

# Dédicace

“

*Je tiens à exprimer ma profonde gratitude et mes sincères  
remerciements à tous ceux qui me sont chers :*

**XXXXXXXX**

*qui a été ma source d'inspiration tout au long de cette année.  
Sa présence inébranlable dans ma vie a été un soutien  
inestimable.*

***A mes chers parents***

*Leur amour inconditionnel a été la fondation sur laquelle j'ai  
pu construire ma vie. Ils m'ont toujours encouragée à suivre  
mes rêves et à ne jamais abandonner face aux difficultés.*

**À XXXXXXXXX**

*mes amis les plus proches et mes plus grands alliés. Leurs  
encouragements, leurs conseils et leur amour sincère ont été  
essentiels à mon épanouissement.*

”

# Remerciements

Je souhaite, tout d'abord, adresser les remerciements les plus sincères à mon encadrante :

*Mme Sonia SAHLI*

Professeur à l'École xxxxxxxxx

pour ne jamais m'avoir refusé, pour sa patience avec moi, pour ses mots encourageants et pour ses conseils et son expérience qu'elle n'a jamais cessé de partager avec moi.

*Mme XXXXXX*

*M. YYYYYYYY*

pour l'excellent travail qu'ils font afin de faciliter la vie à tous les étudiants. Veuillez trouver ici l'expression de tout mes respects et mes gratitude profonde

Que les *membres de jury* trouvent, ici, l'expression de mes remerciements pour l'honneur qu'ils me font en prenant le temps de lire et d'évaluer ce travail.

# Table des matières

<b>Introduction générale</b>	<b>7</b>
<b>1 Etude Préalable</b>	<b>8</b>
Introduction . . . . .	9
1.1 Présentation de l'organisme d'accueil . . . . .	9
1.1.1 IT gate . . . . .	9
1.1.2 services de IT . . . . .	9
1.2 Projet . . . . .	10
1.3 Etude de l'existant . . . . .	10
1.3.1 Solutions existantes . . . . .	10
1.3.2 Critique de l'existant . . . . .	11
1.4 Solution proposée . . . . .	11
1.4.1 Objectifs de la solution . . . . .	12
1.4.2 Description du projet . . . . .	12
1.5 Processus de développement . . . . .	12
1.5.1 Méthodologie de développement . . . . .	12
1.5.2 Présentation de la méthode SCRUM . . . . .	13
Conclusion . . . . .	13
<b>2 Analyse et spécification des besoins</b>	<b>14</b>
Introduction . . . . .	15
2.1 Identification des acteurs . . . . .	15
2.2 Spécification des besoins . . . . .	15
2.2.1 Besoins fonctionnels . . . . .	15

---

2.2.2	Besoins non fonctionnels . . . . .	16
2.3	Diagramme des cas d'utilisation général . . . . .	17
2.4	Conception de l'interface Utilisateur . . . . .	20
	Conclusion . . . . .	21
<b>3</b>	<b>Étude Conceptuelle</b>	<b>22</b>
	Introduction . . . . .	23
3.1	Diagrammes des cas d'utilisations raffinés . . . . .	23
3.1.1	xxxxxxxxxxxxx . . . . .	24
3.1.2	yyyyyyyyyyyyy . . . . .	25
3.1.3	zzzzzzzzzzzzzzz . . . . .	25
3.2	Diagramme de séquences . . . . .	26
3.3	Diagramme des classes de conception . . . . .	28
	Conclusion . . . . .	28
<b>4</b>	<b>Réalisation</b>	<b>29</b>
	Introduction . . . . .	30
4.1	Environnement de travail . . . . .	30
4.2	Architecture adoptée . . . . .	32
4.3	Enchaînement des interfaces . . . . .	34
	Conclusion . . . . .	37
	<b>Conclusion générale</b>	<b>38</b>
	<b>Webographie</b>	<b>39</b>

# Table des figures

1.1	Logo bbbbbb AnyInIT . . . . .	9
1.2	Logo de la solution 1 . . . . .	10
1.3	Logo de la solution 2. . . . .	11
1.4	Logo de la solution 3. . . . .	11
1.5	Gestion de projet méthode agile scrum . . . . .	13
2.1	Besoins fonctionnels à travers le Diagramme des cas d'utilisations . . . . .	18
2.2	Diagramme de cas d'utilisation global . . . . .	19
2.3	Besoins fonctionnels à travers des maquettes . . . . .	20
2.4	Maquette de l'interface d'ajout d'un compte manager . . . . .	20
2.5	Maquette de l'interface d'ajout d'un compte manager . . . . .	21
3.1	Processus de conception . . . . .	23
3.2	Diagramme de cas d'utilisation "xxxxxx" . . . . .	24
3.3	Diagramme de cas d'utilisation "yyyyyyyyyy" . . . . .	25
3.4	Diagramme de cas d'utilisation "yyyyyyyyyy" . . . . .	25
3.5	Diagramme de séquence "nnnnnnnnnn" . . . . .	27
3.6	Diagramme de classes de conception . . . . .	28
4.1	Logo Lucidchart . . . . .	30
4.2	Logo Figma . . . . .	30
4.3	Logo Next.js . . . . .	31
4.4	Logo Node.js . . . . .	31
4.5	Logo Overleaf . . . . .	31

---

4.6	Logo MongoDB . . . . .	32
4.7	Logo Visual Studio Code . . . . .	32
4.8	Architecture MVC . . . . .	33
4.9	Interface AAAAAA . . . . .	34
4.10	Interface de BB . . . . .	35
4.11	DDDDDDDDDD . . . . .	36
4.12	Interface CCCCC . . . . .	37

# Introduction générale

Ici parler de theme de sujet d'une maniere generale : Gestion RH, Mobile , IA ,..... .so-  
metexte sometexte sometexte sometexte sometexte sometexte sometexte sometexte some-  
texte sometexte sometexte sometexte sometexte sometexte sometexte sometexte sometexte  
sometexte sometexte sometexte sometexte sometexte sometexte.ICI lancer la problemati-  
queICI lancer la problematiqueICI lancer la problematique

Suite à ce besoin de xxxxx vient l'idée de notre projet de fin d'étude Plateforme de  
xxxxxx.

En tant qu'étudiante a l'école YYYYYYYYYYYY. Ce stage a été effectué au sein de  
la Société XXXXXXXXXXXX durant la période XXXXXX au YYYYYYYY afin d'analyser  
XXXX objectif du projet XXXXX.

En vue de rendre compte de manière fidèle et analytique des cinq mois passés au sein de  
l'entreprise XXXX, il apparaît logique d'organiser le présent rapport en quatre chapitres  
structurés comme suit :

Le premier chapitre présente le cadre général du projet en présentant toute une étude  
de projet, l'analyse de l'existant et la solution proposée ainsi que la méthodologie utilisée  
lors de la période de réalisation. Le deuxième chapitre est consacré à l'étude fonctionnelle  
et technique tout en reliant les acteurs et leurs besoins avec les solutions proposées ainsi  
qu'une partie qui définit les cas d'utilisations générales. Le troisième chapitre a pour objectif  
de présenter la conception suivie dans le développement de la solution y inclus les cas  
d'utilisation raffinées, des diagrammes de séquences et le diagramme de classe. Le dernier  
chapitre met l'accent sur la réalisation du projet en présentant les différents outils utilisés  
et quelques prises d'écrans de l'outil.

# Chapitre 1

## Etude Préalable



## Introduction

Dans ce chapitre nous commençons par présenter l'organisme d'accueil, notre sujet, une étude comparative entre plusieurs applications existantes et présenter ensuite notre projet et son cahier de charge ainsi que la méthodologie adoptée afin de réaliser ce travail.

### 1.1 Présentation de l'organisme d'accueil

#### 1.1.1 IT gate

#### 1.1.2 services de IT

Fondée depuis 201x, IT Gate Group est un acteur régional dans le secteur de l'IT, spécialisé dans les solutions d'entreprise, les applications mobiles, les solutions e-commerce, les applications industrielles embarquées, les objets connectés, les réseaux sociaux et les systèmes des gestions de contenu, fournissant un savoir-faire concurrentiel, des services et des solutions sur mesure à ses clients en Afrique du Nord, en Europe et au Moyen-Orient via ses différentes entités basées en Tunisie, France, Algérie. XXXXXX repose sur des équipes expérimentées, innovantes et multilingues avec des expertises techniques diversifiées. Les équipes XXXXXX ont forgé leur expertise dans la mise en place de stratégies IT performantes afin de réussir les différents types de projets de leurs clients. XXXXXX base sa relation client sur l'écoute et la réactivité dans le but de correspondre ses solutions aux besoins de ses clients.



FIGURE 1.1 – Logo bbbbbb AnyInIT

## 1.2 Projet

## 1.3 Etude de l'existant

Cette section vise à étudier les solutions existantes. Cette recherche permet de souligner les avantages et les inconvénients de chaque solution. Nous présentons ensuite notre solution.

Notre travail va être destiné au xxxxxx, c'est dans ce cadre que nous allons étudier les applications disponibles xxxxxx. On va distinguer leurs avantages et points faibles.

### 1.3.1 Solutions existantes

- **solution xxxx** : xxx est un serveur de stockage d'objets open source conçu pour stocker et gérer des données non structurées, notamment des fichiers, des photos, des vidéos, etc. Il est compatible avec l'API Amazon S3, ce qui facilite l'intégration des applications à xxxxxx en tant que backend de stockage alternatif. Les différents logiciels de Minio sont open source, téléchargeables et utilisables gratuitement. L'éditeur, Minio génère ses revenus via des prestations de support auprès de grands clients MinIO offre une Intégration Limitée à l'écosystème, Configuration Complexes et une Surveillance et Gestion Exigeantes.

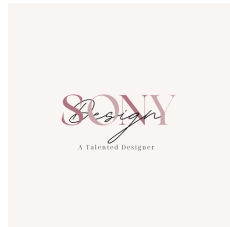


FIGURE 1.2 – Logo de la solution 1

- **solution zzzzzzz** : zzzzzzzzz est un serveur de stockage d'objets open source conçu pour stocker et gérer des données non structurées, notamment des fichiers, des photos, des vidéos, etc. Il est compatible avec l'API Amazon S3, ce qui facilite l'intégration des applications à xxxxxx en tant que backend de stockage alternatif. Les différents logiciels de Minio sont open source, téléchargeables et utilisables gratuitement. L'éditeur, Minio génère ses revenus via des prestations de support auprès de grands clients MinIO offre une Intégration Limitée à l'écosystème, Configuration Complexes et une Surveillance et Gestion Exigeantes.



FIGURE 1.3 – Logo de la solution 2.

- **solution yyyy** : yyyyyy appartient à la catégorie des solutions de stockage d'objets distribuées et offre une gestion des données à grande échelle. La technologie d'yyyyyy déjà séduit plus d'une quarantaine de clients dans le monde, parmi lesquels Dailymotion. Elle repose sur une plate-forme distribuée avec un socle open source travaillant avec les API Amazon S3 et OpenStack Swift. La plate-forme d'OpenIO assure le chiffrement et la compression des données mais pas la déduplication. L'inconvénient de OpenIO, qu'il offre une complexité de configuration et un support limité par rapport aux options commerciales.



FIGURE 1.4 – Logo de la solution 3.

### 1.3.2 Critique de l'existant

xxx est un serveur de stockage d'objets open source conçu pour stocker et gérer des données non structurées, notamment des fichiers, des photos, des vidéos, etc. Il est compatible avec l'API Amazon S3, ce qui facilite

## 1.4 Solution proposée

Le projet vise à ..... some textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome textesome .

ici decrire votre projet et ses avantages .

### 1.4.1 Objectifs de la solution

Le projet vise à atteindre les objectifs suivants :

- Permettre au personnel administratif de numériser facilement et rapidement les documents papier en fichiers numériques.
- Mettre en place une interface utilisateur conviviale et intuitive pour faciliter la numérisation en un seul clic.
- Assurer la sécurité des documents numérisés en mettant en place des protocoles de sécurité avancés.
- Optimiser l'utilisation de l'espace de stockage physique grâce à la conversion des fichiers papier en fichiers numériques.
- Faciliter l'accès aux documents numérisés depuis n'importe quel appareil connecté à Internet.

### 1.4.2 Description du projet

Notre projet consiste à mettre en place une plateforme web pour les personnels administratifs afin de digitaliser leurs documents de manière simple et rapide. L'objectif principal est de développer une solution interactive avec un scanner, permettant aux utilisateurs d'héberger les documents dans le cloud en un seul clic.

## 1.5 Processus de développement

### 1.5.1 Méthodologie de développement

Etant donné que le temps de réalisation de ce projet était relativement serré, nous avons choisi d'utiliser la méthodologie agile afin de mieux organiser le temps et de continuer à produire des incréments à chaque sprint. Les méthodologies agiles sont devenues très populaires dans le domaine du développement de logiciels et de gestion de projets, car elles apportent de nombreux avantages par rapport aux méthodes traditionnelles de gestion de projet.

### 1.5.2 Présentation de la méthode SCRUM

La méthode Agile Scrum est un cadre de gestion de projet très populaire et largement utilisé dans le développement de logiciels et d'autres domaines de gestion de projets. L'utilité de la méthode Agile Scrum réside dans plusieurs aspects clés qui permettent d'améliorer l'efficacité et la flexibilité des équipes de développement.. La figure 1.5 représente la gestion de projet avec la methode scrum.

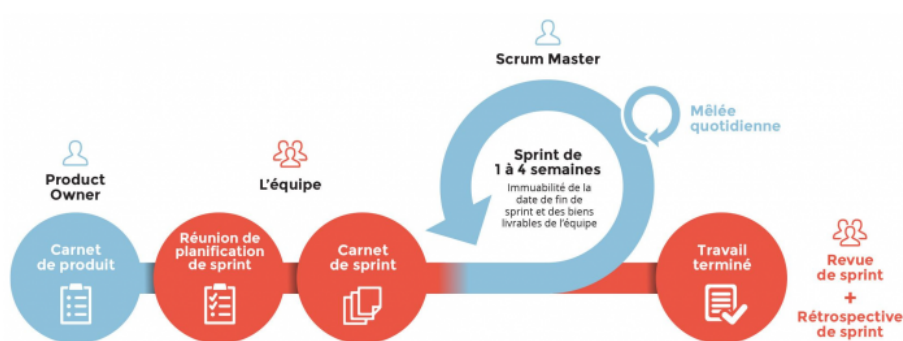


FIGURE 1.5 – Gestion de projet méthode agile scrum

## Conclusion

Ce chapitre a été consacré pour avoir une vue analytique du projet. Nous avons commence par présenter l'organisme d'accueil ensuite nous avons expliqué le processus actuel de stockage des fichiers administratifs tout en donnant un aperçu sur les solutions existantes qui ont initialement inspiré ce projet. Nous avons décrit dans la section suivante notre solution et ses objectifs tout en décrivant la méthodologie adopté lors du développement de la solution.

## Chapitre 2

### Analyse et spécification des besoins

## Introduction

Le travail qui nous a été confié pendant notre stage consiste à décrire votre sujet décrire votre sujet décrire votre sujet décrire votre sujet décrire votre sujet décrire votre sujet décrire votre sujet décrire votre sujet décrire votre sujet.

Pour ce faire, il est indispensable de réaliser une analyse générale. En premier temps, nous allons dégager tous les besoins fonctionnels ainsi que les besoins non fonctionnels. Ensuite, nous allons détailler la branche supérieure concernant la modélisation des cas d'utilisation après avoir identifié les acteurs.

## 2.1 Identification des acteurs

Certains des principaux acteurs sont impliqués dans un système qui aide le personnel de l'administration à numériser ses documents à l'aide d'un scanner. Les interactions et collaborations entre ces acteurs permettent une numérisation et une gestion efficace et efficace des documents.

- **Acteur 1**

Description du rôle de l'acteur 1, ses privilèges , il gère la tâche X , il fait la tâche Y.

- **Acteur 2**

Description du rôle de l'acteur 2, ses privilèges , il gère la tâche X , il fait la tâche Y.

- **Acteur 3**

Description du rôle de l'acteur 3, ses privilèges , il gère la tâche X , il fait la tâche Y.

## 2.2 Spécification des besoins

### 2.2.1 Besoins fonctionnels

Dans cette partie , nous allons détailler les besoins fonctionnels de notre projet a savoir :

- **Besoin fonctionnel N1**

Cette fonctionnalité implique la possibilité de some text some text some text





— *Déconnexion après temps mort d'inactivité, durées, actions.*

### Disponibilité

Il est indispensable que l'application soit disponible à tout moment 24/24, 7/7 sauf période de maintenance. Il est aussi indispensable que les données soient disponibles même en périodes de fortes activités.

### Performance

Une application doit être avant tout performante. En d'autres termes, via ses fonctionnalités, elle se doit de répondre à toutes les exigences des utilisateurs et surtout celle qui représente l'une des fonctionnalités qui avantage notre application par rapport au marché à savoir la recherche et ceci d'une manière optimale et dans un délai précis.

### L'ergonomie et la convivialité

La plateforme doit fournir des interfaces compréhensibles, simples, conviviales et exploitables par l'utilisateur. Les informations doivent être présentées d'une façon lisible, facile et rapidement accessible.

## 2.3 Diagramme des cas d'utilisation général

Dans cette partie nous allons décrire les besoins fonctionnels a l'aide d'un diagramme des cas d'utilisations comme indique la figure ci dessous.

Le diagramme des cas d'utilisations ci dessous illustre les interactions entre les acteurs X , Y . some text some text some text some text some text some text some text some text some text some text some text some text some text.

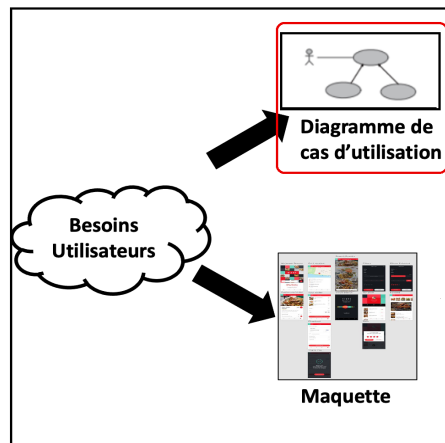


FIGURE 2.1 – Besoins fonctionnels à travers le Diagramme des cas d'utilisations

images/global.png

FIGURE 2.2 – Diagramme de cas d'utilisation global

## 2.4 Conception de l'interface Utilisateur

Cette étape consiste à comprendre le contexte du système à l'aide de conception des interfaces ou Wireframing.

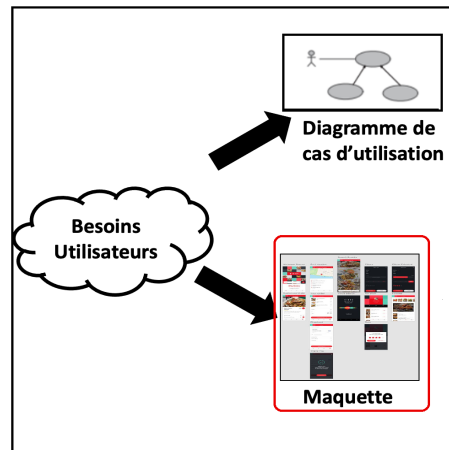


FIGURE 2.3 – Besoins fonctionnels à travers des maquettes

- *Interface d'authentification* some text some text some text some text some text some text some text some text some text some text .

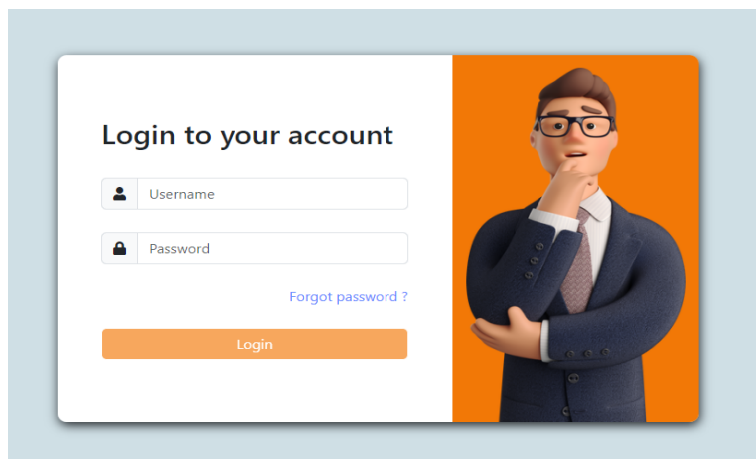


FIGURE 2.4 – Maquette de l'interface d'ajout d'un compte manager

- *Tache 2* some text some text some text some text some text some text some  
text some text text some text some text some text some text.

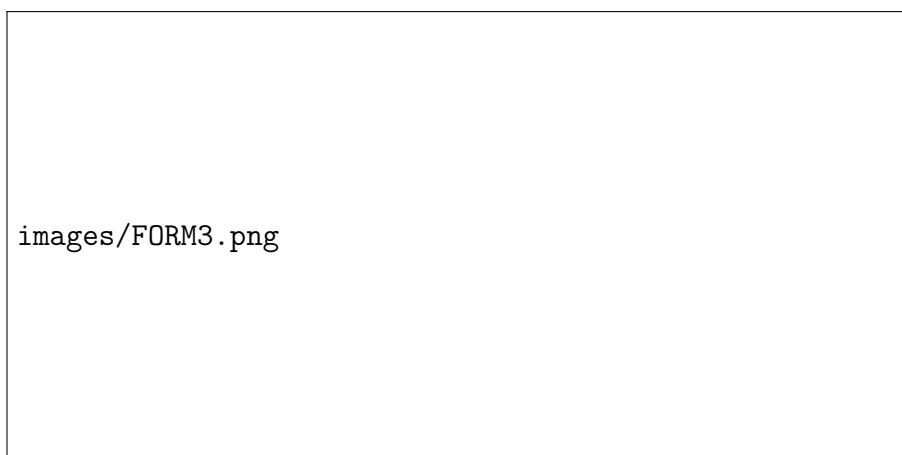


FIGURE 2.5 – Maquette de l’interface d’ajout d’un compte manager

## Conclusion

Ce chapitre a été consacré pour présenter les exigences fonctionnelles et les exigences non fonctionnelles de notre application. Notre solution de numérisation peut transformer les documents administratifs en documents numériques, en optimisant l’efficacité opérationnelle, en améliorant l’accessibilité aux informations et en renforçant la sécurité des données. Nous avons ensuite identifié les acteurs intervenant dans ce projet et en fin nous avons présenté le diagramme de cas d’utilisation global. Dans le chapitre suivant, on va mettre l’accent sur l’étude conceptuelle de l’application.

## Chapitre 3

### Étude Conceptuelle

## Introduction

Dans ce chapitre, nous analysons en profondeur les concepts essentiels qui sous-tendent le développement de la solution. L'analyse conceptuelle fera l'objet de ce chapitre en utilisant les diagrammes des cas d'utilisations raffinées, diagramme de classe et quelques diagrammes de séquence.

### 3.1 Diagrammes des cas d'utilisations raffinés

Pour chaque acteur identifié dans le deuxième chapitre, il est impératif d'explorer en détail les différentes intentions professionnelles qui orientent leur utilisation des systèmes. Commençons par présenter de manière approfondie les cas d'utilisation raffinés.

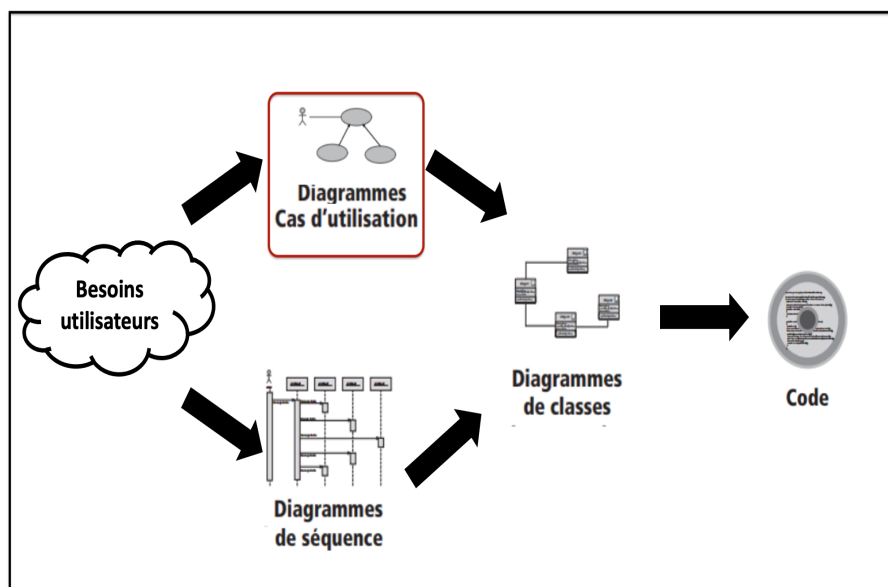
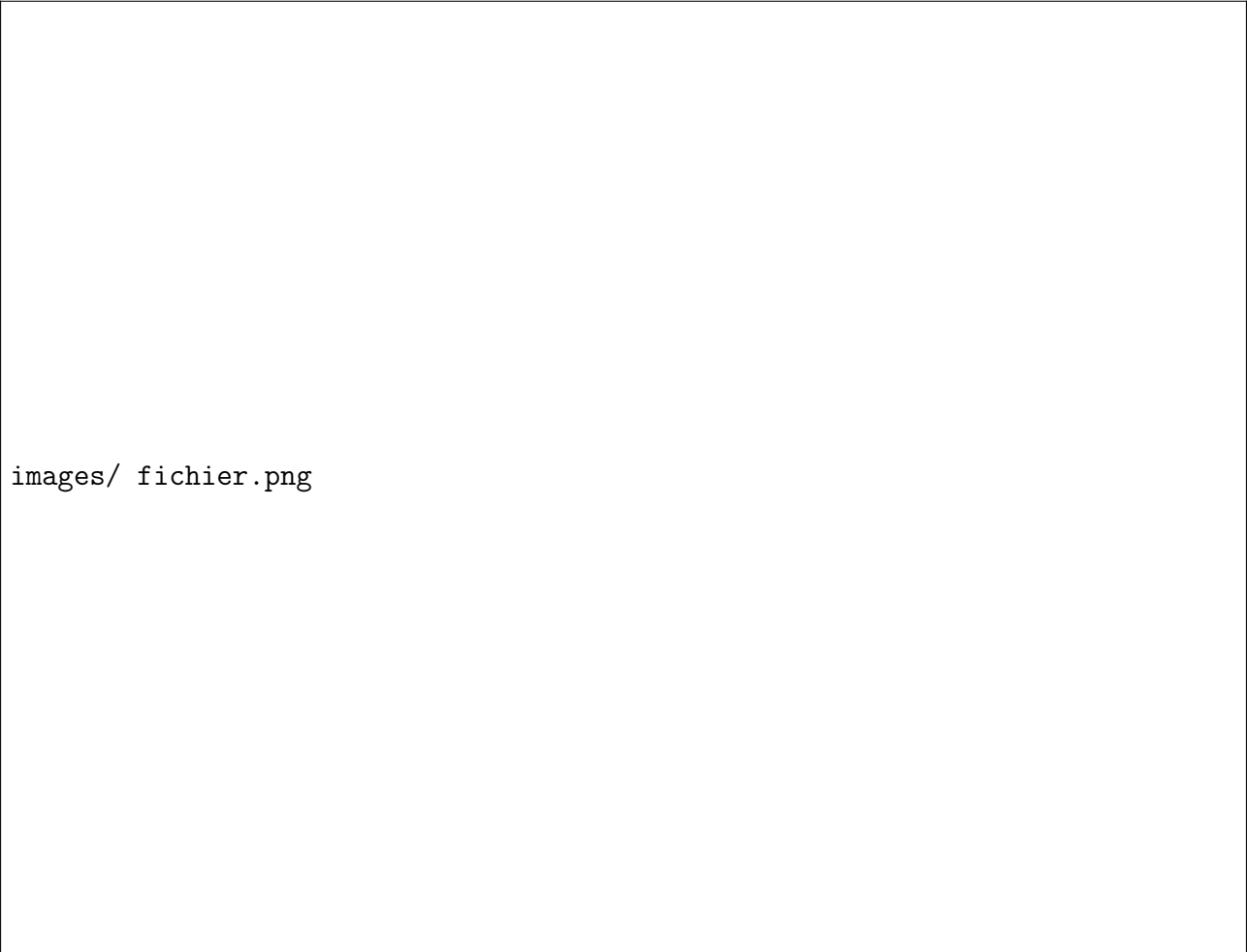


FIGURE 3.1 – Processus de conception

### 3.1.1 xxxxxxxxxxxx

La figure ci dessous décrit le processus de xxxxxx.



images/ fichier.png

FIGURE 3.2 – Diagramme de cas d'utilisation "xxxxx"

Description Description Description Description Description Description Description Description Description Description Description Description Description Description Description .



### 3.1.2 yyyyyyyyyyyy

La figure ci dessous décrit le yyyyyyyyyy.

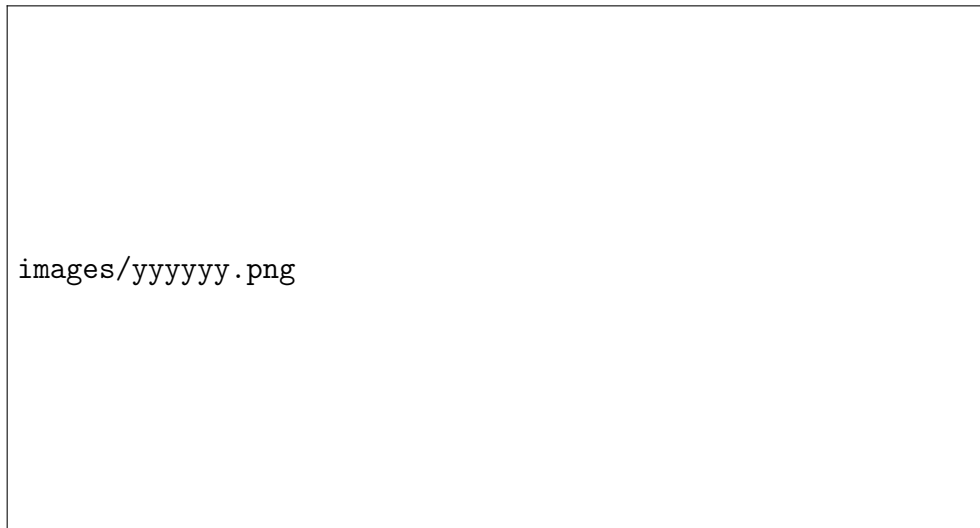


FIGURE 3.3 – Diagramme de cas d'utilisation "yyyyyyyyyy"

Description Description Description Description Description Description Description Description Description Description Description Description Description Description.

### 3.1.3 zzzzzzzzzzzzzzzzz

La figure ci dessous décrit le zzzzzzzzzzz.

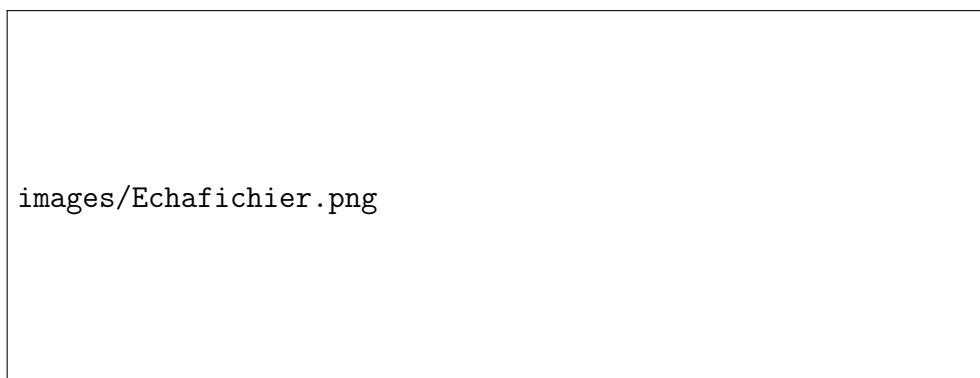


FIGURE 3.4 – Diagramme de cas d'utilisation "yyyyyyyyyy"

Description Description Description Description Description Description Description Description Description Description Description Description Description Description Description.

### 3.2 Diagramme de séquences

Les diagrammes de séquence jouent un rôle crucial dans la modélisation et la conception des systèmes logiciels orientés objet. Ils permettent de représenter visuellement l'ordre des interactions entre les objets et les composants d'un système au fil du temps.

Le passage de diagrammes de cas d'utilisation vers des diagrammes de séquence permet de détailler comment les interactions entre les acteurs et les cas d'utilisation se déroulent au niveau de la séquence d'actions entre les objets du système.

Description Description Description Description Description Description Description Description Description Description Description Description Description Description Description.

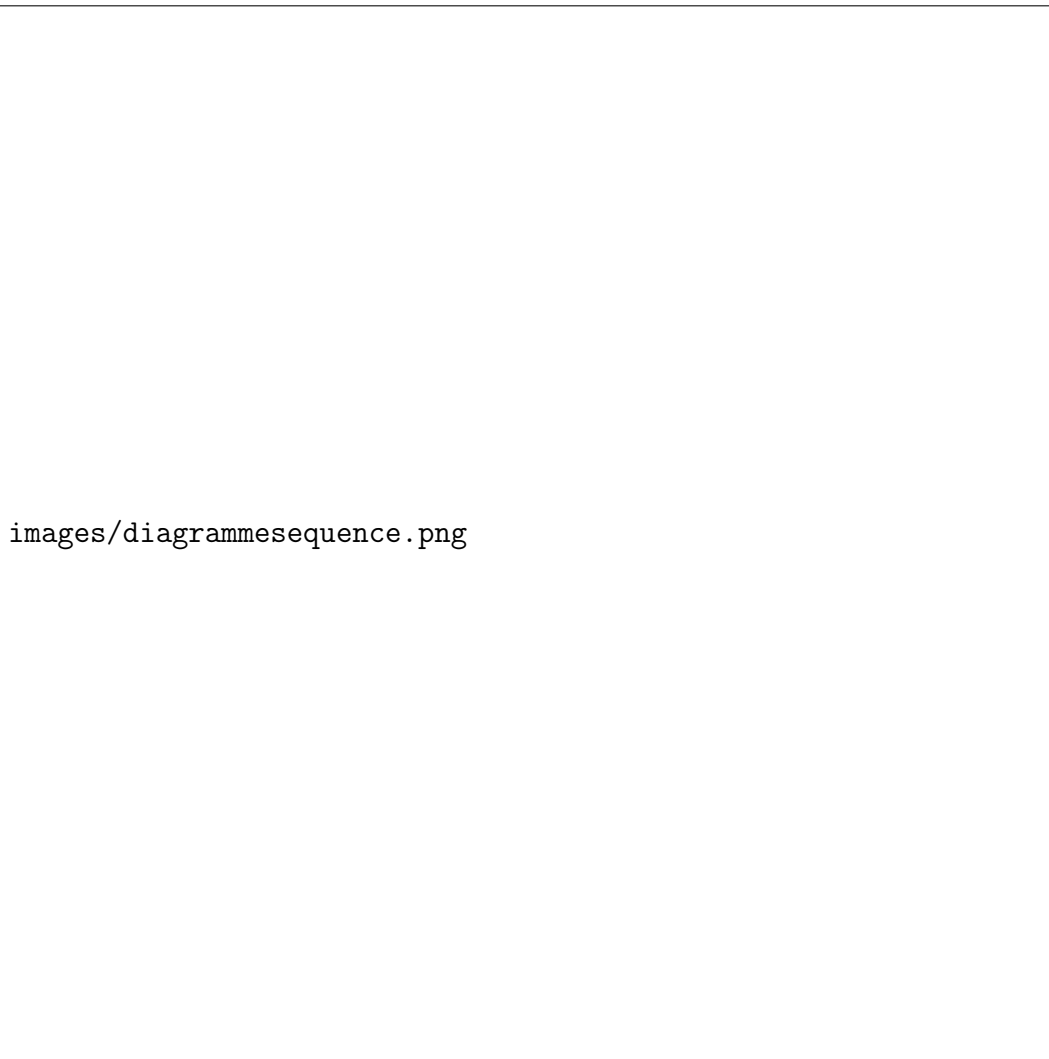


FIGURE 3.5 – Diagramme de séquence "nnnnnnnnnn"

### 3.3 Diagramme des classes de conception

Le diagramme de classes de conception se concentre sur la structure interne des classes. Il vise à montrer comment les classes collaborent pour implémenter les fonctionnalités spécifiques du système. Ce type de diagramme est particulièrement utile lors de la phase de conception détaillée.

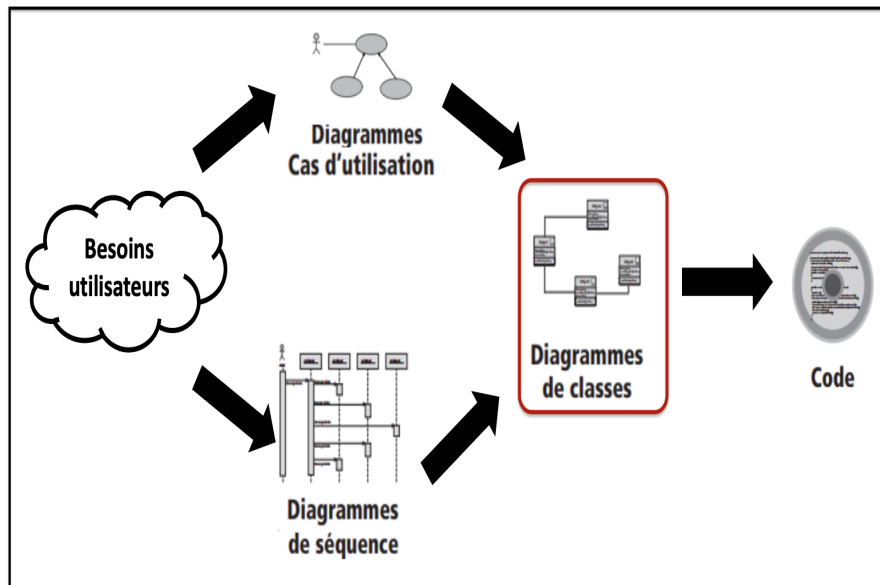


FIGURE 3.6 – Diagramme de classes de conception

## Conclusion

Dans ce chapitre nous avons présenté une conception détaillée à savoir les diagrammes des cas d'utilisation raffinés de xxxxxx, ensuite les diagrammes de séquences et enfin le diagramme de classe. Le prochain chapitre fait l'objet de la partie réalisation du projet.

## Chapitre 4

### Réalisation

## Introduction

Dans ce chapitre, nous plongeons au cœur de notre projet de digitalisation de fichiers à travers l'utilisation d'un scanner. Nous allons découvrir en détail les étapes méthodiques que nous avons suivies, depuis la sélection des technologies jusqu'à la mise en place du projet et le codage.

### 4.1 Environnement de travail

- **Lucidchart :**

Lucidchart est une application Web de création de diagrammes qui permet aux utilisateurs de collaborer visuellement pour dessiner, réviser et partager des graphiques et des diagrammes, et d'améliorer les processus, les systèmes et les structures organisationnelles..



FIGURE 4.1 – Logo Lucidchart

- **Figma :**

C'est un support de présentation et de collaboration. Il permet un travail d'équipe à distance, fluide, centralisé, en toute transparence et instantané. Web-designers, développeurs, clients ou testeurs peuvent intervenir en simultané sur le même fichier[1].

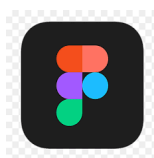


FIGURE 4.2 – Logo Figma

#### 4.1.2.2 Choix des frameworks :

- **Next.js :**

Next.js est un framework de développement Web open-source créé par la société privée Vercel fournissant des applications Web basées sur React avec un rendu côté serveur et une génération de site Web statique.



FIGURE 4.3 – Logo Next.js

- **Node.js :**

Node.js est un environnement de serveur open source multiplateforme qui peut s'exécuter sur Windows, Linux, Unix, macOS, etc. Node.js est un environnement d'exécution JavaScript back-end, s'exécute sur le moteur JavaScript V8 et exécute du code JavaScript en dehors d'un navigateur Web.



FIGURE 4.4 – Logo Node.js

#### Outils de traitement de texte :

- **Overleaf :**

Overleaf est une plateforme en ligne gratuite permettant d'éditer du texte en L<sup>A</sup>-TEX sans aucun téléchargement d'application. En outre, elle offre la possibilité de rédiger des documents de manière collaborative[2].



FIGURE 4.5 – Logo Overleaf

## Base de données et bibliothèques :

- **MongoDB :**

MongoDB est un programme de base de données multiplateforme orienté document disponible en source. Classé comme programme de base de données NoSQL, MongoDB utilise des documents de type JSON avec des schémas facultatifs.



FIGURE 4.6 – Logo MongoDB

- **scanner.js :**

C'est une bibliothèque open-source qui permet d'accéder aux fonctionnalités de numérisation à partir d'un navigateur web en utilisant des périphériques de numérisation compatibles. Avec scanner.js, vous pouvez créer des applications web qui permettent aux utilisateurs de numériser des documents directement depuis leur navigateur, sans avoir besoin d'installer des logiciels tiers.

Cette bibliothèque utilise généralement des technologies telles que WebAssembly pour communiquer avec les scanners et les périphériques de numérisation. Elle offre une interface simple pour déclencher les numérisations, spécifier des paramètres comme la résolution et le format de sortie, et récupérer les données numérisées pour les manipuler dans votre application.

- **Visual Studio Code :**

Il permet de travailler avec différents langages. Il fournit des extensions pour le développement front, back et fullstack. Il supporte le git donc nous pouvons faire tout le travail git à partir de VS studio code[3].



FIGURE 4.7 – Logo Visual Studio Code

## 4.2 Architecture adoptée

Dans le domaine informatique, l'architecture logique décrit les composants abstraits (souvent les services et leurs clients) et les interactions entre eux. Dans ce



contexte nous allons commencer par présenter le modèle MVC[4].

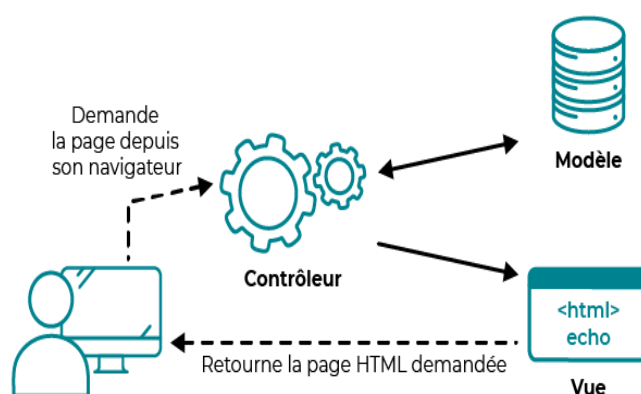


FIGURE 4.8 – Architecture MVC

### Architecture MVC(Model View Controller)

- **Model :**

Le modèle représente la forme des données et la logique métier. Il maintient les données de l'application. Les objets de modèle récupèrent et stockent l'état du modèle dans une base de données. Le modèle est une donnée et une logique métier.

- **View :**

View ou Vue est une interface utilisateur. Afficher les données d'affichage en utilisant le modèle pour l'utilisateur et leur permet également de modifier les données. La vue est une interface utilisateur.

- **Controller :**

Le contrôleur gère la demande de l'utilisateur. En règle générale, l'utilisateur interagit avec la Vue, ce qui déclenche à son tour la demande d'URL appropriée. Cette demande sera gérée par un contrôleur. Le contrôleur rend la vue appropriée avec les données du modèle en réponse. Le contrôleur est un gestionnaire de demandes[4].

### 4.3 Enchaînement des interfaces

Cette partie est principalement dédiée à la présentation de quelques écrans de la solution xxxxx pour comprendre visuellement l'idée du projet

Commençons par présenter l'interface d'accueil :

FIGURE 4.9 – Interface AAAAAA

[illegible]

FIGURE 4.10 – Interface de BB

[illegible]

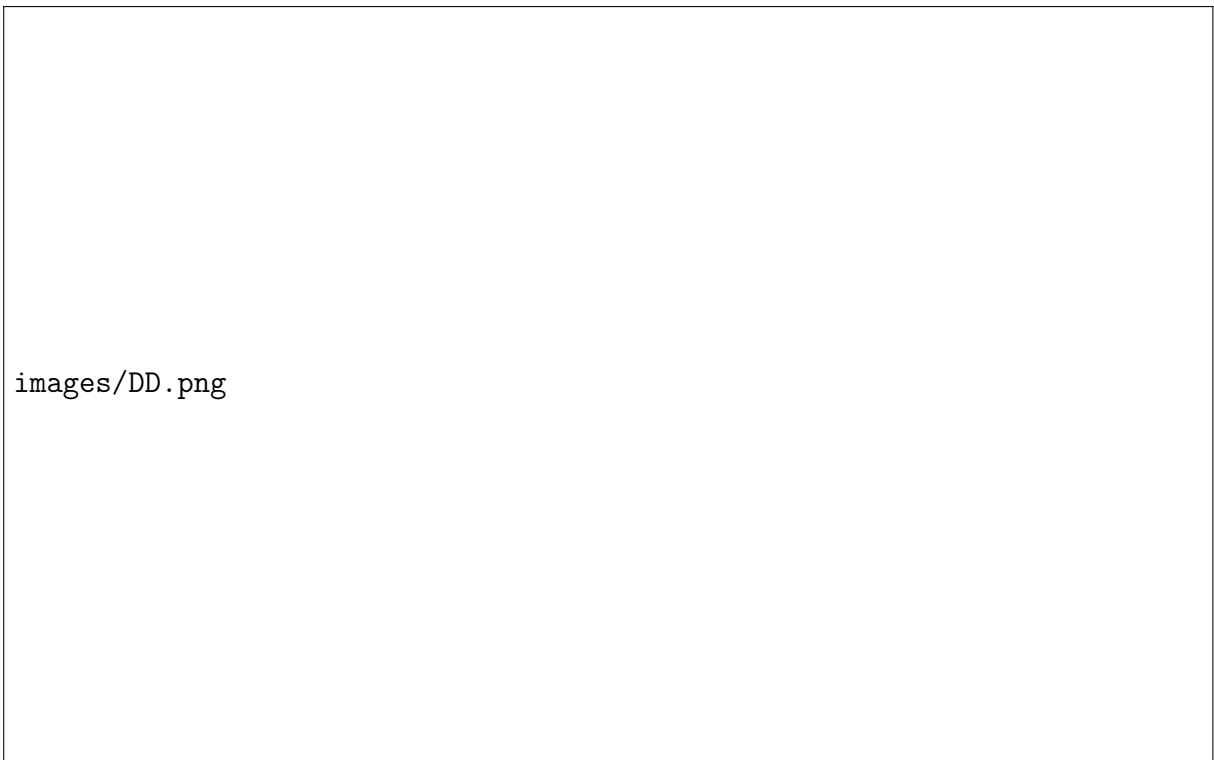


FIGURE 4.11 – DDDDDDDDDDD

Description de l'interface Description de l'interface Description de l'interface  
Description de l'interface Description de l'interface Description de l'interface Des-  
cription de l'interface Description de l'interface comme l'indique la figure suivante :

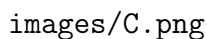


FIGURE 4.12 – Interface CCCCC

Description de l'interface Description de l'interface Description de l'interface  
Description de l'interface Description de l'interface Description de l'interface Des-  
cription de l'interface Description de l'interface comme l'indique la figure suivante :

## Conclusion

En conclusion, le chapitre consacré à la réalisation du projet decrire votre projet decrire votre projet decrire votre projet decrire votre projet. Ce chapitre illustre ainsi la valeur ajoutée de decrire votre projetdecrire votre projetdecrire votre projetdecrire votre projet.

# Conclusion Générale

Ce projet s'inscrit dans le cadre du projet de fin d'étude pour l'obtention de diplôme d'ingénieur en génie logiciel. Son objectif principal est Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte .

Entreprendre un projet technique en utilisant Next.js, Node.js et la bibliothèque Scanner.js présente de nombreux avantages pour un étudiant. Cela offre une opportunité d'apprentissage polyvalente en explorant simultanément diverses compétences. Cette expérience pratique comble le fossé entre les connaissances théoriques et leur application concrète, renforçant ainsi la compréhension de concepts complexes. La portée globale du projet favorise une compréhension holistique du cycle de développement et cela encourage l'exploration créative, incitant l'étudiant à concevoir des solutions inventives et à affiner ses compétences en résolution de problèmes.

Ce rapport détaille de manière exhaustive les étapes du projet, de l'analyse conceptuelle à la conception finale. Il débute par l'exploration théorique, passe par la recherche d'une structure adaptée, et se termine par une planification rigoureuse de l'exécution. En somme, ce rapport met en avant la progression linéaire de l'idée à la réalisation, offrant une vision complète du processus.

Ce projet présente des opportunités d'amélioration significatives. Une direction prometteuse est l'intégration d'un système OCR pour une meilleure collaboration entre l'IA et le processus Some texte Some texte Some texte Some texte. Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte Some texte. En affinant ces aspects, le projet pourrait évoluer vers une solution plus avancée, améliorant l'efficacité dans divers domaines.

# Webographie

- [1] Figma. <http://www.figma.com>, Consulté le 12/03/2024.
- [2] Overleaf. <https://www.overleaf.com/learn>, 2023.
- [3] VSCode. <https://code.visualstudio.com>, Consulté le 12/04/2024.
- [4] Architecture, Consulté le 2023.