**SOFTWARE DESIGN DOCUMENTATION**

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed design for the Student Performance Tracker web application. It outlines the architecture, components, interfaces, and data flow required to develop the system.

1.2 Scope

The application is intended to help educational institutions track, analyze, and display student academic performance. It includes features such as student data management, performance visualization, and performance filtering based on class, subject, and exam type.

2. System Overview

The system is a web-based platform with:

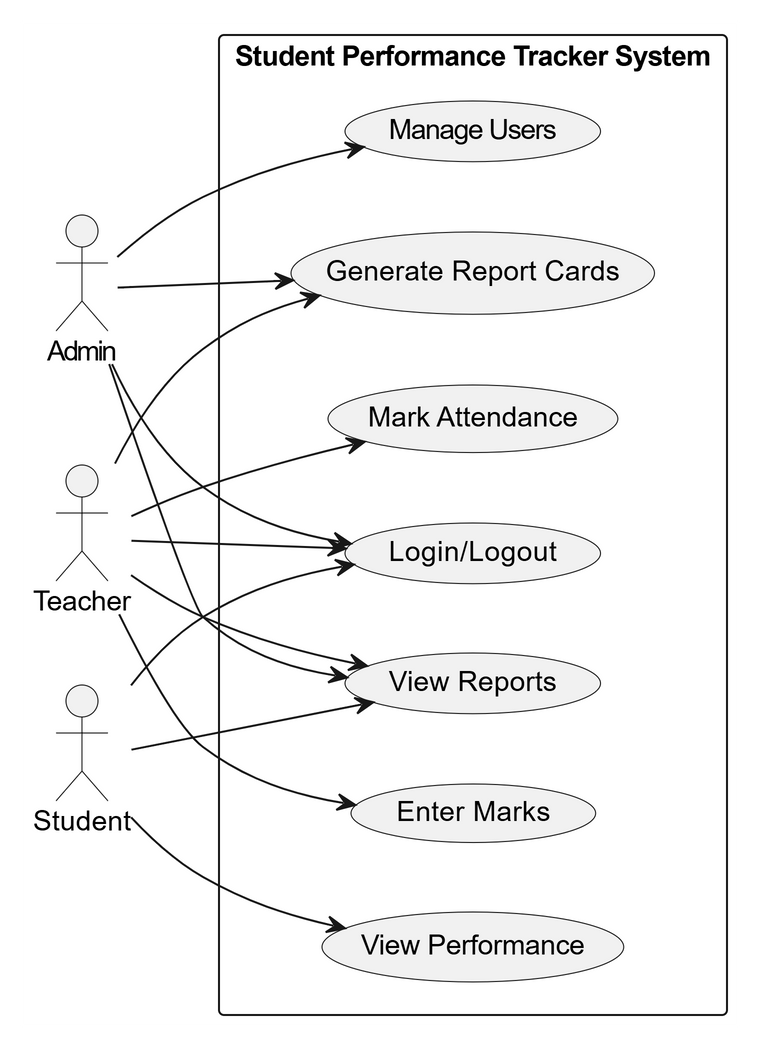
* A **React.js front-end** for a responsive UI
* A **Spring Boot back-end** for business logic and REST API services
* **MySQL** as the database for storing student and performance data

3.Design Constraints

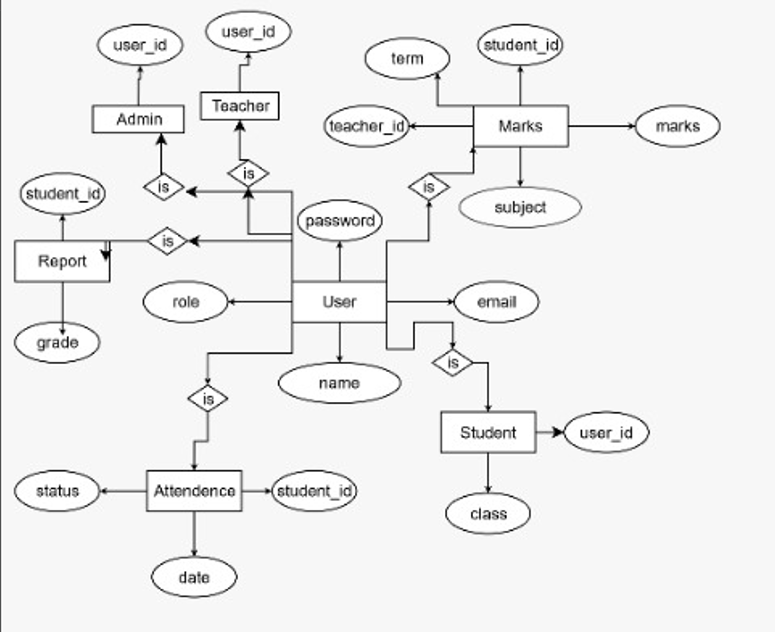
**Frontend**: Built using **React.js** to provide a responsive and interactive user interface for managing and viewing student performance data.

**Backend**: Developed with **Spring Boot**, exposing RESTful APIs to handle business logic, data storage, and communication with the MySQL database.

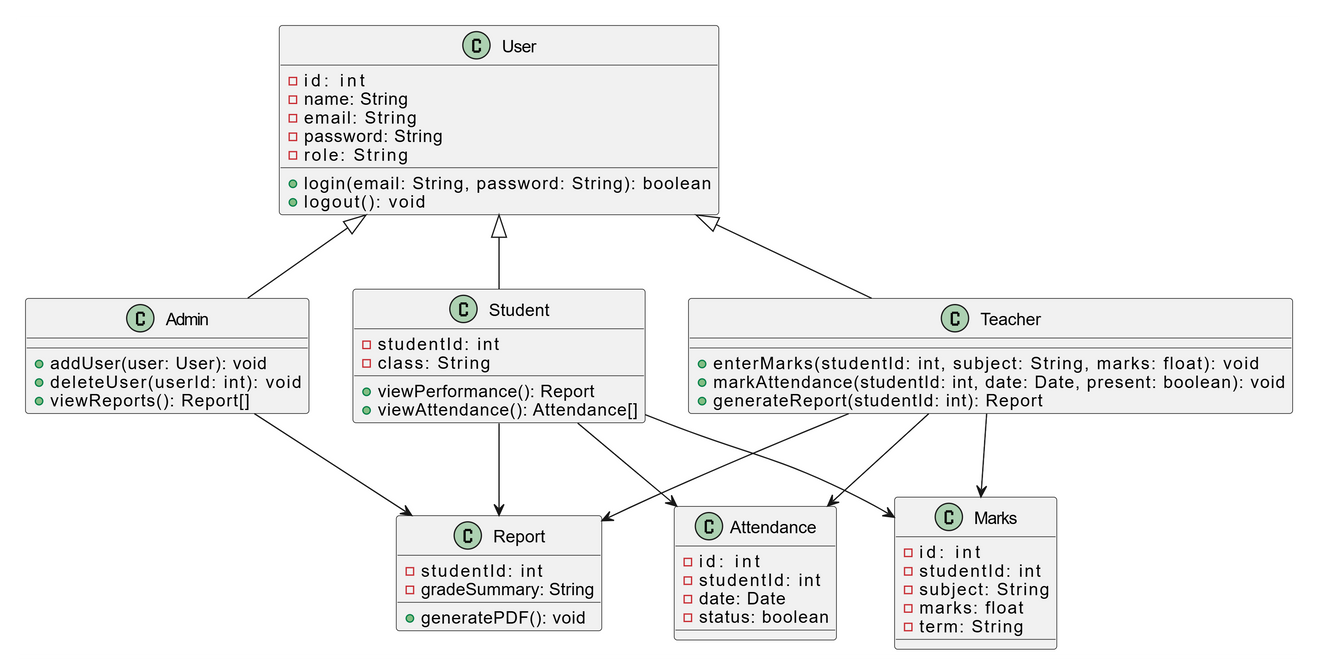
4.UseCase Diagram



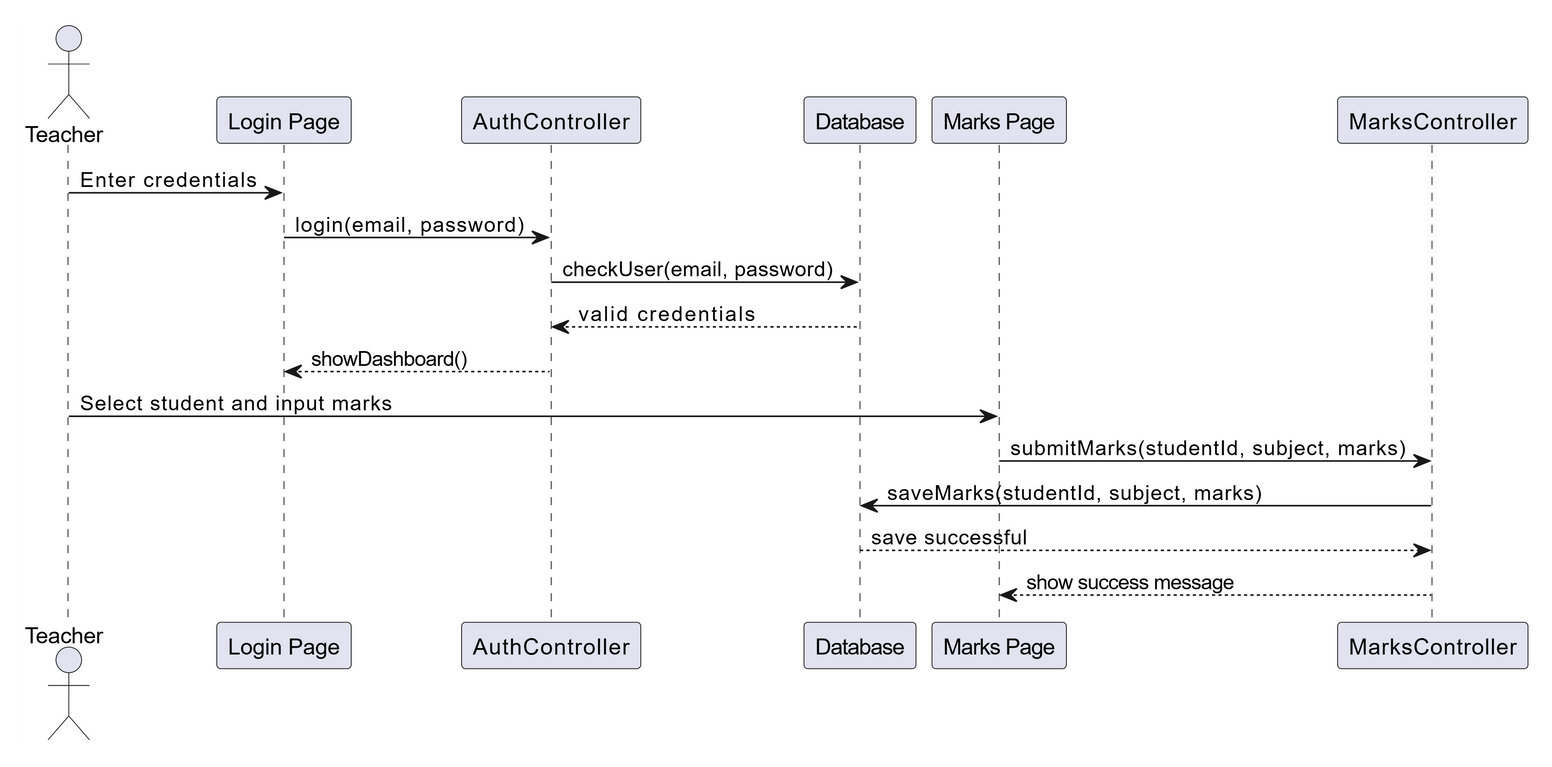
5.ER Diagram



6.Class Diagram



7.Sequential Diagram



8.Conclusion

The Student Performance Tracker system is designed to be a scalable and efficient solution for managing academic records in educational institutions. By leveraging a React.js frontend and a Spring Boot backend integrated with a MySQL database, the system ensures smooth data flow, modular architecture, and responsive user interaction