CNG352 Database Management System

TERM PROJECT

Lecturer: Assoc. Prof. Dr. Yeliz Yesilada
Date given: March 07, 2023

OVERVIEW

You are required to design a relational database system to suit the needs of a large company. This application must meet certain requirements of complexity, just to give you enough scope to try out everything you learn about databases. Below are some suggestions on applications that are suitable. If you have an application in mind, which is not on the list, you are welcome to do it after confirming it with your lecturer. Any application, which is too trivial, will be rejected.

SUBMISSION

The project consists of **5 steps (4 graded)**, submission details of each one is given separately.

GROUPS

Students cannot work individually on this project. They have to form a group to work on the project. They can form groups of at least **two** students. Later on you cannot also change your group members.

PROJECT TOPICS

This application must meet certain requirements of complexity, just to give you enough scope to try out everything you learn about databases. Below are some suggestions on applications that are suitable. If you have an application in mind that is not on the list, you are welcomed to do it after taking confirmation. Too trivial applications will be rejected.

The first thing you should do is to decide on the information needs of the application. To find out what the information needs of the application:

- Ask people who know about the application
- Go to the library and read about the application
- Find forms that describe the information that is normally recorded for that application.
- Check the features available of similar systems.
- Talk to people that you will be developing this application for.
- Example applications from the previous years:
  1. Retail management system.
  2. A social network system for healthcare.
  3. A tour operator system.
  4. A technology park application system.
  5. A personal travel blog.

6. A personal movie/songs management system.
7. An online game and its backend database.
8. Advertising agency management system.
9. An event/organisation/social activities organisation management system.
10. A second hand selling/buying web application for METU NCC.

PROJECT DESCRIPTION

There are four major tasks involved in this project:

- Do data requirement analysis of your application and design an entity-relationship diagram to represent your application;
- derive a relational schema from the entity-relationship;
- implement queries and updates on a standard database instance for this schema which could be mySQL, Oracle or PostgreSQL, etc.;
- implement an interface on the standard database (You can also use Java, C++, Python, PHP – please note that you will need to be able to explain your code).

Step 1: Proposal of the Application

DUE: March 16, 2023, Thursday @ 23:55

Submission: Report via ODTU Class

Write a short (approximately one paragraph) description of the database application you propose to work with throughout the course. Your description should be brief and relatively informal. If there are any unique or particularly difficult aspects of your proposed application, please point them out. Your description will be graded on suitability and conciseness. Try to pick an application with a schema that is relatively substantial, but not too enormous. For example, your E/R design should have in the at least 7 and maximum 15 or so entity sets, and a similar number of relationship sets. Also submit the names of the group-members.

In your report you need to include the following:

- Your report need to have a cover with the project title and group members with their names and IDs.
- You need to include one paragraph which is the clear description of your project description.

Step 2: Requirement Analysis and Design

DUE: March 30, 2023, Thursday @ 23:55

Submission: Report via ODTU Class

Write a short report that includes the following two parts:

Part1: This part of the report should include the requirement analysis of the database application you propose to work with throughout the course. This is the preliminary stage of the database design. Before you can effectively design a database, you must know and analyse the expectations of the users and the intended uses of the database in as much detail as possible. This process is called requirements collection and analysis. Depending on the project topic you have chosen, you can have different approach to do the requirement analysis. If you have real users, you can talk to your users or you can refer to literature. There are different techniques for requirements analysis, you need to choose the best one suitable for your application. If there are any unique or particularly difficult aspects of your proposed application, please point them out. Your requirement analysis will be graded on suitability and conciseness.

Part2: This part of the report should mainly include the entity-relationship design of your project. You should produce an entity-relationship design to represent your application. The design should be expressed in a consistent ER diagram notation and be submitted via ODTU Class including:

- The ER diagram: You should certainly include different kinds of relationships (e.g., many-one, many-many) and different kinds of data (strings, integers, etc.). Depending on the complexity of you application, you can also include advanced features such as weak entity sets, "is-a" relationships, or roles.

- Notes describing the design decisions that led to the final ER design; in particular, note any points where you considered alternative design choices

In your report you need to include the following:

- Your report need to have a cover with the project title and group members.
- You need to include your project description.
- You need to include requirements as two parts: data requirements and transactions requirements. Transaction requirements need to include insert/delete/update and data transactions.
- You need to include your (E)ER diagram. You need to clearly explain your assumptions, design choices and constraints in your design.

Step 3: Relational Schema Design and Normalisation

DUE: April 20, 2023, Thursday @ 23:55

Submission: Report via ODTU Class

Make a logical design of the database. Make sure that your relations are in at least 3NF. You should produce a relational database schema to represent the data of your application. Steps that you should follow are as follows:

(a) Using the method for translating an E/R diagram to relations, produce a set of relations for your database design. As usual, please be sure to underline key attributes in your relations.

(b) For each relation in your schema, specify a set of completely nontrivial functional dependencies for the relation. Any functional dependencies that actually hold in the real-world scenario that you're modelling should be specified, or should follow from the specified dependencies. Don't worry if you find that some of your relations have no nontrivial functional dependencies.

(c) Is each relation in your schema in Boyce-Codd Normal Form (BCNF) with respect to the functional dependencies you specified? If not, decompose the relation into smaller relations so that each relation is in BCNF. Be sure to underline key attributes in your new relations. Don't worry if you don't have any BCNF violations.

(d) Are there any nontrivial multivalued dependencies that hold on any of the relations in your schema? (You needn't consider MVD's that are also functional dependencies.) If so, specify the multivalued dependencies, then decompose the relations into smaller ones so that each one is in Third Normal Form (3NF). Be sure to underline key attributes in your new relations. Don't worry if you don't have any 3NF violations

(e) Now that you've decomposed your relations as far as possible, are there any relations that could be combined without introducing redundancy (i.e., without creating BCNF or 3NF violations)? If so, combine them.

(f) Is there anything you still don't like about the schema (e.g., attribute names, relation structure, etc.)? If so, modify the relational schema to something you prefer.

You should also submit notes that document the design decisions you made in developing the relational model and its schema; in particular, discuss how the relational schema handles the relationships and any inheritance and structured data from the ER design.

Note that both the ER and relational designs have to be submitted at the same time.

In your report, you need to include the following:

- Your report need to have a cover with the project title and group members.
- You need to include your project description.
- You need to include updated data requirements and transaction requirements.
- You need to include your (E)ER diagram. You need to clearly explain your assumptions, design choices and constraints in your design.
- You need to clearly show your mappings, assumptions and techniques used.
- You need to clearly show the functional dependences.
- You need to explain your normalisation process. You need to present them step by step until BCNF including (1NF-3NF).
- You need to clearly show your finalised normalised relations.
- If you also have to explain database tunning and optimization done, in particular denormalization decisions taken (see item (f) above).

Step 4: Relational Database Implementation

DUE: May 11 2023, Thursday @ 22:55

Submission: Submit your report and your scripts via ODTU Class.

Combine all of your scripts (create table, insert, queries) using the 'zip' command.

Make a physical design of the database by assuming a certain workload. The design should be expressed as a collection of SQL table declarations, including any primary key, foreign key and domain constraints. Decide the required queries for your application.

Write at least **five** data retrieval (select) commands that represent the transaction requirements given in step 2 and at least **three** data modification commands from all the following operations: insert, update and delete commands. Your queries should be complex enough (try to use aggregates and set operators). Insert meaningful data to your tables. Discuss the workload of your application, the frequency of the queries to be called and updates to be made, the approximate number of records in your tables. According to the workload of your application decide the indexes to be used. You can implement this part of your application using either ORACLE or MySQL or PostgreSQL, please clearly specify which one you use in your application.

In your report, you need to include the following:

- Your report need to have a cover with the project title and group members.
- You need to include your project description.
- You need to include updated data requirements and transaction requirements.

- You need to include your (E)ER diagram. You need to clearly explain your assumptions, design choices and constraints in your design.
- You need to clearly show your mappings, assumptions and techniques used.
- You need to clearly show the functional dependences. You need to explain your (de)normalisation process. You need to clearly show your normalised/final relations.
- You need to include scripts for your database definition and data manipulation queries. These have to be the actual and real queries that you have used to create your physical database on your machine. These have to be consistent with the earlier steps of your project, especially step2. You also need to include the data population and manipulation queries.
- Include a discussion on the questions asked above, including the workload, indexes, etc.
- Please note that for this step, you will also be asked to give a demo of your application.

Step 5: Interface

DUE: June 01, 2023, Thursday @ 22:55

Submission: Submit your final report and application via ODTU Class. Also each group will make a demo.

Using Java or C++ (you can also consider alternatives such as Phyton), write an interface which accesses to your Oracle or MYSQL or PostgreSQL database. This program should satisfy the following:

- A list of at least five alternative options is offered to the user. (An additional alternative should be quit.)
- The user selects an alternative.
- The system prompts the user for appropriate input values.
- The system accesses the database to perform the appropriate queries and/or modifications.
- Data or an appropriate acknowledgment is returned to the user.
- The system gives error messages if necessary.

You should include both queries and modifications among your options. As in your previous assignment, please include some "interesting" queries or modifications. The data you have inserted to the database should be appropriate enough to show the capabilities of your program.

In your report, you need to include the following:

- Your report need to have a cover with the project title and group members.
- You need to include your project description.
- You need to include updated data requirements and transaction requirements.
- You need to include your (E)ER diagram. You need to clearly explain your assumptions, design choices and constraints in your design.
- You need to clearly show your mappings, assumptions and techniques used.
- You need to clearly show the functional dependences. You need to explain your (de)normalisation process. You need to clearly show your normalised/final relations.
- You need to include scripts for your database definition and data manipulation queries. These have to be the actual and real queries that you have used to create your physical database on your machine. You also need to include the data population and manipulation queries.
- You need to include a new section/chapter on the modifications/changes you need to do on your database created in the previous step.
- You need to include full implementation details of your application. Which languages you used? Which APIs you used, etc?

- You need to include screenshots of your application. For each type of transaction (insert/delete/update and data transactions), you need to have relevant UI implementations, explain your code along with queries and show screenshots of your implementation.