Solution technique.

OC Pizza Solution Technique

# Choix de l'architecture.

Une achitecture en micro service est recommander pour ce type de structure.

## Architecture microservices.

Dans cette architecture on découpe l'application en services indépendant les uns des autres. Chaque microservice est développé, testé et déployé séparément des autres. Ils échangent des données avec les autres à travers les différentes APIs qu'ils exposent. Ils suivent les règles suivantes :

* Ils implémentent une seule fonctionnalité.
* Ils peuvent être écrit avec des langages différents.
* Ils sont faiblement couplés car séparé physiquement les uns des autres.
* Ils sont facilement testés, déployés et maintenus.

Les microservices sont petits comparé à une application monolithique et ils peuvent fonctionner sur un même serveur ou des serveurs très proches. L'accroissement du temps de réponse dû à leur séparation est négligeable par rapport à une architecture monolithique. En outre comme ils sont petits, ils répondent plus vite.

### Scalabilité.

Pour répondre à la demande d'un site internet il est plus facile de dupliquer un microservice qu'une application monolithique. En plus les contrats cher l'hébergeur permet d'avoir des configurations flexibles. En outre on peut séparer les services qui demandent plus de mémoire et ceux qui demandent plus de puissance de calcul. On n'a pas besoin de refaire un contrat avec l'hébergeur pour faire passer de 2 à 5 microservices.

### Maintenance.

Par sa définition un microservice est un programme petit donc facile à maintenir et à faire évoluer. Il est plus facile aussi de rechercher et d'isoler une erreur dans 100 lignes de code que dans 10000 lignes. On peut aussi en cas de problème facilement arrêter et relancer un microservice sans bloquer les autres. On peut aussi porter un microservice dans un autre langage sans que cela ne perturbe les autres.

### Encapsulation.

On utilise des conteneurs comme Docker pour encapsuler les microservices avec les ressources dont ils ont besoin. Ils peuvent être écrit dans des langages différents et faire appel à des librairies ou frameworks différents. En cas de changement de ressource d'un microservice, les autres ne seront pas impactés.

**Décomposition en micro-service**

On scinde l'application en microservices qui sont peu couplés les uns aux autres. Suivant la montée en charge on pourra cibler les APIs à dupliquer pour pouvoir répondre à la demande. On peut faire un premier découpage en dédiant un microservice par package de la spécification fonctionnelle.

## Microservice commande.

C'est la partie utilisée pour la prise de commandes soit par le Client ou l'Employé. C'est lui aussi qui permet de faire évoluer le statut au cours de la vie d'une commande suivant les intervenants. C'est celui qui sera dupliqué pour répondre à la demande croissante des clients.

## Microservice stock.

C'est par se microservice que toutes les modifications et consultations de stock passent. Il modifie la base de données du stock lors de la validation d'une commande où la réception d'une commande fournisseur. C'est lui aussi qui permet de savoir si la quantité d'un produit est suffisante pour être vendu.

## Microservice client.

C'est par lui que transitent toutes créations, modifications, consultations et suppressions de client. Il renvoie les données de livraison du client lors de la création d'une commande. Il permet d'accéder à des informations sur les anciennes commandes des clients à but statistique ou marketing.

## Microservice gestion.

Il regroupe tous les indicateurs et les informations sur les ventes, les coûts de revients, les profits, les pertes pour faire des statistiques. Il permet aussi d'effectuer de la gestion administrative du magasin avec la production de documents pro-formats.

## Microservice authentification.

C'est le service indispensable pour gérer les accès au système pour les Clients et les Employés d'OC Pizza.

### **Base de données.**

Serveur Posgres SQL pour une gestion efficace des différent jeux de donnée à distance avec une interface graphique.

Choix technologie de développement.

***Site web.***

HTML5

CSS 3

Mobile.

Java