



**UNIVERSITE DE FIANARANTSOA**  
**ECOLE NATIONALE D'INFORMATIQUE**

**MEMOIRE DE FIN D'ETUDES POUR L'OBTENTION DU**  
**DIPLOME DE MASTER PROFESSIONNEL**

**Mention : Informatique**

**Parcours : Informatique Général**

Intitulé :

**Conception et création d'un outil de suivi et de  
management de projet**

Présenté le 22 Novembre 2024

**Par** Monsieur RAZAFIMAHEFA Johanne Fanantenana Michaël

**Membres du Jury :**

Président : Monsieur DIMBISOA William Germain, Maître de conférences

Examineur : Madame FANOMEZANA Mihajaso Lea, Doctorante en informatique ;

Rapporteurs : Monsieur RALAIVAO Jean Christian, Maître de Conférences

Monsieur ANDRIATSILAVINA Damien, Chef de département Transformation  
digitale

Année universitaire : 2023 – 2024



## CURRICULUM VITAE

Nom : RAZAFIMAHEFA

Prénoms : Johanne Fanantenana Michaël

Née le : 03 juillet 2002 à Ankadikely Ilafy

Adresse postale: Lot E73AM Ambodivona Sabotsy-Namehana

Adresse e-mail : [johanne.razafimahefa@gmail.com](mailto:johanne.razafimahefa@gmail.com)

Numéro de téléphone : +261 38 21 886 69



### **Formations et diplôme obtenus :**

- 2023-2024 : Deuxième année de formation en master à l'Ecole Nationale d'Informatique.
- 2023 : Obtention du diplôme de licence en informatique général à l'Ecole National d'Informatique.
- 2018 : Obtention du diplôme de baccalauréat série C étant élève au lycée Andrianampoinimerina Sabotsy-Namehana.

### **Stages et Expériences professionnelles :**

- 2024 : Conception et Réalisation d'un outil de gestion de projet pour Ravinala Airports lors d'un stage de 6 mois développer avec C# et React pour faciliter la gestion des ressources et budget à chaque projet.
- 2024 : Réalisation d'un site et application de menue virtuel pour le compte d'un particulier.
- 2024 : Participation a Webcup 2024 et deuxième place lors du devhunt 2024
- 2023-2024 : Réalisation d'une application mobile de gestion de livraison de produit pour l'entreprise Fivatsy.
- 2023 : Participation à l'hackathon inter-universitaire 2023 et deuxième place lors du devhunt 2023
- 2022 : Réalisation d'un chatbot intelligent pour le site de l'entreprise Materauto lors d'un stage de 3 mois en utilisant le langage JavaScript et PHP. Ce chatbot devait répondre correctement et automatiquement aux messages des utilisateurs connectés sur le site de Materauto.
- 2022 : Troisième place lors du concours WCC dev organisé par Techzara.

- 2021 : Réalisation d'un site distant géré par pfSense lors d'un stage de 3 mois à la JiRaMa qui gère les personnes connectées sur le réseau de la JiRaMa.
- 2020 : Surveillance de trafic d'un réseau dans une topologie faite sur Gns3 à l'aide des outils VMWare Workstation, Wireshark et Eyes of Network.
- 2019 : Réalisation d'une application desktop réalisant la gestion des utilisateurs Windows en utilisant spécialement les langages C et C++.

### **Compétences en Informatiques :**

Systèmes d'exploitation : Windows, Linux.

Langages de programmation : JavaScript, PHP, Java, Python,

Frameworks et bibliothèques : React, Laravel, Flask, tailwindcss, TypeScript.

SGBD: MySQL, Mongo, SQLite, PostgreSQL, ORM (Prisma, Hibernate)

Langage de modélisation : UML

Méthodes de conception : Merise, 2TU, Agile

Outils de virtualisation : GNS3, VMWare,

Surveillance réseaux : EON, WireShark, CISCO.

### **Connaissances linguistiques :**

LANGUE	COMPREHENSION AUDITIVE	LECTURE	REDACTION ECRITE	COMUNICATION ORAL
Français	Bien	Bien	Bien	Bien
Anglais	Assez bien	Bien	Bien	Assez bien

### **Centres d'intérêts et loisirs :**

- Lecture et cinématographie
- Jeux vidéo
- Randonnée et natation

## SOMMAIRE GENERAL

<b>CURRICULUM VITAE .....</b>	<b>I</b>
<b>SOMMAIRE GENERAL .....</b>	<b>III</b>
<b>REMERCIEMENTS.....</b>	<b>VI</b>
<b>LISTE DES FIGURES.....</b>	<b>VIII</b>
<b>LISTE DES TABLEAUX .....</b>	<b>X</b>
<b>LISTES DES ABREVIATIONS .....</b>	<b>XI</b>
<b>INTRODUCTION GENERALE .....</b>	<b>1</b>
<b>PARTIE I : PRESENTATIONS .....</b>	<b>2</b>
<b>Chapitre 1. Présentation de l'Ecole Nationale d'Informatique.....</b>	<b>3</b>
<b>1.1. Information d'ordre générale .....</b>	<b>3</b>
<b>1.2. Missions et historiques .....</b>	<b>3</b>
<b>1.3. Organigramme institutionnel .....</b>	<b>4</b>
<b>1.4. Domaine de spécialisation.....</b>	<b>5</b>
<b>1.5. Architecture des formations pédagogiques.....</b>	<b>6</b>
<b>1.6. Relation de l'ENI avec les organismes externes .....</b>	<b>7</b>
<b>1.7. Débouchés professionnels et diplômés.....</b>	<b>8</b>
<b>1.8. Ressources humaines.....</b>	<b>9</b>
<b>Chapitre 2 : Présentation de Ravinala Airports .....</b>	<b>10</b>
<b>2.1. Information d'ordre général .....</b>	<b>10</b>
<b>2.2. Nos valeurs.....</b>	<b>10</b>
<b>2.3. Nos missions.....</b>	<b>11</b>
<b>2.4. Dates clés.....</b>	<b>11</b>
<b>2.5. Organigramme de Ravinala Airports .....</b>	<b>11</b>
<b>2.6. Nos prix et certifications.....</b>	<b>12</b>
<b>2.7. Nos actionnaires .....</b>	<b>13</b>

<b>Chapitre 3 : Description du projet .....</b>	<b>14</b>
<b>3.1. Formulations.....</b>	<b>14</b>
<b>3.2. Objectif et besoin de l'utilisateur .....</b>	<b>14</b>
<b>3.3. Moyen nécessaire à la réalisation du projet.....</b>	<b>15</b>
<b>3.4. Résultats attendus .....</b>	<b>16</b>
<b>3.5. Chronogramme de travail .....</b>	<b>16</b>
<b>PARTIE II : ANALYSE ET CONCEPTION.....</b>	<b>18</b>
<b>Chapitre 4 : Analyse préalable .....</b>	<b>19</b>
<b>4.1. Analyse de l'existant .....</b>	<b>19</b>
<b>4.3. Conception avant-projet.....</b>	<b>20</b>
<b>Chapitre 5. Analyse conceptuelle .....</b>	<b>26</b>
<b>5.1. Désignation des rôles de l'équipe SCRUM .....</b>	<b>26</b>
<b>5.2. Etapes de l'élaboration du Product Backlog .....</b>	<b>26</b>
<b>5.3. Product Backlog .....</b>	<b>28</b>
<b>5.4. Sprint Backlog.....</b>	<b>30</b>
<b>5.5. Dictionnaire des données .....</b>	<b>34</b>
<b>5.6. Règles de gestion .....</b>	<b>39</b>
<b>5.7. Modèle du domaine.....</b>	<b>40</b>
<b>5.8. Diagramme de classe de conception pour chaque sprint .....</b>	<b>41</b>
<b>5.9. Diagramme de classe de conception globale .....</b>	<b>45</b>
<b>5.10. Diagramme de paquetage.....</b>	<b>47</b>
<b>5.11. Diagramme de déploiement.....</b>	<b>49</b>
<b>PARTIE III. REALISATION .....</b>	<b>51</b>
<b>Chapitre 6. Mise en place de l'environnement de développement.....</b>	<b>52</b>
<b>6.1. Installation et configuration des outils .....</b>	<b>52</b>
<b>6.2. Architecture de l'application .....</b>	<b>55</b>
<b>Chapitre 7. Développement de l'application .....</b>	<b>56</b>

7.1. Création de la base de données .....	56
7.2. Codage de l'application .....	58
7.3. Présentation de l'application .....	64
CONCLUSION.....	XIII
BIBLIOGRAPHIE .....	XIV
WEBOGRAPHIE.....	XV
GLOSSAIRE.....	XVI
TABLE DES MATIERES .....	XVIII
RESUME.....	XXII
ABSTRACT .....	XXII

## REMERCIEMENTS

En premier lieu je voudrais remercier Dieu sans lequel ce projet ainsi que mes études n'auraient même pas eu lieu, je remercie Dieu d'avoir été mon pilier durant mes années d'étude et la source de ma détermination pour ce projet.

Je tiens également à remercier toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce mémoire à savoir :

- **Monsieur HAJALALAINA Aimé Richard**, Professeur et Président de l'Université de Fianarantsoa d'avoir bien organisé l'année universitaire ;
- **Monsieur MAHATODY Thomas**, Professeur, Directeur de l'Ecole Nationale d'Informatique de Fianarantsoa qui en m'acceptant au sein de l'école m'a donné l'opportunité d'avoir mes connaissances théoriques actuelles ;
- **Monsieur Daniel LEFEBRE**, Directeur Général de Ravinala Airports qui m'a permis d'accomplir ce stage au sein de l'entreprise qu'il dirige ;
- **Monsieur RABETAFIKA Louis Haja**, Maître de conférences, responsable de la mention informatique, qui a permis que cette année se passe sans encombre ;
- **Monsieur GILANTE Gesazafy**, assistant d'enseignement supérieur et de recherche, Responsable de parcours « informatique générale » en menant à bien cette année ;
- **Monsieur RALAIVAO Jean Christian**, Maître de conférences, pour le temps qu'il m'a accordé lors de mon encadrement, ainsi que pour ses évaluations approfondies et ses recommandations constructives qui ont grandement enrichi ce travail.
- **Monsieur Damien ANDRIANTSILAVINA**, Chef de département de transformation digitale mon encadreur professionnel qui m'a aidé et tutoré en m'inculquant ses expériences professionnelles durant la période de mon stage ;
- **Monsieur DIMBISOA William Germain**, Maître de conférences, pour l'honneur qu'il m'a fait en acceptant de présider le Jury de soutenance.
- **Madame FANOMEZANA Mihajasoana Lea**, Doctorante en informatique, pour l'honneur qu'elle me fait en acceptant d'être l'examinatrice lors de la soutenance et pour les remarques constructives qu'elle apportera.
- **Monsieur Dominique RAKOTOMAMONJY**, Directeur du service informatique de Ravinala Airports qui m'a permis de faire ce stage et m'a guidé durant ce stage ;
- Tous les membres du service informatique de Ravinala Airports qui n'ont cessé de me conseiller et de transmettre leurs savoirs ;



- Tout le personnel de Ravinala qui m'a accueilli avec joie et bonne humeur et m'a permis de travailler dans un environnement calme et chaleureux ;
- Les membres du corps professoral et administratif de l'Ecole Nationale d'Informatique pour toutes les connaissances acquises et savoirs durant ma formation théorique ;
- Ma famille qui durant toute cette période m'a soutenue aussi bien moralement que financièrement ;
- Toutes les personnes grâce auxquelles ce stage a pu se dérouler dans les bonnes conditions.

## LISTE DES FIGURES

Figure 1. Organigramme actuel de l'Ecole .....	5
Figure 2: Organigramme de Ravinala Airports .....	12
Figure 3. Planning de réalisation.....	17
Figure 4. Modèle de domaine.....	41
Figure 5. Diagramme de classe sprint 1 .....	42
Figure 6. Diagramme de classe du sprint 2 .....	43
Figure 7. Diagramme de classe du sprint 3 .....	44
Figure 8. Diagramme de classe du sprint 4 .....	45
Figure 9. Diagramme de classe global .....	47
Figure 10.Type de relation courantes d'un diagramme de paquetage.....	48
Figure 11. Diagramme de paquetage.....	49
Figure 12. Diagramme de déploiement .....	50
Figure 13. Interface d'installation de SQL server : Type d'installation.....	52
Figure 14. Interface d'Installation de SSMS .....	53
Figure 15. Interface de Visual Paradigm .....	53
Figure 16. Interface de VS code .....	54
Figure 17.Code de création de la table project.....	56
Figure 18. Code de création de la table user .....	56
Figure 19. Code de création des tables habilitation projet et habilitation intercontrat.....	57
Figure 20. Code de création des tables habilitation et habilitation admin .....	57
Figure 21. Code de création de la table audit .....	58
Figure 22. Code de création de la table task .....	58
Figure 23. Structure du coté backend.....	58
Figure 24. Extrait de code, population de la table user .....	60
Figure 25. Extrait de code : création d'une tache. ....	61
Figure 26. Extrait de code : déplacement de tâches en utilisant websocket.....	62
Figure 27. Structure du coté frontend.....	63
Figure 28. Interface login .....	64
Figure 29. Interface admin, liste des utilisateurs.....	65
Figure 30. Interface admin, liste des habilitations.....	65
Figure 31. Interface projet, table projet.....	66
Figure 32. Interface tâche par phase de projet.....	66
Figure 33. Détails d'un projet. ....	67

Figure 34. Vue calendaire des activités dans le mois courant. ....	67
Figure 35. Interface historique, mobile .....	68
Figure 36. Interface projet, mobile.....	68
Figure 37. Interface login mobile.....	68
Figure 38. Interface activité, mobile .....	68

## LISTE DES TABLEAUX

Tableau 1. Organisation du système de formation pédagogique de l'Ecole.....	6
Tableau 2. Mention et parcours au sein de l'ENI.....	6
Tableau 3. Liste des formations existantes à l'ENI.....	7
Tableau 4. Débouchés éventuels des jeunes diplômés .....	9
Tableau 5. Caractéristiques des moyens matériels .....	15
Tableau 6. Moyens logiciels.....	15
Tableau 7. Moyen humain .....	16
Tableau 8. Moyen matériel existant .....	19
Tableau 9. Moyen logiciel existant.....	20
Tableau 10. Comparaison des solutions proposées .....	21
Tableau 11. Comparaison des méthodes à utiliser.....	22
Tableau 12. Comparaison des outils à utiliser .....	25
Tableau 13. Product Backlog.....	29
Tableau 14. Liste de travail à faire .....	30
Tableau 15. Planification du sprint.....	32
Tableau 16. Dictionnaire de données .....	34

## LISTES DES ABREVIATIONS

<b>2TUP</b>	Two Tracks Unified Process
<b>ACI</b>	Airports Council International
<b>AD</b>	Active Directory
<b>ADP</b>	Aéroports de Paris
<b>API</b>	Application Programming Interface
<b>Ar</b>	Ariary
<b>AUF</b>	Agence Universitaire de la Francophonie
<b>B2B</b>	Business To Business
<b>BNGRC</b>	Bureau de gestion des Risques et des Catastrophes
<b>BTS</b>	Brevet de Technicien Supérieur
<b>CARI</b>	Colloque Africain sur la Recherche en Informatique
<b>CCNA</b>	Cisco Certified Network Associate
<b>CITEF</b>	Conférence Internationale des Ecoles de formation d'Ingénieurs et Technicien d'Expression Française
<b>CNAPS</b>	Caisse Nationale de Prévoyance Sociale
<b>CNH</b>	Commission Nationale d'Habilitation
<b>COFAV</b>	Corridor forestier de Fandriana jusqu'à Vondrozo
<b>CUR</b>	Centre Universitaire Régional
<b>DEA</b>	Diplôme d'Etudes Approfondies
<b>DSI</b>	Direction des systèmes d'information
<b>DUT</b>	Diplôme Universitaire de Technicien
<b>ENI</b>	Ecole Nationale d'informatique
<b>ESPA</b>	Ecole Supérieure Polytechnique d'Antananarivo

<b>EUR</b>	Euro
<b>GLPI</b>	Gestionnaire Libre de Parc Informatique
<b>Go</b>	Giga-octet
<b>HTTP</b>	Hypertext Transfer Protocol
<b>IRD</b>	Institut de Recherche pour le Développement
<b>IREMIA</b>	Institut de Recherche en Mathématiques et Informatique Appliquées
<b>ISO</b>	International Organization for Standardization
<b>JS</b>	JavaScript
<b>JSON</b>	JavaScript Object Notation
<b>LMD</b>	Licence – Master – Doctorat
<b>PFI</b>	Private Finance Initiative
<b>PRESUP</b>	Programme de renforcement en l’Enseignement Supérieur
<b>RAM</b>	Random Access Memory
<b>RG</b>	Règle de gestion
<b>RH</b>	Ressource humaine
<b>RSE</b>	Responsabilité Sociétale des Entreprises
<b>RSSI</b>	Responsable de la Sécurité des Systèmes d'Information
<b>SDK</b>	Software Development Kit
<b>SEPT</b>	Société d’Exploitation du Port de Toamasina
<b>SGBD</b>	Système de Gestion de Base de Données
<b>SGS</b>	Société Générale de Surveillance
<b>SSLIA</b>	Service de Sauvetage et de Lutte contre les Incendies d'Aviation.
<b>SSMS</b>	SQL Server Manager Studio
<b>SST</b>	Santé et Sécurité au Travail

<b>SSUA</b>	Study Start-Up Agreement
<b>SQL</b>	Structured Query Language
<b>TDS</b>	Tabular Data Stream
<b>TIC</b>	Technologies de l'information et de la communication
<b>UI</b>	User Interface
<b>UML</b>	Unified Modelling Language
<b>UPST</b>	Université Paul Sabatier de Toulouse
<b>VIP</b>	Very Important Person
<b>XP</b>	Extreme Programming

## **INTRODUCTION GENERALE**

Dans le monde professionnel actuel, la gestion efficace des projets et des ressources humaines est devenue un enjeu majeur pour les entreprises. La complexité croissante des projets, la multiplication des équipes et la nécessité d'optimiser l'allocation des ressources posent de nouveaux défis organisationnels. En effet, les entreprises doivent jongler entre la planification des projets, le suivi des délais, la gestion des disponibilités des collaborateurs et la coordination des différentes équipes. Pour répondre à ces besoins et optimiser la gestion de leurs projets et ressources, de nombreuses organisations cherchent à se doter d'outils performants et adaptés à leurs besoins spécifiques.

C'est dans ce contexte que s'inscrit le développement d'un outil de gestion de projet innovant, visant à offrir une vue d'ensemble claire et détaillée sur l'ensemble des projets en cours ainsi que sur les disponibilités des collaborateurs. Cet outil permettra non seulement d'avoir une vision globale du portefeuille de projets, mais aussi d'optimiser l'allocation des ressources en fonction des compétences et des disponibilités de chaque collaborateur.

Ce mémoire se structurera en trois grandes parties. La première partie sera consacrée à la présentation de l'école, de l'entreprise hôte ainsi que du contexte du projet et des enjeux organisationnels qu'il adresse. La seconde partie portera sur l'analyse et la conception détaillée de la solution, en mettant l'accent sur les choix technologiques et architecturaux. Enfin, la troisième partie se focalisera sur la réalisation concrète du projet, présentant les fonctionnalités développées et les résultats obtenus.



# **PARTIE I : PRESENTATIONS**

# **Chapitre 1. Présentation de l'Ecole Nationale d'Informatique**

## **1.1. Information d'ordre générale**

L'Ecole Nationale d'Informatique, en abrégé ENI, est un établissement d'enseignement supérieur rattaché académiquement et administrativement à l'Université de Fianarantsoa. Le siège de l'Ecole se trouve à Tanambao-Antaninarenina à Fianarantsoa. L'adresse pour la prise de contact avec l'Ecole est la suivante : Ecole Nationale d'Informatique (ENI) Tanambao, Fianarantsoa. Le numéro de sa boîte postale est 1487 avec le code postal 301. Téléphone : 034 05 733 36 ou 032 15 204 28. Son adresse électronique est la suivante : **eni@eni.mg**. Il dispose également d'un site web : **www.eni.mg**

## **1.2. Missions et historiques**

L'ENI se positionne sur l'échiquier socio-éducatif malgache comme étant le plus puissant secteur de diffusion et de vulgarisation des connaissances et des technologies informatiques.

Cette Ecole Supérieure peut être considérée aujourd'hui comme la vitrine et la pépinière des élites informaticiennes du pays.

De façon formelle, l'ENI était créée par le décret N° 83- 185 du 24 Mai 1983, comme étant le seul établissement Universitaire Professionnalisé au niveau national, destiné à former des techniciens et des Ingénieurs de haut niveau, aptes à répondre aux besoins et exigences d'Informatisation des entreprises, des sociétés et des organes implantés à Madagascar.

L'ENI a pour conséquent pour mission de former des spécialistes informaticiens compétents et opérationnels de différents niveaux notamment :

- en fournissant à des étudiants des connaissances de base en informatique ;
- en leur transmettant le savoir-faire requis, à travers la professionnalisation des formations dispensées et en essayant une meilleure adéquation des formations par rapport aux besoins évolutifs des sociétés et des entreprises ;
- en initiant les étudiants aux activités de recherche dans les différents domaines des Technologies de l'Information et de la Communication (TIC) ;

La filière de formation d'Analystes Programmeurs a été mise en place à l'Ecole en 1983, et a été gelée par la suite en 1996, tandis que la filière de formation d'ingénieurs a été ouverte à l'Ecole en 1986.

Une formation de troisième cycle a été ouverte à l'Ecole depuis l'année 2003 – 2004 grâce à la coopération académique et scientifique entre l'Université de Fianarantsoa pour le compte de l'ENI et l'Université Paul Sabatier de Toulouse (UPST).

Cette filière avait pour objectif de former certains étudiants à la recherche dans les différents domaines de l'Informatique, et notamment pour préparer la relève des Enseignants-Chercheurs qui étaient en poste.

Pendant l'année 2007-2008, la formation en vue de l'obtention du diplôme de Licence Professionnelle en Informatique a été mise en place à l'ENI avec les deux parcours de formation :

- Génie Logiciel et base de Données.
- Administration des Systèmes et réseaux.

La mise en place à l'Ecole de ces deux options de formation devait répondre au besoin de basculement vers le système Licence – Master – Doctorat (LMD).

En vue de surmonter les difficultés de limitation de l'effectif des étudiants accueillis à l'Ecole, notamment à cause du manque d'infrastructures, un système de « Formation Hybride » a été mis en place à partir de l'année 2010. Il s'agit en effet d'un système de formation semi-présentielle et à distance avec l'utilisation de la visioconférence pour la formation à distance. Le système de formation hybride a été ainsi créé à Fianarantsoa ainsi qu'à l'Université de Toliara. Cette formation est à l'origine du parcours Informatique Générale.

En 2023, une nouvelle mention Intelligence Artificielle (IA) a été ouverte au sein de l'Ecole pour répondre aux besoins des entreprises. La formation est destinée aux étudiants titulaires du diplôme de licence (Bac +3) en Mathématiques ou en Statistiques ou en Informatique, etc. La mention IA comporte deux parcours :

- Gouvernance et Ingénierie de Données (GID),
- Objets connectés et Cybersécurité (OCC).

Le principe de l'enseignement pour le parcours GID offre aux étudiants des compétences scientifiques et techniques spécialisées en Science de données. Pour le parcours OCC, les étudiants obtiennent la double spécialité premièrement en internet des objets et deuxièmement en cybersécurité. La formation de master est axée sur l'ensemble d'applications de l'Intelligence Artificielle.

### **1.3. Organigramme institutionnel**

L'organigramme de l'Ecole est inspiré des dispositions du décret N° 83-185 du 24 Mai 1983. L'ENI est administrée par un Conseil d'Ecole, et dirigée par un directeur nommé par un

décret adopté en Conseil des Ministres. Le Collège des enseignants regroupant tous les enseignants-chercheurs permanents de l'Ecole est chargé de résoudre les problèmes liés à l'organisation pédagogique des enseignements. Le Conseil Scientifique propose les orientations pédagogiques et scientifiques de l'établissement, en tenant compte notamment de l'évolution du marché de travail et de l'adéquation des formations dispensées par rapport aux besoins des entreprises. La figure 1 représente l'organigramme actuel de l'ENI.

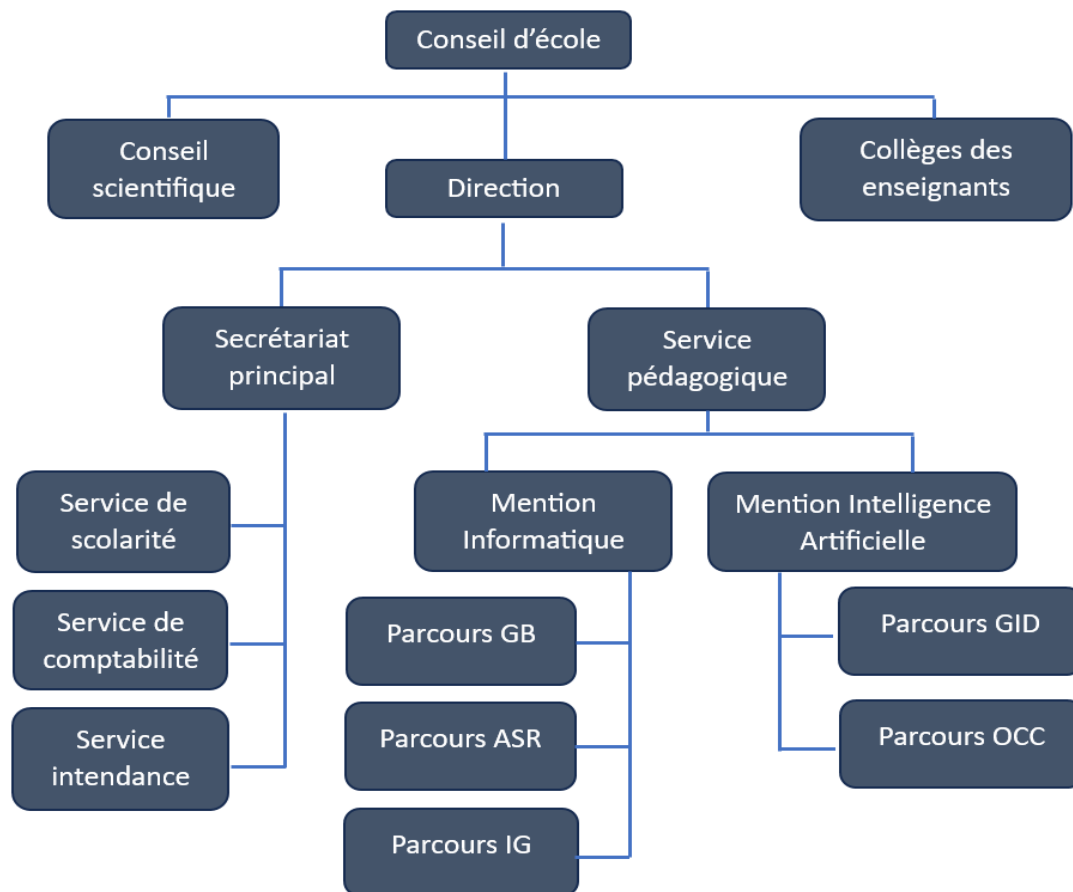


Figure 1. Organigramme actuel de l'Ecole

#### 1.4. Domaine de spécialisation

Les activités de formation et de recherche organisées à l'ENI portent sur les domaines suivants :

- Génie logiciel et Base de Données ;
- Administration des Systèmes et Réseaux ;
- Informatique Générale ;
- Modélisation informatique et mathématique des Systèmes complexes ;
- Intelligence artificielle.

Le tableau 1 décrit l'organisation du système de formation pédagogique de l'Ecole.

Tableau 1. Organisation du système de formation pédagogique de l'Ecole

Formation Théorique	Formation Pratique
<ul style="list-style-type: none"> <li>- Enseignement théorique</li> <li>- Travaux dirigés</li> <li>- Travaux pratiques</li> <li>- Conférences</li> </ul>	<ul style="list-style-type: none"> <li>- Etude de cas</li> <li>- Travaux de réalisation</li> <li>- Projets/ Projets tutorés</li> <li>- Voyages d'Etudes</li> <li>- Stages en entreprise</li> </ul>

### 1.5. Architecture des formations pédagogiques

Le recrutement des étudiants à l'ENI se fait uniquement par voie de concours d'envergure nationale en première année. Les offres de formation organisées à l'Ecole ont été validées par la Commission Nationale d'Habilitation (CNH). Au sein de l'ENI, il existe deux mentions et cinq parcours. Le tableau 2 récapitule les mentions et les parcours au sein de l'Ecole :

Tableau 2. Mention et parcours au sein de l'ENI

Mention	Parcours
<b>Informatique</b>	Génie logiciel et Base de Données (GB)
	Administration des Systèmes et Réseaux (ASR)
	Informatique Générale (IG)
<b>Intelligence Artificielle</b>	Gouvernance et Ingénierie de Données (GID)
	Objets Connectés et Cyber sécurités (OCC)

La figure 2 représente l'architecture des études correspondant au système LMD.

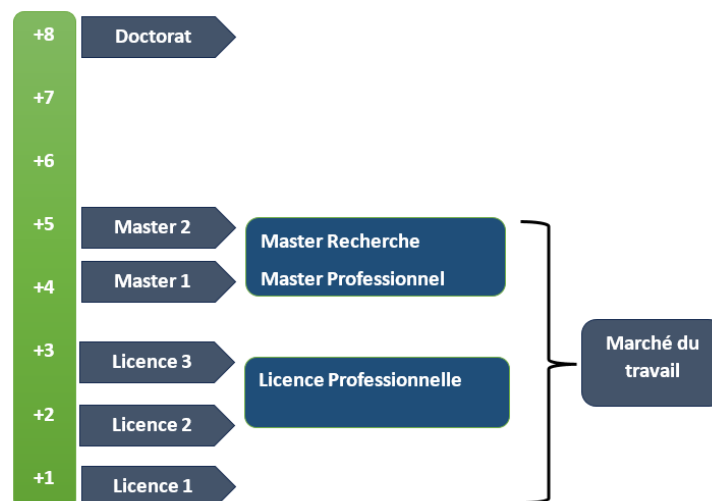


Figure 2. Architecture des études correspondant au système LMD

La licence peut avoir une vocation générale ou professionnelle. Le master peut avoir une vocation professionnelle ou de recherche. L'accès en première année de MASTER se fait automatiquement pour les étudiants de l'Ecole qui ont obtenu le diplôme de Licence Professionnelle. Le tableau 3 illustre la liste des formations existantes à l'ENI.

Tableau 3. Liste des formations existantes à l'ENI

	FORMATION	
	LICENCE PROFESSIONNELLE	MASTER
Condition admission	Par voie de concours	Par voie de concours pour la mention IA
Condition d'accès	Bac de série C, D ou Technique	Être titulaire de licence professionnelle
Durée de Formation	3 ans	2 ans
Diplôme délivré	Diplôme de Licence Professionnelle	Diplôme de Master Professionnel Diplôme de Master Recherche

Le Master Recherche permet à son titulaire de poursuivre directement des études en doctorat et de s'inscrire directement dans une Ecole Doctorale.

Les étudiants diplômés de l'Ecole sont plutôt bien accueillis dans les instituts universitaires étrangères (Canada, Suisse, France, ...)

#### **1.6. Relation de l'ENI avec les organismes externes**

Les stages effectués chaque année par les étudiants mettent l'Ecole en rapport permanent avec plus de 400 entreprises et organismes publics, semi-publics et privés, nationaux et internationaux. L'Ecole dispose ainsi d'un réseau d'entreprises, de sociétés et d'organismes publics et privés qui sont des partenaires par l'accueil en stage de ses étudiants, et éventuellement pour le recrutement après l'obtention des diplômes par ces derniers. Les compétences que l'Ecole cherche à développer chez ses étudiants sont l'adaptabilité, le sens de la responsabilité, du travail en équipe, le goût de l'expérimentation et l'innovation.

En effet, la vocation de l'ENI est de former des licenciés et des ingénieurs de niveau MASTER avec des qualités scientifiques, techniques et humaines reconnues, capables d'évoluer professionnellement dans des secteurs d'activité variés intégrant l'informatique. Les stages en milieu professionnel permettent de favoriser une meilleure adéquation entre les formations à l'Ecole et les besoins évolutifs du marché de l'emploi.

Parmi les sociétés, les entreprises et les organismes partenaires de l'Ecole, on peut citer : ACCENTURE Mauritius, AKATA Goavana, Air Madagascar, Ambre Associates, Airtel, Agence Universitaire de la Francophonie (AUF), AXIAN, B2B, Banque Centrale, , BIANCO, BlueLine, CNaPS, Bureau National de Gestion des Risques et des Catastrophes (BNGRC),

CEDII-Fianarantsoa, Data Consulting, Central Test, Centre National Antiacridien, CNRE, COLAS, Direction Générale des Douanes, DLC, E-Tech Consulting, , FID, FIHARY Soft, FTM, GNOSYS, GENIUS AT WORK, Hello Tana, IBONIA, INGENOSIA, INSTAT, IOGA, JIRAMA, JOUVE, MADADEV, MAEP, MANAO, MEF, MEN, MESupRES, MFB, , MININTER, Min des Postes/Télécommunications et du Développement Numérique, NEOV MAD, Ny Havana, Madagascar National Parks, OMNITEC, ORANGE, OTME, PRACCESS, QMM Fort-Dauphin, SG Madagasikara SMMC, SMMEC, SNEDADRS Antsirabe, Sénat, Société d'Exploitation du Port de Toamasina (SEPT), SOFTWELL, Strategy Consulting, TELMA, VIVETEC, Société LAZAN'I BETSILEO, WWF, UGD, ARATO, MANAO, MNDPT, NG ACADEMY.NG, Relia ...

### **1.7. Débouchés professionnels et diplômés**

Les formations proposées par l'Ecole permettent aux diplômés d'être immédiatement opérationnels sur le marché du travail avec la connaissance d'un métier complet lié à l'informatique aux TIC.

L'Ecole apporte à ses étudiants un savoir-faire et un savoir-être qui les accompagnent tout au long de leur vie professionnelle. Elle a une vocation professionnalisante. Les diplômés en LICENCE et en MASTER issus de l'ENI peuvent faire carrière dans différents secteurs.

L'Ecole bénéficie aujourd'hui de 40 années d'expériences pédagogiques et de reconnaissance auprès des sociétés, des entreprises et des organismes. C'est une Ecole Supérieure de référence en matière informatique.

D'une manière générale, les diplômés de l'ENI n'éprouvent pas de difficultés particulières à être recrutés au terme de leurs études. Cependant, l'ENI recommande à ses diplômés de promouvoir l'entrepreneuriat en TIC et de créer des cybercafés, des SSII ou des bureaux d'études. Le tableau 4 représente les débouchés éventuels des jeunes diplômés.

Tableau 4. Débouchés éventuels des jeunes diplômés

LICENCE	MASTER
<ul style="list-style-type: none"> <li>➤ Analyste - Programmeur</li> <li>➤ Administrateur de site web/de portail web</li> <li>➤ Assistant Informatique et internet</li> <li>➤ Chef de projet web ou multimédia</li> <li>➤ Développeur Informatique ou multimédia</li> <li>➤ Intégrateur web ou web designer</li> <li>➤ Hot liner/Hébergeur Internet</li> <li>➤ Agent de référencement</li> <li>➤ Technicien/Supérieur de help desk sur Informatique</li> <li>➤ Responsable de sécurité web</li> <li>➤ Administrateur de réseau</li> </ul>	<ul style="list-style-type: none"> <li>➤ Administrateur de réseau et système</li> <li>➤ Architecture de système d'information</li> <li>➤ Développeur d'applications</li> <li>➤ Ingénieur réseau</li> <li>➤ Webmaster / Web Designer</li> <li>➤ Concepteur et réalisateur d'application</li> <li>➤ Directeur du système d'informations</li> <li>➤ Chef de projet informatique</li> <li>➤ Responsable de sécurité informatique</li> <li>➤ Consultant fonctionnel ou freelance</li> </ul>

### 1.8. Ressources humaines

Les ressources humaines sont citées ci-dessous selon leurs responsabilités :

- Directeur de l'Ecole : Monsieur MAHATODY Thomas, Docteur HDR
- Responsable de la Mention « Informatique » : Monsieur RABETAFIKA Louis Haja, Maître de Conférences
- Responsable de la Mention « Intelligence Artificielle » : Monsieur DIMBISOA William Germain, Maître de Conférences
- Responsable du Parcours « Génie Logiciel et Base de Données » : Monsieur RALAIVAO Jean Christian, Assistant d'Enseignement Supérieur et de Recherche
- Responsable du Parcours « Administration Systèmes et Réseaux » : Monsieur SIAKA, Assistant d'Enseignement Supérieur et de Recherche
- Responsable du Parcours « Informatique Générale » : Monsieur GILANTE Gesazafy, Assistant d'Enseignement Supérieur et de Recherche
- Responsable du Parcours « Gouvernance et Ingénierie de Données » : Madame RATIANANTITRA Volatiana Marielle, Maître de Conférences
- Responsable du Parcours « Objets Connectés et Cybersécurité » : Monsieur RAZAFIMAHATRATRA Hajarisena, Maître de Conférences

L'ENI compte quinze (15) enseignants permanents dont un (01) Professeur Titulaire, un (01) Professeur, un (01) Docteur HDR, huit (08) Maîtres de Conférences, quatre (04) Assistants d'Enseignement Supérieur et de Recherche, dix (10) enseignants vacataires, quarante un (41) personnel administratif.



## **Chapitre 2 : Présentation de Ravinala Airports**

Dans ce chapitre, on présentera la société d'accueil en marquant ses valeurs, ses missions et les informations importantes la concernant.

### **2.1. Information d'ordre général**

Suite à un appel d'offre réalisé par l'Etat Malagasy, Ravinala Airports a été sélectionné pour la concession des deux aéroports internationaux de Madagascar. Ravinala Airports est le gestionnaire des aéroports internationaux d'Antananarivo et de Nosy Be et a comme vision de devenir un opérateur aéroportuaire de référence dans l'Océan Indien d'ici 2025

Nos missions sont axées sur le développement de la capacité de ces deux aéroports internationaux, de réaménager les installations existantes, de les maintenir aux normes de sécurité et de protection de l'environnement, d'améliorer la qualité des services aéroportuaires, de satisfaire les attentes de tous ses clients (compagnies aériennes, passagers, commerces, fret, Aviation Générale) afin d'accroître leur satisfaction de manière durable, et participer au développement économique de Madagascar grâce à sa stratégie de développement des routes aériennes au départ ou à destination de Madagascar. A travers sa stratégie RSE, Ravinala Airports apporte un impact positif considérable dans les villes où sont implantés les aéroports qu'elle exploite.

A compter de décembre 2016 pour une durée de 28 ans, Ravinala Airports a pour mission de gérer, rénover, et développer les aéroports d'Antananarivo et Nosy Be.

### **2.2. Nos valeurs**

Nos valeurs sont la base de notre raison d'être. Elles s'expriment à travers le mot BASE

- **BIENVEILLANCE** : elle s'exprime par le respect mutuel, l'écoute, l'empathie, la reconnaissance des actions accomplies et la considération de chaque individu interne et externe.
- **AMBITION** : nous sommes une entreprise ambitieuse, agile et dynamique qui est dans une démarche d'amélioration continue par l'innovation. Nous aspirons toujours au meilleur et sommes exigeants, dans le cadre du développement de l'entreprise et de Madagascar.
- **SENS DU SERVICE** : nous cherchons à la satisfaction continue de nos clients internes et externes par la pratique de l'écoute active, par notre réactivité et notre capacité

d'anticiper leurs besoins pour fournir un service de qualité conforme aux normes internationales.

- **ENGAGEMENT** : nous reconnaissons l'intérêt de toutes nos parties prenantes. Nous les mobilisons à cocréer de la valeur de façon transparente

### **2.3. Nos missions**

Les principales missions de Ravinala Airports sont de financer, construire, rénover, maintenir et développer les deux aéroports internationaux d'Antananarivo et de Nosy Be afin de contribuer à l'essor économique de Madagascar, et en particulier :

- La gestion et le développement des infrastructures des aéroports d'Antananarivo et de Nosy Be (terminaux, pistes, parking ...) aux meilleurs standards internationaux ;
- La fourniture de tous les services aéroportuaires à l'exception des activités dites « réservées » (contrôle aérien, sûreté, assistance en escale, fuel et SSLIA) ;
- Le financement, la conception et la réalisation du programme d'investissements
- L'entretien, la maintenance et le renouvellement de tous les biens de la concession
- Le développement du trafic aérien et de nouvelles routes aériennes au départ des deux aéroports

### **2.4. Dates clés**

- 2016 : Transfert de la gestion des aéroports internationaux d'Antananarivo et de Nosy Be à Ravinala Airports.
- 2017 : Début de la construction du Nouveau Terminal International d'Antananarivo.
- 2018 : Fin des travaux de rénovation des pistes des Aéroports internationaux d'Antananarivo et de Nosy Be.
- 2021 : Mise en service commerciale du nouveau terminal international de l'aéroport d'Antananarivo.

### **2.5. Organigramme de Ravinala Airports**

Ravinala Airports est administré par un directeur général, avec 11 collaborateurs sous son autorité chacun dans un des 9 Départements qui composent Ravinala Airports.

Chaque département possède leurs propres hiérarchie et système de sous départements. La figure 2 représente la hiérarchie globale au sein de Ravinala.

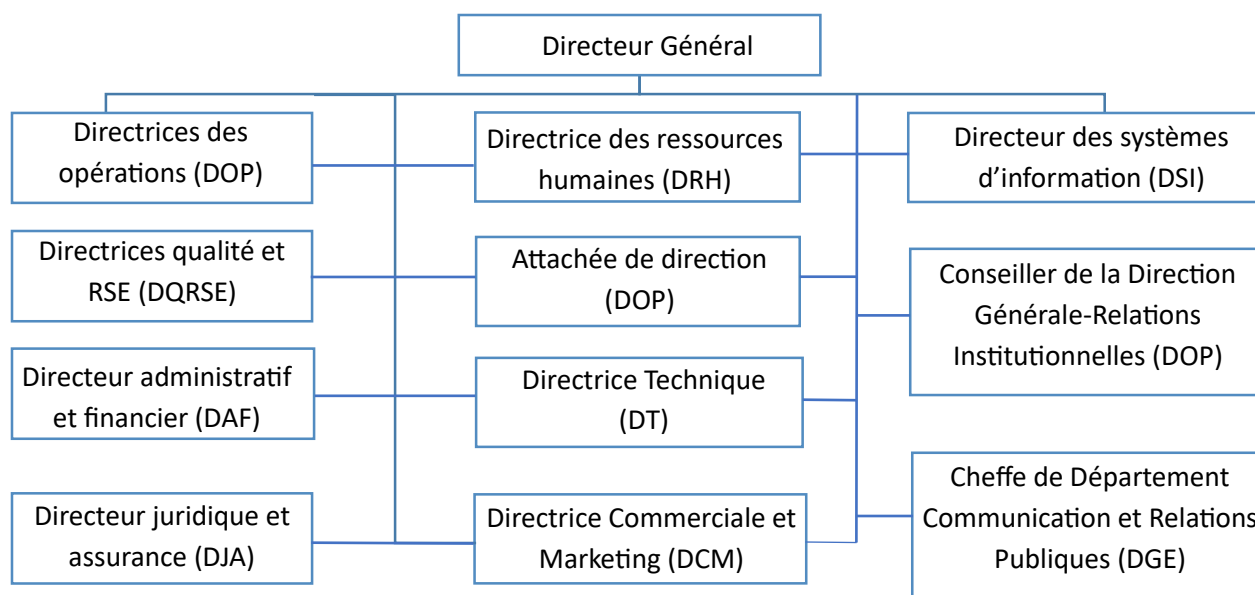


Figure 2: Organigramme de Ravinala Airports

## 2.6. Nos prix et certifications

- **Prix du « Meilleur aéroport de moins de 2 millions de passagers en Afrique en 2023 »** octroyé par l'ACI World et choisi par les passagers.
- **Excellence SGS 2022 de l'ACI Afrique** : Reconnaissance de l'excellence et du professionnalisme dans la gestion de la sécurité des aérodrômes.
- **Routes Africa 2019** : Reconnaissance de l'excellence du marketing aéroportuaire dans la catégorie des aéroports moins de 4 millions de passagers.
- **PFI Awards 2017** : Certification sur la qualité du projet.
- **ISO9001 :2015 / ISO14001:2015** : Qualité de service/ Environnement.
- **EXOVA** : Certification du système de sécurité incendie.
- **ACI Health Certification** : Respect des mesures et l'efficacité des dispositifs sanitaires dans l'aéroport sont en accord avec les normes de l'ICAO.
- **ACI RH** : Première place du prix d'excellence en Ressources Humaines pour les aéroports de taille moyenne.
- **Edge** : Certification environnementale.
- **ACI Environment & Sustainable Development Award**
- **ACA Niveau 1** : Gestion des émissions de Carbone.
- **Rising Sta Award.**
- **Cerification CNAPS** : SST FEHIZORO Santé et Sécurité au Travail.

## **2.7. Nos actionnaires**

### **2.7.1. Meridiam (65%)**

Spécialiste des infrastructures de demain dans les pays développés et émergents d'Europe, d'Afrique et d'Amérique. Meridiam contribue à la réalisation des objectifs de développement durable des Nations Unies en vue d'améliorer la qualité de vie des populations.

Il conçoit, finance et gère des infrastructures durables et résilientes sur le long terme. Il soutient et accélère le développement de PME innovantes, ambitieuses et à forte croissance impliquées dans la transition écologique.

### **2.7.2. Groupe ADP (35%)**

Leader mondial de l'exploitation d'aéroports. Le groupe s'appuie à la fois sur un réseau unique de 27 aéroports dans le monde à la fin de 2022 en contrat de gestion ou en concession et sur un vivier de compétences et de savoir-faire mis au service des marchés locaux. A Madagascar, le groupe réalise des transferts de connaissance aux équipes locales afin d'atteindre la vision de l'entreprise de devenir un opérateur aéroportuaire de référence dans l'Océan Indien.

## Chapitre 3 : Description du projet

Ce chapitre détaillera un peu plus sur les détails du projet.

### 3.1. Formulations

Un outil de gestion de projet est un outil permettant de faciliter le suivi des ressources, budget, phases et tâches d'un projet.

### 3.2. Objectif et besoin de l'utilisateur

#### 3.2.1. Objectif

Les objectifs principaux de la mise en place d'un outil de gestion de projet pour Ravinala Airports, dans un contexte de transformation digitale.

- **Amélioration de la Planification et du Suivi des Projets** : Centraliser les informations pour assurer un suivi en temps réel, faciliter la planification des tâches, des ressources, et des délais, et ainsi anticiper les retards ou les risques.
- **Collaboration Optimisée** : Offrir une plateforme unique où toutes les parties prenantes (internes et plus tard externes) peuvent collaborer et suivre l'avancement des projets en temps réel, ce qui améliore la communication et l'alignement entre les équipes.
- **Meilleure Gestion des Ressources** : Optimiser l'allocation des ressources humaines et matérielles pour les différents projets, en évitant les surcharges de travail et en assurant la disponibilité des équipes sur les projets prioritaires.
- **Standardisation des Processus** : Harmoniser les pratiques de gestion de projet à travers des méthodologies et des outils communs, permettant ainsi de garantir une cohérence dans l'exécution des projets, en particulier pour ceux liés à la transformation digitale et à l'innovation.
- **Réduction des Coûts et Meilleure Rentabilité** : En limitant les erreurs, en optimisant les délais et en améliorant l'efficacité globale, l'outil contribue à un meilleur contrôle des coûts et à une meilleure rentabilité des projets.

#### 3.2.2. Besoins de l'utilisateur

##### a) Besoins fonctionnels

Le système doit offrir une visibilité en temps réel sur la disponibilité des équipes et des ressources mobilisées, permettre le suivi de la durée estimée et de l'avancement des projets, notifier les retards ou avances sur les délais, fournir un aperçu clair des tâches assignées et des priorités des projets, tout en facilitant l'identification des périodes propices aux congés. Il doit

également permettre au Directeur des Systèmes Informatiques de suivre la productivité des membres de son équipe et de consulter leur disponibilité pour l'attribution de nouveaux projets.

### **b) Besoins non-fonctionnel**

Le projet doit être synchronisé avec Office 365 (Teams, Outlook, etc.), offrir un contrôle d'accès basé sur les rôles, intégrer une authentification LDAP pour les utilisateurs internes et garantir un suivi des actions via un système de journalisation et d'audit. Il doit assurer une disponibilité minimale de 99,9 %, avec des maintenances planifiées et communiquées à l'avance. Accessible sur plusieurs plateformes et navigateurs, il doit disposer d'une interface ergonomique adaptée à tous les profils, y compris les utilisateurs ayant des besoins spécifiques. Les performances doivent inclure un temps de réponse rapide et une scalabilité optimale. Enfin, l'outil doit être fiable grâce à des sauvegardes régulières, une rétention des données de 30 jours minimum et des tests rigoureux avant chaque mise à jour.

## **3.3. Moyen nécessaire à la réalisation du projet**

### **3.3.1. Moyen matériel**

Le tableau 5 décrit les caractéristiques des moyens matériels nécessaire pour la réalisation du projet :

*Tableau 5. Caractéristiques des moyens matériels*

Système d'exploitation	RAM	Capacité disque dur	Processeur
Windows 10	8Go	500Go	Corei5 13500T
Windows server	Minimum 8Go	Minimum 60Go	2 Cœurs

### **3.3.2. Moyen logiciel**

Le tableau 6 montre les moyens logiciels utilisés durant e développement de l'application.

*Tableau 6. Moyens logiciels*

Logiciel	Version	Utilisation
Visual studio Code	1.95.2	Développement de l'application

Visual paradigm	17.2	Conception et architecture
SQLServeur	16.0.1000.6	Gestion de la base de données
Git	2.45.2.windows.1	Gestion des versions et avancements dans le projet

### 3.3.3. Moyen humain

Le tableau 7 montre les moyens humains intervenant dans la réalisation de cet outil.

*Tableau 7. Moyen humain*

Fonction	Nombre	Rôle
DSI	1	Valideur final
Chef de département	1	Concepteur, et premier valideur
RSSI	1	Valideur de l'architecture coté sécurité.
Chef de projet	1	Initiateur du projet, concepteur et valideur des avancées.
Développeur	1	Concepteur et développeur de l'outil

### 3.4. Résultats attendus

L'outil à développer doit être déployé, testé et adopté par toutes les directions au sein de Ravinala Airports permettant ainsi de standardiser la gestion de projet au sein de l'organisation.

### 3.5. Chronogramme de travail

Avant la réalisation et la conception, on s'est fixé un planning à suivre pour organiser le travail à faire.

La figure 3 représente le planning suivi lors de la réalisation de ce projet.

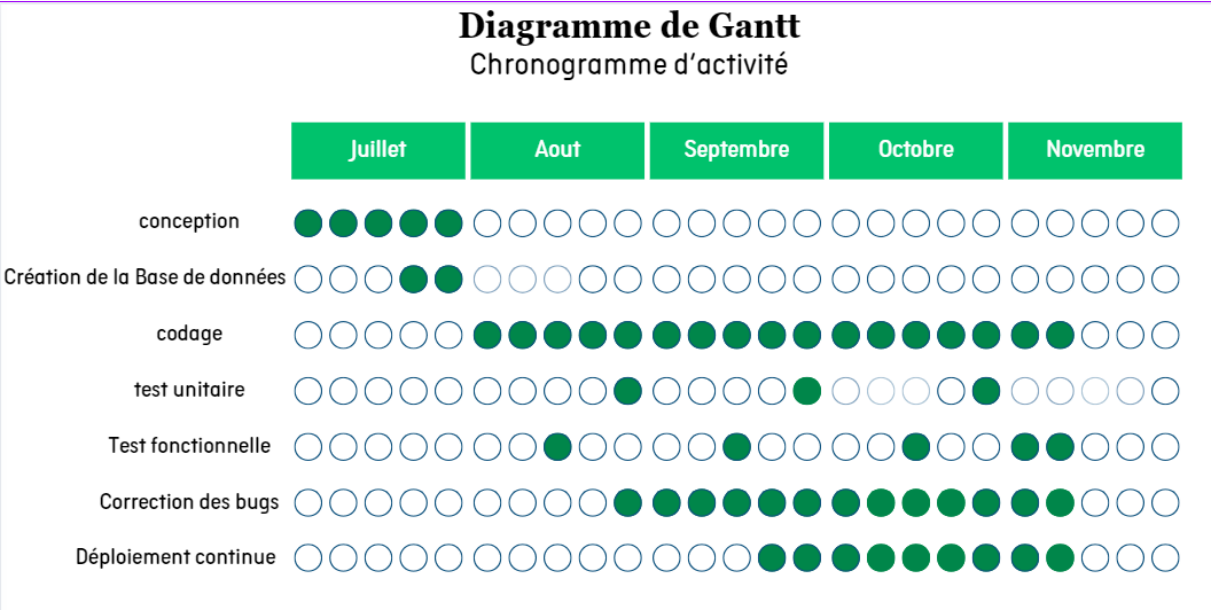


Figure 3. Planning de réalisation



## **PARTIE II : ANALYSE ET CONCEPTION**

## Chapitre 4 : Analyse préalable

Ce chapitre examine la situation actuelle pour identifier les besoins, les défis et les opportunités. Il inclut une analyse approfondie de l'existant ainsi que la phase de conception préliminaire du projet.

### 4.1. Analyse de l'existant

Avant de concevoir et de réaliser le projet, il est primordial de faire une analyse de l'existant. L'analyse de l'existant consiste à étudier la description des ressources technologiques et les matériels informatiques au sein de l'entreprise

#### 4.1.1. Organisation actuelle

Actuellement, chaque département gère ses projets de manière autonome, en utilisant des outils variés tels que Planner ou Loop de Microsoft. Le suivi de l'avancement des projets repose principalement sur des réunions hebdomadaires, où les équipes présentent des tâches réalisées, des éventuels retards observés, ainsi que des actions à mener pour la semaine suivante.

#### 4.1.2. Inventaire des moyens matériels et logiciels

Pour bien avancé dans la réalisation de ce projet, il est tout aussi essentiel de connaître les moyens matériels et logiciels utilisé en ce moment.

En plus de Planner et de Loop le département des systèmes d'information utilise également GLPI pour gérer efficacement les doléances venant de toutes les directions

Le tableau 8 résume les moyens matériels utilisé pendant le développement du projet

*Tableau 8. Moyen matériel existant*

Système d'exploitation	RAM	Capacité disque dur	Processeur
Windows 11	16 Go	500Go	Corei5 13500T

Le tableau 9 résume les moyens logiciels utilisé en ce moment par les directions au sein de Ravinala pour gérer leurs projets.

Tableau 9. Moyen logiciel existant

Logiciel	Version
GLPI	10.0.6
Loop	Office 365
Planner	Office 365

## 4.2. Critique de l'existant

### Points forts :

- **Flexibilité organisationnelle** : L'organisation actuelle permet une gestion flexible des projets au sein de chaque département, favorisant une adaptation locale et une autonomie dans les processus.

### Points faibles :

- **Manque de centralisation** : L'utilisation de méthodes et d'outils disparates crée un cloisonnement des informations, rendant difficile une vision globale.
- **Visibilité limitée** : Les cadres et le directeur peinent à accéder à une vue d'ensemble sur les disponibilités des collaborateurs, freinant la prise de décisions stratégiques.
- **Suivi fragmenté** : La fragmentation des outils complique l'accès à une liste consolidée des projets en cours et entrave un suivi cohérent de leur avancement dans les différents départements.

## 4.3. Conception avant-projet

### 4.3.1. Proposition de solutions

Après analyse il existe plusieurs solutions qui peuvent plus ou moins répondre aux attentes de l'entreprise :

Solution 1 : Utiliser Microsoft Project pro.

Solution 2 : Utiliser la partie gestion de projet dans GLPI

Solution 3 : Créer un outil de gestion et de management de projet

Le tableau 10 compare les solutions proposées

*Tableau 10. Comparaison des solutions proposées*

SOLUTION	AVANTAGES	INCONVENIENTS
Solution 1	<ul style="list-style-type: none"> <li>- Disponible de suite</li> <li>- Facile à réaliser</li> </ul>	<ul style="list-style-type: none"> <li>- Acheter une autre licence incluant Microsoft Project pro pour les employés de Ravinala.</li> </ul>
Solution 2	<ul style="list-style-type: none"> <li>- Disponible de suite</li> <li>- Ravinala dispose déjà de licence GLPI</li> </ul>	<ul style="list-style-type: none"> <li>- Trop de données inutiles.</li> <li>- Difficilement utilisable pour les utilisateurs.</li> </ul>
Solution 3	<ul style="list-style-type: none"> <li>- Fonctionnalité changeable dépendant des demandes de l'entreprise.</li> <li>- Gratuit.</li> <li>- Facilement maintenable.</li> <li>- Répond aux besoins spécifiques de l'organisation</li> </ul>	<ul style="list-style-type: none"> <li>- Prend le temps de développement avant de pouvoir utiliser.</li> </ul>

Nous avons opté pour la **Solution trois (3)** puisqu'il répond le plus aux demandes et attentes de l'entreprise.

#### **4.3.2. Méthodes de conception et outils utilisés**

##### **a) Méthode de conception**

Il existe plusieurs méthodes de conception, mais notre choix se décidera entre 2TUP, MERISE et AGILE.

Le tableau 11 compare les méthode Merise, 2TUP et AGILE

Tableau 11. Comparaison des méthodes à utiliser

Méthode	Avantages	Inconvénients
MERISE	<ul style="list-style-type: none"> <li>- Bonne gestion des données et des processus</li> <li>- Méthodologie structurée et adaptée aux systèmes d'information</li> <li>- Idéal pour des besoins de base de données complexes</li> </ul>	<ul style="list-style-type: none"> <li>- Méthode essentiellement française.</li> <li>- Méthode lourder et rigide.</li> <li>- Difficile à adapter pour les projets à changements fréquents.</li> </ul>
2TUP	<ul style="list-style-type: none"> <li>- Processus piloté par le langage de modélisation UML qui es un support de communication performant.</li> <li>- Bonne traçabilité des exigences.</li> <li>- Modulaire et adaptable</li> <li>- Approche orienté objet qui favorise la réutilisation des composants.</li> </ul>	<ul style="list-style-type: none"> <li>- La conception prend du temps.</li> <li>- Complexe à mettre en œuvre.</li> <li>- Moins flexible face aux changements fréquents.</li> </ul>
AGILE	<ul style="list-style-type: none"> <li>- Très flexible et adaptable.</li> <li>- Permet une livraison rapide de nouvelles fonctionnalités</li> <li>- Favorise la communication et l'adaptation raide aux besoins utilisateurs.</li> </ul>	<ul style="list-style-type: none"> <li>- Exige une implication active des parties prenantes et des utilisateurs.</li> <li>- Exige une bonne discipline dans l'équipe.</li> </ul>

Puisque l'outil de gestion de projet à concevoir nécessite une adaptabilité à chaque département, et nécessite des livrables fréquents ainsi que des possibles altérations tant au niveau du design qu'au niveau de la conception de la base de données, la méthode la plus appropriée dans notre cas est la méthode Agile.

La méthode **Agile** est une approche de gestion de projet axée sur la collaboration [1], l'adaptabilité et la livraison continue de valeur. Elle a été développée pour permettre aux équipes

de mieux répondre aux besoins changeants et aux imprévus, notamment dans les environnements technologiques dynamiques où les exigences peuvent évoluer rapidement.

### **Principes Fondamentaux de la Méthode Agile**

Agile repose sur quatre valeurs et douze principes décrits dans le Manifeste Agile, publié en 2001. Voici les quatre valeurs clés :

- **Individus et interactions** plutôt que processus et outils
- **Logiciel fonctionnel** plutôt que documentation exhaustive
- **Collaboration avec le client** plutôt que négociation de contrat
- **Réponse au changement** plutôt que suivi d'un plan

Ces valeurs illustrent l'importance de la communication et de l'adaptabilité pour maximiser la satisfaction du client et livrer de la valeur en continu.

### **Les Cadres Agile : Scrum, Kanban et XP**

Agile n'est pas une méthode unique, mais une approche générale regroupant différents cadres, comme :

- **Scrum** : Organisation en sprints (cycles courts de 1 à 4 semaines) et basée sur des rôles définis (Product Owner, Scrum Master, Équipe de développement).
- **Kanban** : Méthode visuelle basée sur un tableau de tâches (tableau Kanban), où les tâches sont déplacées dans des colonnes représentant différentes étapes de travail.
- **XP (Extreme Programming)** : Se concentre sur des pratiques de développement spécifiques, comme les revues de code, le développement itératif, et les tests fréquents.

### **Cycle de Vie Agile**

Le cycle de vie d'un projet Agile se divise en plusieurs étapes clés, qui sont répétées de manière itérative :

- **Planification initiale** : Définition des objectifs globaux et vision du produit.
- **Décomposition en itérations (ou sprints)** : Découpage du travail en itérations courtes (typiquement 1 à 4 semaines) permettant une livraison régulière de fonctionnalités.

- **Réunions Scrum (Daily Stand-up, Sprint Review, Sprint Retrospective) :** Évaluations régulières pour ajuster le produit et le processus.
- **Livraison de l'incrément :** À la fin de chaque itération, un incrément du produit est présenté au client, qui peut donner des retours permettant d'ajuster le projet pour la prochaine itération.

### Les Rôles dans une Équipe Agile [2]

- **Product Owner :** Représente les parties prenantes et priorise les fonctionnalités selon les besoins du client.
- **Scrum Master :** Facilite le processus Agile, élimine les obstacles, et s'assure que l'équipe respecte les pratiques Scrum.
- **Équipe de Développement :** Conçoit, développe, et teste les fonctionnalités.

### Les Artefacts et Événements Agile

- **Backlog Produit :** Liste des fonctionnalités et exigences priorisées par le Product Owner.
- **Backlog de Sprint :** Sélection des éléments du Backlog Produit à réaliser lors d'un sprint.
- **Daily Stand-up :** Réunion quotidienne pour synchroniser les activités et identifier les obstacles.
- **Sprint Review :** Réunion à la fin de chaque sprint pour présenter l'incrément au client.
- **Sprint Rétrospective :** Évaluation interne de l'équipe pour identifier les améliorations.

## b) Langage de programmation et outil utilisé

Le tableau 12 compare les outils et langage utilisé.

Tableau 12. Comparaison des outils à utiliser

Langages et outils	Description	Avantages	Inconvénients
Power apps + .NET	Power apps est un outil de développement no code faisant partie de Microsoft office 365.	<ul style="list-style-type: none"><li>- Déploiement facile et rapide</li><li>- Compatibilité avec office 365.</li></ul>	<ul style="list-style-type: none"><li>- Prix de la licence est de 20\$/utilisateur/mois ou de 5\$/application/utilisateur</li></ul>
React js + .NET		<ul style="list-style-type: none"><li>- Totalement gratuit.</li><li>- Léger et facile à modifier.</li><li>- Compatible avec office 365.</li></ul>	<ul style="list-style-type: none"><li>- Nécessite un serveur de déploiement.</li></ul>
Full .NET		<ul style="list-style-type: none"><li>- Totalement gratuit.</li><li>- Compatible avec office 365.</li></ul>	<ul style="list-style-type: none"><li>- Nécessite un serveur de déploiement.</li><li>- Le coté front end est lourd et pas facile a maintenir.</li></ul>

On a décidé d'utiliser React js en front end et .NET en back end. Puisque Ravinala Airports se focalise essentiellement sur les outils Microsoft alors .Net est le meilleur choix, mais en utilisant React en front on a une application plus légère et maintenable.

Quant à la base de données, Ravinala Airports utilise SQL server sur toutes les applications déjà présente alors nous avons opté pour la conformité.



## Chapitre 5. Analyse conceptuelle

Ce chapitre propose une exploration approfondie des concepts, idées et théories fondamentaux, appuyée par l'analyse de divers diagrammes pour offrir une compréhension claire et structurée.

### 5.1. Désignation des rôles de l'équipe SCRUM

Dans Scrum, chaque membre de l'équipe joue un rôle bien défini pour assurer la réussite du projet. Dans ce sens, les personnes impliquées dans ce projet jouent les rôles suivants :

- Product Owner : ce rôle est tenu par le chef de département.
- Scrum master : ce rôle est tenu par le chef de projet
- Equipe de développement : ce rôle est tenu par le stagiaire qui code et livre les fonctionnalités, les membres du RSSI qui testent le code et le produit fini.

### 5.2. Etapes de l'élaboration du Product Backlog

Le Product Backlog est une liste priorisée de toutes les fonctionnalités, modifications et corrections potentielles pour le produit [3]. Voici les étapes de son élaboration :

#### 5.2.1. Etablir la vision du projet

Il s'agit de définir l'objectif que le projet devra atteindre avant l'arrêt de développement. Il est important que cette vision soit acceptée et comprise de tous.

La vision pour cet outil de gestion est d'offrir un outil qui permette la gestion des projets ainsi que leurs budget et ressources respectives, aussi d'avoir une visibilité sur les activités des employés.

#### 5.2.2. Lister les acteurs

Cette étape identifie et détaille tous les intervenants, utilisateurs du système dans tous les aspects. Pour chaque intervenant il faudra préciser le rôle, la description, la fréquence d'utilisation.

- *Acteur 1*

- Rôle : Admin
- Description : Créer les habilitations des utilisateurs, assigner les habilitations aux utilisateurs. L'admin est celui qui décide de ce qu'une personne peut faire sur l'outil.
- Fréquence d'utilisation : Occasionnelle.
- Niveau de connaissance technologique : élevée
- Niveau de connaissance métier : élevée

- *Acteur 2*
  - Rôle : Directeur
  - Description : Regarder les avancées de tous les projets au sein de l'entreprise, regarder les taux de performances des employés et des départements. Le directeur a besoin de savoir ce que font les employés et ce que fait chaque département.
  - Fréquence d'utilisation : Occasionnelle
  - Niveau de connaissance technologique : Basse
  - Niveau de connaissance métier : Moyenne
- *Acteur 3 :*
  - Rôle : Cadres
  - Description : Regarder les projets des subordonnés, regarder le taux de performances des subordonnés, Assigner des tâches et/ou projet à des subordonnés, regarder les activités accomplies par les subordonnés (journalier, hebdomadaire et mensuel). Les cadres sont les personnes en têtes des départements ou ceux avec des subordonnés dans la hiérarchie de l'entreprise.
  - Fréquence d'utilisation : Quotidienne
  - Niveau de connaissance technologique : Moyenne
  - Niveau de connaissance métier : élevée
- *Acteur 4 :*
  - Rôle : Utilisateur/collaborateur.
  - Description : Créer des projets pour soi, créer et modifier des activités projet et hors projets, lister les activités à faire dans la semaine. Tous personnel de Ravinala ayant un identifiant 'Active Directory' et activité.
  - Fréquence d'utilisation : Quotidienne
  - Niveau de connaissance technologique : Moyenne
  - Niveau de connaissance métier : Moyenne

### **5.2.3. Regroupement de fonctionnalités**

L'étape de regroupement de fonctionnalité extrait les thèmes principaux à partir de la vision du produit ainsi que de la liste des utilisateurs.

Les thèmes tirés à partir de la vision et des personas sont donc :

- Gérer les budgets des projets
- Gérer les ressources par projets

- Liste des activités
- Login par Active directory
- Gérer les activités
- Obtenir la hiérarchie
- Statistique de productivité

#### **5.2.4. Recueil des exigences**

Le Product Owner collecte les besoins et attentes des utilisateurs, clients et parties prenantes. Ces informations peuvent être recueillies à l'aide des entretiens, des ateliers (brainstorming, réunion...) et autres.

Les besoins sont ensuite formulés sous forme d'User Stories, qui permettent de décrire les fonctionnalités souhaitées en mettant en avant la perspective de l'utilisateur.

#### **5.2.5. Priorisation des exigences complexité**

La priorisation des exigences se fait en fonction de plusieurs critères, tels que la valeur métier, l'impact sur l'utilisateur, et les risques techniques. Elles sont alors classées par le Product Owner en fonction de ses critères.

La détermination de la complexité se fait se fait en utilisant des techniques de points, en utilisant une suite de Fibonacci comme valeur (1, 2, 3, 5, 8, 13, ...) avec 1 la complexité la plus basse [4].

Les éléments sont décomposés en tâches plus petites si nécessaire pour qu'ils soient réalisables en un sprint. L'équipe de développement estime alors la complexité de chaque tâche pour faciliter la priorisation.

Puisque le Product Backlog est continuellement mis à jour, il doit s'adapter aux nouvelles informations, aux changements de priorité et aux retours des utilisateurs.

Ce processus d'affinage inclut la réévaluation des priorités, l'ajout de nouvelles User stories et l'amélioration de la définition des tâches existantes.

### **5.3. Product Backlog**

Le Product Backlog contient l'ensemble des fonctionnalités et modifications prévues pour le produit.

Le tableau 13 représente le Product Backlog de ce projet.

*Tableau 13. Product Backlog*

User story	Priorité	Estimation (points)
En tant qu'utilisateur je veux pouvoir me connecter avec mes identifiants AD (Active directory).	Moyen	8
En tant qu'admin je veux pouvoir gérer les accès des utilisateurs.	Haute	5
En tant que collaborateur je veux pouvoir créer un projet et gérer les ressources et budget de celui-ci	Haute	5
En tant que collaborateur dans un projet je veux avoir une vision précise de l'avancée de mon projet.	Moyen	2
En tant que collaborateur dans un projet je veux voir le retard ou l'avance que j'ai sur un projet.	Faible	5
En tant qu'initiateur d'un projet je veux voir combien de temps durera ce projet.	Moyen	1
En tant que collaborateur dans un projet je veux voir clairement les tâches qui me sont associées.	Haute	8
En tant que cadre, je veux voir les ressources utilisées par mes équipes.	Moyen	8
En tant que cadre, je veux voir la disponibilité de mes équipes.	Haute	13
En tant que collaborateurs dans un projet je veux voir quand il me sera favorable de prendre un congé.	Moyen	13
En tant que cadre je veux voir quand tel ou tel membre de mes équipes pourrait prendre un nouveau projet.	Moyen	8
En tant que cadre je veux voir les productivités de chaque membre de mon équipe.	Haute	13
En tant que collaborateurs je veux voir les priorités des projets auxquelles je participe.	Haute	5

En tant que collaborateur je veux voir mes activités et ceux de mes subordonnées.	Moyen	8
En tant qu'utilisateur je veux voir l'historique de mes projets et de mes activités.	Moyen	8
En tant que cadre je veux voir les activités de la semaine passée et en cours de mon équipe.	Haute	13

#### 5.4. Sprint Backlog

Le sprint Backlog représente le travail à accomplir durant un sprint spécifique. Un sprint étant définie pendant ce projet à une période de 2 semaines.

##### 5.4.1. Présentation du travail à faire

Le travail à faire est estimé dans le temps en heures ou en jours ou en semaine, chaque user story comportent quant à eux des tâches qui leurs sont spécifiques. Les User stories se sont ajoutés au fil du temps pendant le développement et peuvent être listé comme dans le tableau 14.

Tableau 14. Liste de travail à faire

User story	Tâches	Estimation
En tant qu'utilisateur je veux pouvoir me connecter avec mes identifiants AD (Active directory).	<ul style="list-style-type: none"> <li>- Création de l'interface login</li> <li>- Relier l'interface avec les identifiants AD</li> </ul>	1semaine
En tant qu'admin je veux avoir une liste des utilisateurs	<ul style="list-style-type: none"> <li>- Get de tous les utilisateurs actifs de l'AD</li> <li>- Créer un bouton pour synchroniser les données utilisateurs avec les données AD</li> </ul>	3 jours
En tant qu'admin je veux pouvoir gérer les accès des utilisateurs.	<ul style="list-style-type: none"> <li>- Création de l'interface tableau des accès.</li> <li>- Création du formulaire d'ajout et modification d'accès.</li> </ul>	2semaine

	- Assignation des accès créer aux utilisateurs.	
En tant que collaborateur je veux pouvoir créer un projet et gérer les ressources et budget de celui-ci.	<ul style="list-style-type: none"> <li>- Création, modification, archivage des projets</li> <li>- Interface tableau projet</li> <li>- Interface d'ajout et modification de projet en incluant les ressources et les budgets</li> </ul>	2 semaines
En tant que collaborateur dans un projet je veux avoir une vision précise de l'avancée de mon projet.	- Modification du pourcentage d'avancement d'un projet	2 jours
En tant que collaborateur dans un projet je veux voir le retard ou l'avance que j'ai sur un projet.	- Affichage des détails d'un projet	3 jours
En tant qu'initiateur d'un projet je veux voir combien de temps durera ce projet.	- Ajout des délimitations dans le temps d'un projet	1 jours
En tant que collaborateur dans un projet je veux voir clairement les taches qui me sont associées.	<ul style="list-style-type: none"> <li>- Création et modification des tâches d'un projet</li> <li>- Assignation des tâches aux personnes concernées aux projets</li> </ul>	2 semaines
En tant que Directeur des Systèmes Informatiques, je veux voir les ressources utilisées par mes équipes.	- Relier les accès aux possibilités des utilisateurs.	5 jours
En tant que collaborateurs dans un projet je veux voir quand il me sera favorable de prendre un congé.	- Création d'un tableau calendrier contenant les tâches et disponibilités	2 semaines
En tant que Directeur des Systèmes informatiques je veux voir quand	- Faire que les supérieurs peuvent effectuer une recherche sur les	2 semaines

tel ou tel membre de mes équipes pourrait prendre un nouveau projet.	projets, tâches et activités de ses subordonnées	
En tant que directeur des Systèmes informatiques je veux voir les productivités de chaque membre de mon équipe.	- Faire que les supérieurs peuvent voir un graphique des activités, tâches et projet accomplies durant une période	1 semaine
En tant que collaborateurs je veux voir les priorités des projets auxquelles je participe.	- Ajout d'un champ priorité aux projets et aux tâches de projet - Ajout d'un champ priorité aux champ recherche sur le tableau projet	3 jours
En tant que collaborateur je veux voir mes activités et ceux de mes subordonnées.	- Faire que les supérieurs peuvent voir la liste des activités de ses subordonnées ainsi que la vue calendaire	5 jours
En tant qu'utilisateur je veux voir l'historique de mes projets et de mes activités.	- Création d'une historique par projet - Ajout des détails des modifications projet et tâches dans la table historique	1 semaine
En tant que cadre je veux voir les activités de la semaine passée et en cours de mon équipe	- Faire en sorte que les supérieurs puissent voir la vue calendaire des activités projet et hors projet de ses subordonnées.	1 semaine

#### 5.4.2. Planification du sprint

Dans la planification du sprint, l'équipe définit ce qu'il va accomplir pendant le sprint. Chaque planification se fait après la réalisation du sprint précédent. La durée d'un sprint étant attribué à 2 semaine. La table 15 montre la planification du sprint pour le projet.

Tableau 15. Planification du sprint

Actions	Sprint	Taches	Durée
Architecture cible et Wireframe	Sprint 0, sprint 1	- Etude de l'architecture de l'application - Conception	1 mois

		- Réalisation des wireframes	
Développement de la partie admin	Sprint 2	<ul style="list-style-type: none"> <li>- Liste des utilisateurs actifs dans AD</li> <li>- Création et modification d'accès</li> <li>- Assignment d'accès et habilitations aux utilisateurs</li> <li>- Synchronisation des données utilisateurs avec les données AD</li> </ul>	2 semaines
Développement de la partie création de projet	Sprint 3, sprint 4	<ul style="list-style-type: none"> <li>- Liste des projets de la personne connecté et de ses subordonnés.</li> <li>- Création, modification et suppression de projet.</li> <li>- Affichage des détails de projet ainsi que l'historique de projet</li> <li>- Assignment de ressources, budget et utilisateurs à un projet</li> </ul>	4 semaines
Développement de la partie création de tâches/ assignation période / priorité et attribution des équipes	Sprint 4,	<ul style="list-style-type: none"> <li>- Liste des tâches d'un projet</li> <li>- Création des tâches d'un projet par phase</li> <li>- Avancement des phases et ajout des livrables par phases</li> <li>- Ajout des status des phases et alertes de retard par phases et par tâches</li> </ul>	2 semaines
Développement de la partie activité projet et hors projet	Sprint 6	<ul style="list-style-type: none"> <li>- Liste des activités projet et hors projet de la personne connecté</li> <li>- Recherche des activités des subordonnés de la personne connecté.</li> <li>- Vue calendrier de ses activités</li> </ul>	2 semaines
Développement de la partie Dashboard : visualisation de la disponibilité des	Sprint 7	- Création des graphiques à propos des disponibilités et réalisation de la personne connecté et de ses subordonnés.	2 semaines



équipes, avancement et réalisation de projet			
Déploiement	Sprint 8	<ul style="list-style-type: none"> <li>- Déploiement de la partie front sous IIS</li> <li>- Déploiement de la partie Back sous IIS</li> </ul>	

### 5.5. Dictionnaire des données

Le dictionnaire de données permet de recenser toutes les informations nécessaires à la création de la base de données relationnelle. Il précise le libellé du donné, le nom du champ ainsi que son type.

Le tableau 16 représente le dictionnaire de donnée

*Tableau 16. Dictionnaire de données*

Attributs	Description	Taille	Type	Commentaires
actualize_user_data_habilitation	Habileté à synchronisé la base avec active directory	1	B	
assign_access_habilitation	Assigné une habilitation à un utilisateur	1	B	
audit_id	Identifiant de l'historique	50	A	
beneficiary	Départements propriétaire du projet	150	AN	
budget_id	Identifiant du budget	50	AN	
code	Code du budget	100	AN	
column	Nom de la rangée modifié	50	AN	
completion_percentage	Pourcentage de finition du projet	3	N	Soit 0 / 25 / 50 /75 /100

create_habilitation	Créer une nouvelle habilitation	1	B	
created_at	Date de création du projet	10	D	
criticality	Criticité du projet	50	AN	
currency	Devise du budget	5	A	EUR / AR
daily_effort	Heure consacrée à chaque activité	1	N	Entier allant de 1 à 8
date	Date de l'activité	10	D	YYYY/MM/DD
description	Description de l'activité		AN	
delete_habilitation	Supprimer une habilitation	1	B	
deliverable	Liens du livrable pour chaque phase		AN	
department	Département auquel l'utilisateur est lié	10	AN	
dependant_of	Id de la phase auquel il est dépendant	50	AN	
description	Description du projet		AN	
description	Description de la tâche		AN	
direction	Directions source du budget	100	A	
due_date	Date prévisionnelle de fin de la tâche	10	D	
email	Mail de l'utilisateur	100	AN	
end_date	Date de fin de la phase	10	D	

end_date_change_reason	Raison du changement de la date de fin de projet		AN	
expected_deliverable	Nom du livrable attendu	255	AN	
habilitation_id	Identifiant des habilitation	50	AN	
habilitation_admin_id	Identifiant de l'habilitation admin	50	AN	
habilitation_activity_id	Identifiant de l'habilitation activity	50	AN	
initiator	Nom de la personne qui a créé le projet	100	AN	
is_end_date_immuable	Check si la date de fin peut être modifié ou non	1	B	
label	Nom de l'habilitation	50	AN	
modification_type	Type possible de la modification	12	AN	Création/ modification/ suppression
modified_at	Date et heure de la création de l'audit	16	DT	YYYY/MM/DD T hh:mm
modified_at	Date de modification du projet	10	D	
modified_by	Nom de la personne qui a fait la modification	100	AN	
modify_hierarchy	Modifier la hiérarchie des utilisateurs dans la base	1	B	
name	Nom de l'utilisateur	100	AN	

new_value	Nouvelle valeur après modification		AN	
old_value	Ancienne valeur avant modification		AN	
original_end_date	Stock la date de fin original si elle a été modifiée.	10	D	
phase	Nom de la phase	150	AN	
phase_id	Identifiant de la phase	50	AN	
poste	Nom du poste de l'utilisateur	100	AN	
priority	Priorité du projet	21	AN	
priority	Priorité de la tâche	20	AN	
project_id	Identifiant d'un projet	50	AN	
rank	Rang de phase	2	N	
reason	Raison de la modification d'une valeur		AN	
ressource	Nom de la ressource	150	AN	
ressource_id	Identifiant de la ressource	50	AN	
restore_hierarchy	Resynchroniser les valeurs de la hiérarchie avec l'AD	1	B	
role	Rôle d'un utilisateur au sein d'un projet	20	AN	
source	Source ou fournisseur de la ressource	100	AN	

start_date	Date de début de la tâche	10	D	
start_date	Date de début de la phase	10	D	
status	Status de l'activité	20	AN	
status	Status de la phase	20	A	A faire / en cours / Terminé
status	Status de la tâche	20	AN	Backlog / en cours / traité / standby / abandonné
status	Status de l'utilisateur	7	A	Actif / Inactif
sub_type	Sous type de l'activité	50	AN	
superior_id	Identifiant du supérieur direct de l'utilisateur	50	AN	
superior_name	Nom du supérieur direct	100	AN	
table	Le nom de la table modifié	50	AN	
tache_id	Identifiant de la tâche	50	AN	
task_after	Id de la tâche suivante lié à la tâche	50	AN	
task_before	Id de la tâche précédente lié à la tâche	50	AN	
title	Titre du projet		AN	
title	Titre de la tâche	100	AN	
title	Titre de l'activité	100	AN	

type	Type de l'activité	12	AN	Projet/ transverse/ intercontrat
type	Type de la ressource	50	A	Disponible / A acquérir
update_habilitation	Modifier les habilitation	1	B	
user_id	Identifiant de l'utilisateur dans la base de données	50	AN	

AN : Alpha numérique

B : Booléen

N : numérique

D : date

A : Alphanumérique

## 5.6. Règles de gestion

Une règle de gestion est un principe suivi par l'application. Il peut s'agir d'une disposition légale, ou d'une exigence formulée par un utilisateur. D'après l'analyse faite concernant la situation actuelle, les règles de gestions suivantes sont indispensables :

**RG1** : Un Administrateur peut gérer les Habilitations des Utilisateurs ;

**RG2** : Un Utilisateur peut avoir une ou plusieurs Habilitations ;

**RG3** : Un Utilisateur peut avoir un ou plusieurs Projets ;

**RG4** : Un Projet peut être géré par un ou plusieurs Utilisateurs ;

**RG5** : Un Utilisateur peut gérer un ou plusieurs Activités ;

**RG6** : Un Utilisateur peut voir la liste et gérer les Projets de ses subordonnées ;

**RG7** : Un Projet peut avoir un Budget ou non ;

**RG8** : Un Projet peut avoir plusieurs Ressources ou non ;

**RG9** : Un Projet doit avoir au moins une Phase ;

**RG10** : Une Phase peut avoir un ou plusieurs Tâches

**RG11** : Une Tâche doit être assignée à au moins une personne

**RG12** : Une Activité est assignée à un seul Utilisateur ;

### **5.7. Modèle du domaine**

Un modèle du domaine est une visualisation des concepts d'un domaine du monde réel. L'élaboration du modèle des classes du domaine permet d'opérer une transition vers une véritable modélisation objet [5]. L'analyse du domaine est une étape totalement séparée de l'analyse des besoins. Elle peut être menée avant, en parallèle ou après cette dernière. La phase d'analyse du domaine permet d'élaborer la première version du diagramme de classe appelée modèle du domaine. Ce modèle doit définir les classes qui modélisent les entités ou concepts présents dans le domaine de l'application. Il s'agit donc de produire un modèle des objets du monde réel dans un domaine donné. Ces entités ou concepts sont identifiés directement à partir de la connaissance du domaine ou par des entretiens avec des experts du domaine. Pour établir le diagramme, on suit des étapes :

- Identifier les entités ou concepts du domaine ;
- Identifier et ajouter les associations et les attributs ;
- Organiser et simplifier le modèle en éliminant les classes redondantes et en utilisant l'héritage.

La figure 4 montre le modèle du domaine

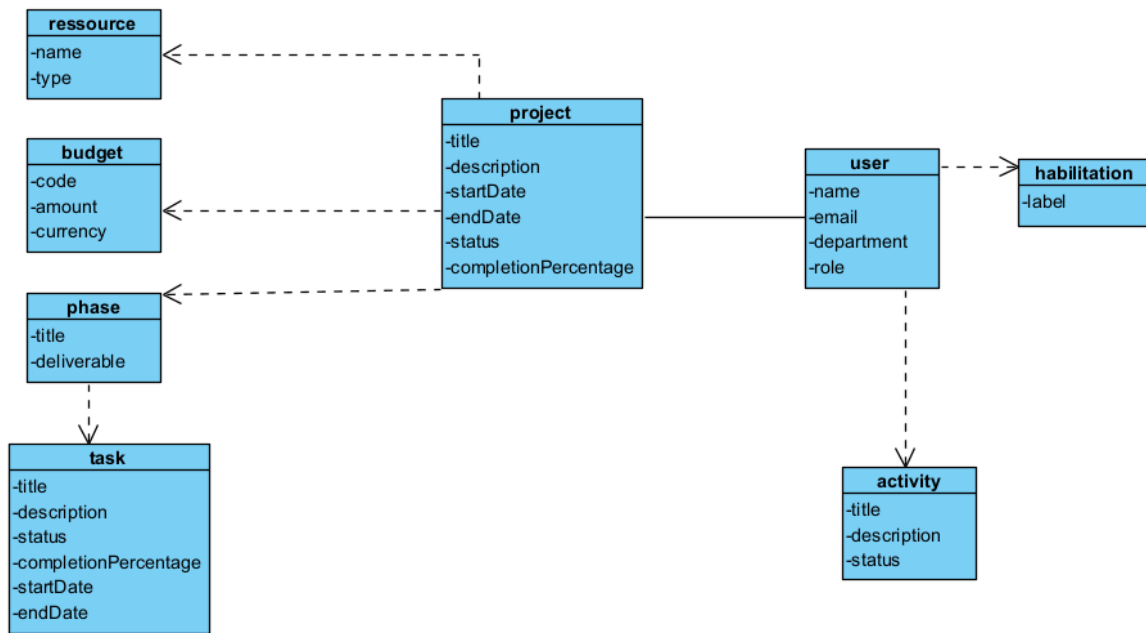


Figure 4. Modèle de domaine

## 5.8. Diagramme de classe de conception pour chaque sprint

Les sprints avec une interactions avec la base données ont leurs diagrammes de classe de conception individuelle.

### 5.8.1. Sprint 1 : User et habilitation

La figure 5 illustre le diagramme de classe de l'entités 'user' et 'habilitation'.



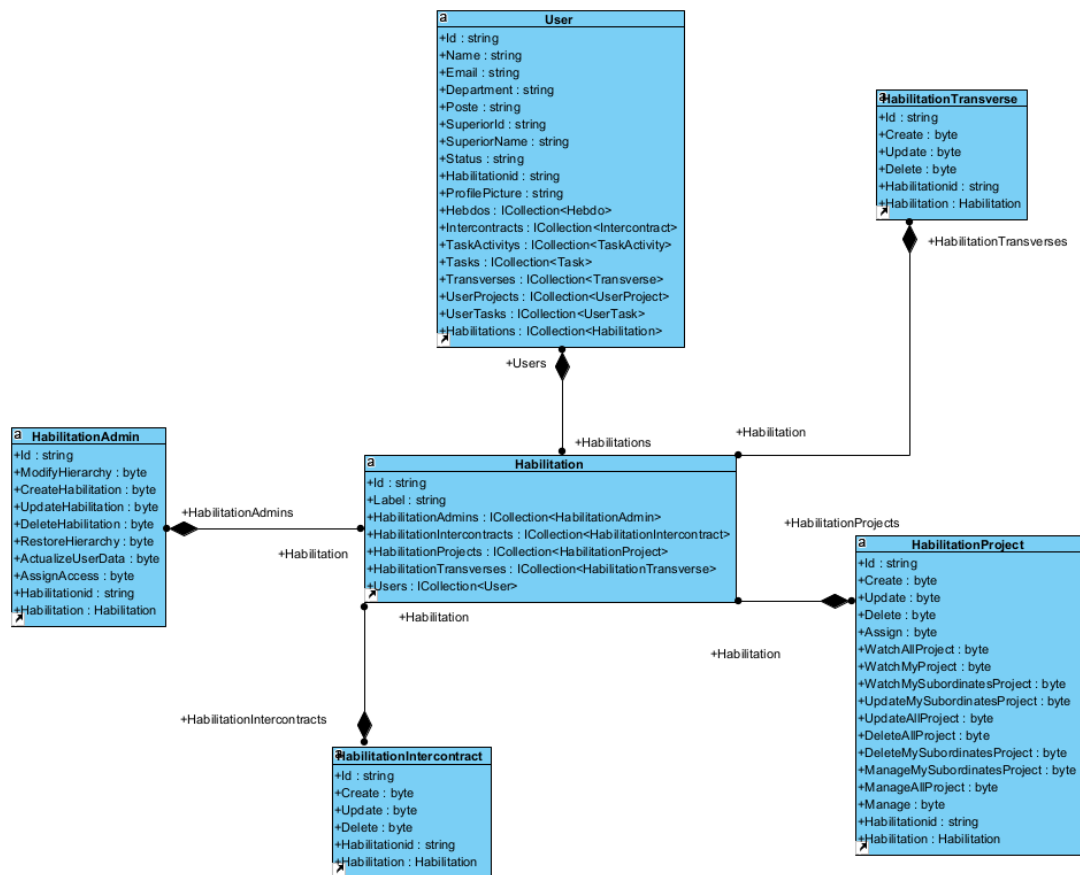


Figure 5. Diagramme de classe sprint 1

### 5.2.2. Sprint 2 : Création des projets

La figure 6 montre le diagramme de classe du sprint 2 en illustrant les relations entre les tables 'Project', 'ressource', 'Budget', 'historicProject' et 'user'

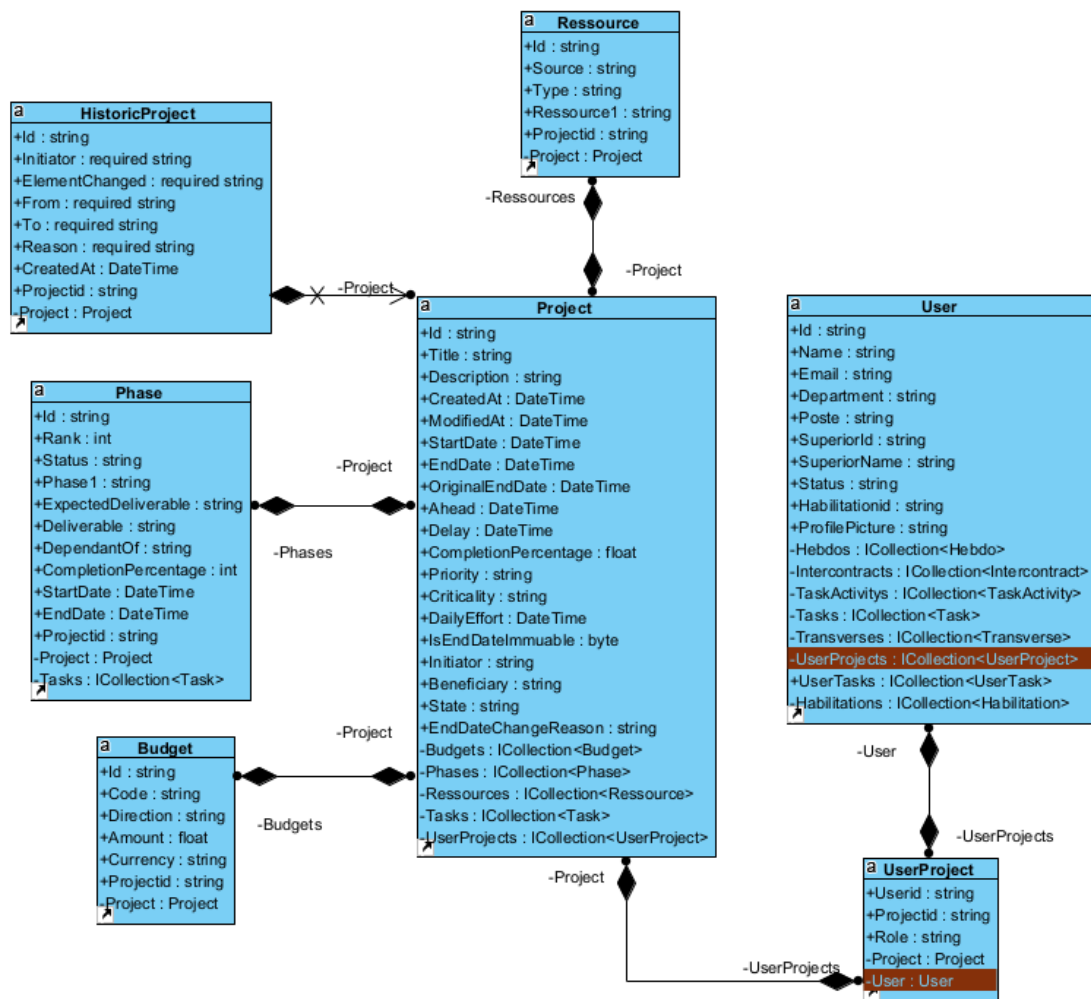


Figure 6. Diagramme de classe du sprint 2

### 5.8.3. Sprint 3 : Gestion des tâches

La figure 7 représente le diagramme de classe du sprint 3 en montrant les relations entre ‘task’, ‘project’, ‘phases’, et ‘user’

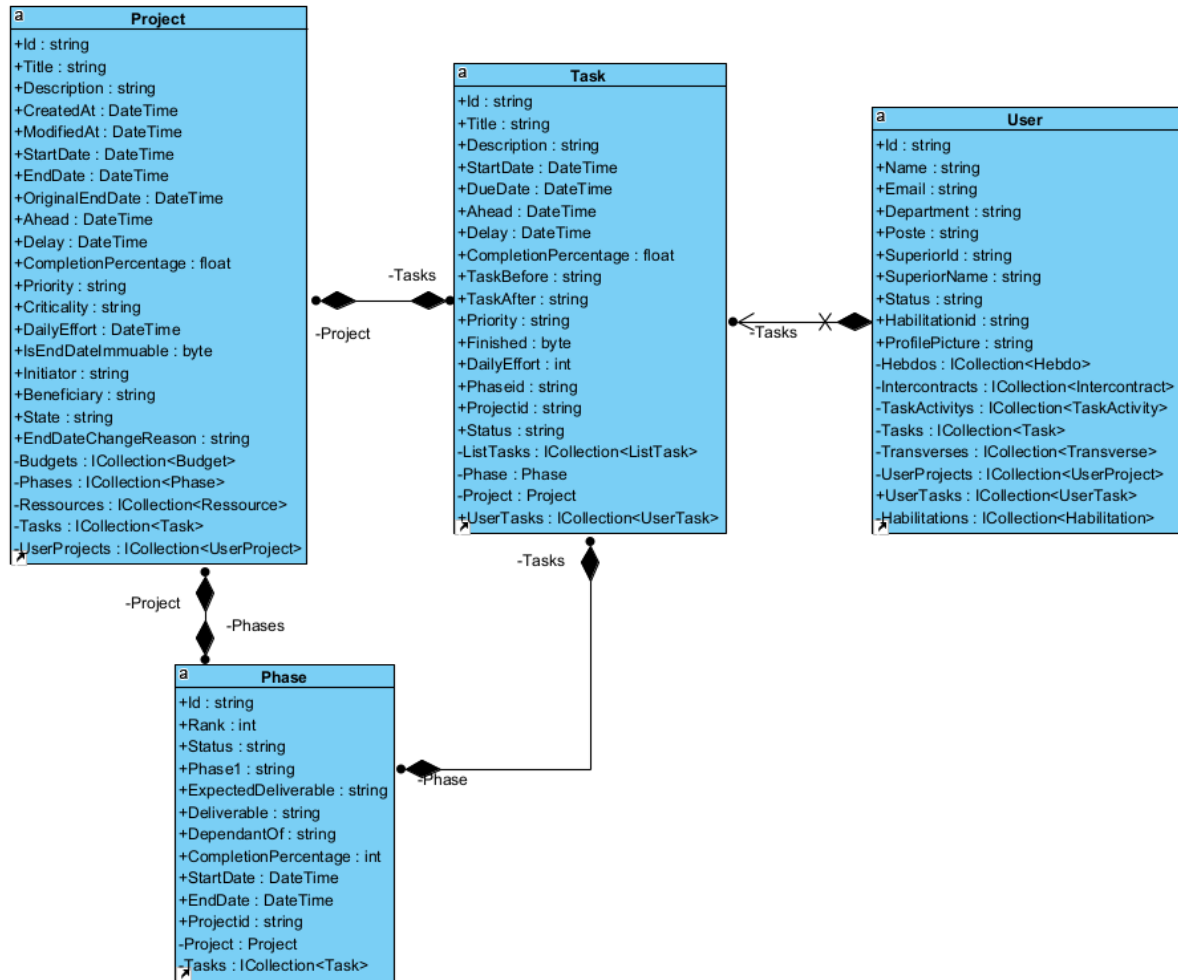


Figure 7. Diagramme de classe du sprint 3

#### 5.8.4. Sprint 4 : Activité

La figure 8 montre le diagramme de classe du sprint 4 en montrant les relations existantes entre les tables 'user', 'transverse', 'task', 'activity', 'intercontract' et 'taskactivity'

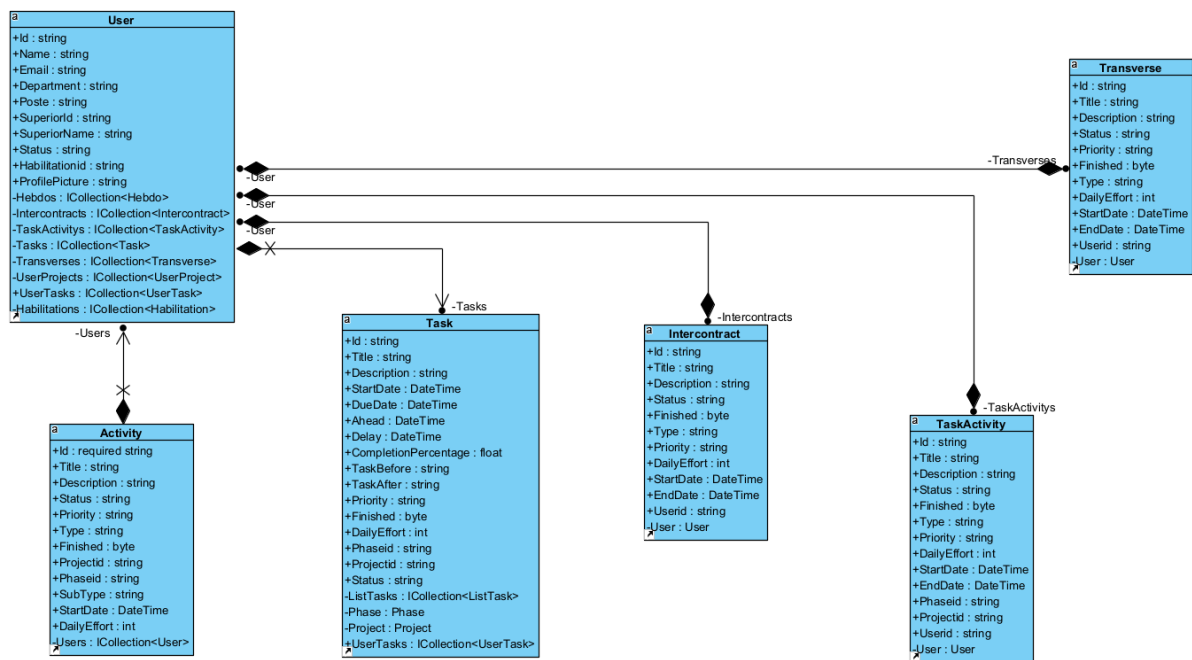


Figure 8. Diagramme de classe du sprint 4

## 5.9. Diagramme de classe de conception globale

Le diagramme de classes est un élément fondamental de la modélisation orientée objet en UML.

C'est une représentation statique qui décrit la structure d'un système en montrant :

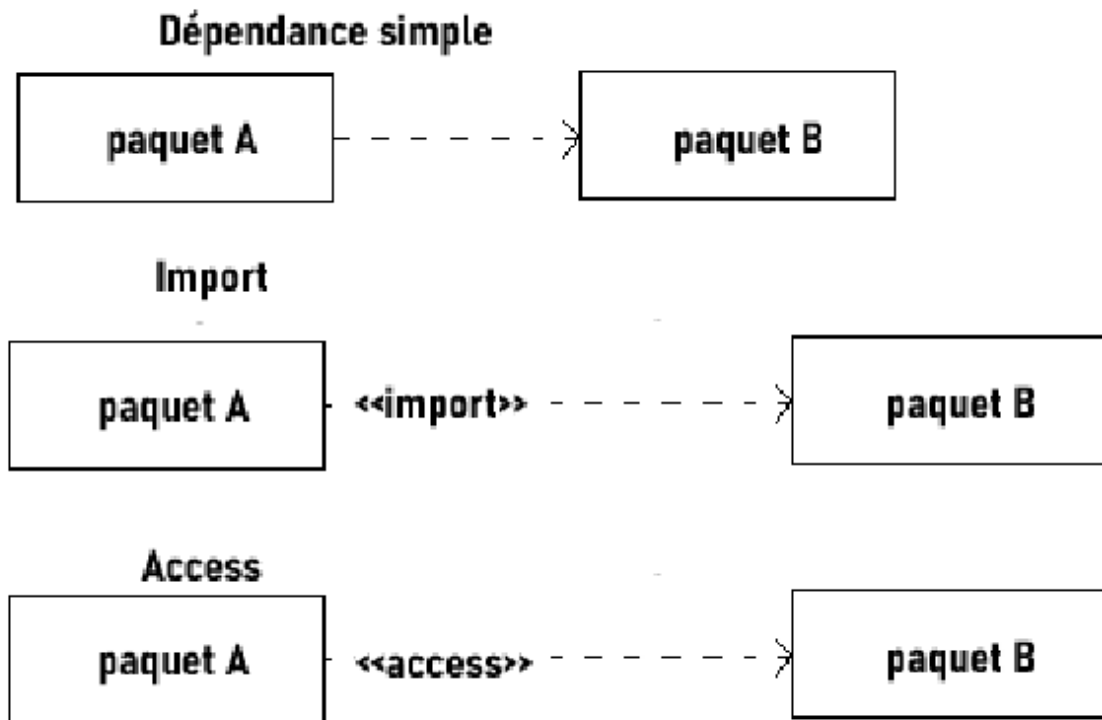
- Les structures de base :
  - Les classes du système
  - Leurs méthodes
  - Les relations entre les classes
- Types de relations principales :
  - Association : Lien simple entre classes
  - Héritage : Une classe hérite d'une autre
  - Agrégation
  - Composition
- Éléments de notation importante
  - Les attributs privé (marqué par un '-')
  - Les attribut public (marqué par un '+')
  - Les attributs protégé (marqué par un '#')
  - Les méthode (marqué par un '+')
- Multiplicité des relations

- 1 : Exactement un
- 0..1 : Zéro ou un
- 1.. \* : Un ou plusieurs
- Utilisation :
  - Phase d'analyse : comprendre le domaine
  - Phase de conception : Définir l'architecture
  - Documentation : Communiquer la structure
  - Base pour le développement
- Le diagramme de classes est essentiel car il :
  - Fournit une vue d'ensemble du système
  - Facilite la communication entre développeurs
  - Sert de base pour l'implémentation
  - Aide à maintenir la cohérence du code

La figure 9 illustre le diagramme de classe de conception global du projet.



- Eléments contenus (optionnel)
- Types de relation courantes (illustré sur la figure 10) :



*Figure 10.Type de relation courantes d'un diagramme de paquetage*

- Principaux objectifs :
  - Structurer le système en modules
  - Montrer les dépendances principales
  - Faciliter la maintenance
  - Améliorer la compréhension du système
- Avantages :
  - Vue d'ensemble claire
  - Gestion de la complexité
  - Guide pour l'organisation du code
  - Communication facilitée entre équipes

La figure 11 illustre le diagramme de paquetage de l'outil de gestion de projet.

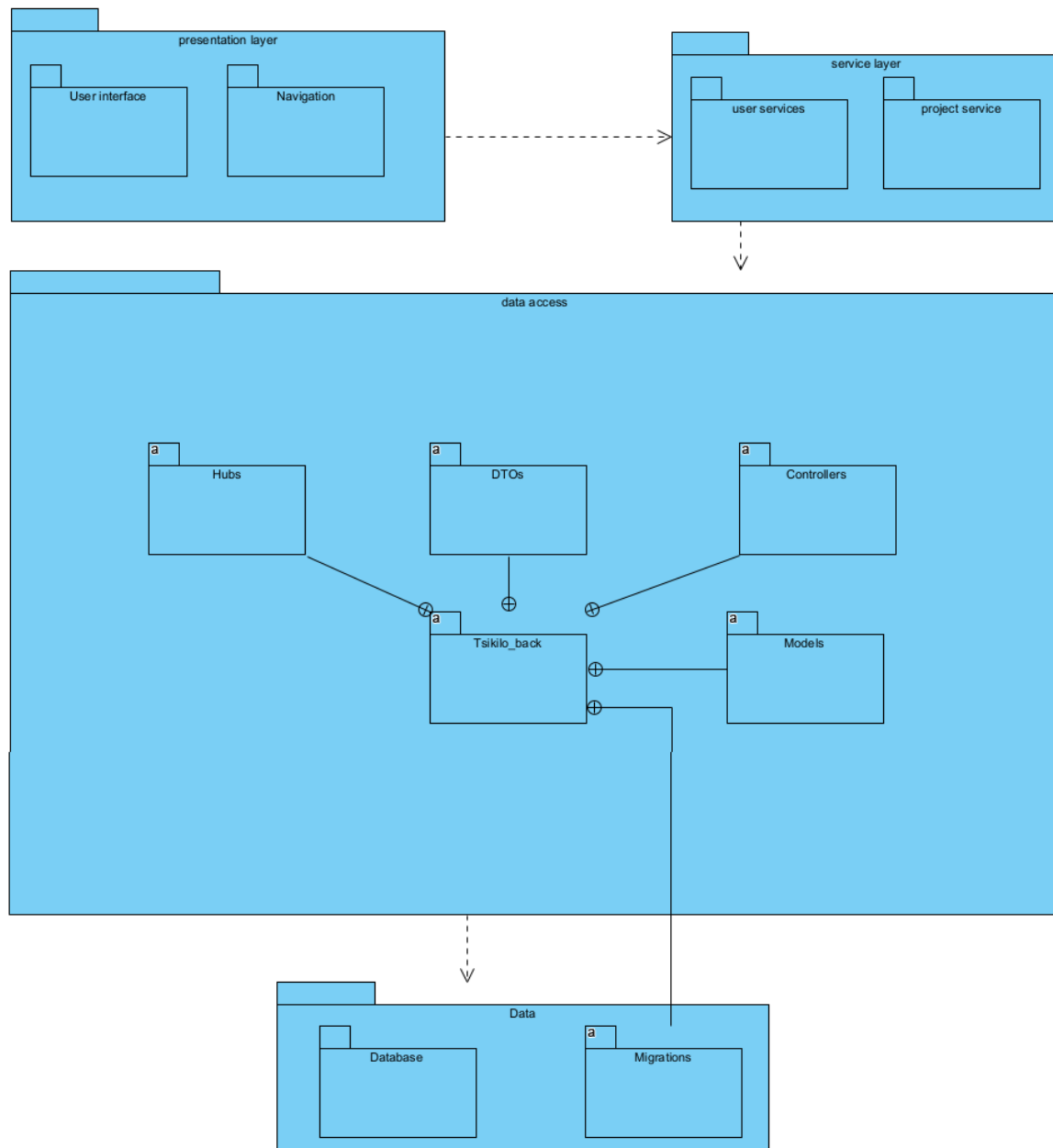


Figure 11. Diagramme de paquetage

### 5.11. Diagramme de déploiement

Le diagramme de déploiement prend en compte les différents composants matériels (tels les ordinateurs, réseaux, périphériques, systèmes de stockage, ...) et représente la manière dont ils s'interagissent et dont ils sont repartis [6]

La figure 12 illustre le diagramme de déploiement de l'outil de gestion.



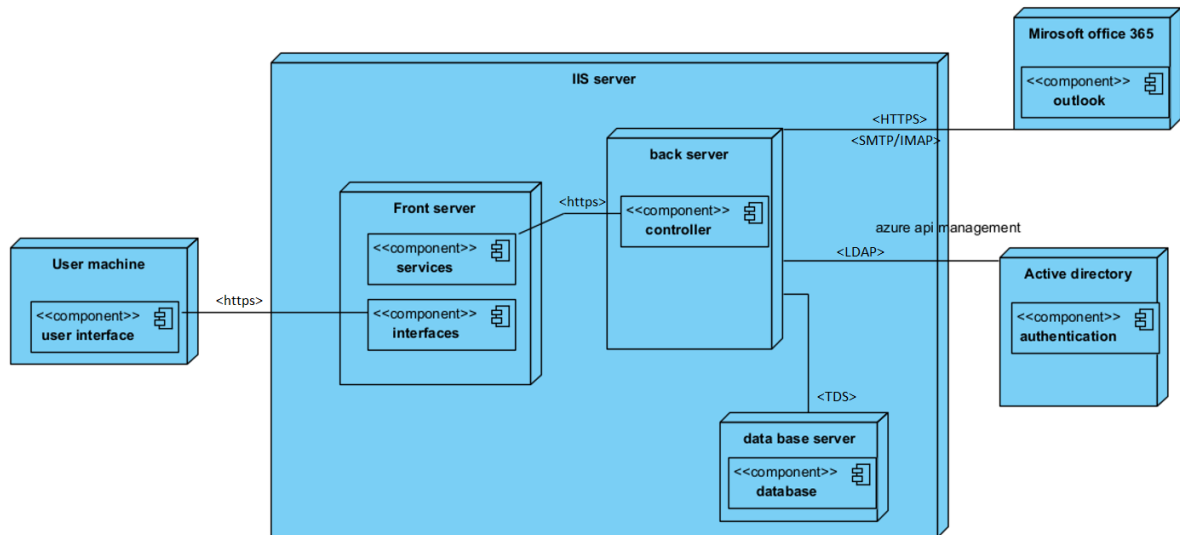


Figure 12. Diagramme de déploiement

## **PARTIE III. REALISATION**

## Chapitre 6. Mise en place de l'environnement de développement

Ce chapitre traitera de l'installation et configuration de tous les outils nécessaires à la réalisation et au développement de l'outil.

### 6.1. Installation et configuration des outils

#### 6.1.1. Installation de SQL server

SQL server va permettre la gestion de base de données, il suffit de le télécharger et de lancer l'installateur. SQL serveur possède une version Express et une version Developer, l'installation reste la même. Après avoir lancé l'installateur, une interface de choix sur le type d'installation s'impose comme sur la figure 13.

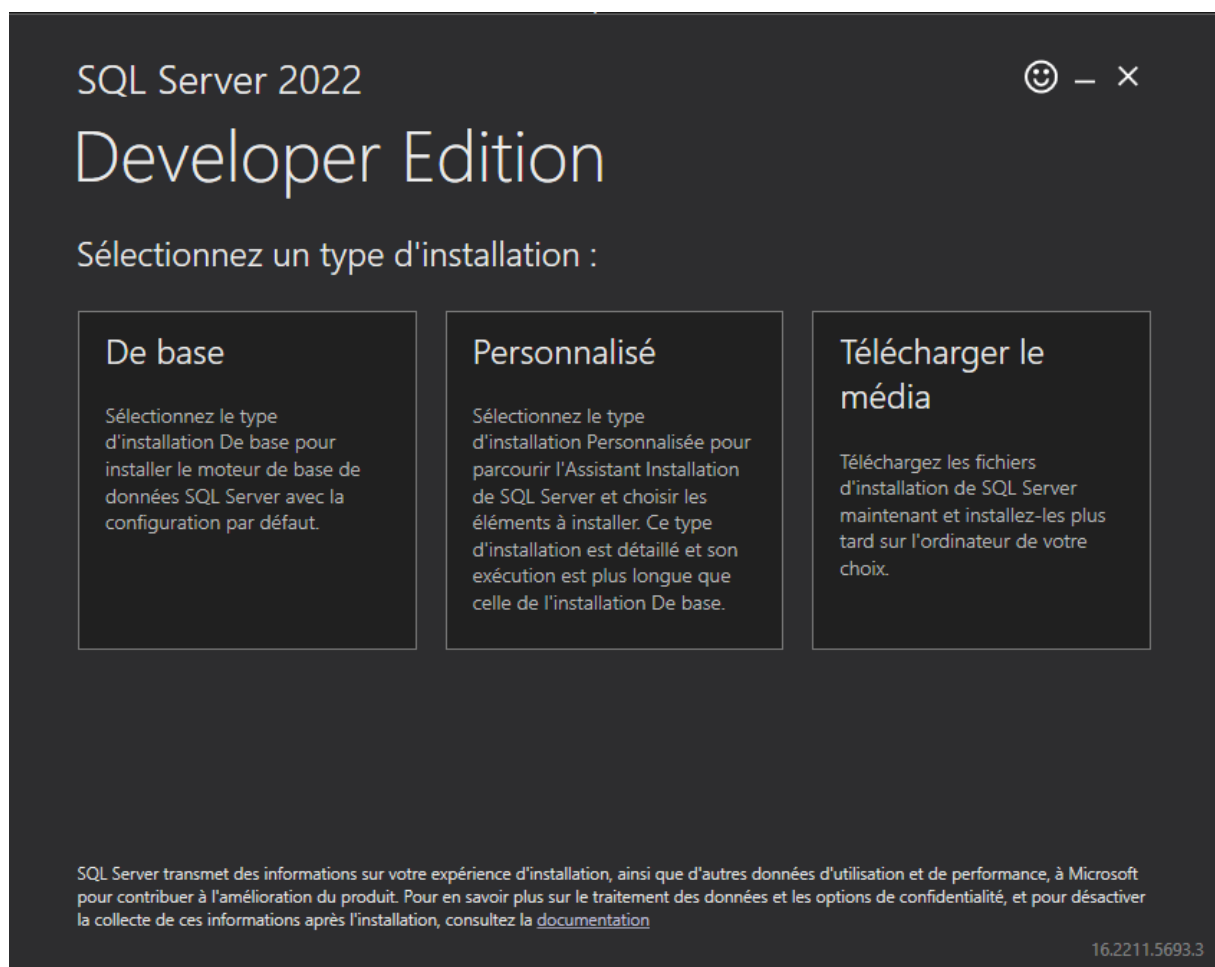


Figure 13. Interface d'installation de SQL server : Type d'installation

- Choisir un type et faire suivant jusqu'à installer.

Après que SQL server soit installé, il faut aussi installer SSMS (SQL Server Manager Studio) qui facilite la visualisation des données.

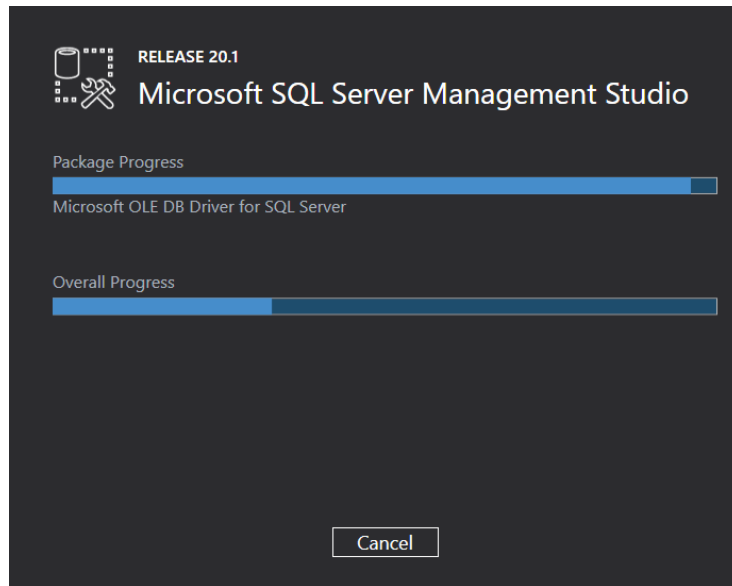


Figure 14. Interface d'Installation de SSMS

### 6.1.2. Installation de Visual Paradigm

Visual Paradigm est un outil de modélisation et est utilisée pour la réalisation de diagramme. Visual Paradigm gère la plupart des diagrammes spécifiée sous la norme UML2.0 qui est basée sur le langage de modélisation UML, ce qui fait de cet outil le meilleur choix pour la partie conception.

L'installation est une installation classique, après installation une interface comme sur la figure 15 s'affiche qui es l'interface première de Visual Paradigm :

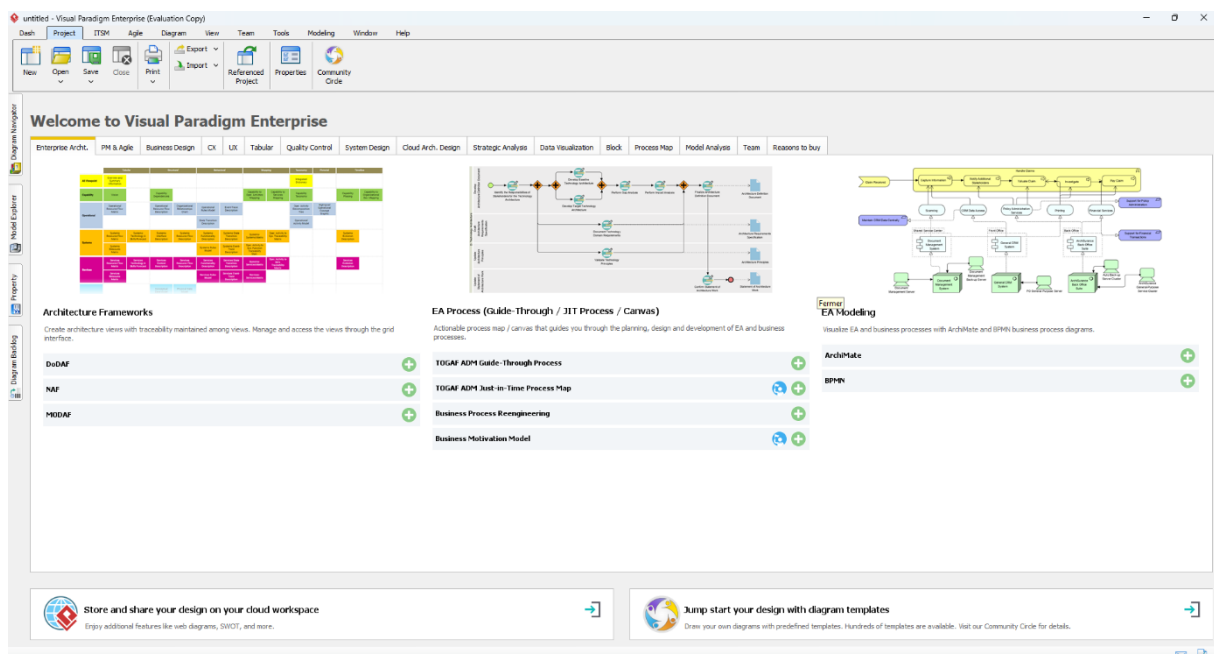


Figure 15. Interface de Visual Paradigm

### 6.1.3. Installation de SDK .NET

SDK .Net (Software Development Kit) est un ensemble d'outil, de bibliothèques et de ressources fourni par Microsoft pour permettre aux développeurs de créer, compiler, tester et exécuter des applications .NET.

Pour l'installer il suffit de le télécharger et de lancer le programme d'installation.

### 6.1.4. Installation de Vs Code

Vs Code est l'éditeur de texte utilisé lors du développement, l'installation n'as pas de spécificité mais ressemble à une installation classique. Mais pour faciliter le développement il est préférable d'installer plusieurs extensions dans Vs Code.

Voici la liste des extensions à installer :

- .NET Extension Pack
- .NET Install Tool
- C# Dev kit
- NuGet Gallery
- C# extensions

En figure 16 l'interface VS code après installation et avec les extensions installer :

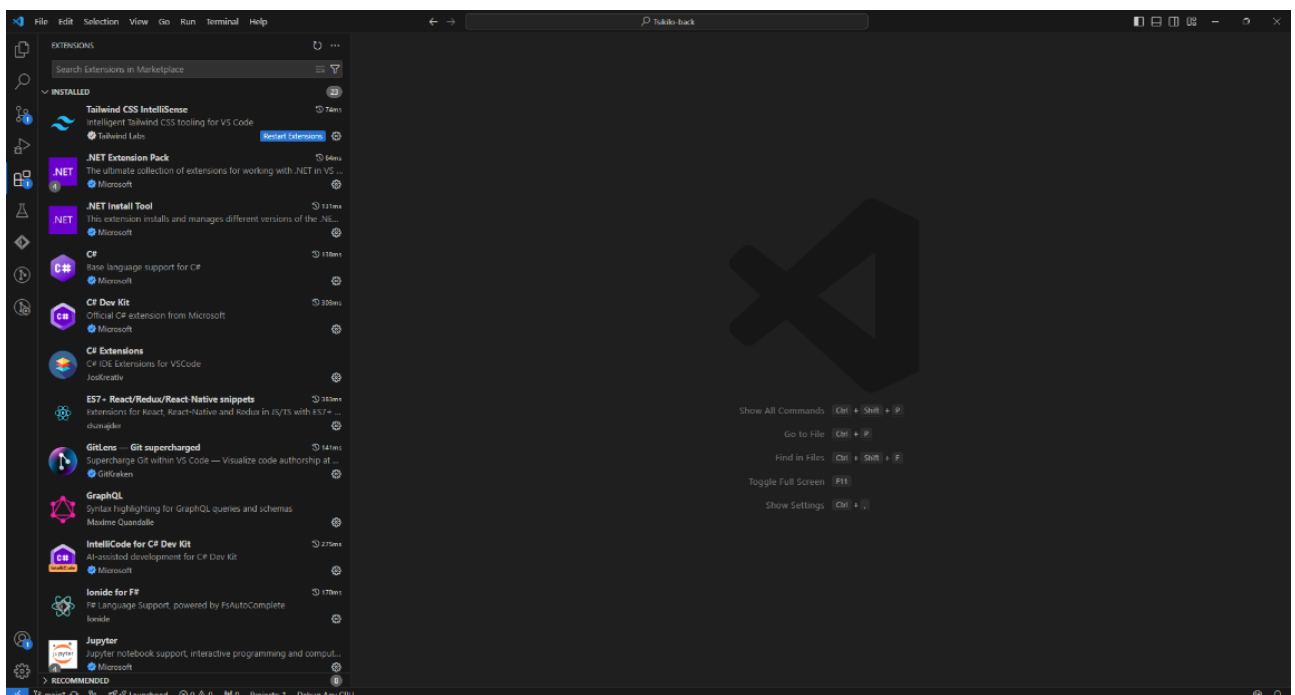


Figure 16. Interface de VS code

## 6.2. Architecture de l'application

L'architecture utilisé est une architecture à **4-Tiers** ou à **4 couches**. Chaque couche est responsable d'une partie spécifique du traitement des données et de l'interaction utilisateur. Dans notre cas on a les 4 couches suivantes :

### - Couche de présentation

- Cette couche gère l'interface utilisateur et l'interaction avec l'utilisateur final. Elle est chargée de présenter les données au client et de collecter les entrées utilisateur.
- Elle communique avec le back-end via des appels API et en envoyant et recevant des données JSON.

### - Couche de services

- Elle contient la logique métier principale de l'application. Cette couche traite les requêtes venant du front-end, exécute es règles métier, et coordonne les échanges entre le front-end et la base de données.

### - Couche d'accès aux données

- Cette couche interagit directement avec la base de données pour récupérer, insérer, mettre à jour ou supprimer des données. Elle fournit au back-end des méthodes pour manipuler les données sans exposer les détails internes de la base de données.

### - Couche de données

- Elle stocke toutes les données persistantes de l'application, comme les informations utilisateurs ou celles des projets.

## Chapitre 7. Développement de l'application

Ce chapitre aborde les outils et langages de développement utilisés, les différentes étapes de la réalisation, ainsi qu'une présentation visuelle de l'application et de son fonctionnement à travers des captures d'écran.

### 7.1. Création de la base de données

La base de données utilisée se nomme « gmp »

Pour créer cette base de données la commande est :

```
CREATE DATABASE gmp;
```

Elle contient les tables project, user, habilitation, task, phase, budget, ressource, audit, habilitation\_admin, habilitation\_project, habilitation\_transverse, habilitation\_intercontract, habilitation, user\_task, et user\_project ce que les figures 17 à 22.

```
CREATE TABLE [dbo].[project](
    [id] [varchar](50) NOT NULL,
    [title] [varchar](1000) NULL,
    [description] [nvarchar](3000) NULL,
    [start_date] [datetime] NULL,
    [end_date] [datetime] NULL,
    [ahead] [datetime] NULL,
    [delay] [datetime] NULL,
    [completion_percentage] [real] NULL,
    [priority] [varchar](21) NULL,
    [daily_effort] [datetime] NULL,
    [initiator] [nvarchar](100) NULL,
    [beneficiary] [varchar](150) NULL,
    [resource] [varchar](255) NULL,
    [is_end_date_immuable] [tinyint] NULL,
    [created_at] [datetime] NULL,
    [modified_at] [datetime] NULL,
    [original_end_date] [datetime] NULL,
    [criticality] [varchar](50) NULL,
    [state] [varchar](50) NULL,
    [end_date_change_reason] [nvarchar](3000) NULL,
    CONSTRAINT [PK__project__3213E83F6B9D5490] PRIMARY KEY CLUSTERED ([
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON,
        OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF
    ]) ON [PRIMARY]
) ON [PRIMARY]
```

Figure 17. Code de création de la table project

```
CREATE TABLE [dbo].[user](
    [id] [varchar](50) NOT NULL,
    [name] [varchar](100) NULL,
    [email] [varchar](100) NULL,
    [department] [varchar](60) NULL,
    [poste] [varchar](100) NULL,
    [superiorId] [varchar](50) NULL,
    [superior_name] [varchar](100) NULL,
    [status] [varchar](10) NULL,
    [habilitationid] [varchar](50) NULL,
    [profile_picture] [varchar](1000) NULL,
    CONSTRAINT [PK__user__3213E83FF82B595B] PRIMARY KEY CLUSTERED ([id] ASC) WITH (
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON,
        OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF
    ) ON [PRIMARY]
) ON [PRIMARY]
```

Figure 18. Code de création de la table user

```

GO
CREATE TABLE [dbo].[habilitation_intercontract](
    [id] [varchar](50) NOT NULL,
    [create] [tinyint] NULL,
    [update] [tinyint] NULL,
    [delete] [tinyint] NULL,
    [habilitationid] [varchar](50) NOT NULL,
    CONSTRAINT [PK_habilita__3213E83F1D697FAD] PRIMARY KEY CLUSTERED ([id] ASC) WITH (
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON,
        OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF
    ) ON [PRIMARY]
) ON [PRIMARY]

GO
/***** Object: Table [dbo].[habilitation_project]    Script Date: 11/11/2024 15:59:28 ***
SET
ANSI_NULLS ON
GO
SET
QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[habilitation_project](
    [id] [varchar](50) NOT NULL,
    [create] [tinyint] NULL,
    [update] [tinyint] NULL,
    [delete] [tinyint] NULL,
    [assign] [tinyint] NULL,
    [habilitationid] [varchar](50) NOT NULL,
    [delete_my_subordinates_project] [tinyint] NULL,
    [manage] [tinyint] NULL,
    [manage_my_subordinates_project] [tinyint] NULL,
    [update_my_subordinates_project] [tinyint] NULL,
    [watch_my_project] [tinyint] NULL,
    [watch_my_subordinates_project] [tinyint] NULL,
    [delete_all_project] [tinyint] NULL,
    [manage_all_project] [tinyint] NULL,
    [update_all_project] [tinyint] NULL,
    [watch_all_project] [tinyint] NULL,
    CONSTRAINT [PK_habilita__3213E83F958A7D07] PRIMARY KEY CLUSTERED ([id] ASC) WITH (
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON,
        OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF
    ) ON [PRIMARY]
) ON [PRIMARY]

GO
/***** Object: Table [dbo].[habilitation_transverse]    Script Date: 11/11/2024 15:59:28 *****/

```

Figure 19. Code de création des tables habilitation projet et habilitation intercontrat

```

CREATE TABLE [dbo].[habilitation](
    [id] [varchar](50) NOT NULL,
    [label] [nvarchar](50) NULL,
    CONSTRAINT [PK_habilita__3213E83FA5E09D93] PRIMARY KEY CLUSTERED ([id] ASC) WITH (
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON,
        OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF
    ) ON [PRIMARY]
) ON [PRIMARY]

GO
/***** Object: Table [dbo].[habilitation_admin]    Script Date: 11/11/2024 15:59:28 *****/
SET
ANSI_NULLS ON
GO
SET
QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[habilitation_admin](
    [id] [varchar](50) NOT NULL,
    [modify_hierarchy] [tinyint] NULL,
    [create_habilitation] [tinyint] NULL,
    [update_habilitation] [tinyint] NULL,
    [delete_habilitation] [tinyint] NULL,
    [restore_hierarchy] [tinyint] NULL,
    [habilitationid] [varchar](50) NOT NULL,
    [actualize_user_data_habilitation] [tinyint] NULL,
    [assign_access_habilitation] [tinyint] NULL,
    CONSTRAINT [PK_habilita__3213E83FD1E43E08] PRIMARY KEY CLUSTERED ([id] ASC) WITH (
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON,
        OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF
    ) ON [PRIMARY]
) ON [PRIMARY]

```

Figure 20. Code de création des tables habilitation et habilitation admin



```
CREATE TABLE [dbo].[audit](
    [id] [varchar](50) NOT NULL,
    [table] [varchar](50) NOT NULL,
    [project_id] [varchar](50) NOT NULL,
    [column] [varchar](50) NOT NULL,
    [old_value] [nvarchar](3000) NOT NULL,
    [new_value] [nvarchar](3000) NOT NULL,
    [modified_by] [nvarchar](150) NOT NULL,
    [modified_at] [datetime] NOT NULL,
    [reason] [nvarchar](3000) NULL,
    [modification_type] [varchar](10) NOT NULL,
    CONSTRAINT [PK_audit] PRIMARY KEY CLUSTERED ([id] ASC) WITH (
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON,
        OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF
    ) ON [PRIMARY]
)
```

Figure 22. Code de création de la table audit

```
CREATE TABLE [dbo].[task](
    [id] [varchar](50) NOT NULL,
    [title] [varchar](1000) NULL,
    [description] [varchar](1000) NULL,
    [start_date] [datetime] NULL,
    [due_date] [datetime] NULL,
    [ahead] [datetime] NULL,
    [delay] [datetime] NULL,
    [completion_percentage] [real] NULL,
    [task_before] [varchar](50) NULL,
    [task_after] [varchar](50) NULL,
    [priority] [varchar](20) NULL,
    [finished] [tinyint] NULL,
    [projectid] [varchar](50) NULL,
    [userid] [varchar](50) NULL,
    [phaseid] [varchar](50) NULL,
    [status] [varchar](20) NOT NULL,
    [daily_effort] [int] NULL,
    CONSTRAINT [PK__task__3213E83FD18286CE] PRIMARY KEY CLUSTERED ([id] ASC) WITH (
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON,
        OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF
    ) ON [PRIMARY]
)
```

Figure 21. Code de création de la table task

## 7.2. Codage de l'application

### 7.2.1. Le coté backend

Le coté backend comporte la liaison avec la base de données, la liaison avec l'Active directory et les requêtes vers la base de données. La figure 23 montre comment est la structure en coté backend

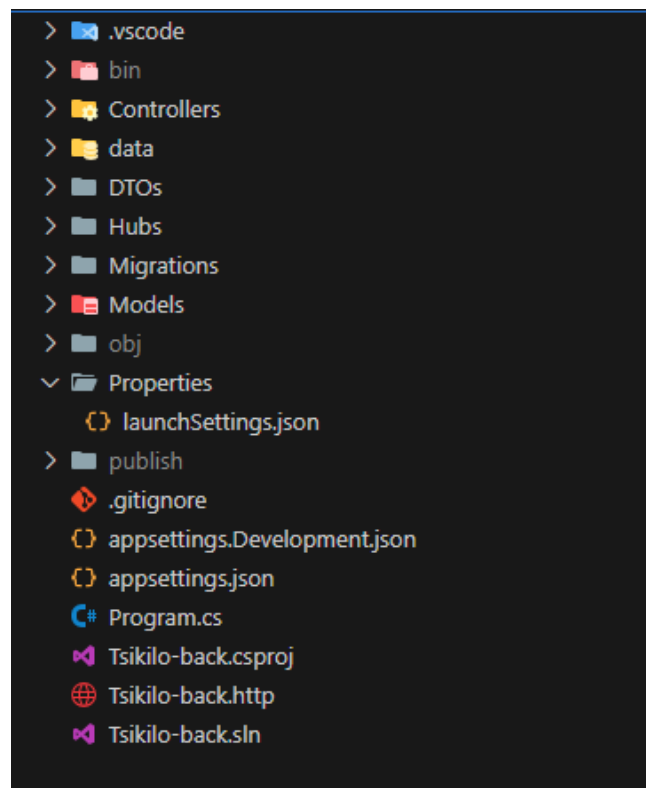


Figure 23. Structure du coté backend

- Le dossier « **Controllers** » contient les contrôleurs qui gèrent les requêtes HTTP entrantes (telles que GET, POST, PUT, et DELETE) et définissent la logique pour traiter les actions requises.
- Le dossier « **Migrations** » contient les fichiers de migration qui enregistrent les modifications de schéma de la base de données. Les migrations sont générées automatiquement lors de changements dans les modèles de données et permettent de synchroniser le schéma de la base de données avec les entités définies dans « **Models** ».
- Le dossier « **Models** » contient les classes de modèles qui définissent la structure des entités de la base de données. Chaque modèle représente une table dans la base de données et définit les propriétés des colonnes, les relations entre tables, et d'autres configurations de données.
- Le fichier « **appsettings.json** » contient les configurations globales de l'application, telles que les chaînes de connexion de base de données, les paramètres de l'application, et d'autres variables de configuration.
- Le fichier « **Program.cs** » contient le code d'initialisation et de démarrage de l'application. Il configure également les services, comme la connexion à la base de données, les dépendances nécessaires, et les configurations de middlewares. Ce fichier exécute l'application lorsqu'elle est lancée.

La figure 24 montre un extrait de code réalisant une requête GET de tous les utilisateurs de l'Active Directory, puis remplit la base de données avec les données reçues.

```
UserController.cs M X
Controllers > UserController.cs
12 {
16 {
23
24 // Populate the db from user ad
25 [HttpPost("populate")]
26 public ActionResult PopulateUsersFromAD()
27 {
28     try
29     {
30         var adUsers = BuildFullOrganisationHierarchy();
31         var users = MapADUsersToUsers(adUsers);
32         int batchSize = 10;
33         for (int i = 0; i < users.Count; i += batchSize)
34         {
35             var batch = users.Skip(i).Take(batchSize).ToList();
36
37             try
38             {
39                 _context.Users.AddRange(batch);
40                 _context.SaveChanges();
41             }
42             catch (Exception ex)
43             {
44                 return StatusCode(500, $"Internal server error on transaction: {ex.Message}");
45             }
46         }
47         return Ok("Users have been successfully populated from Active Directory");
48     }
49     catch (Exception ex)
50     {
51         return StatusCode(500, $"Internal server error: {ex.Message}");
52     }
53 }
```

Figure 24. Extrait de code, population de la table user

La figure 25 quant à elle montre une requête POST créant une tâche par user

```
[HttpPost("create")]
public async Task<IActionResult> CreateTask([FromBody] TaskDTO dto)
{
    if (!ModelState.IsValid)
    {
        var errors = ModelState
            .Where(ms => ms.Value.Errors.Count > 0)
            .Select(ms => new
            {
                Field = ms.Key,
                Errors = ms.Value.Errors.Select(e => e.ErrorMessage).ToList()
            })
            .ToList();
        Console.WriteLine(JsonConvert.SerializeObject(errors, Formatting.Indented));

        return BadRequest(new { message = "Invalid data", errors });
    }
    if (!ModelState.IsValid)
    {
        return BadRequest(ModelState);
    }
    try
    {
        var task = new Models.Task
        {
            Id = dto.Id,
            Title = dto.Title,
            Description = dto.Description,
            Phaseid = dto.Phaseid,
            Projectid = dto.Projectid,
            Priority = dto.Priority,
            StartDate = dto?.StartDate,
            DueDate = dto?.DueDate
        };
        _context.Tasks.Add(task);
        await _context.SaveChangesAsync();
        if (dto.ListUsers != null && dto.ListUsers.Count != 0)
        {
            var userTasks = dto.ListUsers.Select(up => new UserTask
            {
                Userid = up.Userid,
                Taskid = task.Id
            }).ToList();

            _context.UserTasks.AddRange(userTasks);

            await _context.SaveChangesAsync();
        }

        return Ok(task);
    }
    catch (System.Exception ex)
    {
        return StatusCode(500, $"Internal server error {ex.Message}");
    }
}
```

Figure 25. Extrait de code : création d'une tâche.

La figure 26 montre un code utilisant SignalR pour envoyer une notification via WebSocket lorsqu'une tâche est déplacée d'une colonne à une autre. Ce code met à jour l'état de la tâche dans la base de données et informe les clients connectés du changement de statut en temps réel

```
public async System.Threading.Tasks.Task TaskMoved(string taskId, string fromColumn, string toColumn)
{
    try
    {
        var task = await _context.Tasks.FirstOrDefaultAsync(t => t.Id == taskId);

        if (task == null)
        {
            Console.WriteLine($"Tâche avec l'ID {taskId} introuvable.");
            return;
        }
        task.Status = toColumn;

        await _context.SaveChangesAsync();

        await Clients.All.SendAsync("TaskStatusUpdated", taskId, fromColumn, toColumn);
        System.Console.WriteLine($"Tâche {taskId} déplacée de {fromColumn} vers {toColumn}");
    }
    catch (System.Exception ex)
    {
        Console.WriteLine($"Erreur lors du déplacement de la tâche {taskId} : {ex.Message}");
    }
}
```

Figure 26. Extrait de code : déplacement de tâches en utilisant websocket

### 7.2.2. Le côté front end

Le côté front end a été codé en reactJs, couplé avec Vite et typeScript, elle comporte les codes des interfaces, les appels api, et les composants UI. La figure 27 illustre la structure côté front end.

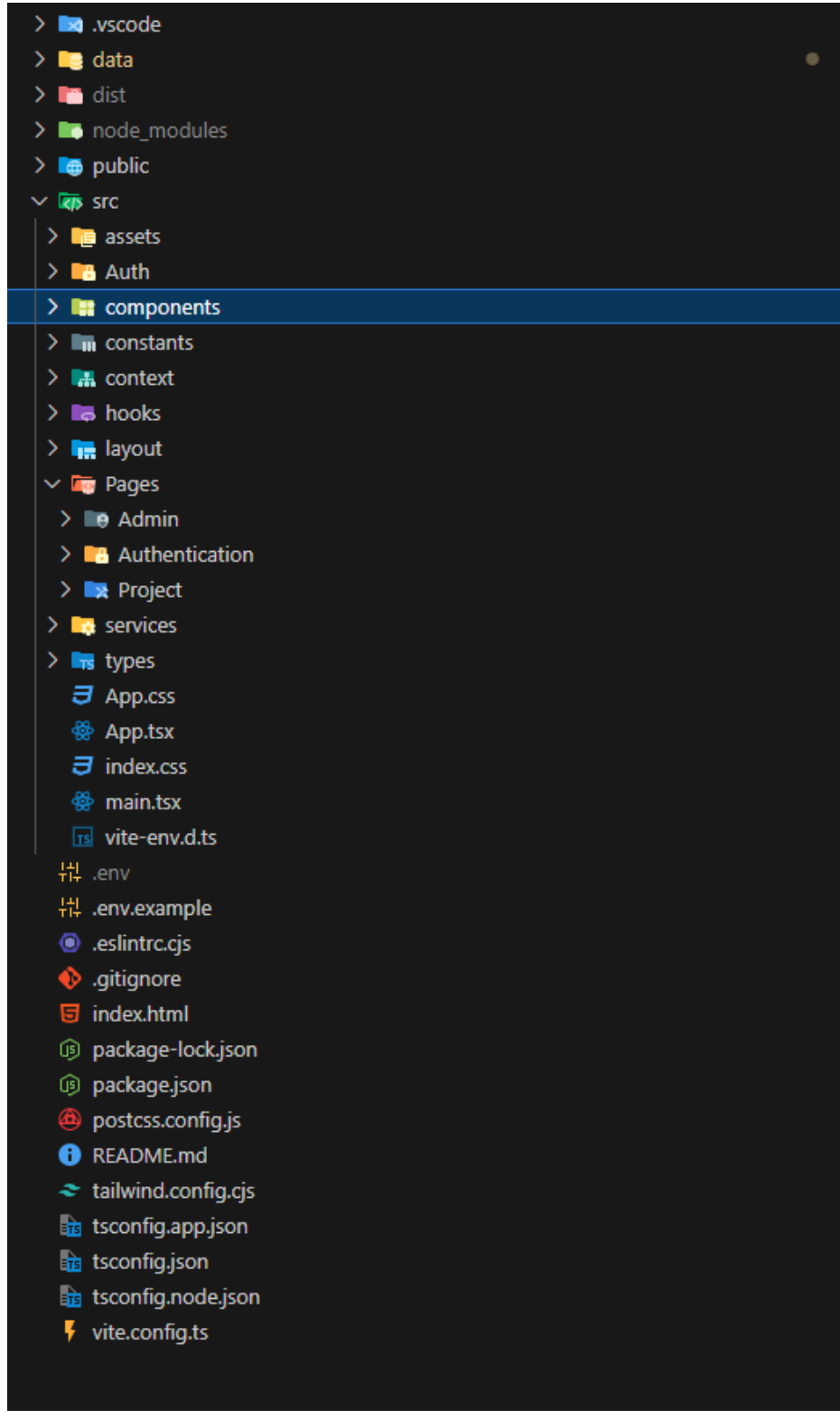
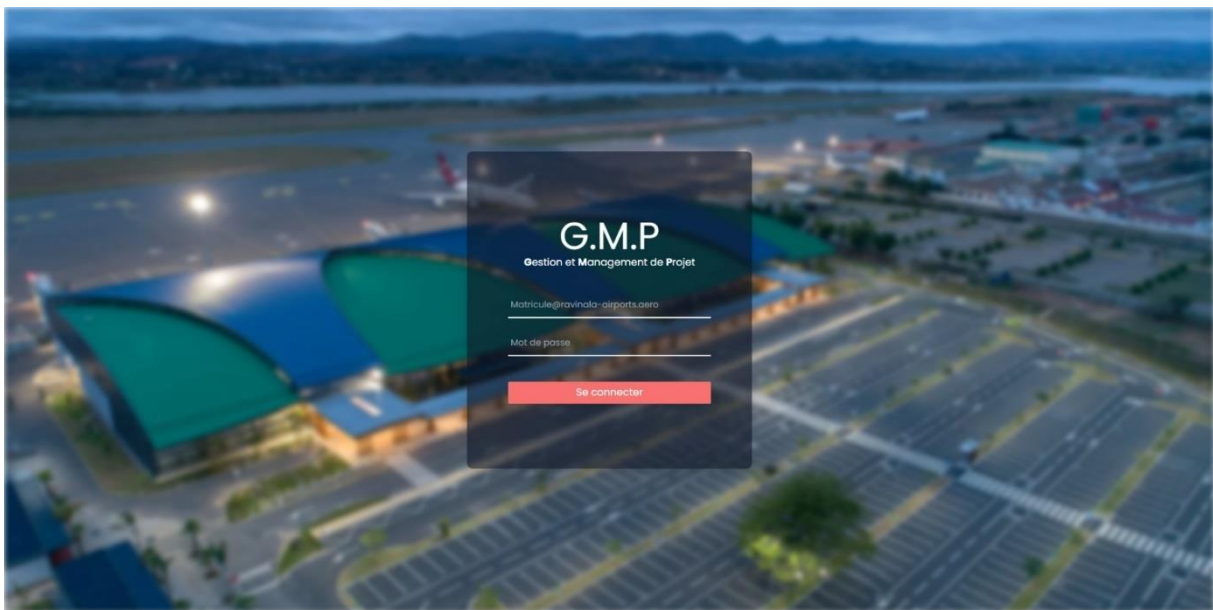


Figure 27. Structure du côté frontend

- Le dossier « **components** » contient tous les composants UI de l'interface, comme les 'buttons', les 'inputs', les 'tables' et les 'modal'
- Le dossier « **Pages** » contient les pages des interfaces ;
- Le dossier « **services** » contient les appels API
- Le dossier « **types** » contient les types des objet Type Script

### 7.3. Présentation de l'application

Pour bien présenter l'outil de gestion de projet, les figures 28 à 34 en guise de capture de l'outil en fonctionnement facilitera la présentation. Les noms des utilisateurs et des projets ont été préalablement flouté par soucis de sécurité et de confidentialité.



*Figure 28. Interface login*

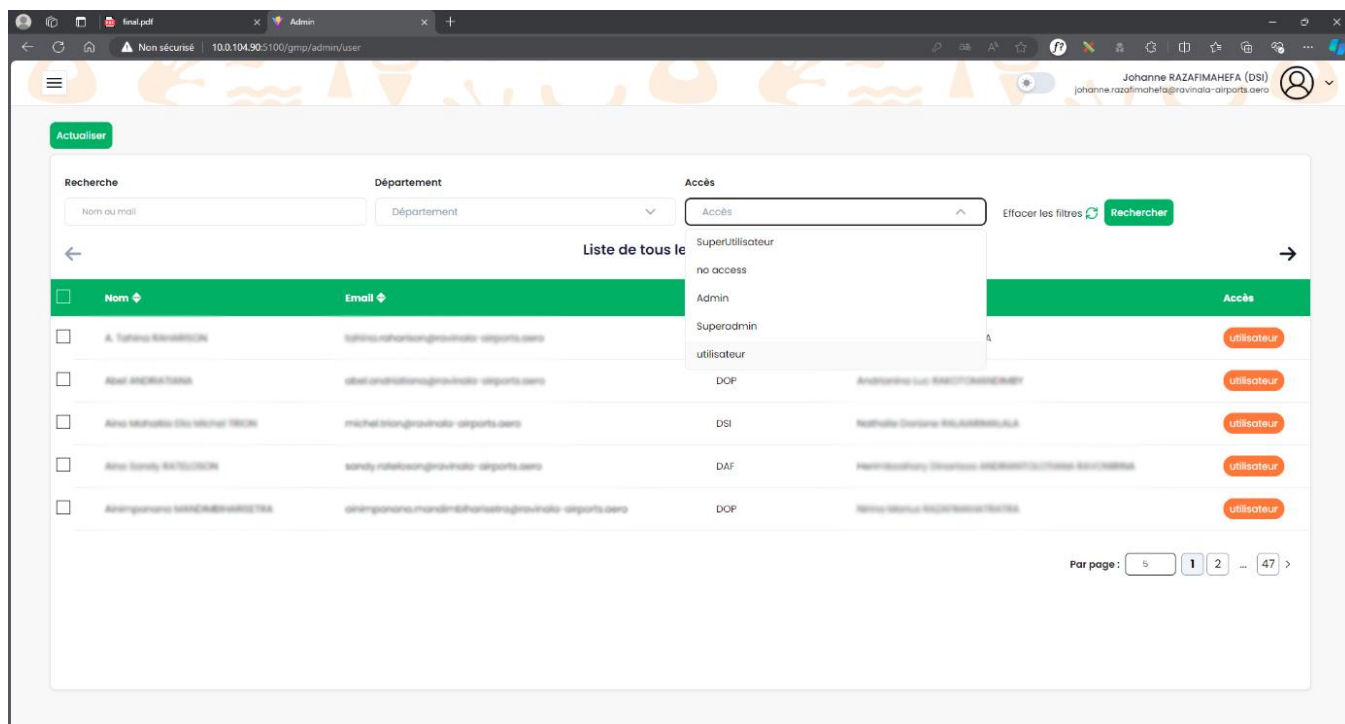


Figure 29. Interface admin, liste des utilisateurs.

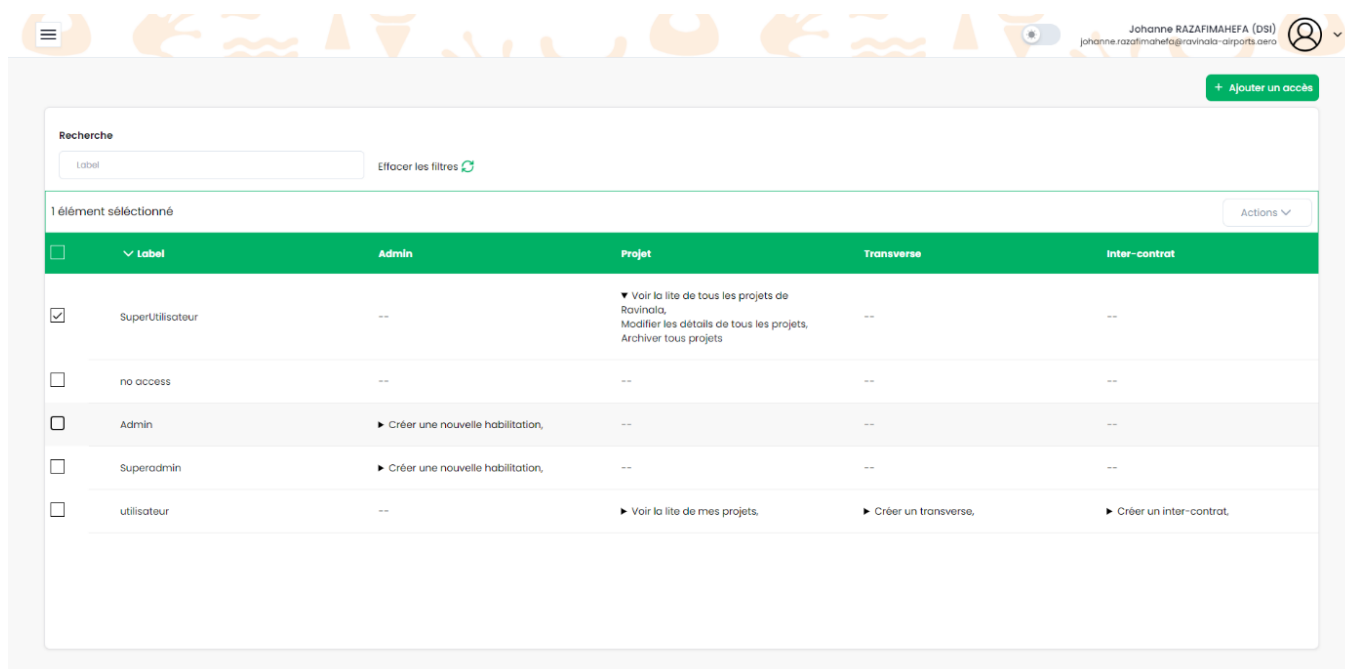


Figure 30. Interface admin, liste des habilitations.



Figure 31 shows the 'Liste de tous les projets' (List of all projects) interface. It features a table with the following columns: Titre, Priorité, Criticité, CDP, Equipes, Date début, Date de fin, and Avancement. The table lists several projects with their respective progress bars and status indicators.

Titre	Priorité	Criticité	CDP	Equipes	Date début	Date de fin	Avancement
ACC : Remodelage de l'utilisation des stations C/UPS	Moyenne	Normale	SR	VR	25/07/2024	30/05/2025	25%
FRET : Ajout module de Vente RAP-CARGO sur ES	Moyenne	Normale	SR		04/12/2023	31/12/2024	50%
Uniformisation salle de réunion	Moyenne	Normale	AR	VR, AR	04/10/2024	31/12/2024	25%
TC : Ajout Caméra ext/Int	Moyenne	Normale	AR	HR, AR, AR	01/10/2024	29/11/2024	25%
TC : Screen FDS extérieur	Moyenne	Normale	AR	AR, VR	01/10/2024	31/10/2024	50%
Séjour : Screen de diffusion	Faible	Normale	AR	VR	27/02/2024	21/11/2024	75%
SAR : mise en redondance unité de stockage des VM	Moyenne	Urgente	AR	VR	05/02/2024	27/11/2024	75%

Figure 31. Interface projet, table projet.

Figure 32 shows the 'Interface tâche par phase de projet' (Task interface by project phase). It displays a task management interface with columns: Backlog, En cours, Traité, En pause, and Abandonné. The 'En cours' column shows tasks like 'Création des activités' and 'Historique de projet'. The 'Traité' column shows tasks like 'Gestion des projets', 'Login', 'Détails d'un projet', and 'Gérer les utilisateurs'.

Backlog	En cours	Traité	En pause	Abandonné
+ ajouter une tâche	Création des activités 14/11/2024	Gestion des projets		
	Historique de projet	Login		
		Détails d'un projet		
		Gérer les utilisateurs		

Figure 32. Interface tâche par phase de projet.

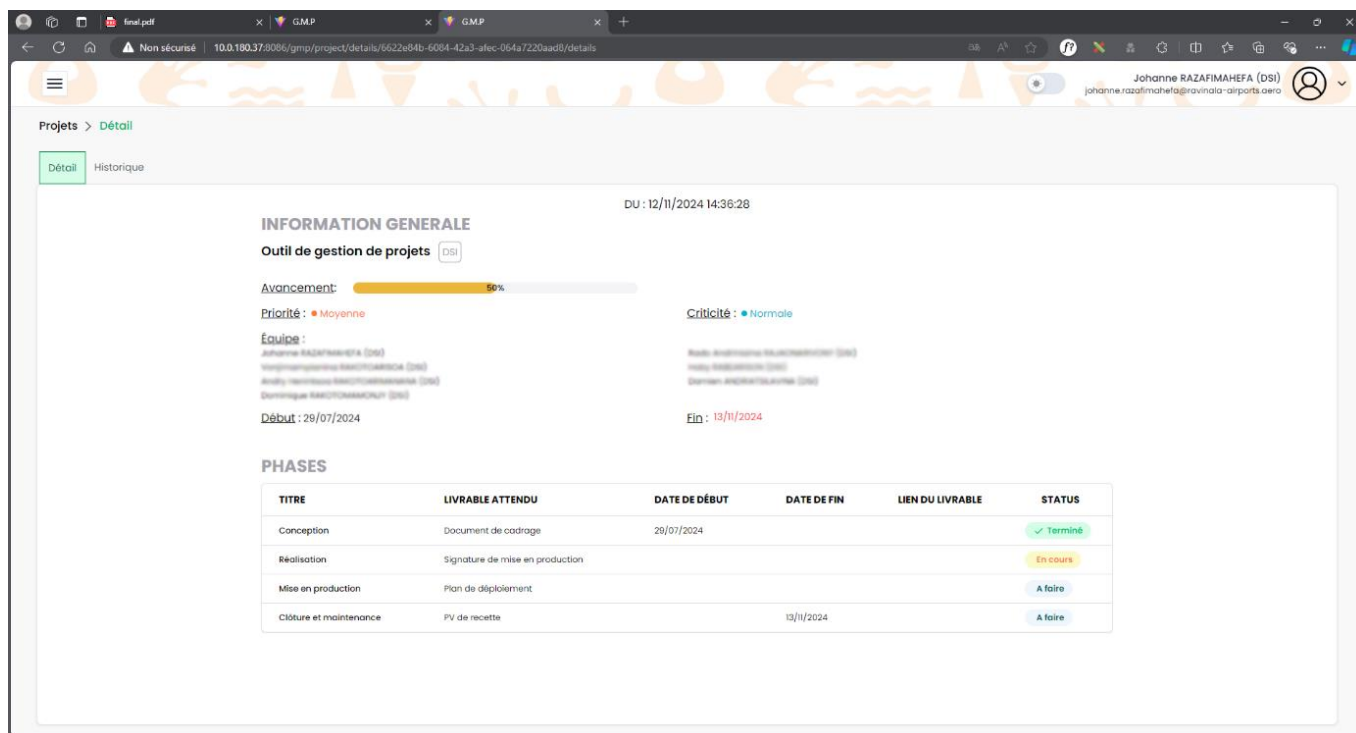


Figure 33. Détails d'un projet.

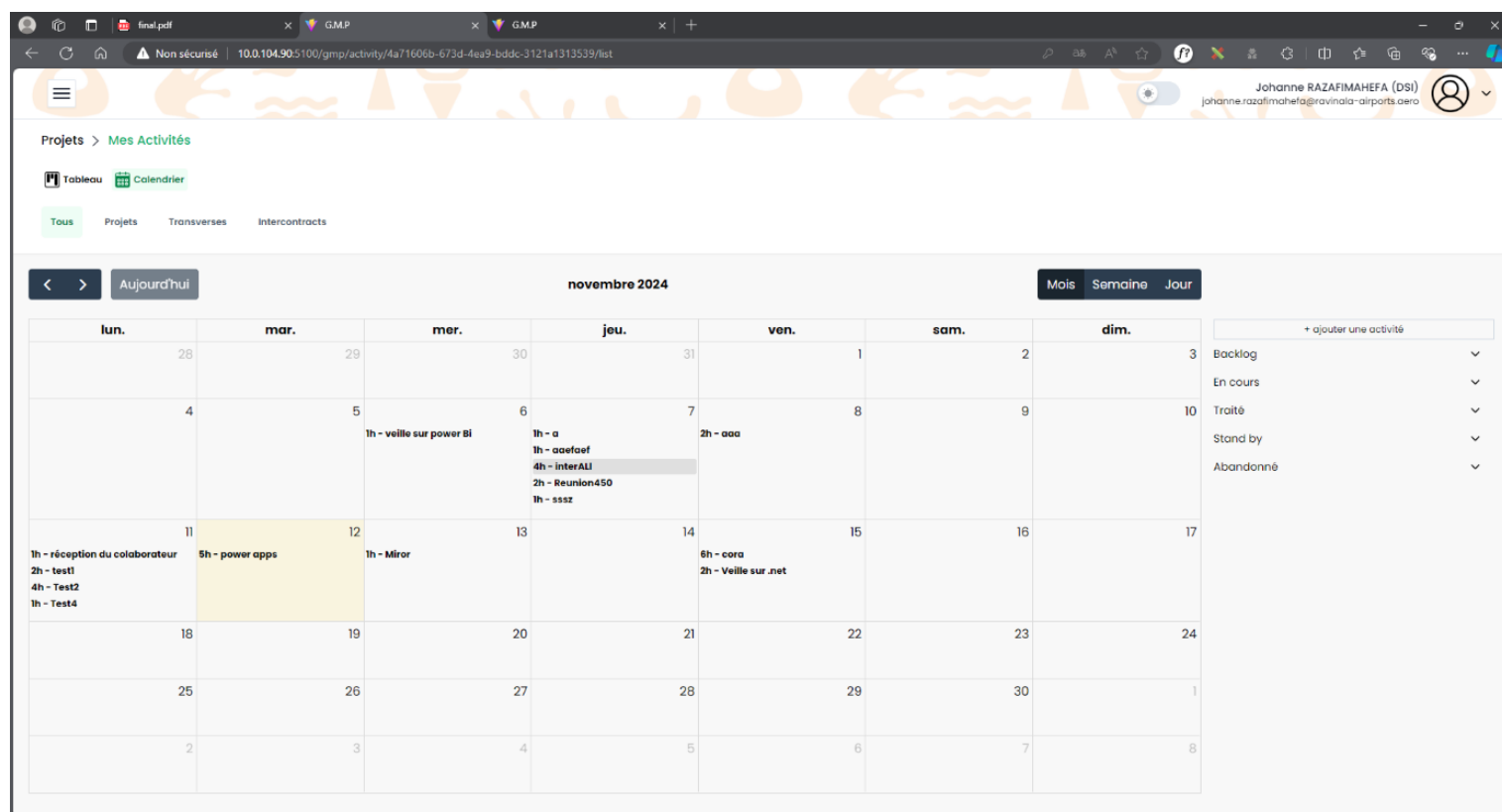


Figure 34. Vue calendrier des activités dans le mois courant.

L'application est déjà tout à fait responsive, comme le montre les figures 35 à 38



Figure 37. Interface login mobile

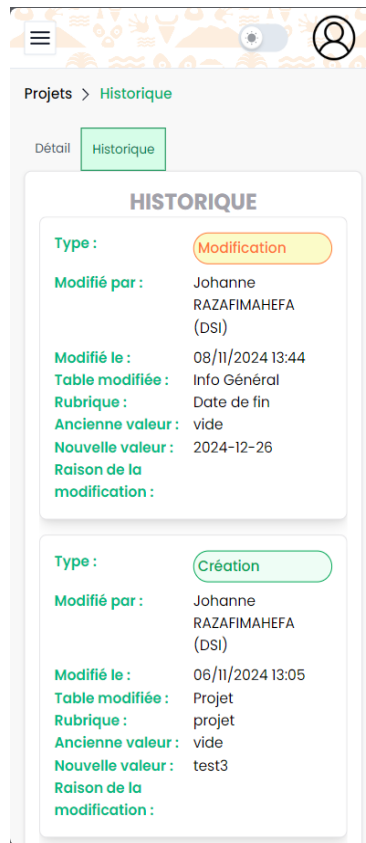


Figure 35. Interface historique, mobile

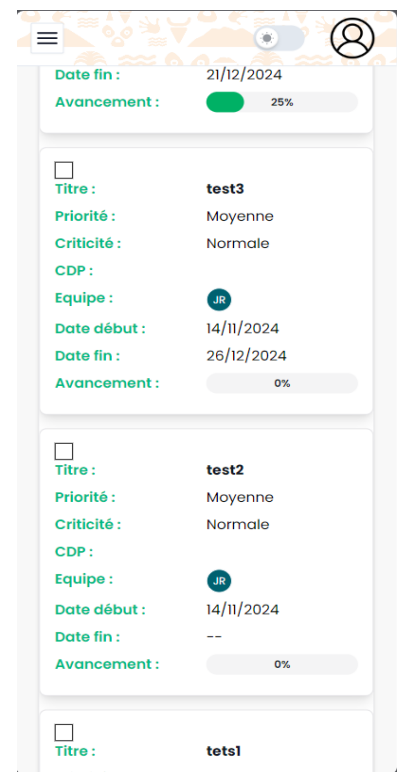


Figure 36. Interface projet, mobile

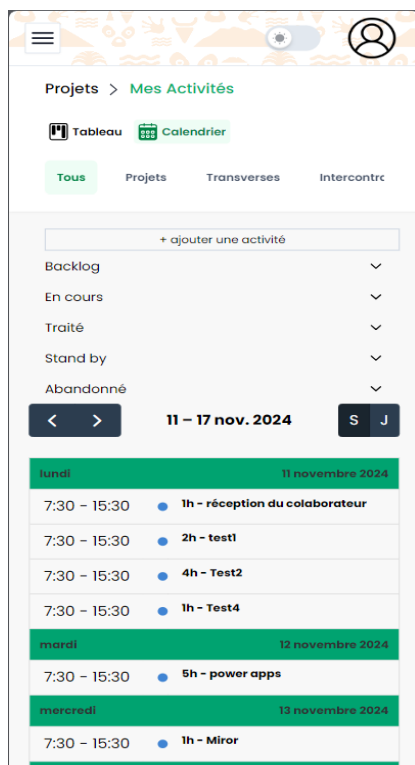


Figure 38. Interface activité, mobile

## CONCLUSION

Ce stage au sein de Ravinala Airports a été très enrichissant et ce fut une occasion de mettre en pratiques les connaissances acquises jusqu'ici. En plus ce stage a permis d'étoffer l'expérience acquis de ce que c'est que la vie en entreprise ainsi que de développer notre épanouissement professionnel.

Au cours de ce stage, nous avons développé un outil de gestion et management de projet qui gère les projets, les ressources et budget qui leurs sont alloués ainsi que les tâches et les personnes qui y sont mandatés. L'outil offre par la même occasion la possibilité de géré les activités hors projets et une visibilité sur les disponibilités de tout un chacun au sein de l'entreprise.

Pour la réalisation de ce projet on a utilisé la méthode de conception Agile couplé avec SCRUM, l'outil de modélisation Visual Paradigm, les langages de programmation .NET coté back et la bibliothèque ReactJs et typeScript coté front end, l'environnement de développement intégré VS Code ainsi que SQL Server en base de données.

L'outil est maintenant déployé et déjà utiliser par le département des systèmes informatiques au sein de l'entreprise. Il est néanmoins envisageable d'encore améliorer ce projet comme le fait de pouvoir relier l'outil avec les outils de gestion de projet précédemment utilisé comme planner, loop ou GLPI.

## BIBLIOGRAPHIE

- [1] **Ken Schwaber and Mike Beedle. 2001.** *Agile Software Development with Scrum.* [ed.] Upper Saddle River. Prentice Hall. NJ : s.n., 2001. p. 158.
- [2] **Ken Schwaber, Jeff Sutherland. 2020.** *The Scrum Guide.* 2020. p. 14.
- [3] **Mike Cohn. 2005.** *Agile Estimating and Planning.* [éd.] Upper Saddle River. NJ : s.n., 2005. p. 368 .

## WEBOGRAPHIE

- [4] <https://www.nutcache.com/fr/blog/quest-ce-que-le-planning-poker-scrum/>, planning poker, consulté le mois de juillet.
- [5] [https://fr.wikipedia.org/wiki/Modele\\_de\\_domaine](https://fr.wikipedia.org/wiki/Modele_de_domaine), Modèle de domaine, consulté le 2 octobre 2024
- [6] <https://www.lucidchart.com/pages/fr/diagramme-de-deploiement-uml>, Qu'est-ce qu'un diagramme de déploiement UML, consulté lors du mois de juillet

## GLOSSAIRE

**Active Directory (AD) :** Service d'annuaire développé par Microsoft pour gérer et centraliser les ressources réseau, les utilisateurs, les groupes, et les permissions dans les environnements Windows. Utilise LDAP comme protocole principal pour l'accès aux données.

**Budget :** Allocation financière destinée à couvrir les coûts liés à un projet ou une activité. Le budget permet de planifier et de contrôler les dépenses pour assurer le bon déroulement des opérations en respectant les contraintes économiques.

**GLPI :** Gestionnaire Libre de Parc Informatique. Un logiciel open source de gestion des services informatiques, utilisé pour l'inventaire des ressources, la gestion des incidents, des demandes et des tâches liées à l'infrastructure informatique.

**Habilitation :** Autorisation ou ensemble de droits accordés à un utilisateur pour accéder à des ressources, des informations ou des fonctionnalités spécifiques d'un système, en fonction de son rôle ou de ses responsabilités.

**Intercontrat :** Un intercontrat désigne la période pendant laquelle un consultant ou un employé n'est pas en mission chez un client. Pendant cette période, l'employé n'a pas de tâche spécifique liée à un contrat client et peut être affecté à d'autres activités internes, telles que les formations, des projets internes ou développement de nouvelles compétences.

**Kanban :** Méthode visuelle basée sur un tableau de tâches (tableau Kanban), où les tâches sont déplacées dans des colonnes représentant différentes étapes de travail.

**LDAP:** Lightweight Directory Access Protocol. Protocole standard pour accéder et gérer des services d'annuaire, utilisé principalement pour l'authentification centralisée et la gestion des utilisateurs dans les réseaux informatiques.

**Livrable :** Un **livrable** est un élément concret ou un résultat qui doit être produit et remis dans le cadre d'un projet. Il représente une étape ou un aboutissement du travail effectué et peut être de différentes formes selon la nature du projet : documents, prototypes, logiciels, rapports ou même plans.

**Poste :** Fonction ou rôle spécifique occupé par une personne au sein d'une organisation, définissant ses responsabilités, ses tâches principales et son positionnement dans la hiérarchie.

**Priorité** : Niveau d'importance ou d'urgence attribué à une tâche, un projet ou une action, permettant d'organiser les activités en fonction de leur impact ou de leur échéance.

**Projet** : Ensemble coordonné d'activités et de ressources mobilisées dans le but d'atteindre des objectifs spécifiques dans un délai défini, souvent avec des livrables à produire. Un projet est généralement structuré en phases et repose sur une planification préalable.

**Tache transverse** : Une tâche transverse, ou mission transversale, désigne une tâche qui traverse plusieurs services ou départements au sein d'une entreprise. Ce type de tâche n'est pas spécifique à un seul domaine mais vise à répondre aux besoins globaux de l'organisation.



## TABLE DES MATIERES

<b>CURRICULUM VITAE .....</b>	<b>I</b>
<b>SOMMAIRE GENERAL .....</b>	<b>III</b>
<b>REMERCIEMENTS.....</b>	<b>VI</b>
<b>LISTE DES FIGURES.....</b>	<b>VIII</b>
<b>LISTE DES TABLEAUX .....</b>	<b>X</b>
<b>LISTES DES ABREVIATIONS .....</b>	<b>XI</b>
<b>INTRODUCTION GENERALE .....</b>	<b>1</b>
<b>PARTIE I : PRESENTATIONS .....</b>	<b>2</b>
<b>Chapitre 1. Présentation de l'Ecole Nationale d'Informatique.....</b>	<b>3</b>
<b>1.1. Information d'ordre générale .....</b>	<b>3</b>
<b>1.2. Missions et historiques .....</b>	<b>3</b>
<b>1.3. Organigramme institutionnel.....</b>	<b>4</b>
<b>1.4. Domaine de spécialisation.....</b>	<b>5</b>
<b>1.5. Architecture des formations pédagogiques.....</b>	<b>6</b>
<b>1.6. Relation de l'ENI avec les organismes externes .....</b>	<b>7</b>
<b>1.7. Débouchés professionnels et diplômés.....</b>	<b>8</b>
<b>1.8. Ressources humaines.....</b>	<b>9</b>
<b>Chapitre 2 : Présentation de Ravinala Airports .....</b>	<b>10</b>
<b>2.1. Information d'ordre général.....</b>	<b>10</b>
<b>2.2. Nos valeurs.....</b>	<b>10</b>
<b>2.3. Nos missions.....</b>	<b>11</b>
<b>2.4. Dates clés.....</b>	<b>11</b>
<b>2.5. Organigramme de Ravinala Airports .....</b>	<b>11</b>
<b>2.6. Nos prix et certifications.....</b>	<b>12</b>
<b>2.7. Nos actionnaires .....</b>	<b>13</b>
<b>2.7.1. Meridiam (65%) .....</b>	<b>13</b>

2.7.2. Groupe ADP (35%) .....	13
<b>Chapitre 3 : Description du projet .....</b>	<b>14</b>
3.1. Formulations.....	14
3.2. Objectif et besoin de l'utilisateur .....	14
3.2.1. Objectif.....	14
3.2.2. Besoins de l'utilisateur .....	14
a) Besoins fonctionnels .....	14
b) Besoins non-fonctionnel .....	15
3.3. Moyen nécessaire à la réalisation du projet .....	15
3.3.1. Moyen matériel.....	15
3.3.2. Moyen logiciel .....	15
3.3.3. Moyen humain .....	16
3.4. Résultats attendus .....	16
3.5. Chronogramme de travail .....	16
<b>PARTIE II : ANALYSE ET CONCEPTION .....</b>	<b>18</b>
<b>Chapitre 4 : Analyse préalable .....</b>	<b>19</b>
4.1. Analyse de l'existant .....	19
4.1.1. Organisation actuelle .....	19
4.1.2. Inventaire des moyens matériels et logiciels .....	19
4.3. Conception avant-projet.....	20
4.3.1. Proposition de solutions .....	20
4.3.2. Méthodes de conception et outils utilisés.....	21
a) Méthode de conception .....	21
b) Langage de programmation et outil utilisé.....	25
<b>Chapitre 5. Analyse conceptuelle .....</b>	<b>26</b>
5.1. Désignation des rôles de l'équipe SCRUM .....	26
5.2. Etapes de l'élaboration du Product Backlog .....	26

5.2.1. Etablir la vision du projet .....	26
5.2.2. Lister les acteurs .....	26
5.2.3. Regroupement de fonctionnalités .....	27
5.2.4. Recueil des exigences .....	28
5.2.5. Priorisation des exigences complexité .....	28
5.3. Product Backlog .....	28
5.4. Sprint Backlog .....	30
5.4.1. Présentation du travail à faire .....	30
5.4.2. Planification du sprint .....	32
5.5. Dictionnaire des données .....	34
5.6. Règles de gestion .....	39
5.7. Modèle du domaine .....	40
5.8. Diagramme de classe de conception pour chaque sprint .....	41
5.8.1. Sprint 1 : User et habilitation .....	41
5.2.2. Sprint 2 : Création des projets .....	42
5.8.3. Sprint 3 : Gestion des tâches .....	43
5.8.4. Sprint 4 : Activité .....	44
5.9. Diagramme de classe de conception globale .....	45
5.10. Diagramme de paquetage .....	47
5.11. Diagramme de déploiement .....	49
<b>PARTIE III. REALISATION .....</b>	<b>51</b>
<b>Chapitre 6. Mise en place de l'environnement de développement .....</b>	<b>52</b>
<b>6.1. Installation et configuration des outils .....</b>	<b>52</b>
6.1.1. Installation de SQL server .....	52
6.1.2. Installation de Visual Paradigm .....	53
6.1.3. Installation de SDK .NET .....	54
6.1.4. Installation de Vs Code .....	54

6.2. Architecture de l'application .....	55
Chapitre 7. Développement de l'application .....	56
7.1. Création de la base de données .....	56
7.2. Codage de l'application .....	58
7.2.1. Le coté backend .....	58
7.2.2. Le coté front end.....	63
7.3. Présentation de l'application .....	64
CONCLUSION.....	XIII
BIBLIOGRAPHIE .....	XIV
WEBOGRAPHIE.....	XV
GLOSSAIRE.....	XVI
TABLE DES MATIERES .....	XVIII
RESUME.....	XXII
ABSTRACT .....	XXII

## RESUME

Ce document est le fruit de plusieurs étapes pour la création d'un outil de gestion et de management de projet ; ces étapes passent de la conception à la réalisation et finit par le déploiement. Pour la réalisation de ce projet, nous avons utilisé Agile en méthode de conception, .NET et react comme langage de programmation, Visual Paradigm pour la modélisation des diagrammes lors de la conception, et vs code comme éditeur de texte.

Cet outil est mis au point suite à une manque de visibilité des projets en cours et des activités des collaborateurs. Les outils et composants utilisé dans ce projet priorise la sécurité, la scalabilité et la compatibilité avec les outils Microsoft.

**Mots-clés** : active directory, activité, gestion de projet, projet, tâche.

## ABSTRACT

This document is the result of several stages to create a tool for project management; those stages go from the conception to the realization and end with the deployment. For the realization of this project, we used Agile for conception method, .NET and react as programming languages, Visual paradigm as modelling tool for the diagram in the conception phase, and vs code as Integrated Development Environment.

This tool is developed cause of a lack of visibility in the ongoing project and the activities of all collaborators. The tools and components used in this project prioritize security, scalability and compatibility with Microsoft tool.

**Keywords**: active directory, activity, project management, project, task

