

High Level Design Document

□ Database Design

approx. 1 day

1. Problem

- Name: string
- Description: string
- Problem ID: string
- Difficulty: string (easy, medium, hard)
- Tag: string
- Constraints: string
- Sample Input: string
- Sample Output: string
- Input Test Cases: string
- Output Test Cases: string

2. Submission

- Problem ID: string
- Submissions: [Language: string,
Code: string,
Verdict: string,
Userid: string,
Timestamp: dateTime field]

3. User

- UserID: string
- Name: string
- Email: string
- Password: string
- Token: Array

□ Web Server Design

- Number of UI screens: 6(Home, Register, Login, Problems, Problem Statement, Submissions). *approx. 2 days*
- Problem Page UI: *approx. 2 days*
 - ✓ Divided into 2 displays left and right
 - ✓ Left one for question display and right for code writing text area.
 - ✓ Right one display has the footer with option submit and header with dropdown menu of language selection.
 - ✓ Output Display beneath the run and submit code button.

- Functionalities:

- ✓ User Authentication Page *approx. 2 days*

- Register & Login
- Login session handle using JWT tokens

- ✓ Problem List *approx. 1 day*

- A simple list consisting of names of each problem, linking it to the individual problem's page.
- GET request to fetch all problem names, tag, difficulty from Problem Schema and return to UI.

- ✓ Show Individual Problem *approx. 3 days*

- Show problem statement along with the code area with run and submit button and language change dropdown menu by clicking on problem name from problems List.
- GET Request to fetch the problem details from the Problem Schema and return to UI.

- ✓ Code Submission *approx. 4 days*

- Code is pasted or written in the text area and submitted using the submit button beneath it.
- POST request to backend to handle execute following:
 - Get the test cases (Test Case Input and Test Case Output) for the Problem Schema.
 - Compare the outputs from the compiler result to the Test Case Output in DB.
 - Save the verdict for this submission in DB.
- All Submission Tab:
 - A list showing the verdict of all the users.
 - GET request fetch the submissions with user's userid, language, verdict, code, when submitted details from submission schema.

□ **Code evaluation system**

approx: 4 days

- Use of Docker for making containers used for AWS hosting using aws-ecr security purposes.
- Code compilation uses the npm child process.
- Single Insolation will be done.