Final Project

Note: The report at each stage should be submitted in **PDF** format. Other file formats will NOT be accepted.

Team Work

Students in teams of 3 are required to select an application domain that requires a back-end database and build the application from start to finish.

Goals

You will implement a database system for an Uber Eats-like food ordering and delivery platform. You will then design the underlying database and define the application functionalities you will provide with the database. Lastly you will implement this application, build and populate the underlying database systems, and write the code needed to embed the database in the application. You will demo your application and possibly showcase it to the whole class.

Schedule

Stage 0: Group formation

Due: Mar 19, 2024

You must form a group of 3 at this link.

Stage 1: Functional description, ER design and development plan (10 points)

Due: Apr 16, 2024

1. **Your application title**
2. **Description:** State as clearly as possible what you want to do.
3. **Usefulness**: State as clearly as possible why your chosen application is useful. Make sure to answer the following questions:
    1. Are there any similar or equivalent websites/applications out there?
    2. If so, what are they, and how is yours different?

4. **Realness**: Describe what your data is and where you will get it. Do you get it from the Web, some other application, or do you make it up?
5. **Description of the functionality that you plan to offer**. This is where you talk about how to meet the functionality requirements.
    1. Basic functions: see Stage 3 for what kind of basic functions you need to offer.
    2. Advanced functions: Each team must implement **at least two** creative things (i.e., advanced functions). The advanced functions are something that doesn't exist in equivalent websites/applications and should go beyond the basic functions listed above. In addition, an advanced function should also be something that is technically challenging, meaning you would need to spend some significant amount of time (at least a few days of work) to implement it. Of course, such functions should be relevant and therefore useful for your application. Remember that you have to define a challenging problem at the time of your final demo and show how you solved it. To have an idea about our expectations, a team of 4 with average programming skill will spend 3-4 days programming each of the advanced functions. It has to have programming involved. This is 3-4 days of useful work so it doesn't include the time spent on learning the material or debugging the code. Make clear what the features are, and explain why you think it is cool to have them.
6. **EER design (Please do not submit hand-drawn figures. You can draw EERD using draw.io)**
    1. **At least 6 entities**.
    2. Your design must include **at least one union or specialization/generalization**.
7. **The relational schema of your database (Please do not submit hand-drawn figures)**. Remember to include all keys.
8. **Development plan**
    1. Describe the labor division among group members if you work on a team.
    2. A project timeline with milestones.

Stage 2: Initial check (15 points)

Due: May 7, 2025

Initial check: **Make an appointment with the TA for a demo**. The demo must demonstrate the following using web pages or mobile app interfaces:

1. (3 pts) At least one example showing how to insert a record into the database
2. (3 pts) At least one example showing how to update an existing record
3. (3 pts) At least one example showing how to delete an existing record from the database
4. (3 pts) At least one example showing how to search the database and list the results

5. (3 pts) User interface (UI) design

   Note: This should not be your page login information (insert username/delete etc). We want to see some actual records from your project updates, else you will get no credits.

Stage 3: Presentation and demo (55 points)

Due: May 28, 2025

Project presentation: We'll randomly pick up some teams for presentation. Each team is allocated 10 mins for presentation. Please make presentation slides.

Project demo video: Each team must create a **5-min long video** for the project demo. Each team needs to demonstrate all of the following using web pages or mobile app interfaces connected with SQL queries. We don't accept the queries directly written in SQL editor at the demo time or any existing tools, such as the DBMS's command-line client, phpMyAdmin, or any admin tools that come with the framework with which you build your app.

Make an appointment with the TA for a demo. Each team needs to demonstrate all of the following.

1. Basic functions: (40 points)
   - (5 points) There is some real data in the database: either crawled from real websites or inserted by your friends - 10 developed accounts (not fake randomly generated values). If you crawl data, the DB should have at least 100 records and if you are going to have user generated data then you must have at least 25 records in your database before your final demo deadline.
   - (15 points) Show how to insert/update/delete records to the database, **at least 2 each**. **The insert/update/delete should be different from the ones in the initial check.**
   - (20 points) Show how to search the database and list or print returned results. You need to show at least 5 different interesting queries over your database. Three of the queries must involve joining multiple (at least 2) tables. **Note that each search (query) should involve at least one table different from other searches.**

     Note: This should not be your page login information (insert username/delete etc). We want to see some actual records from your project updates, else you will get no credits.
2. Demo advanced functions: (15 points). More points will be given if you do more.

    - Easy: 0- 5 points
    - Medium: 6- 10 points
    - Difficult: 11- 15 points
3. UI: (0 ~ 3 bonus points). Bonus points are awarded to projects showcasing good user interface design.

Stage 4: Final report (25 points)

Due: May 28, 2025 (submit the following on the CU)

A. **Project demo video**

B. **Project presentation slides**

C. **Project source code**

D. **Final report** must include the following:

1. Briefly describe what what your project is about and what the project accomplished
2. Discuss the usefulness of your project, i.e., what real problem you solved
3. Include your ER diagram and schema
4. Discuss the data in your database. Briefly discuss from where you collected data and show you did it (if crawling is automated, explain how and what tools are used)
5. Clearly list the functionality of your application (feature specs)
6. Explain on basic functions
7. Show the actual SQL code snippet
8. List and briefly explain the dataflow, i.e., the steps that occur between a user entering the data on the screen and the output that occurs (you can insert a set of screenshots)
9. **Explain your advanced function and why it is considered as advanced.** Being able to do it is very important both in the report and final presentation
10. **Describe one technical challenge that you encountered.** This should be sufficiently detailed such that another future team could use this as helpful advice if they were to start a similar project or were to maintain your project. Say you created a very robust crawler - share your knowledge. You learnt how to visualize a graph and make an interactive interface for it - teach all! Know how to minimize time building a mobile app - describe!
11. State if everything went according to the initial development plan and proposed specifications, if not - why?!
12. Provide a link to your video.

Grading:

You have to submit your source code, presentation slides and report to the CU. More will be expected of larger groups.

The projects are scored by the team, however, different individual contributions can lead to different grades given the same team score. Individual contributions will be conducted in a peer assessment manner.

- i = individual score
- t = team score
- c = your contribution (X %, e.g., 25%)
- Sum of c for all members in the team should equal 100.
- d = contribution divisor (Y %, e.g., 25% for a team of 4, 33.333% for a team of 3, etc.)
- **Formula: i = min (sqrt(c/d) * t, 100)**
- Examples:
    1. Team got a 90 and your contribution was 33% for a 3-person team.
        - i = min (sqrt(33.333/33.333) * 90, 100) = min(1*90, 100) = 90.
    2. Team got an 85 and your contribution was 40% for a 3-person team.
        - i = min (sqrt(40/33.333) * 85, 100) = min(93.11, 100) = 93.11.
    3. Team got a 95 and your contribution was 20% for a 2-person team.
        - i = min (sqrt(20/50) * 95, 100) = min(60.08, 100) = 60.08