# CS 665 Project 1: A CRUD Application (Individual Project)

## General Description

In this project, you will develop a desktop/web/mobile application that uses **relational** databases. NoSQL or XML databases are NOT allowed as they do not reflect the database relations we’ve covered in this class.

This is an individual project.

It is recommended that you choose from popular database software such as 1) SQL Server Express, 2) MySQL, 3) SQLite, 4) PostgreSQL, etc.

This project has three (3) parts. That is, there will be three submission threads on Blackboard. Please submit each part by its own deadline.

## Part 1: Project Proposal (2 points)

Each student needs to submit a proposal (MLA format, 2 – 3 pages) addressing the following topics:

* The business/story of your database
* Database software choice
* Database tentative schema
* Programming tools choice (language, framework)
* (Optional) AI assistant choice and usage. Please give credit (such as prompts you used and answers you get from AI assistant) to your AI assistant.

Submit your proposal document on Blackboard.

## Part 2: SQL Files (6 points), Submit on Blackboard and In-person Check with GTAs

This part has two requirements: 1) submit files on blackboard, and 2) check with GTAs. (03/31/25, Monday)

Please propose a database to model mock businesses, such as bike rentals, community libraries, employee management systems, and service billing software.

* The database should have at least 4 tables, where each table is in 3NF.
* Each table should have at least 4 fields (including primary keys)
* Need to list functional dependencies for each table (as the comment when creating the table.)
* (Optional) Please specify triggers and/or policies on foreign key constraints (if any.)

Submit the following files (do not zip) on Blackboard Database file submission:

* create.sql to create all tables.
* insert.sql to insert all rows to tables.
* crud.sql to perform sample crud operations on your database.

## Part 3 (Application + Git + Video + Blackboard submission)

Note: this part needs to submit only one file: project.md on Blackboard. However, you need to create a public repository on GitHub and an unlisted video on YouTube, then add the two links to the project.md to complete the requirement.

## Application Requirements (no separate submission)

Student may choose their programming languages/frameworks to build a GUI application using your database. The app should serve as a simple front end to perform CRUD on your database. Also have some predefined SQL statements (such as SQLs performing join on multiple tables and subqueries) in buttons to display the results from your database.

Please include some pre-defined buttons in your app to demonstrate joint search results.

There is no separate submission requirement for this part. You will demonstrate your GUI app in your video (see project.md and Video Link.)

## Git Requirements (no separate submission)

Please create a **public** GitHub repo for this project and have at least **10** commits. Each commit should make non-trivial contributions, such as adding handlers to buttons and writing several SQL statements. Fixing typos in docs is important but should be considered only trivial.

***DO NOT PUT SENSITIVE INFORMATION IN A PUBLIC REPO.***

Write meaningful commit messages.

Please evenly distribute your workload and development progress throughout the project. “The ‘last few days' Effort” (e.g., all commits made only in the last one or two days of the project due date) will be penalized (5%—20% of the total project weight) on grading.

There is no separate submission requirement for this part. You will add the link to your repo to the project.md file.

## project.md and the Two Links (7 points)

Please submit a **project.md** file to Blackboard that includes:

* + (IMPORTANT) the link to your project’s GitHub repo.
  + **(IMPORTANT)** the link to your project’s demo video (3 – 5 minutes, **unlisted** on YouTube). Please explain the following points in your video:
    - General description of your database, including the choice of DB software (e.g., SQLite)
    - A running demo of your application.
    - An overview of your GitHub repo’s commits.