|  |
| --- |
| **OC Pizza**  **Projet 8**  Dossier de conception technique  Version 1.0 |
| **Auteur**  Mickaël HORN  *Développeur d'applications iOS junior* |

Table des matières

1 - Versions 3

2 - Introduction 4

2.1 - Objet du document 4

2.2 - Références 4

2.3 - Besoins du client 5

2.3.1 - Rappel du contexte 5

2.3.2 - Rappel des enjeux et objectifs 5

3 - Domaine Fonctionnel 6

3.1 - Modèle Conceptuel de Données (Diagramme de classes) 6

3.1.1 - Employee et Restaurant 7

3.1.2 - Héritage Employee 7

3.1.3 - Restaurant, Ingredient et Stock 7

3.1.4 - Recipe, Ingredient et IngredientQuantity 7

3.1.5 - Order, Article et ArticleQuantity 8

3.1.6 - DeliveryMode et PaymentType 8

3.2 - Modèle Logique de Données 9

3.3 - Modèle Physique de Données 10

4 - Comparatif des méthodes 11

4.1.1 - Tableau comparatif 11

4.1.2 - Choix de la méthode 12

5 - Macroplanning 13

5.1 - Matrice RACI 15

5.1.1 - R - Responsible 15

5.1.2 - A - Accountable 15

5.1.3 - C - Consulted 15

5.1.4 - I - Informed 15

# Versions

|  |  |  |  |
| --- | --- | --- | --- |
| Auteur | Date | Description | Version |
| Mickaël HORN | 08/09/2022 | Création et début du remplissage du document | 1.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Objet du document

Le présent document constitue le dossier de conception techniques des applications d’OC Pizza, à savoir le site web client ainsi que le système informatique interne (site web interne).

Les objectifs du présent document sont de fournir le domaine fonctionnel, caractérisé par les différents éléments suivants :

* Modèle Conceptuel de Données (Diagramme de classes)
* Modèle Logique de Données
* Modèle Physique de Données

De plus, seront présentés un tableau comparatif des méthodologies de projet afin d’en choisir une qui est cohérente aux objectifs.

Par la suite, vous trouverez un macroplanning schématisé par un diagramme de GANTT, afin de représenter sur un axe temporel les différentes tâches du projet et, pour finir, la matrice RACI qui permettra de définir qui fait quoi pour chacune des tâches

## Références

Pour de plus amples informations, se référer également aux éléments suivants:

1. **DCT - Xxx** : Dossier de conception fonctionnelle de l'application
2. **DCT - Xxx** : Dossier d’exploitation
3. **DCT - Xxx** : Procès verbal de livraison finale

## Besoins du client

### Rappel du contexte

« OC Pizza » est un jeune groupe de pizzeria en plein essor et spécialisé dans les pizzas livrées ou à emporter.

Il compte déjà 5 points de vente et prévoit d’en ouvrir au moins 3 de plus d’ici la fin de l’année.

Malgré ces bonnes nouvelles, leur système informatique ne correspond plus à leur besoin, et ils veulent que nous en réalisions un sur-mesure, déployé dans toutes les pizzerias et qui réponde aux exigences citées dans la prochaine partie.

### Rappel des enjeux et objectifs

* 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 et en préparation
* Suivre en temps réel le stock d’ingrédients restants pour savoir quelles pizzas sont encore réalisables
* 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’il le souhaite, 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
* Informer ou notifier les clients sur l’état de leur commande

# Domaine Fonctionnel

## Modèle Conceptuel de Données (Diagramme de classes)

Décortiquons un peu les spécificités de ce diagramme de classe.

### Employee et Restaurant

Ici, nous avons une agrégation qui se justifie par le fait qu’un restaurant est constitué d’un ensemble d’employés.

### Héritage Employee

Notion d’héritage avec les différents types d’employé.

Chaque employée hérite de la super classe Employee mais n’ont pas les mêmes rôles.

### Restaurant, Ingredient et Stock

La table Stock est née du fait de la classe d’association entre Restaurant et Ingredient.

De cette façon, chaque ingrédient aura son restaurant associé ainsi que ses informations telles que la quantité, le type de quantité et la date d’ajout.

### Recipe, Ingredient et IngredientQuantity

Même logique que pour le stock, nous avons de nouveau une classe d’association plusieurs à plusieurs mais cette fois ci, pour savoir la quantité d’ingrédient de nos recettes.

De plus, une aggrégation est située entre Recipe et Ingredient car une recette n’est rien de plus qu’un ensemble d’ingrédients.

### Order, Article et ArticleQuantity

Un client peut commander plusieurs fois la même pizza et on aura cette information en ayant la commande associée, ainsi que l’article concerné.

D’où la classe d’association ArticleQuantity.

### DeliveryMode et PaymentType

Grace à ces deux énumérations, on va pouvoir connaître tous les détails concernant le mode de livraison ainsi que du type de paiement.

On couvre tous les différents cas, ce qui nous permettra d’avoir un historique de commandes précis.

## Modèle Logique de Données

## Modèle Physique de Données

Intéressons-nous maintenant au Modèle Physique de Données.

La base de données a été créée avec l’aide de l’outil MySQLWorkbench.

Vous trouverez trois fichiers dans l’arborescence à l’intérieur du dossier MPD :

1. **horn\_mickael\_3\_mpd\_062022.mwb** : correspond à la représentation de la base de données sur l’outil MySQLWorkbench
2. **horn\_mickael\_5\_create\_database\_062022.sql** : le fichier à lancer afin de créer la base de données
3. **horn\_mickael\_6\_jeu\_de\_donnees\_062022.sql** : enfin, le fichier contenant notre jeu de données

# Comparatif des méthodes

### Tableau comparatif

Il nous faut choisir une méthodologie pour notre projet.

Voici un comparatif des différentes méthodologies à disposition qui aidera à choisir laquelle est la plus concrète pour notre situation.

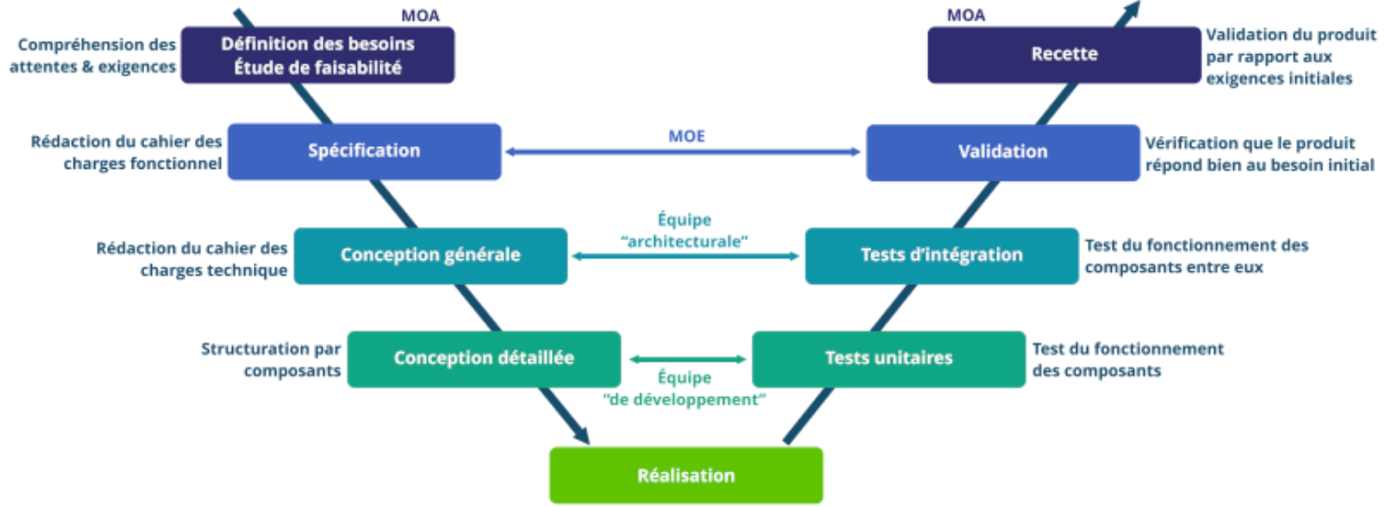
Une image contenant texte

Description générée automatiquement

### Choix de la méthode

Après avoir pris en compte les différences parmi les méthodologies de projet citées plus haut, nous devons prendre une décision sur quelle méthode appliquer.

Le cycle en V semble adapté à notre cas, et nous allons détailler pourquoi.



*Les étapes du cycle en V*

Le client a clairement exprimé ses besoins, sous forme de liste, que nous avons préalablement étayés dans les spécifications fonctionnelles, ce qui est déjà très rassurant pour la réalisation de ce projet avec une méthodologie comme le cycle en V.

Il semble savoir où il va et on peut donc en déduire qu’il ne devrait pas y avoir de modifications importantes qui pourraient compromettre notre méthodologie séquentielle et linéaire.

Nous pouvons nous passer de la méthodologie agile, car nous n’avons pas besoin d’avancer « à petit pas » étant donné qu’on a affaire à un projet bien cadré.

La méthodologie en cascade est plus simple, mais le cycle en V, du fait de ses tests complets, nous assureras une livraison de qualité.

Pour terminer, je m’appuie sur un élément concret qui est survenu lors de la construction des spécifications fonctionnelles qui est la date de livraison.

En effet, le client souhaite que le projet soit livré pour l’ouverture des 3 nouvelles pizzérias, c’est-à-dire dans 6 mois.

Le client a déjà son idée sur la date de livraison et nous devons lui fournir une vraie date.

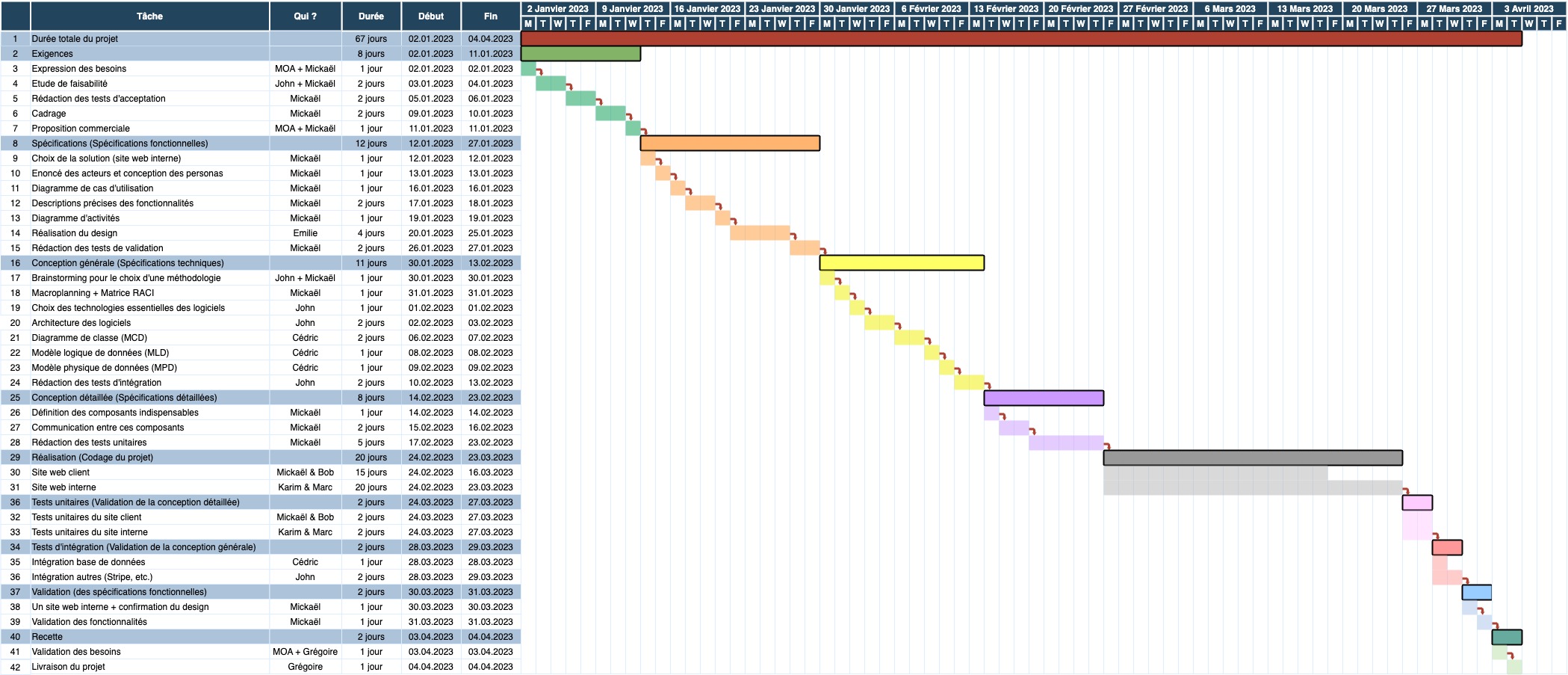
C’est un des plus gros défauts de la méthodologie agile, ce manque de visibilité sur le temps, on n’a pas d’idée précise du temps et du coût final.

# Macroplanning

Voici maintenant un macroplanning sous forme de diagramme de Gantt nous permettant de représenter le projet de manière temporelle.

Équipe :

* Développement -> Mickaël (Lead Developer), Bob, Karim et Marc (Développeurs)
* MOA -> Client
* Équipe Architecturale -> John (Chef De Projet Architecture) et Cédric (Spécialiste base de données)
* Équipe Studio -> Émilie (Design)
* Recette -> Grégoire



Ce diagramme de Gantt reprend de manière logique les différentes étapes de la méthodologie que nous avons choisie.

Comme le cycle en V est une méthodologie de projet séquentielle et linéaire, les étapes s’enchaînent les unes après les autres et non en même temps.

Les tâches, quant à elles, peuvent être réalisées en parallèles, à la condition qu’elles viennent de la même étape.

Par exemple, on constate dans la partie « Réalisation (Codage du projet) », que les deux tâches (le site web côté client et le site web interne) qui la compose sont réalisées en même temps, par deux personnes différentes.

Les week-ends ne sont pas comptés, uniquement les jours ouvrables.

## Matrice RACI

Intéressons-nous maintenant à la matrice RACI, qui nous offre la possibilité de voir qui fait quoi dans les différentes tâches à accomplir.

### R - Responsible

La personne responsable de la tâche.

### A - Accountable

Celle qui effectue la tâche.

### C - Consulted

C’est la personne qu’on va consulter pour avoir son expertise si on a besoin d’aide.

### I - Informed

Si l’on a besoin d’informer quelqu’un à propos de la tâche.

Une image contenant table

Description générée automatiquement

Une image contenant table

Description générée automatiquement