FOOD DEL MERN Progress update:

Cart page (beginning)

Code:  
import React, { useContext } from "react";

import "./Cart.css";

import { StoreContext } from "../../context/StoreContext";

const Cart = () => {

  const { cartItems, food\_list, removeFromCart } = useContext(StoreContext);

  return (

    <div className="cart">

      <div className="cart-items">

        <div className="cart-items-title">

          <p>Items</p>

          <p>Title</p>

          <p>Price</p>

          <p>Quantity</p>

          <p>Total</p>

          <p>Remove</p>

        </div>

        <br />

        <hr />

      </div>

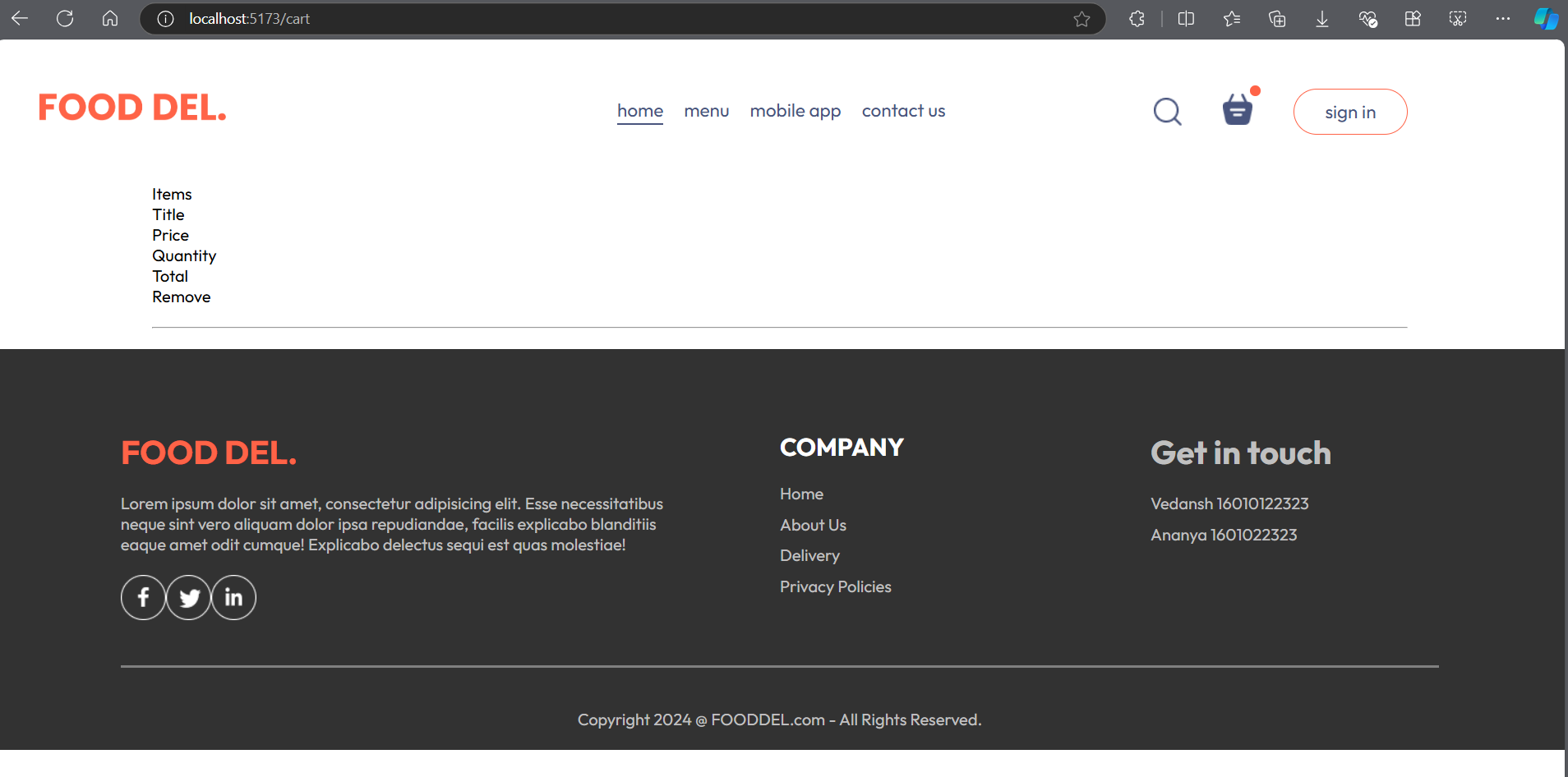
    </div>

  );

};

export default Cart;

output



import React, { useContext } from "react";

import "./Cart.css";

import { StoreContext } from "../../context/StoreContext";

const Cart = () => {

  const { cartItems, food\_list, removeFromCart } = useContext(StoreContext);

  return (

    <div className="cart">

      <div className="cart-items">

        <div className="cart-items-title">

          <p>Items</p>

          <p>Title</p>

          <p>Price</p>

          <p>Quantity</p>

          <p>Total</p>

          <p>Remove</p>

        </div>

        <br />

        <hr />

      </div>

      {food\_list.map((item, index) => {

        if (cartItems[item.\_id] > 0) {

          return (

            <div className="cart-items-title cart-items-item" key={index}>

              <img src={item.image} alt="" />

              <p>{item.name}</p>

              <p>{item.price}</p>

              <p>{cartItems[item.\_id]}</p>

              <p>{item.price \* cartItems[item.\_id]}</p>

            </div>

          );

        }

      })}

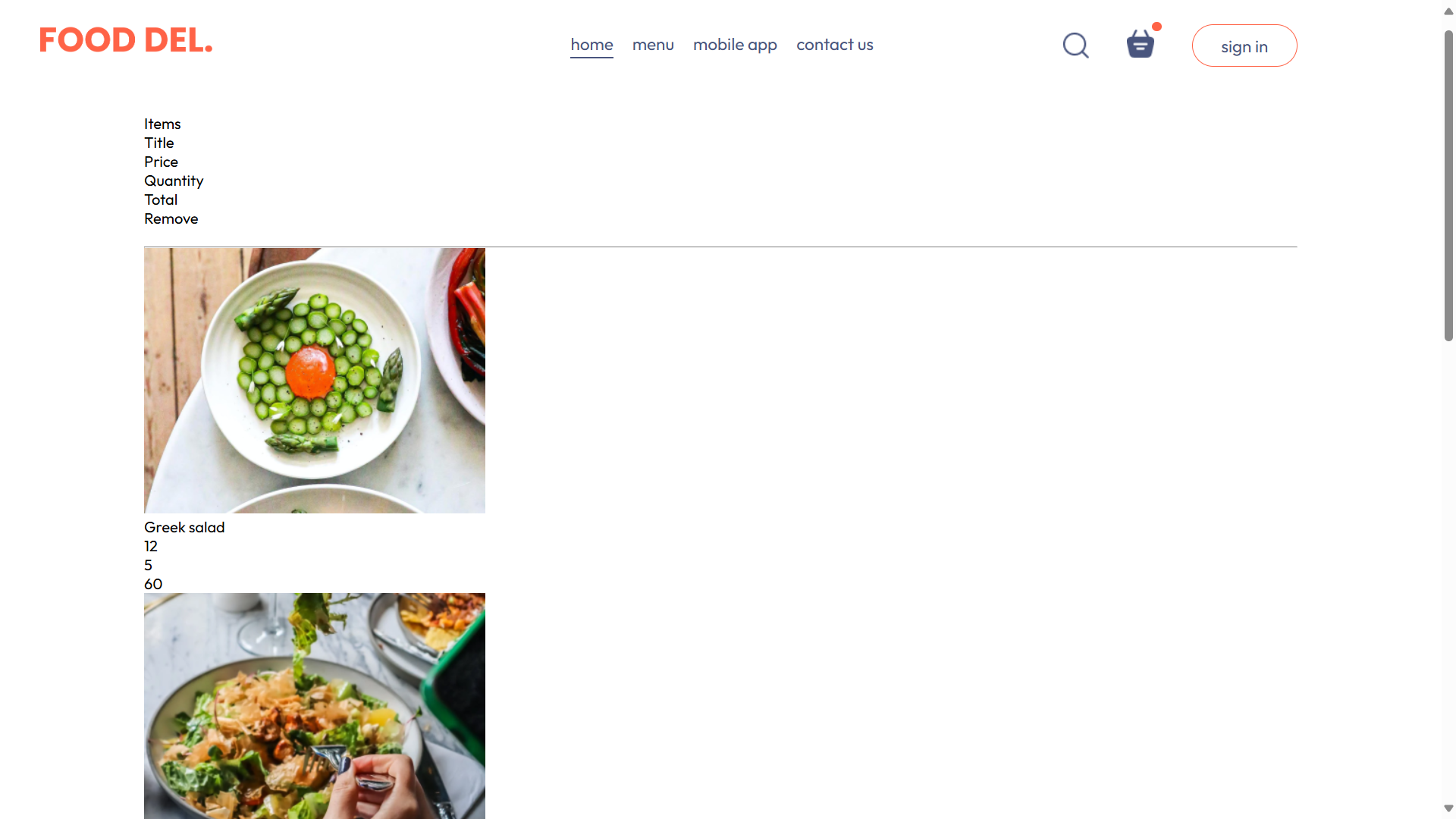
    </div>

  );

};

export default Cart;

output :



Adding style to the cart page   
code :  
  
.cart {

  margin-top: 100px;

}

.cart-items-title {

  display: grid;

  grid-template-columns: 1fr 1.5fr 1fr 1fr 1fr 0.5fr;

  align-items: center;

  color: grey;

  font-size: max(1vw, 12px);

}

.cart-items-item {

  margin: 10px 0px;

  color: black;

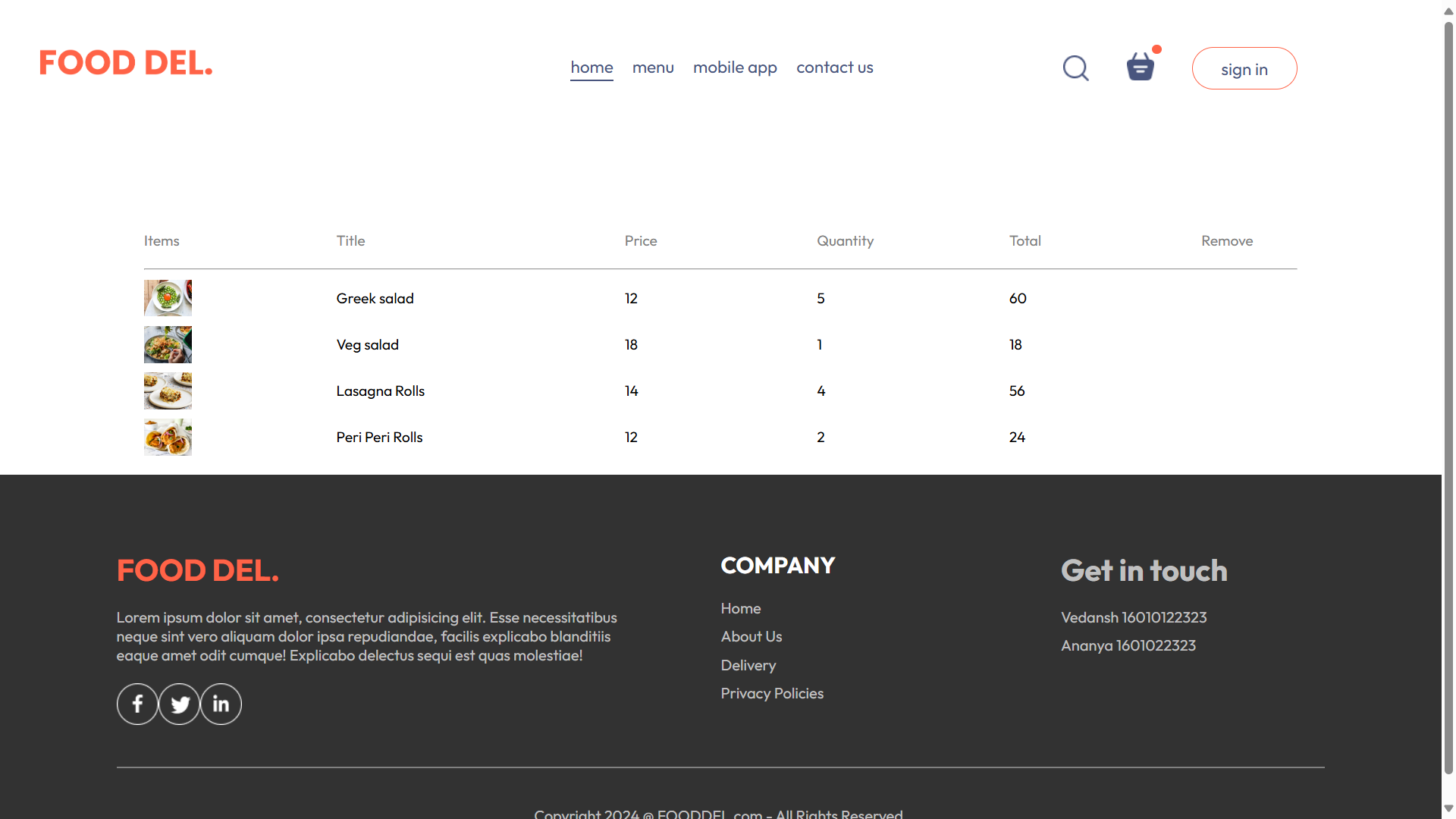
}

.cart-items-item img {

  width: 50px;

}

Output:



import React, { useContext } from "react";

import "./Cart.css";

import { StoreContext } from "../../context/StoreContext";

const Cart = () => {

  const { cartItems, food\_list, removeFromCart } = useContext(StoreContext);

  return (

    <div className="cart">

      <div className="cart-items">

        <div className="cart-items-title">

          <p>Items</p>

          <p>Title</p>

          <p>Price</p>

          <p>Quantity</p>

          <p>Total</p>

          <p>Remove</p>

        </div>

        <br />

        <hr />

      </div>

      {food\_list.map((item, index) => {

        if (cartItems[item.\_id] > 0) {

          return (

            <div key={index}>

              <div className="cart-items-title cart-items-item">

                <img src={item.image} alt="" />

                <p>{item.name}</p>

                <p>₹{item.price}</p>

                <p>{cartItems[item.\_id]}</p>

                <p>₹{item.price \* cartItems[item.\_id]}</p>

                <p onClick={() => removeFromCart(item.\_id)} className="cross">

                  x

                </p>

              </div>

              <hr />

            </div>

          );

        }

      })}

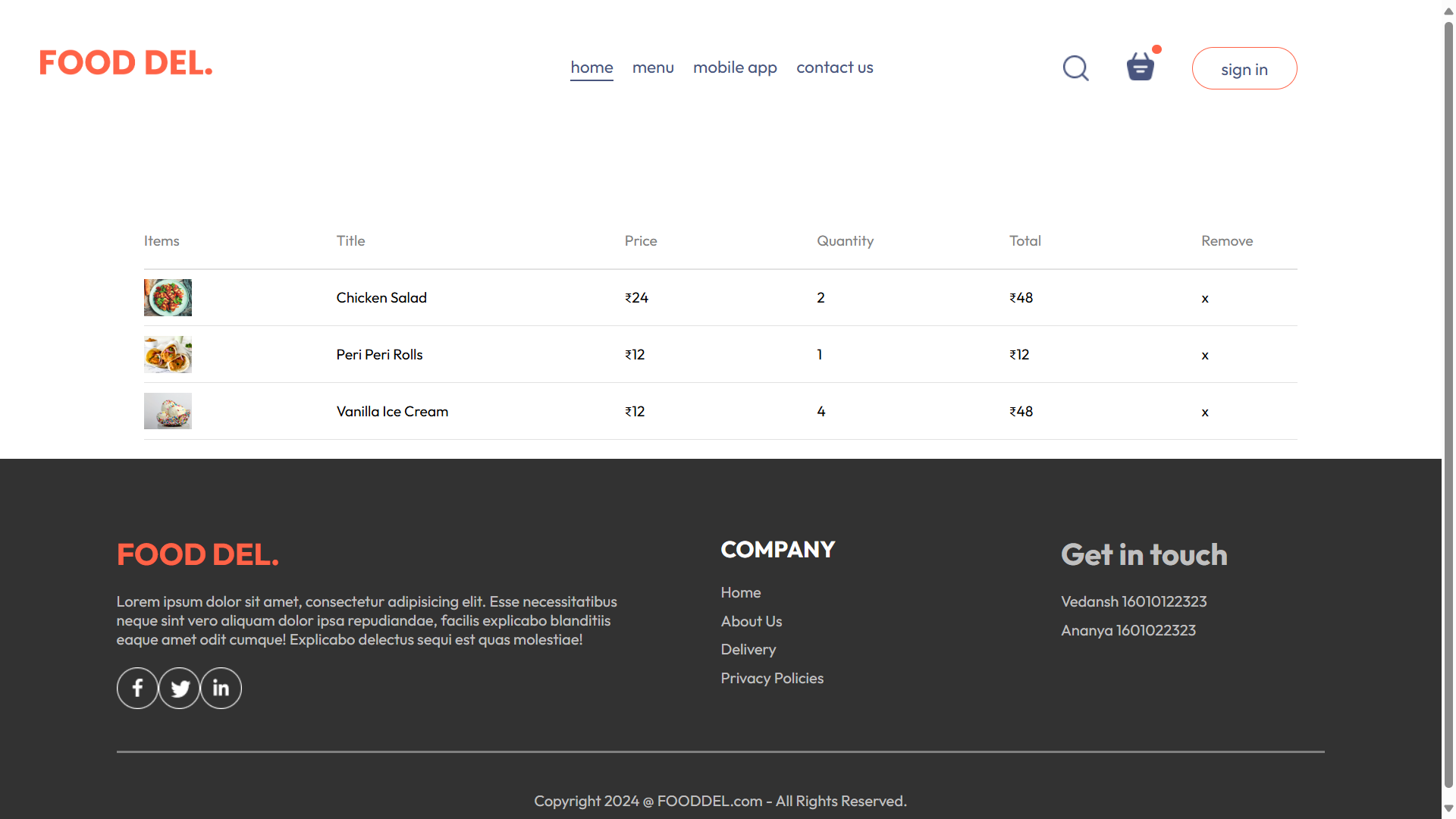
    </div>

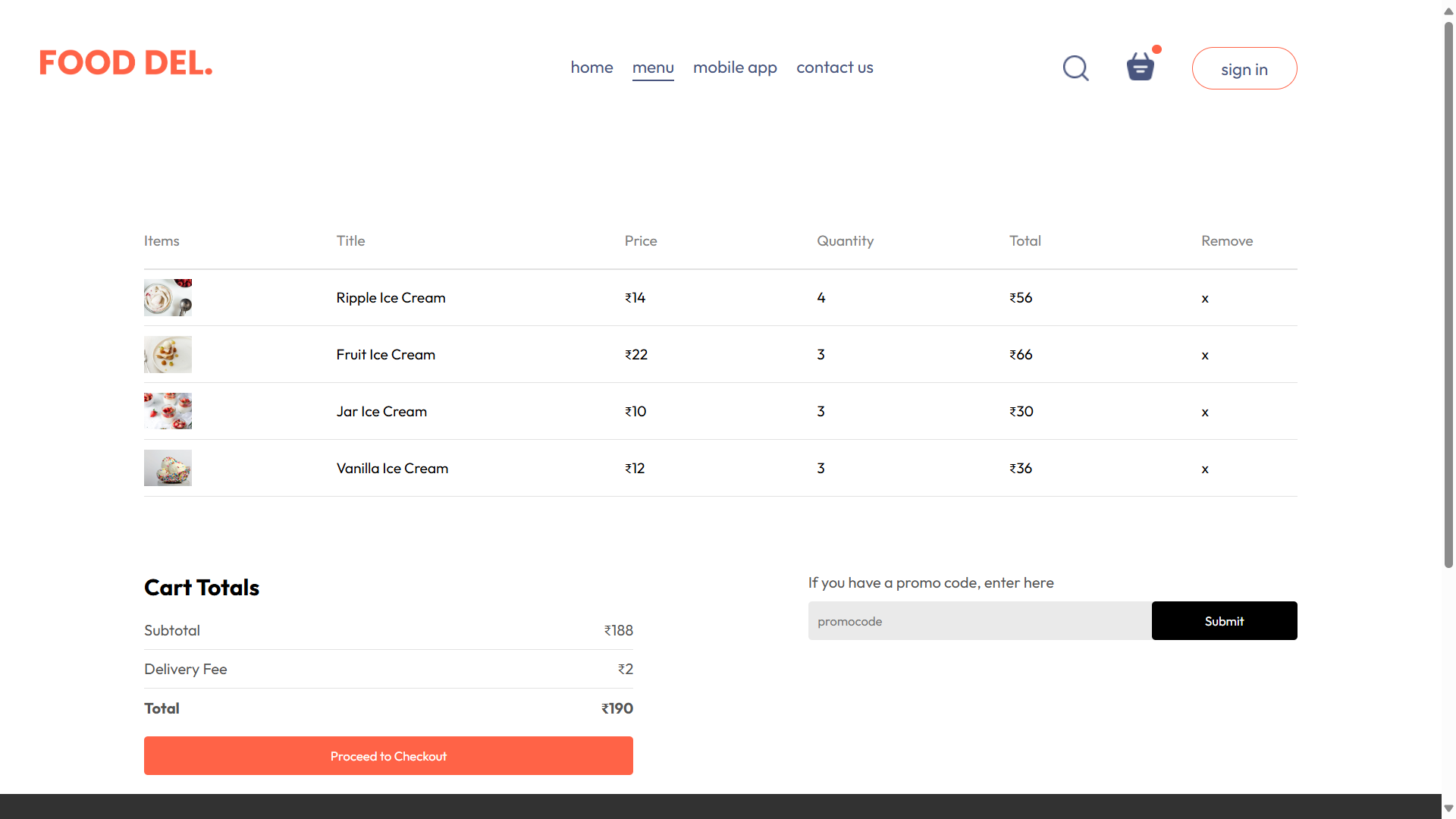
  );

};

export default Cart;

output:



Promocode:  


CHANGES IN THE STORE CONTEXT JSX FILE : getTotalAmount:

import React, { useEffect, useState, createContext } from "react";

import { food\_list } from "../assets/assets";

export const StoreContext = createContext(null);

const StoreContextProvider = (props) => {

  const [cartItems, setCartItems] = useState({});

  const addToCart = (itemId) => {

    setCartItems((prev) => ({

      ...prev,

      [itemId]: (prev[itemId] || 0) + 1,

    }));

  };

  const removeFromCart = (itemId) => {

    setCartItems((prev) => {

      if (!prev[itemId] || prev[itemId] === 1) {

        const { [itemId]: \_, ...rest } = prev;

        return rest;

      }

      return { ...prev, [itemId]: prev[itemId] - 1 };

    });

  };

  const getTotalCartAmount = () => {

    let totalAmount = 0;

    for (const item in cartItems) {

      if (cartItems[item] > 0) {

        let itemInfo = food\_list.find((product) => product.\_id === item);

        totalAmount += itemInfo.price \* cartItems[item];

      }

    }

    return totalAmount;

  };

  const contextValue = {

    food\_list,

    cartItems,

    setCartItems,

    addToCart,

    removeFromCart,

    getTotalCartAmount,

  };

  return (

    <StoreContext.Provider value={contextValue}>

      {props.children}

    </StoreContext.Provider>

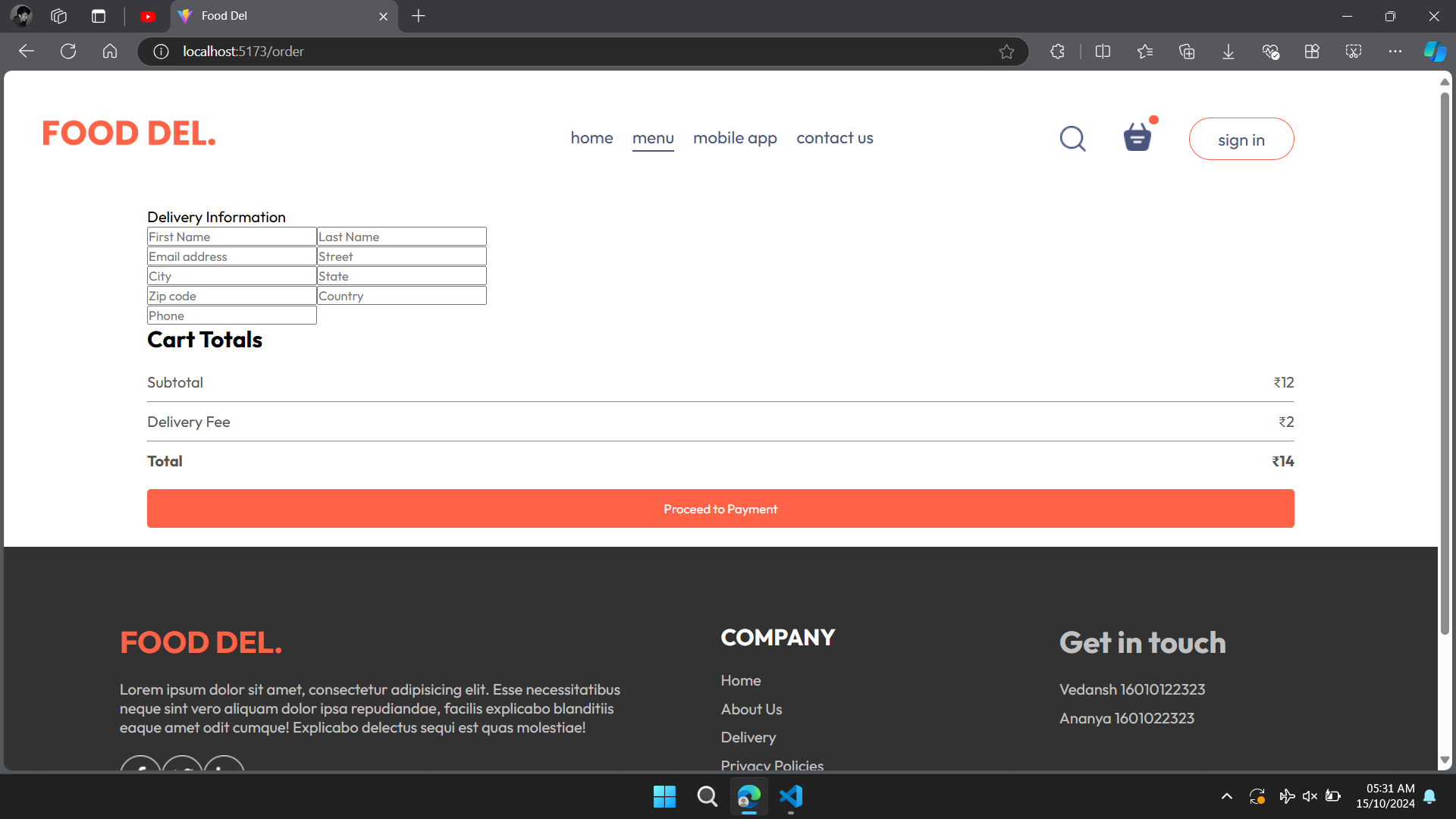
  );

};

export default StoreContextProvider;

change in navbar.jsx to represent dot to cart :

<div className={getTotalCartAmount() === 0 ? "" : "dot"}></div>

Place order :  
  
import React, { useContext } from "react";

import "./PlaceOrder.css";

import { StoreContext } from "../../context/StoreContext";

const PlaceOrder = () => {

  const { getTotalCartAmount } = useContext(StoreContext);

  return (

    <form className="place-order">

      <div className="place-order-left">

        <p className="title">Delivery Information</p>

        <div className="multi-fields">

          <input type="text" placeholder="First Name" />

          <input type="text" placeholder="Last Name" />

        </div>

        <input type="email" placeholder="Email address" />

        <input type="text" placeholder="Street" />

        <div className="multi-fields">

          <input type="text" placeholder="City" />

          <input type="text" placeholder="State" />

        </div>

        <div className="multi-fields">

          <input type="text" placeholder="Zip code" />

          <input type="text" placeholder="Country" />

        </div>

        <input type="text" placeholder="Phone" />

      </div>

      <div className="place-order-right">

        <div className="cart-total">

          <h2>Cart Totals</h2>

          <div>

            <div className="cart-total-details">

              <p>Subtotal</p>

              <p>₹{getTotalCartAmount()}</p>

            </div>

            <hr />

            <div className="cart-total-details">

              <p>Delivery Fee</p>

              <p>₹{2}</p>

            </div>

            <hr />

            <div className="cart-total-details">

              <b>Total</b>

              <b>₹{getTotalCartAmount() + 2}</b>

            </div>

          </div>

          <button className="checkout-button">Proceed to Payment</button>

        </div>

      </div>

    </form>

  );

};

export default PlaceOrder;

place order after styling with code :  
code:

.place-order {

  display: flex;

  align-items: start;

  justify-content: space-between;

  gap: 50px;

  margin-top: 100px;

}

.place-order-left {

  width: 100%;

  max-width: max(30%, 500px);

}

.place-order-left .title {

  font-size: 30px;

  font-weight: 600;

  margin-bottom: 50px;

}

.place-order-left input {

  margin-bottom: 15px;

  padding: 10px;

  width: 100%;

  border: 1px solid #c5c5c5;

  border-radius: 4px;

  outline-color: tomato;

}

.place-order-left .multi-fields {

  display: flex;

  gap: 10px;

}

.place-order-right {

  width: 100%;

  max-width: max(40%, 500px);

}

.place-order .cart-total-button {

  margin-top: 30px;

}

output:

