

## Flight Reservation Project

### Introduction

In this project, you will design and implement a relational database system to support the operations of an online travel reservation system. Your first task is to design an ER/EER schema diagram. Then, you will have to map the ER/EER schema diagram into a relational database schema and implement it on MySQL relational DBMS preferable. Finally, you will load data into your relational database, write some queries, as well as a program to enter the query parameters and display the results.

You are allowed to work in teams of up to three students!

### Project Specification

As you probably know, there are a multitude of online travel reservation systems on the web. Some popular ones are expedia, skyscanner and kayak. I suggest that you visit these web sites to get an understanding of the look-and-feel of a travel web site and how such a system is supposed to function

The basic idea behind your on-line travel reservation system is that it will allow customers to use your application to browse/search the contents of your database (at least that part you want the customer to see) and to make flight reservations. Your application should allow users to make both domestic and international reservations. It should also allow users to query the database for available direct flights between a pair of cities/airports for a given date.

### 1. System Users

The users of your system will be the customers (passengers) that use your system to make a flight reservation, customer representatives who provide customer-related services, and the site's admin. You should assume that the computer knowledge of the users is limited, and thus your system must be easy to access and operate.

The data items required for the travel reservation database can be classified into six categories: *airlines*, *aircrafts*, *airports*, *flights*, *tickets*, *customers* and *employees*.

**The above classification does not imply any particular table arrangement.** You are responsible for arranging the data items into tables, determining the relationships among tables and identifying the key attributes. Finally, you should specify and enforce integrity constraints on the data, including referential integrity constraints.

You will first create an ER diagram of your online travel reservation system before developing your relational model according to the functional requirements described below.

### 2. Functional Requirements

Every airline company owns a number of aircrafts and it is associated with a number of airports from where it operates. Each airline has a two-letter ID from which it is being identified uniquely. For example, the ID for American Airlines is AA, and the ID for United Airlines is UA. Similarly, each airport has a three-letter ID. For example, EWR, LGA, and JFK are well known local airport codes.

A flight is operated by an airline and a specific aircraft and operates on a given set of *days of the week* (e.g. every Monday, Wednesday). Flights can either be *domestic* or *international*. For every flight, it must be recorded its flight number (unique only within that airline), the departure and destination airports, as well as the departure and arrival time.

Customers should be able to make reservations. Customers should first be able to search for specific flights by providing information about the departure and arrival airport as well as the date they wish to fly. The flight ticket can either be one-way, or round-trip and they should be able to set if they are flexible about flight dates (+- 3 days).

A flight ticket has a unique number and is for just a single passenger. Each ticket is associated with a sequence of flights. For example, a ticket might be associated with just one flight if it is one-way or with 2 flights if it is round-trip. Each ticket must include all the associated flights and include information for the departure and arrival airport, flight numbers (along with its airline), departure date and time, and class (economy/business/first). It also has the following attributes: total fare, and date and time when ticket was purchased. In case the class of the ticket is economy, the customer should not be able to

change/cancel their ticket unless a fee is paid. For business/first class, customers should be able to change their ticket with no fee.

A customer may partake in any number of flight transactions and she/he is associated with one member account which includes a reservation portfolio, indicating all the flight history held in this account (past flights and upcoming).

Your online reservation system should have the following functionality:

#### **Admin-Level Functionality**

The admin should be able to:

- Add, Edit and Delete information for a customer representative or customer
- Obtain a sales report for a particular month
- Produce a list of reservations by flight number or by customer name
- Produce a summary listing of revenue generated by a particular flight, airline or customer
- Determine which customer generated most total revenue
- Produce a list of most active flights (most tickets sold)
- Produce a list of all flights for a given airport

#### **Customer-Representative-Level Functionality**

Customer Representatives should be thought of as reservation agents and should be able to:

- Make flight reservations on behalf of users
- Edit flight reservations for a customer
- Add, Edit, Delete information for aircrafts, airports and flights

#### **Customer-Level Functionality**

Customers should be thought of as online airline ticket buyers and should be able to easily navigate through your travel reservation system and buy flight tickets. In particular, they should be able to search and make the following types of reservations:

- One-Way
- Round-Trip Flexible
- Date

A customer should also be able to:

- sort flights by different criteria (price, duration, take-off time, landing time)
- filter the list of flights by various criteria (price, duration, airline)
- cancel their flight reservations (if it is business or first class)
- view all the past reservations with their details
- view all the upcoming reservations with their details

### **3. User Access Control**

Your database system should provide controlled access to the data by distinguishing between the different types of users: admin, customer representatives, and customers.

- Customer Representatives should not be able to perform manager-level transactions;
- A customer should not be allowed access to other customers' account information, or to any employee information

#### **Important Notes:**

- (1) This project can be done individually, or in up to three-person group. If it is a group submission, you will all receive the same grade.
- (2) Copying from other students, groups or internet is not permitted and will result in a grade of zero for the entire project!

#### **Project weight:**

- Phase 1 (10%)
- Phase 2 (12%)
- Phase 3 (13%)

## Phase 1-Submission:

You will first design an ER/EER schema diagram based on the database requirements specified above by using the notation covered in your book. Don't forget to design the cardinality ratios and the participation between entities and relationships. Also, as part of this assignment, you should identify any missing or incomplete requirements and explicitly state them. You should also explicitly state any assumptions you made that were not part of the requirements listed above.

Next, you need to convert/map your ER/EER schema diagram to a relational database schema by using the notation covered in your book. Also, don't forget to denote primary and foreign keys. You should state the choices you made during the EER-to-Relational mapping and the reasons for your choices.

You can draw your ER/EER schema diagram and the relational database schema by hand (neatly) and scan it, take *quality* pictures of your drawn diagram, or use a drawing tool such as PowerPoint, Keynote, or draw.io. to create a computer image file or use any available database conceptual design tool.

For this phase, you need to submit only **ONE** document file (.pdf or .docx) with:

- your ER/EER schema diagram and an explanation of your design choices, [45 points]
- your Relational Database Schema and an explanation of your design choices, [45 points]
- and the honor code. [10 points]

Make sure you follow your book's notation and double-check the following:

- cardinality ratios per relation - relationship,
- participation constraints,
- primary keys and foreign keys,
- to not have overlapping lines in your schemas,
- missing relationship between relations,
- same names for relations or relationships, and
- arrows pointing in the correct direction.

Name your file as **Team##.docx**. Make sure that your document looks professional. For example, you could have a cover page, an introduction (a few lines what is about), the mini-world description, references, etc.

**All students are expected to include and sign the honor code on the first page of each of their assignments. Failing to do so will cost 10 points.**

## HONOR CODE

I pledge, on my honor, to uphold UT Arlington's tradition of academic integrity, a tradition that values hard work and honest effort in the pursuit of academic excellence.

I promise that I will submit only work that I personally create or that I contribute to group collaborations, and I will appropriately reference any work from other sources. I will follow the highest standards of integrity and uphold the spirit of the Honor Code.

**Late policy: -5 points per hour**