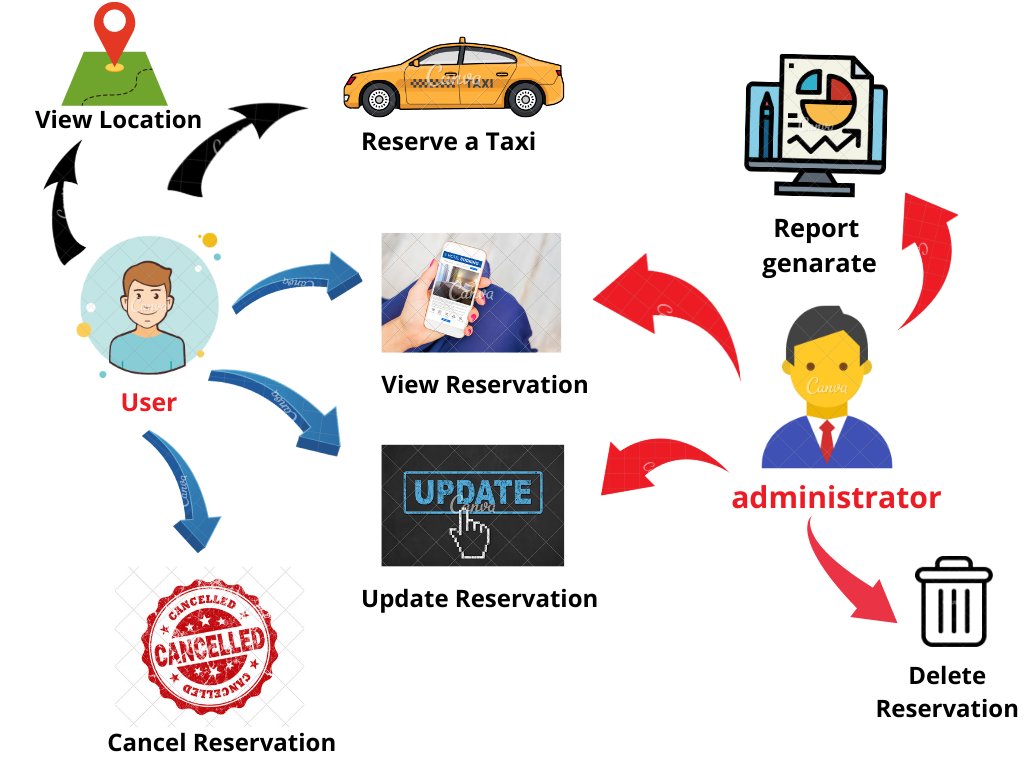
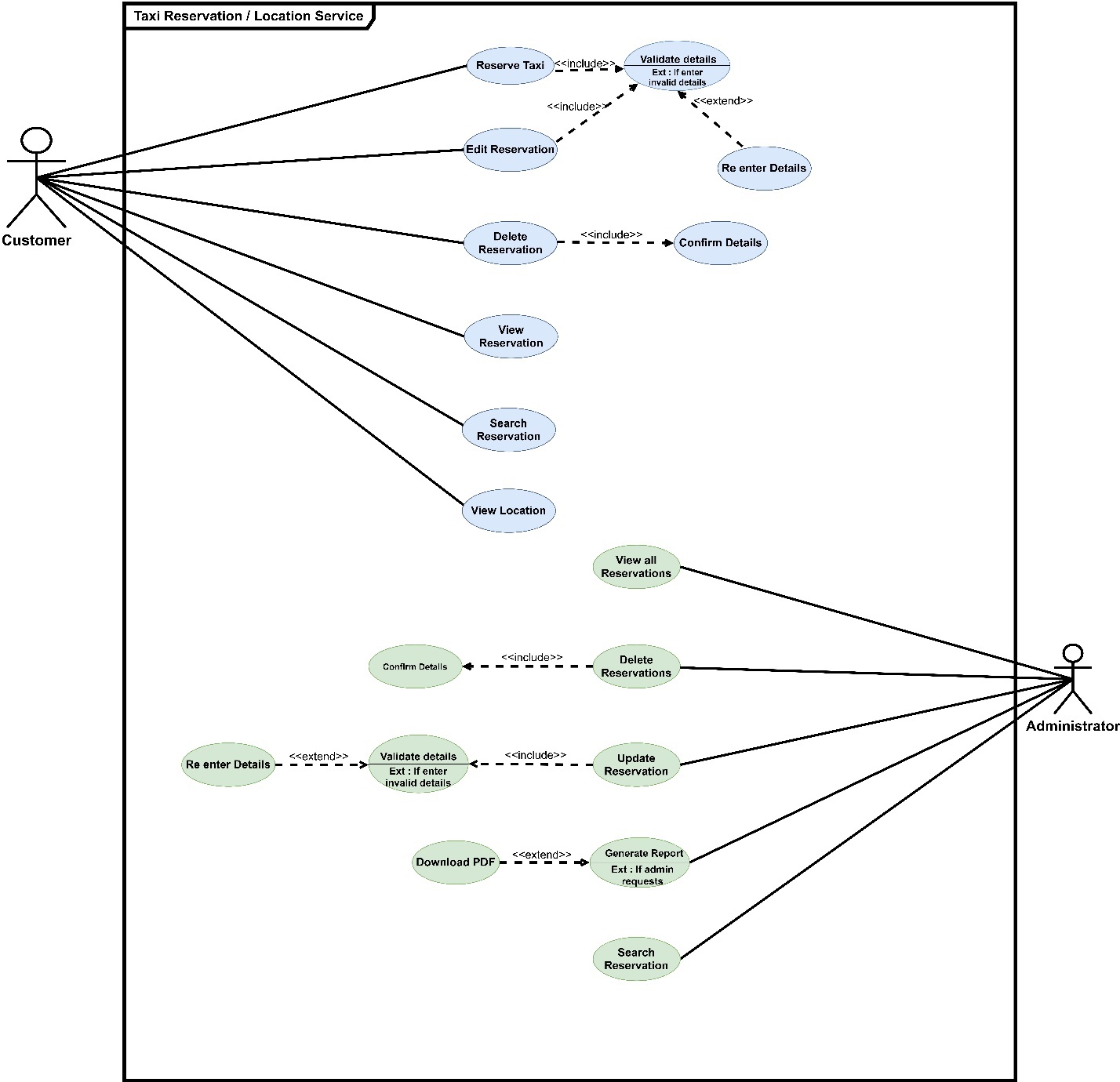
**Taxi Reservation and the Location Service**

**Figure 2. An overview of Taxi reservation and Location Service**

As shown in figure 2, in this function customer can reserve the taxi and use the find the hotel location by google map service. An also administrator can manage the taxi services and generate the report(see the code in **Appendix 05** ) of the taxi bookings done by the customers. There will be two different user roles in this function. Those levels are Administrator level and the customer level. Customer can reserve the taxi(see the code in **Appendix 02** ), After reserve a taxi the customer get a auto generated reference number. Then customer can update the reservation done by them(see the code in **Appendix 03**), Cancel the reservation(see the code in **Appendix 01)** and also view the specific taxi booking details done by each customer(see the code in **Appendix 04)**. There was a search function for the customer so that the customer can easily find there reservation details by search by there details. Apart from that the user has a chance to view the hotel location by the web application(see the code in **Appendix 06)**. Administrator can see the all taxi reservation details done by the customers and search the reservation details. And also the admin can update the taxi reservation details, and also delete the taxi reservations. Administrator can generate the report by daily or monthly and he/she can download the report for administrative purposes (see the code in **Appendix 05** ()).

**Use Case Diagram**

****

**Figure 2. Use Case Diagram of Taxi reservation and Location Service**

**APPENDIX**

**Taxi reservation –**

Backend

Models - shv\_taxi\_model.js

const mongoose = require('mongoose');

const taxiSchema = new mongoose.Schema({

    fistName:{

        type:String,

        required:true

    },

    lastName:{

        type:String,

        required:true

    },

    mobileNumber:{

        type:String,

        required:true

    },

    nic:{

        type:String,

        required:true

    },

    email:{

        type:String,

        required:true

    },

    typeoftaxi:{

        type:String,

        required:true

    },

    condition:{

        type:String,

        required:true

    },

    bookingDate:{

        type:String,

        required:true

    },

    bookingTime:{

        type:String,

        required:true

    },

    numberOfPassengers:{

        type:String,

        required:true

    },

    pickupAddress :{

        type:String,

        required:true

    },

    dropOffAddress:{

        type:String,

        required:true

    }

},

  {

    timestamps: true

  }

);

module.exports = mongoose.model('taxibooking', taxiSchema);

**Routes - shv\_taxi\_route.js**

const express = require('express');

const Taxi = require('../models/shv\_taxi\_model');

const router = express.Router();

//Save

router.post('/postTaxi/save', (req, res) =>{

    let newTaxi = new Taxi(req.body);

    newTaxi.save((err) =>{

            if(err){

                return res.status(400).json({

                    error:err

                });

            }

            return res.status(200).json({

                success:"Taxi Reserved Successfully"

            });

    });

});

// Retrieve

router.get('/postTaxi', (req,res) =>{

    Taxi.find().exec((err,taxi) =>{

        if(err){

            return res.status(400).json({

                error:err

            });

        }

        return res.status(200).json({

            success:true,

            existingtaxi:taxi

        });

    });

});

//get specific

router.get("/postTaxi/:id",(req,res) =>{

    let TaxiId = req.params.id;

    Taxi.findById(TaxiId,(err,taxi)=>{

        if(err){

            return res.status(400).json({success:false, err});

        }

        return res.status(200).json({

            success:true,

            taxi

        });

    });

});

// update

router.put('/postTaxi/update/:id', (req, res)=>{

    Taxi.findByIdAndUpdate(

        req.params.id,

        {

            $set:req.body

        },

        (err, taxi) => {

            if(err){

                return res.status(400).json({error:err});

            }

            return res.status(200).json({

                success:"Updated Successfully"

            });

        }

    );

})

// Delete

router.delete('/postTaxi/delete/:id', (req,res) =>{

    Taxi.findByIdAndRemove(req.params.id).exec((err, deletedtaxi) =>{

        if(err) return res.status(400).json({

            message: "Delete Unsuccessfully", err

        });

        return res.json({

            message:"Deleted successfull", deletedtaxi

        });

    });

});

module.exports = router;

**Backend - Server.js**

const express = require("express");

const mongoose = require("mongoose");

const bodyParser = require("body-parser");

const cors = require("cors");

const app = express();

// import routes

const taxiRouts = require("./routes/shv\_taxi\_route");

// Middlewares

app.use(bodyParser.json());

app.use(cors());

// routs.middlewares

app.use(taxiRouts);

// port

const PORT = 8001;

const DB\_URL =

  "mongodb+srv://dsProj123:dsp2yAbc@dsprojectcluster.iiyxv.mongodb.net/DSDB?retryWrites=true&w=majority";

mongoose

  .connect(DB\_URL)

  .then(() => {

    console.log("Database Connected");

  })

  .catch((err) => console.log("DB connection error !!", err));

app.listen(PORT, () => {

  console.log(`App is running on ${PORT}`);

});

**Frontend**

**shv\_All\_Taxi\_Reservations.js - APPENDIX 01**

import React, { Component } from "react";

import axios from "axios";

import swal from "sweetalert";

export default class Shv\_All\_Taxi\_Reservations extends Component {

  constructor(props) {

    super(props);

    this.state = {

      TaxiPosts: [],

    };

  }

  componentDidMount() {

    this.retrieveTaxiPosts();

  }

  retrieveTaxiPosts() {

    axios.get("http://localhost:8280/taxi/getTaxi").then((res) => {

      if (res.data.success) {

        this.setState({

          TaxiPosts: res.data.existingtaxi,

        });

        console.log(this.state.TaxiPosts);

      }

    });

  }

  onDelete = (id) => {

    swal({

      title: "Are you sure?",

      text: "Once deleted, you will not be able to recover this Taxi Booking Details!",

      icon: "warning",

      buttons: true,

      dangerMode: true,

    }).then((willDelete) => {

      if (willDelete) {

        axios

          .delete(`http://localhost:8280/taxi/deleteTaxi/${id}`)

          .then((res) => {

            this.retrieveTaxiPosts();

          });

        swal("Done! Your Taxi Booking was cancelled!", {

          icon: "success",

        });

      } else {

        swal("Not deleted ! Taxi Booking Details are safe !");

      }

    });

  };

  filterData(TaxiPosts, searchKey) {

    const result = TaxiPosts.filter(

      (TaxiPost) =>

        TaxiPost.\_id.toLowerCase().includes(searchKey) ||

        TaxiPost.fistName.toLowerCase().includes(searchKey) ||

        TaxiPost.lastName.toLowerCase().includes(searchKey) ||

        TaxiPost.mobileNumber.toLowerCase().includes(searchKey) ||

        TaxiPost.nic.toLowerCase().includes(searchKey) ||

        TaxiPost.typeoftaxi.toLowerCase().includes(searchKey) ||

        TaxiPost.condition.toLowerCase().includes(searchKey) ||

        TaxiPost.bookingDate.toLowerCase().includes(searchKey) ||

        TaxiPost.bookingTime.toLowerCase().includes(searchKey) ||

        TaxiPost.numberOfPassengers.toLowerCase().includes(searchKey) ||

        TaxiPost.pickupAddress.toLowerCase().includes(searchKey) ||

        TaxiPost.dropOffAddress.toLowerCase().includes(searchKey)

    );

    this.setState({ TaxiPosts: result });

  }

  handleSearchArea = (e) => {

    const searchKey = e.currentTarget.value;

    axios.get("http://localhost:8280/taxi/getTaxi").then((res) => {

      if (res.data.success) {

        this.filterData(res.data.existingtaxi, searchKey);

      }

    });

  };

  render() {

    return (

      <div>

        {/\* <Shv\_Taxi\_NavBar /> \*/}

        <div>

          <center>

            <h3 style={{ marginTop: "5%" }}>

              <font face="Comic sans MS" size="6">

                <b>Reserved Taxis</b>

              </font>

            </h3>

          </center>

          <br />

          <nav className="navbar navbar-light bg-light">

            <button className="btn btn-success">

              <a

                href="#"

                style={{ textDecoration: "none", color: "white" }}

              ></a>

            </button>

            <form className="form-inline">

              <input

                className="form-control mr-sm-2"

                type="search"

                placeholder="Search"

                aria-label="Search"

                name="searchQuery"

                onChange={this.handleSearchArea}

              />

            </form>

          </nav>

          <table className="table table-hover">

            <thead>

              <tr className="table-primary">

                <th scope="col">#</th>

                <th scope="col">Reference No</th>

                <th scope="col">First name</th>

                <th scope="col">Last Name</th>

                <th scope="col">Contact number</th>

                <th scope="col">NIC</th>

                <th scope="col">Email</th>

                <th scope="col">Type of Taxi</th>

                <th scope="col">Condition</th>

                <th scope="col">Booking Date</th>

                <th scope="col">Booking Time</th>

                <th scope="col">No Of Passengers</th>

                <th scope="col">Pickup Addess</th>

                <th scope="col">Drop Off Address</th>

                <th scope="col">View</th>

                <th scope="col">Edit</th>

                <th scope="col">Delete</th>

              </tr>

            </thead>

            <tbody>

              {this.state.TaxiPosts.map((TaxiPosts, index) => (

                <tr key={index}>

                  <th scope="row">{index + 1}</th>

                  <td>

                    <a

                      href={`/ViewTaxi/${TaxiPosts.\_id}`}

                      style={{ textDecoration: "none" }}

                    >

                      {TaxiPosts.\_id}

                    </a>

                  </td>

                  <td>{TaxiPosts.fistName}</td>

                  <td>{TaxiPosts.lastName}</td>

                  <td>{TaxiPosts.mobileNumber}</td>

                  <td>{TaxiPosts.nic}</td>

                  <td>{TaxiPosts.email}</td>

                  <td>{TaxiPosts.typeoftaxi}</td>

                  <td>{TaxiPosts.condition}</td>

                  <td>{TaxiPosts.bookingDate}</td>

                  <td>{TaxiPosts.bookingTime}</td>

                  <td>{TaxiPosts.numberOfPassengers}</td>

                  <td>{TaxiPosts.pickupAddress}</td>

                  <td>{TaxiPosts.dropOffAddress}</td>

                  <td>

                    <a

                      className="btn btn-primary"

                      href={`/ViewTaxi/${TaxiPosts.\_id}`}

                    >

                      View

                    </a>

                  </td>

                  <td>

                    <a

                      className="btn btn-warning"

                      href={`/UpdateTaxi/${TaxiPosts.\_id}`}

                    >

                      Edit

                    </a>

                  </td>

                  <td>

                    <a

                      className="btn btn-danger"

                      onClick={(event) => this.onDelete(TaxiPosts.\_id)}

                    >

                      Delete

                    </a>

                  </td>

                </tr>

              ))}

            </tbody>

          </table>

          <br />

          <br />

          <br />

          <br />

          <br />

        </div>

      </div>

    );

  }

}

**shv\_Create\_Taxi\_Reservations.js - APPENDIX 02**

import React, { Component } from 'react'

import axios from 'axios';

import swal from 'sweetalert';

import Swal from 'sweetalert2';

import Shv\_Taxi\_NavBar from '../shv\_NavBars/shv\_Taxi\_NavBar';

export default class Shv\_Create\_Taxi\_Reservations extends Component {

  constructor(props){

    super(props);

    this.state={

      fistName:"",

      lastName:"",

      mobileNumber:"",

      nic:"",

      email:"",

      typeoftaxi:"",

      condition:"",

      bookingDate:"",

      bookingTime:"",

      numberOfPassengers:"",

      pickupAddress :"",

      dropOffAddress :""

    }

  }

  handleInputChange = (e) =>{

    const {name,value} = e.target;

    this.setState({

      ...this.state,

      [name]:value

    })

  }

  onSubmit = (e) =>{

    e.preventDefault();

    const {fistName,lastName,mobileNumber,nic,email,typeoftaxi,condition,bookingDate,bookingTime,numberOfPassengers,pickupAddress ,dropOffAddress } = this.state;

    const data ={

      fistName:fistName,

      lastName:lastName,

      mobileNumber:mobileNumber,

      nic:nic,

      email:email,

      typeoftaxi:typeoftaxi,

      condition:condition,

      bookingDate:bookingDate,

      bookingTime:bookingTime,

      numberOfPassengers:numberOfPassengers,

      pickupAddress:pickupAddress,

      dropOffAddress:dropOffAddress

    }

    console.log(data)

     //Validation

     const con = /^[0-9\b]+$/;

     const Temail = /^(([^<>()[\]\\.,;:\s@"]+(\.[^<>()[\]\\.,;:\s@"]+)\*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;

     const name = /^[a-zA-Z-,]+(\s{0,1}[a-zA-Z-, ])\*$/;

     if(fistName.length == 0 || lastName.length == 0 || mobileNumber.length == 0 || nic.length == 0 || email.length == 0 || typeoftaxi.length == 0 || condition.length == 0 || bookingDate.length == 0 || bookingTime.length == 0 || numberOfPassengers.length == 0 || pickupAddress.length == 0 || dropOffAddress.length == 0 ){

      swal("Feilds Cannot Be empty !!", "You Must fill all the feilds !!");

    }else if((!con.test(String(mobileNumber)))||(mobileNumber.length != 10)){

      swal("Invalid Contact Number !", "Please enter valid contact number !", "error");

    }else if((!Temail.test(String(email)))){

      swal("Invalid email address !", "Please enter valid email address !", "error");

    }else if((!name.test(String(fistName)))){

      swal("Invalid Name !", "Name cannot contain numbers ! Please enter valid name !", "error");

    }else if((!name.test(String(lastName)))){

      swal("Invalid Name !", "Name cannot contain numbers ! Please enter valid name !", "error");

    }else{

    axios.post("http://localhost:8280/taxi/saveTaxi",data).then((res) =>{

      if(res.data.success){

        this.setState(

          {

            fistName:"",

            lastName:"",

            mobileNumber:"",

            nic:"",

            email:"",

            typeoftaxi:"",

            condition:"",

            bookingDate:"",

            bookingTime:"",

            numberOfPassengers:"",

            pickupAddress :"",

            dropOffAddress :""

          }

        )

      }

    });

          Swal.fire({

            title: "Your Taxi Booked",

            type: "success",

            showConfirmButton: true,

            confirmButtonText: 'View',

            icon: 'success',

        }).then(function() {

            window.location = "/AllTaxi";

        });

    }

  }

     //method of demo button

     demo =() => {

      //setState

      this.setState( {

        ...this.state,

        fistName:"Kamal",

        lastName:"Perera",

        mobileNumber:"0788899887",

        nic:"675565433V",

        email:"Kamal@gmail.com",

        typeoftaxi:"Van",

        condition:"Npn A/C",

        bookingDate:"2021-09-07",

        bookingTime:"04:29",

        numberOfPassengers:"10",

        pickupAddress :"5th Lane, Colombo 07",

        dropOffAddress:"Hilton Surface Paradise, Colomo 10"

      })

    }

  render() {

    return (

    <div>

      <Shv\_Taxi\_NavBar/>

      <div className="container border"

          style={{

          marginTop: "80px",

          width: '100%',

          backgroundImage: `url('https://bhiroof.com/wp-content/uploads/2020/10/empty-living-room-with-blue-sofa-plants-table-empty-white-wall-background-3d-rendering\_41470-1778.jpg')`,

          backgroundPosition: 'center',

          backgroundSize: 'cover',

          }}>

          <div >

        <div>

          <center>

          <h1 className="h3 mb-1 font-weight-normal" style = {{backgroundColor:'#990099'}}><font face = "Comic sans MS" size ="6" color = "white"><b>Reserve Your Taxi</b></font></h1><br/>&nbsp;

          </center>

          <br/>

          <center>

          <form className="needs-validation">

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}} ><b>First Name</b></label>

              <input type="text"

              className="form-control"

              name="fistName"

              placeholder="Enter fistName"

              value={this.state.fistName}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}}><b>Last Name</b></label>

              <input type="text"

              className="form-control"

              name="lastName"

              placeholder="lastName"

              value={this.state.lastName}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}}><b>Mobile Number</b></label>

              <input type="number"

              className="form-control"

              name="mobileNumber"

              placeholder="Enter contact number"

              value={this.state.mobileNumber}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}}><b>NIC</b></label>

              <input type="text"

              className="form-control"

              name="nic"

              placeholder="Enter nic"

              value={this.state.nic}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}}><b>Email</b></label>

              <input type="text"

              className="form-control"

              name="email"

              placeholder="Enter email"

              value={this.state.email}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group">

            <label style={{marginBottom:'5px'}}><b>Type Of Taxi</b></label>

              <select

              className="form-control"

              style ={{marginBottom:'15px', maxWidth:'500px'}}

              name="typeoftaxi"

              value = {this.state.typeoftaxi}

              onChange = {this.handleInputChange}

              required

              >

              <option value="" disabled selected hidden>Please Choose...</option>

              <option value = "Car">Car</option>

              <option value = "Van">Van</option>

              <option value = "Bus">Bus</option>

              <option value = "Tuk">Tuk</option>

              </select>

          </div>

            <div className="form-group">

            <label style={{marginBottom:'5px'}}><b>Condition</b></label>

              <select

              className="form-control"

              style ={{marginBottom:'15px', maxWidth:'500px'}}

              name="condition"

              value = {this.state.condition}

              onChange = {this.handleInputChange}

              required

              >

              <option value="" disabled selected hidden>Please Choose...</option>

              <option value = "AC">AC</option>

              <option value = "Non/AC">Non/AC</option>

              </select>

          </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}}><b>Booking Date</b></label>

              <input type="Date"

              className="form-control"

              name="bookingDate"

              placeholder="Enter booking date"

              value={this.state.bookingDate}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}}><b>Booking Time</b></label>

              <input type="time"

              className="form-control"

              name="bookingTime"

              placeholder="Enter booking time"

              value={this.state.bookingTime}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}}><b>Number Of Passengers</b></label>

              <input type="number"

              className="form-control"

              name="numberOfPassengers"

              placeholder="No of passengers"

              value={this.state.numberOfPassengers}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}} ><b>Pick Up Address</b></label>

              <input type="text"

              className="form-control"

              name="pickupAddress"

              placeholder="Enter Pick Up Address"

              value={this.state.pickupAddress}

              onChange={this.handleInputChange}

              required/>

            </div>

            <div className="form-group" style={{marginBottom:'15px', maxWidth:'500px'}}>

              <label style={{marginBottom:'5px'}} ><b>Drop Off Address</b></label>

              <input type="text"

              className="form-control"

              name="dropOffAddress"

              placeholder="Enter drop off address"

              value={this.state.dropOffAddress}

              onChange={this.handleInputChange}

              required/>

            </div>

            <button className="btn btn-success btn-lg" type="submit" style={{marginTop:'15px'}} onClick={this.onSubmit}>

              &nbsp; Reserve

            </button>

          </form>

       <br/><br/>

          <button className ="btn btn-warning" onClick={() => {

              this.setState( {

                ...this.state,

                fistName:"Kamal",

                lastName:"Perera",

                mobileNumber:"0788899887",

                nic:"675565433V",

                email:"Kamal@gmail.com",

                typeoftaxi:"Van",

                condition:"Non A/C",

                bookingDate:"2021-09-07",

                bookingTime:"5:00 p.m",

                numberOfPassengers:"10",

                pickupAddress :"5th Lane, Colombo 07",

                dropOffAddress:"Hilton Hotel, Colomo 10"

                })

            }}>Demo</button>

               </center>

        </div>

        </div>

        </div>

        <br/><br/><br/>

    </div>

    )

  }

}

**shv\_Update\_Taxi\_Reservations.js - APPENDIX 03**

import React, { Component } from "react";

import Shv\_Taxi\_NavBar from "../shv\_NavBars/shv\_Taxi\_NavBar";

import axios from "axios";

import swal from "sweetalert";

export default class Shv\_Update\_Taxi\_Reservations extends Component {

  constructor(props) {

    super(props);

    this.state = {

      fistName: "",

      lastName: "",

      mobileNumber: "",

      nic: "",

      email: "",

      typeoftaxi: "",

      condition: "",

      bookingDate: "",

      bookingTime: "",

      numberOfPassengers: "",

      pickupAddress: "",

      dropOffAddress: "",

    };

  }

  handleInputChange = (e) => {

    const { name, value } = e.target;

    this.setState({

      ...this.state,

      [name]: value,

    });

  };

  onSubmit = (e) => {

    e.preventDefault();

    const id = this.props.match.params.id;

    const {

      fistName,

      lastName,

      mobileNumber,

      nic,

      email,

      typeoftaxi,

      condition,

      bookingDate,

      bookingTime,

      numberOfPassengers,

      pickupAddress,

      dropOffAddress,

    } = this.state;

    const data = {

      fistName: fistName,

      lastName: lastName,

      mobileNumber: mobileNumber,

      nic: nic,

      email: email,

      typeoftaxi: typeoftaxi,

      condition: condition,

      bookingDate: bookingDate,

      bookingTime: bookingTime,

      numberOfPassengers: numberOfPassengers,

      pickupAddress: pickupAddress,

      dropOffAddress: dropOffAddress,

    };

    console.log(data);

    //Validation

    const con = /^[0-9\b]+$/;

    const Temail =

      /^(([^<>()[\]\\.,;:\s@"]+(\.[^<>()[\]\\.,;:\s@"]+)\*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;

    const name = /^[a-zA-Z-,]+(\s{0,1}[a-zA-Z-, ])\*$/;

    if (

      fistName.length == 0 ||

      lastName.length == 0 ||

      mobileNumber.length == 0 ||

      nic.length == 0 ||

      email.length == 0 ||

      typeoftaxi.length == 0 ||

      condition.length == 0 ||

      bookingDate.length == 0 ||

      bookingTime.length == 0 ||

      numberOfPassengers.length == 0 ||

      pickupAddress.length == 0 ||

      dropOffAddress.length == 0

    ) {

      swal("Feilds Cannot Be empty !!", "You Must fill all the feilds !!");

    } else if (!con.test(String(mobileNumber)) || mobileNumber.length != 10) {

      swal(

        "Invalid Contact Number !",

        "Please enter valid contact number !",

        "error"

      );

    } else if (!Temail.test(String(email))) {

      swal(

        "Invalid email address !",

        "Please enter valid email address !",

        "error"

      );

    } else if (!name.test(String(fistName))) {

      swal(

        "Invalid Name !",

        "Name cannot contain numbers ! Please enter valid name !",

        "error"

      );

    } else if (!name.test(String(lastName))) {

      swal(

        "Invalid Name !",

        "Name cannot contain numbers ! Please enter valid name !",

        "error"

      );

    } else {

      axios

        .put(`http://localhost:8280/room/updateRoom/${id}`, data)

        .then((res) => {

          if (res.data.success) {

            swal("Successful!", "Taxi Details Updated !", "success");

            this.setState({

              fistName: "",

              lastName: "",

              mobileNumber: "",

              nic: "",

              email: "",

              typeoftaxi: "",

              condition: "",

              bookingDate: "",

              bookingTime: "",

              numberOfPassengers: "",

              pickupAddress: "",

              dropOffAddress: "",

            });

          }

        });

    }

  };

  componentDidMount() {

    const id = this.props.match.params.id;

    axios.get(`http://localhost:8280/taxi/getTaxi/${id}`).then((res) => {

      if (res.data.success) {

        this.setState({

          fistName: res.data.taxi.fistName,

          lastName: res.data.taxi.lastName,

          mobileNumber: res.data.taxi.mobileNumber,

          nic: res.data.taxi.nic,

          email: res.data.taxi.email,

          typeoftaxi: res.data.taxi.typeoftaxi,

          condition: res.data.taxi.condition,

          bookingDate: res.data.taxi.bookingDate,

          bookingTime: res.data.taxi.bookingTime,

          numberOfPassengers: res.data.taxi.numberOfPassengers,

          pickupAddress: res.data.taxi.pickupAddress,

          dropOffAddress: res.data.taxi.dropOffAddress,

        });

        console.log(this.state.taxi);

      }

    });

  }

  render() {

    return (

      <div>

        <Shv\_Taxi\_NavBar />

        <div>

          <div className="card">

            <div>

              <center>

                <h1

                  className="h3 mb-3 font-weight-normal"

                  style={{ backgroundColor: "cyan" }}

                >

                  <font face="Comic sans MS" size="6">

                    <b>Update Taxi Reservation Details</b>

                  </font>

                </h1>

                <br />

                &nbsp;

              </center>

              <br />

              <center>

                <form className="needs-validation">

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>First Name</b>

                    </label>

                    <input

                      type="text"

                      className="form-control"

                      name="fistName"

                      placeholder="Enter fistName"

                      value={this.state.fistName}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>Last Name</b>

                    </label>

                    <input

                      type="text"

                      className="form-control"

                      name="lastName"

                      placeholder="lastName"

                      value={this.state.lastName}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>Mobile Number</b>

                    </label>

                    <input

                      type="text"

                      className="form-control"

                      name="mobileNumber"

                      placeholder="Enter contact number"

                      value={this.state.mobileNumber}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>NIC</b>

                    </label>

                    <input

                      type="text"

                      className="form-control"

                      name="nic"

                      placeholder="Enter nic"

                      value={this.state.nic}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>Email</b>

                    </label>

                    <input

                      type="text"

                      className="form-control"

                      name="email"

                      placeholder="Enter email"

                      value={this.state.email}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div className="form-group">

                    <label style={{ marginBottom: "5px" }}>

                      <b>Type Of Taxi</b>

                    </label>

                    <select

                      className="form-control"

                      style={{ marginBottom: "15px", maxWidth: "500px" }}

                      name="typeoftaxi"

                      value={this.state.typeoftaxi}

                      onChange={this.handleInputChange}

                      required

                    >

                      <option value="" disabled selected hidden>

                        Please Choose...

                      </option>

                      <option value="Car">Car</option>

                      <option value="Van">Van</option>

                      <option value="Bus">Bus</option>

                      <option value="Tuk">Tuk</option>

                    </select>

                  </div>

                  <div className="form-group">

                    <label style={{ marginBottom: "5px" }}>

                      <b>Condition</b>

                    </label>

                    <select

                      className="form-control"

                      style={{ marginBottom: "15px", maxWidth: "500px" }}

                      name="condition"

                      value={this.state.condition}

                      onChange={this.handleInputChange}

                      required

                    >

                      <option value="" disabled selected hidden>

                        Please Choose...

                      </option>

                      <option value="AC">AC</option>

                      <option value="Non/AC">Non/AC</option>

                    </select>

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>Booking Date</b>

                    </label>

                    <input

                      type="Date"

                      className="form-control"

                      name="bookingDate"

                      placeholder="Enter booking date"

                      value={this.state.bookingDate}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>Booking Time</b>

                    </label>

                    <input

                      type="time"

                      className="form-control"

                      name="bookingTime"

                      placeholder="Enter booking time"

                      value={this.state.bookingTime}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>Number Of Passengers</b>

                    </label>

                    <input

                      type="number"

                      className="form-control"

                      name="numberOfPassengers"

                      placeholder="No of passengers"

                      value={this.state.numberOfPassengers}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>Pick Up Address</b>

                    </label>

                    <input

                      type="text"

                      className="form-control"

                      name="pickupAddress"

                      placeholder="Enter Pick Up Address"

                      value={this.state.pickupAddress}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <div

                    className="form-group"

                    style={{ marginBottom: "15px", maxWidth: "500px" }}

                  >

                    <label style={{ marginBottom: "5px" }}>

                      <b>Drop Off Address</b>

                    </label>

                    <input

                      type="text"

                      className="form-control"

                      name="dropOffAddress"

                      placeholder="Enter drop off address"

                      value={this.state.dropOffAddress}

                      onChange={this.handleInputChange}

                      required

                    />

                  </div>

                  <button

                    className="btn btn-success btn-lg"

                    type="submit"

                    style={{ marginTop: "15px" }}

                    onClick={this.onSubmit}

                  >

                    &nbsp; Update

                  </button>

                </form>

              </center>

            </div>

          </div>

        </div>

        <br />

        <br />

        <br />

      </div>

    );

  }

}

**shv\_View\_Taxi\_Reservations.js - APPENDIX 04**

import React, { Component } from 'react';

import axios from 'axios';

import Shv\_Taxi\_NavBar from '../shv\_NavBars/shv\_Taxi\_NavBar';

import jsPdf from 'jspdf'

import 'jspdf-autotable'

export default class Shv\_View\_Taxi\_Reservations extends Component {

  constructor(props){

    super(props);

    this.state={

      taxi:{}

    };

  }

  componentDidMount(){

    const id = this.props.match.params.id;

    axios.get(`http://localhost:8280/taxi/getTaxi/${id}`).then((res) =>{

      if(res.data.success){

        this.setState({

          taxi:res.data.taxi

        });

        console.log(this.state.taxi);

      }

    })

  }

  //Report pdf generating

  jsPdfGenerator = () => {

    //new document in jspdf

    var doc = new jsPdf('l', 'pt', 'a4');

    doc.text(350, 40 ,'Reservation Details');

    var pdfjs = document.querySelector('#taxiReservation');

    var date = new Date()

    doc.text("Generated Date : " + date,24,24)

  // Convert HTML to PDF in JavaScript

  doc.html(pdfjs, {

    callback: function(doc) {

      doc.save("Booking Details.pdf");

    },

    x: 50,

    y: 50

  });

  }

  render() {

    const {fistName,lastName,mobileNumber,nic,email,typeoftaxi,condition,bookingDate,bookingTime,numberOfPassengers,pickupAddress ,dropOffAddress} = this.state.taxi;

    return(

      <div>

      <Shv\_Taxi\_NavBar/>

      <div className="container border"

        style={{

          marginTop: "80px",

          width: '100%',

          backgroundImage: `url('https://bhiroof.com/wp-content/uploads/2020/10/empty-living-room-with-blue-sofa-plants-table-empty-white-wall-background-3d-rendering\_41470-1778.jpg')`,

          backgroundPosition: 'center',

          backgroundSize: 'cover',

        }}>

        <div style={{ marginTop: '20px' }}>

          <h1>Taxi Reservation Details</h1>

          <br/>

          <dl id = "taxiReservation" className="row">

            <dt className="col-sm-2">First Name</dt><br></br>

            <dd className="col-sm-9">{fistName}</dd><br></br>

            <dt className="col-sm-2">Last Name</dt><br></br>

            <dd className="col-sm-9">{lastName}</dd><br></br>

            <dt className="col-sm-2">Mobile Number</dt><br></br>

            <dd className="col-sm-9">{mobileNumber}</dd><br></br>

            <dt className="col-sm-2">NIC</dt><br></br>

            <dd className="col-sm-9">{nic}</dd><br></br>

            <dt className="col-sm-2">Email</dt><br></br>

            <dd className="col-sm-9">{email}</dd><br></br>

            <dt className="col-sm-2">Type Of Taxi</dt><br></br>

            <dd className="col-sm-9">{typeoftaxi}</dd><br></br>

            <dt className="col-sm-2">Condition</dt><br></br>

            <dd className="col-sm-9">{condition}</dd><br></br>

            <dt className="col-sm-2">Booking Date</dt><br></br>

            <dd className="col-sm-9">{bookingDate}</dd><br></br>

            <dt className="col-sm-2">Booking Time</dt><br></br>

            <dd className="col-sm-9">{bookingTime}</dd><br></br>

            <dt className="col-sm-2">Passengers</dt><br></br>

            <dd className="col-sm-9">{numberOfPassengers}</dd><br></br>

            <dt className="col-sm-2">Pick Up address</dt><br></br>

            <dd className="col-sm-9">{pickupAddress }</dd><br></br>

            <dt className="col-sm-2">Drop Off Address</dt><br></br>

            <dd className="col-sm-9">{dropOffAddress}</dd><br></br>

          </dl>

          <button className="btn btn-danger" onClick={this.jsPdfGenerator}>Download PDF</button>

        </div>

      </div>

      </div>

  )

  }

}

**shv\_admin\_taxi\_reservation\_report.js - - APPENDIX 05**

import React, { Component } from 'react'

import axios from 'axios';

import Shv\_Taxi\_NavBar from '../shv\_NavBars/shv\_Taxi\_NavBar';

import jsPdf from 'jspdf'

import 'jspdf-autotable'

export default class Shv\_AdminTaxiReport extends Component {

    constructor(props){

        super(props);

        this.state={

            TaxiPosts:[]

        };

    }

    componentDidMount(){

        this.retrieveTaxiPosts();

    }

    retrieveTaxiPosts(){

        axios.get("http://localhost:8001/postTaxi").then(res=>{

            if(res.data.success){

                this.setState({

                    TaxiPosts:res.data.existingtaxi

                });

                console.log(this.state.TaxiPosts);

            }

        });

    }

    filterData(TaxiPosts, searchKey){

        const result = TaxiPosts.filter((TaxiPost) =>

        TaxiPost.\_id.toLowerCase().includes(searchKey)||

        TaxiPost.fistName.toLowerCase().includes(searchKey)||

        TaxiPost.lastName.toLowerCase().includes(searchKey)||

        TaxiPost.mobileNumber.toLowerCase().includes(searchKey)||

        TaxiPost.nic.toLowerCase().includes(searchKey)||

        TaxiPost.typeoftaxi.toLowerCase().includes(searchKey)||

        TaxiPost.condition.toLowerCase().includes(searchKey)||

        TaxiPost.bookingDate.toLowerCase().includes(searchKey)||

        TaxiPost.bookingTime.toLowerCase().includes(searchKey)||

        TaxiPost.numberOfPassengers.toLowerCase().includes(searchKey)||

        TaxiPost.pickupAddress .toLowerCase().includes(searchKey)||

        TaxiPost.dropOffAddress.toLowerCase().includes(searchKey)

      )

      this.setState({TaxiPosts:result})

    }

    handleSearchArea = (e) =>{

      const searchKey = e.currentTarget.value;

      axios.get("http://localhost:8280/taxi/getTaxi").then(res =>{

        if(res.data.success){

          this.filterData(res.data.existingtaxi, searchKey)

        }

      })

    }

  //Report pdf generating

  jsPdfGenerator = () => {

    //new document in jspdf

    var doc = new jsPdf('l','pt', 'a3');

    var date = new Date()

    doc.text("Generated Date : " + date,24,24)

    doc.text(600, 40 ,'Taxi Reservation Report', { align: 'center', marginTop:50 });

    doc.autoTable({  html:'#taxiReservation-table' })

    doc.autoTable({

      columnStyles: { europe: { halign: 'center' } },

    })

    //save the pdf

    doc.save("Taxi Reservation Report.pdf");

  }

  render() {

    return (

        <div>

          <Shv\_Taxi\_NavBar/>

        <div>

            <center>

            <h3 style = {{marginTop:'5%'}}><font face = "Comic sans MS" size ="6"><b>Reserved Taxis</b></font></h3>

            </center>

            <br/>

        <nav className="navbar navbar-light bg-light">

        <button className="btn-primary" onClick={this.jsPdfGenerator}>Generate Report PDF</button>

          <form className="form-inline">

            <input className="form-control mr-sm-2" type="search" placeholder="Search" aria-label="Search" name = "searchQuery"

          onChange = {this.handleSearchArea} />

            </form>

        </nav>

          <table id = "taxiReservation-table" className = "table table-hover" >

          <thead>

          <tr className = "table-primary">

               <th scope = "col">#</th>

               <th scope = "col">Reference No</th>

               <th scope = "col">First name</th>

               <th scope = "col">Last Name</th>

               <th scope = "col">Contact number</th>

               <th scope = "col">NIC</th>

               <th scope = "col">Email</th>

               <th scope = "col">Type of Taxi</th>

               <th scope = "col">Condition</th>

               <th scope = "col">Booking Date</th>

               <th scope = "col">Booking Time</th>

               <th scope = "col">No Of Passengers</th>

               <th scope = "col">Pickup Addess</th>

               <th scope = "col">Drop Off Address</th>

               </tr>

           </thead>

           <tbody>

            {this.state.TaxiPosts.map((TaxiPosts, index) =>(

              <tr key = {index}>

                <th scope = "row">{index+1}</th>

                    <td>

                    <a href = {`/ViewTaxi/${TaxiPosts.\_id}`} style = {{textDecoration:'none'}}>

                    {TaxiPosts.\_id}

                    </a>

                    </td>

                    <td>{TaxiPosts.fistName}</td>

                    <td>{TaxiPosts.lastName}</td>

                    <td>{TaxiPosts.mobileNumber}</td>

                    <td>{TaxiPosts.nic}</td>

                    <td>{TaxiPosts.email}</td>

                    <td>{TaxiPosts.typeoftaxi}</td>

                    <td>{TaxiPosts.condition}</td>

                    <td>{TaxiPosts.bookingDate}</td>

                    <td>{TaxiPosts.bookingTime}</td>

                    <td>{TaxiPosts.numberOfPassengers}</td>

                    <td>{TaxiPosts.pickupAddress }</td>

                    <td>{TaxiPosts.dropOffAddress}</td>

              </tr>

              ))}

            </tbody>

           </table>

           <br/><br/>

           <br/><br/><br/>

         </div>

         </div>

    )

  }

}

**LocationService.js - APPENDIX 06**

import React, { Component } from 'react'

import {Map, InfoWindow, Marker, GoogleApiWrapper} from 'google-maps-react';

export class LocationService extends Component {

  render() {

    return (

      <div>

        <center>

        <Map

          google={this.props.google}

          initialCenter={{

            lat: 6.9222,

            lng: 79.8549

          }}

          zoom={15}

          onClick={this.onMapClicked}

        >

        <Marker

        title={'Hilton Hotel'}

        name={'Hilton'}

        position={{lat: 6.9222, lng: 79.8549}} />

        <InfoWindow onClose={this.onInfoWindowClose}>

        </InfoWindow>

        </Map>

          </center>

      </div>

    )

  }

}

export default GoogleApiWrapper({

  apiKey: ("AIzaSyBx7EyhXLo23GvkH85n8037DTEgQm8wUtU")

})(LocationService)

**shv\_carList.js - APPENDIX 07**

import "./styles/shv\_carList.css";

const TaxiItem = () => {

  return (

    <div>

      <div className="searchItem">

      <img

        src="https://www.pricelanka.lk/wp-content/uploads/2021/03/Suzuki-Wagon-R-Stingray.png"

        alt=""

        className="siImg"

      />

      <div className="siDesc">

        <h1 className="siTitle">Budget Car</h1>

        <span className="siDistance">Suzuki</span>

        <span className="siTaxiOp">Suzuki Wagon - R</span>

        <span className="siSubtitle">

          Comfortable Ride

        </span>

        <span className="siFeatures">

          Parking • AC • 4 Seat • Wifi

        </span>

        <span className="siCancelOp">Free cancellation </span>

        <span className="siCancelOpSubtitle">

          You can cancel later, so lock in this great price today!

        </span>

      </div>

      <div className="siDetails">

        <div className="siRating">

          <span>Excellent</span>

          <button>8.9</button>

        </div>

        <div className="siDetailTexts">

          <span className="siPrice">100 LKR</span>

          <span className="siTaxOp">Per KM</span>

          <a href="/BookTaxi"><button className="siCheckButton">Reserve</button></a>

        </div>

      </div>

    </div><br/><br/>

    <div className="searchItem">

      <img

        src="https://carsguide-res.cloudinary.com/image/upload/f\_auto,fl\_lossy,q\_auto,t\_cg\_hero\_large/v1/editorial/segment\_review/hero\_image/2020-Mercedes-CLA-200-sedan-red-1001x565-%281%29.jpg"

        alt=""

        className="siImg"

      />

      <div className="siDesc">

        <h1 className="siTitle">Luxery Car</h1>

        <span className="siDistance">CLA-200</span>

        <span className="siTaxiOp">Mercedes Benz</span>

        <span className="siSubtitle">

          Comfortable Ride

        </span>

        <span className="siFeatures">

          Parking • AC • 4 Seat • Wifi

        </span>

        <span className="siCancelOp">Free cancellation </span>

        <span className="siCancelOpSubtitle">

          You can cancel later, so lock in this great price today!

        </span>

      </div>

      <div className="siDetails">

        <div className="siRating">

          <span>Excellent</span>

          <button>9.0</button>

        </div>

        <div className="siDetailTexts">

          <span className="siPrice">500 LKR</span>

          <span className="siTaxOp">Per KM</span>

          <a href="/BookTaxi"><button className="siCheckButton">Reserve</button></a>

        </div>

      </div>

    </div><br/><br/>

    <div className="searchItem">

      <img

        src="https://www.torqueautos.asia/vehicle\_image/22591\_b.jpeg"

        alt=""

        className="siImg"

      />

      <div className="siDesc">

        <h1 className="siTitle">Luxery Van</h1>

        <span className="siDistance">Toyota</span>

        <span className="siTaxiOp">High Ace</span>

        <span className="siSubtitle">

          Comfortable Ride

        </span>

        <span className="siFeatures">

          Parking • AC • 14 Seat • Wifi

        </span>

        <span className="siCancelOp">Free cancellation </span>

        <span className="siCancelOpSubtitle">

          You can cancel later, so lock in this great price today!

        </span>

      </div>

      <div className="siDetails">

        <div className="siRating">

          <span>Excellent</span>

          <button>9.0</button>

        </div>

        <div className="siDetailTexts">

          <span className="siPrice">750 LKR</span>

          <span className="siTaxOp">Per KM</span>

          <a href="/BookTaxi"><button className="siCheckButton">Reserve</button></a>

        </div>

      </div>

    </div><br/><br/>

    <div className="searchItem">

      <img

        src="https://media.ed.edmunds-media.com/land-rover/range-rover/2019/oem/2019\_land-rover\_range-rover\_4dr-suv\_autobiography\_fq\_oem\_1\_1600.jpg"

        alt=""

        className="siImg"

      />

      <div className="siDesc">

        <h1 className="siTitle">Luxery SUV</h1>

        <span className="siDistance">Sport</span>

        <span className="siTaxiOp">Mercedes Benz</span>

        <span className="siSubtitle">

          Comfortable Ride

        </span>

        <span className="siFeatures">

          Parking • AC • 6 Seat • Wifi

        </span>

        <span className="siCancelOp">Free cancellation </span>

        <span className="siCancelOpSubtitle">

          You can cancel later, so lock in this great price today!

        </span>

      </div>

      <div className="siDetails">

        <div className="siRating">

          <span>Excellent</span>

          <button>9.0</button>

        </div>

        <div className="siDetailTexts">

          <span className="siPrice">1000 LKR</span>

          <span className="siTaxOp">Per KM</span>

          <a href="/BookTaxi"><button className="siCheckButton">Reserve</button></a>

        </div>

      </div>

    </div>

    </div>

  );

};

export default TaxiItem;

**shv\_Taxi\_Booking\_Home.js - APPENDIX 08**

import React, { Component } from 'react'

import TaxiItem from './shv\_carList'

import Shv\_Taxi\_NavBar from '../shv\_NavBars/shv\_Taxi\_NavBar';

export default class extends Component {

  render() {

    return (

    <div>

        <Shv\_Taxi\_NavBar/><br/><br/>

      <div className='container'>

          <TaxiItem/>

      </div>

    </div>

    )

  }

}

**shv\_Taxi\_Home\_Admin.js - APPENDIX 09**

import React, { Component } from 'react'

export default class TaxiHomeAdmin extends Component {

    render() {

        return (

            <div>

            <br/><br/>

            <div>

            <header>

            <div className="p-5 text-center bg-image" style={{backgroundImage: 'url("https://wallpaperaccess.com/full/781957.jpg")', backgroundSize:'cover', height:"750px"}}>

                <div className="mask" style={{backgroundColor: 'rgba(0, 0, 0, 0.6)' , marginTop:"10px"}}>

                <div className="d-flex justify-content-center align-items-center h-100">

                    <div className="text-white"><br/>

                    <h1 className="mb-3">Hilton</h1>

                    <h4 className="mb-3">Hotel Reservation System</h4><br/>

                    <a className="btn btn-outline-light btn-lg" href="/AdminTaxiReport" role="button">Manage Taxi Reservation</a>

                    <br/><br/><br/>

                    </div>

                </div>

                </div>

            </div>

            </header>

            <br/><br/><br/><br/>

            </div>

        </div>

        )

    }

}

**shv\_Taxi\_Home\_Client.js - APPENDIX 10**import React, { Component } from 'react'

export default class TaxiHomeClient extends Component {

    render() {

        return (

            <div>

            <br/><br/>

            <div>

            <header>

            <div className="p-5 text-center bg-image" style={{backgroundImage: 'url("https://wallpaperaccess.com/full/781957.jpg")', backgroundSize:'cover', height:"750px"}}>

                <div className="mask" style={{backgroundColor: 'rgba(0, 0, 0, 0.6)' , marginTop:"10px"}}>

                <div className="d-flex justify-content-center align-items-center h-100">

                    <div className="text-white"><br/>

                    <h1 className="mb-3">Hilton</h1>

                    <h4 className="mb-3">Taxi Reservation System</h4><br/>

                    <a className="btn btn-outline-light btn-lg" href="/TaxiBookHome" role="button">Click Here To Reserve Your Taxi</a> &nbsp;&nbsp;&nbsp;&nbsp;

                    <a className="btn btn-outline-light btn-lg" href="/AllTaxi" role="button">Click Here To View Reserved Taxis</a>

                    <br/><br/><br/>

                    </div>

                </div>

                </div>

            </div>

            </header>

            <br/><br/><br/><br/>

            </div>

        </div>

        )

    }

}

**shv\_Taxi\_Reservations\_Home\_Page.js - - APPENDIX 11**

import React, { Component } from 'react'

import Shv\_Taxi\_NavBar from '../shv\_NavBars/shv\_Taxi\_NavBar';

export default class Shv\_Taxi\_Reservations\_Home\_Page extends Component {

  render() {

    return (

      <div>

       <Shv\_Taxi\_NavBar/>

       <center>

          <a href = '/TaxiAdminHome'><button type="button" class="btn btn-primary btn-lg btn-block">Taxi Admin Home Page</button></a><br/><br/>

          <a href = '/TaxiAdminClient'><button type="button" class="btn btn-primary btn-lg btn-block">Taxi Client Home Page</button></a><br/><br/>

          <a href = '/AdminTaxiReport'><button type="button" class="btn btn-primary btn-lg btn-block">Admin Taxi Reservation Report</button></a><br/><br/>

          </center>

      </div>

    )

  }

}