**Project Number:** 71

**Project Title:** Code Solving Web Platform

**Project Clients:** Syed Mahbubul Huq

**Project specializations:** Software Development;Web Application Development;Cloud Computing;

**Number of groups:** 2 group

**Main contact:** Syed Mahbubul Huq

## Background:

Coding platforms like LeetCode, HackerRank, and Codewars are platforms to help students develop their problem-solving skills. These platforms provide coding challenges that help users improve their knowledge in programming.

This project aims to create a problem-solving platform similar to LeetCode, focusing on Python as the language for problem-solving. The goal is to allow users to log in and solve coding problems and get feedback in an interactive coding environment. The web application will be built using a microservices architecture.

## Requirements and Scope:

The project scope will cover the development of a coding problem-solving platform focused on Python. It will provide key functionalities such as problem categorization, login management, and an interactive coding IDE for logged-in users. Non-logged-in users will only be able to browse problems but will not be able to interact with them. In the current scope the user writes Python code to solve a problem and submits it through the platform. The backend receives the submission, executes the code in an isolated Docker container, and fetches the relevant test cases from MongoDB. The code is then run against the test cases, and the results are compared to determine whether the solution is correct.

The specific scope includes:

User Authentication & Management

Problem Categorization & Management

Coding Environment (IDE)

Non-logged-in User Interaction

Leaderboard (Basic)

Admin Panel

## Required Knowledge and skills:

User Authentication & Profile Management

User Registration and Login: Users can sign up by providing email, username, and password, or log in to an existing account.

Session Management: Users will remain logged in unless they manually log out.

Profile Management: Users can update their personal profile, including changing their display name, profile image, and adding a bio.

Problem Management

Problem Categories: Problems will be divided into three categories: Easy, Medium, and Hard.

Problem Creation: Admins can create new problems with a problem statement, expected outputs, and test cases.

Problem Solving: Logged-in users will be able to attempt solving the problems and test their solutions.

Problem Status: Users can mark problems as solved upon successful execution of their code.

IDE Integration for Coding

Python IDE: A real-time Python coding environment will be available for solving problems.

Code Execution: Users can run their code and receive feedback on whether their solution is correct, along with any error messages or outputs.

Non-logged-in Users

Problem Viewing: Non-logged-in users can view the problems, their descriptions, and their categories but cannot interact with the IDE.

Leaderboard

Display Leaderboard: A leaderboard will show the top users based on the number of problems solved (logged-in users only).

Admin Panel

User Management: Admins can view and manage user data.

Problem Management: Admins can manage problems, including creating new ones or deleting existing ones.

Mobile/Desktop Responsiveness

The platform should work seamlessly across both mobile devices and desktop computers.

Scalability

The platform should be scalable to support an increasing number of users and problems.

The backend should handle a growing amount of users simultaneously without performance degradation.

Performance

Code execution should be quick, with results appearing within a reasonable timeframe.

The system should handle concurrent users efficiently.

Security

Data Encryption: All sensitive user data will be securely encrypted (e.g., passwords).

Authentication: Use OAuth or JWT tokens to ensure secure, stateless authentication.

Data Privacy: Users' personal information and solution data will be stored securely and in compliance with regulations.

## Expected outcomes/deliverables:

Complete project documentation; documented source code, requirements, design, software architecture, testing, technology stack and technical dependcies/libraries/frameworks, automated deployment, user manual and deployment guide