University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: _____3_____

Date: ___03/07/23___

Group Number: ___34_____

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Amy Jo	88162888	k4y2b	amyjojjm2001@gmail.com
Sophia Lee	57671240	d7u5t	sophia.ahyl@gmail.com
Bloom Soh Pei Xuan	79461265	f8z3q	bloomspx@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

1. Brief Summary

An on campus bike sharing club/program for university students. Students can pay an annual membership fee to become a member which will allow them to book the club's bicycles and facilities. Students can also volunteer at the club while members can attend member-only social events or lessons.

2. Timeline and task breakdown/assignment:
Very Rough Lofi Outline
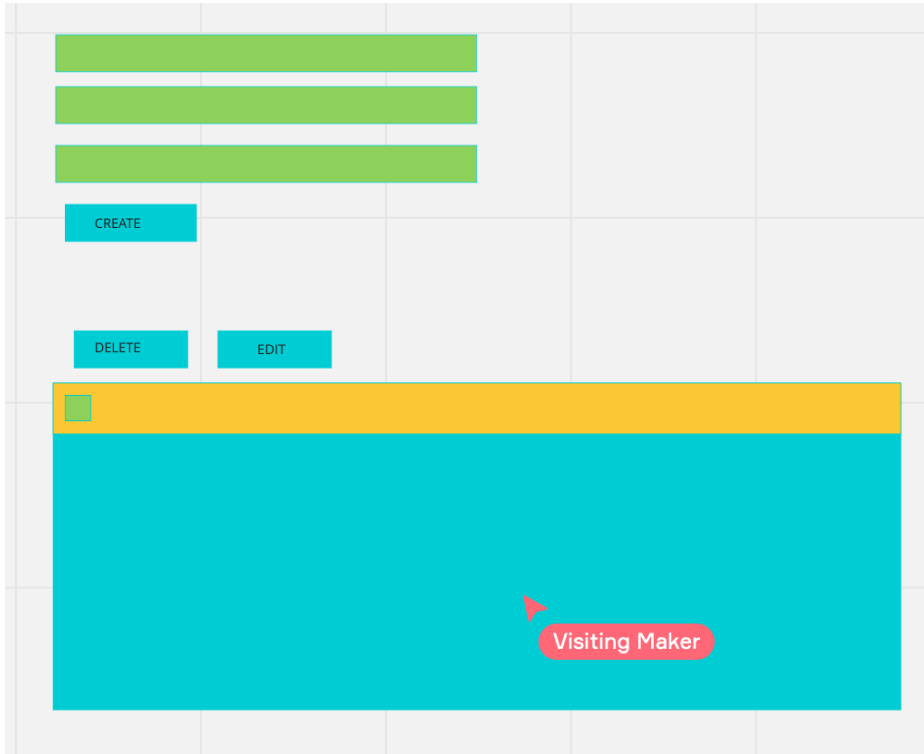
Questions for TA
- Should entities be created through the frontend or Oracle?
- Do we need to be able to do sql operations on ALL entities?
- "The query is clearly listed" in M4 - where does it need to be listed? Frontend? Or just found in our code?
- Basic Security Practices: Sanitization
    - Values from the user are not directly used in the database. Basic security practices have been followed.

Summary of Tasks + Assignments

DUE DATE	TASK	NAME
March 14th	CRUD frontend + data sanitization	Amy
March 14th	SQL Script to make the tables	Sophia
March 28th	Backend CRUD operations for all entities	Amy
March 28th	Backend + Frontend implementation of selection, projection, join	Sophia
March 28th	Backend + Frontend implementation of aggregation group by and having, nested aggregation, division	Bloom
April 1st	Frontend for end product	Amy

AMY: Frontend + CRUD backend logic
SOPHIA: SQL Script + selection to join (backend)
BLOOM: Data sanitization + Division to 3 aggregations
Error handling: ALL

3. Description of Challenges
Frontend
- Assume that all entities can be created through the frontend
    - Reading / Updating / deleting
        i. Show a table to show all entities in the table
            - Should be able to delete an entity by selecting on a checkbox on an entity and clicking on the delete button
            - Should be able to update the table by selecting a checkbox on an entity and clicking on the edit button. This should enable inline editing mode.
    - For creating:
        i. A dropdown should be available to show the entity type the user is creating
        ii. Inputs for adding attributes
        iii. Error handling if required inputs or dropdowns are empty
- Selection
    - Once a user has selected a table on a dropdown, the user can see a dropdown of the entity's attributes and an input where the user can type to query by
- Projection
    - Once a user has selected a table on a dropdown, have checkboxes of the attributes that the user can select to see only the selected attributes in the table
- Join
    - User can select a table on each of the two dropdowns and just click the Join button
- Aggregation (2) + Nested
    - A user can select on a dropdown to choose which attribute to query by and another dropdown for the types of group by query

- For utility purposes, we need a dropdown at the top of the table to show only the entity type we're interested in querying

Backend + Database
- Learn PHP
- Setup Oracle database
- Write SQL script to make the tables
- Create PHP API so backend can communicate to the frontend and vice versa
- Do queries including
    - INSERT
    - DELETE
    - UPDATE
    - Selection
    - Projection
    - Join
    - aggregation with group by
    - aggregation with having
    - nested Aggregation with Group By
    - Division