

Conception-Développement d'Applications et Services Web / Serveur

Application LeBonSandwichEnLigne

Description Générale

LeBonSandwich est un vendeur de sandwich à la carte bien connu sur la place locale, caractérisé par le type et la qualité des produits proposés, issus de producteur locaux et en majorité avec le label "Bio". Pour garantir la qualité et la fraîcheur de ses produits, tous les sandwiches sont réalisés au moment de la commande, ce qui peut conduire à des temps d'attente parfois un peu long. Pour améliorer cela, la boutique souhaite se doter d'un service de commande en ligne de sandwiches.

Le principe est de commander son (ses) sandwiches à l'aide d'une application web ou d'une application mobile. Ces **webapp client front-office** fonctionnent sur tous types de terminaux. Elles permettent de consulter le catalogue, créer, payer et suivre une commande (payée, en cours de préparation, prête ...) et d'accéder à un service de fidélisation en ligne.

En complément, une application de **backoffice** permettant la gestion et le suivi de la fabrication des commandes est utilisée par le point de vente. Cette **webapp point de vente** permet de visualiser les commandes et les paiements, d'enregistrer la prise en charge d'une commande et sa fabrication puis sa livraison.

Architecture des applications

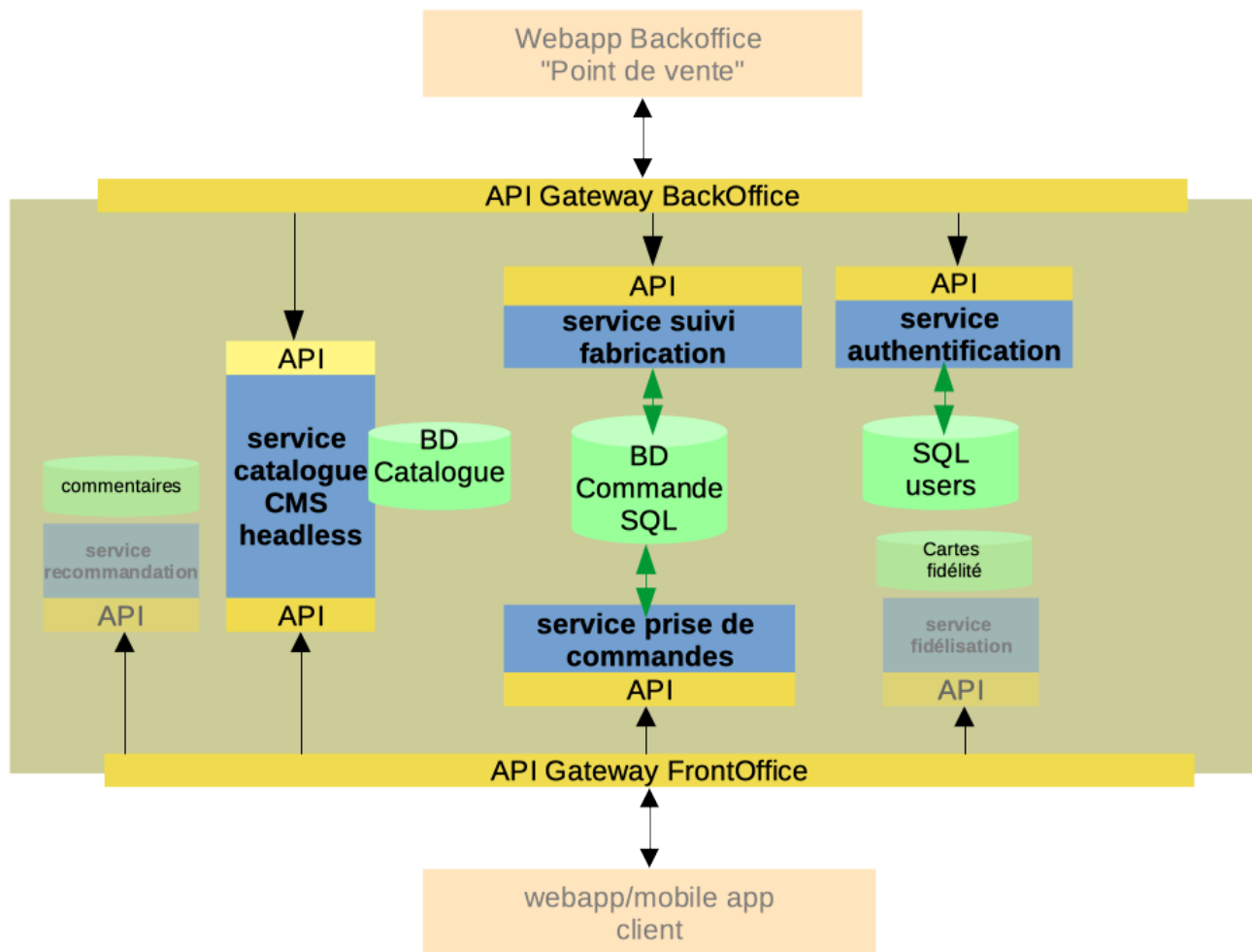
L'objectif du projet est de développer la partie **backend** de ces applications, sous la forme d'un ensemble de services web. **Ces différents services sont réalisés sous la forme d'applications indépendantes, utilisant chacune sa propre base de code.** L'application est structurée en différents services indépendants, chaque service exposant ses fonctionnalités au travers d'une API RestFul retournant des données en json :

1. un **service de prise de commande**, permettant de créer une commande, la payer et suivre son avancement. Ce service utilise une base de données stockant les commandes et leur état.
2. un **service suivi de fabrication** pour la gestion de la fabrication des commandes par le point de vente. Ce service expose une API qui permet de lister les commandes à fabriquer, de prendre en charge une commande et de modifier son état lorsqu'elle est prête ou livrée. Ce service exploite la même base de données que le service de prise de commandes,
3. un **service catalogue**, permettant de parcourir le catalogue des sandwiches proposés par LeBonSandwich : catégories, description, ingrédients, tarifs des sandwiches. Ce service utilise sa propre base de données pour stocker les produits du catalogue (catégories, sandwiches, tarifs).
4. un service **d'authentification**, qui permet d'authentifier les accès en particulier pour l'application de backoffice,
5. un service **fidélisation** permettant aux clients de cumuler les montants de leur commandes successives pour obtenir une réduction.
6. un service de **recommandation**, permettant aux clients de déposer des avis sur les produits qu'ils ont commandés,

Le backend "Le Bon Sandwich" expose deux API publiques :

- une API publique frontoffice utilisée par les applications clientes destinées aux clients du point de vente,
- une API publique backoffice utilisée par les applications de backoffice à destination du staff du point de vente.

Ces deux API sont des passerelles vers les API des services internes du backend. Elles ont pour rôle d'offrir un point d'entrée commun et unifié pour accéder aux services backend.



Objectifs du projet

Le résultat attendu consiste en un sous-ensemble de ce backend : on se concentrera sur les services de prise de commande, de suivi de fabrication, de catalogue et authentification.

L'objectif du projet est de :

- déployer l'architecture backend pour les applications LeBonSandwich,
- concevoir les api RESTFul pour l'accès à une partie des services web : prise de commande, suivi de fabrication, catalogue, authentification,
- concevoir deux api "gateway" : une gateway pour les applications de frontoffice, et une gateway pour les applications de backoffice.
- programmer une partie de ces api.

Quelques détails

Le catalogue de sandwiches :

Chaque sandwich est décrit par :

- un **nom** quelconque, un **type de pain** : blanc, complet, céréales etc ...
- une **description** listant en particulier les ingrédients choisis pour le composer,

- un **tarif** HT

Les sandwiches sont classés par **catégories** : végétarien, bio, chaud, traditionnel ... Un sandwich peut appartenir à plusieurs catégories.

Les commandes : une commande peut comporter plusieurs sandwiches. Elle est créée en précisant une date/heure souhaitée pour le retrait. Les commandes sont modifiables tant qu'elles n'ont pas été payées. En fin de fabrication, la commande est indiquée comme prête à être livrée, puis elle est livrée.

Une commande peut donc être dans les états successifs suivants :

1. **créée** : elle est enregistrée sur le backend et librement modifiable,
2. **payée** : elle n'est plus modifiable,
3. **en cours** : elle est prise en charge pour fabrication,
4. **prête** : la commande est prête, le client peut la retirer,
5. **livrée** : la commande a été retirée par le client

Le processus client : les étapes et actions réalisées par le client sont donc les suivantes :

1. parcourir le catalogue et choisir son/ses sandwiches,
2. créer la commande en précisant la date/heure de retrait, ainsi que les items de la commande ; pour chaque item commandé, on précise : l'uri du sandwich, la quantité souhaitée
3. modifier éventuellement la date/heure de livraison de la commande,
4. payer la commande,
5. suivre l'état de la commande, pour vérifier qu'elle est en cours de fabrication ou prête,
6. recevoir sa commande,
7. obtenir une facture après la livraison.

Le processus de suivi de fabrication : les étapes et actions réalisées par l'application back-office sont les suivantes :

1. lister les commandes à fabriquer : commandes payées dont la date de retrait est le jour même, classées par ordre d'arrivée,
2. prendre en charge la fabrication d'une commande : la commande change d'état,
3. finaliser la fabrication d'une commande : la commande est prête,
4. enregistrer la livraison d'une commande, la commande change d'état et peut être facturée.

Authentification

Les **services de front-office** - parcours du catalogue, prise de commande, suivi et paiement d'une commande - n'utilisent pas d'accès authentifié. La création d'une commande retourne un token unique qui permet de garantir que le client créateur d'une commande est le seul qui peut ensuite la modifier, la payer et suivre son état.

Les **services de backoffice** utilisent une authentification http et un token JWT. L'authentification est réalisée par un service dédié qui est le seul à connaître les identifiants et mots de passe des clients. Toutes les opérations réalisées au travers de l'API passerelle backoffice nécessitent de valider le token auprès du service d'authentification.

Échanges de données

Lorsque c'est nécessaire, les services échangent des données au travers des passerelles. Par exemple, lorsqu'une requête de création d'une commande est reçue par l'API frontoffice, cette API interroge le service catalogue pour obtenir le libellé et le tarif des items commandés, puis crée la commande auprès du service de prise de commande en utilisant ces données.

Description des fonctionnalités

Les fonctionnalités de l'application sont décrites sous la forme d'une liste de "user stories" : fonctionnalités élémentaires

décrites en terme de bénéfice pour l'utilisateur et de conditions de validation.

Ces stories sont relativement indépendantes les uns des autres et peuvent être développées et testées une par une.

Une liste de stories est fournie, ainsi que des exemples de descriptions.