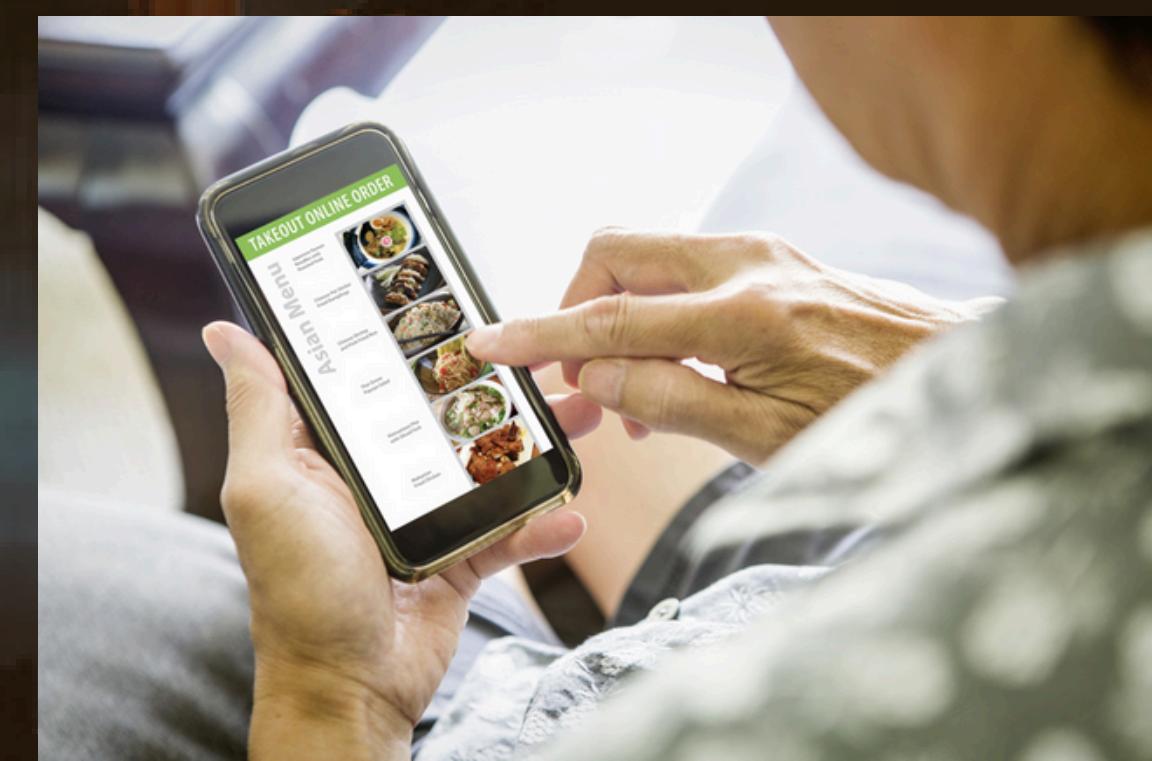


# La conception d'une application mobile pour un Restaurant

Menu Malin

P R É S E N T É P A R :

- BEN AFIA MOHAMED DHIA
- BOUASSIDA DOUAA
- MHADHBI ADAM



# **Plan :**

- Présentation d'une application mobile de restaurant
- Diagramme de flux de données
- Diagramme de cas d'utilisation
- Diagramme de classe
- Les tables de base de données
- Prototypes

# Présentation d'une application mobile de restaurant

## Introduction :

Dans un monde toujours plus connecté, l'industrie de la restauration se doit d'évoluer pour répondre aux besoins de ses clients. C'est dans cette optique que nous présentons **Menu Malin**, une application mobile révolutionnaire conçue pour simplifier votre expérience culinaire.

## Objectifs de l'Application :

Les objectifs de **Menu Malin** sont clairs : offrir une méthode de commande en ligne rapide et intuitive, permettre la personnalisation de vos plats préférés et garantir une expérience utilisateur sans faille.

## Avantages pour les Clients :

Pour vous, cher client, cela signifie pouvoir commander rapidement, personnaliser vos plats à votre guise et recevoir des notifications en temps réel vous tenant informé du statut de votre commande. **Menu Malin** met l'expérience du client au cœur de tout.





## Caractéristiques Principales :

Les fonctionnalités de **Menu Malin** comprennent un menu interactif, une commande en ligne, la personnalisation de vos plats préférés, un historique complet de vos commandes, et des notifications en temps réel vous tenant informé du processus de préparation et de livraison de vos repas.

### Commande en Ligne Sécurisée :

Avec **Menu Malin**, nous comprenons l'importance de la sécurité lors de vos transactions en ligne. C'est pourquoi toutes les transactions sont sécurisées grâce à un cryptage SSL avancé, vous garantissant une expérience de commande en ligne sans soucis.

### Personnalisation des Commandes :

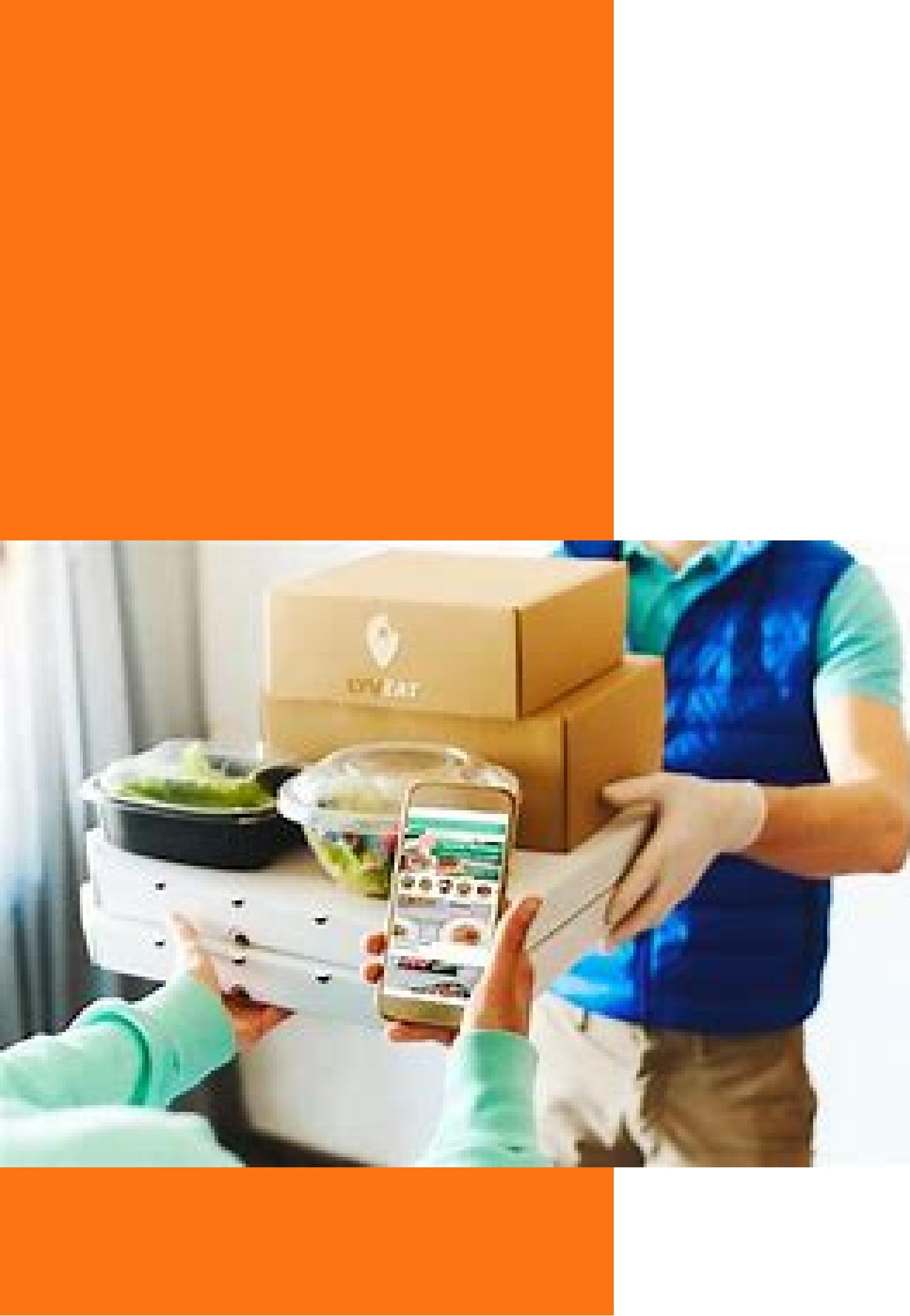
Avec **Menu Malin**, vous avez le pouvoir de personnaliser chaque plat selon vos préférences.

Ajoutez des extras, ajustez les portions - tout est entre vos mains.



### Notifications en Temps Réel :

Restez informé à chaque étape de votre commande grâce à nos notifications en temps réel. Recevez des mises à jour instantanées sur la préparation, la livraison et bien plus encore.



### Programme de Fidélité et Offres Spéciales :

En tant que membre fidèle de **Menu Malin**, vous bénéficierez de notre programme de fidélité exclusif. Gagnez des points à chaque commande et profitez d'offres spéciales réservées à nos membres.

### Témoignages et Avis :

Intégration de citations de clients satisfaits et d'étoiles ou avis positifs provenant d'app stores.  
"Découvrez ce que nos clients disent de leur expérience avec **Menu Malin**".

### Appel à l'Action (Call to Action) :

Invitation à télécharger l'application.

Exemple : "Prêt à vivre une expérience culinaire révolutionnaire ? Téléchargez **Menu Malin** dès aujourd'hui !"

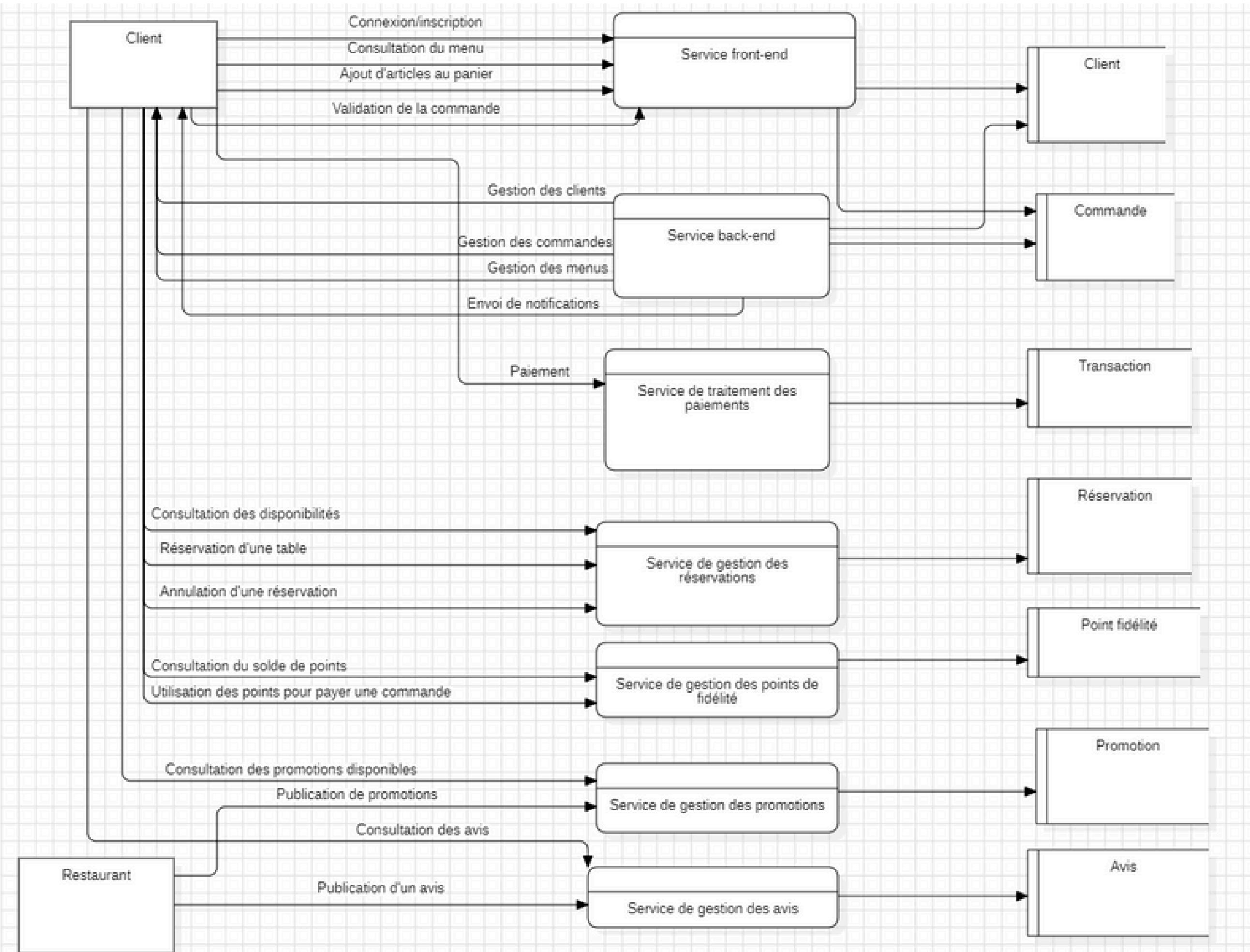
### Coordination pour une Démonstration en Direct :

Pour les présentations en personne, proposez une démonstration en direct de l'application.

### Coordonnées et Support :

Informations de contact pour obtenir un soutien ou des réponses aux questions.

# Diagramme de flux de données



## **Processus de Sélection de Restaurant :**

L'entité "Utilisateur" envoie les préférences de restaurant au processus.

Le processus sélectionne le restaurant et transmet les détails au système de commande.

## **Processus de Saisie du Numéro de Table :**

L'utilisateur entre le numéro de table dans l'interface.

Le système de commande enregistre le numéro de table pour la commande associée.

## **Processus de Gestion du Mode de Paiement :**

L'utilisateur sélectionne le mode de paiement (carte de crédit, espèces, etc.).

Le système de commande traite cette information et assure la sécurité des données.

## **Processus de Décision de Livraison :**

L'utilisateur indique s'il souhaite une livraison.

Le système de commande vérifie cette préférence et, le cas échéant, coordonne avec le service de livraison.

## **Flux de Données :**

Les données de menu, les préférences de restaurant et les détails de commande circulent entre le processus de sélection de restaurant et le système de commande.

Les informations du numéro de table sont transmises au processus de saisie du numéro de table.

Les détails de paiement et la méthode sélectionnée sont gérés dans le processus de gestion du mode de paiement.

Les informations sur la livraison, le cas échéant, sont partagées entre le processus de décision de livraison et le service de livraison.

## **Entités :**

"Utilisateur" : Initiateur de la commande.

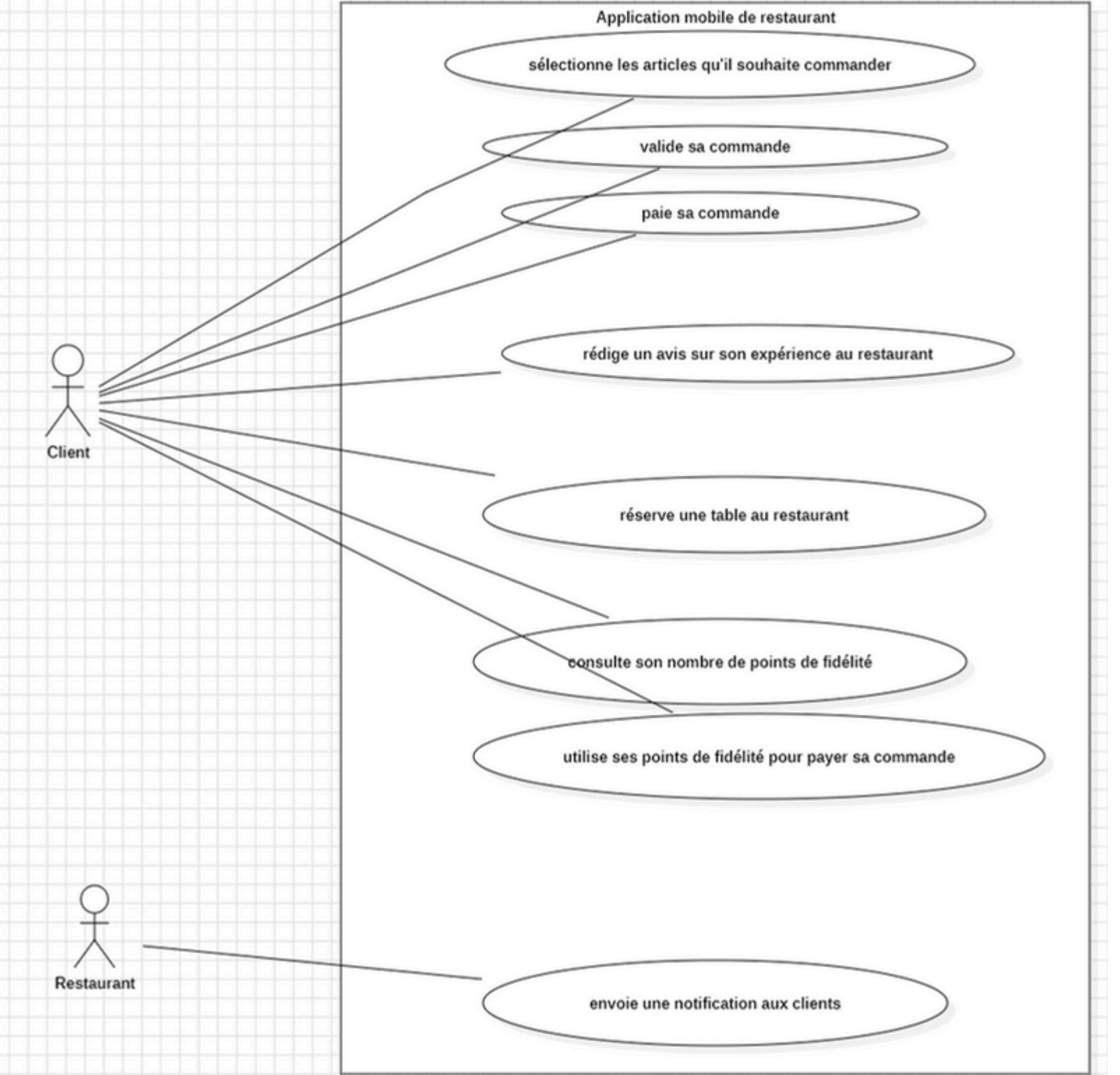
"Système de Commande" : Gère toutes les informations de commande.

"Restaurant" : Source des données de menu et lieu de traitement de la commande.

"Service de Livraison" (le cas échéant) : Partenaire pour la livraison.

Ce DFD représente le flux cohérent des données et des processus au sein de l'application de sélection de restaurant, en prenant en compte les étapes nécessaires pour la commande, la livraison et le paiement.

# Diagramme de cas d'utilisation



**Rechercher un restaurant :** Les utilisateurs peuvent rechercher des restaurants en fonction de divers critères tels que la localisation, le type de cuisine, les évaluations, etc.

**Consulter le menu :** Les clients peuvent parcourir le menu du restaurant pour voir les plats disponibles, leurs descriptions et les prix.

**Passer une commande :** Les clients peuvent sélectionner des plats, les ajouter au panier et passer une commande.

**Effectuer le paiement :** Les clients peuvent effectuer le paiement en ligne une fois leur commande sélectionnée.

**Gérer le profil utilisateur :** Les utilisateurs peuvent créer un compte, se connecter, gérer leurs informations personnelles et suivre l'historique de leurs commandes.

**Recevoir des notifications :** Les utilisateurs peuvent recevoir des notifications sur l'état de leur commande, des offres spéciales, etc.

**Donner des avis et évaluations :** Les clients peuvent donner des avis et attribuer des évaluations aux restaurants et aux plats.

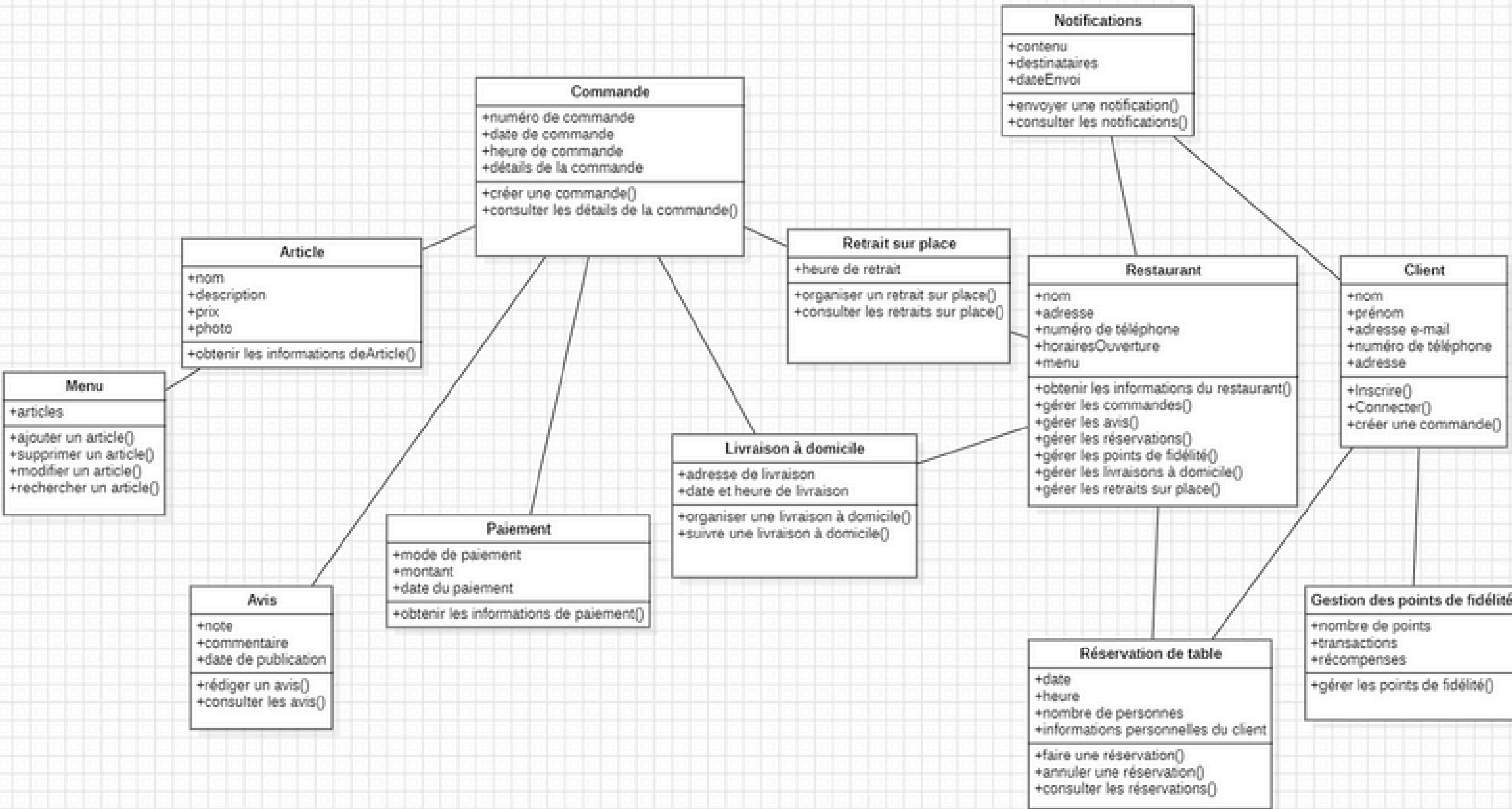
**Contacter le service client :** Les utilisateurs peuvent contacter le service client pour obtenir de l'aide ou résoudre des problèmes.

**Consulter les promotions :** Les clients peuvent consulter les promotions en cours pour bénéficier de réductions ou d'offres spéciales.

**Se connecter/déconnecter :** Les utilisateurs peuvent se connecter pour accéder à des fonctionnalités spécifiques et se déconnecter une fois qu'ils ont terminé.

Les acteurs incluent les différents utilisateurs de l'application mobile, tels que les clients, les cuisiniers, les livreurs, le service client et les administrateurs. Chaque acteur interagit avec l'application de manière spécifique en fonction de ses responsabilités et de ses besoins.

# Diagramme de classe



**Utilisateur** : Représente un utilisateur de l'application, avec des attributs tels que l'identifiant, le nom, le prénom, l'email et le mot de passe. Les opérations incluent la connexion et la déconnexion.

**Restaurant** : Représente un restaurant avec des attributs tels que l'identifiant, le nom, l'adresse et le type de cuisine. Les opérations incluent l'affichage du menu et la liste des avis.

**Commande** : Représente une commande passée par un utilisateur, avec des attributs tels que l'identifiant de commande, l'identifiant de l'utilisateur, l'identifiant du restaurant, la date de la commande et la liste des plats commandés. Les opérations incluent l'ajout de plats à la commande et la validation de la commande.

**Plat** : Représente un plat du menu d'un restaurant, avec des attributs tels que l'identifiant, le nom, la description, le prix et le type (entrée, plat principal, dessert, etc.). Les opérations incluent l'affichage des détails du plat.

Les relations entre les classes (comme l'association entre Utilisateur et Commande, Restaurant et Commande, etc.) ne sont pas représentées ici, mais elles peuvent être ajoutées pour montrer comment les classes interagissent les unes avec les autres.

# Les tables de base de données

```
1 -- Créer la table "avis"
2 CREATE TABLE avis (
3     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
4     client_id INT,
5     plat_id INT,
6     note INT NOT NULL,
7     commentaire TEXT,
8     date_avis DATETIME NOT NULL,
9     FOREIGN KEY (client_id) REFERENCES clients(id),
10    FOREIGN KEY (plat_id) REFERENCES plats(id)
11 );
12 -- insérer deux nouvelles lignes dans la table "avis"
13 INSERT INTO avis (client_id, plat_id, note, commentaire, date_avis)
14 VALUES
15 (1, 2, 5, 'Excellent dish! Highly recommend!', '2023-12-12 12:00:00'),
16 (2, 3, 4, 'Good dish, but not as flavorful as expected.', '2023-12-12 14:00:00');
17 -- afficher toutes les lignes de la table "avis"
18 SELECT *
19 FROM avis;
20
21
22 -- Créer la table "menu"
23 CREATE TABLE menu (
24     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
25     nom_menu VARCHAR(255) NOT NULL,
26     date_debut DATETIME NOT NULL,
27     date_fin DATETIME NOT NULL,
28     description TEXT,
29     image VARCHAR(255)
30 );
31 -- insérer deux nouvelles lignes dans la table "menu"
32 INSERT INTO menu (nom_menu, date_debut, date_fin, description, image)
33 VALUES
34 ('Lunch Menu', '2023-12-15', '2024-01-31', 'A selection of light and delicious dishes perfect for midday dining.', 'lunch_menu.jpg'),
35 ('Dinner Menu', '2023-12-15', '2024-01-31', 'Our signature dishes created with fresh seasonal ingredients.', 'dinner_menu.jpg');
36 -- afficher toutes les lignes de la table "menu"
37 SELECT *
38 FROM menu;
```

```
41 -- CrÃ©er la table "article"
42 CREATE TABLE article (
43     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
44     nom_article VARCHAR(255) NOT NULL,
45     description TEXT,
46     prix DECIMAL(5,2) NOT NULL,
47     image VARCHAR(255),
48     categorie_id INT,
49     menu_id INT,
50     FOREIGN KEY (categorie_id) REFERENCES categories(id),
51     FOREIGN KEY (menu_id) REFERENCES menu(id)
52 );
53 -- insÃ©rer deux nouvelles lignes dans la table "article"
54 INSERT INTO article (nom_article, description, prix, image, categorie_id, menu_id)
55 VALUES
56 ('Steak frites', 'A juicy steak served with crispy fries.', 15.99, 'steak_frites.jpg', 1, 2),
57 ('Salade composÃ©e', 'A mixed salad with fresh vegetables.', 9.99, 'salade_composee.jpg', 2, 1),
58 ('Cheesecake', 'A decadent cheesecake with a graham cracker crust.', 5.99, 'cheesecake.jpg', 3, 2);
59 -- afficher toutes les lignes de la table "article"
60 SELECT *
61 FROM article;
62
63 -- CrÃ©er la table "paiement"
64 CREATE TABLE paiement (
65     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
66     commande_id INT,
67     type_paiement VARCHAR(255) NOT NULL,
68     montant DECIMAL(10,2) NOT NULL,
69     date_paiement DATETIME NOT NULL,
70     FOREIGN KEY (commande_id) REFERENCES commandes(id)
71 );
72 -- insÃ©rer deux nouvelles lignes dans la table "paiement"
73 INSERT INTO paiement (commande_id, type_paiement, montant, date_paiement)
74 VALUES
75 (1, 'Carte bancaire', 25.98, '2023-12-10 18:07:00'),
76 (2, 'EspÃ©ces', 32.99, '2023-12-11 10:10:00'),
77 (3, 'Cheque', 40.99, '2023-12-12 12:00:00');
```

```
78 -- afficher toutes les lignes de la table "paiement"
79 SELECT *
80 FROM paiement;
81
82 -- Créer la table "livraison a domicile"
83 CREATE TABLE livraison_a_domicile (
84     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
85     commande_id INT,
86     adresse_livraison VARCHAR(500) NOT NULL,
87     code_postal VARCHAR(5) NOT NULL,
88     ville VARCHAR(255) NOT NULL,
89     frais_livraison DECIMAL(5,2) NOT NULL,
90     date_livraison DATETIME NOT NULL,
91     heure_livraison TIME NOT NULL,
92     FOREIGN KEY (commande_id) REFERENCES commandes(id)
93 );
94 -- insérer deux nouvelles lignes dans la table "livraison a domicile"
95 INSERT INTO livraison_a_domicile (commande_id, adresse_livraison, code_postal, ville, frais_livraison, date_livraison, heure_livraison)
96 VALUES
97 (1, '123 Main Street, Paris, 75000', '75000', 'Paris', 5.99, '2023-12-10 18:07:00', '19:00'),
98 (2, '456 Elm Street, Paris, 75010', '75010', 'Paris', 7.99, '2023-12-11 10:10:00', '20:00'),
99 (3, '789 Maple Street, Lyon, 69000', '69000', 'Lyon', 10.99, '2023-12-12 12:00:00', '14:00');
100 -- afficher toutes les lignes de la table "livraison a domicile"
101 SELECT *
102 FROM livraison_a_domicile;
103
104 -- Créer la table "retrait sur place"
105 CREATE TABLE retrait_sur_place (
106     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
107     commande_id INT,
108     date_retrait DATETIME NOT NULL,
109     heure_retrait TIME NOT NULL,
110     FOREIGN KEY (commande_id) REFERENCES commandes(id)
111 );
```

```
112 -- insérer deux nouvelles lignes dans la table "retrait sur place"
113 INSERT INTO retrait_sur_place (commande_id, date_retrait, heure_retrait)
114 VALUES
115 (1, '2023-12-10 18:07:00', '19:00'),
116 (2, '2023-12-11 10:10:00', '20:00'),
117 (3, '2023-12-12 12:00:00', '14:00');
118 -- afficher toutes les lignes de la table "retrait sur place"
119 SELECT *
120 FROM retrait_sur_place;
121
122 -- Créer la table "commande"
123 CREATE TABLE commande (
124     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
125     client_id INT,
126     date_commande DATETIME NOT NULL,
127     statut_commande VARCHAR(255) NOT NULL,
128     prix_total DECIMAL(10,2) NOT NULL,
129     FOREIGN KEY (client_id) REFERENCES clients(id)
130 );
131 -- insérer deux nouvelles lignes dans la table "commande"
132 INSERT INTO commande (client_id, date_commande, statut_commande, prix_total)
133 VALUES
134 (1, '2023-12-10 18:00:00', 'En cours', 25.98),
135 (2, '2023-12-11 10:00:00', 'LivrÃ©e', 32.99);
136 -- afficher toutes les lignes de la table "commande"
137 SELECT *
138 FROM commande;
```

```
141 -- Créer la table "restaurant"
142 CREATE TABLE restaurant (
143     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
144     nom_restaurant VARCHAR(255) NOT NULL,
145     adresse VARCHAR(500) NOT NULL,
146     code_postal VARCHAR(5) NOT NULL,
147     ville VARCHAR(255) NOT NULL,
148     telephone VARCHAR(20) NOT NULL,
149     site_web VARCHAR(255),
150     facebook VARCHAR(255),
151     instagram VARCHAR(255),
152     image VARCHAR(255)
153 );
154 -- insérer deux nouvelles lignes dans la table "restaurant"
155 INSERT INTO restaurant (nom_restaurant, adresse, code_postal, ville, telephone)
156 VALUES ('Le Cigalon', '12, rue de la République', '75001', 'Paris', '01 42 61 00 00'),
157 ("La Tour d'Argent", '15, quai de la Tournelle', '75005', 'Paris', '01 43 54 23 31');
158 -- afficher toutes les lignes de la table "restaurant"
159 SELECT *
160 FROM restaurant;
161
162 -- Créer la table "reservation de table"
163 CREATE TABLE reservation_de_table (
164     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
165     client_id INT,
166     date_reservation DATETIME NOT NULL,
167     heure_reservation TIME NOT NULL,
168     nombre_personnes INT NOT NULL,
169     commentaire TEXT,
170     FOREIGN KEY (client_id) REFERENCES clients(id)
171 );
172 -- insérer deux nouvelles lignes dans la table "reservation_de_table"
173 INSERT INTO reservation_de_table (client_id, date_reservation, heure_reservation, nombre_personnes, commentaire)
174 VALUES (1, '2023-12-15', '20:00', 2, ''),
175 (2, '2023-12-16', '19:00', 4, '');
176 -- afficher toutes les lignes de la table "reservation_de_table"
177 SELECT *
178 FROM reservation_de_table;
```

```
180 -- Créer la table "gestion de points de fidelite"
181 CREATE TABLE gestion_de_points_de_fidelite (
182     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
183     client_id INT,
184     nombre_points INT NOT NULL,
185     date_dernier_achat DATETIME NOT NULL,
186     FOREIGN KEY (client_id) REFERENCES clients(id)
187 );
188 -- insérer deux nouvelles lignes dans la table "gestion de points de fidelite"
189 INSERT INTO gestion_de_points_de_fidelite (client_id, nombre_points, date_dernier_achat)
190 VALUES (1, 100, '2023-12-15'),
191 (2, 50, '2023-12-16');
192 -- afficher toutes les lignes de la table "gestion de points de fidelite"
193 SELECT *
194 FROM gestion_de_points_de_fidelite;
195
196
197 -- Créer la table "client"
198 CREATE TABLE client (
199     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
200     nom VARCHAR(255) NOT NULL,
201     prenom VARCHAR(255) NOT NULL,
202     email VARCHAR(255) UNIQUE,
203     telephone VARCHAR(20) NOT NULL,
204     adresse VARCHAR(500),
205     statut_client VARCHAR(255) NOT NULL,
206     date_inscription DATETIME NOT NULL
207 );
208 -- insérer deux nouvelles lignes dans la table "client"
209 INSERT INTO client (nom, prenom, email, telephone, adresse, statut_client, date_inscription)
210 VALUES
211 ('John', 'Doe', 'john.doe@email.com', '0123456789', '1 Main Street', 'actif', '2023-12-10'),
212 ('Jane', 'Doe', 'jane.doe@email.com', '9876543210', '2 Maple Avenue', 'actif', '2023-12-10');
213
214 -- afficher toutes les lignes de la table "client"
215 SELECT *
216 FROM client;
```

```
218 -- Créer la table "notifications"
219 CREATE TABLE notifications (
220     id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
221     client_id INT,
222     type_notification VARCHAR(255) NOT NULL,
223     date_notification DATETIME NOT NULL,
224     message TEXT NOT NULL,
225     lien VARCHAR(255),
226     FOREIGN KEY (client_id) REFERENCES clients(id)
227 );
228 -- insérer deux nouvelles lignes dans la table "notifications"
229 INSERT INTO notifications (client_id, type_notification, date_notification, message, lien)
230 VALUES (1, 'Confirmation de réservation', '2023-12-10 12:00:00',
231         'Votre réservation pour 2 personnes au restaurant Le Chat Botté le 15 décembre 2023 à 20h00 est confirmée.', NULL),
232         (2, 'Mise à jour de commande', '2023-12-10 12:00:00',
233         'Votre commande pour 1 pizza Margherita et 1 salade César a été préparée et sera prête dans 30 minutes.', NULL);
234
235 -- afficher toutes les lignes de la table "notifications"
236 SELECT *
237 FROM notifications;
```

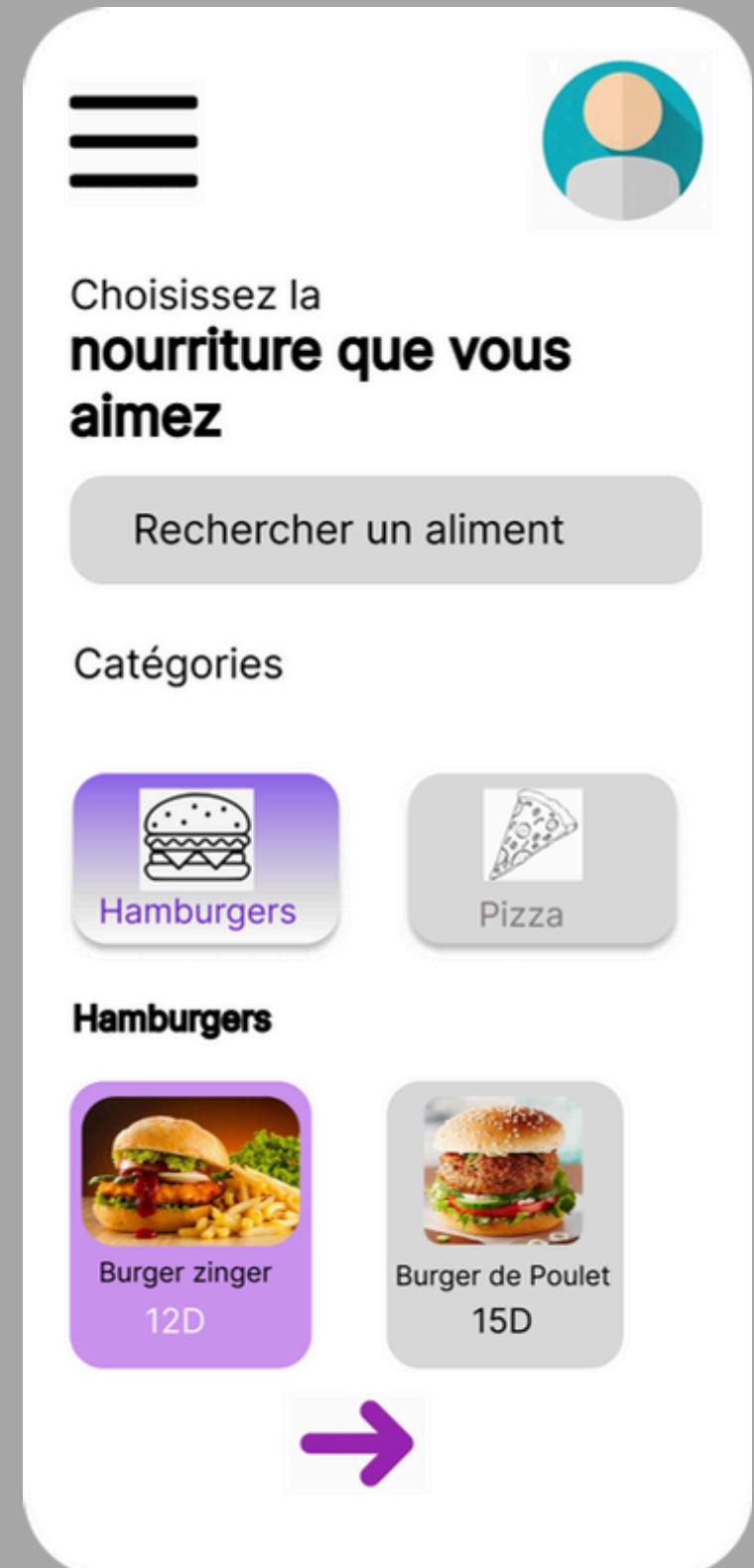
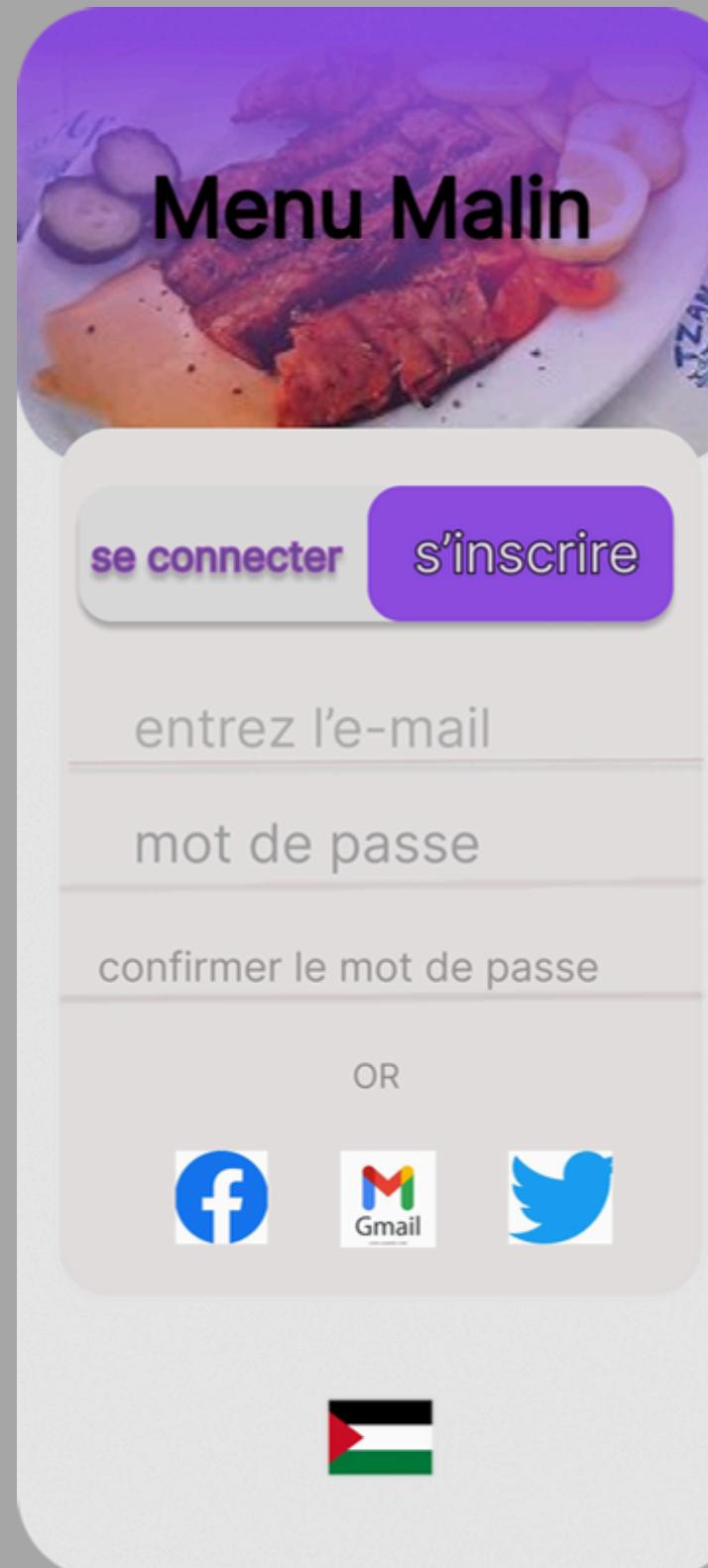
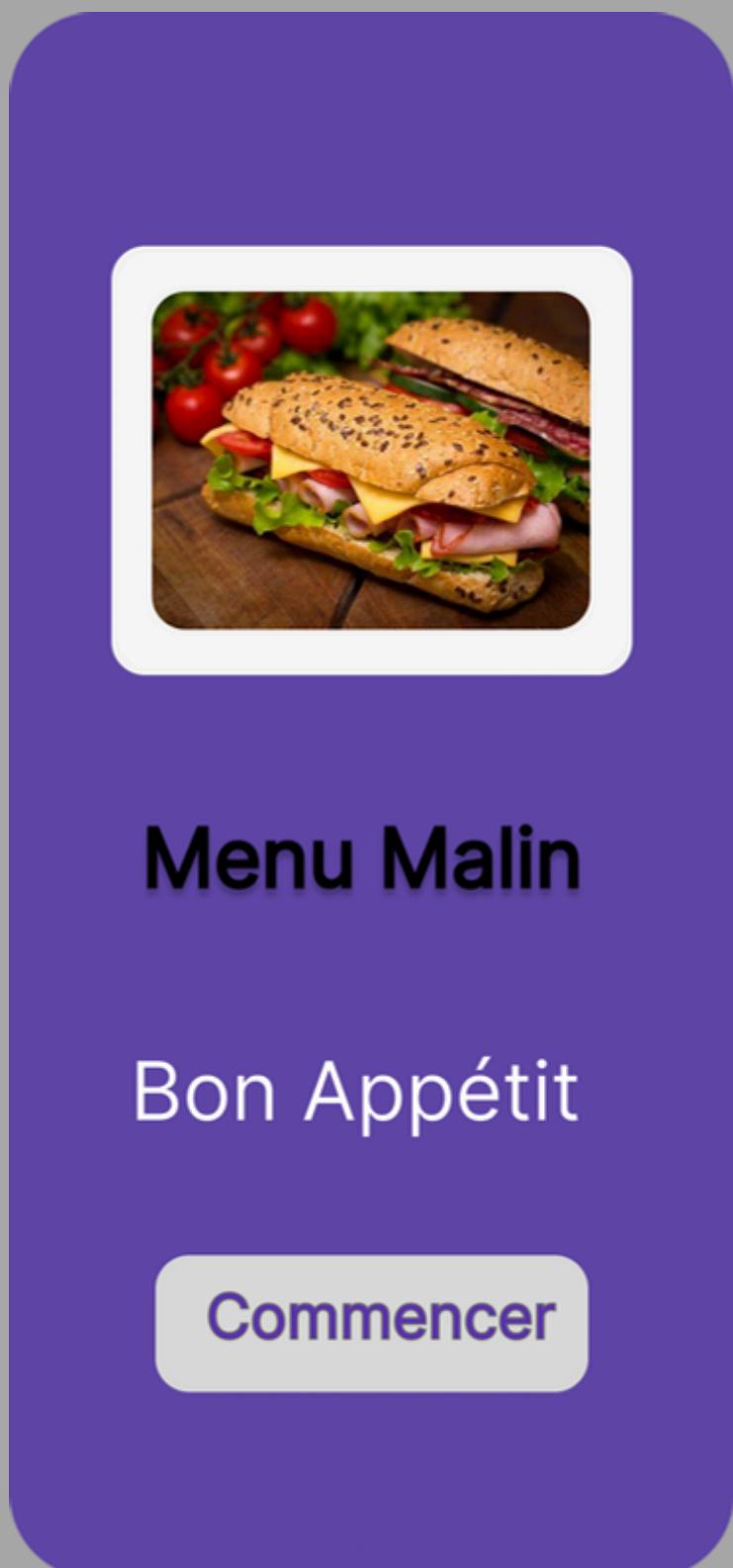
# Sortie du programme

1|1|2|5|Excellent dish! Highly recommend!|2023-12-12 12:00:00  
2|2|3|4|Good dish, but not as flavorful as expected.|2023-12-12 14:00:00  
1|Lunch Menu|2023-12-15|2024-01-31|A selection of light and delicious dishes perfect for midday dining.|lunch\_menu.jpg  
2|Dinner Menu|2023-12-15|2024-01-31|Our signature dishes created with fresh seasonal ingredients.|dinner\_menu.jpg  
1|Steak frites|A juicy steak served with crispy fries.|15.99|steak\_frites.jpg|1|2  
2|Salade composée|A mixed salad with fresh vegetables.|9.99|salade\_composée.jpg|2|1  
3|Cheesecake|A decadent cheesecake with a graham cracker crust.|5.99|cheesecake.jpg|3|2  
1|1|Carte bancaire|25.98|2023-12-10 18:07:00  
2|2|Espèces|32.99|2023-12-11 10:10:00  
3|3|Cheque|48.99|2023-12-12 12:00:00  
1|1|123 Main Street, Paris, 75000|75000|Paris|5.99|2023-12-10 18:07:00|19:00  
2|2|456 Elm Street, Paris, 75010|75010|Paris|7.99|2023-12-11 10:10:00|20:00  
3|3|789 Maple Street, Lyon, 69000|69000|Lyon|10.99|2023-12-12 12:00:00|14:00  
1|1|2023-12-10 18:07:00|19:00  
2|2|2023-12-11 10:10:00|20:00  
3|3|2023-12-12 12:00:00|14:00  
1|1|2023-12-10 18:00:00|En cours|25.98  
2|2|2023-12-11 10:00:00|Livrée|32.99  
1|Le Cigalon|12, rue de la République|75001|Paris|01 42 61 00 00||||  
2|La Tour d'Argent|15, quai de la Tournelle|75005|Paris|01 43 54 23 31||||  
1|1|2023-12-15|20:00|2|  
2|2|2023-12-16|19:00|4|  
1|1|100|2023-12-15  
2|2|50|2023-12-16  
1|John|Doe|john.doe@email.com|0123456789|1 Main Street|actif|2023-12-10  
2|Jane|Doe|jane.doe@email.com|9876543210|2 Maple Avenue|actif|2023-12-10  
1|1|Confirmation de réservation|2023-12-10 12:00:00|Votre réservation pour 2 personnes au restaurant Le Chat Botté le 15 décembre 2023 à 20h00 est confirmée.|  
2|2|Mise à jour de commande|2023-12-10 12:00:00|Votre commande pour 1 pizza Margherita et 1 salade César a été préparée et sera prête dans 30 minutes.|

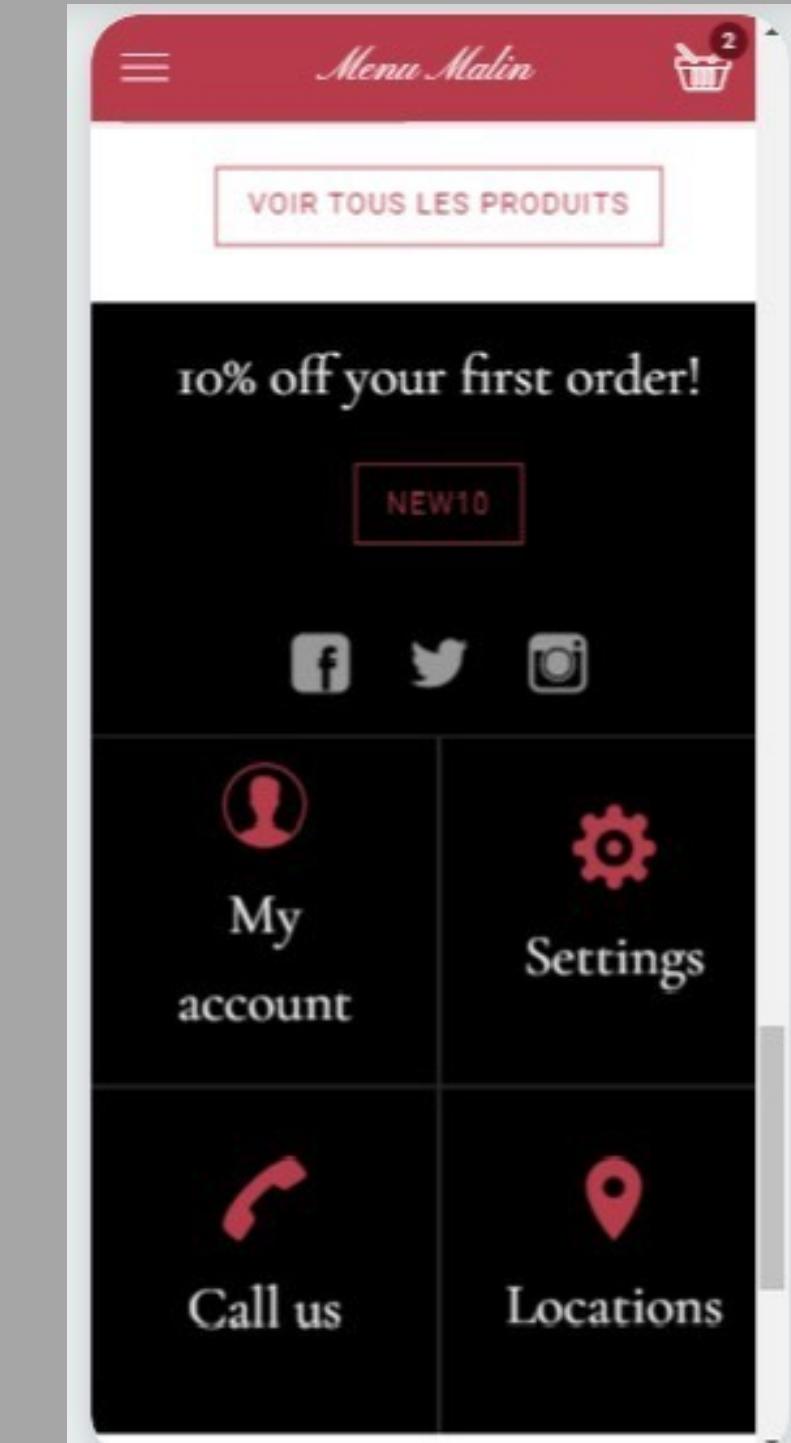
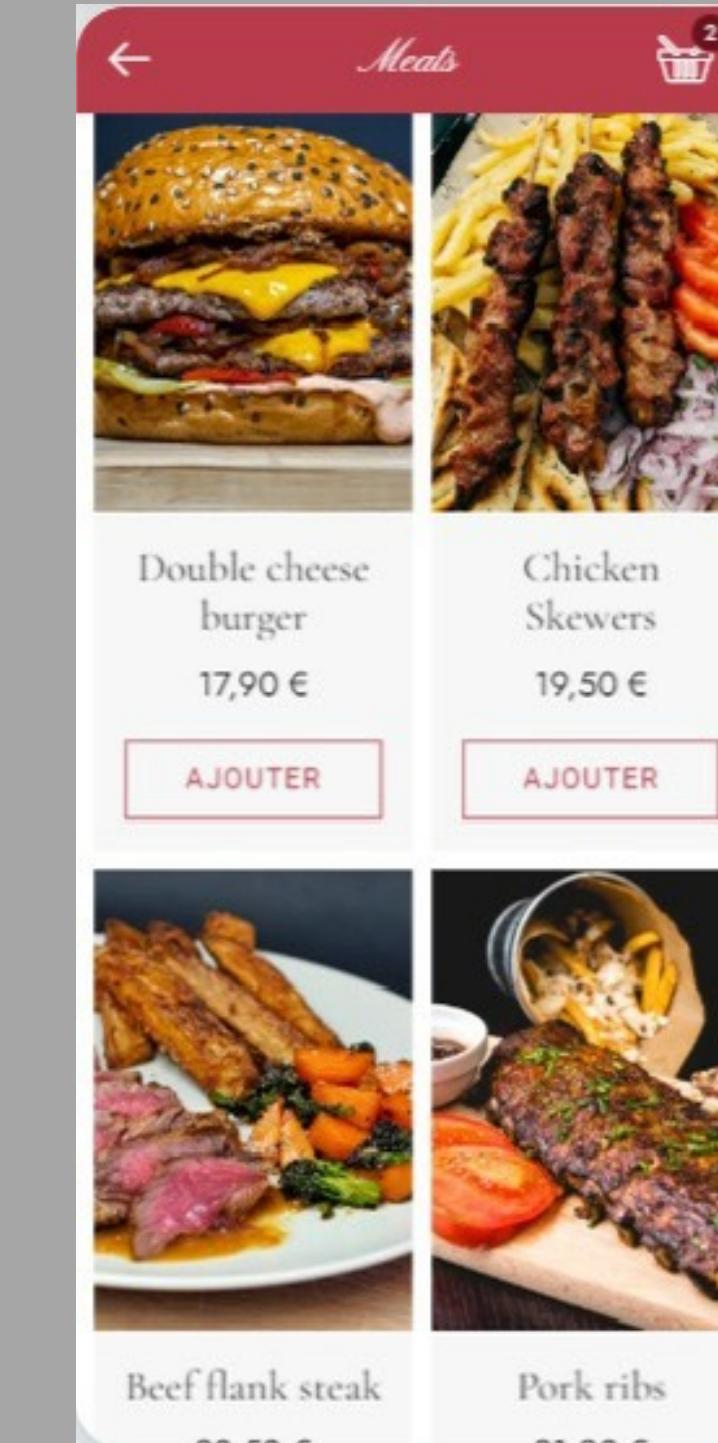
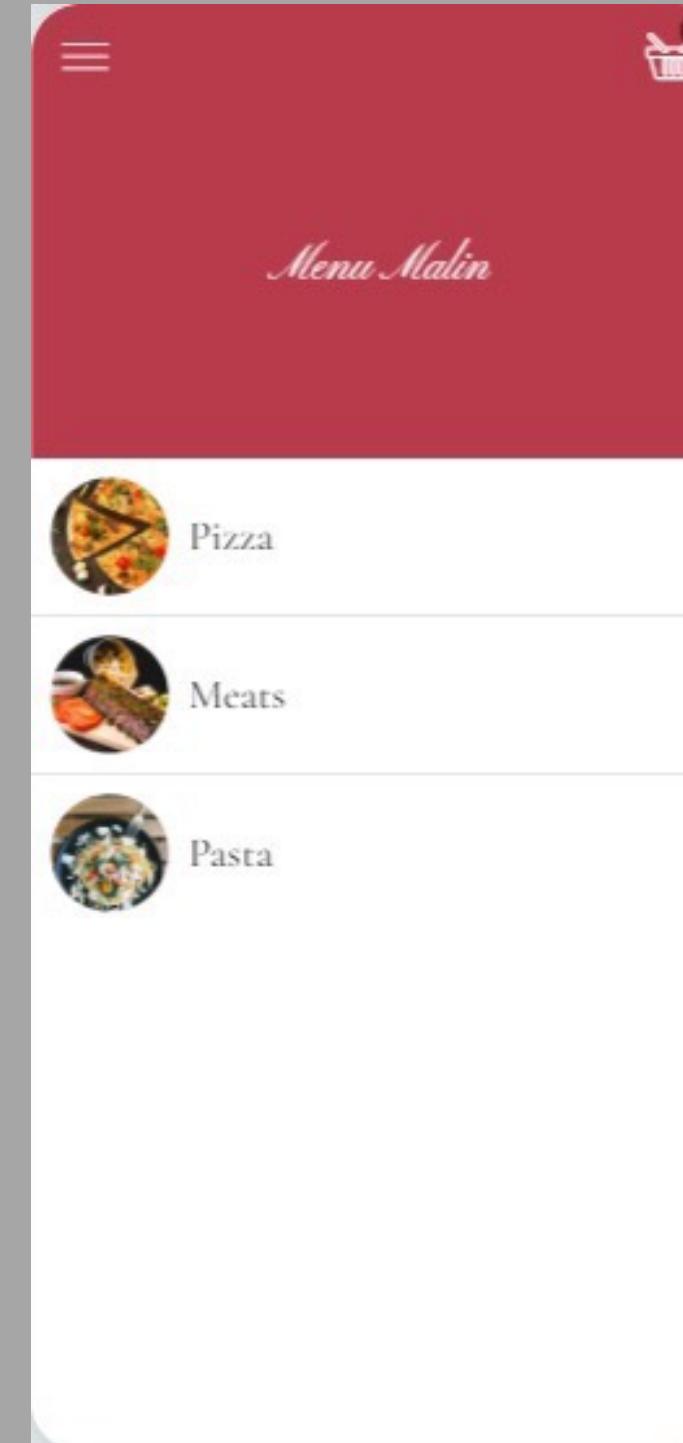
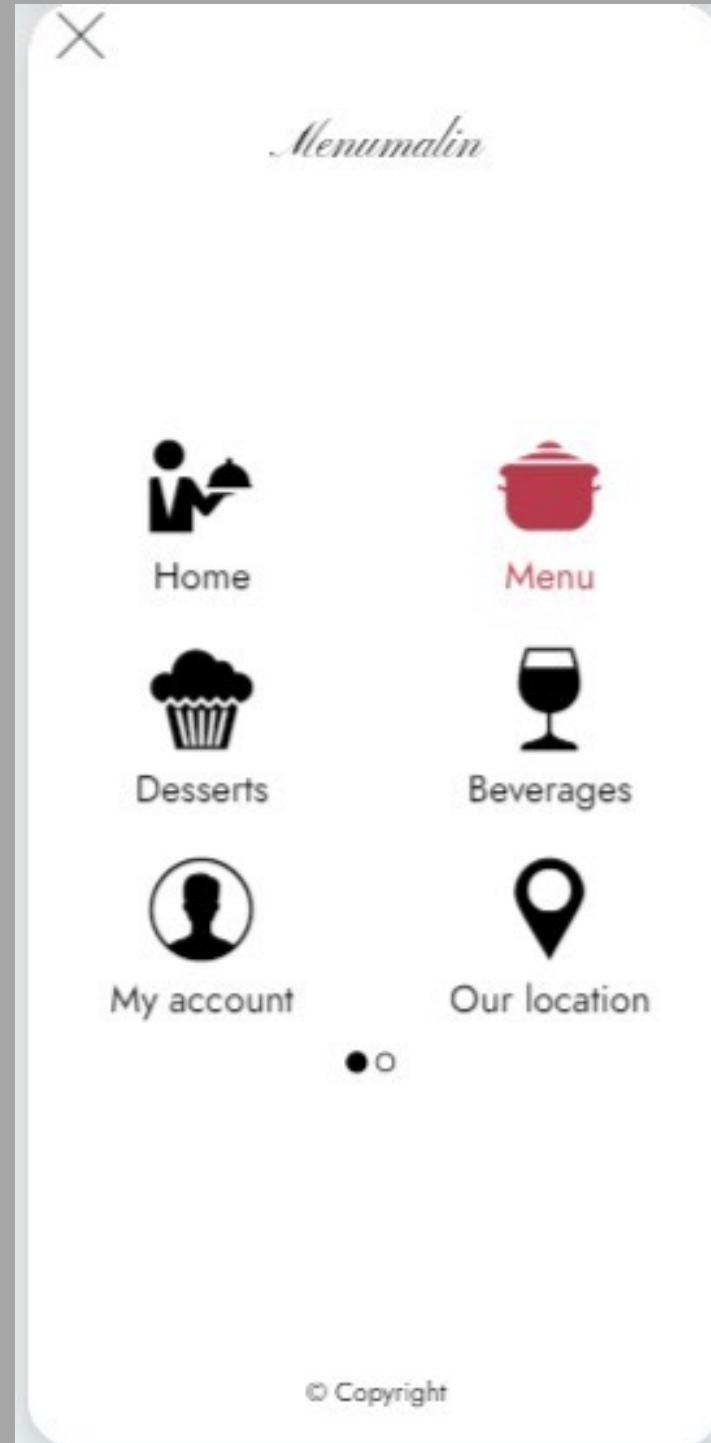
[Execution complete with exit code 0]

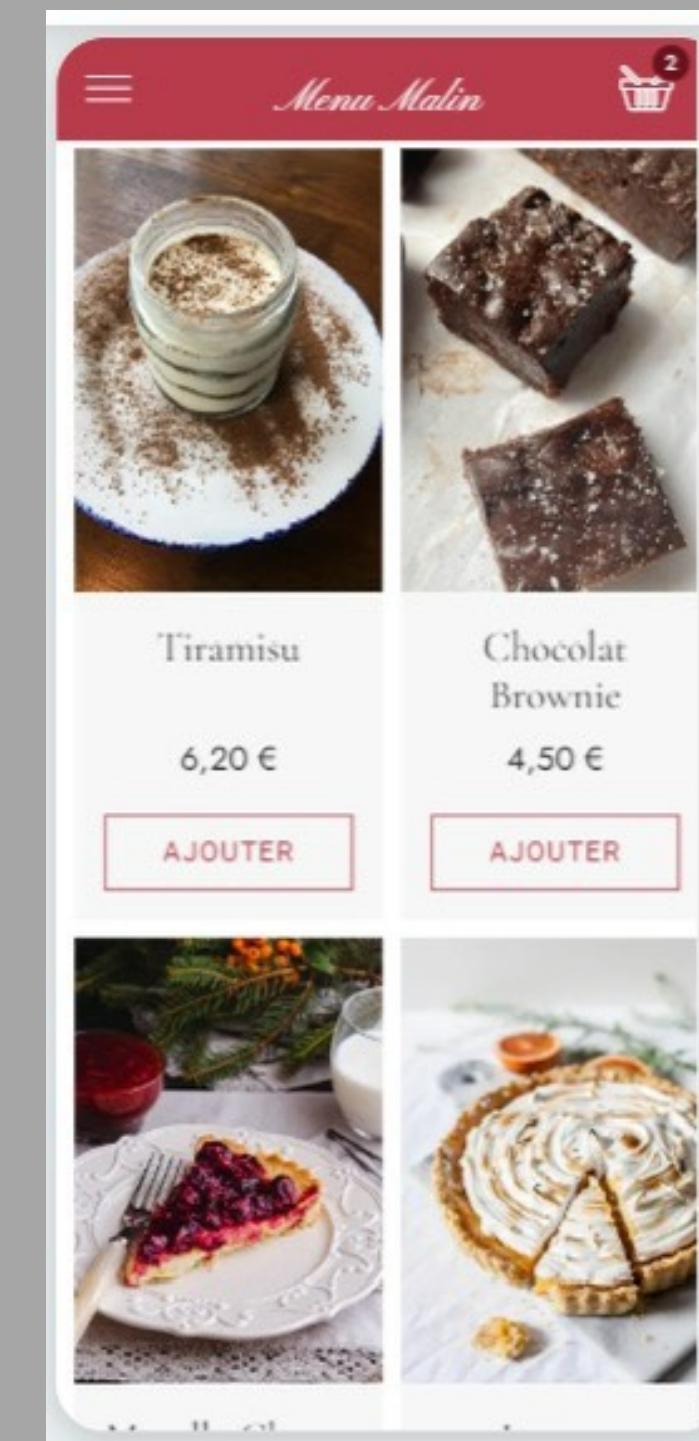
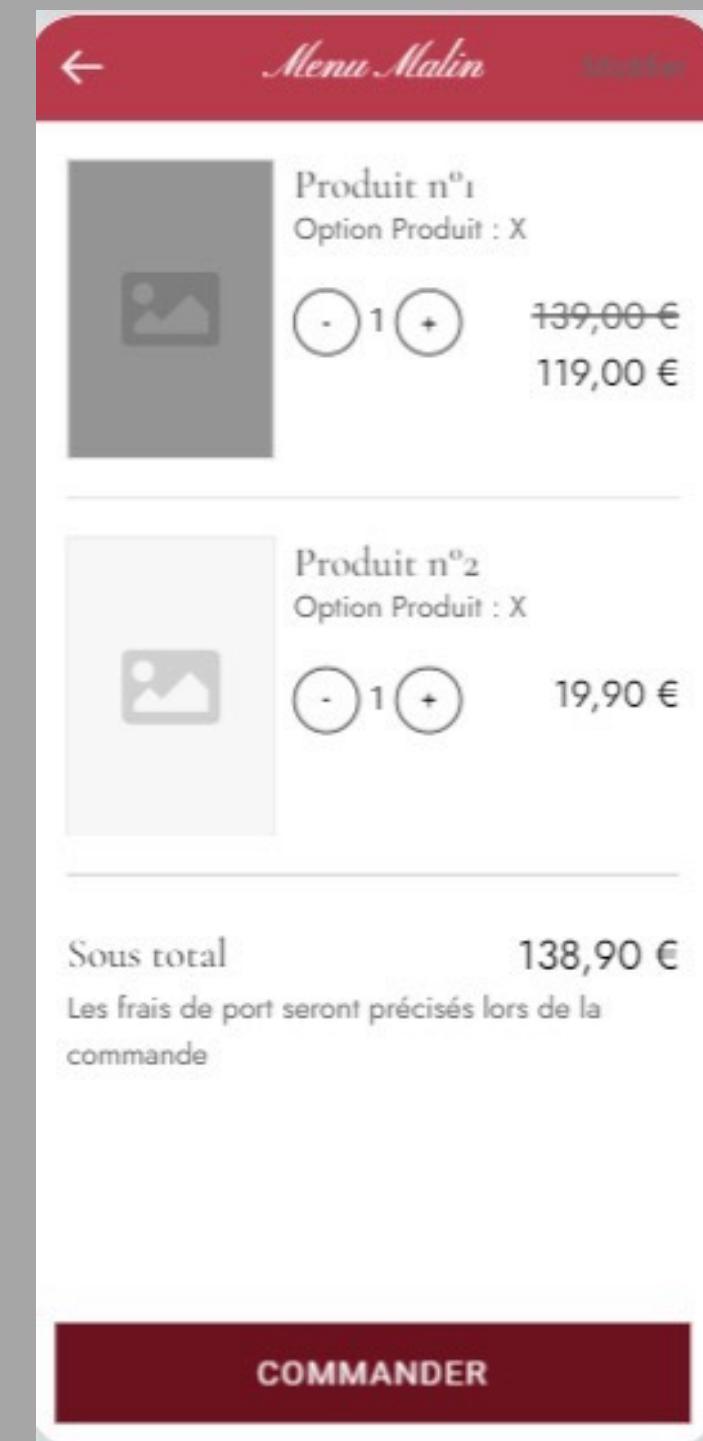
# Prototypes

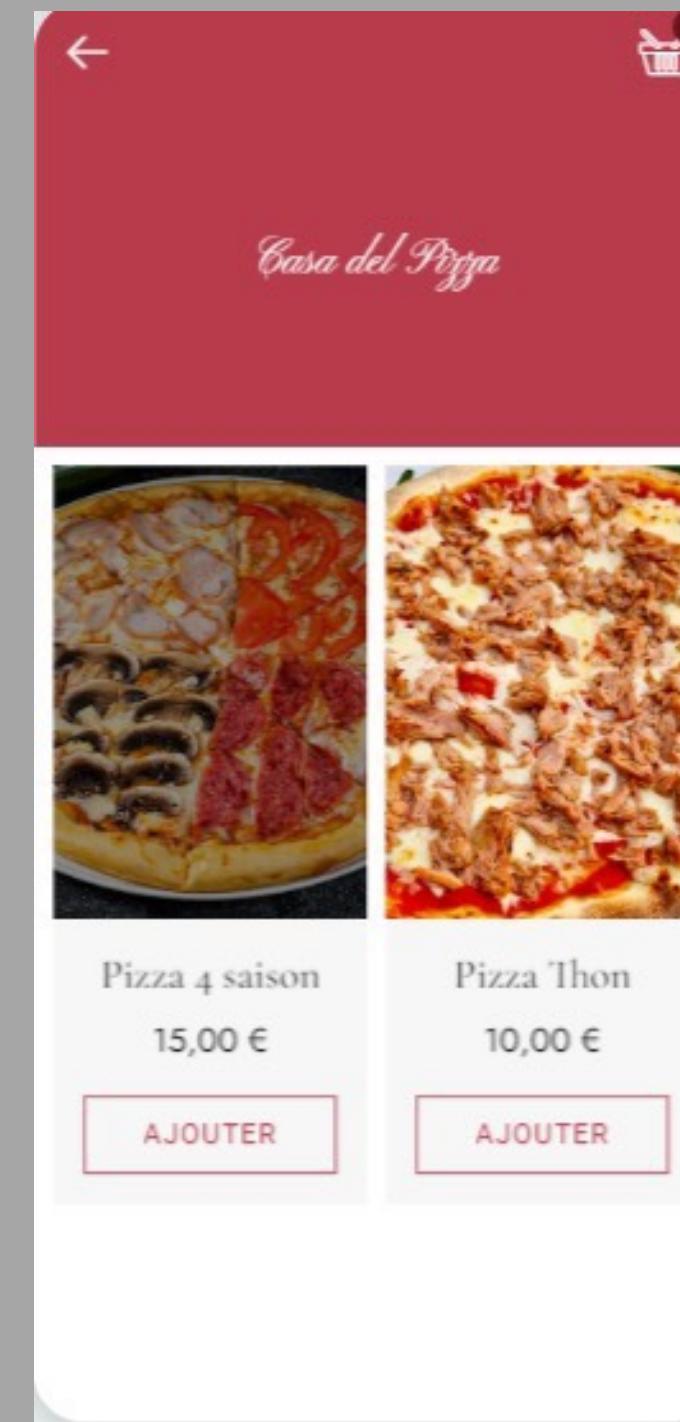
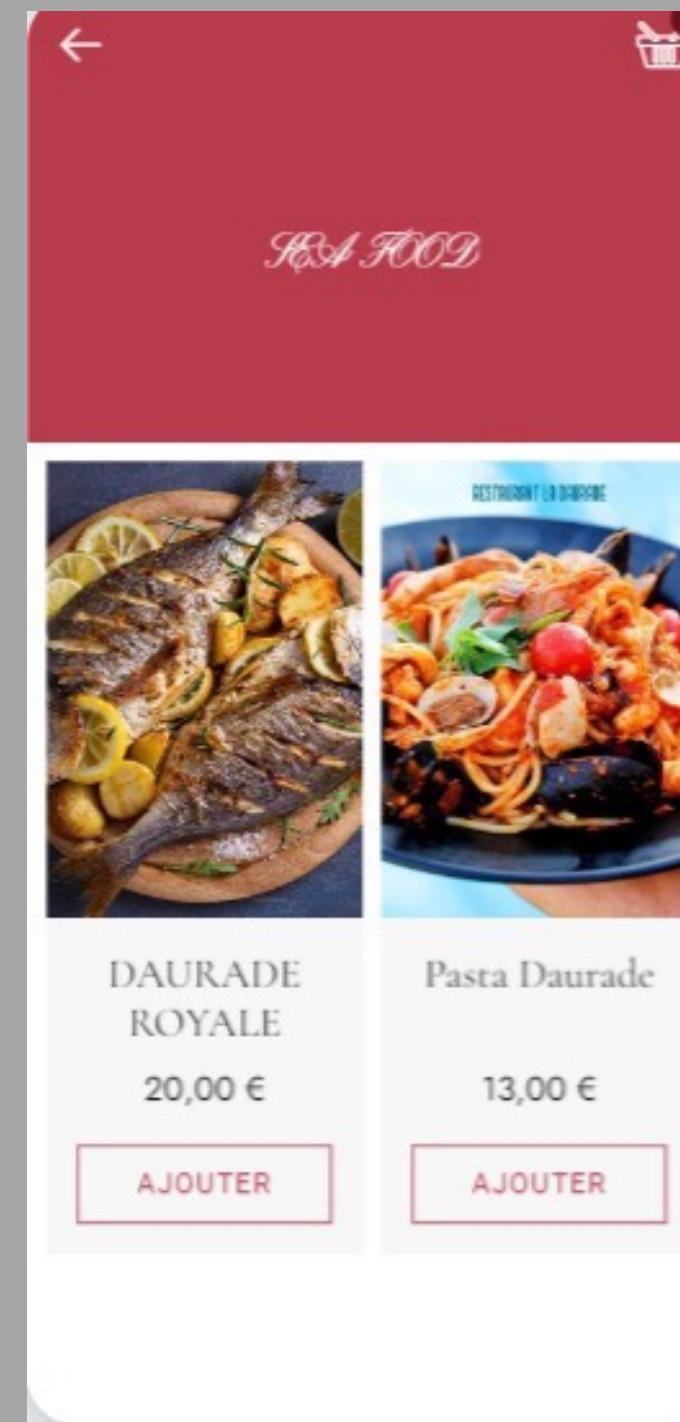
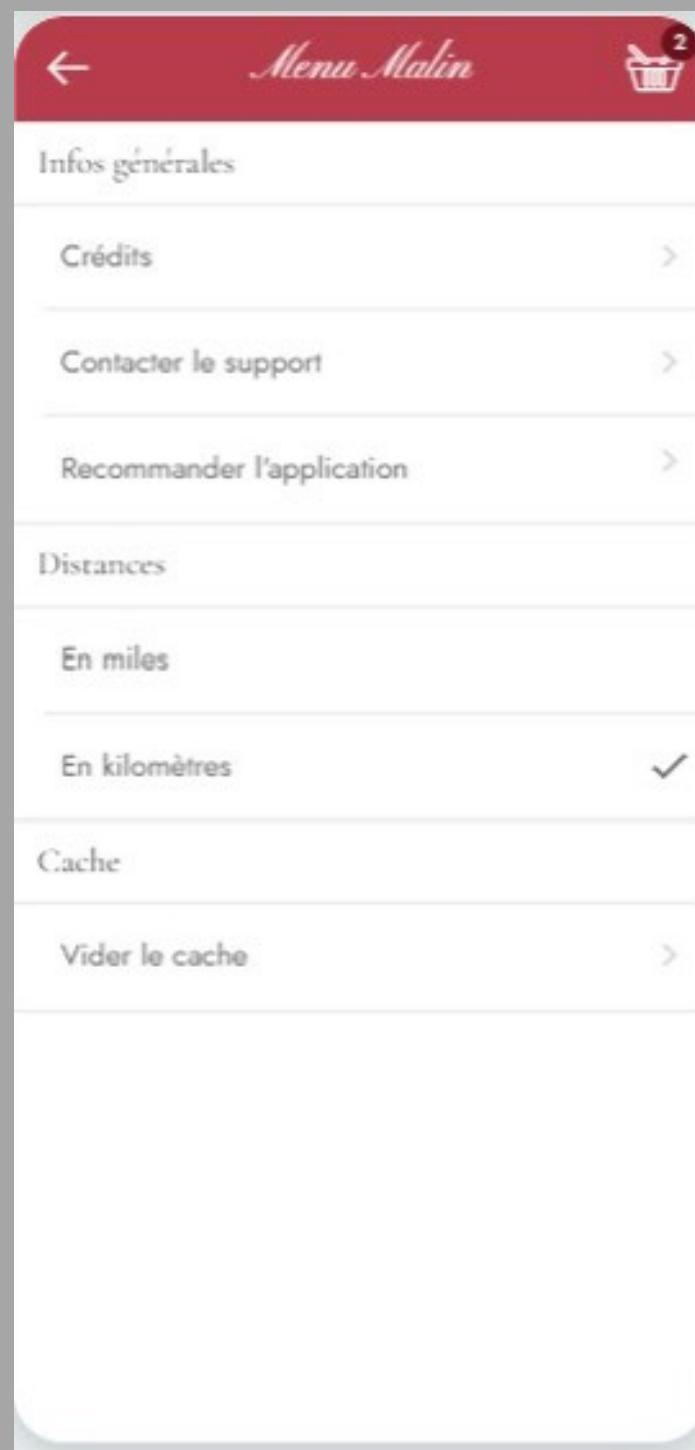
# Première proposition :



## Deuxième proposition :









Merci Pour Votre Attention

Le Bon Plat