

CS 166 Project Description & Phase 1 Requirements

September 27, 2022

1 Introduction

In this project, we will model and build an Online Shopping System for a Retail Shop. The system will be used to track information about different stores of the Retail Shop, availability of products in those stores, stock requests from the managers, customer order history, delivery and billing information, etc. The project consists of three phases: (i) Requirement Analysis using the ER-model, (ii) Relational schema design, and (iii) Implementation.

Projects will 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). Email your TA and the instructor asap if you need help finding a partner.

Phase 1: ER Design

In the first phase we will do the requirement analysis using the ER-model. All the requirements can be obtained from Section 3 of this document. At the end of this phase you should generate an ER-diagram. For the ER-diagram, you can use the erdplus.com graphical editor, and you should finally create a PDF file using the ER notations from the labs. 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 compressed into a single file through elearn(Canvas) by the deadline. In this phase we will evaluate the correctness of your ER-diagram. 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:

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

The due date for this phase is: **Friday, October 14, 11:59PM**.

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 provided 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-diagram 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 eLearn 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 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 on eLearn as a single compressed file, containing all the aforementioned files.

The due date for this phase is: **Friday October 21, 11:59PM.**

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 shopping 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 or extra functionalities will receive an extra credit. Your group needs to submit: (1) a final report in a PDF file about your system, and (2) a zip file containing the complete source code and a text file with the proper instructions to run your code. You **MUST** submit them within the due date via eLearn (no exceptions).

The due date for this phase is: **Friday, December 2nd, 11:59PM.**

After the submission of this phase, there will be a ****required** DEMO session where BOTH of the students in a group MUST be present and demonstrate the project to the TAs.** Your phase 3 project will not be graded without a demo session. Demo sessions will be 15 minutes per group; tentative dates are: Dec. 5th and Dec 6th.

2 Grading

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

- Phase 1 (30%)
 - Conceptual Design (ER Diagram)
 - You must submit your solutions on eLearn.
- Phase 2 (10%)
 - Logical DB Design (Relational Database Schema)
 - You must submit your solutions on eLearn.
- 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%)
 - Client-Application Development (a console application in Java is expected) (10%)

- Extra credit for good GUI design and interface, any dataset or schema changes/extensions, etc. (20%)

3 Requirements Analysis

1. Store

The Retail Shop has multiple stores throughout different locations. Each store should contain the following information:

- Store ID (required, unique)
- Store Name (required)
- Location (latitude and longitude) (required)
- Date Established

2. Product

Each store offers various products to the customer. Given a store ID, the system should list all the products available in that store. Overall, each product should contain the following information:

- Product Name (required)
- Number of units available (required)
- Price per unit (required)
- Description
- Image url

Multiple stores may have products with same product name. For example, Store-1 and Store-2 may both contain products with product name "Soap". So product name is not the primary key of product entity set. (*Hint: Think of weak entity set*)

3. User

A User needs to register to be able to use our application. There will be two types of users in the system: Customer and Manager. Each user will have the following common attributes:

- User ID (required, unique)
- Password (required)

- Name (required)
- Email

In addition, Customers have some additional attributes: current location and credit score. Also Managers have some additional attributes: salary and educational qualification/degree. (*Hint: think about ISA hierarchy*)

Customers can view the products in each store, check the price and availability, place an order, view order history, and visit their profile. A customer can purchase any product from the store if a) the number units ordered is available in the store, and b) the store is within 20 miles radius of his/her current location. Customers can also view their order history, which should include the order date, product name, and quantity ordered. In the order history, orders are listed in chronological order according to their respective order dates. Customers can also update any fields in their profile menu except user ID, password, type and order history.

Each store is managed by a manager and a manager can manage multiple stores. Managers can also add and delete any product from the product table.

Managers may place a supply request of the products for the stores they are managing, and also specify the amount of units for the products needed. Each supply request for a product is assigned to a warehouse, who then delivers the requested products to the stores. (*Hint: think about aggregation to handle this scenario.*)

4. Warehouse

The retail shop has multiple warehouses to supply the products to the stores as per requests from the store managers. The warehouse has the following attributes:

- Warehouse ID (required, unique)
- Name (required)
- Location (latitude and longitude)
- Area in square feet

A manager will need to specify the store ID, the product name, and the number of units required to place a request to a warehouse. We are assuming that the warehouse will always have enough products in the inventory. When the supply from the warehouse arrives, the product's information (number of units) in the product table will be updated.