# Sri Lanka Institute of Information Technology

## Distributed Systems (SE3020)

## Assignment 02 - Assignment Report

## Online Train Ticket Reservation System

Submitted By: IT17096744 (N.J.Godewithana)

# TABLE OF CONTENTS

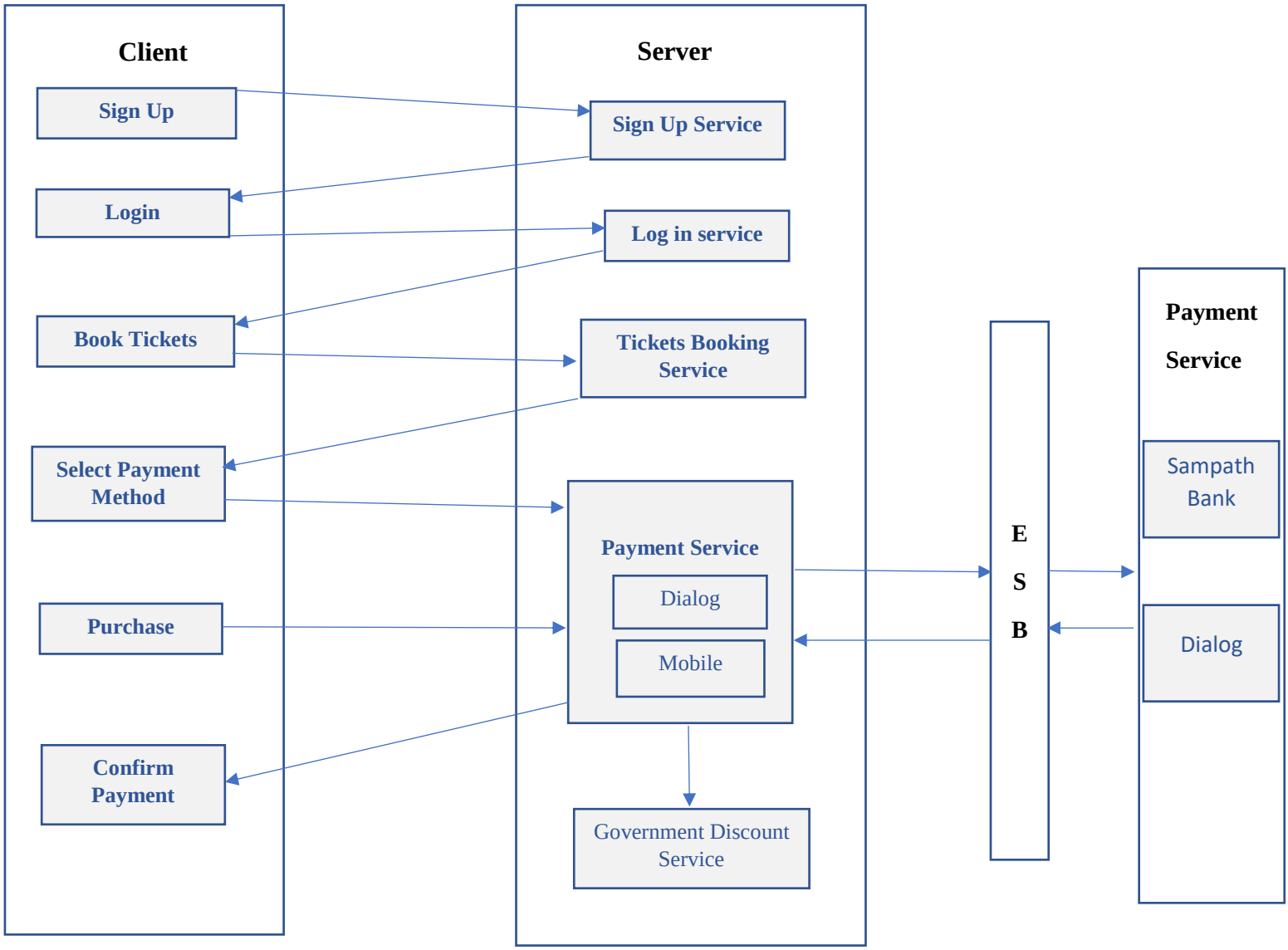
# 1. Introduction

This ‘Online Train Ticket Booking’ is a web application that can be used for the reservation of the train tickets, developed using set of RESTful web services. Customer can firstly log in to the system or sign up. Then the available tickets can be reserved. After that the payments can be done by using Dialog bill payment service or Sampath bank card payments. The government employees can get discounts when ordering train tickets. The government employee status can be verified using a government dept’s service that verifies whether a person is a government employee, given the NIC no.

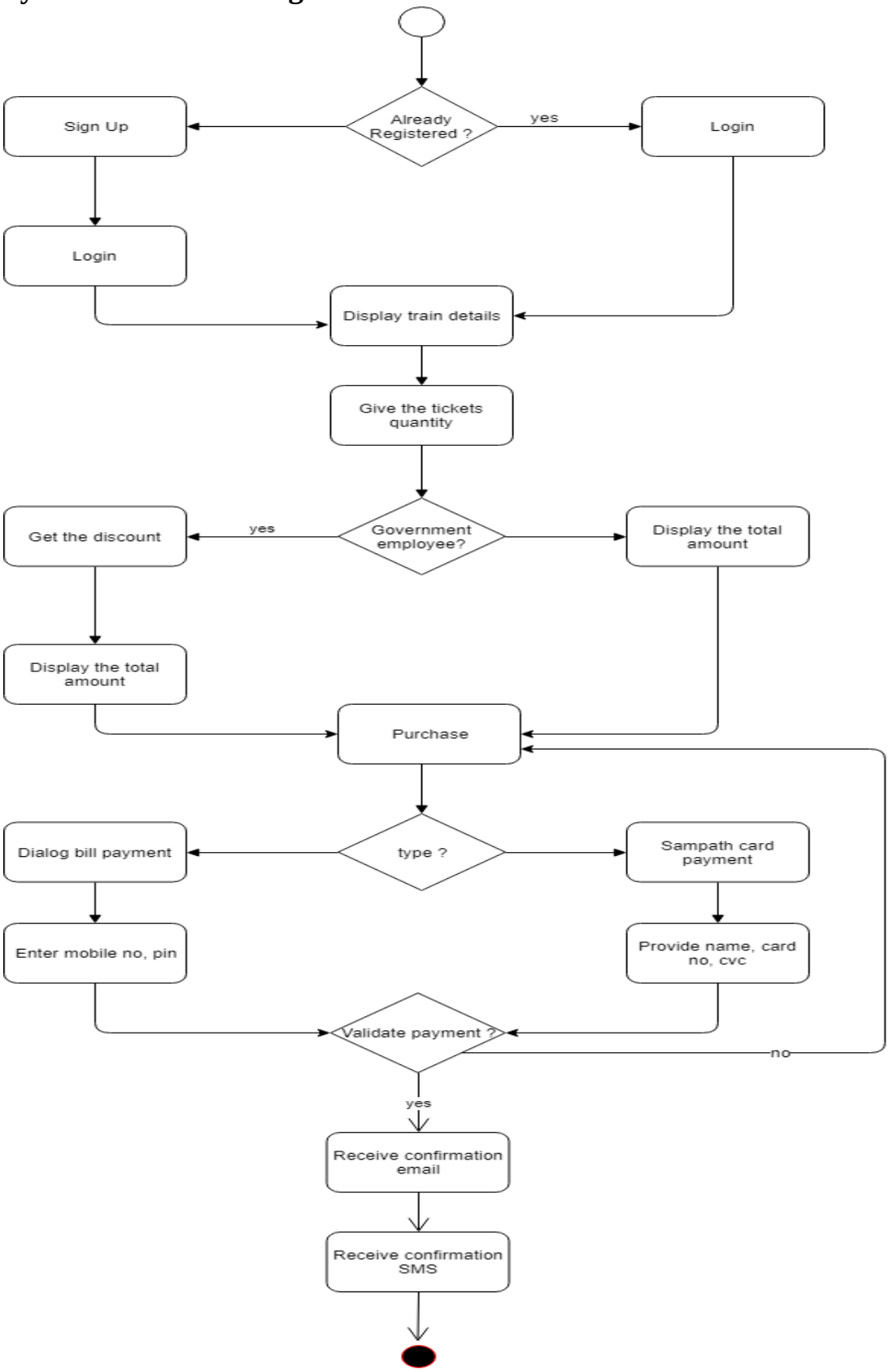
This application is developed using **Service Oriented Architecture**. This application is a combination of Front-End(Website) and Backend(Rest API). And WSO2 EI project part also accompanied.

Front-End is developed by using React Js and Bootstrap. Node and Express Js used for the Back-End development. Mongodb is used as the database. The web application sends automatic emails with the help of Nodemailer module as well as It sends SMS using Nexmo. WSO2 EI (Enterprise Integration – ESB) used to route the payment to either the payment gateway or the mobile operator (Dialog).

## 2. Architectural Diagram



# 3. System Workflow Diagram



# 4. System Workflows Explanation

1. Customer can sign up to the system

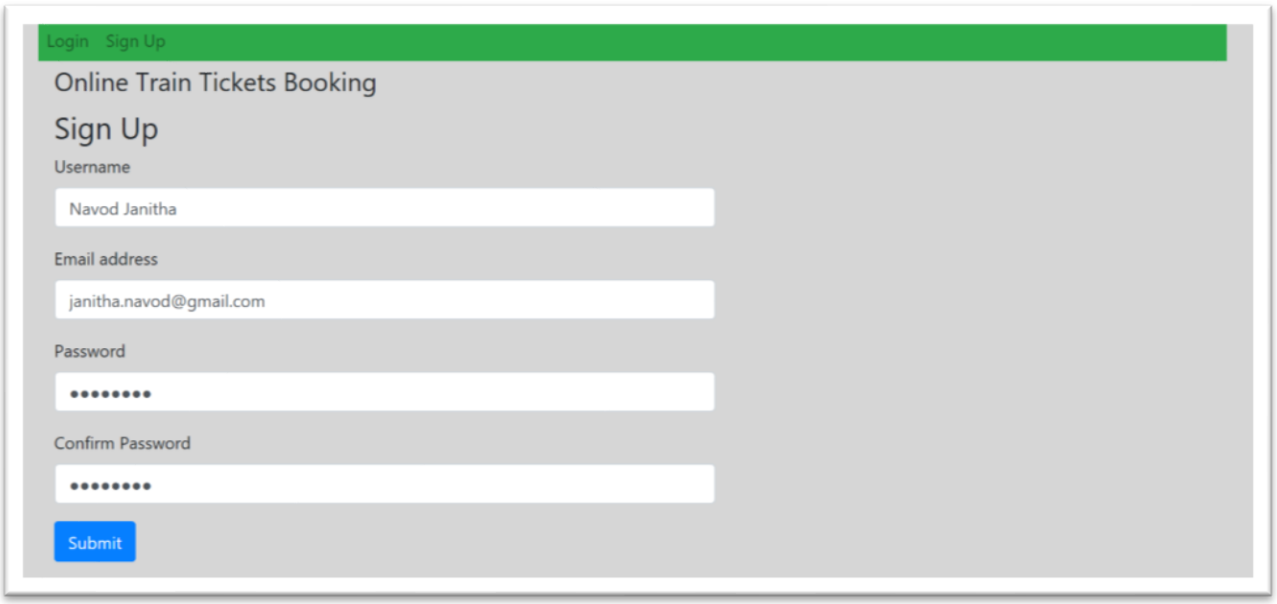



*Figure 1*

2. Once you signup successfully , Signup Confirmation email will be received.

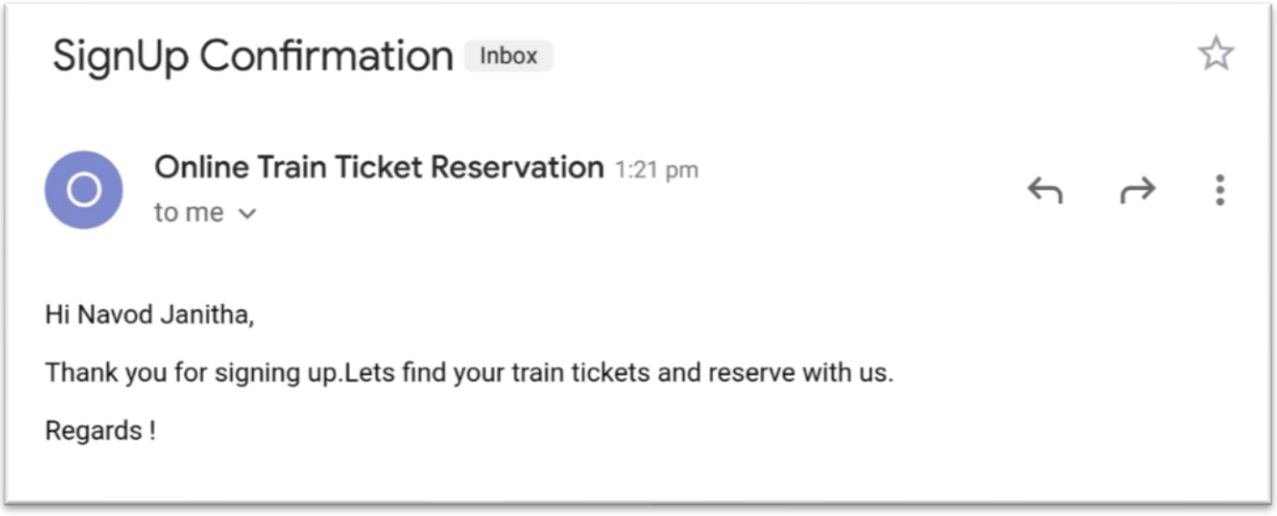



*Figure 2*

3. Then the customer can log into the system.


*Figure 3*

4. Then all the available train tickets are displayed on the home page.


*Figure 4*

5. Customer can reserve the tickets by clicking on the ‘Book’ field and filling the relevant details.
    - If a normal user,

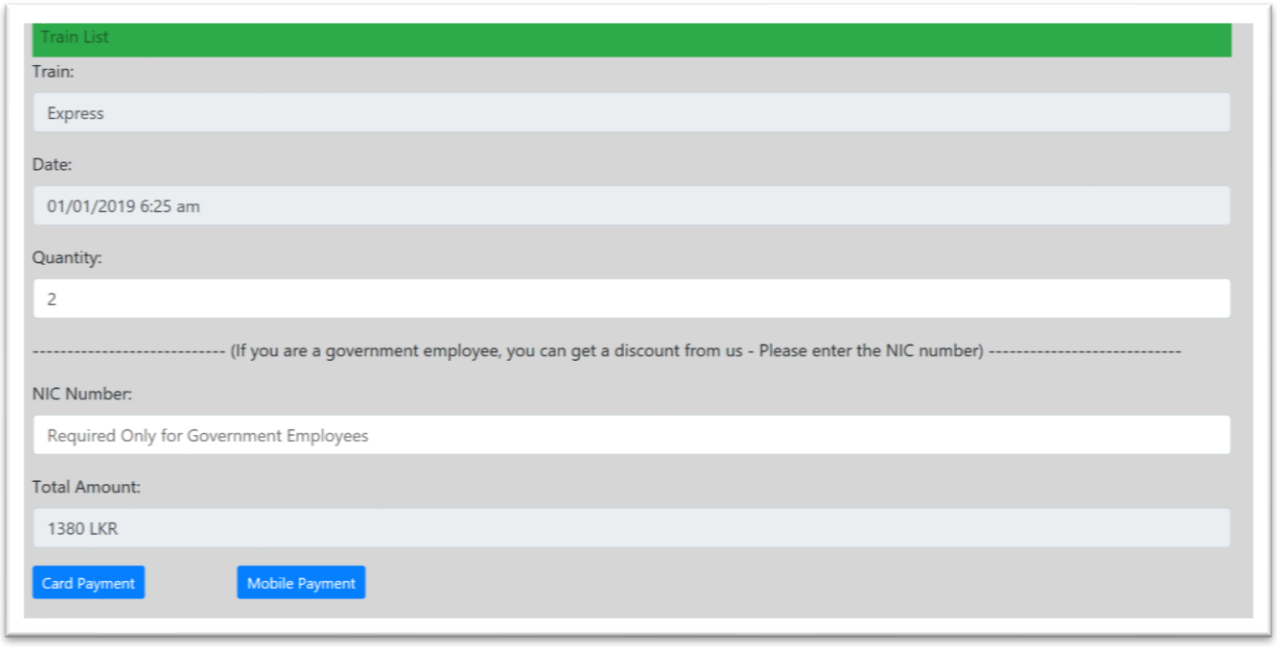



*Figure 5*

- If a government user, can get a discount by providing the NIC

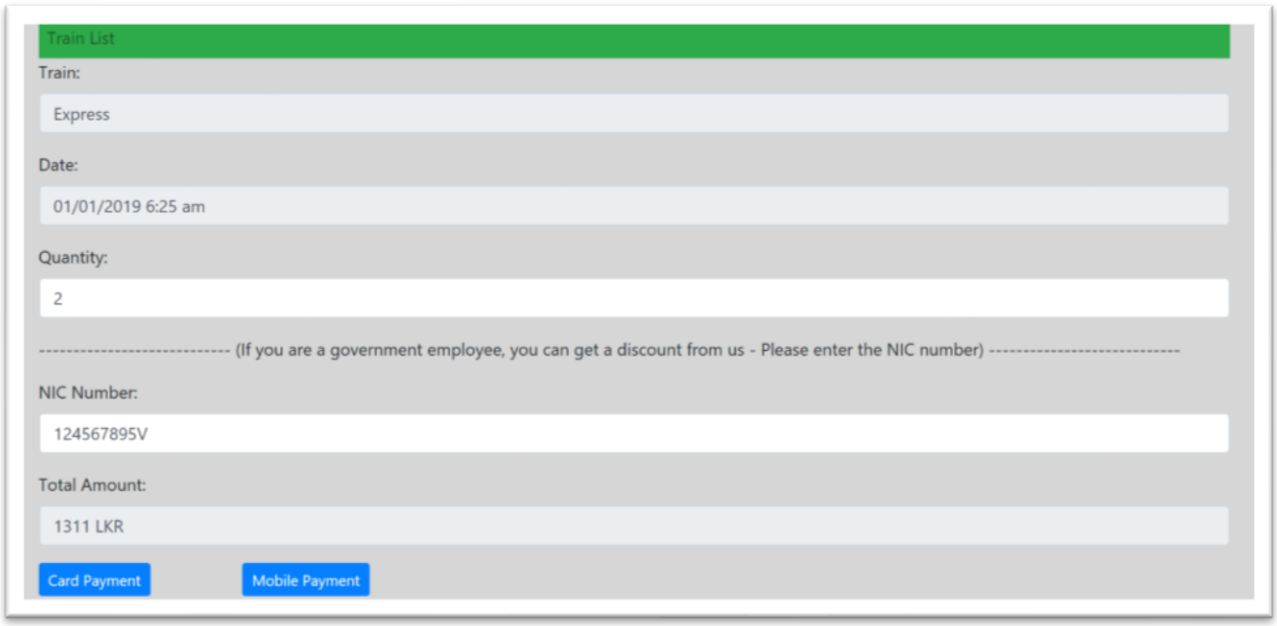



*Figure 6*

6. User can select the payment method and do the purchasing.
    - If user select the card payment option, After filling the following details, the tickets can be reserved.

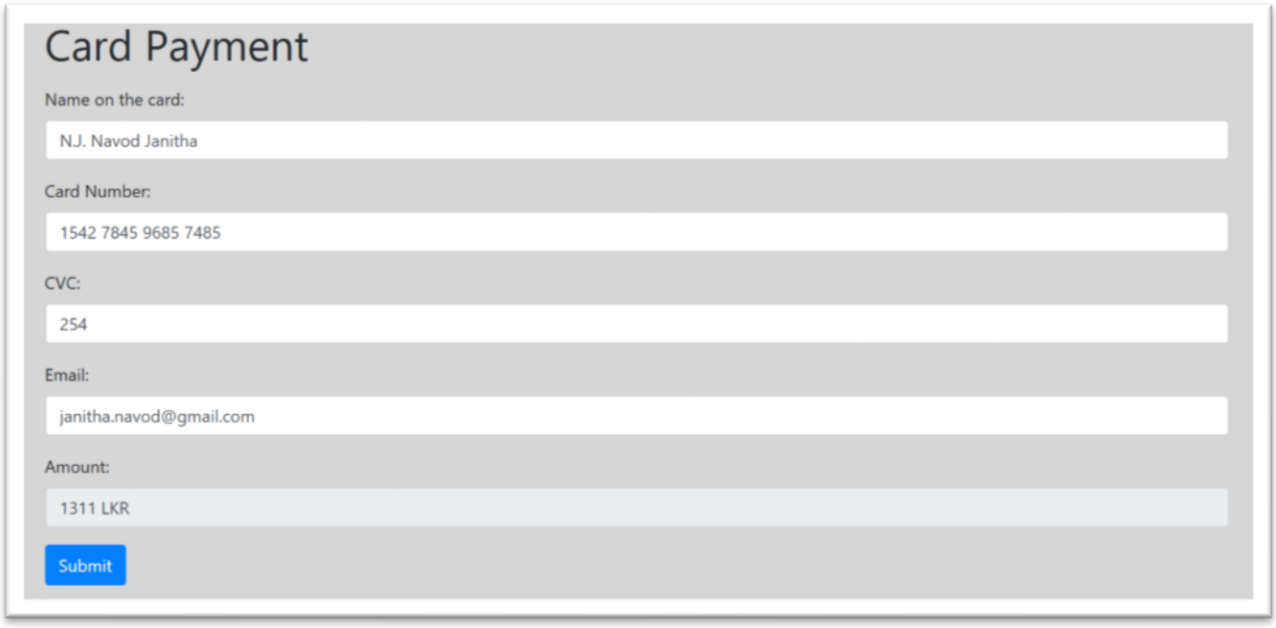



*Figure 7*

Then the customer will get the Payment Confirmation email

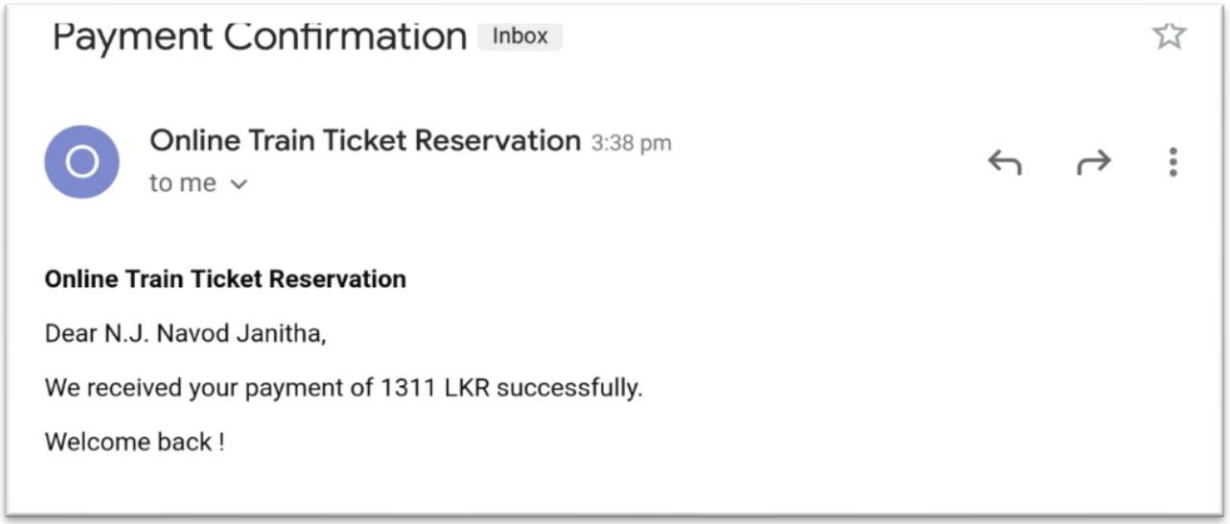



*Figure 8*

- If user select the mobile payment option, After filling the following details can reserve the ticket.

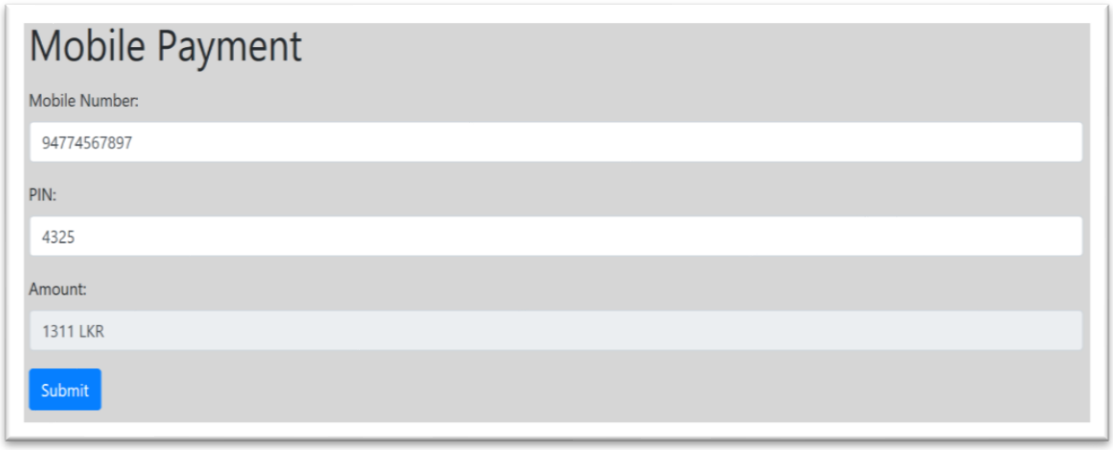



*Figure 9*

Then the customer will get an SMS to the provided mobile number.

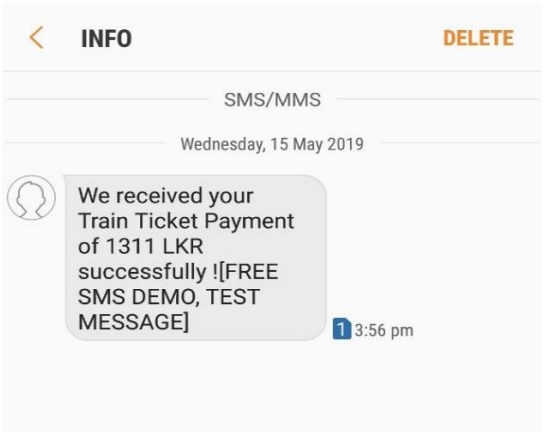



*Figure 10*

## 5. Authentication and Authorization

When the user sign up to the system, he or she will not be able to sign up if the provided email is already used.

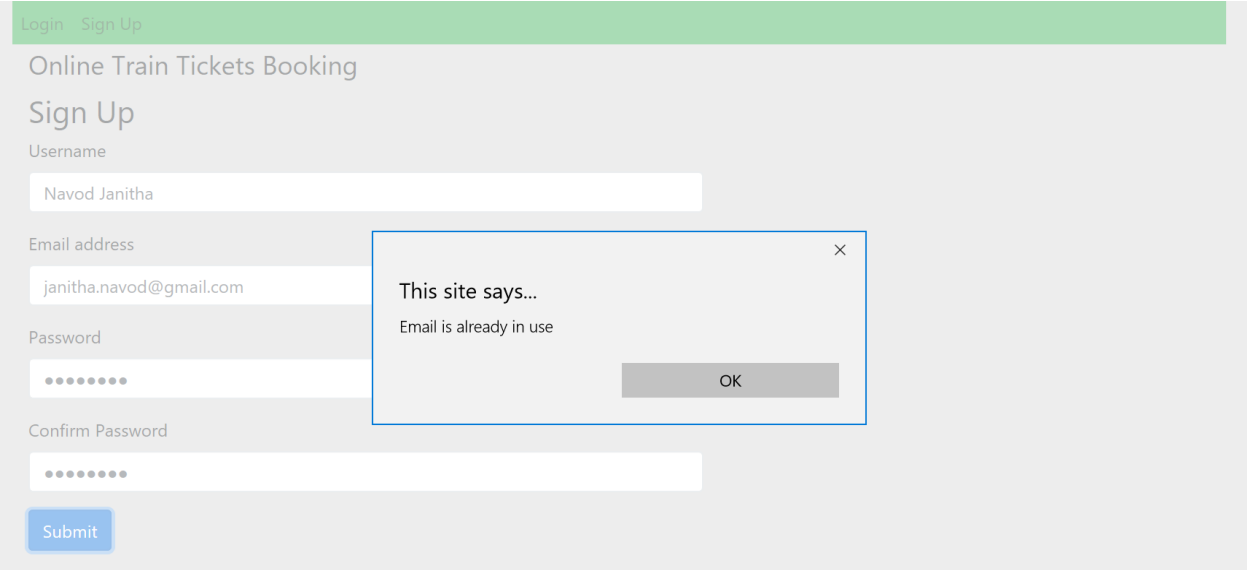



*Figure 11*

If the user try to log into the system using an invalid email or password, following alert will be displayed.

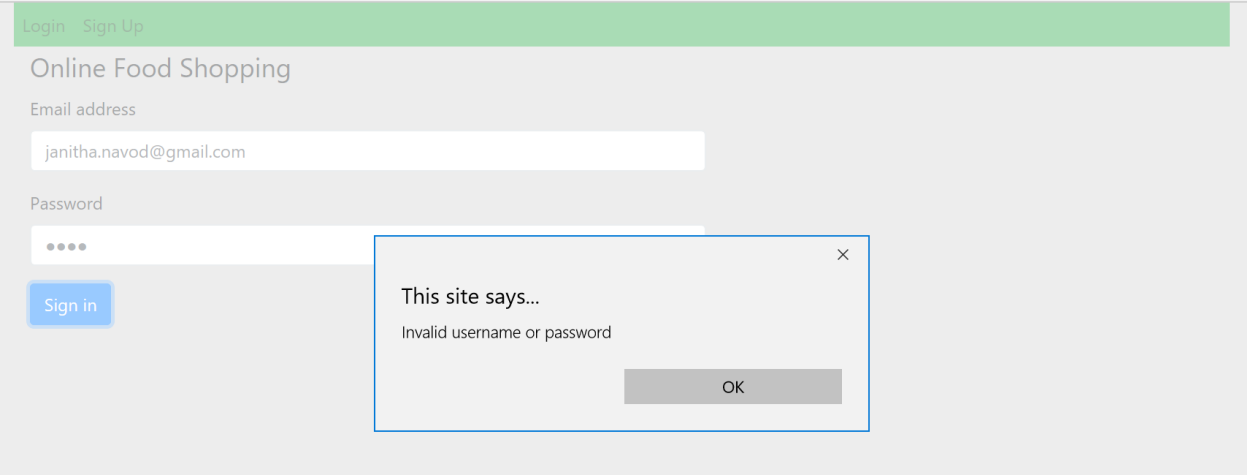



*Figure 12*

# 6. Appendix

## Dropbox Link

https://www.dropbox.com/s/86r6ntqcku4svnk/IT17096744.zip?dl=0

## 6.1 Frontend – Web Client

- Sign Up component

```
import React,{Component} from 'react';
import axios from 'axios';
import ReactDOM from "react-dom";
import TrainList from "./trainList";
import Login from "./login";

class Signup extends Component {
    constructor(props) {
        super(props);
    }

    /* *This method is called wwhen the submit button in the signup page is
clicked, this method used to register a new user if the * email addresse entered is
not already in use */

    signup(){
        const uname = this.refs.uname.value;
        const email = this.refs.email.value;
        const password = this.refs.password.value;
        const cpassword = this.refs.cpassword.value;

        if(uname===''||email===''||password===''||cpassword===''){
            alert('One or more fields empty');
        }

        if(password !== cpassword){
            alert('Password is incorrect.Please enter again')
        }
        else {
            var foundEmail = false;

            fetch('http://localhost:4001/demo/getUser/' + email, {
                method: 'GET',
                headers: {'Content-Type': 'application/json'}
            }).then(response => {
                return response.json();
            }).then(data => {
                var user = JSON.stringify(data);
                console.log(user);

                if (user != '[]') {
                    alert("Email is already in use");
                } else {
                    var data = {"email": email,"uname": uname,  "password":
password};
                    console.log(data);
                    fetch('http://localhost:4001/demo/addUser', {
                        method: 'POST',
```

```
                        body: JSON.stringify(data),
                        headers: {'Content-Type': 'application/json'}
                    }).then(response => {
                        return response.json();
                    }).then(data => {
                        alert('Successfully Sign Up');
                        ReactDOM.render(<Login/>, document.getElementById('root'));
                    }).catch(err => {
                        alert("Second" + err);
                    })
                }
            }).catch(err => {
                alert("First Err:" + err);
            })
        }

    }
render() {

        return(


            <div className="container">
                <div className="backimg">
                    <div className="paddinglog">
                    </div>

                    <div className="row">
                        <div className="col-md-7">
                            <form className="paddingsub">
                                <fieldset>
                                    <legend>Online Train Tickets Booking</legend>
                                    <h3>Sign Up</h3>

                                    <div className="form-group">
                                        <label
htmlFor="exampleInputEmail1">Username</label>
                                        <input className="form-control"
id="exampleInputEmail1"
                                               aria-describedby="emailHelp"
placeholder="Username" type="text"
                                               ref="uname"/>
                                    </div>

                                    <div className="form-group">
                                        <label htmlFor="exampleInputEmail1">Email
address</label>
                                        <input className="form-control"
id="exampleInputEmail1" aria-describedby="emailHelp" placeholder="Email"
type="email" ref="email"/>
                                    </div>

                                    <div className="form-group">
                                    <label
htmlFor="exampleInputPassword1">Password</label>
                                    <input className="form-control"
id="exampleInputPassword1" placeholder="Password" type="password" ref="password"/>
                                    </div>

                                    <div className="form-group">
                                        <label
htmlFor="exampleInputPassword1">Confirm Password</label>
                                        <input className="form-control"
```

```
    id="exampleInputPassword1" placeholder="Password" type="password" ref="cpassword"/>
                                            </div>


                                            <button type="button" className="btn btn-
    primary" onClick={()=>{this.signup()}}>Submit</button>
                                        </fieldset>
                                    </form>
                                </div>

                            </div>
                        </div>
                    </div>

            )
    }

    }

     export default Signup;
```

- Login component

```
import React, {Component} from 'react';
import ReactDOM from 'react-dom';
import App from '../App';
import TrainList from './trainList';
import MobilePayment from'./mobilePayment';
import AllRoutes from'./allRoutes';
import { breakStatement } from "@babel/types";

class Login extends Component{
constructor(props) {
    super(props);
    this.onChangeEmail = this.onChangeEmail.bind(this);
    this.onChangePassword = this.onChangePassword.bind(this);

    this.state={
        name: '',
        totalPrice:'',
        email: '',
        password: ''
    }

}


onChangeEmail(e){
    this.setState({
        email: e.target.email
    });
}

onChangePassword(e){
    this.setState({
        password: e.target.password
    });
}

        myFunc = function(e){
            const email = this.state.email;
```

```
            const password = this.state.password;

            //validation for the fields
            if(this.state.email == '' && this.state.password == ''){
                alert('Please enter the email and password');
                breakStatement(this);
            }
            if (this.state.email == '') {
                alert('Please enter your email');
                breakStatement(this);
            }
            if (this.state.password == ''){
                alert('Please enter the password');
                breakStatement(this);
            }
            else{
                var credentials = {"email": email, "password": password};
                fetch('http://localhost:3001/user/' + credentials.email + '/' +
credentials.password, {
                    method: 'GET',
                    headers: {'Content-Type': 'application/json'}
                }).then(response => {
                    return response.json();
                }).then(data => {
                    var user = JSON.stringify(data);
                    if (user != '[]'){
                        console.log(user);
                        // If the user identify correctly, navigate to the home page
                        ReactDOM.render(<AllRoutes  name={this.state.name} />,
document.getElementById('root'));
                    } else{
                        alert("Invalid username or password");
                    }

                }).catch(err => {
                    alert(err);
                })
            }

        }



    render() {
        var tot = 0;
        return(


            <div className="container">
                <div className="backimg">
                    <div className="paddinglog">

                    </div>
                    <div className="row">
                        <div className="col-md-7">
                            <form className="paddingsub">
                                <fieldset>
                                    <legend>Online Food Shopping</legend>

                                    <div className="form-group">
                                        <label htmlFor="exampleInputEmail1">Email
address</label>
```

```
                                        <input className="form-control"
                                               id="exampleInputEmail1"
                                               aria-describedby="emailHelp"
                                               placeholder="Enter email"
                                               type="email"
                                               ref="email"
                                               value={this.state.email}
                                               onChange={this.onChangeEmail}/></div>

                                    <div className="form-group"><label
htmlFor="exampleInputPassword1">Password</label>
                                        <input className="form-control"
                                               id="exampleInputPassword1"
                                               placeholder="Password"
                                               type="password"
                                               ref="password"
                                               value={this.state.password}

onChange={this.onChangePassword}/></div>

                                    <button type="submit" className="btn btn-primary"
                                            onClick={() => this.myFunc(tot)}>Sign
in</button>
                                </fieldset>
                            </form>
                        </div>

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

export default Login;
```

- TrainList component – Home Page

```
import React,{Component} from 'react';
import {Link} from 'react-router-dom';
import axios from 'axios';

const Train = props => (
    <tr>
        <td>{props.demo.name}</td>
        <td>{props.demo.date}</td>
        <td>{props.demo.from}</td>
        <td>{props.demo.to}</td>
        <td>{props.demo.ticketPrice}</td>
        <td>
            <Link to={"/reserve/"+props.demo._id}>Book</Link>
        </td>
    </tr>
)

export default class TrainList extends Component{

    constructor(props){
        super(props);
        this.state = {demo: []};
    }
```

```
    componentDidMount() {
        axios.get('http://localhost:4001/demo/getAllTrains').then(res => {
            this.setState({
                demo: res.data
            });
            console.log(res.data);
        }).catch(function (err) {
            console.log(err);
        });

    }

    trainList(){
        return this.state.demo.map(function (currentTrain, i) {
            return <Train demo={currentTrain} key={i}/>
        });
    }

    render(){
        return(
            <div>

                <table className="table table-striped " style={{ marginTop: 20}}>
                    <thead>
                    <tr>
                        <th>Train</th>
                        <th>Date / Departure Time</th>
                        <th>From</th>
                        <th>To</th>
                        <th>Ticket Price</th>
                    </tr>
                    </thead>
                    <tbody>
                    { this.trainList() }
                    </tbody>
                </table>

            </div>
        )
    }
}
```

- CreateReservation component

```
import React,{Component} from 'react';
import axios from 'axios';
import CardPayment from "./cardPayment";
import MobilePayment from "./mobilePayment";
import ReactDOM from "react-dom";


export default class CreateReservation extends Component{

    constructor(props){
        super(props);

        this.onChangeTrainName = this.onChangeTrainName.bind();
        this.onChangeDate = this.onChangeDate.bind();
        this.onChangeQuantity = this.onChangeQuantity.bind(this);
        this.onChangeNIC = this.onChangeNIC.bind(this);
```

```
        this.state={
            name: '',
            date: '',
            quantity: '',
            ticketPrice: '',
            totalPrice:'',
            nic:''
        }

    }


    onChangeTrainName(e) {
        this.setState({
            name: e.target.value
        });
    }
    onChangeDate(e) {
        this.setState({
            date: e.target.value
        });
    }

    onChangeQuantity(e) {
        this.setState({
            quantity: e.target.value
        });
    }

    onChangeNIC(e) {
        this.setState({
            nic: e.target.value
        });
    }

    // this will get all the train details from the database
    componentDidMount() {

axios.get('http://localhost:4001/demo/getTrain/'+this.props.match.params.id)
            .then(response => {
                this.setState({
                    name: response.data.name,
                    date: response.data.date,
                    ticketPrice: response.data.ticketPrice
                })
            })
            .catch(function (error) {
                console.log(error);
            })
    }

    // calculate the total amount
    calculateTotalPrice(qty, price){
       var quantity = parseInt(qty);
       var ticketPrice = parseInt(price);

       if (qty == 0){
           return null;
       }

       this.state.totalPrice = quantity * ticketPrice;
      // console.log(quantity);
```

```
        return quantity * ticketPrice;
    }

    // If a government user, this method will calculate the discount and return the
total amount to be paid
    discount(qty,price,nic){
        var quantity = parseInt(qty);
        var ticketPrice = parseInt(price);
        if (nic == null){
            return this.state.totalPrice = ticketPrice* quantity;
        }
        if (qty == 0){
            return '';
        }
        if (nic.length == 10){
            return  this.state.totalPrice = (ticketPrice-((5/100)*ticketPrice)) *
quantity + ' LKR';
        }
        else{
            return this.state.totalPrice = ticketPrice* quantity+ ' LKR';
        }
    }

    // If user select the card payment option, this will render the card payment
component
    getCardPay = function(tot){
        if (this.state.totalPrice != ''){
            ReactDOM.render(<CardPayment total={tot} name={this.state.name}
totalPrice={this.state.totalPrice}/>, document.getElementById('root'));
        }
    }

    // If user select the mobile payment option, this will render the mobile
payment option
    getMobilePay = function(tot){
        if (this.state.totalPrice != ''){
            ReactDOM.render(<MobilePayment total={tot} name={this.state.name}
totalPrice={this.state.totalPrice}/>, document.getElementById('root'));
        }
    }
    render(){
        var tot = 0;
        const qty = this.state.quantity;
        const price = this.state.ticketPrice;
        var nic = this.state.nic;
        return(
            <div style={{margineTop: 20}}>

                <form onSubmit={this.onSubmit}>
                    <div className="form-group">
                        <label>Train: </label>
                        <input type="text"
                               className="form-control"
                                value={this.state.name}
                                readOnly = {true}/>
                    </div>

                    <div className="form-group">
                        <label>Date: </label>
                        <input type="text"
                               className="form-control"
                               value={this.state.date}
                                readOnly={true}/>
```

```
                    </div>

                    <div className="form-group">
                        <label>Quantity:</label>
                        <input type="text"
                               className="form-control"
                               value={this.state.quantity}
                               onChange={this.onChangeQuantity}/>
                    </div>

                    <div className="form-group">
                        <p>----------------------------    (If you are a government
employee, you can get a discount from us - Please enter the NIC number)    --------
--------------------</p>
                    </div>

                    <div className="form-group">
                        <label>NIC Number: </label>
                        <input type="text"
                               className="form-control"
                               placeholder="Required Only for Government Employees"
                               value={this.state.nic}
                               onChange={this.onChangeNIC}/>
                    </div>

                    <div className="form-group">
                        <label>Total Amount: </label>
                        <input type="text"
                               className="form-control"
                               value={this.discount(qty, price, nic)}
                               readOnly={true}/>
                    </div>

                    <div className="row">
                        <div className="col-md-2">
                        <button type="button" className="btn btn-primary btn-sm"
onClick={() => this.getCardPay(tot)}>Card Payment</button>
                        </div>
                        <div className="col-md-5">
                        <button type="button" className="btn btn-primary btn-sm"
onClick={() => this.getMobilePay(tot)}>Mobile Payment</button>
                        </div>
                    </div>
                </form>
            </div>
        );
    }
}
```

- CardPayment component

```
import React,{Component} from 'react';
import axios from 'axios';
import { Link } from "react-router-dom";
import CreateReservation from "./createTicketReservation";

export default class CardPayment extends  Component{
    constructor(props){
        super(props);

        this.onChangeHolderName = this.onChangeHolderName.bind(this);
        this.onChangeMobileNumber = this.onChangeMobileNumber.bind(this);
        this.onChangeCardNumber = this.onChangeCardNumber.bind(this);
        this.onChangeCvc = this.onChangeCvc.bind(this);
        this.onSubmit = this.onSubmit.bind(this);

        this.state = {
            name: '',
            mobileNumber: '',
            cardNumber: '',
            cvc: '',
            total: this.props.totalPrice
        }
    }

    onChangeHolderName(e){
        this.setState({
            name: e.target.value
        });
    }

    onChangeMobileNumber(e){
        this.setState({
            mobileNumber: e.target.value
        });
    }

    onChangeCardNumber(e){
        this.setState({
            cardNumber: e.target.value
        });
    }

    onChangeCvc(e){
        this.setState({
            cvc: e.target.value
        });
    }


    // this method will submit the card payment details
    onSubmit(e){
        e.preventDefault();
```

```
        console.log('Form Submitted:');
        console.log('name: ${this.state.name}');
        console.log('mobileNumber : ${this.state.mobileNumber}');
        console.log('cardNumber: ${this.state.cardNumber}');
        console.log('cvc: ${this.state.cvc}');
        console.log('total: ${this.state.total}');

        const newPayment = {
            name: this.state.name,
            mobileNumber: this.state.mobileNumber,
            cardNumber: this.state.cardNumber,
            cvc: this.state.cvc,
            total: this.state.total
        }

        axios.post('http://localhost:4001/demo/addCardPayment', newPayment)
            .then(res => console.log(res.data));

        this.setState({
            name: '',
            mobileNumber: '',
            cardNumber: '',
            cvc: '',
            total: this.props.totalPrice
        })

    }



    render() {
        return(
        <div style={{margineTop: 50}}>
            <div className="container">
            <form onSubmit={this.onSubmit}>

                <div className="form-group">
                    <h1>Card Payment</h1>
                </div>

                <div className="form-group">
                    <label>Name on the card:</label>
                    <input type="text"
                    className="form-control"
                    value={this.state.name}
                    onChange={this.onChangeHolderName}/>
                </div>

                <div className="form-group">
                    <label>Card Number:</label>
                    <input type="text"
                           className="form-control"
                           value={this.state.cardNumber}
                           onChange={this.onChangeCardNumber}/>
                </div>

                <div className="form-group">
                    <label>CVC:</label>
                    <input type="text"
                           className="form-control"
                           value={this.state.cvc}
                           onChange={this.onChangeCvc}/>
                </div>
```

```
                <div className="form-group">
                    <label>Email:</label>
                    <input type="text"
                           className="form-control"
                           value={this.state.mobileNumber}
                           onChange={this.onChangeMobileNumber}/>
                </div>

                <div>
                    <label>Amount:</label>
                    <input  type="text"
                            className="form-control"
                            value={this.state.total}
                            readOnly={true}/>
                </div>


                <div>
                <p></p>
                </div>

                <div className="form-group">
                    <input type="submit" value="Submit" className="btn btn-primary"/>
                </div>

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

- MobilePayment component

```
import React,{Component} from 'react';
import axios from 'axios';

export default class MobilePayment extends Component{
    constructor(props){
        super(props);

        this.onChangeMobileNumber = this.onChangeMobileNumber.bind(this);
        this.onChangePin = this.onChangePin.bind(this);
        this.onChangeAmount = this.onChangeAmount.bind(this);
        this.onSubmit = this.onSubmit.bind(this);

        this.state={
            mobileNumber: '',
            pin: '',
            amount: this.props.totalPrice
        }
    }

    onChangeMobileNumber(e){
        this.setState({
            mobileNumber: e.target.value
        });
    }
    onChangePin(e){
        this.setState({
            pin: e.target.value
```

```
        });
    }
    onChangeAmount(e){
        this.setState({
            amount: e.target.value
        });
    }

    // this method will submit the payment details , payment will be added to the
user's monthly dialog mobile bill
    onSubmit(e){
        e.preventDefault();

        console.log('Form Submitted:');
        console.log('Mobile Number: ${this.state.mobileNumber}');
        console.log('Pin: ${this.state.pin}');
        console.log('Amount: ${this.state.amount}');

        const newMobilePayment = {
            mobileNumber: this.state.mobileNumber,
            pin: this.state.pin,
            amount: this.state.amount
        }

        axios.post('http://localhost:4001/demo/addMobilePayment', newMobilePayment)
            .then(res => console.log(res.data));


        this.setState({
            mobileNumber: '',
            pin: '',
            amount: ''
        })
    }



    render() {
        return(
            <div style={{margineTop: 20}}>
                <div className="container">
                <form onSubmit={this.onSubmit}>

                    <div className="form-group">
                        <h1>Mobile Payment</h1>
                    </div>
                    <div className="form-group">
                        <label>Mobile Number:</label>
                        <input type="text"
                               placeholder="Ex : 94715465897"
                               className="form-control"
                               value={this.state.mobileNumber}
                               onChange={this.onChangeMobileNumber}/>
                    </div>


                    <div className="form-group">
                        <label>PIN:</label>
                        <input type="text"
                               className="form-control"
                               value={this.state.pin}
                               onChange={this.onChangePin}/>
```

```
                        </div>

                        <div className="form-group">
                            <label>Amount:</label>
                            <input type="text"
                                   className="form-control"
                                   value={this.state.amount}
                                   onChange={this.onChangeAmount}
                                   readOnly={true}/>
                        </div>

                        <div>
                            <p></p>
                        </div>

                        <div className="form-group">
                            <input type="submit" value="Submit" className="btn btn-
primary"/>
                        </div>

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

- AllRoutes component

```
import React, {Component} from 'react';
import { BrowserRouter as Router, Route, Link} from "react-router-dom";
import "bootstrap/dist/css/bootstrap.min.css";
import TrainList from "./trainList";
import CreateReservation from "./createTicketReservation";
import CardPayment from ".//cardPayment";
import MobilePayment from "./mobilePayment";
import Signup from "./signUp";
import Login from "./login";

class AllRoutes extends Component {

    constructor(props){
        super(props);

        this.state={
            name: this.props.name
        }
    }
    render() {
        return (
            <Router>
                <div className="container">
                    <nav className="nav navbar-expand-lg navbar-light bg-success">

                        <div className="collapse navbar-collapse">
                            <ul className="navbar-nav mr-auto">
                                <li className="navbar-item">
                                    <Link to="/" className="nav-link">Train
List</Link>
                                </li>
                            </ul>
```

```
                        </div>

                    </nav>
                    <Route path="/" exact component={TrainList}/>
                    <Route path="/reserve/:id" component={CreateReservation}/>
                    <Route path="/doCardPayment" component={CardPayment}/>
                    <Route path="/doMobilePaymnt" component={MobilePayment}/>
                </div>
            </Router>
        );
    }
}
export default AllRoutes;
```

## ➢ App component

```
import React, {Component} from 'react';
import { BrowserRouter as Router, Route, Link} from "react-router-dom";
import "bootstrap/dist/css/bootstrap.min.css";
import logo from './logo.svg';
import './App.css';
import TrainList from "./components/trainList";
import CreateReservation from "./components/createTicketReservation";
import CardPayment from "./components/cardPayment";
import MobilePayment from "./components/mobilePayment";
import Signup from "./components/signUp";
import Login from "./components/login";

class App extends Component {

    constructor(props){
        super(props);

        this.state={
            name: this.props.name
        }
    }
  render() {
    return (
        <Router>
            <div className="container">
                <nav className="nav navbar-expand-lg navbar-light bg-success">

                    <div className="collapse navbar-collapse">
                        <ul className="navbar-nav mr-auto">

                            <li className="navbar-item">
                                <Link to="/" className="nav-link">Login</Link>
                            </li>

                            <li className="navbar-item">
                                <Link to="/signup" className="nav-link">Sign Up</Link>
                            </li>


                        </ul>
                    </div>

                </nav>
                <Route path="/" exact component={Login}/>
```

```
                <Route path="/signup" component={Signup}/>
                <Route path="/reserve/:id" component={CreateReservation}/>
                <Route path="/doCardPayment" component={CardPayment}/>
                <Route path="/doMobilePaymnt" component={MobilePayment}/>


            </div>
        </Router>
    );
  }
}
export default App;
```

## 6.2 Backend – Web Services

### ➢ DBSchema.js

```
const mongoose = require('mongoose');
const Schema = mongoose.Schema;

var Train = new Schema({
    id: {
        type: String
    },
    name: {
        type: String
    },
    date: {
        type: String
    },
    from: {
        type: String
    },
    to: {
        type: String
    },
    ticketPrice: {
        type: String
    }
})

var CreditCard = new Schema({
    name: {
        type: String
    },
    mobileNumber: {
        type: String
    },
    cardNumber: {
        type: String
    },
    cvc: {
        type: String
    },
    total: {
        type: String
    }
})
```

```
var MobilePayment = new Schema({
    mobileNumber: {
        type: String
    },
    pin: {
        type: String
    },
    amount: {
        type: String
    }
})

var User = new Schema({
    email:{
        type: String
    },
    uname: {
        type: String
    },
    password: {
        type: String
    }
})


mongoose.model('Train', Train);
mongoose.model('CreditCard', CreditCard);
mongoose.model('MobilePayment', MobilePayment);
mongoose.model('User', User);

module.exports = mongoose;
```

## 6.2.2 Controllers

- userController.js

```
const express = require('express');
let mongoose = require('./DBSchema');
var userSchema = mongoose.model('User');
const nodemailer = require('nodemailer');

const addUser = function (req,res,next) {
    let user = new userSchema(req.body);

    user.save().then(user =>{
        res.status(200).json({
            'user': 'user added successfully'
        })
    }).catch(err=>{
        res.status(404).send('adding failed');
    });

    var output =
        ` <p> Hi ${req.body.uname}, </p>
          <p>Thank you for signing up.Lets find your train tickets and reserve with
us.</p>
          <p></p>
          <p>Regards !</p>`;
```

```
    let transporter = nodemailer.createTransport({
        service: 'gmail',
        secure: false,
        port: 25,
        auth: {
            user:'demojayaratne@gmail.com',
            pass: 'demo12396#@'
        },
        tls: {
            rejectUnauthorized:false
        }
    });

    let mailOptions = {
        from: '"Online Train Ticket Reservation" <demojayaratne@gmail.com>',
        to: req.body.email,
        subject: 'SignUp Confirmation',
        text: 'Hello',
        html: output
    };

    // customer will get the sign up confirmation email
    transporter.sendMail(mailOptions, (error, info) =>{
        if(error){
            return console.log(error);
        }
        console.log('Message sent: %s', info.messageId);
        console.log('Preview URL: %s', nodemailer.getTestMessageUrl(info));
    });
};

const getUser = function(req,res){
    let email = req.params.email;
    let password = req.params.password;

    userSchema.find({email: email, password: password}, function (err, userSchema) {
        res.json(userSchema);
    });
};

const getUserEmail = function(req,res){
    let email = req.params.email;

    userSchema.find({email: email}, function (err, userSchema) {
        res.json(userSchema);
    });
};

module.exports = {
    addUser,
    getUser,
    getUserEmail
}
```

- trainController.js

```
const express = require('express');
const todoRoutes = express.Router();
let mongoose = require('./DBSchema');
var TrainSchema = mongoose.model('Train');

const addTrain = function (req,res,next) {
    let train = new TrainSchema(req.body);

    train.save().then(train =>{
        res.status(200).json({
            'train': 'train added successfully'
        })
    }).catch(err=>{
        res.status(404).send('adding failed');
    });
};

const getAllTrains = function(req,res,next){

    TrainSchema.find(function (err, trainSchema) {
        if (err){
            console.log(err);
        } else{
            res.json(trainSchema);
        }
    });

};

const getTrain = function(req,res){
    let id = req.params._id;

    TrainSchema.findById(id, function (err, trainSchema) {
        res.json(trainSchema);
    });
};

module.exports = {
    addTrain,
    getAllTrains,
    getTrain
}
```

- paymentController.js

```
const express = require('express');
let mongoose = require('./DBSchema');
var creditCardSchema = mongoose.model('CreditCard');
var mobilePaymentSchema = mongoose.model('MobilePayment');
const nodemailer = require('nodemailer');
const Nexmo = require('nexmo');

const addCardPayment = function (req,res,next) {
```

```
    let card = new creditCardSchema(req.body);

    card.save().then(card =>{
        res.status(200).json({
            'card': 'payment added successfully'
        })
    }).catch(err=>{
        res.status(404).send('adding failed');
    })

    var output =
        ` <b>Online Train Ticket Reservation</b>
          <p> Dear ${req.body.name}, </p>
          <p>We received your payment of ${req.body.total} successfully.</p>
          <p>Welcome back !</p>`;


    let transporter = nodemailer.createTransport({
        service: 'gmail',
        secure: false,
        port: 25,
        auth: {
            user:'demojayaratne@gmail.com',
            pass: 'demo12396#@'
        },
        tls: {
            rejectUnauthorized:false
        }
    });

    let mailOptions = {
        from: '"Online Train Ticket Reservation" <demojayaratne@gmail.com>',
        to: req.body.mobileNumber,
        subject: 'Payment Confirmation',
        text: 'Hello',
        html: output
    };



    // This methos will send the payment confirmation email to the customer
    transporter.sendMail(mailOptions, (error, info) =>{
        if(error){
            return console.log(error);
        }
        console.log('Message sent: %s', info.messageId);
        console.log('Preview URL: %s', nodemailer.getTestMessageUrl(info));
    });

};

const addMobilePayment = function (req,res,next) {
    let mobile = new mobilePaymentSchema(req.body);

    mobile.save().then(mobile =>{
        res.status(200).json({
            'mobile': 'mobile payment added successfully'
        })
    }).catch(err=>{
        res.status(404).send('adding failed');
    });

    const Nexmo = require('nexmo');
```

```
    const nexmo = new Nexmo({
        apiKey: 'febe4ff9',
        apiSecret: '3ymd49KlBsrmy2i2'
    })

    const from = 'Nexmo'
    const to = req.body.mobileNumber
    const text = 'We received your Train Ticket Payment of ' + req.body.amount +  '
successfully ! '
    nexmo.message.sendSms(from, to, text)
}

module.exports = {
    addCardPayment,
    addMobilePayment
}
```

## 6.2.3 Routes

- userRoutes.js

```
const express = require('express');
const userRoutes = express.Router();
const userController = require('./userController');

userRoutes.route('/addUser')
    .post(userController.addUser);

userRoutes.route('/getUser/:email/:password')
    .get(userController.getUser);

userRoutes.route('/getUser/:email')
    .get(userController.getUserEmail);

module.exports = userRoutes;
```

- trainRoutes.js

```
const express = require('express');
const trainRoutes = express.Router();
const trainController = require('./trainController');

trainRoutes.route('/addTrain')
    .post(trainController.addTrain);

trainRoutes.route('/getAllTrains')
    .get(trainController.getAllTrains);

trainRoutes.route('/getTrain/:_id')
    .get(trainController.getTrain);
module.exports = trainRoutes;
```

- paymentRoutes.js

```
const express = require('express');
const paymentRoutes = express.Router();
const paymentController = require('./paymentController');

paymentRoutes.route('/addCardPayment')
    .post(paymentController.addCardPayment);

paymentRoutes.route('/addMobilePayment')
    .post(paymentController.addMobilePayment);

module.exports = paymentRoutes;
```

## Server

- server.js

```
const express = require('express');
const app = express();
const bodyParser = require('body-parser');
const cors = require('cors');
const mongoose = require('mongoose');
const PORT = 4001;
const componentRoutes = express.Router();
let Todo = require('./DBSchema');

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

// using the all routes
const trainRouteHandler = require('./trainRoutes');
const cardRouteHandler = require('./paymentRoutes');
const userRouteHandler = require('./userRoutes');

// connect to the database
mongoose.connect('mongodb://127.0.0.1:27017/demo', {useNewUrlParser: true});

// establish the connection
const connection = mongoose.connection;

connection.once('open', function () {
    console.log("MongoDB database connection established succesfully !!!");
});

app.use('/demo', trainRouteHandler);
app.use('/demo', cardRouteHandler);
app.use('/demo', userRouteHandler);

// listen to the port
app.listen(PORT, function () {
    console.log("Server is running on Port: " + PORT);
});
```