



OCPizza spécifications techniques

Table des matières

Introduction.....	3
Rappel du contexte.....	3
Rappel des besoins du client.....	4
Description du domaine fonctionnel.....	4
1) Introduction.....	4
2) Diagramme de classes.....	5
3) Description des classes.....	7
- « PointOfSale ».....	7
- « User ».....	7
- « Worker ».....	8
- « Customer ».....	8
- « Address ».....	9
- « Product ».....	9
- « Stock ».....	10
- « Pizza ».....	10
- « PizzaPointOfSale ».....	11
- « PromoCode ».....	11
- « OtherProduct ».....	11
- « OtherProductPointOfSale ».....	12
- « Recipe ».....	12
- « RecipeProduct ».....	13
- « Step ».....	13
- « ShoppingBaket ».....	13
- « CustomerOrder ».....	14
- « CurrentState ».....	14
- « Receipt ».....	15
- « PayementMethod ».....	15
Modèle physique de données.....	16
1) Diagramme.....	16
2) Descriptions des tables.....	17
- « Point_of_sale ».....	17
- « User_app ».....	17
- « Worker ».....	18
- « Customer ».....	18
- « Address ».....	19
- « Product ».....	19
- « Stock ».....	19
- « Pizza ».....	20
- « Pizza_point_of_sale ».....	20
- « Promo_code ».....	21
- « Other_product ».....	21
- « Other_product_point_of_sale ».....	22

- « Recipe ».....	22
- « Recipe_product ».....	22
- « Step ».....	23
- « Shopping_basket ».....	23
- « Customer_order ».....	24
- « Receipt ».....	25
- « Payment_method ».....	26
Diagramme de composant.....	26
Diagramme de déploiement.....	27

Introduction.

Ce document va vous présenter les spécifications techniques réalisées pour le projet commandité par le groupe « OCPizza ». C 'est-à-dire :

- La définition du domaine fonctionnel.
- L 'identification des différents éléments composant le système à mettre en place, ainsi que leurs interactions.
- La description du déploiement des différents composants.
- L 'élaboration du schéma de la base de données

Rappel du contexte.

« OCPizza » 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. Un des responsables du groupe a pris contact avec nous afin de mettre en place un système informatique à déployer dans toutes ses pizzerias.

Rappel des besoins du client.

- Être plus efficace dans la gestion des commandes, de leur réception à leur livraison en passant par leur préparation;
- De suivre en temps réel les commandes passées et en préparation;
- De suivre en temps réel le stock d'ingrédients restants pour savoir quelles pizzas sont encore réalisables;
- De 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
- De proposer un aide-mémoire aux pizzaiolos indiquant la recette de chaque pizza
- D'informer ou notifier les clients sur l'état de leur commande

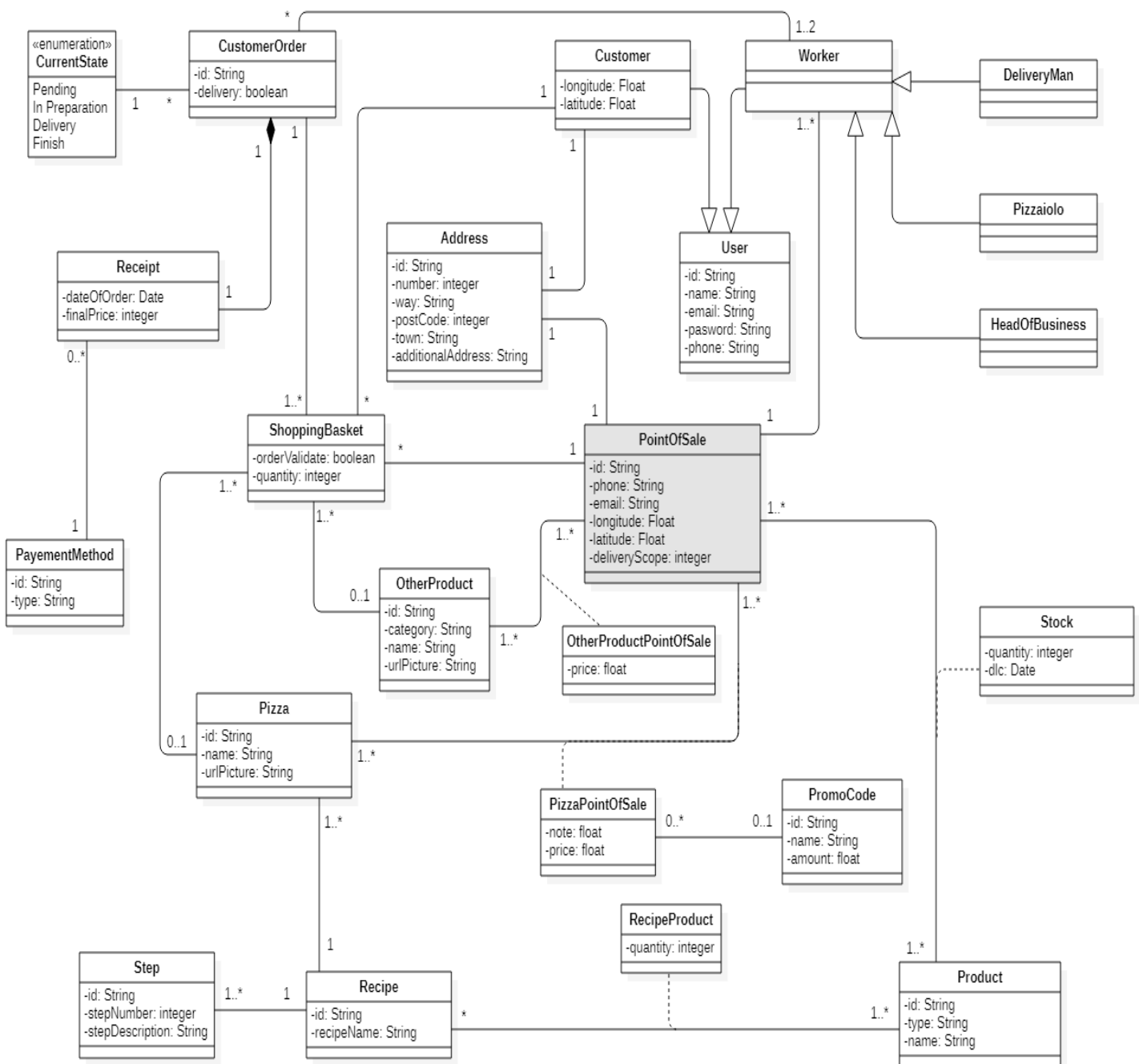
Description du domaine fonctionnel.

1) Introduction.

La modélisation du domaine fonctionnel est réalisé en UML à l'aide du logiciel « StarUML ».

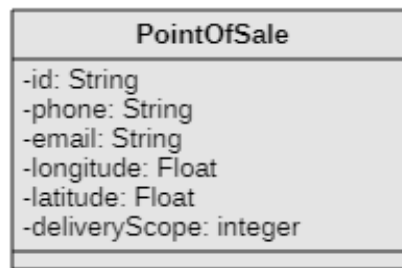
2) Diagramme de classes.

[Cliquez ici pour agrandir.](#)



3) Description des classes.

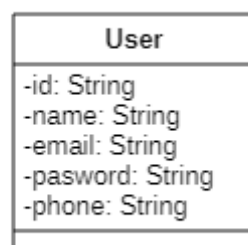
- « PointOfSale »



La classe « PointOfSale » représente les différents points de vente, identifiable par un « id ».

Elle disposera des champs, « longitude » et « latitude », qui permettront de calculer la distance lors du passage d'une commande et ainsi attribuer la commande au point de vente le plus proche et également l'activation de l'option livraison, si la distance rentre dans le périmètre de livraison prédéfini dans le champ « deliveryScope ».

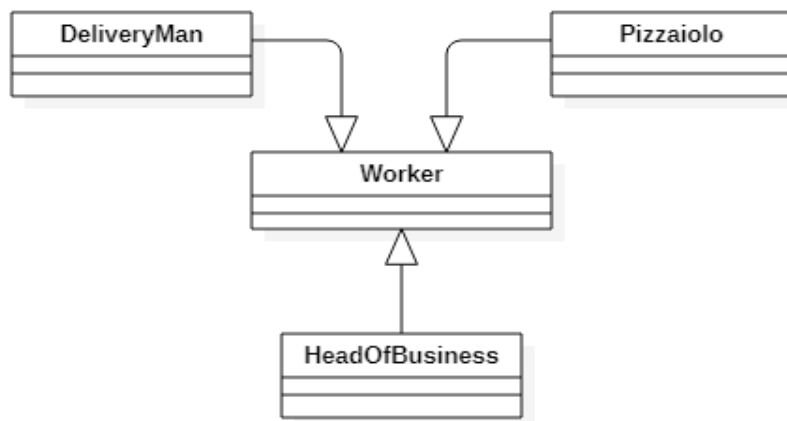
- « User »



La classe « User » est la classe mère pour les différents utilisateurs de l'application.

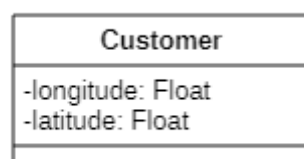
Elle dispose entre autres des différents champs requis pour l'authentification.

- « Worker »



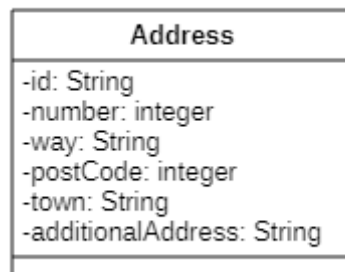
La classe « Worker » est une classe mère qui étend de « User ». Comme le montre le diagramme ci-dessus elle dispose de 3 classes fille qui représente les différents groupes d'employés.

- « Customer »



La classe « Customer » est une classe qui étend de « User ». Elle représente les différents clients qui utilise l'application on y retrouve les champs « longitude » et « latitude » pour le calcul de distance (voir « PointOfSale»).

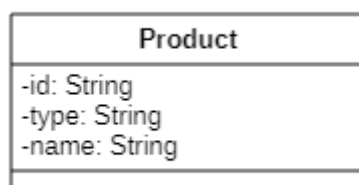
- « Address »



La classe « Address » est une classe qui permet de disposer d'un maximum d'informations sur l'adresse utile pour la livraison.

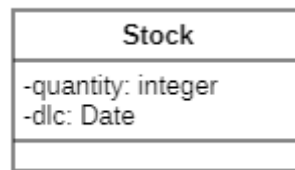
Elle est en relation avec les classes « Customer » et « PointOfSale ».

- « Product »



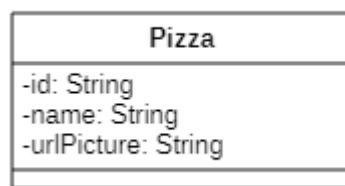
La classe « Product » représente les différents produits du stock des points de vente. Que ce soit pour la réalisation des pizzas mais également pour le ménage ou autres.

- « Stock »

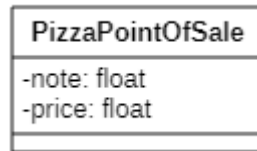


La classe « Stock » est la classe de liaison entre « Stock » et « Product ». Elle dispose du champ « quantity » qui sert à déterminer le nombre de produits présents dans le stock du point de point. Et le champ « dlc » (date limite de consommation).

- « Pizza »

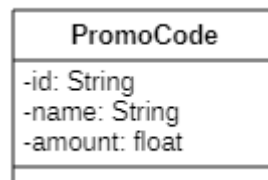


La classe « Pizza » représente les différentes pizzas vendues par les points de vente.

- « PizzaPointOfSale »

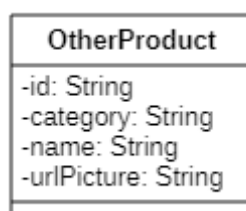
La classe « PizzaPointOfSale » est la classe de liaison entre « Pizza » et « PizzaPointOfSale ».

Elle possède les champs « note » et « price » afin d’attribuer une note et un prix différents aux pizzas en fonction du point de vente.

- « PromoCode »

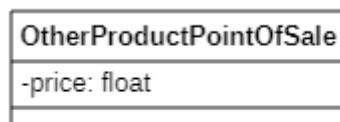
La classe « PromoCode » représente les différents codes promo à attribuer aux pizzas.

Elle est liée à la classe de liaison « PizzaPointOfSale », afin de permettre d’appliquer différents codes promo en fonction du point de vente.

- « OtherProduct »

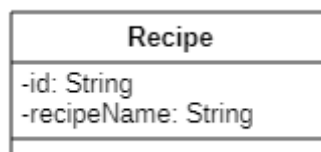
La classe « OtherProduct » représente les autres produits vendus par les points de vente (boisson, dessert...).

- « OtherProductPointOfSale »

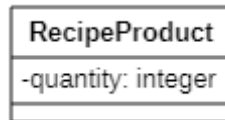


La classe « OtherProductPointOfSale » est la classe de liaison entre « OtherProduct » et « PointOfSale » afin de pouvoir attribuer un prix différent en fonction du point de vente.

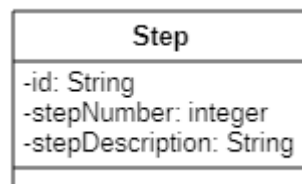
- « Recipe »



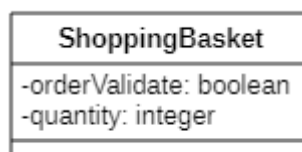
La classe « Recipe » représente les différentes recettes pour chaque pizza.

- « RecipeProduct »

La classe « RecipeProduct » est la classe de liaison entre « Recipe » et « Product ». Afin de déterminer la quantité de produit nécessaire pour chaque recette.

- « Step »

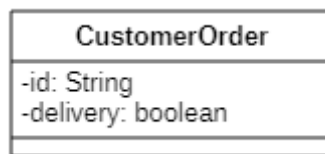
La classe « Step » représente les différentes étapes nécessaires à la réalisation d'une recette.

- « ShoppingBaket »

La classe « ShoppingBasket » représente le panier. Elle servira à faire le lien entre le client, le produit acheté et le point de vente. C'est là que l'on pourra attitrer un point de vente à une commande en fonction de la distance (voir « PointOfSale ») et d'activer l'option livraison.

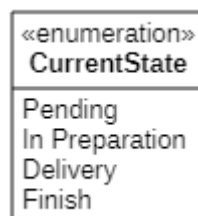
Le champ « orderValidate » permet de récupérer la commande en cours pour l'affichage du panier.

- « CustomerOrder »



La classe « CustomerOrder » représente la commande après validation du panier. Elle comporte les différents employés impliqués dans la réalisation de la commande, ainsi que l'état actuel de la commande (en préparation, en livraison, etc...).

- « CurrentState »



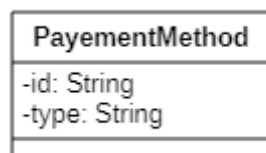
« CurrentState » est une énumération des différents états d'une commande.

- « Receipt »



La classe « Receipt » représente la facture finale après la réalisation complète d'une commande.

- « PaymentMethod »

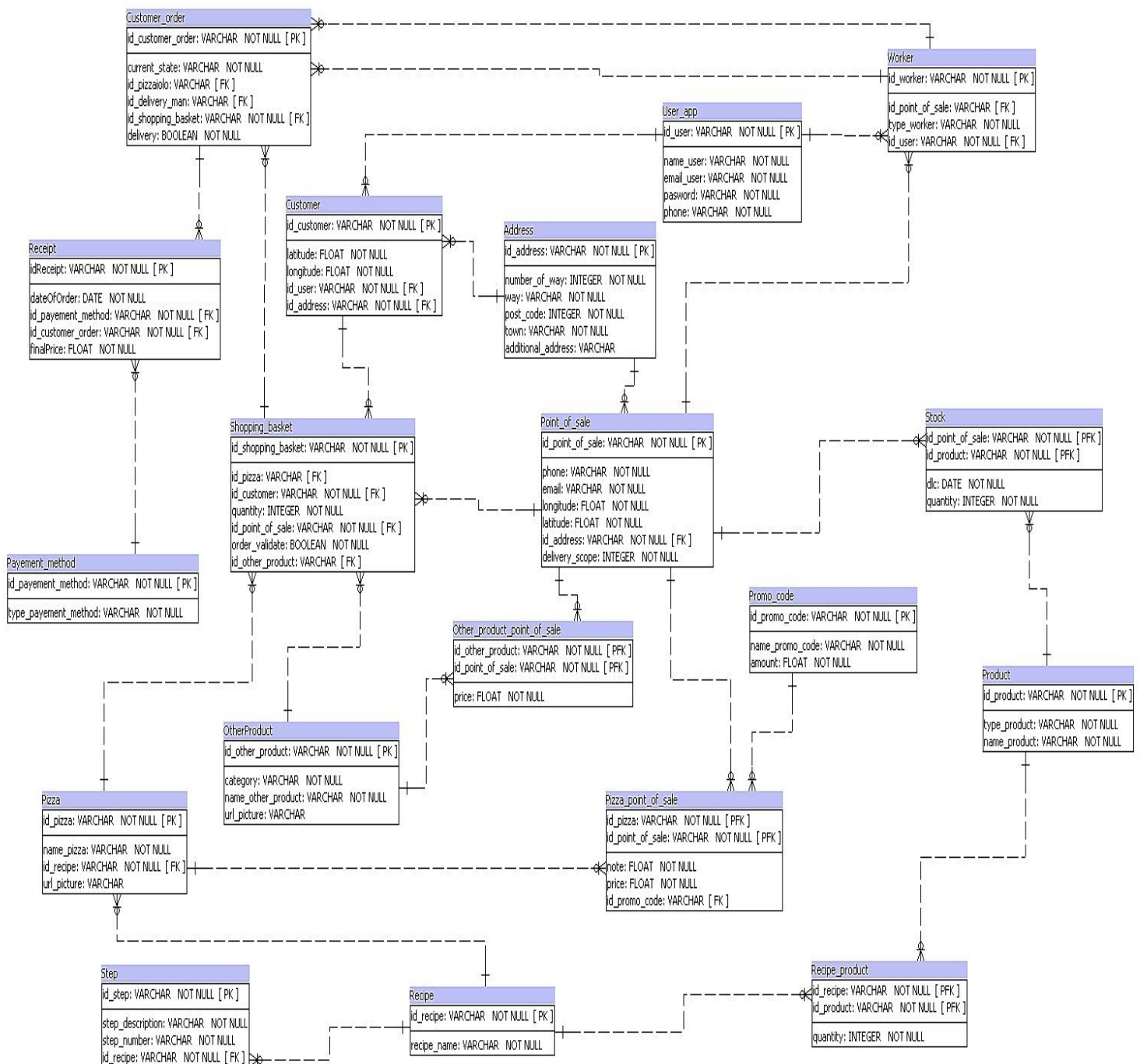


La classe « PaymentMethod » représente les différentes possibilités de paiement proposer.

Modèle physique de données.

1) Diagramme

[ici](#)



2) Descriptions des tables.

- « Point_of_sale »

Point_of_sale
id_point_of_sale: VARCHAR NOT NULL [PK]
phone: VARCHAR NOT NULL
email: VARCHAR NOT NULL
longitude: FLOAT NOT NULL
latitude: FLOAT NOT NULL
id_address: VARCHAR NOT NULL [FK]
delivery_scope: INTEGER NOT NULL

Clé étrangère sur « id_address » :

Table	Colonne	On delete	On update
Address	id_address	No action	No action

- « User_app »

User_app
id_user: VARCHAR NOT NULL [PK]
name_user: VARCHAR NOT NULL
email_user: VARCHAR NOT NULL [AK]
password: VARCHAR NOT NULL
phone: VARCHAR NOT NULL

Il existe une contrainte d'unicité sur le champ « email_user » pour éviter tous conflits de connexion.

- « Worker »

Worker
id_worker: VARCHAR NOT NULL [PK]
id_point_of_sale: VARCHAR [FK]
type_worker: VARCHAR NOT NULL
id_user: VARCHAR NOT NULL [FK]

Clé étrangère sur « id_point_of_sale » :

Table	Colonne	On delete	On update
Point_of_sale	id_point_of_sale	No action	No action

Null = « head of buisness »

Clé étrangère sur « id_user » :

Table	Colonne	On delete	On update
User_app	id_user	Cascade	No action

- « Customer »

Customer
id_customer: VARCHAR NOT NULL [PK]
latitude: FLOAT NOT NULL
longitude: FLOAT NOT NULL
id_user: VARCHAR NOT NULL [FK]
id_address: VARCHAR NOT NULL [FK]

Clé étrangère sur « id_address » :

Table	Colonne	On delete	On update
Address	id_address	No action	No action

Clé étrangère sur « id_user » :

Table	Colonne	On delete	On update
User_app	id_user	Cascade	No action

- « Address »

Address
id_address: VARCHAR NOT NULL [PK]
number_of_way: INTEGER NOT NULL
way: VARCHAR NOT NULL
post_code: INTEGER NOT NULL
town: VARCHAR NOT NULL
additional_address: VARCHAR

- « Product »

Product
id_product: VARCHAR NOT NULL [PK]
type_product: VARCHAR NOT NULL
name_product: VARCHAR NOT NULL

- « Stock »

Stock
id_point_of_sale: VARCHAR NOT NULL [PFK]
id_product: VARCHAR NOT NULL [PFK]
dlc: DATE NOT NULL
quantity: INTEGER NOT NULL

Clé primaire étrangère sur « id_point_of_sale » :

Table	Colonne	On delete	On update
Point_of_sale	id_point_of_sale	Cascade	No action

Clé primaire étrangère sur « id_product » :

Table	Colonne	On delete	On update
Product	id_product	Cascade	No action

- « Pizza »

Pizza
id_pizza: VARCHAR NOT NULL [PK]
name_pizza: VARCHAR NOT NULL
id_recipe: VARCHAR NOT NULL [FK]
url_picture: VARCHAR

Clé étrangère sur « id_recipe » :

Table	Colonne	On delete	On update
Recipe	id_recipe	Cascade	No action

- « Pizza_point_of_sale »

Pizza_point_of_sale
id_pizza: VARCHAR NOT NULL [PFK]
id_point_of_sale: VARCHAR NOT NULL [PFK]
note: FLOAT NOT NULL
price: FLOAT NOT NULL
id_promo_code: VARCHAR [FK]

Clé primaire étrangère sur « id_point_of_sale » :

Table	Colonne	On delete	On update
Point_of_sale	id_point_of_sale	Cascade	No action

Clé primaire étrangère sur « id_pizza » :

Table	Colonne	On delete	On update
Pizza	id_pizza	Cascade	No action

Clé étrangère sur « id_promo_code » :

Table	Colonne	On delete	On update
Promo_code	id_promo_code	No action	No action

- « Promo_code »

Promo_code
id_promo_code: VARCHAR NOT NULL [PK]
name_promo_code: VARCHAR NOT NULL
amount: FLOAT NOT NULL

- « Other_product »

OtherProduct
id_other_product: VARCHAR NOT NULL [PK]
category: VARCHAR NOT NULL
name_other_product: VARCHAR NOT NULL
url_picture: VARCHAR

- « Other_product_point_of_sale »

Other_product_point_of_sale
id_other_product: VARCHAR NOT NULL [PFK]
id_point_of_sale: VARCHAR NOT NULL [PFK]
price: FLOAT NOT NULL

Clé primaire étrangère sur « id_point_of_sale » :

Table	Colonne	On delete	On update
Point_of_sale	id_point_of_sale	Cascade	No action

Clé primaire étrangère sur « id_other_product » :

Table	Colonne	On delete	On update
Other_product	id_other_product	Cascade	No action

- « Recipe »

Recipe
id_recipe: VARCHAR NOT NULL [PK]
recipe_name: VARCHAR NOT NULL

- « Recipe_product »

Recipe_product
id_recipe: VARCHAR NOT NULL [PFK]
id_product: VARCHAR NOT NULL [PFK]
quantity: INTEGER NOT NULL

Clé primaire étrangère sur « id_recipe » :

Table	Colonne	On delete	On update
Recipe	id_recipe	Cascade	No action

Clé primaire étrangère sur « id_product » :

Table	Colonne	On delete	On update
Product	id_product	Cascade	No action

- « Step »

Step
id_step: VARCHAR NOT NULL [PK]
step_description: VARCHAR NOT NULL
step_number: VARCHAR NOT NULL
id_recipe: VARCHAR NOT NULL [FK]

Clé étrangère sur « id_recipe » :

Table	Colonne	On delete	On update
Recipe	id_recipe	Cascade	No action

- « Shopping_basket »

Shopping_basket
id_shopping_basket: VARCHAR NOT NULL [PK]
id_pizza: VARCHAR [FK]
id_customer: VARCHAR NOT NULL [FK]
quantity: INTEGER NOT NULL
id_point_of_sale: VARCHAR NOT NULL [FK]
order_validate: BOOLEAN NOT NULL
id_other_product: VARCHAR [FK]

Clé étrangère sur « id_pizza » :

Table	Colonne	On delete	On update
Pizza	id_pizza	No action	No action

Clé étrangère sur « id_customer » :

Table	Colonne	On delete	On update
Customer	id_customer	Cascade	No action

Clé étrangère sur « id_point_of_sale » :

Table	Colonne	On delete	On update
Point_of_sale	id_point_of_sale	No action	No action

Clé étrangère sur « id_other_product » :

Table	Colonne	On delete	On update
Other_product	id_other_product	No action	No action

- « Customer_order »

Customer_order
id_customer_order: VARCHAR NOT NULL [PK]
current_state: VARCHAR NOT NULL
id_pizzaiolo: VARCHAR [FK]
id_delivery_man: VARCHAR [FK]
id_shopping_basket: VARCHAR NOT NULL [FK]
delivery: BOOLEAN NOT NULL

Clé étrangère sur « id_pizzaiolo » :

Table	Colonne	On delete	On update
Worker	id_pizzaiolo	No action	No action

Clé étrangère sur « id_delivery_man » :

Table	Colonne	On delete	On update
Worker	id_delivery_man	No action	No action

Clé étrangère sur « id_shopping_basket » :

Table	Colonne	On delete	On update
-------	---------	-----------	-----------

Shopping_basket	id_shopping_basket	No action	No action
------------------------	---------------------------	------------------	------------------

- « Receipt »

Receipt
id_receipt: VARCHAR NOT NULL [PK]
date_of_order: DATE NOT NULL
id_payment_method: VARCHAR NOT NULL [FK]
id_customer_order: VARCHAR NOT NULL [FK]
final_price: FLOAT NOT NULL

Clé étrangère sur « id_payment_method » :

Table	Colonne	On delete	On update
Payment_method	id_payment_method	No action	No action

Clé étrangère sur « id_customer_order » :

Table	Colonne	On delete	On update
Customer_order	id_customer_order	No action	No action

- « Payment_method »

Payment_method
id_payment_method: VARCHAR NOT NULL [PK]
type_payment_method: VARCHAR NOT NULL

Diagramme de composant.

[Cliquez ici pour l'agrandir.](#)

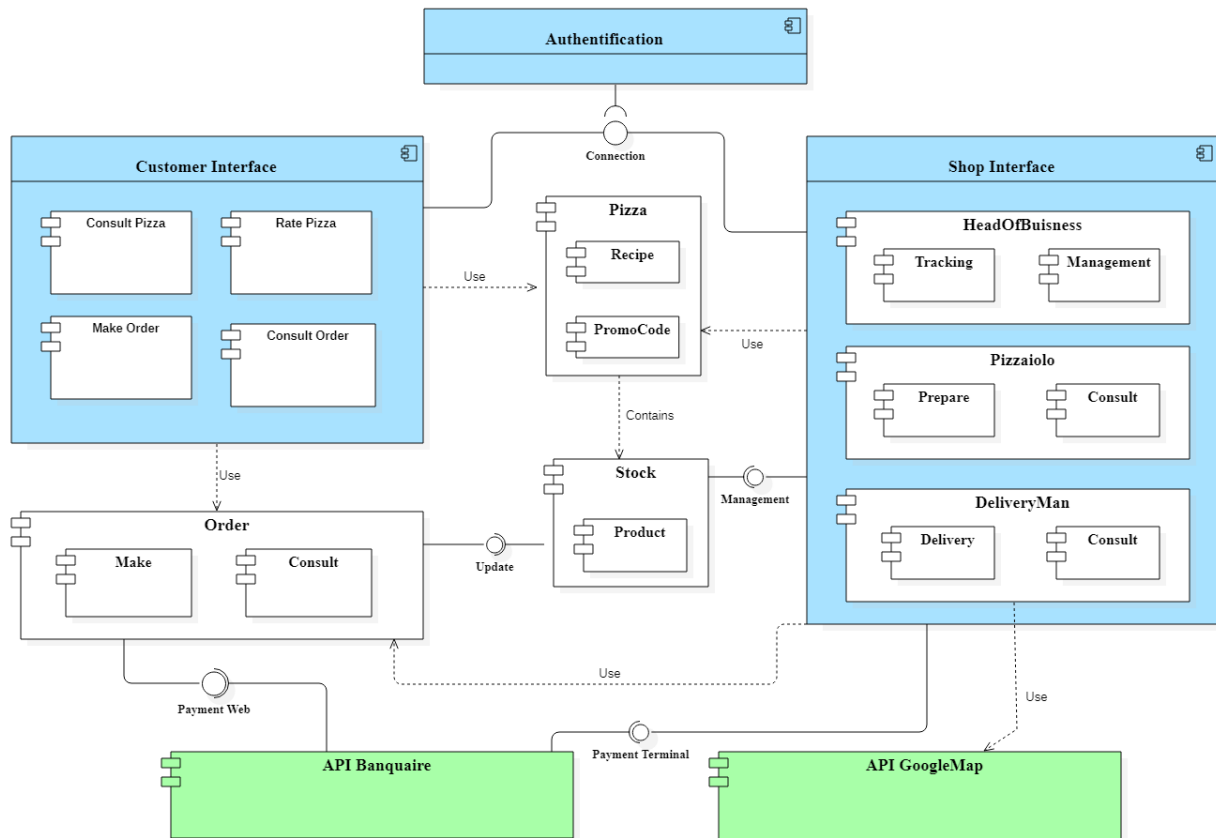


Diagramme de déploiement.

[Cliquez ici pour l'agrandir.](#)

