



Livrables OC PIZZA

Sommaire :

1. Objectif :
2. Personna :
3. Spécifications Fonctionnelles
 - 3.1 Acteurs **principaux**
 Acteurs **secondaires**
4. Diagramme de **package**
 - 4.1 Cas d'utilisations **des stocks**
 Diagramme d'activités **des stocks** :
 Fiches descriptives des stocks :
 - 4.2 Cas d'utilisations **Gestion des livraisons**
 Diagramme d'activités **des livraisons** :
 Fiches descriptives **Livraisons** :
 - 4.3 Cas d'utilisations **gestion des commandes**
 Diagramme d'activités **commande**
 Fiches descriptives **des commandes** :
 - 4.4 Cas d'utilisations **Administration**
 Diagramme d'activité **Administration**
 Fiches descriptives **Administration** :
5. Cycle de vie d'une commande
6. Solutions techniques
 - ? 5.2 Pourquoi privilégions-nous le natif ?

1. Objectif :

L'objectif est de développer un système cohérent pour tout les magasins, utilisable par l'ensemble des acteurs, de la commande des pizzas jusqu'à la livraison, avec un tableau de bord pour la direction permettant d'avoir accès à toute les données et modifier l'ensemble du système.

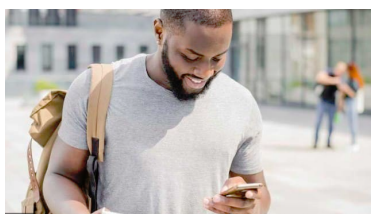
Nous proposons le développement d'un site internet et d'une application mobile, afin de rendre l'expérience utilisateur optimale.

Pour rappel, le cahier des charges défini est le suivant :

- être plus efficace dans la gestion des commandes ,de leur réception à leur livraison en passant par leur préparation
- suivre en temps réel les commandes passées en préparation et en livraison
- suivre en temps réelle stock d'ingrédients restants pour savoir quelles pizzas peuvent encore être réalisées
- proposer un site Internet pour que les clients puissent
- passer leurs commandes en plus de la prise de commande par téléphone ou sur place
- payer en ligne leur commande s'ils le souhaitent –sinon,ils paieront directement à la livraison ; modifier ou annuler leur commande tant que celle-ci n'a pas été préparée.
- proposer un aide-mémoire aux pizzaiolos indiquant la recette de chaque pizza

2. Personna :

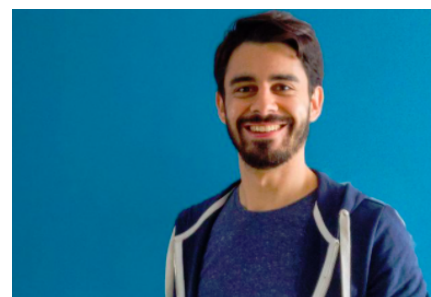
Notre cible à l'habitude et le besoin d'avoir un système facile, ergonomique et efficace. Elle choisira le plus souvent de commander par internet et se faire livrer, sur ordinateur ou smartphone. Sa motivation principal est de pouvoir manger sur le pouce sans devoir faire d'effort et deviendra vite accroc à un service rapide proposant un produit correct en quantité suffisante.



David : Etudiant à la fac, il retravaille ses cours le soir dans un petit studio. il va privilégier les repas rapides pour gagner du temps et sera facilement tenté par la livraison de plat à emporter.



Julien : Alternant en vente, il habite avec sa copine. Une fois par semaine, ils se calent confortablement devant une série Netflix en mangeant un plat à emporter.



Mathieu : Programmeur dans une startup, Mathieu a l'habitude avec son équipe de commander des plats à emporter le midi directement à leur entreprise.

Cible Principale

Aa Caractéristiques Socio-démographiques	≡ Caractéristiques comportementales
<u>Origines : Grande et moyenne ville</u>	Archétype : Etudiant
<u>âge : Entre 20 et 35 ans</u>	Motivation : Plaisir
<u>Profession : Etudiant, salarié</u>	Critère de sélection : Rapidité et confort
<u>Salaire : Entre 18 473,00 et 30 000€/an</u>	

3. Spécifications Fonctionnelles

3.1 Acteurs principaux

Tableau

≡ Les fonctionnalités	Aa pizzaiolos	≡ Livreurs	≡ Client	≡ Vendeur sur place	≡ Responsable
Reception des commandes	<u>X</u>				
Consulter les recettes(mémo)	<u>X</u>				X
Préparation des commandes	<u>X</u>				
Suivi en temps réel des stocks	<u>X</u>				
Suivi en temps réel de l'état des commandes	<u>X</u>	X	X	X	
Changer statut commande	<u>X</u>	x			
Passer une commande en ligne	<u>Untitled</u>		X		
Choisir parmi les pizza en stock	<u>Untitled</u>		X		
Payer en ligne ou sur place	<u>Untitled</u>		X		
Modifier la commande non préparé	<u>Untitled</u>		X		
Supprimer ou ajouter Pizza	<u>Untitled</u>				X
Changer prix des pizzas	<u>Untitled</u>				X
Recevoir un paiement	<u>Untitled</u>			X	

☰ Les fonctionnalités	Aa pizzaiolos	☰ Livres	☰ Client	☰ Vendeur sur place	☰ Responsable
	<u>Untitled</u>				

Acteurs **secondaires**

- System bancaire

4. Diagramme de **package**

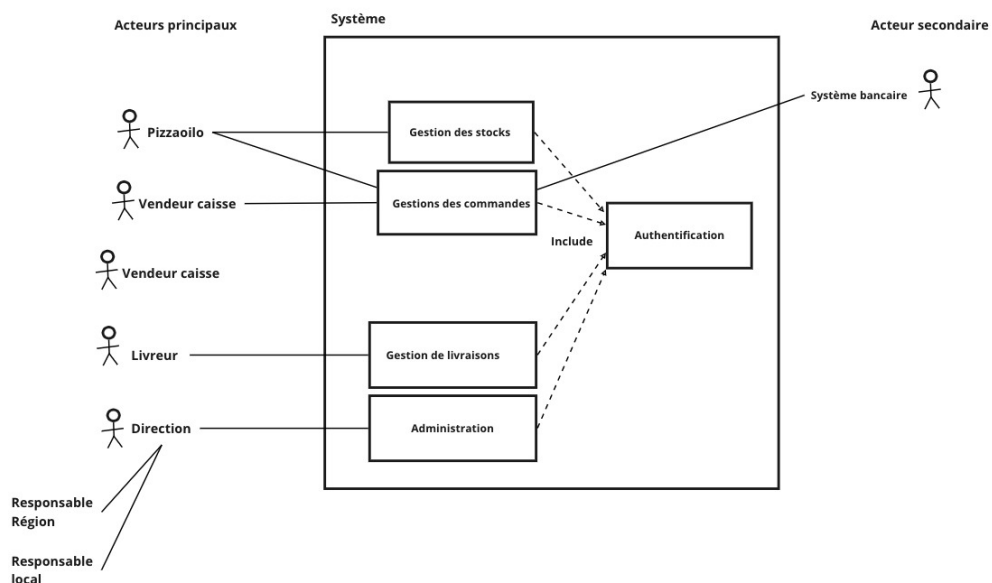
Le système est partagé en différent package toutes relié à l'authentification via un compte personnel : L'acteur Pizzaiolo peut consulter les stocks et les modifier, il a également la possibilité de voir l'état des commandes et d'indiquer lorsqu'une commande est prête.

Le client peut uniquement voir l'état de ces commandes en cours

La vendeur à la caisse peut également voir les commandes et recevoir un paiement

Le livreur peut gérer la livraison

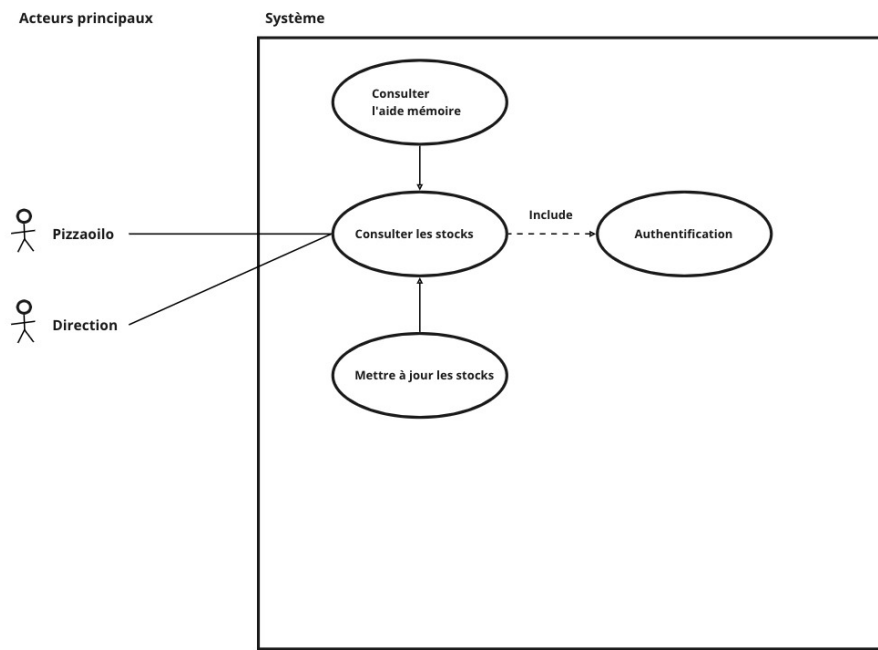
Et enfin, les responsables ont un tableau de bord complet qui englobe toute les informations du système avec des fonctions avancés d'édition pour chacun des packages.



miro

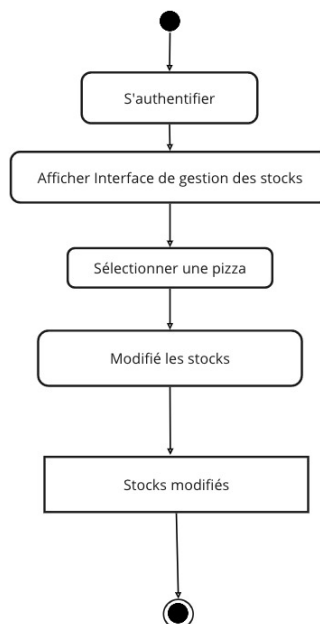
4.1 📦 Cas d'utilisations des stocks

Le Pizzaiolo peut consulter et mettre à jour les stocks, il a également accès à un aide mémoire pour la préparation de sa pizza. Cela inclut la création d'un compte employé, gérer par les responsables (voir Administration des données)



miro

🔄 📦 Diagramme d'activités des stocks :



miro

▼ Cas n°

▲ Nom :

👤 Acteur(s) :

✍️ Description :

👤 Auteur :

🕒 Date :

⚙️ Pré-conditions :

🚀 Démarrage :

👉 Le scénario nominal :

✚ Les scénarios alternatifs :

✓ Fin et post-conditions :

✗ Fin :

⚙️ Post-conditions :

+ COMPLEMENTS

 Ergonomie :

 Performance attendue :

 Problèmes non résolus :

 Fiches descriptives des stocks :

▼ Cas n°1

▲ Nom : Consulter les stocks disponible et les mettre à jour

 Acteur(s) : Pizzaiolo, Direction

 Description : Les acteurs peuvent consulter les stocks des ingrédients pour les pizzas, ils peuvent les mettre à jours.

 Auteur : Maxime

 Date :

 Pré-conditions : Les acteurs doivent être authentifié

(Cas d'utilisation « S'authentifier » – package « Authentification »)

 Démarrage : Le système démarre lorsque les acteurs sont authentifié

Le scénario nominal :

1. *L'utilisateur clique sur la gestion des stocks*
2. Le système affiche la page de gestion des stocks avec les différentes pizzas en vente (carrousel d'image avec titre)
3. *L'utilisateur décide de cliquer sur une des pizzas*
4. Le système affiche une page de la pizza avec les différents ingrédients et leur quantité
5. L'utilisateur peut modifier les stocks pour les mettre à jour
6. Le système affiche une page des stocks disponible pour cette pizza

Les scénarios alternatifs :

- 2.a : l'utilisateur décide de quitter la gestion des stocks
- 4.a. L'utilisateur décide de quitter la page de la pizza et revenir à la gestion des stocks principal

Fin et post-conditions :

 **Fin : 2,4,6 sur décisions de l'utilisateur**

Post-conditions :

- Les stocks ont été mise à jour

COMPLEMENTS

Ergonomie :

Sur le tableau principale des pizza actuellement en vente, elles seront sous forme de carrousel d'image avec leur nom bien visible. Si un ingrédient d'une des pizzas est en faible quantité , une alerte s'affiche par une pastille rouge pour signaler qu'il va bientôt manquer d'un ingrédient pour la création de cette pizza.

Performance attendue :

l'affichage doit se faire en 1 secondes maximum, peu importe la page.
Pareil pour la mise à jour des stocks qui doit s'actualisé en moins d'une seconde.

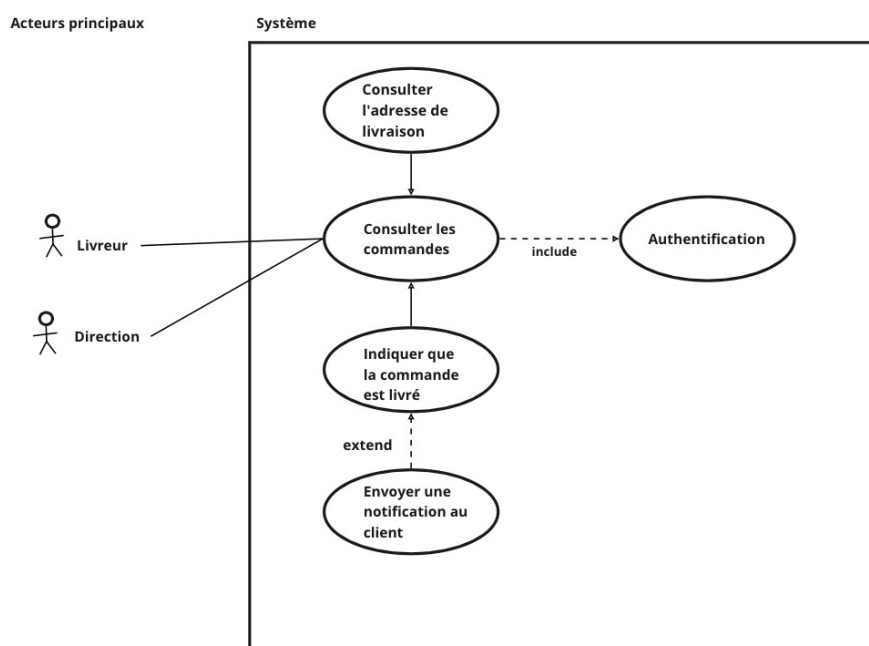
Problèmes non résolus :

- Créer une alerte automatique qui envoi à la personne concerné sur son mail pro ou par sms qu'il manque d'un ingrédient ?
- Quelles acteurs s'occupe de l'achat et l'approvisionnement des stocks ?

4.2 Cas d'utilisations Gestion des livraisons

Le livreur peut consulter les commandes en cours et voir les détails de chaque client avec son adresse de livraison.

Une fois livré, il peut validé la commande via l'application ce qui envoi automatiquement une notification au client.



miro

Diagramme d'activités **des livraisons** :




miro

Fiches descriptives **Livraisons** :

▼ Cas n°1

▲ **Nom** : Consulter les détails des commandes et les mettre à jour

 **Acteur(s)** : Livreur, responsable

 **Description** : Les acteurs peuvent consulter les commandes en cours et les mettre à jour (lorsque la commande a été livré)

 **Auteur** : Maxime

 **Date** :

 **Pré-conditions** : Les acteurs doit être authentifiées en tant que client.

 **Démarrage : Le système démarre lorsque les acteurs sont authentifiés.**

Le scénario nominal :

1. L'acteur demande l'interface des commandes
2. Le système affiche toutes les commandes en cours
3. L'acteur clique sur une commande
4. Le système affiche des détails supplémentaires sur la commande via une nouvelle page, avec les coordonnées du client
5. L'acteur change le statut d'une commande comme étant livré

Les scénarios alternatifs :

- 2.a Le client décide de se déconnecter
- 4.a Le client décide de quitter les détails de la commande et revenir sur l'étape 2

Fin et post-conditions :

 **Fin : 2, 4, 5 sur décision de l'utilisateur**

Post-conditions :

- Les commandes sont mises à jour

+ COMPLEMENTS

Ergonomie :

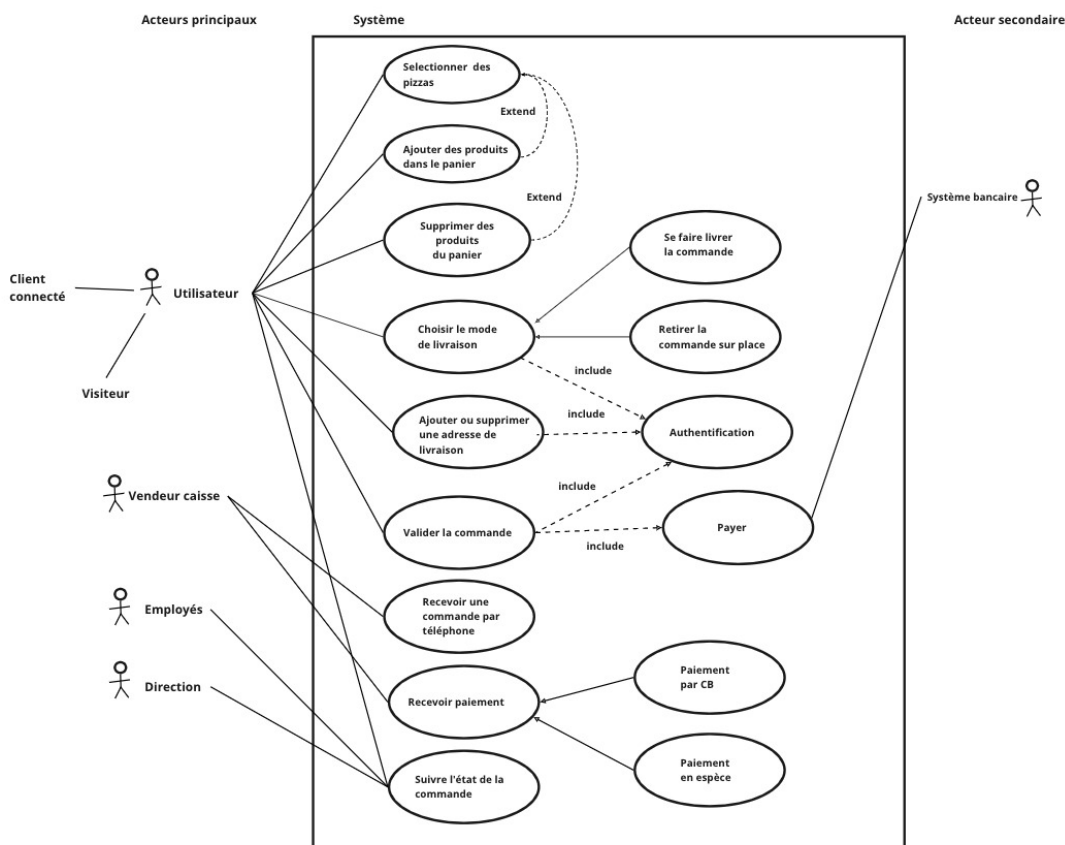
La modification de l'état d'une commande doit se faire facilement. Il faut donc qu'un bouton nommé "livré" soit directement visible, sur chaque commande dans la page globale, mais aussi pour les pages détaillées de chaque commande, en oubliant pas de demander une confirmation en cas d'un changement par erreur.

Performance attendue :

Affichage des commandes en moins d'une seconde. Pareil lors de l'affichage de la page détaillée d'une commande.

Problèmes non résolus :

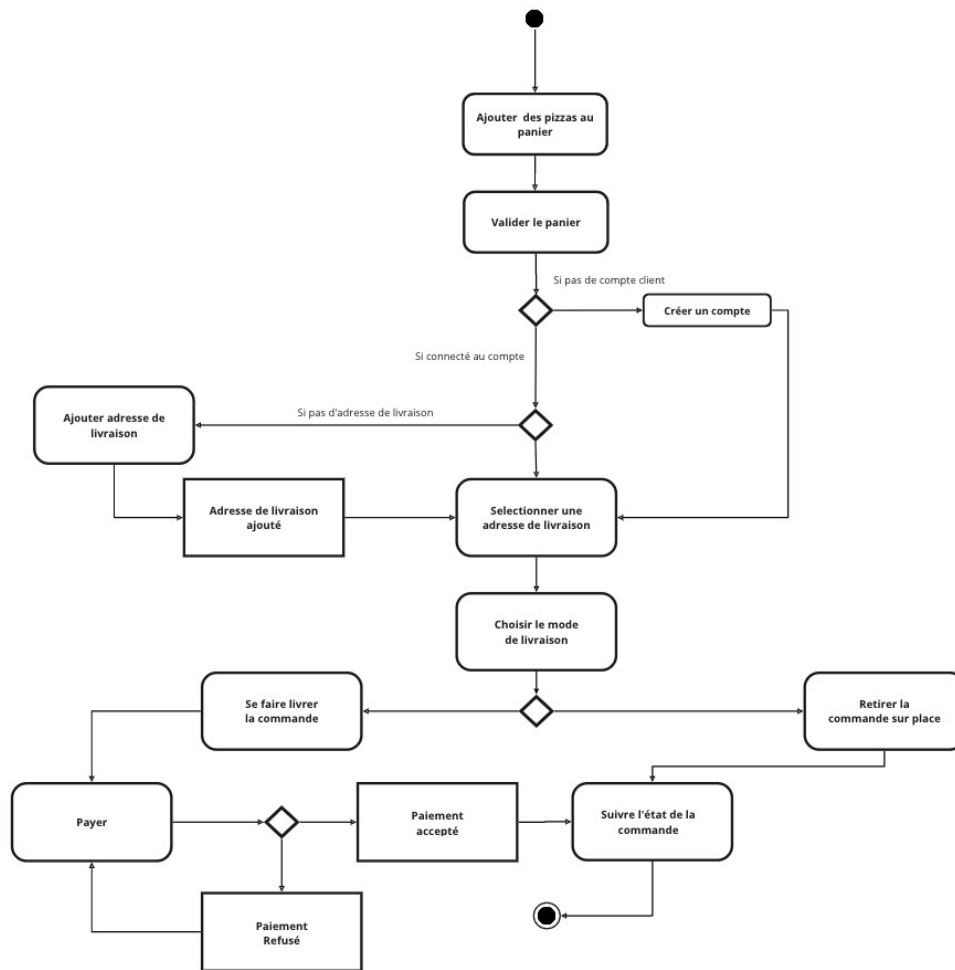
4.3 Cas d'utilisations **gestion des commandes**



miro



Diagramme d'activités commande



miro



Fiches descriptives des commandes :

▼ Cas n° 1

▲ **Nom : Choisir parmi les pizzas disponibles et commander**



Acteur(s) : Client



Description : Le client doit pouvoir choisir parmi un panel de Pizza disponible, ajouter ou soustraire une pizza dans son panier et choisir son mode de livraison et de paiement.



Auteur : Maxime



Date :



Pré-conditions : L'acteur doit être authentifié en tant que client.

(Cas d'utilisation « S'authentifier » – package « Authentification »)



Démarrage : L'utilisateur a demandé la page "Nos Pizzas"



Le scénario nominal :

1. **Le système affiche une page avec les différentes Pizzas (nom et photos)**
2. **Le système agrandit la photo et affiche une description courte des différentes pizza lorsqu'on passe la souris dessus**
3. **Le système affiche un icône d'ajout rapide dans le panier de la pizza**
4. *L'utilisateur peut choisir parmi les choix disponible*
5. **Le système affiche sur une page complète des photos et une description détaillé lorsqu'elle est choisi (clic) avec un bouton d'ajout dans le panier.**
6. *L'utilisateur peut consulter son panier*
7. **Le système affiche le panier**
8. *l'utilisateur peut valider sa commande*
9. **Le système affiche une page de validation avec le choix livraison et de paiement**

✦ Les scénarios alternatifs :

3.b ,L'acteur décide d'ajouter une pizza dans son panier. Cela ne lui fait pas quitter la page. Le panier s'anime et change pour signaler qu'il contient

une pizza. Il y a un chiffre dessus qui varie en fonction du nombre de pizza ajouté.

4.a : L'acteur décide de quitter la page de la pizza choisi.

4.b : L'acteur décide d'ajouter la pizza dans panier, une pop up s'affiche pour lui demander s'il souhaite continuer à choisir des pizzas (**retour à l'étape 1**) ou **consulter son panier**.

7.a : L'acteur peut soustraire une pizza de son panier

9.a : L'acteur L'acteur L'acteur décide de quitter la page de validation de commande.

✓ Fin et post-conditions :

✗ **Fin : Aux étapes 4,6,8,9 sur décision de l'utilisateur**

⚙️ Post-conditions :

Le système affiche le récapitulatif de la commande

Le système retourne à l'affichage général (retour à l'étape 1)

+ COMPLEMENTS

🛋️ Ergonomie :

Le bouton d'ajout d'une pizza doit être bien visible pour faciliter le parcours utilisateur. Une fois qu'il a pris sa décision, il n'a pas à réfléchir et sait automatiquement comment ajouter sa pizza, ce qui fait gagner du temps à l'utilisateur.

Les animations pour chaque actions doivent être soigné.

1. Lors de l'ajout de la pizza dans le panier
2. Lors du retrait d'une pizza dans le panier

3. Transition agréable entre les différentes pages
4. Les images parlent plus que les mots et doivent être bien mise en avant
5. Une fois le client sur la page de validation, le bouton de validation doit être l'objet le plus visible. Avec éventuellement un carrousel des photos des pizza qu'ils s'apprête à valider.

Performance attendue :

L'affichage des Pizzas doit être rapide, moins de 2 secondes. De même que les photos aux passages de la souris ou en description détaillé une fois sélectionné.


Problèmes non résolus :

- Que se passe t-il s'il n'y a pas la pizza en stock ?

▼ **Cas n°2**

▲ **Nom : Consulter l'interface des suivis de commande**

 **Acteur(s) : Vendeur Caisse, Responsable**

 **Description : Le vendeur ou le Responsable peut consulter l'interface de suivi pour voir l'état des commandes des clients. Il peut également recevoir un paiement quand le client paye en espèces.**

 **Auteur : Maxime**

 **Date :**

 **Pré-conditions : L'utilisateur doit être authentifié en tant que vendeur ou responsable sur le site interne**

(Cas d'utilisation « S'authentifier » – package « Authentification »)

Démarrage : L'utilisateur à demander la page suivi des commandes

Le scénario nominal :

1. L'acteur *peut* suivre l'état des commandes d'un client
2. Le système affiche l'état de la commande des clients en cours
3. L'acteur peut cliquer sur un client en particulier
4. Le système affiche le client en question avec plus d'information

✚ Les scénarios alternatifs :

- 2.a L'acteur peut quitter la page d'état des commandes clients

✓ Fin et post-conditions :

✗ Fin : étapes 5 ou 2 sur décision de l'utilisateur

⚙ Post-conditions :

Le système retourne à l'état des commandes général (retour à l'étape 1)

+ COMPLEMENTS

Ergonomie :

Le graphisme doit être clair et épurée, les informations essentiel dois sauter aux yeux.

 Performance attendue :

L'affichage des commandes en cours doit se faire en une seconde maximum.

🤔 Problèmes non résolus :

- En cas de boisson manquante, Faut-il alerter le client d'un changement sur la commande en cas de rupture de stock ?

4.4 Cas d'utilisations Administration

Les responsables peuvent gérer tout le système. Ils ont à la fois une vision globale des informations mais aussi un pouvoir d'édition sur le système.

Lorsqu'un responsable créer un compte, il génère un code provisoire unique pour permettre à l'employé de créer et activer son compte.

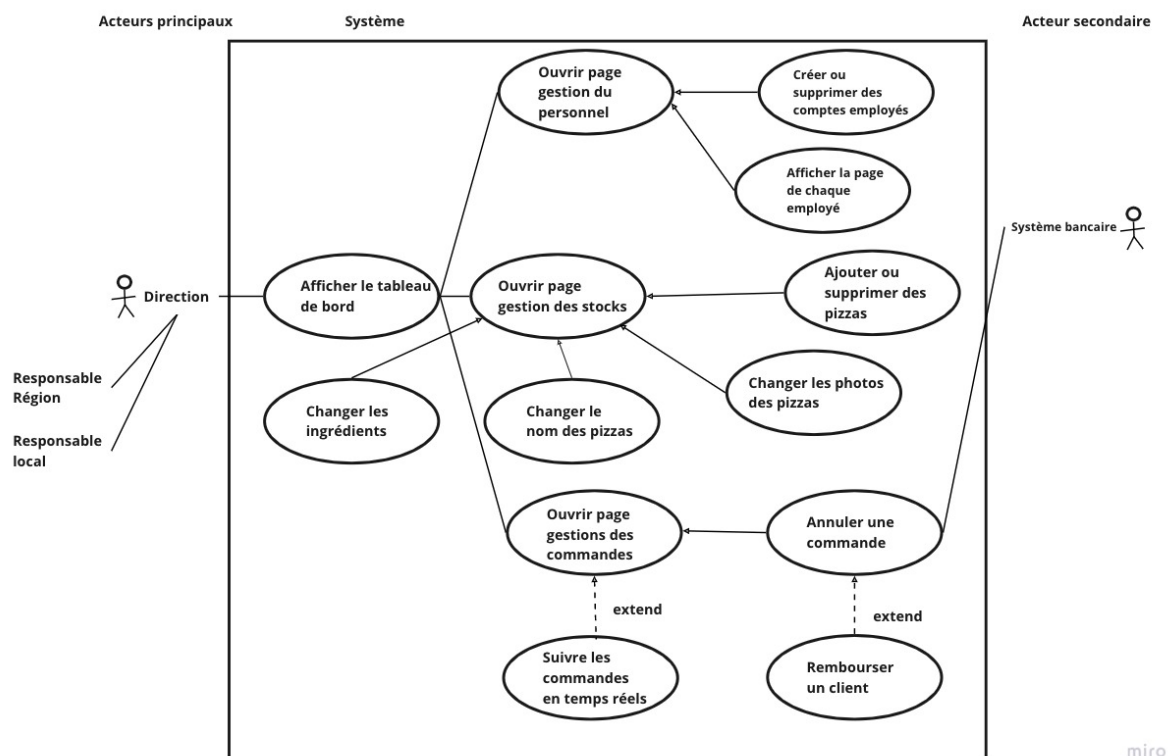
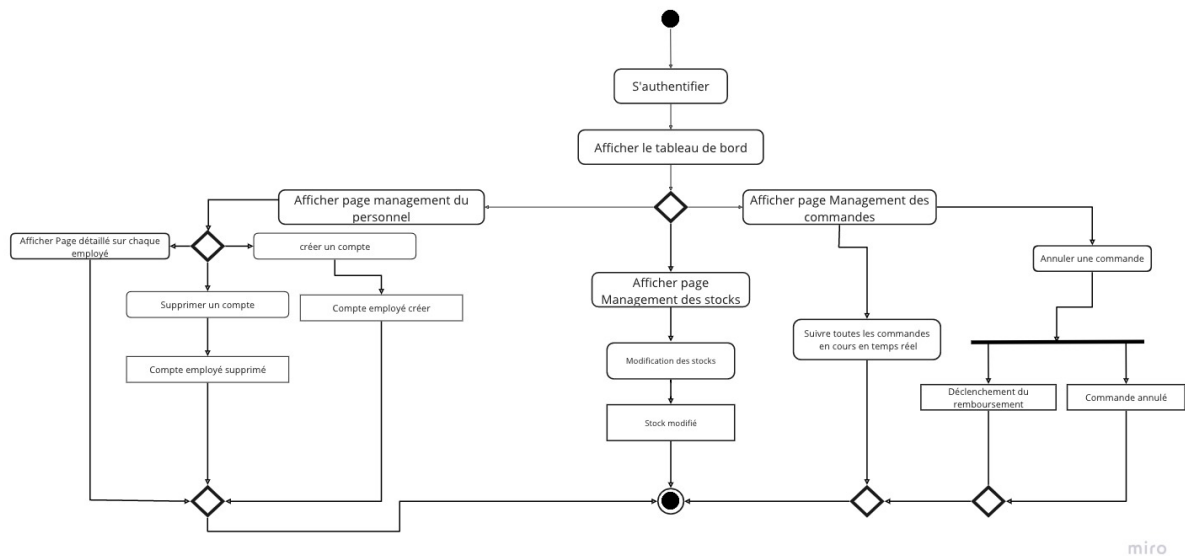


Diagramme d'activité **Administration**



Fiches descriptives **Administration** :

▼ Cas n°1

▲ **Nom** : consulter le tableau de bord avec toute les données des ou du magasin(s)

 **Acteur(s)** : Direction, responsable

 **Description** : La direction a accès à un tableau de bord pour manager toutes les pizzérias et en gérer chaque aspect, tant matériel qu'humain.

 **Auteur** : Maxime

 **Date** :

 **Pré-conditions** : L'utilisateur doit être authentifié

(Cas d'utilisation « S'authentifier » – package « Authentification »)

Démarrage : **Le système démarre lorsque l'utilisateur s'authentifie à son compte**

Le scénario nominal :

1. L'acteur *est authentifié et connecté.*
2. Le système affiche le tableau de bord
3. L'acteur *choisi le management des commandes*
4. Le système affiche le management des commandes
5. L'acteur *choisi le management des stocks*
6. Le système affiche le management des stocks
7. L'acteur *choisi le management du personnel*
8. Le système affiche le management du personnel

Les scénarios alternatifs :

2a. *L'utilisateur décide de quitter le tableau de bord et se déconnecter*

4.a, 6a, 8a, *L'utilisateur décide de quitter la page et revenir au tableau de bord (2)*

Fin et post-conditions :

 **Fin : 2,4,6,8 sur décision de l'utilisateur**

Post-conditions :

•

+ COMPLEMENTS

Ergonomie :

Les informations essentiels doivent être bien visible sur le tableau de bord général, sans éléments superflues.

De même que pour chaque rubrique.

Un Graphisme minimaliste, et une grosseur des différents éléments visuelles (Bouton fonctionnelles et informations pures) proportionnelle à leur niveau d'importance et d'utilité.

Performance attendue :

- Le tableau de bord doit se charger en moins de 5 secondes

Problèmes non résolus :

- Quelles informations essentielles affichées sur le tableau de bord ?

▼ **Cas n°2**

▲ **Nom : L'utilisateur peut manager les commandes de la pizzeria**


 **Acteur(s) : Direction, responsable**

 **Description : La direction ou le responsable peuvent voir les commandes en cours et les gérer en annulant ou modifiant une commande**

 **Auteur : Maxime**

 **Date :**

 **Pré-conditions :** L'utilisateur doit être authentifié et doit cliquer sur la rubrique "Administration des commandes"

 **Démarrage :** Le système démarre lorsque l'utilisateur clique sur la rubrique "Administration des commandes"

Le scénario nominal :

1. L'acteur *clique sur la rubrique administration des commandes*
2. Le système affiche la page avec toutes les commandes en cours classées par restaurant
3. L'acteur *décide de modifier une commande*
4. L'acteur *décide d'annuler une commande et procéder à un remboursement*
5. L'acteur *décide de retourner au menu principal (1)*

Les scénarios alternatifs :

- 2.a L'acteur *décide de quitter l'administration des commandes*

Fin et post-conditions :

 **Fin :** 2.a et 6 sur décision de l'utilisateur

 **Post-conditions :**

- le système affiche le tableau de bord
- Le système retourne à l'étape 1

+ COMPLEMENTS

Ergonomie :

Similaire au cas n°1 :

Les informations essentiels doivent être bien visible sur le tableau de bord général, sans éléments superflues.

De même que pour chaque rubrique.

Un Graphisme minimaliste, et une grosseur des différents éléments visuelles (Bouton fonctionnelles et informations pures) proportionnelle à leur niveau d'importance et d'utilité.

Performance attendue :


voir cas n°1

Problèmes non résolus :

▼ Cas n°3

▲ **Nom : Les acteurs peuvent gérer le management des stocks**

 **Acteur(s) : Direction, responsable**

 **Description : Les acteurs ont accès aux stocks avec les différentes pizza en vente, avec la possibilité de supprimer ou ajouter une pizza, modifier les photos et modifier les ingrédients.**

 **Auteur :**

 **Date :**

 **Pré-conditions :** Les acteurs doivent être authentifié

 **Démarrage :** Le système démarre lorsque les acteurs demandent à afficher la page "management des stocks" dans le tableau de bord.

 **Le scénario nominal :**

1. L'acteur clique sur la rubrique management des stocks
2. Le système affiche la page avec les pizzas en stock
3. L'acteur clique sur le bouton pour modifier une pizza
4. le système affiche une nouvelle page qui permet de modifier la pizza.

✦ Les scénarios alternatifs :

- 2.a l'acteur décide de cliquer sur le bouton d'édition et de supprimer ou ajouter des pizzas
- 4.a l'acteur décide de modifier la photo de la pizza
- 4.b l'acteur décide de modifier le nom de la pizza
- 4.c l'acteur décide de modifier les ingrédients de la pizza
- 4.d l'acteur décide de modifier les stocks d'un ingrédient de la pizza

✓ Fin et post-conditions :

✗ Fin : 2 et 4 sur décision de l'acteur

Post-conditions :

- La photo d'une pizza a été changé
- le nom de la pizza a été modifié
- les ingrédients de la pizza ont été modifié
- Les stocks des ingrédients d'une pizza a été modifié

+ COMPLEMENTS

Ergonomie :

L'édition des pizzas doit être intuitif.

Carrousselle d'image des pizzas à modifier,
modification rapide avec boutons bien en évidence

Performance attendue :


L'affichage et le chargement doit se faire en moins de 2 secondes


Problèmes non résolus :

- Doit-on envoyer une alerte aux pizzaiolo lorsqu'il y a un changement dans les stocks ?

▼ Cas n°4

▲ Nom : **L'acteur peut manager le personnel**


 Acteur(s) : **Responsable**

 Description : **Les responsables peuvent créer ou supprimer un compte employé et consulter une fiche détaillé de ce dernier.**

 Auteur :

 **Date :**

 **Pré-conditions :** **L'acteur doit être authentifié.**

 **Démarrage :** **Le système démarre lorsque l'acteur clique sur le bouton "management du personnel"**

 **Le scénario nominal :**

1. *L'acteur clique sur la rubrique management du personnel*
2. *Le système affiche la page avec les employés du magasin*
3. *L'acteur clique sur le bouton pour modifier une fiche employé*
4. *le système affiche une nouvelle page qui permet de modifier la fiche employé*

+ Les scénarios alternatifs :

- 3.a *l'acteur décide de supprimer une fiche employé*
- 2.a *l'acteur choisi de créer une nouvelle fiche*

✓ Fin et post-conditions :

✗ Fin : 2 et 4 sur décision de l'utilisateur

 **Post-conditions :**

la fiche employé à été créer ou modifié

Un login et un mot de passe a été généré.

+ COMPLEMENTS



Ergonomie :

Comme pour l'ensemble du système, un design épuré, les fonctionnalités importantes plus grande que le reste, des icônes minimaliste pour augmenter la visibilité.



Performance attendue :

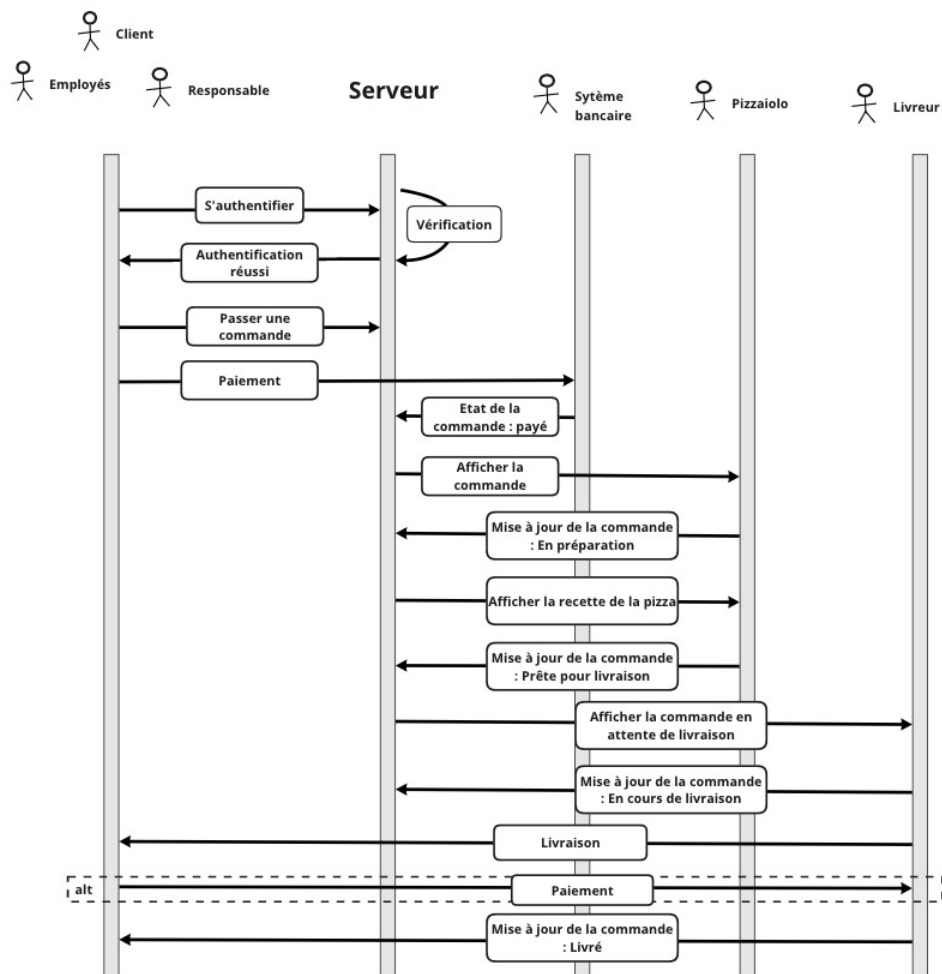
affichage en moins de 3 secondes



Problèmes non résolus :



5. Cycle de vie d'une commande



miro

⚙️ 6. Solutions techniques



- Un site web avec le CMS Magento



Stripe est une infrastructure de paiement web sécurisée (certifiée PCI niveau 1, le

Magento, est en vogue pour la puissance de son système d'administration et sa capacité de personnalisation, il est gratuit et Opensource avec une version payante. Par ailleurs il offre un grand panel de fonctionnalités « natives » ce qui offre un vrai plus au logiciel brut.

Magento dispose d'une Gestion de catalogue très développée : la CMS s'occupe de la gestion des catalogues de ses clients et ce, en contrôlant les stocks, les quantités et les prix.



- **Utilisation de la base de donnée MySQL.** C'est l'option a plus répandu.
- c'est une base de données relationnelle qui permet une meilleur gestion.

niveau de sécurité le plus élevé) et très efficace. Elle permet à vos clients de payer simplement en utilisant leur numéro de CB sans quitter votre app web.

- facile à mettre en place
- Solution adapté à l'e-commerce



- **La création d'une application mobile en natif swift pour iOS et un développement Android en Java**

? 5.2 Pourquoi privilégions-nous le natif ?

OC pizza possède un nombre important de magasin est une entreprise à forte croissance. Par ailleurs, votre cible requière une application **intuitive et parfaitement adapté**. Le développement natif semble requis pour garantir une expérience utilisateur optimale. Par ailleurs, l'entretien sera facilité, car l'hybride peut vite devenir plus couteux à entretenir sur le long terme et il sera toujours en retard par rapport au natif sur les nouvelles fonctionnalités.