

Drone Grocery Delivery

CS 4400: Introduction to Database systems

Spring 2021: Semester Project

Project Purpose

In this project you will analyze, specify, design, implement, document, and demonstrate an online system. You are required to use the classical methodology for relational database development. The system will be implemented using a relational DBMS that supports standard SQL queries. You will use your localhost MySQL Server (Version 5.1 or above) to implement your database and the application. You also cannot use any other software like Access or SQLite. Ask the professors or TAs if you have questions.

Project Phases

Inputs (we give you)

<ul style="list-style-type: none">• Text and UI description	<ul style="list-style-type: none">• Advanced ERD• Raw initial data	<ul style="list-style-type: none">• Database schema• Initialized database• Procedure shell	
Phase I	Phase II	Phase III	Phase IV
<ul style="list-style-type: none">• Entity Relationship Diagram• Logical constraints	<ul style="list-style-type: none">• Relational schema• Database schema• Initialized database	<ul style="list-style-type: none">• Implemented procedures	<ul style="list-style-type: none">• Fully functional application

Outputs (you turn in)

Directions for Phase III

In this phase, each team will implement a list of stored procedures for the Grocery Drone Delivery System.

We are providing you with the following:

Title	Type	Description
Procedure Bank	Excel	Provides descriptions, points values, notes, and sample calls for each of the specified procedures.
Procedure Shell	.sql	Stubs out all of the procedures and provides space for you to fill in your implementations.
Database initialization	.sql	Provides the DDL to set up the database you will be working with as well as inserts the initial data from Phase II.
Stored Procedure Guide	PDF	Briefly reviews the various parts of stored procedures in a concise one-pager.
Project Description	PDF	The same project description from Phase II for your reference.
Phase III Instructions	PDF	This document, which outlines the instructions for Phase III.

Important notes:

- There has been a few minor changes to the project description, and thus the relational schema as well – this was done for your benefit, to make the queries a bit easier.
 - Please review the changes noted in the description document.
 - They are listed in **Orange**
- The only thing you will turn in is the procedure shell (named as `cs4400_phase3_shell.sql`). The rest is for your use.
- Each procedure will be graded on an all or nothing basis.
- You'll notice we have already provided filled in implementations for a few procedures. These are for you to use as examples and are worth 0 points.
- We expect your procedures to be compatible with the schema we have released. The schema we give you will be the one we use to initialize our database before testing your procedures.
- Don't change the stored procedure names, any of the parameter names, the order of the parameters, or the parameter types. These are essential to the interface of your application: if you change them, you would impact how users are able to access your application, and you could potentially impact other systems (e.g., our testing application) that are used to evaluate your submission.
- Some of the parameters for the stored procedures might be empty. In those cases, the input value will be a NULL. If you look at the project description, this case would occur when a user submits an empty textbox or an 'ALL' in a dropdown.

- For SELECTs:
 - the "SELECT * FROM USER" is just a dummy query to get the file to run. You will need to replace that line with your solution.
 - the stored procedure creates a table that has the same name as the stored procedure along with the string "_result" appended to the end. Your query return results must be stored into that table.
 - Do not change the names or parameters of the tables that are used to return the results of your queries.
 - However, you are welcome to move around the "INSERT INTO ____result" command wherever you need it to be, as long as the results of your SELECT are loaded into that result table.
 - All decimals should be rounded to 2 decimal places using the function ROUND(..., 2) and NULLs that should be numbers should be replaced with 0 using IFNULL(..., 0).

Submission Instructions

1. Your file must be named cs4400_phase3_shell.sql (no caps, exactly like this).
2. You will be submitting through Gradescope and this phase will be autograded.
3. Gradescope will run a preliminary autograder on your submission and output results for diagnostic and debugging purposes only. These results should not be assumed to reflect your actual grade but are a relatively strong "sanity-check" for your procedures and submission process. We will actually grade your submission on a more comprehensive autograder, so be sure to manually test your procedures as well.
4. Again, **test your procedures** yourself against edge cases, various provided arguments, etc. The autograder is there to help, but it is not your final grade.
5. You may resubmit as many times as you want, but the last submission will be used as your final one and will be the one used to count late days.
6. The course staff is still setting up the autograder and Gradescope, and we hope to release this by April 9th.

Grading Details

This portion of the project will be graded out of 33 though there are 37 total available points from the stored procedures. Should you find a solution for the remaining 4 points, they will act as a bonus and will factor in when doing overall/final grades. Actual impact of these bonus points is yet to be determined.