



République Tunisienne  
الجمهورية التونسية  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
وزارة التعليم العالي والبحث العلمي  
Institut Supérieur des Etudes Technologiques de Bizerte  
المعهد العالي للدراسات التكنولوجية ببزرت

---

## Rapport de stage de fin d'études

Pour l'obtention du Licence Appliquée en Développement des Systèmes  
d'Information

Option : Développement des Systèmes d'Information

---

# Conception et réalisation d'une application mobile de gestion électronique de document

---

*Réalisé par :*

CHARRAD Raed

CHLENDI Mohamed youssef

*Encadré par :*

Mme. ALLOUCHE Najla (ISETB)

M. MEJAI Rabii (NeoLedge)

PÉRIODE DE STAGE :

Du 06/02/2023 au 28/05/2023

ORGANISME D'ACCUEIL :

NeoLedge

LIEU DE STAGE :

NeoLedge, Zarzis, Tunisie

TÉLÉPHONE :

+216 \*\*\*\*\*

EMAIL :

contact@neolledge.com

Promotion : 2022/2023

# Table des matières

<b>Introduction générale</b>	<b>1</b>
<b>1 Cadre général du projet</b>	<b>3</b>
<b>Cadre général du projet</b>	<b>3</b>
1 Introduction	3
2 Présentation de la société d'accueil	3
2.1 Historique	3
2.2 Produits présentés par NeoLedge	4
2.3 Contexte du projet	4
2.4 Motivation	5
3 Étude de l'existant	5
3.1 Description de l'existant	5
3.2 Critique de l'existant	10
3.3 Solution proposée	10
4 Méthodologie de gestion de projet	11
4.1 Choix de la méthodologie du développement	11
4.2 Pourquoi Scrum	11
5 Environnement de développement	12
5.1 Environnement matériel	13
5.2 Environnement logiciel	13
5.3 Technologies utilisées	14
6 Conclusion	15
<b>2 Analyse et conception</b>	<b>16</b>
<b>Analyse et conception</b>	<b>16</b>
1 Introduction	16
2 Analyse	16
2.1 Identifications des acteurs	16
2.2 Spécifications des besoins fonctionnels	16
2.3 Spécifications des besoins non fonctionnels	19
3 Conception	19
3.1 Diagramme de cas d'utilisation	19
<b>Annexes</b>	<b>20</b>

# Table des figures

1.1	Logo de NeoLedge . . . . .	3
1.2	Logo de Elise . . . . .	5
1.3	Capture d'écran de Elise . . . . .	6
1.4	Capture d'écran de Elise Mobile . . . . .	9
1.5	Cycle de vie Scrum . . . . .	12

# Liste des tableaux

1.1	Environnement matériel . . . . .	13
2.1	Acteurs de l'application . . . . .	17

# Introduction générale

Un organisme public ou privé acquiert et produit tout au long de son activité un grand nombre de documents. Certains sont vitaux (les titres de propriété, les contrats ...), et doivent être conservés pour répondre à l'environnement réglementaire. D'autres encore, les documents dits « de travail » tels que les comptes-rendus, les rapports, les documents bureautiques, peuvent être consultés dans le but de prendre une décision. Par conséquent, la gestion et la conservation des documents au sein de l'organisme sont des activités essentielles, elles l'offrent un gain de temps et d'argent et une centralisation des documents et sécurisation des données ce qui est offert par le système de gestion électronique du document GED.

Pour cela les logiciels de GED est devenu aujourd'hui la meilleure solution pour faire la gestion électronique des documents avec un minimum d'effort.

C'est dans ce cadre que se situe notre projet de fin d'études effectué au sein de la société NeoLedge pendant la période de 06/02/2022 au 27/05/2022 pour l'obtention du diplôme de licence national en technologie d'information.

Il s'agit en fait de concevoir et développer une application mobile « Cross-Platform » Permettant d'offrir aux utilisateurs de l'application « Elise Web » les mêmes fonctionnalités. Cette solution permettra la gestion et la conservation des documents qui sont des activités essentielles, de plus elle permet de faciliter le travail collaboratif.

Le présent rapport synthétise ainsi le déroulement de notre travail sur ce projet. Il est structuré en cinq chapitres :

Le premier chapitre sera consacré au contexte général du projet. Nous allons, tout d'abord, Présenter la société dans laquelle a été réalisée notre application, ainsi, nous allons définir notre mission et les objectifs à atteindre avec l'analyse et la critique de l'existant. Ensuite, nous allons écrire la méthodologie de développement adoptée et nos choix pour la modélisation conceptuelle.

Le deuxième chapitre présentera l'analyse et la conception en détail répondant aux exigences fonctionnelles et non fonctionnelles en se basant sur la méthodologie SCRUM.

Le troisième chapitre présentera les sprint 1 & 2 « Gestion d'authentification » & « Gestion de profil ». Nous allons, tout d'abord, présenter pour chaque sprint le sprint Backlog et le diagramme de cas d'utilisation détaillée. Ensuite nous allons définir le diagramme de classe et le diagramme de séquence. La même démarche pour le quatrième chapitre « Gestion de tableaux de bord & de documents ».

Le cinquième chapitre illustre les outils technologiques et matériels utilisés ainsi les diagrammes de composants et déploiement spécifique pour notre application puis les structures et les interfaces développés durant ce stage. Enfin, nous clôturerons notre travail par une conclusion générale et les éventuelles perspectives

# Chapitre 1

## Cadre général du projet

### 1 Introduction

Dans ce chapitre, nous nous intéresserons à l'introduction du cadre global du projet et à la description des exigences. Il s'agit en fait d'une introduction de l'organisme d'accueil, suivie d'une motivation et de la présentation du projet. Ensuite, nous présentons l'étude de l'existant et la solution proposée.

### 2 Présentation de la société d'accueil

Acteur indépendant et à dimension internationale, NeoLedge est une société française en forte croissance, qui s'appuie sur un réseau de partenaires privilégiés en Europe, en Amérique du Nord et en Afrique. Éditeur spécialisé dans la gestion électronique de documents, NeoLedge compte à son actif des centaines de clients, des dizaines de milliers d'utilisateurs quotidiens et des millions de documents gérés par ses solutions, dans le secteur public comme dans le secteur privé.

Partenaire certifié Gold de Microsoft en ce qui concerne le développement d'applications et les plateformes cloud, NeoLedge accompagne des organisations pendant leur transition numérique. NeoLedge s'appuie depuis ses débuts sur les techniques développées par Microsoft.



FIG. 1.1 : Logo de NeoLedge

#### 2.1 Historique

Archimed est un éditeur de logiciels français, créé en 1993 par trois fondateurs (Mongi Zidi, Olivier Walbecq et Eric Ruyffelaere) et dont le siège social est établi à Lille.

Archimed est un éditeur indépendant spécialisé dans les logiciels de gestion documentaire depuis près de 25 ans. Elle a étendu son activité ECM ces dernières années à travers l'Afrique, l'Europe et l'Amérique du Nord.

Afin d'accompagner son développement à l'international, la société a lancé NeoLedge, une nouvelle marque exclusivement dédiée à ses solutions et activités ECM. Le nom NeoLedge a été inspiré par les "new knowledge" - une nouvelle façon de gérer le contenu, d'optimiser le temps et de gagner en productivité.

## 2.2 Produits présentés par NeoLedge

NeoLedge propose une gamme complète de solutions de gestion électronique de documents (GED) et de gestion de contenus (ECM) pour les entreprises et les administrations.

- **ECM Elise** : Pour les entreprises de toutes tailles, qui sont confrontées à une gestion complexe des flux d'informations nécessitant de nombreuses interactions avec des tiers, la solution ECM Elise, éditée par NeoLedge, propose des fonctions automatisées pour la capture des flux multicanaux, la gestion des dossiers et l'orchestration des processus métiers.
- **DocFactory** : Pour les entreprises à la recherche d'une solution performante capable de les aider dans leur transformation numérique et leur passage au zéro papier, DocFactory est une solution numérique tout-en-un offrant des fonctions centralisées de numérisation, conversion, et capture de documents électroniques, ainsi que d'export et de signature électronique.  
Contrairement aux solutions propriétaires ne gérant pas le processus de dématérialisation dans sa globalité, DocFactory permet d'industrialiser l'ensemble des opérations de capture des flux de documents jusqu'à leur intégration dans votre système d'information.
- **Illico** : Pour les élus et dirigeants de villes de toutes tailles, engagées dans une politique de transition numérique grâce à des outils innovants qui simplifient leur fonctionnement, illico offre aux citoyens et aux équipes de la ville des services de confiance pour la dématérialisation, le traitement des requêtes, la gestion du courrier et l'automatisation des processus métiers. Contrairement aux solutions en silos ne gérant pas la continuité des processus entre les citoyens et les agents, "illico powered by Elise" est un service illimité, rapide à mettre en œuvre, conçu pour les villes et les réseaux de villes.

## 2.3 Contexte du projet

Le présent projet intitulé « Elise Mobile » est réalisé dans le cadre de projet de fin d'études pour l'obtention du diplôme de licence national en technologie d'information au sein de l'Institut Supérieur des Etudes Technologiques de Bizerte.

Ce projet a été réalisé dans la société NeoLedge durant la période s'étalant du 06 février 2022 jusqu'à 28 mai 2022



### 2.4 Motivation

Comme nous l'avons déjà indiqué, la société d'accueil dispose d'une application web de Gestion Électronique de Documents (GED) intitulée Elise et qui est développée avec les technologies Vue.js pour la partie front et C# pour la partie back fournissant des API Rest et services web SOAP. Cette application est commercialisée depuis presque 20 ans sous plusieurs versions, dont la sixième version actuellement. La conception actuelle de l'application ne permet pas aux utilisateurs d'accéder aux différentes informations (documents) pour les nomades, que ce soit via des smartphones/tablettes sous iOS/Android. L'objectif de ce projet est alors la mise en place d'une solution mobile responsive du Elise tout en gardant les mêmes exigences fonctionnelles, non fonctionnelles et les expériences utilisateurs. Cette application permettra aux utilisateurs d'Elise d'accéder à leurs profils à travers leurs smartphones/tablettes.

## 3 Étude de l'existant

Dans cette section, nous allons présenter l'étude de l'existant, c'est-à-dire l'étude de l'application Elise actuelle et de ses fonctionnalités.

### 3.1 Description de l'existant

La solution existante de Gestion Électronique de Documents a plusieurs fonctionnalités très utiles pour l'organisation. En effet, une fois le document répertorié, il est possible de le retrouver très facilement avec certains critères de recherche prédéfinis. C'est ainsi qu'un gain de temps considérable se fait ressentir, lorsque nous avons l'habitude de rechercher nos documents papier dans des dizaines de classeurs. En ce qui concerne l'indexation des documents, le système prend en charge une reconnaissance automatique des informations présentes sur ce document et assure ainsi un classement optimisé et automatisé.

Désormais, les entreprises sont toujours à la recherche des solutions permettant de développer et faciliter le travail collaboratif. En effet, chaque collaborateur de la société peut avoir accès ou non à tel ou tel document. C'est ainsi que des processus de validation / modification / lecture peuvent être définis, permettant de favoriser le travail commun, et cela même à distance.

C'est pour cela que NeoLedge a proposé une solution GED qui est Elise.



FIG. 1.2 : Logo de Elise

Elise est une solution verticalisée capable de prendre en charge tous les flux circulants dans l'organisation (entrants, sortants, internes, etc.), et ce, quel que soit le canal (mail, courrier, etc.). Sa spécificité réside dans sa capacité à gérer les échanges de l'organisation avec son écosystème (partenaires, fournisseurs, citoyens...) par le biais de l'intégration d'une base de tiers, de métadonnées adaptées (expéditeur/destinataire/dates d'échéances...) et de processus de réponse facilités.

Elise peut ainsi gérer des processus transverses et s'appuie sur une modélisation de l'organigramme de la structure concernée (collectivités, administrations, etc.) pour permettre un paramétrage précis des circuits d'acheminement. Ces circuits peuvent bien entendu être modifiés, à tout moment, en cours de route, pour s'adapter à des cas particuliers, fréquents dans le domaine de la Gestion du courrier.

Un service dans Elise est un élément de structure de l'organigramme représentant un groupement d'utilisateurs et/ou de services subordonnés. Les services peuvent être de plusieurs types :

- **Service interne** : service interne non joignable directement par une entité d'une autre instance Elise.
- **Service interne joignable** : service interne joignable directement par une entité d'une autre instance Elise.
- **Service externe** : service appartenant à une autre instance Elise.

L'unité d'organisation dans Elise est un élément de l'organigramme pouvant contenir des utilisateurs et services mais uniquement utilisé dans le but de structurer l'organigramme. Contrairement à un service, il est impossible d'affecter des tâches à une unité d'organisation.

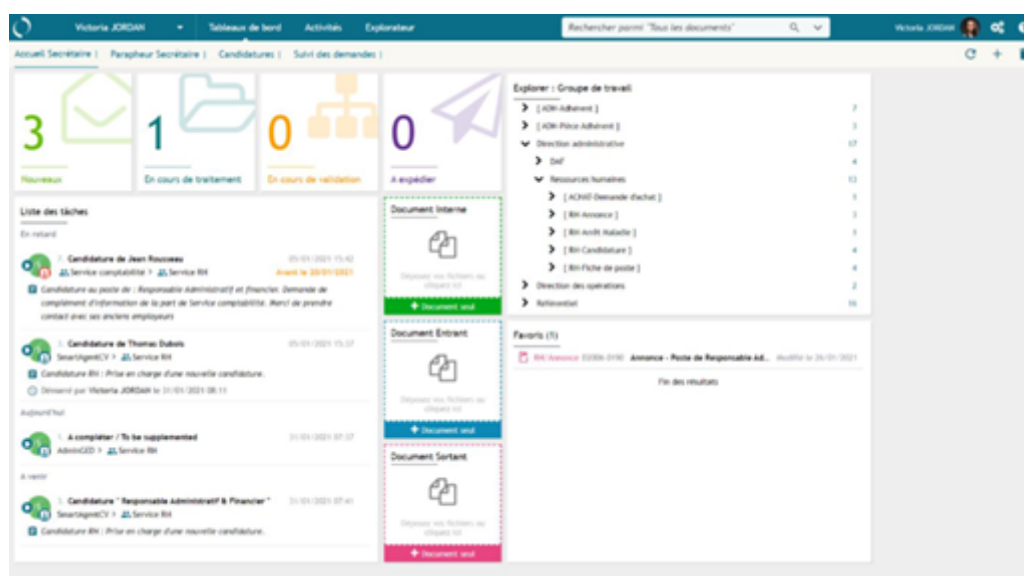


FIG. 1.3 : Capture d'écran de Elise

L'espace de travail Elise est un regroupement de services et d'utilisateurs. Les espaces de travail permettent de rassembler les documents pour lesquels les utilisateurs et services de l'espace de travail ont une tâche à effectuer.

Il existe trois types d'espaces de travail sous Elise :

- **Espace de travail par défaut** : Cet espace de travail est composé du service principal d'utilisateur et de lui-même.
- **Espaces de travail personnels** : Espaces de travail que l'utilisateur peut créer et configurer lui-même selon ses besoins.
- **Espaces de travail de services** : Espaces de travail prédéfinis que l'utilisateur peut ajouter dans leur liste d'espaces de travail pour en faciliter l'accès.

Le tableau de bord d'Elise est une partie personnalisable de l'interface qui permet de proposer des vues de l'espace de travail en fonction des besoins.

Il existe deux types de tableaux de bord sous Elise :

- **Tableaux de bord personnels** : que l'utilisateur peut créer, modifier ou supprimer selon ses besoins.
- **Tableaux de bords partagés** : prédéfinis et déployés par l'administrateur.

Un tableau de bord d'Elise est un ensemble de widgets. Où un widget est un élément de tableau de bord personnalisable afin d'afficher des informations et/ou de créer une interface vers une fonctionnalité de l'application.

Les différents types de widgets sont :

- **Widget compteur** : qui définit les nombres des documents selon leurs types existants dans le courrier.
- **Widget graphique** : qui représente les statistiques de type de documents.
- **Widget liste de documents** : qui définit les documents selon leurs types existants dans le courrier.
- **Widget liste des tâches** : qui définit les tâches affectées aux utilisateurs.
- **Widget note** : qui définit les notes de l'utilisateur.
- **Widget nouveau document** : qui permet à l'utilisateur d'ajouter un document.
- **Widget page web**
- **Widget vue dynamique**
- **Widget raccourci** : qui est un raccourci à un document ou à un site web.

Dans Elise, un document correspond à un enregistrement complet. Un document contient les éléments suivants :

- **Propriétés du document** : Les propriétés sont des données contenant les informations associées à un document.
- **Fichiers** : Fichiers numériques liés au document et qui sont éditables par les utilisateurs.
- **Les tâches** : Les tâches sont des actions à effectuer sur des documents.
- **Un processus** : est un ensemble de tâches définissant les actions à réaliser sur un document.
- **Les liens** : Propriétés de type URL qui sont ajoutées soit par un automate soit manuellement.
- **Historique des événements** : Les événements gardent la trace de toutes les actions effectuées sur le document.

Les différents états du document sont les suivants :

- **Sans affectation** : État des documents qui ont été créés mais pour lesquels l'ensemble des propriétés obligatoires n'est pas renseigné et/ou aucune tâche n'est encore attribuée. (Cas documents numérisés non encore enregistrés)
- **Préparé** : État des documents pour lesquels l'ensemble des propriétés obligatoires est renseigné et au moins une tâche est attribuée. Le changement d'état depuis l'état « Préparé » vers l'état « en cours de traitement » se fait en distribuant le document.
- **En circulation** : État des documents pour lesquels il y a des tâches actives en cours de traitement et qui n'ont pas été clôturés. Le changement d'état depuis l'état « En circulation » vers « Clôturé » se fait en clôturant le document.
- **Clôturé** : État des documents pour lesquels toutes les tâches qui devaient être effectuées sont terminées et qui ont été clôturés. Le changement d'état depuis l'état « Clôturé » vers l'état « Archivé » se fait en archivant le document.
- **Archivé** : État des documents pour lesquels la période d'utilité dans Elise est terminée et qui ne nécessitent donc plus d'être accessibles par les utilisateurs.

NeoLedge a une version actuelle du Elise Mobile qui se trouve déjà sur le marché du mobile hébergée dans Play store et App store.

Elise Mobile est une application mobile pour plateformes iOS et Android :

- **Simple à utiliser** : Elise Mobile est une application intuitive et facile à utiliser. Elle permet de gérer les documents et les tâches depuis n'importe quel endroit et à tout moment.
- **Facile à déployer** : Elise Mobile est une application qui ne nécessite pas de configuration particulière. Elle est disponible sur les plateformes iOS et Android.



FIG. 1.4 : Capture d'écran de Elise Mobile

- **Au ROI immédiat** : Elise Mobile permet de gagner du temps et de l'argent en permettant aux utilisateurs de gérer leurs documents et leurs tâches.

Elle a été développée en Xamarin avec Backend en Asp.Net Core 4 depuis le 10 octobre 2018. Elle est en version 1.1.6 jusqu'à présent.

Avec Elise Mobile, les utilisateurs de la solution de Gestion électronique de Contenu Elise peuvent :

- **Collaborer à distance** :
  - Accéder à l'ensemble de leurs tâches selon leur état de traitement (Nouveaux, En cours, Terminés...)
  - Superviser leurs processus métiers
  - Effectuer des recherches dans la base documentaire
- **Décider en quelques instants** :
  - Traiter leurs tâches
  - Valider/Refuser rapidement les actions demandées
  - Demander/Transférer des tâches
  - Respecter les délais
- **Superviser leurs espaces de travail** :
  - Accéder aux espaces de travail de leurs différents services
  - Visualiser les tâches en cours o Contrôler les échéances

### 3.2 Critique de l'existant

L'étude de l'application existante nous a permis de porter des jugements objectifs afin de projeter de fournir ainsi un système plus fiable.

Parmi les lacunes rencontrées, nous citons :

- Elise est une solution développée de base en ASP. Net Back et front de l'ancienne version d'Asp. Net c'est pour cela l'application actuelle est non responsive et c'est à partir de la version 6 de Elise que NeoLedge a décidé de migrer le front vers les nouvelles technologies.
- Elise utilise une base de données locale qui est SIMWebPack et qui est une solution proposée par NeoLedge pour enregistrer les documents sous la forme de XML.
- La version actuelle d'Elise mobile ne répond pas aux besoins des utilisateurs par rapport à Elise web ce qui invoque de mauvaises expériences utilisateurs.
- La version actuelle d'Elise mobile ne renferme pas toutes les fonctionnalités du Elise version web.

### 3.3 Solution proposée

Notre solution consiste à mettre en place une solution responsive de Elise qui offre les mêmes fonctionnalités que la version web actuelle d'Elise tout en gardant les mêmes interfaces pour améliorer l'expérience d'utilisateur. Parmi les fonctionnalités qu'on va traiter, on peut citer :

- Connecter un utilisateur de Elise à son profile
- Consulter les espaces de travail
- Consulter les tableaux de bord
- Créer un nouveau document
- Accéder à un document
- Rechercher un document / un service ou un utilisateur spécifique
- Signer des documents
- Gestion des documents
  - Possibilité d'approuver et de signer des documents.
  - Signature possible en utilisant des signatures antérieures ou manuellement.
  - Possibilité d'utiliser des signatures électroniques avancées, telles que des signatures numériques basées sur des certificats numériques
- Notifications push : Possibilité de recevoir des notifications push pour les mises à jour de documents ou les tâches à effectuer.

# 4 Méthodologie de gestion de projet

Avec les progrès en technologie de l'information et les investissements dans les infrastructures, beaucoup de méthodes de gestion de projet ont vu le jour. Certes, ces méthodes jouent un rôle primordial dans la réussite ou l'échec d'un projet, d'où le choix, représente une décision importante pour les entreprises. Dans cette partie, nous allons expliquer notre choix de la méthode. Il existe en fait deux grandes familles de méthodes de la gestion du projet : les méthodes dites classiques et les méthodes agiles. On privilégiera plutôt les méthodes classiques lorsqu'on a une idée précise du projet, avec un planning bien détaillé et où on a anticipé tous les risques possibles. Quant aux méthodes Agiles, on les choisira plutôt pour les gros projets, celles-ci permettent une meilleure adaptabilité, visibilité et gestion des risques. On privilégiera également les méthodes Agiles pour les projets où il n'y a pas de documents détaillés, ou quand le client est indécis. Le client pourra alors voir l'évolution du projet et l'adapter à ses besoins sans pour autant vous obliger à recommencer tout le travail que vous avez fourni depuis le début. Elles sont menées dans un esprit collaboratif et elles s'adaptent aux approches incrémentales. Elles engendrent des produits de haute qualité tout en tenant compte de l'évolution des besoins du client.

## 4.1 Choix de la méthodologie du développement

Notre choix méthodologique et en convenance à notre projet, nous avons opté pour l'utilisation de la méthode SCRUM. C'est en fait la méthode la plus utilisée, la plus éprouvée, documentée et supportée. Il s'agit d'un cadre de travail en équipe pluridisciplinaire permettant de répondre à des problèmes complexes et changeant, centré sur l'architecture.

## 4.2 Pourquoi Scrum

Dans le cadre de notre projet et afin d'assurer le bon déroulement des différentes phases de ce dernier, nous avons opté pour la méthode agile Scrum pour la conception et le développement de notre système pour des raisons bien déterminées. En effet, le processus Scrum s'adapte parfaitement à la décomposition du notre projet de fin d'études.

Il se base sur les avantages suivants :

- Plus de souplesse et de réactivité.
- Grande capacité d'adaptation au changement grâce à des itérations courtes.
- Satisfaire au mieux les besoins du client.

SCRUM est une méthodologie agile qui consiste à avoir une équipe soudée orientant le projet au fil de son avancement afin d'atteindre un but. Cette approche est à la fois dynamique et productive, engendre la réalisation des fonctionnalités par itération en incluant la participation du client. Chaque itération peut durer de deux à quatre semaines, à la fin de chaque sprint un produit fonctionnel doit être livré. En effet, Scrum définit trois rôles qui sont :

- **Product Owner** (Le gestionnaire de produit) : Le responsable du produit de l'équipe projet client et il représente les utilisateurs finaux. C'est lui qui va définir et prioriser la liste des fonctionnalités du produit et choisir la date et le contenu de chaque sprint sur la base des valeurs (charges) qui lui sont communiquées par l'équipe.
- **Scrum Master** (Le maître de Scrum) : Véritable facilitateur sur le projet, il veille à ce que chacun puisse travailler au maximum de ses capacités en éliminant les obstacles et en protégeant l'équipe des perturbations extérieures.
- **L'équipe de développement** (L'équipe de projet) : L'équipe de réalisation contient au minimum deux développeurs. Elle regroupe tous les rôles habituellement nécessaires à un projet, à savoir l'architecte, le concepteur, le développeur, le testeur, etc.

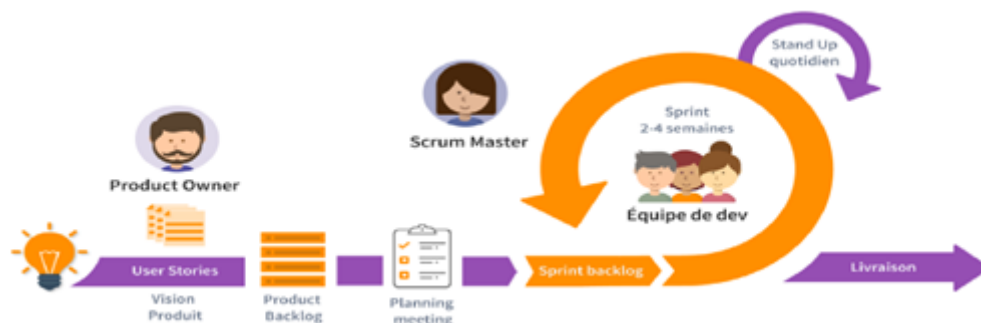



FIG. 1.5 : Cycle de vie Scrum

## 5 Environnement de développement

Avant d'entamer la mise en œuvre de notre projet, nous allons d'abord décrire l'environnement et les outils de travail que nous utiliserons. Nous commencerons par définir l'environnement matériel, suivi de l'environnement logiciel. Enfin, nous présenterons les différents langages et frameworks que nous utiliserons dans le cadre de ce projet.



## 5.1 Environnement matériel

Environnement matériel	Description
	<p><b>PC DELL Latitude 5510</b></p> <ul style="list-style-type: none"> <li>• <b>Quantité</b> : 2</li> <li>• <b>Processeur</b> : Intel Core i7-10610U</li> <li>• <b>Mémoire</b> : 16 Go DDR4</li> <li>• <b>Stockage</b> : 512Go SSD</li> <li>• <b>Carte graphique</b> : NVIDIA GeForce MX250</li> <li>• <b>Système d'exploitation</b> : Windows 10</li> </ul>

TAB. 1.1 : Environnement matériel

## 5.2 Environnement logiciel

Au cours de cette section, nous dresserons une liste des outils utilisés pour étudier et mettre en place notre application tout au long du projet.

- **Windows 10 :**



Windows 10 est un système d'exploitation de Microsoft. Cette nouvelle version introduit plusieurs changements importants. Elle est la première à fonctionner sur toutes les plateformes existantes.

- **Visual Studio Code :**



Un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré.

- **Visual Studio :**



Microsoft Visual Studio est une suite de logiciels de développement pour Windows et macOS conçue par Microsoft

- **Android Studio :**



Android Studio est un environnement de développement intégré (IDE) pour le développement d'applications Android. Il est basé sur IntelliJ IDEA et est développé par Google.

- **Postman :**



Postman est un outil de développement de logiciels qui permet de tester et de déboguer des API. Il permet de créer des requêtes HTTP et de les envoyer à un serveur. Il permet également de créer des collections de requêtes et de les organiser en dossiers.

- **Git :**



Git est un logiciel de gestion de versions décentralisé. Il permet de gérer les versions d'un projet informatique et de collaborer avec d'autres développeurs.

- **Azure DevOps :**



Azure DevOps est un ensemble d'outils de développement logiciel en tant que service (SaaS) fourni par Microsoft. Il fournit des services de gestion de projets, de gestion de code source, de gestion de tests et de gestion de la qualité du logiciel.

- **Figma :**



Figma est un outil de conception graphique. Il permet de créer des maquettes de sites web et d'applications mobiles.

- **Draw.io :**



Draw.io est un logiciel de dessin graphique multiplateforme gratuit et open source développé en HTML5 et JavaScript. Son interface peut être utilisée pour créer tous types de diagrammes.

- **Microsoft Teams :**



Microsoft Teams est un logiciel de messagerie instantanée et de collaboration en ligne développé par Microsoft. Il permet de créer des salons de discussion, de partager des fichiers et de collaborer en temps réel.

- **Latex :**



LaTeX est un langage de description de documents. Il permet de créer tous types de documents. Il est utilisé pour la rédaction de livres, de rapports, de thèses, de journaux, de présentations, de documents techniques, etc.

## 5.3 Technologies utilisées

- **Ionic :**



Ionic est un UI toolkit open source pour la création d'applications mobiles performantes et de haute qualité à l'aide de technologies web telles que HTML, CSS et JavaScript, avec des intégrations pour des frameworks populaires tels que Angular, React et Vue.

- **Vue3 :**



Vue.js est un framework JavaScript open source pour la création d'interfaces utilisateur et d'applications web monopages.

- **Capacitor :**



Capacitor est un runtime natif open source pour la création d'applications Web natives. Il permet de créer des applications iOS, Android et des applications Web progressives multiplateformes à l'aide de JavaScript, HTML et CSS.

- **Typescript :**



TypeScript est un langage de programmation libre et open source développé par Microsoft. Il est un sur-ensemble de JavaScript, c'est-à-dire qu'il ajoute des fonctionnalités à ce dernier.

## 6 Conclusion

Au cours de ce chapitre qui constitue une étape primordiale pour fixer les repères de notre projet. Nous avons présenté l'organisme d'accueil et les attentes du projet. En effet, nous avons mené une étude de l'existant afin de mieux cerner les fonctionnalités de notre solution révélant les limites de la solution existante. Nous avons également déterminé le cadre du travail ainsi que la méthodologie à emprunter lors de ce projet.

Dans le chapitre qui suit nous allons décrire la conception de l'application en détaillant ses spécifications, ses acteurs et ses différents diagrammes de cas d'utilisation, de séquences, de classes, et d'activités.

# Chapitre 2

## Analyse et conception

### 1 Introduction

Afin de pouvoir implémentées les fonctionnalités recensées au début de ce projet, nous s'intéressons dans ce chapitre à l'étude conceptuelle de notre application. C'est une phase de spécification et de modélisation conceptuelle basée sur le langage UML à travers les diagrammes de cas d'utilisation, les diagrammes de séquences et classes. Ceci nous permet de tracer une meilleure stratégie d'implémentation des besoins fonctionnels tout en respectant les contraintes identifiées. Ensuite, nous exposerons les besoins non fonctionnels ainsi que notre backlog de produit.

### 2 Analyse

Dans cette phase d'analyse nous allons identifier les acteurs de notre application et les différents besoins fonctionnels et non fonctionnels.

#### 2.1 Identifications des acteurs

Un utilisateur est une entité extérieure au système de modélisation qui représente et interagit directement avec une personne, un appareil. Chaque acteur dispose d'un ensemble d'actions correspondant à la fonction dont il a besoin. Dans notre projet

L'application « Elise » Mobile fait intervenir plusieurs acteurs comme le montre le tableau ci-après.

Ces rôles/droits sont définis et appliqués par défaut par le système. Il est possible de les modifier selon les besoins spécifiques du client.

#### 2.2 Spécifications des besoins fonctionnels

Notre application est destinée aux utilisateurs suivants : les collaborateurs, les secrétaires et les chefs de service. Chacun d'eux dispose d'un ensemble d'actions correspondant

Acteur	Rôles
<b>Collaborateur</b>	Accède aux documents qui lui sont partagés, peut les modifier et les partager selon les tâches qui lui sont assignées
<b>Secrétaire</b>	Même rôle que le collaborateur avec en plus l'accès aux documents partagés par les autres de son service et les documents partagés par les autres services des collaborateurs
<b>Chef de service</b>	Même rôle que le secrétaire avec en plus l'accès aux documents partagés par les autres services subalternes

TAB. 2.1 : Acteurs de l'application

à la fonction dont il a besoin.

### Besoins fonctionnels des collaborateurs

- Authentification
- Consultation de profil
- Modification de profil
- Declaration d'absence
- Annulation d'absence
- Affectation de delegé
- Suppression de delegé
- Recherche d'un utilisateur/service
- Reception de notification
- Désactivation de notification
- Consulter espace de travail utilisateur
- Consulter espace de travail personnalisé
- Supprimer espace de travail personnalisé
- Modifier espace de travail personnalisé
- Ajouter un espace de travail personnalisé
- Accéder aux espaces de travail de ses différents services
- Consulter tableau de bord partagé et personnel
- Consulter les widgets du tableau de bord personnel
- Ajouter un widget dans le tableau de bord personnel

- Supprimer un widget dans le tableau de bord personnel
- Configurer un widget du tableau de bord personnel
- Ajouter un tableau de bord personnel
- Supprimer un tableau de bord personnel
- Modifier un tableau de bord personnel
- Rechercher un document selon le filtre
- Accéder à l'ensemble de ses tâches
- Superviser son processus métier
- Démarrer une tâche dans un document
- Modifier informations d'un document
- Consulter les fichiers d'un document
- Supprimer un fichier d'un document
- Ajouter un fichier à un document
- Consulter l'historique d'un document
- Consulter les tâches d'un document
- Supprimer une tâche d'un document
- Ajouter une tâche à un document
- Consulter l'historique d'une tâche
- Consulter les signatures
- Ajouter une signature
- Supprimer une signature
- Signer un document manuellement
- Signer un document à l'aide des signatures enregistrées

### 2.3 Spécifications des besoins non fonctionnels

Afin d'optimiser le fonctionnement de notre application, elle doit répondre aux différents besoins non fonctionnels présentés ci-dessous que nous devons prendre en compte afin d'assurer une meilleure utilisation et une meilleure gestion :

- **Performance** : L'application doit être capable de répondre aux besoins des utilisateurs en temps réel.
- **Fiabilité** : L'application doit être fiable et ne pas présenter de dysfonctionnement, et les données fournies par l'application doivent être fiables.
- **Usabilité** : L'application doit être facilement utilisable et compréhensible par les utilisateurs.
- **Sécurité** : L'application doit être sécurisée et ne pas présenter de faille de sécurité.
- **Disponibilité** : L'application doit être disponible 24h/24 et 7j/7.
- **Maintenabilité** : L'application doit être facilement maintenable.
- **Interopérabilité** : L'application doit être compatible avec les différents systèmes d'exploitation.
- **Portabilité** : L'application doit être facilement portable.
- L'application doit connecter au serveur d'application privée de l'entreprise.

## 3 Conception

### 3.1 Diagramme de cas d'utilisation

# Annexes