Findings for Front-End Web Development Project

Introduction: The project is a web application task for an online burger shop. The project is done with Node.JS & React framework, with HTML and CSS using VS code as the IDS.
The code:

The first task was to create the homepage Home.jsx that consist of navigation bar with links to Home, Contact, About, Shopping cart, and a Menu with links to Login, Orders and Logout.
The Home.jsx file is a React component file that imports the required modules such, as React, Framer Motion (for animation) and local components like Founder and Menu. It also defines an object called options for configuring the animation, which includes specifying the off screen and visible states. The Home component is a component that displays a message, inside a div. This entire component is exported as the default option so that it can be used in parts of the application.

Code:

```
import React from "react";
import { motion } from "framer-motion";
import Founder from "./Founder";
import Menu from "./Menu";

const options = {
 initial: {
  x: "-100%",
  opacity: 0,
 },
 whileInView: {
  x: 0,
  opacity: 1,
 },
};

const Home = () => {
 return (
  <div>
   {/* Your home component content */}
   <h1>Welcome to the Home Page</h1>
  </div>
 );
};

export default Home;
```

The `Menu.jsx` file is a React component that has the role of displaying a `MenuCard` with all the information, about a cheeseburger item. It imports images of burgers. Also includes a function called `addToCartHandler` to handle any logic related to the cart. This component is then exported so that it can be used in sections of the application.

```
// Menu.jsx
import React from "react";
import MenuCard from "./MenuCard";
import burger1 from "../../assets/burger1.png";
import burger2 from "../../assets/burger2.png";
import burger3 from "../../assets/burger3.png";

const addToCartHandler = () => {
  // Define your addToCartHandler logic here
};

const Menu = () => {
  return (
    <MenuCard
      itemNum={1}
      burgerSrc={burger1}
      price={200}
      title="Cheese Burger"
      handler={addToCartHandler}
      delay={0.1}
    />
  );
};

export default Menu;
```

The About.jsx file is responsible, for generating a webpage that introduces a Burger Shop. It consists of a heading, a short article section about the shop and a subsection dedicated to the founder. In this subsection you will find an image, the name of the founder (Nelson) and a description of Ifeanyi, the founder along with their association, with taste. The entire content is organized using HTML elements within a React component.

Code:

```
// About.jsx
import React from "react";
```

```
import { RiFindReplaceLine } from "react-icons/ri";
import { Link } from "react-router-dom";
import me from "../../assets/skj.jpg";

const About = () => {
 return (
  <div>
   <h1>About Us</h1>
   <article>
    <h4>Burger Shop</h4>
    <p>This is Burger Shop. The place for the most tasty burgers on the entire earth.</p>
    <p>Explore the various types of food and burgers. Click below to see the menu.</p>
   </article>
   <div>
    <h2>Founder</h2>
    <article>
     <div>
      <img src={me} alt="Founder" />
      <h3>Nelson</h3>
     </div>
     <p>I am Ifeanyi, the founder of Burger Shop. Affiliated to good Taste...</p>
    </article>
   </div>
  </div>
 );
};

export default About;
```

The Contact.jsx uses the reactjs popup library to generate a popup that appears when the "Send" button is clicked. The content of the popup is currently commented out. Can be customized to suit contact related functionalities or forms.

code:

```
import React from 'react';
import Popup from 'reactjs-popup';

const Contact = () => {
 return (
  <Popup trigger={<button type="button">Send</button>}>
   {/* Your popup content */}
  </Popup>
```

```
  );
};

export default Contact;
```

The `Login.jsx` defines a login page using the `mdb-react-ui-kit` library. It includes a tabbed layout with the initial tab showing a login form. The form supports signing in with social media accounts through icons like Facebook, Twitter, Google, and GitHub. Users can also sign in with their email and password, and there is an option for remembering credentials. The page includes a "Forgot password?" link and a prompt for users to register if they are not members. The entire layout is wrapped in an MDBContainer for styling.

Code:

```
import React, { useState } from "react";
import {
  MDBContainer,
  MDBTabs,
  MDBTabsItem,
  MDBTabsLink,
  MDBTabsContent,
  MDBTabsPane,
  MDBBtn,
  MDBIcon,
  MDBInput,
  MDBCheckbox,
} from "mdb-react-ui-kit";

const Login = () => {
  const [justifyActive, setJustifyActive] = useState("tab1");

  return (
    <MDBContainer>
      <MDBTabsPane show={justifyActive === "tab1"}>
        <div className="text-center mb-3">
          <p>Sign in with:</p>
          <div className="d-flex justify-content-between mx-auto" style={{ width: "40%" }}>
            <MDBBtn tag="a" color="none" className="m-1" style={{ color: "#1266f1" }}>
              <MDBIcon fab icon="facebook-f" size="sm" />
            </MDBBtn>
            <MDBBtn tag="a" color="none" className="m-1" style={{ color: "#1266f1" }}>
              <MDBIcon fab icon="twitter" size="sm" />
            </MDBBtn>
```

```
          <MDBBtn tag="a" color="none" className="m-1" style={{ color: "#1266f1" }}>
            <MDBIcon fab icon="google" size="sm" />
          </MDBBtn>
          <MDBBtn tag="a" color="none" className="m-1" style={{ color: "#1266f1" }}>
            <MDBIcon fab icon="github" size="sm" />
          </MDBBtn>
        </div>
        <p className="text-center mt-3">or:</p>
      </div>
      <MDBInput wrapperClass="mb-4" label="Email address" id="form1" type="email" />
      <MDBInput wrapperClass="mb-4" label="Password" id="form2" type="password" />
      <div className="d-flex justify-content-between mx-4 mb-4">
        <MDBCheckbox
          name="flexCheck"
          value=""
          id="flexCheckDefault"
          label="Remember me"
        />
        <a href="!#">Forgot password?</a>
      </div>
      <MDBBtn className="mb-4 w-100">Sign in</MDBBtn>
      <p className="text-center">
        Not a member? <a href="#!">Register</a>
      </p>
    </MDBTabsPane>
  </MDBContainer>
 );
};

export default Login;
```

The next task is to create the Cart and Shipping page

In the Cart page, each burger item is showcased using the CartItem component. You'll find sections for each burger type with buttons to adjust quantities. The total, tax, shipping charges, and grand total are calculated based on item prices and quantities. We've added a link for a smooth transition to the shipping page.
The CartItem component takes care of displaying specific item details like title, image, and buttons for quantity adjustment. Placeholder functions (increment and decrement) are set up for managing quantity changes.The Cart component dynamically calculates essential details like subtotal, tax, shipping charges, and the total amount. Plus, there's a straightforward link for users to proceed to the shipping page and complete their order.

Code:

```
import React from "react";
import { Link } from "react-router-dom";
import burger1 from "../../assets/burger1.png";
import burger2 from "../../assets/burger2.png";


// import burger3 here

const CartItem = ({ value, title, img, increment, decrement }) => (
  <div className="cartItem">
    <div>
      <h4>{title}</h4>
      <img src={img} alt="Item" />
    </div>

    <div>
      <button onClick={decrement}>-</button>
      <input type="number" readOnly value={value} />
      <button onClick={increment}>+</button>
    </div>
  </div>
);

const Cart = () => {
  const increment = (item) => {};

  const decrement = (item) => {};

  return (
    <section className="cart">
      <main>
        <CartItem
          title={"Cheese Burger"}
          img={burger1}
          value={0}
          increment={() => increment(1)}

        // Add the function for decrementing the order by 1
```

```
      />
      <CartItem
        title={"Veg Cheese Burger"}
        img={burger2}
        value={0}
        increment={() => increment(2)}
      // Add the function for decrementing the order by 2

      />

      {/* Fill up the code for Cheese Burger similarly */}


      <article>
        <div>
          <h4>Sub Total</h4>
          <p>Γé╣{2000}</p>
        </div>
        <div>
          <h4>Tax</h4>
          <p>Γé╣{2000 * 0.18}</p>
        </div>
        <div>
          <h4>Shipping Charges</h4>
          <p>Γé╣{200}</p>
        </div>{" "}
        <div>
          <h4>Total</h4>
          <p>Γé╣{2000 + 2000 * 0.18 + 200}</p>
        </div>
        <Link to="/shipping">Checkout</Link>
      </article>
    </main>
  </section>
 );
};

export default Cart;
```

The Shipping page features a user-friendly form for entering shipping details. Users can input house number, city, and phone number. Dropdowns for country and state are interactive, with options dynamically changing based on user selections.

The Country dropdown is populated with a variety of countries, and the State dropdown updates according to the chosen country, providing a personalized experience. A section is included for users to input their contact details, specifically their phone number.
To add a touch of confirmation, we've integrated a Popup component from 'reactjs-popup'. Clicking the "Confirm Order" button triggers a friendly confirmation message, appearing on the right-center of the page with a warm red background, signifying a successful order placement.

Code:

```
import React from "react";
import { Country, State } from "country-state-city";
import Popup from 'reactjs-popup';

const Shipping = () => {
 return (
  <section className="shipping">
   <main>
    <h1>Shipping Details</h1>
    <form>
     <div>
      <label>H.No.</label>
      <input type="text" placeholder="Enter House No." />
     </div>
     <div>
      <label>City</label>
      <input type="text" placeholder="Enter City" />
     </div>
     <div>
       {/* Compelte the code for the COUNTRY DROPDOWN*/}
      <label>Country</label>

      <select>
       <option value="">Country</option>
   // Enter the code here for country dropdown

     <option value="">Country</option>
     {Country && Country.getAllCountries().map((i) => (
     <option value="{i.isoCode}" key="{i.isoCode}">
     {i.name}
     </option>
     ))}
     </select>
     </div>
```

```
        <div>
          {/* Add the code for the STATE DROPDOWN*/}
          <label>State</label><select>
          <option value="">State</option>
          {State && State.getStatesOfCountry("IN").map((i) => (
          <option value="{i.isoCode}" key="{i.isoCode}">
          {i.name}
          </option>
          ))}
          </select>
        </div>
       // Enter thr code for contact
          <div>
//
          <label>Phone No.</label>
          //
          <input type="number" placeholder="Enter Phone No." />
          //
          </div>
        <Popup trigger=
            {<button type = "button">Confirm Order</button>}
            position="right center">
            <div style={{color:"red",position: 'absolute', top: '50%', right: '100%', transform: 'translateY(-
50%)', backgroundColor: '#fff', padding: '10px', borderRadius: '5px', boxShadow: '0 0 10px rgba(0, 0, 0,
0.2)'}}>Order Placed</div>

          </Popup>
       </form>
     </main>
   </section>
  );
};
export default Shipping;
```

Next was creatin the order page that shows the datails of an order made by a customer.

Code:

```
import React from "react";
import { Link } from "react-router-dom";
import { AiOutlineEye } from "react-icons/ai";
```

```
const MyOrders = () => {
  const arr = [1, 2, 3, 4];

  return (
    <section className="tableClass">
      <main>
        <table>
          <thead>
            <tr>
              <th>Order Id</th>
              <th>Status</th>
              <th>Item Qty</th>
              <th>Amount</th>
              <th>Payment Method</th>
              <th>Action</th>
            </tr>
          </thead>

          <tbody>
//    Add the code for the table body

          </tbody>
        </table>
      </main>
    </section>
  );
};

export default MyOrders;
```

The MyOrders component renders a table structure with headers such as "Order Id," "Status," "Item Qty," "Amount," "Payment Method," and "Action," creating the basic layout for displaying order information. The table body section (<tbody>) is currently commented out with a placeholder comment instructing to add the code for the table body. This indicates that the component is expected to dynamically render rows of order information, but the implementation for this part is incomplete. This status of the order placed by the user could be seen on OrderDetails.jsx which is meant to dynamically display the state of the order.