

Project Report | Documentation



MINI PROJECT

Metro System Database Project

Ankit Chouhan
161112051

Jishan Shaikh
161112013

This page intentionally left blank

Maulana Azad National Institute of Technology
(An Institute of National Importance)
Bhopal – 462003 (India)



Department of Computer Science & Engineering

M I N I P R O J E C T

Metro System Database Project

Submitted by : Ankit Chouhan (Scholar no. 161112051)
Jishan Shaikh (Scholar no. 161112013)

Submission date : April 13, 2018

Supervisor : Prof. Sweta Jain
Department of Computer Science & Engg.
MANIT, Bhopal (Madhya Pradesh, India)

Subject : Database Management System Laboratory
CSE-229, IV Sem (B.Tech. in CSE)

Session : Even Semester 2018

This page intentionally left blank

Preface

This project titled Metro System Database Project is one of the academic projects we (as a team of 2 members) take in DataBase Management System Laboratory course (4th Sem) of Bachelor in Technology (Computer Science & Engineering), here at Maulana Azad National Institute of Technology, Bhopal. This project was started in February 2018 and merely ends at the submission date by April 13. (Work in off days)

The complete report is organized into small chapters, sections, subsections, appendices, etc. Snapshots and screenshots are taken for clear idea of what's going. We've tried to cover all information regarding project to be assemble in report but still SRS document, Plan document, Preliminary RMMM plan doesnot find their place in the report. Further they can be merged into report in future as per requirements.

The foundations of project started in January 18, when new session starts and topic of project decided. Some points such that project management, scalability, risk analysis, version control, aims, etc. were discussed primarily. The major aims set at beginning of this project was -

- To gain the technical knowledge and experience and practical "hands behind databases, their management systems, and other technologies
- A clear idea of working title; incorporating traditional accent but with some modern software engineering principles, practices, and standards
- Improving team skills under fully supervision (Project management including risk analysis and Soft-skills)
- To present a final result implementation, its analysis, and evaluation with proper documentation/project report.

Althought the aims were set primarily and broadly, we as a team collaborate on multiple issues and try as much as we can to be stick on aims along with healthy communication from supervisors. This document is neither a fully formal documentation nor a research based report, but it inherits many attributes of both in it. Here are some highlights of the project -

- ~ A single document having combination of documentation, project report, reference manual, tutorial, user manual for the output
- ~ Practical implementation of DataBase System used for Metro system
- ~ Parallel implementation of DataBase in many DataBases platforms, here we used MySQL workbench and LibreOffice Base as major two platforms using SQL as major query language
- ~ Incorporation of Software Engineering elements such as principles, practices, life-cycle, etc.

Most of the collected contents are properly referenced*. If any of work is not, please feel free to notify us at Github.com or via emails.

It is also planned to place this project on GitHub.com soon, where the version control of project is monitored properly and subject experts (actually everyone) will get authority to analyse, evaluate, and even collaborate on this project. This project will be licensed under [Creative Commons ShareAlike Non-distribution 4.0 International licence 2018](#). From software engineering practice we know Software prototype maintainance never ends as no software built is perfect, so the modifications to the first version/release of this project will be completely to github. Checkout the readme.md file at Github. Although we are saying output as a software, it might not be a fully functional SOFTWARE. We can say it as a working prototype of software which can further be enhanced in form of a software.

It was very memorable to work on such a project with such a team and such a mentor. No work is perfect, remember to write your views to us! Healthy critism, suggestions, feedback, and even your opinions are always welcome to enhance this project either at github.

Acknowledgements

With due respect, we express our deep sense of gratitude to our respected and learnt guide, coordinator, supervisor, instructor **Prof. Sweta Jain**, for her valuable help and guidance. We are thankful for the encouragement and motivation that she has given us in completing this project successfully. Her rigorous evaluation and constructive criticism was of great assistance. It is imperative for us to mention the fact that this project could not have been accomplished without the periodic suggestions and advice of our supervisor.

We are also grateful to our respected director **Dr. Narendra Raghuvanshi** for permitting us to utilize all the necessary facilities of the college.

Needless to mention is the additional help and support extended by our respected Head of the department- **Ms. Meenu Chawla**, in allowing us to use the departmental laboratories and other services on the period of time and also maintaining a healthy discipline in management, tutorials, eduventurisation, and other activities.

We are thankful to all the other faculties, Professors, Associate Professors, Assistant Professors, staff-members, teaching assistants, laboratory attendants, seniors, and our fellow branch mates of our work culture for their kind co-operation, periodic evaluation, help, and support. We thank all others who directly/indirectly involved in this project and make it a successful. We also thank all other teams for a maintaining an active communication aside of healthy competition and sportsmans spirit.

We would also like to express our deep appreciation towards our family members for making us such kind to do 'more' by providing the much needed support and encouragement from all aspects of life. At last, we recall our gratitude to the one eternal almighty: The God.

This page intentionally left blank

Contents

i.	Preface.....	5
ii.	Acknowledgements.....	7
iii.	Contents.....	9
iv.	Abstract.....	11
1.	Introduction and Quick review.....	13
1.1	Introduction.....	13
1.2	Aims and Scope of project.....	13
1.3	Is the product output of this project is a software/prototype?.....	14
1.4	CASE tools used.....	14
1.5	Corresponding Conceptual models for this project.....	14
1.6	Project management overview.....	14
1.7	Level-0 Project Synopsis.....	14
2.	Literature Survey.....	17
2.1	Databases.....	17
2.2	Structured Query Language.....	18
3.	Problem Statement.....	20
3.1	Problem Statement.....	20
3.2	Problem (Task) description and major tasks.....	20
3.2.1	Project aspects.....	20
3.2.2	Tasks.....	21
4.	Software and Hardware requirements.....	22
4.1	Minimum System/Software/Hardware requirements.....	22
4.2	Resources Usage.....	22
4.3	Functional requirements.....	22
4.4	Performance requirements.....	22
5.	Methodology used.....	23
5.1	Methodology.....	23
6.	Design framework.....	24
6.1	Entity relationship diagram.....	24
6.2	Entity relationship diagram (Preliminary).....	25
6.3	Sequence diagram.....	26
6.4	Architecture diagram.....	27
6.5	Data flow diagrams.....	28
6.6	Use-Case diagrams.....	29

7.	Implementation.....	30
7.1	Coding.....	30
7.2	Implementation Snapshots.....	30
8.	Testing.....	41
8.1	Procedure.....	41
8.2	Outcomes.....	41
9.	Conclusions and future scope.....	42
9.1	Conclusions.....	42
9.2	Future scope and improvements.....	42
	References and Bibliography.....	44

Appendices

i.	Appendix-A	List of tables, Web-pages, sample queries.....	45
		• List of tables.....	45
		• List of Web-pages.....	45
		• List of sample Queries.....	46
ii.	Appendix-B	SQL code of sample Queries.....	47
iii.	Appendix-C	Critism and Expert reviews.....	48

Abstract

A Metro System is one which helps us to maintain an organisational data regarding a metro system. Since it is very difficult to maintain a record of data items and this information manually or in file system, we are creating a database for the complete management of resources and tasks of metro system. A Metro System database is one which is used for operational ease in place of traditional file system or Excel worksheets for information storage and retrieval with customized user queries implemented in SQL and is a fully functionally required software prototype with proper user interface and some random software engineering standards. Organising database for data collection and retrieval helps us to maintain data easily. We design it with the help of Entity-Relationship diagram, UML diagrams, data flow diagrams, and using tables & proper User interfaces. It consists of several steps to be followed and many webpages are to be maintained simultaneously. It is a working prototype of a database schema which can further be enhanced. The usage of standard webpages in HTML, CSS, JavaScript, PHP helps us to take project online without much modifications.

For main functionalities, we have to maintain many records such as train numbers, train route, departure time, arrival time, train type, fare, managements, departments, customers, etc. When we want to travel we reserve seat (Additional feature for Metro System but it is not compulsory for all passengers to reserve seats for them, they can still check in at last moment in case of availability in first in first serve order), for this we have to fill the reservation form which includes entries such as train number, train name, departure time, arrival time, route, train type. So, we should gather all the information regarding all these.

Keywords: Metro System, Railway, Database, Prototype, Software Engineering principles

This page intentionally left blank

Chapter 1

Introduction & Quick review

Introduction

A Metro System database is one which is used for operational ease in place of traditional file system or Excel worksheets for information storage and retrieval with customized user queries implemented in SQL and is a fully functionally required software prototype with proper user interface and incorporating random software engineering standards.

Output of the Project :

- ✓ Working prototype of a sample metro system with user interfaces and some queries
- ✓ Documentation | Project report

Aims and Scope of project

Aims :

- To build a database used for practical implementation (For Metro System)
- To learn the concepts of databases and their management of systems
- To understand software construction process and its engineering
- To learn various Software Engineering principles and standards
- To enhance team work and individual responsibilities

Scope :

- Software can be used theoretically (Modifications for practical usage)
- Further modification and enhancements possible in maintenance stage
- Research oriented aspect of project may lead to better technology implementations

Is the product output of this project is a software?

Sort of Yes. The work product output of this project is a 'software'; working properly in low scalability and is properly tested on basic functionalities, which is to be incorporated in that. An engineering approach is used for building that software that includes first define problem, analyze it, design it, implement it, test it, and regularly modify it. Technically the product output of this project is a demo prototype.

Open Source CASE tools used

Upper CASE tools used:

- LibreOffice Writer (For documentation)
- LibreOffice Draw (For designing diagrams)

Lower CASE tools used:

- MySQL Workbench/LibreOffice Base (For SQL implementation)

Corresponding Conceptual models for this project

- Relational Model (See Appendix-A for All relations and attributes)
- Enhanced Entity-Relational Model (See Entity-Relationship diagram)
- Object Oriented UML modelling
 - Use case diagrams
 - Sequence diagrams, etc.

See more at Chapter 6.

Project management overview

- Embedded Project Planning
- Project Scheduling (Internal in team)
- Risk management (Overview and Intuitive)
- Project monitoring and Controlling
- W₅HH Principles (See preliminary Synopsis)
- Basic Project estimation
 - Cost estimation
 - Duration estimation
 - Effort estimation

Level-0 Project Synopsis: W₅HH Principles

Why is the project being developed ?

- To learn and understand the conceptual and fundamental knowledge in Database Management System

- To contribute a little to open-source software community
- To have a practical hands on experience on a real life database system built and maintained by us
- To enhance team skills of all the members of team
- To learn modern technologies effectively with software engineering approaches

What will be done ?

- Analysis and design of metro database system
- Implementation of design into various platforms
- Evaluation of all the implementations
- Building a final software with proper user interface
- Prepare documentation, project report, and other documents

When will it be done ?

- The project is supposed to be completed by April 2018
- First release by April 15, 2018
- Maintenance of that will be referred then to GitHub community

Where are the organisations located ?

- This project is a single organization, single team project located at MANIT, Bhopal

Who is responsible for the desired functionality ?

- Each member has assigned his/her own responsibility regarding projects in sections and of risk analysis, they have to follow their rules
- Team members are atomic elements of project and are solely responsible for their work
- Timely supervision is responsibility of the instructor.

How will the job be done technical and management side ?

Technical issues :

- Database configuration and building is to be done on Oracle Database 12c, MySQL Workbench 6.3, etc.
- UML diagrams, Entity-Relationship diagrams are to be prepared on LibreOffice Draw.
- Documentation is to be done on LibreOffice Writer and a LaTeX edition will also be available (prepared on TeXLive)
- For user interface, either Java or Python or web-development languages (HTML, CSS, JS, PHP) will be used with SQL as major query language

Management issues : Software project management guidelines are to be followed on the following-

- Risk analysis
- Cost estimation
- Time scheduling
- Project monitoring and controlling

How much of each resource is needed ?

- Human resource of 3 team members
- Monetary resource of atmost 1,000 INR.
- Hardware resource of a computer with required software installed.
- Software resources used.

Chapter 2

Literature Survey

Databases

Parameter	Description
Data Access	Modern relational database management system programs use a programming language known as structured query language to access, update, and delete data within its tables. These programs, including Microsoft's SQL Server and the open-source MySQL systems, allow outside programs to access its data via SQL queries. For instance, a web site can display product data, including photos, prices and descriptions, when the web server software connects to the data contained in the relational database management system.
Data Relationships	One of the most important aspects of an relational database management system program is how it allows different data tables to relate to each other. When a database contains a table with employee data on its sales staff and another with data on its product sales, the relational database management system can manage the relationship between the two tables. This relationship can help management determine which salesperson has the highest sales totals and which product that salesperson is selling the most.
Data Updates	A fully-functional relational database management system allows users to enter new information, update current records and delete outdated data. As an example, when a salesperson sells 1,000 units, that person will enter the transaction information into the relational database management system. The data can include the salesperson's name, the customer information, the product sold and the quantity sold. The relational database management system enters a new record in the customer table, updates the salesperson's record and subtracts 1,000 units from the inventory record.
Data Searches	The relational database management system also ensures that a company can build and maintain its data over the system's lifetime. The various tables in the relational database management system

	allow users to search through the system using any available criteria. Customers can search a product table by name, brand, price, color or any other feature. The system stores data in a predictable, sequential format, enabling users to look up previous records with relative ease.
--	---

Structured Query Language (SQL)

SQL (Structured Query Language) is a standardized programming language used for managing relational databases and performing various operations on the data in them. Initially created in the 1970s, SQL is regularly used by database administrators, as well as by developers writing data integration scripts and data analysts looking to set up and run analytical queries.

The uses of SQL include modifying database table and index structures; adding, updating and deleting rows of data; and retrieving subsets of information from within a database for transaction processing and analytics applications. Queries and other SQL operations take the form of commands written as statements -- commonly used SQL statements include select, add, insert, update, delete, create, alter and truncate.

SQL became the de facto standard programming language for relational databases after they emerged in the late 1970s and early 1980s. Also known as SQL databases, relational systems comprise a set of tables containing data in rows and columns. Each column in a table corresponds to a category of data -- for example, customer name or address -- while each row contains a data value for the intersecting column.

SQL standard and proprietary extensions

An official SQL standard was adopted by the American National Standards Institute (ANSI) in 1986 and then by the International Organization for Standardization, known as ISO in 1987. More than a half-dozen joint updates to the standard have been released by the two standards development bodies since then; as of this writing, recent version is SQL:2011, approved that year.

Both proprietary and open source RDBMS built around SQL are available for use by organizations. They include MS SQL Server, Oracle Database, IBM DB2, SAP HANA, SAP Adaptive Server, MySQL (now owned by Oracle) and PostgreSQL. However, many of these database products support SQL with proprietary extensions to the standard language for procedural programming and other functions. For example, Microsoft offers a set of extensions called Transact-SQL (T-SQL), while Oracle's extended version of the standard is

PL/SQL. As a result, the different variants of SQL offered by vendors aren't fully compatible with one another.

SQL commands and syntax

SQL commands are divided into several different types, among them data manipulation language (DML) and data definition language (DDL) statements, transaction controls and security measures. The DML vocabulary is used to retrieve and manipulate data, while DDL statements are for defining and modifying database structures. The transaction controls help manage transaction processing, ensuring that transactions are either completed or rolled back if errors or problems occur. The security statements are used to control database access as well as to create user roles and permissions.

Chapter 3

Problem Statement

Problem Statement

The major task is to prepare a fully functional database system prototype (with user interfaces) for a metro network (quite similar to Indian Metro Ltd./Indian Railways) which will be implemented as a working open source prototype/project (hosted at GitHub) prepared with the help of standard software engineering techniques, software project management guidelines, proper documentation & report preparation, etc. using various platforms [LibreOffice Base and MySQL Workbench for database preparation] using CASE tools such as LibreOffice Writer for documentation and diagram designing. User Interfaces are built in form of webpages using HTML, CSS, JavaScript, PHP for proper handling of user queries and data entrance.

Problem (Task) description and major tasks

Project aspects : (P-Poor, G-Good, E-Excellent)

➤ Reliable backup and recovery	G
➤ Distributed functions	G
➤ Performance	G
➤ Heavily used configurations	P
➤ Online data entry	P
➤ Operational ease	G
➤ Updates	G
➤ User Interface	E
➤ Complex processing	G
➤ Reusability of Software	E
➤ Installation ease	G
➤ Multiple sites	P
➤ Facilitated Changes	E

We've explored various aspects of a practical system (Indian Metro/Indian Railways) at overview level and seems basic attributes (basis records) to be included in the database and are listed as-

- Train No.
- Station No.
- Reservation ID
- Passenger ID
- Departure Time

- Train Type
- Fare and Distances
- Source – Destination, etc.

Most of the above attributes are taken into account for project problem and are properly incorporated in relational model design of project and in Entity-Relationship diagram. See E-R diagram.

Tasks :

- Planning and Analysis of commercially available softwares on metro system
- Database implementations
- Standard testing of all implementations
- Performance analysis of all implementations (to evaluate all implementations)
- Providing a user interface for end & front users, administrators, etc.
- Make a fully functional prototype.

Chapter 4

Software and Hardware requirements

Minimum System/Software/Hardware requirements :

- Windows 7/Linux 3/Mac OS/Chromium OS
- Memory requirements: 64 MB (RAM).
 - Recommended 128 MB.
- Secondary memory requirements (Hard disk): 10 GB (ROM).
 - Recommended 256/512GB
- Oracle MySQL Workbench/LibreOffice Base
 - Recommended MySQL Workbench/Command Line Client
- Writer and diagram designing software such as MS Visio or LibreOffice Draw.
- Clock Speed: 866 Mhz
- Virtual Memory: 32 bits
- Cache Memory: 512KB, etc.

Resources Usage :

- Operating System: Windows 10/Ubuntu 17.10
- Memory: 8 GB (RAM)
- Secondary memory: 1 TB (ROM)
- Oracle MySQL Workbench 6.3 CE (Windows)/ LibreOffice Base 5 (Ubuntu 17.10/Linux 4)
- Microsoft Word 16/ LibreOffice Writer 5.4 (Linux)

Functional requirements :

- Customer/Passenger/Employee data storage and retrieval
- Proper User Interface such as input forms
- Customized user queries for complete system including proper error handling

Performance requirements :

- Shouldn't wear out in passage of time
- Shouldn't lag in responding to user enquires
- Shouldn't crash abruptly
- Must be in accordance to data consistency (Proper resemblance of foreign keys)

Chapter 5

Methodology used

Methodology

The methodology adopted in the project is simple and is easy to implement. **Functions and Object oriented methodology** constitute a major portion of methodology. Project is divided into individual tasks. Tasks are then taken as initial problem and solved individually and have to integrate them at last. Objectives are set first and then we tried to fulfill all objectives asserted.

We've first implemented preliminary Entity-Relationship diagram in 2 platforms namely- MySQL Workbench (Windows 10) and LibreOffice Base (Ubuntu 17.10). We then intuitively analyzed which platform is more suitable for this particular schema and required functionalities. After that we enhanced that schema to further more relations, entities, relationships, forms, queries, etc.

The Project development model we used in project development is closely related to hybrid association of waterfall phases, Rapid application development (Rapid delivery of project), Agile methodology (Simplicity and Swift evaluation), Some Extreme Programming practices, and 4th generation tool usage (Usage of CASE tool: LibreOffice Base/Draw). The involvement of multiple model elements in single project leads to higher level of customization, better management of project, better requirements adaptability, and improved rigidity of overall management procedure and development life cycle.

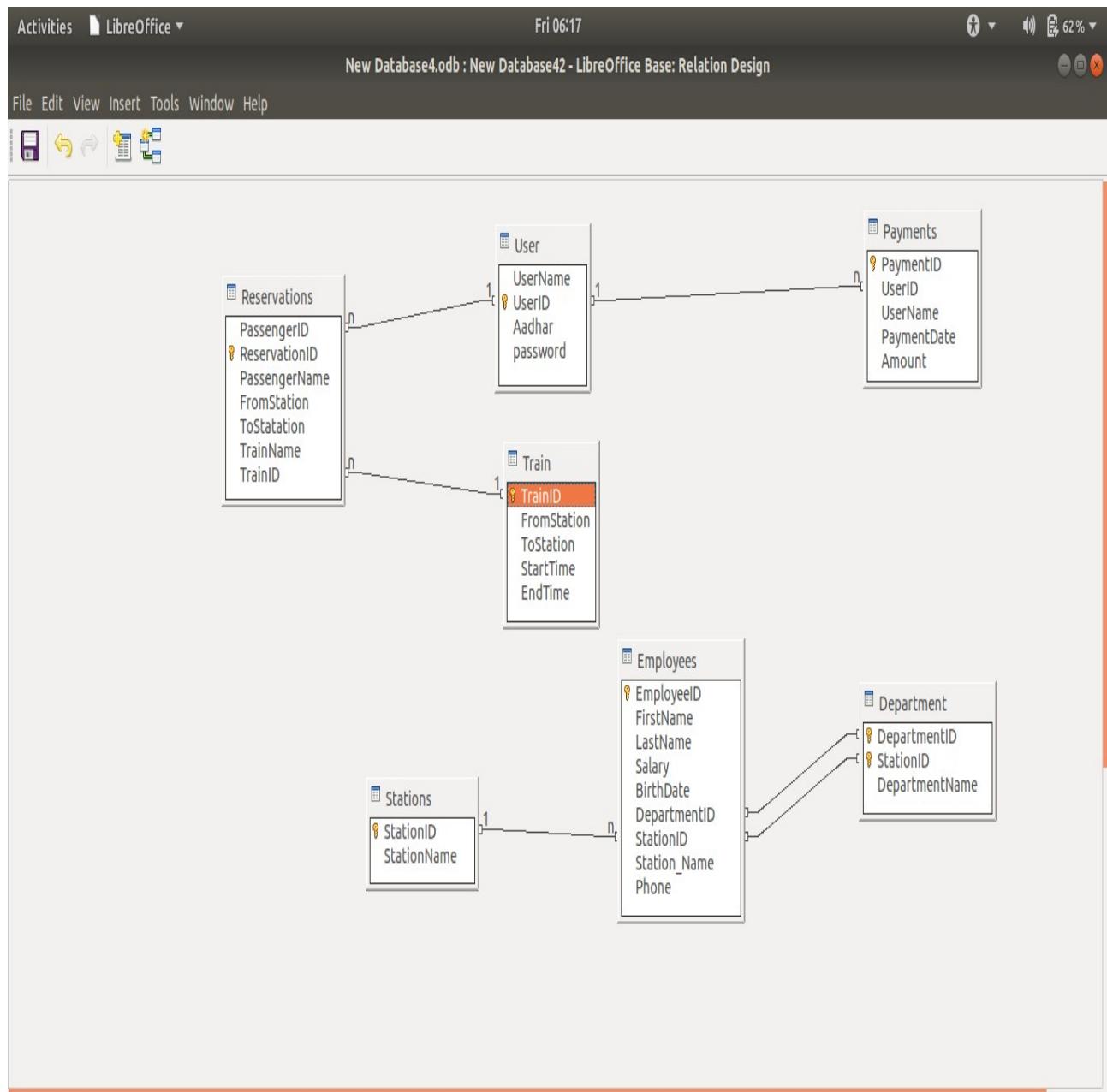
We've often used many Software engineering principles in the project to make it well organised, formal, and more productive. Use of Open-Source tools helps us to contribute Free software foundation and Open-source community. We've modelled the project structure into various diagrams such as Entity-Relationship diagrams, Use-Case diagrams, Sequence diagrams, Data flow diagrams. An Architecture diagram is also designed for an overview of database procedure inside the system and the interconnection of network between Client and Server.

Chapter 6

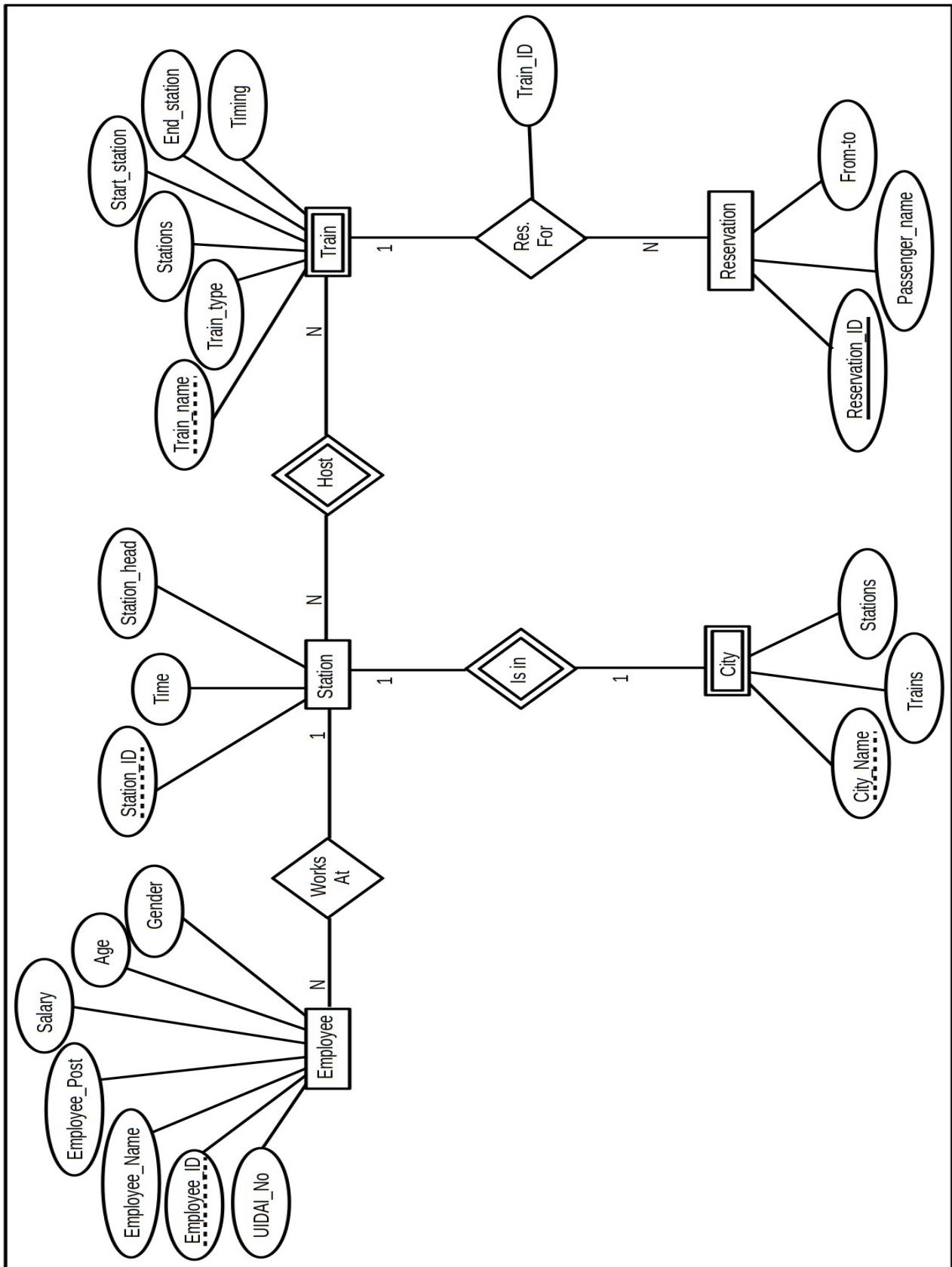
Design framework

Entity-Relationship diagram:

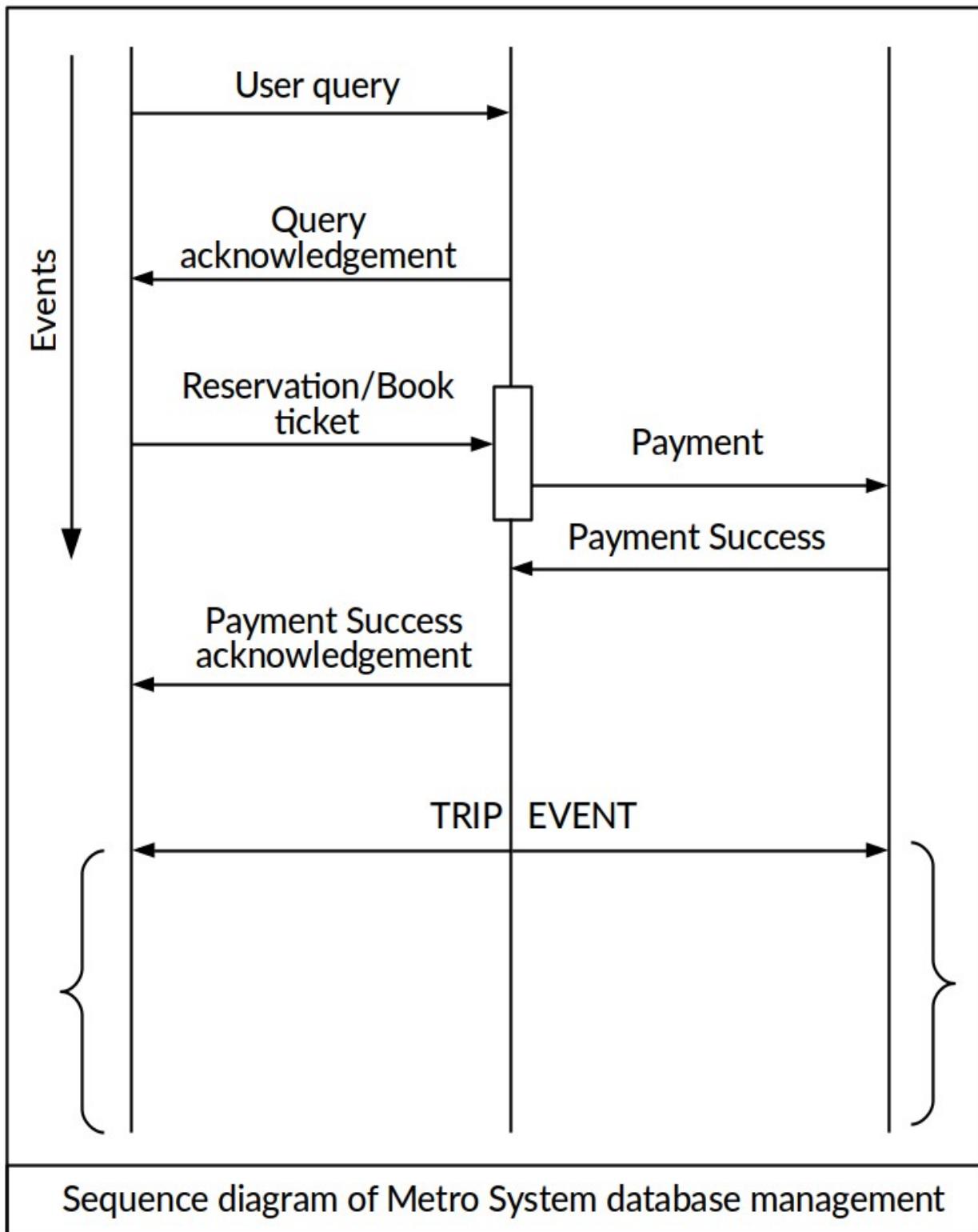
***Screenshot size is modified for better visibility*



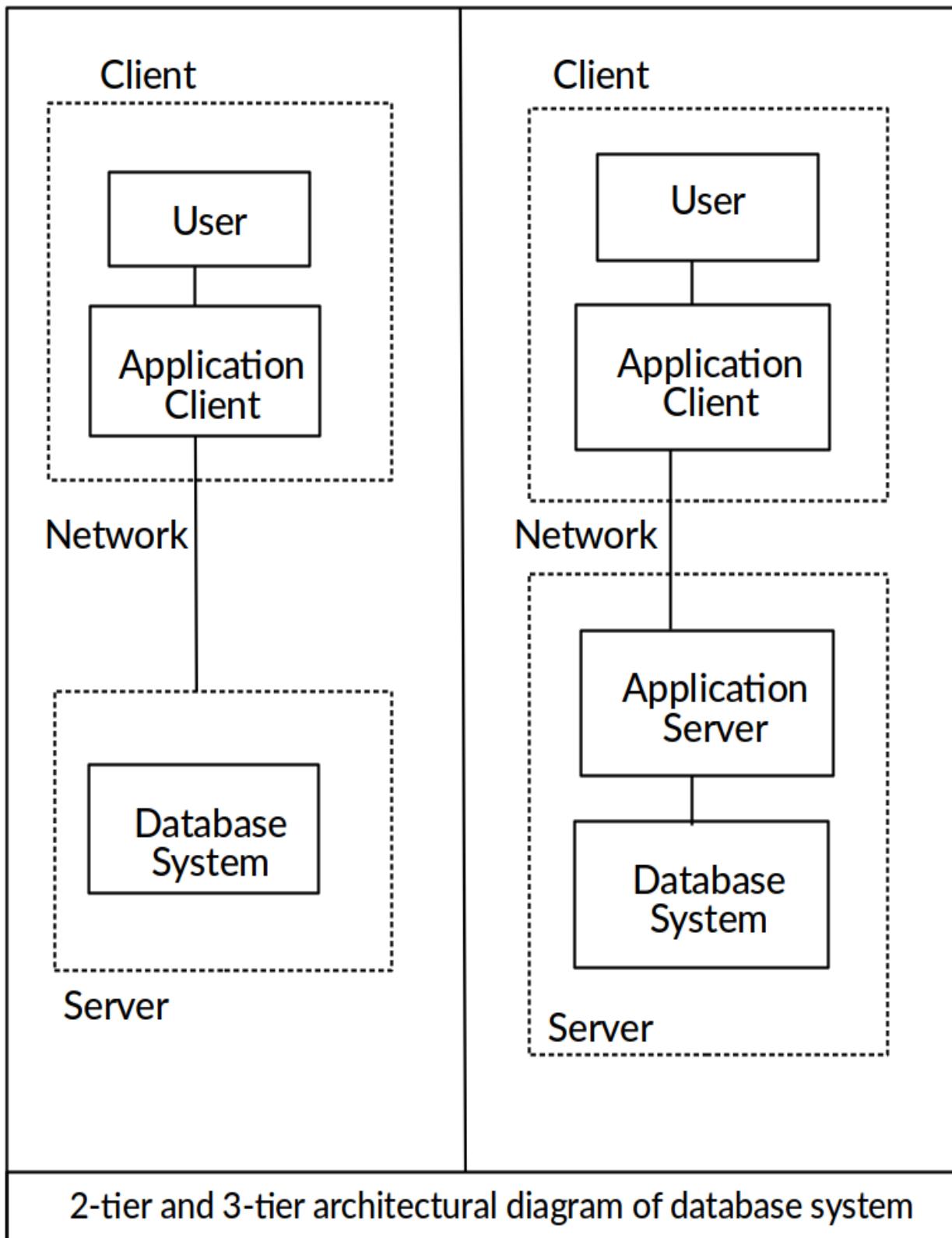
Entity-Relationship diagram (Preliminary) :



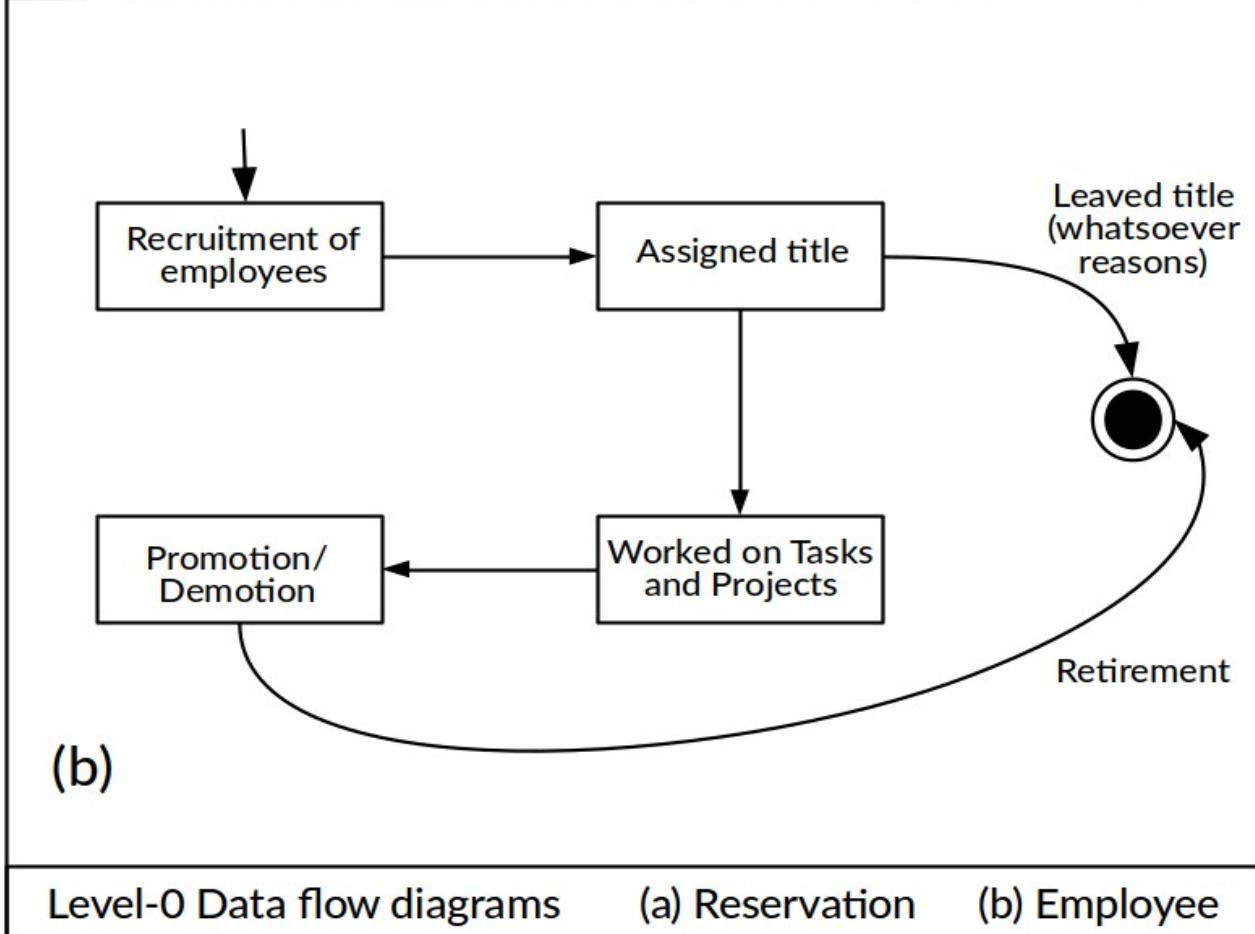
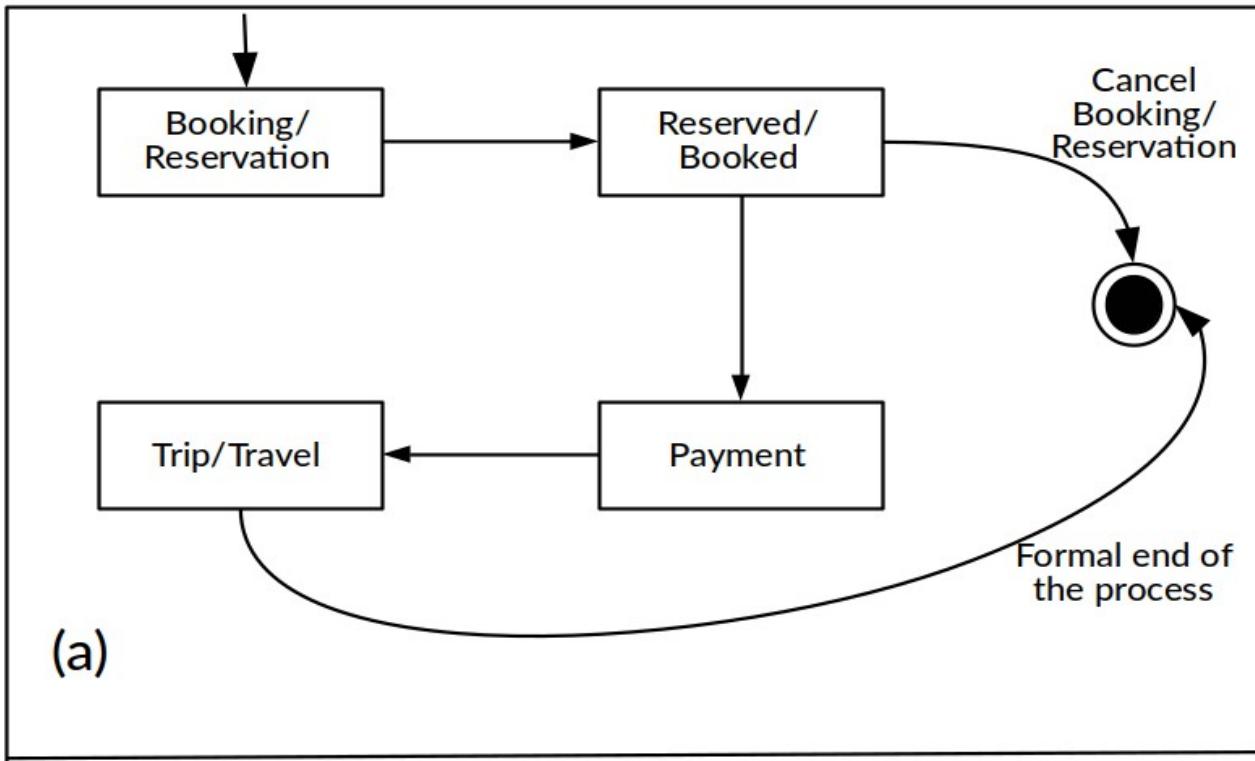
Sequence diagram :



Database architectural diagram :



Data flow diagrams :

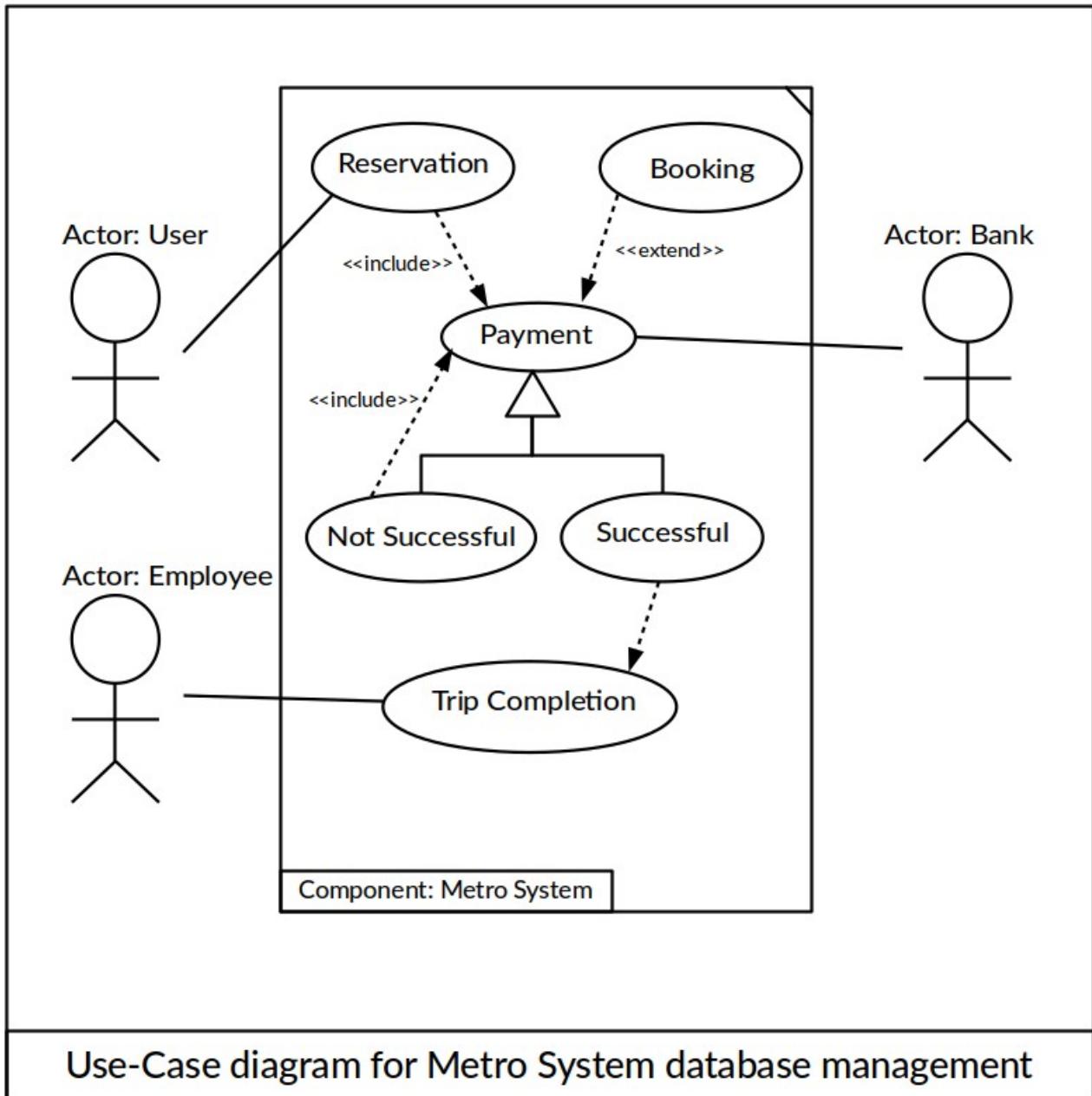


Level-0 Data flow diagrams

(a) Reservation

(b) Employee

Use-Case diagrams :



Chapter 7

Implementation

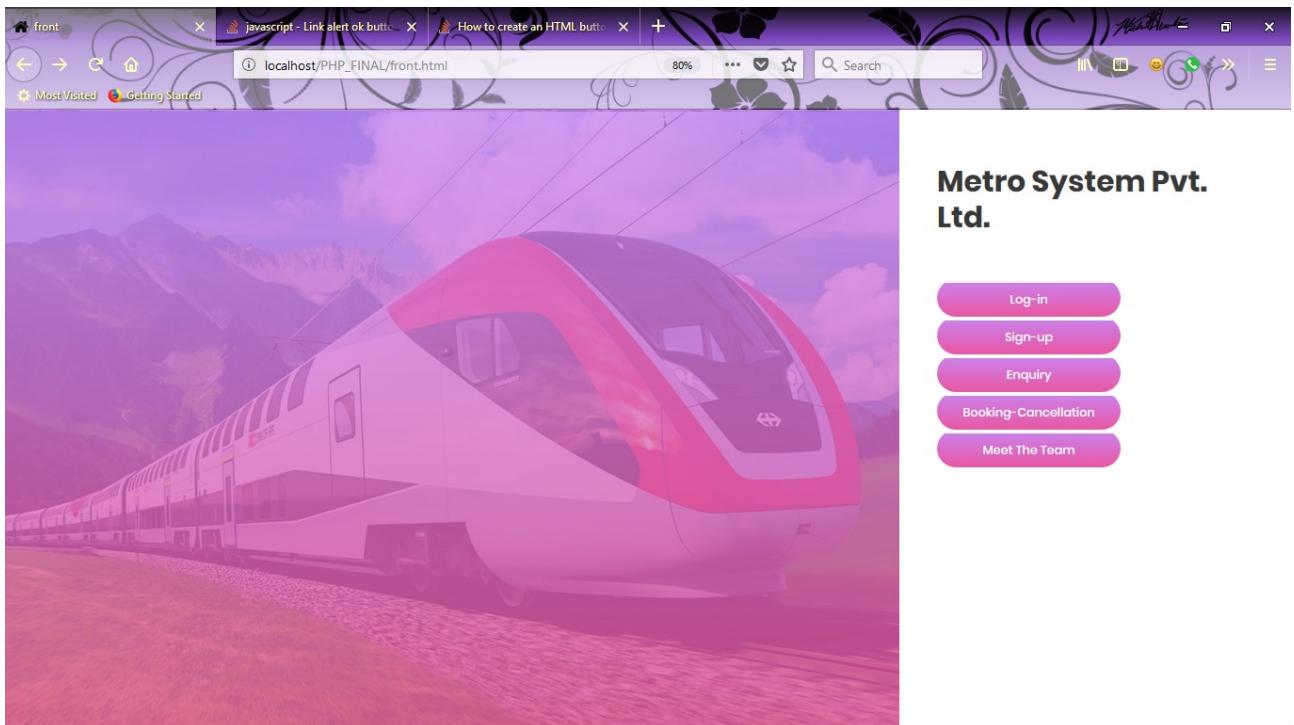
Coding

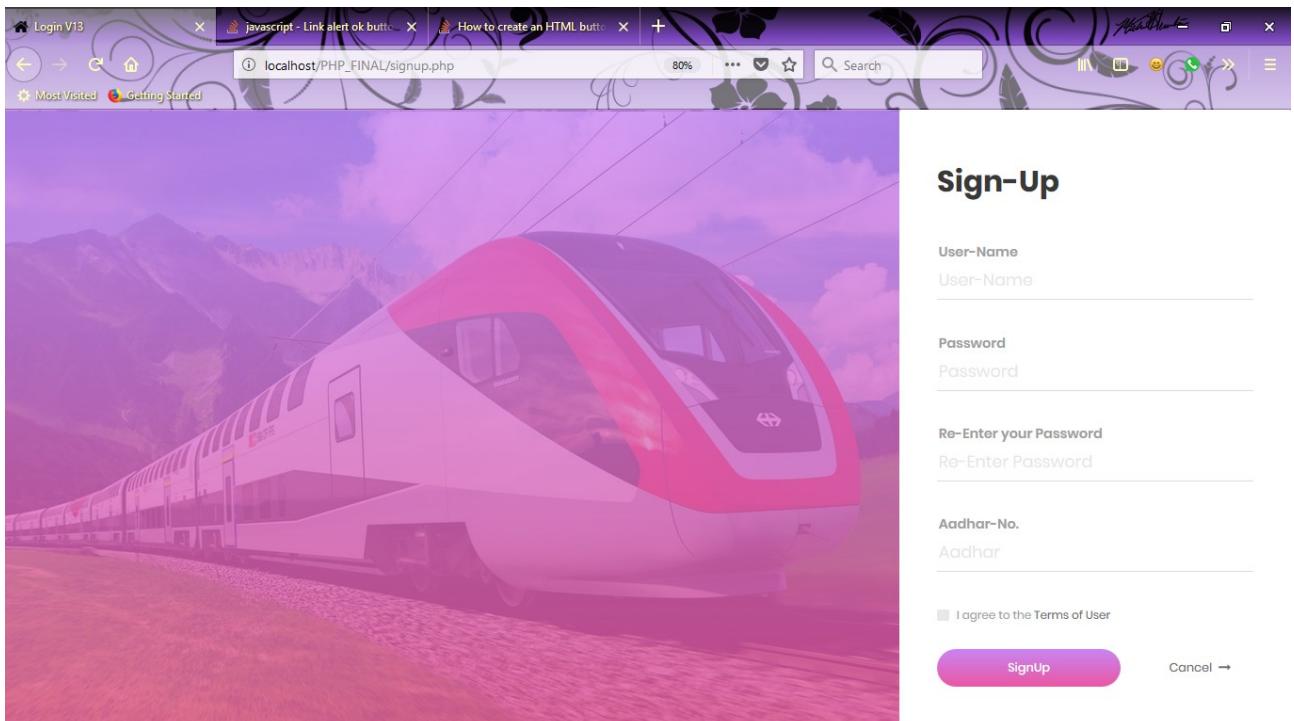
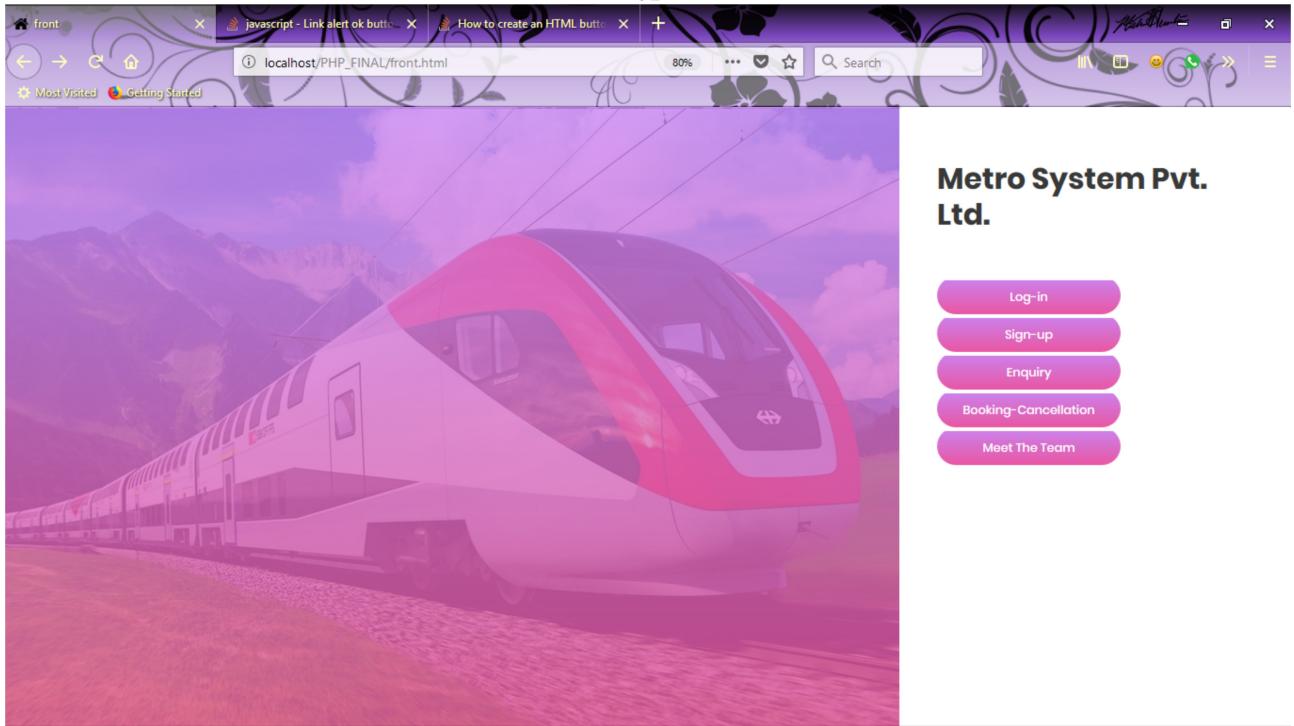
Coding is mainly done in Structured Query Language (SQL) for MySQL Command Line Client for customised queries. The Codes of HTML, CSS, JS, and PHP are formatted, understandable, customizable, and modifiable. We can combine other functionalities using these codes since there we kept a margin for reusability. Check out Implementational snapshots for a view of codes.

Also Check out Appendix-B for runnable codes of sample queries in SQL.

Implementational Snapshots

Front-end Webpages







Sign-Up

User-Name

Password

Re-Enter your Password

Aadhar-No.

I agree to the Terms of User

SignUp Cancel →



Log-in

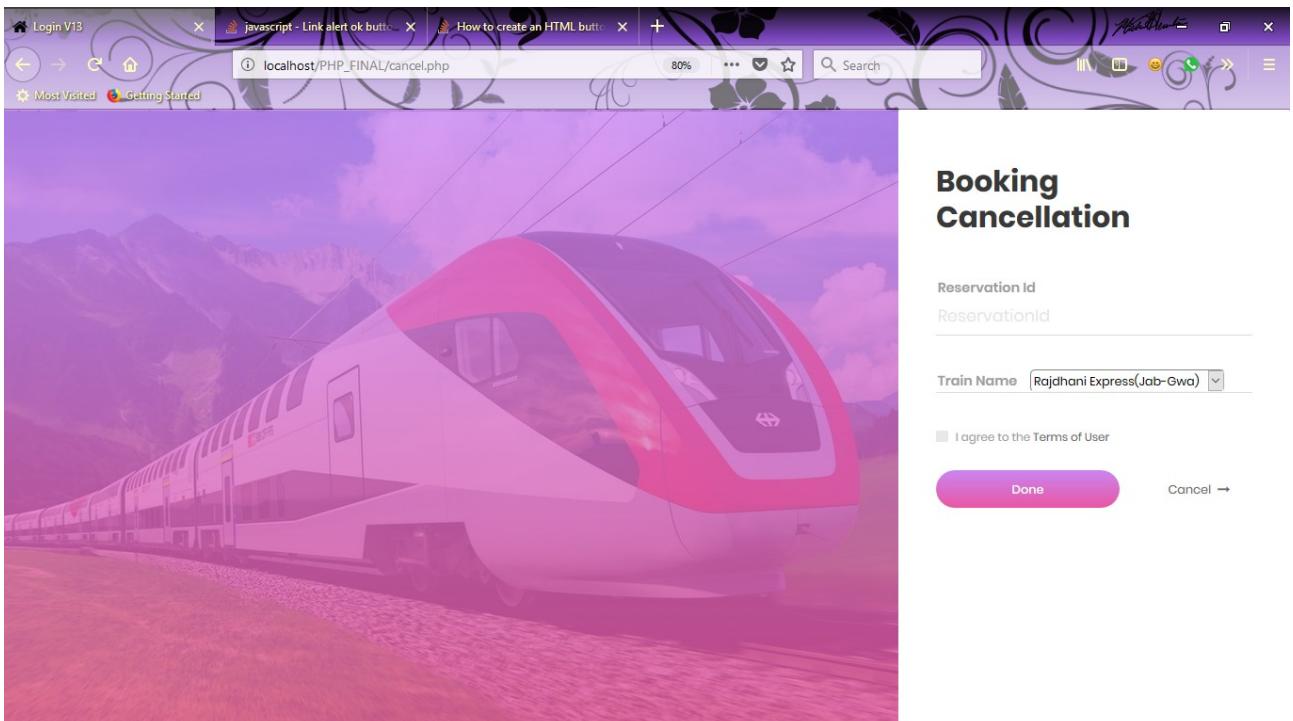
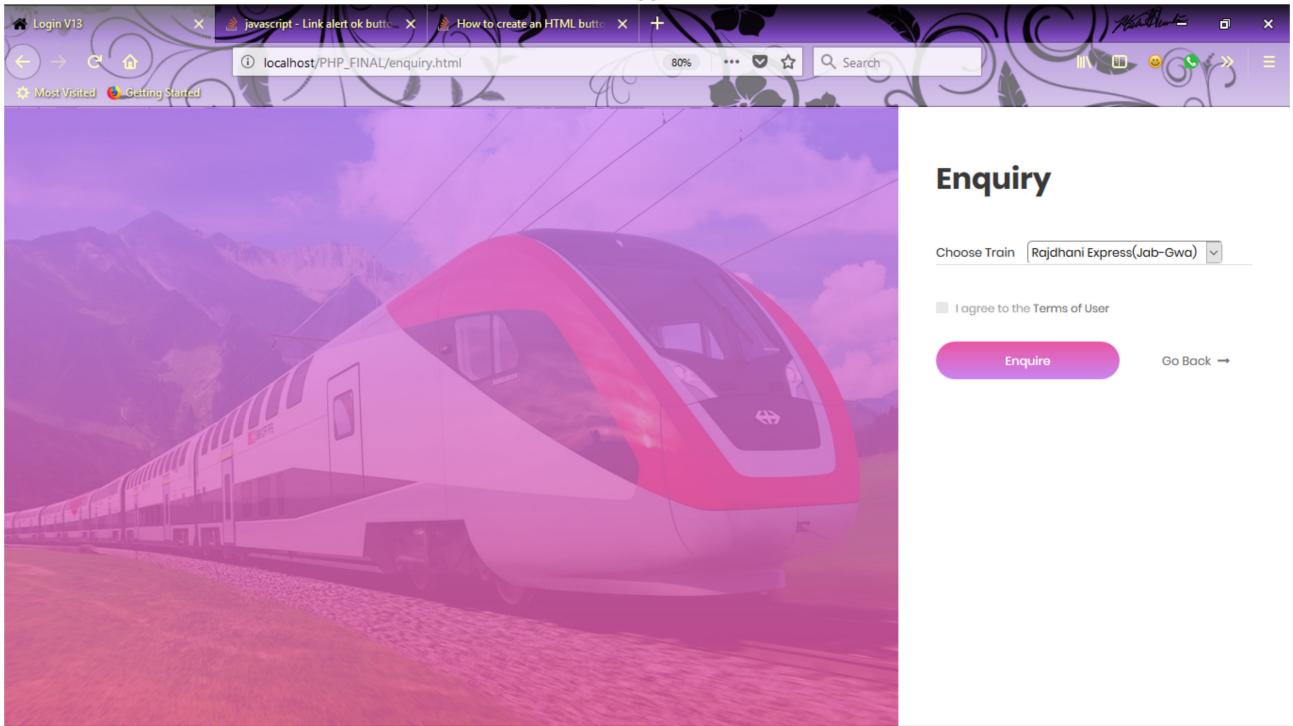
User-Name

User-Id

Password

I agree to the Terms of User

Login Sign-Up Cancel →



About-Us

Ankit Chouhan
Database Administrator, Front-End and Back-End Developer

Jishan Shaikh
Database Administrator and Software Developer

Swapnil Dubey
Web Designer and Database Designer

Sanjeet Gautam
Web Designer and Front-End Developer

[Go Back](#)

Registration

From

To

Train

Enter your Name
Name of Ticket Booker...

I agree to the Terms of User

[Confirm](#) [Enquiry](#) [Cancel →](#)

Back-end Codes

```
E:\WAMP\wamp64\www\PHP_FINAL\front.html - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
login.php front.html insert into Train(TrainId, TrainName, FromStation, ToStation, Date, Time, Duration, Status, Type, Category, Price, Seats, Capacity, Description, ImagePath) values(?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?) Booking.php cancel.php enquiry.php enquiry.html signup.html SQL CODES signup.php

31 <head>
32 <body style="background-color: #999999;">
33
34     <div class="limiter">
35         <div class="container-login100">
36             <div class="login100-more" style="background-image: url('images/bg-01.png');"></div>
37
38         <div class="wrap-login100 p-l-50 p-r-50 p-t-72 p-b-50">
39
40             <span class="login100-form-title p-b-59">
41                 Metro System Pvt. Ltd.|<br/>
42             </span>
43
44             <div class="container-login100-form-btn">
45                 <div class="wrap-login100-form-btn">
46                     <div class="login100-form-bgbtn"></div>
47                     <button class="login100-form-btn" onclick="window.location='login.php'" style="border-top:6px solid white;">
48                         Log-in
49                     </button>
50                 </div>
51
52             <div class="container-login100-form-btn">
53                 <div class="wrap-login100-form-btn">
54                     <div class="login100-form-bgbtn"></div>
55                     <button class="login100-form-btn" onclick="window.location='signup.php'" style="border-top:6px solid white;">
56                         Sign-up
57                     </button>
58                 </div>
59
60             <div class="container-login100-form-btn">
61                 <div class="wrap-login100-form-btn">
62                     <div class="login100-form-bgbtn"></div>
63                     <button class="login100-form-btn" onclick="window.location='enquiry.html'" style="border-top:6px solid white;">
64                         Enquiry
65                     </button>
66                 </div>
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
99
```

E:\WAMP\wamp64\www\PHP_FINAL\cancel.php - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

login.php x front.html x insert into Train[TrainId, TrainName, FromStation, ToStation, SeatsRemaining] x Booking.php x cancel.php x enquiry.php x enquiry.html x signup.html x SQL CODES x nup.php x

```
42 }
43
44
45 if(isset($_POST['submit'])){
46     if( isset($_POST['ReservationId'])&& isset($_POST['Train'])){
47         $ReservationId=$_POST['ReservationId'];
48         $train=$_POST['Train'];
49         $sql = "DELETE FROM Reservations WHERE ReservationId='$ReservationId'";
50
51         if (mysqli_query($conn, $sql))
52         {
53             >>
54             <script type=text/javascript>alert("<?php echo \"Your Booking Has Been Cancelled..\"?>")</script>
55             <?php
56             $sql="select SeatsRemaining from train where TrainName='".$train."'";
57             $result = mysqli_query($conn, $sql);
58
59             if (mysqli_num_rows($result) > 0)
60             {
61                 while($row = mysqli_fetch_assoc($result))
62                 {
63                     $NewSeats = $row["SeatsRemaining"];
64                 }
65
66                 $NewSeats = $NewSeats+1;
67
68                 $sql = "UPDATE Train SET SeatsRemaining='".$NewSeats' WHERE TrainName='".$train."'";
69                 $set = mysqli_query($conn, $sql);
70             }
71             else
72             {
73                 >>
74                 <script type=text/javascript>alert("<?php echo \"Error While Booking Cancellation\"?>")</script>
75                 <?php
76                 die();
77             }
78         }
79     }
80 }
```

```
E:\WAMP\wamp64\www\PHP_FINAL\enquiry.html - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
login.php x front.html x insert into Train(TrainId, TrainName, FromStation, ToStation, Date, Time, Duration, Status) values(?, ?, ?, ?, ?, ?, ?, ?) Booking.php x cancel.php x enquiry.php x enquiry.html x signup.html x SQL CODES x nup.php x
34     <div class="limiter">
35         <div class="container-login100">
36             <div class="login100-more" style="background-image: url('images/bg-01.png');"></div>
37
38             <div class="wrap-login100 p-1-50 p-r-50 p-t-72 p-b-50">
39                 <form class="login100-form validate-form" action="enquiry.php" method="POST">
40                     <span class="login100-form-title p-b-59">
41                         Enquiry
42                     </span>
43
44                     <div class="wrap-input100 validate-input">
45                         Choose Train
46                         <select name="Train" style="margin-left: 13px; border-radius: 6px;">
47                             <option>Rajdhani Express(Jab-Gwa)</option>
48                             <option>Avantika Express(Ind-Jab)</option>
49                             <option>Mayanagri Express(Ind-Gwa)</option>
50                             <option>Ajanta Express(Bpl-Jab)</option>
51                             <option>Ammavati Express(Bpl-Gwa)</option>
52                             <option>Holkar Express(Ind-Ujj)</option>
53                             <option>Bhoj Express(Ind-Dws)</option>
54                             <option>Ujjaini Express(Dws-Ujj)</option>
55                             <option>Kshipra Express(Ujj-Jab)</option>
56                             <option>Narmada Express(Ujj-Gwa)</option>
57                             <option>Godavari Express(Dws-Jab)</option>
58                             <option>Gwaliori Express(Gwa-Dws)</option>
59                             <option>Malwa Express(Bpl-Ujj)</option>
60                             <option>Gatimaan Express(Dws-Bpl)</option>
61                             <option>Intercity Express(Bpl-Ind)</option>
62                             <option>Rajdhani Express(Bpl-Gwa)</option>
63                             <option>Mayanagri Express(Gwa-Jab)</option>
64                             <option>Avantika Express(Jab-Gwa)</option>
65                         </select>
66                     </div>
67
68
69                     <div class="flex-m w-full p-b-33">
70                         <div class="contact100-form-checkbox">
71                             <input class="input-checkbox100" id="ckb1" type="checkbox" name="remember_me">
```

E:\WAMP\wamp64\www\PHP_FINAL\signup.html - Sublime Text (UNREGISTERED)

```

29     <link rel="stylesheet" type="text/css" href="css/util.css">
30     <link rel="stylesheet" type="text/css" href="css/main.css">
31 <!-- End of Header -->
32 </head>
33 <body style="background-color: #999999;">
34
35
36     <form class="login100-form validate-form" action="signup.php" method="post">
37         <div class="limiter">
38             <div class="container-login100">
39                 <div class="login100-more" style="background-image: url('images/bg-01.png');"></div>
40
41             <div class="wrap-login100 p-l-50 p-r-50 p-t-72 p-b-50">
42
43                 <span class="login100-form-title p-b-59">
44                     Sign-Up
45                 </span>
46
47                 <div class="wrap-input100 validate-input" data-validate="Starting Place">
48                     <span class="label-input100">User-Name</span>
49                     <input class="input100" type="text" name="UserName" placeholder="User-Name">
50                     <span class="focus-input100"></span>
51                 </div>
52
53                 <div class="wrap-input100 validate-input" data-validate="Please Enter No of passengers">
54                     <span class="label-input100">Password</span>
55                     <input class="input100" type="password" name="Password" placeholder="Password">
56                     <span class="focus-input100"></span>
57                 </div>
58
59                 <div class="wrap-input100 validate-input" >
60                     <span class="label-input100">Re-Enter your Password</span>
61                     <input class="input100" type="password" name="RePassword" placeholder="Re-Enter Password">
62                     <span class="focus-input100"></span>
63                 </div>
64
65                 <div class="wrap-input100 validate-input">
66                     <span class="label-input100">Aadhar</span>
67                     <input class="input100" type="text" name="Aadhar" placeholder="Aadhar">
68                     <span class="focus-input100"></span>
69                 </div>
70
71             </div>
72         </div>
73     </form>

```

Line 52, Column 1 Tab Size: 4 HTML

E:\WAMP\wamp64\www\PHP_FINAL\signup.php - Sublime Text (UNREGISTERED)

```

38     $password="";
39     $db="Metro";
40
41 $conn = mysqli_connect($host, $user, $password, $db);
42 if(!$conn){
43     die("Connection Failed: ". mysqli_connect_error());
44 }
45
46 if(isset($_POST['submit'])){
47     if(isset($_POST['UserName']) && isset($_POST['Password']) && isset($_POST['RePassword']) && isset($_POST['Aadhar'])){
48         $UserName=$_POST['UserName'];
49         $Password=$_POST['Password'];
50         $RePassword=$_POST['RePassword'];
51         $Aadhar=$_POST['Aadhar'];
52         $UserId=rand(1, 999);
53         if($Password!=$RePassword){
54             die();
55         }
56         else{
57             $sql = "INSERT INTO User(UserID, UserName, Aadhar, Password)
58                   VALUES ('{$UserId}', '{$UserName}', '{$Aadhar}', '{$Password}')";
59
60             if(mysqli_query($conn, $sql)){
61                 <script type=text/javascript>alert("Congratulations! <?php echo $UserName; ?>. You successfully registered. Your User ID is <?php echo
62                         $UserId; ?>.")
63                 </script>
64                 <?php
65             }
66             else{
67                 echo "Error while Registering....";
68             }
69         }
70     }
71 }
72
73 >

```

Line 132, Column 31 Tab Size: 4 PHP

MySQL Command line Client

```
MySQL 5.5 Command Line Client
+-----+-----+-----+-----+-----+-----+
|     138 |      NULL |    Jishan |   Indore |   Bhopal |      NULL | Intercity Express(Bpl-Ind) |
|     754 |      863 |    Jishan |   Bhopal |   Indore |      NULL | Intercity Express(Bpl-Ind) |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.01 sec)

mysql> select * from train;
+-----+-----+-----+-----+-----+-----+-----+
| TrainId | TrainName | FromStation | ToStation | Starttime | Endtime | SeatsRemaining |
+-----+-----+-----+-----+-----+-----+-----+
| 12345 | Rajdhani Express(Jab-Gwa) | Jab/Gwa | Gwa/Jab | 18:50/04:30 | 06:45/14:55 | 100 |
| 12346 | Avantika Express(Ind-Jab) | Ind/Jab | Jab/Ind | 11:50/06:30 | 06:45/12:55 | 100 |
| 12347 | Mayanagri Express(Ind-Gwa) | Ind/Gwa | Gwa/Ind | 18:50/04:30 | 06:45/14:55 | 100 |
| 12348 | Ajanta Express(Jab-Bpl) | Jab/Bpl | Bpl/Jab | 18:50/04:30 | 06:45/14:55 | 100 |
| 12349 | Amravati Express(Bpl-Gwa) | Bpl/Gwa | Gwa/Bpl | 18:50/04:30 | 06:45/14:55 | 100 |
| 12350 | Holkar Express(Ind-Ujj) | Ind/Ujj | Ujj/Ind | 18:50/04:30 | 06:45/14:55 | 100 |
| 12351 | Bhoj Express(Ind-Dws) | Ind/Dws | Dws/Ind | 18:50/04:30 | 06:45/14:55 | 100 |
| 12352 | Ujjaini Express(Dws-Ujj) | Dws/Ujj | Ujj/Dws | 18:50/04:30 | 06:45/14:55 | 100 |
| 12353 | Kshipra Express(Ujj-Jab) | Ujj/Jab | Jab/Ujj | 18:50/04:30 | 06:45/14:55 | 100 |
| 12354 | Narmada Express(Ujj-Gwa) | Ujj/Gwa | Gwa/Ujj | 18:50/04:30 | 06:45/14:55 | 100 |
| 12355 | Godavari Express(Jab-Dws) | Jab/Dws | Dws/Jab | 17:50/03:30 | 06:45/14:55 | 100 |
| 12356 | Gwaliori Express(Dws-Gwa) | Dws/Gwa | Gwa/Dws | 18:50/04:30 | 06:45/14:55 | 100 |
| 12357 | Malwa Express(Bpl-Ujj) | Bpl/Ujj | Ujj/Bpl | 18:50/04:30 | 06:45/14:55 | 100 |
| 12359 | Intercity Express(Bpl-Ind) | Bpl/Ind | Ind/Bpl | 18:50/04:30 | 06:45/14:55 | 98 |
| 12360 | Rajdhani Express(Bpl-Gwa) | Bpl/Gwa | Gwa/Bpl | 18:50/04:30 | 06:45/14:55 | 100 |
| 12361 | Mayanagri Express(Jab-Gwa) | Jab/Gwa | Gwa/Jab | 18:50/04:30 | 06:45/14:55 | 100 |
| 12362 | Avantika Express(Jab-Gwa) | Jab/Gwa | Gwa/Jab | 15:50/02:30 | 08:45/16:55 | 100 |
| 12358 | Gatimaan Express(Dws-Bpl) | Dws/Bpl | Bpl/Dws | 18:50/04:30 | 06:45/14:55 | 100 |
+-----+-----+-----+-----+-----+-----+
18 rows in set (0.00 sec)

mysql> Update Train set SeatsRemaining=100 where SeatsRemaining=98;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from Train;
+-----+-----+-----+-----+-----+-----+
| TrainId | TrainName | FromStation | ToStation | Starttime | Endtime | SeatsRemaining |
+-----+-----+-----+-----+-----+-----+
| 12345 | Rajdhani Express(Jab-Gwa) | Jab/Gwa | Gwa/Jab | 18:50/04:30 | 06:45/14:55 | 100 |
| 12346 | Avantika Express(Ind-Jab) | Ind/Jab | Jab/Ind | 11:50/06:30 | 06:45/12:55 | 100 |
| 12347 | Mayanagri Express(Ind-Gwa) | Ind/Gwa | Gwa/Ind | 18:50/04:30 | 06:45/14:55 | 100 |
| 12348 | Ajanta Express(Jab-Bpl) | Jab/Bpl | Bpl/Jab | 18:50/04:30 | 06:45/14:55 | 100 |
| 12349 | Amravati Express(Bpl-Gwa) | Bpl/Gwa | Gwa/Bpl | 18:50/04:30 | 06:45/14:55 | 100 |
| 12350 | Holkar Express(Ind-Ujj) | Ind/Ujj | Ujj/Ind | 18:50/04:30 | 06:45/14:55 | 100 |
| 12351 | Bhoj Express(Ind-Dws) | Ind/Dws | Dws/Ind | 18:50/04:30 | 06:45/14:55 | 100 |
| 12352 | Ujjaini Express(Dws-Ujj) | Dws/Ujj | Ujj/Dws | 18:50/04:30 | 06:45/14:55 | 100 |
+-----+-----+-----+-----+-----+-----+

```

Ankit Drive (J:)
Tap to choose what happens with removable drives.

```
MySQL 5.5 Command Line Client
mysql> select * from user;
Empty set (0.00 sec)

mysql> select * from reservations;
Empty set (0.00 sec)

mysql> select * from user;
+-----+-----+-----+
| UserId | Username | Aadhar | Password |
+-----+-----+-----+
| 392 | Jishan | 23454375 | jjj |
| 689 | Neeraj | 33454375 | nrj123 |
| 518 | Neeraj | 33454375 | nrj |
| 19 | mahendra singh dhoni | 55555555 | mah |
| 934 | Ankit Chouhan | 12323358 | ankit |
| 941 | Ankit Chouhan | 12323358 | ankit |
| 196 | Ankit Chouhan | 12323358 | ankit |
| 504 | Subhash Tiwari | 98767598 | sanju |
| 880 | Subhash Tiwari | 98767598 | sanju |
| 430 | Neeraj kuar | 12345678 | 1234 |
| 150 | Neeraj kuar | 12345678 | 1234 |
| 974 | Subhash Tiwari | 98767598 | sanju |
+-----+-----+-----+
12 rows in set (0.00 sec)

mysql> select * from reservations;
+-----+-----+-----+-----+-----+-----+-----+
| ReservationId | PassengerId | PassengerName | FromStation | ToStation | TrainId | TrainName |
+-----+-----+-----+-----+-----+-----+-----+
| 822 | 974 | Subhash Tiwari | Indore | Bhopal | NULL | Intercity Express(Bpl-Ind) |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select * from reservations;
Empty set (0.00 sec)

mysql>
```

E:\WAMP\wamp64\www\PHP_FINALSQL CODES - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

SQL CODES

```

1 create table User(
2     UserId int(10) PRIMARY KEY,
3     UserName varchar(30) NOT NULL ,
4     Aadhar int(12) NOT NULL,
5     Password varchar(8) NOT NULL
6 );
7
8
9 create table reservations(
10    ReservationId int(10) PRIMARY KEY,
11    PassengerId int(10) FOREIGN KEY REFERENCES User(UserId),
12    PassengerName varchar(30),
13    FromStation varchar(30),
14    ToStation varchar(30),
15    TrainId varchar(30) FOREIGN KEY REFERENCES Train(TrainId),
16    TrainName varchar(30)
17 );
18
19
20 create table Train(
21     TrainId int(6) PRIMARY KEY,
22     TrainName varchar(30),
23     FromStation varchar(30),
24     ToStation varchar(30),
25     StartTime varchar(15),
26     EndTime varchar(15),
27     SeatsRemaining int(3)
28 );
29
30 create table Station(
31     StationId int(10) PRIMARY KEY,
32     StationName varchar(30) NOT NULL
33 );
34
35 create table Employee(
36     EmployeeId int(10) PRIMARY KEY,
37     FirstName varchar(30) NOT NULL,

```

Line 5, Column 31

Spaces: 2 Plain Text

insert into Train(TrainId, TrainName, FromStation, -- Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

SQL CODES

```

1 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12345, 'Rajdhani Express(Jab-Gwa)', 'Jab/Gwa', 'Gwa/Jab', '18:50/04:30', '06:45/14:55',100);
2
3 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12346, 'Avantika Express(Ind-Jab)', 'Ind/Jab', 'Jab/Ind', '11:50/06:30', '06:45/12:55',100);
4
5 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12347, 'Mayanagri Express(Ind-Gwa)', 'Ind/Gwa', 'Gwa/Ind', '18:50/04:30', '06:45/14:55',100);
6
7 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12348, 'Ajanta Express(Jab-Bpl)', 'Jab/Bpl', 'Bpl/Jab', '18:50/04:30', '06:45/14:55',100);
8
9 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12349, 'Amravati Express(Bpl-Gwa)', 'Bpl/Gwa', 'Gwa/Bpl', '18:50/04:30', '06:45/14:55',100);
10
11 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12350, 'Holkar Express(Ind-Ujj)', 'Ind/Ujj', 'Ujj/Ind', '18:50/04:30', '06:45/14:55',100);
12
13 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12351, 'Bhoj Express(Ind-Dws)', 'Ind/Dws', 'Dws/Ind', '18:50/04:30', '06:45/14:55',100);
14
15 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12352, 'Ujjaini Express(Dws-Ujj)', 'Dws/Ujj', 'Ujj/Dws', '18:50/04:30', '06:45/14:55',100);
16
17 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12353, 'Kshipra Express(Ujj-Jab)', 'Ujj/Jab', 'Jab/Ujj', '18:50/04:30', '06:45/14:55',100);
18
19 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12354, 'Narmada Express(Ujj-Gwa)', 'Ujj/Gwa', 'Gwa/Ujj', '18:50/04:30', '06:45/14:55',100);
20
21 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12355, 'Godavari Express(Jab-Dws)', 'Jab/Dws', 'Dws/Jab', '17:50/03:30', '06:45/14:55',100);
22
23 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12356, 'Gwaliori Express(Dws-Gwa)', 'Dws/Gwa', 'Gwa/Dws', '18:50/04:30', '06:45/14:55',100);
24
25 insert into Train(TrainId, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining) Values(12357, 'Malwa Express(Bpl-Ujj)', 'Bpl/Ujj', 'Ujj/Bpl', '18:50/04:30', '06:45/14:55',100);

```

197 characters selected

Tab Size: 4 Plain Text

Chapter 8

Testing

Procedure

The procedure of testing of SQL codes used is quite simple, standard, effective. The steps used in complete testing procedure is shown below-

1. Checking for SQL Syntax by directly running the queries
2. Checking for broader logics of code: Is code working according to logics or not?
3. White box Checking
 - 3.1 Structural testing which checks outer structure of logic
4. Black box checking
 - 4.1 Functional testing which checks if code working according to functionality or not.
5. Coding standards which is not a testing step but we check for some standard code including good formatting, alignments, procedure calls, etc.

Outcomes

Testing provides us to minimize sort of error prone functionalities as per our implementation. Here, following simple procedure for testing we get desired hassle free functionality. Project is currently working fine and since testing procedure is available, we can further apply that testing procedure in case of malfunction or bug findings.

Chapter 9

Conclusions and future Scope

Conclusions:

Usage of database today is of utmost importance for an enterprise or an institution because it eliminates all the major drawbacks of file system via Programming language or Offline file saving. It helps in improving the whole procedure of data/information storage and retrieval. There are also some risks such as injection techniques, Reliability issues, data inconsistency risks, etc. Hence it requires to construct a proper Risk Mitigation, Monitoring, and Management plan (RMMM Plan) for specific usage of databases practically.

We keep our project small and simple which led to some major improvements in costs, efforts, and time usage in complete life cycle of project development. We've tried to include all major functionalities which must be there in a metro/railway system database but still it is far behind than a practical working software. It requires large number of attributes their manipulation and maintainance. We've completed major objectives of our project and the project is now in its maintainance phase which will continue at Github where it will be proper version controlled and is properly maintained by subject experts, reviewers, students, etc.

We had not built RMMM plan for every risk instead we take look at each and every feasible risk in development procedure, and simple eliminate it by whatsoever reasons.

Future Scope and improvements:

In near future the usage of databases will probably be decreased due to huge sources of data and new technologies such as Big data analytics which will further improve the drawbacks of database management system.

The project output can be used as raw input to a bigger project which may led to practical implementation of various concepts and debugging. This can be enhanced much on following points-

1. User Interface:

- It can be ported to a simple application using android studio/iOS builder which can then be published on Play Store, or iOS Store.
- It can be reconstructed by Graphics programming user Java Swing, Java awt, or Python Tkinter for web applications.

2. Implementation:

- It can be implemented using Oracle database 12C, DB2, PostgreSQL, MongoDB. LibreOffice is currently using HSQLEmbedded as embedded database, instead of which other databases can be used as per requirements.
- The relational schema for Metro System can also be implemented on Object Oriented database which has its own pros and cons.

3. Methodology: We've used simple yet hybrid methodology for customization of events in development of our project.

- Agile methodology can be used completely which will take the performance of project development at high level.
- Extreme Programming practices and activities can also be used in Project development life cycle which has its own further improvements.

4. Standards: We've simply used unorganized guidelines and develop project using random software standards.

- Instead of random software standards we can use organised ISO (International standards organisation) standards and guidelines such as ISO 9001-2008.
- Capability Maturity Model (CMM) Guidelines can be used in organised manner in at least level-3.
- Six Sigma certification aspects can also be incorporated in project development.

References and Bibliography

- 1] Database System concepts. Abraham Silberschatz, Henry F. Korth, S. Sudarshan. 6 th International editon. McGraw-Hill, New York.
ISBN 978-0-07-352332-3. MHID 0-07-352332-1.
- 2] Database Systems The Complete Book. Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom. 2nd editon. Prentce Hall, New Jersey.
ISBN 0-13-606701-8, 978-0-13-606701-6.
- 3] Database Management Systems. Raghu Ramakrishnan, Johannes Gehrke. 2 nd editon. McGraw Hill higher educaton. ISBN 0-07-246535-2.
- 4] Database Management System (DMS) tutorial. Tutorials Point.
SQL tutorial. Tutorials Point. <https://www.tutorialspoint.com/>.
- 5] Scribed Class notes on Database Management Systems (CSE-224) at MANIT, Bhopal. Instructor: Ms. Shweta Jain.
- 6] Why Are Database Management Systems Important to Business Organizations? Gerald Hanks. Retrieved- April 4, 2018.
<https://yourbusiness.azcentral.com/database-management-systems-important-business-organizations-21106.html>
- 7] Techtarget. Retrieved- April 4, 2018. Definition of SQL
<https://searchsqlserver.techtarget.com/definition/SQL>
- 8] Webpage template (login-form-13) from <https://www.colorlib.com/wp/>

*Chapter 2 (Literature Survey) is solely based on (6) and (7).

Appendix A

List of tables, forms, and queries

List of tables

1. User (UserName, UserID, Aadhar, Password)
2. Reservations (ReservationID, PassengerID, PassengerName, FromStation, ToStation, TrainID, TrainName)
3. Department (DepartmentName, DepartmentID, StationID)
4. Employees (EmployeeID, FirstName, LastName, Salary, BirthDate, DepartmentID, StationID, StationName, Phone)
5. Payments (PaymentID, UserID, UserName, PaymentDate, Amount)
6. Stations (StationID, StationName)
7. Trains (TrainID, TrainName, FromStation, ToStation, StartTime, EndTime, SeatsRemaining)

List of web-pages

1. front.php
2. booking.php
3. enquiry.html and enquiry.php
4. signup.php
5. login.php
6. cancel.php

List of Sample queries

1. To find all users with all their details
2. To find description of all trains started from Bhopal
3. To find User having Aadhar=65695486
4. To find names of all employees having salary greater than all other employess
5. To Find out train which ends at Indore and having remaining seats at least 1
6. To find UserID of user paymented INR 10,000 at 23 March 2018.

**Queries can be further written or customized as per requirements.

Appendix B

SQL code for sample queries

1. To find all users with their details

```
SELECT *
FROM "users";
```

2. To find description of all train started from Bhopal

```
SELECT *
FROM "Trains"
WHERE "FromStation" LIKE 'Bhopal';
```

3. To find user whose Aadhar is 65695486

```
SELECT "FirstName", "LastName", "UserID"
FROM "Users"
WHERE "Aadhar" = 65695486;
```

4. To find names of all employees having salary greater than all other employess

```
SELECT "FirstName", "MiddleName"
FROM "Employees"
WHERE "Salary" >= ALL ( SELECT "Salary" FROM "Employees" );
```

5. To Find out train which ends at Indore and having at least 1 seat unreserved

```
SELECT *
FROM "Trains"
WHERE "EndStation" = 'Indore' and "SeatsRemaining">>=1;
```

6. To find UserID of users paymented INR 10,000 at 23 March 2018

```
SELECT "UserID", "UserName"
FROM "Payments" as p, "User" as u
WHERE u."UserID"=p."UserID" and p."Amount"=10,000 and
p."PaymentDate"="23/03/2018";
```

Appendix C

Critism and Expert reviews

To be written soon...

Thanks for reviewing the document :)

End of the document :)