Group Project Guide

COMPSCI 310: Introduction to Databases

by Mustafa Misir, mustafa.misir@dukekunshan.edu.cn, WDR 2106

Project topics are open to any domain, with the only requirement being a significant reliance on a relational database following a project group of size ~3 is built. You are free to utilize your preferred relational database management system (RDBMS), programming language, and frameworks. However, it is essential that all the core tasks and major components of the project are developed from scratch by the team members, rather than relying on pre-built tools or libraries. If you have any doubts or concerns, please consult with the course instructor for clarification.

To ensure your project's success, regular consultations with the course instructor are mandatory as *weekly meetings*. The objective is to receive constructive feedback in a timely manner. During these sessions, group members should provide updates on their current status, outline their upcoming tasks, and engage in discussions to clarify any doubts or concerns.

Here is the overall project timeline:

- **26/10**: project topic approval deadline
- **02/11**: software requirements specification document submission
- **16/11**: database design document submission
- **04/12** : project presentation
- **07/12**: report submission and peer group assessment submission

Proposal

The project proposal is expected to be concise, spanning 1-2 pages in length. There is no specific format requirement.

The project topic is open, with the stipulation that it must necessitate the use of a database as a fundamental component. The chosen topic should be clearly defined to demonstrate the requirement for a database.

Proposals should include:

- ***Motivation***: A clear and concise explanation of the reasons behind the proposed project.
- ***Objectives***: A well-defined statement outlining the project's goals and expected outcomes.
- ***Database Statement***: A justification for the need of a database in the proposed project, highlighting its essential role in achieving the project's objectives.

Prior to submitting your proposal, it is mandatory to obtain approval from the course instructor to ensure that the proposed project meets the requirements.

Note: Only one submission (only by one group member) is needed per group.

Report

Your **project report** should be prepared in LaTeX form using the Springer LNCS template. Overleaf can be used as a preferred LaTex platform which is suitable for group-based writing tasks. The Overleaf free plan allows 2 people to work together at the same time. If your group has more than 2 students, you can go with the *Turn on link sharing* option under *Share*. It will allow editing of the document by anyone who has the document sharing link.

Your project report should have these sections: *Abstract*, *Introduction*, *Problem*, *Method / System*, *Computational Results / Evaluation* and *Conclusion*, each explained below. You can see similar arrangements in the scientific articles.

a. *Abstract* essentially provides a concise summary of your work.
b. *Introduction* discusses the motivation and objectives of your work referring to the relevant, up-to-date / most recent / state-of-the-art academic literature and/or products / systems. Concerning the academic resources, you can reach those related articles through Google Scholar / Baidu Scholar.
c. *Problem* explains the domain you focus on. This part should provide a formal definition, likely using some mathematical terms when possible, besides describing the problem in words.
d. *Method / System* will explain the ideas to be applied. Similarly to the *Problem* section, a formal definition is essential especially if you are utilizing some algorithmic approaches. Furthermore, the complete system design with the reasoning behind your choices, the developed sub-components and how they interact with each other, should be reported. This part should be specifically supported by flow charts showing how the system works and screenshots illustrating some representative use-cases. More importantly, class diagrams (if applicable) and explicit database design should be provided.
e. With *Computational Results / Evaluation*, if your system has empirical aspects, it needs to be tested and the results should be reported while discussing its advantages and disadvantages. When possible, it should be compared against the existing state-of-the-art / best approaches from the literature. Otherwise, the system should be tested with different use-cases to show its performance and capabilities, possibly including screenshots. All should be reported with an in-depth discussion.

f. The report should be finalized with the *Conclusion* section, briefly summarizing the developed system alongside with the results / evaluation, and take-home messages. It should additionally offer follow-up ideas on what to do next.
g. In addition to the main report, an *Appendix* section is required, outlining the specific contributions made by each team member. A weekly project timeline should also be included, highlighting key accomplishments, obstacles encountered, and the strategies employed to overcome them.

The complete code / queries do not need to be provided directly in the report. Yet, the major implementation elements, such as critical SQL queries can be included.

The **final report** should be prepared in the form of an academic article. Thus, it should be carefully handled, making sure all the content is accurate and verifiable besides referencing all the papers benefited from, either as literature entries or certain code / data level resources or algorithmic methods.

The report's length is not strictly defined, as it will depend on the project's specific needs. However, a general guideline is to aim for a report that is substantial enough to cover the key aspects of the project, typically in the range of 15-20 pages.

Presentation

It should be prepared in the form of a **report** (explained above) summary, respecting the given presentation time. It is suggested to have appendix slides, to answer the potential questions. Additionally, live project demos can also be done within a presentation. *A recording of a demo can also be submitted as a supporting material.*

Peer Assessment

This is an evaluation report on the project assigned for assessment as each project group needs to assess the project of another group, to be released after the group formations are finalized. It should essentially provide the strengths and weaknesses of the assigned project. Like the project report, this document should also be prepared in LaTeX. It is expected to be 1-2 pages long.

Submission

All the project materials, including the peer assessment report, should be submitted in a .zip file named as:

COMPSCI310_F24-25-S2-Group_#No_Name1Surname1_..._Name*n*Surname*n*.zip

where the names belong to the group members.

Only one submission per group is required. Thus, there is no need for each group member to re-submit.

It should be noted that the project should be carried out through GitHub for effective team work. The corresponding project link should be stated in the report as a footnote of the report's / project's title. The complete implementation including the data concerning the underlying database should still be in the submitted .zip file.

A README file should be provided so that the delivered project can be run smoothly for testing by the course instructor.

RUBRIC (20%)

- **Presentation** (5%): Planning (3%), Content (2%)
- **Code & Database** (4%): Quality / Value (3.5%), Comments / Explanations / Sources (0.5%)
- **Report** (10%): Motivation + Objective + Background (3%), Individual Contributions + Timeline / Progress Notes (2%), Analysis / Evaluation / Demo (3%), Writing Quality / Format (2%)
- **Peer Assessment** (1%): Questions / Comments (0.8%), Writing Quality (0.2%)

Grades will be assigned on an individual basis, taking into account each team member's unique contributions. This means that team members may receive different scores. Therefore, it is essential to clearly document each individual's role and responsibilities in the project.