

Projet tutoré FIE4 2021-2022 :

Production d'un outil d'aide au diagnostic médical

Par FREDERICK Tristan, GAY Colin et DA COSTA Benjamin

Table des matières

I.	Présentation générale	2
A.	Les différentes parties prenantes	2
B.	Le prestataire	2
C.	Le projet	2
II.	Analyse des besoins	5
A.	Cas d'utilisation	5
1.	Scénario de cas d'utilisation	5
B.	Modèle de données	6
1.	Version 1	6
2.	Version 2	7
C.	Maquettes	7
III.	Veille technologique	9
A.	Utilisation d'un système expert	9
B.	Aide au diagnostic (aide à la décision)	10
C.	Apprentissage par renforcement	10
D.	Langages de programmation possibles/utiles	10
E.	Partie analyse : qu'allons-nous utiliser ?	11
F.	Technologies intéressantes	11
IV.	Élément de gestion de projet	12
A.	Moyens (humains, techniques)	12
B.	Difficultés rencontrées	13
V.	Conclusion	15
VI.	Bibliographie	16
	Annexes	17

I. Présentation générale

A. Les différentes parties prenantes

Le projet Uv'Helper consiste en une application d'aide au diagnostic de l'uvéite pour les ophtalmologistes du CHU de Nice, dont fait partie le maître d'ouvrage de ce projet, le Dr Aïda DRIRA. Ce dernier sera réalisé par un groupe de trois étudiants en 2^e année du cycle ingénieur de l'école d'ingénieurs ISIS à Castres, composé de Benjamin DA COSTA, Tristan FREDERICK et Colin GAY. Afin d'assurer la bonne réalisation de ce projet, les étudiants seront sous la tutelle de Monsieur Elyes LAMINE, maître de conférences à l'Institut National Universitaire Champollion ainsi que Monsieur Rémi BASTIDE, professeur à ISIS.

B. Le prestataire

Ce projet a été proposé par le Centre Hospitalier Universitaire de Nice. Les activités de ce centre sont réparties entre 8 établissements de santé : 5 hôpitaux, 2 instituts et 1 hôpital pédiatrique. Il a été reconnu comme CHU après la création de la faculté de médecine de Nice en 1973.

Les 3 missions de ce CHU sont les soins (diagnostic, éducation, prévention), la recherche et l'enseignement. En 2021, le nombre de lits du CHU était de 1660.

Actuellement, le directeur général est Guepratte Charles, son rôle est d'organiser la politique générale de l'établissement et il gère les recettes et les dépenses de son CHU.

C. Le projet

L'uvéite concerne 1 français sur 1000, il s'agit d'une inflammation des tissus intra-oculaires situés dans les couches profondes du globe oculaire (iris, choroïde, rétine et ses vaisseaux).

Plusieurs symptômes comme la rougeur des yeux, des douleurs oculaires, une baisse de la vision et de la photophobie sont des signes potentiels d'une uvéite.

Le projet consiste à assister des ophtalmologistes ou des orthoptistes dans le diagnostic étiologique de l'uvéite. Dans le cadre de ce projet, nous devrons réaliser une application qui proposera au clinicien une liste de critères à remplir (oui, +/-, quantification, non, non renseigné) pouvant orienter vers une étiologie particulière. Les critères seront nombreux afin d'avoir les résultats les plus fiables ; parmi les différents critères à prendre en compte, il y a le sexe, l'âge, les antécédents connus, les allergies, la douleur, la pression intra-oculaire, l'acuité visuelle initiale, la quantification de l'hyperhémie conjonctivale, l'œdème de cornée, les types de précipités retro-descemétiques, etc.

Sur la base de ces critères, notre application devra générer des propositions d'étiologie et une banque d'imagerie-type. L'application pourra aussi proposer des examens complémentaires ainsi qu'une littérature sur le traitement et l'évolution habituelle de cette inflammation.

L'application permettra ainsi un premier diagnostic en mettant les informations en lien avec des données issues de banques cliniques d'ophtalmologie ce qui fera gagner énormément de temps aux médecins. En somme, notre application devra servir d'aide au diagnostic médical, il s'agit d'un système d'aide à la décision médicale utilisé dans le processus du diagnostic médical, qui est composé d'une base de données et d'un moteur de recherche. Ce moteur de recherche donne accès aux informations de la base de données, et peut proposer des diagnostics différentiels en fonction de données pré-renseignées après un examen clinique, des estimations pronostiques, ou signaler des informations manquantes pour établir un diagnostic.

Les systèmes d'aide à la décision médicale (SADM) sont des applications informatiques dont le but est de fournir aux cliniciens en temps et lieux utiles les informations décrivant la situation clinique d'un patient ainsi que les connaissances appropriées à cette situation, correctement filtrées et présentées afin d'améliorer la qualité des soins et la santé des patients. De nombreuses applications d'aide au diagnostic existent déjà comme Axilios, AideDiag ou encore Medvir.

Par exemple, Medvir se sert de l'intelligence artificielle pour l'aide à la décision et est implanté dans les centres hospitaliers et d'autres établissements de santé. C'est le fruit de plus de 30 ans de recherche et l'application a été créée à partir du langage des patients et de la médecine basée sur les faits. Ce qui est important dans le fonctionnement de Medvir c'est qu'ils utilisent la fuzzy logic (ou logique floue) : c'est une logique qui permet d'étendre la logique booléenne classique avec des valeurs de vérités partielles car les variables ne varient pas entre vrai et faux mais elles oscillent entre 0 et 1.

De plus, notre application devra permettre la création de nouvelle étiologie. Après l'ajout de règles correspondantes à nos critères, l'application devra être capable de proposer la nouvelle étiologie suite aux tests.

L'application devra également permettre l'authentification des cliniciens afin que les données enregistrées soient protégées.

Notre objectif est donc de réaliser l'application la plus complète possible. Celle-ci devra permettre d'orienter les médecins vers la bonne étiologie ou le bon groupe de maladie après avoir rempli le questionnaire sur les différents critères. Pour ce faire, nous allons devoir mettre en place notre base de données avec les différents critères des différentes étiologies. Cela permettra de commencer l'identification de la maladie grâce à des critères simples qui auront été demandés au patient ou constatés par le spécialiste.

Dans un premier temps, nous définirons l'architecture fonctionnelle et technique de l'application puis dans un second temps nous développerons l'application sous forme d'une preuve de concept, le socle de base de cette application, à savoir le générateur de propositions d'étiologies.

II. Analyse des besoins

A. Cas d'utilisation

Dans un premier temps, nous avons estimé que seuls les cliniciens étaient acteurs de l'application. Leur rôle est de définir la cause d'une uvéite pour en déduire la manière de soigner cette maladie. Cette pathologie étant rare et les causes possibles étant nombreuses et variées, il est très difficile pour un ophtalmologiste de définir ces causes précisément.

Notre application devra donc pouvoir remplir ce rôle afin de faciliter le travail des spécialistes.

Afin d'être le plus efficace possible, le docteur DRIRA nous aide en triant les différentes maladies qui pourraient causer l'apparition d'une uvéite et les moyens de les traiter propres à chacune de ces maladies.

L'application devra ainsi prendre en compte de nombreux facteurs tels que le sexe, l'âge, les antécédents du patient et une multitude d'autres afin de réaliser le diagnostic le plus précis possible.

1. Scénario de cas d'utilisation

Une patiente d'une cinquantaine d'années vient voir son ophtalmologiste car elle ressent une douleur sur son œil gauche, présente des rougeurs et elle subit une perte de vision. Après consultation le spécialiste donne son diagnostic : la patiente souffre d'une uvéite. Cette pathologie étant rare avec une multitude de causes différentes, le clinicien ne peut pas directement connaître les étiologies qui ont conduit à cette uvéite. Il se rend alors sur notre application afin d'être mieux renseigné sur la maladie de sa patiente. Il s'authentifie, et est dirigé sur la page d'accueil où il peut avoir un aperçu rapide de sa liste de patients. Il crée donc le profil de cette nouvelle patiente en rentrant les informations nécessaires à la reconnaissance de cette dernière pour de futures consultations (initiales, date de consultation). Le profil patient étant créé, le clinicien peut maintenant rentrer les symptômes dont la patiente souffre. Pour ce faire, il rentre chaque signe important dans l'application à travers une barre de recherche et sélectionne la prédiction de critère la plus appropriée. Une fois le premier signe rentré, il sera pris en compte dans la recherche des étiologies possibles et s'affiche sur l'écran avec un espace dédié à l'insertion de la valeur associée. Le médecin inclut donc tous les signes à prendre en compte dans le diagnostic des étiologies et leurs valeurs. L'application le met sur la piste d'une uvéite antérieure qui serait due à une infection par le HSV (Virus Herpès Simplex), indique quels sont les symptômes marquants de cette pathologie et le conseille ainsi sur les examens complémentaires à réaliser mais

aussi à des solutions pour le traitement de ce virus qui a entraîné l'apparition de l'uvéite de notre patiente ([Figure 1](#)).

B. Modèle de données

1. Version 1

Le but est d'avoir un ensemble des symptômes possibles pour les mettre à disposition du clinicien et ainsi, l'afficher au moyen de liste déroulante. Dans la même idée, on propose les deux sexes pour les intégrer au moyen d'options en HTML.

Ce modèle s'est révélé obsolète pour plusieurs raisons.

Premièrement, un identifiant patient sera plus complet qu'un simple String initialement pensé pour être un ensemble nom/prénom. En effet, quand il s'agit de données de santé, les établissements hospitaliers n'ont pas le droit de conserver des informations pouvant identifier clairement les patients par rapport aux normes RGPD. Ainsi, il faut trouver un moyen de créer un identifiant patient clair pour les docteurs tout en respectant ces normes. L'identification devra être pratique et simple pour le clinicien et donc éviter des confusions au niveau du renseignement de critères et de diagnostic. Plusieurs idées sont à l'étude, utiliser l'id généré dans la base de données à savoir un numéro, cependant, cela ne serait pas intuitif pour le clinicien et cela peut être jugé déshumanisant de caractériser des personnes par des nombres. Il a également été proposé par le Docteur DRIRA de renseigner les initiales accompagnées de la date de consultation. Grâce à cette méthode, les cliniciens auront un premier repère avec les initiales et pourront consulter le registre des rendez-vous pour s'assurer que les initiales accompagnées de la date de consultation correspondent bien avec les patients pris en charge lors des rendez-vous. Cette solution sera à privilégier.

Deuxièmement, il n'y aura pas uniquement trois symptômes à renseigner mais un certain nombre variable d'un diagnostic à l'autre, il faudra donc implémenter une liste de critères au sein d'un diagnostic.

Enfin, l'attribut « date » peut être ambigu. Initialement prévu pour représenter la date de naissance du patient, il est apparu comme la date de consultation pour le Docteur DRIRA qui souhaite donc l'intégrer dans la solution d'identification des patients.

Autre problème posé par la date de naissance, comme pour le nom et le prénom, cette caractéristique est présentée comme une donnée d'identification, elle ne peut donc pas être utilisée dans le cadre de cette application ([Figure 2](#)).

2. Version 2

Suite à la deuxième réunion avec la cliente, nous avons mis au point cette deuxième version du modèle de données, elle ne limite plus le nombre de critères (anciennement symptômes) par diagnostic à 3 mais à une liste de critères dans les diagnostics. On dispose toujours de l'énumération des deux sexes disponibles. De plus, on a également précisé ce qu'étaient les identifiants patients, un clinicien devra préciser les initiales du patient ainsi que sa date de consultation pour lui permettre de le reconnaître. Pour finir, on a également créé une classe étiologie qui, au même titre que la classe critère, recensera les étiologies dont nous disposerons.

Après des discussions avec la cliente, la volonté d'avoir un système d'authentification se révèle de plus en plus nécessaire, il nous faudra donc prévoir un espace dans la base de données dédiés aux cliniciens, cependant, nous devons connaître les informations utiles pour les décrire, comment les identifier, s'il existera diverses autorisations d'accès à la base de données, etc. Le manque de possibilité de désigner des critères comme des critères parents de certains critères et donc d'avoir des critères enfants et aussi nécessaire et à ajouter lors d'une prochaine version ([Figure 3](#)).

C. Maquettes

Présentation de notre maquette version 1 :

<https://drive.google.com/drive/u/0/folders/1XUSDB56vwRrIsIMDbV22BvUt6dqZL0kR>

Cette maquette a finalement reçu un accueil assez neutre, les fonctionnalités de prédictions censées être présentes dans le livrable final ont été démontrées avec notre vision. Les graphismes n'ont pas du tout été abordés car absents de la maquette (pas de logo, pas de nom d'application, pas de charte graphique).

Il en résulte qu'il y a des points à améliorer. L'intuitivité de l'application n'est pas évidente, il y a de nombreux critères à rentrer, et non uniquement trois comme nous l'avons présenté précédemment, ce qui a été proposé est de mettre à l'écran des critères qui doivent obligatoirement être remplis par le clinicien peu importe la pathologie puis de compléter les critères à remplir par une ou plusieurs recherches de critères pour que le clinicien ne sélectionne que les critères qu'il a constatés. Il faudra également créer un moyen optimisé pour afficher les sous-critères d'un critère mais également de ne plus les afficher s'il n'y en a pas le besoin pour une quelconque raison.

Pour ce qui est des remarques de notre tuteur école, il nous a indiqué qu'il faudrait présenter un design ou du moins des prémices et ne pas délaisser cette partie du maquettage. Les maquettes manquaient de fond et de formes, elles ne permettaient pas une intuitivité évidente à l'utilisateur, il faudra donc mettre en place des fonctionnalités particulières qui peuvent être intéressantes pour les utilisateurs telles que la recherche de critères, l'ajout rapide de sous-critères si besoin et des interfaces plus pratiques pour l'utilisateur par rapport à la vie réelle, la cliente a

indiqué qu'elle aimerait pouvoir désigner l'œil touché comme si elle avait le patient en face d'elle, c'est une piste à exploiter pour la future maquette.

Enfin, il ne faut pas afficher de probabilité pour l'estimation des étiologies, il est impossible d'afficher une estimation fiable des étiologies, on se contentera de les classer de la plus probable à la moins probable.

Présentation de notre maquette version 2 :

<https://drive.google.com/drive/u/0/folders/1XUSDB56vwRrlsIMDbV22BvUt6dqZL0kR>

Cette maquette a reçu un accueil plus positif que la première, la présentation a été plus complète que la première version, cependant, les critères prennent encore trop de temps à compléter, il faudra présenter les critères pour chaque œil dans un format spécifique rappelant les fiches que complètent les praticiens actuellement.

Lorsqu'un clinicien entrera l'œil ou les yeux touchés, il devra ensuite indiquer la région touchée (panuvéite, antérieure, etc.) et l'application devra griser les cellules des critères non-importants, par exemple, pour une uvéite antérieure, il y aura des critères qui ne seront pas marquants de l'uvéite et qui donc seront moins importants. La cellule devra donc être grisée mais il faudra laisser la possibilité au clinicien de renseigner la valeur du critère malgré tout.

La question du design rentrera plus en ligne de compte lorsque nous aurons trouvé le moyen de créer les règles pour être fidèle aux pourcentages indiqués dans les données transmises par le Docteur DRIRA.

III. Veille technologique

A. Utilisation d'un système expert

Il s'agit d'un outil d'aide à la décision.

À l'aide d'un raisonnement effectué à l'aide de faits et de règles, le logiciel va être capable de répondre à une question donnée. Il va être capable de reproduire le raisonnement d'un être humain. Un système expert est composé de 3 parties : une base de faits, une base de règles et un moteur d'inférence.

Le moteur d'inférence est capable d'utiliser la base de faits et celle de règles pour produire de nouveaux faits, et va parvenir jusqu'à prévoir la réponse à la question experte posée. L'utilisation de la logique formelle va permettre à ces systèmes d'utiliser un raisonnement déductif. La règle d'inférence principale est : Si le fait P est vrai et que P implique la règle Q alors, Q est vrai (nouveau fait).

Le système expert va utiliser un ensemble de réponses vraies ou fausses afin d'obtenir un résultat. Mais pour obtenir ce résultat, le système expert doit avoir un moteur d'inférence qui applique les règles afin de prévoir des résultats avec les connaissances déjà connues

Il existe 3 méthodes de fonctionnement :

- Le moteur à "chaînage avant" qui cherche à déterminer les informations recherchées en utilisant les faits et les règles qu'il connaît. C'est une méthode de déduction qui applique des règles en partant des prémisses pour en déduire de nouvelles conclusions. Ces conclusions enrichissent la mémoire de travail et peuvent devenir les prémisses d'autres règles. L'un des avantages du chaînage avant sur le chaînage arrière est que la réception de nouvelles données peut déclencher de nouvelles inférences, ce qui rend le moteur plus adapté à des situations dynamiques dans lesquelles les conditions sont susceptibles de changer.
- Le moteur à "chaînage arrière" qui cherche à déterminer les faits en utilisant les règles qu'il connaît et les objectifs qu'il doit obtenir.
- Le moteur à "chaînage mixte" est une combinaison des 2 autres méthodes.

Les données médicales données par le docteur vont donc nous permettre de créer nos règles et les différents critères entrés par le clinicien deviendront nos faits ce qui nous permettra d'orienter les médecins vers la bonne étiologie.

Cette technique va nous permettre de reproduire le comportement d'un être humain à l'aide d'un ensemble de règles et de fait ce qui est utile dans notre projet où le Machine Learning est inutile dû au faible nombre de données.

B. Aide au diagnostic (aide à la décision)

Une aide au diagnostic médical est un système d'aide à la décision médicale utilisé dans le processus du diagnostic médical, qui est composé d'une base de données et d'un moteur de recherche. Ce moteur de recherche donne accès aux informations de la base de données, et peut proposer des diagnostics différentiels en fonction de données pré-renseignées après un examen clinique, des estimations pronostiques, ou signaler des informations manquantes pour établir un diagnostic. Il existe plusieurs applications qui font de l'aide au diagnostic (AideDiag, Axilios). Les systèmes d'aide à la décision médicale (SADM) sont des applications informatiques dont le but est de fournir aux cliniciens en temps et lieux utiles les informations décrivant la situation clinique d'un patient ainsi que les connaissances appropriées à cette situation, correctement filtrées et présentées afin d'améliorer la qualité des soins et la santé des patients.

C. Apprentissage par renforcement

En IA, l'apprentissage par renforcement consiste, pour un agent autonome, à apprendre les actions à prendre, à partir d'expériences, de façon à optimiser une récompense quantitative au cours du temps. L'agent est plongé au sein d'un environnement, et prend ses décisions en fonction de son état courant. En retour, l'environnement procure à l'agent une récompense, qui peut être positive ou négative. L'agent cherche, au travers d'expériences itérées, un comportement décisionnel optimal, en ce sens qu'il maximise la somme des récompenses au cours du temps. Plus l'agent sera confronté à diverses expériences, plus il sera efficace dans les décisions qu'il prendra. L'application de cet apprentissage se fait dans plusieurs domaines comme la robotique, la gestion de ressource, le vol d'hélicoptères et même la chimie.

D. Langages de programmation possibles/utiles

Pour le frontend, nous utiliserons HTML/CSS qui est nécessaire pour la création du site web et la gestion de la mise en page de ce dernier. Nous allons certainement ajouter du code en JavaScript afin d'ajouter des fonctionnalités dynamiques aux pages du site web

En ce qui concerne le traitement de nos données, il y a plusieurs langages informatiques possibles que nous pourrions utiliser : PHP, Java, Python ou C/C++/C#.

C/C++/C# : aucune connaissance par l'équipe et ne semble pas correspondre aux besoins assez basiques du projet.

PHP : fonctionne très bien avec les bases de données mais nous n'en aurons pas si le client trouve ça inutile. Aucune connaissance actuelle de la technologie.

Java : application sur ordinateur mais utilise beaucoup de mémoire. Bonne connaissance de la technologie de la part de l'équipe.

Python : nombreux outils et fonctionnalités qui facilitent la programmation mais plus long pour les scripts compilés. Cela pourrait nous permettre d'utiliser des fonctions d'intelligence artificielle si le client veut intégrer cette fonctionnalité dans l'application. Assez bonne connaissance de la technologie par l'équipe et peut-être simplement appréhendé.

Après réflexion, nous avons pensé que l'utilisation de Python serait la plus optimale d'une part pour notre projet tuteuré, mais également dans l'optique d'autres améliorations avec par exemple un modèle de Machine Learning qui pourrait être implémenté une fois qu'il y aurait une quantité d'informations suffisantes.

E. Partie analyse : qu'allons-nous utiliser ?

Comme nous l'avons vu plus haut, l'un des avantages du chaînage avant sur le chaînage arrière est que la réception de nouvelles données peut déclencher de nouvelles inférences, ce qui rend le moteur plus adapté à des situations dynamiques dans lesquelles les conditions sont susceptibles de changer. Dans notre cas, le nombre d'étiologies de l'uvéite sont tellement nombreuses que nous serons obligés d'ajouter des données, ce qui nous dirige à utiliser le moteur de chaînage avant. De plus, le besoin du client est de déduire les étiologies à partir des critères et du type d'uvéite observés, ainsi ce mode de fonctionnement est le principe même du chaînage avant.

F. Technologies intéressantes

Nous nous sommes renseignés sur l'utilisation de SysExpert : un programme développé sur Python permettant la mise en place de système expert et donc de moteur de règle par chaînage avant ou arrière sur une base de connaissances.

<https://github.com/theodeze/SystemeExpert>

Cet exemple illustre assez bien ce que nous cherchons et semble illustrer notre besoin, le seul problème pourrait être le manque d'activité autour de ce programme, il n'a reçu que deux versions en deux jours d'intervalle en Novembre 2018 et peut avoir été entamé mais laissé à l'abandon, c'est pour cela qu'il nous faut explorer d'autres possibilités au cas où cette solution soit incomplète.

Nous avons également effectué des recherches sur Prolog : un langage de programmation qui permet de créer des règles logiques pouvant être utilisé pour créer des règles métiers. Ce langage peut être implémenté grâce à Pyke, un programme également développé en Python permettant de créer des moteurs d'inférences à partir de Prolog.

Pyke possède sa dernière mise à jour en 2013 et peut ne plus être adapté aux besoins actuels ou être devenu obsolète ou du moins, avoir perdu en efficacité.

IV. Élément de gestion de projet

Comme montré dans la Figure 4 peut remarquer que dans ce cas de gestion de projet, le projet est le rapport que nous devons présenter à la fin du 1er semestre, cette remarque nous a été faite par notre tuteur-école lors d'une réunion avec le client. En effet, le client n'est pas intéressé par notre rendu de rapport, son intérêt se porte dans les tâches que l'on va mettre en œuvre pour développer l'application qui répondra à son besoin. Nous avons donc dû recentrer la gestion de projet sur le projet lui-même, à savoir l'application d'aide au diagnostic des uvéites. Suite à ces remarques, nous avons créé le suivi de projet suivant pour l'analyse des besoins avec une vue journalière (Figure 4 et 5).

Petite précision concernant les tâches nommées « réunion », elles ne se déroulent pas sur une journée entière mais sur des créneaux horaires précis pour une durée située entre 1h et 2h.

Ainsi que ce suivi de projet pour la deuxième phase du projet qui consistera à créer l'application web avec une vue hebdomadaire (Figure 6).

A. Moyens (humains, techniques)

Les moyens humains mis en place pour la réalisation de ce projet sont l'ensemble des 3 élèves-ingénieurs Benjamin DA COSTA, Tristan FREDERICK et Colin Gay, mais aussi le Dr DRIRA et le professeur Elyes Lamine qui nous accompagnent dans l'avancement de ce projet.

Les moyens techniques sont nos ordinateurs personnels avec des logiciels tels qu'Adobe xd pour la création de maquettes ou encore Trello pour organiser la gestion de projet. Nous avons également les ordinateurs mis à disposition dans notre école avec certains logiciels adaptés à nos besoins tels que MagicDraw pour la conception du diagramme des cas d'utilisations et du modèle de données ainsi que les notions de programmation que nous avons apprises durant notre cursus. A compter du 09/12/2021, nous avons eu accès à un premier exemple d'interactions entre les données. Il s'agit d'un fichier Excel (Figure 7) recensant les types d'uvéites, le groupe d'étiologies de l'étiologie concernée puis cette dernière et ensuite les différents critères qui seront à renseigner par le clinicien.

Une approche sous forme de graphe pourrait être adaptée pour la réalisation de l'application.

B. Difficultés rencontrées

Le projet tuteuré est arrivé en même temps que de nombreuses autres matières et le rythme du semestre s'est nettement accéléré à partir de début Novembre, il a donc fallu conjuguer l'emploi du temps chargé et les révisions de partiels avec le projet tuteuré.

Étant donné qu'il s'agit du premier projet que nous devons réaliser pour un client, cela implique également une responsabilité supplémentaire de qualité au niveau de la compréhension des besoins du client, en s'assurant que l'on comprend bien ces derniers.

Il faut également découvrir des technologies qui peuvent être utiles à la réalisation du projet et enfin choisir la technologie la plus adaptée.

Ensuite, le plus important est de réaliser le projet selon le souhait des clients et de bien le réaliser. Pour cela, il faut avant tout que les fonctionnalités primaires et qui font l'utilité de l'application soient effectives et efficaces, en plus de cela, il faut y ajouter une interface simple d'utilisation, intuitive et ergonomique afin que le client puisse utiliser l'application sans perdre de temps ni pour comprendre son fonctionnement, ni pour l'utiliser. Il faut donc se renseigner sur les moyens possibles pour améliorer tous ces points et discuter avec le client pour connaître leur préférence sur les propositions que nous allons leur faire parmi notre sélection et trouver le moyen de les implémenter. Enfin, il est nécessaire de produire une application possédant un graphisme de qualité afin de développer l'appétence du client à utiliser notre application.

Il en résulte que ce n'est pas évident pour des étudiants de connaître directement les outils adaptés aux besoins du client, de ce fait, il est nécessaire de découvrir les outils d'aide au diagnostic déjà existants. Pour cela, il faut donc découvrir tout un vocabulaire issu d'un domaine métier particulier, mais aussi les technologies qui peuvent y être associées et utilisées dans le cadre de notre projet.

De plus, il est également compliqué de comprendre parfaitement le besoin du client, deux imaginations (celle du groupe et celle du client) se concurrencent pour la modélisation et la réalisation de l'application. Le groupe doit alors se plier aux exigences du client et quand bien même on pense avoir trouvé une bonne idée, cette dernière peut être balayée car elle serait inutile pour le client. Il est donc important de prendre le plus de réunions possibles avec le client pour orienter le groupe vers la bonne solution. Ce dernier point n'est pas forcément évident à mettre en place au vu des emplois du temps des deux parties. Nous avons pu en effectuer trois, ce qui nous semble avoir permis d'avoir une bonne idée de l'application à organiser, cependant, si nous avions eu plus de temps pour se consacrer au projet, nous aurions pu en réaliser quelques-unes de plus afin d'avoir la meilleure orientation possible.

Cette somme de recherches à effectuer pour nous faire découvrir de nouvelles notions n'est pas forcément évidente à conjuguer avec les autres matières que nous

avons à réviser, cela peut porter à confusion à certains moments et donc amener à des incompréhensions ou des malentendus.

Par moment, il y a également eu des erreurs de communication, une réunion était prévue le Lundi 13 Décembre à 18h, cependant le mail du lien de la réunion étant envoyé trop tard (environ 15 minutes avant le début de la réunion), le client a pensé que la réunion était annulée et donc cette réunion n'a pas pu avoir lieu. Désormais, l'équipe devra s'assurer de fournir les informations nécessaires au bon déroulement des futures réunions plus tôt afin d'éviter ce genre de situations.

La réalisation de l'approche de l'application nous a également causé des problèmes, nous nous attendions à devoir effectuer des choix binaires pour organiser les règles du moteur d'inférence, cependant, l'approche a dû être repensée car ne correspondait pas aux besoins du projet. Pour ce dernier, nous devons créer des chemins mais qui ne vont pas éliminer totalement l'étiologie si le critère ne correspond pas. Il nous faudra donc plus de temps lors du 2nd semestre pour effectuer cette représentation

V. Conclusion

Après avoir effectué des recherches sur l'uvéite, ses symptômes et ses traitements, nous nous sommes vite rendu compte de plusieurs points importants.

Tout d'abord, la rareté d'apparition de l'uvéite (environ une dizaine de cas par an) ne facilite pas le diagnostic de cette pathologie, ce qui inclut que notre application ne nécessitera pas de Machine Learning au vu du peu de données qui seront rentrées dans l'application.

Nous allons donc devoir mettre en place un système expert ainsi qu'un moteur de règles afin de satisfaire au mieux les besoins de notre client et ainsi faciliter l'aide au diagnostic.

Notre outil devra donc permettre d'orienter le médecin sur les bonnes étiologies à l'aide des critères indiqués avant de lancer le test. Dans le cas où nous n'arrivons pas à départager plusieurs étiologies, le type d'étiologie (infections, maladie de système, ...) sera affiché ainsi que les étiologies pour lesquelles il y a une hésitation. Sinon, les trois étiologies les plus probables ce qui aidera le clinicien et lui fera donc gagner du temps.

VI. Bibliographie

<https://www.chu-nice.fr/>

[Moteur d'inférence : Définition, Exemples, Fonctionnement](#)

[Dompter le domaine grâce aux moteurs de règles](#)

[Système expert : définition, fonctionnement et exemples](#)

[Principe de fonctionnement de MEDVIR](#)

[Code SysExpert](#)

[Base de compréhension Pyke](#)

<https://github.com/theodeze/SystemeExpert/tree/master/sysexpert> : exemple d'utilisation de système expert

https://drive.google.com/drive/folders/1Z_Naa41OXFRYHv4X9-rIEq25mD83Ruv8?usp=sharing : lien contenant nos différentes annexe (vidéos de création de la maquette + référence bibliographique utilisée par le Docteur DRIRA.

Annexes

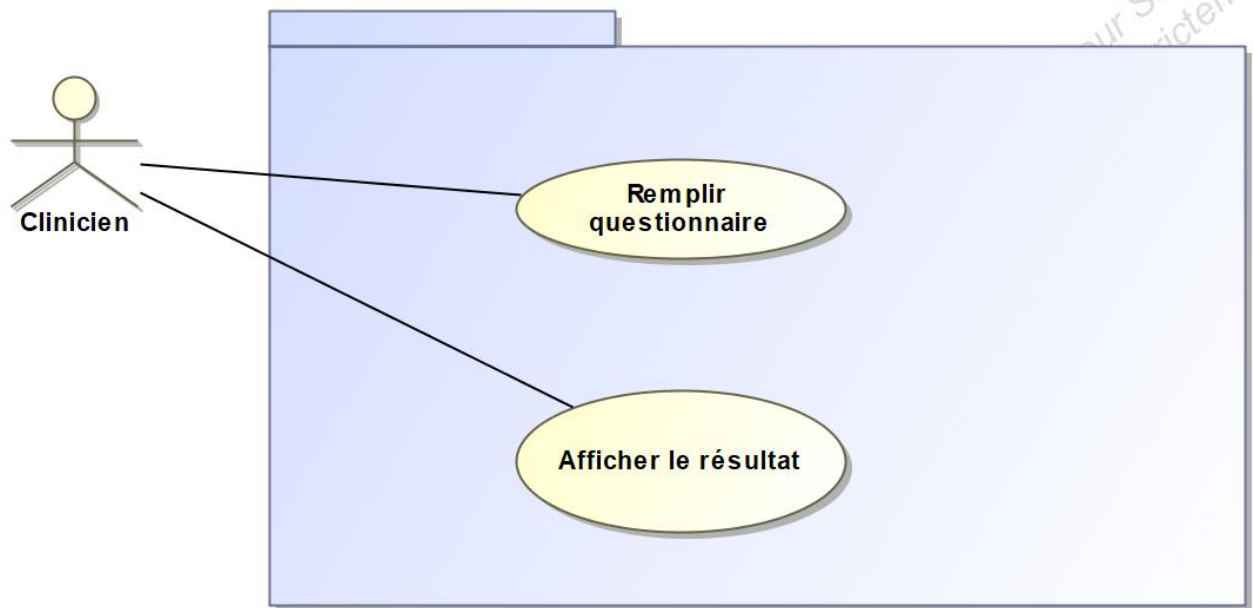


Figure 1 : Modèle de cas d'utilisation de notre application

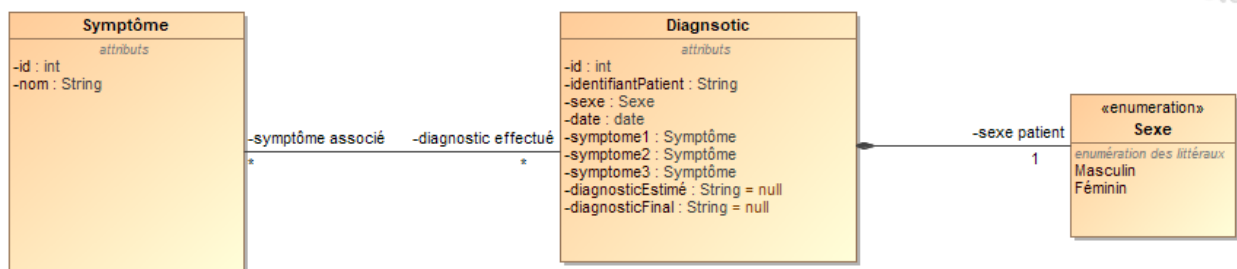


Figure 2 : Première version de notre modèle de données

Compte rendu projet FIE 4 – Aide au diagnostic pour les uvéites

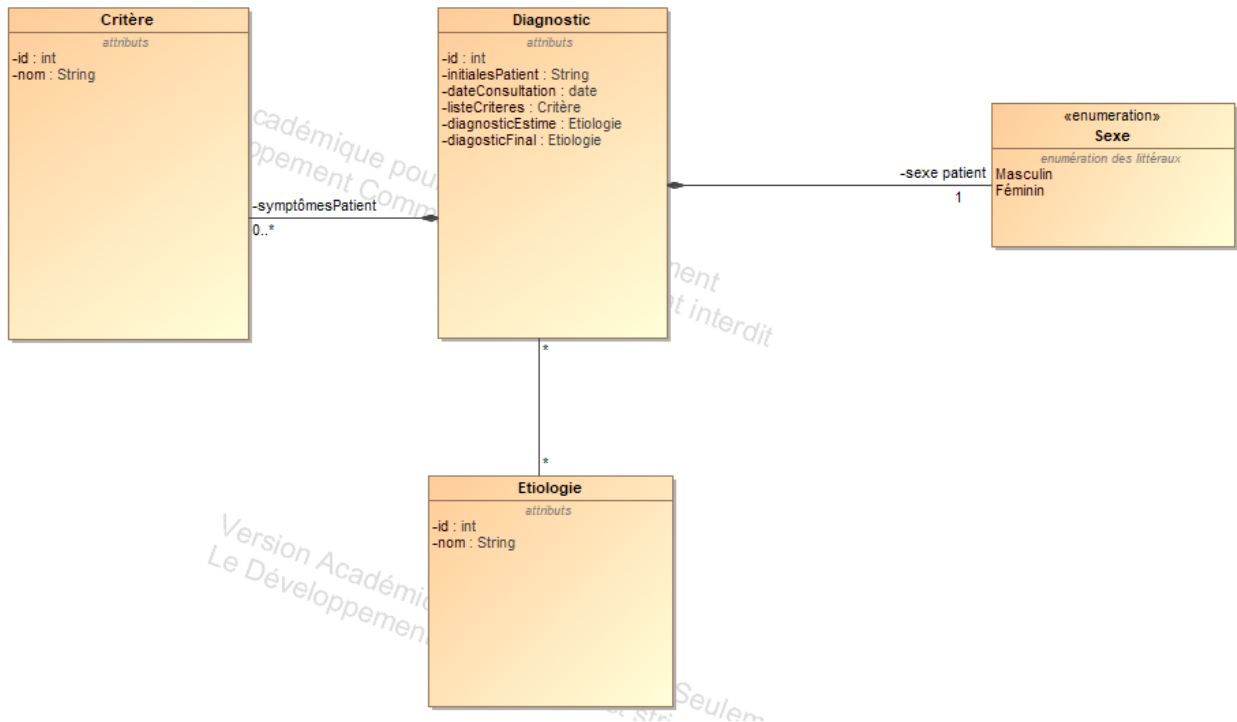


Figure 3 : Seconde version de notre modèle de donnée

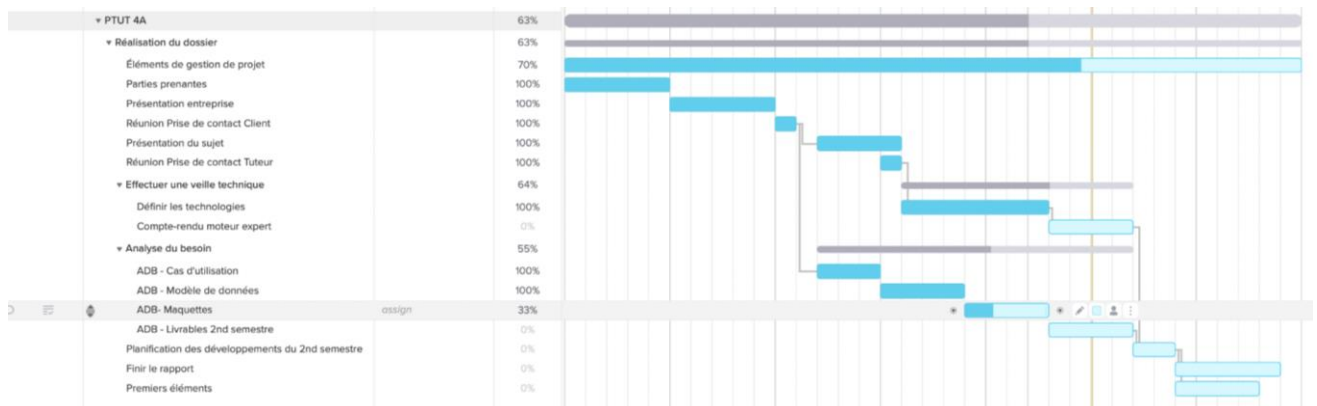


Figure 4 : Première version de notre gestion de projet sur Trello

Compte rendu projet FIE 4 – Aide au diagnostic pour les uvéites

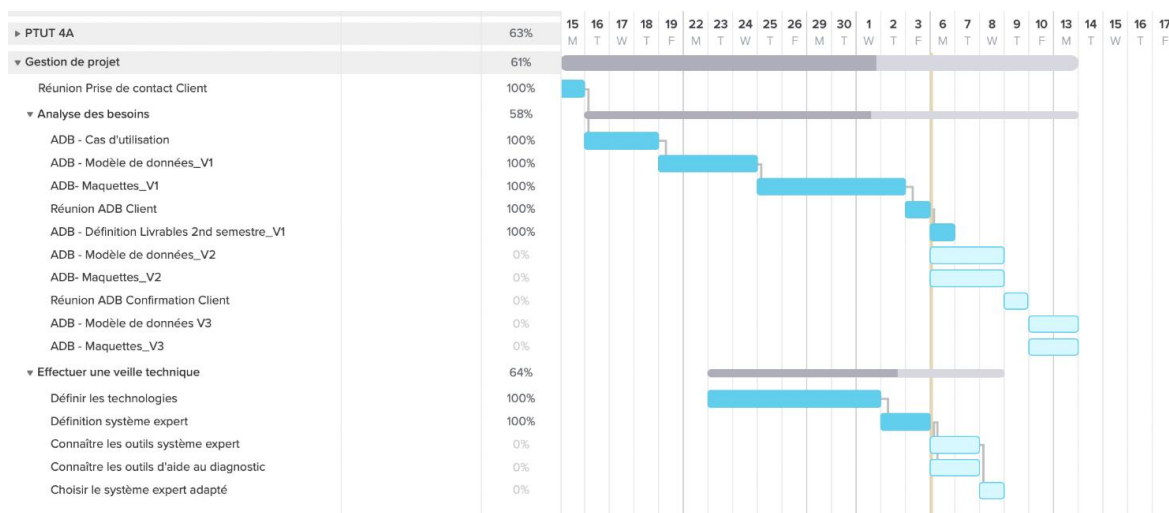


Figure 5 : Suivi de l'analyse des besoins sur Trello

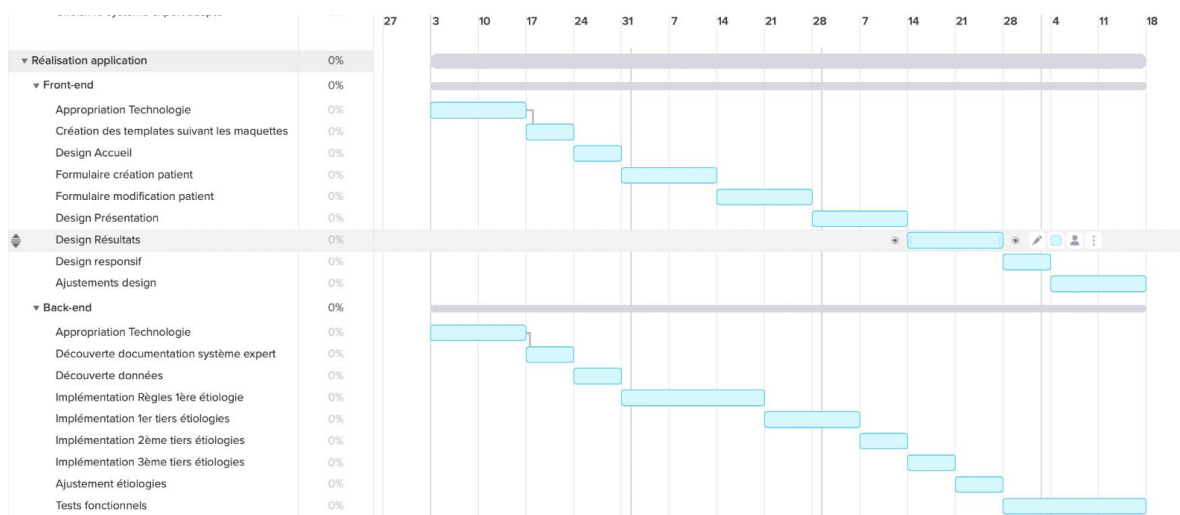


Figure 6 : Prévisions des tâches durant la seconde phase du projet

	A	B	C	D	E	F	G	H	I	J	K
1			Etiologie	Genre	Age	Ethnie	Allergies	Antécédents	Extra oculaire	Œil touché	Mode
2	Antérieure	Infections	HSV	Femme 58%	44 médian	Blanc 65 %				unilatéral 99%	aigu 1
3	Antérieure	Infections	VZV	femme 60%	63 médian	blanc 72%,		immunodépression 8%	eruption dermatome 86%	unilatéral 96%	aigue
4	Antérieure	Infections	Tuberculose								
5	Antérieure	Maladie de système	Bactéries (lyme, rickettsies)								
6	Antérieure	Maladie de système	HLA B27								
7	Antérieure	Maladie de système	Arthrite chronique juvénile								
8	Antérieure	Maladie de système	Sarcoidose								
9	Antérieure	Maladie de système	Behcet								
10	Antérieure		SEP								
11	Intermédiaire	Maladie de système	Iridocyclite de Fuchs	50/50	35 médian	blanc 75%				unilatéral 98%	aigu (
12	Intermédiaire	Maladie de système	SEP								
13	Intermédiaire	Infections	Sarcoidose								
14	Intermédiaire	Infections	lyme								
15	Intermédiaire	Infections	tuberculose								
16	Postérieures	Infections	rickettsies								
17	Postérieures	Infections	rétinchoroïdite toxoplasmique								
18	Postérieures	Infections	syphilis								
19	Postérieures	Infections	whipple								
20	Postérieures	maladie de système	tuberculose								
21	Postérieures	maladie de système	behcet								
22	Postérieures	maladie de système	sarcoidose								
23	Postérieures	maladie de système	rétinochoroïdite de birdshot								
24	Postérieures	maladie de système	syndrome des tâches blanches évanescences								
25	Panuvéite	maladie de système	onhthalmia suppurativa								

Figure 7 : Capture d'écran du fichier Excel envoyé par le Docteur DRIRA