

**A MINI-PROJECT (1) REPORT ON**  
**CODER BLOG**  
**Submitted in partial fulfillment of requirements**  
**for the award of the degree of**  
**BACHELOR OF TECHNOLOGY**  
**IN**  
**COMPUTER SCIENCE AND ENGINEERING**

**Submitted by:**

<b>K. VAMSI MOHAN REDDY</b>	<b>(19091A05H3)</b>
<b>A. SAI THANU SREE</b>	<b>(19091A05C7)</b>
<b>L. VASAVI</b>	<b>(19091A05H4)</b>
<b>V. SANJAY</b>	<b>(18091A05C0)</b>

**Under the Guidance of**

**Dr. N. MADHUSUDHANA REDDY** M. Tech, Ph.D.  
**Professor, Dept. of CSE**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**  
**RAJEEV GANDHI MEMORIAL COLLEGE OF ENGINEERING & TECHNOLOGY**  
**(AUTONOMOUS)**  
AFFILIATED TO JNTUA ANANTAPURAMU, APPROVED BY A.I.C.T.E., NEW DELHI,  
ACCREDITED BY NAAC of UGC, NEWDELHI with "A+" grade,  
ACCREDITED BY N.B.A, NEWDELHI,  
NANDYAL-518501, (Estd-1995)

# **Rajeev Gandhi Memorial College of Engineering & Technology (AUTONOMOUS)**

AFFILIATED TO JNTUA ANANTAPURAMU, APPROVED BY A.I.C.T.E., NEW DELHI,  
ACCREDITED BY NAAC of UGC, NEWDELHI with “A+” grade,  
ACCREDITED BY N.B.A, NEWDELHI,  
NANDYAL-518501, (Estd-1995)

**(ESTD – 1995)**



## **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

### **CERTIFICATE**

This is to certify that **K. VAMSI MOHAN REDDY** (*19091A05H3*), **A. SAI THANU SREE** (*19091A05C7*), **L. VASAVI** (*19091A05H4*) and **V. SANJAY** (*18091A05C0*) of III- B. Tech II- semester, have carried out the mini-project (1) work entitled “**CODER BLOG**” under the supervision and guidance of **Dr. N. MADHUSUDHANA REDDY**, Professor, CSE Department, in partial fulfillment of the requirements for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering** from **Rajeev Gandhi Memorial College of Engineering & Technology (Autonomous)**, Nandyal is a bonafied record of the work done by them during 2021-2022.

#### **Project Guide**

**Dr. N. Madhusudhana Reddy** M. Tech, Ph.D.

Professor, Dept. of CSE

**Place:** Nandyal

**Date:**

#### **Head of the Department**

**Dr. K. Subba Reddy** M. Tech, Ph.D.

Professor & H.O.D, Dept. of CSE

**External Examiner**

## ***Candidate's Declaration***

We hereby declare that the work done in this project entitled “**CODER BLOG**” submitted towards completion of mini-project (1) in *III Year II Semester of B. Tech (CSE)* at the **Rajeev Gandhi Memorial College of Engineering & Technology**, Nandyal. It is an authentic record of our original work done under the esteemed guidance of **Dr. N. Madhusudhana Reddy**, Professor, Department of **COMPUTER SCIENCE AND ENGINEERING**, RGMCET, Nandyal.

We have not submitted the matter embodied in this report for the award of any other Degree in any other institutions.

**By**

**K. Vamsi Mohan Reddy** (19091A05H3)

**A. Sai Thanu Sree** (19091A05C7)

**L. Vasavi** (19091A05H4)

**V. Sanjay** (18091A05C0)

Dept. of CSE,

RGMCET.

**Place:** Nandyal

**Date:**

## **ACKNOWLEDGEMENT**

We manifest our heartier thankfulness pertaining to your contentment over our project guide **Dr. N. Madhusudhana Reddy** garu, Professor of Computer Science Engineering Department, with whose adroit concomitance the excellence has been exemplified in bringing out this project to work with artistry.

We express our gratitude to **Dr. K. Subba Reddy** garu, Head of the Department of Computer Science Engineering department, all teaching and non-teaching staff of the Computer Science Engineering Department of Rajeev Gandhi memorial College of Engineering and Technology for providing continuous encouragement and cooperation at various steps of our project.

Involuntarily, we are perspicuous to divulge our sincere gratefulness to our Principal, **Dr. T. Jaya Chandra Prasad** garu, who has been observed posing valiance in abundance towards our individuality to acknowledge our project work tangentially.

At the outset we thank our honorable **Chairman Dr. M. Santhi Ramudu** garu, for providing us with exceptional faculty and moral support throughout the course.

Finally, we extend our sincere thanks to all the **Staff Members** of CSE Department who have co-operated and encouraged us in making our project successful.

Whatever one does, whatever one achieves, the first credit goes to the **Parents** be it not for their love and affection, nothing would have been responsible. We see in every good that happens to us their love and blessings.

### **BY**

**K. Vamsi Mohan Reddy** (19091A05H3)

**A. Sai Thanu Sree** (19091A05C7)

**L. Vasavi** (19091A05H4)

**V. Sanjay** (18091A05C0)

## **ABSTRACT**

In this trending world of computer science most of the students are concentrating on programming knowledge. To maximize the ability in programming students are learning code from different sites and practicing in those websites. Here in this website, we aimed to increase the programming related question with the help of person who has knowledge in this field. The answers from users will be considered by faculty by analyzing complexities and program logic, the best one will have selected by faculty and it will post in previous programs page with details of that first ranked person. The users of this website can see programs and can be able to ask doubts. It builds competitiveness among students.

# **CONTENTS**

<b>Chapter No.</b>	<b>Title</b>	<b>Page No.</b>
<b>1.</b>	<b>INTRODUCTION</b>	
1.1	Introduction	1
1.2	Objectives	1
<b>2.</b>	<b>SYSTEM ANALYSIS</b>	
2.1	Existing System	2
2.1.1	Disadvantages	2
2.2	Motivation to the problem	3
2.3	Proposed System	3
2.3.1	Advantages	4
<b>3.</b>	<b>FEASIBILITY STUDY</b>	
3.1	Technical Feasibility	5
3.2	Operational Feasibility	5
3.3	Economic Feasibility	6
<b>4.</b>	<b>SYSTEM REQUIREMENT SPECIFICATION</b>	
4.1	Requirement Analysis	7
4.2	Software Requirements Specification (SRS)	7
4.3	Functional Requirements	8
4.4	Non-Functional Requirements	9
4.5	Software Requirements	10
4.6	Hardware Requirements	10

<b>5.</b>	<b>SYSTEM DESIGN</b>	
5.1	Modules	11
5.1.1	Admin	11
5.1.2	User	11
5.2	Introduction to UML	12
5.3	UML diagrams	14
<b>6.</b>	<b>SYSTEM IMPLEMENTATION</b>	
6.1	Technologies	20
6.1.1	PHP	22
6.2	Sample code	26
6.3	Database tables	36
6.4	Results/ Screenshots	40
<b>7.</b>	<b>SYSTEM TESTING</b>	
7.1	Test case Description	52
7.2	Types of Testing	53
<b>8.</b>	<b>CONCLUSION</b>	56
<b>9.</b>	<b>FUTURE ENHANCEMENT</b>	57
	<b>BIBLIOGRAPHY</b>	58

## **LIST OF FIGURES**

---

Fig. 5.1 Use Case Diagram.....	14
Fig. 5.2 Class Diagram.....	15
Fig. 5.3 Sequence Diagram for admin.....	16
Fig. 5.4 Sequence Diagram for user.....	17
Fig. 5.5 Activity Diagram for admin.....	18
Fig. 5.6 Activity Diagram for user.....	19

## **LIST OF TABLES**

---

Table 6.1: Database table for Admin login.....	36
Table 6.2: Database table for Post question.....	37
Table 6.3: Database table for Student login.....	38
Table 6.4: Database table for Submit answer.....	38
Table 6.5: Database table for Feedback.....	39

## **1. INTRODUCTION**

### **1.1 Introduction:**

Coder Blog is a web-based application regarding daily assignments especially for computer science students. In this website problems/ programs could be posted by admin or faculty. Those problems will have some time limit. The answers or solutions would be posted by the students by using login id. Those answers would be analyzed and gives credits to the answers of each and every individual student. Finally after timed out the results will be announced by the admin i.e., the best answer or the solution for the given problem will be taken and the remaining solutions will be discarded.

This project explains about implementing a website which is conveniently to use for both student and faculty or admin about daily assignments of the programming related questions. Faculty can be easily broadcast a question to all the users of the website and student can be able to submit the answer with valid user account details.

### **1.2 Objectives:**

Our project “CODER BLOG” aims at providing an effective programming acknowledgement. The objectives of our project are noted as follows:

- Question or program will be displayed.
- Transparent view of solutions for the given programming question.
- A way to know the effective solutions or to improve our programming knowledge.
- Easy way to credit and announce the marks.

## **2. SYSTEM ANALYSIS**

### **2.1 Existing system:**

We have observed that faculty give list of programming related questions to solve as an assignment. The programs which are related to what they are teaching currently at that time. To improve the knowledge in particular topic what they have taught they give assignments to students every day.

#### **2.1.1 Disadvantages of Existing system:**

##### **1. Need more man power**

As we know, most of the universities would post any questions on the notice board. This can be major problem for the most of the users to go and check.

##### **2. No effective solutions**

Not all the students are expertized in programming and everyone can't get effective solutions for programs.

##### **3. Student cannot get previously posted questions and solutions easily after submission**

After submitting the assignments, not all the students save them for further use. So, this will be creating a problem at the time of revision.

##### **4. Access issues**

No universities will provide previous solutions for a given program for a long time.

##### **5. Hectic Schedule**

Due to the busy schedule of faculty, sometimes they may not be available to take submissions of students physically.

## 2.2 Motivation to the Problem:

The project “CODER BLOG” is a website which will be helpful to the student to get an effective solution for a program and an easy communication between higher authorities and students, provides immediate response to the students through feedback.

## 2.3 Proposed system:

The proposed system is computerizing the all the operations what the existing system consist. For that we need one website or application. So, we proposed the “CODER BLOG” website which facilitates all the operations from where the user present. Each user has one account which unique id and password. Admin can be able to post questions, analyze submitted programs and announce results. Students can be post answers within the time limit of the programming question. If they encountered with any queries, they can be able to ask in comment section.

- **Availability**

As it is an online website, availability is ensured all the ways.

- **Zero Expenses**

Services are provided 24\*7 at free of cost.

- **Programming Acknowledgement**

It not only provides you with the effective solutions but also acts as a knowledge hub regarding programming languages.

- **Accessing**

You can access the best solutions at any time with free of cost.

- **Security**

Your login details will not be shared among any other person.

- You can get a clear idea on the programming with an effective solution.

### **2.3.1 Advantages of Introduced System:**

1. Availability is ensured all the ways.
2. Services at free of cost.
3. Acts as a knowledge hub regarding programming.
4. It not only suggests programs but also provides effective solution.

### **3. FEASIBILITY STUDY**

Feasibility study is an important phase in the software development process. Preliminary investigation examines project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system.

Feasibility study should be performed on the basis of various criteria and parameters. The various feasibility studies are:

- Technical Feasibility
- Operation Feasibility
- Economic Feasibility

#### **3.1 Technical Feasibility:**

Coder Blog is a complete web-based application. The main technologies and tools that are associated with this application are freely available and technical skills required are manageable. Time limitations of the product development and the ease of implementing using these technologies are synchronized.

Initially the website will be hosted in free web hosting space. Bandwidth required in this application is very low, since it doesn't incorporate any multimedia aspect.

From these it's clear that the project Coder Blog is technically feasible.

#### **3.2 Operational Feasibility:**

##### **User-friendly:**

The system will be supportive and easy to use for the user. The interfaces are designed to make it easy for any potential user to get familiar with the system within less time.

##### **Security:**

Security measures are provided in many aspects in this system.

**User authentication:**

Users will have to authenticate using username and passwords. Depending on the access level each user will gain functionality of the system. Passwords can be changed by the user.

**Portability:**

The application will be developed using standard open-source software (Except Oracle) like Java, tomcat web server, xampp web server, Internet Explorer Browser etc. this software will work both on Windows and Linux o/s. Hence portability problems will not rise.

**Availability:**

As it is an online website, availability is ensured always.

**Maintainability:**

The system called the wheels uses the 2-tier architecture. The 1st tier is the GUI, which is said to be front- end and the 2nd tier is the database, which uses My-Sql, which is the back-emblems will not arise.

**3.3 Economic Feasibility:**

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economic feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs. The system is economically feasible. It does not require any additional hardware or software.

## **4. SYSTEM REQUIREMENT SPECIFICATION**

### **4.1 Requirement analysis**

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified. It involves all the tasks that are conducted to identify the need of different stakeholders.

Requirement Analysis is a software engineering task that bridges the gap between system level software allocation and software design. Requirement Analysis is to specify software function and performance indicates software interface with other system elements.

Requirement Analysis of the system starts with the specification given by the user. This user-required specification could be formal or informal. In formal method, the user clearly states the purpose of the software. This acts as the good basis for the software engineers for the requirement analysis. In informal method purpose and the outputs are not clearly specified by the user. The responsibility is on the software engineer to get the purpose and outputs by interacting more the user.

### **4.2 Software Requirements Specification (SRS)**

A software requirements specification, a requirements specification for a software system is complete description of the behavior of a system to be developed and may include a set of use cases that describe interactions the user will have with the software. In addition, it also contains non-functional requirements. Non-functional requirements impose constraints on the design or implementation.

The software requirements specification document enlists all necessary requirements that are required for the project development. To derive the requirements, we need to have clear and through understanding of the products to be developed. This is prepared after detailed communications with the project team and customer.

A Software Requirements Specification minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-world situations.

### 4.3 Functional Requirements:

In software engineering a functional requirement defines as a function of system or its components where a function is described as a specification of behavior between output and input, in our project functional requirements is according to user details. It provides the following actions.

- Username or Id \* and password\*

This feature is mandatory for accessing the website

### Benefits of Functional Requirements:

Here, are the pros/advantages of creating a typical functional requirement document-

- Helps you to check whether the application is providing all the functionalities that were mentioned in the functional requirements of that application.
- A functional requirement document helps you to define the functionality of a system or one of its subsystems.
- Functional requirements along with requirement analysis help identify missing requirements. They help clearly define the expected system service and behavior.
- Errors caught in the Functional requirement gathering stage are the cheapest to fix.
- Support user goals, tasks, or activities.

## 4.4 Non-Functional Requirements:

### Performance requirements:

The performance of our website is best when the following are regularly updated.

- Questions or Programs management.
- Database management.

### Security requirements:

Each member is required to enter an individual username or id when accessing the website, the administrator will provide high security to user.

The data in database is secured through multiple layers of protection.

### Software quality:

The quality of software is maintained in such a way so that it can be very friendly.

The software quality assumed as under specified measures.

- Secured
- Fast speed
- Compatibility
- Accurate
- Any time available

#### **4.5 Software Requirements:**

- Server : **XAMPP Version 8.0.14**
- Front end : **HTML5, CSS3, JAVASCRIPT**
- Back end : **PHP & MySQL**
- Database : **MySQL**
- Server-side scripting : **PHP**
- Client-side scripting : **Java Script**

#### **4.6 Hardware requirements:**

- Hard Disk Drive : **4GB or more**
- Processor : **intel Processor**
- Main Memory : **1GB or more**

## **5. SYSTEM DESIGN**

Design is a meaningful engineering representation of something that is to be built. It is the most crucial phase in the developments of a system. Software design is a process through which the requirements are translated into a representation of software. The logical system design arrived at as a result of systems analysis is converted into physical system design.

### **5.1 Modules:**

A module is defined as the unique and addressable components of the software which can be solved and modified independently without disturbing (or affecting in very small amount) other modules of the software. Thus, very software design should follow modularity. The process of breaking down a software into multiple independent modules where each module is developed separately is called Modularization. In this “Coder Block” Website there are two modules those are user module and admin module

#### **5.1.1 Admin Module:**

Admin holds an account having both username and password. Admin can be able to post the programming question in particular language with expired date and time. Before or after the time expired time of question admin evaluate the user submitted programs and credits marks according to that. After the time is expired the admin can announce the result.

The admin can't announce results even at least one submitted user program not credited with marks. And Admin can be able to announce results only after timed out of the posted question for getting all the response from the user.

#### **5.1.2 User Module:**

Here user means student. When the student opens the website, he/she able to see the trending question with the timer how much have been left to expire the posted question, which defines whether the question is expired or not. First students have to create the account, in order to create account they have to fill some basic details. After successful login student can be able to send the answer to the posted question in pdf format, check

the previous own submissions with credited marks and previous programs with answers, which is best among the all the submitted programs. This page contains the best programs with the submitted user profile.

## 5.2 Introduction to UML:

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.

Each UML diagram is designed to let developers and customers view a software system from a different perspective and in varying degrees of abstraction. UML diagrams commonly created in visual modeling tools include.

### 5.2.1 Class Diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. It describes the attributes and operations of a class and also the constraints imposed on the system

### 5.2.2 Use case Diagram

A use case diagram at its simplest is a representation of user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other of diagrams as well. The use cases are represented by either circles or ellipses.

### 5.2.3 State Diagram

State chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. The most important purpose of State chart diagram is to model lifetime of an object from creation to termination.

#### 5.2.4 Activity Diagram

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join etc.

#### 5.2.5 Sequence Diagram

Sequence diagrams in UML shows how object interact with each other and the order those interactions occur. It's important to note that show the interactions for a particular scenario. The processes are represented vertically and interactions are show as arrows.

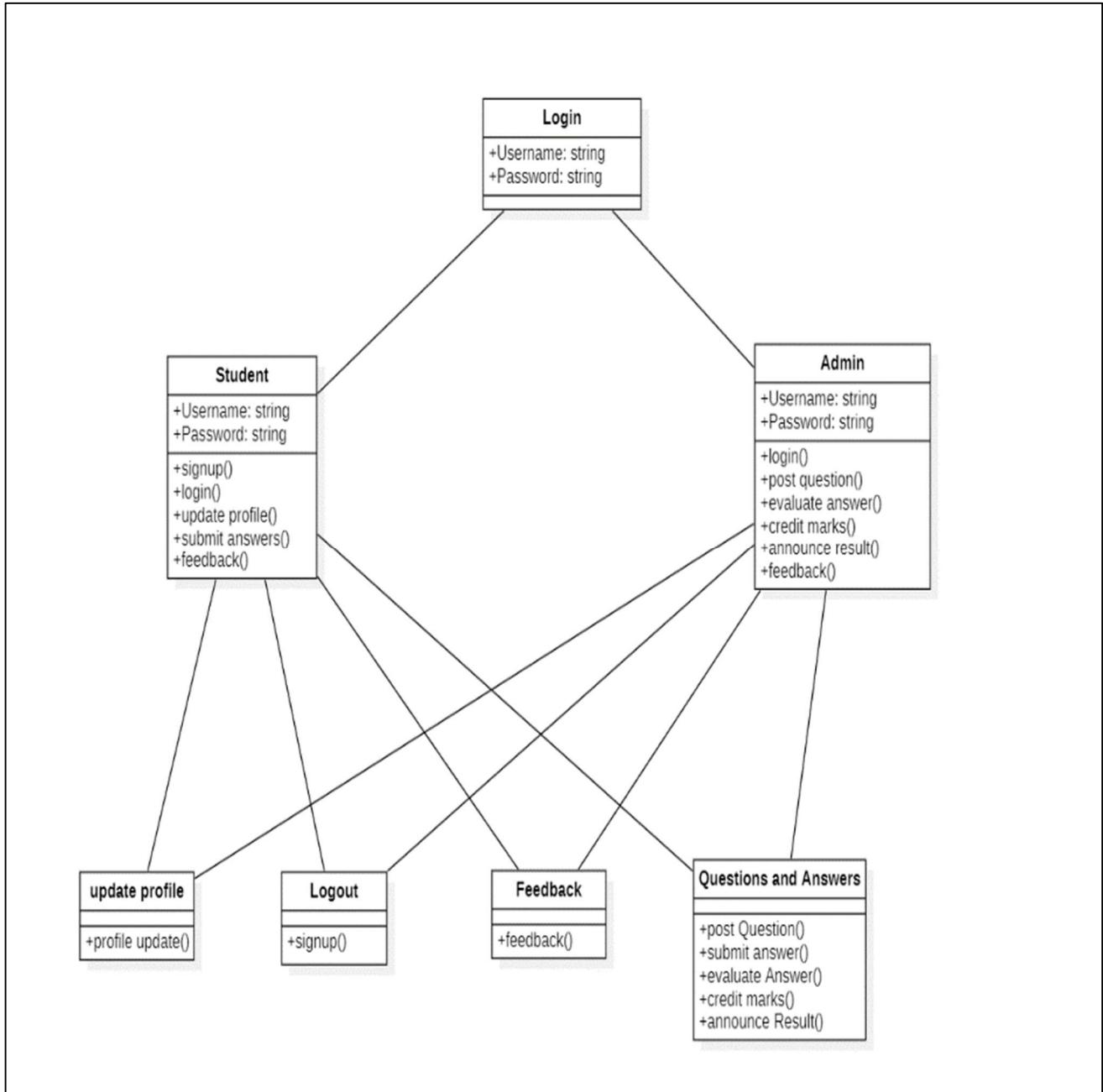
### 5.3 UML Diagrams:

The figure 5.1 shows the use case diagram of the system. The system consists of mainly two actors or users interacts with it, they are Admin and Student. The use cases they involved and their interaction with the system are depicted in the below figure.



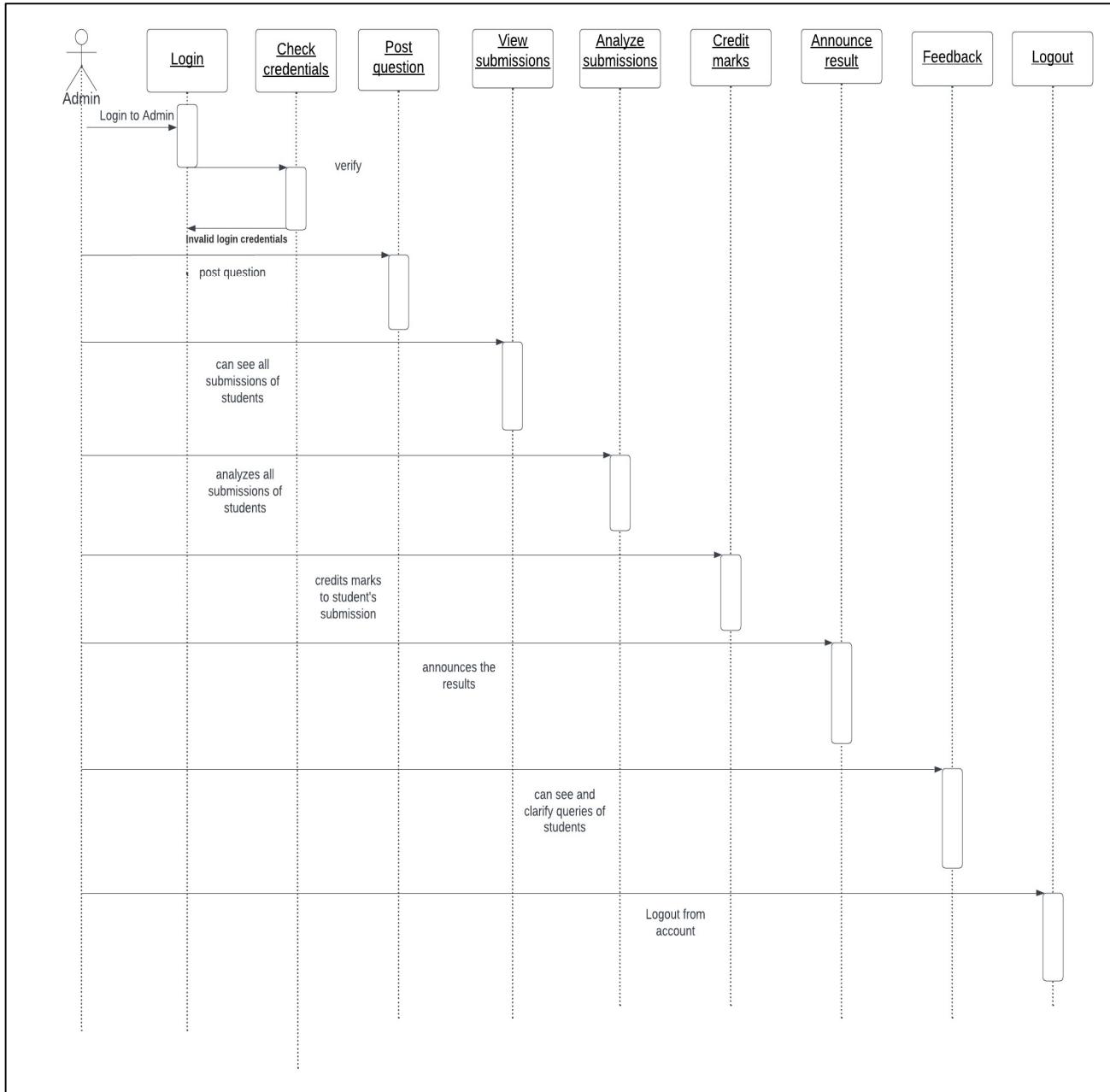
**Fig 5.1: Use Case Diagram**

The figure 5.2 shows the class diagram of the system. It represents static view of application. The below figure describes attributes and operations of classes and also the constraints imposed on the system.



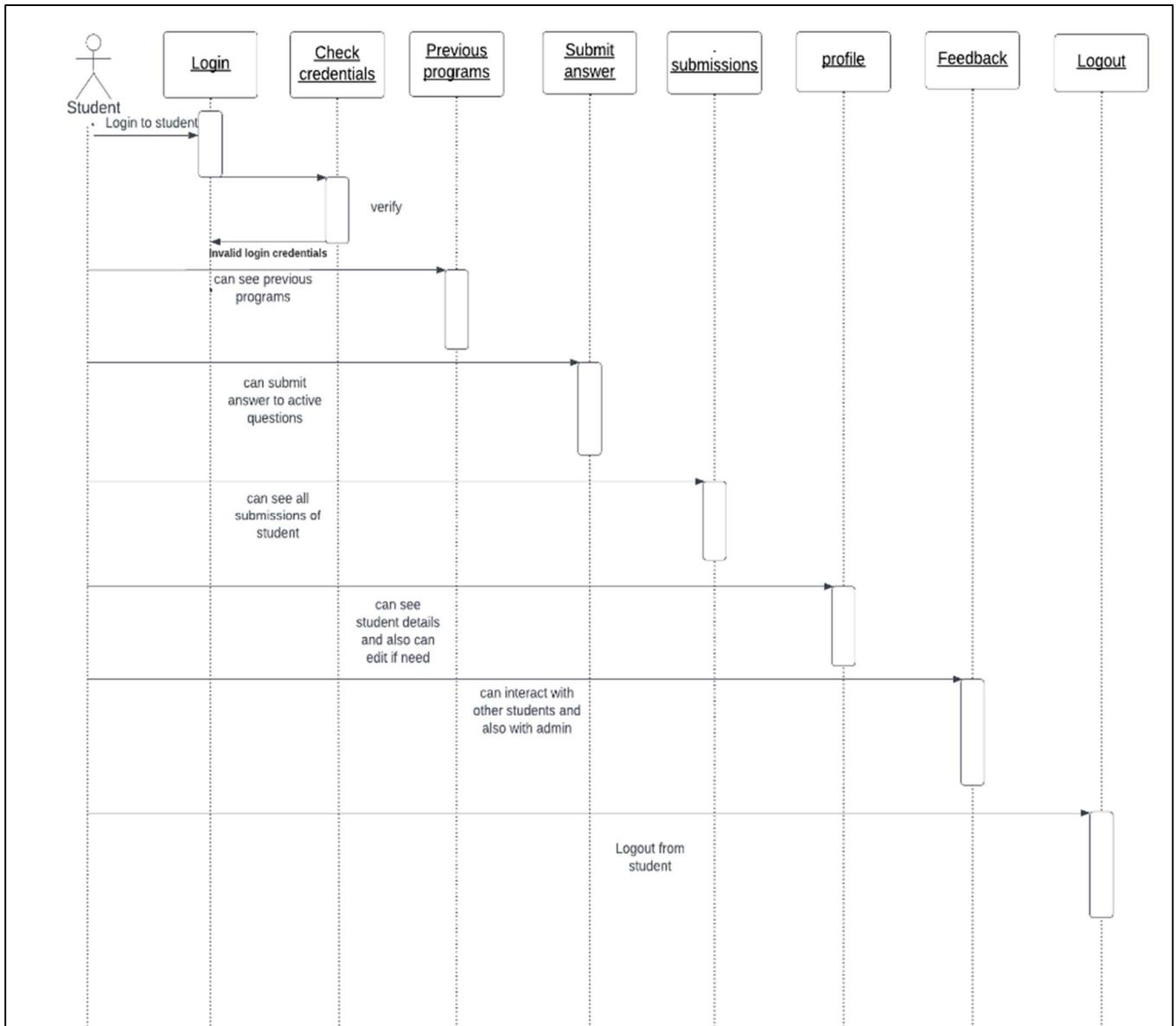
**Fig 5.2: Class Diagram**

The figure 5.3 shows the sequence diagram for admin module of the system. The below figure depicts the processes involved and the sequences of messages exchanged between the processes needed to carry out the functionality in admin module.



**Fig 5.3: Sequence Diagram for Admin**

The figure 5.4 shows the sequence diagram for user module of the system. The below figure depicts the processes involved and the sequences of messages exchanged between the processes needed to carry out the functionality in user module.



**Fig 5.4: Sequence diagram for User**

The figure 5.5 shows the activity diagram for admin module of the system. The below figure basically represents the flow from one activity to another activity in admin module. In the below figure each activity can be described as an operation of admin module of the system.

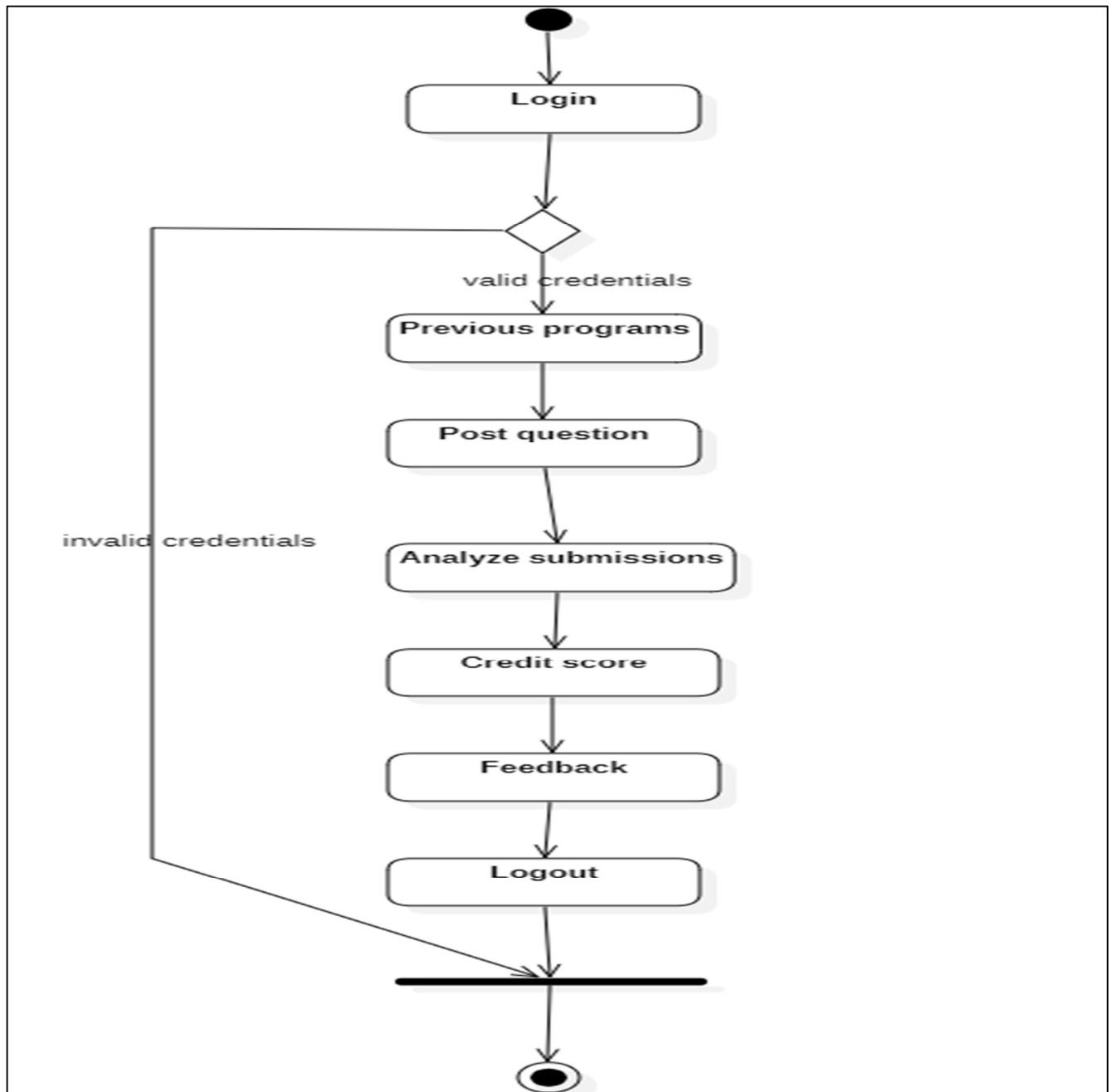
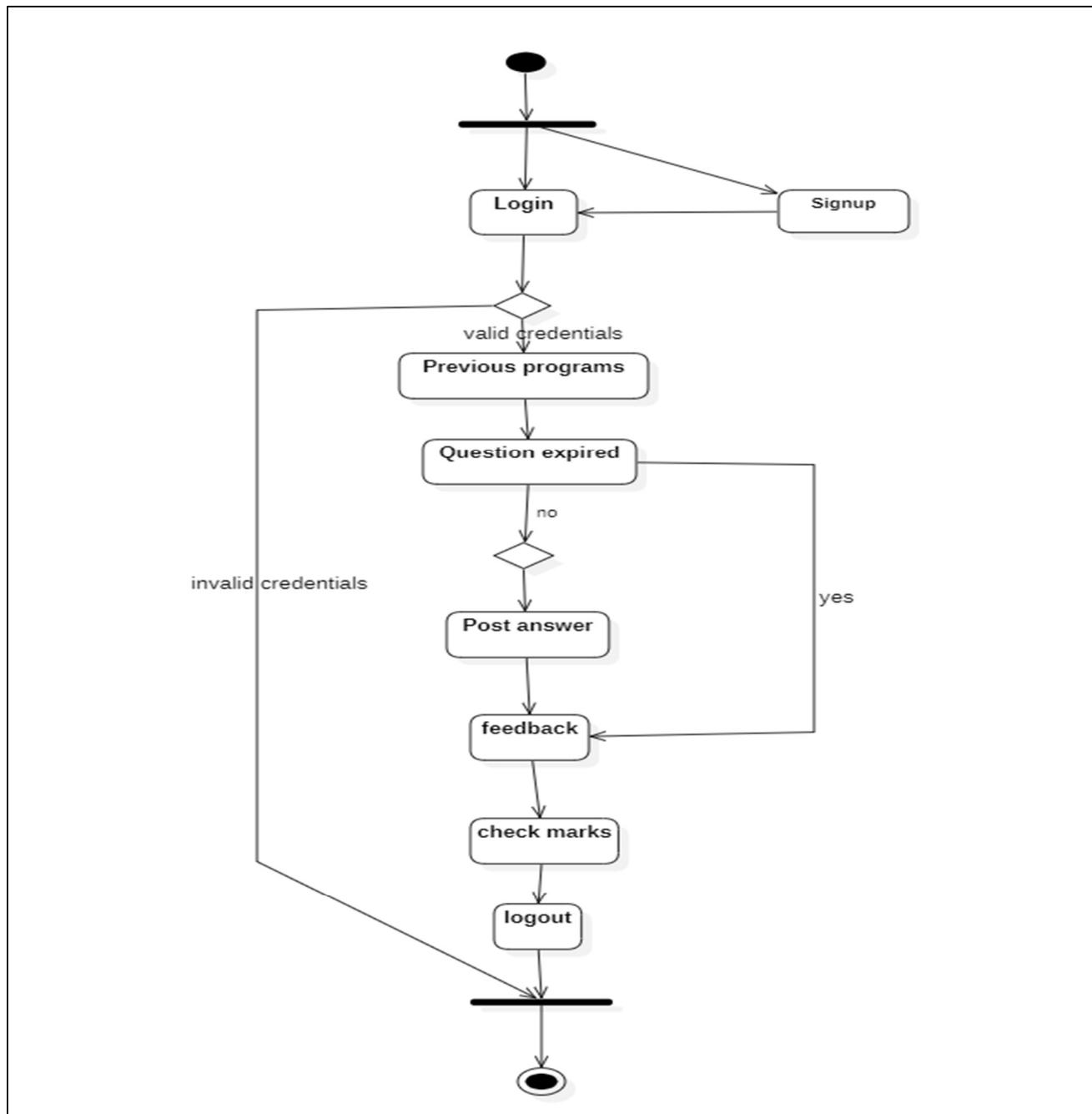


Fig 5.5: Activity Diagram for Admin

The figure 5.6 shows the activity diagram for user module of the system. The below figure basically represents the flow from one activity to another activity in user module. In the below figure each activity can be described as an operation of user module of the system.



**Fig 5.6: Activity Diagram for user**

## **6. SYSTEM IMPLEMENTATION**

### **6.1 Technologies**

- **HTML**

HTML or Hyper Text Markup Language is the main markup language for creating web pages and other information that can be displayed in a web browser. HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>), within the web page content. HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent empty elements and so are unpaired, for example <img>. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, further tags, comments and other types of text-based content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML elements form the building blocks of all websites.

HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behavior of HTML webpages.

- **CSS**

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation. CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design). CSS can also allow the same markup page

to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. However, if the author or the reader did not link the document to a specific style sheet the default style of the browser will be applied. CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities or weights are calculated and assigned to rules, so that the results are predictable.

## • JAVA SCRIPT

JavaScript (JS) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side programming, game development and the creation of desktop and mobile applications. JavaScript is a prototype- based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multi- paradigm language, supporting object-oriented, imperative, and functional programming styles. The application of JavaScript to use outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the popularity of JavaScript for server-side web applications. On the client side, JavaScript was traditionally implemented as an interpreted language but just-in-time compilation is now performed by recent (post-2012) browsers.

- **MYSQL**

MySQL ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open-source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open-source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

### 6.1.1 PHP

- **PHP**

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1995, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Preprocessor, a recursive backronym. PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

PHP was originally created by Rasmus Lerdorf in 1995. The main implementation of PHP is now produced by The PHP Group and serves as the PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. As of January 2013, PHP was installed on more than 240 million websites (39% of those sampled) and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1994, the reference implementation of PHP (powered by the Zend Engine) is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Preprocessor, which is a recursive backronym.

PHP code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. PHP code is usually processed by a PHP interpreter, which is usually implemented as a web server's native module or a Common Gateway Interface (CGI) executable. After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web page – for example, PHP code can generate a web page's HTML code, an image, or some other data. PHP has also evolved to include a command-line interface (CLI) capability and can be used in standalone graphical applications.

**The main features of the PHP scripting language include the following:**

**Open Source:**

PHP is an open-source language and is freely available for use. The community of open-source PHP developers provides technical support and is constantly improving updating the core PHP functionalities. The PHP Extension and Application Repository system provides and maintains a library of PHP code packages that are available for use. The packages can include functions such as authentication, caching, destructors, encryption, error handling etc.

**Compatibility:**

PHP provides high compatibility with leading operating systems and web servers such as thereby enabling it to be easily deployed across several different platforms.

**File Handling:**

PHP can be used to read text and generate files in various formats such as PDF and XML. Using the file manipulation functions, files and documents can be uploaded and stored on the server. The uploaded documents can be accessed and manipulated through PHP code. PHP can be used to access flat files and perform basic file and directory maintenance tasks thereby enabling files/documents to be edited remotely.

**Improved Performance:**

The PHP compiler includes features to optimize and improve the quality of compiled code by reducing the execution time of the code thereby leading to improved performance.

**Debuggers:**

Several debuggers are available with PHP enabling developers to identify and analyze the code for potential bugs and bottlenecks.

**Sessions:**

PHP provides extensive session and cookie management features and functions enabling the creation and development of personalized web pages.

**Graphics:**

PHP can be used to generate images and graphics dynamically. Using the image functionalities available with PHP, the header information of images can be accessed and manipulated. The GD library of PHP includes a host of features and functionalities that can be used to create images in various formats such as gif, jpeg and png.

**Extensible:**

The source code of PHP can be modified to include custom created extensions and components thereby increasing its extensibility.

## Advantages:

PHP application development offers the following benefits and advantages over application development in other server-side scripting languages:

- PHP is available under the open-source license thereby making it a cost-effective option.
- Speeds the load time and improves browsing experience as processing happens on the server.
- Major open-source packages are written in PHP thereby enabling easy customization and development.
- Compatible with all major operating systems, web servers and browsers.
- Includes a host of in-built features that simplify programming of common tasks.
- Corporate websites
- Intranet applications

Hypertext Preprocessor (PHP) is a cross platform open-source server-side scripting language that can be used to generate dynamic and interactive web pages. PHP scripts or codes can be embedded within HTML pages. When a request is sent for a PHP page, the server parses and executes the code and returns the output as plain HTML. PHP scripts can be written to interact with databases and include a host of database connectivity features providing support for a wide range of platforms such as MySQL, Microsoft SQL Server, Oracle, PostgreSQL etc. PHP can be used to create web applications ranging from personal websites to e-commerce applications and community web portals i.e., discussion forums, blogs etc. Apart from web applications and server-side scripting, PHP can be used for command-line scripting and client-side GUI application development.

PHP is based on an Object-Oriented Architecture and most of its features, concepts and syntax are based on the C and Perl programming languages. Being an open-source language, a large number of libraries and extensions, to extend its core functionalities, are available for download. PHP extensions include support for features such as XML parsing, compression utilities, dynamic generation of images, translation functions etc.

PHP scripts can run across operating systems such as Linux, Windows, Solaris, Open BSD, Mac OSX etc. and also provide support for all major web servers such as Apache, IIS, I Planet etc.

## 6.2 Sample Code

### Index.php:

```
<?php  
session_start();  
include "connection.php";  
?>  
<html>  
<head>  
<title>Coder Blog</title>  
<link rel="stylesheet" type="text/css" href="style.css">  
<meta charset="utf-8">  
<meta name=viewport content="width=device-width, initial-scale=1">  
<linkrel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-  
awesome.min.css">  
<style type="text/css">  
nav {  
    float: right;  
    word-spacing: 30px  
    padding: 20px;  
}  
/* to display in one line of navigation of list */  
nav li {  
    display: inline-block;
```

```
}
```

```
. logo h1{
```

```
    color: white;
```

```
    margin-left:42%;
```

```
}
```

```
. index_image. box h1{
```

```
    color: #d6481c;
```

```
    text-align: center;
```

```
    font-size:30px;
```

```
}
```

```
. index_image. box h3{
```

```
    color: #d6481c;
```

```
    text-align: right;
```

```
    margin-top: -50px;
```

```
    font-size:15px;
```

```
    margin-top: 10px;
```

```
    margin-right: :10px;
```

```
}
```

```
section {
```

```
    font-family: courier, arial;
```

```
    background-attachment: fixed;
```

```
    background-position: absolute;
```

```
    background-size: cover;
```

```
}

.index_image.box {
    height:420px;
    width:750px;
    background-color: #f7f7bd;
    margin:23px auto;
    opacity:0.7;
    border-radius: 50px;
}

h2 {
    text-align: center;
    margin-top:10px;
    color: black;
    font-size:27px;
}

#Demo {
    text-align: center;
    font-size: 35px;
    margin-top: 20px;
}

.wrapper {
    height:580px;
    width:100%;
```

```
}

/*for header block */

header {

    height:90px;

    width:100%;

    background-color: #99ceb1;

}

section {

    height:410px;

    width:100%;

}

footer {

    height:80px;

    width:100%;

    background-color: #99ceb1;

}

body {

    width:99%;

    height: auto;

}

.details {

    color: white;

    margin-top: 160px;
```

```
margin-left:15px;  
}  
  
a {  
background-color: #99ceb1;  
opacity:0.9;  
padding: 0px 7px;  
text-align: center;  
color: black;  
text-decoration: none;  
display: inline-block;  
}  
  
a: hover, a: active {  
background-color: #f7f7bd;  
opacity:0.7;  
color: black;  
}  
  
body {  
background-image:url('images/mbg1.jpg');  
background-repeat: no-repeat;  
background-size: cover;  
}  
  
</style>  
</head>
```

```

<body>

<?php

$p=mysqli_query ($db,"SELECT due_date, due_time FROM post_question WHERE trend=1;");

$h2=0;

if ($p!=false) {

    $h2=mysqli_num_rows($p);

}

if ($h2!=0) {

    $row=mysqli_fetch_assoc($p);

    $pp=$row['due_date']." ".$row['due_time'];

?

<! -----countdown timer----->

<script>

// Set the date we're counting down to

var countDownDate = new Date ("<?php echo $pp? ;>").getTime ();

// Update the count down every 1 second

var x = setInterval (function () {

    // Get today's date and time

    var now = new Date ().getTime ();

    // Find the distance between now and the countdown date

    var distance = countDownDate - now;

    // Time calculations for days, hours, minutes and seconds

    var days = Math.floor(distance / (1000 * 60 * 60 * 24));

```

```

var hours = Math.floor((distance % (1000 * 60 * 60 * 24)) / (1000 * 60 * 60));

var minutes = Math.floor((distance % (1000 * 60 * 60)) / (1000 * 60));

var seconds = Math.floor((distance % (1000 * 60)) / 1000);

// Output the result in an element with id="demo"

document.getElementById("demo").innerHTML = days + "d " + hours + "h "
+ minutes + "m " + seconds + "s ";

// If the countdown is over, write some text

if (distance < 0) {

    clearInterval(x);

    document.getElementById("demo").innerHTML = "TIMED OUT";

}

}, 1000);

</script>

<?php

}

?>

<div class="wrapper">

<header>

<div class="logo">

<?php

echo "<img src='images/pp3.png' width=50 height=40>";

?>

<h1>CODER&nbsp; &nbsp; BLOG</h1>

```

```
</div>

<?php

if(isset($_SESSION['login_user']))

{

?>

<nav>

<ul>

<li><a href="Index.php">HOME</a></li>

<li><a href="logout.php">LOGOUT</a></li>

<li><a>COMMENTS</a></li>

</ul>

</nav>

<?php

}

else {

?>

<nav>

<ul>

<li><a href="index.php">HOME</a></li>

<li><a href="login.php">LOGIN</a></li>

<li><a href="signup.php">SIGN-UP</a></li>

<li><a href="feedback.php">FEEDBACK</a></li>

</ul>
```

```
</nav>

<?php

}

?>

</header>

<section>

<div class="index_image">

<br><br>

<div class="box">

<center>

<h1>&nbsp; &nbsp; WELCOME TO CODER BLOG</h1><br>

</center>

<div style="text-align: center;">

<img src='images/ind_logo.png' width=90 height=80>

</div>

<h1>Today question</h1><br>

<h2>

<?php

$sql="SELECT * FROM post_question WHERE trend=1;";

$res=mysqli_query ($db, $sql);

if ($res!=false) {

$row=mysqli_fetch_assoc($res);

echo $row['question'];


```

```
?>

<p id="demo">TIMER</p>

</h2>

<h3>

<span style="color: black">Due_Date:</span>

<?php

echo $row['due_date']."'<br>.<br> "


?>

<span style="color: black">Due_Time:</span>

<?php

echo $row ['due_time'];

?>

</h3>

<?php

}

?>

</div>

</div>

</section>

<footer>

<div class="details"><br>

Email: &nbsp; &nbsp; admin@gmail.com<br><br>

Phone: &nbsp; &nbsp;6301190308


```

```

    </div>

    </footer>

    </div>

</body>

</html>

```

### 6.3 Database Tables

#### Admin login Table: -

**Table 6.1: Admin Login Table**

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 first_name	varchar(30)	latin1_swedish_ci		No	None		 Change	 Drop 
<input type="checkbox"/>	2 last_name	varchar(30)	latin1_swedish_ci		No	None		 Change	 Drop 
<input type="checkbox"/>	3 username 	varchar(30)	latin1_swedish_ci		No	None		 Change	 Drop 
<input type="checkbox"/>	4 password	varchar(40)	latin1_swedish_ci		No	None		 Change	 Drop 
<input type="checkbox"/>	5 email	varchar(40)	latin1_swedish_ci		No	None		 Change	 Drop 
<input type="checkbox"/>	6 contact	bigint(10)			No	None		 Change	 Drop 
<input type="checkbox"/>	7 image	varchar(50)	latin1_swedish_ci		No	user.jpeg		 Change	 Drop 

The table 6.1 is used to store the details of admin. The fields description of Admin login table is as below

**first\_name:** It consist of first name of admin.

**last\_name:** It consist of last name of admin.

**username:** The unique identifier to identify every admin.

**password:** Password of the admin corresponding to their username.

**email:** Consist of email of the admin.

**contact:** Phone number of the admin.

**image:** It stores the profile image name.

## Post Question Table:

**Table 6.2: Post Question Table**

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	qno 	int(11)			No	None		AUTO_INCREMENT	 Change  Drop  More
2	start_date	varchar(15)	latin1_swedish_ci		No	None			 Change  Drop  More
3	start_time	varchar(15)	latin1_swedish_ci		No	None			 Change  Drop  More
4	due_date	varchar(30)	latin1_swedish_ci		No	None			 Change  Drop  More
5	due_time	varchar(15)	latin1_swedish_ci		No	None			 Change  Drop  More
6	question	text	latin1_swedish_ci		No	None			 Change  Drop  More
7	status	int(11)			No	1			 Change  Drop  More
8	trend	int(11)			No	1			 Change  Drop  More

The table 6.2 maintains the details of questions posted by admin. The fields description of post question table is as below

**qno:** Unique way to identify the each question posted by admin.

**start\_date:** The date at the admin sent the question.

**start\_time:** The time at which the admin sent the question.

**due\_date:** Defines the expired date of the posted question.

**due\_time:** Defines the expired time of the posted question.

**Question:** It stores the question of what the admin posted.

**Status:** It defines the whether the it is available to post answers or not by using 0 and1.

**Trend:** It defines the trending question or the last question what has been sent.

**Student login Table:****Table 6.3: Student Login Table**

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
1	first_name	varchar(40)	latin1_swedish_ci		No	None			Change	Drop
2	last_name	varchar(40)	latin1_swedish_ci		No	None			Change	Drop
3	username	varchar(40)	latin1_swedish_ci		No	None			Change	Drop
4	password	varchar(30)	latin1_swedish_ci		No	None			Change	Drop
5	email	varchar(30)	latin1_swedish_ci		No	None			Change	Drop
6	contact	bigint(10)			No	None			Change	Drop
7	image	varchar(50)	latin1_swedish_ci		No	user.jpeg			Change	Drop

The table 6.3 is used to maintain student details. The fields description of student login table is as below:

**first\_name:** It consist of first name of student.

**last\_name:** It consist of last name of student.

**username:** The unique identifier to identify every student.

**password:** Password of the student corresponding to their username.

**email:** Consist of email of the student.

**contact:** Phone number of the student.

**image:** It stores the profile image name.

**Submit Answer Table:****Table 6.4: Submit Answer Table**

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
1	qno	int(11)			No	None			Change	Drop
2	date	varchar(15)	latin1_swedish_ci		No	None			Change	Drop
3	time	varchar(15)	latin1_swedish_ci		No	None			Change	Drop
4	username	varchar(20)	latin1_swedish_ci		No	None			Change	Drop
5	pdfname	varchar(50)	latin1_swedish_ci		No	None			Change	Drop
6	visible	int(11)			No	None			Change	Drop
7	marks	float			No	-1			Change	Drop
8	credit	varchar(100)	latin1_swedish_ci		No	None			Change	Drop

The table 6.4 maintains the details of submissions of students. The fields description of submit answer table is as below:

**qno:** It uniquely identifies the each question posted by the admin.

**Date:** Defines the date at which the user send the answer.

**Time:** Defines the time at which the user sent the answer in 24 hours format.

**Username:** The username who send the answer.

**Pdfname:** The name of the pdf posted which consist of answer.

**Visible:** It identifies the whether the answer is visible on previous programs page using 0 and 1 values.

**Marks:** The marks assigned by the admin between 0 to 10.

**credit:** To identify the whether the admin credited the marks or not.

### Feedback Table:

**Table 6.5: Feedback Table**

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	sno	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	rid	int(11)			No	None			Change  Drop  More
3	username	varchar(20)	latin1_swedish_ci		No	None			Change  Drop  More
4	message	text	latin1_swedish_ci		No	None			Change  Drop  More
5	date	varchar(100)	latin1_swedish_ci		No	None			Change  Drop  More
6	time	varchar(100)	latin1_swedish_ci		No	None			Change  Drop  More

The table 6.5 is used to maintain the messages and details of communication in feedback page. The fields description for feedback table is as below:

**sno:** Serial number which is unique used to display the most recent comments at top of section.

**Rid:** To identify the admin(1),user(2) and guest(0) based on the number stored.

**Username:** To know the who sent the comment.

**Message:** Used to store the contents of the message.

**Date:** It defines the at which date the user posted the message.

**Time:** It defines the at what time the user posted the message.

## 6.4 Results:

### Server Start Screen:

The figure 6.1 shows the server start screen of xampp server. This is used to start apache server and MySQL modules.

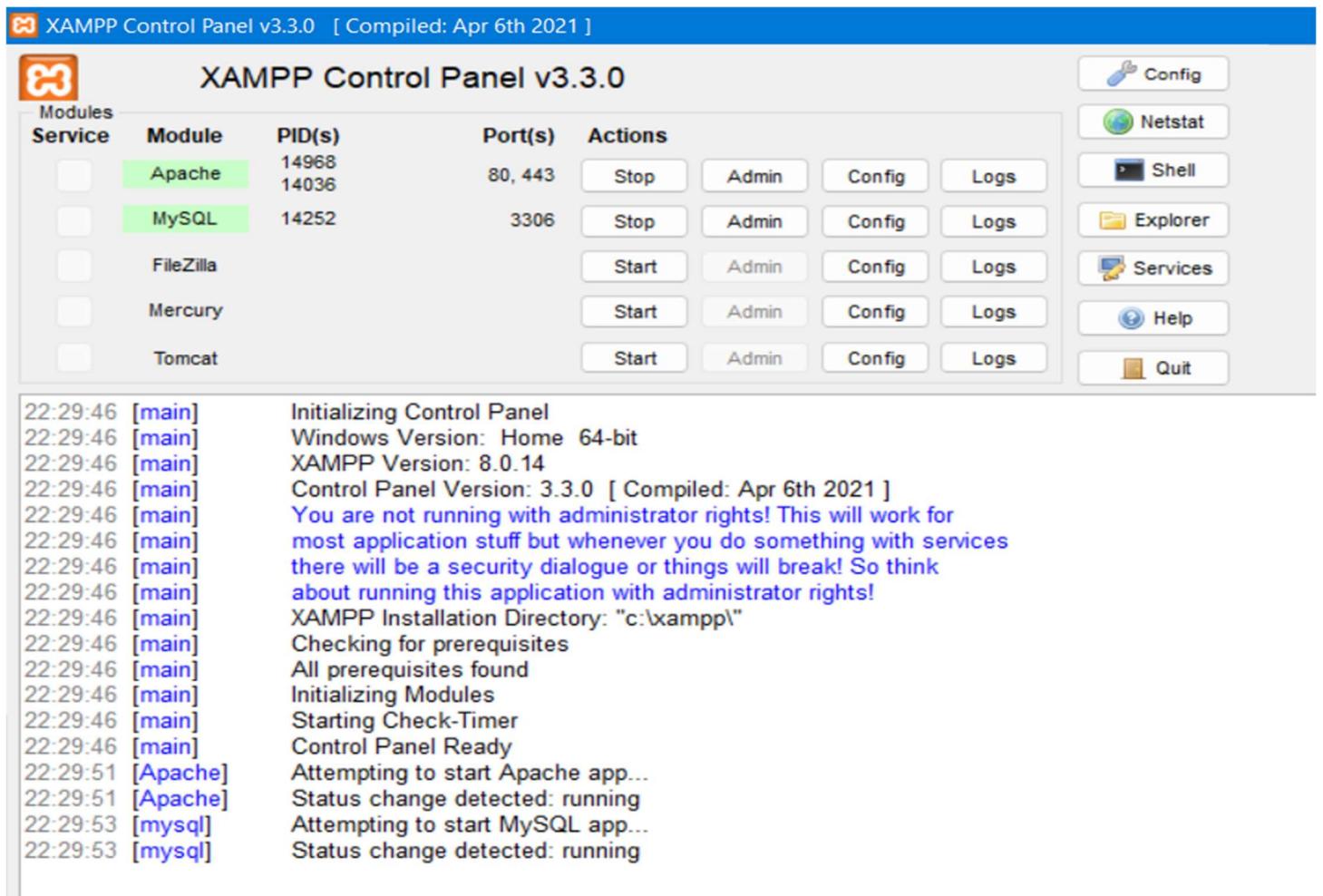
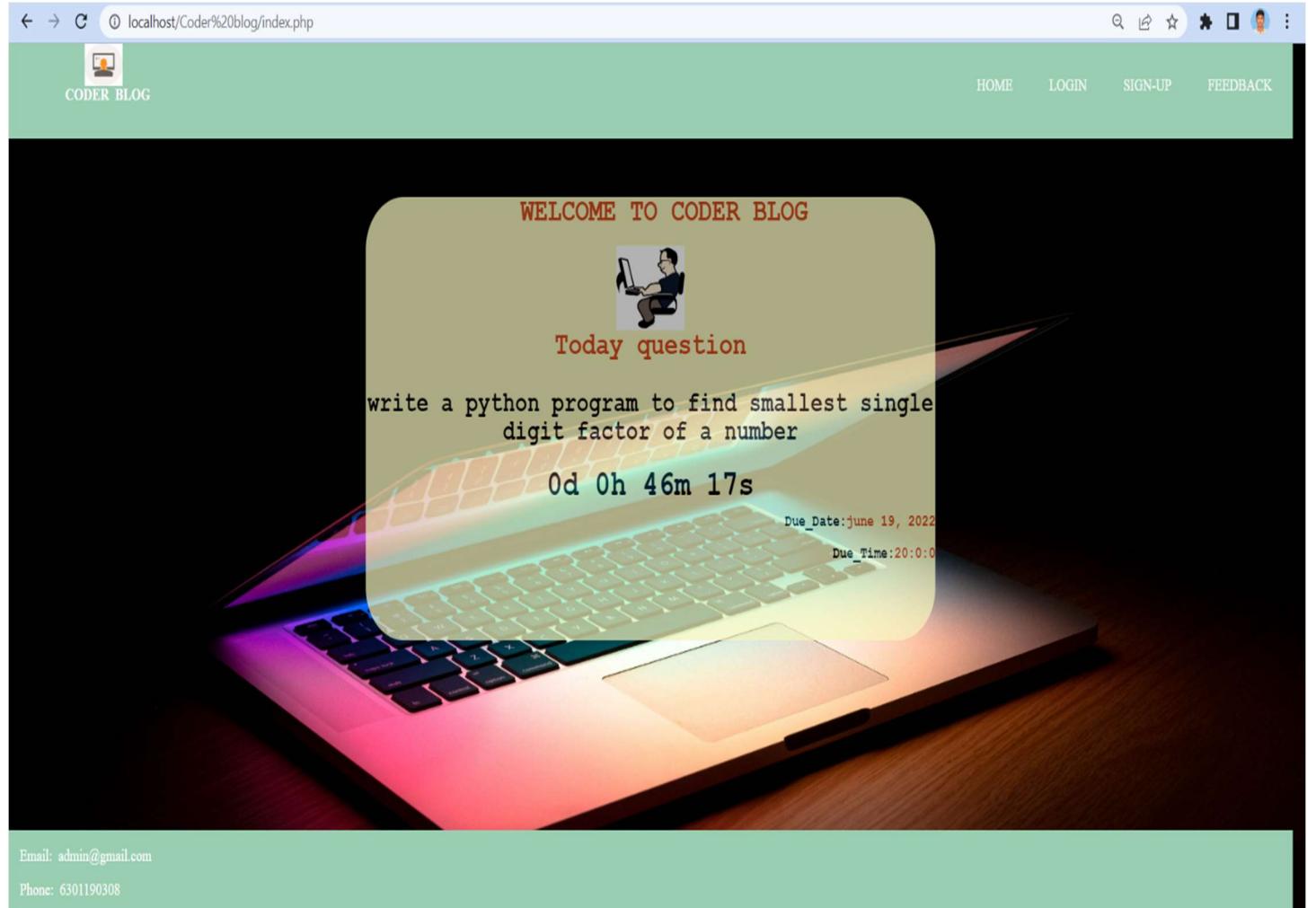


Fig 6.1: XAMPP start screen

**Homepage:**

The figure 6.2 shows the home page of the application. This page is used to display the questions posted by admin and other details of questions as shown in the below figure.

**Fig 6.2: Home Page**

## Login Page:

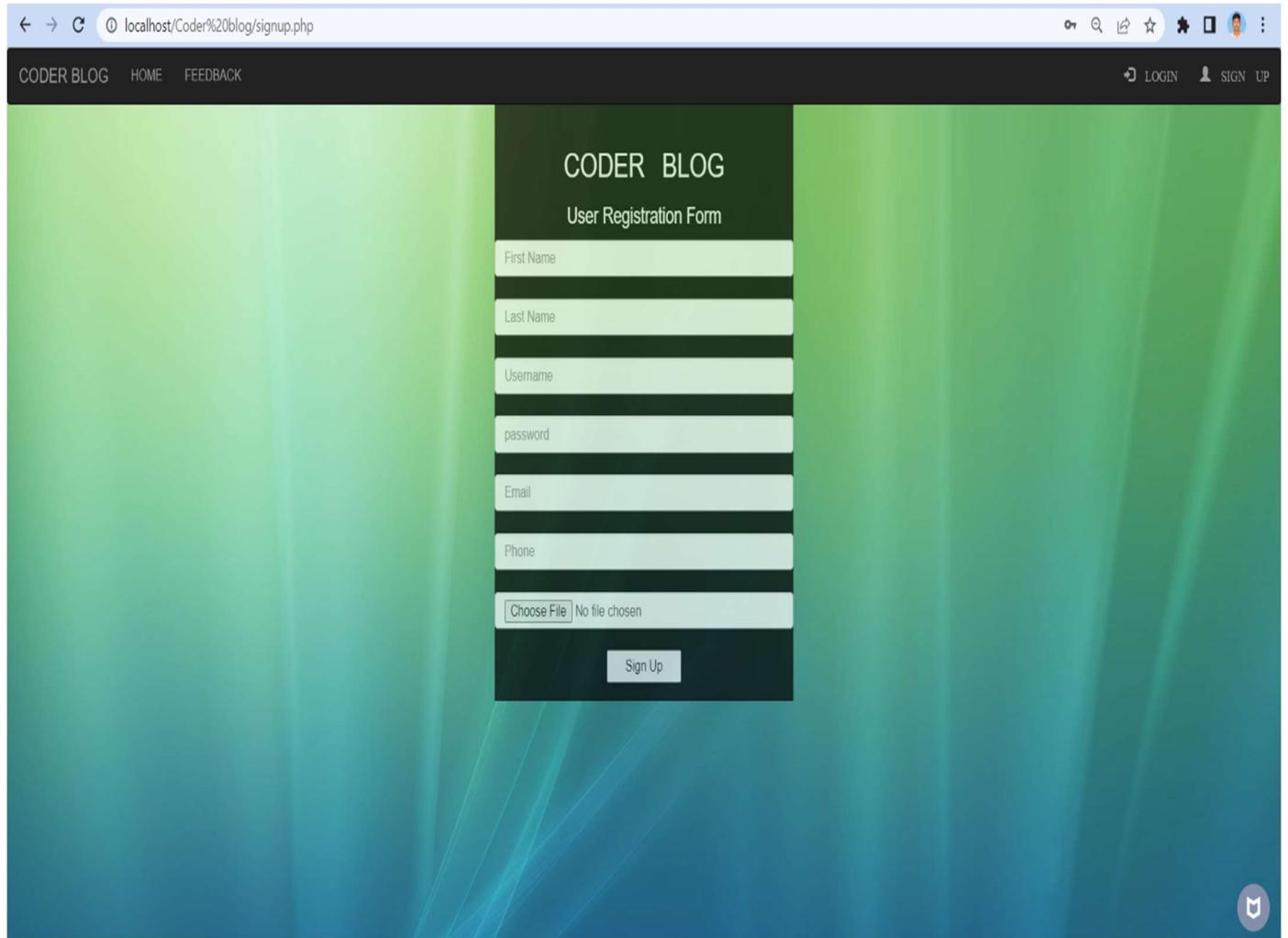
The figure 6.3 shows the login form of the application. The users can enter their credentials to login into their accounts as shown in the below figure.



**Fig 6.3: Login Page**

**Signup Page:**

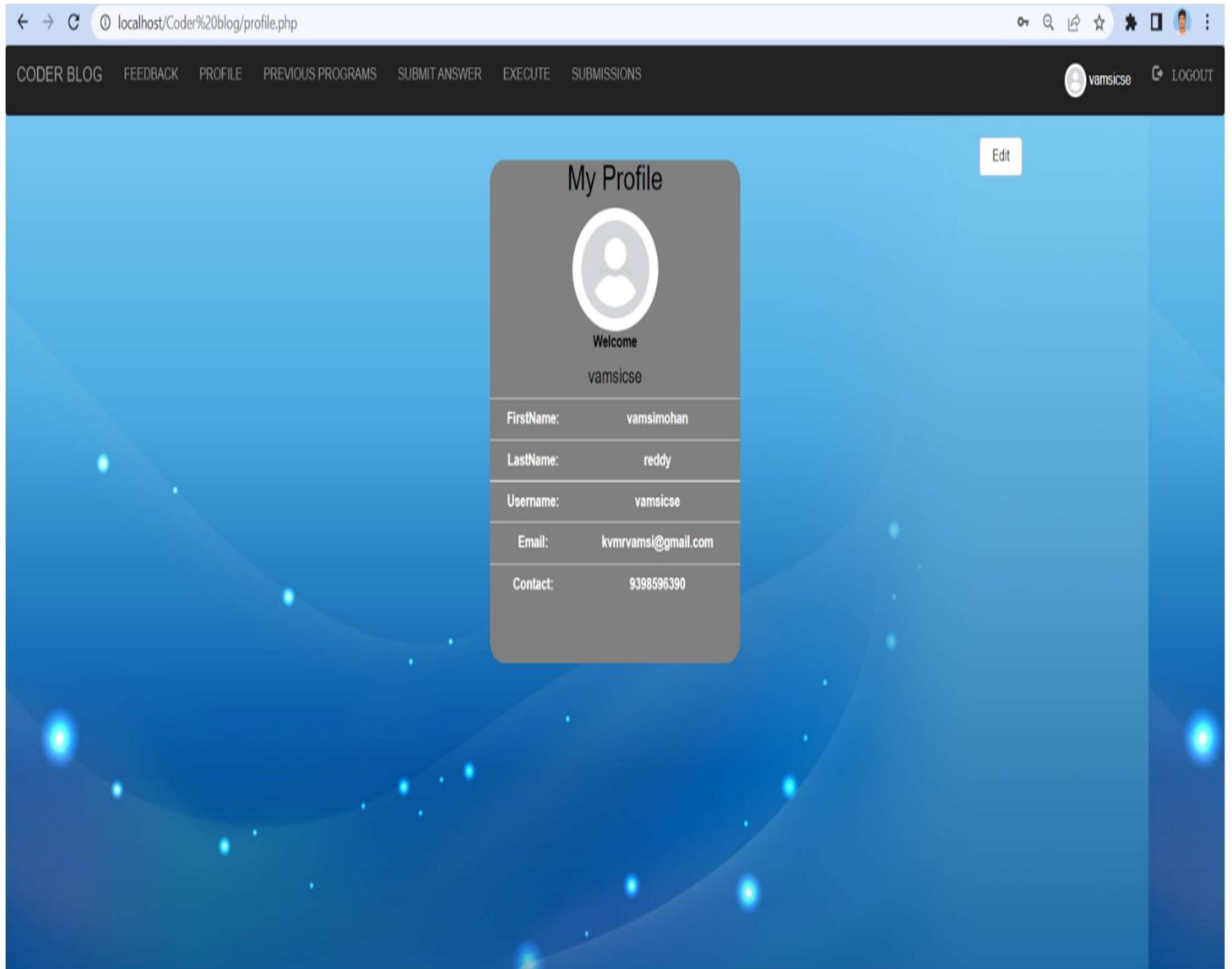
The figure 6.4 shows the registration form for student to create new account. The student can create new account by filling some basic details as shown in the below figure.



**Fig 6.4: Signup Page**

**Profile Page:**

The figure 6.5 shows the profile page of users. After successful login admin or student can view their respective details by using profile link as shown in below figure.

**Fig 6.5: Profile Page**

## Previous programs page:

The figure 6.6 shows the previous programs page of the application. After successful login admin and student are navigated to this page. This page shows all submissions of students with the details of question and username of submitted students as shown in the below figure.

The screenshot displays the 'Previous programs' page from a web application. At the top, there is a navigation bar with links for CODER BLOG, FEEDBACK, PROFILE, PREVIOUS PROGRAMS, POST QUESTION, and PROGRAM SUBMISSION. A user profile for 'mohan' is visible on the right, along with a 'LOGOUT' link. The main content area has a light blue background with a large title 'Previous programs' in blue. Below it, there are two separate code editor windows, each representing a submitted program. The first window is titled '5. write a java program to reverse digits of a number'. It contains the following Java code:

```

1 public class ReverseNumberExample{
2 {
3 public static void main(String[] args)
4 {
5 int number = 387654, reverse = 0;
6 while(number != 0)
7 {
8 int remainder = number % 10;
9 reverse = reverse * 10 + remainder;
10 number = number/10;
11 }
12 System.out.println("The reverse of the given number is: " + reverse);
13 }

```

The second window is titled '4. write a java program to print factorial of a number'. It contains the following Java code:

```

1 class Test
2 {
3     // method to find factorial of given number
4     static int factorial(int n)
5     {
6         if (n == 0)
7             return 1;
8         else
9             return n * factorial(n - 1);
10    }
11 }

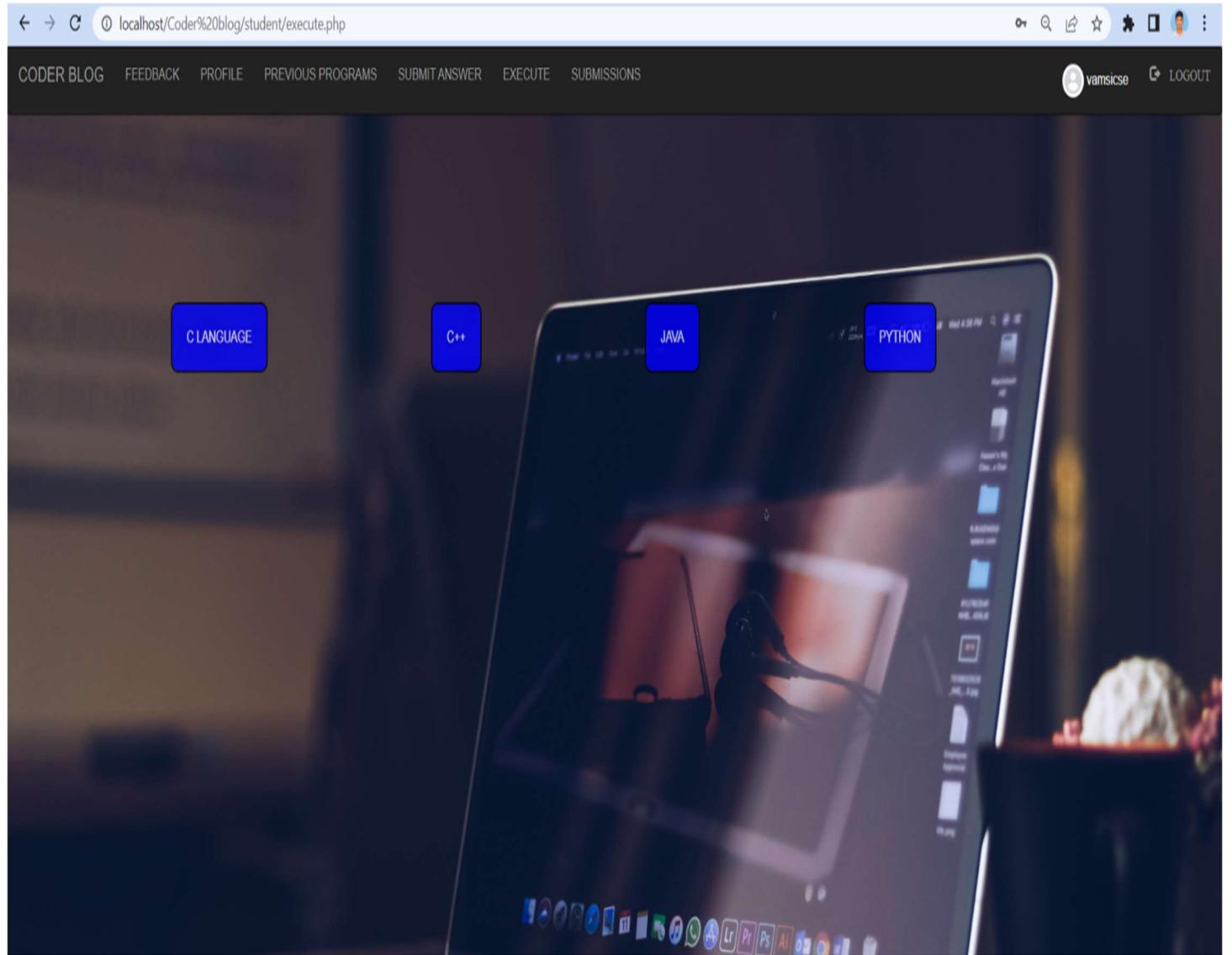
```

A small blue bar at the bottom right of the page says 'Username: vamsicse'.

**Fig 6.6: Previous Programs Page**

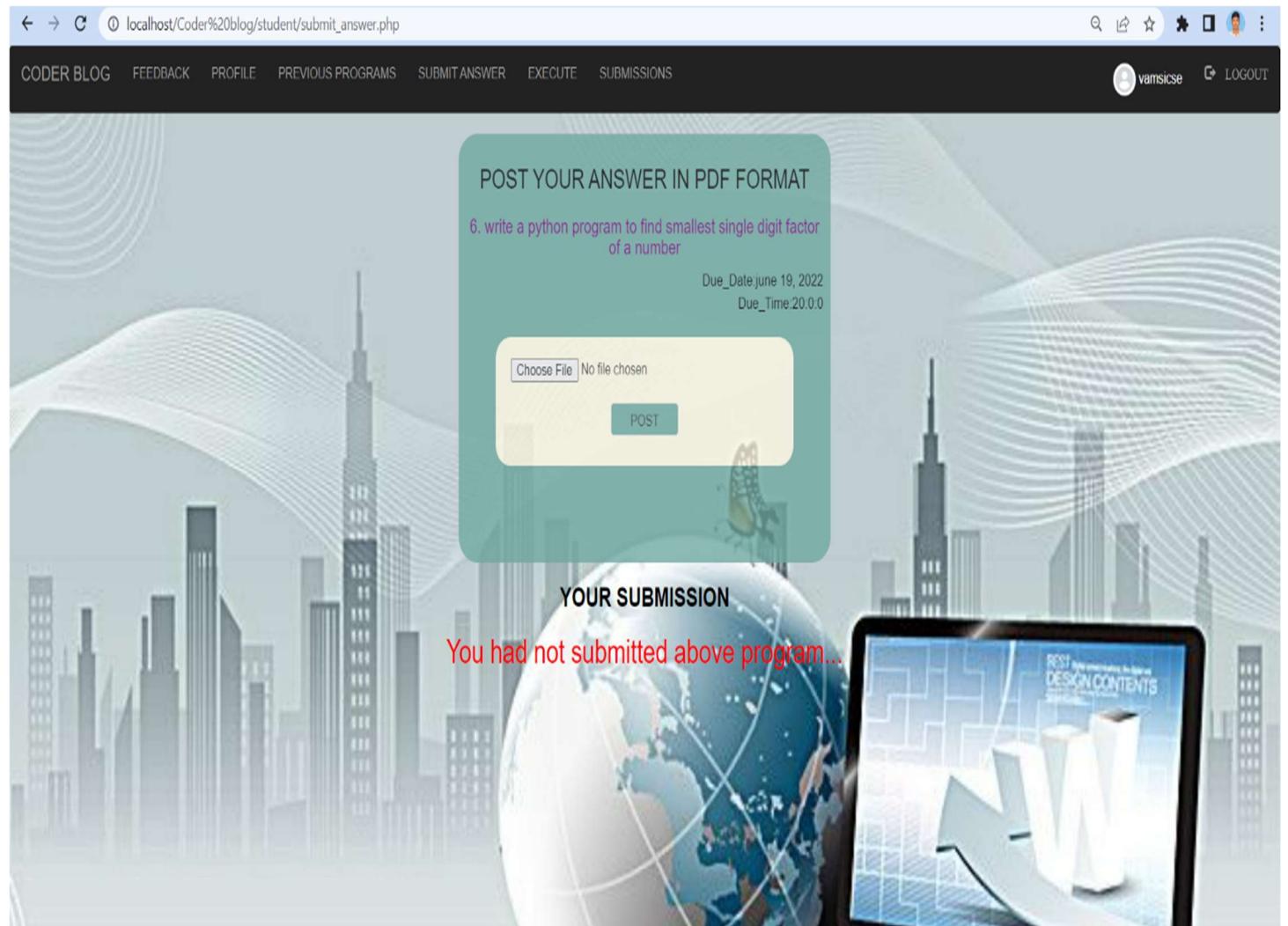
**Execution Page:**

The figure 6.7 shows the execution page for student. After successful login students can execute the program posted by admin by clicking on button based on specified language. After button click the student will be navigated to online compiler of the specified language.

**Fig 6.7: Execution page**

**Submit Answer Page:**

The figure 6.8 shows the submit answer page of application. After successful login students can submit answer to an active question by uploading a pdf file which consists of solution of that question as shown in the below figure.



**Fig 6.8: Submit Answer Page for User**

## Submitted Answers Page:

The figure 6.9 shows the program submission page of admin module. After successful login admin can view the submissions of students by entering their username, based on username the submission will be displayed.

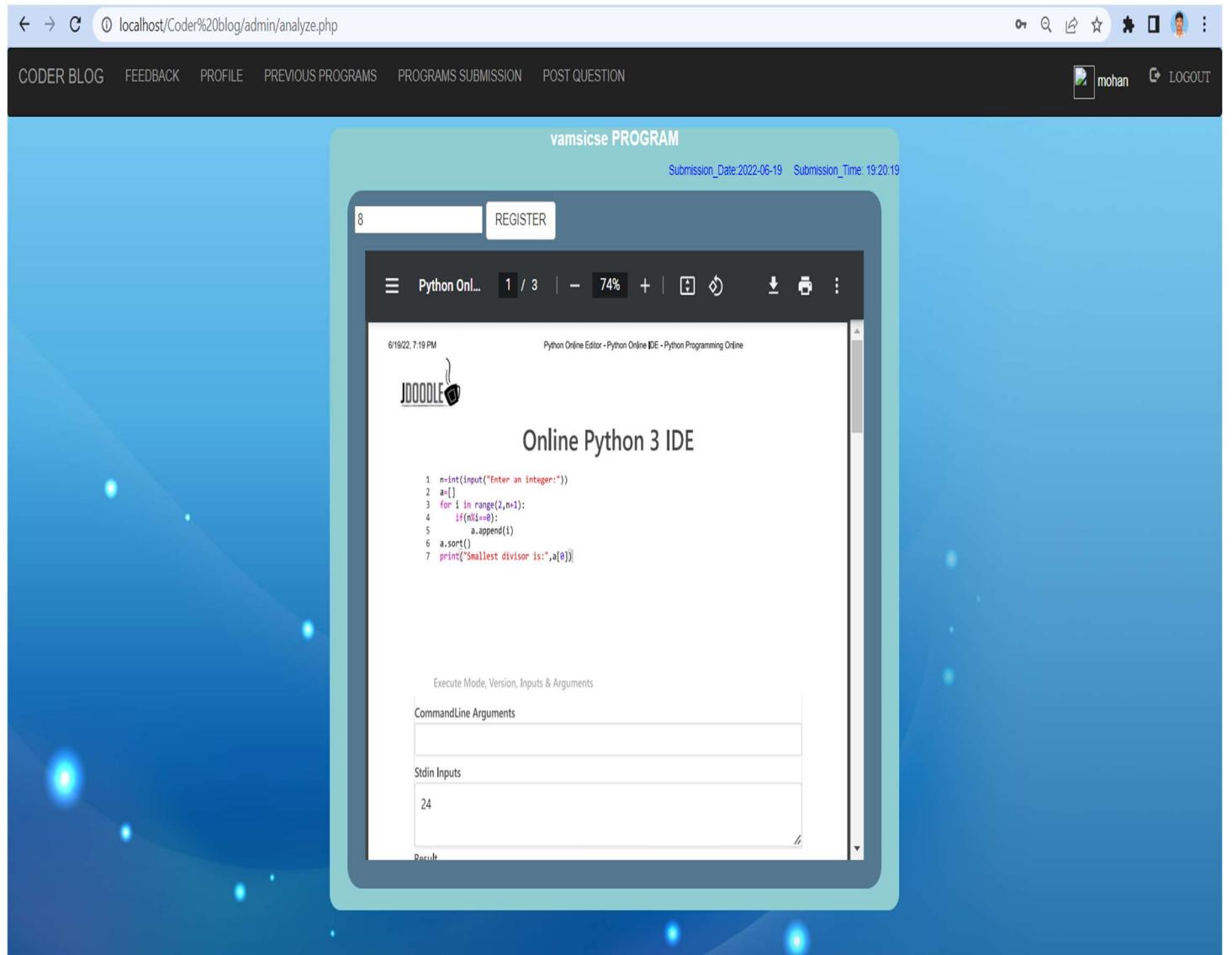
The screenshot displays a web application interface for managing program submissions. At the top, a navigation bar includes links for Coder Blog, Feedback, Profile, Previous Programs, Programs Submission, Post Question, and Logout. A user profile for 'mohan' is shown on the right. Below the navigation, a button labeled 'ALL SUBMISSIONS' is highlighted in red. The main content area is titled 'PROGRAM SUBMISSIONS' and contains a question: '6. write a python program to find smallest single digit factor of a number'. It specifies a due date of June 19, 2022, and a due time of 20:00. A table lists the submitted programs, with one entry for 'vamsicse' showing details like Sub Date (2022-06-19), Sub Time (19:20:19), PDF Name (sf.pdf), Credits (CREDITED), and Marks (9). Buttons for 'SHOW' and 'ANNOUNCE RESULT' are also visible.

USERNAME	SUB_DATE	SUB_TIME	PDF_NAME	CREDITS	MARKS
vamsicse	2022-06-19	19:20:19	sf.pdf	CREDITED	9

**Fig 6.9: Submitted Answers Page**

## Analyze Page:

The figure 6.10 shows the analyse page of admin module. After successful login admin can view the submissions of students by entering their username, based on username the submission will be displayed. Then admin analyses the submission credit marks as shown in the below figure.



**Fig 6.10: Analyse page**

## Submissions Page:

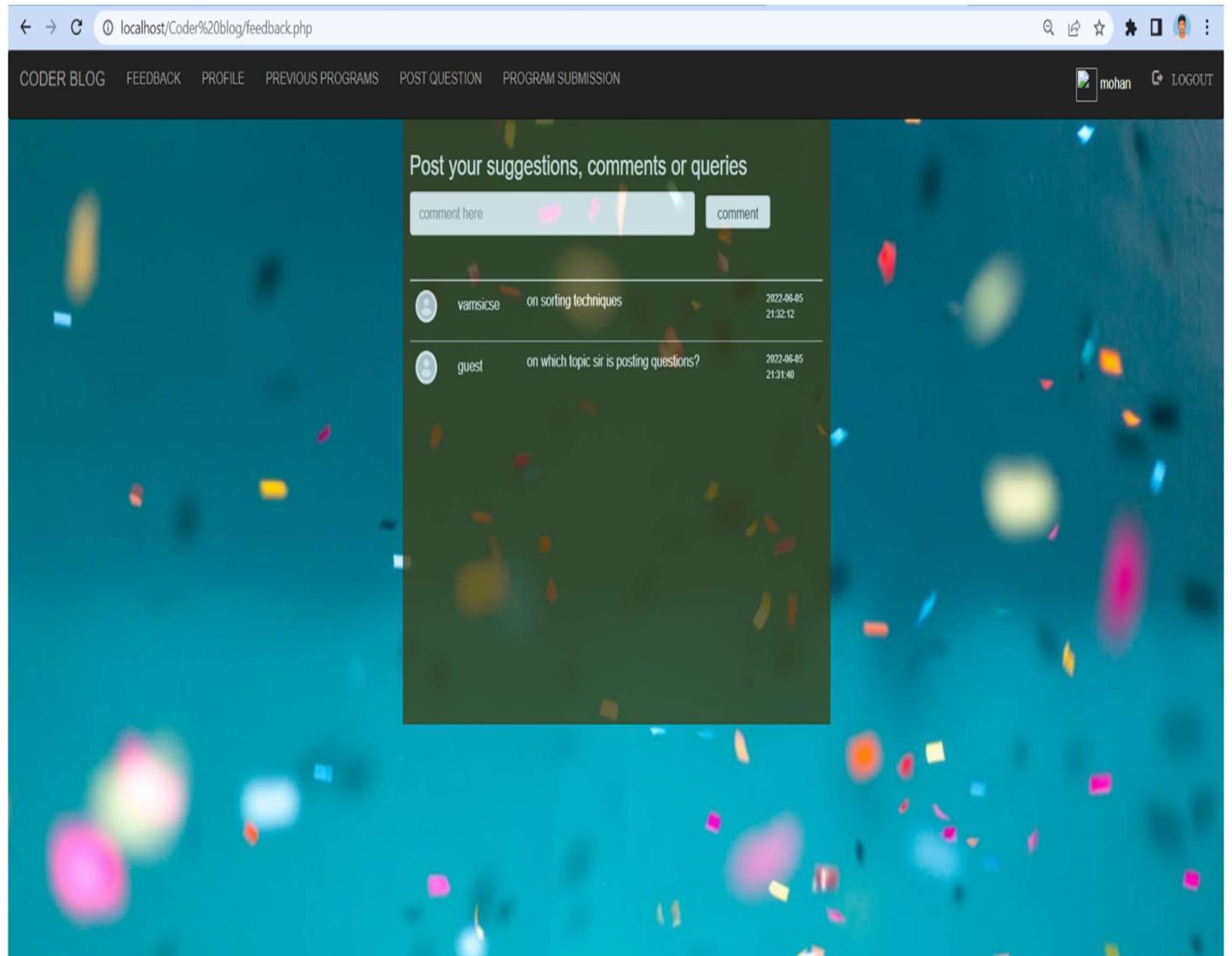
The figure 6.11 shows the submissions page of student module. After successful login students can view their submissions and also status of their submission by the credit label. This page shows submission details of student in a detailed way as shown in the below figure.

QUESTION NO.	SUBMISSION_DATE	SUBMISSION_TIME	PDF_NAME	MARKS	CREDIT
6	2022-06-19	19:20:19	sf.pdf	-1	NOT-CREDITED
5	2022-06-09	21:10:17	reverse.pdf	8	CREDITED
4	2022-05-30	20:53:42	factorial.pdf	8	CREDITED
3	2022-05-23	19:33:54	insertion.pdf	8	CREDITED
1	2022-05-22	12:08:11	bubblesort.pdf	-1	NOT-CREDITED

Fig 6.11: Submissions Page for User

**Feedback page:**

The figure 6.12 shows the feedback page of the application. This page is used for communication between users of application. Users can post their doubts and other users can respond to that eventually. If they post the comments after login the respective username will be displayed otherwise it will be displayed as guest user.

**Fig 6.12: Feedback Page**

## **7. SYSTEM TESTING**

### **7.1 Test case Description:**

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also.

#### **Psychology of Testing:**

The aim of testing is often to demonstrate that a program works by showing that it has no errors. The basic purpose of testing phase is to detect the errors that may be present in the program. Hence one should not start testing with the intent of showing that a program works, but the intent should be to show that a program doesn't work. Testing is the process of executing a program with the intent of finding errors.

#### **Testing Objectives:**

The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time. Stating formally, we can say,

- Testing is a process of executing a program with the intent of finding an error.
- A successful test is one that uncovers an as yet undiscovered error.
- A good test case is one that has a high probability of finding error, if it exists.
- The tests are inadequate to detect possibly present errors.
- The software more or less confirms to the quality and reliable standards.

#### **Levels of Testing:**

In order to uncover the errors, present in different phases we have the concept of levels of testing.

#### **System Testing:**

The philosophy behind testing is to find errors. Test cases are devised with this in mind. A strategy employed for system testing is code testing.

## Code Testing:

This strategy examines the logic of the program. To follow this method, we developed some test data that resulted in executing every instruction in the program and module i.e., every path is tested. Systems are not designed as entire nor are they tested as single systems. To ensure that the coding is perfect two types of testing is performed or for that matter is performed or that matter is performed or for that matter is performed on all systems.

## 7.2 Types of Testing

- Unit Testing
- Link Testing

### 7.2.1 Unit Testing:

Unit testing focuses verification effort on the smallest unit of software i.e., the module. Using the detailed design and the process specifications testing is done to uncover errors within the boundary of the module. All modules must be successful in the unit test before the start of the integration testing begins.

In this project each service can be thought of a module. There are so many modules like Login, Registration, View Profile, Search Name, Edit Profile and Image Uploading. Giving different sets of inputs has tested each module. When developing the module as well as finishing the development so that each module works without any error. The inputs are validated when accepting from the user.

In this application developer tests the programs up as system. Software units in a system are the modules and routines that are assembled and integrated to form a specific function. Unit testing is first done on modules, independent of one another to locate errors. This enables to detect errors. Through this error resulting from interaction between modules initially avoided.

### 7.2.2 Link Testing:

Link testing does not test software but rather the integration of each module in system. The primary concern is the compatibility of each module. The Programmer tests where modules are designed with different parameters, length, type etc.

### **7.2.3 Integration Testing:**

After the unit testing, we have to perform integration testing. The goal here is to see if modules can be integrated properly, the emphasis being on testing interfaces between modules. This testing activity can be considered as testing the design and hence the emphasis on testing module interactions.

In this project integrating all the modules forms the main system. When integrating all the modules I have checked whether the integration effects working of any of the services by giving different combinations of inputs with which the two services run perfectly before Integration.

### **7.2.4 System Testing:**

Here the entire software system is tested. The reference document for this process is the requirements document, and the goals to see if software meets its requirements.

Here entire ‘ATM’ has been tested against requirements of project and it is checked whether all requirements of project have been satisfied or not.

### **7.2.5 Acceptance Testing:**

Acceptance Test is performed with realistic data of the client to demonstrate that the software is working satisfactorily. Testing here is focused on external behavior of the system; the internal logic of program is not emphasized.

Test cases should be selected so that the largest number of attributes of an equivalence class is exercised at once. The testing phase is an important part of software development. It is the process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

### **7.2.6 White Box Testing:**

This is a unit testing method where a unit will be taken at a time and tested thoroughly at a statement level to find the maximum possible errors. I tested step wise every piece of code, taking care that every statement in the code is executed at least once. The white box testing is also called Glass Box Testing.

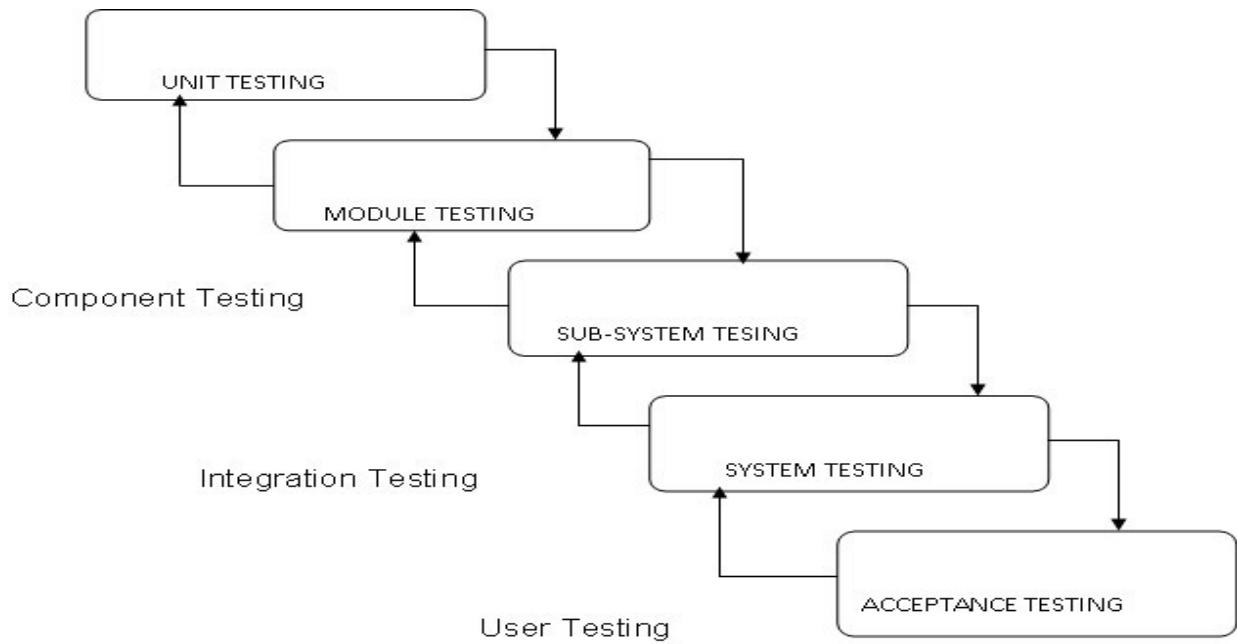
I have generated a list of test cases, sample data which is used to check all possible combinations of execution paths through the code at every module level.

### 7.2.7 Black Box Testing:

This testing method considers a module as a single unit and checks the unit at interface and communication with other modules rather getting into details at statement level. Here the module will be treated as a block box that will take some input and generate output. Output for a given set of input combinations are forwarded to other modules.

Ping is a computer network tool used to test whether a particular host is reachable across an IP network. Ping works by sending ICMP packets to the target host and listening to ICMP replies. Using interval timing and response rate, ping estimates the round-trip time and packet loss rate between hosts.

The below figure 7.1 shows the testing cycle which depicts the sequence of testing methods that has to be performed on the system.



**Fig 7.1: Testing Cycle**

## **8. CONCLUSION**

This website provides a computerized version of coder blog system which will benefit the students as well as the faculty of the college/university. It makes entire process online where students can submit their assignments, works related to programming language or other. It facilitates the timer to submit the work within the given limit which is entered by the admin. Admin can be able to easily check who has submitted the answers with the details of when they submitted. Admin can be easily recognizing the users who haven't got marks with the label of credit. The system automatically announces the top marks with username and with program by simply clicking the announce result button.

User can be able to see the previous programs which are best and analyses his/her growth based on marks they got. Facilitate to ask question in feedback section for any references.

The “CODER BLOG” system which builds the competitiveness in the user which makes them to improve the programming knowledge.

## **9. FUTURE ENHANCEMENTS**

The project Coder Blog will be more interactive and accurate if some more features can be added to it. The future enhancements includes:

- The feature of online lectures can be included, so that faculty can upload their lectures in the application for students.
- The feature of group chat can also be included, so that students can discuss various issues of engineering.
- The online submission facility can also be enhanced so that this feature makes the system more user-friendly and fulfills the needs of users in the best way possible.

## **BIBLIOGRAPHY**

References for the Project Development are taken from the following Web Sites

- <http://www.w3schools.com>
- <https://www.quora.com>
- <https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/bootstrap.min.css>
- <http://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js>
- <http://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js>
- <http://en.wikipedia.org>
- [www.geeksforgeeks.org](http://www.geeksforgeeks.org)
- [www.tutorialspoint.com](http://www.tutorialspoint.com)
- PHP basic tutorial from YouTube