High-Level Document for Online Judge Project

Project Title:
Online Judge System

1. Introduction

Purpose: Design and implement a full-stack Online Judge using the MERN stack, capable of managing coding challenges and evaluating code submissions.

Target Audience: Coding enthusiasts, students, professionals participating in competitive programming.

2. Project Overview

Description: A platform where users can participate in coding challenges, submit solutions, and have them automatically evaluated.

Key Components:

User Registration & Profile Management : Participants should be able to register for future competitions by providing their personal details such as name, email, and password.

Solution Submission : Participants should be able to submit their solutions to the problems during the competitions. They can upload their code or provide a text-based solution through the platform.

Competition Leaderboard : Participants should be able to fetch the leaderboard of a specific competition. This leaderboard will display the rankings of participants based on their scores in that particular competition.

Practice Problems : The platform should provide practice problems that do not contribute to the scoring or rankings. These problems allow participants to hone their skills and gain experience without the pressure of competition.

Automated Solution Evaluation and Scoring : The platform should have a mechanism to evaluate the submitted solutions against the underlying test cases and generate scores. This evaluation process should be automated to ensure fairness and accuracy in scoring.

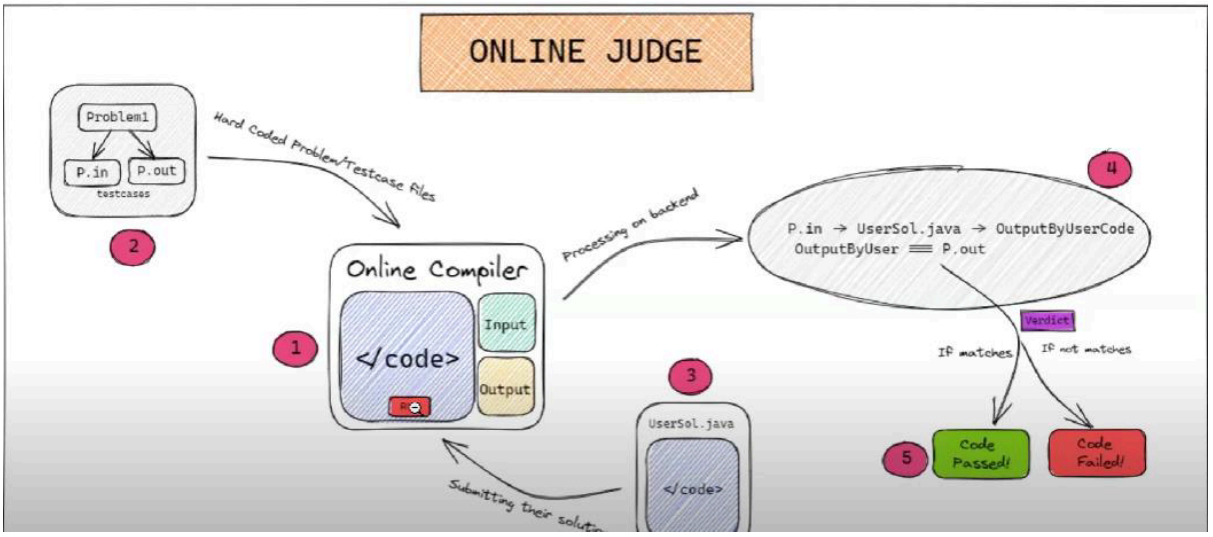
3. System Architecture

Backend: Node.js with Express.js for RESTful API development.

Database: MongoDB for storing problems, solutions, user data, and test cases.

Frontend: React for building a dynamic and responsive user interface.

Evaluation System: Docker containers for code execution and isolation.



4. Database Design

Collection 1: problems - Stores problem statements, names, codes, and difficulty levels.

Collection 2: solutions - Links to problems, stores verdicts, and timestamps of submissions.

Collection 3: test_cases - Contains input-output pairs for each problem.

Collection 4: users - User details including credentials and personal information.

5. Web Server Design

UI Components:

Home Screen: Problem list, Login/Signup options.

Problem Screen: Language selection, code submission interface.

Leaderboard Screen: Rankings and scores.

Backend Logic:
API endpoints for listing problems, individual problem details, code submission, and fetching leaderboard data.

6. Evaluation System Design

Docker-based Code Execution:
Secure and isolated environment for code evaluation.
Limit resource usage and prevent malicious code execution.

Additional Features:
Plagiarism detection using tools like MOSS.
Efficient cache handling for quick response.

7. Features

User Registration: Secure signup process for new users.

Solution Submission: Interface for submitting solutions with real-time feedback.

Profile Management: Users can view their participation history and personal details.

Competition Leaderboard: Real-time rankings based on user scores.

Practice Problems: A separate section for skill development without scoring pressure.

8. Development and Deployment Plan

Development Methodology: Agile approach with iterative development and testing.

Testing Strategy: Unit tests, integration tests, and load testing.

Deployment: Cloud hosting with considerations for scalability and security.

9. Conclusion

This Online Judge system aims to provide a robust and user-friendly platform for coding enthusiasts to challenge themselves and grow their skills in a competitive environment.