|  |  |  |
| --- | --- | --- |
|  |  |  |



Gestion des ressources humaines.

Application de gestion des ressources humaines pour SAMIFIN : Focus sur la gestion des congés et de la santé des employés.

Par

RAMAROSON Andrivola Fanambony Benjamina

Mémoire présenté  
en vue de l’obtention du grade de Licence

en informatique

option Développement

Novembre 2024

Jury :

Dr Olivier Robinson, président  
 M. ANDRIAMBELOMA Tovo, examinateur  
 M. Tombo Hebert, encadreur professionnel

© RAMAROSON Andrivola Fanambony Benjamina, 2024

Table des matières

Liste des tableaux v

Liste des figures vi

Glossaire vii

Avant-propos 1

L’IT University 1

L’Institution/Entreprise d’accueil 2

Remerciements (exemple) 2

Introduction 3

1 Présentation du projet 4

1.1 Objectifs du projet 4

1.2 Planning de réalisation 4

1.3 Technologies utilisées 5

2 Réalisation de l’application 6

2.1 Analyse et conception 6

2.1.1 Analyse de l'existant 6

2.1.2 Conception de l’application 6

2.2 Développement par fonctionnalité ou module 7

2.2.1 Module ou Fonctionnalité numéro 1 7

2.2.2 Module ou Fonctionnalité numéro 2 7

2.2.3 Gestion des utilisateurs 7

2.3 État d’Analyse et Statistiques 8

2.3.1 État numéro 1 8

2.3.2 État numéro 2 8

2.3.3 Statistique numéro 1 8

2.4 Problèmes rencontrés et solutions 9

3 Évaluation du projet et connaissances acquises 10

3.1 Bilan pour l'entreprise 10

3.2 Bilan personnel 10

3.3 Extension et évolution de l’application 10

Conclusion 11

Bibliographie 12

Annexe i

Table des matières

[Liste des tableaux iv](#_Toc181188896)

[Liste des figures v](#_Toc181188897)

[Glossaire vi](#_Toc181188898)

[Avant-propos 1](#_Toc181188899)

[L’IT University 1](#_Toc181188900)

[SAMIFIN 2](#_Toc181188901)

[Remerciements (exemple, soyez simple) 1](#_Toc181188902)

[Introduction 2](#_Toc181188903)

[1 Présentation du projet 3](#_Toc181188904)

[1.1 Objectifs du projet 3](#_Toc181188905)

[1.2 Planning de réalisation 3](#_Toc181188906)

[1.3 Technologies utilisées 4](#_Toc181188907)

[2 Réalisation de l’application 5](#_Toc181188910)

[2.1 Analyse et conception 5](#_Toc181188911)

[2.1.1 Analyse de l'existant 5](#_Toc181188912)

[2.1.2 Conception de l’application 5](#_Toc181188913)

[2.2 Développement par fonctionnalité ou module 6](#_Toc181188914)

[2.2.1 Module ou Fonctionnalité numéro 1 6](#_Toc181188915)

[2.2.2 Module ou Fonctionnalité numéro 2 6](#_Toc181188916)

[2.2.3 Gestion des utilisateurs 6](#_Toc181188917)

[2.3 État d’Analyse et Statistiques 7](#_Toc181188918)

[2.3.1 État numéro 1 7](#_Toc181188919)

[2.3.2 État numéro 2 7](#_Toc181188920)

[2.3.3 Statistique numéro 1 7](#_Toc181188921)

[2.4 Problèmes rencontrés et solutions 8](#_Toc181188922)

[3 Évaluation du projet et connaissances acquises 9](#_Toc181188923)

[3.1 Bilan pour l'entreprise 9](#_Toc181188924)

[3.2 Bilan personnel 9](#_Toc181188925)

[3.3 Extension et évolution de l’application 9](#_Toc181188926)

[Conclusion 10](#_Toc181188927)

[Bibliographie 11](#_Toc181188928)

[Annexe i](#_Toc181188929)

# Liste des tableaux

[Tableau 1 : Classement TIOBE 2014 de quelques langages de programmation 8](#_Toc441148074)

# Liste des figures

[Figure 1: Part de marché des navigateurs en février 2014. 4](#_Toc441148082)

# Glossaire

(A ordonner alphabétiquement)

Astéroïde. Minuscule planète dont la taille varie entre 1 000 km à moins d'un km de diamètre. Les astéroïdes gravitent généralement autour de planètes plus grandes.

Atmosphère. Masse gazeuse entourant les planètes, dont la Terre.

Densité. Nombre de particules par unité de mesure.

**Table.** Structure de données qui organise et stocke des informations sous forme de lignes et de colonnes.

# Avant-propos

Le présent mémoire présente les résultats du travail effectué lors de mon stage de fin d’études de Licence en Informatique de l’IT University; stage effectué au département de SAMIFIN durant 3 mois, d’août à novembre 2024.

Afin de poser clairement le contexte de ce mémoire, je vais présenter succinctement d’une part l’IT University et d’autre part mon [institution/entreprise] d’accueil.

## L’IT University

Fondée en 2011, l’IT University (ou ITU) est une université privée, spécialisée en informatique, formant les jeunes bacheliers, de préférence scientifiques :

* En trois ans, pour l‘obtention d’une :
  + Licence, option Développement, Réseaux et Bases de Données ou Web et Design
  + Licence, option Graphic Design ou Communication Digitale
* En cinq ans, pour l’obtention d’un :
  + Master MBDS en coopération avec l’Université Côte d’Azur à Nice Sophia Antipolis – France
  + Master BIHAR en coopération avec l’ESTIA du Pays Basque - France

Étant une formation professionnalisante, l’ITU a tissé des liens forts avec ses partenaires industriels, dont l’opérateur convergent TELMA et la plupart des entreprises et institutions du secteur des TIC[[1]](#footnote-1). Ces partenaires participent effectivement à la formation par la fourniture de connexion Internet à haut débit, l’envoi de conférenciers ou par l’accueil des étudiants en stage. Beaucoup de ces partenaires recrutent aussi les sortants dès leur sortie d’École.  
D’autre part, le corps enseignant de l’ITU est constitué intégralement de spécialistes de très haut niveau et obligatoirement actifs professionnellement dans leurs domaines respectifs.

## SAMIFIN

Le service de Renseignement Financier, en abrégé « SAMIFIN » est l’entité nationale en charge de conduire la LBC/FT à Madagascar.

Crée en 2007 et réorganisé en 2015, le SAMIFIN est un service de renseignement financier de type administratif, rattaché à la Présidence de la République. Ses rapports d’analyse ont simple valeur de renseignement d’intelligence. Disposant d’une indépendance opérationnelle et d’une autonomie de gestion, le SAMIFIN s’article en 01 Direction Générale, 04 Directions techniques et 01 Direction d’appui.

Entérinée par la loi n° 2018- 043 du 13 février 2019 sur la LBC/ FT, la mission du SAMIFIN consiste à lutter contre les flux financiers illicites liés à des faits de blanchiment de capitaux, de financement du terrorisme, des infractions économiques et financières et à tout crime organisé :

1. Prévention et Détection : Mettre en place des mécanismes pour identifier les activités suspectes et les transactions potentiellement liées au blanchiment d'argent ou au financement du terrorisme.
2. Analyse et Enquête : Analyser les données financières pour détecter les anomalies et collaborer avec les autorités judiciaires dans le cadre d'enquêtes.
3. Coordination et Formation : Travailler avec les institutions financières, les autorités publiques et les organisations internationales pour renforcer les capacités locales en matière de lutte contre le blanchiment d'argent et le financement du terrorisme.
4. Conformité : Assurer la conformité avec les normes et recommandations internationales en matière de lutte contre le blanchiment d'argent et le financement du terrorisme.

## Remerciements

Tout d'abord, je tiens à exprimer ma profonde gratitude envers Dieu pour sa grâce, ses bénédictions, et pour m'avoir accordé la santé, la force et la sagesse tout au long de mes années d'études et dans la réalisation de ce mémoire de fin d'études.

Je souhaite également exprimer ma reconnaissance sincère envers le corps enseignant et les personnels de l'IT University. Leurs conseils, leur engagement et les connaissances partagées ont grandement enrichi mes trois années d'apprentissage.

Mes remerciements chaleureux vont à mes encadreurs, M. TOMBO Hebert et M. RAZAFINJATOVO Aina, pour leur soutien stratégique, leurs orientations, et leur assistance précieuse tout au long de la réalisation de ce présent mémoire.

Un grand merci à mon entreprise d'accueil, SAMIFIN, ainsi qu'à toutes les personnes qui ont généreusement partagé leurs connaissances et leurs perspectives pendant mes trois mois de stage au sein de l'entreprise.

Je souhaite également exprimer ma gratitude envers les membres du jury pour l'intérêt qu'ils ont porté en acceptant d’examiner mon travail et de l’enrichir par leurs propositions.

Enfin, un grand merci à toutes les personnes, qui ont contribué de près ou de loin à la réalisation et à l'accomplissement de ce mémoire.

# Introduction

L’essentiel du travail que j’ai réalisé a porté sur le thème : « Application de gestion des ressources humaines : Focus sur la gestion des congés et de la santé des employés »

La gestion des ressources humaines (GRH[[2]](#footnote-2)) est devenue un élément clé pour améliorer l’efficacité des organisations, y compris au sein des institutions publiques comme SAMIFIN. Cependant, les processus de gestion des congés et des documents de santé sont encore gérés de manière traditionnelle, ce qui engendre des inefficacités et des risques de perte de données. Actuellement, la gestion des congés repose sur des fichiers Excel, tandis que les documents de santé, sont générés au format PDF[[3]](#footnote-3) ou DOCX[[4]](#footnote-4) et stockés localement, exposant ainsi l’organisation à des menaces techniques telles que les virus ou les pannes de système.

Pour répondre à ces défis, la mise en place d'une solution numérique dédiée permettrait de centraliser et de sécuriser la gestion des congés et des documents de santé. Cette solution viserait à automatiser les tâches répétitives et à minimiser les erreurs humaines, tout en offrant une meilleure transparence et un accès facilité aux données en temps réel. En modernisant ces processus, elle répondrait aux besoins de sécurité et de continuité opérationnelle de SAMIFIN.

Ainsi, ce projet constitue un atout stratégique pour optimiser la gestion des ressources humaines au sein de l’organisation, en modernisant les méthodes de travail et en améliorant l’efficacité globale.

De ce fait, mon travail consistait à concevoir une **solution numérique** offrant les fonctionnalités suivantes :

* **Gestion des utilisateurs.**
* **Gestion et génération des documents de santé.**
* **Gestion des congés.**
* **Alertes et notifications.**

La structure de ce mémoire se divise en trois parties principales, qui seront :

* **Présentation du projet.**
* **Réalisation de l’application.**
* **Évaluation du projet et connaissances acquises.**

# Présentation du projet

## Objectifs du projet

Ce projet a pour objectif de créer une solution numérique intégrée pour SAMIFIN, axée sur l'optimisation de la gestion des ressources humaines. En se concentrant sur des fonctionnalités clés, cette application vise à rendre les processus plus efficaces et à améliorer l'expérience des utilisateurs.

**Fonctionnalités essentielles :**

* **Gestion des utilisateurs** : Cette fonctionnalité facilitera l'administration des profils des employés et de leurs familles, permettant ainsi une gestion simplifiée des données personnelles.
* **Génération de documents de santé** : L'application automatise la création de documents cruciaux, tels que les bulletins de consultation, les demandes de remboursement, et les attestations de non-paiement, allégeant ainsi le travail administratif et assurant la disponibilité rapide des documents.
* **Gestion des congés** : La solution propose un système complet pour la gestion des congés, où les employés peuvent facilement soumettre leurs demandes. Ils auront également la possibilité de visualiser leurs demandes sur un calendrier, de suivre l'état de leurs congés, et de recevoir des notifications pour chaque étape du processus. En outre, l'application permet de calculer le solde de congé et de régulariser les congés antérieurs.

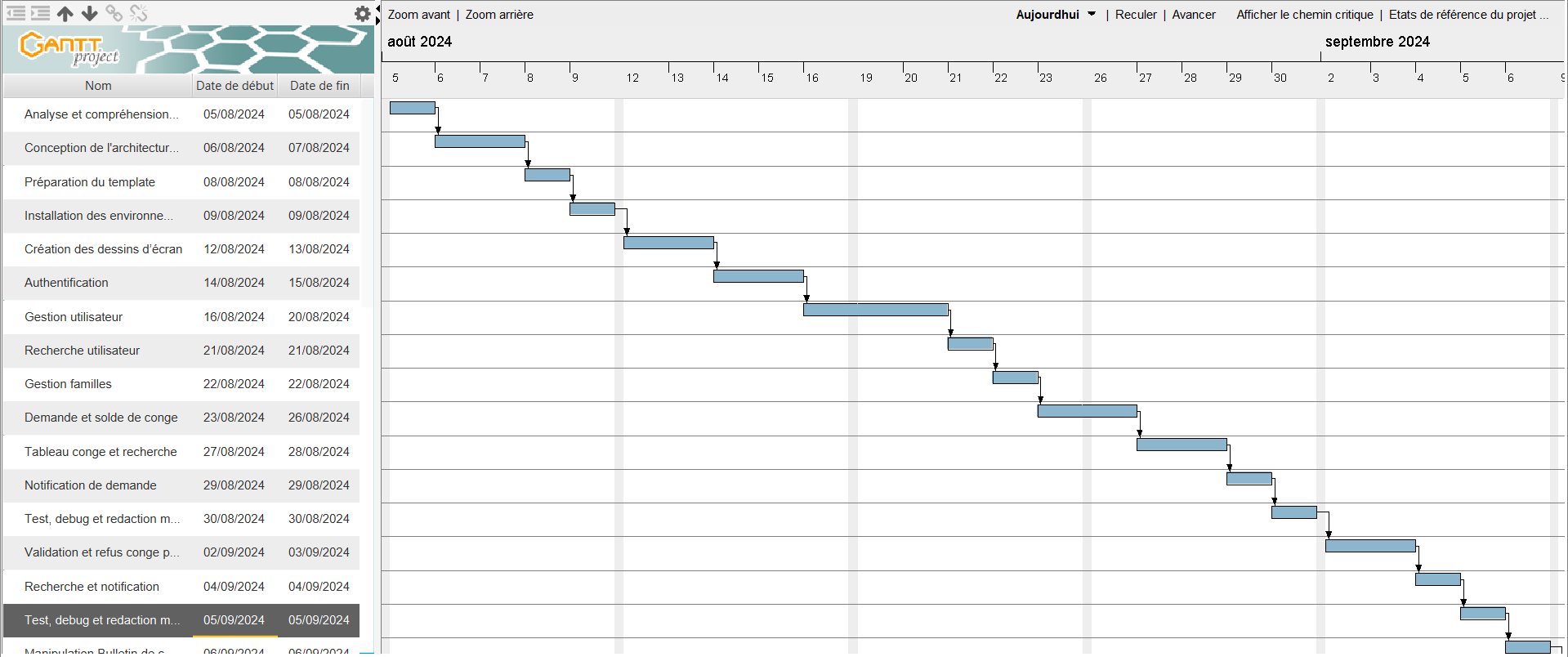
Tous ces objectifs se résument en une seule vision : créer un système de gestion des ressources humaines efficace et intégré qui facilite le travail des employés tout en garantissant la sécurité des données et la conformité aux normes. En centralisant les informations et en automatisant les processus, cette solution aspire à améliorer non seulement la productivité et l'efficacité au sein de SAMIFIN, mais aussi la satisfaction et le bien-être des employés.

En somme, ce projet représente une avancée significative vers la modernisation des pratiques de gestion des ressources humaines, permettant à SAMIFIN de répondre aux défis actuels tout en se préparant pour l'avenir.

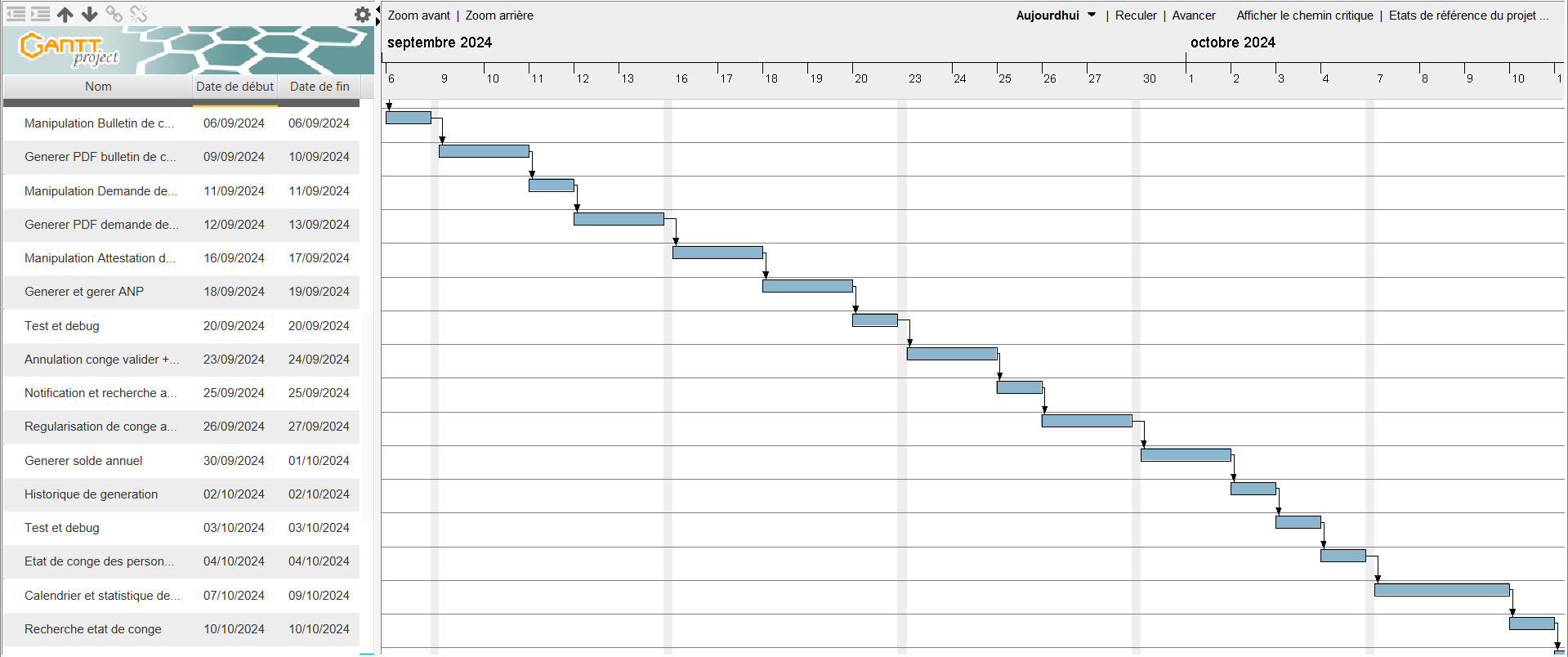
## Planning de réalisation

Le projet s’est déroulé sur une période de trois mois, avec un processus structuré en trois phases essentielles :

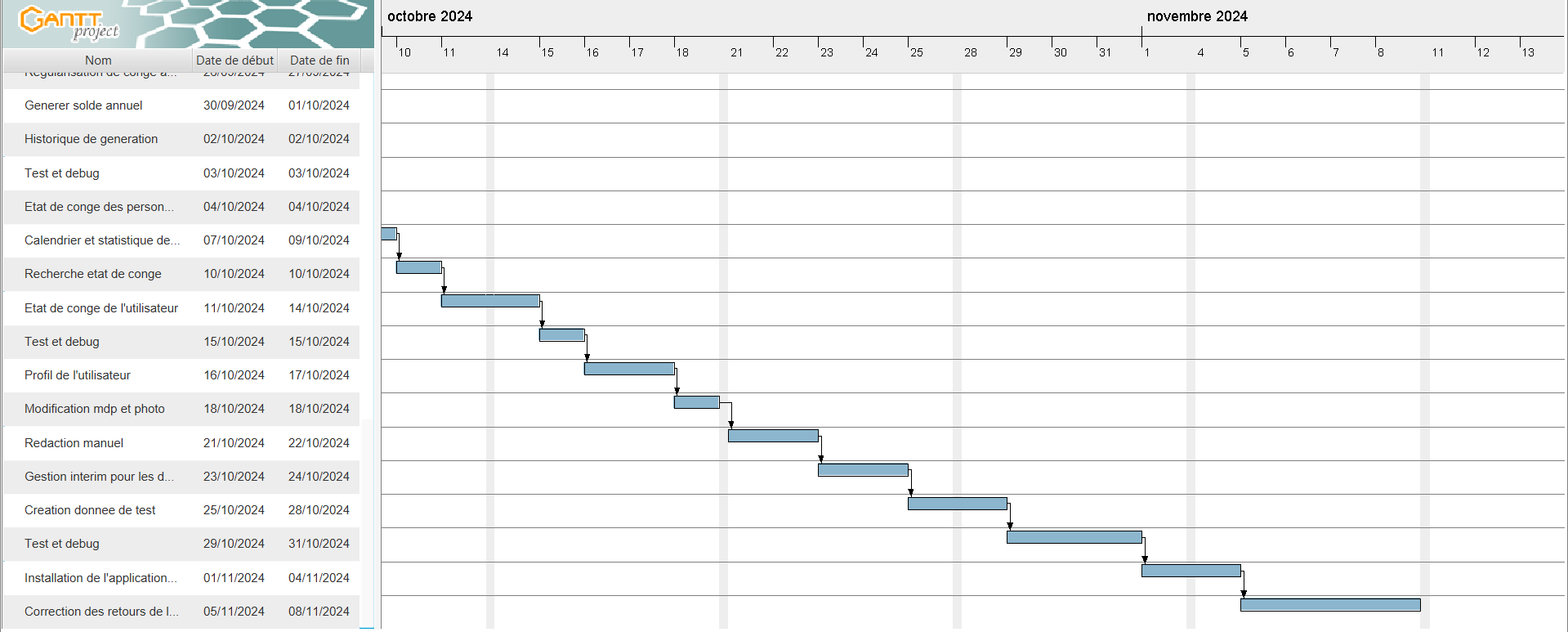
* **La phase de préparation :** analyse et compréhension du projet, conception du projet, préparation du Template, installation des environnements de travail, création des dessins d’écran.
* **La phase de développement :** développement des fonctionnalités.
* **La phase de finalisation :** tests, installation sur un ordinateur du personnel pour validation, corrections des retours et débogage.



**Figure 1 : Planning de réalisation avec diagramme de Gantt Partie 1**



**Figure 2 : Planning de réalisation avec diagramme de Gantt Partie 2**



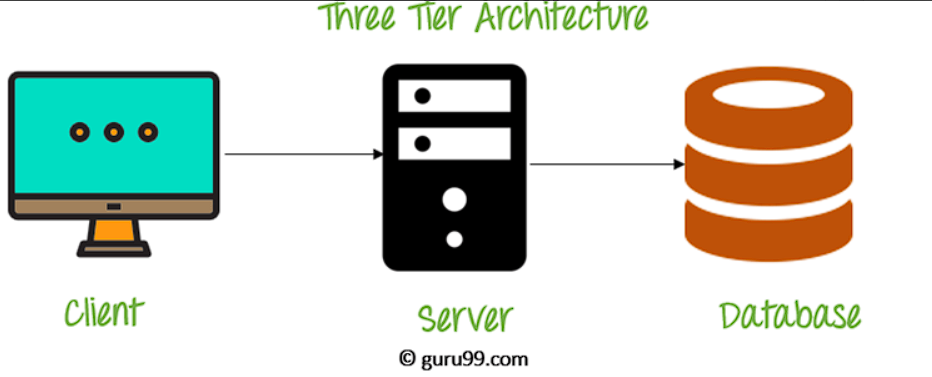
**Figure 3 : Planning de réalisation avec diagramme de Gantt Partie 3**

## Technologies utilisées

### Architecture de l’application

L'application adopte une architecture multi-tiers basée sur le modèle **MVC[[5]](#footnote-5) (Modèle-Vue-Contrôleur)**. Ce modèle, qui est l'une des configurations les plus courantes en architecture multi-tiers, divise l'application en trois niveaux distincts :

* **Couche de présentation (Niveau Client)** : Cette couche est responsable de l'interface utilisateur, où l'interaction avec les utilisateurs a lieu. Elle utilise des technologies telles que JSP[[6]](#footnote-6) pour afficher les données.
* **Couche de logique métier (Niveau Application)** : Cette partie gère la logique et les règles métier de l'application. Elle est implémentée avec **Spring Boot**, qui permet de créer des applications robustes et évolutives.
* **Couche de données (Niveau Données)** : Cette couche s'occupe de la gestion et de la manipulation des données. Dans notre application, cela est réalisé avec **PostgreSQL**, qui sert de système de gestion de base de données.



**Figure 4 : Architecture Multi-tiers**[[7]](#footnote-7)

**Tableau 1 : Tableau comparatif entre l’architecture multi-tiers et monolithique**

|  |  |  |
| --- | --- | --- |
| **Critères** | **Architecture Multi-tiers** | **Architecture Monolithique** |
| **Structure** | Séparée en plusieurs couches distinctes | Un seul bloc de code contenant toutes les fonctionnalités |
| **Flexibilité** | Facilité d'évolution et de mise à jour des composants | Modifications difficiles, impactant souvent l'ensemble de l'application |
| **Couches typiques** | Présentation, logique métier, données | Pas de séparation claire entre couche |
| **Maintenance** | Plus facile grâce à la séparation des préoccupations | Complexe en raison de l'interdépendance des composants |
| **Développement** | Peut impliquer plusieurs équipes travaillant sur des couches spécifiques | Développé généralement par une seule équipe |
| **Performance** | Latence potentielle entre les couches, mais possibilité d'optimisation | Performances optimales en raison de l'intégration, mais difficultés à gérer la complexité |
| **Déploiement** | Nécessite souvent des configurations plus complexes | Déploiement simple, mais des mises à jour peuvent être risquées |
| **Avantages** | Meilleure organisation du code | Mise a l’échelle globale de toute l’application |

L’architecture multi-tiers est souvent préférée à l'architecture monolithique pour sa modularité et sa flexibilité. Elle sépare l’application en couches distinctes (présentation, logique métier, et données) permettant une maintenance plus facile, une évolutivité accrue, et une gestion indépendante de chaque composant. Contrairement aux architectures monolithiques, où toutes les fonctionnalités sont intégrées dans une seule structure, le multi-tiers permet des mises à jour et des déploiements plus simples, ce qui est particulièrement utile dans les environnements modernes de déploiement en cloud. Cette approche permet donc de mieux répondre aux exigences de performance, d’agilité, et de scalabilité des applications actuelles.

### Java Spring Boot

Java est un langage de programmation orienté objet et une plateforme informatique, bien établi pour le développement d'applications robustes et sécurisées. Il est largement utilisé dans divers secteurs pour créer des applications fiables, maintenables et portables sur plusieurs plateformes, ce qui en fait un choix de premier plan pour le développement web et logiciel d'entreprise.

Spring Boot, qui repose sur le framework Java Spring, est conçu pour simplifier le développement d'applications Java en réduisant la complexité de configuration. Grâce à Spring Boot, les développeurs peuvent créer des applications Java plus rapidement, car il automatise de nombreuses configurations standard, en particulier celles liées à la création d'applications web. Ce framework est idéal pour les projets nécessitant une croissance progressive et une maintenance à long terme, car il permet d’ajouter des fonctionnalités facilement tout en offrant une bonne gestion de la performance et de la sécurité.

Spring Boot tire ainsi parti de la fiabilité de Java tout en optimisant le processus de développement, ce qui en fait un excellent choix pour des applications modernes, évolutives et orientées vers l’avenir. En 2024 Java reste un acteur majeur et incontournable dans le monde du développement logiciel.



**Figure 6 :** Classement des langages de programmation les plus populaires en octobre 2023 et octobre 2024[[8]](#footnote-8)

Le classement des langages de programmation montre que Java conserve une place de choix parmi les développeurs. Bien que de nouveaux langages émergent, Java reste une valeur sûre pour les projets à long terme et les applications exigeantes. Sa stabilité, sa polyvalence et sa grande communauté en font un choix judicieux pour de nombreux développeurs et entreprises.

**Tableau 2 : Tableau comparatif entre les Framework Spring Boot et .NET**

|  |  |  |
| --- | --- | --- |
| **Critères** | **Spring Boot** | **.NET** |
| **Langage principal** | **Java** | **C#** |
| **Écosystème** | Open-source, basé sur Java | Maintenu par Microsoft, open-source |
| **Flexibilité** | Très modulaire, choix dans les configurations | Construit pour l’intégration fluide avec les services Microsoft |
| **Performance** | Performant, surtout dans des environnements Java | Élevée, optimisé pour les environnements Microsoft |
| **Outils de Développement** | Large choix d’IDE[[9]](#footnote-9) (Eclipse, IntelliJ IDEA) | Visual Studio, Visual Studio Code |
| **Sécurité** | Bonne sécurité, dépend des configurations (Spring Security) | Fort soutien de sécurité Microsoft, intégration native |
| **Popularité dans les entreprises** | Large adoption dans les grandes entreprises | Très populaire dans les entreprises utilisant des technologies Microsoft |

### ****JavaServer Page (JSP)****

JSP[[10]](#footnote-10) (JavaServer Pages) est une technologie utilisée pour créer des pages web dynamiques dans des applications Java. Elle permet d'intégrer du code Java directement dans le HTML, ce qui facilite la création de contenu dynamique côté serveur.

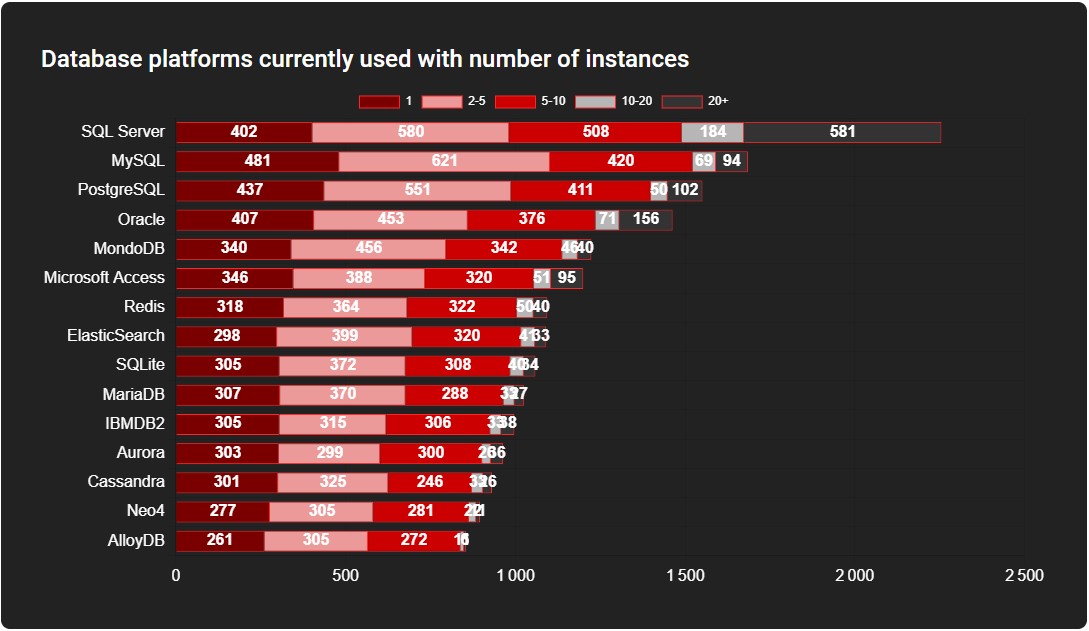
**Tableau 3 : Tableau comparatif entre JSP et Thymeleaf**

|  |  |  |
| --- | --- | --- |
| **Critère** | **JSP** | **Thymeleaf** |
| **Type** | Technologie de pages web dynamiques | Moteur de template |
| **Syntaxe** | Mélange de HTML et de code Java | Attributs HTML déclaratifs |
| **Performance** | Généralement plus rapide | Peut-être moins performant |
| **Intégration Java** | Intégration native avec Java | Séparation plus nette |
| **Flexibilité** | Plus flexible pour le code côté serveur | Moins flexible pour le code côté serveur |
| **Standardisation** | Basé sur les standards Java EE[[11]](#footnote-11) | Propriété de Thymeleaf |
| **Maturité** | Plus ancien et établi | Plus récent |
| **Compatibilité outils** | Bonne compatibilité avec les outils Java existants | Meilleure intégration avec les outils de conception web |

En résumé, bien que Thymeleaf offre certaines avantages modernes, JSP reste une option solide pour les projets Java qui valorisent la stabilité, les performances et l'intégration native avec Java.

### PostgreSQL

### PostgreSQL est un système de gestion de base de données relationnelle open-source, réputé pour sa robustesse, sa conformité aux normes ACID (Atomicité, Cohérence, Isolation, Durabilité), et ses nombreuses fonctionnalités avancées. Il est conçu pour gérer des charges de travail de grande envergure, allant des applications simples aux applications complexes de traitement de données. PostgreSQL supporte les types de données complexes, les transactions, les procédures stockées, ainsi que la gestion de données non structurées grâce au support de types JSON et XML.



**Figure 7 :** Top 15 des plateformes SGBD les plus déployées[[12]](#footnote-12)

**PostgreSQL** (souvent présenté comme l’alternative open source à Oracle Database pour sa forte compatibilité avec cette dernière) **est désormais passé devant Oracle** en termes d’instances déployées. Voilà qui confirme à la fois la très forte progression de PostgreSQL ces dernières années mais aussi la montée des bases en open source, très largement représentées dans ce TOP 15.

**Tableau 4 : Tableau comparatif entre PostgreSQL et MySQL**

|  |  |  |
| --- | --- | --- |
| **Critère** | **PostgreSQL** | **MySQL** |
| **Type de Base de Données** | Relationnelle avec support des données non structurées (JSON[[13]](#footnote-13), XML[[14]](#footnote-14)) | Relationnelle |
| **Licence** | Open-source | Open-source (avec versions commerciales) |
| **Conformité aux standards SQL** | Haute conformité aux standards SQL | Moins strict dans la conformité SQL |
| **Gestion des transactions** | Transactions ACID, support complet des transactions | Transactions ACID (dans les versions InnoDB) |
| **Performances** | Bonne gestion des grandes bases de données, optimisation des requêtes | Rapide pour les lectures |
| **Extensibilité et plugins** | Très extensible avec de nombreux plugins | Moins d’extensions disponibles que PostgreSQL |
| **Sécurité** | Sécurité avancée avec les droits d'accès et le chiffrement | Sécurité de base, nécessite des configurations avancées pour certaines fonctionnalités |

### Outils de développement

#### Visual Studio Code

Visual Studio Code ou VS Code est un éditeur de code source gratuit et open source, développé par Microsoft. Il est utilisé par les développeurs de logiciels pour créer, modifier et déboguer du code informatique dans divers langages de programmation. Ce logiciel est compatible avec différentes plateformes, notamment Windows, macOS et Linux. Il propose des fonctionnalités pratiques telles que la coloration syntaxique, l'auto-complétions, la détection des erreurs en temps réel et un outil de débogage intégré. VS Code est essentiel pour les développeurs car il prend en charge le contrôle de version Git, ce qui simplifie la gestion des modifications de code et la collaboration entre les membres de l'équipe de développement.

#### DbSchema

DBSchema est un outil informatique conçu pour faciliter la création, la visualisation et la gestion des schémas de bases de données. Il offre une interface graphique conviviale qui permet aux utilisateurs de concevoir des structures de bases de données, de générer automatiquement le code SQL correspondant, et de prendre en charge diverses tâches liées à la gestion des bases de données.

#### Outils de génération Framework S5 (Mr Naina)

# Réalisation de l’application

## Analyse et conception

### Analyse de l'existant

### Conception de l’application

Pourquoi doit-on faire la conception ? Pourquoi ne pas faire directement ?

Lister vos travaux, exemple :

* Nombre de tables créées
* Nombre de classes
* Nombre d’écrans
* Etc…

## Développement par fonctionnalité ou module

### Module ou Fonctionnalité numéro 1

### Module ou Fonctionnalité numéro 2

### Gestion des utilisateurs

* Gestion des utilisateurs
  + Description des fonctionnalités du module
  + Pourquoi a-t-on développé ce module pour l'entreprise
  + Scénario d'utilisation clé (1 ou 2)
    - Quel scénario ?
    - Dessin écran + explication
    - Importance
    - Comment ?
      * Pas forcément besoin d’un extrait du code source
      * Diagramme de séquence

## État d’Analyse et Statistiques

### État numéro 1

### État numéro 2

### Statistique numéro 1

Tableau 1 : Classement TIOBE 2014 de quelques langages de programmation (**exemple**)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Position Jan ‘14 | Position Jan ‘13 | Language | Share Jan ‘14 | Trend |
| 1 | 1 | Java | 26.2% | -0.6% |
| 2 | 2 | PhP | 13.2% | -1.6% |
| 3 | 6 | Python | 10.2% | +1.3% |
| 4 | 3 | C# | 9.6% | -0.4% |
| 5 | 4 | C++ | 8.9% | 0% |
| 6 | 5 | C | 8.1% | -0.2% |
| 7 | 7 | JavaScript | 7.6% | +0.3% |

Il ne s’agit pas de reproduire ce tableau, surtout si vous ne faites pas une comparaison entre les langages de programmation : ceci est un exemple de présentation de données statistiques.  
REMARQUEZ que **des données chiffrées doivent être datées et récentes**.

## Problèmes rencontrés et solutions

Il s’agit de **présenter ici les problèmes, les plus intéressants/importants**, soulevés durant le stage et les solutions que vous avez apporté.

**Il ne s’agit pas de raconter vos propres lacunes,** que vous avez dû résoudre pour réaliser le travail.

# Évaluation du projet et connaissances acquises

## Bilan pour l'entreprise

**A faire avec le contrôle de votre encadreur d’entreprise**

## Bilan personnel

N’insistez pas sur vos lacunes.

## Extension et évolution de l’application

# Conclusion

Doit inclure **au moins** 3 points :

* Les avantages et satisfactions exprimés par l’entreprise/institution d’accueil au vu de l’atteinte des objectifs fixés au début du stage,
* Vos impressions personnelles
* Une partie **perspective** d’extension du travail effectué.

**Insistez** alors sur les apports de votre travail par rapport à l’existant.

La conclusion doit tenir **obligatoirement** sur une page maximum.

# Bibliographie

Vous avez ici 3 exemples (livre, référence Web, thèse/mémoire) : veuillez suivre scrupuleusement les formats.

Un travail de mémoire comporte obligatoirement une recherche documentaire sur les problèmes et techniques abordés : **veillez à étoffer vos références pour montrer que vous avez fait correctement ce travail de recherche**.

DELLEY, A., FRANCIOLI, M., ZBINDEN, P., *Technologies d’accès aux réseaux*, Fribourg : Ecole d’ingénieurs et d’architectes de Fribourg, 2007. 220p.

Damien A., *Concevez votre réseau TCP/IP* [en ligne]. Disponible sur : https://openclassrooms.com/en/courses/6944606-concevez-votre-reseau-tcp-ip?archived-source=857447 (consultée le 31-05-2022)

GAUTHIER C. *Contribution à l'étude du fractionnement de l'aluminium libéré dans des solutions de sols forestiers : influence de la quantité et de la nature de la matière organique*. Thèse de doctorat d’université. Limoges : Université de Limoges, 2003.

# Annexe

S’il y en a : pour isoler des éléments techniques afin de laisser le mémoire plus lisible, surtout par des non spécialistes.

1. TIC : Technologies de l’Information et de la Communication [↑](#footnote-ref-1)
2. GRH : Gestion des Ressources Humaines [↑](#footnote-ref-2)
3. PDF : Portable Document Format [↑](#footnote-ref-3)
4. Document Open XML [↑](#footnote-ref-4)
5. MVC : Modèle-Vue-Contrôleur [↑](#footnote-ref-5)
6. JSP : JavaServer Pages [↑](#footnote-ref-6)
7. Source : [091318\_0745\_DBMSArchite3.png (749×302)](https://www.guru99.com/images/1/091318_0745_DBMSArchite3.png) [↑](#footnote-ref-7)
8. Source : [Top 20 des langages de programmation en octobre 2024 - Codeur Blog](https://www.codeur.com/blog/top-20-des-langages-de-programmation-en-octobre-2024/) [↑](#footnote-ref-8)
9. IDE : Environnement de Développement Intégré [↑](#footnote-ref-9)
10. JSP : JavaServer Page [↑](#footnote-ref-10)
11. Java EE : Java Platform, Enterprise Edition [↑](#footnote-ref-11)
12. Source : [Quels sont les SGBD les plus répandus en entreprise en 2024 ?](https://www.itforbusiness.fr/les-sgbd-se-multiplient-en-entreprise-mais-sql-server-resiste-73189) [↑](#footnote-ref-12)
13. JSON : JavaScript Object Notation [↑](#footnote-ref-13)
14. XML : eXtensible Markup Language [↑](#footnote-ref-14)