

Capstone Project – Lecture Notes for CSCI 3410 Database Systems

1 Overview

Test, debug and improve a code, exercise and notes bank for a class taught at Augusta University. The bank makes use of a large variety of softwares to version the source code, draw the diagrams, run and test the database code, and produce a document in various formats.

2 Rationale

The lack of good, open-source, free and accessible lecture notes for an undergraduate database class led Dr. Aubert to write his own, currently hosted at <http://spots.augusta.edu/caubert/db/ln/>. The notes quickly grew to reach ~200 pages, ~100 figures, ~500 lines of `SQL` and ~500 lines of `java` code. This growth was not carefully planned, and resulted in incoherences, poor naming choices, and imperfect file organization.

The team accepting this project would get an excellent overview of many of the subdisciplines of software engineering (testing, design, construction, maintenance, development process, etc.) while working on a project using multiple languages and tools that will be useful in their career (virtualization, versionning, scripting, Unix systems, etc.).

3 Technical Requirements

If you took CSCI 3410 Database Systems with me, then you already know some of the tools taught and used. There are no “requirement”, but being willing to learn how to use the following tools will be a big help:

- a Linux distribution (that could be virtualized on a virtual machine),
- shell, scripting and Makefile (cf. <https://www.gnu.org/software/make/manual/make.html>),
- git (cf. <https://git-scm.com/>),
- LaTeX (cf. <https://www.latex-project.org/>) and some of its tools, like latexmk (cf. [https://ctan.org/pkg/latexmk/](https://ctan.org/pkg/latexmk)),
- pandoc (cf. <http://pandoc.org/>) and some of its filter: pandoc-numbering (cf. <https://github.com/c-hdemko/pandoc-numbering#installation>) and pandoc-include-code (cf. <https://github.com/owickstrom/pandoc-include-code>),
- pdf2svg (cf. <https://github.com/dawbarton/pdf2svg>),
- MySQL (cf. <https://dev.mysql.com/>) or MariaDB (cf. <https://mariadb.com/>),
- MySQL Workbench (cf. <https://www.mysql.com/fr/products/workbench/>),
- MongoDB (cf. <https://www.mongodb.com/>).

4 Tasks

4.1 Setting Up

The first task will be to set up a development environment to be able to compile the lecture notes. As the current guide on how to compile those notes on your own (at <http://spots.augusta.edu/caubert/db/ln/README.html>) is probably incomplete, you’ll have to work out some of the details on your own.

The **deliverables** for this task could be either

1. more detailed instructions on how to get a running environment,
2. a minimal virtual machine containing all the requirements,
3. a docker (<https://www.docker.com/>) containing all the requirements,
4. improving the compilation scheme for the notes.

4.2 Organize

In this task, you'll be asked to develop naming conventions for

- the file names,
- the figures,
- the folders,
- the tags in the document,
- the links in the document,
- the classes and objects in the object-oriented programs.

The **deliverables** for this task could be

- a clear and documented naming convention for one of all of the aforementioned objects,
- the application of the convention you'll have developed, that need to be mindful of git's requirements.

4.3 Testing the Code

The code for this lecture is spread out, and some times have dependencies (like “*you cannot complete exercise XX without running the code from lecture YY*”), but sometimes can be run on its own. It would be profitable to clearly label which parts of the code can be run independently, and which parts has pre-requisite(s). Also, the code was written over a long period of time, and probably suffer from incoherences in the choice of the methods, datatype, and comment style.

The **deliverables** for this task could be

- a reorganization of the code in fewer, better organized pieces,
- a macro that guarantees that the entirety of the code can be compiled in one go,
- a reorganization of the comments.

4.4 Pushing Further...

Some other directions could be explored by the team:

- As of now, the lecture notes can be exported in `pdf`, `html` and `odt`. But pandoc can export the document in other formats, and it could be of interest to explore some of them (`docx`, `epub`, `Jupyter notebook`, etc.).
- The `html` version of the notes (cf. http://spots.augusta.edu/caubert/db/ln/CSCI_3410_lecture_notes.html) could use some improvements in the style, presentation of the table of contents, etc.
- Generally speaking, there is a list of known bugs at https://rocketgit.com/user/caubert/CSCI_3410/source/tree/branch/master/blob/KNOWN_BUGS.md that need to be addressed, and can also be expanded!