



المدرسة العليا  
للتكنولوجيا - اسفي  
ÉCOLE SUPÉRIEURE  
DE TECHNOLOGIE -SAFI

Université Cadi Ayyad  
Ecole Supérieure de Technologie-Safi  
Département : Informatique Filière :  
Génie Informatique

# Rapport de Projet de Fin d'Études

## Conception et Développement d'une Application Web de Diagnostic Médical Automatisé avec IA, OCR et Chat- GPT

**Réalisé par :**

Annouar Hakima.

Ouarradi Halima.

**Encadré par :**

Mr. AGRIMA ABDELLAH

**Assister par :**

Elgharras Hajar

Année Universitaire:2024-2025

## **Dédicaces :**

***À nos chers parents***, qui nous ont offert un magnifique modèle de labeur et de persévérance, aucune expression ne pourra jamais suffire à exprimer nos sentiments envers vous et notre éternelle reconnaissance pour vos sacrifices et votre soutien continu.

***À nos sœurs et nos frères***, qui nous ont toujours encouragés tout au long de la réalisation de ce projet.

***À nos chères professeures***, ce n'est vraiment pas dans nos habitudes d'exprimer notre gratitude, mais pour vous, c'est un peu différent. Merci pour les valeurs nobles, l'éducation et le soutien constants que nous avons reçus de votre part.

***À tous nos amis***, dont la présence et le soutien ont rendu ce parcours encore plus significatif et mémorable.

Merci

## **Remerciement :**

Nous tenons à exprimer notre reconnaissance envers Dieu pour la force et la patience qu'Il nous a accordées pour accomplir ce travail. Un grand merci à nos enseignants et à notre encadreur pour leurs conseils et leur patience tout au long de ce processus. Nos remerciements s'étendent à toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce travail. Nous souhaitons également exprimer notre gratitude envers notre famille, malgré la distance, pour leur soutien constant.

## **Table de matière :**

<b>1</b>	<b>INTRODUCTION :</b>	<b>9</b>
1.1	Définition du projet :	9
1.2	Importance du projet :	9
<b>2</b>	<b>Présentation du projet :</b>	<b>9</b>
2.1	Idée du projet :	9
2.2	Composants du projet :	10
<b>3</b>	<b>Objectifs du Projet :</b>	<b>10</b>
<b>4</b>	<b>PROBLEMATIQUE :</b>	<b>10</b>
4.1	Défis dans le domaine médical :	11
4.2	Solution proposée :	11
<b>5</b>	<b>Impact et Évolutions Futures du Projet :</b>	<b>12</b>
5.1	Impact attendu :	12
5.2	Perspectives futures :	12
<b>6</b>	<b>Conclusion :</b>	<b>13</b>
<b>1</b>	<b>Introduction :</b>	<b>15</b>
<b>2</b>	<b>Intelligence Artificielle en MEdecine :</b>	<b>15</b>
<b>3</b>	<b>Reconnaissance Optique de Caractères (OCR) en Médecine :</b>	<b>16</b>
<b>4</b>	<b>ChatGPT et son Application en MEDECINE :</b>	<b>18</b>
<b>5</b>	<b>Conclusion :</b>	<b>20</b>
<b>1</b>	<b>INTRODUCTION :</b>	<b>23</b>
<b>2</b>	<b>EQUIPE DU PROJET :</b>	<b>23</b>
<b>3</b>	<b>Planification du projet :</b>	<b>24</b>
<b>4</b>	<b>Diagramme de gantt :</b>	<b>24</b>
<b>5</b>	<b>suivi du projet :</b>	<b>24</b>
5.1	Modeles de Developpement :	24
5.2	Modeles Agiles :	25
5.3	Application de la méthode Scrum :	25

5.4	Présentation de la méthode Scrum :	26
5.5	Rôles et Artefacts Scrum dans notre projet :	26
5.6	Les processus Scrum :	27
6	Logicieles UTILISES :	28
7	Conclusion :	28
1	Introduction :	30
2	RECUEIL DES BESOINS FONCTIONNELS ET NON FONCTIONNELS :	30
2.1	Besoins fonctionnels :	30
2.2	Besoins non fonctionnels :	30
3	Conception de base de donnee :	31
3.1	Choix du langage de conception de base de donnée(UML) :	31
3.2	Notions UML :	31
4	Conception de l'application :	32
4.1	Introduction au Diagramme de cas d'utilisation :	32
4.2	Introduction au Diagramme de classe :	33
4.3	Introduction au Diagramme de Séquence :	34
5	Conclusion :	36
1	Introduction :	38
2	Environnements de travail et choix techniqueS :	38
2.1	Framework et Environnements de travail :	38
2.2	Langages de programmation utilisés :	41
3	Principales interfaces graphiques :	43
3.1	Interface d'accueil :	43

---

# Liste de figures

---

## **Introduction générale :**

Dans ce rapport, nous présentons notre application de diagnostic médical automatisé, conçue pour répondre aux besoins croissants du secteur de la santé en matière d'analyse et d'interprétation des documents médicaux. Avec l'essor des technologies d'intelligence artificielle et de reconnaissance optique de caractères (OCR), il est devenu essentiel de fournir aux professionnels de santé un outil performant pour extraire et analyser les informations médicales. Notre application vise principalement à simplifier l'exploitation des documents médicaux en permettant aux utilisateurs de télécharger des fichiers, d'en extraire le texte via OCR et d'interagir avec ces informations grâce à un modèle d'IA avancé. En offrant une solution moderne et intuitive, nous aidons les utilisateurs à mieux comprendre et traiter les données médicales, améliorant ainsi la prise de décision clinique et la gestion des diagnostics.

Pour planifier et réaliser cette application, nous avons adopté la méthodologie Agile. Cette approche itérative et flexible nous permet d'adapter le développement en fonction des retours des utilisateurs et d'améliorer en continu les fonctionnalités de l'application. En utilisant Agile, nous avons découpé le projet en sprints successifs, chacun se concentrant sur une partie spécifique du développement, de la définition des besoins à la mise en production. Cette approche favorise une meilleure réactivité face aux exigences changeantes et garantit une application plus adaptée aux besoins réels des utilisateurs.

Pour le développement de notre application, nous avons choisi d'utiliser les langages HTML, CSS, JavaScript, PHP et Bootstrap. Cette combinaison technologique nous permet de garantir une application web robuste, accessible et optimisée pour une utilisation fluide. De plus, l'intégration de l'IA et de l'OCR nous permet de fournir une interface intelligente et interactive, garantissant une expérience utilisateur optimale pour tous les utilisateurs, quel que soit leur niveau de compétence technologique.

---

# Section 1 : Contexte Global du Projet

---



# 1 INTRODUCTION :

## 1.1 Définition du projet :

Les technologies d'intelligence artificielle (IA) sont devenues une partie intégrante de notre quotidien, en particulier dans le domaine médical, où elles peuvent jouer un rôle clé dans la facilitation du diagnostic médical et l'analyse des données rapidement et avec précision. Ce projet a pour objectif de développer une plateforme de diagnostic médical automatisé, basée sur l'IA, la reconnaissance optique de caractères (OCR) et le modèle Chat-GPT, afin d'améliorer la qualité des soins de santé.

## 1.2 Importance du projet :

Ce projet permet aux médecins et aux professionnels de la santé d'extraire rapidement les informations médicales importantes des documents et de mieux les comprendre, contribuant ainsi à accélérer le processus de diagnostic et à réduire les erreurs humaines.

# 2 PRESENTATION DU PROJET :

## 2.1 Idée du projet :

L'idée du projet est de développer une plateforme capable de lire des documents médicaux numérisés (tels que des ordonnances, des analyses, des rapports médicaux, etc.) en utilisant la technologie OCR, puis d'analyser et de comprendre ces textes grâce à un modèle d'intelligence artificielle avancé comme Chat-GPT, afin de fournir des explications ou d'aider au processus de diagnostic.

## 2.2 Composants du projet :

**OCR (Reconnaissance Optique de Caractères)** : pour extraire les textes des documents médicaux.

**Modèle Chat-GPT** : pour analyser et comprendre les textes médicaux.

**Interface utilisateur simple et interactive** : permettant à l'utilisateur de télécharger des documents et d'interagir avec les résultats.

**Base de données** : pour stocker les documents extraits et les historiques des utilisateurs

## 3 OBJECTIFS DU PROJET :

L'objectif principal de ce projet est de développer une plateforme intelligente de **diagnostic médical automatisé**, en intégrant les technologies de **l'intelligence artificielle**, de la **reconnaissance optique de caractères (OCR)** et de **ChatGPT**. Cette plateforme vise à :

- **Faciliter l'analyse de documents médicaux** (ordonnances, bilans, comptes rendus, etc.) en extrayant automatiquement les informations pertinentes grâce à l'OCR.
- **Offrir un assistant intelligent** capable de comprendre et de répondre aux questions des utilisateurs en se basant sur le contenu médical extrait.
- **Améliorer l'accessibilité aux informations médicales** pour les patients et les professionnels de santé.
- **Automatiser et simplifier les démarches de compréhension des diagnostics** médicaux à partir de fichiers médicaux numériques.
- **Conserver un historique** des documents traités et des interactions, pour un meilleur suivi.

## 4 PROBLEMATIQUE :

De nombreux patients rencontrent des difficultés à comprendre les documents médicaux en raison de l'utilisation de termes complexes, ou du manque de temps

des médecins. De plus, les médecins passent beaucoup de temps à expliquer les diagnostics et à traiter les documents manuellement.

C'est pourquoi nous soulevons les problématiques suivantes :

- Comment automatiser la lecture et la compréhension des documents médicaux ?
- Comment permettre à l'utilisateur (patient ou médecin) d'interagir de manière simple avec le contenu des documents ?
- Est-il possible de concevoir un outil intelligent, facile à utiliser, qui respecte la confidentialité des données médicales et aide au diagnostic ?

#### 4.1 Défis dans le domaine médical :

Le secteur médical fait face à plusieurs problèmes, notamment :

- **Le volume élevé de documents médicaux** à examiner et analyser.
- **Le manque de temps et la surcharge de tâches** empêchant les médecins d'étudier chaque cas en détail.
- **Les erreurs humaines** pouvant avoir des conséquences graves sur la santé des patients.
- **Le manque de ressources humaines**, en particulier dans les zones rurales.

#### 4.2 Solution proposée :

Le projet propose une solution innovante qui consiste en :

- L'automatisation de la lecture et de la compréhension des documents médicaux grâce à l'OCR et à Chat-GPT.
- La réduction des efforts et du temps nécessaire pour analyser les informations médicales.
- La fourniture d'explications instantanées aux utilisateurs, qu'ils soient patients ou professionnels de santé.

## 5 IMPACT ET ÉVOLUTIONS FUTURES DU PROJET :

### 5.1 Impact attendu :

- À travers ce projet, nous aspirons à :
- Accélérer le traitement des documents médicaux.
- Réduire les erreurs dues à une mauvaise compréhension ou à une saisie manuelle.
- Aider les patients à mieux comprendre leur état de santé de manière simplifiée.
- Alléger la charge des médecins, notamment dans les zones manquant de ressources médicales..

### 5.2 Perspectives futures :

À l'avenir, le projet pourrait être développé pour inclure :

- **L'analyse des images médicales** (telles que les radiographies, les IRM, etc.).
- **Le support multilingue** pour étendre l'utilisation de la plateforme dans diverses régions géographiques.
- **L'extraction de texte à partir de l'écriture manuscrite** pour permettre de lire et comprendre les documents rédigés à la main, facilitant ainsi l'analyse des prescriptions et rapports écrits manuellement.

## 6 CONCLUSION :

Ce projet vise à améliorer l'efficacité et la précision du diagnostic médical en automatisant la lecture et la compréhension des documents médicaux. Grâce à l'intégration de l'OCR et de Chat-GPT, il permet une meilleure gestion du temps et une réduction des erreurs humaines. Les perspectives futures du projet, notamment l'analyse des images médicales et l'intégration de l'écriture manuscrite, promettent de renforcer son impact, offrant ainsi des solutions novatrices pour les professionnels de la santé et les patients.

---

## **Section 2 : l'Etat d'art**

---

## 1 INTRODUCTION :

L'intégration de l'intelligence artificielle (IA) dans le domaine médical transforme la manière dont les informations sont traitées et analysées. Notre plateforme web de diagnostic médical automatisé exploite des technologies avancées telles que la reconnaissance optique de caractères (OCR) et des modèles de traitement du langage naturel (NLP) pour améliorer l'efficacité et la précision des processus médicaux. Cette section explore les outils principaux de notre solution, **Tesseract OCR** et **Mistral Small 3.1**, en détaillant leurs avantages, défis et considérations éthiques.

## 2 INTELLIGENCE ARTIFICIELLE EN MEDECINE :

L'IA transforme le domaine médical en introduisant des technologies capables d'améliorer la précision des diagnostics, d'optimiser les traitements et de personnaliser les soins aux patients. Elle englobe diverses techniques, notamment le traitement du langage naturel, l'apprentissage automatique et l'analyse d'images, pour interpréter des données complexes et assister les professionnels de santé.

### 2.1 Applications de l'IA en Médecine

- **Automatisation des Processus Administratifs** : Les hôpitaux adoptent des systèmes basés sur l'IA pour automatiser des tâches administratives répétitives, telles que la planification des rendez-vous, la gestion des dossiers médicaux et la facturation, permettant ainsi au personnel de se concentrer davantage sur les soins aux patients.
- **Reconnaissance Optique de Caractères (OCR)** : L'OCR, combinée à l'IA, convertit les documents médicaux papier en formats numériques exploitables, facilitant l'accès, la recherche et l'analyse des informations patient.
- **Assistants Virtuels et Chatbots** : Des agents conversationnels, tels que ChatGPT, fournissent des informations médicales aux patients, répondent à leurs questions et les guident vers des ressources appropriées, améliorant ainsi l'engagement et l'éducation des patients.
- **Médecine Personnalisée** : L'IA analyse des données génétiques et cliniques pour adapter les traitements aux caractéristiques individuelles des patients, augmentant l'efficacité thérapeutique et réduisant les effets secondaires.

- **Découverte de Médicaments** : Les algorithmes d'IA accélèrent le processus de découverte de nouveaux médicaments en prédisant l'efficacité de composés chimiques et en identifiant des cibles thérapeutiques potentielles.

## 2.2 Avantages de l'IA en Médecine

- **Précision Améliorée** : Les systèmes d'IA peuvent surpasser les humains dans certaines tâches diagnostiques en identifiant des motifs invisibles à l'œil nu.
- **Efficacité Accrue** : L'automatisation des tâches administratives et cliniques libère du temps pour les professionnels de santé, leur permettant de se concentrer sur les soins directs aux patients.
- **Personnalisation des Soins** : L'analyse de grandes quantités de données permet de concevoir des plans de traitement adaptés aux besoins spécifiques de chaque patient.

## 2.3 Défis et Considérations Éthiques

- **Qualité des Données** : Les algorithmes d'IA dépendent de données de haute qualité. Des données incomplètes ou biaisées peuvent entraîner des conclusions erronées.
- **Confidentialité et Sécurité** : L'utilisation de l'IA nécessite l'accès à des données sensibles des patients, soulevant des préoccupations concernant la confidentialité et la protection des informations personnelles.
- **Responsabilité Médicale** : Déterminer la responsabilité en cas d'erreur d'un système d'IA est complexe et soulève des questions juridiques et éthiques.
- **Acceptation par les Professionnels de Santé** : L'adoption de l'IA nécessite une formation adéquate et peut rencontrer une résistance au changement de la part du personnel médical.

# 3 RECONNAISSANCE OPTIQUE DE CARACTERES (OCR) EN MEDECINE :

La reconnaissance optique de caractères (OCR) est une technologie qui permet de convertir des documents physiques ou numériques en textes exploitables par des systèmes informatiques. En médecine, l'OCR est utilisé pour :

- La numérisation des dossiers médicaux manuscrits.
- L'automatisation de la saisie des ordonnances et des rapports médicaux.
- L'extraction d'informations à partir de documents scannés.



## 3.1 Tesseract OCR



Dans notre plateforme, nous utilisons **Tesseract OCR**, un moteur open-source initialement développé par Hewlett-Packard et actuellement maintenu par Google. Tesseract est reconnu pour sa précision et sa capacité à reconnaître des textes imprimés et manuscrits dans plus de 100 langues. Il peut être combiné avec des bibliothèques de prétraitement d'image comme OpenCV pour améliorer la qualité de la reconnaissance.

### 3.1.1 Caractéristiques Techniques de Tesseract OCR :

- **Open Source** : Disponible sous licence Apache 2.0, Tesseract peut être utilisé et modifié librement.
- **Support Multilingue** : Prise en charge de plus de 100 langues, facilitant son utilisation dans divers contextes internationaux.
- **Formats d'Image Compatibles** : Capable de traiter des images aux formats tels que TIFF, JPEG et PNG.
- **Reconnaissance de la Mise en Page** : Capacité à détecter la disposition des paragraphes, colonnes et autres éléments de mise en forme dans les documents.

### 3.1.2 Avantages de Tesseract OCR

- **Open-source et gratuit** : Permet une intégration sans coût de licence, favorisant l'adaptabilité et la personnalisation selon les besoins spécifiques du projet.
- **Polyvalence linguistique** : Supporte une large gamme de langues, ce qui est crucial dans un environnement médical multiculturel.
- **Communauté active** : Bénéficie de mises à jour régulières et d'une amélioration continue grâce à une communauté de développeurs engagés.

### 3.1.3 Défis Associés à Tesseract OCR

- **Qualité des documents source** : La précision de la reconnaissance dépend fortement de la clarté et de la qualité des documents scannés. Les documents de mauvaise qualité peuvent entraîner des erreurs de reconnaissance.

- **Complexité des mises en page** : Tesseract peut rencontrer des difficultés avec des documents comportant des mises en page complexes, des tableaux ou des colonnes multiples.

### 3.1.4 Considérations Éthiques

L'utilisation de Tesseract OCR dans le domaine médical nécessite une attention particulière à la confidentialité et à la sécurité des données patient. Il est impératif de garantir que les informations extraites soient protégées conformément aux réglementations en vigueur, telles que le RGPD.

## 4 CHATGPT ET SON APPLICATION EN MEDECINE :

**ChatGPT**, développé par OpenAI, est un modèle de traitement du langage naturel basé sur des architectures de deep learning. Il est utilisé en médecine pour :

- **Assistance aux patients** : fournir des réponses aux questions médicales générales, améliorant ainsi l'engagement et l'éducation des patients.
- **Soutien aux professionnels de santé** : aide à la rédaction et à la classification des documents médicaux.
- **Interprétation des résultats d'analyses** : fournir des explications sur les résultats médicaux pour faciliter la compréhension.

## 4.1 Mistral Small 3.1 pour le Traitement du Langage Naturel



Dans notre implémentation, nous utilisons **Mistral Small 3.1**, une API de traitement du langage développée par Mistral AI. Ce modèle est reconnu pour sa rapidité et son efficacité dans l'analyse et la génération de texte. En exploitant Mistral Small 3.1, notre plateforme peut traiter les demandes des utilisateurs et générer des réponses adaptées aux besoins médicaux tout en respectant la confidentialité des données.

### 4.1.1 Présentation de Mistral Small 3.1 :

Mistral Small 3.1 est un modèle **IA open-source** de 24 milliards de paramètres, offrant des performances élevées tout en étant suffisamment léger pour fonctionner sur des infrastructures matérielles limitées.

### 4.1.2 Avantages de Mistral Small 3.1

- **Efficacité et légèreté** : Fonctionne efficacement sur des configurations matérielles modestes, facilitant son déploiement dans diverses applications.
- **Polyvalence** : Capable de gérer une variété de tâches, y compris l'assistance conversationnelle et la compréhension d'images.
- **Open-source** : Offre une transparence et une flexibilité accrues pour les développeurs souhaitant adapter le modèle à des besoins spécifiques.

### 4.1.3 Défis Associés à Mistral Small 3.1

- **Biais et précision** : Comme tout modèle NLP, Mistral Small 3.1 peut hériter de biais présents dans les données d'entraînement, ce qui nécessite une vigilance particulière lors de son utilisation dans des contextes sensibles comme la médecine.

- **Personnalisation** : Bien que polyvalent, le modèle peut nécessiter un ajustement ou un fine-tuning pour des applications très spécifiques, impliquant des ressources supplémentaires.

#### 4.1.4 Considérations Éthiques

L'intégration de Mistral Small 3.1 dans une plateforme médicale soulève des questions éthiques concernant la fiabilité des informations générées et la responsabilité en cas d'erreurs. Il est essentiel d'établir des protocoles clairs pour valider les réponses fournies par le modèle et assurer une supervision humaine adéquate.

## 5 CONCLUSION :

L'intégration de technologies avancées telles que l'intelligence artificielle, l'OCR (notamment Tesseract) et des modèles de traitement du langage comme ChatGPT ou Mistral Small 3.1 dans une plateforme de diagnostic médical représente une avancée prometteuse pour renforcer la précision des diagnostics et optimiser les processus médicaux. En adoptant une approche pragmatique basée sur l'utilisation d'outils spécialisés sans recourir à des techniques complexes d'apprentissage automatique, il est possible de déployer rapidement des solutions efficaces et accessibles. Cependant, cette transformation numérique ne peut se faire sans une réflexion approfondie sur les défis techniques, les implications éthiques, et surtout la protection des données sensibles des patients. Il est donc essentiel de concilier innovation technologique et responsabilité, afin de garantir la fiabilité, la sécurité et l'acceptabilité de ces nouvelles solutions dans le domaine médical.



---

## **Section 3 : Organisation du Projet**

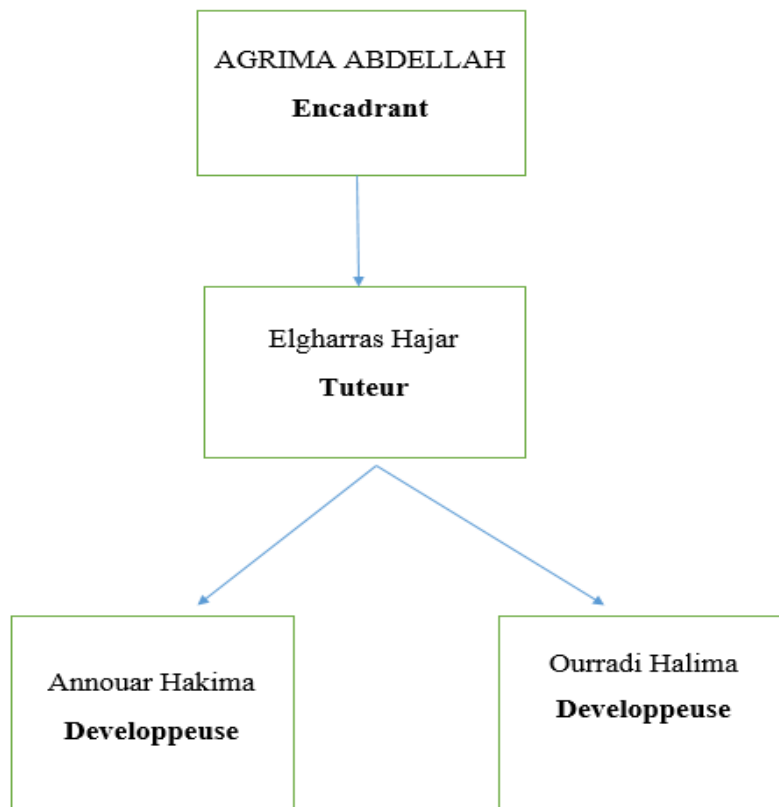
---

## 1 INTRODUCTION :

L'organisation du projet de développement d'une application web de gestion de projet. Il décrit les rôles des différents intervenants, les processus de développement et les outils utilisés.

## 2 EQUIPE DU PROJET :

|



### 3 PLANIFICATION DU PROJET :

La planification constitue l'une des étapes fondamentales des projets préliminaires. Elle englobe la définition et la structuration des différentes tâches à réaliser, ainsi que l'estimation de leur durée. Par ailleurs, la planification joue un rôle essentiel dans l'établissement de la portée globale de l'effort nécessaire, la clarification et le raffinement des objectifs, ainsi que l'élaboration d'une stratégie d'action pour les atteindre.

### 4 DIAGRAMME DE GANTT :

Pour planifier et suivre l'avancement de notre projet, nous avons opté pour l'utilisation du diagramme de Gantt. Ce diagramme offre une méthode efficace pour planifier le projet de manière optimale et simplifie le suivi de son avancement. Il permet également une visualisation claire et rapide de la séquence temporelle des différentes tâches durant la réalisation du projet. La figure ci-dessous présente le diagramme de Gantt détaillant les différentes étapes à effectuer 17 Table des figures pour mener à bien notre projet.

Ce diagramme permet donc de visualiser d'un seul coup :

- Les différentes tâches à envisager.
- La date de début et la date de fin de chaque tâche.
- La durée escomptée de chaque tâche.
- Le chevauchement éventuel des tâches, et la durée de ce chevauchement.
- La date de début et la date de fin du projet dans son ensemble.

### 5 SUIVI DU PROJET :

#### 5.1 Modeles de Developpement :

Il existe deux paradigmes de développement logiciel : les modèles traditionnels et les modèles agiles. Les modèles traditionnels, tels que le modèle en cascade ou le modèle en V, suivent une approche séquentielle et linéaire du développement, tandis que les modèles agiles, tels que Scrum, RAD (Rapid



Application Développement) ou le modèle en spirale, privilégient la flexibilité, l'adaptabilité et la livraison itérative de fonctionnalités. Le choix du modèle de développement est crucial pour le succès du projet, c'est pourquoi nous avons opté pour la méthodologie agile, qui correspond le mieux à nos besoins et à notre environnement de projet.

## 5.2 Modeles Agiles :

Les modèles agiles sont des méthodologies de développement logiciel qui mettent l'accent sur la flexibilité, l'adaptabilité et la livraison itérative de produits de haute qualité. Ils favorisent la collaboration entre les membres de l'équipe et les parties prenantes du projet, ainsi que la réponse aux changements de priorités et de besoins tout au long du cycle de vie du projet. Les méthodes agiles les plus populaires en usage aujourd'hui sont :

- Extreme Programming (XP)
- Scrum
- La méthode RAD (Rapid Application Développement) Nous allons adopter le modèle Agile Scrum, qui permet de structurer des projets de développement plus complexes.

## 5.3 Application de la méthode Scrum :

Pour la gestion du projet, nous avons choisi d'adopter la méthode **Scrum**, une approche agile reconnue pour sa flexibilité et son efficacité dans la gestion de projets complexes et innovants. La méthode Scrum repose sur des itérations appelées **sprints** et favorise la collaboration entre les membres de l'équipe et les parties prenantes tout au long du développement.

## 5.4 Présentation de la méthode Scrum :

La méthode Scrum est une approche itérative et incrémentale permettant de gérer les projets en cycles courts appelés **sprints**. Chaque sprint dure généralement de deux à quatre semaines et vise à livrer un produit fonctionnel ou une fonctionnalité complète. L'un des principaux objectifs est de fournir de la valeur ajoutée de manière continue et d'adapter les processus en fonction des retours obtenus à la fin de chaque sprint.

## 5.5 Rôles et Artefacts Scrum dans notre projet :

Dans le cadre de notre projet, nous avons identifié les rôles et artefacts suivants :

- **Product Owner** : Responsable de la définition et de la priorisation des fonctionnalités. Ce rôle a été attribué au responsable du projet, qui a collaboré étroitement avec les parties prenantes pour déterminer les exigences du système de diagnostic médical.
- **Scrum Master** : En charge de faciliter l'application de Scrum, en assurant que l'équipe suit bien les principes et pratiques Scrum, tout en enlevant les obstacles qui pourraient bloquer l'avancement du travail.
- **Équipe de développement** : Composée de développeurs et d'experts techniques, l'équipe a travaillé sur l'implémentation des fonctionnalités, notamment l'intégration de l'OCR et du module Chat-GPT.

## 5.6 Les processus Scrum :

La méthode Scrum consiste à enchaîner de façon itérative des timeboxes (blocs de temps) qui vont produire des artefacts (des outils, du logiciel ou des documents qui vont contribuer à orienter et faire avancer le projet). La figure ci-dessous représente le processus Scrum que nous avons suivi pour mettre en œuvre notre projet :



- **User story** : Description d'une fonctionnalité du point de vue de l'utilisateur, détaillant ce qu'il veut accomplir.
- **Product backlog** : Liste priorisée de toutes les fonctionnalités à développer dans un projet.
- **Sprint backlog** : Liste des tâches à réaliser pendant un sprint, tirées du product backlog.
- **Planification de sprint** : Réunion où l'équipe sélectionne les tâches à réaliser lors du prochain sprint
- **Sprint** : Période de temps fixe pendant laquelle l'équipe développe les fonctionnalités sélectionnées.
- **Mêlée quotidienne** : Réunion courte quotidienne où l'équipe synchronise son travail et identifie les obstacles.
- **Revue de sprint** : Réunion à la fin du sprint où l'équipe présente les fonctionnalités développées au Product Owner et aux parties prenantes.
- **Rétrospective de sprint** : Réunion à la fin du sprint où l'équipe examine son fonctionnement et identifie des améliorations à apporter.

## 6 LOGICIELES UTILISES :

**GitHub :**



GitHub est une plateforme de développement collaboratif et de gestion de code source utilisant Git. Elle permet aux développeurs de stocker, de gérer et de suivre les modifications du code source de leurs projets, tout en facilitant la collaboration et le partage de code avec d'autres développeurs

## 7 CONCLUSION :

L'organisation du projet a été définie en tenant compte des compétences des membres de l'équipe et des contraintes du projet. La méthode Scrum a permis une gestion efficace du projet et une livraison rapide de résultats. Les outils utilisés ont facilité la collaboration et la communication entre les membres de l'équipe

---

## **SECTION 4 : ANALYSE ET CONCETION DU PROJET**

---

## 1 INTRODUCTION :

Dans cette section, nous abordons l'analyse et la conception du projet, en mettant l'accent sur les éléments clés nécessaires à la réalisation du système. Cette étape est cruciale pour assurer une compréhension commune de l'architecture, des exigences et des composants du système avant de passer à la phase de développement.

## 2 RECUEIL DES BESOINS FONCTIONNELS ET NON FONCTIONNELS :

Dans cette section, nous détaillons les besoins du projet, tant du point de vue fonctionnel que non fonctionnel. Ces besoins sont cruciaux pour le bon déroulement du projet et doivent être clairement définis avant toute phase de développement.

### 2.1 Besoins fonctionnels :

Les besoins fonctionnels du projet sont les fonctionnalités que le système doit impérativement remplir pour répondre aux attentes des utilisateurs. Cela comprend :

- **Extraction des textes** : Le système doit pouvoir extraire des textes de différents types de documents médicaux numérisés (images, PDF, etc.) à l'aide de la technologie OCR.
- **Analyse des textes** : Le système doit être capable d'analyser les textes extraits et de fournir des explications ou des résumés en utilisant un modèle comme ChatGPT.
- **Interaction avec l'utilisateur** : L'interface utilisateur doit permettre de télécharger des documents, d'interagir avec les résultats et de poser des questions sur les documents médicaux traités.
- **Stockage et gestion des données** : Le système doit être capable de stocker les documents et les historiques des interactions des utilisateurs dans une base de données sécurisée

### 2.2 Besoins non fonctionnels :

Les besoins non fonctionnels déterminent les caractéristiques du système qui ne sont pas directement liées aux fonctionnalités, mais qui sont néanmoins essentielles pour sa réussite :

- **Performance** : Le système doit être capable de traiter rapidement de grands volumes de documents médicaux sans latence importante.

- **Sécurité** : Le système doit garantir la confidentialité et la sécurité des données médicales des utilisateurs, notamment par des mécanismes de cryptage et de protection des accès.
- **Scalabilité** : Le système doit pouvoir évoluer pour gérer un nombre croissant d'utilisateurs et de documents à mesure que le projet se développe.
- **Facilité d'utilisation** : L'interface utilisateur doit être simple et intuitive, même pour les utilisateurs sans compétences techniques.

### 3 CONCEPTION DE BASE DE DONNEE :

#### 3.1 Choix du langage de conception de base de donnée(UML) :

Le langage UML (Unified Modeling Language) est choisi pour la conception de la base de données car il permet de représenter graphiquement les entités et leurs relations. UML est largement utilisé dans le domaine de la modélisation orientée objet, ce qui est pertinent pour notre projet. Il offre une approche structurée et standardisée pour la représentation des objets et des interactions entre eux, permettant ainsi une meilleure compréhension et communication entre les membres de l'équipe de développement.

#### 3.2 Notions UML :



L'UML comprend plusieurs diagrammes qui permettent de modéliser différents aspects d'un système. Parmi ceux-ci, les diagrammes de classes, de cas d'utilisation

et de séquence sont particulièrement utiles pour la conception d'une base de données relationnelle et l'interaction avec les utilisateurs.

## 4 CONCEPTION DE L'APPLICATION

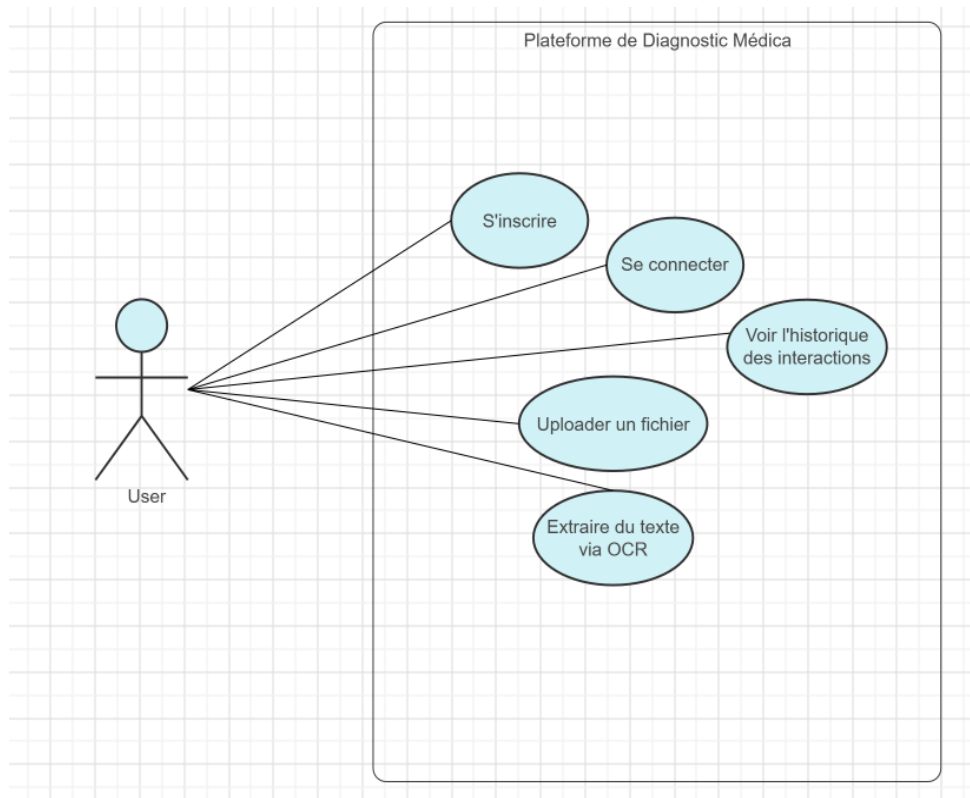
### 4.1 Introduction au Diagramme de cas d'utilisation :

Ce diagramme est destiné à représenter les besoins des utilisateurs par rapport au système. Il constitue un de diagrammes le plus structurants dans l'analyse d'un système.

Acteur : représente un rôle joué par une personne qui interagit directement avec le système étudié.

Cas d'utilisation (use case) : représente un ensemble des séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier. L'utilisation d'un diagramme de cas d'utilisation s'avère indispensables pour décrire les besoins fonctionnels. Ces diagrammes permettent de décrire l'interaction entre l'acteur et le système. C'est l'image d'une fonctionnalité de système déclenchée en réponse à la stimulation d'un acteur externe





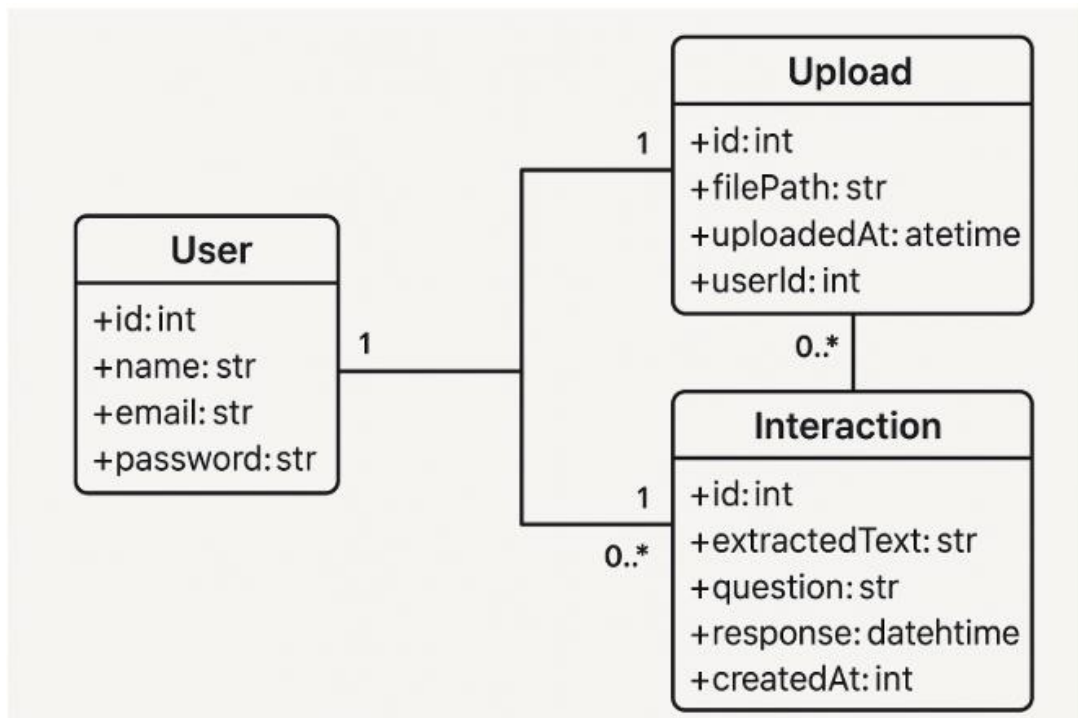
## 4.2 Introduction au Diagramme de classe :

Le diagramme de classe est un type de diagramme structurel utilisé dans le cadre de la modélisation orientée objet. Il représente les classes du système, ainsi que leurs attributs, méthodes, et les relations entre elles.

Le diagramme de classe est essentiel pour comprendre la structure du système, les données qu'il manipule, et la façon dont ces données sont traitées et partagées entre les différentes parties du système.

Ce diagramme permet de visualiser l'architecture interne du système et de planifier l'implémentation en fonction des classes et de leurs interactions.

**Dans notre projet**, un diagramme de classe pourrait inclure des entités comme **User**, **Upload**, et **Interaction**, représentant les principales structures de données et leur relation dans le système.



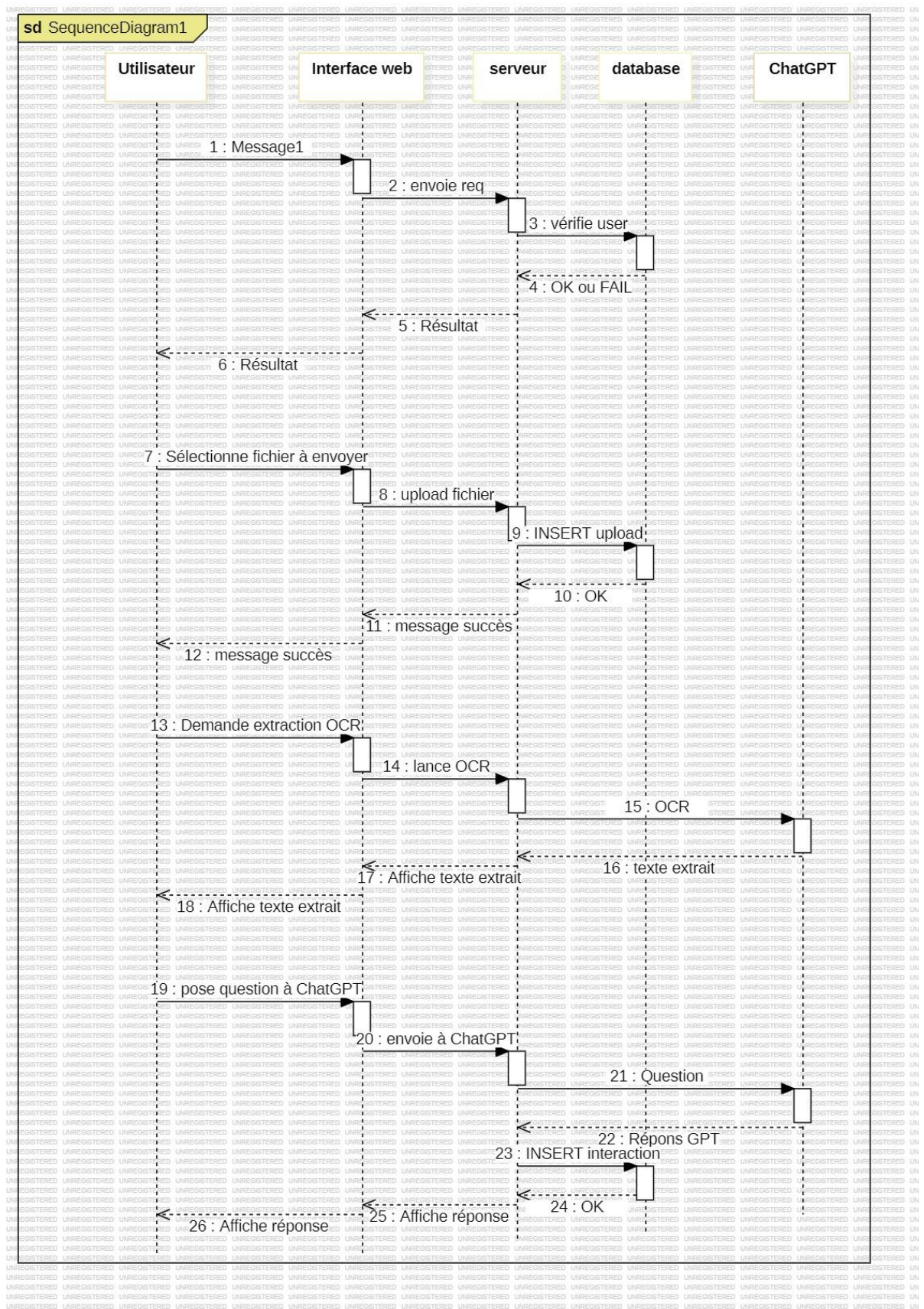
### 4.3 Introduction au Diagramme de Séquence :

Le diagramme de séquence est un type de diagramme comportemental qui montre comment les objets d'un système interagissent entre eux au fil du temps.

Ce diagramme illustre l'ordre des messages échangés entre les différents objets pour accomplir une tâche ou un processus particulier. Chaque ligne verticale représente un objet, tandis que les messages sont représentés par des flèches horizontales entre ces objets.

Le diagramme de séquence est utile pour comprendre les processus détaillés et les échanges entre les composants du système dans un flux temporel.

**Par exemple**, dans notre projet, un diagramme de séquence pourrait montrer les étapes d'une interaction utilisateur : de l'inscription à l'upload d'un fichier, puis à l'extraction du texte via OCR et l'interaction avec ChatGPT.



## 5 CONCLUSION :

Cette section a permis de définir les éléments essentiels pour la réalisation de notre projet, en présentant l'analyse de la conception du système, les diagrammes de modélisation, et en identifiant les besoins fonctionnels et non fonctionnels du projet. Ces éléments servent de base solide pour le développement du système

---

## **SECTION 5 : REALISATION**

---

## 1 INTRODUCTION :

Après avoir mené à bien l'étude et la conception de notre application, nous sommes passés à l'étape cruciale de la réalisation. Cette phase consiste à transformer les modèles conceptuels et les spécifications fonctionnelles en une application concrète, opérationnelle et répondant aux besoins identifiés. Elle présente également les résultats du travail effectué durant ce projet de fin d'études. Nous y détaillons les outils de développement utilisés, et nous clôturons ce chapitre par des captures d'écran illustrant les différentes fonctionnalités de notre application.

## 2 ENVIRONNEMENTS DE TRAVAIL ET CHOIX TECHNIQUES :

Cette phase met en lumière les outils logiciels, les Framework, et les langages de programmation choisis pour mener à bien le développement de notre plateforme.

### 2.1 Framework et Environnements de travail :

#### **Visual Studio Code :**



Visual Studio Code est un éditeur de texte open source, gratuit et multiplateforme (Windows, Mac et Linux), développé par Microsoft. Principalement conçu pour le développement d'applications avec JavaScript, TypeScript et Node.js, l'éditeur peut s'adapter à d'autres types de langages grâce à un système d'extension bien fourni.

### **Overleaf :**



Overleaf est une plateforme pratique pour les utilisateurs de LaTeX, offrant des fonctionnalités de collaboration et une expérience d'édition en ligne sans tracas

### **XAMPP :**



XAMPP est un logiciel gratuit et open-source qui facilite la création et la gestion de serveurs web locaux. Il comprend des composants tels qu'Apache, MySQL, PHP et Perl, ce qui en fait une solution tout-en-un pour développer des applications web sur votre propre ordinateur.

### **StarUML :**



StarUML est un outil de modélisation UML (Unified Modeling Language) largement utilisé dans le domaine du développement logiciel. Il permet aux développeurs de créer des diagrammes UML pour concevoir, visualiser et documenter des systèmes logiciels.

## **GIT :**



Git est un **système de gestion de versions décentralisé** permettant de suivre l'historique des modifications apportées aux fichiers d'un projet. Il facilite la collaboration entre développeurs en permettant de travailler simultanément sur différentes branches du code, tout en gardant une trace claire et détaillée de chaque modification. Grâce à Git, il est possible de tester de nouvelles fonctionnalités sans impacter la version principale du projet, puis de fusionner les changements lorsque ceux-ci sont validés.



## 2.2 Langages de programmation utilisés :

**PHP :**



PHP, acronyme de "Hypertext Preprocessor", est un langage de programmation open-source largement utilisé pour le développement web. Il est spécialement conçu pour être intégré dans des pages HTML et est souvent utilisé pour créer des sites web dynamiques et interactifs.

**HTML :**



HTML (*HyperText Markup Language*, en français : **langage de balisage hypertexte**) est le **langage de base utilisé pour structurer le contenu des pages web**. Il permet de définir les différents éléments d'une page tels que les titres, les paragraphes, les images, les liens ou encore les tableaux.

## **CSS :**



Les feuilles de style en cascade (CSS, Cascading Style Sheets) sont un langage de style utilisé pour décrire la présentation des documents HTML et XML. CSS permet de contrôler l'apparence 37 Table des figures visuelle des pages.

## **JavaScript :**



JavaScript est un langage de programmation de scripts principalement utilisé pour créer du contenu web interactif. Il permet de rendre les pages web dynamiques et réactives en manipulant le HTML et le CSS.

## **Bootstrap :**



Bootstrap est un framework front-end open-source développé par Twitter. Il fournit des outils et des modèles pour la création d'interfaces web réactives et mobiles conviviales. Bootstrap utilise HTML, CSS et JavaScript pour faciliter le développement de sites web responsives et attrayants.

### 3 PRINCIPALES INTERFACES GRAPHIQUES :

#### 3.1 Interface d'accueil :











































