



Université Cadi Ayyad
Ecole National des Sciences Appliquées de Safi
Département Informatique, Réseaux et Télécommunication
Filière : Génie Informatique et Intelligence Artificielle



Rapport de Mini Projet

Réalisation d'une Application Web

Healify Intégrant L'IA dans l'Analyse Médicale

Réalisé par:

NEDJAOUI Ihssane

AJGAGAL Asma

Encadré par:

Prof. ZETTAM Manal

Soutenu le 23 juin 2025 devant le jury composé de :

Prof. ZETTAM Manal, Professeur à l'ENSAS, Jury

Prof. SOUNNI Hind, Professeur à l'ENSAS, Jury

2ème année Ingénierie des donnée et Intelligence Artificielle

Année Universitaire: 2024/2025

DÉDICACE

À Dieu, pour Sa guidance inébranlable tout au long de ce voyage.
Nous Le remercions de tout cœur pour Sa lumière qui a éclairé notre chemin et pour Sa grâce qui a rendu notre réussite possible.

À nos chers parents, vos encouragements incessants, vos sacrifices inestimables et votre confiance absolue en nous ont été les fondations solides sur lesquelles nous avons pu construire nos rêves et atteindre nos objectifs.

À nos professeurs et mentors, pour votre dévouement sans faille à nourrir nos esprits, à éveiller notre curiosité et à élargir constamment nos horizons. Nous vous sommes profondément reconnaissants pour votre précieuse influence sur nos vies.

Que cette dédicace soit le reflet sincère de notre profonde gratitude envers toutes celles et tous ceux qui, de près ou de loin, ont contribué à notre succès et à l'aboutissement de ce travail.

Avec tout notre amour et notre reconnaissance.

REMERCIEMENT

Nous tenons à exprimer notre profonde gratitude à **Madame Zettam Manal**, dont le soutien indéfectible et les conseils avisés ont été une source d'inspiration tout au long de ce projet. Sa présence bienveillante et son expertise ont été des éléments essentiels de notre réussite.

Nous adressons également nos sincères remerciements à nos **professeurs et encadrants**, pour leur mentorat précieux et leur soutien inestimable. Leurs connaissances approfondies et leur dévouement ont été des atouts précieux dans notre parcours académique.

À nos **familles et amis**, nous exprimons notre reconnaissance pour leur soutien constant et leur encouragement inébranlable.

Que cette page de remerciements soit le témoignage de notre gratitude envers tous ceux qui ont rendu cela possible.

LISTE DES TABLEAUX

Table 1 : les résultats de l'entrainement	22
---	----

LISTE DES FIGURES

Figure 1 : VS Code	5
Figure 2 : Looping	5
Figure 3 : Laragon	6
Figure 4 : Python	6
Figure 5 : Django	6
Figure 6 : Pytorch	6
Figure 7 : TensorFlow	7
Figure 8 : SQLite	7
Figure 9 : MongoDB Atlas	7
Figure 10 : diagramme de cas d'utilisation	9
Figure 11 : diagramme de classes	10
Figure 12 : le principe générale	14
Figure 13 : le téléchargement de llm de base	15
Figure 14 : les paramètres de Lora	15
Figure 15 : les paramètres de l'entraînement	17
Figure 16 : le principe du rag	20
Figure 17 : schéma de fonctionnement du modèle DeepSeek	24
Figure 18 : Navigation principale	25
Figure 19 : Section "Why Choose Healify?"	26
Figure 20 : la section des services	27
Figure 21 : La page About	28
Figure 22 : La page Contact	29
Figure 23 : page Login	30
Figure 24 : page Register	31
Figure 25 : Page Healytics	32
Figure 26 : Healytics-chat	32
Figure 27 : medical assistant(nouvelle-conversation)	33
Figure 28 : medical assistant(ancienne-conversation)	33
Figure 29 : Interaction textuelle avec le Chatbot (envoi de la requête)	34
Figure 30 : Interaction textuelle avec le Chatbot (inférence)	34
Figure 31 : Analyse d'images médicales (envoi de la requête)	35
Figure 32 : Analyse d'images médicales (inférence)	35

TABLE DES MATIERES

DÉDICACE	II
REMERCIEMENT	III
LISTE DES TABLEAUX	IV
LISTE DES FIGURES	V
INTRODUCTION GÉNÉRALE	1
CHAPITRE 1: REALISATION DU CAHIER DE CHARGE ET CHOIX DES OUTILS DE DEVELOPPEMENT	2
1. Introduction	2
2. L'état d'art de l'IA en Médecine	2
3. Cahier de charge	4
3.1. Problématique	4
3.2. Objectifs	4
3.3. Fonctionnalités principales	4
4. Outils de développement	5
4.1. Les logiciels	5
4.2. Les langages	6
4.3. Les frameworks	6
4.4. Base de données	7
4.5. Fournisseur Cloud	7
5. Conclusion	8
CHAPITRE 2 : SPÉCIFICATIONS TECHNIQUES	9
1. Introduction	9
2. Modélisation UML	9
2.1. Diagramme de cas d'utilisation	9
2.2. Diagramme de classes	10
3. Fonctionnalités principales	10
3.1. Healytics	10
3.2. HealyChat	17
3.3. SkinGuard (Détection des Maladies de Peau par CNN)	20
3.4. Assistant Médical basé sur DeepSeek	23
4. Conclusion	24
CHAPITRE 3 : RÉALISATION ET MISE EN OEUVRE	25
1. Introduction	25
2. Page d'accueil	25
2.1 Navigation principale	25

2.2 Section "Why Choose Healify?"	26
2.3 Présentation des services	26
3. La page About.....	27
4. La page Contact.....	28
5. Pages d'authentification	29
5.1 Page Login.....	29
5.2 Page Register	30
6. Présentation des services	31
7. Conclusion	36
CONCLUSION.....	37
WEBOGRAPHIE	38

INTRODUCTION GÉNÉRALE

Comprendre un dossier médical peut souvent s'apparenter à déchiffrer un code complexe. Face à un jargon technique, des abréviations et des données cliniques, de nombreuses personnes se sentent démunies, incapables de saisir pleinement les informations concernant leur propre santé. Cette **lacune de compréhension** crée une dépendance constante vis-à-vis d'une aide externe – qu'il s'agisse de médecins, d'infirmiers ou de spécialistes – pour interpréter des documents pourtant essentiels à leur bien-être et à leur prise de décision.

Dans un contexte où l'autonomie du patient et l'accès à une information claire sont primordiaux, l'idée de notre projet **Healify** est née. Il s'agit d'une réponse concrète au sentiment profond de la nécessité d'offrir aux individus la capacité de comprendre et de s'approprier leur propre dossier médical, sans nécessiter d'assistance systématique. Nous nous positionnons comme les porte-parole de ceux qui peinent à décrypter un langage qui devrait leur être accessible.

Notre projet, **Healify**, consiste au développement d'une solution numérique visant à faciliter la compréhension des rapports médicaux. Notre objectif est de rendre ces documents **clairs, accessibles et compréhensibles** pour chacun, permettant ainsi aux patients de mieux s'approprier leur parcours de soins et de participer activement à leur santé. Cette initiative offre des opportunités aux individus d'interagir plus directement avec leurs informations médicales.

Healify a aussi l'ambition de transformer la manière dont les patients interagissent avec leurs données de santé, en démystifiant le langage médical et en favorisant une meilleure autonomie.

Dans ce rapport, nous présentons trois chapitres distincts. le premier chapitre explore le cahier des charges en mettant en relief la problématique et les objectifs à atteindre, ainsi que les outils de développement choisis. Le deuxième identifie les exigences techniques et les fonctionnalités principales de notre solution. Et finalement, met en lumière les efforts investis dans la création de notre application et les résultats obtenus.

CHAPITRE 1: REALISATION DU CAHIER DE CHARGE ET CHOIX DES OUTILS DE DEVELOPPEMENT

1. Introduction

Ce chapitre présente le contexte générale de notre projet qui se base sur la mise en relief de l'état d'art de l'intelligence artificielle dans le domaine de la médecine, ses défis et son future, la réalisation du cahier de charge avec la problématique que notre projet essaye de résoudre et finalement les outils de développement auxquels nous avons eu recours pour la réalisation de ce projet.

2. L'état d'art de l'IA en Médecine

De nos jours l'intelligence artificielle est devenue un levier pour le développement des pays et des économies, elle a démontré son influence et son impact dans plusieurs domaines et le domaine médicale ne fait pas l'exception, cela peut être facilement constater par ses applications dans le secteur médical.

2.1. Applications de l'IA dans le Secteur Médical

Les applications de l'IA dans le secteur médical varient amplement, ce qui s'explique par la demande croissante des outils intelligents d'assistance pour les médecins et les cadres hospitaliers dont on peut citer à titre d'exemple:

- **Diagnostic et Détection Précoce:** L'intelligence artificielle révolutionne le diagnostic et la détection précoce des maladies en donnant aux professionnels de la santé des outils d'analyse évolués. Par exemple, l'analyse d'imagerie médicale à l'aide d'IA permet d'examiner des radiographies, des IRM et des scanners avec une exactitude renforcée, identifiant des anomalies finement subtiles souvent indécélables à l'œil nu.
- **Traitement et Personnalisation des Soins:** L'IA ouvre la voie à des traitements plus précis et adaptés à chaque patient. La planification personnalisée du traitement s'appuie sur l'étude des données individuelles (génomiques, cliniques, mode de vie) pour prescrire des stratégies thérapeutiques idéales, maximisant l'efficacité et les minimisant les effets secondaires
- **Prévention et Gestion de la Santé:** L'IA joue un rôle crucial dans la prévention des maladies et la gestion proactive de la santé. L'analyse prédictive des risques de maladie, basée sur l'étude

de vastes ensembles de données permet d'identifier les facteurs de risque et de prédire la probabilité de développer certaines affections, ouvrant la voie à des interventions préventives ciblées.

2.2. Défis et Limites de l'IA dans le Domaine Médical

Bien que le potentiel de l'intelligence artificielle en matière de domaine médical soit immense, son adoption et intégration à grande échelle sont confrontées à des défis et limites majeures.

- **Qualité, Disponibilité et Biais des Données:** L'IA besoin de données de grande qualité et représentatives, fréquemment difficiles à obtenir et sujets aux biais qui peuvent conduire à des disparités dans les soins.
- **Manque de Transparence et d'Interprétabilité:** La nature complexe de certains modèles d'IA rend difficile à faire comprendre leur raisonnement, ce qui soulève des interrogations sur la confiance et la responsabilité des décisions médicales.
- **Défis Réglementaires et d'Intégration Clinique:** L'ajustement des réglementations et l'intégration sans heurts de l'IA dans les pratiques médicales en place demandent des cadres clairs et une gestion du changement efficace.

2.3. Tendances Actuelles et Futures de l'IA en Médecine

L'avenir de l'IA en médecine repose sur l'intégration de plusieurs avancées majeures. L'IA multimodale combine diverses sources de données patient (imagerie, génomique, texte clinique, signaux physiologiques) pour une compréhension plus complète et des diagnostics plus précis. L'intégration des grands modèles de langage (LLM) exploite leur vaste connaissance textuelle pour extraire des informations complexes, assister la prise de décision clinique et améliorer la communication avec les patients. Les nouvelles techniques d'IA générative, telles que le fine-tuning (adaptation de modèles à des tâches spécifiques) et la computer vision (analyse d'images médicales), révolutionnent le diagnostic précoce, la découverte de médicaments et la personnalisation des traitements. Ces tendances interconnectées sont fondamentales pour le progrès de l'IA en médecine et constituent des approches clés que nous utilisons activement dans notre projet pour développer des solutions d'assistance médicale intelligentes et innovantes, visant à transformer positivement les soins aux patients.

3. Cahier de charge

3.1. Problématique

La problématique centrale de notre projet réside dans la difficulté qu'éprouvent de nombreux patients à interpréter leurs résultats d'analyses médicales. Rédigés dans un jargon technique, ces documents cruciaux restent souvent obscurs, entravant une compréhension éclairée de leur état de santé. Cette complexité conduit à une dépendance envers des explications vulgarisées, souvent incomplètes, fournies par des non-médecins. De plus, la réticence des patients à poser des questions aux professionnels de santé, par timidité ou crainte de déranger, freine l'acquisition d'une information complète et nuit à leur implication active dans la gestion de leur santé.

En parallèle, nous observons un besoin pressant concernant la gestion de l'anxiété liée aux anomalies cutanées. L'apparition de symptômes dermatologiques visibles génère une inquiétude immédiate, incitant les individus à rechercher une évaluation rapide pour apaiser leurs craintes. L'attente d'une consultation spécialisée peut exacerber cette angoisse.

Notre projet ambitionne de répondre à ces deux enjeux majeurs : rendre l'information médicale complexe plus accessible et compréhensible pour les patients, et offrir une première ligne d'assistance et d'orientation face aux préoccupations dermatologiques courantes, réduisant ainsi l'anxiété et facilitant une prise en charge appropriée.

3.2. Objectifs

Notre objectif principal est de développer une solution d'aide automatique aux patients qui, par mise en clair des résultats des analyses médicales par une description simple et facilement accessible, tient à autonomiser les patients en matière de santé. Cette solution réalise également une première évaluation et orientation sur les dermatologiques les plus fréquentes, dissipant l'anxiété et favorisant ainsi une prise en charge rapide et éclairée, renforçant ainsi leur parcours de soins global.

3.3. Fonctionnalités principales

- **Home Page:** La page d'accueil de notre site est le point de départ pour découvrir l'association des animaux et tout ce que nous avons à offrir. Mettant en lumière :
 - ✓ Les options de navigation
 - ✓ Les services fournies par la platform “**healytics**”.

✓ Les informations nécessaire pour le contact de l'équipe

- **Healytics Page:** Autorise l'utilisateur à télécharger un document d'analyse médicale en forme PDF pour obtenir une interprétation et un éclaircissement des résultats.
- **Login et Register Page:** Les traditionnelles fonctions d'authentification pour la paramétrage des comptes utilisateurs.
- **Healytics Chat Page:** Un chat intelligent en mesure de répondre aux questions des utilisateurs sur les analyses médicales ils ont soumis.
- **Page Skinguard:** Fonction spécialisée en dermatologie pouvant prédire la forme de maladie dermatologique en fonction d'informations fournies par l'utilisateur. Elle donne également des informations supplémentaires pour permettre à l'utilisateur de mieux s'engager avec la condition déterminée.
- **Page de contact:** Section donnant la possibilité aux utilisateurs de prendre contact avec l'équipe de support ou de demander l'aide.

4. Outils de développement

4.1. Les logiciels

Vs Code

Souvent abrégé VS Code Visual Studio Code est un éditeur de code open source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions. Il supporte l'auto complétion, la coloration syntaxique, le débogage, et les commandes git, l'intégration de ce logiciel a joué un rôle dans le développement de notre projet.



Figure 1 : VS Code

Looping

C'est un logiciel de modélisation conceptuelle de données qui nous a permis d'organiser et structurer les données en des diagrammes et représentations graphiques ordonnées. L'utilisation de cet outil était une étape cruciale dans la réalisation du MCD et du MLD.



Figure 2 : Looping

Laragon

Laragon est un environnement de développement local portable et ultra-rapide pour Windows. Il permet de configurer en quelques secondes un serveur web complet (Apache/Nginx, MySQL/PostgreSQL/MongoDB, PHP/Python/Node.js), idéal pour développer et tester localement des applications web ou des projets nécessitant une base de données.



Figure 3 : Laragon

4.2. Les langages

Python

Python est un langage de programmation de haut niveau, polyvalent et reconnu pour sa lisibilité et sa simplicité. Il est utilisé pour une vaste gamme d'applications, incluant le développement web, l'analyse de données, l'intelligence artificielle et le calcul scientifique.



Figure 4 : Python

4.3. Les frameworks

Django

Django est un framework web Python de haut niveau qui permet le développement rapide de sites web sécurisés et maintenables. Il suit le modèle architectural Model-View-Template (MVT) et fournit un ensemble robuste d'outils pour la création d'applications web.



Figure 5 : Django

Pytorch

PyTorch est un framework open-source d'apprentissage automatique (machine learning) développé par le laboratoire de recherche en IA de Facebook. Il est reconnu pour sa flexibilité et est populaire pour la recherche et le prototypage en apprentissage profond (deep learning).



Figure 6 : Pytorch

Tensor Flow

TensorFlow est un framework open-source d'apprentissage automatique (machine learning) développé par Google. Il est largement utilisé pour une variété de tâches d'apprentissage automatique, y compris l'apprentissage profond (deep learning), et est reconnu pour son évolutivité et sa capacité à être mis en production.



Figure 7 : TensorFlow

4.4. Base de données

SQLite

SQLite est une bibliothèque logicielle écrite en langage C, qui implémente un moteur de base de données SQL autonome, sans serveur et ne nécessitant aucune configuration. Contrairement aux systèmes de gestion de bases de données traditionnels (comme MySQL ou PostgreSQL), SQLite est directement intégrée aux applications et stocke l'intégralité de la base de données dans un seul fichier indépendant de la plateforme.



Figure 8 : SQLite

4.5. Fournisseur Cloud

MongoDB Atlas

MongoDB Atlas se distingue pour l'IA grâce à sa recherche vectorielle, une fonctionnalité clé pour stocker et interroger efficacement les embeddings. Ces représentations numériques sont vitales pour les recherches de similarité rapides, essentielles aux moteurs de recommandation, à la recherche sémantique, et à l'optimisation des LLM via la récupération augmentée de données (RAG). Il offre ainsi une plateforme unifiée pour vos données et vecteurs, optimisant les performances de vos applications d'IA.



Figure 9 : MongoDB Atlas

5. Conclusion

Ce chapitre a offert un survol de niveau élevé de plusieurs des technologies clés du monde du développement logiciel moderne. Nous avons également examiné la position actuelle de l'IA, le cahier des charges et les outils de développement. Dans le prochain chapitre, nous plongerons dans les aspects techniques de ces technologies, en examinant leur architecture, leurs fonctionnalités, la manière dont le travail a été réellement effectué

CHAPITRE 2 : SPÉCIFICATIONS TECHNIQUES

1. Introduction

Dans ce chapitre nous allons découvrir les détails techniques derrière la réalisation de ce projet, nous explorons en détails les modèles choisis ainsi que les données sur lesquels entraînement était fait, nous aurons une idée sur les mesures de performance de chaque modèle.

2. Modélisation UML

La modélisation UML constitue la base sur laquelle repose la réalisation de n'importe quel projet. Une bonne modélisation reflète la compréhension du cahier des charges et des besoins fonctionnels, et elle assure le bon déroulement de la phase de développement du projet. Pour la conception de ce projet, nous avons eu recours à la méthode UML qui nous a permis de créer des diagrammes traduisant les besoins des utilisateurs et les interactions principales de l'application.

2.1. Diagramme de cas d'utilisation

Ce diagramme illustre les fonctionnalités principales offertes aux utilisateurs (patients) dans le système de santé AI. Il met en évidence les interactions possibles, notamment la gestion des conversations, l'analyse des dossiers médicaux, et les échanges avec le chatbot.

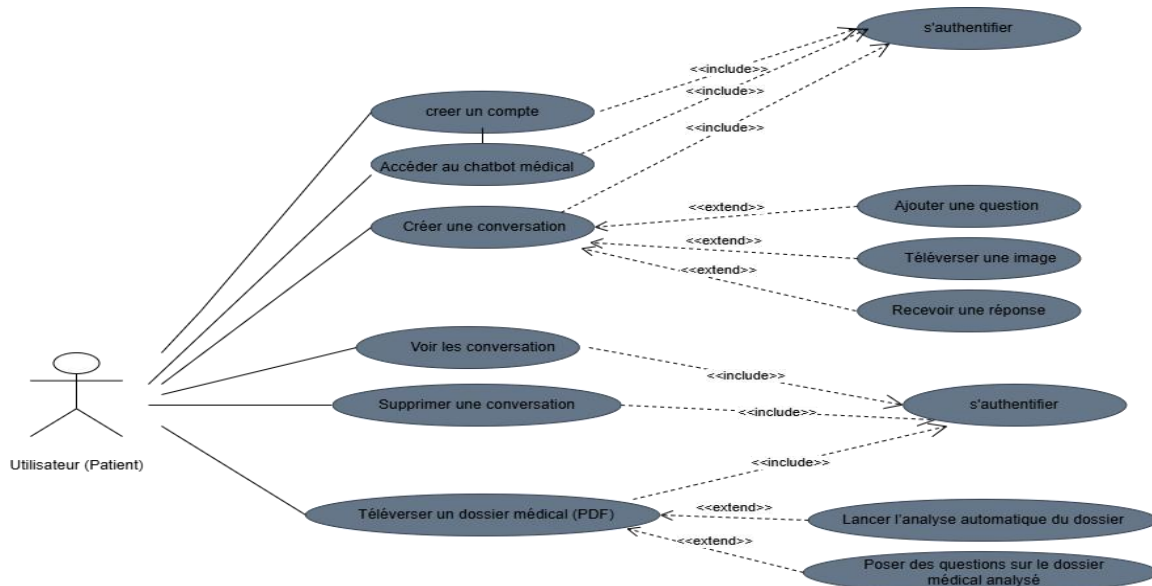


Figure 10 : diagramme de cas d'utilisation

2.2. Diagramme de classes

Ce diagramme schématise les entités principales du système et leurs relations. Avec les classes Patient, Conversation, Message, MedicalRecord, History, et MedicalChat.

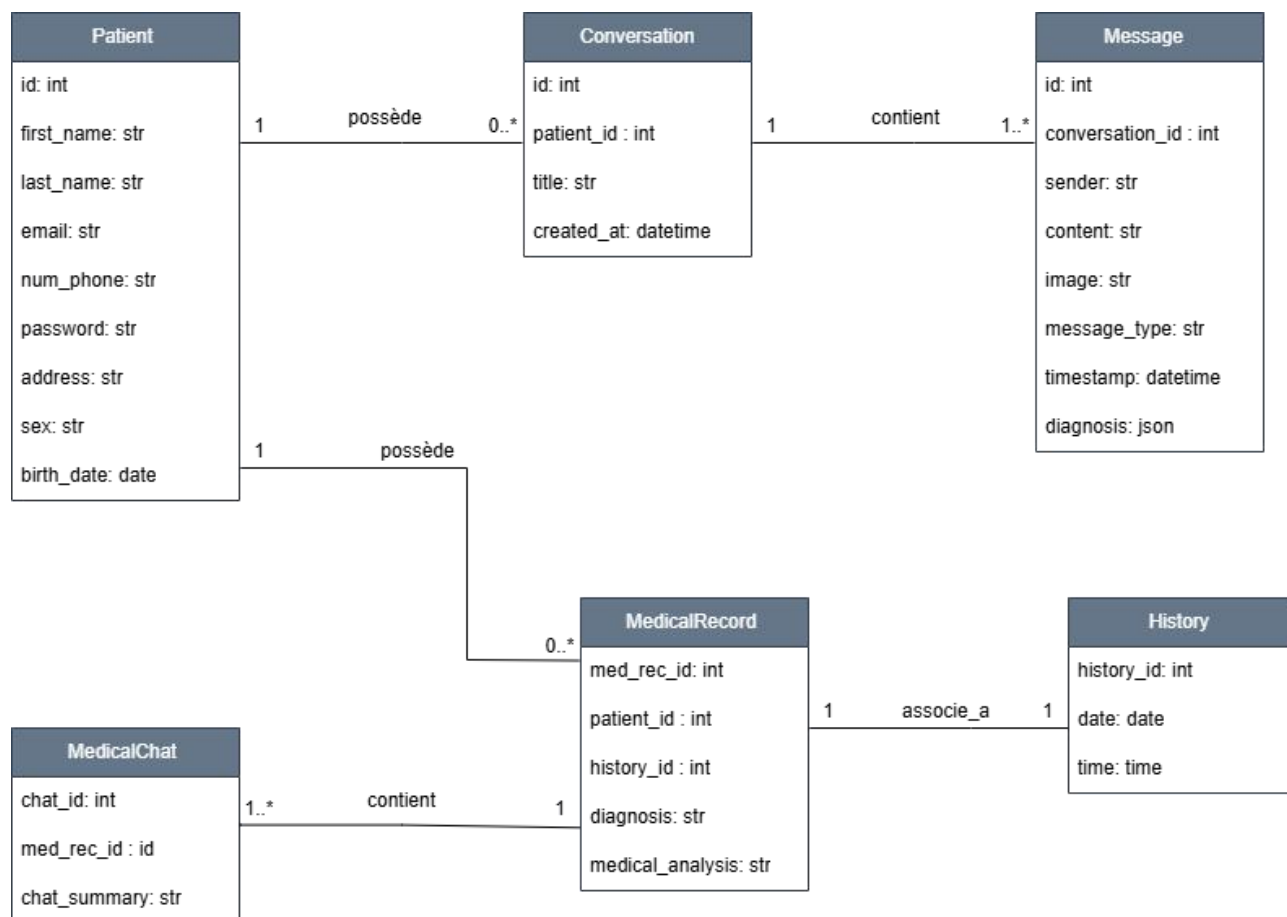


Figure 11 : diagramme de classes

3. Fonctionnalités principales

3.1. Healytics

3.1.1.Introduction au modèle

Notre projet s'est concentré sur le développement d'un modèle dont la mission principale est d'interpréter les résultats médicaux d'une manière simplifiée pour les utilisateurs ne possédant pas le jargon médical. Pour ce faire, nous avons utilisé un grand modèle linguistique (LLM) pré-entraîné, spécifiquement adapté à cette tâche d'interprétation des résultats médicaux grâce à une technique

appelée "fine-tuning". La génération de données synthétiques n'a été qu'une solution nécessaire face à l'absence de données réelles adaptées à cet objectif.

Le modèle de base choisi pour cette première itération est **meta-llama/Llama-3.2-1B-Instruct**. Ce modèle appartient à la famille des architectures Transformer, qui sont devenues la norme dans le domaine du traitement du langage naturel (NLP) en raison de leur capacité à traiter de longues séquences de données et à capturer des dépendances complexes.

➤ **le Fine-tuning**

Le fine-tuning (ou "réglage fin" en français) est une technique d'apprentissage automatique qui consiste à prendre un modèle qui a déjà été entraîné sur un très grand ensemble de données (un "modèle pré-entraîné") et à l'adapter à une tâche plus spécifique ou à un ensemble de données plus petit.

Dans le cadre de ce projet, nous avons spécifiquement utilisé la technique de LoRA (Low-Rank Adaptation) pour le fine-tuning. LoRA est une méthode qui permet d'adapter de grands modèles linguistiques de manière beaucoup plus efficace en termes de calcul et de mémoire. Au lieu de modifier tous les poids du modèle pré-entraîné, LoRA introduit de petites matrices de faible rang dans les couches du Transformer, et seuls les poids de ces matrices sont entraînés. Cela réduit considérablement le nombre de paramètres entraînables, ce qui rend le processus de fine-tuning plus rapide et moins gourmand en ressources.

➤ **les cas d'utilisation**

Le fine-tuning est une technique qui est amplement dans le domaine du traitement du langage naturel pour plusieurs raisons:

- ✓ **Exploitation des Connaissances Préexistantes** : Les LLM comme Llama-3.2-1B-Instruct sont entraînés sur des quantités massives de texte, ce qui leur confère une compréhension générale du langage, de la grammaire, de la sémantique et même de certaines connaissances factuelles. Le fine-tuning nous permet de capitaliser sur cette base de connaissances étendue, plutôt que de repartir de zéro.
- ✓ **Efficacité de l'Entraînement** : Entraîner un LLM à partir de zéro est extrêmement coûteux en termes de temps et de ressources computationnelles. Le fine-tuning est beaucoup plus rapide et

moins gourmand en ressources, car il ne nécessite que des ajustements mineurs aux poids du modèle pré-entraîné.

- ✓ **Moins de Données Nécessaires** : Pour une tâche spécifique, le fine-tuning requiert généralement beaucoup moins de données d'entraînement que l'entraînement d'un modèle à partir de zéro, car le modèle a déjà appris des représentations de haut niveau du langage.
- ✓ **Meilleures Performances sur Tâche Spécifique** : En adaptant le modèle à notre domaine spécifique (les notes médicales) et à notre tâche (génération à partir de rapports), le fine-tuning permet au modèle de mieux comprendre et de produire du texte pertinent et cohérent dans ce contexte particulier, surpassant souvent un modèle générique ou un modèle entraîné de zéro avec un ensemble de données limité.

En somme, le fine-tuning nous a permis d'obtenir un modèle performant pour la génération de notes médicales synthétiques, en tirant parti de la puissance d'un LLM pré-entraîné tout en l'adaptant précisément aux spécificités de notre projet.

3.1.2. Les exigences des données

➤ La Rareté des Données Médicales Réelles

Le développement de systèmes d'IA performants dans le domaine médical est intrinsèquement lié à la disponibilité de données de haute qualité. Cependant, l'accès aux données médicales réelles, et plus particulièrement aux notes de médecins (physician notes), est extrêmement limité. Cette rareté est principalement due à des contraintes strictes de confidentialité des patients (réglementations telles que HIPAA, RGPD) et à des considérations éthiques rigoureuses. Les notes cliniques, qui contiennent des observations détaillées, des raisonnements diagnostiques et des plans de traitement, sont considérées comme des informations de santé protégées (PHI) et ne peuvent être partagées ou utilisées à grande échelle sans des processus d'anonymisation complexes et des accords de données spécifiques.

Cette limitation pose un défi majeur pour l'entraînement de modèles de traitement du langage naturel (TLN) visant à comprendre et à générer du texte médical, car ces modèles nécessitent de vastes quantités de données pour apprendre les nuances du langage clinique.

➤ **Approche par Données Synthétiques**

Face à cette pénurie de notes médicales réelles et à la nécessité de disposer de données pour le fine-tuning de notre modèle, une approche innovante a été adoptée : la génération de notes médicales synthétiques. Cette stratégie nous a permis de contourner les défis de confidentialité tout en créant un corpus de données pertinent pour notre tâche.

Le processus a impliqué les étapes suivantes :

Utilisation de Données Tabulaires Disponibles : Nous avons tiré parti du dataset de démonstration MIMIC-III, qui est publiquement accessible et dé-identifié. Bien que ce dataset soit principalement sous forme tabulaire, il contient des informations cliniques structurées précieuses.

Conversion en Rapports Médicaux Mimétiques : Une sélection de ces tables structurées a été transformée et organisée pour imiter la structure et le contenu des rapports médicaux réels. L'objectif était de créer des "rapports médicaux" synthétiques qui serviraient de base pour la génération de notes narratives.

Génération de Notes Synthétiques : Ces rapports médicaux mimétiques ont ensuite été utilisés comme intrants pour un grand modèle linguistique (LLM) pré-entraîné (**meta-llama/Llama-3.2-3B-Instruct**), qui a généré les notes médicales synthétiques correspondantes. Le LLM a été guidé pour interpréter les informations des rapports et les transformer en un texte narratif simplifié, simulant ainsi le processus d'un médecin rédigeant une note.

Utilisation pour le Fine-tuning : C'est cet ensemble de paires "rapport médical mimétique + note médicale synthétique" qui a constitué le dataset d'entraînement pour le fine-tuning de notre modèle principal. Cette approche a permis au modèle d'apprendre à interpréter des informations structurées et à les reformuler dans un langage simplifié, atteignant ainsi notre objectif de rendre les résultats médicaux accessibles aux non-spécialistes.

Cette méthodologie a été cruciale pour surmonter les obstacles liés à la disponibilité des données et a permis de créer un environnement d'entraînement contrôlé et éthique pour notre modèle.

3.1.3. Principe de fonctionnement

Le processus débute par l'extraction du texte brut à partir du document PDF fourni par l'utilisateur. Par la lecture directe du contenu textuel du pdf. Une fois le texte récupéré et nettoyé, il est alors envoyé directement au Grand Modèle de Langage (LLM) qui a été préalablement affiné (fine-tuned). Ce LLM spécifique est entraîné pour comprendre et traiter ce type de données contextuelles, lui permettant ainsi d'analyser le contenu du PDF avec une pertinence accrue et de générer une réponse précise basée sur les informations extraites. C'est cette transmission ciblée du texte du PDF au LLM affiné qui permet au système de fournir des réponses intelligentes et contextualisées.

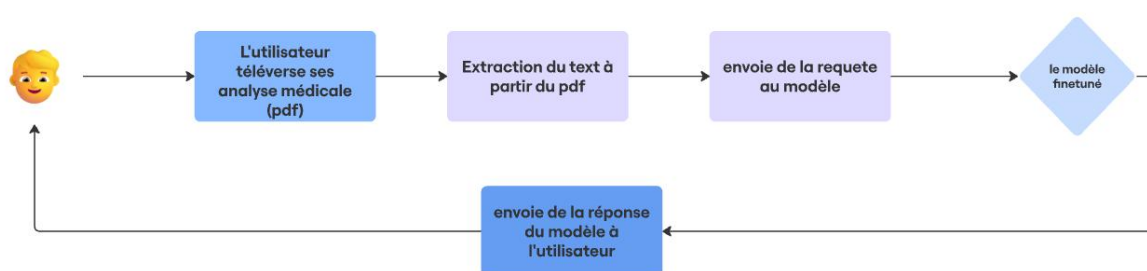


Figure 12 : le principe générale

3.1.4. La phase de l'entrainement

➤ Le téléchargement du modèle

L'affinage (fine-tuning) d'un Grand Modèle de Langage (LLM) consiste à spécialiser un modèle pré-entraîné, qui possède une vaste connaissance du langage, pour une tâche ou un domaine spécifique. Plutôt que de repartir de zéro, on entraîne le modèle sur un petit ensemble de données pertinent. Cette approche permet au LLM d'acquérir une compréhension et des capacités de génération plus précises pour un domaine particulier, le tout de manière plus efficace en termes de ressources. Des techniques comme la quantification sont souvent utilisées pour réduire l'empreinte mémoire et accélérer le processus.

Ce bloc de code illustre le début de ce processus d'affinage. Il prépare le modèle de base "meta-llama/Llama-3.2-1B-Instruct" à être entraîné sur le jeu de données `medical_dataset` (situé à `/content/sampled_datasets.csv`), avec l'objectif de créer un nouveau modèle affiné nommé

"Ihssane123/Llama-3.2-1B-Instruct-medical". La configuration inclut le chargement du modèle en précision float16 pour optimiser l'utilisation de la mémoire.

```
from transformers import AutoModelForCausalLM, AutoTokenizer, BitsAndBytesConfig, HfArgumentParser, TrainingArguments
import os, torch
from datasets import load_dataset, Dataset
from trl import SFTTrainer
import os
from kaggle_secrets import UserSecretsClient
user_secrets = UserSecretsClient()
access_token = user_secrets.get_secret("mistral_token_r")

#Base Model and Dataset
base_model = "meta-llama/Llama-3.2-1B-Instruct"
medical_dataset = "/content/sampled_datasets.csv"
new_model_name = "Ihssane123/Llama-3.2-1B-Instruct-medical"

bnb_config = BitsAndBytesConfig(
    load_in_4bit= False,
    load_in_8bit= False,
    bnb_4bit_quant_type= "nf4",
    bnb_4bit_compute_dtype= torch.bfloat16,
    bnb_4bit_use_double_quant= False,
)
model = AutoModelForCausalLM.from_pretrained(
    base_model,
    quantization_config=bnb_config,
    torch_dtype=torch.float16,
    trust_remote_code=True,
    token = access_token
)
model.config.use_cache = False
model.config.pretraining_tp = 1
model.gradient_checkpointing_enable()
```

Figure 13 : le téléchargement de llm de base

➤ Les paramètres de Lora

Ce code implémente LoRA (Low-Rank Adaptation), une technique clé pour l'affinage efficace des grands modèles de langage. Au lieu d'entraîner tous les paramètres du modèle, LoRA gèle la majorité d'entre eux et n'ajoute que de petites matrices "adaptateurs" dans des couches spécifiques (q_proj, k_proj, v_proj, etc.). `prepare_model_for_kbit_training` optimise le modèle pour cet entraînement efficace en basse précision. `LoraConfig` définit les hyperparamètres de ces adaptateurs (comme `r`, `lora_alpha`), et `get_peft_model` intègre ces adaptateurs au modèle de base. Cela permet d'entraîner le modèle avec beaucoup moins de ressources, en ne mettant à jour qu'une petite fraction des paramètres, tout en obtenant des performances comparables à un affinage complet.

```
#Adding the adapter to the layer
model = prepare_model_for_kbit_training(model)
peft_config = LoraConfig(
    lora_alpha=16,
    lora_dropout=0.1,
    r=64,
    bias="none",
    task_type="CAUSAL_LM",
    target_modules=["q_proj", "k_proj", "v_proj", "o_proj", "gate_proj"]
)
model = get_peft_model(model, peft_config)
```

Figure 14 : les paramètres de Lora

➤ Données d'Entraînement

Le modèle a été fine-tuné sur un ensemble de données composé de paires où le rapport médical structuré servait d'entrée (input) et la note médicale synthétiquement générée correspondante servait de cible (output). Ces paires de données ont permis au modèle d'apprendre la corrélation entre les informations structurées des rapports et le langage narratif des notes cliniques. L'objectif était que le modèle puisse, à partir d'un rapport médical donné, générer une note synthétique cohérente et pertinente. Les données utilisées sont disponibles sur la plateforme Hugging Faceⁱ. Pour ce projet on a utilisé juste 20 échantillons sélectionnés d'une manière aléatoire à partir du dataset.

➤ Arguments d'Entraînement

L'entraînement du modèle a été effectué en utilisant Lora avec les arguments suivants, choisis pour optimiser la performance et la stabilité du fine-tuning sur une architecture de type Transformer :

Per_device_train_batch_size: 3 (Taille du lot d'entraînement par appareil.)

gradient_accumulation_steps: 2 (Nombre d'étapes d'accumulation de gradient. **learning_rate:** 2e-4 (Taux d'apprentissage. Un taux d'apprentissage faible est typique pour le fine-tuning afin de ne pas "oublier" les connaissances pré-entraînées.)

fp16: False (Désactive l'entraînement en précision mixte (float16).

logging_steps: 100 (Fréquence à laquelle les métriques d'entraînement sont enregistrées.)

output_dir: ./Llama-3.2-1B-Instruct-medical (Répertoire où les checkpoints du modèle et les logs d'entraînement seront sauvegardés.)

optim: paged_adamw_8bit (Optimiseur utilisé. paged_adamw_8bit est une variante optimisée en mémoire d'AdamW, particulièrement utile pour les grands modèles.)

save_steps: 100 (Fréquence à laquelle les checkpoints du modèle sont sauvegardés.)

report_to: none (Outil de reporting pour visualiser les métriques d'entraînement.)

```

training_arguments = TrainingArguments(
    output_dir="./Llama-3.2-1B-Instruct-medical",
    num_train_epochs=1,
    per_device_train_batch_size=3,
    gradient_accumulation_steps=2,
    optim="paged_adamw_32bit",
    save_strategy="epoch",
    logging_steps=100,
    logging_strategy="steps",
    learning_rate=2e-4,
    fp16=False,
    bf16=False,
    group_by_length=True,
    disable_tqdm=False,
    report_to="none",
)

```

Figure 15 : les paramètres de l'entraînement

Ces arguments ont permis de guider le processus d'apprentissage du modèle, en l'aidant à converger efficacement vers une solution capable de générer des notes médicales synthétiques de haute qualité à partir des données structurées.

3.2. HealyChat

3.2.1. Introduction au Modèle

Cette section de notre projet utilise une architecture de Génération Augmentée par Récupération (RAG) pour permettre au chatbot de répondre aux questions liées aux dossiers médicaux des patients. Le choix de RAG était primordial en raison de la nature critique des informations traitées. Lorsqu'il s'agit de données médicales de patients, la capacité à incorporer de nouvelles connaissances spécifiques et à garantir des prédictions très précises n'est pas seulement bénéfique, mais constitue une étape essentielle pour la sécurité des patients et l'efficacité des soins.

➤ Génération Augmentée par Récupération (RAG)

La Génération Augmentée par Récupération (RAG) est un cadre de traitement du langage naturel (TLN) qui améliore les capacités des grands modèles linguistiques (LLM) en y intégrant un composant de récupération d'informations. Traditionnellement, les LLM génèrent des réponses

uniquement basées sur les connaissances pour lesquelles ils ont été entraînés. Bien que puissante, cette approche peut entraîner plusieurs limitations :

1. **Hallucinations** : Les LLM peuvent générer des informations factuellement incorrectes ou absurdes.
2. **Connaissances obsolètes** : Leur base de connaissances est limitée aux données de leur entraînement.
3. **Manque d'expertise spécifique à un domaine** : Ils peuvent avoir des difficultés avec des questions nécessitant des informations très spécialisées ou privées qui ne sont pas largement couvertes dans leurs données d'entraînement publiques.

En intégrant ce mécanisme de récupération, l'architecture RAG transforme un LLM standard en un système puissant, dynamique et conscient du contexte, ce qui en fait un choix idéal pour les applications nécessitant une grande précision et un accès à des informations spécialisées et fréquemment mises à jour, telles que l'interrogation des dossiers médicaux des patients.

➤ **Cas d'utilisation**

Dans le contexte du chatbot répondant à des questions sur les dossiers médicaux des patients, le RAG offre des avantages significatifs :

- **Précision Améliorée** : En basant les réponses du LLM sur des données réelles de patients, le risque d'hallucinations est drastiquement réduit, ce qui conduit à des réponses plus fiables et factuelles.
- **Accès à des Informations à Jour** : De nouveaux dossiers médicaux ou directives peuvent être continuellement ajoutés à la base de connaissances, garantissant que le chatbot a toujours accès aux dernières informations sans nécessiter de réentraînement de l'ensemble du LLM.
- **Expertise Spécifique au Domaine** : RAG permet au chatbot de répondre à des questions très spécifiques sur les antécédents, les diagnostics, les traitements et les médicaments de patients individuels, ce qui serait impossible pour un LLM à usage général.
- **Coûts de Formation Réduits** : Au lieu de réentraîner continuellement des LLM massifs avec de nouvelles informations, RAG permet des mises à jour plus flexibles et efficaces de la base de connaissances.

- **Traçabilité et Explicabilité** : Dans certaines implémentations RAG, il est possible d'identifier les documents sources à partir desquels l'information a été récupérée, offrant un degré de traçabilité et d'explicabilité aux réponses du chatbot, ce qui est crucial dans un contexte médical.

3.2.2. Les données utilisés:

Les données utilisées pour ce projet ont été directement dérivées des dossiers médicaux des patients qui ont été téléversés dans le système. Ces dossiers constituent la base de connaissances fondamentale à partir de laquelle notre chatbot, propulsé par l'architecture RAG, récupère les informations pertinentes.

Chaque dossier médical téléversé a été traité et indexé afin de permettre une recherche et une récupération efficaces par le module de récupération de notre système. Cela garantit que toutes les réponses fournies par le chatbot sont directement ancrées dans les informations cliniques réelles et spécifiques à chaque patient, telles qu'elles figurent dans leurs dossiers officiels.

3.2.3. Le principe du RAG

RAG répond aux limitations déjà citées en récupérant dynamiquement des informations pertinentes à partir d'une base de connaissances externe avant de générer une réponse. Le processus implique généralement deux étapes principales :

Récupération : Lorsqu'un utilisateur pose une question, un composant de récupération recherche dans une vaste collection de documents (par exemple, dossiers médicaux de patients, directives cliniques, articles de recherche) les passages les plus pertinents ou sémantiquement similaires à la requête. Cette récupération peut être alimentée par diverses techniques, y compris les bases de données vectorielles, la correspondance par mots-clés ou les modèles de récupération denses.

Augmentation et Génération : Les passages pertinents récupérés sont ensuite fournis comme contexte supplémentaire au LLM, en même temps que la requête originale de l'utilisateur. Le LLM utilise alors cette entrée augmentée pour générer une réponse plus informée, précise et contextuellement pertinente.

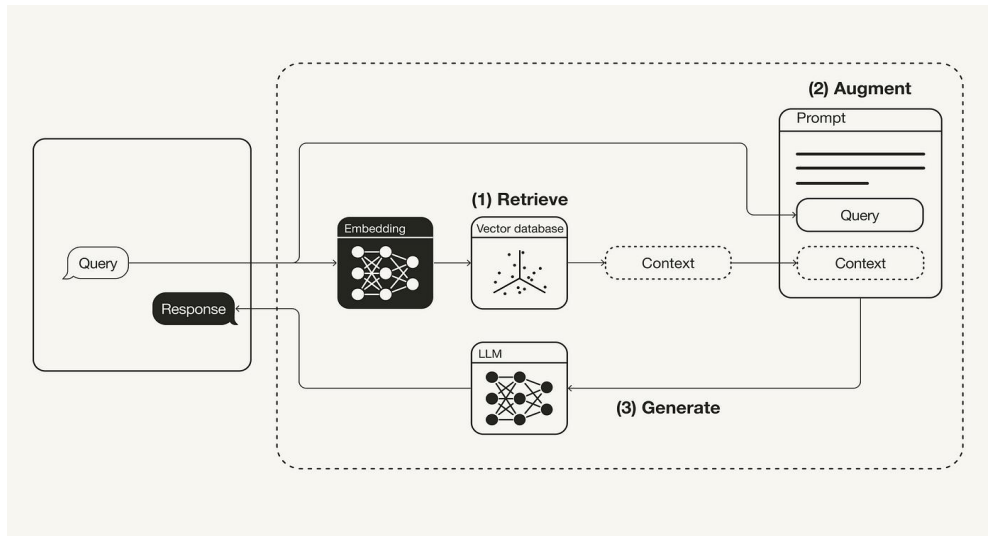


Figure 16 : le principe du rag

3.3. SkinGuard (Détection des Maladies de Peau par CNN)

3.3.1.Introduction au modèle

Notre projet vise à développer un système de diagnostic assisté par IA capable de classer automatiquement les lésions cutanées à partir d'images. Pour cela, nous avons conçu un réseau de neurones convolutifs (CNN) spécialisé dans l'analyse d'images dermatologiques. Contrairement aux modèles linguistiques, notre approche exploite la puissance des CNN pour extraire des motifs visuels hiérarchiques (textures, contours, couleurs) caractéristiques des différentes pathologies.

Le modèle final repose sur une architecture empilement de couches convolutives optimisée pour traiter des images de 300x300 pixels avec une précision de 84% sur les données de test. Il distingue 5 classes de maladies :

- Carcinome basocellulaire (bcc)
- Dermatofibrome (df)
- Mélanome (mel)
- Nævus (nv)
- Lésions vasculaires (vasc)

Cas d'utilisation

Le modèle CNN développé dans ce projet présente plusieurs applications concrètes dans le domaine médical et dermatologique :

- **Aide au diagnostic clinique** : Il peut assister les dermatologues en fournissant une classification automatique des lésions cutanées à partir d'images capturées via smartphone ou appareils médicaux, facilitant ainsi une première évaluation rapide.
- **Triage automatisé des patients** : En identifiant efficacement les lésions suspectes comme le mélanome, le modèle peut aider à prioriser les cas critiques nécessitant une intervention urgente, améliorant la gestion des ressources médicales.
- **Outil de sensibilisation et d'éducation** : Le système peut être intégré dans des applications pour aider le grand public à reconnaître les signes visibles de maladies de peau et encourager une consultation précoce.
- **Support à la recherche dermatologique** : En analysant de grandes bases d'images, le modèle peut contribuer à la collecte et à la classification automatisée des données, accélérant les études épidémiologiques et le développement de nouveaux traitements

3.3.2. Architecture du Modèle

L'architecture du modèle se compose des éléments suivants :

Étape de prétraitement : Les images sont redimensionnées à une taille standard de 300×300 pixels et normalisées pour avoir des valeurs de pixels comprises entre 0 et 1. Ces transformations sont appliquées grâce à un bloc `resize_and_rescale` intégré directement dans le modèle, assurant ainsi la cohérence lors de l'entraînement et de l'inférence.

Augmentation des données : Pour améliorer la robustesse du modèle et limiter l'overfitting, une couche d'augmentation des données (`data_augmentation`) est appliquée au jeu d'entraînement. Cette couche effectue des rotations aléatoires et des inversions horizontales et verticales des images, simulant ainsi la variabilité des images réelles.

Bloc convolutif : Le cœur du modèle est constitué d'une série de six blocs convolutionnels. Chaque bloc comprend :

- Une couche Conv2D avec des filtres croissants (32, puis 64) pour extraire les caractéristiques locales des images.
- Une couche MaxPooling2D pour réduire la dimension spatiale des cartes de caractéristiques tout en conservant les informations principales. L'activation ReLU est utilisée dans toutes les couches convolutionnelles pour introduire de la non-linéarité.

Bloc dense (classification) : Après les couches convolutives, les cartes de caractéristiques sont aplaties (Flatten) et transmises à une couche dense de 64 neurones avec l'activation ReLU, puis à une couche de sortie dense avec une activation softmax pour la classification multi-classes.

3.3.3.Exigences des Données

Dataset et Préparation :

Le dataset utilisé pour ce projet contient 8 589 images réparties en cinq classes distinctes. Ces images proviennent d'un dossier nommé "Skin_disease_img"

➤ Répartition des données :

- Entraînement : 6 871 images (80 %)
- Validation : 859 images (10 %)
- Test : 859 images (10 %)

➤ Techniques de prétraitement :

- Redimensionnement : les images sont redimensionnées à 300×300 pixels.
- Normalisation : les valeurs des pixels sont ramenées entre 0 et 1.
- Augmentation de données : rotations aléatoires et inversions pour améliorer la robustesse.
- Optimisations TensorFlow :
 - ✧ Mise en cache (cache) : accélère l'entraînement en gardant les images en mémoire.
 - ✧ Prélèvement anticipé (prefetch) : précharge les lots (batches) pour optimiser l'utilisation du GPU/CPU.

3.3.4.Phase d'Entraînement

➤ Paramètres Clés

Optimiseur : Adam, avec un taux d'apprentissage adaptatif pour une convergence efficace.

Fonction de perte : SparseCategoricalCrossentropy, adaptée aux labels entiers.

Métrique de suivi : Accuracy. Batch Size : 32 (compromis entre vitesse d'entraînement et précision).

Nombre d'époques : 10, avec un arrêt anticipé implicite basé sur la validation.

➤ Résultats

Époque	Accuracy (Train)	Accuracy (Validation)
10	81.9%	82.7%

Table 1 : les résultats de l'entrainement

La performance finale du modèle sur l'ensemble de test atteint 84,2 %

3.4. Assistant Médical basé sur DeepSeek

3.4.1.Introduction au modèle

Dans le cadre de notre projet, nous avons intégré un modèle basé sur le modèle de langage DeepSeek, un LLM (Large Language Model) de type chatbot médical. Il joue un rôle essentiel en tant qu'assistant médical virtuel interactif, capable de répondre aux questions des utilisateurs et de fournir des informations médicales pertinentes sur les maladies de peau détectées. Grâce à sa capacité de compréhension contextuelle et à son accès à une vaste base de connaissances médicales, DeepSeek enrichit l'expérience utilisateur en apportant des réponses claires, concises et adaptées aux besoins des patients.

Cas d'utilisation

Voici les principaux cas d'utilisation de ce modèle :

- **Réponses aux questions médicales générales** : l'utilisateur peut poser des questions sur les maladies de peau ou la santé dermatologique, et le modèle fournit des explications détaillées, en langage naturel.
- **Complément aux diagnostics visuels** : lorsque l'utilisateur envoie une image, le modèle CNN détecte la pathologie (ex. : mélanome). Le résultat de cette détection est ensuite transmis au modèle DeepSeek, qui fournit des informations médicales supplémentaires sur la maladie détectée (causes, traitements possibles, conseils pratiques).
- **Éducation et sensibilisation** : en délivrant des informations médicales claires et fiables, le modèle favorise la sensibilisation du patient et l'incite à consulter un professionnel de santé en cas de besoin.

3.4.2.Fonctionnement général

1. L'utilisateur envoie un message textuel (question) ou une image.
2. Si c'est une image, le modèle CNN détecte la maladie de peau et transmet le résultat (nom de la maladie) à DeepSeek.
3. DeepSeek reçoit le nom de la maladie (ou la question de l'utilisateur) et génère une réponse textuelle détaillée, en se basant sur ses connaissances médicales internes.
4. La réponse est renvoyée sous forme de texte enrichi (markdown) dans l'interface du chatbot.

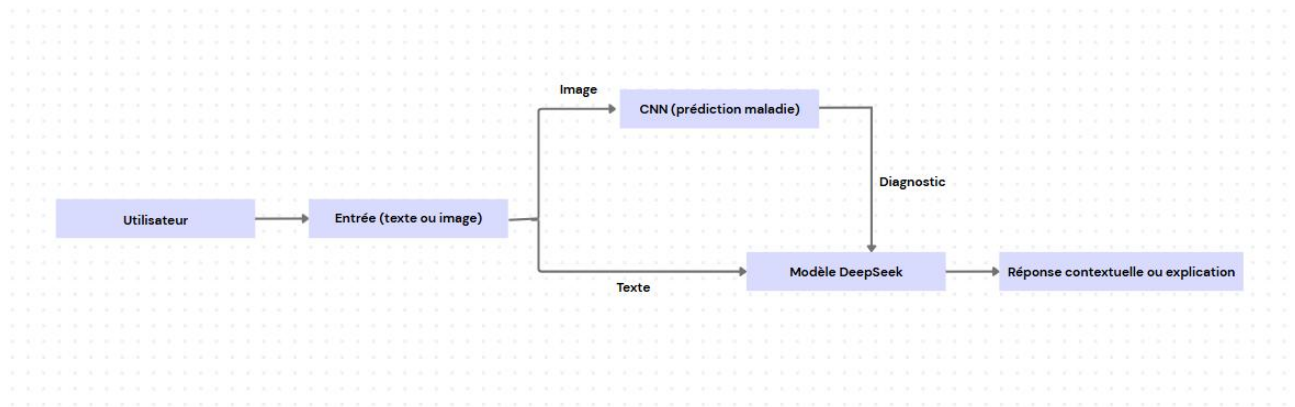


Figure 17 : schéma de fonctionnement du modèle DeepSeek

4. Conclusion

Dans ce chapitre nous avons eu l'occasion de découvrir les aspects techniques derrière la réalisation de notre projet, les techniques de l'IA générative utilisées tel que le finetuning et le rag et comment ils ont été implémenté lors de ce projet.

CHAPITRE 3 : RÉALISATION ET MISE EN OEUVRE

1. Introduction

Dans ce chapitre, nous présentons l'implémentation concrète de notre projet web nommé Healify, une plateforme intelligente dédiée à l'analyse des dossiers médicaux et à la détection des maladies de la peau à l'aide de l'intelligence artificielle. Nous détaillerons chaque partie de l'interface utilisateur avec des captures d'écran et des commentaires explicatifs. La réalisation de ce site a été assurée en respectant les standards modernes de développement web et en intégrant des technologies centrées sur l'accessibilité, l'expérience utilisateur et la performance.

2. Page d'accueil

2.1 Navigation principale

Cette partie contient les informations de contact (numéro de téléphone, adresse), ainsi que les liens vers les réseaux sociaux. Et le menu principal qui comprend les liens vers les pages Home, About, Services et Contact, ainsi qu'un bouton Login.

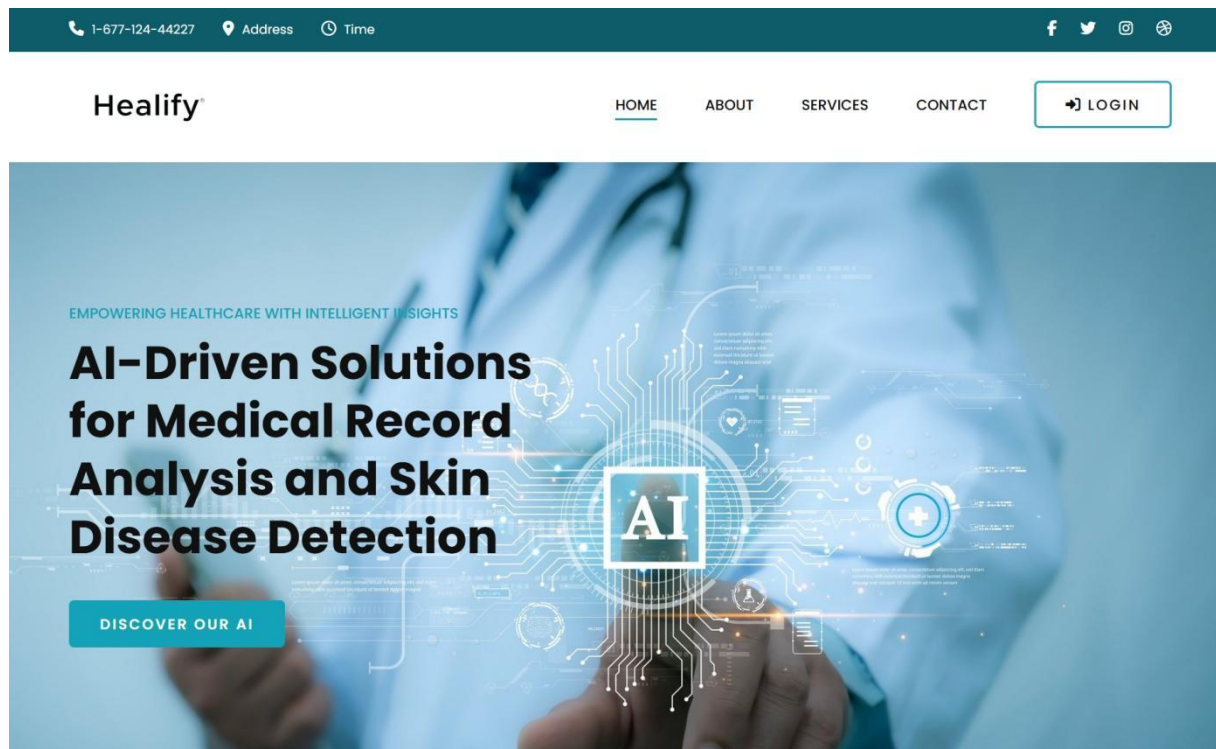


Figure 18 : Navigation principale

2.2 Section "Why Choose Healify?"

Dans cette section, nous mettons en avant les quatre services intelligents proposés par la plateforme :

- Analyse de dossiers médicaux par IA
- Détection de maladies de la peau
- Prédictions basées sur les données
- Suivi médical personnalisé

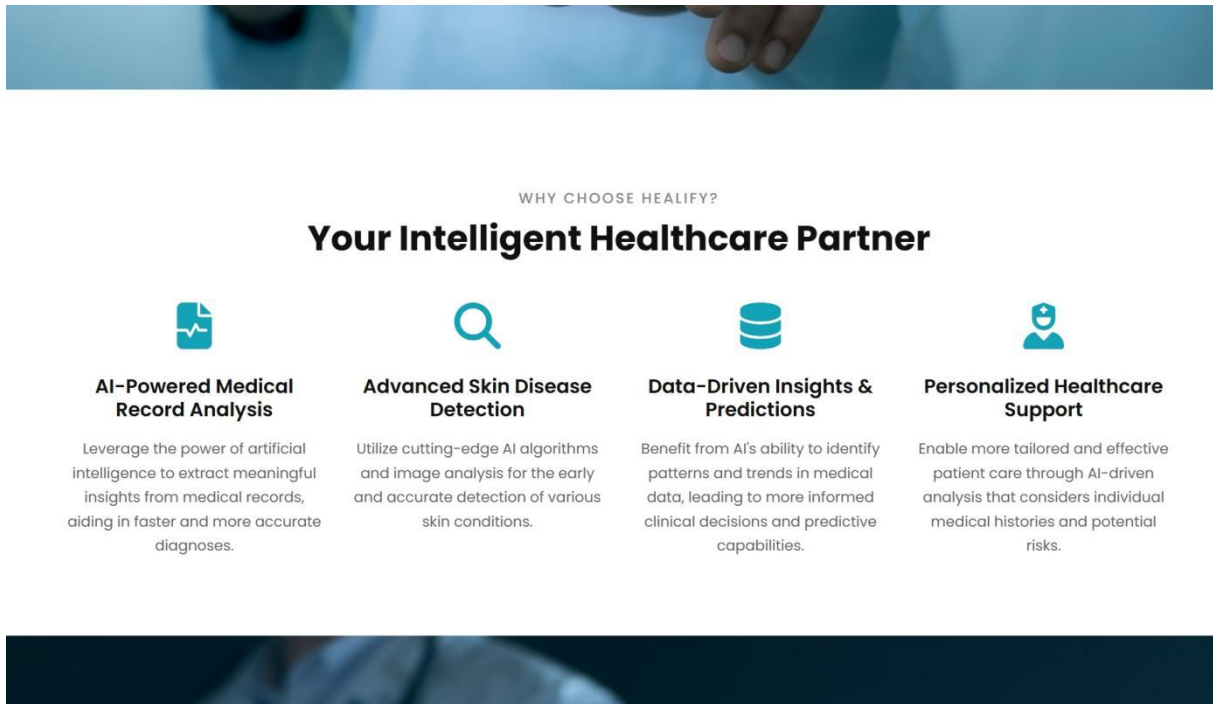


Figure 19 : Section "Why Choose Healify?"

2.3 Présentation des services

La capture d'écran ci-dessous montre la section dédiée aux services intelligents de la plateforme.

Deux fonctionnalités y sont mises en avant :

- L'analyse intelligente des dossiers médicaux, facilitant les diagnostics personnalisés.
- La détection des maladies de la peau par IA, pour une identification précoce via l'imagerie.

Chaque service est accompagné d'une illustration représentative et d'un bouton d'accès.



UNLOCK THE POWER OF PATIENT DATA

Medical Record Analysis

Our AI algorithms intelligently analyze medical records, extracting key insights, identifying patterns, and supporting faster, more informed clinical decisions. Improve diagnostic accuracy and personalize patient care.

- ✓ Routine and medical care
- ✓ Excellence in Healthcare every
- ✓ Building a healthy environment
- ✓ cumsan lacus vel facilisis.

HEALYTICS

INTELLIGENT SKIN HEALTH AT YOUR FINGERTIPS

Skin Disease Detection & Analysis

Utilizing advanced AI and image recognition, our platform assists in the early detection and analysis of skin diseases. Empowering patients and dermatologists with timely and accurate information.

- ✓ Routine and medical care
- ✓ Excellence in Healthcare every
- ✓ Building a healthy environment
- ✓ cumsan lacus vel facilisis.

SKINGUARDIAN



Figure 20 : la section des services

3. La page About

Cette page présente la vision et les valeurs de Healify de manière claire et professionnelle. Elle met en avant l'expertise combinée en médecine et IA, avec un ton engageant qui vise à inspirer confiance.

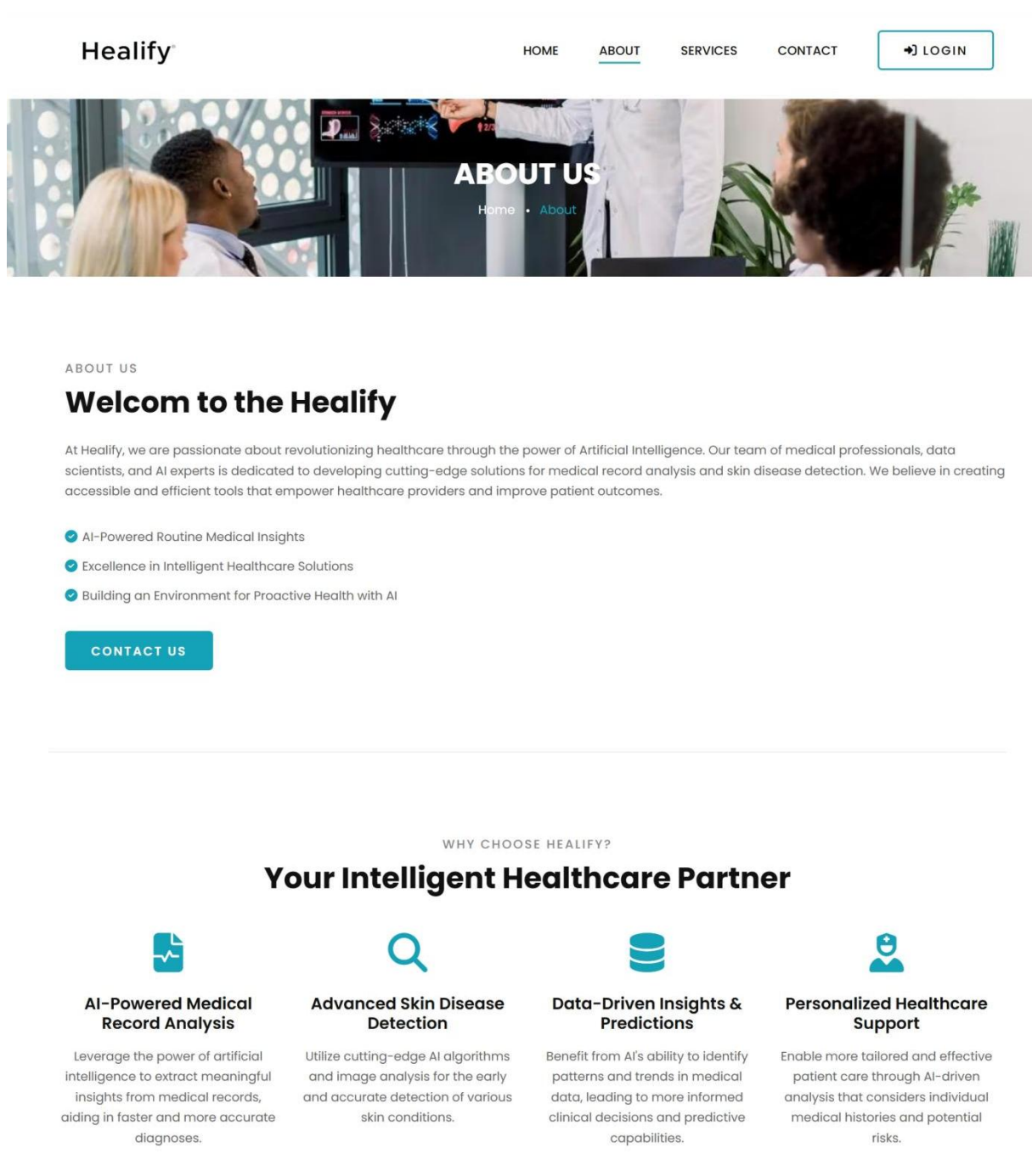
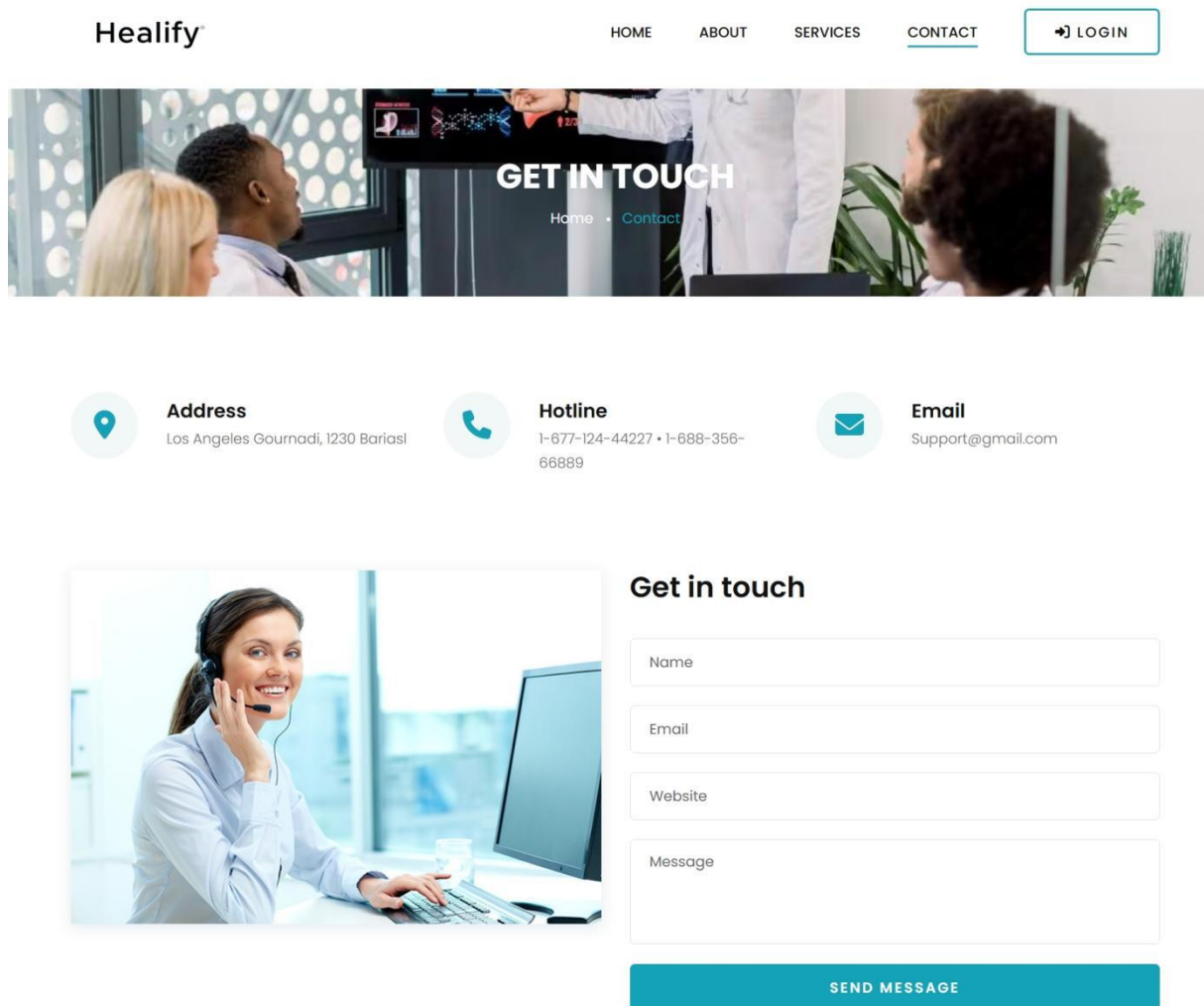


Figure 21 : La page About

4. La page Contact

Cette section permet aux utilisateurs de prendre facilement contact avec la plateforme. Elle regroupe les informations essentielles telles que l'adresse, la hotline et l'email de support. Un formulaire de contact est également disponible pour envoyer un message directement depuis le site.

L'interface est épurée, intuitive et conçue pour faciliter la communication entre les visiteurs et l'équipe de Healify.



Healify®

HOME ABOUT SERVICES CONTACT [LOGIN](#)

GET IN TOUCH
Home • Contact

Address
Los Angeles Gournadi, 1230 Bariasl

Hotline
1-677-124-44227 • 1-688-356-66889

Email
Support@gmail.com

Get in touch

Name

Email

Website

Message

SEND MESSAGE

Figure 22 : La page Contact

5. Pages d'authentification

Les pages d'authentification sont essentielles pour garantir la sécurité des données médicales des utilisateurs. Elles permettent aux patients de créer un compte sécurisé et d'accéder à leurs informations personnelles.

5.1 Page Login

- Cette page permet à l'utilisateur de se connecter rapidement et facilement à son compte. Elle présente un design clair et intuitif, avec les champs classiques 'Email' et 'Mot de passe' et un bouton de connexion bien visible.

- Un lien 'Forgot password' est proposé pour permettre aux utilisateurs de réinitialiser leur mot de passe en toute autonomie.
- Un bouton 'Create new account' est également disponible pour les nouveaux utilisateurs qui souhaitent rejoindre la plateforme.

HOME ABOUT AI HEALTH SUPPORT PAGES NEWS CONTACT SIGN UP

WELCOME BACK!

Login to Your Account

Email

Password

LOGIN

[Forgot password?](#)

CREATE NEW ACCOUNT

Figure 23 : page Login





5.2 Page Register

Cette page d'inscription est conçue pour faciliter la création d'un compte utilisateur. Les champs obligatoires sont clairement indiqués ('Nom', 'Email', 'Mot de passe', etc.).

Le bouton 'Register' est mis en évidence pour guider l'utilisateur, tandis que le lien 'Do you already have an account?' permet de revenir facilement à la page de connexion.

REGISTER NOW

Create Your Account

<input type="text" value="First Name"/>	<input type="text" value="Last Name"/>
<input type="text" value="Email"/>	<input type="text" value="Phone Number"/>
<input type="password" value="Password"/> 	<input type="password" value="Confirm Password"/> 
<input type="text" value="Address"/>	<input type="text" value="Sex"/> 
<input type="text" value="Birthdate"/> 	

REGISTER

Do you already have an account? [Login](#)

Figure 24 : page Register

6. Présentation des services

La première étape consiste à se connecter ou à s'inscrire pour accéder aux fonctionnalités du notre site.

6.1 La page healytics

Cette page permet aux utilisateurs de télécharger leurs fichiers d'analyse médicale. Le fichier d'analyse de laboratoire est directement envoyé à nos modèles affinés, qui le traitent et en extraient les informations clés. Ces modèles génèrent ensuite une explication et une interprétation claires des résultats, présentées à l'utilisateur.

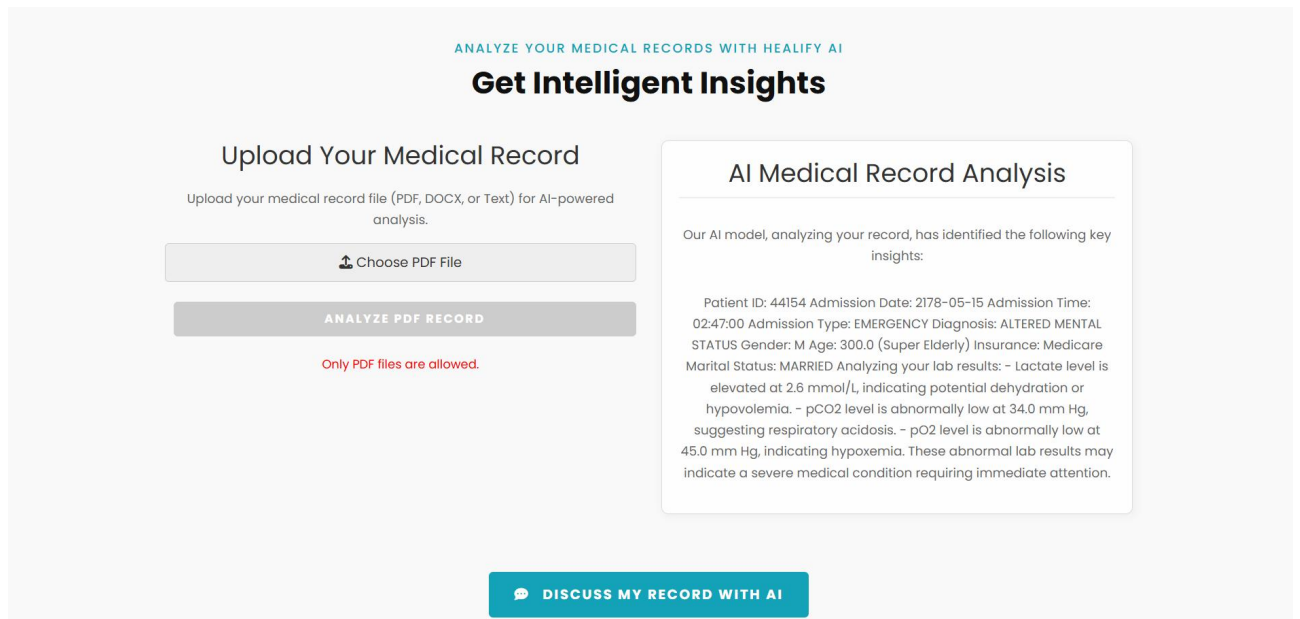


Figure 25 : Page Healytics

6.2 La page Healytics-chat

Ce chatbot répond aux requêtes des utilisateurs sur leurs dossiers médicaux en s'appuyant sur une base de données vectorielle. Les données médicales sont converties en vecteurs et stockées. La requête de l'utilisateur est également vectorisée, permettant une recherche sémantique pour identifier les informations pertinentes. Ces données sont ensuite utilisées par un modèle linguistique pour générer une réponse précise et personnalisée.

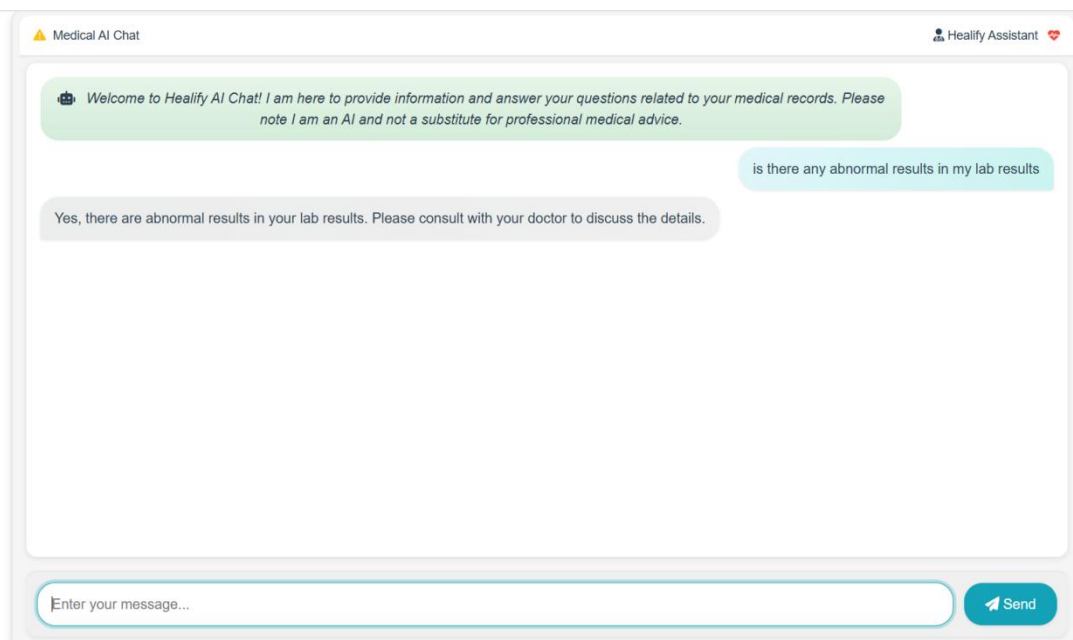


Figure 26 : Healytics-chat

6.3 La page SkinGuard

Une fois connecté, l'utilisateur accède à l'interface principale du chatbot médical. Cet écran présente la liste des conversations précédentes dans un panneau latéral ('Historique des conversations'), ainsi qu'un espace de chat central pour poser des questions textuelles.

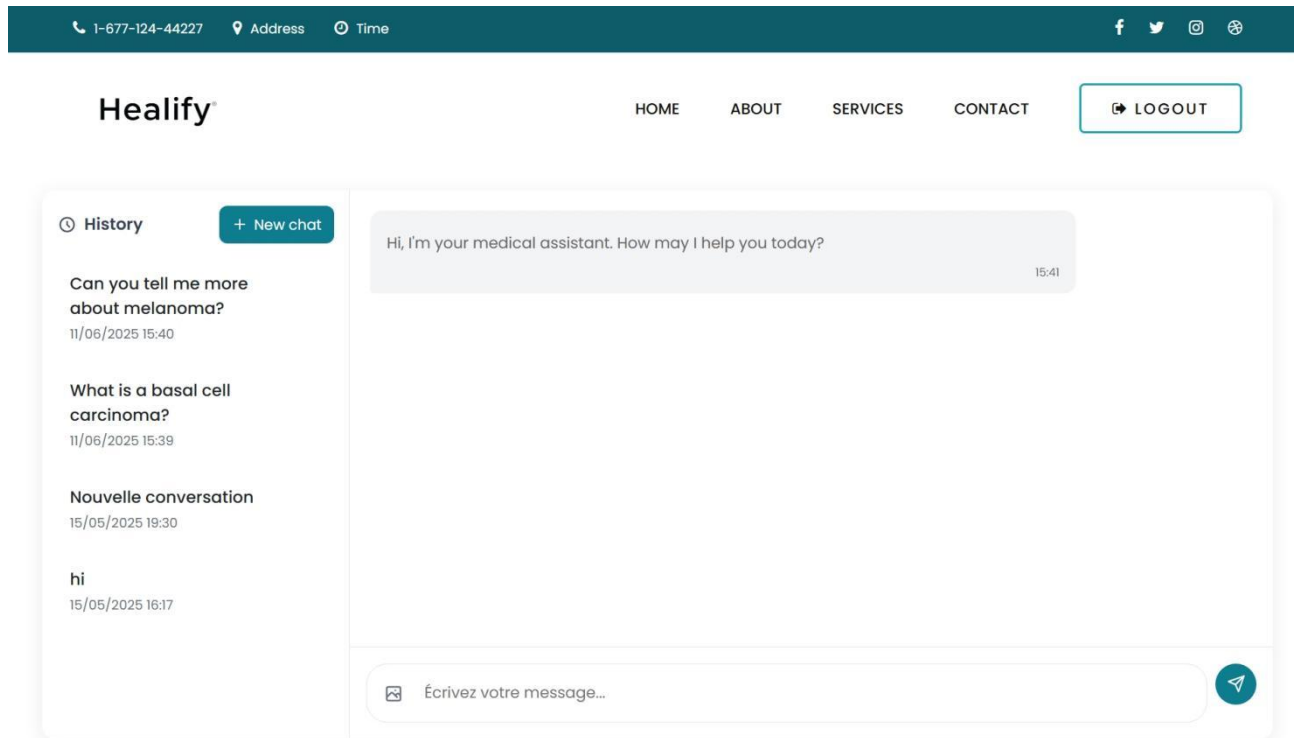


Figure 27 : medical assistant(nouvelle-conversation)

Le bouton 'New chat' permet de démarrer une nouvelle discussion, tandis que l'icône 'Supprimer' offre la possibilité de supprimer des conversations obsolètes.

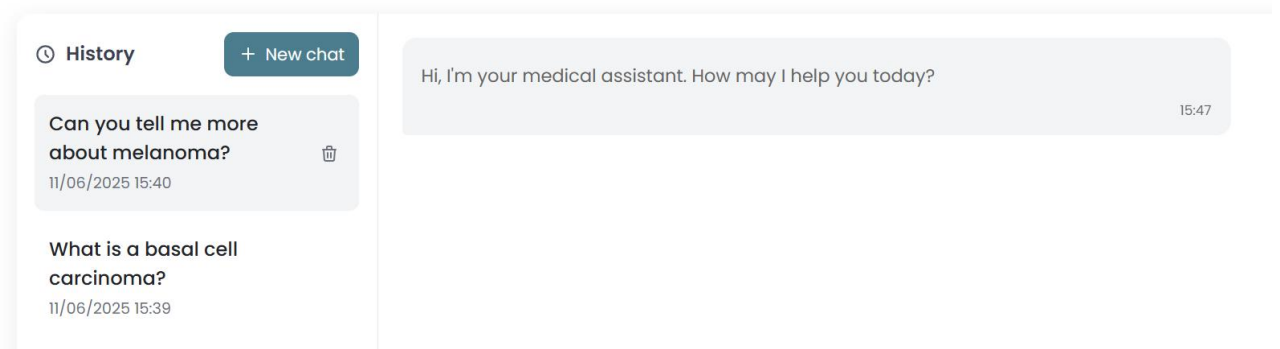


Figure 28 : medical assistant(ancienne-conversation)

➤ Interaction avec le Chatbot

L'utilisateur peut poser des questions médicales en langage naturel (ex. 'Can you tell me more about melanoma?'). Le modèle analyse la requête et fournit une réponse claire et contextualisée, accompagnée de conseils adaptés si nécessaire.

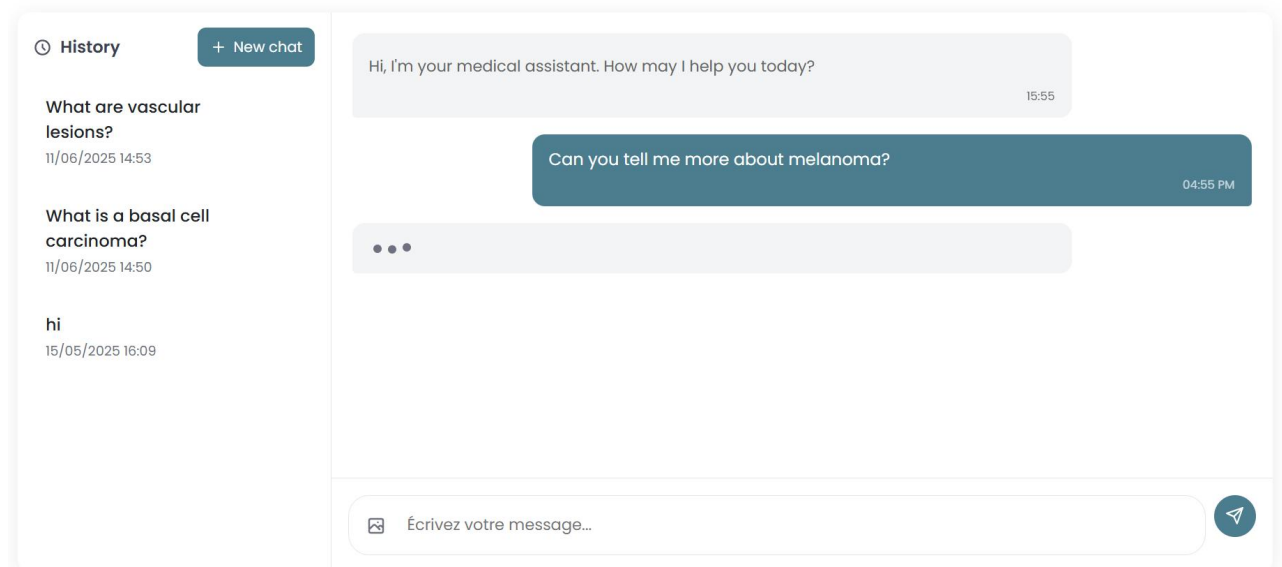


Figure 29 : Interaction textuelle avec le Chatbot (envoi de la requête)

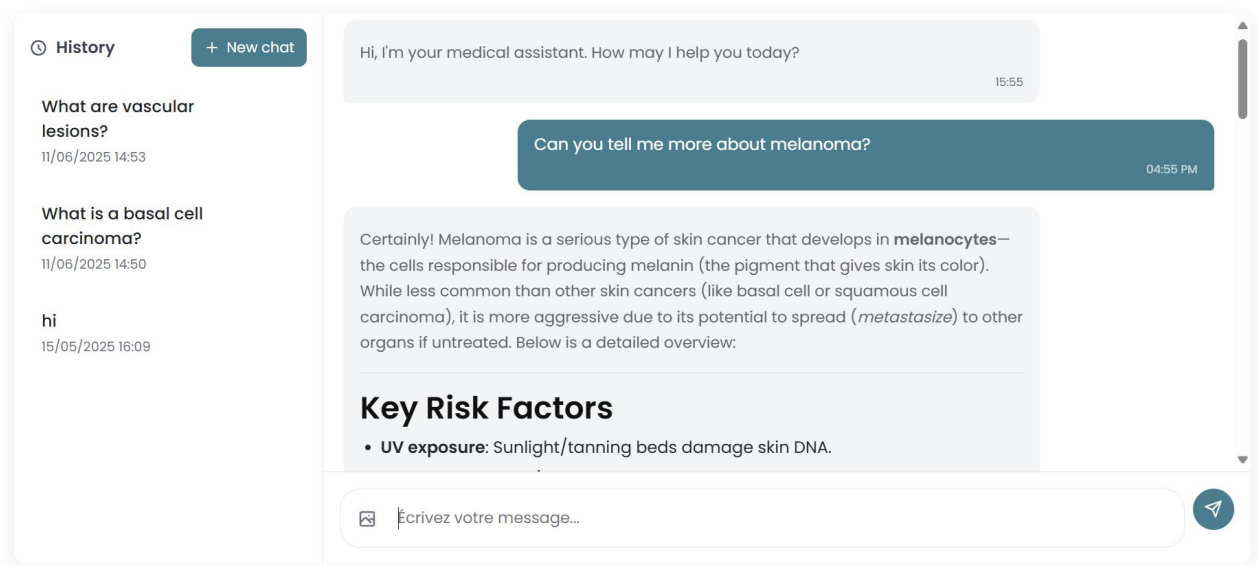


Figure 30 : Interaction textuelle avec le Chatbot (inférence)

L'utilisateur peut aussi téléverser une image médicale (ex. photo d'une lésion cutanée) pour obtenir un diagnostic automatisé du module de détection.

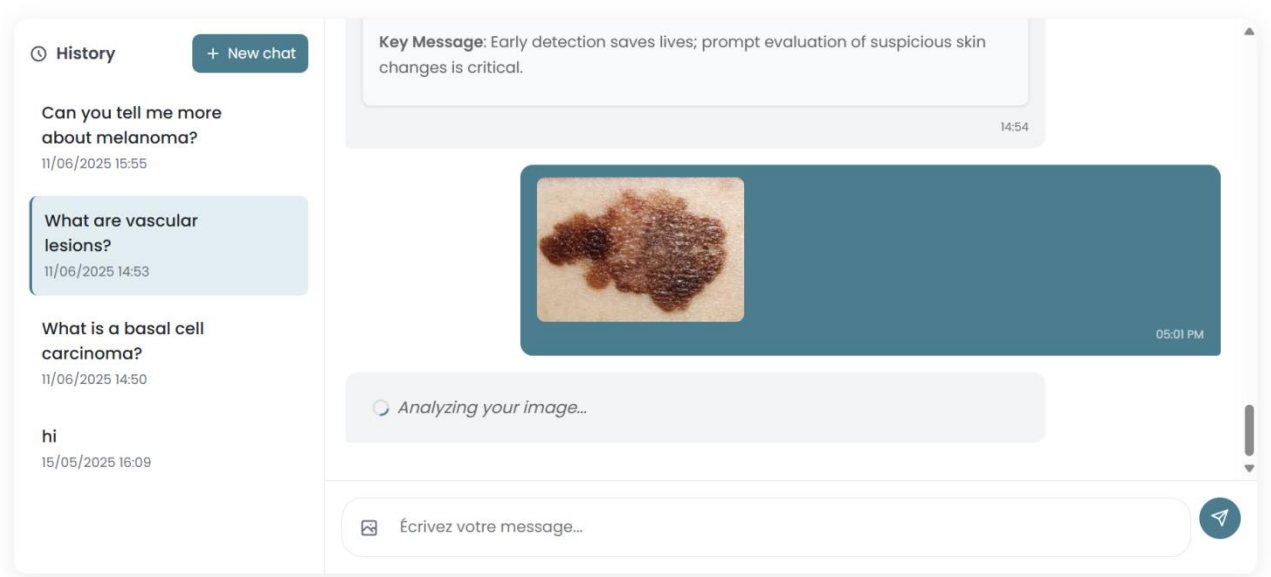


Figure 31 : Analyse d'images médicales (envoi de la requête)

Lorsque l'utilisateur envoie une image, le module de détection fournit un diagnostic précis, le type de pathologie détectée, ainsi qu'un taux de confiance. Des explications supplémentaires sur la pathologie sont également disponibles.

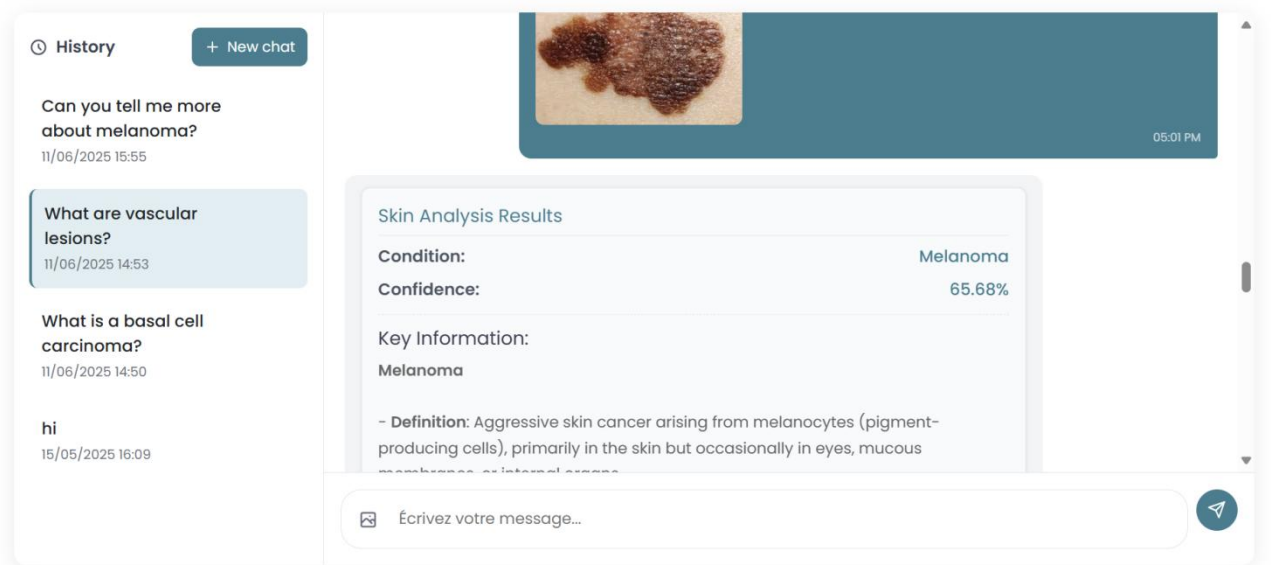


Figure 32 : Analyse d'images médicales (inférence)

7. Conclusion

Ce chapitre a concrétisé notre solution, de la modélisation à l'implémentation. Nous avons déployé une architecture robuste intégrant des modèles affinés pour l'interprétation des analyses et une base de données vectorielle pour la gestion des dossiers. Les fonctionnalités clés, comme la gestion des analyses et le chatbot, démontrent la transformation des concepts en une application fonctionnelle, fiable et sécurisée.

CONCLUSION

En somme, notre démarche a abouti à la réalisation d'une solution fonctionnelle qui pose les bases d'un système intelligent de gestion et d'interprétation des données médicales. L'intégration de modèles affinés pour l'analyse des résultats et l'emploi d'une base de données vectorielle pour une récupération sémantique des informations démontrent le potentiel de l'intelligence artificielle dans ce domaine.

Pour l'avenir, la transformation de cette solution en application mobile est une perspective clé. Cela permettrait de considérablement faciliter l'expérience utilisateur, rendant l'accès aux informations médicales plus intuitif et immédiat, directement depuis le smartphone du patient. Une telle interface favoriserait une meilleure gestion de la santé personnelle et renforcerait l'autonomie des individus face à leurs données.

Par ailleurs, l'accès à des données cliniques réelles et anonymisées constituerait un levier majeur pour le développement de solutions encore plus évolutives et précises. Avec un volume et une diversité de données accrues, nos modèles pourraient être entraînés et affinés de manière exponentielle, ouvrant la voie à des analyses plus sophistiquées, à la détection précoce de tendances et à la création d'outils d'aide à la décision encore plus performants pour les professionnels de santé. En somme, c'est en combinant l'innovation technologique et l'accès à des données enrichies que nous pourrions bâtir les solutions de santé de demain.

WEBOGRAPHIE

ⁱ **https://huggingface.co/datasets/Ihssane123/Synthetic_Medical_Notes**