GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Railway Reservation System

A Project Report of Project – 1

Submitted by

Krishna Kant Gautam

(1713104074)

In partial fulfillment for the award of the degree

of

Bachelor of Computer Application

IN

School of Computer Science & Engineering

Under the Supervision of

Mr Punitharaja.K

April/May – 2020

TABLE OF CONTENTS

Declaration

I Krishna Kant Gautam, hereby declare that the work which is being presented in this project report titled “Railway Reservation System” by me, in partial fulfillment of the requirements for the award of Bachelor of Computer and Application Degree in “School of Computer Science & Engineering” , is an authentic record of our own work carried out under the guidance of Mr Punitharaja.K

KRISHNA KANT GAUTAM

(17SCSE104076)

(1713104074)

ABSTRACT

The Railway Reservation System facilitates the passengers to enquire about the trains available on the basis of source and destination, Booking and Cancellation of tickets, enquire about the status of the booked ticket, etc. The aim of case study is to design and develop a database maintaining the records of different trains, train status, and passengers. This project contains Introduction to the Railways reservation system. It is the computerized system of reserving the seats of train seats in advanced. It is mainly used for long route. On-line reservation has made the process for the reservation of seats very much easier than ever before. In our country India, there are number of counters for the reservation of the seats and one can easily make reservations and get tickets. Then this project contains entity relationship model diagram based on railway reservation system and introduction to relation model. There is also design of the database of the railway reservation system based on relation model. Example of some SQL queries to retrieves data from rail management database.

INTRODUCTION

About Indian Railway:

Indian Railways is an Indian state-owned enterprise, owned and operated by the Government of India through the Ministry of Railways. It is one of the world's largest railway networks comprising 115,000 km (71,000 mi) of track over a route of 65,000 km (40,000 mi) and 7,500 stations. As of December 2012, it transported over 25 million passengers daily (over 9 billion on an annual basis). In 2011, IR carried over 8,900 million passengers annually or more than 24 million passengers daily (roughly half of which were suburban passengers) and 2.8 million tons of freight daily. In 2011–2012 Indian Railways had revenues of 1119848.9 million (US$17 billion) which consists of 696759.7 million (US$11 billion) from freight and 286455.2 million (US$4.4 billion) from passengers tickets.

Railways were first introduced to India in 1853 from Bombay to Thane. In 1951 the systems were nationalised as one unit, the Indian Railways, becoming one of the largest networks in the world. IR operates both long distance and suburban rail systems on a multi-gauge network of broad, metre and narrow gauges. It also owns locomotive and coach production facilities at several places in India and are assigned codes identifying their gauge, kind of power and type of operation. Its operations cover twenty- eight states and seven union territories and also provide limited international services to Nepal, Bangladesh and Pakistan. Indian Railways is the world's ninth largest commercial or utility employer, by number of employees, with over 1.4 million employees. As for rolling stock, IR holds over 239,281 Freight Wagons, 59,713 Passenger Coaches and 9,549 Locomotives (43 steam, 5,197 diesel and 4,309 electric locomotives). The trains have a 5- digit numbering system as the Indian Railways runs about 10,000 trains daily. As of 31 March 2013, 23,541 km (14,628 mi) (36%) of the total 65,000 km (40,000 mi) km route length was electrified. Since 1960, almost all electrified sections on IR use 25,000 Volt AC traction through overhead catenary delivery

Purpose of project

This system is basically concerned with the reservation and cancellation of railway tickets to the passengers. 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 much easier if the system is computerized.

To be more specific, our system is limited in such a way that a train starting from a particular source will have a single destination

The basic functions being performed by our system are:

1. RESERVATION MANAGEMENT
2. FARE MANAGEMENT
3. TIMETABLE MANAGEMENT

LIMITATIONS OF EXISTING SYSTEM

 Data redundancy:

It means that same data fields appear in many different files and often in different formats. In manual system, it poses quite a big problem because the data has to be maintained in large volumes but in our system, this problem can be overcome by providing the condition that if the data entered is duplicate, it will not be entered, otherwise, updating will take place.

 Difficulty in accessing the data:

In manual system, searching information is time consuming but in our system, any information can be accessed by providing the primary key.

 Unsatisfactory security measures

In manual system, no security measures were provided but in this system, password security has been provided. The person can access the system by providing the correct password otherwise he is denied the access

PROPOSED SYSTEM

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing systems. The system provides proper security and reduces the manual work

<< Security of data.

<< Ensure data accuracy's.

<< Proper control of the higher officials.

<< Minimize manual data entry.

<< Minimum time needed for the various processing.

<< Greater efficiency.

<< Better service.

<< User friendliness and interactive.

<< Minimum time required.

HARDWARE REQUIREMENT

Computer hardware is the collection of physical elements that constitutes a computer system. Computer hardware refers to the physical parts or components of a computer such as the monitor, mouse, keyboard, computer data storage, hard drive disk (HDD), system unit (graphic cards, sound cards, memory, motherboard and chips), etc. all of which are physical objects that can be touched.[1] In contrast, software is instructions that can be stored and run by hardware.

Minimum Hardware Requirements for our Program:

 PROCESSOR : Pentium IV processor or Greater

 RAM : 128 Mega Byte (MB) or Greater

 HARDDISK : 1.2 Giga Byte (GB) or Greater

 Keyboard & Mouse

 MONITOR : Colour (For Best Result)

 Printer

SOFTWARE REQUIREMENT

The software requirements to successfully run all the programs are

• Operating System - Windows 7, Windows XP.

• Microsoft IIS WEB server 5.0

• SQL Server 2005

• Visual Studio 2005

PROGRAMMING LANGUAGES INVOLVED

1.MySQL(MySQL server)

MySQL server is basically a database server which is mostly used for storing user data into the required database in a specific table for easy access of these data in the future. The database server works when a local application invokes it. Before storing data into a database with the help of database server, PHP sends a SQL query to MySQL database server for establishing a connection to the server by using the loopback address, i.e., 127.0.0.1 along with the user name and password for getting authentication from the database server. Since connection is established locally with the database server by PHP, so there is no chance of getting access to database server for modifying database by the client. Also, when PHP request for information retrieval from the database server by sending SQL query then the database server sends back the result of query after executing it by the database server. Another method of accessing database server is only possible by locally, i.e., only administrator of that particular computer can get all the access facilities of the database server as well as all the database which are not available to restricted users. PHP can update information in the database server if it has the administrator username and password to access the specific database, otherwise connection will be rejected by MySQL server and the database cannot be updated.

2.EVOLUTION OF ASP.NET

The first beta version of ASP was code named denali. Denali allowed the developer to execute code within a web page. Advanced functions could be performed using ActiveX Data Objects. The next version of ASP was ASP 1.0, WHICH WAS available as an add-on IIS 3.0. In this version of ASP, ADO had had become faster and effective as compared with the previous version. The next version ASP 2.0, came as a part of windows NT 4.0. Now components were easier to build because of the introduction of Microsoft Transaction Server, which allowed the components to be a part of transaction. Then, ASP 3.0 was introduced which came along with windows 2000. In this version, Microsoft combined MTS with core COM into COM+ and included it as a part of windows 2000 itself. The latest version, of ASP is ASP.NET. ASP.NET has come to us along with the .NET framework. ASP.NET is an object- oriented server -side scripting tool. Hence, it facilitates easy development of object oriented and granularized code.

DRABACKS OF ASP

1.VBScripts and JavaScript are the only two scripting languages available in ASP. These are basic non-typed languages. You cannot use strongly typed languages like Visual Basic or c++.

2. ASP pages are interpreted. This makes their execution slower.

3. ASP pages are very untidy. They are a spaghetti-like mixture of code. HTML and text.

4. While creating a web application using ASP pages, the programmers and the designer had to work on the same life. The programmers had to write the code to provide required functionality and the designers too had to create the graphics and content, and all this had to be combined into a single file.

5.In ASP, you have to write code to provide any functionality you require. For example, if you required any validations to be performed you had to write code for it to occur. A huge numbers of lines of code had to be written.

6. Re-use of code was not given much focus in ASP. You could do it only by using the include statement.

7. There was no debugging mechanism in ASP. You could debug ASP application only by using the response write. This is very tedious and not very effective.

8. In ASP, you had to stop the web server to install a new version of a DLL. DLL'S had to be registered in the registry to be available for use in an application. Moreover, the DLL registration process is a very complex one.

ADVANTAGE OF ASP.NET

1.ASP.NET supports strongly typed languages like VB, c#,and much more.

2.ASP.NET pages get compiled instead of being interpreted, thus their execution speed is faster than ASP pages.

3.ASP.NET pages are tidier than ASP pages. The code can be separated from the HTML design and text. Thus, programmers can work separately from the designers.

4. ASP.NET provides server controls that are declarative. You just have to declare them

5.ASP.NET provides server controls that are declarative. You just have to declare them and you can use them. Thus, the number of lines of code to be written is reused.

6.ASP.NET supports re-use of code by the mechanism of inheritance. For example, you can inherit c# classes and use them to provide the required functionality.

7.ASP.NET has an inbuilt method called Trace, which helps you in debugging a page. You can either display the trace information at the end of the page or redirect it to another location to check later.

FEATURE OF ASP.NET

1.ASP.NET can recognize the type of browser the client is using and accordingly display the content to the client. For example, if the client uses an uplevel browser (ie> 4.0 version), then the validation is performed on the client –side and on the server-side. However, if the client is using a downlevel browser, then the validation is performed only on the server-side.

2.ASP.NET improves performance by using server-side caching. It allows you to cache the entire output of a page for re-use by other clients.

3.ASP.NET functionality can be coded using different languages like C# or VB.NET. However, only one language can be used for coding in a single page.

4.ASP.NET ships with many built –in server controls that have the common required functionalities. For example, the developer need not create a control for accepting data; the textbox control is provided for this purpose. Thus, the developer need not to recreate these controls to obtain the required functionality.

5.A web service can be described as a function that can be deployed over the web and can be called by any application or other services. It can be a business application or a system function. ASP.NET allows you to create such web services.

IMPLEMENTATION OR ARCHITECTIURE DIAGRAM(System Overview)

Online railway ticket reservation system is a web- based application. It has been developed using ASP.NET and C# as the code end programming language. A MS SQL Server database consisting of multiple tables is used for data storage. This Website is a online travel booking website, which is capable of booking tour of national and international destination with easy steps at the cheapest rate. It allows the Administrator to perform all operations and view bookings. The general user is however given only restricted access. Travel helps in booking tour packages. The user can't book any tour until he is a registered user.

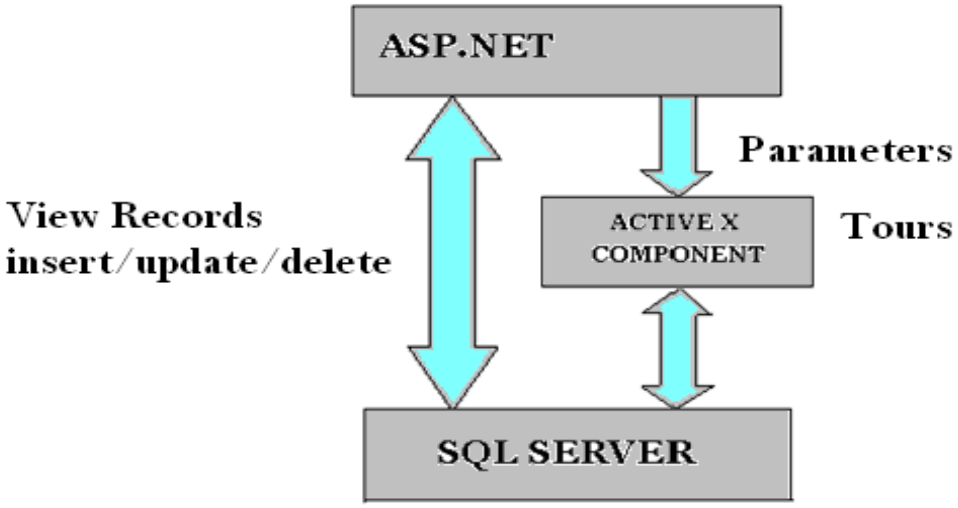
Database Tables

The following tables were created for the application :

Table Name	Description
Train details	Store details of each train
Booking	Stores details of user and particular train
Bank	Stores the details of all banking information of user while booking a ticket.

High-level Diagram

Following is a high-level diagram depicting the overall working of the system:



BASIC SYSTEM ARCHITECTURAL DESIGN

The system has a three-level architecture:

1.The first level (front end) uses ASP.NET technology with C# at the code end, to provide the users with the front-end.

2.The lower lever (back end) uses MS SQL Server to house the database.

3.The middle level comes into play only when Available tours are to be searched. This is achieved through the use of Active X Components (Data Grid). This picks up the requisite data from the database (depending upon the tours available and any other required conditions) and uses them to show the available tours.

RESERVATION FORM: This form is used for the reservation of ticket. The main advantage of the form is that it has the easiest of the user interface. This makes it user friendly and easy to use. It has Passenger's name, address, contact no, source and destination station name and codes.

CANCELLATION FORM: This form consists of cancelation. the user interface is again easy. One just needs the PNR number to cancel a ticket.

FARE RECORDS: This form is used for the fare between two station with a specific **class.**

TRAIN ENQUIRY: Train enquiry is used for knowing available trains between two stations. We need to provide source and destination names or codes and then we will be given the train names between those two stations.

RESERVATION ENQUIRY: This form is used to know if there is any seat available in a train. Here we need to provide date, train no, source and destination stations.

SEAT DETAILS: This form is used to know about the seat later using your PNR no.

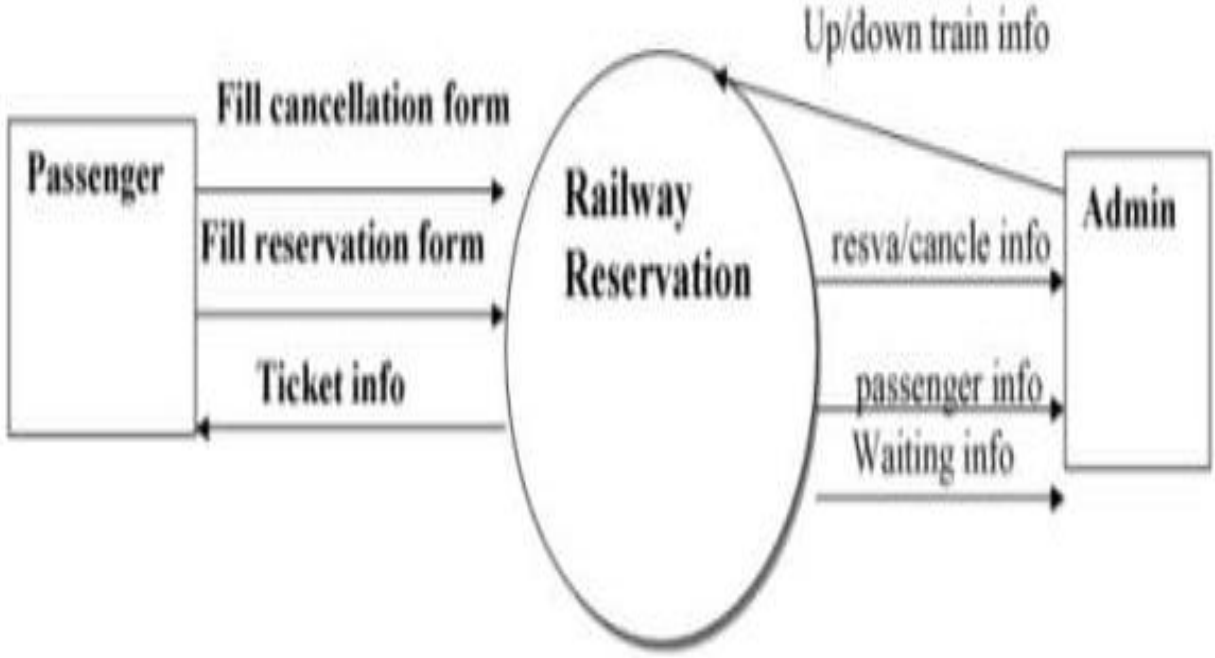
DATA FLOW DIAGRAMS

Definition of DFD:

The data flow diagram is a graphical representation that depicts information flow and the transforms that are applied as data moves from input to output. The DFD may be used to represent a system or software at any level of abstraction. In fact DFD may be partitioned into levels that represent increasing information flow and functional detail.

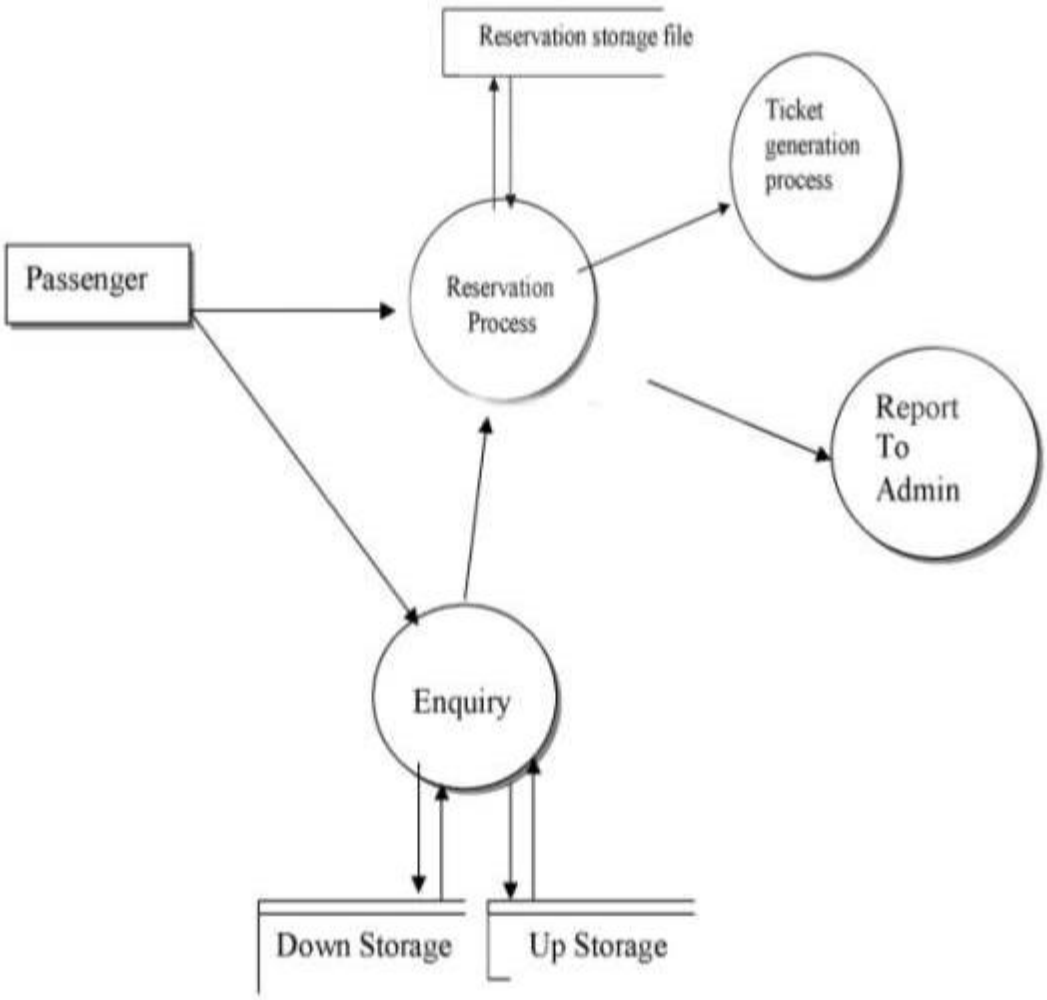
Level 0 DFD or Context Free Diagram:

The level 0 DFD or a context model represents the entire software element as a single bubble with input and output data indicated by incoming and outgoing arrows, respectively. In level 0 diagram shown below, the passenger fills either the reservation or cancellation form as input. He gets the ticket as the output and the report is sent to the administration.



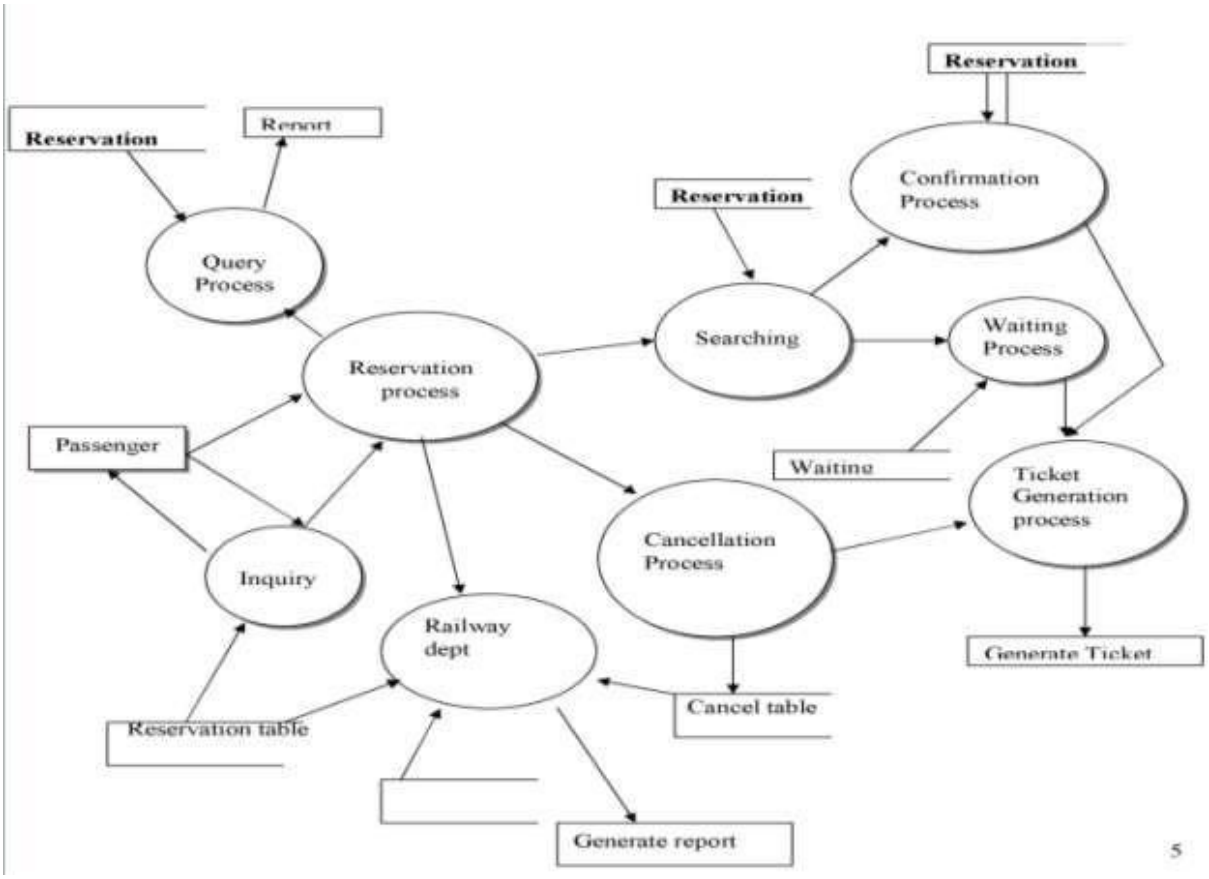
Level 1 DFD:

A level 1 DFD is the furthur refinement of level 0 DFD showing greater details and functionalities. In this, the single bubble of level 0 DFD is refined furthur . Each of the processes depicted at level 1 is a subfunction of the overall system depicted in the context model. As shown in the DFD above, the passenger either enquires about the trains or goes directly for the reservation or the cancellation processes as a result of which he gets the ticket generated. The reports are then sent to the administration



Level 2 DFD :

The level 2 DFD is the further refinement of the level 1 DFD. As shown in the DFD above the passenger has many options like he can directly go to the reservation counter or can first inquire and then go to the reservation counter or he can just inquire and return back. If the passenger wants reservation then the seats are checked for availability and if the seats are available the confirmation ticket is generated otherwise he is asked for waiting and waiting ticket is generated if he wants. If the user wants tickets to be cancelled he is given the cancellation ticket and the reports of all the transactions are sent to the administrator.



FLOW CHART

DESCRIPTION OF FLOW CHART:

The program flowchart shows how the system proceeds from the input form to the output form of the system. It explains how the system is actually processed step by step. It represents the flow of control as the system is processed.

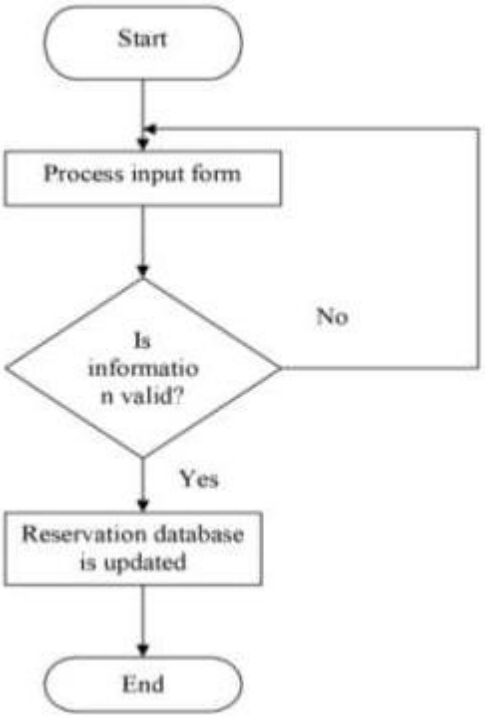
There are three types of program flow chart:

1. Input flow chart: This flowchart depicts the basic input operations in the system. In railway reservation system, first of all the password is checked then if the password is valid then we process the input form if the data is valid then the entries are updated in the data base otherwise the form is refilled.

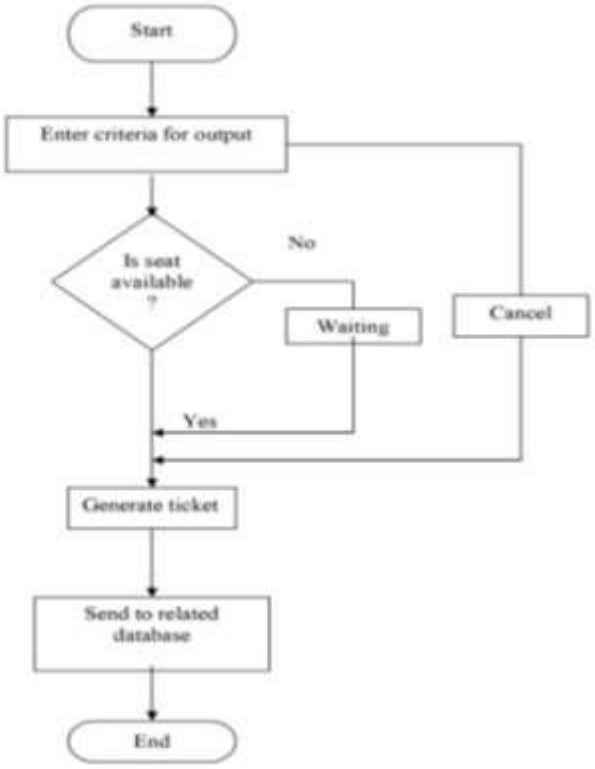
2. Output flow chart: This flowchart depicts the basic output operations in the system. The user is required to enter the criteria for output. If it is for the reservation then the availability of seats is checked. If the seats are available then the confirmed ticket is generated otherwise the user is asked for waiting and if he wants then waiting ticket is generated. If the user wants the seat to be cancelled it is done and the cancelled ticket is generated for the user. The information about all the above transactions is then transferred to the related databases.

3. Report flow chart: This flowchart depicts the basic operations for the generation of reports. If the entries from the processed database are valid the concerned reports are generated otherwise the process will have to be repeated

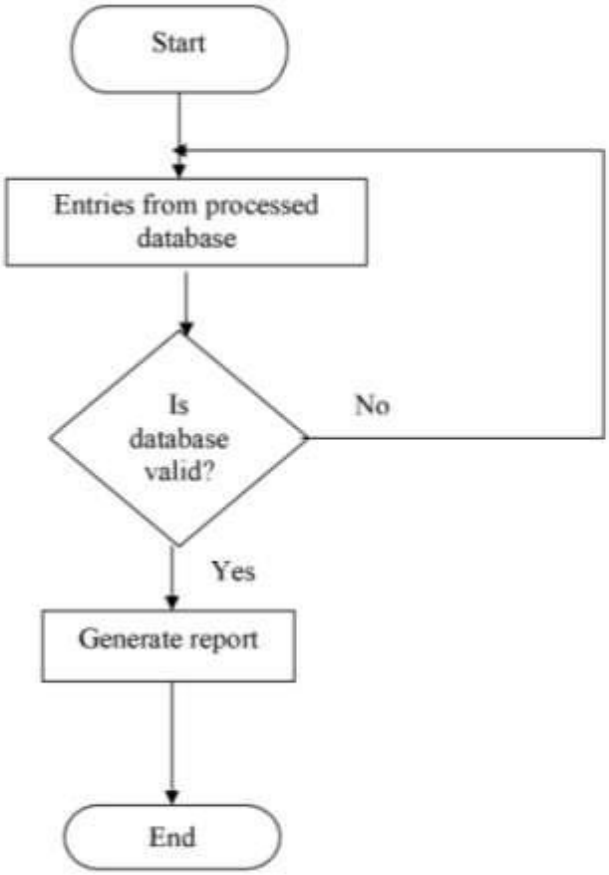
INPUT FLOW CHART



OUTPUT FLOW CHART



REPORT FLOW CHART



ENTITY RELATIONSHIP DIAGRAM

Entity relationship diagram expresses the overall logical structure of a database graphically. It shows the relationship between different entities. The entities can have composite, multivolume or derived attributes. The entities and their attributes are: -

1 Passenger

#. Name

*. First name

*. Middle name

*. Last name

#. Gender

#. Address

#. Age

#. Phone no.

2 Form

#. Form no.

3 Ticket

#. Ticket no.

*. Waiting

*. Confirmed

*. Cancelled

4 Reservation counter

5 Administrator

6 Train

#. Train no

#. Train name

#. Source

#. Destination

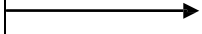
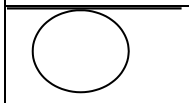
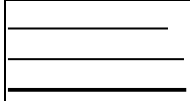
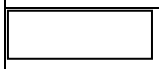
7 Seat

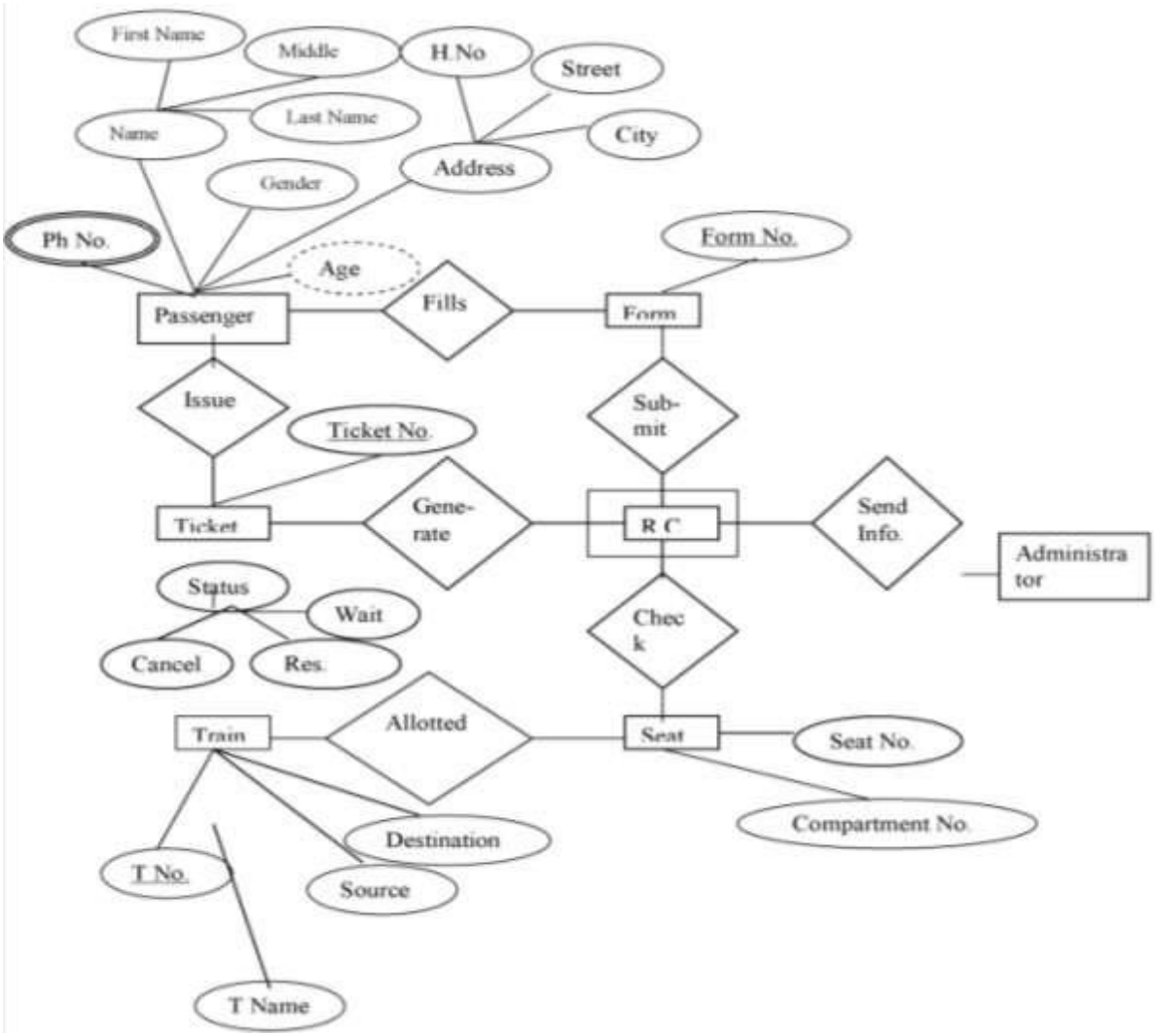
#. Seat no.

#. Compartment no.

The relationships between different entities are

1. Fill: The passenger fills the form.
2. Submit: The form is submitted to the reservation counter.
3. Check: The reservation counter checks the seats.
4. Generate: Reservation counter generates the ticket.
5. Issue: Reservation counter issues ticket to the passenger.
6. Send info: The reservation counter sends information to the administrator.
7. Allotted: The seat is allotted in the train.

Symbols	Meanings
	Data flow
	Process
	Data store
	Entity


First Name
Middle
H.No
Street
Name
Last Name
City
Address
Gender
Ph No.
Age
Form No.
Passenger
Fills
Form
Issue
Sub-
mit
Ticket No.
Ticket
Gene-
rate
R C
Send
Info.
Administra
tor
Status
Wait
Chec
k
Cancel
Res.
Train
Allotted
Seat
Seat No.
Compartment No.
Destination
T No.
Source
T Name

ERD

TESTING

Testing:

Testing is the process of exercising software with the intent of finding errors and ultimately correcting them. The following testing techniques have been used to make this project free of errors.

Content Review

The whole content of the project has been reviewed thoroughly to uncover typographical errors, grammatical error and ambiguous sentences.

Navigation Errors

Different users were allowed to navigate through the project to uncover the navigation errors. The views of the user regarding the navigation flexibility and user friendliness were taken into account and implemented in the project.

Unit Testing

Focuses on individual software units, groups of related units.

Unit – smallest testable piece of software.

A unit can be compiled /assembled / linked/loaded; and put under a test harness.

Unit testing done to show that the unit does not satisfy the application and /or its implemented software does not match the intended designed structure.

Integration Testing

Focuses on combining units to evaluate the interaction among them Integration is the process of aggregating components to create larger components. Integration testing done to show that even though components were individually satisfactory, the combination is incorrect and inconsistent. Integration is the process of aggregating components to create larger components. Integration testing done to show that even though components were individually satisfactory, the combination is incorrect and inconsistent.

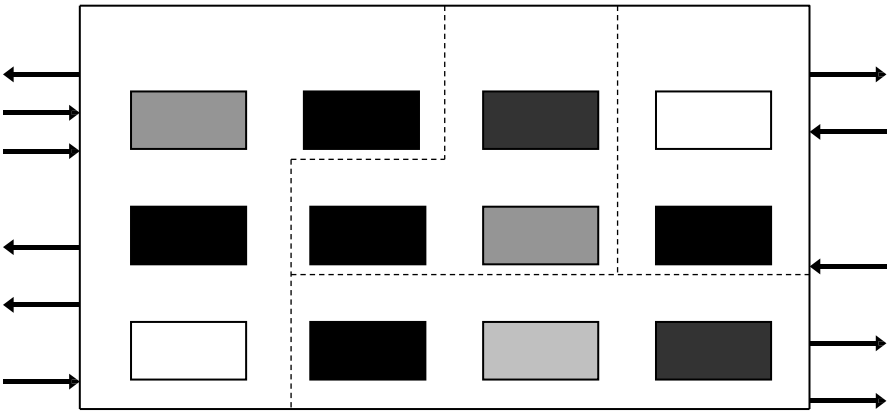
System Testing

Focuses on a complete integrated system to evaluate compliance with specified requirements (test characteristics that are only present when entire system is run)

A system is a big component.

System testing is aimed at revealing bugs that cannot be attributed to a component as such, to inconsistencies between components or planned interactions between components.

Concern: issues, behaviors that can only be exposed by testing the entire integrated system (e.g., performance, security, recovery)



each form encapsulates (labels, texts, grid etc.). Hence in case of project in V.B. form are the basic units. Each form is tested thoroughly in term of calculation, display etc.

Regression Testing

Each time a new form is added to the project the whole project is tested thoroughly to rectify any side effects. That might have occurred due to the addition of the new form. Thus regression testing has been performed.

White-Box testing

White-box testing (also known as clear box testing, glass box testing, transparent box testing and structural testing) tests internal structures or workings of a program, as opposed to the functionality exposed to the end-user. In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. This is analogous to testing nodes in a circuit, e.g. in-circuit testing (ICT).

While white-box testing can be applied at the unit, integration and system levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test. Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements.

Techniques used in white-box testing include:

API testing (application programming interface) – testing of the application using public and private APIs

Code coverage – creating tests to satisfy some criteria of code coverage (e.g., the test designer can create tests to cause all statements in the program to be executed at least once)

Fault injection methods – intentionally introducing faults to gauge the efficacy of testing strategies

Mutation testing methods

Static testing methods

Code coverage tools can evaluate the completeness of a test suite that was created with any method, including black-box testing. This allows the software team to examine parts of a system that are rarely tested and ensures that the most important function points have been tested. Code coverage as a software metric can be reported as a percentage for:

Function coverage, which reports on functions executed

Statement coverage, which reports on the number of lines executed to complete the test

100% statement coverage ensures that all code paths, or branches (in terms of control flow) are executed at least once. This is helpful in ensuring correct functionality, but not sufficient since the same code may process different inputs correctly or incorrectly.

Black-box testing

Black-box testing treats the software as a "black box", examining functionality without any knowledge of internal implementation. The tester is only aware of what the software is supposed to do, not how it does it. Black-box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, state transition tables, decision table testing, fuzz testing, model-based testing, use case testing, exploratory testing and specification-based testing.
Specification-based testing aims to test the functionality of software according to the applicable requirements. This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behaviour), either "is" or "is not" the same as the expected value specified in the test case. Test cases are built around specifications and requirements, i.e., what the application is supposed to do. It uses external descriptions of the software, including specifications, requirements, and designs to derive test cases. These tests can be functional or non-functional, though usually functional.
Specification-based testing may be necessary to assure correct functionality, but it is insufficient to guard against complex or high-risk situations.
One advantage of the black box technique is that no programming knowledge is required. Whatever biases the programmers may have had, the tester likely has a different set and may emphasize different areas of functionality. On the other hand, black-box testing has been said to be "like a walk in a dark labyrinth without a flashlight." Because they do not examine the source code, there are situations when a tester writes many test cases to check something that could have been tested by only one test case, or leaves some parts of the program untested.
This method of test can be applied to all levels of software testing: unit, integration, system and acceptance. It typically comprises most if not all testing at higher levels, but can also dominate unit testing as well.

Alpha Testing

Alpha testing is simulated or actual operational testing by potential users/customers or an independent test team at the developers' site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.

Beta Testing

Beta testing comes after alpha testing and can be considered a form of external user acceptance testing. Versions of the software, known as beta versions, are released to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the product has few faults or bugs. Sometimes, beta versions are made available to the open public to increase the feedback field to a maximal number of future users.

PROGRAM CODING

Form1:

```
Private Sub Command1_ Click()
If Text1.Text = "" Then
MsgBox "Enter PNR Number"
Else
Dim sql1
sql1 = "select * from RESERVATION_DETAILS WHERE PNR_NUMBER LIKE '" &
Text1.Text & "%'"
Adodc1.RecordSource = sql1
Adodc1.Refresh
DataGrid1.Visible = True
End If
End Sub
Private Sub Command2_Click()
End
End Sub
```

Form2:

```
DataGrid1.Visible = False
DataGrid2.Visible = True
Label4.Visible = True
Command3.Visible = True
End Sub
Private Sub Command3_Click()
Form2.Hide
Form3.Show
End Sub
Private Sub Command1_Click()
Dim SQL
If Combo1.Text = "" Or Combo2.Text = "" Or Combo3.Text = "" Then
MsgBox "Please enter all the details"
Else
```

```
SQL = "SELECT * FROM TRAIN WHERE FROM LIKE '" & Combo1.Text & "%' AND
TO LIKE '" & Combo2.Text & "%' AND DATE LIKE '" & Combo3.Text & "%'"
Adodc1.RecordSource = SQL
Adodc1.Refresh
DataGrid2.Visible = False
Label4.Visible = False
DataGrid1.Visible = True
Command2.Visible = True
Command3.Visible = False
End If
End Sub
Private Sub Command2_Click()
Dim sql1
sql1 = "SELECT * FROM TRAIN_DETAILS WHERE TRAIN_NUMBER LIKE '" &
Text1.Text & "%'"
Adodc2.RecordSource = sql1
Adodc2.Refresh
DataGrid1.Visible = False
DataGrid2.Visible = True
Label4.Visible = True
Command3.Visible = True
End Sub
Private Sub Command3_Click()
Form2.Hide
Form3.Show
End Sub
Private Sub Form Load()
End Sub
Private Sub Label1_Click()
End Sub
Private Sub Label5_Click()
End Sub
```

Form3:

```
Private Sub Command1_Click()
Adodc1.Recordset.Delete
Adodc1.Refresh
MsgBox "Tickets Has Been Sucessfully Booked"
MsgBox ("Your PNR Number is : " + Text14.Text)
End Sub
Private Sub Command2_Click()
Text5.Text = ""
Text11.Text = ""
Combo3.Clear
Combo2.Clear
End Sub
Private Sub Command3_Click()
End
End Sub
Private Sub Command4_Click()
Form3.Hide
Form4.Show
End Sub
Private Sub Command5_Click()
Adodc1.Recordset.AddNew
Text5.SetFocus
End Sub
Private Sub Command6_Click()
Text6.Text = Form2.Text1
Text7.Text = Form2.Label4 Caption
Text8.Text = Form2.Text2 Text
Text9.Text = Form2.Text3 Text
Text10.Text = Form2.Text4 Text
Text12.Text = Text14.Text
Text15.Text = Form2.Text5 Text
a = Val (Text16.Text)
If a > 0 Then
```

```
Text18.Text = "Confirmed"
Text16.Text = Val (Text16.Text) - 1
Else
Text18.Text = "Waiting"
Text16.Text =Val (Text16.Text) - 1
End If
Combo2.Refresh
Combo2.Text = " --Choose Class--"
MsgBox "You Can Add Another!!"
End Sub
Private Sub Form_Load()
Text1.Text = Form2.Text1
Text2.Text = Form2.Label4 Caption
Text3.Text = Form2.Text2 Text
Text4.Text = Form2.Text3 Text
Text17.Text = Form2.Text4 Text
Text13.Text = Form2.Text5 Text
Text16.Text = Form2.Text6 Text
Adodc1.Recordset.MoveLast

Adodc1.Recordset.AddNew
Text6.Text = Form2.Text1
Text7.Text = Form2.Label4.Caption
Text8.Text = Form2.Text2.Text
Text9.Text = Form2.Text3.Text
Text10.Text = Form2.Text4.Text
Combo2.Text = " --Choose Class--"
Text12.Text = Int(Rnd * 110097)
Text14.Text = Text12.Text
Text15.Text = Form2.Text5.Text
Text16.Text = Form2.Text6.Text
a = Val(Text16.Text)
If a > 0 Then
Text18.Text = "Confirmed"
```

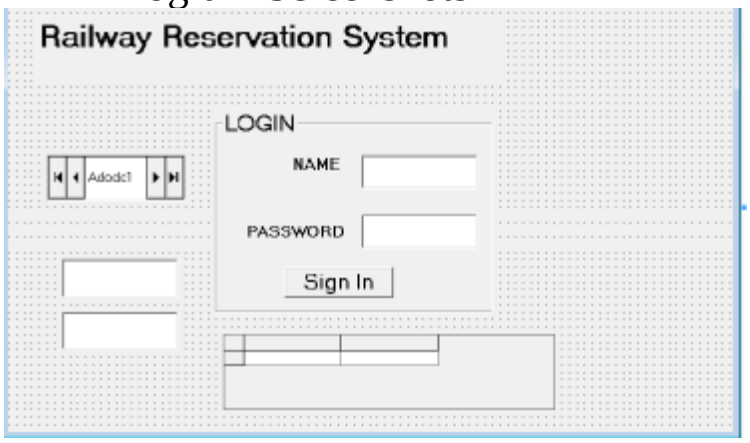
```
Text16.Text = Val(Text16.Text) - 1
Else
Text18.Text = "Waiting"
Text16.Text = Val(Text16.Text) - 1
End If
End Sub
```

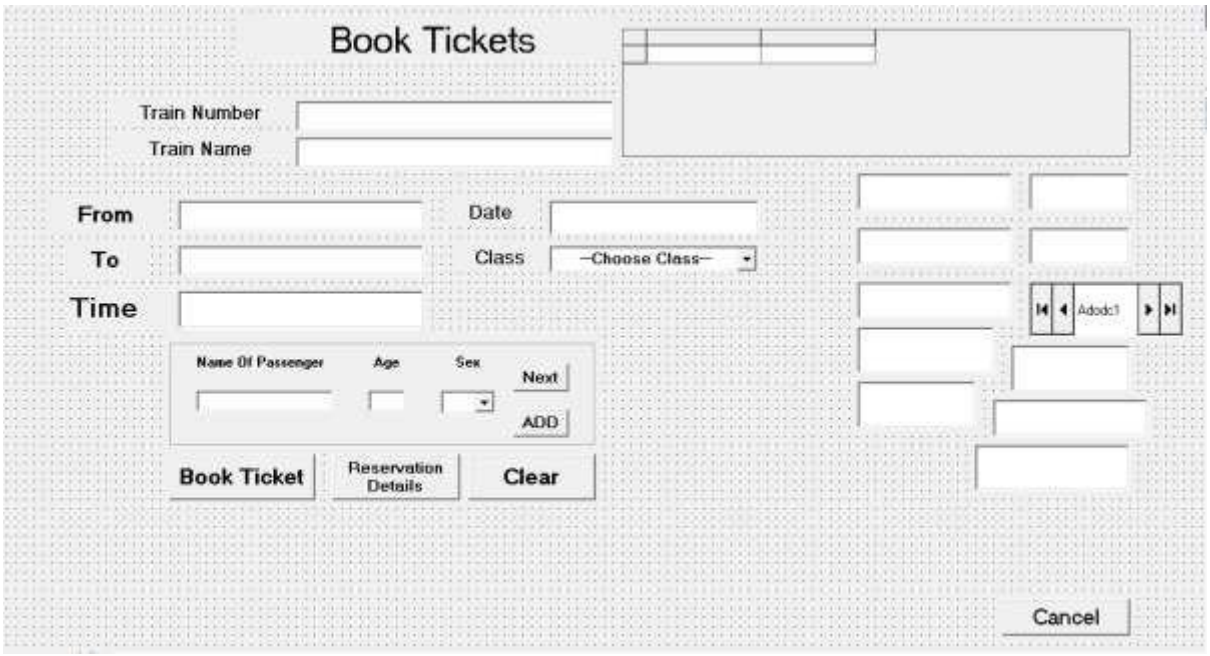
Form4:

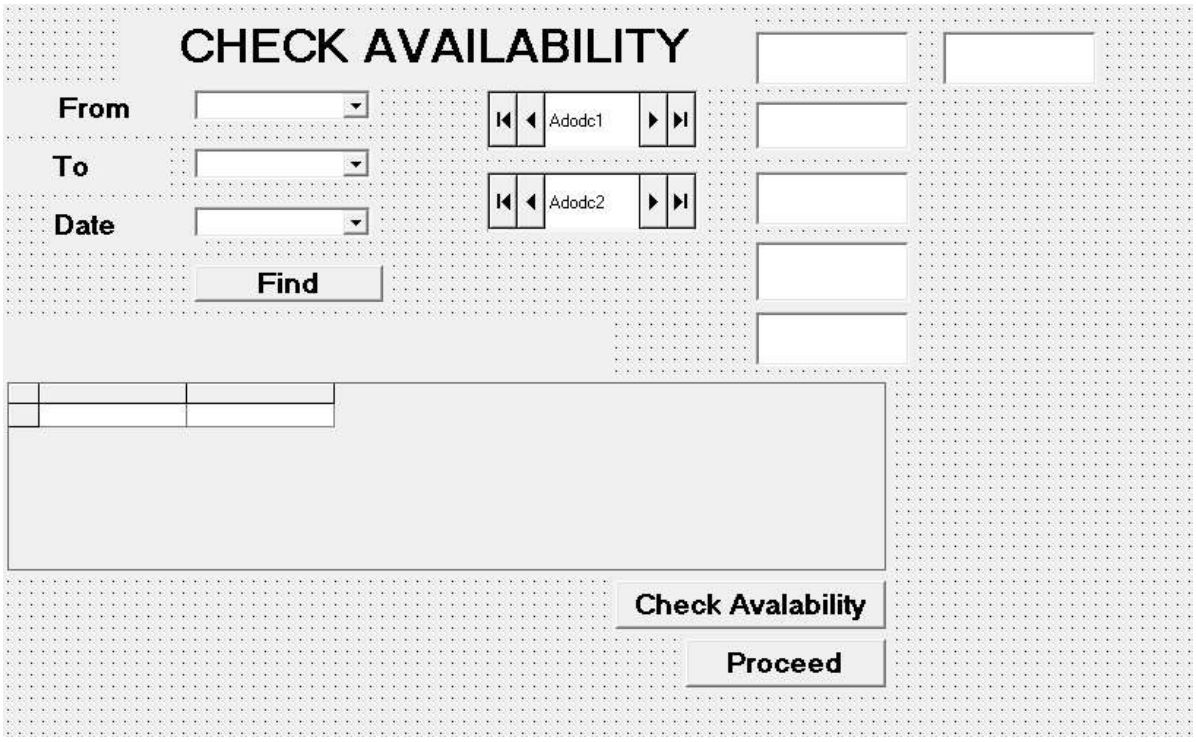
```
Private Sub Command1_Click()
If Text1.Text = "" Then
MsgBox "Enter PNR Number"
Else
Dim sql1
sql1 = "select * from RESERVATION_DETAILS WHERE PNR_NUMBER LIKE '" &
Text1.Text & "%'"
Adodc1.RecordSource = sql1
Adodc1.Refresh
DataGrid1.Visible = True
End If
End Sub
Private Sub Command2_Click()
End
End Sub
```

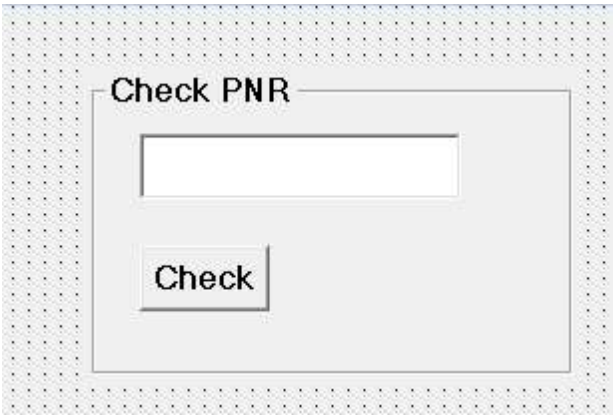
SCREENSHOTS

Program Screenshots






CHECK AVAILABILITY
From
To
Date
Find
Adodc1
Adodc2
Check Avalability
Proceed


Check PNR
Check

FUTURE ENHANCEMENT

1.Extra and user defined security needs can be employed during request and response.

2.Records of client activities can be maintained.

3.Another enhancement would be to add more modules to cover all information of the system.

CONCLUSION

Rather than designing manually we have made use of computer as once that data's are input it performs accurate function. There is no chance of fault or miscalculation if the data are feeded correctly. Use of the computers has solved many problems, which are faced while manual calculation.

This is not the end but beginning of the versatile, efficient and outsourcing railway reservation system. This is the one which is compatible to all operating system. By making this project we made a small footstep towards the path of progress of platform independent railway revervation system.

REFERENCES

- Reference Books:
    - Mastering Visual Basic 6 by Evangelos Petroutsos
- Study Materials:
    - www.w3school.com
    - www.tutorialspoint.com

- www.creatly.com (for making diagrams)