Deliverable-3

Project Title: eMed-a complete pharmacy cart at your doorstep.

Group Name: Team Delta

Group Members:

S. No	Full Name	Student ID	Email
1.	Shabana Syed	11723237	ShabanSyed2@my.unt.edu
2.	Ganesh Gundekarla	11700551	ganeshgundekarla@my.unt.edu
3.	Rishika Kandrigal	11699010	rishikakandrigal@my.unt.edu
4.	Md Ariful Hasan	11428575	MdArifulHasan@my.unt.edu
5.	Vishnu Priya Vulichi	11727751	vishnupriyavulichi@my.unt.edu
6.	Vivek Nelluri	11691249	Viveknelluri@my.unt.edu
7.	Pranav Chalasani	11712496	PranavChalasani@my.unt.edu
8.	Ajay Kumar Reddy Sammeta	11646754	Ajaykumarreddysammeta@my.unt.edu

Requirements:

- **User Authentication & Registration:**

  Users can sign up by creating an account along with their personal information. With the help of secure authentication account protection is ensured which includes account verification and password reset options. Register the account and the user need to authenticate is the key step.

- **Medicine Browsing & Purchase:**

  Users can go through a detailed list of medicines and can easily add required items to a cart for easy purchase. Users can get an instant discount by applying promo codes with real-time price differences as they adjust their carts. The medicines would have details about them.

- **Appointment Scheduling:**

  Users can access the doctor's information and schedule appointments based on their requirements. From the time of booking to confirmation, the system manages flawless appointment administration. One appointment and other appointment wont overlap.

- **Blood Donation and Emergency Requests:**

  This platform provides an advanced search system to find blood donors and users can announce urgent blood requests in an emergency. Requesting a donor for blood would be easy.

- **Admin Control Panel:**

  Admins have total control over the system and every service they provide. They can modify or delete content and fulfill orders and services. They can verify and add everything.

- **User-Admin Communication:**

A live chat feature will help users to communicate with professionals and clear their queries regarding orders, or any other matters regarding the medical field. This enables to have good interaction.

- **Ambulance and Emergency Response Services:**

  For immediate assistance during an emergency, users can request ambulance services.

- **Payment And Checkout Options:**

  With the help of this platform, consumers can choose a payment platform option and manage their shipping addresses. Payment gateway should be setup for this.

- **SEO-Optimized Category Pages:**

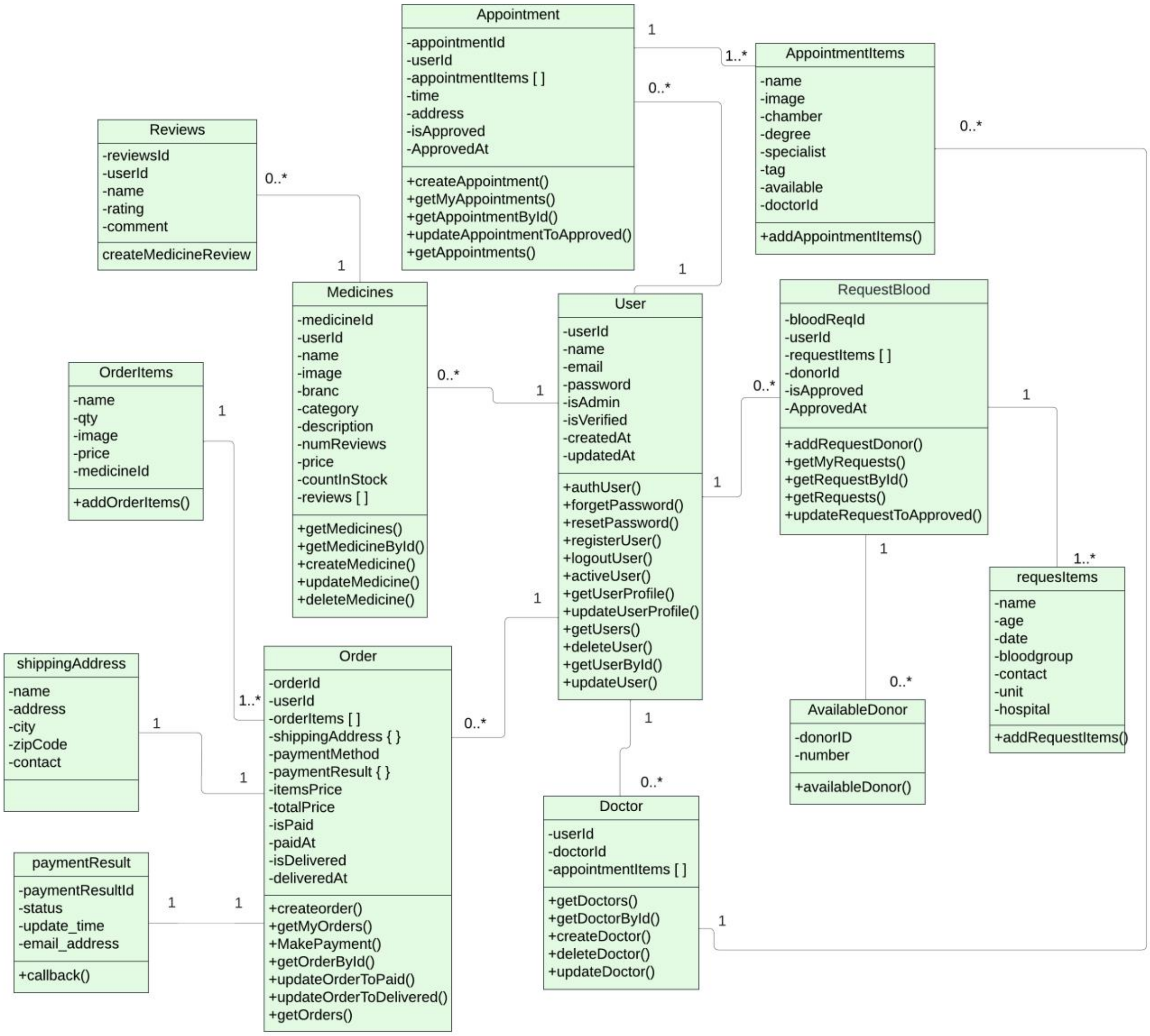
  This website is equipped with category pages that are geared toward search engines to make sure consumers can find patient health records.

- **Remote Health Monitoring:**

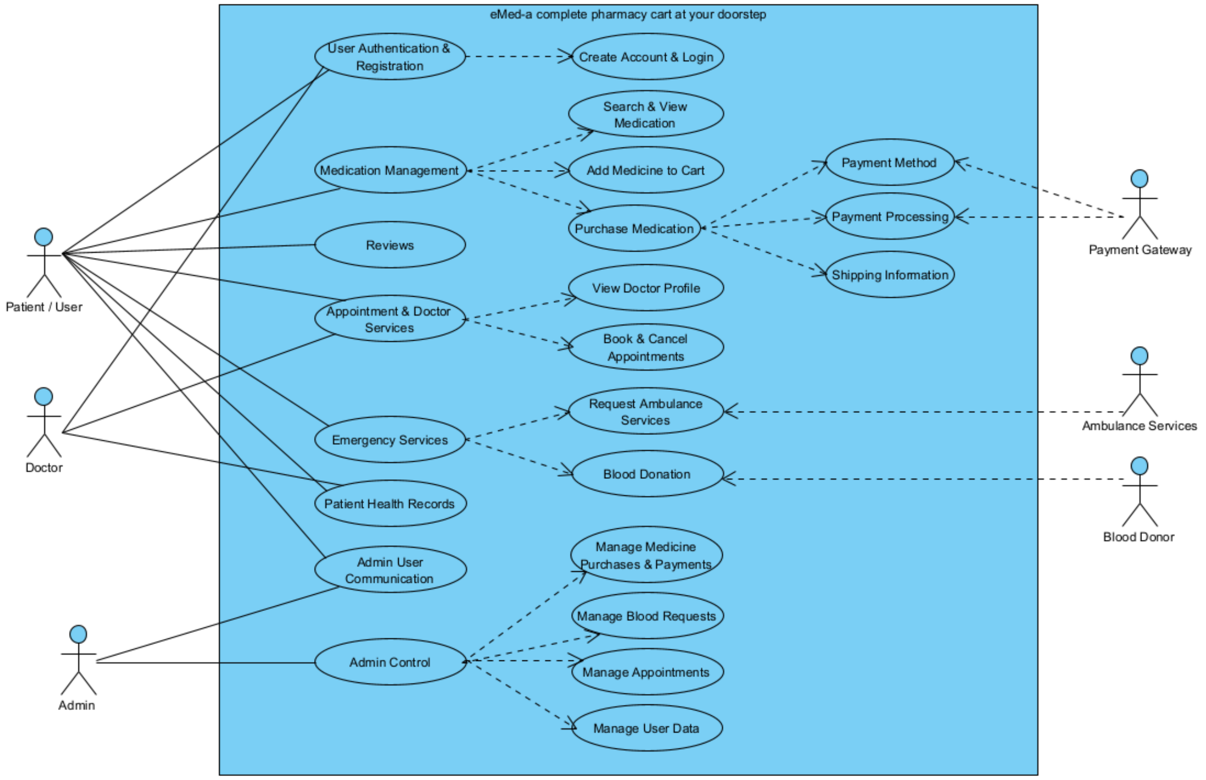
  The platform enables quick and remote health monitoring, providing real-time insight into patients' health information. Setting up a video call with the patient and doctor.

UML Diagrams:

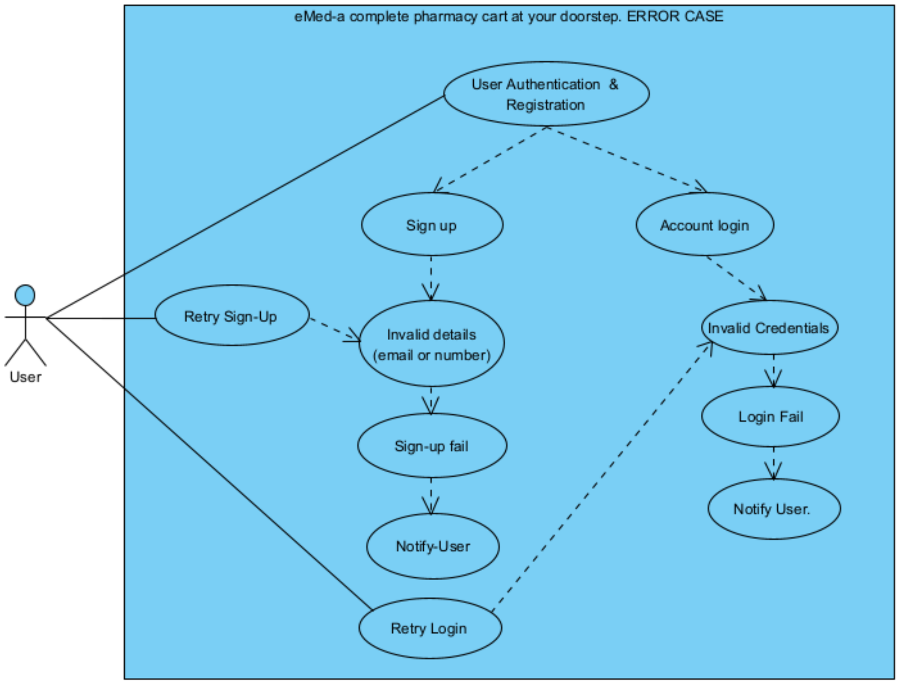
Class Diagram:



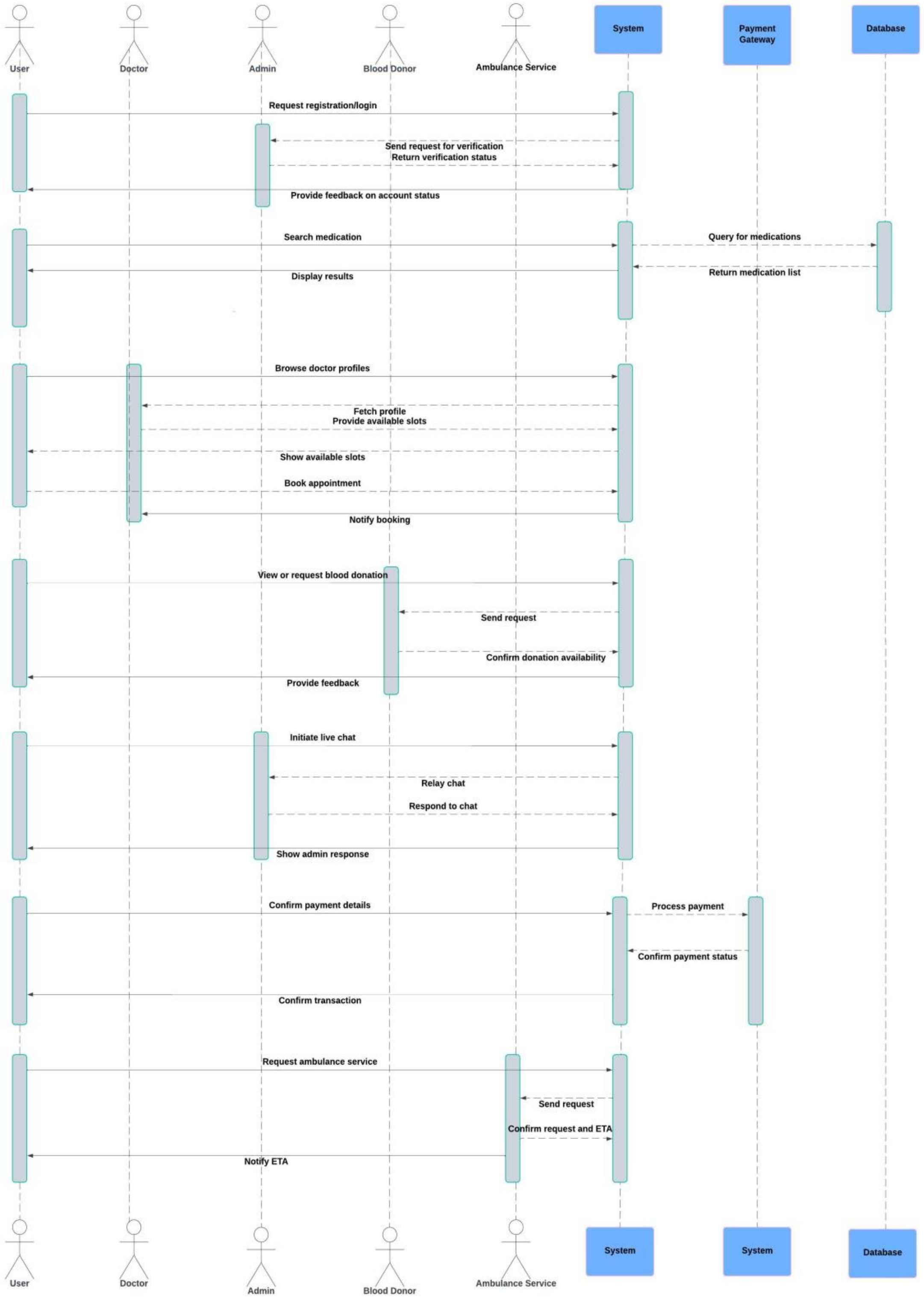
Use Case Diagram:



Use Case Error Diagram:


eMed-a complete pharmacy cart at your doorstep. ERROR CASE
User Authentication & Registration
Sign up
Account login
Retry Sign-Up
Invalid details (email or number)
Invalid Credentials
User
Sign-up fail
Login Fail
Notify-User
Notify User.
Retry Login

Sequence Diagram:


User
Doctor
Admin
Blood Donor
Ambulance Service
System
Payment Gateway
Database
Request registration/login
Send request for verification
Return verification status
Provide feedback on account status
Search medication
Query for medications
Return medication list
Display results
Browse doctor profiles
Fetch profile
Provide available slots
Show available slots
Book appointment
Notify booking
View or request blood donation
Send request
Confirm donation availability
Provide feedback
Initiate live chat
Relay chat
Respond to chat
Show admin response
Confirm payment details
Process payment
Confirm payment status
Confirm transaction
Request ambulance service
Send request
Confirm request and ETA
Notify ETA
User
Doctor
Admin
Blood Donor
Ambulance Service
System
System
Database

Test Cases:

Functionality	Input	Output	Expected outcome	Test pass/fail
User Registration	Valid user details	Successful registration	User registered successfully with confirmation message	Pass
User Registration	Duplicate email	Registration error	Error message indicating email already in use	Pass
User Login	Correct credentials	Successful login	Login successful, redirect to user dashboard	Pass
User Login	Incorrect password	Login error	Error message indicating invalid username or password	Pass
User Login	Non-existing email	Login error	Error message indicating invalid username or password	Pass
Medicine Search	Valid medicine name	List of matching medicines	Display list of medicines with details and 'Add to Cart' option	Pass
Medicine Search	Invalid medicine name	No results found	Message indicating no medicines found	Pass
Add to Cart	Select medicine and quantity	Medicine added to cart	Cart updated with selected medicine and quantity	Pass
Checkout Process	Valid shipping and payment details	Order confirmation	Order placed successfully with confirmation message and email	Pass
Doctor Appointment Booking	Select doctor and available time slot	Appointment confirmation	Appointment booked successfully with confirmation message	Pass
Doctor Appointment Booking	Select unavailable time slot	Booking error	Error message indicating time slot is not available	Pass
Doctor appointment booking without user registration	Scheduling appointment without user registration	error	Error message indication no user registered, pops up as a console error saying name, and name is undefined.	fail

Description:

- **User Registration (Valid/Non-unique email):** Verifies that a user can register successfully with valid details and receive a confirmation message, while checking that an error message is displayed when the email is already in use.

- **User Login (Correct/Incorrect credentials):** Ensures that a user with valid credentials can log in and be redirected to the dashboard, while checking that incorrect credentials trigger an error message indicating invalid username or password.
- **Medicine Search (Valid/Invalid medicine name):** Tests the search functionality by ensuring that entering a valid medicine name displays matching results, and entering an invalid name shows a "no results found" message.
- **Add to Cart (Select medicine and quantity):** Confirms that selecting a medicine and specifying a quantity updates the cart accurately with the correct items.
- **Checkout Process (Valid/Invalid details):** Verifies that the checkout process completes successfully with valid shipping and payment details, placing an order and sending a confirmation email.
- **Doctor Appointment Booking (Available/Unavailable time slot):** Ensures that selecting an available time slot successfully books the appointment, while choosing an unavailable slot result in an error message.
- **Doctor Appointment Booking** This test checks if the system properly handles an attempt to book a doctor's appointment without user registration, ensuring that an error message is displayed. But when we test it, we are getting the Error message indication that no user is registered it pops up as a console error saying name, and name is undefined.

User Manual:

How to use our website.

General:

Every new user or an existing user that interacts with our Emed pharmacy website will must and should have account which is composed of unique credentials. This will be consisting of a completely unique username or an email address provided by the user and that should come with a unique password.

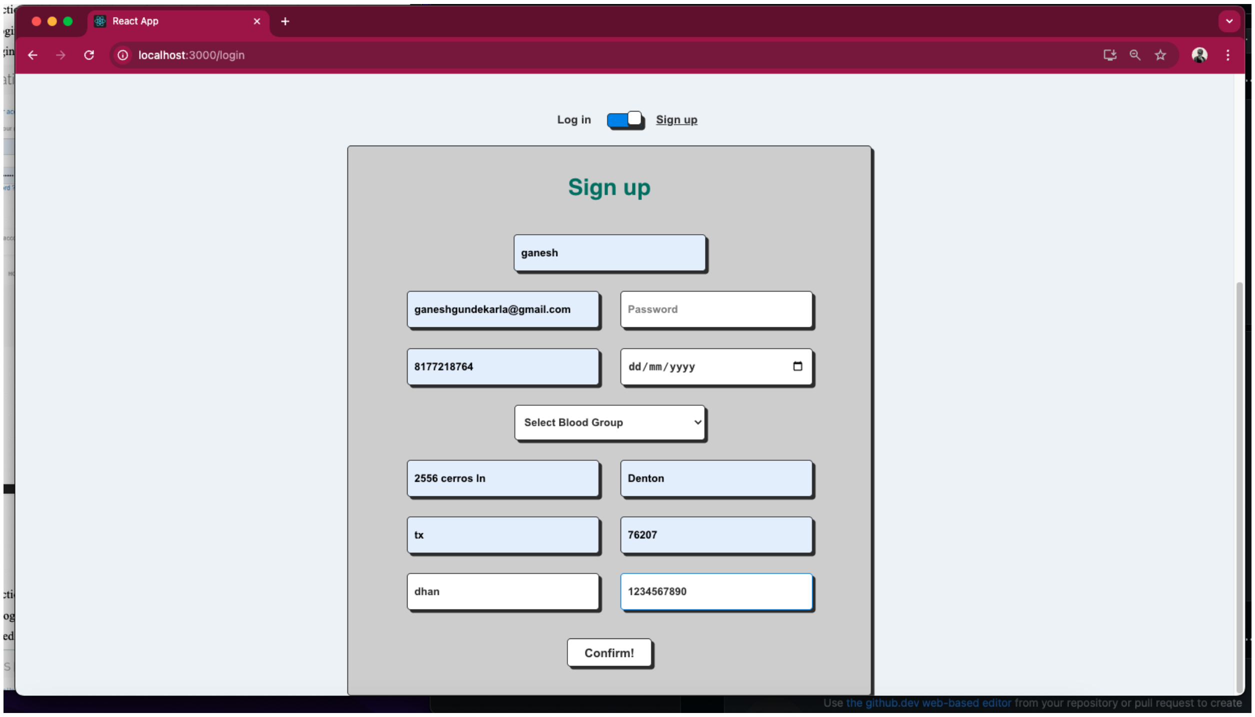
In our website, our administrator or the one who manages of databases will be able to create the accounts of our current existing doctors while the patients / users will get to create their very own accounts.

In addition, our users when logging in must be specifying their blood group so that whenever they choose to be a blood donor, the patients who require an emergency blood could easily be mapped to those donors.

The User Signup:

A) How to sign up as a new user:

1. At the top right corner, we have a login/register in order to give both options to a new user or an existing user.
2. Click on the login/register button to get redirected into a new page where we can either sign up or login.
3. Now, fill in the required details and then enter confirm.



ganesh

ganeshgundekarla@gmail.com

8177218764

2556 cerros ln

Denton

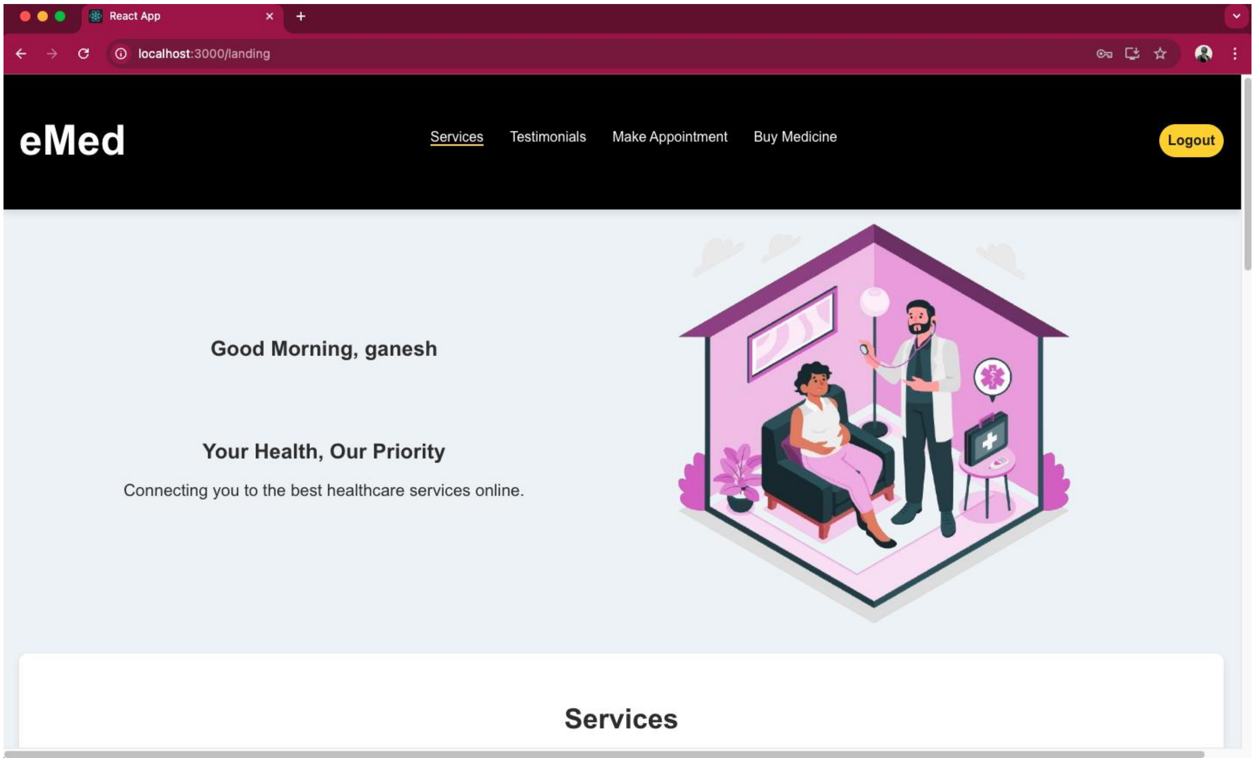
tx

76207

dhan

1234567890

4. This now takes us to the landing home page with us signed in with a message saying welcome back [username]



Good Morning, ganesh

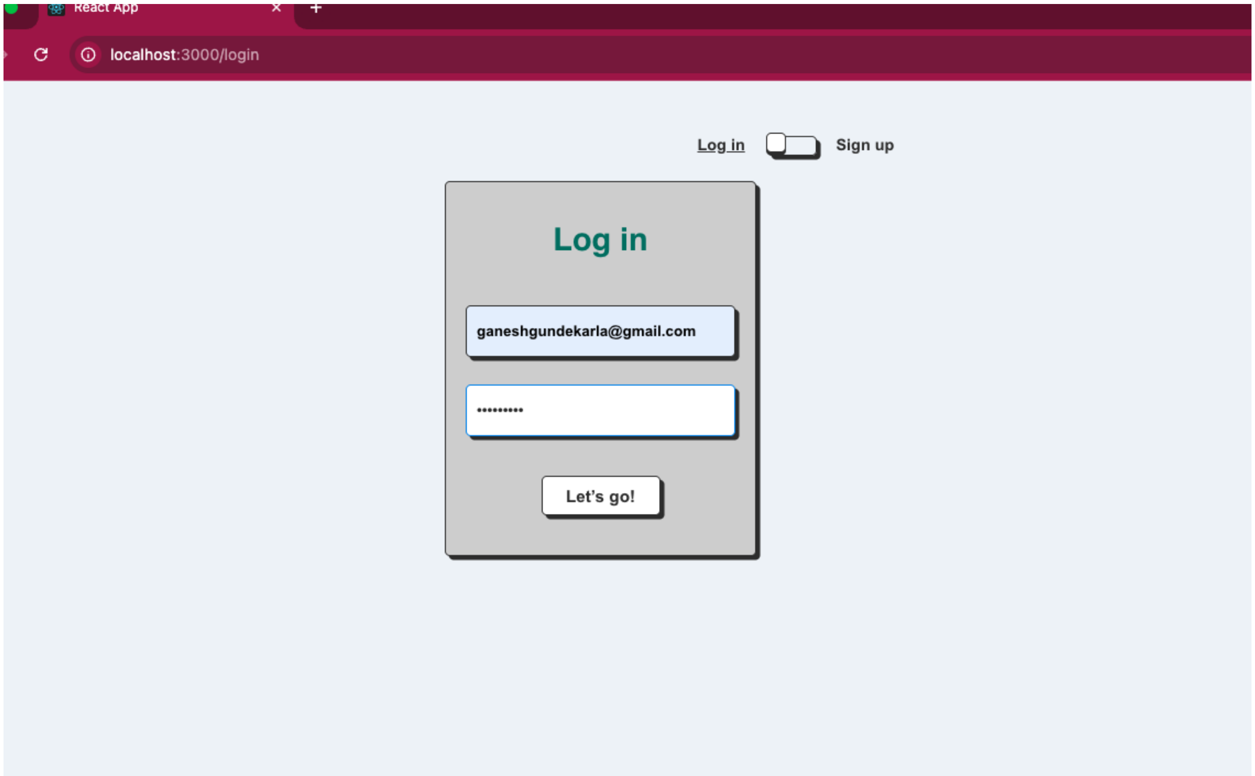
Your Health, Our Priority

Connecting you to the best healthcare services online.

Services

B) How to login from an existing user:

1. In the home page on the top right corner, click on the login/register button which directs us to login or signup page.
2. Now switch to login mode.
3. In the login page, enter your email address and password.
4. Now click on Let’s go! button.



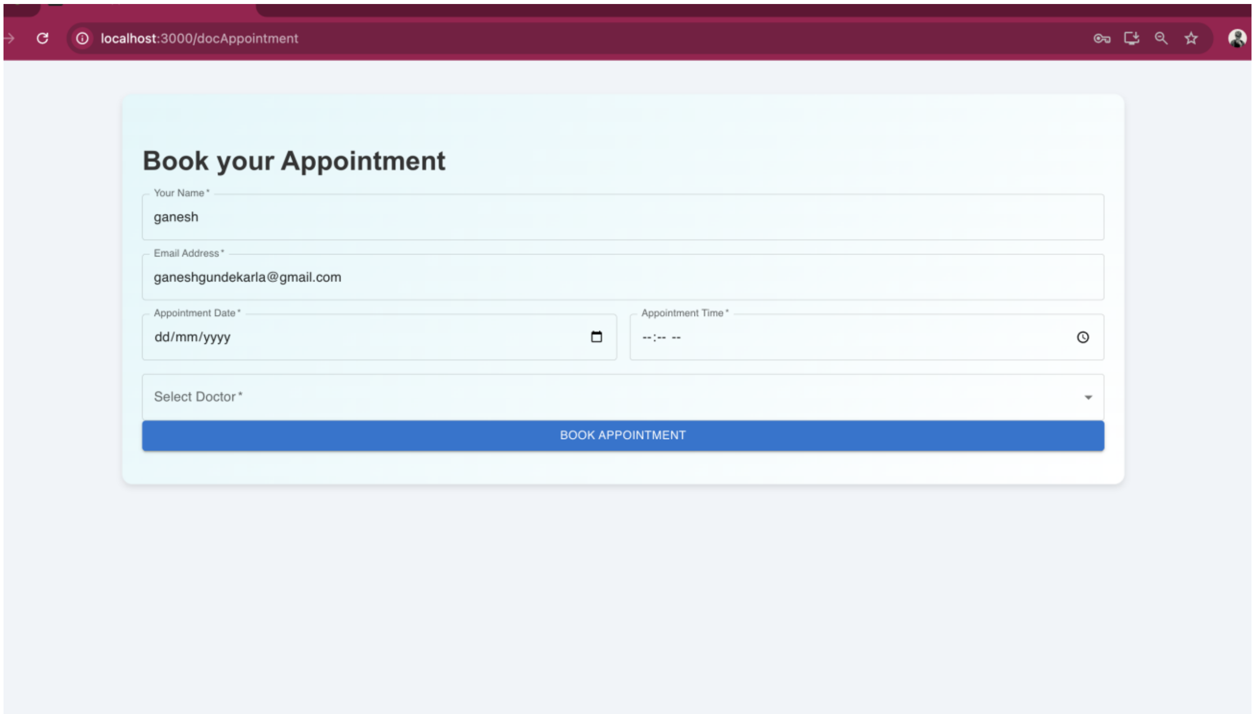
Log in

ganeshgundekarla@gmail.com

•••••••••

C) Steps on how to book an appointment:

1. In the home page, as you slide down, under the ‘Make an Appointment ‘section, select ‘Book now ‘to make an appointment.
2. Fill in the required details and enter your appointment date and time.
3. You are also presented with an option to select / choose your required doctor with his expertise in that field and press ‘BOOK APPOINTMENT’.



Book your Appointment

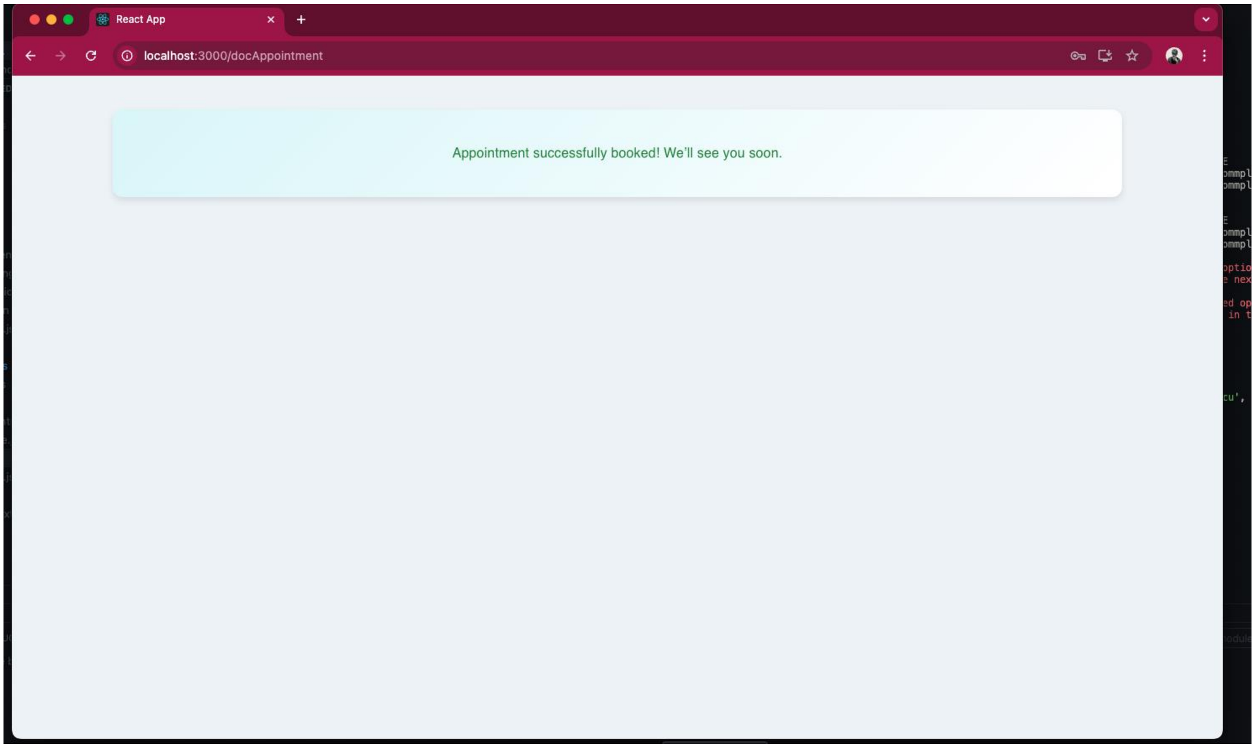
Your Name * ganesh

Email Address * ganeshgundekarla@gmail.com

Appointment Date * dd/mm/yyyy

Appointment Time * --:-- --

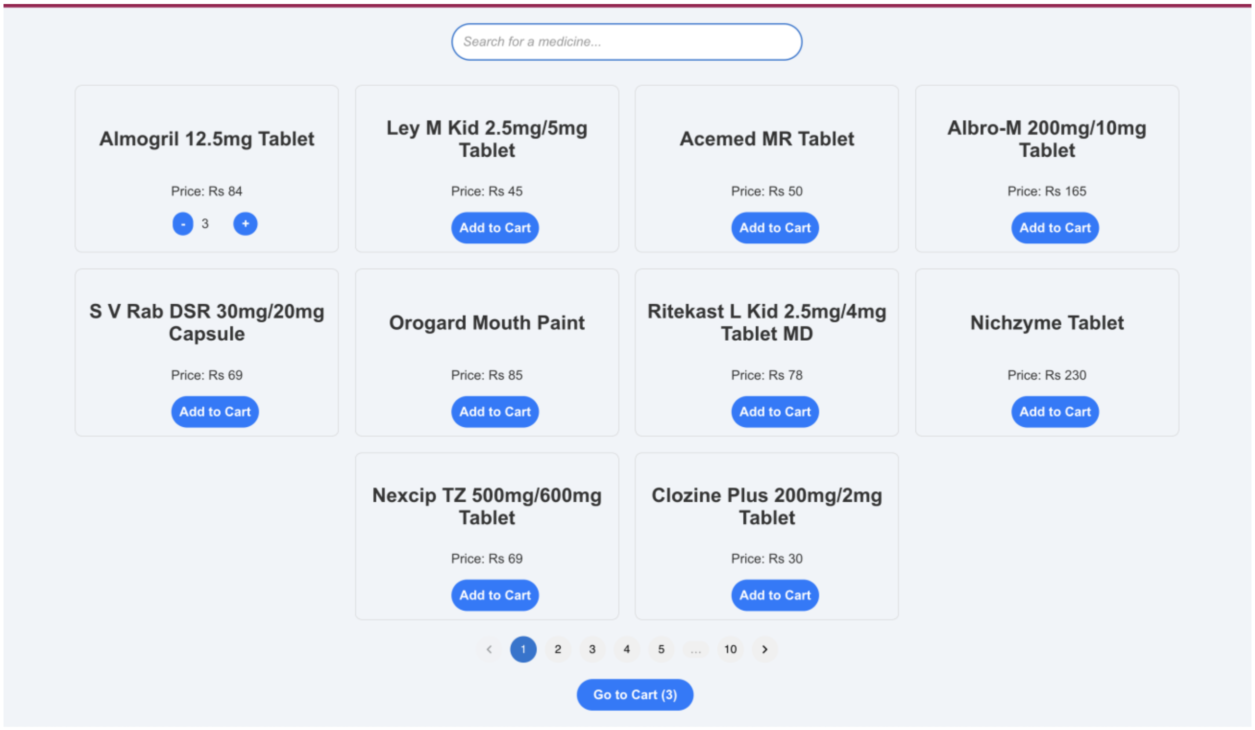
Select Doctor *



Appointment successfully booked! We'll see you soon.

D) Steps on how to order medicines:

1. In the landing page at in the bottom we will be having a ‘Order Medicines’ section where we can order medicines which are available currently.
2. After this we get redirected from landing page to the medicines page where we are presented with a wide variety of medicines.
3. Here, we can choose the medicine with the amount of tablets and it can be added to cart.
4. After we are done, we can press ‘Go to Cart’ which takes us to the cart page.
5. Here we have a proceed to checkout page which takes us to the final billing stage which will be implemented later.



Search for a medicine...

Almogril 12.5mg Tablet
Price: Rs 84
- 3 +

Ley M Kid 2.5mg/5mg Tablet
Price: Rs 45
Add to Cart

Acemed MR Tablet
Price: Rs 50
Add to Cart

Albro-M 200mg/10mg Tablet
Price: Rs 165
Add to Cart

S V Rab DSR 30mg/20mg Capsule
Price: Rs 69
Add to Cart

Orogard Mouth Paint
Price: Rs 85
Add to Cart

Ritekast L Kid 2.5mg/4mg Tablet MD
Price: Rs 78
Add to Cart

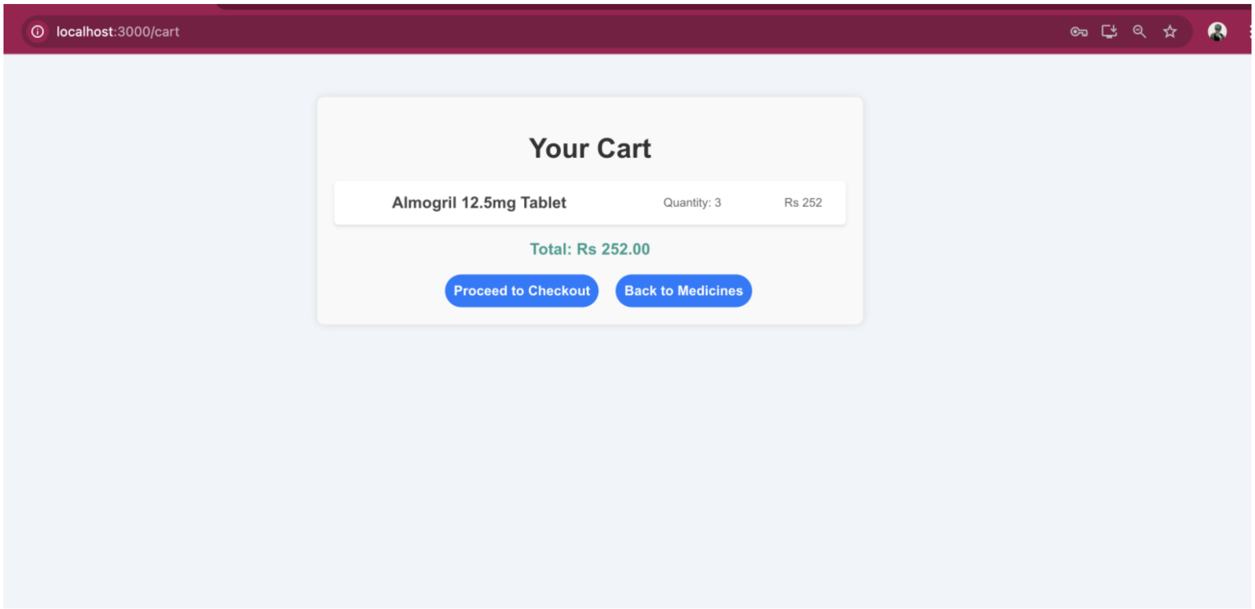
Nichzyme Tablet
Price: Rs 230
Add to Cart

Nexcip TZ 500mg/600mg Tablet
Price: Rs 69
Add to Cart

Clozine Plus 200mg/2mg Tablet
Price: Rs 30
Add to Cart

‹ 1 2 3 4 5 ... 10 ›

Go to Cart (3)



localhost:3000/cart

Your Cart

Almogril 12.5mg Tablet Quantity: 3 Rs 252

Total: Rs 252.00

Proceed to Checkout Back to Medicines

Instructions:

Environmental Configuration and for installation guide follow steps below:

- Download and install **Node.js** from Node.js official website.
- Download and install **MongoDB** from MongoDB official website
- Open a terminal or command and clone the code from GitHub and copy the code file.
- Open a terminal and start the MongoDB server using the command **mongod**
- Once started use the command line to create a new database.
- Name the database as we wish like hmtd**.**
- Import the MongoDB initial data from the **hmtd.json** file into MongoDB.
- Open a terminal and navigate to cloned project folder SOFTWARE_ENGINEERING_PROJECT_group7/src
- Install the required dependencies by running **npm install**
- Once the dependencies are installed, start the Node.js server using **npm start** command in the frontend and in the backend, you should navigate to server.js location and hit 'node server.js'.
- After the server is started and running, open your browser and go to the following URL to access the project. Make sure server is running on specified port i,e 3000.
- **https://localhost:3000**
- You can now run and test the project using the JavaScript (Node.js) backend and MongoDB database.

Reflection on the eHealth Project:

This eHealth project determined to create an extensive platform which connects patients with healthcare services, focusing on features like appointment scheduling, medicine purchases, blood donation information, and other is emergency services. The goal of this project is to provide an accessible and efficient way for users to interact with the healthcare facilities, making the whole process easier.

Challenges Faced

- At the development process, our team faced several challenges. One of the major hurdles was confirming user authentication and data security, given the sensitive nature of healthcare information. We knew it was hard to protect users' data, so we implemented strong security measures.

- Another significant challenge which was maintaining database performance. With features for example doctor profiles and patient records, efficient data storage and retrieval became a priority. For solving this issue, we designed a scalable database schema using MongoDB and focused on implementing indexing strategies to increase performance.

Solutions Implemented

- To solve these challenges, we decided on a modular architecture, which allowed us to develop and test multiple features independently. This structure helped keep our code keep organized and made it easier to update single components without slaughter the entire system. We used Node.js for the back-end services, which enabled efficient handling of user requests.

- For ensuring security, we integrated multiple tools to handle password encryption and also used some other tools to protect against common security vulnerabilities. We also utilized some tools for secure communication regarding account verification and reset password.

Lessons Learned:

- Working on this project which highlighted the importance of team collaboration and clear communication. Regular meetings allowed us to review our discuss challenges, progress, and improve our strategies. This collaborative spirit was important, especially when integrating multiple modules, which was required continuous feedback and testing to clarify everything functioned well together perfectly.

- We also learned that early testing and iterative development how essential is that. Some issues we solved during integration were overlooked initially, emphasizing the need for more frequent testing. Practicing a test-driven development (TDD) approach from the start could have helped us identify potential problems sooner.

Future Enhancements

- When we checked the project, we see several areas for future improvement. Increasing the user experience could involve adding advanced search features for blood donation and emergency requests. We also imagine introducing real-time updates for critical services, like ambulance tracking, and leveraging machine learning which could help us to analyze patient data for early risk detection.

- Additionally, as we prepare the project for larger healthcare networks, further optimization of the database schema and load balancing strategies will be necessary to manage increased traffic and data volume.

Impact and Value

- The eHealth project which holds the potential to significantly enhance patient care by simplifying access to healthcare services. With features that could support remote health monitoring, appointment scheduling, and medicine purchases, the platform fosters a more connected healthcare ecosystem. This connection enables users to receive timely care and assistance.

By providing a secure and user-friendly platform, we aim to reduce some of the pressure on healthcare facilities while providing a reliable resource for patients. Ultimately, this project not only improves the efficiency of healthcare support but also enables users to take control of their health through easy access to medical services and support.

Peer Review:

At the beginning of this, we held a group meeting to discuss Phase 1 tasks. During this conversation, we chose to use JavaScript (Node.js) for our project because of its high efficiency especially with asynchronous operations. We choose MongoDB for the database since its NoSQL structure met our requirements. Because many of our team members were already familiar with JavaScript, it was an ideal fit for both backend development and rapid team adoption.

We had a follow-up meeting to discuss progress on the tasks assigned at the initial meeting. We examined the loops that impact our coding standards and database structures. We pondered ways to implement the code in accordance with standards. -

During our final session, we chose to assess the code's performance and functionality to confirm that all test cases passed. So far, we have created the functionalities for Phase 1.

Concerns and how we addressed those issues:

a. There could be some possible vulnerable dependencies which are addressed by using the npm audit in order to check for any vulnerabilities in our node modules and updating them to latest security.
b. Sensitive api keys or the database credentials / tokens can be easily exposed when creating a website like this, we addressed this scenario by deciding to create a .env file and use environment variables in order to store sensitive data.
c. As our project grows every day, it might be possible that our code can be difficult for having readability. To address this primary concern, we have enforced a proper coding style across our system by using prettier in vscode.
d. Database concerns like no schema validation in MongoDB can be a possible risk and we decided to have a proper structure and also thinking to implement a backup mechanism.
e. We had reviewed all the test error cases and took preventive measures.

Roles Assignment:

Task	**Top preference**
Project Management Lead	Rishika Kandrigal
Requirements Lead	Ganesh Gundekarla
Design Lead	Vivek Nelluri
Implementation Lead for front-end	Ganesh Gundekarla
Implementation Lead for back-end	Vishnu Priya Vulichi
Configuration Management Lead	Vivek Nelluri
Testing Lead	Shabana Syed
Documentation Lead	Ajay Kumar Reddy Sammeta

Demo and Presentation Lead	Pranav Chalasani
System Administrator Lead	Md Ariful Hasan

SKILLS ASSIGNMENT:

Language/Technology	Assigned to:
Node.js with Express.js	Ganesh Gundekarla, Pranav Chalasani
MongoDB (via Mongoose)	Vishnu Priya Vulichi, Shabana Syed
React.js	Ganesh Gundekarla, Rishika Kandrigal, Pranav Chalasani
Axios for API requests	Vishnu Priya Vulichi, Ajay Kumar Reddy Sammeta
PayPal SDK (client-side)	Ajay Kumar Reddy Sammeta, Ganesh Gundekarla
JavaScript Object Notation	Md Ariful Hasan, Vivek Nelluri
JavaScript	Vivek Nelluri, Md Ariful Hasan
HTML5, CSS	Shabana Syed, Rishika Kandrigal

Team members Description:

Member name	Contribution description	Overall Contribution (%)	Note (if applicable)
Shabana Syed	I worked developing the frontend code for login page or home page along with building the sequence diagram for the project.	12.5%	
Ganesh Gundekarla	Worked on developing the app.js, index.js and reportWebVitals.js files which are pushed into the src directory in our git along with this i had also done the backend part of the server.js which acts as a core part for starting up our app. After execution , i had specified a neat user manual on how to operate the website .	12.5%	

Rishika Kandrigal	Worked on styling the frontend part mostly, i had implemented css sheets for login , medicine and swiper pages, along with this i also styled the index and app sheets. In the documentation part, i checked whether all the requirements are correctly used.	12.5%	
Md Ariful Hasan	Worked on the reflection of our eMed project, analyzing objectives and challenges. Implementing the testing part. Key insights on requirements and teamwork were gathered, along with recommendations for clear communication and adaptability.	12.5%	
Vishnu Priya Vulichi	Worked on implementing frontend code for login, medicine browsing and adding to cart. Also created class diagram for the models used and to be used in the project.	12.5%	
Vivek Nelluri	I worked on integrating jest-dom into our project's testing to extend Jest with custom matchers for testing and worked on compilation instructions and peer review.	12.5%	
Pranav Chalasani	Worked on frontend for creating the cart, creating the doctor appointment, creating a carousel. Also worked on test cases.	12.5%	
Ajay Kumar Reddy Sammeta	I worked on implementing the frontend code for the landing page and developed the Use-Case diagrams for the project.	12.5%	