

2021

## MA-08 - CdC Restaurants



Jeffrey MVUTU MABILAMA

Nithujan JEGATHEESWARAN

CPNV

08/12/2021

## Table des matières

Description .....	2
Objectifs .....	2
Objectifs – requêtes espérées.....	3
Contraintes.....	3
Autres réflexions de conception et nos choix .....	4
Concernant le MCD .....	4

## Description

Nous avons décidé de modéliser une base de donnée pour pouvoir fournir un service similaire à [just-eat.ch](https://just-eat.ch). Idéalement nous voulons une architecture qui permette de créer un service équivalent quelque peu simplifié.

Ainsi, en plus d'une liste de restaurant, nous garderons également leurs types :

- ◇ Au sens du type de cuisine (thaïlandaise, asiatique, africaine)
- ◇ Au sens de la gamme de plats disponibles (pizzeria ne propose quasiment que des pizzas. Un restaurant axé pâtisserie)
- ◇ Au sens du type de service : fast-food, restaurant familial, Food truck, ...

Nous tiendrons une liste d'ingrédients. Des plats peuvent contenir plusieurs ingrédients et peuvent avoir des ingrédients à choix lors de la commande.

Les différents types de plats (pizza, durum=kebab, etc.)

Nous tiendrons une liste d'allergènes connus.

Nous garderons aussi les avis clients sur les restaurants de notre base.

Nous garderons aussi trace des utilisateurs de notre plateforme :

- ◇ Les propriétaires, employés (ou plus généralement représentants) de restaurants
- ◇ Les clients passant par notre plateforme
- ◇ Les administrateurs du système

Les différents quartiers (NIP)

Une liste des commandes. (On ne traitera pas de réservations de restaurant).

Nous permettrons l'enregistrement de promotions (par ex. rabais, code promo, ...).

## Objectifs

Nous voulons pouvoir utiliser la base de données pour :

- Avoir un catalogue des restaurants dans différentes villes ou quartiers
- Permettre de rechercher des restaurants à recommander à des utilisateurs. (A la façon de just-eat.ch). Par exemple, en fonction de l'avis (reviews) d'un restaurant, de leurs allergies, des types de plats ou d'autres critères.
- Permettre de répondre à des questions concernant la cartographie du paysage alimentaire d'un lieu.
- On veut permettre à des restaurants de se faire découvrir par notre plateforme. (Des propriétaires de restaurant peuvent ajouter leur établissement).
- Permettre à des utilisateurs de passer des commandes ou des réservations.
- Permettre de passer des commandes ou faire des réservations dans ces derniers

- Un client désire manger des boulettes de viande ou un fallafel. Ce plat peut être commandé dans plusieurs types de restaurants (kebab, restaurant asiatique, restaurant africain).
- Qu'un client puisse rechercher un restaurant de type pizzeria-kebab qui propose des plats sans son allergène (p.ex céréales).
- [contexte] : On a un même plat (poulet curry) proposé dans plusieurs restaurants. Il est proposé avec des ingrédients différents selon le restaurant. Chaque restaurant doit pouvoir éditer la liste des ingrédients de ce même plat pour son propre restaurant (sans que ça modifie les ingrédients des autres restaurants). Il doit en outre pouvoir attribuer des prix différents pour chaque ingrédient supplémentaire (p.ex. salade à 2 CHF dans restaurant A, salade 4 CHF dans restaurant B, etc.)

## Objectifs – requêtes espérées

- Obtenir la liste des propriétaires du restaurant A ?
- Dans quels types de restaurants l'utilisateur X a-t-il déjà passé au moins 1 commande ?
- Combien d'utilisateurs ont bénéficié de la promotion K du restaurant A
- Quels types de plats a commandé l'utilisateur Y ?
- Seuls les clients d'un restaurant (utilisateur qui a déjà commandé) peuvent le noter (donner un avis). Quels utilisateurs ont le droit de noter le restaurant B ?
- Dans quels types de plats l'allergène H se retrouve-t-il le plus souvent ?
- Quels types de restaurants servent le plus de plats contenant l'allergène H ?

## Contraintes

- On considère que le **prix** des ingrédients est fixe pour tous les restaurants.
  - o On sait qu'en pratique, ça ne serait pas le cas, mais pour rendre le projet réalisable, on va poser cette contrainte.  
Si chaque restaurant pouvait avoir des prix d'ingrédients différents, on serait à la limite de l'utilité des relations d'un SGBDR par rapport à la redondance.
- On considère que les **plats** qui ont le même nom seront forcément les mêmes. Dans la réalité, ce n'est pas le cas car les ingrédients de base peuvent être différents pour un même repas selon le restaurant. Cependant, au vu de l'utilisation que l'on compte faire pour notre application, orienté client en recherche de plats, on va rester sur ce choix.
- De même, on va considérer qu'un même plat a toujours les mêmes **accompagnements** (i.e. les ingrédients à choix, par exemple le jambon sur une pizza mozzarella), peu importe le restaurant qui le propose. Dans la réalité, chaque restaurant aurait normalement à proposer ses propres accompagnements, mais nous allons simplifier cet aspect.

## Autres réflexions de conception et nos choix

- Les ratings seront des entiers.

### Concernant le MCD

Généralités :

- Nous axerons notre service sur la Suisse. Autrement, il y aurait des subtilités (prix, code postaux, autres subtilités non évidentes).

Utilisateurs :

- On veut qu'un utilisateur qui serait admin, propriétaire et client n'aie qu'un seul compte (à moins qu'il utilise plusieurs adresses emails). Ainsi l'utilisateur a le choix, certains sites internet font cela.
- On ne va pas considérer séparément l'adresse de facturation et l'adresse de livraison / habitation.
- Cardinalité : quartiers - utilisateurs
  - o Selon le type d'utilisateur l'adresse ne sera pas obligatoire
  - o Un utilisateur de l'application n'aura pas forcément besoin de fournir une adresse
- Pros
- Cons
  - o Si l'admin se fait voler son accès sur la partie acheteurs, c'est une grande perte de sécurité.

Ingrédients :

- Questions à se poser
  - o Est-ce qu'un ingrédient est juste ce que l'on peut ajouter à un plat que l'on commande ?
  - o Ou est-ce qu'on considère qu'un ingrédient est la composition d'un plat ?
  - o Est-ce que pour un burger, le type de pain est un ingrédient ?
- Considérations
  - o Si un propriétaire change le nom d'un ingrédient, ça ne doit pas changer le nom de l'ingrédient pour les plats de ses autres plats ni ceux de ses concurrents.  
Mais cette différenciation peut se faire au niveau du serveur web (PHP par ex).
- Note : on considère ici les accompagnements et autres aliments optionnels comme des ingrédients. Les ingrédients d'un plat proposé par un restaurant qui ne sont pas obligatoires ne seront pas listés dans la table.
  - o EX : pour un kebab le pain n'est pas considéré comme un ingrédient car il est obligatoire, tandis que la viande est un ingrédient car on peut choisir entre poulet et agneau
- L'unité de mesure pour un même ingrédient ne changera pas d'un restaurant à un autre
  - o EX : La sauce est parfois exprimée en gr mais aussi en ml, on pourrait choisir de ne prendre que des ml comme unité de mesure

#### Commandes :

- Problèmes à résoudre :
  - o Cas concret : 3 amis font commande
    - 1 kebab avec 3 tomates + 4 viande pour NJ
    - 1 kebab avec 5 tomates + 2 viande agneau pour JM
    - 1 kebab avec 2 tomates + 3 viande bœuf + 1 maïs pour JM
  - o Solutions
    - On utilisera une table intermédiaire « Détails de commande » pour stocker 1 ligne ci-haut. Elle sera liée à *plats* et *ingrédients* par associations.

#### Restaurants

- Les horaires
  - o On ne va pas faire une table pour ça. Plus simple avec du texte simplement.

#### Prix :

- Tous les prix sont en francs suisse (CHF)
- Si plusieurs monnaies étaient acceptés un même ingrédient aurait un attribut prix qui varierait d'une monnaie à une autre

#### Types de plats :

- On considère qu'un plat peut être de plusieurs types (même si souvent il n'y en a qu'un).
  - o p.ex. un fallafel peut être considéré comme africain et asiatique (à notre connaissance).

#### Quartiers :

- Les codes postaux seront uniquement considérés pour la suisse (de 1000 à 9999), tout comme les prix seront tous en CHF.
- Un quartier peut accueillir des personnes (utilisateurs, par exemple) ou peut accueillir un restaurant. Un quartier devrait accueillir au moins *l'un* des deux (restaurant OU utilisateur).