**ISM103 - Relational Database Systems and SQL**

**Assignment #3: Creating an ERD using PostgreSQL and pgAdmin 4**

**Cloud Gourmet**

Student: Şafak Atakan Çelik (CT1009470)

Date: Mar 4, 2025

This is an explanatory document for the schema prepared for the Cloud Gourmet described in the assignment document. Explanations below are categorized under concerns “users, purchases, subscriptions, deliveries, meal kits”:

* Users: There’s the “users” base class. Customers, delivery agents or other potential classes are to derived from it by having 1:1 relationship and referencing their ids to users(id). Common user details are stored in the users table and the users.role field specifices the type of user. In current design, if a customer has an active subscription with a portion size higher than 1, their account considered under the family plan, otherwise it’s under the individual plan.
* Purchases: Customers can browse meal kits stored in the database, add them to their cart which is handled on the front end. Once a user clicks on their cart and continues with the purchase; potential discounts are applied to their order total if any. Different discount types are stored in the discounts table. discount\_type clarifies the type of the discount, discount\_multiplier contains the multiplier formula to be applied on the order total (e.g. “n\*0.1”). orders.discounted\_total is not null only if there’s a discount applied. Once users click purchase, they’re redirected to the payment page which is handled outside. Database provides the necessary information for the transaction, and the link of the receipt received during a successful transaction stored in the orders.receipt\_link. Meal kits sale prices will be used for individual order prices; portion\_size, kits\_per\_week, and preferences will be used to calculate weekly recurring subscription prices.
* Subscriptions: Users can subscribe to a weekly meal plan by purchasing a weekly subscription product. They can create their custom subscription product on the UI. They need to select default portion size for their meal kits, the number of kits per week they’re subscribing to, types of meals they prefer and days of the week they prefer. Users can have 1 active subscription at a time, ended\_date’s state of being null indicates activity of a subscription. kits\_per\_week is the number of free kits that will be granted to the customer account. preferences is a text representation of a JSON to be able to store additional information if needed. While finalizing the subscription purchase users can select the next week’s dates of deliveries. There will be automatic meal kit suggestions, but users are allowed to change them. In addition to a subscription order which does not lead to a delivery by itself, each week on friday next week’s deliveries will be recorded as orders with the status “scheduled” based on the previous week and the users’ preferences. Users are always allowed to modify or turn off scheduled deliveries to customize their subscriptions further. They can create individual meal kit orders with the remaining credits as well.
* Deliveries: When it is time to deliver for a product (subscription’s scheduled order or individual order) order status converted into “in-transit” and a new delivery is created. An available delivery agent is assigned to the order; delivery and the pickup address (assigned delivery depot) shared with the delivery agent. This database only records fundamental information for deliveries, delivery agents are required to pickup, deliver and update the status of the deliveries.
* Meal kits: Kits have 1:1 relationship with products and reference their id to the product ids similar to subscriptions. Created\_at, name, sale\_price, description, type, detail are fields to be used for operations. Type could have been derived from ingredients, but it is included on the meal kit level as well to decrease the amount of search queries. Meal kits and ingredients have many to many relationships. For each ingredient there’s an ingredient supply contract which links ingredients and suppliers. For the same ingredient different contracts can be made with different suppliers. Existence of an ended\_date value indicates activeness similar to the subscriptions and discounts.