network settings that are associated with the database may be adjusted in the appsettings.json file, which is in the project folder. Users can alter the launchSettings.json file located in the project's Properties folder to set their profiles for usage with this project. The serviceDependencies.json file, which is in the project's Properties folder, may be used to manage the project's dependencies.

3.3 Security and Fraud Prevention

The procedure of securely processing a large number of requests sent by a single user or entity to a web-based application or service is referred to as session management. HTTP is the protocol that allows websites and browsers to connect, and a session is the collection of HTTP requests and transactions that are started by the same user. Whenever a user validates their identity by providing a password or participating in another kind of authentication, a session is considered to have officially begun.

Because session management involves the trading of secrets with users who have already been authorized, the adoption of cryptographically sound interactions across the network is essential for ensuring the confidentiality of session management.

Spring Security offers a method that controls and restricts the maximum number of open sessions that may be held by a single user. Users are unable to establish more concurrent connections than the system will safely tolerate thanks to this feature. For instance, the number of screens on which you may simultaneously view content on Netflix is contingent on the type of membership plan you have.

3.4 Release and Transition Plan

Agile Release planning is the current best practice for the release and transition plan utilized by most businesses. In Agile, a release is often described as the smallest bundle of software features that can be efficiently packaged and provided to users.

This idea of the smallest group, which is sometimes referred to as the fundamental market features, varies from what we see in certain other development techniques, in which releases may be major events that include a substantial amount of new information all at once.

The completion of a typical Agile release might take anywhere from two to six months and could include anywhere from three to ten sprints or even more.

The process of making decisions within a team environment should always include agile release planning. The amount of functionality that can be provided to teams as well as the amount of time that each feature will take to construct may be determined with the assistance of release plans.

They are also fantastic tools for communicating with those working within the company and for establishing crystal clear expectations for what is to come.

Release plans provide the development team with a perspective by locating individual sprints within the context of the project's larger strategic purpose. At the same time, the release plans must be specific enough to support day-to-day operations and guarantee the project stays on schedule.

4 Project Design Description

The Spring Online MVC framework offers a Model-View-Controller (MVC) architecture as well as ready components that can be integrated into web applications to make them more adaptable and less dependent on external dependencies. The MVC pattern allows for a loose coupling between the different pieces of an application, which leads to the distinct portions of the program (input logic, business logic, and UI logic) being separated from one another (Abdul Majeed & Ibtisam Rauf, 2018, 7). The Model is responsible for enclosing all the application's data, and in most cases, it will be composed of POJO. The View is accountable for rendering the model data, and in most cases, it provides HTML output that can be understood by the client's browser. It is the job of the Controller to handle user requests, construct a suitable model, and then send that model along to the view so that it may be shown.

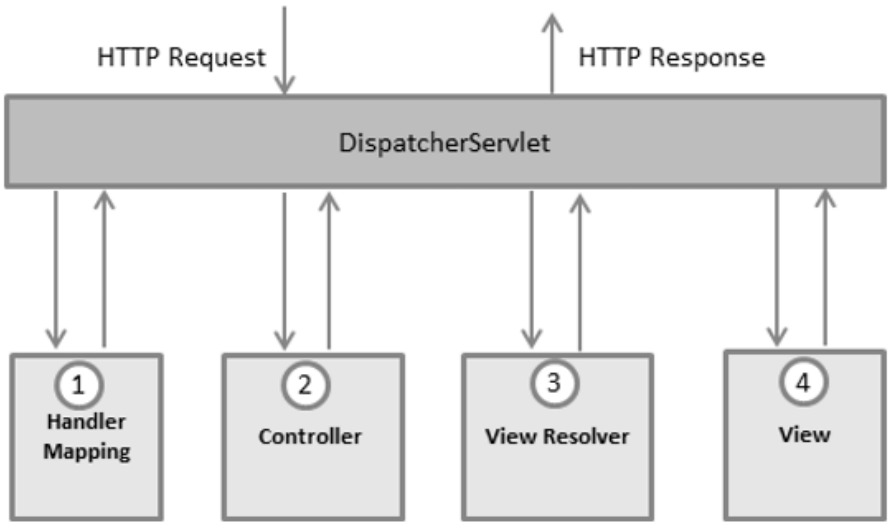

Figure 6: HTTP request flow

5 Internal/external Interface Impacts and Specification

To Design the interface, there are a few technologies that participate in making the application user-friendly; they are jQuery, Bootstrap, Java 8, Spring Boot, SQL

Spring MVC Architecture:
Spring is a modular framework. It is not sold as a package or a bundle that contains several different modules altogether. Several spring components are packaged separately in their modules (Abdul Majeed & Ibtisam Rauf, 2018, 7). This provides us with the freedom to use only what we require and disregard the remaining portions. We don't need to include the Spring Web Module in our project, for instance, if we intend to make use of the Spring JMS module. This results in a lightweight and focused application from our company.

Let's take a high-level look at the Spring Architecture (refer to the figure 7), The Core serves as the foundation of the Spring Framework, as can be seen in the diagram. Additional modules consist of things like Web, Data Access, and a variety of other modules. The test module, which is likewise built on the core, is distinct from the rest of the application since it is responsible for testing the spring-backed components of the program.

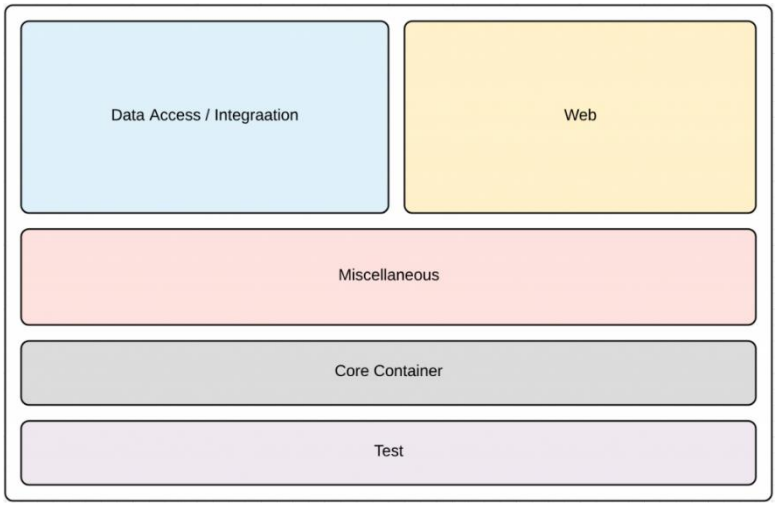

Figure 7: Spring Architecture

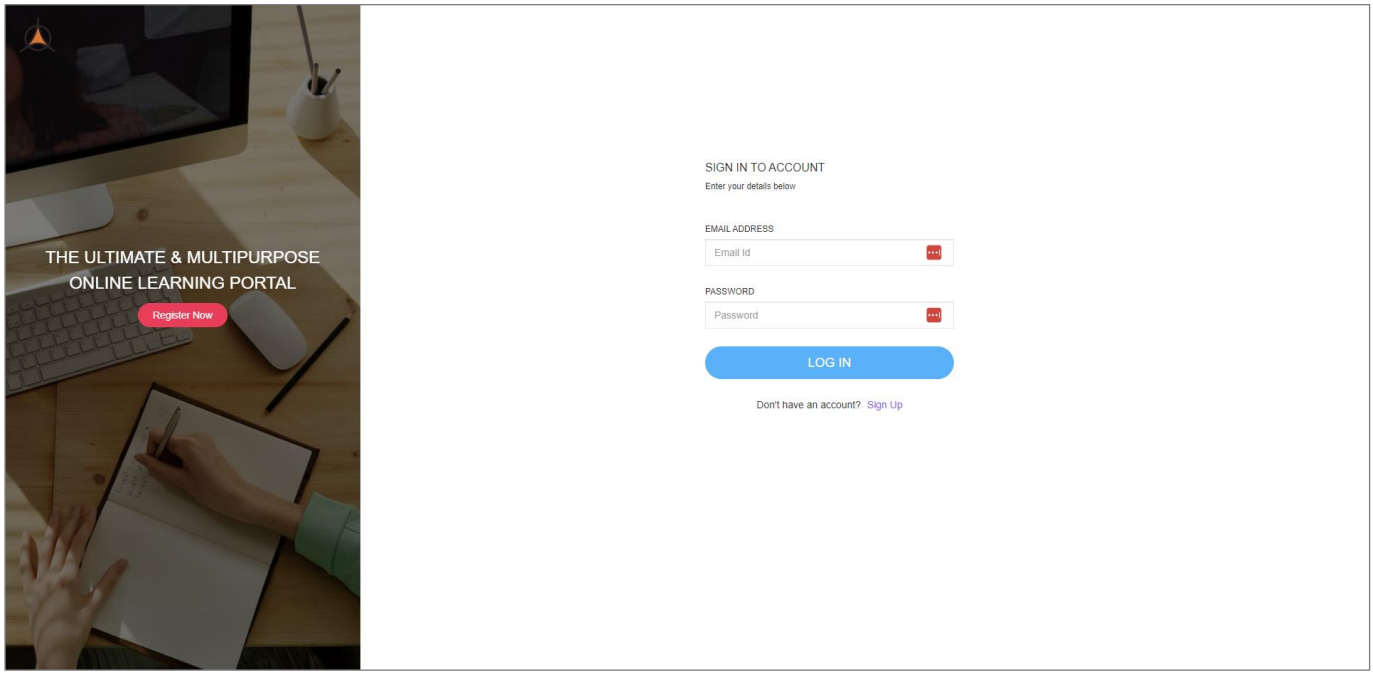

Figure 8: Login Page

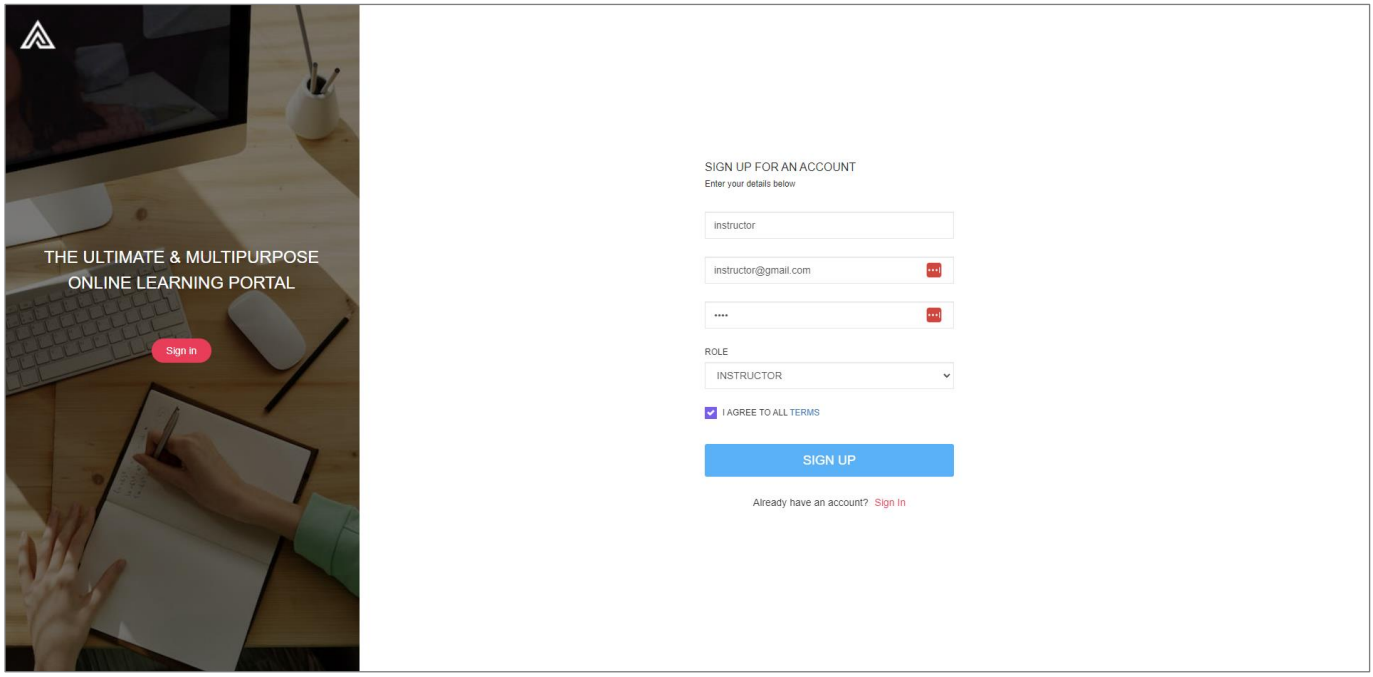

Figure 9: Registration Page

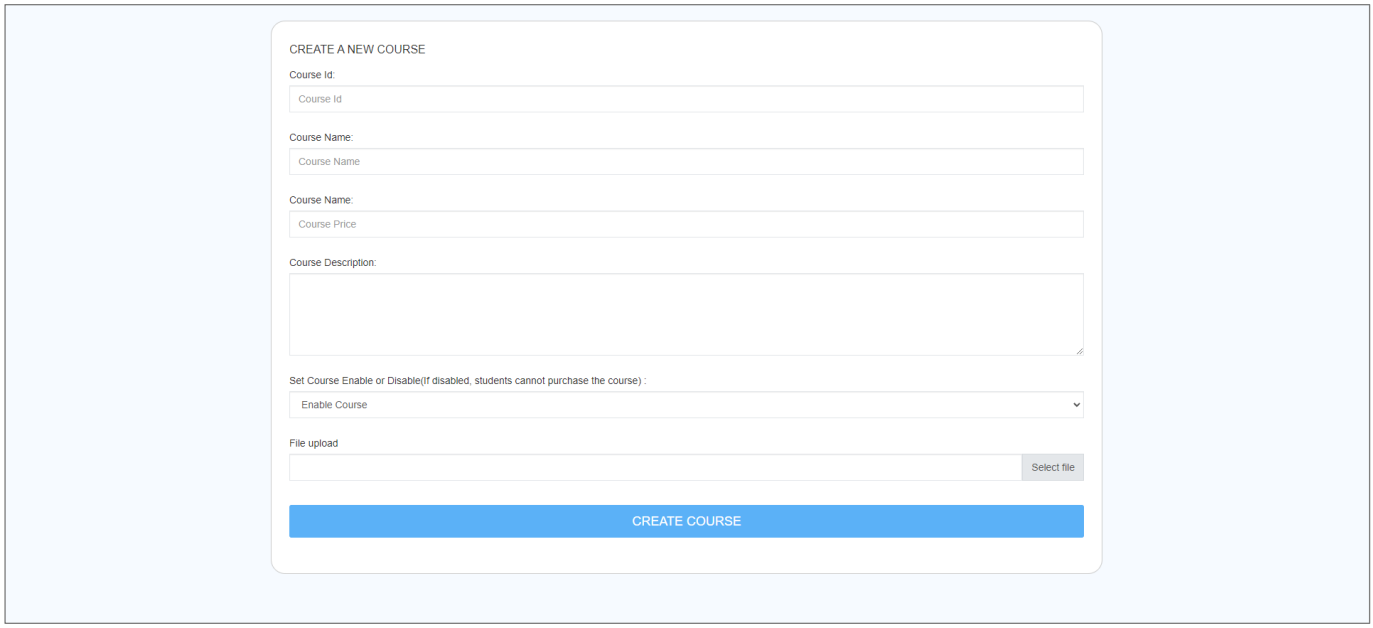

Figure 10: Instructor Course Creating a page

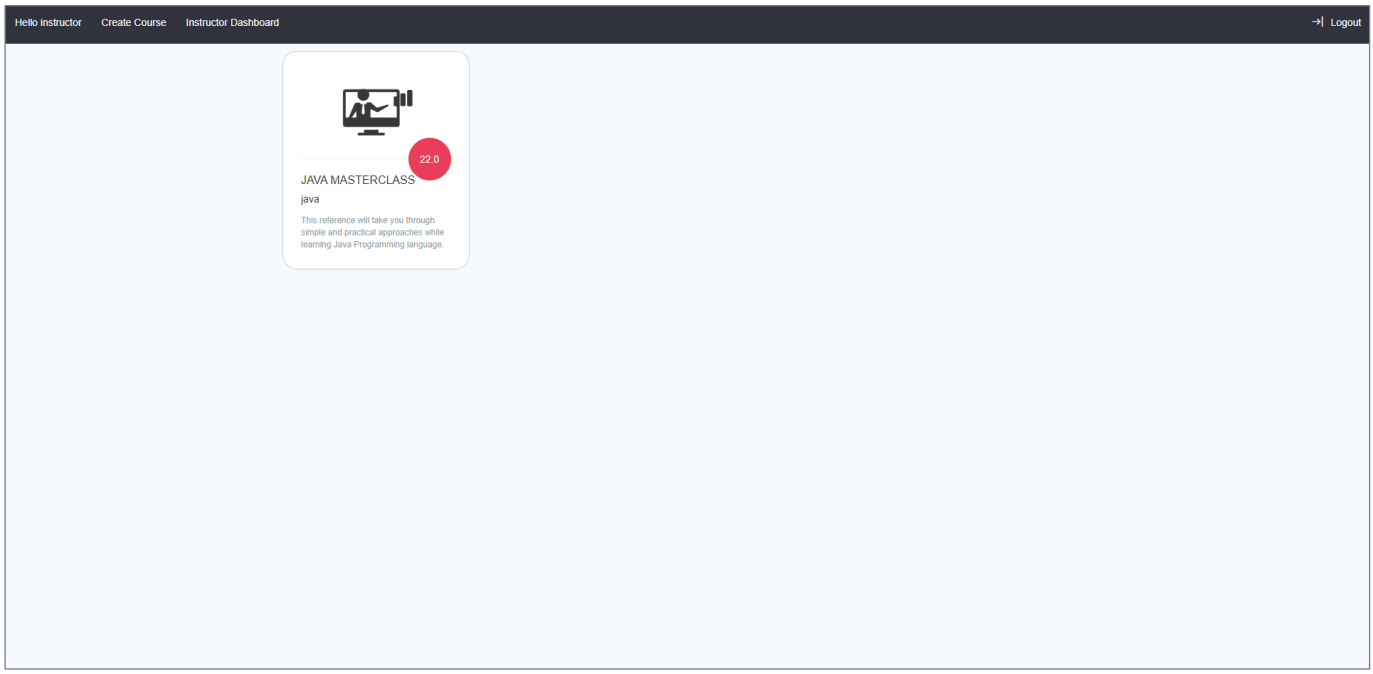

Figure 11: Instructor – homepage

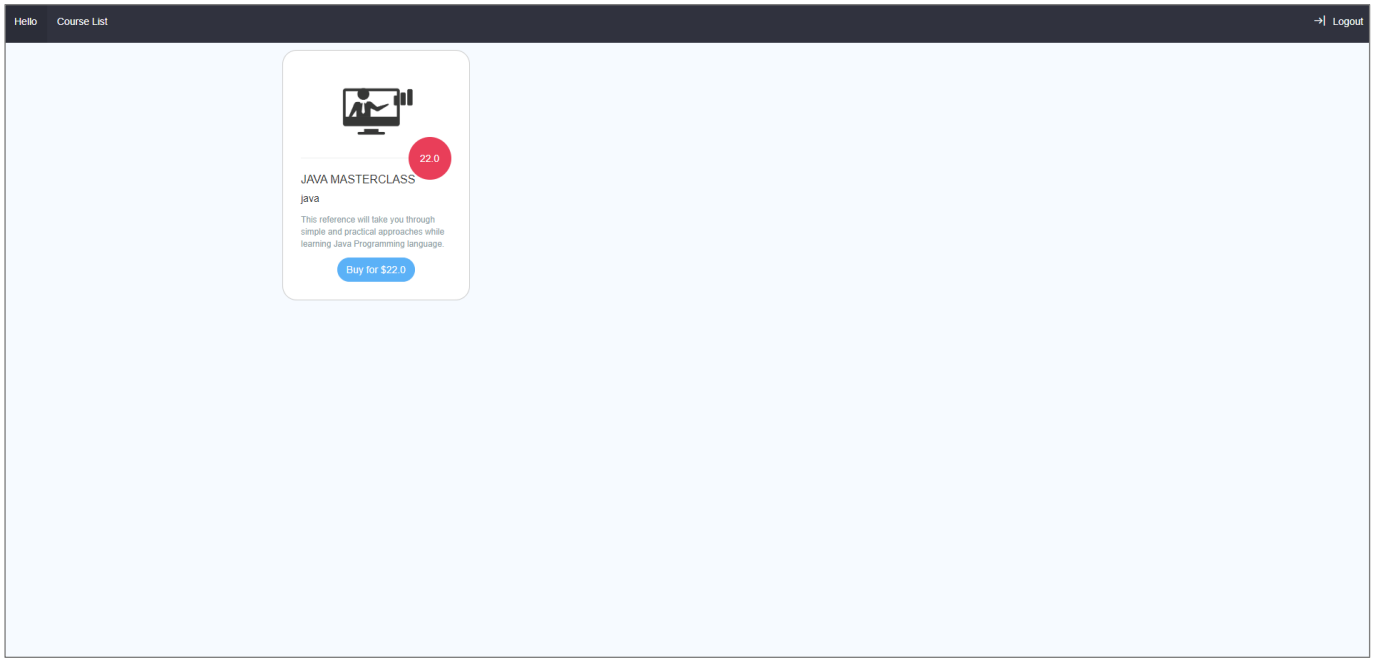

Figure 12: Student – Course List page

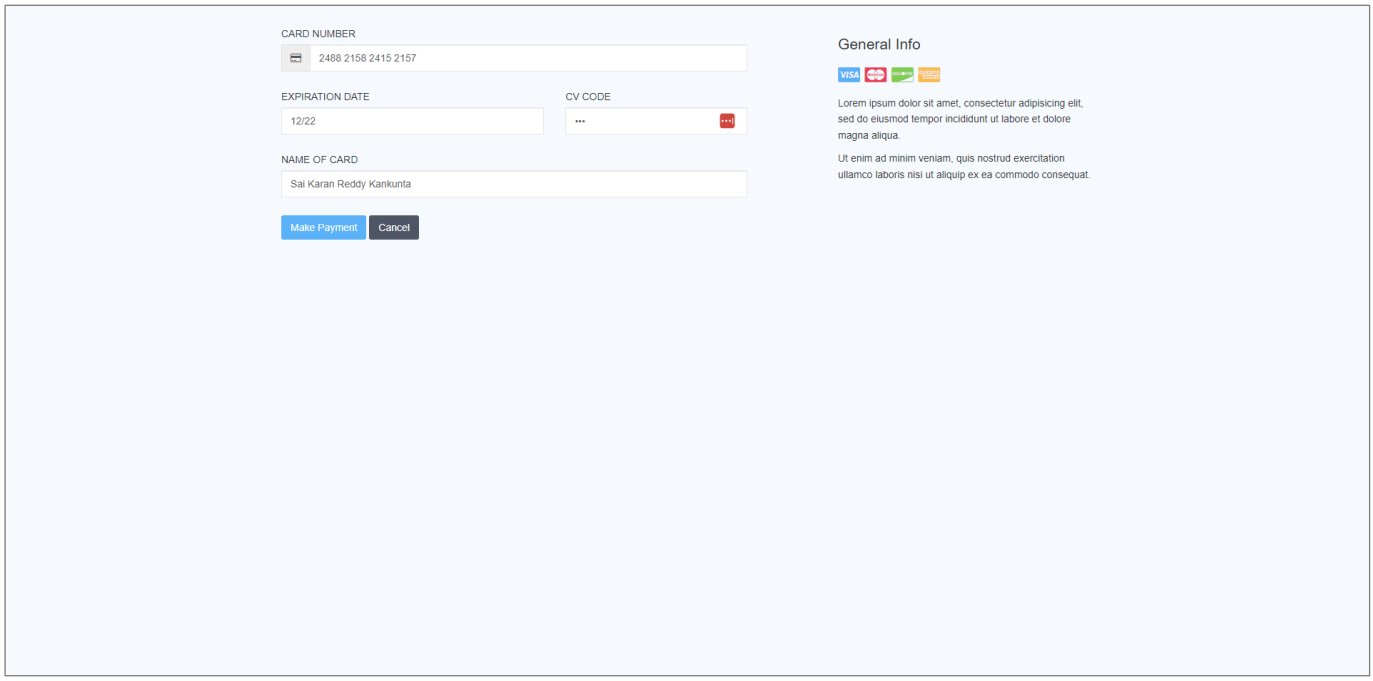

Figure 13: Student-Purchase Page

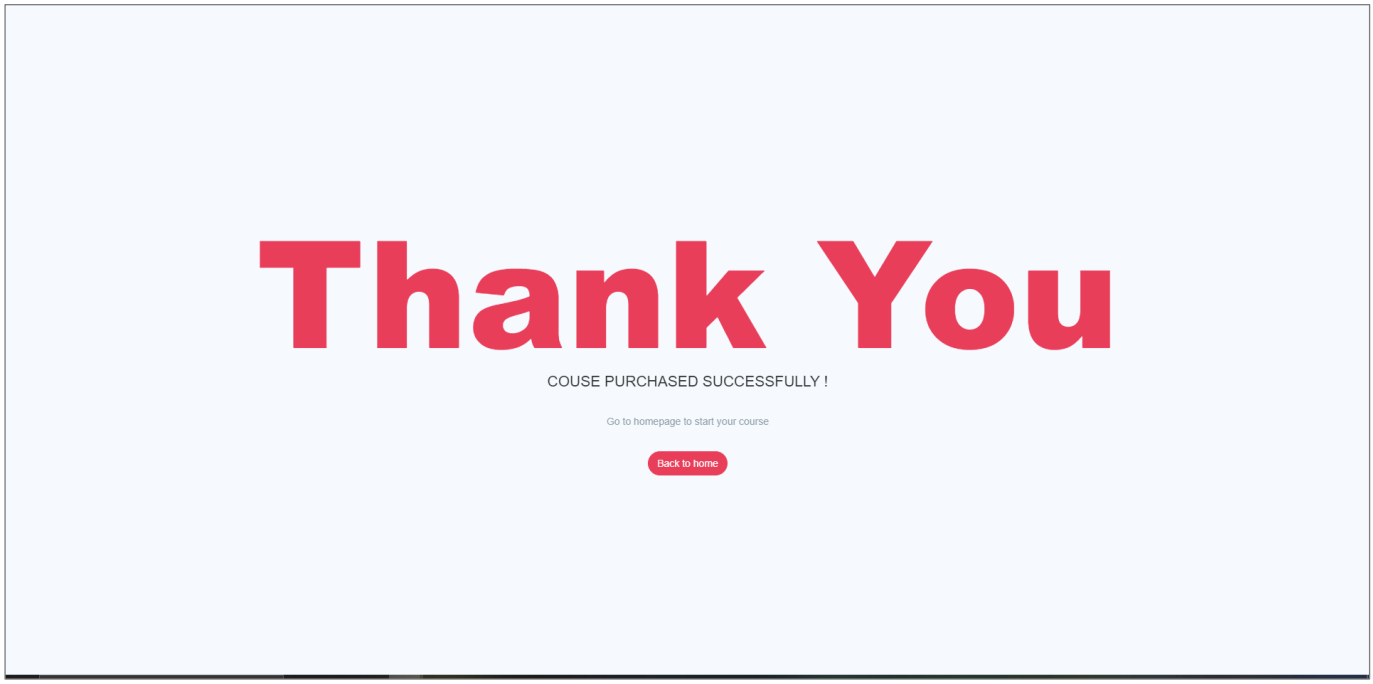

Figure 14: Purchase Successful page

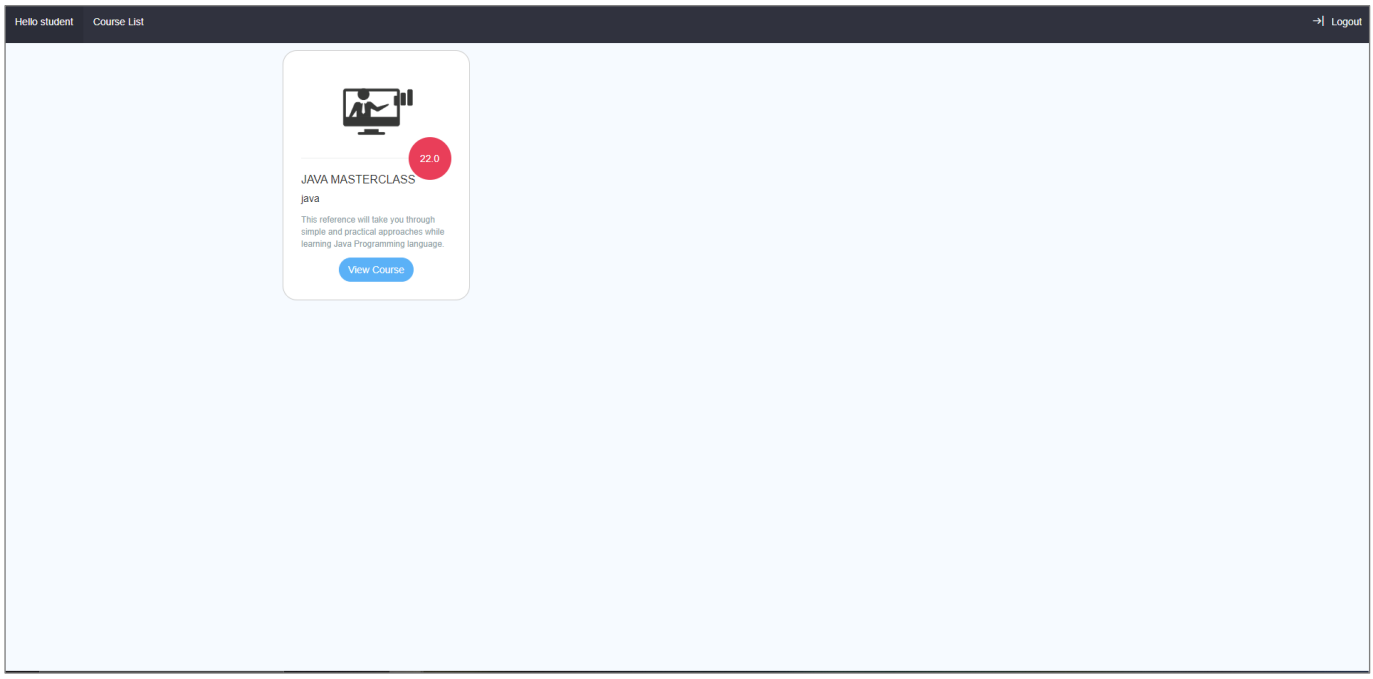

Figure 15: Student Home Page

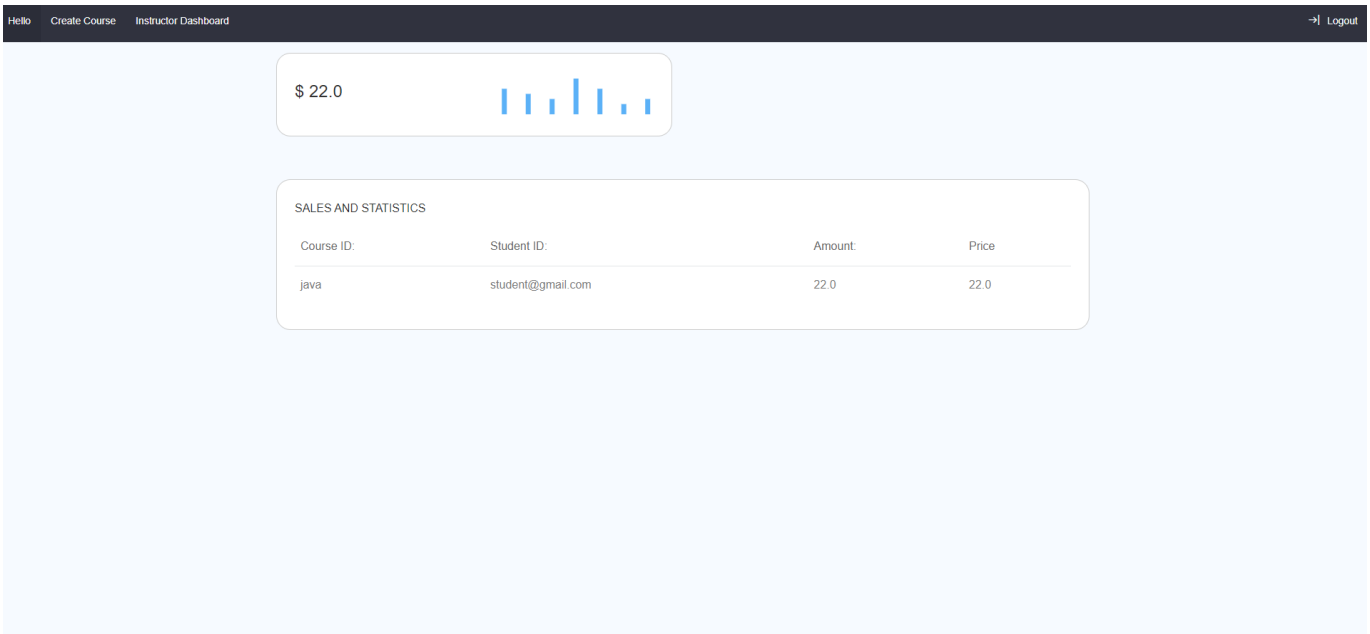

Figure 16: Instructor Dashboard Page

6 Design Units Impacts

If your software is well-designed, maintaining it won't be difficult for you at all. The architecture of software plays a significant role in determining how straightforward it is to add new capabilities to a system. It must be capable of incorporating updates without necessitating alterations to the code that is already there. Extra features added to the software to cover up design flaws are easy to see, which is an important point to keep in mind. This is a common tactic that is utilized for covering up design delays. Over time, we have witnessed developers introducing new features that customers have not asked for. Be mindful of this tactic and remember that if you find an additional component in your design, you should always study it further.

We decided to construct our application using the Model-View-Controller design pattern since it not only helps us avoid several problems that would have occurred otherwise but also makes it easier to maintain, extend, and expand the program. Model-View-Controller (often abbreviated as MVC) is another name for the MVC design pattern. It is a standard architectural pattern that is utilized in the process of designing and developing user interfaces as well as the structure of applications.

The application is broken up into three distinct sections by this structure, all of which are interdependent and linked to one another. These designs are utilized to differentiate the presentation of data from how the data is accepted by the user to the data that is shown by the system. It is now normal practice to apply these design principles when building graphical user interfaces (GUIs) and online applications.

This model is used by most app developers nowadays. It assists in code reuse and parallel development. The task is made simpler and less complicated because of this. The MVC design pattern results in components that are, by their very nature, self-contained units. Because of this feature, developers can easily and quickly reuse components and scripts in a variety of different applications.

Spring Boot MVC, MySQL Server, SQL Workbench, HTML, Cascading Style Sheets, Bootstrap, jQuery, and JavaScript are the technologies that we use for the project depending on the needs.

6.1 Functional Area A/Design Unit A

6.1.1 Functional Overview

Requirement Specification is a major step in the software development life cycle. Although various software and hardware specifications can be used to develop this application, the software and hardware specifications that I have used to develop this Online Learning Management System are mentioned below:
Software requirements:

➢ Operating System: Windows 10

➢ IDE/Text Editor: IntelliJ, Git Bash CLI

➢ Application Server: Tomcat

➢ Frameworks/APIs: Spring boot, Spring Security, MySQL Workbench

➢ Database: MongoDB

➢ Front End: HTML5, CSS3, JavaScript, jQuery

➢ Web Service: Spring MVC

➢ Browser: Preferable Google Chrome or Mozilla Firefox

6.1.2 Impacts

The conventional method, which centers on instruction delivered in a classroom setting, is rapidly losing its relevance and becoming less effective as a result of developments in technology and the always-shifting requirements of today's students. The implementation of Learning Management Systems in educational settings provides an intelligent alternative for educational institutions to consider in such a scenario. This enables educators to provide students with content that is customized to their needs, to leverage a variety of pedagogical models, and to engage their students in a manner that was not previously feasible.

7 Acknowledgments

I would like to express my gratitude to Yunchuan Liu, my major professor, for having trust in me and my talents and for continuing to have faith that I would be able to complete the project on time thanks to his unwavering confidence in me. The completion of this Project was made possible due to the encouragement, consistent coaching, and meaningful feedback that was provided. In addition, I would want to express my deepest gratitude to my guide for acting as both a member of my committee and as my academic adviser, assisting me with all of the important decisions, and having faith in my capabilities.

8 References

[1] https://www.hurix.com/learning-management-systems-in-education/
[2] Moore, M. G., & Thompson, M. M. (1990). The effects of distance learning: A summary of literature (ACSDE Research Monograph No 15.). University Park, PA: The Pennsylvania State University, The American Center for the Study of Distance Education
[3] Roschelle, J. (2003). Learning by collaborating: conceptual change. The Journal of the Learning Sciences, 2, 235-276.
[4] Stratmann E, Ousterhout J, Madan S (2011) Integrating long polling with an MVC framework. Stanford University, USA, p. 10.

9 Appendices

1. https://www.tutorialspoint.com/spring/spring_web_mvc_framework.htm
2. https://www.baeldung.com/spring-mvc-tutorial
3. https://www.w3schools.com/bootstrap4/
4. https://www.w3schools.com/sql/