



# Projet : Développement d'Application Informatique

Réalisé au sein de : l'Ecole Nationale des Sciences Appliquées de Fès

## Conception et réalisation d'un système de gestion de restaurant en ligne

### Réalisé par :

- Kawtar TLEMÇANI
- Wiam FAHOUL

### Encadré par :

- Pr. Abdelhak BOULAALAM

Soutenu le : 27/04/2024

Devant le jury composé de :

- Pr. EL AKKAD NABIL

Année Universitaire 2023/2024



## **Remerciement**

Nous remercions Allah le Tout-Puissant pour nous avoir accordé la force, la volonté et la patience nécessaires pour mener à bien ce projet.

Un remerciement spécial à notre encadrant, M. Abdelhak Boulaalam, pour sa guidance, ses conseils éclairés et son soutien constant tout au long du développement de ce travail. Sa présence et son expertise ont été des éléments déterminants dans la réussite de ce projet.

Nous tenons également à exprimer notre gratitude envers les membres du jury qui ont consacré leur temps et leurs connaissances pour évaluer notre travail et nous fournir des retours constructifs, contribuant ainsi à son amélioration.

Enfin, nos remerciements vont à tous les enseignants qui ont partagé leurs connaissances et leur expérience avec nous tout au long de notre parcours académique, en particulier ceux du département de Génie Informatique. Leurs enseignements ont été une source d'inspiration et ont joué un rôle crucial dans notre développement professionnel.

# Résumé

Ce rapport présente les résultats d'un projet visant à concevoir un système de gestion de restaurant en ligne, mené dans le cadre de notre formation à l'École Nationale des Sciences Appliquées de Fès. L'objectif principal de ce projet était de répondre aux besoins croissants de l'industrie de la restauration en matière de gestion efficace et sécurisée des opérations.

Pour atteindre cet objectif, notre projet s'est appuyé sur des technologies modernes telles que Express JS pour le backend, MongoDB comme système de base de données, et React pour le frontend. L'accent a été mis sur la création d'une plateforme robuste et conviviale, permettant une intégration fluide entre le frontend et le backend, une authentification sécurisée des utilisateurs grâce à l'utilisation de jetons JWT, un tableau de bord administrateur complet pour la gestion opérationnelle, un système de commande en ligne intuitif et un traitement sécurisé des paiements via l'intégration de Stripe.

En suivant un processus de développement rigoureux, nous avons conçu notre système pour offrir une expérience utilisateur optimale et une gestion efficace des opérations de restaurant en ligne. Ce rapport détaille les différentes étapes du projet, ainsi que les choix technologiques effectués et les fonctionnalités intégrées dans notre application, tout en mettant en avant les perspectives d'amélioration et les livrables attendus.

## ملخص

يعتبر هذا التقرير نتيجة مشروع يهدف إلى تطوير نظام إدارة لمطعم عبر الإنترنت، وقد تمت هذه الجهود ضمن إطار تكويننا في المدرسة الوطنية للعلوم التطبيقية بفاس، بهدف تلبية الاحتياجات المتزايدة في صناعة الخدمات الغذائية لإدارة العمليات بشكل فعال وآمن.

اعتمد مشروعنا على تقنيات حديثة مثل Express JS للجزء الخلفي، و MongoDB كنظام قاعدة بيانات، و React للواجهة الأمامية. الهدف الرئيسي هو توفير منصة قوية وسهلة الاستخدام تتيح التكامل السلس بين الجزء الأمامي والخلفي، وتوفير مصادقة آمنة للمستخدمين باستخدام رموز JWT، وتوفير لوحة تحكم إدارية كاملة لإدارة العمليات، ونظام طلبات عبر الإنترنت سهل الاستخدام، ومعالجة آمنة للمدفوعات عبر تكامل Stripe.

تم تصميم نظامنا لتوفير تجربة مستخدم مثالية وإدارة فعالة لعمليات المطعم عبر الإنترنت، وذلك من خلال اتباع عملية التطوير المعتمدة. يقدم هذا التقرير تفاصيل الخطوات المختلفة التي اتبعناها في تنفيذ هذا المشروع، بالإضافة إلى التقنيات التي تم اعتمادها والميزات التي تم دمجها في تطبيقنا.

## **Abstract**

This project aims to develop an online restaurant management system using Express JS for the backend, MongoDB as the database management system, and various frontend technologies, including React. The primary objective is to ensure seamless integration between the frontend and backend, establish a secure authentication system using JWT tokens, provide an administrative dashboard for operational management, secure payment processing through Stripe integration, host product images, offer an online ordering system, and efficiently manage all orders. The overarching goal is to create a comprehensive and secure platform facilitating restaurant operations management through an online interface.

This report outlines the detailed requirements and objectives of the project, including the integration of frontend and backend systems, secure authentication, administrative dashboard creation, product management, payment processing, online ordering system implementation, and image hosting with Firebase. Additionally, it discusses future perspectives for project enhancements and outlines the deliverables, including the complete source code and a comprehensive project documentation.

# TABLES DES MATIÈRES

INTRODUCTION GÉNÉRALE .....	9
CHAPITRE 1 : GÉNÉRALITÉS .....	11
I. Introduction.....	12
II. Contexte du projet .....	12
1. Définition du système de gestion de restaurant en ligne .....	12
2. Besoins et défis de l'industrie de la restauration.....	12
3. Problématique.....	12
4. Objectif du projet.....	13
5. Principaux enjeux : .....	13
III. Méthode de conception et démarche de développement.....	13
1. La méthode de conception.....	13
2. Processus de développement .....	13
IV. Conclusion .....	15
CHAPITRE 2 : ANALYSE DES BESOINS .....	16
I. Introduction.....	17
II. Identification des acteurs .....	17
III. Capture des besoins .....	17
IV. Besoins fonctionnels spécifiques .....	21
V. Besoins non fonctionnels.....	22
VI. Conclusion .....	23
CHAPITRE 3 : CONCEPTION.....	24
I. Introduction.....	25
II. Diagramme de Classe .....	25
III. Diagramme de séquence.....	27
IV. Diagramme d'activité .....	29
V. Diagramme de déploiement.....	30
VI. Conclusion .....	32
CHAPITRE 4 : RÉALISATION .....	33
I. Introduction.....	34
II. Méthodologie de Développement.....	34

1. Méthodologie Agile.....	34
2. Diagramme de Gantt .....	35
III. Technologies Utilisées .....	36
1.Langages de Programmation et Frameworks .....	36
IV. Description de notre application .....	39
1. Page d'accueil : .....	39
2. Les pages d'authentification et d'inscription : .....	43
3. Page de produits .....	45
4. Page du panier .....	47
5. Interface de l'administrateur .....	49
V. Conclusion.....	52
CONCLUSION GÉNÉRALE.....	53
BIBLIOGRAPHIE .....	53



# TABLE DES FIGURES

Figure 1 Le processus de développement décrit est appelé "2TUP" (Two Tracks Unified Process) .....	15
Figure 2 le diagramme de cas d'utilisation relatif au client .....	19
Figure 3 le diagramme de cas d'utilisation relatif à l'administrateur .....	20
Figure 4 le diagramme de cas d'utilisation de l'application .....	21
Figure 5 : Diagramme de classe.....	26
Figure 6 : Diagramme de séquence.....	28
Figure 7: Diagramme d'activité.....	29
Figure 8: Diagramme de deployment.....	31
Figure 9: Page d'Accueil .....	41
Figure 10: Page d'accueil en mode sombre, en français .....	42
Figure 11: Page Login.....	43
Figure 12: se connecter avec "google".....	43
Figure 13: Page Signup.....	44
Figure 14: Navbar .....	44
Figure 15: Modifier le profile .....	45
Figure 16: Page du menu .....	46
Figure 17: Ajouter un produit au panier .....	47
Figure 18: Le produit existe déjà dans le panier .....	47
Figure 19: Page du panier .....	48
Figure 20: Page du paiement.....	48
Figure 21: Suivre les commandes .....	49
Figure 22: Dashboard.....	49
Figure 23: Gestion des commandes .....	50
Figure 24: Ajouter un produit .....	50
Figure 25: Gestion des produits .....	51
Figure 26: Gestion des utilisateurs.....	51

# INTRODUCTION GÉNÉRALE

Dans un paysage commercial en constante évolution où la compétitivité est une exigence quotidienne pour les entreprises, l'industrie de la restauration est confrontée à une série de défis sans précédent. Les attentes des clients évoluent à un rythme effréné, poussées par les progrès technologiques qui révolutionnent les modèles d'affaires traditionnels. Parallèlement, la concurrence sur le marché de la restauration est devenue plus intense que jamais, avec de nouveaux acteurs entrant en scène et des innovations disruptives émergeant constamment.

Face à ce paysage en mutation rapide, notre projet de gestion de restaurant en ligne émerge comme une réponse innovante et opportune à ces défis. Nous nous sommes engagés à créer une plateforme intégrée qui répond de manière flexible et efficace aux besoins complexes des restaurants modernes, tout en offrant une expérience utilisateur exceptionnelle.

Notre vision dépasse la simple automatisation des processus existants. Nous aspirons à transformer fondamentalement la manière dont les restaurants interagissent avec leurs clients, gèrent leurs opérations et évoluent dans un environnement commercial concurrentiel. Pour concrétiser cette vision, nous nous appuyons sur les dernières avancées technologiques disponibles dans le domaine du développement logiciel.

En utilisant des technologies de pointe telles qu'Express JS pour le backend, MongoDB comme base de données et React pour le frontend, nous visons à créer une architecture logicielle robuste et évolutive. Cette approche nous permet de garantir une intégration harmonieuse entre les différentes composantes du système, offrant ainsi une expérience utilisateur fluide et intuitive.

En outre, notre projet s'appuie sur des pratiques de développement agiles et itératives, nous permettant de rester flexibles et réactifs aux besoins changeants du marché. Nous adoptons une approche centrée sur l'utilisateur, en mettant l'accent sur la conception d'interfaces conviviales et d'expériences utilisateur engageantes.

Notre engagement envers l'innovation se reflète dans les fonctionnalités que nous proposons. Outre les fonctionnalités de base telles que la gestion des réservations et des commandes en ligne, nous intégrons des solutions avancées telles qu'un système d'authentification sécurisé avec JWT, un tableau de bord administrateur intuitif et des outils d'analyse de données avancés. Afin de mener à bien ce projet, nous avons décidé de le diviser en quatre chapitres :

- Chapitre 1 : Contexte général du projet.
- Chapitre 2 : La conduite du projet.
- Chapitre 3 : Conception et modélisation.
- Chapitre 4 : Présentation de l'application.

Ces chapitres permettent d'aborder de manière détaillée les différentes phases de développement de notre solution, depuis sa conception jusqu'à sa mise en œuvre pratique.

Ce rapport détaillera notre approche méthodologique, nos choix technologiques, ainsi que les différentes étapes de développement de notre solution. Nous sommes convaincus que notre projet jouera un rôle clé dans la transformation de l'industrie de la restauration en ligne, offrant aux restaurants les outils nécessaires pour prospérer dans un environnement commercial en constante évolution.

# CHAPITRE 1 : GÉNÉRALITÉS

## **I. Introduction**

Ce premier chapitre marque le début de notre rapport sur le système de gestion de restaurant en ligne. Nous présentons les bases du projet en définissant son contexte, ses objectifs, les principaux défis rencontrés dans l'industrie de la restauration, ainsi que la problématique à laquelle notre projet cherche à répondre.

## **II. Contexte du projet**

### **1. Définition du système de gestion de restaurant en ligne**

Le système de gestion de restaurant en ligne est une solution numérique qui vise à simplifier et à optimiser les opérations quotidiennes d'un restaurant. Il comprend la prise de commandes en ligne, la gestion des réservations, la gestion des menus, la gestion des stocks, ainsi que la facilitation des paiements et des transactions.

### **2. Besoins et défis de l'industrie de la restauration**

L'industrie de la restauration est confrontée à plusieurs défis, notamment la concurrence accrue, les attentes élevées des clients en matière de service et d'expérience, ainsi que la nécessité d'optimiser les processus internes pour garantir une rentabilité durable. Un système de gestion de restaurant en ligne peut aider les établissements à relever ces défis en offrant des solutions innovantes pour améliorer leur efficacité opérationnelle et leur service client.

### **3. Problématique**

Face à l'évolution rapide des habitudes de consommation et à la demande croissante pour des solutions de restauration en ligne, comment concevoir et développer un système de gestion de restaurant qui soit à la fois efficace, sécurisé et convivial pour les utilisateurs ?

## **4. Objectif du projet**

Notre objectif principal est de développer un système de gestion de restaurant en ligne complet et fonctionnel, répondant aux besoins spécifiques de l'industrie de la restauration. Nous cherchons à fournir une plateforme intuitive et conviviale pour les clients, tout en offrant des outils puissants de gestion et d'analyse pour les propriétaires et les gestionnaires de restaurants.

## **5. Principaux enjeux :**

Les principaux enjeux auxquels notre projet cherche à répondre incluent la sécurité des transactions en ligne, l'intégration transparente entre le frontend et le backend, la convivialité de l'interface utilisateur, ainsi que la gestion efficace des commandes et des stocks.

# **III. Méthode de conception et démarche de développement**

## **1. La méthode de conception**

Nous adopterons une approche basée sur la modélisation pour concevoir notre système de gestion de restaurant en ligne. Nous utiliserons des outils de modélisation tels que UML pour représenter les différents aspects de notre système de manière claire et précise.

## **2. Processus de développement**

Pour assurer un bon démarrage et un déroulement efficace de notre projet, nous avons opté pour la méthode de développement 2TUP (Two Tracks Unified Process), un processus unifié qui gère la complexité technologique en intégrant pleinement la technologie dans toutes les phases de développement. Ce processus commence par une étude préliminaire qui consiste à identifier les acteurs impliqués dans le système à construire, à définir les interactions entre ces acteurs et le système, à produire le cahier des charges et à modéliser le contexte. Le terme "2 Tracks" signifie littéralement que le processus suit deux chemins distincts : le chemin fonctionnel et le chemin d'architecture technique.

**La branche fonctionnelle comprend :**

- La capture des besoins fonctionnels, qui aboutit à un modèle des besoins centré sur les utilisateurs et leurs activités métier.
- L'analyse, qui consiste à étudier en détail les spécifications fonctionnelles pour comprendre précisément ce que le système doit accomplir en termes d'activités métier. Les résultats de cette analyse ne dépendent pas d'une technologie particulière.

**La branche architecture technique comprend :**

- La capture des besoins techniques, qui répertorie toutes les contraintes et les choix influençant la conception du système. Cela inclut la sélection des outils et des technologies ainsi que la prise en compte des contraintes d'intégration avec les systèmes existants.
- La conception générique, qui définit les composants nécessaires à la construction de l'architecture technique du système. Cette conception est indépendante des aspects fonctionnels du système.

**La branche centrale comprend :**

- La conception préliminaire, une étape critique qui intègre le modèle d'analyse dans l'architecture technique pour cartographier les composants du système à développer.
- La conception détaillée, qui étudie ensuite comment réaliser chaque composant de manière précise.
- L'étape de codage, où les composants sont effectivement développés et testés au fur et à mesure que le code est écrit.
- L'étape de recette, qui consiste à valider que les fonctions du système développé répondent aux exigences spécifiées.

Ce processus garantit une approche structurée et itérative du développement, en assurant que les aspects fonctionnels et techniques du système sont pris en compte à chaque étape du processus.

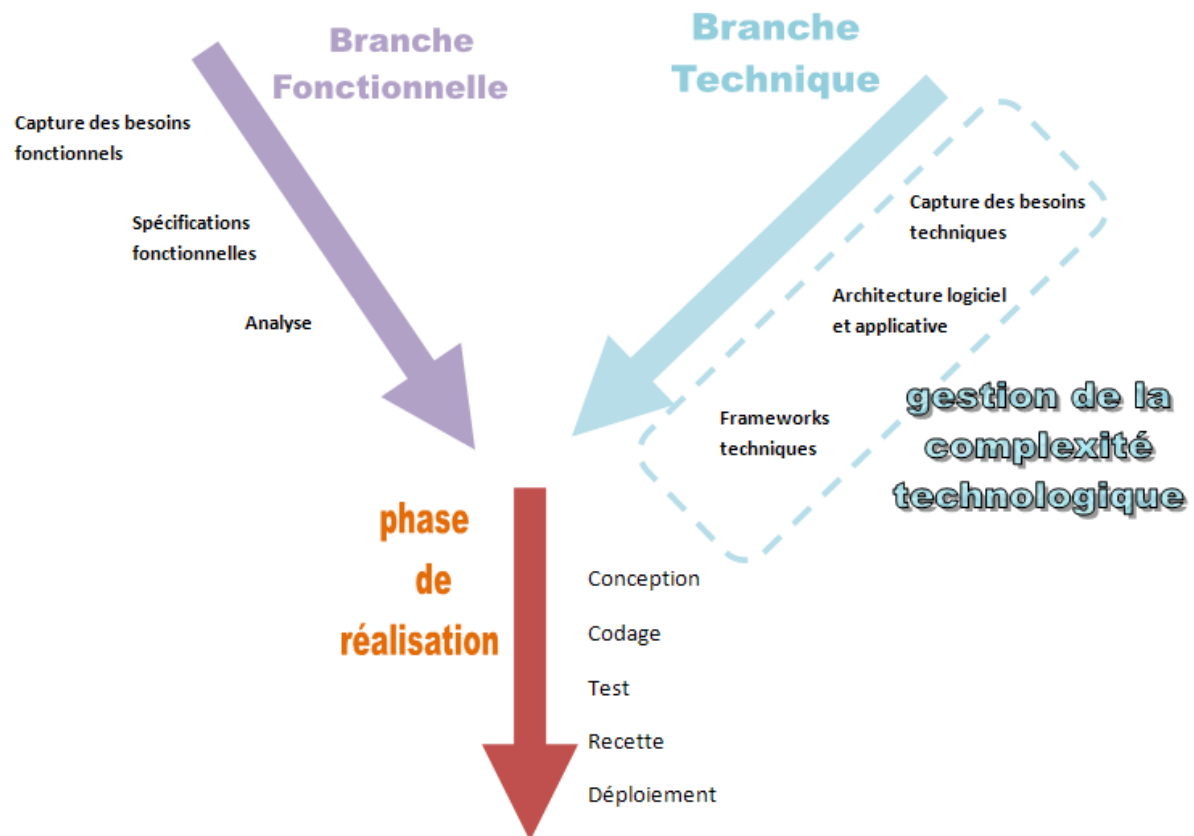


Figure 1 Le processus de développement décrit est appelé "2TUP" (Two Tracks Unified Process)

## IV. Conclusion

Ce chapitre a posé les bases de notre projet de système de gestion de restaurant en ligne en définissant son contexte, ses objectifs, les défis rencontrés et la problématique à laquelle nous cherchons à répondre. Dans les chapitres suivants, nous explorerons en détail les aspects techniques et fonctionnels de notre système, en suivant la méthodologie présentée dans ce chapitre.



# **Chapitre 2 : Analyse des Besoins**

## I. Introduction

Dans ce chapitre, nous procéderons à une analyse approfondie des besoins pour notre système de gestion de restaurant en ligne. Cette phase est essentielle pour comprendre les exigences des utilisateurs finaux ainsi que les fonctionnalités nécessaires pour répondre à ces besoins de manière efficace et efficiente.

## II. Identification des acteurs

Les acteurs sont des entités externes qui interagissent directement avec le système. Ils peuvent être humains, logiciels ou matériels. Dans le cadre de notre projet, nous avons identifié les acteurs suivants :

**Client :** Personne qui utilise l'application pour effectuer des réservations ou des commandes.

**Administrateur :** Personne chargée de gérer les aspects administratifs de l'application, tels que la gestion des menus, des réservations et des comptes utilisateurs.

**Système de paiement en ligne :** Plateforme permettant aux utilisateurs d'effectuer des transactions monétaires sécurisées via Internet pour régler des achats de biens ou de services

## III. Capture des besoins

### 1.1 Les besoins des utilisateurs

#### Client :

Le client doit pouvoir :

- Consulter le menu pour voir les plats disponibles avec leurs descriptions et prix.
- Ajouter des produits au panier en sélectionnant les articles souhaités.
- Finaliser la commande de manière sécurisée en passant à l'étape de paiement.
- Suivre l'état de la commande.

### Administrateur :

L'administrateur doit pouvoir :

- Accéder au tableau de bord pour consulter les statistiques de ventes, les revenus et d'autres données pertinentes.
- Gérer les produits en ajoutant de nouveaux articles, en modifiant les prix, en supprimant des éléments obsolètes et en gérant les niveaux de stock.
- Gérer les utilisateurs en créant, modifiant ou supprimant des comptes d'utilisateurs.

### Système de paiement en ligne :

Pour le système de paiement en ligne :

- Assurer la gestion des transactions sécurisées afin de garantir la sécurité des paiements des clients.

## **1.2 Les diagrammes de cas d'utilisation**

Un diagramme de cas d'utilisation illustre les interactions entre les utilisateurs (acteurs) et les différentes fonctionnalités (cas d'utilisation) d'un système. Les relations entre un acteur et un cas d'utilisation sont représentés par des associations, indiquant que l'acteur est impliqué dans l'exécution du cas d'utilisation. Ces cas d'utilisation représentent les actions ou les fonctionnalités spécifiques que le système doit offrir pour répondre aux besoins des utilisateurs, en réaction à des événements externes.

La figure ci-dessous (Figure 2) représente le diagramme de cas d'utilisation relatif au client.

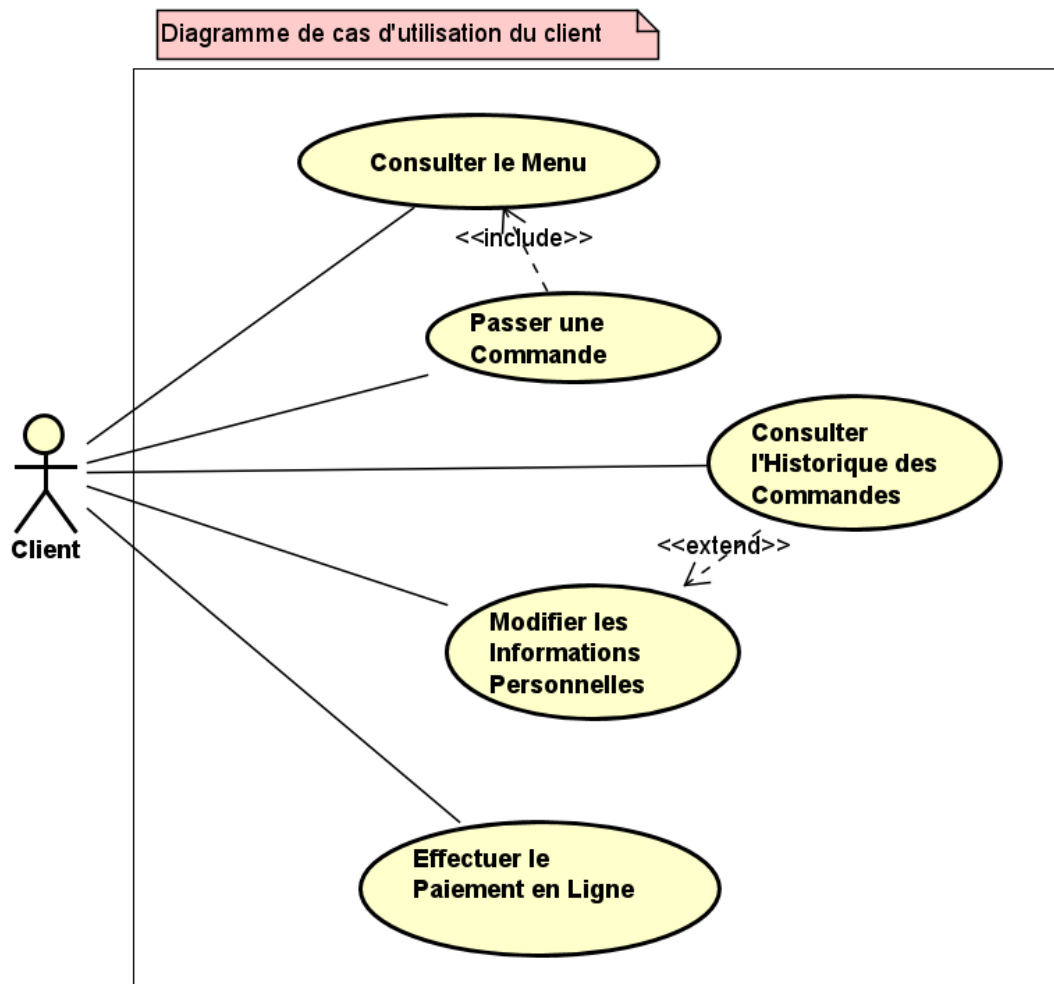


Figure 2 le diagramme de cas d'utilisation relatif au client

Ce diagramme de cas d'utilisation expose les différentes fonctionnalités accessibles à un client lorsqu'il utilise l'application de gestion de restaurant en ligne. Pour accéder à son espace personnel, le client doit s'authentifier. Une fois connecté, il peut consulter le menu du restaurant, effectuer une réservation en ligne, passer une commande pour emporter ou livraison, effectuer un paiement en ligne....

- La figure ci-dessous (Figure 3) représente le diagramme de cas d'utilisation relatif à l'administrateur de notre application.

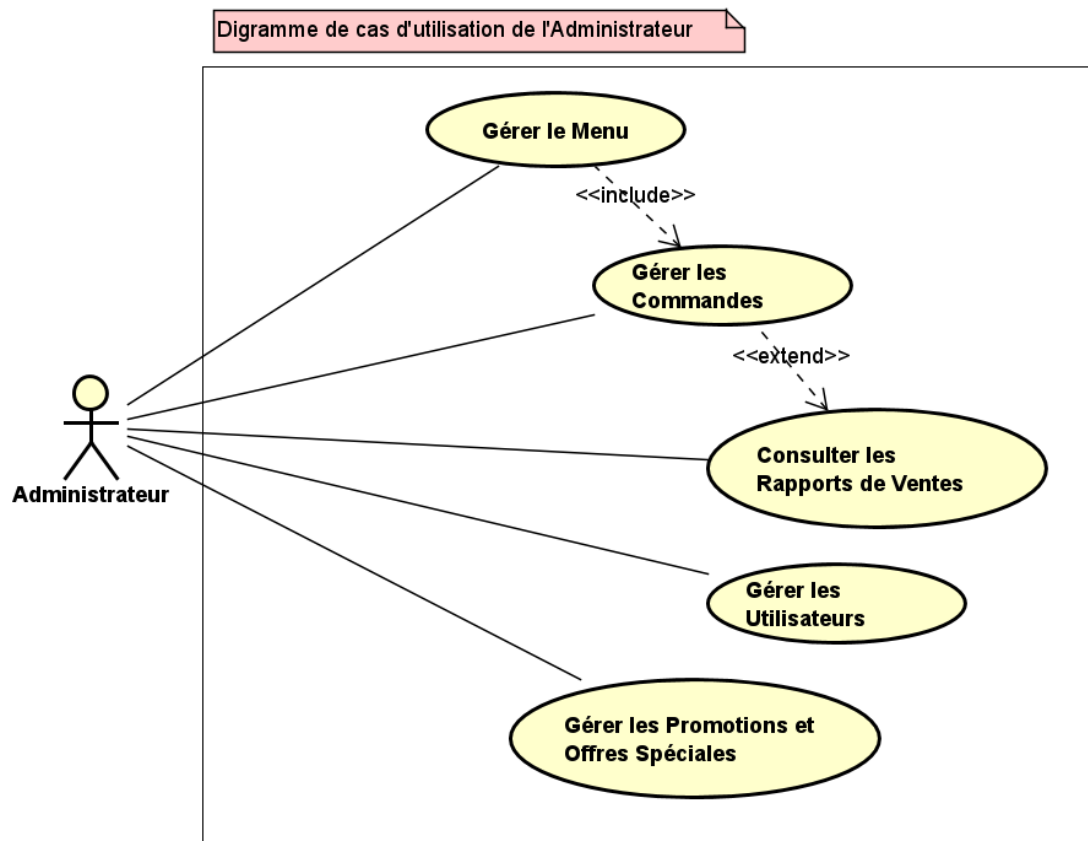


Figure 3 le diagramme de cas d'utilisation relatif à l'administrateur

Ce diagramme de cas d'utilisation expose les différentes fonctionnalités accessibles à un administrateur lorsqu'il utilise l'application de gestion de restaurant en ligne. Pour accéder à son espace personnel, l'administrateur doit s'authentifier. Une fois connecté, il peut gérer le menu du restaurant, gérer les commandes, gérer les utilisateurs, gérer les promotions et offres spéciales, ainsi que consulter les rapports de ventes. L'administrateur est responsable de l'ajout, de la modification et de la suppression des produits du menu, ainsi que de la gestion des commandes et des utilisateurs. De plus, il est chargé de la mise en place et de la gestion des promotions, et il peut consulter les rapports de ventes pour suivre les performances du restaurant.

### Diagramme de cas d'utilisation de l'application :

Le diagramme de cas d'utilisation présente les différentes fonctionnalités de notre application de gestion de restaurant en ligne, accessibles aux acteurs principaux : les clients et les administrateurs. Les clients peuvent effectuer diverses actions telles que consulter le menu, effectuer des réservations en ligne, passer des commandes pour emporter ou livraison, consulter

les horaires d'ouverture, lire les avis sur les plats, effectuer des paiements en ligne, et gérer leurs réservations. De leur côté, les administrateurs disposent de fonctionnalités de gestion avancées, leur permettant de gérer le menu en ajoutant, modifiant ou supprimant des produits, de gérer les commandes en les validant ou les annulant, de gérer les utilisateurs en créant, modifiant ou supprimant des comptes, de gérer les promotions et offres spéciales, et de consulter les rapports de ventes pour évaluer les performances du restaurant. Cette représentation garantit une expérience utilisateur fluide et une gestion efficace de l'ensemble du système.

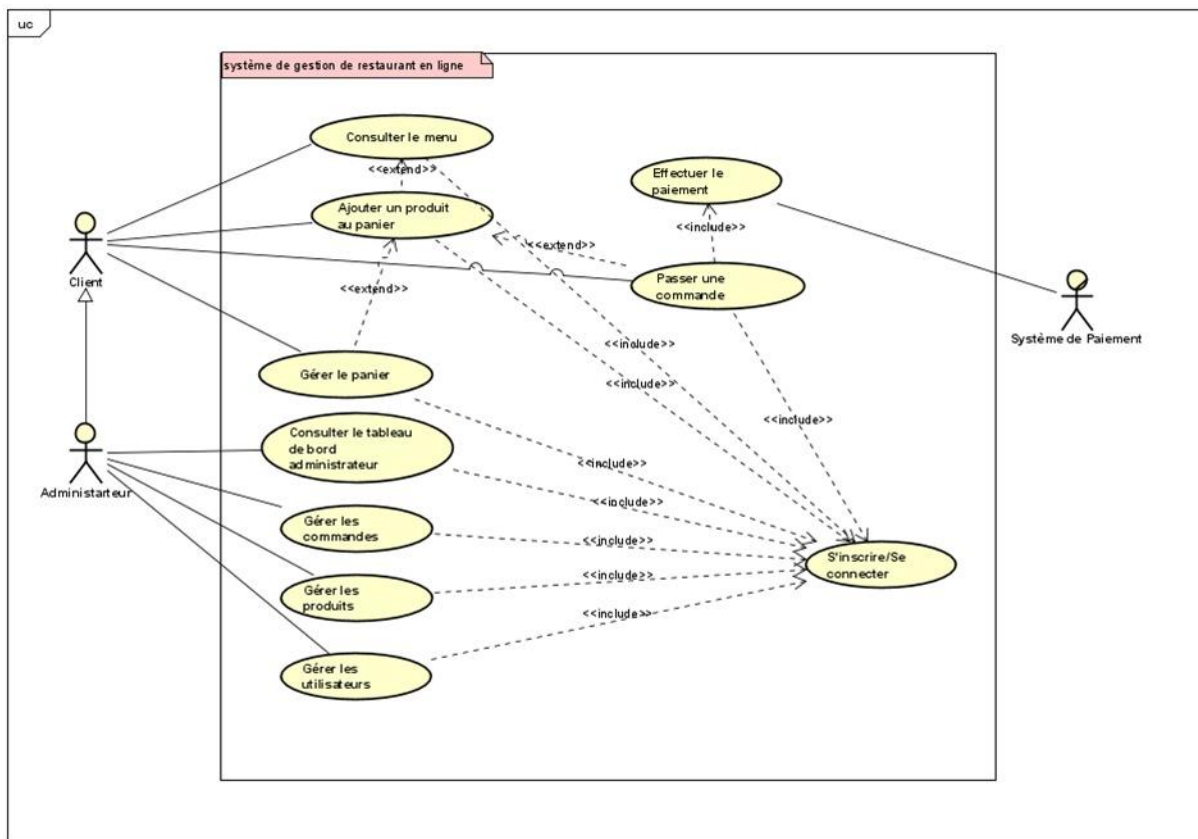


Figure 4 le diagramme de cas d'utilisation de l'application

## IV. Besoins fonctionnels spécifiques

### A. Fonctionnalités pour les clients

- ✓ Navigation conviviale pour parcourir le menu du restaurant et rechercher des plats spécifiques.

- ✓ Système de commande en ligne intuitif permettant aux clients de consulter le menu, d'ajouter des produits au panier et de finaliser leur commande de manière sécurisée.
- ✓ Processus de paiement sécurisé et convivial offrant différentes options de paiement pour la commodité des clients.
- ✓ Suivi des commandes en temps réel, permettant aux clients de suivre l'état de leur commande depuis la préparation jusqu'à sa prise en charge par le restaurant.
- ✓ Gestion du compte utilisateur, offrant aux clients la possibilité de modifier leurs informations personnelles et de consulter l'historique de leurs commandes pour une expérience personnalisée et transparente.

## **B. Fonctionnalités pour les administrateurs**

- ✓ Accès au tableau de bord pour consulter les statistiques de ventes, les revenus et d'autres données pertinentes.
- ✓ Gestion des produits, incluant l'ajout de nouveaux articles, la modification des prix, la suppression des éléments obsolètes et la gestion des niveaux de stock.
- ✓ Gestion des utilisateurs en créant, modifiant ou supprimant des comptes d'utilisateurs selon les besoins du restaurant.
- ✓ Gestion du menu, y compris l'ajout, la modification et la suppression de plats.
- ✓ Gestion des commandes en temps.

## **V. Besoins non fonctionnels**

Outre les fonctionnalités, notre système doit également répondre à des besoins non fonctionnels tels que la sécurité, la performance, la convivialité, et la scalabilité. Ces aspects sont essentiels pour garantir la fiabilité et la stabilité du système dans des conditions d'utilisation réelles.

## **VI. Conclusion**

En conclusion, ce chapitre a permis de réaliser une analyse approfondie des besoins pour notre système de gestion de restaurant en ligne. En identifiant les acteurs principaux, tels que les clients, les serveurs et les administrateurs, ainsi qu'en capturant les besoins fonctionnels et non fonctionnels, nous avons posé les bases nécessaires à la conception d'un système répondant efficacement aux attentes des utilisateurs. Les diagrammes de cas d'utilisation ont mis en lumière les fonctionnalités accessibles à chaque acteur, garantissant une expérience utilisateur optimale et une gestion efficace des opérations du restaurant. Cette phase préliminaire est essentielle pour orienter le développement du système et assurer sa pertinence et sa performance dans un environnement opérationnel.



# **Chapitre 3 :**

# **Conception**

## **I. Introduction**

Dans ce chapitre, nous avons conçu notre système de gestion de restaurant en ligne en passant par plusieurs étapes clés. Tout d'abord, nous avons défini la structure statique du système à l'aide d'un diagramme de classe, identifiant les entités, leurs attributs et leurs relations. Ensuite, nous avons analysé les interactions dynamiques entre les acteurs du système à travers un diagramme de séquence, modélisant les échanges de messages lors de l'exécution des cas d'utilisation principaux. Nous avons également examiné le flux des actions des utilisateurs via un diagramme d'activité, décrivant séquentiellement les étapes du processus métier. Enfin, nous avons abordé l'architecture physique du système avec un diagramme de déploiement, mettant en évidence les composants logiciels et matériels et leurs interactions dans l'environnement de déploiement. Ce chapitre offre une vision globale de notre processus de conception, jetant les bases du développement futur de notre système de gestion de restaurant en ligne.

## **II. Diagramme de Classe**

Dans cette section, nous présentons le diagramme de classe représentant la structure statique de notre système de gestion de restaurant en ligne. Ce diagramme illustre les différentes classes du système, ainsi que leurs attributs et leurs relations.

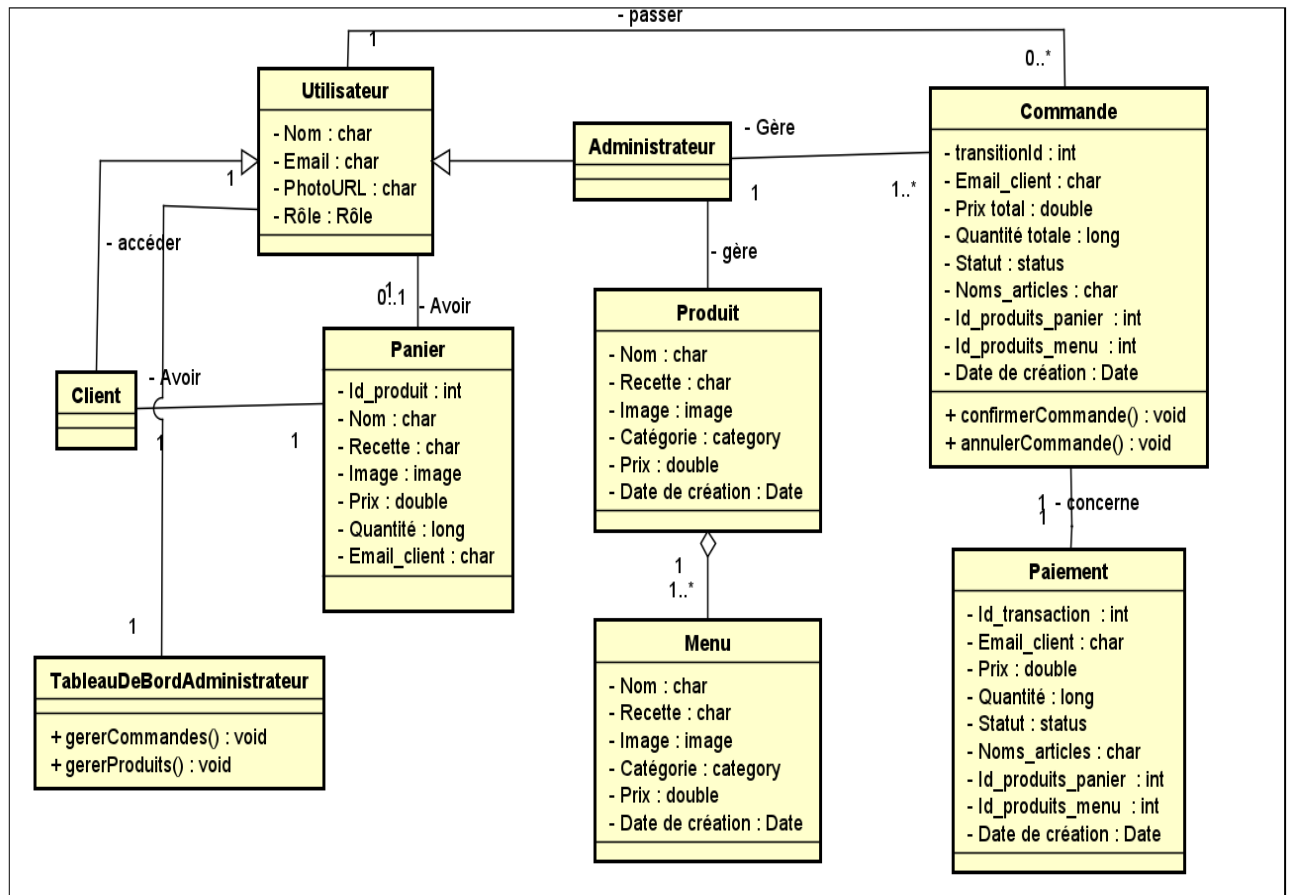


Figure 5 : Diagramme de classe

Le diagramme de classe ci-dessus décrit les classes principales du système, ainsi que leurs attributs et leurs relations. Voici une explication détaillée des classes et de leurs relations :

- **Client** : Représente un client du système, avec des attributs tels que le nom, l'email et l'URL de la photo. Il est associé à un panier pour stocker les produits sélectionnés et à une commande pour passer des commandes.
- **Administrateur** : Représente un administrateur du système, avec des attributs similaires à ceux du client. Il est associé à la gestion des commandes et des produits.
- **Utilisateur** : Cette classe représente un utilisateur générique du système, partageant les mêmes attributs que le client et l'administrateur, mais avec un attribut supplémentaire pour le rôle. Il est associé à un panier et à la possibilité de passer des commandes.
- **Produit** : Représente un produit disponible dans le système, avec des attributs tels que le nom, la recette, l'image, la catégorie, le prix, etc. Il est associé à un menu et peut être contenu dans un panier ou une commande.
- **Panier** : Représente le panier d'un utilisateur contenant les produits sélectionnés. Il est associé à un utilisateur et peut contenir plusieurs produits.

- **Commande** : Représente une commande passée par un utilisateur, contenant les détails de la commande tels que le prix total, la quantité totale, le statut, etc. Elle est associée à un utilisateur et peut contenir plusieurs produits.
- **Menu** : Représente un menu proposé par le restaurant, composé de plusieurs produits. Il est en agrégation avec la classe Produit.
- **TableauDeBordAdministrateur** : Représente le tableau de bord utilisé par l'administrateur pour gérer les commandes et les produits.
- **Paiement** : Représente les informations de paiement associées à une commande. Il est associé à une commande.

Ce diagramme de classe permet de visualiser la structure statique du système ainsi que les relations entre les différentes entités.

### III. Diagramme de séquence

Dans cette section, nous présentons le diagramme de séquence décrivant le flux des interactions entre les différents acteurs du système. Ce diagramme illustre le processus de commande et de paiement dans notre système, en mettant en évidence les étapes clés telles que la sélection des produits par le client, la création de la commande, et le traitement du paiement.

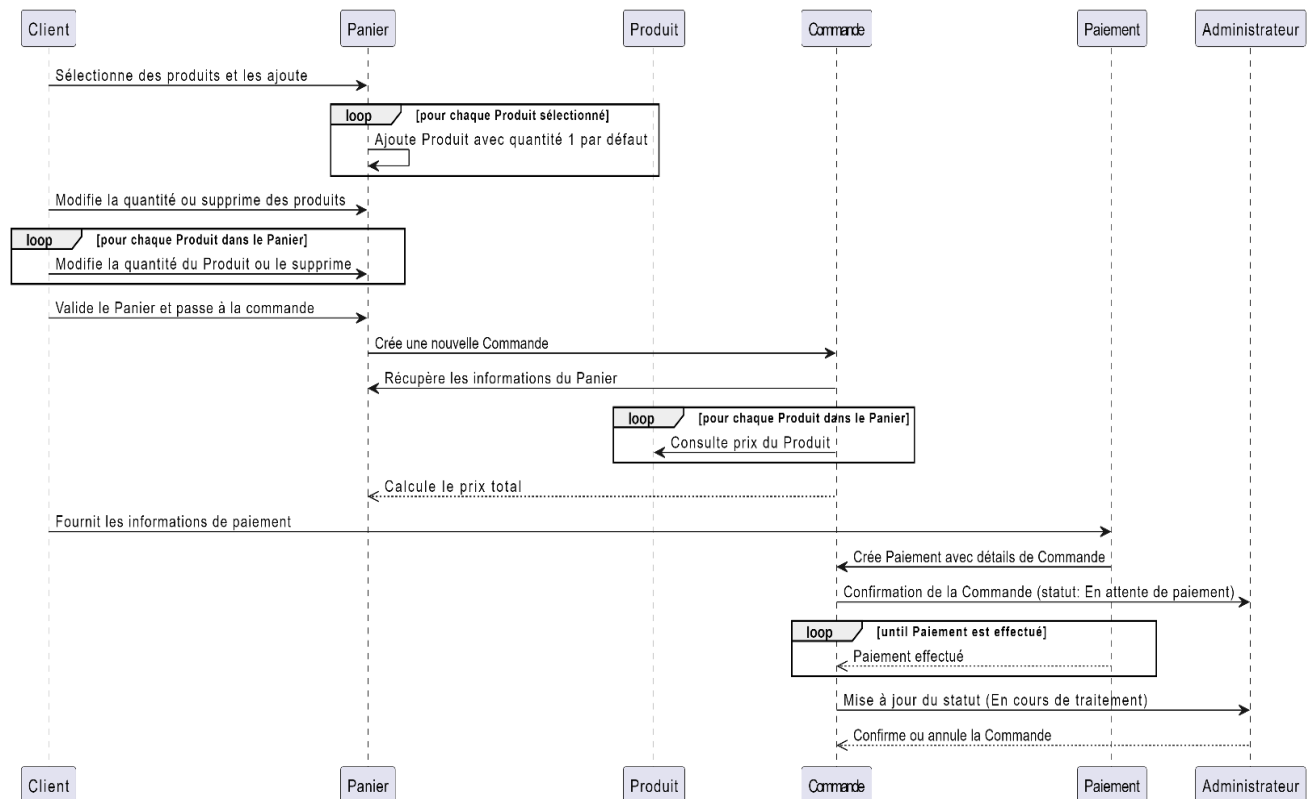


Figure 6 : Diagramme de séquence

Le diagramme de séquence ci-dessus représente les interactions entre les acteurs principaux du système lors du processus de commande et de paiement. Voici une explication détaillée des étapes principales :

- **Client** : Interagit avec le système en sélectionnant des produits, modifiant le panier et validant la commande.
- **Panier** : Gère les produits ajoutés par le client et les modifications apportées au panier.
- **Produit** : Consulté pour récupérer le prix des produits dans la commande.
- **Commande** : Crée une nouvelle commande à partir du panier, récupère les informations sur les produits et calcule le prix total.
- **Paiement** : Crée un paiement avec les détails de la commande et confirme lorsque le paiement est effectué.
- **Administrateur** : Confirme ou annule la commande et met à jour le statut de la commande.

Ce diagramme de séquence permet de visualiser de manière claire et concise le déroulement des interactions entre les différentes entités du système lors du processus de commande et de paiement.

## IV. Diagramme d'activité

Dans cette section, nous présentons le diagramme d'activité représentant le flux des actions effectuées par un client lors du processus de sélection, de modification du panier, de validation du panier et de confirmation de la commande dans notre système de gestion de restaurant en ligne.

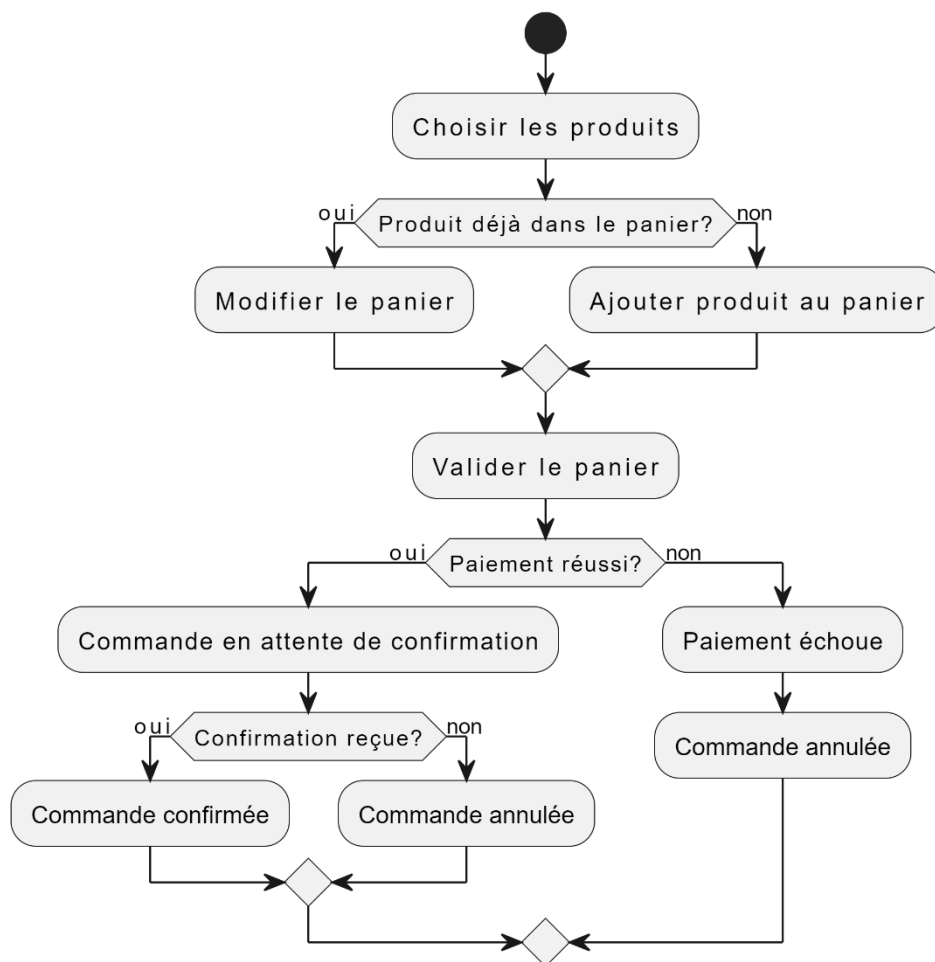


Figure 7: Diagramme d'activité

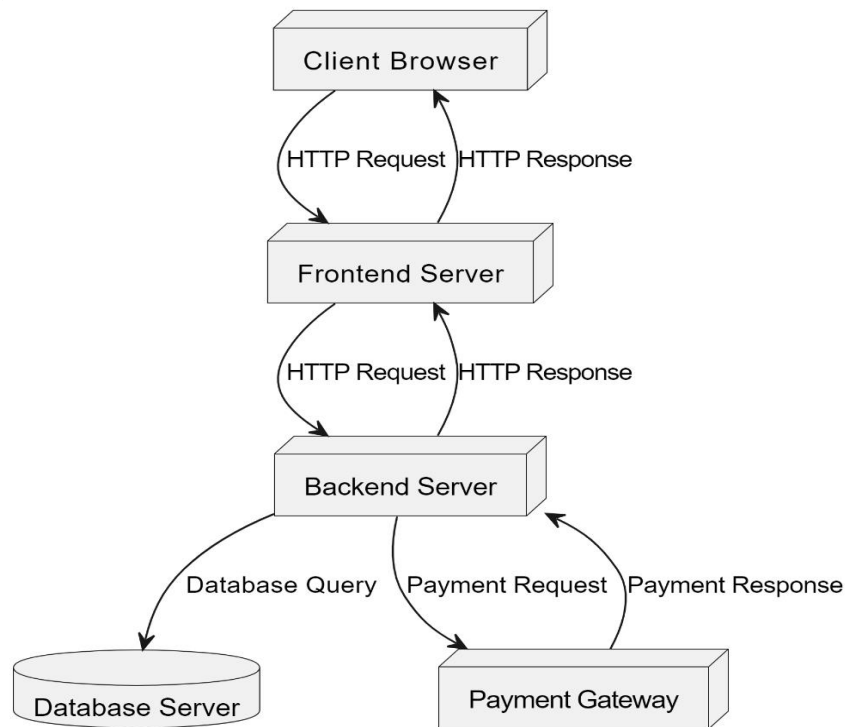
Le diagramme d'activité ci-dessus décrit le processus suivi par un client lors de l'utilisation de notre système. Voici une explication détaillée des différentes étapes :

- Choisir les produits : Le processus démarre avec le client choisissant les produits qu'il souhaite commander.
- Produit déjà dans le panier ? : Le système vérifie si le produit sélectionné est déjà présent dans le panier du client.
- Modifier le panier : Si le produit est déjà dans le panier, le client a la possibilité de modifier la quantité ou de supprimer le produit du panier.
- Ajouter produit au panier : Si le produit n'est pas déjà dans le panier, il est ajouté au panier du client.
- Valider le panier : Une fois que le client a terminé la sélection des produits, il valide son panier pour passer à l'étape suivante.
- Paiement réussi ? : Le système vérifie si le paiement est réussi.
- Commande en attente de confirmation : Si le paiement est réussi, la commande est placée en attente de confirmation.
- Confirmation reçue ? : Le système attend la confirmation de la commande.
- Commande confirmée : Si la confirmation est reçue, la commande est confirmée et le processus est terminé.
- Paiement échoué : Si le paiement échoue, la commande est annulée et le processus est terminé.

Ce diagramme d'activité permet de visualiser de manière claire et concise le flux des actions effectuées par un client lors du processus de commande dans notre système.

## V. Diagramme de déploiement

Dans cette section, nous présentons le diagramme de déploiement illustrant l'architecture physique de notre système de gestion de restaurant en ligne, en mettant en évidence les différents composants logiciels et matériels et leurs interactions.



*Figure 8: Diagramme de déploiement*

Le diagramme de déploiement ci-dessus décrit l'architecture physique de notre système de gestion de restaurant en ligne. Voici une explication détaillée des différents composants et de leurs interactions :

- Client Browser : Représente le navigateur web utilisé par le client pour accéder au système.
- Frontend Server : Serveur web responsable de fournir l'interface utilisateur et de traiter les requêtes HTTP provenant du navigateur client.
- Backend Server : Serveur web responsable de traiter la logique métier et de communiquer avec la base de données et la passerelle de paiement.
- Database Server : Serveur de base de données utilisé pour stocker les données du système, telles que les informations des clients, des produits, des commandes, etc.
- Payment Gateway : Passerelle de paiement utilisée pour gérer les transactions de paiement entre le système et les fournisseurs de services de paiement externes.

Les interactions entre les composants se déroulent comme suit :

- Le Client Browser envoie une requête HTTP au Frontend Server pour accéder au système.
- Le Frontend Server transfère la requête au Backend Server pour traiter la logique métier.



- Le Backend Server effectue des requêtes de base de données au Database Server pour récupérer ou modifier les données nécessaires.
- Le Backend Server envoie une demande de paiement à la Passerelle de paiement.
- La Passerelle de paiement traite la demande de paiement et envoie une réponse au Backend Server.
- Le Backend Server génère une réponse HTTP qu'il envoie au Frontend Server.
- Le Frontend Server renvoie une réponse HTTP au Client Browser.

Ce diagramme de déploiement permet de visualiser la manière dont les différents composants logiciels et matériels interagissent pour assurer le bon fonctionnement de notre système.

## VI. Conclusion

Dans ce chapitre, nous avons entamé la phase de conception de notre système de gestion de restaurant en ligne en élaborant plusieurs diagrammes. Nous avons débuté par modéliser la structure statique du système avec le diagramme de classe, identifiant les entités, leurs attributs et leurs relations. Ensuite, nous avons examiné les interactions dynamiques entre les acteurs du système à travers le diagramme de séquence, offrant une vision séquentielle des échanges de messages lors de l'exécution des cas d'utilisation principaux. Par la suite, le diagramme d'activité nous a permis de visualiser séquentiellement les étapes du processus métier effectuées par les utilisateurs, soulignant ainsi l'aspect opérationnel du système. Enfin, le diagramme de déploiement a mis en évidence l'architecture physique du système, détaillant les composants logiciels et matériels et leurs interactions dans l'environnement de déploiement. Ces diagrammes ont jeté les bases solides pour le développement ultérieur du système, nous permettant d'aborder cette phase avec confiance et précision.

# **Chapitre 4 :**

# **Réalisation**

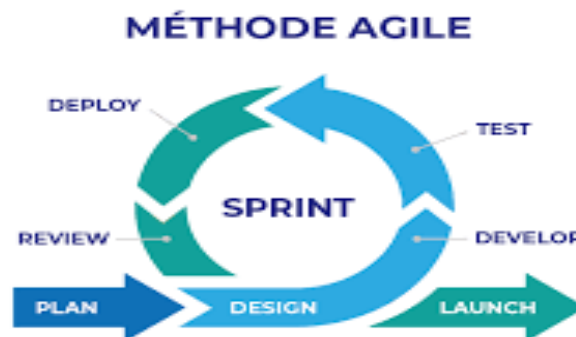
## I. Introduction

Ce chapitre marque la conclusion finale de notre projet de système de gestion de restaurant en ligne. Nous entrons dans la phase de réalisation où nous détaillerons l'environnement de développement ainsi que les outils qui ont été essentiels pour la concrétisation de notre application. Ce chapitre représente l'aboutissement de nos efforts de planification et de conception, et met en lumière les choix technologiques et les méthodes de développement adoptés tout au long du processus. De plus, nous présenterons une vue d'ensemble de l'application résultante, démontrant comment elle répond aux besoins identifiés et aux objectifs fixés dans les précédents chapitres. En résumé, cette section offre un aperçu complet du chemin parcouru depuis la phase initiale jusqu'à la réalisation concrète de notre solution de gestion de restaurant en ligne.

## II. Méthodologie de Développement

### 1. Méthodologie Agile

Dans le cadre de notre projet, nous avons opté pour une approche Agile afin de répondre de manière flexible et efficace aux besoins changeants du client. Nous avons utilisé des méthodes itératives et incrémentales pour développer notre application, favorisant la collaboration étroite avec le client et l'adaptation continue aux retours d'expérience.



### **Sprint Planning**

Nous avons planifié nos travaux en sprints de deux semaines, lors desquels nous avons défini les fonctionnalités à implémenter, estimé les efforts nécessaires et réparti les tâches entre les membres de l'équipe. Cette planification itérative nous a permis de maintenir un rythme de développement soutenu tout en assurant une flexibilité pour ajuster nos priorités en fonction des besoins du projet.

### **Daily Stand-up Meetings**

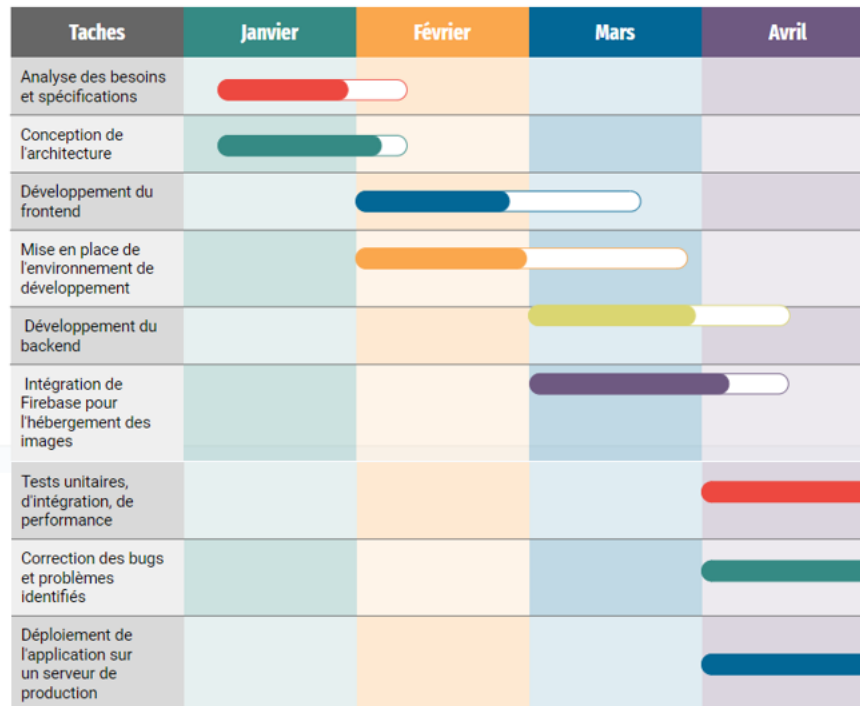
Chaque jour, nous avons tenu des réunions de stand-up pour synchroniser l'équipe, discuter des progrès réalisés, des obstacles rencontrés et des objectifs pour la journée à venir. Ces réunions courtes et régulières ont favorisé la transparence, la communication et la résolution rapide des problèmes.

### **Sprint Review et Sprint Retrospective**

À la fin de chaque sprint, nous avons organisé une revue avec le client pour présenter les fonctionnalités développées et recueillir ses retours. Nous avons également réalisé une rétrospective d'équipe pour évaluer notre performance, identifier les points forts et les axes d'amélioration, et ajuster notre processus de développement en conséquence.

## **2. Diagramme de Gantt**

## Projet de Gestion de Restaurant en Ligne



### Chapitre 4 : Réalisation

Cette section nous permettra d'intégrer efficacement la méthodologie Agile et le diagramme de Gantt dans notre rapport, en mettant en évidence notre approche de gestion de projet et notre planification temporelle.

## III. Technologies Utilisées

### 1.Langages de Programmation et Frameworks

#### Node.js et Express.js :

Node.js est un environnement d'exécution JavaScript côté serveur, tandis que Express.js est un framework minimaliste pour construire des applications web avec Node.js. Nous avons choisi cette combinaison pour développer notre backend, car elle offre une grande flexibilité et une gestion efficace des requêtes HTTP.



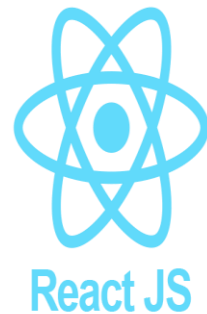
#### MongoDB :

MongoDB est une base de données NoSQL orientée document, choisie pour stocker les données de notre application. Son modèle de données flexible est bien adapté à la nature variée des informations d'un système de gestion de restaurant.



### **React :**

React est une bibliothèque JavaScript développée par Facebook, utilisée pour la construction d'interfaces utilisateur interactives. Nous l'avons choisi pour développer le frontend de notre application en raison de sa modularité et de sa performance.



## Chapitre 4 : Réalisation

---

### **Tailwind CSS :**

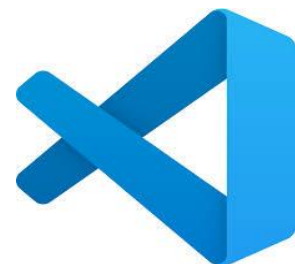
Tailwind CSS est un framework CSS utilitaire qui permet de construire rapidement des interfaces utilisateur modernes et esthétiques. Contrairement aux autres frameworks CSS comme Bootstrap, Tailwind CSS adopte une approche basée sur les classes utilitaires, ce qui offre une plus grande flexibilité et une personnalisation aisée des styles.



## **2.Outils de Développement**

### **Visual Studio Code :**

Visual Studio Code est un éditeur de code source gratuit et open-source, largement utilisé pour le développement web. Nous l'avons choisi pour sa légèreté, sa polyvalence et son large éventail d'extensions.



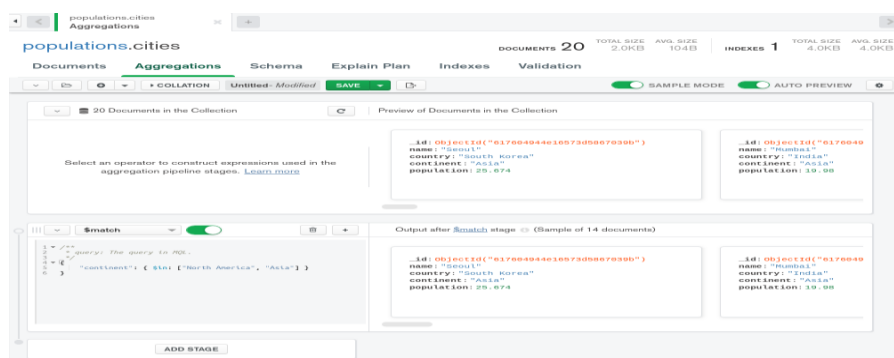
## Postman :

Postman est un outil de collaboration pour le développement d'API. Nous l'avons utilisé pour tester et déboguer nos API RESTful, assurant ainsi leur bon fonctionnement avant l'intégration dans l'application.



## MongoDB Compass :

MongoDB Compass est une interface graphique utilisateur pour MongoDB, qui nous a aidés à explorer et à manipuler nos données MongoDB de manière visuelle et intuitive.



## Chapitre 4 : Réalisation

---

## GitHub :

Git est un système de contrôle de version distribué, tandis que GitHub est une plateforme de développement collaboratif basée sur Git. Nous les avons utilisés pour gérer le code source de notre projet, faciliter la collaboration et assurer la traçabilité des modifications.



## Swagger :

Swagger est un ensemble d'outils open-source pour la conception, la construction, la documentation et la consommation d'API RESTful. Nous l'avons utilisé pour documenter nos API et faciliter l'intégration avec d'autres parties de l'application.



## Stripe :

Stripe est une plateforme de paiement en ligne offrant des solutions sécurisées pour les entreprises. Intégrée à notre système, elle assure une expérience de paiement fluide et sécurisée, acceptant divers modes de paiement tout en garantissant la sécurité des transactions.



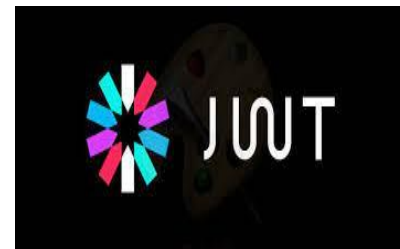
### **Firebase :**

Firebase est une plateforme de développement d'applications mobiles et web de Google. Nous l'avons utilisée pour l'authentification des utilisateurs, le stockage de fichiers et d'autres fonctionnalités backend, offrant ainsi une solution clé en main pour nos besoins.



### **JSON Web Tokens :**

JSON Web Tokens (JWT) est un format de jeton sécurisé utilisé pour transmettre des informations entre deux parties de manière compacte et autonome. Ils sont couramment utilisés pour l'authentification et l'autorisation dans les applications web et les services API.



### **Chapitre 4 : Réalisation**

---

Cette sélection de technologies et d'outils a été faite en tenant compte des exigences fonctionnelles et non fonctionnelles de notre projet de gestion de restaurant en ligne, afin d'assurer une implémentation efficace et performante.

## **IV. Description de notre application**

### **1. Page d'accueil :**





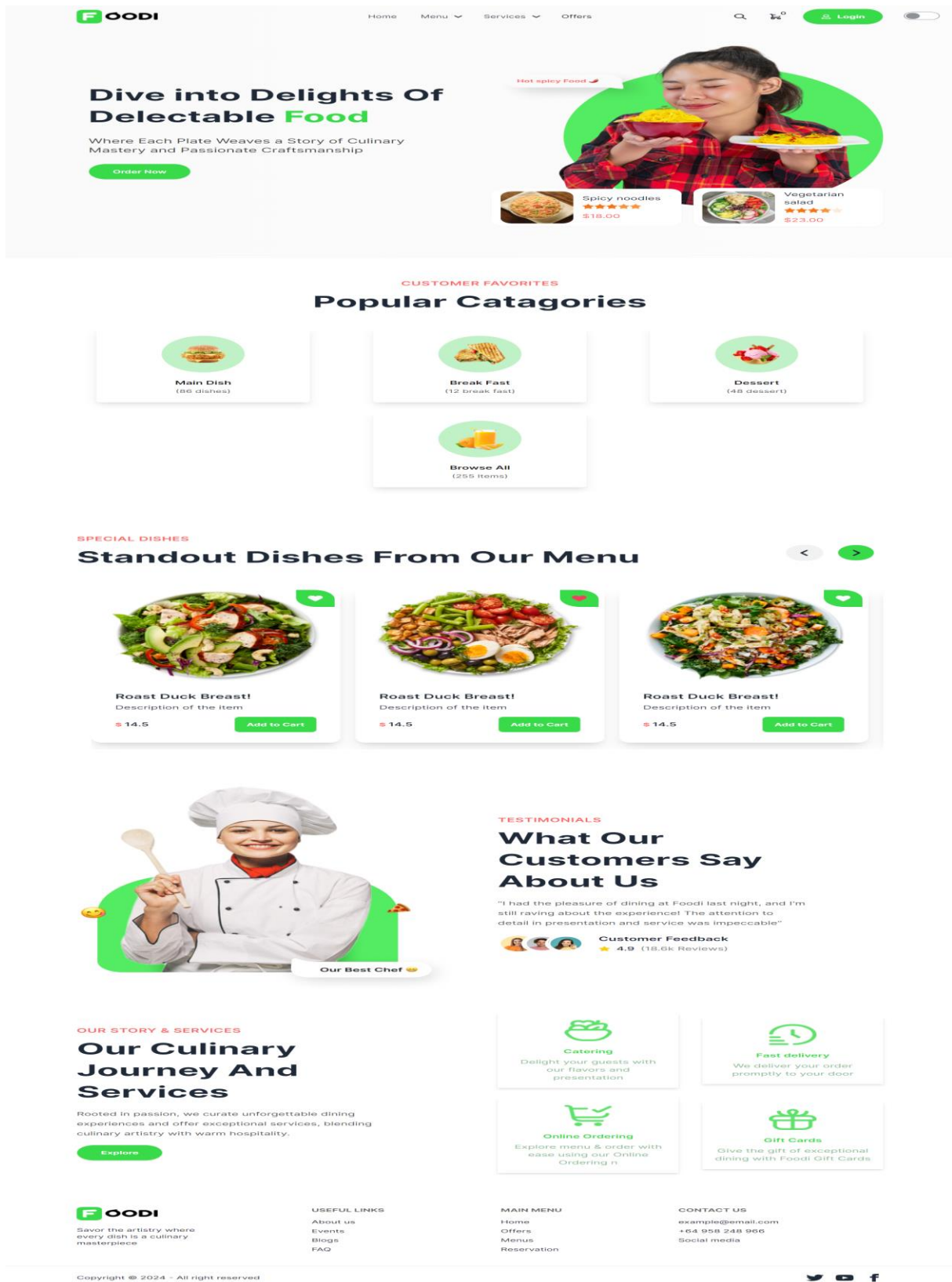


Figure 9: Page d'Accueil



Figure 10: Page d'accueil en mode sombre, en français

## 2. Les pages d'authentification et d'inscription :

### Login :

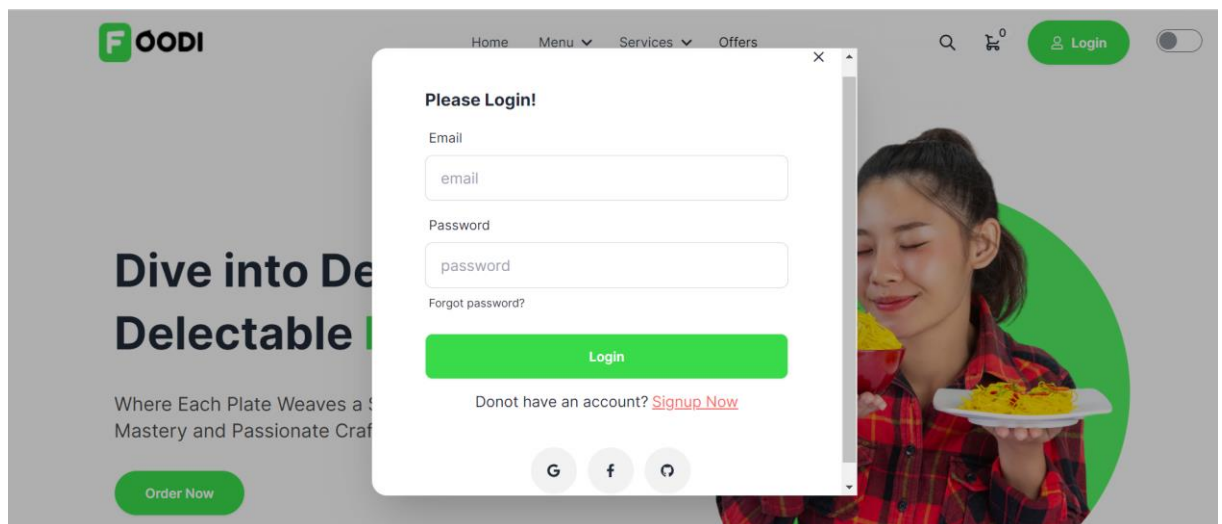


Figure 11: Page Login

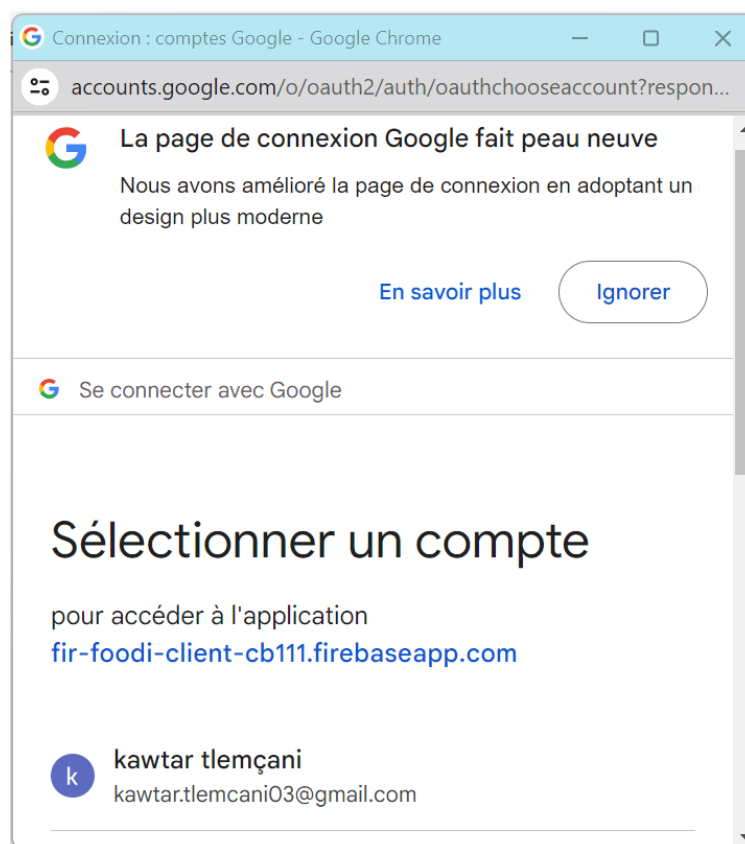
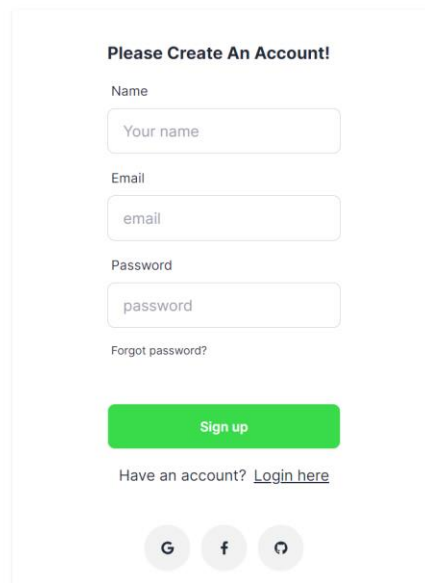


Figure 12: se connecter avec "google"

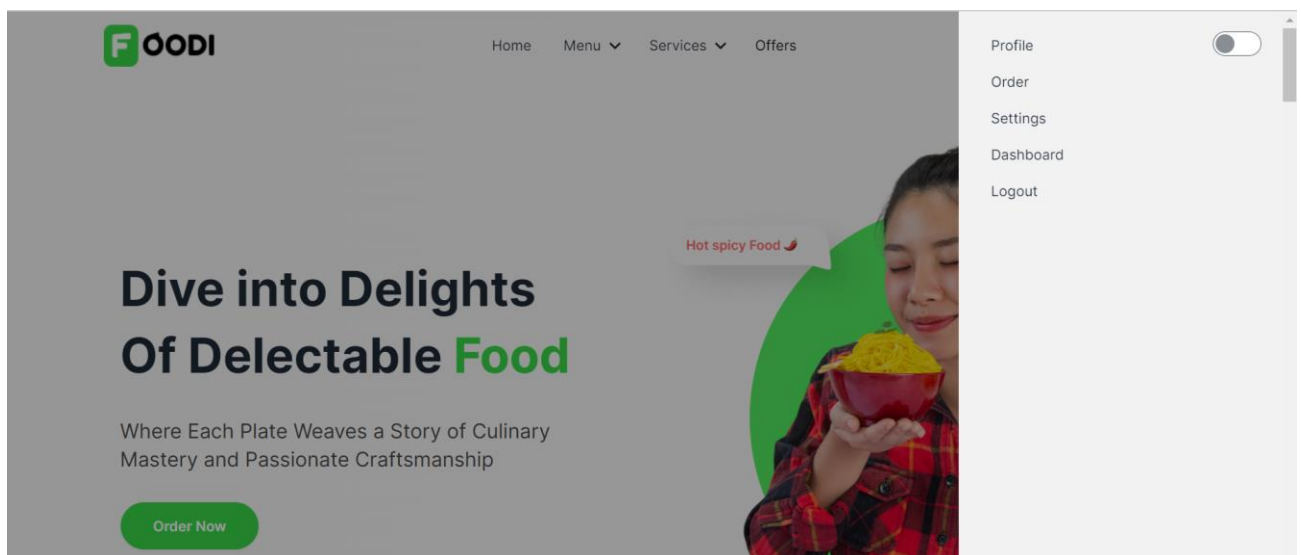
## Signup :



The image shows a 'Please Create An Account!' form. It includes input fields for Name (placeholder: 'Your name'), Email (placeholder: 'email'), and Password (placeholder: 'password'). Below the password field is a link for 'Forgot password?'. A green 'Sign up' button is positioned below the form. At the bottom, there is a link 'Have an account? [Login here](#)' and three social media icons for Google, Facebook, and Twitter.

*Figure 13: Page Signup*

## Profile:



*Figure 14: Navbar*

Name

Upload Photo

CHOISIR UN FICHIER

Aucun... choisi

Update

*Figure 15: Modifier le profile*

### 3. Page de produits

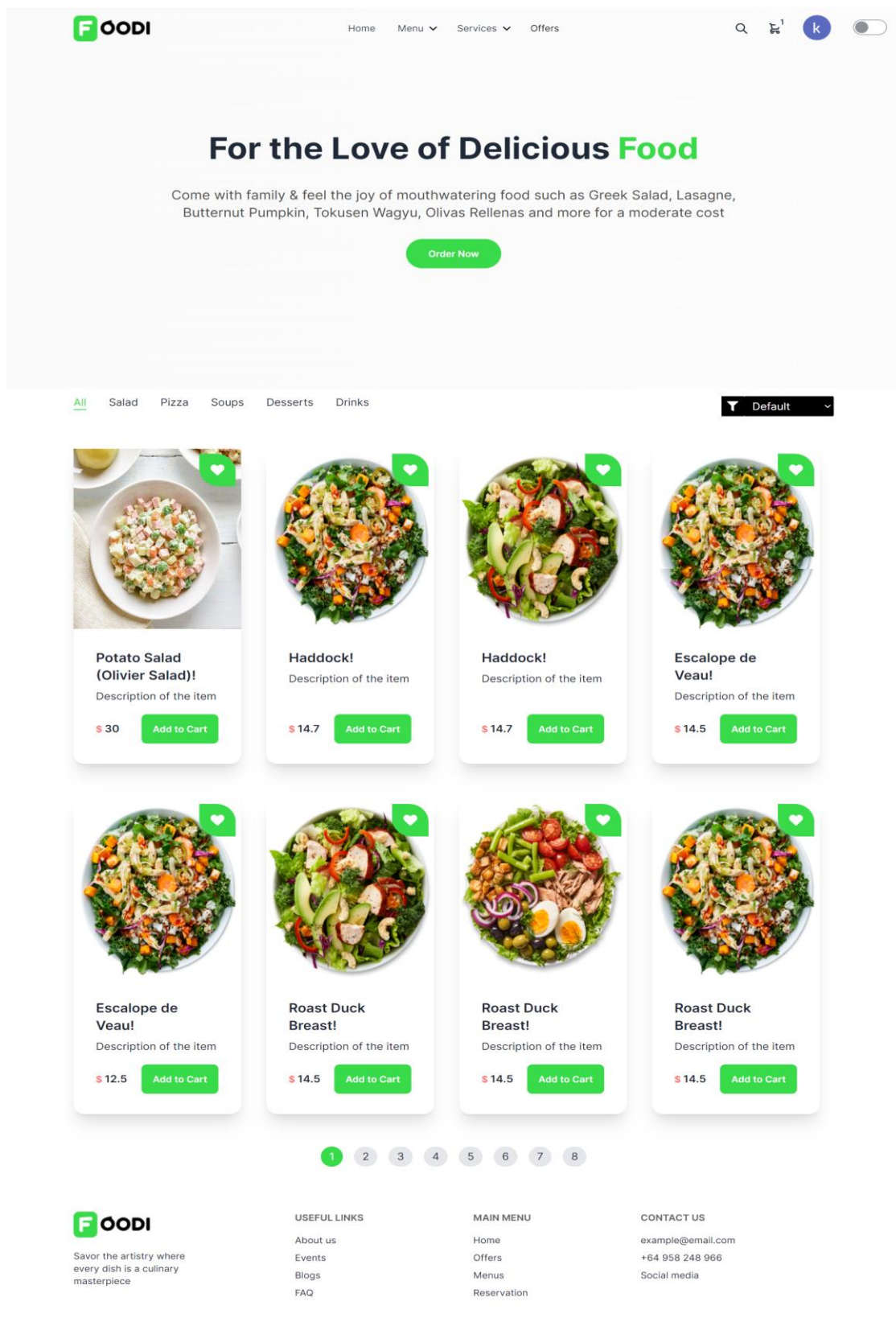
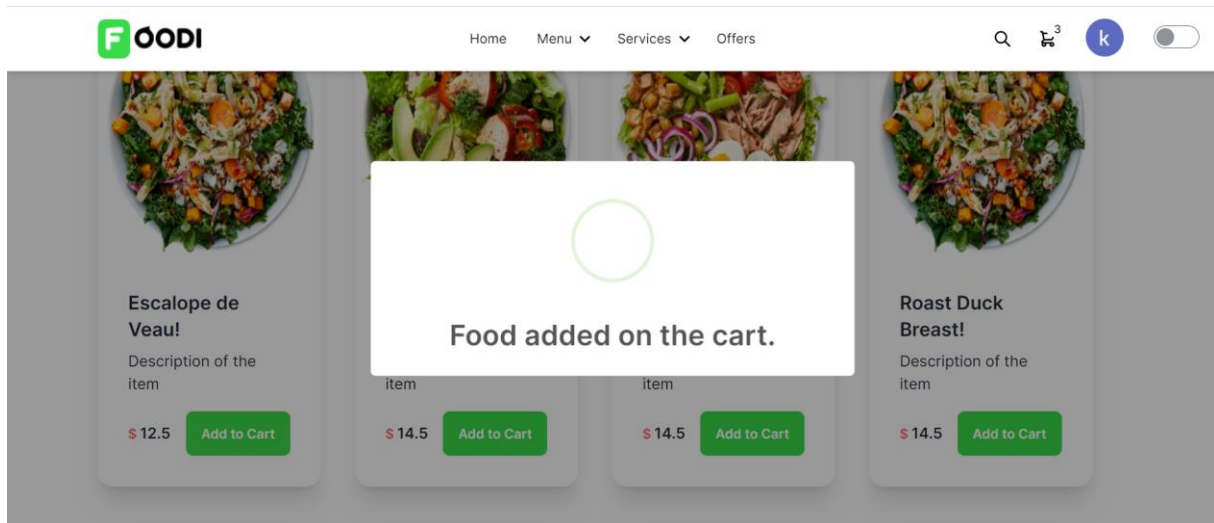
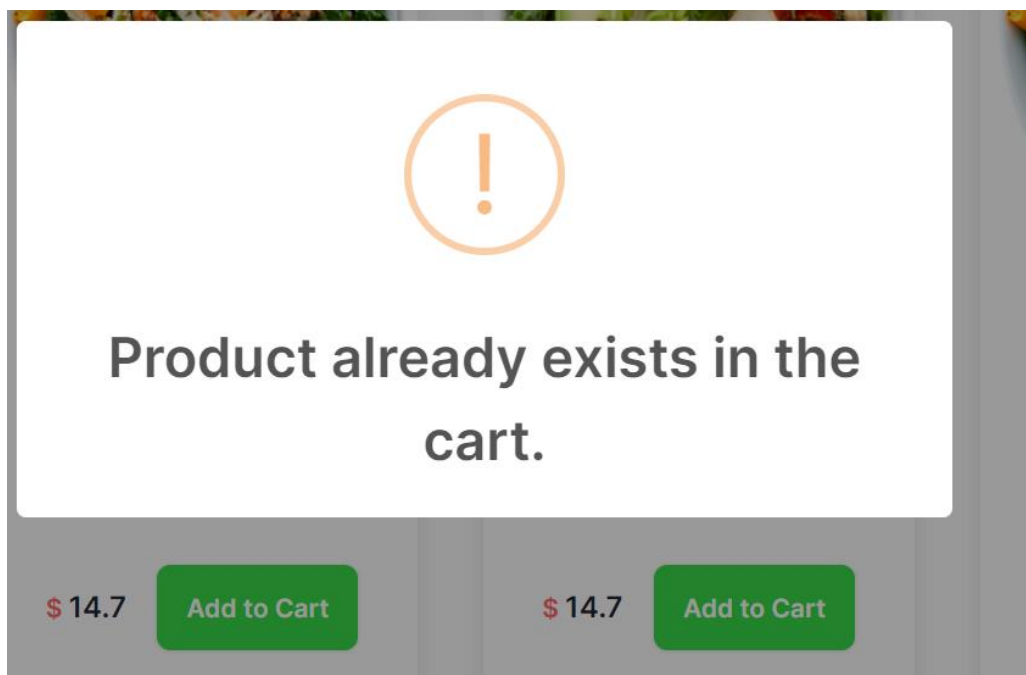


Figure 16: Page du menu





*Figure 17: Ajouter un produit au panier*



*Figure 18: Le produit existe déjà dans le panier*

## 4. Page du panier



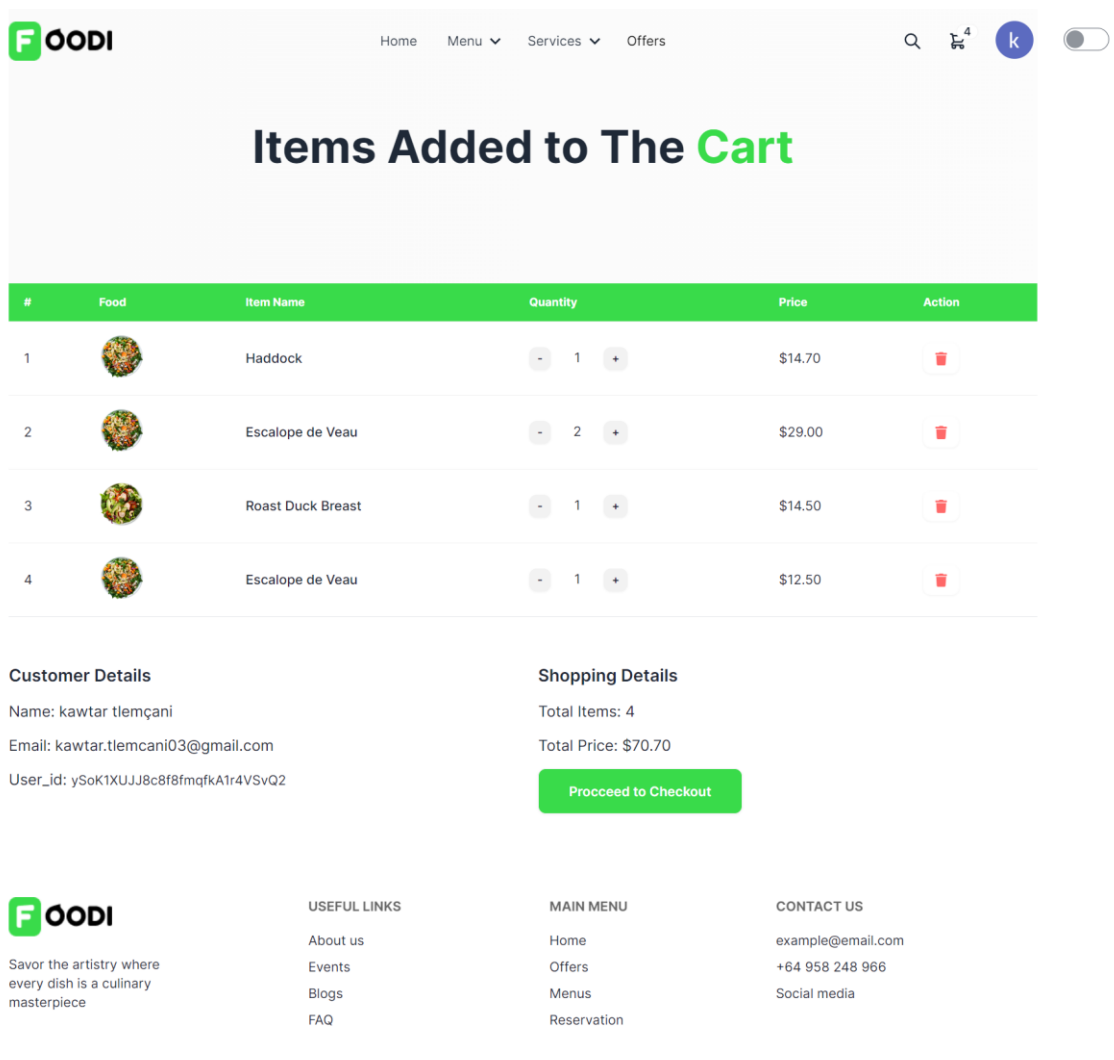


Figure 19: Page du panier

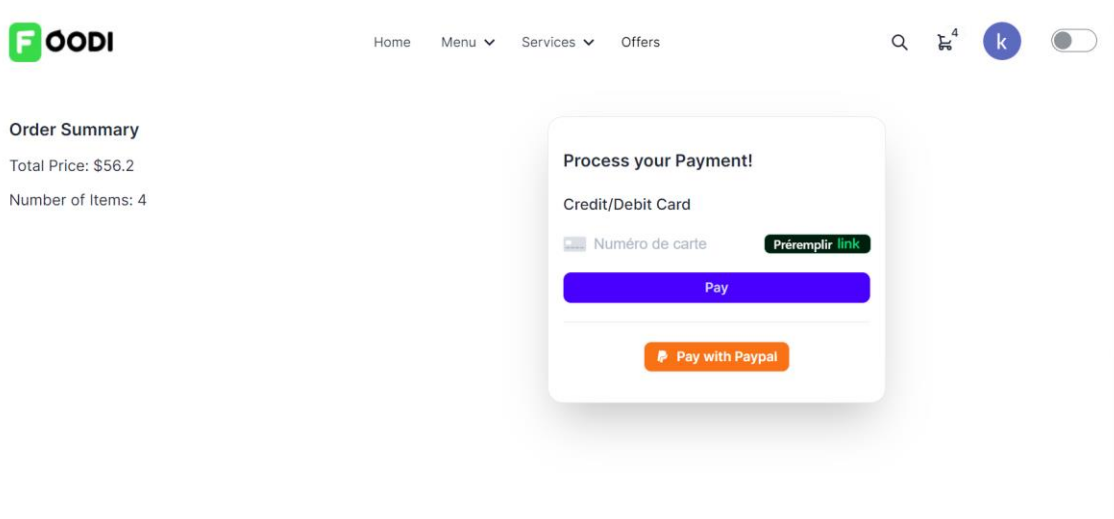


Figure 20: Page du paiement

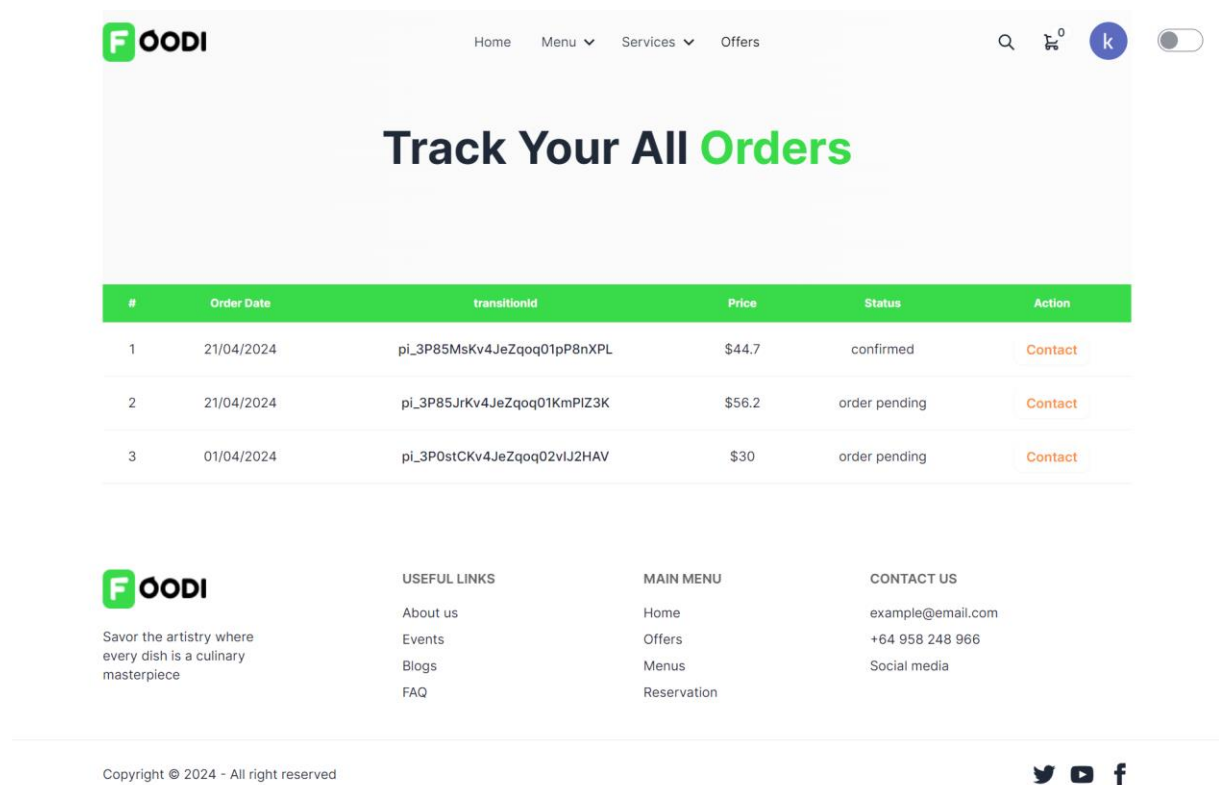


Figure 21: Suivre les commandes

## 5. Interface de l'administrateur

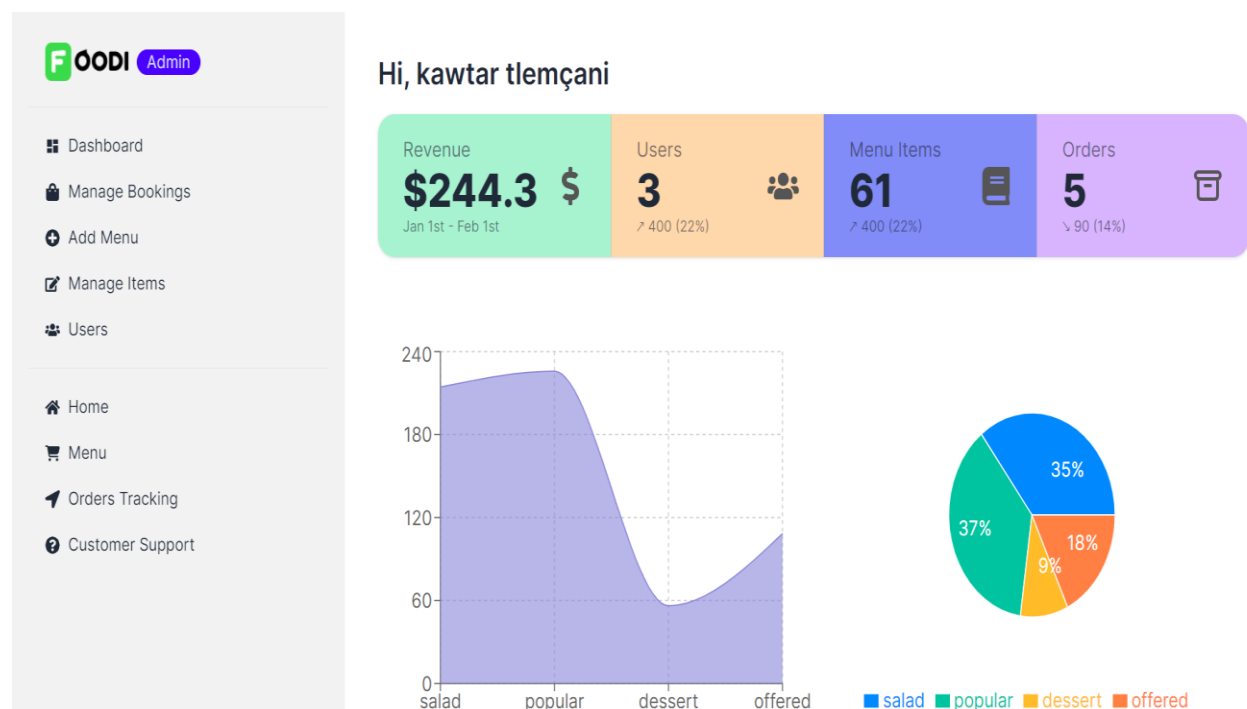


Figure 22: Dashboard

**Manage All Bookings!**

#	User	Transition Id	Price	Status	Confirm Order	Delete
1	kawtar.tlemcani03@gmail.com	pi_3P85MsKv4JeZqoq01pP8nXPL	\$44.7	confirmed	done	
2	kawtar.tlemcani03@gmail.com	pi_3P85JrKv4JeZqoq01KmPIZ3K	\$56.2	order pending		
3	wiam.fahoul@usmba.ac.ma	pi_3P6VWuKv4JeZqoq02xZpDmsK	\$59.2	confirmed	done	
4	fahoulwiam@gmail.com	pi_3P6VOjkv4JeZqoq02hMKNryn	\$54.2	order pending		
5	kawtar.tlemcani03@gmail.com	pi_3P0stCKv4JeZqoq02vjJ2HAV	\$30	order pending		

[< Previous](#) [Next >](#)

Figure 23: Gestion des commandes

**Upload A New Menu Item**

Recipe Name\*

Recipe Name

Category\* Price\*

Select a category Price

Recipe Details

Bio

CHOISIR UN FICHIER Aucun... choisi

Add Item

Figure 24: Ajouter un produit

**F OODI** Admin
 

- Dashboard
- Manage Bookings
- Add Menu
- Manage Items
- Users

- Home
- Menu
- Orders Tracking
- Customer Support

### Manage All Menu Items!

#	Image	Item Name	Price	Update	Delete
1		Potato Salad (Olivier Salad)	\$30		
2		Haddock	\$14.7		
3		Haddock	\$14.7		
4		Escalope de Veau	\$14.5		
5		Escalope de Veau	\$12.5		
6		Roast Duck Breast	\$14.5		
7		Roast Duck Breast	\$14.5		
8		Roast Duck Breast	\$14.5		
9		Roast Duck Breast	\$14.5		
10		Escalope de Veau	\$12.5		

← Previous
Next →

Figure 25: Gestion des produits

**F OODI** Admin
 

- Dashboard
- Manage Bookings
- Add Menu
- Manage Items
- Users

- Home
- Menu
- Orders Tracking
- Customer Support

### All Users

#	Name	Email	Role	Action
1	kawtar tlemçani	kawtar.tlemcani03@gmail.com	Admin	
2	WIAM FAHOUL	fahoulwiam@gmail.com	Admin	
3	WIAM FAHOUL	wiam.fahoul@usmba.ac.ma		

Total Users: 3

Figure 26: Gestion des utilisateurs

## V. Conclusion

Dans ce chapitre, nous avons abordé la phase cruciale de mise en œuvre de notre projet de gestion de restaurant. Nous avons commencé par exposer les outils de développement ainsi que l'environnement de travail que nous avons employés pour donner vie à notre application. Par la suite, nous avons présenté en détail l'architecture de l'application, tant pour sa version web que mobile. Chaque aspect a été minutieusement examiné, avec des définitions claires et des explications détaillées des différentes pages de l'application. Des illustrations représentatives ont été incluses pour offrir une visualisation concrète de notre travail. Ce chapitre marque une étape décisive dans notre parcours, où la planification et la conception se concrétisent en une application fonctionnelle et intuitive, prête à répondre aux besoins des utilisateurs et à offrir une expérience enrichissante dans le domaine de la gestion de restaurant.

## CONCLUSION GÉNÉRALE

Au cours de ce rapport, nous avons exposé les différentes étapes de conception et de réalisation de notre projet de gestion de restaurant en ligne.

Le projet a été réalisé dans le cadre de notre formation en 2ème année Génie Informatique à l'Ecole Nationale des Sciences Appliquées Fès comme un projet de fin d'année. Notre objectif était de concevoir une plateforme web innovante pour la gestion complète des restaurants, offrant une expérience utilisateur exceptionnelle et répondant aux besoins spécifiques de l'industrie de la restauration.

Pour la réalisation de ce projet, nous avons utilisé une combinaison de technologies modernes, notamment Express JS pour le backend, MongoDB comme système de gestion de base de données, React pour le frontend, ainsi que Firebase pour l'hébergement des images des produits. Cette approche nous a permis de développer une solution robuste et flexible, capable de s'adapter aux exigences changeantes du marché.

Les perspectives d'évolution de notre application sont nombreuses et variées. Nous envisageons d'intégrer des fonctionnalités telles que la gestion des réclamations, des recommandations personnalisées, ainsi qu'un système de chat pour faciliter la communication entre les clients et les administrateurs de l'application. Ces ajouts contribueront à enrichir l'expérience utilisateur et à renforcer la valeur ajoutée de notre solution sur le marché.

En conclusion, cette expérience a été extrêmement enrichissante sur les plans professionnel et académique. Elle nous a permis de mettre en pratique nos connaissances théoriques dans un projet concret et de développer des compétences précieuses en matière de développement web. Nous sommes convaincus que notre projet de gestion de restaurant en ligne apportera une contribution significative à l'industrie de la restauration, en offrant aux restaurateurs les outils nécessaires pour prospérer dans un environnement commercial de plus en plus numérique et compétitif.

## BIBLIOGRAPHIE

1. **Eat App.** (s. d.). *Eat App est un système de réservation de restaurant en ligne basé sur le cloud.* <https://eatapp.co/> (Consulté le 15 février 2024).
2. **Resy.** (s. d.). *Resy propose un système de réservation de restaurant en ligne avec des fonctionnalités de gestion des tables, d'intégration avec les systèmes de point de vente (POS) et de gestion des listes d'attente. Il permet également aux clients d'annuler ou de confirmer leur réservation par SMS.* <https://restaurant.eatapp.co/> <https://restaurant.eatapp.co/> (Consulté le 16 février 2024).
3. **OpenTable.** (s. d.). *OpenTable est un système de réservation de restaurant en ligne populaire qui permet aux clients de réserver une table dans de nombreux restaurants. Il offre également des fonctionnalités de gestion des tables et des réservations pour les propriétaires de restaurants.* <https://os.resy.com/portal/> (Consulté le 17 février 2024).
4. **Tock.** (s. d.). *Tock est un système de réservation de restaurant en ligne qui propose des fonctionnalités avancées telles que la vente de billets pour des événements spéciaux et des menus prépayés. Il offre également des fonctionnalités de gestion des tables et des réservations.* <https://www.exploretock.com/> (Consulté le 19 février 2024).
5. **WISK.** (s. d.). *Restaurant analytics: Top metrics to track and optimize to increase profits.* <https://www.wisk.ai/features/restaurant-analytics-software> (Consulté le 19 février 2024).
6. **Gestion Restaurant.** (s. d.). *Logiciels et outils de gestion de stocks, caisse.* <https://es.wisk.ai/features/software-de-an%C3%A1lisis-para-restaurantes> (Consulté le 20 février 2024).
7. **Restaurant Industry Tech Trends.** (s. d.). *Restaurant Industry Tech Trends in 2024.* <https://www.wisk.ai/pos/restaurant-manager> (Consulté le 21 février 2024).
8. **Lightspeed.** (s. d.). *8 Restaurant Technology Trends Shaping Businesses in 2024.* <https://www.g2.com/products/wisk/reviews> (Consulté le 21 février 2024).
9. **Technology In Restaurants.** (s. d.). *Systems and Trends Shaping the Market.* <https://fr.wisk.ai/features/logiciel-danalyse-des-restaurants> (Consulté le 21 février 2024).

