Project Report SUST SWE Online Judge

By

Prottya Roy Chayan

2018831001

Md. Jamilur Rahman

2018831035

Department Of Software Engineering
Shahjalal University Of Science and Technology, Sylhet

To

Dr. Ahsan Habib
Assistant Professor

Institute of Information and Communication Technology
Shahjalal University Of Science and Technology, Sylhet

November 12, 2022

Contents

1 Introduction **2**

2 Requirement Gathering **2**
2.1 User Registration and Login management module . . . 2
2.2 Problem Creation and Test Case setup module . . . 3
2.3 Problem solving and Judge engine module . . . 3
2.4 Contest Creation and management module . . . 3

3 Use Case Diagram **3**
3.1 **Overall Use case:** . . . 3
3.2 **Use Case Diagram for only Participant:** . . . 5
3.3 **Use Case Diagram for only Judge:** . . . 6
3.4 **Use Case Diagram for unregistered user:** . . . 7

4 Data Flow Diagram **7**
4.1 Registration and Login Management Module . . . 8
4.2 Problem Creation and Test Case Setup Module . . . 9
4.3 Problem solving and Judge engine module . . . 9
4.4 Contest Creation and management module . . . 10

5 ER Diagram **10**
5.1 User Entity . . . 10
5.2 Problem Entity . . . 11
5.3 Contest Entity . . . 11
5.4 Submissions Entity . . . 11
5.5 Test Case Entity . . . 12

6 UI Design **13**

7 Database Diagram **21**

8 Process Flow **21**

9 Implementation **23**

10 Testing **23**

11 Training and Maintenance **23**
11.1 Non-Functional Requirements . . . 24

1 Introduction

For our 350 project we decided to build an online judge system which would be helpful for all of us in many ways. We are a department where problem solving using various coding languages is prominent and we participate in different contests but we do not have an online judge system of our own. That is why we wanted to build an online judge system where we can upload problems that we want other to practice and we can even host many different contests by using the problems set by the judge. This system can give us a flexibility in hosting contests for our students and giving a platform to test out their skills of creating problems for others to solve. Before starting the project work we gathered information from people who regularly use online judge systems to practice or set problems and host different contests.

2 Requirement Gathering

We approached people in our university who are connected to the culture of problem solving and have experience of setting up contests for other students to participate. They gave us some valuable information about the whole process and the functionalities we need to look out for while designing and building the system. we were able to get a rough idea of the modules and processes we need to have in the online judge system.
At first we got to distinguish the modules related to the projects.

Modules:

1 User Registration and Login management module

2 Problem Creation and Test Case setup module

3 Problem solving and Judge engine module

4 Contest Creation and management module

2.1 User Registration and Login management module

This is one of simplest module in the whole system. As the name suggests in this module we register our users and manage their login activities and information. During the register process it is required to distinguish the user who can set contest and who cannot. This would give us a great clarity about the participants and judges. The main entity found in this module is the user entity which has some really mandatory and optional attributes about those we are going know in detail later. The main processes that we find in this module is the registration and login process.

2.2 Problem Creation and Test Case setup module

This module is certainly the module where the users can create new problems with the test cases and sample outputs. Here we get one strong entity that is the Problem entity that contains two weak entities in itself that are the sample input and output entities. These three entity work in harmony to give us the opportunity to set problems for others to practice. Here we have one process that enables us to set up the problems which is called problem creation process and test case set process

2.3 Problem solving and Judge engine module

This is another of our main modules that works on the problem solution submission and verdict generator. The main entity is the solution submission entity that stores all the submitted solutions by our users and we have a judge engine which compiles the submitted code and works on generating the an output for the test cases provided by the problem setter. The judge engine also does the work of comparing the output with the provided outputs to know if the solution is actually correct or not. This whole judge engine is in itself a process that we are going to design and the submission would be done through the problem solve or problem submission process.

2.4 Contest Creation and management module

The last of our module is the contest creation and management module where an authorized person aka the judge can set up a contest for a certain duration with the preferred problemset. This whole contest details come together to make the contest entity and the whole creation and maintenance would be done through the contest creation process which would handle the ranking system generation too.

3 Use Case Diagram

We have many Use Cases in the system but we have two different type of actors in the use case and they are normal users and judge.

3.1 Overall Use case:

Here we have some of the most important use cases. Some of them are,

1 Register and Login

2 Problem Creation

3 Problem solving

4 Contest Creation

5 Contest Participation.

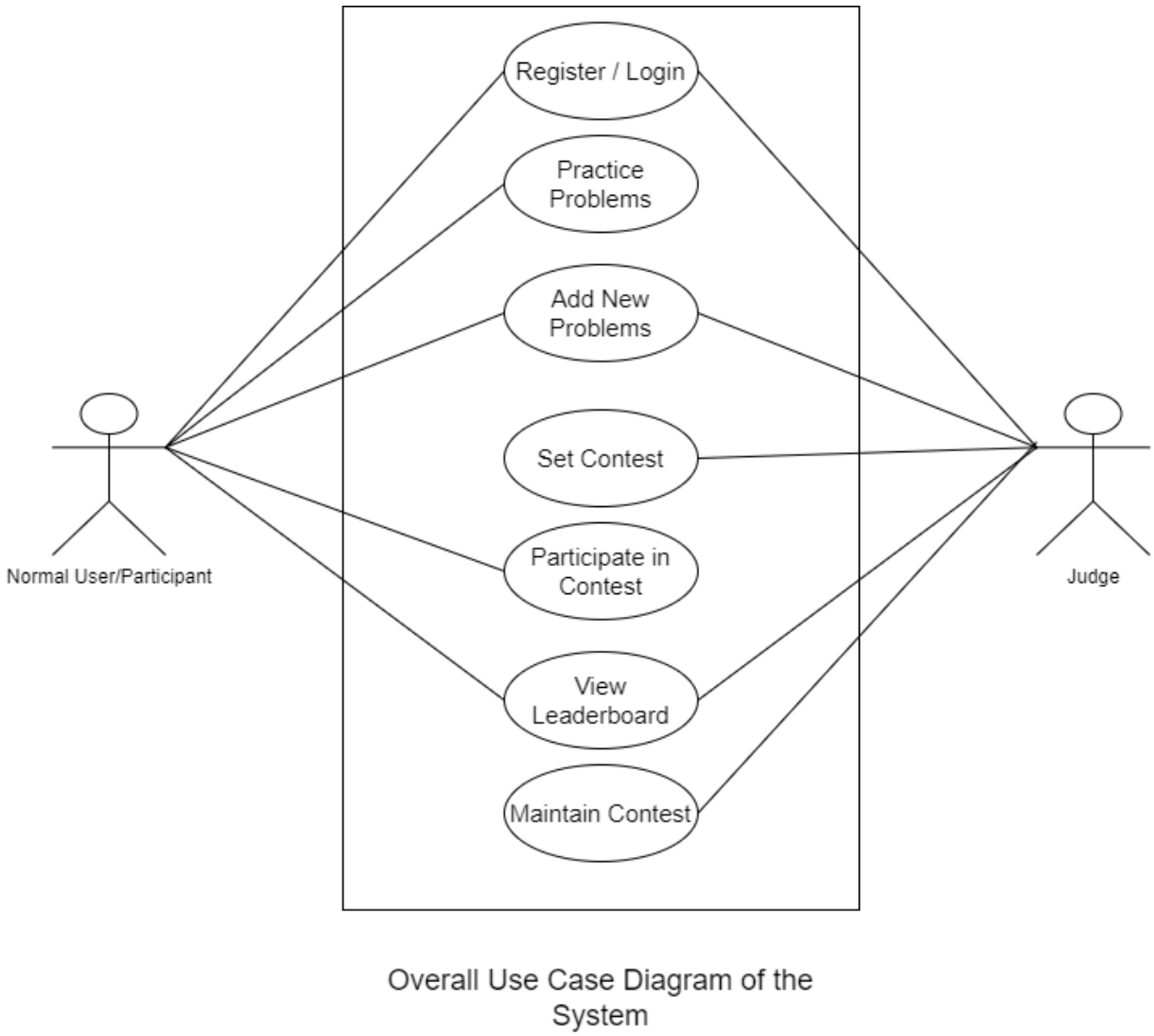

Figure 1: Overall Use Case Diagram

3.2 Use Case Diagram for only Participant:

Here we have some of the most important use cases that are done by the actor User/Participant.

1 Register and Login.

2 Problem Creation.

3 Problem solving.

4 Contest Participation.

5 Viewing existing problems.

6 Viewing the contest leaderboard.

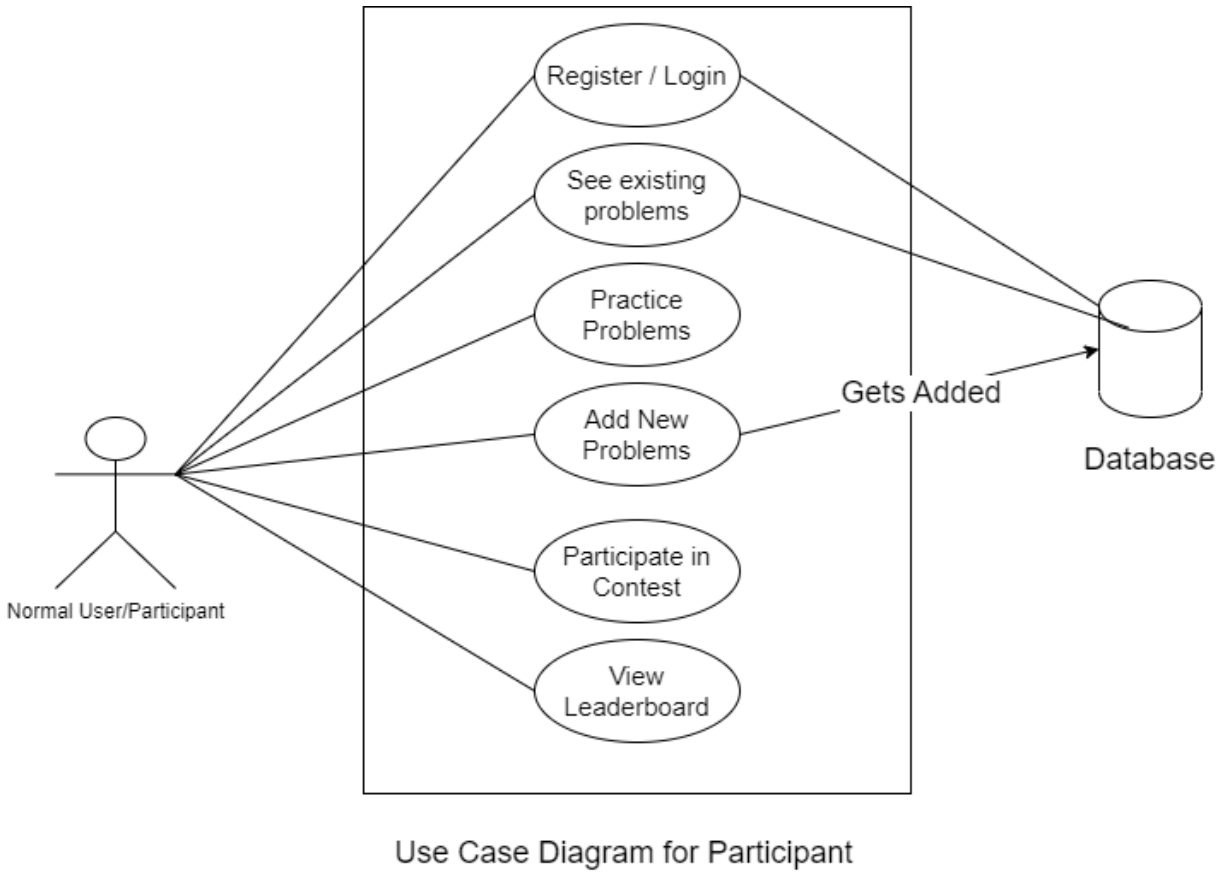

Figure 2: Use Case Diagram Particpant

3.3 Use Case Diagram for only Judge:

Here we have some of the most important use cases that are done by the actor Judge.

1 Register and Login.

2 Problem Creation.

3 Problem solving.

4 Contest creation.

5 Viewing existing problems.

6 Viewing the contest leaderboard.

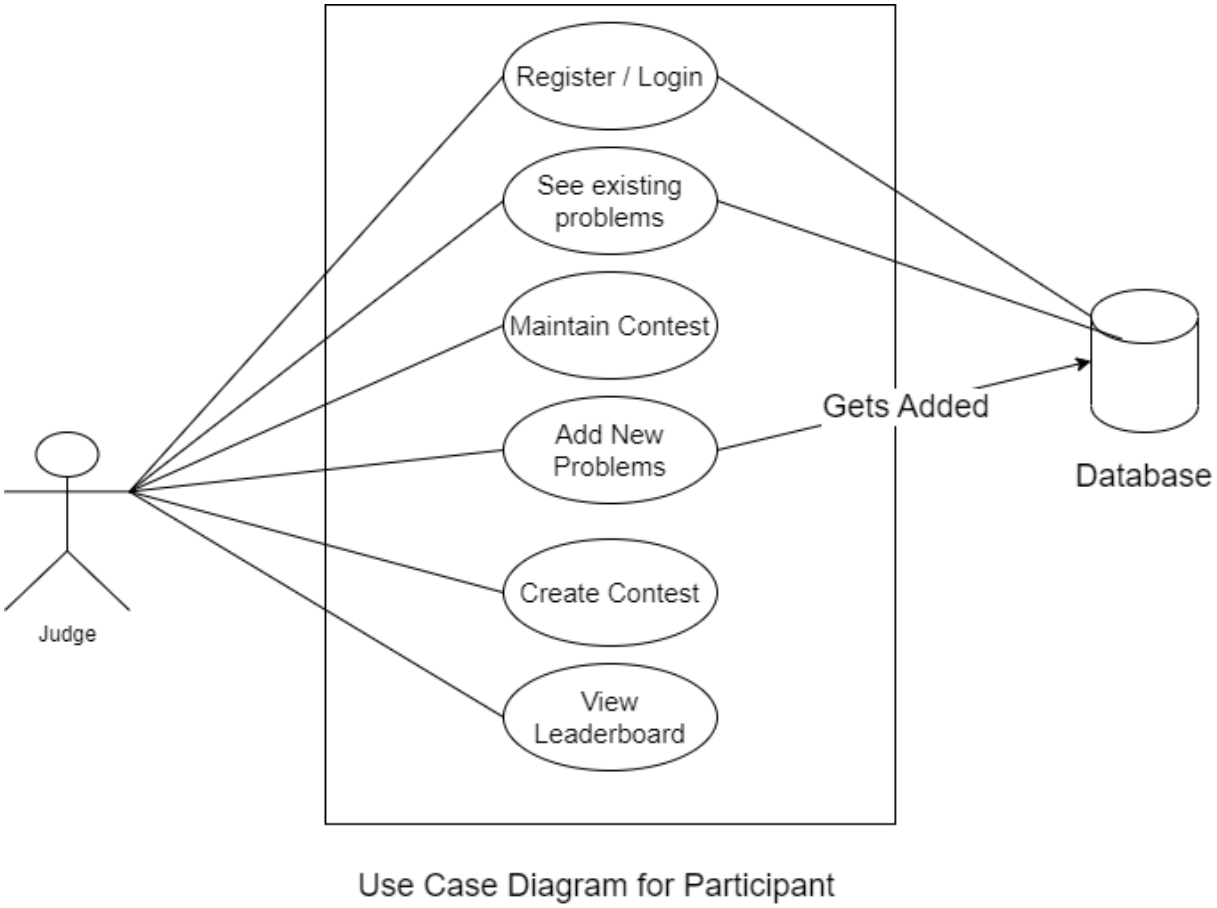

Figure 3: Use Case Diagram Judge

3.4 Use Case Diagram for unregistered user:

Here we have some of the most important use cases that are done by the actor Judge.

1 Register and Login.

2 Viewing existing problems.

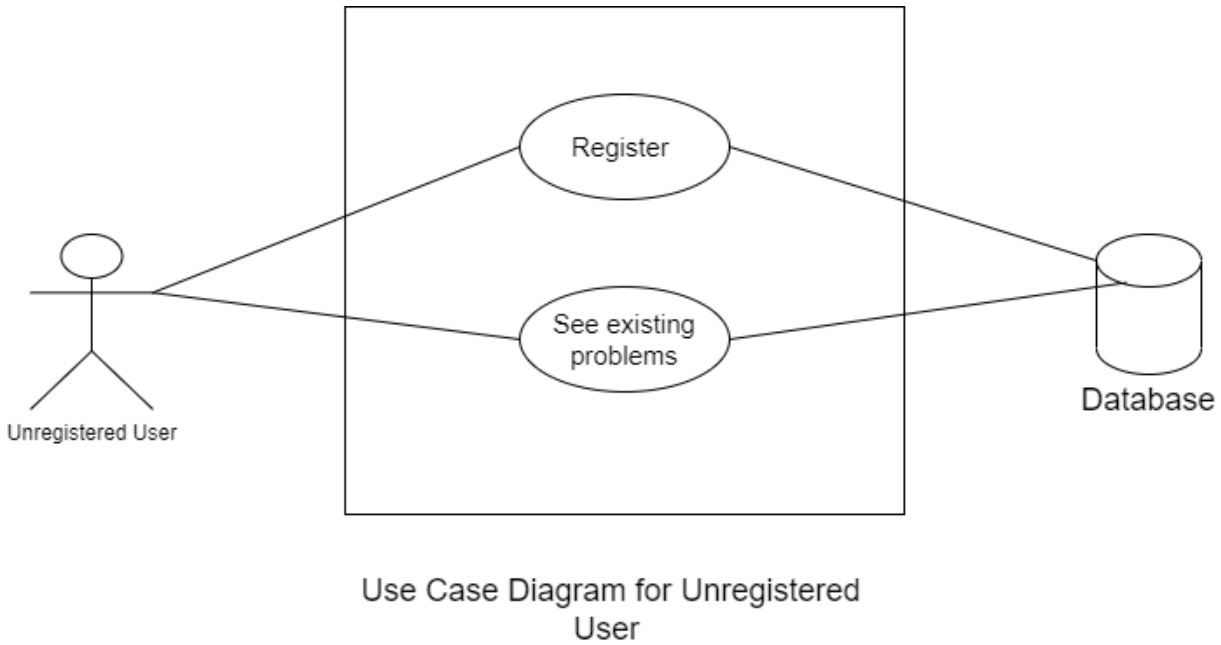

Figure 4: Use Case Diagram Unregistered User

4 Data Flow Diagram

We two Data Flow Diagrams(DFD) here which are level-1 and level-2 DFDs. **Level-1 DFD**
In the level-1 DFD we give a simple data flow within the project user and the database.
It has some major processes :

1 Manage User Registration process.

2 Manage User Login process.

3 Manage Problem Creation process.

4 Manage Problem Solve process.

5 Manage Contest Creation process.

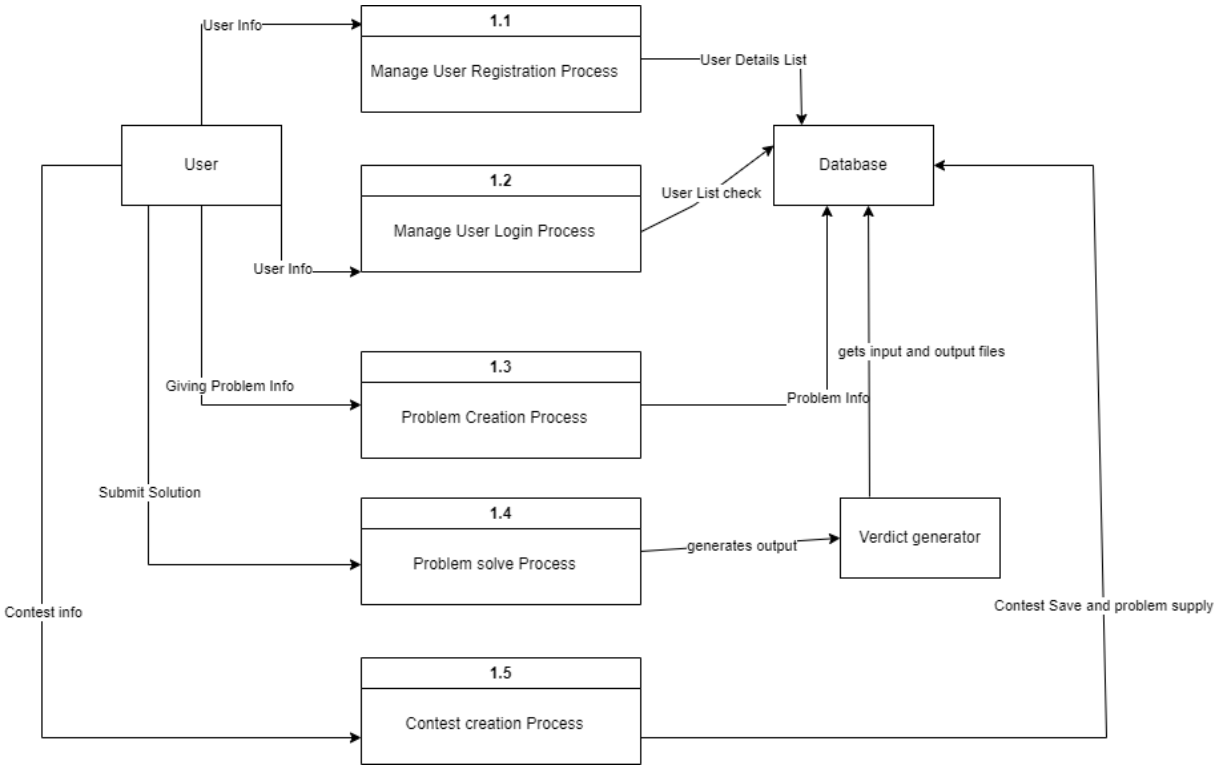

Figure 5: Level-1 Data Flow Diagram

4.1 Registration and Login Management Module

We have a level-2 DFD of the Registration and Login Management Module.

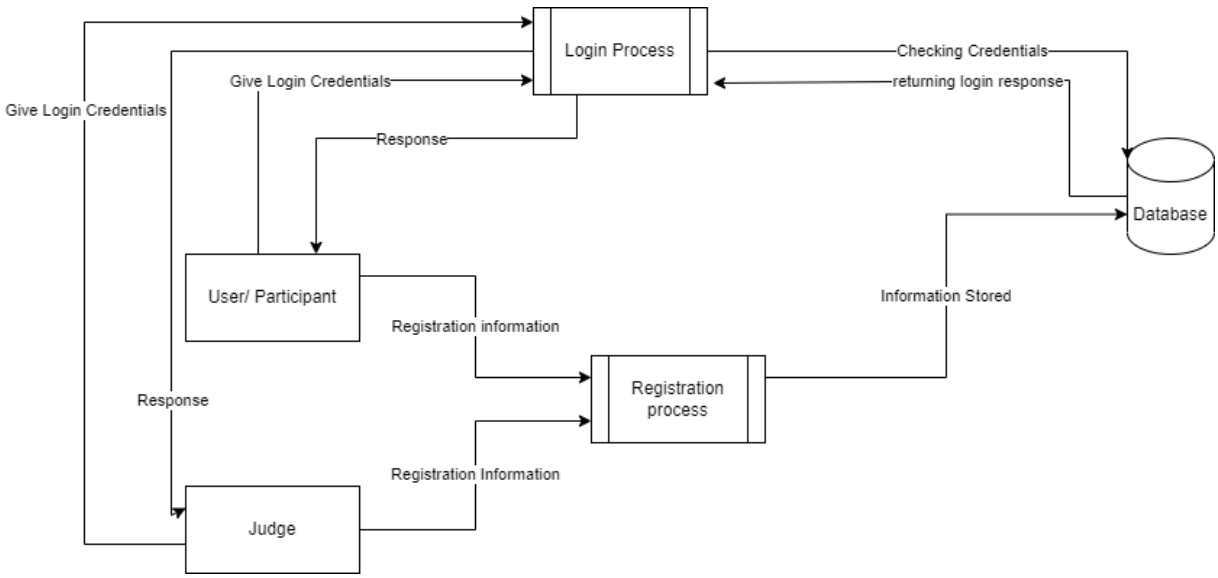

Figure 6: Level-2 DFD of Registration and Login Management Module

4.2 Problem Creation and Test Case Setup Module

We have a level-2 DFD of the Problem Creation and Test Case Setup Module.

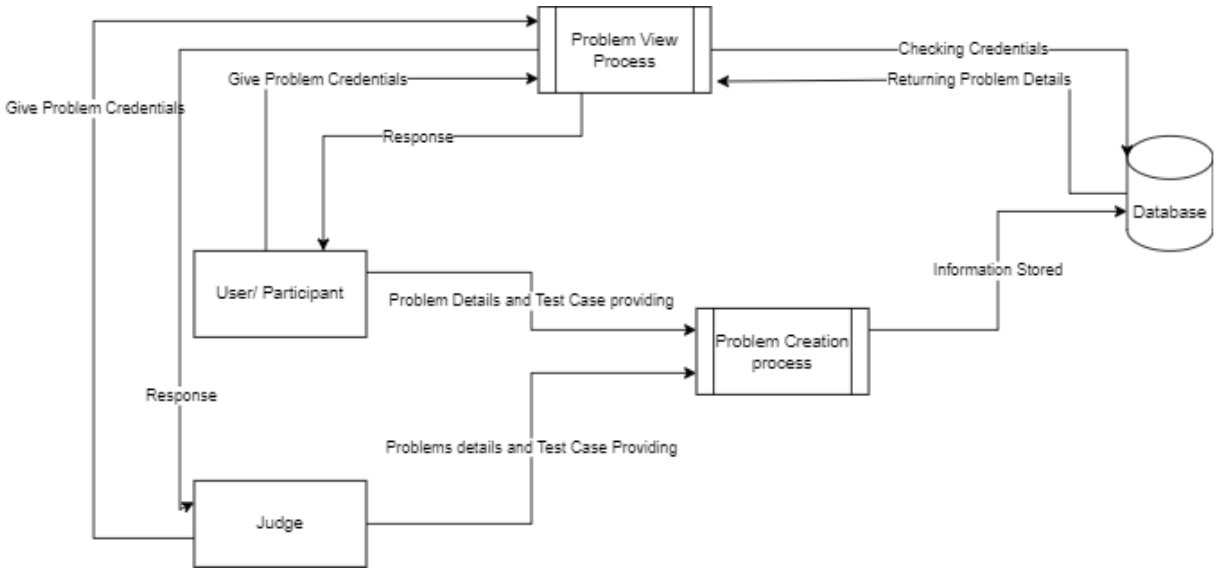

Figure 7: Level-2 DFD of Problem Creation and Test Case Setup Module

4.3 Problem solving and Judge engine module

We have a level-2 DFD of the Problem solving and Judge engine module.

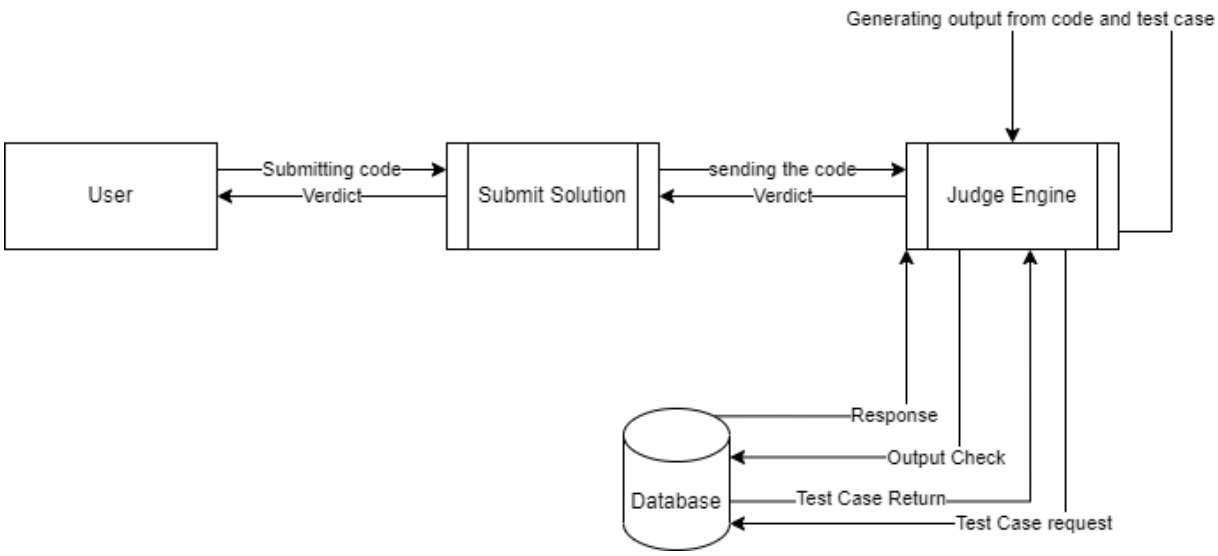

Figure 8: Level-2 DFD of Problem solving and Judge engine module

4.4 Contest Creation and management module

We have a level-2 DFD of the Contest Creation and management module.

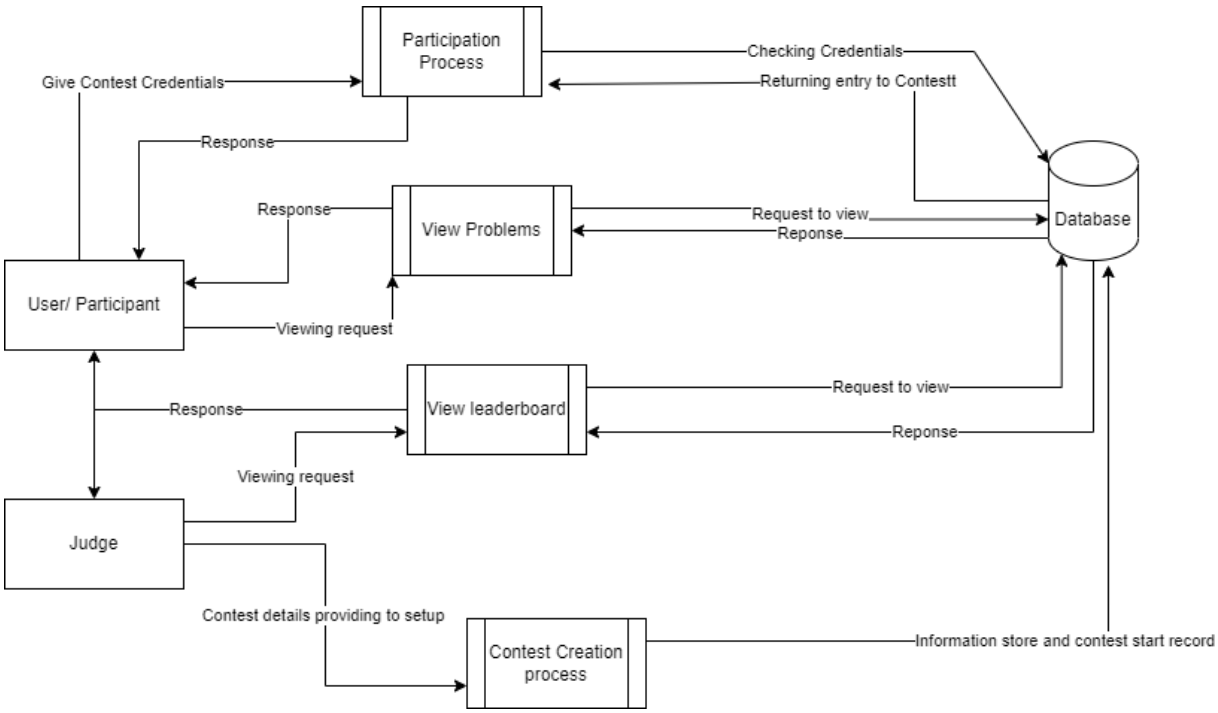

Figure 9: Level-2 DFD of Contest Creation and management module

5 ER Diagram

Here we have 4 Strong. Namely,

1 User

2 Problem

3 Contest

4 Submission

We also have one weak entity. That is,

1 Test Cases

5.1 User Entity

This strong entity has some attributes that are listed below :

1 Id

2 Name

3 Username

4 Password

5 email

6 phone number

7 Date of Birth

5.2 Problem Entity

This strong entity has some attributes that are listed below :

1 Id

2 Name

3 Description

4 Test Case

5.3 Contest Entity

This strong entity has some attributes that are listed below :

1 Id

2 Name

3 Description

4 Password

5 Start time

6 End time

5.4 Submissions Entity

This strong entity has some attributes that are listed below :

1 Id

2 Solution

3 Language

4 verdict

5.5 Test Case Entity

This strong entity has some attributes that are listed below :

1 case No

2 Input

3 Output

The Entity Relationship(ER) diagram is given below:

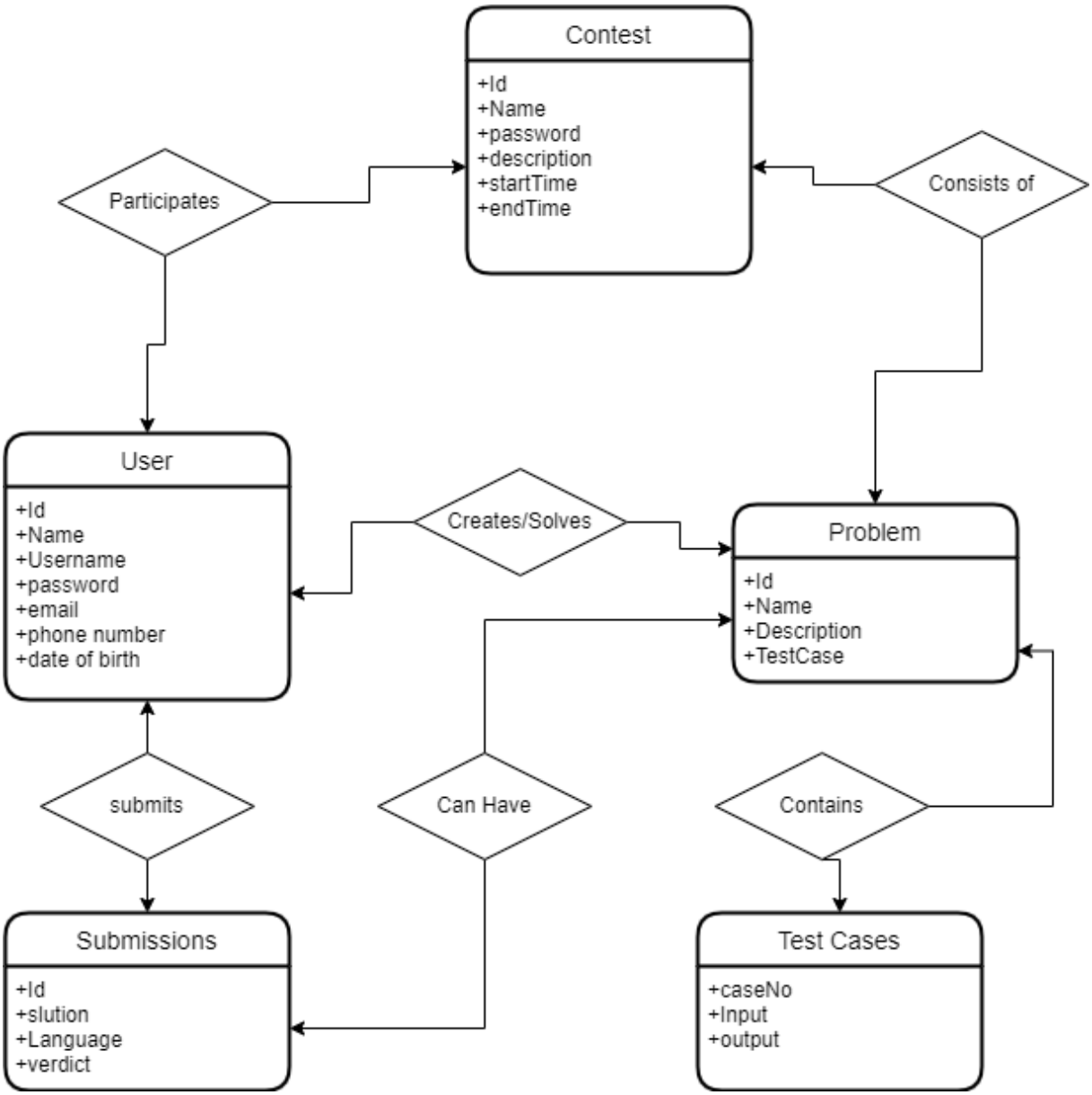

Figure 10: Entity Relationship Diagram of the System

6 UI Design

The UI designs of all our pages are shown below serially.

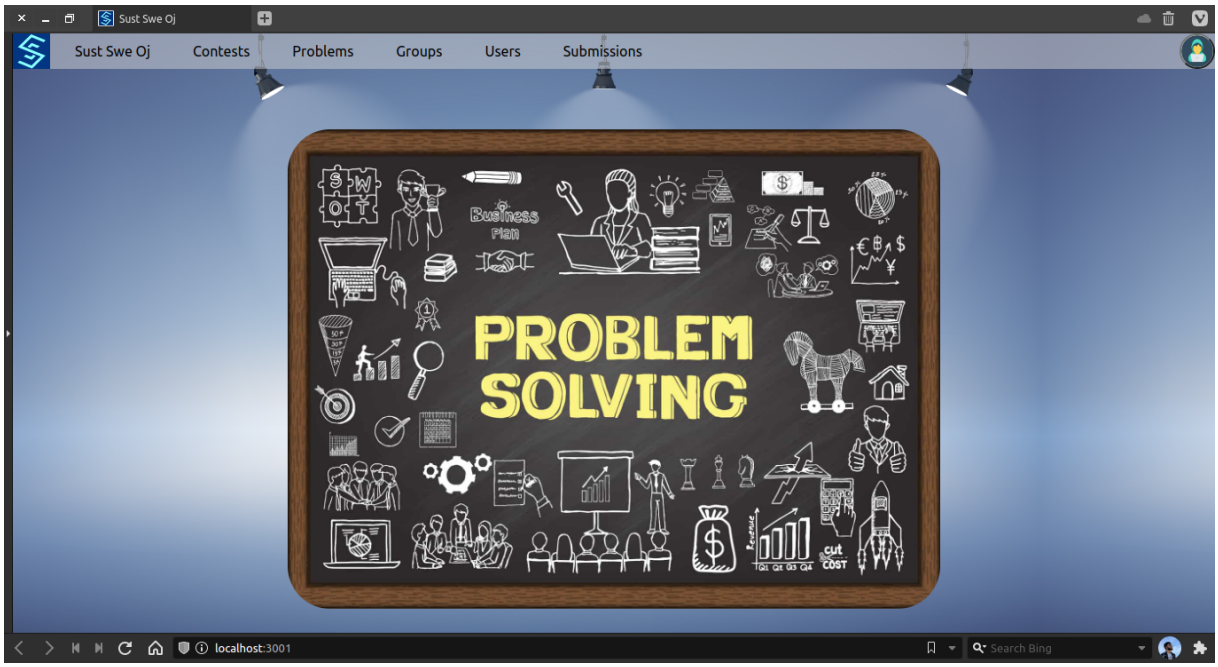

Figure 11: Main Home Page

Figure 12: Home page with Action Panel-1

Figure 13: Home page with Action Panel-2

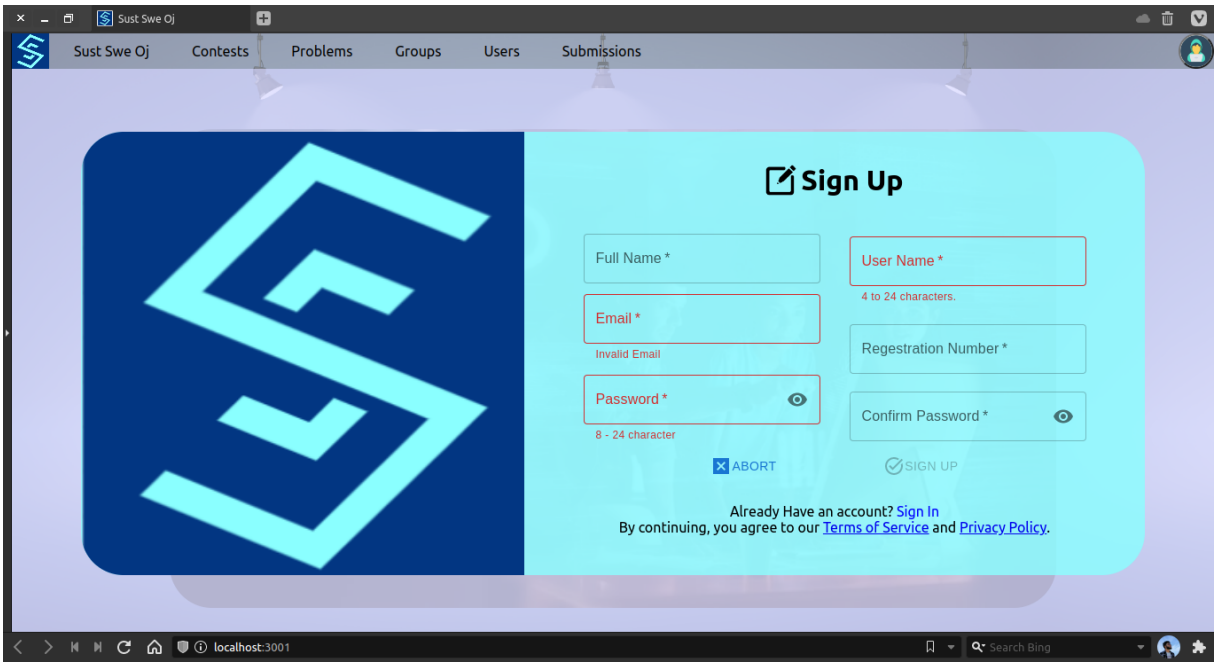

Figure 14: Sign Up page

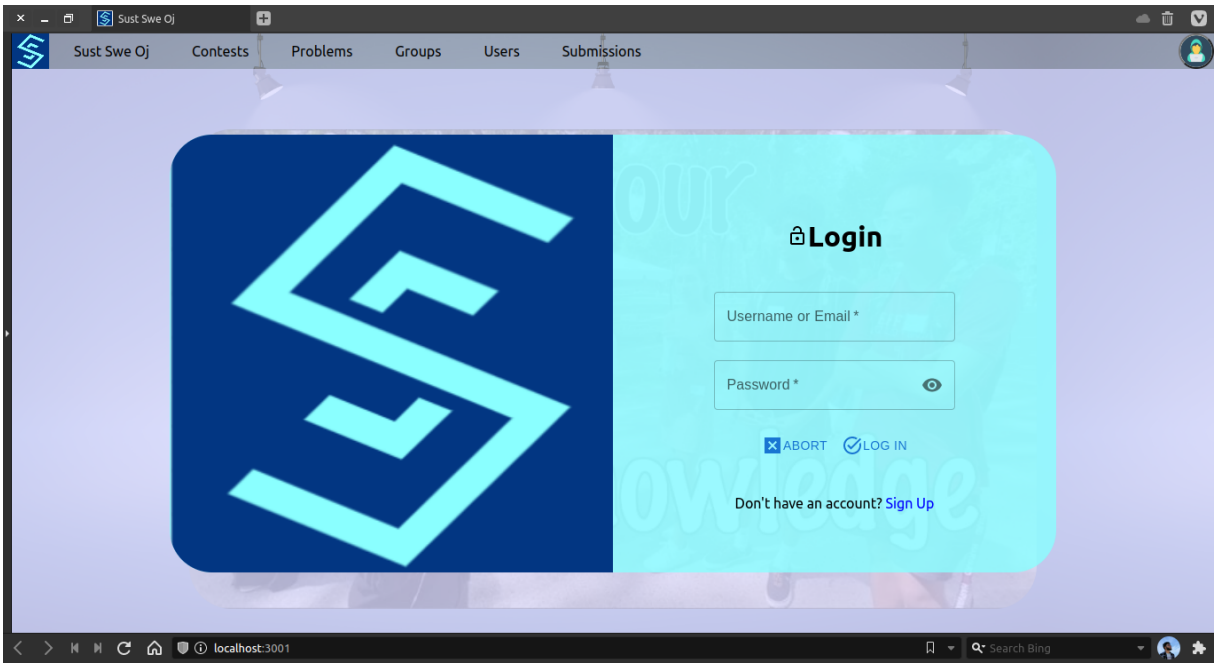

Figure 15: Login page

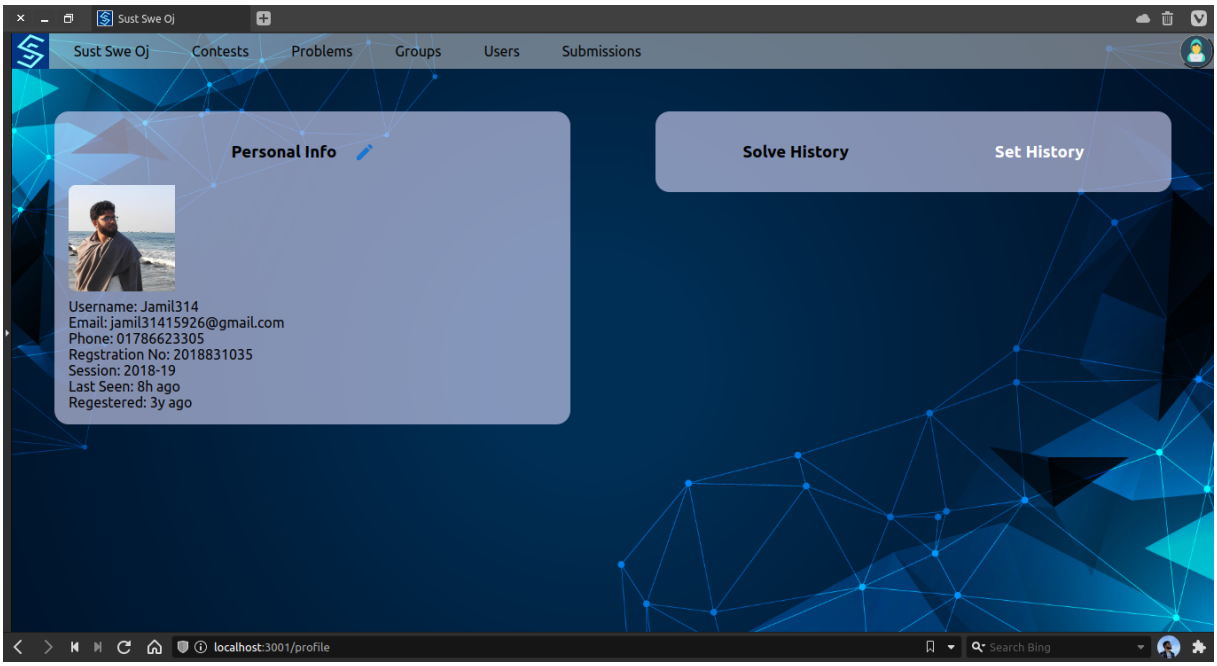

Figure 16: Profile page

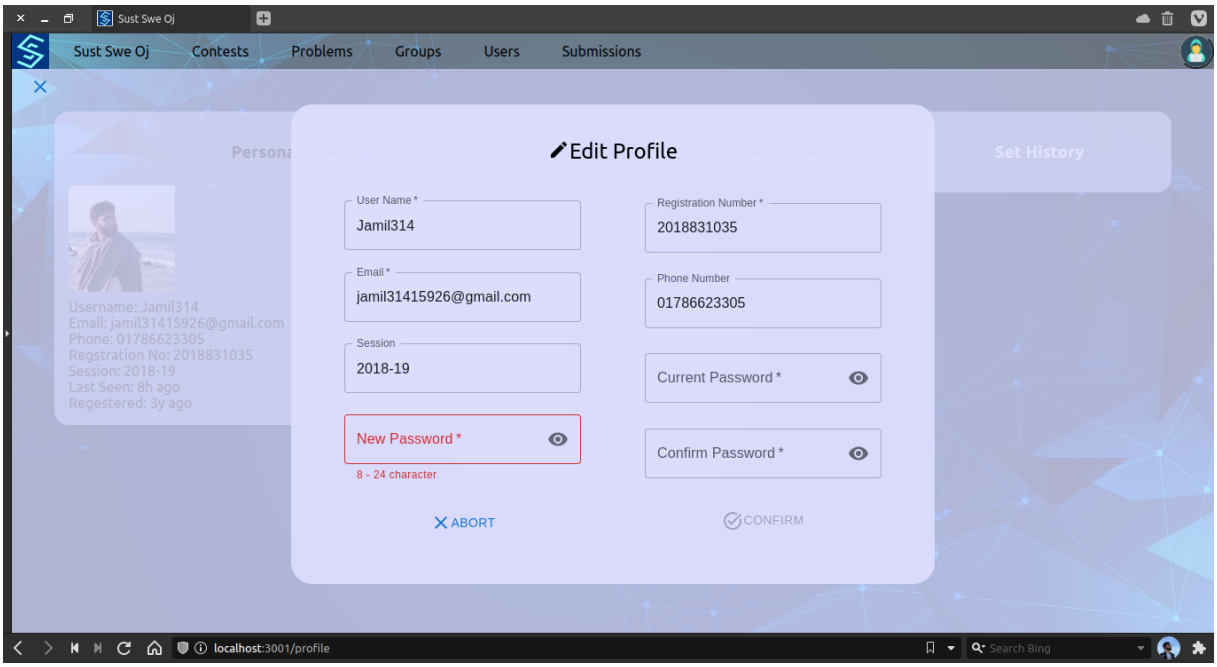

Figure 17: Profile Edit page

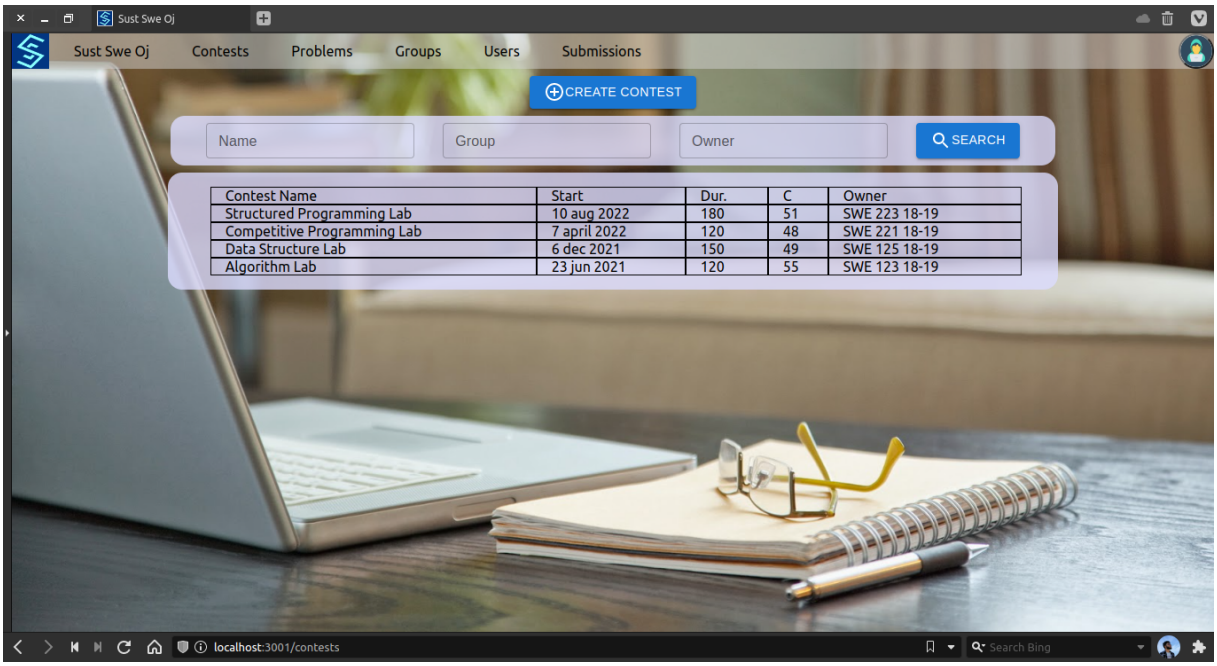

Figure 18: Contest Home page

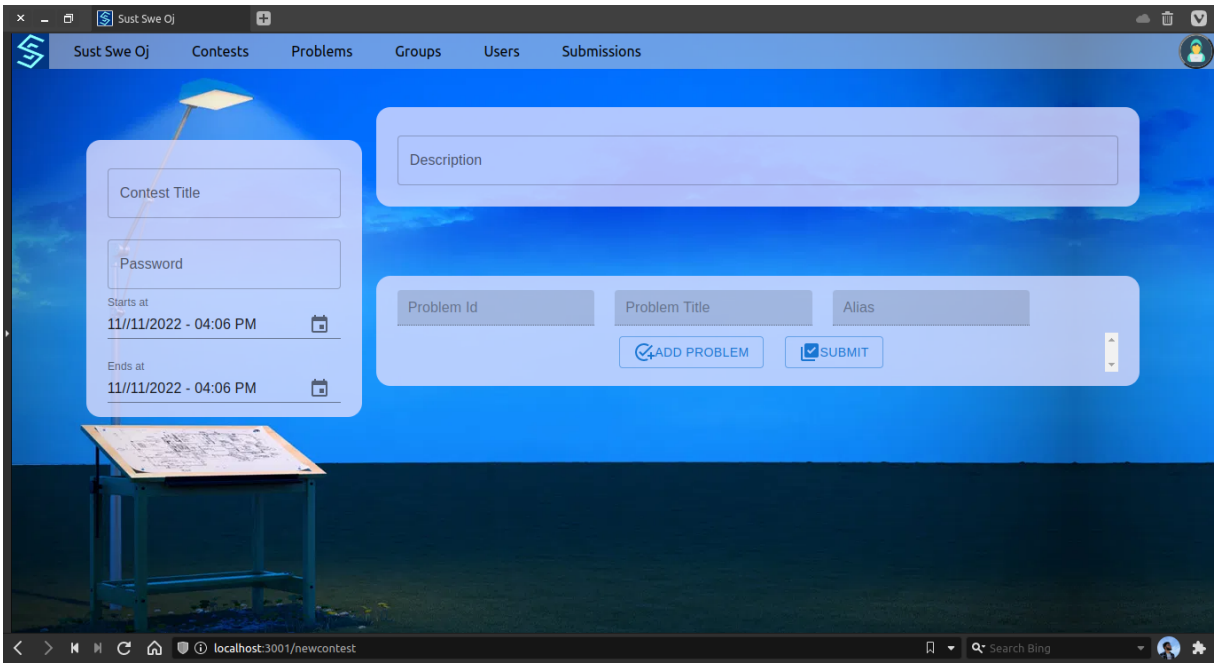

Figure 19: Contest Create page

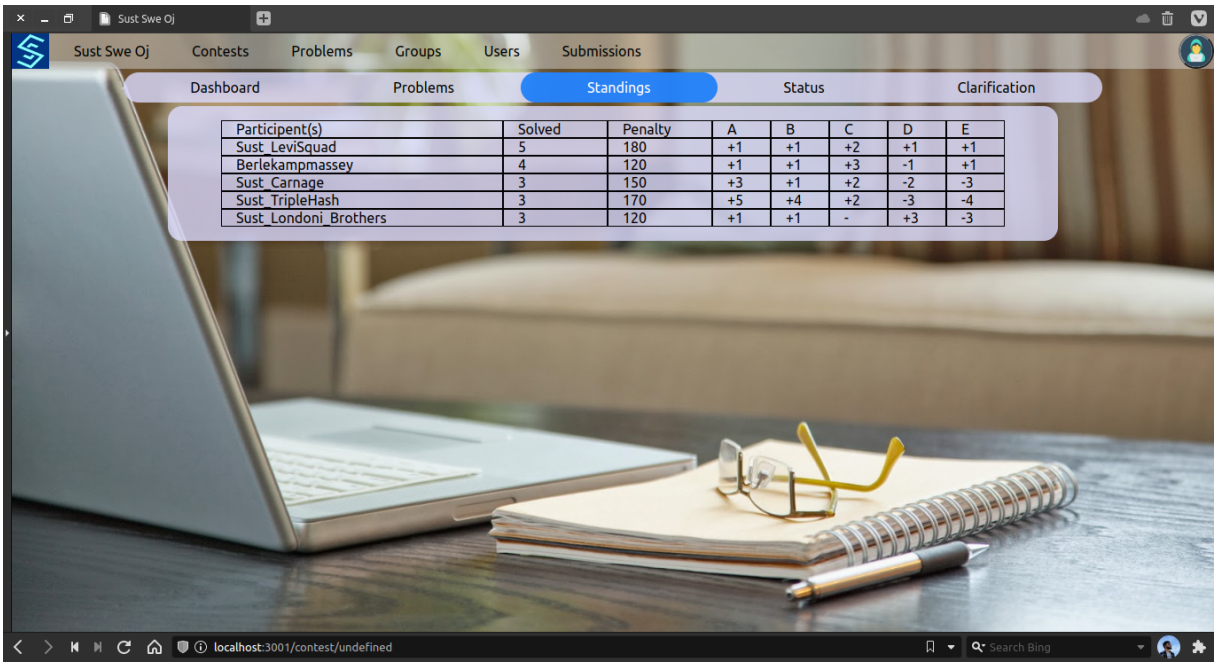

Figure 20: Contest Standing page

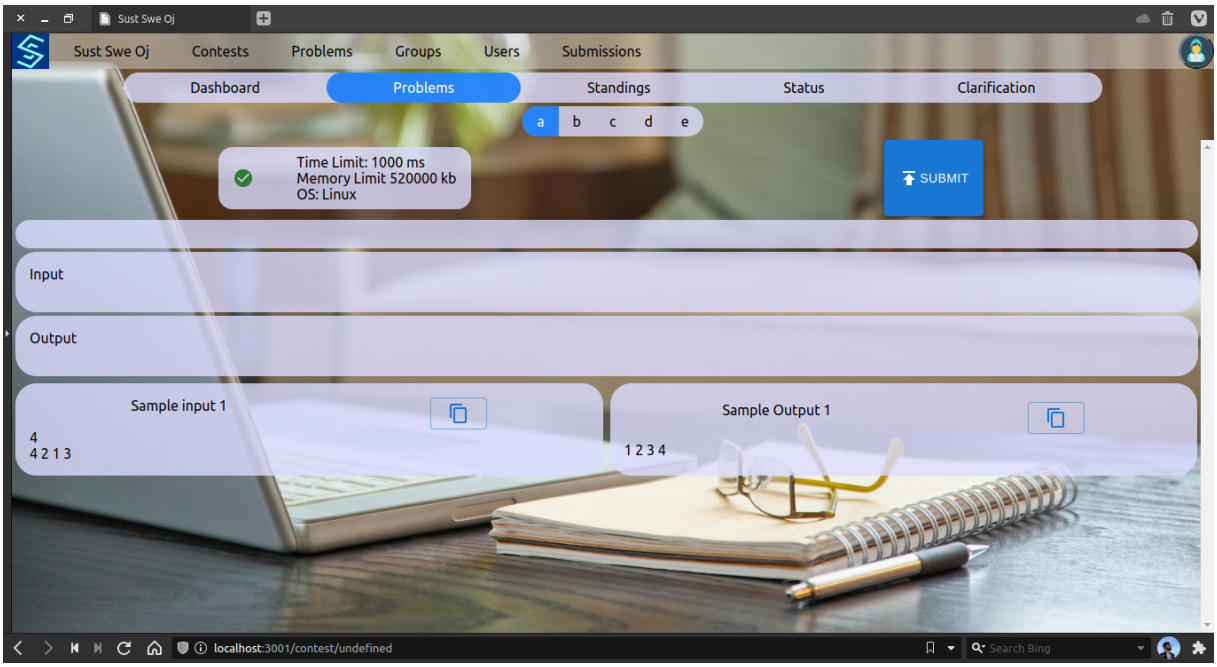

Figure 21: Contest Problem page

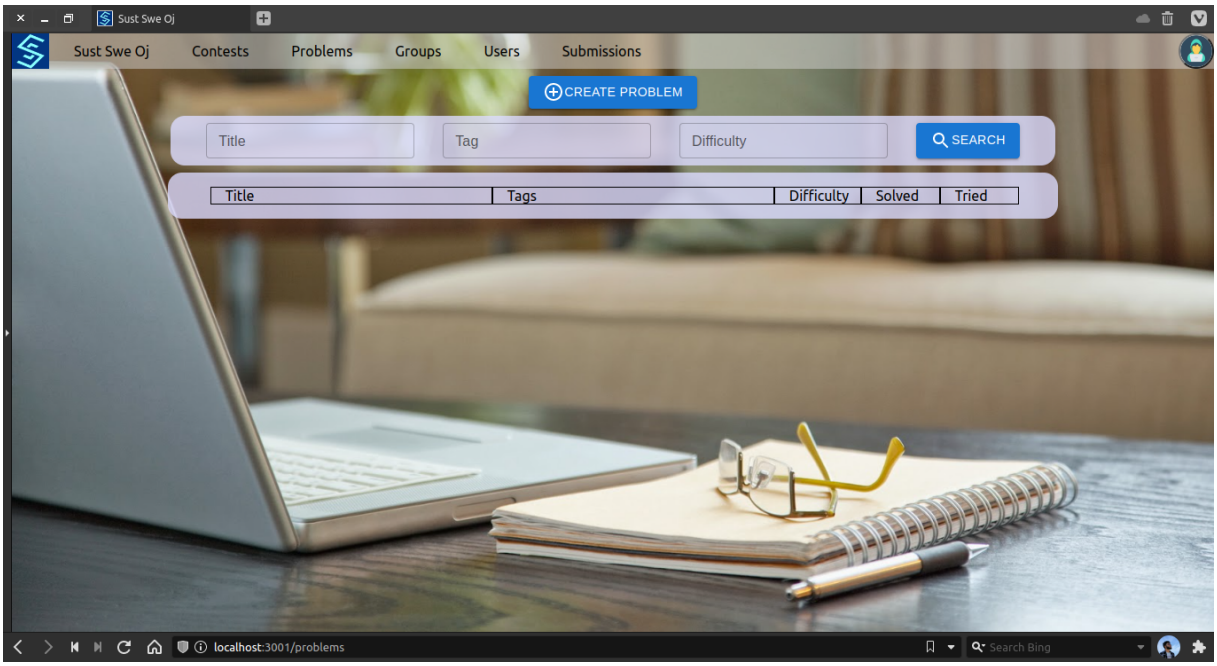

Figure 22: Create Problem page-1

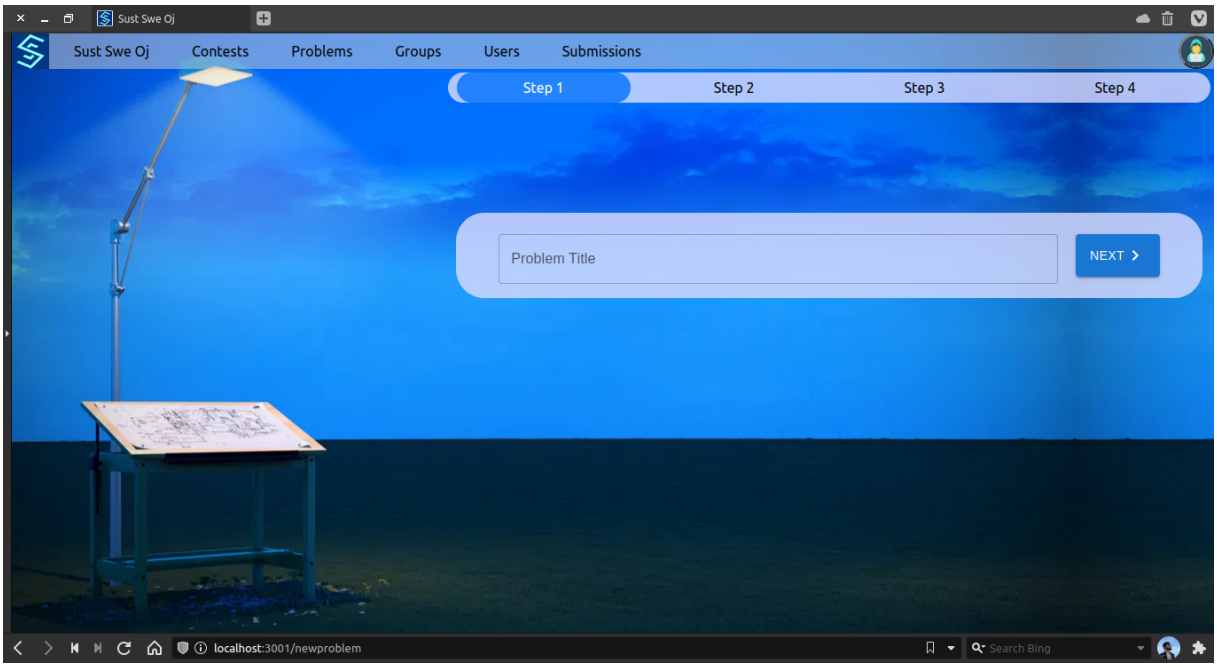

Figure 23: Create Problem page-2

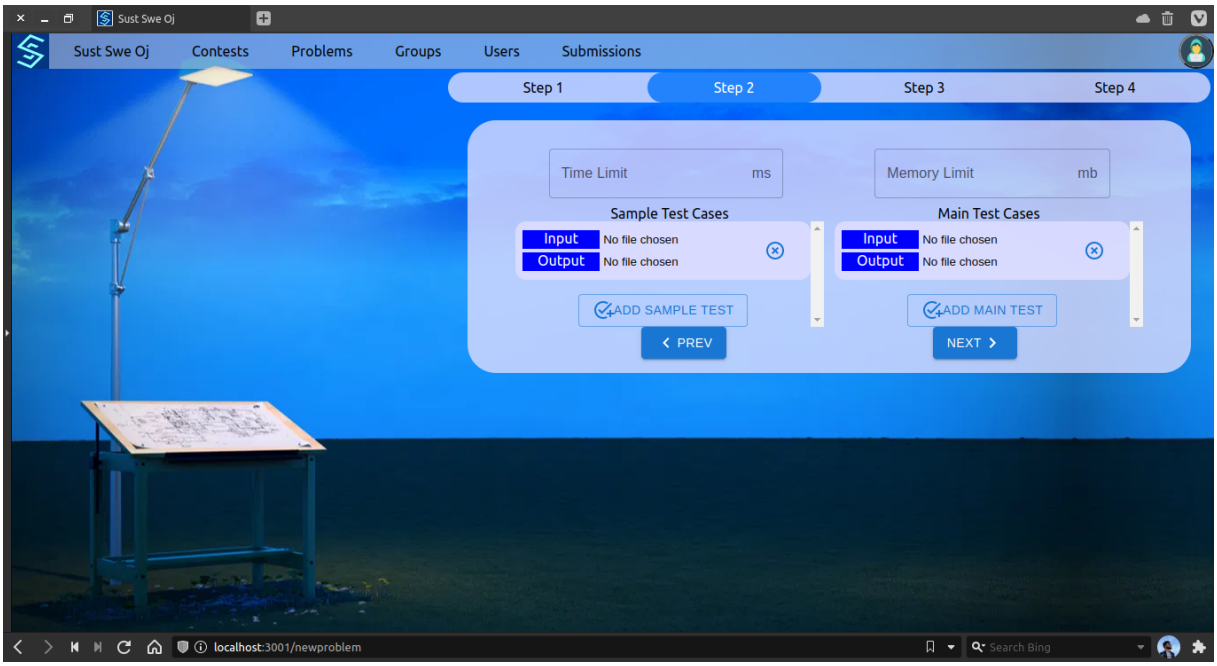

Figure 24: Create Problem page-3

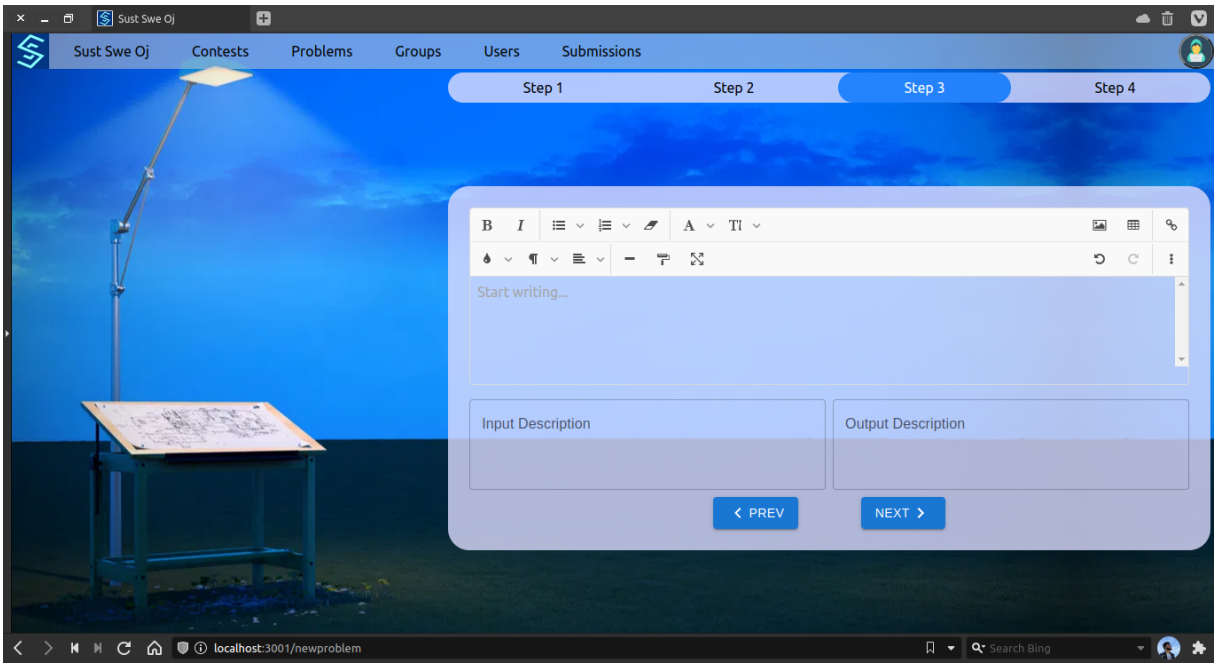

Figure 25: Create Problem page-4

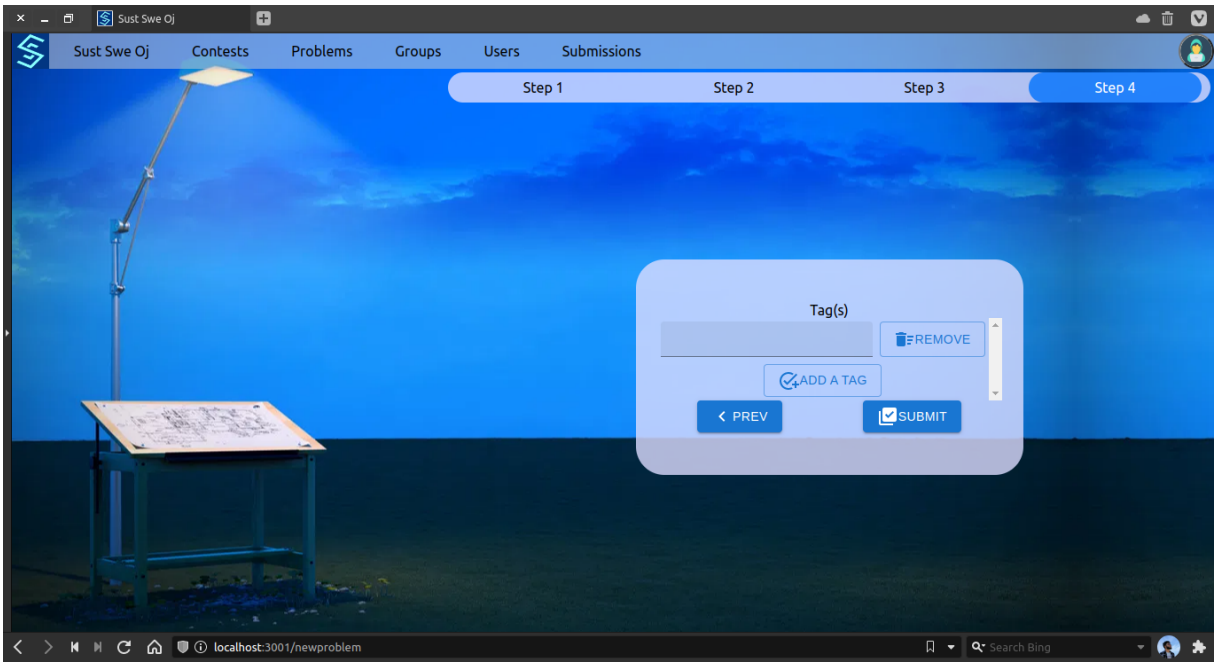

Figure 26: Create Problem page-5

7 Database Diagram

In our Database there are 6 tables with connection of one to many between a lot of them and each table has some unique columns and foreign keys to establish the connections between the tables.
The database or table diagram is added below for easier understanding about the tables and the attributes present in all the tables :

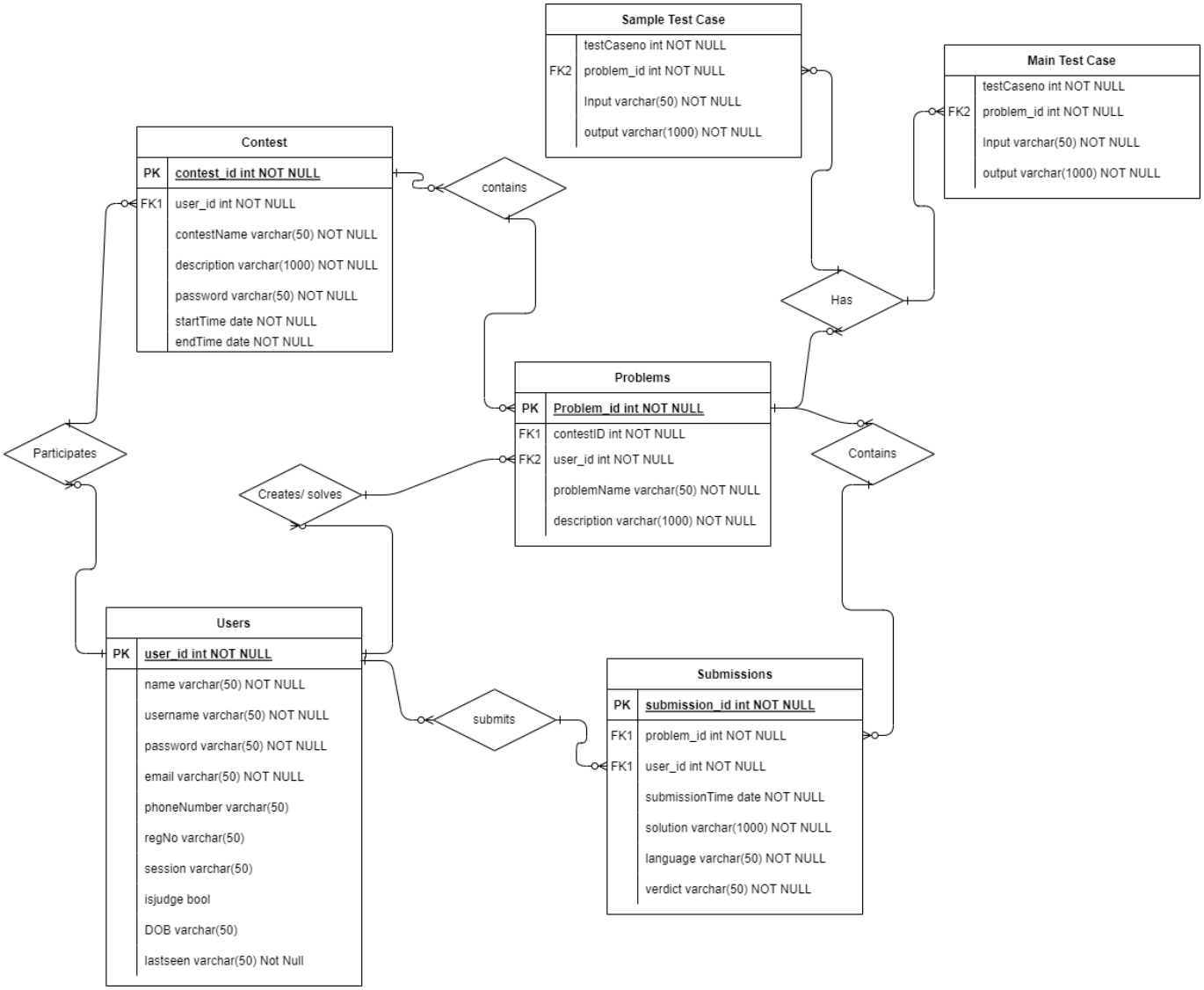

Figure 27: Database or Table Diagram of the System

8 Process Flow

The simple process flow diagram of our Online Judge system is given below for easier understanding:

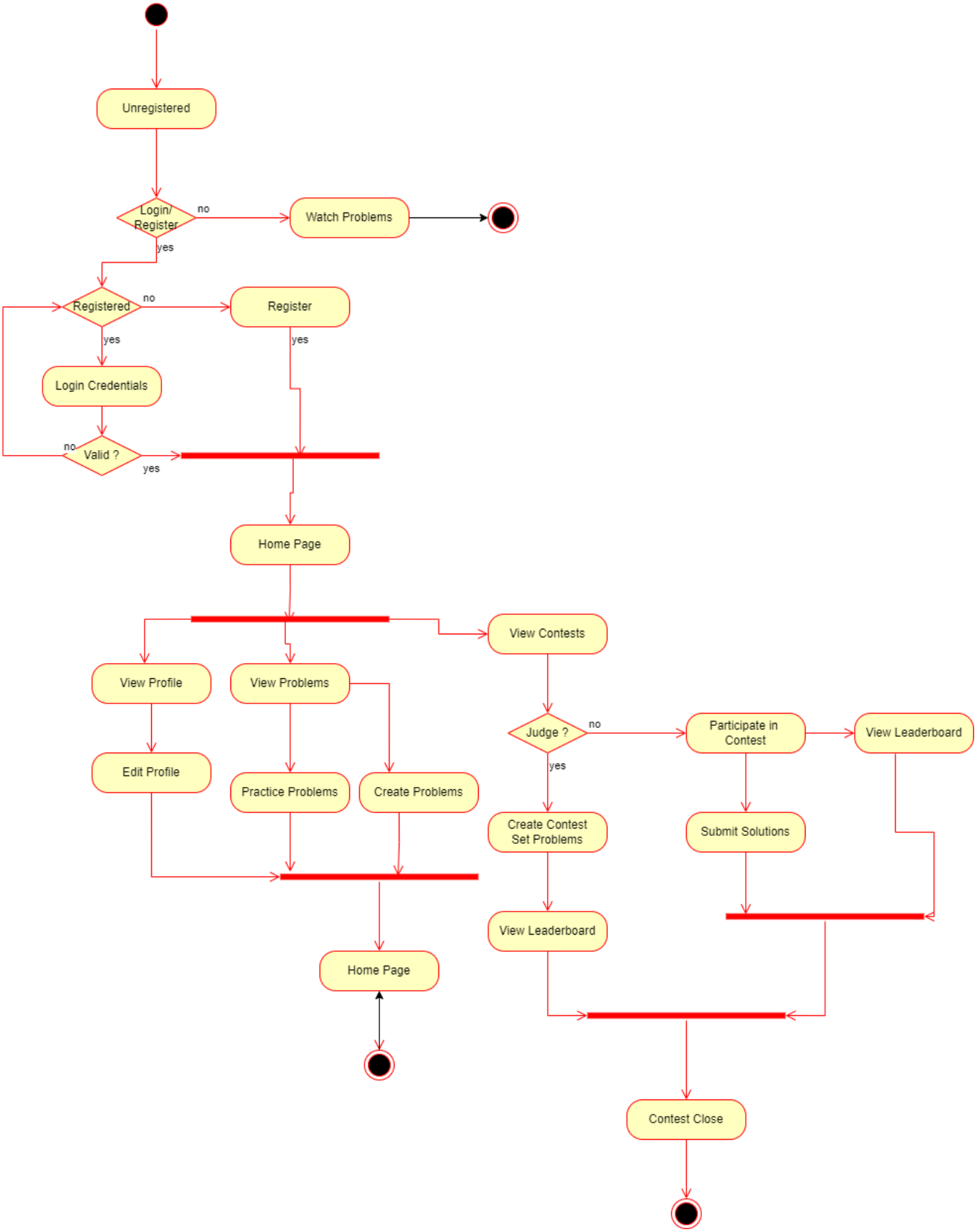

Figure 28: Process Flow Diagram of the System

9 Implementation

Our Project was implemented using some technologies and languages. The Languages are the following:

1 JavaScript

2 HTML

3 CSS

5 MySQL

We took help from some technologies while creating the Project.

1 NodeJs

2 Express

3 ReactJs

4 JSON Web Token

5 MaterialUI

6 Sequelize

The actual code implementation of our project is on Github to track our work progress.

10 Testing

We did unit test on the modules that were implemented. The tests included if our backend routes were sending requests and receiving them correctly while it also helped us to sync with our database correctly. The tool used for testing our routes and request response correctly is "Postman". Though we are yet to do testing with live users on the project that would be done when the whole system comes together and it is ready for the system testing phase.

11 Training and Maintenance

This portion contains our nonfunctional requirements and security assurance steps.

11.1 Non-Functional Requirements

A nonfunctional requirement is an attribute that dictates how a system operates. It makes applications or software run more efficiently and illustrates the system's quality. We tried our best to ensure some of the most important non-functional requirements like security, performance, reliability, data integrity and easier navigation for our users. Some of the non-functional requirements we want to highlight are listed below:

1 The User Interface was made easier to navigate from one page to another and the steps to complete tasks are simple for smooth user experience.

2 The load time for the website was kept at a minimum.

3 Account creations is mandatory to access our problem practice, problem creation and contest functionalities.

4 User account safety was met using strong password systems and hash functions while storing

5 User information change can be done only when the user is authenticated through the use of JSON Web Token

6 The contest creation and participation was also made secure with the use of passwords.

7 The system is easily accessible throughout the time our system is up and running without delay.

8 The data was handled carefully so that we can ensure data integrity and the data accuracy remains at its peak.