ECE 656 Project

Fall 2021

The course project is a database-design and implementation exercise, together with a data-mining exercise. Broadly speaking, you are required to

1. select a non-trivial dataset, thinking not just about the dataset size and complexity, but also the purpose for which you are selecting it;
2. create an appropriate database design for that dataset;
3. implement a prototype of the design, both server-side and client-side;
4. data-mine some aspect of the dataset.

1 Overall Description

The course project is a database-design and implementation exercise, together with a data-mining exercise. The starting point for this exercise will be a sizable dataset from a particular domain. When you select your dataset, you need to think about what kind of client application you have in mind for it. While some example projects were suggested in the syllabus, no mention was made of the dataset you would need to think about in the context of suc projects. For a used-car sales application, used car sales data would be appropriate and is available on Kaggle (see below).

There are several possible sources for datasets, including, but not limited to:

1. `https://www.kaggle.com/datasets/`
2. `https://duckduckgo.com/?t=ffab&q=large+dataset+sources+free`
3. `https://www.forbes.com/sites/bernardmarr/2018/02/26/big-data-and-ai-30-amazing-and-free-public-data-sources-for-2018`
4. `https://www150.statcan.gc.ca/n1/en/type/data`

By September 22nd you should submit an initial proposal to the relevant DropBox on Learn describing the following:

1. The dataset selected, including the source
2. The purpose to which you intend to put it
3. A description of the ideal functionality of the client
4. A description of the more limited functionality that you expect to prototype
5. A proposed data-mining exercise that you will do on the dataset selected. This should primarily be in the form of some question that you wish to investigate and for which it is plausible that the dataset around which you are forming your project is likely to have data relevant to the question

A couple of sample proposals based on a un*x filesystem replacement and a social network will be made available for you to consider.

No two groups may use the same dataset as another group, nor is NHL hockey data to be used since we will be using that for examples and assignments in the course.

The first sample proposal should have sufficient data as a typical filesystem contains several gigabytes of data in utilities alone; it may or may not be reasonable as a project based on database complexity, though, since it is possible to do that project using few relations/attributes. The second sample proposal plans to scrape data from the web; this is possible in principle, but difficult in practice, and likely to result in too small a dataset and/or too few attributes. Kaggle has some social media datasets that would be a better choice.

Feedback for this initial proposal will be within a week and you will then have a further week in which to fix any issues identified as requiring remedy and, if necessary, submit a final proposal.

Given that the requirements for project are as follows:

1. A command-line client application appropriate to the domain
2. An entity-relationship design to model the data
3. A relational schema based on the ER design
4. A data-mining investigation of the dataset

we will now briefly elaborate on each aspect.

2 *Client Application*

The client application is required to be one that is appropriate to the dataset domain. It must allow for two key requirements:

1. Querying the data in a way that a customer in the domain would do
2. Modifying the data in a way that a customer in the domain would do

For example, a used-car sales dataset would need a client that allows a customer to search for used cars on some reasonable basis that a person looking for a used car would want: by year, by make and/or model, by price, *etc.* Likewise, a person should be able to list a car for sale, modify the listing to change the price and/or add additional information, and remove the listing once the car is sold.

The user-interface need only be a simple command-line interface, even with single-letter commands, as this aspect will not form any part of the grading scheme.

If you wish to do a more sophisticated user interface you are welcome to do so, but you should be cautioned that (a) it will not affect your grade in the project; and (b) it will make creating testcases to demonstrate the quality of your finished product substantially more difficult.

It is expected of your project team to work out an appropriate set of things that a user would wish to do, though allowing for the fact that this is a course project and therefore you should scope your project accordingly. In particular, you should decide as a team

1. What you think an ideal client *should* be able to do
2. What you plan to actually implement for your client given the time constraints
3. (At the end of the project, when you write your report) What you actually implemented from your plan, and what you left
4. An explanation justifying each of the above choices

3 *Entity-Relationship Design*

You are required to create an entity-relationship design appropriate to the dataset domain. Your design is required to clearly identify:

1. All entity sets, specifying the entity set name and attributes, showing any compound attributes, multivalued attributes, and optional attributes per the methods described in the course
2. All relationship sets, specifying the relationship set name and any attributes it might have
3. All primary keys, cardinality constraints, and attribute domains
4. Any weak, specialized, or aggregations
5. Any other aspects relevant to an ER design

You are required to create an ER diagram for your design, and explain why you choice the entity sets, relationship sets, *etc.* that you chose. Where appropriate, you shoud specify what alternatives you considered and explain why you chose the design that you did rather than the alternative.

We will be covering an approrpiate ER design for NHL hockey data in the coming weeks, so that you will understand what kind of design choices are available to you.

4 *Relational Schema*

You are required to translate your ER design into a relational schema. There are a number of places where there will be choices for you to make in this regard, and in those places you should explain why you made the choices that you made.

In converting your ER design into a relational schema you are expected to write the necessary SQL code to:

1. create the required tables, views, *etc.* for the relational schema
2. create the required primary keys, foreign keys, and integrity constraints
3. create indexes as necessary for the query operations you will do both in the client and in the data-mining exercise
4. load the data from your dataset CSVs into the tables

It is quite likely, as you have already seen with the NHL game data, that the data will contain errors and inconsistencies relative to your design. You are required to handle those issues in appropriate ways, including:

1. fixing obvious data errors
2. removing any duplicate data
3. modifying your design in certain cases

In any situation where you must handle such issues, you should document what you did to handle the issues.

5 *Data-Mining Investigation*

Given a large set of data, we can determine information from that data. Indeed, this is how all of science proceeds. Data represents facts. We wish to see if those facts allow us to formulate a theory about something, and to validate that theory if possible. In this course we will teach you three specific techniques: classification, association discovery, and clustering. You will be required to implement one of those techniques and apply it to answering a question appropriate to the domain.

Specifically, you are required to

1. Select a domain-appropriate question that you want data mining to answer
2. Select a technique or techniques that will be appropriate to the question you are investigating
3. Implement said technique *efficiently* to build a data model
4. Determine the validity of your model
5. Report the results of your investigation

6 *Deliverables*

There are no formal intermediate deliverables for this project. However, it is *strongly* recommended that as you go through the term you report progress and solicit feedback from the course instruction team. A suggested project timeline will be made available to help you in this regard.

The final project deliverables are as follows:

1. Final Written Report: this should describe the client application, the ER design, the relational schema, and your data-mining investigation, detailing the specific issues required above. In addition, you should include a testcase plan that describes how you test the various code aspects of your project.
2. Code: you are not expected to submit your code but rather store it in the university github repository `https://git.uwaterloo.ca/`. The code in the repository should include the following:
   (a) All client code
   (b) The SQL code necessary to implement the relational schema and load the data from the CSV files
   (c) The code, SQL and otherwise, needed to implement your data-mining investigation
   (d) Test cases for the above
3. Video Demo: a 20-minute walk-through/presentation of your project. It should describe all of the aspects of your design, implementation, and results.