

# CPSC 304 Project Cover Page

Milestone #: 1

Date: 2024/07/11

Group Number: 33

| Name                    | Student Number | CS Alias (Userid) | Preferred E-mail Address |
|-------------------------|----------------|-------------------|--------------------------|
| Kevin Chu (Chao-Wu Chu) | 85406312       | b0u5x             | kevinchu6601@gmail.com   |
| Kaicheng Lu             | 14723720       | o1u1i             | kaichenglu2020@gmail.com |
| Jason Liang             | 75499566       | w3s8d             | cajasonliang@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

## **Project Description**

The domain of the application is food delivery service for all customers. Our application allows individuals to order food from restaurants and have that food delivered through courier drivers. We aim to facilitate how people buy and get their food on a single and trusted application platform. To build this, some main aspects of food delivery service are required, including account management, restaurant and menu searching, restaurant and delivery partnership, order placing, payment handling, delivery processing, and customer review.

The platform provides customers convenience and variety by making online orders from the menus of partner restaurants. These orders and following payments are immediately handled by the platform system keeping all transaction records. A trustworthy source of restaurants and delivery is established by the platform's verification and customer review mechanism. Three real-life use cases of the application are given as follows. In case one, a potential customer may feel hungry but refuse to go out because of bad weather or just being lazy, he or she can solve this problem by making orders easily on our platform. In the second case, another potential customer feels annoyed with accessing different restaurants' websites when deciding what to eat for dinner. Our platform provides integrated information about all partner restaurants, such as ratings and menus, with only one single account required. In the final case, one of our customers has a meeting that ends just before lunchtime, luckily he owns an account on our platform, which lets him make food orders quickly, track the process directly, and get his meal smoothly on time after the meeting.

We would like to build an application to match the advantages of relational databases, such as ACID properties, and explore more of its potential. A food delivery service relies on a database with correctness, consistency, and high security, especially when storing bank information and handling transactions. We consider this a good reason for the topic selection of CPSC 304.

### **Database Specification**

Users can create an account and choose to become a member, which will allow them to receive points, and discounted deliveries. Users will be able to browse a catalog of restaurants, filtering by cuisine, location, and rating, and view menus to place food orders. From there, they are then able to make an order, which involves making a payment to the restaurant for their order and the driver that will deliver their food, and be able to receive real-time updates on the estimated time that the food will arrive. Through the database, users can also write a review on the driver and restaurant, which for the restaurant and driver will have different rating items, while the user can choose to also add a comment. Users will be able to look at the history of their orders, allowing them to see the date, time, and payment amount of their previous orders.

### **Application Platform**

We plan to build a web application for the food delivery service. The technology stack in our project includes Java Spring MVC framework for front-end development and Java Spring Boot for back-end development. The database will be Oracle Database, while we will also use Java Spring Security for authentication.

## ER Diagram

The ER diagram includes 8 entities, 8 relationships, and 2 ISAs. Additionally, we have 3 weak entities (connected with ISA) with another 3 relationships connected with each of them. The primary key is marked with a bottom line for each entity, while each entity set has at least one non-primary key attribute. For entities of menus, orders, and payments, we used an artificial primary key for each to keep the design more simple and straightforward. To have better diagram clarity and scaling, we keep the entity and attribute on the top of the ER diagram, and each entity is connected with the attributes right below it. For example, the Food entity connects with attributes of foodName, foodPrice, and foodType.

