

## Module 4 – Sales App UI Design with React.js

### (Putting it all together – Creating Web-App Part 1)

- By Ajeet Kumar (Batch: 15th June)

**Project Title:** Sales App UI Design

**Date Prepared:** 30<sup>th</sup> September 2023

**Developed By:** Ajeet Kumar

**GitHub Repository:** <https://github.com/ajeetkumarrauniyar/Sales-App-UI>

## ## Introduction

The **Sales App UI** is a dynamic web application meticulously crafted using React.js, a versatile JavaScript library renowned for its capacity to build interactive and responsive user interfaces. This project report delves into the comprehensive development process, highlights the key features and functionalities of the Sales App UI, and outlines the project's architecture and setup.

## ## System Architecture

The Sales App UI follows a single-page application (SPA) architecture using React.js. It is a client-side application that provides a user-friendly interface for managing sales-related data. The primary components of the system include:

- **App.js:** This serves as the main entry point of the application and handles routing to various pages.
- **HeaderComponent.jsx:** This component provides navigation links and a search bar for user interaction.
- **AddSales.jsx:** Empowers users to effortlessly add sales entries by offering product selection, quantity, and amount input fields.
- **TopSales.jsx:** Showcases the top-performing five sales entries in an aesthetically pleasing and responsive table format.
- **Revenue.jsx:** Provides an at-a-glance view of the day's total revenue in an elegantly designed card.
- **Login.jsx:** Renders a login form for user authentication.

- **Register.jsx:** Extends the user experience with a registration form for new users.

## ## Resources Used:

- Visual Studio Code as the code editor
- Google Chrome as the web browser
- Bootstrap 5 as the frontend framework imported via CDN.
- Google Fonts “Poppins” as custom font, imported via CDN.

## ## Key Features:

### 1. Navigation Excellence

The application boasts an exceptional navigation bar, enabling users to seamlessly switch between various sections of the app. It offers easy access to functionalities such as adding sales, viewing top sales, checking daily revenue, and user authentication.

### 2. Streamlined Sales Entry

The Sales App UI provides a user-friendly interface for adding sales entries. Users can add sales entries by selecting a product, entering the quantity, and the amount. This data is then stored for analysis and reporting.

### 3. Top Sales Visualization

The "Top Sales" section is a visual delight, presenting the top 5 sales entries in a table format. The presentation includes product names, quantities, and sale amounts, allowing users to gain valuable insights effortlessly.

### 4. Revenue Insight

In the "Revenue" section, users are presented with a succinct view of today's total revenue. A clear and concise card format, complete with a currency symbol (₹), facilitates quick tracking of daily sales performance.

## 5. User Authentication

The application includes a user authentication system with login and registration pages, allowing authorized users to access and manage sales data.

### ## Getting Started / Project Setup

To get started with the Sales App UI, follow these steps:

1. Clone the repository to your local machine.

```
git clone https://github.com/ajeetkumarrauniyar/Sales-App-UI
```

2. Navigate to the project directory.

```
cd client
```

3. Install the required dependencies by running the following command:

```
npm install
```

4. Start the development server.

```
npm start
```

### ## Components Overview

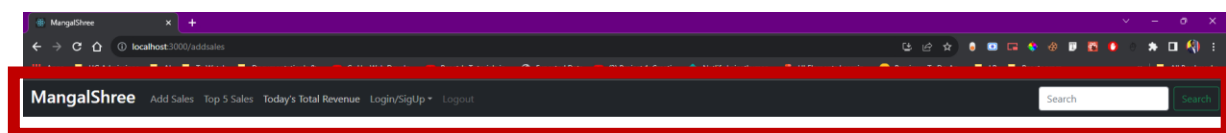
#### 1. App.js

- Acts as the main entry point for the application.
- Manages routing to different pages using React Router.
- Includes routes for adding sales, viewing top sales, checking revenue, login, and registration.

#### 2. HeaderComponent.jsx

- Provides a responsive navigation bar with links to different sections of the app.
- Includes a search bar for user convenience.
- The key elements in the navigation structure:

1. **Brand Logo:** The header includes a brand logo or application name, in this case, "MangalShree."
2. **Navigation Links:** The primary navigation links are presented as an unordered list (`<ul>`) with list items (`<li>`) inside. Each list item contains a `<Link>` element from React Router (**react-router-dom**) that directs the user to a specific route when clicked. These links allow users to navigate to different sections or pages of the application.
  - "Add Sales": Takes the user to the "Add Sales" page.
  - "Top 5 Sales": Navigates to the "Top 5 Sales" page.
  - "Today's Total Revenue": Leads to the "Today's Total Revenue" page.
  - "Login/SignUp" (Dropdown): Presents a dropdown menu with options for "Login" and "Register," which lead to the respective authentication pages.
3. **Search Bar:** A search bar is included for users to perform searches within the application. It consists of a search input field and a search button.
4. **Responsive Navbar:** The navigation bar is designed to be responsive. On smaller screens, a hamburger menu icon is displayed, allowing users to toggle the navigation menu.
5. **Logout (Disabled):** A "Logout" link is provided but is currently disabled. It's likely intended to log the user out when the authentication system is implemented.



Add Sale Entry

Product

Select the product

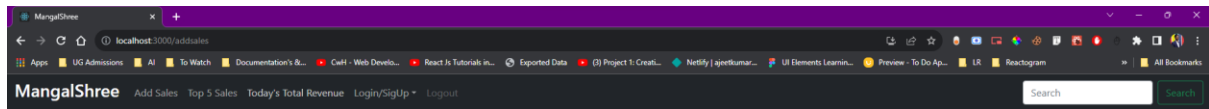
Quantity

Amount

Add Sale

### 3. AddSales.jsx

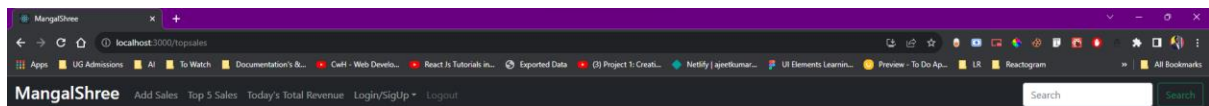
- Allows users to add sales entries.
- Includes fields for product selection, quantity, and amount.
- Provides a "Add Sale" button for submission.



localhost:3000/revenue

### 4. TopSales.jsx

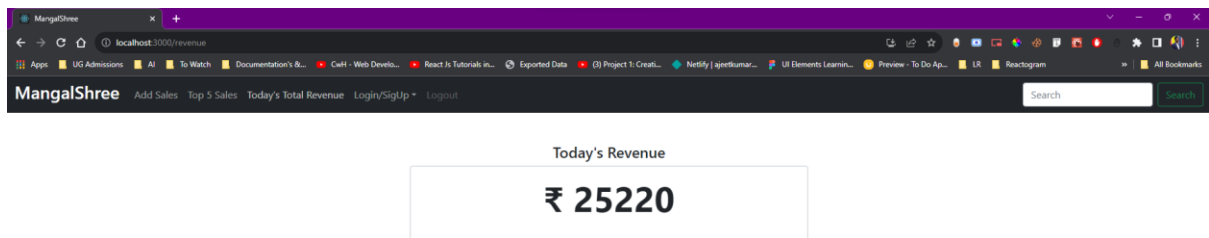
- Displays the top 5 sales entries in a table format.
- Includes columns for sales ID, product name, quantity, and sale amount.



#	Sales ID	Product Name	Quantity	Sale Amount
1	MS0001	RSO 1 LT PCH	10	17500
2	MS0002	KGMO 1 LT PCH	4	7720
3	MS0003	RBO 1 LT PCH	2	5500
4	MS0004	Sunflower Oil 1 LT PCH	1	4000
5	MS0005	Coconut Oil 1 LT PCH	2	5900

## 5. Revenue.jsx

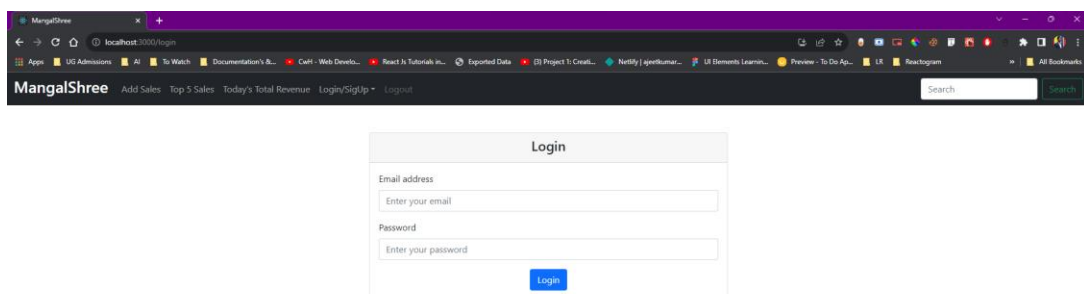
- Displays today's total revenue in a card format.
- Utilizes a currency symbol (₹) for visual clarity.



---

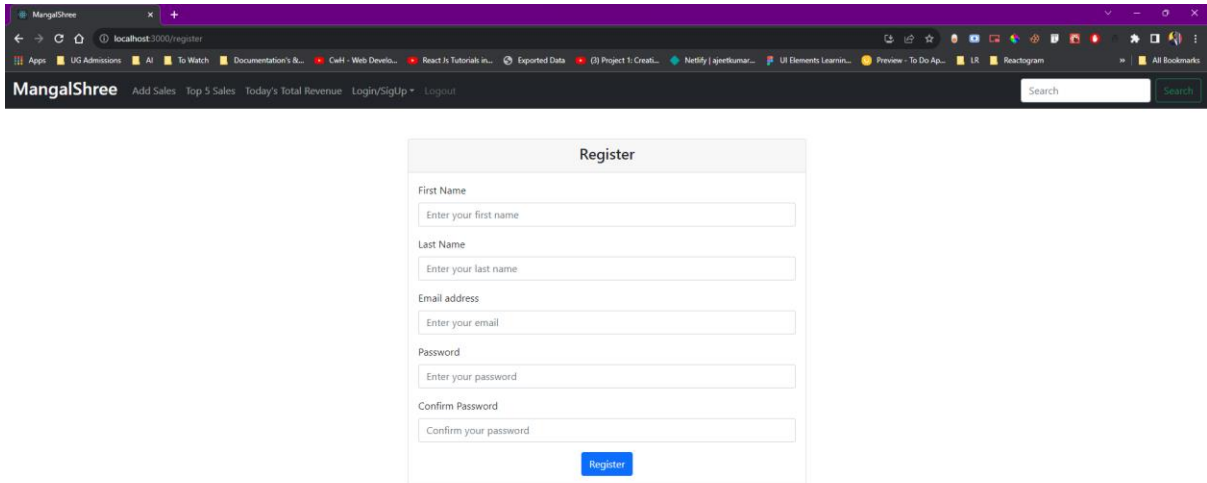
## 6. Login.jsx

- Renders a login form with fields for email and password.
- Provides a "Login" button for user authentication.



## 7. Register.jsx

- Provides a registration form with fields for first name, last name, email, password, and confirm password.
- Includes a "Register" button for user registration.



The screenshot displays a web browser window with the URL `localhost:3000/register`. The page features a navigation bar at the top with the logo "MangalShree" and links for "Add Sales", "Top 5 Sales", "Today's Total Revenue", "Login/SigUp", and "Logout". A search bar is also present on the right. The main content area contains a "Register" form with the following fields: "First Name" (placeholder: "Enter your first name"), "Last Name" (placeholder: "Enter your last name"), "Email address" (placeholder: "Enter your email"), "Password" (placeholder: "Enter your password"), and "Confirm Password" (placeholder: "Confirm your password"). A blue "Register" button is located at the bottom right of the form.

---

## ## Conclusion

The Sales App UI built with React.js offers an intuitive and user-friendly interface for managing sales data. It includes essential features such as adding sales, viewing top sales, and tracking daily revenue.

Future enhancements to the application could include, user authentication, data validation, user profiles, and the ability to edit and delete sales entries. The modular structure of the application allows for easy scalability and customization, making it suitable for various business needs.

This project showcases the capabilities of React.js in creating modern and responsive user interfaces for web applications. It provides a solid foundation for further development and refinement based on specific business requirements.