Prepared By:

Mohammad Enan Al Harun Sahan,

Reg No. 20101095

Section: B2

MD. Asadujjaman Noor

Reg No. 20101101

Section: B

Sheikh Nafez Sadnan

Reg:20101106

Section: B

Presented to:

Fahad Ahmed

Lecturer, CSE, UAP

Group Number

B2-G5

Project Title

An Online Judge for programming contests.

Project Repository Link

https://github.com/asadcop/cponlinejudge

Project Members

Mohammad Enan Al Harun Sahan	Reg: 20101095
MD. Asadujjaman Noor	Reg:20101101
Sheikh Nafez Sadnan	Reg:20101106

Problem Definition

Creating an interactive platform such as online judge comes with various problems. Here is a few of the problem we have noted:

- Server Load
- Economic Cost
- Responsiveness
- Accuracy

Server Load: In a contest there will be multiple people using the platform at the same time. Multiple people will be uploading multiple files and data. If multiple contests are being hosted at the same time the load increases by a lot. We will need proper server management and a good server to implement the project.

Economic Cost: Maintaining a good server continuously not only is difficult but also costly. We have determined we can offer premium contest hosting platform or include third party advertisement to cope with the ongoing cost.

Responsiveness: This will be an interactive platform. So good response is a must. Users need to be informed about their submission as soon as possible. Good server maintenance can resolve the issue.

Accuracy: Here accuracy is pointing to the systems problem checking capability. A single solution can have multiple variants. Checking the solution thoroughly and accurately to make sure no judgment error is there is a must. Our admin panel will ensure and continuously maintain this aspect of the system.

Objective

Our objective is to create an online platform where users will be able participate in programming contests or host contests to provide other users the chance to take part in competition.

- This website will host problems for users to practice.
- This system will host programming contests for competition.

Solution and Project Outputs

Solution:

Notable features for us to tackle:

- Creating and hosting contests

- Collaborative Code Editor – Allowing different users to work on the same solution simultaneously.

Building a proper server with program such as **MongoDB** (storing data) or **Express** (to host web server). Our current choice of building the client is Angular or third-party libraries.

We are hoping to provide users the freedom to choose their own programming language to solve a problem unless the contest is bound to any specific language. Thus, we will be building a judge engine capable of automatically running and detecting different languages.

Project Outputs:

This will give us the opportunity to improve our knowledge of building an interactive website platform. Allow programmer community to hone their skills through competition. Here are few notable points:

1. **An online platform:** A web-based platform that allows users to submit and test their code for a range of problems.
2. **Problem Management System:** A system for creating, managing, and storing problems and their test cases.
3. **User Guide:** A user guide that provides instructions on how to use the online judge platform.

Requirement Analysis

Basic Requirements:

Performance: Our target is to make a quick responsive website that will confirm the users the results within a few seconds.

Information: The information collected will be user provided. Their email will be collected to create and store data for the account. Their contests data will be collected from the code or program they submit.

Economy: Using a less responsive server for contest that doesn't need to provide immediate leaderboard might reduce the cost. Most profits will be expected to come through ad revenue or private contests. Estimated development time is 3 months.

Control: The privacy requirements for the users are just their email and passwords. Contact info. Will be collected if the private contest requires.

Efficiency: Non frequently used programming languages won't be available for users to increase efficiency in evaluation.

Service: The service will be interactive. Our target audience are programmers. There will be three types of users in this system.

1. Regular users,
2. Contest hosts and
3. Admin panel.

Admin panel will be maintaining the backend and front end.

Functional Requirements:

An online judge for competitive programming should have the following functional requirements:

- **Authentication:** The system should have measures in place to prevent code injection, hacking, or other malicious activities. This starts with an authentication system in place. Sectors including authentication are:
  - Registration
  - Log in
  - Profile

- **User data management:** Profile will contain users all required info. History portion will contain their competitive history, solved problems list and progress in any contest they participated in.

- **Code execution and compilation:** Live updates are one the core functions for an online judge. The system should be able to compile and execute user-submitted code in real-time.

- **Input/Output handling:** The system should be able to handle both standard input and output, as well as input and output files.

- **Test case management:** The system should allow the creation, management and execution of test cases for problem submissions.

- **Problem statement and resource management:** The system should allow for the creation and management of problem statements and resources such as sample input and output files.

- **Time and memory limits:** The system should enforce time and memory limits to prevent infinite loops or excessively long computations.

- **Verdict generation:** The system should be able to generate verdicts such as "Accepted", "Wrong Answer", "Time Limit Exceeded", etc. based on the results of code execution and comparison with expected results.

- **Reporting and feedback:** The system should have mechanisms for reporting bugs, issues, and providing feedback to the users.

Non-functional requirements:

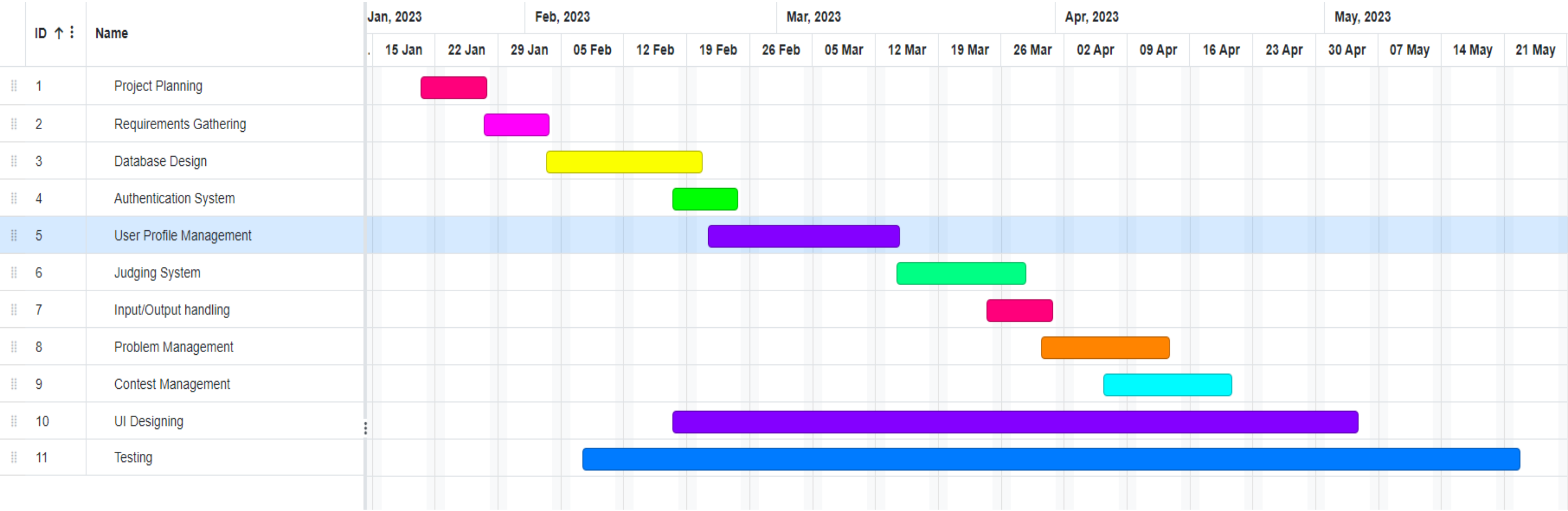
- **Scalability:** The system should be designed to scale to handle a large number of users and submissions.

- **Portability:** The system should be able to be ported to other hardware systems. The portability allows the system to be easily installed and deployed by third parties to provide services on demand.

- **Availability:** The system must be continuously running and ready to accept submission at all time.

Project Management:

Activity	Required Time
Requirements gathering	1 Week
User authentication system	2 Weeks
User profile management	3 Week
Question management	2 Weeks
Database management	3 Weeks
Judging system	3 Weeks
Contest Management	2 Weeks
UI	4-5 Weeks



ID ↑	Name
1	Project Planning
2	Requirements Gathering
3	Database Design
4	Authentication System
5	User Profile Management
6	Judging System
7	Input/Output handling
8	Problem Management
9	Contest Management
10	UI Designing
11	Testing

Project Cost:

Activity	Time Cost	Estimated expense in Taka
Information gathering	100 Hours	50000
Pre-production setup	300 Hours	50000
Setting up user experience	300 Hours	100000
Database setup	300 Hours	50000
Server management	Project lifetime	200000+
UI development	300 Hours	125000
Documentation	50 Hours	25000
Approximate Project Costs	1350 Hours	600000