

Filière Ingénierie Des Systèmes

Embarque et Mobile (ISEM)

Stage Fin De 1^{er} Année

Projet Intitulé

**Conception, développement d'une application
de gestion des adresses IP public d'USMBA**

Présenté Par :

JAMALI Hatim

Encadrant :

Mr.JANATI Mohammed

Année Universitaire : 2017/2018

Je dédie ce Modest travail

*A ma mère, ma raison d'être, ma raison de vivre, la lanterne qui éclaire mon
chemin et m'illumine de douceur et d'amour.*

*A mon père, en signe d'amour, de reconnaissance et de gratitude pour tous les
soutiens et les sacrifices dont il a fait preuve à mon égard.*

A mes chers frères

Aucun mot ne pourra décrire vos dévouements et vos sacrifices.

*A tous mes amis, En témoignage de l'amitié sincère qui nous a liés et des bons
moments passés ensemble. Je vous dédie ce travail en vous*

Souhaitant un avenir radieux et plein de bonnes promesses.

*A tous les gens qui ont cru en moi et qui m'ont donné l'envie d'aller en avant,
Je vous remercie tous, votre soutien et vos encouragements me donnent la force
de continuer.*

JAMALI Hatim



Remerciements

Au terme de notre projet de fin d'année, notre gratitude et nos sincères remerciements vont à tous ceux qui ont contribué efficacement à l'élaboration et la réussite de ce projet.

Nous profitons cette occasion pour adresser nos vifs remerciements à nos honorable équipe pédagogique, nos enseignants pour ses efforts, sa disponibilité et ses précieux conseils qui ont été un bon apport au long de cette formation.

Je remercie chaleureusement notre cher encadrant dans l'organisme Mr. JA NATI Mohammed pour son encadrement, son soutien, ainsi que pour les conseils instructifs qu'ils m'ont donné durant la période de ce travail.

Nous tenons aussi à remercier La présidence de l'USMBA pour l'accueil qui nous a été réservé et particulièrement Mr. MRABET Radouane, Président de l'université Sidi Mohamed Ben Abdellah, qui m'a donné la permission de passer ce stage au sein de la présidence de l'USMBA et Mr. EL BEKKALI Moulhime, vice-président pour la mission qui nous a été confiée.

Je remercie aussi vivement Mr. BATA Zakaria pour son aide, son soutien aussi pour ses conseils afin de réussir ce projet, Ainsi Mr. FIRDAOUSI Mohamed et Mr. HAICHAR JIBRIL pour leur disponibilité.

Ainsi, Je tiens à remercier tous les membres du jury qui m'ont fait l'honneur d'accepter de juger mon travail.

Pour tous ceux que j'ai oubliés, je vous dis Merci.

Pour améliorer sa performance, l'entreprise d'aujourd'hui vise à automatiser la gestion interne de ses activités en faisant appel à des technologies informatiques. D'ailleurs c'est le cas de l'USMBA qui souhaite optimiser la gestion interne des adresses IP public à l'aide d'une application Desktop.

Notre projet consiste à identifier et analyser les besoins liés à l'université qui s'articulent autour des modules fonctionnels à savoir : gestion des adresses IP pour chaque établissement.

Pour y arriver, il a fallu tout d'abord une étude de langage utilisé.

A l'aide de ce système unifié, les utilisateurs de différents métiers travaillent dans un environnement applicatif identique qui repose sur une base de données unique. Ce modèle permet d'assurer l'intégrité des données, la non-redondance de l'information, ainsi que la réduction du temps de traitement.

La réalisation de ce projet, quant à elle, est composée de deux parties essentielles qui sont le paramétrage et le développement spécifique des modules fonctionnels.

En fin, pour assurer la fonctionnalité de cette application nous avons déployée sur un PC de la Présidence.

Liste des abréviations

Abréviation	Désignation
UML	Unified Modeling Language
XP	eXtreme Programming
2TUP	Two Track Unified Process
XML	eXtended Markup Language
SQL	Structured Query Language
MVC	Modèle Vue Contrôleur
ORM	Object Relational Mapping
USMBA	Université Sidi Mohamed Ben Abdellah
SI	Système d'information

Table des matières

Table des matières

Dédicace.....	2
Remerciements	3
Résumé.....	4
Liste des abréviations.....	5
Introduction générale	8
Partie 1 : Contexte général du projet.....	9
Chapitre 1 : Présentation de l'USMBA	10
.....	10
1. Présentation de l'organisme d'accueil	11
1.1 Université Sidi Mohamed Ben Abdelah.....	11
1.2 Mission de l'USMBA.	11
1.3 Fiche technique	11
Chapitre 1 : Présentation de la mission de projet de stage	13
.....	13
1. Présentation du projet de fin d'étude	14
1.1 Problématique	14
2.2 Objectif du projet	14
2. Conduite et planification du projet	15
2.1 Méthodologie de travail	15
2.2 Planification de projet	18
Conclusion.....	20
Partie 2 : mise en oeuvre	21
Mise en œuvre.....	21
Chapitre 3 : Etude analytique des besoins	22
Introduction	23
1. Système actuel : présentation des défaillances.....	23
2.1 Présentation du système de gestion actuel :.....	23
2.2 Défaillances du système traditionnel :.....	23
2.3 Analyse des besoins et cahier des charges :.....	24
2.4 Etude fonctionnel :	25
2. MVC : Modèle vue contrôleur	25
Conclusion.....	26
Chapitre 4 : Etude fonctionnelle et conception du système.....	27

<i>1. Analyse et méthodologie de conception</i>	28
1.1 Analyse du projet	28
1.2 Choix de la méthodologie de conception.....	28
1.3 Processus 2TUP	29
1.4 UML (Unified Modeling Language)	31
<i>2. Conception détaillé du projet</i>	32
2.1 Cas d'utilisation.....	32
2.2 Processus métiers	34
2.3 Diagrammes de séquence	35
2.4 Diagrammes de classe	37
2.5 . Les composants de notre système :	38
<i>Conclusion</i>	38
Chapitre 5 : Technologies utilisées.....	39
<i>Définition</i>	40
1. Java	40
2. FXML :	42
3. SQL Lite	43
4 Outils de travail.....	44
4.1 Outils de développement	44
4.2 Conception et modélisation	45
Chapitre 6 : Réalisation	46
<i>Introduction</i>	47
1. Choix de la technologie :	47
2.1. Présentation de la technologie java :	47
2.2. Présentation de JAVA FX (JAVA Flash FLEX) :	48
2. Présentation de l'application :	49
3.1 Formulaire d'authentification :	50
3.2. Gestion des utilisateurs.....	51
3.3. Gestion des adresses IP	53
<i>Conclusion</i> :	55
Conclusion générale	56
Liste des figures.....	57
Liste des tableaux.....	58
Références	59

Introduction générale

De nos jours, savoir gérer les informations, sous contrainte de délai est un défi important pour chaque entreprise, pour relever de tels défis, plusieurs solutions ont été proposées, la solution la plus intuitive serait de développer des applications autonomes dédiées pour chaque structure de l'entreprise. Cette solution peut être au départ satisfaisante, mais elle perd vite son efficacité à long terme, ceci est due au besoin de partager l'information entre ces différents structures.

Pour l'université Sidi Mohamed ben Abdellah, la gestion des adresses IP se fait manuellement ou par des applications de bureautique. L'université souhaite gérer cette activité à l'aide d'une application desktop.

Notre Projet de Fin d'année s'articule autour du développement de solution intégrée pour l'USMBA, notre choix s'est porté alors sur JAVA, qui est un langage évolué utilisé dans la majorité des entreprises.

La démarche de ce travail s'appuie sur une analyse des besoins techniques et fonctionnels qui permettra de souligner les différents processus métiers de l'entreprise, visant à contextualiser le SI (système d'information) de l'organisme.

Partie 1 : Contexte général du projet

*"L'erreur la plus courante que font les analystes est de produire
une conception approximative plutôt que de formuler les
besoins".*

Dijkstra

Chapitre 1 : Présentation de l'USMBA

Dans ce chapitre, nous allons présenter l'organisme d'accueil.

1. Présentation de l'organisme d'accueil

1.1 Université Sidi Mohamed Ben Abdellah

Par Dahir n° 2.75.662 du 11 Chaoual 1395 (17 Octobre 1975) et conformément aux directives Royales visant d'une part à rapprocher les établissements de l'enseignement supérieur des chercheurs et des étudiants et, d'autre part, à intégrer l'université dans son environnement propre et à promouvoir la recherche pour le développement et le bien-être régional et national, a été créé l'Université Sidi Mohamed Ben Abdellah à Fès. Avant la création de l'université, existaient déjà à Fès des établissements annexes des Facultés des Lettres et des Sciences Humaines et des Sciences Juridiques, Economiques et Sociales de Rabat ainsi qu'une cité universitaire.

Et depuis sa création, l'Université Sidi Mohamed Ben Abdellah a connu une expansion importante tant au niveau de la structure et du nombre d'établissement qu'à celui des effectifs des enseignants chercheurs, cadres administratifs et étudiants. [1]

1.2 Mission de l'USMBA.

Elle a une place de choix dans la carte universitaire du Royaume. Elle avait pour mission d'assurer l'enseignement et la recherche scientifique dans les domaines des sciences, des techniques, des sciences économiques, sociales, juridiques et humaines. L'Université Sidi Mohamed Ben Abdellah a su, dès sa création, accompagner du développement socio-économique de la région centre du Nord par la formation de cadres et de techniciens qualifiés et spécialisés. Aujourd'hui, l'Université est engagée dans un vaste programme de réforme qui lui permet de s'ouvrir sur des champs disciplinaires et des formations nouvelles plus diversifiées, créant ainsi de nouvelles perspectives de formation académiques et professionnelles conformément aux recommandations de la Charte Nationale d'Education et de Formation et aux dispositions de la Loi 01.00 portant organisation de l'enseignement supérieur.

1.3 Fiche technique

L'Université Sidi Mohamed Ben Abdellah est une université des performances qui a une diplomatie universitaire de la coopération internationale, ses domaines d'informations sont :

- sciences et techniques ;
- Langues, littérature et communication ;
- Economie et management ;

- Sciences humaines et sociales ;
- Sciences juridiques et politiques ;
- Santé ;
- Sciences de l'éducation ;
- Plantes médicinales et aromatiques.

Elle est très active au niveau de la recherche scientifique telle qu'on trouve 6 centres d'Etudes Doctorales, 81 laboratoires de recherche accrédités, et il y a 60 projets nationaux et internationaux financés en cours. Sans oublier la cité de l'innovation qui est une plateforme technologique qui valorise la recherche, aussi elle appuie la recherche, développement et les projets de recherche collaboratifs.



Figure 1: Domaines d'informations de l'USMBA

Chapitre 1 : Présentation de la mission de projet de stage

***Dans ce chapitre, nous allons décrire le contexte et la conduite
puis la planification du projet.***

1. Présentation du projet de fin d'étude

La présidence de l'Université Sidi Mohammed Ben Abdellah de Fès (USMBA) possède plus de 10 établissements à gérer, qui sont réparties sur plusieurs adresses IP public, plus les adresses privées et les noms des services. Ces adresses qui constituent une dizaine d'établissements, est sous exploité avec le système de gestion actuel qui est une grande partie manuel, malgré les efforts énormes fournies par l'équipe de service informatique (réseaux). En effet, la confusion entre les adresses IP réserver et libérer ne peut pas gérer que par le moyen d'un système de gestion totalement automatisé. Une bonne gestion des adresses IP doit permettre une facilité d'accès à l'information pour les gestionnaires, afin de minimisé le temps de réponse.

1.1 Problématique

Pour gérer la confusion des adresses libre et réserver de l'université, les personnes utilisent des outils basic comme Excel et Word. Ceci génère, non seulement une difficulté d'échange de données entre les différents personnels de service, mais également une confusion de l'information ce que cela implique un retard de repense au client qui a besoin d'une adresse IP.

2.2 Objectif du projet

Notre projet de fin d'année a pour objectif principale d'automatiser un ensemble des processus d'entreprise tout en travaillant sur une base de données unique et homogène, afin de gagner en productivité et réduire les travaux redondants. Aussi, la flexibilité et la rapidité de temps de repense au client.

De plus, ce projet propose une solution répondant aux besoins fonctionnels d'entreprise assurant :

- La cohérence et l'homogénéité de l'information interne ;
- La gestion optimise des adresses IP public, avec la possibilité de consulter tous les informations n'importe quand, et même la possibilité d'insérer des nouvelles données dans la base de données via les interfaces disponibles ;
- La gestion des utilisateurs et de sécurité ;
- La possibilité de développer de nouvelles fonctionnalités ;

2. Conduite et planification du projet

2.1 Méthodologie de travail

2.1.1 Méthode agile

La méthode agile est une méthode de gestion et de développement de projets ou programmes informatiques, elle vise à satisfaire les besoins du client au terme du contrat de développement, elle se base sur l'itératif et l'incrémental.

La première méthode agile fut apparue en 1991 sous le nom de RAD (Développement Rapide d'Application).

De nos jours, il existe une variété de méthodes agiles dont les plus courantes sont : Scrum, Extrême Programming (XP), Rationnal Unified Process (RUP) et Cristal Clear.

-Scrum est la méthode agile la plus populaire. Bien que publiée en 2001 par Ken Schwber, elle est caractérisée par des itérations assez courtes et un formalisme court.

-Extrême Programming (XP) est une méthode agile qui consiste à réduire les coûts de changement. Elle met l'accent sur la revue de code, les tests, la conception continue et la simplicité.

-Rationnal Unified Process (RUP) est la moins agile parmi les autres méthodes. C'est un mélange des pratiques issues des méthodes traditionnelles et des méthodes agiles. Son principe est de parcourir le sigle de vie durant une itération.[2]

La méthode agile donc demeure un outil irremplaçable dans la réalisation de programmes informatiques. Elle prône quatre principales valeurs.

La communication en équipe est l'une des toutes premières valeurs. En effet, il est très important qu'une équipe de développeurs soit bien soudée et communique parfaitement, le changement est aussi l'une des valeurs de la méthode agile. Pour mener à bien son travail, le prestataire doit être ouvert à toute modification de projet en cours, ce dernier peut retirer ou ajouter de nouveaux éléments. En outre, la collaboration est une autre valeur clé, les développeurs et leurs clients doivent beaucoup collaborer, cela permettra aux développeurs de répondre efficacement aux besoins de leur clientèle et pour que le produit livré au client soit opérationnel et fonctionnel.

Au-delà de ses valeurs, la méthode agile fait l'objet de douze principes. Ils se résument comme suit :

- La satisfaction du client est le principe le plus prioritaire ;
- Les changements doivent procurer un avantage concurrentiel.
- Le client doit recevoir le plus régulièrement possible les versions opérationnelles du produit ;

- Les équipes de travail doivent collaborer constamment ensemble de même qu'avec leurs clients ;
- Le projet ne peut se faire qu'avec des personnes soudées et motivées ;
- Les conversations doivent se faire face à face pour une plus simple et efficace transmission des informations ;
- L'avancement du projet doit être réalisé à l'aide d'un logiciel opérationnel ;
- Le projet doit être construit autour d'un rythme soutenable et constant ;
- L'équipe doit travailler efficacement pour fournir une bonne conception ;
- Le volume de travail sans intérêts doit être au maximum réduit ;
- Les équipes auto-organisées doivent être encouragées de façon continue ;
- L'équipe doit avoir une réflexion constante afin d'améliorer son efficacité.

2.1.2 Méthode «Scrum »

La réalisation de notre projet s'est appuyée sur la méthodologie agile : SCRUM, puisqu'elle est particulièrement adaptée pour les projets ne répondant pas à un périmètre strict, décomposables en modules et pour lesquels le client souhaite un suivi, une visibilité, un respect des délais et une implication, ce qui est très important dans le cycle de développement.

En général, les méthodes de développement agile sont des méthodes axées sur les personnes, les objectifs et la satisfaction du client en s'appuyant sur la réalisation d'un logiciel entièrement fonctionnel tout au long de sa fabrication.

Le principe de la méthodologie SCRUM est de développer un logiciel de manière incrémentale en maintenant une liste totalement transparente des demandes d'évolutions ou de corrections à implémenter (backlog), avec des livraisons très fréquentes. Le client reçoit un logiciel possédant toujours plus de fonctionnalités et en parfait état de fonctionnement. De ce fait, la méthode s'appuie sur des développements itératifs. [3][4]

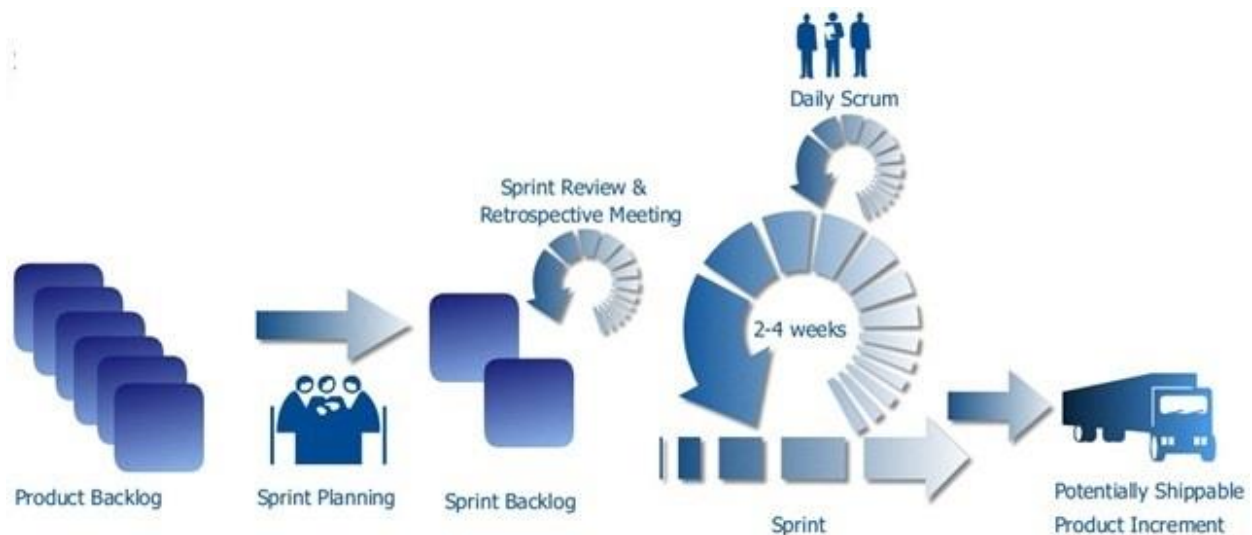


Figure 2: Méthode agile SCRUM

Le schéma montre la production d'une version « release » du logiciel, qui se déroule généralement en quelques mois. Les fonctionnalités souhaitées sont collectées dans le backlog de produit et classées par priorité. C'est le directeur de produit (product owner) qui est responsable de la gestion de ce backlog. Dans la majorité des projets, le Directeur du produit est le responsable de l'équipe projet.

La « release » est produite par une série d'itérations de 2 à 4 semaines appelées des sprints. Le contenu d'un sprint est défini par l'équipe, avec le Directeur de produit, en tenant compte des priorités et de la capacité de l'équipe. A partir de ce contenu, l'équipe identifie les tâches nécessaires et s'engage pour réaliser les fonctionnalités sélectionnées pour le sprint.

Pendant un sprint, des points de contrôle sur le déroulement des tâches sont effectués lors des mêlées quotidiennes (scrums), qui sont des réunions qui se tiennent chaque jour entre tous les membres de l'équipe, pendant 15 minutes environ, pour répondre aux 3 questions suivantes:

Qu'ai-je fait hier ?

Que vais-je faire aujourd'hui ?

Y a-t-il un obstacle gênant aujourd'hui ?

Cela permet au ScrumMaster, animateur chargé de faire appliquer le Scrum et véritable facilitateur sur le projet, de déterminer l'avancement par rapport aux engagements et d'appliquer, avec l'équipe, des ajustements pour assurer le succès du sprint. Il veille à ce que

chacun puisse travailler au maximum de ses capacités en éliminant les obstacles et en protégeant l'équipe des perturbations extérieures.

A la fin de chaque sprint, l'équipe produit un logiciel partiel (un incrément) qui fonctionne. Cet incrément du produit est potentiellement livrable et son évaluation permet d'ajuster le backlog pour le sprint suivant.

Chaque sprint regroupait les phases suivantes :

- Spécifications fonctionnelles ;
- Conception détaillée ;
- Codage ;
- Tests unitaires (recette).

Les réunions quotidiennes s'effectuaient le matin lors de l'arrivée dans les locaux de l'entreprise. La réunion est intéressante car elle permet d'avoir de réels objectifs quotidiens et de s'y tenir. D'autre part, les discussions post-réunion qui concernent les problèmes rencontrés sont très enrichissantes et permettent de progresser plus rapidement.

Ainsi, nous avons pris conscience de la flexibilité et l'adaptation de cette méthode aux changements et à l'évolution des besoins. De plus, elle permet à l'organisme d'être satisfait, ce qui permet à l'équipe de travailler dans un environnement adéquat pour que le produit corresponde mieux aux attentes de l'organisme.

2.2 Planification de projet

La planification consiste à prévoir le déroulement du projet tout au long des phases constituant le cycle de vie prévu. C'est ainsi qu'on peut diviser le projet en quatre grandes parties :

- **Description et l'analyse des besoins spécifiques** : les objectifs de cette phase sont de bien cerner le sujet de délimiter le périmètre du projet au niveau fonctionnel.
- **La modélisation et la conception** : cette partie a pour objectif d'élaborer des modèles détaillés de l'architecture du système à partir du modèle obtenu lors de l'étape d'analyse des besoins. Elle vise également à réduire la complexité du système.
- **La phase de développement et de test** : L'objectif de cette phase est de justifier le choix de la technologie utilisée, par la suite, il décrit les différents outils de développement choisis pour l'élaboration pratique du projet et enfin, on expose le travail réalisé à travers les captures écrans des interfaces de l'application réalisée.

4.2.1 Diagramme de Gantt

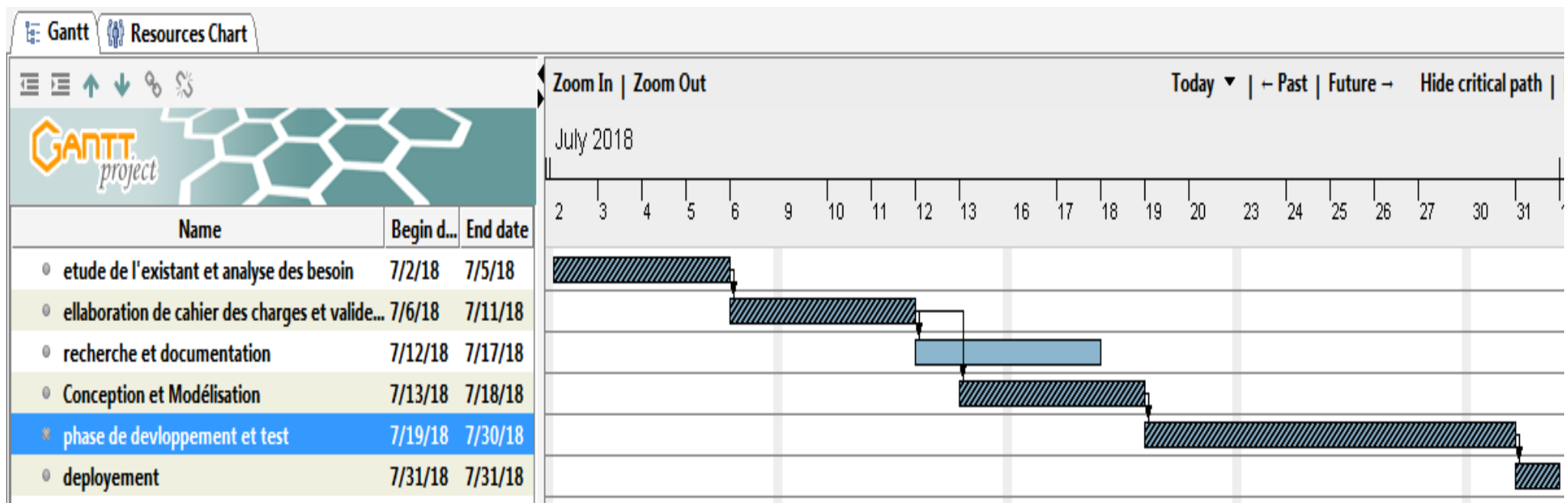


Figure 1. 1: Diagramme de Gantt

Conclusion

Dans ce chapitre on a décrit le contexte général dans lequel s'inscrit notre stage de fin d'année. Au début, nous avons présenté l'entreprise d'accueil USMBA Présidence. Puis, on a déterminé la problématique et les objectifs du projet qui se résument à la mise en place d'un système de gestion des adresses IP public. Après la description et l'élaboration du cahier des charges de l'université, on a cherché la solution adéquate pour assurer une bonne gestion. Puis, pour bien montrer les étapes du dit projet, on a établi un diagramme de Gantt.

Partie 2 :

Mise en œuvre

"Ne me parlez pas de vos efforts. Parlez-moi de vos résultats"

Ling, James Joseph

Chapitre 3 : Etude analytique des besoins

Ce Chapitre décrit la phase d'analyse et de spécification du projet. Nous présenterons un bilan de la procédure de système existant avec ses problèmes, Puis identifie les fonctionnalités auxquelles le système doit répondre, à partir de l'analyse des besoins.

Introduction

L'élaboration du cahier des charges est une phase primordiale. Elle est le point de passage obligatoire qui matérialise le premier contact des concepteurs avec les utilisateurs. La connaissance de la situation actuelle de l'organisation est indispensable pour pouvoir porter un jugement correct, et avoir toutes les informations nécessaires afin d'établir une bonne conception.

1. Système actuel : présentation des défaillances

2.1 Présentation du système de gestion actuel :

Le personnel de service informatique gère (interne) d'une manière manuelle et lente en faisant les opérations suivantes :

- **Recherche des adresses IP** : chaque adresse a des informations sur les services, l'adresse de serveur, l'établissement sur un fichier Excel qui fait l'objet de bases des données.
- **La gestion des réservations** : chaque adresse IP peut être réservée par un établissement mais par la demande d'un personnel de service informatique le personnel vérifie à l'aide d'Excel les adresses et peut-il réserver ou pas.

2.2 Défaillances du système traditionnel :

Le système utilisé présente actuellement des défaillances sur le niveau technique et sur le niveau fonctionnel.

a) Défaillances fonctionnelles :

Le système ne répond pas à certaines exigences telles que :

La rapidité : L'opération de la réservation se déroule d'une manière manuelle, l'employé cherche une adresse libre s'il y a plusieurs demandes de réservation qui sont avant lui, pour chercher les adresses disponibles. Cette opération perd un temps

précieux pour les personnels de service informatique et exige beaucoup de patience et d'effort, ce processus est très lent !

Efficacité :

- Ce système ne donne pas une visibilité pour les responsables afin de faire des recherches ou libérer des adresses qui non pas utilisé.
- Pas de partage de l'information au temps réel.

b) Défaillances techniques :

Les défaillances techniques sont nombreuses et posent beaucoup de problèmes aux différents intervenants dans le système, à savoir :

- Les ouvrages sont enregistrés sur plusieurs fichiers Excel qui fait l'objet des bases des données et l'Excel reste un outil limité pour une gestion rapide des documents
- Tâches manuelles et répétitives avec beaucoup de paperasses
- Sécurité des données : il y a un grand risque que les fichiers Excel soient perdus.

2.3 Analyse des besoins et cahier des charges :

Le système à concevoir doit répondre aux besoins fonctionnels suivants :

- **Réservation** : l'employé doit garder une information sur les adresses IP réservé.
- **Base de données** : avoir une base de données centralisée et partagé avec tous les personnels de service informatique dans un serveur qui regroupe tous la plage des adresses IP de la Présidence.
- **Recherche** : Assurance d'une gestion efficace, rapide et parfaite de recherche afin de fournir le maximum d'informations sécurisés et correctes.
- **Classification des adresses** : classer les adresses par établissement, service, date de réservation afin de faciliter la consultation et l'organisation.

2.4 Etude fonctionnel :

Pour mener à bien notre projet, il est nécessaire de faire une analyse préliminaire des besoins et de lister les différentes fonctionnalités à mettre en œuvre.

Fonctionnalités	
Gestion des adresses IP	Gestion des rapports de : <input type="checkbox"/> Service <input type="checkbox"/> Etablissements <input type="checkbox"/> la date <input type="checkbox"/> NAT
Gestion des utilisateurs	<input type="checkbox"/> Gestion d'accès (sécurise) <input type="checkbox"/> Gestion des privilèges <input type="checkbox"/> authentification

Tableau 1: Les fonctionnalités à mettre en œuvre dans notre système

2. MVC : Modèle vue contrôleur

MVC est une architecture de modèles utilisée en génie logiciel, dans les applications complexes qui présentent des lots de données aux utilisateurs, nous souhaitons souvent séparer les données (modèle) et l'interface utilisateur (vue), de sorte que les changements à l'interface utilisateur n'affectent pas le traitement des données, et que les données peuvent être réorganisées sans changer l'interface utilisateur. Le MVC résout ce genre de problème en découplant l'accès des données et la logique des applications de la présentation des données et de l'interaction utilisateur, en introduisant un composant intermédiaire : « le contrôleur ». [5]

Dans notre application, nous pouvons appliquer cette sémantique de Model View Controller avec :

- ☐ Model : les modèles sont les objets déclarés dans base des données. Ils sont également des tables dans SQL.
- ☐ View : les vues sont définies en fichiers FXML.
- ☐ Controller : le contrôleur est les classes qui contrôlent.

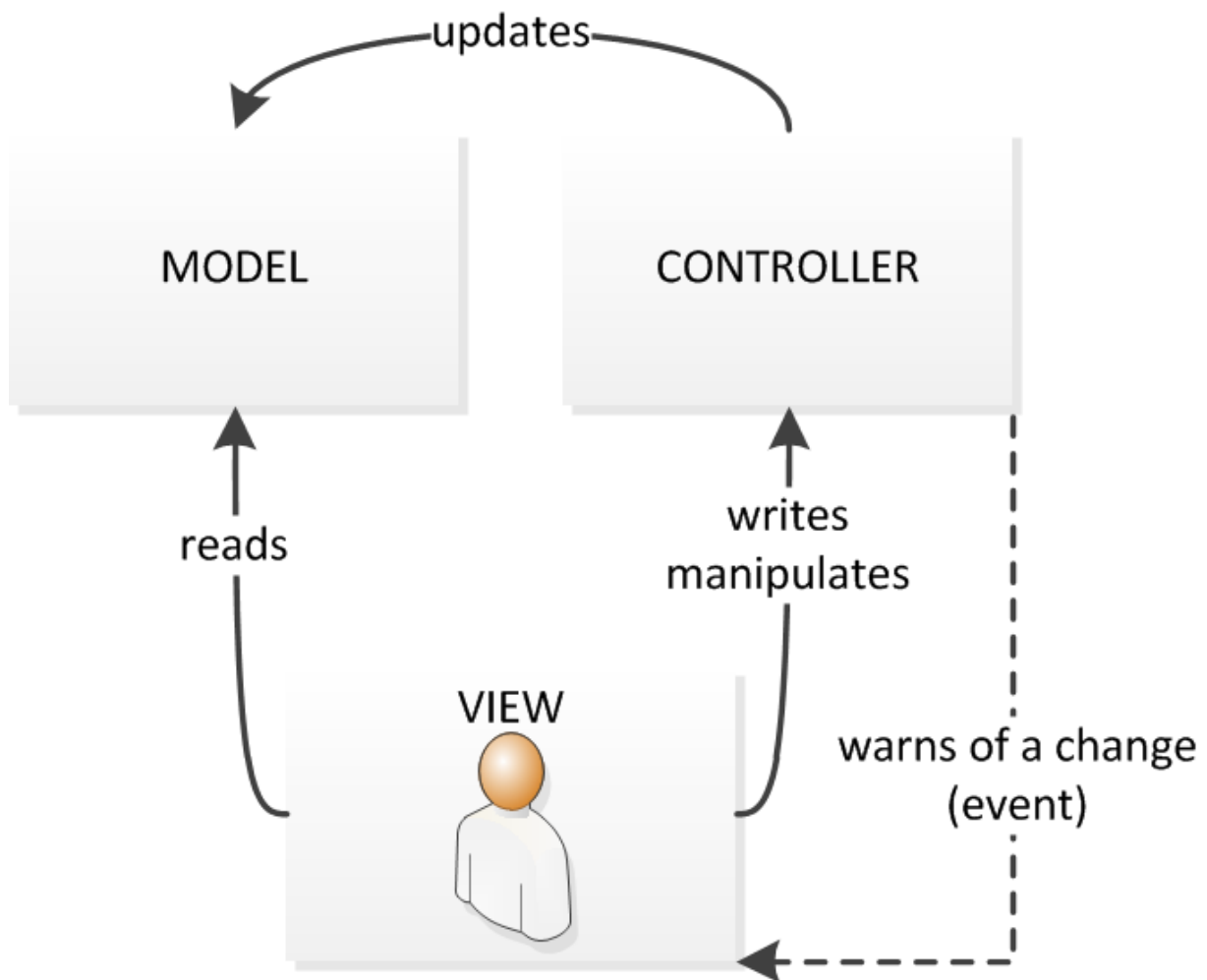


Figure 3 :Modèle vue contrôleur

Conclusion

Après avoir étudié avec délicatesse les failles du système actuel et élaboré le cahier des charges à partir de l'analyse des besoins, nous allons entamer la conception et la modélisation de l'application pour traduire tous ce qui été décrit au préalable en un langage intermédiaire entre les besoins fonctionnels et un langage de programmation compréhensible par l'ordinateur.

Chapitre 4 : Etude fonctionnelle et conception du système

***Dans ce chapitre, nous allons décrire la méthodologie de
conception et les besoins de l'organisme, par la suite nous
allons les analyser en utilisant le formalisme UML.***

1. Analyse et méthodologie de conception

1.1 Analyse du projet

La complexité croissante des systèmes informatiques a conduit les concepteurs à s'intéresser aux méthodes. On a comptabilisé en 1994 jusqu'à 50 méthodes. Chaque méthode se définit par une notation et un processus spécifique. UML a ouvert le terrain en fusionnant la notation. Il reste cependant à définir le processus pour réellement capitaliser des règles dans le domaine du développement logiciel. Les groupes de travail UML ont donc travaillé à unifier non pas les processus, mais plus exactement les meilleures pratiques de développement objet. Ces processus se distingueront par le générique «UNIFIED PROCESS ».

Un processus définit une séquence d'étapes, en partie ordonné, qui concourt à l'obtention d'un système logiciel ou à l'évolution d'un système existant. Pour produire des logiciels de qualité, qui répondent aux besoins des utilisateurs dans des temps et des coûts prévisibles. On découpe le processus en deux axes :

- L'axe de développement technique, qui concentre principalement sur la qualité de production.
- L'axe de gestion du développement, qui permet la mesure et la prévision des coûts et des délais.

1.2 Choix de la méthodologie de conception

Avant d'adopter une méthode, il faut d'abord faire une comparaison entre les différentes méthodes existantes, voir les points forts et les points faibles de chacune, puis déterminer celle qui va mieux dans le contexte du projet. Ci-dessous un tableau qui résume cette comparaison.

Méthode	Description	Points forts	Points faibles
Rational Unified Process (RUP)	-Méthodologie centrée sur l'architecture et couplée aux diagrammes UML -Concerne des projets de +10 personnes -Processus complet assisté par des outils exhaustifs	-Itératif -Spécifie le dialogue entre les différents intervenants du projet : les livrables, plannings et prototypes... -Propose des modèles de documents, et des canevas pour des projets types -Rôles bien définis, modélisation	-Coûteux à personnaliser -Très axé processus, au détriment du développement -Lourd, largement étendu, il peut être difficile à mettre en œuvre de façon spécifique -Convient pour les grands projets qui

			génèrent beaucoup de documentation
<i>eXtreme Programming (XP)</i>	<ul style="list-style-type: none"> -Développement guidé par les besoins du client -Equipes réduites, centrées sur les développeurs 	<ul style="list-style-type: none"> -Itératif -Simple à mettre en œuvre -Laisse une large place aux aspects techniques -Builds journaliers -Amélioration constante adaptation aux modifications 	<ul style="list-style-type: none"> -Ne couvre pas les phases en amont et en aval du développement -Assez flou dans sa mise en œuvre : quels intervenants ? Quels livrables ? -Focalisé sur l'aspect individuel du développement, au détriment d'une vue globale et des pratiques de management ou de formalisation.
<i>Two Track Unified Process (2TUP)</i>	<ul style="list-style-type: none"> -Articulé autour de l'architecture -Cycle de développement en Y -Convient aux projets de toutes tailles 	<ul style="list-style-type: none"> -Itératif, laisse une large partie à la technologie et à la gestion du risque -Définit les profils des intervenants, les livrables, les prototypes 	<ul style="list-style-type: none"> -Superficiel sur les phases en amont et en aval du développement -Aucune proposition de document type

Tableau 2: Comparaison des méthodes de développement

Pour atteindre les objectifs, nous avons suivi la méthode 2TUP (2Track Unified Process), qui sera plus détaillée ci-dessous.

1.3 Processus 2TUP

2TUP est un processus UP (processus unifié). Le processus 2TUP apporte une réponse aux contraintes de changement continu imposées aux systèmes d'information de l'entreprise. Dans ce sens, il renforce le contrôle sur les capacités d'évolution et de correction d'un tel système. «2Track» signifie littéralement que le processus suit deux chemins. Il s'agit des chemins « fonctionnels » et « d'architecture technique », qui correspondent aux deux axes de changements imposées au système informatique. [11]

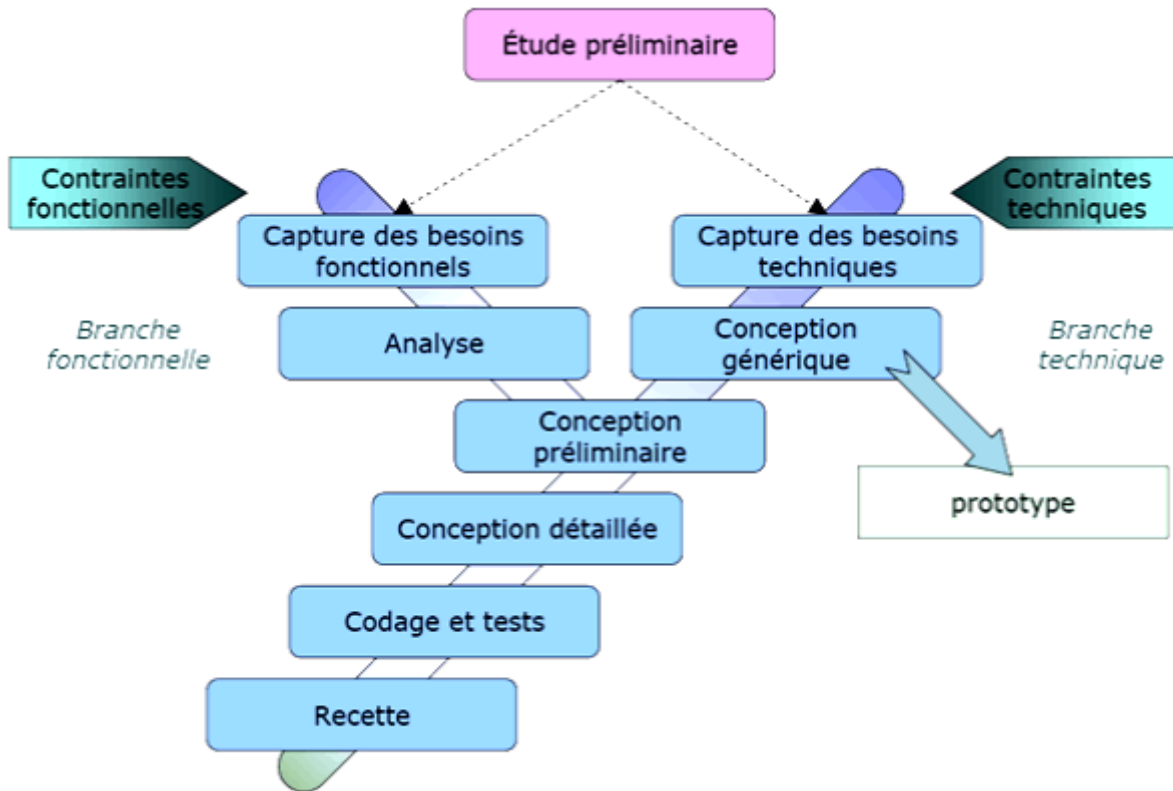


Figure 4 : Cycle de développement en Y

L'axe fonctionnel comporte :

- Capture des besoins fonctionnels, qui produit un modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie au plus tôt le risque de produire un système inadapté aux utilisateurs.
- Analyse, qui consiste à étudier précisément la spécification fonctionnelle de manière à obtenir une idée de ce que va réaliser le système en terme de métier.

L'axe technique, quant à lui, comporte :

- Capture des besoins techniques, qui recense toutes les contraintes et les choix Dimensionnant la conception du système.
- Conception générique, qui définit ensuite les composants nécessaires à la conception de l'architecture technique. Elle a pour objectif d'uniformiser et de réutiliser les mêmes mécanismes pour tout le système. L'architecture technique construit le squelette du système informatique et écarte la plupart des risques du niveau technique.

- Conception préliminaire, qui représente une étape délicate, car elle intègre le modèle d'analyse fonctionnelle dans l'architecture technique de manière à tracer la cartographie des composants du système à développer.
- Conception détaillée, qui étudie ensuite comment réaliser chaque composant.
- Codage, qui produit ses composants et teste au fur et à mesure les unités de code réalisées.
- Recette, qui consiste enfin à valider les fonctionnalités du système développé.

1.4 UML (Unified Modeling Language)

UML se définit comme un langage de modélisation graphique et textuel destiné à comprendre et décrire des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue.

UML unifie est à la fois les notations et les concepts orientés objet. Il ne s'agit pas d'une simple notation, mais les concepts transmis par un diagramme ont une sémantique précise et sont porteurs de sens au même titre que les mots d'un langage. UML a une dimension symbolique et ouvre une nouvelle voie d'échange de visions systémiques précises, Ce langage est certes issu du développement logiciel mais pourrait être appliqué à toute science fondée sur la description d'un système. Dans l'immédiat, UML intéresse fortement les spécialistes de l'ingénierie système. [12][13][14]

UML fournit un moyen astucieux permettant de représenter diverses projections d'une même représentation grâce aux vues.

Une vue est constituée d'un ou plusieurs diagrammes. On distingue trois types de vues :

- **Vue fonctionnelle** : donnant une vue globale de notre projet.
 - **Diagrammes de cas d'utilisation** : identifient les utilisateurs du système (acteurs) et leurs interactions avec le système.
- **La vue statique** : permettant de représenter le système physiquement :
 - **Diagrammes de classes** : représentent des collections d'éléments de modélisation statique (classes, paquetages...), qui montrent la structure d'un modèle.
 - **Diagrammes d'objets** : ces diagrammes montrent des objets (instances classes dans un état particulier) et des liens (relations sémantiques) entre objets.

- **Diagrammes de composants** : permettent de décrire l'architecture physique statique d'une application en termes de modules : fichiers sources, librairie exécutables, etc.
- **La vue dynamique** : montrant le fonctionnement du système :
 - **Diagrammes de collaboration** : montrent des interactions entre objet (instances de classes et acteurs).
 - **Diagrammes de séquence** : permettent de représenter des collaborations en objets selon un point de vue temporel, on y met l'accent sur la chronologie (envois de messages).
 - **Diagrammes d'états-transitions** : permettent de décrire les changements d'états d'un objet ou d'un composant, en réponse aux interactions avec d'autres objets/composants ou avec des acteurs.
 - **Diagrammes d'activités** : (une variante des diagrammes d'états-transitions) servent à représenter graphiquement le comportement d'une méthode ou déroulement d'un cas d'utilisation.

La conception de notre projet a été élaborée en suivant la démarche suivante :

- L'élaboration des diagrammes de cas d'utilisation. Cette étape a été réalisée suite à la spécification fonctionnelle de l'application.
- Dresser les diagrammes de séquences pour mettre en évidence les interactions entre les différents objets du système.
- Recensement des classes candidates et élaboration du diagramme des classes.
- Elaborer le diagramme d'activité pour montrer les différentes activités pour les méthodes des objets.

2. Conception détaillé du projet

2.1 Cas d'utilisation

2.1.1 Identification des processus

- Processus lié à l'administration du système
- Processus de gestion des adresses IP

2.1.2 Administration du système

L'administration complète du système est assurée par l'administrateur. Celui-ci se charge de contrôler l'accès des utilisateurs et de gérer l'ensemble des fonctionnalités du système. Ceci est modélisé par le diagramme de cas d'utilisation suivant :

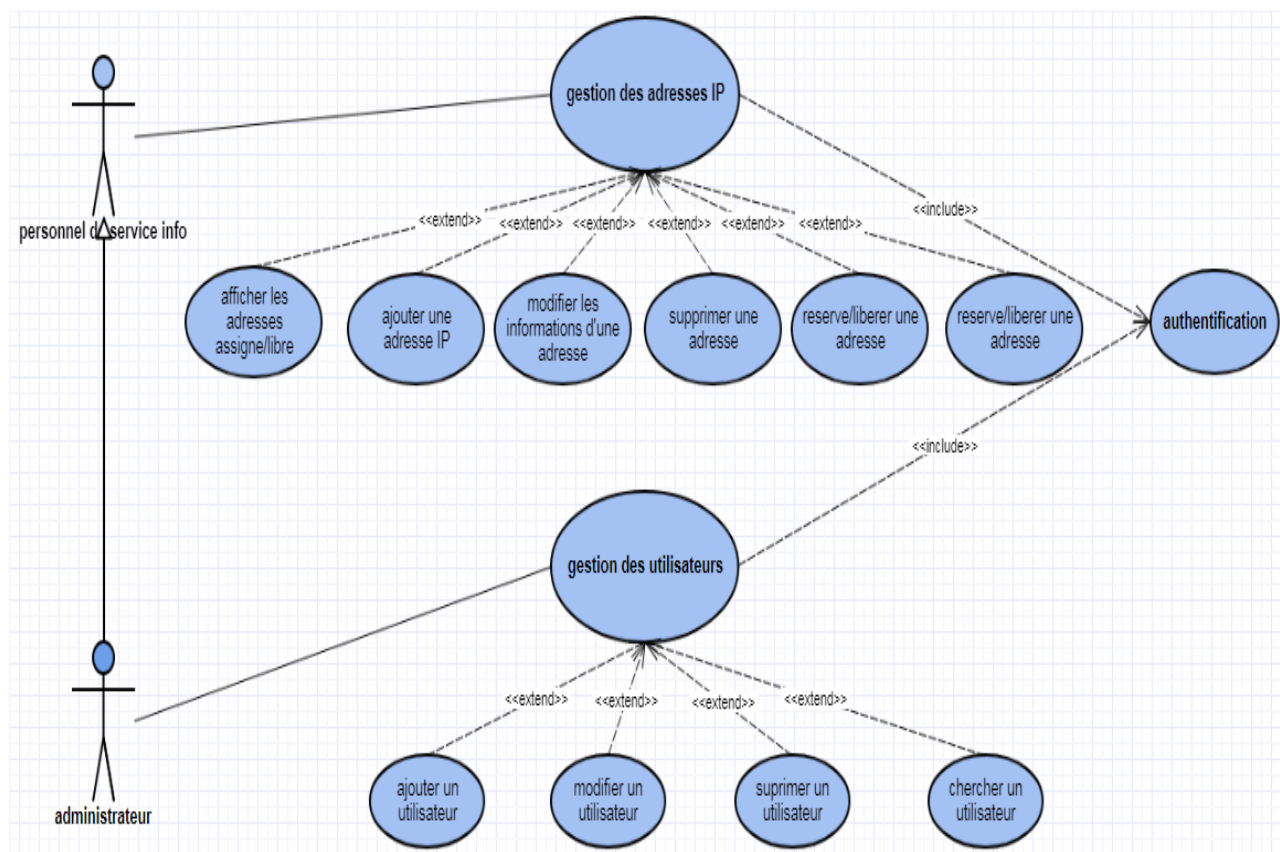


Figure 5: Diagramme de cas d'utilisation du système

- Description des cas d'utilisation d'administration du système

Cas d'utilisation	Description
Authentification	avant de se connecter au système, chaque acteur doit être identifié par un login et un mot de passe afin d'avoir les permissions d'accès au système. Le système vérifie l'existence de l'utilisateur et la validité de son mot de passe, sinon il ouvre une session et redemande la saisie à nouveau.
Gestion des utilisateurs	permet à chaque utilisateur du système d'avoir son compte qui comporte l'ensemble des processus métiers dont lesquels il intervient. Comme, il attribue les droits d'accès à l'utilisateur afin de garantir la fiabilité du système.
Gestion des adresses IP	permet à l'administrateur de gérer la plage des adresses qui ont donné à la présidence et partager sur les établissements en cas de besoin.

Tableau 3: Description des cas d'utilisation d'administration du système

2.2 Processus métiers

Un processus métier livre une spécification complète des besoins, issu des cas d'utilisateurs, et les structures sous une forme qui facilite la compréhension (scénarios), la préparation, la modification et la maintenance du futur système.

2.2.1 Processus métier de la gestion des adresses IP :

Afin de comprendre le processus métier de gestion des adresses et d'éclaircir la relation entre les différentes phases d'une action de cette gestion, on a opté pour le diagramme suivant :

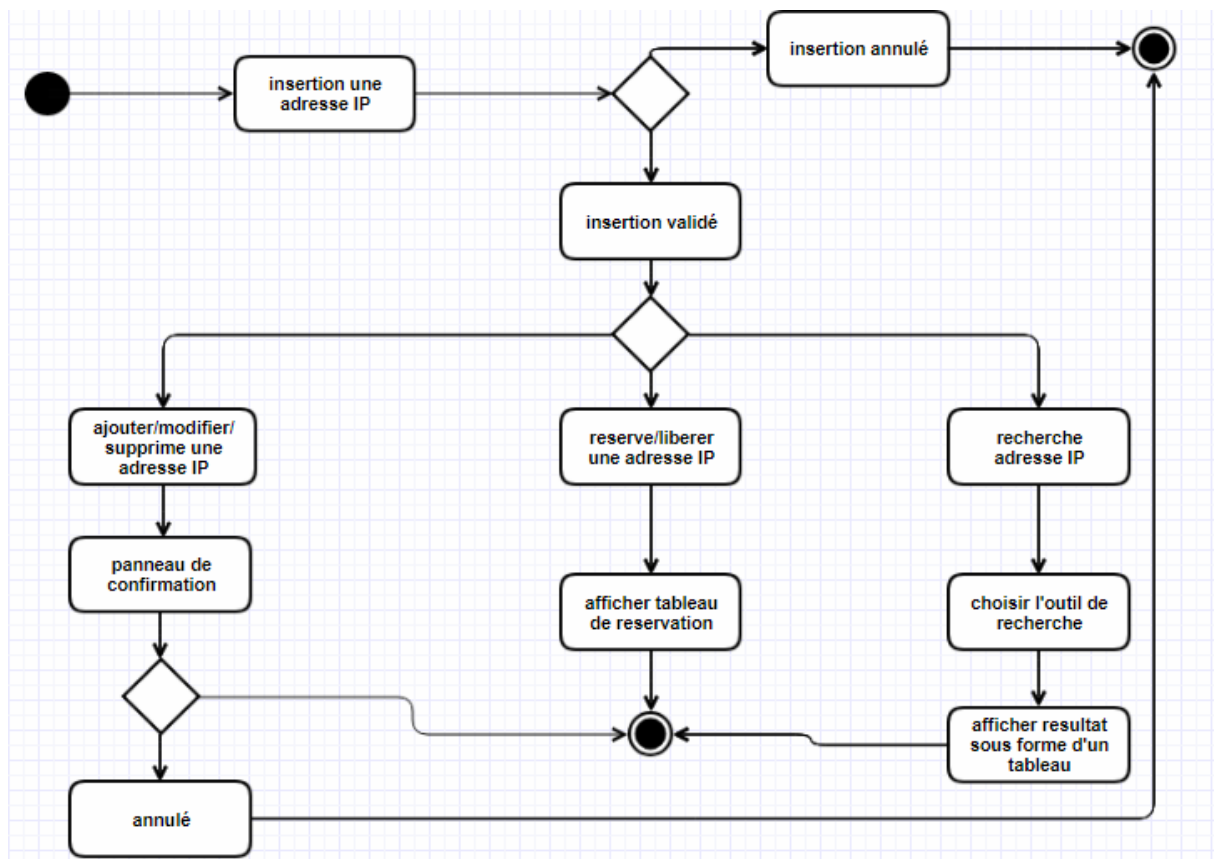


Figure 6 : Processus métier de gestion des adresses IP

Une fois le fonctionnaire se connecte il est invité à insère les adresses voulu, puis consulter la liste, ajouter /modifier /supprimer ou réserver des adresse IP, ou même de rechercher les nombres des adresses IP déjà inséré, en passant par la consultation des listes, il peut trier les données selon le critère voulu.

2.3 Diagrammes de séquence

Le diagramme de séquence permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets.

Les concepts de base utilisés dans ce diagramme sont les suivant :

2.3.1 Diagramme de séquence « Authentification »

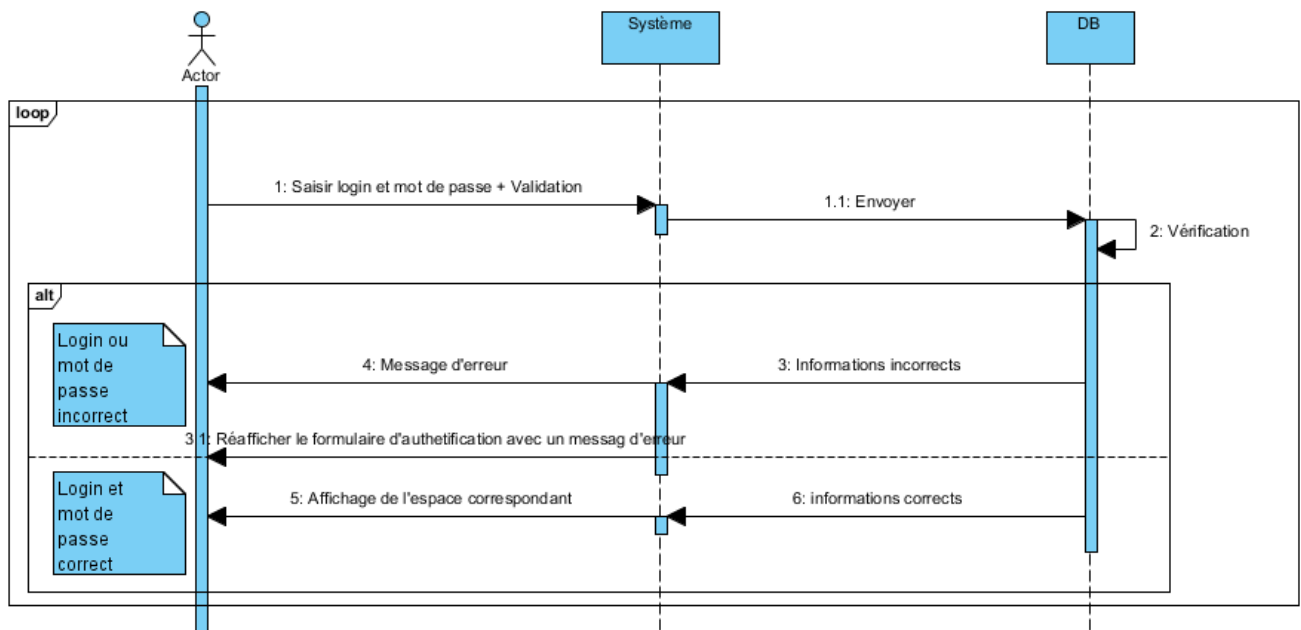


Figure 4. 1: Diagramme de séquence "Authentification"

Les scénarios :

Dans le processus d'authentification, chaque utilisateur doit interagir avec le système :

- L'utilisateur demande de se connecter au système.
- Le système affiche l'interface du login et mot de passe.
- L'utilisateur insert son login et mot de passe.
 - S'ils sont incorrects ➡ le système affiche une fenêtre d'erreur et réaffiche l'interface du login et mot de passe.
 - S'ils sont corrects ➡ le système affiche l'espace correspondant.

2.3.2 Diagramme de séquence «ajouter une adresse IP »

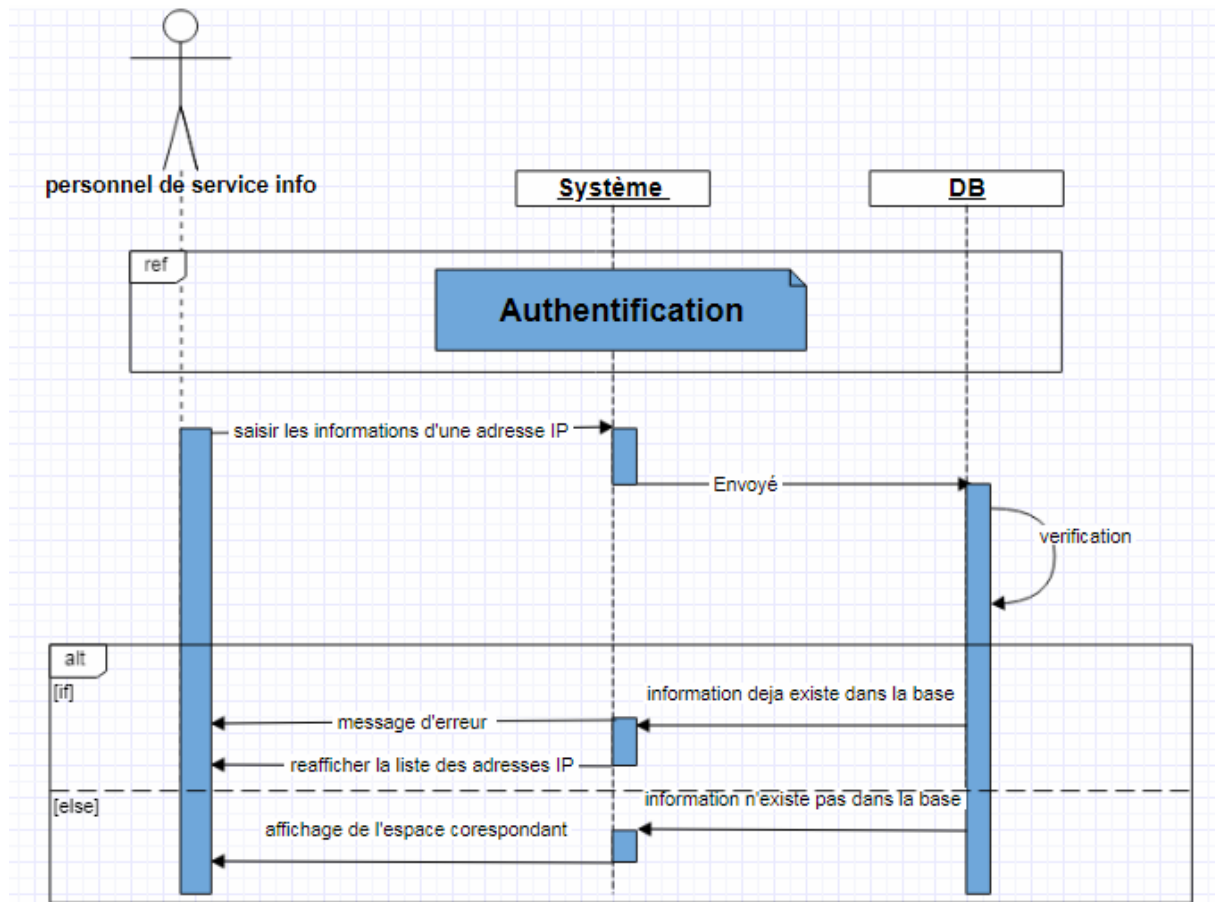


Figure 7: Diagramme de séquence "Scolarité"

Les scénarios :

Dans le processus d'ajout des adresses IP on a l'acteur personnel de service informatique qui interagit avec le système.

- ☐ L'acteur demande de se connecté à la l'application.
- ☐ L'acteur insère les données
- ☐ Le système envoi les données à la base de données.
- ☐ La base de données assure que les enregistrements sont bien enregistrés.
- ☐ L'acteur demande de consulter la liste des données enregistré
- ☐ Le système récupère la liste.

2.4 Diagrammes de classe

2.4.1 Présentation du diagramme de classe

Le diagramme de classes est le point central dans un développement orienté objet, en analyse, il a pour objectif de décrire la structure des entités manipulées par les utilisateurs. En conception, le diagramme de classes représente la structure d'un code orienté objet ou, à un niveau de détail plus important, les modules du langage de développement.

De plus, ce diagramme possède deux intérêts principaux. Il permet, à la fois la modélisation d'une base de données et la génération automatique de code pour plusieurs langages de programmation, qui concrétisent la conception. Le diagramme de classe ouvre la voie à une programmation complètement automatisée.

Ainsi, il a toujours été le diagramme le plus important dans toutes les méthodes orientées objet, à savoir, les outils de génération automatique de code l'utilisent en priorité.

2.4.2 Diagramme de classe du module « Scolarité »

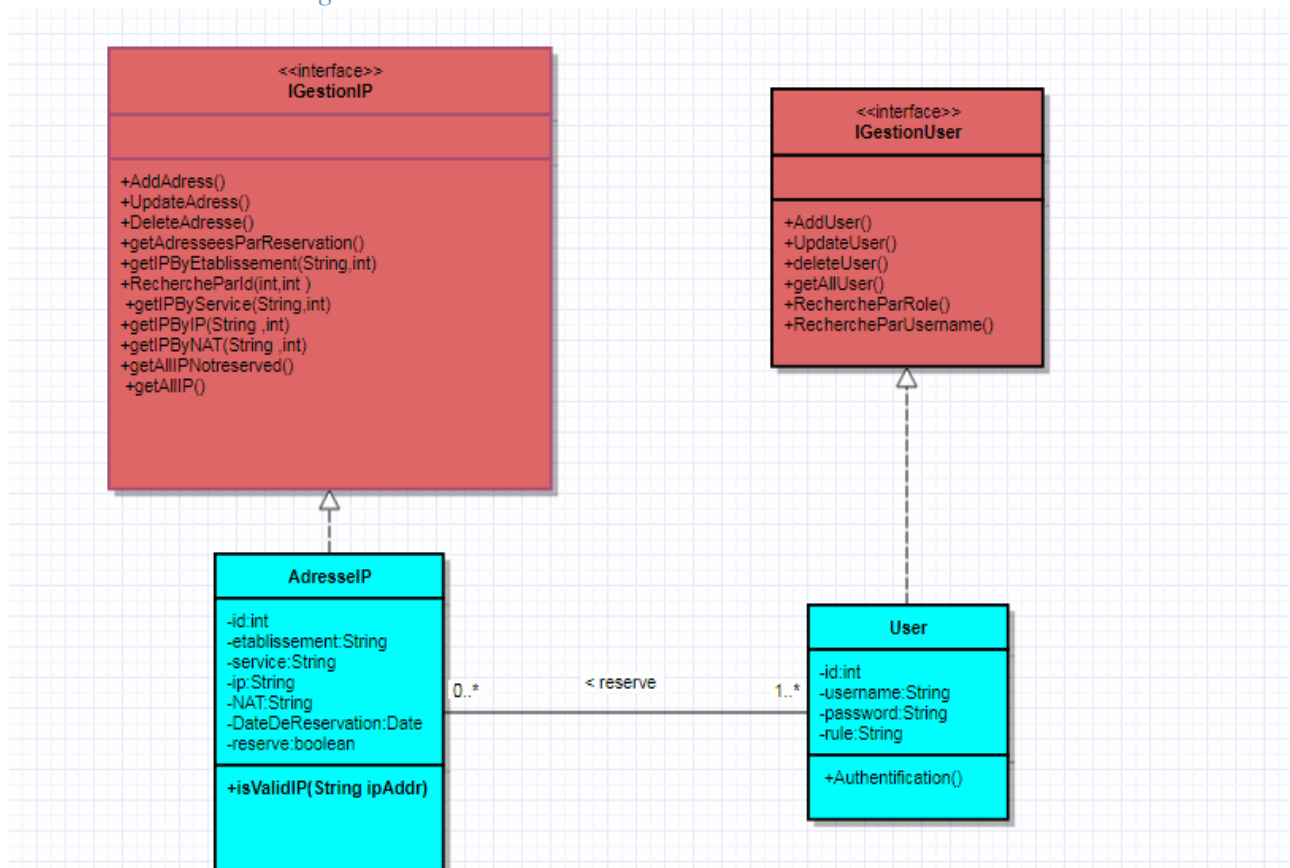


Figure 8: Diagramme de classe du système.

Ce diagramme représente les classes nécessaires pour assurer un bon fonctionnement du système.

Un utilisateur de service informatique peut réserver des adresses IP, L'adresse IP se compose

d'un établissement, un Service, l'adresse réserver, le NAT, date de réservation il peut aussi effectuer des opérations pour chercher les adresses IP par un mot clé.

2.5. Les composants de notre système :

- **La couche présentation** : joue le rôle de la vue dans le model MVC, il décrit tous ce qui est interface, fenêtres et les actions. Dans notre cas les vues sont développées en FXML
- **La couche logique métier** : joue le rôle de contrôleur de model MVC il permet de décrire les objets les classes du projet via le langage de programmation JAVA.
- **La couche persistance** : concerne la persistance des données dans un serveur SQLite.

Conclusion

La conception met en œuvre un ensemble d'activités, précise l'implémentation technique de l'application, elle consiste en la fusion de la spécification détaillée et de la conception générale, pour déterminer « comment faire » le « quoi faire » dans le détail. Nous trouvons le diagramme de classe et les diagrammes de séquences qui détaillent les interactions entre les composants du logiciel pour les scénarios les plus complexes.

Ce chapitre consacré à la conception de notre système. Tout d'abord, nous avons commencé par la capture des besoins fonctionnels, où nous avons représenté l'ensemble des séquences d'actions, en identifiant les cas d'utilisations de tous les acteurs du système, suivi d'un diagramme de cas d'utilisation pour chaque cas d'utilisation.

Ensuite, nous avons représenté les diagrammes de processus de chaque métier, et recensé la description graphique des cas d'utilisation, en réalisant des diagrammes de séquences décrivant les scénarios nominaux des cas d'utilisations essentiels.

La capture des spécifications techniques et logicielles du système existant, nous a permis de définir le cas d'utilisation technique.

En fin, nous avons présenté les concepts de base des diagrammes de classe et les règles de modélisation.

Chapitre 5 : Technologies utilisées

Dans ce chapitre, nous allons présenter les différents outils et technologies utilisés pour la réalisation de ce projet.

Définition

Durant la réalisation de l'application nous avons utilisé plusieurs outils et technologies.

Le présent chapitre décrit chacune des technologies utilisées en commençant par une vision globale sur l'architecture technique de l'application.

1. Java

Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que Unix, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications, mais qui ont l'inconvénient d'être plus lourd à l'exécution (en mémoire et en temps processeur) à cause de sa machine virtuelle. Pour cela, divers plateformes et Framework associés visent à guider, sinon garantir, cette portabilité des applications développées en Java.

Il est conçu pour être orienté objet, il ne dispose d'aucun outils permettant de se livrer à la programmation fonctionnelle ou impérative; c'est d'ailleurs une des raisons qui lui vaut son appellation de « langage agile ».

Parmi les autres raisons, citons la rapidité de développement (qualité propre aux langages interprétés), la grande quantité de modules fournis dans la distribution de base ainsi que le nombre d'interfaces disponibles avec des bibliothèques écrites en C, C++ ou Python. Il est également apprécié pour la clarté de sa syntaxe. [6]

Versi on	Last upd ate	Dénomin ation JSE/JRE	Nom de code	Spécifica tions	JDK	Statut mai 2018	Période de mainten ance	Suppo rt étend u ²³
-------------	--------------------	-----------------------------	-------------------	--------------------	-----	--------------------	----------------------------------	---

1.0	1.0.2	Java 1.0	Oak	JSR 52	JDK 1.0.2	N'est plus soutenu de façon active	1996-2000	
1.1	8_16	Java 1.1		JSR 52	1.1.8_16	N'est plus soutenu de façon active	1997-2000	
1.2	2_01_7	J2SE 1.2	Playground	JSR 52	1.2.2_11	N'est plus soutenu de façon active	2000-2006	
1.3	1_29	J2SE 1.3	Kestrel	JSR 58	1.3.1_29	Obsolète	2000-2001	
1.4	2_30	J2SE 1.4	Merlin	JSR 59	1.4.2_30	Obsolète	2000-2008 ²⁴	
1.5	0_22 à 0_85	J2SE 5.0	Tiger	JSR 176	1.5.0_22	Obsolète, 5u51 à 5u85 uniquement disponibles avec un support Oracle spécifique ²⁵	2002-2009 ²⁴	Mai 2015
1.6	0_45 à 0_111	Java SE 6	Mustang	JSR 270	6u113	Obsolète, 6u51 à 6u111 uniquement disponibles avec un support Oracle spécifique ²⁵	2005-2013 ²⁴	Décembre 2018
1.7	0_79 à 0_80	Java SE 7	Dolphin	JSR 336	1.7.0_79	Stable, version 1.7.0_79 ²⁶	2011-2015 ²⁴	Juillet 2022
1.8	0_171	Java SE 8	Kenai ²⁷ puis Spider	JSR 337	1.8.0_171	Stable, version 1.8.0_171 ²⁸ pr	2014-sept2018	Juillet 2019

						opposée aux utilisateurs ²⁹		
9	9.0.4	Java SE 9	Umbrella	JSR379 ³⁰	9.0.4	Stable	2018-?	
10	10.0.1	Java SE 10		JSR383	10.0.1	supporté 6 mois avant prochaine sortie du JDK Java11-LTS	2018-?	
11				JSR384		<i>proposé aux seuls développeurs pour correction de bugs</i>		

Tableau 4 : Historique des versions de JAVA

2. FXML :

FXML est un langage informatique de **balisage générique basé sur XML** (eXtensible Markup Language). Il s'agit effectivement d'un langage permettant de mettre en forme des documents grâce à des balises (markup). [7] [8]

Il fournit une alternative pratique à la construction de tels graphiques dans le code procédural, et convient parfaitement à la définition de l'interface utilisateur d'une application JavaFX, car la structure hiérarchique d'un document XML est étroitement liée à la structure du graphe de scène JavaFX. Cependant, tout ce qui est créé ou implémenté dans FXML peut être exprimé directement avec JavaFX.

Contrairement à HTML, qui est considéré comme un langage défini et figé (avec un nombre de balises limité), FXML peut être considéré comme un métalangage permettant de définir d'autres langages, c'est-à-dire définir de nouvelles balises permettant de décrire la présentation d'un texte (Qui n'a jamais désiré une balise qui n'existait pas ?). La force de FXML réside dans sa capacité à pouvoir décrire n'importe quel domaine de données grâce à son extensibilité. Il va permettre de structurer, poser le vocabulaire et la syntaxe des données qu'il va contenir.

En réalité les balises XML décrivent le contenu plutôt que la présentation (contrairement à HTML). Ainsi, XML permet de séparer le contenu de la présentation, ce qui permet par exemple d'afficher un même document sur des applications ou des périphériques différents sans pour autant nécessiter de créer autant de versions du document que l'on nécessite de représentations.

3. SQL Lite

SQLite est une bibliothèque in-process qui implémente un moteur de base de données SQL transactionnel autonome, sans serveur et sans configuration. Le code pour SQLite est dans le domaine public et est donc gratuit pour toute utilisation, commerciale ou privée. SQLite est la base de données la plus largement déployée au monde avec plus d'applications que nous ne pouvons en compter, y compris plusieurs projets de grande envergure.

SQLite est un moteur de base de données SQL intégré. Contrairement à la plupart des autres bases de données SQL, SQLite ne possède pas de processus serveur distinct. SQLite lit et écrit directement sur des fichiers disques ordinaires. Une base de données SQL complète avec plusieurs tables, index, déclencheurs et vues est contenue dans un seul fichier de disque. Le format du fichier de base de données est multi-plateforme - vous pouvez librement copier une base de données entre des systèmes 32 bits et 64 bits ou entre des architectures big-endian et little-endian. Ces fonctionnalités font de SQLite un choix populaire en tant que format de fichier d'application. Les fichiers de base de données SQLite sont un format de stockage recommandé par la US Library of Congress. Pensez à SQLite non pas en remplacement d'Oracle mais en remplacement de fopen ().



SQLite est une bibliothèque compacte. Avec toutes les fonctionnalités activées, la taille de la bibliothèque peut être inférieure à 500 Ko, en fonction des paramètres d'optimisation de la plateforme cible et du compilateur. (Le code 64 bits est plus grand. Et certaines optimisations du compilateur telles que l'inclinaison des fonctions agressives et le déroulement des boucles peuvent entraîner une augmentation importante du code objet.) Il existe un compromis entre l'utilisation de la mémoire et la vitesse. SQLite fonctionne généralement plus vite, plus vous lui donnez de mémoire. Néanmoins, les performances sont généralement bonnes même dans les environnements à faible mémoire. Selon son utilisation, SQLite peut être plus rapide que les E/S directes du système de fichiers. [9]

4 Outils de travail

4.1 Outils de développement

	<p>Eclipse Luna IDE for Java EE Developers : Eclipse est un environnement de développement intégré (IDE). Il contient un espace de travail de base et un système de plug-in extensible pour la personnalisation de l'environnement.</p> <p>Écrit principalement en Java, Eclipse peut être utilisé pour développer des applications. Par le biais de divers plug-ins. Luna est la version de l'IDE sortie en 12 janvier 2014</p>
	<p>Sublime Text est un éditeur de texte générique codé en C++ et Python, disponible sur Windows, Mac et Linux. Le logiciel a été conçu tout d'abord comme une extension pour Vim, riche en fonctionnalités.</p> <p>Depuis la version 2.0, sortie le 26 juin 2012, l'éditeur prend en charge 44 langages de programmation majeurs, tandis que des plugins sont souvent disponibles pour les langages plus rares.</p>
	<p>DB Browser for SQLite est un outil visuel, open source de haute qualité permettant de créer, concevoir et éditer des fichiers de base de données compatibles avec SQLite. [10]</p> <p>C'est pour les utilisateurs et les développeurs qui souhaitent créer des bases de données, rechercher et modifier des données. Il utilise une interface de type tableur familière et vous n'avez pas besoin d'apprendre des commandes SQL compliquées.</p> <p>Des contrôles et des assistants sont disponibles pour les utilisateurs :</p> <ul style="list-style-type: none"> • Créer et compacter des fichiers de base de données • Créer, définir, modifier et supprimer des tableaux • Créer, définir et supprimer des index • Parcourir, éditer, ajouter et supprimer des enregistrements • Rechercher des notices • Importer et exporter des enregistrements sous forme de texte • Importer et exporter des tables depuis / vers des fichiers CSV • Importer et exporter des bases de données depuis / vers des fichiers de vidage SQL • Émettre des requêtes SQL et inspecter les résultats • Examiner un journal de toutes les commandes SQL émises par l'application

4.2 Conception et modélisation

	<p>Gantt Project : est un logiciel libre de gestion de projet écrit en Java, ce qui permet de l'utiliser sur divers système d'exploitation (Windows, Linux, MacOS); il permet d'éditer un diagramme de Gantt.</p>
	<p>Unified Modeling Language : UML est un langage de modélisation pour unifier les modèles utilisés dans les méthodes de développement. Il propose un ensemble de langages algorithmiques graphiques qui permettent de Réaliser les étapes de la conception. Ces langages sont particulièrement adaptés pour la programmation orientée objet.</p>

Chapitre 6 : Réalisation

Dans ce chapitre, nous allons présenter l'ensemble des fonctionnalités que nous avons développé sous forme de capture d'écran.

Introduction

Dans le chapitre précédent, la modélisation fonctionnelle, statique et dynamique établie par le biais du formalisme de modélisation UML, nous a permis de réaliser les différents diagrammes d'analyse permettant la mise en œuvre de l'application, par ailleurs, ce chapitre illustre la justification du choix de la technologie utilisée, par la suite, il décrit les différents outils de développement choisis pour l'élaboration pratique du projet et enfin, on expose le travail réalisé à travers les captures écrans des interfaces de l'application réalisée.

1. Choix de la technologie :

2.1. Présentation de la technologie java :

La technologie Java est à la base de la plupart des applications en réseau, elle est exploitée dans le monde entier pour développer et fournir des applications mobiles et imbriquées, des jeux, du contenu Web et des logiciels d'entreprise. Utilisée par plus de 9 millions de développeurs dans le monde, la technologie Java permet de développer, de déployer et d'utiliser efficacement des applications et des services fascinants.

Des ordinateurs portables aux centres de données, des consoles de jeux aux superordinateurs scientifiques, des téléphones portables à Internet, la technologie Java est présente sur tous les fronts !

Des chiffres sur l'utilisation de JAVA :

- ✓ 97 % des bureaux d'entreprise exécutent Java ;
- ✓ 89 % des bureaux (ou ordinateurs) des Etats-Unis exécutent Java ;
- ✓ 9 millions de développeurs Java dans le monde ;
- ✓ Choix n° 1 des développeurs ;
- ✓ Plate-forme de développement n° 1 ;
- ✓ 3 milliards de téléphones mobiles exécutent Java ;
- ✓ 100 % des lecteurs Blu-ray livrés avec Java ;
- ✓ 5 milliards de cartes Java utilisées ;
- ✓ 125 millions de périphériques TV exécutent Java ;
- ✓ Les 5 fabricants d'équipement d'origine principaux fournissent Java ME.

La technologie Java a été testée, ajustée, étendue et mise à l'épreuve par une communauté dédiée de développeurs, d'architectes et de passionnés de Java. Elle a été conçue pour permettre le développement d'applications portables hautes performances sur une large gamme de plates-formes informatiques. Grâce à la mise à disposition

d'applications dans des environnements hétérogènes, les entreprises peuvent proposer davantage de services et dynamiser la productivité, la communication et la collaboration de l'utilisateur final, tout en réduisant considérablement le coût de propriété des applications d'entreprise et grand public. [15]

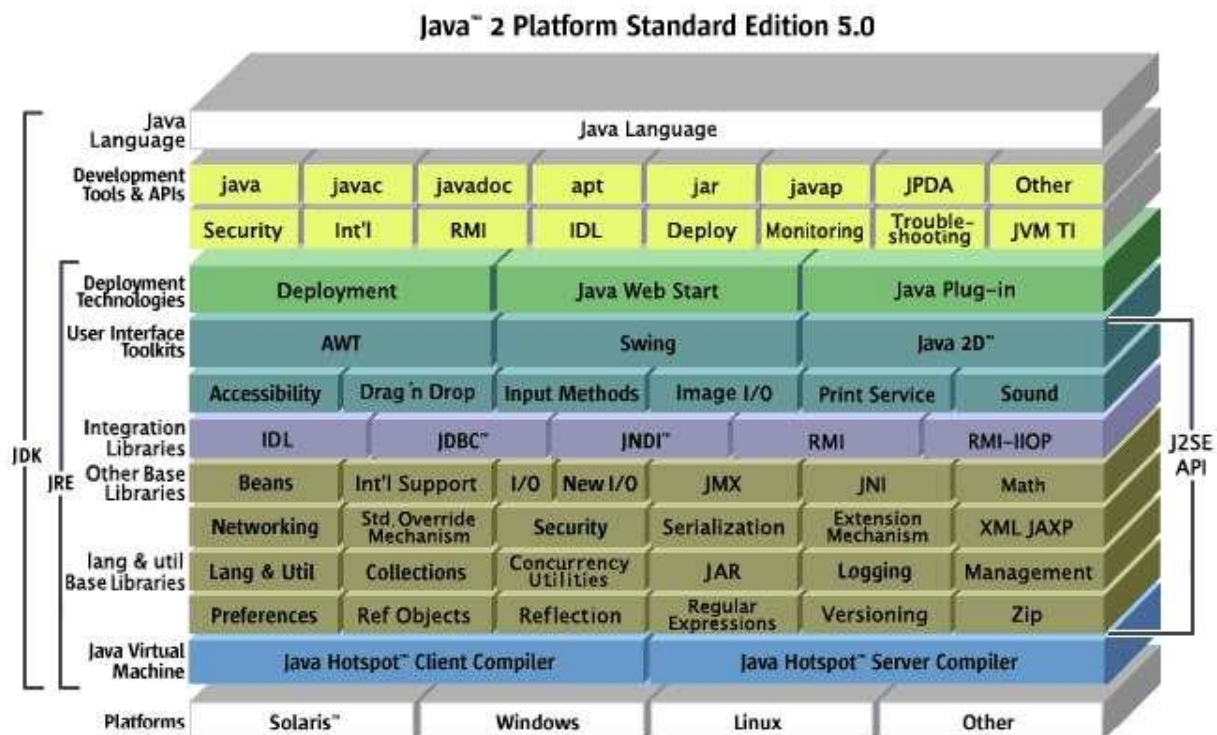


Figure 9: Architecture JAVA

2.2. Présentation de JAVA FX (JAVA Flash FLEX) :

JavaFX est une plate-forme logicielle permettant de créer et de fournir des applications de bureau, ainsi que des applications Internet riches (RIA) pouvant être exécutées sur une grande variété de périphériques. JavaFX est destiné à remplacer Swing en tant que bibliothèque d'interface graphique standard pour Java SE, mais les deux seront inclus dans un avenir prévisible. [16] JavaFX prend en charge les ordinateurs de bureau et les navigateurs Web sous Microsoft Windows, Linux et macOS.

WebView, le composant de navigateur intégré prend en charge les fonctionnalités HTML5 suivantes :

- Canvas,
- Media Playback,
- Form controls (except for <input type="color">),
- Editable content,
- History maintenance,
- Support for the <meter> and <progress> tags,
- Support for the <details> and <summary> tags,
- DOM,
- MathML,
- SVG,

- Support for domain names written in national languages/

Avant la version 2.0 de JavaFX, les développeurs utilisaient un langage déclaratif, typé statiquement, appelé JavaFX Script pour créer des applications JavaFX. Comme JavaFX Script a été compilé en Java bytecode, les programmeurs peuvent également utiliser du code Java. Les applications JavaFX peuvent être exécutées sur tous les ordinateurs de bureau pouvant exécuter Java SE ou sur tout téléphone mobile pouvant exécuter Java ME.

JavaFX 2.0 et les versions ultérieures sont implémentées en tant que bibliothèque Java "native" et les applications utilisant JavaFX sont écrites en code Java "natif". JavaFX Script a été supprimé par Oracle, mais le développement se poursuit dans le projet Visage. [17] JavaFX 2.x ne prend pas en charge le système d'exploitation Solaris ou les téléphones mobiles; Toutefois, Oracle prévoit d'intégrer JavaFX à Java SE Embedded 8 et Java FX pour les processeurs ARM est en phase de prévisualisation pour les développeurs. [18]

Sur les ordinateurs de bureau, JavaFX prend en charge les systèmes d'exploitation macOS et Linux Windows Vista, Windows 7, Windows 8, Windows 10, [19]. À partir de JavaFX 1.2, Oracle a publié des versions bêta pour OpenSolaris. [20] Sur mobile, JavaFX Mobile 1.x peut fonctionner sur plusieurs systèmes d'exploitation mobiles, notamment Symbian OS, Windows Mobile et les systèmes d'exploitation temps réel propriétaires.

Open Source JavaFXPorts fonctionne pour iOS (iPhone et iPad) et Android et embarqué (Raspberry Pi) et le logiciel commercial associé créé sous le nom de "Gluon" prend en charge les mêmes plates-formes mobiles avec des fonctionnalités supplémentaires et le bureau. Cela permet à un seul code source de créer des applications pour les ordinateurs de bureau, iOS et Android.

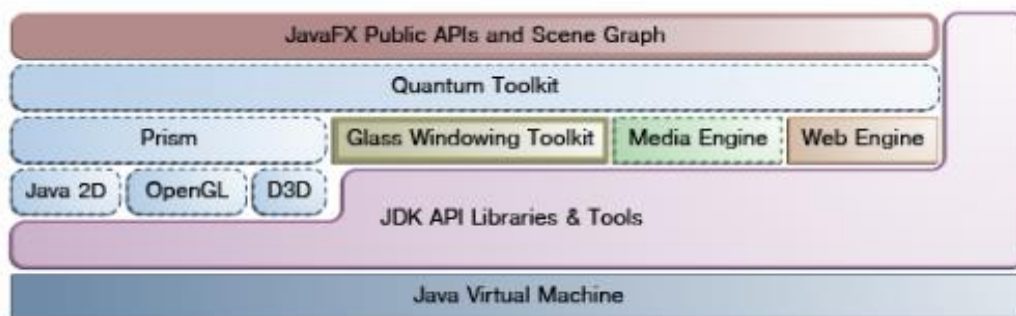


Figure 10: Architecture JAVA FX

2. Présentation de l'application :

Dans cette partie nous allons présenter les différentes interfaces graphiques de notre application.

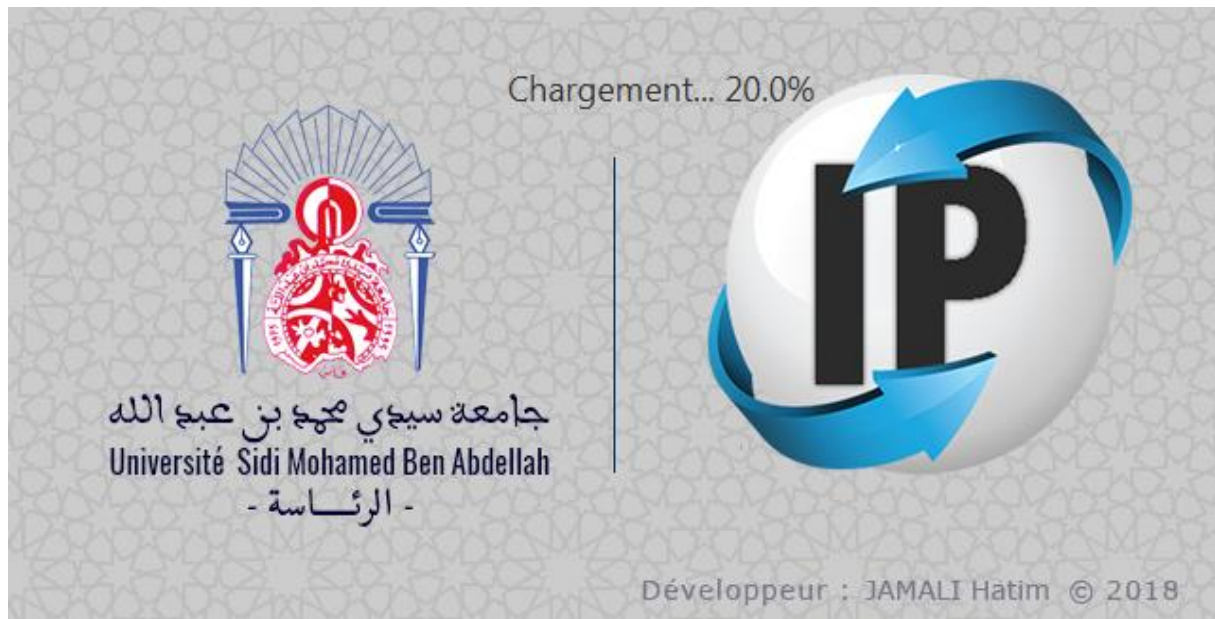


Figure 11:splash screen ou écran de lancement de notre application

3.1 Formulaire d'authentification :

Cette page permet à l'utilisateur de notre système de s'authentifier avant d'accéder à son espace.

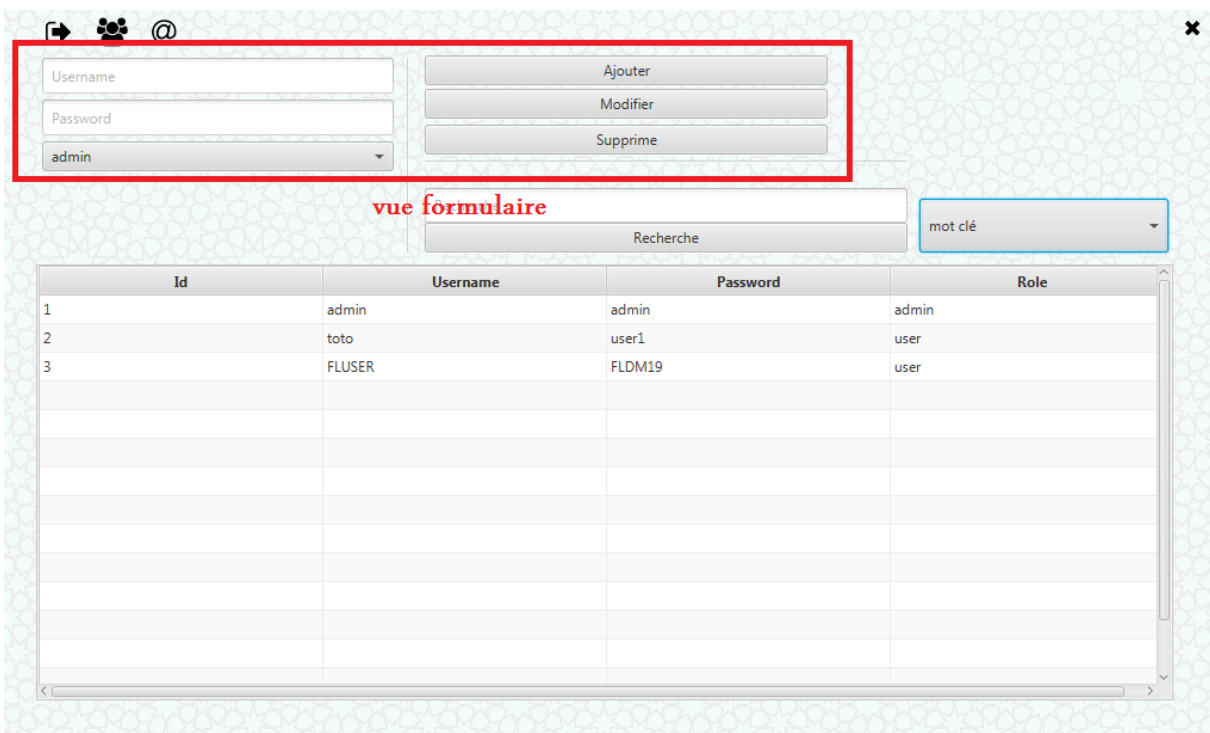
Figure 12:fenêtre d'authentification de l'application

3.2. Gestion des utilisateurs

Les figures suivantes montrent la gestion des utilisateurs on trouve les modes d’affichage suivants :

- Vue Kanban
- Vue formulaire
- Vue liste

Pour ajouter un nouvel utilisateur (modifier ou supprimer) il suffit de renseigner le formulaire ci-après :

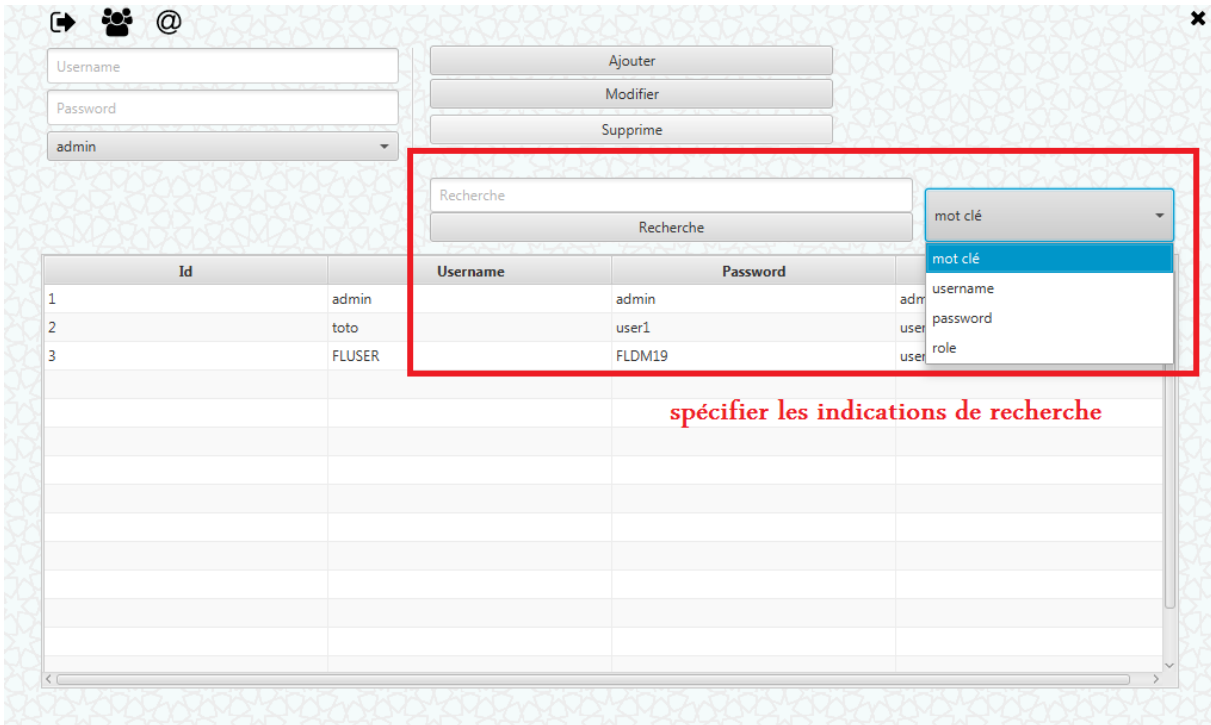


vue formulaire

Id	Username	Password	Role
1	admin	admin	admin
2	toto	user1	user
3	FLUSER	FLDM19	user

Figure 13:: Vue formulaire de gestion des utilisateurs

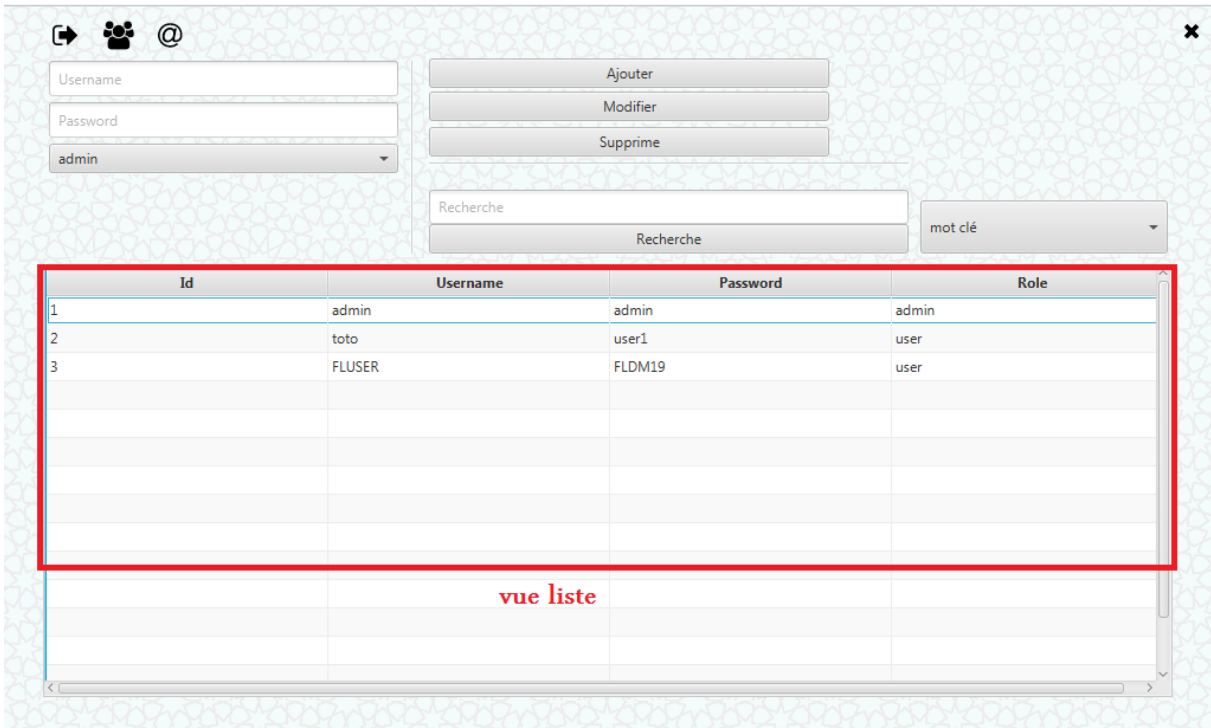
De surcroît, On peut aussi rechercher les utilisateurs suivant des indicateurs prédéfinis à savoir : username, password, rule.



The screenshot shows a user management interface. At the top, there are input fields for 'Username' and 'Password', and a dropdown menu currently set to 'admin'. To the right of these fields are three buttons: 'Ajouter', 'Modifier', and 'Supprimer'. Below these, there is a search section with a 'Recherche' input field and a 'Recherche' button. A dropdown menu is open next to the search button, showing the following options: 'mot clé', 'mot clé', 'username', 'password', and 'role'. The main table below has columns for 'Id', 'Username', 'Password', and 'Role'. The first three rows are populated with data: (1, admin, admin), (2, toto, user1), and (3, FLUSER, FLD19). The text 'spécifier les indications de recherche' is written in red over the table.

Id	Username	Password	Role
1	admin	admin	admin
2	toto	user1	user
3	FLUSER	FLDM19	user

Figure 14: barre de recherche des utilisateurs

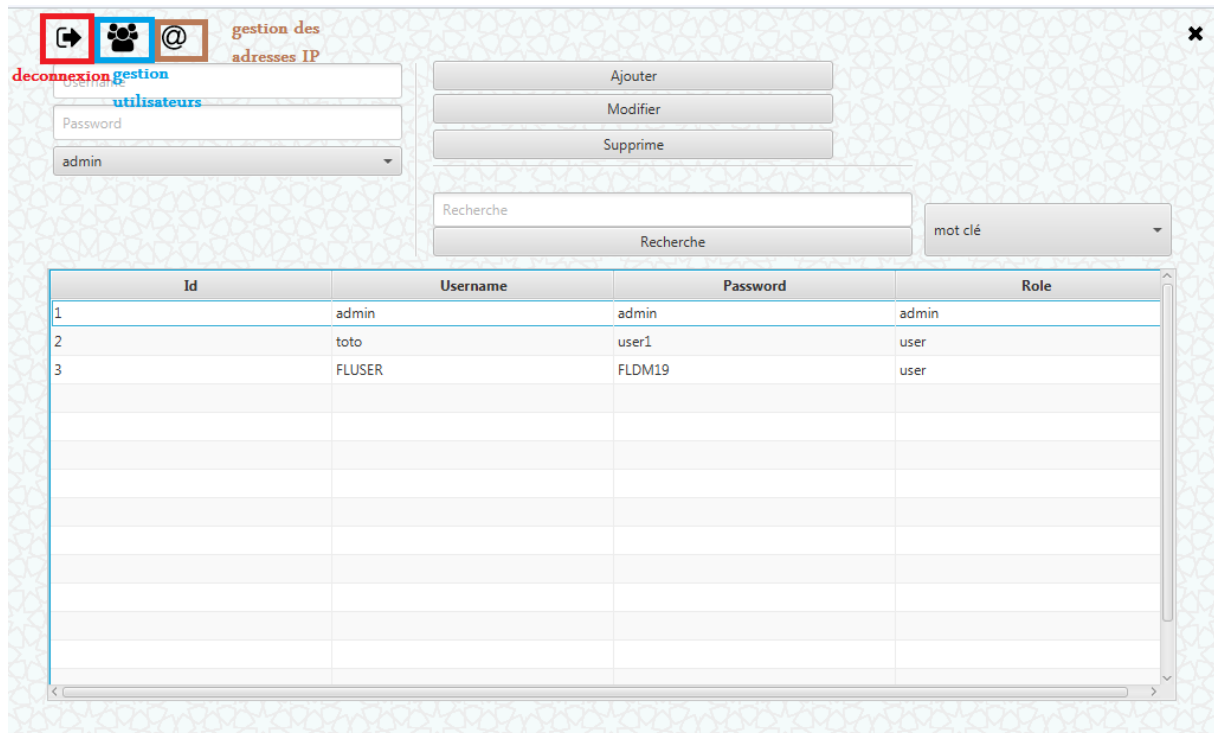


The screenshot shows the same user management interface as Figure 14, but with the search dropdown menu closed. The main table is now fully visible and contains the following data:

Id	Username	Password	Role
1	admin	admin	admin
2	toto	user1	user
3	FLUSER	FLDM19	user

The text 'vue liste' is written in red over the table.

Figure 15: liste de tous les utilisateurs



Id	Username	Password	Role
1	admin	admin	admin
2	toto	user1	user
3	FLUSER	FLDM19	user

Figure 16: fonctionnalités de menu

3.3. Gestion des adresses IP

Dans la partie de gestion des adresses IP nous avons faire 2 actions :

- Développer l'environnement de gestion (pour le futur).
- Intégrer l'ancienne base des données qui est sur EXCEL.

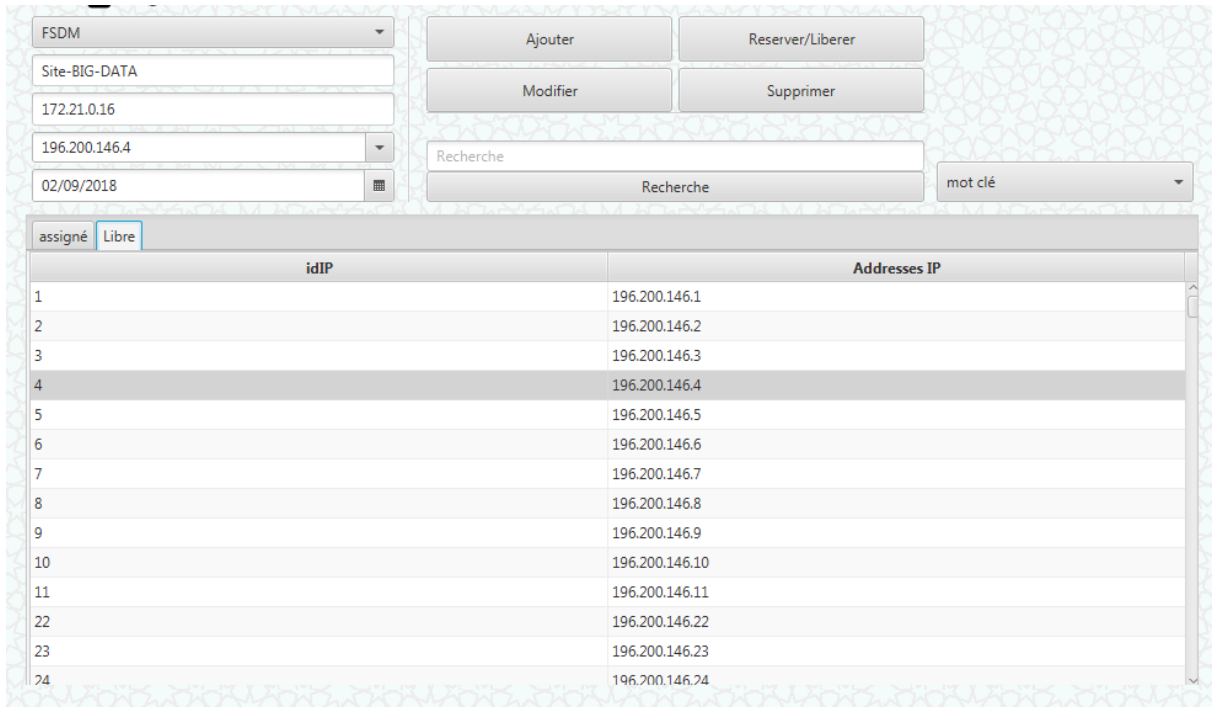
3.3.1. Interface de Gestion des adresses IP

Pour ajouter un nouvel utilisateur (modifier ou supprime) il suffit de renseigner le formulaire ci-après :



Figure 5. 9 : vue formulaire de gestion des adresses IP

Pour réserver une adresse IP il faut chercher sur la liste des adresses IP libre sélectionné et cliquer sur réserver

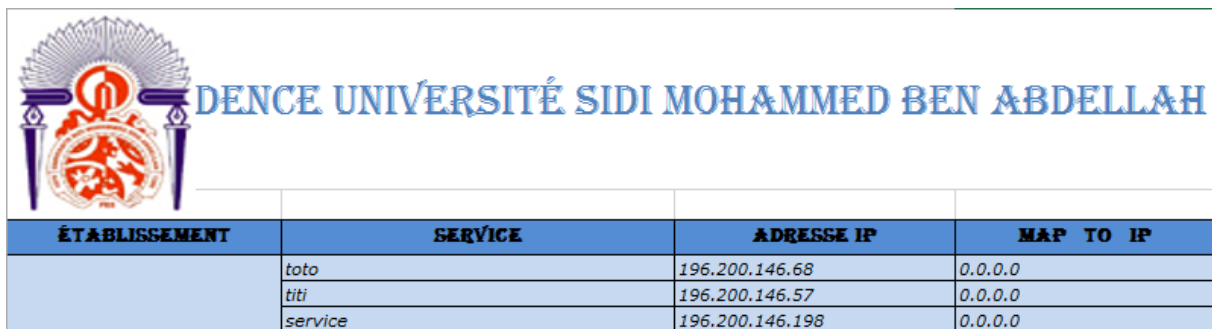


The interface shows a form on the left with fields for 'FSDM', 'Site-BIG-DATA', '172.21.0.16', '196.200.146.4', and '02/09/2018'. On the right are buttons for 'Ajouter', 'Reserver/Liberer', 'Modifier', and 'Supprimer'. Below these is a search bar with 'Recherche' and a 'mot clé' dropdown. The main table has two tabs: 'assigné' and 'Libre'. The 'Libre' tab is active, showing a table with columns 'idIP' and 'Addresses IP'.

idIP	Addresses IP
1	196.200.146.1
2	196.200.146.2
3	196.200.146.3
4	196.200.146.4
5	196.200.146.5
6	196.200.146.6
7	196.200.146.7
8	196.200.146.8
9	196.200.146.9
10	196.200.146.10
11	196.200.146.11
22	196.200.146.22
23	196.200.146.23
24	196.200.146.24

Figure 17: liste des adresses libre

3.3.2. Transformer excel a la base des données



The screenshot shows the header of an Excel spreadsheet with the logo of 'UNIVERSITÉ SIDI MOHAMMED BEN ABDELLAH'. The table has four columns: 'ÉTABLISSEMENT', 'SERVICE', 'ADRESSE IP', and 'MAP TO IP'.

ÉTABLISSEMENT	SERVICE	ADRESSE IP	MAP TO IP
	toto	196.200.146.68	0.0.0.0
	titi	196.200.146.57	0.0.0.0
	service	196.200.146.198	0.0.0.0

Figure 18: liste des adresses réservées sur excel

L'idée c'est de transformer c'est données à une base des données pour ne perde pas les anciens informations et c'est ennuyeux et fatigant de faire manuellement adresse par adresse donc je transforme cette format à CSV après importé cette fichier à l'aide de DB browser for

SQLite

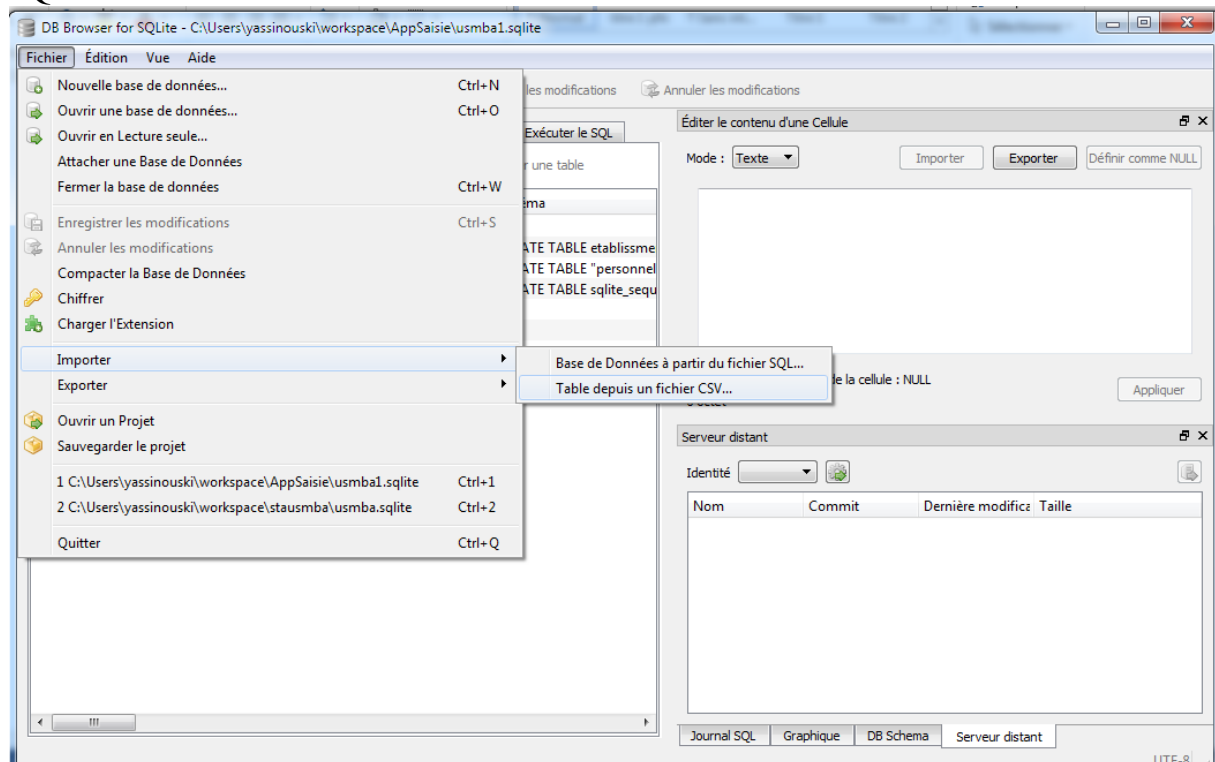


Figure 19: importer la base des données

Conclusion :

Dans ce chapitre, au départ, nous avons mis en avant les outils de développement utilisés pour la programmation de l'application par la suite nous avons présenté des écrans de l'application réalisée.

Conclusion générale

Pendant toute cette période de stage nous avons réalisé une application qui permet la gestion des adresses IP public de la plage des adresses disponible pour USMBA d'une manière simple, sécurisée et efficace.

La première phase de notre projet nous a permis de détecter les problèmes qui entravent le bon fonctionnement de service informatique.

A l'issue de notre stage de fin d'année, nous avons proposés un système informatisé de gestion des adresses IP dont dispose à la présidence de l'USMBA. Le système proposé permet des réservations et une gestion rapide et efficace de toute la plage des adresses disponibles.

D'autre part, notre application permet à l'administrateur d'enregistrer dans une base de données l'ensemble des adresses, ainsi que les informations d'identification et autres concernant les adresses réservé via une interface intuitive. Ce nouveau système permet à l'utilisateur de consulter les adresses libres et effectuer des réservations, il peut aussi gérer les utilisateurs de système en les enregistrant dans une base de données.

Dans le présent projet, vue la contrainte du temps, nous nous sommes concentré sur la partie gestion, les autres fonctionnalités pourront être facilement intégrés ultérieurement.

Nous n'allons pas prétendre que notre travail soit parfait et complet mais c'est un pas de géant vers le bon sens. De plus nous proposons que cette application soit utile pour connexion automatique à l'aide d'un code sur les serveurs, D'autre part nous souhaitons développer une version mobile de l'application.

Afin d'alléger la charge du responsable de service informatique, nous prévoyons dans un futur proche, intégrer dans notre application l'accès au serveur de l'USMBA avec toute la sécurité nécessaire, améliorer le design de l'application ainsi d'intégrer dans le site web de la présidence cette fonctionnalité

Liste des figures

Figure 1: Domaines d'informations de l'USMBA	12
Figure 2: Méthode agile SCRUM	17
Figure 3 :Modèle vue contrôleur.....	26
Figure 4 :Cycle de développement en Y	30
Figure 5: Diagramme de cas d'utilisation du système.....	33
Figure 6 : Processus métier de gestion des adresses IP	34
Figure 7: Diagramme de séquence "Scolarité"	36
Figure 8: Diagramme de classe du système.	37
Figure 9: Architecture JAVA.....	48
Figure 10: Architecture JAVAFX.....	49
Figure 11:splash screen ou écran de lancement de notre application	50
Figure 12:fenêtre d'authentification de l'application	50
Figure 13:: Vue formulaire de gestion des utilisateurs	51
Figure 14: barre de recherche des utilisateurs	52
Figure 15: liste de tous les utilisateurs	52
Figure 16: fonctionnalités de menu	53
Figure 17: liste des adresses libre.....	54
Figure 18: liste des adresses réservées sur excel	54
Figure 19: importer la base des données	55

Liste des tableaux

Tableau 1: Les fonctionnalités à mettre en œuvre dans notre système.....	25
Tableau 2 :Historique des versions de JAVA	42
Tableau 3: Comparaison des méthodes de développement	29
Tableau 4: Description des cas d'utilisation d'administration du système	33

Références

- [1] : <http://www.usmba.ac.ma/~usmba2/a-propos/historique/>
- [2] : <http://www.idnext.net/la-methode-agile-quest-ce-que-cest/>
- [3] : « introduction methodes **agiles** » [Florent Lothon](#)
- [4] : Pete Deemer, Gabrielle Benefied Craig Larman Bas Vodde «Guide Léger de la Théorie et de la Pratique de Scrum », 2012.
- [5] : Stephen Walther, *ASP.NET MVC Framework Unleashed*, Sams Publishing – 2009
- [6] : « So why did they decide to call it Java? » [archive], sur JavaWorld, 4 octobre 1996.
- [7] : Meyer, David (2011-10-06). "JavaFX 2.0 arrives and heads for open source".
- [8] : Greg Brown (2011-08-15). "Introducing FXML".
- [9] : <https://www.sqlite.org/about.html>
- [10] : <https://sqlitebrowser.org>.
- [11] : UML en action De l'analyse des besoins à la conception en Java Pascal ROQUES & Franck VALLÉE
- [12] : PASCAL ROQUES & FRANCK VALEE, « Architecture logiciel, UML2 en action de l'analyse des besoins à la conception », 4eme édition EYROLLES, 2009
- [13] : JOSEPH GABAY & DAVID GABAY, « UML 2 analyse et conception », 1ere édition, DUNOD 2008.
- [14] : Définition et caractéristique d'UML. 2008.
- [15] : <https://www.java.com/fr/about/>
- [16] : " JavaFX Overview (Release 8)". Docs.oracle.com. Retrieved 2016-08-01.
- [17] : "JavaFX FAQ". Oracle.com. Retrieved 2016-08-01.
- [18] : "JavaFX ComboBox not responding on Windows 10". stackoverflow.com. Retrieved 2018-05-01.
- [19] : "JavaFX Supported Configurations". Oracle.com. 2014-04-08. Retrieved 2016-08-01.
- [20] : "JavaFXPorts - Gluon". Gluon. Retrieved 2018-05-01.