DBMS Project Synopsis
By :
Madhav Sainanee (2017063)
Tanmay Bansal (2017268)

Swaggy - Food Ordering/Delivery system

About the project -

Swaggy is a food ordering system for our restaurant Foodology, based in our own city, New Delhi! Through our implementation, we intend to design a system for users to easily have food delivered to their doorstep, for us to keep track of all the orders that they must fulfil, and delivery personnel to seamlessly deliver happiness to homes across the city.

The working of Swaggy is rather simple. The **customer** orders food via Swaggy and the restaurant is alerted. Then one or more **chefs** is assigned to prepare the **food items** in the **order**. After the food is cooked and packed, a **delivery boy** is assigned to deliver the lip-smacking delicacies to the customer's doorstep.

Let us have a look at the different entities involved in this entire process.

1. Customer - The Customer is the entity that starts the process by selecting food items to order from Swaggy. A customer is uniquely identified by his customer_id and the database also stores name, address, phone numbers(s) and cart information.
2. Order - The order is the heart of the food-ordering process. It is uniquely identified by an order_id, and also consists of the date and time the order was placed on, the status (delivered or not), and the total price. It contains one or more food items.
3. Cart - The cart holds the food items that the customer orders from Foodology. It is uniquely identified by cart_id, and also contains total price.
4. Food items - Food items are the many dishes that Foodology prepares. It is uniquely identified by an item_id. Consists of the name of the dish, a small description, the price and the quantity ordered.
5. Chef - The chef prepares food items in an order that has arrived. One order will be assigned to exactly one chef. However, one chef may be working on 2

different orders. Chef is uniquely identified by a chef_id and the database also stores personal information like Name, birthdate, phone number etc and also salary that the chef is earning and login details.

6. Delivery Boy - After the food is prepared, the order must be delivered to the customer. This is the role of the Delivery Boy. The Delivery Boy is uniquely identified by deliveryboy_id. Furthermore, the database also stores personal information like Name, birthdate, phone number etc and also salary that the delivery boy is earning and login details.

Let us now look at the different relations:

1. Customer **places** order - 1 to many

Customer places an order at Swaggy. Furthermore, there is total participation of order entity.

2. Customer **has** a cart - 1 to 1

Every customer orders through a cart wherein he selects food items. There is total participation of both customer and cart entities.

3. Cart **contains** food items - many to many

Every cart may contain many food items and a specific food item may be present in many carts. Included quantity of food item present.

4. Order **includes** food items - many to many

Every order consists of one or more food items. Similarly a food item may be present in more than one order. There is total participation of order entity. Included quantity of food item present.

5. Chef **prepares** order - 1 to many

Every order is prepared by one chef. However, a chef may be preparing one or more orders. There is total participation of order entity.

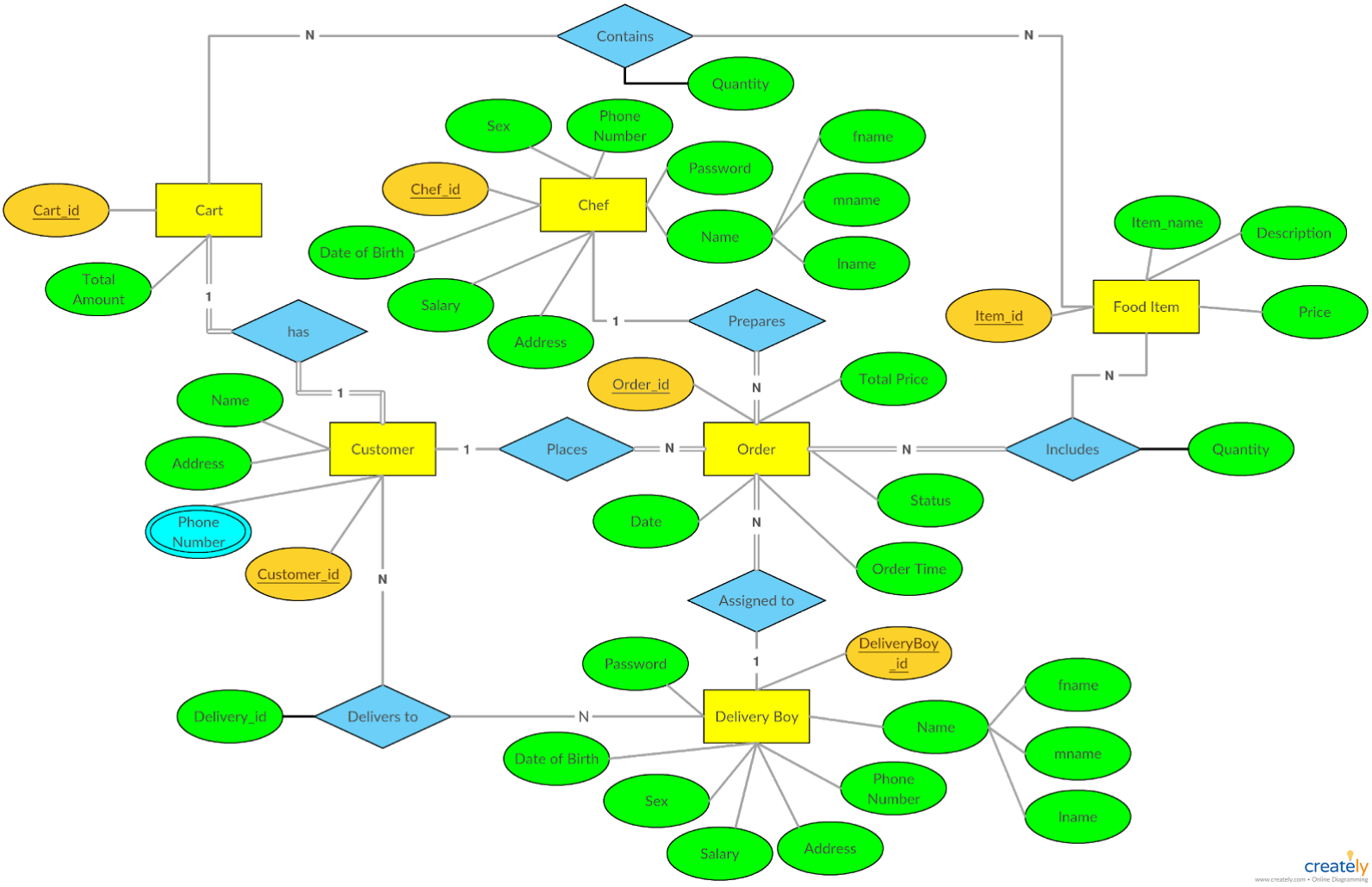
6. Order is **assigned** to Delivery boy - many to 1

Every order is assigned to exactly one delivery boy, but a delivery boy may be assigned many orders.There is total participation of order entity.

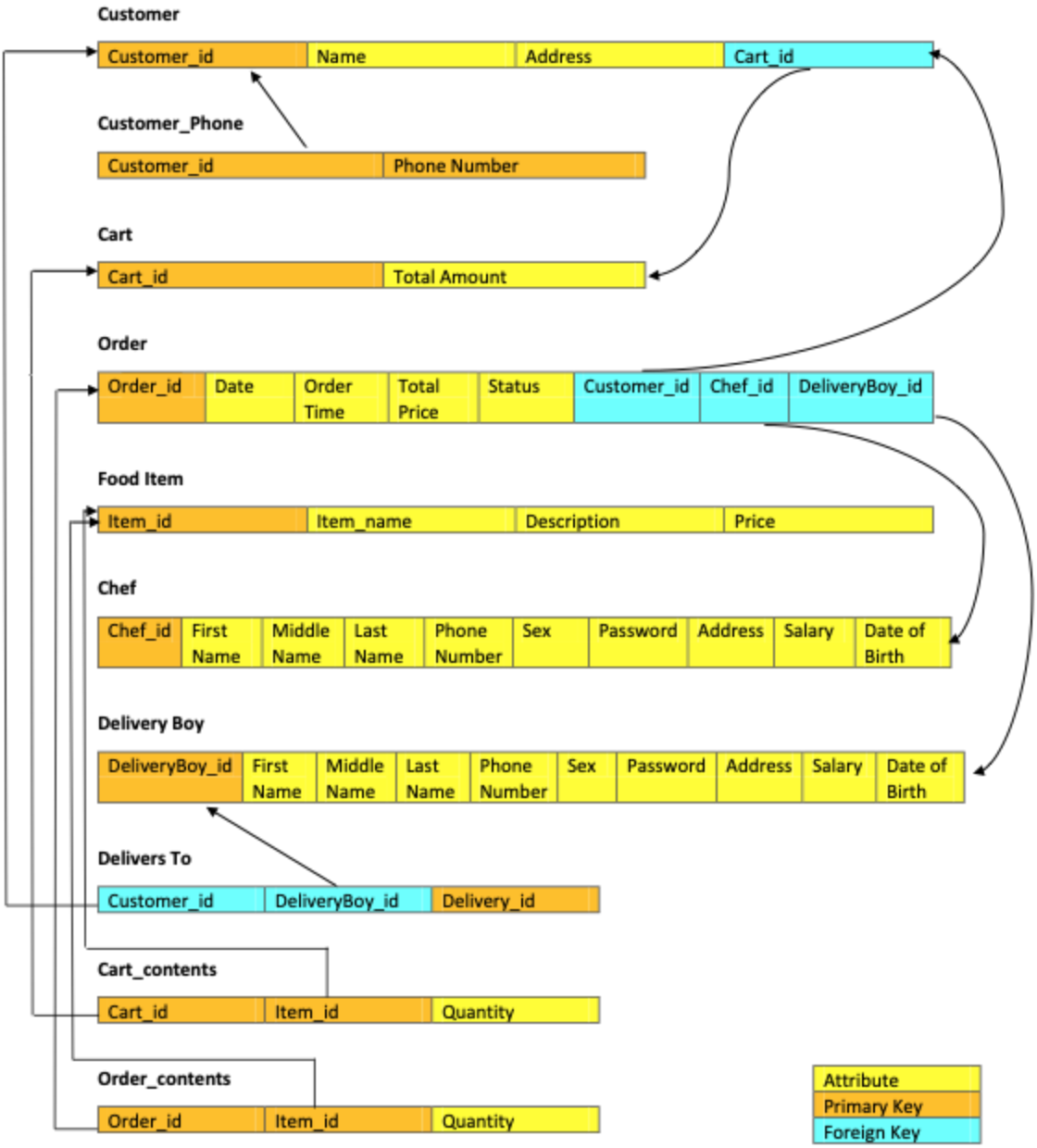
7. Delivery boy **delivers to** customer - many to many

A delivery boy may deliver to multiple customers, and customers may have orders delivered by multiple delivery boys. Included delivery_id.

Entity-Relationship Diagram



RELATIONAL SCHEMA



Implementation Platform Details

Python and MySQL