Assignment

React Router DOM Practice: SpaceX Web Page Layout

In this assignment, you will practice using React Router DOM to create a layout for a **SpaceX** web page. The page will consist of a header and a footer displayed on every route, while the content between them will change dynamically. Additionally, you will implement three routes and nested routing to display different components.

Instructions:

1. **Setup Environment:**

   - Set up a new React project using create-react-app or any other preferred method.

2. **Install Dependencies:**

   - Install React Router DOM: **npm install react-router-dom**.

3. **Create Header and Footer Components:**

   - Create a **Header.js** component to display the header with the text "SpaceX" centered.
   - Create a **Footer.js** component to display the footer with the text "All rights reserved 2024".

4. **Implement Routing:**

   - In the main App component, wrap the entire application with **BrowserRouter** Component from react-router-dom.
   - Define three routes using Route components from react-router-dom: Home, Capsule, and View.
   - Each route should render a component when its path matches the current URL.

5. **Create Capsule Component:**

   - Create a **Capsule.js** component to display multiple cards, each representing a capsule.
   - Each card should contain a "**View**" button.
   - Implement **onClick** functionality on the "**View**" button to navigate to a different route.
   - There will be a button named "**Go Back**" it will move back to previous route.

6. **Implement Nested Routing:**

   - Within the Capsule component, define a nested route using `Routes` and `Route` components.

   - When a "**View**" button is clicked, it should navigate to a new route by dynamically passing the Id of the selected capsule in the route which will be displayed in the address bar. Only details of the capsule with the id passed in the route parameter will be displayed on the page.

7. **Styling:**

   - Style the header and footer components to ensure they are visually appealing and consistent across all routes.

   - Apply CSS to Capsule component to display cards in a grid layout with appropriate spacing and styling.
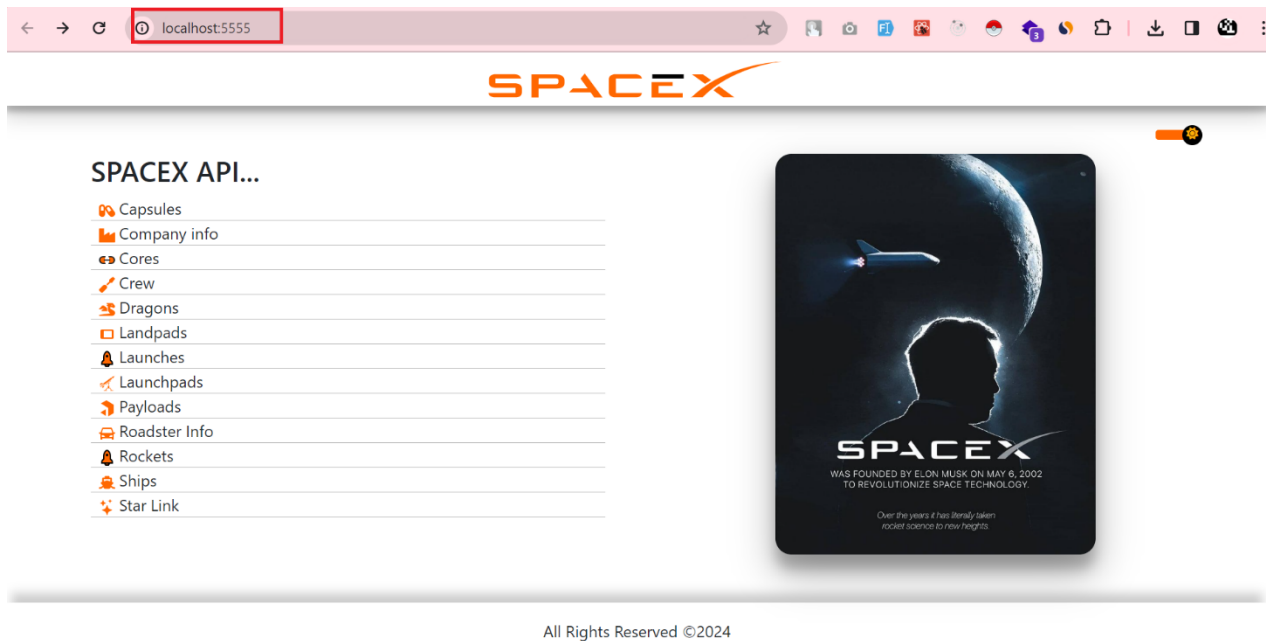
8. **Testing:**

   - Test the application by navigating between different routes and ensuring that the header and footer remain consistent.

   - Verify that clicking on the "View" button navigates to the correct route and displays the ID of the selected capsule in the address bar.
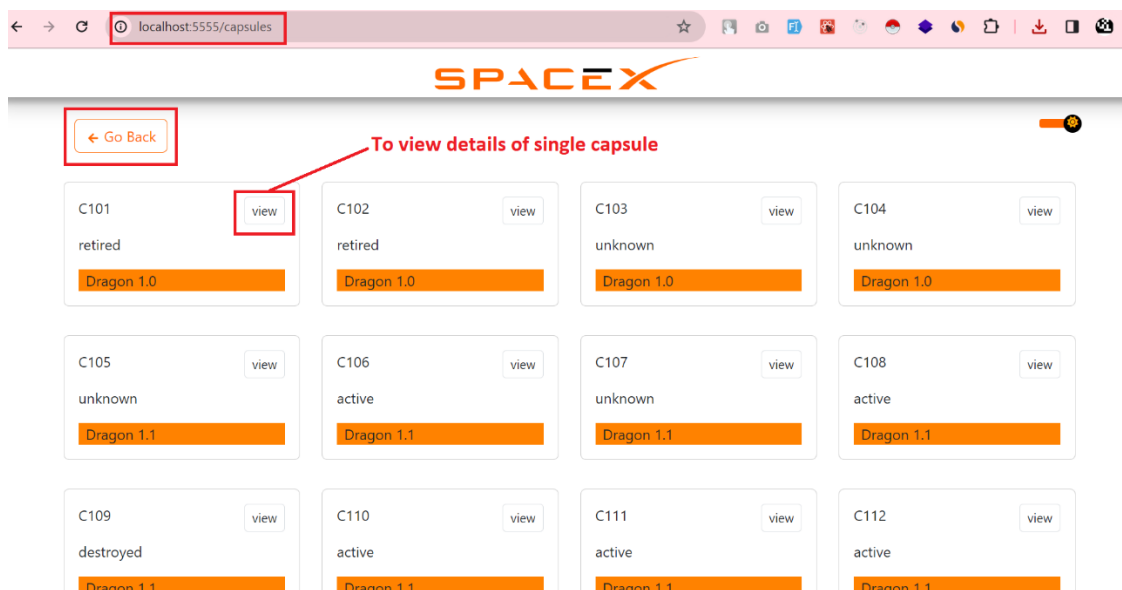
9. **Submission:**

- Submit the entire project directory, including all source code files and any additional assets or dependencies used.

Home Page Layout:

All Capsules Display Page:

Single Capsule details:

Individual Capsule Id

localhost:5555/capsules/5e9e2c5bf35918ed873b2664

Go Back

Upon clicking on go back you will move one route back

Single capsule details

Last Updata: Hanging in atrium at SpaceX HQ in Hawthorne

Reuse Count: 0

Water Landings: 1

Land Landings: 0

Launches: 5eb87cdeffd86e000604b330

Serial: C101

Status: retired

Type: Dragon 1.0

ID: 5e9e2c5bf35918ed873b2664