School of Computer Science, McGill University

# COMP-421 Database Systems, Winter 2024

Project 1: Database Design and Data Modelling

Due Date January 31, 12:00pm (*NOON*)

In the programming project of this course, you will develop and build a database application for a **real-world** domain. Step by step, you will design a schema, create a database using DB2, populate your database with data, maintain, query and update your data, develop application programs, and implement a user-friendly interface. The interface can be very simple so no requirement for web-programming, etc. You will only use a standard programming language in the last project deliverable. The course itself will cover Java but you can use another programming language (with approval from the instructor and the TAs). Each team turns in one solution for each project deliverable.

**Submission Format:** Your submission will contain **two** separate documents:

1. The ER diagram of your project. This can be a PDF file or PNG. Make sure that the ER diagram you submit is readable (such as font size, etc). You may use a page size larger than letter for your ER diagram page if that ensures it is readable. **Your group number should also be included in the ER diagram** (top right side).
2. Rest of your project document (requirements, descriptions, relational schema, etc.). This should have at least 3 pages and should not exceed 6 pages and can be submitted as a PDF or DOC format. The first page should include the project title, the group number, the names of the group members. That first page can then continue with the requirement analysis, etc. You do NOT need an extra title page.

## 1 Assignment

In this first assignment you have to choose an application domain and design your database. Below are several possibilities. But you can choose any other application that can be typically found on the Internet. If you choose an application not listed below consult with the instructor that is responsible for the project, whether it is ok. You have to perform the requirement analysis for your application, design the entity-relationship schema (ER) for the data described in the data analysis, and translate it into a relational schema. Choose an application you are interested in; then you will have more motivation doing this project!

The application should be substantial but not too big. Consider a range of 8 to 12 entity sets, and a similar number of relationship sets. The schema should include different kinds of relationships and different data types. Your schema should have at least one "special" construct such as a weak entity set or an is-a-hierarchy. But make sure that such modelling makes sense in your application context. Do not force this if it is not appropriate. The total number of entity sets + relationship sets should be at least 16 and NOT to exceed 25. Expand/Reduce your requirements so that you can adjust your ER schema. DO NOT skip adding entity sets and relationships in the ER schema to keep the total number below 25. **You will loose points if requirements are not mapped into the ER schema. Instead, adjust your requirements so that you can have a smaller schema.** Remove the less essential requirements if otherwise the schema would be too large. For example if your application is a store selling something, it is important to keep track of the product inventory, sales info (who bought what for how much, etc.), but you might not need to worry about supporting product returns or changing prices of products with time etc.

DO NOT design a "star schema", i.e., where all the relationship sets in the schema are from one particular entity set. As a rule of thumb, if you have $n$ relationship sets in your schema, see to it that no single entity set is participating in more than $n-3$ relationship sets. This is to ensure that your ER schema is reasonably sophisticated so that you can build interesting features in the succeeding project deliverables.

You have to turn in the following.

1. (35 Points) A requirement analysis of the application. This is a half-formal specification. It should be structured as follows: short introduction/overview of the database application, description of the data requirements (which real-world entities are captured, such as clients, purchases, etc.), and the functional requirements, i.e., a description of the tasks/operations that will be performed on the data (performing a purchase, booking a room, registering a new client, etc). In our evaluation, more value will be put on the data requirements but the overall description should be complete. The ER schema developed in the next step should not contain data that is not described here in the specification.

   If there are any unique or difficult aspects, point them out. Be precise about the real-life concepts that you want to model, their relationships etc. Also consider constraints, restrictions or special requirements that your application might have. We also discussed examples in class when we looked at Minerva and other examples. Your description is expected to be very detailed.

2. (40 Points) An ER schema/diagram including your data requirements. Be careful not to forget to underline key attributes, indicate the types of relationship sets etc. If there are any constraints within the application that you cannot depict in the ER diagram, point them out.

3. (15 Points) Use the method for translating an ER diagram to relations described in class and depict each resulting relation in the form `Relationname(attr1, attr2, attr3,...)` underlining the key attributes (e.g. `Students(sid, name, age, gpa)`). Indicate when attributes are foreign keys to other relations by writing something like "attr3 foreign key referencing relation X" beside the relation. Are there opportunities to combine relations without introducing redundancy? If so, indicate which, and if not, tell us there are none.

   Note: You do not yet need to give the SQL create statements or decide on the data types.

4. (10 Points) For creativity and complexity of your application/design. This is to ensure that you do not turn in a bland, simple, "flat" application that does not explore some of the more sophisticated areas of ER modeling. Here are examples of things we are looking for.

   - How your application domain stands out among the other project applications.
   - Appropriate usage of special concepts such as is-a hierarchy, weak entity sets, ternary relationship sets (more than one).
   - Thoroughly reasoned usage of key constraints.
   - Particularly well described requirement analysis.

   **Please do not "force" anything into your ER schema** if it does not fit your requirements. If doing so, you introduce errors into the schema, you will not get the creativity points and instead may loose points for the errors. In the past, many groups have used the is-a-hierarchy where it was not meant to be. Even if you are a little short in any of the above features but end up with a very good schema and relational translation, we may decide to give you points for the good work you put in.

   Write a very short description of what features in your project stand out for the above creativity part.

5. (0) Points. Indicate one or two web-sites that inspired your design.

6. (0) Points. Indicate in one paragraph how you worked together (how many meetings) and how each of the team members contributed to the project deliverable.

# 2 Project Topics

Below are several, pretty widely defined topics that your application could be chosen from. Of course, you have to decide on a more specific domain/area/application/enterprise within the topic and do research on what are the specific characteristics of the application.

The data you want to store should be realistic in the sense that for the chosen application domain, this is really the information that is relevant and should be maintained.

- A store/company/enterprise/organization that *sells* something to customers. Choose a specific enterprise of your choice, e.g., a bookstore selling books to clients, maintaining its stock, etc. A music center selling concert tickets, an online music store, etc. There are two minimum conditions: (i) The process of a customer buying something (i.e., a purchase) must be reflected in the schema. (ii) A purchase should allow the inclusion of more than one product item (you can buy more than one book in one purchase). You may want to go through one of the online stores and see what information is all needed to perform such a purchase. Look also what is the other functionality provided by such an online store.
- A car rental company, a hotel, a spa, a travel agency or any other type of enterprise that includes a *reservation system*. The process of making a reservation by a customer must be reflected in the schema. Try do be realistic. You may want to go through one of the systems online and see what information is all needed to perform such a reservation (but do not finally submit your reservation !).
- A social networking site.

If the application you would like to develop does not fit in the three topics listed above, please talk to the instructor responsible for the project to check whether it is ok.

WE DO NOT ACCEPT

- A university database
- An airline company
- A general purpose enterprise consisting of employees, projects, products etc. Although your application item might include any of these entity sets, you should choose a more specific enterprise (what kind of item products, projects etc.)
- A database that resembles the example database project that is provided from a previous year.
- The running text book example and the database description in Assignment 1.

# 3 ChatGPT

If you can't resist, you can try to use ChatGPT to help you with this task. We tried and the results were not great (of course, this might depend on the questions you post). **If you use ChatGPT you must add a third document to your submission**. This document must contain

- the questions you asked ChatGPT and the answers you got (reasonably formatted);
- a short description of how you used/transformed this information to create the solution you have submitted.