



BVB Campus, Vidyanagar, Hubballi – 580031, Karnataka, INDIA.

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Project report on

Codeolize

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

Name	USN
Pooja M Malagund	01FE18BCS142
Raksha Burli	01FE18BCS170
Rishabh Gautam	01FE18BCS172
Saloni Shah	01FE18BCS183
Akhilesh C Harti	01FE17BCS023

Under the guidance of

Lalita Madanbhavi

Pratiksha Benagi

School of Computer Science and Engineering

KLE Technological University, Hubballi

2021-2022



SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

2021 - 22

CERTIFICATE

This is to certify that the Senior Design Project entitled “**Codeolize**” is a bonafied work carried out by the student team Pooja M Malagund - 01FE18BCS142, Raksha Burli - 01FE18BCS170, Rishabh Gautam - 01FE18BCS172, Saloni Shah – 01FE18BCS183, Akhilesh C Harti – 01FE17BCS023, in partial fulfilment of the completion of 7th semester B.E. course during the year 2021 – 2022. The project report has been approved as it satisfies the academic requirement with respect to the project work prescribed for the above said course.

Guide

Lalitha Madanbhavi
Pratiksha Benagi

Head, SoCSE
Dr. Meena S. M

Viva -Voce:

Name of the Examiners

Signature with date

- 1.
- 2.

ABSTRACT

Technological disruption is visible almost everywhere, from smart TVs to technologically upgraded cooking equipment to online games, puzzles, and over-the-top (OTT) services. Much more is taking place right now, and much more is expected in the coming years. Much more is taking place right now, and much more is expected in the coming years.

The Coding Contests, Competitions, and Problem Set Gym allow you to evaluate your talents and proficiency while also comparing yourself to other programmers from across the world. Aside from that, the programmers that participate are provided a range of scholarships, prizes, jobs, and internships. While these platforms provide several advantages in terms of problem-solving ability, they do not bring together a community of developers. They don't provide coders a place to share their ideas or expertise with others who may benefit from the coders' network. Our initiative's main goal is to use social pressure to bring out a person's best coding skills. We concentrated on mixing elements of social networking sites with online coding websites; users code all the way through to gain their best coder caps, while auto-generated postings encourage other users to continue practising their coding skills, which benefits the users.

Keywords : *Online Compiler, Coding Platform, Social Networking*

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of a number of individuals whose professional guidance and encouragement helped us in the successful completion of this project work.

We take this opportunity to thank Dr. Ashok Shettar, Vice-chancellor and to Dr. N H Ayachit, Registrar, KLE Technological University, Hubballi.

We also take this opportunity to thank Dr. Meena S M, Professor and Head of Department, Department of Computer Science and Engineering for having provided us academic environment which nurtured our practical skills contributing to the success of our project.

We sincerely thank our guide Dr./Mr/Ms.(Guide Name) Designation, Department of Computer Science and Engineering for his guidance, inspiration and wholehearted co-operation during the course of completion.

We sincerely thank our project co-ordinators Dr. S. G. Totad for his support, inspiration and wholehearted co-operation during the course of completion.

Our gratitude will not be complete without thanking the Almighty God, our beloved parents, our seniors and our friends who have been a constant source of blessings and aspirations.

Pooja M Malagund - 01FE18BCS142

Raksha Burli - 01FE18BCS170

Rishabh Gautam - 01FE18BCS172

Saloni Shah - 01FE18BCS183

Akhilesh C Harti - 01FE17BCS023

CONTENTS

ABSTRACT	i
ACKNOWLEDGEMENT	i
CONTENTS	iii
LIST OF FIGURES	iv
1 INTRODUCTION	1
1.1 Motivation	1
1.2 Literature Survey	2
1.2.1 LeetCode	2
1.2.2 CodeForces	2
1.2.3 HackerRank	3
1.2.4 Codechef	3
1.2.5 Aspects of Online Coding Platforms	4
1.2.6 Understanding the error codes	5
1.3 Problem Statement	7
1.4 Problem Analysis	7
1.4.1 Design principle 1	7
1.4.2 Design principle 2	8
1.4.3 Scope and Constraints	8
1.5 Objectives	8
2 REQUIREMENT ANALYSIS	9
2.1 Functional Requirements	9
2.2 Non Functional Requirements	11
2.3 Hardware Requirements	11
2.4 Software Requirements	12
3 SYSTEM DESIGN	13
3.1 Architectural Framework	13
3.2 System Design	14
3.3 Database Design	15
3.4 Flow diagram of the Application	16
3.5 Design Principles	16

3.5.1	Design Principle 1 - APIs	16
3.5.2	Design Principles 2 - UI/UX	19
4	IMPLEMENTATION	22
5	RESULTS AND DISCUSSIONS	32
6	CONCLUSION AND FUTURE SCOPE OF THE WORK	33
	REFERENCES	35

LIST OF FIGURES

1.1	Comparison of existing systems	4
1.2	Block Diagram for Codeolize	8
3.1	Module Diagram	14
3.2	Overview of System design	14
3.3	Database Design for module 2, module 3 and module 4	15
3.4	Database Design for module 1 and module 5	15
3.5	Application flow	16
3.6	Input Parameters to Jdoodle API	17
3.7	Output Parameters for successful execution	17
3.8	Output Parameters when execution fails	17
3.9	Main Page - Sign Up	19
3.10	Main Page - Sign in	20
3.11	Home Page	20
3.12	Chat Dashboard	21
4.1	Login page	22
4.2	Sign-up page	23
4.3	Home page	23
4.4	Code Compiler page	24
4.5	An example of Python code, input and output	25
4.6	An example of C code, input and output	25
4.7	Problem sets page	26
4.8	Detailed view of problem (1)	27
4.9	Detailed view of problem (2)	27
4.10	User profile page	28
4.11	Likes, comments on a post	29
4.12	Followers list page	29
4.13	Following list page	30
4.14	Uploading new post	30
4.15	Search results page	31

Chapter 1

INTRODUCTION

From smart TVs to technologically improved kitchen equipment to online games, puzzles, and over-the-top (OTT) services, technological disruption is evident practically everywhere. Much more is happening now, and much more is predicted in the future years. From smart TVs to technologically improved kitchen equipment to online games, puzzles, and over-the-top (OTT) services, technological disruption is evident practically everywhere. Much more is happening now, and much more is predicted in the future years.

Online compilers are a sort of technology that allows you to build and run source code. Compilers run online in a variety of programming languages. They execute our code using a variety of compilers, including Dev C++, Visual Studio, Turbo C++, and others. But today it's so simple that all we have to do is connect to the internet, enter our code, and execute it. An online compiler performs the same functions as a traditional compiler.

The Coding Contests, Competitions, and Problem Set Gym allow you to assess your abilities and competency while also allowing you to compare yourself to other programmers from around the world. Aside from that, the participating programmers are offered a variety of scholarships, prizes, employment, and internships.

These platforms, while providing numerous benefits in terms of working on problem-solving abilities, do not bring together a community of coders. They don't provide a forum for coders to share their ideas or knowledge with others who may benefit from the coders' network.

1.1 Motivation

Coding is undoubtedly one of the most important talents to master for current and future generations. Programming helps young students develop problem-solving abilities, such as the ability to solve an issue in a rational as well as creative manner. Coding improves one's cognitive abilities by allowing them to think rationally, tactically, and analytically. Most significantly, coding is the way of the future, making it an immensely vital talent to have. Students who learn to code at an early age will have a plethora of career options accessible to them in the future.

Every programmer aspires to be the greatest! Yes, it is correct. But have you ever thought how someone may be regarded the best coder? So, let's say a person wants to be the best racer. Can he do this by only practising? No, all he needs to do is prove himself, defeat the

opposition, and win the war. Similarly, if you want to be the best programmer, you must overcome obstacles and demonstrate your programming abilities. Competitive programming and coding competitions are the greatest approach to do so.

Coding competitions or contests allow you to assess your abilities and competency while also allowing you to compare yourself to other programmers from around the world.

Students may be influenced by peer pressure to do or say things they would not ordinarily do or say. It isn't always a terrible thing: peer pressure to study more or stand up to bullying can result in great outcomes. It leads to a stronger sense of belonging and support, as well as a rise in self-confidence, the introduction of good hobbies and interests, and the reinforcing of positive behaviours and attitudes.

The main point of our initiative is to use peer pressure to bring out a person's greatest coding abilities. We focused on combining parts of social networking sites with online coding websites; users code all the way through to earn their best coder caps, while postings auto-generated keep other users inspired to continue coding practise, benefiting the users.

1.2 Literature Survey

1.2.1 LeetCode

LeetCode is a renowned platform that provides a variety of coding tasks to help you improve your coding abilities. It hosts a variety of programming contests for participants on a weekly and biweekly basis.

LeetCode's questions accept many programming languages, allowing you to complete them in the languages you are most familiar with and avoid needless and time-consuming learning. Aside from coding, one of LeetCode's most valued advantages is its ability to host community debates. The community is the most often mentioned benefit among users, and it serves as a valuable resource for learning from others' errors and getting insight into how they address difficulties.

LeetCode is designed for software developers who want to improve their abilities and practise answering technical questions. By mastering the questions at each level, users may prepare for technical interviews and keep their skills strong. LeetCode also provides a big number of solutions with explanations for each step, so even less experienced software developers may profit from it.

1.2.2 CodeForces

ITMO University is in charge of it, and it hosts up to two short contests every week. During the "hacking phase," it allows us to verify the validity of other contenders' solutions. Codeforces is

one of the top venues for competitive coding, and it's best known for hosting brief challenges and tournaments in which programmers from all over the world compete. Here you may practise problems ranging from the very beginning to the most sophisticated.

Contests are held often in Codeforces. Every week, there are two to three games, each lasting about two to three hours. Some contests are also open to you based on your standings. If you're a newbie, you can start with Division 2 and Division 3 contests. Your ranking will rise or fall depending on how well you answer tasks in each contest and how long it takes you to complete them. The lesser time you take for each problem, the more will be your rating.

1.2.3 HackerRank

HackerRank lets you tackle programming tasks in a variety of languages, including C, Java, Python, and Ruby. Different fields, such as algorithms, machine learning, and artificial intelligence, can also be used to address issues.

HackerRank is a website where programmers from all over the globe get together to practise different programming paradigms such as functional programming and solve issues in a variety of Computer Science fields such as machine learning, artificial intelligence, and algorithms. HackerRank is a pre-employment testing tool for organisations of all sizes that uses a single dashboard to plan, source, screen, interview, and recruit competent applicants.

To establish remote hiring operations, HackerRank delivers virtual whiteboards and customisable IDE environments for evaluating and grading diverse candidates based on their designing talents and strengths. Managers may also use the platform to build test cutoff scores and a list of selected students from career sites to make on-campus visits go more smoothly.

HackerRank also has practise questions for software engineers who wish to prepare for technical interviews in environments that are comparable to those in which they will be assessed. People who want to acquire new skills or languages might benefit from the programme. Each question is categorised according to the level of difficulty in algorithms, data structures, languages, and general math. HackerRank also has contests for users to enter, as well as a job board for prospective software developers. Their programming challenges include a wide range of computer science areas and use a variety of programming languages (including SQL, JavaScript, C++, PHP, Java, and Python).

1.2.4 Codechef

Directi, an Indian software startup, has launched CodeChef, a non-profit teaching effort. It's a worldwide programming community built on top of the world's largest competitive programming platform that encourages learning and friendly competition. It is run by Unacademy

and hosts a 10-day contest, as well as an IOI-styled lunchtime contest and an ICPC-styled Cook-Off event. It also offers educational organisations a free contest hosting platform.

Leetcode	Codeforces	HackerRank	Codechef
Frontend: HTML, Javascript, jQuery	Frontend: Clipboard.js, moment.js, jQuery	Frontend: Backbone.js, bootstrap	Frontend: HTML, Javascript, jQuery
Backend: Python, Django framework, Angular JS	Backend: Java	Backend: Ruby on Rails	Backend: Drupal, Node.js
CDN: Cloudflare	Storage: Amazon S3	Storage: Amazon S3	Storage: Amazon S3
Video Player: Vimeo			Video Player: YouTube

Figure 1.1: Comparison of existing systems

1.2.5 Aspects of Online Coding Platforms

How code evaluation works on online judges?

Let's have a look at how online judges assess your code. If you understand how problems function, you will be able to address them more effectively and quickly.

Behind the scenes, each challenge includes a collection of input and intended output files. Let's say I₁, I₂,... Input files are in, and output files are in EO₁, EO₂,... EO_n.

Your code is executed against all of these input files when you submit it. This produces a list of created output files with names like GO₁, GO₂,... GO_n. The created output files are then compared to the intended output files. If they match, your answer is considered accurate.

However, a lot of things may go wrong along the road. Before running compiled language code, it must first be compiled. The compilation stage of your code might fail.

Your solution must function under certain time and memory limits. The execution is halted if it violates such limitations.

At run-time, your code may meet problems and break. The judge will produce an error message in all of these circumstances, which you may use to correct your code.

Why should you test your solution before submitting?

Most of the time, you won't need to submit your code to find out whether it has problems. The majority of judges include a test button that you can use to, well, test your code. This is frequently accompanied with the ability to supply custom input. Your code is built and then executed against this specific input when you "test" it. This may be used to check for compilation issues in your code. The output created by your function is also returned by

the judge. You may manually check if your code produces the outcome you anticipate by experimenting with different input values.

It's a good idea to test and double-check your code before submitting it. For incorrect contributions, several platforms levy fines. Your accuracy suffers as a result. Penalties have an impact on your score and standing in contests.

Understanding the problem statement

When you open up a problem on a programming (problem solving) website, you are likely to encounter the following components:

- **Problem Statement** This is a description of the problem. You are supposed to write code that solves this problem.
- **Input and Output Format** This defines how your code should take input and what the format of the output generated by your code should be. You need to ensure that you follow this format strictly, otherwise your solution won't be accepted even if its correct logically.
- **Sample Input and Expected Output** In order to assist you in understanding what the problem expects of you, you will often find examples in the form of sample input and expected output. Use this as an example for input output format as well.
- **Constraints** These are the constraints (rules) that apply to the input values. This will help you in designing solutions which are optimal for these size of inputs. These are guaranteed by the the problem setter and you don't need to check for them explicitly in your code.
- **Limits** These are the time and memory limit that your code must abide by. These are the resources allocated for your code to run. If your code exceeds these limitations, you will encounter an error. If your code encounters an issue, your output does not match expected result or you exceed limits defined, you will encounter an error on submission.

1.2.6 Understanding the error codes

It's quite common to face errors when you test or submit your code on online judges. It can be challenging to understand what is wrong with your code and fix it. However online judges usually return an error code or a message which can help you understand more about what caused the failure.

Below are some of the most common errors seen on online judges, what they mean, and some of the common reasons why they occur.

Compilation Error (CE):

Your program did not get compiled successfully. Try to understand the issue based on the Compilation Error Message. Common reasons for CE:

- Syntax Error
- Missing Imports
- Using restricted functionalities

Wrong Answer (WA):

Your program ran successfully but returned a different output than what was expected. Common reasons for WA:

- Incorrect interpretation of problem.
- Incorrect solution.
- Bug in the code.
- Edge cases. Program is tested on cases not mentioned in the problem statement. Make sure that you have identified and tested your code with multiple edge cases.
- Incorrect output format (Different from format mentioned in problem)

Time Limit Exceeded (TLE):

Your program did not complete execution in the allotted time. Your program gets a predefined time limit for every test case. If it takes more time, it is killed. TLE does not guarantee that your program is correct. Common reasons for TLE:

- Solution is not optimal.
- Infinite Loop.

Memory Limit Exceeded (MLE):

Your program tried to allocate more memory than it was allowed to. Common reasons for MLE:

- Declaring large arrays or lists.
- Adding a lot of data.
- Stack Overflow Error

Runtime Error (SIGSEGV)

: Your program tried to access or write to a memory that it cannot access or is invalid. Also known as Segmentation Fault. Common reasons for SIGSEGV:

- Accessing array/string index outside its range.
- Using too much memory in some languages.
- Uninitialized or incorrectly initialized pointers.

Runtime Error (SIGFPE):

Your program encountered a floating-point error. Generally caused if you do an invalid math operation. Common reasons for SIGSEGV:

- Division by zero.
- Square root/Log of negative numbers.

Runtime Error (SIGABRT):

Your program aborted the program due to fatal error. Common reasons for SIGABRT:

- Using assert or abort in the code.

Runtime Error (NZEC/Non-zero error code):

Your program did not return a zero-error code from the main method and did not fall into any of the above buckets. Common reasons for NZEC:

- Not returning 0 from main method.
- Not catching exceptions.

1.3 Problem Statement

Design and develop a web application that offers an online IDE to compile code, a problem solving gym, code storage unit, and also provide a social networking platform for coders.

1.4 Problem Analysis

1.4.1 Design principle 1

UI/UX - Design well structured UI and UX designs for the complete application.

1.4.2 Design principle 2

APIS - Jdoodle API will be used to compile the codes.

1.4.3 Scope and Constraints

In order to implement a Coding Platform from scratch, our major challenge is to get a dataset containing the question and its test cases.

- **Solution 1: Web Scraping the Problem sets**

We can web scrap the data from any existing platforms like leetcode, codechef, codeforces, etc. But it is very difficult to scrap the test cases from online platforms as they are hidden.

- **Solution 2: Manually Designing Problem sets**

We can manually design the problem statements and their test cases and feed it to the database. We can limit ourselves to making some countable problems (15-20).

1.5 Objectives

1. Develop an Online Integrated Development Environment (IDE) for executing codes in languages like C, C++, Java, Python.
2. Provide users with a problem gym and allow them to build problem solving skills.
3. Provide an integrated social networking platform for the users.

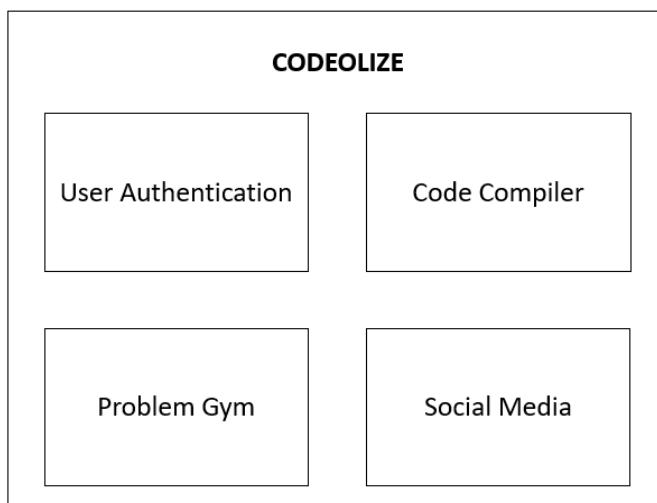


Figure 1.2: Block Diagram for Codeolize

Chapter 2

REQUIREMENT ANALYSIS

The project's Software Requirement Specification (SRS) specifies the web application Codeolize's needs and specifications. The project's functional and non-functional requirements are displayed in this SRS document. It also includes the system's use case diagrams for a better understanding of the process.

2.1 Functional Requirements

Functional requirements (FR) determine the Codeolize web application's core system behaviour. It specifies the user's interaction with the application interface. The specifics for achieving the functional requirements are included in the system design for the Codeolize web application.

Module 1: User Authentication

- The system should allow new users to create their account.
- The system should allow users to login to their account.
- The system should allow users to reset their password.
- The system should allow users to logout from the application.
- The system should allow users to report any issue.

Module 2: Code Compiler

- The system must allow users to open a code editor.
- The system should allow users to code in the following languages: C, C++, Java, Python.
- The system should allow users to give custom input for the code.
- The system should allow users to compile and test their code.
- The system should output the status of code and the required output.

Module 3: Problem Gym

- The system should show users the list of available questions to practice.
- The system should also show users the stats of a particular question.
- The system should allow users to submit the solution for a particular question.
- The system should allow users to filter the question based on difficulty rating.
- The system should allow users to filter the question based on tags.
- The system should allow users to sort the question on the basis of accuracy.

Module 4: Social Media

1. Feed

- The user shall be able to view the post.
- The user shall be able to like the post.
- The user shall be able to unlike the post.
- The user shall be able to comment on the post.
- The user shall be able to share the post.

2. Social

- The user shall be able to follow/unfollow other users.
- The user shall be able to see the followers and following list of other users.
- The user shall be able to search other users with their username.

3. Profile

- The user shall be able to change his profile picture.
- The user shall be able to change his profile name.
- The user shall be able to change his password.
- The user shall be able to edit his posts.
- The user shall be able to see his statistics of submission.
- The user shall be able to see his submission history.

4. Messaging

- The user shall be able to send/receive text messages.

-
- The user shall be able to send/receive multimedia messages.
 - The user shall be able to see whether his friend is online or offline.
 - The user shall be able to see the last seen of the friend.
 - The user shall be able to see the time and date of the messages.

2.2 Non Functional Requirements

A non-functional requirement (NFR) is a standard or benchmark that is used to assess the overall performance of the Codeolize web application, rather than specific behaviours. The system design shows how these non-functional criteria will be implemented in detail.

- Application must handle at least 200 requests at a time.
- The response time for viewing a submission status must be less than 10 seconds.
- The leaderboard or the list of submissions must be updated for every 15 minutes.
- The site should load in 3 seconds when the number of simultaneous users are > 200 .
- System generated posts should be posted with a latency of no greater than 2 hours.
- User cannot login into the application from more than one device at any given point of time.

2.3 Hardware Requirements

The following requirements are for developing the application.

- Processor: Intel Core i5
- RAM: 8GB
- Graphics: 2GB
- Hard Disk: atleast 16GB
- Keyboard, Mouse, Monitor

2.4 Software Requirements

The following requirements are for developing the application.

- OS: Windows 10
- Editor: VS Code
- Programming Language: Python (3.7.6)
- Web Framework: HTML, CSS, Django, Javascript
- API: Jdoodle (Compiler API), Sphere Engine Problems API

Chapter 3

SYSTEM DESIGN

The process of defining the aspects of a system, such as the architecture, modules, and components, as well as the many interfaces between those components and the data that flows through it, is known as system design. It's designed to meet a company's or organization's particular goals and requirements by constructing a well-functioning system.

A systematic approach to the design of a system is referred to as systems design. It can be done from the bottom up or from the top down, but either way, the process is systematic in that it considers all related variables of the system that needs to be built—from the architecture to the required hardware and software, all the way down to the data and how it travels and transforms throughout its journey through the system. The terms "systems design," "systems analysis," "systems engineering," and "systems architecture" are all used interchangeably.

3.1 Architectural Framework

"The process of specifying a collection of hardware and software components and their interfaces to provide the foundation for the creation of a computer system," according to the IEEE. Software designed for computer-based systems can take on any of these architectural styles. Each style shall specify a system category that includes the following elements:

- A collection of components (for example, a database, computational modules) that will execute a system-required function.
- The set of connections will aid in component coordination, communication, and collaboration.
- The conditions under which components can be combined to form a system.
- Semantic models that aid the designer in comprehending the system's overall qualities.

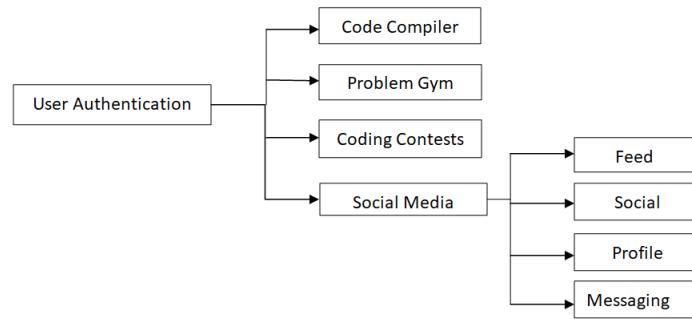


Figure 3.1: Module Diagram

3.2 System Design

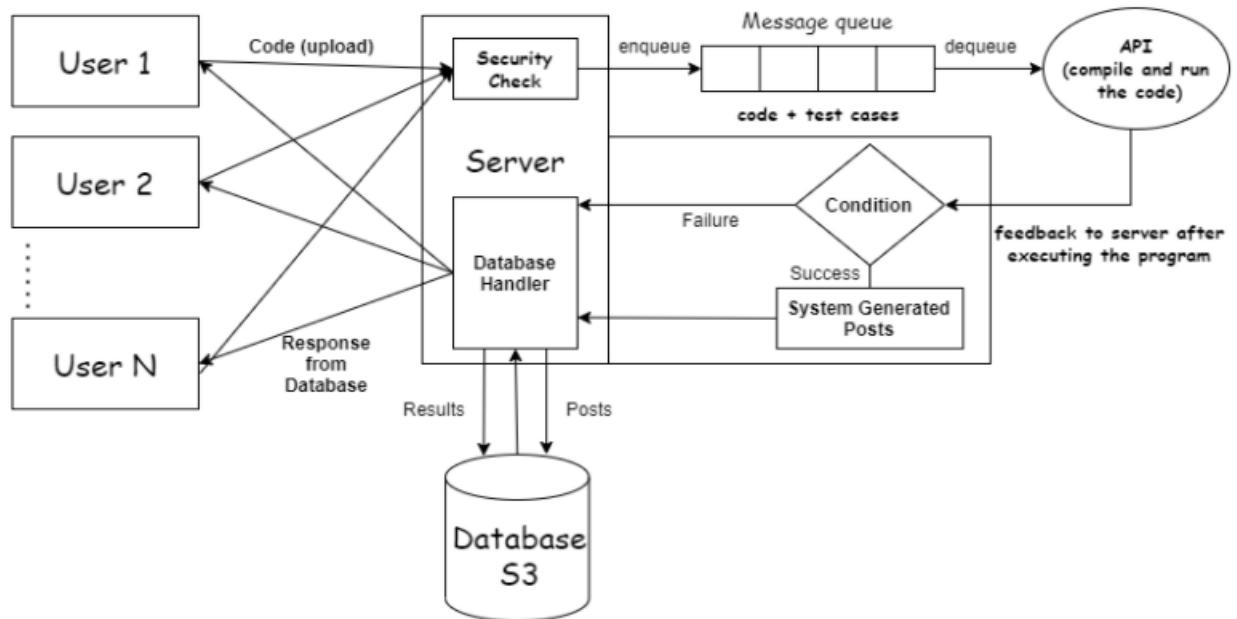


Figure 3.2: Overview of System design

3.3 Database Design

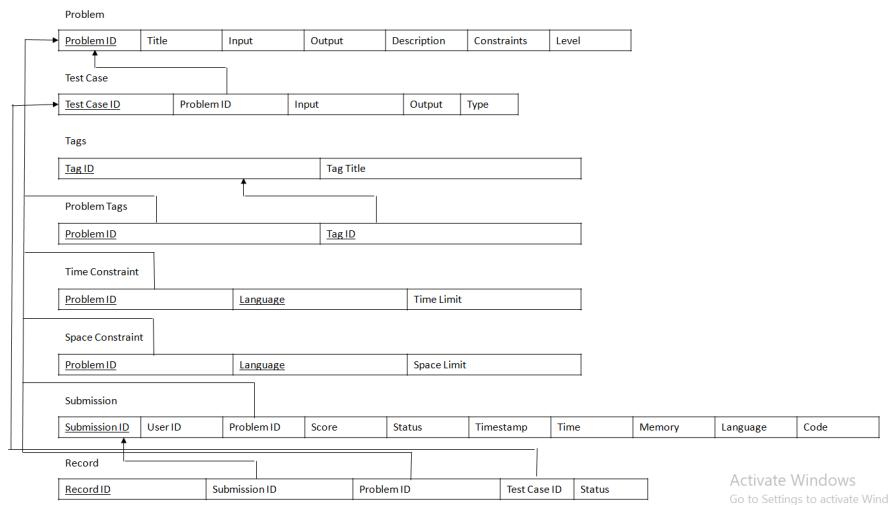


Figure 3.3: Database Design for module 2, module 3 and module 4

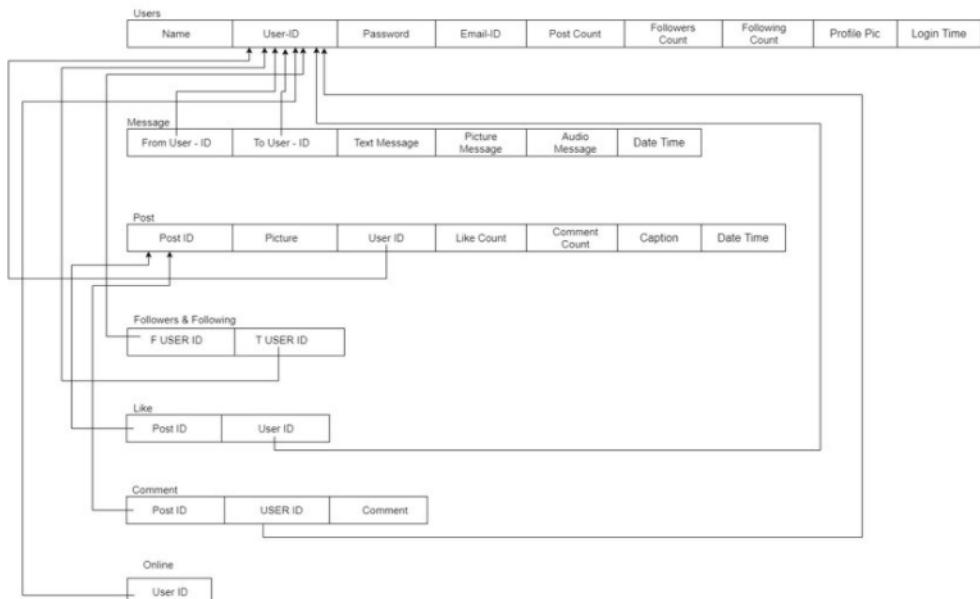


Figure 3.4: Database Design for module 1 and module 5

3.4 Flow diagram of the Application

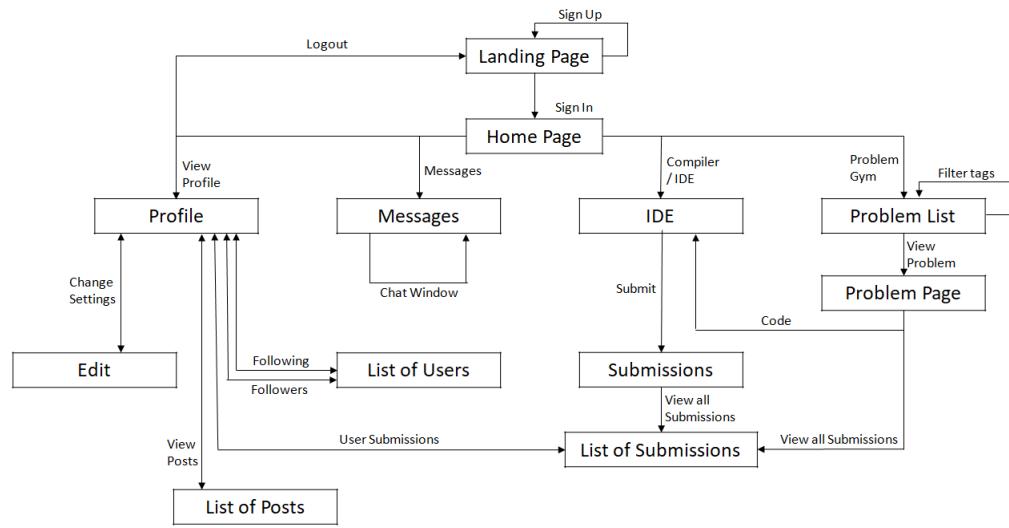


Figure 3.5: Application flow

3.5 Design Principles

Design principles are key bits of advice that will help you create designs that are simple to use and enjoyable to look at. When you pick, create, and arrange pieces and features in your work, you use them. The cumulative expertise of scholars and practitioners in design and related topics is represented through design principles.

3.5.1 Design Principle 1 - APIs

JDoodle Compiler is an online API service to compile and execute Programs online via APIs, it supports Java, C/C++, PHP, Perl, Python, Ruby and many more languages. It is a JSON based API.

Parameter	Description
clientId	Client Id for subscription
clientSecret	Client Secret for subscription
script	Program to compile and execute
stdin	Stdin
language	Language of the script
versionIndex	Version of the language to be used

Figure 3.6: Input Parameters to Jdoodle API

Parameter	Description
error	Error message
statusCode	Status Code of the result

Figure 3.7: Output Parameters for successful execution

Parameter	Description
output	Output of the program
statusCode	Status Code of the result
memory	Memory used by the program
cpu Time	CPU time used by the program

Figure 3.8: Output Parameters when execution fails

The Sphere Engine Problems API allows you to create and utilise programming problems for automated programming talent verification in processes like recruiting, education, e-learning, onboarding, and training. When generating a programming issue, you may include the following information:

- description of the problem,
- assessment type (e.g. maximization, minimization),

-
- test cases (i.e. data used to test solutions), consisting of:
 - input data,
 - reference output data,
 - judge program (the component responsible for determining the correctness of the test case solution),
 - master judge program (the component responsible for the final evaluation of the correctness of the solution).

When submitting a solution to a problem we can specify:

- the source code of the program,
- the programming language.

Sphere Engine Problems API offers the ability to execute programs in dozens of programming languages, including C++, C, Go, Haskell, Java, Kotlin, Node.js, PHP, Python, Ruby, Scala, or Swift among others (full list of supported programming languages).

Data available after the process of evaluating the solution includes:

- the verdict regarding the correctness of the solution,
- memory consumption,
- score (in the case of optimization problems or problems assessed partially),
- data streams containing:
 - generated output data,
 - runtime error,
 - warnings and compilation errors,
 - additional information returned by the judge programs.

If a submission meets all the requirements of your challenge, it will show up as "accepted" in the system. Otherwise, you can check what the reasons were for its failure.

There are five possible statuses of a submission:

- Accepted - the program is executed within the required time limit and the answer is consistent with the pre-defined solution.
- Wrong answer - the program is executed correctly, but the answer is not consistent with the pre-defined solution.

- Time limit exceeded - the execution time exceeds the imposed time limit. Possible causes of this status include bugs in the submitted code, such as an unforeseen infinite loop, or a solution which is far from optimal and runs too slowly.
- Runtime error - an error appears during execution, which causes the process to stop. One of the reasons could be a memory or stack overflow error in a programming language such as C.
- Compilation error - an error which causes the submitted source code to not compile correctly.

3.5.2 Design Principles 2 - UI/UX

The process of designing user interfaces in software or digital devices with an emphasis on appearance or style is known as user interface (UI) design. Designers strive to design user interfaces that are simple to use and enjoyable to use. Graphic user interfaces and various types of user interface design are referred to as UI design.

The process through which design teams build products that give consumers with meaningful and relevant experiences is known as user experience (UX) design. This includes features of branding, design, usability, and function, as well as the full process of obtaining and integrating the product.

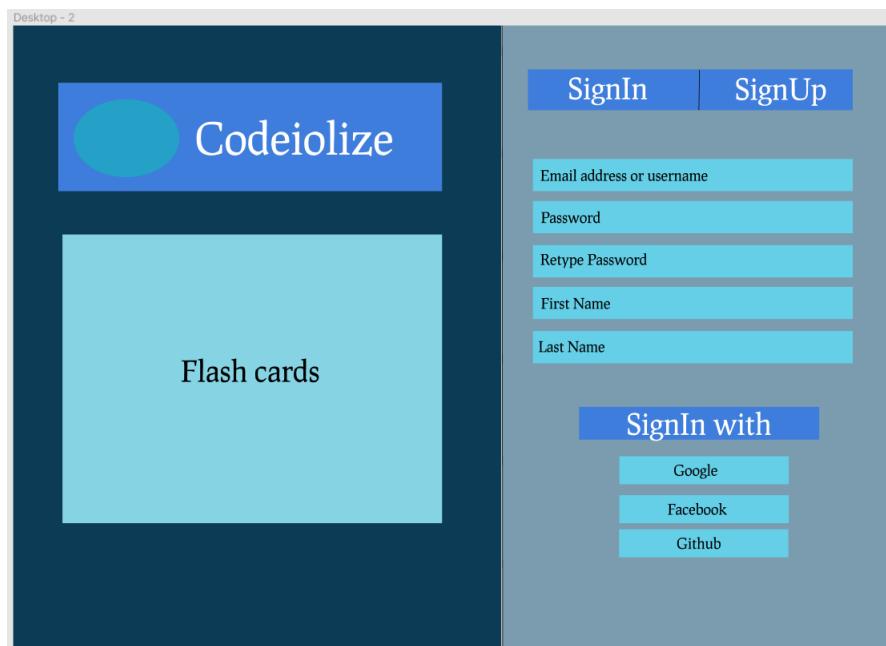


Figure 3.9: Main Page - Sign Up

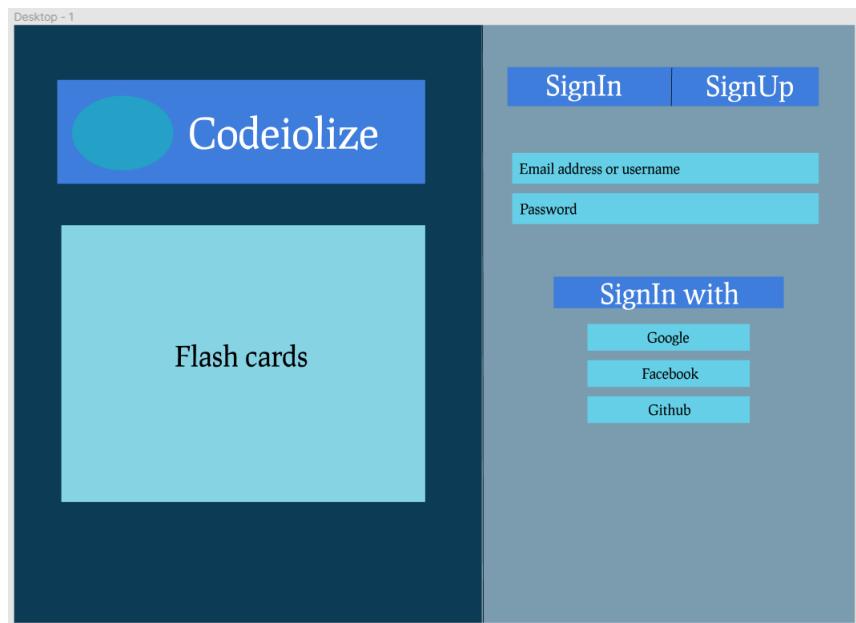


Figure 3.10: Main Page - Sign in

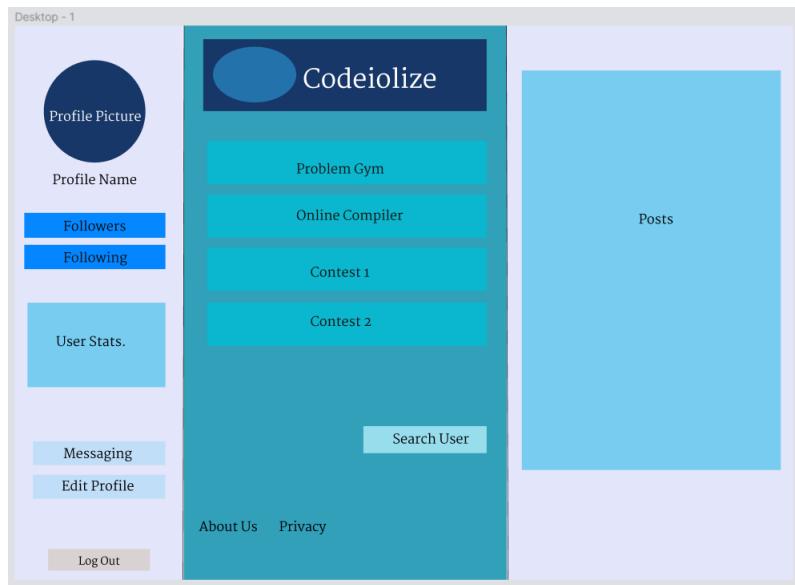


Figure 3.11: Home Page

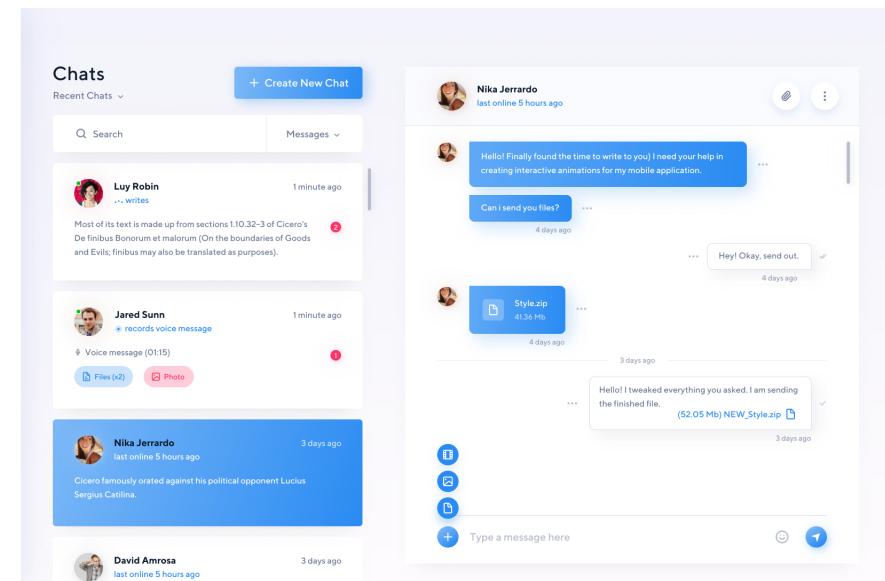


Figure 3.12: Chat Dashboard

Chapter 4

IMPLEMENTATION

Following are the screenshots of the project implementation.

User Authentication

This module comprises of login and signup pages.

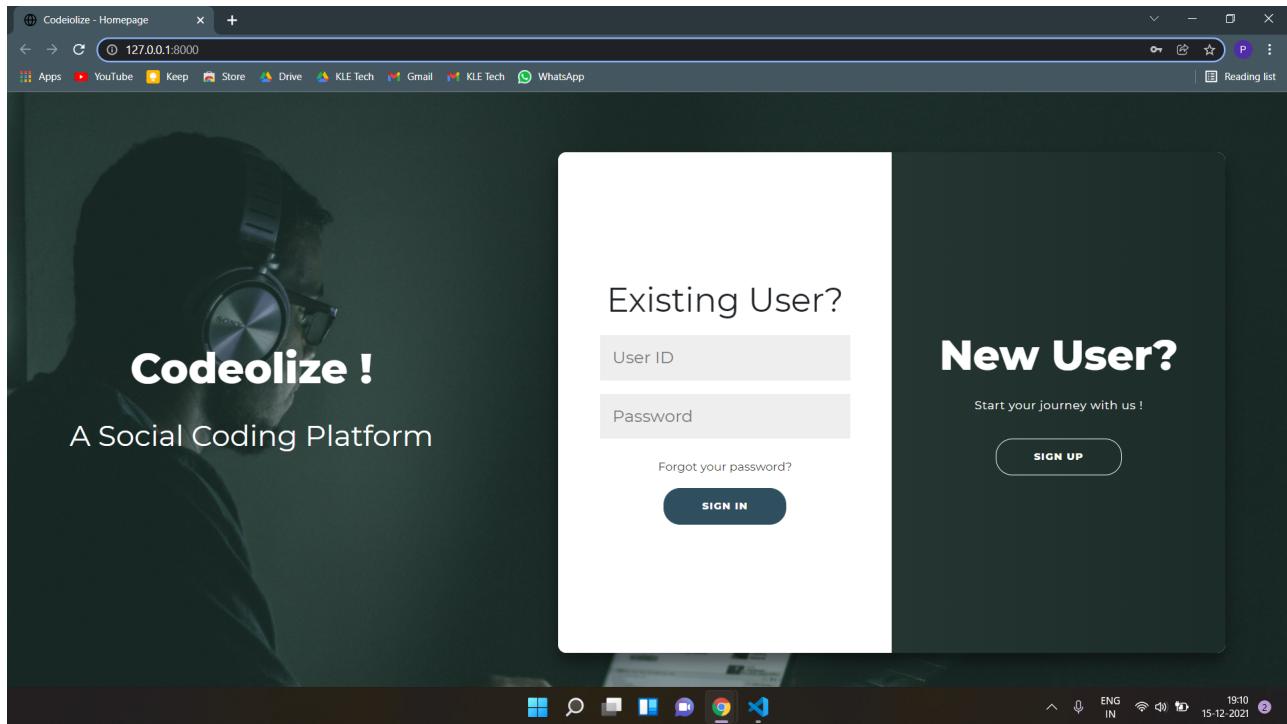


Figure 4.1: Login page

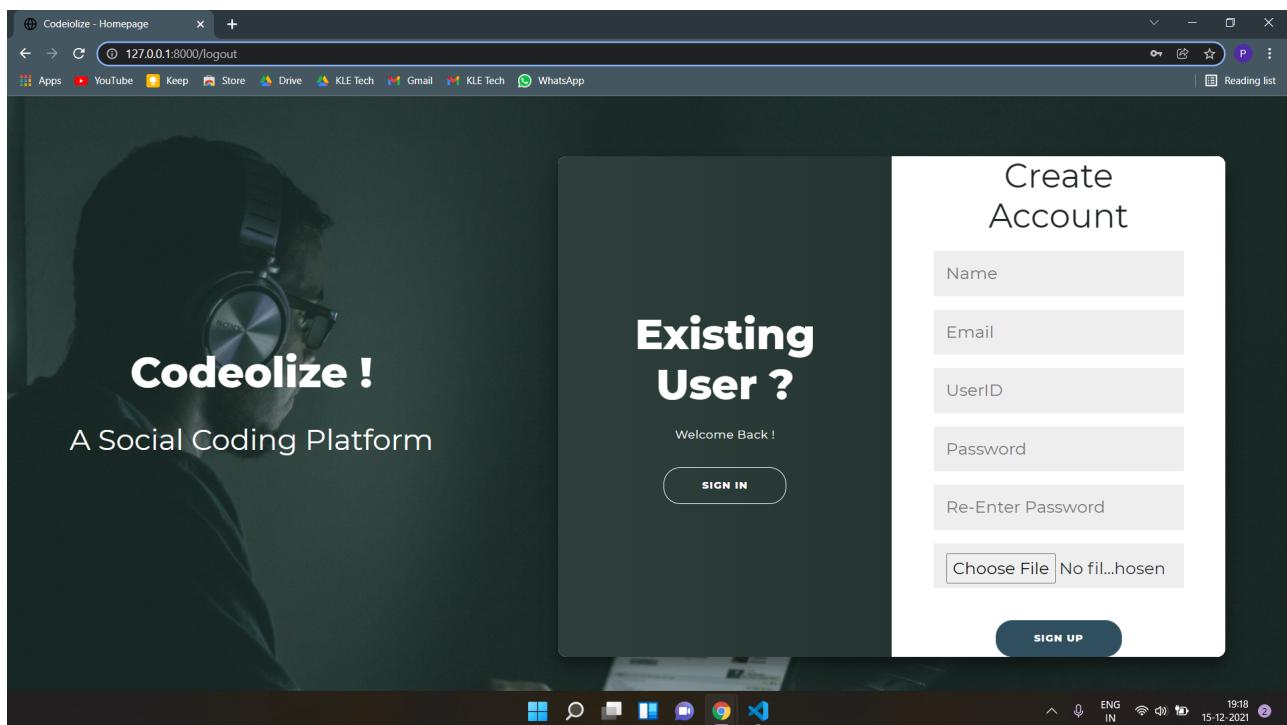


Figure 4.2: Sign-up page

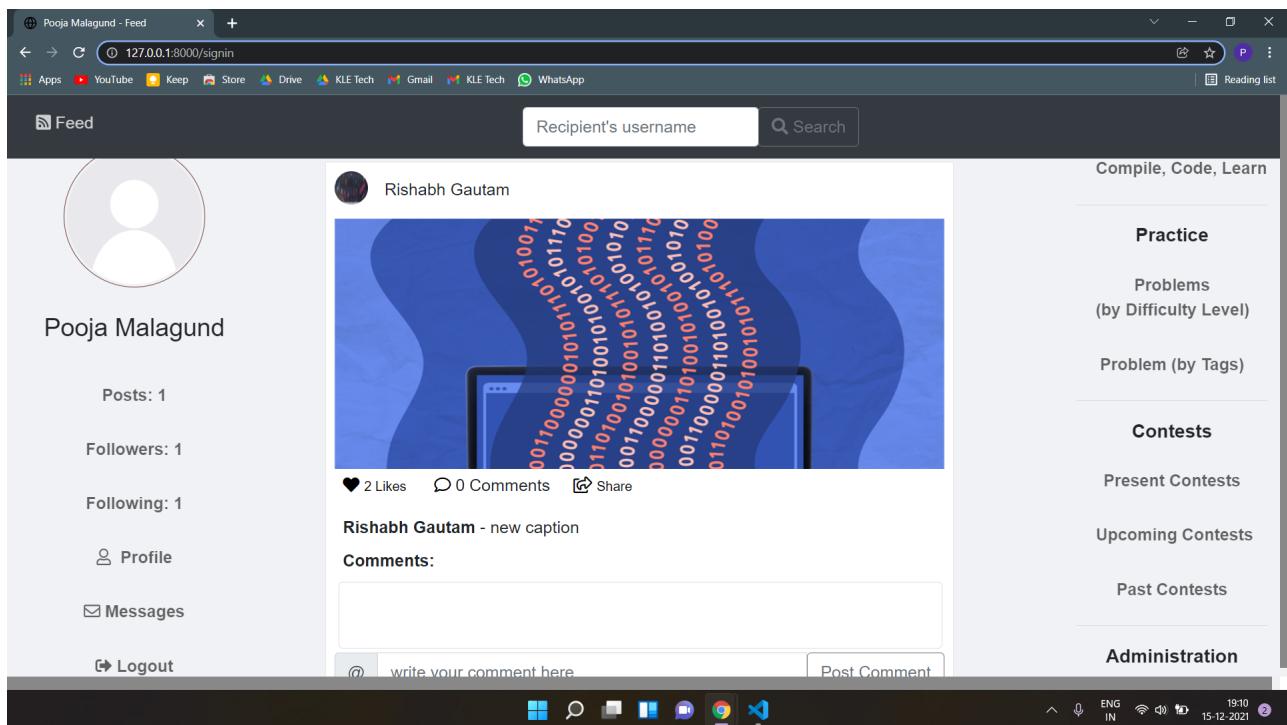


Figure 4.3: Home page

Code Compiler

This module comprises of a code compiler that supports more than fifty languages.

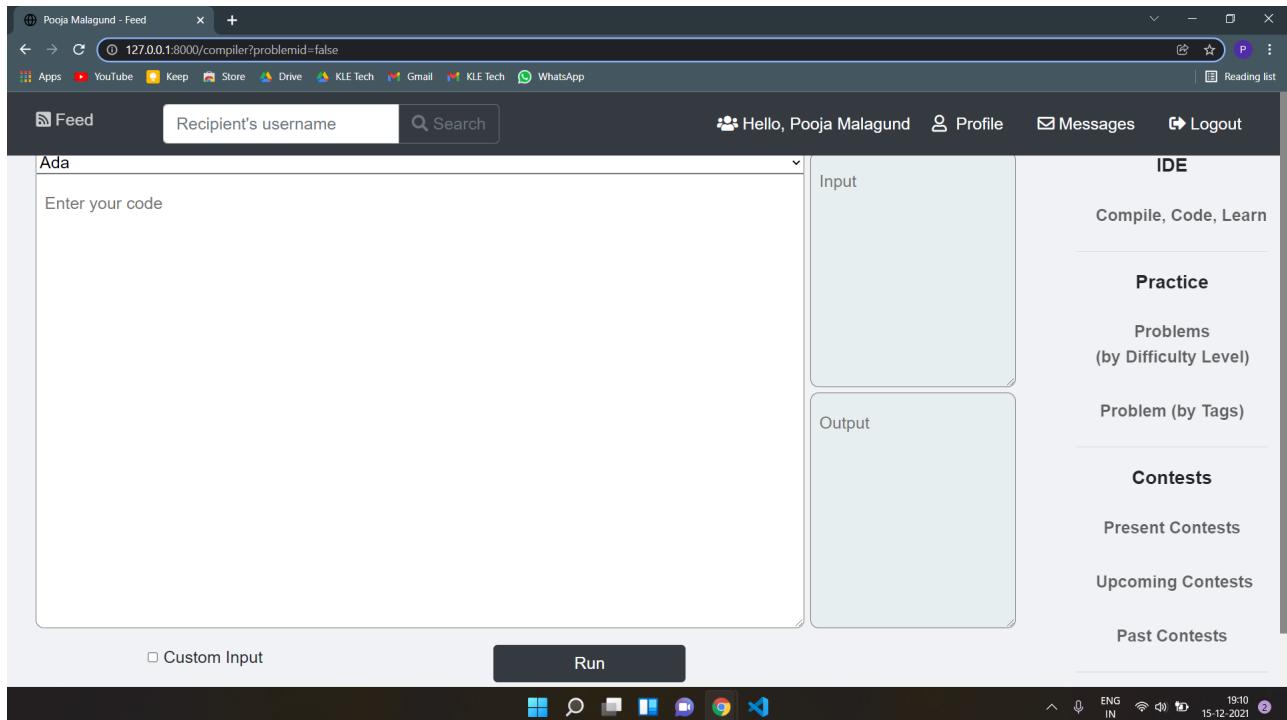


Figure 4.4: Code Compiler page

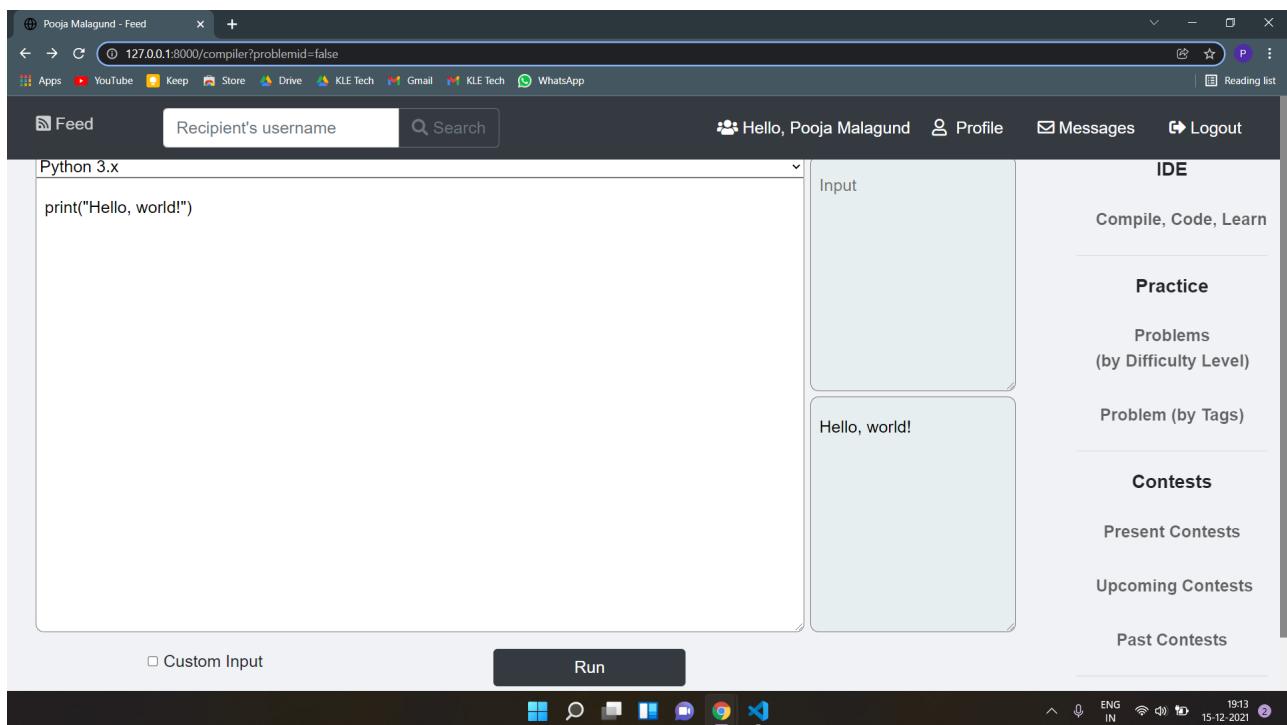


Figure 4.5: An example of Python code, input and output

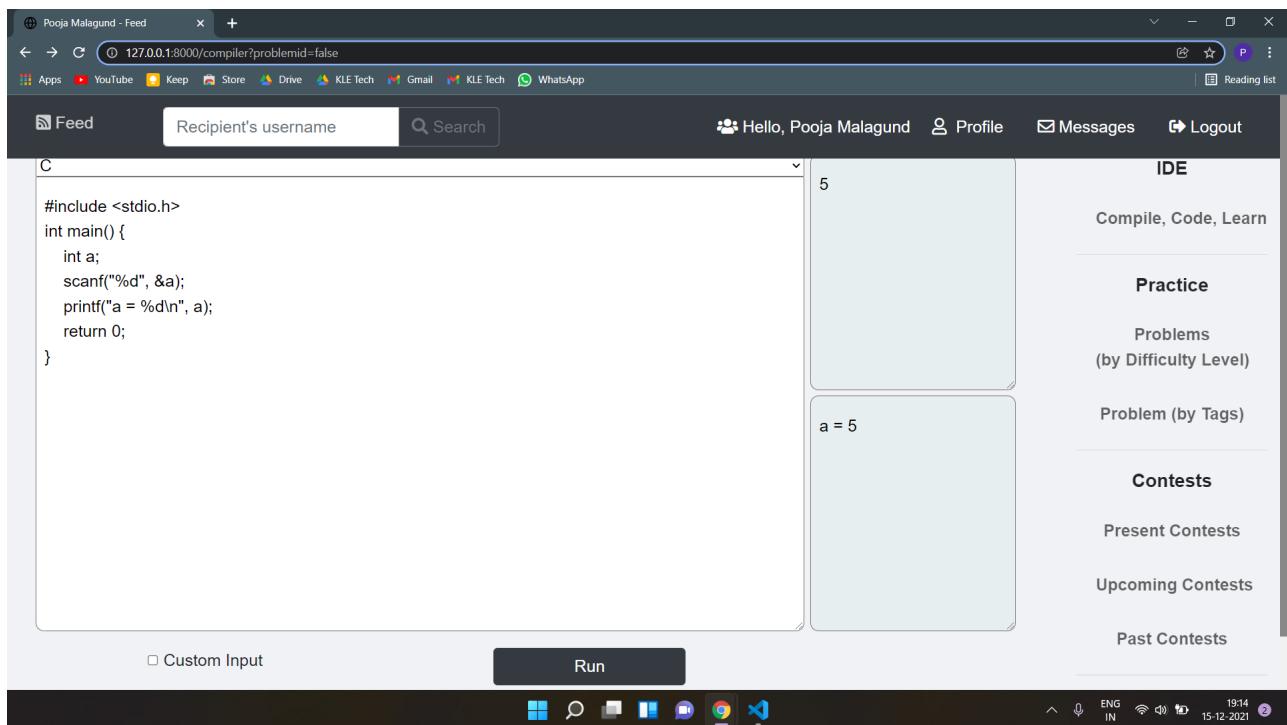


Figure 4.6: An example of C code, input and output

Problem Gym

This section comprises of set of problems where user can practice his/her coding skills. The problems are organized according to difficulty levels and topics.

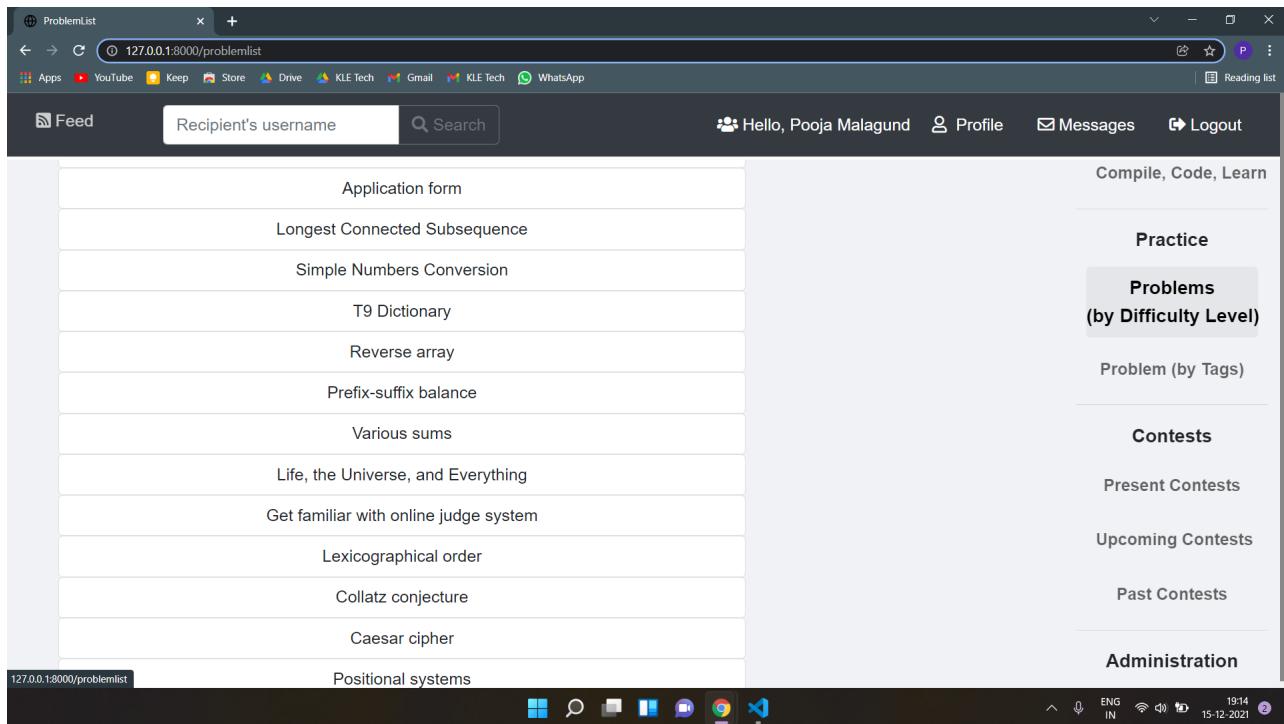


Figure 4.7: Problem sets page

Application form - SE_0025

Your task is to write a program which verifies whether a certain form was filled correctly or not. Data in these forms is of the following format:

Name: F; Surname: L; Birth date: YYYY-MM-DD

where F has the length not exceeding 10, L has length not exceeding 20, YYYY is 4 digit year format, MM is 2 digit month format and DD is 2 digit day format.

Input

An undefined amount of data sets will appear in the input. Every set consists of 3 fields separated with a semicolon and you can assume that fields F, L, YYYY, MM and DD consist of non-white signs only, other than semicolon. Each data set will be separated with a new line character.

Output

A sequence of numbers should appear on output, where i^{th} term of the sequence equals to:

- 0 if i^{th} form does not contain a correct name; correct name should begin with a capital letter and may contain minuscules afterwards;

Figure 4.8: Detailed view of problem (1)

• 2 if i^{th} form contains correct name and surname, but does not contain a correct birth date; the date is correct if YYYY is an integer in the range 1900-2000, MM is an integer in the range between 1-12 and DD is an integer in the range 1-31;

• 3 in other cases.

Each element of the output sequence should be separated with a new line character.

Example

Input:

```
Imie: Roman; Nazwisko: Kowalski6; Data ur.: 1900-01-30
Imie: Andrzej; Nazwisko: Kowal; Data ur.: 1899-10-10
Imie: roman; Nazwisko: No-wak; Data ur.: 1099-11-12
Imie: Alicja; Nazwisko: Nowak; Data ur.: 1990-01-01
```

Output:

```
1
2
0
3
```

Figure 4.9: Detailed view of problem (2)

Social Media

This module comprises of user profile, posts, followers and following, and chats. The user can also search for existing users in the network using search tab.

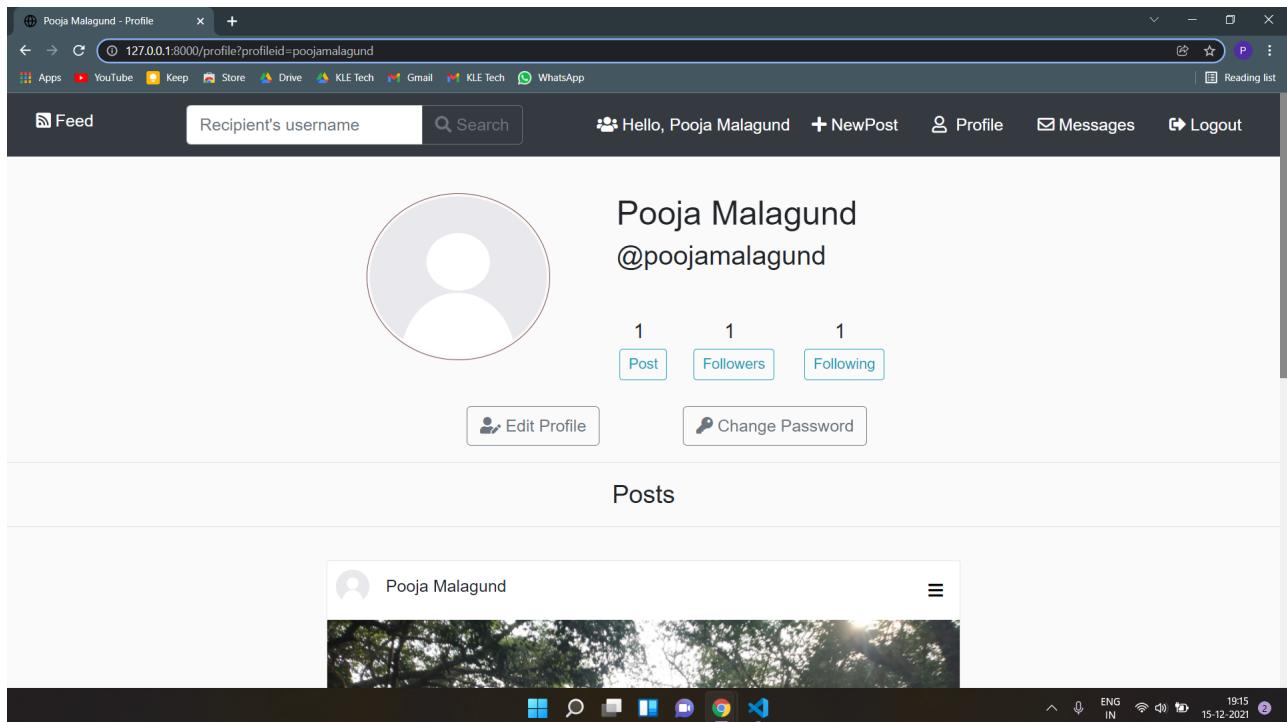


Figure 4.10: User profile page

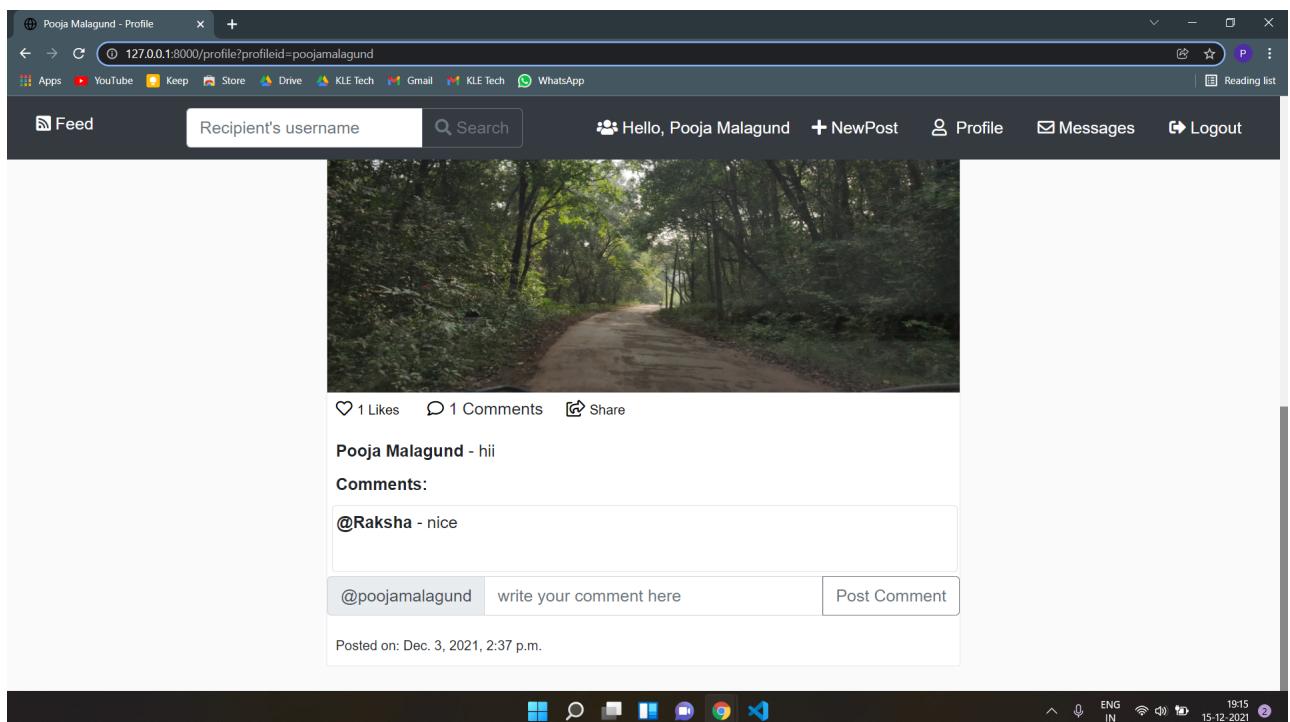


Figure 4.11: Likes, comments on a post

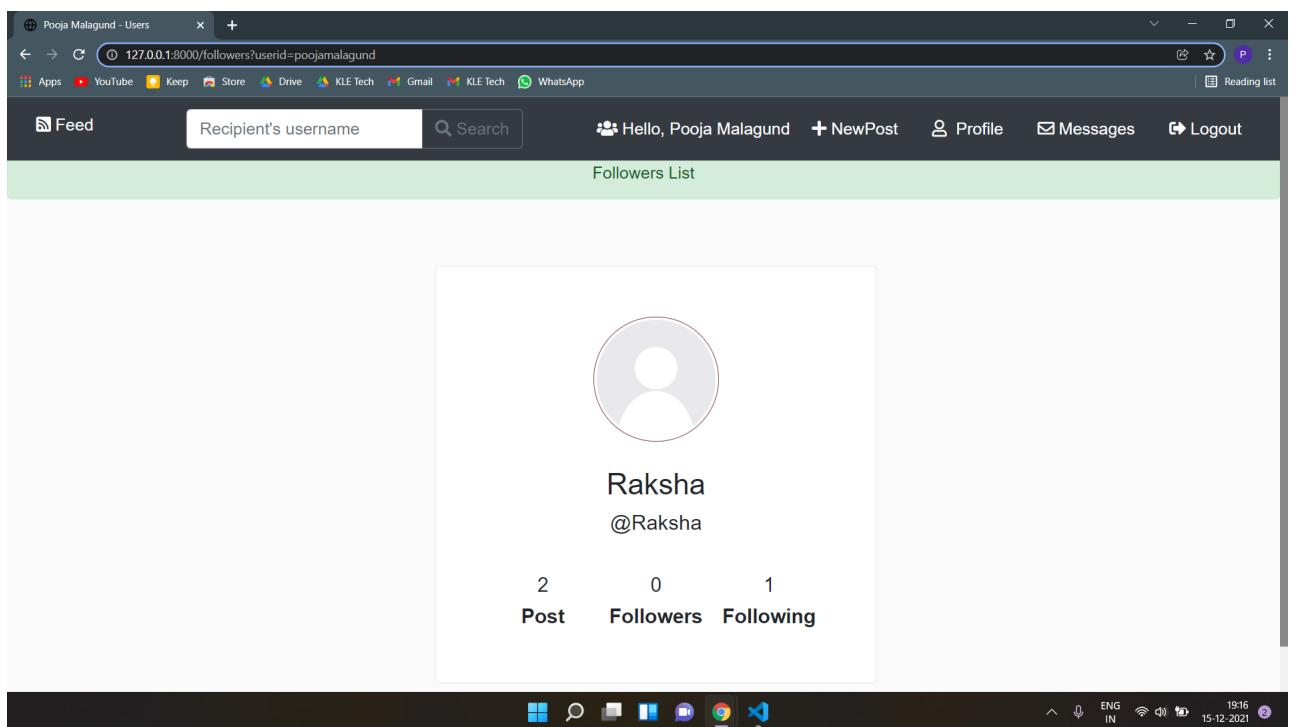


Figure 4.12: Followers list page

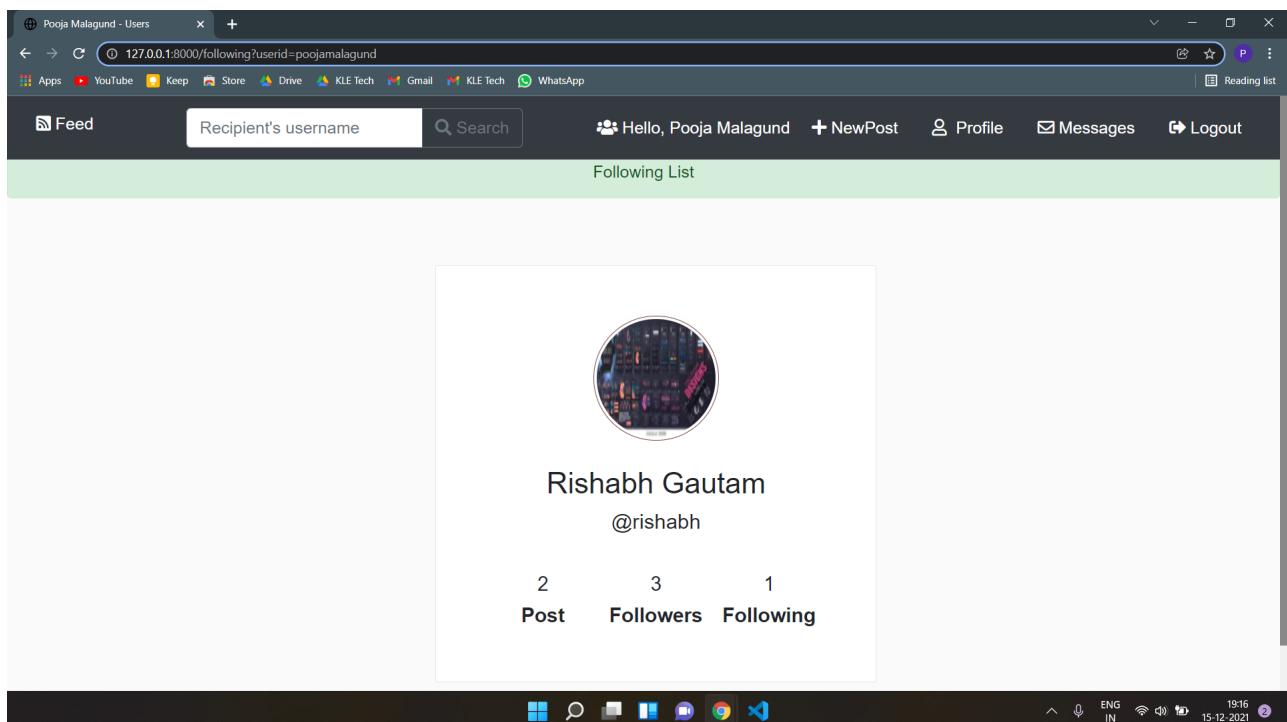


Figure 4.13: Following list page

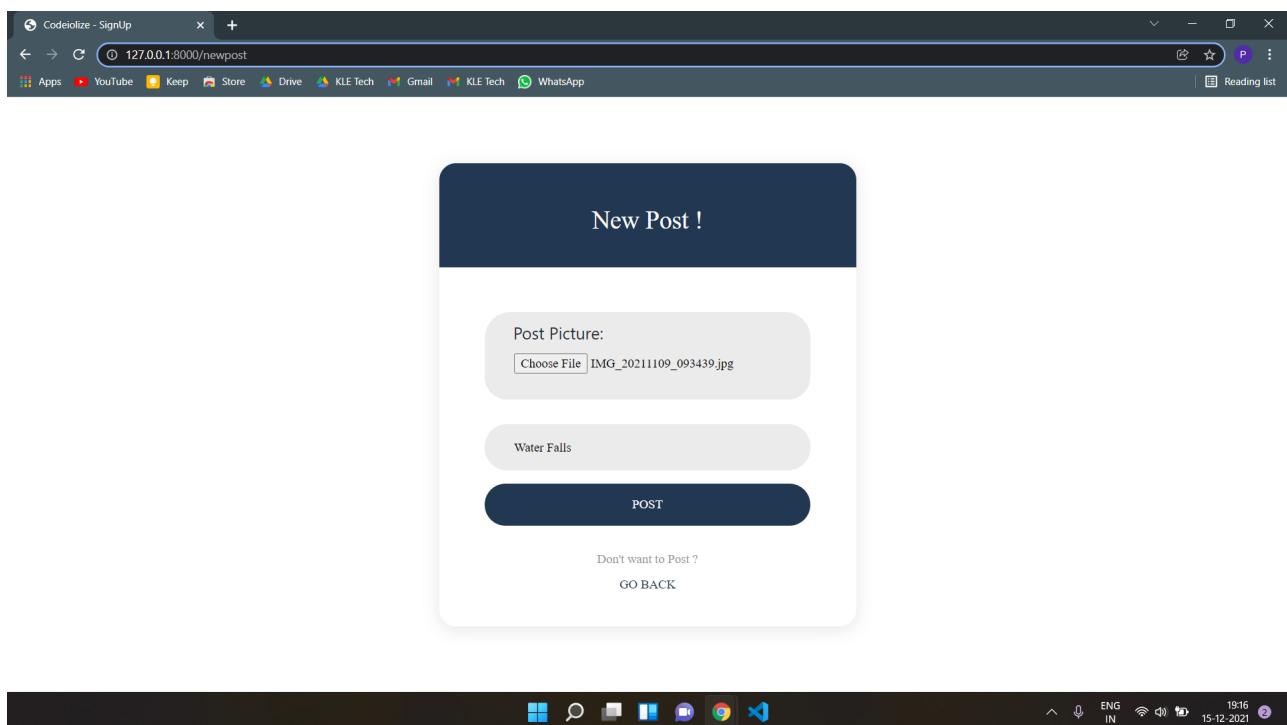


Figure 4.14: Uploading new post

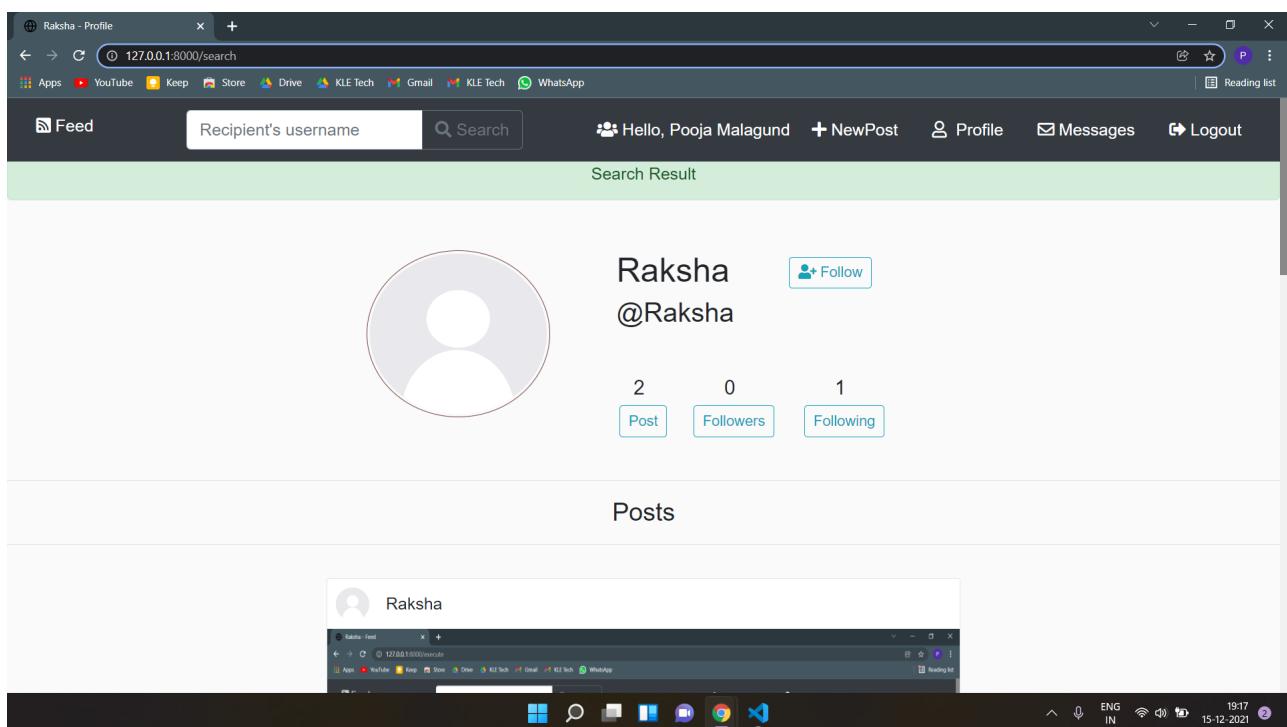


Figure 4.15: Search results page

Chapter 5

RESULTS AND DISCUSSIONS

The objectives mentioned,

- Develop an Online Integrated Development Environment (IDE) for executing codes in languages like C, C++, Java, Python
- Provide users with a problem gym and allow them to build problem solving skills.
- Provide an integrated social networking platform for the users.

have been implemented with all the functionalities mentioned in each of the module.

The Online IDE, social media integration and the problem gym to solve the questions have been worked upon with most of the error handling mechanisms provided. The user can solve while providing custom input to the compiler, post his/her achievements and also maintain a contact with the other coders on the platform. The platform allows users to communication via messages allowing them to send audio as well as image files. Every keen detail, as the last seen of a user, the status of an user being offline or online has been added.

Chapter 6

CONCLUSION AND FUTURE SCOPE OF THE WORK

Further, there is a scope of much development over the existing project. To mention some,

- Conduct online coding contests,
- Provide a platform for job-hiring,
- Integrate hackathons and codeathons.

There can be many more functionalities that can be added, above were to name some.

Report Title:	SDP Team D6
Report Link: (Use this link to send report to anyone)	https://www.check-plagiarism.com/plag-report/35658bee0d41f9c3a057b83fd574ba9363dd51640080850
Report Generated Date:	21 December, 2021
Total Words:	4235
Total Characters:	28471
Keywords/Total Words Ratio:	0%
Excluded URL:	No
Unique:	88%
Matched:	12%

REFERENCES

1. Problems API Version 4 Overview, Sphere Engine, accessed on 16 August 2021,
[<https://docs.sphere-engine.com/problems/api/overview-version-4>](https://docs.sphere-engine.com/problems/api/overview-version-4)
2. Problems, DMOJ: Modern Online Judge, accessed on 18 August 2021,
[<https://dmoj.ca/problems/>](https://dmoj.ca/problems)
3. Problem Set, CSES, accessed on 18 August 2021, [<https://cses.fi/problemset/list/>](https://cses.fi/problemset/list)
4. Online Compiler API for Java, C, C++, PHP, Pearl, Jdoodle, accessed on 21 August 2021, [<https://www.jdoodle.com/compiler-api/>](https://www.jdoodle.com/compiler-api)
5. CodeForces Employee Details, Technology Stack and more, Slintel.com, accessed on 04 August 2021,
[<https://www.slintel.com/company/codeforces/5c3b0280d55ae49f1b7975f6>](https://www.slintel.com/company/codeforces/5c3b0280d55ae49f1b7975f6)
6. Scaling Database with Django and HAProxy, HackerEarth, accessed on 25 August 2021, [<http://engineering.hackerearth.com/2013/10/07/scaling-database-with-django-and-haproxy/>](http://engineering.hackerearth.com/2013/10/07/scaling-database-with-django-and-haproxy)
7. HackerEarth Technology Stack, HackerEarth, accessed on 04 August 2021,
[<http://engineering.hackerearth.com/2013/03/20/hackerearth-technology-stack/>](http://engineering.hackerearth.com/2013/03/20/hackerearth-technology-stack)