FOOD COURT

TEAM MEMBERS:

ROLL NUMBER	NAME
22ALR097	SUDHARSHAN S
22ALR099	SUDHEKSHA K
22ALR080	RITHANYA M

INTRODUCTION:

Welcome to Food Court – Your Ultimate Destination for Culinary Delights!.
At Food Court, we believe that food is more than just sustenance; it's an experience that brings people together. Our mission is to provide a seamless and enjoyable dining experience for everyone, whether you're a foodie, a family looking for a great meal, or someone in search of a quick bite. Explore a diverse range of restaurants and cuisines, all in one convenient place. From mouth-watering appetizers to delectable desserts, our curated selection ensures there's something for every palate. With FoodCourt, discovering new flavors and favorite dishes has never been easier.

Key features:

1. Easy Ordering and Payment.
2. User-Friendly Interface.
3. Easy Ordering and Payment.
4. Personalized Recommendations.
5. Real-Time Order Tracking.
6. Customer Reviews and Ratings.\
7. Exclusive Deals and Discounts.
8. Multi-Platform Accessibility.

REQUIREMENTS:

- **Front-End Programming**

    1. **CSS (Cascading Style Sheets):** CSS is used to define the visual appearance and layout of web pages. It allows developers to apply styles, such as colors, fonts, spacing, and positioning, to HTML elements, creating a visually appealing and consistent user interface.

    2. **JavaScript:** JavaScript is a powerful programming language that adds interactivity and dynamic behavior to web pages. It enables developers to manipulate and modify HTML and CSS, handle user interactions, perform calculations, validate forms, and make asynchronous requests to servers.

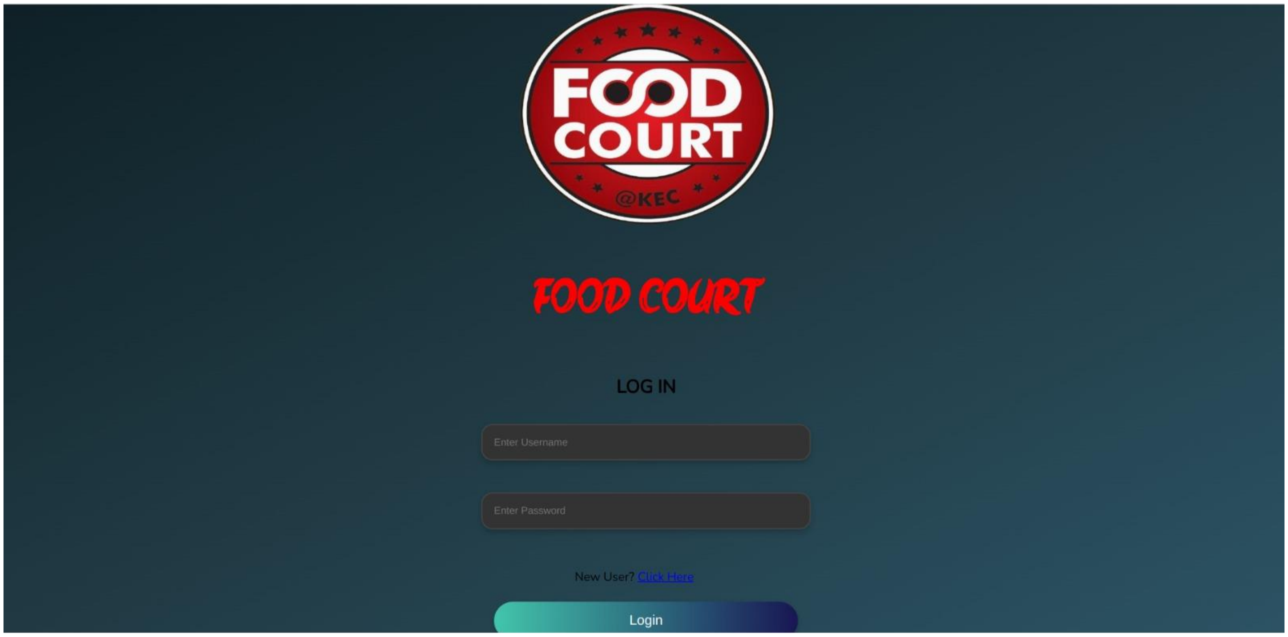
    3. **React:** React is a popular JavaScript library for building user interfaces. It provides a component-based approach to web development, allowing developers to create reusable UI components that update efficiently based on changes in data.React uses a virtual DOM (Document Object Model) to optimize performanceand facilitate the building of complex web applications.

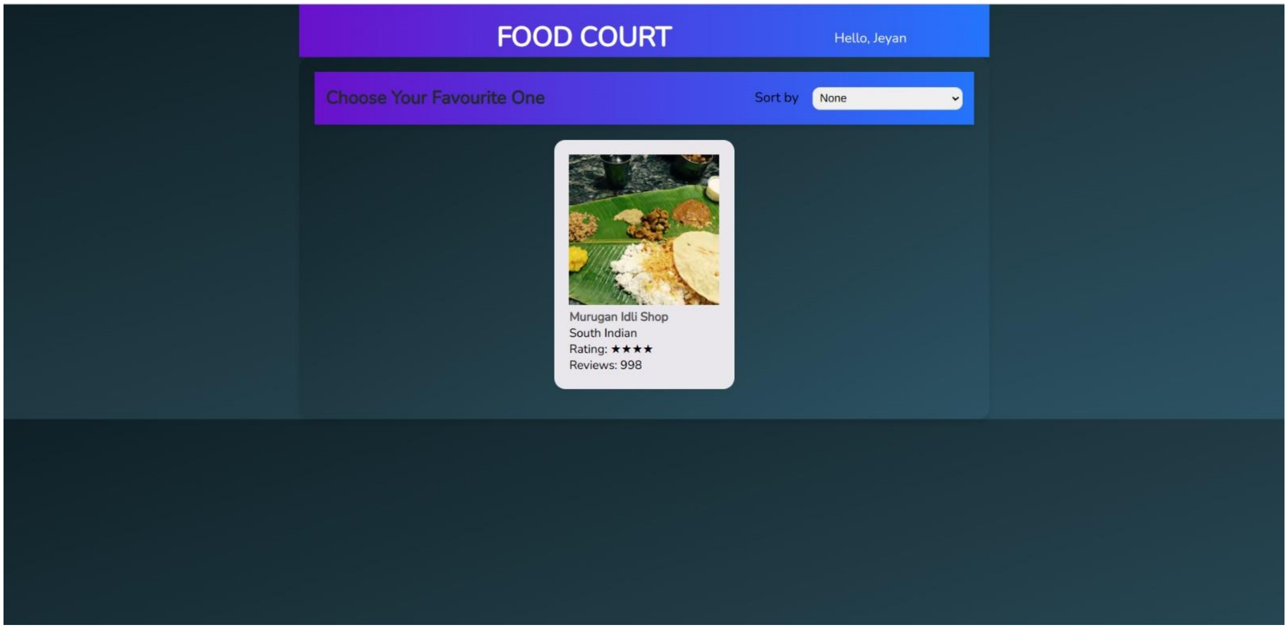
- **Back-End Programming**

    1. **Database Management:** Backend programming involves working with databases to store and retrieve data. Developers interact with databases using query languages like SQL (Structured Query Language) or NoSQL databases. They design and implement database schemas, optimize queries, and ensure data integrity and security.

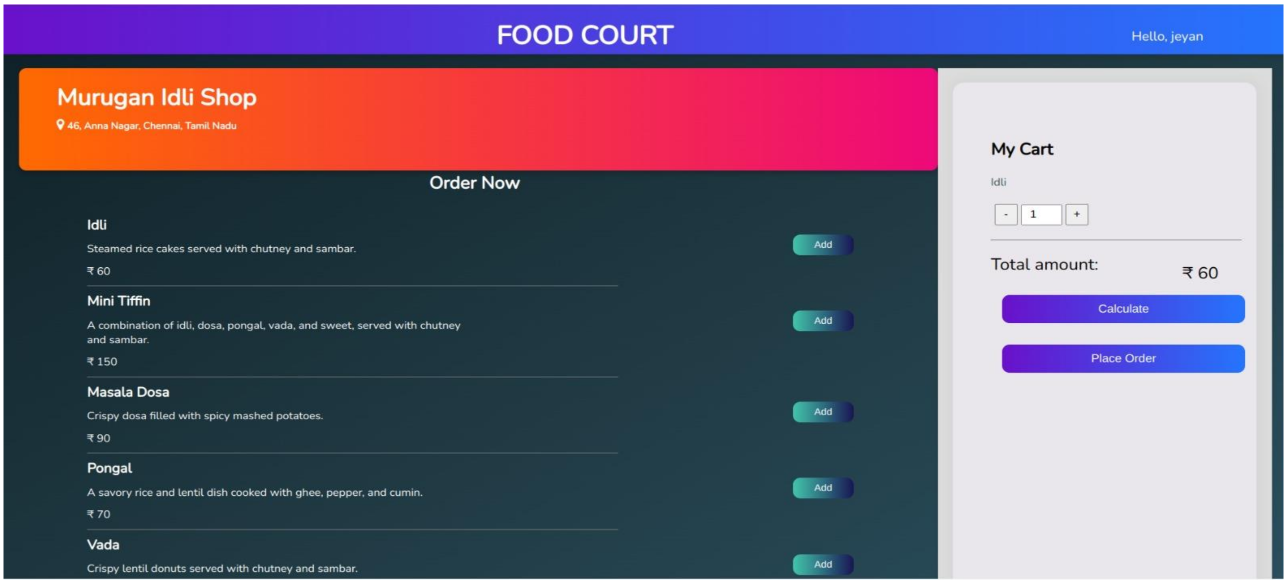
    2. **APIs (Application Programming Interfaces)**: Backend developers build APIs that allow communication between the frontend and backend components of a web application. APIs define the protocols and rules for how different software components can interact and exchange data. Commonly used API standards include RESTful APIs and GraphQL.

UI DESIGN:





Murugan Idli Shop
South Indian
Rating: ★★★★
Reviews: 998



Murugan Idli Shop

46, Anna Nagar, Chennai, Tamil Nadu

Order Now

Idli

Steamed rice cakes served with chutney and sambar.

₹ 60

Mini Tiffin

A combination of idli, dosa, pongal, vada, and sweet, served with chutney and sambar.

₹ 150

Masala Dosa

Crispy dosa filled with spicy mashed potatoes.

₹ 90

Pongal

A savory rice and lentil dish cooked with ghee, pepper, and cumin.

₹ 70

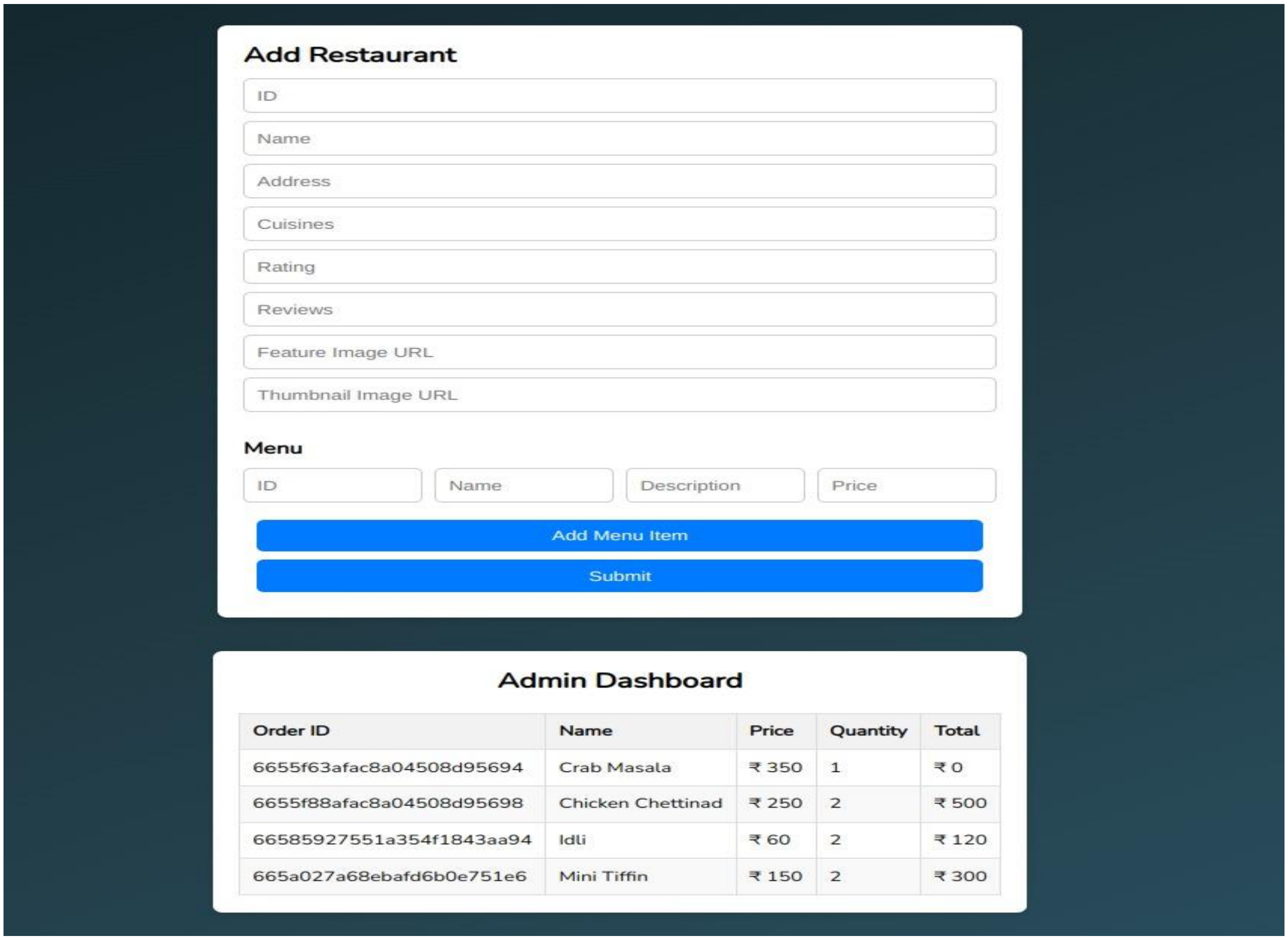
Vada

Crispy lentil donuts served with chutney and sambar.

My Cart

Idli

Total amount: ₹ 60



Add Restaurant

Admin Dashboard

Order ID	Name	Price	Quantity	Total
6655f63afac8a04508d95694	Crab Masala	₹ 350	1	₹ 0
6655f88afac8a04508d95698	Chicken Chettinad	₹ 250	2	₹ 500
66585927551a354f1843aa94	Idli	₹ 60	2	₹ 120
665a027a68ebafd6b0e751e6	Mini Tiffin	₹ 150	2	₹ 300

SAMPLE CODING:

1.SERVER.JS

```
const express = require('express');
const mongoose = require('mongoose');
const bodyParser = require('body-parser');
const cors = require('cors');

const app = express();

app.use(bodyParser.json());
app.use(cors("*"));

mongoose.connect('mongodb://localhost:27017/restaur
    antdb', {
  useNewUrlParser: true,
  useUnifiedTopology: true
});
const db = mongoose.connection;
db.once('open', () => {
  console.log('Connected to MongoDB');
});

const menuItemSchema = new mongoose.Schema({
  id: Number,
  name: String,
  desc: String,
  price: Number
});

const restaurantSchema = new mongoose.Schema({
  id: String,
  name: String,
  address: String,
  cuisines: String,
  rating: String,
  reviews: String,
  feature_image: String,
  thumbnail_image: String,
  menu: [menuItemSchema]
});

const orderSchema = new mongoose.Schema({
  name: String,
  price: Number,
  quantity: Number,
  total: Number,
});

const Restaurant = mongoose.model('Restaurant',
   restaurantSchema);

app.post('/api/restaurants', (req, res) => {
  const newRestaurant = new Restaurant(req.body);
  newRestaurant.save((err, restaurant) => {
    if (err) return res.status(500).send(err);
    res.status(200).send(restaurant);
  });
});

app.get('/api/restaurants', (req, res) => {
  Restaurant.find({}, (err, restaurants) => {
    if (err) return res.status(500).send(err);
    res.status(200).send(restaurants);
  });
});

app.delete('/api/restaurants/:id', (req, res) => {
  Restaurant.deleteOne({ id: req.params.id }, (err) =>
   {
    if (err) return res.status(500).send(err);
    res.status(200).send({ message: 'Restaurant
   deleted successfully' });
  });
});

const Order = mongoose.model('Order', orderSchema);

app.post('/api/orders', async (req, res) => {
  const order = new Order(req.body);
  try {
    await order.save();
    res.status(201).send(order);
  } catch (error) {
    res.status(400).send(error);
  }
});
```

```
app.get('/api/orders', async (req, res) => {
    try {
        const orders = await Order.find();
        res.status(200).send(orders);
    } catch (error) {
        res.status(500).send(error);
    }
});

const PORT = process.env.PORT || 5000;
app.listen(PORT, () => {
    console.log(`Server is running on port ${PORT}`);
      });
```

2. MENU.JS

```
import React from 'react'

class Menu extends React.Component{

    constructor(){
        super();
        this.state = {
            price: '',
            quantity:0
        }
    }

    render(){
        return(
            <div>
                <h3 className='fname'>
                {this.props.name}
                </h3>
                    <div className='desc' 
      style={{marginRight:"500px"}}>
                        <p>{this.props.desc}</p>
                        <button className="btn" 
      value={this.props.price} 
      onClick={()=>this.props.action(this.props.price
      ,this.props.name,this.state.quantity)}>Add</bu
      tton>
                    </div>
                    <br/>
                    <p className='amount'>{'\u20B9'} 
      {this.props.price} </p>
      </div>

    )
  }
}

 export default Menu;
```

3.MYCART.JS

```
import React from 'react'

class MyCart extends React.Component{
  render(){
    return(
      <div>
        {this.props.name.map((name,number) => (
          <p key={number} id="pitem">
            {name}
          <br/><br/>
          <input className="ip" type="button" value="-" onClick={()=>this.props.decrement(this.props.quantity)}/>
          <input className="ip" id="tx-w" type="text" value={this.props.quantity}/>
          <input className="ip" type="button" value="+" onClick={()=>this.props.increment(this.props.quantity)}/>
          </p>
        ))}
      </div>
    )
  }
}

export default MyCart
```

4.RESTAURANTCARD.JS

```
import React from 'react'
import {Link} from 'react-router-dom'
import data from '../data/data.json'
class RestaurantCard extends React.Component{
    constructor(){
        super();
        this.state={
            list: data
        };
    }

    render(){
        return(
            <div className="cart">
                <Link to={{
                    pathname:`/order/${this.props.name}`}}>
                    <center><img src={this.props.thumbnail_image} alt={this.props.name}/></center>
                </Link>
                    <h3 id="hotel-title">{this.props.name}</h3>
                    <p>{this.props.cuisines}</p>
                    <p>Rating: {'★'.repeat(this.props.rating)}</p>
                    <p>Reviews: {this.props.reviews}</p>
            </div>
        );
    }
}

export default RestaurantCard
```

5.TEXTINPUT.JS

```
import React from 'react';

class TextInput extends React.Component {
    render(){
        return(
            <div>
                <input type={this.props.type}
                       id="input"
                       name={this.props.name}
                       placeholder={this.props.placeholder}
                       value={this.props.value}
                       onChange={this.props.onChange}
                       autocomplete="off"/><br/><br/>
            </div>
        )
    };
}

export default TextInput
```

6.HOTELS.JSX

```
import React from 'react';
import '../styles/HotelStyle.css'
import data from '../data/data.json'
import RestaurantCard from '../components/RestaurantCard'
class Hotels extends React.Component {
    constructor(){
        super();
        this.state={
            list: data
        };
    }



    sortMenu=(e)=>{
        if (e.target.value === 'rating'){
            this.setState({
                list: data.sort(function(a,b){return b.rating - a.rating})
            })

        }
    else if (e.target.value === 'review'){
      this.setState({
        list: data.sort(function(a,b){return b.reviews - a.reviews})
      })

    }
    else if (e.target.value === 'name'){
      function compareName  (a, b)  {
        // case-insensitive comparison
        a = a.toLowerCase();
        b = b.toLowerCase();

        return (a < b) ? -1 : (a > b) ? 1 : 0;
       }
      this.setState({
        list: data.sort(function(a,b){return compareName(a.name, b.name)})
      })

    }

  }

  render(){
    return(
    <div>
      <div className="nav" style={{backgroundColor: "gray"}}>
        <div id="logo">
          <h2>FOOD COURT</h2>
        </div>

        <div id="user">
          <div className="name">Hello, Jeyan</div>
          <div className="profile">
          </div>
        </div>
      </div>

      <div className="maincart">

      <div id="menubar">
        <h2 id="menu-title">Choose Your Favourite One</h2>
        <p id ="sort">  Sort by &nbsp; &nbsp;
```

```
                    <select id="sort-metrics" defaultValue={"none"} onChange={(e) =>
this.sortMenu(e)}>
                        <option value="none" disabled hidden>None</option>
                        <option class="sort-option" value="name">Name</option>
                        <option class="sort-option" value="rating">Ratings</option>
                        <option class="sort-option" value="review">Reviews</option>
                    </select>
                </p>
            </div>
            {this.state.list.map(
                x =>
                    <RestaurantCard thumbnail_image={x.thumbnail_image} name = {x.name}
cuisines = {x.cuisines} rating = {x.rating} reviews = {x.reviews}/>


            )}




            </div>
        </div>
        )
    };
}

export default Hotels;
```

7.LOGINPAGE.JSX

```
import React from 'react';
import '../styles/LoginStyle.css'
import Front from '../images/FB.png'
import TextInput from '../components/TextInput'
class LoginPage extends React.Component{
    constructor(){
        super();
        this.state={
            username:'',
            password:''
        };
    }

    onChangeUser=(event)=>{
        this.setState({username:event.target.value})
    }

    onChangePass=(event)=>{
        this.setState({password:event.target.value})
    }

    click=()=>{
        if(this.state.username === 'jeyan' && this.state.password === 'jeyan123'){
            this.props.history.push(`/hotels`);
        }
        else if(this.state.username === 'admin' && this.state.password === 'admin123'){
            this.props.history.push(`/admin`);
        }
        else{
            window.alert("Invalid Username or Password. Try Again!")
        }
    }


    render(){
        return(
            <div>
                <meta name="viewport" content="width=device-width, initial-scale=1.0"/>

            <div id = "container">
            <div id="id1">
            <img src={Front} height="300" width="350" alt="Banner"/>
         </div>

         <div id="id2">
            <center><h1 id="logoApp">FOOD COURT</h1></center>
               <div id="id3">

               <center><h2>LOG IN</h2><br/>
               { this.state.clickable &&
               <div>Username: {this.state.username} <br/>
               Password: {this.state.password}</div>}
               <form action="#" id="form">
                  <TextInput type="text" name="username" placeholder="Enter Username"
onChange={(e)=>{this.setState({username: e.target.value})}} />
                  <TextInput type="password" name="password" placeholder="Enter
Password"  onChange={(e)=>{this.setState({password: e.target.value})}}/>

               </form>
               </center>
               <br/>
               <span style={{marginLeft:'123px'}}>New User? <a href="#">Click
Here</a></span>
               <br/><br/>
               <center>

                     <input type="submit" name="signin" id="btn" value="Login"
onClick={this.click} />

               </center>




            </div>
         </div>
      </div>
      </div>
      )
   };
}

export default LoginPage;
```

8.ORDERS.JSX

```
import React from 'react';
import '../styles/OrderStyle.css';
import Menu from '../components/Menu';
import data from '../data/data.json';
import MyCart from '../components/MyCart';
import axios from 'axios';

class Orders extends React.Component {
    constructor() {
        super();
        this.state = {
            list: data,
            id: '',
            newList: [],
            price: '',
            name: [],
            total: 0,
            quantity: 0,
            clickable: false
        };
    }

    componentWillMount() {
        const hotel = this.props.history.location.pathname.slice(7);
        const List = this.state.list.filter((rec) => rec.name === hotel);
        this.setState({
            id: hotel,
            newList: List
        });
    }

    childHandler = (ChildPrice, ChildName, ChildQuantity) => {
        const names = 
this.state.name;
    names.push(ChildName);
    this.setState({
      price: ChildPrice,
      name: names,
      quantity: ChildQuantity + 1,
      clickable: true
    });
  };

  incrementQuantity = (incQuan) => {
    this.setState({
      quantity: incQuan + 1
    });
  };

  decrementQuantity = (decQuan) => {
    if (this.state.quantity >= 1) {
      this.setState({
        quantity: decQuan - 1
      });
    }
  };

  total = (p, q) => {
    this.setState({
      total: p * q
    });
  };

  handlePlaceOrder = () => {
    const order = {
      name: this.state.name.join(', '),
      price: this.state.price,
      quantity: 
```

```
this.state.quantity,
      total: this.state.total
    };

    axios.post('http://localhost:5000/api/orders', order)
      .then(response => {
        alert('Order placed successfully');
      })
      .catch(error => {
        console.error('There was an error placing the order!', error);
      });
  };

  render() {
    return (
      <div>
        <div className="nav">
          <div id="logo">
            <h2>FOOD COURT</h2>
          </div>

          <div id="user">
            <div className="name">Hello, jeyan</div>
            <div className="profile"></div>
          </div>
        </div>

        <div id="content">
          <div id="head">
            <h1
```

```
className='hname'>{this.state.newList.map(x => x.name)}</h1>
            <h5 className='aname'><i className="fa fa-map-marker" style={{ fontSize: 18 }}></i> {this.state.newList.map(x => x.address)}</h5>
            <div id='items'>
              <center><h2>Order Now</h2></center>
              <br />
              {this.state.newList.map(
                x => x.menu.map(item => <Menu
                  id={item.id}
                  desc={item.desc}
                  price={item.price}
                  name={item.name}
                  action={this.childHandler} />)
              )}
            </div>
          </div>
          <div id="panel">
            <div id="logo"></div>
            <div id="right">
              <div id="right-in">
                <h4>My Cart</h4>
                {this.state.clickable &&
                  <div>
```

```
                        <MyCart
                          name={this.state.name}
                          price={this.state.price}
                          quantity={this.state.quantity}
                          increment={this.incrementQuantity}
                          decrement={this.decrementQuantity}>
                        </MyCart>
                    </div>
                }
                <div id="total">
                    <p id="total" style={{ marginRight: "100px" }}> Total amount:
                        <span className="spn">{'\u20B9'} {this.state.total}</span>
                    </p>
                    <input id="pay" type="button" value="Calculate"
                        onClick={() => this.total(this.state.price, this.state.quantity)} />
                    <br />
                    <input id="pay" type="button" value="Place Order" onClick={this.handlePlaceOrder} />
                </div>
```

```
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        );
    }
}

export default Orders;
```

9.RESTAURANTDASHBOARD.JSX

```
import React, { useState, useEffect } from 'react';
import axios from 'axios';
import "../styles/admin.css";

const AdminDashboard = () => {
    const [orders, setOrders] = useState([]);

    useEffect(() => {
        axios.get('http://localhost:5000/api/orders')
            .then(response => {
                setOrders(response.data);
            })
            .catch(error => {
                console.error('There was an error fetching the orders!', error);
            });
    }, []);

    return (
        <div className="dashboard-container">
        <h2>Admin Dashboard</h2>
        <table className="order-table">
            <thead>
                <tr>
                    <th>Order ID</th>
                    <th>Name</th>
                    <th>Price</th>
                    <th>Quantity</th>
                    <th>Total</th>
                </tr>
            </thead>
            <tbody>
                {orders.map(order => (
                    <tr key={order._id}>
                        <td>{order._id}</td>
                        <td>{order.name}</td>
                        <td>{'\u20B9'} {order.price}</td>
                        <td>{order.quantity}</td>
                        <td>{'\u20B9'} {order.total}</td>
                    </tr>
        ))}
      </tbody>
    </table>
  </div>
  );
};

export default AdminDashboard;
```

10.RESTAURANTFORM.JSX

```
import React, { useState } from 'react';
import axios from 'axios';
import "../styles/admin.css";


const RestaurantForm = () => {
  const [formData, setFormData] = useState({
    id: '',
    name: '',
    address: '',
    cuisines: '',
    rating: '',
    reviews: '',
    feature_image: '',
    thumbnail_image: '',
    menu: [{ id: '', name: '', desc: '', price: '' }]
  });

  const handleChange = (e) => {
    const { name, value } = e.target;
    setFormData({ ...formData, [name]: value });
  };

  const handleMenuChange = (index, e) => {
    const { name, value } = e.target;
    const newMenu = formData.menu.slice();
    newMenu[index][name] = value;
    setFormData({ ...formData, menu: newMenu });
  };

  const addMenuItem = () => {
    setFormData({ ...formData, menu: [...formData.menu, { id: '', name: '', desc: '', price: '' }] });
  };

  const handleSubmit = (e) => {
    e.preventDefault();
    axios.post('http://localhost:5000/api/restaurants', formData)
      .then(response => {
        console.log(response.data);
      })
      .catch(error => {
        console.error(error);
      });
  };
```

```
    return (
        <form onSubmit={handleSubmit} className="form-container">
            <h2>Add Restaurant</h2>
            <input type="text" name="id" placeholder="ID" value={formData.id}
onChange={handleChange} required className="form-input" />
            <input type="text" name="name" placeholder="Name" value={formData.name}
onChange={handleChange} required className="form-input" />
            <input type="text" name="address" placeholder="Address" value={formData.address}
onChange={handleChange} required className="form-input" />
            <input type="text" name="cuisines" placeholder="Cuisines" value={formData.cuisines}
onChange={handleChange} required className="form-input" />
            <input type="text" name="rating" placeholder="Rating" value={formData.rating}
onChange={handleChange} required className="form-input" />
            <input type="text" name="reviews" placeholder="Reviews" value={formData.reviews}
onChange={handleChange} required className="form-input" />
            <input type="text" name="feature_image" placeholder="Feature Image URL"
value={formData.feature_image} onChange={handleChange} required className="form-input" />
            <input type="text" name="thumbnail_image" placeholder="Thumbnail Image URL"
value={formData.thumbnail_image} onChange={handleChange} required className="form-input"
/>

            <h3 className="form-title">Menu</h3>
            {formData.menu.map((item, index) => (
                <div key={index} className="menu-item">
                    <input type="text" name="id" placeholder="ID" value={item.id} onChange={(e) =>
handleMenuChange(index, e)} required className="form-input menu-item-input" />
                    <input type="text" name="name" placeholder="Name" value={item.name}
onChange={(e) => handleMenuChange(index, e)} required className="form-input menu-item-
input" />
                    <input type="text" name="desc" placeholder="Description" value={item.desc}
onChange={(e) => handleMenuChange(index, e)} required className="form-input menu-item-
input" />
                    <input type="text" name="price" placeholder="Price" value={item.price}
onChange={(e) => handleMenuChange(index, e)} required className="form-input menu-item-
input" />
                </div>
            ))}
            <button type="button" onClick={addMenuItem} className="form-button">Add Menu
Item</button>
            <button type="submit" className="form-button">Submit</button>
        </form>
    );
};

export default RestaurantForm;
```

11.APP.JS

```
import React from 'react';
import LoginPage from './pages/LoginPage'
import Hotels from './pages/Hotels'
import Orders from './pages/Orders'
import {BrowserRouter as Router, Switch, Route} from 'react-router-dom';
import Admin from './Admin';

function App() {
  return (
    <Router>
      <div className="App">
      <Switch>
        <Route exact path="/" component={LoginPage}></Route>
        <Route exact path="/hotels" component={Hotels}></Route>
        <Route exact path="/order/:id" component={Orders}></Route>
        <Route exact path="/admin" component={Admin}></Route>
        </Switch>
      </div>
    </Router>
  );
}

export default App;
```

CONCLUSION:

At FoodCourt, we are dedicated to transforming your dining experiences into memorable culinary journeys. Our platform brings together a diverse selection of cuisines, user-friendly features, and top-notch services to cater to all your dining needs. Whether you're exploring new flavors, looking for reliable reviews, or seeking convenient ordering and delivery options, FoodCourt is your go-to solution.

Join our community of food lovers and discover the joys of hassle-free dining. With personalized recommendations, exclusive deals, and real-time order tracking, FoodCourt ensures that every meal is a delightful experience. Trust in our secure and private platform, and enjoy the benefits of our loyalty program and exceptional customer support.

Savor the best, embrace variety, and let FoodCourt be your ultimate dining companion. Bon appétit!

.