



**Projet de Fin d'Études**

En vue de l'obtention du diplôme de Licence professionnelle en Informatique

**Option** : Développement des Applications Web et Mobile (DAWM)

**THEME :**

**Conception et réalisation d'une application de gestion d'entrée-sortie :**

**Cas de l'entreprise Valamdou**

**Réalisé et présenté par :**

ATEIB ABDOULAYE YOUSOUF

**Sous la direction de :**

Mr VOUNDAÏ VALAMDOUN MAHAMAT

Année Académique : **2022-2023**

**Dédicaces**

“

*Je dédie ce modeste travail à celle qui a bercé mes rêves ma mère **MOUNIRA HASSAN** et à celui qui a nourri mes ambitions mon père **ABDOULAYE YOUSSEUF** ainsi qu'à mes frères, mes sœurs, mes camarades et amis.*

”

## Remerciements

“

« L'imagination est plus importante que la connaissance, mais la connaissance donne du pouvoir à l'imagination (**Albert Einstein 1879-1955**) ». Je suis loin d'avoir toutes les connaissances, mais n'empêche que j'ai mes géants à qui je tiens à exprimer ma plus profonde reconnaissance.

Tout d'abord, je souhaite exprimer ma gratitude envers Allah le tout puissant de m'avoir accordé la vie, du courage et de la patience nécessaires pour mener à bien ce travail.

Je tiens à remercier profondément l'équipe pédagogique et administrative de l'ENASTIC antenne de N'Djamena et celle d'Amdjarass pour leurs efforts inestimables visant à nous offrir une formation de qualité.

Je tiens également à exprimer ma plus profonde gratitude à Monsieur **VOUNDAÏ VALAMDOU MAHAMAT** qui m'a guidé et aidé pour réaliser le projet.

Je tiens aussi à exprimer ma profonde gratitude à mes condisciples de classe, mes camarades, mes collègues et amis pour les liens qui nous ont unis.

Pour finir, mes sincères remerciements à toutes les personnes ayant collaborées de près ou de loin à la réalisation de ce travail.

”

## Avant-Propos

L'école Nationale Supérieure des Technologies de l'Information et de la Communication est un établissement public d'enseignement, de recherche, d'innovation et de formations supérieures dans les domaines technologiques. Elle est créée par ordonnance n° 005/PR/2015 et est dotée d'une personnalité juridique et d'autonomie financière. L'ENASTIC, administrée par un Conseil d'Administration (CA), dirigée par un Directeur Général (DG) et placée sous la tutelle du Ministère des Postes et de l'Économie Numérique. Le ministère de l'Enseignement Supérieur, de la Recherche et de l'Innovation (MESRI) assure l'organisation pédagogique et académique. Elle est siégée à N'Djamena et possède de deux (2) autres antennes dont l'une à Amdjarass et l'autre à Sarh. L'ENASTIC a pour objectifs :

- ✚ D'assurer la formation initiale et continue dans les domaines des TIC ;
- ✚ De promouvoir la recherche scientifique et technologique.

L'école possède de départements qui sont :

- ✚ Département des Affaires Académiques, de la Scolarité et des Examens (DAASE) ;
- ✚ Département des Enseignements et de la Recherche (DER) ;
- ✚ Département de la Formation Continue et du Perfectionnement (DFCP).

L'ENASTIC antenne d'Amdjarass a été créée par décision conjointe N°047/ENASTIC/PCA/SC/2019 portant son ouverture dans la province de l'Ennedi EST, elle possède trois (3) filières dont deux (2) sont activées à savoir :

- ✚ Développement des Applications Web et Mobile (DAWM) ;
- ✚ Administration de Sécurité des Réseaux et Système (ASRS).

Elle est dirigée par :

- ✚ Un chef d'antenne ;
- ✚ Un chef chargé de matériels ;
- ✚ Un agent comptable.

Le DER dispose d'un complexe dynamique qui peut répondre aux exigences des activités pédagogiques, des travaux et de recherches. Les enseignements et encadrements sont garantis par les enseignants du supérieur, des professionnels et des experts dans le domaine des TIC. Elle a inscrit à son programme de formation un stage aux étudiants de la 3<sup>e</sup> année plus précise en S6 dans les différentes institutions. [1]

### Résumé

Dans le cadre de notre travail, nous avons développé un logiciel dédié à la gestion de l'entreprise Valamdou. Ce logiciel a pour objectif de simplifier les tâches de l'administrateur en automatisant la gestion des utilisateurs, la gestion de recette, la gestion de dépense et en fin la visualisation de la courbe.

Notre application offre des fonctionnalités permettant aux utilisateurs (administrateur) de consulter le compte de tous utilisateurs, voir les entrées-sorties et la visualisation de la courbe. De même pour le gestionnaire c'est de voir les entrées-sorties et la visualisation de la courbe. Pour la conception de cette solution informatique, on a suivi le processus de développement en V et utilisé UML pour la représentation visuelle de cette dernière.

Le logiciel a été développé sous le langage python avec ses différentes bibliothèques (matplotlib, tkcalendar, mysql\_connector), conçu pour le développement de logiciels, ensuite nous avons utilisé le MySQL et PhpMyAdmin comme SGBD et enfin le langage SQL pour interroger la base de données et faire toutes les requêtes possibles à travers le système.

En somme, notre mémoire représente la conception et la réalisation d'une application de gestion d'entrée-sortie cas de l'entreprise Valamdou, utilisant les technologies ci-dessus.

**Table des matières**

<b>Dédicaces .....</b>	<b>I</b>
<b>Remerciements .....</b>	<b>II</b>
<b>Avant-Propos.....</b>	<b>III</b>
<b>Résumé .....</b>	<b>IV</b>
<b>Liste des sigles.....</b>	<b>VIII</b>
<b>Introduction générale .....</b>	<b>1</b>
a) <b>Contexte .....</b>	<b>1</b>
b) <b>Problématique .....</b>	<b>1</b>
c) <b>Objectif général.....</b>	<b>1</b>
d) <b>Objectifs spécifiques .....</b>	<b>1</b>
e) <b>Méthodologie des recherches .....</b>	<b>2</b>
f) <b>Organisation du mémoire.....</b>	<b>2</b>
<b>CHAPITRE 1 : .....</b>	<b>3</b>
<b>ÉTUDE PREALABLE, ANALYSE ET SPECIFICATION DES BESOINS</b>	<b>3</b>
<b>1.1. Introduction.....</b>	<b>4</b>
<b>1.2. Présentation de l'entreprise.....</b>	<b>4</b>
<b>1.3. Présentation du projet .....</b>	<b>5</b>
<b>1.4. Étude de l'existant .....</b>	<b>5</b>
<b>1.5. La spécification des besoins .....</b>	<b>7</b>
1.6.1. <b>Besoins fonctionnels .....</b>	<b>7</b>
1.6.2. <b>Besoins non-fonctionnels .....</b>	<b>7</b>
<b>1.6. UML.....</b>	<b>8</b>
1.7.1. <b>Cycle de vie en génie logiciel .....</b>	<b>8</b>
1.7.2. <b>Le processus en V .....</b>	<b>9</b>
<b>1.7. Conclusion .....</b>	<b>11</b>
<b>CHAPITRE 2 : CONCEPTION.....</b>	<b>12</b>
<b>2.1. Introduction .....</b>	<b>13</b>
2.2. <b>Diagramme de cas d'utilisation .....</b>	<b>13</b>
2.2.1. <b>Le diagramme de cas d'utilisation « général » .....</b>	<b>13</b>

2.2.2. Le diagramme de cas d'utilisation « administrateur » .....	14
2.2.3. Le diagramme de cas d'utilisation « gestionnaire » .....	15
2.3. Diagramme de séquences .....	15
2.3.1. Le diagramme de séquence « authentification » .....	15
2.3.2. Le diagramme de séquence « créer un compte » .....	17
2.3.3. Le diagramme de séquence « modifier un compte » .....	17
2.3.4. Le diagramme de séquence « supprimer un compte » .....	18
2.3.5. Le diagramme de séquence « ajouter un enregistrement » .....	19
2.3.6. Le diagramme de séquence « modifier un enregistrement » .....	20
2.3.7. Le diagramme de séquence « supprimer un enregistrement » .....	21
2.3.8. Le diagramme de séquence « password oublié » .....	22
2.3.9. Le diagramme de séquence « voir courbe » .....	23
2.3.10. Le diagramme de séquence « déconnecter » .....	24
2.4. Le diagramme de classe .....	25
2.5. Le modèle MVC .....	26
2.6. Conclusion .....	27
<b>CHAPITRE 3 : RÉALISATION .....</b>	<b>28</b>
<b>3.1. Introduction .....</b>	<b>29</b>
<b>3.2. L'environnement matériel .....</b>	<b>29</b>
<b>3.3. Outils de développement .....</b>	<b>29</b>
<b>3.4. Présentation de l'application .....</b>	<b>31</b>
<b>3.5. Conclusion .....</b>	<b>40</b>
<b>Conclusion générale .....</b>	<b>41</b>
<b>Bibliographie .....</b>	<b>42</b>
<b>Annexe .....</b>	<b>43</b>

## Liste des figures

<b>Figure 1. 1 Organigramme</b> .....	<b>5</b>
<b>Figure 1. 2 modèle V</b> .....	<b>10</b>
<b>Figure 2. 1 cas d'utilisation général</b> .....	13
<b>Figure 2. 2 cas d'utilisation administrateur</b> .....	14
<b>Figure 2. 3 cas d'utilisation gestionnaire</b> .....	15
<b>Figure 2. 4 séquence authentification</b> .....	16
<b>Figure 2. 5 séquence créer un compte</b> .....	17
<b>Figure 2. 6 séquence modifier un compte</b> .....	18
<b>Figure 2. 7 séquence supprimer un compte</b> .....	19
<b>Figure 2. 8 séquence ajouter un enregistrement</b> .....	20
<b>Figure 2. 9 séquence modifier un enregistrement</b> .....	21
<b>Figure 2. 10 supprimer un enregistrement</b> .....	22
<b>Figure 2. 11 password oublié</b> .....	23
<b>Figure 2. 12 diagramme afficher courbe</b> .....	24
<b>Figure 2. 13 séquence déconnecté</b> .....	25
<b>Figure 2. 14 diagramme de classe</b> .....	26
<b>Figure 3. 1 logo UML</b> .....	29
<b>Figure 3. 2 logo python</b> .....	29
<b>Figure 3. 3 logo VSC</b> .....	30
<b>Figure 3. 4 logo MySQL</b> .....	30
<b>Figure 3. 5 logo PhpMyAdmin</b> .....	31
<b>Figure 3. 6 wampserver</b> .....	31
<b>Figure 3. 7 environnement VSC</b> .....	32
<b>Figure 3. 8 interface login</b> .....	32
<b>Figure 3. 9 interface créer compte</b> .....	33
<b>Figure 3. 10 interface password_oublié</b> .....	33
<b>Figure 3. 11 interface modifier_supprimer_user</b> .....	34
<b>Figure 3. 12 interface administrateur</b> .....	34
<b>Figure 3. 13 : histogramme_unique</b> .....	36
<b>Figure 3. 14 histogramme_générale</b> .....	36
<b>Figure 3. 15 pourcentage unique</b> .....	37
<b>Figure 3. 16 pourcentage générale</b> .....	37
<b>Figure 3. 17 interface gestionnaire</b> .....	38

**Liste des sigles**

**ENASTIC** : Ecole Nationale de Technologies de L'information et de la Communication

**DAWM** : Développement des Applications Web et Mobile

**ASRS** : Administration Sécurité de Réseaux et Système

**UML** : Unified Modeling Language

**SQL** : Structured Query Language

**SGBD** : Système de Gestion des Base de Données

**SGBDR** : Système de Gestion des Base de Données Relationnels

**TIC** : Technologies de l'Information et de la Communication

**VSC** : Visual Studio Code

**DG** : Directeur Général

**CA** : Conseil d'Administration

**MPNTI** : Ministère des Postes et des nouvelles Technologies de l'Information

**MESRI** : Ministère de l'Enseignement Supérieur, de la Recherche et de l'Innovation

**DAASE** : Département des Affaires Académiques, de la Scolarité et des Examens

**DER** : Département des Enseignements et de la Recherche

**DFPC** : Département de la Formation Continue et du Perfectionnement

**BD** : Base de données

**MVC** : Modèle Vue Contrôleur

**PFE** : Projet de Fin d'Etudes

**WAMPSEVER** : Windows Apache Php Server

## **Introduction générale**

Dans notre ère actuelle, les projets technologiques ont révolutionné tous les secteurs grâce à l'Informatique, une discipline dédiée aux traitements automatisés de l'information. L'impact de l'Informatique sur le développement et la bonne gestion des entreprises et autres institutions est indéniable.

### **a) Contexte**

Avant l'arrivée des ordinateurs, toutes les informations étaient enregistrées manuellement sur des supports en papier, ce qui entraînait de nombreux problèmes tels que la perte de temps, la recherche d'une information, la saisie avec des erreurs et la perte de données en cas de perte de document.

De nos jours, l'ordinateur est le moyen le plus fiable pour le traitement et la sauvegarde de l'information. Cette invention a permis l'informatisation des systèmes de gestion de données des entreprises et institutions.

### **b) Problématique**

La gestion d'entrée-sortie (gestion de fonds de l'entreprise) est un aspect très important dans la gestion des opérations quotidiennes. A cet effet, une gestion d'entrée-sortie devient très efficace pour les entreprises qui ont une vue à long-terme plus précisément « **Entreprise Valamdou** » afin de connaître les nombres exactes ses employés avec tous leurs informations possibles et voir la courbe statistique pour savoir où en est-on le avec niveau financier avant son appréciation et relativement à une période donnée.

### **c) Objectif général**

L'objectif général de ce projet est d'explorer en profondeur le concept de gestion informatisée des activités (entrée et sortie) des entreprises. Cette approche présente des avantages significatifs pour un développeur de bien maîtriser théoriquement et pratiquement un langage de programmation afin de mieux orienter le sujet.

### **d) Objectifs spécifiques**

L'objectif spécifique de ce travail est de concevoir et de déployer une application interactive, fiable et facile à manipuler afin de bien maîtriser la gestion centralisée des

opérations de l'Entreprise Valamdou en utilisant le langage de programmation python et ses autres accessoires.

### e) **Méthodologie des recherches**

La méthodologie de recherche est un cadre qui consiste à résoudre le problème de recherche en utilisant les meilleures méthodes et les plus réalisables pour mener à bien la recherche tout en s'alignant sur le but et les objectifs de notre recherche.

Cette méthodologie met en avant les différentes étapes et règles de mise en œuvre des modèles de développement à chaque étape de conception d'une application, de la capacité d'aide, de réutilisabilité d'expériences, d'uniformisation des représentations et de communication entre les différentes personnes impliquées par la définition de ces modèles structurés, de représentations graphiques et de règles de mise en œuvre de ces modèles à chaque étape du développement.

### f) **Organisation du mémoire**

Pour le besoin de notre étude, notre travail est structuré en 3 chapitres comme suit :

- ✚ Le premier chapitre « Étude préalable, analyse et spécification des besoins » qui sera dédié à la présentation de l'entreprise « Valamdou », clarifiant son contexte et ses particularités. On procède ensuite à une analyse approfondie des besoins, ce qui aidera de délimiter clairement notre problème que nous souhaitons résoudre ;
- ✚ Le deuxième chapitre « conception », montrera le cycle de développement de notre système ainsi que les différents diagrammes (cas d'utilisation, séquence, classe) en utilisant le langage UML ;
- ✚ Le troisième et dernier chapitre « réalisation » qui décrira et montrera les différents outils utilisés pour la réalisation de ce modeste travail. Il montrera aussi les différentes interfaces de notre système.

En somme, nous allons conclure notre travail par un résumé général qui montrera les principales contributions de notre recherche.

En structurant notre mémoire de cette manière, nous souhaitons fournir une vue d'ensemble complète de notre travail, depuis l'Étude préalable, analyse et spécifications des besoins jusqu'à la réalisation concrète de l'application.

**CHAPITRE 1 :**  
**ÉTUDE PREALABLE, ANALYSE ET SPECIFICATION DES BESOINS**

### 1.1. Introduction

L'étude préalable joue un rôle essentiel en tant qu'étape préparatoire de notre projet. Dans un premier temps, nous procéderons à la présentation détaillée de notre entreprise. Ensuite, nous définirons les objectifs et les besoins que nous visons atteindre durant ce chapitre.

### 1.2. Présentation de l'entreprise

L'entreprise Valamdou est une entreprise privée créée en février 2016 au quartier Farcha dans le 1<sup>er</sup> arrondissement de la ville de N'Djamena. Elle a pour objectifs de :

- ✚ Garantir la disponibilité d'un nombre maximal des moulins au sein de la commune du 1<sup>er</sup> arrondissement ;
- ✚ Garantir la disponibilité des activités ;
- ✚ Bien veiller sur la santé de la population en leur offrant du bon farine (mil, mais etc.), propre et sain.

Afin d'assurer le bon fonctionnement de celle-ci, l'administrateur doit tout d'abord superviser la gestion d'utilisateurs (gestionnaires), l'enregistrement d'entrée-sortie et voir la courbe statistique afin d'avoir une idée sur l'évolution des activités de l'entreprise. Le gestionnaire quant à lui, se charge de l'enregistrement des entrées-sorties journalière, et l'employé quant à lui se charge de bon fonctionnement du moulin, de l'accueil des clients et les conseillent dans le choix de grains et celui de la bonne farine.

La figure ci-après nous montre l'organigramme de l'entreprise :

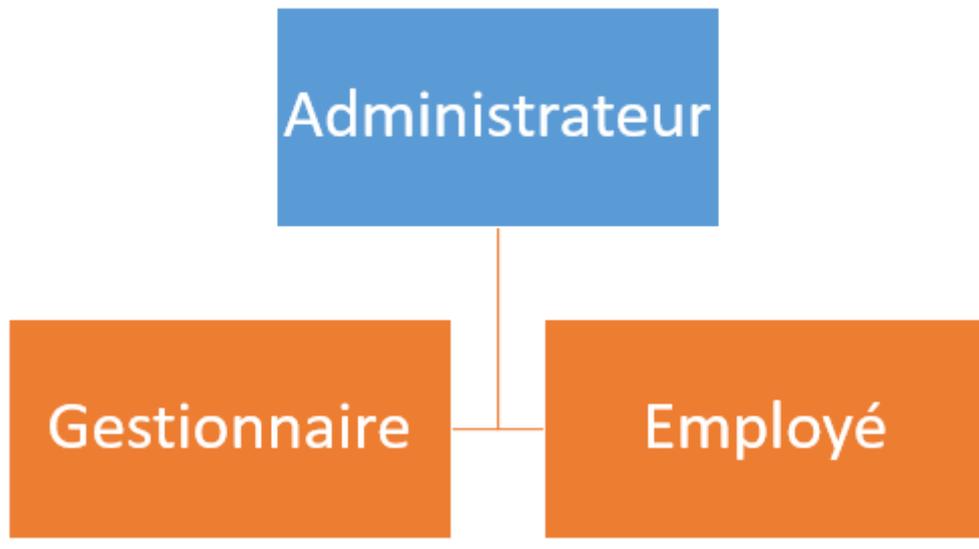


Figure 1. 1 Organigramme

### 1.3. Présentation du projet

Le projet que nous souhaitons réaliser porte sur la « conception et la réalisation d'une application de gestion d'entrée-sorties » visant à répondre aux besoins de l'entreprise Valamdou.

### 1.4. Étude de l'existant

L'étude de l'existant comprend quatre 4 phases qui sont entres-autres :

- ✚ La description ;
- ✚ La problématique ;
- ✚ Le critique ;
- ✚ La solution proposée.

✓ **La description**

En tant que stagiaire dans cette entreprise, nous avons constaté qu'au sein de celle-ci, existe un ordinateur utilisé pour de tâches administratives et la plateforme déployée est développée sous Microsoft Excel. Il est utilisé pour la gestion de données, enregistrement d'entrée-sorties etc...

✓ **Le critique**

L'existant dans cette entreprise avec l'utilisation de Microsoft Excel, présente à la fois des **avantages** et des **inconvénients** qui sont :

- **Les avantages**

Le logiciel Microsoft Excel est largement répandu et facile d'y accéder, il s'est familiarisé avec n'importe quel utilisateur au sein de l'entreprise.

- **Les inconvénients**

Le logiciel Microsoft Excel est principalement destiné à la création de documents et aux calculs de base et ne peut pas répondre à tous les fonctionnalités demandées.

La saisie manuelle de données ce qui peut entraîner des erreurs humaines et la perte de temps, le risque de la duplication de données ou la commission des erreurs pendant la saisie.

✓ **La problématique**

Pour améliorer l'efficacité de gestion d'entrée-sortie (gestion de la caisse) tout en tenant compte des limites actuelles liées à l'utilisation de Microsoft Excel, on a pu soulever quelques questions essentielles ci-après :

- Comment savoir qu'on a combien de source (moulins en opération) ?
- Combien de gens qui travaillent au sein de l'entreprise ? avec quel rôle et salaire ?
- Quelle source (moulin) qui fait beaucoup de recette par rapport aux autres ? beaucoup de dépense ? dans quel mois ? dans quel jour ?
- Comment garantir la sécurité des données et la protection de la confidentialité des informations des utilisateurs ?

Ces questions soulignent la nécessité d'explorer des solutions plus avancées et spécifiquement adaptées aux besoins de l'entreprise Valamdou afin de surmonter les limites actuelles et d'optimiser le processus de la gestion de caisse.

✓ **La solution proposée**

Pour régler ce problème, nous proposons le développement d'une application de gestion d'entrée-sorties destinée au personnel de l'entreprise pour bien gérer la caisse (entrée-sortie).

### 1.5. La spécification des besoins

Dans cette partie, nous identifierons les exigences fonctionnelles et non-fonctionnelles du système à concevoir. Certaines exigences sont ajoutées pour clarifier d'avantage les besoins des utilisateurs.

#### 1.6.1. Besoins fonctionnels

Elles expliquent les attentes de chaque intervenant de notre future application. Avant la conception d'une solution informatique, il est essentiel de satisfaire ces besoins afin de définir clairement le champ d'action du système et d'assurer la traçabilité des besoins tout au long de la phase développement. Le présent système satisfait les besoins fonctionnelles suivants :

- ✚ Gestion des utilisateurs : cette opération consiste à créer, modifier et supprimer un utilisateur par l'administrateur ;
- ✚ Gestion de la caisse : cette opération consiste à enregistrer, modifier et supprimer une entrée ;
- ✚ Gestion de la courbe : cette opération consiste à voir la courbe représentative en temps réel de la recette, de la dépense et du bilan de chaque source.

#### 1.6.2. Besoins non-fonctionnels

Les besoins non-fonctionnels de notre application font référence aux critères et aux contraintes qui concernent la performance, la sécurité, l'utilisabilité et d'autres aspects liés au fonctionnement global de l'application. Cette dernière doit répondre aux exigences suivantes :

- ✚ Performance : il s'agit de la capacité de l'application à répondre aux demandes des utilisateurs et à traiter efficacement les opérations. Cela peut inclure des exigences de temps de réponse, de temps de changement des interfaces, de capacité de traitement, de disponibilité etc.
- ✚ Sécurité : elle concerne la protection des données, la confidentialité des utilisateurs et la prévention des accès non autorisés. Cela peut inclure des exigences telles que l'authentification, l'autorisation des utilisateurs et le chiffrement des données, la protection contre les attaques et les vulnérabilités ;
- ✚ Accessibilité : l'application doit être mobile (responsive), ce que signifie qu'elle doit être accessible et fonctionnelle pour l'administrateur et/ou le gestionnaire, même en dehors de l'entreprise. Cela implique que l'application doit être compatible et adoptable avec

n'importe quel écran, permettant à l'administrateur ou le gestionnaire de se connecter et d'accéder aux fonctionnalités de l'application à tout moment et de n'importe où.

### 1.6. UML

Le processus de développement décrit une méthode du développement logiciel. Il établit une séquence d'étapes, en partie organisée, qui contribuent à l'acquisition d'un système existant. [5]

#### 1.7.1. Cycle de vie en génie logiciel

Il est un processus itératif pour le développement d'un logiciel, qui commence par la spécification des exigences et se termine par la maintenance. Il comporte les étapes telles que prise en compte du contexte, analyse des besoins, spécification, conception, implémentation, test, déploiement, maintenance et le support du produit.

Chaque projet est différent mais généralement il comprend au minimum les activités suivantes :

- ✚ Définition du besoin : elle permet de définir les objectifs du projet et sa stratégie globale ;
- ✚ Conception générale : elle est l'élaboration théorique de la solution et la préparation de l'architecture générale ;
- ✚ Conception détaillée : c'est la phase durant laquelle les exigences sont transformées en caractéristiques spécifiques.
- ✚ Développement : c'est la traduction des fonctionnalités définies dans la phase de conception par son implémentation en utilisant une technologie appropriée ;
- ✚ Test, c'est est une phase importante du cycle de vie du génie logiciel. Son objectif est de vérifier que le logiciel fonctionne comme prévu et détecter les éventuelles erreurs (bugs).
- ✚ Déploiement : il consiste à installer le logiciel finalisé dans l'environnement de production (entreprise, organisation, etc...) afin qu'il soit accessible aux utilisateurs cibles ;
- ✚ Maintenance : c'est la partie finale du cycle de vie du génie logiciel et est présente tout au long de la vie du logiciel. Cependant, il existe plusieurs processus de cycle de vie en génie logiciel tels que :
  - ✓ Le cycle de vie en V ;
  - ✓ Le cycle de vie en cascade ;
  - ✓ Le cycle de vie spirale etc. Dans tout cela nous avons choisi le processus de cycle de vie en V. [4]

### 1.7.2. Le processus en V

Le modèle en V est une extension du modèle en cascade et se caractérise par sa représentation graphique sous la forme V. Il met l'accent sur la relation entre les différentes phases du développement logiciel. Ses principales étapes sont :

- ✚ Expression des besoins : Les besoins et les exigences recueillis et documentés en détail.
- ✚ Spécification fonctionnelle : Les besoins et les exigences du système sont recueillis, analysés et documentés. Elle implique la compréhension de fonctionnalités et les performances attendues du système.
- ✚ Conception architecturale : Cette étape consiste à définir l'architecture globale du système.
- ✚ Conception détaillée : Elle représente les modules ou les sous-systèmes identifiés dans la conception précédente sont spécifiés plus en détail dans cette phase.
- ✚ Codage et réalisation : Il est appelé aussi implémentation, cette phase consiste à traduire la conception en code source. Les développeurs et les programmeurs écrivent des lignes de code (programment), mettent en œuvre les fonctionnalités et les modules définies dans la conception détaillée et c'est ici que notre fameux logiciel sera créé.
- ✚ Tests unitaires : Au fur et à mesure que le code est écrit, des tests unitaires sont réalisés pour la vérification du bon fonctionnement du logiciel.
- ✚ Tests d'intégrations : Une fois que tous les modules sont développés et testés individuellement, ils seront intégrés pour la formation complète du système. Des tests d'intégrations sont effectués pour vérifier la communication parfaite entre les différents modules.
- ✚ Test de validation et de vérification : Cette étape comprend deux parties essentielles qui sont entre autres :
  - ✓ La validation : Les tests sont effectués pour vérifier que le logiciel répond aux besoins spécifiés ;
  - ✓ La vérification : Elle consiste à s'assurer que le logiciel répond aux besoins du client.
- ✚ Test de recette : C'est une étape critique dans le processus de développement, car il permet de s'assurer que le logiciel répond aux besoins et aux attentes du client. Cela implique la minimisation de risques d'erreurs ou de problèmes une fois que le logiciel déployé. [3]

La figure ci-dessous résume le modèle en V :

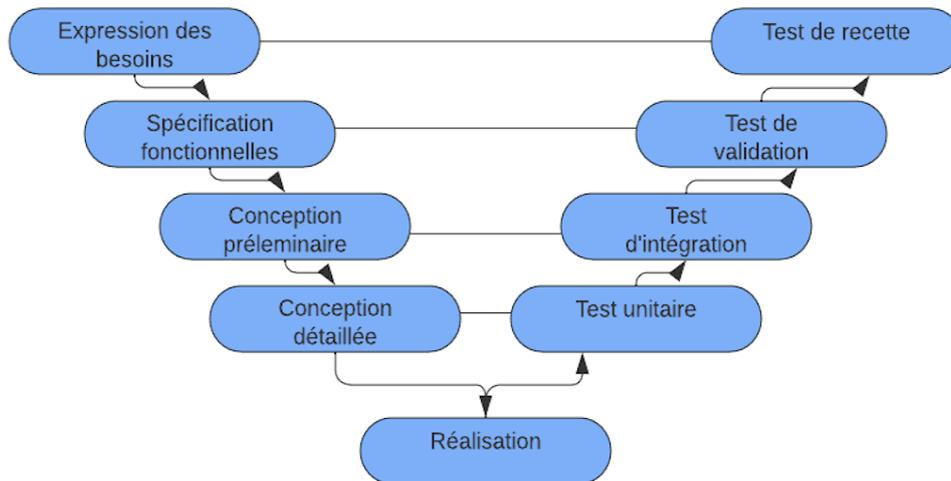


Figure 1. 2 modèle V

### 1.7. Conclusion

Dans ce chapitre, nous avons fait la présentation détaillée de l'organigramme de l'entreprise ainsi que les objectifs à atteindre. Ensuite, nous avons parlé de la phase analyse et spécification des besoins. Enfin, l'examen de la situation actuelle, nous avons pu réaliser une étude conceptuelle de la solution proposée. Ainsi dans le chapitre qui suit, nous parlerons de la conception.

**CHAPITRE 2 : CONCEPTION**

## 2.1. Introduction

La conception de toute solution logicielle est une étape cruciale dans la réalisation d'une application. Pour ce fait, notre démarche va s'appuyer sur le langage UML, conçu pour la visualisation, la spécification et la conception des systèmes. UML est un langage unifié pour la modélisation dans le cadre de la conception orientée objet. La démarche que nous avons adoptée pour la conception de l'application s'appuie sur quatre (4) étapes :

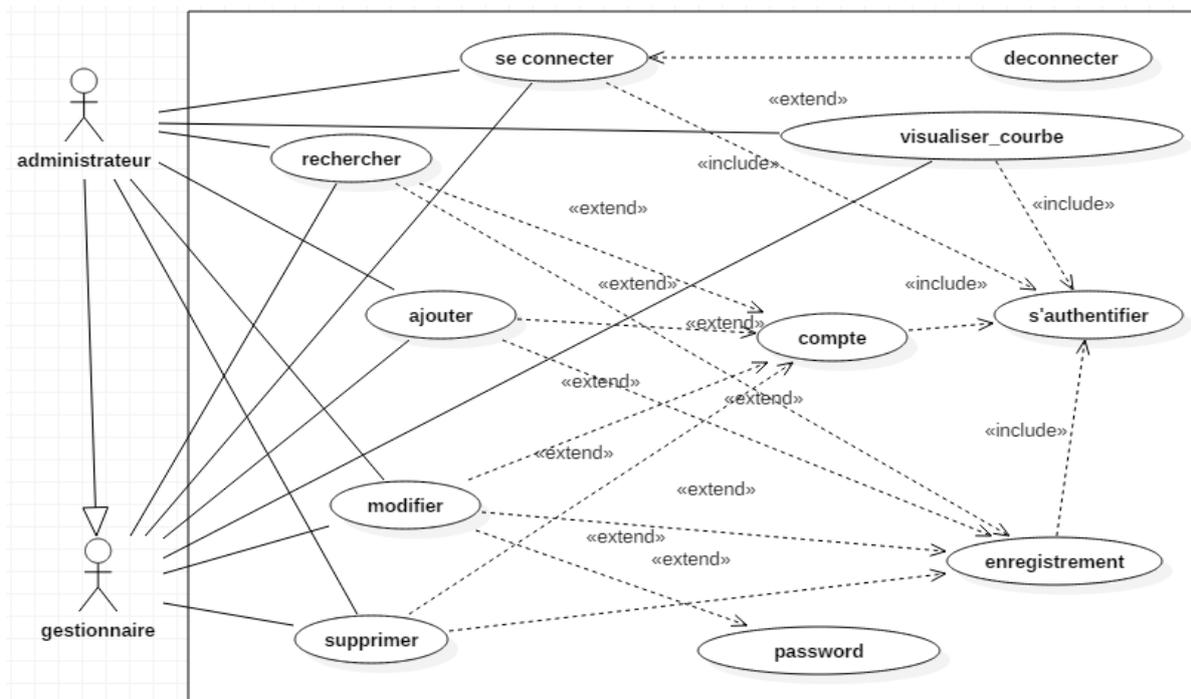
1. Identification des acteurs et besoins ;
2. Identification et représentation des cas d'utilisations ;
3. Élaboration des diagrammes de séquences ;
4. Élaboration des diagrammes de classes.

## 2.2. Diagramme de cas d'utilisation

**Un cas d'utilisation** représente un ensemble de séquences d'actions qui sont réalisés par le système et qui produisent un résultat observable intéressant par un acteur particulier.[9]

**Un acteur** représente le rôle de l'utilisateur dans le système, il est associé à un cas d'utilisation. Dans notre cas nous distinguons les cas d'utilisations suivants :

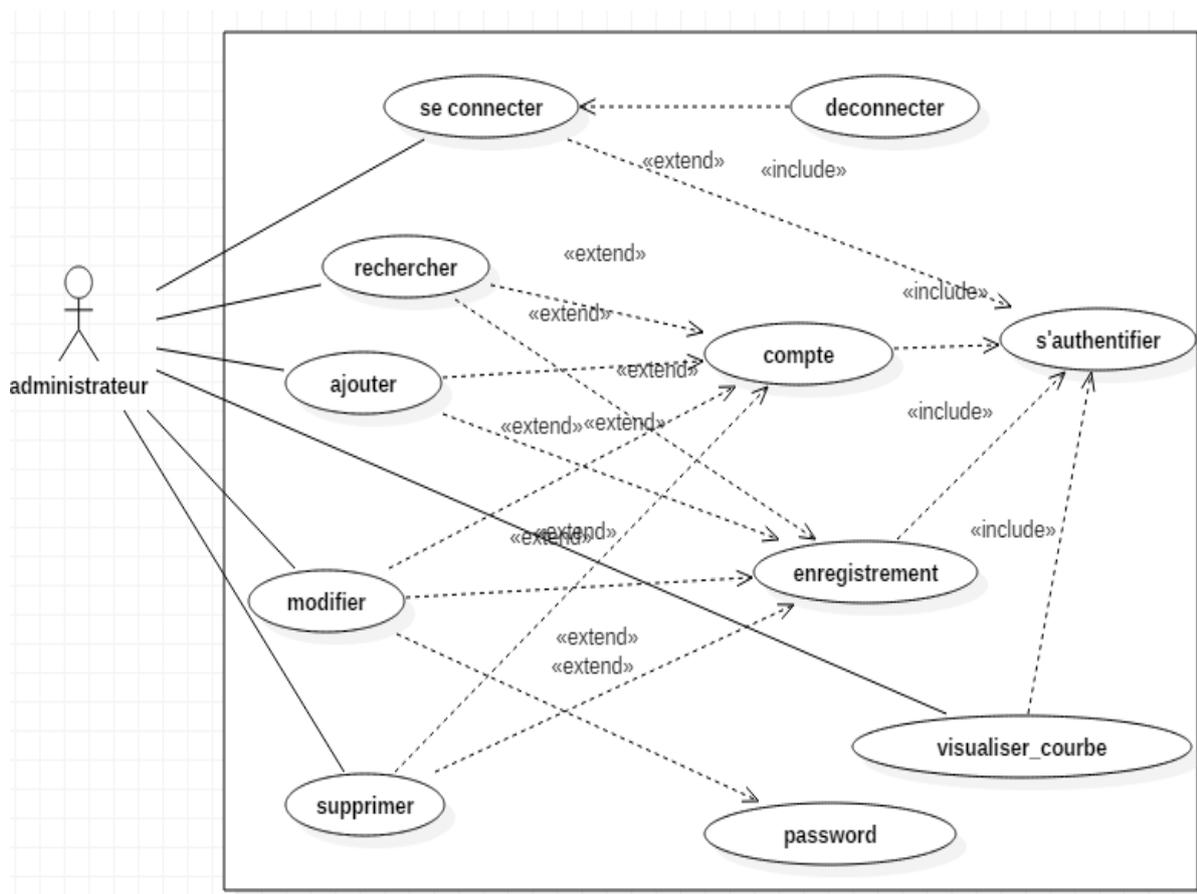
### 2.2.1. Le diagramme de cas d'utilisation « général »



*Figure 2. 1 cas d'utilisation général*

- ✚ L'administrateur peut :
  - ✓ Se connecter en s'authentifiant ;
  - ✓ Ajouter, modifier et supprimer un compte ;
  - ✓ Ajouter, modifier et supprimer un enregistrement ;
  - ✓ Modifier son password ;
  - ✓ Visualiser la courbe statistique ;
  - ✓ Se déconnecter.
- ✚ Le gestionnaire peut :
  - ✓ Se connecter en s'authentifiant ;
  - ✓ Ajouter, modifier et supprimer un enregistrement ;
  - ✓ Modifier son password ;
  - ✓ Visualiser la courbe statistique ;
  - ✓ Se déconnecter.

**2.2.2. Le diagramme de cas d'utilisation « administrateur »**



*Figure 2. 2 cas d'utilisation administrateur*

2.2.3. Le diagramme de cas d'utilisation « gestionnaire »

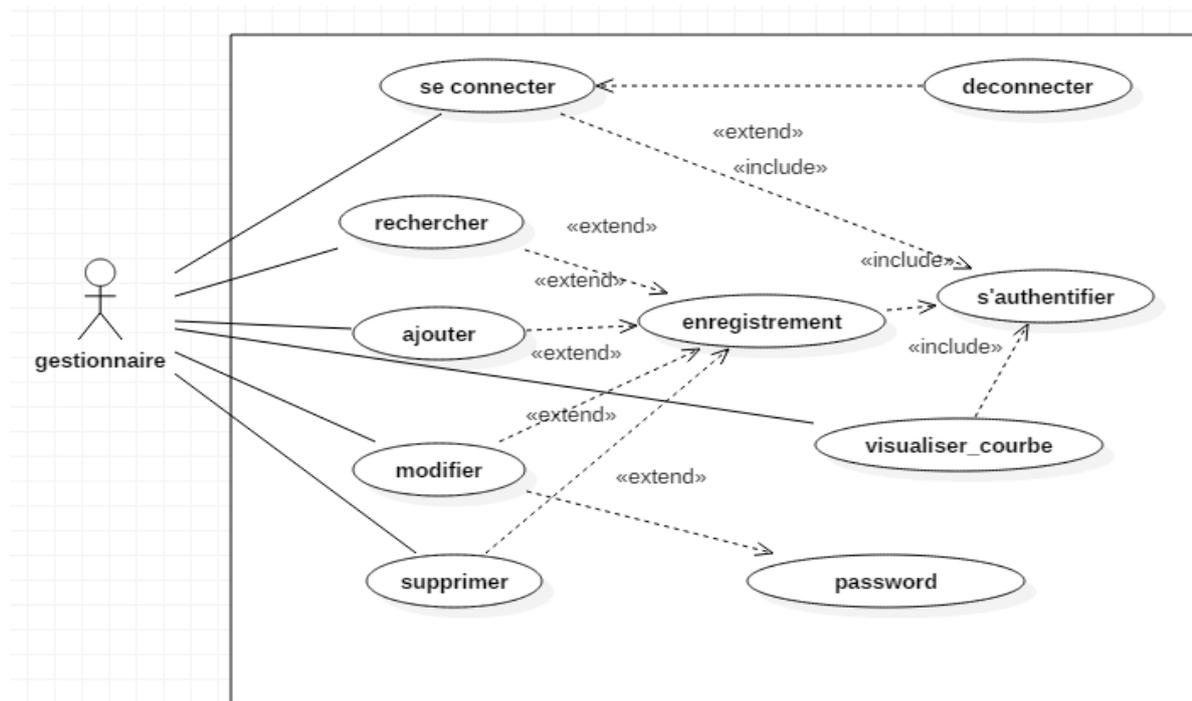
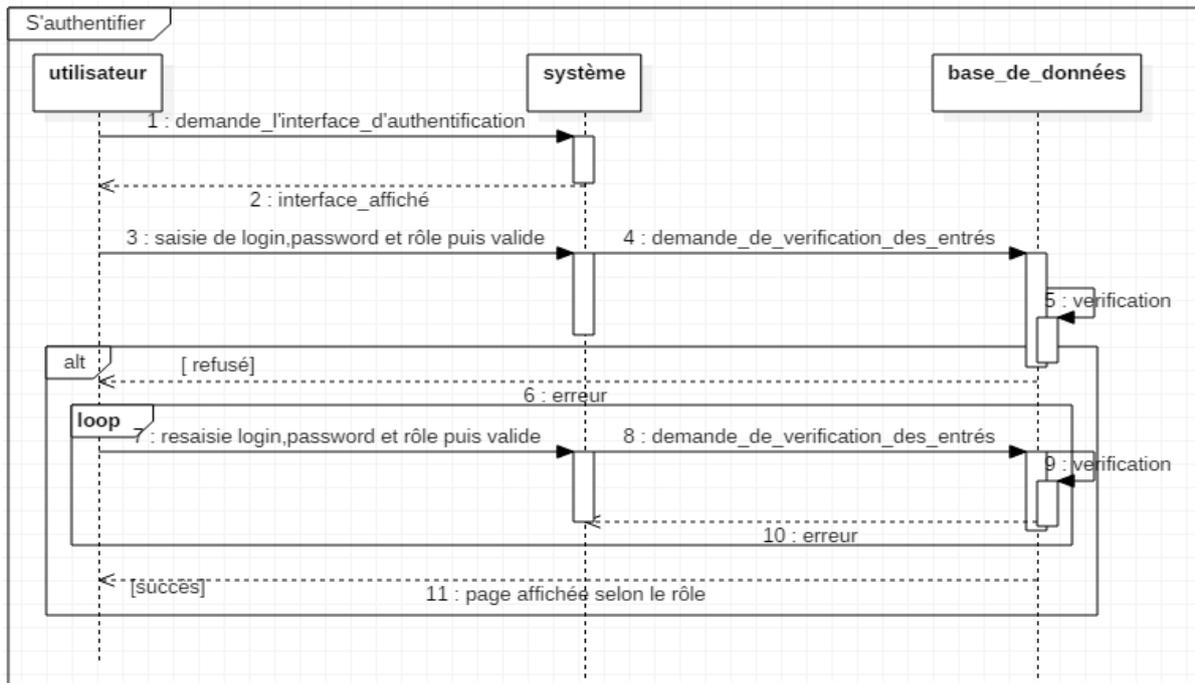


Figure 2. 3 cas d'utilisation gestionnaire

2.3. Diagramme de séquences

Un **diagramme de séquence** est l'un de types des diagrammes qui clarifie les interactions entre les utilisateurs et le système, il montre aussi les interactions et les messages échangés entre les utilisateurs et celui-ci.

2.3.1. Le diagramme de séquence « authentification »



**Figure 2. 4 séquence authentification**

- 1- L'utilisateur lance l'application et l'interface login lui sera affiché ;
- 2- Ce dernier saisie son login (username), son password et son rôle ;
- 3- Le système interroge la base de données pour vérifier les entrées ;
- 4- Si les entées sont correctes l'interface sera affichée selon le rôle choisit ;
- 5- Sinon un message d'erreur sera affiché et l'utilisateur répéteras l'action jusqu'à ce qu'à ce que le login, le password et le rôle soient vrais(corrects).

2.3.2. Le diagramme de séquence « créer un compte »

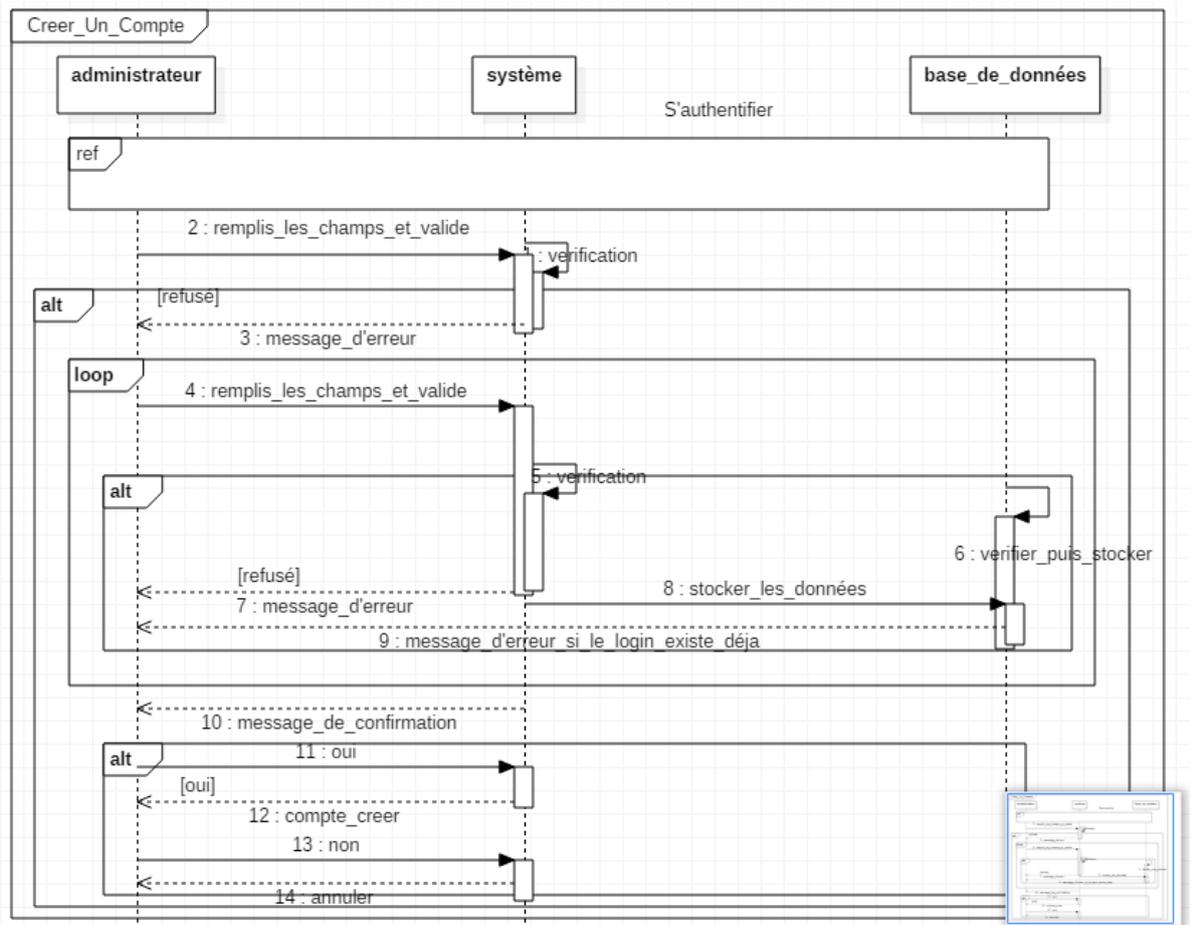


Figure 2. 5 séquence créer un compte

- 1- Une fois l’authentification est faite celui-ci (l’administrateur) sera dirigé vers l’interface créer compte s’il a cliqué sur le bouton « créer compte » ;
- 2- Il remplit les champs et valide ;
- 3- Le système vérifie les entrées ;
- 4- Si y’a erreur ça l’affiche ;
- 5- Sinon, il envoie les données vers la BD pour le stockage ;
- 6- La base de données vérifie et/ou stock ;
- 7- Elle envoie un résultat à l’interface et un message de réussite sera affiché sinon un message d’erreur ;
- 8- Un message de confirmation sera affiché à celui-ci et il choisira oui ou non ;
- 9- Si cela est fait un message de réussite sera affiché.

2.3.3. Le diagramme de séquence « modifier un compte »

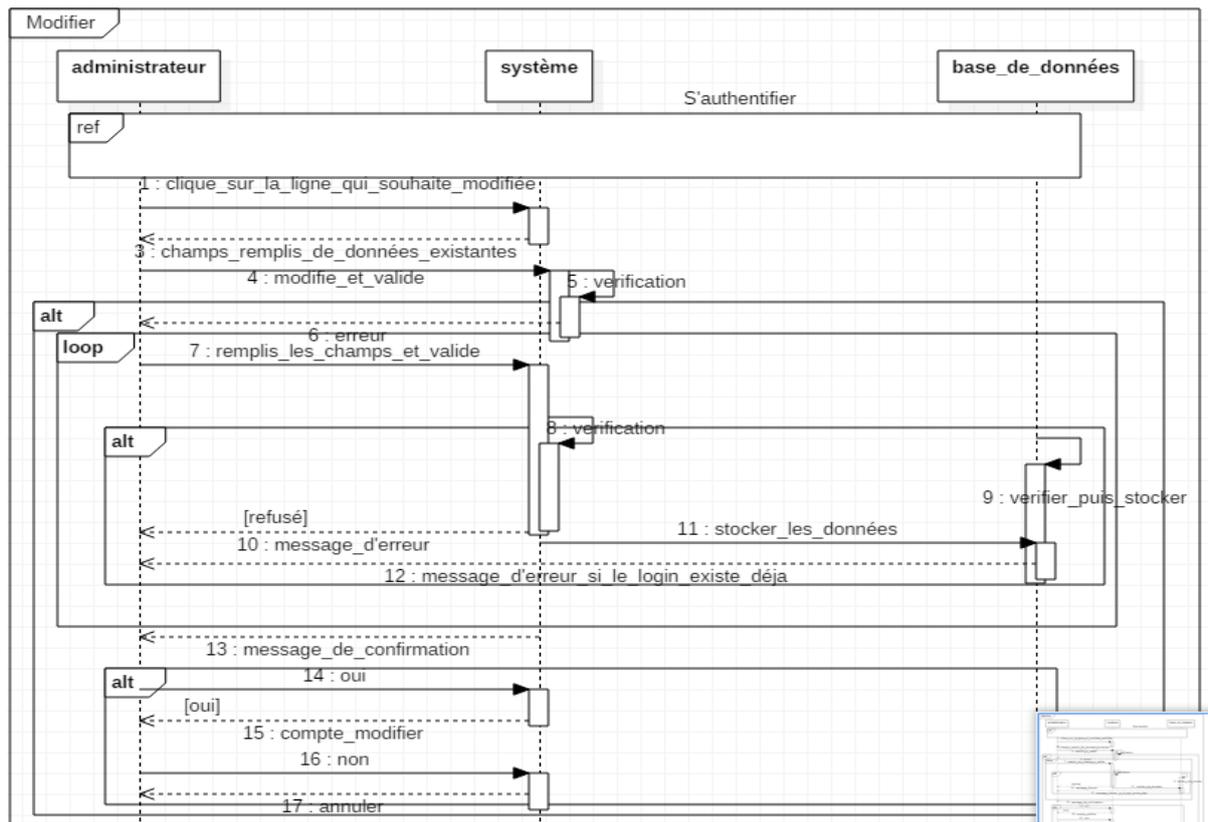
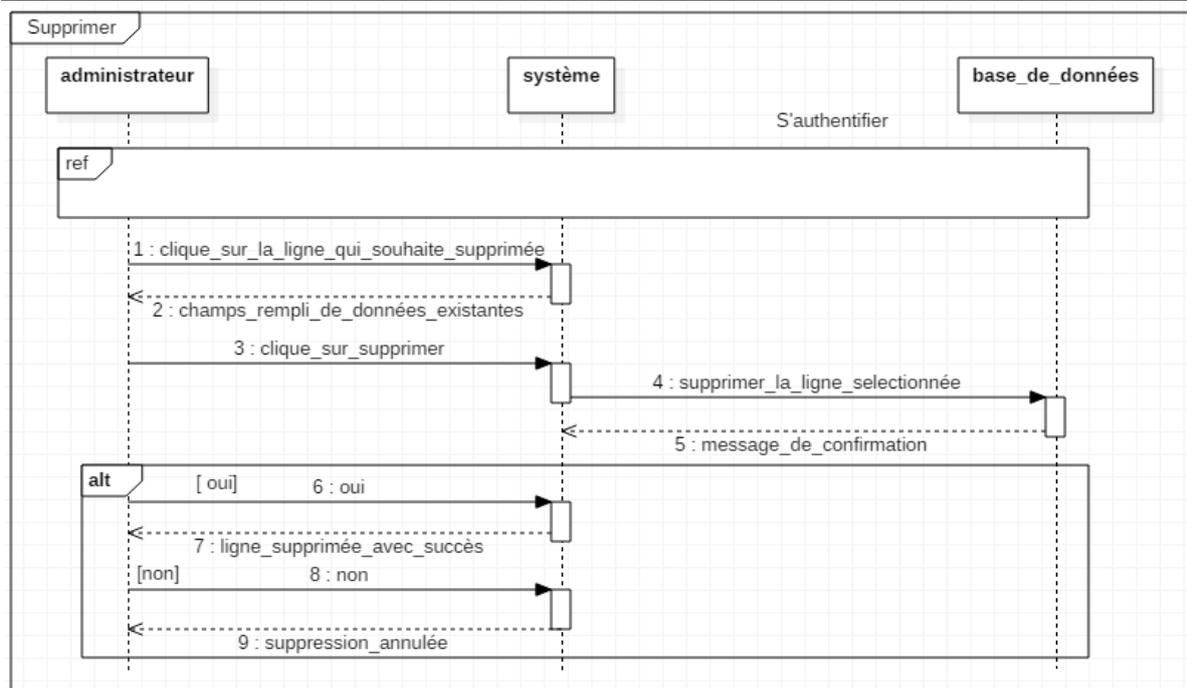


Figure 2. 6 séquence modifier un compte

- 1- Une fois l'authentification est faite Celui-ci (l'administrateur) sera dirigé vers l'interface administrateur ;
- 2- Il clique sur le menu « cliquez-ici » et une liste lui sera présentée ensuite il choisira créer compte et l'interface s'affichera ;
- 3- Il clique sur la ligne qu'il souhaite modifier ;
- 4- Les champs seront remplis des données existantes ;
- 5- L'administrateur modifie là où il souhaite modifier et valide ;
- 6- Le système vérifie les entrées et un message sera affiché au cas d'erreur, sinon il renvoie les données vers le stockage ;
- 7- Un message de confirmation sera affiché à celui-ci et il choisira oui ou non ;
- 8- Si cela est fait un message de réussite sera affiché et les données seront mises à jour.

#### 2.3.4. Le diagramme de séquence « supprimer un compte »



**Figure 2. 7 séquence supprimer un compte**

- 1- Une fois l’authentification est faite celui-ci (l’administrateur) sera dirigé vers l’interface administrateur ;
- 2- Il clique sur le menu « cliquez-ici » et une liste lui sera présentée ensuite il choisira créer compte et l’interface s’affichera ;
- 3- Il clique sur la ligne qu’il souhaite supprimer ;
- 4- Les champs seront remplis des données existantes ;
- 5- L’administrateur clique sur supprimer;
- 6- Le système envoie un message à la base de données en soi-disant supprime les données qui ont l’identifiant tel ;
- 7- Un message de confirmation sera affiché à celui-ci et il choisira oui ou non ;
- 8- Un message de réussite sera affiché.

**2.3.5. Le diagramme de séquence « ajouter un enregistrement »**

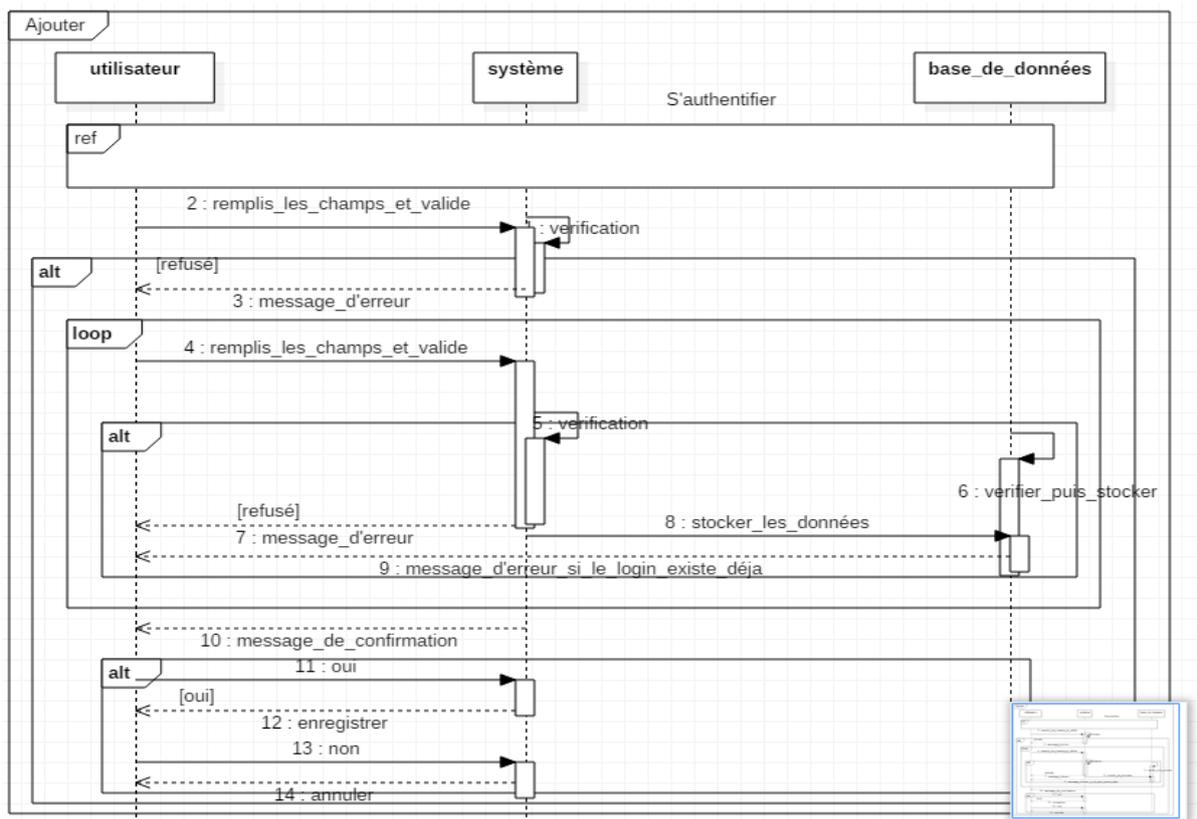
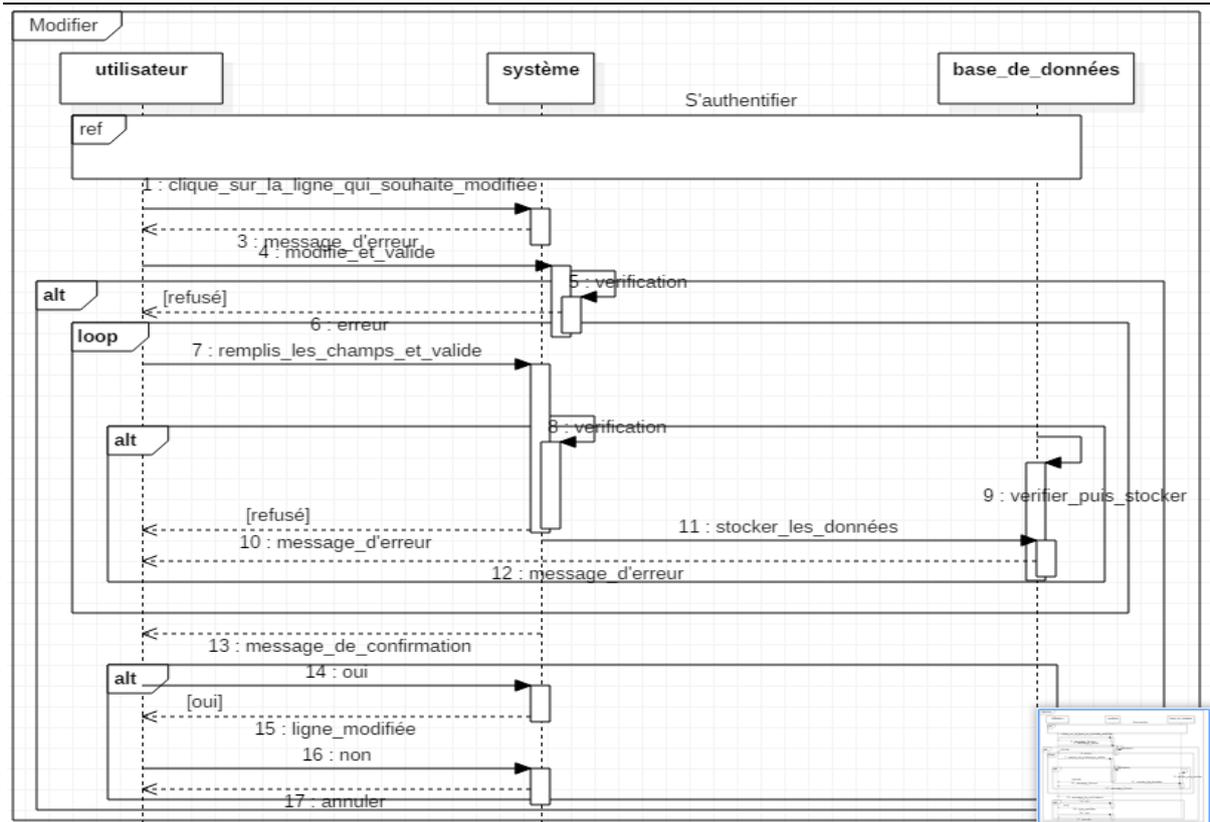


Figure 2. 8 séquence ajouter un enregistrement

- 1- Une fois l’authentification est faite celui-ci sera dirigé vers cette interface ;
- 2- Le formulaire d’ajout lui sera présenté ;
- 3- Il remplit les champs et valide ;
- 4- Le système vérifie les entrées ;
- 5- S’il y’a erreur ça l’affiche ;
- 6- Sinon, il envoie les données pour le stockage ;
- 7- La base de données vérifie et stock ;
- 8- Renvoi un résultat à l’interface et un message de réussite sera affiché sinon un message d’erreur.

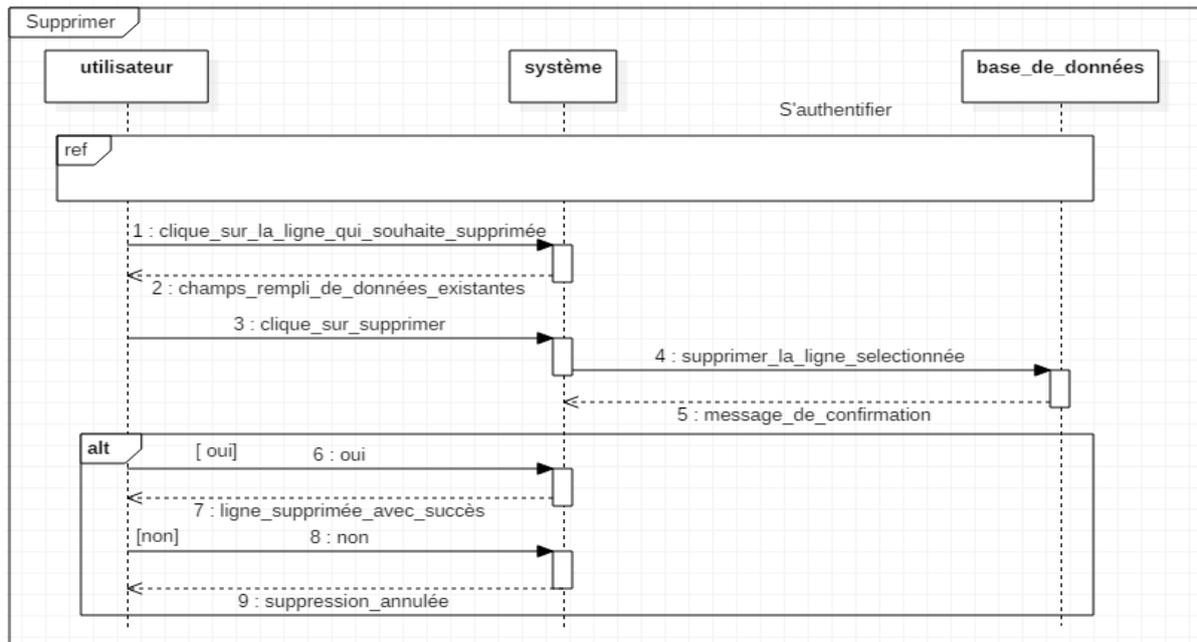
### 2.3.6. Le diagramme de séquence « modifier un enregistrement »



**Figure 2. 9 séquence modifier un enregistrement**

- 1- Une fois l’authentification est faite celui-ci sera dirigé vers cette interface ;
- 2- Il clique sur la ligne qu’il souhaite modifier ;
- 3- Les champs seront remplis des données existantes ;
- 4- L’utilisateur modifie là où il souhaite modifier ;
- 5- Le système vérifie les entrées et un message sera affiché au cas d’erreur, sinon il renvoie les données vers le stockage ;
- 6- Un message de confirmation sera affiché à celui-ci et il choisira oui ou non ;
- 7- Un message de réussite sera affiché.

**2.3.7. Le diagramme de séquence « supprimer un enregistrement »**



**Figure 2. 10 supprimer un enregistrement**

- 1- Une fois l’authentification est faite celui-ci sera dirigé vers cette interface ;
- 2- Il clique sur la ligne qu’il souhaite supprimer ;
- 3- Les champs seront remplis de données existantes ;
- 4- L’utilisateur clique sur supprimer ;
- 5- Le système envoie un message à la base de données en soi-disant supprime les données qui l’identifiant tel ;
- 6- Un message de confirmation sera affiché à celui-ci et il choisira oui ou non ;
- 7- Un message de réussite sera affiché.

**2.3.8. Le diagramme de séquence « password oublié »**

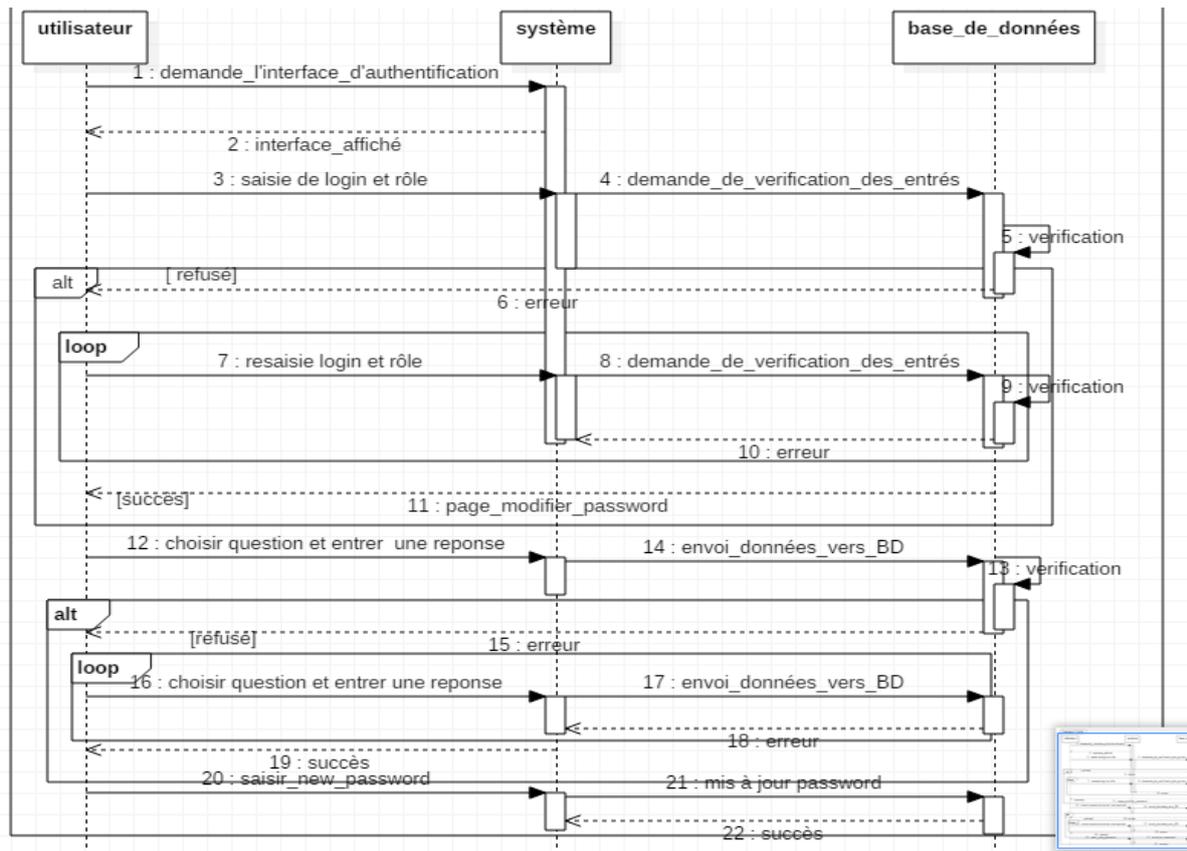
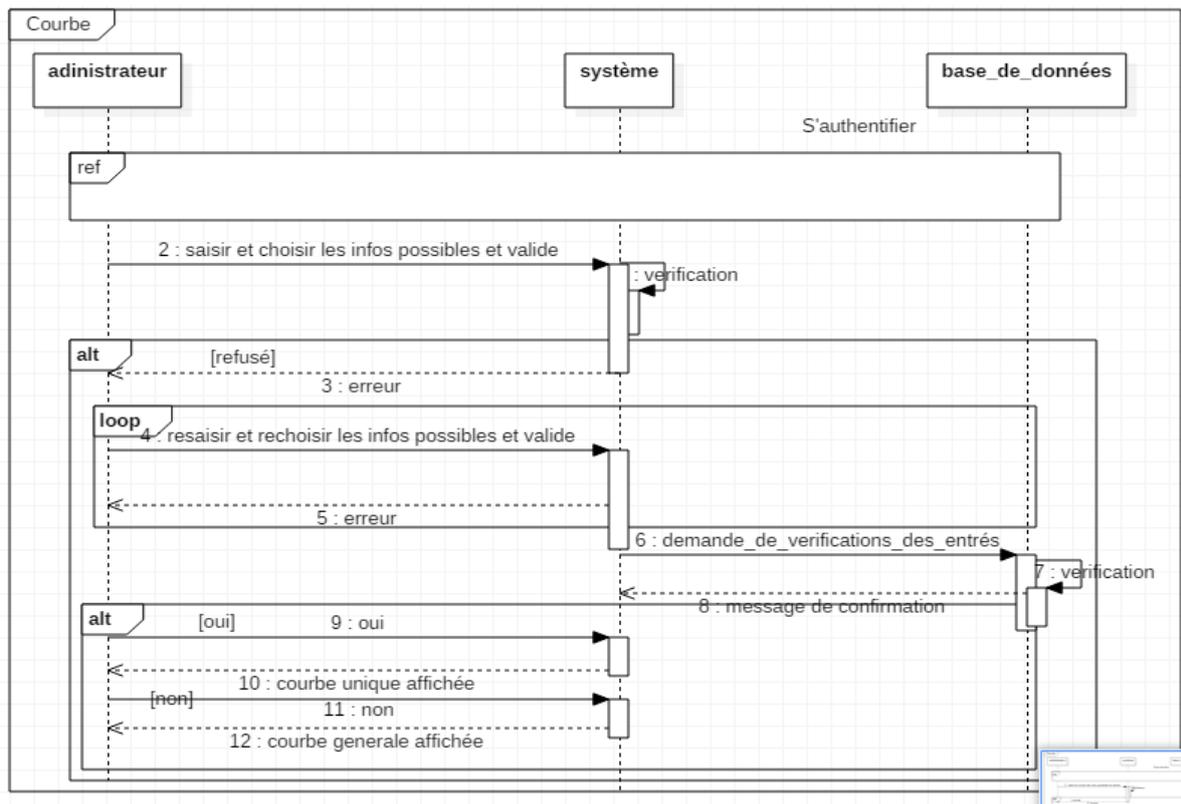


Figure 2. 11 password oublié

- 1- L'utilisateur lance l'application et l'interface login lui sera affichée ;
- 2- Il saisit son login et choisit un rôle ;
- 3- Les données seront envoyées vers la BD pour la vérification ;
- 4- Si les données sont correctes l'interface password oublié sera affichée ;
- 5- Sinon un message d'erreur s'affichera et celui-ci répétera l'action jusqu'à ce qu'il trouve le login et rôle soient corrects ;
- 6- Une fois l'interface password oublié affichée, ce dernier choisit une question, donne une réponse et entre son nouveau password ;
- 7- Les données seront envoyées vers la BD pour la vérification ;
- 8- Si les données sont correctes un message de succès sera affiché ;
- 9- Sinon un message d'erreur et celui-ci répétera l'action jusqu'à ce qu'il trouve la question et la réponse correctes ;

### 2.3.9. Le diagramme de séquence « voir courbe »



**Figure 2. 12 diagramme afficher courbe**

- 1- Une fois l'authentification est faite celui-ci sera dirigé vers cette interface ;
- 2- Il choisit une opération, une source, type de courbe, la courbe par, mois s'il a choisi la courbe par mois ;
- 3- Il entre une année ;
- 4- Si tout cela est vérifié un message de confirmation lui sera présenté ;
- 5- S'il a choisi « oui » une courbe unique sera affichée
- 6- c'est-à-dire d'une seule opération ;
- 7- Sinon une courbe générale sera affichée c'est-à-dire (recette, dépense, bilan).

### 2.3.10. Le diagramme de séquence « déconnecter »

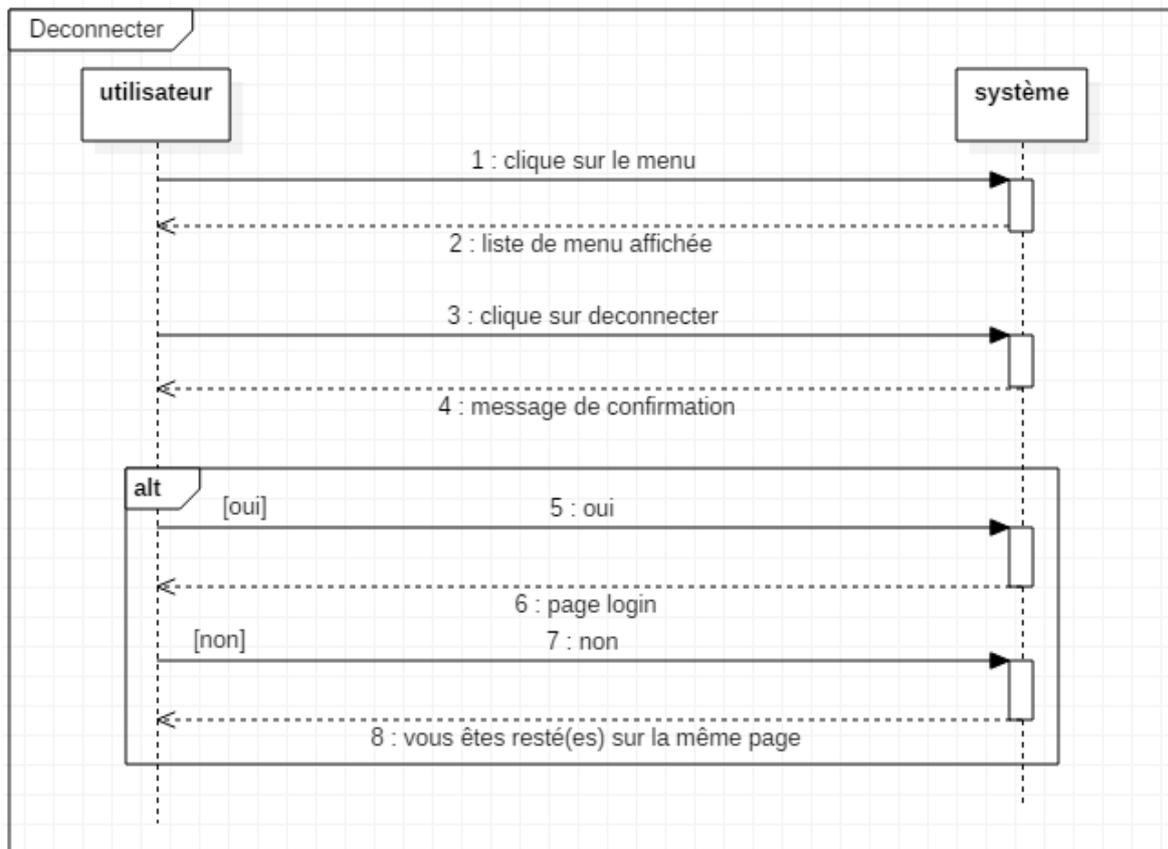


Figure 2. 13 séquence déconnectée

#### 2.4. Le diagramme de classe

Le diagramme de classe est le plus important dans la modélisation orientée objet. Il représente un ensemble de classes, d'interfaces et de collaboration ainsi que leurs relations. Ce diagramme est utilisé pour modéliser la vue statique de conception. Dans notre cas on va représenter le diagramme de classes suivant :

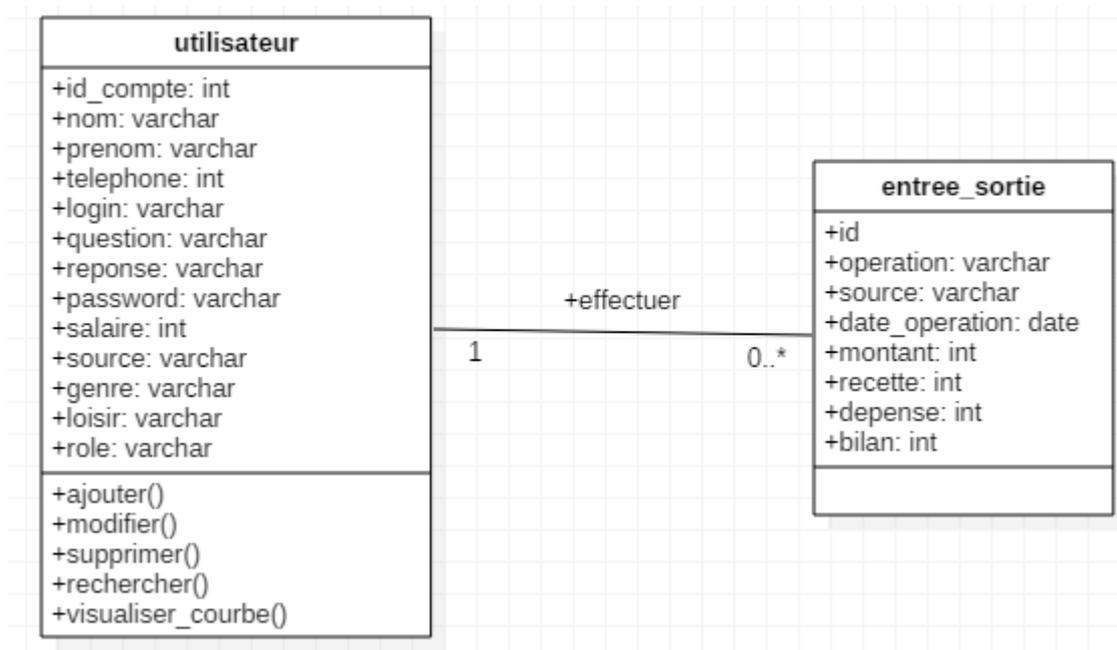


Figure 2. 14 diagramme de classe

## 2.5. Le modèle MVC

Le MVC est un modèle architectural qui sépare la logique de présentation et la mise en forme de données dans une application. Il est basé sur le principe selon lequel les données, le modèle et le traitement sont indépendants, ce qui permet une plus grande flexibilité et une meilleure organisation. Il est expliqué comme suit :

### ✚ Modèle

Il représente les données à afficher et peut être constitué de classes et d'objets d'une base de données.

### ✚ Vue

Elle est une interface graphique qui affiche les données fournies par le modèle et gère les interactions entre l'utilisateur et celle-ci.

### ✚ Contrôleur

Le contrôleur se définit comme couche intermédiaire entre le modèle et la vue, il reçoit et traite les requêtes faites par l'utilisateur et modifie le modèle en fonction des actions de ce dernier.

## **2.6. Conclusion**

Dans ce chapitre, nous nous sommes concentrés sur les aspects analytiques et conceptuels de notre application ainsi que les bases qui interagissent avec elle.

Pour la phase analyse, nous avons défini les différents cas d'utilisation, nous les avons traduits à travers la construction des diagrammes de séquences et de classes, dans la phase conception. Le chapitre suivant sera consacré à la réalisation de notre système.

**CHAPITRE 3 : RÉALISATION**

### 3.1. Introduction

Ce chapitre constitue l'étape finale de notre travail. Son objectif principal est de nous présenter les différentes étapes et logiciels utilisés pour la réalisation de ce dernier.

### 3.2. L'environnement matériel

Pour réaliser ce modeste travail on a utilisé une machine (ordinateur) ayant de caractéristiques ci-dessous :

- ✚ Marque: HP core i3 10egeneration;
- ✚ RAM : 8Go ;
- ✚ Processeur : Intel(R) Core (TM) i3-1005G1 CPU @ 1.20GHz 1.19 GHz;
- ✚ Système d'exploitation : Système d'exploitation 64 bits, processeur x64.

### 3.3. Outils de développement

- ✚ StarUML



*Figure 3. 1 logo UML*

StarUML est un outil (logiciel) développé par MKLabs qui est une société coréenne et conçu spécialement pour la conception et la modélisation UML (Unified Modeling Language).

- ✚ Python



*Figure 3. 2 logo python*

Il est un langage de programmation open source, portable, orienté objet, dynamique, extensible, support l'intégration d'autres langages développés en 1989 par Guido Van Rossum. [2]

- ✚ Matplotlib

Est une bibliothèque développée sous python qui sert à la réalisation des graphes, des courbes etc.

### ✚ Mysql.connector

C'est une bibliothèque écrite sous le langage python qui aide les développeurs à connecter leur code source avec un serveur local quelconque.

### ✚ Tkcalendar

Le tkcalendar est une bibliothèque développée sous python qui gère les dates c'est-à-dire le calendrier.

### ✚ Auto-py-to-exe

C'est un ensemble des bibliothèques écrites en python qui facilite aux développeurs à transformer leur code source en une application (executable).

### ✚ Visual Studio Code(VSC)



*Figure 3. 3 logo VSC*

VSC est un éditeur de texte avancé qui permet aux développeurs de travailler de manière efficace et personnalisée. [6]

### ✚ MySQL



*Figure 3. 4 logo MySQL*

MySQL est un Système de Gestion de Base de Données Relationnelles(SGBDR) open source. Il utilise le SQL comme langage pour qu'un développeur ou administrateur interroge la base de données pour faire ses requêtes.

### ✚ PhpMyAdmin (PMA)



*Figure 3. 5 logo PhpMyAdmin*

Le PMA est un logiciel gratuit et open source développée en PHP, son objectif principal est de :

- ✓ Fournir une interface web pour la gestion de Base de Données (BD) MySQL et Maria DB ; [7]

Faciliter les différentes opérations de SGBD.

- ✚ WAMPSEVER



*Figure 3. 6 wampserver*

Le WAMPSEVER est un logiciel gratuit qui aide aux développeurs de créer un environnement web sur le système d'exploitation Windows. Il est composé de :

- ✚ PhpMyAdmin ;
- ✚ Apache ;
- ✚ MySQL ;
- ✚ Maria DB.

### 3.4. Présentation de l'application

Une fois l'analyse terminée, la conception et la modélisation des fonctionnalités de l'application, nous avons procédé au développement (codage) des interfaces de notre logiciel. Dans cette partie, nous souhaitons vous présenter des interfaces graphiques de notre application.

- ✚ L'interface de VSC

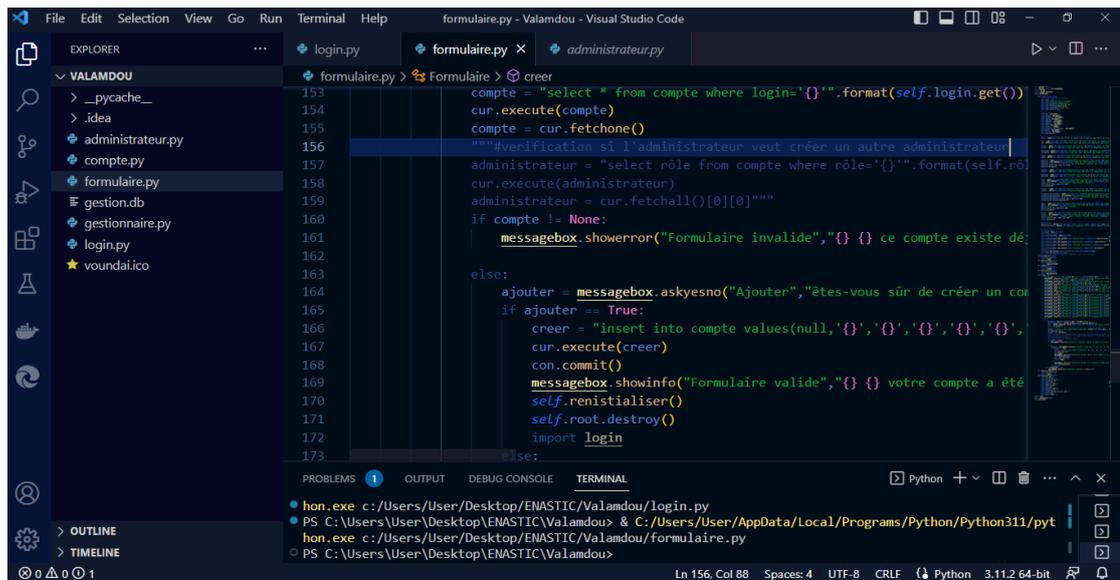


Figure 3. 7 environnement VSC

Les utilisateurs de notre logiciel doivent se connecter à partir de cette interface :

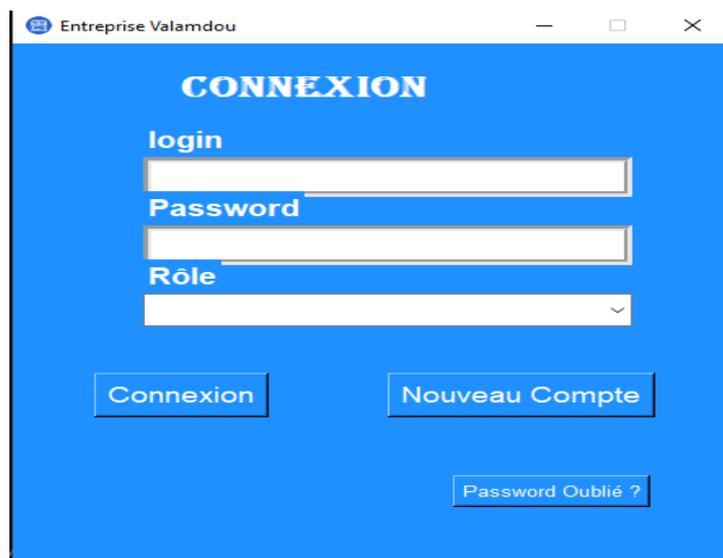


Figure 3. 8 interface login

L'interface créer un compte

Entreprise Valamdou  
cliquez-ici

### CRÉER UN COMPTE

Nom

Téléphone

Selectionnez une question

Password

Salaire

Genre

Prénom

Login

Réponse

Rôle

Source

Loisir

**Créer** **Connexion**

Figure 3. 9 interface créer compte

✚ L'interface password\_oublié

Entreprise Valamdou

### CONNEXION

login

Password

Rôle

**Connexion** **Nouveau Compte**

Password Oublié ?

Entreprise Valamdou

### MOT DE PASSE OUBLIÉ

Selectionnez une question

Répondre

Nouveau Password

**Modifier**

Python

Figure 3. 10 interface password\_oublié

✚ L'interface afficher, modifier et supprimer un utilisateur

## CHAPITRE 2 : RÉALISATION

Id	Nom	Prénom	Contact	Login	Question	Réponse	Password	Rôle	Salaire	Situation	Genre	Loisir
1	ateib	abdoulaye	90446656	ateibabdo	film préfé	stalk	lindefini	administr	200000	célibataire	masculin	musique
2	abdelhadi	yacoub	60676015	abdelhadi	artiste pré	manal-alb	marabout	employé	100000	mariée	masculin	lecture

Figure 3. 11 interface modifier\_supprimer\_user

### L'interface de l'administrateur

Numéro	Opération	Source	Date	Montant	Recette	Dépense	Bilan	Employé
1	recette	ab	2023-01-01	1500	1500	0	1500	ateib
2	recette	ab	2023-01-01	500	500	0	500	abdelhadi
3	dépense	ab	2023-01-01	250	0	250	-250	abdelhadi
4	recette	goudji	2023-08-04	5000	5000	0	5000	abdelhadi

Figure 3. 12 interface administrateur

### ✚ Pour la fonction ajouter :

- ✓ L'administrateur remplira tous les champs et clique sur le bouton 'ajouter' et un message de confirmation sera affiché 'êtes-vous sûr d'ajouter ce nouveau enregistrement ?';
- ✓ Il choisira une option 'oui' ou 'non' ;
- ✓ S'il a choisi 'oui' un message s'affiche 'vous venez un nouvel enregistrement' ;
- ✓ S'il a choisi 'non' un message s'affiche 'L'ajout est annulé avec succès' ;
- ✓ Le système vérifie si l'employé travaille vraiment dans la source entrée ;

### ✚ Pour la fonction modifier :

- ✓ L'administrateur cliquera sur la ligne qui veut modifier, les champs seront remplis de données existantes donc ce dernier modifiera le champ qu'il souhaite modifier et clique sur le bouton 'modifier' ;
- ✓ Une fois cliquer sur le bouton 'modifier' un message de confirmation s'affiche 'êtes-vous sûr de modifier la ligne ?' ;
- ✓ Il choisira une option 'oui' ou 'non' ;
- ✓ S'il a choisi 'oui' un message s'affiche 'vous venez de mettre à jour la ligne' ;
- ✓ S'il a choisi 'non' un message s'affiche 'La modification de la ligne est annulée avec succès'.

### ✚ Pour la fonction supprimer :

- ✓ L'administrateur cliquera sur la ligne qui veut modifier, les champs seront remplis de données existantes donc cette dernière clique sur le bouton 'supprimer' ;
- ✓ Une fois cliquer sur le bouton 'supprimer' un message de confirmation s'affiche 'êtes-vous sûr de supprimer la ligne ?' ;
- ✓ Il choisira une option 'oui' ou 'non' ;
- ✓ S'il a choisi 'oui' un message s'affiche 'vous venez de supprimer la ligne' ;
- ✓ S'il a choisi 'non' un message s'affiche 'La suppression de la ligne est annulée avec succès'.

### ✚ Pour la fonction courbe :

- ✓ L'administrateur choisira l'opération, la source, type courbe, courbe par ;
- ✓ Il entrera l'année ;
- ✓ S'il a choisi courbe par 'mois' donc il doit choisir le mois aussi par contre il n'aura pas besoin du mois même s'il la choisie le système lui enverra un message d'erreur 'vous n'avez pas besoin de mois' ;

- ✓ Si tous les champs sont choisis et/ou remplis selon les conditions demandés un message de confirmation s'affiche 'voulez-vous afficher la courbe de l'opération choisie de source choisie seulement' ;
- ✓ Il choisira une option 'oui' ou 'non' ;
- ✓ S'il a choisi 'oui' la courbe s'affiche comme suit :

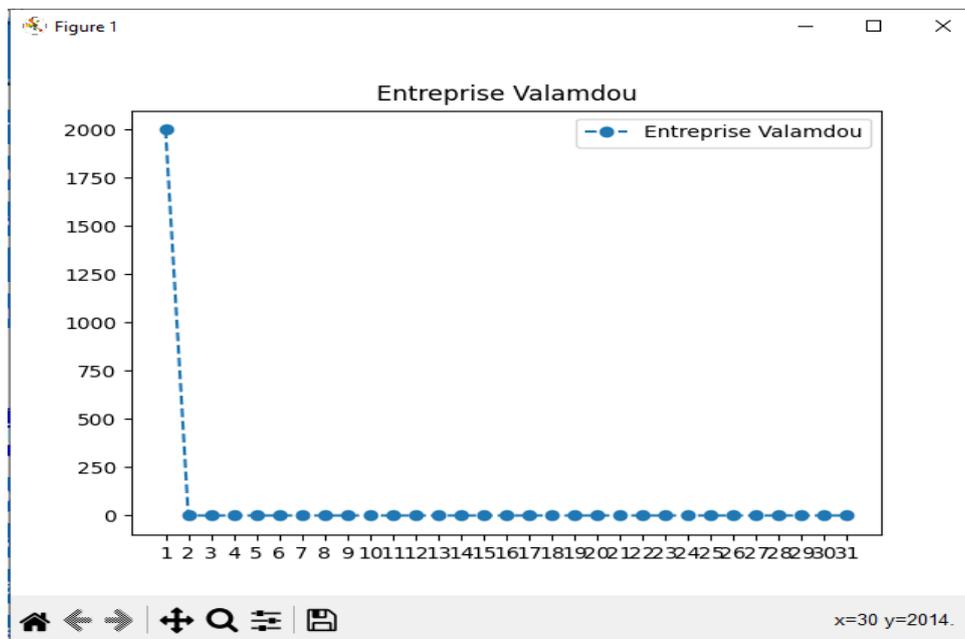


Figure 3.13 : *histogramme\_unique*

- ✓ S'il a choisi 'non' elle s'affichera comme suit :

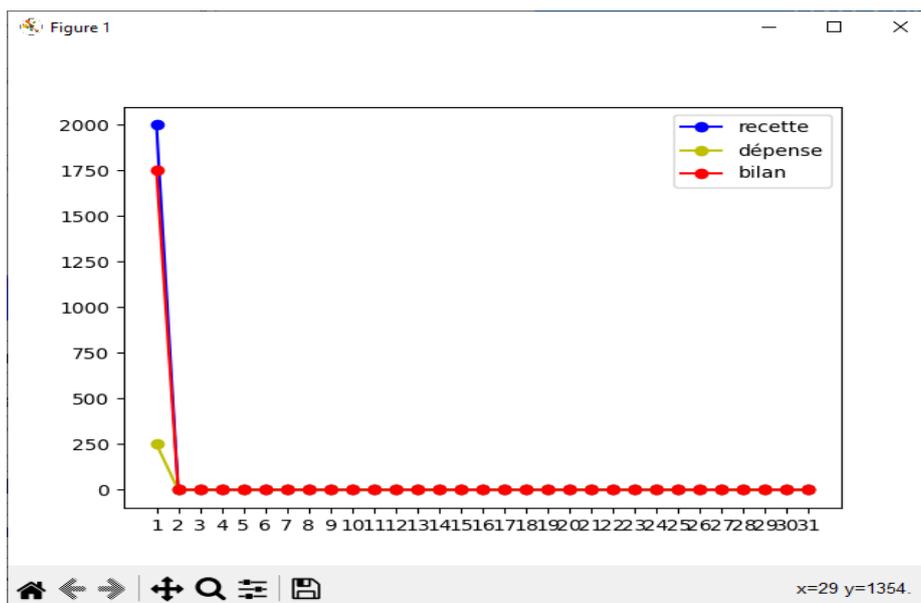


Figure 3.14 *histogramme\_générale*

- ✓ S'il a choisi le type courbe en pourcentage et 'oui' celle-ci s'affiche comme ça :

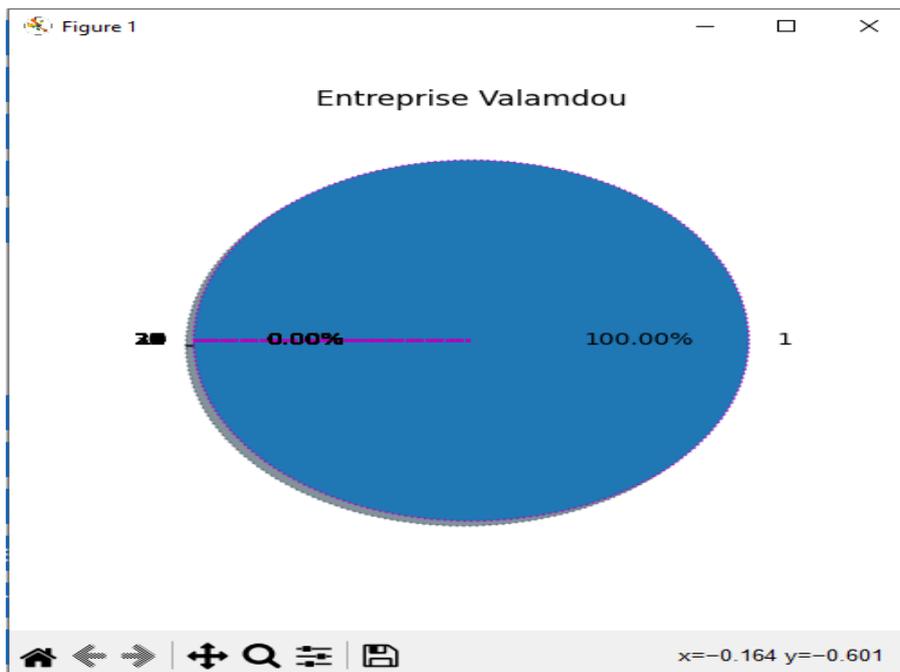


Figure 3. 15 pourcentage unique

- S'il a choisi le type courbe en pourcentage et 'non' celle-ci s'affiche comme ça :

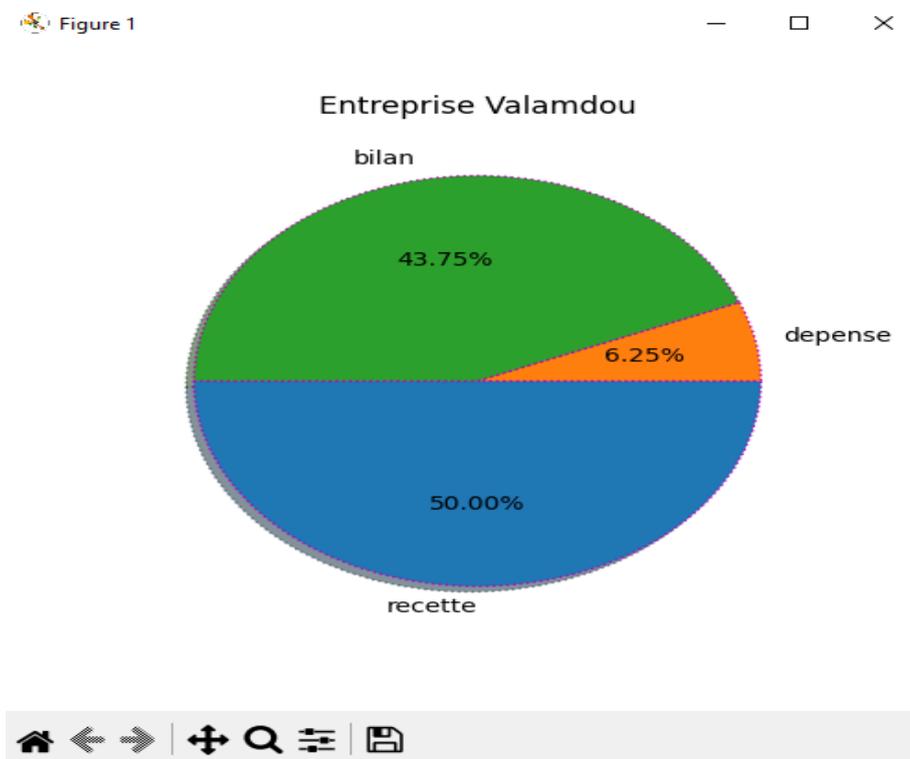


Figure 3. 16 pourcentage générale

✚ Pour la fonction recherche :

- ✓ L'administrateur choisira 'recherche par' et entrera ce qu'il veut rechercher dans le label et valide ;
- ✓ Si l'information demandée existe, elle sera affichée sinon un message d'erreur s'affiche 'l'information demandée n'existe pas vos données enregistrées' ;
- ✓ S'il a choisi recherche par sans saisir quelque chose dans le label un message d'erreur s'affiche 'Veuillez entrer une valeur' ;
- ✓ S'il a entré quelque chose dans le label sans choisir recherche par un message d'erreur s'affiche 'Veuillez choisir une option'.

✚ Pour la fonction afficher :

- ✓ L'administrateur clique sur 'afficher' et toutes les informations existantes seront affichées.

✚ L'interface du gestionnaire :

Numéro	Opération	Source	Date	Montant	Recette	Dépense	Bilan	Employé
1	recette	ab	2023-01-01	1500	1500	0	1500	ateib
2	recette	ab	2023-01-01	500	500	0	500	abdelhadi
3	dépense	ab	2023-01-01	250	0	250	-250	abdelhadi
4	recette	goudji	2023-08-04	5000	5000	0	5000	abdelhadi

*Figure 3. 17 interface gestionnaire*

✚ Pour la fonction ajouter :

- ✓ Le gestionnaire remplira tous les champs et clique sur le bouton 'ajouter' et un message de confirmation sera affiché 'êtes-vous sûr d'ajouter ce nouveau enregistrement ?';
- ✓ Il choisira une option 'oui' ou 'non' ;

- ✓ S'il a choisi 'oui' un message s'affiche 'vous venez d'effectuer un nouvel enregistrement' ;
- ✓ S'il a choisi 'non' un message s'affiche 'L'ajout est annulé avec succès'.
- ✚ Pour la fonction modifier :
  - ✓ Le gestionnaire cliquera sur la ligne qui veut modifier, les champs seront remplis de données existantes donc ce dernier modifiera le champ qu'il souhaite modifier et clique sur le bouton 'modifier' ;
  - ✓ Une fois cliquer sur le bouton 'modifier' un message de confirmation s'affiche 'êtes-vous sûr de modifier la ligne ?' ;
  - ✓ Il choisira une option 'oui' ou 'non' ;
  - ✓ S'il a choisi 'oui' un message s'affiche 'vous venez de mettre à jour la ligne' ;
  - ✓ S'il a choisi 'non' un message s'affiche 'La modification de la ligne est annulée avec succès'.
- ✚ Pour la fonction supprimer :
  - ✓ Le gestionnaire cliquera sur la ligne qui veut modifier, les champs seront remplis de données existantes donc cette dernière clique sur le bouton 'supprimer' ;
  - ✓ Une fois cliquer sur le bouton 'supprimer' un message de confirmation s'affiche 'êtes-vous sûr de supprimer la ligne ?' ;
  - ✓ Il choisira une option 'oui' ou 'non' ;
  - ✓ S'il a choisi 'oui' un message s'affiche 'vous venez de supprimer la ligne' ;
  - ✓ S'il a choisi 'non' un message s'affiche 'La suppression de la ligne est annulée avec succès'.
- ✚ Pour la fonction recherche :
  - ✓ Le gestionnaire choisira 'recherche par' et entrera ce qu'il veut rechercher dans le label et valide ;
  - ✓ Si l'information demandée existe, elle sera affichée sinon un message d'erreur s'affiche 'l'information demandée n'existe pas vos données enregistrées' ;
  - ✓ S'il a choisi recherche par sans saisir quelque chose dans le label un message d'erreur s'affiche 'Veuillez entrer une valeur' ;
  - ✓ S'il a entré quelque chose dans le label sans choisir recherche par un message d'erreur s'affiche 'Veuillez choisir une option'.
- ✚ Pour la fonction afficher :
  - ✓ Le gestionnaire clique sur 'afficher' et toutes les informations existantes seront affichées.

### 3.5. Conclusion

Dans ce chapitre nous avons passé en revue le langage de programmation ainsi que les différents outils utilisés pour la réalisation de notre système en précisant les raisons qui nous ont poussés à entreprendre la réalisation de notre système grâce à ces outils. On a ensuite décrit les différentes fenêtres de notre application ainsi que le fonctionnement général de notre application. De cette description nous avons pu nous rendre compte de la bonne ergonomie qui caractérise les différentes interfaces de notre système, une ergonomie qui permet une utilisation simple et confortable de l'application.

### Conclusion générale

Au cours de ce travail, nous avons entrepris la conception et la réalisation d'une application pour la **gestion d'entrée-sortie** dédiée à l'**entreprise Valamdou** à farcha. Nous avons commencé par la présentation de l'entreprise et analyse des besoins afin de proposer une solution adaptée. Pour ce fait, nous avons utilisé le langage de modélisation UML et le processus de développement en V qui nous ont permis d'analyser en détail les besoins et de concevoir l'application en utilisant les différents diagrammes UML. Ensuite, nous avons procédé à la réalisation de l'application en utilisant :

- ✚ Le python comme langage de programmation ;
- ✚ PhpMyAdmin pour la location de nos bases de données ;
- ✚ MySQL comme système un SGBD ;
- ✚ SQL comme langage d'interrogation de la BD.

Le travail réalisé durant ce projet de fin d'études, nous a permis de combler un certain nombre de lacunes dont les plus importantes sont les insuffisances théoriques et pratiques concernant nos connaissances en conception et en programmation.

La **gestion d'entrée-sortie** est une tâche très importante pour mener un travail, et la mise en place d'un logiciel qui répond à ces différentes tâches et procédures va permettre d'accomplir d'une façon automatisée la **gestion d'entrée-sortie**, ce qui allège le travail de l'entreprise Valamdou.

Pour finir, ce mémoire nous a permis de réaliser une application de gestion d'entrée-sortie répondant aux besoins spécifiques (caisse) de l'**Entreprise Valamdou**. Ensuite, il nous a aidés à mettre en œuvre nos compétences et connaissances théoriques acquises durant notre formation professionnelle.

### **Bibliographie**

- [1] ENASTIC disponible sur : <https://www.enastic.td/> consulté le 10 septembre 2023
- [2] Python disponible sur : <https://docs.python.org/> consulté le 25 août 2023
- [3] document Cours d'UML semestre 3 consulté le 15 septembre 2023
- [4] Cycle de vie en génie logiciel semestre 3 consulté le 18 septembre 2023
- [5] UML disponible sur : <https://www.futura-sciences.com/tech/definitions/informatique-uml-3979/> consulté le 17 septembre 2023
- [6] VSC disponible sur : <https://www.visualstudiocode.com/> consulté le 20 septembre 2023
- [7] PhpMyAdmin disponible sur : <https://www.websiterating.com/fr/web-hosting/glossary/what-is-phpmyadmin/> consulté le 5 août 2023
- [8] apprenez à programmer en python semestre 1 consulté le 20 juillet 2023
- [9] cours en génie logiciel semestre 3 consulté le 28 juillet 2023

Annexe

id_compte	nom	prenom	telephone	login	question	reponse	password	role	salaire	source
1	ateib	abdoulaye	60446654	lindefini	artiste préféré	dadju	lindefini	administrateur	20000	amdjarass
2	abdelhadi	yacoub	60616070	abdelhadi	sur nom	django	6061	employé	25000	farcha
3	hassan	alkhali	65412031	hassan	lieu de naissance	mongo	1234	employé	50000	diguel

id	operation	source	date_operation	montant	recette	depense	bilan	nom_employé
1	dépense	farcha	2023-01-30	1000	0	1000	-1000	abdelhadi
2	dépense	farcha	2023-01-30	5000	0	5000	-5000	abdelhadi
3	recette	farcha	2023-01-10	8000	8000	0	8000	abdelhadi
4	recette	amdjarass	2023-01-01	7000	7000	0	7000	ateib
5	dépense	farcha	2023-09-27	10000	0	10000	-10000	abdelhadi
6	recette	farcha	2023-10-07	5000	5000	0	5000	abdelhadi

```

C:\Users\User>pip install mysql.connector
Requirement already satisfied: mysql.connector in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (2.2.9)

C:\Users\User>pip install tkcalendar
Requirement already satisfied: tkcalendar in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (1.6.1)
Requirement already satisfied: babel in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from tkcalendar) (2.11.0)
Requirement already satisfied: pytz>=2015.7 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from babel->tkcalendar) (2022.6)

C:\Users\User>pip install matplotlib
Requirement already satisfied: matplotlib in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (3.6.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (1.0.6)
Requirement already satisfied: cycler>=0.10 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (4.38.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (1.4.4)
Requirement already satisfied: numpy>=1.19 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (1.23.5)
Requirement already satisfied: packaging>=20.0 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (22.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (9.3.0)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\user\appdata\local\programs\python\python311\lib\site-packages (from python-dateutil->2.7->matplotlib) (1.16.0)
    
```

```

153     compte = "select * from compte where login='{}'".format(self.login.get())
154     cur.execute(compte)
155     compte = cur.fetchone()
156
157     """#verification si l'administrateur veut créer un autre administrateur
158     administrateur = "select rôle from compte where rôle='{}'".format(self.rôle)
159     cur.execute(administrateur)
160     administrateur = cur.fetchall()[0][0]"""
161     if compte != None:
162         messagebox.showerror("Formulaire invalide", "{} {} ce compte existe déjà".format(self.login.get(), self.rôle))
163     else:
164         ajouter = messagebox.askyesno("Ajouter", "êtes-vous sûr de créer un compte")
165         if ajouter == True:
166             creer = "insert into compte values(null, '{}', '{}', '{}', '{}', '{}', '{}', '{}', '{}', '{}')".format(self.login.get(), self.rôle, self.password.get(), self.question.get(), self.reponse.get(), self.password.get(), self.password.get(), self.password.get(), self.password.get(), self.password.get())
167             cur.execute(creer)
168             con.commit()
169             messagebox.showinfo("Formulaire valide", "{} {} votre compte a été créé".format(self.login.get(), self.rôle))
170             self.reinitialiser()
171             self.root.destroy()
172             import login
173         else:
    
```