

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/mobile. Cette **webapp client** fonctionne sur tous types de terminaux. Cette application permet 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 permet 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.

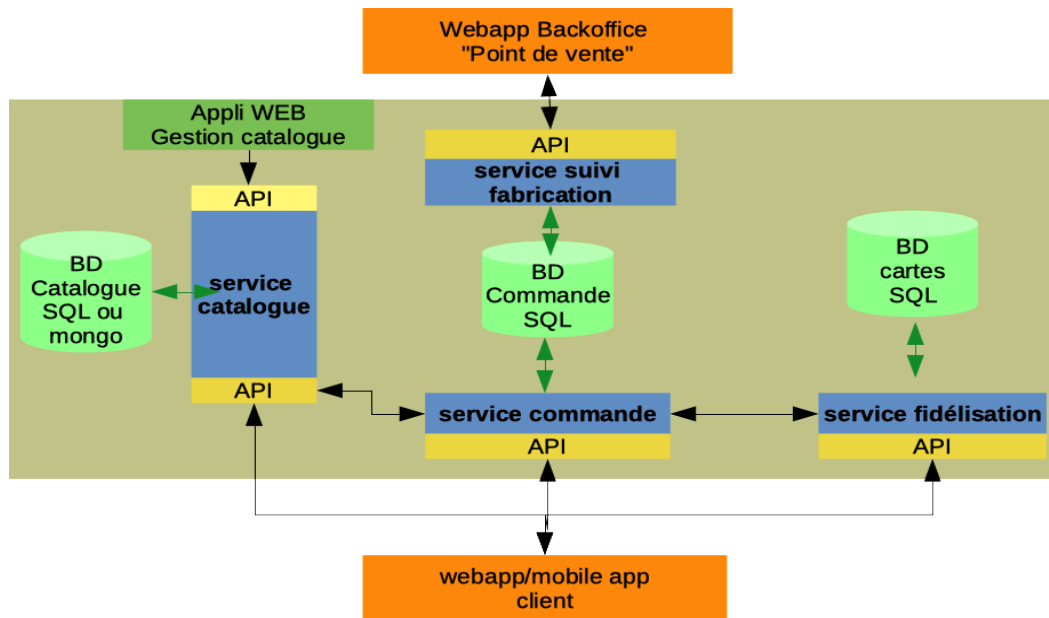
Une 3ème application web permet au point de vente de gérer le catalogue de produits et les tarifs des sandwiches proposés à la vente.

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 exposant des API et retournant des données en json. Ces différents services sont réalisés sous la forme d'applications indépendantes, utilisant chacune sa propre base de code. L'application est structurée en 4 services indépendants :

1. 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).
2. 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.
3. un service **fidélisation** permettant aux clients de cumuler les montants de leur commandes successives pour obtenir une réduction.
4. un **service suivi de fabrication** pour la gestion de la fabrication des commandes par le point de vente. Ce service expose une API RestFul privée 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.

L'application de gestion catalogue est une application web traditionnelle qui accède et manipule les données du catalogue au travers de l'API et produit des vues pour l'interface utilisateur HTML.



Objectifs du projet

L'objectif du projet est de :

- déployer l'architecture backend pour les applications LeBonSandwich,
- concevoir les api RESTful pour l'accès aux différents services web : catalogue, prise de commande, suivi de fabrication, fidélisation,
- 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, éventuellement en mode fidélisé (inscription sur sa carte de fidélité),
 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.

L'application de gestion catalogue : utilisée par le staff de gestion du point de vente, elle permet de mettre à jour le catalogue.

Authentification

Le **service catalogue** n'utilise pas de mécanisme d'identification/authentification : les clients du service peuvent l'utiliser librement pour parcourir le catalogue et accéder aux données.

Le **service de prise de commande** n'utilise pas non plus de mécanisme d'identification/authentification, mais utilise un token pour garantir que le créateur d'une commande est le seul qui peut la modifier, la payer et suivre son état.

Le **service de fidélisation** utilise une authentification http et un token JWT. Il est le seul à connaître les identifiants et mots de passe des clients. L'authentification auprès de ce service permet ensuite au client de payer en mode fidélisé.

Fidélisation

Un client peut s'enregistrer auprès du service de fidélisation. Ensuite, il peut demander à enregistrer les commandes payées sur sa carte de fidélité. Ceci se fait au moment du **paiement** d'une commande. Les montants sont cumulés, et une remise de 5% peut être appliquée lorsque 10 commandes ont été enregistrées.

Lors d'un paiement en mode fidélisé, le client indique son n° de carte de fidélité et doit fournir une preuve d'authentification auprès du service fidélisation sous la forme d'un token, qu'il a obtenu au préalable.

Le client peut également demander à bénéficier de la remise accordée aux clients fidèles.

Echanges de données

Lors de la création d'une commande, le client transmet ses coordonnées, la date et l'heure de livraison souhaitées, et la liste des URI des sandwiches commandés avec leur quantité. Le service de prise de commande utilise ces URI pour obtenir les tarifs des différents sandwiches, calcule le montant total puis enregistre cette commande.

Lors du paiement, le client transmet la référence de la commande et les informations de paiement par carte bancaire (n° carte, date d'expiration). S'il souhaite inscrire le montant de cette commande sur sa carte de fidélité, il ajoute un token obtenu auprès du service de fidélisation.

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 unes des autres et peuvent être développées et testées une par une.

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