



IUBAT- International University of Business Agriculture and Technology

GroupName:Confusion-matrix

Submitted To

Rubayea Ferdows

Assistant Professor

Department of Computer Science &Engineering

IUBAT- of International University of Business Agriculture and Technology

Submitted By

SL	Name	ID	Signature
01	Eaqub Mia	19103135	
02	Md. Masud Rana	19203050	
03	Mehedi Hasan	19303001	
04	Fabiha Khan	19303004	
05	Moshiur Rahman	19303008	

Date of Submission: 18-12-2022

Abstract

This project is mainly developed for the project requirement of International University of Business Agriculture and Technology to ensure the development skill of knowledge. We know that Competitive programming is all about solving coding problems using algorithm and data structure. Therefore, coding contests help to improve one's logical and analytical skills. This is why we are creating a system which will help one to do practice and level up their programming skills. This is a very basic system mainly focused on the beginners who are the newbies of this journey. There will be a range of problems, from simple to complex. In this system, there will be a registration panel because users must create accounts in order to solve the problems. Each challenge that the user solves results in a rating.

Letter to Transmittal

18th December,2022

Rubayea Ferdows

Assistant Professor

Department of Computer Science &Engineering

IUBAT- of International University of Business Agriculture and Technology

4 Embankment Drive Road, Sector- 10 , Uttara Dhaka – 1230 ,Bangladesh.

Dear mam,

It is great pleasure for us to submit our proposal report on the topic entitled – Competitive Programming Assessment System. We have prepared this report to fulfill CSC470 course. This report will help you to know about our project .We have tried our best to prepare this report to the required standard.

We are expressing our heartiest gratitude to you to go through this report and make your valuable remarks. It would be very kind of you,If you evaluate our performance regarding our report.

Thank you for your kind assistance.

Yours sincerely,

Group members of Confusion Matrix

DECLARATION

We are the members of confusion matrix are students of BCSE program at IUBAT- of International University of Business Agriculture and Technology. This is to inform you that the report entitled **Competitive Programming Assessment System** has been prepared for partial fulfillment of Our CSC 470 course.

This report contains the field work , which has been performed by all of the group members Confusion Matrix . All module and procedure of this project after proper inspection and internet information.

It has not been submitted or copied elsewhere for any purpose.

.....

Confusion Matrix

Program :BSC in CSE

ACKNOWLEDGEMENT

We would like to express all praise to the Almighty Allah , my creator , sustainer to whom we all must have to return .

The successful complain of this report is the outcome of the contribution of the number for people to whom we are grateful and thanks them from the deep of my heart , so I would like to take this opportunity to thank all those people who helped me preparing this report on **Competitive Programming Assessment System** .

We are very thankful to our respected course instructor Rubayea Ferdaws , Lecture of Dept. of Computer Science & Engineering , to give us the opportunities to do a wonderful fieldwork. We would also like to express heartiest respect for her guidance and all kind of support. Without her support it would not be possible for us prepare this report .

We would like to give special thanks to professor Dr. Abdur Rab Vice Chancellor of IUBAT for giving us the opportunities to continue our study through the wonderful university facilities and support

Last but not best we wonderful to convey my guide to all those people inside the campus an in the field for giving advise suggestion , inspiration to complete this report.

Letter of Authorization

Date : 18th December 2022

Confusion Matrix

Program : BSc in CSE

Dear Group,

I hereby authorized your group to develop **Competitive Programming Assessment System** to fulfill the requirement of the project of course of project of course CSC 470 to complete the degree of Bachelors of Computer Science and Engineering.

I wish you to complete this project successfully as the part of completion to the degree of Bachelors of Computer Science and Engineering.

.....

Rubayea Ferdows

Assistant Professor

Department of Computer Science & Engineering

IUBAT- of International University of Business Agriculture and Technology

4 Embankment Drive Road, Sector- 10 , Uttara Dhaka – 1230 , Bangladesh.

Table of Contents

Contents	Page
Cover Page -----	
Abstract -----	I
Letter of Transmittal -----	II
Student's Declaration -----	III
Acknowledgement -----	IV
Letter of Authorization -----	V
Table of Content -----	VI- VIII
List of figures -----	VIII
List of tables -----	IX-X

Chapter 1 Introduction

1 Introduction of project -----	1
1.1 Project Overview -----	1
1.2 Objective -----	1
1.3 Specific objective -----	1-2
1.4 Limitations -----	2
1.5 System Benefits -----	3
1.6 Selection of project development process model -----	3
1.7 Reason of choose XP Model -----	4
1.8 Feasibility study-----	4

Chapter 2. Requirement Engineering

2.1 Requirement Engineering-----	6
2.1.1 User Requirements for user -----	6

2.1.2 System Requirement for use -----	6
2.1.3 Functional Requirement -----	6
2.1.4 Non- Functional Requirement-----	7
2.2 Use case diagram -----	8

Chapter 3. System planning

3.1 Functions of Proposed System-----	9
3.1.1 Function Description-----	9
3.2 System Project Planning -----	10-11
3.2.1 System Project Estimation-----	12
3.2.2 Function Oriented Metrics-----	12
3.2.3 Function Point Estimation -----	13
3.2.4 Effort Distribution-----	14
3.2.5 Project Schedule Chart -----	14
3.2.6 Cost Estimation -----	15
3.2.6.1 Personnel cost -----	15
3.2.6.2 Hardware cost:	15-16
3.2.6.3 Software Cost	16
3.2.6.4 Other Cost	17

Chapter 4. Risk Management

4.1 Risk Management -----	19
4.2 RMMM Plan	20

Chapter 5: Analysis and design

5.1 Activity Diagram	22
5.1.1 Activity diagram for user.....	22
5.1.2 Activity diagram for admin.....	23
5.1.3 Activity diagram for instructor.....	23-24
5.2 Sequence diagram.....	24
5.3 Class diagram.....	25
5.4 CRC.....	25
5.4.1 CRC for user.....	25
5.4.2 CRC for admin.....	25-26
5.4.3 CRC for instructor.....	26
5.4.3 CRC for problem statement.....	26
5.5 ER diagram.....	27
5.6 Data flow diagram.....	28
5.6.1 Context level DFD.....	29
5.6.2 DFD level 1.....	29
5.6.3 DFD level 2 process 1.....	30
5.6.4 DFD level 2 process 2.....	30
5.6.5 DFD level 2 process 3.....	30
5.6.6 DFD level 2 process 4.....	30
5.6.7 DFD level 2 process 5.....	31
5.6.8 DFD level 2 process 6.....	31

Chapter 6: Database, UI design, Testing

6.1 Database table structure.....	33
6.2 UI design screenshots.....	34
6.3 Testing.....	37
Conclusion.....	38

List of figures

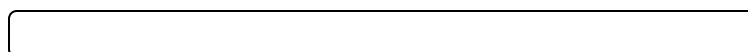
Figure: Software Project Model -----	12
Figure: Use Case Diagram -----	17
Figure: Effort Based Estimation -----	24
Figure: Project Scheduling Chart -----	2
Figure: Activity Diagram -----	9
Figure: Sequence Diagram -----	33
Figure: Class Diagram -----	34
Figure: ERD (Entity Relationship Diagram) -----	36
Figure: 5.6.2 DFD level 1.....	38
Figure: 5.6.3 DFD level 2 process 1.....	38
Figure: 5.6.4 DFD level 2 process 2.....	39
Figure: 5.6.5 DFD level 2 process 3.....	39
Figure: 5.6.7 DFD level 2 process 5.....	40
Figure: 5.6.8 DFD level 2 process 6.....	40
Figure: 6.1 Database table structure.....	42
Figure: UI design screenshots.....	42

List of Tables

Table 3.1 Identifying Complexity for Transition Function.....	22
Table 3.2 Identifying Complexity for Data Function.....	22
Table: 3.3 Performance and environmental impact.....	23
Table: 3.4 Schedule Chart.....	24
Table: 3.5 Personal cost.....	25
Table: 3.6 Hardware Cost.....	25
Table: 3.7 Software Cost.....	26
Table: 3.8 Others Cost.....	26
Table: 5.4.1 CRC for user.....	34
Table: 5.4.2 CRC for admin.....	3
Table: 5.4.3 CRC for instructor.....	35
Table: 5.4.3 CRC for problem statement.....	35

Chapter 01

Introduction



1. Introduction

The software developed basically for getting concern about programming competition from the beginners level programmers. The aim of competitive programming is to write source code of computer programs which are able to solve given problems. A vast majority of problems appearing in programming contests are mathematical or logical in nature. In this system there is some problem statement for beginner programmers. The programmers who want to participate in ICPC. By solving these problems user can improve himself for intermediate level. To solve these problems user need to open an account, so there will be a register panel. Following this user have a login option. After solving each problem user will get a rating. So that's a short story about our project.

We have selected XP software model for this model. As our customer IUBAT CSE department want this software quickly and it's a tiny project also. So for these criteria we have selected the XP(extreme programming model)

1.1 Project Overview

In **Competitive Programming Assessment System** there is some problem statement for beginner programmers. The programmers who want to participate in ICPC. By solving these problems user can improve himself for intermediate level. To solve these problems user need to open an account, so there will be a register panel. Following this user have a login option. After solving each problem user will get a rating. So that's a short story about our project. Its user Friendly its design for beginners level Programmers, it will increase the skills of a Programmer and it will create the opportunities to participation in different organizations programming competitions. This report based on the lab project CSC470, We have completed this project collecting all the data from beginners. In this report, I have described how I developed this system how it will work.

1.2 Objective

The aim of competitive programming is to write source code of computer programs which are able to solve given problems. A vast majority of problems appearing in programming contests are mathematical or logical in nature. In the career objective of the software engineer strong computer science and technology engineering skills have to write in a resume is important. You have the main focus on Programming and mathematics knowledge.

1.3 Specific Objective

- In a Computer Science degree we aim to teach you skills that will be relevant many years in the future.

- The ability to see patterns and to abstract from specific examples to the more general case is crucial
- Being able to think logically so you can predict in advance the behavior of a system working to a fixed set of rules is essential.

1.4 Limitations

Definitely there are some limitation in our system, in this system it does not available detail concept about set topic to the programmer right now, If it could have been implemented the system would become more efficient in the time of competition. We will add this feature also in near future.

1.5 System Benefits

- Developed skills
- Make confidentiality
- Ensure data accuracy
- Security of data
- Greater Efficiency
- Better service
- User friendliness and interactive
- Backup System

1.6 Selection of project development process model

In our system there is some problem statement for beginner programmers . The programmers who wants to participate in ICPC . By solving these problems user can improve himself for intermediate level. To solve these problems user need to open an account , so there will be a register panel . Following this user have a login option . After solving each problem user will get a rating . So that's a short story about our project .

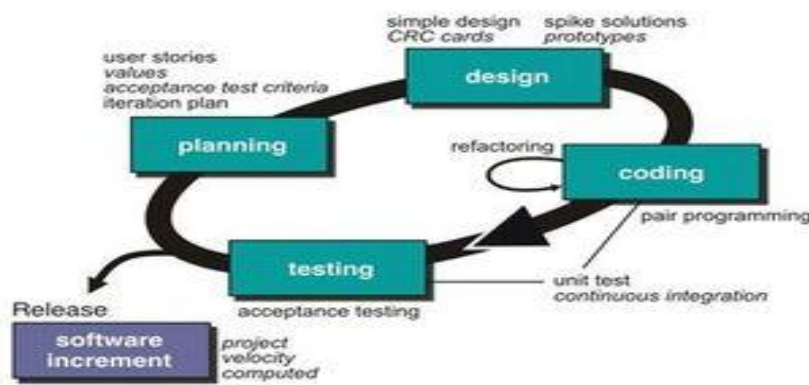


Fig: XP Programming Model

1.7 Reason of choose XP Model

We have selected XP software model for this model. As our customer IUBAT CSE department want this software quickly and it's a tiny project also . So for these criteria we have selected the XP(extreme programming model)

1.8 Feasibility Study

A feasibility study assess the operational, technical economics merit of the proposed project. The feasibility study is initiated to be a preliminary review of the fact to see if it is worthy of proceeding to the analysis phase. From the system analysis analyst perspective the feasibility study is the primary tool for recommending weather to proceed to the next phase or to discontinue the project .The feasibility study is management oriented activity. The object of a feasibility study is to find out if an information system can be done and to suggest alternatives solutions.

Technical Feasibility

For developing this project we need some programming language like

- HTML
- CSS
- Bootstrap
- JavaScript

All the technologies are available. So this project is technically feasible.

Operational Feasibility

In our proposed system is designed from a user's point of view.

For that reason all of the features are included only for the benefit of the user.

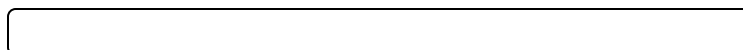
Economic Feasibility

The entire development period the organization required

- Manpower,
- Computer
- Internet connection etc.

Chapter 2

Requirement Engineering



2.1 Requirement Engineering:

- ❖ User Requirement
- ❖ System Requirement
- ❖ Functional Requirement
- ❖ Non-functional Requirement

2.1.1 User Requirements for user:

1. User will carry an id via which he/she can login (with password).
2. User will sort the problems in terms of difficulty level.
3. User will sort the problem in terms of problem category.
4. Problem setter can change the statement during contest time.
5. Admin can remove a problem from site.
6. All beginner programmer can submit the problem in offline and during contest time also
7. Anyone can see all submitted code for a particular problem.
8. User will get a badge (newbie, pupil, specialist) according to his/her number of solved problems.

2.1.2 System Requirement for user

- 1.1 System must have the login panel
- 1.2 System must have user name and password of the user
- 2.1 System should have the tag and rating number of each problem statement for sorting
- 3.1 System must have problem statement category (easy, medium, hard)
- 4.1 System must contain the information of problem setter
- 4.2 System must have a panel to add, update and delete problem statement
- 5.1 System must have information of admin
- 6.1 System will give accessibility all the time for submission online and offline
- 7.1 System will give accessibility to see all submitted code by any user.
- 8.1 System must have record of the number of submitted code by each user.

2.1.3 Functional Requirement:

1. User will go to home page with his or her password to login and verification.
2. The server should automatically back up the database every 24 hours.
3. The system should allow only admin to access a user's personal data.

4. User can click the side navigation page to browse different website sections.
5. User's submission must be verified with the pre-defined output.
6. Each user should be rated based on their acceptance of the code.
7. The system should allow user to write any kind of queries in the comment section.
8. The software passes all security requirements.

2.1.4 Non-Functional requirement:

1. The website's interface has to be user-friendly and easy to use.
2. Only the users with the role "admin" can view the user's verified phone number.
3. When editing the code for user's profile pages, the rest of the site stays up and running.
4. The system shall maintain data integrity by keeping backups of all updates to the
5. database for every record transaction.
6. The website's load time should not be more than one second for users.

2.2 Use case diagram:

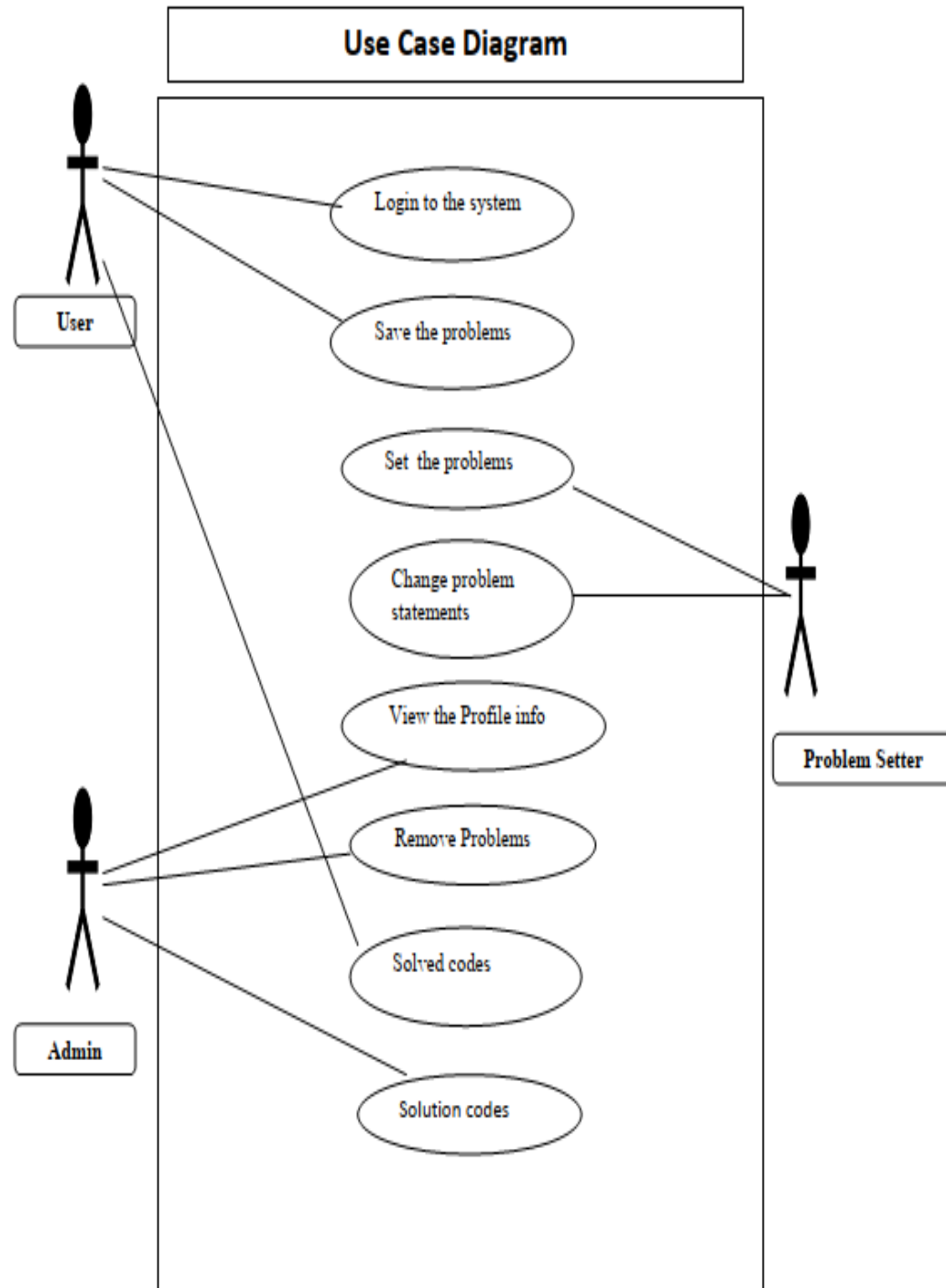


Fig- Use case Diagram

System Planning

3.1 Functions of Proposed System

1. New Member Create account	[F01]
2. Login into Application	[F02]
3. Edit profile inside Application	[F03]
4. Read problem statement	[F04]
5. submit problem solution	[F05]
6. Application verify code submission	[F06]
7. Application give ranking	[F07]
8. Verify Admin	[F08]
9. Admin create event and notice	[F09]
10. Admin set problem statement	[F10]
11. Instructor Analysis the performance of user	[F11]

3.1.1 Function Description

The function description is a detailed description of the function. It is concerned with three factors: what are the available inputs and outputs for a certain function, as well as which database table the function uses.

- **New Member Create account**
Input: User information
Output: If all the information is valid then it will show a successful message else error message. **Use table of the database:** user
- **Login into Application**
Input: User information
Output: If all the information is valid then it will show a successful message and login else error message.
Use table of the database: user
- **Edit profile inside Application**
Input: User information check and update
Output: If all the information is valid then it will show a successful message and update else error message.
Use table of the database: user
- **Read problem statement**
Input: user request for the problem list application query

- Output:** If all the information is valid then it will show a successful the problem list to user and problems are not available.
Use table of the database: problemlist, user
- **submit problem solution**
Input: User submit a specific problem
Output: If all the code is valid then it will show a successful accepted else error wrong .
Use table of the database: user, problem, submission
 - **Application verify code submission**
Input: submit a specific problem get verdict
Output: If all the code is valid then it will show a successful accepted else error wrong .
if Accepted the get score on that
Use table of the database: user, problem, submission
 - **Application give ranking**
Input: user Get score based on the valid submission for the problem
Output: If all the code is valid then it will show a successful accepted else error wrong .
if Accepted the get score on that and Rank list will updated.
Use table of the database: problemlist, user, submission, score.
 - **Verify Admin**
Input: Admin information
Output: If all the information is valid then it will show a successful message and login
else error message.
Use table of the database: Admin
 - **Admin create event and notice**
Input: Admin give proper information for notice and event
Output: If all the information is valid then it will show a successful message else error
message.
Use table of the database: notice and event
 - **Admin set problem statement**
Input: Admin give problem information
Output: If all the information is valid then it will show a successful message and added
new problem into Application else error message.
Use table of the database: problem, Admin
 - **Instructor Analysis the performance of user**
Input: User problem submission information
Output: If all the information is valid then it will show a successful message and rank
and performance updated else error message.
Use table of the database: user, submission, score

3.2 System Project Planning

CPF's second activity is software project planning. The first step in software project management is to complete a set of activities known as software project planning. Before beginning any project, it is necessary to estimate the work to be done, the resources that will be required, the

time it will take to complete the project, and to analyze the project to see if it is feasible. The following activities of software project planning that have followed in this project are:

- System Project Estimation
- Function Oriented Metrics
- Effort Distribution
- Project Schedule Chart
- Cost Estimation

3.2.1 System Project Estimation

The accuracy of a software project estimate predicated based on a number of things:

- Properly estimated the size of the product to build.
- The ability to translate the size estimation into human effort, calendar time and money.
- The degree to which the project plan reflects the abilities of the software team or engineer.
- The stability of the product requirements and the environment that supports the software engineering effort

Software size estimation is the most important matter that I have to consider during the software project. If the software size not calculate properly, then this will cause various problems such as scheduling problems, budget problem etc. As the project goes on. Before estimating the software size, I have to confirm that software scope is bounded.

3.2.2 Function Oriented Metrics

Instead of focusing on software values, function point based estimate focuses on information domain values. Five information domain properties are compared to calculate function points. The following are the information domain values:

Number of external inputs – Inputs should be differentiated from inquiries since each user input offers distinct application-oriented data to the software.

Number of external outputs – Each user output that provides the user with application-specific information is counted.

Number of external inquiries – An inquiry is described as an on-line input that results in the development of an on-line output in the form of a software response. Each enquiry was counted separately.

Number of files – Each logical master file counted

Numbers of external interfaces – The number of machine-readable interfaces utilized to transport data to another system was counted.

The following formula is used to calculate the FP count:

$$FP\ Count = (((4 * Most\ Likely) + Optimistic + Pessimistic) / 6) * Weight$$

To compute function points (FP), the following relationship is used:

$$FP = Count\ Total * [0.65 + 0.01 * \sum (Fi)]$$

Complexity adjustment factor = $[0.65 + 0.01 * \text{Sum of factor values}]$

FP estimated = count total * Complexity adjustment factor.

Function Point Estimation = Total FP estimated / No. of function point.

The count total is the sum of all FP entries.

Function Point Estimation for Transaction Function

Table 3.1 Identifying Complexity for Transition Function

Transaction Function	FTRS	DETs	Complexity	Q
Search problem (EQ)	1	7	Low	3
Problem statement (EQ)	2	7	Average	4
Check validity (EI)	1	7	Low	3
Profile modification (EI)	3	13	High	6
Update Problem (EI)	1	10	Low	6
Performance Report (EO)	1	5	Low	3
Event Add (EI)	1	4	Low	3
Member modification (EI)	1	13	Low	4
Total				32

EI	1-4 DETs	5-15 DETs	16 or more DETs	EQ/EO	1-5 DETs	6-19 DETs	20 or more RETs	Complexity	Truncation Function Type	
1 FTR	Low	Low	Average	1 FTR	Low	Low	Average		EI/EQ	EO
3 or more FTRs	Average	High	High	2 to 3	Low	Average	High	Low	3	4
2 FTRs	Low	Average	High	4 or more FTRs	Average	High	High	Average	4	5
								High	6	7

Table 3.2 Identifying Complexity for Data Function

Data Function	RETs	DETs	Complexity	UFP
Get solution (ILF)	1	9	Low	7
Compile code (EIF)	1	2	Low	5
Submit Code (ILF)	1	3	Low	7
Point Calculation (ILF)	3	13	Low	7
Tot				26

ILF/EIF	1-19 DETs	20-50 DETs	51 or more RETs
1 RET	Low	Low	Average
2 to 5 RETs	Low	Average	High
6 or more RETs	Average	High	High

Complexity	Data Function	Type
	ILF	EIF
Low	7	5
Average	10	7
High	15	10

Table: 3.3 Performance and environmental impact

	GSC	TDI
1	Data Communication	2
2	Distributed Data Processing	0
3	Performance	3
4	Heavily Used Configuration	1
5	Transaction Rate	0
6	Online Data Entry	3
7	End user Efficiency	4
8	Online Update	2
9	Complex Processing	2
10	Reusability	3
11	Installation Ease	3
12	Operational Ease	3
13	Multiple States	3
14	Facilitate Change	3
Total Degree of Influence (TDI)		32

3.2.3 Function Point Estimation

Value Adjustment Factor (VAF) = $(0.65 + (0.01 * \text{TDI}))$
= $(0.65 + (0.01 * 32)) = 0.97$

UFP = UFP (Data function) + UFP (Transaction function)
= 56 + 52 = 108

AFP = UFP * VAF = 108 * 0.97 = 104 Approx

Total time calculation frame = 104 * 14.5 = 1508 per hour
= 1508 person hours / 8 hours
= 188 person days / 5 [person in a group]
= 37 days per person
= 1 months 7 days for one person

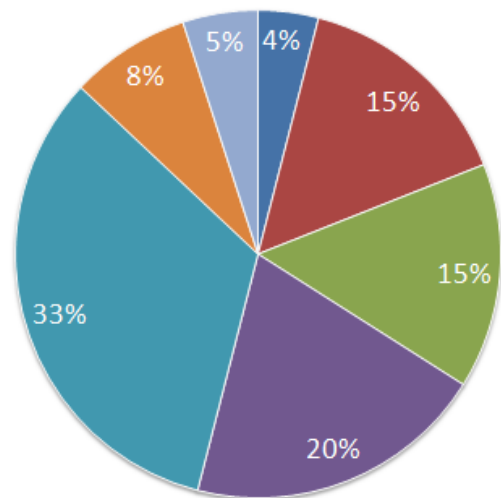
Approximately 1 months 15 days are required for 5 persons to finish the project

3.2.4 Effort Distribution

The project estimation method yields estimates of the number of work units needed to complete software development. The 40-20-40 rule is a recommended distribution of effort between the definition and development phases. Front-end analysis and design received 40% of total effort, coding received 15%, and back-end testing received 33% of total effort. This rule should only be used as a guideline. In this project, 35% of full software development has been allocated to analysis and design, 36% has allocated to coding and the remaining 20% is allocated to software testing and support.

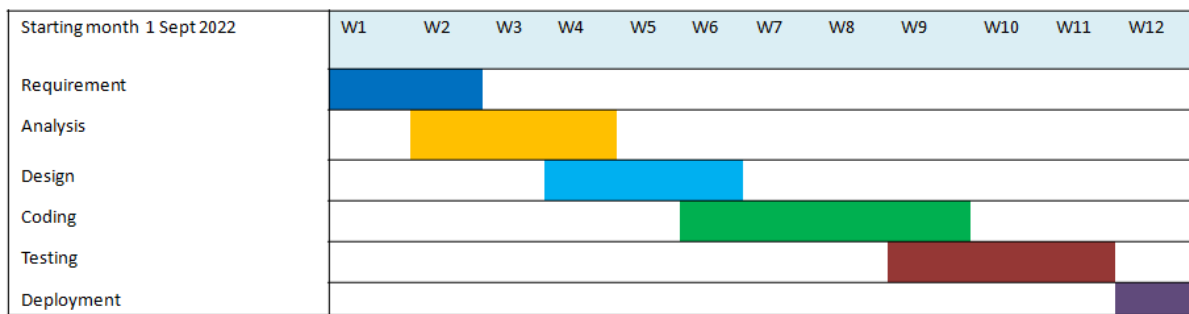
Effort Based Estimation

- 4% - Employee Communication
- 15% - Planning
- 15% - Analyzing
- 20% - Designing
- 33% - Coding
- 8% - Testing
- 5% - Implementation



3.2.5 Project Schedule Chart

Total system development is a combination of set of tasks. These set of tasks should done sequentially and timely. Project schedule works as the guideline of the system developer. The following is the schedule chart of this project:

Table: 3.4 Schedule Chart**3.2.6 Cost Estimation**

- Personnel Cost
- Software Cost
- Hardware Cost

3.2.6.1 Personnel cost

Number of days in a year = 365

Number of government holidays in a year =24

Number of weekly holidays in a year =52

Total number of working days to develop the project =365-(52+24) =289 days

Total number of working days per months to develop the project =289/5 =58 days

Organization working hours per day = 8hours

Organization working hours per month=58*8= 462 hours

Table: 3.5 Personal cost

Type	Number of members	Months	Salary
Project Manager	1	1	30000
Coder & Tester	1	1	60000
UI Designer	1	1	20000
Total	3	3	110000

3.2.6.2 Hardware cost:**Table: 3.6Hardware Cost**

Hardware Name	Number	Amount	Price
Desktop	1	1	40000
Total	1	1	40000

3.2.6.3 Software Cost

Table: 3.7Software Cost

Software Name	Number	Amount	Total
Windows operating system	1	10000	10000
Microsoft Office 2019	1	3000	3000
Visual Studio Code	1	0	0
Total	3	3	13000

3.2.6.4 Other Cost

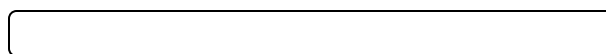
Table: 3.8 Others Cost

Type	Amount
Electricity Bill	1500
Transport	2500
Total	4000

$$\begin{aligned}
 \text{Total cost in BDT} &= \text{Development cost} + \text{Hardware cost} + \text{Software cost} + \text{Others} \\
 &= 110000 + 40000 + 13000 + 4000 \\
 &= 167000
 \end{aligned}$$

Chapter 4

Risk Management



4.1 Risk Management

Risk management is the process of identifying, assessing and controlling threats to an organization's capital and earnings. These risks stem from a variety of sources including financial uncertainties, legal liabilities, technology issues, strategic management errors, accidents and natural disasters.

- **Technology risk**
- **Organizational risk**
- **People risk**

4.2 RMMM Plan

Technology risk

Name	Inappropriate database
Probability	Moderate
Impact	Serious
Description	The organization may face numerous number of hits from the customer or the system may need to do lots of crud operation that the database can not handle with
Mitigation & monitoring	The system may try to minimize the response time of queries by making the best use of system resources.
Management	The organization will investigate the possibility of buying a higher performance database
Status	Managed

Organizational risk

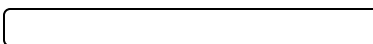
Name	Organizational financial problems
Probability	Moderate
Impact	Catastrophic
Description	The organization may not have enough capital or no preparation for unforeseen expenses
Mitigation & monitoring	i. Re-evaluate the plan ii. Adjust your budget to fit the expenses
Management	Prepare a briefing document showing the project is making very important contribution to the goals of the business.
Status	Mitigation steps initiated.

People risk

Name	Unavailability of key staffs at certain times
Probability	Moderate
Impact	Catastrophic
Description	There can be a situation arise when the key members can be unavailable to some unavailable reasons
Mitigation & monitoring	<ul style="list-style-type: none">i. Understand the problem.ii. Underlying issues can be solved.iii. Acknowledging their struggles can make them feel more comfortable.
Management	Reorganize the team so that there is more overlap of work and people therefore understand each others job
Status	Nor occurred

Chapter 5

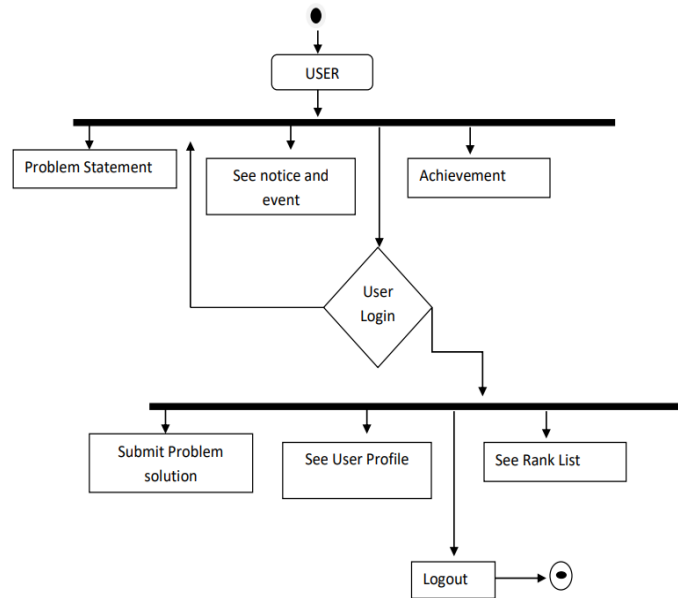
Analysis and Design



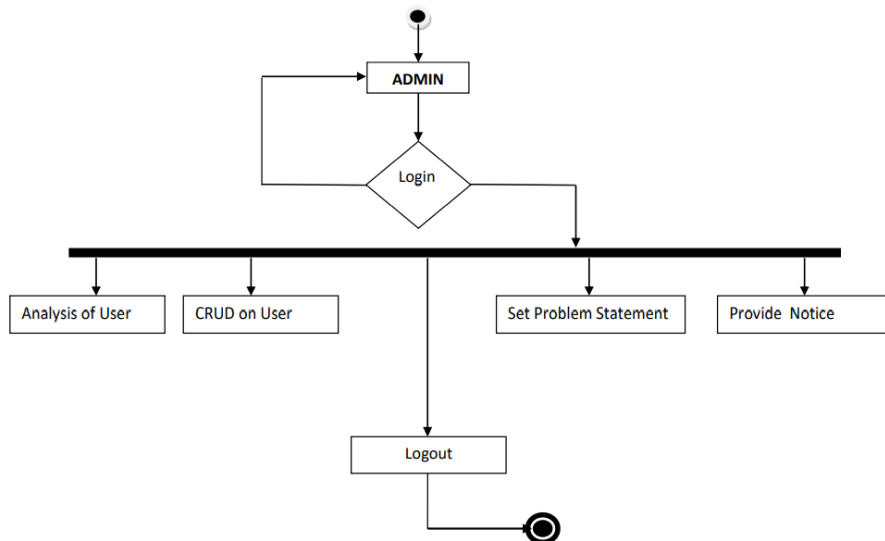
5.1 Activity Diagram

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.

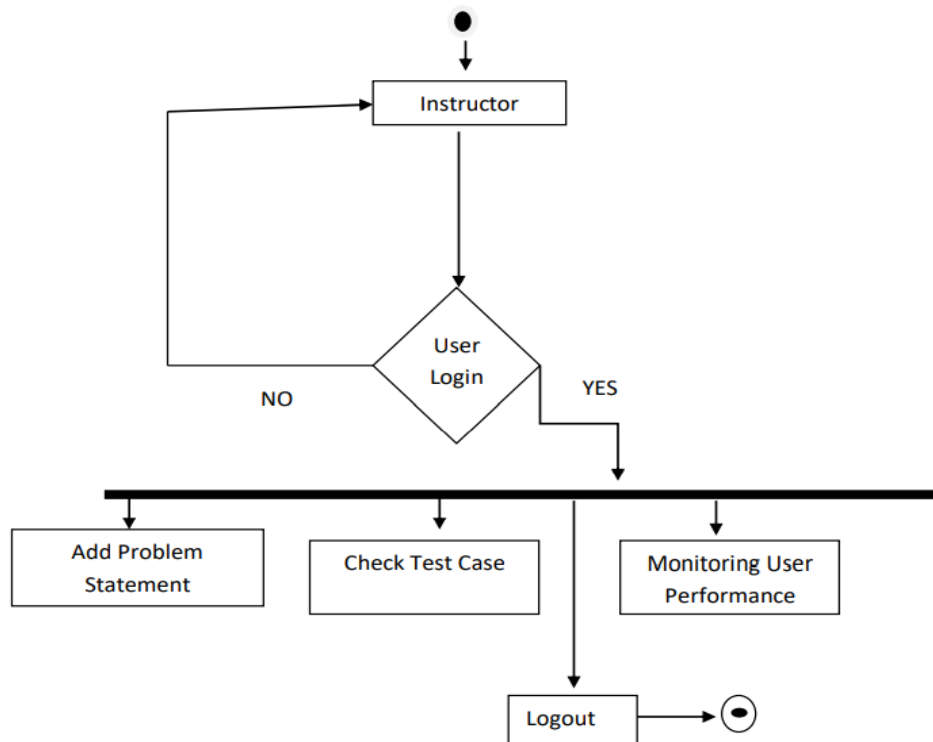
5.1.1 Activity diagram for user



5.1.2 Activity diagram for admin

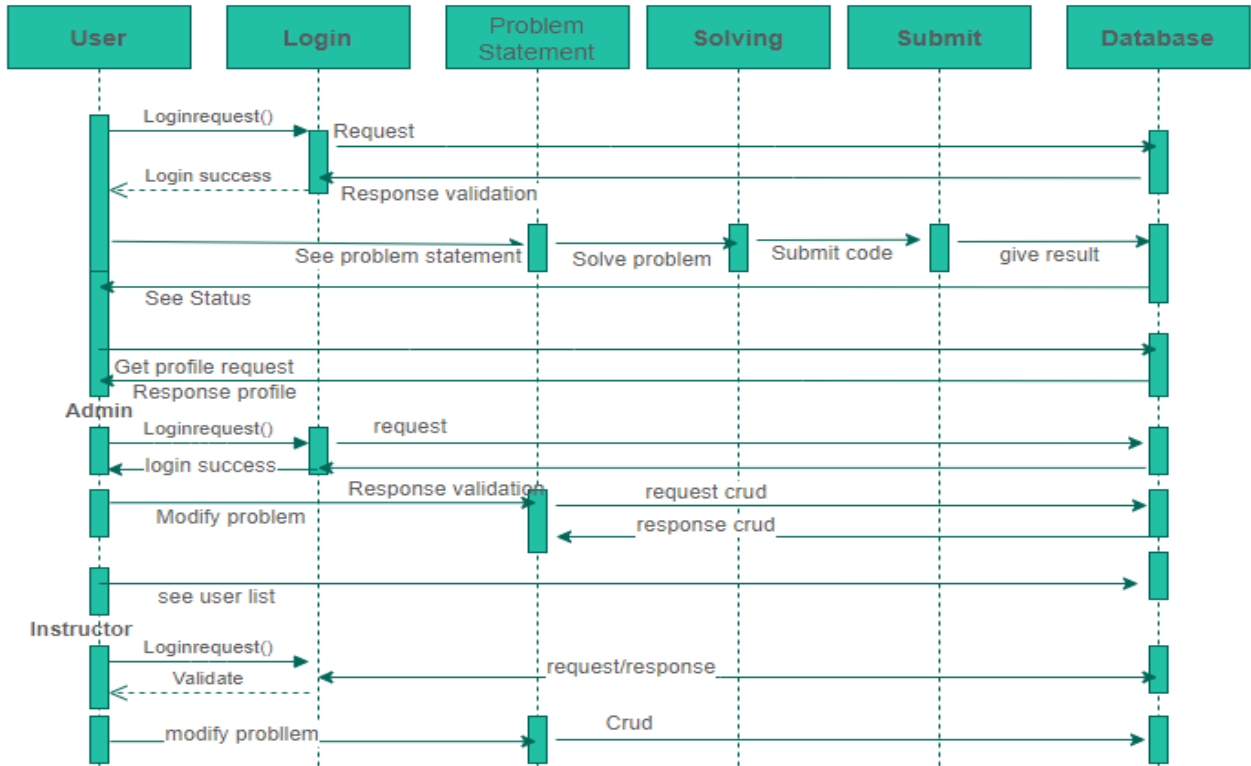


5.1.3 Activity diagram for instructor



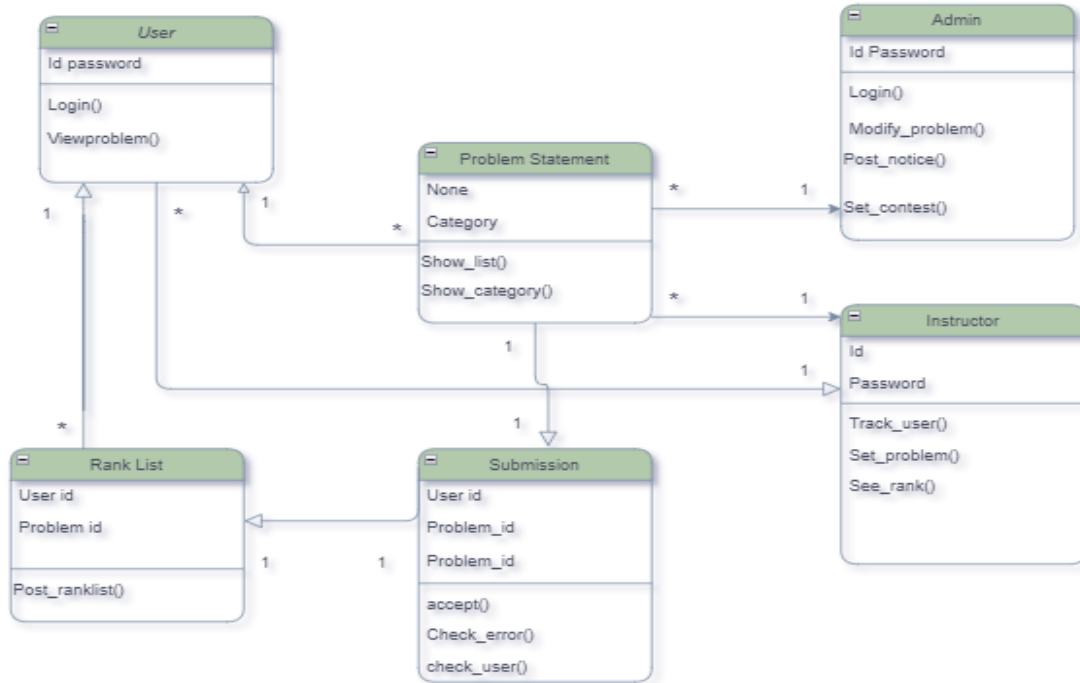
5.2 Sequence diagram

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.



5.3 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.



5.4 CRC

CRC (Class Responsibility Collaborator) cards are a simple way to break down the functionality of an application to reason about what the basic entities are (class), what they do (responsibility), and which other entities they need to communicate with (collaborator).

5.4.1 CRC for user

User	
Create account	
Login	
See problem	Problem statement
Submit the problem	Submission
View rank list	Rank
Check notice	Admin

5.4.2 CRC for admin

Admin	
Login	
View user profile	User
Modify problem	Problem statement
Post notice	
Set contest	

5.4.3 CRC for instructor

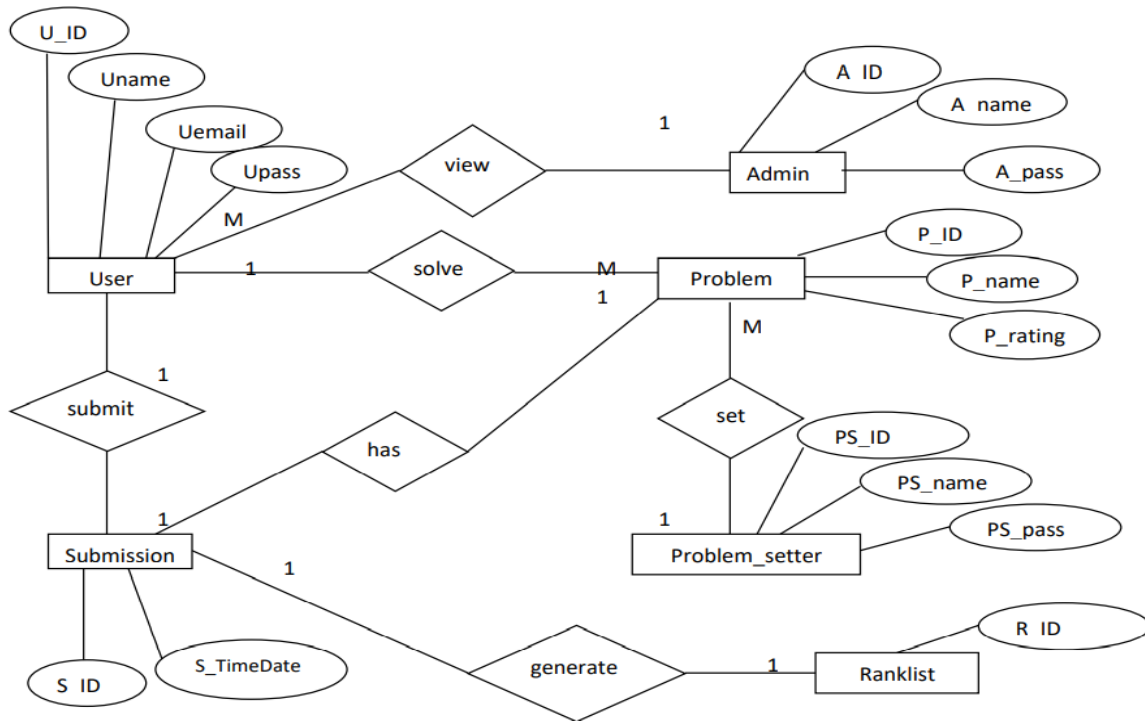
Instructor	
Login	
Keep track of user	User
Set problem	
Performance analysis	Rank
Post guidelines	

5.4.3 CRC for problem statement

Problem-Statement	
Show problem list	
Show category list	

5.5 ER diagram

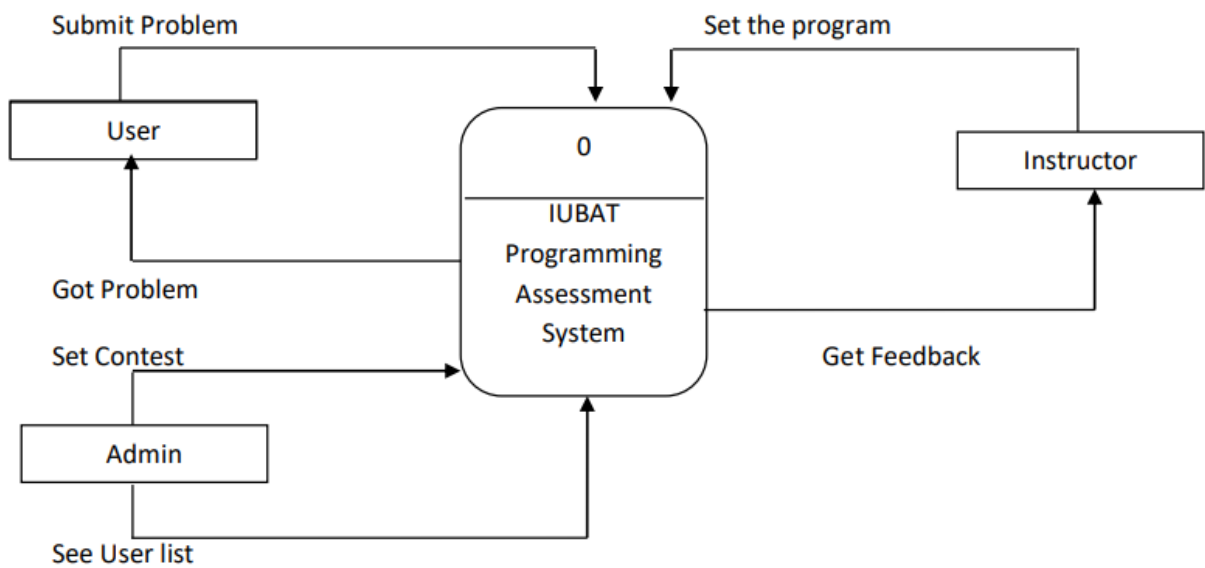
ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.



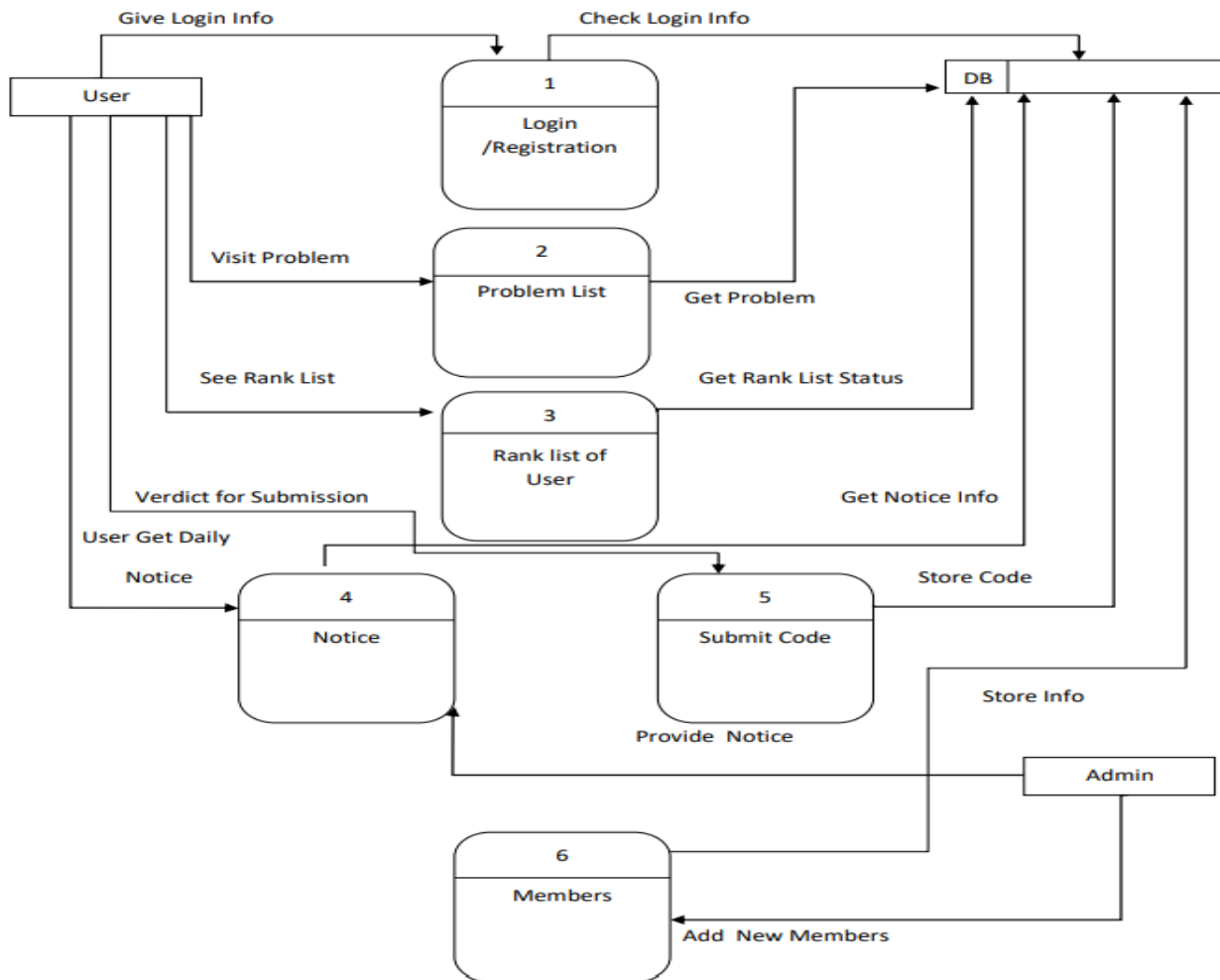
5.6 Data flow diagram

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.

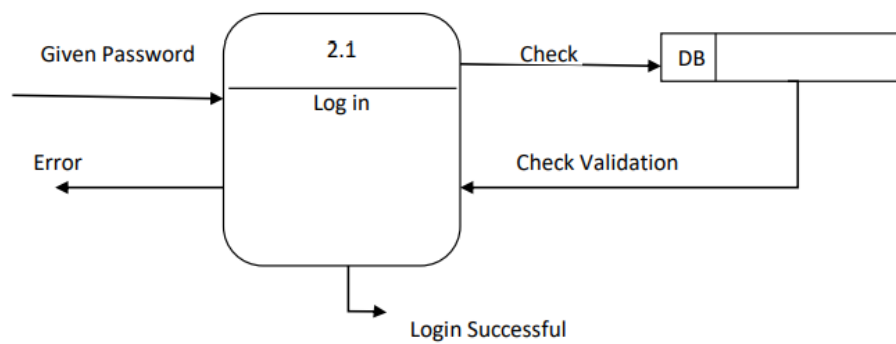
5.6.1 Context level DFD



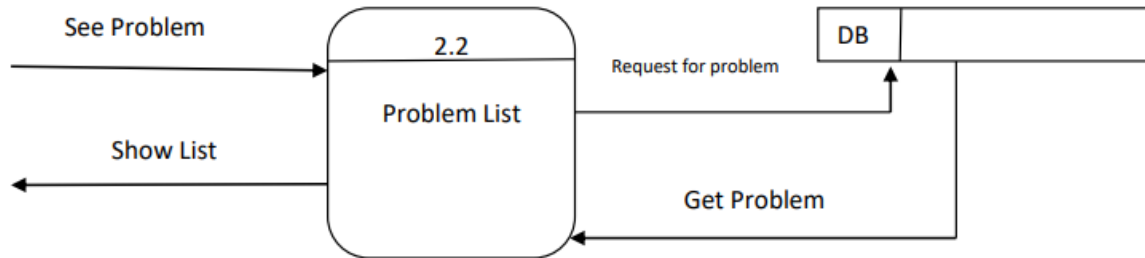
5.6.2 DFD level 1



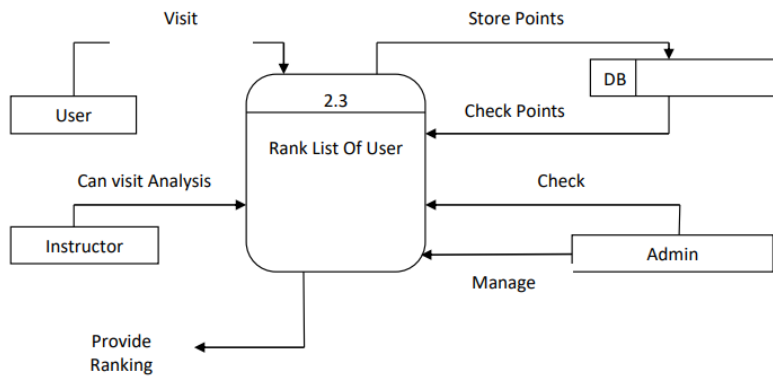
5.6.3 DFD level 2 process 1



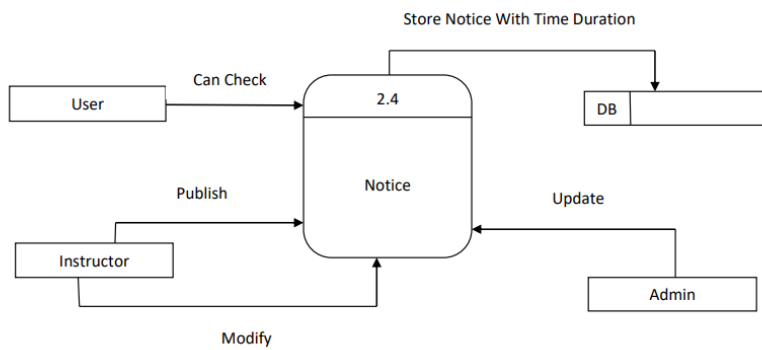
5.6.4 DFD level 2 process 2



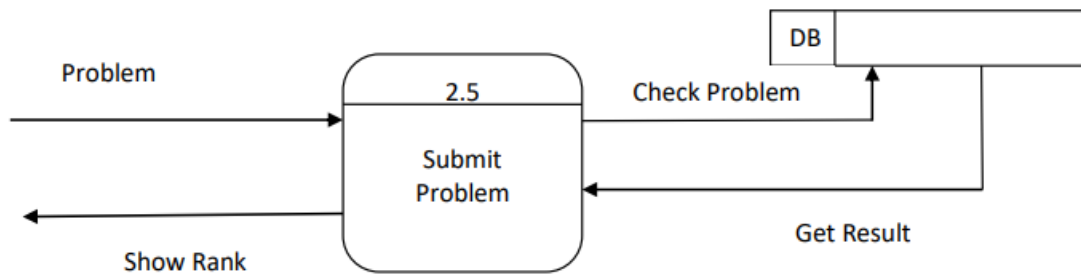
5.6.5 DFD level 2 process 3



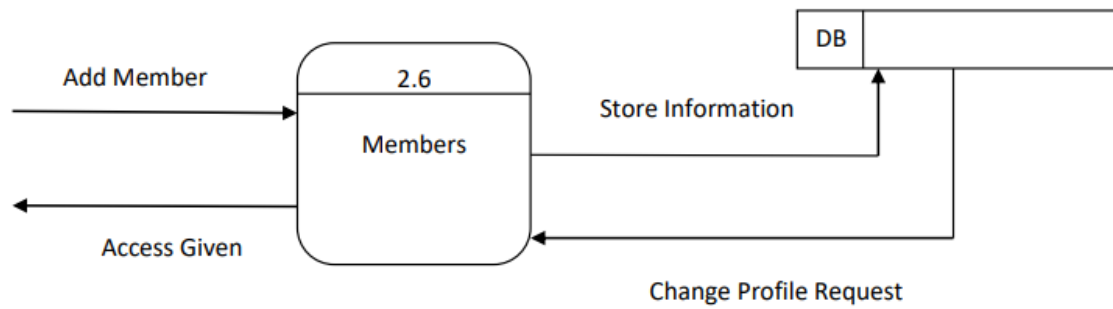
5.6.6 DFD level 2 process 4



5.6.7 DFD level 2 process 5



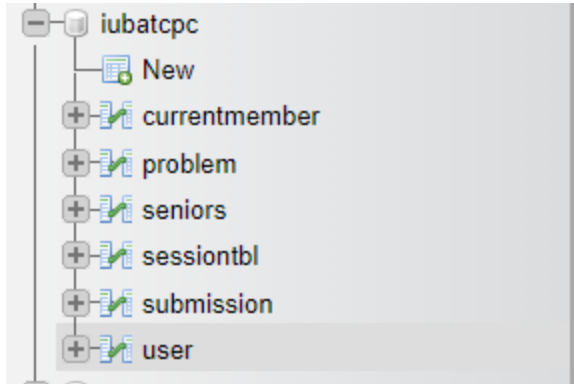
5.6.8 DFD level 2 process 6



Chapter 6

Database, UI design, Testing

6.1 Database table structure



		uid	username	password	cntac	cntpoint	cntwa
<input type="checkbox"/>	Edit Copy Delete	5	karim	\$2a\$10\$O2AuYvTaMvYBq1ZDzV3iieSws3f8VKaAmAQhjpcWJvB...	10	100	0
<input type="checkbox"/>	Edit Copy Delete	6	hasan	\$2a\$10\$EC5mm5J9bztqQ1u5bgtTSu/Kv4LJz1Ov/pSA6Kj5XL5...	0	0	0
<input type="checkbox"/>	Edit Copy Delete	7	suchana	\$2a\$10\$rQJ9R41Siu9Exr3Coj9ER.moblbJqBbS5P3tWXbE55v...	1	10	1
<input type="checkbox"/>	Edit Copy Delete	9	goodboy	\$2a\$10\$KwFLL9FE5oAf4yzQ9tEUEugL5SWtfr.AGwAh/UZESSh...	0	0	0
<input type="checkbox"/>	Edit Copy Delete	10	rahim1	\$2a\$10\$rKmZMkdlibbZiRXKT9jvFv.zMhKHx9QdJXonuJlffY8D...	0	0	0

		pid	name	statement	inputs	outputs	input	output	difficulties	topic	solution
<input type="checkbox"/>	Edit Copy Delete	1	Two Sum	Given an array of integers nums and an integer tar...	<p>Take n in first line next line n numbers of inp...	Gives the Indexes of pair that make the target sum	<p>4</p><p>2 7 11 15</p><p>9</p>	0 1	Easy	two pointer	#include<bits/stdc++.h> using namespace std; int ...
<input type="checkbox"/>	Edit Copy Delete	2	Print number	<p>Write a program that print numbers 1 to ...	<p>Take input from user</p>	<p>Each line print Number</p>	<p>5</p>	<p>1</p><p>2</p><p>3</p><p>4</p><p>5</p>	Easy	loop	#include<bits/stdc++.h> using namespace std; int m...
<input type="checkbox"/>	Edit Copy Delete	3	Palindrome Number	<p>Given an integer x, return true<i> if </i>x<i> ...	<p>Take a number from user</p>	<p>Print YES if it is a palindrome, otherwise NO</...>	<p>121</p><p>YES</p>		Easy	palindrome	#include<bits/stdc++.h> using namespace std; int m...

suid	spid	status	ssolution	visitsolution	acStatus
1	2	Accepted	#include<bits/stdc++.h> using namespace std; int...	NO	1
1	3	Accepted	#include<bits/stdc++.h> using namespace std; int...	NO	1
5	3	Accepted	#include<bits/stdc++.h> using namespace std; int m...	NO	1
7	2	Wrong	#include<bits/stdc++.h> using namespace std; int m...	NO	1

6.2 UI design screenshots



Registration

First Name	Last Name	user Name
<input type="text"/>	<input type="text"/>	<input type="text"/>
Email		
<input type="text" value="xyz@yahoo.com"/>		
Password		Confirm Password
<input type="text"/>		<input type="text"/>
Phone	Mobile	
<input type="text"/>	<input type="text"/>	
<input type="button" value="Submit"/> Login		

Login

Username

Password

[Register](#)

IUBAT_CPC

Problem

instructor

current member

Senior member

Notice

Event

SignOut

×

Admin

Problem

Add new problem as well as modify

Problem page

Current Members

Team members achieve their goals

Current Member

Notice

Regularly post some notice

Notice Page

Instructor

Review and analyze user performance over a period of time

Instructor Page

Senior Members

With supporting and inspiring to others

Senior Members

Event

Upcomming and running event give here

Event Page

Admin

Add Problem

Problem Name

Problem Statement

Paragraph

B *I*

write

Input Statement

Paragraph

B *I*

write

Output Statement

Paragraph

B *I*

write

Example Input

Paragraph

B *I*

1 Two Sum

Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target. You may assume that each input would have exactly one solution, and you may not use the same element twice. You can return the answer in any order.

Input

Take n in first line next line n numbers of input Then a Target

$2 \leq \text{nums.length} \leq 10^4$

$-10^9 \leq \text{target} \leq 10^9$

Output

Gives the Indexes of pair that make the target sum

Test Case

Input:

4

Submit Code

Code

```

1 #include<bits/stdc++.h>
2 using namespace std;
3 int main()
4 {
5     int n;
6     cin >> n;
7     vector<int> nums(n);
8     for(int i = 0; i < n; i++)
9     {
10         cin >> nums[i];
11     }
12     int target;
13     cin >> target;
14     int a=0;
15     int b=0;
16     for(int i=0; i<nums.size()-1; i++)
17     {
18         for(int j=i+1; j<nums.size(); j++)
19         {
20             if(nums[i]+nums[j]==target)
21             {
22                 a=i;
23                 b=j;
24             }
25         }
26     }
27 }
```

6.3 Testing

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

In our system, we have used black box testing. Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing.

Testing scenario No:1	
Scenario	User Login testing scenario of my system
Input's	E-mail, password of User for Login
Desired Output's	When enter E-mail, password then get access after verification.
Actual Output's	For login my system works correctly
Verdict	The process worked correctly and successfully.

Testing scenario No: 2	
Scenario	Admin can add, view, modify and delete Problem statement
Input's	Submit all the problem statement
Desired Output's	Save the problem statement into the system
Actual Output's	User can be able to see all the problem statement without having any issues.
Verdict	The process worked correctly and successfully.

Conclusion

Competitive programming is a sport, perhaps even a form of art. It's an activity that requires creativity and analytical thinking to tackle difficult coding problems. The skills needed for competitive programming have long-lasting benefits to career as a developer. There are numerous benefits to participating in competitive programming. Our system will provide an environment where beginner level programmer can be benefited. It will help to increase problem-solving speed which provides an edge over other applicants. The problems that is given here might seem simple but constantly practicing these problems at the beginning of the journey will improve speed and develop essential skill of writing error free code.