

# CS353. DATABASE SYSTEMS

## Spring 2024

### TERM PROJECT

#### GENERAL ISSUES

Throughout the semester, you are required to develop a database application as a term project. The aim of the project is to practice actual database design, implementation, and query formulation.

This project specification document will serve as the most essential guideline while you are developing your term project. So, please read it very carefully and keep it through the semester. You can also use this document as a *checklist* before submitting a report, i.e., at every major stage of the project, to see if you have satisfied the basic requirements of that particular stage. You should also check course announcements Web page regularly to be informed on the latest announcements about the projects.

The general issues about this term project are listed below:

- First, you must form project groups of **5 students**. Groups with less number of students will be allowed only in exceptional cases (and the expected workload for the project will still be no less than a typical 5-student group project). You must determine your groups and e-mail to the TA until **February 9, 2024**.
- Then, **we** will randomly assign each group a project topic. Topics assigned to each group will be announced at the course Moodle page on **February 15, 2024**. You are **not** allowed to choose a topic other than the one assigned to you.
- The project has 3 major stages (proposal, design, implementation) and a demo. For each stage, you will submit a report, and each of them will have some weight on your final project grade, as well as the demo. Tentatively, **proposal and design reports** will add up to **40%**, **final (implementation) report** will be **20%** and **demo** will be **40%** (percentages are subject to change).
- All projects will be web-based applications including the end-user and administrative interfaces. Desktop applications are not allowed!
- At proposal and design stages of the project, your reports will be graded and returned to you with various comments. Please check the spelling of your reports before submission.

#### Project Proposal

The project proposal that you'll prepare should clearly and completely describe the application system that your team is proposing, and discuss why/how a database is going to be used as a part of the system. After this problem statement, you should define the requirements and limitations of your application system.

You should next provide a **conceptual design** of the database using the E/R model. A typical E/R design may include strong and weak entities, binary/ternary relationships, aggregations, generalizations/specialization, cardinality constraints, keys and descriptive attributes, etc. As long as your application permits, we favor a **large** number and variety of entities, relationships, etc. Of course, we **don't want** artificial entities etc. that are irrelevant to your project, but we believe, for most of the projects, there is enough room to add various things by using your

creativity & imagination. So, try to have a reasonably rich database from the very beginning. Of course, you will have the chance of making corrections, extensions (and even deletions, sometimes) based on the TA feedbacks to your proposal and design reports, but it is important to think/brainstorm about the requirements of upcoming stages and start with a reasonable initial design.

## Design

In the second phase of your project work, you should provide the design of the *database* and *user interface*.

The *design of the database* should include the following:

- Revise the E/R diagram based on the comments you received.
- The E/R diagram should then be converted into a set of **relations**.
- For each relation identify attribute **domains**, **candidate keys (and the selected primary key)**, and **foreign keys** due to referential integrity.

The *user interface design* and corresponding *SQL statements*:

- For all major tasks for achieving the functionality requirements of your project, provide sketches of GUI and corresponding SQL queries that would be executed to perform that task. In particular, the GUI sketches would illustrate how the user interacts with the system in the typical scenarios and what underlying queries would be called to achieve the tasks in these scenarios. Notice that, SQL queries should include data retrieval queries and data modification (add/delete/update) queries on one or more tables or views.
- Advanced database components, such as views, triggers, and constraints, should also be considered to be included in the user interface design.

You should prepare a design document that clearly presents all of the items discussed above (the design of the database and user interface) and the implementation plan for your project. The implementation plan should describe what hardware/software system will be used to implement the database, tasks, and user interface.

- Possible DBM systems for the project include the latest releases of Oracle, Interbase, MSSQL Server, MySQL and PostgreSQL (MS Access is not allowed for this project). Note that, the DBMS you will use **must** support all modern features like views, triggers, constraints, etc.
- Possible development technologies include back-end technologies PHP, JSP, Spring, Django, and front-end technologies Angular, Vue, React frameworks. Note that while interacting with the database you must write raw SQL queries. Some backend technologies allow you to create and manipulate the database without writing a single line of SQL code. This is strictly forbidden. Such technologies can only be used if they support raw queries as well. Remember that you may be asked to show your SQL queries during the demo.

Please consult to your TA if you plan to use platforms other than the ones listed above.

## Implementation

In the last phase of your project work, you should implement the database, user interface, and SQL queries of your application system. You need to make sure that you can manipulate your database through your user interface. Therefore, your front-end and back-end must communicate perfectly. In order to make sure that your system works correctly, you are required to perform a detailed test on the system.

## Final Report

The final report should include the following:

- A brief description of your application system.
- Contribution of each group member to the term project in detail.
- Final E/R, if you have made any modifications after your design report.
- Final list of tables, if you have made any modifications after your design report. (List the tables, primary and foreign keys, no need for SQL create table statements).
- Implementation details: The implementation details should include how you have implemented some important basic concepts in your programming environment, and any other details you want to mention. For instance, if you use JSP and MySQL, you can tell about how you connect/access DB (how can you create SQL statements and send to DB), how you prepare your GUI, how you enforce various constraints, what the problems/limitations you have encountered are, etc.
- Advanced database components, including views, triggers, and constraints generated by the system in performing various user operations.
- A *User's Manual* that contains all the information necessary to use and maintain your system. More specifically, it should be a very detailed description of your system for all possible user groups, with GUIs, etc.

## Schedule (subject to change, please check the announcements)

- Project Group Proposal: **February 9**, 2024 (e-mail group members to the TA).
- Project Proposal: **February 26**, 2024.
- Design Document: **March 25**, 2024.
- Final Report: **May 17**, 2024.
- Demonstration: **May 20**, 2024.