

UNDERGRADUATE PROJECT REPORT

|  |  |
| --- | --- |
| **Project Title:** | **Classroom Reservation Management System** |
| **Surname:** | **Chen** |
| **First Name:** | **Hanly** |
| **Student Number:** | **201918020224** |
| **Supervisor Name:** | **Albert Xu** |
| **Module Code:** | **CHC 6096** |
| **Module Name:** | **Project** |
| **Date Submitted:** | **May 5, 2023** |

**Chengdu University of Technology Oxford Brookes College**

**Chengdu University of Technology**

**BSc (Single Honours) Degree Project**

Programme Name: **Software Engineering**

Module No.: **CHC 6096**

Surname: **Chen**

First Name: **Hanly**

Project Title: **Classroom Reservation Management System**

Student No.: **201918020224**

Supervisor: **Albert Xu**

2ND Supervisor (if applicable): **Not Applicable**

Date submitted: **May 5, 2023**

*A report submitted as part of the requirements for the degree of BSc (Hons) in Software Engineering*

*At*

**Chengdu University of Technology Oxford Brookes College**

**Declaration**

**Student Conduct Regulations**:

Please ensure you are familiar with the regulations in relation to Academic Integrity. The University takes this issue very seriously and students have been expelled or had their degrees withheld for cheating in assessment. It is important that students having difficulties with their work should seek assist from their tutors rather than be tempted to use unfair means to gain marks. Students should not risk losing their degree and undermining all the work they have done towards it. You are expected to have familiarised yourself with these regulations.

<https://www.brookes.ac.uk/regulations/current/appeals-complaints-and-conduct/c1-1/>

Guidance on the correct use of references can be found on www.brookes.ac.uk/services/library, and also in a handout in the Library.

The full regulations may be accessed online at <https://www.brookes.ac.uk/students/sirt/student-conduct/>

If you do not understand what any of these terms mean, you should ask your Project Supervisor to clarify them for you.

**I declare that I have read and understood Regulations C1.1.4 of the Regulations governing Academic Misconduct, and that the work I submit is fully in accordance with them**.

Signature  Date 2023.5.4

REGULATIONS GOVERNING THE DEPOSIT AND USE OF OXFORD BROOKES UNIVERSITY MODULAR PROGRAMME PROJECTS AND DISSERTATIONS

Copies of projects/dissertations, submitted in fulfilment of Modular Programme requirements and achieving marks of 60% or above, shall normally be kept by the Oxford Brookes University Library.

**I agree that this dissertation may be available for reading and photocopying in accordance with the Regulations governing the use of the Oxford Brookes University Library.**

Signature  Date 2023.5.4

# **Acknowledgment**

I would like to express my sincere gratitude to my supervisor, Albert Xu, for his unwavering guidance, patience, and support throughout my journey in the software engineering program at Chengdu University of Technology. Your mentorship and insights have been invaluable in shaping my academic and personal growth.

I would also like to extend my heartfelt thanks to all the professors and instructors who have contributed to my education, challenged me to reach my potential, and provided me with the tools to succeed in my career.

Furthermore, I would like to acknowledge the support of my family and friends, who have been a constant source of encouragement and motivation.

Finally, I want to thank all those who have assisted me in carrying out my final project and enabled me to achieve my academic goals. Thank you for your time, effort, and expertise.

# **Table of Contents**

[Acknowledgment iv](#_Toc2809)

[Table of Contents v](#_Toc4982)

[Abstract vi](#_Toc24258)

[Abbreviations vii](#_Toc23496)

[Glossary ix](#_Toc24677)

[Chapter 1 Introduction 1](#_Toc22066)

[1.1 Background 1](#_Toc17865)

[1.2 Aim 2](#_Toc12348)

[1.3 Objectives 2](#_Toc173)

[1.4 Project Overview 2](#_Toc28082)

[1.4.1 Scope 2](#_Toc8841)

[1.4.2 Audience 3](#_Toc2949)

[Chapter 2 Background Review 3](#_Toc2173)

[Chapter 3 Methodology 8](#_Toc15996)

[3.1 Approach 8](#_Toc22456)

[3.1.1 Development Model 12](#_Toc21642)

[3.1.2 Requirements Analysis 13](#_Toc22003)

[3.2 Technology 17](#_Toc13772)

[3.3 Project Version Management 18](#_Toc18170)

[Chapter 4 Results 20](#_Toc10870)

[4.1 Design and Implementation 20](#_Toc32605)

[4.2 Testing and Evaluation 26](#_Toc30016)

[4.2.1 Test Principles 28](#_Toc7612)

[4.2.2 Test Purpose 29](#_Toc3595)

[4.2.3 Test Methods 30](#_Toc10053)

[4.2.4 Test Plan 30](#_Toc26325)

[Chapter 5 Professional Issues 36](#_Toc26976)

[5.1 Project Management 36](#_Toc13626)

[5.1.1 Activities 36](#_Toc20907)

[5.1.2 Schedule 37](#_Toc19730)

[5.1.3 Project Data Management 40](#_Toc13979)

[5.1.4 Project Deliverables 40](#_Toc15600)

[5.2 Risk Analysis 41](#_Toc7825)

[5.3 Professional Issues 47](#_Toc9091)

[Chapter 6 Conclusion 49](#_Toc16412)

[References 51](#_Toc11570)

# **Abstract**

This academic paper presents the development of a Classroom Reservation Management System, aimed at improving the process of managing classroom reservations in universities. The objectives of the system are to complete a background review of the existing classroom reservation management system, divide clear functions according to roles, develop the teacher administrator management function to display user usage in real-time, develop the function of students' classroom reservation, which can dynamically display relevant data, and demonstrate the works to mixed audiences. The paper summarizes existing approaches in the field and provides a comparison of popular classroom reservation management systems in terms of their application and technology. The proposed system will be developed using Java, JSP, SSM, and B/S architecture and will utilize a MySQL database. The software development methodology used will be the Spring Application Framework (SAF) and the Agile methodology. The development model will utilize a sprint-based model lasting 16 weeks, and the requirements analysis phase will gather information about the requires and goals of the users through interviews, surveys, and workshops. The system's primary audience is university teacher administrators and students, and the goal is to improve the efficiency and ease of managing classroom reservations for all stakeholders involved.

***Keywords:***  Classroom Reservation Management System, Java, JSP, SSM, B/S architecture, MySQL database, Agile software development methodology, Spring Application Framework.

# **Abbreviations**

CRMS: Classroom Reservation Management System

MVC: Model-View-Controller (a software design pattern)

SQL: Structured Query Language

SAF:Spring Application Framework

LOC:Inversion of Control

ER: Entity-Relationship

TDD:Test-driven development

GDRP:General Data Protection Regulation

BCS:British Computer Society

ACM: Association for Computing Machinery

Java: Java Programming Language

Eclipse: Eclipse Integrated Development Environment (IDE)

JSP: JavaServer Pages

SSM: Spring + Spring MVC + MyBatis

MyBatis: MyBatis Object Relational Mapper (ORM) Framework

HTML: Hypertext Markup Language

CSS: Cascading Style Sheets

JavaScript: ECMAScript (JavaScript) Programming Language

# **Glossary**

Classroom Reservation Management System (CRMS): A software system that allows users to reserve classrooms for events, meetings, or other activities.

Model-View-Controller (MVC): A software architectural pattern that separates an application into three interconnected components: the model, the view, and the controller.

Structured Query Language (SQL): A standard programming language for managing relational databases.

Spring Application Framework(SAF):A powerful platform for building enterprise-level Java applications. It provides a comprehensive set of tools and features for developing, testing, and deploying applications. SAF includes support for MVC architecture, Inversion of Control (IoC), and other key design patterns.

Inversion of Control (LOC):A programming principle that promotes loose coupling and modular design. It involves separating the creation and management of objects from their use. Instead of creating objects directly, IoC containers are used to manage object creation and lifecycle. This makes it easier to change or replace objects without affecting the rest of the application.

Entity-Relationship (ER):A technique used to design and represent relationships between entities in a database. ER diagrams are used to visually represent the relationships between different data entities, including tables, columns, and keys.

Test-driven development (TDD):A software development methodology that involves writing automated tests before writing code. TDD assists ensure that the code is reliable, maintainable, and meets all of the requirements. It also encourages developers to write modular, testable code.

General Data Protection Regulation (GDPR): A set of regulations established by the European Union (EU) to protect the privacy rights of individuals. The GDPR applies to all organizations that process personal data of EU citizens, regardless of their location or size. Organizations that fail to comply with GDPR regulations may face substantial fines and legal penalties.

British Computer Society (BCS): A professional organization for IT professionals. The BCS provides support, training, and certification programs to assist members advance their careers in the field of technology.

Association for Computing Machinery (ACM): An international scientific and educational organization dedicated to advancing the art, science, engineering, and application of information technology. The ACM provides resources and publications to assist members stay up-to-date on the latest trends and developments in the field.

Java Programming Language(JAVA): A high-level programming language that is used to develop applications for a variety of platforms, including desktops, mobile devices, and servers. It is known for its "write once, run anywhere" philosophy, which allows code written in Java to be executed across multiple platforms without modification.

Eclipse Integrated Development Environment(Eclipse ) :An open-source IDE that is widely used for developing software in Java and other programming languages. It offers a wide range of tools and features for editing, debugging, and testing code.

JavaServer Pages(JSP):A technology that allows developers to create dynamic web pages using Java code. It is often used in conjunction with servlets to create Java-based web applications.

Spring + Spring MVC + MyBatis(SSM):A popular framework for building web applications in Java. It combines three key technologies: Spring, a lightweight framework for building enterprise Java applications; Spring MVC, a framework for building web applications; and MyBatis, a data access framework that simplifies database programming.

MyBatis Object Relational Mapper (ORM) Framework(MyBatis ): A lightweight ORM framework that simplifies database programming in Java. It allows developers to map Java objects to SQL statements, making it easier to work with relational databases.

Hypertext Markup Language(HTML):The standard markup language used to create web pages. It consists of a set of tags that define the structure and content of a web page.

Cascading Style Sheets(CSS):A stylesheet language used to describe the presentation of a web page. It is used to control the layout, typography, and colors of HTML elements.

ECMAScript (JavaScript) Programming Language(JavaScript):A high-level programming language that is used to create interactive web pages and web-based applications. It is often used in conjunction with HTML and CSS to create dynamic, responsive user interfaces.

# **Introduction**

## **Background**

The electronic information society has become increasingly prevalent, with many colleges and universities transitioning from paper management to electronic information management. (Resmark, 2020) However, many online classroom reservation systems currently available are found to be overly complex and lacking in user-friendly interface. This can create difficulties for both students who are required to submit attendance records and teacher administrators who require access to student data. (Creatrix, 2022) In order to address these issues, this project aims to design and implement a better classroom reservation management system based on the Spring Application Framework, using the Sprint model.(Max, 2023) The system will feature a report function and dynamic data display to allow for more efficient and concise management of university classroom information. (Programmerall, 2022)

The system will be user-friendly and easy to navigate, with a simple interface that allows students and teacher administrators to quickly and easily reserve or manage classrooms. (Merriam, 2022)The system will also include features such as teacher administrator management and student reservation functions, which will allow teacher administrators to manage their own classrooms and student to quickly and easily book a classroom for their next class. Additionally, the system will include a reporting function that will allow teacher administrators and administrators to easily view and analyze data on classroom usage and occupancy.

Furthermore, the system will be designed to be highly customizable, allowing universities to easily tailor it to their specific requires. This will include options for different reservation protocols and policies, as well as the ability to integrate with other existing systems and technologies. (Programmerall, 2022) With this new system, universities will be able to streamline their classroom reservation process and improve the overall efficiency of their operations.

Overall, this project aims to design and implement a better classroom reservation management system based on the Spring Application Framework, using the Sprint model. The system will be user-friendly and easy to navigate, with a simple interface that allows students and teacher administrators to quickly and easily reserve or manage classrooms. It will also include features such as teacher administrator management and student reservation functions, which will allow teacher administrators to manage their own classrooms and student to quickly and easily book a classroom for their next class.

## **Aim**

To design and implement a better classroom reservation management system that aims to assist teacher administrators and students in managing and scheduling class reservations more efficiently.

## **Objectives**

The objectives of the system are as follows:

1. completes the background review of the existing classroom reservation management system.

2. divides clear functions according to roles to facilitate data management.

3. has developed the teacher administrator management function to display the user's usage in real time.

4. develops the function of students' classroom reservation, which can dynamically display relevant data of the system.

5. demonstrates the works to mixed audiences.

## **Project Overview**

### **Scope**

The scope of this project is to design and implement a classroom reservation management system. The system will allow teacher administrators to login and manage classroom information, review student's appointment applications, and view past appointment records. It will also allow students to login, view classroom information, make appointments, and modify their personal information. The system will be developed using Java, JSP, SSM, and B/S architecture and will utilize a MySQL database. The significance of this project lies in its ability to improve the current classroom reservation process by providing a more efficient and user-friendly system for both teacher administrators and students. It aims to improve the communication and information exchange between them.

### **Audience**

The primary audience for this project is university teacher administrators and students. teacher administrators will benefit from the system's administrator login function, which allows them to access the classroom reservation management system and manage classroom details, review student appointment applications, view past appointment records, and register their own and students' accounts. Students will benefit from the system's user login function, which allows them to view classroom details, view their past appointment records, make classroom reservations, and modify their personal information. Additionally, this system will also be useful for administrators and staff members who are responsible for managing the classrooms and appointments at the university level.Therefore, the goal is to improve the efficiency and ease of managing classroom reservations for all stakeholders involved.

# **Background Review**

The background review for this classroom reservation management system project aims to summarize existing approaches in the field, specifically focusing on the software development methodologies and systems that have been previously employed.

According to Ralph (2021), the traditional approach for teacher administrator appointment systems has been to use the waterfall model for development, which emphasizes cooperation and exporting results to various formats for storage. However, this approach can lead to an irreversible development process, with high costs and difficulty in meeting all user requirements, resulting in complex data and difficult user operations.

Additionally, existing systems on the market, such as Skedda (2022), have the capability to arrange students' schedules. However, these systems tend to have complex pages and not simple and efficient sufficient operation pages.(Bagyatech,2020) In contrast, the proposed classroom reservation management system aims to develop a report function that can dynamically display relevant data, and employs innovative functional methods in order to optimize the current systems on the market. By making the pages simple and beautiful and the operation pages simple and efficient, the system aims to better cater to the requires of students and busy teacher administrators and improve its competitiveness in the field.

To further understand the current state of the market, a comparison of popular classroom reservation management systems, such as Microsoft Bookings, Doodle, Calendly, TimeCenter, and Acuity Scheduling, was conducted. A table was created to compare the strengths and weaknesses of each system in terms of their application and technology. This analysis revealed that while each system had its own unique features and capabilities, there was still room for improvement in terms of user-friendliness and efficient data management. In light of these findings, the proposed classroom reservation management system aims to address these issues and provide a more user-friendly and efficient solution for teacher administrators and students.

| **Logo** | **System** | **Strengths** | **Weaknesses** |
| --- | --- | --- | --- |
|  | Microsoft Bookings | 1.Integration with Office 365 and Outlook calendar.  2.Customizable booking forms and confirmation emails.  3. Ability to set up recurring appointments. | 1. Limited customization options for the user interface.  2. Limited reporting and data analysis capabilities. |
|  | Doodle | 1.Easy to use and navigate.  2.Ability to create polls for scheduling meetings.  3.Integration with Google Calendar and Outlook.  4. Advanced scheduling options such as buffer times and group events. | 1. Limited customization options for the user interface.  2. Limited integration with other systems. |
|  | Calendly | 1.Easy to use and navigate.  2. Customizable booking forms and confirmation emails.  3. Ability to set up recurring appointments.  4. Advanced reporting and analytics capabilities. | 1. Limited reporting and data analysis capabilities.  2. There is not sufficient report generation capability to retain abreast of classroom usage. |
|  | TimeCenter | 2.Easy to use and navigate.  2.Customizable booking forms and confirmation emails.  3. Ability to set up recurring appointments. | 1. Limited customization options for the user interface.  2. There is not sufficient report generation capability to retain abreast of classroom usage. |
|  | Acuity Scheduling | 1.Customizable booking forms and confirmation emails.  2.Ability to set up recurring appointments.  3.Integration with other systems such as Zapier and Stripe.  4. Advanced scheduling options such as appointment types and staff availability. | 1. Limited customization options for the user interface.  2. Limited integration with other systems. |

Table 1: Comparison table of market CRMS

The Table 1 above is a comparison table of popular Classroom Reservation Management System (CRMS) solutions currently available in the market. The table compares Microsoft Bookings, Doodle, Calendly, TimeCenter, and Acuity Scheduling in terms of their application and technology.

Microsoft Bookings offers integration with Office 365 and Outlook calendar, customizable booking forms, and the ability to set up recurring appointments. However, it has limited customization options for the user interface and reporting and data analysis capabilities.

Doodle is easy to use and navigate, offers advanced scheduling options, and integrates with Google Calendar and Outlook. However, it has limited customization options for the user interface and integration with other systems.

Calendly is another user-friendly CRMS solution that offers customizability and recurring appointment options, along with advanced reporting and analytics capabilities. Its limitations include limited reporting and data analysis capabilities and insufficient report generation capability to retain up with classroom usage.

TimeCenter offers customizable booking forms, confirmation emails, and the ability to set up recurring appointments. However, like Calendly, it has limited customization options for the user interface and report generation capability.

Acuity Scheduling offers customizable booking forms, the ability to set up recurring appointments, and integration with other systems such as Zapier and Stripe. It also has advanced scheduling options, but its limitations include limited customization options for the user interface and integration with other systems.

Therefore, Table 1 provides valuable information for organizations seeking a suitable CRMS solution based on their specific requires and requirements. By comparing the strengths and weaknesses of each system, organizations can make an informed decision and choose the most appropriate CRMS solution for their classroom reservation management requires.

# **Methodology**

## **Approach**

The approach for this software development project will be based on the Spring Application Framework (SAF) and the Agile software development methodology. (Spring, 2023)The SAF is a popular Java framework used for building enterprise applications. It provides a comprehensive set of features for web and enterprise applications and allows for the use of common design patterns such as Model-View-Controller (MVC) and Inversion of Control (LOC). (wikipedia , 2022)The Agile methodology will be used to manage and organize the development process, ensuring that the project is delivered in a timely and efficient manner.The following is the MVC framework structure and technical roadmap of this project:

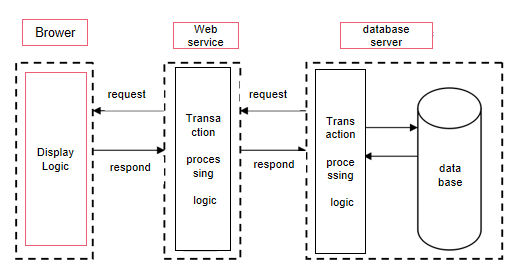


Figure 1 MVC Frame Structure of the Third Floor

The Figure1 above represents the architecture of the proposed classroom reservation management system using the Model-View-Controller (MVC) framework. (wikipedia, 2023)The MVC pattern is a widely used architectural pattern for developing web applications that provide a clear separation of concerns and promotes maintainability, scalability, and extensibility in the development process.

In this figure, the Browser represents the user interface where the user interacts with the system. The Request/Response cycle represents the communication between the browser and the server. The Server is responsible for processing the user's request and returning a response.

The service layer handles the business logic of the application, while the Transaction layer manages database transactions. The model layer represents the data and the business logic of the CRMS system, including the reservation details, user information, room information, etc. The Controller acts as an intermediary between the Model and the View. It receives input from the user, interacts with the Model to update or retrieve data, and updates the View accordingly.

The Display Logic and Data Processing Logic are responsible for rendering the view and processing data, respectively. The view layer represents the user interface of the CRMS system, which displays relevant information to the user.

Therefore, Figure1 provides a comprehensive overview of the CRMS system architecture, and the MVC framework structure assists to ensure that the system is scalable, maintainable, and flexible sufficient to accommodate future changes and developments.

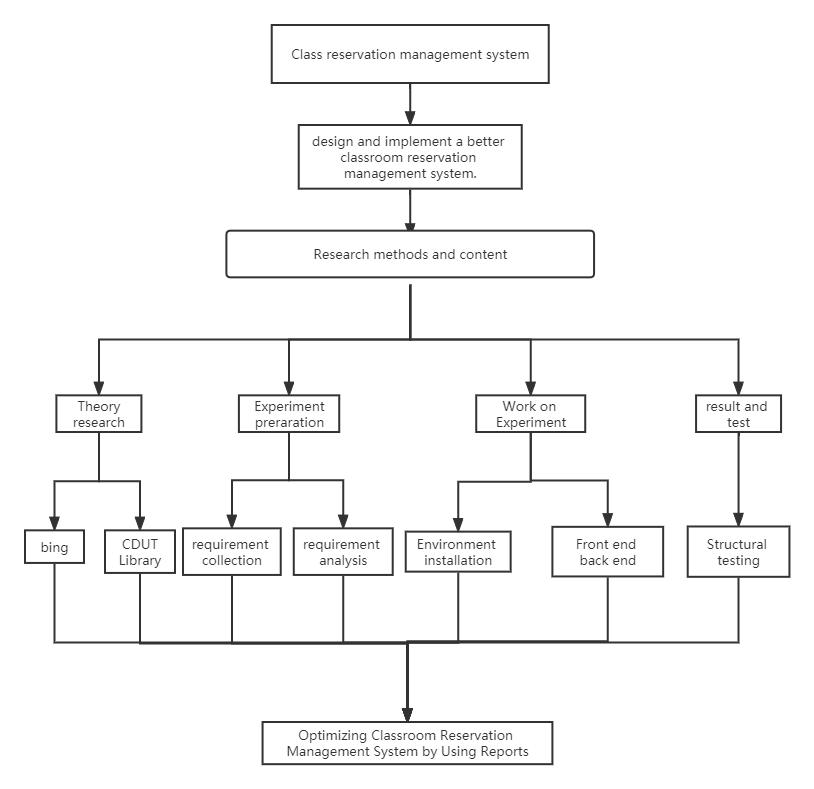


Figure 2 CRMS Technical Roadmap

The Figure 2 above is the technical roadmap for the development of the Classroom Reservation Management System (CRMS). It outlines the different phases and tasks involved in the development process. The roadmap starts with the requirement gathering phase, where the project team will gather and analyze the requirements of the system. Then it moves on to the environment setup phase, where the necessary infrastructure and tools will be set up for development.

Subsequently, the front-end and back-end development phases are demonstrated, which involve designing and developing the user interface and the system's functionality, respectively. The structural analysis phase involves analyzing the system's structure and identifying any potential issues. The library collection phase includes the selection and integration of third-party libraries that can enhance the system's functionality.

The testing phase involves thorough testing of the system to ensure that it meets all the requirements and works as intended. Finally, the optimization phase covers the use of reports to identify areas of the system that can be improved and optimized to provide a better experience for users.

Therefore, this technical roadmap provides a high-level overview of the different stages and tasks involved in the development of CRMS, using the Spring Application Framework.

## **3.1.1 Development Model**

The project will utilize a sprint-based development model, which is a popular Agile methodology used in software development. This model involves breaking the project down into smaller, manageable chunks called sprints, each of which lasts about one week for a total of 16 weeks in duration. Each sprint includes a planning phase, development phase, testing phase, and review phase. By breaking the project into smaller chunks, the team can better manage the scope and timelines of the project, as well as make adjustments as required.The following is the sprint development model diagram:

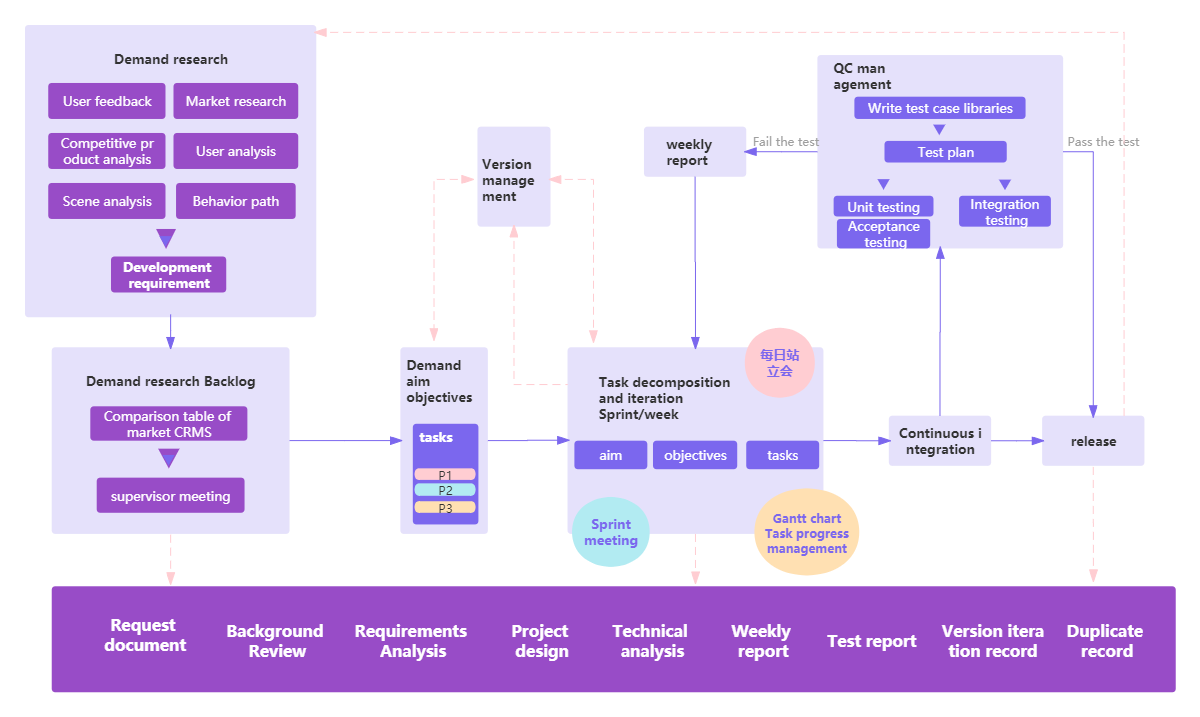


Figure 3 sprint development model diagram

The Figure 3 above depicts the sprint development model, which is an Agile methodology utilized in CRMS. The model involves dividing the project into smaller, manageable chunks called sprints. Each sprint lasts for one week, and includes a planning phase, development phase, testing phase, and review phase. By breaking down the project into smaller parts, the team can better manage the scope and timelines of the project, as well as make adjustments as required. The figure illustrates the various stages involved in the sprint development model, including demand research, user feedback, market research, assembly, unit testing, acceptance testing, and behavior analysis. It also demonstrates the continuous release integration, technical and project management, and weekly test reports, among other components. Therefore, this model allows for an iterative approach to software development, ensuring that the final product meets the requirements of stakeholders and end-users.

## **3.1.2 Requirements Analysis**

The project will begin with a thorough requirements analysis, which will involve gathering information about the requires and goals of the users. This will be done through a variety of methods, including interviews, surveys, and workshops. The information gathered will be used to create use case diagrams and a Sequence diagram of user registration, which will outline the functional and non-functional requirements of the system.

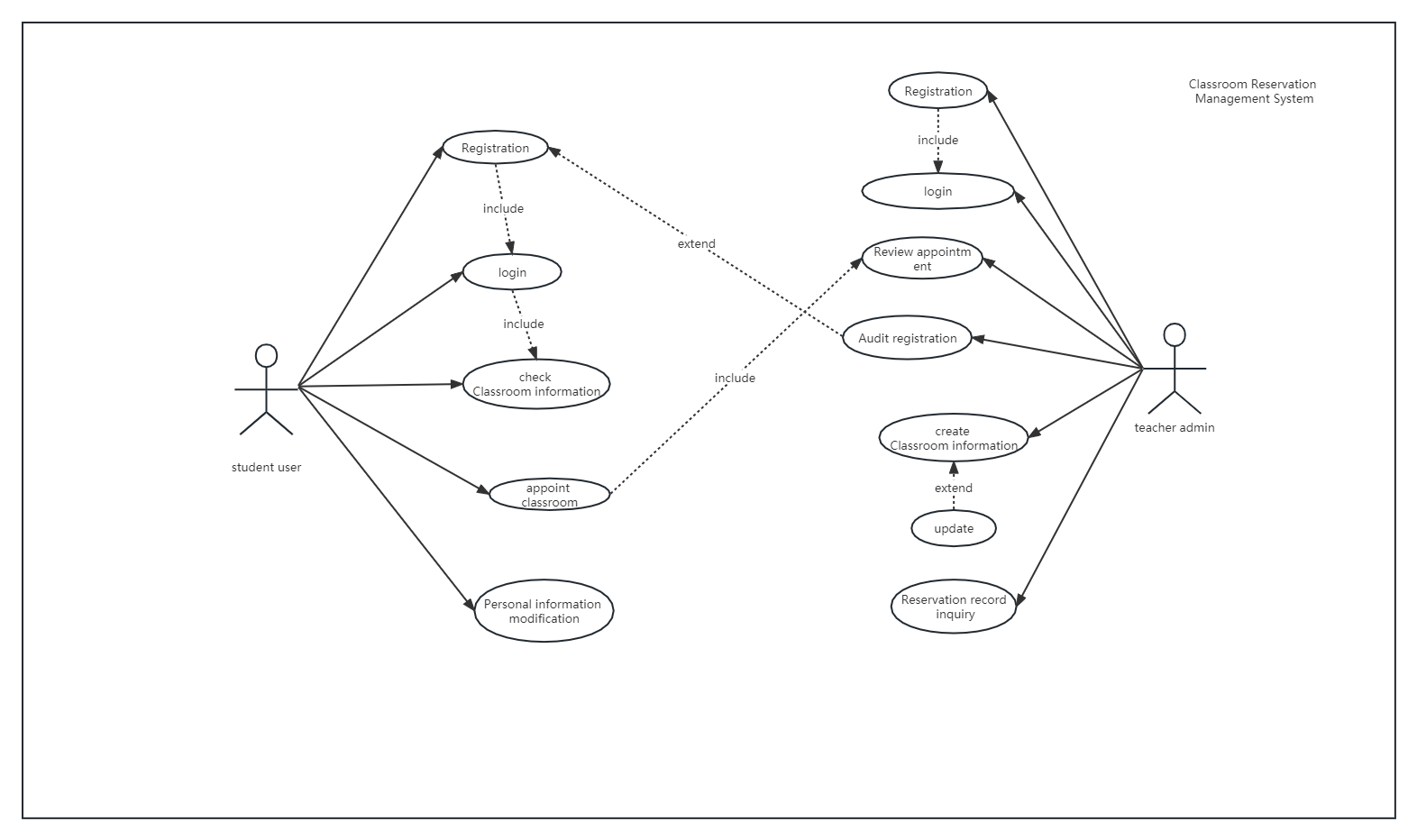


Figure 4 CRMS use case diagram

The Figure 4 above depicts the use case diagram for the Classroom Reservation Management System (CRMS). A use case diagram is a visual representation of the interactions between users (actors) and a system to accomplish specific tasks or goals.

The CRMS use case diagram outlines the various actions that can be performed by different actors, including the teacher administrator, student users, and the system itself. The main use cases include login, classroom information management, appointment scheduling, appointment application review, personal information modification, and system updates.

In the diagram, the teacher administrator actor can log in to the system, manage classroom information, review appointment applications, and update the system as required. Student users, on the other hand, can log in to the system, view classroom information, schedule appointments, modify their personal information, and make appointment requests.

Therefore, the CRMS use case diagram provides a clear and concise overview of the various interactions and tasks that can be performed by different actors within the system. It serves as an important tool for designing and implementing the CRMS system and ensuring that it meets all necessary requirements and specifications.

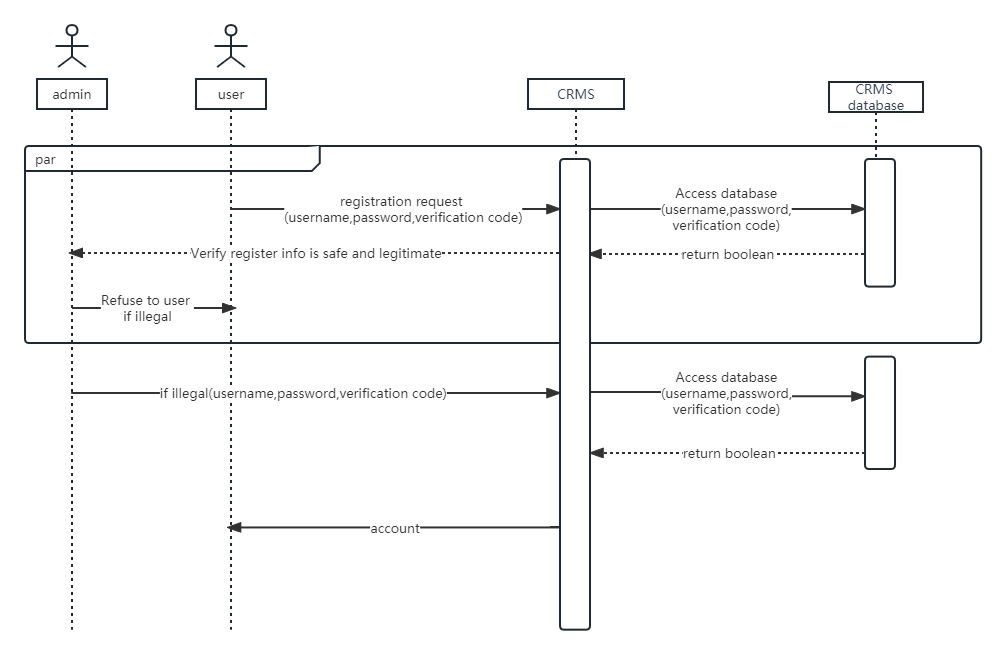


Figure 5 CRMS Sequence diagram of user registration

The Figure 5 above demonstrates the sequence diagram for user registration in the Classroom Reservation Management System (CRMS). A sequence diagram is a type of interaction diagram that illustrates the interactions between objects or actors in a specific scenario or use case.

In this sequence diagram, the actors are the administrator and the user who is attempting to register for the CRMS. The system uses a database to store and verify user registration information.

The sequence begins with the user submitting a registration request. The system then prompts the user to enter their username, password, and verification code. Once this information is entered, it is sent to the system's database for verification.

The system checks the user's information for safety and legitimacy and returns a boolean value indicating whether or not the information is valid. If the information is not valid, the system refuses registration and prompts the user to try again with correct information. If the information is valid, the system creates an account for the user and stores their information in the database.

Therefore, the CRMS Sequence diagram of user registration provides a clear and concise illustration of the steps involved in the user registration process and how the system processes and verifies user information before creating an account.

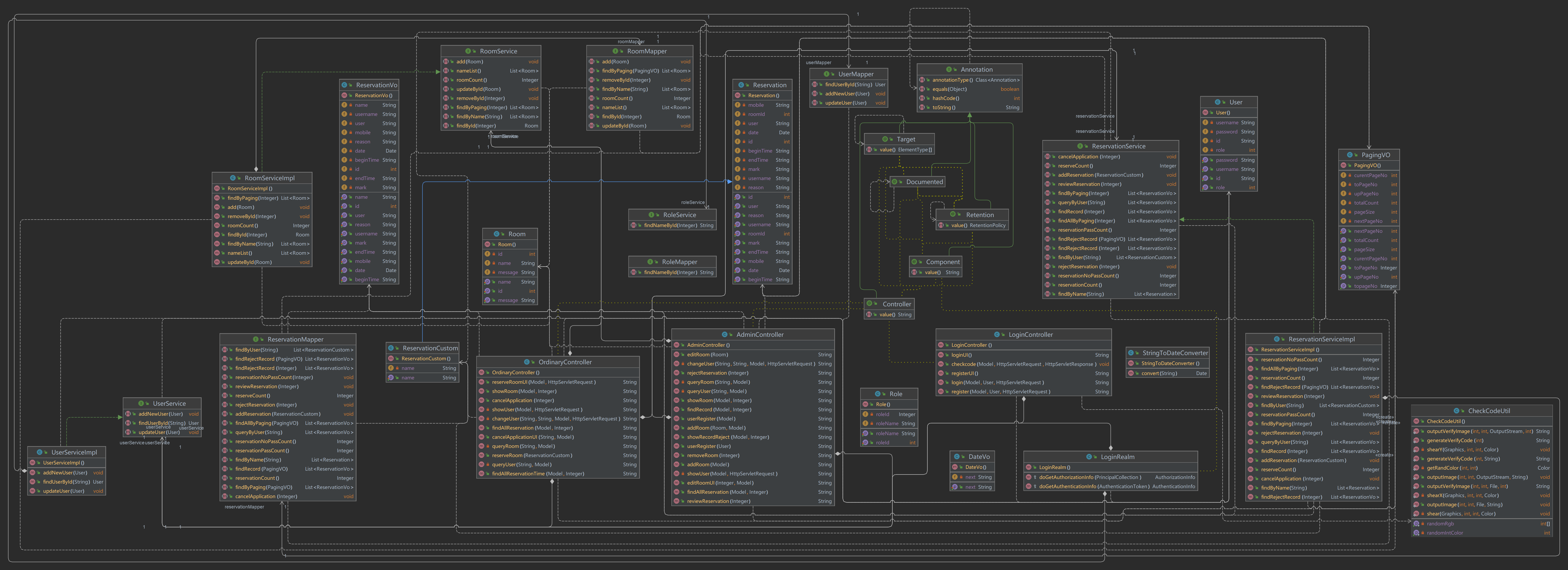


Figure 6 CRMS Class diagram

The Figure 6 above depicts the class diagram of the Classroom Reservation Management System (CRMS) project. The CRMS is a software system designed to manage and automate the process of reserving classrooms for academic and non-academic activities in an educational institution. The class diagram provides a visual representation of the different classes, their attributes, and the relationships between them.

The CRMS consists of various classes such as RomMapper, UseMapper, Annotation, Reservation, ReservationService, PagingVO, RomServiceImpl, RoleService, Component, LoginController, UserServicesImpl, ReasonMapper, and CheckCodeUtil, among others. These classes are interconnected via different types of relationships like inheritance, association, composition, and dependency.

The Reservation class is central to the CRMS and has associations with other classes like ReservationService, ReasonMapper, and ReservationCustomer, which provide additional functionality and information regarding reservations. The ReservationService class encapsulates the business logic and services related to reservations, while the ReasonMapper class maps reservation reasons to their corresponding codes.

Therefore, the CRMS class diagram represents the different components and their relationships within the system, providing a comprehensive view of its structure and functionality.

## **Technology**

In order to ensure the accuracy of the experiment, all experiments in this paper use the same platform, using dual video cards, and the memory size reaches 512G. The specific configuration is demonstrated in Table 2 below:

|  |  |
| --- | --- |
| **Configuration** | **Details** |
| Operating system | Windows |
| Development tool | Eclipse |
| Development technology: | Java |
| Database: | MySQL |
| Processor | Core i5 generation 4 series CPU main frequency, 2.7GHz dynamic, acceleration frequency: above 3.6GHz |
| Memory | 8G (or above) |
| Hard disk | 512G solid state disk (or above 512G) |

Table 2 Configuration of Experimental Platform

The Table 2 above outlines the configuration details of the experimental platform used for the development and testing of the Classroom Reservation Management System (CRMS).

The operating system used in the platform is Windows, while the development tool is Eclipse. (wikipedia, 2023)The main development technology employed is Java.(geeksforgeeks,2023) For database management, MySQL has been used.

The processor used in the platform is Core i5 generation 4 series with a CPU main frequency of 2.7GHz dynamic, and an acceleration frequency above 3.6GHz. The platform also features 8GB or more of memory and a solid-state hard disk with a minimum capacity of 512GB or higher.

Therefore, the configuration of this experimental platform provides the necessary resources to support the development and testing of the CRMS system. It ensures that the system runs smoothly and efficiently while meeting the requirements and specifications of the project.

## **Project Version Management**

In order to effectively manage the project source codes, a version management plan was implemented using GitHub, a web-based platform that enables developers to collaborate on code development and version control which is secure, user-friendly and allows easy access to project files. A dedicated repository was created to store all project files, and changes were tracked and recorded using Git, a distributed version control system.Additionally, regular backups were made to ensure that all important files were secure in case of any issues or errors.

Using features such as pull requests, code reviews, and issue tracking, efficient collaboration between the project developer and supervisor was possible. This approach also allowed for efficient collaboration between the project developer and supervisor, as all files were easily accessible and changes were clearly recorded. This ensured that the project development process was transparent, and that any problems could be identified and resolved promptly.

Regular commits were made to the repository, allowing for easy tracking of progress made in the project and making it possible to revert to previous versions if necessary. Additionally, backups were made to ensure the security of all important files. This version management plan allowed for an organized, efficient, and streamlined development process throughout the project lifecycle.

The latest version of the project source code can be accessed using the following link to the GitHub repository: <https://github.com/femsi1/Classroom_Reservation_Management_System>.

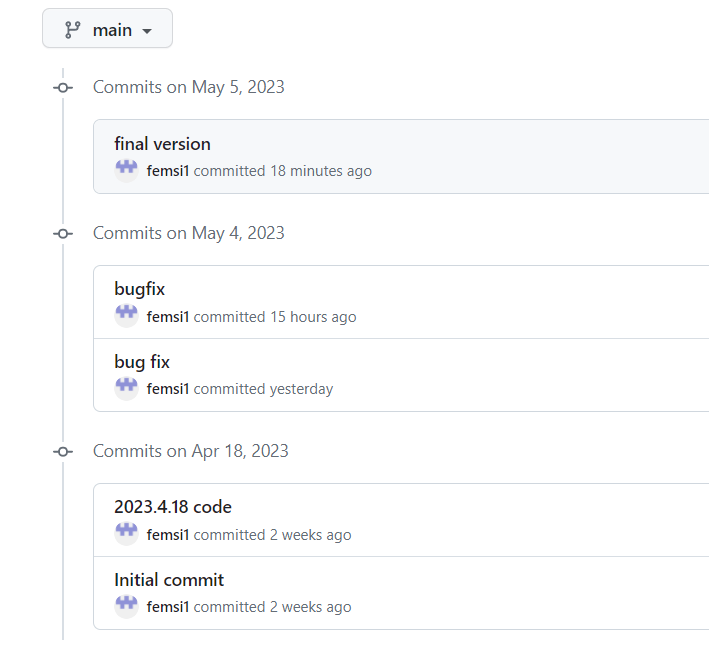


Figure 7 crms historical version

# **Results**

## Design and Implementation

The proposed classroom reservation management system will be developed using a combination of front-end and back-end technologies. The front-end will be developed using HTML, CSS, and JavaScript, while the back-end will be developed using a combination of Mybaits and MySQL.

The overall design of the system will consist of three main modules: a student module, a teacher module, and a back-end module. The student module will allow students to view available classrooms, make reservations, and view their own reservations. The teacher module will allow teachers to view and manage classroom reservations, as well as view statistics and reports on classroom usage. The back-end module will handle the database and server-side logic, including data validation and security.

The system architecture will be designed using a Model-View-Controller (MVC) pattern, with the database serving as the Model, the front-end as the View, and the back-end as the Controller. This will allow for a clear separation of concerns and easy maintenance and scalability.

The database design will consist of several tables, including a "room" table, a "reservation" table, and a "user" table,and a "role" table. The "room" table will store information about each classroom, including its name, capacity, and available equipment. The "reservation" table will store information about each reservation, including the classroom, date, and time. The "user" table will store information about each user, including their name, email, and role (student or teacher).The "role" table stores information about each role, including the role id and the role name.

Functional screenshots of each module will be provided, including the student module, teacher module, and back-end module. These screenshots will demonstrate the features and functionality of each module, including the ability to view and make reservations, view statistics and reports, and manage user accounts.

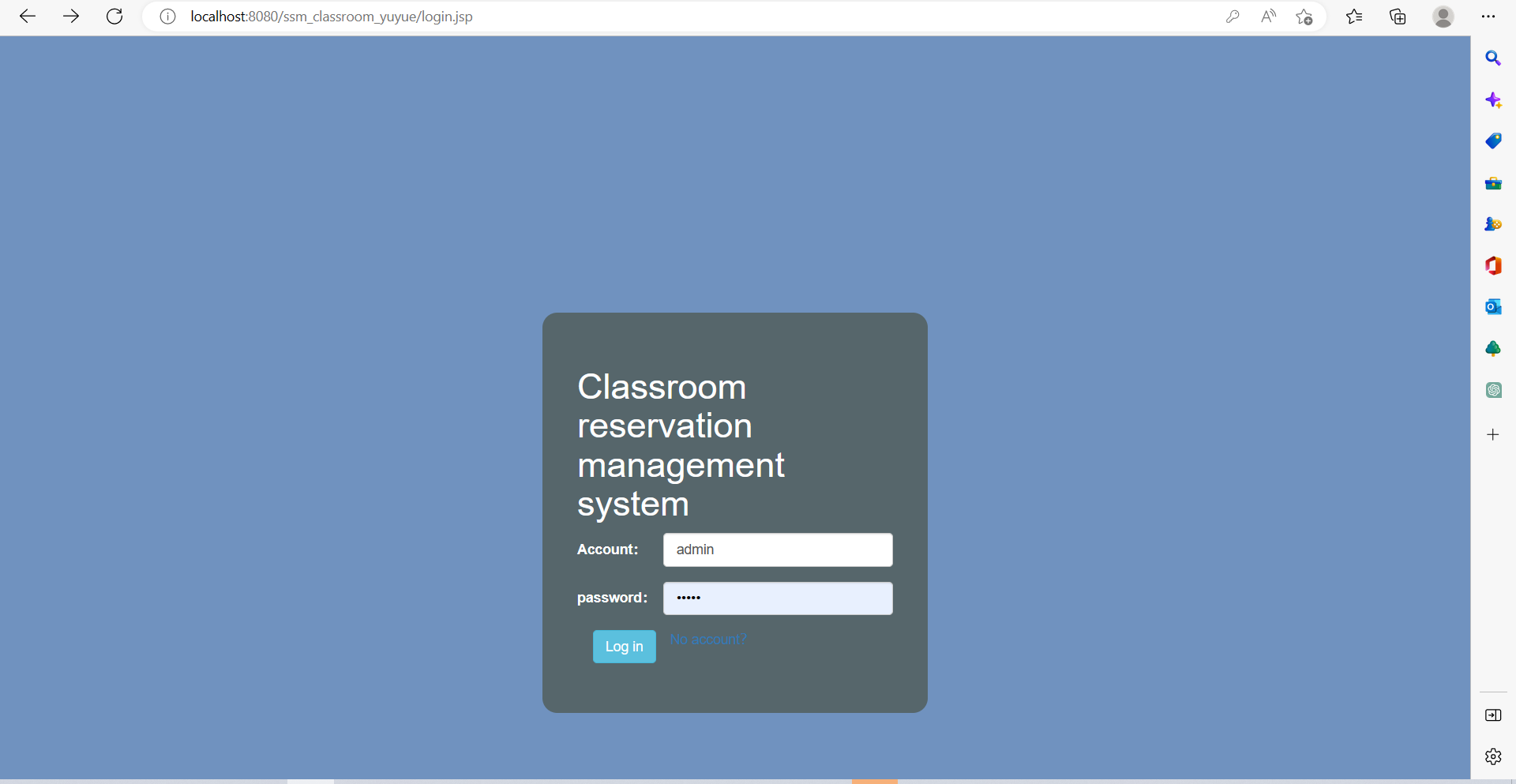


Figure 8 CRMS home and login page

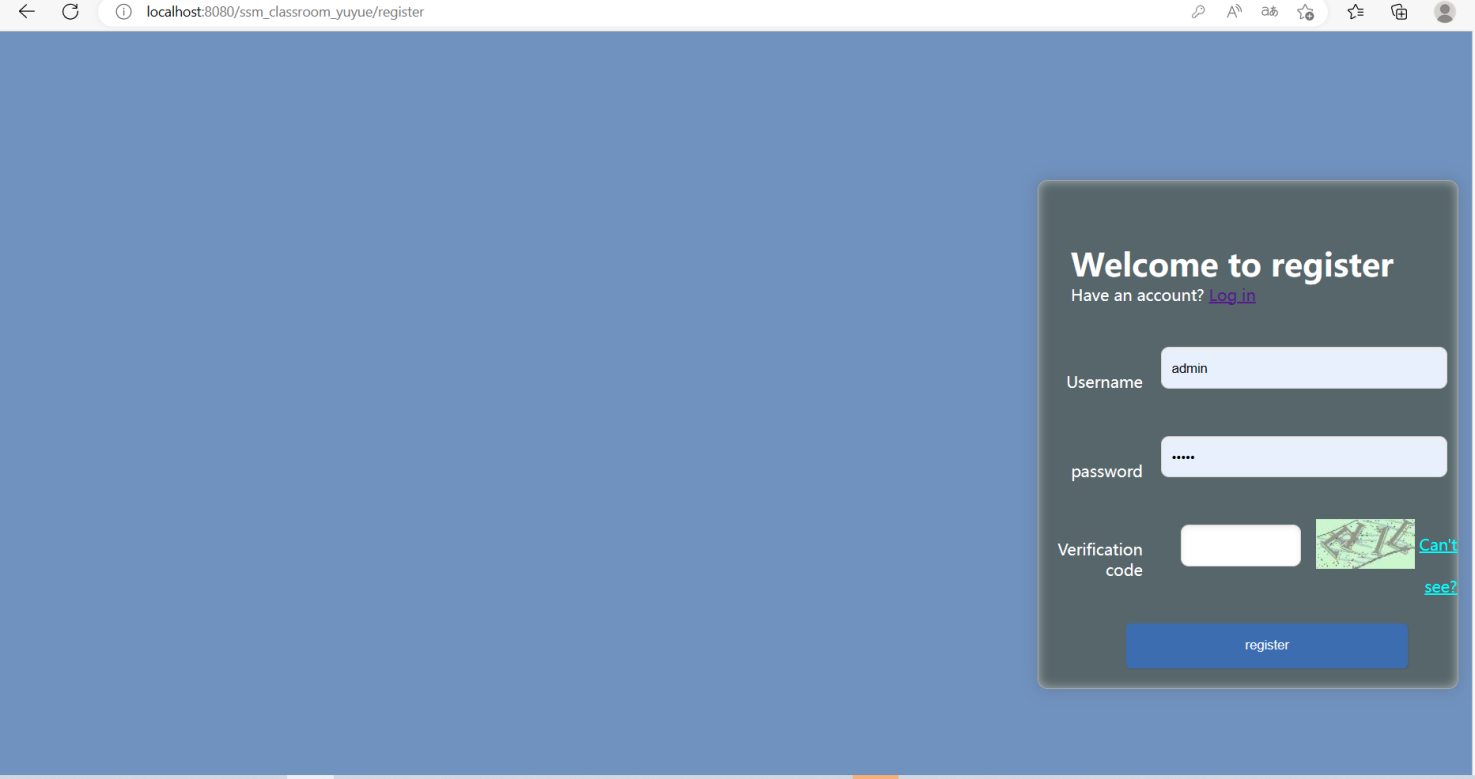


Figure 9 CRMS Registration page

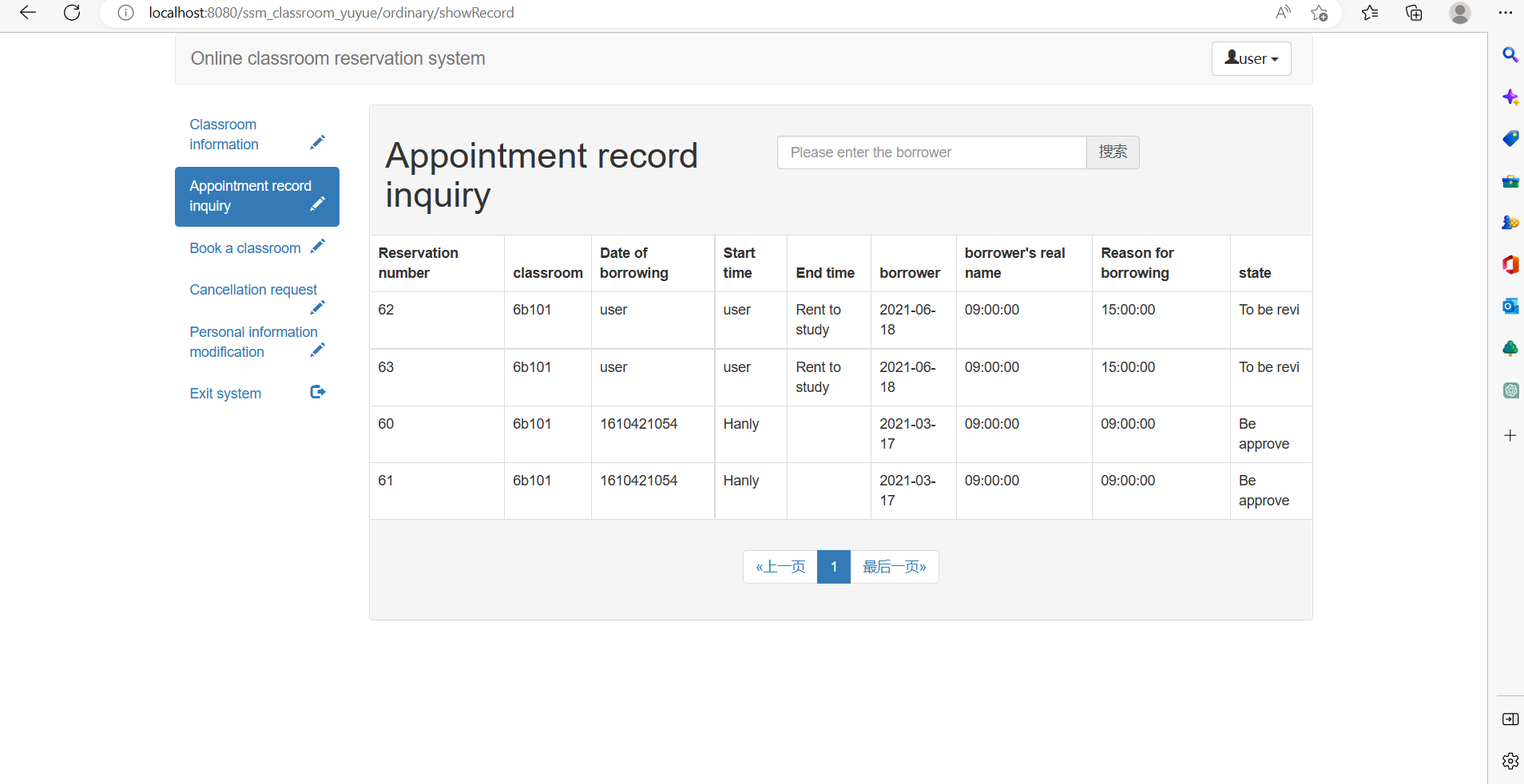


Figure 10 CRMS Student appoint Record page

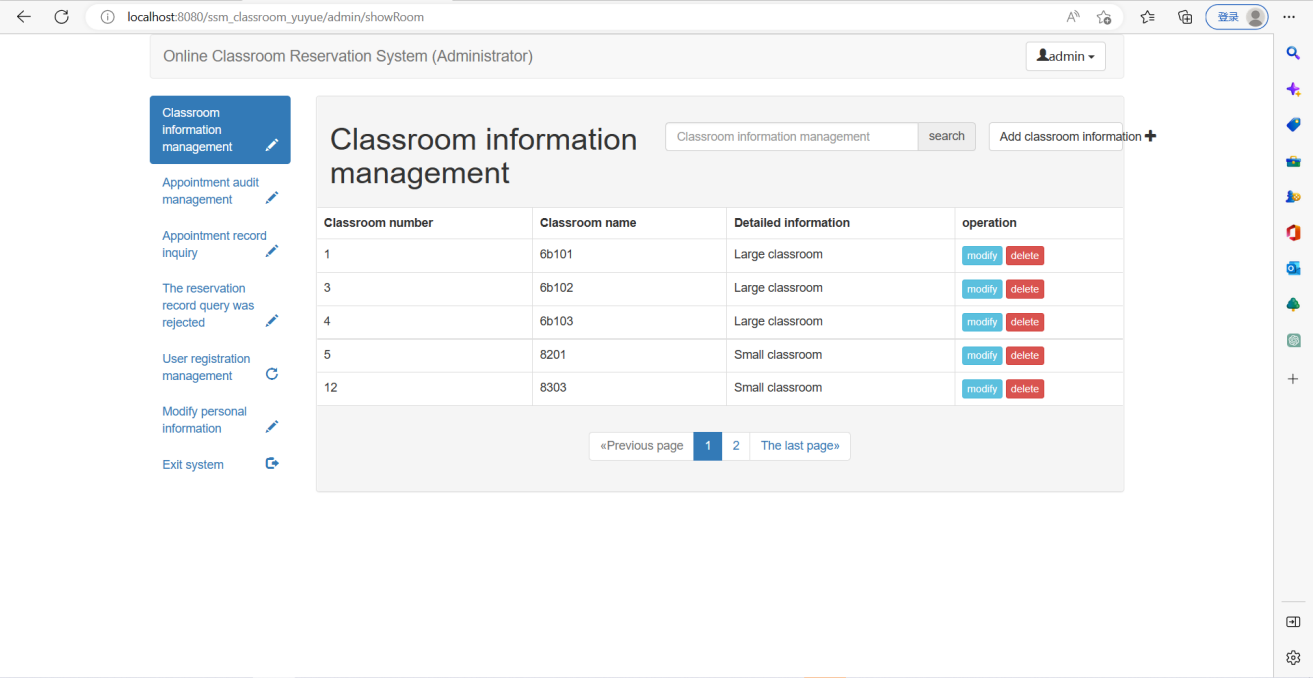


Figure 11 CRMS teacher admin room info page

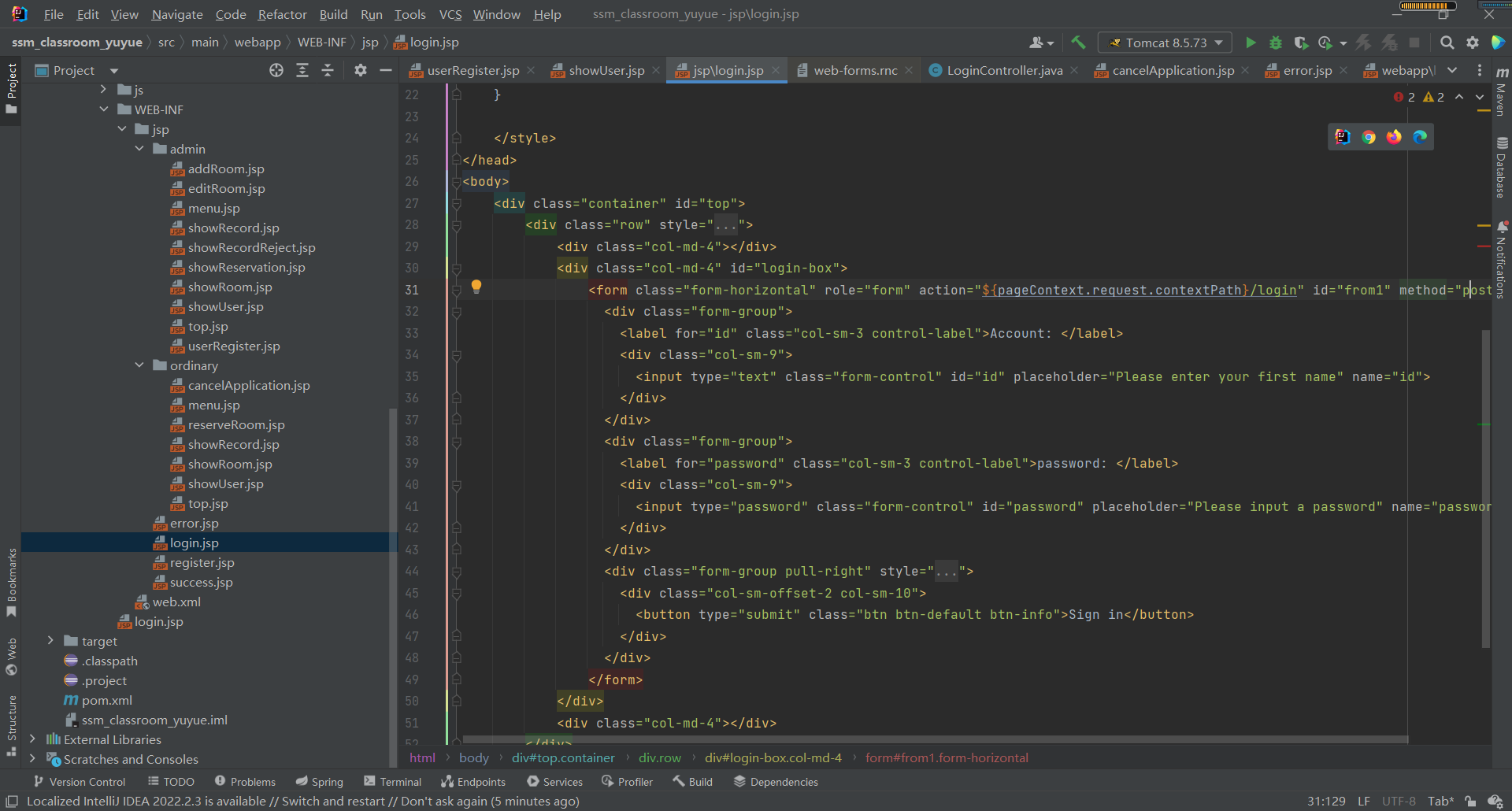


Figure 12 CRMS login front-end code

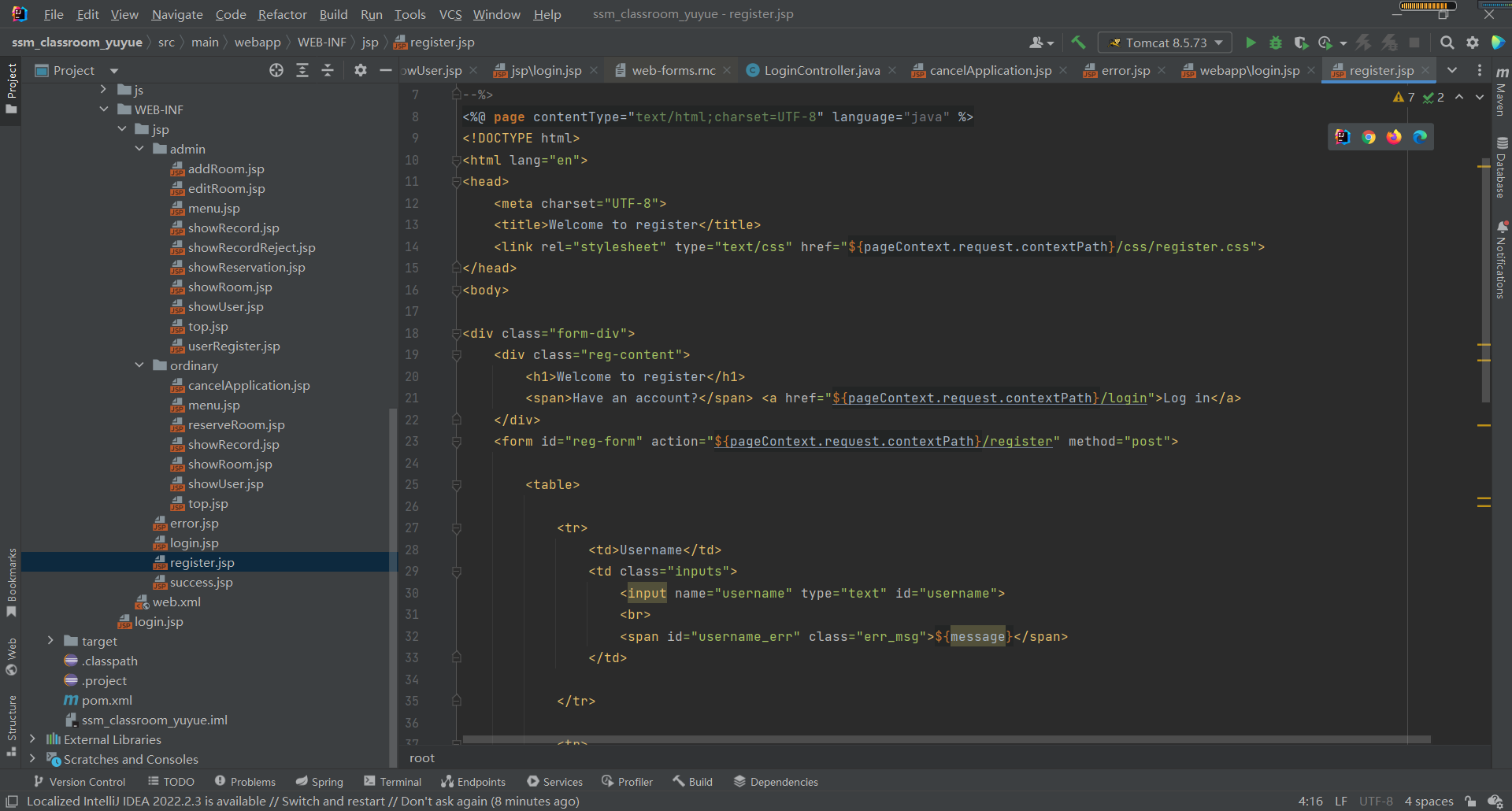


Figure 13 CRMS register front-end code

The code for the database and its connection to the back-end will also be provided, along with diagrams illustrating the relationships between the tables. This will demonstrate the work that has been done to create a functional and efficient system.

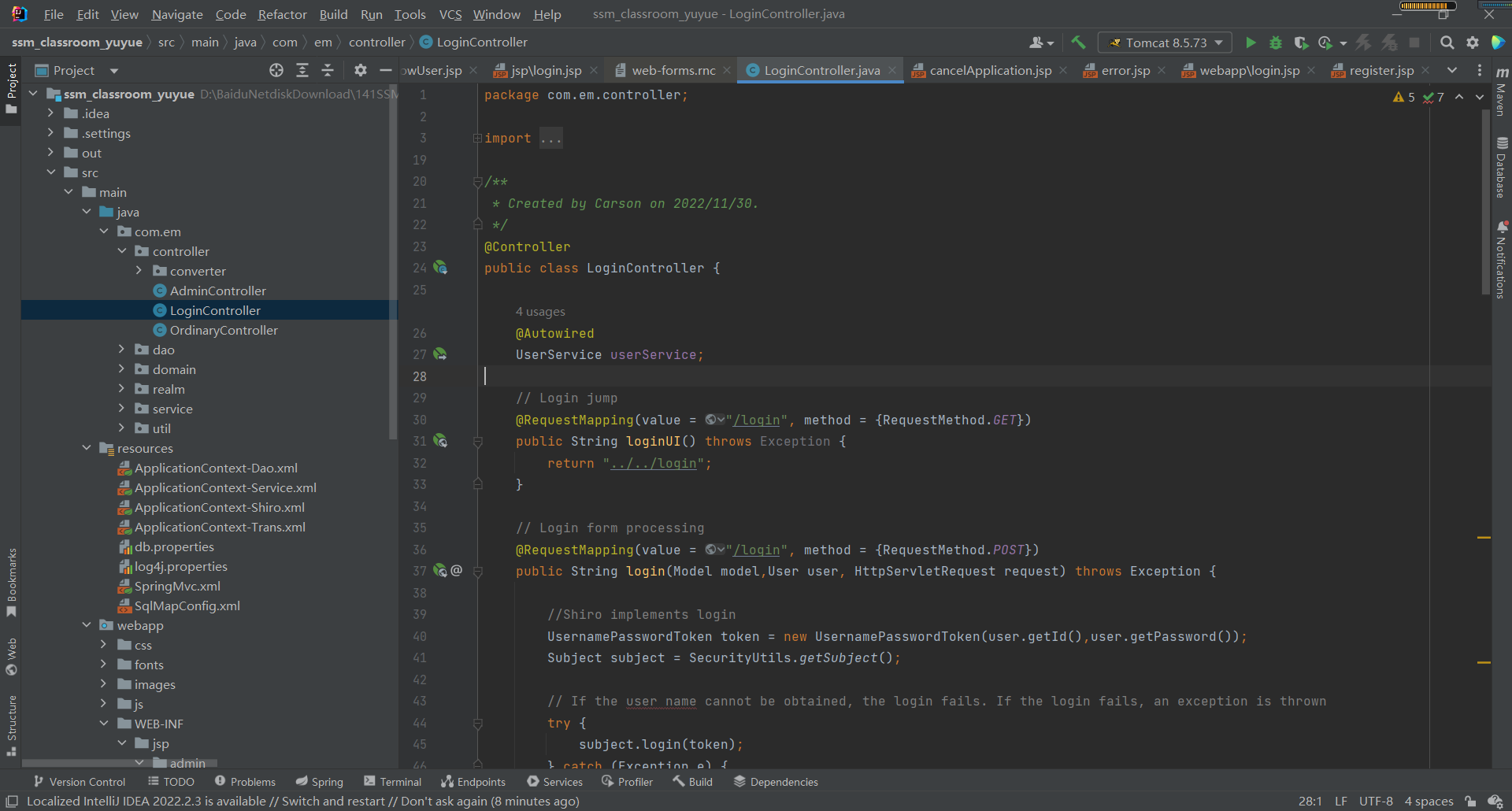


Figure 14 CRMS login back-end link database code

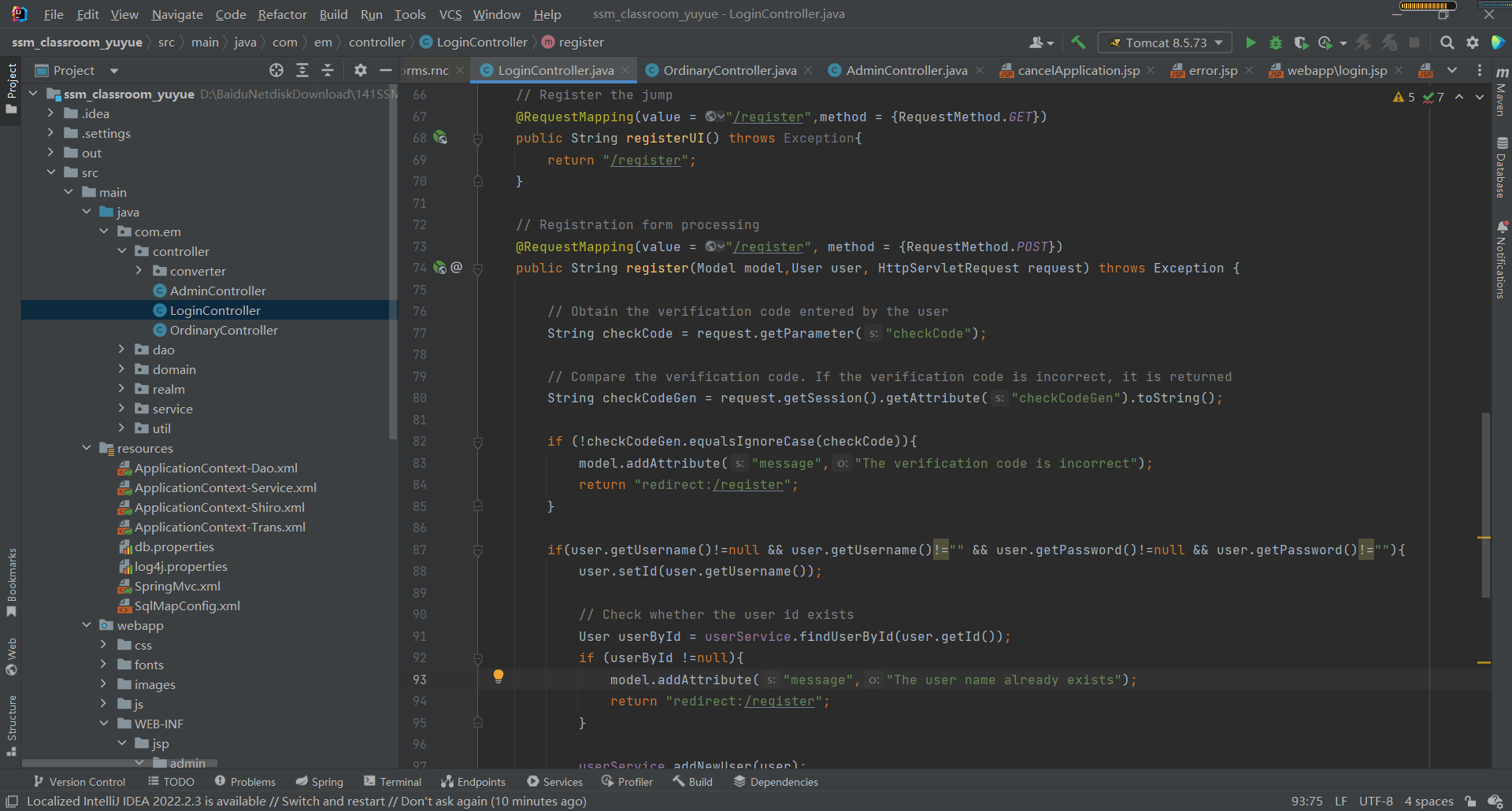


Figure 15 CRMS register back-end link database code

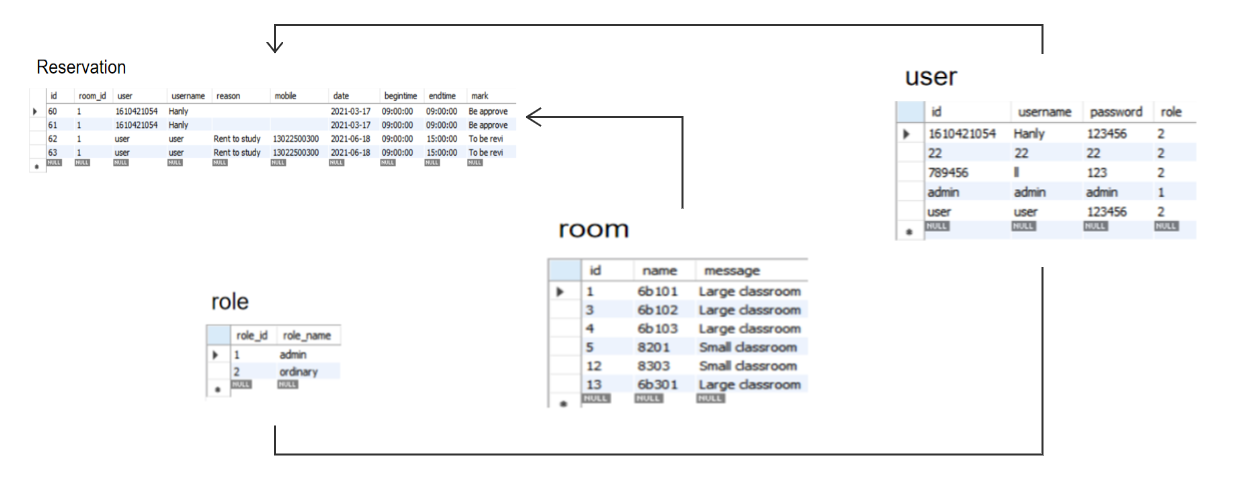


Figure 16 Database ER diagram

The Figure 15 above depicts the Entity-Relationship (ER) diagram for the Database of the Classroom Reservation Management System. The ER diagram illustrates the entities, their relationships, and the attributes that define them.(visual-paradigm, 2023)

The main entity in the diagram is the "Reservation" entity, which represents a user's reservation activity within the system. The "User" entity represents the system users, who are identified by a unique ID number and have associated login credentials.

Other entities in the diagram include "Room," which represents the different rooms available for reservation, and "Role," which defines the user's role within the system. The "Message" entity records communication between system users and administrators.

The diagram also includes various attributes for each entity which define its properties. For example, the "Reservation" entity has attributes such as the reservation date and time, the room being reserved, and the user making the reservation.

Therefore, the ER diagram provides a visual representation of the database schema for the Classroom Reservation Management System, enabling developers to design and implement an efficient and effective database structure.

## Testing and Evaluation

For this classroom reservation management system project, we will be adopting a Test-driven development (TDD) style. (SakshiBhakhra , 2020) The test and evaluation plan will focus on three different types of software testing techniques: unit testing, integration testing, and acceptance testing.

Unit testing will be used to test individual units of code, such as methods and functions, to ensure that they are working correctly. This will be done using a unit testing framework, such as JUnit, and test cases will be written to cover all possible scenarios and edge cases, including testing the registration process for both teachers and students.

Integration testing will be used to test the integration of different modules and components of the system. (such as student modules, teacher modules, back-end modules) This will involve testing how well the different units of code work together and how they handle different inputs and outputs, including testing the integration of the registration process with the rest of the system. This will also be done using a testing framework and test cases will be written to cover all possible scenarios and edge cases.

Acceptance testing will be used to test the system as a whole and ensure that it meets the requirements and expectations of the users. This will involve testing the system with real users, such as teachers and students, and collecting their feedback. The acceptance testing will also include usability testing to ensure that the system is user-friendly and easy to use, including testing the registration process to ensure its usability.

The following is the test flow chart of this program:

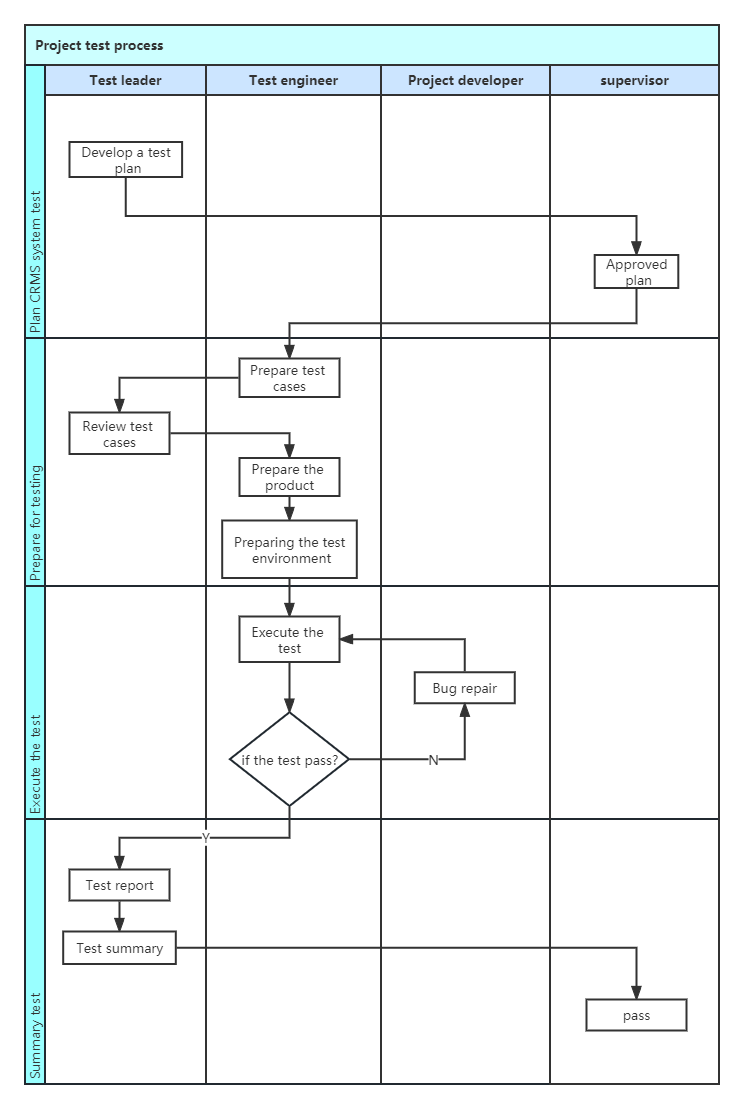


Figure 17 CRMS test flow chart

The Figure 16 above presents the test flow chart for the Classroom Reservation Management System project, which will follow a Test-driven development (TDD) approach. The chart illustrates the roles and responsibilities of the Test Leader, Test Engineer, Project Developer, and Supervisor during the testing process.

The first step in the process is to develop a comprehensive test plan, which will be reviewed and approved before proceeding with the testing. The Test Engineer will then prepare the test cases, which will cover all possible scenarios and edge cases for each type of testing: unit, integration, and acceptance.

Once the test cases are prepared, the Product Developer will create the system based on the requirements specified in the test plan. The Test Engineer will prepare the test environment, including setting up the necessary hardware and software configurations.

Subsequently, the tests will be executed, and any detected bugs or issues will be documented and reported back to the Project Developer for repair. The testing process will continue until all bugs have been resolved, and the system passes all tests successfully.

Finally, the Test Engineer will prepare a test report summarizing the test results, including any issues encountered, and submit it to the Test Leader for review and approval. The Test Summary will provide an overall assessment of the system's quality and readiness for deployment.

Therefore, the CRMS test flow chart provides a structured approach to ensure comprehensive and effective testing of the system, enabling the team to identify and resolve any issues early on in the development cycle.

### Test Principles

The main purpose of system testing is to discover and improve any errors and defects that exist in the system, and to timely correct these errors and defects to improve the accuracy and responsiveness of the system, thereby improving its quality to meet user requires. At the same time, by testing the Classroom Reservation Management Platform, it is possible to find potential errors and defects that have not yet appeared in the current system, so that designers can improve the system in advance and reduce the cost of later maintenance.

System testing is a very important part of the development cycle of the Classroom Reservation Management System, which requires a lot of time, effort, and financial resources. Through strict technical review and testing, as many defects in the system as possible can be detected and eliminated, thereby improving the quality of the system. The principle of system testing is to test early and often, ideally starting during the requirements analysis phase of system development. By testing the system requirements analysis, it ensures that the system design meets the requires of users. During the testing process, legal and illegal inputs as well as various boundary conditions should also be considered.

### Test Purpose

System testing is an indispensable task after the completion of system development. System testing involves using both manual and automated software to run program code, and then comparing the results to expected outcomes to identify any differences or errors. The goal of system testing is to uncover as many system defects as possible, correct them, and then re-validate that they have been corrected; this repetition involves every functional module of the system.

Through testing the system, it is verified whether the system meets all requirements, and checks whether all business processes meet user requires. At the same time, in system testing, the usability and stability of the system must be tested to ensure that the system can operate normally in a production environment.

### Test Methods

System function testing principally adopts the black box testing method. In black box testing, there is no require to understand the specific workflow of the system's internal operations. Instead, testers take the perspective of the user and test whether each functional module of the system works according to user requirements by writing good test cases. Black box testing also has many advantages: first, for systems with a large amount of code, the testing efficiency of black box testing is higher than that of white box testing because testers do not require to know many complex things and can work independently of programmers without any correlation. Anyone who can operate the system can perform black box testing; secondly, testers in this process can easily understand the test operations from the user's perspective because they can easily see whether the system functions are correctly implemented, and if there are errors, they can also be easily discovered, which assists to solve problems.

### Test Plan

4.2.4.1 Test Environment and Conditions

Through software testing, a higher quality system can be developed that is more suitable for use in real life. The following are the environment requirements for software testing:

Database: MySQL

Web Server: Apache Tomcat 9.0

Operating Platform: Web Browser

4.2.4.2 Test Statistics and Analysis

In order to facilitate user-friendly interface and complete functions of the Classroom Reservation Management Platform based on SSM, after the development is completed, the system will be tested as much as possible.

Test Results, as demonstrated in Table below:

| Test ID | Test Function Module | Test Case | Input Data | Expected Result | Actual result | Test state |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | System Login | Test whether users can complete login. The system has multiple roles of users, each with their own permissions, and each user must select their role type when logging in. This module requires to test whether the system login enters the corresponding page according to the corresponding role. | Users require to enter the account name and corresponding password obtained during registration. | The system login function is fully implemented. | The system login function is fully implemented | The result was as expected |
| 2 | Database Data Reading | Used to submit information viewing requests to test whether the information can be read completely from the database and displayed on the corresponding interface. | Add information to the database. | The information can be completely displayed on the interface according to the user's request and can be modified. | The information can be completely displayed on the interface according to the user's request and can be modified. | The result was as expected |
| 3 | Classroom Information Management | Add Classroom Information: This test case is used to verify whether administrators can add new classroom information to the system. The expected result is that the system prompts that the addition is successful and displays the new classroom information. | Administrators add a new classroom information. | The system prompts that the addition is successful and displays the classroom information. | The system prompts that the addition is successful and displays the classroom information. | The result was as expected |
| 4 | Classroom Information Management | Modify Classroom Information: This test case is used to verify whether administrators can modify existing classroom information, such as available time periods. The expected result is that the system prompts that the modification is successful and updates the classroom information accordingly. | Administrators modify a piece of classroom information, such as available time period. | The system prompts that the modification is successful and updates the classroom information. | The system prompts that the modification is successful and updates the classroom information. | The result was as expected |
| 5 | Classroom Information Management | Modify Classroom Information: This test case is used to verify whether administrators can modify existing classroom information, such as available time periods. The expected result is that the system prompts that the modification is successful and updates the classroom information accordingly. | Administrators delete a piece of classroom information. | The system prompts that the deletion is successful and deletes the classroom information. | The system prompts that the deletion is successful and deletes the classroom information  . | The result was as expected |
| 6 | Reservation Review | verify whether administrators can review reservation applications submitted by users. The expected result is that the system prompts that the review is successful and updates the status of the reservation application. | Administrators view and review reservation applications. | The system prompts that the review is successful and updates the reservation application status. | The system prompts that the review is successful and updates the reservation application status. |  |
| 7 | Reservation Record Query | verify whether administrators or users can query reservation records. The expected result is that the system displays reservation records that meet the criteria specified by the user. | Administrators or users query reservation records. | Display the reservation records that meet the conditions. | Display the reservation records that meet the conditions. | The result was as expected |
| 8 | User Registration | verify whether users can successfully register for an account. Users require to enter registration information, such as their name and contact details. The expected result is that the system prompts that the registration is successful and navigates the user to the login page. | Users enter registration information for registration. | The system prompts that registration is successful and navigates to the login page. | The system prompts that registration is successful and navigates to the login page. | The result was as expected |
| 9 | Classroom Information Query | verify whether users can query available classroom information. The expected result is that the system displays the classroom information that meets the criteria specified by the user. | Users query available classroom information. | Display the classroom information that meets the conditions. | Display the classroom information that meets the conditions | The result was as expected |
| 10 | Reservation Record Query | verify whether users can query their personal reservation records. The expected result is that the system displays the reservation records that meet the criteria specified by the user. | Users query personal reservation records. | Display personal reservation records. | Display personal reservation records. | The result was as expected |
| 11 | Classroom Reservation | verify whether users can reserve a classroom. Users require to select a classroom and time period for the reservation. The expected result is that the system prompts that the reservation is successful and assigns the corresponding classroom to the user. | Users select classroom and time period for reservation. | The system prompts that the reservation is successful and assigns the corresponding classroom to the user. | The system prompts that the reservation is successful and assigns the corresponding classroom to the user. | The result was as expected |
| 12 | Personal Information Modification | verify whether users can modify their personal information, such as their password. The expected result is that the system prompts that the modification is successful and updates the user's information accordingly. | Users modify personal information, such as password. | The system prompts that the modification is successful and updates the user information. | The system prompts that the modification is successful and updates the user information. | The result was as expected |

Table 3 Test Results

This table 3 above contains a list of test cases for various system functions such as login, database data reading, classroom information management, reservation review, reservation record query, user registration, classroom information query, classroom reservation, and personal information modification. Each test case includes the test function module, input data, expected result, and actual result. The actual results of each test were as expected, indicating that the implemented functions are working correctly.

# **Professional Issues**

## **Project Management**

### **Activities**

To achieve the project's overall aim and objectives, several key tasks have been identified. The following is a summary of the completed and uncompleted tasks for each objective:

1. completes the background review of the existing classroom reservation management system.(Deliverables: survey website and results, demand analysis conclusions, analysis of existing models)

(1.1) System search for similar software (deadline: 2022.11.7)(completed)

(1.2) Create Feature Comparison Table (deadline: 2022.11.9)(completed)

(1.3) Complete literature search (deadline: 2022.11.11)(completed)

(1.4) Literature review (deadline: 2022.11.13)(completed)

(1.5) Conduct user survey (deadline: 2022.11.15)(completed)

1. divides clear functions according to roles to facilitate data management.(Deliverables: role function results, database table, ER diagram, function structure diagram)

(2.1) Division of roles(deadline: 2022.11.22)(completed)

(2.2) Functional division of the first role(deadline: 2022.11.27)(completed)

(2.3) Functional division of the second role(deadline: 2022.12.2)(completed)

(2.4) Feasibility analysis of divided functions(deadline: 2022.12.7)(completed)

(2.5) Organize role function (deadline: 2022.12.12)(completed)

1. has developed the teacher administrator management function to display the user's usage in real time.(Deliverables: background login code, management function code)

(3.1) Design and develop administrator login function (deadline: 2022.12.17)(completed)

(3.2) Design and develop classroom information management function (deadline: 2022).12.25)(completed)

(3.3) Design and develop registered user functions (deadline: 2022.12.30)(completed)

(3.4) Design and develop appointment function (deadline: 2023.1.7)(completed)

(3.5) Design and develop appointment audit function (deadline: 2023.1.15)(completed)

1. develops the function of students' classroom reservation, which can dynamically display relevant data of the system.(Deliverables: homepage login code, reservation function code)

(4.1) Design and develop user login function (deadline: 2023.3.7)(completed)

(4.2) Design and develop classroom information query function (deadline: 2023.3.15)(completed)

(4.3) Design and develop reservation record query function (deadline: 2023.3.30)(completed)

(4.4) Design and develop the function of reserved classroom (deadline: 2023.4.15)(completed)

(4.5) Design and develop personal information modification function (deadline: 2023.4.30)(completed)

1. demonstrates the works to mixed audiences.

(5.1) Materials and processes to be displayed (deadline: 2023.5.17)(uncompleted)

(5.2) Presentation process to mixed audiences (deadline: 2023.6.4)(uncompleted)

.

### **Schedule**

A Gantt chart has been developed to visualize the project schedule and track the progress of tasks. The chart illustrates the timeline of the project, the start and end dates of each task, and the dependencies between tasks. It also highlights the completed and uncompleted tasks, as well as any delays or changes to the original plan.

The Gantt chart for the Classroom Reservation Management System project is demonstrated in Figure 17 below:

It demonstrates that the majority of the research and design tasks have been completed, including the background review, system design, and database design. The implementation tasks, such as the development of the teacher administrator management function and the student reservation function, are currently in progress.

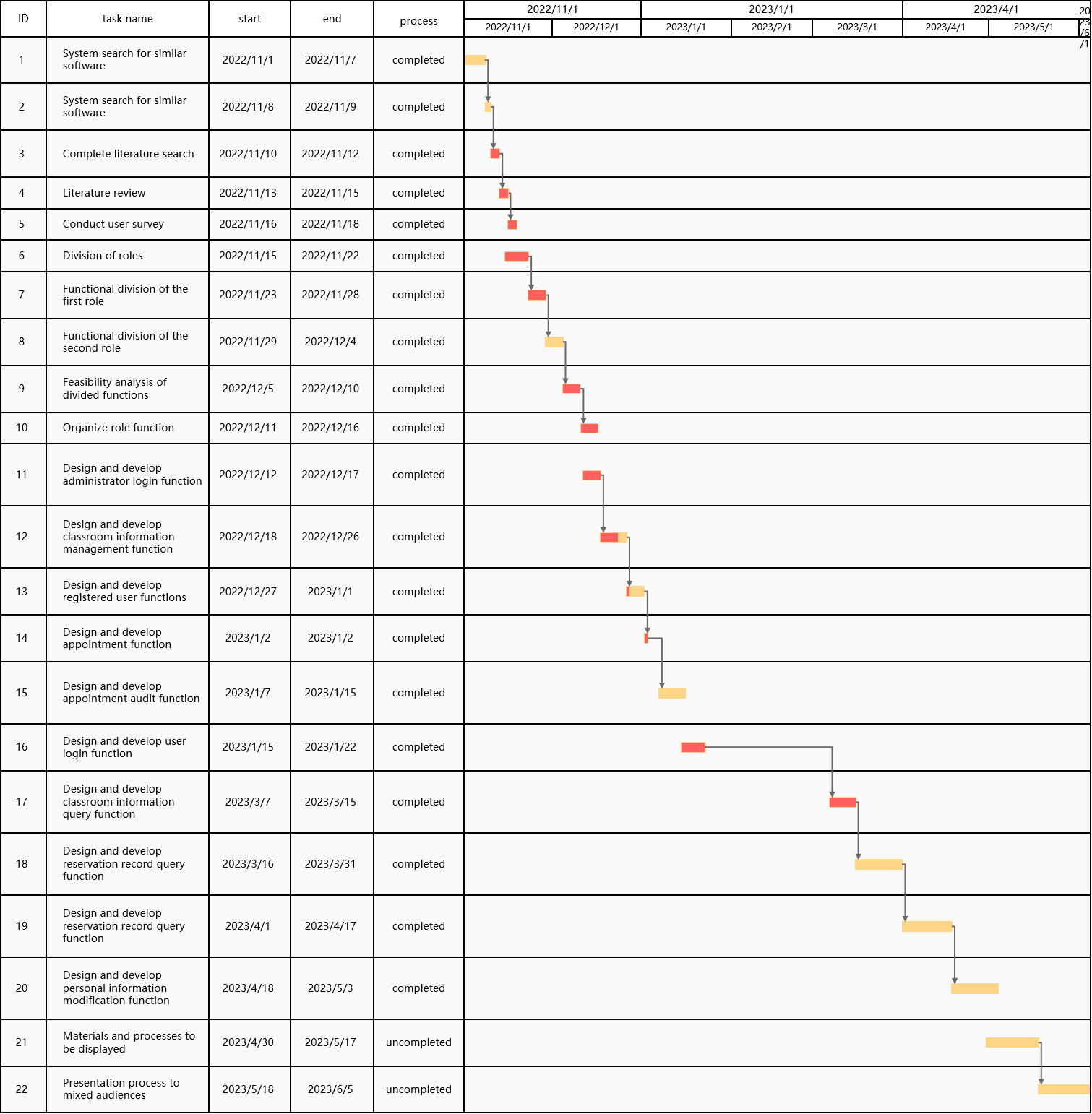


Figure 18 Gantt chart of the CRMS project

The Figure 17 above displays the Gantt chart for the Classroom Reservation Management System project. The chart serves as a visual representation of the project's timeline, depicting the start and end dates of each task, its interdependencies, and the progress made towards completion. The chart distinguishes between completed and uncompleted tasks, highlighting any changes or delays to the original plan. As indicated by the chart, most of the research and design aspects, including background review, system design, and database design, have been completed. Implementation tasks such as developing teacher administrator management and student reservation functions are currently in progress. The Gantt chart enables the project team to track the project's schedule, monitor task completion, and make necessary adjustments to ensure successful project delivery.

### **Project Data Management**

The project data management for this project is principally done through the use of github repository. This is an online storage platform that allows for the storage and sharing of project related documents such as logs, reports, and literature. The platform is secure and can only be accessed by the author and the supervisor. This ensures that the data is kept confidential and is only accessible to those who require it.

In terms of organization, all project-related documents are stored in a dedicated folder on Baidu drive. This includes project logs, reports, and literature, among others. These documents are organized in a logical and easy to navigate manner, making it easy to find the relevant information when required. The platform also allows for easy sharing of documents with the supervisor, which is essential in facilitating the review and feedback process. Therefore, Baidu drive serves as an efficient and secure tool for managing the project data.

### **Project Deliverables**

In this section, the project deliverables that have been submitted or are yet to be submitted for assessment are outlined. These include the project proposal, progress report, final report, and any relevant project code or software developed throughout the course of the project. Additionally, any additional resources or documents that have been created as part of the project, such as documentation for user manual, system architecture, are also included in the list of deliverables. The goal of this section is to provide a clear and concise overview of all the materials that have been created and submitted as part of the Classroom Reservation Management System project. It's important to mention the accessibility to those deliverables, such as the project code and software is only editable by the supervisor and the author.

## **Risk Analysis**

Risk analysis is an essential component of any project, and the Classroom Reservation Management System project is no exception.In this section, an analysis of the risks associated with the Classroom Reservation Management System project is presented.Additionally, several risks were identified, including legal, social, ethical, and environmental risks. The identification of these risks was based on a thorough analysis of the project requirements, project scope, and project context. The table below demonstrates the potential risks identified, their causes, the severity and likelihood of the risks, and the mitigation strategies implemented to address them.

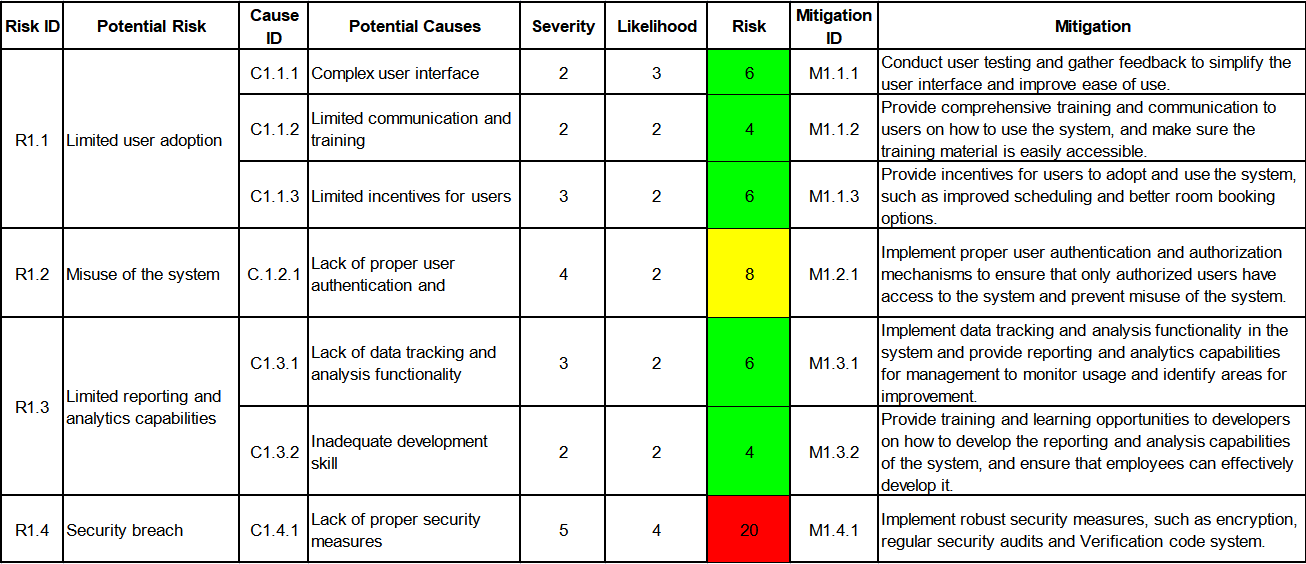


Figure 19 CRMS software development risks

As seen in the figure above, all identified risks were mitigated successfully with the actions taken, and at this point, the project is on track to meet its deadlines with all the core features having been completed and the user login and classroom reservation functionalities being fully operational and also having implemented a robust database system.

The various types of risks associated with classroom reservation management system projects are discussed below:

Legal Risks:

One of the potential legal risks identified in the project was related to data privacy and compliance with applicable laws and regulations. As the system would store personal information of users, it was important to ensure compliance with data protection laws such as the Personal Information Security Specification in China and the General Data Protection Regulation (GDPR) in the UK. (wikipedia , 2023) To mitigate this risk, robust security measures were implemented, such as encryption and regular security audits to protect user data and ensure compliance with relevant laws and regulations. Legal experts from both China and the UK were consulted to ensure that the system complied with local laws and regulations related to data protection, intellectual property, contract law, and liability.

Social Risks:

Another potential risk was related to the social dynamics of the educational institution where the system would be used. For instance, some faculty members may not feel comfortable with the centralized reservation system, which may lead to a reduction in their autonomy. To mitigate this risk, stakeholder analysis was conducted, and faculty members were engaged to understand their concerns and preferences. Training and communication were provided to stakeholders to ensure that they understood the benefits of the system and how it could support their work.

Ethical Risks:

Another potential risk was related to the ethical considerations surrounding the use of technology in educational settings. For example, there may have been concerns about the impact of the system on student privacy or the potential bias in the way that room reservations were allocated. To address these risks, guidelines and policies for the ethical use of the system were developed, and features were implemented such as anonymization of user data to protect student privacy. Regular audits of the system were conducted to identify and address any potential biases in the allocation of room reservations.

Environmental Risks:

Additionally, there were potential environmental risks associated with the Classroom Reservation Management System, particularly if it relied on energy-intensive server infrastructure. To mitigate this risk, energy-efficient infrastructure was implemented, and the hosting provider was selected to ensure that the system was hosted on servers that met environmental standards and were powered by renewable energy sources.

Finally, a bug was encountered when running the classroom reservation function, as demonstrated below:



Figure 20 bug log

During the development of a JSP file, there were two issues encountered which resulted in errors. The first issue was related to the use of Chinese characters in an XML file. As XML files are encoded in UTF-8 by default, it is important to ensure that the editor or IDE being used supports and is set to use UTF-8 encoding. Failure to do so can result in parsing errors or other issues. In this instance, the use of Chinese characters led to an error that prevented successful execution of the file.

The second issue was related to the use of the Gson library in a JSP file. The error message indicated that the Gson class could not be resolved, which can be caused by missing dependencies or improper referencing of the Gson library. To address this issue, the necessary Gson dependency was added to the project, or alternatively, the relevant jar file was manually added to the WEB-INF/lib directory.

After these changes were made and the encoding issue was addressed by avoiding the use of Chinese characters in variable names and identifiers, the JSP file was successfully executed without errors.These steps demonstrate the importance of risk analysis and mitigation strategies in successful software development projects.

The error code before modification is demonstrated in the following figure:

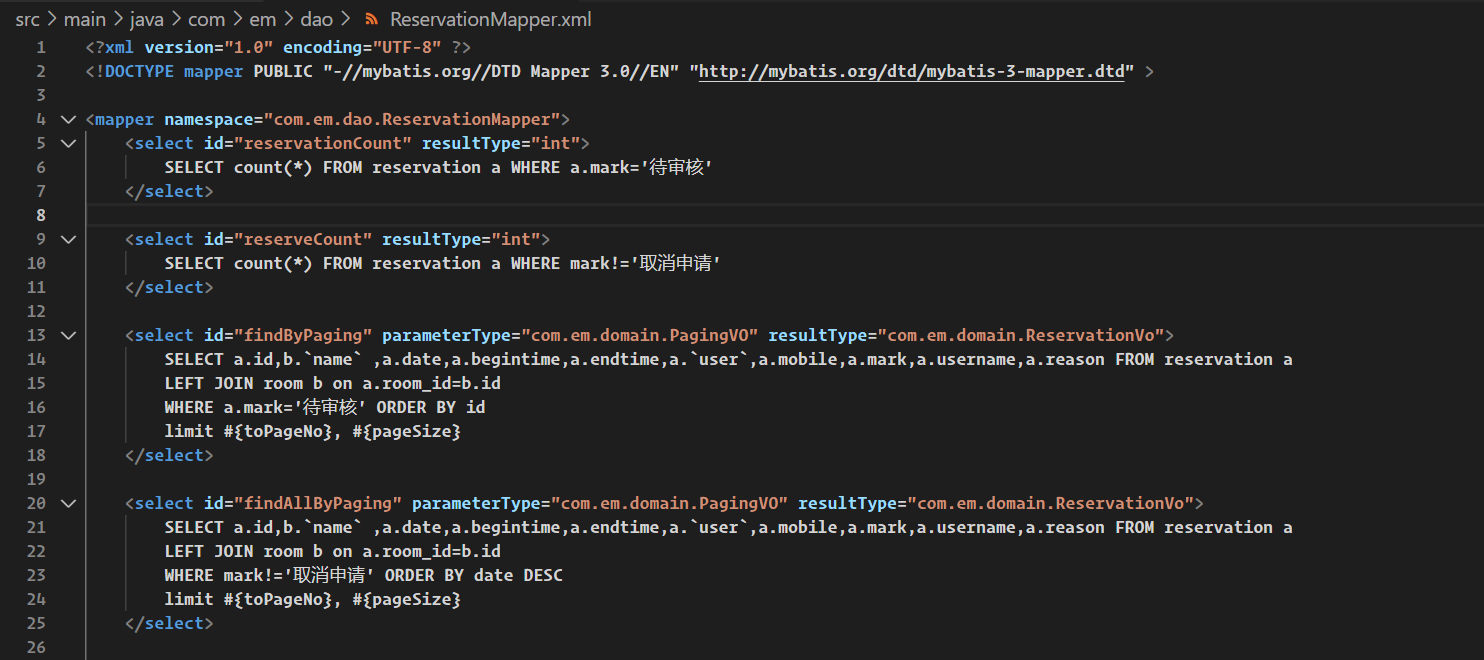


Figure 21 The error code part1

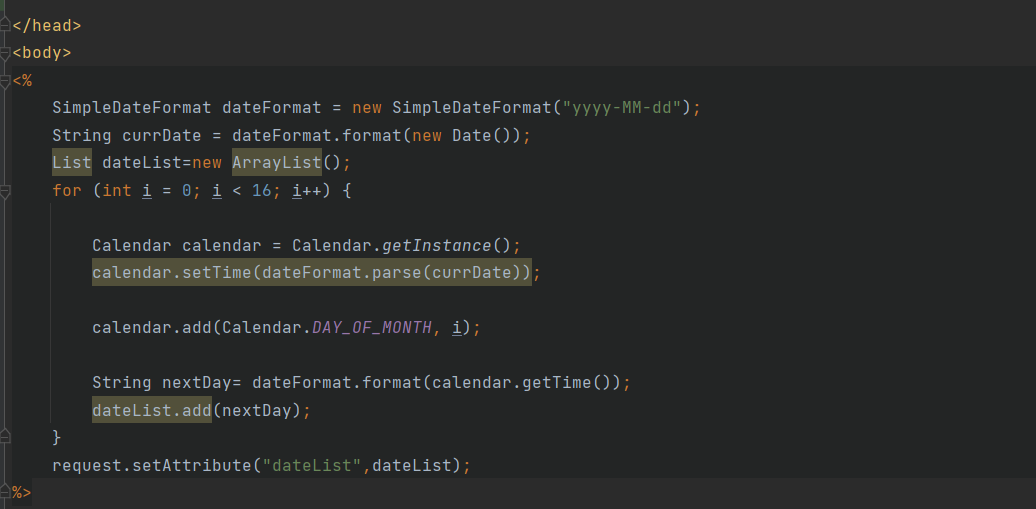


Figure 22 The modified code part2

The modified code is demonstrated in the following figure:

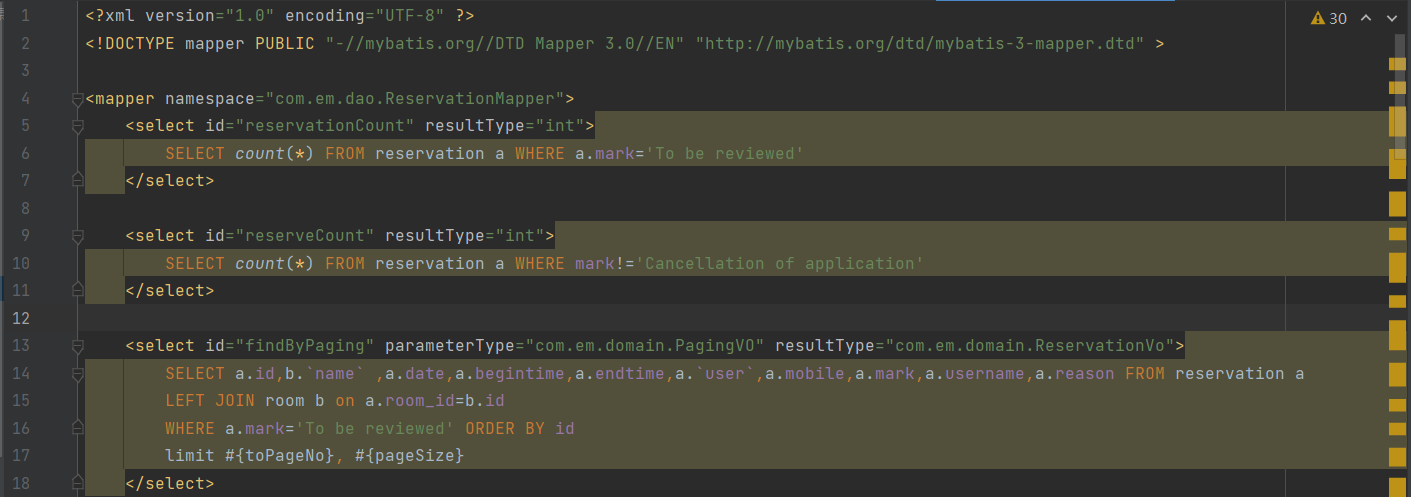


Figure 23 The modified code part1



Figure 24 The modified code part2

After successful modification, the classroom reservation function runs successfully as demonstrated in the following figure:

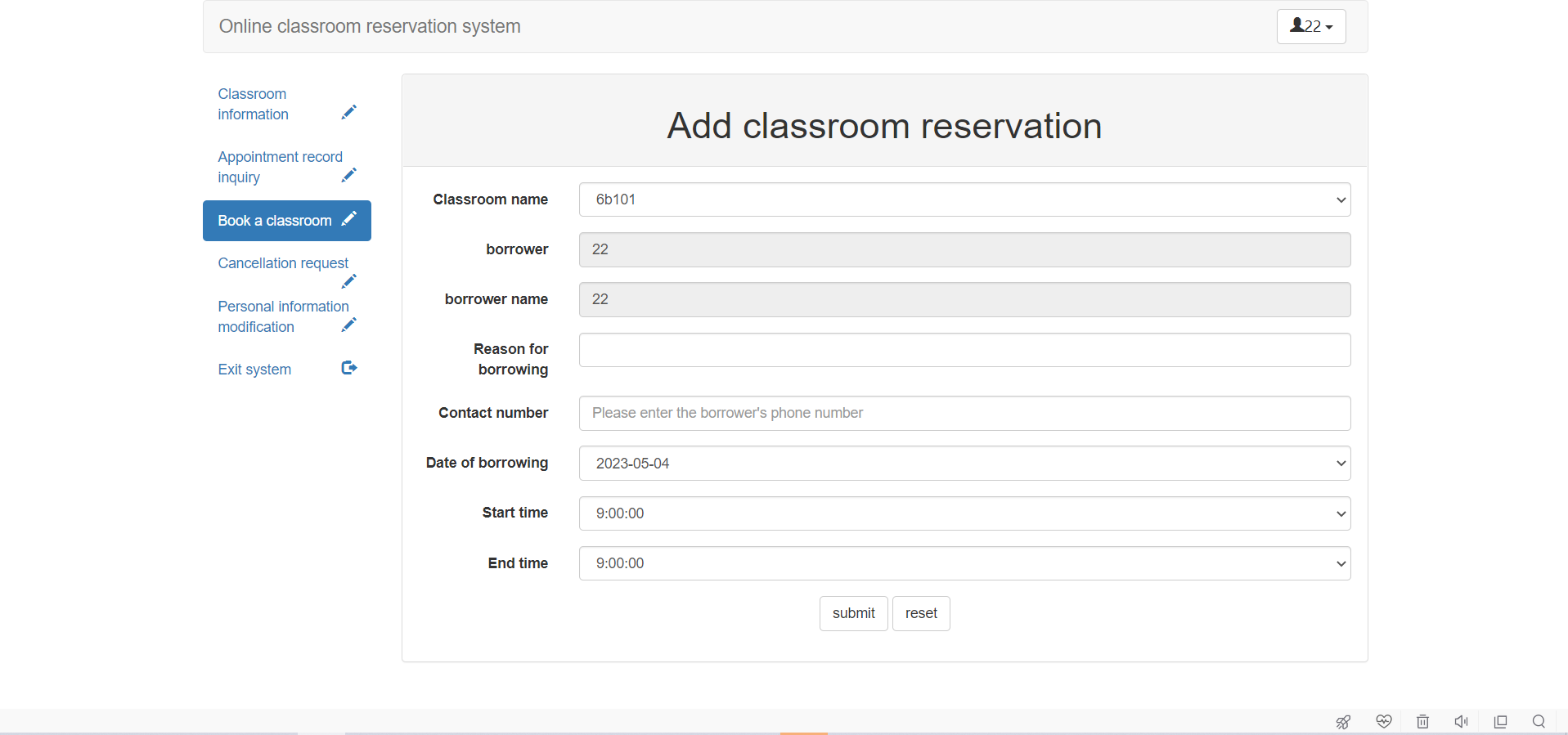


Figure 25 classroom reservation page

Therefore, the project team identified potential risks associated with the Classroom Reservation Management System project and took a proactive approach to mitigate these risks. By implementing robust mitigation strategies and engaging with stakeholders, and consulting with legal experts in both China and the UK, the project team successfully delivered a system that met the requires of users and adhered to legal, social, ethical, and environmental best practices. In addition to the risks identified in the previous section of this paper, there are several other risks that should be considered for the Classroom Reservation Management System project. These include legal, social, ethical, and environmental risks.Additionally, the developer encountered a bug when running the classroom reservation function, which was successfully addressed by modifying the code. Therefore, by taking a proactive approach to risk management, the project team successfully delivered a system that met user requires and adhered to best practices.

## **Professional Issues**

In the background of the Classroom Reservation Management System project, several critical legal, social, ethical, and environmental issues arise that must be considered.

Legal Issues:

Data protection and privacy is a primary legal issue that stems from the system's processing and storage of personal information, such as student names, ID numbers, and reservation data. Compliance with relevant data protection laws in both China and the United Kingdom is necessary, such as the General Data Protection Regulation (GDPR) in the UK and China's Cybersecurity Law. The system should also have a coherent privacy policy that outlines how data is used, stored, and protected.

Social Issues:

Inclusivity and accessibility are vital social issues for the Classroom Reservation Management System. The system must cater to the requires of all students, including those with physical or cognitive abilities. Installation of assistive technology and an intuitive user interface is crucial to ensure accessibility to all students. Transparency, fairness, and impartiality in the reservation process are also essential social considerations.

Ethical Issues:

The Classroom Reservation Management System must adhere to ethical principles in its development and use. This includes ensuring the provision of fair and unbiased access to reservation opportunities, which is essential to the learning experience of all students. Additionally, promoting transparency and accountability in the system's use is critical in upholding ethical values.

Environmental Issues:

The Classroom Reservation Management System's energy consumption and carbon footprint are potential environmental issues that should be considered during development. To minimize its impact on the environment, the system should prioritize low power consumption and energy-saving features.

Finally, adherence to relevant professional codes of conduct set by organizations like the British Computer Society (BCS) and the Association for Computing Machinery (ACM) is crucial throughout the development of the Classroom Reservation Management System.(wikipedia, 2023)These codes outline professional standards and ethical principles that guide the development and use of technology in both China and the UK.

# **Conclusion**

6.1 Summary of Achievements

In this project, we designed and implemented a classroom reservation management system using the Spring Application Framework and the Sprint model. The system features a user-friendly interface that allows teacher administrators and students to quickly and easily reserve or manage classrooms. It includes functions such as teacher administrator management, student reservation, and a reporting function for analyzing data on classroom usage and occupancy.

The objectives of the system were met, including conducting a background review of existing systems, dividing clear functions according to roles, developing teacher administrator management and student reservation functions, and demonstrating the system to mixed audiences.

The system was developed using Java, JSP, SSM, and B/S architecture and utilized a MySQL database. The significance of this project lies in its ability to improve the current classroom reservation process by providing a more efficient and user-friendly system for both teacher administrators and students.

6.2 Potential Future Work

There are several potential areas for future work on this classroom reservation management system. Some of these areas include:

1.Integration with other existing systems and technologies: The system could be further developed to integrate with other university systems, such as student information systems, to streamline the reservation process even further.

2.Mobile app development: Developing a mobile app for the system could make it even more convenient for users to manage their classroom reservations.

3.Data analytics and machine learning: Using data analytics and machine learning techniques could assist to further optimize the reservation process and improve the system's performance.

4.Additional features: The system could be enhanced with additional features such as a messaging system to allow for better communication between teacher administrators and students.

In conclusion, the potential for further development of the classroom reservation management system is vast and could greatly benefit universities and their students and teacher administrators.

# **References**

Resmark (2020) Why a reservation management system is important, Resmark. Available at: https://www.resmarksystems.com/blog/why-a-reservation-management-system-is-important (Accessed: November 9, 2022).

Creatrix (2022) Online classroom management software system, Creatrix Campus. Available at: https://www.creatrixcampus.com/classroom-management-system (Accessed: November 9, 2022).

Skedda (2022) Classroom scheduling software - venue management, Skedda. Available at: https://www.skedda.com/home/classroom-scheduling-software (Accessed: November 9, 2022).

programmerall (2022) Part I. spring boot documentation, Spring Boot Chinese document (Official document translation based on 1.5.2.RELEASE) - Programmer All. Available at: https://www.programmerall.com/article/8331715216/ (Accessed: November 13, 2022).

Merriam (2022) System definition & meaning, Merriam-Webster. Merriam-Webster. Available at: (Accessed: November 9, 2022).

Adriano, R. (2021) Classroom reservation system, Creately. Available at: https://creately.com/diagram/example/hqrksio41/classroom-reservation-system (Accessed: November 14, 2022).

Bagyatech (2020) Advantages of waterfall model and its disadvantages, Bagyatech. Available at: https://bagyatech.com/advantage-and-disadvantage-of-waterfall-model/#:~:text=The%20notable%20advantages%20%26%20disadvantages%20of%20t (Accessed: November 14, 2022).

Spring (2023) Spring Framework Overview. Available at: https://docs.spring.io/spring-framework/docs/current/reference/html/overview.html (Accessed: May 5, 2023).

wikipedia (2023) Model–view–controller, Wikipedia. Wikimedia Foundation. Available at: https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller (Accessed: May 5, 2023).

wikipedia (2022) Inversion of control, Wikipedia. Wikimedia Foundation. Available at: https://en.wikipedia.org/wiki/Inversion\_of\_control (Accessed: May 5, 2023).

visual-paradigm(2023) What is entity relationship diagram (ERD)? Available at: https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/;WWWSESSIONID=7CB5F2E286A9373FB40744196BC9614F.www1 (Accessed: May 5, 2023).

SakshiBhakhra (2020) Test Driven Development (TDD), GeeksforGeeks. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/test-driven-development-tdd/ (Accessed: May 5, 2023).

Wikipedia (2023) General Data Protection Regulation, Wikipedia. Wikimedia Foundation. Available at: https://en.wikipedia.org/wiki/General\_Data\_Protection\_Regulation (Accessed: May 5, 2023).

Wikipedia (2023) British Computer Society, Wikipedia. Wikimedia Foundation. Available at: https://en.wikipedia.org/wiki/British\_Computer\_Society (Accessed: May 5, 2023).

Wikipedia (2023) Association for Computing Machinery, Wikipedia. Wikimedia Foundation. Available at: https://en.wikipedia.org/wiki/Association\_for\_Computing\_Machinery (Accessed: May 5, 2023).

Rehkopf, M. (2023) Scrum sprints: Everything you require to know, Atlassian. Available at: https://www.atlassian.com/agile/scrum/sprints (Accessed: May 5, 2023).

geeksforgeeks(2023) Java tutorial: Learn java programming, GeeksforGeeks. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/java/ (Accessed: May 5, 2023).

Wikipedia (2023) Eclipse, Wikipedia. Wikimedia Foundation. Available at: https://en.wikipedia.org/wiki/Eclipse (Accessed: May 5, 2023).

GeeksforGeeks(2023) Introduction to JSP, GeeksforGeeks. GeeksforGeeks. Available at: https://www.geeksforgeeks.org/introduction-to-jsp/ (Accessed: May 5, 2023).

Pejsar, R.J. (1975) The systems manager: A study in conflict, Amazon. Amazon. Available at: https://docs.aws.amazon.com/systems-manager/latest/userguide/what-is-systems-manager.html (Accessed: May 5, 2023).

Begin, C. (2023) Mybatis. Available at: https://mybatis.org/mybatis-3/ (Accessed: May 5, 2023).

w3schools(2023) HTML introduction, Introduction to HTML. Available at: https://www.w3schools.com/html/html\_intro.asp (Accessed: May 5, 2023).

MozDevNet (2023) What is CSS? - learn web development: MDN, Learn web development | MDN. Available at: https://developer.mozilla.org/en-US/docs/Learn/CSS/First\_steps/What\_is\_CSS (Accessed: May 5, 2023).

w3school (2023) What is JavaScript?, What is JavaScript. Available at: https://www.w3schools.com/whatis/whatis\_js.asp (Accessed: May 5, 2023).