Assignment 1

Part 1

Bounding contexts

We have identified the following core bounding contexts: Users, Menu, Stores, Orders. As a generic domain we have identified the context Authentication. As a supporting domain we have identified the context Coupons.

The core bounding context Users contains in itself the information about a specific user's allergies and their preferred location, and the users privileges. It is upstream to the Menu and Orders context as it is important for them both to be aware what are the users allergies and it is important for Orders to be aware of the subscription plan.

The core context Menu contains information on the available assortment of pizza, default and available for configuration toppings for each given pizza. It also contains the list of possible allergens that each pizza contains. This content should also be able to filter the results it is returning based on the request. It is downstream to the Users context as it retrieves the allergies of the user requesting and returns an alert for every pizza shown to the user that conflicts with their allergies.

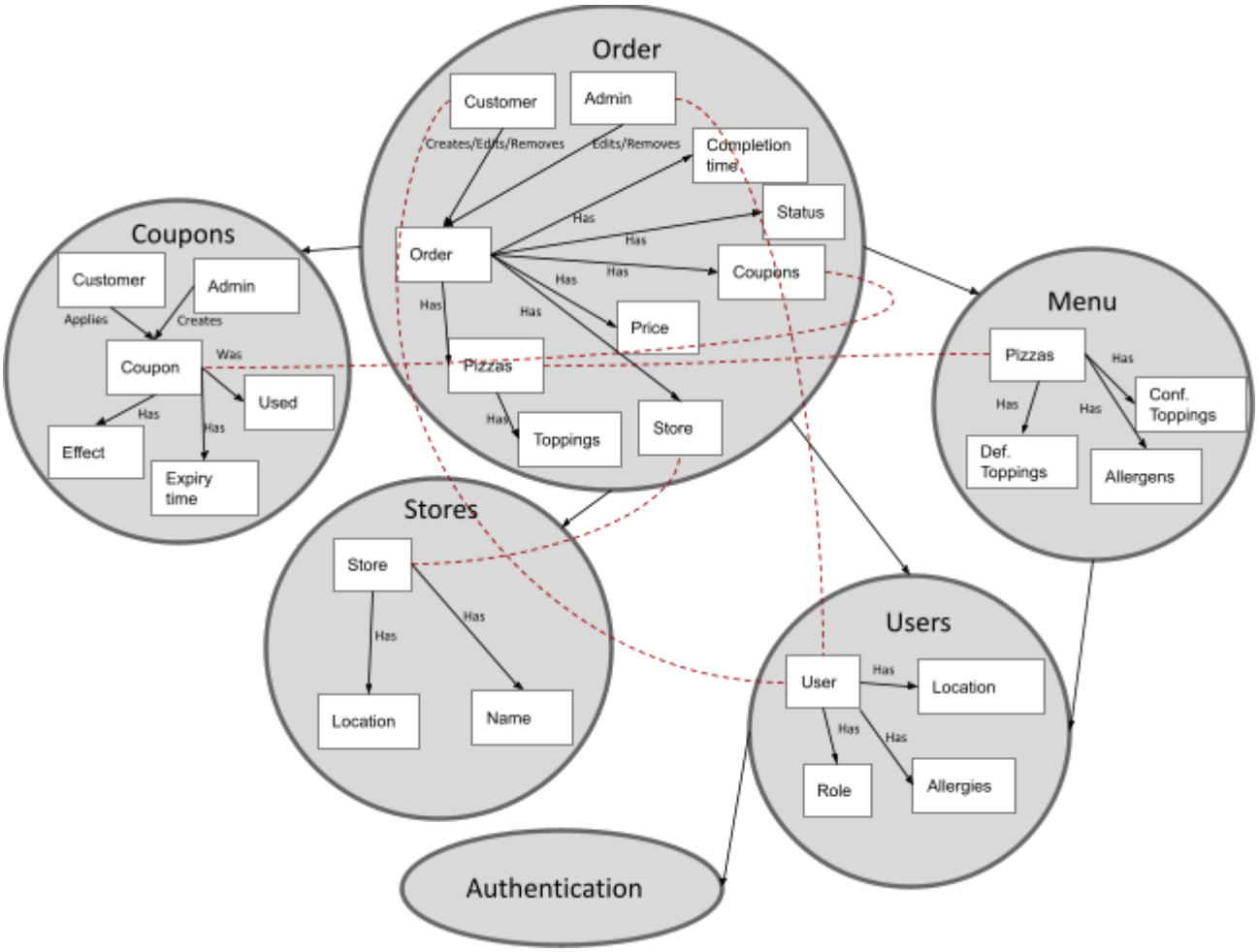
The core context Menu contains information about the stores in the chain. Each store has a location and a name. This context can be queried to present the list of the locations available to place an order to. This context is upstream to the Order context.

The core context Orders handles the information about the currently active orders. Any order that is not yet delivered is considered active. For each order there are currently 3 identified stages: collecting, placed, delivered. While the order is in the collection stage new pizzas and coupons can be added/edited/removed to it. This context is downstream to the Users, and Coupons, and Stores contexts. From Users it requests the allergies (to raise alerts of conflict) and subscription of the user. To Coupons it hands over details of the order such that the coupons can be verified and can be applied to the order in an optimal manner, it receives back the price and the coupon applied. Once the collection stage is done the order is considered placed and can be edited by managers/regional managers and the user (if it is within 30 mins of completion). The order can be marked as delivered upon completion by managers.

The generic context Authentication handles the authentication of the user within the system and provides them with a token that corresponds to their privileges within the system and uniquely identifies them.

The supporting context Coupons is upstream from the Orders context, it contains the currently available coupons and the information about what each coupon entails (it’s effects, expiry time). It receives the information about the order and calculates which coupon to

apply and what is the price after application. Coupons can be used by the customers and added by admins.



Microservices individually explained

1. Authentication
   The Authentication microservice is responsible for the security of the application. It will be able to perform two types of operations: account creation and JWT Token generation. The account will be created with an email and password. The JWT Token generation will generate a token that will include the user and its permission in addition to different security parameters (such as expiry time).

   The role of this microservice is to enhance the security of the service by reducing the risk of user impersonation. It will also store the users with their email and password.

We have decided to have a separate microservice for authentication to allow an increase in the security of the service (eg. adding more proxies) without adding overhead to the rest of the business logic.

2. User
   The User microservice keeps track of allergies that the user has and any privilege they may have. The user is authenticated using the token given from the authentication microservice. The allergens are polled by the menu and order microservices in order to filter out items or notify the user that they might be allergic to the item.
   The User microservice also determines whether a user is a customer or an Admin that can view all orders that have been placed and edit them. The User microservice takes in a token that is used to fetch the user's information from a database.

   We have created a microservice for the users because we feel that it is best to store sensitive information about a customer in a secure way (using authentication).

3. Menu
   The Menu microservice is responsible for providing the available pizzas to order from. It will be able to provide all pizzas on the menu with their default toppings while being able to filter them out based on the allergens of the user.

   The role of this microservice is to provide the users with the menu of the pizzeria so they know what they can order from.

   We have created a microservice for the menu because it has a semantically different meaning than ordering and people might just want to make an order without seeing the menu or the other way around.
4. Order
   The order microservice is responsible for handling orders. It does this in a few ways. It first queries the user microservice to get the user's allergies and finds any conflict with the allergies and an order. Upon finding a conflict, it notifies the user that they may be allergic to the item(s).

   If the user applies coupons, the order is sent to the coupon microservice and gets back the coupon that is used and the final price of the order. Once an order is placed, it writes it to a database or adds it to a data structure.

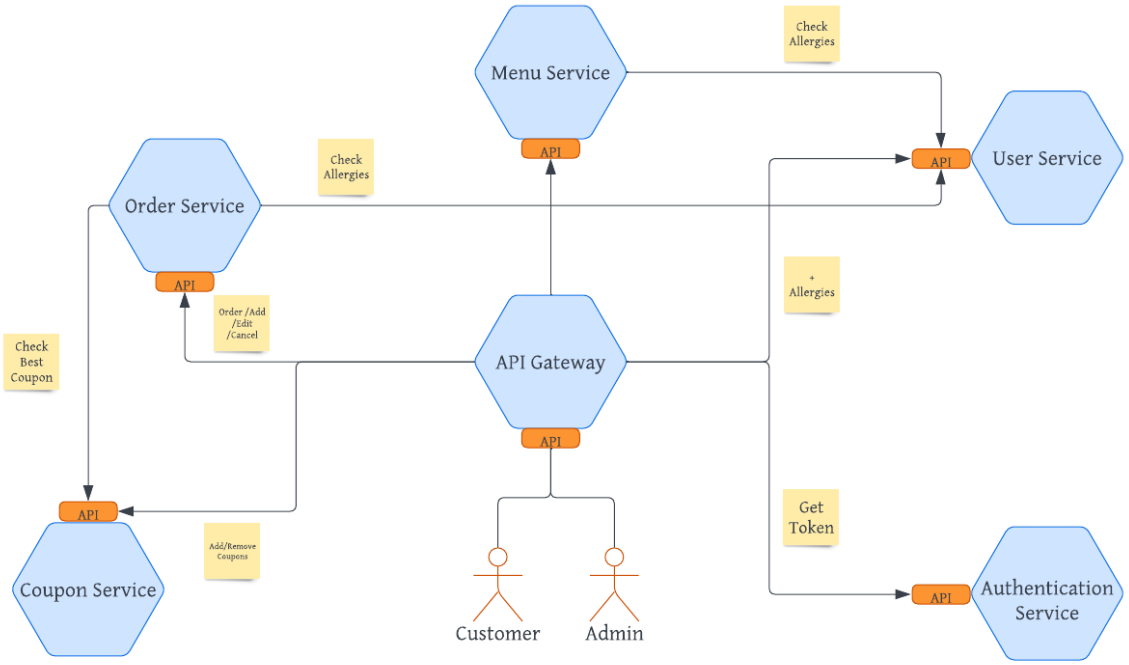
   The order can also be removed or have items added or removed from it 30 minutes prior to the selected time. It then gives notifications to the user on any updates to the order (on the status or if any changes have been made).

5. Coupon
   The Coupon microservice is responsible for validating coupon codes and for choosing the best coupon code for minimizing order type. This microservice allows new coupon codes to be added and also to take a possible order and to check if a coupon code gives a better price and return the new price.

The role of this microservice is to contain the business logic for coupon codes and also store the available coupon codes.

We decided to have a separate microservice for handling coupon codes because over time multiple types of coupon codes might be added to the service. If the coupon logic is encapsulated outside of the system it will improve deployment times, reliability and robustness. If something fails when adding the logic and it makes the Coupon service crash, Ordering will still be possible so it would not affect that many consumers. Moreover, having a separate system with the business logic will make the system easier to test.

How they work together



Microservice Architecture

All the microservices propagate through the gateway. As a customer, I will use the authentication service to create an account and receive a token to authenticate myself. Once I have logged in, I will be able to set up my allergens through the user service. I will be able to choose at which store I would like to collect my pizza through the stores service. The menu service will help me to fetch all available pizzas from the pizza database so I can decide which pizza to order (or not), pizzas that contain the allergens can be filtered out during the fetching process. I will be able to add my pizzas to the cart through the order service, the system will notify me if there are any allergies. Besides, I would also like to use some coupons, this is done through the coupon service. After making my choice, I can

always edit or remove pizzas from my order through the order service, the system will notify me and the store when there are any changes. Once I have determined everything, I will check out through the order service again.

As an admin, I will use the authentication service to create an account and receive a token to authenticate myself. I can edit the set of coupons through the coupon service by adding new coupons or deleting existing ones. In addition, I want to be able to manage the existing orders, if needed, I can cancel them, this is done through the order service.

Part 2