**Train Ticket Management System**



**Submitted to**

**Mr. Muhammad Fawad**  
(Head of Software Engineering)

**Submitted by**

Abdul Samad (**48865**)  
Shakeel Ahmad (**48237**)

**School of Computing and Innovation**Riphah International University, Lahore Campus.

**[December 11, 2023]**

1. **Introduction:**

This project consists of train ticket management system. It is a computerized system that is being developed for online seat booking of passengers.

In this, the passenger will get informed about the trains availability according to the passenger's source to destination, get the available train routes along with prices and also book or cancel a ticket to start their journey efficiently. And an online ticket will be issued on which the passenger's name and seat number will be recorded.

1. **Objectives:**

* Create a user-friendly Online System through which User can generate ticket of train.
* Facilitate passengers in specifying their departure and destination locations, ensuring accurate travel information for the reservation system.
* Generate online tickets automatically upon successful reservation, including crucial details such as the passenger's name and assigned seat number.

1. **Problem Description: --------- [i]**

The Train Ticket Reservation System is basically developed in order to perform Online ticket reservation and enquiry. Every time a ticket is booked there must be a proper record to be maintained and also the ticket status, availability status must be updated time to time. There may be any new addition of trains and train being deleted or a station added or deleted which must be updated prior in the database.

The reservation option enables a person to reserve for a ticket in an easier and efficient manner. Once the train is available the person will be asked to enter his personal details and his requirements regarding the coach and berth. After this the reservation database is updated with the person details, train name and also the source and destination place and a unique ID number is generated and given as the reference number for the user. The availability status is changed according to the number of tickets booked.

The enquiry option allows the user to enter the ID number and get the current details about the ticket. The software ensures safety and security of details about the passengers, his payment details and ensures integrity of the data being processed.

1. **Features:**

* Allow users to create accounts and authenticate their identity for secure access.
* Users will be able to search the train & seat availability.
* Pay the ticket dues by using the debit, credit or master card, payoneer.
* Enable users to update their profiles and manage personal information.
* System will send the notification before 2 hour from the departure of train.
* Maintain Passenger Travel backup.
* System work on Cross-Platform.

1. **Users:**

The Train Ticket Management System is designed to serve various stakeholders within the transportation and travel industry.

* Admin
* Passengers
* Finance and Accounting Personnel
* Payment service provider
* System Developers and IT Staff
* Ticket collector

1. **Requirements:**
2. User will locate the destination. **[1]**
   1. Select target location within Pakistan. **[1a]**
   2. Choose the feasible path and train in required time by user. **[1b]**
3. Passenger will Register/Login to reserve the seat. **[2]**
   1. Passenger register personal information. (name, age, sex, phone number, CNIC number). **[2a]**
   2. Passenger can view profile, update profile and change password. **[2b]**
   3. User can view notification(s). **[2c]**
4. Book the ticket by entering date and time. **[3]**
5. Cancel the ticket by giving the valid reason. **[4]**
6. Pay invoice online payment gateways. **[5]**
7. Admin can approve or reject ticket. **[6]**
8. Admin can view records and see history of any passenger. **[7]**
   1. Admin can’t delete user history. **[7a]**
9. Admin can schedule, add, update, delete and view train(s). **[8]**
10. Admin can notify the passenger for departure time 2 hours before. **[9]**
11. Admin, User, and Ticket collector get access cross-platform. **[10]**
    1. Admin, User, Ticket collector are dependable on system information. **[10a]**
    2. The system should be available 24/7 to accommodate every type of user. **[10b]**
12. Ability to recover data in case of system failures or disasters. **[11]**
13. User can interact easily with responsive and user-friendly UI-UX design. **[12]**
14. **Constraint:**
15. Train ticket reservation require internet to make online booking. **[C1]**
16. System required any type of device for use. Without device we can’t use this system. **[C2]**
17. System performance may vary based on the geographical distribution of users and servers. **[C3]**
18. **Domain Requirements:**
19. Dynamic fare calculation based on factors such as distance, class, and time of booking, and government taxes. **[D1]**
20. Can’t cancel ticket after due date or train departure. **[D2]**

**Functional Requirements Description:**

|  |  |  |  |
| --- | --- | --- | --- |
| Functional Req. ID # | Function Name | Functional Requirement Description | Software Requirement |
| Fr1 | **Search Location** | Set destination point | **[1]** |
| Fr2 | **Create Account** | Create accounts and authenticate their user identity | **[2], [2a]** |
| Fr3 | **Update Profile** | Edit, delete, and update password | **[2b]** |
| Fr4 | **Book ticket** | Book ticket for specific time and date | **[3]** |
| Fr5 | **Cancel ticket** | Cancel ticket by valid reason | **[4]** |
| Fr6 | **Payment** | Through payment gateways | **[5]** |
| Fr7 | **Approve / Cancel** | Admin can approve/ Cancel ticket | **[6]** |
| Fr8 | **History maintain** | Admin can’t delete user history | **[7a]** |
| Fr9 | **Admin responsibility** | Schedule , add, delete, view train | **[8]** |
| Fr10 | **Notification** | Passengers get Notification tab | **[2c]** |
| Fr11 | **Notify Passengers** | Send reminder message of train | **[9]** |

**Non-functional Requirements Description:**

|  |  |  |  |
| --- | --- | --- | --- |
| Non-Functional Req. ID # | Function Name | Non-Functional Requirement Description | Software Requirement |
| Nfr1 | **Across platform** | Access from cross-platform | **[10]** |
| Nfr2 | **Dependable Software** | Admin, ticket collector are dependable | **[10a]** |
| Nfr3 | **Availability** | System is available all time | **[10b]** |
| Nfr4 | **Data recovery** | Backup user data | **[11]** |
| Nfr5 | **User-Friendly UI UX** | Easily navigate by users | **[12]** |

**Constraint Requirements Description:**

|  |  |  |  |
| --- | --- | --- | --- |
| Constraint Req. ID # | Function Name | Constraint Requirement Description | Software Requirement |
| C1 | **Internet connection** | System require internet for working | **[C1]** |
| C2 | **Device need** | Must have device to use. | **[C2]** |
| C3 | **Geo-graphical Performance** | System performance vary. | **[C3]** |

**Constraint Requirements Description:**

|  |  |  |  |
| --- | --- | --- | --- |
| Constraint Req. ID # | Function Name | Constraint Requirement Description | Software Requirement |
| D1 | **Fare calculation** | Amount calculation due tax changing by government | **[D1]** |
| D2 | **Ticket canceling policy** | Can’t cancel ticket after date passed | **[D2]** |

1. **References:**

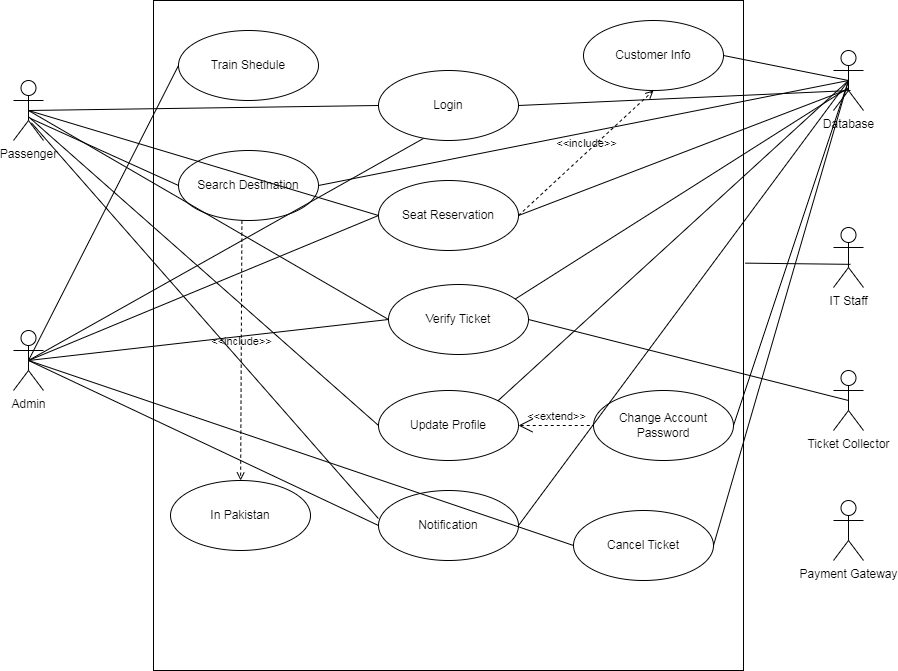
We took inspiration from Orange line train system <https://pakrail.gov.pk/>

**[i]**  <https://www.scribd.com/document/482566472/218168478>

Google search is included for research in this project

**Diagrams**

1. **Use-Case Analysis:**
   1. **Use case model.**

****

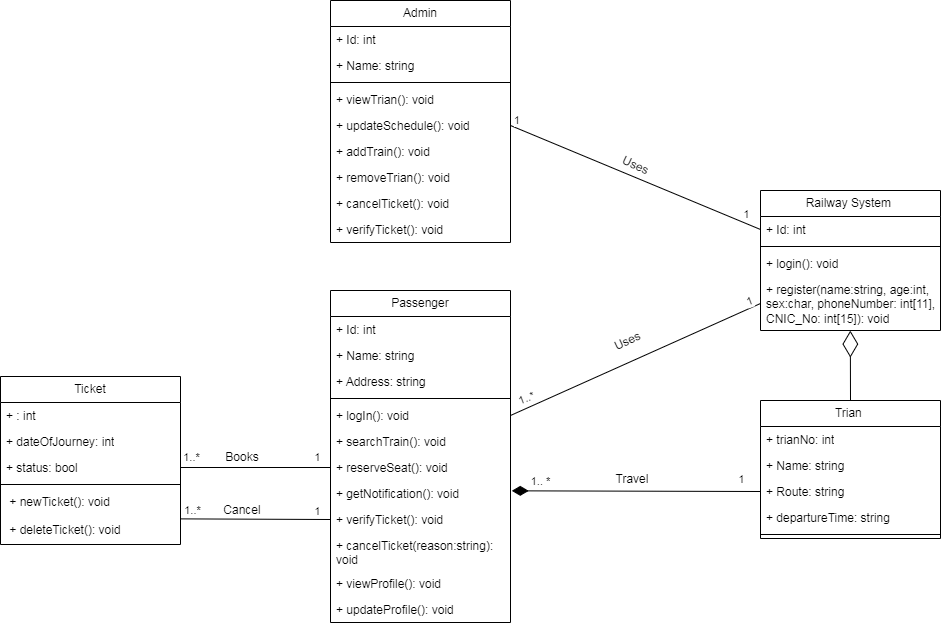
**Figure # 01**

* 1. **Use case model:**

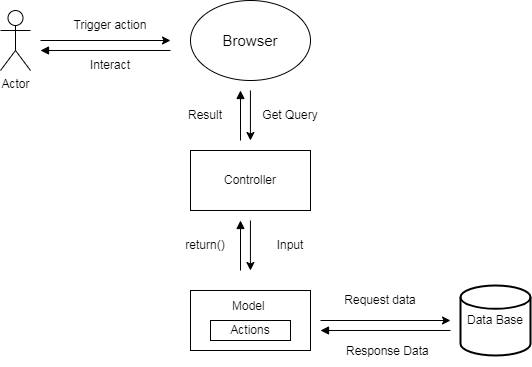
|  |  |
| --- | --- |
| Use Case | Reserve Train Seat |
| Description | This use case starts when Passenger wants to reserve a Railway ticket from the system. To do that Passenger must log in. If account is registered the system validates the login details, if the validation succeeds the Passenger can secure seat & get ticket. |
| Primary Actor | Passenger (New User, old user) |
| Goal | The Passenger is able to generate the ticket & reserve seat. |
| Pre-Condition | * The passenger has a valid account with the system. * The passenger has the permissions to reserve seat & generate ticket. |
| Post-Condition | * The ticket has been generated for Passenger. * The seat has been reserved for him. |
| Trigger | The Passenger initiate by filling the form on the main page. |
| Scenario | 1. The user login to the system. 2. The Passenger locate the form to reserve the seat & receive ticket. 3. After filing the form with source, destination & date of journey, the Passenger click on “Book Ticket” Button. 4. The system checks the status of seat and train. If available, then Passenger redirected to payment page to complete the payment. 5. After submitting the payment, Passenger hit the “Reserve seat” button to complete the process. 6. Now, the system allows the Passenger to download his ticket. |

**Fully-Dressed Use Case Table**

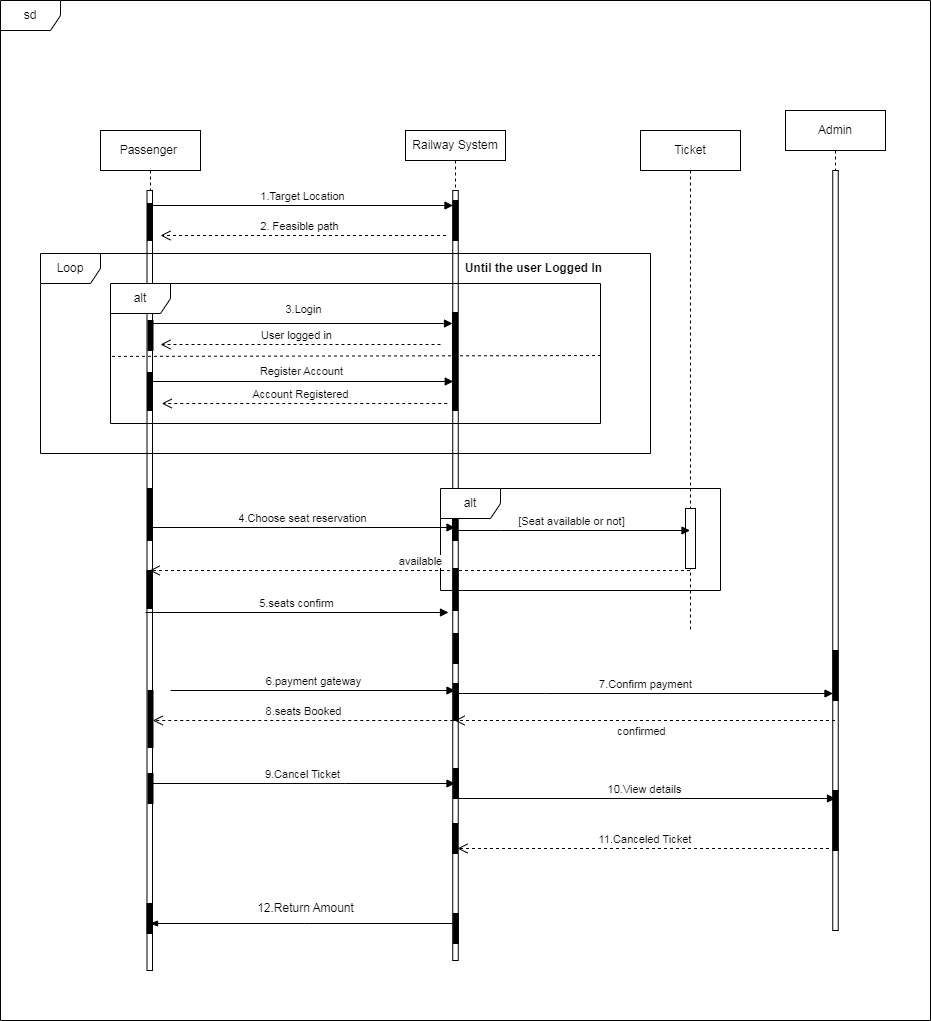
1. **Class Diagram:**

****

**Figure # 02**

1. **Architecture diagram:**

**Figure # 03**

1. **Sequence diagram:**

**Figure # 04**