# Avant propos

L'industrie pétrolière est un secteur d'une importance capitale à l'échelle mondiale, jouant un rôle essentiel dans l'économie. Cependant, les opérations pétrolières sont complexes et impliquent des investissements considérables à chaque étape, de l'exploration à la production en passant par la logistique. La gestion efficiente des dépenses devient alors un enjeu crucial pour les entreprises opérant dans ce domaine.

Ce mémoire se concentre sur le thème du suivi et du contrôle des dépenses liées aux activités pétrolières, explorant les défis, les solutions et les meilleures pratiques dans ce domaine. À travers une analyse approfondie, nous chercherons à comprendre comment les entreprises pétrolières peuvent optimiser leurs ressources financières, et améliorer leur performance globale grâce à une gestion efficace des dépenses.

Le mémoire est structuré en plusieurs chapitres, chacun abordant des aspects spécifiques du suivi et du contrôle des dépenses dans l'industrie pétrolière. Nous commencerons par une introduction générale à l'industrie pétrolière, mettant en lumière son importance économique et les défis particuliers auxquels elle est confrontée en matière de gestion financière. Et, Nous explorerons également les technologies et les outils disponibles pour le suivi des dépenses.

# Dédicaces

A ALLAH Le Tout Miséricordieux, ton amour, ta miséricorde et Tes grâces à mon endroit m’ont fortifié dans la persévérance et l’ardeur au travail.

A mon Père, SOULEYMANE Illiassou en vous, je vois un père dévoué à sa famille. Ta présence en toute circonstance m’a maintes fois rappelé le sens de la responsabilité.

A ma Mère, RAKIA Gambo en vous, je vois la maman parfaite, toujours prête à se sacrifier pour le bonheur de ses enfants. Merci pour tout.

A mes frères et sœurs pour qui, je le sais, ma réussite est très importante. Que Dieu vous Paye pour tous vos bienfaits.

À toute ma famille particulièrement mon tuteur et sa femme, pour leurs soutiens, leurs conseils partagés.

A tous mes amis.

# Remerciements

Avant toute chose, je tiens à remercier Allah pour cette grâce d'être en vie et en bonne santé, et pour avoir terminé ce travail dans les meilleures conditions et ce malgré toutes les contraintes et les obstacles que j'ai rencontrés.

Je tiens à exprimer mes très sincères remerciements à mon encadreur Alparissou A. Wahidou pour sa disponibilité, ses conseils et ses encouragements qui m'ont permis de réaliser ce travail dans les meilleures conditions.

Je tiens à remercier vivement les membres de jury qui ont accepté d’évaluer mon mémoire. Votre participation à mon jour de soutenance est un grand honneur pour moi.

J'adresse aussi mes reconnaissances à tous les professeurs du département d’Informatique de l’UIN, pour la richesse et la qualité de leurs enseignements et les efforts fournis pour assurer à leurs étudiants une bonne formation.

Je voudrais aussi exprimer ma gratitude envers tous ceux qui m'ont accordé leur soutien, tant par leur gentillesse que par leur dévouement.

Je ne peux nommer ici toutes les personnes qui de près ou de loin m'ont aidé et encouragé mais je les en remercie vivement.

Enfin je tiens à dire combien le soutien quotidien de ma famille a été important tout au long de ces quelques années, je leur dois beaucoup.

# Sigles et abréviation

AFE : authorization for expenditures

Cpp**:** contrat de partage de production

CNPC-NP : China Nationnal Petrolium Compagny Niger

DEFSIP : Direction de l’Économie, de la Fiscalité et du Suivi des Investissements Pétroliers

SG : Secrétariat Général

IGS : Inspection Générale des Services

DG : direction générale

DGH : Direction Générale des Hydrocarbures

DGM : La Direction Générale des Mines

DGE/ER : La Direction Générale de l’Energie et des Energies Renouvelables

UML : Unified Modeling Language

IDE : Integrated Development Environment

SGBD : Système de Gestion de Base de Donnée

SQL : Structured Query Language

JDBC : Java Database Connectivity

JAVA EE ou J2EE : Java Entreprise Edition

# Liste des figures

[Figure 1:organigramme du Ministère du Pétrole 13](#_Toc165919240)

[Figure 2:diagramme de cas d’utilisation général 24](#_Toc165919241)

[Figure 3:diagramme des cas d’utilisation (gestion des utilisateurs) 26](#_Toc165919242)

[Figure 4:diagramme des cas d’utilisation (gestion des départements) 28](#_Toc165919243)

[Figure 5:diagramme des cas d’utilisation (gestion des projets) 30](#_Toc165919244)

[Figure 6:: diagramme des cas d’utilisation (gestion des autorisations de dépense) 32](#_Toc165919245)

[Figure 7:scenario supprimer un utilisateur 33](#_Toc165919246)

[Figure 8:diagramme de séquence (scenario supprimer un utilisateur) 34](#_Toc165919247)

[Figure 9:diagramme d’activité (scenario authentification) 35](#_Toc165919248)

[Figure 10:diagramme d’activité (scenario suppression) 36](#_Toc165919249)

[Figure 11:diagramme de cas d’utilisation (scenario modifier) 37](#_Toc165919250)

[Figure 12:diagramme de séquence (scenario modifier) 38](#_Toc165919251)

[Figure 13:diagramme d’activité (scenario modifier) 39](#_Toc165919252)

[Figure 14:scenario enregistrer 40](#_Toc165919253)

[Figure 15:diagramme de séquence (scenario enregistrer) 41](#_Toc165919254)

[Figure 16:: diagramme d’activité (scenario enregistrer) 42](#_Toc165919255)

[Figure 17:scenario rechercher 43](#_Toc165919256)

[Figure 18:: diagramme de séquence (scenario rechercher) 44](#_Toc165919257)

[Figure 19:diagramme d’activité (scenario rechercher) 45](#_Toc165919258)

[Figure 20:scenario afficher 46](#_Toc165919259)

[Figure 21:diagramme de séquence (scenario afficher) 47](#_Toc165919260)

[Figure 22:diagramme d’activité (scenario afficher) 48](#_Toc165919261)

[Figure 23:scenario imprimer 49](#_Toc165919262)

[Figure 24:diagramme de séquence (scenario imprimer) 50](#_Toc165919263)

[Figure 25:diagramme d’activité (scenario imprimer) 51](#_Toc165919264)

[Figure 26:diagramme de classe 52](#_Toc165919265)

[Figure 27:interface de startUML 57](#_Toc165919266)

[Figure 28:: interface de Balsamiq Wireframes 59](#_Toc165919267)

[Figure 29:interface de netbeans 60](#_Toc165919268)

[Figure 30:interface wampServer 61](#_Toc165919269)

[Figure 31:: interface PhpMyAdmin 62](#_Toc165919270)

[Figure 32:architecture 2 tiers 65](#_Toc165919271)

[Figure 33:cycle en V 66](#_Toc165919272)

[Figure 34:maquette authentification 70](#_Toc165919273)

[Figure 35:maquette du menu 70](#_Toc165919274)

[Figure 36:interface authentification 71](#_Toc165919275)

[Figure 37:interface utilisateur 72](#_Toc165919276)

[Figure 38:interface du menu 72](#_Toc165919277)

[Figure 39:code utilisateur 76](#_Toc165919278)

[Figure 40:code login 76](#_Toc165919279)

# Liste des tableaux

[Tableau 1:Présentation des acteurs du système 23](#_Toc165920071)

[Tableau 2:Description des classes 53](#_Toc165920072)

[Tableau 3:Description des attributs 54](#_Toc165920073)

[Tableau 4:Description des méthodes 55](#_Toc165920074)

[Tableau 5:Organisation 67](#_Toc165920075)

[Tableau 6:test 69](#_Toc165920076)

# Sommaire

[Avant propos 1](#_Toc165933696)

[Dédicaces 2](#_Toc165933697)

[Remerciements 3](#_Toc165933698)

[Sigles et abréviation 4](#_Toc165933699)

[Liste des figures 5](#_Toc165933700)

[Liste des tableaux 7](#_Toc165933701)

[Sommaire 8](#_Toc165933702)

[Introduction générale 11](#_Toc165933703)

[Chapitre I : Présentation de la structure d’accueil 13](#_Toc165933704)

[1.Introduction 13](#_Toc165933705)

[2. Présentation du centre d'accueil 13](#_Toc165933706)

[2.1 Présentation historique 13](#_Toc165933707)

[2.2 Missions 13](#_Toc165933708)

[2.3. Organisation de l’administration générale 13](#_Toc165933709)

[Du Cabinet du Ministre (CAB/MP/M/E) 13](#_Toc165933710)

[Du Secrétariat Général (SG) 14](#_Toc165933711)

[De l’Inspection Générale des Services (IGS) 14](#_Toc165933712)

[Des Directions Générales (DG) 14](#_Toc165933713)

[Des Directions Techniques Nationales et les Directions Nationales Transversales ou d’Appui 14](#_Toc165933714)

[2.4. Présentation de la Direction de l’Economie, de la Fiscalité et du Suivi des Investissements Pétroliers (DEFSIP) ; 15](#_Toc165933715)

[3. Présentation du thème 16](#_Toc165933716)

[3.1 Présentation 16](#_Toc165933717)

[3.2. Travail attendu 17](#_Toc165933718)

[4. Étude de l'existant 17](#_Toc165933719)

[4.1. Système actuel 17](#_Toc165933720)

[4.2 critiques 18](#_Toc165933721)

[4.3 Forces 19](#_Toc165933722)

[4.4 Solutions proposées 19](#_Toc165933723)

[4.4.1. Première solution 19](#_Toc165933724)

[4.4.2. Deuxième solution 19](#_Toc165933725)

[4.5 Solution retenue 19](#_Toc165933726)

[5. Conclusion 20](#_Toc165933727)

[Chapitre II : Analyse et spécification 21](#_Toc165933728)

[1. Introduction 21](#_Toc165933729)

[2. Analyse des besoins 21](#_Toc165933730)

[3. Diagramme des cas d'utilisation général 23](#_Toc165933731)

[4.Conclusion 31](#_Toc165933732)

[Chapitre III : Étude conceptuelle 32](#_Toc165933733)

[1.Introduction 32](#_Toc165933734)

[2. Description des cas d'utilisation 32](#_Toc165933735)

[2.1. Cas d’utilisation gestion des utilisateurs (scenario supprimer un utilisateur) 32](#_Toc165933736)

[2.2. Cas d’utilisation gestion des départements (scenario modifier un département) 36](#_Toc165933737)

[2.3. Cas d’utilisation gestion des utilisateurs (scenario enregistrer une AFE) 39](#_Toc165933738)

[2.4. Cas d’utilisation gestion des utilisateurs (scenario rechercher un utilisateur) 42](#_Toc165933739)

[2.5. Cas d’utilisation gestion des utilisateurs (scenario afficher un utilisateur) 45](#_Toc165933740)

[2.6. Cas d’utilisation gestion des utilisateurs (scenario imprimer la liste des projets) 48](#_Toc165933741)

[3.Diagramme de classe 50](#_Toc165933742)

[4.Description détaillée des classes, attributs et des méthodes 51](#_Toc165933743)

[4.1. Description des classes 51](#_Toc165933744)

[4.2. Description des attributs 52](#_Toc165933745)

[4.3. Description des méthodes 52](#_Toc165933746)

[5. Conclusion 53](#_Toc165933747)

[Chapitre IV : Réalisation 54](#_Toc165933748)

[1.Introduction 54](#_Toc165933749)

[2. Environnement de travail 54](#_Toc165933750)

[2.1 Outils matériel 54](#_Toc165933751)

[2.2. Outils logiciels 54](#_Toc165933752)

[2.2.1. SE 54](#_Toc165933753)

[2.2.2. Outils de modélisation UML 54](#_Toc165933754)

[2.2.3. Outils maquettage 55](#_Toc165933755)

[2.2.3.1. Balsamiq Wireframes 56](#_Toc165933756)

[2.2.4. Environnement de développement intégré 57](#_Toc165933757)

[2.2.5. WampServer 57](#_Toc165933758)

[2.2.6. PhpMyAdmin 58](#_Toc165933759)

[2.2.7. Langages de programmation 59](#_Toc165933760)

[2.2.7.1. Langage java 59](#_Toc165933761)

[2.2.7.2. Langage SQL 60](#_Toc165933762)

[2.2.8. JDK (Java Development Kit) 60](#_Toc165933763)

[3.L’architecture de l'application 61](#_Toc165933764)

[3.1 Architecture 3-tiers 61](#_Toc165933765)

[4.Développement 62](#_Toc165933766)

[4.1 Méthodologie et l’organisation de travail 62](#_Toc165933767)

[4.1.1 Méthodologie (Cycle en V) 62](#_Toc165933768)

[4.1.2 Organisation 63](#_Toc165933769)

[4.2 La plateforme Java SE 64](#_Toc165933770)

[5. Phase de test et de validation 64](#_Toc165933771)

[6. Les interfaces graphiques 65](#_Toc165933772)

[6.1. Les maquettes 65](#_Toc165933773)

[6.1.1. Présentation de l’interface authentification 66](#_Toc165933774)

[6.1.2. Présentation de l’interface menu 66](#_Toc165933775)

[6.2. L’application 66](#_Toc165933776)

[6.2.1. Présentation de l’interface authentification 67](#_Toc165933777)

[6.2.2. Présentation de l’interface ajout utilisateur 68](#_Toc165933778)

[6.2.3. Présentation de l’interface menu général 68](#_Toc165933779)

[7. Conclusion 69](#_Toc165933780)

[Conclusion générale 70](#_Toc165933781)

[Bibliographie 71](#_Toc165933782)

[Webographie 71](#_Toc165933783)

[Annexes 72](#_Toc165933784)

# Introduction générale

L'industrie pétrolière occupe une place prépondérante dans l'économie mondiale, fournissant une source essentielle d'énergie pour les activités humaines et alimentant le développement industriel et technologique à travers le globe.

Cependant, cette industrie est confrontée à plusieurs défis complexes, au cœur de ces défis se trouve la question cruciale de la gestion financière, en particulier le suivi et le contrôle des dépenses liées aux activités pétrolières. Les entreprises opérant dans ce secteur doivent jongler avec des budgets importants, des coûts opérationnels élevés et une diversité d'activités allant de l'exploration et la production aux opérations logistiques complexes.

C’est dans ce sens qu’en 2008, le Niger a signé un contrat de partage de production (CPP) avec la société CNPC-NP (China Nationnal Petrolium Compagny Niger) sur le bloc Agadem. Le contrat qui a connu des avenants court jusqu’à 2045.

Le CPP est l’une des formes de contrats pétroliers (les contrats de concession, les contrats de services et les contrats de production). Dans le CPP, c’est l’opérateur qui investit pour la recherche, l’exploration et l’exploitation.

En cas de découverte, l’opérateur récupère le montant qu’il a investi jusqu’à hauteur de 70% (ce taux est dégressif en long terme). Ce montant est appelé cout récupérable.

L’objet de notre travail est de mettre à la disposition de la Direction de l’Économie, de la Fiscalité et du Suivi des Investissements Pétroliers (DEFSIP), un outil informatique lui permettant de suivre annuellement les dépenses qui deviendraient des coûts récupérables.

En effet, Le suivi et contrôle des dépenses revêt une importance capitale pour plusieurs raisons. Tout d'abord, il permet aux entreprises de maintenir leur rentabilité en contrôlant efficacement les coûts associés à chaque phase de leurs activités, de l'acquisition de concessions pétrolières à la mise en production des champs. Ensuite, il contribue à une gestion plus efficiente des ressources en identifiant les gisements d'économies et en optimisant les processus. Enfin, le suivi et contrôle des dépenses est indispensable pour se conformer aux réglementations comptables et fiscales, assurant ainsi la transparence et la responsabilité financière.

C’est dans ce cadre que se situe ce présent mémoire qui a pour objectif d'explorer en profondeur les enjeux, les solutions et les bonnes pratiques liés au suivi et au contrôle des dépenses dans l'industrie pétrolière. Nous aborderons les outils technologiques disponibles pour faciliter le suivi des dépenses.

Ce mémoire est structuré en quatre (04) chapitres, chacun explorant un aspect spécifique du suivi et du contrôle des dépenses dans l'industrie pétrolière.

Nous commencerons par le premier chapitre qui décrit la structure d’accueil et l’analyse de l’existant. En effet, il permet de présenter le Ministère du pétrole et de faire ressortir l’organisation de l’administration générale ainsi que LaDirection de l’Économie, de la Fiscalité et du Suivi des Investissements Pétroliers (DEFSIP) qui est chargée des dépenses.

Le deuxième chapitre décrit l’analyse et la spécification. Il permet d’exposer les problèmes que rencontre les dépenses liées aux activités pétrolières. Ce chapitre permet de fixer les objectifs de ce présent mémoire.

Le troisième chapitre porte sur l’étude conceptuelle. Il aborde l’ensemble des diagrammes de composants, mais aussi la conception détaillée illustrant les diagrammes de classes et du dictionnaire de données nécessaire pour la réalisation du projet.

Le quatrième et dernier chapitre dénommé réalisation illustre le codage des différentes entités, les interfaces, mais aussi les outils de développement utilisés pour l’implémentation et fait une présentation générale de l’application.

# Chapitre I : Présentation de la structure d’accueil

## 1.Introduction

Ce chapitre comprend une présentation du Ministère du pétrole à travers ses attributions et missions, son organisation générale et la Présentation de la Direction de l’Economie, de la Fiscalité et du Suivi des Investissements Pétroliers (DEFSIP) qui est le lieu particulier de notre étude.

## 2. Présentation du centre d'accueil

### 2.1 Présentation historique

Le ministère du pétrole du Niger est le ministère chargé des ressources liées au pétrole au Niger.

A la suite du remaniement ministériel du 23 Avril 2022, le Ministère du pétrole, de l’Energie et des Énergies Renouvelables est scindé en deux départements ministériels : le Ministère de l’Energie et des Énergies Renouvelables et le Ministère du Pétrole.

Ce département ministériel du gouvernement nigérien est chargé de la conception, de la mise en œuvre et du suivi de la politique de gestion des ressources énergétiques pétrolières sur l’ensemble du territoire du Niger.

### 2.2 Missions

Le ministère du pétrole du Niger dont le cabinet est situé au sixième étage de l’Immeuble ONAREM a pour mission la conception, la mise en œuvre et le suivi de la politique de gestion des ressources énergétiques pétrolières sur l’ensemble du territoire du Niger.

### 2.3. Organisation de l’administration générale

L’Administration Centrale du Ministère de Pétrole comprend :

* Le Cabinet du Ministre ;
* Le Secrétariat Général ;
* L’Inspection Générale des Services ;
* Les Directions Techniques Nationales et les Directions Nationales Transversales ou d’Appui ;
* Les Organes Consultatifs ;
* Les Administrations de Mission.

#### Du Cabinet du Ministre (CAB/MP/M/E)

Le Cabinet du Ministre est organisé ainsi qu’il suit et comprend :

* Deux (2) ou trois (3) Conseillers(ères) Techniques ;
* Un Chef de Cabinet ;
* Un Secrétariat Particulier ;
* Un Responsable de la Communication ;
* Un Attaché de Protocole ;
* Un ou deux (2) Agent(s) de Sécurité.

#### Du Secrétariat Général (SG)

Le Secrétariat Général est organisé ainsi qu’il comprend :

* Un Secrétaire Général ;
* Un Secrétaire Général Adjoint ;
* Un Bureau d’Ordre (BO) ;
* Un Secrétariat.

#### De l’Inspection Générale des Services (IGS)

L’Inspection Générale des Services est placée sous l’autorité directe du Ministre. Elle est organisée ainsi qu’il suit et comprend :

* Un Inspecteur Général des Services ;
* Des Inspecteurs des Services ;
* Un Secrétariat.

#### Des Directions Générales (DG)

Les Directions Générales sont les suivants :

* La Direction Générale des Hydrocarbures (DGH) ;
* La Direction Générale des Mines (DGM) ;
* La Direction Générale de l’Energie et des Energies Renouvelables (DGE/ER) ;

#### Des Directions Techniques Nationales et les Directions Nationales Transversales ou d’Appui

Les Directions d’Appui sont :

* Une Direction des Ressources Financières et du Matériel (DRFM) ;
* Une Direction de la Législation (DL) ;
* Une Direction des Statistiques (DS) ;
* Une Direction des Etudes et de la Programmation (DEP) ;
* Une Direction des Ressources Humaines (DRH) ;
* Une Direction des Marchés Publics et des Délégations des Services Publics (DMPDSP) ;
* Une Direction des Archives de l’Information, de la Documentation et des Relations Publiques (DAIDRP) ;

Figure 1:organigramme du Ministère du Pétrole

### 2.4. Présentation de la Direction de l’Economie, de la Fiscalité et du Suivi des Investissements Pétroliers (DEFSIP) ;

LaDirection de l’Economie, de la Fiscalité et du Suivi des Investissements Pétroliers fait partie des directions nationales. Elle est placée sous la coupe de la Direction Générale des Hydrocarbures**.**

Elle a, entre autres, pour attributions :

* Veiller à l’application des dispositions économiques et fiscale dans les contrats pétroliers ;
* Liquider les taxes et les autres obligations financières incombant aux opérateurs du secteurs des hydrocarbures, ainsi que le suivi de leurs exécutions ;
* Suivre l’exécution des contrats de vente du pétrole brut et s’assurer que toutes les procédures requises en matière de chargement ou d’exportation de pétrole brut sont observées ;
* Etudier les demandes des sociétés pour les autorisations de franchises d’importation ou d’exportation des matériels, matériaux, produits et équipements définis par la réglementation en vigueur ;
* Contrôler l’approvisionnement, l’utilisation et le stockage des matériels, matériaux, produits et équipements des titulaires de permis ou d’autorisation et leurs sous-traitants ;
* Elaborer et mettre en œuvre les politiques, stratégies, plan et programmes nationaux dans les domaines de l’économie et de la fiscalité pétrolières ;
* Analyser les contrats directement ou indirectement liés à l’amont et l’aval pétrolier, ainsi que le suivi de leurs exécutions en collaboration avec les autres directions ;
* Veiller à la réalisation des programmes d’investissement des autorisations exclusives de recherche et les autorisations exclusives d’exploitation en collaboration avec les directions concernées ;
* Suivre la participation de l’Etat dans les partenariats, contrôler et analyser la récupération des coûts pétroliers, ainsi que l’évolution du facteurs « R » ;
* Faire des audits ou participer aux audits des comptes et livres des sociétés pétrolières ;
* Elaborer et mettre en œuvre une stratégie et un plan de formation répondant aux besoins réels de la Direction et veillez au respect de la formation continue du personnel nigérien des opérateurs pétroliers ;
* Mettre en œuvres des textes d’application pour concrétiser les droits et obligations contenus dans les codes pétroliers en vigueur, notamment l’élaboration de textes pour le contenu local ;
* Définir et mettre en œuvre les moyens, outils et procédures pour contrôler le prix de ventes des hydrocarbures, les coûts pétroliers ;
* Veiller à la mise en œuvre de la stratégie et du plan d’actions du contenu local dans le secteur pétrolier ;
* Participer aux négociations des conventions, contrats et accords avec les partenaires et suivre leurs exécutions ;
* Participer à l’élaboration des projets de textes relatifs aux hydrocarbures ;
* Contribuer à la promotion du potentiel pétrolier national dans son volet avantages fiscaux ;
* Proposer la mise en place d’un dispositif institutionnel permanent(cellule) chargé de la levée de fonds, de la rédaction des conventions de financement pour tous les projets pétroliers, en collaboration avec les structures concernées et au montage de projets structurants ;
* Participer à l’analyse des études de faisabilité de l’amont et de l’aval pétrolier et aussi assister à l’analyse financière des programmes des travaux de recherche, d’exploitation ou de construction soumis par des investisseurs nationaux ou étrangers ;
* Compiler, les rapports et budgets de la Direction Générale ;
* Impulse, superviser, coordonner, suivre et évaluer l’ensemble des activités des divisions placées sous sa responsabilité.

## 3. Présentation du thème

### 3.1 Présentation

L'industrie pétrolière dans le bloc Agadem au Niger nécessite une gestion financière efficace pour suivre et contrôler les dépenses liées aux différentes activités, telles que l'exploration, la production, la logistique, etc…d’où le besoin d’une application de suivi et contrôle des dépenses liées aux activités pétrolières.

### 3.2. Travail attendu

Le secteur pétrolier, par sa complexité et ses exigences réglementaires, nécessite une approche rigoureuse en matière de gestion financière. Le système proposé fournira une plateforme permettant la gestion des utilisateurs, la gestion des départements, la gestion des projets, la gestion des autorisations de dépense (AFE), et de garantir la confidentialité des données financières, offrant ainsi aux décideurs une vision globale et en temps réel des dépenses engagées et des budgets alloués.

* L’authentification : Garantir la sécurité des données financières sensibles et assurer la confidentialité des informations concernant les dépenses et les budgets.
* La gestion des utilisateurs : Le système devrait être capable d’ajouter des utilisateurs de les enregistrer, de les modifier de les supprimer, de les rechercher, de les afficher et de les imprimer au besoin ;
* La gestion des départements : Le système devrait être capable d’enregistrer, de modifier, de supprimer, d’afficher, de rechercher et d’imprimer tous les départements avec des informations propres au département telles que :
* Le code du département ;
* Le nom du département ;
* Le budget prévu pour ce département ;
* La date ;
* La gestion des projets : dans cette gestion l’enregistrement, la modification, la suppression, la recherche, l’affichage et l’impression des différents projets devront primer ;
* La gestion des autorisations de dépense ;
* L’exécution des autorisations de dépense ;
* Modification de mot de passe ;

## 4. Étude de l'existant

### 4.1. Système actuel

La CNPCP-NP envoie son programme annuel de travaux ainsi que le budget au sous-comité technique qui :

* Envoi le programme annuel de travaux et du budget au sous-comité technique, qui examine et valide la proposition. Cela garantit que les programmes et budgets sont évalués par des experts avant d'être approuvés ;
* Une fois validé par le sous-comité technique, le programme est envoyé au sous-comité d’association (DGH, opérateur) qui examine et valide le budget. Cette étape implique la collaboration entre les parties prenantes internes et externes pour s'assurer que le budget est réaliste et aligné sur les objectifs stratégiques ;
* Le programme est ensuite envoyé au comité de gestion (comprenant l'État et l'opérateur) qui examine et approuve le budget. Cette étape est cruciale car elle implique les autorités de tutelle et garantit la conformité aux réglementations et politiques gouvernementales ;

Le programme comprend plusieurs entités et chaque entité a des rubriques spécifiques avec leur budget alloué, ce qui permet une allocation précise des ressources.

Chaque dépense est liée à une autorisation de dépense (AFE) comprenant un numéro unique, l'objet de la dépense, la rubrique concernée, l'entité concernée, le montant et le type d'AFE (approuvé ou approbation en fonction du montant). Cela assure un suivi détaillé des dépenses et des allocations budgétaires.

Les demandes d'AFE sont envoyées par lot avec une lettre explicative. Les services techniques du ministère du pétrole examinent chaque AFE du lot et fournissent une réponse détaillée concernant l'approbation ou le rejet. Cette approche garantit un contrôle strict des dépenses.

La possibilité de réviser le budget alloué à une dépense permet d'ajuster les allocations en fonction des besoins et des changements de contexte, assurant ainsi une gestion financière agile et réactive.

L'exécution des dépenses se fait en envoyant les factures comportant le numéro de l'AFE au ministère du pétrole. À la fin de l'année, les comités se réunissent pour examiner le budget exécuté, ce qui permet un suivi continu des dépenses et une évaluation de la performance financière.

Dans l'ensemble, ce processus démontre une approche structurée et rigoureuse dans la gestion financière des activités pétrolières, avec des mécanismes de contrôle, de validation et de suivi bien définis à chaque étape du processus budgétaire.

### 4.2 critiques

Le suivi et contrôle occupe une place importante et indispensable au sein du Ministère du pétrole. Malgré cela, ce Ministère rencontre d’énormes problèmes dans sa gestion de contrôle et de suivi dans son mode de fonctionnement manuel. Nous avons essayé dans cette partie d’énumérer les problèmes liés à son fonctionnement manuel :

* Les difficultés d’avoir les dépenses totales par autorisation de dépense à chaque instant ;
* Les difficultés d’avoir les dépenses totales par projet à chaque instant ;
* Les difficultés d’avoir les dépenses totales par département à chaque instant ;
* Le suivi se fait via des tableaux Excel ;
* Les difficultés de retrouver rapidement une autorisation de dépense ;
* Les difficultés de produire des statistiques périodiques ;

### 4.3 Forces

La force du système actuel est la suivante :

* La DEFSIP dispose des cadres compétents (ingénieurs, fiscalistes, économistes pétroliers etc…) ;
* Les cadres vont sur le terrain pour le suivi et le contrôle ;
* Les cadres disposent des ordinateurs individuels ;
* La DEFSIP dispose d’un accès internet via un Wi-Fi.

### 4.4 Solutions proposées

#### 4.4.1. Première solution

Pour optimiser la gestion financière et assurer une vision globale et en temps réel des activités économiques dans le secteur pétrolier du bloc Agadem au Niger, l'implémentation d'un système de gestion financière intégré se révèle indispensable. Cette approche vise à :

* Déployer un système de gestion financière intégré spécifiquement conçu pour le secteur pétrolier, permettant une vue d’ensemble sur toutes les activités.
* Intégrer des fonctionnalités avancées telles que le suivi en temps réel des dépenses, la gestion des budgets et des approbations ;

#### 4.4.2. Deuxième solution

Pour garantir une gestion financière robuste et conforme aux normes réglementaires, une approche essentielle consiste au renforcement des contrôles internes et de la conformité. Cette initiative vise à instaurer des mécanismes de contrôle efficaces et à assurer une conformité rigoureuse aux politiques et réglementations en vigueur dans le secteur pétrolier du bloc Agadem au Niger. Ces mécanismes comprennent :

* Mettre en place des procédures et des contrôles internes robustes pour vérifier et valider les dépenses, les factures, les contrats et autres transactions financières, en conformité avec les politiques et réglementations en vigueur ;
* Effectuer régulièrement des audits financiers pour détecter les écarts, les anomalies et les risques potentiels, et mettre en œuvre des actions correctives appropriées pour assurer l'intégrité des données financières ;
* Collaborer étroitement avec les autorités réglementaires et les partenaires externes pour garantir une conformité totale aux normes du secteur et pour renforcer la confiance des parties prenantes en matière de gestion financière ;

### 4.5 Solution retenue

Après avoir détecté les insuffisances dans la procédure actuelle du suivi et contrôle des dépenses au sein du ministère du pétrole, notre solution consiste à concevoir et implémenter un système de suivi et de contrôle des dépenses liées aux activités pétrolières qui comblera les défaillances notées. Les objectifs spécifiques de ce travail sont :

* La gestion des utilisateurs ;
* La gestion des départements ;
* La gestion des projets ;
* La gestion des autorisations de dépenses (AFE) ;
* L’exécution des autorisations de dépenses (AFE) ;
* L’implémentation une base de données complète pour le suivi et contrôle des dépenses.

## 5. Conclusion

Au terme de ce chapitre, nous avons effectué une présentation succincte du Ministère du Pétrole, ses attributions, missions, et son organisation générale, en mettant un accent particulier sur la DEFSIP, qui représente notre structure d’accueil.

# Chapitre II : Analyse et spécification

# 1. Introduction

Ce deuxième chapitre porte sur l’analyse et spécification. Dans un premier temps, nous entamerons par l’analyse des besoins fonctionnels. L’analyse nous permet de comprendre les besoins de la structure. Cette compréhension induit évidemment à l’identification des acteurs, à la définition des fonctionnalités du système et à l’élaboration des diagrammes de cas d’utilisation qui est la seconde partie de ce chapitre.

# 2. Analyse des besoins

**-Besoins fonctionnels**

Pour répondre aux défis complexes de gestion financière dans le secteur pétrolier du bloc Agadem au Niger, il est essentiel de définir clairement les besoins fonctionnels de notre système automatisé. Ces besoins fonctionnels constituent le cœur de notre approche, visant à fournir une plateforme robuste et efficace pour le suivi et le contrôle des dépenses liées aux activités pétrolières.

* La gestion des utilisateurs : il s’agit d’un outil permettant d’enregistrer, de modifier, de supprimer, de rechercher, d’afficher et d’imprimer les utilisateurs ;
* La gestion des départements : il s’agit d’un outil permettant d’enregistrer, de modifier, de supprimer, de rechercher, d’afficher et d’imprimer les départements ;
* La gestion des projets : il s’agit d’un outil permettant d’enregistrer, de modifier, de supprimer, de rechercher, d’afficher et d’imprimer les projets ;
* La gestion des autorisations de dépense : il s’agit d’un outil permettant d’enregistrer, de modifier, de supprimer, de rechercher, d’afficher et d’imprimer les autorisations de dépense ;
* L’exécution des autorisations de dépense ;
* Modification de mot de passe ;
* Sécurité et confidentialité : Garantir la sécurité des données financières sensibles en mettant en œuvre des mesures de sécurité robustes, telles que l'authentification, et la gestion des accès ;

**-Besoins non fonctionnels**

En plus des fonctionnalités essentielles, notre système de suivi et de contrôle des dépenses doit également répondre à des critères non fonctionnels cruciaux. Ces besoins sont entre autres :

* Facilité d’utilisation : Offrir une interface utilisateur conviviale et intuitive, avec des fonctionnalités de recherche, de filtrage et de personnalisation pour faciliter l'accès et l'utilisation des informations financières ;
* Evolutivité : Il doit être évolutif pour s'adapter à l'augmentation du volume de données et à l'extension des fonctionnalités à mesure que l'entreprise se développe et évolue dans le secteur pétrolier ;
* Performance : Le système doit être capable de gérer de grandes quantités de données financières tout en maintenant des performances élevées ;
* L’extensibilité : dans le cadre de ce travail, l'application devra être extensible, c'est-à-dire qu'il pourra y avoir une possibilité d'ajouter ou de modifier de nouvelles fonctionnalités ;
* La sécurité : l’application devra être hautement sécurisée, les informations ne devront pas être accessibles à tout le monde, c'est-à-dire que l’application est accessible par un identifiant et un mot de passe attribué à une personne physique ;
* La convivialité **:** l’application doit être simple et facile à manipuler même par des non experts ;
* L’ergonomie : le thème adopté par l’application doit être inspiré des couleurs et du logo type de l’entreprise d’accueil ;

**-acteurs**

Un acteur désigne un rôle externe joué par une personne ou une chose qui interagit avec le système. Les acteurs représentent les rôles joués dans le système. Le tableau 1 ci-dessous donne l’ensemble des acteurs qui interviendront dans le futur système avec leurs rôles.

|  |  |
| --- | --- |
| **Acteurs** | **Rôles** |
| Agent | Gestion des autorisations de dépense : enregistrer, modifier, supprimer, afficher, rechercher et imprimer des AFE.  Modification de son mot de passe |
| Agent\_projet | Gestion de projet : enregistrer, modifier, supprimer, rechercher, afficher et imprimer des projets.  Modification de son mot de passe. |
| Administrateur | Gestion de projet : enregistrer, modifier, supprimer, rechercher, afficher et imprimer des projets.  Gestion des AFE : enregistrer, modifier, supprimer, rechercher, afficher et imprimer des AFE.  Gestion de département : enregistrer, modifier, rechercher, supprimer, afficher et imprimer des départements.  Exécution des AFE : enregistrer, afficher, rechercher et imprimer des AFE.  Modification de son mot de passe. |
| Super\_administrateur | Gestion des utilisateurs : enregistrer, modifier, supprimer, rechercher, afficher et imprimer des utilisateurs.  Gestion de département : enregistrer, modifier, supprimer, rechercher et imprimer, afficher des départements.  Gestion de projet : enregistrer, modifier, supprimer, rechercher, afficher et imprimer des projets.  Gestion des AFE : enregistrer, modifier, supprimer, rechercher, afficher et imprimer des AFE.  Exécution des AFE : enregistrer, rechercher, afficher et imprimer des AFE.  Modification de son mot de passe. |

Tableau 1:Présentation des acteurs du système

# 3. Diagramme des cas d'utilisation général

Le diagramme de cas d’utilisation définit un ensemble d’opérations d’un système tel que l’utilisateur le voit de l’extérieur. Il capture le comportement du système. Cette vision orientée utilisateur nous permet donc d’exprimer les besoins des utilisateurs qui constituent les acteurs du système.

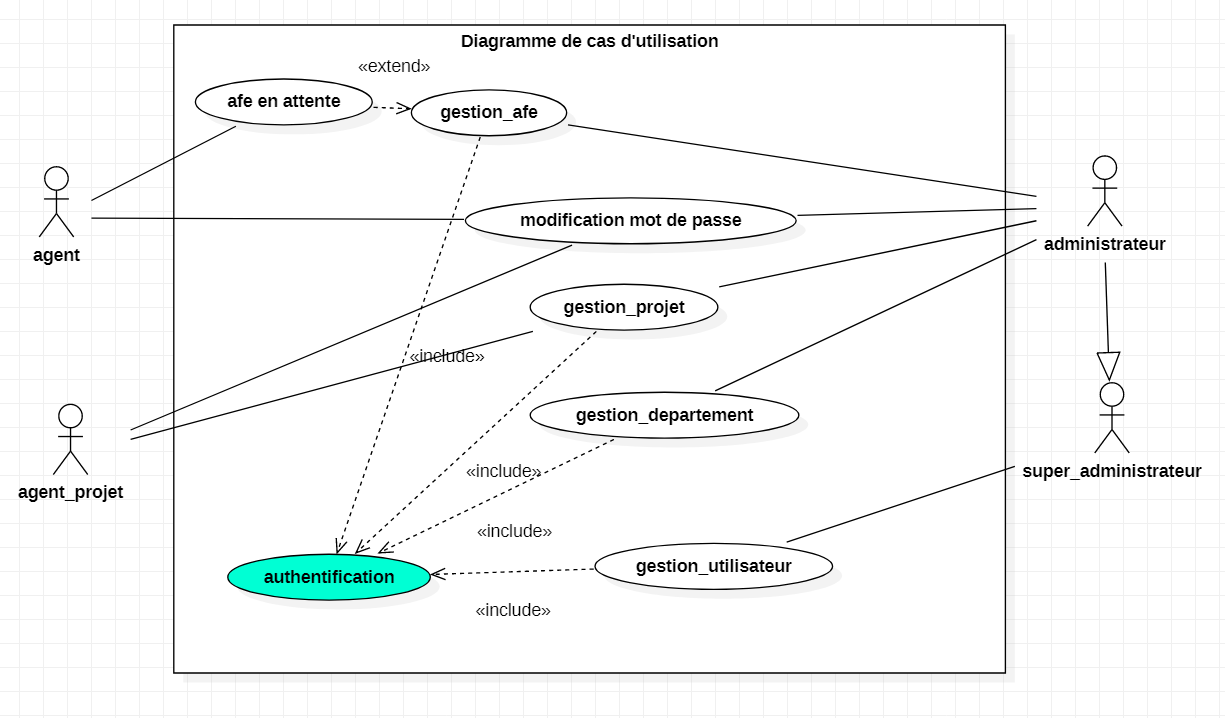
****

Figure 2:diagramme de cas d’utilisation général

* **Sommaire d’identification (gestion des utilisateurs)**

**Titre** : gestion des utilisateurs

**But** : Pouvoir gérer les utilisateurs

**Résumé** : enregistrer, modifier, supprimer, rechercher par mot clé, imprimer la liste des utilisateurs.

**Acteur(s)** : Super\_Administrateur

**Description des enchainements**

**Préconditions** :

-L’utilisateur ne doit pas être déjà enregistré dans la base de données mais doit exister dans le cas de modification ou de suppression.

**Scénario nominal :**

Ce cas d’utilisation commence lorsque le super\_administrateur clic sur le menu ‘gestion des utilisateurs’.

**Enchaînement (a) Enregistrer un utilisateur**

L’administrateur saisi l’utilisateur à enregistrer et clic sur enregistrer

**Enchaînement (b) modifier un utilisateur**

Le super\_administrateur choisi un utilisateur dans la liste puis modifie les informations et valide.

**Enchaînement (c) supprimer un utilisateur**

Le super\_administrateur choisi (clic) sur un utilisateur dans la liste puis confirme la suppression par un clic sur le bouton ‘supprimer’.

**Enchaînement (d) imprimer les utilisateurs**

Le super\_administrateur clic sur ‘imprimer’.

**Enchaînement (e) rechercher un ou des utilisateur(s)**

Le super\_administrateur fourni au système un mot clef et il valide

**[Exception 1 : zone de texte vide]**

Si une zone de texte est vide lors de rechercher ou de modification d’un utilisateur un message indiquant que la zone correspondante doit être remplie est affiché.

**[Exception 2 : dépassement de nombre de caractère].**

Si le nombre de caractère est supérieur aux nombres de caractères prévu un message d’erreur est affiché indiquant que le nombre de caractère maximum est dépassé.

**[Exception 3 : utilisateur inexistant].**

Si le mot clef mis lors de la recherche d’un utilisateur ne correspond à aucun utilisateur enregistrer dans la base de données un message indiquant qu’aucun utilisateur avec ce mot clef n’est trouvé.

**Post conditions :**

- Les utilisateurs enregistrés sont prêts à être imprimés.

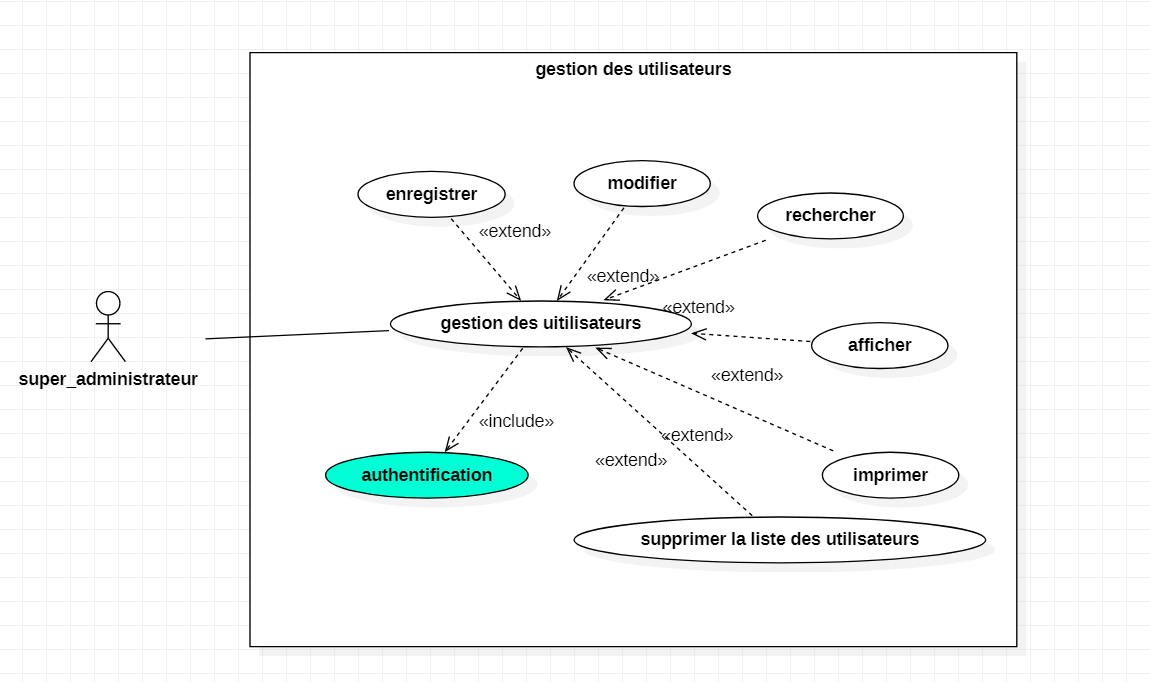


Figure 3:diagramme des cas d’utilisation (gestion des utilisateurs)

**Sommaire d’identification**

**Titre** : gestion des départements

**But** : Pouvoir gérer les départements

**Résumé** : enregistrer, modifier, supprimer, rechercher par mot clé, imprimer la liste des départements.

**Acteur(s)** : Super\_Administrateur et l’administrateur.

**Description des enchainements**

**Préconditions** :

-Le département ne doit pas être déjà enregistré dans la base de données mais doit exister dans le cas de modification ou de suppression.

**Scénario nominal :**

Ce cas d’utilisation commence lorsque l’administrateur clic sur le menu ‘gestion des départements’.

**Enchaînement (a) Enregistrer un département**

L’administrateur saisi le département à enregistrer et clic sur enregistrer

**Enchaînement (b) modifier un département**

L’administrateur choisi un département dans la liste puis modifie les informations et valide.

**Enchaînement (c) supprimer un département**

L'administrateur choisi (clic) sur un département dans la liste puis confirme la suppression par un clic sur le bouton ‘supprimer’.

**Enchaînement (d) imprimer les départements**

L’administrateur clic sur ‘imprimer’.

**Enchaînement (e) rechercher un ou des département(s)**

L’administrateur fourni au système un mot clef et il valide

**[Exception 1 : zone de texte vide]**

Si une zone de texte est vide lors de rechercher ou de modification d’un département un message indiquant que la zone correspondante doit être remplie est affiché.

**[Exception 2 : dépassement de nombre de caractère].**

Si le nombre de caractère est supérieur aux nombres de caractères prévu un message d’erreur est affiché indiquant que le nombre de caractère maximum est dépassé.

**[Exception 3 : département inexistant].**

Si le mot clef mis lors de la recherche d’un département ne correspond à aucun département enregistrer dans la base de données un message indiquant qu’aucun département avec ce mot clef n’est trouvé.

**Post conditions :**

- Les départements enregistrés sont prêts à être imprimés.

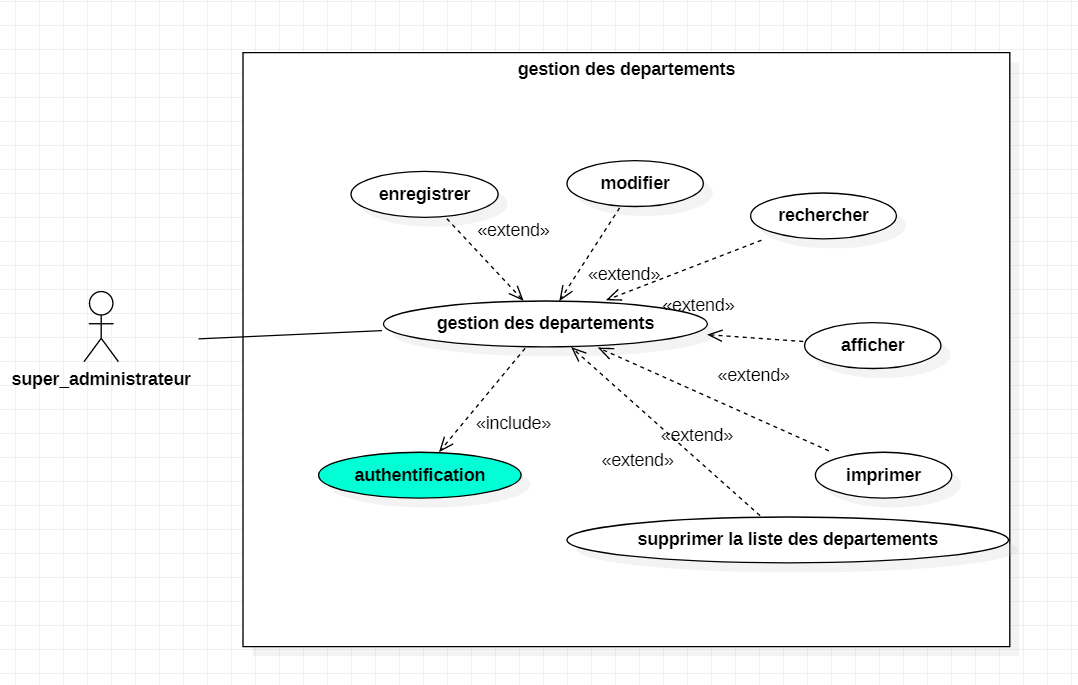


Figure 4:diagramme des cas d’utilisation (gestion des départements)

* **Sommaire d’identification (gestion des projets)**

**Titre** : gestion des projets

**But** : Pouvoir gérer les projets

**Résumé** : enregistrer, modifier, supprimer, rechercher par mot clé, imprimer la liste des projets.

**Acteur(s)** : Super\_Administrateur, l’administrateur et l’agent\_projet

**Description des enchainements**

**Préconditions** :

-Le projet ne doit pas être déjà enregistré dans la base de données mais doit exister dans le cas de modification ou de suppression.

**Scénario nominal :**

Ce cas d’utilisation commence lorsque l’agent\_projet clic sur le menu ‘gestion des projets’.

**Enchaînement (a) Enregistrer un projet**

L’agent\_projet saisi le projet à enregistrer et clic sur enregistrer

**Enchaînement (b) modifier un projet**

L’agent\_projet choisi un projet dans la liste puis modifie les informations et valide.

**Enchaînement (c) supprimer un projet**

L'agent\_projet choisi (clic) sur un projet dans la liste puis confirme la suppression par un clic sur le bouton ‘supprimer’.

**Enchaînement (d) imprimer les projets**

L’agent\_projet clic sur ‘imprimer’.

**Enchaînement (e) rechercher un ou des projet(s)**

L’agent\_projet fournit au système un mot clef et il valide

**[Exception 1 : zone de texte vide]**

Si une zone de texte est vide lors de rechercher ou de modification d’un projet un message indiquant que la zone correspondante doit être remplie est affiché.

**[Exception 2 : dépassement de nombre de caractère].**

Si le nombre de caractère est supérieur aux nombres de caractères prévu un message d’erreur est affiché indiquant que le nombre de caractère maximum est dépassé.

**[Exception 3 : projet inexistant].**

Si le mot clef mis lors de la recherche d’un projet ne correspond à aucun projet enregistrer dans la base de données un message indiquant qu’aucun projet avec ce mot clef n’est trouvé.

**Post conditions :**

- Les projets enregistrés sont prêts à être imprimés.

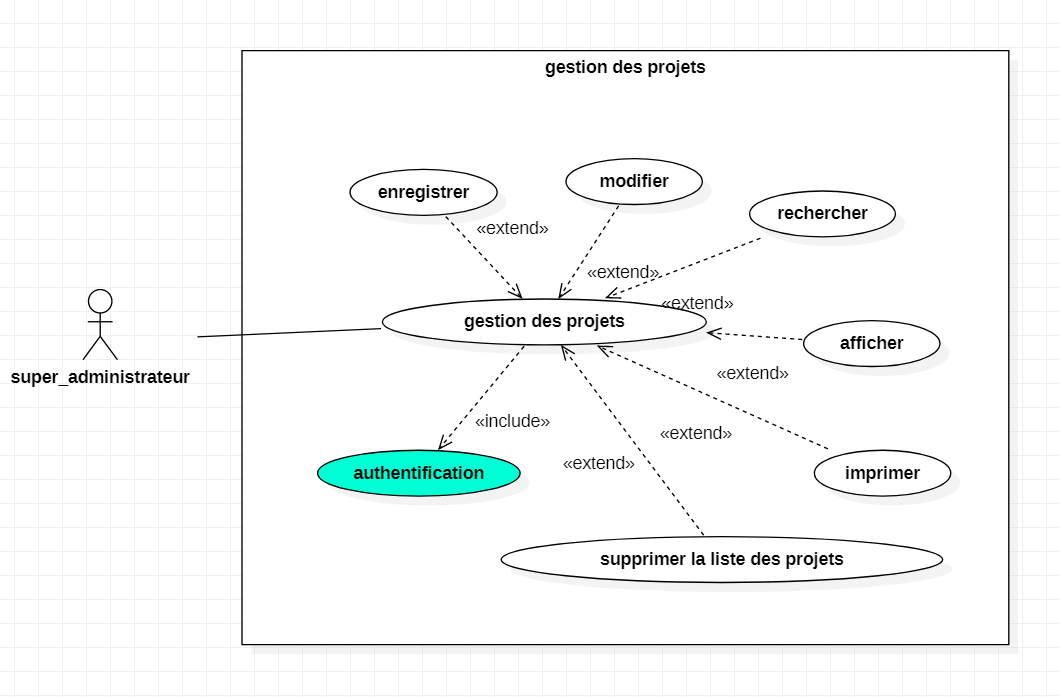


Figure 5:diagramme des cas d’utilisation (gestion des projets)

* **Sommaire d’identification (gestion des autorisations de dépense)**

**Titre** : gestion des autorisations de dépense (AFE)

**But** : Pouvoir gérer les AFE

**Résumé** : enregistrer, modifier, supprimer, rechercher par mot clé, imprimer la liste des AFE.

**Acteur(s)** : Super\_Administrateur, l’administrateur et l’agent

**Description des enchainements**

**Préconditions** :

-L’AFE ne doit pas être déjà enregistrée dans la base de données mais doit exister dans le cas de modification ou de suppression.

**Scénario nominal :**

Ce cas d’utilisation commence lorsque l’agent clic sur le menu ‘gestion des AFE’.

**Enchaînement (a) Enregistrer une AFE**

L’agent saisi l’AFE à enregistrer et clic sur enregistrer

**Enchaînement (b) modifier une AFE**

L’agent choisi une AFE dans la liste puis modifie les informations et valide.

**Enchaînement (c) supprimer une AFE**

L'agent choisi (clic) sur uneAFE dans la liste puis confirme la suppression par un clic sur le bouton ‘supprimer’.

**Enchaînement (d) imprimer les AFE**

L’agent clic sur ‘imprimer’.

**Enchaînement (e) rechercher un ou des AFE (s)**

L’agent fournit au système un mot clef et il valide

**[Exception 1 : zone de texte vide]**

Si une zone de texte est vide lors de rechercher ou de modification d’un projet un message indiquant que la zone correspondante doit être remplie est affiché.

**[Exception 2 : dépassement de nombre de caractère].**

Si le nombre de caractère est supérieur aux nombres de caractères prévu un message d’erreur est affiché indiquant que le nombre de caractère maximum est dépassé.

**[Exception 3 : AFE inexistante].**

Si le mot clef mis lors de la recherche d’une AFE ne correspond à aucune AFE enregistrer dans la base de données un message indiquant qu’aucune AFE avec ce mot clef n’est trouvé.

**Post conditions :**

- Les AFE enregistrées sont prêtes à être imprimées.

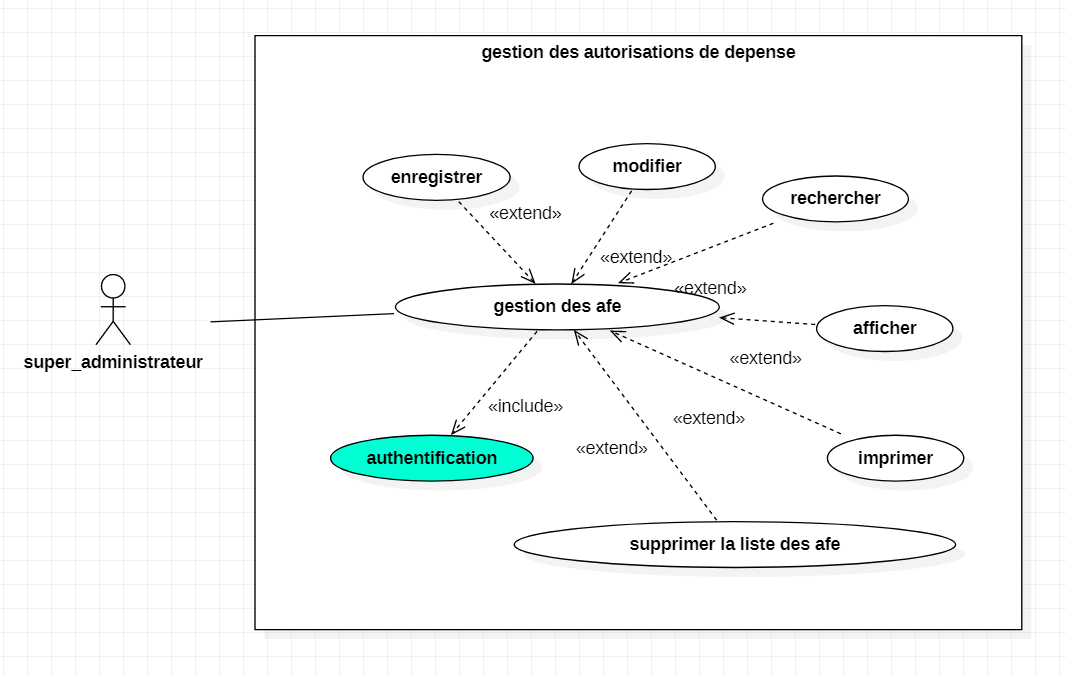
****

Figure 6:: diagramme des cas d’utilisation (gestion des autorisations de dépense)

# 4.Conclusion

Au terme de ce chapitre, nous avons effectué une analyse et spécification de notre projet.

# Chapitre III : Étude conceptuelle

# 1.Introduction

Dans ce chapitre nous allons faire une étude conceptuelle de notre système.

# 2. Description des cas d'utilisation

## 2.1. Cas d’utilisation gestion des utilisateurs (scenario supprimer un utilisateur)

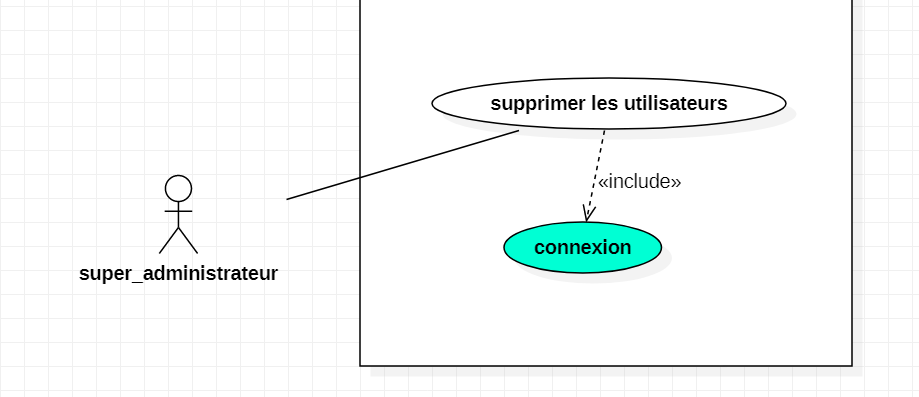


Figure 7:scenario supprimer un utilisateur

**Description textuelle**

Lorsque le super\_administrateur veut supprimer les utilisateurs, il consulte la liste des utilisateurs.

**Scenario**

a. Le super\_administrateur se connecte avec un login et un mot de passe.

b. Il clic sur l’espace gestion des utilisateurs.

c.Il clic sur afficher les utilisateurs.

d. Il clic sur supprimer un utilisateur.

e. Si au moins un utilisateur est enregistré dans le système alors une liste des utilisateurs lui est renvoyée sinon un message indiquant que la liste est vide.

Le diagramme de séquence suivant nous montre l’interaction entre l’utilisateur et le système.

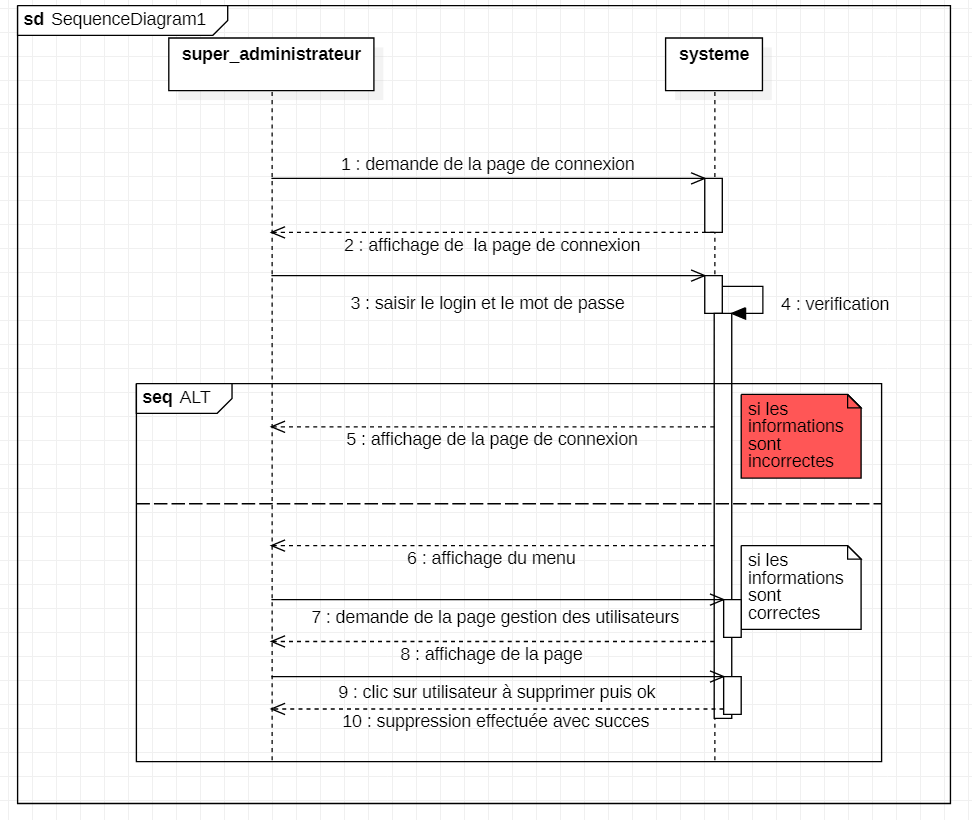


Figure 8:diagramme de séquence (scenario supprimer un utilisateur)

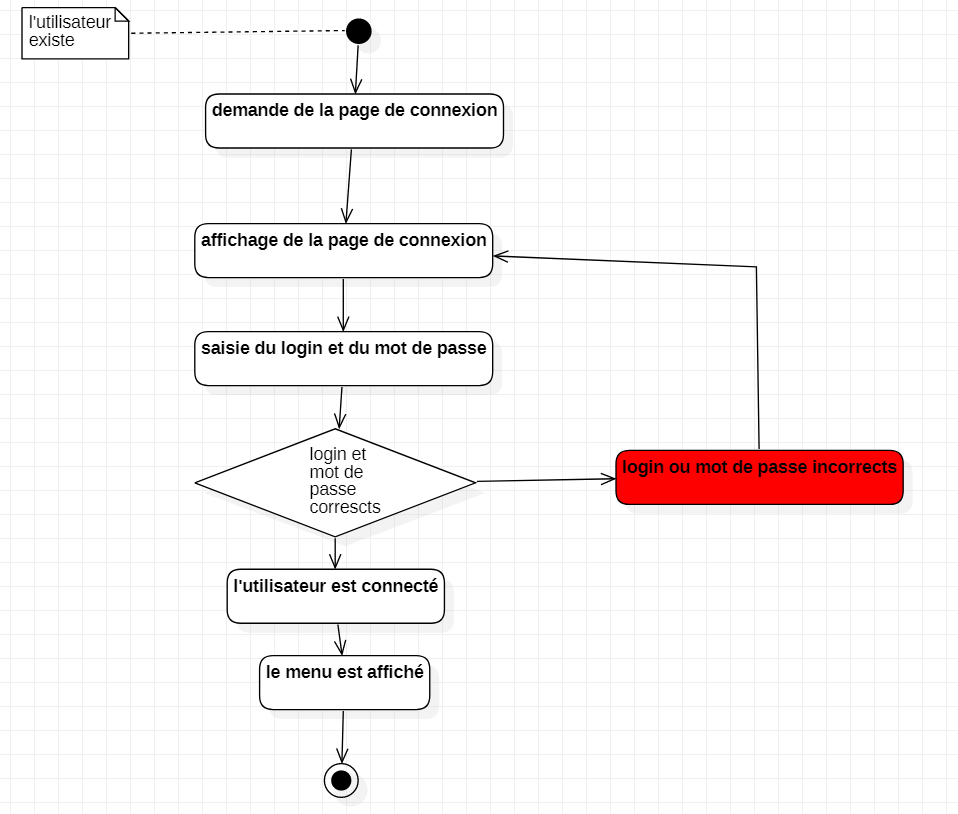


Figure 9:diagramme d’activité (scenario authentification)

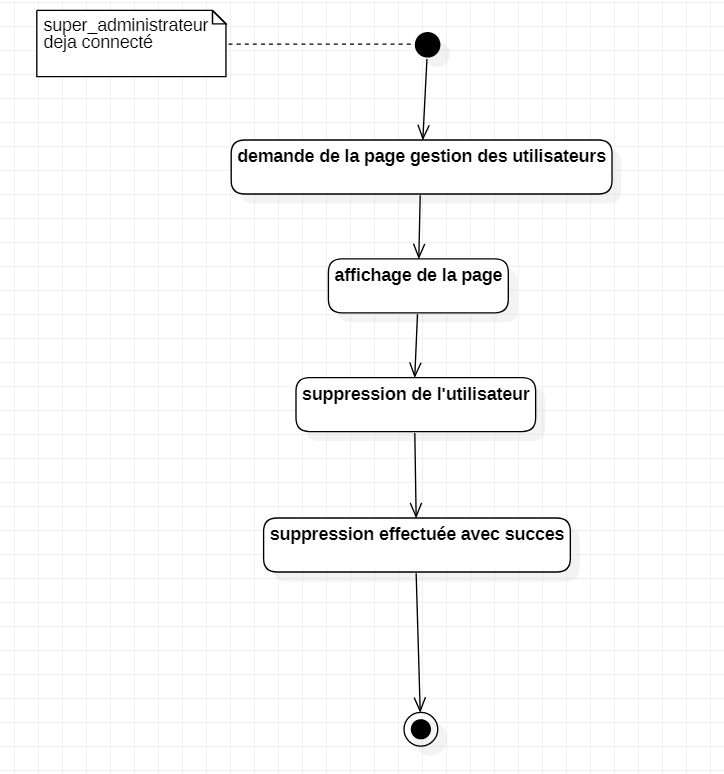


Figure 10:diagramme d’activité (scenario suppression)

**Description du diagramme d’activité <supprimer un utilisateur>**

* Le super\_administrateur demande la page de gestion des utilisateurs
* Le système affiche la page
* Le super\_administrateur supprime l’utilisateur
* Le système envoie un message de confirmation de suppression.

## 2.2. Cas d’utilisation gestion des départements (scenario modifier un département)

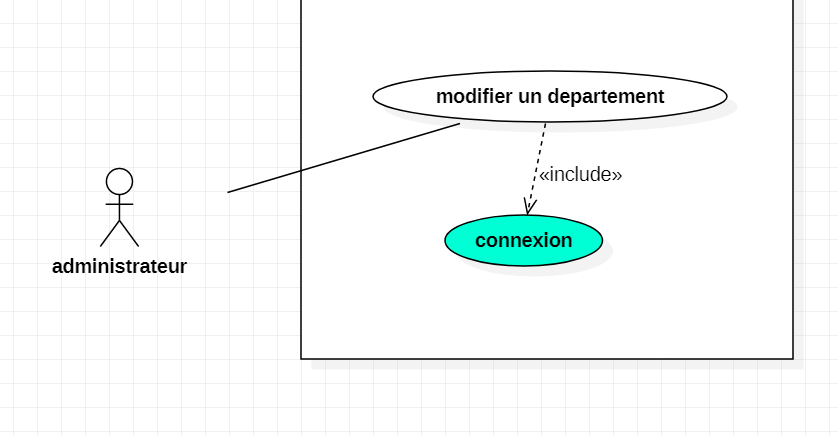
****

Figure 11:diagramme de cas d’utilisation (scenario modifier)

**Description textuelle**

Lorsque l’administrateur veut modifier un département, il consulte la liste des départements.

**Scenario**

a. L’administrateur se connecte avec un login et un mot de passe.

b. Il clic sur l’espace gestion des départements.

c.Il clic sur afficher les départements.

d. Il clic sur modifier un département.

e. Si au moins un département est enregistré dans le système alors une liste des départements lui est renvoyée sinon un message indiquant que la liste est vide.

Le diagramme de séquence suivant nous montre l’interaction entre l’utilisateur et le système.

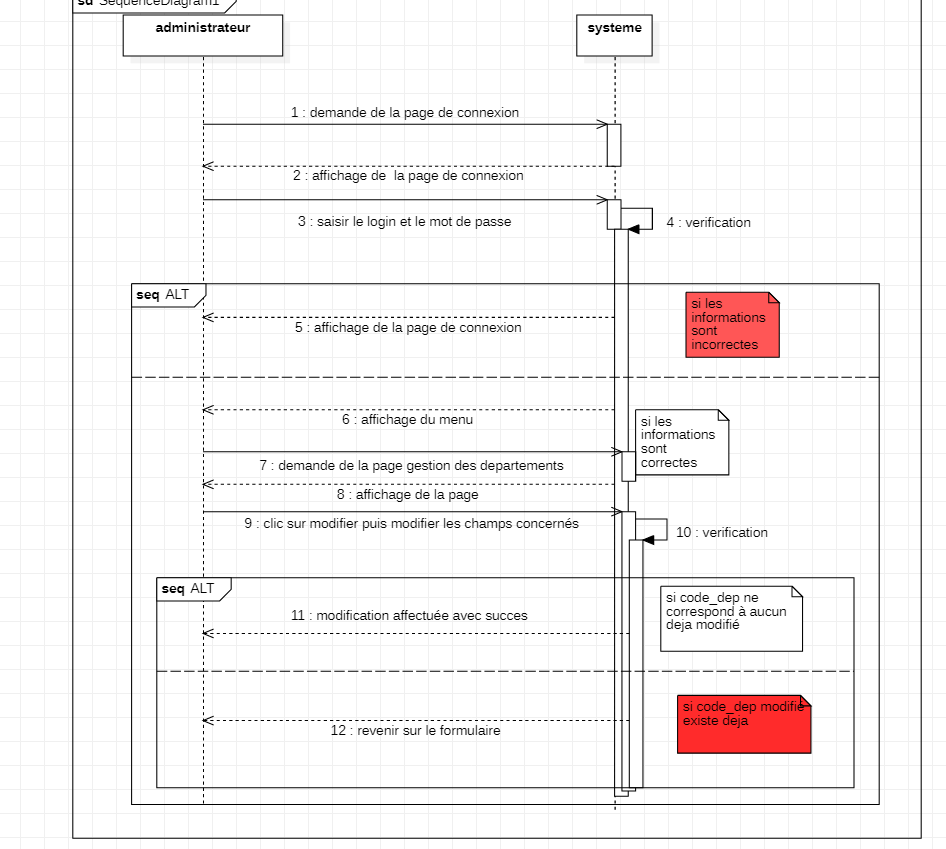


Figure 12:diagramme de séquence (scenario modifier)

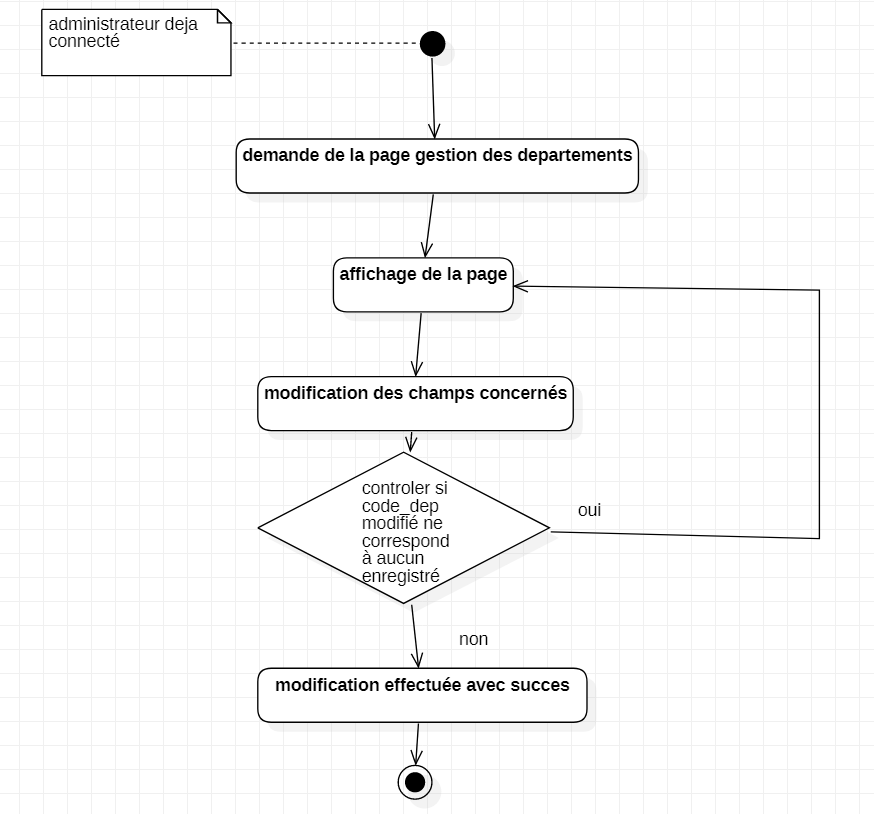
****

Figure 13:diagramme d’activité (scenario modifier)

**Description du diagramme d’activité <modifier un département>**

* L’administrateur demande la page de gestion des départements
* Le système affiche la page
* L’administrateur modifie les champs concernés
* Le système traite les informations modifiées
* En cas d’anomalie le système renvoie au formulaire
* Si non, le système valide la modification

## 2.3. Cas d’utilisation gestion des utilisateurs (scenario enregistrer une AFE)

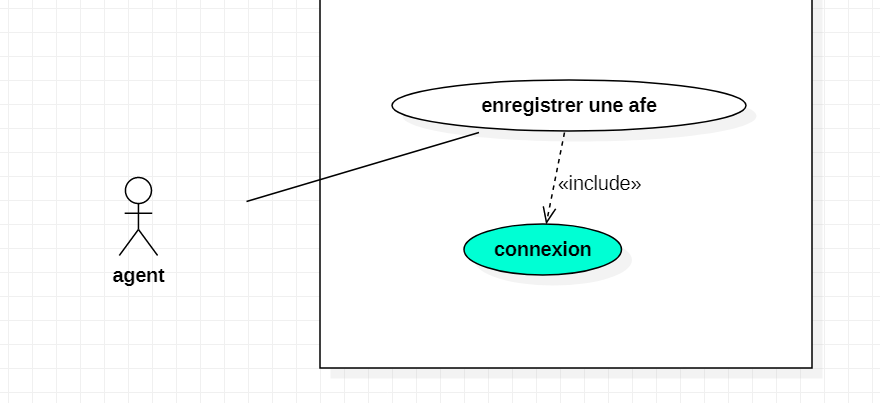
****

Figure 14:scenario enregistrer

**Description textuelle**

Lorsque l’agent veut enregistrer une autorisation de dépense, il consulte l’espace de gestion des AFE.

**Scenario**

a. L’agent se connecte avec un login et un mot de passe.

b. Il clic sur l’espace gestion des AFE.

c.Il saisit les informations dans les différents champs.

d. Il clic sur enregistrer.

e. Si une AFE déjà enregistrée dans le système a le même code alors un message d’erreur lui sera envoyé sinon un message indiquant l’enregistrement est effectué avec succès.

Le diagramme de séquence suivant nous montre l’interaction entre l’utilisateur et le système.

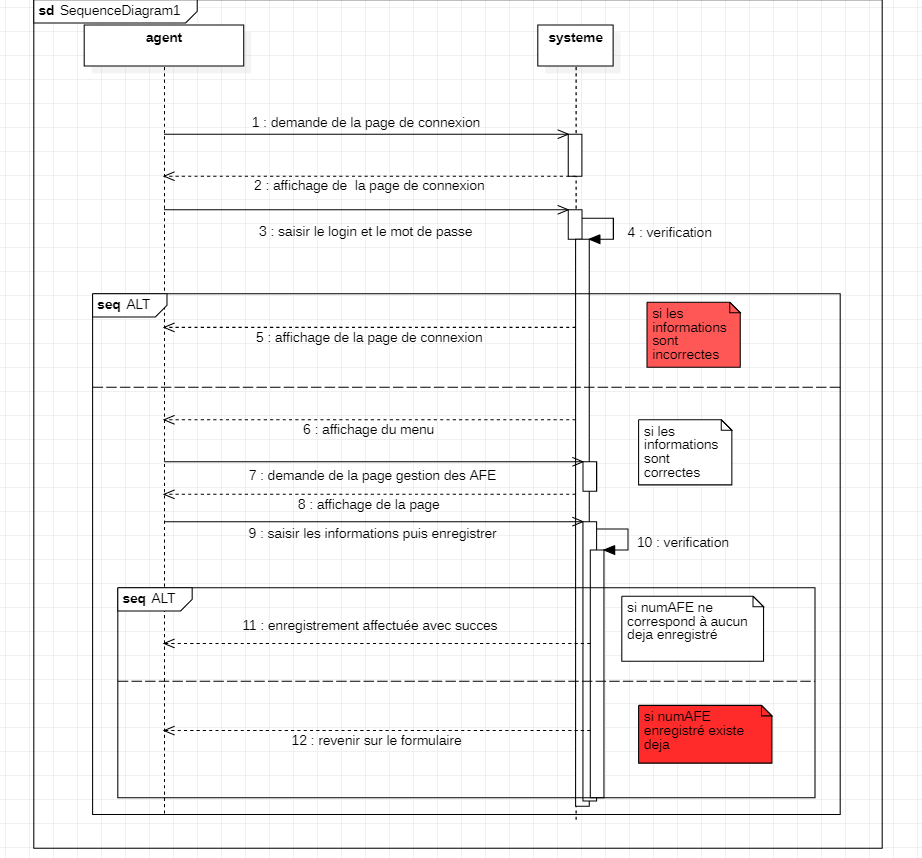


Figure 15:diagramme de séquence (scenario enregistrer)

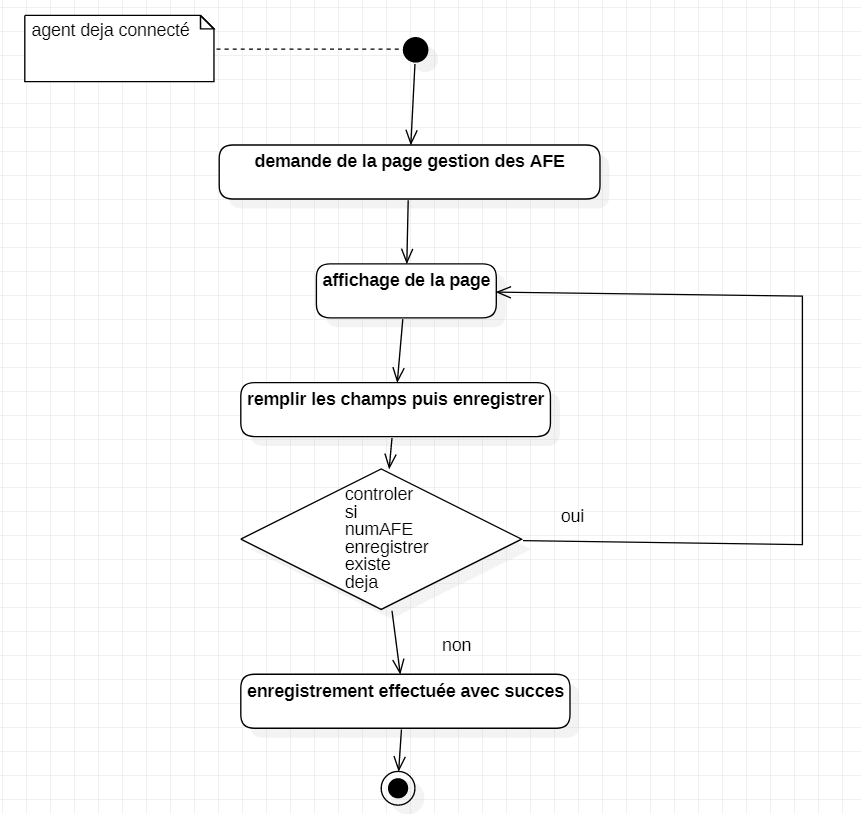
****

Figure 16:: diagramme d’activité (scenario enregistrer)

**Description du diagramme d’activité <enregistrer une AFE>**

* L’agent demande la page de gestion des AFE
* Le système affiche la page
* L’agent saisit les informations
* Le système traite les informations saisies
* En cas d’anomalie le système renvoie au formulaire
* Si non, le système valide l’enregistrement

## 2.4. Cas d’utilisation gestion des utilisateurs (scenario rechercher un utilisateur)

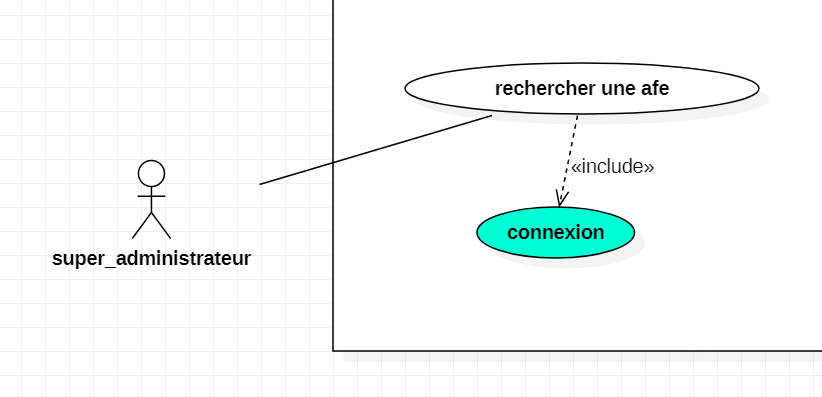
****

Figure 17:scenario rechercher

**Description textuelle**

Lorsque le super\_administrateur veut rechercher un utilisateur, il consulte l’espace de gestion des utilisateurs.

**Scenario**

a. Le super\_administrateur se connecte avec un login et un mot de passe.

b. Il clic sur l’espace gestion des utilisateurs.

c.Il clic sur rechercher un utilisateur en mettant son nom de connexion.

d. Il clic sur ok.

e. L’utilisateur recherché lui sera envoyé s’il existe.

Le diagramme de séquence suivant nous montre l’interaction entre l’utilisateur et le système.

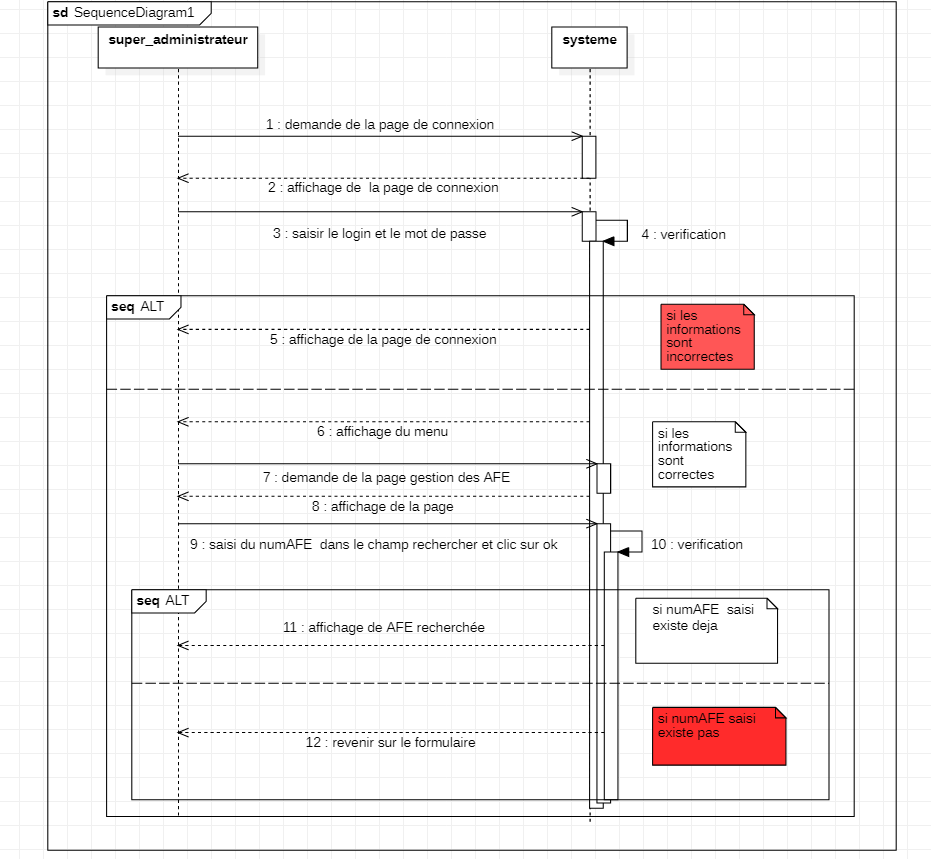


Figure 18:: diagramme de séquence (scenario rechercher)

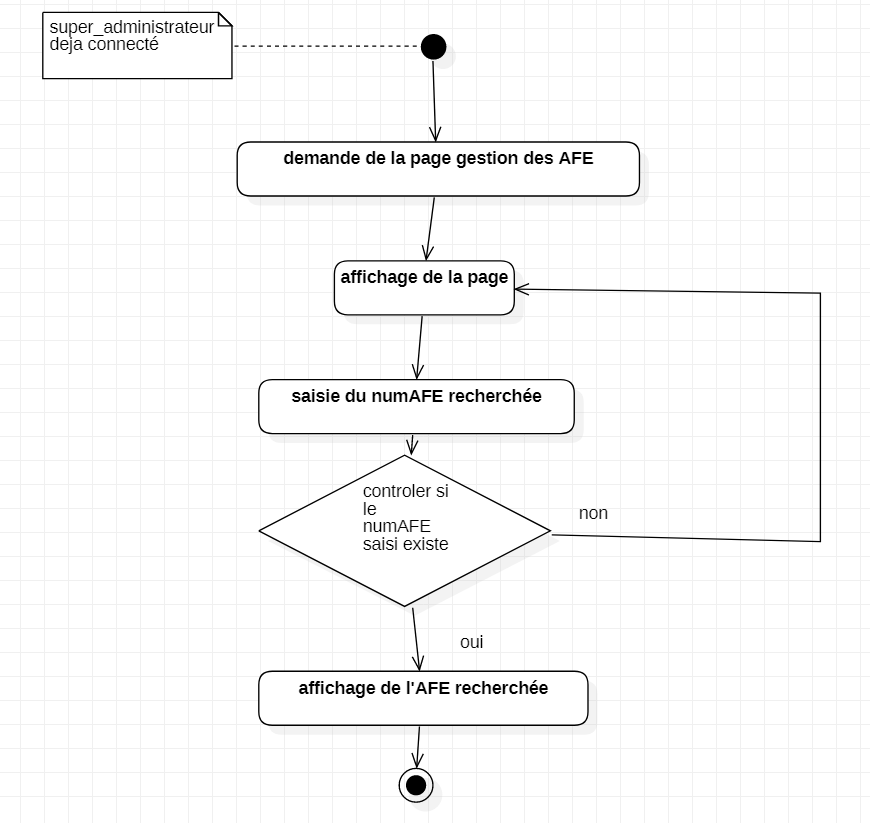
****

Figure 19:diagramme d’activité (scenario rechercher)

**Description du diagramme d’activité <rechercher une AFE>**

* Le super\_administrateur demande la page de gestion des AFE
* Le système affiche la page
* Le super\_administrateur saisit l’identifiant de l’AFE recherchée
* Le système traite les informations modifiées
* En cas d’anomalie le système renvoie au formulaire
* Si non, le système valide la recherche

## 2.5. Cas d’utilisation gestion des utilisateurs (scenario afficher un utilisateur)

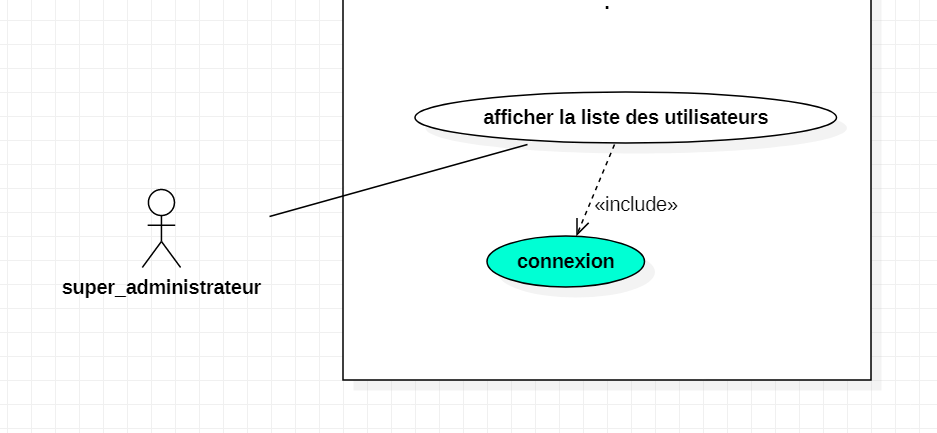
****

Figure 20:scenario afficher

**Description textuelle**

Lorsque le super\_administrateur veut afficher un utilisateur, il consulte la liste de des utilisateurs.

**Scenario**

a. Le super\_administrateur se connecte avec un login et un mot de passe.

b. Il clic sur l’espace gestion des utilisateurs.

c.Il clic sur afficher les utilisateurs.

d. Si au moins un utilisateur est enregistré dans le système alors une liste des utilisateurs lui est renvoyée sinon un message indiquant que la liste est vide.

Le diagramme de séquence suivant nous montre l’interaction entre l’utilisateur et le système.

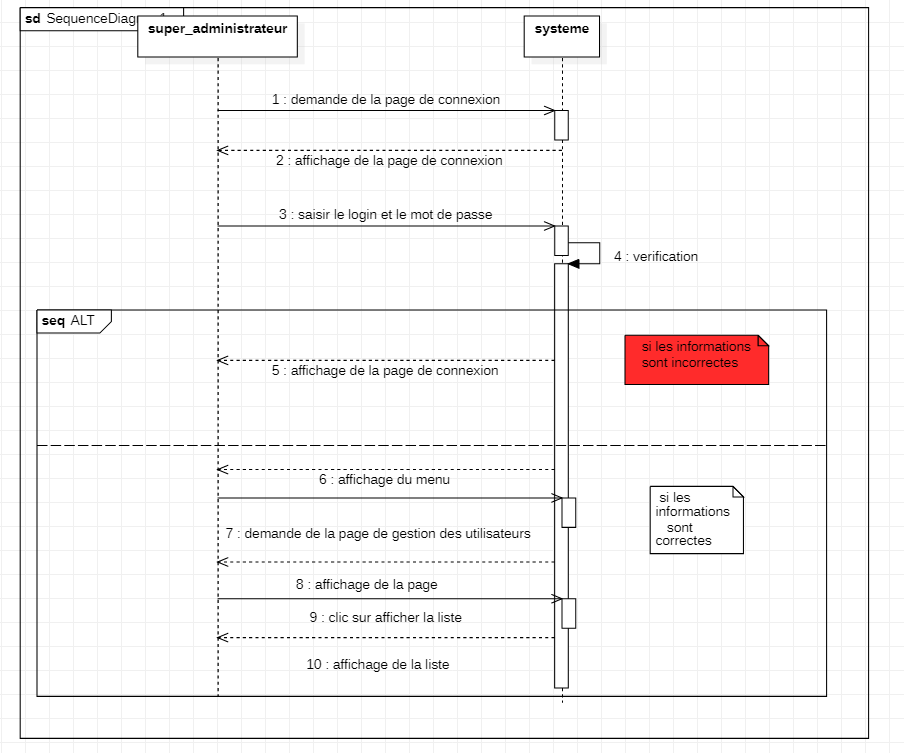


Figure 21:diagramme de séquence (scenario afficher)

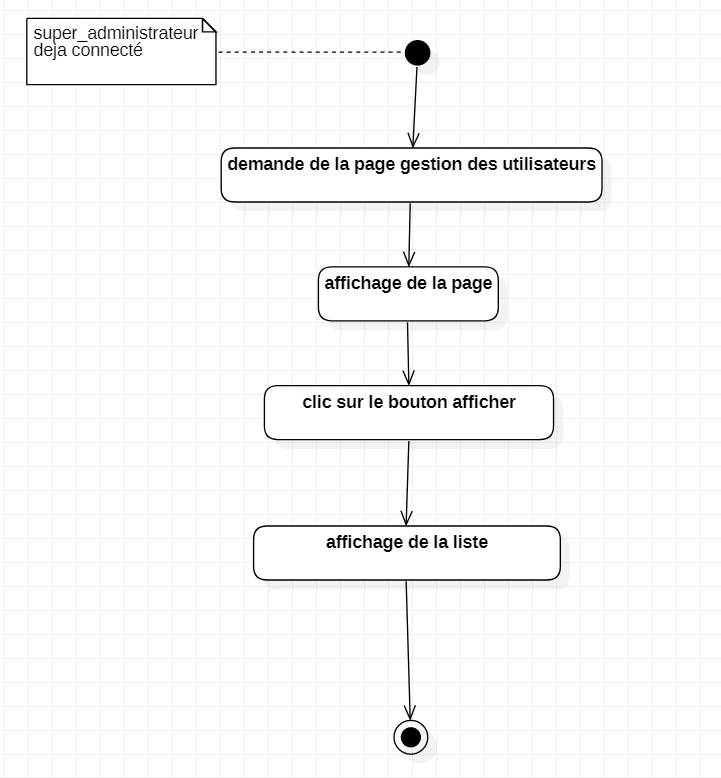


Figure 22:diagramme d’activité (scenario afficher)

**Description du diagramme d’activité <afficher la liste des utilisateurs>**

* Le super\_administrateur demande la page de gestion des utilisateurs
* Le système affiche la page
* Le super\_administrateur clic sue le bouton afficher
* Le système lui affiche la page

## 2.6. Cas d’utilisation gestion des utilisateurs (scenario imprimer la liste des projets)

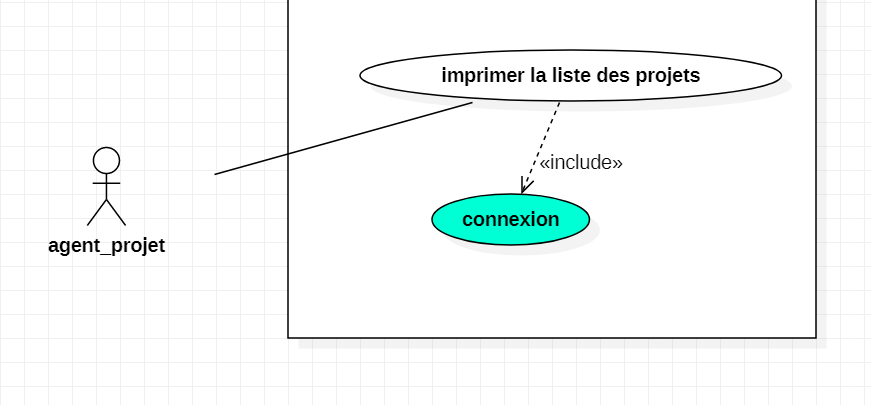


Figure 23:scenario imprimer

**Description textuelle**

Lorsque l’agent projet veut imprimer une liste, il consulte la liste de des projets.

**Scenario**

a. L’agent projet se connecte avec un login et un mot de passe.

b. Il clic sur l’espace gestion des projets.

c.Il clic sur afficher les projets.

D. Il clic sur imprimer.

d. Un message indiquant que l’impression a été effectuée avec succès lui sera envoyé.

Le diagramme de séquence suivant nous montre l’interaction entre l’utilisateur et le système.

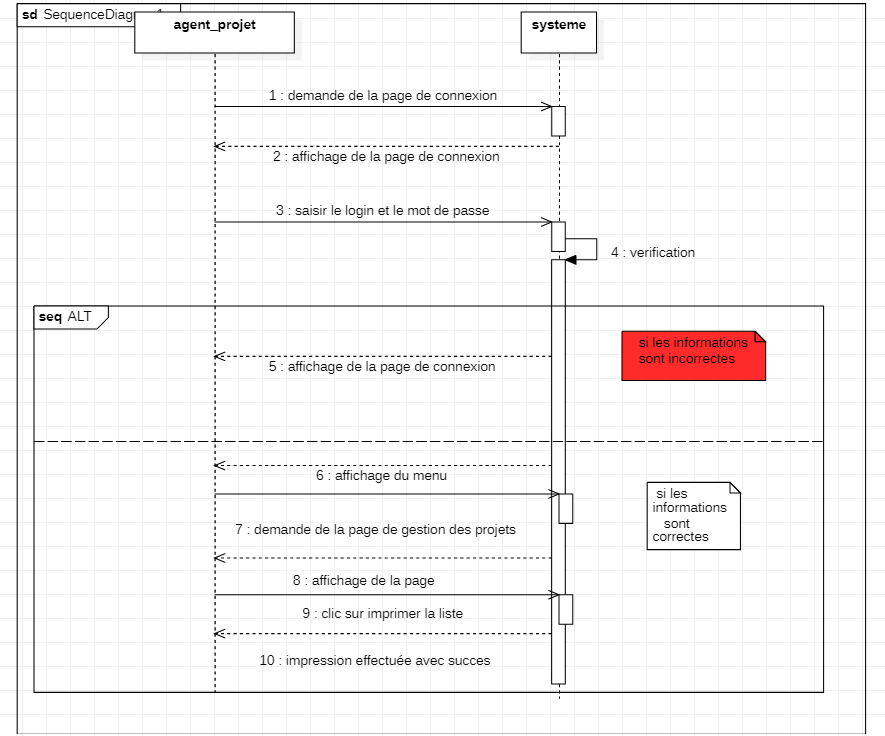


Figure 24:diagramme de séquence (scenario imprimer)

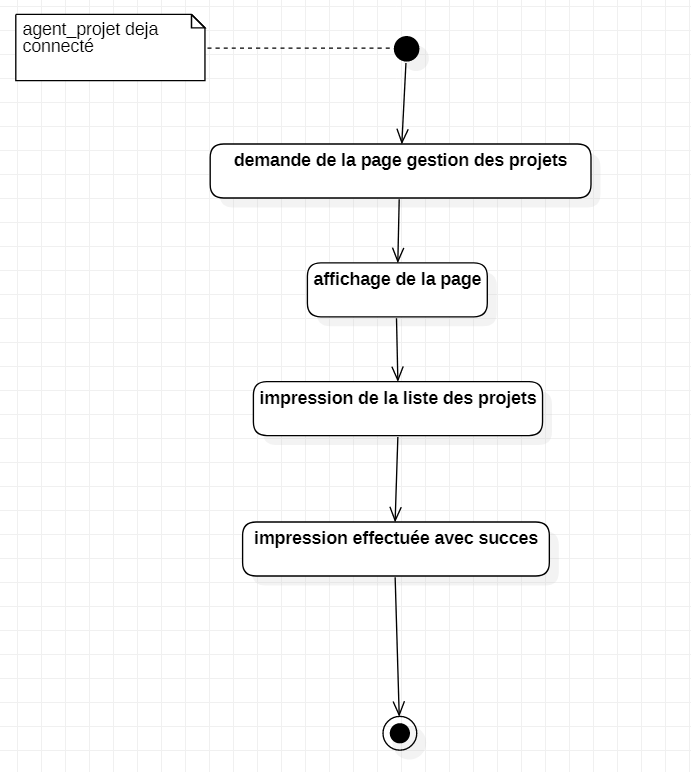


Figure 25:diagramme d’activité (scenario imprimer)

**Description du diagramme d’activité <imprimer la liste des projets>**

* L’agent\_projet demande la page de gestion des projets
* Le système affiche la page
* L’agent\_projet clic sur le bouton imprimer
* Le système valide l’impression

# 3.Diagramme de classe

Une classe est une représentation abstraite d’un ensemble d’objets, elle contient les informations nécessaires à la construction de l’objet. La classe peut donc être considérée comme le modèle, le moule ou la notice qui va permettre la construction d’un objet. Le diagramme de classe est un diagramme structurel (statique) qui permet de représenter :

* Les classes (attributs + méthodes) ;
* Les associations (relations) entre les classes.

Le diagramme de classes est le plus important des diagrammes UML, c’est le seul qui soit obligatoire lors de la modélisation objet d’un système. Il modélise les concepts du domaine d'application, ainsi que les concepts internes créés de toutes pièces dans le cadre de l'implémentation d'une application.

Ainsi, le diagramme de classe suivant représente notre modèle opté pour ce projet.

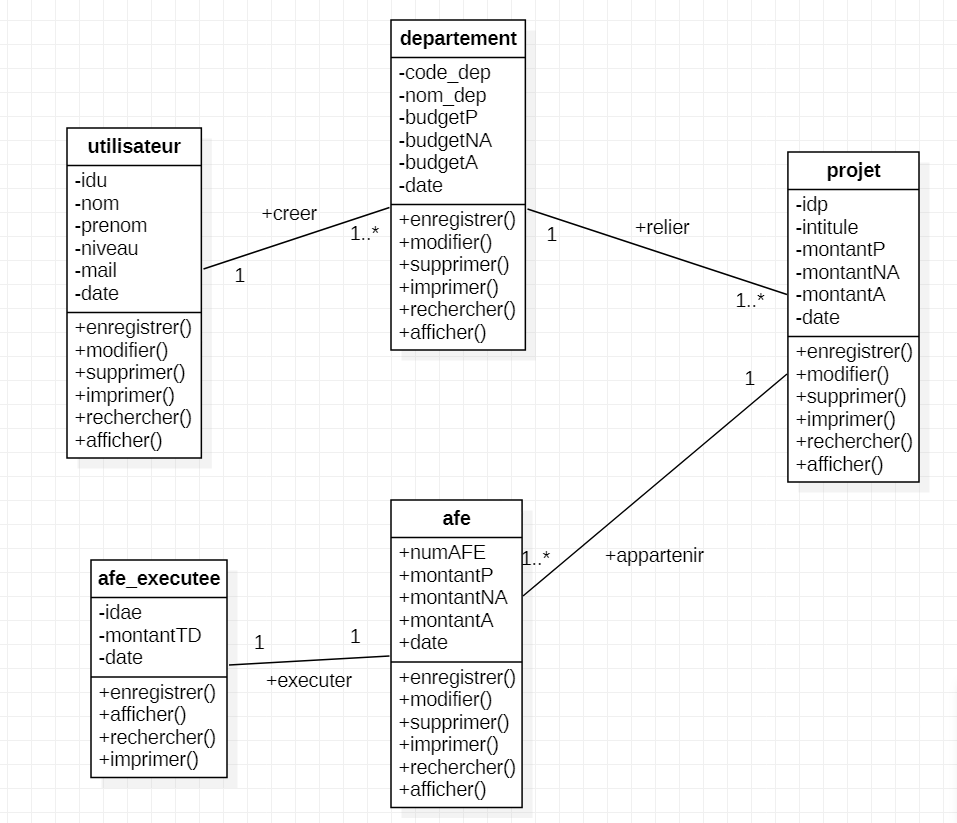


Figure 26:diagramme de classe

# 4.Description détaillée des classes, attributs et des méthodes

## 4.1. Description des classes

|  |  |
| --- | --- |
| Classe | Désignation |
| Utilisateur | La classe qui contient et gère les utilisateurs |
| Departement | La classe qui contient et gère les départements |
| Projet | La classe qui contient et gère les projets |
| AFE | La classe qui contient et gère les AFE |
| Afe\_executee | La classe qui contient et gère l’exécution des AFE |

Tableau 2:Description des classes

## 4.2. Description des attributs

|  |  |  |
| --- | --- | --- |
| Classe | Attribut | Désignation |
| Utilisateur | -idu  -nom  -prenom  -niveau  -mail  -date | -identifiant de l’utilisateur  -nom de l’utilisateur  -prenom de l’utilisateur  -le droit d’accès de l’utilisateur  -l’e\_mail de l’utilisateur  -la date de l’enregistrement de l’utilisateur |
| Departement | -code\_dep  -nom\_dep  -budgetP  -budgetNA  -budgetA  -date | -identifiant du département  -nom du département  -budget prévu  -budget non approuvé  -budget approuvé  -date de l’enregistrement du département |
| Projet | -idp  -intitule  -date  -montantP  -montantNA  -date | -identifient du projet  -intitule du projet  -date de l’enregistrement du projet  -montant prévu pour le projet  -montant non approuvé  -date de l’enregistrement du projet |
| AFE | -numAFE  -montantP  -montantNA  -montantA  -date | -identifiant de l’afe  -montant prévu  -montant non Approuvé  -montant approuvé  -date de l’enregistrement de l’afe |
| Afe\_executee | -idae  -montantTD  -date | -identifiant de l’afe executée  -montant total  -date de l’exécution de l’afe |

Tableau 3:Description des attributs

## 4.3. Description des méthodes

|  |  |  |
| --- | --- | --- |
| Classe | Methode | Désignation |
| Utilisateur | -Enregistrer ()  -Modifier ()  -Supprimer ()  -Rechercher ()  -afficher ()  -Imprimer () | -permet d’enregistrer un utilisateur  - permet de modifier un utilisateur  - permet de supprimer un utilisateur  - permet de rechercher un utilisateur  - permet d’afficher un utilisateur  -permet d’imprimer un utilisateur |
| Département | -Enregistrer ()  -Modifier ()  -Supprimer ()  -Rechercher ()  -afficher ()  -Imprimer () | -permet d’enregistrer un département  - permet de modifier un département  - permet de supprimer un département  - permet de rechercher un département  - permet d’afficher un département  -permet d’imprimer un département |
| Projet | --Enregistrer ()  -Modifier ()  -Supprimer ()  -Rechercher ()  -afficher ()  -Imprimer () | -permet d’enregistrer un projet  - permet de modifier un projet  - permet de supprimer un projet  - permet de rechercher un projet  - permet d’afficher un projet  -permet d’imprimer un projet |
| AFE | -Enregistrer ()  -Modifier ()  -Supprimer ()  -Rechercher ()  -afficher ()  -Imprimer () | -permet d’enregistrer une afe  - permet de modifier une afe  - permet de supprimer une afe  - permet de rechercher une afe  - permet d’afficher une afe  -permet d’imprimer une afe |
| Afe\_executee | -Enregistrer ()  -afficher ()  -recherche r()  -Imprimer () | -permet d’enregistrer l’exécution d’une afe  - permet de modifier l’exécution d’une afe  - permet de supprimer l’exécution d’une afe  -permet d’imprimer l’exécution d’une afe |

Tableau 4:Description des méthodes

# 5. Conclusion

Dans ce chapitre, nous avons présenté, les méthodes de conception les plus connues en premier puis l’étude conceptuelle du système à réaliser à travers les diagrammes présentes. Nous allons alors entamer la phase suivante qui est la phase de réalisation.

# Chapitre IV : Réalisation

# 1.Introduction

Ce chapitre présente la mise en place de notre système de suivi et de contrôle des dépenses liées aux activités pétrolière du bloc Agadem au Niger à travers une multitude d’outils. Ces derniers seront également décrits dans ce chapitre dans lequel nos idées deviennent réelles grâce à la programmation et à la conception d'interfaces.

# 2. Environnement de travail

Pour développer notre application, nous avons utilisé plusieurs outils et nous allons faire leurs présentations dans cette partie.

## 2.1 Outils matériel

L’environnement matériel sur lequel nous avons travaillé est constitué d’un ordinateur portable LENOVO caractérisé par :

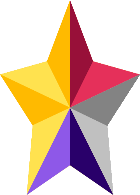
* Modèle : LENOVO
* Processeur : 11th Gen Intel(R) Core (TM) i3-1115G4 @ 3.00GHz 3.00 GHz
* Mémoire : 8,00 Go
* Disque Dur : 474 Go SSD
* Modèle du système : 82H7

## 2.2. Outils logiciels

### 2.2.1. SE

Nous avons utilisé un système d’exploitation 64 bits, processeur x64 pour l’accomplissement de notre travail et un Windows 11 Family.

### 2.2.2. Outils de modélisation UML

Pour modéliser d’une manière claire et précise la structure et le comportement de notre système indépendamment de tout langage de programmation, nous allons adopter le langage de modélisation UML (Unified Modeling Language).

StarUML est un logiciel de modélisation UML, qui a été « cédé comme open source » par son éditeur, à la fin de son exploitation commerciale, sous une licence modifiée de GNU GPL. UML est un langage constitué d’un ensemble de schémas, appelés des diagrammes, qui donnent chacun une version différente du projet à traiter. Aujourd’hui la version StarUML V3 n’existe qu’en licence propriétaire, StarUML gère la plupart des diagrammes spécifies dans la norme UML.

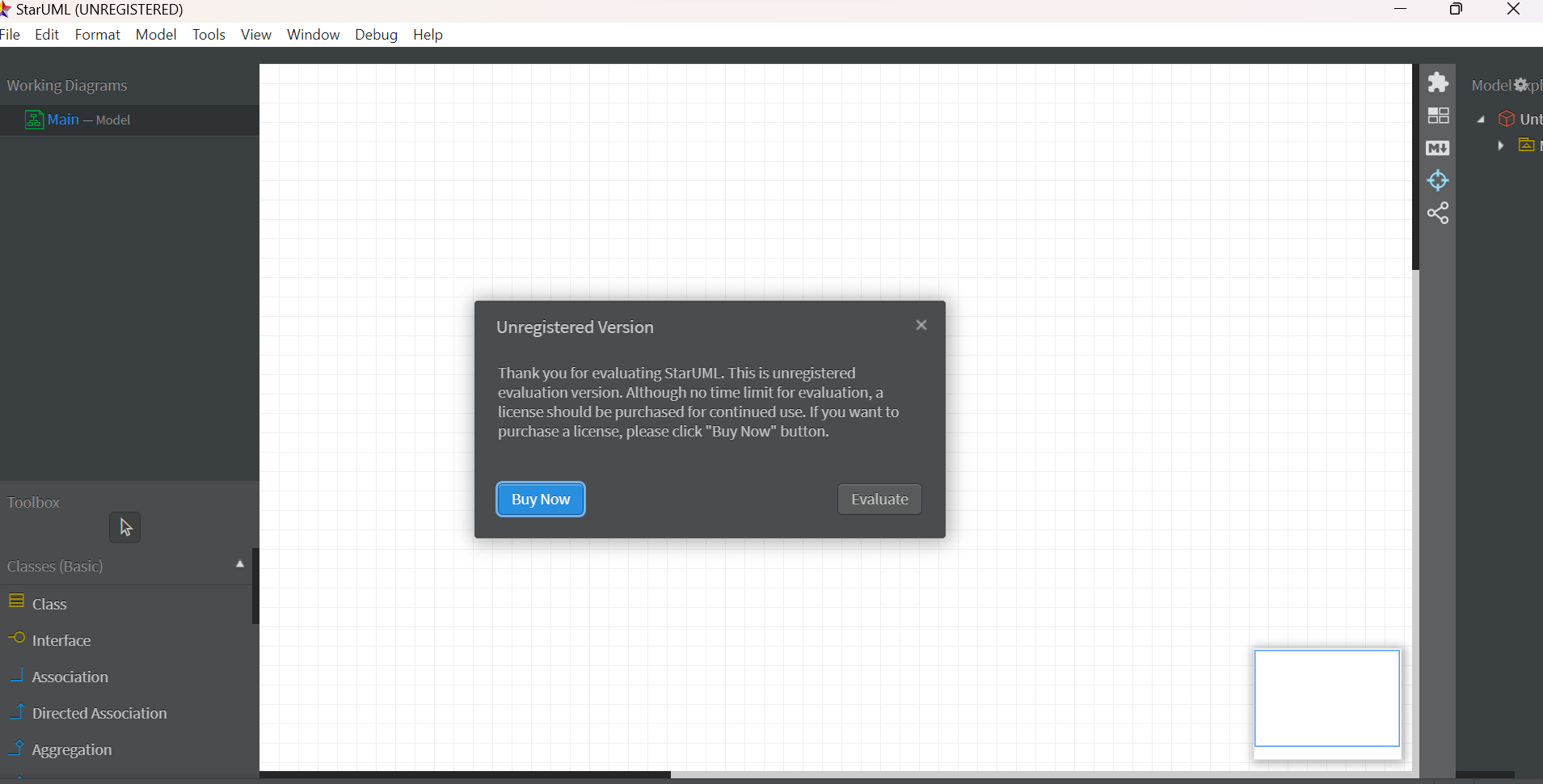


Figure 27:interface de startUML

### 2.2.3. Outils maquettage

Un outil de maquettage est un logiciel ou une application utilisée pour créer des maquettes ou des prototypes visuels interactifs qui représentent l'apparence et le fonctionnement d'un produit, tel qu'un site web, une application mobile ou un logiciel. Ces maquettes sont des versions simplifiées et non fonctionnelles du produit final, conçues pour visualiser et valider le design, l'architecture et les fonctionnalités avant le développement complet.

Les outils de maquettage offrent généralement une variété de fonctionnalités pour faciliter le processus de conception, telles que :

* Création d’interfaces graphiques ;
* Interactivité ;
* Gestion des maquettes ;
* Collaborations ;
* Intégration des données.

Les avantages d'utiliser un outil de maquettage incluent la possibilité de visualiser rapidement les idées de conception, de recueillir des retours précoces des parties prenantes et des utilisateurs, de réduire les risques liés au développement en identifiant et en corrigeant les problèmes de conception en amont, et de faciliter la communication et la collaboration au sein de l'équipe de projet.

#### 2.2.3.1. Balsamiq Wireframes

Dans notre cas nous avons utilisé Balsamiq Wireframes qui est un outil de création de wireframes et de maquettes interactives conçu pour les concepteurs, les développeurs et les équipes de projet. Il offre une interface conviviale et intuitive qui permet de concevoir rapidement des prototypes visuels pour les applications web et mobiles. Les fonctionnalités clés de Balsamiq Wireframes incluent une bibliothèque de composants prêts à l'emploi, des options de collaboration en temps réel, la possibilité d'ajouter des interactions simples pour simuler le comportement des interfaces, et la génération de spécifications détaillées pour les développeurs. Cet outil est largement utilisé pour créer des prototypes fonctionnels et pour communiquer efficacement les concepts de design au sein des équipes de projet.

L’image ci-dessous nous montre un aperçu visuel d'une application de maquettage, mettant en lumière son interface conviviale et ses outils intuitifs pour créer des prototypes interactifs avec facilité et précision.

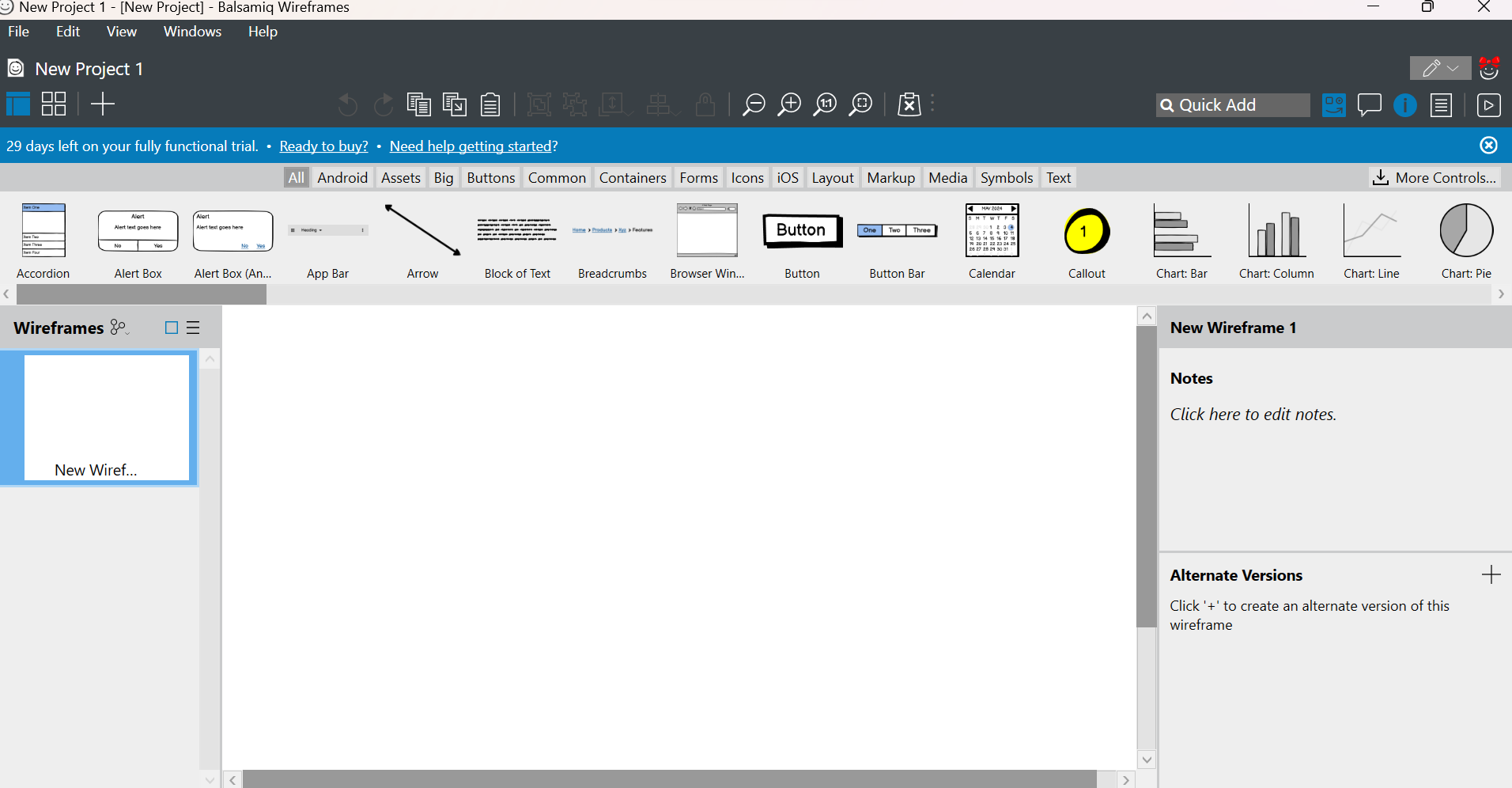


Figure 28:: interface de Balsamiq Wireframes

### 2.2.4. Environnement de développement intégré

NetBeans est un environnement de développement intégré (IDE) opensource multiplateforme, utilisé principalement pour le développement des logiciels en Java, mais qui prend également en charge plusieurs autres langages de programmation tels que C, C++, PHP, HTML, CSS, JavaScript et bien d'autres. Il fournit un ensemble complet d'outils de développement, y compris un éditeur de code, un débogueur, un profilage, un support de création de projet, une intégration de contrôle de version, une interface graphique utilisateur (GUI) pour la création d'interface utilisateur, et bien d'autres fonctionnalités pour simplifier le développement de logiciels.

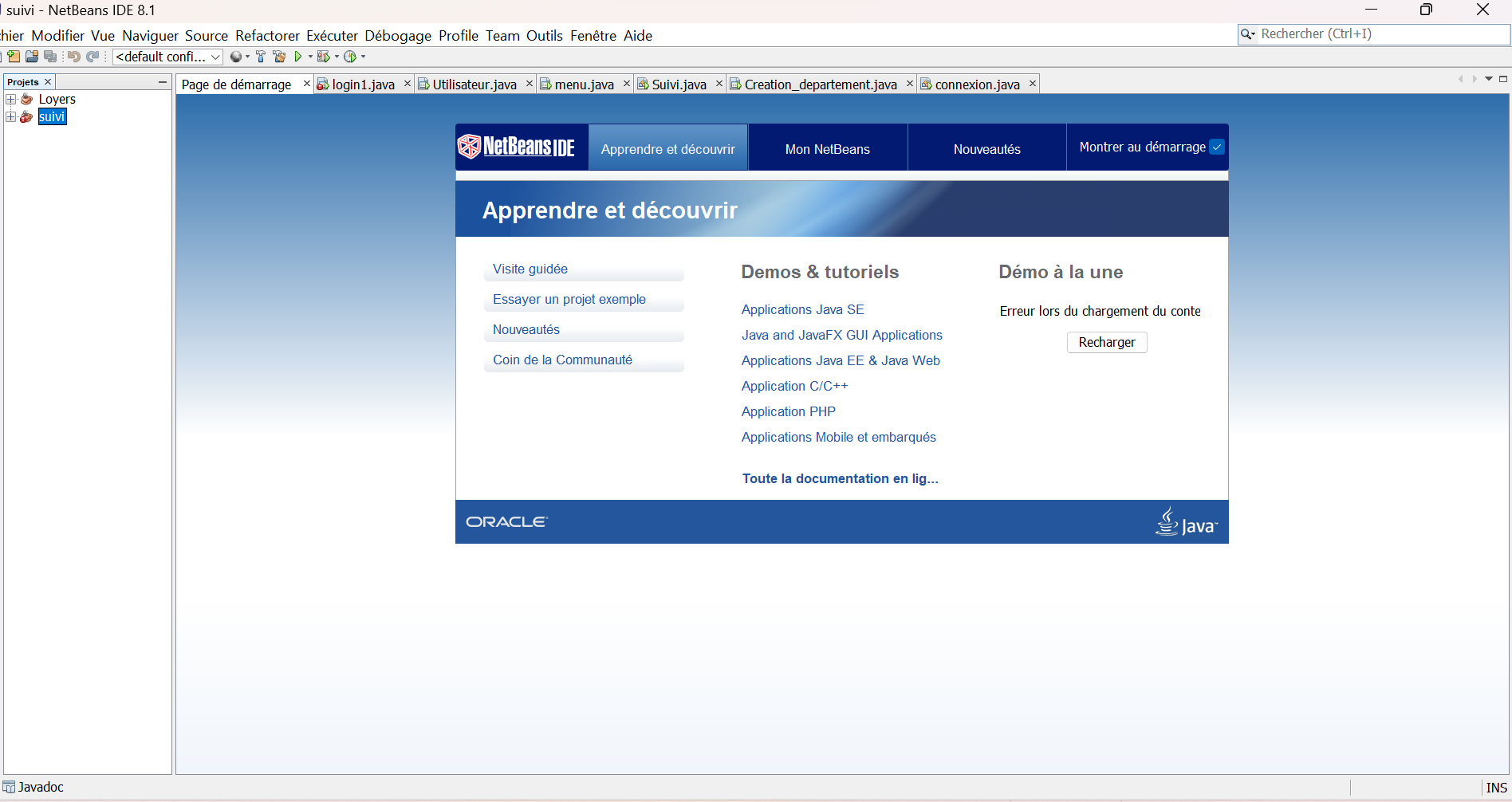


Figure 29:interface de netbeans

### 2.2.5. WampServer

WAMP (Windows, Apache, MySQL, PHP) Server est un ensemble de logiciels open source qui permettent de créer un environnement de développement web sur une machine Windows. Plus précisément, WAMP Server fournit une plateforme pour exécuter des applications web dynamiques développées en utilisant des langages de programmation tels que PHP, Perl, Python et Ruby. Il comprend un serveur web Apache, un système de gestion de base de données MySQL, et le langage de programmation PHP pour le développement web côté serveur.

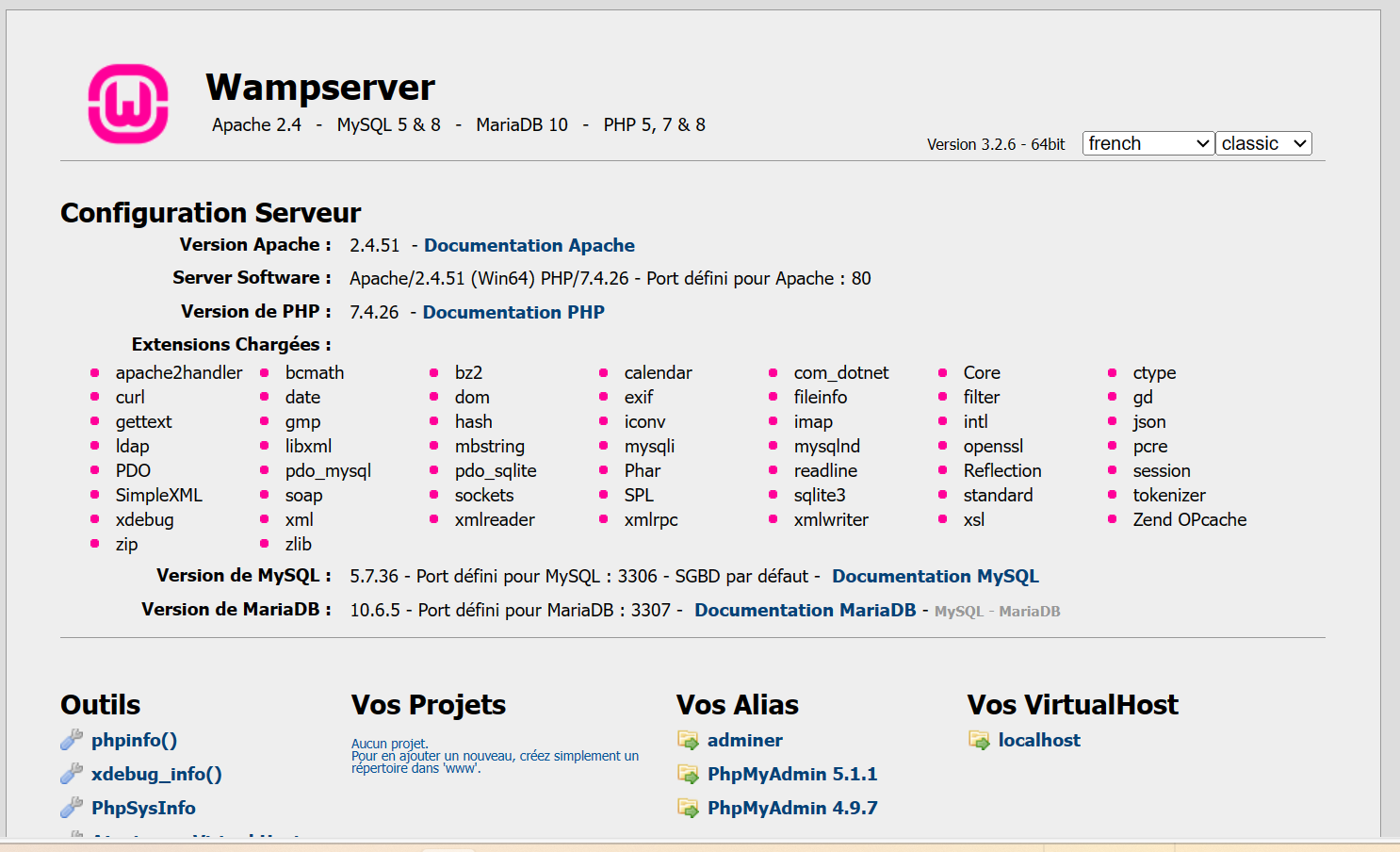
****

Figure 30:interface wampServer

### 2.2.6. PhpMyAdmin

PhpMyAdmin est une application web open source écrite en PHP qui fournit une interface graphique conviviale pour gérer des bases de données MySQL. Il permet aux utilisateurs de créer, modifier et supprimer des tables, des champs, des enregistrements des champs des enregistrements et des relations dans une base de données, ainsi que d'importer et d'exporter des données. PhpMyAdmin est souvent utilisé par les développeurs web pour simplifier la gestion de leurs bases de données MySQL, en évitant d'avoir à écrire des requêtes SQL complexes.

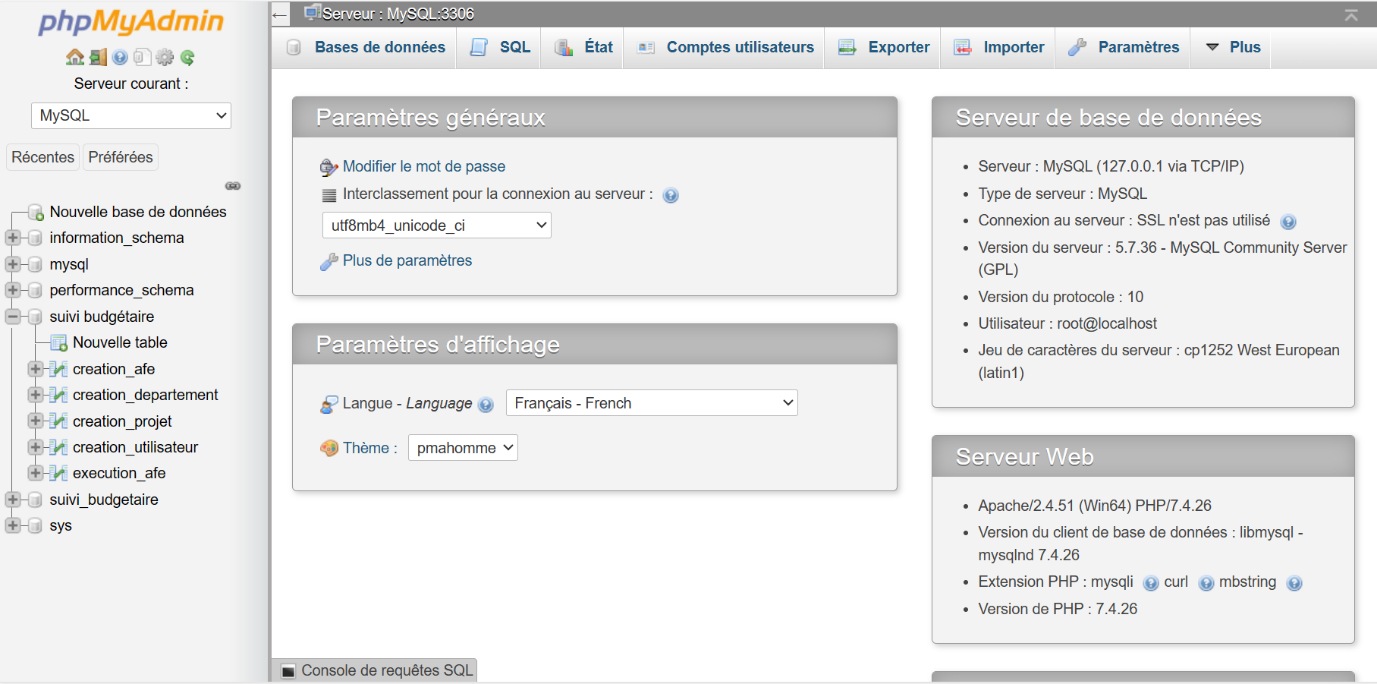
****

Figure 31:: interface PhpMyAdmin

### 2.2.7. Langages de programmation

#### 2.2.7.1. Langage java

Le langage Java a débuté dans les années 1990 avec James

Gosling qui souhaitait développer un langage de programmation indépendant de la plate-forme hardware. Java est une technique informatique développée initialement par Sun Microsystems puis acquise par Oracle suite au rachat de l'entreprise. Il est surtout un langage de programmation portable ce qui signifie qu’il est base sur le concept d’objet qui encapsulent les données et les méthodes. Cela facilite la conception, la conception la modélisation et la maintenance des applications ainsi que la réutilisation du code. Le langage Java peut être défini comme un langage de programmation, Java a évolué pour devenir un ensemble cohérent d'éléments techniques et non techniques. Ainsi, la technologie Java regroupe :

Des standards (la plate-forme Java) définis sous forme de spécification par le Java Community Process (JCP), en trois éditions :

* Java SE (standard édition),
* Java EE (entreprise édition), s'appuyant sur Java SE,
* Java ME (micro édition), indépendante des deux précédentes ;
* Des logiciels (langages informatiques, bibliothèques, frameworks, serveurs d'application, outils d'aide au développement), dont :
* Des implémentations (concurrentes) de ces spécifications,
* Un écosystème d'autres logiciels s'appuyant sur tout ou partie de ces standards, voire leur faisant concurrence ;

#### 2.2.7.2. Langage SQL

SQL (sigle de Structured Query Language, en français langage de requête structurée) est un langage informatique normalisé servant à exploiter des bases de données relationnelles. La partie langage de manipulation des données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données relationnelles. Outre le langage de manipulation des données :

* Le langage de définition des données permet de créer et de modifier l'organisation des données dans la base de données ;
* Le langage de contrôle de transaction permet de commencer et de terminer des transactions ;
* Le langage de contrôle des données permet d'autoriser ou d'interdire l'accès à certaines données à certaines personnes. Créé en 1974, normalisé depuis 1986, le langage est reconnu par la grande majorité des systèmes de gestion de bases de données relationnelles (abrégé SGBDR) du marché.

### 2.2.8. JDK (Java Development Kit)

Le Java Development Kit (JDK) désigne un ensemble de bibliothèques logicielles de base du langage de programmation Java, ainsi que les outils avec lesquels le code Java peut être compilé, transformé en bytecode destiné à la machine virtuelle Java. Il existe plusieurs éditions de JDK, selon la plate-forme Java2 considérée (et bien évidemment la version de Java ciblée) :

* JSE pour la Java 2 Standard Edition également désignée J2SE ;
* JEE, sigle de Java Enterprise Edition également désignée J2EE ;
* JME 'Micro Edition', destinée au marché mobile…….

À chacune de ces plateformes correspond une base commune de Development Kits, plus des bibliothèques additionnelles spécifiques selon la plate-forme Java que le JDK cible, mais le terme de JDK est appliqué indistinctement à n'importe laquelle de ces plates-formes.

# 3.L’architecture de l'application

## 3.1 Architecture 3-tiers

L'architecture 2-tiers est un modèle de conception logicielle où la logique métier et l'interface utilisateur sont séparées en deux niveaux principaux : le niveau client et le niveau serveur.

* **Niveau client :** C'est la partie de l'architecture avec laquelle l'utilisateur interagit directement. Elle comprend l'interface utilisateur, généralement une application web ou desktop, qui communique avec le serveur pour obtenir des données ou effectuer des actions. Dans une application web, le navigateur web du client joue souvent ce rôle.
* **Niveau serveur :** C'est la partie de l'architecture qui gère la logique métier, l'accès aux données et les calculs. Il peut s'agir d'un serveur d'applications, d'une base de données, ou d'un ensemble de services web. Le serveur répond aux requêtes du client, traite les données, effectue des opérations et renvoie les résultats au client.

Dans cette architecture, la communication entre le client et le serveur se fait généralement via des protocoles comme HTTP pour les applications web, et des protocoles spécifiques pour les applications desktop ou mobiles.

L'architecture 2-tiers offre une séparation claire entre la présentation et la logique métier, ce qui facilite la maintenance, l'évolutivité et la gestion des applications. Cependant, elle peut parfois présenter des limitations en termes de scalabilité et de performance, notamment lorsque le nombre d'utilisateurs ou la complexité des opérations augmente.

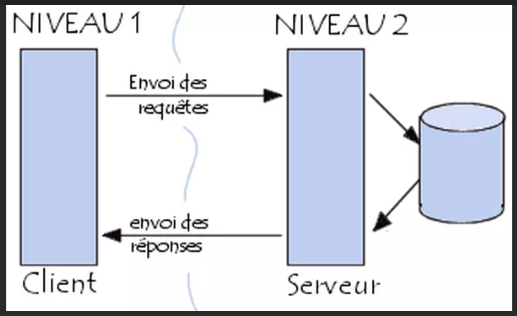


Figure 32:architecture 2 tiers

# 4.Développement

## 4.1 Méthodologie et l’organisation de travail

### 4.1.1 Méthodologie (Cycle en V)

Le cycle en V est un modèle de gestion de projet qui représente graphiquement le déroulement des activités du début à la fin, avec une forte emphase sur la vérification et la validation à chaque étape. Voici les principales phases du cycle en V :

* **Cadrage**: Cette phase initiale consiste à définir clairement les besoins du projet, les objectifs à atteindre et les critères de réussite ;
* **Spécifications** : cette étape permet de construire à l’aide des concepts de base fournis, un modèle su système (logiciel). Ce modèle doit avoir les propriétés du système ;
* **Conception générale** : Sur la base des besoins et des analyses, la conception du système est réalisée. Cela inclut la conception architecturale, la conception détaillée des composants, et la préparation des spécifications techniques ;
* **Conception détaillée** : il s’agit de concevoir la manière dont sera effectuée la logique fonctionnelle codée du produit final ;
* **Implémentation** : À cette étape, le développement réel du système est effectué en suivant les spécifications établies lors de la phase de conception ;
* **Tests unitaires** : Les tests unitaires sont effectués pour chaque composant du système afin de s'assurer qu'ils fonctionnent correctement individuellement ;
* **Test d’intégration** : Les différents composants du système sont intégrés pour former un système complet, puis des tests d'intégration sont effectués pour vérifier leur bon fonctionnement ensemble ;
* **Test de validation** : Cette phase consiste à valider que le système répond aux attentes du client et aux besoins définis au départ ;
* **Recette** : à la fin du cycle de vie du projet, la recette fonctionnelle, appelée également test d’acceptation, permet au client de valider le produit par rapport aux besoins et aux exigences exprimés lors de la première phase, celle de l’analyse du besoin ;

La figure du cycle en V ci-dessous illustre le déroulement méthodique et structuré des activités dans la gestion de projet, mettant en évidence les phases de vérification et de validation à chaque étape du processus.

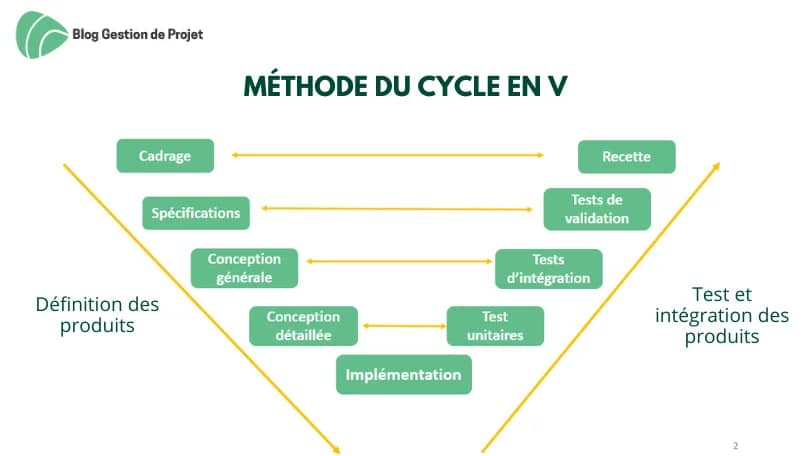
****

Figure 33:cycle en V

### 4.1.2 Organisation

|  |  |  |  |
| --- | --- | --- | --- |
| Nom de la tache | Durée | Début | Fin |
| Organisation |  |  |  |
| Lancement du projet | 1 jour | 01\_04\_2024 | 01\_04\_2024 |
| Cahier de charge | 3 jours | 03\_04\_2024 | 05\_04\_2024 |
| Apprentissage de Github et Git | 3 jours | 23\_04\_2024 | 25\_04\_2024 |
| Analyse et conception |  |  |  |
| Diagramme de cas d’utilisation | 3 jours | 05\_04\_2024 | 07\_04\_2024 |
| Diagramme de classe | 3 jours | 08\_04\_2024 | 10\_04\_2024 |
| Diagramme de séquence | 3 jours | 11\_04\_2024 | 13\_04\_2024 |
| Diagramme d’activité | 3 jours | 14\_04\_2024 | 16\_04\_2024 |
| Analyse | 7 jours | 17\_04\_2024 | 23\_04\_2024 |
| **Réalisation** |  |  |  |
| Cadre théorique | 10 jours | 01\_03\_2024 | 10\_03\_2024 |
| Maquettage | 3 jours | 30\_04\_2024 | 02\_05\_2024 |
| Les interfaces | 7 jours | 20\_03\_2024 | 26\_03\_2024 |

Tableau 5:Organisation

## 4.2 La plateforme Java SE

La plateforme Java SE comprend plusieurs composants essentiels pour le développement d'applications Java :

* **Java Development kit (JDK)**: C'est l'ensemble d'outils nécessaires pour compiler, déboguer et exécuter des programmes Java. Il inclut le compilateur Java (javac), l'outil de débogage (jdb), et d'autres utilitaires nécessaires au développement.
* **Java Runtime Environment (JRE) :** C'est l'environnement d'exécution qui permet aux utilisateurs de lancer des applications Java sur leur machine. Le JRE comprend la machine virtuelle Java (JVM) et les bibliothèques de classes Java standard (comme java.lang, java.util, etc.)
* **Bibliothèques java standard :** Java SE fournit une vaste collection de bibliothèques de classes standard qui couvrent divers domaines tels que l'entrée/sortie, les collections de données, le traitement des chaînes, les opérations mathématiques, etc.
* **Java virtual Machine (jvm) :** C'est une machine virtuelle qui exécute le bytecode Java. Elle permet aux applications Java de s'exécuter de manière indépendante de la plateforme sous-jacente, offrant ainsi la portabilité des applications Java.
* **javaFX :** Java SE inclut JavaFX, une plateforme pour la création d'interfaces graphiques riches (GUI) et d'applications multimédias. JavaFX offre des fonctionnalités avancées pour la conception d'interfaces utilisateur modernes.
* **Outils de développement :** Outre le JDK, Java SE comprend des outils de développement tels que l'IDE (Integrated Development Environment) NetBeans et Eclipse, qui facilitent le développement, le débogage et le déploiement des applications Java.

En résumé, Java SE fournit une base solide pour le développement d'applications Java, en offrant à la fois les outils de développement nécessaires et les bibliothèques standard pour créer des applications robustes et portables.

# 5. Phase de test et de validation

En informatique, un test peut être définit comme une procédure de vérification partielle d'un système. Le fait qu’il est impossible de prouver qu'un logiciel fonctionne bien donc il faut trouver des problématiques maximums de comportements du logiciel. Plus on trouve d'erreurs, plus il y a de chances qu'il y ait d’avantage d'erreurs dans le composant logiciel visé. Les tests de vérification ou de validation visent à s'assurer que ce système réagit de la façon prévue par ses concepteurs (spécifications) ou est conforme aux attentes du client l'ayant commandé (besoins), respectivement. Dans cette partie nous allons présenter une liste des modules de test, scénario de test et le résultat obtenu à partir du tableau suivant :

|  |  |  |
| --- | --- | --- |
| Module | Scenario | Résultat obtenu |
| Connexion à l’application | Entrer du login et du mot de passe corrects | Exécution correcte |
| Entrer du login et du mot de passe incorrects | Exécution correcte |
| Gestion des exceptions | Insertion des données existantes | Exécution correcte |
| Sélection des données inexistantes | Exécution correcte |
| Transfert correcte des informations entre les interfaces | Exécution incorrecte |
| Mise à jour de la BD à chaque opération | Exécution incorrecte |
| Insertion à la base de données | Exécution correcte |

Tableau 6:test

# 6. Les interfaces graphiques

## 6.1. Les maquettes

En interface homme-machine et en ergonomie informatique, le maquettage papier (ou prototypage) est une technique encore très utilisée dans la conception centrée sur l'utilisateur.

Cette technique aide les parties prenantes d'un projet (concepteurs, développeurs, commerciaux, etc.) à concevoir un logiciel ou un site web qui répondent aux attentes et besoins des utilisateurs finaux. Il s'agit le plus souvent des premières idées brutes jetées sur le papier, faites d'esquisses et de dessins, représentant l'agencement et les zones de contenus (ou zoning).

### 6.1.1. Présentation de l’interface authentification

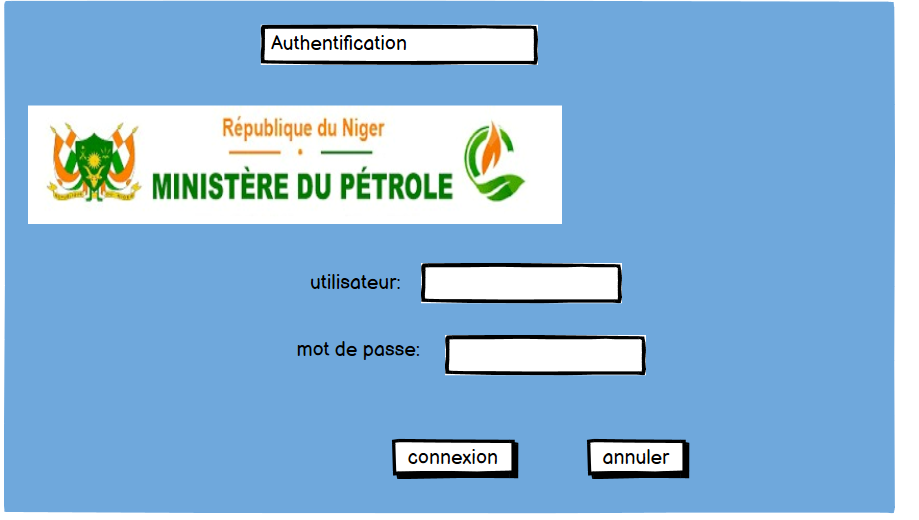
****

Figure 34:maquette authentification

### 6.1.2. Présentation de l’interface menu

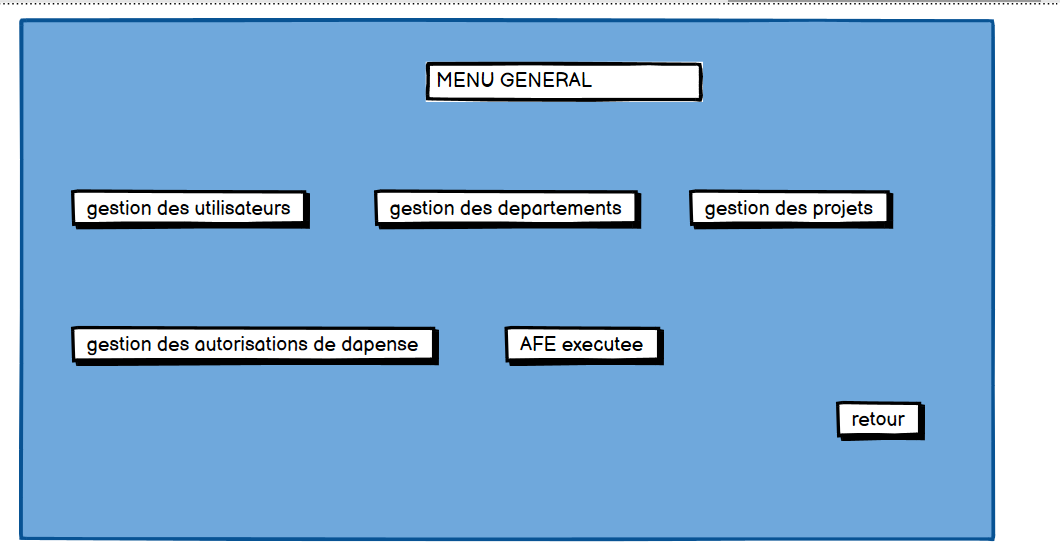
****

Figure 35:maquette du menu

## 6.2. L’application

L’interface graphique est une partie très importante pour la réalisation d’une application convenable et conviviale offrant un certain plaisir à l’utilisateur lors de sa navigation.

Ainsi ce critère peut faire la différence entre une application et une autre bien qu’elles aient les mêmes fonctionnalités.

### 6.2.1. Présentation de l’interface authentification

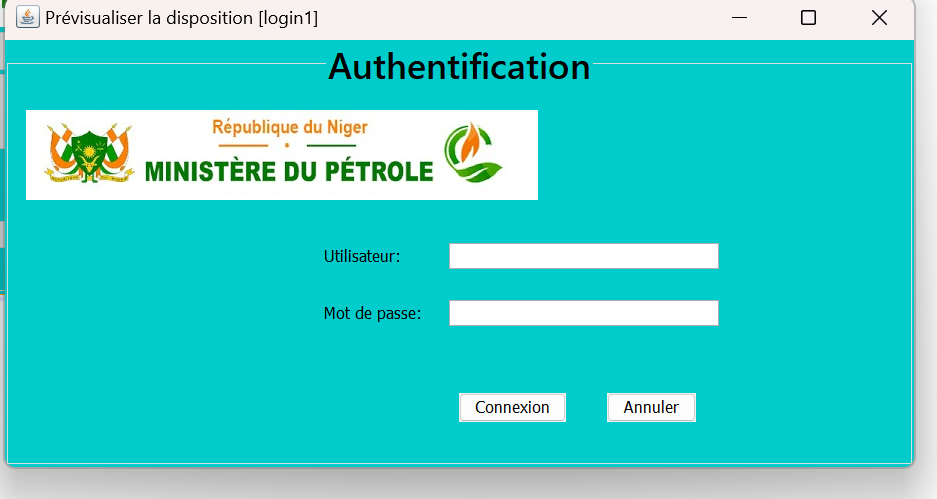
****

Figure 36:interface authentification

### 6.2.2. Présentation de l’interface ajout utilisateur

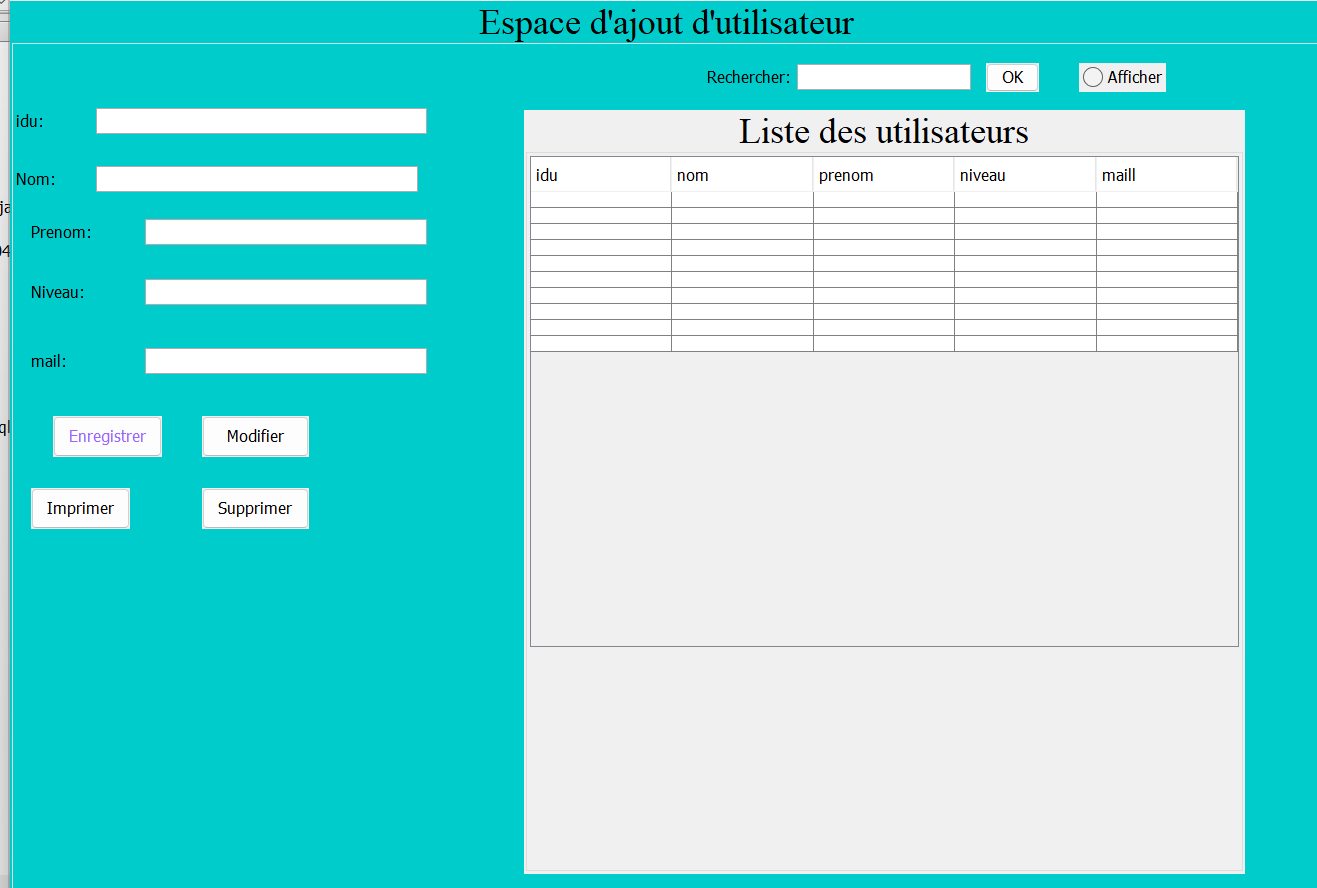
****

Figure 37:interface utilisateur

### 6.2.3. Présentation de l’interface menu général

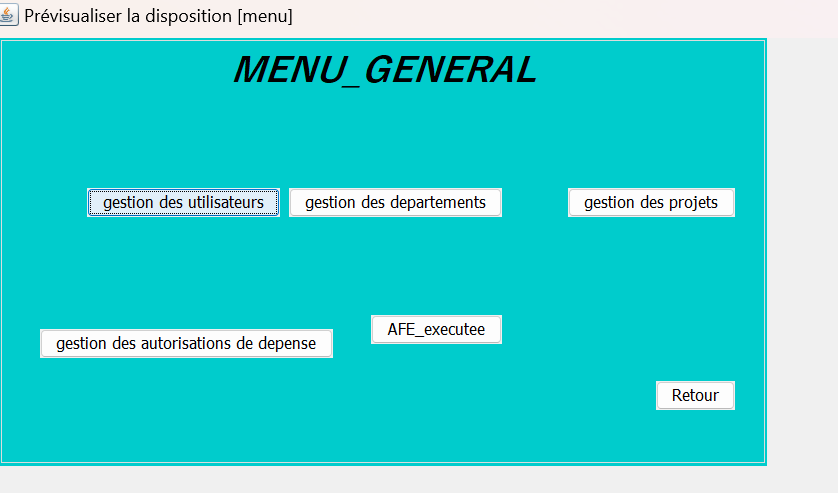
****

Figure 38:interface du menu

# 7. Conclusion

Au cours de ce chapitre, nous avons présenté les différentes phases de la réalisation de notre application. Nous avons commencé par recenser les différentes technologies utilisées ainsi que l’environnement matériel et logiciel de notre travail.

# Conclusion générale

Notre projet consiste à concevoir et mettre en œuvre une application de suivi et contrôle des dépenses liées aux activités pétrolières.

Dans ce travail nous avons présenté les étapes de développement de notre application. Nous nous sommes focalisés également durant tout le processus de conception sur l'UML, qui est un outil graphique pratique pour illustrer notre démarche. Le but de l'application était de montrer l'intérêt de l'informatisation des différentes activités manuelles au sein de l’entreprise en réalisant un prototype de suivi et de contrôle des activités pétrolières en assurant plusieurs avantages par rapport à la gestion manuelle à savoir : Le thème qui nous a été attribué est très instructif sur le plan pédagogique et très intéressant sur le plan technologique et du développement. En tant qu'étudiante en fin de cycle il nous a permis de :

* Accroitre nos connaissances ;
* Initier aux différentes technologies de développement ;
* Améliorer nos compétences dans la programmation orienté objet ;

Finalement étant donné que nul ne peut se prétendre aborder un domaine dans son ensemble nous souhaiterons venir :

* Améliorer les interfaces pour qu’elles répondent aux critères ergonomiques.
* Etablir un système de sécurité des bases de données

# Bibliographie

* Code pétrolier édition 2019
* Conception et réalisation d’une application de gestion de parc informatique de la DGI (GPI-DGI)

# Webographie

[**https://fr.wikipedia.org/wiki/Java\_(technique)**](https://fr.wikipedia.org/wiki/Java_(technique))

<https://fr.wikipedia.org/wiki/NetBeans>

<https://fr.wikipedia.org/wiki/Java_Database_Connectivity>

<https://www.lucidchart.com>

<https://blog.hubspot.fr>

# Annexes

**Annexe1** : extrait du code de la page utilisateur



Figure 39:code utilisateur

**Annexe2** : extrait du code de la page de connexion à la base de données

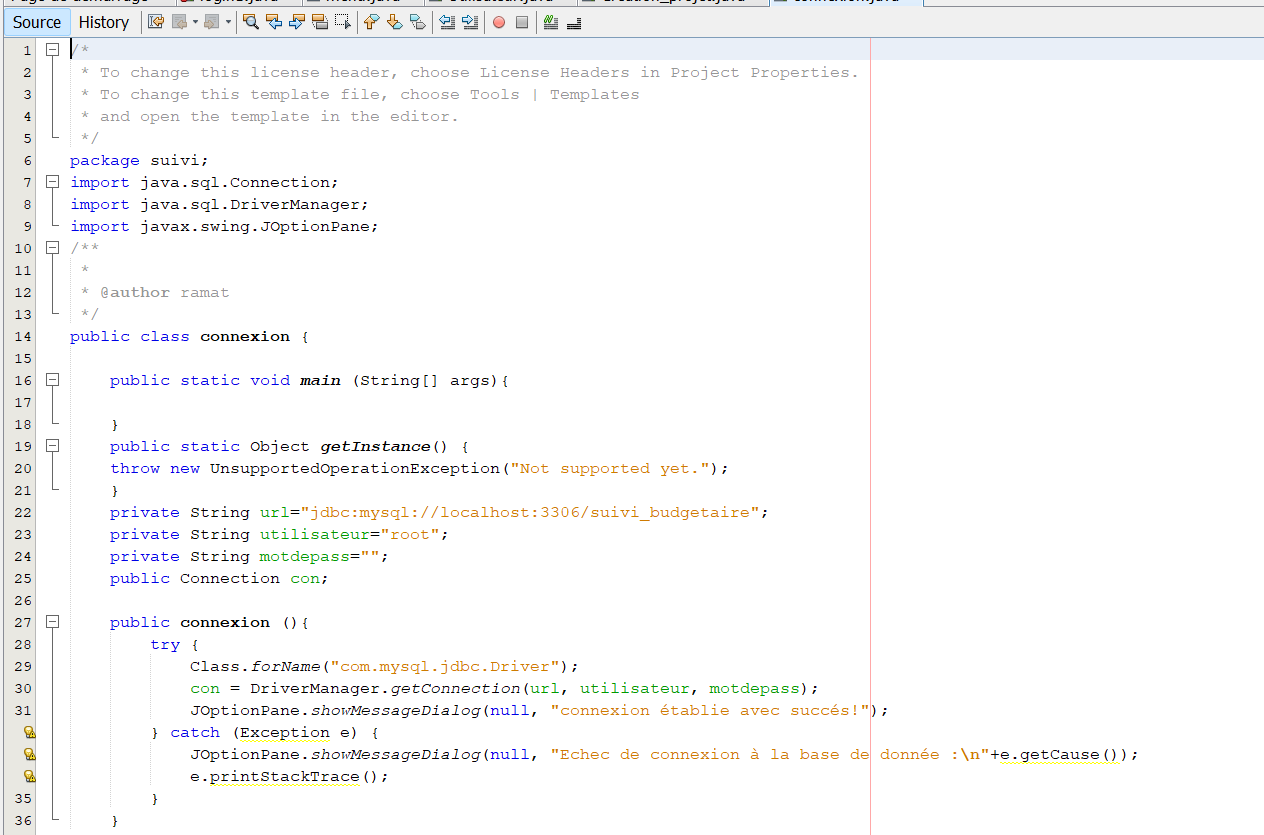


Figure 40:code login