

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI-590014, KARNATAKA



A Mini Project Report

On

“OPTIMIZED RAILWAY RESERVATION SYSTEM”

Submitted in Partial Fulfillment of the Requirement for

“DBMS Laboratory with Mini Project -V Semester”

For the Award of Degree

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE & ENGINEERING

Submitted By:

VIKAS SHARMA (1SG17CS099)

SIDHANT KUMAR (1SG17CS084)

Under the Guidance of:

Prof. Kavya N L
Assistant Professor

Dr. Praveen Kumar
Professor



Department of Computer Science and Engineering

SAPTHAGIRI COLLEGE OF ENGINEERING

14/5, Chikkasandra, Hesarghatta Main Road

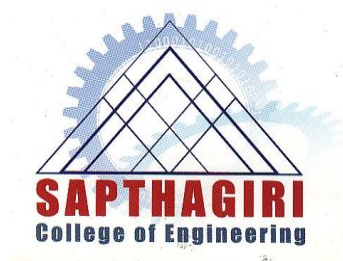
Bengaluru - 560057

2019-2020

SAPTHAGIRI COLLEGE OF ENGINEERING

14/5, Chikkasandra, Hesaraghatta Main Road, Bengaluru – 560057.

Department of Computer Science and Engineering



Certificate

Certified that the Mini Project Work entitled **“OPTIMIZED RAILWAY RESERVATION SYSTEM”** carried out by **VIKAS SHARMA (1SG17CS099) & SIDHANT KUMAR (1SG17CS084)**, bonafide students of **Sapthagiri College of Engineering**, in partial fulfillment for the award of **Bachelor of Engineering** degree in **Computer Science and Engineering** of **Visvesvaraya Technological University, Belagavi** during the academic year 2019-2020. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The mini-project report has been approved as it satisfies the academic requirements in respect of **DBMS Laboratory with Mini Project (17CSL58)** prescribed for the said Degree.

Signature of the Guide
Prof. Kavya N L
Assistant Professor

Signature of the Guide
Dr. Praveen Kumar
Professor

Signature of the HOD
Dr. Kamalakshi Naganna
Professor & Head

EXTERNAL EXAMINATION

Name of the Examiners

Signature with Date

1. _____

2. _____

CHAPTER 1

INTRODUCTION

ARCHITECTURE OF DBMS

A database is a collection of related data. However, the common use of the term database is usually more restricted. A database has the following implicit properties:

- A database represents some aspect of the real world, sometimes called the miniworld or the universe of discourse (UOD). Changes to the miniworld are reflected in the database.
- A database is a logically coherent collection of data with some inherent meaning. A random assortment of data cannot correctly be referred to as a database.
- A database is designed, built, and populated with data for a specific purpose.

It has an intended group of users and some preconceived applications in which these users are interested. In other words, a database has some source from which data is derived, some degree of interaction with events in the real world, and an audience that is actively interested in its contents.

A Database Management System (DBMS) is a computerized system that enables users to create and maintain a database. The DBMS is a general-purpose software system that facilitates the processes of defining, constructing, manipulating, and sharing databases among various users and applications. Defining a database involves specifying the data types, structures, and constraints of the data to be stored in the database. The database definition or descriptive information is also stored by the DBMS in the form of a database catalog or dictionary; it is called meta-data. Constructing the database is the process of storing the data on some storage medium that is controlled by the DBMS. Manipulating a database includes functions such as querying the database to retrieve specific data, updating the database to reflect changes in the miniworld, and generating reports from the data. Sharing a database allows multiple users and programs to access the database simultaneously.

In this architecture, schemas can be defined at the following three levels:

- **INTERNAL LEVEL**

The internal level has an internal schema, which describes the physical storage structure of the database. The internal schema uses a physical data model and describes the complete details of data storage and access paths for the database.

- **CONCEPTUAL LEVEL**

The conceptual level has a conceptual schema, which describes the structure of the whole database for a community of users. The conceptual schema hides the details of physical storage structures and concentrates on describing entities, data types, relationships, user operations, and constraints. Usually, a representational data model is used to describe the conceptual schema when a database system is implemented. This implementation conceptual schema is often based on a conceptual schema design in a high-level data model.

- **EXTERNAL OR VIEW LEVEL**

The external or view level includes a number of external schemas or user views. Each external schema describes the part of the database that a particular user group is interested in and hides the rest of the database from that user group. As in the previous level, each external schema is typically implemented using a representational data model, possibly based on an external schema design in a high-level conceptual data model.

DBMS ARCHITECTURE

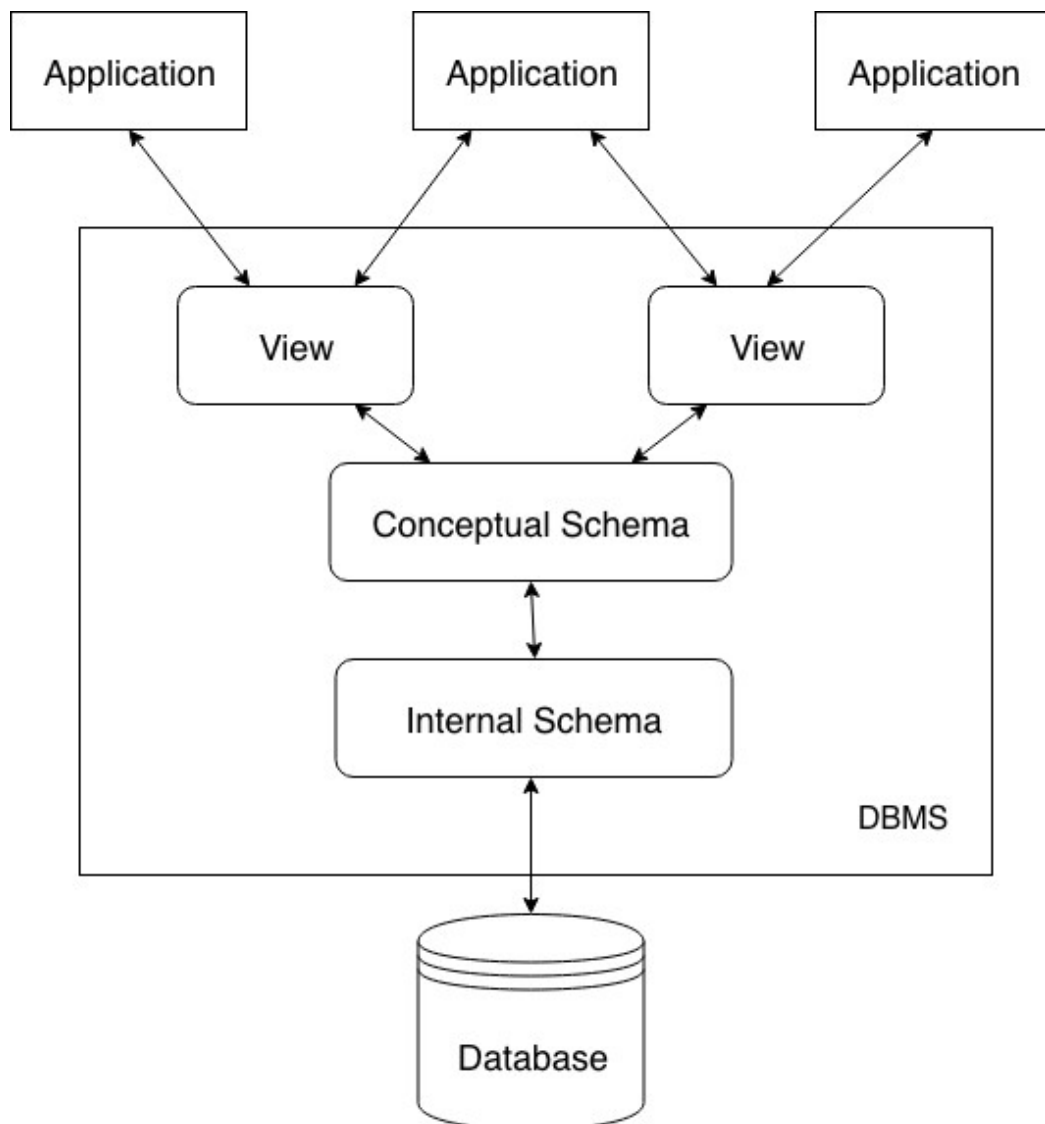


Figure 1.1 Three Schema Architecture

The above Figure 1.1 describes the three-schema architecture divided into three levels to relate the end users with the stored database.

ORACLE/SQL

A database is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information. A database server is the key to information management. In general, a server reliably manages a large amount of data in a multiuser environment so that many users can concurrently access the same data. A database server also prevents unauthorized access and provides efficient solutions for failure recovery.

Oracle Database is the first database designed for enterprise grid computing, the most flexible and cost-effective way to manage information and applications. Enterprise grid computing creates large pools of industry-standard, modular storage and servers. With this architecture, each new system can be rapidly provisioned from the pool of components. There is no need to provide extra hardware to support peak workloads, because capacity can be easily added or reallocated from the resource pools as needed.

The database has physical structures and logical structures. Because the physical and logical structures are separate, the physical storage of data can be managed without affecting access to logical storage structures.

OVERVIEW OF THE PROJECT

The objective of the online railway ticket reservation system Project is to design software to fully automate the process of issuing a railway ticket. That is:-

1. To create a database of the trains
2. To search the trains it's arrival and departure time ,distance between source and destination.
3. To check the availability of the ticket.
4. To calculate fare.
5. To book the ticket.
6. To cancel the ticket if necessary.

PROBLEM STATEMENT

To maintain and manipulate the data stored in Railway Reservation System.

OBJECTIVES OF THE PROJECT

- To be able to perform CRUD (create,read,update,delete) operations on the given reservation database as well as the registered details of the users.
- To filter search results for a particular train.
- To be able to know whether the ticket has been booked or not.

CHAPTER 2

SYSTEM DESIGN AND METHODOLOGY

System design is a process of planning a new system or to replace the existing system. Simply ,system design is a blueprint for a building, it specifies all the features that are to be in the finished product.

SYSTEM ARCHITECTURE

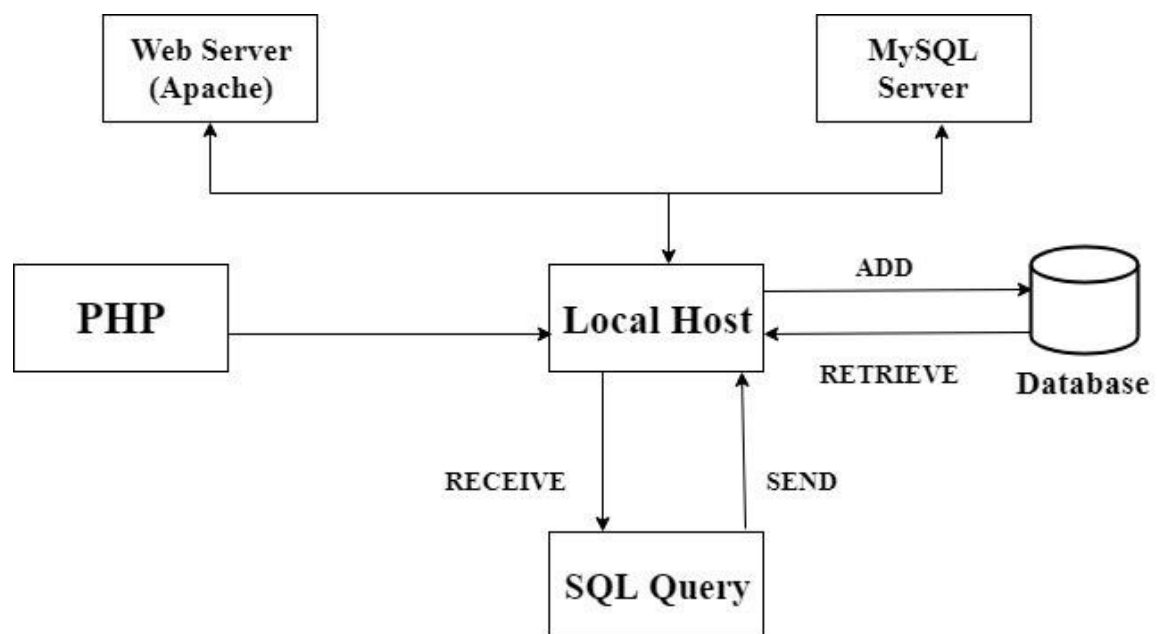


Figure 2.1 System Architechture for optimized railway reservation system

The aboveFigure 2.1 represents the System Architecture of the mini project. It mainly contains the framework called PHP and a centralized Database from which data can be accessed by end users.

Administrative access is granted to the admin wherein they can login using the correct username and password.

Once the login is successful the index page is seen wherein the admin can optimize the booking list.

ER DIAGRAM

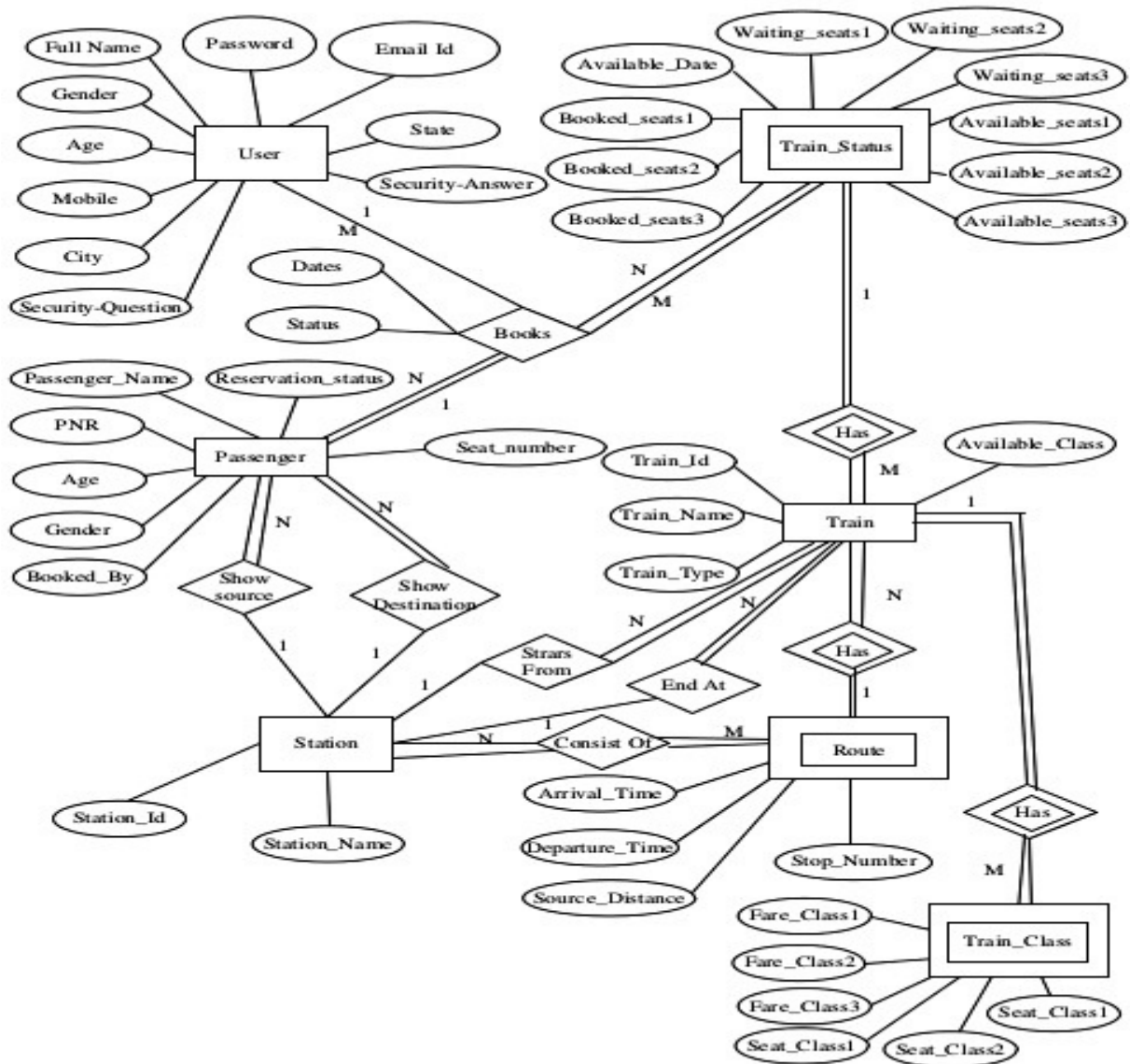


Figure 2.2 ER Diagram

The above Figure 2.2 describes the relationship diagram between different entities constituting a database system.

An Entity-Relationship Diagram (ERD) is a data modelling technique that illustrates an information system's entities and relationships between those entities. An ERD is a conceptual and a representational model of data used to represent the entity framework infrastructure.

The ER diagram shown in Figure 2.2 consists of 4 entities namely,

1. admin
2. train
3. route
4. user
5. booking
6. station
7. status

- The **Admin** table contains following attributes: **admin_ID**, **username**, **password** through which admin can login.
- The **train** table contains the following attributes **train_id**, **name**, **type**, **seat_class**, **fare_class**, **available_days** which gives a description about each train along with the id and train date.
- The **route** table contains following attributes: **train_id**, **stop_number**, **source_distance**, **arrival_time**, **departure_time** gives a brief information about the route.
- The **Booking** table contains following attributes: **booking_id**, **confirmation**, **department**, **destination**, **trainid**, **email**, **finished**, **mobile**, **username** which gives the complete details while booking a new ticket for a ride.
- The **Station** table contains following attributes: **station_id**, **station_name** gives all the information about the station.
- The **Cost** table contains following attributes: **booking_id**, **cost**, **username** through which the data about fare and distance of the entire travel is included.

- The **User** table contains following attributes : **first_name**, **last_name**, **admin**, **username**, **Email**, **Password** into which the registered user's data can be stored.

SCHEMA DIAGRAM

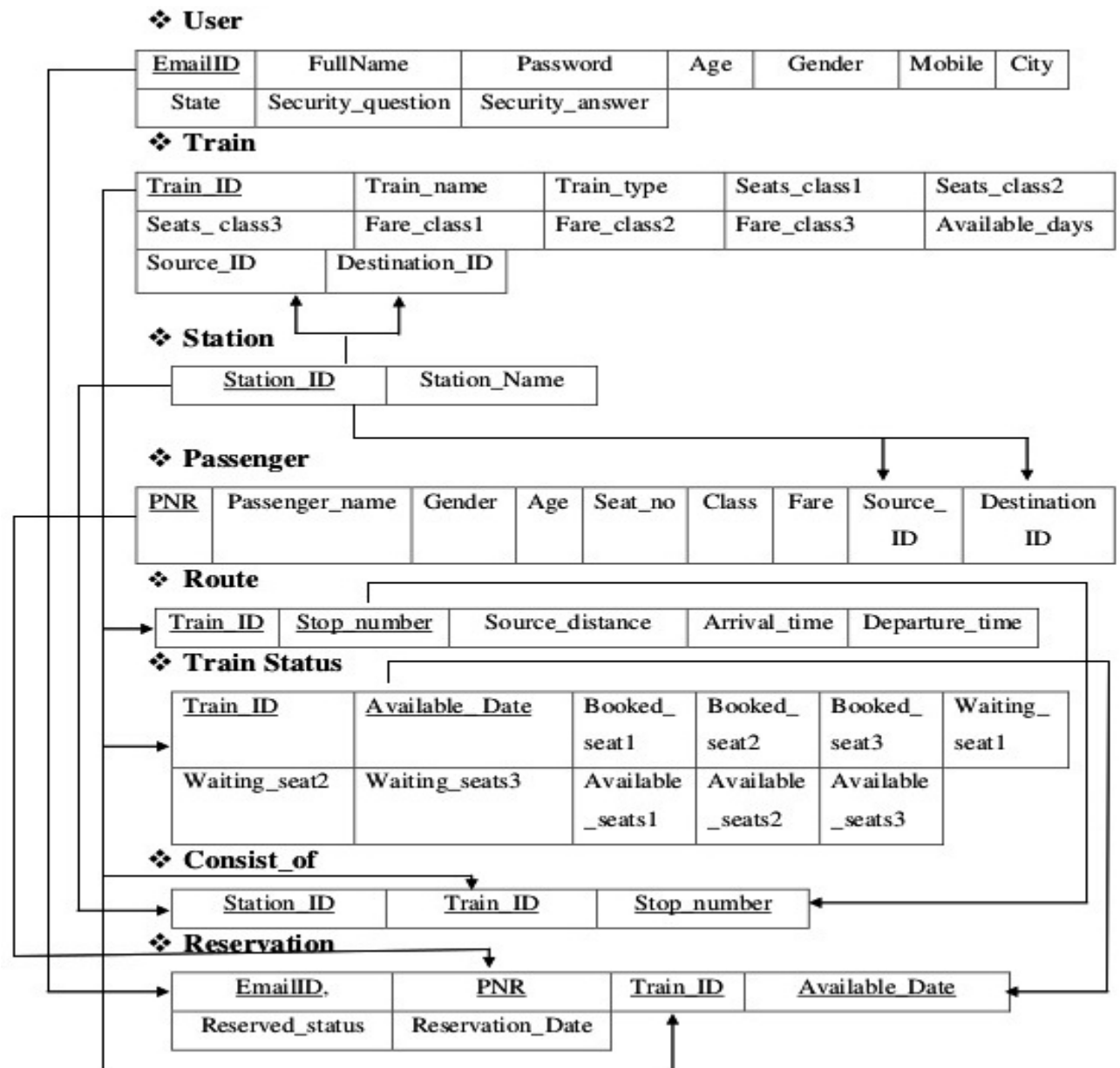


Figure 2.3 Schema Diagram

The above Figure 2.3 describes the relationship in a schematic diagram along with each entity attributes relation with different entity attributes.

The logical mapping of ER diagram to a relational model is as shown in the fig 2.3

ALGORITHMS

STORED PROCEDURE

Step 1 : BEGIN

Step 2 : DECLARE **trainRoutine**.

Step 3 : Store all from booking where (**TrainID=tid**) and displays them on the screen.

Step 4 : DISPLAY trainRoutine.

Step 5 : END

Description: The stored procedure basically stores the function which does the mathematical relational operation for getting the different values (trainID) which is equal to the given value(tid). The given value satisfies the condition that is the trainID for tid mentioned in the bookingtable. The trainID for the needed tid can be displayed on the screen from the table.

TRIGGER

Step 1 : BEGIN

Step 2 : INSERT/UPDATE/DELETE into trainrecord.

Step 3 : Insert the actions performed with ID, TrainName and Train Date into **traintable**.

Step 4 : END

Description: The trigger keeps track of the action performed on the Trains record. That is the deleted Brands of an old brand of the already stored record. The deletedTrain stored as the delete action is performed on the train name and Present date at which the operation is performed.

CHAPTER 3

MODULE IMPLEMENTATION

To implement this mini project we have used MYSQL for the backend and HTML, Java for the frontend (GUI).

FEATURES OF JAVA ARE:

- It is a very **easy and simple** language. It can be easily understood and modified.
- It is **object oriented language**.
- It is very **secure ,robust**.
- It is **Architecture-neutral**.
- It is **platform-independent** because it can be displayed on any platform like Windows, Linux, Macintosh etc.
- It facilitates the programmer to add **Graphics, Videos, and Sound** to the web pages which makes it more attractive and interactive.

FEATURES OF MYSQL ARE:

- It is an open-source relational database management system (RDBMS).
- The MYSQL development mini project has made its source code available under the terms of the General Public License (GNU), as well as under a variety of proprietary agreements.

- MySQL was owned and sponsored by a single for-profit firm, the Swedish company MYSQL AB, now owned by Oracle Corporation.
- The MYSQL server package will install the MYSQL database server which can interact with using a MYSQL client. User can use the MYSQL client to send commands to any MYSQL server on a remote computer. The MYSQL server is used to persist the data and provide a query interface for it (SQL).
- The MYSQL client purpose is to allow you to use that query interface. The client package also comes with utilities that allows you to easily backup/restore data and administer the server.
- MYSQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MYSQL, Perl / PHP / Python".
- Applications that use the MYSQL database include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, and Drupal. MYSQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.

3.1 MODULE DESCRIPTION

The modules included in this mini project are:

1. USER or CUSTOMER REGISTRATION

Input: Name, Phone No, Email ID, Password.

Output: Above data will be stored in users table.

Description: After filling up all the required fields and clicking on the **signup** button, the inserted data will create a new detail record for that user through which their booking record will be created later on for checking the booking status by the admin. Multiple registration for the same user is restricted by the Email they registered before.

2. LOGIN PAGE

- **USER LOGIN**

Input: Email, Password.

Output: The portal for vehicle booking, queries and contact us pages can be accessed.

Description: After inserting all the required fields and clicking on the **login** button, if any credentials do not match with the user's detail table it will **trigger** an error in inserting credentials. If the credentials are correct, it will redirect to the Home page where they can book vehicles, access a form for the admin to get in touch with the users.

- **ADMIN LOGIN**

Input: Admin ID, Password.

Output: The Admin Panel can be accessed.

Description: After inserting all the required fields and clicking on the **login** button, if any credentials do not match with the admin's detail table it will **trigger** an error in inserting credentials. Otherwise it will redirect to the Admin Panel. Through the Admin Panel the Admin can modify the registered customer's detail into their registered details record. Along with that the Admin can have a view onto the **bookings, queries and messages** sent by the customer and the **triggered logs** actions done by the admin on the brand record.

3. TICKET BOOKING

Input: Name ,Date,Train name, Email, Mobile.

Output: A booking has been successfully requested.

Description: The user has to book the ticket by entering his name, date, train name and he has to submit the form.

4. FINALIZING THE STATUS OF TICKET

Input: train name, date, admin password.

Output: confirmation status is obtained for list of all passengers.

Description: The admin has to enter the train name , date , admin password to finalize the tickets. confirmation status is obtained for list of all passengers.

CHAPTER 4

RESULTS AND SCREENSHOTS

HOME PAGE

A screenshot of a web browser window displaying a login page. The page has a white background. At the top, there is a blue header bar with the text "LOGIN AS" in white. Below the header, there are two radio buttons: "ADMIN" and "USER". The "ADMIN" radio button is selected. At the bottom of the page, there are two buttons: "SUBMIT" and "WANT TO KNOW ABOUT US". The browser window has a standard title bar with a minimize, maximize, and close button.

Figure 4.1 Homepage

The above Figure 4.1 Homepage gives us the choice how should we go forward.

ABOUT US

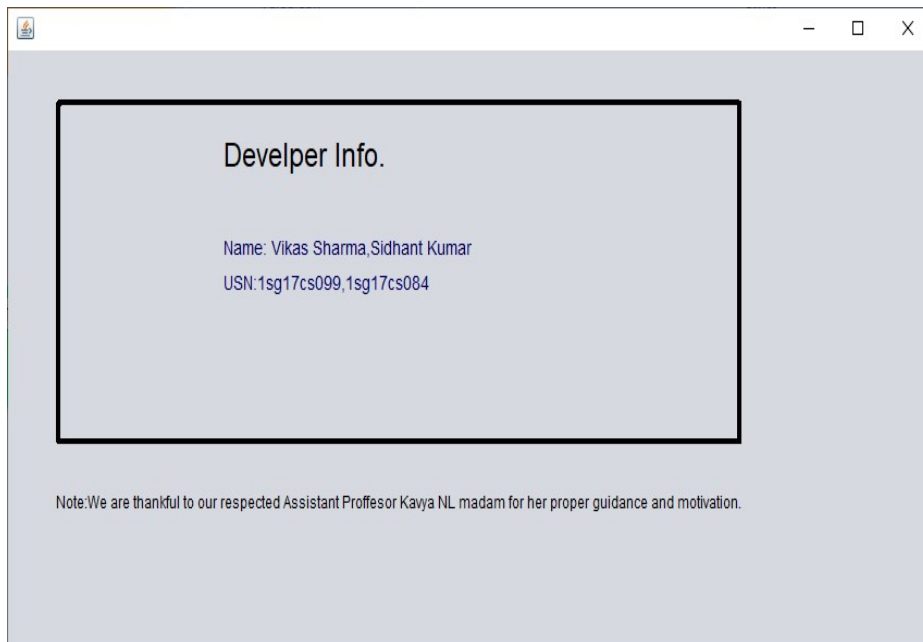


Figure 4.2 About us

The above Figure 4.2 About us gives us the information about the developer information.

USER LOGIN

A screenshot of a web application window titled "USER LOGIN". The window has a light gray background. At the top, there is a solid orange horizontal bar. Below this, there is a light gray section containing the label "USERID:" followed by a light blue text input field. Below this, there is a white section containing the label "PASSWORD:" followed by a light blue text input field. At the bottom, there is a solid green horizontal bar containing two white buttons with black text: "SIGN IN" and "SIGN UP". The window has standard minimize, maximize, and close buttons in the top right corner.

Figure 4.3 User Login

The above Figure 4.3 User Login lets us enter our login id and password to validate us and

allow us to the booking page.

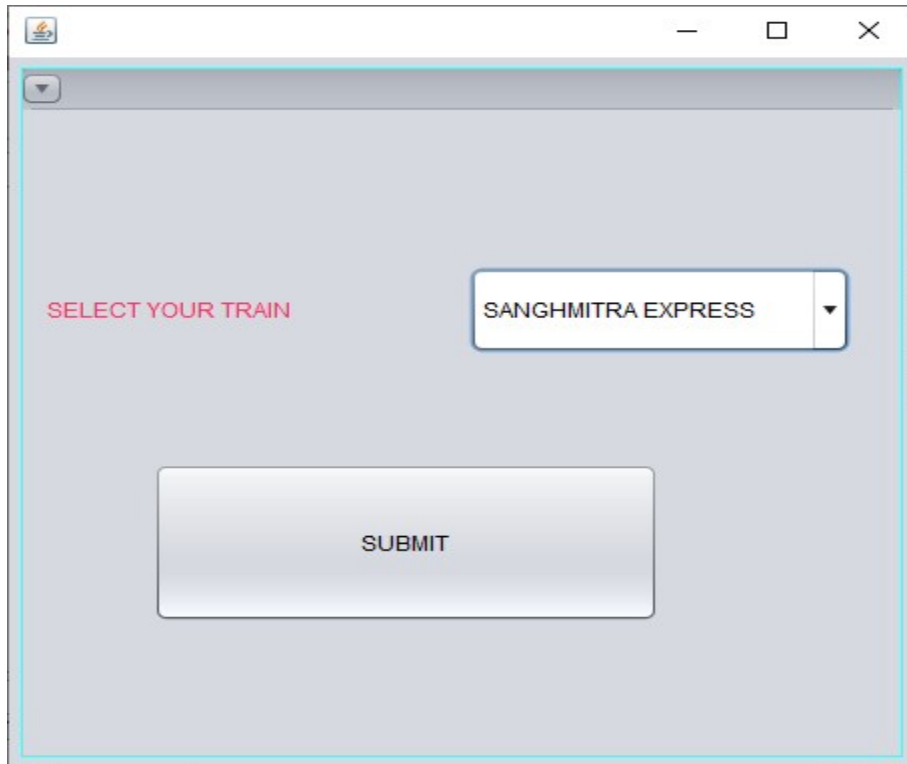
SIGN UP



Figure 4.4 Sign Up

This page lets us sign up so that we can register ourselves on this portal.

SELECT TRAIN

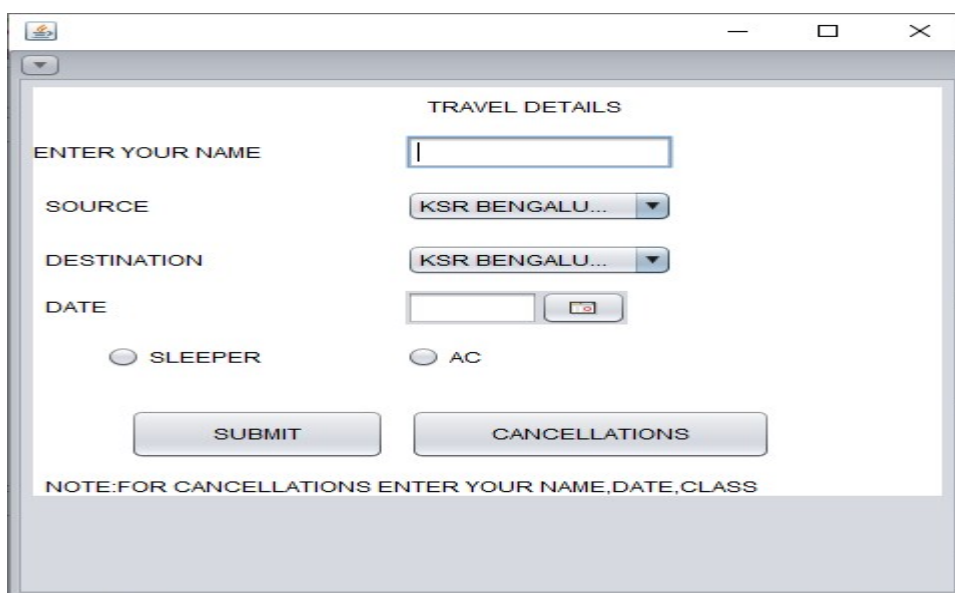


The screenshot shows a web application window titled "SELECT YOUR TRAIN". Inside the window, there is a text input field containing "SANGHMITRA EXPRESS" with a dropdown arrow on the right. Below this field is a large, light blue button labeled "SUBMIT". The window has a standard OS-style title bar with minimize, maximize, and close buttons.

Figure 4.5 Select Train

This page lets us choose the train of our choice for the purpose of ticket booking.

FINAL BOOKING/CANCELLATIONS



The screenshot shows a web application window titled "FINAL BOOKING/CANCELLATIONS". The window contains a form with the following fields and controls:

- ENTER YOUR NAME:** A text input field.
- SOURCE:** A dropdown menu showing "KSR BENGALU..." with a downward arrow.
- DESTINATION:** A dropdown menu showing "KSR BENGALU..." with a downward arrow.
- DATE:** A text input field next to a calendar icon.
- Class Selection:** Two radio buttons labeled "SLEEPER" and "AC".
- Buttons:** Two buttons labeled "SUBMIT" and "CANCELLATIONS".
- Note:** A text label at the bottom stating "NOTE:FOR CANCELLATIONS ENTER YOUR NAME,DATE,CLASS".

The window has a standard OS-style title bar with minimize, maximize, and close buttons.

Figure 4.6 Final Booking/Cancellations

This page lets us book a ticket or cancel a ticket on the given date by choosing source station and destination date on a given date.

FINALIZATION BY ADMIN:

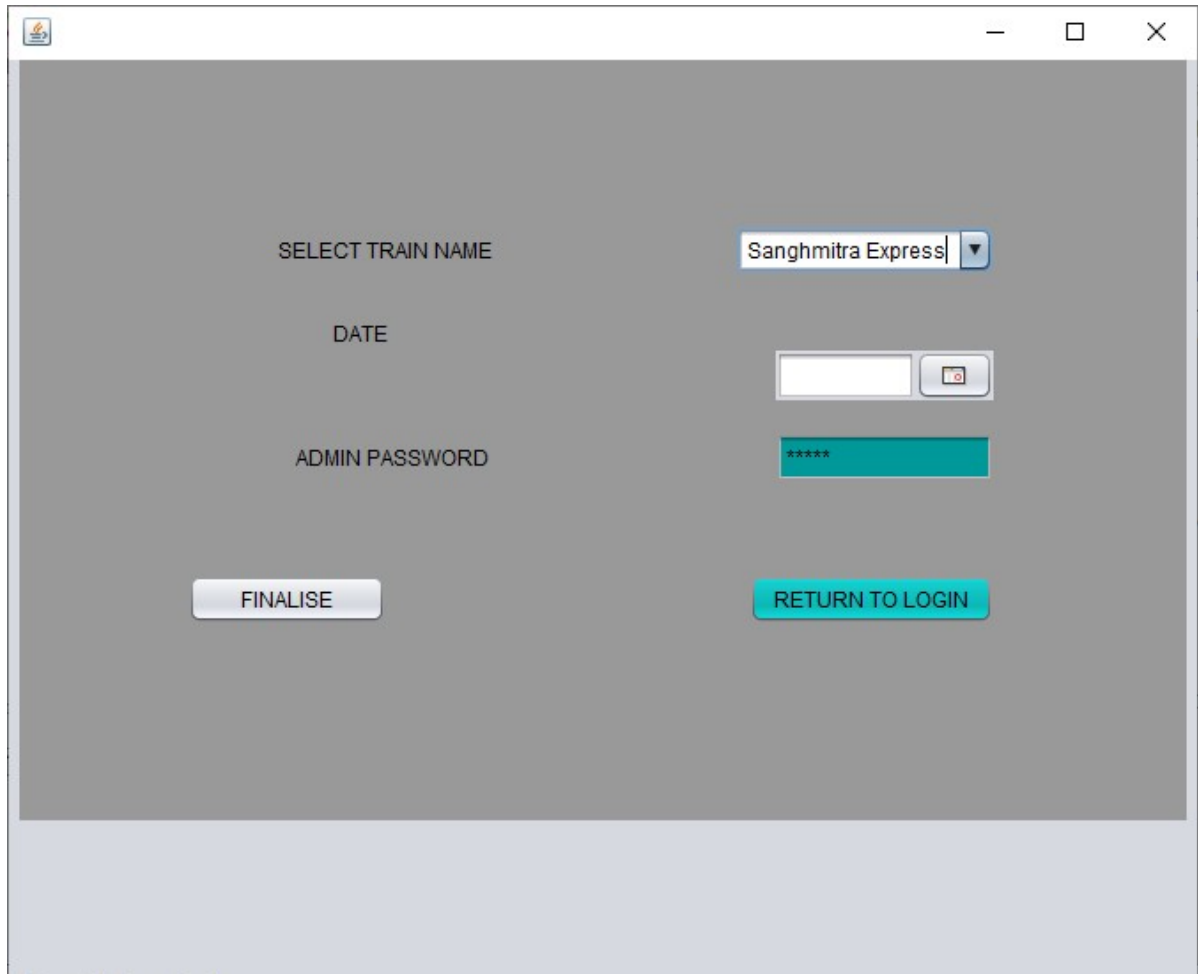
A screenshot of a web application window titled "Optimized Railway Reservation System". The window has a standard Windows-style title bar with minimize, maximize, and close buttons. The main content area has a light gray background. It contains three labels: "SELECT TRAIN NAME", "DATE", and "ADMIN PASSWORD". To the right of "SELECT TRAIN NAME" is a dropdown menu showing "Sanghmitra Express" with a downward arrow. To the right of "DATE" is a date input field with a calendar icon. To the right of "ADMIN PASSWORD" is a password input field with six asterisks. At the bottom left is a "FINALISE" button, and at the bottom right is a "RETURN TO LOGIN" button.

Figure 4.7 Finalization by admin

The admin finalize the ticket by entering the train name , date , password.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

The main objective of this mini project is to generate booking of a tickets and getting a conformation and to manage booking , **optimizing**. Several user-friendly codes have been adopted. The system shall prove to be a powerful package which satisfies all the requirements of the user.

FUTURE WORKS

1. The system shall host the platform on online servers to make it accessible worldwide.
2. The system shall integrate multiple load balancers to distribute loads on system.
3. The system shall be implemented for the different aspects related to different classes and coaches.
4. The system shall include a master-slave database structure to reduce overload on databases on regular basis on different servers.

CHAPTER 6

BIBLIOGRAPHY

1. <https://www.tutorialspoint.com/mysql/>
2. <https://www.apachefriends.org/>
3. <https://www.youtube.com>
4. <https://stackoverflow.com/>
5. <https://www.w3schools.com/>
6. <https://getbootstrap.com/docs/4.0/getting-started/introduction/>
7. <https://fontawesome.com/>
8. [Database Management Systems \(DBMS\) by prof. EIMASRI AND NAVATHE](#)