Group Project Log

Group Name:	Tech Intellect

Group Members:	4

Deliverable:	Technical Report

Group Member Name	Work Done (%)
Shrey Amin (B00822245)	25
Chintan Patel (B00826089)	25
Harsh Pamnani (B00802614)	25
Deep Shah (B00796368)	25
Total:	**100%**

TECHNICAL REPORT

Tech Intellect
RateMyDalCourse

Members and Contributors

Shrey Amin
B00822245
shrey.amin@dal.ca

Deep Shah
B00796368
dp371796@dal.ca

Harsh Pamnani
B00802614
hr340096@dal.ca

Chintan Patel
B00826089
chintan.patel@dal.ca

Faculty of Computer Science
Dalhousie University

31 July 2019

ABSTRACT

RateMyDalCourse is a platform where students can get the reviews and ratings for courses offered during each term based on other students' feedback. The most common target user base of our web application in Computer Science Students. We will cover technical and design aspects of the web application in this document. We have also provided justification for the tools we have used. By the end of this report, users will have insights on what is RateMyDalCourse all about and how it is designed.

KEYWORDS

Courses, Dalhousie, Dashboard, Discussion Froum, Education, MEAN, Posts, Rating, RateMyDalCourse, Students

CONTENTS

TABLE OF FIGURES

1. INTRODUCTION

At Dalhousie University, several courses are offered by the computer science department in each term at graduate and undergraduate level. The problem is that the students do not have a clear idea and information about the course outcome and the course content. Further, students do not have contacts with their senior students. So, this can create a dilemma among the students regarding which course to take and what will be the learning outcome. Thus, we are addressing this scenario to create a centralized where students can help each other by providing authenticate information about the courses [1].

The primary purpose of our web application is to provide a platform for upcoming and current Dalhousie students. We have developed an application for students of computer science background [2]. We have created a discussion forum for undergraduate and graduate students. Our application is referred as "**RateMyDalCourse**." This portal will also assist aspiring students who wish to start their graduation at Dalhousie University. We are developing this application specifically for students in the computer science department [3]. The aim of our web application is to provide a centralized platform where students can discuss regarding different courses offered at the undergraduate and the graduate level in the computer science department. Students can make posts, share their ideas and reviews regarding the courses. Overall, the idea is to allow students to rate the courses they have taken and solve their doubts and queries regarding the subjects they wish to take in upcoming semesters [4].

We have completed following features as a part of this project [5]:

1. Registration: Firstly, all users need to register by filling registration form which asks their various details. Once the user is registered, they can log in by providing email and password.

2. Login: User needs to login to access discussion forum and dashboard. User needs to provide the same credentials which they used for registration. After a successful login, the user can access the entire website by traversing from one page to another.

3. Profile management: User can add the courses which they have completed using this feature. Also, they can update their registration information, such as Email and password.

4. Search course: There are numerous subjects on the dashboard for which the user can provide the ratings and create or view the post. It is difficult for a user to search the subject manually. This includes a search bar which allows the user to find the subject easily using the subject name or course code.

5. Rate course: It might be possible that two different professors have taught one subject. This feature allows the user to rate the course and professor by giving the rating from one to five, which is useful for many students who want to learn this subject in the future term.

6. Create post: Users can create a post which provides different subject related information such as overall course structure, learning outcome, technologies used, assignment and project related information. Also, if students want to know any information related to the particular subject, they can ask by posting the question.

7. Display overall rating: Every student wants to know what kind of feedback is given to the subject which they are going to study. This feature enables the user to check the overall rating of the subject based on the feedback of past students who have already studied the subject.

8. Mark post as helpful: This is one of the essential features of this web application. Users can mark any post as helpful, if they find the information useful. All the posts will be displayed in the descending orders based on the timestamp [1].

1.1 Live Project URL

Bluenose Server URL:

http://129.173.22.34:24111/home

Git Repo Link:

https://git.cs.dal.ca/chintan/ratemydalcourse_group3

2. BACKGROUND

2.1 Competitive Landscape

RateMyProfessor.com: This web application provides reviews of different professor. Students can provide feedback from the professor by filling a review form. This web application does not have a proper layout and structure. As a result, the User Interface of the application is not attractive. It does not provide information on the courses taught by the professor [6].

RateMyTeachers.com: In this web application students can search for different teachers from specific countries like Canada and Australia. The website lacks a proper navigation structure and content hierarchy. A new user cannot identify the purpose of the application. Apart from this, it provides the details of very few professor or teachers [7].

However, the major difference between the above web applications and RateMyDalCourse is, we plan to focus more on the courses and not on the professors specifically. Moreover, we intend to provide a structure and effective web application with a rich user interface. Our platform will connect students and share their experience regarding the learning outcome of the courses.

2.2 Problem and Approach

At Dalhousie University, numerous courses are offered by the computer science department. The problem is that the students do not have a clear information about the course outcome and the difficulty level of the course before registering for the course. Hence, the primary purpose of our web application is to provide a platform where students can get insights on various courses and connect with other students if they have any doubts regarding the course. Students can rate the courses they have completed in their previous terms and also participate in the discussion forum of any subjects in which they might be interested in [1].

3. APPLICATION DETAILS

Our primary goal is to develop a web application which will provide information on every subject of Computer Science at the Dalhousie University by providing course ratings and discussion forum. All the student of Dalhousie Computer department (Graduate and Under Graduate) and student who is planning to study at this university can access this website to gain knowledge before selecting their curriculum [5]. Thus, our goal is to convey the information of different courses in such a manner so that student can easily decide whether the subject will be helpful to them or not and also get an idea about learning the outcome of the particular subject. Hence, our web application will eradicate the communication barrier among the student and resolve their queries using the discussion forum.

3.1 Target User Insights

The target audience for our web application is students of Dalhousie University. Usually, students have to contact the seniors through message or call to get the review of any particular subject. This might become a cumbersome process for each term if the student is taking four courses each semester. Furthermore, it is not always possible to find a senior who has completed a similar subject in previous terms. RateMyDalCourse web application will bring all the students together and help one another regarding the subjects' reviews [4].

There is no restriction on the age or gender of the user. Therefore, anyone who is a registered student with Dalhousie University will be able to use this platform. As a pre-requisite, the users are required to have Dalhousie account, because this web application is only for Dalhousie University Students. User does not need any specific knowledge to use the web application, as we will keep the design of the web application simple, which is easy to follow by the novice users [4].

3.2 User-Centered Design Approach

The user-centered design aims to improve user satisfaction with the web application. It includes Usability, Accessibility, System Performance, Learnability and Visual Design. RateMyDalCourse web application will consider navigation, familiarity, error prevention, feedback, and consistency to make the web application easy to understand for the users [8].

The web application will be used by Dalhousie University students to share the review of the courses offered in the university. The web application will maximize the use of Neilsen's Components of Usability to make the User Experience smooth and consistent. With the simple and aesthetic design of the web application, even novice users will be able to create and find helpful posts.

The website will make the registration process simpler by asking only minimal details from the user. We will follow the "2-Second Rule" for the registration process, where users don't need to wait more than 2 seconds for the same. The error message in our web application will be clear and to the point, which will help the users to recover from the error. The

logically related content in our web application will be visually related. For example, all posts on the discussion forum will have the same outline for the card, as shown in figure-1 below.

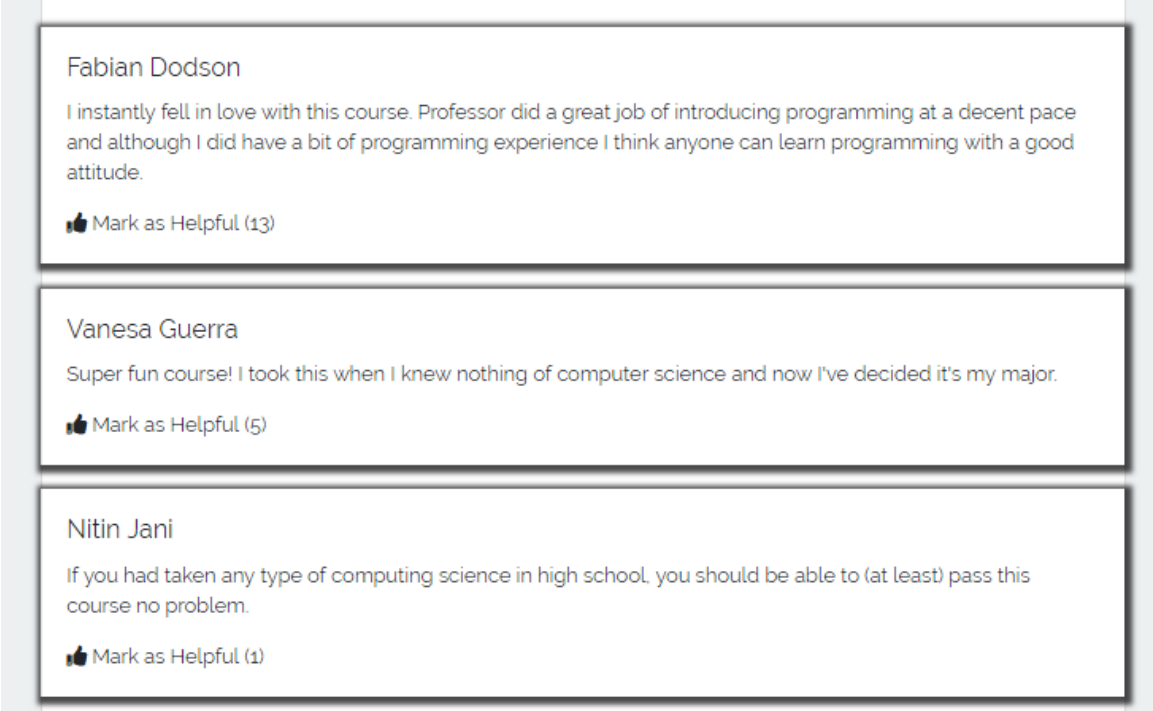

Figure 1 Posts on Discussion Forum [1]

The web application will follow the "7 ± 2 Principle" for the navigation bar of the web application. As shown in figure 2, it will have only five option in the navigation bar, which users can remember easily. If the user is logged in, the navigation bar will not show buttons for Login and Register; instead, it will show the Search bar, where the student can search for the courses. The logo of the web application will redirect the user to the homepage, which increases the usability of the app.

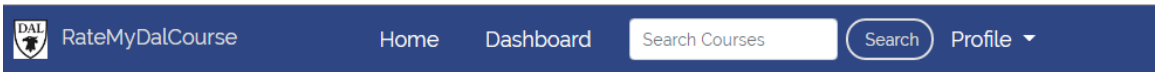

Figure 2 Navigation Bar for logged-in users [1]

Figure 3 Navigation Bar for not logged-in users [1]

3.2.1 Information Architecture

For our web application to be successful in the market, it is important to have good information architecture. It is the foundation of valuable user experience. Hence, the design solutions of the information architecture and user-centered design approach were primary aspect of our application.

A sitemap is the first step of the information architecture phase, and it shows the relationship between the different pages in the web application. Wireframes depict the initial design of each web page, and it helps us to define the content hierarchy [5].

3.2.1.1 Sitemap

The sitemap for our application is shown in figure 4. The sitemap depicts the navigation flow of the web application. Initially, students need to register and create their personalized account by entering course information. Then the course page shows the list, of course, they have completed and the course they wish to take in next term. Every student can then rate a course and create posts in the discussion forum. Further, a user can edit their profile by makes changes in the user profile section. Here they can add a course, delete course and update their account details [1].

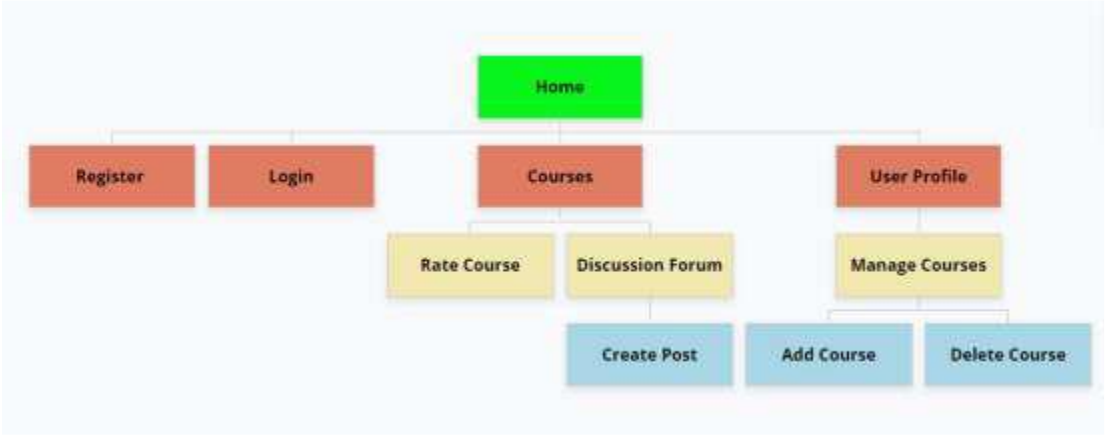

Figure 4 RateMyDalCourse application sitemap [1]

3.2.1.2 Task Flow Diagrams

Figure 5 below shows the high-level workflow of our application.

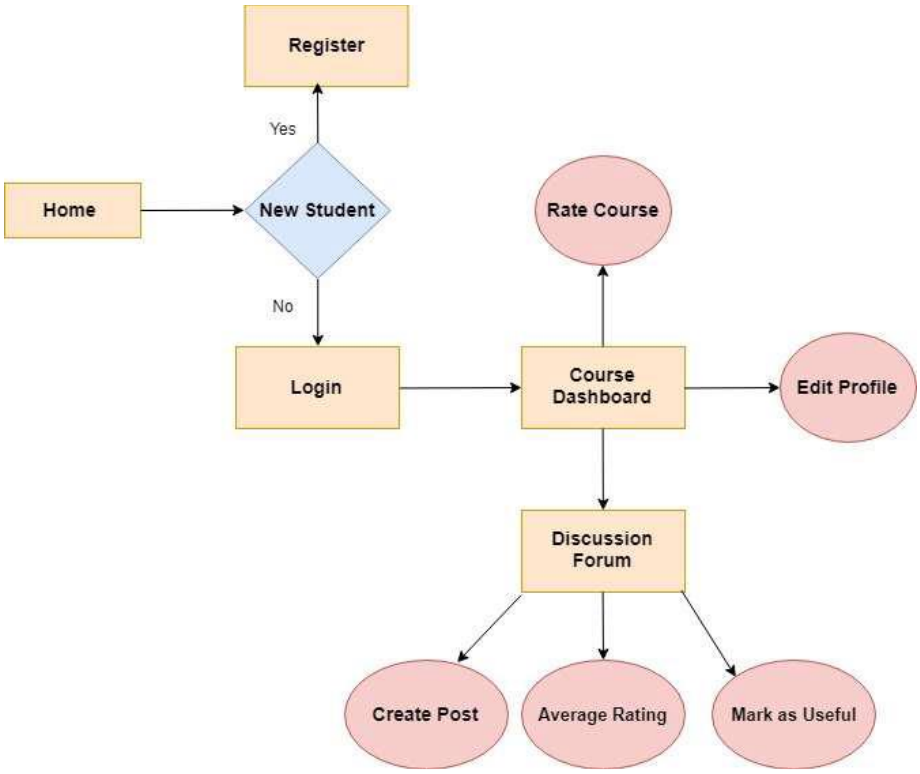

Figure 5 High-level workflow [1]

A student need to register by entering their personal details with course information. After then, the user can log in using their account details and go to the course page.

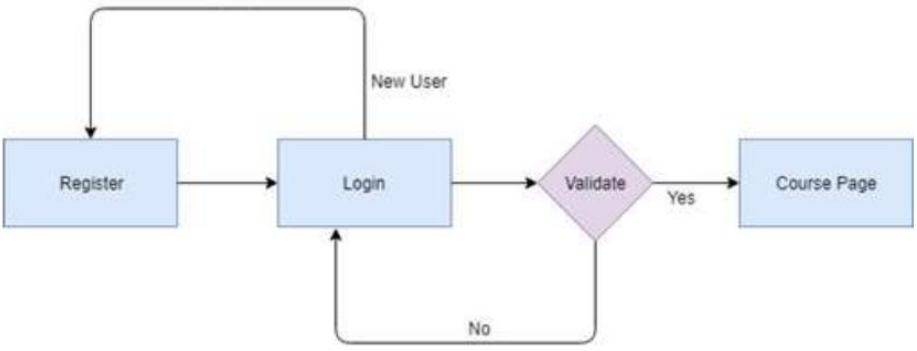

Figure 6 Register Activity [9]

Existing students can log in into their account with the help of email id and password. If their credentials are correct, they will we directed to courses dashbord page.

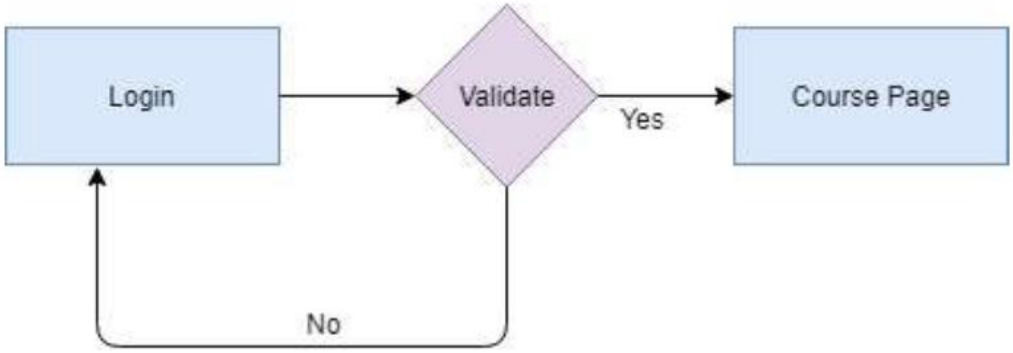

Figure 7 Login Activity [9]

When users are on the home page, they can rate and give a review of the courses they have completed in their previous terms (As per figure 8).

Figure 8 Rate Course activity [9]

On-course page, a user has the flexibility to search any course by entering a keyword in the search bar (As per figure 9).

Figure 9 Search Course [9]

If students want to ask questions regarding the upcoming course, then they go the discussion forum and create a new post (As per figure 10).

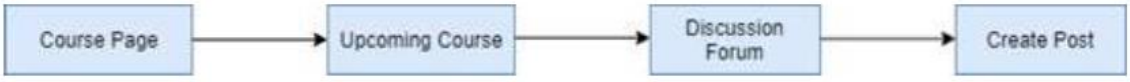

Figure 10 Create post activity [9]

If the students want to change the courses which they have registered, they can do it from the profile management page (As per figure 11).

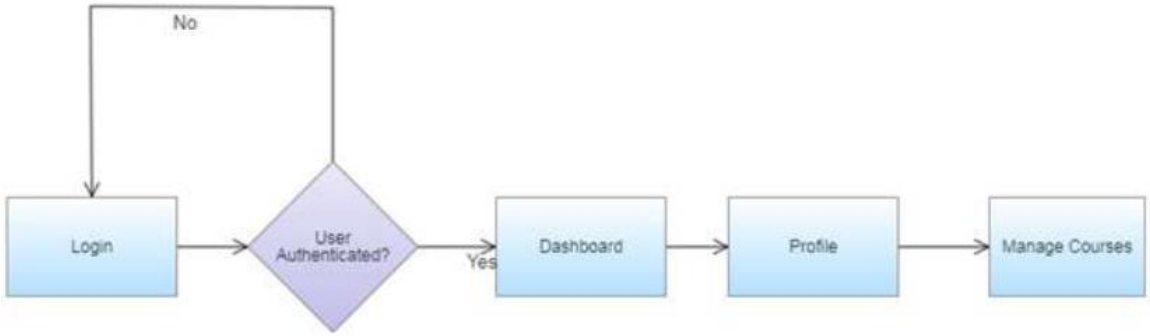

Figure 11 Mange profile [9]

If the user finds any specific post as helpful, the user can user "Mark as Helpful" option to mark the post as helpful. The posts on the discussion forum will be sorted based on "Mark as Helpful" count (As per figure 12).

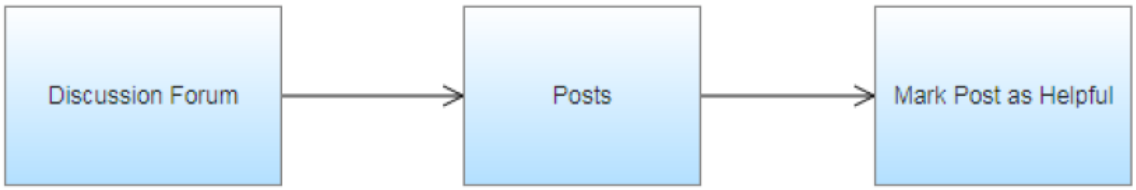

Figure 12 Mark as helpful [9]

3.2.2 Design and Layout

3.2.2.1 Homepage

The home page is the generalized page, which will indicate the purpose and goals of the web application to the end users. The content displayed on the home page specifies the tasks that can be accomplished on the web portal.

Figure 13 Home Page [10]

3.2.2.2 Registration

As shown in figure 14, students need to register if they are using the web application for the first time. Here, they will create their account by entering personal details and course information.

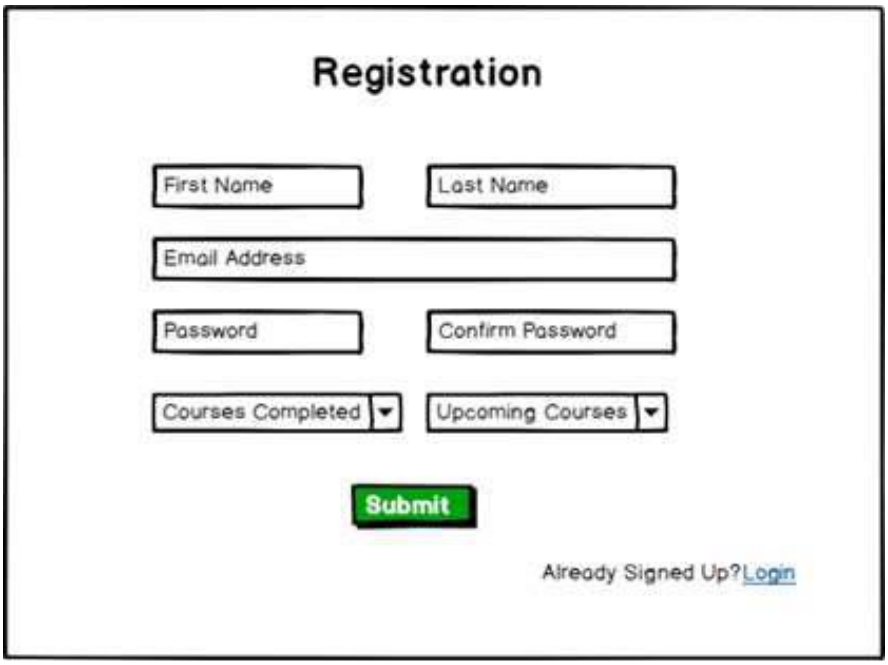

Figure 14 Registration Page [10]

3.2.2.3 Login

As per figure 15, students can log in in their account, and they can rate a course or participate in the discussion forum. Apart from this, they can change or reset their password using the “Forgot Password” option.

Figure 15 Login page [10]

3.2.2.4 Edit Profile

As shown in figure 16, students can change their account details in the edit profile section.

Figure 16 Edit Profile page [10]

3.2.2.5 Discussion Forum

One of the important parts of our website is the discussion forum, which is shown in figure 17. Here, students can get feedback from their peers regarding the course. They can solve their queries by creating new posts. Furthermore, there is a feature to like any post if they find content to be useful.

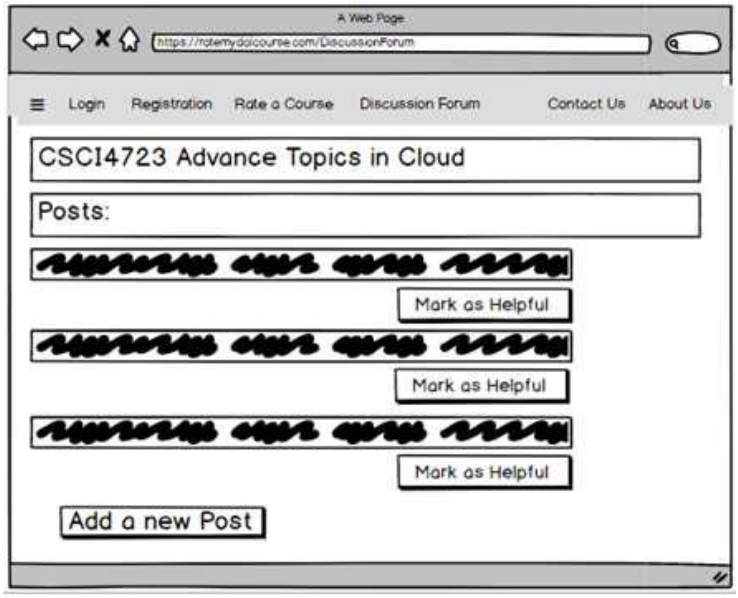

Figure 17 Discussion Forum page [10]

3.2.2.6 Dashboard

Dashboard shows the courses completed by the user and all the other courses available in the system. Users can provide rating for the completed courses. Also, users can go to discussion forum page for any subjects' discussion they might be interested in. Users can also search the courses on their dashboard using the search bar provided in navigation bar. Figure 18 shows the wireframe for dashboard.

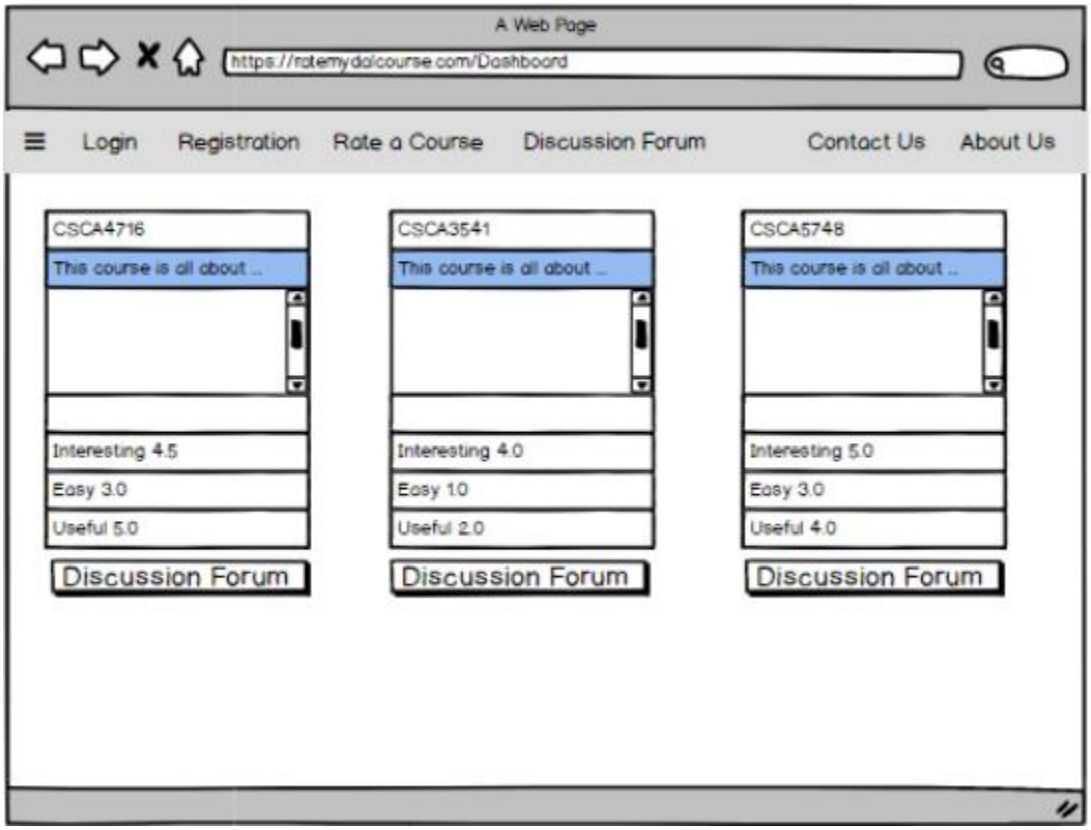

Figure 18 Student Dashboard page [10]

4. APPLICATION WORKFLOW

RateMyDalCourse web application will be developed using JavaScript software stack – MEANstack. MEAN stack stands for MongoDB, ExpressJS, AngularJS, and NodeJS. The primary benefit of MEAN stack is that it allows using JavaScript as both front-end and back-end technology. Moreover, it uses MongoDB database. As it is a NoSQL database, data can be altered in any format required by the web application without changing the configuration. Apart from this, it has a significant advantage that it has a lot of community support [11].

As shown in figure 19, there are mainly two sides: client-side and server-side. Client-side is what the user will see as their webpages on the browser. Server-side is the place where we run the business logic of the application. On the server, we will use a database to store our data. For client-side, AngularJS is used to create user interfaces and for the presentation. On the serverside, NodeJS and ExpressJS are used to running the business logic, and MongoDB is used as a database to store the data [4].

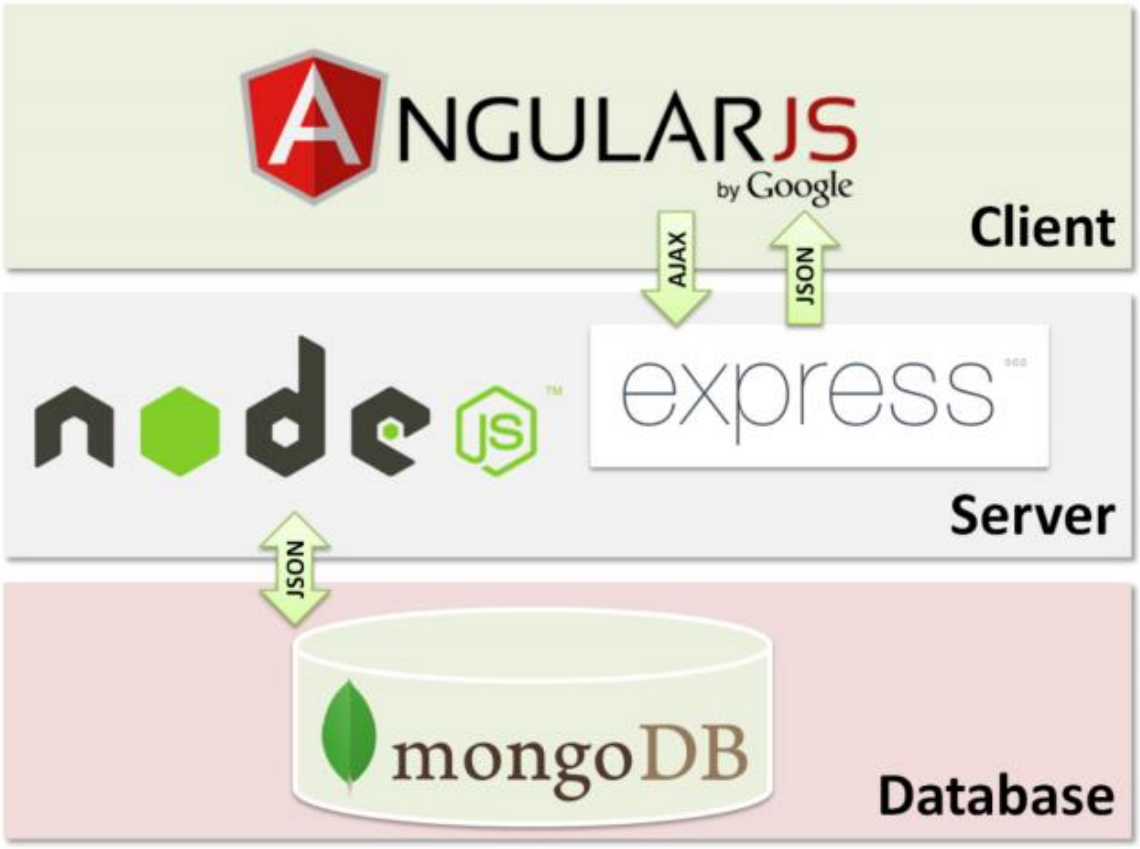

Figure 19 MEAN Stack [11]

4.1 Interaction Design

Interaction design is the process followed by designers to focus on how to improve the user interaction with the web application in a thoughtful way. To provide desired user experience, interaction design should use technologies and fundamental concepts of excellent communication. For developing RateMyDalCourse web application, we will be using HTML, CSS, Bootstrap and AngularJS for building frontend UI. For the backend business logic, NodeJS and ExpressJS will be used [12].

Students will be able to sign-in on the web application with very minimal detail and take advantage of the web application with few clicks. The purpose of Login is to authenticate the student, which will prohibit random creation of posts by the user. When the user registers with the web application, he will be asked to fill a form which will contain information regarding courses completed and courses aimed in upcoming terms by the student [1].

The student can rate a course based on their previous experience with the course. The student may also go to the discussion forum for the subjects aimed in the next term. The discussion forum will have a feature of comment, which will help students to directly ask a specific question to other students. Moreover, the post will have the "Mark as useful" button to mark it as a useful comment to get a better idea about the comments' usefulness. The web application will maximize the use of Nielsen's Components of Usability to make the User Experience smooth and consistent [1].

AngularJS applications work as a Single Page Applications, which means the content is displayed dynamically on the web page based on defined business logic. The app will start from the AppModule, which is mentioned in the main.ts file as a bootstrap module. AngularJS works with components instead of following the MVC architecture. As shown in figure 20, app.moudle.ts will contain all the components which are used in the web application, and we have to mention which component will start the application [13].

```
@NgModule({
  declarations: [
    AppComponent,
    HomepageComponent,
    NavbarComponent,
    FooterComponent,
    LoginComponent,
    RegistrationComponent,
    DashboardComponent,
    EditprofileComponent,
    DiscussionforumComponent,
    ConfirmValidatorDirective,
    PagenotfoundComponent,
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    FormsModule
  ],
  providers: [
    {
      provide:UrlSerializer,
      useClass:LowerCaseUrlSerializer
    },  AuthenticationService,
  ],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Figure 20 App.module.ts file [1]

As shown in figure 20, all the components used by the web application are specified inside the "declarations" section. The file also imports some of the modules. For example, it will import "AppRoutingModule," which is an essential part of AngularJS. We can specify different routes, and the content will be shown/hidden based on the router link provided. The bootstrap component is mentioned as AppComponent, which will kick start the application.

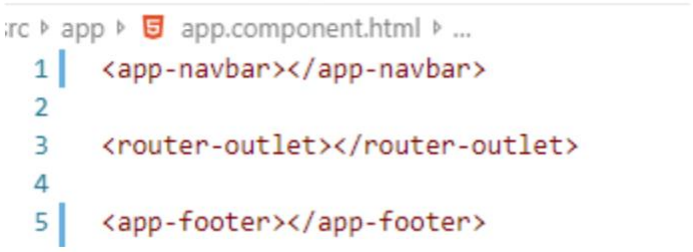

Figure 21 App.component.html file [4]

Now, app.component.html will have only the selectors. As shown in figure- 21, our app.component.html file has only 3 selectors: app-navbar, router-outlet, and app-footer. "appnavbar" selector will show the navigation bar of the webpage. "app-footer" selector will display the content of footer on each webpage. "router-outlet" will be responsible for presenting the different content based on the route selected by the user. This is how the AngularJS works as Single Page Applications.

4.1.1 Click Streams

Click streams are the sequence of clicks followed by the user to reach to the desired webpage. When the user visits the homepage of our web application, Login and Registration options will be displayed in the navigation bar. If the user clicks on the Login button, he will be redirected to the Login page, where the user can provide credentials and view his dashboard. If the user wants to post any question, they can go to the discussion forum page to that specific subject and post the question there. Other users will answer this question.

Click streams for both Login feature, is shown in figure 22

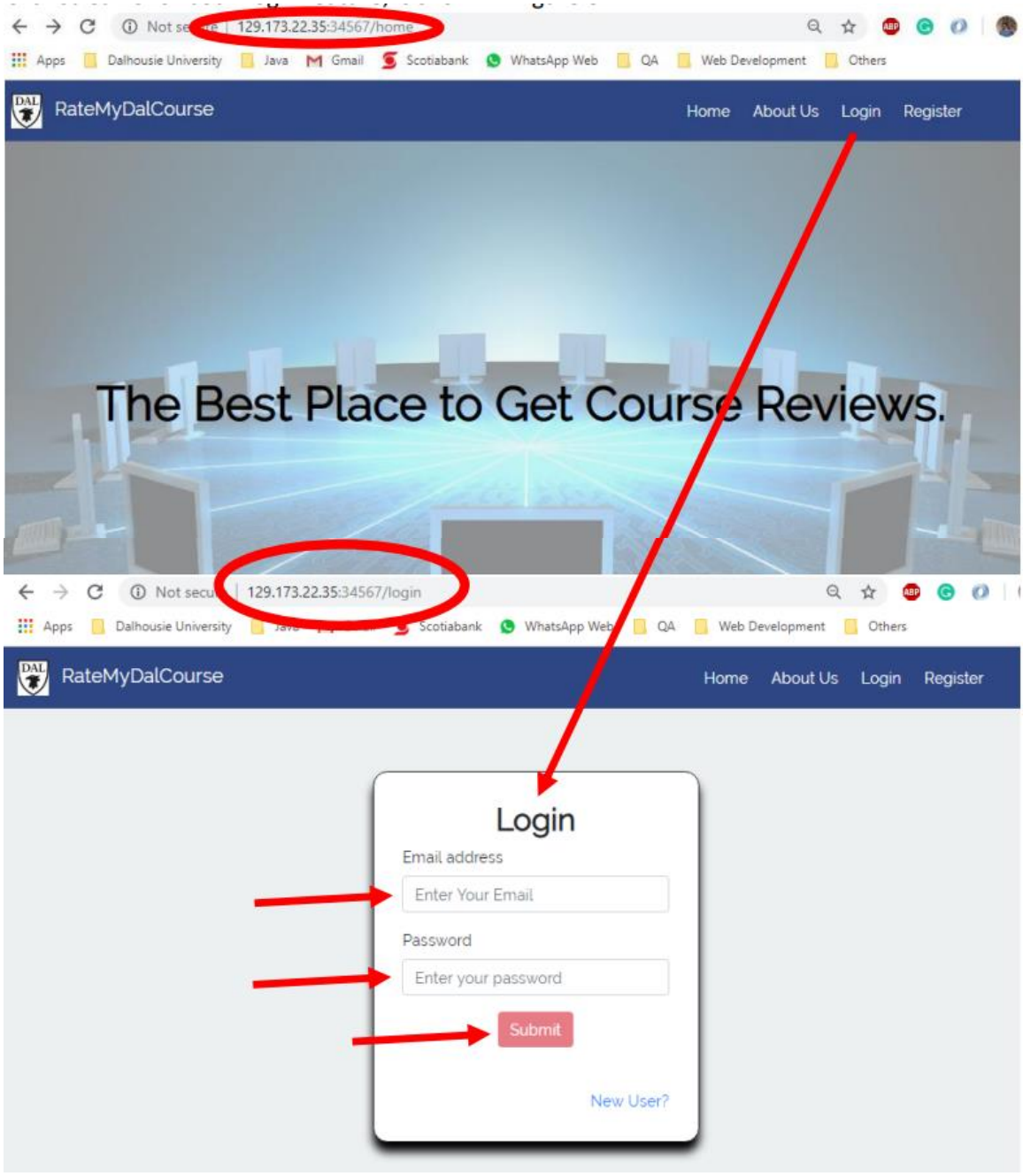

Figure 22 Login clickstream [4]

Clickstream for edit profile feature is shown in figure 23 below:

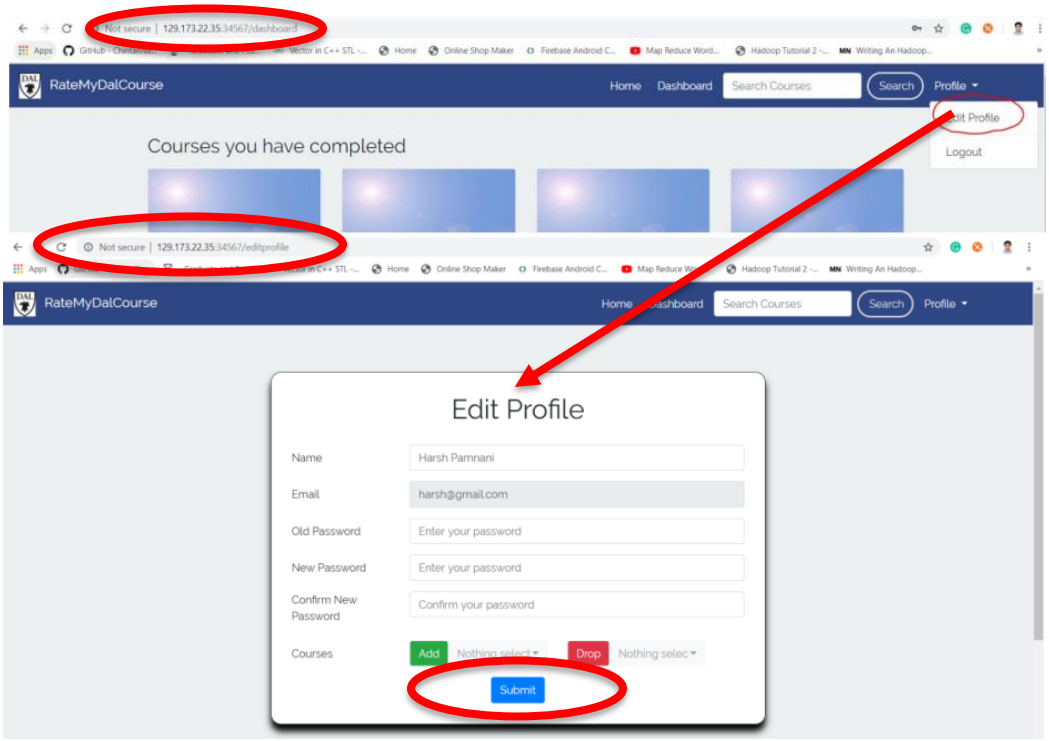

Figure 23 Edit profile clickstream [3]

Clickstream for "Mark as helpful" feature is shown below:

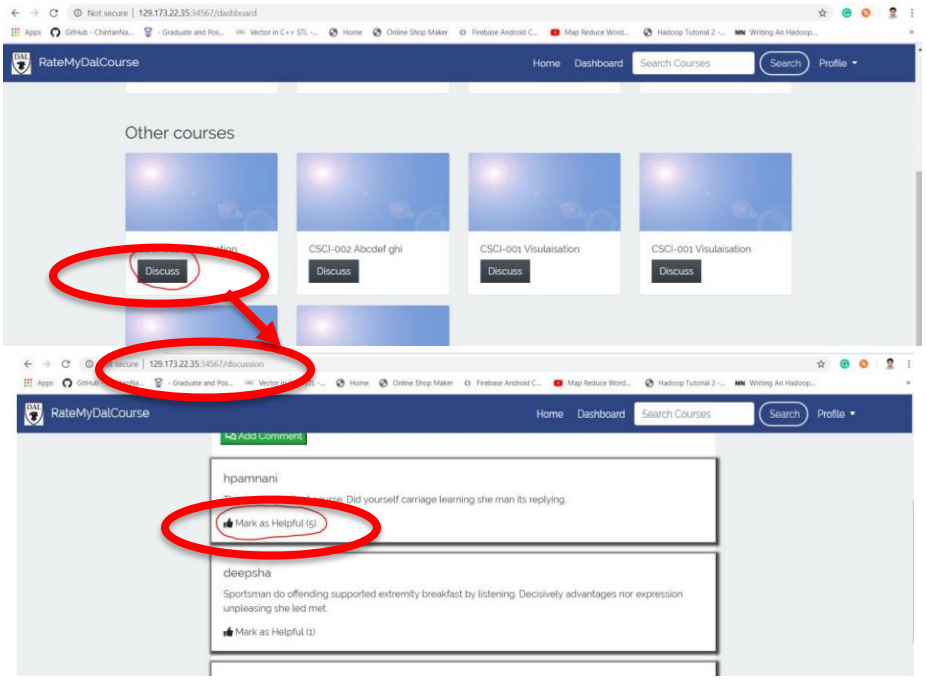

Figure 24 Mark as helpful clickstream [3]

Clickstream for registration feature is shown in figure 25 below:

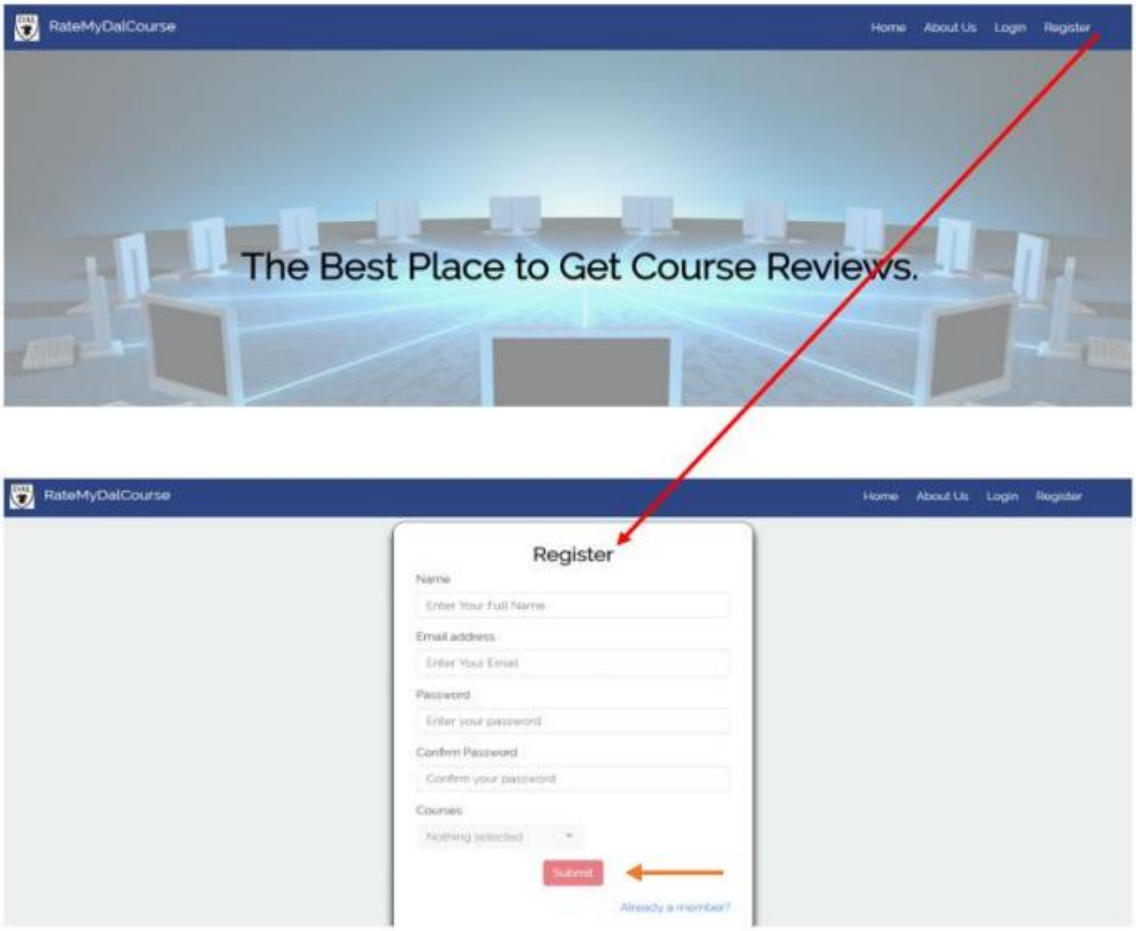

Figure 25 Registration clickstream [2]

Clickstream for rating a course is shown in figure 26 below:

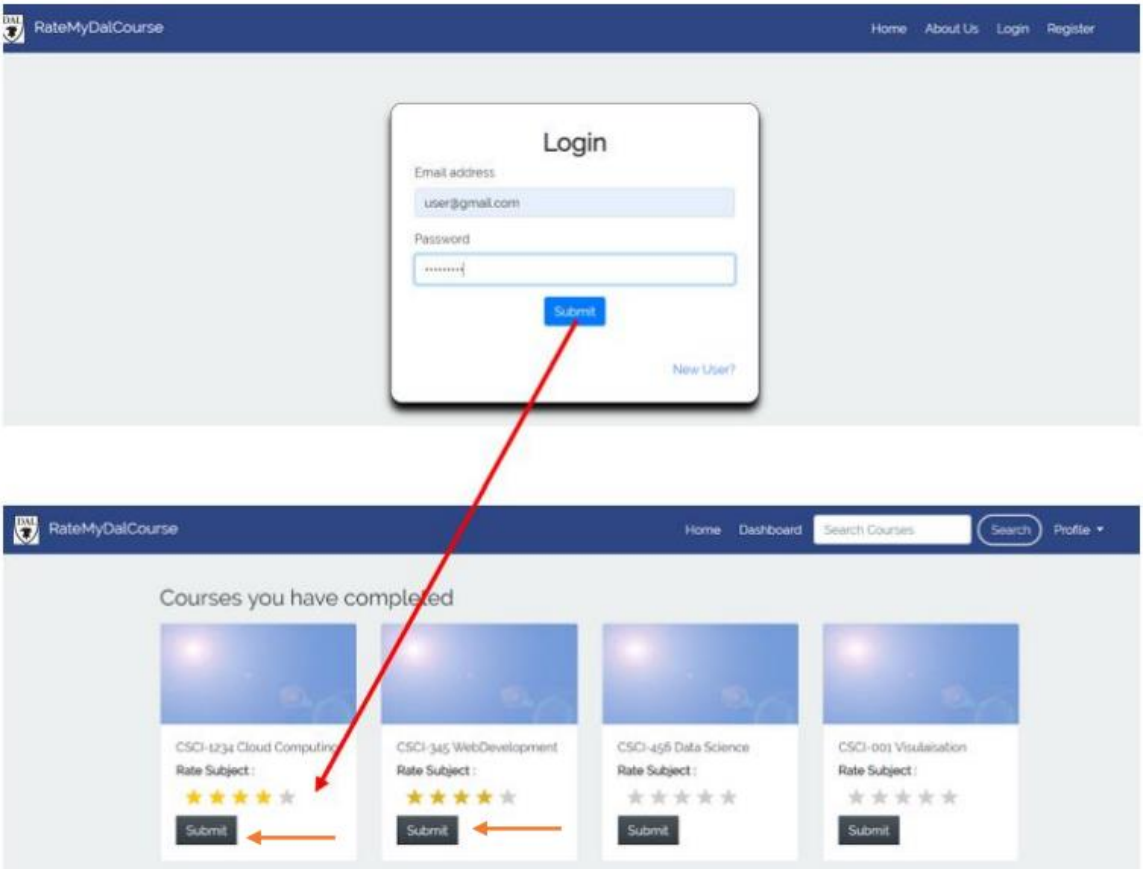

Figure 26 Course rating clickstream [2]

On successful login, the user will be redirected to the dashboard page, where all the subjects will be displayed. If the user wants to create a post, they can go specific course's discussion forum page and post their question. Click stream for "Create Post" feature is shown in figure 27.

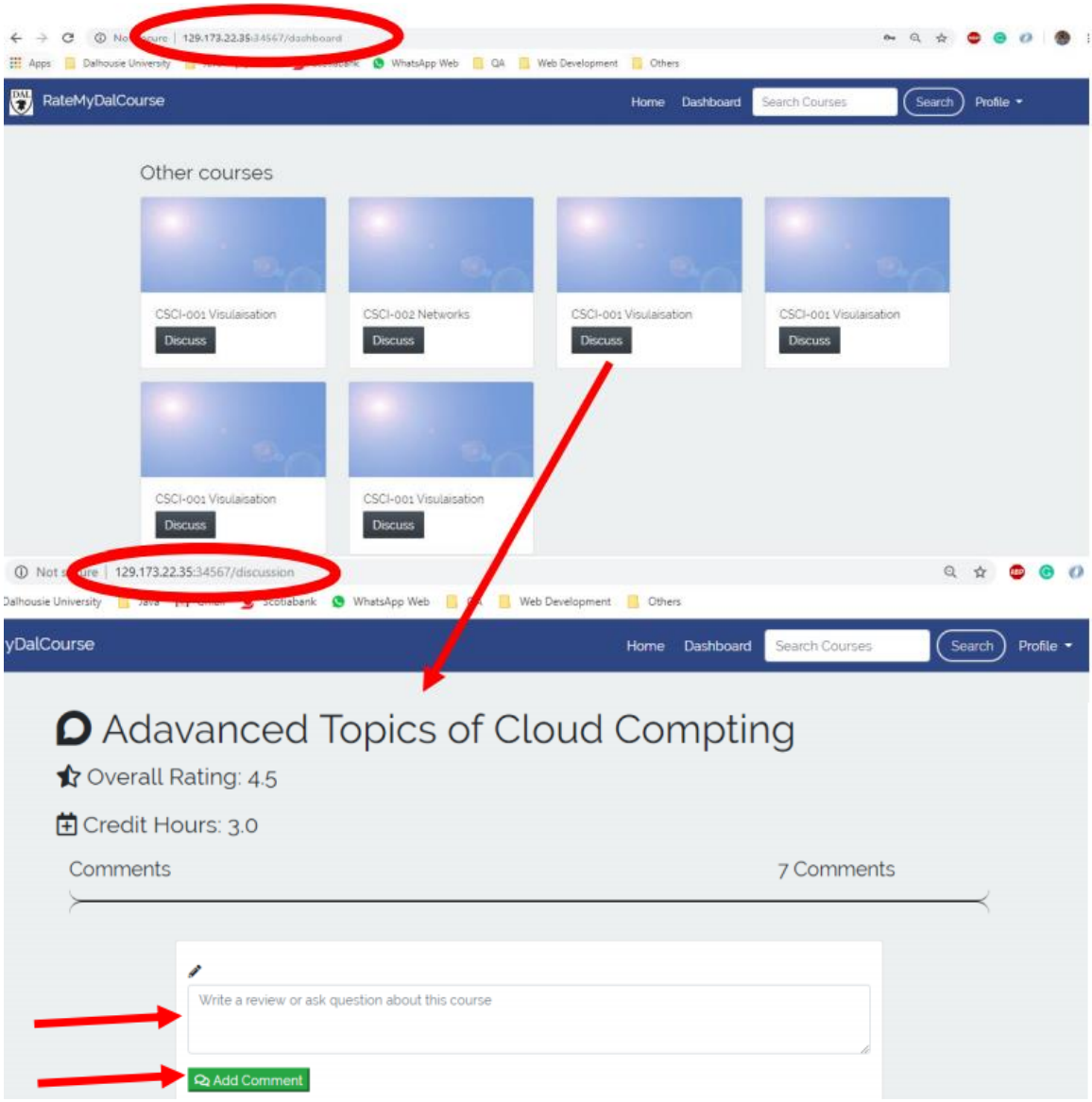

Figure 27 Create post clickstream [4]

Clickstream for “Search Course” feature is shown in figure 28 below:

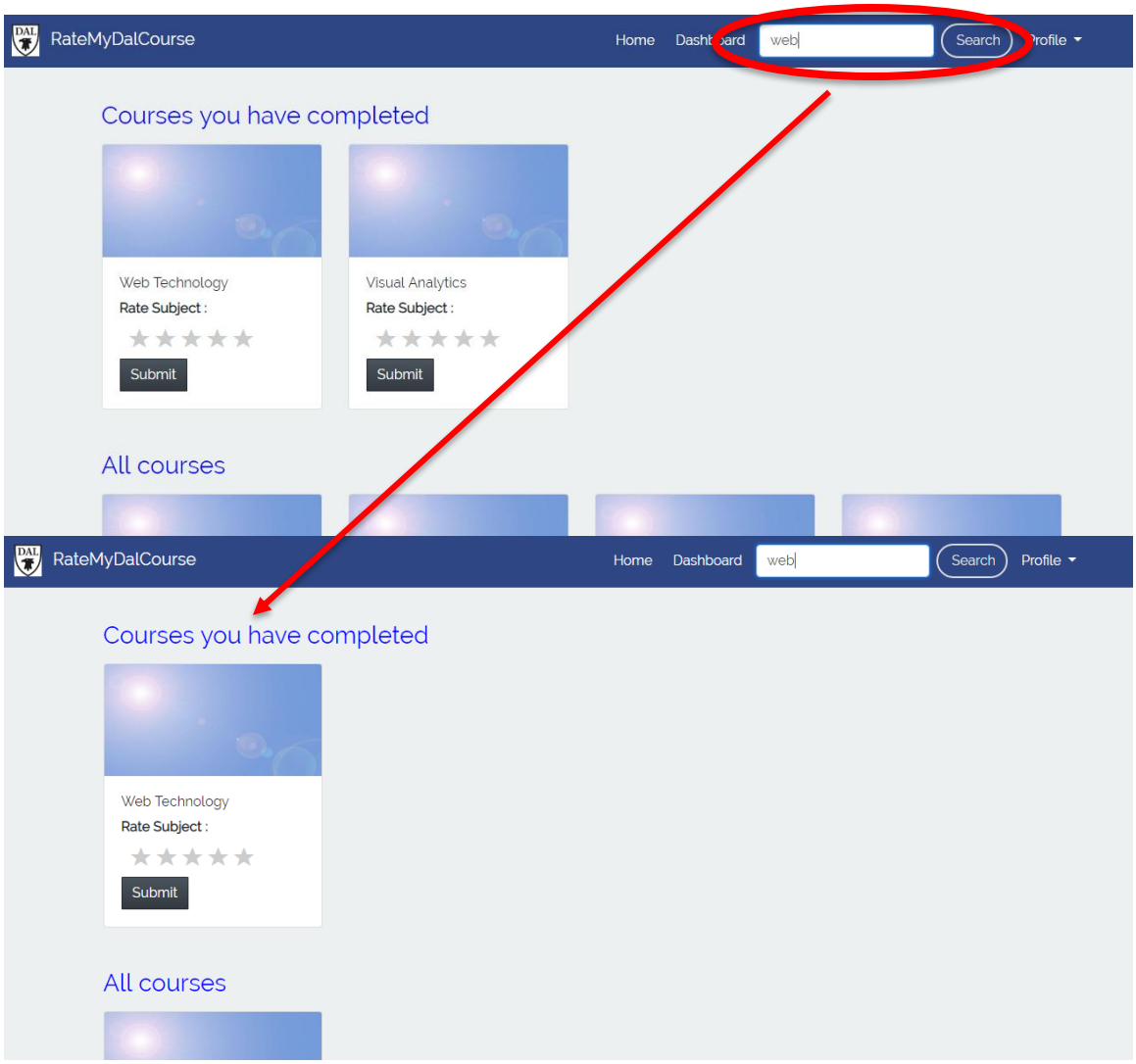

Figure 28 Search course clickstream [5]

4.1.2 User Personas

Use Case 1:
Name: Jay Desai
Age: 24
Profession: Student
University: Dalhousie University

Scenario: Jay wants to learn Advanced topics in Network Security in his Fall term. Before selecting the subject, he wants to know overall ratings of the course as well as professor and also wants to know different technologies and tools which will use to him in this subject so that he can learn before this course [1].

Basic Flow:
1. Jay can select this subject from the list of the courses on the dashboard by course name or course code in the search bar.
2. By clicking on the subject, he views the overall ratings of professor and course, which is 4.3. He explores various post related to this subject, which provides detailed information about the course.
3. Jay upvotes the post which answers his doubts and thanks to the student who has posted that content [1].

Alternate Flow:
1. Jay does not find any post which answers his doubt.
2. He posts his doubt by creating a new post.
3. John, who has already completed this subject, answers Jay's question by commenting on that post [1].

Case 2:
Name: Rob Delano
Age: 28
Profession: Student
University: Dalhousie University

Scenario: Rob is a Computer Science student at Dalhousie University. He completed Software Engineering course in Winter term and wants to give his feedback on the subject so that it will be helpful to the future students. Use the "Rate My Course" website to provide feedback on any Computer Science subject [1].

Basic Flow:
1. Rob logs in and finds the course by and searching course code in the navigation bar.
2. He provides a rating of the Software Engineering by selecting the value from one to five.
3. He creates one post and shares his overall learning experience of the subject. He also shares that student who has basic knowledge of java can easily score A+ grade in the subject.

4. He also solves queries of other students by commenting below their posts and guide them [1].

Alternate Flow:
1. Rob is unable to login into the website.
2. Invalid details are provided.
3. Rob is inadequate to rate the course due to the poor internet connection [1].

4.2 Process and Service Workflow

NodeJS is a JavaScript run-time environment which executes JS codes on the server side. We can implement our business logic using NodeJS on the backend server. It helps to build reliable, scalable and fast server applications. NPM is the package manager for NodeJS. Using which we can install many public and private packages from the npm registry [14]. For the database, MongoDB is used. MongoDB has a JSON like structure for storing the documents. So, whenever the application needs to do any data operation, it will use CRUD (Create, Read, Update, and Delete) operation with the database. NodeJS works as a server and browser (from which we make requests) will work as a client. A high-level overview of web application architecture is shown in figure 29.

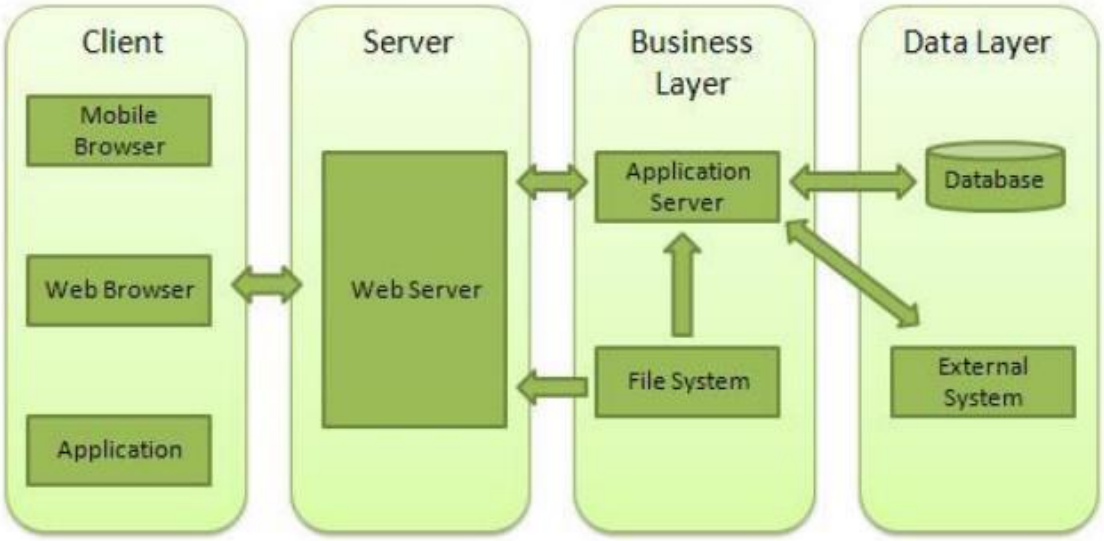

Figure 29 Web application architecture [14]

Workflow for Login feature

For the login feature, the user has to enter email id and password. There will be validations for the email type to verify the email structure. Also, the password can't be less than eight characters (because that is the validation done at the registration time). These both validations will be done in AngularJS. Once the email and password have valid format string, a request will be sent by the Client (browser) to the Server. There will be business rules for email and password validation at the business layer. User's email and password will be stored in the MongoDB database. So, NodeJS server will make a request to the MongoDB database to validate the user credentials. If the credentials are valid, a successful response will be sent by the Server to the Client. However, if the credentials are not valid, then form error will be sent to the Client, and appropriate error messages will be displayed using AngularJS based on certain conditions [4].

Workflow for Create Post feature

Once the user is logged-in, he will be redirected to the dashboard, where a list of courses will be displayed. If the user has already completed the course, he will be able to rate the course out of 5. If the user has not completed the course and would like to ask some question, then he can select that particular subject and go to its discussion forum. On the discussion forum page, there will be multiple comments. If the user has any question, he can create a new post on the discussion forum. Once the user clicks "Submit Post" button, the post content will be validation by the server. If the post is empty, it will be rejected. If the user has mentioned any abusive content inside the post, this comment will not be posted on the discussion forum. If the post is successfully validated by the server, then the comment will be added to the MongoDB database. After that, success response will be sent back to the client, and the post will be added the discussion forum page [4].

Workflow for Registration feature

For registration to our web application, user has to enter following information: Name, Email-Id, Password, and Courses Completed. Once the user has entered all the details, POST request would be sent to server. The server will validate all the information. If there is any error, it would be sent back to the client and it will be displayed on the webpage. If there is no error and user is successfully registered, then the registration form will be cleared and user will be informed that the registration has been successful.

Workflow for Dashboard

If the user is not logged in and tries to access the dashboard then, he will be redirected to dashboard. If the user is logged-in, the courses will be fetched from the server for dashboard. There are two types of courses: courses completed by the user and all the other courses.

Workflow for rating a course

The user will be able to rate the courses completed by him. If the user has already rated the subject, then stars will be present for that course on dashboard. If the user wishes to rate the subject again, then the rating will be updated. If the user has not already rated the subject, then rating will be added for that course in database.

Workflow for edit profile

If the user is already logged-in, they will be able to edit their profile. For editing the profile, user has to enter old password for security reasons. If the user wants to change his password, he will have to enter the old password, new password and confirm the new password. From the edit profile page, users can also add new courses or drop the existing courses. These courses will updated for that student in database, and when the user logs in to view his dashboard, the courses will be updated.

Folder Structure

Figure 30 to figure 32 shows the expected files and folder structure for our web application. The root directory shows the list of folders and file. It will contain package.json file, which will contain all the dependencies required for RateMyDalCourse web application. The "src" folder will contain all the component of the application and app module to kick start the web application. Inside each component there will be .html, .css, .specs.ts, and .ts file for the UI and business logic [1].

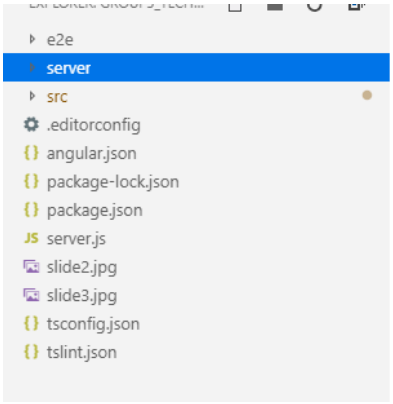

Figure 30 Root folder of the application [1]

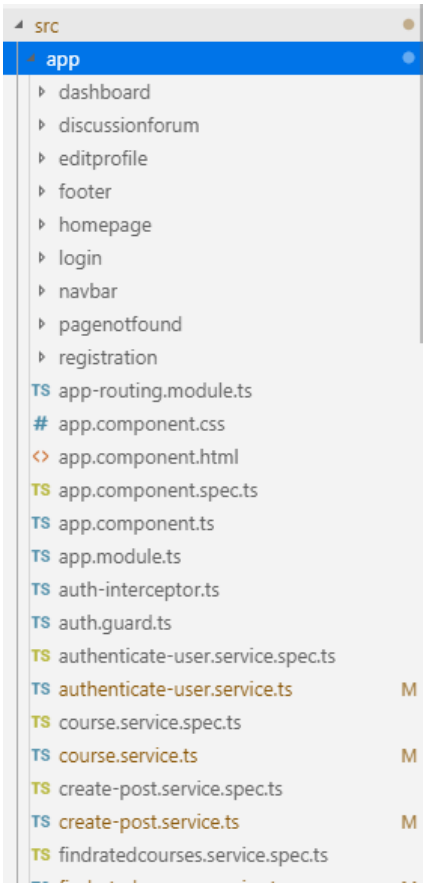

Figure 31 Src folder structure [1]

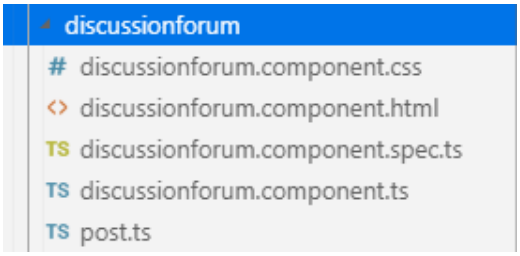

Figure 32 Individual component files [1]

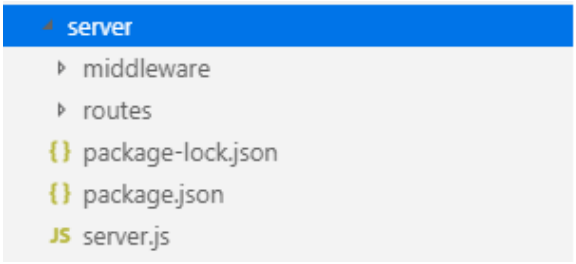

Figure 33 Server folder structure [1]

5. CONCLUSION

RateMyDalCourse will provide a platform where students can get the reviews and ratings for courses offered during each term based on other students' feedback. Through this platform, students will be able to know the subjects' evaluation before actually signing it up for next term, which might help students to decide whether they want to continue with the courses or drop it. The students will get an idea about the numerous course information like the number of assignments, difficulty level for the tasks, group project details, whether the attendance is mandatory or not, etc.

To conclude, developing RateMyDalCourse web application would provide a platform that helps to connect student of Dalhousie University.

6. RECOMMENDATIONS

Following recommendations should be considered to improve the usability of RateMyDalCourse web application.

- There should be one admin for the administration of the comments on the discussion forum. Therefore, if there is any offensive comment regarding the course or professor, it can be deleted by the admin panel.
- To make the web application more secure, the user's registered email should be verified before giving him access to the dashboard.
- "Forgot Password" feature should be provided to the users. This will be helpful to the users, in case they forget their password.
- There should be pagination feature for the comments displayed on the discussion form. This feature will be helpful in future, if the number of comments are increasing for any particular subject. Therefore, we can only load top 20 comments when the page is accessed and rest comments should be provided using pagination feature.
- We are storing the password using Caesar Cipher with key = 13. However, Caesar Cipher can be broken easily. Hence, password should be stored in hash format, which is a one-way process. In this way, we can secure user's passwords.
- There should be a "Contact Us" page, in which the users can contact the developers and admin panel to add missing courses from the curriculum in the list.
- We have already provided the "Mark as Helpful" option for the posts to the users. However, there is not option to report the comment. Hence, an option to report the comment should also be given to users.

7. REFERENCES

[1] H. Pamnani, C. Patel, D. Shah and S. Amin, "Project Proposal Group-3", CSCI5709 – Advanced Topics in Web Development, 2019. [Accessed: 24 - Jul - 2019].

[2] S. Amin, "Assignment-3", CSCI5709 – Advanced Topics in Web Development, 2019. [Accessed: 24 - Jul - 2019].

[3] C. Patel, "Assignment-3", CSCI5709 – Advanced Topics in Web Development, 2019. [Accessed: 24 - Jul - 2019].

[4] H. Pamnani, "Assignment-3", CSCI5709 – Advanced Topics in Web Development, 2019. [Accessed: 24 - Jul - 2019].

[5] D. Shah, "Assignment-3", CSCI5709 – Advanced Topics in Web Development, 2019. [Accessed: 24 - Jul - 2019].

[6] "RateMyProfessors.com – Find and rate your professor or campus.", Ratemyprofessors.com, 2019. [Online]. Available: https://www.ratemyprofessors.com/. [Accessed: 24 - Jul- 2019].

[7] "Rate My Teachers", Ratemyteachers.com, 2019. [Online]. Available: https://ratemyteachers.com/. [Accessed: 24 - Jul - 2019].

[8] "What is User Centered Design?", The Interaction Design Foundation, 2019. [Online]. Available: https://www.interaction-design.org/literature/topics/user-centered-design. [Accessed: 25 - Jul- 2019].

[9] "Flowchart Maker & Online Diagram Software", Draw.io, 2019. [Online]. Available: https://www.draw.io/ [Accessed: 25 - Jul- 2019].

[10] "Balsamiq Wireframes | Balsamiq", Balsamiq.com, 2019. [Online]. Available: https://balsamiq.com/wireframes/ [Accessed: 25 - Jul - 2019].

[11] "Mean Stack Development [For Developers]", Hacker Noon, 2019. [Online]. Available: https://hackernoon.com/mean-stack-development-for-developers-4d88c40c4103. [Accessed: 25 - Jul - 2019].

[12] C. Cousins, "What is an Interaction Designer? - Designmodo", Designmodo, 2019. [Online]. Available: https://designmodo.com/interaction-design/. [Accessed: 25 - Jul - 2019].

[13] "Single Page Apps with AngularJS Routing and Templating", Scotch, 2019. [Online]. Available: https://scotch.io/tutorials/single-page-apps-with-angularjs-routing-and-templating. [Accessed: 25 - Jul - 2019].

[14] "Node.js Web Module", www.tutorialspoint.com, 2019. [Online]. Available: https://www.tutorialspoint.com/nodejs/nodejs_web_module.htm. [Accessed: 25 - Jul - 2019].