



AIRPLANE RESERVATION SYSTEM SCENARIO



SAIFEDEEN KHAROUF

19110044

Table of Contents

Introduction.....	2
1. System scenario:	2
2. User groups.....	3
Database design.....	4
a. Conceptual model	4
b. Logical design:.....	4
a. Schema.....	4
b. Normalization.....	4
c. Mapping.....	5
d. Final logical design.....	6
3. Physical design	6
Technical documentation.....	7
a. Flowchart and data movement diagram	7
b. Installation and using of the xampp and the phpmyadmin.....	8
c. My designs and the scenario are linked here	11
d. DDL (tables and views and procedures)	11
e. DML	14
f. DCL / security	18
g. Database maintenance.....	18
h. GUI development	19
User documentation	32
Testing.....	35
a. Data validation	35
b. Output validation	35
c. Security validation.....	36
d. GUI validation.....	37
Assessment.....	39
a. Data assessment	39
b. Effectiveness of the design	39
c. Recommendations for the improvements of the system.....	39
References.....	40
Declaration	Error! Bookmark not defined.
Plagiarism	Error! Bookmark not defined.

Introduction

1. System scenario:

The united airlines is an office that provides many services for customers wants to reserve a fly in a certain airline that they like to travel internationally. This website will have also employees to work on it. The database will have the passengers, employees, airlines data.

Although this an office, it has also a website that accessed through an URL. This website main objective to show an interface to the passengers to see their destinations and available seats in their suitable airplane and reserve them. Also, they can see the ticket and print it to show it to the employees in the airport.

The passenger can reserve the flights, also he can reserve at least one seat, and they can choose multiple destinations, also they choose any airline to reserve on. And the flight can consist of one airline and only one but the airline can be in more than one flight.

The employees can enter data to the website, like the destinations and the times and the seats available, this will also be corporation with the airlines employees to help with the entry of the data.. And, they can modify the passengers schedule when they miss the flight, also if they needed to cancel the flight can help with the refunding the money.

Considering the following requirements:

- Each costumer (passenger) should have date of birth, passport id, name (first and middle and last), their gender, nationality, phone number, email and password to login to the website.
- Each flight have a number and type, departure country and time and date and airport, arrival country and time and date and airport, seat number and type. The flight types will be economy and first class.
- Each reservation has a number and a price and a seat number and type. The seat types will be economy and first class.
- The passenger can reserve many flights.
- The flight has many reservations and vice versa.
- The airline have a unique id and a name. Each flight can have one airline but many airline on the flight.

2. User groups

1. Costumers (passengers):

- (a) Reserve a schedule to a certain destination and in certain airplane that he prefer. Ex: Ahmad wants to go to the USA he go through our website and choose an economy class seat in the midnight of the 9th of December with the Jordanian airlines.
- (b) Review the ticket information. Ex: in Ahmad's flight ticket it will which seat and destination and the time he reserved for.
- (c) Can create an account and sign in to the website.
- (d) View the available flights and their details.

2. Employees:

- (a) Update the system with a newer destinations or schedule or airline or any combination of them. Ex: employee Tamer wanted to update the system that there is a new destination to Japan at the afternoon of the 30th of January with the Qatari airways.
- (b) Modify or cancel the reservation schedule of the customers. Ex: employee Kamal received Ahmad's request to change his schedule and he changed the schedule.

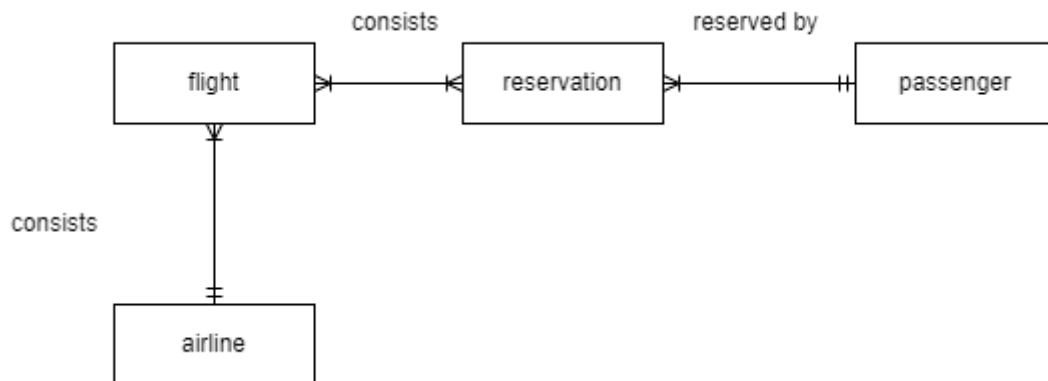
Database design

a. Conceptual model

(Data Modelling: Conceptual, Logical, Physical Data Model Types, no date)

Organized view of the concepts and relations of the database. Its main goal to create entities and relations between them. But there is minimum details in this design.

This diagram shows the entities and relations of my system. There is a relation between the flight and the reservation entities that the flight should consists of at least one reservation and can have many. Also a relation between the reservation and the passenger the reservation can be reserved by one and only one passenger. And another relation between the airline and the flight the airline that flight can be consist by one and only one airline.



b. Logical design:

a. Schema

Passenger (passport id, name, nationality, gender, birth date, phone number, password, email). The name is consists of first and middle and last name.

Flight (flight number, seat number, seat type, flight type, departure time, arrival time, departure country, arrival country ,departure airport, arrival airport)

Reservation (seat number, seat type, reservation number, price)

Airline (id, name)

b. Normalization

It is a design technique used to reduce the redundancy and removes any unwanted weird behaviours like the insertions, deletion, update. And it works by dividing the tables into a smaller tables and make relationships by linking them. The three main forms are:

- 1NF (first normal form)

In this stage we need to look for the multivalued and compound attributes

Passenger (passport id, name, nationality, gender, birth date, phone number, password, email) because the name has three types it should turn into this:

Passenger (passport id, first name, middle name, last name, nationality, gender, birth date, phone number, password, email)

- 2NF (second normal form)

In this stage it needs to be normalized in the first form and be fully functional dependent

*red colour: foreign key / blue colour: primary key

Passenger (passport id, first name, middle name, last name, nationality, gender, birth date, phone number, password, email)

Flight (flight number, airline id, seat number, seat type, flight type, departure time, arrival time, departure country, arrival country, departure airport, arrival airport)

Reservation (seat number, seat type, reservation number, price, passport id)

Airline (id, name)

- 3NF (third normal form)

It should be in the second normal form and transitive dependent

Passenger (passport id, first name, middle name, last name, nationality, gender, birth date, phone number, password, email)

Flight (flight number, airline id, flight type, departure time, arrival time, departure country, arrival country, departure airport, arrival airport)

Reservation (reservation number, price, passport id)

Airline (id, name)

Seats (seat number, seat type, reservation number, flight number) * the reservation and flight number are also foreign key

c. Mapping

Passenger (passport id, name, nationality, gender, birth date, phone number, password, email).

Flight (flight number, airline id, flight type, departure time, arrival time, departure country, arrival country, departure date, arrival date, departure airport, arrival airport)

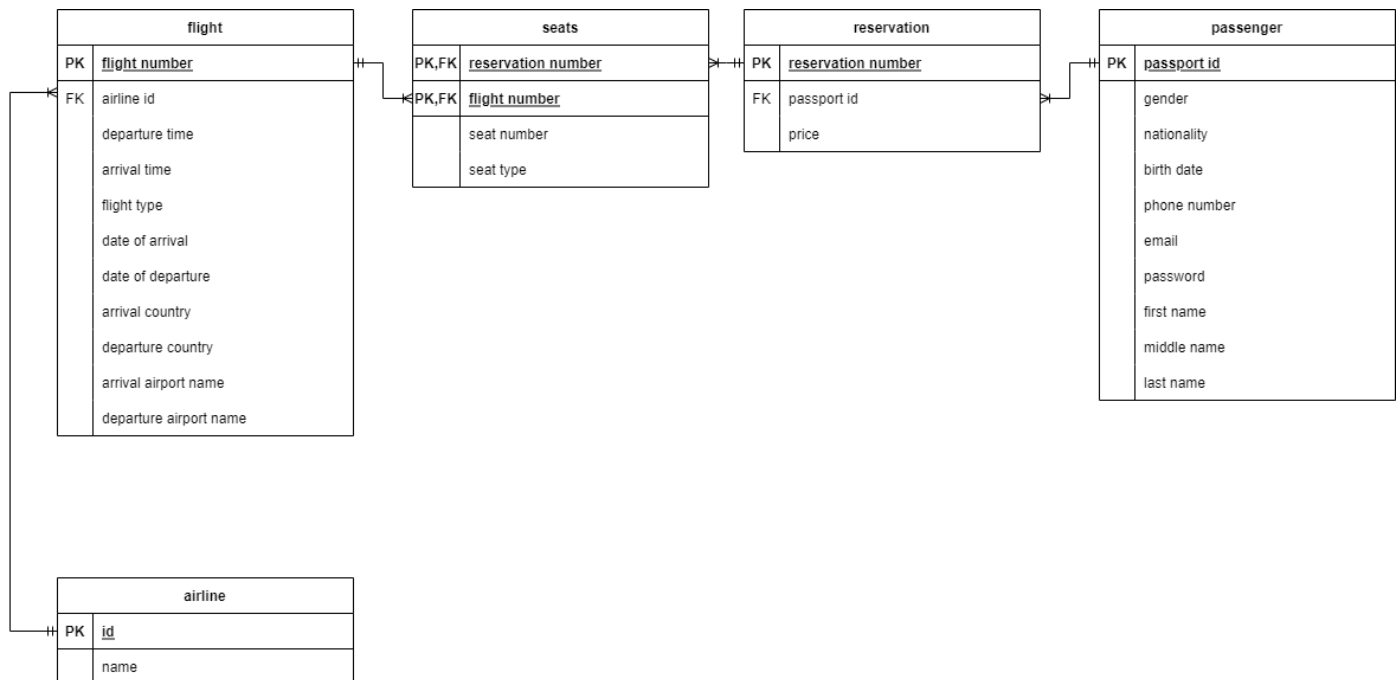
Reservation (reservation number, price, passport id)

Airline (id, name)

Seats (seat number, seat type, reservation number, flight number) * the reservation and flight number are also foreign key

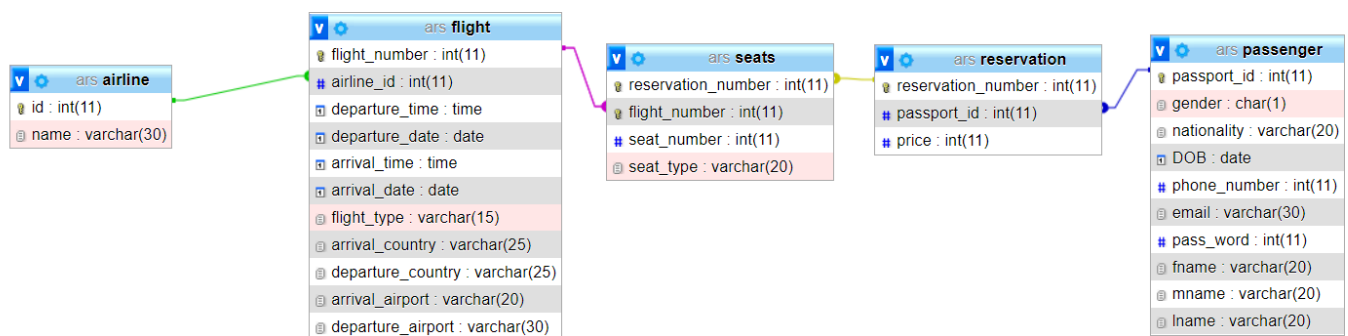
d. Final logical design

It adds a further information upon the [conceptual model](#) and it is the foundation based to the logical design. But still here it is in a generic level. (*Data Modelling: Conceptual, Logical, Physical Data Model Types*, no date)



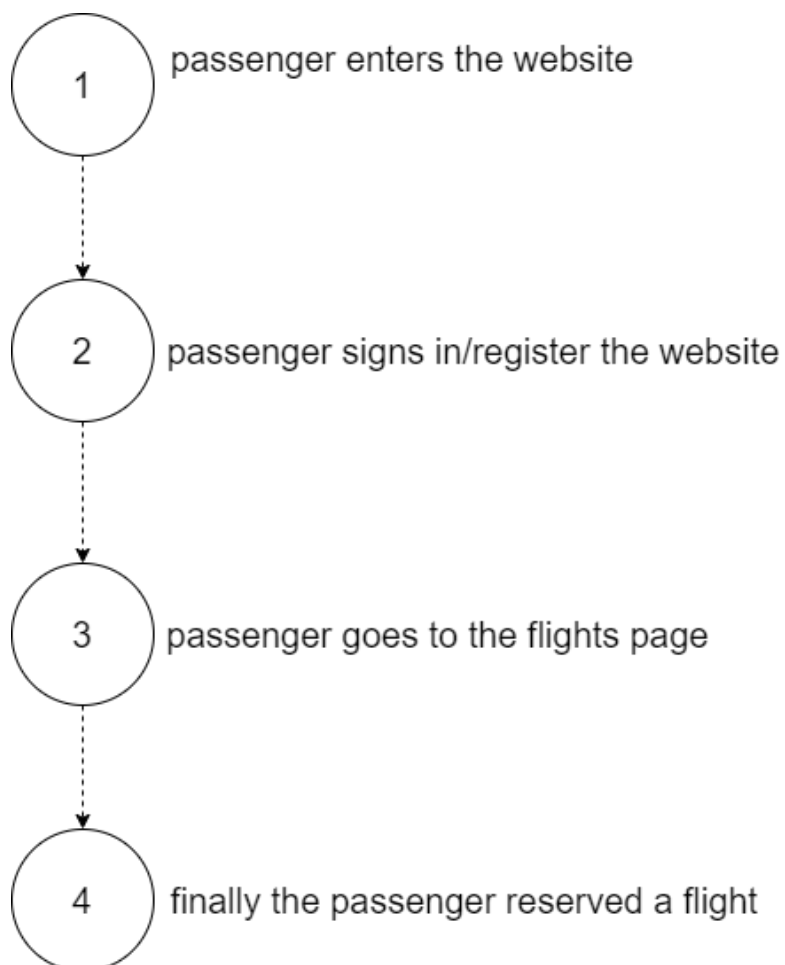
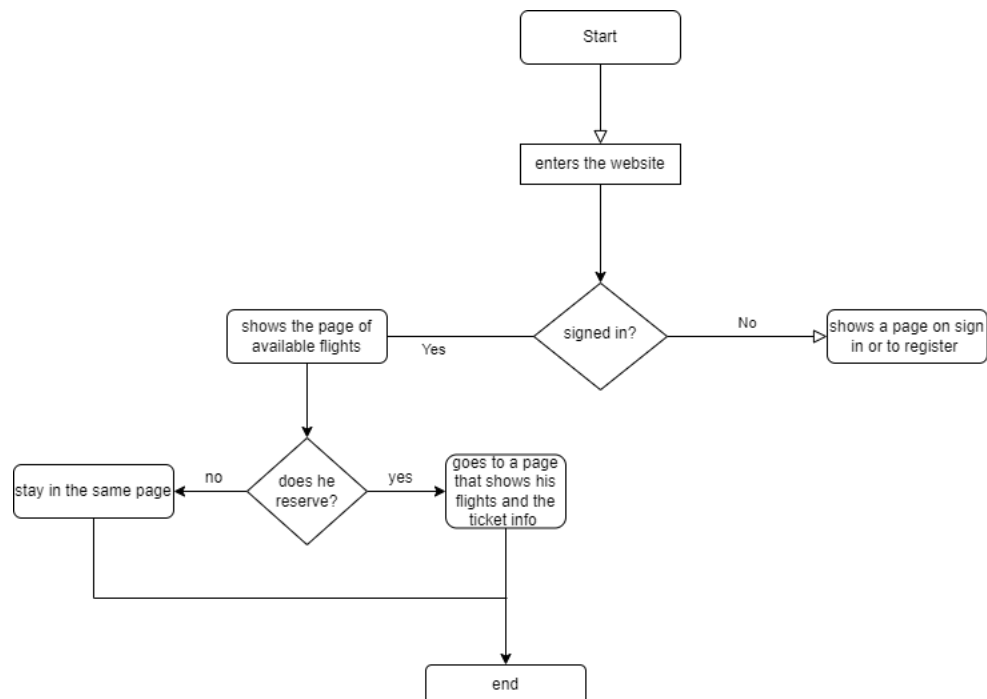
3. Physical design

It is the level where the specific implementation of the database. And it helps to generate a schema also it helps to visualize the structure of the database replicating database column keys, constraints, indexes, triggers, and other RDBMS features. (*Data Modelling: Conceptual, Logical, Physical Data Model Types*, no date)



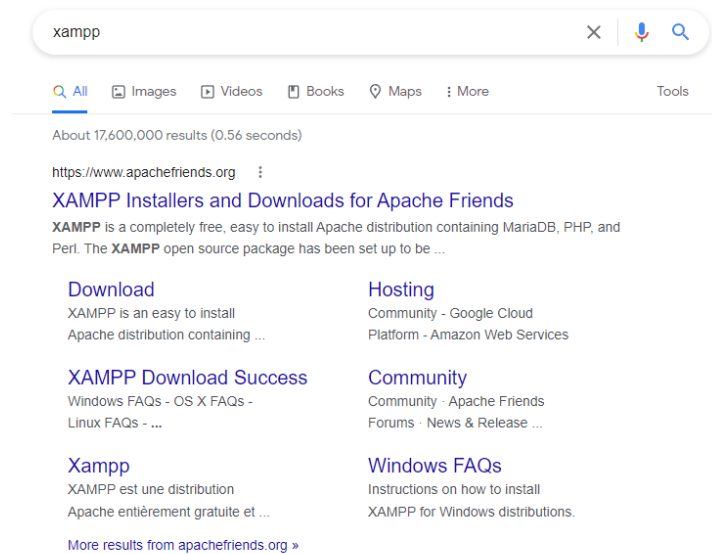
Technical documentation

a. Flowchart and data movement diagram



b. Installation and using of the xampp and the phpmyadmin

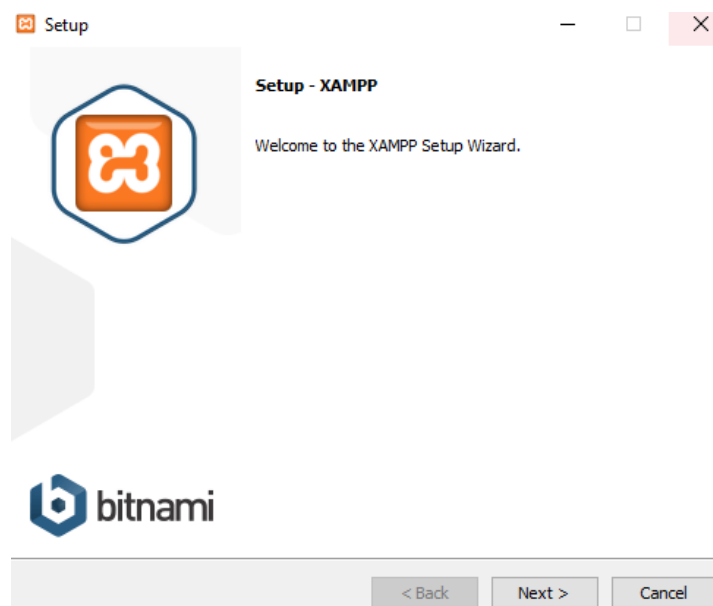
1) First step is to search for xampp on google and go to their website



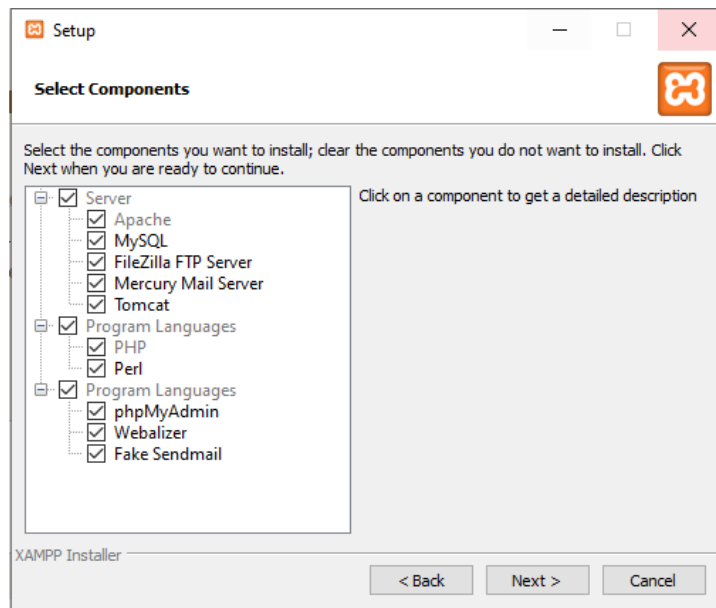
2) Second step is to choose your appropriate OS and download xampp



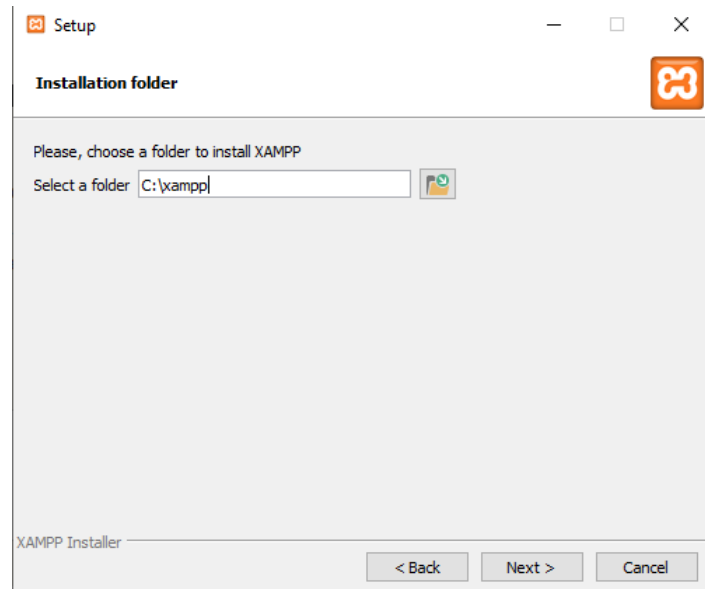
3) Third step is to click next



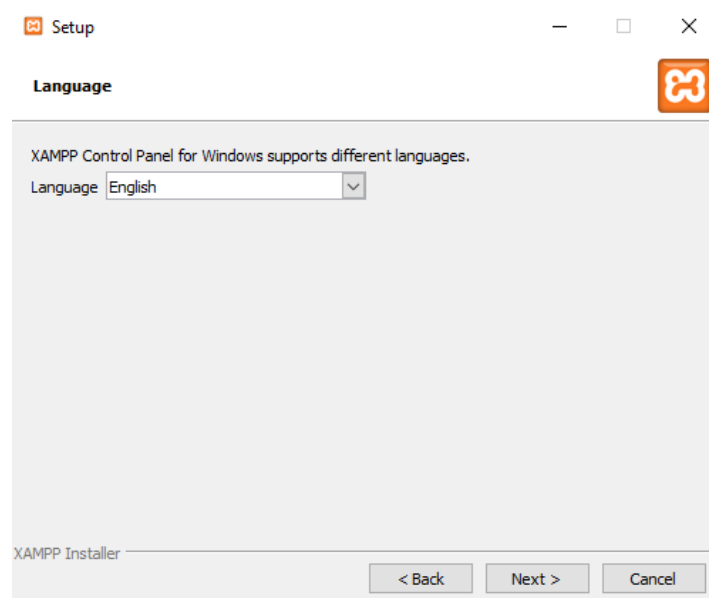
4) Fourth step is to choose them all as it is or choose your appropriate options



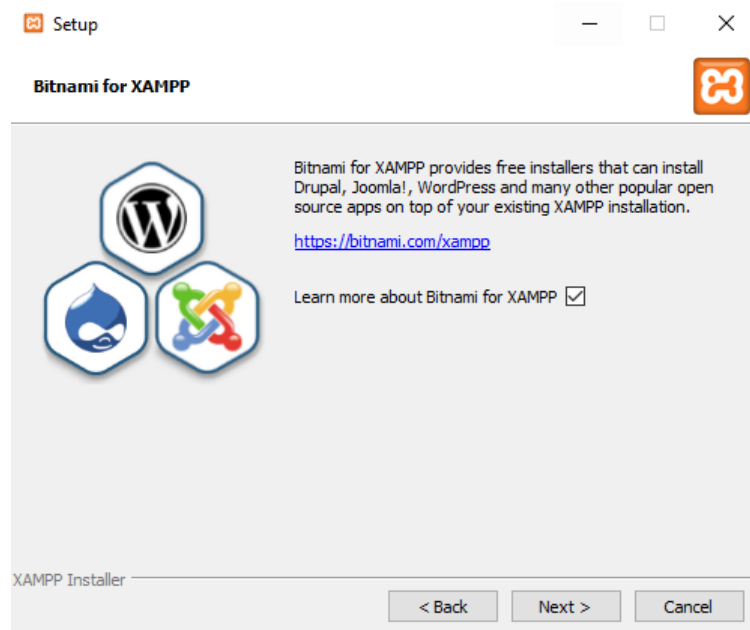
5) fifth step is to select the folder to go with (I recommend stay on the default folder)



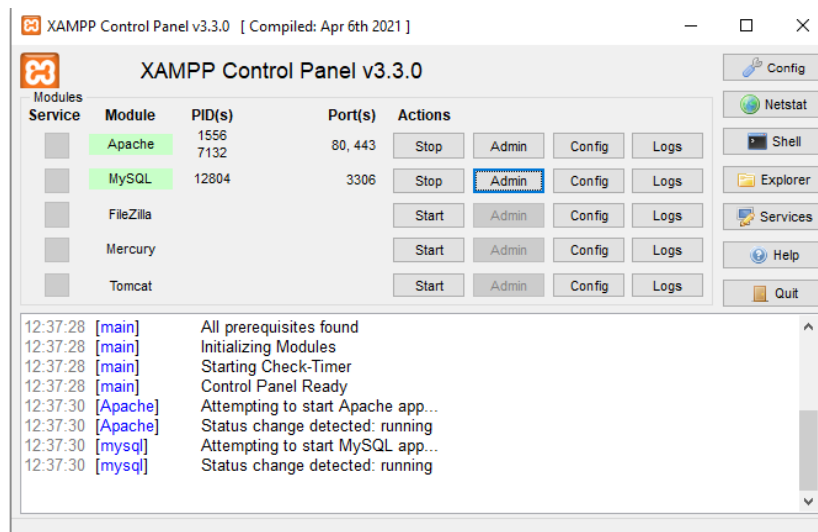
6) Sixth step is to select the language you want



7) Seventh step is to click next



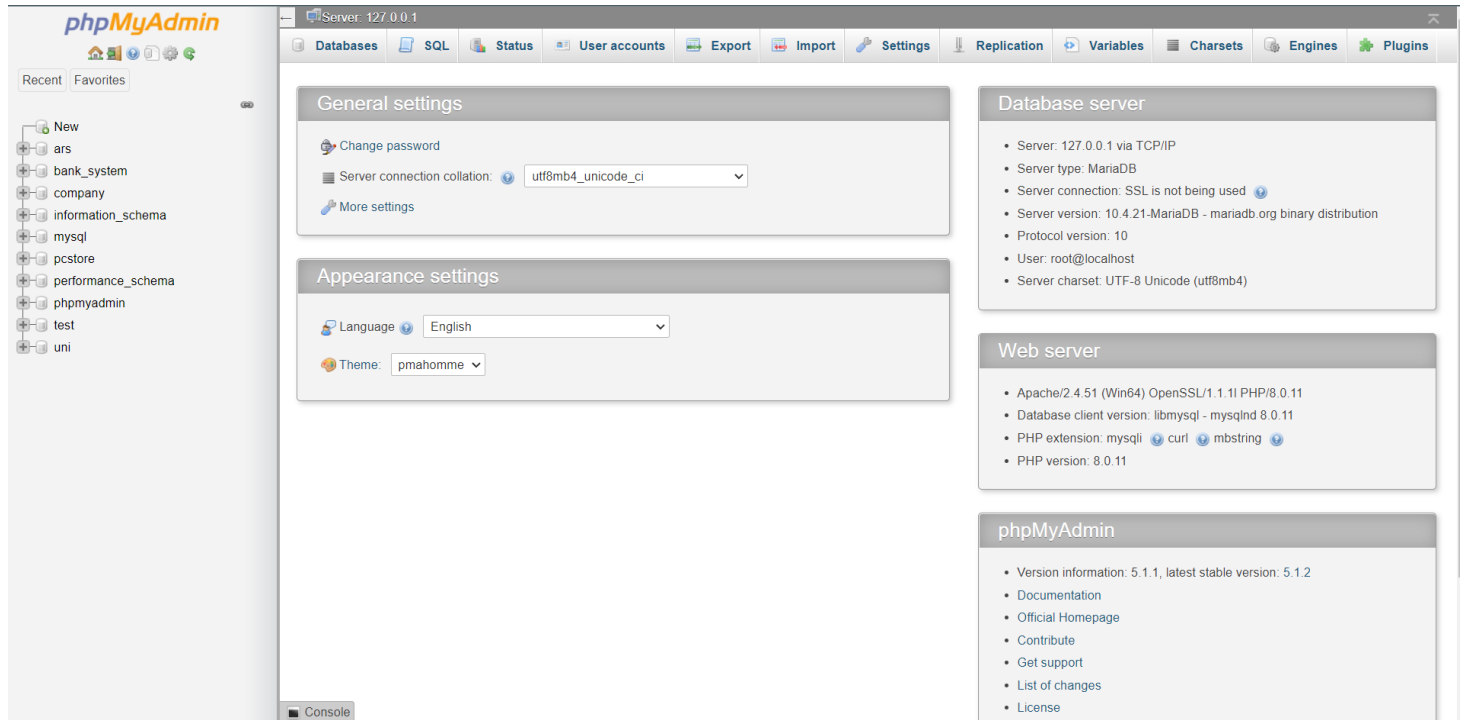
8) Eighth step when it finishes installation this window will appear. If you want to go to the phpmyadmin page start Apache and MySQL then click on admin on the MySQL.



9) Ninth and last step the phpmyadmin will appear in the browser as a new tab.

This window has these main components:

- Databases: the databases available.
- SQL: write queries in MySQL language.
- Export: exports the database as a SQL file.
- Import: imports a database.
- The tree on the left is also to show the databases available and in each one you can see the tables on this database.



c. My designs and the scenario are linked here

- [Scenario and user groups](#)
- [The design](#)

d. DDL (tables and views and procedures)

- Tables created:

```
CREATE TABLE flight (  
    flight_number INT,  
    airline_id INT,  
    departure_time TIME,  
    departure_date DATE,  
    arrival_time TIME,  
    arrival_date DATE,  
    flight_type VARCHAR(15),  
    arrival_country VARCHAR(25),  
    departure_country VARCHAR(25),
```

```
arrival_airport VARCHAR(20),  
departure_airport VARCHAR(30),  
PRIMARY KEY(flight_number),  
FOREIGN KEY(airline_id) REFERENCES airline(id) ON UPDATE  
CASCADE ON DELETE SET NULL  
)
```

```
CREATE TABLE passenger (  
    passport_id int,  
    gender CHAR,  
    nationality VARCHAR(20),  
    DOB date,  
    phone_number int,  
    email VARCHAR(30) NOT NULL,  
    password int NOT NULL,  
    fname VARCHAR(20) NOT NULL,  
    mname VARCHAR(20),  
    lname VARCHAR(20) NOT NULL  
)
```

```
ENGINE = INNODB;
```

```
ALTER TABLE passenger
```

```
ADD PRIMARY KEY (passport_id);
```

```
CREATE TABLE reservation (  
    reservation_number INT,
```

```
    passport_id INT,
```

```
    price INT,
```

```
    PRIMARY KEY(reservation_number),
```

```
    FOREIGN KEY(passport_id) REFERENCES passenger(passport_id) ON
```

```
UPDATE CASCADE ON DELETE SET NULL  
)
```

```
ENGINE = INNODB;
```

```

CREATE TABLE seats (
    reservation_number INT,
    flight_number INT,
    seat_number INT,
    seat_type VARCHAR(20),
    PRIMARY KEY(reservation_number, flight_number),
    FOREIGN KEY(reservation_number) REFERENCES
reservation(reservation_number) ON UPDATE CASCADE,
    FOREIGN KEY(flight_number) REFERENCES flight(flight_number) ON
UPDATE CASCADE
)
ENGINE = INNODB;

```

```

CREATE TABLE airline (
    id INT,
    name VARCHAR(30),
    PRIMARY KEY(id)
)
ENGINE = INNODB;

```

How I made the procedure (it got [DML](#) syntax)

```

DELIMITER //

CREATE PROCEDURE getflightinfo(IN passport_number INT)

BEGIN

SELECT DISTINCT f.*, a.name, s.seat_number,r.price

FROM seats AS s JOIN reservation AS r ON r.reservation_number =
s.reservation_number

JOIN flight AS f ON f.flight_number = s.flight_number JOIN airline AS a ON
a.id = f.airline_id

WHERE r.passport_id = passport_number;

END //

DELIMITER ;

```

How I made the view (it got [DML](#) syntax)

```
CREATE VIEW flights AS
```

```
SELECT DISTINCT f.*, a.name, s.seat_number,r.price
```

```
FROM seats AS s JOIN reservation AS r ON r.reservation_number =  
s.reservation_number
```

```
JOIN flight AS f ON f.flight_number = s.flight_number JOIN airline AS a ON  
a.id = f.airline_id;
```

e. DML

```
INSERT INTO reservation (reservation_number, passport_id, price) VALUES
```

```
(1234, 1888, 500), (3434, 1998, 200), (5234, 1928, 1000), (55887, 24440, 600), (789456, 27985, 300)
```

```
INSERT INTO seats (reservation_number, flight_number, seat_number, seat_type) VALUES
```

```
(1234, 123, 45, 'economy'), (3434, 153, 35, 'economy'), (5234, 154, 25, 'first class'), (55887, 234, 65, 'first-  
class'),
```

```
(789456, 334, 75, 'first-class');
```

```
INSERT INTO passenger (passport_id, gender, nationality, DOB, phone_number, email, pass_word, fname,  
mname, lname) VALUES
```

```
(1888, 'F', 'Jordanian', '2002-03-26', 7922223, 'lamis.smara@htu.edu.jo', 22345428, 'lamis', '', 'smara'),
```

```
(1928, 'M', 'Jordanian', '2001-04-09', 790290514, 'saifedeem@htu.edu.jo', 12345678, 'saifedeem',  
'mohammed', 'kharouf'),
```

```
(1998, 'F', 'Jordanian', '2002-08-13', 792222323, 'shahed.nwasieh@htu.edu.jo', 2229878, 'shahed', '',  
'nwasieh'),
```

```
(9997, 'F', 'jordanian', '2022-02-03', 2369987, '2222#@ffdfd', 222, 'saif', 'll', 'kharouf'),
```

```
(24440, 'M', 'jordanian', '2000-03-05', 7894500, 'hussein.tariq@htu.edu.com', 158755, 'hussien', 'tareq',  
'emad'),
```

```
(27985, 'M', 'jordanian', '2001-02-23', 79258211, 'mohammed.masharqa@htu.edu.com', 12555, 'mohammed',  
'saleh', 'masharqa'),
```

```
(222222, 'F', 'jordanian', '2021-12-30', 7902984, 'omar.jerab@htu.edu.jo', 5555522, 'omar', '', 'jerab');
```

INSERT INTO flight (flight_number, airline_id, departure_time, departure_date, arrival_time, arrival_date, flight_type, arrival_country, departure_country, arrival_airport, departure_airport) VALUES

(123, 2, '10:55:00', '2022-03-26', '22:00:00', '2022-04-09', 'economy', 'paris', 'amman', 'CDG', 'AMM'),

(153, 4, '12:00:00', '2022-04-10', '14:00:00', '2022-04-15', 'economy', 'Dubai', 'amman', 'DXB', 'AMM'),

(154, 5, '22:30:00', '2022-05-04', '01:30:00', '2022-05-15', 'economy', 'Doha', 'amman', 'DOH', 'AMM'),

(234, 5, '02:30:00', '2022-04-04', '06:30:00', '2022-05-15', 'first class', 'Jeddah', 'amman', 'JED', 'AMM'),

(334, 2, '13:30:00', '2022-04-05', '15:30:00', '2022-05-15', 'first class', 'Jeddah', 'amman', 'JED', 'AMM');

INSERT INTO airline (id, name) VALUES

(1, 'Royal jordinian'), (2, 'Qatar airways'), (3, 'gulf air'), (4, 'emirates'), (5, 'etihad airline');

The select statements and joins are in the views and procedures in the [DDL section](#) or any other ones are in the [GUI section](#).

The previous [DDL](#) or [DML](#) statements resulted in these changes in the database and they are part of the [output and the data validations](#):

The tables that has been made:

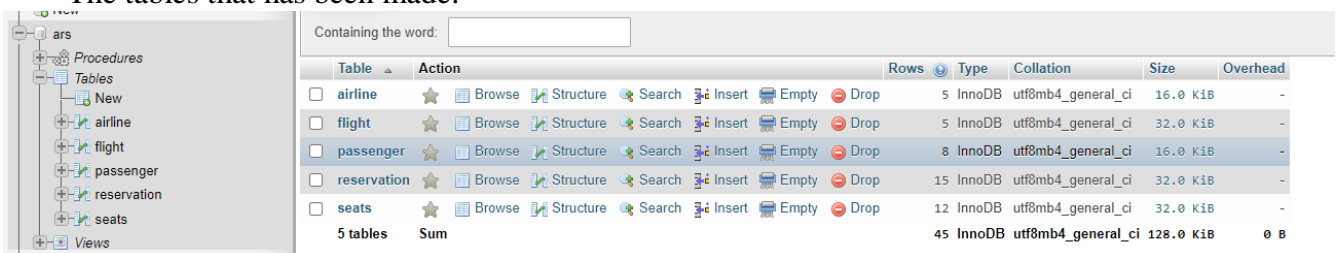
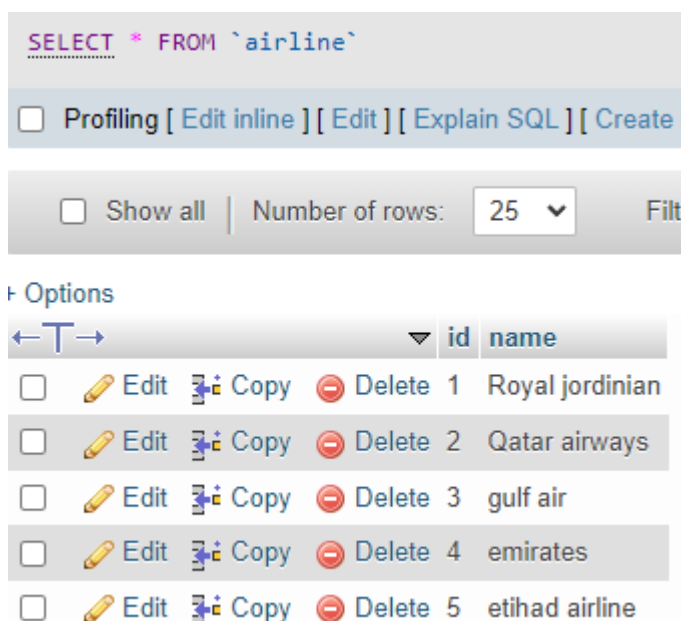


Table	Action	Rows	Type	Collation	Size	Overhead
airline	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	16.0 KiB	-
flight	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	32.0 KiB	-
passenger	★ Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_general_ci	16.0 KiB	-
reservation	★ Browse Structure Search Insert Empty Drop	15	InnoDB	utf8mb4_general_ci	32.0 KiB	-
seats	★ Browse Structure Search Insert Empty Drop	12	InnoDB	utf8mb4_general_ci	32.0 KiB	-
5 tables	Sum	45	InnoDB	utf8mb4_general_ci	128.0 KiB	0 B

The airline table with its data.



```
SELECT * FROM `airline`
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create]

☐ Show all | Number of rows: 25 Filter

Options

	id	name
<input type="checkbox"/> Edit Copy Delete	1	Royal jordinian
<input type="checkbox"/> Edit Copy Delete	2	Qatar airways
<input type="checkbox"/> Edit Copy Delete	3	gulf air
<input type="checkbox"/> Edit Copy Delete	4	emirates
<input type="checkbox"/> Edit Copy Delete	5	etihad airline

The flight table and the data that has been inserted to it.

SELECT * FROM `flight`

Profiling

Edit inline

Edit

Explain SQL

Create PHP code

Refresh

Show all

Number of rows: 25

Filter rows:

Sort by key: None

+ Options

←

→

flight_number

airline_id

departure_time

departure_date

arrival_time

arrival_date

flight_type

arrival_country

departure_country

arrival_airport

departure_airport

Edit

Copy

Delete

123

2

10:55:00

2022-03-26

22:00:00

2022-04-09

economy

paris

amman

CDG

AMM

Edit

Copy

Delete

153

4

12:00:00

2022-04-10

14:00:00

2022-04-15

economy

Dubai

amman

DXB

AMM

Edit

Copy

Delete

154

5

22:30:00

2022-05-04

01:30:00

2022-05-15

economy

Doha

amman

DOH

AMM

Edit

Copy

Delete

234

5

02:30:00

2022-04-04

06:30:00

2022-05-15

first class

Jeddah

amman

JED

AMM

Edit

Copy

Delete

334

2

13:30:00

2022-04-05

15:30:00

2022-05-15

first class

Jeddah

amman

JED

AMM

The reservation table with its inserted data.

SELECT * FROM `reservation`

Profiling

[Edit inline]

[Edit]

[Explain SQL]

[Create PHP code]

[Refresh]

Show all

Number of rows: 25

Filter rows:

Search this table

+ Options

Edit

Copy

Delete

111

1888

500

Edit

Copy

Delete

1234

1888

500

Edit

Copy

Delete

3434

1998

200

Edit

Copy

Delete

5234

1928

1000

Edit

Copy

Delete

55887

24440

600

Edit

Copy

Delete

789456

27985

300

The seats table with the data that is inserted to it.

reservation_number	flight_number	seat_number	seat_type
111	123	125	eco
1234	123	45	economy
3434	153	35	economy
5234	154	25	first class
55887	234	65	first-class
789456	334	75	first-class

The view table that has been made.

SELECT * from flights;														
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]														
Show all Number of rows: 25 Filter rows: Search this table														
+ Options														
flight_number	airline_id	departure_time	departure_date	arrival_time	arrival_date	flight_type	arrival_country	departure_country	arrival_airport	departure_airport	name	seat_number	price	
123	2	10:55:00	2022-03-26	22:00:00	2022-04-09	economy	paris	amman	CDG	AMM	Qatar airways	125	500	
123	2	10:55:00	2022-03-26	22:00:00	2022-04-09	economy	paris	amman	CDG	AMM	Qatar airways	45	500	
334	2	13:30:00	2022-04-05	15:30:00	2022-05-15	first class	Jeddah	amman	JED	AMM	Qatar airways	75	300	
153	4	12:00:00	2022-04-10	14:00:00	2022-04-15	economy	Dubai	amman	DXB	AMM	emirates	35	200	
154	5	22:30:00	2022-05-04	01:30:00	2022-05-15	economy	Doha	amman	DOH	AMM	etihad airline	25	1000	
234	5	02:30:00	2022-04-04	06:30:00	2022-05-15	first class	Jeddah	amman	JED	AMM	etihad airline	65	600	

The procedure with its data when we execute it

Showing rows 0 - 1 (2 total, Query took 0.0007 seconds.)

CALL getflightinfo(1888);

[Edit inline] [Edit] [Create PHP code]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

flight_number	airline_id	departure_time	departure_date	arrival_time	arrival_date	flight_type	arrival_country	departure_country	arrival_airport	departure_airport	name	seat_number	price
123	2	10:55:00	2022-03-26	22:00:00	2022-04-09	economy	paris	amman	CDG	AMM	Qatar airways	125	500
123	2	10:55:00	2022-03-26	22:00:00	2022-04-09	economy	paris	amman	CDG	AMM	Qatar airways	45	500

The passenger table with its data

		passport_id	gender	nationality	DOB	phone_number	email	fname	mname	lname	pass_word
<input type="checkbox"/>	Edit	Copy	Delete	1888	F	Jordanian	2002-03-26 7922223	lamis.smara@htu.edu.jo	lamis	smara	764456a30b11a0231ed4405a45ffc1a4
<input type="checkbox"/>	Edit	Copy	Delete	1928	M	Jordanian	2001-04-09 790290514	saifedeen@htu.edu.jo	saifedeen	mohammed	kharouf 4d0c5af710fb34ba9877183bf7e05b15
<input type="checkbox"/>	Edit	Copy	Delete	1998	F	Jordanian	2002-08-13 792222323	shahed.nwasieh@htu.edu.jo	shahed	nwasieh	a44987440c0b9efdbd270eb6e4afe279
<input type="checkbox"/>	Edit	Copy	Delete	24440	M	jordanian	2000-03-05 7894500	hussein.tariq@htu.edu.com	hussien	tareq	emad a1285459e416a3c2fc3b67025265fe4e
<input type="checkbox"/>	Edit	Copy	Delete	27985	M	jordanian	2001-02-23 79258211	mohammed.masharqa@htu.edu.com	mohammed	saleh	masharqa cd2cbd0a7d28268acfa68c32fcd6bcd6
<input type="checkbox"/>	Edit	Copy	Delete	36999	M	jordanian	2001-04-09 790290514	kharoofsaif@htu.edu.jo	fahed	moh	hossam 4d0c5af710fb34ba9877183bf7e05b15
<input type="checkbox"/>	Edit	Copy	Delete	222222	F	jordanian	2021-12-30 7902984	omar.jerab@htu.edu.jo	omar	jerab	a44987440c0b9efdbd270eb6e4afe279

Note I did an alter to this table

ALTER TABLE passenger

DROP COLUMN pass_word; (to change the type from INT to varchar)

ALTER TABLE passenger

ADD COLUMN pass_word varchar(32)

And the update statements

UPDATE passenger SET pass_word = md5('12dd') WHERE passport_id = 1888

UPDATE passenger SET pass_word = md5('13c') WHERE passport_id =222222

UPDATE passenger SET pass_word = md5('13c') WHERE passport_id =1998;

UPDATE passenger SET pass_word = md5('14c') WHERE passport_id =1928;

UPDATE passenger SET pass_word = md5('14c') WHERE passport_id =36999;

UPDATE passenger SET pass_word = md5('17c') WHERE passport_id =27985;

UPDATE passenger SET pass_word = md5('18c') WHERE passport_id =24440;

The md5 is a hashing function that can encrypt the data which is related to the [security](#).

f. DCL / security

[As in the user groups discussed in its section](#), I made one of the users and it's the passenger and I granted him these statements to do as the requirements stated exactly without seeing the other data:

```
ADD USER passenger IDENTIFIED BY 123 (to make a user to the system)
```

```
GRANT EXECUTE ON PROCEDURE getflightinfo TO passenger
```

```
GRANT SELECT ON flights TO passenger
```

```
GRANT SELECT ON passengers TO passenger
```

```
GRANT INSERT ON reservation TO passenger
```

```
GRANT INSERT ON seats TO passenger
```

```
GRANT INSERT ON passengers TO passenger
```

g. Database maintenance

s(MySQL Maintenance Automation – SqlBak Blog, no date)

- Optimize: when we do the delete command it is not deleted actually it is marked as deleted this can lead to fragmentation which can cause the database files large in the relation to the data and database slowdown. We need to optimize it if we did a big update or delete commands using this query we can understand the degree of fragmentation in tables:

```
SELECT ENGINE,
```

```
concat(TABLE_SCHEMA, '.', TABLE_NAME) as Table_Name,
```

```
round(DATA_LENGTH/1024/1024, 2) as Data_Length,
```

```
round(INDEX_LENGTH/1024/1024, 2) as Index_Length,
```

```
round(DATA_FREE/1024/1024, 2) as Data_Free,
```

```
(data_free/(index_length+data_length)) as Frag_Ratio
```

```
FROM information_schema.tables
```

```
WHERE DATA_FREE > 0
```

```
ORDER BY frag_ratio DESC;
```

If the percentage is less than 5% nothing to be done even it is 20%. If there were any slow performance we need to use the optimize queries:

```
mysql>OPTIMIZE TABLE sakila.actor (for tables )
```

```
mysqlcheck --user=root --password=secret-pass --optimize sakila (for one schema)
```

```
mysqlcheck --user=root --password=secret-pass --optimize --all-databases (for all schemas)
```

- Analyse: it do a read lock when executed so we need to use it when it is at minimum load on the servers. The queries to do it are:

```
ANALYZE TABLE sakila.actor; (for single table)
```

```
mysqlcheck --user=root --password=secret-pass --analyze sakila (for a schema)
```

```
mysqlcheck --user=root --password=secret-pass --all-databases (for all schemas)
```

h. GUI development

To establish the connection we need to use this code

```
<?php
$host = 'localhost';
$dbname = 'ars';
$user = 'passenger';
$pass = '123';

$pdo = new PDO("mysql:host=$host;dbname=$dbname",$user,$pass);

?>
```

The index page which he can reserve the flight and see the available flights [\(the final result will be the user documentation\)](#)

```
<?php
session_start();
if(!isset($_SESSION['email'])){
    header('location: login-page.php');
}
require "DB conection.php";
$sql = "SELECT * FROM filghts";
$statement = $pdo->prepare($sql);
$statement->execute();
$data= $statement->fetchAll();

if ($_SERVER['REQUEST_METHOD'] == 'POST'){
    $sql3 = "INSERT into reservation values (:reservation_number, :passport_id, :price)";
    $statement2 = $pdo->prepare($sql3);
```

```

$rese = mt_rand();//gives a unique random number
$statement2 -> bindParam(':reservation_number' , $rese , PDO::PARAM_INT);
$statement2 -> bindParam(':passport_id' , $_SESSION['pass_id'] , PDO::PARAM_INT);
$statement2 -> bindParam(':price' , $_POST['price'] , PDO::PARAM_INT);
$statement2 -> execute();

$sql4 = "INSERT into seats values (:reservation_number,
:flight_number,:seat_number,:seat_type)";
$statement3 = $pdo->prepare($sql4);
$statement3 -> bindParam(':reservation_number' , $rese , PDO::PARAM_INT);
$statement3 -> bindParam(':flight_number' , $_POST['f_number'] , PDO::PARAM_INT);
$statement3 -> bindParam(':seat_number' , $_POST['s_number'] , PDO::PARAM_INT);
$statement3 -> bindParam(':seat_type' , $_POST['s_type'] , PDO::PARAM_STR);
$statement3 -> execute();
$pdo = null;
header('location:ticiket_info.php');
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <link rel="stylesheet" href="style/header&footer.css">
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style>
table {
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 100%;
}

td, th {
    border: 1px solid #dddddd;
    text-align: left;
    padding: 8px;
}

tr:nth-child(even) {
    background-color: #dddddd;
}
</style>
</head>
<body>
<?php

if(isset($_SESSION['email'])){
    require 'header - logged-in.php';
} else {
    require 'header - not-logged-in.php';
}
?>
<table>

```

```

<tr>
  <th>flight number</th>
  <th>departure time</th>
  <th>departure date</th>
  <th>arrival time</th>
  <th>arrival date</th>
  <th>airline name</th>
  <th>arrival country</th>
  <th>departure country</th>
  <th>arrival airport</th>
  <th>departure airport</th>
  <th>flight type</th>
  <th>price</th>
  <th>seat number</th>
  <th>reserve</th>
</tr>
<?php foreach($data as $row){?>
<tr>
  <td><?php echo $row['flight_number']?></td>
  <td><?php echo $row['departure_time']?></td>
  <td><?php echo $row['departure_date']?></td>
  <td><?php echo $row['arrival_time']?></td>
  <td><?php echo $row['arrival_date']?></td>
  <td><?php echo $row['name']?></td>
  <td><?php echo $row['arrival_country']?></td>
  <td><?php echo $row['departure_country']?></td>
  <td><?php echo $row['arrival_airport']?></td>
  <td><?php echo $row['departure_airport']?></td>
  <td><?php echo $row['flight_type']?></td>
  <td><?php echo $row['price']?></td>
  <td><?php echo $row['seat_number']?></td>
  <td><form action="index.php" method = 'post' ><button>reserve</button>
    <input type = 'hidden' name='price' value='<?php echo $row['price']?>'>
    <input type = 'hidden' name='s_number' value='<?php echo $row['seat_number']?>'>
    <input type = 'hidden' name='s_type' value='<?php echo $row['flight_type']?>'>
    <input type = 'hidden' name='f_number' value='<?php echo $row['flight_number']?>'>
  </form></td>
</tr>
<?php }?>
</table>

<?php include 'footer.php'?>
</body>
</html>

```

This is the login page where he should be if he does not have logged in ([the final result will be the user documentation](#))

```

<?php
  session_start();
  $msgBox = '';

  if ($_SERVER['REQUEST_METHOD'] == 'POST'){

```

```

        require "DB conection.php";

        $sql = "SELECT * FROM passenger WHERE email = :email AND pass_word =
md5(:pass_word)";

        $email = $_POST["email"];
        $pass_word = $_POST["pass"];

        $statement = $pdo->prepare($sql);
        $statement->bindParam(':email', $email, PDO::PARAM_STR);
        $statement->bindParam(':pass_word', $pass_word, PDO::PARAM_STR);
        $statement->execute();
        $data = $statement->fetch();
        $pdo = null;

        if(!$data){
            $msgBox = "<script>alert('error')</script>";
        }else{
            $_SESSION['email'] = $data["email"];
            $_SESSION['pass_id'] = $data["passport_id"];
        }
    }

    if(isset($_SESSION['email'])){
        header("Location: index.php");
        exit();
    }
?>

<!DOCTYPE html>
<html>

<head>
    <meta name = "viewport" content = "width = device-width , initial-scale =1.0">
    <link rel="stylesheet" href="style/header&footer.css">
    <link rel="stylesheet" href="style/login.css">
    <script src="home-btn-effect.js"></script>
</head>

<body>
<?php

    if(isset($_SESSION['email'])){
        require 'header - logged-in.php';
    } else {
        require 'header - not-logged-in.php';
    }

?>

<section>
    <span>
        <h2>Login</h2>

```

```

        <div class="container">

            <form action = 'login-page.php' method='post'>
                <div class="text">

                    <label for="email">Email</label>
                    <input type="email" id="emai" placeholder="Your
email...." name = 'email'>

                </div>

                <div class="text">

                    <label for="pass">Password</label>
                    <input type="password" id="pass" name="pass"
placeholder="Your last name.." >

                </div>

                <div class="text">

                    <input type="submit" value="Submit">

                </div>
                <?php echo $msgBox ?>
            </span>

        </form>
    </div>

</section>
<?php include 'footer.php'?>

</body>

</html>

```

This is the register page where he can make an account this is the PHP code

```

<?php
session_start();
require 'DB conection.php';
if ($_SERVER['REQUEST_METHOD'] == 'POST'){
    $sql = 'INSERT into passenger values (:passport_id, :gender, :nationality, :DOB,
:phone_number,:email, fname,:mname,:lname, md5(:pass_word))';

    $statement = $pdo->prepare($sql);
    $statement -> bindParam(':passport_id' , $_POST ['passport_id'] , PDO::PARAM_INT);
    $statement -> bindParam(':fname' , $_POST['fname'] , PDO::PARAM_STR);
    $statement -> bindParam(':mname' , $_POST['mname'] , PDO::PARAM_STR);
    $statement -> bindParam(':lname' , $_POST['lname'] , PDO::PARAM_STR);
    $statement -> bindParam(':phone_number' , $_POST['phone-number'] , PDO::PARAM_INT);
    $statement -> bindParam(':email' , $_POST['email'] , PDO::PARAM_STR);
}

```



```

$statement -> bindParam(':pass_word' , $_POST['pass'] , PDO::PARAM_STR);
$statement -> bindParam(':DOB' , $_POST['DOB'] , PDO::PARAM_LOB);
$statement -> bindParam(':gender' , $_POST['pay'] , PDO::PARAM_STR);
$statement -> bindParam(':nationality' , $_POST['nationality'] , PDO::PARAM_STR);
$statement -> execute();

$sql2 = "SELECT * FROM passenger WHERE email = :email";
$statement2 = $pdo->prepare($sql2);
$statement2 -> bindParam(':email' , $_POST['email'] , PDO::PARAM_STR);
$statement2 -> execute();
$row = $statement2->fetch();
$pdo = null;

$_SESSION['email'] = $row['email'];
$_SESSION['pass_id'] = $row['passport_id'];

if(isset($_SESSION['email'])) {
    header('location:index.php');
    exit();
}
}
?>

```

This is the logout code.

```

<?php
    session_start();
    session_destroy();
    header("Location: index.php");
?>

```

This is the ticket info where the user can see his flights information ([the final result will be the user documentation](#))

```

<?php
session_start();
require 'DB conection.php';
$sql100 = 'CALL getflightinfo(:passport_id)';
$statment10 = $pdo -> prepare($sql100);
$statment10 -> bindParam(':passport_id' , $_SESSION['pass_id'] , PDO::PARAM_INT);
$statment10 -> execute();
$data100 = $statment10->fetchAll();

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <link rel="stylesheet" href="style/header&footer.css">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style>
table {
    font-family: arial, sans-serif;
    border-collapse: collapse;

```

```

width: 100%;
}

td, th {
border: 1px solid #dddddd;
text-align: left;
padding: 8px;
}

tr:nth-child(even) {
background-color: #dddddd;
}
</style>
</head>
<body>
<body>
<?php
if(isset($_SESSION['email'])){
    require 'header - logged-in.php';
} else {
    require 'header - not-logged-in.php';
}

?>

<main>

<table>
<tr>
<th>flight number</th>
<th>departure time</th>
<th>departure date</th>
<th>arrival time</th>
<th>arrival date</th>
<th>airline name</th>
<th>price</th>
<th>seat number</th>
<th>flight type</th>
<th>arrival airport</th>
<th>arrival country</th>
<th>departure country</th>

</tr>
<?php foreach($data100 as $row){?>
<tr>
<td><?php echo $row['flight_number']?></td>
<td><?php echo $row['departure_time']?></td>
<td><?php echo $row['departure_date']?></td>
<td><?php echo $row['arrival_time']?></td>
<td><?php echo $row['arrival_date']?></td>
<td><?php echo $row['name']?></td>
<td><?php echo $row['price']?></td>
<td><?php echo $row['seat_number']?></td>
<td><?php echo $row['flight_type']?></td>
<td><?php echo $row['arrival_airport']?></td>

```

```

                <td><?php echo $row['departure_country']?></td>
                <td><?php echo $row['arrival_country']?></td>
            </tr>
        <?php }?>
    </table>

</main>

<?php
    include 'footer.php';
?>
</body>
</html>

```

This is the register page where he can make an account ([the final result will be the user documentation](#))

```

<!DOCTYPE html>
<html>

<head>
    <meta name = "viewport" content = "width = device-width , initial-scale =1.0">
    <link rel="stylesheet" href="style/header&footer.css">
    <link rel="stylesheet" href="style/registiter.css">
    <script src="home-btn-effect.js"></script>
    <style>

        @media (max-width: 800px) {

            .active11{
                top:9%;
            }

        }
    </style>
</head>

<body>

<?php
session_start();

    if(isset($_SESSION['email'])){
        require 'header - logged-in.php';
    } else {
        require 'header - not-logged-in.php';
    }

?>

<section class="container">

```

```
<h2>Registration</h2>
<form action = 'regesiter-inserting.php' method = 'post'>

    <div class="text">

        <label for="passport_id">passport number</label>
        <input type="text" id="passport_id" placeholder="Your number.." name
='passport_id'>

    </div>
    <div class="text">

        <label for="fname">First Name</label>
        <input type="text" id="fname" placeholder="Your name.." name ="fname">

    </div>

    <div class="text">

        <label for="mname">middle Name</label>
        <input type="text" id="mname" placeholder="Your last name.." name
="mname">

    </div>

    <div class="text">

        <label for="lname">Last Name</label>
        <input type="text" id="lname" placeholder="Your last name.." name
="lname">

    </div>

    <div class="text">

        <label for="email">email</label>
        <input type="email" id="email" placeholder="Your email...." name
="email">

    </div>

    <div class="text">

        <label for="password">password</label>
        <input type="password" id="password" placeholder="Your password.."
name ="pass">

    </div>

    <div class="text">

        <label for="nationality">nationality</label>
```

```

        <input type='text' id="nationality" placeholder="Your password.."
name ="nationality">

    </div>
    <div class="text">

        <label for="DOB">date of bith</label>
        <input type="date" id="start" name="DOB">

    </div>

    <div class="text">

        <label for="female"> <input type="radio" id="female"
name="pay" class="choice" value='F'> female </label>
        <label for="male"> <input type="radio" id="male" name="pay"
class="choice" value='M'> male </label>

    </div>

    <div class="text">

        <label for="phone-number">phone number</label>
        <input type="text" id="phone-number" placeholder="Your number.." name ='phone-
number'>

    </div>

<div class="text">

    <input type="submit" value="Submit">

</div>

</form>

</section>

<?php include 'footer.php'?>

</body>

</html>

```

This is the header page where it should appear if he have logged in [\(the final result will be the user documentation\)](#)

```

<header>

    <div id="logo-pic">

```

```

        <a href="index.php">  </a>
    </div>

    <div id="menu-item-btn">
        <button id="menubtn">
            <svg xmlns="http://www.w3.org/2000/svg" x="0px" y="0px"
width="30" height="30"
viewBox="0 0 172 172"
style=" fill:#09BC8A;"><g fill="none" fill-rule="nonzero"
stroke="none" stroke-width="1" stroke-linecap="butt" stroke-linejoin="miter" stroke-
miterlimit="10" stroke-dasharray="" stroke-dashoffset="0" font-family="none" font-
weight="none" font-size="none" text-anchor="none" style="mix-blend-mode: normal"><path
d="M0,172v-172h172v172z" fill="none"></path><g fill="#09bc8a"><path d="M86,11.46667c-
1.50095,0 -2.84095,0.61145 -3.86328,1.55651l-0.02239,-0.0224l-0.16797,0.14557l-
62.9211,51.45443l0.0112,0.0336c-1.11873,1.04651 -1.83646,2.51402 -1.83646,4.16562c0,3.1648
2.56853,5.73333 5.73333,5.73333h5.73333v63.06667c0,6.33533 5.13133,11.46667
11.46667,11.46667h91.73333c6.33533,0 11.46667,-5.13133 11.46667,-11.46667v-
63.06667h5.73333c3.1648,0 5.73333,-2.56853 5.73333,-5.73333c0,-1.6516 -0.71773,-3.11912 -
1.83646,-4.16562l0.0112,-0.0336l-21.10808,-17.26719v-24.40026c0,-3.1648 -2.56853,-5.73333
-5.73333,-5.73333h-5.73333c-3.1648,0 -5.73333,2.56853 -5.73333,5.73333v10.33568l-
24.61302,-20.12266l-0.16797,-0.14557l-0.02239,0.0224c-1.02233,-0.94507 -2.36233,-1.55651 -
3.86328,-1.55651zM68.8,91.73333h34.4v45.86667h-34.4z"></path></g></g></svg>
        </button>
    </div>

    <nav class="main-menu active11">
        <div id="menu-items1">

            <div id="search-bar">
                <form action ="" method ='get' id="header-itmes">
                    <input type="searchbox" placeholder="type to search"
id="search" size="40" name ='searching'>
                    <button>search</button>
                </form>
            </div>

            <div id="account">
                <a href="ticiket_info.php" class="links">
                    <b>account</b>
                </a>
            </div>

            <div class="logout">
                <a href="logout.php" class="links">
                    <b>Logout</b>
                </a>
            </div>

        </div>
    </nav>
</header>

```

This is the header page where it should appear if he does not logged in ([the final result will be the user documentation](#))

```
<header>

    <div id="logo-pic">
        <a href="index.php">  </a>
    </div>

    <div id="menu-item-btn">
        <button id="menubtn">
            <svg xmlns="http://www.w3.org/2000/svg" x="0px" y="0px"
width="30" height="30"
viewBox="0 0 172 172"
style=" fill:#09BC8A;"><g fill="none" fill-rule="nonzero"
stroke="none" stroke-width="1" stroke-linecap="butt" stroke-linejoin="miter" stroke-
miterlimit="10" stroke-dasharray="" stroke-dashoffset="0" font-family="none" font-
weight="none" font-size="none" text-anchor="none" style="mix-blend-mode: normal"><path
d="M0,172v-172h172v172z" fill="none"></path><g fill="#09bc8a"><path d="M86,11.46667c-
1.50095,0 -2.84095,0.61145 -3.86328,1.55651l-0.02239,-0.0224l-0.16797,0.14557l-
62.9211,51.45443l0.0112,0.0336c-1.11873,1.04651 -1.83646,2.51402 -1.83646,4.16562c0,3.1648
2.56853,5.73333 5.73333,5.73333h5.73333v63.06667c0,6.33533 5.13133,11.46667
11.46667,11.46667h91.73333c6.33533,0 11.46667,-5.13133 11.46667,-11.46667v-
63.06667h5.73333c3.1648,0 5.73333,-2.56853 5.73333,-5.73333c0,-1.6516 -0.71773,-3.11912 -
1.83646,-4.16562l0.0112,-0.0336l-21.10808,-17.26719v-24.40026c0,-3.1648 -2.56853,-5.73333
-5.73333,-5.73333h-5.73333c-3.1648,0 -5.73333,2.56853 -5.73333,5.73333v10.33568l-
24.61302,-20.12266l-0.16797,-0.14557l-0.02239,0.0224c-1.02233,-0.94507 -2.36233,-1.55651 -
3.86328,-1.55651zM68.8,91.73333h34.4v45.86667h-34.4z"></path></g></g></svg>
        </button>
    </div>

    <nav class="main-menu active11">
        <div id="menu-items1">

            <div id="search-bar">
                <form action="" method='get' id="header-itmes">
                    <input type="searchbox" placeholder="type to search"
id="search" size="40" name="searching">
                    <button>search</button>
                </form>

            </div>

            <div class="login">
                <a href="login-page.php" class="links">
                    <b>login</b>
                </a>
            </div>

            <div class="register">
                <a href="register.php" class="links">
```

```

                <b>register</b>
            </a>
        </div>

    </div>

</nav>

</header>

```

This is the footer for the pages

```

<footer>

    <div class="footer-info">
        <div id="contact-us" class="footer-text">
            <b>contact-us</b>
        </div>
        <ul>
            <li>facebook</li>
            <li>facebook</li>
            <li>facebook</li>
        </ul>
    </div>

    <div class="footer-info">
        <div id="help-center" class="footer-text">
            <b>help-center</b>
        </div>
        <ul>
            <li>Phonenumber: 0790290514</li>
        </ul>
    </div>

    <div class="footer-info">
        <div id="about-us" class="footer-text">
            <b>about-us</b>
            <span>Lorem ipsum dolor sit amet consectetur adipisicing elit.
Tenetur sed laboriosam debitis maiores quod eligendi, vero et odio est accusantium.</span>
        </div>

    </div>

</footer>

```


User documentation

The first page will show up if he does not have an account is the login page



Email

Your email....

Password

Your last name..

Submit

contact-us

facebook

facebook

facebook

about-us

Lorem ipsum dolor sit amet consectetur adipisicing elit. Tenetur sed laboriosam debitis maiores quod eligendi, vero et odio est accusantium.

help-center

Phonenumber: 0790290514

This is the elements in this page:

NO.	Name	Place	Type	Description	Message
1.	Email	Middle	Textbox	Input for the phone email of the user	This email is not available
2.	Password	Middle	Textbox	Input for the password number of the user	Wrong password

The second page is if he does not have any account he can go through this page and make one



Registration

passport number

First Name

middle Name

Last Name

email

password

nationality

date of bith


☐ female
 ☐ male

phone number

The elements in this page

NO.	Name	Place	Type	Description	Message
1.	Phone number	Middle	Textbox	Input for the phone number of the user	
2.	Fname	Middle	Textbox	Input for the first name of the user	
3.	Lname	Middle	Textbox	Input for the last name of the user	
4.	Email	Middle	Textbox	Input for the phone email of the user	
5.	nationality	Middle	Textbox	Input for the nationality of the user	
6.	Password	Middle	Textbox	Input for the password number of the user	
7.	gender	Middle	Radio button	A choice between female or male	
8.	Date of birth	Middle	Date	He can put what date he was born in	
9.	Submit	Middle	Button	He can submit the information he entered	
10.	Mname	Middle	Textbox	Input for the middle name of the user	

The third page is if he got logged in or registered will go through which is the index page or the flights available

<div>  <div> <input type="text" value="type to search"/> <input type="button" value="search"/> </div> <div> account Logout </div> </div>													
flight number	departure time	departure date	arrival time	arrival date	airline name	arrival country	departure country	arrival airport	departure airport	flight type	price	seat number	reserve
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	paris	amman	CDG	AMM	economy	500	125	reserve
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	paris	amman	CDG	AMM	economy	500	45	reserve
153	12:00:00	2022-04-10	14:00:00	2022-04-15	emirates	Dubai	amman	DXB	AMM	economy	200	35	reserve
154	22:30:00	2022-05-04	01:30:00	2022-05-15	etihad airline	Doha	amman	DOH	AMM	economy	1000	25	reserve
234	02:30:00	2022-04-04	06:30:00	2022-05-15	etihad airline	Jeddah	amman	JED	AMM	first class	600	65	reserve
334	13:30:00	2022-04-05	15:30:00	2022-05-15	Qatar airways	Jeddah	amman	JED	AMM	first class	300	75	reserve

[contact-us](#)

- facebook
- facebook
- facebook

[help-center](#)

- Phonenumber: 0790290514

[about-us](#) Lorem ipsum dolor sit amet consectetur adipisicing elit. Tenetur sed laboriosam debitis maiores quod eligendi, vero et odio est accusantium.

The elements in this page

NO.	Name	Place	Type	Description	Message
1.	Reserve	Right	button	It will lead to the account or the ticket info page and reserve a flight	

The fourth page is the account page or the ticket info page where he can see the flights he reserved.

No elements in this page

type to search

search

account

Logout

flight number	departure time	departure date	arrival time	arrival date	airline name	price	seat number	flight type	arrival airport	departure airport	arrival country	departure country
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	500	125	economy	CDG	AMM	amman	paris
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	500	45	economy	CDG	AMM	amman	paris

contact-us

facebook

facebook

facebook

help-center

Phonenumber: 0790290514

about-us

Lorem ipsum dolor sit amet consectetur adipisicing elit. Tenetur sed laboriosam debitis maiores quod eligendi, vero et odio est accusantium.

The main elements in these pages on the header

NO.	Name	Place	Type	Description	Message
1.	account	Top	link	It will lead to the account or the ticket info page (it will show when he sign in)	
2.	logout	Top	link	It will lead to get out of the session and logout of the system(it will show when he sign in)	
3.	login	Top	link	It will lead to the login page (it will show when he is not sign in)	
4.	Register	Top	link	It will lead to the register page (it will show when he is not sign in)	
5.	Logo pic	Top	Link	Lead to the index page (always shown)	

Testing

- a. Data validation
- b. Output validation

The data has meaning to it and the select statements and the execution of the tables are working properly as it showed in the end of the section of the [DML](#).

Then I tested the primary keys if they will give me errors if I put a duplicate value and it worked as intended

```
INSERT INTO airline(id) VALUES (1);
```

MySQL said: ?

#1062 - Duplicate entry '1' for key 'PRIMARY'

```
INSERT INTO flight(flight_number) VALUES (123);
```

MySQL said: ?

#1062 - Duplicate entry '123' for key 'PRIMARY'

```
INSERT INTO passenger(passport_id) VALUES (1888);
```

MySQL said: ?

#1062 - Duplicate entry '1888' for key 'PRIMARY'

```
INSERT INTO reservation(reservation_number) VALUES (111);
```

MySQL said: ?

#1062 - Duplicate entry '111' for key 'PRIMARY'

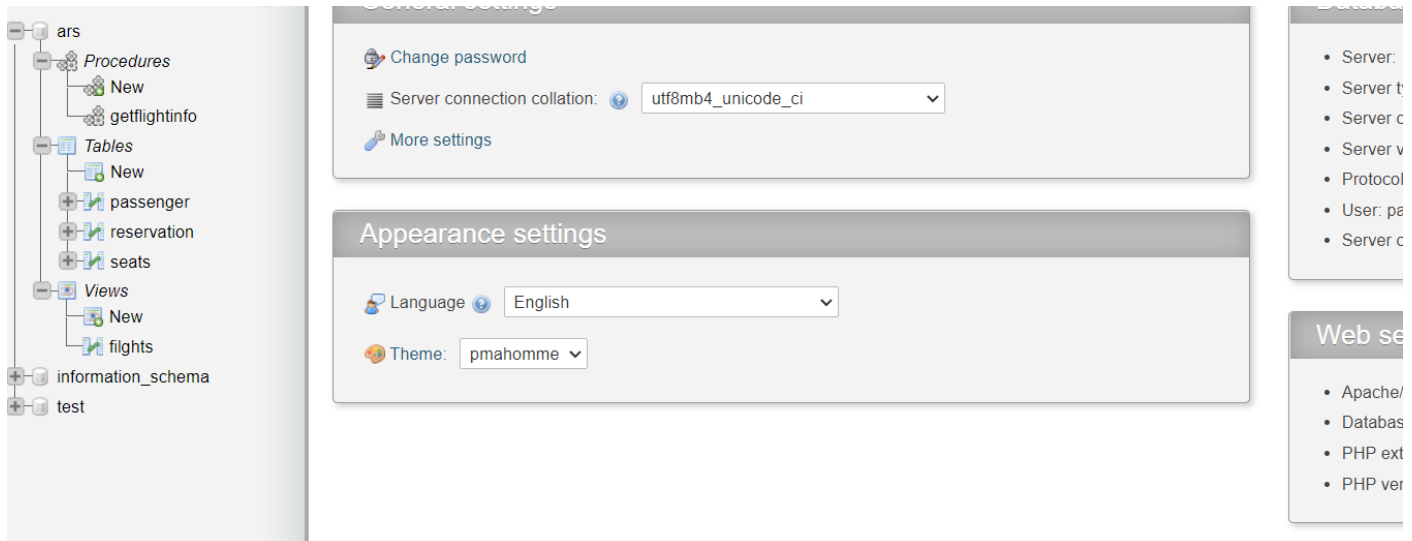
```
INSERT INTO seats(reservation_number, flight_number) VALUES (111,123);
```

MySQL said: ?

#1062 - Duplicate entry '111-123' for key 'PRIMARY'

c. Security validation

The user has the access of the intended tables



The user has not access to select from the reservation table and it worked as intended

```
SELECT * FROM reservation LIMIT 0, 25
```

MySQL said: ?

```
#1142 - SELECT command denied to user 'passenger'@'localhost' for table 'reservation'
```

The user has not access to select from the seats table and it worked as intended

```
SELECT * FROM seats LIMIT 0, 25
```

MySQL said: ?

```
#1142 - SELECT command denied to user 'passenger'@'localhost' for table 'seats'
```

d. GUI validation

As it is show in these pictures the user can see the data he meant to see only

<div><div><input type="text" value="type to search"/> <input type="button" value="search"/></div><div>accountLogout</div></div>													
flight number	departure time	departure date	arrival time	arrival date	airline name	arrival country	departure country	arrival airport	departure airport	flight type	price	seat number	reserve
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	paris	amman	CDG	AMM	economy	500	125	<input type="button" value="reserve"/>
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	paris	amman	CDG	AMM	economy	500	45	<input type="button" value="reserve"/>
153	12:00:00	2022-04-10	14:00:00	2022-04-15	emirates	Dubai	amman	DXB	AMM	economy	200	35	<input type="button" value="reserve"/>
154	22:30:00	2022-05-04	01:30:00	2022-05-15	etihad airline	Doha	amman	DOH	AMM	economy	1000	25	<input type="button" value="reserve"/>
234	02:30:00	2022-04-04	06:30:00	2022-05-15	etihad airline	Jeddah	amman	JED	AMM	first class	600	65	<input type="button" value="reserve"/>
334	13:30:00	2022-04-05	15:30:00	2022-05-15	Qatar airways	Jeddah	amman	JED	AMM	first class	300	75	<input type="button" value="reserve"/>

contact-us <ul style="list-style-type: none">facebookfacebookfacebook	help-center <ul style="list-style-type: none">Phonenummer: 0790290514	about-us Lorem ipsum dolor sit amet consectetur adipisicing elit. Tenetur sed laboriosam debitis maiores quod eligendi, vero et odio est accusantium.
---	---	---

<div><div><input type="text" value="type to search"/> <input type="button" value="search"/></div><div>accountLogout</div></div>													
flight number	departure time	departure date	arrival time	arrival date	airline name	price	seat number	flight type	arrival airport	departure airport	arrival country	departure country	
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	500	125	economy	CDG	AMM	amman	paris	
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	500	45	economy	CDG	AMM	amman	paris	

contact-us <ul style="list-style-type: none">facebookfacebookfacebook	help-center <ul style="list-style-type: none">Phonenummer: 0790290514	about-us Lorem ipsum dolor sit amet consectetur adipisicing elit. Tenetur sed laboriosam debitis maiores quod eligendi, vero et odio est accusantium.
---	---	---

Registration

passport number

5556699

First Name

test

middle Name

test

Last Name

test

email

test@gmail.com

password

...

nationality

jordanian

date of birth

08 / 05 / 2001

☐ female

☒ male

phone number

0790825205

Submit

Also he can register and make an account as intended and it saves to the database and this is also a [security validation](#) that he has an access to the insert statement on the passenger table

Edit	Copy	Delete	5556699	M	jordanian	2001-05-08	790825205	test@gmail.com	test	test	test	098f6bcd4621d373cade4e832627b4f6
------	------	--------	---------	---	-----------	------------	-----------	----------------	------	------	------	----------------------------------

Email

test1@gmail.com

Password


....

Submit



And when he tries to login with no real account or stored in the database he got an error, and this is also a [security validation](#) that he has an access to the select statement on the passenger table.

And when he tries to click reserve on the index page he goes to the account page or the ticket info and it inserts into the table of seats and has an execute grant on the procedure, and this is also a [security validation](#) that he has an access to the select statement on the passenger table.



type to search

search

account

Logout

flight number	departure time	departure date	arrival time	arrival date	airline name	price	seat number	flight type	arrival airport	departure airport	arrival country	departure country
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	500	125	economy	CDG	AMM	amman	paris
123	10:55:00	2022-03-26	22:00:00	2022-04-09	Qatar airways	500	45	economy	CDG	AMM	amman	paris

contact-us

- facebook
- facebook
- facebook

help-center

- Phonenumber: 0790290514

[about-us](#)

Lorem ipsum dolor sit amet consectetur adipisicing elit. Tenetur sed laboriosam debitis maiores quod eligendi, vero et odio est accusantium.

Assessment

a. Data assessment

As discussed in the [user groups section](#) for the passenger it got aligned as intended as shown in the [testing section](#) he could see the ticket info and view the available flights and he can sign in to the system and make an account and reserve a flight.

b. Effectiveness of the design

As discussed in the [user groups section](#) for the passenger it got aligned as intended with [the design of the database](#) and they were blended in with each other as we saw in the database and tables that have been made and he could do the requirements as intended.

c. Recommendations for the improvements of the system

I recommend doing these changes as they will improve the effectiveness of the system:

- I would make another table for the reviews if any passenger had any problem with his flight or a problem in the system.
- I would make him to delete or modify the flights that has been reserved.
- I would make a search bar to search about a certain flights.
- I would make a payment method to pay for his reserved flights.

References

Data Modelling: Conceptual, Logical, Physical Data Model Types (no date). Available at: <https://www.guru99.com/data-modelling-conceptual-logical.html> (Accessed: 27 January 2022).

MySQL Maintenance Automation – SqlBak Blog (no date). Available at: <https://sqlbak.com/blog/mysql-maintenance-automation> (Accessed: 28 January 2022).