**University of Liberal Arts Bangladesh (ULAB)**

Term Final Project

Fall **201**9

Database Systems Lab

**CSE** 304 **(Sec-0**2**)**

3. Md. Amanullha (173014040)

**List of Index**

|  |  |  |
| --- | --- | --- |
|  |  | **Page** |
| **Chapter 1** | **Introduction** | **3** |
| **1.1** | Introduction | **4** |
| **1.2** | Motivation | **4** |
| **1.3** | Scope | **4-5** |
|  |  |  |
| **Chapter 2** | **Requirements Analysis** | **6** |
| **2.1** | Required Tools | **7** |
| **2.2** | Users of the Database | **7** |
| **2.3** | Entity and Attributes of the Database | **7** |
| **2.4** | Schema diagram | **8** |
| **2.5** | Entity Relationship Diagram (ERD) | **9** |
| **2.6** | Normalization | **10** |
|  |  |  |
| **Chapter 3** | **Implementation of the Database** | **11** |
| **3.1** | Database creation | **12** |
| **3.2** | Table creation | **12-14** |
| **3.3** | Relationship among the Tables | **15** |
| **3.4** | Sample data insertion | **15-18** |
| **3.5** | View reports (SQL query) | **18-21** |
|  |  |  |
| **Chapter 4** | **Conclusion** | **22/23** |

**Chapter 1: Introduction**

* 1. **Introduction**

A database management system is software that provides a user with a systematic process of creating, storing, organizing, retrieving, updating, and managing data in database. A critical component of a database management system is the centralized view of data that verifying users can access from multiple locations. Additional attributes include logging and auditing of activity, change management, performance monitoring, and backup and recovery. Concurrency, security and data integrity are also pivotal. The main purpose behind our database project is to create a Bus Reservation Database management system. This project presents a review on the “Bus ticket reservation system” that facilitates the users to search for buses for a destination, book tickets, cancel tickets etc.

The aim of this project is to design and develop of a system that can accomplish the tasks automatically. The reservation system is a complex system in reality. It is not feasible to develop the whole system. Therefore, a small sample case study has been created to demonstrate the workings of a reservation system.

We have made some assumptions to keep the database as simple as possible. The number of buses are kept minimum, there is no password for the user yet, there are no classes in ticket category.

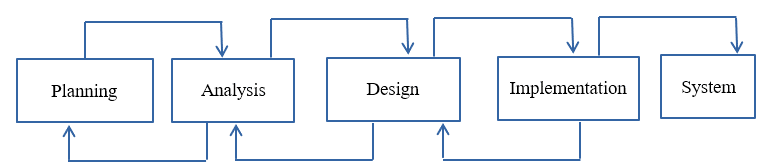
* 1. **Motivation**

The purpose behindcreating the database is to make ourselves familiar with an online ticket reservation database management system. The database of a complex system like railway online reservation or flight reservation intrigued us. A database for such purposes is extremely valuable to all the related entities. How big systems like IRCTC (Indian Railway Catering and Tourism Limited) books millions of tickets in a day using their database had interested us. Therefore, we decided to study a system like this and come up with a simplified version of the database of a Bus Ticket reservation system.

* 1. **Scope**

This database is created for those people who wants to book an online bus ticket for their journey. To make the life easy of the common people we want to improve this database. With the help of this kind of DB user can buy tickets from their home easily. Here they must be users of this database. The database we are designing is similar to any online transportation reservation system. Therefore, this system can be implemented in railway reservation, flight reservation or any kind of transportation system in general with very little modification.

**Approaches for creating our database:**



We have followed the above approach to build our database. Really the above process helps us a lot to establish the database. Because without any calculation, planning it is not possible to reach the final goal.

**Chapter 2: Requirements Analysis**

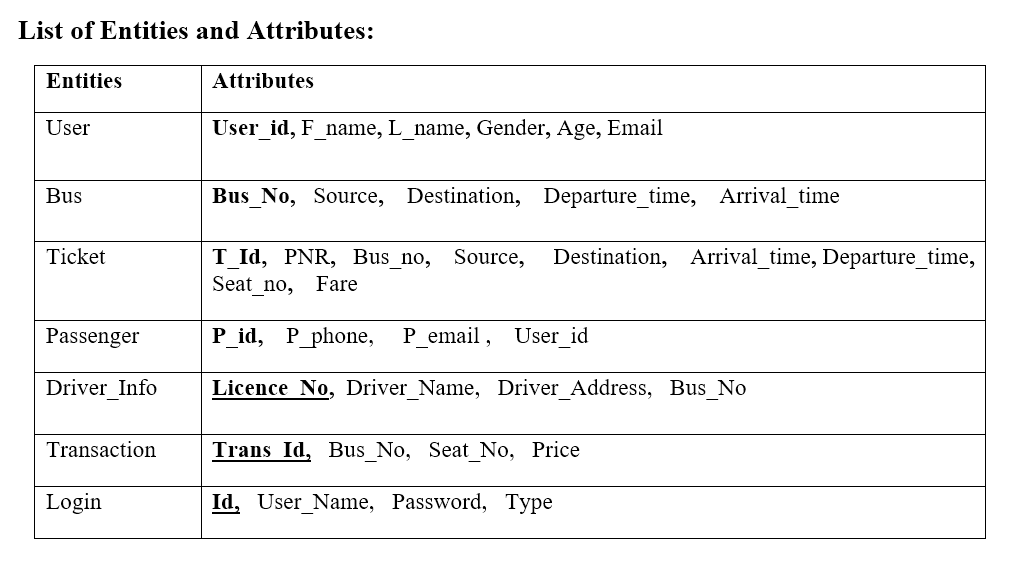
* 1. **Required Tools**

Mainly we have used the XAMPP software to create the database. We have followed phpMYSQL language to create the tables, creating the relations among tables, declaring primary key, foreign key and also updating, deleting, searching, retrieving the records from table records etc. The main aim of the project is the management of the buying tickets, make the journey easy for the passenger, to help the people know about the price from source to destination easily. This is done by creating a database which we actually give name **Bus ticket reservation System**. The database is then connected to the main program by using interconnection of the Visual Basic program and the database already created.

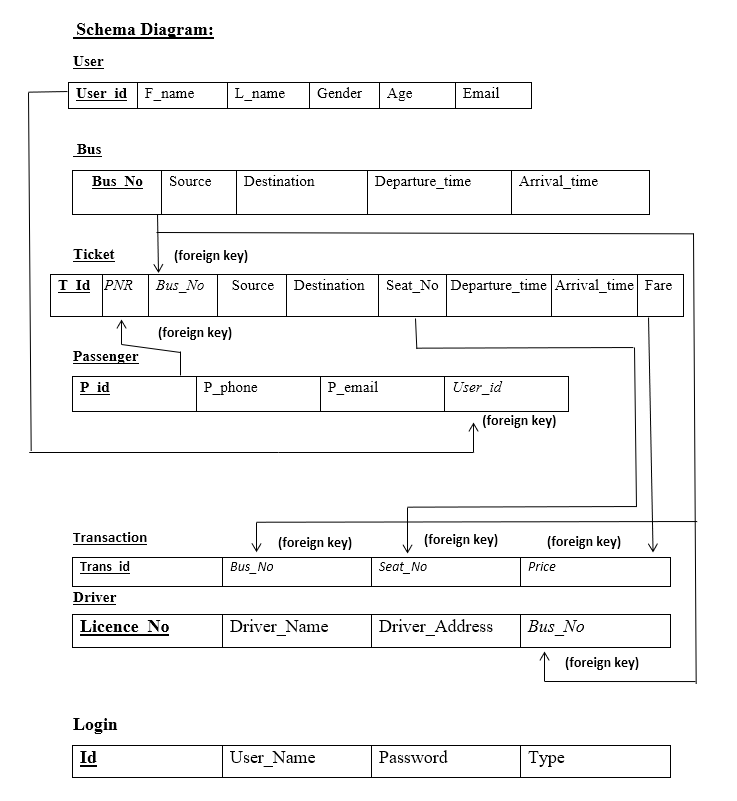
**2.2** **Users of the Database**

Database users are the one who really use and take the benefits of database. There will be different types of users depending on their need and way of accessing the database. There are also database developers, who write SQL queries to select/insert/delete/update data. They do not use any application or programs to request the database. They directly interact with the database by means of query language like SQL. These users will be scientists, engineers, analysts who thoroughly study SQL and DBMS to apply the concepts in their requirement. In short, we can say this category includes designers and developers of DBMS and SQL. Here our team plays the role as a database developer. We are also acting as an admin part. The lion share part of the database are the passengers. They are the lion share users. These are the users who use the existing application to interact with the database. For example, online library system, ticket booking systems, ATMs etc. which has existing application and users use them to interact with the database to fulfill their requests. In our case our Bus Ticket reservation System is an application that is for passengers who can use the database. They are called native users.

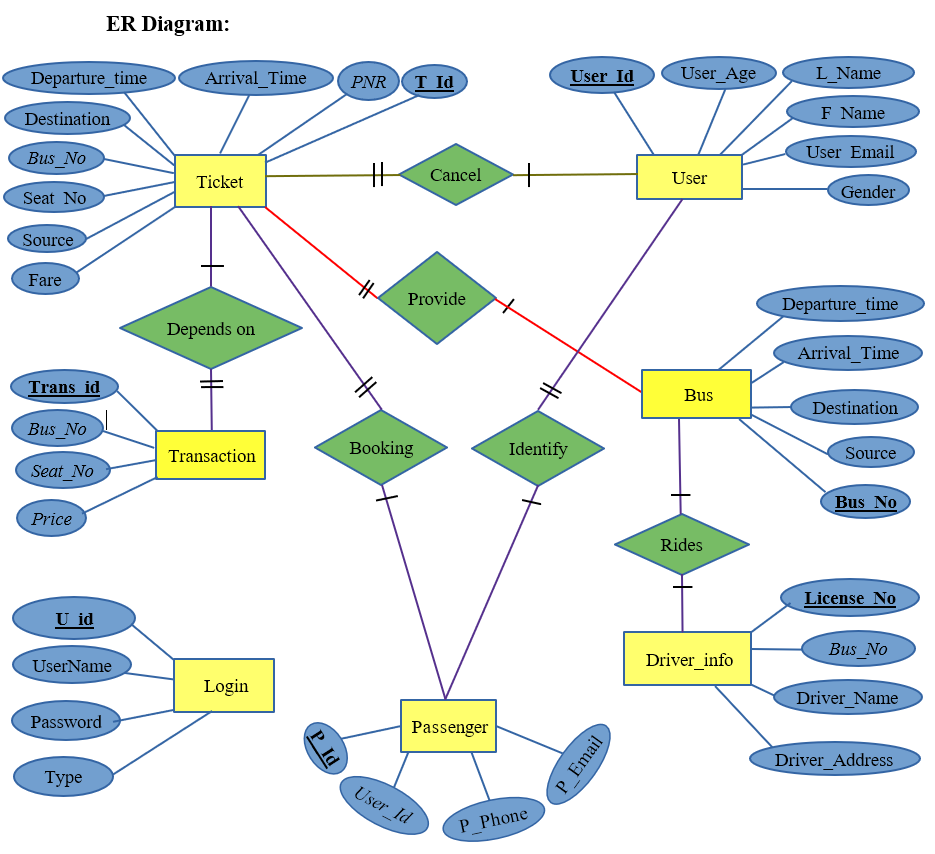
**2.3 Entity and Attributes of the Database**



**2.4 Schema Diagram:**

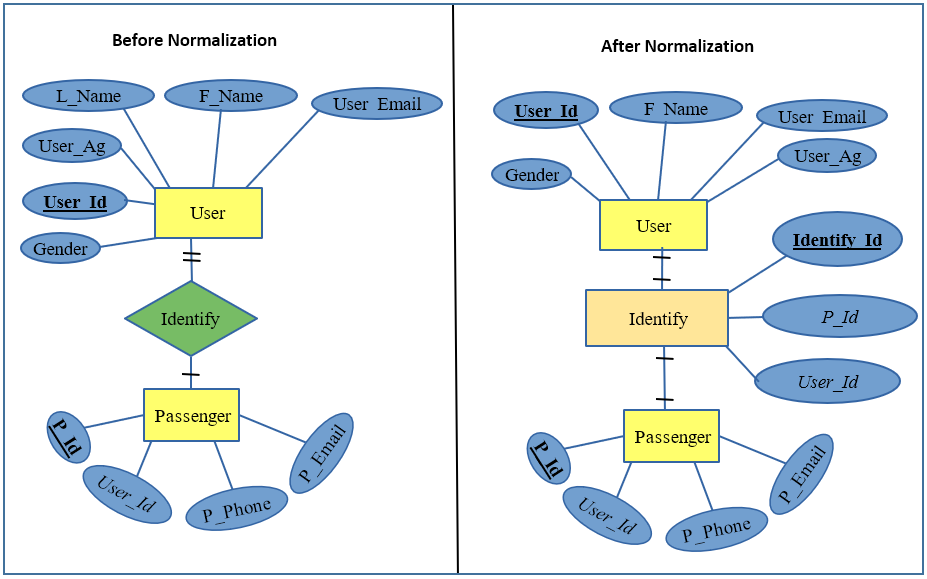


**2.5 Entity Relationship Diagram**



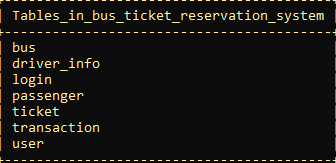
**2.6 Normalization**

Normalization split a large table into smaller tables and define relationships between them to increases the clarity in organizing data. It reduces the redundancy of data and also make the relation so specific. In our database we also use the normalization method to make the relation so specific. Here in user and passenger relation there were actually one to many relations at first time. But later we create another table name **Identify** that in indeed contains the primary key from two tables named **P\_id, User\_id** that makes the user passenger relation so specific. And this third table also contain its own primary key called **Identify\_id**.



**Chapter 3: Implementation of the Database**

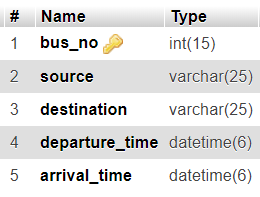
**3.1 Database creation**



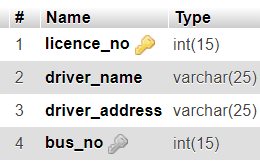
**3.2 Table Creation:**

**Tables…**

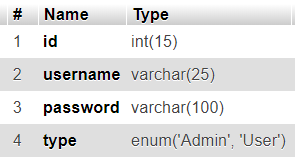
1. **Bus**



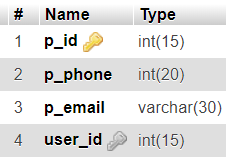
1. **Driver\_info**



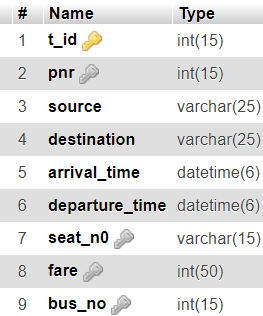
1. **Login**



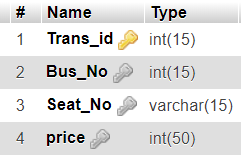
1. **Passenger**



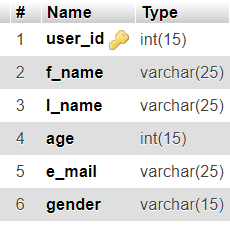
1. **Ticket**



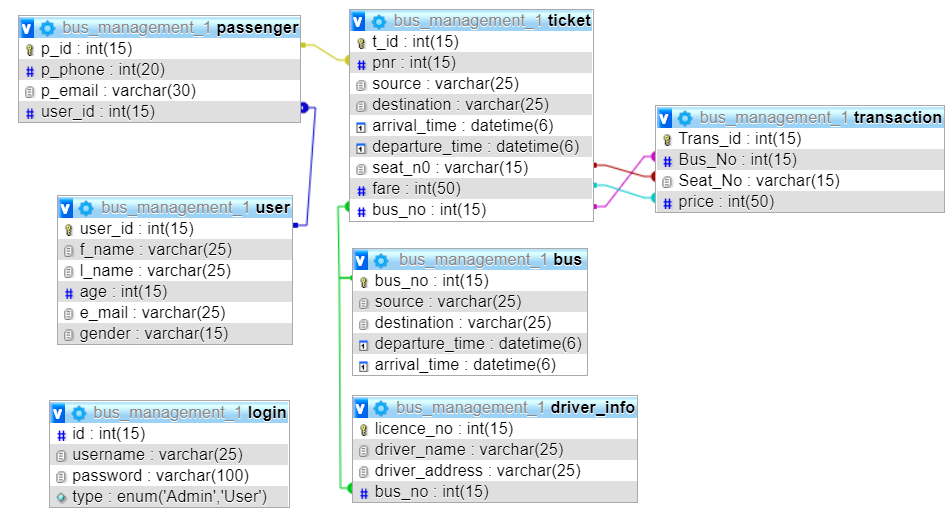
1. **Transaction**



1. **User**

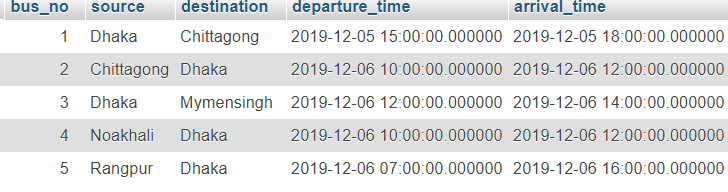


**3.3 Relationship among the Tables**

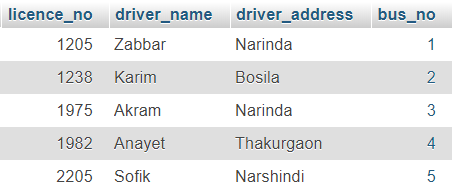


**3.4 Sample data insertion**

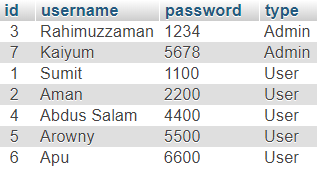
1. **Bus**



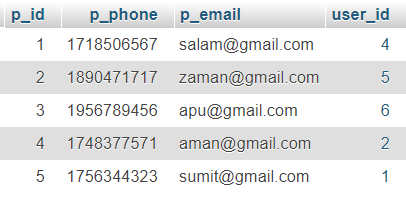
1. **driver\_info**



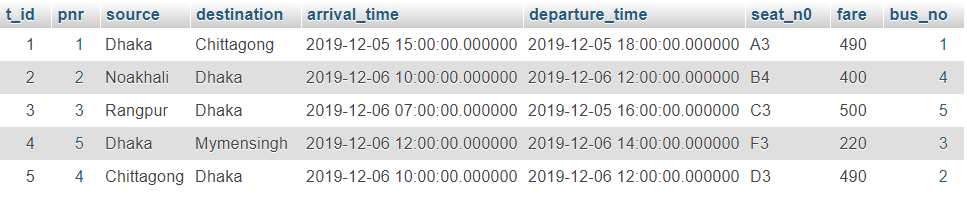
1. **login**



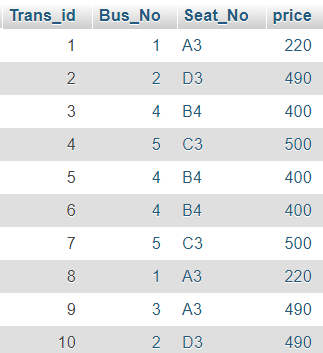
1. **passenger**



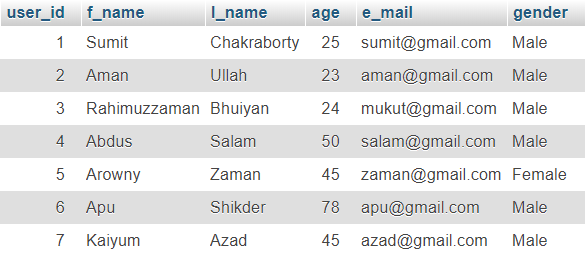
1. **ticket**



1. **transaction**



1. **user**



**3.5 View reports (SQL query)**

**DB project SQL:**

**1.** Passenger want to know the time of bus from Dhaka to Chittagong:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) bus.source,bus.destination,bus.departure\_time,bus.arrival\_time FROM

bus WHERE bus.source='Dhaka' [AND](mysql_doc) bus.destination='Chittagong'



2. Suppose a passanger (5) want to go from Dhaka to Mymensingh. He/She wanted to know his/her seat number and ticket price:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) passenger.p\_id,ticket.t\_id,ticket.source,ticket.destination,ticket.seat\_n0,

ticket.fare,ticket.bus\_no FROM passenger JOIN ticket WHERE passenger.p\_id=5

[AND](mysql_doc) ticket.pnr=5



3. Suppose a passanger want to know the fare of two place:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) ticket. source, ticket. destination, ticket. fare FROM ticket WHERE

ticket. source='Dhaka' [AND](mysql_doc) ticket. destination='Chittagong' [OR](mysql_doc)

ticket. source='Rangpur' [AND](mysql_doc) ticket. Destination='Dhaka'



4. An admin want to know how many buses are there whose source is not Noakhali and Rangpur:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [COUNT](mysql_doc)(bus.bus\_no) FROM bus WHERE [NOT](mysql_doc) bus.source='Noakhali' [AND](mysql_doc) [NOT](mysql_doc) bus.source='Rangpur'



5. **Suppose** the admin want to find out driver who is responsible for the bus containing the highest bus\_id:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [MAX](mysql_doc)(bus.bus\_no),driver\_info.driver\_name,driver\_info.driver\_address FROM bus JOIN driver\_info WHERE bus.bus\_no=driver\_info.bus\_no



6. Suppose two passengers face problem that they are not getting email or message on their mobile phone on time when they purchase a ticket. Now the respective admin want to see the whole information about them from database so that he can solve this:

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) [user](mysql_doc).f\_name ,passenger.p\_phone,passenger.p\_email,passenger.p\_id,passenger.user\_id FROM [user](mysql_doc) JOIN passenger WHERE user. User\_id=passenger.user\_id [AND](mysql_doc) [user](mysql_doc).f\_name [IN](mysql_doc)('Arowny','Apu')



7. **Suppose** an admin want to know the total transaction which is included the total monthly and yearly income from a bus and also the total monthly and yearly income from all bus. The admin also want to know the yearly income from a particular route:

**Monthly income from a particular bus(bus number 2) depands on total seats has sold**









**Monthly income from all buses depands on total seats has sold**



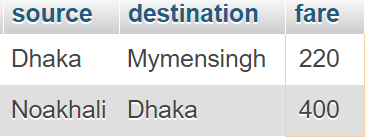






1. **. Suppose** an user want to see how many source and destination are there in a database that satisfy his/her price range:





**Chapter 4: Conclusion**

|  |  |
| --- | --- |
|  |  |

**Conclusion:**

In our bus ticket reservation system, we have stored all the information about the bus schedule, the user booking tickets even the status of buses seats etc. The database is helpful for the applications which facilitate passengers to book the bus tickets and check the details of bus and their place itself it avoids inconveniences of going to bus station. We have tried our best to make this application user friendly where the lion share users are passengers and there is admin also to maintain the database. As this is our database lab-based project, we have done a lot to build our project by applying our whole knowledge and also tried to give this a premium taste. But we confess that there are lot of bus management system applications now a days and we will improve our project in future by acquiring more knowledge on this field. Here to testing the project weather all queries are run well on not we have inserted only few records as well in every entity. And we have successfully run the all queries that we have learn from this course.

**THE END**