

Software Engineering (IT314)

Course Project Report

EdNet

Educational Networking Tool

Course Instructor: Prof. Saurabh Tiwari



Dhirubhai Ambani Institute
of Information and Communication Technology

May 7, 2021

Table of Contents

Functional Requirements	4
Non-functional Requirements	5
Use Case Model	6
User Stories	7
Activity Diagram	9
Class Diagram	10
Sequence Diagram	11
Deployment Diagram	12
Testing:	13
GUI and Usability testing:	13
Black box Testing:	16
Requirements that are not implemented	21
System Architecture	21
Why Use Node Js?	21
Why MongoDB?	22
Why Profcode and Admincode?	22

Group Members

Vishwa Shah	201801036
Rishabh Makwana	201801090
Devanshu Chicholikar	201801099
Himanshu Ramoliya	201801156
Ved Joshi	201801171
Jainit Patel	201801172
Vismay Tandel	201801228
Aditya Goyal	201801253
Madhavendra Choksi	201801255
Aman Popatia	201801451

Problem Statement

The objective of our project is to provide a platform for students across the globe to interact with others with similar interests. The students from different colleges can take part in discussion and problem solving. It will be a tool where students can network among themselves and with professors from different universities. We wish to provide a one-go platform for academicians to learn and grow. The students can post their doubts on discussion forums and get them solved by the community. The professors can find passionate collaborates easily by posting their project openings on the project forum. This forum is great for students who are interested in exploring research options.

Process Model: Iterative Waterfall

Functional Requirements

1.1. Students

- A student user can log in to the portal by making a profile on the website.
- A student user can choose their areas of interest, which can be modified by the user whenever required.
- The user can post a question in the discussion forum of a particular topic and can reply to other threads.
- A student user can go on the project page and explore different opportunities in their areas of interest. They can sign up for a project/research posted on the page.
- A student user can upvote/like a question and an answer to a thread.
- A student user can view other profiles and can get access to information (like email id, contact) mentioned on the profile.
- A student user can sign up for an email push notification for a particular discussion forum to stay updated.

1.2. Professors

- Professor can log in to the portal by making a professor profile.
- Professors can create a thread for particular fields.
- Professors can answer students' questions in the discussion forum.
- A professors' thread will be shown differently from other users i.e students in order to distinguish faculty/professor from students.
- A professor has the rights to post projects/research opportunities on the project forum.

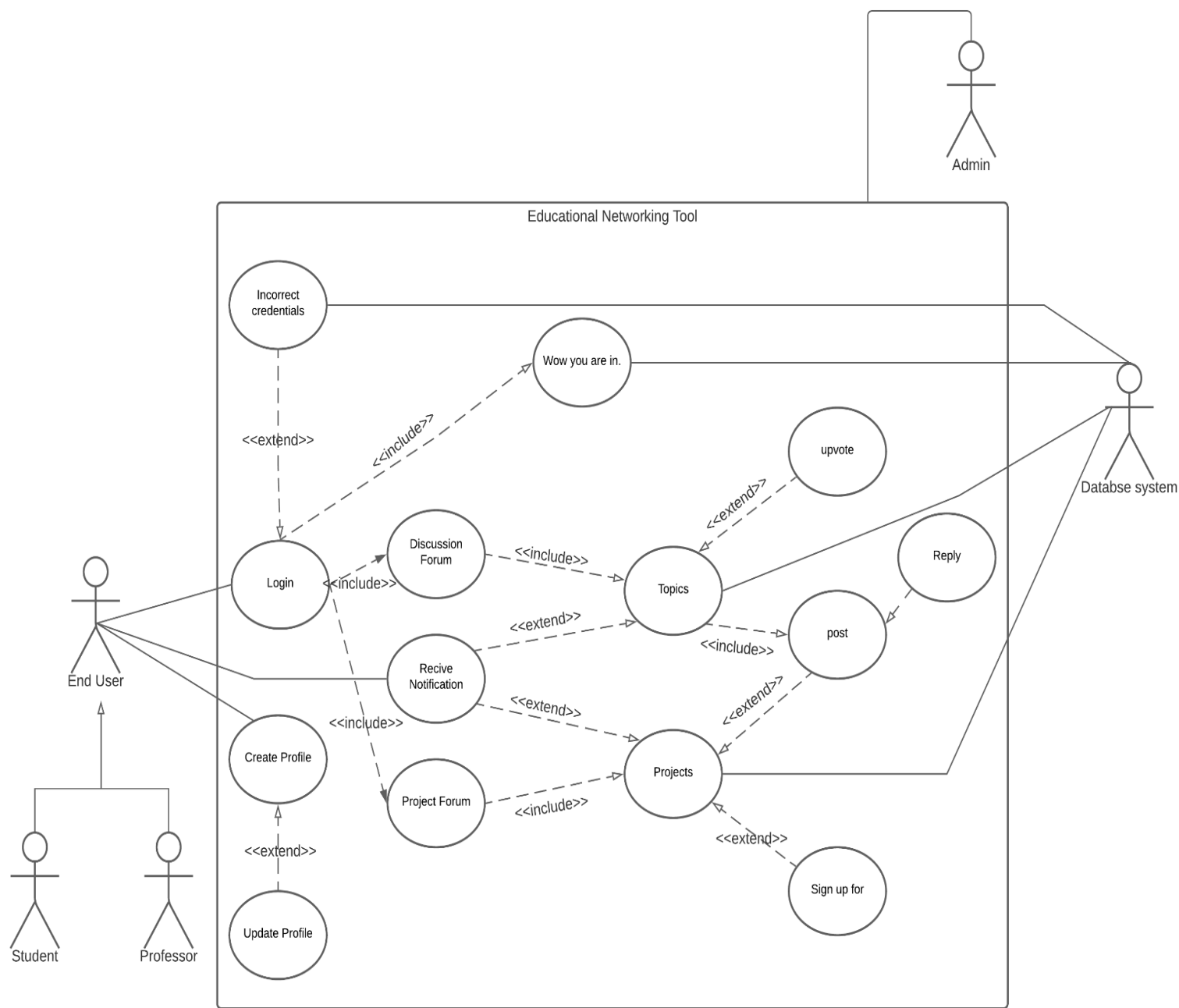
1.3. Admin

- Admin can remove/ban any user (student/professor) for any kind of misbehaviour.
- Admin can edit/delete chats in the discussion forum
- Admin can approve student/professor accounts so that no fake account can be created.

Non-functional Requirements

- The website should be compatible with any browser such as Google Chrome, Mozilla Firefox, Internet Explorer, Safari etc.
- The website should return the search result entered by the user in less than 5 seconds.
- The user will have to login again every 24 hrs if he/she is not active.
- The forums will be updated automatically as a new thread is posted.
- If the user forgets his/her password. Then there will be an option for “forgot password” which will take the user to a window for the further process to retain their account back.

Use Case Model



User Stories

User Stories(Prioritized according to MoSCoW method)

Note:

- We have assigned alphabets to the user stories based on the following sequence.
Must have - 'M'
Should have - 'S'
Could have - 'C'
Would have - 'W'
- The term 'user' is commonly used for student/professor in section 5.1.

5.1 User

1. **M** - As a user, I must be able to make a new profile on EdNet.
2. **M** - As a user, I must be able to log in to EdNet.
3. **M** - As a user, I must be able to access the discussion forum and project forum as well.
4. **M** - As a user, I must be able to post my questions or blogs in the discussion forum.
5. **M** - As a user, I must be able to answer any of the questions from the discussion forum.
6. **S** - As a user, I should be able to update my profile whenever I want.
7. **S** - As a user, I should be able to choose the topics which are of my interest.
8. **S** - As a user, I should be able to upvote/downvote comments on any of the questions posted in the discussion forum.
9. **S** - As a user, I should have the option to get notified through email if anything is posted under the topics chosen by me.
10. **S** - As a user, I should be able to request account recovery (for setting up a new password).
11. **C** - As a user, I could be able to view other user's profiles and get access to information such as e-mail id or phone number.
12. **W** - As a user, I would be able to edit the questions posted by me at any time.
13. **W** - As a user, I would be able to chat with other users.
14. **W** - As a user, I would be able to mark a question as duplicate.
15. **W** - As a user, I would be able to report if a question posted under a given topic is of some other topic.
16. **W** - As a user, I would be able to report any user if he/she uses foul language or is found to harass any other user.

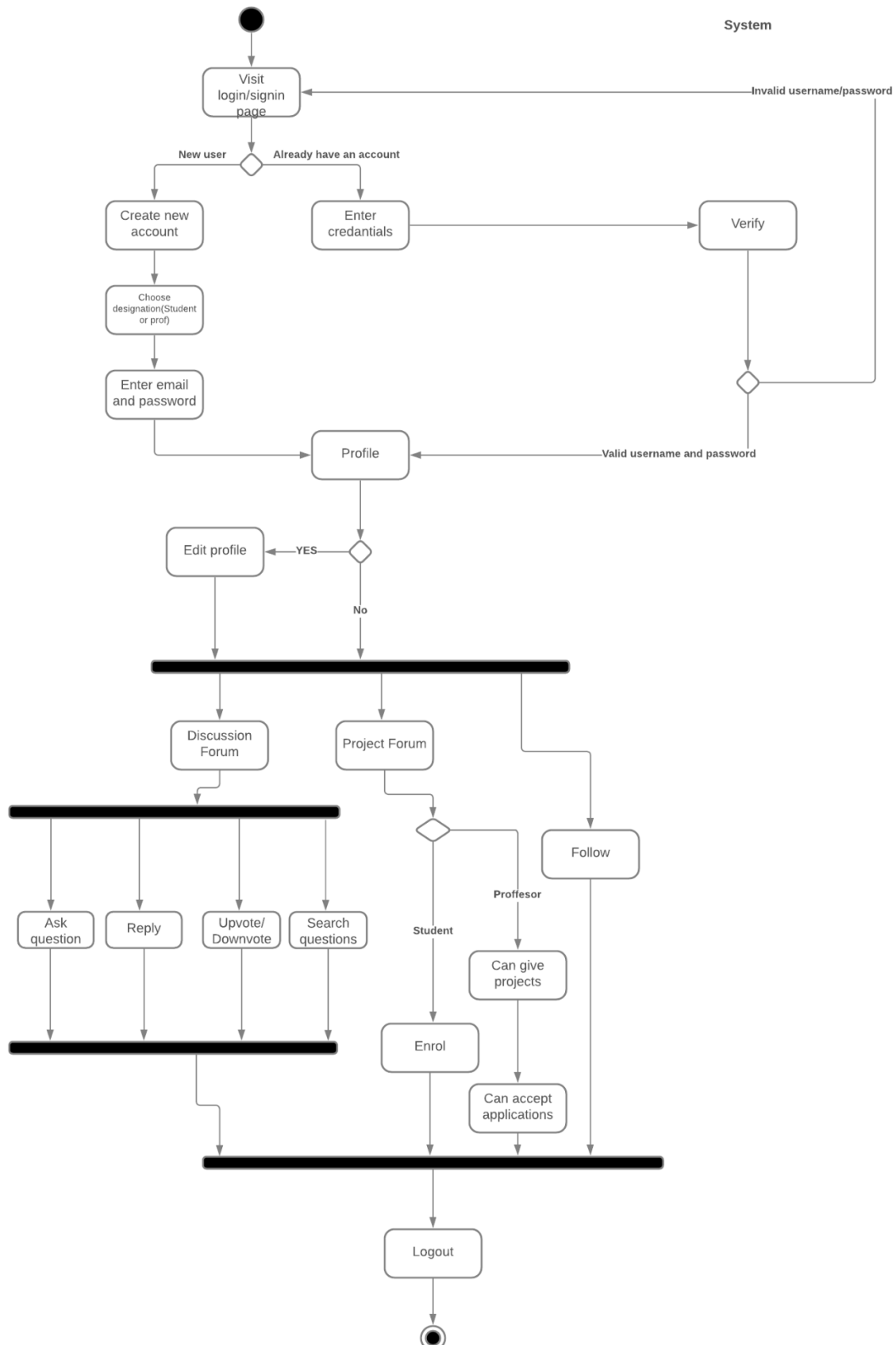
5.2 Professor

17. **M** - As a professor, I must be able to post project topics in the project forum.
18. **S** - As a professor, my profile should be different from the student's profile.
19. **C** - As a professor, my upvotes/downvotes and my answers could be distinguishable from other users.

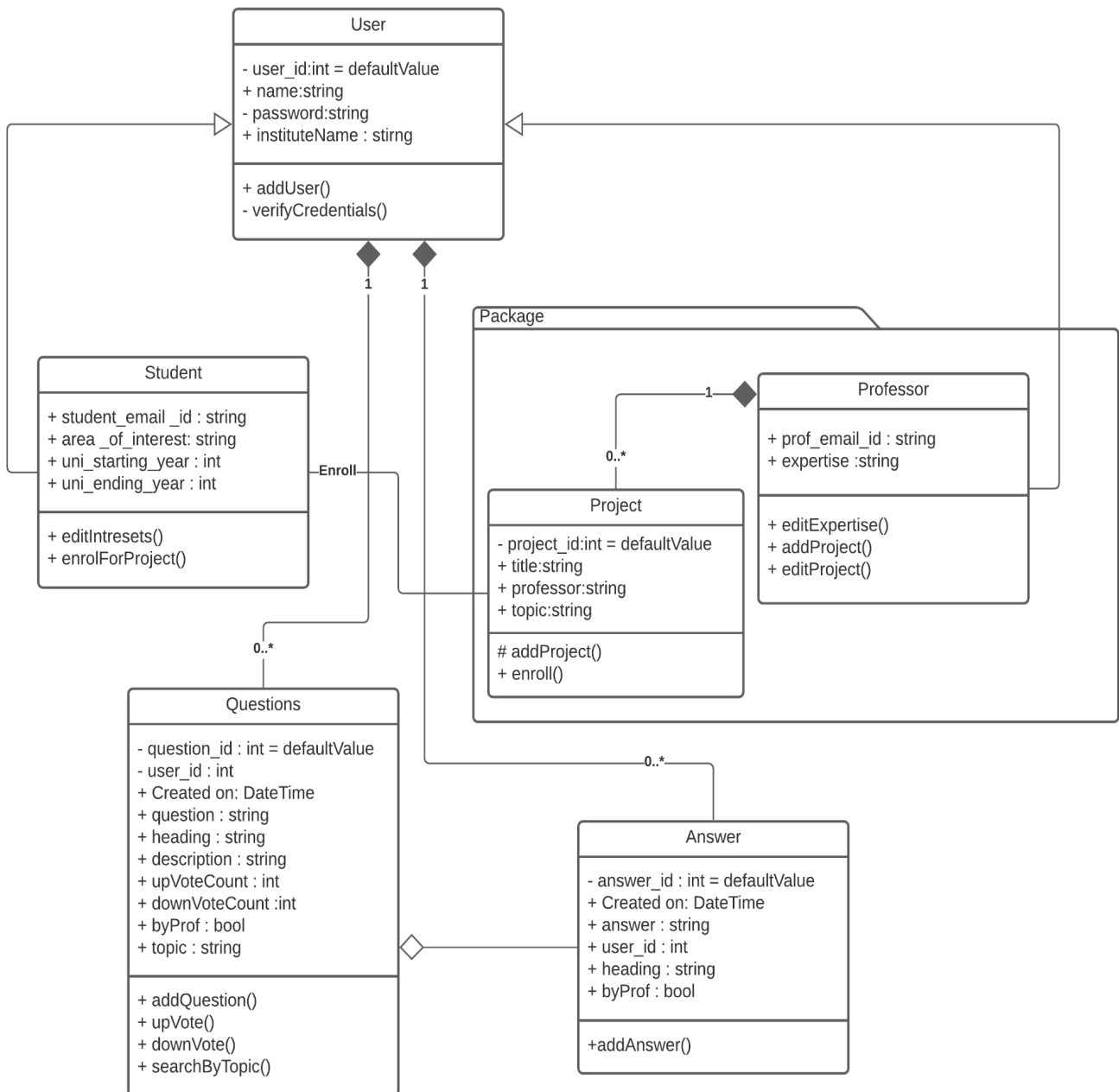
5.3 Admin

- 20. **M** - As an administrator, I must get access to add topics.
- 21. **M** - As an administrator, I must recover the accounts of any of the user, in case they have forgotten their passwords.
- 22. **S** - As an administrator, I should get access to the database of all users.

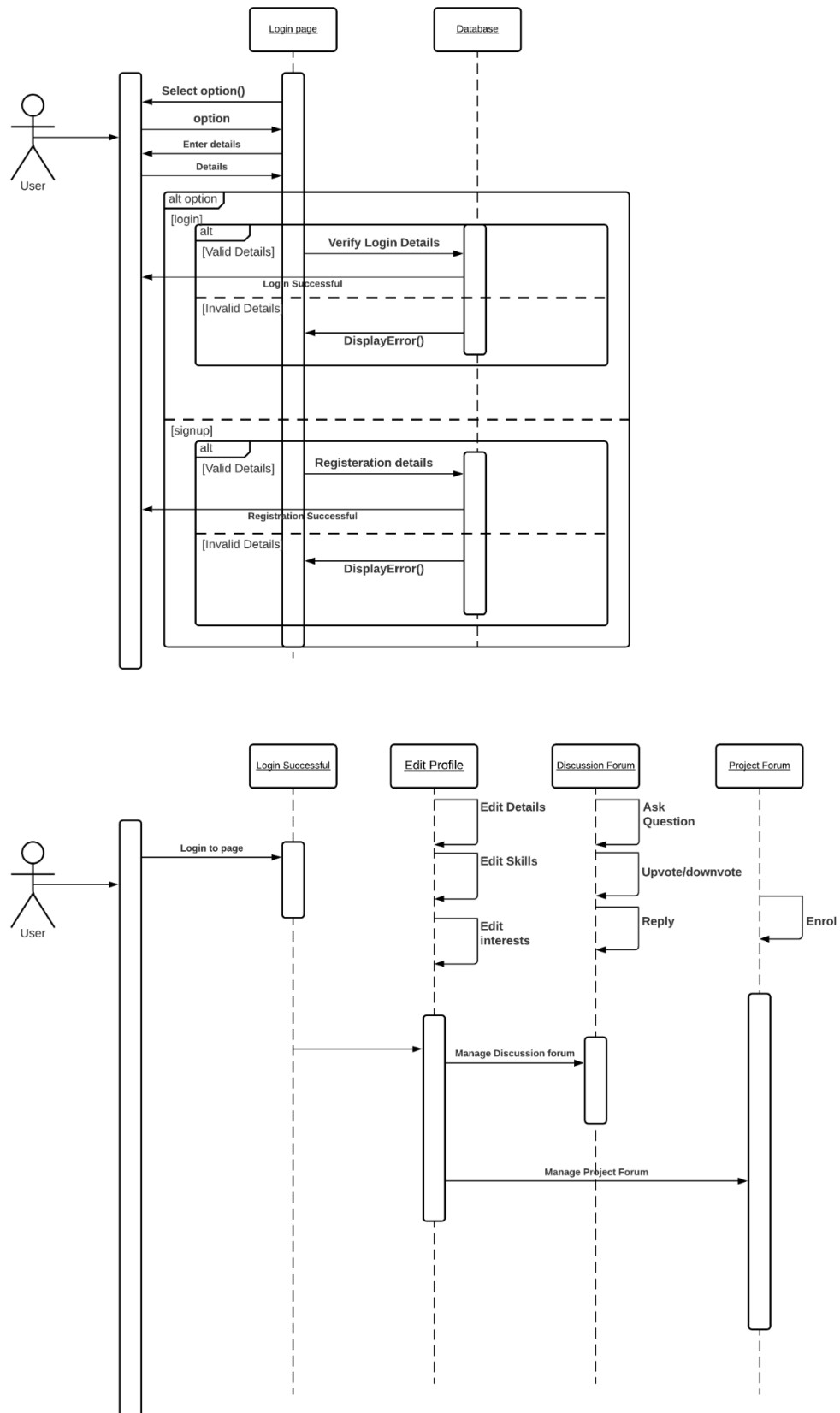
Activity Diagram



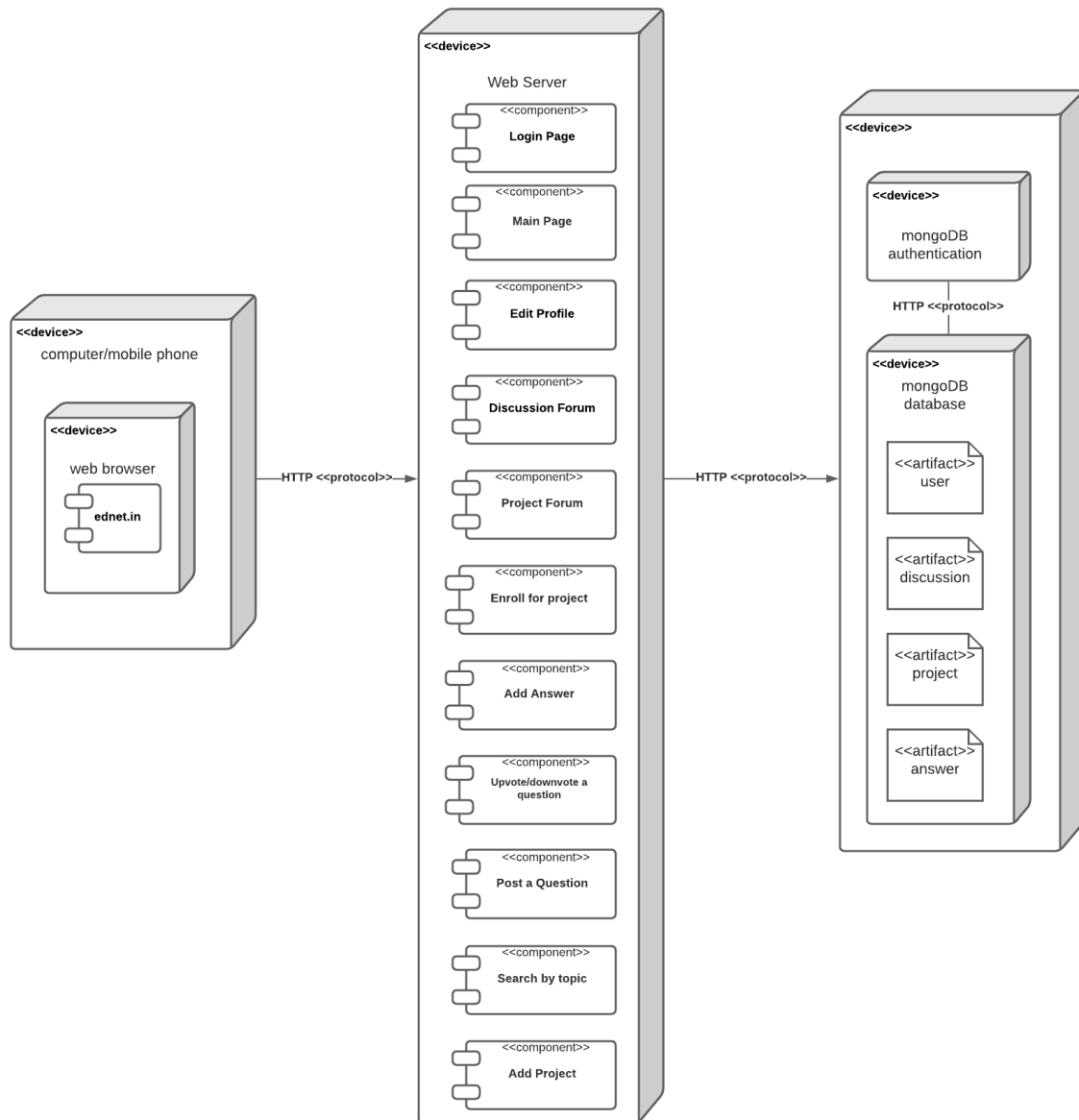
Class Diagram



Sequence Diagram



Deployment Diagram



Testing:

GUI and Usability testing:

Deployment 1:

Page:	Issues:
Home, Contact Us	Logo dimensions are not correct, change font size, navigation bar button should change colour when we hover over them. Font not visible due to the background image, reduce its opacity.
Contact-us	Change the logos, add an effect to the Email-Id, contact information.
Login page	EdNet logo blurs out while zooming, font not appealing to eyes, ratio of the logos need to be adjusted.
Create and Account page	No login button to go back to No buttons to go to the home page No popup when the account successfully created. No way to know which fields are compulsory
Profile page	Change the appearance of button, profile and EnNet icons, fields inconsistent with discussion forum.
Discussion Forum	No pop-up when question added, question added goes to the bottom of the page, there should be a signout button at the end of the page too. Field/tag button too large and text below

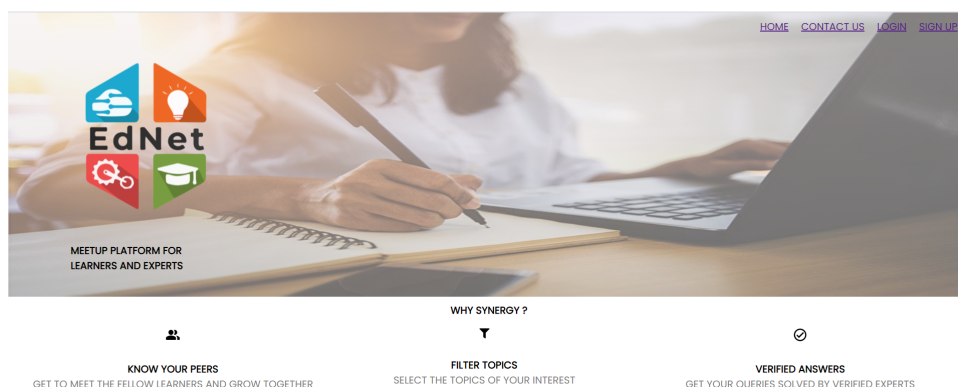
	<p>not visible. Inconsistencies in the fields and spelling errors that need to be corrected.</p> <p>Enter key feature not working for the search box.</p>
Project Forum	<p>No way to know if we have successfully enrolled for a project or if a professor has successfully added a question.</p> <p>Enter key feature not working for the search box.</p>
General	Nav-bar not there on all pages

Final Deployment:

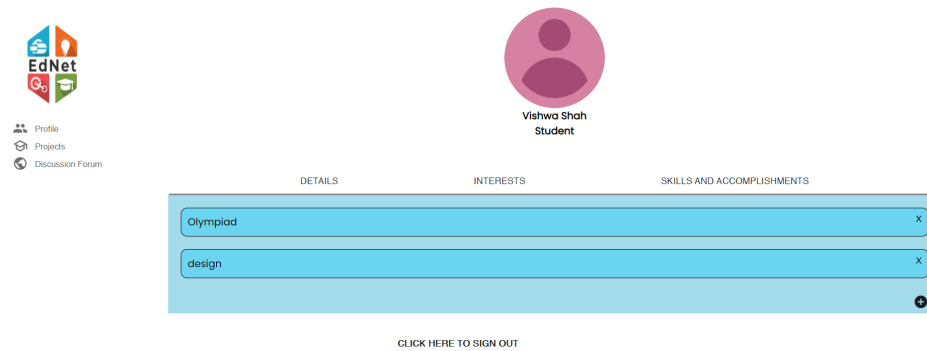
Some screenshots of changes done to the UI after testing

Main Page:

Logo dimensions changed, background image made opaque and changes to the nav-bar buttons



Profile Page:

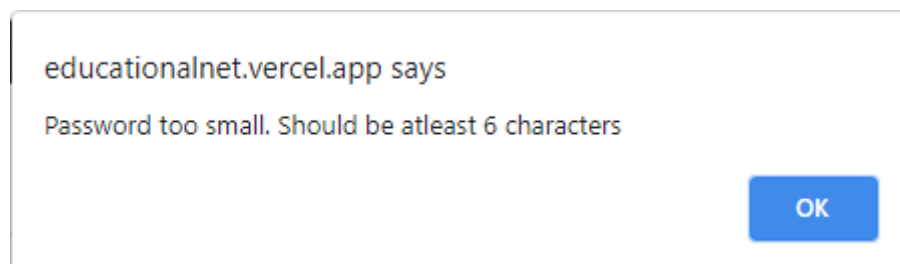


Changes that needs to be taken care of and suggestions after final deployment:

Page:	Suggested changes
Home, Contact Us	Text in the upper section could be aligned and underlined The content in the lower(white section) could be changed for a better appeal
Contact-us	Better effects can be applied. Image can be inserted and hovering on the image might show the details could have looked better.
Discussion Forum	Enter key feature could be made working for easier use.
Project Forum	Enter key feature not working for the search box.
General	Nav-bar not there on all pages can be implemented for easy use.

Black box Testing:**→ Create an Account (Student)**

- ◆ All fields are necessary.
- ◆ The “Password” and “Confirm Password” must be the same or else an error message will be displayed.
 - Eg : User - Password : HelloWorld
Confirm Password : Hello
Status : **Error** (Incorrect password)
- ◆ The “Terms and Condition” checkbox is a must or an error message will be displayed.
- ◆ Same first and last name can be used.
 - Eg : 1. Account 1 - First name - Random, Last Name - Student
2. Account 2 - First name - Random, Last Name - Student
Status : **Accepted.**
- ◆ Passwords must have more than or equal to six letters or digits in it.
 - Eg : Password - abcd
Status - Error (Must be more than six letters)



- ◆ No same email address can be used. It should be unique for every student.
 - Eg : Account 1 - Email - helloworld@gmail.com
Account 2 - Email - helloworld@gmail.com
Status : **Error** (Email Address already exist)

→ Create an Account (Professor)

- ◆ A “Professor Code” will be provided by the admin by which the professor can login into his/her account.
- ◆ All other points are the same as students.

→ Profile Page (Student)

- ◆ All the details of the student are shown under the “Details” section.
- ◆ A user can edit profile details as per his/her choice.

- ◆ All the fields are to be necessarily filled while editing details of the student.

InstituteName:
Daiict

Batch Start:
2018

Batch End:
2022

Enrollment Number:
|

SUBMIT

[CLICK HERE TO SIGN OUT](#)

- ◆ Under the “Interests” the student can add/delete his/her interests. They are provided with a list of options from which they can choose.

Details
Interests

Maths Remove

Choose a topic ▼

Choose a topic

Web Development

Machine Learning

Micro Controller

Physics

Maths

- ◆ Under the “Skills and Accomplishments” the student can add/delete their achievements.

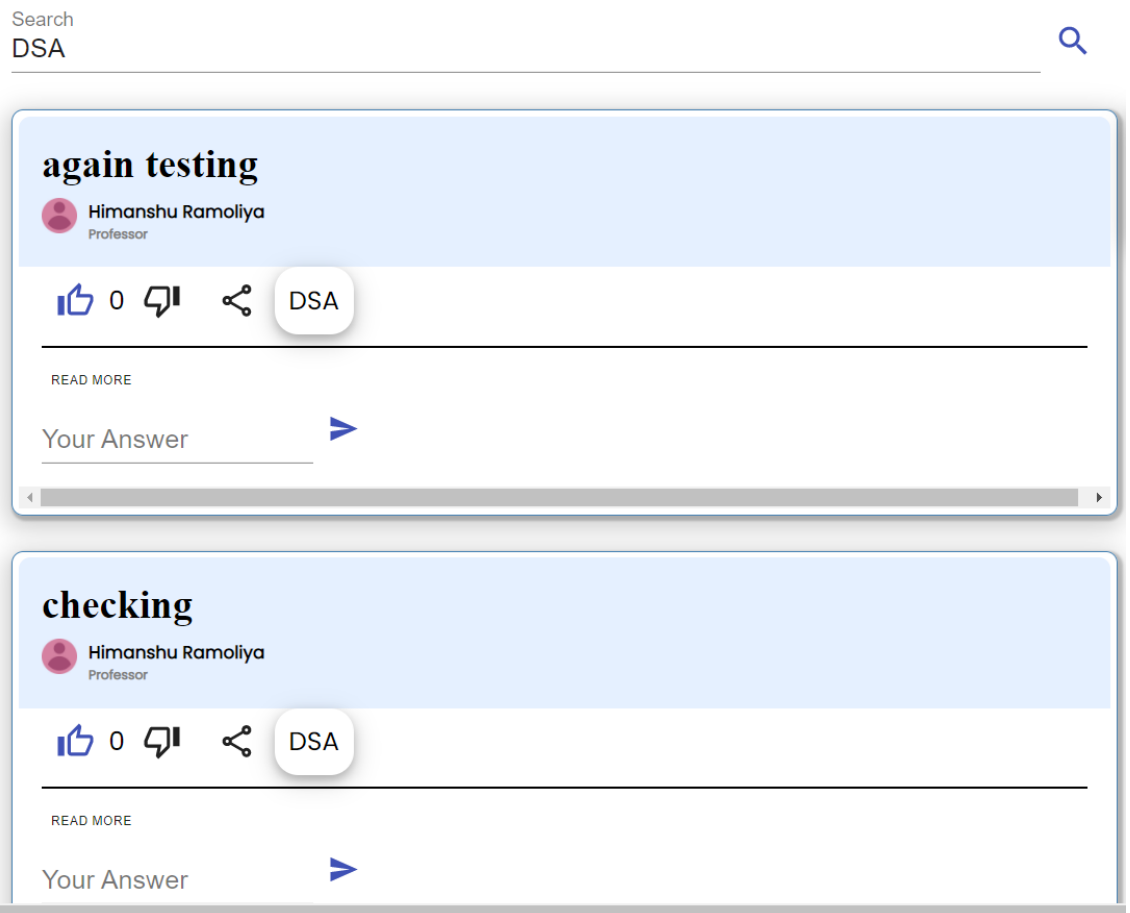
→ Discussion forum

- ◆ Students can ask questions in the forum and can also answer them.
 - An Ask Question box appears on the top left corner of the discussion page.
 - After clicking a student must choose a topic from which they have the question related to, and from the given options.
 - The student can write their question in the given area.
 - A box pops up saying “Question successfully added”.

- ◆ Students and professors can comment on the questions asked.
- ◆ Users can also upvote/downvote a question/answer.

- ◆ A professor's answer on a question has a priority over students

- ◆ Search feature will search specific topic in all Questions/Discussions

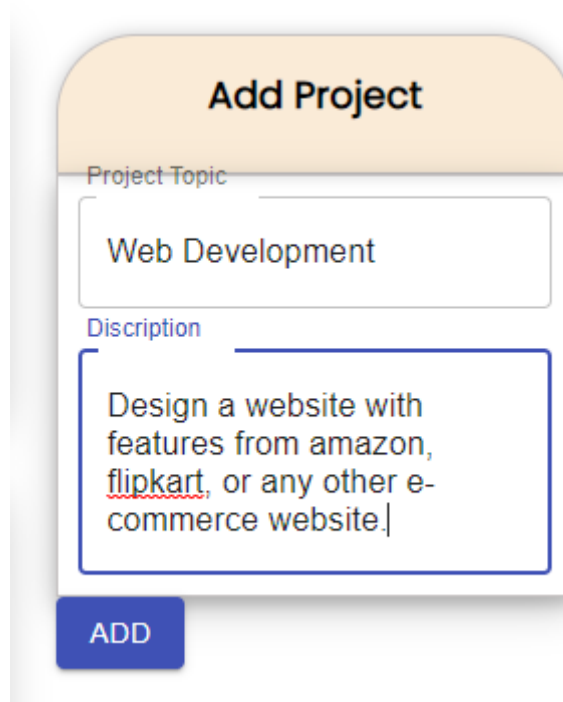


- ◆ After adding a Question/Answers You will receive a notification



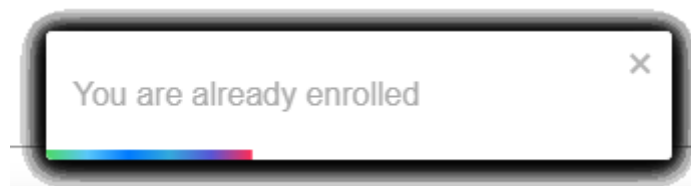
→ Projects

- ◆ Here the professors can post about project work.



The image shows a web form titled "Add Project". It has a light orange header with the title. Below the header, there are two input fields. The first is labeled "Project Topic" and contains the text "Web Development". The second is labeled "Discription" (note the spelling) and contains the text "Design a website with features from amazon, flipkart, or any other e-commerce website.". Below the description field is a blue button labeled "ADD".

- ◆ Students can enroll for the project work. And for enrollment student will receive notification and if already registered then will received notification like below



- ◆ And also has same search feature like discussionForum

Requirements that are not implemented

- Feature for upvoting/liking a question has been implemented but a particular answer cannot be upvoted.
- The feature where users can chat with each other has not been implemented yet.
- A student cannot view other profiles currently, search option has been implemented for discussion and project forums but not for profiles.
- We could not implement the notification feature, notifications were to be sent using emails, so currently this feature isn't working. Another feature not implemented by us is sending notifications to the professor when someone applies to his/her project.
- Login with third party applications couldn't be implemented. We wanted to add login with google, facebook and linkedin accounts.
- We have not given the provision for the administrator to create an account and hence the requirements specified by us earlier such as remove an user, edit or delete threads is fulfilled by the developers in the back end part itself.

System Architecture

Why Use Node Js?

JavaScript as programming language and data format (JSON) has changed web development drastically. Integrating Node.js with it to do things on the server as well as in browsers is a trend lately. So basically Javascript is everywhere, so that gives us one of the main reasons to use Node Js, as it uses Javascript unlike Django and similar environments which use different languages.

Node Js is very fast as it uses the V8 engines by Google. Google V8 engines compile javascript into a native machine code and runs very speedy and because of it Node.js increases any framework's speed.

We can make highly scalable applications with a great speed and Node.js is also useful for agile and iterative development.

As we use JavaScript for both frontend and backend the code is more consistent and cleaner, which is another advantage of Node.js.

Also Node.js has the largest ecosystem of open source libraries which makes our task easy and simple so this is also a great benefit of Node.js.

Why MongoDB?

MongoDB is a document-oriented NoSQL database that we can use for storing the high volume of data.

Database Management systems of this type use dynamic schemas. This scheme means that you have the option to create records without giving the definition to the structure. This is the biggest advantage of NoSQL databases and which makes them powerful. MongoDB gives you permission for changing the structure of the records. We can call this document after we add new fields or delete the already existing one.

In today's world of Web Development as we mentioned earlier, .json data format is changing the scene drastically and MongoDB makes the representation in the form of JSON documents. On the other hand, MySQL represents the data in the form of tables, rows, and columns. So MongoDB has an upper hand over MySQL.

Also, MongoDB uses JavaScript as the query language so no need to learn another language for the purpose of the database collection, storing and queries.

Why Profcode and Admincode?

We can imagine that what happens if all users i.e. students and professors have the same status and benefits, surely the application will be misused as it is pretty reasonable to think. So we have come up with the simpler and less tedious way. As the application is used by the college, the administration is the only one who has prior knowledge of the profcode and the admincode, so when any professors wants to join the ednet the administration will provide the profcode without which you can't login as a professor.