SOFTWARE ENGINEERING PROJECT

TIME TABLE MANAGEMENT SYSTEM

BY TEAM: TD05

PRASANTH.S - 220701201

PRRANAV.P - 220701207

RAHUL KISHORE.D - 220701213

RANJITH KUMAR.R - 220701219

RISHIKESH.T - 220701226

RUDRAPRIYAN.N - 220701232

Problem Statement

Educational institutions face significant challenges in managing and scheduling classes, exams, and other academic activities. Creating an efficient timetable that accommodates the needs of students, teachers, and administrative staff, while also utilizing available resources effectively, is a complex task. Manual timetable creation is often time-consuming, prone to errors, and inflexible to changes. Therefore, there is a need for an automated Timetable Management System to streamline this process.

Objectives

The primary objective of the Timetable Management System is to automate the creation, management, and distribution of timetables for educational institutions. This system aims to:

Reduce manual effort: Minimize the time and effort required to create and update timetables.

Improve accuracy: Eliminate human errors associated with manual scheduling.

Enhance flexibility: Allow easy modifications to timetables in response to changing needs.

Optimize resource utilization: Ensure efficient use of classrooms, laboratories, and other resources.

Facilitate accessibility: Provide easy access to timetables for students, teachers, and staff through a centralized system.

Proposed Solution

The proposed solution is to develop a comprehensive, automated Timetable Management System (TMS) that leverages data integration, advanced algorithms, and user-friendly interfaces to create, manage, and distribute academic timetables efficiently. This system will be designed to address the complexities of scheduling, optimize resource utilization, and enhance communication among stakeholders.

Key Components of the Solution

Automated Timetable Generation

Algorithm Design: Implementation of scheduling algorithms that consider all constraints (e.g., teacher availability, classroom capacity, course prerequisites) to generate conflict-free timetables.

Optimization: Use of optimization techniques to ensure efficient use of resources and balanced teacher workloads.

Centralized Database

Data Storage: A robust and scalable database to store all relevant data, including class details, teacher schedules, resource availability, and student information.

Data Integration: Seamless integration with existing school management systems (e.g., student information systems, HR systems) to import and synchronize data.

User Management System

Role-Based Access Control: Different levels of access for admins, teachers, and students to ensure data security and privacy.

Personalized Dashboards: Customizable dashboards for each user role, providing easy access to relevant timetable information and updates.

User Interface

Web Application: Responsive web application for easy access to timetables.

Calendar View: Intuitive calendar view for visualizing schedules.

Notification System: Real-time notifications for timetable changes and updates.

Conflict Resolution and Flexibility

Automated Conflict Detection: Real-time detection of scheduling conflicts and suggestions for resolution.

Manual Adjustments: Allow admins to manually adjust timetable when necessary, with system suggestions to maintain consistency and avoid errors.

Dynamic Updates: Capability to handle last-minute changes, such as teacher absences or classroom unavailability, and update timetables accordingly.

Security and Privacy

Data Encryption: Implement data encryption to protect sensitive information.

Access Control: Enforce strict access controls to prevent unauthorized access to the system.

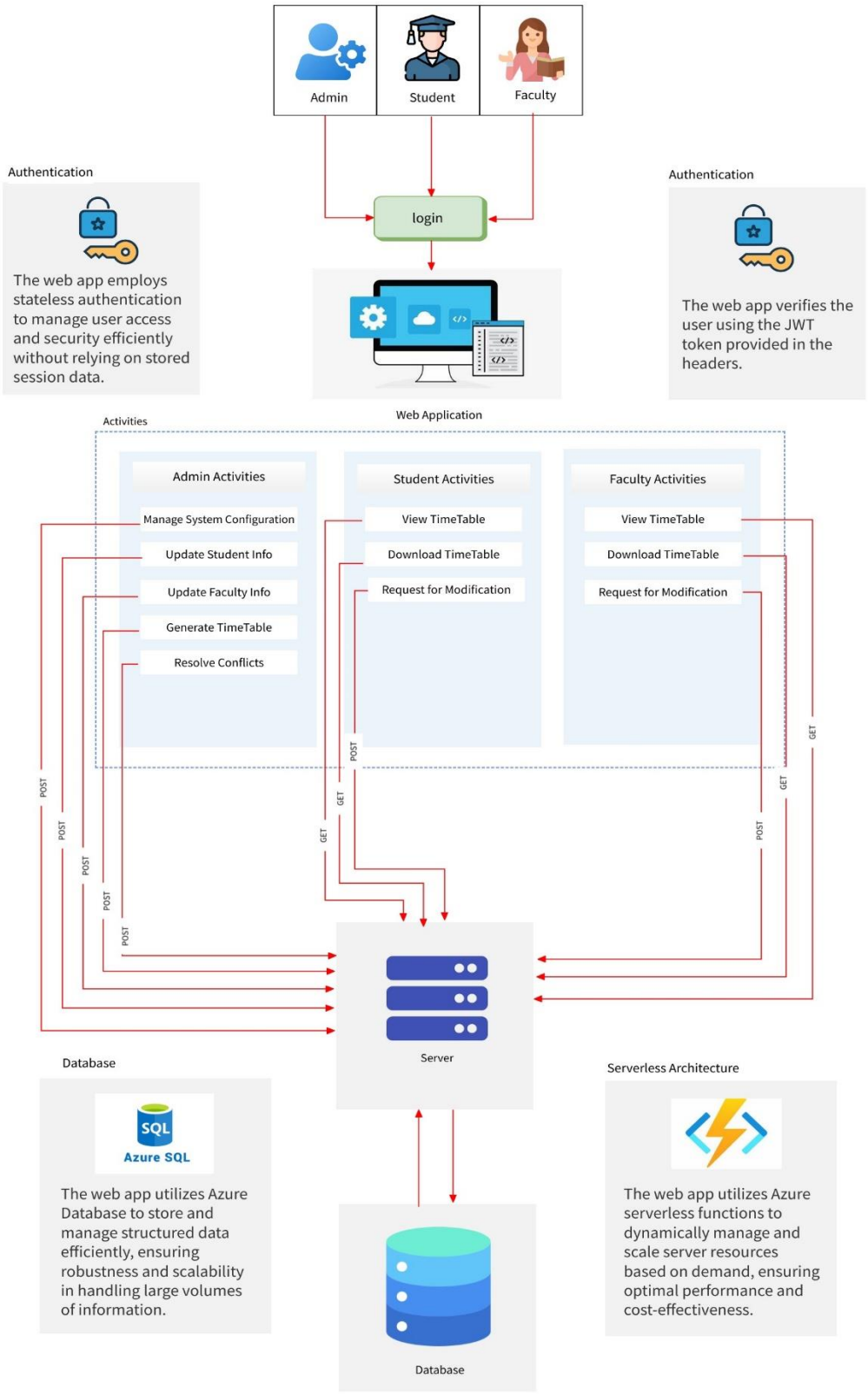
Backup and Recovery: Regular backups and a disaster recovery plan to ensure data integrity and availability.

Business Architecture

Business architecture in software engineering is the structural framework that aligns technological solutions with business objectives. It provides a blueprint for integrating business processes, information flows, and organizational structures within software systems. This discipline involves mapping business requirements to software solutions to ensure that technology investments support strategic goals effectively. Key components include analyzing and optimizing business processes, defining information architecture for data storage and flow, designing systems for enterprise integration, understanding organizational structure for effective collaboration, and determining technology infrastructure requirements. By focusing on these areas, organizations can drive innovation, improve operational efficiency, and gain competitive advantages through their software solutions.

Business Architecture Diagram

TimeTable Management System Business Architecture



User Stories

1. As an Admin, I want to create, edit, and delete teacher profiles so that I can manage the teaching staff's information accurately.

    • Estimate: 3 story points

2. As an Admin, I want to input and update classroom details (capacity, equipment, etc.) so that I can ensure proper allocation of rooms for classes.

    • Estimate: 2 story points

3. As a Teacher, I want to view my weekly timetable so that I can know my schedule and prepare for my classes accordingly.

    • Estimate: 2 story points

4. As a Student, I want to view my class timetable so that I can attend my classes on time and plan my study schedule.

    • Estimate: 2 story points

5. As an Admin, I want to assign subjects to teachers and create a master schedule to ensure all subjects are covered and there are no scheduling conflicts.

    • Estimate: 5 story points

6. As an Admin, I want to generate a timetable automatically based on teacher availability and room constraints so that the scheduling process is efficient and error-free.

    • Estimate: 8 story points

7. As a Teacher, I want to request timetable changes and view the status of my requests so that I can manage any scheduling conflicts or personal needs.

    • Estimate: 3 story points

8. As an Admin, I want to approve or reject timetable change requests from teachers to maintain an up-to-date and accurate schedule.

    • Estimate: 3 story points

9. As a Student, I want to receive notifications about any changes to my timetable so that I can stay informed about schedule updates.

    • Estimate: 3 story points

10. As an Admin, I want to generate reports on room utilization and teacher workload so that I can analyze and optimize resource allocation.

- Estimate: 5 story point

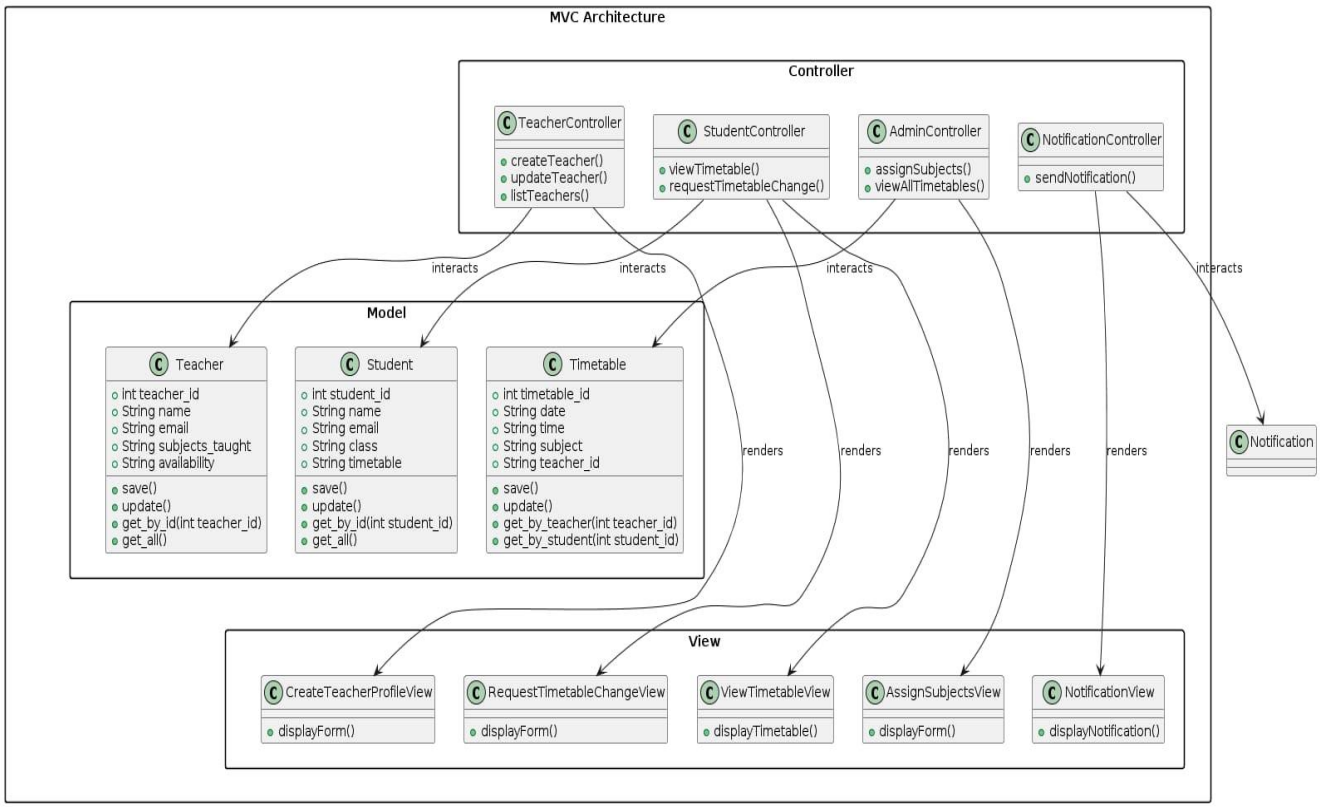
Non-Functional Requirements

Performance: The system should generate a complete timetable for an institution with up to 100 teachers and 1,000 students within 30 seconds to ensure quick updates and user satisfaction.

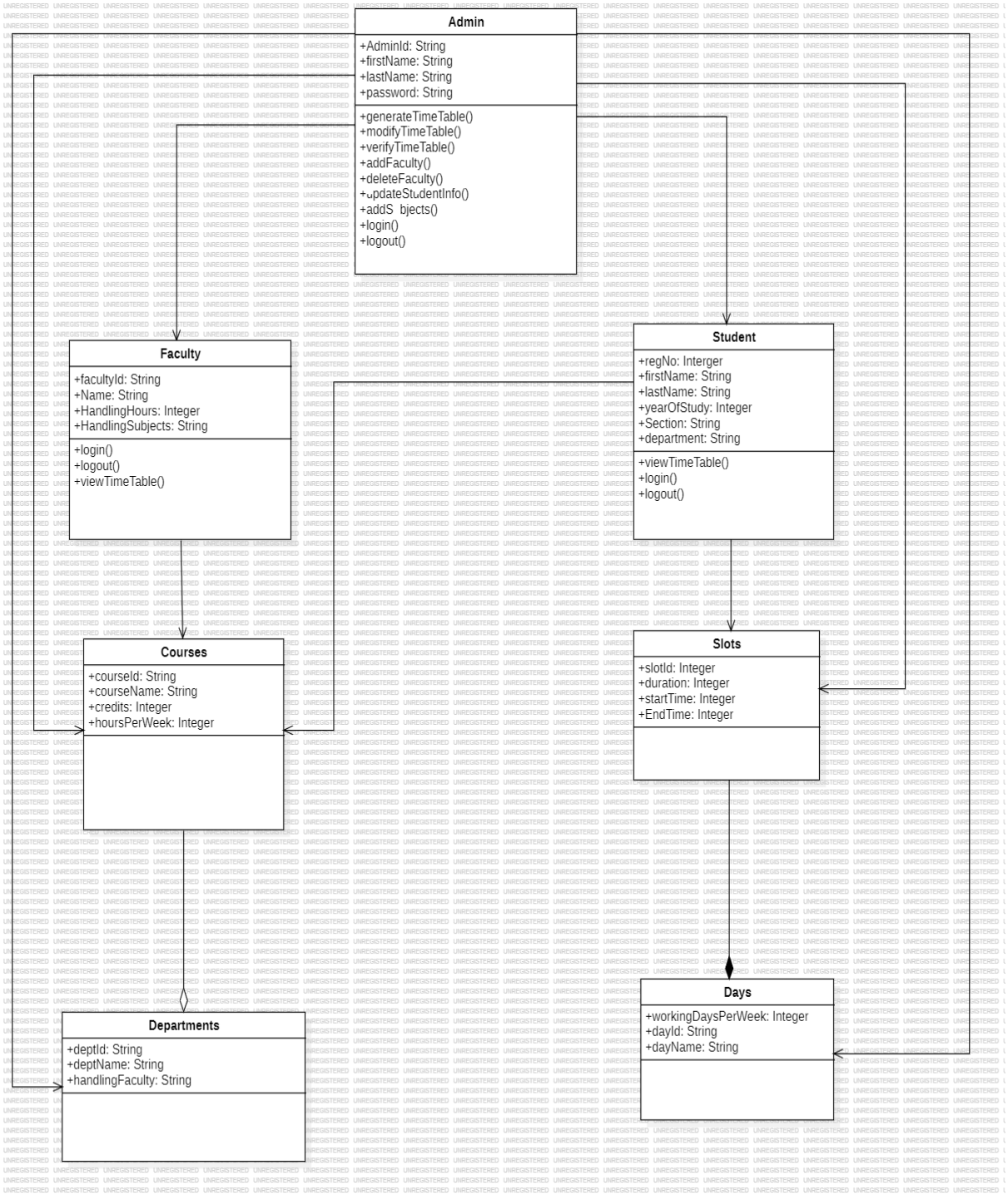
Usability: The user interface should be intuitive and accessible, allowing users to perform tasks such as viewing timetables, requesting changes, and managing profiles with minimal training (less than 1 hour).

Reliability: The system should have an uptime of 99.9% to ensure that users can access their schedules and make necessary updates without significant downtime.

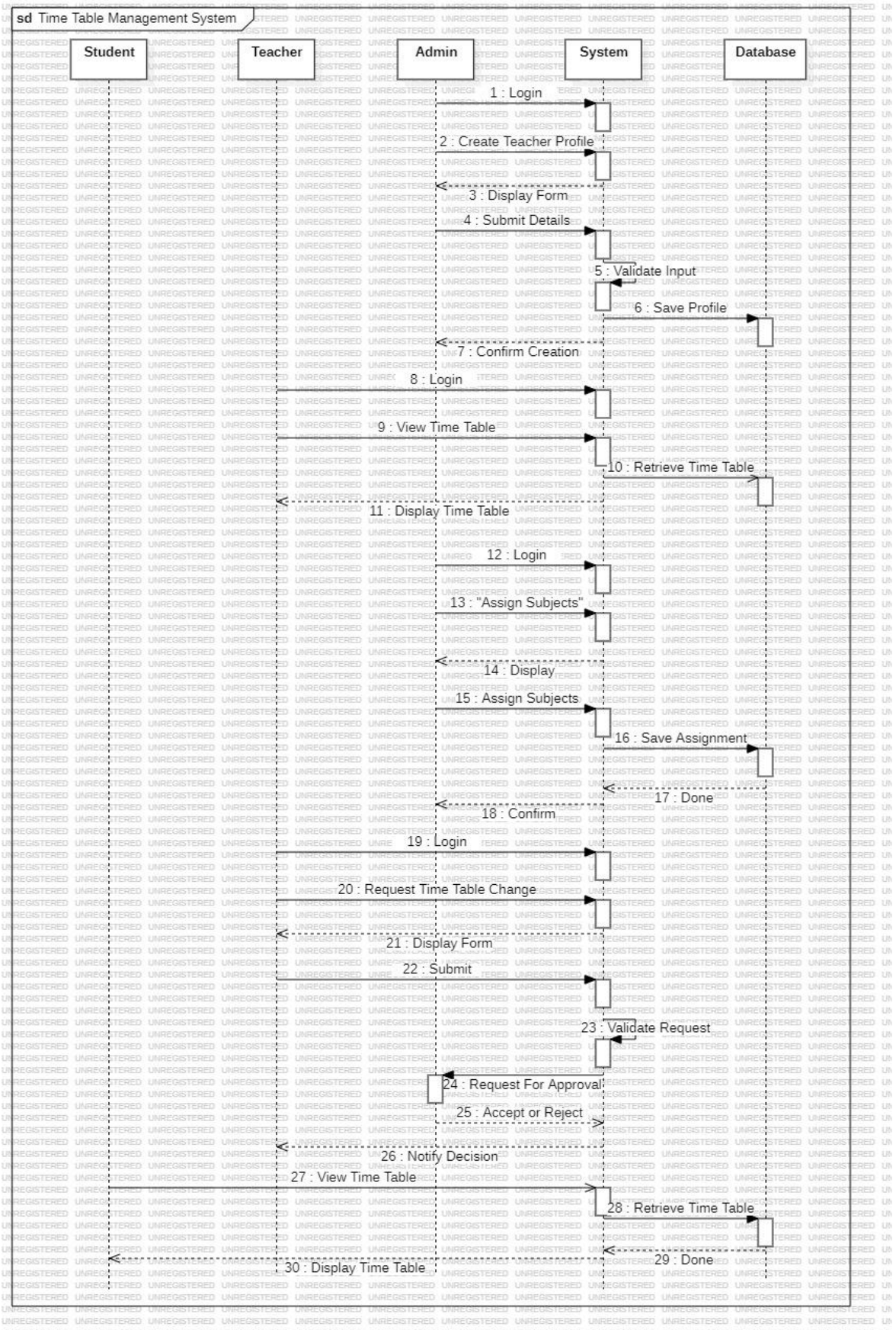
MVC ARCHITECURE



CLASS DIAGRAM



SEQUENCE DIAGRAM



TEST CASES

User Story: Admin Creates Teacher Profiles

Test Case 1: Admin Creates a Teacher Profile (Happy Path)

Test ID : TC001

- Scenario: Admin is logged into the system.

- Steps:

1. Navigate to the "Create Teacher Profile" page.

2. Fill in the teacher's details (name, email, subjects taught, availability).

3. Submit the form.

- Expected Result: Teacher profile is created successfully, and a confirmation message is displayed.

Test Case 2: Admin Submits Incomplete Teacher Profile (Error Scenario)

Test ID : TC002

- Scenario: Admin is logged into the system.

- Steps:

1. Navigate to the "Create Teacher Profile" page.

2. Fill in the teacher's details, leaving required fields (e.g., email) empty.

3. Submit the form.

- Expected Result: Error message is displayed indicating that all required fields must be filled.

User Story: Teacher Views Weekly Timetable

Test Case 3: Teacher Views Weekly Timetable (Happy Path)

Test ID : TC003

- Scenario: Teacher is logged into the system.

- Steps:

1. Navigate to the "View Timetable" page.

- Expected Result: Teacher's weekly timetable is displayed correctly.

Test Case 4: Teacher Views Timetable with No Classes Assigned (Error Scenario)

Test ID : TC004

- Scenario: Teacher is logged into the system.

- Steps:

1. Navigate to the "View Timetable" page.

- Expected Result: Message indicating "No classes assigned for the current week" is displayed.

User Story: Admin Assigns Subjects to Teachers

Test Case 5: Admin Assigns Subjects to Teachers (Happy Path)

Test ID : TC005

- Scenario: Admin is logged into the system.

- Steps:

1. Navigate to the "Assign Subjects" page.

2. Select a teacher from the list.

3. Assign subjects to the selected teacher.

4. Submit the assignment.

- Expected Result: Subjects are assigned successfully, and a confirmation message is displayed.

Test Case 6: Admin Assigns Subjects to Teachers Without Selecting a Teacher (Error Scenario)

Test ID : TC006

- Scenario: Admin is logged into the system.

- Steps:

1. Navigate to the "Assign Subjects" page.

2. Attempt to assign subjects without selecting a teacher.

3. Submit the assignment.

- Expected Result: Error message indicating that a teacher must be selected before assigning subjects.

User Story: Teacher Requests Timetable Change

Test Case 7: Teacher Requests Timetable Change (Happy Path)

Test ID : TC0007

- Scenario: Teacher is logged into the system.

- Steps:

1. Navigate to the "Request Timetable Change" page.

2. Fill in the change request form with valid details.

3. Submit the request.

- Expected Result: Change request is submitted successfully, and confirmation message is displayed.

Test Case 8: Teacher Requests Timetable Change with Invalid Details (Error Scenario)

Test ID : TC008

- Scenario: Teacher is logged into the system.

- Steps:

1. Navigate to the "Request Timetable Change" page.

2. Fill in the change request form with invalid details (e.g., invalid time or date).

3. Submit the request.

- Expected Result: Error message indicating that the details provided are invalid.

User Story: Student Receives Timetable Change Notifications

Test Case 9: Student Receives Timetable Change Notification (Happy Path)

Test ID : TC009

- Scenario: Timetable change is made, and the student is logged into the system.

- Steps:

1. System detects a timetable change.

2. Student receives a notification about the timetable change.

- Expected Result: Student receives a notification message about the change.

Test Case 10: Student Does Not Receive Timetable Change Notification (Error Scenario)

Test ID : TC0010

- Scenario: Timetable change is made, but the notification system fails.

- Steps:

1. System detects a timetable change.

2. Notification system fails to send a notification to the student.

- Expected Result: Student does not receive any notification. (Note: This error scenario is to ensure proper error handling/logging in case of system failures.

DEPLOYMENT ARCHITECURE