Mi-12 Assignment Requirement

Assignment12_category_0022

Multi-Vendor Medicine Selling E-commerce Website

We're looking for a talented and eager Junior MERN Stack Developer to join our fantastic team. You'll collaborate closely with our experienced developers on state-of-the-art projects utilizing top technologies such as MongoDB for databases, Express.js for backend development, React.js for frontend interfaces, and Node.js for the runtime environment. This is your opportunity to enhance your skills and contribute to creating groundbreaking web applications that truly make a difference. Expect to work on features like user authentication, real-time updates, and responsive design, all while adhering to best practices in security and performance. If you're passionate about coding and eager to learn and grow in a dynamic environment, we want to hear from you!

Objective

You must develop a comprehensive web application for a multi-vendor e-commerce platform specializing in the sale of medicines and healthcare products. The platform should provide a seamless user experience for purchasing medications, submitting queries, and receiving expert advice using the MERN stack.

Key Rules:

- Include a minimum of 20 notable GitHub commits on the client side
- Include a minimum of 12 notable GitHub commits on the server side
- Add a meaningful readme.md file with the name of your website, Admin username, password and live site URL. Include a minimum of 10 bullet points to feature your website.
- Make it responsive for all devices. You need to make it responsive for mobile, tablet and desktop views. Make the dashboard responsive as well.
- After reloading the page of a private route, the user should not be redirected to the login page.

- Use the Environment variable to hide the Firebase config keys and Mongodb credentials.
- Don't use any Lorem ipsum text in your website.
- Show sweet alert/toast/notification for all the crud operations, successful authentication login, and sign-up. Don't use the default browser alert.
- Implement tanstack query in all the data fetching functionality (For GET method only)

Main Tasks

HOME PAGE:

1. Navbar has a **logo+website name, Home, Shop , cart icon, languages dropdown** and **Join US (when not logged in)** button. If the user is logged in, his/her profile picture should appear on the navbar.
If the user clicks on the **profile picture**, a drop-down will appear with the following items: **Update Profile, Dashboard,** and **Logout** button.

2. **Slider Section:** Here you will see all the Product slides added by the Admin. (note: admin can add and remove product slides from his dashboard). .**(see details in admin dashboard page instruction)**

3. **Category Card Section**: In this section, users will see all Category cards(minimum 6 cards), each displaying basic information such as the **category name, a basic image, and the number of medicines in that category**. Clicking on any Category card will navigate the user to a page where all medicines within that specific category are displayed in a tabular format..**(see details in Category details page instruction)**

4. **Discount products:** here the user will see all the product cards which have discounts. This section will be a draggable multiple card slider. (look swiperjs slider demos)

5. **Make two extra sections related to your website.**
6. **Footer:** Make a Relevant footer for the website

User Registration and Authentication:

1. **Sign Up Page :** On this page, users will see a sign-up form containing the following fields: username, email, photo upload input, password, select role (user or seller), and a sign-up button.
2. **Login Page**: In this page, users will see a login form with fields for email and password, along with a login button.
3. **Social Login:** Users can also log in using Google or GitHub. If a user logs in using Google or GitHub, the default role will be set to "user."

SHOP PAGE:

In this page the user will see all medicines information in tabular format. Each row will have a **select** button and an **eye** button. Clicking the eye button a modal will pop up where the user can see all the information of that medicine with its image. And clicking the select button will add the medicine to the cart.

CATEGORY DETAILS MEDICINE PAGE:

In this page the user will see all medicines information of a specific category(tablet,syrup,capsule,injection,others), in tabular format. Each row will have a **select** button and an **eye** button. Clicking the eye button a modal will pop up where the user can see all the information of that medicine with its image. And clicking the select button will add the medicine to the cart.

CART PAGE:

In this page the user will see all of his selected medicines with some basic information (name,company,price per unit, quantity). Users can increase or decrease quantity, can remove specific medicine or can clear all of the cart.
There will be a checkout button which will navigate to the checkout page.

Checkout page:

Here the user has to make payment of his grand total price of purchase using stripe payment. After payment is done it will navigate to an invoice page.

Invoice page: here user will see a nice invoice page with website logo, his information and purchase information. There will be a print button, by clicking the button the user can download the invoice as pdf.

Admin Dashboard PAGE:

Admin homepage: admin will see total sales revenue of the website. Paid total, pending total

(Features of admin dashboard)

1. **Manage Users.** In this page admin can make any user to seller or admin if he wants. Also he can downgrade any seller to a normal user.

2. **Manage Category:** Here admin can see all the categories of medicines in tabular format. He can update or delete any category from this page. There will be an **add category** button, clicking that button admin can add a new category using a modal form. Form will contain minimum(categoryName, categoryImage url or image upload.)

3. **Payment management:** Here admin will see all the payment information (pending and paid). Pending payment row will have an **accept payment** button, clicking that payment status will be paid.

4. **Sales Report:** In this page admin will see all sales information**(medicine name, seller email, buyer email, total price, etc)** in Tabular format, admin can filter reports using data range. Report can be downloaded in pdf/docs/csv/xlsx any of this format. (explore react data table related component libraries)

5. **Manage banner Advertise:** Here admin will see all Advertise medicine details (medicine image, medicine name ,description, seller email). Here will be a toggle button for each row, add to slide remove from slide. Added medicines images will be shown on the homepage slider section banner.

Seller Dashboard PAGE:

Seller homepage: seller will see total sales revenue of his medicines. (Paid total, pending total)

(Features of seller dashboard)

1. **Manage Medicines:** Here sellers will see all of his added medicines information in a tabular form. There will be an add **medicine** button. Clicking the button a modal form will pop up with some input fields(Item name, item generic name, short description, image upload, category dropdown, company dropdown, Item Mass Unit (Mg or ML), per unit price, discount percentage(DEFAULT ZERO) if available). It will be stored in the database.

2. **Payment History:** Here the seller will see all of the purchases history of his medicines by users. With their payment status(paid,pending)

3. **Ask For Advertisement:** Here seller will see all of his referred medicine information for advertisement slider, their status(if they are getting used in slider section or not). There will be an add advertise button, clicking the button modal will pop up where the seller can input a medicine image and some description for the advertisement slider.

USER DASHBOARD:

1. **Payment history:** here the user will see all of his payment history in tabular form with transaction id and status (pending/paid).

CHALLENGE PART:

1. For all of the medicine tables, you have to implement pagination, sort by price, and search for any information (medicine name, generic name, company name. etc)
2. Store access token in local storage and verify it on every private route api calls
3. Download pdf/excel format, for sales report page (link)
4. Date range filter for sales report page.
5. Use react hook form
6. Use react helmet

OPTIONAL PART:

- Translate whole website into different languages
- Add a digital time clock in the navbar of the website

What to Submit

1. **Assignment Category/variant:**
2. **Admin email:**
3. **Admin password:**
4. **Agent email:**
5. **Agent password:**
6. **Front-end Live Site Link:**
7. **Client Side Github Repository Link:**
8. **Server Side Github Repository Link:**