Aineopintojen harjoitustyö: Tietokantasovellus

Pizzapalvelu

Atte Keltanen

Introduction

The goal of this project is to offer a pizza ordering service which will be secure, convenient, and easy for the customer to use. There will be support for kebabs and drinks as well.

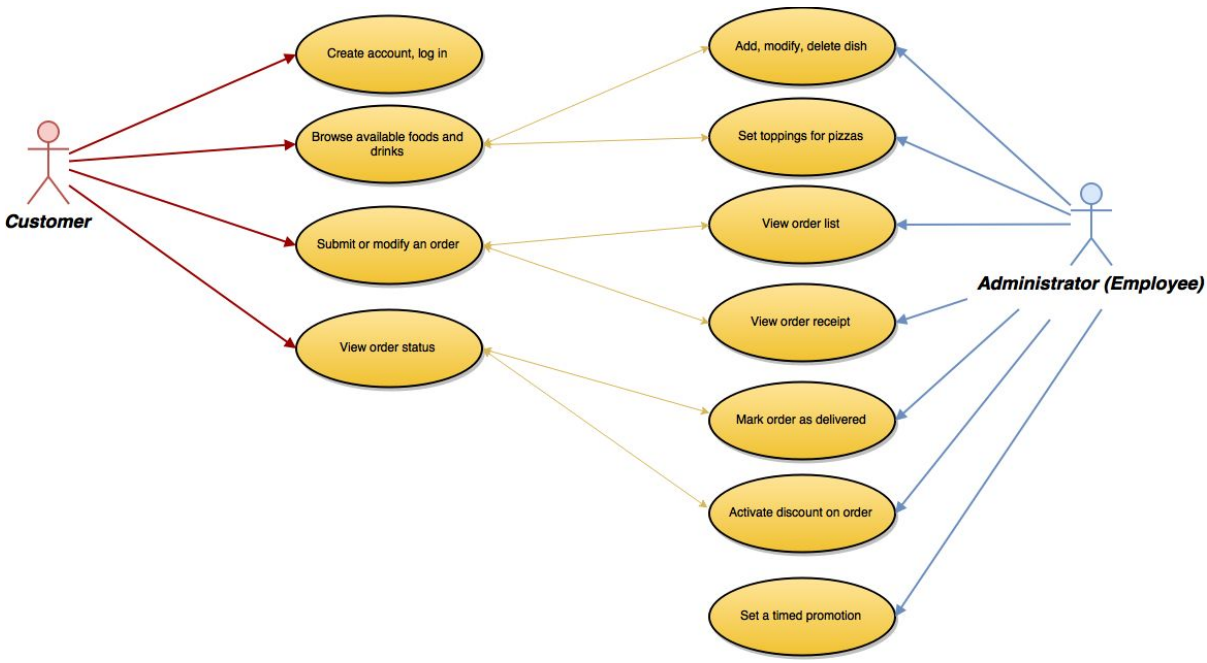
The project will be done in English to target a wider audience.

The back end will run on Python with Flask. The front end will very likely use React. PostgreSQL has been selected as the database of choice.

Heroku is supported out of the box and will be used to host the development version.

Use cases

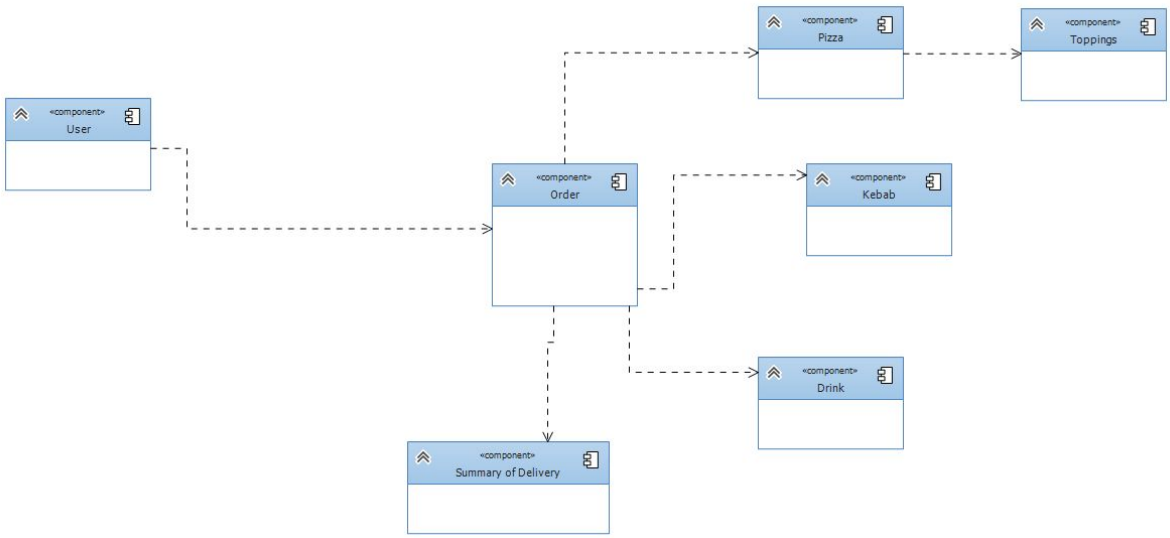
User types:
- Customer: A customer who wants to order food.
- Administrator (employee): An administrator who will fulfill orders. In this case an employee.



Customer:
- Account creation and logging in: The customer can create an account and log in with it.
- Browsing available pizzas, kebabs, and drinks: The customer can view the available offerings.
- Submitting and modifying an order: The customer can create an order with selected products and modify it if it hasn't been fulfilled yet.
- Viewing order status: The customer can see the status of their order.

Administrator:
- Adding, modifying, and deleting a dish: The administrator can add stuff for the user to order.
- Adding and setting toppings for pizzas: The administrator can tweak their pizzas.
- Viewing a list of active orders: The administrator can view what orders need to be fulfilled.
- Viewing an order receipt: The administrator can view the details of an order.
- Marking orders as delivered: The administrator can confirm orders as fulfilled.
- Activating discounts on orders: The administrator can add a discount on orders that were delivered late.
- Setting a timed promotion: The administrator can set a lunch offer on selected dishes.

Data content



Component: Pizza

Attribute	Value type	Description
Name	String (maximum of 512 characters)	Name of pizza ‘bolognese’
Price	Money	Price of pizza
Image	String (maximum of 512 characters)	Image of pizza

Component: Topping

Attribute	Value type	Description
Name	String (maximum of 512 characters)	Name of topping
Price	Money	Price of topping

Component: Kebab

Attribute	Value type	Description
Name	String (maximum of 512 characters)	Name of kebab
Price	Money	Price of kebab
Image	String (maximum of 512 characters)	Image of kebab

Component: User

Attribute	Value type	Description
Username	String (maximum of 512 characters)	Username the user uses to log in
Password	String (maximum of 512 characters)	Password the user uses to log in
Registration date	Date with time	Timestamp when the user account has been created
Admin status	Boolean	Designates whether the user is considered an admin or not

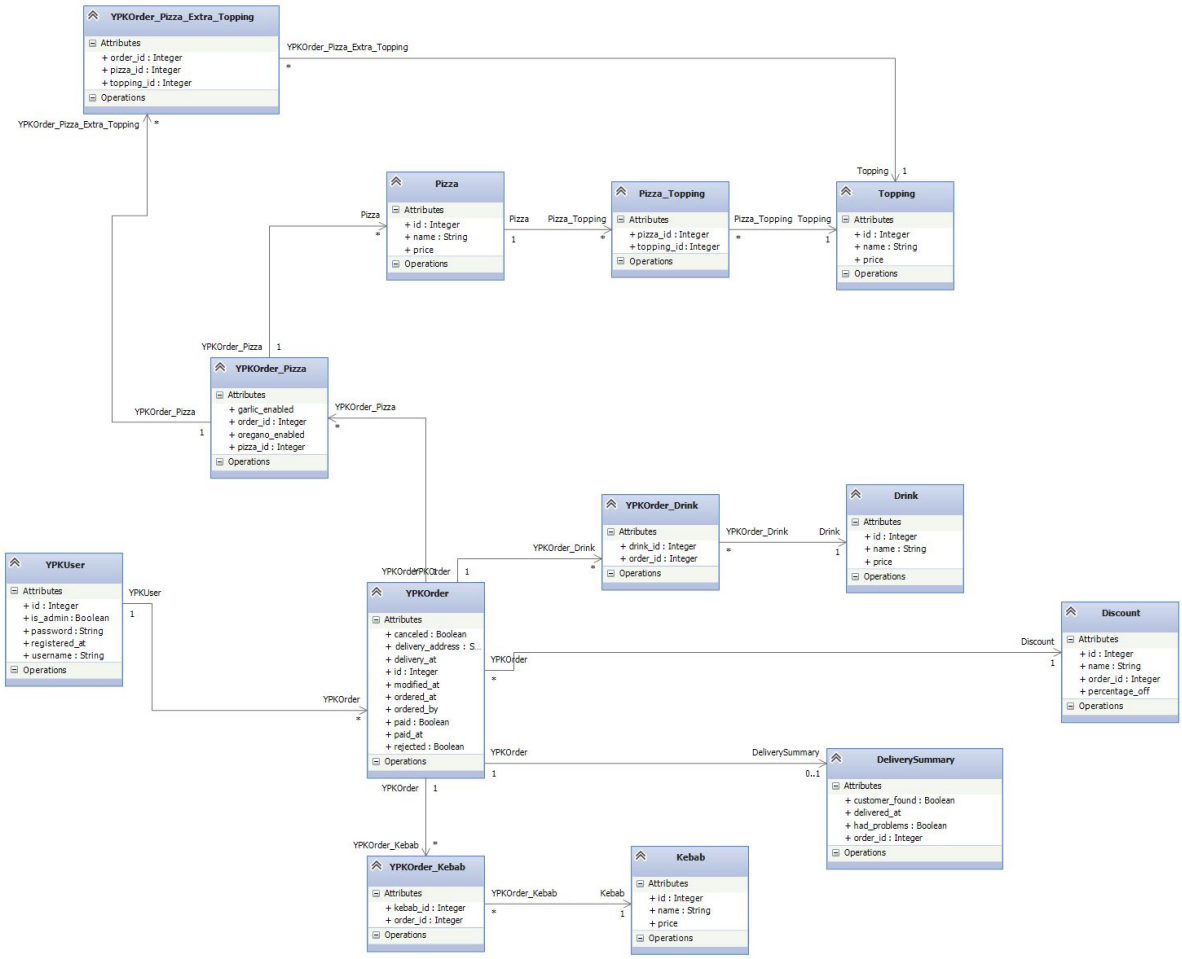
Component: Drink

Attribute	Value type	Description
Name	String (maximum of 512 characters)	Name of drink
Price	Money	Price of drink
Image	String (maximum of 512 characters)	Image of drink

Component: Order

Attribute	Value type	Description
Ordered at	Date with time	Timestamp of when the order has been created
Modified at	Date with time	Timestamp of when the order has been modified
Delivery address	String (maximum of 512 characters)	Address to deliver the order to
Delivery at	Date with time	Agreed delivery time
Canceled	Boolean	Set to true if the user has canceled the order
Rejected	Boolean	Set to true if the seller has rejected the order due to some reasons

Database diagram



All prices are in ‘money’ value type.
Modied_ats, ordered_ats, etc are timestamps with timezone.
Ordered_by is a foreign key to YPKUser.
Percentage_off is an integer.

There still may be some important things missing, but for now the database looks like this.