COCSC-11 SOFTWARE ENGINEERING

PRACTICLE FILE

NAME: AMOGH GARG
ROLL NUMBER: 2020UCO1688
BRANCH: COE
SECTION: 3

CONTENTS

PROBLEM STATEMENT

Many times it so happens that students at NSUT are able to arrange the previous year question papers for themselves, but they are unable to find solutions to those questions. Previous year question papers are available on various platforms like NSUTx and NSUT-Resources but these platforms do not provide proper means for the students to access the solutions of these papers. The problem here is to devise a mechanism which will make solutions to these question papers accessible to the students at NSUT. Apart from this, students should also be able to ask doubts separately which are not part of previous year question papers. This would also increase the interaction amongst students from different branches, year and even interact with professors.

The solution to this problem should ensure that the answers of the previous year question papers are accessible with ease and no special training and configuration is required from the client-side. Only those who belong to NSUT i.e. either students or professors should be able to utilize this facility.

The solution should ensure proper security of the data and should be reliable in terms of availability and efficiency. It should also be cost effective in order to maximize the number of users in the form of professors and students so that everyone in the university can benefit from this facility. Proper checks should be done to prevent the violation of the terms of use and rules which are made by the university in order to ensure a formal and productive utilization of this facility.

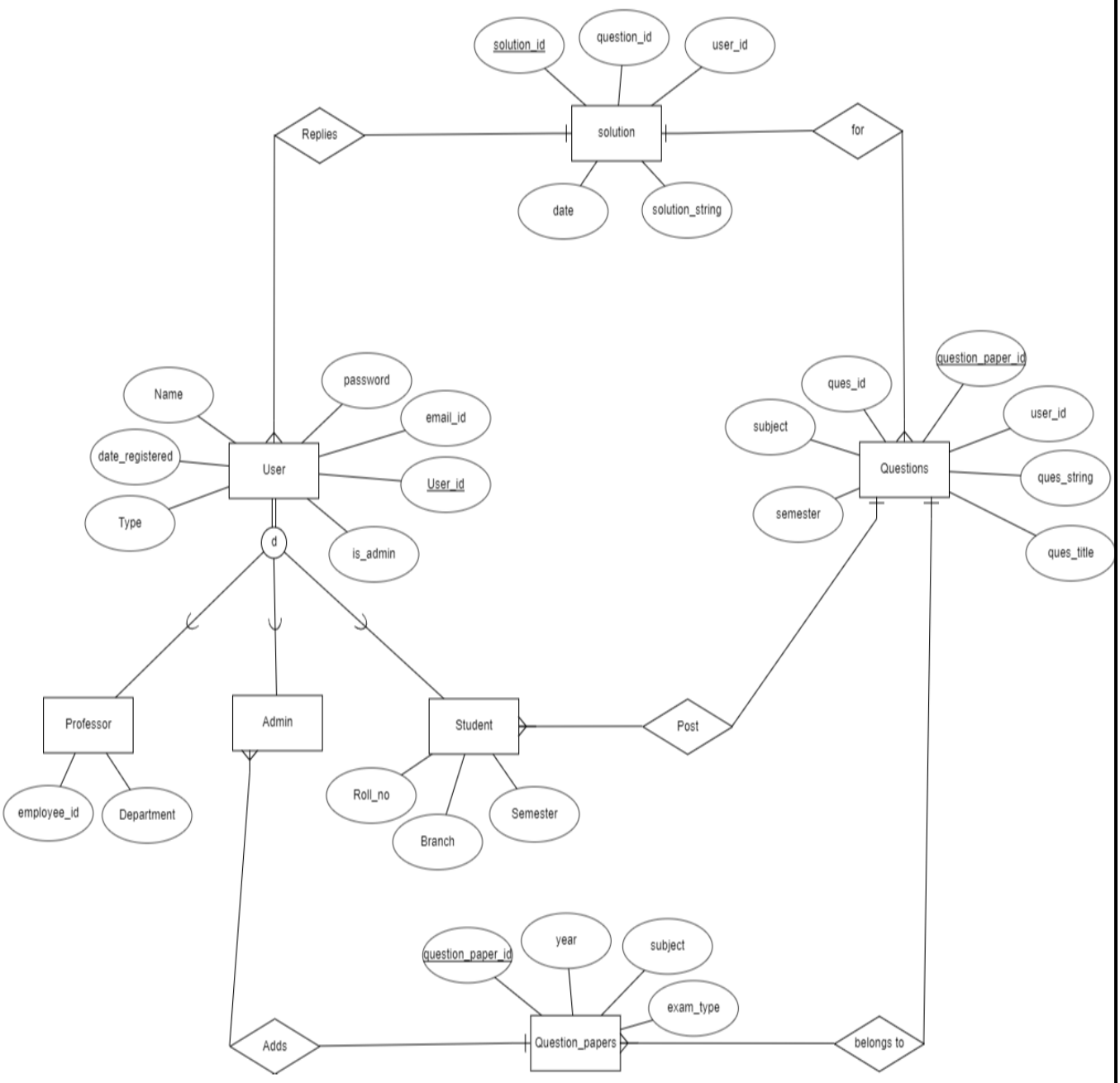
Therefore, the solution should be such that it should be able to meet all of the above requirements and yet be simple in its usage.

SOLUTION

Our proposed solution is to develop software which will allow students of the university to discuss the Previous question paper with each other and with the professor. There will be a separate discussion forum for each paper which will be uploaded. When selecting a particular question-paper, a discussion forum will open where students can post their doubts and questions. The discussion forum will be question specific so that users can find answers to their specific problems if the problem has already been discussed. In each discussion forum, faculty responses and solutions will be kept separate as expert solutions.

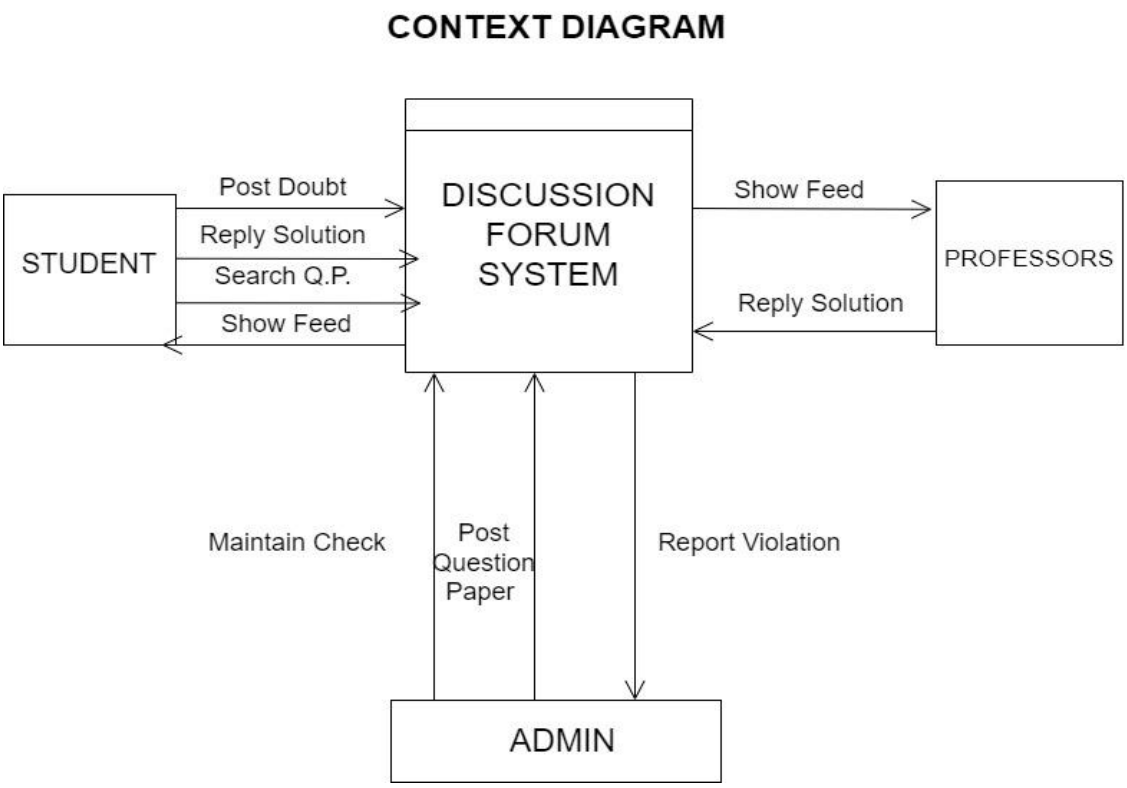
Additionally, there will be a feed section where students can post their question irrespective of any question paper, the questions that are recently posted, regardless of whether they are in question papers or posted independently, will be displayed in the feed section. Also, there is a section for searching questions and narrowing them down by semester, subject and topic.
Users can upload their solutions in jpg/jpeg/pdf/png formats. They will also be able to upvote a solution. NSUT students and faculties will be the only ones able to register with the technology.
The scope of the software can be expanded to include other universities. All the user data such as the University roll number and password will be stored on a cloud based database that is securely encrypted to preclude the data from being misused by unauthorized sources. Apart from this the UI will be fast, interactive and easy to use.

ER DIAGRAM

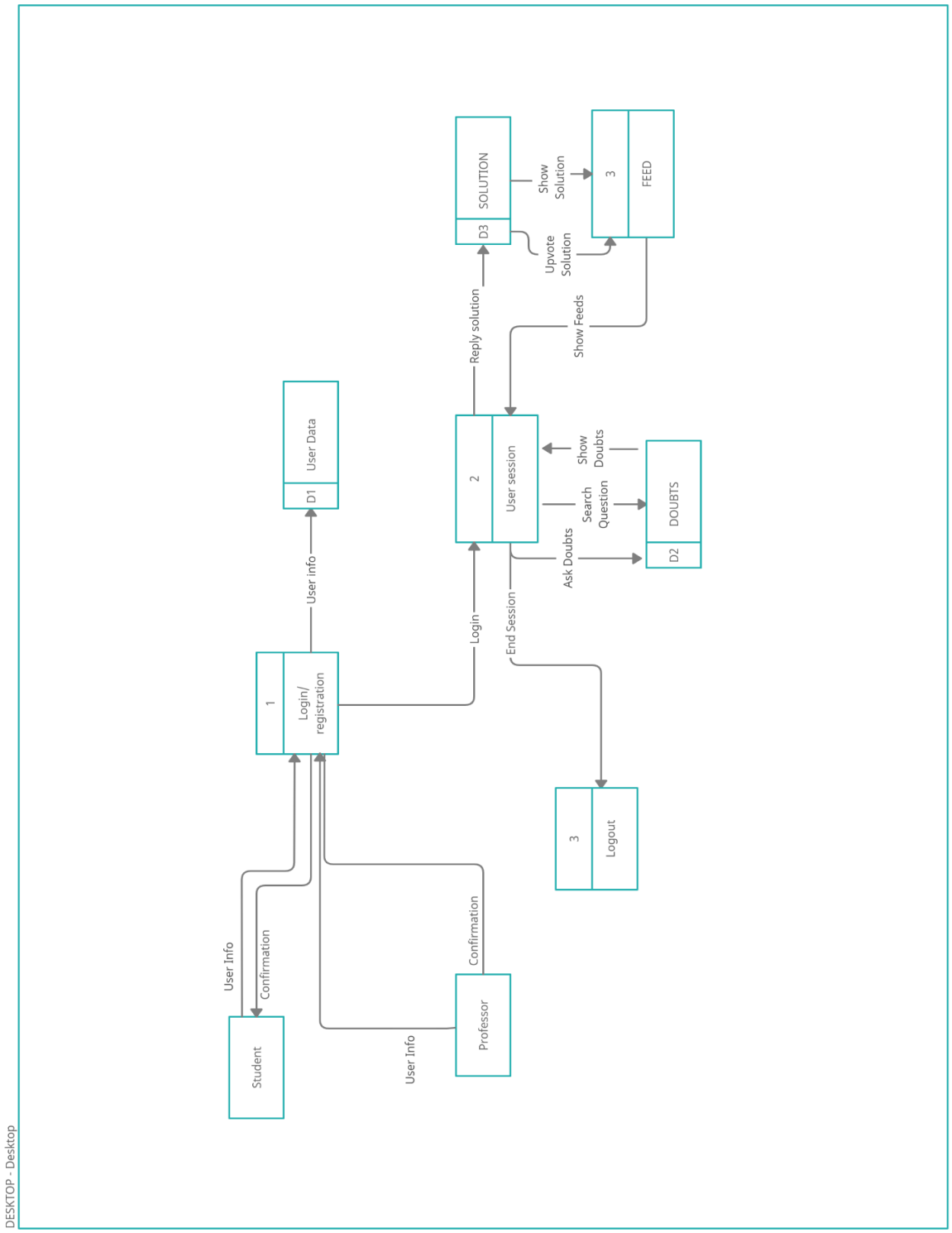

solution_id
question_id
user_id
Replies
solution
for
date
solution_string
Name
password
email_id
date_registered
User
User_id
Type
d
is_admin
question_paper_id
ques_id
user_id
subject
Questions
ques_string
semester
ques_title
Professor
Admin
Student
Post
employee_id
Department
Roll_no
Semester
Branch
question_paper_id
year
subject
exam_type
Adds
Question_papers
belongs to

DATA FLOW DIAGRAM

CONTEXT DAIGRAM



LEVEL-1 DIAGRAM



DATA DICTIONARY

ATTRIBUTE	ENTITY	DATA TYPE	DESCRIPTION	RANGE OF VALUES	VALIDATION
Name	User	Char	Name of the user	1-30 chars	
Password	User	Varchar	Login Password	At Least 8 chars	Must be at least 8 chars long with one special char
Email_ID	User	Varchar	Registered email-id	1-50 chars	Must contain @ symbol
User_ID	User	Integer	User-ID/Username	Exact 8 digits	Must be unique
Is_Admin	User	Boolean	Is user admin or not	True/False	
Type	User	Char	Is user a student or professor	"Student"/ "Professor"	
Date_Register	User	Date	Date of Registering	Date	=Current Date
Employee_ID	Professor	Integer	Employee_ID of professor	Exact 6 digits	Must be unique
Department	Professor	Char	Department of professor	1-50 chars	
Roll_No	Student	Varchar	Roll-number of the student	Exact 11 digits	Of the form "2020UC01688"
Branch	Student	Char	Branch of the student	1-30 chars	

Semester	Student	Integer	Current semester of student	1 digit	Must be between 1-8
QuestionPaper_ID	Question_Papers	Integer	ID of the question-paper	6 digits	Must be unique
Year	Question_Papers	Integer	Question-Paper year	4 digits	Exactly 4 digits
Subject	Question_Papers	Char	Question-Paper subject	1-30 chars	
Exam_type	Question_Papers	Char	Mid Sem/End Sem/CT	1-10 chars	"Mid Sem"/ "End Sem"/ "CT"
Ques_ID	Questions	Integer	ID of the question	8 digits	Must be unique
Ques_Title	Questions	Varchar	Title of the question	1-30 chars	
Question_string	Questions	Varchar	Text of the question	1-1000 chars	
Solution_ID	Solution	Integer	ID of the solution	8 digits	Must be unique
Solution_string	Solution	Varchar/Media	Text of the solution	1-5000 chars/Image/Pdf	
Date	Solution	Date	Date of posting the solution	Date	

PROCESS SPECIFICATION

PROCESS LOGIN /REGISTRATION

SPECIFICATION BY STRUCTURED ENGLISH:

```
IF REGISTERED
        INPUT EMAIL_ID
        INPUT PASSWORD
        IF PASSWORD IS CORRECT
                SET USERID = EMAIL_ID
        ELSE
                PRINT "WRONG USERNAME OR PASSWORD"
                RESET
ELSE
        INPUT NAME
        INPUT EMAIL_ID
        INPUT PASSWORD
        INPUT IS PROFESSOR OR IS STUDENT

        IF EMAIL_ID NOT REGISTERED
                        REGISTER USER
                        SET USERID = EMAIL_ID
```

SHOW FEEDS

SPECIFICATION BY PRE/POST CONDITIONS:

PRE-CONDITION1: USER HAS A VALID EMAIL_ID AND PASSWORD
POST-CONDITION1: GET ALL THE RECENT DOUBTS FROM DOUBTS
PRE-CONDITION2: PRE-CONDITION1 FAILS FOR ANY REASON
POST-CONDITION2: DISPLAY ERROR MESSAGE

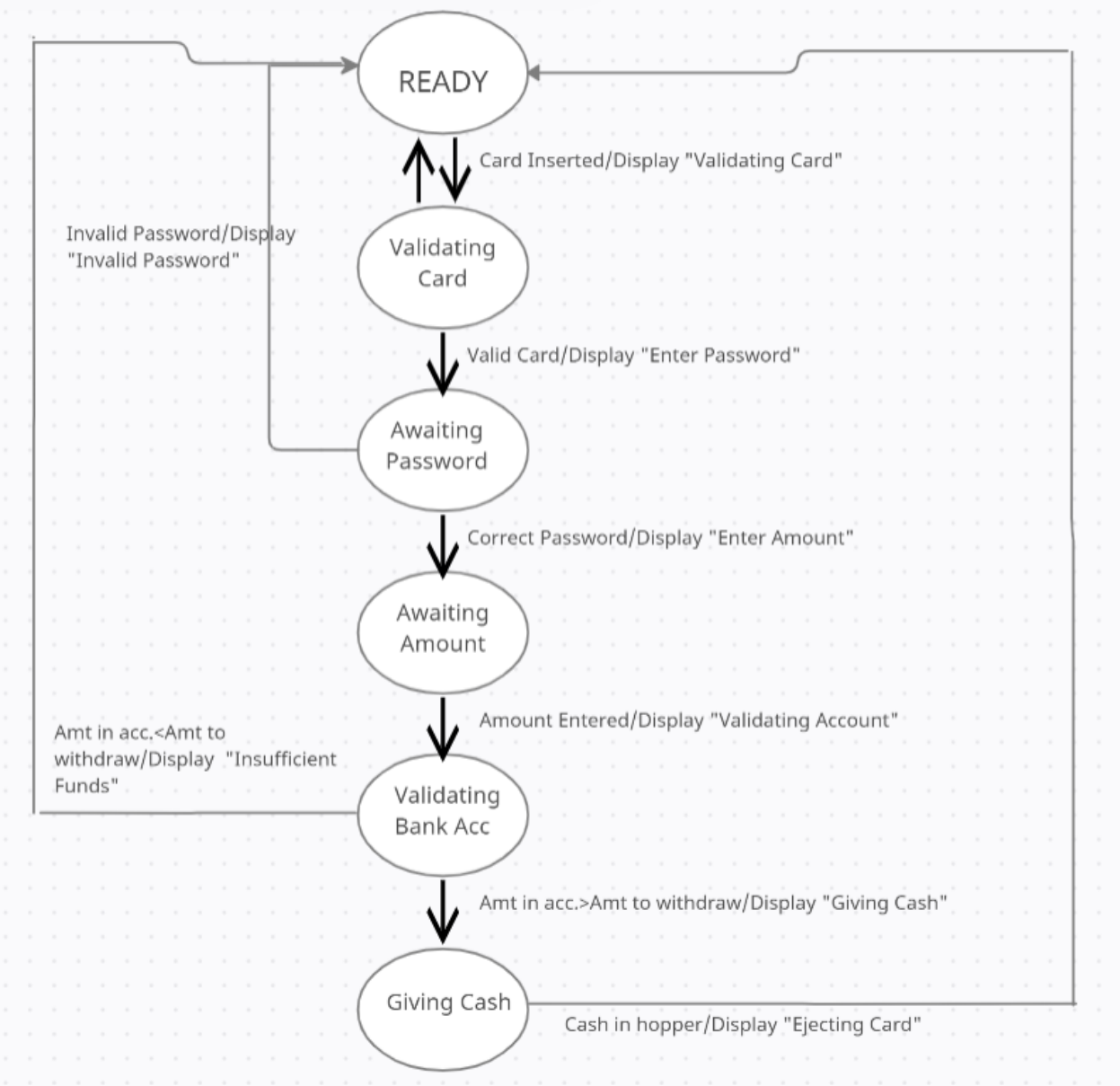
POST QUESTION

SPECIFICATION BY STRUCTURED ENGLISH:

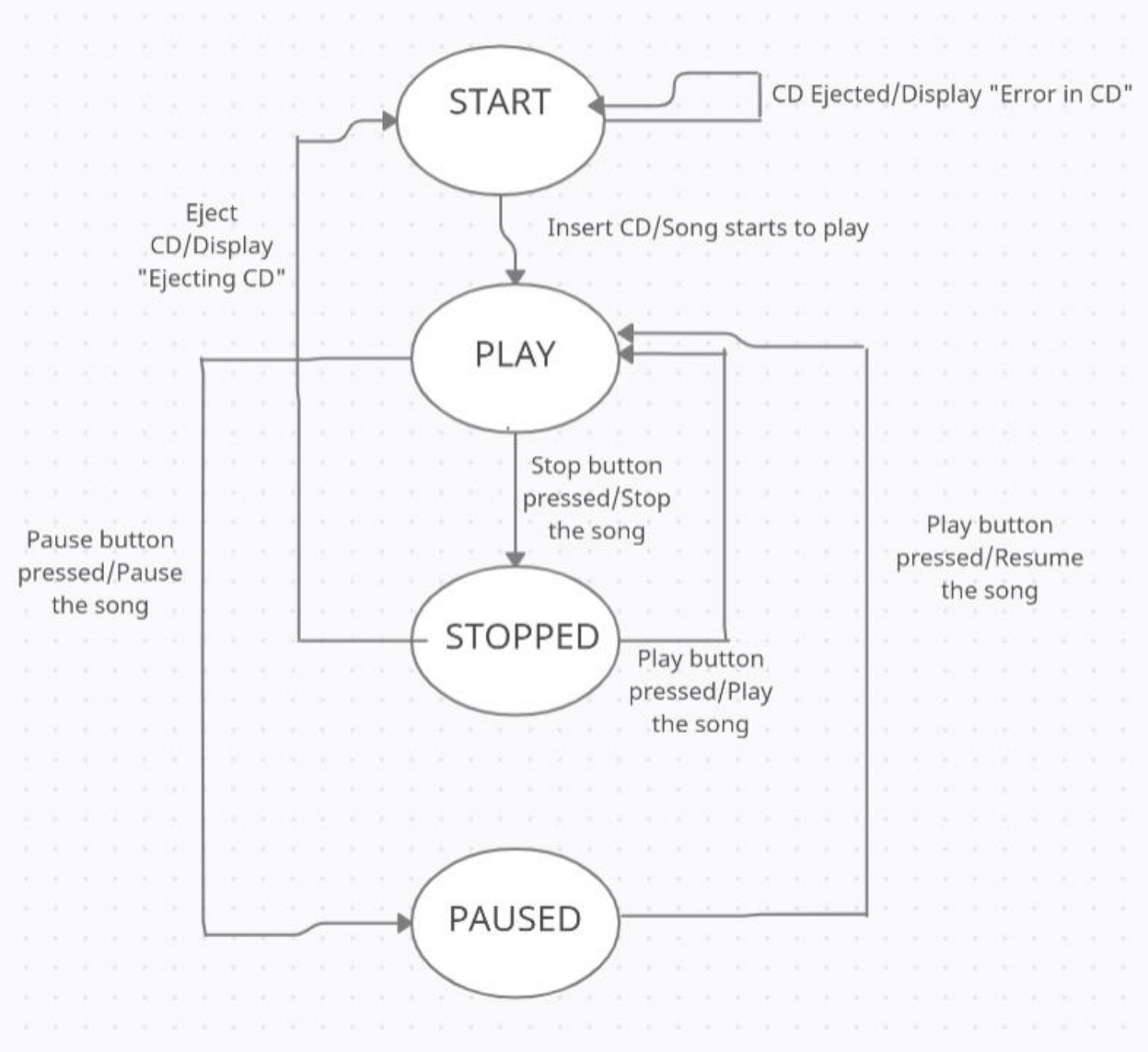
```
IF NOT LOGED IN :
        LOGIN OR REGISTER
ELES
        SEARCH QUERY IN QUESTIONS
        IF FOUND
                READ SOLUTIONS TO THAT QUESTION_ID FROM SOLUTIONS
                PRINT THEM CHRONOLOGICALLY
        IF NOT FOUND
                POST NEW QUESTION
                INPUT QUESITON STRING
                INPUT SUBJECT
                INPUT SEMESTER
                INPUT QUESTION TITLE
                REFERENCE TO ANY QUESTION PAPER IF ANY
                PRESS “POST THE QUESTION”
                STORE QUESTION IN QUESTIONS DATABASE
```

STATE TRANSITION DIAGRAM

ATM MACHINE



CD PLAYER


START
CD Ejected/Display "Error in CD"
Eject CD/Display "Ejecting CD"
Insert CD/Song starts to play
PLAY
Stop button pressed/Stop the song
Pause button pressed/Pause the song
Play button pressed/Resume the song
STOPPED
Play button pressed/Play the song
PAUSED

SRS-FUNCTIONAL REQUIREMENTS

R.1. REGISTRATION MODULE

FIRST-TIME USERS WILL HAVE TO REGISTER ON THE WEBSITE BY PROVIDING THE REQUESTED DETAILS. THERE ARE THREE TYPES OF USERS:

STUDENTS: STUDENTS NEED TO PROVIDE THEIR NAME, COLLEGE ROLL NUMBER, BRANCH, AND GRADUATION YEAR.

PROFESSORS: PROFESSORS MUST PROVIDE THEIR NAME, TEACHER ID, AND DEPARTME.

ADMIN: A NEW ADMIN USER CAN ONLY BE REGISTERED BY THE EXISTING ADMINS. THE ADMINS ARE RESPONSIBLE FOR MAINTAINING THE SOFTWARE, FOR UPLOADING QUESTION PAPERS, AND FOR MONITORING THE DISCUSSION FORUM

R.1.1. SIGNUP

INPUT : ALL THE INFORMATION THAT HAS BEEN MENTIONED IN THE DESCRIPTION AND A PASSWORD OF MINIMUM 8 CHARACTERS GENERATED BY THE USER.

OUTPUT : REGISTRATION CONFIRMATION MESSAGE AND UNIQUE USER_ID

PROCESSING : IF THE COLLEGE ID PROVIDED BY THE USER IS ALREADY REGISTERED, AN ERROR MESSAGE WILL BE DISPLAYED, OTHERWISE A NEW USER_ID WILL BE GENERATED.

R.1.2. LOGIN

DESCRIPTION : AFTER REGISTERING ON THE SITE, SUBSEQUENTLY THE USERS CAN LOGIN FROM NEXT TIME ONWARDS BY ENTERING THEIR USERNAME AND PASSWORD WHICH THEY ENTERED DURING REGISTRATION.

INPUT : ENTER USERNAME AND PASSWORD.

OUTPUT : USERS WILL BE LOGGED IN ON THE WEBSITE AND WOULD BE ABLE TO USE THE SOFTWARE FEATURES.

PROCESSING : IF THE USERNAME AND PASSWORD PROVIDED BY THE USER DO NOT MATCH, THE USER WILL RECEIVE A ERROR MESSAGE “INVAILD CREDENTIALS”.

R.2 PREVIOUS YEAR QUESTIONS

R.2.1 POST QUESTION

INPUT : USER WILL PROVIDE QUESTION STRING, QUESTION TITLE, SEMESTER, SUBJECT AND QUESTION PAPER YEAR.

OUTPUT : THE DOUBT WILL BE DISPLAYED IN THAT QUESTION PAPER SECTION WITH SEMESTER, SUBJECT AND QUESTION PAPER YEAR AS TAGS.

R.2.2 REPLY SOLUTION

INPUT: THE USER MUST PROVIDE THE SOLUTION STRING AND THE ATTACHMENT (IF ANY).

OUTPUT: IF THE USER WHO REPLIED IS A STUDENT, THEN THE SOLUTION IS ADDED TO THE DISCUSSION FORUM, AND IF THE USER IS A PROFESSOR, THEN THE SOLUTION IS DISPLAYED IN THE EXPERT SOLUTION SECTION.

R.3 FEED

R.3.1 SHOW FEED

DESCRIPTION :- THE QUESTIONS ARE SORTED BY THEIR DATES, AND ARE DISPLAYED IN REVERSE CHRONOLOGICAL ORDER.

R.3.2 GENERAL POST

INPUT : USER WILL PROVIDE QUESTION STRING , QUESTION TITLE , SEMESTER AND SUBJECT.

OUTPUT : A QUESTION WILL BE DISPLAYED IN THE FEEDS SECTION WITH SEMESTER, SUBJECT AND GENERAL AS TAGS.

R.4 SEARCH

INPUT : USER WILL PROVIDE QUESTION STRING , QUESTION TOPIC AND SEMESTER

OUTPUT : ALL POSTS MATCHING THE INPUT WILL BE FILTERED AND DISPLAYED IN THE SEARCH RESULTS SECTION.

NON-FUNCTIONAL REQUIREMENTS

USABILITY REQUIREMENT:

The software shall allow the users to use the website interoperably on laptops and smartphones. The software uses a web application as an interface. Since all users are familiar with the general usage of websites, no special training and configuration is required.The website will be user friendly. Users will be able to access the website provided that they belong to the institution for which this software is being developed.

AVAILABILITY & EFFICIENCY REQUIREMENT:

The software will be available for use 24 hours a day and 365 days a year.

Even if the software fails for some time due to unexpected errors, it will be recovered within 3-4 hours.

COST:

The software will be free to use and the users would only require the URL of the website and proper internet connection to access the website.

ACCURACY:

The website will accurately provide real time information on various discussion forums taking into consideration various concurrency issues. The system shall provide 100% access reliability owing to the importance of the user information stored in the database.

SECURITY:

In order to develop secure software, the database server and the website server will be different so that even if the website server is attacked, the database server with important information remains safe. The database would be protected by a firewall to deny access to traffic by default and prevent malicious SQL query injection. Further the data will be stored in an encrypted manner in the database and due backup will also be made. For transfer of data

from the website to the database Advanced Encryption Standard (AES) would be used to secure the passwords.

PERFORMANCE REQUIREMENTS:

The website will be refreshed every 5 seconds automatically or can be refreshed manually by user any time.The website shall respond to the user in not less than 5 seconds from the time of the request submittal. The website shall be allowed to take more time when doing large processing jobs.

STORAGE:

Databases would be stored on MongoDB and the back-end of the software would be developed using Node and Express.

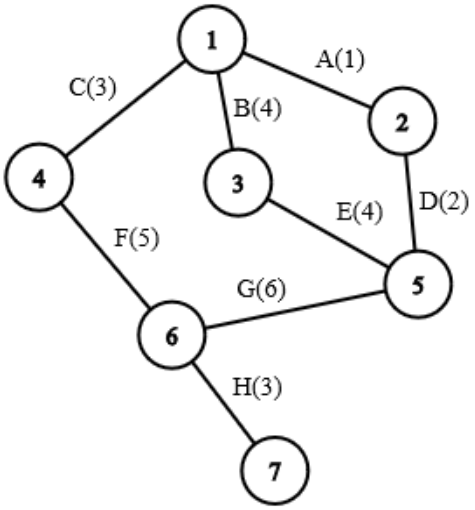
CRITICAL PATH METHOD

Activity	*Immediate Predecessors*	*Duration*
A : Finalising tech stack to be used	*-*	*1*
B : Wireframing of the project	*-*	*4*
C : Initializing Backend technology and database	*-*	*3*
D : Initialize Frontend framework , libraries and dev dependencies	*A*	*2*
E : Implementing basic templates for project	*B*	*4*
F : creating server	*C*	*5*
G : Completing Frontend	*D,E*	*6*
H : Connecting Frontend and Backend technologies	*F,G*	*3*

Edge and it's preceded and succeeded node

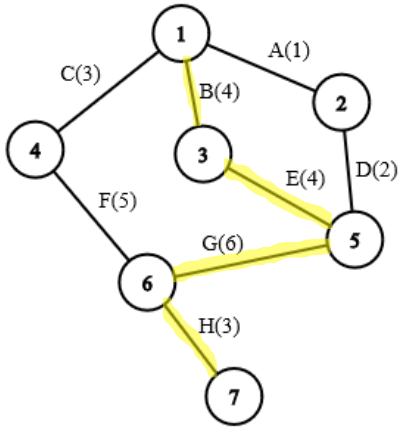
Edge	*Node1 → Node2*
A	*1→2*
B	*1→3*
C	*1→4*
D	*2→5*
E	*3→5*
F	*4→6*
G	*5→6*
H	*6→7*

The network diagram for the project, along with activity time, is



Activity (i,j) (1)	*Duration (tij) (2)*	*Earliest time Start (Ei) (3)*	*(Ej) (4)*	*(Li) (5)*	*Latest time Finish (Lj) (6)*	*Earliest time Finish (Ei+tij) (7)=(3)+(2)*	*Latest time Start (Lj-tij) (8)=(6)-(2)*	*Total Float (Lj-tij)-Ei (9)=(8)-(3)*	*Free Float (Ej-Ei)-tij (10)=((4)-(3))-(2)*	*Independent Float (Ej-Li)-tij (11)=((4)-(5))-(2)*
1-2	*1*	*0*	*1*	*0*	*6*	*1*	*5*	*5*	*0*	*0*
1-4	*3*	*0*	*3*	*0*	*8*	*3*	*5*	*5*	*0*	*0*
2-5	*2*	*1*	*8*	*6*	*8*	*3*	*6*	*5*	*5*	*0*
4-6	*5*	*3*	*13*	*8*	*13*	*8*	*8*	*5*	*5*	*0*

The critical path of the project is : 1-3-5-6-7 and critical activities are B,E,G,H



BOUNDARY VALUE ANALYSIS

INPUTS ARE DAY, MONTH AND YEAR. OUTPUT WILL BE THE DATE (IF VALID) OR "INVALID DATE".

Constraints:

1<=Month<=12
1<=Day<=31
2000<=Year<=2022

Test Case	***Month***	***Day***	***Year***	***Expected Output***
1	*6*	*15*	*2000*	*15 JUNE,2000*
2	*6*	*15*	*2001*	*15 JUNE,2001*
3	*6*	*15*	*2011*	*15 JUNE,2011*
4	*6*	*15*	*2021*	*15 JUNE,2021*
5	*6*	*15*	*2022*	*15 JUNE,2022*
6	*6*	*1*	*2011*	*1 JUNE,2011*
7	*6*	*2*	*2011*	*2 JUNE,2011*
8	*6*	*30*	*2011*	*30 JUNE,2011*
9	*6*	*31*	*2011*	*INVALID DATE*
10	*1*	*15*	*2011*	*15 JANUARY,2011*
11	*2*	*15*	*2011*	*15 FEBRUARY,2011*
12	*11*	*15*	*2011*	*15 NOVEMBER,2011*
13	*12*	*15*	*2011*	*15 DECEMBER,2011*

CYCLOMATIC COMPLEXITY PROBLEM

```
1   begin int x, y, power;
2   float z;
3   input(x, y);
4   if(y<0)
5   power = -y;
6   else power = y;
7   z=1;
8   while(power!=0){
9       z=z*x;
10      power=power-1;
11  } if(y<0)
12  z=1/z;
13  output(z);
14  end
```