

République Algérienne Démocratique et Populaire
Ministre de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira-Bejaia

Faculté des Sciences Exactes
Département Informatique



En vue de l'obtention du diplôme de master en informatique
Option : Génie logiciel

T H È M E

**Conception et Réalisation d'une application
desktop pour la gestion de la restauration**

Présenté par:

M^{lle} IDIRI Sylia

M^{lle} LAMARI Naima

Devant le jury composé de :

Président	SELLAMI Lynda	Maitre de Conférences	Université de Bejaïa
Rapporteur	SELLAMI Khaled	Maitre de Conférences	Université de Bejaïa
Examineur	SALHI Nadir	Maitre Assistant A	Université de Bejaïa

Année universitaire : 2022/2023

Remerciements

Nous tenons à témoigner notre profonde gratitude et notre sincère reconnaissance à DIEU tout puissant, qui nous a accompagnés et bénis par sa volonté tout au long de cette période.

Nous exprimons nos remerciements les plus chaleureux à notre promoteur, M K. SELLAMI, pour sa présence continue et son encouragement tout au long de ce travail. Sa bienveillance et son soutien ont été inestimables.

Nous souhaitons également exprimer notre reconnaissance au personnel de l'entreprise GOURAYA FROID, qui nous a apporté une aide précieuse en répondant à chacun de nos appels à l'aide. Leur soutien inconditionnel a été d'une valeur inestimable.

Nos remerciements s'adressent également aux membres du jury, devant lesquels nous avons eu l'honneur de présenter notre travail. Leur lecture attentive de ce mémoire et leur jugement éclairé nous ont été d'une grande aide.

Enfin, nous tenons à réserver une place spéciale pour remercier sincèrement toutes les personnes qui, de près ou de loin, nous ont apporté leur aide et leur encouragement dans la réalisation de ce modeste travail. Votre soutien a été précieux et nous en sommes extrêmement reconnaissants.

Dédicaces

Je dédie ce modeste travail à :

À mes chers parents, qui m'ont prodigué un soutien inébranlable et une confiance sans faille. Que Dieu vous protège et vous garde à mes côtés.

À mes précieuses sœurs, qui ont été présentes à chaque étape de ma vie et ont été une source constante de soutien et d'inspiration.

À mes adorables amies, pour votre fidélité indéfectible et votre soutien inconditionnel. Votre présence a été un réconfort et une source de motivation tout au long de ce parcours.

Et enfin, à tous mes amis, avec qui j'ai partagé des moments de joie et de bonheur. Votre amitié sincère et votre soutien ont été des éléments essentiels de mon épanouissement personnel et de la réussite de ce travail.

Que cette dédicace témoigne de ma reconnaissance éternelle envers chacun d'entre vous. Merci d'avoir fait partie de cette aventure et d'avoir enrichi ma vie de votre présence précieuse.

LAMARI Naima

Je dédie ce modeste travail à :

À mes chers parents, Pour tout l'amour et le soutien inconditionnel dont vous m'avez entouré, je dédie ce travail. Je ferai de mon mieux pour être une source de fierté pour vous et réaliser vos vœux et prières quotidiennes. Je vous aime profondément.

À mes chères sœurs et frère, Je vous dédie ce travail en vous souhaitant un avenir radieux, empli de bonheur et de succès. Que Dieu vous préserve et vous accorde une vie en bonne santé et de longue durée.

À mes chères amies, En témoignage de notre amitié indéfectible et des précieux souvenirs que nous avons partagés, je vous dédie ce travail. Je vous souhaite une vie comblée de santé et de bonheur.

Que cette dédicace exprime mon amour, ma gratitude et mon soutien inconditionnels envers vous tous. Votre présence dans ma vie est une bénédiction, et je suis profondément reconnaissant de vous avoir à mes côtés. Que chaque page de ce travail témoigne de notre lien précieux et de notre amitié éternelle.

IDIRI Sylia

TABLE DES MATIERES

INTRODUCTION GENERALE	1
CHAPITRE I	3
I.1. Introduction.....	3
I.2. Présentation de l'organisme d'accueil	4
I.2.1. Historique de Gouraya Froid	4
I.2.2. L'organigramme général	4
I.2.3. Missions des différents services.....	6
I.2.3.1. Service commercial.....	6
I.2.3.2. Service des méthodes.....	6
I.2.3.3. Service de production.....	6
I.2.3.4. Service de stock	6
I.3. Cadre d'étude.....	7
I.3.1. Présentation de sujet	7
I.3.2. Description du restaurant	7
I.3.3. Organigramme.....	7
I.3.3.1. Directeur du restaurant.....	8
I.3.3.2. Cuisinier	8
I.3.3.3. Serveurs	8
I.3.3.4. Caissier.....	8
I.4. Etude des postes de travail	9
I.4.1. Description	9
I.4.2. Etude du poste de travail N°01.....	9
I.4.3. Etude du poste de travail N°02.....	10
I.4.4. Etude du poste de travail N°03.....	11
I.4.5. Etude du poste de travail N°04.....	12
I.5. Etude des supports d'information.....	13
I.6. Etude des fichiers	18
I.7. La codification existante.....	20
I.8. Graphe des flux d'information	21
I.8.1. Définition	21
I.8.2. Tableau explicatif du graphe des flux.....	22

I.9.	Critique de l'existant et Objectifs	23
I.9.1.	Critique de l'existant	23
I.9.2.	Les Objectifs	23
I.10.	Conclusion	25
CHAPITRE II	26
II.1.	Introduction.....	26
II.2.	Présentation de l'UML.....	26
II.2.1.	Définition	26
II.2.2.	Diagrammes UML.....	26
II.3.	Présentation du processus de développement.....	27
II.3.1.	Définition	27
II.3.2.	Phases du processus unifié.....	28
II.3.3.	Les activités du processus unifié	29
II.4.	Diagramme de cas d'utilisation	29
II.4.1.	Représentation des diagrammes de cas d'utilisation	31
II.4.1.1.	Diagramme de cas d'utilisation associé à l'utilisateur	31
II.4.1.2.	Diagramme de cas d'utilisation associé au « Caissier »	31
II.4.1.3.	Diagramme de cas d'utilisation associé au « Directeur du restaurant »	32
II.4.1.4.	Diagramme de cas d'utilisation global	33
II.4.2.	Description textuelle des cas d'utilisation.....	35
II.5.	Présentation du diagramme de séquence	42
II.5.1.	Éléments de base d'un diagramme de séquence.....	42
II.5.1.1.	Diagramme de séquence du cas d'utilisation « S'authentifier»	44
II.5.1.2.	Diagramme de séquence du cas d'utilisation « Consulter les statistiques»	45
II.5.1.3.	Diagramme de séquence du cas d'utilisation « Gérer les catégories»	46
II.5.1.4.	Diagramme de séquence du cas d'utilisation « Gérer le Menu »	47
II.5.1.5.	Diagramme de séquence du cas d'utilisation « Définir les paramètres»	48
II.5.1.6.	Diagramme de séquence des cas d'utilisation « Etablir les commandes et les facture »	49
II.6.	Présentation du diagramme de classes.....	50
II.6.1.	Éléments de base d'un diagramme de classes	50
II.7.	Le modèle relationnel.....	53
II.7.1.	Les règles de passage au modèle relationnel.....	53
II.8.	Conclusion	56

CHAPITRE III	57
III.1. Introduction.....	57
III.2. Les outils de développement	57
III.2.1. Présentation du WampServer	57
III.2.2. Les outils principaux du WampServer	58
III.2.2.1. Présentation du MySQL-Server	58
III.2.2.2. Présentation du phpMyAdmin.....	58
III.2.3. Présentation du JDBC	58
III.2.4. Java Développement Kit (JDK).....	58
III.2.5. Présentation de Scenebuilder	59
III.2.6. Présentation de NetBeans.....	59
III.3. Les langages de programmation	59
III.3.1. Présentation du langage JAVA.....	59
III.3.1.1. Présentation de JavaFx.....	60
III.3.2. Présentation de FXML	60
III.3.3. Présentation du CSS	60
III.3.4. Présentation du SQL.....	61
III.4. Présentation du concept MVC.....	61
III.5. Présentation des interfaces de l'application.....	62
III.5.1. Interface d'Authentification	62
III.5.2. Interface Accueil.....	63
III.5.3. L'interface de gestion	63
III.5.4. L'interface des Commandes	64
III.5.5. L'interface des Paramètres.....	65
III.6. Conclusion	66
CONCLUSION GENERALE	67
BIBLIOGRAPHIE.....	69

LISTE DES FIGURES

Figure1 : Organigramme de Gouraya Froid.....	5
Figure 2: Organigramme de restaurant.....	8
Figure 3: Codification du numéro de bon de commande	20
Figure 4: Codification du numéro de la facture	20
Figure 5: Codification du numéro de la table.....	20
Figure 6: Codification du numéro de la cuisine.....	21
Figure 7: Graphe des flux d'information de restaurant.....	22
Figure 8: les diagrammes UML.....	27
Figure 9: Diagramme de cas d'utilisation associé à l'utilisateur.....	31
Figure 10:Diagramme de cas d'utilisation associé au Caissier	31
Figure 11: Diagramme de cas d'utilisation associé au Directeur	32
Figure 12: Diagramme de cas d'utilisation global	33
Figure 13: Diagramme de séquence du cas d'utilisation S'authentifier.....	44
Figure 14: Diagramme de séquence du cas d'utilisation Consulter les statistiques	45
Figure 15: Diagramme de séquence du cas d'utilisation Gérer les catégories	46
Figure 16: Diagramme de séquence du cas d'utilisation Gérer le menu	47
Figure 17: Diagramme de séquence du cas d'utilisation Définir les paramètres.....	48
Figure 18: Diagramme de séquence des cas d'utilisation Etablir les commandes et les factures	49
Figure 19:Diagramme de classes	52
Figure 20: Interface d'authentification	62
Figure 21: Interface d'accueil	63
Figure 22: Interface de gestion de Menu	64
Figure 23: Interface des Commande [1].....	64
Figure 24: Interface des commandes (2).....	65
Figure 25: Interface des paramètres	65

LISTE DES TABLEAUX

Tableau 1: Représente les tâches exécutées par le poste N° 01.....	9
Tableau 2: Représente les documents provenant au poste N° 01.....	10
Tableau 3: Représente les documents diffusés par le poste N° 01.....	10
Tableau 4: Représente les tâches exécutées par le poste N° 02.....	10
Tableau 5: Représente les documents provenant au poste N° 02.....	10
Tableau 6: Représente les documents diffusés par le poste N° 02.....	11
Tableau 7: Représente les tâches exécutées par le poste N° 03.....	11
Tableau 8: Représente les documents provenant au poste N° 03.....	11
Tableau 9: Représente les documents diffusés par le poste N° 03.....	11
Tableau 10: Représente les tâches exécutées par le poste N° 04.....	12
Tableau 11: Représente les documents provenant au poste N° 04.....	12
Tableau 12: Représente les documents diffusés par le poste N° 04.....	12
Tableau 13: Fiche d'étude de la fiche de Menu	13
Tableau 14: Fiche d'étude de la fiche technique de préparation	14
Tableau 15: Fiche d'étude de bon de commande.....	15
Tableau 16: Fiche d'étude de la facture	16
Tableau 17: Fiche d'étude de la fiche de paie.....	17
Tableau 18: Etude du fichier de personnel	18
Tableau 19: Etude du fichier de stock	19
Tableau 20: Tableau explicatif du graphe des flux.....	22
Tableau 21: Les cas d'utilisation du système	30
Tableau 22: Description du cas d'utilisation s'authentifier.....	35
Tableau 23: Description du cas d'utilisation Consulter les statistiques	36
Tableau 24: Description du cas d'utilisation Gérer les catégories	38
Tableau 25: Description du cas d'utilisation Gérer le Menu	39
Tableau 26: Description du cas d'utilisation Définir les paramètres.....	40
Tableau 27: Description des cas d'utilisation Etablir les commandes et les factures.....	41

LISTE DES ABREVIATIONS

- **TIC :** **Technologies de l'Information et de Communication**
- **UML :** **Unified Modeling Language**
- **UP :** **Unified Process**
- **OMT :** **Object Management Group**
- **MySQL :** **My Structured Query Language**
- **SQL :** **Structured Query Language**
- **JDBC :** **Java Database Connectivity**
- **PHP :** **Pre Hypertext Processor.**
- **HTML :** **Hypertext Markup Language.**
- **JDK :** **Java Développement Kit.**
- **JRE :** **Java Runtime Environnement.**
- **EDI :** **Environnement de Développement Intégré en français.**
- **XML :** **Extensible Markup Language.**
- **CSS :** **Cascading Style Sheets.**
- **MVC :** **Model-View-Controller.**
- **OOSE :** **Object Oriented Software Engineering**

INTRODUCTION GENERALE

Actuellement, les technologies de l'information et de communication (TIC) se sont largement répandues dans le monde entier et dans des secteurs d'activités différents, ils ont grandement affecté la vie humaine et provoqué un changement radical. La restauration est l'un des secteurs économiques dont la croissance est la plus rapide dans le monde et s'est vue obligé de s'aligner à cette tendance pour des raisons concurrentielle et évolutives.

L'avancement rapide des technologies et des logiciels a ouvert de nouvelles perspectives pour optimiser l'efficacité opérationnelle, la productivité et la rentabilité des entreprises évoluant dans le secteur de la restauration.

C'est dans ce contexte que notre projet de recherche a pris forme, en exploitant nos compétences en programmation et en technologies de développement. L'objectif primordial de notre étude était de concevoir une application de gestion de restauration sur mesure, simplifiant ainsi les tâches administratives complexes, tout en offrant des fonctionnalités avancées pour la gestion des stocks, des commandes, de la facturation, du personnel et la génération de rapports analytiques.

Dans le cadre de notre stage au sein de l'entreprise Gouraya Froid, nous avons eu l'opportunité de collaborer étroitement avec une équipe de professionnels passionnés de la restauration. Cette immersion au cœur de leur environnement de travail nous a permis d'appréhender les défis spécifiques auxquels ils font face au quotidien, tout en identifiant les domaines où notre application de gestion de restauration pouvait apporter une réelle valeur ajoutée.

Afin de garantir le succès de notre projet, nous allons adopter une approche de développement basée sur le langage de modélisation UML qui est le processus unifié UP, et avons exploité l'environnement de développement NetBeans. Cette approche itérative nous a permis de progresser rapidement dans le développement de l'application, en tenant compte des retours et des besoins exprimés par l'équipe interne de l'entreprise Gouraya Froid.

Bien que nous n'ayons pas été directement en contact avec les clients finaux, notre principal interlocuteur était l'entreprise elle-même. Nous avons travaillé en étroite collaboration avec une équipe interne dédiée à l'analyse des besoins et à la validation des fonctionnalités. Leur expertise dans le domaine de la restauration a été essentielle pour guider nos décisions de développement, afin de nous assurer que notre application répondrait de manière précise aux exigences spécifiques du secteur.

Le choix des technologies appropriées a également joué un rôle déterminant dans le développement d'une application de bureau solide et performante. En utilisant (NetBeans, Wamp, Scenenbuilder), nous avons pu bénéficier des avantages offerts par ces outils bien établis, notamment leur flexibilité, leur évolutivité et leur adéquation avec les besoins spécifiques de notre application de gestion de restauration.

Ce projet de fin d'études revêt une importance capitale, car il nous a permis de conjuguer nos compétences en informatique avec notre passion pour le domaine de la restauration. Grâce à cette expérience, nous avons pu développer nos aptitudes techniques, acquérir une compréhension approfondie des besoins du secteur et apporter une contribution concrète à l'amélioration des opérations au sein de l'entreprise Gouraya Froid.

Nous avons organisé notre mémoire en trois chapitres distincts, chacun abordant des aspects spécifiques de notre projet de gestion de restauration.

Le premier chapitre est consacré à la présentation de l'organisme d'accueil, dans notre cas l'entreprise Gouraya Froid, ainsi que des problématiques rencontrées dans le domaine de la restauration. Nous avons également défini nos objectifs clés pour ce projet, en mettant en évidence les besoins spécifiques de l'entreprise et les défis auxquels nous devons faire face.

Le deuxième chapitre se concentre sur la conception de notre application. Nous avons utilisé des diagrammes, tels que les diagrammes de cas d'utilisation, les diagrammes de classes et les diagrammes de séquence, pour modéliser les différentes fonctionnalités et interactions de notre application. De plus, nous avons choisi un cycle de vie adapté, tel que le cycle de vie UP, pour guider notre processus de conception, en nous assurant d'itérer régulièrement et de prendre en compte les retours et les besoins des utilisateurs.

Dans le troisième chapitre, nous avons mis en avant les interfaces utilisateur de notre application et expliqué les choix de logiciels que nous avons faits pour l'implémentation. Nous avons présenté les différentes vues de l'application, en mettant l'accent sur leur convivialité et leur ergonomie. De plus, nous avons décrit les outils et les technologies spécifiques que nous avons utilisés pour le développement de notre application de gestion de restauration, tels que NetBeans pour l'environnement de développement et les bibliothèques graphiques pour la création des interfaces.

Cette structure en trois chapitres permet une approche complète et bien organisée de notre projet, en couvrant les aspects liés à l'organisme d'accueil, à la conception et à l'implémentation de l'application de gestion de restauration.

CHAPITRE I

ETUDE DE L'EXISTANT

I.1. Introduction

L'étude de l'existant est le point de passage obligatoire qui matérialise le premier contact des concepteurs avec un domaine d'expertise non maîtrisé.

Avant toute conception, une vue claire des besoins doit être établie, les objectifs poursuivis doivent être connus, le double but visé par cette étape est donc la connaissance et la critique des besoins. L'étude de l'existant comporte l'étude :

- Des postes du travail.
- Des documents manipulés par ces derniers.
- Des différentes procédures exécutées par ceux-ci.
- Du cheminement d'information.
- Des règles d'organisation et de gestion.

Les objectifs de cette étape est de prendre connaissance dans le détail, du domaine dont l'entreprise souhaite améliorer son fonctionnement.

Le résultat de cette étape sera un bilan sur la situation actuelle dans lequel sont énumérées toutes les anomalies constatées et les solutions à entreprendre.

I.2. Présentation de l'organisme d'accueil

I.2.1. Historique de Gouraya Froid

C'est une entreprise algérienne industrielle créée à Béjaia en 1998 par Mr BRAHMI SAID, spécialisée dans la fabrication de matériel frigorifique, conception et aménagement, chambres froides et équipements professionnels destinés à l'agencement des hôtels, restaurants, cafétérias, snacks, pâtisseries, boucheries, petites et grandes surfaces ainsi que toutes autres structures alimentaires ou industrielles.

Est une entreprise algérienne de droit privé au capital social de 10 000 000.00 DA, occupe une superficie de 3000 m² dont l'effectif s'élève à plus de 100 spécialistes répartis dans différents services. (1)

Pour mener à bien sa mission, le directeur de l'entreprise doit choisir minutieusement les éléments à affecter pour chaque service, de sorte à avoir un organigramme équilibré, selon les besoins de gestion de son portefeuille clients. Pour cela, l'entreprise Gouraya froid est répartie suivant un organigramme bien défini pour la bonne continuité de ses prestations.

I.2.2. L'organigramme général

La figure [1] qui suit, illustre l'organigramme de Gouraya froid :

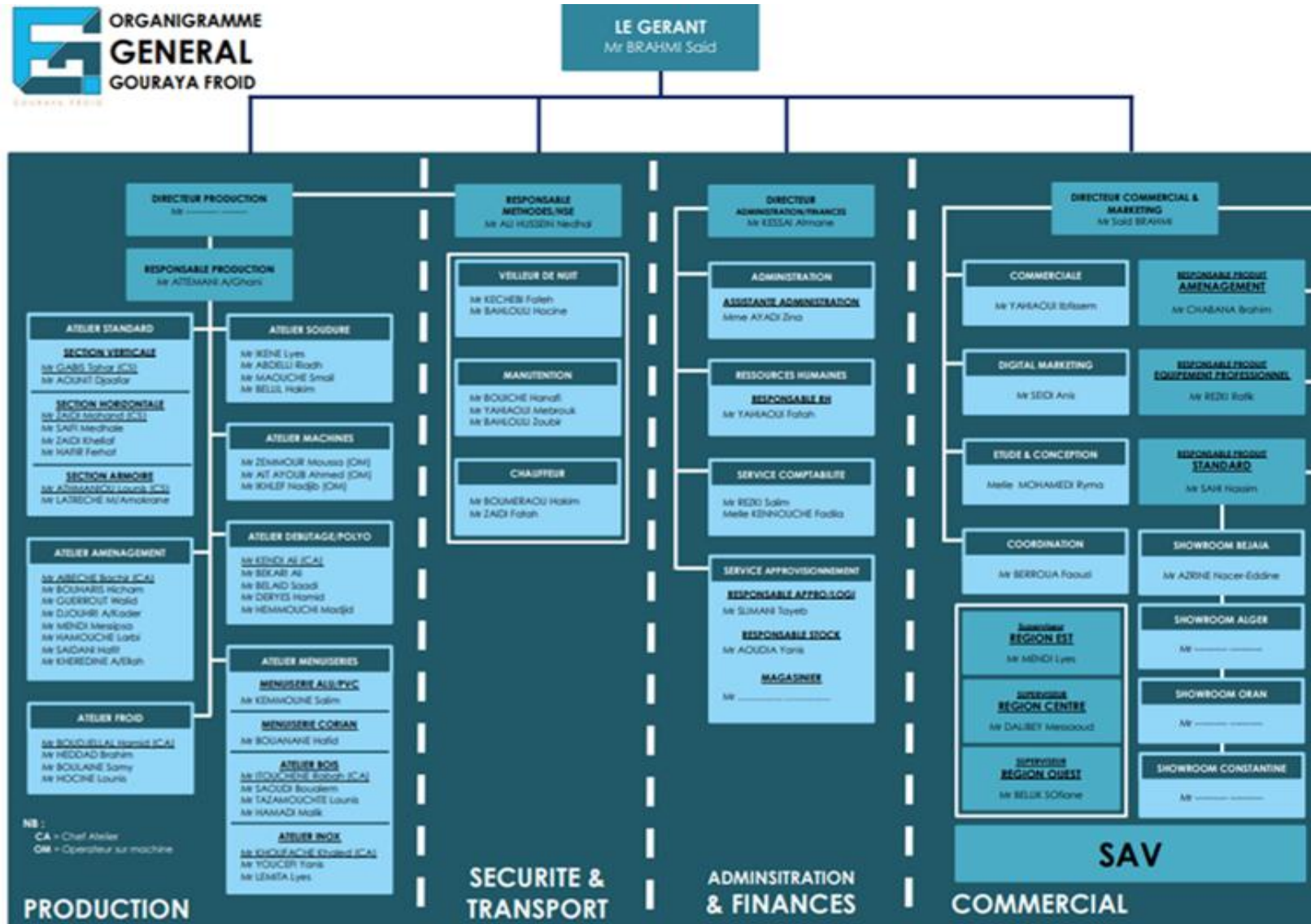


Figure1 : Organigramme de Gouraya Froid

I.2.3. Missions des différents services

I.2.3.1. Service commercial

Constitué de trois personnes travaillant sous la direction d'un chef de service, accomplit les tâches suivantes :

- S'occupe de la mise en vente.
- La gestion de l'administration des ventes.
- La publicité autour d'un produit donné.

I.2.3.2. Service des méthodes

C'est un département dont la mission est d'améliorer la productivité et l'efficacité de la production. Il travaille en étroite collaboration avec la ligne de production et le bureau d'études pour concevoir et fournir les outils nécessaires à la production, améliorer les conditions de travail et analyser les coûts de production.

I.2.3.3. Service de production

Responsable de diverses tâches liées à la production de produits proprement dite des produits finaux parmi ces tâches :

- Assure le bon fonctionnement des équipements.
- Effectue des contrôles réguliers de la qualité tout au long du processus de fabrication.
- Elaboration des rapports techniques des études.

I.2.3.4. Service de stock

Gérer les stocks efficacement est essentiel pour maximiser la rentabilité, que ce soit pour les matières premières nécessaires à la production ou pour les produits finis et semi-finis prêts à être expédiés, une gestion appropriée des stocks.

I.3. Cadre d'étude

I.3.1. Présentation de sujet

Notre projet ambitieux consiste à établir une collaboration en tant que stagiaires en sous-traitance avec le service informatique de Gouraya Froid qui est une branche spécialisée dans la production d'aménagements au sein du secteur des services commerciaux. Dans le cadre de cette collaboration, nous nous engageons à développer une application informatique sur mesure, dédiée à la gestion de restaurants.

Notre approche proactive, combinant des visites sur le terrain, des rencontres avec le personnel des établissements de restauration et des recherches approfondies, nous a fourni une vision claire de l'existant. Cette connaissance approfondie nous a servi de base solide pour concevoir et développer une application informatique de gestion de restaurant qui répondra aux attentes et aux besoins uniques de Gouraya Froid.

I.3.2. Description du restaurant

Un restaurant est bien plus qu'un simple lieu où l'on mange. C'est un endroit où les gens viennent chercher une expérience culinaire unique, une atmosphère agréable et un service attentif. Chaque restaurant a sa propre identité et offre une expérience gastronomique distinctive, adaptée aux préférences et aux besoins de ses clients.

Il existe différents organigrammes selon la taille de l'établissement : petit restaurant, grand restaurant ou restaurant gastronomique.

Dans notre projet nous sommes intéressés aux petits restaurants.

I.3.3. Organigramme

La figure suivante représente l'organigramme de restaurant sur lequel s'est basée notre étude :

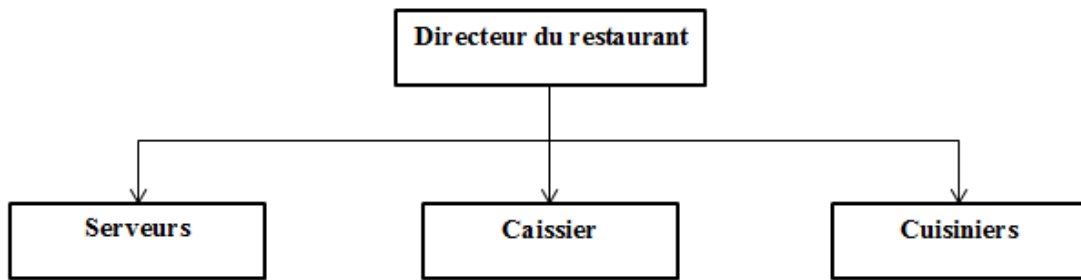


Figure 2: Organigramme de restaurant

I.3.3.1. Directeur du restaurant

- Il applique les décisions prises dans les différents conseils d'administration.
- Coordonne les travaux entre les différents services .
- Gère et dirige le personnel .
- Gère le stock , les revenus et les ventes .

I.3.3.2. Cuisinier

- Elabore le menu.
- Préparer les plats.
- Etablir les fiches de préparation.

I.3.3.3. Serveurs

- Dressent les tables.
- Prennent les commandes des clients.
- Servent les clients et encaisse leurs règlement.
- Néttoient et débarassent les tables .

I.3.3.4. Caissier

- Se charge de la caisse.

I.4. Etude des postes de travail

I.4.1. Description

Le poste de travail est un emplacement où s'effectue une phase dans l'exécution d'un travail, ou bien est un centre d'activité comprenant tout ce qui est nécessaire (Machine, outillage) à l'exécution d'un travail.

Liste des postes de travail à étudier :

1. Directeur du restaurant.
2. Caissier.
3. Serveur.
4. Cuisinier.

I.4.2. Etude du poste de travail N°01

a. Description

Le responsable du poste : Directeur du restaurant

Service de rattachement : La direction.

Rôle : Gestion du restaurant.

Effectif : 01.

Matériel utilisé : Pc + imprimante + scanner.

b. Les tâches exécutées

N° Tâche	Désignation	Fréquence
01	Définir les tables et les salles du restaurant.	Aléatoire
02	Définir la Fiche de Menu et ses catégories.	Aléatoire
03	Définir les utilisateurs du système et leurs rôles.	Aléatoire
04	Gérer les budgets du restaurant (revenus et dépenses).	Chaque jour

Tableau 1: Représente les tâches exécutées par le poste N° 01

c. Documents provenant à ce poste

Désignation	Fréquence	Structure Emettrice	Nbr exemplaire
Fiche de Menu	Aléatoire	Cuisine	01

Tableau 2: Représente les documents provenant au poste N° 01

d. Documents diffusés par ce poste

Désignation	Fréquence	Nbr exemplaire	Structure Réceptrice
Fiches de paie	Fin de mois	01	Employés
Fiche de Menu	Aléatoire	01	Cuisine

Tableau 3: Représente les documents diffusés par le poste N° 01

I.4.3. Etude du poste de travail N°02**a. Description**

Le responsable du poste : Serveur.

Service de rattachement : Restauration.

Rôle : S'occupe des clients.

Effectif : 04.

b. Les tâches exécutées

N° Tâche	Désignation	Fréquence
01	Présenter le menu.	Chaque jour
02	Prendre les commandes des clients.	Chaque jour
03	Servir les plats aux clients.	Chaque jour
04	Nettoyer les tables.	Chaque jour

Tableau 4: Représente les tâches exécutées par le poste N° 02

c. Documents provenant à ce poste

Désignation	Fréquence	Structure Emettrice	Nbr exemplaire
Fiche de menu	Chaque jour	Cuisine	01
La facture	Chaque commande	Caisse	01

Tableau 5: Représente les documents provenant au poste N° 02

d. Documents diffusés par ce poste

Désignation	Fréquence	Nbr exemplaire	Structure Réceptrice
Fiche de Menu	Chaque commande	01	Client.
Bon de commande	Chaque commande	01	Cuisine, Caisse.

Tableau 6: Représente les documents diffusés par le poste N° 02

I.4.4. Etude du poste de travail N°03**a. Description**

Le responsable du poste : Cuisinier

Service de rattachement : Cuisine.

Rôle : Gérer la cuisine.

Effectif : 02.

b. Les tâches exécutées

N° Tâche	Désignation	Fréquence
01	Planifier et préparer le menu.	Aléatoire
02	Préparer les plats commandés.	Chaque commande
03	Suivre les normes d'hygiène et de sécurité alimentaire.	Chaque jour

Tableau 7: Représente les tâches exécutées par le poste N° 03

c. Documents provenant à ce poste

Désignation	Fréquence	Structure Emettrice	Nbr exemplaire
La Fiche de Menu	Aléatoire	Directeur	01
Bon de commande	Chaque commande	Serveur	01

Tableau 8: Représente les documents provenant au poste N° 03

d. Documents diffusés par ce poste

Désignation	Fréquence	Nbr exemplaire	Structure Réceptrice
La fiche de menu	Aléatoire	01	Directeur

Tableau 9: Représente les documents diffusés par le poste N° 03

I.4.5. Etude du poste de travail N°04

a. Description

Le responsable du poste : Caissier

Service de rattachement : Caisse.

Rôle : Gérer le paiement.

Effectif : 01.

b. Les tâches exécutées

N° Tâche	Désignation	Fréquence
01	Calculer les factures des clients.	Chaque commande
02	Assurer la propreté et l'organisation de la zone de la caisse.	Chaque jour

Tableau 10: Représente les tâches exécutées par le poste N° 04

c. Documents provenant à ce poste

Désignation	Fréquence	Structure Emettrice	Nbr exemplaire
Bon de commande	Chaque commande	Serveur	01

Tableau 11: Représente les documents provenant au poste N° 04

d. Documents diffusés par ce poste

Désignation	Fréquence	Nbr exemplaire	Structure Réceptrice
Facture	Chaque commande	01	Client
Reçu de paiement	Chaque commande	01	Client
Rapports de fin de journée	Chaque jour	01	Directeur

Tableau 12: Représente les documents diffusés par le poste N° 04

I.5. Etude des supports d'information

- **N** : Numérique
- **AN** : Alphaumérique
- **AB** : Alphabétiques
- **Etude de la fiche de Menu**

Fiche d'étude des supports d'information		
Nom du support :Fiche de Menu Nature du support :Externe Objet du support : Contient la liste des plats et des boissons Origine du support : Cuisine Destination : Client Nbre d'exemplaires :01		
Les Informations Manipulées		
Données et Désignation	Type	Longueur
Nom du restaurant	AN	25
L'adresse	AN	50
Le contact	AN	30
Nom des catégories	AB	30
Nom des plats	AB	30
Nom des boissons	AB	30
Type	AB	20
Prix	N	05
Taille	AB	06

Tableau 13: Fiche d'étude de la fiche de Menu

- Etude de la fiche technique de préparation

Fiche d'étude des supports d'information		
<p>Nom du support :Fiche technique de préparation. Nature du support :Interne . Objet du support : Détermine les ingrédients ,les quantités et étapes de préparation d'un plat . Origine du support : Cuisine. Destination : Cuisine . Nbre d'exemplaires :01.</p>		
Les Informations Manipulées		
Données et Désignation	Type	Longueur
Date	AN	25
Nom du plat	AB	50
Type	AB	30
Nom des ingrédients	AB	06
Quantité	N	10
Unité	AB	05
Etapes de réalisation	AN	02
Temps de préparation	AN	15
Temps de cuisson	AN	02
Les portions	N	06
La nutrition	AN	08
Les remarques	AN	30

Tableau 14: Fiche d'étude de la fiche technique de préparation

- **Fiche d' étude de bon de commande**

Fiche d'étude des supports d'information		
Nom du support :Bon de commande Nature du support :Interne Objet du support : Contient les plats et des boissons commandés Origine du support : Serveur Destination : Cuisine Nbre d'exemplaires :01		
Les Informations Manipulées		
Données et Désignation	Type	Longueur
Nom du restaurant	AN	25
L'adresse	AN	50
Le contact	AN	30
Le code	N	06
La date	AN	10
L'heure	AN	05
Numéro de table	N	02
Titre	AB	15
Quantité	N	02
Prix	N	05

Tableau 15: Fiche d'étude de bon de commande

- **Fiche d'étude de la facture**

Fiche d'étude des supports d'information		
<p>Nom du support :Facture . Nature du support :Externe. Objet du support : Détermine l'addition à payer par le client. Origine du support : Caisse. Destination : Client. Nbre d'exemplaires :01.</p>		
Les Informations Manipulées		
Données et Désignation	Type	Longueur
Nom du restaurant	AN	25
L'adresse	AN	50
Le contact	AN	30
Le code	N	06
La date	AN	10
L'heure	AN	05
Numéro de table	N	02
Titre	AB	15
Quantité	N	02
Prix	N	06
Prix total	N	08
Message	AB	30

Tableau 16: Fiche d'étude de la facture

- **Fiche d'étude de la fiche de paie**

Fiche d'étude des supports d'information		
Nom du support : Fiche de paie Nature du support : Externe Objet du support : Détermine le salaire des employés Origine du support : Directeur du restaurant Destination : Employés Nbre d'exemplaires: 01		
Les Informations Manipulées		
Données et Désignation	Type	Longueur
Nom du restaurant	AN	25
L'adresse du restaurant	AN	30
Nom d'employé	AB	10
Prénom d'employé	AB	10
Adresse d'employé	AN	25
Date de naissance d'employé	AN	10
Situation familiale	AB	10
Date	AN	15
Numéro de sécurité social	N	15
Matricule	N	12
Fonction	AB	10
Jours de travail	N	05
Rub	N	05
Libeller indemnités	AB	10
Base	N	20
Nombre / Taux	N	05
Gains	N	10
Retenues	N	10
Salaire de poste	N	08
Total	N	10
Net à payer	N	10

Tableau 17: Fiche d'étude de la fiche de paie

I.6. Etude des fichiers

- Etude du fichier de personnel

Nom de fichier : Fichier de personnel.		
Rôle : Contient des information sur les employés.		
Emplacement : Direction.		
Nom du champ	Taille	Type
Nom d'employé	10	AB
Prénom d'employé	10	AB
Adresse d'employé	20	AN
Date de naissance	10	AN
Adresse postale	10	AN
Numéro de téléphone	10	AN
Date d'embauche	10	AN
Date de fin de contrat	10	AN
Statut de l'employé	15	AB
Numéro de sécurité social	10	N
Rôle de l'employé	08	AB
Salaire	08	N
Heures de travail	03	AN
Niveau de formation	06	AN

Tableau 18: Etude du fichier de personnel

- **Etude du fichier de stock**

Nom de fichier : Fichier de stock.

Rôle : Contient des information sur la gestion des stocks.

Emplacement : Direction.

Nom du champ	Taille	Type
Date	10	AN
Code	04	AN
Emplacement	10	AB
Catégorie	10	AB
Produit	15	AB
Unité	10	AB
Prix (HT)	06	N
Nombre	04	N
Montant	07	N
Fournisseur	11	AB
Remarque	30	AN

Tableau 19: Etude du fichier de stock

I.7. La codification existante

Chaque étude informatique dans le centre de gestion conduit aux choix du système technique de codification qui permet de représenter les données sous forme abrégée à l'aide d'un certain nombre de symboles ou de codes, ce choix dépend des caractéristiques des données à représenter et du contexte d'utilisation.

Le but de cette technique est la simplicité, la facilité et la conception de l'élément codifié.

Voici quelques codifications existantes :

a. N° de Bon de commande

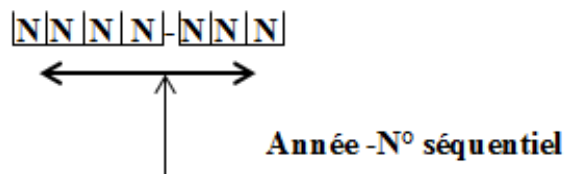


Figure 3: Codification du numéro de bon de commande

Exemple : 2023-008

b. N° de la facture

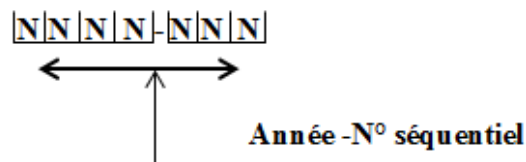


Figure 4: Codification du numéro de la facture

Exemple : 2023-014

c. N° de la table

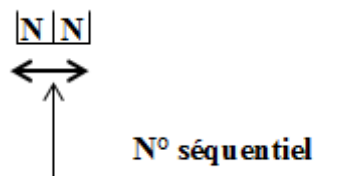


Figure 5: Codification du numéro de la table

Exemple : N°19

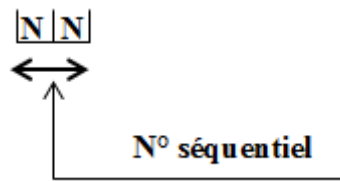
d. N° de la cuisine

Figure 6: Codification du numéro de la cuisine

Exemple : N°02**I.8. Graphe des flux d'information****I.8.1. Définition**

Le graphe des flux permet de mettre en évidence les flux d'information entre les différents acteurs du domaine établis ainsi qu'avec leur environnement.

- **Définition d'un acteur**

Un acteur représente une personne ou une entité qui effectue une action ou une activité dans le système. Les acteurs peuvent être internes ou externes à l'entreprise ou à l'organisation représentée par le diagramme. Les acteurs peuvent être représentés graphiquement sous la forme de boîtes ou de symboles spécifiques.

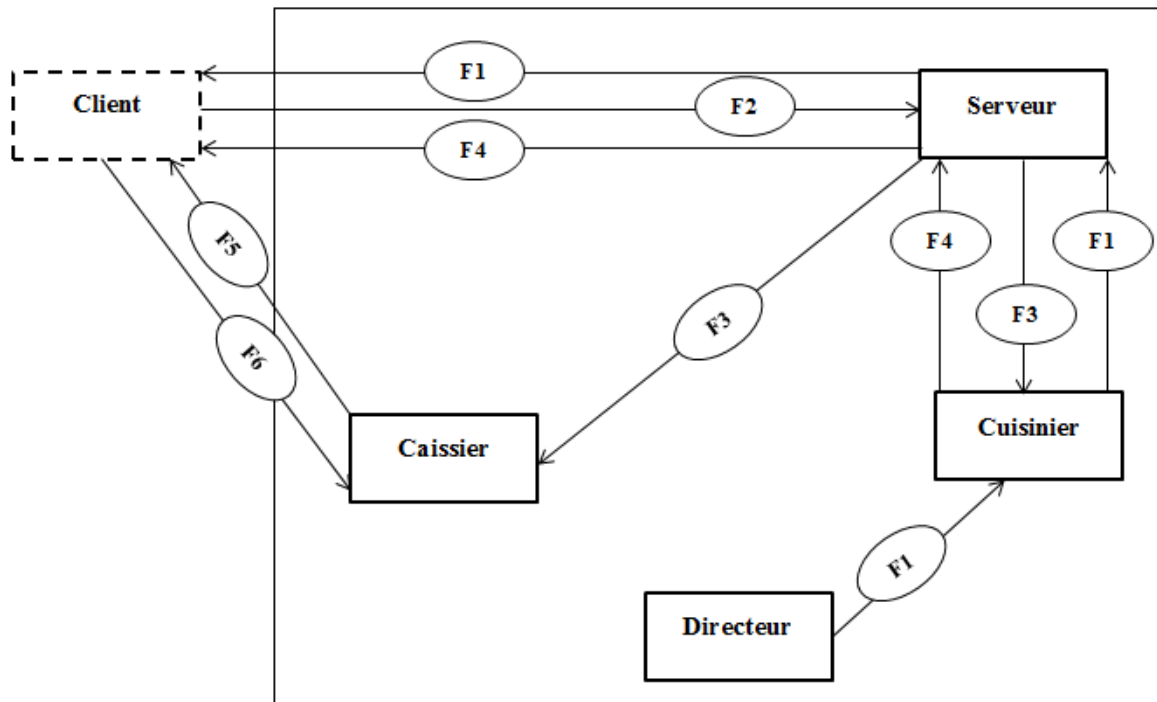


Figure 7: Graphe des flux d'information de restaurant

I.8.2. Tableau explicatif du graphe des flux

N° Flux	Désignation des Flux
F1	Fiche de Menu
F2	Commande
F3	Bon de commande
F4	Plats à servir
F5	Facture
F6	Paiement

Tableau 20: Tableau explicatif du graphe des flux

I.9. Critique de l'existant et Objectifs

I.9.1. Critique de l'existant

La restauration occupe une place importante et indispensable. Cependant, malgré son importance indéniable, l'industrie de la restauration est confrontée à d'énormes défis en matière de gestion, en particulier lorsqu'il s'agit de traiter les commandes, les factures et les statistiques de ventes en utilisant des méthodes manuelles traditionnelles.

Le fonctionnement manuel de la gestion dans le secteur de la restauration engendre un ensemble de problèmes qui impactent à la fois les propriétaires de restaurants et les clients. Parmi ces problèmes, on retrouve :

- Erreurs de saisie de commandes ou de calculs incorrects des factures qui peuvent entraîner des pertes financières et une insatisfaction de la clientèle.
- Lenteur des opérations , engendrant ainsi ; (1) des retards dans le service, (2) une augmentation du temps d'attente des clients et (3) une diminution du taux de la satisfaction globale.
- Difficulté de suivi de toutes les transactions, ce qui rend la gestion et l'analyse des données plus complexe, voire impossibles..
- Manque de visibilité et de transparence : Le fonctionnement manuel limite la visibilité des données en temps réel.
- L'absence d'automatisation dans la gestion des commandes, des factures et des statistiques de ventes augmente le risque d'erreurs, et limite les opportunités d'innovation et d'amélioration.

I.9.2. Les Objectifs

Face à ces défis majeurs dans la gestion de la restauration, notre objectif fondamental est de tirer parti des avantages offerts par les solutions informatiques et technologiques pour moderniser et optimiser la gestion de la restauration.

En adoptant un système informatisé de gestion des commandes, des factures et des statistiques de ventes qui permet de:

- Faciliter le suivi des transactions.
- Réduire le taux d'erreurs concernant les calculs.
- Accélérer les opérations et réduire le temps des différents traitements.
- Obtenir une vision plus claire et plus précise de l'activité commerciale.

Cette transformation numérique contribue à améliorer l'efficacité opérationnelle, la satisfaction client et la rentabilité globale du restaurant, plaçant ainsi notre établissement à l'avant-garde de l'industrie de la restauration moderne.

I.10. Conclusion

Ce chapitre marque une étape cruciale de notre travail, où nous avons procédé à une analyse approfondie de l'existant afin de mettre en évidence les faiblesses du système et d'identifier les critiques formulées à son encontre. Les données collectées seront maintenant analysées et traitées en profondeur dans le chapitre suivant afin de guider la réalisation de l'étude conceptuelle et de formuler des recommandations précises.

CHAPITRE II

CONCEPTION

II.1. Introduction

Dans le chapitre précédent nous avons effectué une étude des besoins nous permettant de recueillir l'ensemble des besoins pour faire face à notre étude conceptuelle et pour ce faire, nous avons choisi d'adopter l'approche orientée objet utilisant la méthode UML.

II.2. Présentation de l'UML

II.2.1. Définition

L'UML, ou Unified Modeling Language, est un langage de modélisation graphique et textuel utilisé pour comprendre, décrire, spécifier et documenter les besoins, les systèmes et les solutions logicielles. Il est le fruit de la fusion de trois méthodes de modélisation orientées objet couramment utilisées : OMT, Booch et OOSE. L'UML a été développé à partir de 1994 et a été officiellement standardisé en 1997 par l'OMG (Object Management Group).

Ce langage s'appuie sur les principes et les concepts de la technologie orientée objet et permet d'unifier les notations graphiques et les concepts de modélisation. En fournissant des notations adaptées aux différentes activités du processus de développement logiciel, l'UML facilite la communication et la compréhension entre les différentes parties prenantes d'un projet. (2)

II.2.2. Diagrammes UML

UML2 comprend treize diagrammes qui sont utilisés pour décrire un système. Ces diagrammes sont regroupés en deux grands ensembles :

- **Les diagrammes structurels** : Ils sont au nombre de six, ils incluent des représentations visuelles des éléments statiques du système, tels que les classes, les objets, les composants et les packages.

- **Les diagrammes de comportement** : Ils sont au nombre de sept, ils permettent de modéliser les aspects dynamiques du système, tels que les interactions entre les objets, les séquences d'actions, les états et les transitions. (3)

Ces différents diagrammes sont représentés dans le schéma suivant :

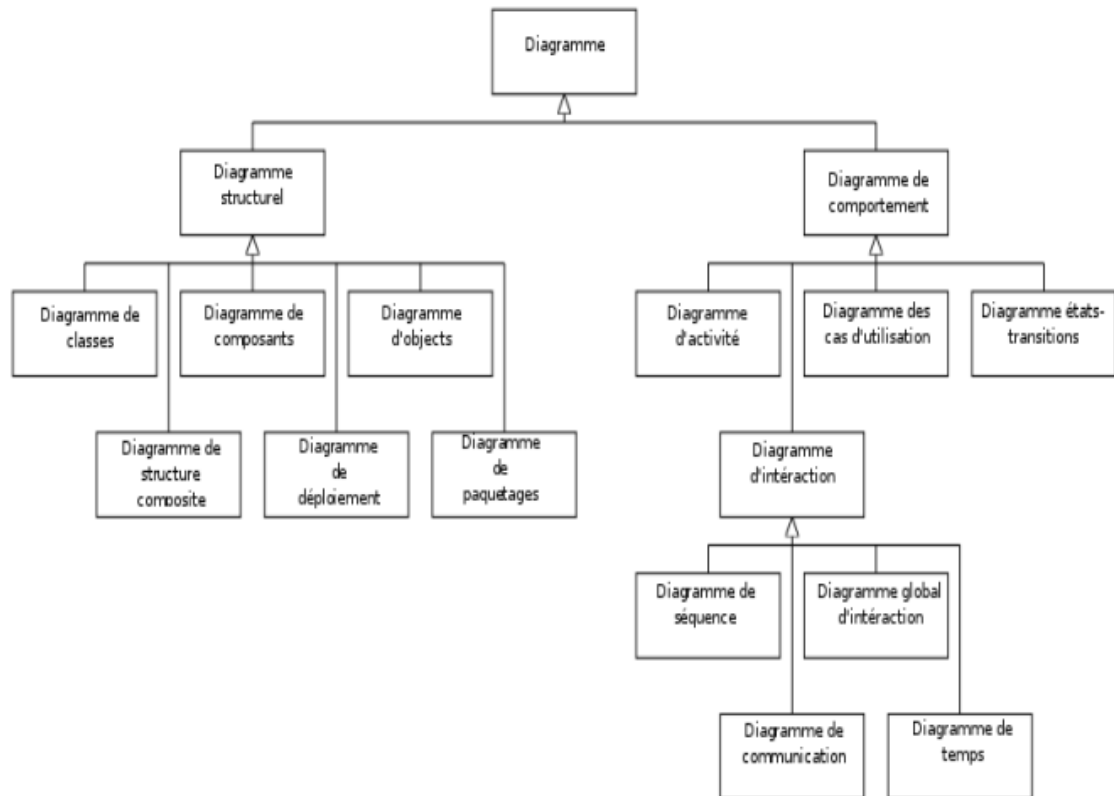


Figure 8: les diagrammes UML

II.3. Présentation du processus de développement

L'utilisation des diagrammes UML requiert l'adoption d'un processus de développement unifié pour garantir la gestion et le suivi complets de la mise en œuvre du système, depuis la collecte des besoins initiaux jusqu'à la livraison finale. Dans notre cas, nous avons opté pour le Processus Unifié (UP), qui propose une approche méthodologique structurée pour le développement logiciel.

II.3.1. Définition

Le Processus Unifié (UP) est un processus de développement logiciel associé à UML. Il met en œuvre les principes suivants (4) :

- **Processus piloté par les cas d'utilisation** : Le projet est mené en prenant en compte les besoins et les exigences des utilisateurs. Les cas d'utilisation permettent d'exprimer les interactions du système avec les utilisateurs, ce qui permet de capturer les besoins. Les cas d'utilisation jouent également un rôle structurant dans le développement et les tests du système.
- **Processus centré sur l'architecture** : Tout système complexe doit être décomposé en parties modulaires afin de faciliter la maintenance et l'évolution. Cette architecture, qu'elle soit fonctionnelle, logique, matérielle ou autre, doit être modélisée en UML plutôt que simplement documentée en texte.
- **Processus itératif et incrémental** : Le projet est découpé en itérations de courte durée, ce qui permet de suivre plus efficacement l'avancement global. À la fin de chaque itération, une partie exécutable du système final est produite de manière progressive.

II.3.2. Phases du processus unifié

Le Processus Unifié (UP) se compose généralement de quatre phases : (2)

- **Phase de création** :Vise à définir les objectifs et la portée du système à développer. Les principales activités comprennent l'identification des besoins des utilisateurs, l'évaluation de la faisabilité du projet, la définition de l'architecture initiale et l'établissement du plan de développement préliminaire.
- **Phase d'élaboration** :Dans cette phase, une analyse plus détaillée est réalisée pour comprendre et spécifier les besoins du système. Des itérations supplémentaires sont effectuées pour raffiner l'architecture, définir les cas d'utilisation et élaborer un plan de développement plus détaillé.
- **Phase de construction** : Cette phase consiste à finaliser l'analyse, la conception, l'implémentation et les tests du système. L'objectif principal est de transformer l'architecture de référence en un produit exécutable tout en veillant à maintenir son intégrité.
- **Phase de transition** : Cette phase marque la livraison du système au client pour qu'il puisse faire des essais et détecter d'éventuelles anomalies.

II.3.3. Les activités du processus unifié

Chaque phase du processus UP est composée d'une série d'activités spécifiques. Les activités du processus UP sont les suivantes (2) :

- **Expression des besoins** : Cette activité consiste à comprendre et à exprimer les besoins et les exigences du client, qu'ils soient fonctionnels ou non fonctionnels.
- **Analyse et Conception** : Cette activité consiste à effectuer une analyse approfondie des besoins du système et à concevoir son architecture. Cela implique l'utilisation de modèles UML pour représenter graphiquement les différents aspects du système.
- **Implémentation** : Dans cette activité, le système est mis en œuvre sous forme de composants, de bibliothèques et de fichiers. L'objectif est de planifier l'intégration de ces éléments.
- **Tests** : Les activités de test sont réalisées pour vérifier les résultats de l'implémentation de toutes les exigences et s'assurer de l'intégration correcte de tous les composants du logiciel. (5)

II.4. Diagramme de cas d'utilisation

Il représente les interactions entre les acteurs et le système, en mettant l'accent sur les fonctionnalités offertes par ce dernier. Il permet de comprendre les fonctionnalités du système, d'identifier les besoins des utilisateurs et de spécifier les exigences. (6)

- Les cas d'utilisation sont des actions que les acteurs peuvent effectuer dans le système, représentés par des ellipses.
- Les acteurs sont les utilisateurs ou systèmes externes, représentés par des icônes humaines ou des boîtes rectangulaires.
- Les cas d'utilisation sont reliés aux acteurs par des associations, montrant leur participation.

Dans le tableau ci-dessous nous illustrons les cas d'utilisation de notre système :

Acteurs	Cas d'utilisation
Utilisateur	Authentification.
Directeur du restaurant	Consulter les statistiques. Gérer le menu. Gérer les catégories de consommation. Gérer les tables. Gérer les employés. Définir les paramètres.
Caissier	Etablir les commandes. Etablir les factures.

Tableau 21: Les cas d'utilisation du système

II.4.1. Représentation des diagrammes de cas d'utilisation

II.4.1.1. Diagramme de cas d'utilisation associé à l'utilisateur

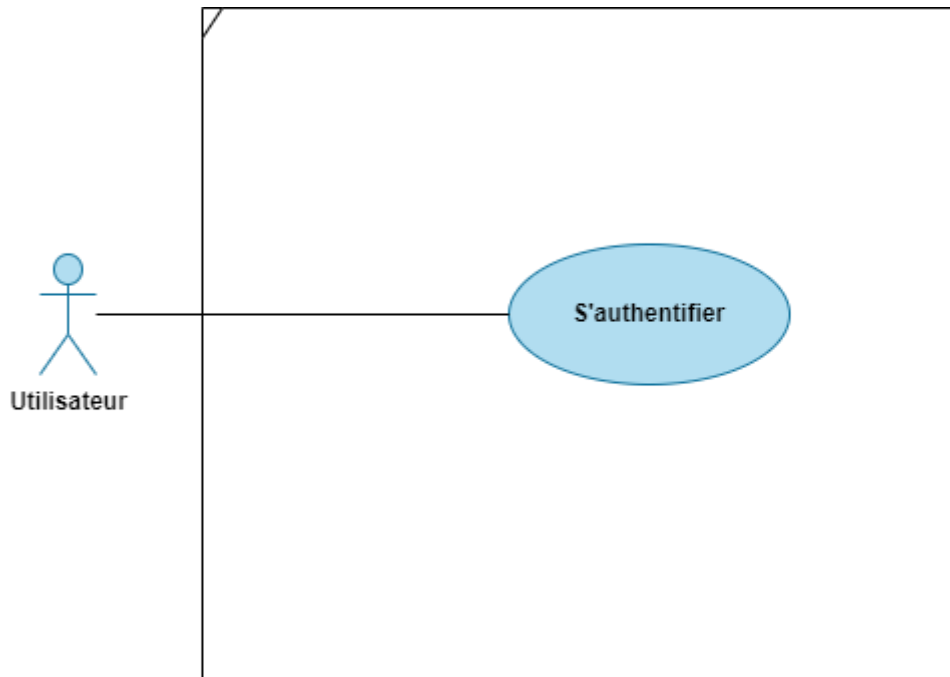


Figure 9: Diagramme de cas d'utilisation associé à l'utilisateur

II.4.1.2. Diagramme de cas d'utilisation associé au « Caissier »

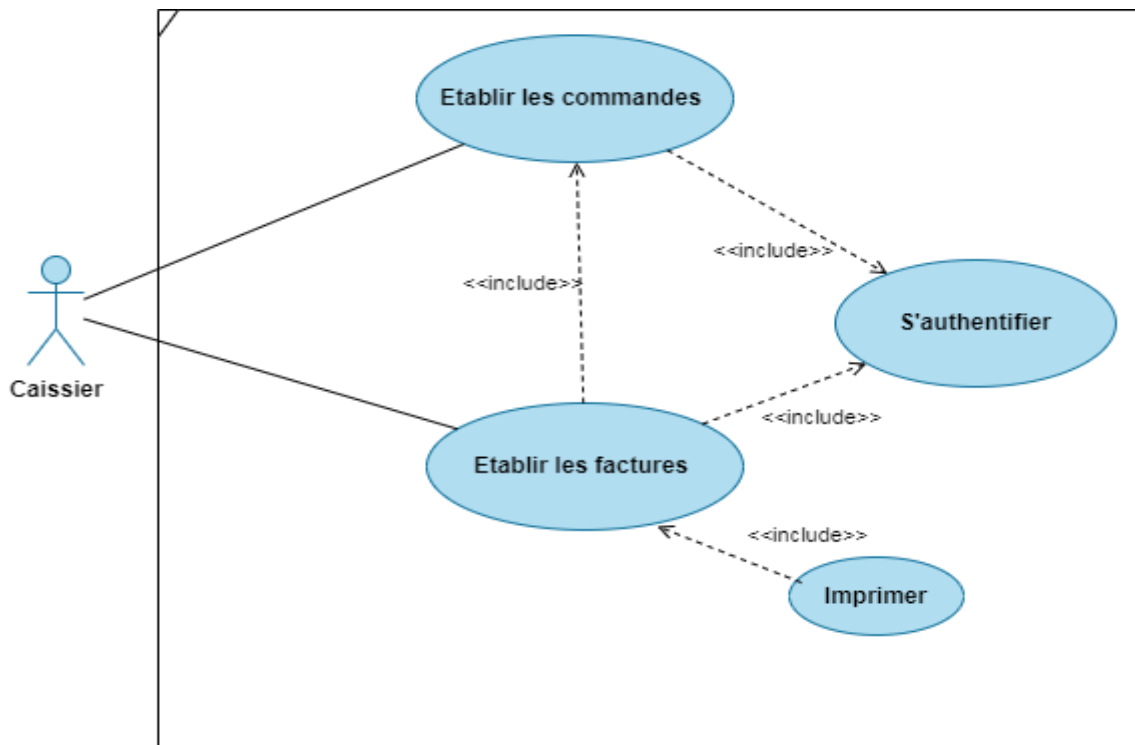


Figure 10: Diagramme de cas d'utilisation associé au Caissier

II.4.1.3. Diagramme de cas d'utilisation associé au « Directeur du restaurant »



Figure 11: Diagramme de cas d'utilisation associé au Directeur

II.4.1.4. Diagramme de cas d'utilisation global

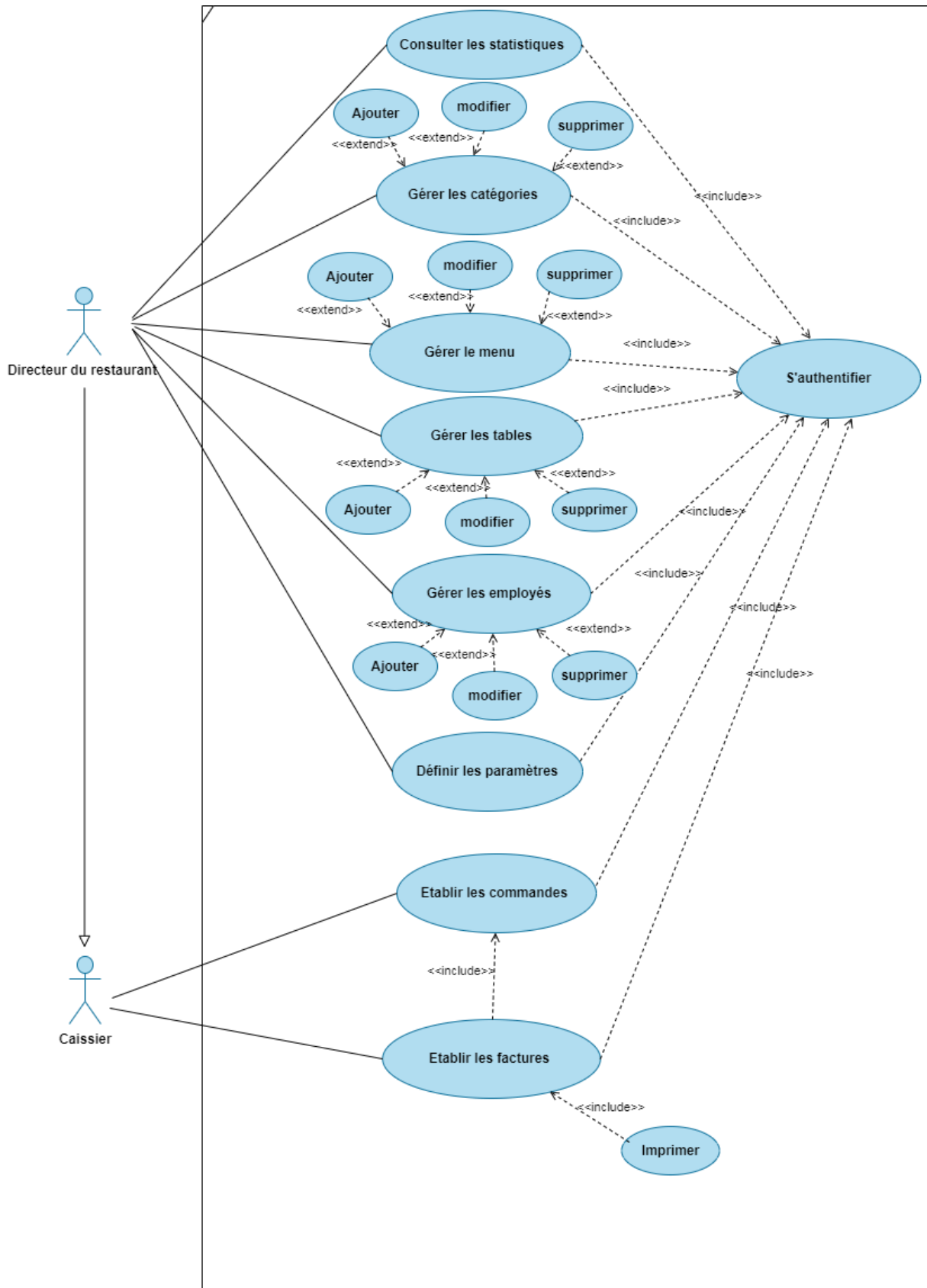


Figure 12: Diagramme de cas d'utilisation global

II.4.2. Description textuelle des cas d'utilisation

La description textuelle des cas d'utilisation est une méthode de modélisation qui consiste à décrire les différentes interactions entre les utilisateurs et un système logiciel. Elle comprend des éléments tels que le titre du cas d'utilisation, les acteurs impliqués, les conditions préalables, le scénario principal, les post-conditions, les extensions, les variantes et les exceptions.

Dans ce qui suit, nous allons décrire d'une manière détaillée les cas d'utilisation «Authentification», « Consulter les statistiques »,« Gérer les catégories», « Gérer le Menu », « Définir les paramètres» ainsi que les cas « Etablir les commandes »et « Etablir les factures».

Les descriptions des cas d'utilisation « Gérer les employés» et « Gérer les tables» sont semblables à celle des cas d'utilisation « Gérer les catégories »et « Gérer le Menu ».

Description textuelle de cas d'utilisation « S'authentifier »

Cas d'utilisation	S'authentifier.
Objectif	Permettre à chaque utilisateur d'accéder aux fonctionnalités qui lui sont spécifiques en vérifiant son identité (login et mot de passe).
Acteur	Utilisateur (Directeur du restaurant, Caissier)
Pré-condition	L'utilisateur doit lancer l'application.
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur accède à l'application. 2. Le système affiche l'interface d'authentification. 3. L'utilisateur saisit son nom d'utilisateur et son mot de passe. 4. Le système vérifie la validité des données saisies.[E1] 5. Le système donne l'accès à l'interface correspondante.
Enchaînement Alternatif	E1 : Erreur de nom d'utilisateur et/ou de mot de passe. <ol style="list-style-type: none"> 1. Le système affiche un message d'erreur. 2. Le système reprend de 3.

Tableau 22: Description du cas d'utilisation s'authentifier

Description textuelle de cas d'utilisation « Consulter les statistiques »

Cas d'utilisation	Consulter les statistiques.
Objectif	Permettre au directeur du restaurant d'accéder aux informations et aux données statistiques.
Acteur	Directeur du restaurant.
Pré-condition	Le directeur du restaurant doit s'authentifier après avoir lancé l'application.
Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche l'interface des statistiques. 2. Le directeur consulte le graphe des commandes ainsi que celui des revenus générés par le système. 3. Le directeur peut aussi consulter les statistiques susmentionnées dans un tableau en introduisant une date: <ol style="list-style-type: none"> 3.1. Le directeur demande l'interface. 3.2. Le système affiche l'interface. 3.3. Le directeur introduit l'intervalle des dates puis valide. 3.4. Le système génère les statistiques associées à l'intervalle introduit dans un tableau.
Enchaînement Alternatif	Aucun.

Tableau 23: Description du cas d'utilisation Consulter les statistiques

Description du cas d'utilisation « Gérer les catégories »

Cas d'utilisation	Gérer les catégories.
Objectif	Permettre au directeur de gérer les catégories de consommation en ajoutant, modifiant ou supprimant des catégories, afin d'organiser efficacement les différents types de consommation dans le système.
Acteur	Directeur du restaurant.
Pré-condition	Le directeur du restaurant doit s'authentifier après avoir lancé l'application.
Scénario nominal	<ol style="list-style-type: none"> 1. Le directeur accède à l'application. 2. Le directeur demande l'interface de gestion. 3. Le système affiche l'interface de gestion. 4. Le directeur sélectionne l'option Catégories. 5. Le système affiche l'interface de gestion des catégories. 6. Le directeur choisit l'action à effectuer : <ol style="list-style-type: none"> 6.1.Ajout <ol style="list-style-type: none"> 6.1.1. Le directeur remplit les champs. 6.1.2. Le directeur valide en cliquant sur le bouton « Ajouter». 6.1.3. Le système ajoute la catégorie à la liste. 6.2.Modification <ol style="list-style-type: none"> 6.2.1. Le directeur sélectionne la catégorie à modifier. 6.2.2. Le système remplit les champs avec les données relatives à la catégorie sélectionnée. 6.2.3. Le directeur effectue les modifications souhaitées puis valide. 6.2.4. Le système modifie la catégorie et affiche les modifications.

	<p>6.3. Suppression</p> <p>6.3.1. Le directeur sélectionne la catégorie à supprimer de la liste des catégories.</p> <p>6.3.2. Le directeur clique sur «Supprimer».</p> <p>6.3.3. Le système supprime la catégorie.</p>
Alternative	Aucune.

Tableau 24: Description du cas d'utilisation Gérer les catégories

Description du cas d'utilisation « Gérer le Meu »

Cas d'utilisation	Gérer le menu.
Objectif	Permettre à l'administrateur de gérer le menu du restaurant en ajoutant, modifiant ou supprimant des plats et des boissons.
Acteur	Directeur du restaurant.
Pré-condition	Le directeur du restaurant doit s'authentifier après avoir lancé l'application.
Scénario nominal	<ol style="list-style-type: none"> 1. Le directeur demande l'interface de gestion. 2. Le système affiche l'interface de gestion. 3. Le directeur sélectionne l'option Menu. 4. Le système affiche l'interface de gestion de Menu. 5. Le directeur choisi l'action à effectuer : <ol style="list-style-type: none"> 5.1.Ajout <ol style="list-style-type: none"> 5.1.1. Le directeur remplit les champs requis et importe l'image du nouvel élément. 5.1.2. Le directeur valide en cliquant sur le bouton « Ajouter». 5.1.3. Le système ajoute l'élément au menu. 5.2.Modification <ol style="list-style-type: none"> 5.2.1. Le directeur sélectionne l'élément à modifier. 5.2.2. Le système remplit les champs du formulaire avec les détails de l'élément sélectionné. 5.2.3. Le directeur effectue les modifications nécessaires puis valide. 5.2.4. Le système enregistre les nouvelles informations. 5.3.Suppression <ol style="list-style-type: none"> 5.3.1. Le directeur sélectionne l'élément à supprimer. 5.3.2. Le directeur clique sur le bouton «Supprimer». 5.3.3. Le système supprime l'élément.
Enchaînement Alternatif	Aucun.

Tableau 25: Description du cas d'utilisation Gérer le Menu

Description du cas d'utilisation « Définir les paramètres »

Cas d'utilisation	Définir les paramètres.
Objectif	Permettre au directeur de définir les paramètres spécifiques du restaurant dans l'application.
Acteur	Directeur du restaurant.
Pré-condition	Le directeur du restaurant doit s'authentifier après avoir lancé l'application.
Scénario nominal	<ol style="list-style-type: none"> 1. Le directeur sélectionne l'option « Paramètres » dans le menu principal. 2. Le système affiche l'interface des paramètres. 3. Le directeur effectue l'action : <ol style="list-style-type: none"> 3.1. Si le directeur utilise pour la première fois l'application Alors il définit les paramètres relatifs au restaurant tel que le nom du restaurant, l'adresse, le numéro de téléphone, etc. <ol style="list-style-type: none"> 3.1.1. Le directeur remplit les champs requis avec les informations relatives au restaurant puis valide. 3.1.2. Le système enregistre les paramètres sur le ticket. 3.2. Si non le directeur peut modifier les paramètres existants. <ol style="list-style-type: none"> 3.2.1. Le système remplit les champs avec les détails du restaurant. 3.2.2. Le directeur effectue les modifications nécessaires puis valide. 3.2.3. Le système affiche les nouvelles informations sur le ticket du restaurant.
Enchaînement Alternatif	Aucun.

Tableau 26: Description du cas d'utilisation Définir les paramètres

Description des cas d'utilisation «Etablir les commandes » et « Etablir les factures »

Cas d'utilisation	Etablir les commandes.
Objectif	Permettre au caissier d'enregistrer les commandes des clients dans le système en sélectionnant les plats et les boissons commandés et précisant les quantités.
Acteur	Le caissier, Directeur.
Pré-condition	Le caissier doit s'authentifier après avoir lancé l'application.
Scénario nominal	<ol style="list-style-type: none"> 1. Le système affiche l'interface des Commandes. 2. Le caissier sélectionne l'option appropriée: <ol style="list-style-type: none"> 2.1.A table si le client souhaite prendre sa commande sur place. <ol style="list-style-type: none"> 2.1.1. Le caissier saisi le numéro de la table à laquelle appartient la commande. 2.2.A Emporter si le client souhaite emporter sa commande. 3. Le caissier saisi la commande de client en sélectionnant les plats et les boissons commandés, et en précisant la quantité de chaque élément. 4. Le système affiche la liste des éléments sélectionnés, y compris leur prix total en fonction de la quantité commandée, ainsi que le prix total de la commande. 5. Le caissier valide la commande. 6. Le système génère le ticket contenant les détails de la commande ainsi le prix total. 7. Le caissier imprime le ticket.
Enchaînement Alternatif	Aucune.

Tableau 27: Description des cas d'utilisation Etablir les commandes et les factures

II.5. Présentation du diagramme de séquence

Le diagramme de séquence permet de représenter les échanges entre les acteurs externes et le système lui-même. Il met en évidence la séquence chronologique de ces échanges, en montrant les messages échangés et les actions effectuées. Cette représentation peut se réaliser par cas d'utilisation en considérant les différents scénarios associés (5).

II.5.1. Eléments de base d'un diagramme de séquence

a. Formalisme d'un diagramme de séquence

Le diagramme de séquence est généralement représenté par un grand rectangle qui englobe l'ensemble du diagramme, avec le nom du diagramme indiqué en haut à gauche (5).

b. Objet

Un objet est une instance d'une classe et il est représenté par un rectangle dans lequel le nom de la classe est écrit et souligné (7).

c. Ligne de vie

Une ligne de vie représente l'existence temporelle d'un objet ou d'une entité. Elle se présente sous la forme d'une ligne verticale, généralement accompagnée du nom de l'objet ou de l'entité à côté. Elle indique le laps de temps pendant lequel l'objet ou l'entité est actif et participe aux interactions du système (5).

d. Message

Un message est un moyen de communication unidirectionnel entre objets, déclenchant une action dans l'objet destinataire. La flèche pointillée représente un retour ou une réponse directe au message précédent (4).

Dans un diagramme de séquence, deux types de message peuvent être distingués (5) :

- **Message asynchrone** : Il ne nécessite pas de réponse immédiate de la part du destinataire. L'émetteur peut continuer son exécution sans attendre de retour. Il se représente par une flèche dont l'extrémité n'est pas pleine.

- **Message synchrone** : Dans ce cas, l'émetteur se bloque jusqu'à ce que le destinataire traite le message et renvoie éventuellement une réponse. Ce type de message est représenté par une flèche avec une extrémité pleine.

e. Fragment d'interaction

Un fragment d'interaction combiné est un regroupement d'interactions auquel un opérateur est appliqué. Il est représenté globalement comme un diagramme de séquence, avec le nom de l'opérateur indiqué dans le coin supérieur gauche.

Dans le langage de modélisation UML, treize opérateurs ont été définis, tels que alt, opt, loop, par, strict, break, ignore, critical, negative, assertion et ref (5).

Dans ce qui suit nous décrivons les opérateurs que nous utilisons dans les diagrammes de séquence de notre projet : alt, opt, loop et ref (5).

- Opérateur « **alt** » : Est un opérateur conditionnel utilisé pour représenter une exécution à choix multiples basée sur des conditions de garde. Seul le sous-fragment dont la condition est vraie sera exécuté.
- Opérateur « **opt** » : Est utilisé pour indiquer des interactions optionnelles, où une séquence d'interactions peut être exécutée ou ignorée en fonction d'une condition donnée.
- Opérateur « **loop** » : Est utilisé pour modéliser des boucles ou des répétitions, où une séquence d'interactions est répétée tant qu'une condition est satisfaite.
- Opérateur « **ref** » : Permet d'appeler une séquence d'interactions déjà définie ailleurs, agissant ainsi comme une référence à un sous-diagramme de séquence.

Dans ce qui suit, nous allons présenter les diagrammes de séquence pour les cas d'utilisation «Authentification», « Consulter les statistiques », « Gérer les catégories», « Gérer le menu », « Définir les paramètres » ainsi que les cas « Etablir les commandes » et « Etablir les factures »:

Les diagrammes de séquence pour les cas d'utilisation « Gérer les employés » et « Gérer les tables » sont semblables à celle du cas d'utilisation « Gérer les catégories » et « Gérer le menu ».

II.5.1.1. Diagramme de séquence du cas d'utilisation « S'authentifier »

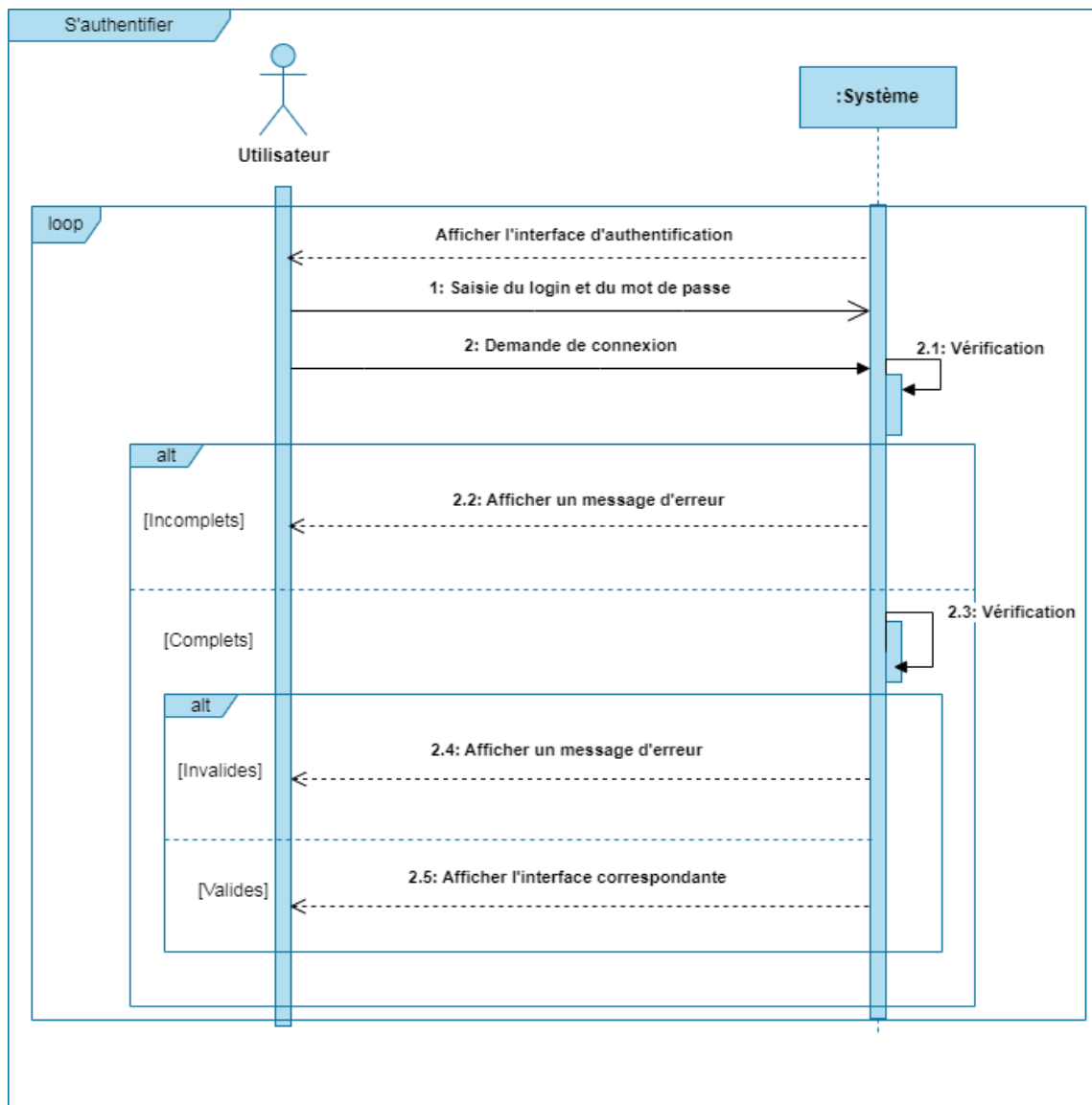


Figure 13: Diagramme de séquence du cas d'utilisation S'authentifier

II.5.1.2. Diagramme de séquence du cas d'utilisation «Consulter les statistiques»

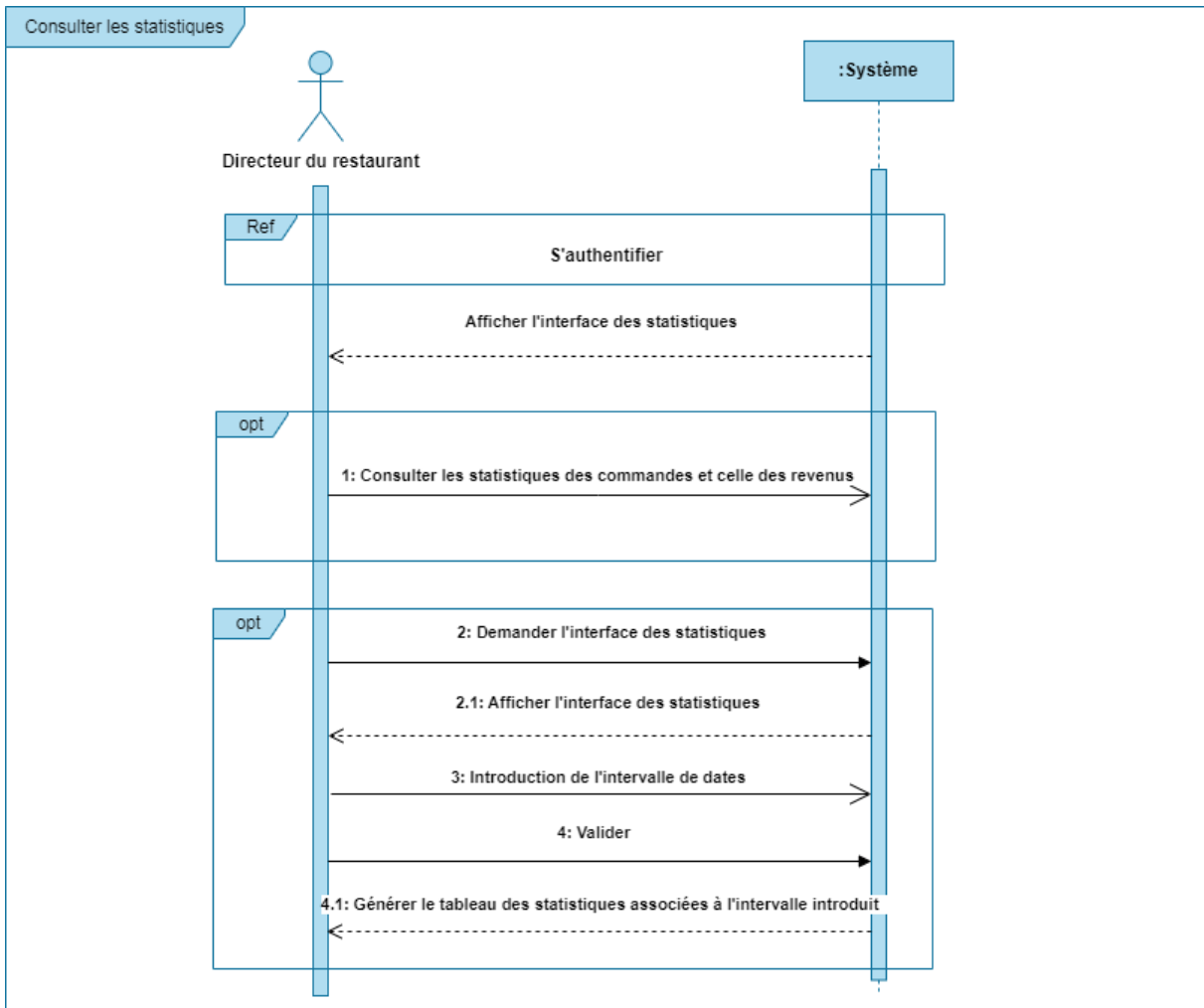


Figure 14: Diagramme de séquence du cas d'utilisation Consulter les statistiques

II.5.1.3. Diagramme de séquence du cas d'utilisation « Gérer les catégories»

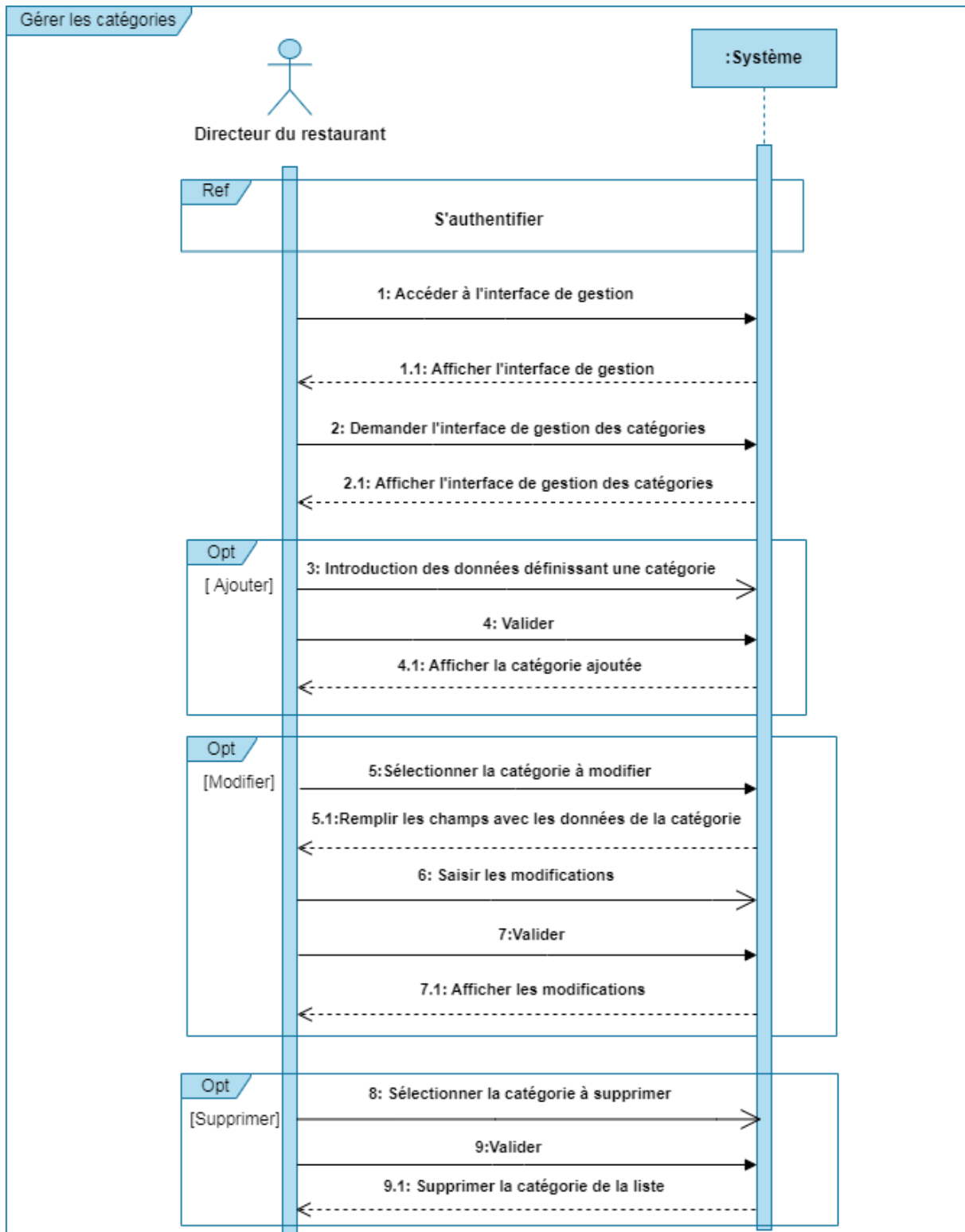


Figure 15: Diagramme de séquence du cas d'utilisation Gérer les catégories

II.5.1.4. Diagramme de séquence du cas d'utilisation « Gérer le Menu »

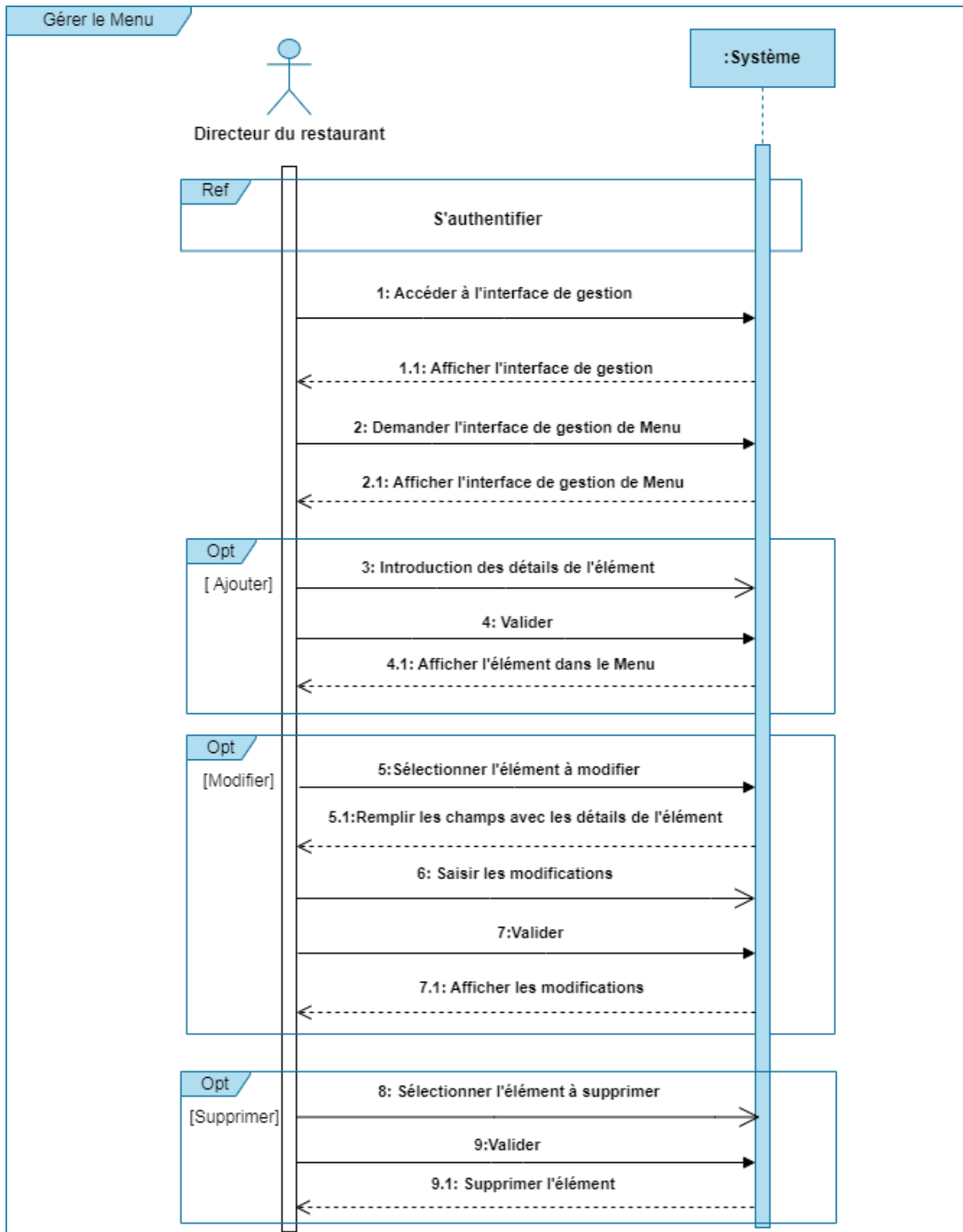


Figure 16: Diagramme de séquence du cas d'utilisation Gérer le menu

II.5.1.5. Diagramme de séquence du cas d'utilisation «Définir les paramètres»

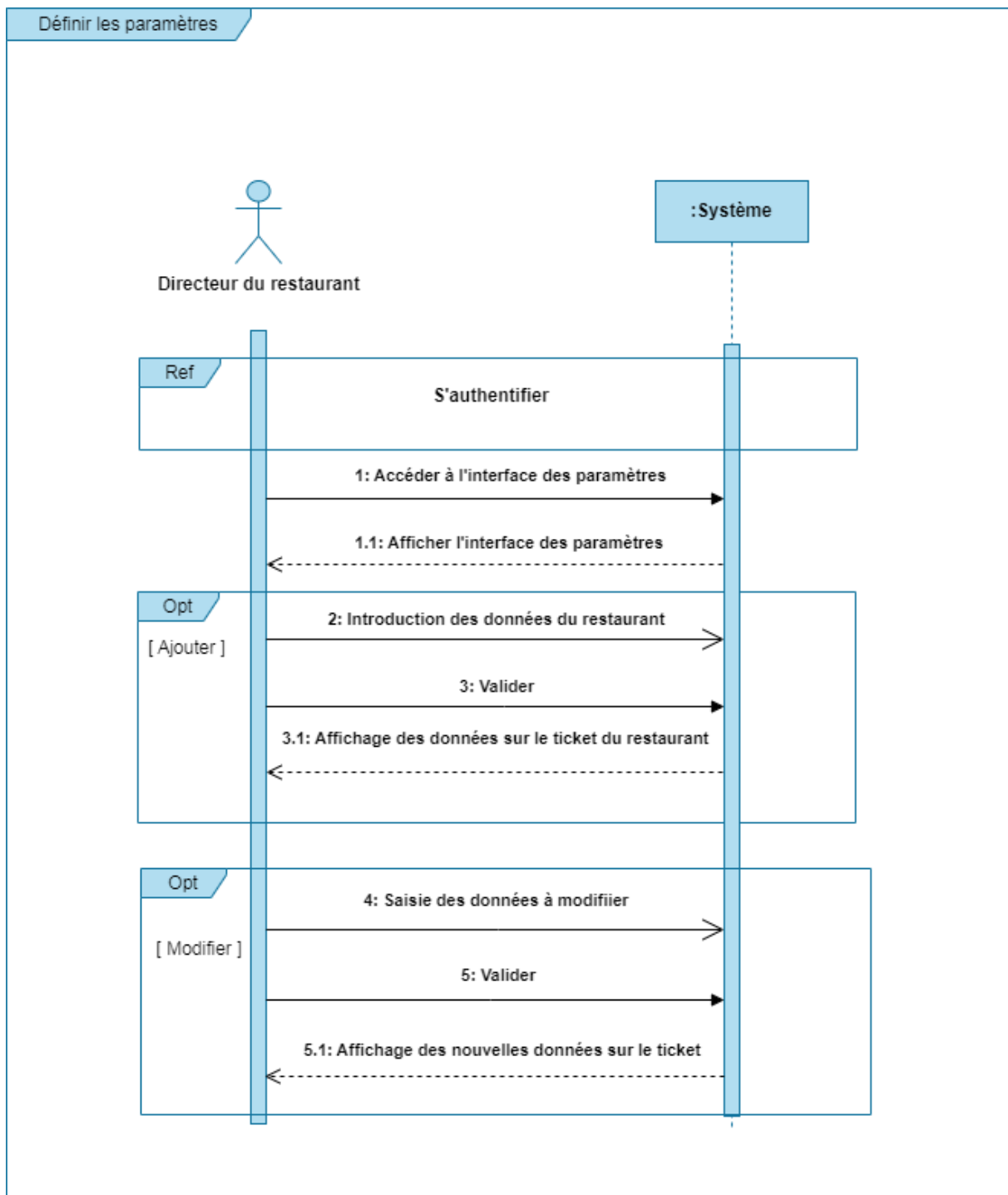


Figure 17: Diagramme de séquence du cas d'utilisation Définir les paramètres

II.5.1.6. Diagramme de séquence des cas d'utilisation «Etablir les commandes et les facture »

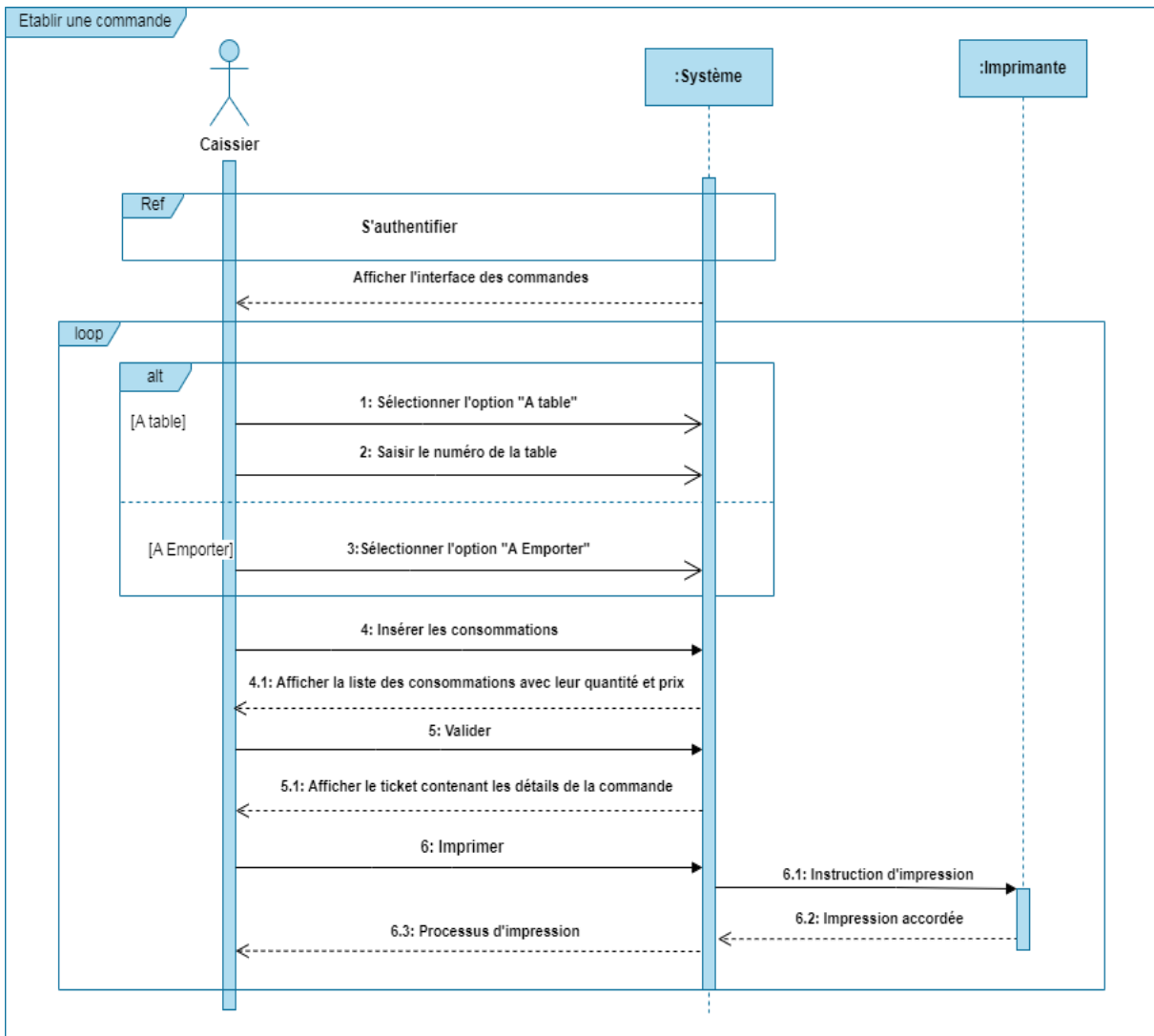


Figure 18: Diagramme de séquence des cas d'utilisation Etablir les commandes et les factures

II.6. Présentation du diagramme de classes

Le diagramme de classes occupe une place centrale dans le développement orienté objet en fournissant une représentation statique du système à développer. Il met en évidence les classes, les attributs, les opérations et les relations entre elles, permettant ainsi de visualiser la structure et les interactions du système de manière claire. En se concentrant sur les concepts de classe et d'association, ce diagramme facilite la compréhension et la conception du système logiciel (8).

II.6.1. Éléments de base d'un diagramme de classes

Les éléments de base d'un diagramme de classes sont les suivants :

- a. **Classe** : Une classe est la représentation d'une entité ou d'un groupe d'objets partageant des attributs (propriétés) et des opérations (comportement) communs. L'instanciation d'une classe permet ainsi de créer un objet spécifique basé sur ce modèle commun.

Une classe est représentée par un rectangle séparé en trois parties qui sont :

- **Nom d'une classe** : Doit refléter le concept décrit par la classe.
 - **Attributs** : Sont des données déclarées au niveau d'une classe et chaque objet de cette classe leur donne une valeur spécifique (4).
 - **Opération** : est une fonction qui peut être appliquée aux objets d'une classe et qui décrit leur comportement (5).
- b. **Relations entre les classes** : En UML, il existe plusieurs types de relations entre les classes (5) :
 - **Association** : Est une relation durable et sémantique entre deux classes qui décrit un ensemble de liens entre les instances de ces classes.
 - **Classe-association** : Est une association qui est élevée au statut de classe. Elle combine à la fois les caractéristiques d'une association et d'une classe, ce qui lui permet de posséder des attributs qui prennent des valeurs pour chaque lien spécifique.

- **Agrégation** : Est une forme spécifique d'association asymétrique qui exprime une relation de contenance. Contrairement aux autres types d'associations, les agrégations n'ont pas besoin d'être nommées, car implicitement elles représentent le sens de « contient » ou « est composé de ».
- **Composition** : La composition est une forme d'agrégation où il existe une contrainte de durée de vie entre le composant et la ou les classes composées, ce qui signifie que la suppression de la classe « composé » entraîne également la suppression de la ou des classes « composant ».
- **Héritage** : L'héritage entre classes en UML peut être considéré comme une relation d'inclusion, où les objets instances des sous-classes sont également des objets instances des superclasses.

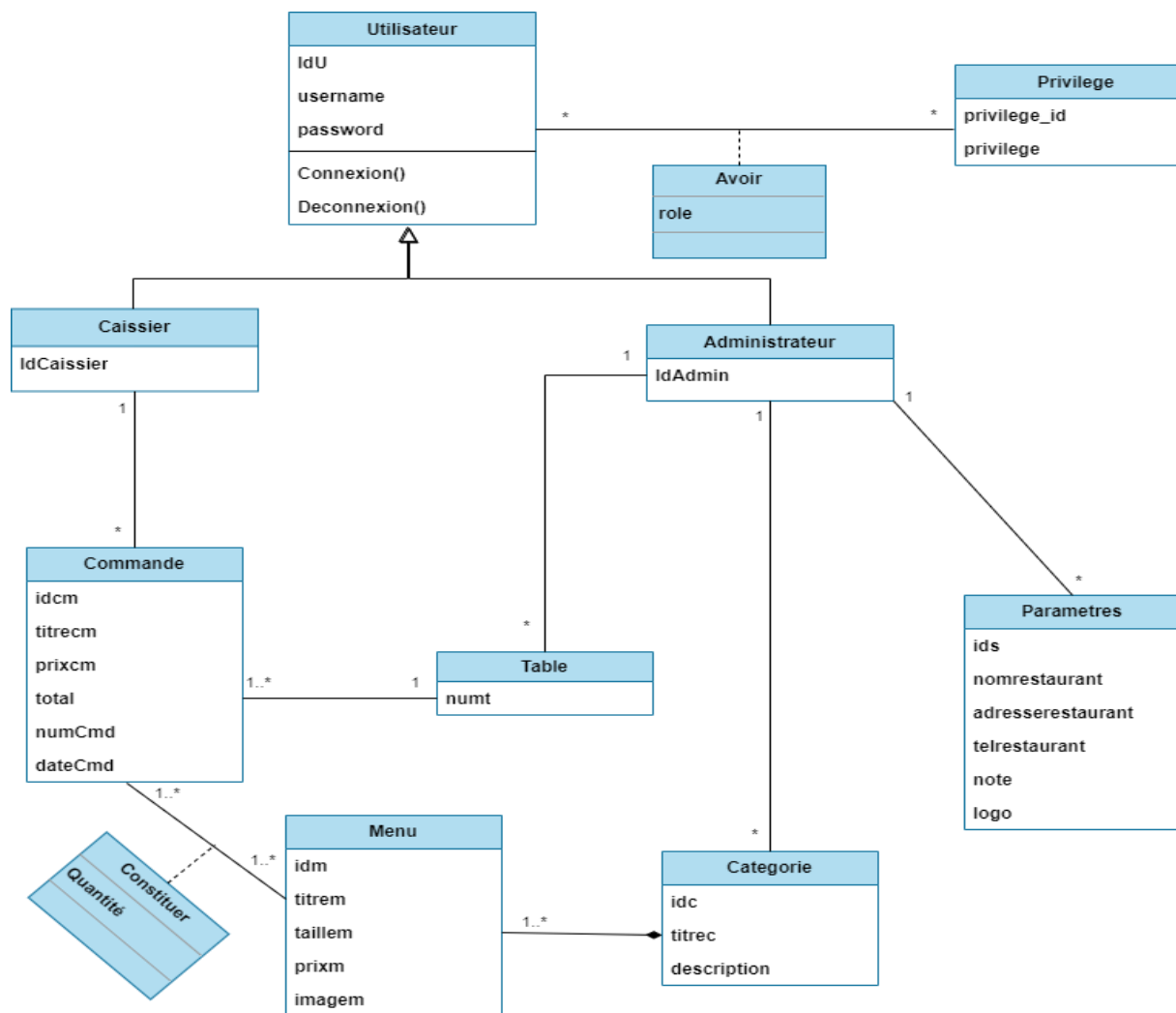


Figure 19:Diagramme de classes

II.7. Le modèle relationnel

Le modèle relationnel est une représentation structurée de l'univers des données défini dans le modèle conceptuel (diagramme de classe). Il traduit les concepts et les relations du modèle conceptuel en une forme compréhensible par une machine, en tenant compte du type de base de données choisi.

II.7.1. Les règles de passage au modèle relationnel

Les règles utilisées pour le passage du diagramme de classes de notre application au modèle Relationnel sont les suivantes (9):

- **Règle 1 : Transformation des classes**

Chaque classe du diagramme devient une relation dans le modèle relationnel. Il est nécessaire de choisir un attribut dans chaque classe qui pourra servir d'identifiant pour la relation. Si aucun attribut ne convient en tant qu'identifiant, il est possible d'ajouter un attribut supplémentaire de manière à ce que la relation dispose d'une clé primaire.

- **Règle 2 : Association un-à-plusieurs**

Il est nécessaire d'ajouter un attribut de type clé étrangère dans la relation "fils" de l'association. Cet attribut sera nommé d'après la clé primaire de la relation "père" de l'association.

- **Règle 3 : Association plusieurs-à-plusieurs**

L'association se transforme en une relation dont la clé primaire est formée en concaténant les identifiants des classes connectées à l'association. Les attributs de l'association sont inclus dans cette nouvelle relation.

- **Règle 4 : Association un-à-un**

Pour la classe ayant une multiplicité minimale égale à un, il est nécessaire d'ajouter un attribut clé étrangère dans la relation dérivée correspondante. Cet attribut doit être nommé d'après la clé primaire de la relation dérivée de la classe connectée à l'association.

Dans le cas où les deux cardinalités minimales sont à zéro, il est possible de choisir entre les deux relations dérivées conformément à la règle R1. Si les deux cardinalités minimales sont à un, il est généralement recommandé de fusionner les deux entités (classes) en une seule.

- **Règle 5 : Transformation de l'héritage**

Il existe trois options pour traduire une association d'héritage en fonction des contraintes existantes :

1. **Décomposition par distinction** : Chaque sous-classe est transformée en une relation distincte. La clé primaire de la surclasse est migrée vers la (les) relation(s) correspondante(s) des sous-classes, où elle devient à la fois clé primaire et clé étrangère.
2. **Décomposition descendante (push-down)** : Si une contrainte de totalité ou de partition est présente sur l'association d'héritage, la relation issue de la surclasse peut ne pas être traduite. Les attributs de la surclasse sont alors migrés vers la (les) relation(s) correspondante(s) des sous-classes.
3. **Décomposition ascendante (push-up)**: Les relations correspondantes des sous-classes sont supprimées et les attributs sont migrés vers la relation de la surclasse. Cette approche permet de regrouper les attributs au niveau de la surclasse.

- **Règle 6 : Composition**

Les relations issues des classes composantes doivent avoir une clé primaire qui inclut l'identifiant de la classe composite, indépendamment des multiplicités.

En appliquant les règles de passage précédemment énumérées, nous obtenons le modèle relationnel suivant pour la base de données de l'application à mettre en œuvre :

Utilisateur (IdU, username ,password);

Caissier (IdCaissier,#IdU);

Administrateur (IdAdmin,#IdU);

Commande (idcm, titrecm,prixcm,total,numCmd,dateCmd,#IdCaissier,#idt);

Menu (idm,#IdC,titre,m,taille,m,prix,m,imagem);

Categorie (idc,titre,c,description,#IdAdmin);

Table (numt,#IdAdmin);

Parametres (Ids, nomrestaurant, adresserestaurant,telrestaurant, logo,note,#IdAdmin);

Privilege (privilege_id , privilege);

Avoir (#privilege_id,#IdU,role);

Constituer (#IdCmd,#IdM, Quantité);

II.8. Conclusion

Ce chapitre a été consacré à la conception de notre application en utilisant une approche orientée objet. Nous avons commencé par présenter le langage de modélisation UML et le processus de développement UP, qui ont été les piliers de notre démarche. Ensuite, nous avons identifié les cas d'utilisation associés aux différents acteurs et représenté leurs interactions à l'aide des diagrammes de cas d'utilisation et de séquence. Enfin, nous avons élaboré le diagramme de class et abordé les règles de passage vers le modèle relationnel, ce qui nous a permis de définir le schéma de la base de données de notre application.

CHAPITRE III

REALISATION

III.1. Introduction

Précédemment, une étude conceptuelle a été établie dans le but de préparer la réalisation de notre application qui fait l'objet de ce chapitre, pour se faire, l'aspect visuel de notre plateforme est pris en considération, ce qui facilite son utilisation grâce à des interfaces intuitives. Les moyens technologiques utilisés pour son développement sont également décrits. Notre objectif principal était de concevoir et réaliser une plateforme pratique conviviale, ergonomique et personnalisable.

III.2. Les outils de développement

Pour le développement de notre application, nous avons utilisé les outils définis ci-dessous :

III.2.1. Présentation du WampServer

WampServer, anciennement connu sous le nom de WAMP5, est bien plus qu'une simple plateforme de développement web. Il offre un environnement complet pour exécuter des scripts PHP localement, sans avoir besoin d'une connexion à un serveur externe. Composé de serveurs Apache, MySQL et MariaDB, ainsi qu'un interpréteur de scripts PHP et de phpMyAdmin pour l'administration des bases de données, WampServer facilite grandement le processus de développement web. Avec son interface d'administration conviviale accessible via une icône dans la barre des tâches de Windows, il permet une gestion et une administration simples des serveurs.

La grande nouveauté de WampServer 3 réside dans sa flexibilité, offrant la possibilité d'installer et d'utiliser différentes versions de PHP, Apache, MySQL ou MariaDB en un seul clic. Cela permet aux développeurs de recréer fidèlement leur environnement de production sur leur machine locale, garantissant une cohérence et une fiabilité optimales (10).

III.2.2. Les outils principaux du WampServer

WAMP contient des outils tels que MySQL, PhpMyAdmin. Nous verrons à propos de ces outils :

III.2.2.1. Présentation du MySQL-Server

MySQL (MyStructuredQueryLanguage), est un gestionnaire de base de données. Cet outil puissant permet de créer et de gérer des bases de données en utilisant des fonctions et des instructions SQL. Les instructions SQL sont écrites dans un langage de script appelé PHP, qui est intégré dans les pages HTML pour effectuer des actions et des réponses vers une base de données. Grâce à cette combinaison harmonieuse, PHP et MySQL offrent une synergie parfaite pour une manipulation efficace des données dans le cadre de notre application (11).

III.2.2.2. Présentation du phpMyAdmin

PhpMyAdmin est une interface conviviale qui permet de gérer très facilement une base de données, sans nécessiter une connaissance avancée des requêtes SQL. Le fait que l'interface soit développée en PHP la rend parfaitement adaptée à l'utilisation conjointe d'une base MySQL et d'un moteur de scripts PHP (12).

Il permet d'effectuer différentes tâches telles que la création, la modification ou la suppression de bases de données, des tables, des champs ou des lignes, exécution des requêtes SQL, ou la gestion des utilisateurs et des permissions.

III.2.3. Présentation du JDBC

JDBC (Java DataBaseConnectivity) créé par Sun Microsystems et actuellement détenu par Oracle Corporation, est une interface de programmation conçue pour les applications Java. Elle offre aux applications Java la capacité d'accéder à diverses sources de données en utilisant une interface standardisée, en s'appuyant sur les pilotes JDBC correspondants (13).

III.2.4. Java Développement Kit (JDK)

Le JDK (Java Development Kit) est un ensemble d'outils logiciels essentiels utilisés pour le développement d'applications et d'applets Java. Il comprend l'environnement d'exécution Java (JRE) qui permet d'exécuter les programmes Java, ainsi que des composants tels qu'un interpréteur/chargeur, un compilateur, un archiveur, un générateur de

documentation et d'autres outils nécessaires au développement Java. Le JDK fournit aux développeurs un environnement complet et puissant pour créer, tester et déployer des applications Java de manière efficace (14).

III.2.5. Présentation de Scenebuilder

SceneBuilder est un outil de conception d'interfaces graphiques convivial et intuitif pour les applications JavaFX. Il permet aux développeurs de créer facilement des interfaces utilisateur en utilisant une approche de glisser-déposer.

SceneBuilder offre une vue en temps réel de l'interface en cours de création, ce qui facilite la personnalisation visuelle et l'organisation des éléments. Il permet également de gérer les propriétés, les événements et les styles des composants graphiques de manière visuelle, sans nécessiter de programmation manuelle (15).

III.2.6. Présentation de NetBeans

NetBeans est un environnement de développement intégré (EDI) polyvalent et puissant, largement utilisé pour la programmation Java. Il offre une gamme complète d'outils et de fonctionnalités pour faciliter le développement d'applications.

NetBeans comprend un éditeur de code avancé avec des fonctionnalités de complétion automatique et de débogage intégrées, ainsi qu'une interface conviviale pour la création d'interfaces graphiques. Il prend en charge la gestion de projets, la compilation, le déploiement et offre une intégration fluide avec d'autres technologies telles que XML et JavaScript. NetBeans est une solution polyvalente pour les développeurs, offrant une productivité accrue et une facilité de développement dans un environnement intuitif (16).

III.3. Les langages de programmation

III.3.1. Présentation du langage JAVA

Java est un langage orienté objet, ce qui signifie qu'il est conçu pour modéliser des concepts du monde réel à l'aide de classes et d'objets. Il offre une syntaxe claire et concise. Une des principales caractéristiques de Java est sa plateforme indépendante. Les programmes Java sont compilés en byte code, un langage intermédiaire compris par la machine virtuelle Java (JVM). Cela permet à un programme Java de s'exécuter sur différents systèmes

d'exploitation, tels que Windows, macOS et Linux, sans nécessiter de modifications supplémentaires.

Java est utilisé dans une grande variété de domaines, allant du développement d'applications desktop et de sites web à la création de logiciels d'entreprise et d'applications mobiles. Il dispose également d'une vaste bibliothèque standard (Java Standard Edition) offrant de nombreuses fonctionnalités prêtes à l'emploi (17).

III.3.1.1. Présentation de JavaFx

JavaFX est une technologie développée initialement par Sun Microsystems, qui a été acquise par Oracle en 2009. À partir de Java 8, JavaFX est devenue la bibliothèque officielle de création d'interfaces graphiques pour le langage Java, remplaçant progressivement Swing.

JavaFX offre des fonctionnalités diverses telles que le support des médias audio et vidéo, la création de graphiques en 2D et 3D, la programmation Web et la gestion des threads.

Cette technologie permet de développer des applications pour diverses plateformes, notamment les applications mobiles, les applications de bureau et les applications Web (18).

III.3.2. Présentation de FXML

FXML est un langage de balisage scriptable qui se base sur XML et permet de construire des graphes d'objets Java de manière simplifiée. Il offre une alternative pratique à la construction de ces graphes en utilisant du code procédural, ce qui le rend idéal pour définir l'interface utilisateur des applications JavaFX.

La structure hiérarchique d'un document XML utilisé en FXML correspond étroitement à la structure du graphe de scène de JavaFX, ce qui facilite la création et la manipulation des éléments de l'interface utilisateur de manière intuitive et claire.

III.3.3. Présentation du CSS

CSS (Cascading Style Sheets) est un langage de programmation utilisé pour styliser et mettre en forme les éléments d'une page web. Il complète le langage HTML en permettant la gestion de l'apparence visuelle des sites web. Avec CSS, on peut définir des styles tels que la couleur, la taille, la police, les marges et les espacements pour les différents éléments HTML.

Il offre une séparation claire entre la structure et la présentation des pages, permettant ainsi une maintenance et une mise à jour faciles.

CSS est compatible avec la plupart des navigateurs et offre une flexibilité et une cohérence dans la conception des sites web. Il est largement utilisé dans le développement web pour créer des interfaces attrayantes et réactives (19).

III.3.4. Présentation du SQL

SQL (StructuredQueryLanguage) est un langage permettant d'accéder et de manipuler des bases de données. Il est devenu une norme internationale reconnue par l'ANSI en 1986 et l'ISO en 1987. SQL offre une syntaxe standardisée et des commandes pour réaliser diverses opérations sur les bases de données, telles que la création, l'interrogation, la modification et la suppression de données.

Pour donner vie à notre projet, nous avons adopté le concept MVC (Modèle-Vue-Contrôleur) qui offre une structure claire et organisée pour la gestion des données, de l'interface utilisateur et de la logique de contrôle (20).

III.4. Présentation du concept MVC

Le MVC (Modèle-Vue-Contrôleur) est un modèle d'architecture qui vise à clairement séparer les différentes couches d'une application : la couche de présentation (UI : Interface Utilisateur), la couche métier (BLL : Business Logic Layer) et la couche d'accès aux données (DAL : Data Access Layer). L'objectif principal est de minimiser les dépendances entre ces couches, de sorte que les modifications apportées à l'une n'affectent pas les autres (21).

- **Le modèle :** représente les données et la logique métier de l'application. Il est responsable de l'accès aux données, de leur manipulation et de leur mise à jour.
- **La vue :** est responsable de l'interface utilisateur de l'application. Elle affiche les données au sein d'une interface graphique et permet aux utilisateurs d'interagir avec l'application.
- **Le contrôleur :** agit comme l'intermédiaire entre le modèle et la vue. Il gère les interactions utilisateur, traite les entrées, met à jour le modèle en conséquence et actualise la vue en fonction des changements.

L'architecture MVC favorise la séparation des préoccupations, ce qui facilite la maintenance, la réutilisation du code et la collaboration entre les membres de l'équipe de développement. Il offre également une meilleure extensibilité et évolutivité de l'application.

III.5. Présentation des interfaces de l'application

Dans ce qui suit, nous présentons les interfaces qui illustrent les fonctionnalités principales de notre application.

III.5.1. Interface d'Authentification

Pour accéder aux services de notre application, l'utilisateur doit s'authentifier en fournissant son nom d'utilisateur et son mot de passe, garantissant ainsi la sécurité et la confidentialité de son accès.

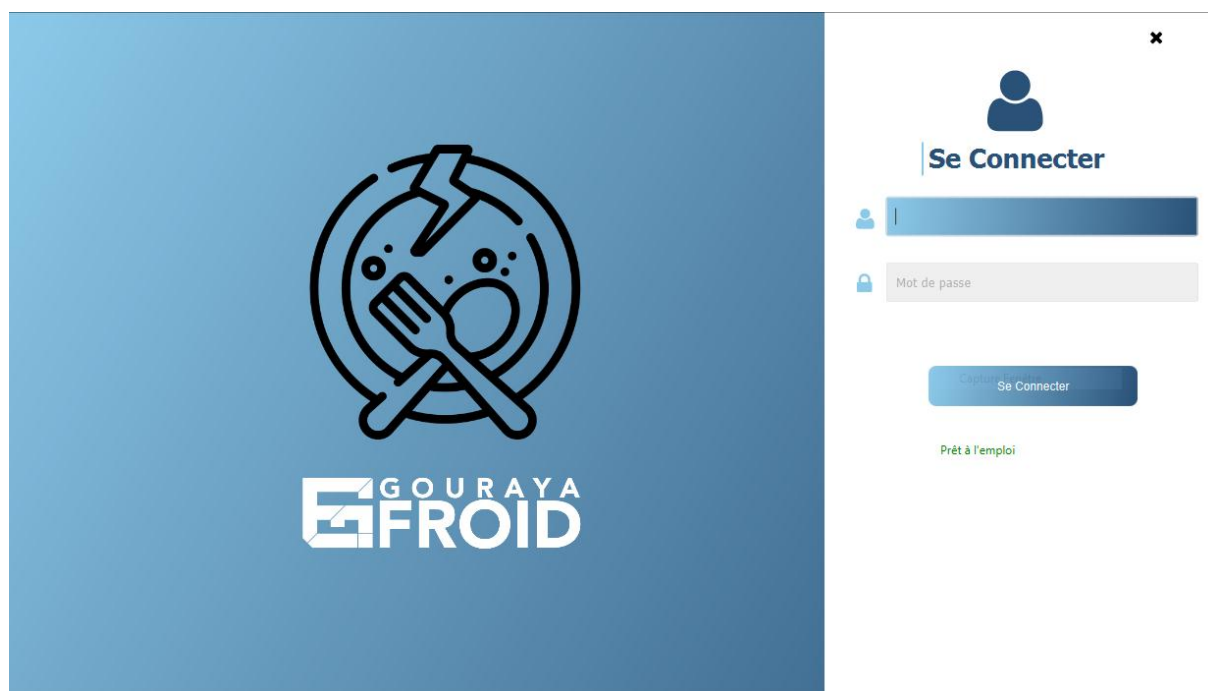


Figure 20: Interface d'authentification

III.5.2. Interface Accueil

L'interface d'accueil de notre application offre des rapports globaux et détaillés, présentés sous forme de graphiques, qui fournissent une vue complète des entrées, y compris le nombre total de commandes

Ces rapports peuvent être filtrés par caissier ou par administrateur, permettant ainsi une analyse approfondie selon les besoins spécifiques. Il est à noter que seule l'administration dispose du privilège d'accéder à ces informations, garantissant la confidentialité et la sécurité des données.



Figure 21: Interface d'accueil

III.5.3. L'interface de gestion

L'interface de gestion de notre application est le pilier central pour administrer les catégories, le menu, les tables et les employés. Elle offre des fonctionnalités complètes de gestion, incluant l'ajout, la modification et la suppression des éléments dans chaque section. Cette approche cohérente de gestion facilite la navigation et la compréhension de l'application.

En se concentrant sur l'interface du menu, elle permet une présentation intuitive et claire des plats, des prix et des options disponibles, offrant ainsi une expérience utilisateur optimale.

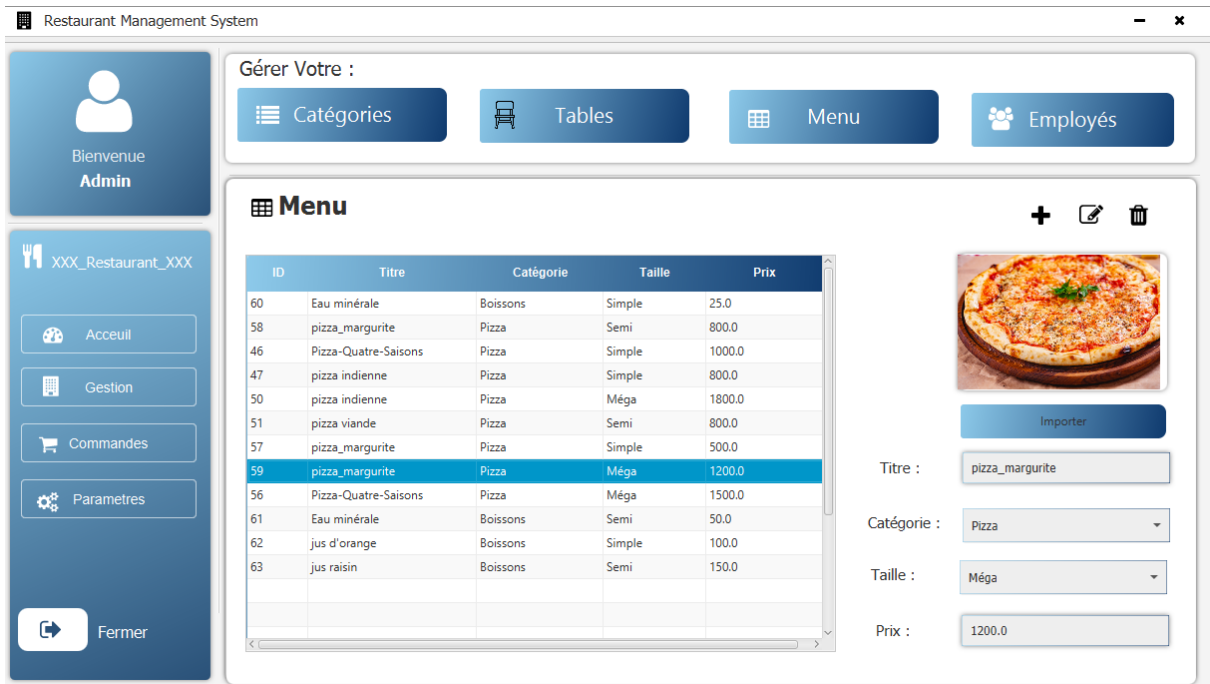


Figure 22: Interface de gestion de Menu

III.5.4. L'interface des Commandes

Cette interface permet à l'utilisateur (Caissier/Directeur) d'établir les commandes des clients.

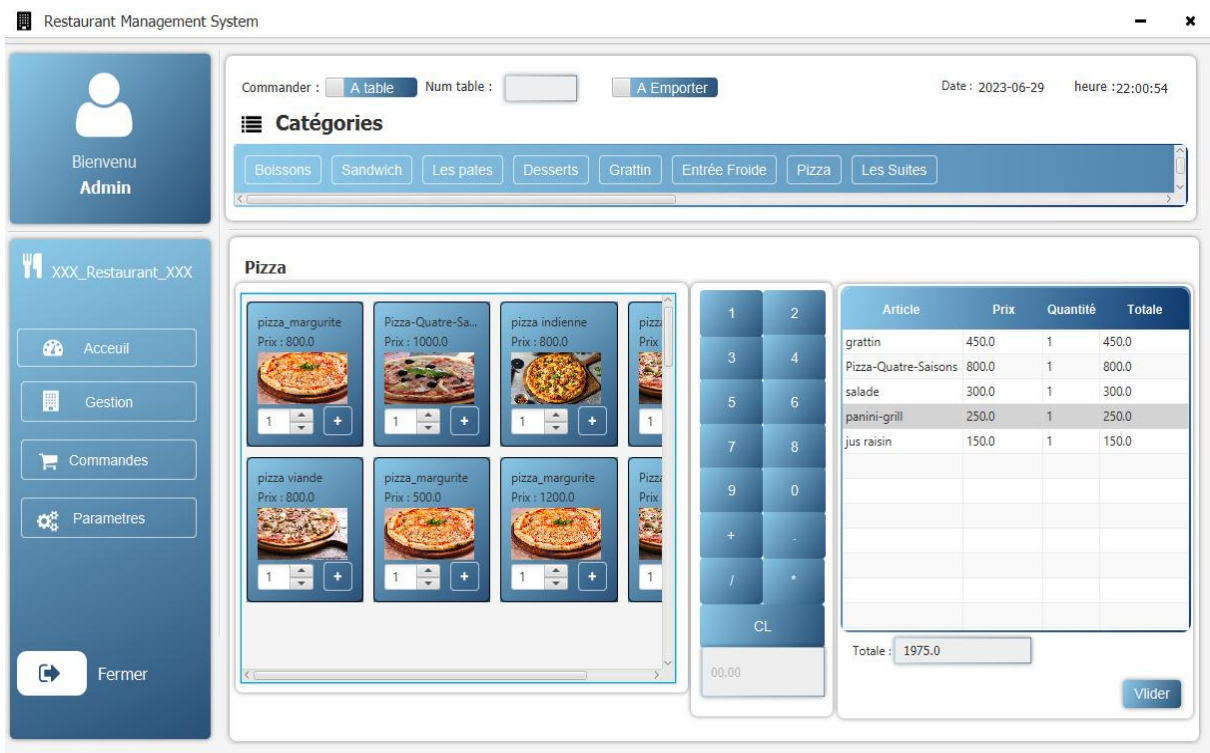


Figure 23: Interface des Commande [1]

Après avoir validé la commande un ticket contenant les détails de la commande sera généré par le système.

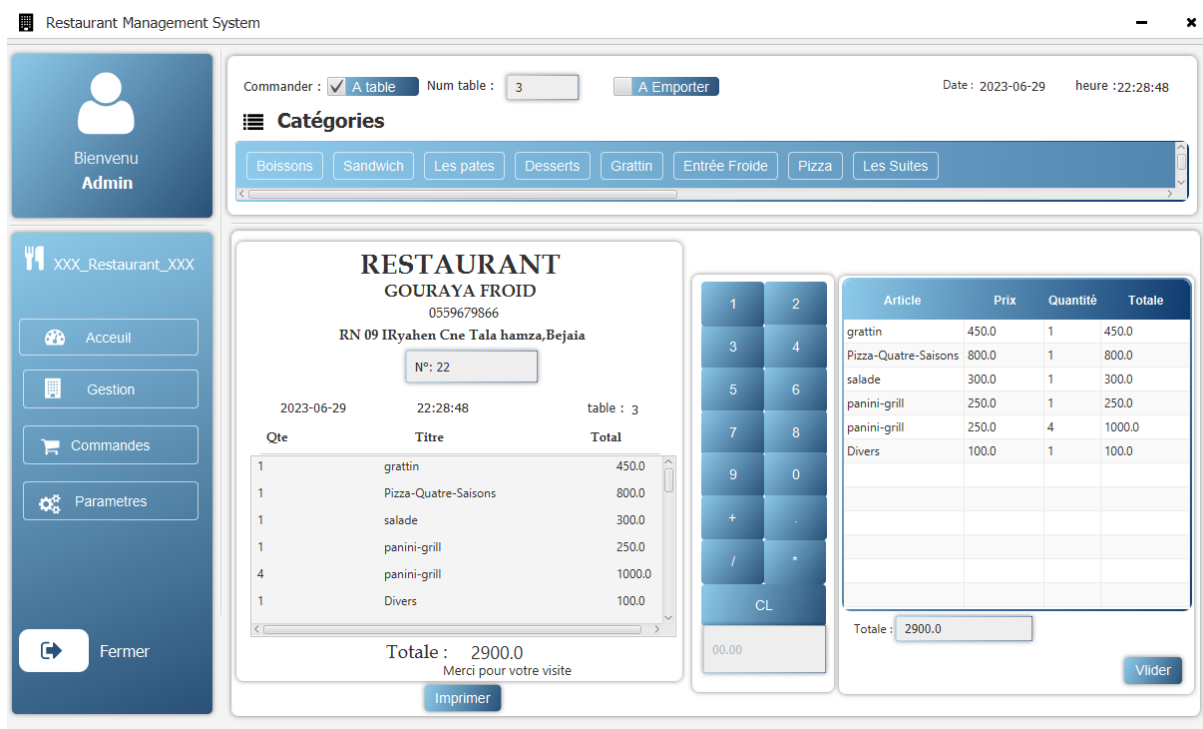


Figure 24: Interface des commandes (2)

III.5.5. L'interface des Paramètres

L'interface "Paramètres" regroupe les informations relatives au restaurant, telles que son nom, son numéro de téléphone, son adresse, son logo et une note attribuée. offrant ainsi une personnalisation et une identification claires de l'établissement.

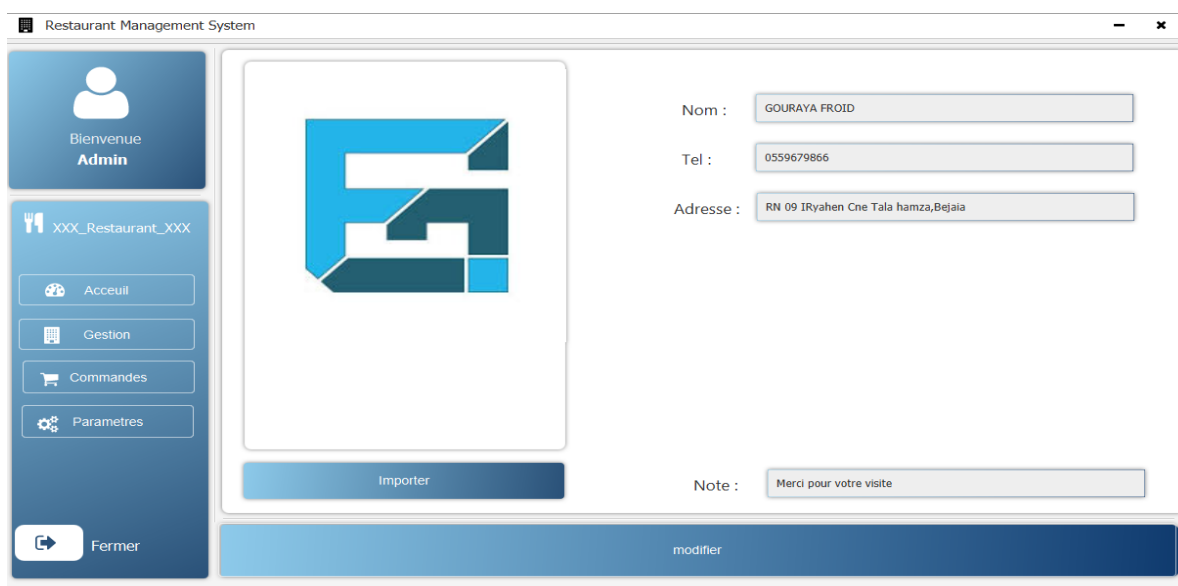


Figure 25: Interface des paramètres

III.6. Conclusion

La phase de réalisation constitue l'étape concrète dans le cycle de vie d'une application, où chaque détail compte. Dans ce chapitre, nous avons dévoilé les outils que nous avons utilisés pour concrétiser notre vision, ainsi que les différentes fonctionnalités que nous avons méticuleusement conçues sous forme de fenêtres intuitives. Chaque élément de notre application a été soigneusement élaboré pour offrir une expérience utilisateur exceptionnelle, reflétant notre engagement envers la qualité et l'innovation.

CONCLUSION GENERALE

En conclusion, notre mémoire sur le développement d'une application de gestion de restauration témoigne de l'implication croissante de l'informatique dans le secteur de la restauration. Nous avons pu constater comment les TIC ont ouvert de nouvelles perspectives pour améliorer l'efficacité et la rentabilité des entreprises du domaine.

Comme toute réalisation commence par l'étude de ce qui existe, nous nous sommes pas sortis de cette règle et en réalisant une étude approfondie de l'existant, identifiant les besoins spécifiques du secteur de la restauration et les lacunes des solutions existantes. Cette étude a conduit à la fin à la réalisation d'une application sur mesure qui simplifie les tâches administratives complexes et offre des fonctionnalités avancées de gestion.

L'utilisation d'une approche de développement orienté objet, basée sur le cycle de vie UP, nous a permis de répondre efficacement aux besoins et aux retours des utilisateurs, tout en travaillant en étroite collaboration avec l'équipe de l'entreprise. Cette approche itérative a favorisé une amélioration continue de l'application et a assuré qu'elle réponde de manière précise aux exigences spécifiques du secteur de la restauration.

La conception de notre application de gestion de restauration a été rigoureuse et méthodique, guidée par des diagrammes détaillés et une réflexion approfondie sur l'architecture logicielle. Nous avons choisi d'utiliser le langage de modélisation UML (Unified Modeling Language) pour représenter et visualiser les différentes composantes de l'application, y compris les cas d'utilisation, les classes et les séquences.

La réalisation de l'application a permis de concrétiser notre vision et nos objectifs. Nous avons pu développer des interfaces utilisateur conviviales et intuitives, offrant une expérience fluide aux utilisateurs finaux. Les tests rigoureux effectués tout au long du processus de développement ont assuré la fiabilité et la qualité de l'application.

Ce projet de fin d'études a été une expérience enrichissante qui nous a permis de mettre en pratique nos compétences en informatique et de les associer à notre passion pour la restauration. Nous avons acquis une compréhension approfondie des enjeux spécifiques du secteur de la restauration et avons contribué de manière concrète à l'amélioration des opérations de l'entreprise Gouraya Froid.

En perspective de ce travail, la création d'un module pour le programme de fidélité, la pré-réservation, la mise sous réseau pour des chaînes de restaurants.

BIBLIOGRAPHIE

1. QUI Somme Nous ?- Gouraya Froid. [En ligne] [Citation : 25 02 2023.] <https://gourayafroid.com/qui-sommes-nous/>.
2. **CNAM ANGOULEME**. *Mcours*. [En ligne] 28 11 2001. [Citation : 28 03 2023.] https://www.mcours.net/cours/pdf/info/Methodologie_des_systemes_d_information_UML.pdf.
3. **Pierre-Alain, Muller**. *Modélisation objet avec UML*. 1997.
4. **Pascal, Roques**. *Les cahiers de programmeurs, UML 2 Modéliser une application web*. 4eme édition. s.l. : Eyrolles, 2008.
5. **KHALED, SELLAMI**. *Vers un modèle de support de l'interopérabilité des fédérations de composants logiciels*. s.l. : Diss. Université de bejaia-Abderahmane, 2005, 2005.
6. **Gilles , Roy**. *Conception de bases de données avec UML*. 1e édition. s.l. : Presses de l'Université de Québec, 2009.
7. **Blanc, Xavier, Isabelle, Mounier et Cédric, Besse**. *UML2 pour les développeurs, cours avec exercices corrigés*. s.l. : EYROLLES.
8. **Khaled, SELLAMI, Chelouah, Yanis et Farouk, Boussaa**. *ATELIER DE GENIE LOGICIEL CENTRE PROCEDURE BASE UML*. 5e édition . s.l. : EDITIONS UNIVERSITAIRES E,, 2021.
9. **Soutou, C**. *UML2 pour les bases de données*. 1ère édition. s.l. : EYROLLES, 2006.
10. WampServer. [En ligne] [Citation : 04 05 2023.] <https://www.wampserver.com/>.
11. *MySQL*. [En ligne] [Citation : 06 05 2023.] <https://www.mysql.com/fr/>.
12. *PhpMyAdmin*. [En ligne] [Citation : 06 05 2023.] <https://www.phpmyadmin.net>.
13. [En ligne] [Citation : 06 05 2023.] <https://www.geeksforgeeks.org/introduction-to-jdbc/>.
14. *techopedia*. [En ligne] [Citation : 08 05 2023.] <https://www.techopedia.com/definition/5594/java-development-kit-jdk>.
15. *Oracle*. [En ligne] [Citation : 08 05 2023.] <http://www.oracle.com/java/technologies/javase/javafxscenebuilder-info.html>.
16. *Apache NetBeans* . [En ligne] [Citation : 10 05 2023.] <https://netbeans.apache.org/>.
17. **Gauthier , Picard et Laurent , Vercouter**. *Initiation à la programmation orientée-objet*. Saint-ETIENNE : s.n., 2014.
18. *Javapoint*. [En ligne] [Citation : 10 05 2023.] <https://www.javatpoint.com/javafx-tutorial>.
19. [En ligne] [Citation : 10 05 2023.] <https://www.atinternet.com/glossaire/css>.
20. **Defrance, J**. *PHP / MySQL avec Dreamweaver 8*. s.l. : Eyrolles, 2006.

21. *Developpez*. [En ligne] [Citation : 13 05 2023.] <http://dico.developpez.com/html/3020-Conception-MVC-Model-View-Controller.php>.

22. **Wood, Daryl**. *“Learning WAMP Server Development*. s.l. : InfiniteSkills, 2014. 9781771372848.

Résumé

Le but de ce travail est de réaliser une application desktop de gestion de restauration en utilisant l'environnement de développement (NetBeans, Wamp, Scenenbuilder). En collaboration avec l'entreprise Gouraya Froid, une étude approfondie des besoins spécifiques a été menée, conduisant à la conception d'une application sur mesure. Grâce à une approche de développement orienté objet basée sur le cycle de vie UP et à l'utilisation du langage UML pour la modélisation, conduisant à la mise en œuvre dans la fin du processus. Une application conviviale et intuitive a été développée avec les outils évoqués. Les tests rigoureux effectués ont assuré la fiabilité et la qualité de l'application. Ce projet a été une expérience enrichissante, combinant compétences informatiques et passion pour la restauration, avec une contribution concrète à l'amélioration des opérations de Gouraya Froid.

Abstract

The aim of this work is to create a desktop restoration management application using the development environment (NetBeans, Wamp, Scenenbuilder). In collaboration with Gouraya Froid, an in-depth study of specific needs was carried out, leading to the design of a custom application. Through an object-oriented development approach based on the UP lifecycle and the use of UML language for modeling, leading to implementation at the end of the process. A user-friendly and intuitive application has been developed with the tools mentioned. The rigorous tests carried out ensured the reliability and quality of the application. This project has been a rewarding experience, combining IT skills and a passion for catering, with a concrete contribution to improving Gouraya Froid's operations.