

Université de Poitiers Année Universitaire 2023- 2024

U.F.R. Sciences Fondamentales et Appliquées

Master 2 TDSI – parcours Objets Connectés

Rapport de stage

Développement d'un orchestrateur d'IA

Stage effectué entre le 2/04/2024 et le 27/09/2024

Réalisé par :

BINAN Omaima

Tuteur de stage :

Mr. NAUDIN Mathieu, Mme THOMARAT Celine

Enseignant référent :

Mr. BOURDON Pascal

Lieu de stage :

2 Rue de la Milétrie, Bâtiment 7
TESLA

Remerciement

Je tiens à exprimer ma profonde gratitude à ma grand-mère, qui m'a soutenu tout au long de mes études en m'encourageant sans cesse à persévérer. J'espère que tu es fière de moi au paradis. À ma petite famille au Maroc, qui a fait l'impossible pour que je puisse être ici aujourd'hui sans manquer de rien, je vous remercie du fond du cœur.

Je remercie également mes amis de Poitiers pour leur accueil chaleureux et leur humour unique qui ont facilité mon intégration à l'université. Merci pour les moments de rire et de soutien tout au long de cette année.

Un grand merci à Madame Christine Fernandez pour son support et soutien durant mon stage, et à Monsieur Naudin, mon tuteur, pour ses encouragements, son aide précieuse à développer mes compétences, et pour m'avoir appris à être autonome et à continuer à apprendre et à m'améliorer constamment.

Enfin, j'adresse ma reconnaissance à Madame Touhani et à l'équipe académique du Master IA-RV pour avoir offert aux étudiants l'opportunité de suivre une codiplomation et de favoriser les échanges enrichissants. Je tiens aussi à remercier Monsieur Clency pour son dévouement envers la classe, ainsi que pour le temps et l'aide qu'il accorde aux étudiants.

TABLE DES MATIÈRES

1	État d’art	11
	Introduction	11
1.1	Anatomie et classification	11
1.2	Méthodes d’acquisition	13
1.3	Archivage des données	14
1.4	Intégration de l’IA dans le flux de travail médical	15
	Conclusion	15
2	Contexte et objectifs	16
	Introduction	16
2.1	Cadre du projet	16
	2.1.1 Présentation de la structure d’accueil	16
	2.1.2 Membres	17
	2.1.3 Encadrement	18
2.2	Contexte du projet	18
	2.2.1 Problématique	18
	2.2.2 Objectifs	19
2.3	Missions	19
	Conclusion	20
3	Méthodologie	21
	Introduction	21
3.1	Méthodologie de développement	21
	3.1.1 Méthode pseudo-agile	22
	3.1.2 Équipe du projet	23
	3.1.3 Adaptation du pseudo-agile	23
3.2	Planification et suivi du projet	24

3.2.1	Diagramme de Gantt	24
3.2.2	Tableau de Kanban	25
3.3	Gestion de la charge	26
3.3.1	Coût salariaux	26
3.3.2	Coût matérielle	27
3.4	Gestion des risques	28
3.4.1	Identification et évaluation des risques	28
3.4.2	Plan d'action	29
	Conclusion	29
4	Réalisation et résultats	30
	Introduction	30
4.1	Mission 1 : Segmentation automatique des données de méningiome	30
4.1.1	Préparation de données	30
4.1.2	Segmentation Manuelle	31
4.1.3	Méthode et application	32
4.1.4	Architectures proposées	32
4.1.5	Métriques d'évaluation	35
4.1.6	Configuration expérimentale	37
4.1.7	Résultat et discussion	37
4.1.8	Limites	40
4.2	Mission 2 : Développement d'un orchestrateur d'IA	42
4.2.1	Choix techniques	42
4.2.2	Étude architecturale	42
4.2.3	Méthodes	45
4.2.4	Architecture logicielle	46
	Conclusion	49

TABLE DES FIGURES

1.1	Localisation du méningiome dans le cerveau humain	11
1.2	Différentes coupes axiales pour le méningiome en séquence T1	13
1.3	Architecture du PACS [8]	14
3.1	Méthode pseudo-agile	22
3.2	Méthode pseudo-agile détaillée et appliqué sur mon projet de stage	23
3.3	Diagramme de Gantt du Projet	24
3.4	Methode Kanban appliqué dans notre projet	26
3.5	Risque rencontré dans le projet, position initiale et finale	29
4.1	Création du masque tumoral de méningiome à l'aide de 3D slicer par le radiologue à I3M.	31
4.2	Architecture d'UNet	33
4.3	Architecture UNETR	34
4.4	Distance de Hausdorff	36
4.5	Résultat de la segmentation prédit par le modèle UNet et le modèle UNETR avec la vérité terrain en vue axiale.	38
4.6	Courbe de loss pour UNet et UNETR	39
4.7	Docker Logo	42
4.8	FastAPI Logo	42
4.9	RabbitMQ Logo	42
4.10	Architecture micro-services	43
4.11	flux de travail pour le développement clinique de l'orchestrateur d'IA	45
4.12	Architecture logicielle du développement d'orchestrateur	47
4.13	Table d'évaluation et analyse des risques	55
4.14	Plan d'action pour la surveillance des risque identifiées au cours du projet	56

LISTE DES TABLEAUX

1.1	Caractéristiques des tumeurs cérébrales	12
1.2	Classification des méningiomes par grade <i>Credit : NCI-CONNECT</i>	12
3.1	Tableau des salaires des employés	26
3.2	Tableau des caractéristiques matériel	27
4.1	Tableau des patients exclus	31
4.2	Temps de calcul pour les algorithmes de segmentation	37
4.3	Analyse des performances du UNet et UNETR	40

Résumé

Mon stage porte sur le développement d'un orchestrateur d'IA, un projet exploratoire mené par le laboratoire commun I3M. Le premier cas d'usage consiste à étudier la segmentation des données IRM ultra-haute résolution (3T et 1.5T) des tumeurs cérébrales méningiomes. Le deuxième cas d'utilisation vise à automatiser ce processus de segmentation et à déployer un orchestrateur qui gère les éléments d'inférence des modèles d'IA utilisés.

L'objectif principal est de développer un orchestrateur qui coordonne et gère différents composants d'IA, tels que des modèles de *Deep Learning*, des systèmes de traitement de données et des workflows d'automatisation, afin de faciliter l'intégration, la gestion et l'optimisation des processus liés à l'IA.

Le développement de cet orchestrateur nécessite une compréhension approfondie des différentes structures et éléments du laboratoire, tels que le fonctionnement des machines, le processus d'acquisition d'images, l'archivage des données, le pré-traitement et la communication des modèles d'IA. Cette connaissance est cruciale pour concevoir un orchestrateur qui reflète efficacement l'ensemble du flux de travail et qui s'adapte à la structure du laboratoire. Il est également important de trouver des solutions pour implémenter ce projet dans une infrastructure hospitalière rigoureuse et peu flexible.

Abstract

My internship concerns the development of an AI orchestrator, an exploratory project by the I3M joint laboratory. The first phase consists in studying the segmentation of ultra-high resolution (3T and 1.5T) MRI data of meningioma brain tumors. The second phase aims to automate this segmentation process and deploy an orchestrator that manages the training elements of the models used.

The main objective is to develop an orchestrator that coordinates and manages different AI components, such as Deep Learning models, data processing systems and automation workflows, to facilitate the integration, management and optimization of AI-related processes.

The development of this orchestrator requires an in-depth understanding of the structures and elements of the laboratory, such as machine operation, image acquisition process, data archiving, pre-processing and AI model communication. This knowledge is crucial for designing an orchestrator that effectively reflects the entire workflow and adapts to the structure of the laboratory. It is also important to find solutions for implementing this project in a rigorous, inflexible hospital infrastructure.

Introduction générale

L'utilisation de l'intelligence artificielle dans le domaine médical est un développement relativement récent, mais il occupe un grand intérêt et beaucoup d'enthousiasme parmi les chercheurs. Cela est dû à la perspective de l'automatisation et de l'exécution par l'IA de tâches actuellement réalisées par des humains, ce qui pourrait entraîner des améliorations tant en précision qu'en efficacité. Ces facteurs attirent l'attention de ceux qui cherchent à améliorer les résultats cliniques, ainsi que de ceux qui s'intéressent à la recherche sur l'IA.

Cependant, l'intégration de l'IA dans les soins de santé est parsemée de difficultés. L'hétérogénéité des applications des systèmes de soins de santé, la dépendance à l'égard de logiciels propriétaires à code source fermé, ainsi que les menaces croissantes en matière de cybersécurité posent des défis importants. De plus, avant d'être déployés en milieu clinique, les modèles d'IA doivent démontrer leur efficacité dans un large éventail de scénarios et être validés par des études prospectives. Pour ce faire, il est nécessaire de les tester dans un environnement reflétant le flux de travail clinique, ce qui est difficile à réaliser sans une transformation vers une solution dédiée.

Pour faciliter cette transformation, des orchestrateurs d'IA ont émergé comme des outils sophistiqués qui englobent divers aspects de la gestion des données et de la construction des flux de travail, ainsi que de la régulation et de la supervision. Dans le cas de l'IA dans les soins de santé, il s'agit d'une plateforme logicielle spécifiquement conçue pour gérer, surveiller et automatiser le cycle de vie des applications d'IA et d'analyses avancées.

En somme, l'intégration de l'IA dans le domaine médical, facilitée par les orchestrateurs d'IA, pourrait transformer le paysage des soins de santé en améliorant l'efficacité et les résultats cliniques. Toutefois, il est essentiel d'aborder cette transformation avec soin, en tenant compte des défis techniques, réglementaires et éthiques qu'elle implique. L'avenir des soins de santé pourrait bien dépendre de notre capacité à tirer parti des avancées de l'IA tout en veillant à la sécurité et à l'éthique dans son utilisation.

le mémoire s'articule autour de quatre chapitres qui nous permettront de présenter les différents aspects de notre travail.

- **chapitre 1** : Ce chapitre est dédié au cadre d'application de ce mémoire. Il introduit les termes et concepts essentiels pour appréhender l'analyse des images IRM cérébrales.
- **Chapitre 2** : Ce chapitre est une présentation du cadre du projet, et structure

d'accueil ainsi qu'une explication des missions et l'objectif du projet de stage.

- **Chapitre 3 :** Ce chapitre décrit la méthodologie utilisée en justifiant notre choix de méthode pour développer le projet. Il aborde également d'autres aspects de gestion qui nous ont permis de mener à bien le projet et d'atteindre nos objectifs.
- **Chapitre 4 :** Ce chapitre est dédié à la présentation de la méthodologie et des résultats obtenus grâce à l'algorithme que nous avons implémenté, ainsi qu'au travail réalisé au niveau de l'orchestrateur.

Nous concluons ce travail par une conclusion générale, une conclusion spécifique au projet, notre point de vue personnel sur le projet et le stage, et enfin, les perspectives futures du projet.

Ce chapitre se propose d'explorer les caractéristiques anatomiques des méningiomes, en mettant un accent particulier sur leur classification, les méthodes d'acquisition d'images et l'intégration croissante de l'intelligence artificielle (IA) en radiologie pour améliorer le diagnostic et la prise en charge. À travers cette analyse, nous visons à fournir une compréhension approfondie à les notions théoriques et techniques qui soutien notre projet.

1.1 Anatomie et classification

Un méningiome est une tumeur qui se développe à partir des membranes qui entourent le cerveau et la moelle épinière, appelées méninges (fig1.1).

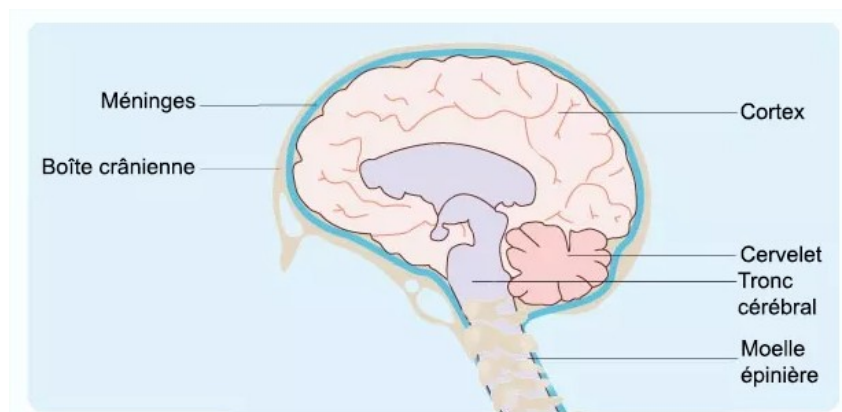


FIGURE 1.1 – Localisation du méningiome dans le cerveau humain

Le méningiome [2] est le type de tumeur le plus courant qui se forme dans la tête. La plupart des méningiomes se développent très lentement. Ils peuvent se développer pendant de nombreuses années sans provoquer de symptômes. Les méningiomes surviennent plus souvent chez les femmes. On les trouve souvent à un âge avancé, mais ils peuvent survenir à tout âge.

Caractéristiques	Détails
Sexe ratio F/H	2/1
Âge moyen	60 ans
Les facteurs de risque	— Doses élevées de rayonnement ionisant
	— Mutations génétiques : NF2, TERT
	— Iatrogène : Androcur®

TABLE 1.1 – Caractéristiques des tumeurs cérébrales

Grades

Les tumeurs primaires du système nerveux central (SNC) sont classées sur la base d'une analyse du tissu tumoral effectuée par un neuropathologiste. Les méningiomes sont classés en trois catégories en fonction de leurs caractéristiques. Chaque grade (I, II ou III) comprend différents sous-types de méningiomes.

Grade	Description
Grade I	Les méningiomes bénins de grade I sont les plus courants. Il s'agit de tumeurs de bas grade, ce qui signifie que les cellules tumorales se développent lentement.
Grade II	Les méningiomes atypiques de grade II sont des tumeurs de grade moyen. Cela signifie qu'ils ont un risque plus élevé de réapparaître après avoir été enlevés.
Grade III	Les méningiomes anaplasiques de grade III sont malins (cancéreux). Cela signifie qu'il s'agit de tumeurs à croissance rapide.

TABLE 1.2 – Classification des méningiomes par grade
Credit : NCI-CONNECT

1.2 Méthodes d'acquisition

Lorsque l'on aborde les méthodes d'acquisition de l'imagerie médicale, il y a plusieurs modalités à prendre en compte, mais dans ce contexte, nous nous concentrons spécifiquement sur l'imagerie par résonance magnétique (IRM). avec les systèmes 3 Tesla (3T) et 1,5 Tesla (1,5T) de Siemens Healthineers. Il est essentiel de comprendre les spécificités de ces systèmes d'IRM, car la modalité et l'intensité du champ magnétique peuvent influencer de manière significative la qualité de l'image et les caractéristiques des données acquises.

IRM 1.5T : est largement utilisée en milieu clinique en raison de son équilibre entre qualité d'image, temps de scan et confort du patient, offrant un rapport signal/bruit (SNR) adéquat pour la plupart des besoins cliniques. De plus, les systèmes 1,5T sont moins sensibles aux artéfacts de susceptibilité magnétique, ce qui les rend avantageux pour l'imagerie des patients avec des implants métalliques.

IRM 3T : IRM 3T, avec sa plus grande intensité de champ magnétique, offre un meilleur rapport signal/bruit (SNR) et une résolution d'image supérieure, permettant des séquences d'imagerie plus rapides et une visualisation plus claire des structures plus petites et des pathologies subtiles. Cependant, le champ plus fort peut entraîner une augmentation des artéfacts de susceptibilité et du réchauffement, nécessitant des ajustements soigneux des protocoles d'imagerie pour garantir la sécurité et une qualité optimale.

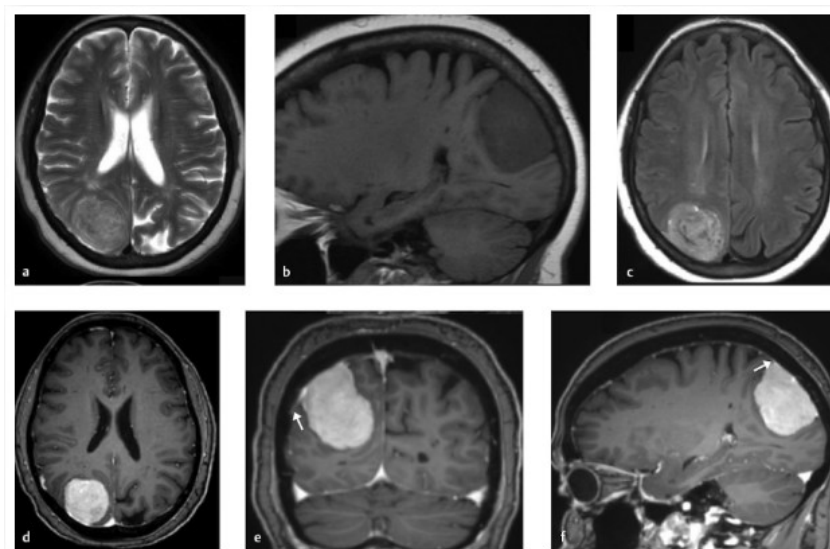


FIGURE 1.2 – Différentes coupes axiales pour le méningiome en séquence T1

1.3 Archivage des données

PACS

Tous les dossiers médicaux, y compris les images médicales, doivent être stockés et récupérés à différents moments au cours des flux de travail cliniques. La plupart des images médicales sont aujourd'hui numériques et nécessitent des formes de stockage électroniques. Le PACS (fig. 1.3) permet à la fois de stocker et d'accéder aux images médicales, y compris les tomodensitogrammes, les IRM, les ultrasons et les radiographies numériques standard.

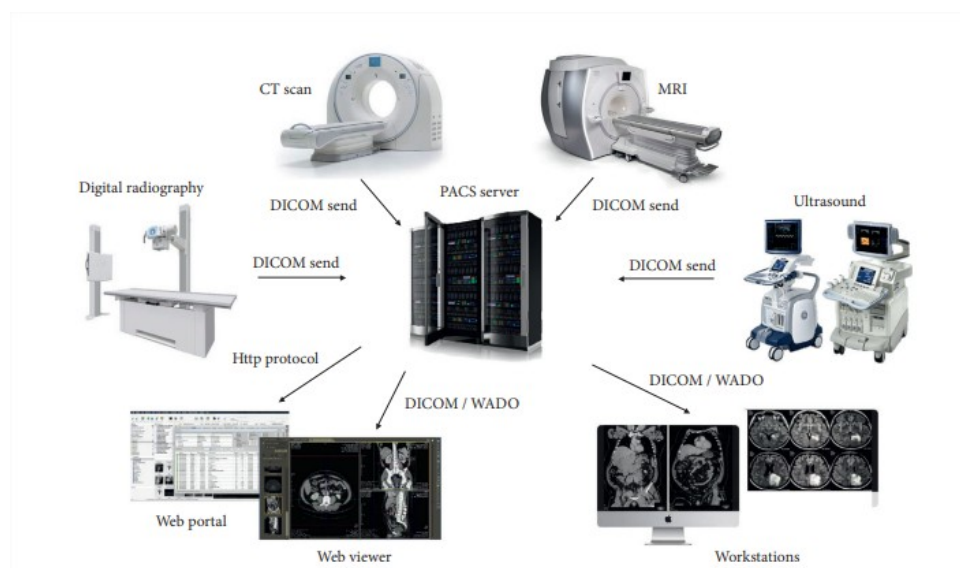


FIGURE 1.3 – Architecture du PACS [8]

En général, un serveur PACS remplit les fonctions suivantes [14] :

- Stockage
- Archivage et sauvegarde
- Récupération

Format de fichiers

Les données sur lesquelles j'ai travaillé ont été issues d'une base de données préétablie au sein de centre hôpital universitaire de Poitiers, contenant les images scanners de nombreux patients. Cette base de données a été établie au format DICOM (Digital Imaging and Communication in Medicine), est une norme pour ce qui est de la gestion informatique des données liées à l'imagerie médicale. Son intérêt principal est de permettre une standardisation des données transmises par les différents appareils médicaux [1].

Cette standardisation permet un meilleur accès aux données et une plus grande facilité pour travailler dessus, puisque les fichiers sont maintenant visibles via des logiciels

Syngovia. Ce format va contenir un champ correspondant aux voxel (qui représentent l'image), mais également d'autres champs qui vont correspondre à d'autres informations de base, telles que les dates d'examen et de naissance du patient, un identifiant pour le reconnaître ou encore le nombre d'images pour chaque série et la modalité d'examen (IRM dans mon cas).

1.4 Intégration de l'IA dans le flux de travail médical

Les algorithmes d'intelligence artificielle (IA), en particulier l'apprentissage profond, ont fait des progrès remarquables dans les tâches de reconnaissance d'images. Des méthodes allant des réseaux neuronaux convolutifs aux autoencodeurs ont trouvé une myriade d'applications dans le domaine de l'analyse d'images médicales, surtout dans la pratique de la radiologie, les médecins formés évaluaient visuellement les images médicales pour la détection, la caractérisation et l'analyse des images.

Les méthodes D'IA peuvent jouer un rôle d'assistant pour les radiologues afin d'améliorer les capacités de diagnostic et de rationaliser les opérations en automatisant les tâches répétitives, en améliorant l'analyse des images et en soutenant la prise de décision clinique.

L'intégration de l'intelligence artificielle dans le workflow médical pose effectivement plusieurs défis importants[7] telles que :

- Disponibilité et qualité des données
- Absence de Validation Technique
- Difficulté de l'intégration dans le workflow médicale
- Nécessité d'une surveillance standardisé
- Préoccupations éthiques et réglementaires

Conclusion

Ce chapitre résume les éléments clés liés à notre projet, en fournissant des explications sur les matériaux sur lesquels nous travaillons. Nous tentons également de démontrer comment les systèmes d'IA peuvent être appliqués de manière responsable pour guider les décisions cliniques surtout en radiologie.

Introduction

Dans ce premier chapitre, nous nous intéressons tout d'abord au cadre général du projet. Nous commençons par la présentation de la structure d'accueil, puis la deuxième partie est consacrée au contexte du projet : la problématique et les objectifs.

2.1 Cadre du projet

Le présent travail s'inscrit dans le cadre du projet de fin d'études en vue de l'obtention du diplôme de Master en Objets Connectés pour l'année universitaire 2023/2024.

- **Lieu du stage** : Le stage se déroulera sur la plate-forme ultra-haut champ 3T-7T du Centre Hospitalier Universitaire (CHU) de Poitiers, plus spécifiquement au Laboratoire commun Imagerie Métabolique Multinoyaux Multiorganes (I3M),
- **Durée du Stage** : Le stage a duré 6 mois, avec une période de vacances de 3 semaines à partir du 27 juillet 2024 jusqu'au 21 août 2024.
- **Financement du Stage** : Le stage est encadré et financé conjointement par le laboratoire XLIM et l'Université de Poitiers.

2.1.1 Présentation de la structure d'accueil

Le laboratoire I3M (Imagerie Métabolique Multi-noyaux Multi-organes) est un centre de recherche basé sur l'imagerie métabolique des organes, dédié à la recherche fondamentale et clinique. L'objectif du laboratoire est de mettre en œuvre des techniques innovantes d'Intelligence Artificielle pour le traitement et l'analyse automatique des images

multimodales, afin d'aider au diagnostic et au suivi thérapeutique dans les pathologies suivantes :

- Du cerveau (neuro-oncologie, psychiatrie, études de maladies neuro-dégénératives, recherche en sciences cognitives).
- Du cœur (accidents et maladies cardio-vasculaires).
- Du rein (étude du greffon avant prélèvement, suivi du greffon chez le transplanté).

2.1.2 Membres

- **Équipe DACTIM-MIS** : L'équipe Dactim-MIS (Data Analysis and Computation Through Imaging and Modeling-Maths Images Santé), intégrée au laboratoire CNRS de mathématiques et applications (LMA) 7348 de l'Université de Poitiers, est située au bâtiment 7T du CHU de Poitiers. Équipe pluridisciplinaire, elle réunit radiologues, informaticiens et mathématiciens. Ses recherches portent notamment sur la compréhension du métabolisme des organes, des phénomènes de croissance tumorale et de la réponse thérapeutique.
- **Equipe ICONÉ** : les activités de recherche de cette équipe sont organisées autour de la modélisation et le traitement des informations couleurs et spectrales des images et des vidéos. Les enjeux adressés sont liés aux questions théoriques et aux applications dans le domaine du patrimoine, de l'industrie et du multimédia au sens large. Les enjeux scientifiques s'articulent autour de la mesure et les outils d'analyse pour images multivariées, les mesures composites à partir d'images multivaluées, la modélisation des mesures multivaluées pour la décision et l'imagerie médicale.
- **Centre Hospitalier Universitaire de Poitiers** : Après la mise en service fin 2011 de l'IRM 3 Tesla, le CHU de Poitiers s'est équipé en octobre 2019 d'une IRM 7 Tesla. Cette dernière a été mise en service en novembre 2019 dans un bâtiment dédié, accolé à Jean-Bernard. Poitiers est le premier établissement de France à utiliser cet équipement à des fins cliniques.
- **La société Siemens Healthineers** : Leader technologique dans le secteur médical, l'entreprise entend poursuivre sur la voie de l'innovation afin de façonner la santé de demain.

2.1.3 Encadrement

Mathieu NAUDIN : Ingénieur chercheur, son domaine d'expertise se concentre principalement sur l'imagerie par résonance magnétique (IRM) et son application dans le domaine médical et IA.

Celine THOMARAT : Ingénieure en IA et imagerie médicale, elle jouera également un rôle d'encadrante dans le développement de l'orchestrateur d'IA.

2.2 Contexte du projet

Notre projet vise à développer un orchestrateur d'IA. Cet outil a pour objectif de gérer de manière automatisée et efficace différents projets d'intelligence artificielle développés au sein du laboratoire I3M, en intégrant divers modèles dans un flux de travail cohérent. Le projet comporte deux volets principaux :

- **Mission 1** : La segmentation automatique de tumeurs cérébrales de type méningiome par IA.
- **Mission 2** : Le développement d'un orchestrateur d'IA.

2.2.1 Problématique

La segmentation des tumeurs représente un défi majeur en neuro-imagerie, en raison de la complexité et de la variabilité des tumeurs, ainsi que de la nécessité d'une précision élevée pour un diagnostic et un suivi fiables, le processus manuel de segmentation est non seulement chronophage, mais il est également sujet à des erreurs, limitant ainsi son utilisation dans les routines cliniques quotidiennes. L'intégration de modèles d'apprentissage profond pour automatiser cette tâche offre une solution potentielle, mais soulève également des questions cruciales :

- Comment garantir que ces modèles maintiennent une précision et une fiabilité comparables à celles des experts humains ?
- Comment assurer une gestion efficace de ces modèles d'IA, en particulier lorsqu'ils sont intégrés dans un orchestrateur conçu pour créer des *workflows* automatisés et optimisés ?
- comment peut-on harmoniser l'interaction entre les radiologues et ces systèmes automatisés pour maximiser la valeur clinique des résultats générés ?

Ces questions soulignent la nécessité d'un cadre robuste qui non seulement améliore la segmentation des tumeurs, mais qui intègre également ces modèles dans un flux de travail clinique fluide, scalable, et adaptable aux besoins évolutifs du domaine médical.

2.2.2 Objectifs

Le projet vise principalement à approfondir la compréhension du flux de travail des modèles d'IA utilisés au sein du laboratoire, ainsi que les besoins et les éléments nécessaires pour les intégrer efficacement dans un workflow structuré et orchestré. Ce workflow se concentrera sur la gestion des données et des algorithmes d'IA développés par le laboratoire, dans le but d'optimiser la gestion et l'utilisation de multiples modèles d'IA, tout en simplifiant leur maintenance et leur déploiement.

- ▷ Assurer une intégration transparente des différents modèles d'IA développés par le laboratoire I3M dans un système unifié accessible et utilisable par les radiologues et les ingénieurs.
- ▷ Rationaliser le flux de travail entre les radiologues et les ingénieurs afin de réduire le temps consacré aux tâches répétitives.

Dans le but d'améliorer la productivité et de favoriser l'innovation dans le domaine de l'IA au laboratoire I3M.[7].

2.3 Missions

Mission 1 : Segmentation automatique des données de méningiome

La mission consiste à développer et mettre en œuvre une solution automatisée pour la segmentation des données de méningiome, une forme de tumeur cérébrale. Les étapes et objectifs principaux de cette mission sont les suivants :

- **Acquisition des données :**
 - Rassembler les données pertinentes associées aux méningiomes (image IRM).
 - Assurer que les données collectées sont de haute qualité et la bonne extension pour permettre un apprentissage efficace des modèles d'IA.
- **Pré-traitement des données :**
 - Nettoyer et normaliser les données pour qu'elles soient prêtes à être utilisées dans le processus de segmentation.
 - Éventuellement, appliquer des techniques de data augmentation pour améliorer la diversité des données d'entraînement.
- **Développement du modèle de segmentation :**
 - Créer ou adapter un modèle d'IA spécialisé dans la segmentation des images médicales.

- Entraîner ce modèle sur les données pré-traitées afin qu'il puisse identifier et segmenter les zones pertinentes des images .

- **Évaluation et optimisation du modèle :**

- Tester le modèle sur un ensemble de données de validation pour évaluer sa performance en termes de précision, de sensibilité, de spécificité, etc.
- Affiner le modèle en ajustant les hyperparamètres.

Mission 2 : Développement d'un orchestrateur d'IA

Une fois que les modèles d'IA sont développés, nous les intégrons dans un système capable de centraliser et d'automatiser les processus de traitement des données, et gérer l'inférence de différents modèles d'IA.

- **Routage des données** : Les données provenant des IRM et d'autres sources sont reçues, pré-traitées, et ensuite routées à travers l'orchestrateur pour être acheminées vers les destinations appropriées.
- **Gestion des modèles d'IA** Les utilisateurs doivent simplement définir le modèle, spécifier les fonctions supplémentaires (par exemple, la fonction de perte, l'optimiseur et les métriques utilisées pendant l'entraînement). Une fois les modèles définis, ils peuvent être entraînés, évalués et des inférences peuvent en être tirées.
- **Moteur d'orchestration** : Mettre en place une orchestration dynamique qui déploie et exécute des microservices, éventuellement avec des conteneurs Docker, pour des tâches spécifiques (ex. traitement de fichiers NIFTI, analyse via des algorithmes d'IA)
- **Gestion de l'inférence** : Disposition d'une infrastructure robuste pour exécuter les modèles d'IA en production.
- **Gestion de ressources de calcul** : surveillance et optimisation d'utilisation des ressources de calcul, en ajustant dynamiquement l'allocation des ressources en fonction de la charge de travail et des besoins des modèles.

Conclusion

En conclusion ce chapitre a présenté le cadre du projet, ses objectifs, et les missions principales, notamment la segmentation des tumeurs cérébrales et le développement d'un orchestrateur d'IA. En définissant le contexte et les enjeux, nous avons posé les bases pour les développements méthodologiques à venir, en soulignant l'importance de l'innovation pour améliorer les pratiques en imagerie médicale.

Introduction

Dans cette section, nous aborderons mon interaction avec les différents individus impliqués dans le projet et explorerons l'utilisation de la gestion de projet agile en utilisant le cadre Scrum. Ensuite, nous examinerons la méthode agile que nous avons adoptée, en expliquant pourquoi et comment nous l'avons appliquée dans notre projet. Enfin, nous identifierons les risques évalués à travers les différentes phases de notre projet.

3.1 Méthodologie de développement

Lorsqu'il s'agit de choisir la méthode de développement appropriée parmi les nombreuses qui existent, il est essentiel de prendre en compte la nature et la taille du projet. Dans notre cas, étant donné que nous travaillons sur un projet de recherche exploratoire, il est crucial de bien comprendre les caractéristiques spécifiques du projet et de définir les défis associés. Cela nous permettra de choisir la méthodologie la plus adaptée pour mener à bien le développement du projet.

Les projets d'exploration sont des objets de gestion atypiques. Ils sont souvent porteurs d'innovation et ne font pas nécessairement consensus quant à leur pertinence pour l'organisation. En effet, il n'y a pas toujours de client clairement identifié ou de demande explicite de la part de celui-ci. Le résultat attendu d'un tel projet est également difficile à définir, pouvant aller d'un prototype à d'autres formes de livrables. De plus, l'horizon temporel de ces projets est complexe à appréhender, car il englobe à la fois des enjeux à long terme, à moyen terme, et parfois à court terme.

Compte tenu de ces cinq caractéristiques, on comprend que ce type de projet pose des

défis spécifiques en termes de gestion. Il devient donc nécessaire d'établir une gestion de projet flexible, combinant des méthodes agiles et des approches de gestion traditionnelles, afin de structurer et de lancer le projet de manière efficace.

3.1.1 Méthode pseudo-agile

Notre projet a adopté une approche pseudo-agile (fig3.1), qui ne nécessite pas de respecter strictement toutes les pratiques ou principes de l'Agilité. Bien que nous ne soyons pas entièrement engagés envers les principes fondamentaux de l'Agilité, nous intégrons plusieurs aspects clés, tels que la tenue de réunions hebdomadaires, la collaboration au sein de l'équipe, et l'amélioration continue en adaptant des pratiques agiles flexibles qui correspondent aux besoins de gestion de notre projet.

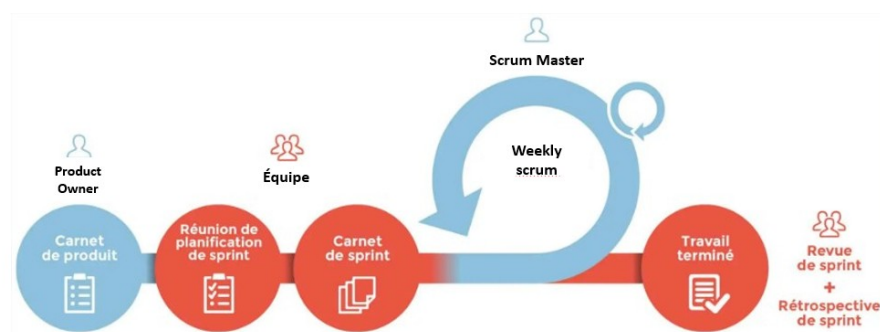


FIGURE 3.1 – Méthode pseudo-agile

Nous avons choisi cette méthode car elle nous permet de maintenir un certain niveau de structure tout en offrant la flexibilité nécessaire pour un projet de recherche exploratoire. L'approche pseudo-agile nous permet d'expérimenter et d'itérer sans être contraints par les aspects plus rigides des cadres agiles traditionnels, qui pourraient ne pas convenir à la nature incertaine et évolutive de nos objectifs de recherche. De plus, cette approche nous permet d'adapter nos processus aux exigences spécifiques de notre projet, en nous assurant de pouvoir réagir rapidement aux nouvelles perspectives et développements au fur et à mesure qu'ils se présentent.

3.1.2 Équipe du projet

- **Product owner** : Le laboratoire I3M, qui a besoin de l'outil, est responsable de définir les exigences du projet et de prioriser les détails.
- **Scrum Master** : Mathieu Naudin, responsable du bon déroulement du projet et du respect des processus Scrum.
- **Equipe scrum** : Omaira Binan et Céline Thomarat travaillent sur la création du produit sont responsable aussi de livraison.

3.1.3 Adaptation du pseudo-agile

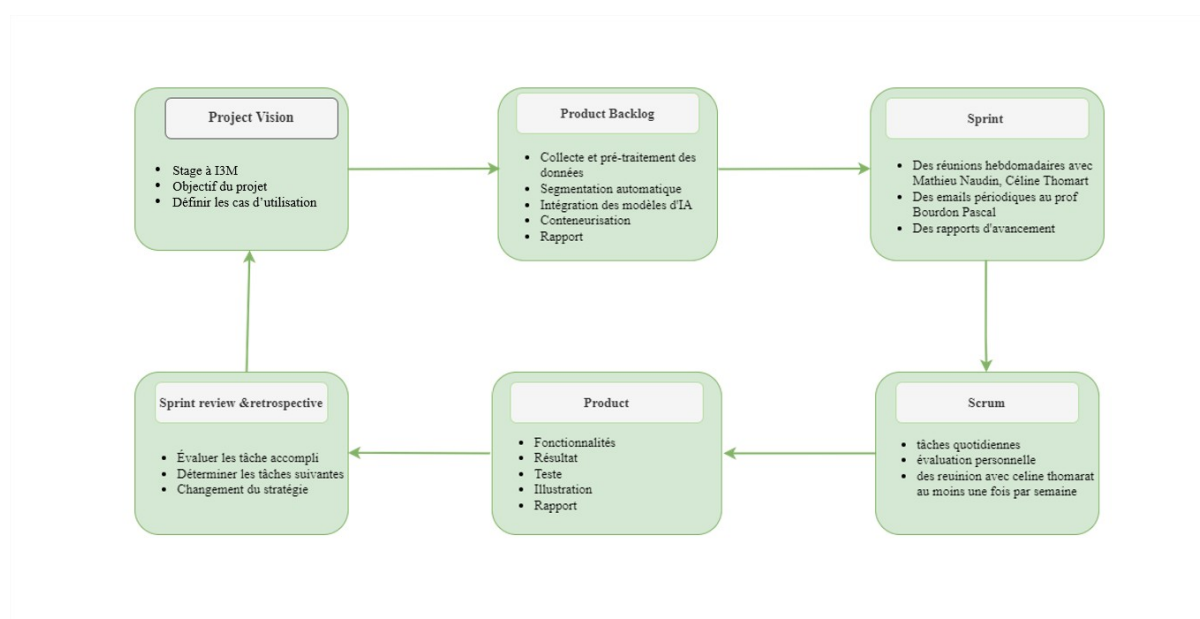


FIGURE 3.2 – Méthode pseudo-agile détaillée et appliqué sur mon projet de stage

Project vision

L'approche pseudo-agile que nous avons adoptée reflète les principes de Scrum (fig3.2). La vision et l'initiation du projet ont été définies lors d'une réunion approfondie avec l'équipe DACTIM du laboratoire I3M, en particulier avec Mathieu Naudin et Céline Thomarat. Cette réunion a permis de clarifier les objectifs, d'identifier les contraintes techniques à respecter, de définir la stratégie à adopter, et de planifier le sprint pour la semaine suivante. En parallèle, une communication continue a été maintenue par des échanges réguliers par e-mail et des rapports de suivi envoyés toutes les trois semaines à mon enseignant référent, le Professeur Bourdon Pascale, pour rendre compte de l'avancement du projet.

Sprint Review et Retrospective

Notre approche de sprint était basée sur des retours fréquents de la part de mes deux tuteurs, qui ont joué un rôle déterminant dans l'évolution du projet. Les discussions avec Mathieu et Céline ont permis d'améliorer nos résultats d'entraînement ainsi que le développement de l'orchestrateur, en appliquant de nouvelles techniques de pré-traitement, en optant pour d'autres choix techniques afin de mieux contrôler l'entraînement, et en définissant l'architecture de l'orchestrateur, ainsi que ses limites en termes d'application hospitalière. Ces échanges ont également amélioré l'interprétation des résultats. Des réunions hebdomadaire régulières étaient planifiées pour discuter de l'avancement du projet, des difficultés rencontrées, des résultats obtenus et pour adapter les objectifs si nécessaire.

3.2 Planification et suivi du projet

3.2.1 Diagramme de Gantt

Le diagramme de Gantt offre une vue d'ensemble claire et organisée de la planification des tâches de notre projet répartie sur une période allant du 2 avril jusqu'au 27 septembre 2024, ce qui nous permet de suivre facilement les progrès, d'identifier les dépendances entre les tâches et de gérer efficacement les ressources et les échéances en respectant les jalons définis pour chaque tâche.

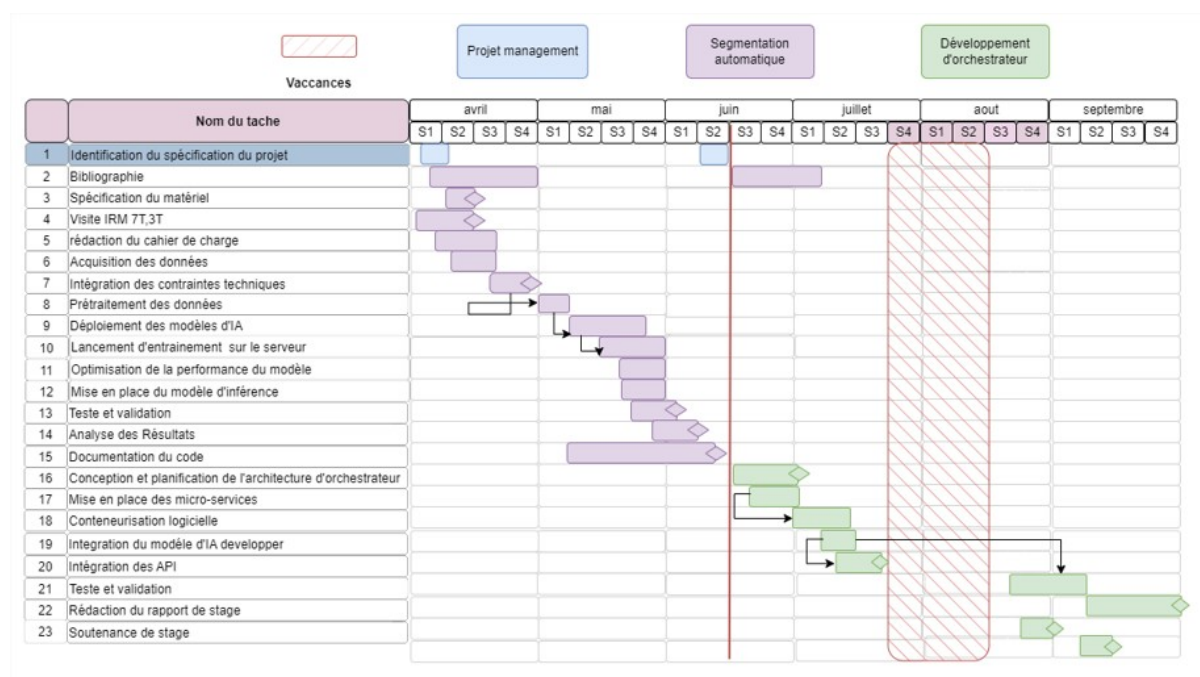


FIGURE 3.3 – Diagramme de Gantt du Projet

le diagramme (fig 3.3) trouver dans l'annexe permet de distinguer deux principaux cas d'utilisation et les sous tâches de chaque cas d'utilisation.

- **Développement d'un modèle de segmentation automatique des données de méningiomes** : Avant de commencer cette mission, une bonne compréhension du sujet, des tumeurs, et des types de solutions présentées dans l'état de l'art est nécessaire pour pouvoir progresser correctement dans les autres tâches telles que le traitement des données, la construction du modèle d'IA, et l'optimisation de l'entraînement du modèle.
- **Développement de l'orchestrateur** : Cette partie constitue l'élément principal du projet. Elle se concentre sur la mise en place d'un orchestrateur pour gérer et coordonner différents modèles d'IA. Cela inclut l'intégration du modèle que nous avons développé dans le pipeline de l'orchestrateur, l'automatisation des flux de travail, intégration de l'architecture micro-services.

Chaque tâche est minutieusement estimée en termes de délais pour permettre une planification réaliste et une gestion efficace des ressources. Cela inclut l'identification des dépendances entre les tâches et la mise en place de points de contrôle réguliers pour évaluer l'avancement du projet.

3.2.2 Tableau de Kanban

La méthode Kanban (fig3.4) nous a permis de visualiser et de gérer les tâches en temps réel, en nous assurant que les activités prioritaires étaient toujours réalisées. Cette combinaison de pseudo agile et kanban nous a offert la structure et la visualisation globale du projet, et la capacité d'adaptation. Chaque semaine, nous réévaluons les priorités en fonction des avancées et des retours reçus, tout en maintenant un suivi constant avec les parties prenantes grâce aux réunions et aux mises à jour via le tableau Kanban.

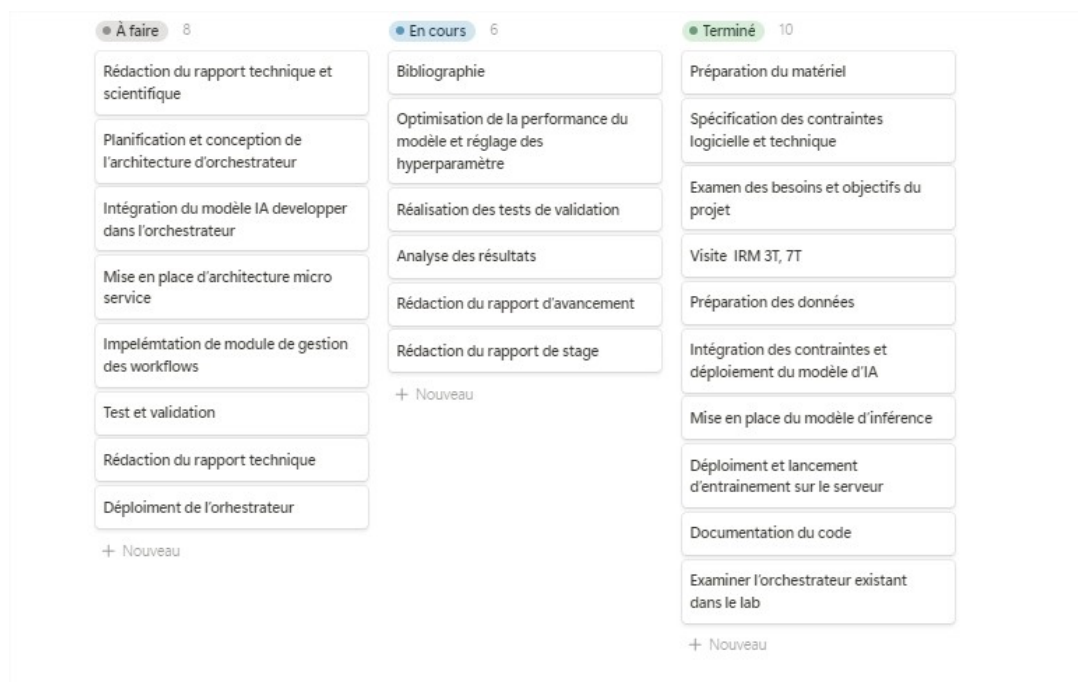


FIGURE 3.4 – Methode Kanban appliqué dans notre projet

3.3 Gestion de la charge

3.3.1 Coût salariaux

Estimation des coûts : En ce qui concerne les coûts associés à ce projet, il est important de prendre en compte les différents niveaux de rémunération du personnel impliqué.

Employé	Salaire Brut (en €)	Salaire par Heure (en €)	Durée de Travail (h)
Stagiaire	609	4,35	140
Ingénieur hospitalier	2 550	18,21	140

TABLE 3.1 – Tableau des salaires des employés

- **Salaire d'un ingénieur :** Selon les données disponibles sur le site de l'emploi collectif, le salaire brut pour un ingénieur principal d'études sanitaires (échelon 5) varie entre 2 550 € et 3 396 € par mois sur une période de trois ans.
- **Salaire d'un stagiaire :** Pour un stagiaire, le taux horaire de la gratification est égal au minimum à 4,35 € par heure de stage, correspondant à 15% du plafond horaire de la sécurité sociale (soit $29 \text{ €} \times 0,15$).

Pour une estimation plus précise des coûts, il est important de considérer que le salaire de base pour l'ingénieur hospitalier peut évoluer vers 3 396 € en fonction de l'expérience

et des échelons atteints au fil des années.

3.3.2 Coût matérielle

Après avoir effectué une analyse technique et fonctionnelle des exigences du projet, le dispositif matériel a été défini. Le tableau 3.2 ci-dessous présente les caractéristiques du matériel disponible sur place ainsi que celui requis pour mener à bien le projet.

Matériel	Description	Caractéristiques	Coût (en €)
Ordinateur poste bureau	Dell N°CH6604, pour le développement et les tests de logiciels.	Carte GPU : 5 Go, Mémoire RAM : 64 Go, Processeur Intel Xeon	4500
Serveur Nvidia A100	Prise en charge de l'inférence et de l'entraînement en IA, ainsi que du calcul haute performance	40G mémoire HBM2, Tencor core de 3ème génération	8000

TABLE 3.2 – Tableau des caractéristiques matériel

Le prix d'un GPU NVIDIA A100 peut varier selon la configuration, les accords de volume et l'unité de calcul TB/s . Cependant, pour une estimation générale, un seul GPU NVIDIA A100 peut coûter entre 8 000 et 10 000 euros.

Calcul de l'utilisation en terme de capacité

Supposons que la "capacité" ici se réfère au pourcentage de temps pendant lequel le serveur est utilisé chaque jour :

Le pourcentage d'utilisation quotidienne du serveur est calculé comme suit :

$$\frac{10 \text{ heures}}{24 \text{ heures}} \times 100 = 41,67\%$$

Cela signifie que le serveur utilise 25% de sa capacité quotidienne.

Estimation du coût basé sur l'utilisation

Le coût total du serveur est toujours de 8 000 euros pour une utilisation de 10 heures par jour :

$$41,67\% \times 8000 \text{ euros} = 3333,60 \text{ euros par an}$$

3.4 Gestion des risques

3.4.1 Identification et évaluation des risques

Pour notre projet, l'identification des risques prend une place au début de la phase préparatoire, aussi nous avons établi une analyse des risques projet qualitative tout au long du cycle de vie du projet pour être capable de s'adapter à des situations difficiles dans un temps réel. Il s'agit d'évaluer la sévérité potentielle de chaque risque sur l'atteinte des objectifs du projet, puis de déterminer un ordre de priorité pour l'analyse approfondie et les actions ultérieures, en présentant un processus pour la gestion de risque qui englobe :

- Identification des risques
- Évaluation et analyse des risques
- Plan d'action

La table de risque prévisionnel dans la figure **Fig 4.13** est une solution adoptée afin d'identifier, évaluer et analyser les principaux risques potentiels liés au développement d'orchestrateur et l'intégration d'un modèle d'apprentissage automatique pour la segmentation des méningiomes. Chaque risque est catégorisé selon son impact, sa nature, sa probabilité, sa gravité et sa criticité, et des plans d'action correctifs sont proposés pour atténuer ces risques.

Niveau de criticité est obtenu par :

$$\text{Niveau de criticité} = \text{code Impact} \times \text{code Probabilité}$$

Risque N°5

Lors du développement de l'orchestrateur, nous avons rencontré un risque majeur lié aux limitations des API utilisées, en particulier leur incapacité à communiquer correctement, à cause de l'environnement hospitalier qui impose des contraintes rigides sur l'accès aux systèmes et aux données en raison des exigences de sécurité et de confidentialité des informations de santé.

En raison de ces restrictions, les API de notre orchestrateur ont eu du mal à s'intégrer avec les systèmes hospitaliers existants, ce qui a affecté la fluidité et l'efficacité des échanges de données nécessaires pour le bon fonctionnement de l'orchestrateur.

Les conséquences incluent des retards dans l'exécution d'orchestrateur, ainsi il a menacé la fiabilité des processus automatisés que nous souhaitons mettre en place

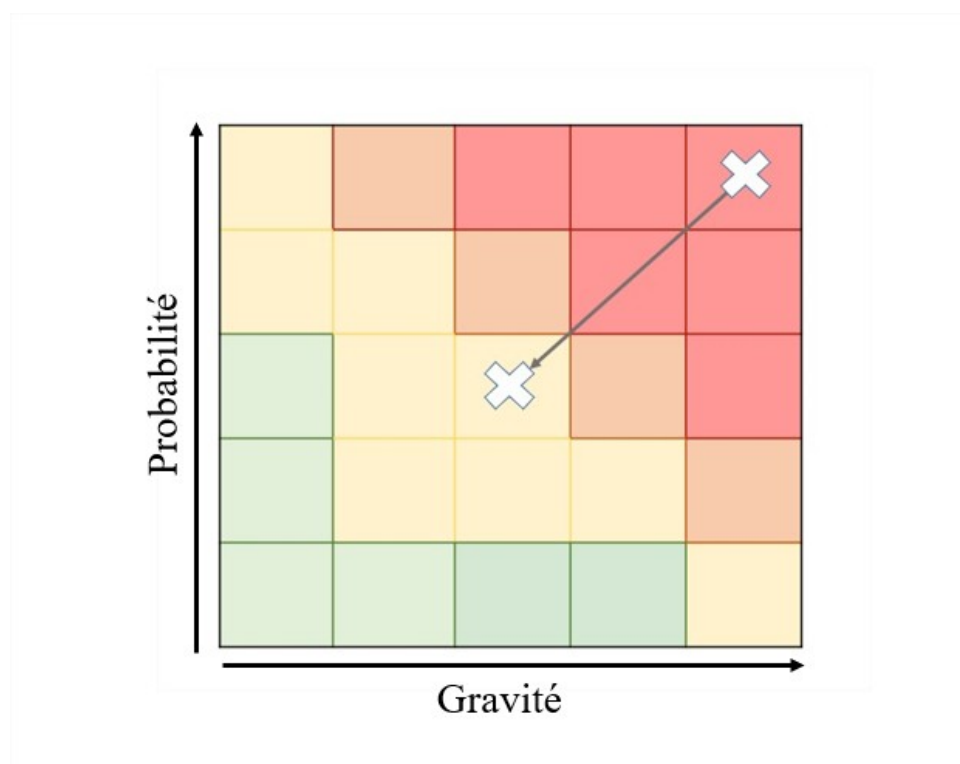


FIGURE 3.5 – Risque rencontré dans le projet, position initiale et finale

Pour surmonter ce défi (fig 3.5), il a été nécessaire de travailler étroitement avec l'équipe informatique de l'hôpital pour ajuster les paramètres d'accès tout en respectant les normes de sécurité, aussi essayer d'adopter les bons choix technique flexible et facile à s'installer en prend en considération les périmètres de projet.

3.4.2 Plan d'action

cette étape de gestion des risques nous avons mis en œuvre les plans de réponses qui décrit les mesures spécifiques à prendre pour atténuer, gérer ou éliminer les risques identifiés fig (4.1) trouver dans l'annexe. Il inclut les étapes nécessaires les personnes responsables, la méthode de surveillance, l'état de chaque risque après quand nous avons appliqué différents solutions pour remonter le risque.

Conclusion

Ce chapitre résume les différents étapes de la démarche de gestion de projet mise en œuvre. Nous y aborderons la méthode pseudo-agile, ainsi que l'organisation et la gestion des délais pour les différentes tâches. Enfin, nous détaillerons la gestion des risques pour tous les éléments susceptibles de perturber la réalisation du projet.

Introduction

Ce dernier chapitre présente le processus de développement de modèles de segmentation automatique des tumeurs méningiomes à l'aide d'algorithmes d'apprentissage profond, et explique comment cette étude est la base utiliser pour développer un orchestrateur intégrant l'ensemble du workflow d'IA, permettant une gestion efficace des modèles existants et des applications en temps réel en radiologie.

4.1 Mission 1 : Segmentation automatique des données de méningiome

4.1.1 Préparation de données

Parmi 332 patients opérés entre mars 2018 et juillet 2023, nous avons analysé un ensemble de données provenant de 285 volumes d'IRM pondérés en T1 Gado acquises à l'aide de différente machine, notamment des appareils de healthcare et siemens, à différent champs magnétiques sont 1.5T et 3T. Ces données sont issues de patients dont le diagnostic de méningiome a été confirmé confirmés par des examens radiologiques et histopathologiques.

Les données incluent 234 patients appartenant au grade I, 51 patients appartenant aux grades II et III. Les 47 patients restants ont été exclus pour les raisons suivantes :

Nombre de patients	Exclusion
21	Patient sans images dans le Pacs
14	IRM post-opératoire
5	Sans séquence T1 injecté
7	Localisation extra crânienne

TABLE 4.1 – Tableau des patients exclus

Les examens ont été téléchargés au format DICOM et stockés dans un environnement institutionnellement protégé. Des séquences T1 rehaussées par contraste (avec au moins plus de 100 coupes dans le plan axial) ont été identifiées à partir de notre cohorte finale, puis converties au format NIfTI afin d’assurer la dé-identification des patients et de disposer d’un format compatible et adaptable à l’entraînement de modèles d’apprentissage profond.

4.1.2 Segmentation Manuelle

Les images IRM des méningiomes sont segmentées manuellement par un radiologue spécialisé utilisant le logiciel 3D Slicer [5], qui offre des outils d’étiquetage manuels et semi-automatiques. Le radiologue peut choisir ces outils pour effectuer la segmentation et délimiter avec précision la tumeur. La segmentation manuelle de tumeur méningiome prend environ une heure, bien que la segmentation manuelle chronophage, demeure une référence standard pour valider les résultats de notre segmentation entièrement automatique, renforçant ainsi sa crédibilité.

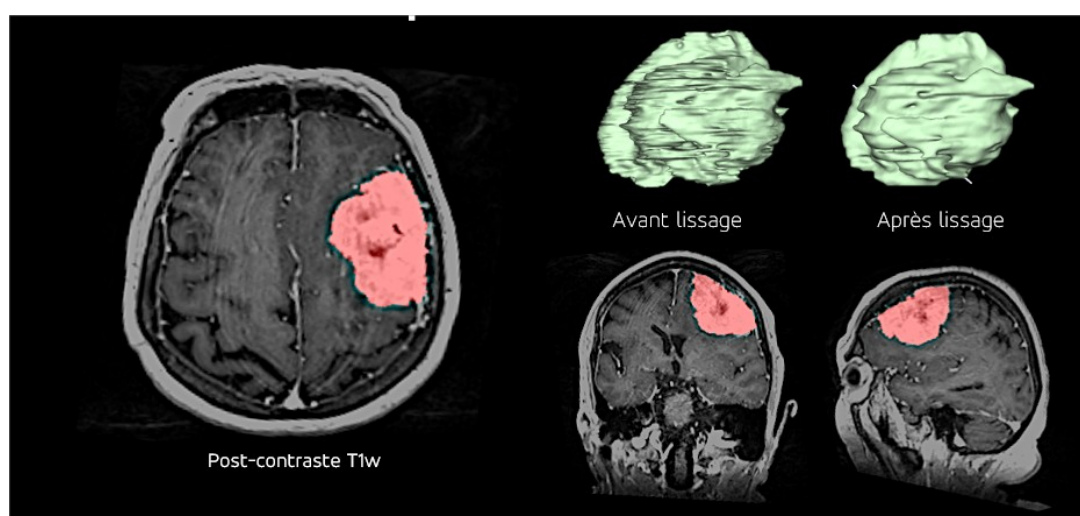


FIGURE 4.1 – Création du masque tumoral de méningiome à l’aide de 3D slicer par le radiologue à I3M.

4.1.3 Méthode et application

4.1.3.1 Pré-traitement des données

Les données IRM que nous avons obtenues ont des orientations, des formes et des tailles de voxels différentes. Pour cela nous devons appliquer un pré-traitement sur l'ensemble des données pour mettre les images dans le même espace de voxels avant d'appliquer une méthode de segmentation pour ce but nous avons utilisé plusieurs techniques de transformation indépendantes pour le pré-traitement et l'augmentation des données.

- ré-échantillonnage des image à un espacement isotropique de 1 mm.
- L'architecture du modèle nécessite des entrées de mêmes dimensions, un centre de $256 \times 256 \times 256$ pixels a été obtenu à partir des volumes pré-traités.
- Normalisation des intensités des volumes IRM dans l'intervalle [0-1].

Normalisation des intensités des volumes IRM dans l'intervalle [0-1] Afin de

En suite pour augmenter les données, plusieurs transformations aléatoires ont été utilisées, telles que des rotations d'images, des transformations affines, des déformations élastiques, des ajustements de gamma, du bruit gaussien, Ces transformations permettent de diversifier les images d'entraînement en simulant différentes variations et artefacts, améliorant ainsi la robustesse du modèle face aux variations présentes dans les données réelles.

Pour réaliser ces différentes applications de pré-traitement des données, nous avons utilisé TorchIO, une bibliothèque open source spécialement conçue pour le traitement des images médicales en apprentissage profond. TorchIO offre une gamme de fonctionnalités pour charger les données, appliquer des pré-traitements avancées [9].

4.1.4 Architectures proposées

Pour expliquer en détail les architectures proposées pour la segmentation automatique des données de méningiome, nous nous concentrerons sur les deux modèles choisis : le 3D U-Net et le UNeTR. Ces modèles sont basés sur le principe de l'encodeur-décodeur, chacun ayant ses propres caractéristiques distinctives et applications optimales ;

Modèle U-Net

Un modèle 3D U-Net [10] tridimensionnel standard a été utilisé pour la segmentation des images médicales. Son architecture repose sur un schéma encodeur-décodeur, qui est particulièrement adapté pour capturer à la fois des informations de bas niveau et de haut niveau à partir des images d'entrée. L'architecture de cet algorithme est montrée dans la (fig 4.2)

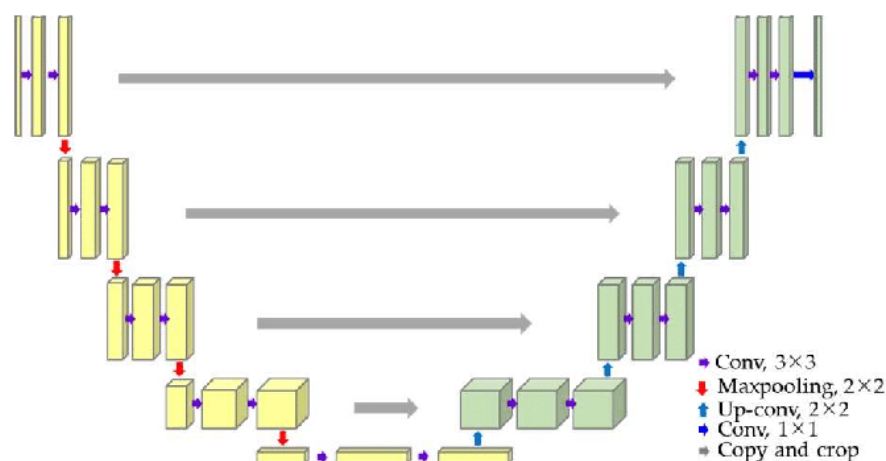


FIGURE 4.2 – Architecture 3D U-Net utilisée dans cette étude. Le nombre de couches et le nombre de filtres pour chaque couche peuvent varier en fonction de la résolution de l'échantillon d'entrée.

L'architecture du modèle utilisé se compose :

- **Encodeur** : L'encodeur est conçu pour extraire des informations contextuelles riches à partir de l'image d'entrée. Il est constitué de 4 blocs d'encodage.
- **Convolutions 3D avec activation ReLU** : Ces convolutions permettent d'extraire des caractéristiques complexes des données volumétriques.
- **Pooling Max et normalisation** : Après chaque convolution, un pooling max est appliqué pour réduire la dimension spatiale tout en conservant les informations importantes.
- **Décodeur** : Le décodeur reconstruit ensuite l'image segmentée en appliquant des opérations de convolution et d'*upsampling*.

Le nombre de filtres commence à 16 et double à chaque bloc, ce qui permet au modèle d'apprendre des caractéristiques plus abstraites et complexes à mesure que la profondeur du réseau augmente.

Modèle UNETR

Des études ont souligné l'efficacité d'UNETR dans les tâches de segmentation, attribuant son succès à une extraction robuste des caractéristiques et à des capacités de localisation spatiale précises. C'est pourquoi nous avons décidé de comparer ses performances à celles de U-Net 3D, afin d'évaluer les améliorations en termes de précision et d'efficacité pour les applications d'imagerie médicale, en particulier pour des tâches telles que la segmentation des méningiomes [13].

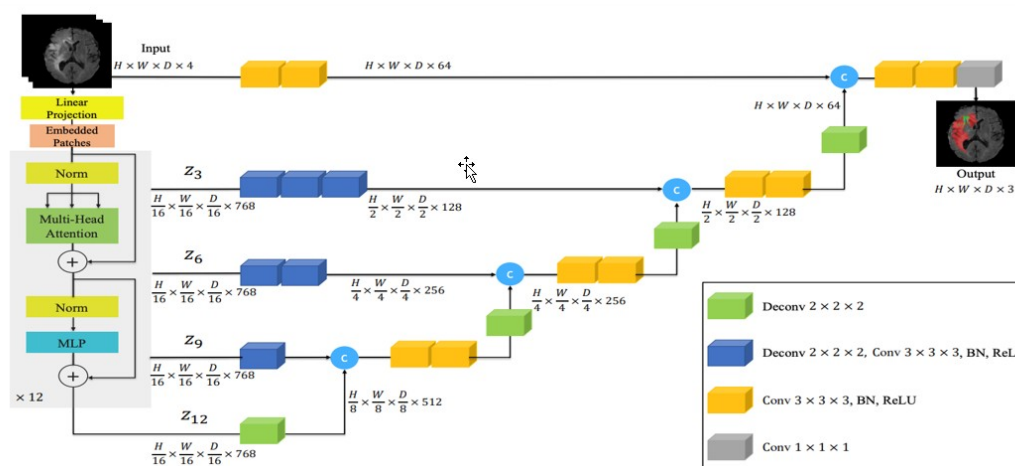


FIGURE 4.3 – Vue de l'architecture d'UNETR. Le modèle proposé consiste en un encodeur basé sur un transformer qui est connecté à un décodeur basé sur CNN via des connexions de type skip [4].

UNETR utilise un schéma de contraction-expansion constitué d'une pile de *transformers* en tant qu'encodeur, qui est connecté à un décodeur via des connexions de type *skip*. Comme couramment utilisé en traitement du langage naturel (NLP), les transformers opèrent sur des séquences 1D d'embeddings en entrée, L'architecture du modèle utilisé se compose de :

- **Transformateurs** : Le cœur de l'UNETR repose sur les transformateurs, qui sont conçus pour capturer des dépendances globales dans les données. Contrairement à l'architecture U-Net, qui se concentre sur les dépendances locales.
- **Blocs de Transformateurs** : Chaque bloc de transformateur suit la séquence suivante :
 - **Multi-Head Self-Attention (MSA)** : Le mécanisme d'attention à plusieurs têtes permet au modèle de se concentrer sur différentes parties de l'image simultanément, capturant ainsi des informations importantes de différentes régions.
 - **Normalization and Residual Connections** : Après chaque couche de MSA, une normalisation est appliquée, suivie d'une connexion résiduelle qui aide à

stabiliser l'entraînement et à éviter la perte de gradient.

- **Multi-Layer Perceptron (MLP)** : Cette partie traite les caractéristiques extraites par les couches précédentes et applique des transformations non linéaires supplémentaires, comme illustré dans les équations suivantes :

$$z'_i = \text{MSA}(\text{Norm}(z_{i-1})) + z_{i-1}, \quad i = 1, \dots, L, \quad (4.1)$$

$$z_i = \text{MLP}(\text{Norm}(z'_i)) + z'_i, \quad i = 1, \dots, L. \quad (4.2)$$

Ces équations montrent le processus d'apprentissage résiduel dans les blocs de transformateurs. Chaque sortie de couche est normalisée, suivie d'une attention multi-tête ou d'un MLP, puis additionnée à l'entrée d'origine pour former une connexion résiduelle.

4.1.5 Métriques d'évaluation

Dans cette section, nous décrivons deux métriques fondamentales utilisées pour évaluer l'entraînement des modèles : le coefficient de Dice et la distance de Hausdorff. [15]

Dice coefficient

Pour quantifier la performance de notre segmentation d'image, le score de Dice est utilisé. L'algorithme est validé en calculant le score de Dice, qui mesure la similarité entre les objets. Il s'agit donc de la taille de l'intersection des deux segmentations divisée par la taille totale des deux objets.

Le coefficient de Dice entre le volume segmenté A et le volume de la segmentation manuelle de référence B s'exprime comme suit :

$$DICE = \frac{2|A \cap B|}{|A| + |B|}$$

Le coefficient de Dice prend ses valeurs entre 0 et 1, 0 signifiant que la lésion segmentée n'a pas d'intersection avec la segmentation de référence, et 1 que le recouvrement est total entre le volume de la segmentation et celui de la segmentation de référence.

Distance du Hausdorff :

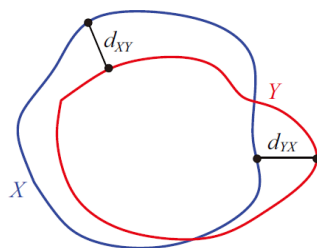


FIGURE 4.4 –

Schéma montrant la distance de Hausdorff entre les ensembles de points X et Y.

La distance de Hausdorff est une autre métrique qui est de plus en plus utilisée dans la segmentation d'images médicales [6]. Contrairement aux métriques précédentes qui considèrent chaque pixel, la distance de Hausdorff ne prend en compte que les pixels des contours de chaque patch de pixels appartenant à la même classe. Elle calcule la distance entre chaque point correspondant à l'intérieur d'une ou deux frontières. Par définition, la distance de Hausdorff ne considère que la distance maximale entre deux points. La distance de Hausdorff moyenne est également utilisée pour tenir compte de tous les pixels de contour d'un patch.

$$d_{\text{AHD}}(X, Y) = \frac{1}{2} \left(\frac{1}{|X|} \sum_{x \in X} \min_{y \in Y} d(x, y) + \frac{1}{|Y|} \sum_{y \in Y} \min_{x \in X} d(x, y) \right) \quad (4.3)$$

4.1.5.1 Entraînement

Lors du déploiement des deux architectures à l'aide de la bibliothèque MONAI[3], des ajustements ont été apportés pour tenir compte des caractéristiques des méningiomes. Les modèles ont été entraînés selon des configurations identiques, à l'exception des tailles de *batch* : 48 pour U-Net et 1 pour UNETR. Lors du traitement des images 3D, le pré-traitement est essentiel pour gérer efficacement les contraintes de mémoire GPU. Une approche courante consiste à réduire la taille du volume d'entrée ou à le diviser en patches séquentiels pour le traitement. Alternativement, réduire la taille du lot à 1 est une autre méthode, bien que cela puisse entraîner une convergence plus lente en raison de mises à jour de gradient moins efficaces par itération. Le choix de la taille du lot dans ce cas a été influencé par la nécessité de s'assurer que le modèle s'adapte aux limites disponibles de la mémoire GPU sans compromettre trop la vitesse de convergence.

4.1.6 Configuration expérimentale

le modèle a été entraîné en utilisant l'optimiseur Adam, pour 200 époques en utilisant un seul GPU NVIDIA A100-PCI (40 Go), avec un taux d'apprentissage de $l=1e-3$, qui a été ajusté de manière dynamique au cours de l'entraînement pour éviter le surajustement aussi une régularisation L2 avec un coefficient de $r = 1e-5$ a été appliquée au poids de chaque couche, l'objectif étant d'encourager le modèle à apprendre des modèles d'échantillonnage en gardant le poids plus petit et pour optimiser les performances.

La taille de l'ensemble de données est 285 avec 80 % d'échantillons aléatoires pour l'entraînement, 10% pour la validation et 10% pour le test.

4.1.7 Résultat et discussion

L'entraînement des réseaux de neurones de segmentation utilisés dans cette étude est relativement exigeant en termes de calcul. Pour cette raison, nous comparons les architectures résultantes également du point de vue du temps de calcul nécessaire pour l'entraînement et du temps de calcul nécessaire pour la prédiction. Les résultats peuvent être consultés dans le (tableau 4.2).

Algorithme	Temps de Prédiction	Temps d'Entraînement (200 époques)
UNet	10 s	33 h
UNETR	15 s	50 h

TABLE 4.2 – Comparaison des temps de prédiction et d'entraînement pour l'algorithme de segmentation UNet et UNETR

Comme il est évident dans le tableau, le temps de calcul augmente avec la complexité du modèle.

4.1.7.1 Résultats

Lorsque nous comparons visuellement les résultats de segmentation de UNet et de UNETR (fig 4.5) avec la vérité de terrain, nous remarquons qu'UNet montre des difficultés à capturer les bordures précises des tumeurs, en particulier dans les régions avec des textures hétérogènes il manque des parties de la tumeur aussi, en revanche UNETR à segmenter les contours fins et à capturer les détails des structures.

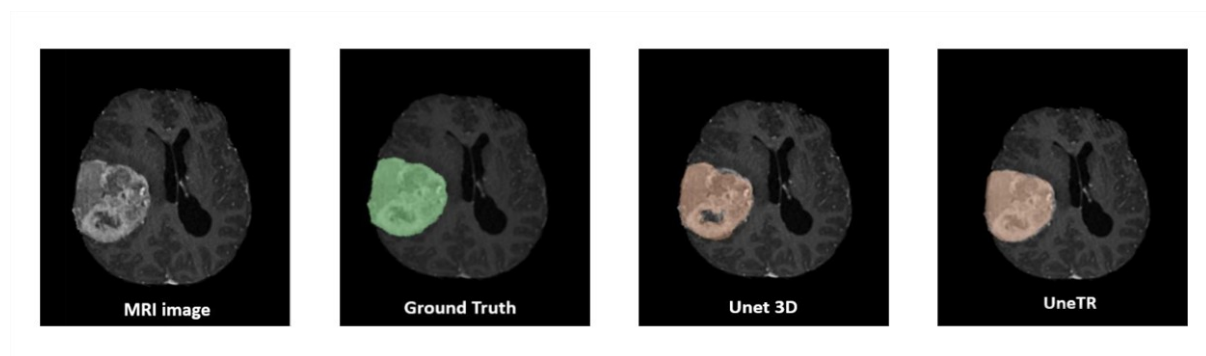


FIGURE 4.5 – Résultat de la segmentation prédit par le modèle UNet et le modèle UNETR avec la vérité terrain en vue axiale.

Nous expliquons cette différence au niveau de performance à ces circonstances suivantes :

- Unet peut avoir des difficultés à segmenter les tumeurs aux textures hétérogènes, car les couches convolutives peuvent ne pas saisir pleinement la variabilité au sein de la région tumorale,
- Grâce à l'analyse spatiale des *transformers*, UneTR comprend que la partie qui n'est pas segmentée par Unet est une nécrose cellulaire appartenant à la tumeur donc il la segmente aussi, ça montre une meilleure couverture des zones tumorales, en incluant les parties manquantes ,
- La complexité anatomique des méningiomes peut rendre la segmentation difficile, en particulier pour les tumeurs qui sont proches de structures complexes particulières.

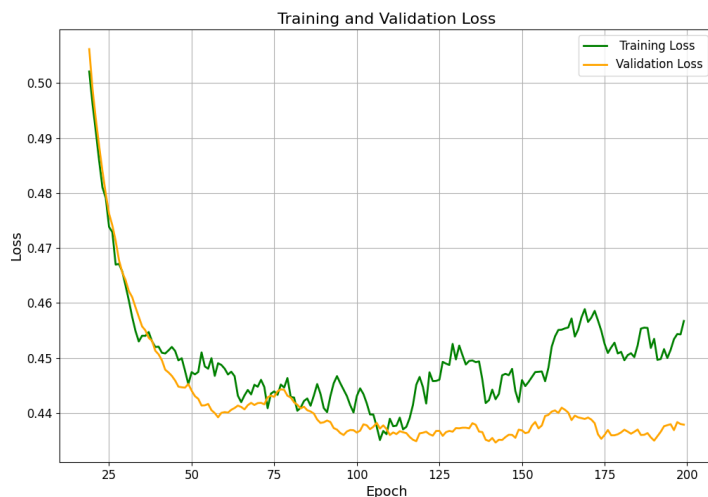
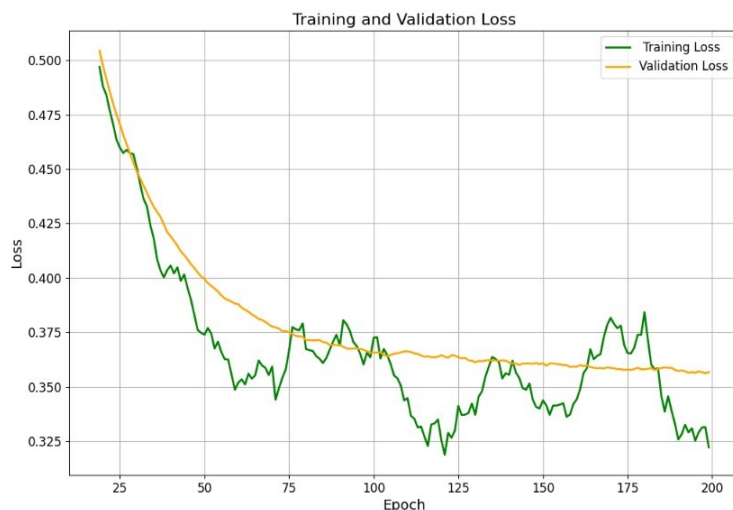
(a) Courbe de *loss* pour UNETR(b) Courbe de *loss* pour UNet

FIGURE 4.6 –

Comparaison de la courbe du *loss* entre le modèle UNet et UNETR, **Courbe verte** : présente la *loss* pour l'entraînement. **Courbe jaune** : présente la *loss* pour l'ensemble de validation.

Graps (a) - UNETR : La perte d'entraînement (ligne jaune) et la perte de validation (ligne verte) diminuent significativement lors des premières époques, indiquant un apprentissage efficace du modèle. La perte de validation présente quelques fluctuations mais conserve une tendance générale à la baisse, ce qui suggère une certaine instabilité, mais avec une amélioration continue. La perte d'entraînement diminue de manière cohérente, montrant que le modèle s'adapte bien aux données d'entraînement.

Graphes (b) - UNet :

La valeur de perte d'entraînement diminue fortement lors des premières époques et continue de baisser de façon régulière, de manière similaire à UNETR. La perte de validation diminue également mais avec des fluctuations plus prononcées comparées à UNETR, ce qui indique un potentiel surapprentissage ou une variabilité dans la généralisation. Il y a un écart notable entre les valeurs de pertes d'entraînement et de validation, suggérant que UNet pourrait légèrement surapprendre les données d'entraînement.

Comparaison :

- **Tendance Générale** : Les modèles UNETR et UNet montrent tous deux un apprentissage efficace avec des baisses significatives des pertes d'entraînement et de validation lors des premières époques.
- **Stabilité** : UNETR montre une perte de validation relativement plus stable avec moins de fluctuations par rapport à UNet, ce qui indique une meilleure généralisation et stabilité durant l'entraînement pour UNETR.
- **Surapprentissage** : UNet présente plus de fluctuations dans la perte de validation et un écart plus important entre les pertes d'entraînement et de validation, ce qui suggère qu'il pourrait surapprendre davantage par rapport à UNETR.

En résumé, UNETR démontre une meilleure stabilité et généralisation, tandis que UNet, bien qu'efficace dans l'apprentissage, montre des signes de surapprentissage et une variabilité plus importante dans la performance de validation. U-NeTR surpasse U-Net dans

Modèle	Dice coefficient	Hausdorff Average
U-Net	0.57	80
U-NeTR	0.70	45

TABLE 4.3 – Analyse des performances de validation entre le modèle UNet et UNETR

les trois mesures (*Dice loss*, *Dice coefficient* et distance de Hausdorff) (Table 4.3). Cette performance supérieure est attribuée à la capacité d'UNeTR à exploiter les mécanismes basés sur les transformateurs pour capturer efficacement le contexte global et les dépendances spatiales, ce qui permet une segmentation plus précise et plus exacte des tumeurs du méningiome.

4.1.8 Limites

Lors de la réalisation de cette étude, plusieurs limites étaient présentes, surtout des contraintes techniques telles que :

- Tant que l'apprentissage supervisé est utilisé, la disponibilité d'ensembles de données annotées de haute qualité nécessaires à l'entraînement est un obstacle majeur.

- Validation technique : une fois que l'algorithme a été entraîné, il est difficile de prouver sa robustesse et sa fiabilité.
- Une autre raison pour laquelle il est difficile d'intégrer les algorithmes dans le flux de travail de la radiologie est l'absence de normes pour le partage des données entre les systèmes numériques.

Défis rencontrés

Le traitement des images médicales, est rendu difficile notamment en IRM, au cours de la réalisation de la 1ère mission j'ai adresser plusieurs problèmes telles que :

- La difficulté de traiter les méta-données liées à la position physique et à la taille du voxel.
- Coûts de calcul élevés
- Absence de consensus sur les meilleures pratiques de normalisation [11]

Conclusion

En résumé, UNeTR tend à surpasser U-Net dans les tâches de segmentation des tumeurs, notamment dans des conditions impliquant des formes de tumeurs complexes, une variabilité de taille, des textures hétérogènes, du bruit et des artefacts. Sa structure avancée lui permet de capturer des détails plus complexes et de produire des segmentations plus précises, en en faisant un choix préféré pour les tâches difficiles en imagerie médicale. le travail est encore au cours d'amélioration. Nous expérimentons la configuration et le réglage fin des hyperparamètres pour trouver les meilleurs paramètres. Notre objectif est d'améliorer les résultats de la segmentation et la qualité de notre travail.

4.2 Mission 2 : Développement d'un orchestrateur d'IA

4.2.1 Choix techniques

Nous avons choisi un ensemble de technologie robuste, et solide pour notre projet après une comparaison faite entre différents solutions techniques en analysant les points forts et les points négatives de chaque outils pour choisir les contraintes techniques nécessaire à la mise en oeuvre de notre solution.



FIGURE 4.7 – Docker Logo

Docker encapsule les applications dans des conteneurs légers et portables, garantissant un fonctionnement cohérent à travers différents environnements, ce qui simplifie le développement, le déploiement et la maintenance.



FIGURE 4.8 – FastAPI Logo

FastAPI est un framework web performant pour créer des API RESTful en Python. Il offre la validation automatique des données, la gestion des erreurs, et une documentation interactive des API, idéal pour des environnements plus petits.



FIGURE 4.9 – RabbitMQ Logo

RabbitMQ est un courtier de messages léger et open source qui prend en charge plusieurs protocoles, adapté pour des configurations distribuées et des exigences de haute disponibilité.

4.2.2 Étude architecturale

4.2.2.1 Architecture micro-services

Pour la mise en oeuvre d'orchestrateur nous avons adopté le paradigme des micro-services. Cette approche construit une application comme un ensemble de petits services indépendants, chacun fonctionnant dans son propre processus et interagissant via des méthodes simples et légères. Ces services, organisés autour de fonctions métier ou de mission spécifiques, peuvent être déployés indépendamment via des systèmes [7]. Cette méthode simplifie la création et la maintenance de certains types d'applications en les

décomposant en éléments plus petits et inter-opérables. Chaque composant est développé séparément, ce qui fait de l'application un composite des parties individuelles.

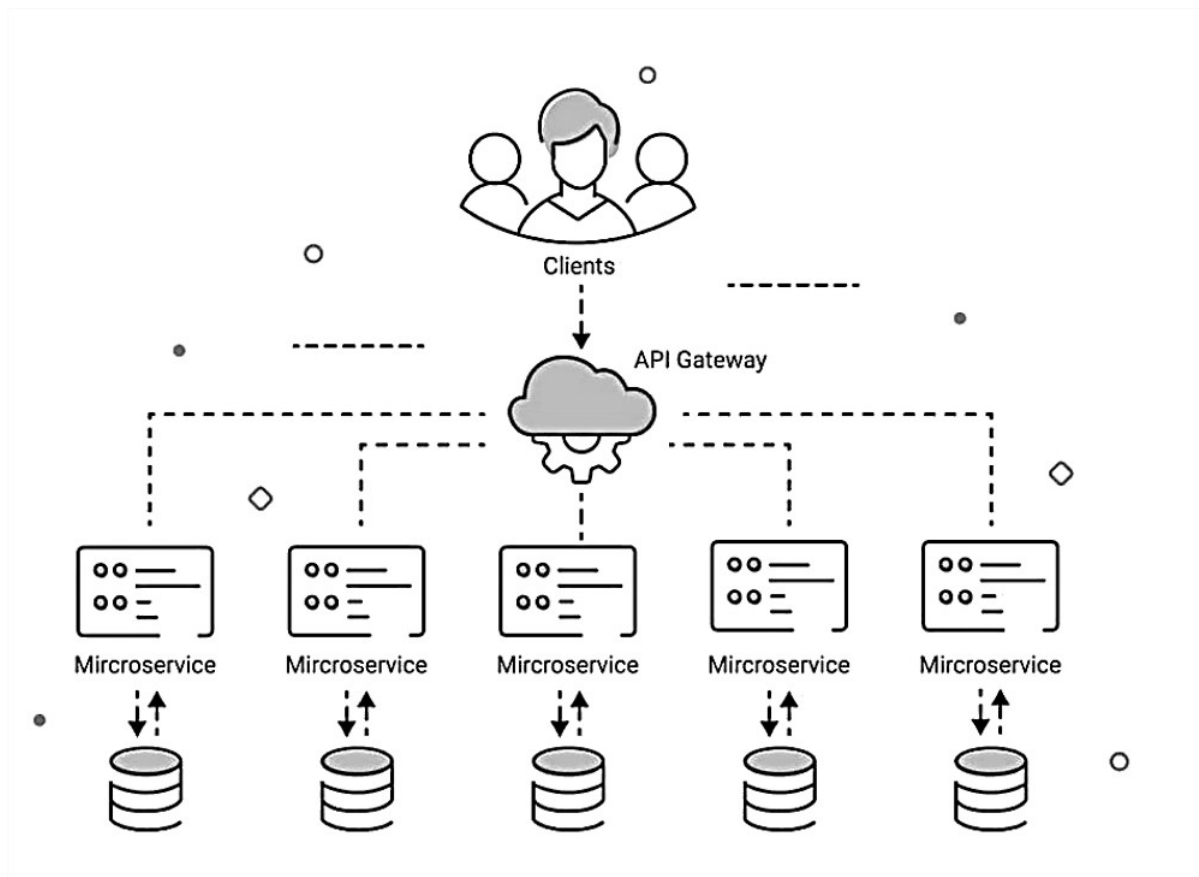


FIGURE 4.10 – Architecture micro-services

Dans la suite, nous présenterons l'ensemble des avantages et des inconvénients puisqu'il est nécessaire de connaître notre micro-service à quoi être capable de nous offrir

Avantages de l'architecture micro-services

- Limite les problèmes de la complexité,
- l'Agilité technologique,
- Optimise les ressources,
- Rapidité de déploiement des nouvelles fonctionnalités.

Inconvénients de l'architecture micro-services

- Longue durée de lancement d'application.
- Incohérence de données.
- Besoin d'automatisation des tests obligatoire.

4.2.2.2 Micro-services associé à l'orchestrateur

— Service de Réception des Données

- **Fonction** : Ce service agit en amont de l'orchestrateur, recevant les données du PACS ou d'autres sources et les transmettant à l'orchestrateur avec les métadonnées nécessaires.
- **Interaction avec l'Orchestrateur** : L'orchestrateur utilise les informations de ce service pour prendre des décisions sur le modèle IA à utiliser.

— Service de Gestion des Modèles

- **Fonction** : Ce service maintient un registre des modèles disponibles, enregistre les performances et gère les versions des modèles.
- **Interaction avec l'Orchestrateur** : L'orchestrateur interroge ce service pour obtenir des informations sur les modèles disponibles et choisir le modèle le plus adapté à une tâche donnée.

— Services d'inférence IA

- **Fonction** : Chaque micro-service exécute un modèle IA spécifique (par exemple, un modèle de classification d'image, un modèle de segmentation).
- **Interaction avec l'orchestrateur** : L'orchestrateur appelle ces micro-services pour exécuter les inférences nécessaires, passant les données reçues et récupérant les résultats d'inférence.

Pour une application avec ce type d'architecture en micro-services, il est nécessaire d'exécuter l'ensemble (y compris les bibliothèques, les outils système, le code et le runtime) dans des conteneurs Docker. Cela garantit que les micro-services se comportent de manière identique, quel que soit l'endroit où ils sont déployés. Les

conteneurs Docker offrent un environnement idéal pour le déploiement des micro-services, car chaque conteneur peut héberger un micro-service distinct, assurant ainsi l'isolation et réduisant les conflits entre les services. Cette isolation simplifie les processus de développement, de test et de déploiement, rendant l'application plus efficace et évolutive .

4.2.3 Méthodes

le diagramme (fig 4.11) représente une plate-forme orchestratrice pour la gestion des images médicales et l'intégration de modèles d'intelligence artificielle (IA) dans un flux de travail automatisé dans une environnement clinique. L'orchestrateur joue un rôle central en gérant la réception des images médicales (DICOM), en sélectionnant les études pertinentes pour l'IA, en exécutant des inférences à l'aide de modèles d'IA déployés, et en redirigeant les résultats vers le système PACS pour un stockage et une consultation ultérieure, Voici une explication détaillée de chaque composant et du flux d'information :

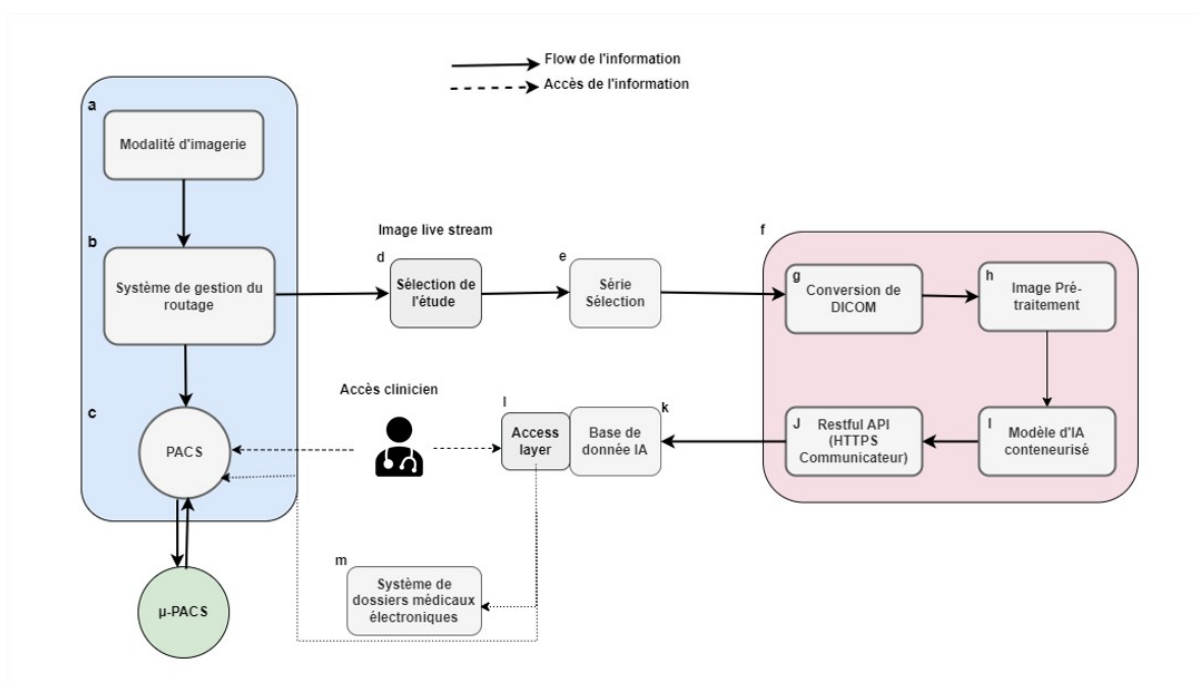


FIGURE 4.11 – flux de travail pour le développement clinique de l'orchestrateur d'IA

Développement du modèle

Nous avons développé des modèles avancés d'apprentissage profond spécifiquement conçus pour la segmentation automatique des tumeurs méningiomes. Pour valider l'efficacité de notre algorithme d'IA dans un contexte clinique réel, nous l'avons intégré dans notre système d'orchestration permettant ainsi d'effectuer des inférences sur des données d'imagerie acquises en temps réel.

Routage et sélection d'images médicales en temps réel

Les examens d'imagerie médicale évalués par un modèle d'IA lors de l'inférence doivent présenter des caractéristiques similaires aux données d'apprentissage par exemple même la modalité d'acquisition, la vue de l'image d'acquisition, l'espacement des pixels, etc. Nous classons le processus consistant à fournir l'image médicale la plus appropriée à un modèle d'IA spécifique en trois catégories connexes : sélection de l'étude, sélection de la série, et le pré-traitement des images.

Sélection d'étude

Nous devons d'abord identifier efficacement et automatiquement les études qui présentent un intérêt pour un modèle d'IA particulier et les transmettre à la plateforme d'inférence d'IA. Cette tâche est réalisée par un système de gestion du routage 4.11.b qui reçoit les images acquises par toutes les modalités d'imagerie de notre hôpital 4.11.a et sélectionne les examens d'imagerie spécifiques à l'aide d'éléments de données DICOM.

Conteneurisation des modèles d'IA

Notre flux de travail de développement de modèles produit une image Docker qui contient le code de prétraitement décrit, le modèle, ainsi que le code pour stocker les résultats. L'image Docker est configurée pour lire les fichiers NIFTI à partir d'un dossier d'entrée et écrire les résultats dans un dossier de sortie. Une fois que le conteneur du modèle d'IA a terminé son exécution, on prend la sortie et la fournit comme entrée pour le prochain conteneur de la chaîne.

4.2.4 Architecture logicielle

l'architecture logiciel(fig4.12) illustre un système d'orchestration robuste pour intégrer des modèles d'IA dans les flux de travail en radiologie. Il met en œuvre une approche basée sur les microservices avec des conteneurs Docker pour la gestion et le traitement des images, tout en assurant la sécurité et la performance à travers des interfaces bien définies et des points de terminaison API pour accéder à les données.

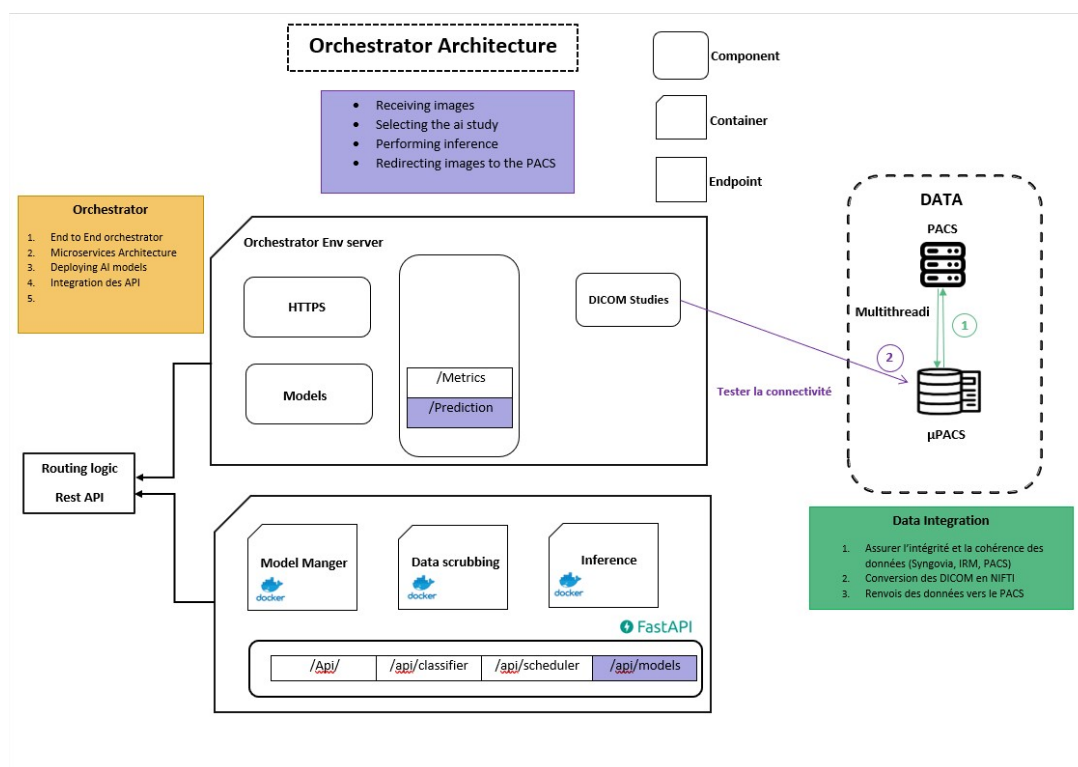


FIGURE 4.12 – Architecture logicielle du développement d'orchestrateur

Notre architecture est basée sur les micro-services et les API signifie que nous pouvons connecter à tous et définir ce qu'on veut dans le pipeline du workflow d'IA.

L'architecture est conçue pour gérer les images médicales en réalisant les tâches suivantes :

- Stocker les images
- Sélectionner l'étude
- Effectuer les inférences
- Renvoyer les images vers le μ -PACS

Voici une analyse technique détaillée de chaque composant et de son fonctionnement :

Orchestrateur

— **Fonctions** : L'orchestrateur est responsable de la gestion de bout en bout des opérations, y compris la réception des images médicales, la sélection des études pertinentes pour l'IA, l'exécution des inférences à l'aide de modèles d'IA, et la redirection des images et résultats vers le système PACS.

— **Architecture** :

— **Microservices** : Chaque fonction principale de l'orchestrateur est encapsulée dans un microservice déployé via Docker, permettant une gestion indépendante et un déploiement flexible.

- **Communication** : Les microservices communiquent entre eux via des API REST sécurisées avec HTTPS.
- API Endpoints :
 - **/Metrics** : Endpoint pour la surveillance des performances et l'état des services.
 - **/Prediction** : Endpoint principal pour soumettre les images et obtenir les prédictions des modèles d'IA.

Environnement Serveur de l'Orchestrateur

- **Composants** :
 - **HTTPS** : Assure la sécurité des communications entre les différents modules et les systèmes externes.
 - **Modèles** : Contient les modèles d'IA déployés et gérés par le gestionnaire de modèles.
 - **DICOM Studies** : Reçoit les études DICOM en entrée pour traitement et exécution des inférences.
- **Connectivité** : Un test de connectivité assure la disponibilité et la fiabilité de l'accès aux données DICOM depuis le PACS.

Gestionnaire de Modèles (Model Manager)

- **Déploiement** : Utilise Docker pour déployer des modèles d'IA, facilitant la gestion des versions et la scalabilité des modèles.
- **Fonctions** : Gère l'enregistrement, la mise à jour et la suppression des modèles d'IA, et fournit les modèles nécessaires pour les inférences sur demande via les endpoints API [12].

Intégration des Données

- **Composants** :
 - **µPACS** : Système de stockage intermédiaire optimisé pour les opérations multithreadées, permettant un accès rapide et concurrentiel aux données DICOM.
- **Processus** :
 1. **Assurer l'intégrité et la cohérence des données** : Validation des formats et synchronisation des données provenant de différentes sources (PACS, systèmes de IRM, etc.).
 2. **Conversion DICOM en NIFTI** : Standardise les images pour l'inférence, facilitant le traitement par les modèles d'IA.
 3. **Renvoyer les données vers le PACS** : Après traitement, les résultats et images sont renvoyés au PACS pour stockage et accessibilité par les cliniciens.

Conclusion

En conclusion, ce chapitre résume notre travail et la réalisation de deux missions principales. Nous mettons en évidence les résultats obtenus tout au long du projet, démontrant les avantages et l'efficacité de l'intégration de l'IA dans le traitement des images médicales grâce à une architecture logicielle bien conçue et modulaire.

Conclusion générale

Conclusion du Projet

Le projet d'orchestrateur d'IA pour le laboratoire I3M est ambitieux et stratégique, visant à transformer la manière dont les modèles d'intelligence artificielle sont intégrés et gérés au sein des flux de travail cliniques et de recherche. Jusqu'à présent, les efforts ont été principalement concentrés sur la compréhension approfondie du pipeline de segmentation des images médicales, l'identification des éléments clés nécessaires à l'intégration, et la préparation du terrain pour le développement de l'orchestrateur.

Nous avons soigneusement examiné les défis inhérents à ce projet, notamment les contraintes liées à la sécurité des données, la conformité aux réglementations en vigueur, et la complexité des intégrations avec les systèmes hospitaliers existants tels que les PACS. Ces éléments sont cruciaux pour garantir que l'orchestrateur ne se contente pas de fonctionner correctement, mais qu'il respecte également les normes rigoureuses de l'environnement médical.

Il reste encore un travail conséquent pour mener à bien ce projet, notamment en ce qui concerne le développement technique et la mise en œuvre de l'orchestrateur. La phase de développement exigera des choix technologiques judicieux pour s'assurer que l'orchestrateur soit à la fois performant, sécurisé et adaptable aux évolutions futures. Il sera également essentiel de valider continuellement l'alignement du projet avec les besoins du laboratoire et de ses utilisateurs finaux, pour s'assurer que les solutions apportées répondent effectivement aux enjeux identifiés.

Dans ce contexte, il est essentiel de faire les meilleurs choix techniques pour garantir le succès du développement de l'orchestrateur.

Conclusion Personnelle

Au cours de ces six mois de stage, j'ai pu mettre en pratique ce que j'ai appris au fil des ans dans mes études universitaires, en commençant par :

- Vision par ordinateur
- Apprentissage profond
- Analyse des données
- Gestion de projet

J'ai orienté ces connaissances vers un domaine qui m'intéresse. J'ai découvert que je voulais en apprendre davantage et améliorer mes compétences dans ce domaine. De plus, le fait d'être dans un environnement collaboratif riche comme le laboratoire I3M a été très encourageant. Cela m'a constamment motivé à apprendre et

à améliorer mes compétences techniques tout en partageant des informations et en discutant de divers sujets avec des médecins, des radiologues, des techniciens et des chercheurs.

Je dois également mentionner qu'au départ, je n'avais aucune connaissance ou expérience des technologies de conteneurs ou de la manière de déployer une architecture de micro-services. Une grande partie de mon travail a d'abord consisté à comprendre le fonctionnement de ces systèmes. C'était une excellente occasion d'apprendre quelque chose de nouveau.

- [1] W. D. Bidgood, S. C. Horii, F. W. Prior, and D. E. Van Syckle. Understanding and using DICOM, the data interchange standard for biomedical imaging. 4(3) :199–212.
- [2] J. Boetto, C. Birzu, M. Kalamarides, M. Peyre, and M. Sanson. Les méningiomes : mise au point sur les connaissances actuelles. 43(2) :98–105.
- [3] M. Jorge Cardoso, Wenqi Li, Richard Brown, Nic Ma, Eric Kerfoot, Yiheng Wang, Benjamin Murrey, Andriy Myronenko, Can Zhao, Dong Yang, Vishwesh Nath, Yufan He, Ziyue Xu, Ali Hatamizadeh, Wentao Zhu, Yun Liu, Mingxin Zheng, Yucheng Tang, Isaac Yang, Michael Zephyr, Behrooz Hashemian, Sachidanand Alle, Mohammad Zalbagi Darestani, Charlie Budd, Marc Modat, Tom Vercauteren, Guotai Wang, Yiwen Li, Yipeng Hu, Yunguan Fu, Benjamin Gorman, Hans Johnson, Brad Genereaux, Barbaros S. Erdal, Vikash Gupta, Andres Diaz-Pinto, Andre Dourson, Lena Maier-Hein, Paul F. Jaeger, Michael Baumgartner, Jayashree Kalpathy-Cramer, Mona Flores, Justin Kirby, Lee A. D. Cooper, Holger R. Roth, Daguang Xu, David Bericat, Ralf Floca, S. Kevin Zhou, Haris Shuaib, Keyvan Farahani, Klaus H. Maier-Hein, Stephen Aylward, Prerna Dogra, Sebastien Ourselin, and Andrew Feng. MONAI : An open-source framework for deep learning in healthcare. Version Number : 1.
- [4] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L. Yuille, and Yuyin Zhou. TransUNet : Transformers make strong encoders for medical image segmentation. Version Number : 1.
- [5] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, John Buatti, Stephen Aylward, James V. Miller, Steve Pieper, and Ron Kikinis. 3d slicer as an image computing platform for the quantitative imaging network. 30(9) :1323–1341.

- [6] Davood Karimi and Septimiu E. Salcudean. Reducing the hausdorff distance in medical image segmentation with convolutional neural networks. 39(2) :499–513.
- [7] Elmar Kotter and Erik Ranschaert. Challenges and solutions for introducing artificial intelligence (AI) in daily clinical workflow. 31(1) :5–7.
- [8] Hyunhee Lee, Jaechoon Jo, and Heuseok Lim. Study on optimal generative network for synthesizing brain tumor-segmented MR images. 2020 :1–12.
- [9] Fernando Pérez-García, Rachel Sparks, and Sébastien Ourselin. TorchIO : A python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning. 208 :106236.
- [10] Rehan Raza, Usama Ijaz Bajwa, Yasar Mehmood, Muhammad Waqas Anwar, and M. Hassan Jamal. dResU-net : 3d deep residual u-net based brain tumor segmentation from multimodal MRI. 79 :103861.
- [11] Jacob C. Reinhold, Blake E. Dewey, Aaron Carass, and Jerry L. Prince. Evaluating the impact of intensity normalization on MR image synthesis. In Elsa D. Angelini and Bennett A. Landman, editors, *Medical Imaging 2019 : Image Processing*, page 126. SPIE.
- [12] Carlos Rodríguez, Marcos Baez, Florian Daniel, Fabio Casati, Juan Carlos Trabucco, Luigi Canali, and Gianraffaele Percannella. REST APIs : A large-scale analysis of compliance with principles and best practices. In Alessandro Bozzon, Philippe Cudre-Maroux, and Cesare Pautasso, editors, *Web Engineering*, volume 9671, pages 21–39. Springer International Publishing. Series Title : Lecture Notes in Computer Science.
- [13] Guanqun Sun, Yizhi Pan, Weikun Kong, Zichang Xu, Jianhua Ma, Teeradaej Racharak, Le-Minh Nguyen, and Junyi Xin. DA-TransUNet : integrating spatial and channel dual attention with transformer u-net for medical image segmentation. 12 :1398237.
- [14] Hamidreza Tadayon, Behjat Nafari, Gholamreza Khadem, Reza Darrudi, and Monireh Sadeqi Jabali. Evaluation of picture archiving and communication system (PACS) : Radiologists' perspective. 39 :101266.
- [15] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation : analysis, selection, and tool. 15(1) :29.

Annexe

N°	Identification des risques		Evaluation et analyse des risques			Zone d'impact	
	Risques	Description	Nature de risque	Probabilité (1-5)	Gravité (1-5)		Criticité (P*G)
1	Qualité des données	Basse qualité d'image, différentes modalités d'imagerie, erreurs dans la segmentation manuelle.	Technique	2	4	8	<ul style="list-style-type: none"> Intégrité des données, Performance des modèles, Expérience utilisateur
2	Complexité algorithmique	le modèle de segmentation peuvent être complexe à implémenter.	Technique	3	3	9	<ul style="list-style-type: none"> Efficacité de l'implémentation Temps de développement Performance du système
3	Compatibilité des formats de fichiers	la compatibilité entre différents formats de données peut poser des problèmes lors de l'intégration dans l'orchestrateur ou perte de données.	Technique	5	3	15	<ul style="list-style-type: none"> Intégrité des données Compatibilité du système Perte de données
4	Sécurité des données	Risques de violation de la sécurité des données lors de l'intégration et du transfert de données au niveau d'orchestrateur	Sécurité	3	4	12	<ul style="list-style-type: none"> Confidentialité des données Conformité réglementaire Confiance des utilisateurs
5	Système en panne	les interfaces de programmation d'application (API) utilisées par l'orchestrateur n'arrive pas à communiquer	Opérationnel	5	5	25	<ul style="list-style-type: none"> Disponibilité du système Continuité du service

FIGURE 4.13 – Table d'évaluation et analyse des risques

N°	Plan d'action				Méthode de Surveillance	Efficacité
	Risque	Réponse au Risque	Action	Responsable		
1	Qualité des données	Réduire	<ul style="list-style-type: none"> • Utiliser des techniques de prétraitement • Augmentation des données 	Omaïma	fait	Surveillance continue de la qualité des images ✓
2	Complexité algorithmique	Optimiser	<ul style="list-style-type: none"> • Développement d'une architecture modulaire • Utilisation d'algorithmes de segmentation efficaces • Optimisation des hyperparamètres du modèle 	Omaïma	Fait	Analyse des performances des modèles ✓
3	Compatibilité des formats de fichiers	Atténuer	<ul style="list-style-type: none"> • Développement des modules de conversion robustes • Utilisation des solutions techniques comme threading 	Céline	Fait	Tests réguliers des conversions de format ✓
4	Sécurité des données	Protéger	<ul style="list-style-type: none"> • Anonymisation des données • Mettre en œuvre des protocoles sécurisés pour le transfert des données 	Technicien Informatique	Fait	Audits de sécurité et revues périodiques ✓
5	Problèmes d'API (Système en panne)	Prévenir	<ul style="list-style-type: none"> • Explorer d'autres choix techniques alternatifs • Travailler en collaboration avec l'équipe informatique pour obtenir plus de flexibilité 	Omaïma, technicien Informatique	En cours	Surveillance des logs API et des temps de réponse ✓

FIGURE 4.14 – Plan d'action pour la surveillance des risque identifiées au cours du projet