ASIA PACIFIC INSTITUTE OF

INFORMATION TECHNOLOGY

**DATABASE AND WEB DATABASE SYSTEMS**

**Module Code: CE00318-2**

**INDIVIDUAL PROJECT**



**Submitted By: Submitted To:**

Prince Kumar Ms. Sulekh Sharma

PT1082219 Level-2 (Module Lecturer)

COMPUTING

# CERTIFICATE

This is to certify that Prince Kumar of 6th Semester student (Computing Branch) of APIIT SD INDIA have successfully completed her individual assignment of DWDS module on ‘Online Railway Reservation System’ with the complete guidance of Ms. Sulekh Sharma.

Prince Kumar (PT1082219) Ms. Sulekh Sharma

Level-2 (Module Lecturer)

COM

# ACKNOWLEDGEMENT

Around a period of two months I was working hard to achieve my goal. I have strived hard to complete my assignment. My goal was to give an exceptional and appreciating performance. Though there were lots of problems I had faced but I moved on and on.

Firstly, I would like to thank APIIT SD INDIA to provide me the opportunity to do such an assignment. I am very much thankful to my Module Lecturer Ms. Sulekh Sharma. For her uninterrupted support in the development of the assignment and have enabled me to complete my assignment on the given period of time. She was always there to listen and to give advice and she also taught me how to express my ideas and views professionally. I greatly acknowledge the effective contribution of Director R.K Choudhary, who encouraged me and provided me with all kinds of conveniences including Ultra modern Library and Computer Lab. I would also like to thank all our friends, family and everybody else who helped me anyways in our assignment.

Last but not the least; I want to acknowledge all my friends and colleagues who supported me in my assignment.

Prince Kumar

Level 2

Computing

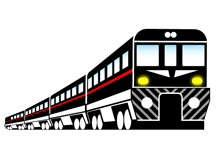


Table of Contents

[CERTIFICATE 2](#_Toc354649320)

[ACKNOWLEDGEMENT 3](#_Toc354649321)

[1.0. Introducing Online Railway Reservation System 5](#_Toc354649322)

[2.0. Abstract 5](#_Toc354649323)

[3.0. Problem Background 6](#_Toc354649324)

[4.0. Proposed Solution 6](#_Toc354649325)

[5.0. Programming Environment 7](#_Toc354649326)

[6.0 Entity Relationship Diagram 8](#_Toc354649327)

[7.0 Mapping of ERD to Relational Model 9](#_Toc354649328)

[8.0 Initial Schema **Error! Bookmark not defined.**](#_Toc354649329)

[9.0 Normalization 10](#_Toc354649330)

[10.0 Database Design 12](#_Toc354649331)

[11.0 Table Design 13](#_Toc354649332)

[12.0 Toolbar 16](#_Toc354649333)

[13.0 Form Design 17](#_Toc354649334)

[14.0 Report 21](#_Toc354649335)

[15.0 DDL Queries AND Constraints 22](#_Toc354649336)

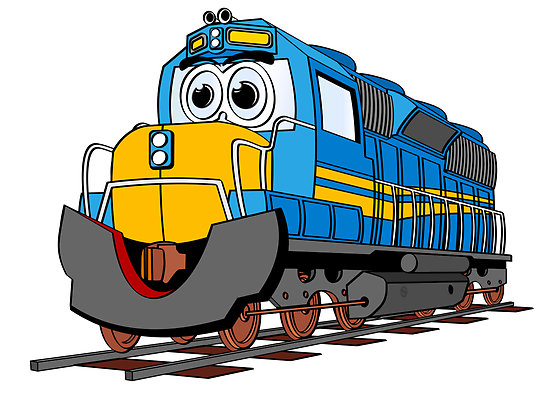
[16.0 DML Queries 24](#_Toc354649337)

[17.0 Conclusion 26](#_Toc354649338)

[18.0 References **Error! Bookmark not defined.**](#_Toc354649339)

# Introducing Online Railway Reservation System

Technology has transformed many aspects of life in the 21st century, including the way many of us make train reservations. For example, to make ticketing more convenient for travelers, Indian Railways\* has started an online reservation system, which helps us in booking tickets from the comfort of our homes or offices. While this is convenient for most people, it has made things particularly easier for people residing in remote locations.

The system is basically concerned with the reservation and cancellation of railways tickets of the passenger. The need of this system arose because as is the known fact that India has the largest railway network in the whole of the world and to handle it manually is quite a tough job. By computerizing it, we will be able to overcome many of its limitations and will be able to make it more efficient. The handling of data and records for such a vast system is a very complex task if done manually but it can be made easier if the system is computerized.

This System is basically concerned with the reservation and cancellation of railways tickets of the passenger. The need of this system arose because as is the known fact that India has the largest railway network in the whole of the world and to handle it manually is quite a tough job. By computerizing it, we will be able to overcome many of its limitations and will be able to make it more efficient. The handling of data and records for such a vast system is a very complex task if done manually but it can be made easier if the system is computerized. The Customers are required to register on the server for getting Access to the database and query result retrieval. Upon registration, each user has an account which is essentially the ‘view level’ for the customer. The account contains comprehensive information of the user entered during registration and permits the customer to get access to his past reservations, enquire about travel fare and availability of seats, make afresh reservations, update his account details, etc.

# Abstract

Railway Reservation System is a complex online distributed transaction application based on client server architecture. The salient features of the software include allowing user from anywhere to do a booking for a journey in any train in any class from anywhere to anywhere; handling reservation, modifications cancellation/refunds.

This project is all about the railway reservation online i.e., Software includes allowing user from anywhere to do a booking for a journey in any train in any class from anywhere to anywhere; handling reservation, modifications cancellation/refunds.

# Problem Background

In the existing system Reservation is done manually by a user at railway reservation counters. User has to wait for a long time in the queue to get the reservation. Like this wasting the time of user and for this lot of man power is required. and lot of paper work is also required.

# Proposed Solution

To avoid all the above pitfalls the system proposed is “Railway Reservation System”. In this all the reservation process through online. This system saves money, manpower, time. It provides security compared to existing system.

The basic functions being performed by our system are status, reservation and cancellation.

These functions will be handles with the help of following sub functions:-

* It reserves and cancels seats of passenger.
* It contains Information about the stations.
* It contains information about the trains.
* It contains information about the passenger.
* It contains the details of reservations fare.
* It makes entries for reservation, waiting, cancelled tickets.
* It will update for uptime and downtime trains.
* The passenger could search for trains from a particular source to destination.
* The record of train status includes dates for which tickets can be booked, total number of seats available, number of seats already booked and waiting slot

**How does it work?**

Railway Reservation system is to eliminate heavy load paper work by converting it into web based computer process. This web-based application is completely client - server oriented for processing

# Programming Environment

Visual Studio .NET2008

Visual Studio .Net is the rapid application development tool for BASIC. Visual Studio .Net offers complete integration with ASP.NET and enables to drag and drop server controls and design Web Forms as they should appear when user views them.

ASP.NET programming languages

ASP.NET has been designed to work seamlessly with WYSIWYG HTML editors and other programming tools, including Microsoft Visual Studio .NET. Not only does this make Web development easier, but it also provides all the benefits that these tools have to offer, including a GUI that developers can use to drop server controls onto a Web page and fully integrated debugging support. Developers can choose from the following two features when creating an ASP.NET application, Web Forms and Web services, or combine these in any way they see fit.

SQL Server 2008

A database system must provide the following features:

* A variety of user interfaces
* Physical & Logical data independence
* Query optimization
* Data integrity
* Concurrency control
* Backup and recovery
* Security and authorization

SQL Server is a Relational Database Management System. The SQL Server relational language is called Transact-SQL.SQL is asset-oriented language. This means that SQL can query many rows from one or more tables using just one statement. This feature allows the use of this language at a logically higher level than procedural languages. Another important property of SQL is its non- procedurally. SQL contains two sub languages DDL and DML.

# 

# 6.0 Entity Relationship Diagram

Station

End

M

1

Status

Status

1

Start

M

Book

Has

M

Train

1

1

Has

Passenger

Book

Has

M

user

Route

Route

List of entity type:

|  |  |  |
| --- | --- | --- |
| **S No.** | **Entity Type** | **Attributes** |
| 1. | Passenger | ticket-id, pssngr\_name,pssngr\_age,pssngr\_sex,pssngr\_seatno,train\_id |
| 2. | Train | Train\_no, Train\_name, source\_id, destination\_id, days, no of seats |
| 3. | Train Status | No. of seat available, No. of booked seat , Waiting seat,  Booking date. |
| 4. | Route | Tr\_no, St\_id, stop\_no. , arr\_time, dep\_time, dis. |
| 5. | Station | Station\_id,Station\_ name |
| 6. | User | User\_id,user\_name,usr\_age,user\_email,user\_password,user\_phn |

# 7.0 Mapping of ERD to Relational Model

# 8.0 Normalization

User\_registration

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| User\_id | User\_name | User\_address | User\_password | User\_Email | User\_Phone |

Station

|  |  |
| --- | --- |
| Station\_id | Station\_name |

TRAIN\_info

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Train\_no, | train\_name | train\_typ | Soucre\_name | Destination\_name |

Trninfo\_days

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Train\_no | Sun | Mon | Tue | Wed | Thu | Fri | Sat |

route

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Train\_no | Stop\_no | Arrival\_time, | Deptaure\_time, | Distance | Station\_id, |

Station\_route

|  |  |  |
| --- | --- | --- |
| Station\_id | Stop\_no | Train\_no |

Train\_status

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Train\_no | journey\_date | booked\_date | total\_seats, | A1 | A2 | SL | W1 | W2 | W3 |

Passerger

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ticket\_id | Passngr\_name | Passngr\_age | Passngr\_gender | Passngr\_seatno | Train\_id |

Passenger\_info

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Train\_id | Ticket\_id | Source\_id | Destination\_id | Class\_type | Resrvtion\_type |

Booking

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Train\_no | Ticket\_id | User\_id | Booking\_steats | Available\_date | Date\_sts |

Trninfo\_AC1

|  |  |  |
| --- | --- | --- |
| Train\_id | Trainstatus\_AC1 | Trainnfr\_AC1 |

Trninfo\_AC2

|  |  |  |
| --- | --- | --- |
| Train\_id | Trainsts\_AC2 | Trainfr\_AC2 |

Trninfo\_SL

|  |  |  |
| --- | --- | --- |
| Train\_id | Trainsts\_SL | Trnfr\_SL |

TRAIN\_info

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Train\_no, | train\_name | train\_type | Source\_name | Destination\_name |

# 

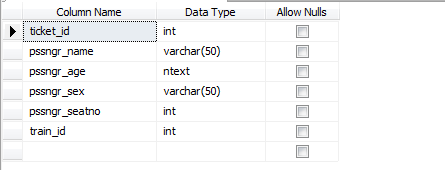
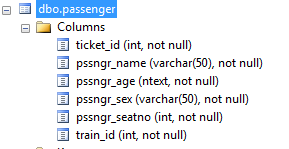
# 9.0 Database Design



# 

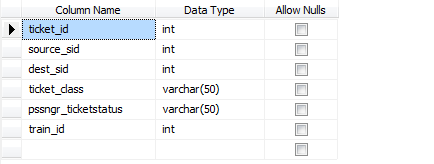
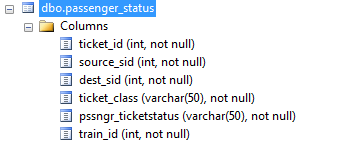
# 10.0 Table Design

Passenger



The passenger status consisting of six columns: ticket\_id, source\_sid, dest\_sid, ticket\_class, pssngr\_ticketstatus, train\_id

Passenger Status



Route

## route1.PNGroute.PNG

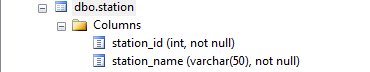
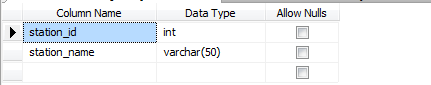
Route consisting of five columns

Train\_no, stop\_no, arrival\_time (i.e. arrival time of train), deprt\_time (departure time of train),dist(distance covered by train)

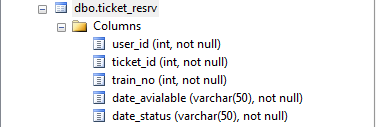
Station Route

## station_route.PNGstation_route1.PNG

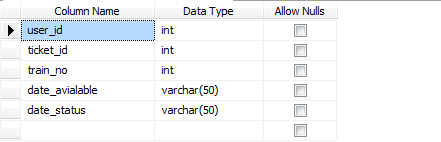
The station route It Consists of station\_id , train\_no , stop\_no

Station

Station has two column Station\_id , station\_name

Ticket Reservation

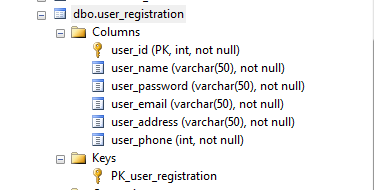
Ticket Reservation consist of User\_id , ticket\_id , train\_no , Available\_ date , date\_status

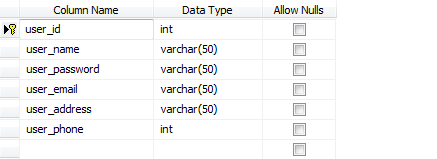


Train

## train.PNGtrain1.PNG

Train consists of Train\_no, Train\_name, Source\_sid, dest\_sid , train\_type

User Registration



Registration form of user consists of user\_id, user\_name, user\_password, user\_email, user\_address and user\_phone.

# 11.0 Toolbar

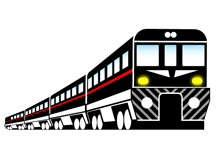
 (For Logout) – From this user can go back to login page. If user terminates session then user can easily get on “Login Form”.

(Booking) – For Reservation of ticket. User can book ticket by giving some Details.

 (Train between Stations)- To find no of running train with source id and destination id.

 Contact Us – For Further info about Admin or website handlers or organization

 Add Train

Add Route – For adding Route

# 12.0 Form Design

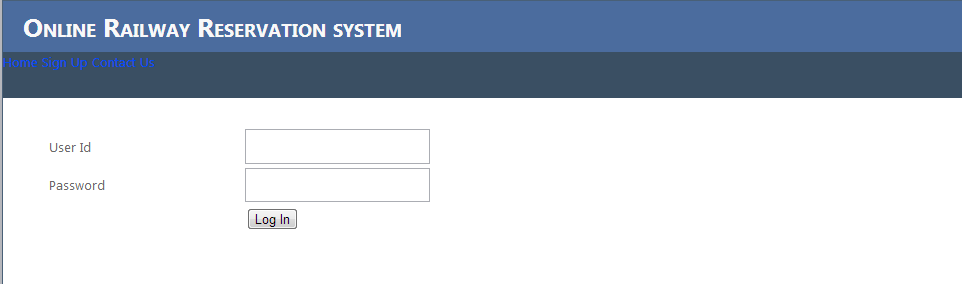


Fig: Login Form

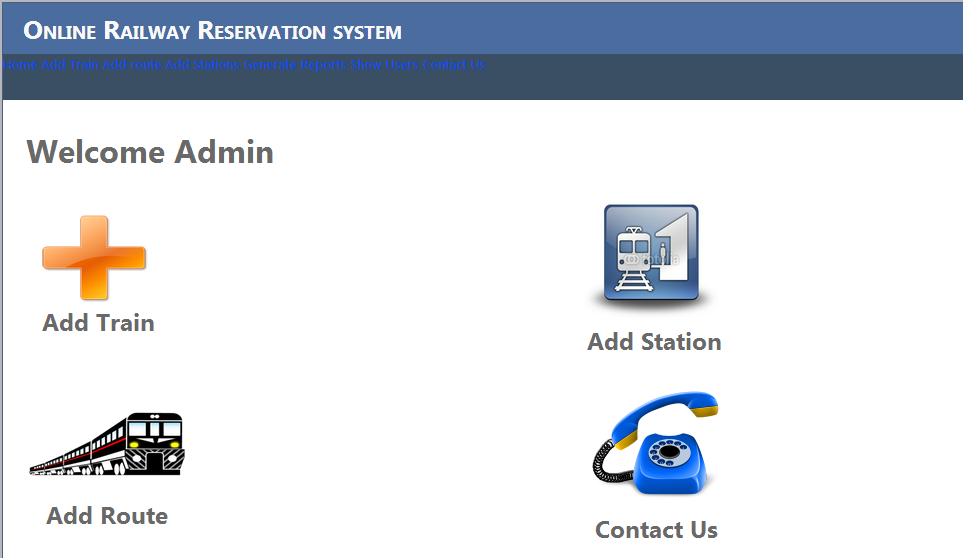


Fig: Home page Admin

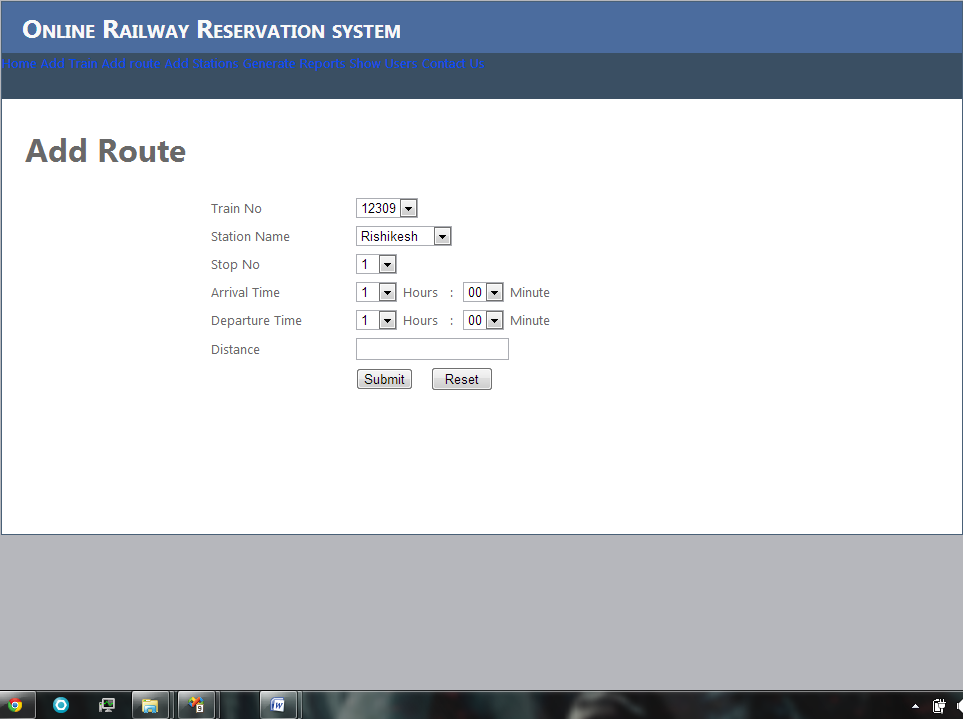


Fig: Add Route

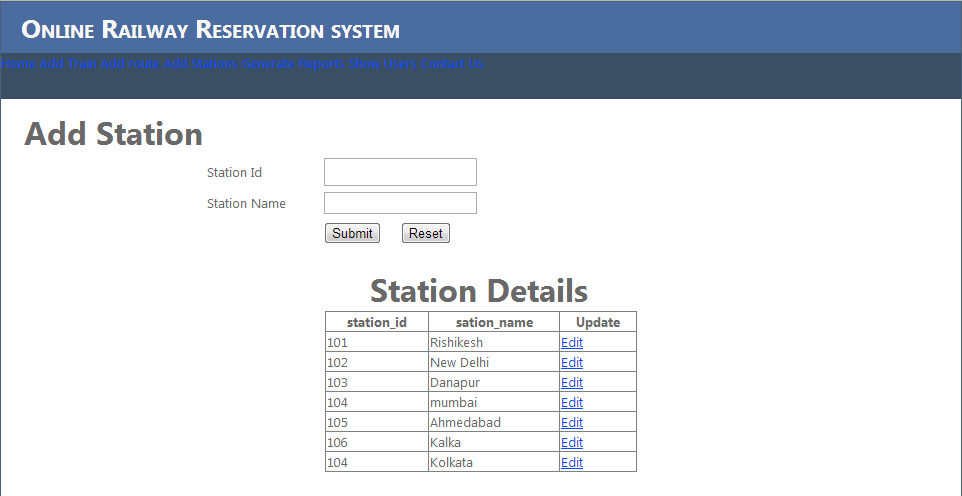


Fig: Add Station

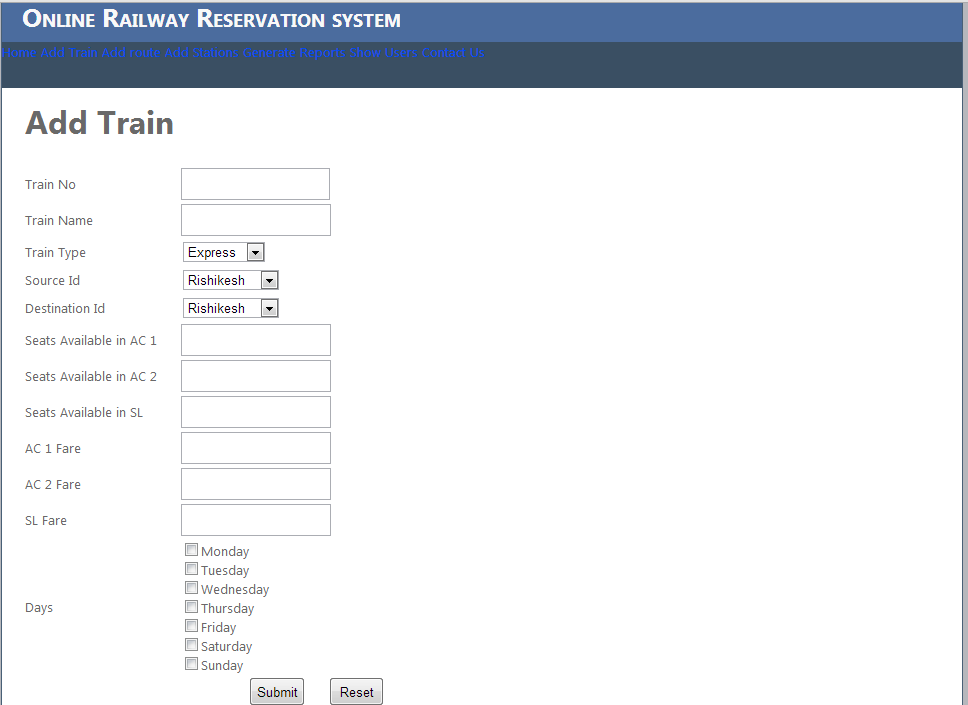


Fig: Add Train

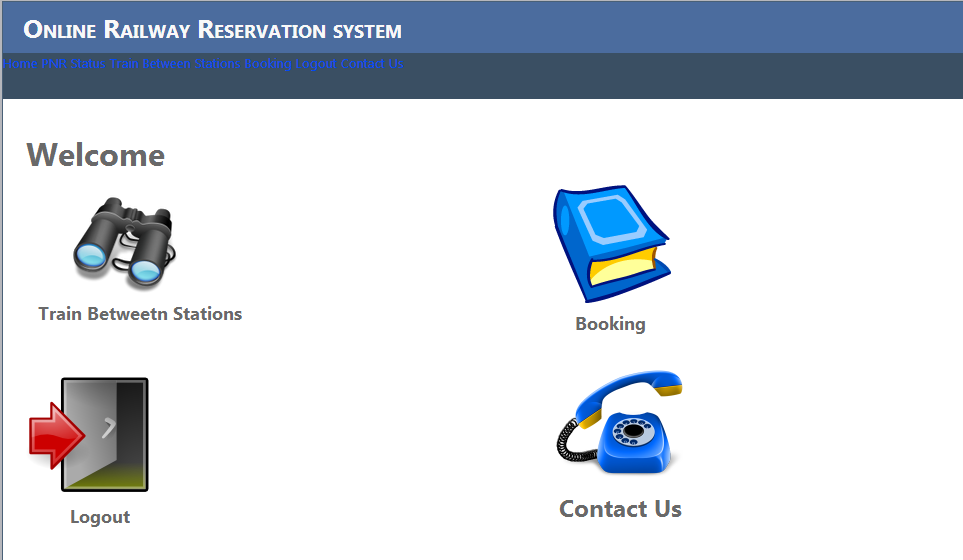
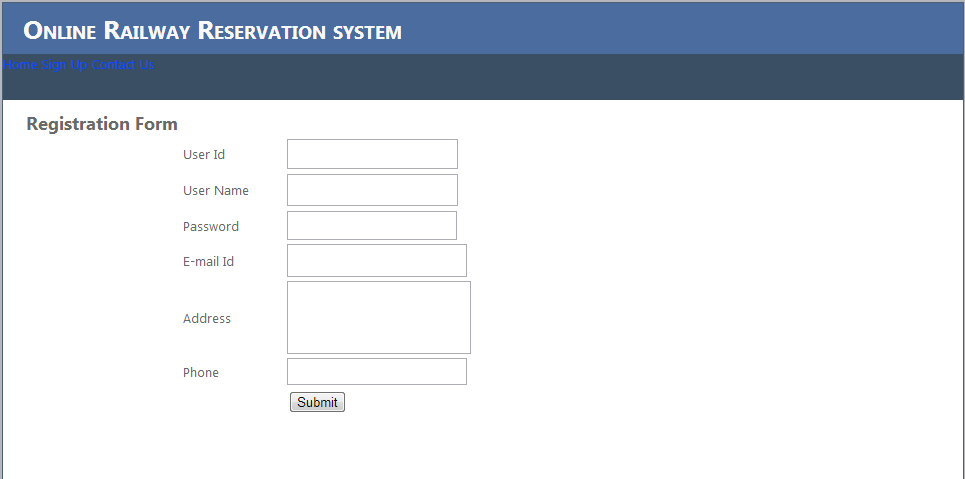


Fig: Welcome page of User

Fig: Registration Form

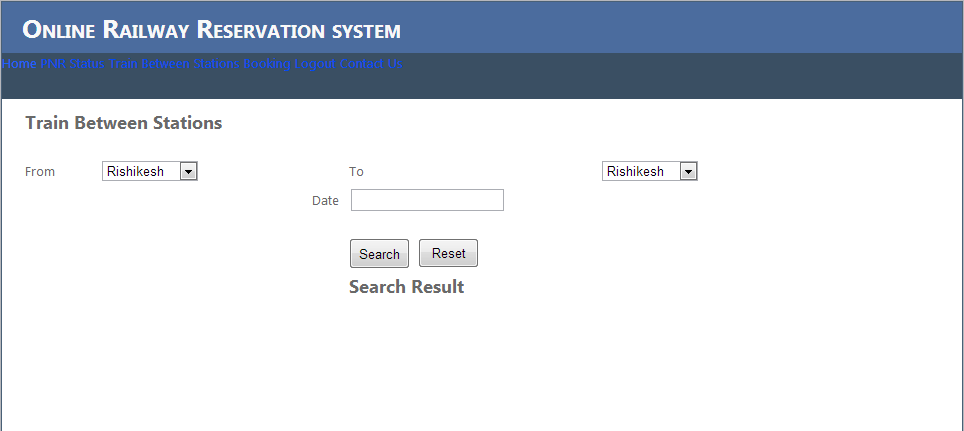


Fig: Train between Stations

# 13.0 Report



Fig : Users Report

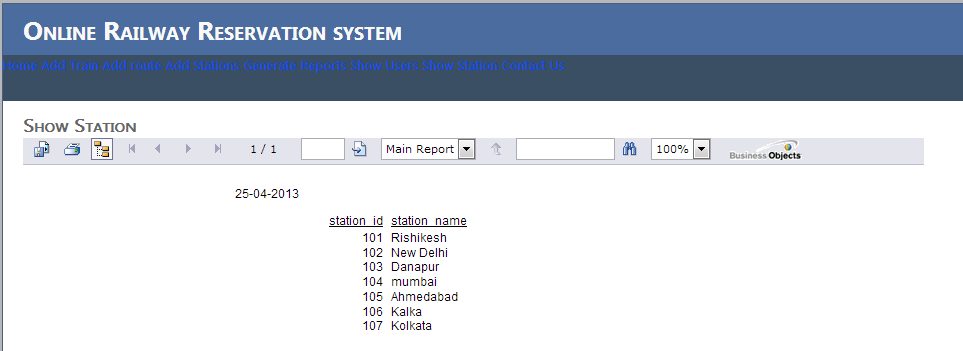


Fig : Show Station

# 14.0 DDL Queries AND Constraints

* Creation of Database

CREATE DATABASE, railway;

* Creation Of Table

Ticket Reservation

CREATE TABLE [dbo].[ticket\_resrv](

[user\_id] [int] NOT NULL,

[ticket\_id] [int] NOT NULL,

[train\_no] [int] NOT NULL,

[date\_avialable] [varchar](50) NOT NULL,

[date\_status] [varchar](50) NOT NULL

) ON [PRIMARY]

Train

CREATE TABLE [dbo].[train](

[train\_no] [int] NOT NULL,

[train\_name] [varchar](50) NOT NULL,

[source\_sid] [varchar](50) NOT NULL,

[dest\_sid] [varchar](50) NOT NULL,

[train\_type] [varchar](20) NOT NULL,

CONSTRAINT [PK\_train] PRIMARY KEY CLUSTERED

(

[train\_no] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

Train Running Days

CREATE TABLE [dbo].[train\_rdays](

[train\_no] [int] NOT NULL,

[sun] [varchar](50) NOT NULL,

[mon] [varchar](50) NOT NULL,

[tues] [varchar](50) NOT NULL,

[wed] [varchar](50) NOT NULL,

[thurs] [varchar](50) NOT NULL,

[fri] [varchar](50) NOT NULL,

[sat] [varchar](50) NOT NULL

) ON [PRIMARY]

ALTER TABLE [dbo].[train\_rdays] WITH CHECK ADD CONSTRAINT [FK\_train\_rdays\_train] FOREIGN KEY([train\_no])

REFERENCES [dbo].[train] ([train\_no])

Passenger

CREATE TABLE [dbo].[passenger](

[ticket\_id] [int] NOT NULL,

[pssngr\_name] [varchar](50) NOT NULL,

[pssngr\_age] [ntext] NOT NULL,

[pssngr\_sex] [varchar](50) NOT NULL,

[pssngr\_seatno] [int] NOT NULL,

[train\_id] [int] NOT NULL

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

Train AC2\_Status

CREATE TABLE [dbo].[trainAC2\_status](

[train\_no] [int] NOT NULL,

[date\_available] [varchar](50) NOT NULL,

[AC2\_reserveseats] [int] NOT NULL,

[AC2\_waitingseats] [int] NOT NULL,

[AC2\_availableseats] [int] NOT NULL,

CONSTRAINT [PK\_trainAC2\_status] PRIMARY KEY CLUSTERED

(

[train\_no] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

User Registration

CREATE TABLE [dbo].[user\_registration](

[user\_id] [int] NOT NULL,

[user\_name] [varchar](50) NOT NULL,

[user\_password] [varchar](50) NOT NULL,

[user\_email] [varchar](50) NOT NULL,

[user\_address] [varchar](50) NOT NULL,

[user\_phone] [int] NOT NULL,

CONSTRAINT [PK\_user\_registration] PRIMARY KEY CLUSTERED

(

[user\_id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

# 15.0 DML Queries

**Add Station**

Imports System.Data

Imports System.Data.SqlClient

Imports System.Configuration

Partial Public Class Add\_station

Inherits System.Web.UI.Page

Dim rail As String = ConfigurationManager.ConnectionStrings("database").ConnectionString

Dim con As New SqlConnection(rail)

Public Sub Details()

End Sub

Protected Sub Page\_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

If Page.IsPostBack = False Then

bindata()

End If

End Sub

Public Sub bindata()

con.Open()

Dim cmd As New SqlCommand("select \* from station", con)

Dim dr As SqlDataReader = cmd.ExecuteReader()

Grid\_Station.DataSource = dr

Grid\_Station.DataBind()

dr.Close()

con.Close()

End Sub

Protected Sub btn\_Submit\_Click(ByVal sender As Object, ByVal e As EventArgs) Handles btn\_Submit.Click

Try

con.Open()

Dim cmd As New SqlCommand("insert into station values ('" & TxtStnt\_Id.Text & "','" & TxtStnt\_Name.Text & "')", con)

cmd.ExecuteNonQuery()

con.Close()

MsgBox("Successfully Inserted", MsgBoxStyle.OkCancel)

bindata()

TxtStnt\_Id.Text = " "

TxtStnt\_Name.Text = " "

Catch ex As SqlException

MsgBox(ex.Message)

End Try

End Sub

Private Sub Grid\_Station\_RowEditing(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewEditEventArgs) Handles Grid\_Station.RowEditing

Grid\_Station.EditIndex = e.NewEditIndex

bindata()

End Sub

Private Sub Grid\_Station\_RowUpdating(ByVal sender As Object, ByVal e As System.Web.UI.WebControls.GridViewUpdateEventArgs) Handles Grid\_Station.RowUpdating

Dim row As GridViewRow = Grid\_Station.Rows(e.RowIndex)

Dim station\_id As String = CInt(CType(row.Cells(0).Controls(0), TextBox).Text)

Dim station\_name As String = CType(row.Cells(1).Controls(0), TextBox).Text

con.Open()

Dim sql As String = ("update station set station\_name='" & station\_name & "' where station\_id='" & station\_id & "'")

Dim cmd As New SqlCommand(sql, con)

cmd.ExecuteNonQuery()

con.Close()

Grid\_Station.EditIndex = -1

bindata()

MsgBox("Details has been sucessfully updated !!!)")

End Sub

Protected Sub Grid\_Station\_SelectedIndexChanged(ByVal sender As Object, ByVal e As EventArgs) Handles Grid\_Station.SelectedIndexChanged

End Sub

Protected Sub TxtStnt\_Id\_TextChanged(ByVal sender As Object, ByVal e As EventArgs) Handles TxtStnt\_Id.TextChanged

End sub

End Class

# 16.0 Conclusion

The main aim of developing Reservation system is to provide all information that is required by the users. User friendliness is a must that is the user must get the details without complicated searching procedures. Other important requirements of software are data security, extensibility and maintainability. All these features are included in this web application.

The project greatly helped in understanding the various phases in website development and exposure to a new developer platform MS Visual Studio .Net and database MS SQL Server.