Project Title – Online Cab System.

Project Statement

Project Statement :

Online Cab system is about booking a cab online from any place within the provided source cities. The main objective in this project is to synchronize free/available cabs such that travelling distance of cabs remains approximately equal to average travelling distance of cabs, as much as possible. External cab owners cabs can also be added to this system, through two ways, either on contract or only for advertisement purposes, as described below.

Data Required

S.No.		DATA
1	CAB CATEGORIES	Type of category
		Number of cabs available for corresponding type.
2.	CAB DETAILS	Cab Registration number
		Number of maximum passenger seats available
		Luggage space available
		Fare per kilometer
		vehicle name
		Cab category
3.	DRIVER DETAILS	Driver name
		Father name
		Pan card number
		Driving License Number
		Blood group
		Cab category in which, he/she have expertise
		Salary
4.	Passenger/User Details	Name
		Email id
		Contact Number
5.	External cab owner details(providing cab on contract)	Name
		Email id
		Contact number
		Rent per month to be given.

		Pan card number
6.	External cab owner	Name
	details(non-contract, for advertisement)	Email id
		Contact number
		Fare per kilometer
		Pan card number
		Share percent required by us
7.	Booking Details	Pickup date
		Pickup time
		Source place
		Landmark(optional)
		destination

Functionality Provided :

PASSENGER / USER FUNCTIONALITY :

User can book cab online, within the provided cities. Payment will be done at the end of ride.

DRIVER FUNCTIONALITY :

A driver would have to drive the cab only when appointed by the cab system.

EXTERNAL CAB OWNER FUNCTIONALITY (on contract cab) :

An external owner can provide his cab to online cab system on contract. This cab may will be used independently by online cab system. All income made through this cab will be added to online cab system money account. The original owner will be paid a fixed monthly rent.

EXTERNAL CAB OWNER FUNCTIONALITY (non-contract cab, for advertisement purposes) :

Any car owner can provide an advertisement for his cab. This types of cab will be under non-contract agreement. Online cab system cannot use these types of cabs independently. Online cab system can only book these types of cab. Income made through it will be given directly to cab owner, after deducting charges (share per kilometer) of online cab system.

System constraints :

Each driver will be associated to a cab.

E-R Model :

E-R model of the project is attached in the folder.

Relational Schema :

Relational schema of the project is attached in the folder.

Normalization :

All the tables inside relational schema are NF-3 normalized.

Queries :

<u>Simple</u> :

- SELECT $tag FROM $tablename WHERE `email_id` = '$_POST[email_id]' and `password` = '$_POST[password]'

- INSERT INTO `passenger`(`fname`, `lname`, `email_id`, `contact_no`) values ('$_POST[fname]', '$_POST[lname]', '$_POST[email_id]', '$_POST[contact_no]'

- UPDATE `synchronize` SET `booked`='0' WHERE `cab_id` = NEW.cab_id

<u>Triggers</u> :

- CREATE TRIGGER `booking_to_log` AFTER UPDATE ON `booking` FOR EACH ROW BEGIN
  IF NEW.active = 0 THEN
  INSERT INTO `booking_log`(`booking_id`, `booking_datetime`, `pickup_datetime`, `source_place`, `destination_place`, `cab_id`, `driver_id`, `passenger_id`) VALUES (NEW.id, NEW.booking_datetime, NEW.pickup_datetime, NEW.source_place, NEW.destination_place, NEW.cab_id, NEW.driver_id, NEW.passenger_id);
  END IF;

END

- CREATE TRIGGER `free_cab` AFTER INSERT ON `booking_log`
  FOR EACH ROW BEGIN
  UPDATE `synchronize` SET `booked`='0' WHERE `cab_id` = NEW.cab_id;
  END

- CREATE TRIGGER `add_cab_sync` AFTER INSERT ON `cab`
  FOR EACH ROW BEGIN
  INSERT INTO `synchronize`(`cab_id`, `category_id`) VALUES (NEW.id,
  NEW.category_id);
  END

Transaction :

```
……………………………Transaction Begin……………………………………….
mysqli_autocommit($db_con, false);
…………………………………………………………………………………………..
$sql = "SELECT * FROM `passenger`WHERE `email_id` = '$_POST[email_id]'";
$result = mysqli_query($db_con, $sql);
if(mysqli_num_rows($result)==0){
  $sql = "INSERT INTO `passenger`(`fname`, `lname`, `email_id`, `contact_no`)
  values ('$_POST[fname]', '$_POST[lname]', '$_POST[email_id]',
  '$_POST[contact_no]')";
  if(mysqli_query($db_con, $sql)) {
  #echo "Signup sucessfull"."<br>";
  }
  else {
          $flag=false;
  echo "Error: " . $sql . "<br>" . mysqli_error($conn);
  }

  $sql = "SELECT `id` FROM `passenger` WHERE `email_id` =
  '$_POST[email_id]';";
  $result = mysqli_query($db_con, $sql);
```

```php
  $id = mysqli_fetch_assoc($result);
  $passenger_id = $id["id"];


  $sql = "INSERT INTO `passenger_login`(`email_id`, `password`, `passenger_id`)
  values ('$_POST[email_id]', '$_POST[password]', '$passenger_id')";
  if(mysqli_query($db_con, $sql)) {
  #echo "Account created"."<br>";
  }
  else {
              $flag=false;
  echo "Error: " . $sql . "<br>" . mysqli_error($db_con);
  }
  session_start();
  session_unset();
}
else{
  session_start();
  session_unset();
  $_SESSION["signup_error"]="Email-id already registered... Please login or try
  with different Email-id";
  header('location:login.php');
}

if($flag){
  ...........................Transaction committed.................................
  mysqli_commit($db_con);
  .................................................................................
  #echo "All queries were executed successfully";
} else {
  ........................Transaction rollback................................
  mysqli_rollback($db_con);
  .................................................................................
  #echo "All queries were rolled back";
}
mysqli_close($db_con);
```

Platform used :

Server   :   Apache2
Front-end   :   HTML , Bootstrap
Front-end styling :   CSS
Back-end   :   PHP
Database   :   SQL

Reports Generated :

First Report was generated on April 26, 2015. The following topics were included in the report :
    > Project Title
    > Project Statement
    > ER Model
    > Relational Schema

This final report is the extension of first reports having additional topics :
    > Normalized Relational Schema
    > Sample Queries
    > Platform Used
    > Reports Generated
    > Datasets
    > References
    > Summary
    > Feedback

Datasets :

In this project dataset were generated by team members. Non of the datasets were taken from anywhere.

<u>Test Done</u>

The following tests were done during testing phase of the project :
    > Login check
    > Signup check
    > User account duplicacy check
    > Booking a cab by passenger
    > Freeing the cab after the journey
    > Updating priority of cab after each journey
    > Adding external cabs
    > Searching a cab using search engine
    > Booking a cab via search engine results
    > logout check

<u>Size of the Data</u>

Data taken for testing was small but sufficient enough to test and demonstrate the working of projects. About eight to ten records were inserted in each table.

Reference :

The following reference were used in this project:

> http://www.w3schools.com/html/
> http://www.w3schools.com/php/
> http://www.w3schools.com/sql/
> http://www.w3schools.com/css/
> http://www.w3schools.com/bootstrap/

Summary :

This project was about building a automated online booking website. Several features were proposed during project selection. All the proposed tasks are successfully implemented in the project. Apart from them, additional features are implemented in the project like search engine, giving cabs on advertisement. Overall, this project successfully can be used as an online cab booking system.

Feedback :

The working experience was amazing while doing this project. It was because this project was more of back-end thing than front-end thing. We became familiar with lots of new things used in DBMS like, transactions, triggers, views, nested queries, joints , etc. Moreover, the project was itself awesome, because of which we never lose interest in the working of project. Overall, it was a nice experience.