THAPATHALI CAMPUS
Institute of Engineering

A FINAL REPORT

ON

BUS RESERVATION MANAGEMENT SYSTEM

IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS FOR THE COURSE OF

BACHELOR OF ELECTRONICS, COMMUNICATION, AND INFORMATION ENGINEERING

SUBMITTED BY

RIWAJ NEUPANE	THA075BEI034
SAMIT BARAL	THA075BEI036
SUSHIL RAI	THA075BEI045

SUBMITTED TO

Rama Bastola

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

THAPATHALI CAMPUS

AUGUST, 2021

DECLARATION

We, hereby declare that the project work report entitled "**BUS RESERVATION MANAGEMENT SYSTEM**" submitted for the partial fulfillment of the requirements for the course of CT 610 Database Management Systems are our original work and the project work report has not been formed on the basis for the award of any degree, diploma, or other similar titles.

Signature:

Name of Student: Riwaj Neupane

Signature: 

Name of Student: Samit Baral

Signature: 

Name of Student: Sushil Rai

Signature: 

ACKNOWLEDGEMENTS

It is a matter of pleasure of acknowledgement by indebtedness to Mrs. Rama Bastola for providing us the opportunity to work on this project. The experience of doing this project will surely enrich our technical and teamwork skills to a great extent.

ABSTRACT

‘Bus Reservation Management System’ is an application which helps to maintain the day-to-day records and information for the bus transportation organizations. It should be used in a bus transportation system, a facility which is used to reserve seats, cancellation of reservation and different types of route enquiries used on securing quick reservations. So, our project ‘Bus Reservation Management System’ is made to make the people’s life easier by providing the information about the availability of the vehicles and keeping track of such details and easy records for the data. It helps the vehicle organization to keep perform their task more efficiently in an errorless manner. Also, keeping the records offline risks the loss or theft of data. So, the application can be used to keep the records of all such data in database to be secured from all such possibilities.

There are many existing systems for ‘Bus Reservation Management System’ currently being used. Our proposed program will not only help people to know about the current fares for the buses for different locations, but it will also help people from the organization to properly manage the data and deal with both these features along with other problem’s solution like making reports on profit of the company in the future. So, for all these our program will come handy.

Some module included in or program are:

- Ticket Info: User can get all the info related to availability of tickets.
- Ticket booking: In this module user can book their ticket as per their preferred time.

Key Words:

Booking: To reserve place for participation in some event before that specific event is carried out is known as booking. Our project also deals with details for ticket booking.

Fare: It is the money paid for a journey on a public transport. Our project also maintains the fare cost.

.

Table of Contents

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Background

With the modernization and perfectly packed busy schedule of people it has been one of the tedious tasks to purchase the tickets for bus for travel from the offline stores. It is extremely important to keep records for the different operations such as keeping track of tickets for bus, records of customers and many more which is too hard and time consuming for offline stores which mainly rely on offline data record keeping system. Especially in context of Nepal, there aren't enough online services for online ticketing of such vehicles. Also, the ones that are present aren't user friendly for all. Recording keeping is one of the key aspects in day-to-day business life. So, suitable alternative is must. It helps the organization for information gathering processing and proper use of such records. For this reason, managers and other staffs in transport industries are demanding more kinds of information to support management and operations. They must need to address this increasing requirement for information and data management.

Currently, staff at the bus ticket counter is using an on-site ticket system to sell tickets at the counter at the site and customers who are unable to buy bus ticket online at this moment would have to go to the counter to a buy bus ticket. Sometimes, customers' needs to queue up a long queue to buy bus ticket and ask for information and this brings a lot of inconveniences to customers. Also, there might not be availability of tickets/ or all tickets might get sold, which might be both time consuming for people, who might have stayed in the queue as well. So, Online Bus Ticket Reservation System enables the customer to query about bus ticket, and ask for information online easily.

1.2 Problem Statement

There are some of problems that lead us to develop this program. Currently, the type of system being used at the counter is an internal system which is manually used in selling the bus tickets. The problems facing the company are that customers have to go to the counter to buy bus ticket or ask for bus schedule, customers will also have to queue up for a long time in order to secure a bus ticket and will also need to pay cash when they buy the bus ticket.

1.3 Objectives

The main objectives behind developing this program are that:

- To increase the speed of services, and customers satisfaction.
- To eliminate paper work and increase level of accuracy.
- To know reservation status, reserved seats and remaining available seats for booking from bus_id.
- To be able to know about schedules of different buses simply from their unique bus_id along with their arrival time and departure time.

1.4 Application

Our program will come handy in managing bus reservation centers from view point of provider along with the benefit for viewer of consumer.

- It is designed to increase efficiency, save money and time.
- It shows the correct schedules and enable the customers to order the ones that are available.
- It helps to avoid long queues at the counter.

1.5 Project Feature

Our project features are given below:

- Information about passengers
- Viewing information about tickets.
- Booking ticket
- Watching ticket info

1.6 Feasibility Analysis

Feasibility Study is a test of the system according to its workability, impact of the organization, ability to meet user needs and effective use of the resources. It has good user interface which can be easily understood by the people with minimum

knowledge of the use of such programs as well. We have checked the feasibility in the basis of three factors:

1.6.1 Economic Feasibility

Economic feasibility is the important factor for determining the efficiency of a new project. It is also known as cost analysis. If the cost is too high for the small software, it is not economically feasible. It helps in identifying profit against investment expected from a project. Cost is one of the most essential factors involved in the field of study. This program is economically highly feasible program since it is written in SQL So economically it is highly satisfying with minimum time consumption.

1.6.2 Technical Feasibility

Technical feasibility study is the complete study of the project in terms of input, processes, output, fields, programs and procedures. It is a very effective tool for long term planning and trouble shooting. The technical feasibility study should most essentially support the financial information of an organization. It is technically well managed program whose source code can be easily modified for better improvement and for debugging any error.

1.6.3 Operational Feasibility

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. For the fulfillment of the operational feasibility, the program should be able to be brought changes as per the requirement of the user and new features should be able to be added in it. It takes care that the management and the users support the project. It is very easy to use. A person who understands English language can use this program. Data added to this program are easily modifiable.

1.7 System Requirement

1.7.1 Software Requirement

- MS Word
- XAMPP
- Windows OS
- MYSQL

1.7.2 Hardware Requirement

- Computer
- Flash Drive

CHAPTER 2: LITERATURE REVIEW

Introduction

This chapter summarizes findings from a literature review related to different bus reservation offices and counters. Reports are group into two categories: Case Study and Online Research.

2.1 Database Management System

Database is an organized collection of related information so that it can easily be accessed, managed, and updated. It is a repository of logically linked and similar data. Database Management System (DBMS) is a collection of interrelated data and a set of programs to access those data. DBMS provides a way to store and retrieve database. It is designed to manage large bodies of information.

Management of data involves defining structure for storage of information, providing mechanism for manipulation of information, data security, data integrity and consistency. Examples: MySQL, PostgreSQL, Microsoft Access, SQL Server, etc.

2.2 Entity Relationship Model

Entity Relationship (ER) modeling is a logical database design modeling technique. After the business requirements and data requirements are gathered, development of logical data model is started describing the data involved in the real word enterprise in terms of objects and their relationships. It is widely used to develop an initial database design. It was initially purposed by Dr Peter Chen in 1976.

An entity–relationship model describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types and specifies relationships that can exist between entities.

2.3 Case Study

Firstly, we visited different transportation offices and inquired about how the reservation system works. We asked the managers of various offices about their ways of storing the details and information about the buses, fares, passengers and many more. We took the survey on how the transportation offices have been keeping the track of various records related to bus reservation and how efficient they had been.

Besides that, we also took the interview of the people coming for the reservation of the bus tickets. We inquired them about how convenient it is for them to do the reservation and the problems that they have faced while booking the tickets. Taking all this information we analyze the problems and discussed on how they can be made effective and sustainable. The manual way of storing data had made it difficult for the company to generate the annual reports on the passengers and their reservations.

2.4 Online Research

We also did the online research and study on how the data can be stored in more secured and efficient way. A TRID (Transportation Research International Documentation) search was conducted using various keywords to aid the literature review. We read various blogs and reports on Bus reservation system and their ways of storing data.

2.5 Summary

The literature review summarizes major studies as well as reports looking at specific aspects of the transportation offices such as reservation process, report making and so on. Hence, the studies made us easier to understand the problems and facilitated us in finding the best alternatives of bus reservation and the data storage.

CHAPTER 3: METHODOLOGY

3.1. System Development:

System Design: It includes all the tables and data and deal with how they are managing the database.

Implementation: Different tables are created for different necessities and data were filled based on the required conditions.

3.1.1 Problem analysis:

The problem analysis is the process to identify all the problem related to the project and to make necessary mapping to solve them individually. The problems that should be addressed in our project are shown below:

- To be able to show details of reserved and not reserved seats in the bus.

3.1.2 Development approach

Our process at first takes take various types of input from the user and store those inputs in various tables present in databases. There are various internal procedures on the software on how the system works and after running various queries on thus created database can be used to show outputs demanded by the user.

3.2 System Designing:

The system design is the most complex process. The program can be operated with SQL. The system designing includes all the management of the project related things. It also deals with the overall look of the system and how the data and tables are organized inside the database. So, we approached this step keeping in mind, the most convenient and easier ways to deal with the use of this system. The data items need to be added and deleted; the table are needed to be managed safely.

3.3 **Work Schedule:**

The chart below gives the visualization of our task and the time to complete it. The plan was to complete the project within a month. In the initial stage, we planned to research what can we do and how we can do it. We also needed to collect the information regarding the bus reservation system in different places. Afterwards, we made a view level design with ER diagram and created conceptual level design. At the end, several testing and debugging were performed.

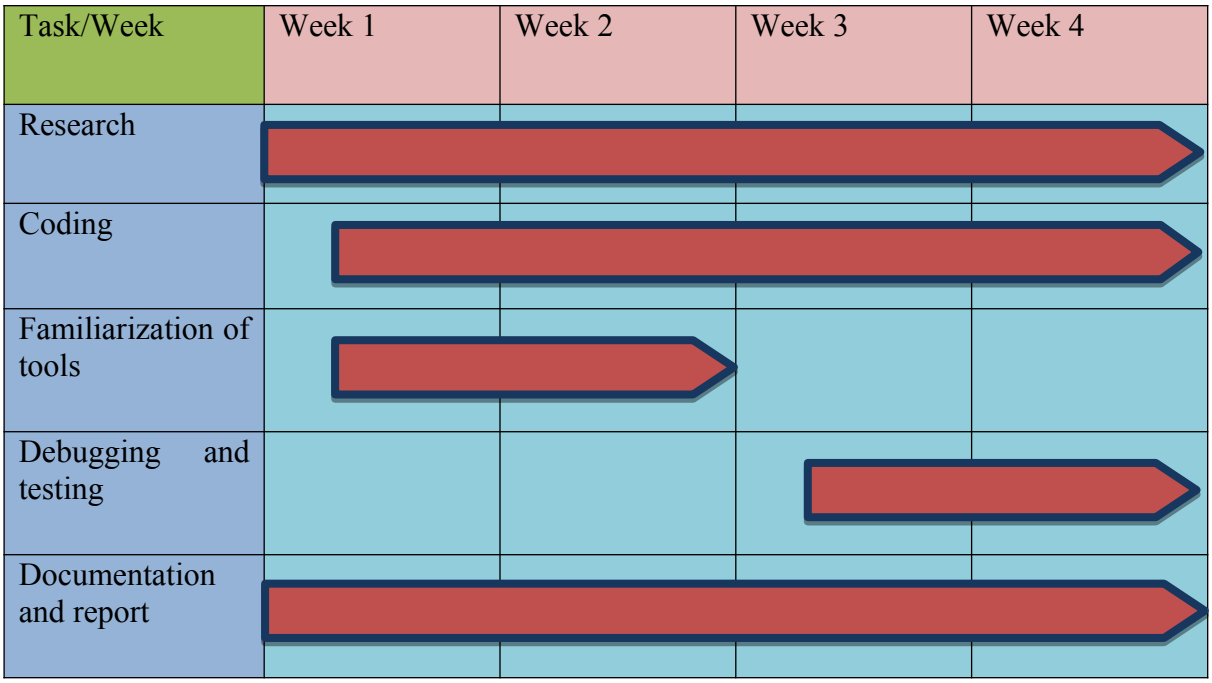
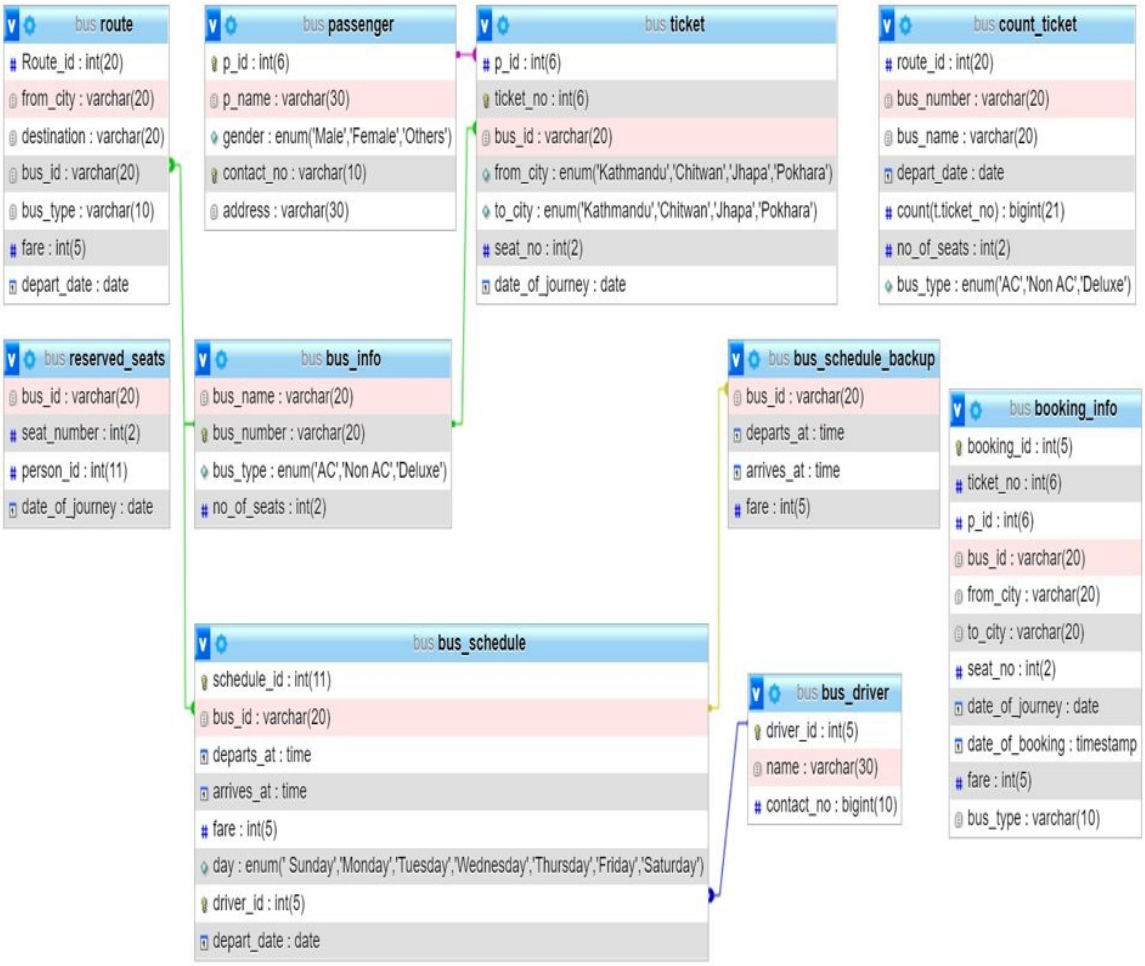


Fig: Work Schedule

ER Diagram



We created 8 different entities for this project. All the entities and their attributes are listed below:

- Passenger: p_id, p_name, gender, contact_no, address
- Bus_info: bus_name, bus_number, bus_type, no_of_seats
- Route: route_id, from_city, destination, bus_id, bus_type, fare, depart_date
- Ticket: p_id, ticket_no, bus_id, from_city, to_city, seat_no, date_of_journey
- Bus_driver: driver_id, name, contact_no
- Bus_schedule: schedule_id, bus_id, departs_at, arrives_at, fare, day, driver_id, depart_date
- Booking_info: booking_id, ticket_no, p_id, bus_id, from_city, to_city, seat_no, date_of_journey, date_of_booking, fare, bus_type
- Bus_schedule_backup: bus_id, departs_at, arrives_at, fare

CHAPTER 4: IMPLEMENTATION AND RESULTS

4.1 Implementation Details

A list of tables were created with their attributes. The lists of tables created along with their codes are as follows:

```
a) CREATE TABLE `booking_info` (
  `booking_id` int(5) NOT NULL,
  `ticket_no` int(6) DEFAULT NULL,
  `p_id` int(6) DEFAULT NULL,
  `bus_id` varchar(20) DEFAULT NULL,
  `from_city` varchar(20) NOT NULL,
  `to_city` varchar(20) NOT NULL,
  `seat_no` int(2) DEFAULT NULL,
  `date_of_journey` date NOT NULL,
  `date_of_booking` timestamp NOT NULL DEFAULT current_timestamp(
),
  `fare` int(5) DEFAULT NULL,
  `bus_type` varchar(10) DEFAULT NULL
)
```

```
b) CREATE TABLE `bus_driver` (
  `driver_id` int(5) NOT NULL,
  `name` varchar(30) DEFAULT NULL,
  `contact_no` bigint(10) DEFAULT NULL
   )
```

```
c) CREATE TABLE `bus_info` (
  `bus_name` varchar(20) DEFAULT NULL,
  `bus_number` varchar(20) NOT NULL,
  `bus_type` enum('AC','Non AC','Deluxe') DEFAULT NULL,
  `no_of_seats` int(2) NOT NULL DEFAULT 40
)
```

```
d) CREATE TABLE `bus_schedule` (
  `schedule_id` int(11) NOT NULL,
  `bus_id` varchar(20) NOT NULL,
  `departs_at` time DEFAULT NULL,
  `arrives_at` time DEFAULT NULL,
  `driver_id` int(5) DEFAULT NULL,
  `depart_date` date DEFAULT NULL
)
```

```
e) CREATE TABLE `bus_schedule_backup` (
  `driver_id` int(5) DEFAULT NULL,
  `bus_id` varchar(20) DEFAULT NULL,
  `departs_at` time DEFAULT NULL,
  `arrives_at` time DEFAULT NULL
)
```

```
f) CREATE TABLE `count_ticket_occupancy` (
`bus_id` varchar(20)
,`bus_name` varchar(20)
,`from_city` enum('Kathmandu','Chitwan','Jhapa','Pokhara')
,`to_city` enum('Kathmandu','Chitwan','Jhapa','Pokhara')
,`No of tickets` bigint(21)
,`no_of_seats` int(2)
,`date_of_journey` date
);
```

```
g) CREATE TABLE `driver_of_bus` (
`driver_id` int(5)
,`name` varchar(30)
,`contact_no` bigint(10)
,`bus_name` varchar(20)
,`bus_number` varchar(20)
,`bus_type` enum('AC','Non AC','Deluxe')
,`no_of_seats` int(2)
);
```

```
h) CREATE TABLE `passenger` (
  `p_id` int(6) NOT NULL,
  `p_name` varchar(30) DEFAULT NULL,
  `gender` enum('Male','Female','Others') DEFAULT NULL,
  `contact_no` varchar(10) DEFAULT NULL,
  `address` varchar(30) DEFAULT NULL
)
```

```
i) CREATE TABLE `reserved_seats` (
  `bus_id` varchar(20) DEFAULT NULL,
  `seat_number` int(2) DEFAULT NULL,
```

```
  `person_id` int(11) DEFAULT NULL,
  `date_of_journey` date DEFAULT NULL
)
```

```
j) CREATE TABLE `route` (
  `Route_id` int(20) NOT NULL,
  `from_city` varchar(20) DEFAULT NULL,
  `destination` varchar(20) DEFAULT NULL,
  `bus_id` varchar(20) DEFAULT NULL,
  `bus_type` varchar(10) DEFAULT NULL,
  `fare` int(5) DEFAULT NULL,
  `depart_date` date DEFAULT NULL
)
```

```
k) CREATE TABLE `ticket` (
  `p_id` int(6) DEFAULT NULL,
  `ticket_no` int(6) NOT NULL,
  `bus_id` varchar(20) DEFAULT NULL,
  `from_city` enum('Kathmandu','Chitwan','Jhapa','Pokhara') NOT N
ULL,
  `to_city` enum('Kathmandu','Chitwan','Jhapa','Pokhara') NOT NUL
L,
  `seat_no` int(2) NOT NULL,
  `date_of_journey` date DEFAULT NULL
)
```

4.2 Results

i. List of passengers

p_id	p_name	gender	contact_no	address
1	Sushil Rai	Male	987654654	baneshwor
2	Riwaj Neupane	Male	9899898897	Koteshwor
3	Samit Baral	Male	987212354	Bagbazar
4	Babu Chiri Sherpa	Female	987474147	Lokanthali
5	Ang Rita	Female	9879879877	abcxyz
6	Apa Sherpa	Others	9876546542	Chabel
7	Basanta Regmi	Male	9879876545	Naikap
8	Paras Khadka	Male	9877897897	Kalimati
9	Dayaram Dahal	Female	9877878787	Teku
10	Atul Gautam	Female	987879877	Tripureshwor
11	Harsha Bahadur	Male	987654321	kathmandu

ii. Ticket details

p_id	ticket_no	bus_id	from_city	to_city	seat_no	date_of_journey
1	1	2	Kathmandu	Chitwan	5	2021-08-15
2	2	2	Kathmandu	Chitwan	7	2021-08-16
3	3	2	Chitwan	Kathmandu	9	2021-08-17
4	4	2	Kathmandu	Pokhara	11	2021-08-15
5	5	2	Pokhara	Kathmandu	13	2021-08-16
6	6	6	Kathmandu	Jhapa	15	2021-08-17
5	7	2	Jhapa	Kathmandu	17	2021-08-15
4	8	2	Kathmandu	Pokhara	19	2021-08-15
7	9	2	Kathmandu	Pokhara	21	2021-08-15
8	10	4	Kathmandu	Pokhara	23	2021-08-15
5	11	2	Kathmandu	Pokhara	25	2021-08-16
9	12	4	Kathmandu	Jhapa	27	2021-08-15
5	13	7	Kathmandu	Pokhara	29	2021-08-17

iii. Bus Info

bus_name	bus_number	bus_type	no_of_seats
Makalu bus	1	AC	40
BlueSky Bus	2	AC	40
Makalu bus	3	Non AC	40
BlueSky Bus	4	Non AC	40
Makalu Bus	5	Deluxe	40
BlueSky Bus	6	Deluxe	40
Sunrise Bus	7	AC	40
Sunrise Bus	8	Non AC	40
Sunrise Bus	9	Deluxe	40

iv. Routes

Route_id	from_city	destination	bus_id	bus_type ▲ 1	fare	depart_date
1	- Drag to reorder. - Click to mark/unmark. - Double-click to copy column name.	Pokhara	1	AC	550	2020-08-20
3		kathmandu	7	AC	825	2020-08-21
2	Jhapa	kathmandu	2	AC	1100	2020-08-21
1	Pokhara	kathmandu	1	AC	550	2020-08-21
2	kathmandu	Jhapa	2	AC	1100	2020-08-20
3	kathmandu	Chitwan	7	AC	825	2020-08-20
2	Jhapa	kathmandu	6	Deluxe	1200	2020-08-21
1	Pokhara	kathmandu	5	Deluxe	600	2020-08-21
3	Chitwan	kathmandu	9	Deluxe	900	2020-08-21
3	kathmandu	Chitwan	9	Deluxe	900	2020-08-20
2	kathmandu	Jhapa	6	Deluxe	1200	2020-08-20
1	kathmandu	Pokhara	5	Deluxe	600	2020-08-20
1	kathmandu	Pokhara	3	Non AC	500	2020-08-20
1	Pokhara	kathmandu	3	Non AC	500	2020-08-21
2	Jhapa	kathmandu	4	Non AC	1000	2020-08-21
3	Chitwan	kathmandu	8	Non AC	750	2020-08-21
3	kathmandu	Chitwan	8	Non AC	750	2020-08-20
2	kathmandu	Jhapa	4	Non AC	1000	2020-08-20

v. Booking info

booking_id	ticket_no ▲ 1	p_id	bus_id	from_city	to_city	seat_no	date_of_journey	date_of_booking	fare	bus_type
44	1	1	2	Kathmandu	Chitwan	5	2021-08-15	2021-08-20 12:09:14	825	AC
45	2	2	2	Kathmandu	Chitwan	7	2021-08-16	2021-08-20 12:09:14	825	AC
46	3	3	2	Chitwan	Kathmandu	9	2021-08-17	2021-08-20 12:09:14	825	AC
47	4	4	2	Kathmandu	Pokhara	11	2021-08-15	2021-08-20 12:09:14	550	AC
48	5	5	2	Pokhara	Kathmandu	13	2021-08-16	2021-08-20 12:09:15	550	AC
49	6	6	6	Kathmandu	Jhapa	15	2021-08-17	2021-08-20 12:09:15	1200	Deluxe
50	7	5	2	Jhapa	Kathmandu	17	2021-08-15	2021-08-20 12:09:15	1100	AC
51	8	4	2	Kathmandu	Pokhara	19	2021-08-15	2021-08-20 12:09:15	550	AC
52	9	7	2	Kathmandu	Pokhara	21	2021-08-15	2021-08-20 12:09:15	550	AC
53	10	8	4	Kathmandu	Pokhara	23	2021-08-15	2021-08-20 12:09:15	500	Non AC
54	11	5	2	Kathmandu	Pokhara	25	2021-08-16	2021-08-20 12:09:15	550	AC
55	12	9	4	Kathmandu	Jhapa	27	2021-08-15	2021-08-20 12:09:15	1000	Non AC
56	13	5	7	Kathmandu	Pokhara	29	2021-08-17	2021-08-20 12:09:15	550	AC

vi. Total ticket sales for bus/day

bus_id	bus_name	from_city	to_city	No of tickets	no_of_seats	date_of_journey
2	BlueSky Bus	Kathmandu	Chitwan	1	40	2021-08-15
2	BlueSky Bus	Kathmandu	Chitwan	1	40	2021-08-16
2	BlueSky Bus	Kathmandu	Pokhara	3	40	2021-08-15
2	BlueSky Bus	Kathmandu	Pokhara	1	40	2021-08-16
2	BlueSky Bus	Chitwan	Kathmandu	1	40	2021-08-17
2	BlueSky Bus	Jhapa	Kathmandu	1	40	2021-08-15
2	BlueSky Bus	Pokhara	Kathmandu	1	40	2021-08-16
4	BlueSky Bus	Kathmandu	Jhapa	1	40	2021-08-15
4	BlueSky Bus	Kathmandu	Pokhara	1	40	2021-08-15
6	BlueSky Bus	Kathmandu	Jhapa	1	40	2021-08-17
7	Sunrise Bus	Kathmandu	Pokhara	1	40	2021-08-17

vii. Bus drivers

driver_id	name	contact_no
1	Ram Khadka	9876546541
2	Sworup Jha	9877654321
3	Manish Shrestha	9876546563
4	Harish Bhattrai	9876546754
5	Sagar Karki	9877663344
6	Jenis Dahal	9841564738
7	Rohit Rauniyar	9851140001
8	Ming Sherpa	9811223344
9	Anup Bam	9876354662
10	Samir Khadka	9876546633
11	Ashish Gurung	9876546644
12	Sudip Neupane	9818367942
13	Sanskar Pradhan	9869416158
14	Bipin Poudel	9818267565
15	Baibhav Sharma	9818625576
16	Ashok Puri	9818265767
17	Sashikanth Jha	9879879874
18	Ansal Tamang	9841587879
20	driver 10	9874563214

viii. Reserved seats

bus_id	seat_number	person_id	date_of_journey
2	5	1	2021-08-15
2	7	2	2021-08-16
2	9	3	2021-08-17
2	11	4	2021-08-15
2	13	5	2021-08-16
6	15	6	2021-08-17
2	17	5	2021-08-15
2	19	4	2021-08-15
2	21	7	2021-08-15
4	23	8	2021-08-15
2	25	5	2021-08-16
4	27	9	2021-08-15
7	29	5	2021-08-17

ix. Bus Schedule

schedule_id	bus_id	departs_at	arrives_at	driver_id	depart_date
1	6	07:15:00	03:00:00	2	2021-08-18
2	1	08:30:00	15:00:00	18	2021-08-20
3	4	07:15:00	03:00:00	9	2021-08-18
13	2	08:30:00	18:00:00	3	2021-08-23
17	3	20:00:00	04:00:00	4	2021-08-22
18	5	22:00:00	05:00:00	10	2021-08-21
19	8	06:00:00	12:00:00	12	2021-08-18

x. Bus schedule Backup table (data are added once update are made to items in bus schedules)

driver_id	bus_id	departs_at	arrives_at

CHAPTER 5: CONCLUSION AND FURTHER WORKS

5.1 Conclusion

To sum up, the project was completed with the coordination of our team and help from various sources and under the supervision of our great teacher. This project took us about 5 weeks to complete and be fully working.

5.2 Further Works

This project is not perfect as it has various aspects of improvement that can be done to meet the needs as required. Some of the areas might be as follows:

- Use for frontend for GUI interface
- Whole bus booking for special occasions
- Adding more routes and buses
- Better management of data
- Calculations of profit and loss from bus expenses

REFERENCES

[1] Tutorialspoint.com. 2021. SQL Tutorial - Tutorialspoint. [online] Available at: [Accessed 10 July 2021].

[2] "MySQL - Wikipedia", En.wikipedia.org, 2021. [Online]. Available: https://en.wikipedia.org/wiki/MySQL. [Accessed: 12- Jul- 2021]

[3] "MySQL Tutorial", W3schools.com, 2021. [Online]. Available: https://www.w3schools.com/mysql/. [Accessed: 14- Jul- 2021]

[4] SQL Tutorial – freecodecamp.org

[5] https://youtu.be/ZBgXb66Ckz0

Code For MySQL File:

```
CREATE TABLE `bus_driver` (
  `driver_id` int(5) NOT NULL,
  `name` varchar(30) DEFAULT NULL,
  `contact_no` bigint(10) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `bus_driver` (`driver_id`, `name`, `contact_no`) VALUES
(1, 'Ram Khadka', 9876546541),
(2, 'Sworup Jha', 9877654321),
(3, 'Manish Shrestha', 9876546563),
(4, 'Harish Bhattrai', 9876546754),
(5, 'Sagar Karki', 9877663344),
(6, 'Jenis Dahal', 9841564738),
(7, 'Rohit Rauniyar', 9851140001),
(8, 'Ming Sherpa', 9811223344),
(9, 'Anup Bam', 9876354662),
(10, 'Samir Khadka', 9876546633),
(11, 'Ashish Gurung', 9876546644),
(12, 'Sudip Neupane', 9818367942),
(13, 'Sanskar Pradhan', 9869416158),
(14, 'Bipin Poudel', 9818267565),
(15, 'Baibhav Sharma', 9818625576),
(16, 'Ashok Puri', 9818265767),
(17, 'Sashikanth Jha', 9879879874),
(18, 'Ansal Tamang', 9841587879),
```

```
(20, 'driver 10', 9874563214);


CREATE TABLE `bus_info` (
  `bus_name` varchar(20) DEFAULT NULL,
  `bus_number` varchar(20) NOT NULL,
  `bus_type` enum('AC','Non AC','Deluxe') DEFAULT NULL,
  `no_of_seats` int(2) NOT NULL DEFAULT 40
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `bus_info` (`bus_name`, `bus_number`, `bus_type`, `no_of_seats`) VALUES
('Makalu bus', '1', 'AC', 40),
('Makalu bus', '10', 'AC', 40),
('BlueSky Bus', '2', 'AC', 40),
('Makalu bus', '3', 'Non AC', 40),
('BlueSky Bus', '4', 'Non AC', 40),
('Makalu Bus', '5', 'Deluxe', 40),
('BlueSky Bus', '6', 'Deluxe', 40),
('Sunrise Bus', '7', 'AC', 40),
('Sunrise Bus', '8', 'Non AC', 40),
('Sunrise Bus', '9', 'Deluxe', 40);

CREATE TABLE `bus_schedule` (
  `schedule_id` int(11) NOT NULL,
  `bus_id` varchar(20) NOT NULL,
  `departs_at` time DEFAULT NULL,
  `arrives_at` time DEFAULT NULL,
  `driver_id` int(5) DEFAULT NULL,
  `depart_date` date DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `bus_schedule` (`schedule_id`, `bus_id`, `departs_at`, `arrives_at`, `driver_id`, `depart_date`) VALUES
(1, '6', '07:15:00', '03:00:00', 2, '2021-08-18'),
(2, '1', '08:30:00', '15:00:00', 18, '2021-08-20'),
(3, '4', '07:15:00', '03:00:00', 9, '2021-08-18'),
(13, '2', '08:30:00', '18:00:00', 3, '2021-08-23'),
(17, '3', '20:00:00', '04:00:00', 4, '2021-08-22'),
(18, '5', '22:00:00', '05:00:00', 10, '2021-08-21'),
(19, '8', '06:00:00', '12:00:00', 12, '2021-08-18');

DELIMITER $$
CREATE TRIGGER `copy_busschedule` BEFORE UPDATE ON `bus_schedule` FOR EACH ROW begin
insert into bus_schedule_backup(driver_id,bus_id,departs_at,arrives_at) values (old.driver_id,old.bus_id,old.departs_at, old.arrives_at);
end
$$
DELIMITER ;
```

```
CREATE TABLE `bus_schedule_backup` (
  `driver_id` int(5) DEFAULT NULL,
  `bus_id` varchar(20) DEFAULT NULL,
  `departs_at` time DEFAULT NULL,
  `arrives_at` time DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `count_ticket_occupancy` (
`bus_id` varchar(20)
,`bus_name` varchar(20)
,`from_city` enum('Kathmandu','Chitwan','Jhapa','Pokhara')
,`to_city` enum('Kathmandu','Chitwan','Jhapa','Pokhara')
,`No of tickets` bigint(21)
,`no_of_seats` int(2)
,`date_of_journey` date
);

CREATE TABLE `driver_of_bus` (
`driver_id` int(5)
,`name` varchar(30)
,`contact_no` bigint(10)
,`bus_name` varchar(20)
,`bus_number` varchar(20)
,`bus_type` enum('AC','Non AC','Deluxe')
,`no_of_seats` int(2)
);

CREATE TABLE `passenger` (
  `p_id` int(6) NOT NULL,
  `p_name` varchar(30) DEFAULT NULL,
  `gender` enum('Male','Female','Others') DEFAULT NULL,
  `contact_no` varchar(10) DEFAULT NULL,
  `address` varchar(30) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;--

INSERT INTO `passenger` (`p_id`, `p_name`, `gender`, `contact_no`, `address`)
VALUES
(1, 'Sushil Rai', 'Male', '987654654', 'baneshwor'),
(2, 'Riwaj Neupane', 'Male', '9899898897', 'Koteshwor'),
(3, 'Samit Baral', 'Male', '987212354', 'Bagbazar'),
(4, 'Babu Chiri Sherpa', 'Female', '987474147', 'Lokanthali'),
(5, 'Ang Rita', 'Female', '9879879877', 'abcxyz'),
(6, 'Apa Sherpa', 'Others', '9876546542', 'Chabel'),
(7, 'Basanta Regmi', 'Male', '9879876545', 'Naikap'),
(8, 'Paras Khadka', 'Male', '9877897897', 'Kalimati'),
(9, 'Dayaram Dahal', 'Female', '9877878787', 'Teku'),
(10, 'Atul Gautam', 'Female', '987879877', 'Tripureshwor'),
(11, 'Harsha Bahadur ', 'Male', '987654321', 'kathmandu'),
(12, 'Riwaj Neupane', 'Male', '9869416157', 'Chabhail'),
```

```
(13, 'Samit Baral', 'Male', '123456789', 'Bagbazar'),
(14, 'Ram Sharma', 'Male', '9818367943', 'Lokanthali'),
(15, 'Hai Kandel', 'Male', '9841411648', 'Thapathali'),
(16, 'Sita Magar', 'Female', '981234567', 'Jhapa'),
(17, 'Laxmi Aryal', 'Female', '9870665544', 'Kaski'),
(18, 'Maya Lama ', 'Female', '9811223344', 'Banepa'),
(19, 'Manoj Khadka', 'Male', '9855667711', 'Lamjung'),
(20, 'Raju Ray', 'Male', '9871234778', 'Biratnagar');

CREATE TABLE `reserved_seats` (
  `bus_id` varchar(20) DEFAULT NULL,
  `seat_number` int(2) DEFAULT NULL,
  `person_id` int(11) DEFAULT NULL,
  `date_of_journey` date DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `reserved_seats` (`bus_id`, `seat_number`, `person_id`,
`date_of_journey`) VALUES
('2', 5, 1, '2021-08-15'),
('2', 7, 2, '2021-08-16'),
('2', 9, 3, '2021-08-17'),
('2', 11, 4, '2021-08-15'),
('2', 13, 5, '2021-08-16'),
('6', 15, 6, '2021-08-17'),
('2', 17, 5, '2021-08-15'),
('2', 19, 4, '2021-08-15'),
('2', 21, 7, '2021-08-15'),
('4', 23, 8, '2021-08-15'),
('2', 25, 5, '2021-08-16'),
('4', 27, 9, '2021-08-15'),
('7', 29, 5, '2021-08-17');

CREATE TABLE `route` (
  `Route_id` int(20) NOT NULL,
  `from_city` varchar(20) DEFAULT NULL,
  `destination` varchar(20) DEFAULT NULL,
  `bus_id` varchar(20) DEFAULT NULL,
  `bus_type` varchar(10) DEFAULT NULL,
  `fare` int(5) DEFAULT NULL,
  `depart_date` date DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `route` (`Route_id`, `from_city`, `destination`, `bus_id`, `bus_type`,
`fare`, `depart_date`) VALUES
(1, 'Kathmandu', 'Chitwan', '2', 'AC', 825, '2021-08-15'),
(1, 'Kathmandu', 'Chitwan', '2', 'AC', 825, '2021-08-16'),
(1, 'Chitwan', 'Kathmandu', '2', 'AC', 825, '2021-08-17'),
(2, 'Kathmandu', 'Pokhara', '2', 'AC', 550, '2021-08-15'),
(2, 'Pokhara', 'Kathmandu', '2', 'AC', 550, '2021-08-16'),
(3, 'Kathmandu', 'Jhapa', '6', 'Deluxe', 1200, '2021-08-17'),
```

```
(3, 'Jhapa', 'Kathmandu', '2', 'AC ', 1100, '2021-08-15'),
(2, 'Kathmandu', 'Pokhara', '2', 'AC', 550, '2021-08-15'),
(2, 'Kathmandu', 'Pokhara', '2', 'AC', 550, '2021-08-15'),
(2, 'Kathmandu', 'Pokhara', '4', 'Non AC', 500, '2021-08-15'),
(2, 'Kathmandu', 'Pokhara', '2', 'AC', 550, '2021-08-16'),
(3, 'Kathmandu', 'Jhapa', '4', 'Non AC', 1000, '2021-08-15'),
(2, 'Kathmandu', 'Pokhara', '7', 'AC', 550, '2021-08-17');

CREATE TABLE `ticket` (
  `p_id` int(6) DEFAULT NULL,
  `ticket_no` int(6) NOT NULL,
  `bus_id` varchar(20) DEFAULT NULL,
  `from_city` enum('Kathmandu','Chitwan','Jhapa','Pokhara') NOT NULL,
  `to_city` enum('Kathmandu','Chitwan','Jhapa','Pokhara') NOT NULL,
  `seat_no` int(2) NOT NULL,
  `date_of_journey` date DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

INSERT INTO `ticket` (`p_id`, `ticket_no`, `bus_id`, `from_city`, `to_city`, `seat_no`,
`date_of_journey`) VALUES
(1, 1, '2', 'Kathmandu', 'Chitwan', 5, '2021-08-15'),
(2, 2, '2', 'Kathmandu', 'Chitwan', 7, '2021-08-16'),
(3, 3, '2', 'Chitwan', 'Kathmandu', 9, '2021-08-17'),
(4, 4, '2', 'Kathmandu', 'Pokhara', 11, '2021-08-15'),
(5, 5, '2', 'Pokhara', 'Kathmandu', 13, '2021-08-16'),
(6, 6, '6', 'Kathmandu', 'Jhapa', 15, '2021-08-17'),
(5, 7, '2', 'Jhapa', 'Kathmandu', 17, '2021-08-15'),
(4, 8, '2', 'Kathmandu', 'Pokhara', 19, '2021-08-15'),
(7, 9, '2', 'Kathmandu', 'Pokhara', 21, '2021-08-15'),
(8, 10, '4', 'Kathmandu', 'Pokhara', 23, '2021-08-15'),
(5, 11, '2', 'Kathmandu', 'Pokhara', 25, '2021-08-16'),
(9, 12, '4', 'Kathmandu', 'Jhapa', 27, '2021-08-15'),
(5, 13, '7', 'Kathmandu', 'Pokhara', 29, '2021-08-17');

DELIMITER $$
CREATE TRIGGER `bus_fare` AFTER INSERT ON `ticket` FOR EACH ROW
BEGIN
UPDATE booking_info SET booking_info.bus_type="AC"  WHERE bus_id='1' or
bus_id=2 or bus_id=7;
UPDATE booking_info SET booking_info.bus_type="Non AC"  WHERE bus_id=3
or bus_id=4 or bus_id=8;
UPDATE booking_info SET booking_info.bus_type="Deluxe"  WHERE bus_id=5 or
bus_id=6 or bus_id=9;

UPDATE booking_info SET booking_info.fare=550 WHERE
(booking_info.from_city="Kathmandu" AND booking_info.to_city="Pokhara" AND
booking_info.bus_type="AC");
UPDATE booking_info SET booking_info.fare=825 WHERE
(booking_info.from_city="Chitwan" AND booking_info.to_city="Kathmandu" AND
booking_info.bus_type="AC");
```

```
UPDATE booking_info SET booking_info.fare=1100 WHERE (booking_info.from_city="Jhapa" AND booking_info.to_city="Kathmandu" AND booking_info.bus_type="AC");
UPDATE booking_info SET booking_info.fare=550 WHERE (booking_info.from_city="Pokhara" AND booking_info.to_city="Kathmandu" AND booking_info.bus_type="AC");
UPDATE booking_info SET booking_info.fare=1100 WHERE (booking_info.from_city="Kathmandu" AND booking_info.to_city="Jhapa" AND booking_info.bus_type="AC");
UPDATE booking_info SET booking_info.fare=825 WHERE (booking_info.from_city="Kathmandu" AND booking_info.to_city="Chitwan" AND booking_info.bus_type="AC");

UPDATE booking_info SET booking_info.fare=1200 WHERE (booking_info.from_city="Jhapa" AND booking_info.to_city="Kathmandu" AND booking_info.bus_type="Deluxe");
UPDATE booking_info SET booking_info.fare=600 WHERE (booking_info.from_city="Kathmandu" AND booking_info.to_city="Pokhara" AND booking_info.bus_type="Deluxe");
UPDATE booking_info SET booking_info.fare=600 WHERE (booking_info.from_city="Pokhara" AND booking_info.to_city="Kathmandu" AND booking_info.bus_type="Deluxe");
UPDATE booking_info SET booking_info.fare=900 WHERE (booking_info.from_city="Chitwan" AND booking_info.to_city="Kathmandu" AND booking_info.bus_type="Deluxe");
UPDATE booking_info SET booking_info.fare=900 WHERE (booking_info.from_city="Kathmandu" AND booking_info.to_city="Chitwan" AND booking_info.bus_type="Deluxe");
UPDATE booking_info SET booking_info.fare=1200 WHERE (booking_info.from_city="Kathmandu" AND booking_info.to_city="Jhapa" AND booking_info.bus_type="Deluxe");

UPDATE booking_info SET booking_info.fare=500 WHERE (booking_info.from_city="Kathmandu" AND booking_info.to_city="Pokhara" AND booking_info.bus_type="Non AC");
UPDATE booking_info SET booking_info.fare=500 WHERE (booking_info.from_city="Pokhara" AND booking_info.to_city="Kathmandu" AND booking_info.bus_type="Non AC");
UPDATE booking_info SET booking_info.fare=1000 WHERE (booking_info.from_city="Jhapa" AND booking_info.to_city="Kathmandu" AND booking_info.bus_type="Non AC");
UPDATE booking_info SET booking_info.fare=1000 WHERE (booking_info.from_city="Kathmandu" AND booking_info.to_city="Jhapa" AND booking_info.bus_type="Non AC");
UPDATE booking_info SET booking_info.fare=750 WHERE (booking_info.from_city="Chitwan" AND booking_info.to_city="Kathmandu" AND booking_info.bus_type="Non AC");
UPDATE booking_info SET booking_info.fare=750 WHERE (booking_info.from_city="Kathmandu" AND booking_info.to_city="Chitwan" AND booking_info.bus_type="Non AC");
```

```
END
$$
DELIMITER ;
DELIMITER $$
CREATE TRIGGER `test` AFTER INSERT ON `ticket` FOR EACH ROW begin
insert into booking_info(p_id,ticket_no, bus_id,from_city,to_city,seat_no,date_of_journey) values (new.p_id,new.ticket_no, new.bus_id,new.from_city,new.to_city,new.seat_no,new.date_of_journey);
END
$$
DELIMITER ;
DROP TABLE IF EXISTS `count_ticket_occupancy`;

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY DEFINER VIEW `count_ticket_occupancy` AS SELECT `t`.`bus_id` AS `bus_id`, `b`.`bus_name` AS `bus_name`, `t`.`from_city` AS `from_city`, `t`.`to_city` AS `to_city`, count(`t`.`ticket_no`) AS `No of tickets`, `b`.`no_of_seats` AS `no_of_seats`, `t`.`date_of_journey` AS `date_of_journey` FROM (`ticket` `t` join `bus_info` `b` on(`t`.`bus_id` = `b`.`bus_number`)) GROUP BY `t`.`bus_id`, `t`.`from_city`, `t`.`to_city`, `t`.`date_of_journey` ORDER BY `t`.`bus_id` ASC ;

DROP TABLE IF EXISTS `driver_of_bus`;

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY DEFINER VIEW `driver_of_bus` AS SELECT `bd`.`driver_id` AS `driver_id`, `bd`.`name` AS `name`, `bd`.`contact_no` AS `contact_no`, `b`.`bus_name` AS `bus_name`, `b`.`bus_number` AS `bus_number`, `b`.`bus_type` AS `bus_type`, `b`.`no_of_seats` AS `no_of_seats` FROM ((`bus_schedule` `bs` join `bus_info` `b` on(`b`.`bus_number` = `bs`.`bus_id`)) join `bus_driver` `bd` on(`bd`.`driver_id` = `bs`.`driver_id`)) ORDER BY `bd`.`driver_id` ASC ;

ALTER TABLE `booking_info`
  ADD PRIMARY KEY (`booking_id`),
  ADD KEY `p_id` (`p_id`),
  ADD KEY `bus_id` (`bus_id`),
  ADD KEY `ticket_no` (`ticket_no`);

ALTER TABLE `bus_driver`
  ADD PRIMARY KEY (`driver_id`);

ALTER TABLE `bus_info`
  ADD PRIMARY KEY (`bus_number`);

ALTER TABLE `bus_schedule`
  ADD PRIMARY KEY (`schedule_id`),
  ADD UNIQUE KEY `driver_id` (`driver_id`),
  ADD KEY `bus_id` (`bus_id`);

ALTER TABLE `bus_schedule_backup`
```

```
  ADD KEY `bus_id` (`bus_id`);

ALTER TABLE `passenger`
  ADD PRIMARY KEY (`p_id`),
  ADD UNIQUE KEY `contact_no` (`contact_no`);

ALTER TABLE `route`
  ADD KEY `bus_id` (`bus_id`);

ALTER TABLE `ticket`
  ADD PRIMARY KEY (`ticket_no`),
  ADD KEY `p_id` (`p_id`),
  ADD KEY `bus_id` (`bus_id`);

ALTER TABLE `booking_info`
       MODIFY `booking_id` int(5) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=57;

ALTER TABLE `bus_schedule`
      MODIFY `schedule_id` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=20;

ALTER TABLE `passenger`
         MODIFY `p_id` int(6) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=21;

ALTER TABLE `ticket`
       MODIFY `ticket_no` int(6) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=14;

ALTER TABLE `booking_info`
    ADD CONSTRAINT `booking_info_ibfk_1` FOREIGN KEY (`ticket_no`)
REFERENCES `ticket` (`ticket_no`),
     ADD CONSTRAINT `booking_info_ibfk_2` FOREIGN KEY (`p_id`)
REFERENCES `passenger` (`p_id`),
    ADD CONSTRAINT `booking_info_ibfk_3` FOREIGN KEY (`bus_id`)
REFERENCES `bus_info` (`bus_number`),
    ADD CONSTRAINT `booking_info_ibfk_4` FOREIGN KEY (`ticket_no`)
REFERENCES `ticket` (`ticket_no`),
     ADD CONSTRAINT `booking_info_ibfk_5` FOREIGN KEY (`fare`)
REFERENCES `fare` (`fare`),
     ADD CONSTRAINT `booking_info_ibfk_6` FOREIGN KEY (`fare`)
REFERENCES `fare` (`fare`);

ALTER TABLE `bus_schedule`
    ADD CONSTRAINT `bus_schedule_ibfk_1` FOREIGN KEY (`bus_id`)
REFERENCES `bus_info` (`bus_number`),
    ADD CONSTRAINT `bus_schedule_ibfk_2` FOREIGN KEY (`driver_id`)
REFERENCES `bus_driver` (`driver_id`);
```

```
ALTER TABLE `bus_schedule_backup`
  ADD CONSTRAINT `bus_schedule_backup_ibfk_1` FOREIGN KEY (`bus_id`) REFERENCES `bus_schedule` (`bus_id`);

ALTER TABLE `route`
  ADD CONSTRAINT `route_ibfk_1` FOREIGN KEY (`bus_id`) REFERENCES `bus_info` (`bus_number`);

ALTER TABLE `ticket`
  ADD CONSTRAINT `ticket_ibfk_1` FOREIGN KEY (`p_id`) REFERENCES `passenger` (`p_id`),
  ADD CONSTRAINT `ticket_ibfk_2` FOREIGN KEY (`bus_id`) REFERENCES `bus_info` (`bus_number`);
COMMIT;
```