**\*Royal Railways**

* **Introduction**

Website aims to booking train tickets, this system helps to reduce the time, effort and cost to the customer.

Determines the way that the customer will are used by the train.

Easy way for all those who want to travel within the kingdom.

The system will enable the customer from modification, deletion and print booking, and ensures that the system does not meet the trains to avoid any problems.

**2. Project Description**

**2.1 Purpose**

The purpose of this project provides a detailed description of booking train tickets. It will explain the purpose and features of the system, and interfaces to the system, the system helps customers to book tickets, how the restrictions would work and how the system will react to external factors.

**2.2 Project Objectives**

1. Confirmation the reservation after the completion of the last steps of the booking and allows the customers to print copy of confirmation.

2. Safe time, effort and reduce traffic problems .

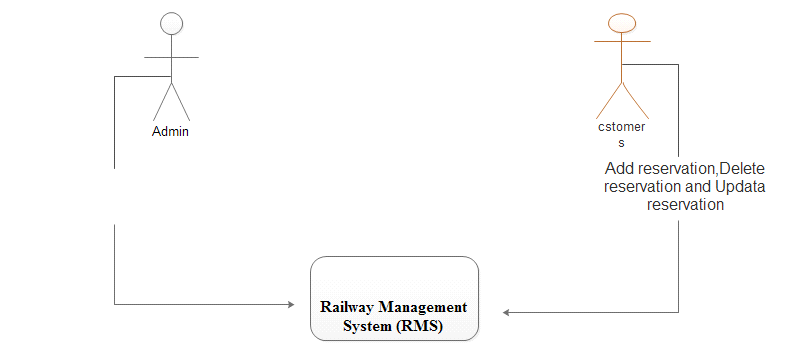
3. Reduce paperwork and electronic mails that registration department received.

4. Facilitate administrative processes and reduce the number of employees.

5. Reduce time and cost.

**3. Overall Description**

**3.1 System Environment**



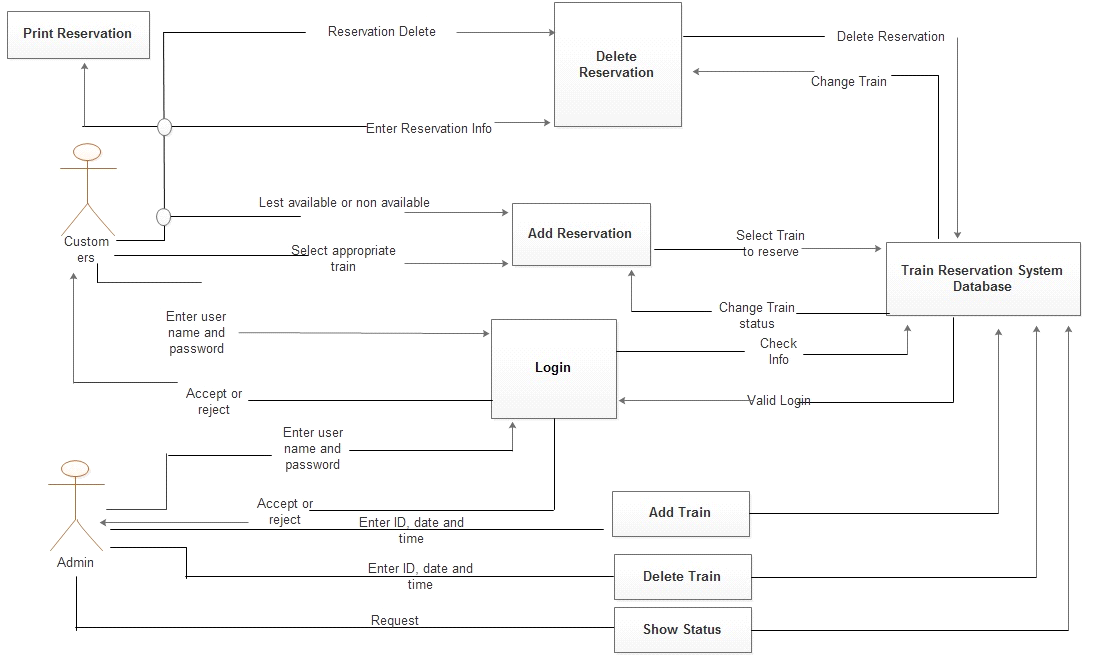
Add Train, Delete train, Show status, Print Reservation



**Figure1 - System Environment drawn using Edraw Max**

The Web System has two active actors .The Admin and customer accesses the database through the Internet. The admin accesses the entire system directly.

**3.2 Data Flow Diagram (DFD) "drawn using Edraw max tool**



**4. Functional and Nonfunctional Requirements Specifications.**

This section outlines the use cases drawn using Edraw Max for each of the active user separately. The admin and the customers have only one use case apiece.

**4.1 Functional Requirements:**

* The system shall be allowing the customers to create an account.
* The system shall be allowing the customers to add a reservation.
* The system shall be allowing the customers to delete a reservation.
* The system shall be allowing the customers to print the reservation.
* The system show the customers all of reservations assigned to hem\her in a particular date and time.
* The system shall be allowing the admin to adding new train.
* The system shall be allowing the admin to delete train.
* The system shall be allowing the admin to confirm reservation.
* The system shall be allowing the admin to print reservation.

**4.2 Nonfunctional Requirements:**

1) The system must be secure for both costumers and admins.

2) The system must be fast to save users time.

**5. Admin Use Cases**

**5.1 Login**



|  |  |
| --- | --- |
| **Use case name** | Login. |
| **Trigger** | The admin must click on the login button. |
| **Precondition** | \_\_\_ |
| **Basic Paths** | 1. The admin open the home page.  2. The admin enters the identity information.  3. The admin select login button. |
| **Actor** | Admin. |
| **Alternative Paths** | \_\_\_ |
| **Post Condition** | The admin logged in. |
| **Exception Path** | \_\_\_ |

**5.2 Add Train**



|  |  |
| --- | --- |
| **Use case name** | Add train |
| **Trigger** | The admin must click on the add train button. |
| **Precondition** | The admin must has logged in to the system then add train. |
| **Basic Paths** | 1. The admin select to add train.  2. The admin insert train information.  3. The admin select ok bottom. |
| **Actor** | Admin. |
| **Alternative Paths** | \_\_\_\_ |
| **Post Condition** | The station added. |
| **Exception Path** | \_\_\_ |

**5.3 Delete Station**



|  |  |
| --- | --- |
| **Use case name** | Delete Station |
| **Trigger** | The admin must click on the delete station button. |
| **Precondition** | The admin must has logged in to the system then remove station from the Database. |
| **Basic Paths** | 1. The admin select to delete station.  2. The system provides a list of station.  3. The admin select station to be deleted. |
| **Actor** | Admin. |
| **Alternative Paths** | \_\_\_ |
| **Post Condition** | The station deleted. |
| **Exception Path** | \_\_\_ |

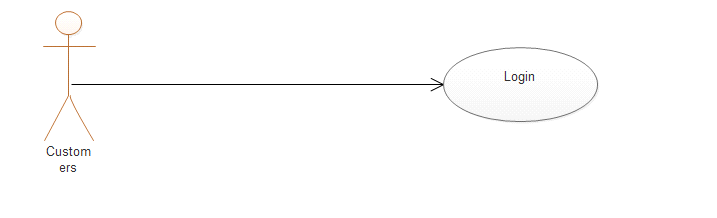
**5.4 Logout**



|  |  |
| --- | --- |
| **Use case name** | Logout. |
| **Trigger** | The admin must click on the logout button. |
| **Precondition** | The admin must logged in to the online system website. |
| **Basic Paths** | The admin click on logout button. |
| **Actor** | Admin. |
| **Alternative Paths** | \_\_\_ |
| **Post Condition** | The admin logged out. |
| **Exception Path** | \_\_\_ |

**6. Customers Use Cases**

**6.1 Login**



|  |  |
| --- | --- |
| **Use case name** | Login. |
| **Trigger** | The customers must click on the login button. |
| **Precondition** | \_\_\_ |
| **Basic Paths** | 1. The customers open the home page.  2. The customers enters the identity information.  3. The customers click on login bottom. |
| **Actor** | Customers. |
| **Alternative Paths** | \_\_\_ |
| **Post Condition** | The customers logged in. |
| **Exception Path** | \_\_\_ |

**6.2 Add Reservation**



|  |  |
| --- | --- |
| **Use case name** | Add Reservation. |
| **Trigger** | The customers click on the add reservation button. |
| **Precondition** | The customers logged in to the system then adds a reservation. |
| **Basic Paths** | 1. The customers select to add reservation.  2. The system provides a list of the dates of the trip.  3. The customers selects the trip  4. The system save the reservation. |
| **Actor** | Customers |
| **Alternative Paths** | \_\_\_ |
| **Post Condition** | The reservation added. |
| **Exception Path** | \_\_\_ |

**6.3 Delete Reservation**



|  |  |
| --- | --- |
| **Use case name** | Delete Reservation. |
| **Trigger** | The customers must click on the delete reservation button. |
| **Precondition** | The customers must logged in to the system then removes reservation from the Database. |
| **Basic Paths** | 1. The customers selects to delete reservation.  2. The system provides returns reservation information.  3. The customers deletes reservation. |
| **Actor** | Customers |
| **Alternative Paths** | \_\_\_ |
| **Post Condition** | The reservation deleted. |
| **Exception Path** | \_\_\_ |

**6.4 Print Reservation**



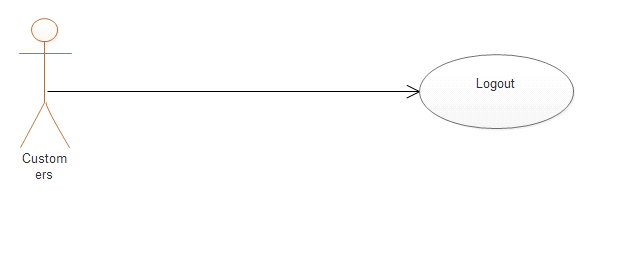
|  |  |
| --- | --- |
| **Use case name** | Print Reservation. |
| **Trigger** | The customers click on the print button. |
| **Precondition** | The customers must login to the system then print the reservation or report. |
| **Basic Paths** | 1. The customers adds a reservation.  2. The customers selects to print a copy of reservation. |
| **Actor** | Customers. |
| **Alternative Paths** |  |
| **Post Condition** | The reservation or report printed. |
| **Exception Path** | \_\_\_ |

**6.5 Change Reservation**



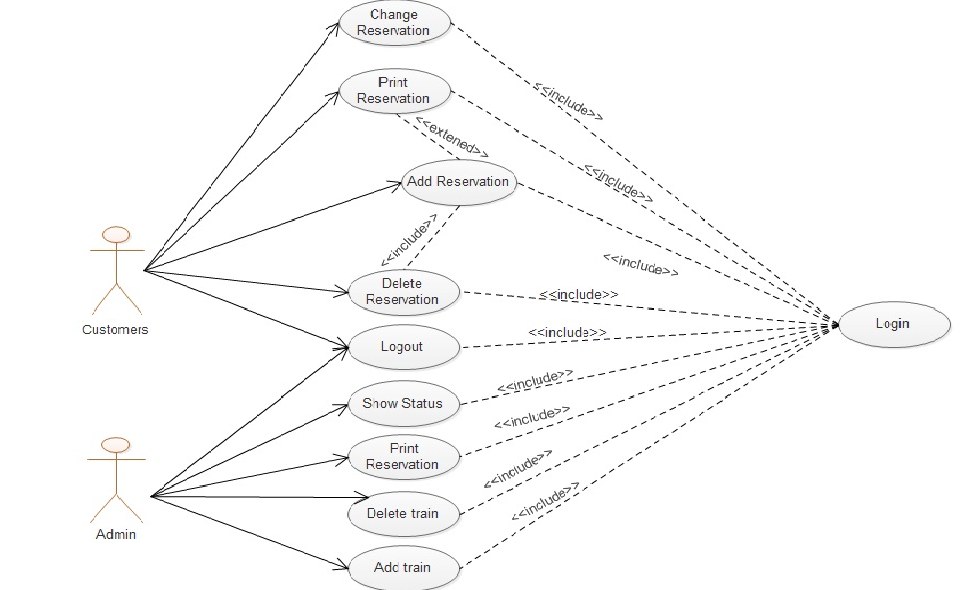
|  |  |
| --- | --- |
| **Use case name** | Change Reservation. |
| **Trigger** | The customers click on the change button. |
| **Precondition** | The customers must login to the system then print the reservation or report. |
| **Basic Paths** | 1. The customers change a reservation.  2. The customers selects to change reservation. |
| **Actor** | Customers. |
| **Alternative Paths** | \_\_\_ |
| **Post Condition** | The reservation changed.. |
| **Exception Path** | \_\_\_ |

**6.6 Logout**



|  |  |
| --- | --- |
| **Use case name** | Logout. |
| **Trigger** | The customers must click on the logout button. |
| **Precondition** | The customers must login to the system. |
| **Basic Paths** | 1. The customers clicks on logout button. |
| **Actor** | Customers. |
| **Alternative Paths** | \_\_\_ |
| **Post Condition** | The customers logged out. |
| **Exception Path** | \_\_\_ |

**7. The System Use Cases drawn using Edraw Max**



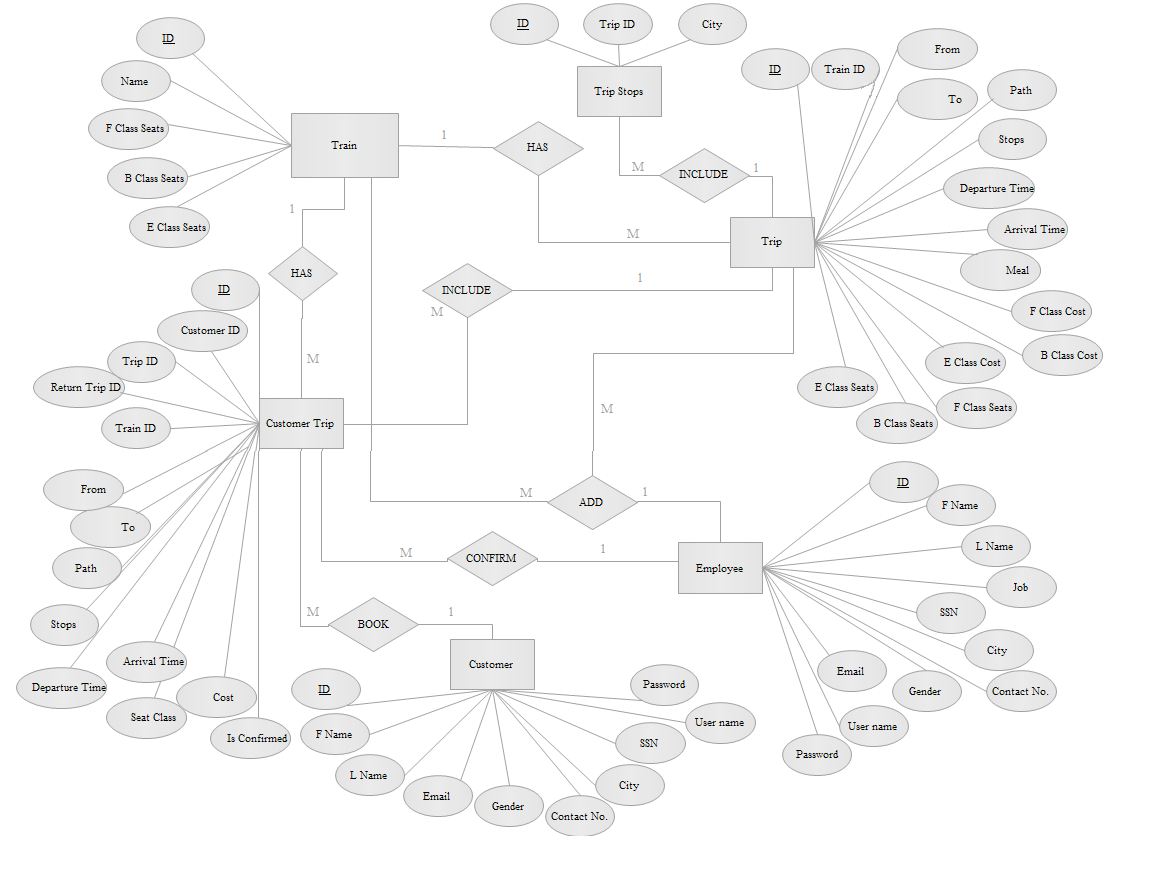
**Figure3-System Use Case drawn using Edraw max tool**

**\*Phase Two**

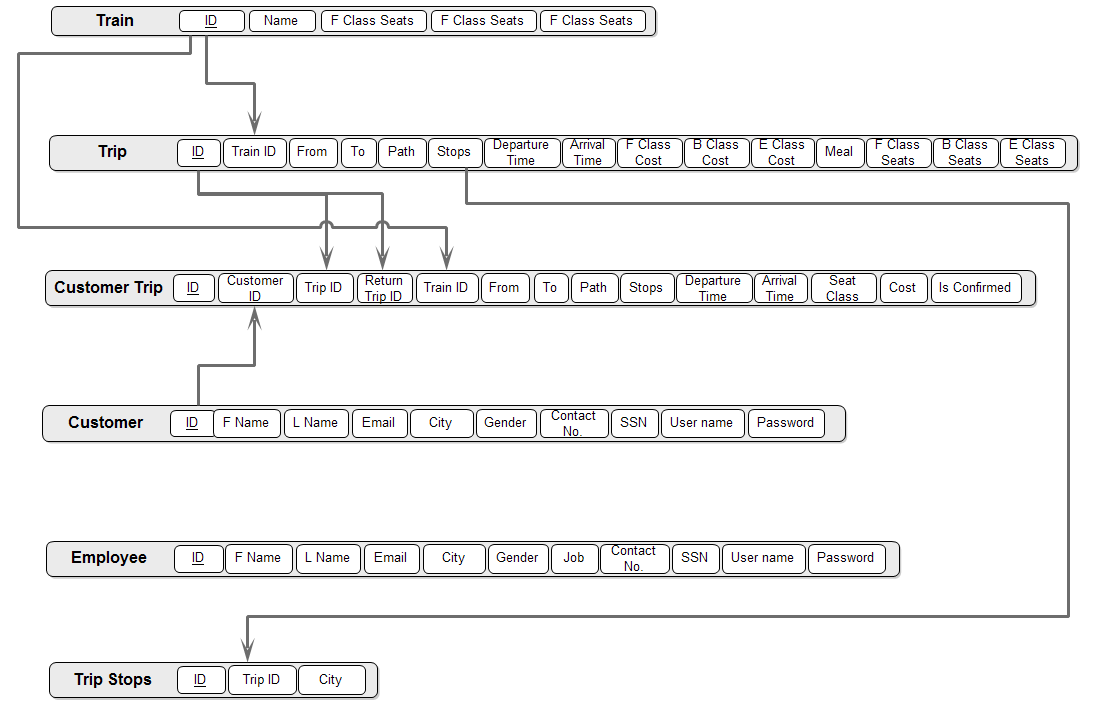
1. **Introduction**

This is phase two of our project, “Royal Railways” system, which covers the ER-Diagram of the system for more clearance view of the system functionality, and includes the Relation Schema between the database tables, which shows the primary and foreign keys, finally this work will cover all table fields and indexes.

1. **ER-Diagram**

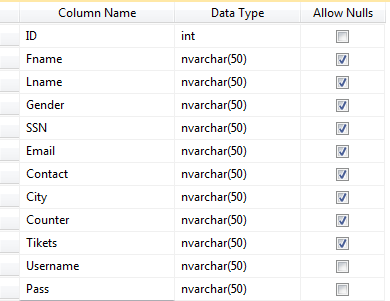
****

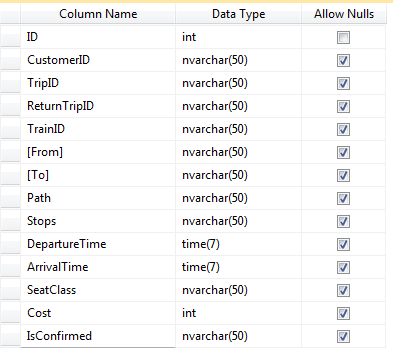
1. **Relation Schema**

****

1. **Tables, Fields and Indexes**

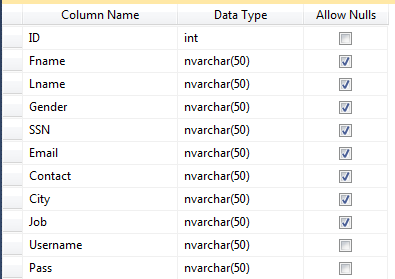
Customer Table

****

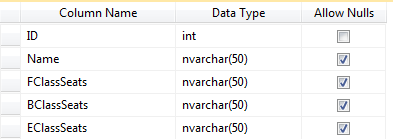
****

Customer Trip

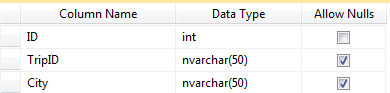
Table

****

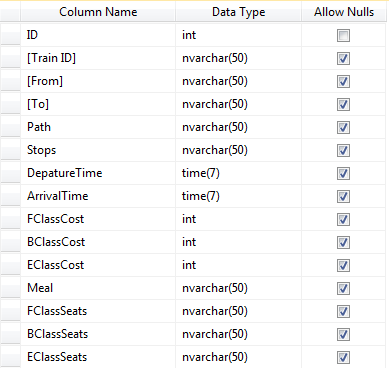
Employee Table

****

Train Table

****

Trip Stops Table

****

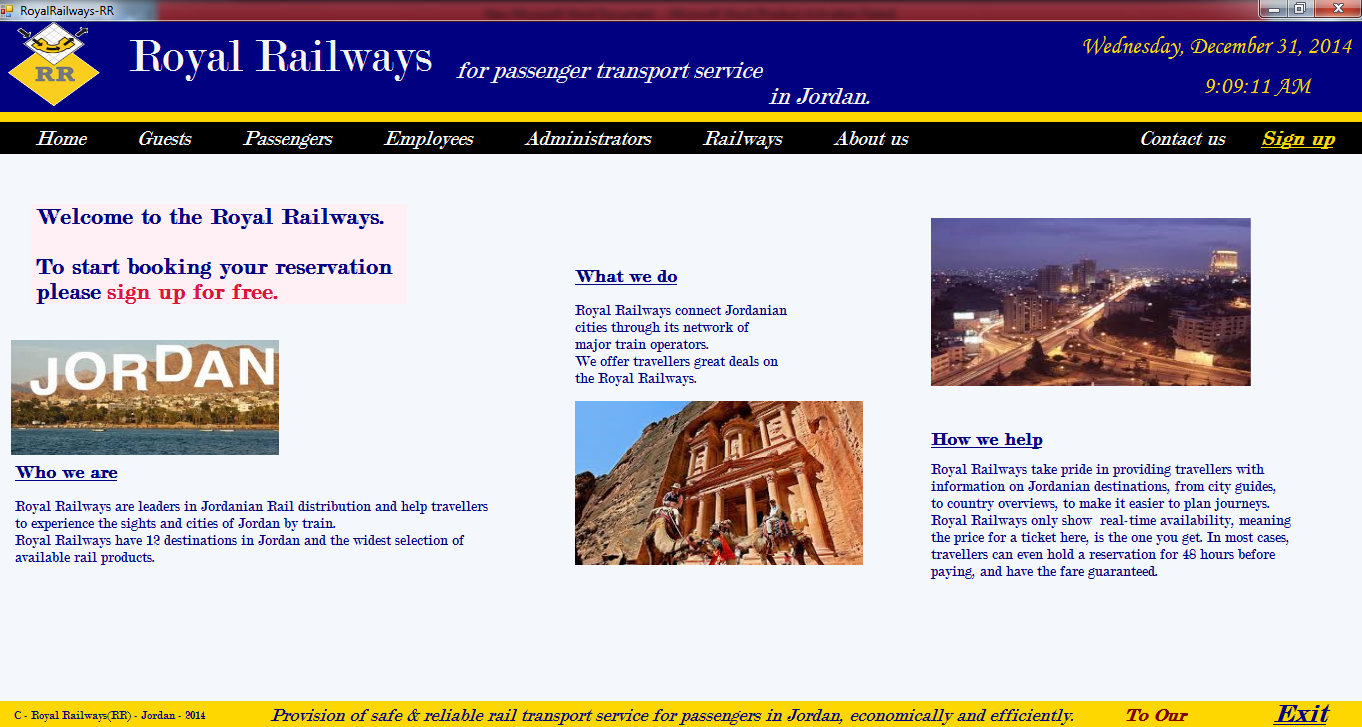
Trip Table

**Screen shoots**

**Home page**



**Guest page**



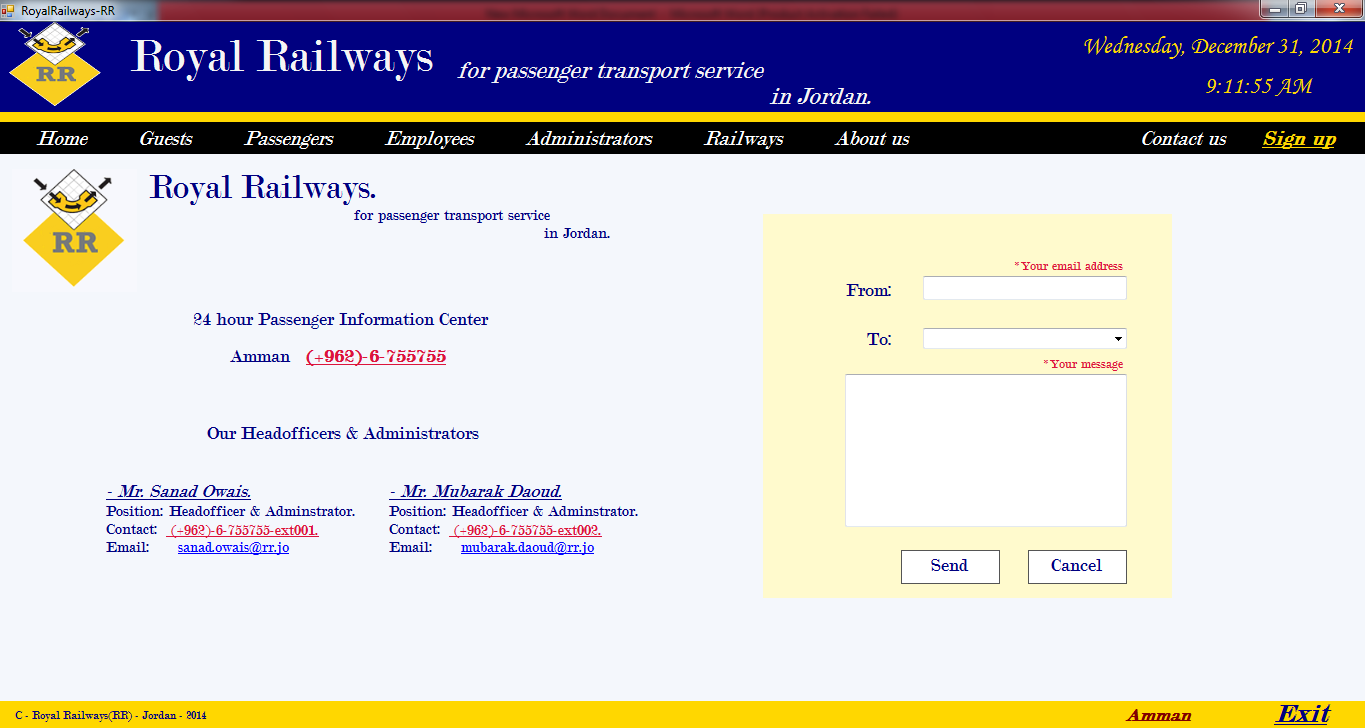
**Railway maps page**



**About us page**



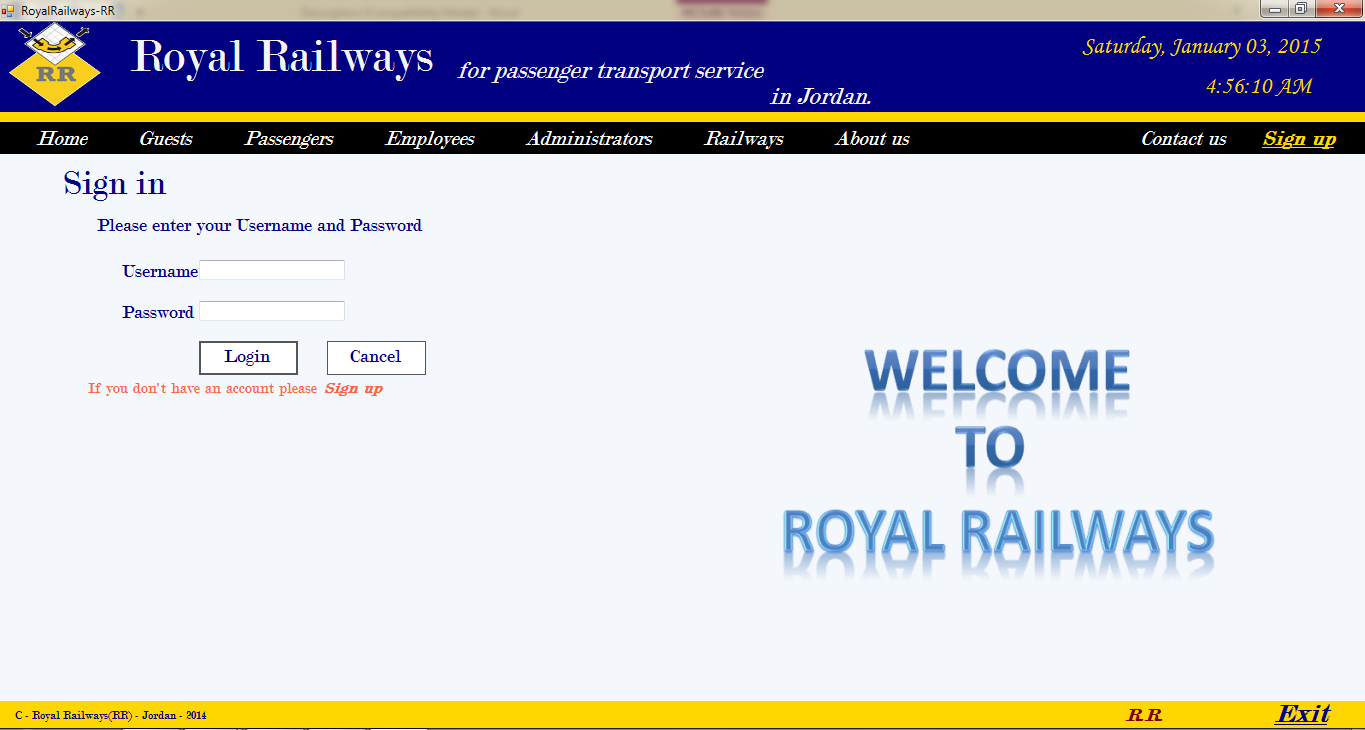
**Contact us page**



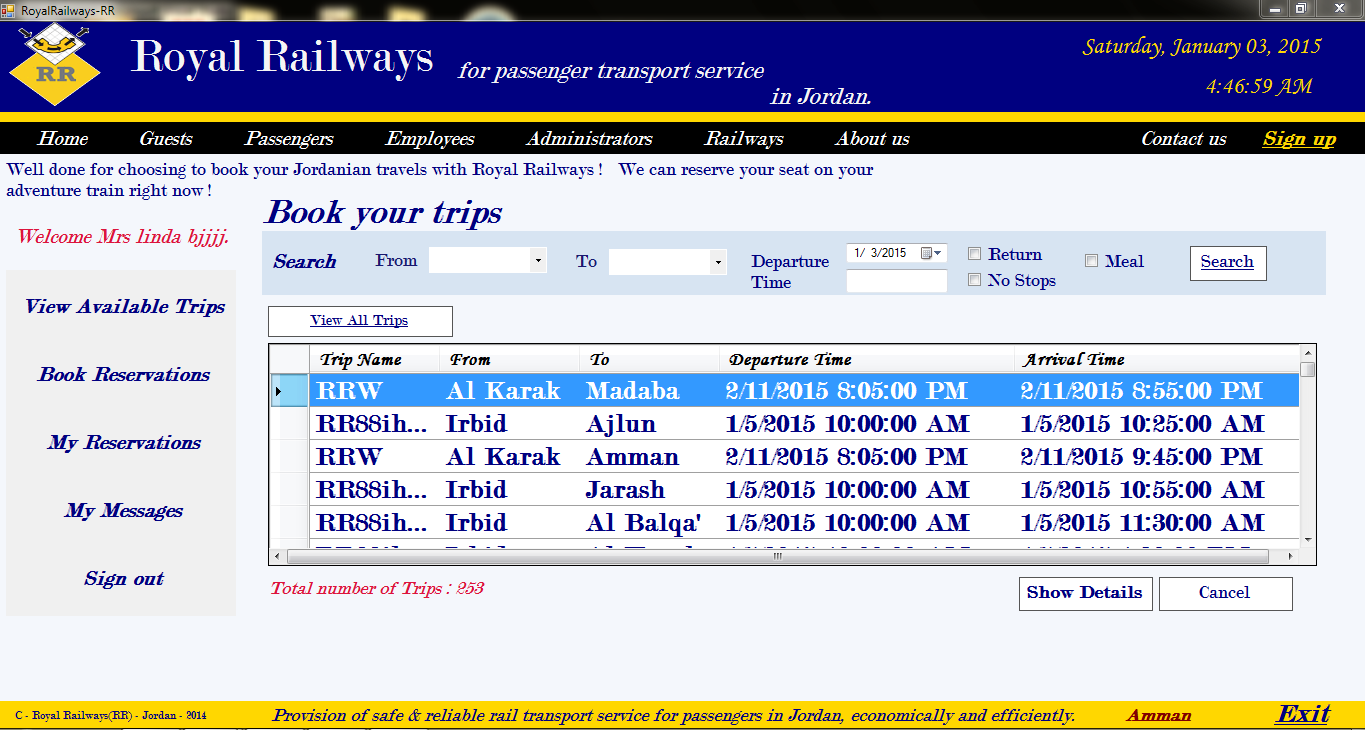
**Signup page**



**Signin page**



**Passenger View Trips page**



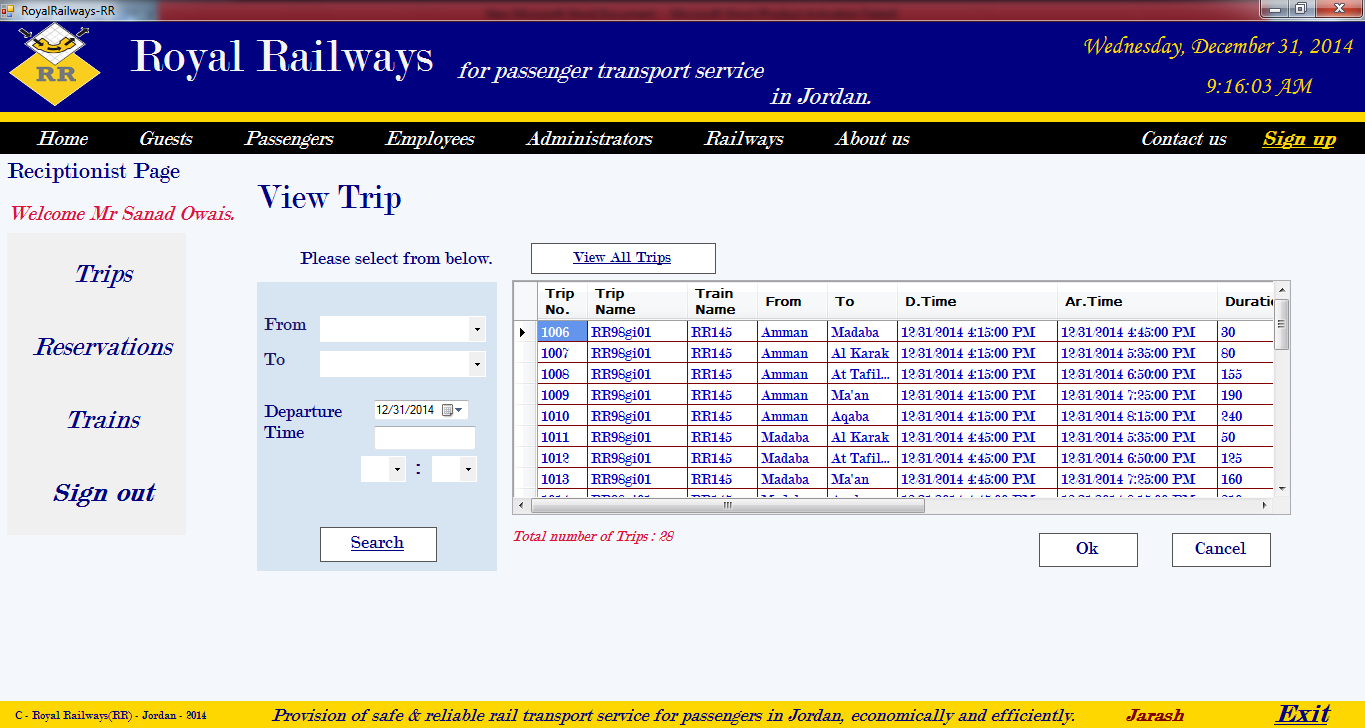
**Passenger Book Trips page**



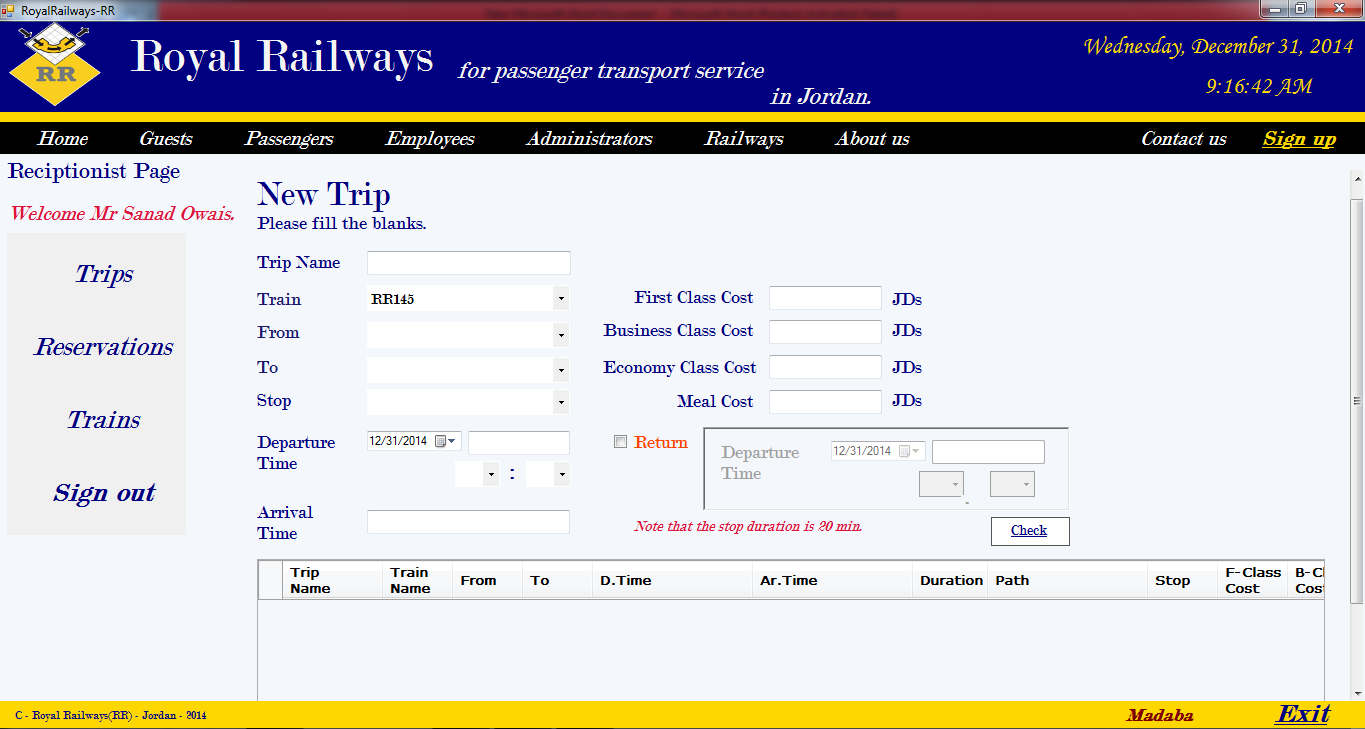
**Passenger Confirm Trips page**



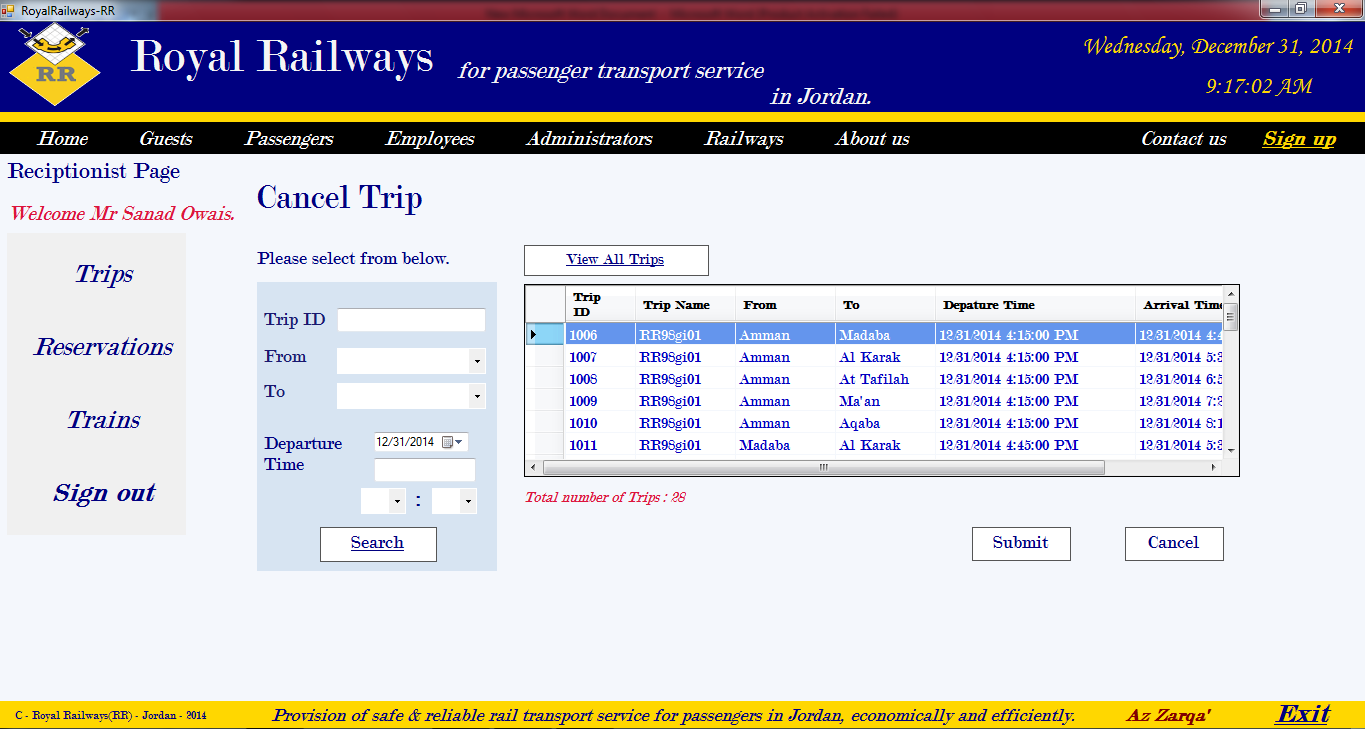
**Employee & Admin View Trips page**



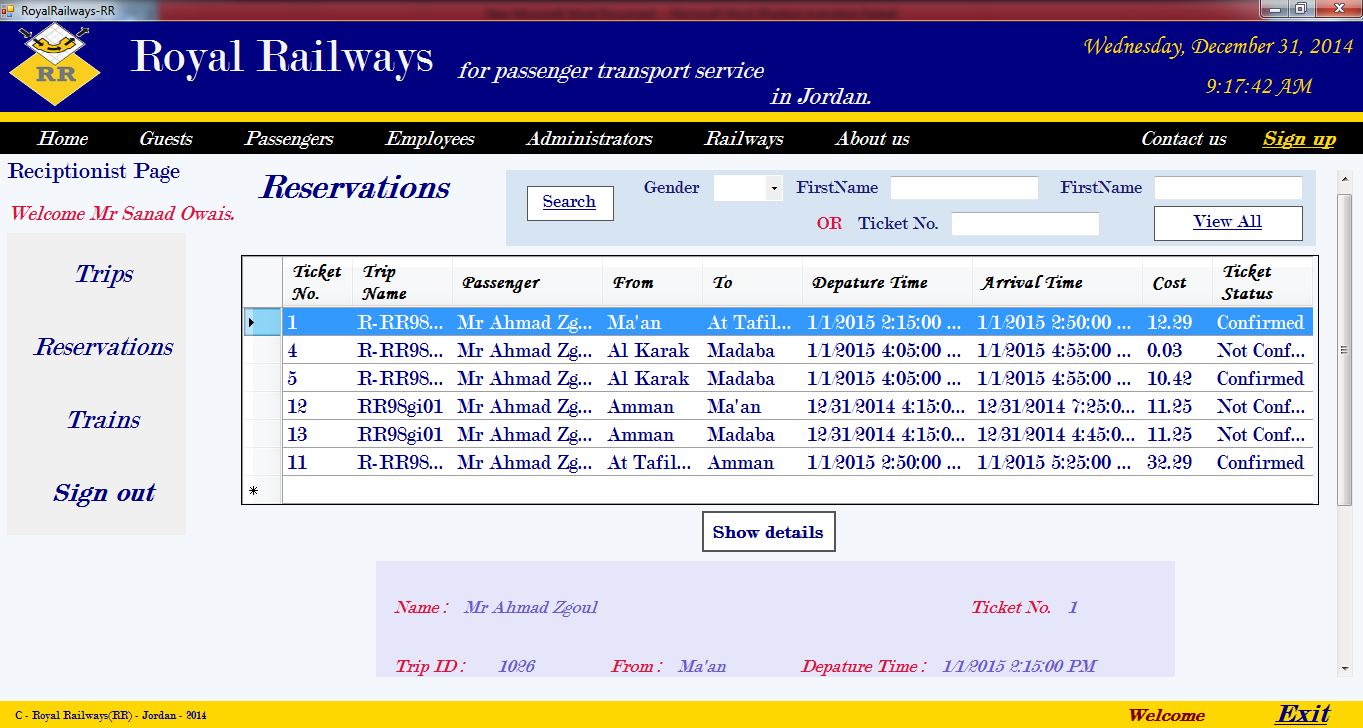
**Employee & Admin Add Trips page**



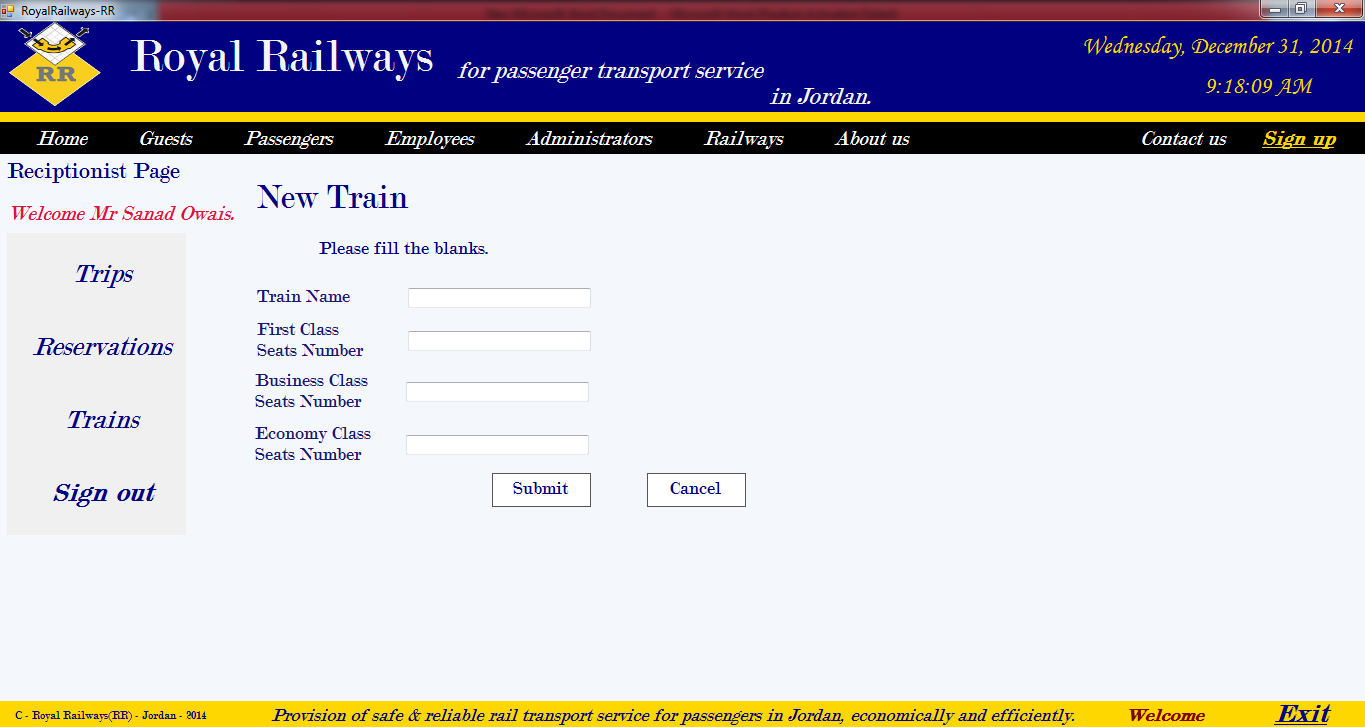
**Employee & Admin Cancel Trips page**



**Employee & Admin Confirm Trips page**



**Employee & Admin Add Train page**



**Employee & Admin Delete Train page**



**Admin Add Employee page**



**Admin Delete Employee page**

