"RAILWAY RESERVATION SYSTEM"

A project report

In partial fulfilment of the requirements for the award of the degree

BACHELOR of TECHNOLOGY
in
INFORMATION TECHNOLOGY

Under the guidance of,

PALLABI SAHA

ARDENT COMPUTECH PVT. LTD.
High-End Technology Training and Project

Ardent Computech Pvt. Ltd. (An ISO 9001:2015 Certified)

SDF Building, Module#132, Ground Floor , Salt Lake City, CP Block,Sector -5
Kolkata - 700 091

Submitted by

Bandana Bharti
Sharmistha Sett
Debarati Paul
Suchismita Das
Pallabi Maji

GOVERNMENT COLLEGE OF ENGINEERING & LEATHER TECHNOLOGY
1919 2019
GLORIOUS JOURNEY OF 100 YEARS TRAVERSED

IN ASSOCIATED WITH

ARDENT COMPUTECH PVT. LTD.
High-End Technology Training and Project

(Note: ***All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.***

1. Title of the Project: "**Railway Management System"**
2. Project Members: 1. BANDANA BHARTI 2. SHARMISTHA SETT
   3. DEBARATI PAUL 4. SUCHISMITA DAS 5. PALLABI MAJI
3. Name and Address of the Guide: **Miss. Pallabi Saha**
   a. Trainer (CORE JAVA)
   b. Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)
   c. Module 132, SDF Building, Sector 5, Salt Lake, Kolkata - 64

4. Project Version Control History

Version	**Primary Authors**	**Description of Version**	**Date Completed**
Final	1. Bandana Bharti 2. Sharmistha Sett 3. Debarati Paul 4. Suchismita Das 5. Pallabi Maji	Project Report	8TH September, 2022

1.

2.

3.

4.

5.

Signature of Team Members

Date:

Signature of Approval

Date:

For Office Use Only

Approved	**Not Approved**

Miss. PALLABI SAHA
Project Proposal Evaluator

DECLARATION

We hereby declare that the project work being presented in the project proposal entitled **"RAILWAY RESERVATION SYSTEM"** in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** at **ARDENT COMPUTECH PVT. LTD, SALTLAKE, KOLKATA, WEST BENGAL** is an authentic work carried out under the guidance of **Miss. PALLABI SAHA.** The matter embodied in this project work has not been submitted elsewhere for the award of any degree of our knowledge and belief.

Date:

SL NO	NAME OF THE STUDENTS	SIGNATURE OF THE STUDENTS
1	**BANDANA BHARTI**	
2	**SHARMISTHA SETT**	
3	**DEBARATI PAUL**	
4	**SUCHISMITA DAS**	
5	**PALLABI MAJI**	

PROJECT RESPONSIBILITY FORM

RAILWAY RESERVATION SYSTEM

NAME OF MEMBER	RESPONSIBILITY
1.Bandana Bharti 2.Sharmistha Sett	Coding & Testing
3.Debarati Paul 4. Pallabi Maji 5.Sucismita Das	Designing & Documentation

Each group member must participate in project development and developing the ideas for the required elements. Individual group members will be responsible for completing tasks which help to finalize the project and the performance. All group members must be assigned a task.

Date:

SL NO	NAME OF THE STUDENTS	SIGNATURE OF THE STUDENTS
1	BANDANA BHARTI	
2	SHARMISTHA SETT	
3	DEBARATI PAUL	
4	SUCHISMITA DAS	
5	PALLABI MAJI	

Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified) SDF Building,
Module#132,Ground Floor , Salt Lake City, CP Block,Sector-5 Kolkata - 700 091

[ARDENT COMPUTECH pvt. Ltd.]

CERTIFICATE

This is to certify that **"Bandana Bharti , Sharmistha Sett, Debarati Paul, Suchismita Das, Pallabi Maji** " have successfully completed the project titled "Railway Reservation System" under my supervision during the period from "26/07/2022" to "08/09/2022" which is in partial fulfilment of requirements for the award of the **B.Tech** degree and submitted to the Department of "**Information Technology**" of **"Government College of Engineering and Leather Technology"**.

_____

Guide / Supervisor
Miss. Pallabi Saha

Ardent Computech Pvt Ltd (An ISO 9001:2015 Certified)

Module 132, SDF Building, Sector 5, Salt Lake, Kolkata – 64

ACKNOWLEDGEMENT

Thanks giving seem to be the most pleasant of all jobs but it is none of the less difficult when one tries this put them in words. Before we get thick of the things we would like to add a few heart felts words for the people who were part of this project "**RAILWAY RESERVATION SYSTEM**" in numerous ways people who gave unending support right from the project was conceived.

I express my sincere thanks to our Faculty Miss. PALLABI SAHA for his novel association of ideas, encouragement, appreciation and intellectual zeal which motivated us to made this project successfully.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success

TABLE OF CONTENTS

SL.NO:	Table of Contents	Page No.
1.	Abstract	8
2.	Introduction	9
	2.2 Project Objective	9
	2.3 Scope	9
3.	System Analysis	10 - 13
	3.1 Identification of Need	10
	3.2 Feasibility Study	10 – 11
	3.3 Workflow Model	11
	3.4 Functional Requirement	12
	3.5 Non – Functional Requirement	12 – 13
	3.6 Hardware Requirement	13
	3.7 Software Requirement	13
4.	System Design	14 - 19
	4.1 DFD (level 0)	15
	4.2 ER Diagram	16 - 17
	4.3 Database Design	18 - 19
5.	Coding and Implementation	20 - 26
	5.1 Codes	20 – 22
	5.2 Screenshots as Outputs with labels	23 - 26
6.	Testing and Maintenance	27 - 28
	6.1 Testing	27
	6.2 Objective of Testing	27
	6.3 Test cases	28
7.	Security Measures	29
	7.1 Database Security Measures	29
	7.2 App security Measures	29
	7.3 Limitation of App	29
8.	Future Scope	30
9.	Conclusion	31
10.	References	32

1. ABSTRACT

The Railway Reservation System facilitates the passengers to book 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 Railway reservation system. It is the computerized system of reserving the seats of train seats in advance. 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.

2. INTRODUCTION

This project introduces the Railway Reservation System. It explains how reservation is being done in Indian Railways. The step by step procedure is explained. This project is developed in JAVA language. Almost all the header files have been used in this project. Proper comments have been given at desired locations to make the project friendly. Various functions and structures are used to make a complete use of this language. This project is well versed with the programming. Railway Reservation can easily be accompanied with the help of this.

2.1 PROJECT OBJECTIVE

This project on Railway Reservation System is the automation of reservation process of railway system. The railway reservation system facilitates the passengers to enquiry about the trains available on the basis of source and destination, booking and cancellation of tickets, enquiry about the status of the booked ticket, etc.

2.2 SCOPE

It may help collecting perfect management in details. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the management of the past year perfectly and vividly. It also helps in current all works related to railway reservation system. It will be also reduced the cost of collecting the management and collection procedure will go on smoothly.

3. SYSTEM ANALYSIS

3.1 IDENTIFICATION OF NEED

The railway reservation system facilitates the passengers to enquiry about the trains available on the basis of source and destination, booking and cancellation of tickets, enquiry about the status of the booked ticket, etc. The aim of case study is to design and develop a data base maintaining 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 advance. It is mainly used for a long route. Online 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. Railway reservation system, has described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help organization it better utilization of resources. Administrator of the project, with the help of a password, can enter new train record, display all train records, modify train records and delete train records. The record of train includes its number, name, source, destination, and days on which it is available, whereas record of train status includes dates for which tickets can be booked, total number of seats available, and number of seats already booked.

3.2 FEASIBILITY STUDY

Feasibility is the determination of whether or not a project is worth doing. The process followed in making this determination is called a feasibility study. This type of study determines if a project can and should be taken.

Since the feasibility study may lead to the commitment of large resources, it becomes necessary that it should be conducted competently and that no fundamental errors of judgment are made. Depending on the results of the initial investigation, the survey is expanded to a more detailed feasibility study. Feasibility study is a test of system proposal according to its workability, impact on the organization, ability to meet user needs. And effective use of resources.

The objective of the feasibility study is not to solve the problem but to acquire a sense of its scope. During the study, the problem definition is crystallized and aspects of the problem to be include in the system are determined.

Operational Feasibility

Operational analysis is the most frequently used method for evaluating the effectiveness of a system. More commonly known as cost/benefit analysis and savings the are expected from a system and compare them with cost.

Technical Feasibility

Technical Feasibility centres around the existing computer system and also it can support the modification.

In manual processing there are more chance of errors are there, creating lot of complications, less technical or logical.

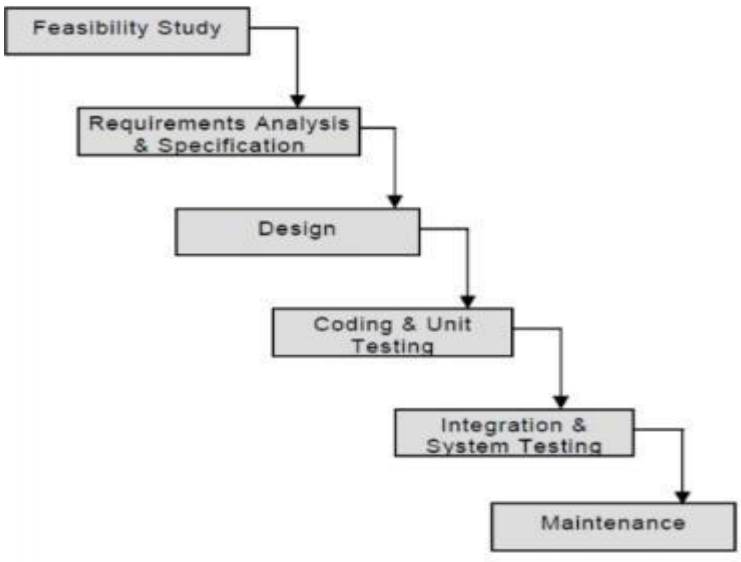
Through proposed system we can set this process in a very systematic pattern, which is more technical, full proof, authentic, safe and reliable.

Economic Feasibility

Economic feasibility is the most frequently used and important method for know the effectiveness of the proposed system. It is necessary because the main goal of the proposed railway reservation system Java project is to give economically better result and also increased efficiency. Cost benefit analysis can also be performed for this purpose.

Cost benefit analysis is the comparative study of the cost verses benefit and savings that are expected from the proposed system. Since the organization is well equipped with the required hardware and software so the Railway Reservation System was found to be economically.

3.2 WORKFLOW



Waterfall Model

3.4 FUNCTIONAL REQUIRNMENTS

- **CONNECTION CLASS**

The JDBC Connection class, JAVA, SQL connection represent a connection to a relational database.

- **LOGIN CLASS**

A login is a set of credentials used to authenticate admin. These consist of a username and password.

- **ADD TRAIN CLASS**

Admin can add trains and take reservations here for passengers.

- **RESERVATION CLASS**

Admin can put source and destination credentials to check trains or modify train availability for passengers. Admin can reserve seats for passengers here.

3.5 NON – FUNCTIONAL REQUIRNMENTS

This is a desktop based application and is held in Offline mode.

- **Efficiency of use:** The average time is takes to accomplish a user, goals, how many task a user can complete without any help, the number of transactions completed without errors etc.
- **Intuitiveness:** How simple it is to understand the interface, buttons, heading etc.
- **Low perceived workload:** How many attempts are needed by users to accomplish a particular task.
- **Security:** Security requirements ensure that the software is protected from unauthorised access to the system and it store data.
- **Reliability:** Reliability defines how likely it is for the software to work without failure for a given period of time.
- **Performance:** Performance is quality attribute that describes the responsiveness of the system to various user interactions with it.
- **Availability:** It is gauged by the period of time that the system's functionality and services are available for use with all operations.

- **Scalability:** Scalability requirements describe how the system must grow without negative influence on its performance.

3.6 HARDWARE REQUIRNMENT

1. Computer Processor Core i3 Processor Speed 2.3 GHz Processor
2. Hard Disk 400 GB or more RAM Min 2GB.

3.7 SOFTWARE REQUIRNMENT

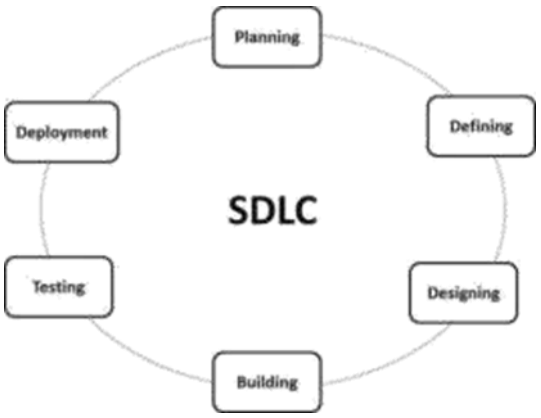
1. Operating System : Windows 10
2. Frontend Software: Apache NetBeans 13 : JDK 17
3. Backend Software: MySQL

4. SYSTEM DESIGN

It is a process of planning a new business system or replacing an existing system by defining its components or modules to satisfy the specific requirements. Before planning, you need to understand the old system thoroughly and determine how computers can best be used in order to operate efficiently.

SDLC

In this project we have followed the Waterfall model. The waterfall model is the most familiar model. This model has five phases: requirements analysis and specifications, design, implementation and unit testing, integration and system testing, and operation and maintenance.

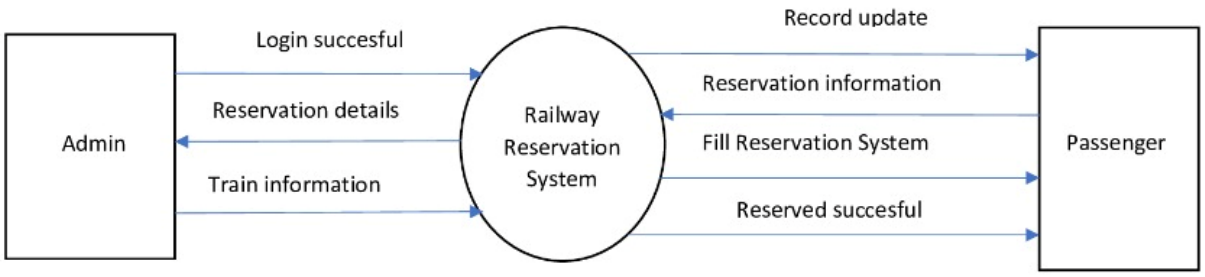


SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

4.1 DATA FLOW DIAGRAM

A Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. A structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts.

When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, specifies the implementation plan etc.



DFD (0 LEVEL)

4.2 ENTITY RELATIONSHIP DIAGRAM

In software engineering, an entity–relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them. However, variants of the idea existed previously, and have been devised subsequently such as super type and subtype data entities and commonality relationships.

An entity–relationship model is a systematic way of describing and defining a business process. The process is modelled as components (entities) that are linked with each other by relationships that express the dependencies and requirements between them, such as: one building may be divided into zero or more apartments, but one apartment can only be located in one building. Entities may have various properties (attributes) that characterize them. Diagrams created to represent these entities, attributes, and relationships graphically are called entity– relationship diagrams

ENTITY RELATIONSHIP DIAGRAM

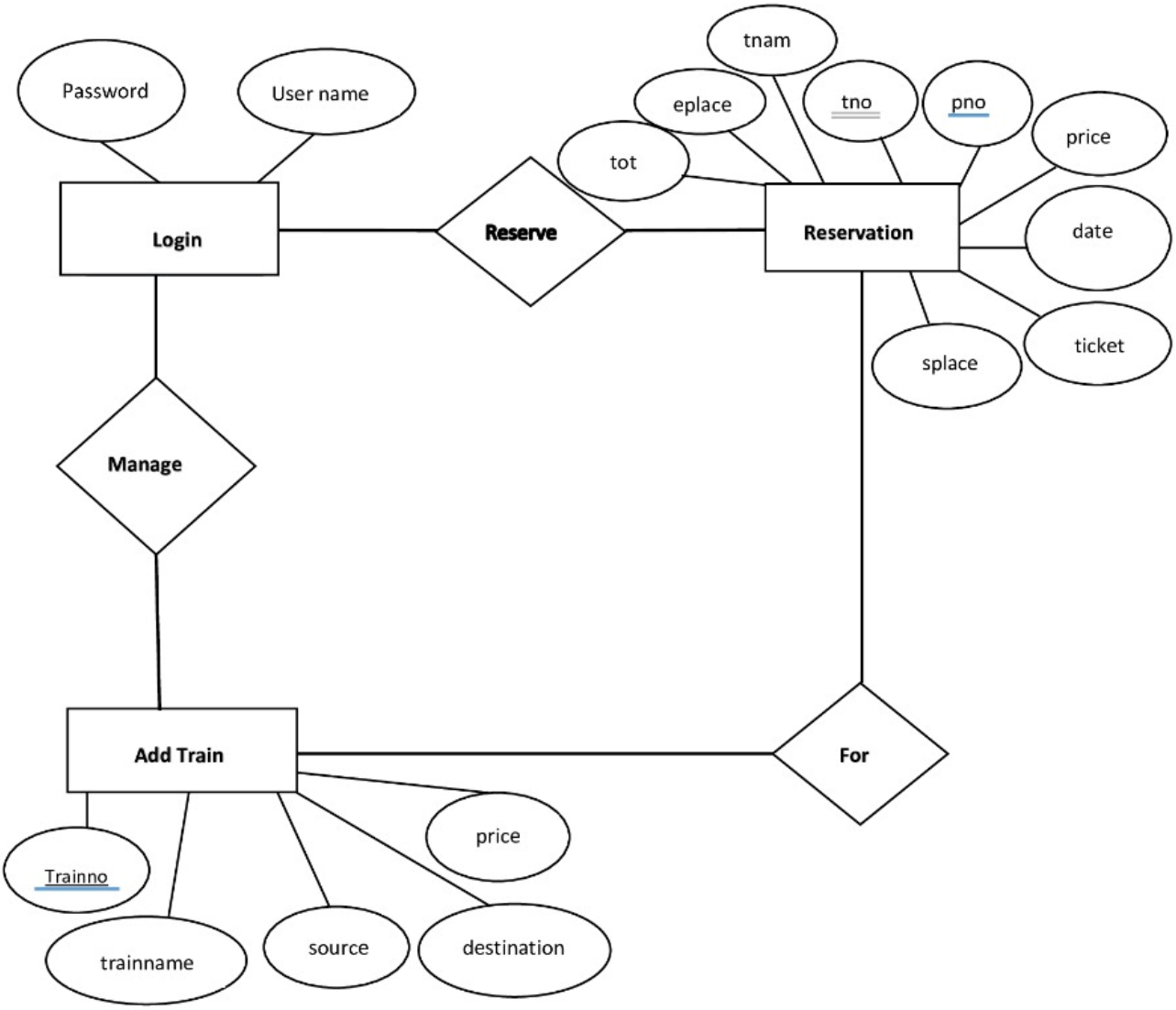
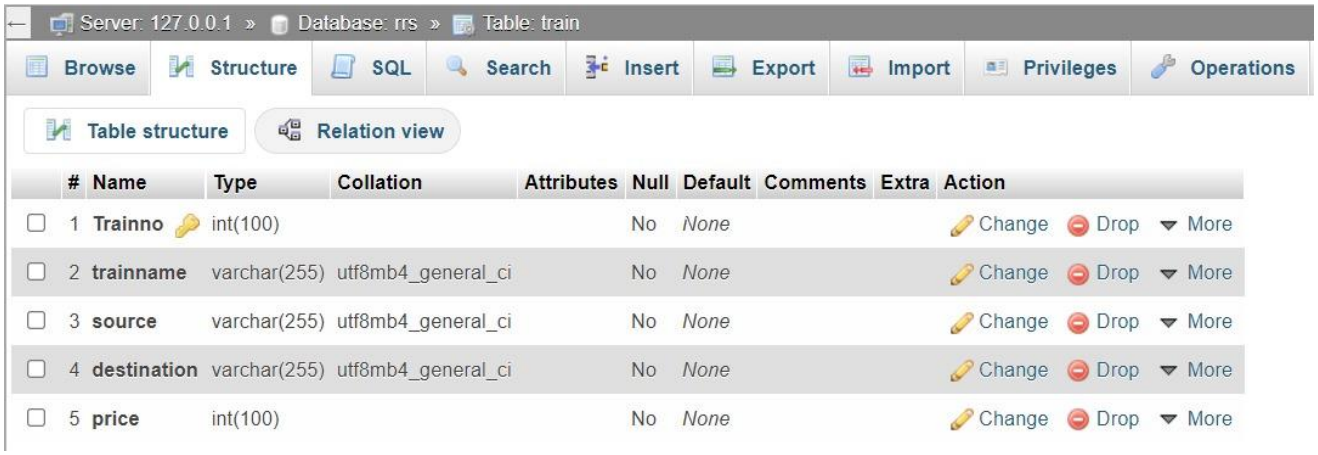


Figure: Entity – Relationship Diagram of Railway Reservation System

4.3 DATABASE DESIGN

A database is an organized mechanism that has capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected. The database design is two level processes. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called information Level design and it is taken independent of any individual DBMS. In the following snapshots we display the way we have used SQL Server as the backend RDBMS for our project and the various entities that have been used along with their table definition and table data.

TRAIN DATABASE



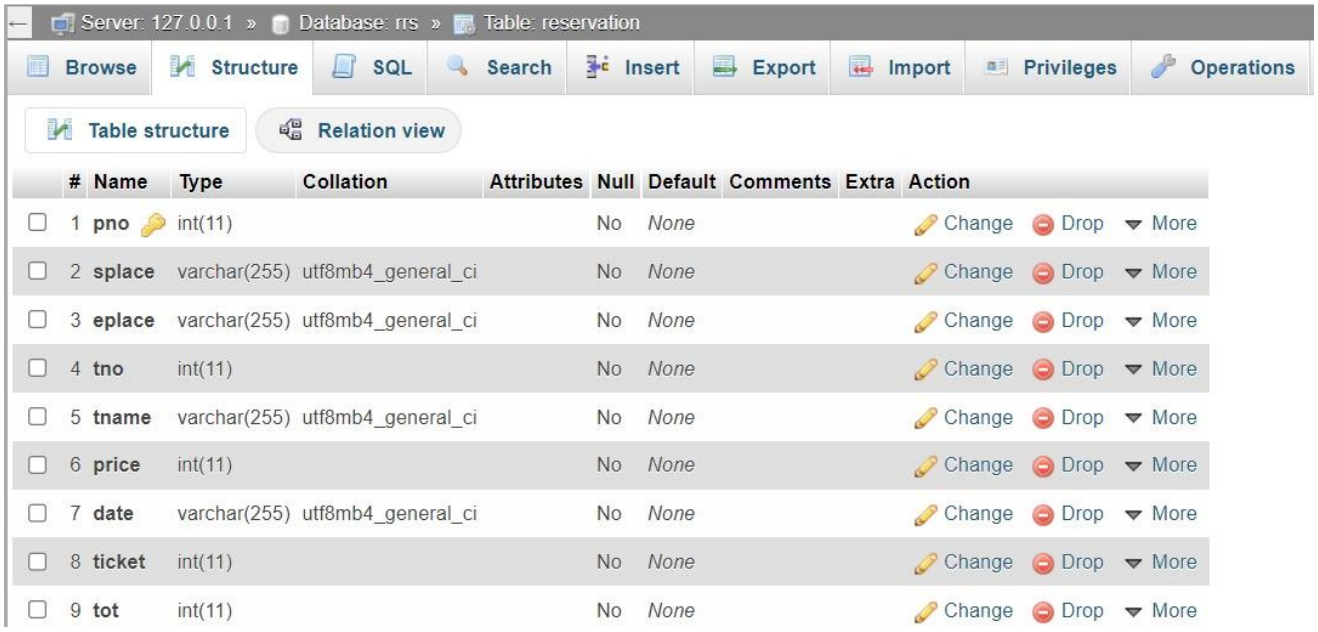
Server: 127.0.0.1 » Database: rrs » Table: train

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations

Table structure | Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Trainno	int(100)			No	None			Change Drop More
2	trainname	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
3	source	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
4	destination	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
5	price	int(100)			No	None			Change Drop More

RESERVATION DATABASE



Server: 127.0.0.1 » Database: rrs » Table: reservation

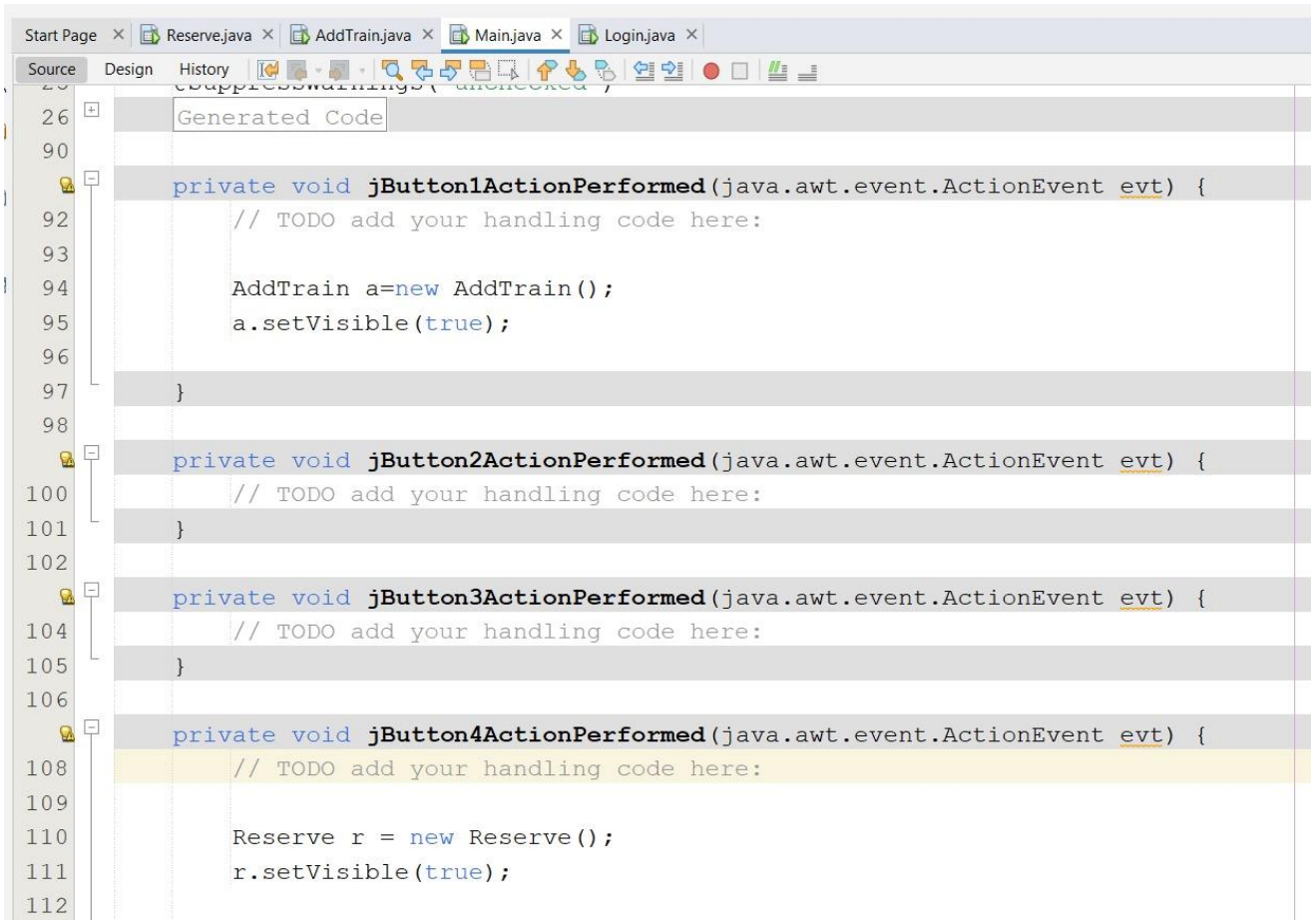
Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations

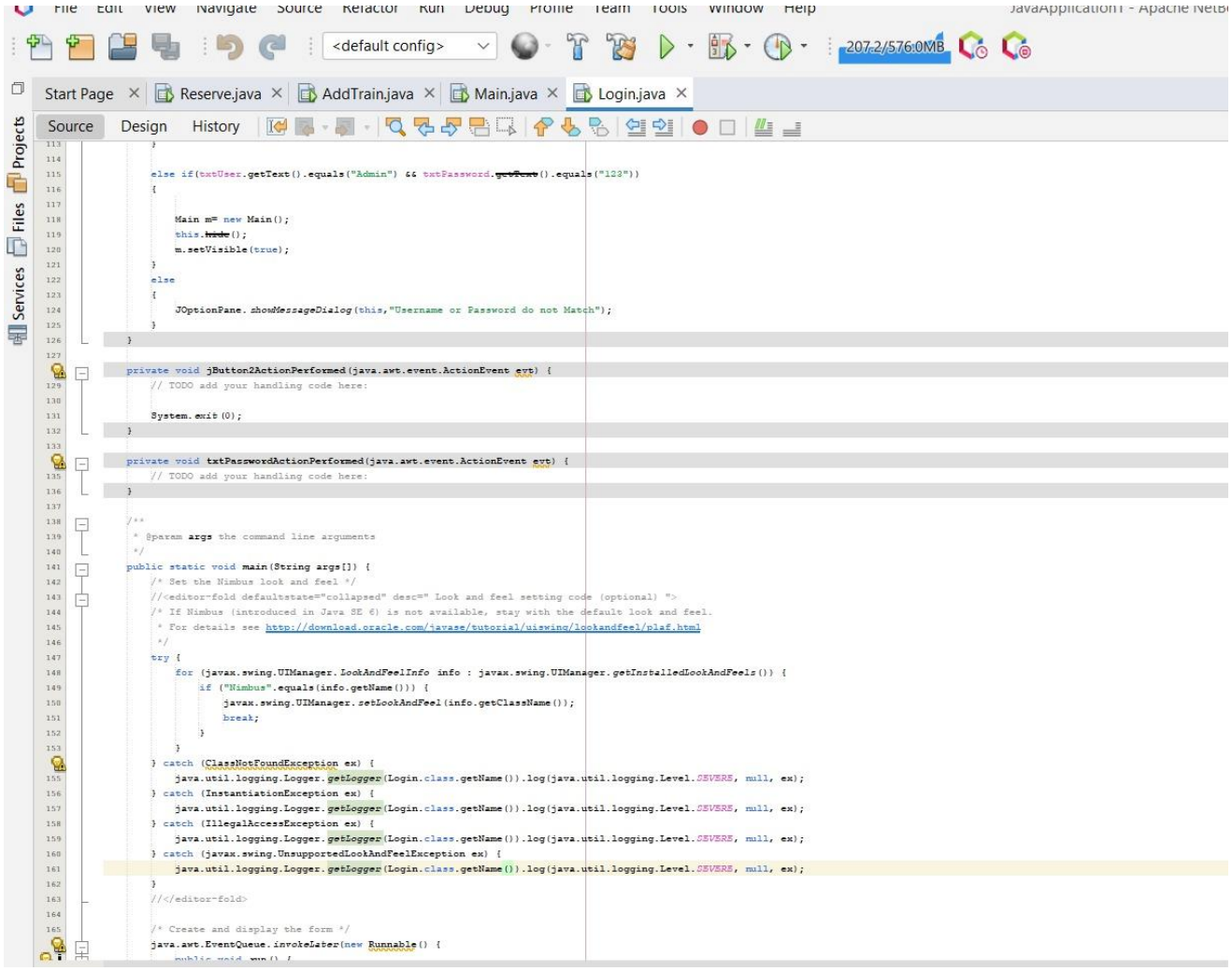
Table structure | Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	pno	int(11)			No	None			Change Drop More
2	splace	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
3	eplace	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
4	tno	int(11)			No	None			Change Drop More
5	tname	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
6	price	int(11)			No	None			Change Drop More
7	date	varchar(255)	utf8mb4_general_ci		No	None			Change Drop More
8	ticket	int(11)			No	None			Change Drop More
9	tot	int(11)			No	None			Change Drop More

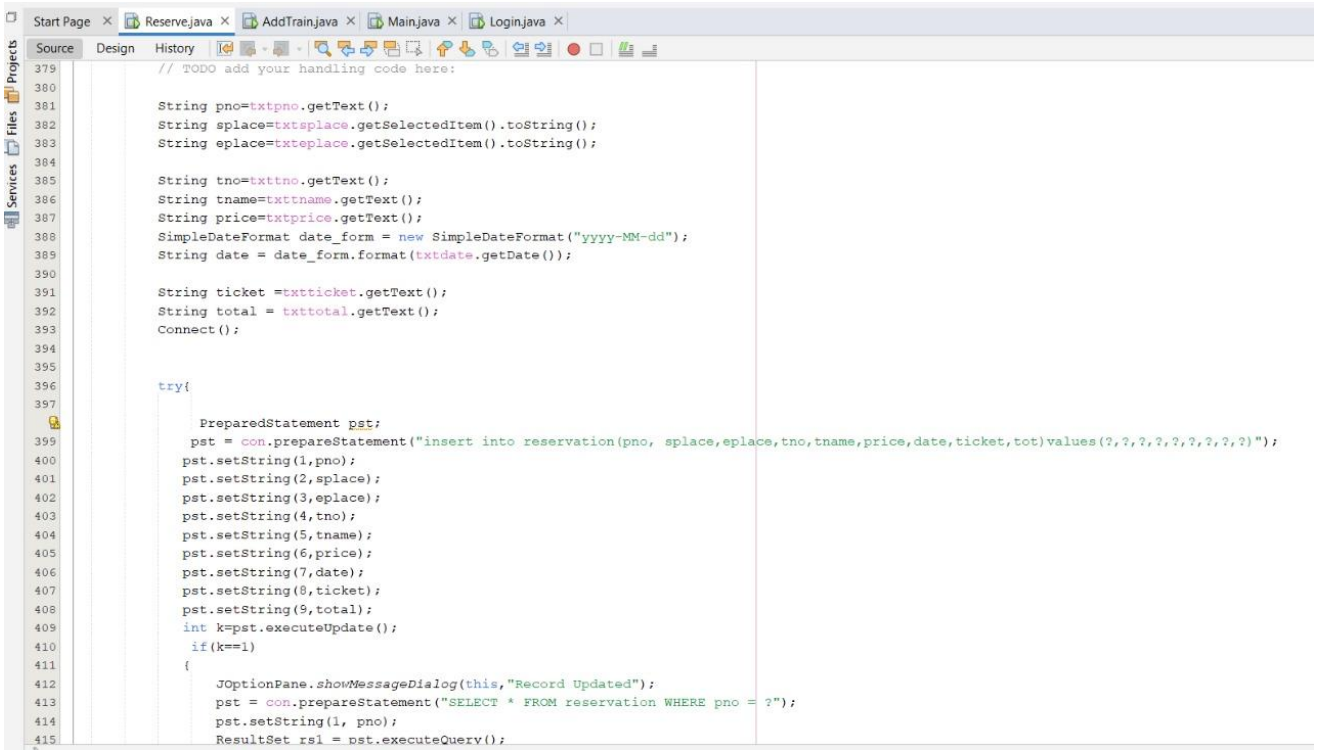
5. CODING AND IMPLIMANTATION

5.1 CODES

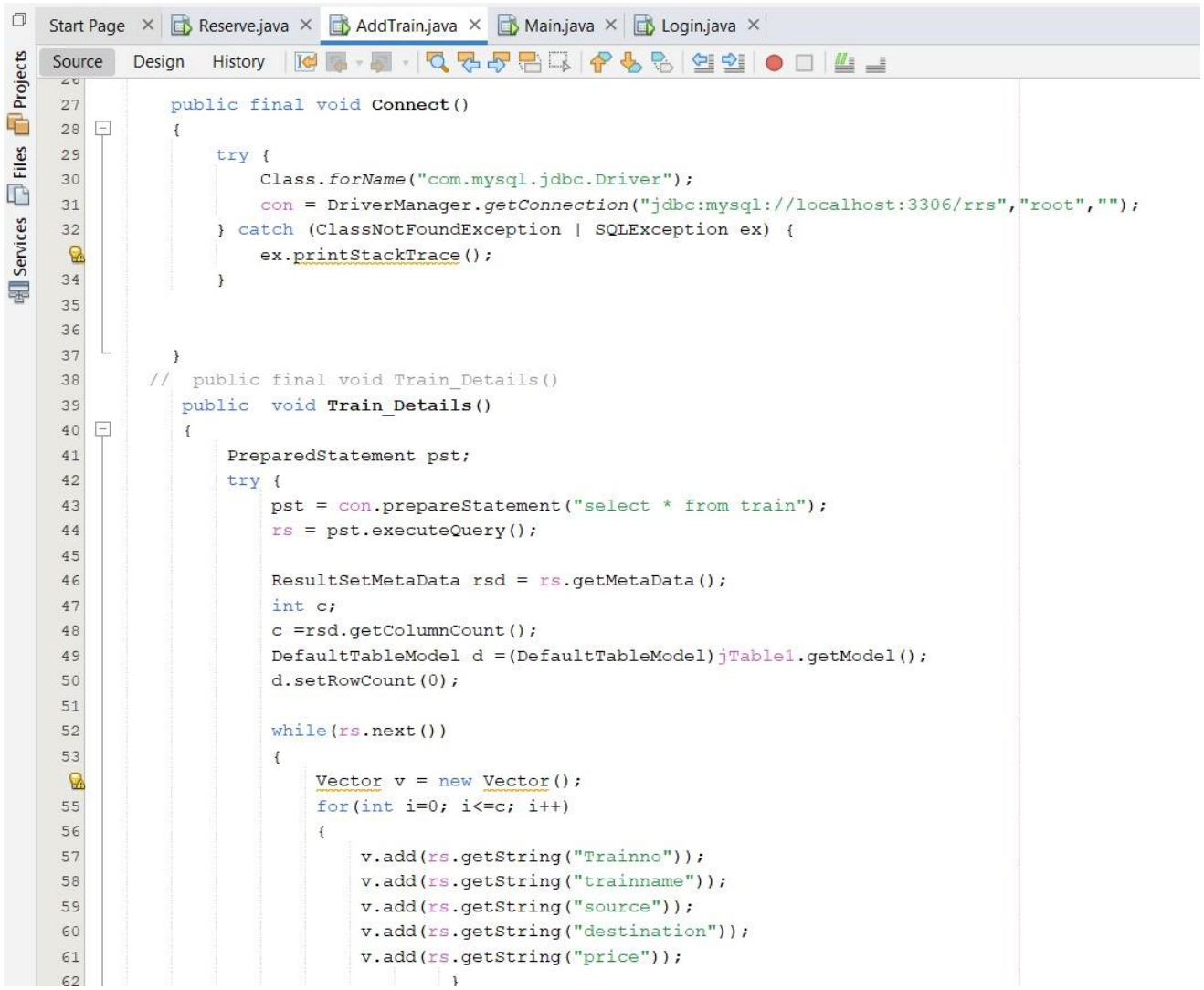




```
        }

        else if(txtUser.getText().equals("Admin") && txtPassword.getText().equals("123"))
        {

            Main m= new Main();
            this.hide();
            m.setVisible(true);
        }
        else
        {
            JOptionPane.showMessageDialog(this,"Username or Password do not Match");
        }
    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

        System.exit(0);
    }

    private void txtPasswordActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
         * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
            java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {
            java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {
            java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
            java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
```



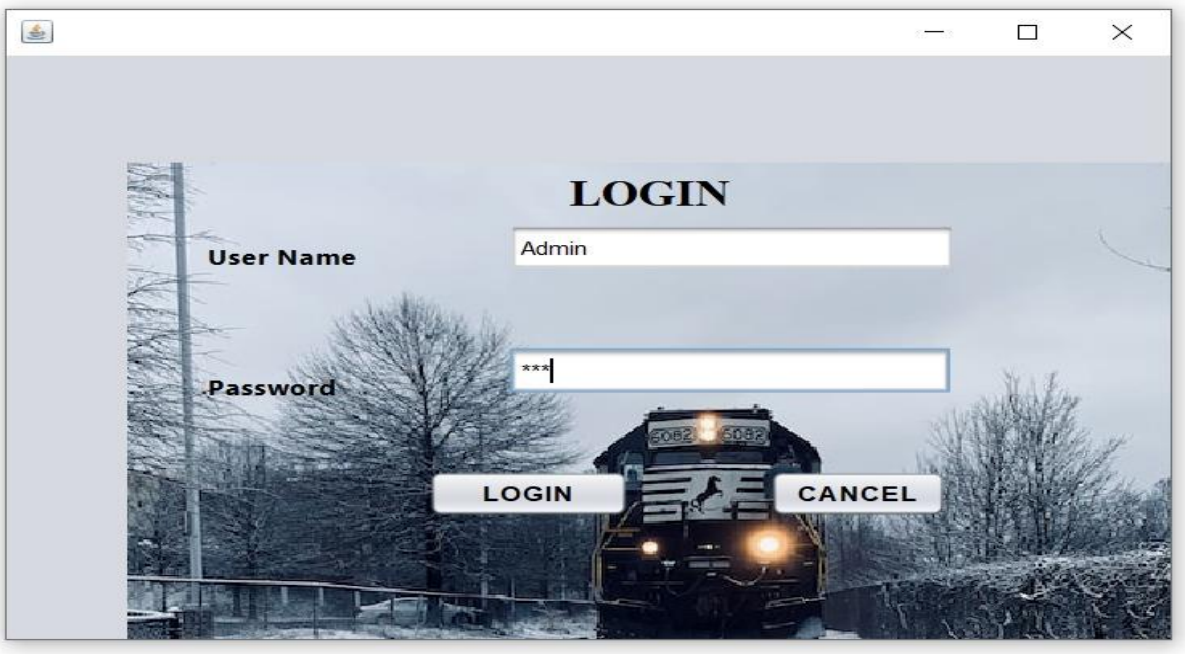
```
        // TODO add your handling code here:

        String pno=txtpno.getText();
        String splace=txtsplace.getSelectedItem().toString();
        String eplace=txteplace.getSelectedItem().toString();

        String tno=txttno.getText();
        String tname=txttname.getText();
        String price=txtprice.getText();
        SimpleDateFormat date_form = new SimpleDateFormat("yyyy-MM-dd");
        String date = date_form.format(txtdate.getDate());

        String ticket =txtticket.getText();
        String total = txttotal.getText();
        Connect();


        try{

             PreparedStatement pst;
            pst = con.prepareStatement("insert into reservation(pno, splace,eplace,tno,tname,price,date,ticket,tot)values(?,?,?,?,?,?,?,?,?)");
           pst.setString(1,pno);
           pst.setString(2,splace);
           pst.setString(3,eplace);
           pst.setString(4,tno);
           pst.setString(5,tname);
           pst.setString(6,price);
           pst.setString(7,date);
           pst.setString(8,ticket);
           pst.setString(9,total);
           int k=pst.executeUpdate();
            if(k==1)
           {
               JOptionPane.showMessageDialog(this,"Record Updated");
               pst = con.prepareStatement("SELECT * FROM reservation WHERE pno = ?");
               pst.setString(1, pno);
               ResultSet rs1 = pst.executeQuery();
```




```
    public final void Connect()
    {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection("jdbc:mysql://localhost:3306/rrs","root","");
        } catch (ClassNotFoundException | SQLException ex) {
            ex.printStackTrace();
        }


    }
//  public final void Train_Details()
    public  void Train_Details()
    {
        PreparedStatement pst;
        try {
            pst = con.prepareStatement("select * from train");
            rs = pst.executeQuery();

            ResultSetMetaData rsd = rs.getMetaData();
            int c;
            c =rsd.getColumnCount();
            DefaultTableModel d =(DefaultTableModel)jTable1.getModel();
            d.setRowCount(0);

            while(rs.next())
            {
                Vector v = new Vector();
                for(int i=0; i<=c; i++)
                {
                    v.add(rs.getString("Trainno"));
                    v.add(rs.getString("trainname"));
                    v.add(rs.getString("source"));
                    v.add(rs.getString("destination"));
                    v.add(rs.getString("price"));
                            }
```

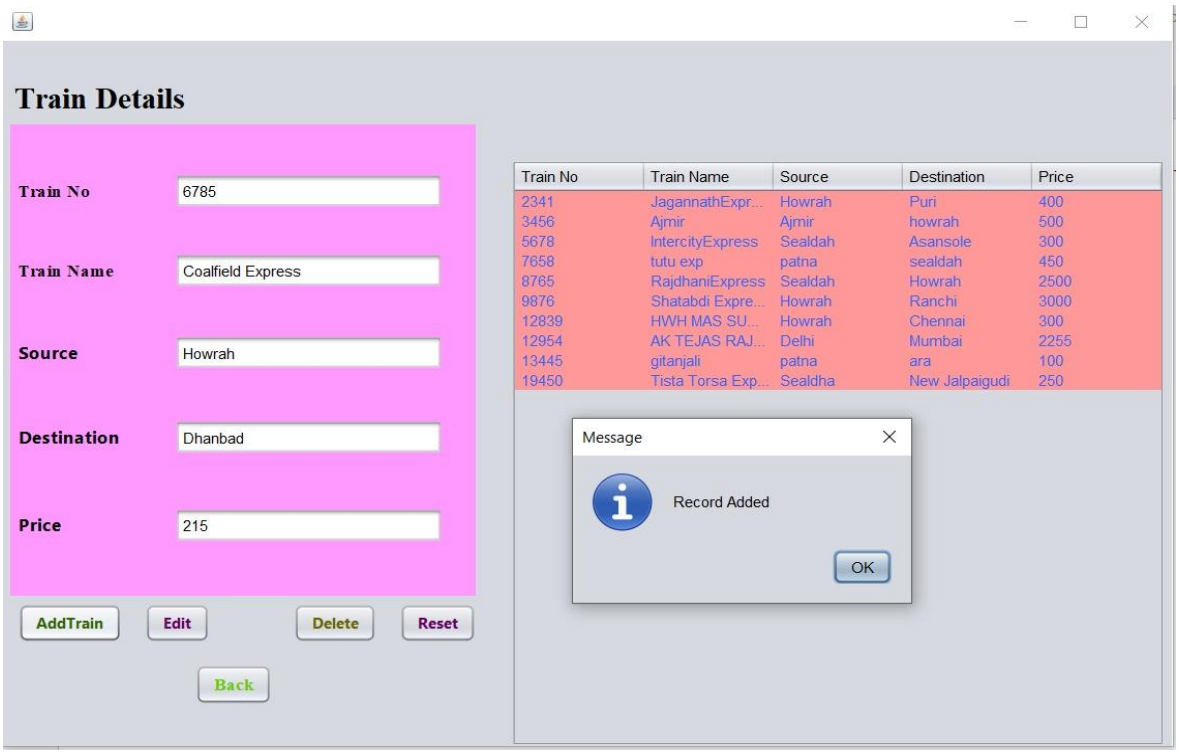
5.2 OUTPUTS

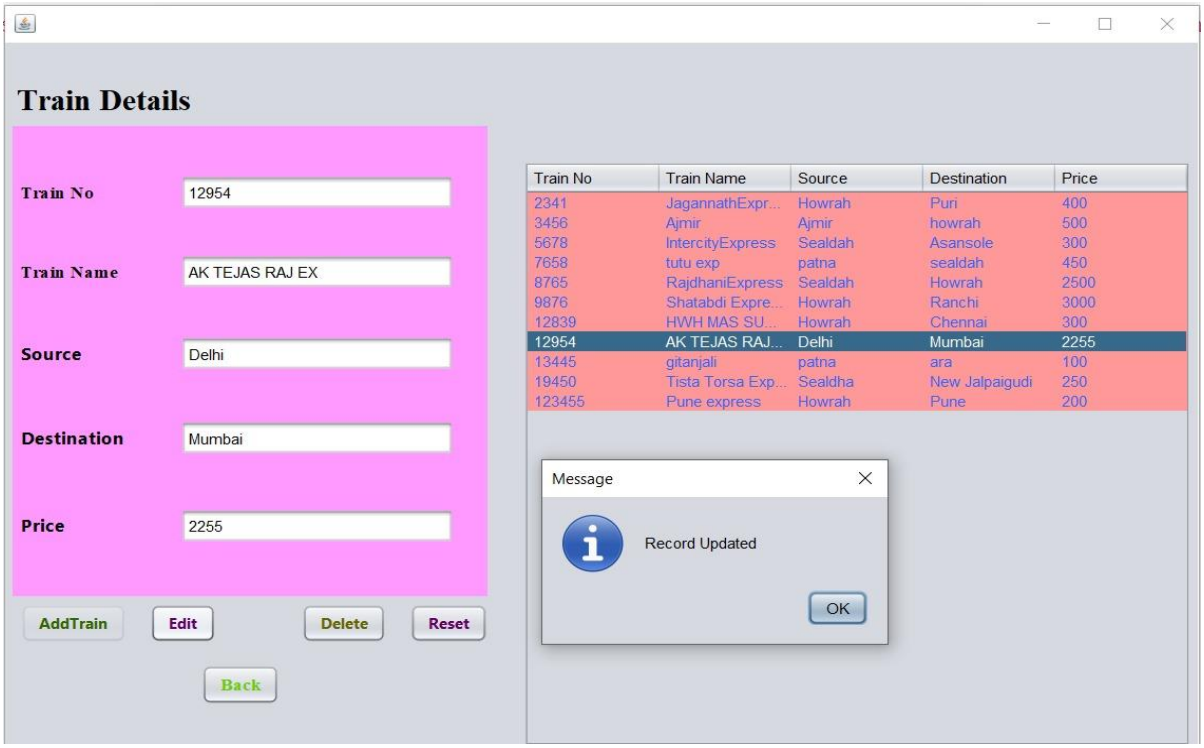
LOGIN

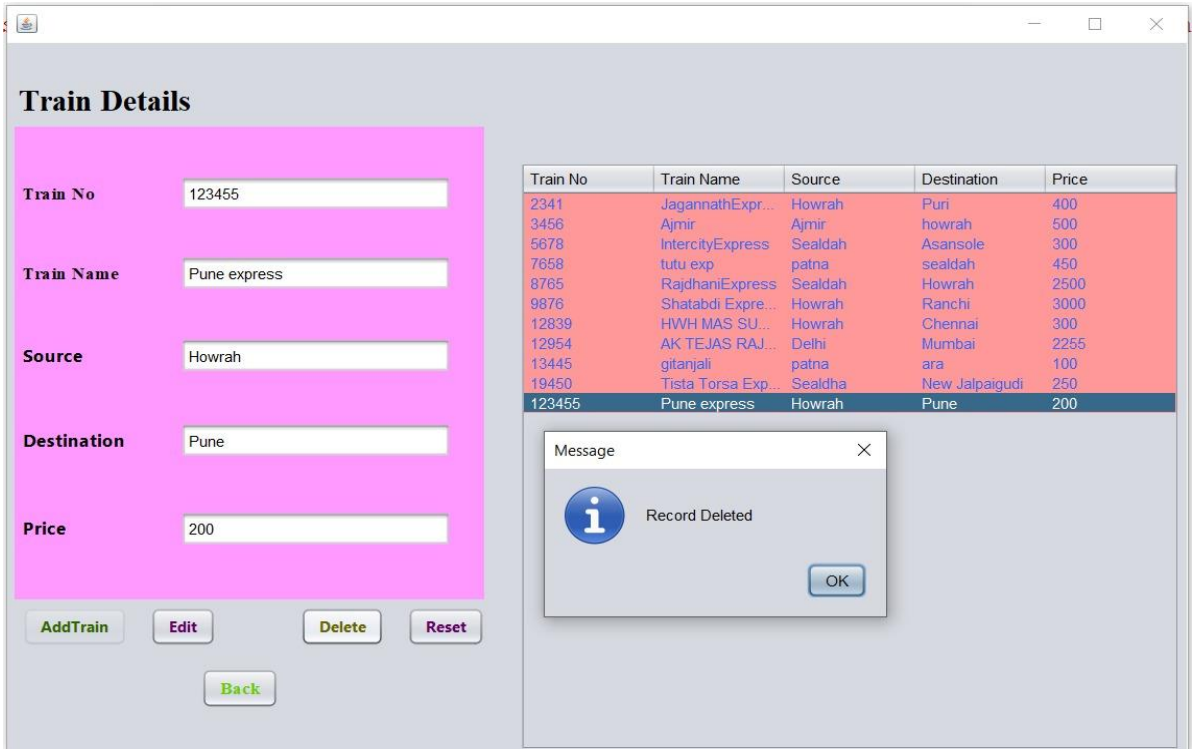


MAINFRAME

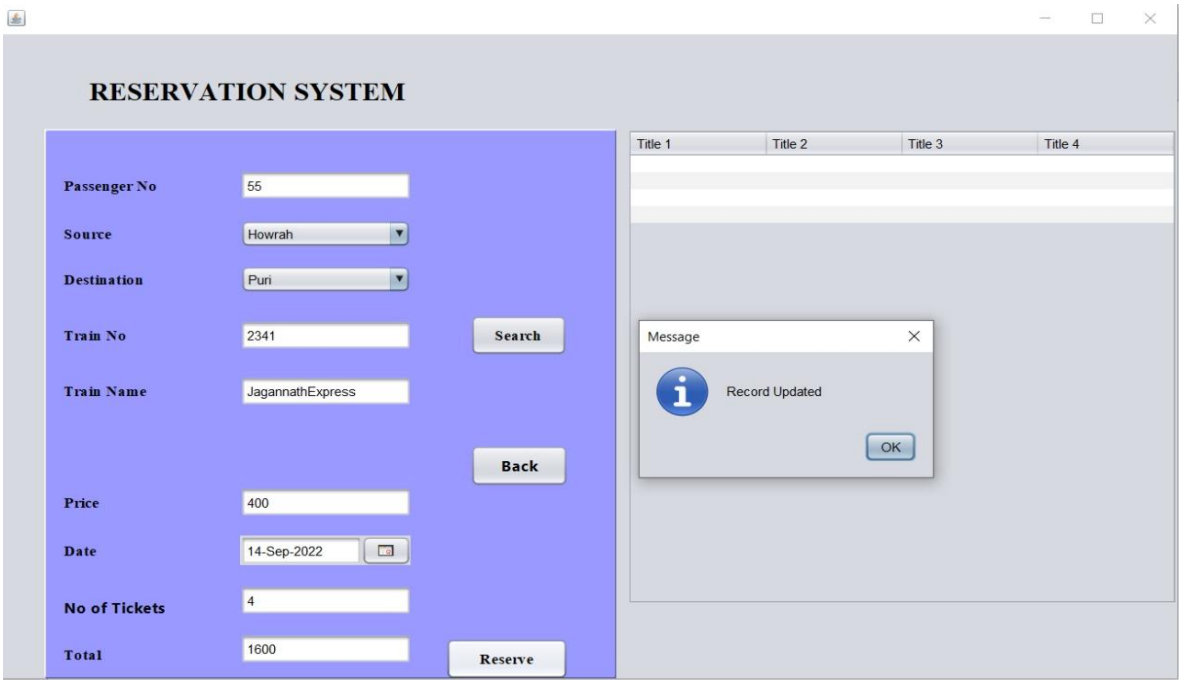
TRAIN DETAILS



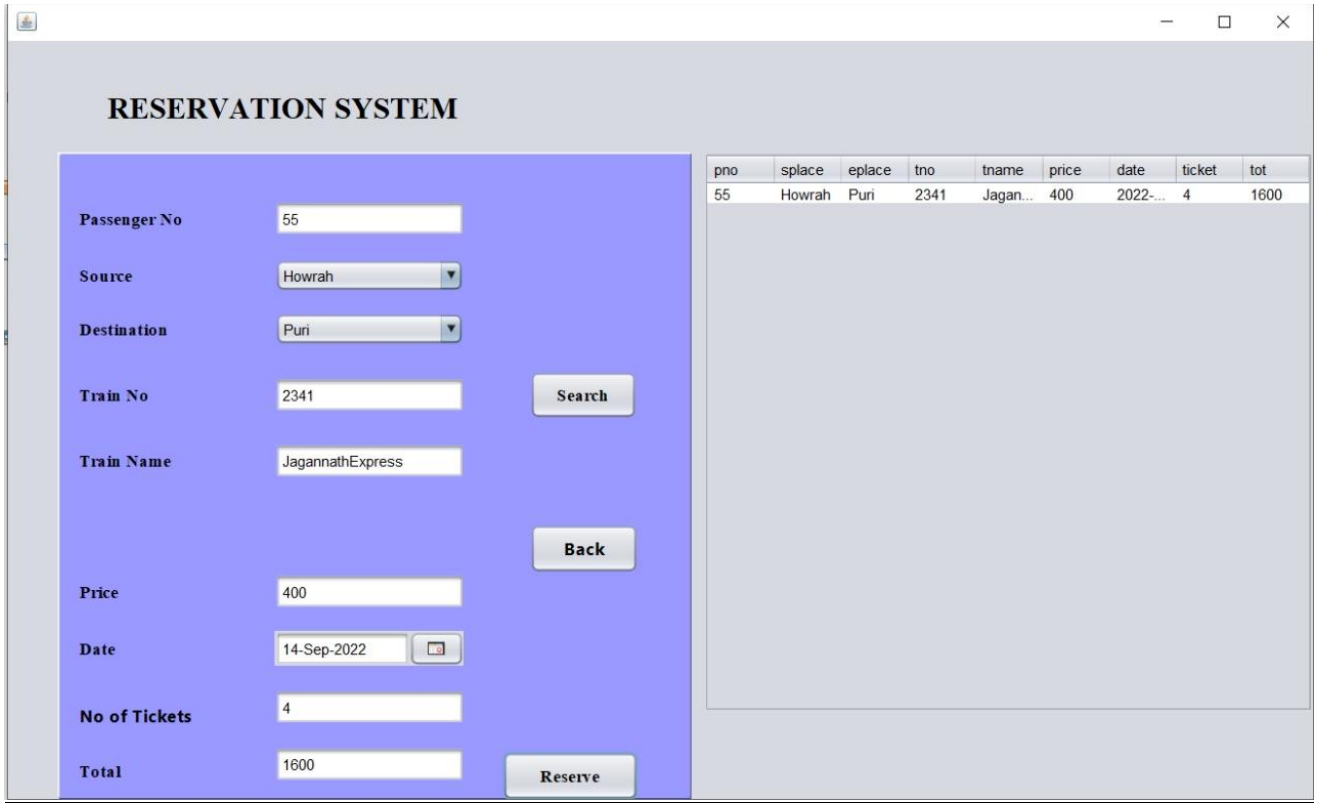




RESERVATION



RESERVATION STATUS



6. TESTING AND MAINTANANCE

6.1 TESTING

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements.

The main purpose of testing is to detect errors and error-prone areas in a system. Testing must be thorough and well-planned. A partially tested system is as bad as an untested system. And the price of an untested and under-tested system is high.

6.2 OBJECTIVE OF TESTING

The objective our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal. Our user interface to utilize these functions is designed to be user-friendly and provide easy manipulation of the tree. The application will only be used as a demonstration tool, but we would like to ensure that it could be run from a variety of platforms with little impact on performance or usability.

6.3 TEST CASES

Requirements	User	Description	Successful End Condition	Failure Condition	Achievement
Login	Admin	User Name, Password	Login Successful	Invalid Username/ Invalid Password/ Blank Not Allowed	Reservation System Page Open
Add Train	Admin	Train no, Train name, Source, Destination, Price	Record Updated	Record Failed	Details Are Added And We Can Move To Nest Frame
Reservation	Admin	Passenger no, Train no, Train name, Source, Destination, Price, Total	Reservation Successful	Record Not Found	Reservation Successful

7. SECUIRTY MEASURES

7.1 DATABASE SECURITY MEASURES

System security measure is meant to be provided to make your system reliable and secured from unauthorized user may create threats to the system. So you should follow some security measures. We have used security levels in database level at system level.

7.2 APP SECURITY MEASURES

If we talk about the system security in our proposed system we have implemented with the help of maintain the session throughout the system's use. Once a user has disconnected the internet they will not be able to access the app.

7.3 LIMITATION OF APP

- Reservation done by only the admin.
- Passenger cannot transfer his/her reservation from one train to another train.

8. FUTURE SCOPE

In Future, we would like to keep working on this project and make new addition to provide users with more advanced features and more detailed information. We have set our sights on the following additions in future:

- Addition of More Stations
- Addition of Availability Checking
- Addition of Waiting List & RAC

9. CONCLUSION

In our project "**Railway Reservation System**" we have stored all the information about the trains scheduled and the user booking tickets. This data is helpful for the applications which facilities passengers to book the train tickets and check the details of trains and their status from their place itself it avoids inconveniences of going to railway station for each and every query they get. We had considered the most important requirements only, many more features and details can be added to our project in order to obtain even more user friendly applications. These applications are already in progress and in future they can be upgraded and may become of amazing technology.

10. REFERENCES

I. "Database System Concepts" by Abraham Silberschatz, Henry F. Korth, S. Sudharshan
II. https://projectworlds.in/java
III. https://projectsgeek.com/
IV. Youtube