Concevez la solution technique de votre client pour son groupe de pizzerias

Baptiste Fina

29 juin 2018

Table des matières

I Introduction **2**

II Domaine fonctionnel **4**

1 Règles de gestion fonctionnelle **5**

2 Diagramme de classe **6**

3 Base de données et modèle physique de données **8**

III Description physique du système **10**

1 Composants du système **11**

2 Diagramme de déploiement **13**

IV Conclusion **15**

Première partie

Introduction

Dans ce rapport, nous allons décrire toutes les étapes nécessaires à l'établissement de la base de données du site OCPizza. Cette base de données aura pour but d'agréger toutes les données concernant à la fois les utilisateurs d'OCPizza, les clients et les employés, ainsi que toutes les données propres à chaque restaurant telles que les stocks d'ingrédients.

Nous allons commencer par définir le domaine fonctionnel du système. Cela consistera à commencer par définir les règles de gestion fonctionnelle, puis à tracer le diagramme de classe du système, afin d'identifier tous les objets nécessaires à son bon fonctionnement. Ce diagramme fait, nous pourrons passer au tracer du schéma de la base de données avant de le compléter avec le modèle physique de données. Ce diagramme sera la visualisation de la base de données associée à chaque restaurant OCPizza ainsi que les règles régissant leurs relations.

Enfin, nous finirons par décrire l'implémentation physique du système. Pour cela, nous commencerons par établir un diagramme de composants puis un diagramme de déploiement. Le premier permettra de prendre un peu de recul sur le travail précédent en décrivant le fonctionnement du système du point de vue de ses fonctionnalités et de leurs relations. Nous finirons par la description physique du système, donc quel hardware utiliser.

Deuxième partie

Domaine fonctionnel

Chapitre 1

Règles de gestion fonctionnelle

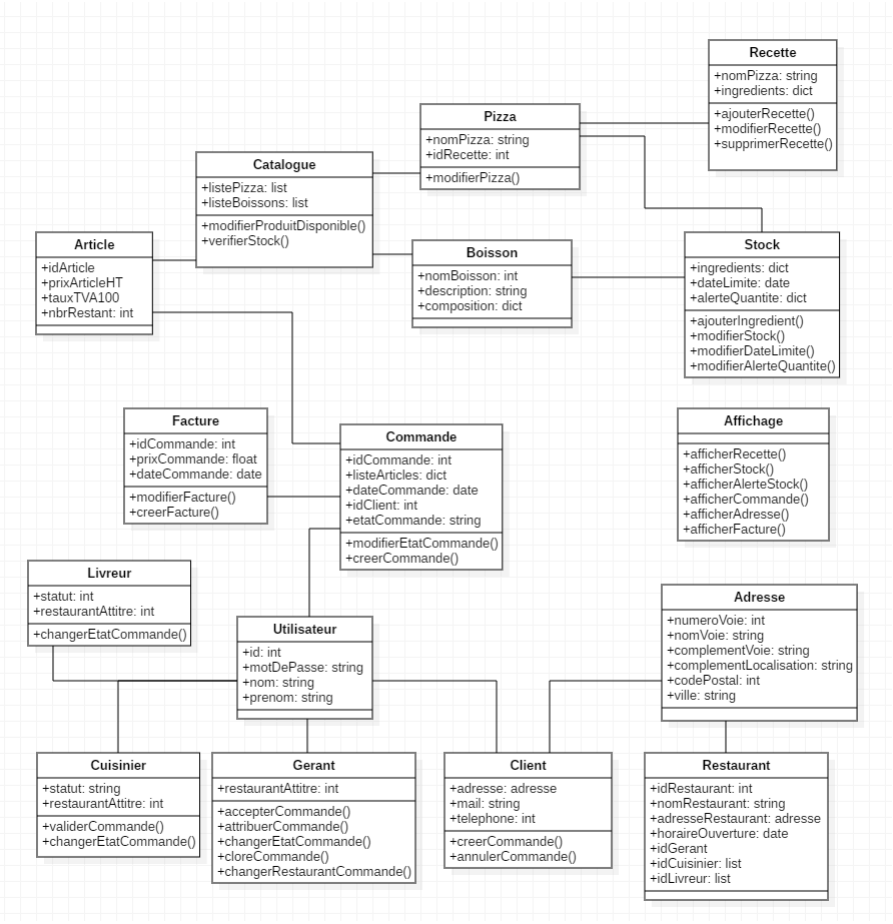
	client	cuisinier	livreur	gérant
CRÉER commande	✓			
PAYER commande	✓			
ANNULER commande	✓			
ACCEPTER commande				✓
ATTRIBUER commande				✓
ATTRIBUER livraison				✓
CLORE commande				✓
CONSULTER stocks		✓		✓
AFFICHER recette		✓		
AFFICHER adresse client			✓	
MODIFIER restaurant en charge de la commande				✓

Afin de préparer la conception du diagramme de classe, de la base de données et du domaine fonctionnel, nous allons commencer par créer un tableau décrivant les permissions accordées à chaque utilisateur. Nous avons retenu quatre utilisateurs : le client, le cuisinier, le livreur et le gérant.

Le client n'a accès qu'au site web et pas à la base de données. Ses seules actions possibles sont donc de créer une commande, la payer et l'annuler si elle n'a pas été validée par le gérant du restaurant. Les cuisiniers n'ont besoin que de consulter les stocks et les recettes des pizzas commandées, tandis que les livreurs auront comme seule action possible d'afficher l'adresse de livraison. L'utilisateur ayant le plus de droits, et donc de responsabilité sera le gérant de chaque restaurant. Sa tâche sera d'accepter ou non les commandes en fonction de la charge du restaurant, gérer les stocks et attribuer les commandes aux cuisiniers et livreurs en fonction de leur disponibilité.

Chapitre 2

Diagramme de classe



Le diagramme de classe est un schéma qui permet de présenter les classes et interfaces des systèmes ainsi que leurs relations. Il nous servira de base pour réaliser le schéma de la base de données dans la partie suivante. Dans la partie précédente, nous avons défini les quatre utilisateurs qui interagiront

avec le système. Il est donc logique de créer quatre classes pour ces utilisateurs. Cela permet de délimiter correctement les attributions de chaque classe, donc de chaque utilisateur. Les quatre classes d'utilisateurs partagent quelques caractéristiques communes telles qu'un id ou un mot de passe. Il est donc logique de créer une classe mère dont vont hériter les quatres classes.

Le deuxième grand groupe de classes de diagramme concerne les articles vendus par les restaurants. On retrouve donc deux classes Pizzas et Boissons qui regroupent les deux types d'articles disponibles dans un restaurant. Ces deux classes se rejoignent dans une classe Catalogue, afin de pouvoir faciliter son affichage. La gestion des stocks est modélisée par la classe du même nom. A chaque ingrédient est associé une quantité restante, une date limite de consommation et enfin une alerte sur la quantité restante. Cela permettra au gérant du restaurant d'être informé en temps réel des stocks disponibles pour éventuellement modifier le catalogue. Ensuite, à chaque pizza nous avons associé une recette. Cela permettra aux cuisiniers d'avoir pour chaque commande la composition et le grammage de chaque pizza à préparer afin de faciliter son travail.

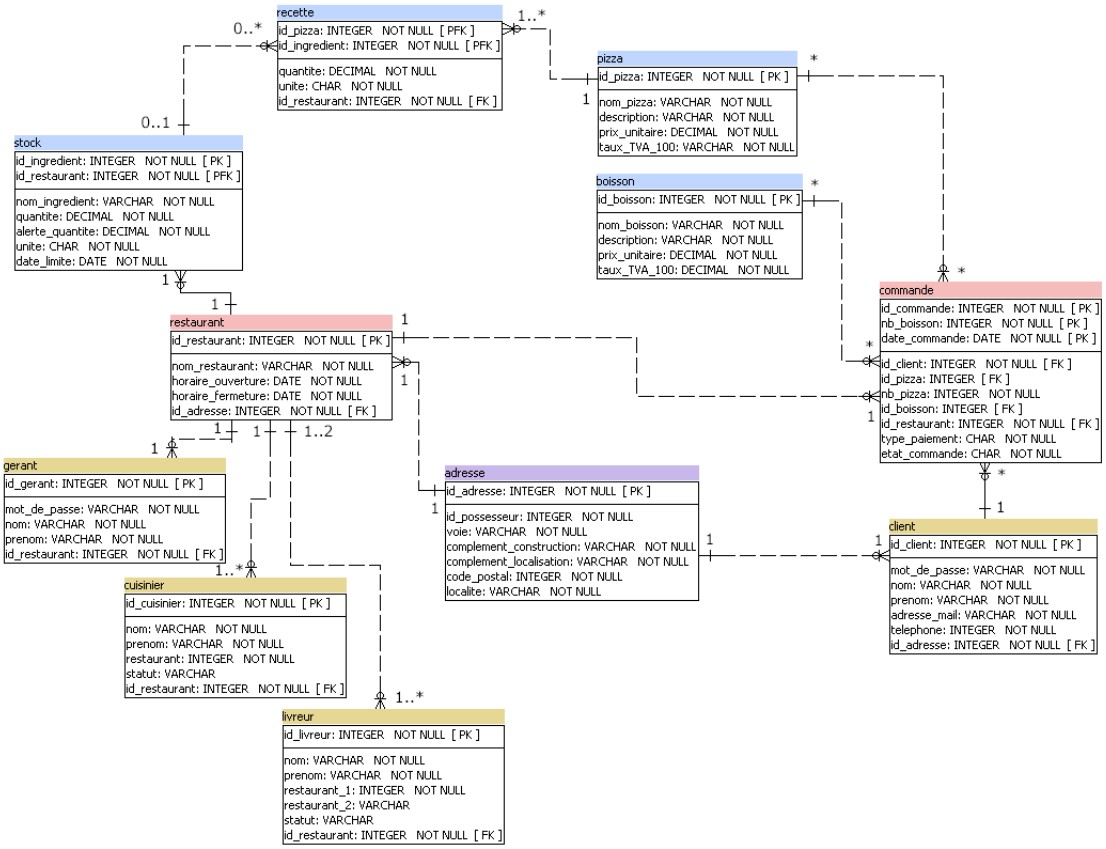
Le troisième groupe de classes est le plus important puisqu'il fait la liaison entre les deux groupes précédents. Il s'agit du groupe des commandes et est composé des classes Restaurant, Articles, Commandes et Facture. La première classe permet de décrire tous les restaurants composants le groupe OCPizza, avec leur adresse et l'équipe qui les compose. La deuxième classe, Article, permet d'attribuer à chaque produit un prix et une disponibilité en fonction du stock. C'est surtout cette classe qui permettra de créer une commande, puisque chaque instance de la classe Commande sera composée d'articles en plus d'une date de commande, de l'id du client et de l'état de la commande. Enfin, la classe Facture permettra aux gérants des restaurants d'établir les factures chaque jour facilement.

Enfin, les deux dernières classes restantes dans ce diagramme sont Affichage et Adresse. La classe Adresse a été créée pour alléger les classes Client et Restaurant, car c'est un élément qui nécessite beaucoup de détails. De même pour la classe Affichage, au lieu de définir pour chaque classe des méthodes pour les afficher, il est plus sain de définir une classe entièrement dédiée à cette tâche.

Cette décomposition permet de bien identifier tous les processus qui entrent en jeu lors de la prise de commande. La réalisation du modèle physique de données peut maintenant être réalisée facilement.

Chapitre 3

Base de données et modèle physique de données



Maintenant que nous avons décrit le diagramme de classe du système, nous pouvons modéliser la base de données. C'est elle qui contiendra toutes les données concernant le bon fonctionnement de chaque restaurant. Nous re-

trouvons l'essentiel des classes décrites précédemment mais sous forme de tables. Les classes Facture, Article et Catalogue ont néanmoins disparues, elles n'étaient pas nécessaires au bon fonctionnement de la base et avaient même tendance à le complexifier. Les attributs de ces objets ont donc été reportés vers les classes correspondantes. Nous observons tout de mê me que les grands groupes de classes se retrouvent.

Du fait des grandes similarités entre le diagramme de classe et le modèle physique de données, nous allons plutôt nous attacher à décrire les liens entre les différentes tables. Afin d'éviter les erreurs de saisie, dès que possible, nous avons implémenté des clés étrangères entre les tables. Nous observons ainsi que certaines tables jouent un rô le beaucoup plus important que d'autres.

La table Restaurant est en interaction avec de nombreuses autres tables : elle permet en effet d'associer une commande à un restaurant, de mê me pour les stocks. Elle nous sert aussi à associer chaque employé à un restaurant, sauf pour les livreurs qui peuvent dépendre de plusieurs restaurants. Ce nombre est pour l'instant fixé à deux, du fait de la taille relativement restreinte du nombre de restaurants OCPizza. Cette feature permettra, si un des restaurants est surchargé, de transférer une partie des livreurs afin de répondre à la demande plus facilement.

La deuxième table qui concentre une grande partie des interactions, c'est la table commande. C'est logique, étant donné son rôle vital dans le fonctionnement du système. Une commande dépend logiquement d'un client, d'un restaurant où elle sera réalisée, mais aussi du catalogue d'articles actuellement disponibles.

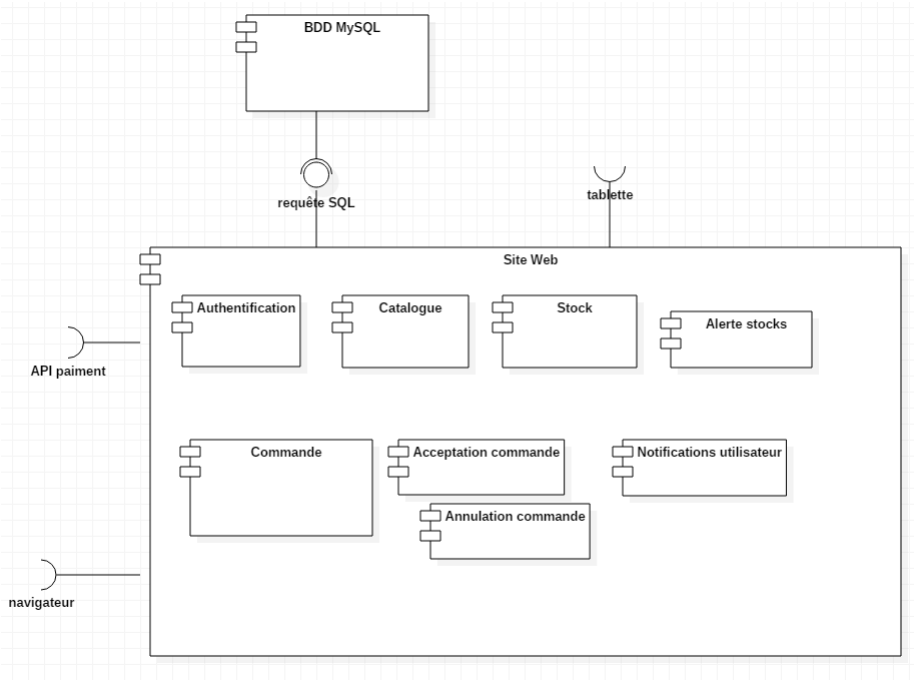
Nous obtenons donc au final une base de données relativement délicate à construire, compte tenu de toutes les dépendances, mais qui permettra pendant son utilisation toute mauvaise manipulation.

Troisième partie

Description physique du système

Chapitre 1

Composants du système



Nous allons maintenant décrire le fonctionnement du système en prenant un peu de hauteur. Nous allons réaliser un diagramme de composants, afin d'avoir une vue d'ensemble sur toutes les fonctionnalités.

Le diagramme est composé d'un bloc central : le site web. C'est en effet lui qui regroupe l'essentiel des fonctionnalités qui nous intéressent. Ce bloc est lui-même composé de sous-parties, ou composants que nous allons détailler. On retrouve donc l'authentification, le catalogue, le stock et l'alerte des stocks sur la première ligne. Ce sont les fonctionnalités les plus basiques nécessaires au fonctionnement du site. Ainsi, le catalogue est impacté par les stocks, et si ceux-ci sont trop bas une alerte s'affichera sur la tablette du

gérant du restaurant en question.

La deuxième ligne de sous-composants comporte les actions relatives à la commande par un utilisateur. On retrouve donc logiquement la commande, la possibilité pour l'utilisateur de l'annuler si elle n'a pas été acceptée par le gérant, les notifications utilisateur qui permettent au restaurant de tenir au courant l'utilisateur de l'avancement de sa commande.

Le site web, vital au bon fonctionnement du système, a néanmoins besoin de composants extérieurs pour fonctionner. De toute évidence, l'utilisateur a besoin d'un navigateur pour se rendre sur le site de la pizzeria et créer sa commande. L'interface entre les deux se fait par le protocole hhtps, qui est désormais la norme de transfert.

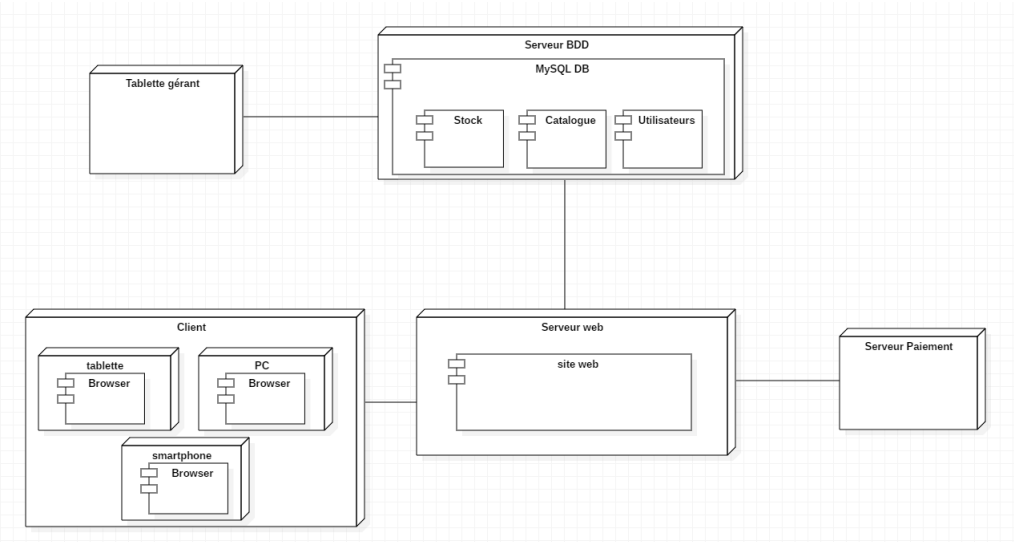
Afin d'effectuer son paiement, s'il choisit de le faire en ligne, il faudra un composant dédié au paiement, via Paypal ou le site de la banque du client.

Nous avons vu que le catalogue du site web du restaurant se mettait à jour en fonction des stocks. Il faut donc pour cela connecter le site web à la base de données du même site. En fonction, des stocks, le catalogue sera donc facilement mis à jour.

Cette liaison entre le site web et la base de données permettra aussi au gérant de chaque restaurant de consulter les stocks actuels et d'être éventuellement alerté s'ils sont trop bas via une tablette dédiée à cet usage.

Chapitre 2

Diagramme de déploiement



Le diagramme de déploiement sert à représenter l'infrastructure du système. Dans notre cas, elle est assez simple. Elle consiste en un serveur hébergeant le site web où les clients pourront passer leurs commandes, et d'un autre serveur qui hébergera lui la base de données. Les deux serveurs sont séparés pour des raisons de sécurité. En effet, la base de données héberge les informations personnelles des utilisateurs et des restaurants. Il ne serait donc pas prudent de tout mettre sur un même serveur.

Les clients doivent pouvoir accéder au site via le navigateur de leur ordinateur, smartphone ou tablette. Les deux derniers se démocratisant de plus en plus au détriment du premier concernant l'usage du web. Les gérants auront quant à eux accès à la base de données via une tablette. Celle-ci leur permettra de consulter les stocks, les commandes en cours et d'attribuer aux livreurs et cuisiniers disponibles les nouvelles commandes.

Quatrième partie

Conclusion

Nous avons désormais décrit tout le fonctionnement du système : depuis la description des fonctions attribuées à chaque objet jusqu'à l'implémentation physique du système.

Nous avons une base de données qui devrait pouvoir répondre Ã toutes les situations qui se prÃ©sentent. Nous avons vu aussi que pour fonctionner il faudra deux serveurs : l'un pour hÃ©berger le site web, l'autre pour la base de donnÃ©es. Enfin, chaque gÃ©rant sera muni d'une tablette afin de gÃ©rer les flux de commandes.