CS 166 Project Description

1 Introduction

In this project, we will model and build an airline management system. We will use this system to track information about different airlines, the planes they own, the maintenance of those planes, the pilots they employ and the trips that the pilots make, as well as information about the customers that use the airline services.

The project consists of three phases: (i) Requirement Analysis using the ER-model, (ii) Relational schema design, and (iii) Implementation.

Projects can be done in teams of two. Choose your partner wisely because the final evaluation is based on the group performance! In your report (for each phase) explicitly enumerate the tasks that each member of your group was responsible for and how you collaborated. If one of the group members does most of the work, the grade will be proportional to the effort. If you are not able to find a partner, one will be assigned to you at random. Please e-mail the TA and myself immediately if you need help finding a partner.

Phase 1: ER Design

In the first phase, you are given a set of requirements for your database in section 2 of this document, you need to design a logical model of your database using the ER-model. Note that we are giving you a set of requirements, and not the complete set of attributes each entity set will hold. From the description, you must first identify the entity sets to be included in your model, and the attributes for each entity set that will enable you to answer the given queries. Following these guidelines, you should generate an ER-diagram using your favorite diagram editing software. Use only the basic ER model that includes entities, relationships and attributes. Do not forget to indicate any key and participation constraints. Also, make sure to include additional documentation describing the assumptions that you made during the design process. You have to submit all your files via iLearn on the due date. Your submission should be a single zip file. Make sure to check that everything is included in your submission and it can be uncompressed without any errors.

You can make reasonable assumptions on your design, as long as:

- That you state them clearly in the documentation for this phase.
- They do not contradict the system requirements analysis we provide.

The due date for this phase is: **Feb. 21st at midnight**.

Phase 2: Relational Schema Design

In this phase, we will provide you with an ER-diagram that is a solution to Phase 1 (so that the whole class will proceed with the same design). This final ER-diagram will be the starting point for the second phase, which involves the creation of the relational schema.

Your task in this phase will be to translate the provided ER design to a PostgreSQL relational database schema. The database schema will be in a form of a single executable SQL script (*.sql file with SQL statements). You must submit this SQL script via iLearn on the due date. The SQL script should include the necessary drop statements at the beginning so it is easy to test. Check how the drop statement works and consider using the IF EXISTS statements where necessary.

In this phase, we will evaluate you for the correctness and completeness of your relational schema. You may find some constraints in the model and/or system requirement analysis that are not possible to represent or enforce in the relational schema. You may specify all these issues in the documentation for this phase. Your submission should be a single compressed file, containing all the aforementioned files.

The due date for this phase is: **Feb. 28th at midnight**.

Phase 3: Implementation

Your tasks in this phase will be:

- Develop a client application using the Java Database Connector (JDBC) for psql.
- Use the client application to support specific functionality and queries for your online booking system.

In this phase, we will provide you with a create.sql script that recreates the relational schema of phase 2. You will use this schema to test and demo your application to us. Additionally, we will give you a collection of .csv files containing dummy data that are compatible with the provided relational schema. You will have to create your own .sql scripts to insert the data from the given .csv files into the database.

Finally, we will give you a skeleton code for the client application. The code will be in Java and will contain some basic functionality that will help you to communicate with the database and issue various .sql statements.

This phase of the project is challenging, therefore we advise you to start early and allocate at least 25 hours per person to get it finished. Make sure to consider all possible scenarios for the client application and try to handle any exceptions that arise during the regular operation of your application.

For this phase you will be evaluated based on the system requirements. Your GUI and source code will also be taken into consideration in your final evaluation. Groups that implement systems with user-friendly interfaces, extra functionalities will receive an extra credit. A final report about your system along with its source code has to be sent to your TA before the due date. You have to submit the documentation and final source code on iLearn.

The due date for this phase is: **March 13th at midnight**.

1.1 Grading

Your contribution to this project will be graded based on the following characteristics:

1. Phase 1 (30%)
   - Conceptual Design (ER Diagram)
   - You must submit your solutions on iLearn.
2. Phase 2 (10%)
   - Logical DB Design (Relational Database Schema)
   - You must submit your solutions on iLearn.
3. Phase 3 (60%)
   - Documentation of the project including details about your assumptions (10%).
   - Implementation of SQL queries in the Client Application (30%).
   - Physical DB Design (DB performance tuning indexes) (10%).
   - Extra credit for good GUI design and interface, any dataset or schema changes/extensions, etc. (20%).

2 Requirement Analysis

You will design an airline database that serves the needs of Airline Managers and Customers. Each of these types of individuals needs access to the following information:

Airline Management:

1. Given a flight number, get the flight's schedule for the week
   - A flight may be scheduled on multiple days in a week
2. Given a flight and a date, get (1) the number of seats still available and (2) number of seats sold
3. Given a flight and date, find whether (1) the flight departed on time, and (2) arrived on time

4. Given a date, get all flights scheduled on that day
5. Given a flight and date, get a list of passengers who (1) made reservations, (2) are on the waiting list, (3) actually flew on the flight (for flights already completed)
6. Given a reservation number, retrieve information about the travelers under that number
    - First & Last Name, Gender, Date of birth, Address, Phone number, Zip Code
7. Given a plane number, get its make, model, age, and last repair date
8. Given a maintenance technician ID, list all repairs made by that person
9. Given a plane ID and a date range, list all the dates and the codes for repairs performed

Customers:

1. Given a destination and departure city, find all flights on a given date.
    - Must return departure and arrival time, number of stops scheduled, and on-time record (as a percentage)
2. Given a flight number, find the ticket cost.
3. Given a flight number, find the airplane type (make and model)
4. Make a reservation for a flight
    - Get on the waitlist for a flight if the flight is full

Maintenance Staff:

1. Given a plane ID and a date range, list all the dates and the codes for repairs performed
2. Given a pilot ID, list all maintenance requests made by the pilot
3. After each repair, make an entry showing plane ID, repair code, and date of repair

Pilots:

1. Make maintenance request listing plane ID, repair code requested, and date of request