

# **OC PIZZA**

## **Système informatique de gestion des pizzerias**

Dossier de conception technique

Version 1.0  
01/2021

**Buglioni Thomas**

# Index

1.Présentation générale .....	2
Contexte .....	2
Besoins exprimés par le client .....	2
2.Diagramme de classe .....	4
Section : « classe » .....	4
Section : « lien » .....	6
3.Modele physique de données .....	7
4.Diagramme de composants .....	10
5.Diagramme de déploiement .....	11



## 1.Présentation générale

Mise en place d'un nouveau système informatique pour l'ensemble des pizzerias du groupe.

---

### Contexte

« OC Pizza » est un jeune groupe de pizzeria en plein essor. Créé par Franck et Lola, le groupe est 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 6 mois.

Le système informatique actuel ne correspond plus aux besoins du groupe car il ne permet pas une gestion centralisée de toutes les pizzerias.

De plus, il est très difficile pour les responsables de suivre ce qui se passe dans les points de ventes. Enfin, les livreurs ne peuvent pas indiquer « en live » que la livraison est effectuée.

---

### Besoins exprimés par le client

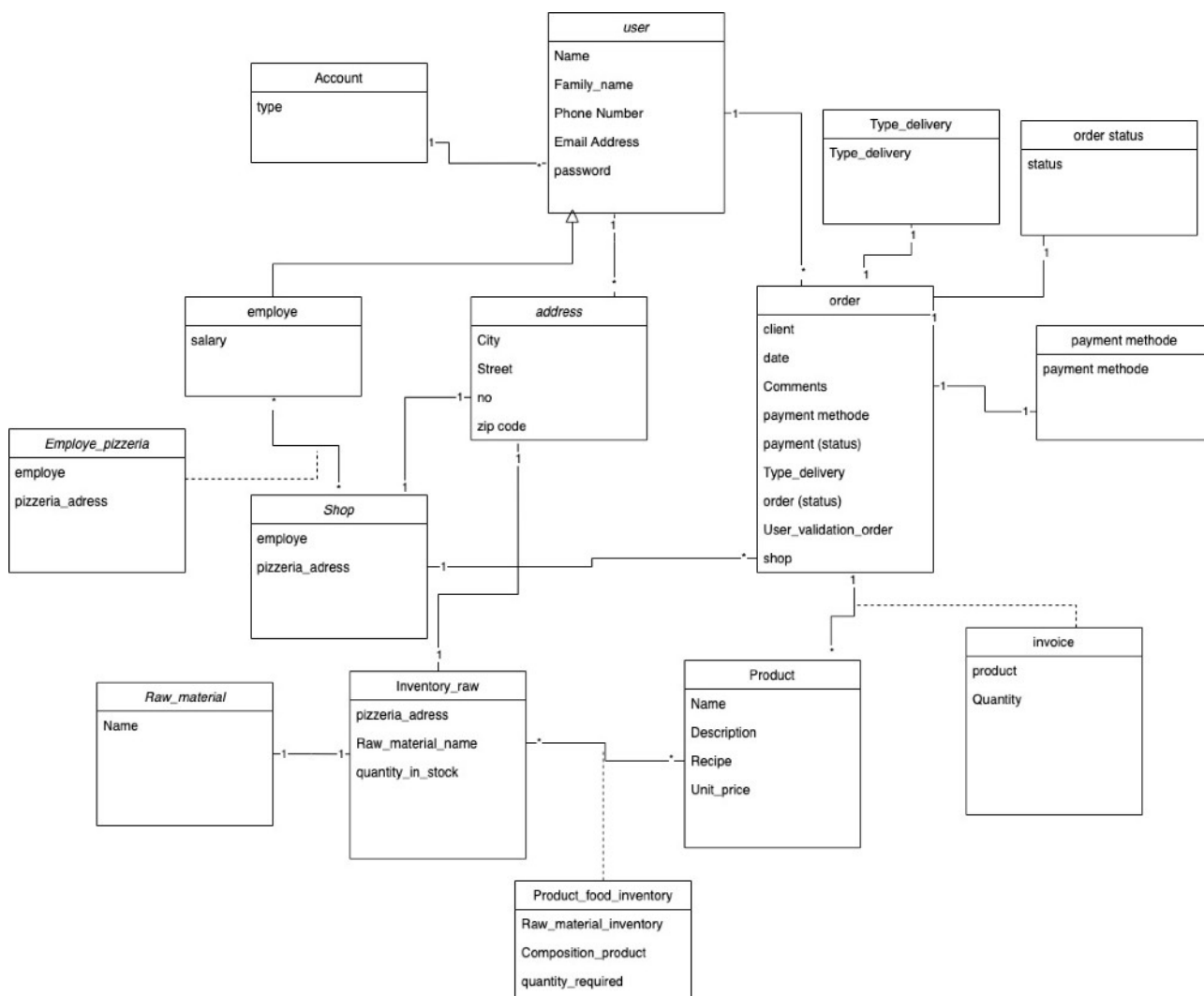
- être plus efficace dans la gestion des commandes, de leur réception à leur livraison en passant par leur préparation ;
- suivre en temps réel les commandes passées, en préparation et en livraison ;
- suivre en temps réel le stock d'ingrédients restants pour savoir quelles pizzas peuvent encore être réalisées ;
- proposer un site Internet pour que les clients puissent :
  - o passer leurs commandes, en plus de la prise de commande par téléphone ou sur place ;
  - o payer en ligne leur commande s'ils le souhaitent – sinon, ils paieront directement à la livraison ;
  - o 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

Date de livraison du système informatique :

Pour l'ouverture des 3 nouvelles pizzerias, dans 6 mois.

## 2. Diagramme de classe



### Section : « classe »

Account	Type
address	city
	no
	street
	Zip_code
employee	salary
Employe_pizzeria (table d'association)	employee
	Pizzeria_adress
inventory_raw	Pizzeria_address
	Quantity_in_stock

	Raw_material_name
Invoice (table d'association)	Product
	quantity
order	client
	comments
	date
	Order_status
	Payment_methode
	Payment_status
	shop
	Type_delivery
	user_validation_order
Order_status	status
Payment_methode	Payment_methode
Product	description
	name
	recipe
	unit_price
Product_food_inventory (table d'association)	Composition_product
	Quantity_required
	Raw_material_inventory
Raw_material	name
shop	employe
	Pizzeria_adress
Type_delivery	Type_delivery
User	Email
	family_name
	name
	password
	Phone_number

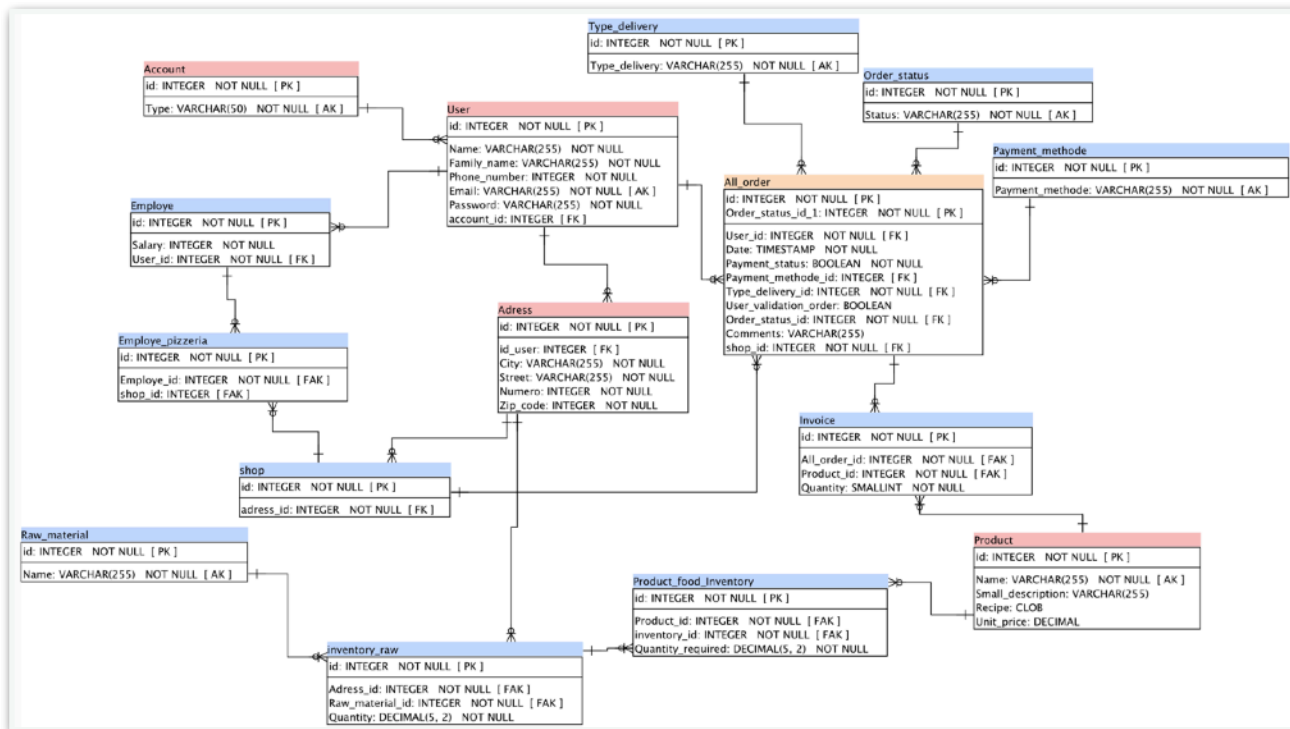
---

Section : « lien »

ADDRESS & INVENTORY_RAW	1--1
ADDRESS & SHOP	1--1
EMPLOYE	hérité de USER
EMPLOYE & SHOP	x--x EMPLOYE_PIZZERIA (table association)
INVENTORY_RAW & PRODUCT	x--x PRODUCT_FOOD_INVENTORY (table association)
INVENTORY_RAW & RAW_MATERIAL	1--1
ORDER & ORDER_STATUS	1--1
ORDER & PAYMENT_METHODE	1--1
ORDER & PRODUCT	x--x INVOICE (table association)
ORDER & SHOP	x--1
ORDER & TYPE_DELIVERY	1--1
USER & ACCOUNT	x--1
USER & ADDRESS	1--x
USER & ORDER	1--x



### 3. Modele physique de données

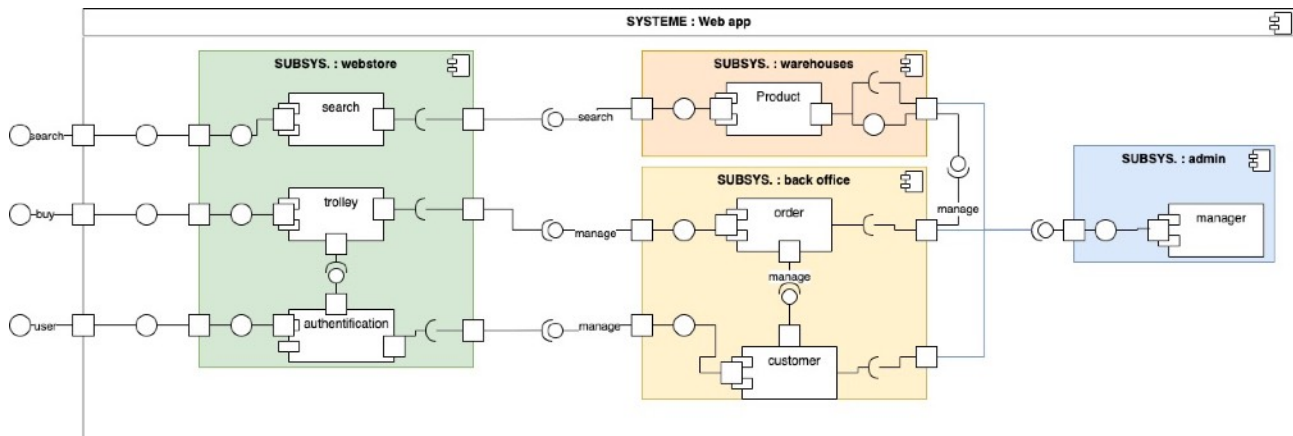


Account	Id	PK/ INTEGER / NOT NULL / AUTO	
	Type	VARCHAR(50) / NOT NULL / UNIQUE	
address	Id	PK/ INTEGER / NOT NULL / AUTO	
	id_user	INTEGER	FK user.id
	city	VARCHAR(255) / NOT NULL	
	street	VARCHAR(255) / NOT NULL	
	no	INTEGER / NOT NUL	
	Zip_code	INTEGER / NOT NUL	
employe	Id	PK/ INTEGER / NOT NULL / AUTO	
	salary	INTEGER / NOT NUL	
	user_id	INTEGER	FK user.id
Employe_pizzeria (table d'association)	Id	PK/ INTEGER / NOT NULL / AUTO	
	employe_id	INTEGER / NOT NUL	
	Shop_id	INTEGER / UNIQUE	FK shop.id
	Id	PK/ INTEGER / NOT NULL / AUTO	

inventory_raw	address_id	INTEGER / NOT NUL / UNIQUE	FK address.id
	Raw_material_id	INTEGER / NOT NUL / UNIQUE	FK raw_material.id
	Quantity	DEC(5,2) / NOT NULL	
Invoice (table d'association)	Id	PK/ INTEGER / NOT NULL / AUTO	
	all_order_id	INTEGER / NOT NUL / UNIQUE	FK all_order.id
	Product_id	INTEGER / NOT NUL / UNIQUE	FK product.id
	quantity	SMALLINT / NOT NULL	
order	Id	PK/ INTEGER / NOT NULL / AUTO	
	User_id	INTEGER / NOT NUL	FK user.id
	date	TIMESTAMP / NOT NULL	
	Payment_status	BOOL / NOT NULL	
	Payment_methode_id	INTEGER	FK payment_methode.id
	Type_delivery_id	INTEGER / NOT NUL	FK type_delivery.id
	user_validation_order	BOOL	
	Order_status_id	INTEGER / NOT NUL	FK order_status.id
	comments	VARCHAR(255)	
	shop_id	INTEGER / NOT NUL	FK shop.id
Order_status	Id	PK/ INTEGER / NOT NULL / AUTO	
	status	VARCHAR(255) / NOT NULL/ UNIQUE	
Payment_methode	Id	PK/ INTEGER / NOT NULL / AUTO	
	Payment_methode	VARCHAR(255) / NOT NULL/ UNIQUE	
Product	Id	PK/ INTEGER / NOT NULL / AUTO	
	name	VARCHAR(255) / NOT NULL/ UNIQUE	
	description	VARCHAR(255)	
	recipe	CLOB	
	unit_price	DECIMAL	
	Id	PK/ INTEGER / NOT NULL / AUTO	

Product_food_inventory (table d'association)	Product_id	INTEGER / NOT NULL / UNIQUE	FK product.id
	Inventory_id	INTEGER / NOT NULL / UNIQUE	FK inventory_raw.id
	Quantity_required	DECIMAL(5,2) / NOT NULL	
Raw_material	Id	PK/ INTEGER / NOT NULL / AUTO	
	name	VARCHAR(255) / NOT NULL/ UNIQUE	
shop	Id	PK/ INTEGER / NOT NULL / AUTO	
	address_id	INTEGER / NOT NULL	FK address.id
Type_delivery	Id	PK/ INTEGER / NOT NULL / AUTO	
	Type_delivery	VARCHAR(255) / NOT NULL/ UNIQUE	
User	Id	PK/ INTEGER / NOT NULL / AUTO	
	name	VARCHAR(255) / NOT NULL	
	family_name	VARCHAR(255) / NOT NULL	
	Phone_number	INTEGER / NOT NULL	
	Email	VARCHAR(255) / NOT NULL / UNIQUE	
	password	VARCHAR(255) / NOT NULL	
	account_id	INTEGER	FK account.id

## 4. Diagramme de composants



Description du diagramme de composants :

Localisation : système web app

Web\_store :

- search : permet la recherche de produit
- Trolley : permet le stockage de produit(s) en vue d'une commande
- Authentification : permet d'obtenir les droits nécessaires en fonction d'un status  
Back\_office.customer

Warehouse :

- Product : regroupe toute la gestion de produits (ajout / modification) et peut être modifié par  
Back\_office.order

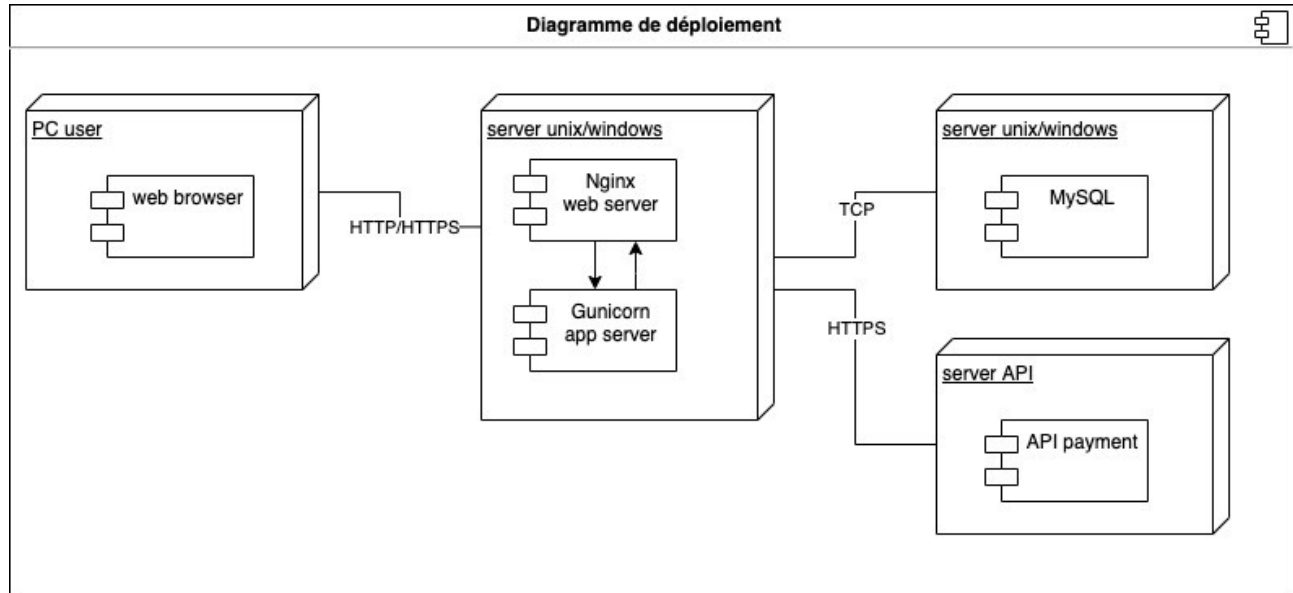
Back\_office :

- customer : gestion de la base de donnée des utilisateurs
- Order : permet la creation d'une commande et depend du status de l'utilisateur Admin.manager /  
Back\_office.customer

Admin :

- manager : permet la gestion de Warehouse.product / Back\_office.order / Back\_office.customer

## 5. Diagramme de déploiement



Description du diagramme de déploiement :

- Le système « pc user » accède au server principal (unix / windows) via HTTP/HTTPS.
- Le serveur principal gère l'application via le serveur web Nginx et le server app Gunicorn qui communiquent entre eux.
- Le serveur principal communique avec le serveur secondaire (unix/windows) afin de gérer la base de donnée mysql via TCP
- Le serveur principal communique avec le serveur secondaire API afin de gérer le système de paiement via HTTPS.