

# CPSC 304 Project Cover Page

Milestone #: \_\_\_\_3\_\_\_\_

Date: \_\_\_\_28/07/22\_\_\_\_

Group Number: \_\_\_\_39\_\_\_\_

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Jack Tam	38017240	t9v4h	jackjktam@gmail.com
Carmen Yu	11828167	w7t0b	cjyu98@gmail.com
Steven Mezei	79000873	e2s1b	stevenmezei@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Populate the SQL database

- SQL script to create all tables and data CARMEN

Load JDBC Driver onto our application JACK

Implement the login feature STEVEN

Allow SQL commands to be done on application

- Insert (provide interface for user to specify input) JACK
  - Insert new employees, ingredients, suppliers etc.
- Update (interface to provide input for update) STEVEN
  - Update employee payroll, ingredient price etc.
- Delete (must be cascade-on-delete) CARMEN
  - Remove menu item, employee etc.
- Selection (user specified values of selection) JACK
- Projection (can be hardcoded select statement) STEVEN
- Join (joins at least 2 tables and performs meaningful query + interface for user to execute query, user must provide at least 1 value in WHERE) CARMEN
- Aggregation with GROUP BY (interface to allow min/max/average/count) JACK
- Aggregation with HAVING (query which requires HAVING) STEVEN
- Nested aggregation with GROUP BY (use a nested subquery e.g. finding average number of items purchased per customer, subject to some constraint) CARMEN
- Division (need interface) STEVEN

Timeline

- Aug. 1st: Configure Insert, Update and Delete
- Aug. 3rd: Selection, Projection, Join Completed
- Aug. 4th: Login feature has been configured
- Aug 6th: Aggregations and Divisions have been finished
- Aug 7th: Implementation will be finished