

**CS 353**  
**Database Management Systems**  
**Term Project**

**Design Report:**  
**Travel Agency Data Management**  
**System**

**Section 1 Group 10**

*Ahmet Salman, 21901004, CS*

*Muhammed Can Küçükaslan, 21901779, CS*

*Mustafa Yasir Altunhan, 21802530, CS*

*Güven Gergerli, 21803393, CS*

**Instructor**

Prof. Dr. Özgür Ulusoy

# Contents

|  |           |
|--|-----------|
| <b>1. Revised E/R Model</b>                                      | <b>3</b>  |
| 1.1. Revision List   | 3         |
| 1.2 E/R Diagram  | 4         |
| <b>2. Relational Schemas</b>                                     | <b>5</b>  |
| 2.1. hotel   | 5         |
| 2.2. room  | 5         |
| 2.3. customer  | 6         |
| 2.4. employee  | 7         |
| 2.5. tour_guide  | 8         |
| 2.6. tour  | 8         |
| 2.7. activity  | 9         |
| 2.8. airport   | 9         |
| 2.9. tour_section  | 10        |
| 2.10. customer_review  | 11        |
| 2.11. tour_activity  | 11        |
| 2.12. booking  | 12        |
| 2.13. reservation  | 13        |
| 2.14. reservation_activity                                       | 14        |
| 2.15. tour_guide_review  | 14        |
| 2.16. guides   | 15        |
| 2.17. flight   | 16        |
| 2.18. flight_reservation   | 17        |
| <b>3. User Interface Design and Corresponding SQL Statements</b> | <b>18</b> |
| 3.1 Customer Register  | 18        |
| 3.2 Employee Register  | 19        |
| 3.3 Tour Guide Register  | 20        |
| 3.4 Customer Login   | 21        |
| 3.5 Employee Login   | 22        |
| 3.6 Tour Guide Login   | 23        |
| 3.7 Customer Dashboard   | 23        |
| 3.8 Customer Past Tours  | 24        |
| 3.9 Customer Rating  | 25        |
| 3.10 Customer Profile  | 25        |
| 3.11 Customer Hotel Reservation Details                          | 26        |
| 3.12 Customer Reserve a Hotel                                    | 27        |
| 3.13 Customer Book a Tour  | 27        |
| 3.14 Customer View Tour Guide Profile                            | 28        |

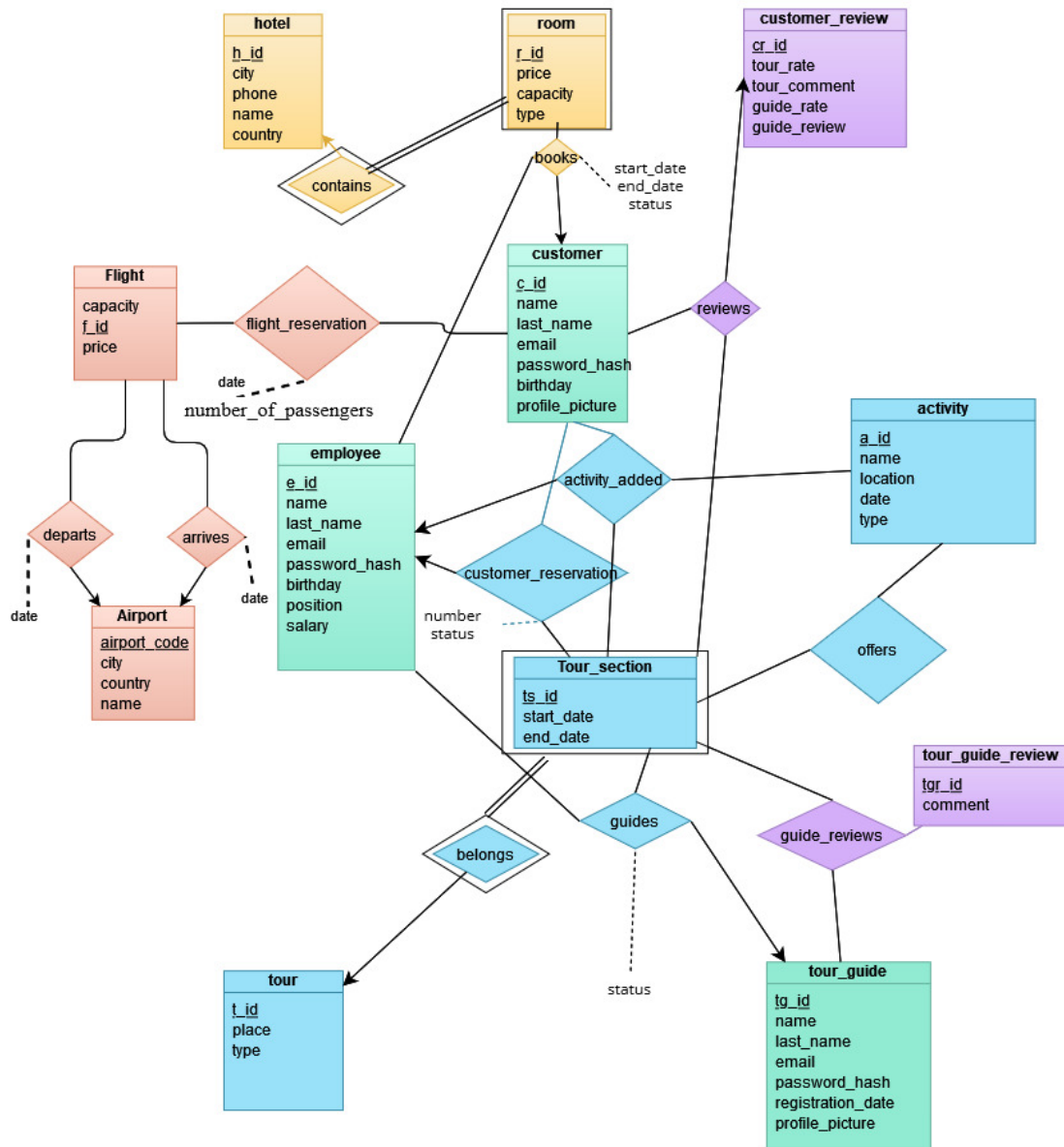
|  |    |
|--|----|
| 3.15 Customer Reserve Extra Activity     | 29 |
| 3.16 Customer Tour Details               | 29 |
| 3.17 Customer Reserve Flight             | 30 |
| 3.18 Employee Dashboard                  | 31 |
| 3.19 Employee Create New Tour            | 32 |
| 3.20 Employee Create New Activity        | 32 |
| 3.21 Employee Create New Extra Activity  | 33 |
| 3.22 Employee Assign Tour Guide          | 34 |
| 3.23 Edit Tour Information               | 34 |
| 3.24 Employee Register Hotel             | 35 |
| 3.25 Employee Pending Tours              | 35 |
| 3.26 Employee Pending Hotel Reservations | 36 |
| 3.27 Employee Profile                    | 37 |
| 3.28 Tour Guide Dashboard                | 37 |
| 3.29 Tour Guide Available Tours          | 38 |
| 3.30 Tour Guide Feedback Form            | 39 |
| 3.31 Tour Guide Profile                  | 39 |

# 1. Revised E/R Model

## 1.1. Revision List

- Removed the hotel\_company entity which was useless
- Hotel is no longer a weak entity.
- Hotel\_booking relation is renamed to books
- Removed the type attribute from the books relation
- Added status attribute to the books relation
- Customer\_review is no longer a weak entity.
- Review relation is renamed to reviews
- Foreign keys written Customer\_review are removed.
- Customer\_review's rating\_comment attribute is replaced by these four attributes: tour\_rate, tour\_comment, guide\_rate, guide\_comment.
- Tour\_guide\_review is no longer a weak entity.
- Remove the foreign keys buried in tour\_guide\_review
- Added tour\_section entity as a weak entity of tour, having start\_date, end\_date attributes.
- Added offers relation between tour\_section and activity.
- Added belongs relation between tour\_section and tour.
- Added place and type attributes to the tour entity.
- Remove Tour\_activity relation
- Activity is no longer a weak entity
- Remove the foreign keys in the activity entity
- Activity\_reservation relation is renamed to activity\_added
- Activity\_added is also connected to employee
- Removed "start\_date, end\_date" attributes from tour.
- Added credentials (e-mail and password\_hash) to the tour guide entity.
- Added registration\_date to tour guide entity.
- Renamed the tours relation to guides relation.
- Guides relation connects to the employee instead of the manager.
- Added status attribute to the guides relation
- Added credential attributes to the customer entity
- Added profilePicture attribute to the customer entity
- Added credential attributes to the employee entity
- Added salary attribute to the employee entity
- Removed the manager entity, its roles are taken by the employee.
- Add flight entity
- Add flight\_reservation entity connecting flight and customer.
- Add airport entity
- Add departs entity connecting flight and airport.
- Add arrives entity connecting flight and airport.

## 1.2 E/R Diagram



## 2. Relational Schemas

### 2.1. hotel

#### Relational Model:

hotel(h\_id, h\_name, phone, city, country)

#### Functional Dependencies:

h\_id -> h\_name, phone, city, country

#### Candidate Keys:

{h\_id}

#### Normal Form:

BCNF

#### Table Definition:

```
CREATE TABLE `hotel` (  
  `h_id` int(8) unsigned NOT NULL auto_increment,  
  `h_name` varchar(255),  
  `phone` varchar(12) default NULL,  
  `city` varchar(255) default NULL,  
  `country` varchar(255) default NULL,  
  PRIMARY KEY (`id`)  
) AUTO_INCREMENT=1;
```

### 2.2. room

#### Relational Model:

room(r\_id, price, capacity, type, h\_id)

#### Functional Dependencies:

r\_id -> price, capacity, type, h\_id

#### Candidate Keys:

{r\_id}

#### Normal Form:

BCNF

#### Table Definition:

```
CREATE TABLE `room` (  
  `h_id` int(11) NOT NULL,
```

```

`price` int(11) NOT NULL,
`capacity` int(11) NOT NULL,
`type` int(11) NOT NULL,
`r_id` int(11) NOT NULL AUTO_INCREMENT,
PRIMARY KEY (`r_id`),
KEY `fk_hotel_room` (`h_id`),
CONSTRAINT `fk_hotel_room` FOREIGN KEY (`h_id`) REFERENCES
    `hotel` (`h_id`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf16 COLLATE=utf16_turkish_ci

```

## 2.3. customer

### Relational Model:

customer(c\_id, name, last\_name, email, password\_hash, birthday, profile\_picture)

### Functional Dependencies:

c\_id -> name, last\_name, email, password\_hash, birthday, profile\_picture

### Candidate Keys:

{c\_id}

### Normal Form:

BCNF

### Table Definition:

```

CREATE TABLE `customer` (
    `c_id` int(11) NOT NULL AUTO_INCREMENT,
    `name` varchar(255) DEFAULT NULL,
    `lastname` varchar(255) DEFAULT NULL,
    `email` varchar(255) DEFAULT NULL,
    `password_hash` varchar(255) DEFAULT NULL,
    `birthday` date DEFAULT NULL,
    `profile_picture` blob NOT NULL,
    PRIMARY KEY (`c_id`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf16
COLLATE=utf16_turkish_ci

```

## 2.4. employee

### Relational Model:

employee(e\_id, name, last\_name, email, password\_hash, birthday, position, salary)

### Functional Dependencies:

e\_id -> name, last\_name, email, password\_hash, birthday, position, salary

### Candidate Keys:

{e\_id}

### Normal Form:

BCNF

### Table Definition:

```
CREATE TABLE `employee` (  
  `e_id` int(11) NOT NULL AUTO_INCREMENT,  
  `name` varchar(255) COLLATE utf16_turkish_ci DEFAULT NULL,  
  `lastname` varchar(255) COLLATE utf16_turkish_ci DEFAULT  
    NULL,  
  `email` varchar(255) COLLATE utf16_turkish_ci DEFAULT NULL,  
  `password_hash` varchar(255) COLLATE utf16_turkish_ci DEFAULT  
    NULL,  
  `birthday` date DEFAULT NULL,  
  `salary` int(11) NOT NULL,  
  `position` varchar(255) COLLATE utf16_turkish_ci NOT NULL,  
  PRIMARY KEY (`e_id`)  
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf16  
  COLLATE=utf16_turkish_ci
```



## 2.5. tour\_guide

### Relational Model:

tour\_guide(tg\_id, name, last\_name, email, password\_hash, registration\_date, profile\_picture)

### Functional Dependencies:

tg\_id -> name, last\_name, email, password\_hash, registration\_date, profile\_picture

### Candidate Keys:

{tg\_id}

### Normal Form:

BCNF

### Table Definition:

```
CREATE TABLE `tour_guide` (  
  `tg_id` int(11) NOT NULL AUTO_INCREMENT,  
  `name` varchar(255) DEFAULT NULL,  
  `lastname` varchar(255) DEFAULT NULL,  
  `email` varchar(255) DEFAULT NULL,  
  `password_hash` varchar(255) DEFAULT NULL,  
  `registration` date DEFAULT NULL,  
  `profile_picture` blob NOT NULL,  
  PRIMARY KEY (`tg_id`)  
) ENGINE=InnoDB
```

## 2.6. tour

### Relational Model:

tour(t\_id, place, type)

### Functional Dependencies:

t\_id -> place, type

### Candidate Keys:

{t\_id}

### Normal Form:

BCNF

### **Table Definition:**

```
CREATE TABLE `tour` (  
  `t_id` int(11) NOT NULL,  
  `place` varchar(255) NOT NULL,  
  `type` varchar(255) NOT NULL,  
  PRIMARY KEY (`t_id`)  
) ENGINE=InnoDB
```

## **2.7. activity**

### **Relational Model:**

activity(a\_id, name, location, date, type)

### **Functional Dependencies:**

a\_id -> name, location, date, type

### **Candidate Keys:**

{a\_id}

### **Normal Form:**

BCNF

### **Table Definition:**

```
CREATE TABLE `activity`(  
  `a_id` INT NOT NULL,  
  `name` VARCHAR(255) NOT NULL,  
  `location` VARCHAR(255) NOT NULL,  
  `date` DATE NOT NULL,  
  `type` VARCHAR(255) NOT NULL,  
  PRIMARY KEY(`a_id`)  
) ENGINE = InnoDB;
```

## **2.8. airport**

### **Relational Model:**

airport(airport\_code, city, country, name)

### **Functional Dependencies:**

airport\_code -> city, country, name

### **Candidate Keys:**

{airport\_code}

### **Normal Form:**

BCNF

### **Table Definition:**

```
CREATE TABLE `airport` (  
    `airport_code` CHAR(3) NOT NULL,  
    `name` VARCHAR(255) NOT NULL,  
    `city` VARCHAR(255) NOT NULL,  
    `country` VARCHAR(255) NOT NULL,  
    PRIMARY KEY(`airport_code`)  
) ENGINE = InnoDB;
```

## **2.9. tour\_section**

### **Relational Model:**

tour\_section(ts\_id, start\_date, end\_date, t\_id)

### **Functional Dependencies**

ts\_id -> start\_date, end\_date, t\_id

### **Candidate Keys:**

{ts\_id}

### **Normal Form:**

BCNF

### **Table Definition:**

```
CREATE TABLE `tour_section` (  
    `ts_id` INT NOT NULL AUTO_INCREMENT,  
    `start_date` DATE NOT NULL,  
    `end_date` DATE NOT NULL,  
    `t_id` INT NOT NULL,  
    CONSTRAINT `fk_section_tour` FOREIGN KEY (`t_id`)  
    REFERENCES `tour` (`t_id`) ON DELETE CASCADE ON  
    UPDATE CASCADE,  
    PRIMARY KEY(`ts_id`)  
) ENGINE = InnoDB;
```

## 2.10. customer\_review

### Relational Model:

customer\_review(cr\_id, tour\_rate, tour\_comment, guide\_rate, guide\_comment, c\_id, ts\_id)

### Functional Dependencies

cr\_id -> tour\_rate, tour\_comment, guide\_rate, guide\_comment, c\_id, ts\_id

### Candidate Keys:

{cr\_id}

### Normal Form:

BCNF

### Table Definition:

```
CREATE TABLE `customer_review` (  
    `cr_id` INT NOT NULL AUTO_INCREMENT,  
    `tour_rate` INT NOT NULL,  
    `tour_comment` VARCHAR(1000) NOT NULL,  
    `guide_rate` INT NOT NULL,  
    `guide_comment` VARCHAR(1000) NOT NULL,  
    `c_id` INT NOT NULL,  
    `ts_id` INT NOT NULL,  
    CONSTRAINT `fk_cr_customer` FOREIGN KEY(`c_id`)  
    REFERENCES `customer`(`c_id`) ON DELETE CASCADE ON  
    UPDATE CASCADE,  
    CONSTRAINT `fk_cr_tour_section` FOREIGN KEY(`ts_id`)  
    REFERENCES `tour_section`(`ts_id`) ON DELETE CASCADE  
    ON UPDATE CASCADE,  
    PRIMARY KEY(`cr_id`)  
) ENGINE = InnoDB;
```

## 2.11. tour\_activity

### Relational Model:

tour\_activity(ts\_id, a\_id)

### Functional Dependencies

None

### Candidate Keys:

{ts\_id, a\_id}

## Normal Form:

BCNF

## Table Definition:

```
CREATE TABLE `tour_activity`(  
    `ts_id` INT NOT NULL,  
    `a_id` INT NOT NULL,  
    CONSTRAINT `fk_ta_activity` FOREIGN KEY(`a_id`)  
    REFERENCES `activity`(`a_id`) ON DELETE CASCADE ON  
    UPDATE CASCADE,  
    CONSTRAINT `fk_ta_tour_section` FOREIGN KEY(`ts_id`)  
    REFERENCES `tour_section`(`ts_id`) ON DELETE CASCADE  
    ON UPDATE CASCADE  
)  
ENGINE = InnoDB;
```

## 2.12. booking

### Relational Model:

booking(b\_id, c\_id, r\_id, e\_id, start\_date, end\_date, type, status)

### Functional Dependencies

b\_id -> c\_id, r\_id, e\_id, start\_date, end\_date, type, status

### Candidate Keys:

{b\_id}

### Normal Form:

BCNF

## Table Definition:

```
CREATE TABLE `booking`(  
    `b_id` INT NOT NULL AUTO_INCREMENT,  
    `c_id` INT NOT NULL,  
    `r_id` INT NOT NULL,  
    `e_id` INT NOT NULL,  
    `start_date` DATE NOT NULL,  
    `end_date` DATE NOT NULL,  
    `type` VARCHAR(255) NOT NULL,  
    `status` VARCHAR(255) NOT NULL,
```

```

        CONSTRAINT `fk_booking_customer` FOREIGN KEY(`c_id`)
        REFERENCES `customer`(`c_id`) ON DELETE CASCADE ON
        UPDATE CASCADE,

        CONSTRAINT `fk_booking_employee` FOREIGN KEY(`e_id`)
        REFERENCES `employee`(`e_id`) ON DELETE CASCADE ON
        UPDATE CASCADE,

        CONSTRAINT `fk_booking_room` FOREIGN KEY(`r_id`)
        REFERENCES `room`(`r_id`) ON DELETE CASCADE ON
        UPDATE CASCADE,

        PRIMARY KEY(`b_id`)
    ) ENGINE = InnoDB;

```

## 2.13. reservation

### Relational Model:

reservation(res\_id, c\_id, e\_id, ts\_id, number, status)

### Functional Dependencies

res\_id -> c\_id, e\_id, ts\_id, number, status

### Candidate Keys:

{res\_id}

### Normal Form:

BCNF

### Table Definition:

```

CREATE TABLE `reservation` (
    `res_id` INT NOT NULL AUTO_INCREMENT,
    `c_id` INT NOT NULL,
    `ts_id` INT NOT NULL,
    `e_id` INT NOT NULL,
    `number` INT NOT NULL,
    `status` VARCHAR(255) NOT NULL,
    CONSTRAINT `fk_reservation_customer` FOREIGN
    KEY(`c_id`) REFERENCES `customer`(`c_id`) ON DELETE
    CASCADE ON UPDATE CASCADE,
    CONSTRAINT `fk_reservation_employee` FOREIGN
    KEY(`e_id`) REFERENCES `employee`(`e_id`) ON DELETE
    CASCADE ON UPDATE CASCADE,

```

```

        CONSTRAINT `fk_reservation_tour_section` FOREIGN
        KEY(`ts_id`) REFERENCES `tour_section`(`ts_id`) ON
        DELETE CASCADE ON UPDATE CASCADE,
        PRIMARY KEY(`res_id`)
    ) ENGINE = InnoDB

```

## 2.14. reservation\_activity

### Relational Model:

reservation\_activity(res\_id, a\_id)

### Functional Dependencies

None

### Candidate Keys:

{{res\_id, a\_id}}

### Normal Form:

BCNF

### Table Definition:

```

CREATE TABLE `reservation_activity`(
    `res_id` INT NOT NULL,
    `a_id` INT NOT NULL,
    CONSTRAINT `fk_reservation_activity_reservation`
    FOREIGN KEY(`res_id`) REFERENCES
    `reservation`(`res_id`) ON DELETE CASCADE ON UPDATE
    CASCADE,
    CONSTRAINT `fk_reservation_activity_activity`
    FOREIGN KEY(`a_id`) REFERENCES `activity`(`a_id`) ON
    DELETE CASCADE ON UPDATE CASCADE
) ENGINE = InnoDB

```

## 2.15. tour\_guide\_review

### Relational Model:

tour\_guide\_review(tgr\_id, comment, tg\_id, ts\_id)

### Functional Dependencies

tgr\_id -> comment, tg\_id, ts\_id

**Candidate Keys:**

{tgr\_id}

**Normal Form:**

BCNF

**Table Definition:**

```
CREATE TABLE `tour_guide_review` (  
    `tgr_id` INT NOT NULL AUTO_INCREMENT,  
    `tour_comment` VARCHAR(1000) NOT NULL,  
    `tg_id` INT NOT NULL,  
    `ts_id` INT NOT NULL,  
    CONSTRAINT `fk_tgr_tour_guide` FOREIGN KEY(`tg_id`)  
    REFERENCES `tour_guide`(`tg_id`) ON DELETE CASCADE  
    ON UPDATE CASCADE,  
    CONSTRAINT `fk_tgr_tour_section` FOREIGN  
    KEY(`ts_id`) REFERENCES `tour_section`(`ts_id`) ON  
    DELETE CASCADE ON UPDATE CASCADE,  
    PRIMARY KEY(`tgr_id`)  
) ENGINE = InnoDB;
```

## 2.16. guides

**Relational Model:**

guides(tg\_id, ts\_id, e\_id, status)

**Functional Dependencies**

tg\_id, ts\_id e\_id -> status

**Candidate Keys:**

{{(tg\_id, ts\_id, e\_id)}}

**Normal Form:**

BCNF

**Table Definition:**

```
CREATE TABLE `guides` (  
    `tg_id` INT NOT NULL,  
    `ts_id` INT NOT NULL,  
    `e_id` INT NOT NULL,  
    `status` VARCHAR(255) NOT NULL,
```



```

CONSTRAINT `fk_guides_tour_guide` FOREIGN
KEY(`tg_id`) REFERENCES `tour_guide`(`tg_id`) ON
DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `fk_guides_tour_section` FOREIGN
KEY(`ts_id`) REFERENCES `tour_section`(`ts_id`) ON
DELETE CASCADE ON UPDATE CASCADE,

CONSTRAINT `fk_guides_employee` FOREIGN KEY(`e_id`)
REFERENCES `employee`(`e_id`) ON DELETE CASCADE ON
UPDATE CASCADE

```

```
) ENGINE = InnoDB;
```

## 2.17. flight

### Relational Model:

flight(f\_id, capacity, ticket\_price, dept\_airport, dest\_airport, dept\_date, arrive\_date)

### Functional Dependencies

f\_id -> capacity, ticket\_price, dept\_airport, dest\_airport, dept\_date, arrive\_date

### Candidate Keys:

{f\_id}

### Normal Form:

BCNF

### Table Definition:

```

CREATE TABLE `flight` (
    `f_id` INT NOT NULL AUTO_INCREMENT,
    `capacity` INT NOT NULL,
    `ticket_price` INT NOT NULL,
    `dept_airport` CHAR(3) NOT NULL,
    `dest_airport` CHAR(3) NOT NULL,
    `dept_date` DATE NOT NULL,
    `arrive_date` DATE NOT NULL,
    CONSTRAINT `fk_flight_dept_airport` FOREIGN
    KEY(`dept_airport`) REFERENCES
    `airport`(`airport_code`) ON DELETE CASCADE ON
    UPDATE CASCADE,
    CONSTRAINT `fk_flight_dest_airport` FOREIGN
    KEY(`dest_airport`) REFERENCES
    `airport`(`airport_code`) ON DELETE CASCADE ON
    UPDATE CASCADE,
    PRIMARY KEY(`f_id`)) ENGINE = InnoDB;

```

## 2.18. flight\_reservation

### Relational Model:

flight\_reservation(fr\_id, f\_id, c\_id, number\_of\_passengers, date)

### Functional Dependencies

fr\_id -> f\_id, c\_id, number\_of\_passengers, date

### Candidate Keys:

{fr\_id}

### Normal Form:

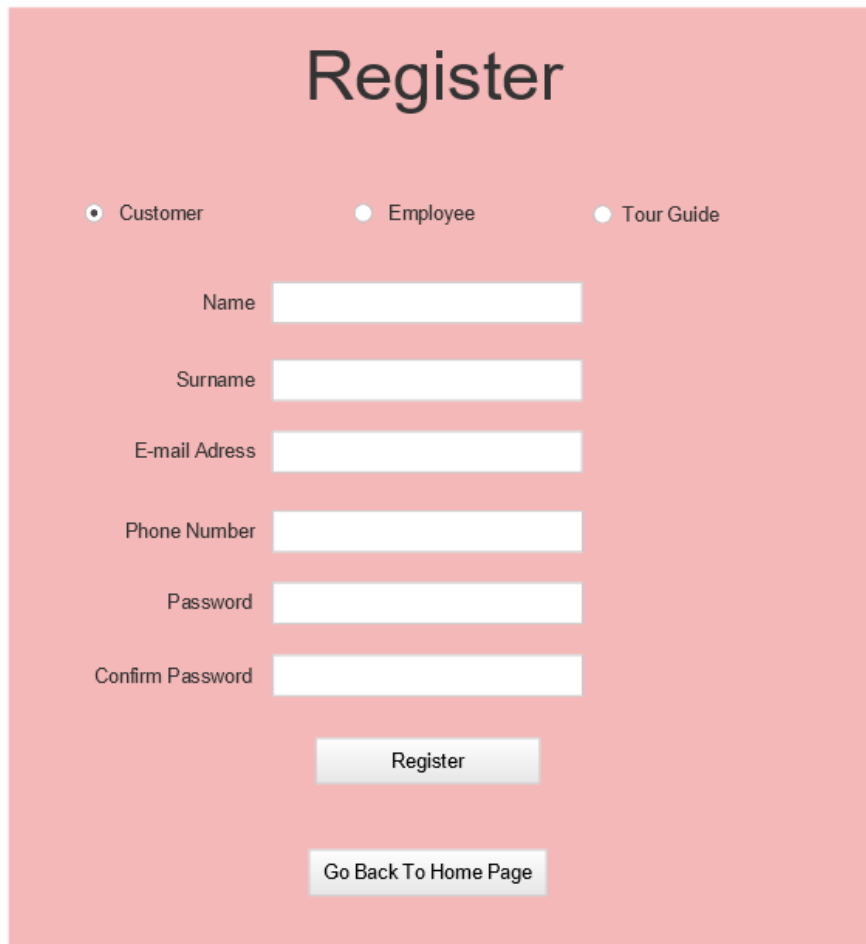
BCNF

### Table Definition:

```
CREATE TABLE `flight_reservation`(  
    `fr_id` INT NOT NULL AUTO_INCREMENT,  
    `f_id` INT NOT NULL,  
    `c_id` INT NOT NULL,  
    `number_of_passengers` INT NOT NULL,  
    `date` DATE NOT NULL,  
    CONSTRAINT `fk_fr_flight` FOREIGN KEY(`f_id`)  
    REFERENCES `flight`(`f_id`) ON DELETE CASCADE ON  
    UPDATE CASCADE,  
    CONSTRAINT `fk_fr_customer` FOREIGN KEY(`c_id`)  
    REFERENCES `customer`(`c_id`) ON DELETE CASCADE ON  
    UPDATE CASCADE,  
    PRIMARY KEY(`fr_id`)  
) ENGINE = InnoDB
```

## 3. User Interface Design and Corresponding SQL Statements

### 3.1 Customer Register



The image shows a user interface for a registration form titled "Register". The form is set against a light red background. At the top, there are three radio buttons for user roles: "Customer" (selected), "Employee", and "Tour Guide". Below these are six input fields stacked vertically, each with a label to its left: "Name", "Surname", "E-mail Adress", "Phone Number", "Password", and "Confirm Password". At the bottom of the form, there are two buttons: "Register" and "Go Back To Home Page".

```
INSERT INTO customer
VALUES (@name, @surname,
@emailAddress, @phone number,
@password)
```

## 3.2 Employee Register

# Register

☐ Customer☒ Employee☐ Tour Guide

Name

Surname

E-mail Address

Phone Number

Password

Confirm Password

Register

Go Back To Home Page

```
INSERT INTO employee
VALUES (@name, @surname,
@emailAddress, @phone number,
@password)
```

### 3.3 Tour Guide Register

# Register

☐ Customer☐ Employee☒ Tour Guide

Name

Surname

E-mail Address

Phone Number

Password

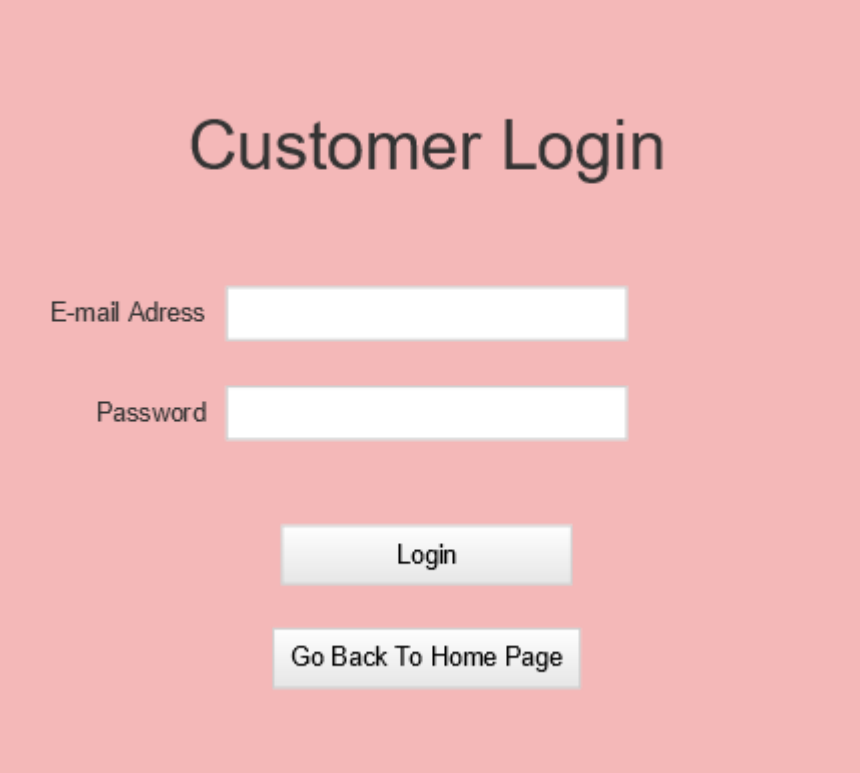
Confirm Password

Register Tour Guide

Go Back To Home Page

```
INSERT INTO tour_guide
VALUES (@name, @surname,
@emailAddress, @phone number,
@password)
```

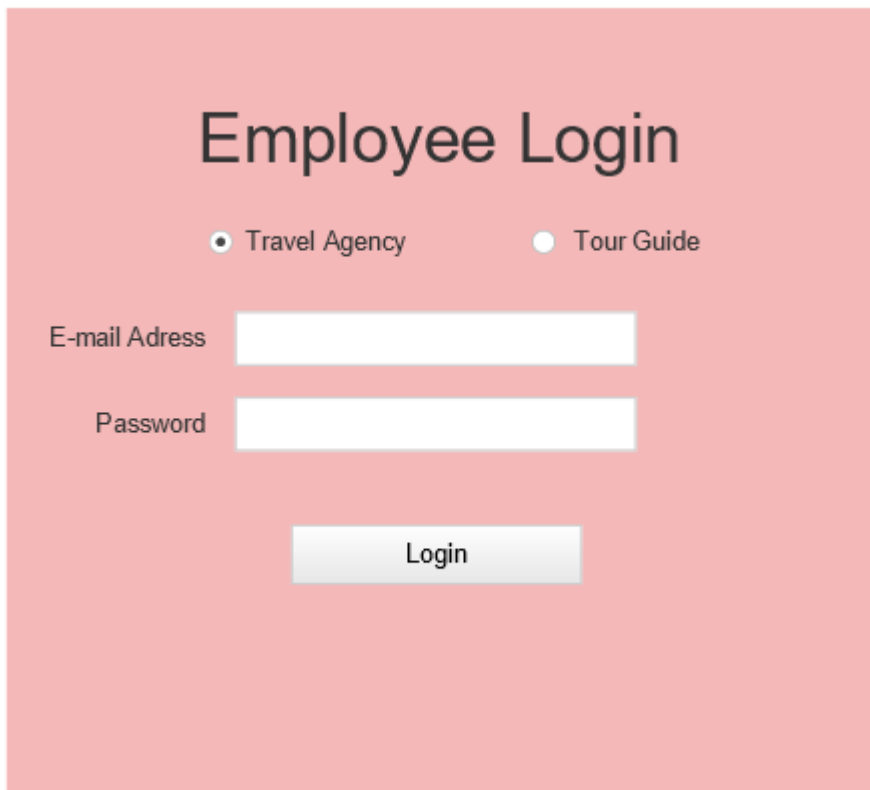
### 3.4 Customer Login



The image shows a 'Customer Login' form on a light red background. At the top, the title 'Customer Login' is centered in a large, dark font. Below the title, there are two input fields: 'E-mail Adress' and 'Password'. Each label is to the left of its respective white input box. Below the input fields, there are two buttons: a 'Login' button and a 'Go Back To Home Page' button, both with a light gray gradient and rounded corners.

```
SELECT * FROM customer
WHERE email=@email AND
password_hash = @password
```

### 3.5 Employee Login

A screenshot of an "Employee Login" form. The form has a light red background. At the top, the title "Employee Login" is centered in a large, dark font. Below the title, there are two radio buttons: "Travel Agency" (which is selected) and "Tour Guide". Below these, there are two input fields: "E-mail Address" and "Password". Both fields are empty. At the bottom, there is a "Login" button.

```
SELECT * FROM employee
WHERE email=@email AND
password_hash = @password
```

### 3.6 Tour Guide Login

## Employee Login

☐ Travel Agency☒ Tour Guide


E-mail Address

Password

Login

```
SELECT * FROM tour_guide
WHERE email=@email AND
password_hash = @password
```

### 3.7 Customer Dashboard



[Reserve a Flight](#)[My Past Tours](#)[Book a Tour](#)[Reserve a Hotel](#)[Profile](#)[Log Out](#)

Hello Customer X

Your Incoming Tours

| <u>Tour Name</u> | <u>Tour Date</u>                    | <u>Tour Guide</u> |                         |                        |
|------------------|-------------------------------------|-------------------|-------------------------|------------------------|
| 1- TourName1     | 10 November 2021 - 13 November 2021 | PersonX           | <a href="#">Details</a> | <a href="#">Cancel</a> |
| 2- TourName2     | 12 November 2021 - 14 November 2021 | PersonY           | <a href="#">Details</a> | <a href="#">Cancel</a> |

Your Hotel Reservations

| <u>Hotel Name</u> | <u>Reservation Date</u>             | <u>Room Number</u> |                         |                        |
|-------------------|-------------------------------------|--------------------|-------------------------|------------------------|
| 1- HotelName1     | 10 November 2021 - 16 November 2021 | Room 2             | <a href="#">Details</a> | <a href="#">Cancel</a> |



For the *Incoming Tours*:

```
SELECT start_date, end_date, first_name, last_name
FROM customer_reservation, tour_section, guides, tour_guide
WHERE customer_reservation.cid = @cid AND start_date > NOW()
AND guides.tour_guide_id = tour_guide_id AND
customer_reservation.section_id = section_id
```

For *Hotel Reservations*:

```
SELECT hotel.name, start_date, end_date, r_id
FROM books, contains, hotel
WHERE books.cid = @cid AND
contains.roomID = books.RoomID
```

## 3.8 Customer Past Tours




### Your Past Tours

| <u>Tour Name</u> | <u>Tour Date</u>                    | <u>Tour Guide</u> |                    |         |      |
|------------------|-------------------------------------|-------------------|--------------------|---------|------|
| 1- TourName1     | 10 November 2021 - 13 November 2021 | PersonX           | Tour Guide Profile | Details | Rate |
| 2- TourName2     | 12 November 2021 - 14 November 2021 | PersonY           | Tour Guide Profile | Details | Rate |

```
SELECT start_date, end_date, first_name, last_name
FROM customer_reservation, tour_section, guides, tour_guide
WHERE customer_reservation.cid = @cid AND start_date < NOW() AND
guides.tour_guide_id = tour_guide_id AND
customer_reservation.section_id = section_id
```

## 3.9 Customer Rating



[Reserve a Flight](#)[My Past Tours](#)[Book a Tour](#)[Reserve a Hotel](#)[Profile](#)[Log Out](#)

You Are Rating Tour X with Tour Guide Y

Tour Rating:

Tour Guide Rating:

Tour Comment


Tour Guide review

INSERT INTO customer\_review

VALUES (Tour Rating, Tour Comment,


Tour Guide Rating, Tour Guide Review, @cid, @section\_id)

## 3.10 Customer Profile



[Reserve a Flight](#)[My Past Tours](#)[Book a Tour](#)[Reserve a Hotel](#)[Profile](#)[Log Out](#)

Update Information

  
[Update Profile Picture](#)

Email:

Old Password:

New Password:

Confirm New Password:

For *Email Update*:

```
UPDATE customer SET email = @email WHERE cid = @cid
```

For *Password Update*:

```
UPDATE customer SET password_hash = @new_Password
```

```
WHERE cid = @cid AND
```

```
password_hash = @oldPassword
```

## 3.11 Customer Hotel Reservation Details


The screenshot shows the LEVART website interface. At the top, there is a green navigation bar with the LEVART logo on the left and a series of buttons on the right: "Reserve a Flight", "My Past Tours", "Book a Tour", "Reserve a Hotel", "Profile", and "Log Out". Below the navigation bar, the main content area is titled "Details of Your reservation in Hotel X". This section contains a light gray box with the following reservation details:

- Reservation Hotel: Hotel Name 1
- Reservation Date: 13 November 2021 - 19 November 2021
- Hotel Address: In city C
- Room Number(s): Room 132, Room 133, Room 134
- Extra Facilities: Gym, Swimming Pool, Spa

At the bottom of the reservation details box, there are two buttons: "Cancel" and "Make Another Reservation".

```
SELECT hotel.name, start_date, end_date, address, r_id FROM  
books, contains, hotel WHERE  
books.cid = @cid AND  
books.r_id = contains.r_id AND  
hotel.h_id = contains.h_id
```

## 3.12 Customer Reserve a Hotel



[Reserve a Flight](#)
[My Past Tours](#)
[Book a Tour](#)
[Reserve A Hotel](#)
[Profile](#)
[Log Out](#)

Available Hotels

| Hotel Name    | Hotel Location | # of Available Rooms | Room Number                     | Start-End date |                              |
|---------------|----------------|----------------------|---------------------------------|----------------|------------------------------|
| 1- HotelName1 | Location 1     | 7                    | Room 1 - 2 People - 150\$       |                | <a href="#">Reserve Room</a> |
| 2- HotelName2 | Location 2     | 3                    | Room 5 - 3 People - 230\$       |                | <a href="#">Reserve Room</a> |
| 3- HotelName3 | Location 3     | 4                    | Room 13 - 2 People King - 320\$ |                | <a href="#">Reserve Room</a> |
| 4- HotelName4 | Location 4     | 2                    | Room 9 - 1 People - 100\$       |                | <a href="#">Reserve Room</a> |

INSERT INTO books

VALUES (@cid, @eid, @r\_id, @start\_date, @end\_date, type, status )

## 3.13 Customer Book a Tour



[Reserve a Flight](#)
[My Past Tours](#)
[Book A Tour](#)
[Reserve a Hotel](#)
[Profile](#)
[Log Out](#)

Available Tours

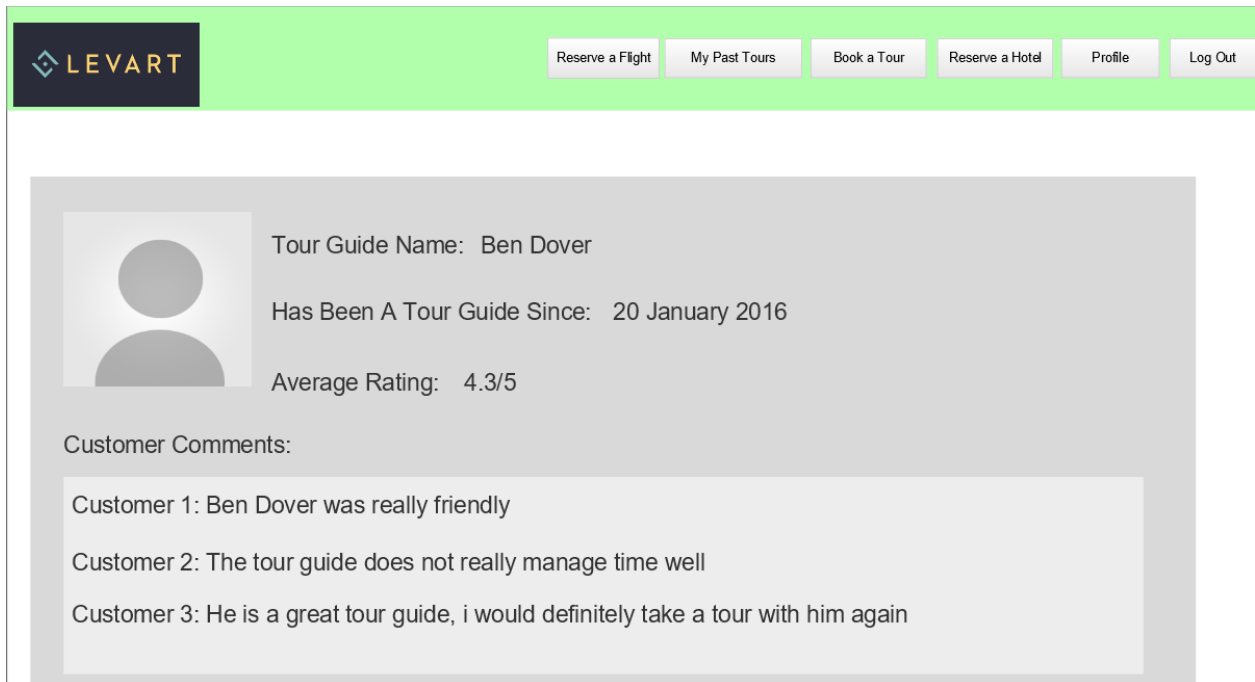
| Tour Name    | Tour Date                           | Depart Location | Tour Guide |   |
|--------------|-------------------------------------|-----------------|------------|---|
| 1- TourName1 | 10 November 2021 - 13 November 2021 | Location1       | PersonX    | <a href="#">Tour Guide Profile</a> <a href="#">Details</a> <a href="#">Reserve</a> <a href="#">Reserve Extra Activity</a> |
| 2- TourName2 | 12 November 2021 - 14 November 2021 | Location2       | PersonY    | <a href="#">Tour Guide Profile</a> <a href="#">Details</a> <a href="#">Reserve</a> <a href="#">Reserve Extra Activity</a> |

```

SELECT start_date, end_date, tour_guide.name FROM
guides, tour_section, tour_guide WHERE
guides.ts_id = tour_section.ts_id AND
guides.tg_id = tour_guide.tg_id AND
start_date > NOW()

```

## 3.14 Customer View Tour Guide Profile




For *Tour Information*:

```
SELECT tour_guide.name, created_at, AVG(guide_rate) FROM
guides, customer_review WHERE guides.ts_id = customer_review.ts_id AND
guides.tg_id = @tg_id
```

For *Customer Comments*:

```
SELECT tour_comment FROM
guides, customer_review WHERE guides.ts_id = customer_review.ts_id AND
guides.tg_id = @tg_id
```

## 3.15 Customer Reserve Extra Activity




[Reserve a Flight](#)[My Past Tours](#)[Book a Tour](#)[Reserve a Hotel](#)[Profile](#)[Log Out](#)

### Extra Activities

| <u>Activity Name</u> | <u>Activity Date</u>  | <u>Activity Time</u> |   |
|----------------------|-----------------------|----------------------|---|
| 1- Visit Place X     | 9th of September 2021 | 7:00-9:00            | <a href="#">Reserve This Extra Activity</a> |
| 1- Visit Place Y     | 9th of September 2021 | 14:00-16:00          | <a href="#">Reserve This Extra Activity</a> |
| 1- Visit Place Z     | 9th of September 2021 | 16:30-17:00          | <a href="#">Reserve This Extra Activity</a> |

INSERT INTO activity\_added VALUES (@cid, @a\_id, @ts\_id)

## 3.16 Customer Tour Details



[Reserve a Flight](#)[My Past Tours](#)[Book A Tour](#)[Reserve a Hotel](#)[Profile](#)[Log Out](#)

### Available Extra Activities

| <u>Activity Name</u> | <u>Activity Date</u>   | <u>Activity Time</u> |
|----------------------|------------------------|----------------------|
| 1- Visit Place X     | 9th of September 2021  | 7:00-9:00            |
| 2- Visit Place Y     | 9th of September 2021  | 14:00-16:00          |
| 3- Visit Place Z     | 10th of September 2021 | 16:30-17:00          |

### Tour Basic Activities

| <u>Activity Name</u> | <u>Activity Date</u>   | <u>Activity Time</u> |
|----------------------|------------------------|----------------------|
| 1- Museum            | 8th of September 2021  | 10:00-12:00          |
| 2- Beach             | 8th of September 2021  | 14:00-16:00          |
| 3- Visit Place A     | 10th of September 2021 | 11:30-14:00          |
| 4- Visit Place B     | 11th of September 2021 | 12:00-15:00          |

For *Tour Basic Activity*:

```
SELECT name, date, location FROM
activity, offers WHERE
activity.a_id = offers.a_id AND
type = "basic"
```

For *Available Extra Activity*:

```
SELECT name, date, location FROM
```

activity, offers WHERE  
 activity.a\_id = offers.a\_id AND  
 type = "extra"

## 3.17 Customer Reserve Flight




| <u>Flight Number</u> | <u>Depart Location</u> | <u>Arrival Location</u> | <u>Departs on</u> | <u>Arrives on</u> | <u>Price Per Person</u> | <u>Number of Passenger</u> |                  |
|----------------------|------------------------|-------------------------|-------------------|-------------------|-------------------------|----------------------------|------------------|
| 1- Flight1           | Esenboga Airport       | Istanbul Airport        | 24/11/2021        | 24/11/2021        | 50\$                    | <input type="text"/>       | Book This Flight |
| 2- Flight2           | Frankfurt Airport      | Istanbul Airport        | 26/11/2021        | 26/11/2021        | 300\$                   | <input type="text"/>       | Book This Flight |
| 3- Flight3           | Istanbul Airport       | London Airport          | 27/11/2021        | 28/11/2021        | 500\$                   | <input type="text"/>       | Book This Flight |
| 4- Flight4           | Esenboga Airport       | Moscow Airport          | 24/12/2021        | 24/12/2021        | 240\$                   | <input type="text"/>       | Book This Flight |

SELECT f\_id, Airport.name, departs.date, arrives.date, price FROM  
 flight, departs, arrives, airport WHERE  
 flight.f\_id = departs.f\_id AND  
 flight.f\_id = arrives.f\_id AND  
 airport.airport\_code = departs.airport\_code AND  
 airport.airport\_code = arrives.airport\_code

//\*\*\*\*\*

## 3.18 Employee Dashboard



Pending Hotel ReservationsPending ToursCreate New TourRegister HotelProfileLog Out

Hello Employee X of company Y

### Incoming Tours

| <u>Tour Name</u> | <u>Tour Date</u>                    | <u>Tour Guide</u> |   |
|------------------|-------------------------------------|-------------------|---|
| 1- TourName1     | 10 November 2021 - 13 November 2021 | PersonX           | <div>Add activityAdd Extra ActivityAssign Tour GuideEdit Tour InformationDetailsCancel Tour</div> |
| 2- TourName2     | 12 November 2021 - 14 November 2021 | PersonY           | <div>Add activityAdd Extra ActivityAssign Tour GuideEdit Tour InformationDetailsCancel Tour</div> |

### Available Hotels

| <u>Hotel Name</u> | <u>Reservation Date</u>             | <u>Room Number</u> |  |
|-------------------|-------------------------------------|--------------------|--|
| 1- HotelName1     | 10 November 2021 - 16 November 2021 | Room 2             | <div>EditDetailsCancel Reservation</div> |

For incoming tours:


```
SELECT type, start_date, end_date, tour_guide.first_name, tour_guide.last_name
FROM tour, tour_section, guides, tour_guide
WHERE start_date > NOW() AND
guides.tour_guide_id = tour_guide_id AND
customer_reservation.section_id = section_id AND tour.t_id = tour_section.t_id AND
guides.ts_id = tour_section.ts_id
```

For available hotels:

```
SELECT hotel.name, start_date, end_date, r_id
FROM books, contains, hotel, room
WHERE contains.roomID = books.RoomID AND hotel.h_id = room.h_id
AND books.r_id = room.r_id
```



## 3.19 Employee Create New Tour

Pending Hotel ReservationsPending ToursCreate New TourRegister HotelProfileLog Out

### Create New Tour

Tour Name:

Start date:

End date:

Depart Location:


Create Tour

INSERT INTO tours VALUES (@t\_id, @place, @type )

INSERT INTO tour\_section VALUES (@ts\_id, @start\_date, @end\_date, @t\_id )

INSERT INTO guides VALUES (@employee\_id, @ts\_id, status)

## 3.20 Employee Create New Activity

Pending Hotel ReservationsPending ToursCreate New TourRegister HotelProfileLog Out

### Create New Activity For Tour X

Activity Name:

Activity location:

Activity date:


Activity hour:

Create Activity

INSERT INTO activity VALUES (@a\_id, name, location, date, type )

INSERT INTO tour\_activity VALUES (@ts\_id, @a\_id)

## 3.21 Employee Create New Extra Activity

Pending Hotel ReservationsPending ToursCreate New TourRegister HotelProfileLog Out

Create New Extra Activity For Tour X

Extra Activity Name:

Extra Activity location:

Extra Activity date: 11/17/2021

Extra Activity hour:

Create Extra Activity

INSERT INTO activity VALUES (@a\_id, name, location, date, type )

INSERT INTO tour\_activity VALUES (@ts\_id, @a\_id)

## 3.22 Employee Assign Tour Guide




[Pending Hotel Reservations](#) [Pending Tours](#) [Create New Tour](#) [Register Hotel](#) [Profile](#) [Log Out](#)

### Assign Tour Guide For Tour X

Tour guide:

```
UPDATE guides SET @tg_id, status = 1 FROM tour_section WHERE tour_section.ts_id = @ts_id
```

## 3.23 Edit Tour Information



[Pending Hotel Reservations](#) [Pending Tours](#) [Create New Tour](#) [Register Hotel](#) [Profile](#) [Log Out](#)

### Edit Tour Information For Tour X

Tour Name:

Start date:

End date:

Depart Location:

```
UPDATE tours SET place=@place, type=@type WHERE t_id=@t_id
```

```
UPDATE tour_section SET start_date=@start_date, end_date=@end_date WHERE
ts_id=@ts_id
```

## 3.24 Employee Register Hotel

```
INSERT INTO hotel VALUES (@h_id, name, phone, address, hc_id )
```

```
INSERT INTO room VALUES (@r_id, price, capacity, type, h_id )
```

## 3.25 Employee Pending Tours

| <u>Tour Name</u> | <u>Tour Date</u>                    | <u>Customer name</u> | <u>Tour Guide</u> | <u>Tour Guide Rating</u> |                        |
|------------------|-------------------------------------|----------------------|-------------------|--------------------------|------------------------|
| 1- TourName1     | 10 November 2021 - 13 November 2021 | PersonA              | PersonX           | 4.2/5                    | Accept Decline Details |
| 2- TourName2     | 12 November 2021 - 14 November 2021 | PersonB              | PersonY           | 3.8/5                    | Accept Decline Details |


```
SELECT tour.name, start_date, end_date, tour_guide.name, customer.name
FROM guides, tour_section, tour_guide, customer, tour, reservation, customer
```

```

WHERE
guides.ts_id = tour_section.ts_id AND
guides.tg_id = tour_guide.tg_id AND
tour.t_id = tour_section.ts_id AND
start_date > NOW() AND
customer.c_id = reservation.c_id AND
reservation.ts_id = tour_section.ts_id AND
status = "pending"

```

## 3.26 Employee Pending Hotel Reservations


| <div>  <div> Pending Hotel Reservations Pending Tours Create New Tour Register Hotel Profile Log Out </div> </div> |                       |                             |                                 |                      |   |
|---|-----------------------|-----------------------------|---------------------------------|----------------------|---|
| Pending Hotel Reservations  |                       |                             |                                 |                      |   |
| <u>Hotel Name</u>   | <u>Hotel Location</u> | <u># of Available Rooms</u> | <u>Room Number</u>              | <u>Customer Name</u> |   |
| 1- HotelName1   | Location 1            | 7                           | Room 1 - 2 People - 150\$       | PersonA              | <input type="button" value="Accept"/> <input type="button" value="Decline"/> <input type="button" value="Details"/> |
| 2- HotelName2   | Location 2            | 3                           | Room 5 - 3 People - 230\$       | PersonB              | <input type="button" value="Accept"/> <input type="button" value="Decline"/> <input type="button" value="Details"/> |
| 3- HotelName3   | Location 3            | 4                           | Room 13 - 2 People King - 320\$ | PersonC              | <input type="button" value="Accept"/> <input type="button" value="Decline"/> <input type="button" value="Details"/> |
| 4- HotelName4   | Location 4            | 2                           | Room 9 - 1 Person - 100\$       | PersonD              | <input type="button" value="Accept"/> <input type="button" value="Decline"/> <input type="button" value="Details"/> |

```

SELECT hotel.name, start_date, end_date, address, r_id, c_id, customer.name
FROM books, contains, hotel, room, customer
WHERE
books.r_id = contains.r_id AND
hotel.h_id = contains.h_id AND room.h_id = hotel.h_id AND
start_date > NOW() AND
customer.c_id = books.c_id AND
status = "pending"


```

## 3.27 Employee Profile



[Pending Hotel Reservations](#)[Pending Tours](#)[Create New Tour](#)[Register Hotel](#)[Profile](#)[Log Out](#)

### Update Information

[Update Profile Picture](#)


Email:   
[Update Email](#)

Old Password:   
New Password:   
Confirm New Password:   
[Update Password](#)

UPDATE employee SET email = @email WHERE e\_id = @e\_id

UPDATE employee SET password\_hash = @new\_Password WHERE e\_id = @e\_id

## 3.28 Tour Guide Dashboard



[Available Tours](#)[Profile](#)[Log Out](#)

Hello Guide X

Completed Tours

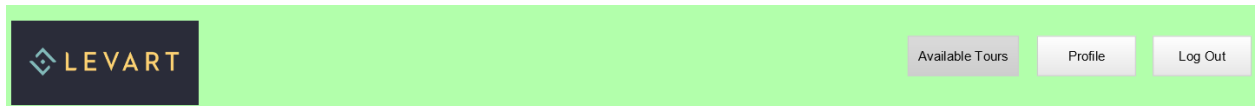
|    | <u>Tour Name</u> | <u>Tour Date</u>                    |   |
|----|------------------|-------------------------------------|---|
| 1- | TourName1        | 10 November 2021 - 13 November 2021 | <a href="#">Details</a> <a href="#">Give Feedback</a> |
| 2- | TourName2        | 12 November 2021 - 14 November 2021 | <a href="#">Details</a> <a href="#">Give Feedback</a> |

```
SELECT tour.type, start_date, end_date
FROM tour_section, guides, tour_guide, tour
WHERE end_date < NOW()
```

AND guides.tour\_guide\_id = tour\_guide\_id

AND tour.t\_id = tour\_section.ts\_id

## 3.29 Tour Guide Available Tours

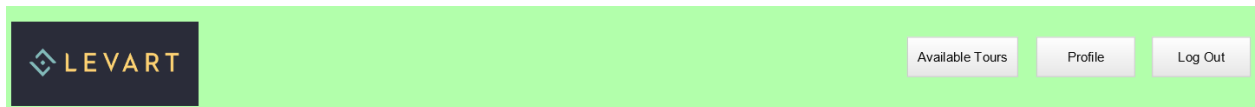


### Available Tours

| <u>Tour Name</u> | <u>Tour Date</u>                    |   |
|------------------|-------------------------------------|---|
| 1- TourName1     | 10 November 2021 - 13 November 2021 | <button>Details</button> <button>Accept</button> <button>Decline</button> |
| 2- TourName2     | 12 November 2021 - 14 November 2021 | <button>Details</button> <button>Accept</button> <button>Decline</button> |

```
SELECT tour.name, start_date, end_date
FROM tour_section, guides, tour_guide, tour
WHERE start_date > NOW()
AND guides.tour_guide_id = tour_guide_id
AND tour.t_id = tour_section.ts_id AND
guides.status = "pending"
```

### 3.30 Tour Guide Feedback Form



You Are Giving Feedback to Tour X

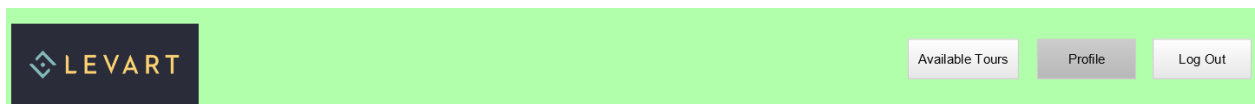
Tour Feedback

The tour was really nice but the weather was no favorable


Submit

INSERT INTO tour\_guide\_review VALUES (@tgr\_id, comment, @tg\_id, @ts\_id)

### 3.31 Tour Guide Profile



Update Information



[Update Profile Picture](#)

Email:

Old Password:

New Password:

Confirm New Password:

UPDATE tour\_guide SET email = @email WHERE e\_id = @e\_id



```
UPDATE tour_guide SET password_hash = @new_Password WHERE e_id = @e_id
```