



Sri Lanka Institute of Information Technology

Distributed Systems (SE3020)

Assignment 02_Report

Train Ticketing Service

Submitted by: IT17100076

(SILVA.S.R.R.M)

Table of Contents

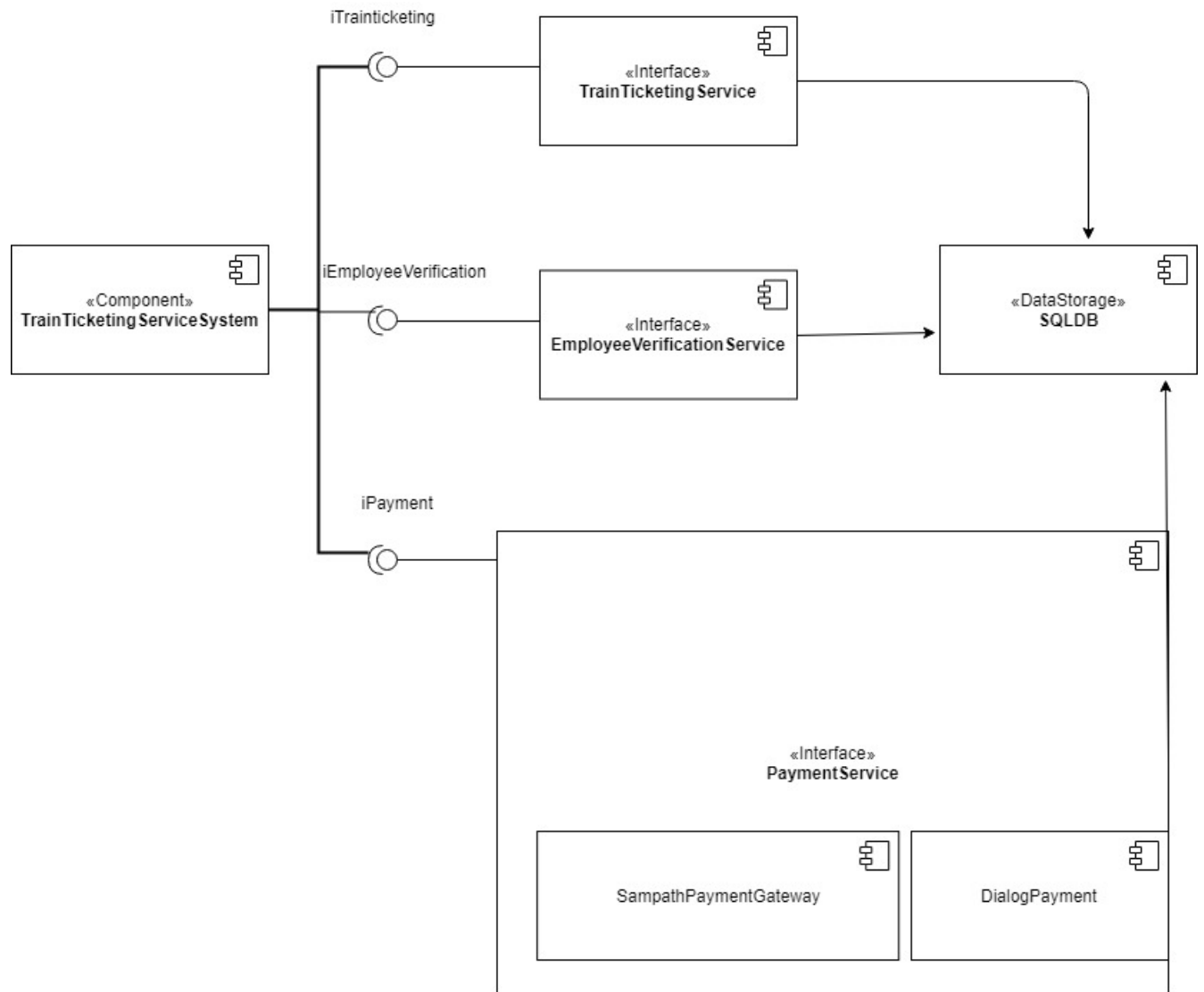
Introduction to Train Ticketing Service	3
High Level Architectural Diagram.....	4
System Workflow	5
System Interfaces	6
Appendix	
System Frontend (Client Side)	9
System Backend (Server Side)	16

Introduction to Train Ticketing Service

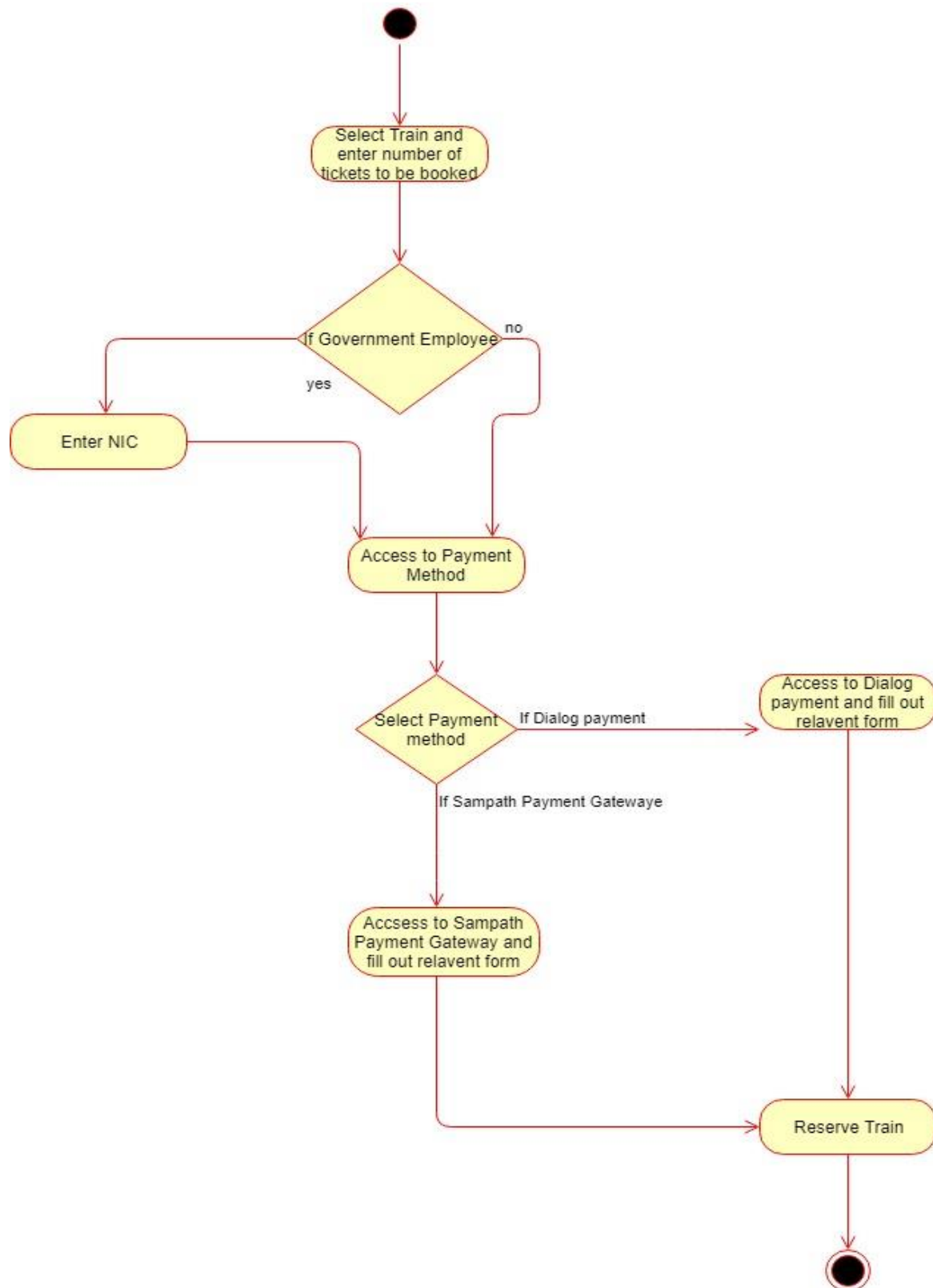
The 'Train Ticketing Service' web application has implemented using different kind of technologies. For the frontend implementation purpose, it is used React JS and for the backend implementation purpose it is used Node Js and to store data has used SQL DB.

The user can access this web application without any sign in or registration. There user can select their relevant train and number of tickets they need to be booked. Then system asks whether user is a government employee, if so, user have to verify it using their NIC. Once it verified by system it redirects to payment procedure. If user not government, user can directly move on to payment procedure. If user is a government one, he/she can get a discount. user must choose either Sampath bank payment gateway or dialog. According to payment method user must fill out relevant form details. Every payment method system is displayed their amount.

High Level Architectural Diagram



System Workflow



System Interfaces

Welcome to Train Ticketing Service

TICKET BOOKING

1 Train Info


Train

Colombo-Kandy

2 Number of Tickets Info

Number of Tickets to be booked

Next




WE MAKE YOUR LIFE SIMPLE AND FAST

Welcome to Train Ticketing Service

If you are Government Employee ?

Yes No




Welcome to Train Ticketing Service

We will be processed you a discount

NIC Number

Please Enter Your NIC Number

Payment Procedure



WE MAKE YOUR LIFE SIMPLE AND FAST


Welcome to Train Ticketing Service

GET READY TO YOUR PAYMENT PROCEDURE

Available Payment Methods

Sampath Bank Payment Gateway

Dialog Payment




WE MAKE YOUR LIFE SIMPLE AND FAST

7

Welcome to Train Ticketing Service

PAYMENT PROCESS IS ON GOING



සමපත් வங்கி
சம்பத் வங்கி
SampathBank

Sampath Bank Card Info

Credit Card Number

Please enter your valid Card Number

Name of Card Holder

Please enter your name

CVC Number

Amount


Rs.

Reserve Your Train

WE MAKE YOUR LIFE SIMPLE AND FAST

Welcome to Train Ticketing Service

PAYMENT PROCESS IS ON GOING



Dialog

Your In Dialog Payment Process

Mobile Number

PIN Number

Amount

Rs.

Reserve Your Train

WE MAKE YOUR LIFE SIMPLE AND FAST

Appendix

System Frontend (Client Side)

TrainHome.js

```
import React, {Component} from 'react';
import Homepic from './Homepic';
import {NavLink} from "react-router-dom";

class TrainHome extends Component{
  render() {
    return (
      <div >

        <div className='Homepic' >
          <div>
            <h1><center> <font color="white">Welcome to Train Ticketing Service</font></center></h1>
          </div>
        </div>

        <div className='div1' width="100%" height="100%" >

          <h4><font color="white"><center>TICKET BOOKING</center></font></h4>

          <div className="form-style-5">
            <form>
              <fieldset>
                <legend><span className="number">1</span> Train Info</legend>

                <label htmlFor="job">Train</label>
                <select id="job" name="field4">
                  <optgroup label="Intercity Train">
                    <option value="1">Colombo-Kandy</option>
                    <option value="2">Colombo-Galle</option>
                    <option value="3">Colombo-Jaffna</option>
                  </optgroup>
                  <optgroup label="Normal Train">
                    <option value="4">Panadura-Maradana</option>
                    <option value="5">Kaluthara-Kotuwa</option>
                    <option value="6">Negombo-Kotuwa</option>
                  </optgroup>
                </select>
              </fieldset>
              <fieldset>
                <legend><span className="number">2</span> Number of Tickets Info</legend>
                <textarea name="number" placeholder="Number of Tickets to be
booked"></textarea>
              </fieldset>
              <NavLink to="/employee" exact>

                <button type="submit" className="btn btn-primary">Next</button>
              </NavLink>

            </form>
            <Homepic/>
          </div>
          <div>
            <div className="blockquote-footer">
              <p>WE MAKE YOUR LIFE SIMPLE AND FAST</p>
            </div>
          </div>
        </div>
      );
    }
  }
}
export default TrainHome;
```

Employee.js

```
import React, {Component} from 'react';
import Homepic from './Homepic';
import logo from './employee.jpg'
import {NavLink} from "react-router-dom";

class TrainHome extends Component{
  render() {
    return (
      <div >

        <div className='Homepic' >
          <div>
            <h1><center> <font color="white">Welcome to Train Ticketing
Service</font></center></h1>
          </div>
        </div>

        <div className='div1' width="100%" height="100%" >

          <div className="form-style-5">
            <form>
              <legend><span className="number"></span> If you are Government Employee
?</legend><br/><br/>

              <NavLink to="/gemployee" exact>

                <button type="submit" className="btn btn-primary">Yes</button>
              </NavLink>

              <NavLink to="/Paymentmethods" exact>

                <button type="submit" className="btn btn-primary">No</button><br/><br/>
              </NavLink>
            </form>
            <img src={logo} className="rounded" width={500} height={300}/>
          </div>
          <div className="blockquote-footer">
            <p>WE MAKE YOUR LIFE SIMPLE AND FAST</p>
          </div>

        </div>

      </div>
    );
  }
}
export default TrainHome;
```

Gemplyee.js

[illegible]

Paymentmethods.js

```

import React, {Component} from "react";
import Homepic from './Homepic';
import {NavLink} from "react-router-dom";

class Paymentmethods extends Component{
  render() {
    return (
      <div>
        <div className='Homepic' >
          <div>
            <h1><center> <font color="white">Welcome to Train Ticketing
Service</font></center></h1>
            </div>
          </div>

          <div className='div1' width="100%" height="100%" >

            <h4><font color="white"><center>GET READY TO YOUR PAYMENT PROCEDURE</center></font></h4>

            <div className="form-style-5">
              <form>

                <legend><span className="number"></span> Available Payment
Methods</legend><br/><br/>

                <NavLink to="/sampathpayment" exact>

                  <button type="submit" className="btn btn-primary">Sampath Bank Payment
Gateway</button>

                </NavLink>

                <NavLink to="/dialogpayment" exact>

                  <button type="submit" className="btn btn-primary">Dialog
Payment</button><br/><br/>

                </NavLink>

              </form>
            </div>
            <Homepic/>

          </div>
        </div>
        <div className="blockquote-footer">
          <p>WE MAKE YOUR LIFE SIMPLE AND FAST</p>
        </div>
      </div>
    );
  }
}

export default Paymentmethods;

```

Sampathpayment.js

```
import React,{Component} from "react"
import Homepic from './Homepic';
import logo from './Sampath.jpg'
import {NavLink} from "react-router-dom";

class Sampathpayment extends Component {
  render() {
    return (
      <div>

        <div className='Homepic' >
          <div>
            <h1><center> <font color="white">Welcome to Train Ticketing
Service</font></center></h1>
          </div>

          <div className='div1' width="100%" height="100%" >

            <h4><font color="white"><center>PAYMENT PROCESS IS ON GOING</center></font></h4>

            <div className="form-style-5">
              <img src={logo} className="rounded" width={400} height={200}/>
              <form>
                <fieldset>
                  <legend><span className="number"></span>Sampath Bank Card Info</legend>

                  <label htmlFor="job">Credit Card Number</label>
                  <input type="text" placeholder="Please enter your valid Card Number"/>

                  <label htmlFor="job">Name of Card Holder</label>
                  <input type="text" placeholder="Please enter your name"/>

                  <label htmlFor="job">CVC Number</label>
                  <input type="text" />

                  <label htmlFor="job">Amount</label>
                  <input type="text" placeholder="Rs. "/>

                </fieldset>

                <NavLink to="/" exact>

                  <button type="submit" className="btn btn-primary">Reserve Your
Train</button><br/><br/>

                </NavLink>

              </form>

            </div>
          </div>
          <div className="blockquote-footer">
            <p>WE MAKE YOUR LIFE SIMPLE AND FAST</p>
          </div>
        </div>

      </div>
    );
  }
}

export default Sampathpayment;
```

Dialogpayment.js

```
import React,{Component} from "react"
import Homepic from './Homepic';
import {NavLink} from "react-router-dom";
import logo from './Dialog.png';

class Dialogpayment extends Component {
  render() {
    return (
      <div>

        <div className='Homepic' >
          <div>
            <h1><center> <font color="white">Welcome to Train Ticketing
Service</font></center></h1>
          </div>

          <div className='div1' width="100%" height="100%" >

            <h4><font color="white"><center>PAYMENT PROCESS IS ON GOING</center></font></h4>

            <div className="form-style-5">
              <img src={logo} className="rounded" width={400} height={200}/>
              <form>
                <fieldset>
                  <legend><span className="number"></span>Your In Dialog Payment
Process</legend>

                  <label htmlFor="job">Mobile Number</label>
                  <input type="number" />

                  <label htmlFor="job">PIN Number</label>
                  <input type="text" />

                  <label htmlFor="job">Amount</label>
                  <input type="text" placeholder="Rs. " />

                </fieldset>

                <NavLink to="/" exact>

                  <button type="submit" className="btn btn-primary">Reserve Your
Train</button><br/><br/>

                </NavLink>

              </form>

            </div>
          </div>
          <div className="blockquote-footer">
            <p>WE MAKE YOUR LIFE SIMPLE AND FAST</p>
          </div>
        </div>

      </div>

    );
  }
}

export default Dialogpayment;
```

App.js

```
import React, {Component} from 'react';
import 'bootstrap/dist/css/bootstrap.min.css';
import {BrowserRouter, Switch, Route, Link} from "react-router-dom";
import './App.css';
import TrainHome from './TrainHome';
import Gemployee from './Gemployee";
import Paymentmethods from './Paymentmethods"
import Sampathpayment from './Sampathpayment"
import Dialogpayment from './Dialogpayment"
import Employee from './Employee"

class App extends Component {
  render() {
    return (
      <BrowserRouter>
      <div>
        <Switch>
          <Route path="/" component={TrainHome} exact/>
          <Route path="/employee" component={Employee} exact/>
          <Route path="/gemployee" component={Gemployee} exact/>
          <Route path="/paymentmethods" component={Paymentmethods} exact/>
          <Route path="/sampathpayment" component={Sampathpayment} exact/>
          <Route path="/dialogpayment" component={Dialogpayment} exact/>
        </Switch>
      </div>
    </BrowserRouter>

    );
  }
}

export default App;
```

System Backend (Server Side)

```
package org.example.datalayerImpl;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import org.example.dbConnection.DBConnection;
import org.example.entities.Train;

public class TrainInfoImpl {

    public static List<Train> displayStarts() {
        String sql = null;
        ResultSet result = null;
        List<Train> trains = null;
        Train train = null;

        try {
            DBConnection dbcon = DBConnection.getInstance();
            Connection con = dbcon.getConnection();
            Statement st = con.createStatement();
            sql = "SELECT tid,amount,train from Train";
            result = st.executeQuery(sql);
            trains = new ArrayList<>();

            while (result.next()) {
                train = new
Train(result.getInt("tid"),result.getDouble("amount"),result.getString
("train"));
                train.setTid(result.getInt("tid"));
                train.setAmount(result.getDouble("amount"));
                train.setTrain(result.getString("trains"));
                trains.add(train);
            }

        }
        catch(SQLException ex){
            Logger.getLogger(TrainInfoImpl.class.getName()).log(Level.SEVERE
, null, ex);
        }

        finally {
            return trains;
        }
    }
}
```



```

package org.example.datalayerImpl;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import org.example.dbConnection.DBConnection;
import org.example.entities.Train;

public class TrainInfoImpl {
    public static List<Train> displayStarts() {
        String sql = null;
        ResultSet result = null;
        List<Train> trains = null;
        Train train = null;
        try {
            DBConnection dbcon = DBConnection.getInstance();
            Connection con = dbcon.getConnection();
            Statement st = con.createStatement();

            sql = "SELECT tid,amount,train from Train";

            result = st.executeQuery(sql);

            trains = new ArrayList<>();

            while (result.next()) {
                train = new
Train(result.getInt("tid"),result.getDouble("amount"),result.getString
("train"));
                train.setTid(result.getInt("tid"));
                train.setAmount(result.getDouble("amount"));
                train.setTrain(result.getString("trains"));
                trains.add(train);
            }
        }

        catch(SQLException ex){

            Logger.getLogger(TrainInfoImpl.class.getName()).log(Level.SEVERE
, null, ex);
        }

        finally {
            return trains;
        }
    }
}

```

```

package org.example.entities;

public class DialogPayment {

    int pid;
    double amount;
    int mobilenumbr;
    int pinnumber;
    public DialogPayment(int pid, double amount, int mobilenumbr,
int pinnumber) {
        super();
        this.pid = pid;
        this.amount = amount;
        this.mobilenumbr = mobilenumbr;
        this.pinnumber = pinnumber;
    }
    public int getPid() {
        return pid;
    }
    public void setPid(int pid) {
        this.pid = pid;
    }
    public double getAmount() {
        return amount;
    }
    public void setAmount(double amount) {
        this.amount = amount;
    }
    public int getMobilenumbr() {
        return mobilenumbr;
    }
    public void setMobilenumbr(int mobilenumbr) {
        this.mobilenumbr = mobilenumbr;
    }
    public int getPinnumber() {
        return pinnumber;
    }
    public void setPinnumber(int pinnumber) {
        this.pinnumber = pinnumber;
    }
}

```

```

package org.example.entities;

public class Train {

    private int tid ;
    private String train;
    private double amount;

    public Train(int tid, double amount, String train) {
        super();
        this.tid = tid;
        this.train = train;
        this.amount = amount;// TODO Auto-generated constructor
stub
    }
    public int getTid() {
        return tid;
    }
    public void setTid(int tid) {
        this.tid = tid;
    }
    public String getTrain() {
        return train;
    }
    public void setTrain(String train) {
        this.train = train;
    }
    public double getAmount() {
        return amount;
    }
    public void setAmount(double amount) {
        this.amount = amount;
    }
}

```

```

package org.example.entities;

public class User {
    int uid;
    int pid;
    String NIC;
    String email;
    int phonenumber;
    public User(int uid, int pid, String nIC, String email, int
    phonenumber) {
        super();
        this.uid = uid;
        this.pid = pid;
        NIC = nIC;
        this.email = email;
        this.phonenumber = phonenumber;
    }
    public int getUid() {
        return uid;
    }
    public void setUid(int uid) {
        this.uid = uid;
    }
    public int getPid() {
        return pid;
    }
    public void setPid(int pid) {
        this.pid = pid;
    }
    public String getNIC() {
        return NIC;
    }
    public void setNIC(String nIC) {
        NIC = nIC;
    }
    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
    public int getPhonenumber() {
        return phonenumber;
    }
    public void setPhonenumber(int phonenumber) {
        this.phonenumber = phonenumber;
    }
}

```

```

package org.example.service;

import java.awt.PageAttributes.MediaType;
import java.util.List;

import javax.ws.rs.DELETE;
import javax.ws.rs.GET;
import javax.ws.rs.NotFoundException;
import javax.ws.rs.OPTIONS;
import javax.ws.rs.POST;
import javax.ws.rs.PUT;
import javax.ws.rs.Path;
import javax.ws.rs.PathParam;
import javax.ws.rs.Produces;
import javax.ws.rs.core.Context;
import javax.ws.rs.core.Response;

import org.example.datalayerImpl.TrainInfoImpl;
import org.example.entities.Train;
import org.wso2.ms4j.Request;

public class TrainService {

    List <Train> trains;

    @GET
    @Path("/get-train")
    @Produces(MediaType.APPLICATION_JSON)
    public List<Train> getTrain(@Context Request request, @Context
Response response) {
        response.setHeader("Access-Control-Allow-Origin",
request.getHeader("Origin"));
        System.out.println("GET Train");
        List<Train> trainList = TrainInfoImpl.displayStarts();

        return trainList;
    }

    @POST
    public Train addTrain(Train newTrain) {
        trains.add(newTrain);
        for(Train t:trains) {
            if(t.getTid() == newTrain.getTid())
                return t;
        }
        throw new NotFoundException();
    }
}

```

```

@PUT
@Path("/")
public void put() {
    // TODO: Implementation for HTTP PUT request
    System.out.println("PUT invoked");
}

@Path("/{id}")
@DELETE
@Produces(MediaType.APPLICATION_JSON)
public Train deleteTrain(@PathParam("id") String train) {
    for(Train t : trains) {
        if(t.getTid() == train) {
            trains.remove(t);
            return t;
        }
    }
    throw new NotFoundException();
}

@Path("/get-train")
@OPTIONS
@Produces(MediaType.APPLICATION_JSON)
public javax.ws.rs.core.Response getResponse(@Context
Request request) {
    javax.ws.rs.core.Response response =
    javax.ws.rs.core.Response.status(200)
        .header("Access-Control-Allow-Origin",
request.getHeader("Origin"))
        .header("Access-Control-Allow-Methods",
"GET, POST, PUT, DELETE")
        .header("Access-Control-Allow-Headers",
"content-type")
        .build();

    return response;
}
}

```