1. INTRODUCTION :

This “NITC query system” is a web application that is designed and built to give the students of NITC a platform where they can have discussions about anything related to the campus. The pandemic has forced us to follow an online system of education for the past two years. This meant that the students did not get a chance to interact with each other, have conversations about important issues on campus and build a strong and interactive student community, which is an integral part of one's college experience. The B20s did not even get a chance to meet each other even once. They do not have any information about the ins and arounds of the campus. In general, Whenever a new student joins NITC, they lack a platform where they can get all their doubts or queries rectified. Whether you are a new joiner to NITC or a final year student, everyone has many doubts and questions which can be rectified by asking others. Especially in the case of new joiners, they hesitate to ask as they don't know anyone. The Junior-Senior bond is also weak among them. So we came up with this platform where anyone can ask anything publicly. Any student can ask any question on this website while others can help him/her by answering those questions. Hence, We aim at building a platform where any of these queries can be answered and any type of discussion can take place with complete transparency.

The design of the website is such that students can easily ask any question, answer any question, and also read from the previously asked question. In addition to this users have their profile page where they can see their questions and answers.

The design of the database is designed to store the information about the question-answer and students. The database is used to show question-answers in reverse chronological order that is latest comes first. The data is stored in SQL. Access to data will be through a web interface running on a web server and using Node.js.

The driving philosophy behind the database design was to have an efficient, normalized database that would be easy to maintain and expand, as well as allow easy data entry and access. Easy deletion and update of data are also taken under consideration.

2. PURPOSE :

This is a System design Document for the “NITC query system” website for the NITC. This document is intended to provide the implementation information about the website.This document describes about the software and hardware specifications and also about the database design. Database Design maps the logical data model to the target database management system with consideration to the system’s performance requirements.

This document can be used for:

- Knowing the information about the software selection
- Understanding the various hardware architecture, system design, website architecture, hardware interface, and use case model.
- Knowing the basis for application’s database design
- Provide expected data volumes, functional/non-functional usages of tables.
- Ensure database transaction meets or exceed performance requirements.

  The Database Design converts logical or conceptual data constructs to physical storage constructs (e.g., tables, files) of the target Database Management System (DBMS).

2.1 DOCUMENT OBJECTIVES :

The System Design Document has the following objectives:

- Understanding the various hardware architecture, system design, website architecture, hardware interface, and use case model.
- To serve as basis for implementing database and related software units.
- It provides basic idea of design to acquirer and help in designing and developing software required for the application.
- Describing design of database that can be accessed by users or computer developers.

2.2 Intended Audience:

This system design document is intended for all NITC students, including system designers and developers.

- This system design document will help students who would be posting the Questions on website to know all the features.
- The developer would be benefited by knowing the requirements of the client.
- The designer can use this system design document to design the system to meet all requirements.
- The admin can get an idea of their role and special features they are provided.
- The database designer can use this for designing the database according to needs of the project.

- Quality Assurance personnel, whose test cases must validate the requirements specified in this document.

2.3 Document Conventions

User	Someone who uses or interacts with the web application
Admin	Who manages and organizes the Question answer website
Query-Website	A web application consisting of Questions and their answers typically displayed in reverse chronological order.
DBA	Database Administration.

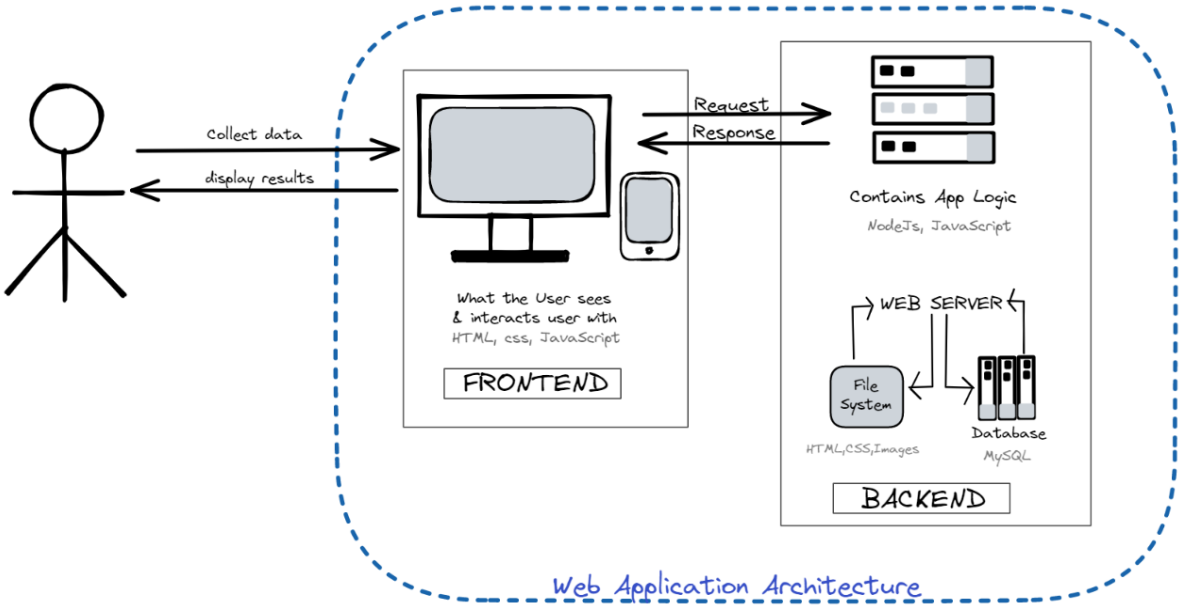
3. SOFTWARE AND HARDWARE SPECIFICATIONS

3.1 Software Selection :

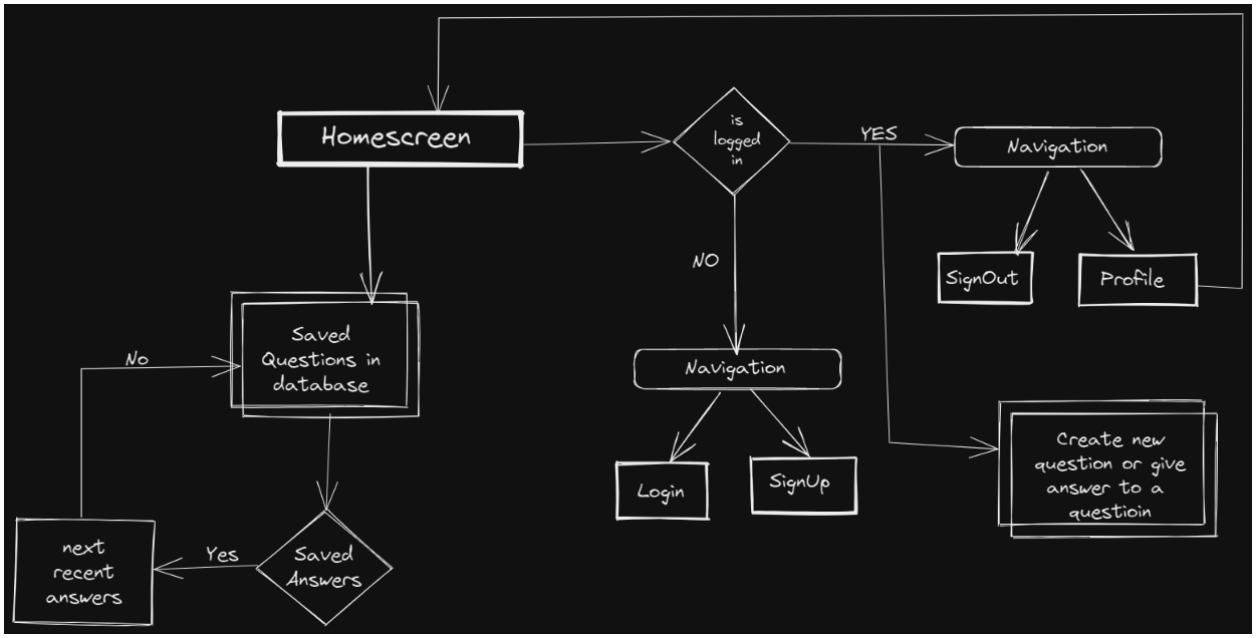
Following reviews of the data that would be entered into the database, and the requirements of retrieving the data several criteria were identified that need to be met by the database software. The minimum requirements for the software were:

- There must be restrictions on data values entered in the columns of relation. By restricting type and range of data values entered we can control data we get and hence reduce error.
- Multiple user access to the database simultaneously must be allowed. Since multiple students are allowed to post queries simultaneously, many users will be posting at the same time. Further many users will be retrieving data from the database at same time.
- Allow data entry from the internet. Most of the data will be entered into the forms from the internet.
- Allow triggers on the data tables. Triggers will allow predetermined actions to be taken when information is entered, edited, or deleted from a data table.
- Should support the relational database model, and at least basic version of SQL.
- Allow running of stored SQL scripts. There are many processes that can be automated with stored scripts, to facility management, updates, editing, and querying of the database. This is especially important if users are accessing the data from the internet. Being able to call and run a stored script is far easier and more efficient than trying to code all the information into a web form.
- Allow joins within a query. The joins are different ways of selecting items from one or more tables, in either a query or a view.

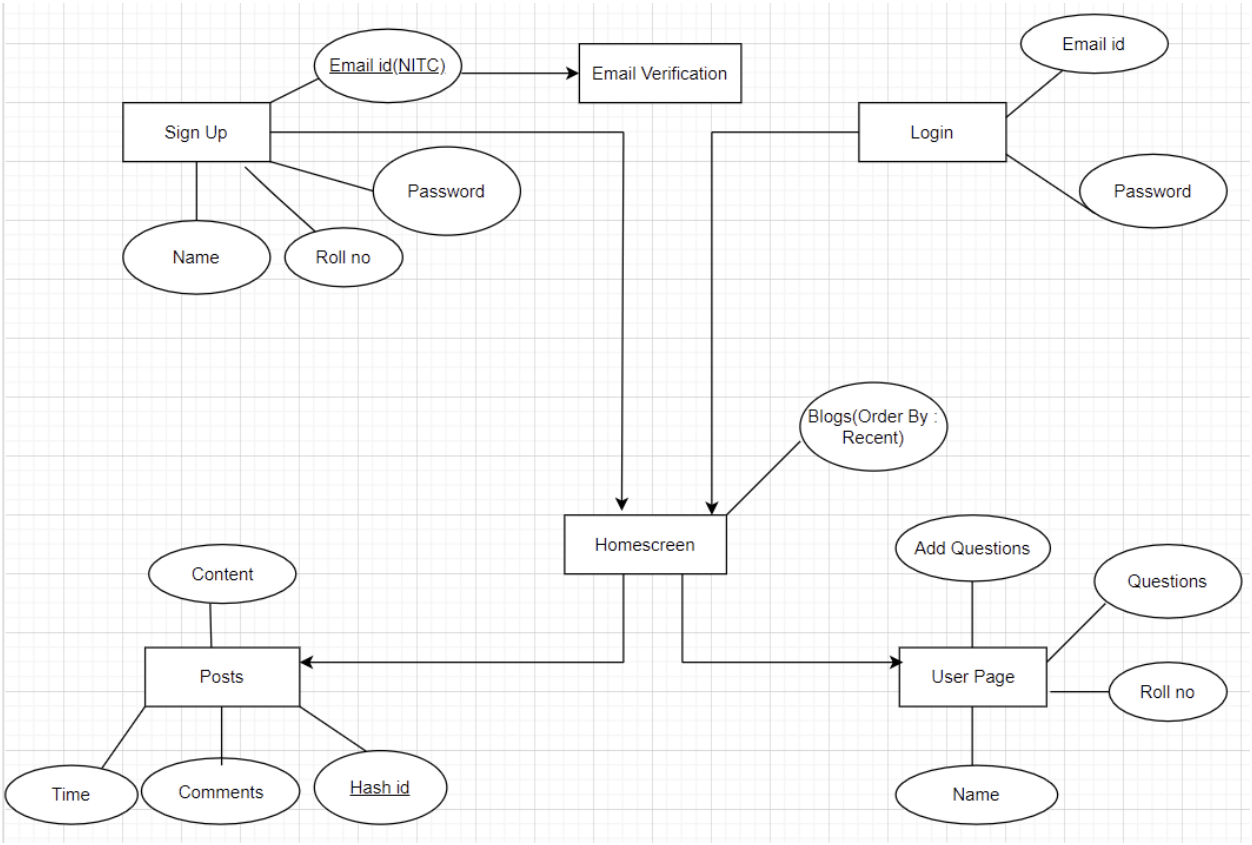
3.2 HARDWARE ARCHITECTURE :



3.3 WEBSITE ARCHITECTURE :



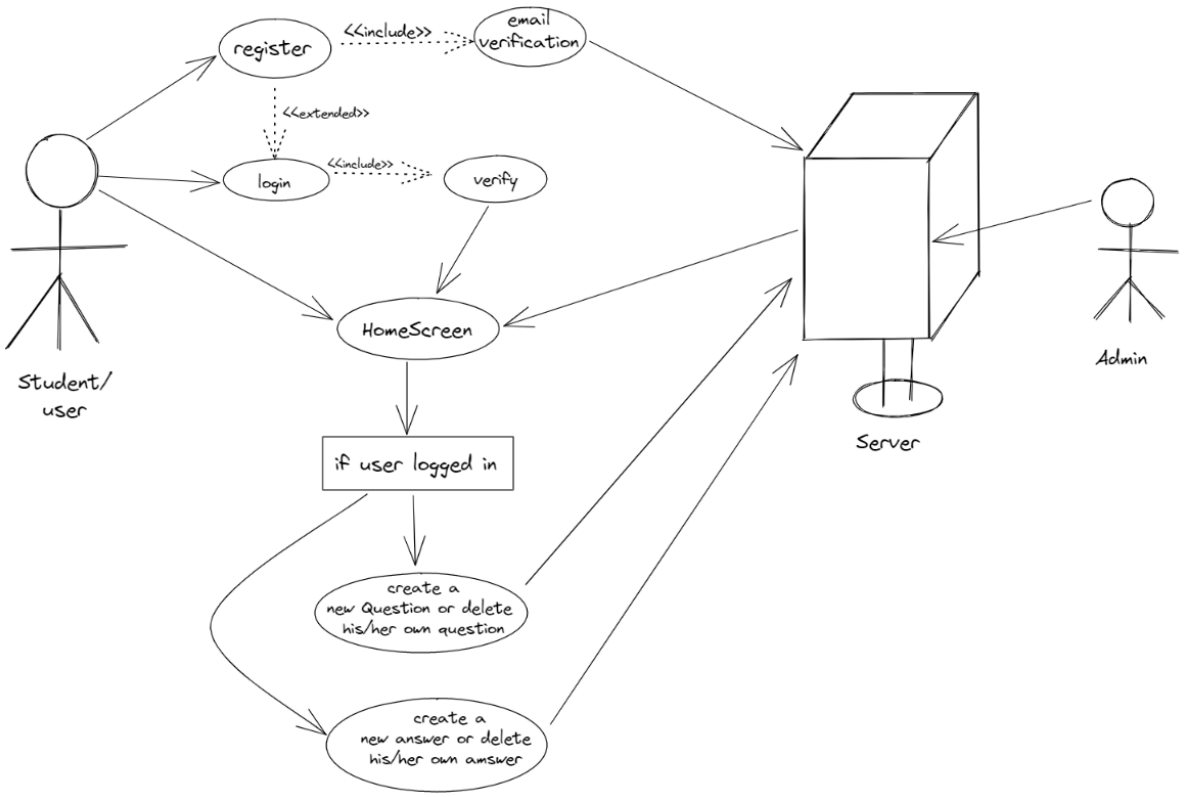
3.4 System Design:



3.5 HARDWARE INTERFACE :

- Since the application runs over the internet, the system must always be connected to the internet at the time the user is running the application.
- Devices (computer, mobile, tablet) to access the site are required.
- Backup storage for retrieval of data in case of unexpected failures.

3.6 CASE MODEL DESIGN:



4.Other Non-functional Requirements

4.1 Performance Requirements

- A server is required to run this web application.
- The database is required to store the user details and their questions and answers on this web application.
- The request response time will depend on network speed and latency of the server.

- So, a server capable of handling approx. 1000 concurrent requests is required.
- The site load time depends on speed of internet connection and proximity from the server.
- The load time may even depend on type of hardware on which site is running.
- For decreasing the time in email verification, the email will be sent instantly. To avoid the load on the server the mails be sent using the parallel jobs.
- In every request created by the user whether it is posting questions, deleting questions or answers, the information is always a string. The data type dealt in question answers is a string.

4.2 Safety and Security Requirements

- To have full functional use of the website user need to sign up or sign in (if signed up), otherwise the website remains to be in read-only mode.
- At the time of the sign-up user is asked the name and roll number. This is to ensure the correct identity of the user.
- Unique Login Id (only NITC mail id people allowed so as to avoid outsiders) and password (may or may not be unique) for every user.
- Every new user signing up is checked in the backend for the nitc mail-id, other ids are not allowed.
- To have real users on websites, email verification is done at the time of sign-up.
- The signed-In user gets signed out after the specified time as signed-in credentials are stored in the cookies for a specified time.
- Only Signed In users can post a new question, answer to asked the questions from the home page, and upvote answers.
- Signed Out users can only see the question and answers to them. The website remains to be in read-only mode.
- Signed-In users have access to their Profile Page which shows all the questions asked by them also shows the answers associated with corresponding questions.
- Signed-In users can only delete those questions and answers which they wrote.
- Data backup to be done periodically.

- Only Signed In user can post a new question or answer to asked the questions from the home page.
- Signed Out user can only see the question and answers to the them.
- Signed In user have access to their Profile Page which shows all the questions asked by the user also gives user feature of deleting the questions along with all the associated answers to it or deleting the unwanted answer to the questions.
- User's id and password are secured and no visible to even admins.
- Users have option to save their login credentials.
- All Users have unique id and passwords.
- Password will be hidden while typing.

Sign out and change password features are given to users.

4.3 Software Quality Attributes Reliability

- New Questions and corresponding answers to them in database are reflected without any delay.
- Any registration by the user is reflected in database almost instantly.
- All the actions to post and delete happen by signed users.
- Questions are shown in reverse chronological order, hence newest Question will be shown first.
- Application is made to protect inside information from users, only few selected people have access to handle all features of blog website.

4.3.1 Usability

- In case of invalid signup (out of NITC) user is taken back to the home screen without registering him or her.
- To have the accessibility to feature to the website, the user should sign-in.
- This application works in every browser.
- This application is responsive, so works great with every size of screen.

4.3.2 Correctness

- To provide only reliable content in the Question only few people are allowed to handle admin panel.

- To provide genuine answers to questions to questions, users have can refer to upvotes to the answers.
- Admin have full control to manipulate Questions and Answers. So in case someone post inappropriate question or writes the inappropriate answer then Admin can delete them instantly without the user choice.
- Admin panel which is accessible to a few specific people, check the database regularly.

4.3.3 Maintainability

- Backup of database will be kept.
- SQL server would be having a database maintenance routine scheduled monthly.
- Periodic testing and debugging will be done based on users' feedbacks. Also, the application is designed such that it can incorporate the new features without hassle.
- New features will be added with time to have an easy user-friendly experience
- In case of any error, re-initialization of program will be done.

5.Design Decisions

5.1 Factors affecting the Design :

- The registered users will only be able to post questions.
- The NITC mail id of entered by the users should be verified.
- The questions will be ordered by their time of posting.
- The users will have a user page where they will be able to see the questions posted by them and their details.

5.2 Database Details :

- Database used for the web application will be MySQL.
- The server version is 5.7.36-log.

5.3 Backup and Maintenance :

- Backup of database will be kept.
- SQL server would be having a database maintenance routine scheduled monthly.
- Periodic testing and debugging will be done based on users' feedbacks. Also, the application is designed such that it can incorporate the new features without hassle.
- New features will be added with time to have an easy user-friendly experience
- In case of any error, re-initialization of program will be done.

5.4 Design and Implementation Constraints :

- The users are provided with a sign-up where they enter their NITC mail id and set up a password.
- The mail id is verified through email verification.
- After their sign-up/login-up they will be redirected to a home page where recent Questions and Answers would have listed.
- The users will also have a user page where they will be able to see the details of their profile.

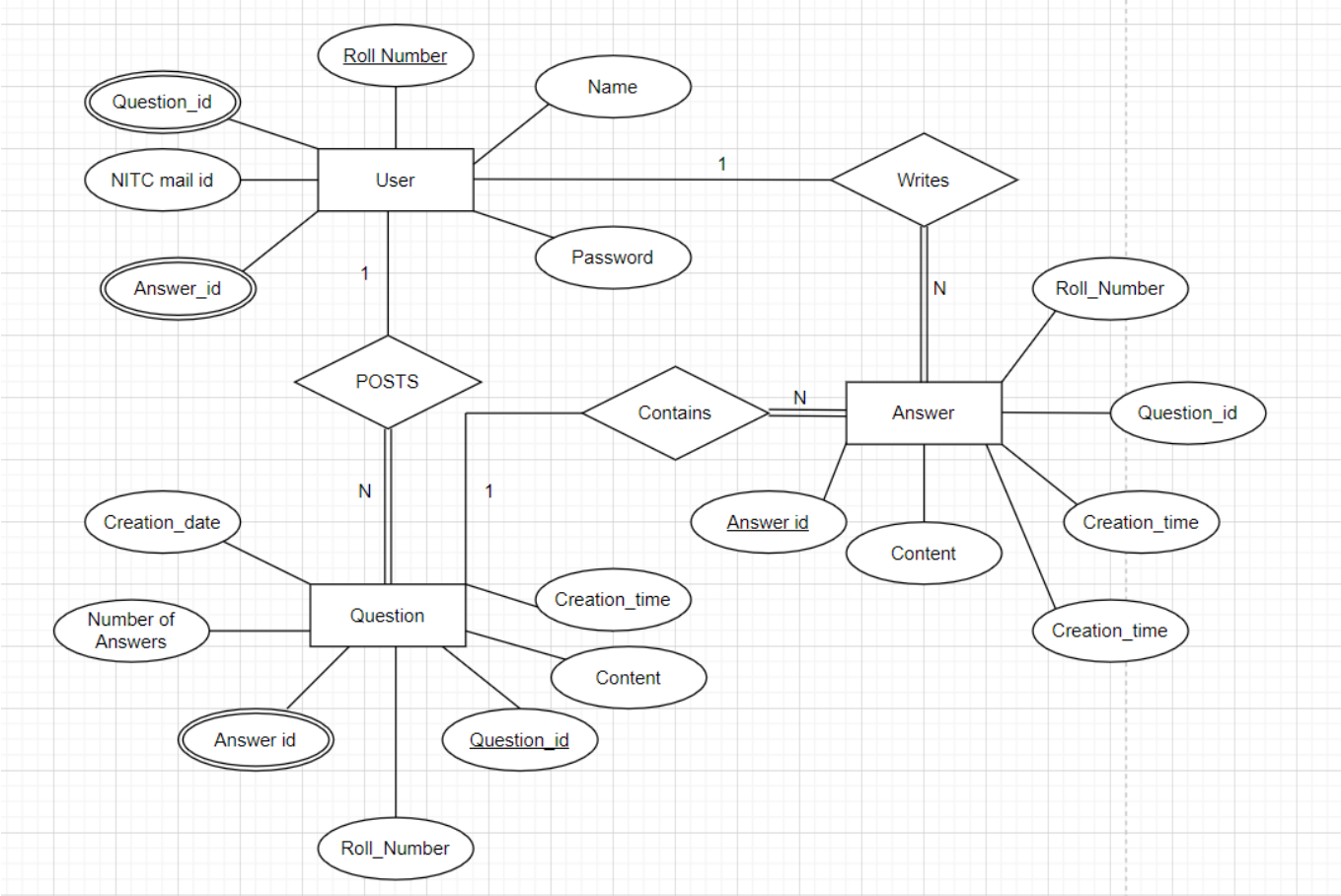
- The users will have the option to post a question and answer to others questions.
- Implementation of the database will be through a centralized database system.
- The major challenge after the design is to scale it according to the number of users in order to provide a decent response time for read and write requests.

5.5 Security :

- Since the registration portal accepts only NITC mail id so the users will from NITC only.
- External changes to the database can be made only by the admin.

6.Database Design :

6.1 Entity – Relation Model :



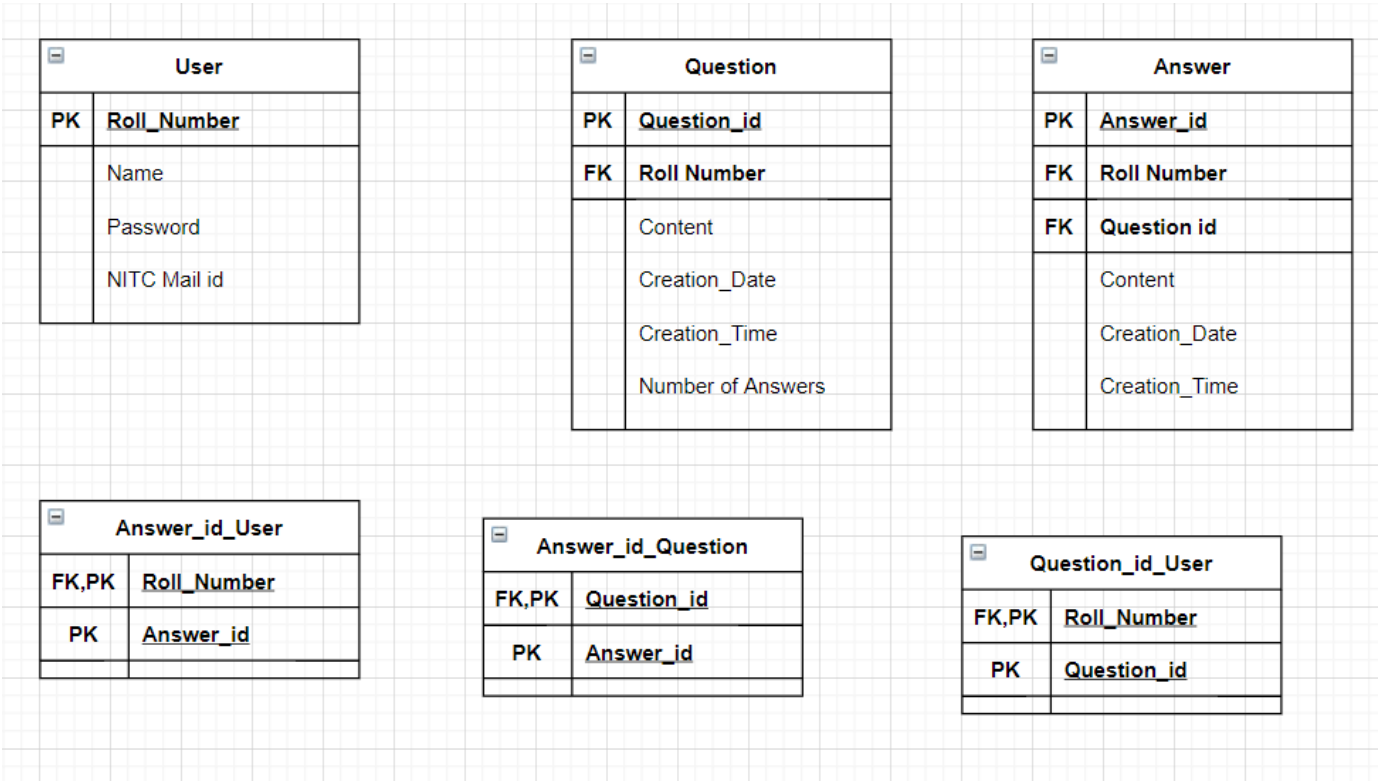
Entities –

- **User :** Primary key – Roll Number
- **Question :** Primary key – Question id
- **Answer :** Primary key – Answer id

6.2 Relational Model :

Notations –

- **PK** – Primary Key
- **FK** – Foreign Key



6.3 Table Description :

User Table :

Attribute	Type	Description
Roll Number	VARCHAR(9)	Roll Number(which is unique) of every student.
Name	VARCHAR(20)	Name of the user.
Email id	VARCHAR(20)	The NITC mail id of the user.
Password	VARCHAR(20)	Password for the user's account.

Question Table :

Attribute	Type	Description
Question id	INT	The unique question id for each question.
Roll Number	VARCHAR(9)	Foreign-key,Roll Number of user who posted the question.
Content	VARCHAR(200)	The content of the question.
Creation Date	DATE	The date on which question was created.
Creation Time.	TIME	The time at which question was created.
Number of Answers	INT	The count of answers of the question.

Answer Table :

Attribute	Type	Description
Answer id	INT	The unique answer id for an answer.

Roll Number	VARCHAR(9)	Foreign Key, the Roll Number of the user who posted the answer.
Question id	INT	Foreign-key, the question of the answer.
Content	VARCHAR(200)	The content of the answer
Creation Date	DATE	The date on which the answer was created.
Creation Time	TIME	The time at which the answer was created.

Answer Id Table for User (The answers that the user has posted).

Attribute	Type	Description
Roll Number	VARCHAR(9)	The Roll Number of the user who posted the answer.
Answer id	INT	The answer id of the answer.

Answer Id Table for Question (The answers that the question has).

Attribute	Type	Description
Question id	INT	The question id of the question to which the answer belongs.
Answer id	INT	The answer id of the answer.

Question Id Table for User (The questions that user has posted).

Attribute	Type	Description

Roll Number	VARCHAR(9)	The Roll Number of the user who posted the question.
Question id	INT	The question id of the question.