 MyMuffin

**Dossier d’Architecture Technique**

|  |  |
| --- | --- |
| Révision | 1.0.0 |
| Effectué par | Fofana Moussa |

Architecture applicative

# Objectifs

L’objectif de ce projet est de proposé une solution permettant la création de commande de muffins en ligne, du choix des muffins et de leurs ingrédients au paiement, ainsi que l’envoi au service de livraison local à la ville, le tout se faisant le jour même afin de garder des produits frais et succulent.

Il sera par la suite possible de proposer a des particuliers de créer leur « Marketplace » permettant de mettre en vente leurs propres muffins.

# Acteurs

## Externes

|  |  |  |
| --- | --- | --- |
| Acteur | Description | Localisation |
| Uber Serve | Service proposant des coursiers afin d’effectuer les livraisons | Méru, Lyon 03 |
| Stripe | Permet le règlement en ligne de la commande | Monde |

## Internes

|  |  |  |
| --- | --- | --- |
| Acteur | Description | Localisation |
| MyMuffins Fabric | Ateliers de conceptions des muffins | Esches, Lyon 05 |
| MyMuffins Devs | Équipe de developeur interne | Lyon 03 |

# Contraintes

## Budgétaire

Pas de contraintes budgétaires au moment de l’écriture mais limiter tant que possible les dépenses.

## Planning

Alpha de l’API REST fin janvier 2023, beta fin Mars 2023

Alpha Web view mi-mars 2023 beta fin mai 2023

Alpha application mobile fin juillet beta début novembre 2023

## Urbanisation

Il doit être impossible pour un client en dehors de la zone desservie par notre partenaire Uber Serve sous peine de frais supplémentaire à notre charge.

Une commande ne pourra être modifié que pendant les 5 premières minutes afin de garantir nos délais et limiter le gâchis des matières première.

Chaque MyMuffin factory doit préciser le nombre de commande simultané qu’elle peut gérer, de même pour les particuliers qui souhaiterais ouvrir une marketplace pour éviter leur surcharge et l’attente des clients.

# Exigences

L’API REST doit pourvoir changer de service de paiement avec le minimum de casse possible.

Prévoir en amont l’évolution concernant la Marketplace de particulier dans le design de l’API REST.

Il doit être possible de pouvoir communiqué avec plusieurs services de livraison a la fois afin d’étendre facilement notre couverture des villes françaises.

# Architecture cible

Notre application sera composée de 3 grandes parties :

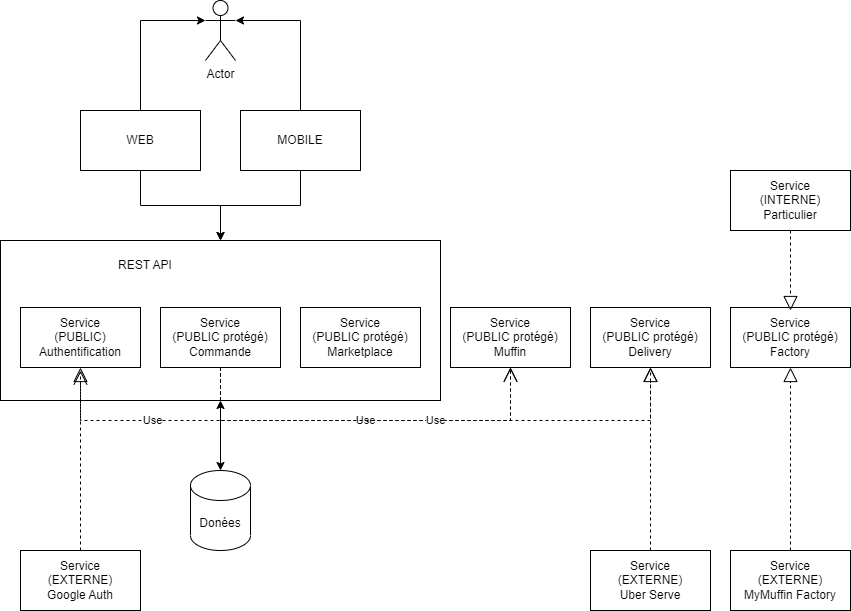
* L’API REST en micro-services
* La Web View
* La Mobile View

L’API REST est au centre du projet et doit être robuste elle permettra la communication entre les différents acteurs du projet, à des fins de scalabilité elle sera découpée en micro service :

* Service d’authentification (public)
* Service de commande (public protégé par l’authentification)
* Service de récupération des différentes marketplaces et de leurs produits (public)
* Service de modification du muffin (public protégé par l’authentification)
* Service d’envoi des commandes au coursiers (interne)
* Service d’envoi des commandes a une factory (interne)

### Schéma

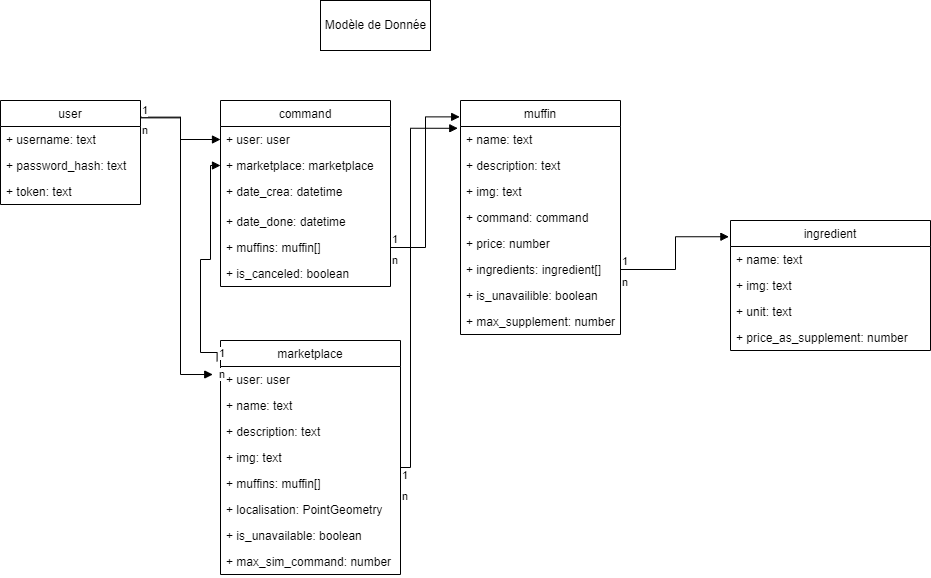
// TODO ADD COLORS HERE AND IN TEXT



L’api permettras à un client de s’authentifier via le service d’authentification puis de choisir une boutique via le service de récupération des différentes marketplaces une fois dans une boutique le service de commande prend le relais, celui-ci communiquera avec le service factory afin de savoir si on peut encore y passer commande et avec les services de livraison pour savoir si un coursier est disponible. Une fois toute ses condition réunie la commande est passé.

# Architecture en détail

## Modèle de données



## Data Access Objects

