

UNIVERSITE ASSANE SECK DE ZIGUINCHOR



L'excellence, ma référence

UFR DES SCIENCES ECONOMIQUES ET SOCIALES

Département Economie – Gestion

Mémoire de Master

Mention : Management des systèmes d'information

Spécialité : Management des Systèmes d'Information Automatisés (MSIA)

Sujet :

**Mise en place d'un espace familial de gestion de données
médicales dans un Guichet Médical Unique (GMU)**

Présenté par :

Mme Fatoumata DIALLO

Le 23 Décembre 2023 à la salle CISCO de L'UASZ

Sous l'encadrement et la supervision respective de :

| | | | |
|-----------------|-------|---------------------------------|------|
| Dr Papa Alioune | CISSE | Maitre de conférences titulaire | UASZ |
| Pr Melyan | MENDY | Professeur Assimilé | UASZ |

Devant le Jury composé de :

| Prénom et Nom | Fonction | Grade | Institution |
|--------------------------|-----------------|---------------------------------|--------------------|
| Pr. Melyan MENDY | Président | Professeur Assimilé | UASZ |
| Dr Mor NDONGO | Examineur | Maitre de conférences titulaire | UASZ |
| Dr Thierno Amadou DIALLO | Rapporteur | Maitre de conférences titulaire | UASZ |
| Dr Papa Alioune CISSE | Encadrant | Maitre de conférences titulaire | UASZ |

Année universitaire 2021-2022

DEDICACES

Je dédie ce travail à ma mère, AMINATA NDIAYE et à mon feu père, IBRAHIMA BAÏDY
CARLOS DIALLO.

REMERCIEMENTS

Je remercie DIEU le tout puissant de nous avoir donné la santé et la volonté d'entamer et de terminer ce mémoire.

Tout d'abord, ce travail ne serait pas aussi riche et n'aurait pas pu voir le jour sans l'aide et l'encadrement de Dr Papa Alioune CISSE que je remercie tout particulièrement pour ses qualités humaines, pour sa patience, sa rigueur, ses précieux conseils et sa disponibilité durant l'élaboration de ce mémoire. Mes remerciements les plus sincères s'adressent également aux membres du jury qui ont accepté d'apprécier mon travail. Aussi, à tous mes professeurs pour leurs générosités et la grande patience dont ils ont su faire preuve.

A ma chère et tendre mère, Aminata NDIAYE que je remercie pour ses sacrifices, son amour et sa joie de vivre. A M. IBRAHIMA BAÏDY CARLOS DIALLO, Je prie DIEU pour qu'il t'accueille dans son plus haut paradis. Je n'ai jamais oublié d'être une bonne personne et de viser l'excellence. Repose toi Bien Mamcar.

A mes frères : Souleymane, Elhadj Amadou Baba dobe, Mouhamed, vous m'inspirez de jours en jours. Une dédicace particulière à Mouhamed DIALLO, ma première source de motivation, n'oublie jamais l'amour que je te porte, deviens un grand homme intègre et juste, je te souhaite plein de succès mon futur astronaute.

A mes petites sœurs : Mariama, Khadija et Yande, vous êtes mes rayons de soleil, ne cessez jamais de briller. Trouvez votre voie et que rien ne vous arrête. Ce monde vous appartient.

A mes oncles, tantes, cousins et cousines, un grand MERCI.

Un merci bien particulier adressé à Adja Ndiereby SALL, la Jeune dame qui a su durant toutes ces longues années, me prendre sous son aile comme une sœur de sang, être patiente et tolérante face à mes défauts et partager mes joies et peines. Ces mots vont aussi à l'endroit de Fatou Bintou NDIAYE (FABINA), pour ton amitié sincère et pour ta tendresse. Tu es mon paradis. Que Dieu nous guide et nous accorde une longue vie.

Je remercie également mon Mentor, un cadeau du ciel, Ibrahima DIAVE. Mille pages ne suffiraient pas pour te montrer ma gratitude. Tu m'as soutenu sans rien attendre en retour.

Que Dieu te récompense aux centuples tes bonnes œuvres. Merci pour nos fous rires, nos karaokés, nos balades.

A toi Mamadou Moundaw DIOUF, ton soutien infaillible est écrit dans l'histoire de ma vie, Merci pour ta patience, ton affection et tes encouragements.

A Bineta SECK et Ousmane CISSE comme je vous aime, vos enseignements islamiques me suivent dans chaque pas, votre grandeur, votre amour, tout de vous m'inspire. Une vie longue à ma princesse Khady Lison.

Un grand merci à ma famille, plus particulièrement à mes oncles Mamadou NDIAYE, Abibou NDIAYE, Alioune NDIAYE.

Merci à M. Ousseynou SENE, Bocar SOW, Alain Philippe SAGNA, Ablaye Niang FAYE, Thierno Mamadou Aliou DIALLO, Niokhobaye DIOUF, Mamadou gallis SY, Alioune CISSE, Seyni Ousmane SEYDI, Marie SOW, Adja Mariama BA, Baïdy Biram TOURE, Mouhamed Mouhidine SOW, Geneviève Isabelle Agnen Tending, Mariama COULIBALY, Kadiatou DIALLO, Khady Mbaye DRAME, Oumoul Khayri GUEYE, ma fille Dialika NGA DIOUF, Mon fils Ousmane William KEBE, Oumou Salima TOURE, Aida DIOUF, Mame Penda, Boubacar SENGHOR, Papa Mamadou, Ahmadou Bamba NDAO, Maman Mbissine DIONE, Talla TOURE, Habib Rufin KANDE, Djibril Yoro SY, Libass Thiam, Oumou DIAO bana, Charles Koumaya SAMBOU, Ouleymatou CAMARA, Zeyna NDIAYE, Baba DIEYE, Amadou DIEYE, Abdoulaye DIONE, le syndicat dans sa totalité. Tonton SAMB (bibliothèque) pour tes conseils dès mon premier jour à l'université. A mes camarades de classe en Management Informatisé des Organisations et Management des Systèmes d'Information Automatisé, mes camarades Thiessois.

Merci à l'actuel Directeur du CROUS de Ziguinchor, M. Sana SANE pour sa confiance dans le travail, au Chef des Services Administratifs, M. Oumar GNING et à tous mes collègues.

A l'équipe de la communication du CROUSZ : M. Souleymane SALL, Serigne Modou Habib SENE, KITOKO (alias fady 2).

Que tous ceux qui ont contribué à mener à bien ce mémoire trouvent ici l'expression de ma parfaite considération.

RÉSUMÉ

L'Afrique s'est engagée depuis des décennies à résoudre les défis d'accès aux soins de santé par d'importants investissements, résultant en une infrastructure médicale robuste. Grâce à l'intégration des technologies de l'information dans ce secteur, les professionnels sont arrivés à offrir une prise en charge personnalisée à chaque patient et un plan de prévention efficace. Pour que le système sanitaire sénégalais arrive à ce stade, il est primordial d'avoir une gestion attentive des données sanitaires et extra-sanitaires de la population qui va garantir l'interopérabilité des systèmes de gestion hospitalière et l'accès à temps réel du dossier médical par le patient. C'est ainsi que nous avons proposé d'intégrer un module permettant aux patients d'accéder eux-mêmes à leurs dossiers médicaux dans l'application de gestion des données médicales partagées appelée Guichet Médical Unique (GMU).

Dans la création de l'application web, nous avons utilisé le Framework Django basé sur le langage Python et pour une bonne prise en main de l'application, nous avons choisi le modèle de cycle de vie 2TUP et le langage de modélisation UML afin de définir les différentes phases et modèles de conception du système.

Mot clés : Santé numérique ; dossier médical électronique ; généalogie médicale

ABSTRACT

Africa has been committed for decades to solving healthcare access challenges through significant investments, resulting in a robust medical infrastructure. Thanks to the integration of information technologies in this sector, professionals have managed to offer personalized care to each patient and an effective prevention plan. For the Senegalese health system to reach this stage, it is essential to have careful management of the health and extra-health data of the population which will guarantee the interoperability of hospital management systems and real-time access to the file. medical by the patient. This is how we proposed integrating a module allowing patients to access their medical records themselves in the shared medical data management application called Guichet Medical Unique (GMU).

In the creation of the web application, we used the Django Framework based on the Python language and for a good handling of the application, we chose the 2TUP life cycle model and the UML modeling language in order to define the different phases and design models of the system.

Keywords: Digital health; electronic medical record; Medical Genealogy

Table des matières

| | |
|--|------|
| DEDICACES | I |
| REMERCIEMENTS | II |
| RÉSUMÉ..... | IV |
| ABSTRACT..... | V |
| LISTE DES FIGURES | VIII |
| LISTE DES TABLEAUX | IX |
| LISTE DES ABREVIATIONS | X |
| INTRODUCTION GENERALE..... | 1 |
| 1. Contexte Général | 1 |
| 2. Problématique de notre étude..... | 2 |
| 3. Objectif et intérêt de notre étude..... | 4 |
| 4. Plan du mémoire | 5 |
| Chapitre I. État de l’art et généralités du projet..... | 6 |
| I.1 Définitions de quelques concepts..... | 6 |
| I.2 Revue de quelques applications de gestion de données médicales | 8 |
| I.2.1 Les projets du CSSDOS | 8 |
| I.2.2 E-YONE MEDICAL | 9 |
| I.2.3 SIMENS..... | 10 |
| I.2.4 IPROMED | 11 |
| I.2.5 DHIS2 | 11 |
| I.3 Généralités du projet et cadrage de nos travaux de mémoire | 12 |
| I.3.1 Architecture de l’application..... | 12 |
| I.3.2 Cadrage de nos travaux de mémoire..... | 15 |
| Chapitre II. Analyse et conception de notre module | 15 |
| II.1 Méthodologie..... | 15 |
| II.1.1 Méthode Classique | 16 |
| II.1.2 Méthode agile..... | 18 |
| II.2 Aperçu sur UML | 20 |

| | |
|--|----|
| II.2.1 Définitions et historique | 20 |
| II.2.1.1 Le diagramme de cas d'utilisation | 21 |
| II.2.1.2 Le diagramme de classes | 22 |
| II.2.1.3 Le diagramme d'activité | 22 |
| II.3. Analyse de notre module | 23 |
| II.3.3 Les diagrammes de cas d'utilisation | 26 |
| II.4 Conception de notre module | 27 |
| II.4.1 Les données de notre module | 27 |
| II.4.2 Le diagramme de classes de notre module | 27 |
| Chapitre III. Implémentation et présentation de notre module | 28 |
| III.1 Outils et techniques de développement | 28 |
| III.1.1 Le design pattern MVC | 29 |
| III.1.2 Le Framework Django | 30 |
| III.1.3 Le scaffolding | 32 |
| III.1.4 La base de données PostgreSQL | 32 |
| III.2 Vue d'ensemble sur l'implémentation de notre module | 34 |
| III.2.1 Implémentation du modèle | 35 |
| III.2.2 Implémentation des vues | 35 |
| III.2.3 Implémentations des Template | 38 |
| III.3 Présentation de notre module | 39 |
| III.3.1 Gestion des membres | 39 |
| III.3.2 Gestion de la généalogie | 40 |
| III.3.3 Fenêtres d'authentification | 42 |
| CONCLUSION ET PERSPECTIVE | 43 |
| Bibliographie et Webographie | 45 |

LISTE DES FIGURES

| | |
|---|----|
| Figure 1-Interface client passeport médical de EYONE | 11 |
| Figure 2-interface client eyassur de Eyone | 11 |
| Figure 3-Architecture de l'application "GMU" | 13 |
| Figure 4-Synthèse de la méthode MERISE | 18 |
| Figure 5-Branche 2TUP | 20 |
| Figure 6-processus de SCRUM | 21 |
| Figure 7. Première version du modèle de dossier médical | 25 |
| Figure 8-diagramme de cas d'utilisation du module famille | 27 |
| Figure 9. Modèle de dossier médical comportant les liens familiaux du patient | 29 |
| Figure 10- schéma descriptif Du MVC | 31 |
| Figure 11- image de Django | 32 |
| Figure 12-logo PostgreSQL | 34 |
| Figure 13-Les différentes classes "métier" de l'application "familyarea" | 37 |
| Figure 14-Urls correspondant aux fonctionnalités de l'application "familyarea" dans son fichier « urls.py » | 37 |
| Figure 15-Les vues de l'application "familyarea" dans son fichier "views.py"..... | 39 |
| Figure 16-Fenêtre de visualisation et d'ajout des membres d'une famille | 40 |
| Figure 17-Fenêtre de visualisation du dossier médical d'un membre..... | 41 |
| Figure 18-Fenêtre de visualisation des détails d'une consultation médicale | 41 |
| Figure 19-Fenêtre de gestion des couples dans une famille | 42 |
| Figure 20-Fenêtre de gestion d'un couple | 43 |
| Figure 21-Fenetre de connexion au GMU | 43 |
| Figure 22- Fenêtre de changement de mot de passe | 44 |
| Figure 23-fenetre de récupération de mot de passe | 44 |

LISTE DES TABLEAUX

| | |
|--|----|
| Tableau 1-Liste des classes représentant les données de notre module | 28 |
| Tableau 2-Tableau comparatif de PHP et PYTHON | 33 |
| Tableau 3- Tableau comparatif MYSQL et POSTGRESQL | 35 |

LISTE DES ABREVIATIONS

- HTML : HyperText Markup Language
- CSS : Cascading Style Sheets
- MVT : Modèle-vue-template
- ORM : object-relational mapping
- URL : Uniform Resource Locator
- CRUD : create, read, update, delete
- SQL : Structured Query Language
- SGBD : Système de Gestion de Bases de Données
- SGBDR : Système de Gestion de Bases de Données Relationnelles
- UML : Unified Modeling Language
- VSCode : Visual Studio Code
- IOT : The internet of things
- IA : Intelligence Artificielle
- BI : Business Intelligence
- SIS : système d'information sanitaire
- OMG : l'Object Management Group
- UML : Unified Modeling Language
- DRY : Don't Repeat Yourself
- MVC : Modèle-Vue-Contrôleur
- CSRF : Cross-Site Request Forgery
- DES : Dossier de Sante Electronique
- OMS : Organisation Mondiale de la Sante
- UE : Union européenne

- PSE : Programme Sénégal Émergent
- CSSDOS : Cellule de la Carte Sociale, de la Sante Digital et de l'Observation de la Santé
- SIG : Système d'Information Géographique de Santé
- ROSSA : Répertoire de l'Offre de Soins de Santé
- ERP : Entreprise Resource Planning
- LANI : Laboratoire d'Analyse Numérique et d'Informatique
- MERISE : Merise Méthode d'Etudes et de Réalisation Informatique pour les Systèmes d'Entreprise
- DHIS2 : Le District Health Information Software 2
- 2TUP : Two Track Unified Process
- CSRF : Cross-Site Request Forger

INTRODUCTION GENERALE

1. Contexte Général

Dans ce monde dominé par le numérique, le plus grand défi pour chaque pays est de bénéficier de ce progrès pour avoir une longueur d'avance et assurer une bonne qualité de vie à la population. De l'informatisation des procédures administratives à la robotique, en passant par la prise de décision par le biais de l'intelligence artificielle, les pays développés ont su très vite investir dans la technologie et l'exploitent à des fins qui sont d'intérêt vital, en particulier le domaine de la santé. Cette dernière occupe une place centrale dans la problématique du développement international. Les nations émergentes ont engagé des ressources colossales dans ce secteur en termes de stratégies, d'infrastructures, de temps, de ressources humaines et financières etc. [1] Ces efforts ont donc abouti à des découvertes répondant à des besoins urgents de la population, tels que l'accessibilité des soins de santé, la rapidité de la prise en charge des urgences, la compréhension et la prévention de nombreuses maladies etc.

En Afrique, particulièrement dans les pays en développement tels que le Sénégal, un grand pas reste à faire malgré les efforts constatés par l'Organisation Mondiale de la Santé (OMS) dans l'élaboration de plans stratégiques sur la santé numérique entreprise par ces nations. L'État du Sénégal, dans son plan stratégique santé digitale 2018-2023 établi par le ministère de la santé et de l'action sociale (MSAS), avait pour objectif en 2019, de développer la sante digitale d'ici 2023 et stipulés comme suit : « D'ici 2023, la Stratégie Santé Digitale permettra au Sénégal d'améliorer durablement la couverture sanitaire universelle des populations et d'assurer une prise de décision des acteurs basée sur des informations de qualité et sécurisée ». Ce large programme a aujourd'hui favorisé le développement et l'utilisation constante de plusieurs outils de gestion dans les hôpitaux. Malgré cela, lors de nos enquêtes dans les structures de santé locales (hôpitaux de niveaux 1, 2, 3 ; centres de santé, cabinets médicaux privés, etc.), nous avons relevé ces faits :

- Des hôpitaux sont encore dans le système traditionnel, autrement dit la rédaction manuelle (papier et stylo). Une situation qui pourrait s'expliquer par la qualité relativement médiocre de l'infrastructure informatique (connectivité réseau...) et le

manque de ressources, mais aussi des difficultés relatives à l'implication et l'acceptabilité globale de la transition technologique par les usagers.

- D'autres mieux dotés ont informatisé leur processus de gestion, avec une majorité qui s'est limitée au stade de la facturation et de la comptabilité. Un petit nombre dispose d'un système en interne qui implique la gestion des dossiers médicaux des patients.

Dans le deuxième cas, la plupart des systèmes informatisés que nous avons trouvés sont de type conventionnel. C'est-à-dire qu'ils implémentent simplement les opérations basiques CRUD : Create (créer un dossier médical pour un patient), Read (consultation, recherche, récupération d'informations, dossiers, etc.), Update (mettre à jour un dossier médical, ajouter ou modifier des informations dans un dossier) et Delete (supprimer un dossier ou une information d'un dossier). Ces systèmes, bien qu'importants dans la production de données médicales, se révèlent désormais inadaptés pour faire face aux défis de gestion et d'utilisation des données de santé.

En effet, à l'heure actuelle, l'évolution rapide des technologies a ouvert de nouvelles perspectives. Les données médicales, sanitaires et extra-sanitaires sont utilisées pour anticiper et accélérer le diagnostic des maladies, prédire des épidémies, mieux comprendre certaines pathologies, améliorer et personnaliser les traitements chez les patients, assister à distance, et bien d'autres. Cela découle de l'intégration de nouvelles disciplines telles que l'intelligence artificielle (IA), la Business Intelligence (BI), l'Internet des Objets ou Internet Of Things (IOT), etc.

2. Problématiques de notre étude

Lors des échanges constructifs que nous avons eus avec des médecins et des directeurs d'établissements de santé chevronnés, il est ressorti de manière frappante que les ingénieurs, bien que doués dans la conception technologique, ont parfois omis de considérer avec objectivité les besoins concrets des professionnels de la santé et des usagers. Nos interlocuteurs ont clairement signifié leur désintérêt pour l'adoption de système informatique qui ne serait pas aligné sur leurs attentes. Ils se sont manifestés à travers une série de questions percutantes, reflétant leur préoccupation quant à l'adéquation des solutions technologiques aux réalités du terrain.

- Un médecin nous a proposé un sujet de réflexion scientifique : « *Si pour une pathologie P donnée, on peut proposer 3 traitements (T1, T2, T3) par exemple. Comment savoir quel*

traitement parmi T1, T2 et T3 est mieux adapté pour traiter la pathologie P chez un patient donné, en fonction de critères d'ordre démographique (âge, sexe, ...) culturel (habitudes alimentaires, activités sociales), professionnel (activités professionnelles et lucratives), géographique (environnement d'habitations), etc. ».

Le médecin nous a révélé que c'est dans cette perspective qu'il demande toujours à ses patients de revenir même s'ils sont guéris de leurs maladies afin de déterminer la durée et l'efficacité des traitements prescrits. Pour cela, il doit disposer d'une quantité importante de données sur papiers dont il ne sait pas quoi en faire. En effet, un traitement manuel de ces données est quasi impossible et les outils informatiques qu'on lui présente jusqu'ici ne permettent pas de répondre à son type de questionnements.

- Un chirurgien pédiatre nous dit : *« Ces temps-ci, j'interviens beaucoup pour opérer des hernies ombilicales et c'est une situation anormale dont on devrait avoir une explication sur les causes et les origines ».*
- Un autre médecin dit ceci : *« Un patient qui était suivi pour une pathologie donnée dans un cabinet d'une ville donnée, a été affecté récemment dans ma ville. Je dois le suivre et je n'ai pas accès à son historique médical et lui non plus n'y a pas accès. Comment faire pour que les patients aient accès eux-mêmes à leurs propres données médicales et qu'ils puissent aussi eux-mêmes accorder un accès total, ou du moins partiel, de ces données aux médecins qui les traitent ? ».* Une façon de dire que les données médicales des patients ne doivent plus être détenues exclusivement par les structures de santé en écartant totalement les patients.
- Un autre médecin nous a transmis une requête d'un patient qui dit être préoccupé par la situation suivante : *« Je suis un chef de famille et j'ai en charge la santé de tous les membres de ma famille : je prends des rendez-vous pour eux et je paye leurs frais médicaux. Parfois je suis en déplacement et quand c'est le cas, c'est difficile. Je veux pouvoir me connecter à distance, réserver un rendez-vous pour un parent et payer les frais sans me déplacer. Je dois aussi pouvoir avoir accès à leurs données médicales pour un bon suivi à distance. ».*
- Un directeur d'une structure de santé nous rapporte aussi les préoccupations d'un de ses homologues banquiers : *« Certaines données médicales de nos clients sont déterminantes pour nous aider quant à la décision d'accorder un prêt. Malheureusement, ces données sont*

inaccessibles et en plus certains clients choisissent parfois délibérément de fournir de fausses informations sur leurs états de santé. ».

A partir de là, il serait important, avant de plonger dans le vif du sujet, de se poser une question importante à savoir : « Comment Intégrer le patient dans le système de suivi de son dossier médical ? ».

3. Objectif et intérêt de notre étude

Ce travail de recherche aborde une problématique d'une importance capitale et d'une grande sensibilité, qui concerne l'accessibilité, la gestion et le traitement des données médicales. Il s'agit d'un sujet délicat qu'il faut traiter avec beaucoup d'attentions et de précautions, car s'agissant de données à caractère personnel. Cependant, nous nous concentrons exclusivement sur une approche scientifique pour développer un outil informatique capable de répondre aux questionnements des uns et des autres, évitant ainsi de s'embourber dans des considérations juridiques et réglementaires.

L'objectif central de notre étude est de concevoir une application de gestion des données médicales partagées qui réponde aux préoccupations établies ci-dessus. Il s'agit entre autres de :

- Permettre aux patients d'accéder eux-mêmes à leurs propres données médicales.
- Permettre aux patients d'accorder l'accès à des parties de leurs données médicales à des structures de santé et à d'autres structures telles que les assurances, les mutuelles de santé, les banques, etc.
- Permettre aux patients de gérer aussi les données médicales des membres de leur famille.
- Fournir des services d'ordre médical et administrative tels que :
 - La réservation à distance de rendez-vous médicaux.
 - Le paiement en ligne des frais médicaux. ○ Le suivi médical à distance.
 - La gestion des files d'attente dans les structures de santé.

- L'aide à la décision avec l'utilisation d'outils BI (Business Intelligence) et IA (Intelligence Artificielle) :
 - Tableaux de bord.
 - Apprentissage automatique pour l'aide au diagnostic et aux traitements.
 - Etc.

4. Plan du mémoire

Ce document est réparti en trois principaux chapitres :

- Le premier chapitre, intitulé « état de l'art et généralités du projet », présente les généralités du projet dans le cadre duquel se situent nos travaux de mémoire. Nous proposons dans un premier temps quelques définitions de notions essentielles qui permettront de faciliter la lecture du document. Ensuite, nous présentons quelques outils de gestion de données médicales. Enfin, nous présentons l'architecture générale de la plateforme qui est en cours de développement dans le cadre de ce projet. Ce qui nous permet par la suite de faire un cadrage fixant les limites de nos travaux de mémoire dans le projet global.
- Le deuxième chapitre intitulé « Analyse et conception » donne un aperçu sur UML qui est l'outil d'analyse et de conception utilisé dans ce travail. Ensuite, nous présentons de façon détaillée le problème spécifique abordé dans ce mémoire. Ensuite, nous faisons une analyse de la solution que nous avons proposée. Cette analyse consiste à faire l'inventaire des fonctionnalités de notre solution. Cela inclut les diagrammes de cas d'utilisation montrant les relations entre les acteurs du système et leurs besoins fonctionnels. Enfin, nous y aborderons la conception de notre solution qui est illustrée par un dictionnaire des données et un diagramme de classes pour l'implémentation de notre module.
- Le troisième et dernier chapitre décrit, d'abord, les technologies utilisées pour implémenter notre solution. Ensuite, nous présentons les détails d'implémentation de la solution. Enfin, nous présentons les différentes fonctionnalités et interfaces de notre module.

Enfin, nous terminerons par une conclusion générale et dégagerons quelques perspectives.

Chapitre I. État de l'art et généralités du projet

I.1 Définitions de quelques concepts

Système d'information : c'est une infrastructure technologique et organisationnelle qui permet de collecter, stocker, traiter et diffuser des données et des informations au sein d'une organisation ou d'un réseau. Il englobe les logiciels, le matériel informatique, les procédures et les personnes qui travaillent ensemble pour gérer les données et soutenir les processus décisionnels au sein de l'entreprise. Il permet d'améliorer l'efficacité du fonctionnement interne de l'entreprise en restituant la bonne information sous le format approprié [2].

Système d'information sanitaire : Le système d'information sanitaire (SIS) est un ensemble intégré de composants, de processus, de technologies et de ressources qui collecte, stocke, gère, analyse et diffuse des informations liées à la santé, aux soins de santé, à la gestion des services de santé et aux décisions médicales [3]. Il vise à améliorer la gestion des systèmes de santé, la prestation des soins, la surveillance de la santé publique et la prise de décisions basées sur des données dans le domaine de la santé.

Santé numérique ou E-santé : La santé numérique également appelée e-santé, selon l'OMS, se réfère à l'utilisation des technologies de l'information et de la communication (TIC) dans le domaine de la santé. Elle englobe des domaines tels que la télémédecine, les dossiers médicaux électroniques, les applications mobiles de santé, les dispositifs médicaux connectés, les S.I de vigilance et d'orientation, et d'autres solutions technologiques visant à améliorer la prestation des soins de santé, la gestion des données médicales et la communication entre les professionnels de la santé et les patients [4].

M-santé : M pour mobile, s'agit d'une partie de la sante numérique. C'est l'ensemble des ressources sanitaires exploitables avec un smartphone, les dispositifs de surveillance de patients (objets connectes) etc. c'est un dispositif qui facilite à tous les patients l'accès à ses informations sanitaires [5], [6].

Généalogie médicale : constitue l'étude de la transmission des maladies héréditaire et génétique par la généalogie. Principalement orientée vers les familles porteuses de maladies transmissibles, la généalogie médicale d'un individu doit permettre d'identifier des ancêtres

porteurs sur plusieurs générations. Le généalogiste prend ainsi en compte l'ensemble des antécédents médicaux et familiaux [7].

Dossier médical : Un dossier médical est un ensemble de documents, de données et d'informations concernant la santé d'un patient. Il comprend des informations sur les antécédents médicaux, les diagnostics, les traitements, les examens médicaux, les allergies, les médicaments pris, et d'autres données médicales pertinentes. Les dossiers médicaux peuvent être sous forme papier ou électronique et sont essentiels pour assurer une prise en charge médicale appropriée et coordonnée [8] [9].

Dossier médical électronique : c'est une version numérique des informations de santé d'un individu. Il rassemble, stocke et gère de manière électronique les données médicales d'un patient provenant de diverses sources, telles que les hôpitaux, les cliniques, les laboratoires, les pharmacies, etc. Le DME vise à centraliser l'information médicale d'un patient, facilitant ainsi l'accès et le partage sécurisés des données entre les professionnels de la santé autorisés [9] [10].

Le dossier médical partagé : est un outil électronique qui vise à centraliser et à partager des informations médicales importantes sur un patient [10]. Il a pour objectif d'améliorer la coordination des soins, d'assurer la continuité des traitements, et de faciliter l'accès aux informations médicales pertinentes pour les professionnels de la santé autorisés [11].

Guichet unique : Un guichet unique est un point d'accès unique à des services, des informations ou des ressources [12]. Il simplifie l'accès à diverses prestations en regroupant des services ou des informations provenant de différentes sources en un seul endroit. Dans le contexte des services publics ou administratifs, un guichet unique permet aux citoyens ou aux entreprises d'interagir avec le gouvernement ou d'autres entités de manière plus efficace en réduisant la bureaucratie et en offrant des services centralisés.

Business Intelligence : La Business Intelligence (BI), également appelée "informatique décisionnelle", est un ensemble de technologies, de processus et d'outils qui permettent aux entreprises de collecter, de stocker, d'analyser et de présenter des données liées à leurs activités pour prendre des décisions éclairées. La BI vise à transformer les données brutes en informations pertinentes, en connaissances exploitables et en rapports significatifs pour aider

les dirigeants et les professionnels de l'entreprise à comprendre, évaluer et améliorer leurs performances.

Intelligence artificielle : L'intelligence artificielle (IA) est un domaine de l'informatique qui se concentre sur la création de systèmes informatiques capables de réaliser des tâches qui nécessitent normalement une intelligence humaine. Cela inclut la capacité d'apprendre à partir des données, de résoudre des problèmes, de prendre des décisions, de comprendre le langage naturel, de percevoir l'environnement et d'effectuer des tâches diverses [13]. Les techniques d'IA comprennent l'apprentissage automatique, le traitement du langage naturel, la vision par ordinateur et bien d'autres. L'IA est utilisée dans de nombreux domaines, y compris la santé, la finance, l'automobile, la robotique et bien d'autres.

I.2 Revue de quelques applications de gestion de données médicales

Il existe actuellement une diversité de solutions numériques dans le secteur de la santé à l'échelle mondiale. Parmi celles qui ont cherché à s'implanter sur le marché sénégalais, la plupart ne prend pas toujours pleinement en compte les spécificités de la société sénégalaise, mais aussi subsiste l'inquiétude liée à la gestion des données hors du territoire national. Cette constatation a suscité l'intérêt de nombreuses entreprises qui se sont lancées dans le développement de logiciels de gestion inclusifs, adaptés aux besoins et aux réalités locales.

C'est dans cette mouvance que nous avons choisi d'aborder les solutions locales en phase de conception ou d'implémentation.

I.2.1 Les projets du CSSDOS

L'État du Sénégal, dans le cadre du Programme Sénégal Émergent (PSE), a manifesté son engagement envers l'amélioration des services de santé pour l'ensemble de la population.

Cette démarche est concrétisée par l'orientation 4 portant sur le développement de la E-santé et de la télémédecine dans le plan stratégique du ministère de la santé. Pour la période allant de 2019 à 2023 [14], il a été planifié une série d'initiatives destinées à transformer le paysage de la santé. Parmi celles-ci, on note le projet de Dossier Patient Partagé, visant à centraliser les informations médicales pour une meilleure coordination des soins. La télésanté également étendue, afin d'ouvrir la voie à des services à distance. De plus, un système d'information hospitalier sera mis en place pour optimiser la gestion des établissements de santé. L'ensemble

de ces projets, d'envergure nationale, est confié à la responsabilité du CSSDOS, chargé de coordonner et d'administrer de manière globale la mise en œuvre de ce programme ambitieux [15].

A nos jours, nous constatons que la dimension géographique a été pleinement mise en œuvre grâce à la création d'un Système d'Information Géographique de Santé (SIG), permettant une analyse spatiale des besoins en santé. On note la ROSSA (Répertoire de l'Offre de Soins de Santé) [16] qui permet de localiser une structure ou des services particuliers. La situation en rapport avec le DPP est méconnue pour le moment. Ce qui permet d'en conclure que le Sénégal ne dispose pas pour le moment d'un système qui répond aux besoins énumérés en dessus.

I.2.2 E-YONE MEDICAL

Eyone est une solution de gestion hospitalière de bout en bout. Très connu pour être une solution innovante dans l'écosystème, la solution donne accès à trois (3) interfaces : le ERP médical-SIH (cliniques, cabinets médicaux, hôpitaux, État, pharmacies et laboratoires), EYASSUR (assurances, IPM, mutuelles et aux autres financeurs de soins) et le PASSPORTMEDICAL (patients, assurés et l'État). Elle permet de digitaliser les dossiers patients ainsi que la logistique, l'administration et les finances des établissements sanitaires. De même, elle donne l'occasion aux patients d'interagir avec les professionnels de la santé en partageant leur dossier médical, d'effectuer un paiement en ligne et de prendre des rendez-vous à distance.

En somme, Eyone médical est le logiciel qui se rapproche le plus de notre dynamique puisqu'il implique les structures sanitaires et non sanitaires dans sa démarche. Mais aussi d'un identifiant unique qui permettrait aux autorités étatiques de disposer d'une vue plus large sur la santé de la population. En revanche, il ne fournit pas un espace dédié à la gestion de la généalogie médicale qui permet de prédire les maladies consanguines, et génétique [17].

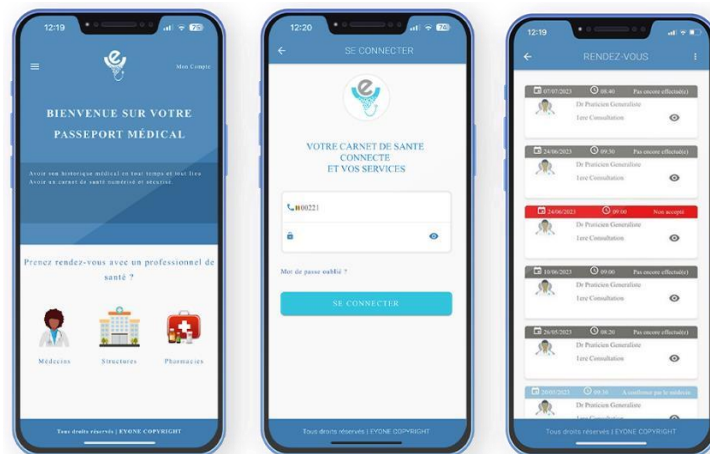


Figure 1-Interface client passeport médical de EYONE © eyonemédica.com



Figure 2-interface client eyassur de Eyone image © eyonemédica.com

I.2.3 SIMENS

Le système d'information médical national pour le Sénégal est un ensemble de service destiné à la gestion des activités et données médicales et administratif des structures sanitaires du Sénégal. Il a pour ambition de créer un système d'identification médicale unique au niveau national à travers l'automatisation des dossiers médicaux. C'est un large projet propose en 2011 et porté par le Laboratoire d'Analyse Numérique et d'Informatique (LANI) de l'université Gaston Berger et l'équipe interdisciplinaire de Recherche en Informatique Médicale et en Technologie de l'Information et de la communication de l'Enseignement (UMTICE) de L'université Alioune Diop de Bambey [18]. le SIMENS a conçu un système de gestion de dossier médical électronique (DME) pour la drépanocytose pour faire le suivi et les soins de santé des nouveau-nés dépistés positifs et un système d'information de laboratoire pour la gestion des tests de dépistage de la drépanocytose.

I.2.4 IPROMED

Crée par le CIFRECOM (international de formation continue, conseil en acquisition de solutions Pro logiciels et communication), C'est un logiciel de gestion des cliniques et établissements de santé publique qui a pour objectif d'accélérer le traitement automatisé des données. Il contient 8 modules permettant de gérer le suivi du dossier médical, la fiche patiente, les prises en charge des assurés de la caisse de sécurité sociale et autres assureurs privées, les prises de rendez-vous, la comptabilité et le stock des médicaments, et les laboratoires [19].

Ce logiciel bien que pertinent, est plus orienté vers la gestion administrative des hôpitaux. Le patient n'a aucun accès à ses informations sanitaires à temps réel et ne permet pas le partage d'information entre d'autres structures.

I.2.5 DHIS2

Le District Health Information Software 2 (DHIS2) est une plate-forme de données de gestion de la santé gratuite et open source, développée par le programme Système d'Information sur la Santé (HISP) et utilisée par plusieurs organisations, y compris l'union européenne (UE). Au total, 54 pays déploient le DHIS2 à l'échelle nationale, dont 30 sont en phase pilote ou en phase précoce de leur déploiement [20]. Il permet de faire le suivi et la surveillance de la santé des patients, d'améliorer la surveillance des maladies et la détection des flambées, et accélérer l'accès aux données de santé à travers la collecte, la validation, l'analyse, la gestion et la présentation de données statistiques. Il offre des fonctionnalités de visualisation et la possibilité de créer des analyses à partir de données en direct en quelques secondes. Par contre, le DHIS2 n'implémente pas la gestion des prises de rendez-vous, le paiement des frais et tout comme les autres applications présentées au-dessus, il ne permet pas la mise en relation des informations familiales pour la compréhension des maladies liées aux génétiques [21].

I.3 Généralités du projet et cadrage de nos travaux de mémoire

I.3.1 Architecture de l'application

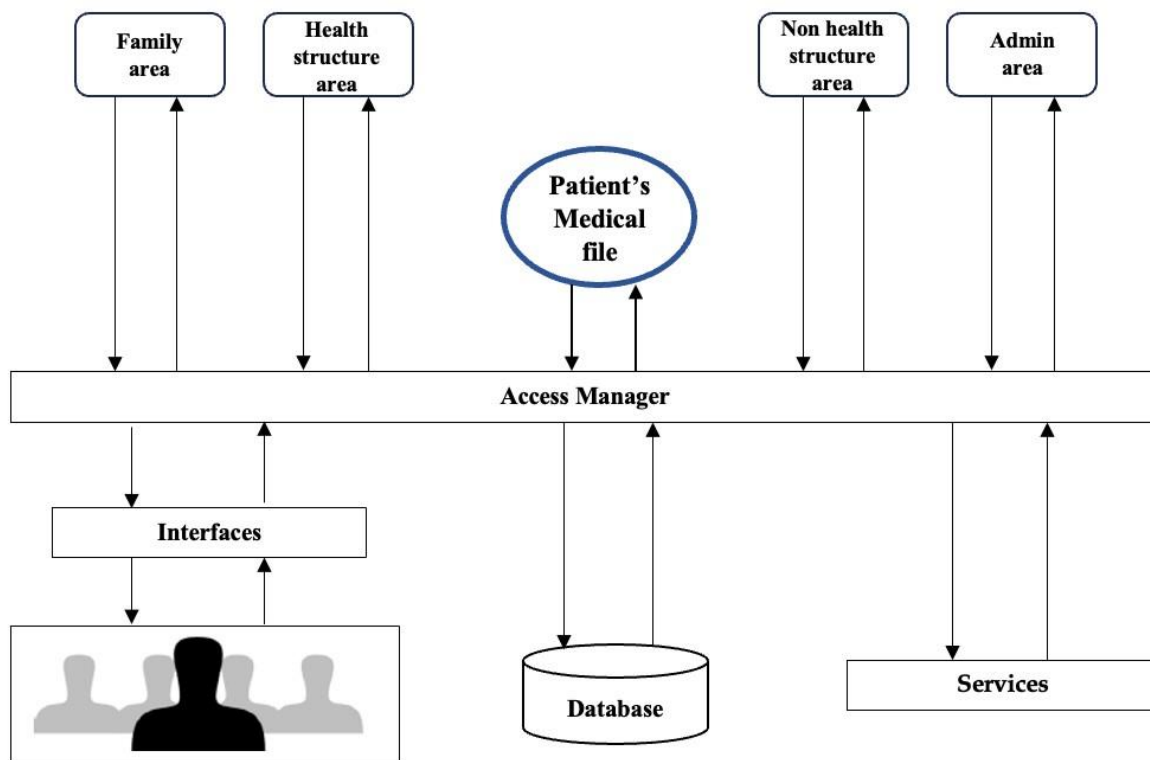


Figure 3-Architecture de l'application "GMU" © GMU

Nous avons appelé cette application « GMU (Guichet Médical Unique) ». Le choix de ce nom est motivé par l'orientation que nous donnons à notre plateforme. En effet, l'entité au cœur de cette application est le patient et son dossier médical. Il est l'*origine* et le *destinataire* de toutes les données du système. Chaque patient (et son dossier médical) existe de façon unique dans ce système malgré la multiplicité des autres entités pouvant y accéder.

Dans cette architecture, le module « Patient's Medical file » (ou dossier médical du patient) représente à la fois le dossier médical d'un patient et les dossiers médicaux de tous les patients. Il représente également toutes les données médicales, sanitaires et extra-sanitaires des patients, quelles que soient les sources, la nature et la complexité de ces données.

Toutes les entités du système interagissent avec le patient et son dossier médical par l'intermédiaire du module « Access manager » ou « gestionnaire des accès ». Ce module gère

et contrôle les accès aux dossiers médicaux des patients. Il permet éventuellement de définir et d'appliquer des règles restrictives telles que :

- Une donnée médicale ne peut être ajouté/modifié que par un médecin depuis l'espace d'une structure de santé.
- Un patient a accès total et en lecture seule à tout son dossier médical.
- Un patient peut accorder un accès en lecture/écriture à un médecin/structure de santé.
- Une donnée médicale cachée n'est visible que par le patient.
- Une structure de santé a accès en lecture/écriture à toutes les données médicales non cachées qu'elle a elle-même créées.

Dans cette architecture, nous avons un espace dédié pour chaque type d'entité impliquée dans la gestion, l'accès et/ou le traitement des dossiers médicaux :

- « **Family area** » ou « **espace familiale** ». C'est l'espace à partir duquel les patients accèdent à leurs dossiers médicaux. Un « Family area » est créé pour chaque famille souhaitant détenir ses propres données médicales. Il peut être administré par un gestionnaire qui gère (ajouter, modifier, supprimer) les membres. Chaque membre ajouté dans un espace familiale est un patient dans notre système. Le responsable d'un espace familial a accès à tous les dossiers médicaux des membres de la famille. De la même manière qu'il gère les membres de cet espace, il gère (ajout, modification, suppression) aussi les données ménagères et familiales (liens de parenté, mariages, fratrie, ...) qui sont nécessaires pour déterminer un arbre généalogique médical et l'historique d'une maladie familiale, prédire l'apparition de certaines maladies "héréditaires". C'est aussi à partir de cet espace qu'un patient accorde un accès à son dossier médical à un médecin.

- « **Health structure area** » ou « **espace des structures de santé** ». Cet espace est réservé aux structures de santé et à leurs personnels (utilisateurs) pour leur permettre la gestion des dossiers médicaux de leurs patients. Un espace est créé pour chaque structure de santé souhaitant utiliser notre application. Il est administré par un responsable qui gère (ajout, modification, suppression, suspension de comptes utilisateurs, ...) les membres (médecins, aides soignant, secrétaires, etc.) et le paramétrage (définition et gestion des profils, accès et restrictions, services médicaux proposés, grilles tarifaires des services médicaux, ...). Une fois

qu'un espace est créé pour une structure, sa gestion est totalement assurée par le(s) responsable(s) désigné(s).

- « **Non Health structure area** ». C'est l'espace à partir duquel les autres structures (assurances, banques, mutuelles de santé, ...) peuvent accéder à certaines informations les concernant (dont les accès et les restrictions sont définis dans le module « Access manager ») sur les dossiers médicaux des patients. C'est aussi à partir de leurs espaces respectives que ces structures fournissent à notre système des informations/données non médicales sur les patients, mais qui peuvent être particulièrement utiles dans certaines études et pratiques sanitaires.

- « **Admin area** » ou « **espace d'administration** » de l'application « GMU ». C'est dans cet espace que le système est globalement administré : créer un espace pour une famille et désigner le responsable, créer un espace pour une structure et désigner le responsable, définir et gérer les services que nous proposons.

L'application « GMU » propose également un ensemble de « **Services** » aux utilisateurs (patients, structures, médecins, directeurs des structures de santé, ...) de la plateforme. Parmi ces services, il y'a entre autres :

- Les prises de rendez-vous médicaux à distance. Ce module, accessible depuis les espaces familiales, permet à un utilisateur de prendre rendez-vous avec un médecin dans une structure de santé. Il est aussi accessible aux médecins depuis les espaces de leurs structures pour leur permettre de prédéfinir des jours et des horaires de disponibilités qui vont constituer des rendez-vous pour les patients.

- Les paiements en ligne des frais médicaux. Ce module permet aux patients et à certaines structures telles que les mutuelles de santé de payer des frais médicaux avec les moyens de paiement classiques tels que la carte bancaire et le mobile money.

- La gestion des files d'attente. Ce module permet aux structures de santé de bien gérer les files d'attente des patients qui arrivent dans une structure sans avoir pris de rendez-vous.

- Arbre généalogique médical. Ce service permet aux familles de pouvoir disposer de leurs arbres généalogiques médical et de pouvoir, par conséquent, bénéficier des services y afférents : historique ou trajectoire d'une maladie, maladies héréditaires, incompatibilité sanguine dans les couples, etc.

- L'aide à la décision. Ce service permet entre autres :

- Aux directeurs des structures de santé de disposer de tableaux de bord sur les activités et pratiques médicales et administratives de leurs structures.
- Aux médecins de disposer de tableaux de bord sur leurs pratiques médicales.
- Aux patients de disposer de tableaux de bord sur leurs données médicales.
- D'utiliser des outils d'apprentissage automatique pour améliorer et personnaliser les traitements proposés aux patients, améliorer le diagnostic des pathologies, faire des alertes etc.

Tous ces espaces et services sont accessibles aux utilisateurs via des « interfaces utilisateurs » spécifiques et toutes les données de l'application sont hébergées dans la base de données (« Databases »).

I.3.2 Cadrage de nos travaux de mémoire

L'architecture présentée ci-dessus permet de voir que le projet de « Guichet Médical Unique » est composé de plusieurs modules indépendants. En effet, chaque espace et service présenté fera l'objet d'un module qui viendra se greffer sur ce qui est appelé le « noyau » du système. Le « noyau » est le dossier médical (« Patient's Medical File ») accompagné du module « Access Manager » qui gère son accès.

Nos travaux de mémoire portent sur l'espace familiale (Family area) de l'architecture du guichet (Figure 1). Ils consistent plus précisément à développer un module permettant aux patients d'accéder eux-mêmes à leurs dossiers médicaux et d'y effectuer les opérations dûment citées.

Chapitre II. Analyse et conception de notre module

II.1 Méthodologie

L'analyse consiste, d'une part, à comprendre et à modéliser le fonctionnement d'un domaine de gestion au sein d'une organisation, et d'autre part, à concevoir les solutions informatiques adéquates. Elle permet de définir les objectifs du système d'information et d'identifier les contraintes potentielles. La phase de conception requiert l'utilisation de méthodes permettant d'obtenir un modèle de référence. Cette étape permet de décrire le système en utilisant un langage de modélisation afin de faciliter sa réalisation. En raison de la diversité de ces techniques, il est essentiel de faire un choix judicieux adapté à notre projet [22].

Ainsi Il existe deux grandes familles de méthodes de gestion de projet : les méthodes classiques et les méthodes agiles. Les méthodes traditionnelles, telles que le modèle en cascade, impliquent une approche linéaire, tandis que les méthodes agiles, comme 2TUP (Two Track Unified Process) et SCRUM, favorisent des cycles de développement itératifs et flexibles. Le choix entre ces approches dépend souvent de la nature du projet, des exigences spécifiques et des préférences de l'équipe de développement. Il convient également de noter l'importance croissante des nouvelles technologies telles que l'intelligence artificielle, l'apprentissage automatique et l'analyse de données dans le processus d'analyse et de conception.

II.1.1 Méthode Classique

Dans les méthodes classiques ou traditionnelles il existe plusieurs modèles comme le modèle en cascade, en V, le modèle Spiral... Ils suivent une approche séquentielle du développement. Ce processus linéaire est divisé en phases distinctes, telles que l'analyse, la conception, la mise en œuvre, les tests et la maintenance. Chaque étape dépend de la réussite de la précédente, et les modifications sont souvent difficiles à intégrer une fois le processus enclenché. Ces méthodes conviennent bien aux projets dont les exigences sont clairement définies dès le départ et qui ne devraient pas subir de changements importants au cours du développement.

L'une des méthodes les plus utilisées parmi les méthodes classiques est la méthode MERISE (Merise Méthode d'Etudes et de Réalisation Informatique pour les Systèmes d'Entreprise). C'est une méthodologie de modélisation à usage général dans le domaine du développement de système d'information. Elle procède à un traitement séparé des données et des processus, où la vue des données est modélisée en trois étapes : la conception, la logique et la physique. Cette méthode a été utilisé dans plusieurs grand projets du monde. Il a donc des points d'appréciation et présente ces limites [23] [24].

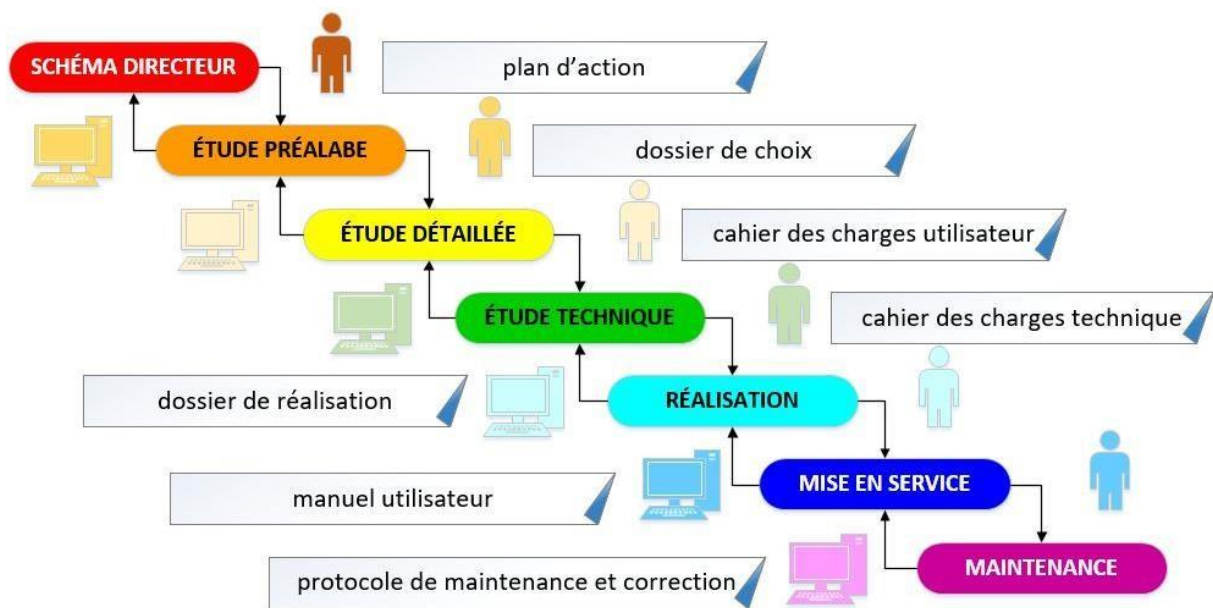


Figure 4-Synthèse de la méthode MERISE © fr.wikipedia.org Les

points forts :

- La méthode s'appuie sur une approche systémique : C'est donc une approche globale.
- Les concepts sont peu nombreux et Simples.
- Elle est assez indépendante vis à vis de la technologie.

Les points faibles :

- Elle ne s'occupe pas de l'interface utilisateur.
- Elle ne permet pas réellement une validation rapide de la part des utilisateurs.
- Il est très difficile de valider les traitements par rapport aux données et cela au niveau conceptuel ou organisationnel.
- Les méthodes traditionnelles présentent généralement une version complète du produit à la fin du cycle de développement.
- Peuvent être rigides en présence de modifications importantes après le début du développement.

Tenant compte l'aspect modulaire de ce projet, Merise présente ces limites sur l'adaptabilité et le mouvement d'ensemble exigé, avec cette multitude de donnée. Nous souhaitons faire du patient le centre de ce projet. Ce qui lui permet d'intervenir dans les champs essentiels.

II.1.2 Méthode agile

Les méthodes agiles sont conçues pour répondre aux défis des projets où l'incertitude, l'évolution des besoins et la nécessité d'une flexibilité accrue sont prédominants. SCRUM et 2TUP sont parmi les méthodologies agiles les plus répandues. Contrairement aux approches traditionnelles, les méthodes agiles privilégient des cycles de développement itératifs et incrémentaux qui consiste à découper le projet un certain nombre de cycle, ou itérations. Ces itérations sont en fait des mini-projets définis avec le client en détaillant les différentes fonctionnalités qui seront développées en fonction de leur priorité. Le chef de projet établit un calendrier correspondant aux tâches nécessaires pour le développement de ces fonctionnalités avec son équipe [25].

Le 2TUP pilote les risques et est centré sur l'architecture ou le système est décomposé en modules pour des besoins de maintenabilité et d'évolutivité. Il est aussi conduit par les cas d'utilisation en mettant les exigences des futurs utilisateurs du système. Il constitue un procédé intégré qui vise à répondre aux contraintes de changement, que ce soit sur le plan fonctionnel ou technique, auxquelles les systèmes d'information font face. Cette approche préconise un cycle de développement qui sépare distinctement les aspects techniques des aspects fonctionnels. L'idée centrale est que toute évolution imposée au système d'information peut être fragmentée et traitée de manière simultanée, en suivant une approche fonctionnelle d'une part et technique d'autre part. Ces deux orientations, fonctionnelle et technique, sont différenciées en deux branches distinctes, dont les résultats sont ultimement fusionnés pour donner forme au système. Le schéma de développement adopté prend la forme d'un "Y", comme représenté dans la figure ci-dessous. Il est important de noter que le processus 2TUP débute par une phase préliminaire d'étude, au cours de laquelle les intervenants du système sont identifiés, les échanges de messages entre les acteurs et le système sont scrutés, un cahier des charges est élaboré, puis le contexte est modélisé.

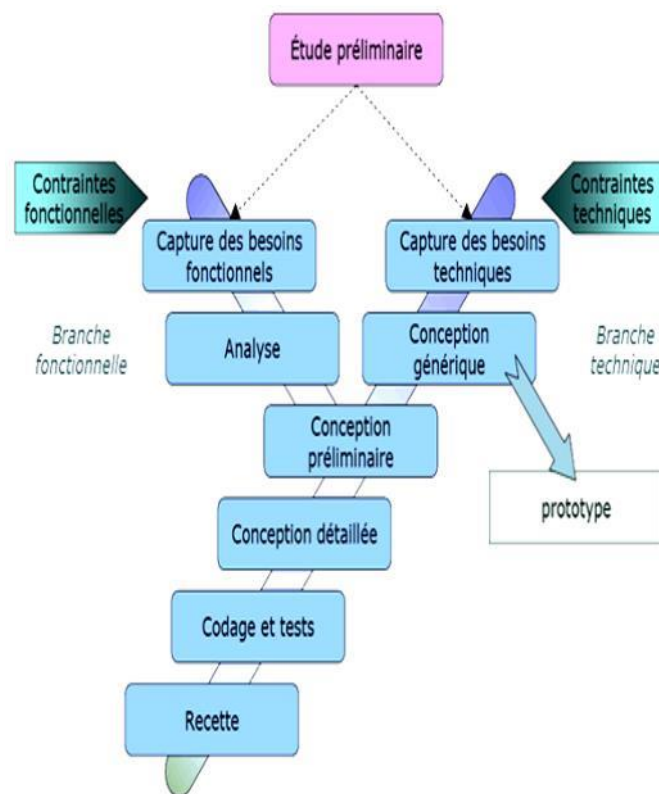


Figure 5-Branche 2TUP

En ce qui concerne SCRUM, c'est une méthodologie qui encourage une approche expérimentale, basée sur l'apprentissage tiré des expériences, en s'ajustant de manière continue aux réalités changeantes de l'environnement de travail. Scrum préconise un cycle de développement itératif, où chaque étape (appelée "sprint") débute par la définition d'un élément du produit final et aboutit à sa concrétisation. L'approche convient aux projets de taille importante faisant appel à des compétences variées et nécessitant un laps de temps considérable par rapport à d'autres méthodes agiles et orientées objet. Elle accorde une importance particulière à l'implication du client, présent au commencement et à la fin de chaque itération, augmentant ainsi les chances d'obtenir un résultat conforme à ses attentes.

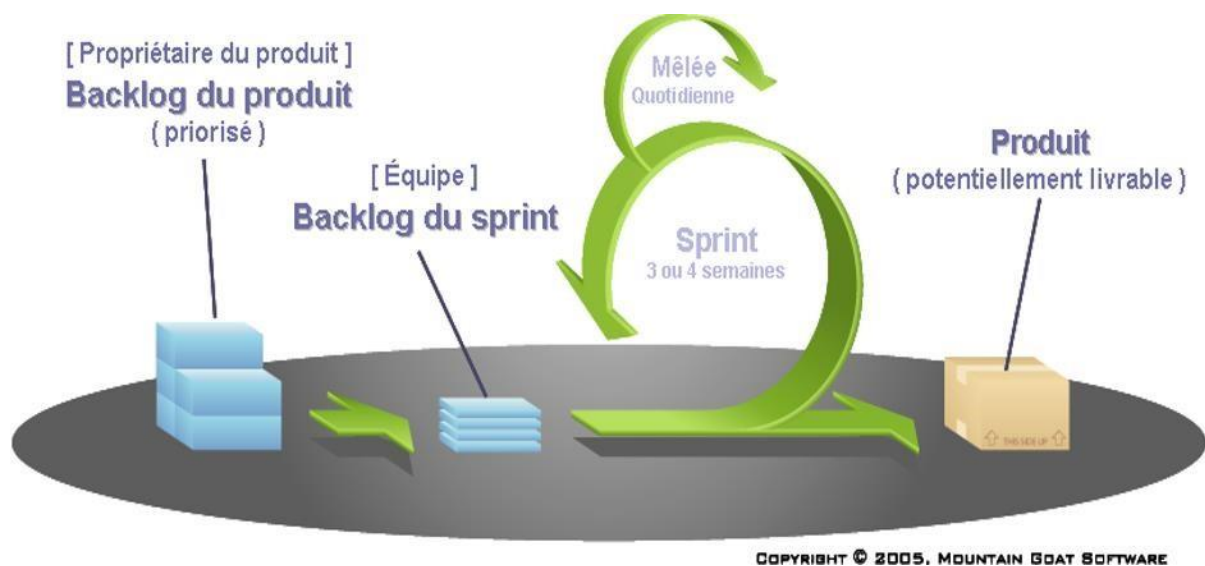


Figure 6-processus de SCRUM

Néanmoins, certaines limites sont à noter, scrum évolue avec une équipe compose d’au moins trois personnes idéalement : Product Owner, Scrum Master, Développeur.

Contrairement à SCRUM, le processus 2TUP n’implique pas forcément la participation de toute une équipe de travail et surtout revient en détail sur le comment s’organiser pour produire une application informatique. De plus, construit sur UML (Unified Modeling Language), 2TUP propose un tableau de correspondance renseignant sur les digrammes d’UML à produire à chaque phase, ce qui permet d’avoir une documentation riche, imagée et bien calqué pour chaque fonctionnalité.

Au terme de cette comparaison notre choix est porté sur la méthode agile et plus particulièrement le 2TUP. Il offre une meilleure gestion des risques grâce à des itérations fréquentes et à la possibilité d'ajustements continus. Dans ce projet, L'accent est mis sur la collaboration continue avec les parties prenantes et la possibilité d'ajuster les fonctionnalités en cours de route en réponse aux retours d'utilisateur.

II.2 Aperçu sur UML

II.2.1 Définitions et historique

L’Unified Modeling Language (UML) est un langage de modélisation graphique utilisé dans le domaine du génie logiciel pour représenter visuellement un système logiciel. Il fournit un ensemble de notations standardisées qui permettent aux développeurs de comprendre, concevoir, visualiser et documenter la structure et le comportement d'un système logiciel

complexe. L'UML utilise différents types de diagrammes pour représenter différents aspects d'un système [22]. Certains des diagrammes les plus couramment utilisés sont les diagrammes de cas d'utilisation, les diagrammes de classe, les diagrammes d'objets, les diagrammes de séquence, les diagrammes d'activités, les diagrammes d'états, les diagrammes de composants, et les diagrammes de déploiement. Le langage a émergé au milieu des années 1990 comme une réponse à la nécessité de normaliser les pratiques de modélisation dans le domaine du génie logiciel. Avant l'UML, diverses méthodes de modélisation, telles qu'OMT, Booch Method, et OOSE, coexistaient avec leurs propres notations et conventions [22] [24].

En 1994, les pionniers du génie logiciel James Rumbaugh, Grady Booch, et Ivar Jacobson ont collaboré pour unifier ces méthodes. Le résultat de cette collaboration a abouti à la création de l'UML, dont la première version a été publiée en 1995. Il a rapidement gagné en popularité et a été adopté comme norme de modélisation par l'Object Management Group (OMG) en 1997. Cette normalisation a renforcé la crédibilité de l'UML et a contribué à sa diffusion à grande échelle dans l'industrie du logiciel. Au fil des années, l'UML a évolué avec plusieurs versions successives, chacune introduisant de nouvelles fonctionnalités et améliorant la spécification. En 2005, la version 2.0 d'UML a été publiée, marquant une étape importante dans son développement [26].

Aujourd'hui, l'UML est largement utilisé pour la modélisation des systèmes logiciels. Sa normalisation par l'OMG a contribué à son adoption généralisée, et il continue d'être un outil essentiel pour la compréhension, la conception, et la documentation des systèmes complexes. Il demeure un pilier dans l'arsenal du génie logiciel et de la conception logicielle et nous sera très utile pour l'aboutissement de ce projet.

II.2.1.1 Le diagramme de cas d'utilisation

Le diagramme de cas d'utilisation (Use Case Diagram en anglais), est un élément fondamental de l'Unified Modeling Language (UML) et est l'un des types de diagrammes les plus couramment utilisés. Il offre une représentation visuelle des interactions entre les acteurs externes et un système logiciel. Il permet de visualiser les fonctionnalités du système du point de vue de l'utilisateur final, facilitant ainsi la compréhension des interactions essentielles entre les parties prenantes et le logiciel [26].

Dans ce type de diagramme, les acteurs, représentant des utilisateurs externes, des entités ou d'autres systèmes, interagissent avec le système à travers des cas d'utilisation spécifiques. Chaque cas d'utilisation décrit une séquence d'actions réalisées par le système en réponse à une requête de l'utilisateur. Ces interactions sont représentées par des flèches reliant les acteurs aux cas d'utilisation, définissant ainsi le périmètre fonctionnel du système.

II.2.1.2 Le diagramme de classes

Celui-ci est un type de diagramme UML qui décrit les classes et les interfaces d'un système ainsi que leurs relations. Il est utilisé pour modéliser la structure statique d'un système, c'est-à-dire sa structure à un instant donné. Une classe est un type d'objet, c'est-à-dire une collection d'attributs et d'opérations. Les attributs représentent les données stockées par un objet, tandis que les opérations représentent les comportements d'un objet. Un diagramme de classe est composé de classes, d'attributs, d'opérations et de relations entre classes, fournissant ainsi une vue claire de la composition et de l'organisation du système. Les composants clés d'un diagramme de classe comprennent les classes, représentées par des rectangles, chacun comportant le nom de la classe, les attributs, et les méthodes. Les relations entre les classes sont symbolisées par des lignes, avec des types courants tels que l'association, l'agrégation, et la composition. L'association indique une relation entre deux classes, l'agrégation signifie qu'une classe est composée de plusieurs autres classes, et la composition indique une composition plus forte [27].

La multiplicité, exprimée par des nombres ou des intervalles, spécifie le nombre d'objets qui peuvent participer à une relation. Les relations d'héritage, représentant l'héritage entre les classes, sont également incluses, montrant comment une classe peut hériter des attributs et des méthodes d'une autre classe.

Encourageant le concept d'encapsulation, le diagramme de classe illustre la visibilité des attributs et des méthodes, déterminant leur accessibilité depuis l'extérieur de la classe.

II.2.1.3 Le diagramme d'activité

Le diagramme d'activité, intégré à l'Unified Modeling Language (UML), est une représentation graphique qui capture le flux de contrôle entre différentes activités dans un système logiciel. C'est un outil puissant pour modéliser des processus métier, des workflows, des algorithmes, ou tout scénario impliquant des actions successives [28].

II.3. Analyse de notre module

II.3.1. Aperçu sur notre module

Dans la section 3.2 du chapitre 1, nous avons présenté brièvement les contours de nos travaux de mémoire dans le projet de mise en place du guichet médical unique. Dans cette partie, nous présentons de façon détaillée ces travaux qui consistent à développer le module « Family area » à partir duquel les patients accèdent à leurs dossiers médicaux.

Un « Family area » est un espace permettant de gérer les données médicales des membres d'une famille. Il est administré par un responsable qui gère (ajouter, modifier, supprimer) les membres et les accès, mais aussi les données ménagères et familiales (liens de parenté, mariages, fratrie, ...). Comme nous l'avons vu dans le chapitre 1, ces données (non médicales) seront versées dans les dossiers médicaux des patients.

Le module que nous proposons se greffe sur le « noyau » du guichet médical unique (cf. section 3.2 du chapitre 1) qui est déjà en place. Dans ce noyau, il est établi un modèle de dossier médical qui comporte toutes les données médicales permettant de constituer un dossier médical.

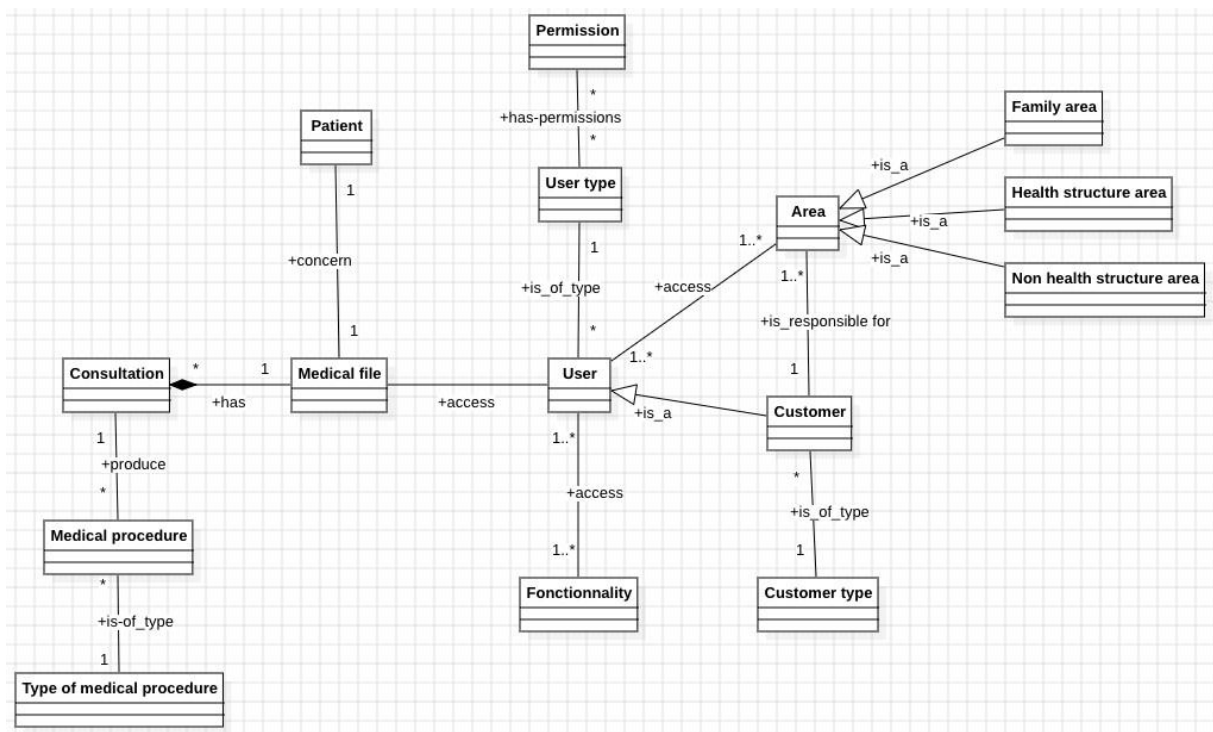


Figure 7. Première version du modèle de dossier médical ©GMU

Comme on peut le voir dans ce modèle, un dossier médical concerne un et un seul patient (un patient a aussi un et un seul dossier médical) et comporte pour le moment que des consultations médicales. Les consultations médicales produisent des actes médicaux qui sont les données médicales du patient.

Nos travaux consistent dans un premier temps à étendre ce modèle en y ajoutant les données familiales et ménagères (lesdites données sont présentées dans la section 4.1 de ce chapitre).

Dans un second temps, il s'agit de proposer un module permettant aux patients d'accéder (en lecture seule) à leurs dossiers médicaux, mais aussi de gérer (ajouter, modifier, supprimer) leurs données familiales et ménagères.

II.3.2 Les fonctionnalités et les acteurs

Le module que nous proposons de développer est constitué de 4 grandes fonctionnalités :

- Gérer les membres d'une famille :
 - Acteur : **Responsable de la « Family area »**. Dans le guichet médical unique, le responsable d'une « Family area » est un utilisateur (de type client) de la plateforme. Il correspond à l'entité « Customer » dans le modèle de dossier médical de la Figure 4 qui est un responsable d'un espace familial dans le système.
 - Description : La gestion des membres d'une famille est une grande fonctionnalité qui regroupe des sous fonctionnalités plus élémentaires : Ajouter un membre en renseignant ses parents ; modifier les informations sur un membre (lui associer un partenaire, modifier sa situation matrimoniale, ...) ; supprimer un membre ; visualiser sous forme de table la liste des membres ; visualiser les détails des informations sur un membre ; rechercher un membre.
- Générer l'arbre généalogique d'une famille :
 - Acteur : **Membre d'une « Family area »**. Chaque membre d'une famille est un utilisateur (entité « User » dans le modèle de dossier médical de la Figure 4) du guichet.
 - Description : Cette fonctionnalité permet aux membres d'une famille de pouvoir disposer de leur arbre généalogique médical et de pouvoir, par conséquent, bénéficier des services y afférents : historique ou trajectoire d'une maladie, maladies héréditaires, incompatibilité sanguine dans les couples, etc.

- Accéder au dossier médical :
 - Acteur : **Membre d'une « Family area »**.
 - Description : L'accès au dossier médical est une grande fonctionnalité qui regroupe des sous fonctionnalités plus élémentaires : Visualiser sous forme de table la liste des consultations médicales ; visualiser les détails d'une consultation médicale ; rechercher une consultation médicale ; rechercher un acte médical ; visualiser les détails d'un acte médical.
- Accorder un accès de son dossier médical à un médecin :
 - Acteur : **d'une « Family area »**.
 - Description : Cette fonctionnalité permet à un patient de donner accès de son dossier médical à un médecin. Pour cela, il génère un « token » qu'il transmet au médecin et qui garantit à ce dernier l'accès au dossier médical.

II.3.3 Les diagrammes de cas d'utilisation

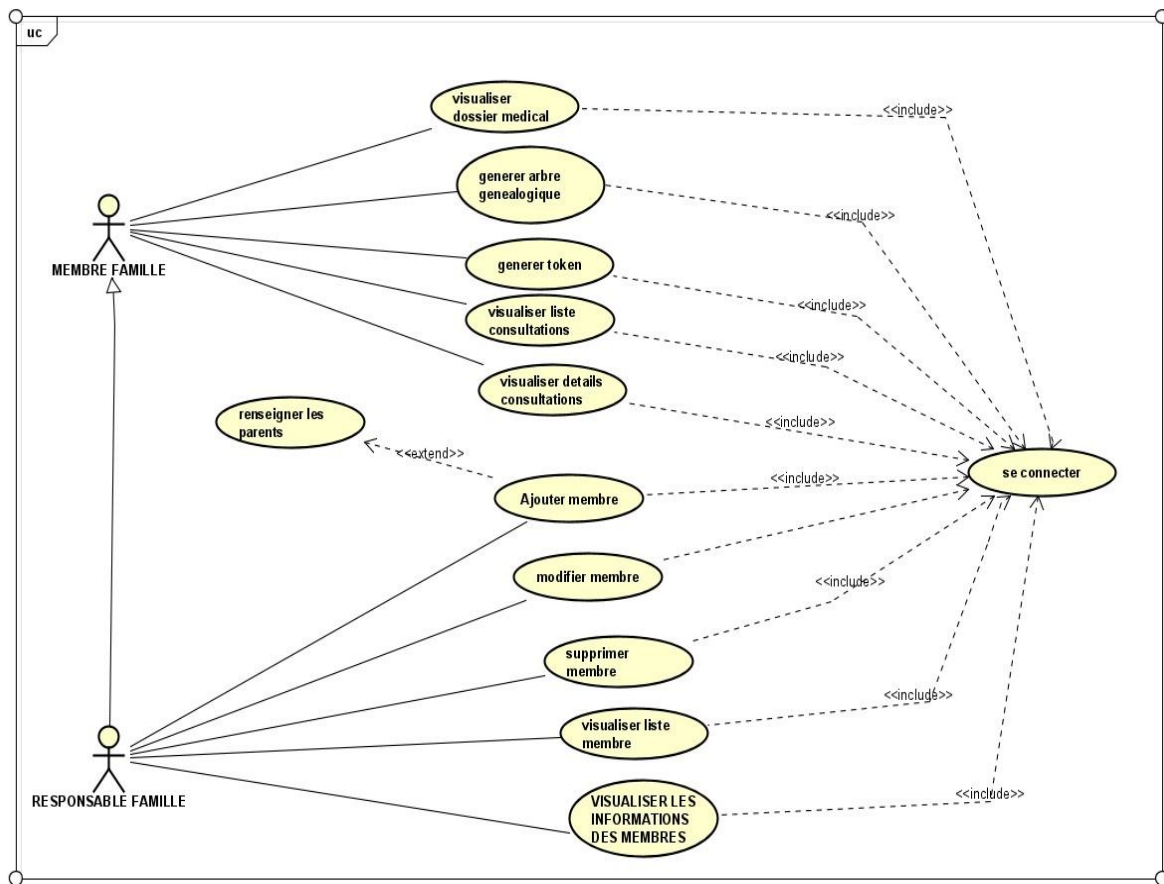


Figure 8-diagramme de cas d'utilisation du module famille © GMU

Dans ce schéma ci-dessus, nous avons deux acteurs principaux placés à gauche et liés par une relation de généralisation, le responsable de famille et le membre de la famille. Cette relation de généralisation signifie que toutes les fonctions accessibles au membre de la famille (pointé par la flèche) sont également accessibles au responsable de famille. Il est constaté dans notre diagramme de cas d'utilisation qu'il y a deux types de relation entre les cas d'utilisation que sont « include » et « extend ». Une relation est dite « includ » lorsqu'il faut obligatoirement exécuter le cas d'utilisation pointé par la flèche avant l'exécution du cas origine de la relation. Par exemple dans ce diagramme, pour « ajouter un membre dans une famille » il faudrait que le responsable se connecte. Par contre une relation est dite « extend » lorsque l'exécution du cas d'utilisation « renseigner famille » nécessite optionnellement l'exécution du cas d'utilisation « ajouter membre ».

II.4 Conception de notre module

II.4.1 Les données de notre module

Dans cette partie, nous proposons une liste des données (exprimées en termes de classes) de notre module.

| Classe | | Description |
|-------------------------------------|--|--|
| Patient | | Représente un patient (un membre d'une « Family area »). Elle comporte les données identitaires du patient et ses constantes médicales. |
| Marital Situation | | Représente les différentes situations matrimoniales possibles : célibataire, mariée, veuve, divorcée, ... |
| Marital Situation of Patient | | Représente les situations matrimoniales des membres d'une « Family area ». La situation matrimoniale d'un individu pouvant évoluer dans le temps, cette classe permet de représenter aussi cette évolution. En effet, quand la situation matrimoniale d'un individu change, la date de ce changement est renseignée. |
| Couple | | Représente une relation de couple entre deux membres d'une « Family area » pouvant engendrer une descendance. Elle comporte les deux individus en couple, la date formation du couple et les enfants issus de cette union. |
| Nature of union | | Représente les différentes situations conjugales possibles des couples : mariage, union libre, séparation, ... |
| Nature of Couple | | Représente les natures des couples dans une « Family area ». La situation conjugale d'un couple pouvant évoluer dans le temps, cette classe permet de représenter aussi cette évolution. En effet, quand la situation conjugale d'un couple change, la date de ce changement est renseignée. |

Tableau 1-Liste des classes représentant les données de notre module

II.4.2 Le diagramme de classes de notre module

Ce diagramme de classe est une extension du modèle de dossier médical présenté plus haut. Il comporte les données relatives à l'évolution de la situation conjugale du patient (les classes « Marital situation of patient » et « Nature of couple »), ses descendances, ses ascendances (et donc de son historique médical), ses liens familiaux (fratrie, polygamie, etc.) avec les autres patients. Même si ces données ne sont pas explicitement représentées, elles peuvent être déduites des données ici représentées.

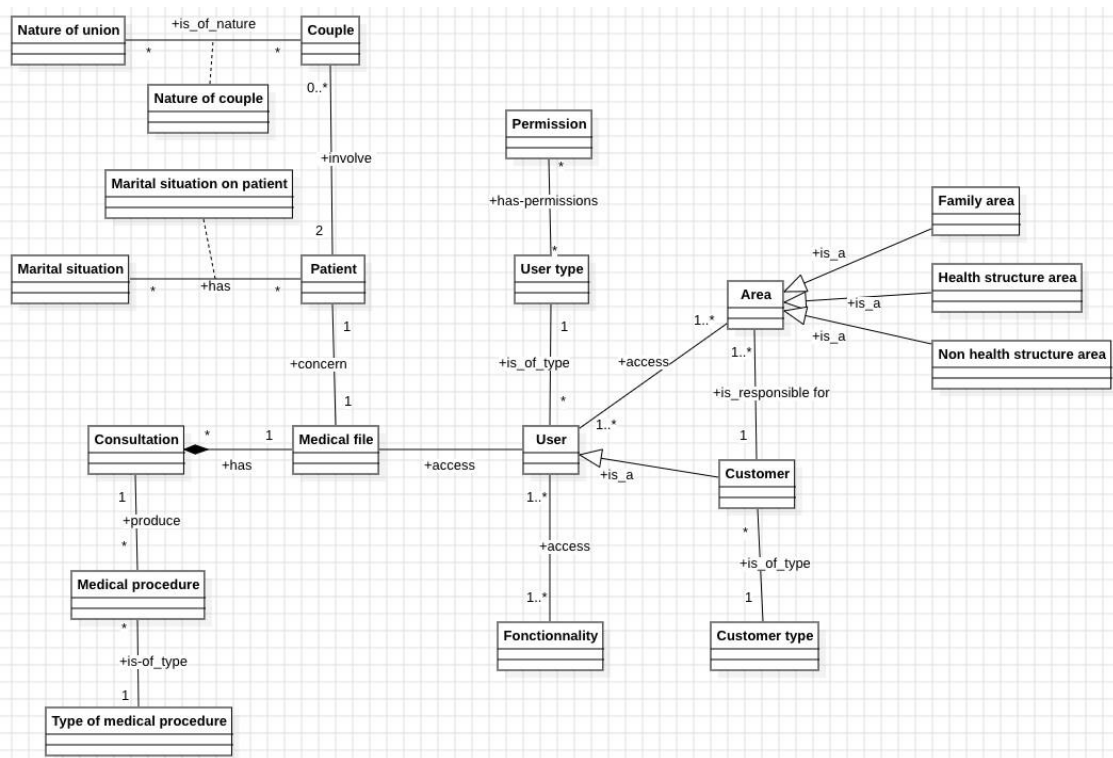


Figure 9. Modèle de dossier médical comportant les liens familiaux du patient GMU

Chapitre III. Implémentation et présentation de notre module

III.1 Outils et techniques de développement

Dans cette partie, nous allons mettre en avant l'ensemble des outils utilisés pour l'intégration du module. Pour concrétiser un projet d'application, l'exploration de diverses technologies s'avère incontournable. Cette exploration englobe différents aspects tels que l'environnement de travail, la programmation, la documentation etc. Et Face aux multiples approches disponibles pour chaque étape du projet, le processus de sélection des technologies à adopter revêt une importance cruciale. Il est donc essentiel de prendre en compte des critères spécifiques tels que la performance, la flexibilité et la compatibilité avec les besoins du projet. Ainsi, en se basant sur ces critères discernant et l'adaptabilité aux exigences spécifiques, certaines technologies émergent en tant que choix privilégiés parmi d'autres, conférant ainsi au projet une base solide

et une orientation optimale. Nous avons évalué ces derniers avec prudence et une vision lointaine ce qui nous a permis de faire le choix de ces outils présentes ci-dessous.

III.1.1 Le design pattern MVC

Le design pattern MVC (Modèle-Vue-Contrôleur) est un modèle architectural qui divise une application logicielle en trois composants interconnectés pour améliorer la séparation des préoccupations et la gestion de la complexité. Le motif MVC est largement utilisé dans le développement logiciel, en particulier dans les applications basées sur l'interface utilisateur. Il a été popularisé par le Framework Smalltalk-80 dans les années 1970 et est devenu un concept fondamental dans le développement logiciel [29].

- **Modèle (Model) :** Le Modèle représente la structure des données et la logique métier de l'application. Il traite des opérations liées à la manipulation des données, de leur validation et de leur mise à jour. Le Modèle est indépendant de l'interface utilisateur et des interactions directes avec l'utilisateur.
- **Vue (View)** La Vue est responsable de l'affichage des informations à l'utilisateur et de la présentation des données provenant du Modèle. Elle réagit aux changements dans le Modèle et met à jour l'interface utilisateur en conséquence. La Vue est généralement passivement à l'écoute des événements provenant du Contrôleur.
- **Contrôleur (Controller) :** Le Contrôleur agit comme un intermédiaire entre le Modèle et la Vue. Il gère les interactions de l'utilisateur, traite les entrées de l'utilisateur et met à jour le Modèle en conséquence. Il reçoit également des notifications du Modèle en cas de changements et met à jour la Vue en conséquence. Le Contrôleur est souvent associé à la logique métier [30].

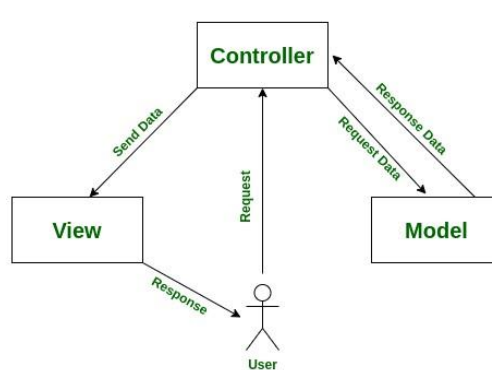


Figure 10- schéma descriptif Du MVC

Les avantages du MVC incluent une meilleure organisation du code, un changement facile sur une couche sans affecter les autres, une facilité de maintenance, une séparation claire des responsabilités, et la possibilité de réutiliser les composants dans d'autres parties de l'application. Vu la complexité du projet avec les différents modules, nous avons porté un choix sur ce modèle pour une meilleure prise en mains du logiciel.

III.1.2 Le Framework Django

Django est un Framework open-source pour le développement rapide d'applications web en utilisant le langage de programmation Python. Il a été conçu pour être pragmatique, propre et suivre le principe du DRY (Don't Repeat Yourself) ainsi que le principe de la convention plutôt que la configuration. Django utilise une architecture MTV, qui est similaire au modèle MVC (Modèle-Vue-Contrôleur). Dans le contexte de Django, les composants sont appelés Modèle (Model), Template (Template), et Vue (View).

- Modèle : Gère la logique métier et la persistance des données.
- Template : Gère la présentation et le rendu des données à l'utilisateur final. Architecture MTV (Modèle, Template, Vue) :
- Vue : Gère la logique de traitement des données et la communication entre le Modèle et le Template.

De même, Django propose un ORM intégré qui permet aux développeurs d'interagir avec la base de données à l'aide d'objets Python plutôt qu'en écrivant des requêtes SQL directes. Cela facilite grandement la gestion des données et rend l'application plus portable entre différents systèmes de gestion de bases de données. [31] Il fournit une interface d'administration générée automatiquement pour les modèles enregistrés, ce qui permet de gérer et de manipuler les données de la base de données sans avoir à créer une interface d'administration personnalisée. Garni d'un moteur de Template puissant, il permet de séparer la logique métier et la présentation des données et intègre des mécanismes de sécurité tels que la protection contre les attaques CSRF (Cross-Site Request Forgery), la gestion des sessions utilisateur, et la protection contre l'injection SQL. Django est extensible grâce à son architecture modulaire, donc permet l'intégration des fonctionnalités spécifiques à travers des applications tierces [32].



Figure 11- image de Django

Django est très souvent mis en confrontation avec Symfony, un Framework qui fonctionne avec du PHP. En faisant la comparaison basée sur le Langage de programmation on peut s'apercevoir de l'avantage de Django avec python sur les autres technologies énuméré dans le tableau ci-dessous.

| Paramètre | PHP | Python |
|-------------------------------|--|---|
| Type de langage | Langage de script pour le développement Web | Langage de programmation à usage général |
| Courbe d'apprentissage | Plus difficile à comprendre | Plus facile à comprendre pour un nouveau |
| Vitesse | | |
| Syntaxe | Complexe, utilise des accolades, des règles de nommage, des symboles | Claire, nette. N'utilise pas d'accolade et nécessite pas de déclarer une variable |

| | | |
|-----------------------------------|---|---|
| Utilisation | Principalement pour le développement web | Usage plus large, y compris le développement site web et d'applications ainsi que les technologies émergentes telles que l'IA, ML, la science de données, le Big data |
| Supports des bibliothèques | Offre packagist, qui est robuste, mais PHP est toujours à la traîne pour ce qui est de la prise en charge des bibliothèques | Vaste support de bibliothèque pour presque tous les types d'applications |
| Lisibilité | Moins facile et à entretenir | Il est conçu pour réduire les complexités et être très lisible avec un anglais simple |
| sécurité | De nombreuses attaques ont été observées | Plus sûr grâce aux fonctions de cybersécurité intégrées |

Tableau 2-Tableau comparatif de PHP et PYTHON

III.1.3 Le scaffolding

Le terme "scaffolding" dans le contexte du développement web, et plus spécifiquement dans le cadre de Framework comme Django, fait référence à la génération automatique de structures de fichiers, de code ou de composants pour faciliter le démarrage rapide d'un projet. Il est conçu pour fournir une base solide à partir de laquelle les développeurs peuvent construire leur application. Dans le cas de Django, le scaffolding est souvent associé à la création automatique de l'arborescence de fichiers de base pour un projet [33].

III.1.4 La base de données PostgreSQL

PostgreSQL est un système de gestion de base de données relationnelle (SGBDR) opensource et puissant. Il est reconnu pour sa robustesse, sa conformité aux normes SQL, ses fonctionnalités avancées, et sa capacité à gérer des charges de travail complexes. Il détient un bon nombre de fonctionnalités pouvant faciliter l'aboutissement de notre travail.

PostgreSQL offre une architecture extensible qui permet aux utilisateurs d'ajouter des types de données, des fonctions, des langages de programmation, et des gestionnaires d'index personnalisés [34].



Figure 12-logo PostgreSQL ©google

Dans ce projet, la confrontation a été faite entre PostgreSQL et MySQL, ils sont les principales sources de données de nombreuses fortes applications. Au Delas de leurs nombreuses ressemblances, il existe un certain nombre de caractéristiques qui les distinguent les uns des autres. Ces points sont éclaircis dans le tableau ci-dessous.

| | PostgreSQL | MYSQL |
|--|---|--|
| Technologie de base de données | PostgreSQL est un système de gestion de base de données relationnelle objet. | MySQL est un système de gestion de base de données purement relationnelle. |
| Types de données pris en charge | Numérique, date/heure, caractère, booléen, énuméré, géométrique, adresse réseau, JSON, XML, HSTORE, tableaux, plages, composite | Numérique, date/heure, caractère, spatial, JSON |

| | | |
|------------------------|---|--|
| Performance | Adapté aux applications avec un volume élevé de lectures et d'écritures | Convient aux applications avec un volume élevé de lectures |
| Securite | Contrôle d'accès, plusieurs options de connexion cryptées | Contrôle d'accès, connexions cryptées |
| Conformité ACID | PostgreSQL est toujours conforme à la norme ACID | MySQL ne répond aux propriétés ACID qu'avec les moteurs de stockage InnoDB et NDB Cluster. |

Tableau 3- Tableau comparatif MYSQL et POSTGRESQL

Pour notre cas, utiliser postgress est une évidence. L'application est programmée pour recevoir un nombre important d'utilisations avec une multitude de types de données.

III.2 Vue d'ensemble sur l'implémentation de notre module

Le module que nous avons développé est une « application Django » ajoutée dans ce que nous avons appelé « le noyau » du guichet dans la section 3.2 du chapitre I. En effet, dans la mise en place de ce « noyau »¹, un « projet Django » appelé « gmu-sn » (pour « Guichet Médical Unique du Sénégal ») a été créé et une application appelée « medicalfile » ajoutée.

Dans cette application, il est établi un « modèle du dossier médical » tel que présenté dans le diagramme de classes de la figure 8 ; un module « Access Manager » qui gère les accès aux dossiers médicaux et un module d'administration général (back-office) du guichet qui permet d'administrer les données de paramétrage ; les utilisateurs ; les profils utilisateurs et les autorisations sur le guichet, sont aussi mis en place.

Pour mettre en place une plateforme avec Django, il faut d'abord créer un « projet Django », puis y ajouter des « applications Django » qui constituent les modules applicatifs de la plateforme. Ensuite, développer/implémenter un « projet » dans Django, c'est développer/implémenter une-à-une chaque application ajoutée. Le développement

« d'application Django » consiste à implémenter chacun des composants du MVT : le modèle, les Vues et les Template.

Pour notre cas, un « projet Django » appelé « gmu-sn » est déjà en place. Nous y avons ajouté une application nommée « familyarea ». Dans la suite, nous présentons les détails d'implémentation de notre application « familyarea » en présentant l'implémentation du « modèle », des « views » et des « templates ».

III.2.1 Implémentation du modèle

Dans Django, le développement du "Model" d'une application consiste à ajouter les différentes classes « métiers » de l'application dans le fichier "models.py". A priori, pour chaque classe ajoutée dans ce fichier, une table est créée par l'ORM Django dans la base de données. Ce qui permet d'enregistrer et de pérenniser un objet d'une classe donnée dans la table correspondante.

L'ensemble des classes présentées dans le Tableau 1 (exceptée la classe « Patient » qui existe déjà dans l'application « medicalfile ») sont donc ajoutées dans le fichier « models.py » de l'application « familyarea », comme on peut le voir dans la figure suivante :

1 Il s'agit d'un travail antérieur au notre qui a fait l'objet d'un autre mémoire dont le sujet est « Vers la mise en place d'un "Guichet Médical Unique" pour le Sénégal ».

```
▶ class MaritalSituation(models.Model): ...  
▶ class MaritalSituationOfPatient(models.Model): ...  
▶ class Couple(models.Model): ...  
▶ class NatureOfUnion(models.Model): ...  
▶ class NatureOfCouple(models.Model): ...
```

Figure 13-Les différentes classes "métier" de l'application "familyarea"

III.2.2 Implémentation des vues

Une vue ("View") est une fonction définie dans un fichier « views.py » et qui reçoit comme paramètres la requête (l'URL) à partir de laquelle elle est invoquée et d'éventuels autres paramètres à utiliser dans les traitements de la fonction. Les traitements effectués par la vue

sont donnés dans le corps de la fonction. La vue rend enfin un Template (une page HTML) qui correspond à l'url de départ ou une redirection vers une autre vue.

Concernant l'application « familyarea », nous avons les fonctionnalités suivantes qui correspondent aux "URLs" dans le fichier "urls.py" (Figure 14) et donc aux "View" dans le fichier "views.py" (Figure 15) de l'application :

```
urlpatterns = [  
    path('manageMembers/', views.members),  
    url(r'^manageMembers/delete/(?P<idMember>[0-9]+)/$', views.delete),  
    url(r'^manageMembers/edit/(?P<idMember>[0-9]+)/$', views.edit),  
    url(r'^manageMembers/file/(?P<idMember>[0-9]+)/$', views.file),  
    url(r'^manageMembers/file/(?P<idMember>[0-9]+)/o/(?P<idConsultation>[0-9]+)/$', views.detailsConsultation),  
  
    path('genealogie/sf/u/', views.manageUnion),  
    url(r'^genealogie/sf/u/delete/(?P<idUnion>[0-9]+)/$', views.deleteUnion),  
    url(r'^genealogie/sf/u/edit/(?P<idUnion>[0-9]+)/$', views.editUnion),  
    url(r'^genealogie/sf/u/details/(?P<idUnion>[0-9]+)/$', views.detailsUnion),  
    url(r'^genealogie/sf/u/details/(?P<idUnion>[0-9]+)/d/sf/(?P<idSF>[0-9]+)/$', views.deleteSF),  
    url(r'^genealogie/sf/u/details/(?P<idUnion>[0-9]+)/d/desc/(?P<idDESC>[0-9]+)/$', views.deleteDESC),  
]
```

Figure 14-Urls correspondant aux fonctionnalités de l'application "familyarea" dans son fichier « urls.py »

- Gestion des membres ("manageMembers"). Il s'agit de la fonctionnalité permettant de gérer les membres d'une famille. Elle est composée des sous fonctionnalités suivantes :
 - "manageMembers/". Il s'agit de l'accueil dans la gestion des membres permettant de visualiser la liste des membres d'une famille dans un tableau d'une part et, d'autre part, d'ajouter un membre.
 - "manageMembers/delete/---". Il s'agit de la fonctionnalité permettant de retirer de la famille un membre qui avait été ajouté.
 - "manageMembers/edit/---". Il s'agit de la fonctionnalité permettant de mettre à jour les informations sur un membre.
 - "manageMembers/file/---". Il s'agit de la fonctionnalité permettant d'ouvrir un dossier médical et de visualiser l'historique des consultations.
 - "manageMembers/file/---/o/---". Il s'agit de la fonctionnalité permettant de visualiser les détails d'une consultation.

- Gestion de la généalogie ("genealogie/sf/u"). Il s'agit de la fonctionnalité permettant de gérer l'arbre généalogique des membres d'une famille. Elle est composée des sous fonctionnalités suivantes :

- "genealogie/sf/u". Il s'agit de l'accueil dans la gestion de la généalogie des membres permettant de visualiser la liste des couples d'une famille dans un tableau d'une part et, d'autre part, d'ajouter un couple.
- " genealogie/sf/u/delete/---". Il s'agit de la fonctionnalité permettant de retirer de la famille un couple qui avait été ajouté.
- "manageMembers/edit/---". Il s'agit de la fonctionnalité permettant de mettre à jour les informations sur un couple.
- "genealogie/sf/u/details/---". Il s'agit de la fonctionnalité permettant, d'une part, de visualiser les détails sur un couple : historiques des situations (évolutions) du couple, les descendances du couple. D'autre part, elle permet d'ajouter une situation évolutive du couple, mais aussi ajouter des descendances du couple.
- "genealogie/sf/u/details/---/d/sf/---". Il s'agit de la fonctionnalité permettant de supprimer une situation évolutive d'un couple qui avait été ajouté par erreur.
- "genealogie/sf/u/details/---/d/des/---". Il s'agit de la fonctionnalité permettant de supprimer une descendance d'un couple qui avait été ajouté par erreur.

Les vues ("view") correspondant à ces URLs sont les suivantes :

```

@login_required
@permission_required('familyarea.view_members')
@permission_required('familyarea.add_member')
▶ def members(request): ...

@login_required
@permission_required('familyarea.edit_member')
▶ def edit(request, idMember): ...

@login_required
@permission_required('familyarea.delete_member')
▶ def delete(request, idMember): ...

@login_required
@permission_required('medicalfile.view_file')
▶ def file(request, idMember): ...

@login_required
@permission_required('medicalfile.view_file')
▶ def detailsConsultation(request, idMember, idConsultation): ...

@login_required
@permission_required('familyarea.view_couples')
@permission_required('familyarea.add_couple')
▶ def manageUnion(request): ...

@login_required
@permission_required('familyarea.edit_couple')
▶ def editUnion(request, idUnion): ...

@login_required
@permission_required('familyarea.delete_couple')
▶ def deleteUnion(request, idUnion): ...

@login_required
@permission_required('familyarea.details_couple')
▶ def detailsUnion(request, idUnion): ...

@login_required
@permission_required('familyarea.deleteSF')
▶ def deleteSF(request, idUnion, idSF): ...

@login_required
@permission_required('familyarea.deleteDESC')
▶ def deleteDESC(request, idUnion, idDES): ...

```

Figure 15-Les vues de l'application "familyarea" dans son fichier "views.py"

Il convient de remarquer les "décorateurs" placés devant les fonctionnalités (« views ») qui permettent de restreindre l'accès aux utilisateurs connectés et qui possèdent les permissions appropriées. Par exemple, pour accéder à la fonctionnalité permettant de modifier un membre (« edit »), il faut que l'utilisateur soit connecté (« @login_required ») et qu'il possède la permission de modifier un membre (« @permission_required('familyarea.edit_member') »). Ces permissions sont accordées depuis l'espace d'administration général du guichet.

III.2.3 Implémentations des Template

Les « Template » associées aux vues de l'application « familyarea » sont des pages « HTML » dont nous choisissons de présenter les « rendus » (à la section 3 suivante) au lieu de se pencher sur les « codes ».

III.3 Présentation de notre module

III.3.1 Gestion des membres

La page d'accueil de cette fonctionnalité permet d'ajouter un membre et de visualiser, sous forme de tableau, la liste des membres ajoutés.

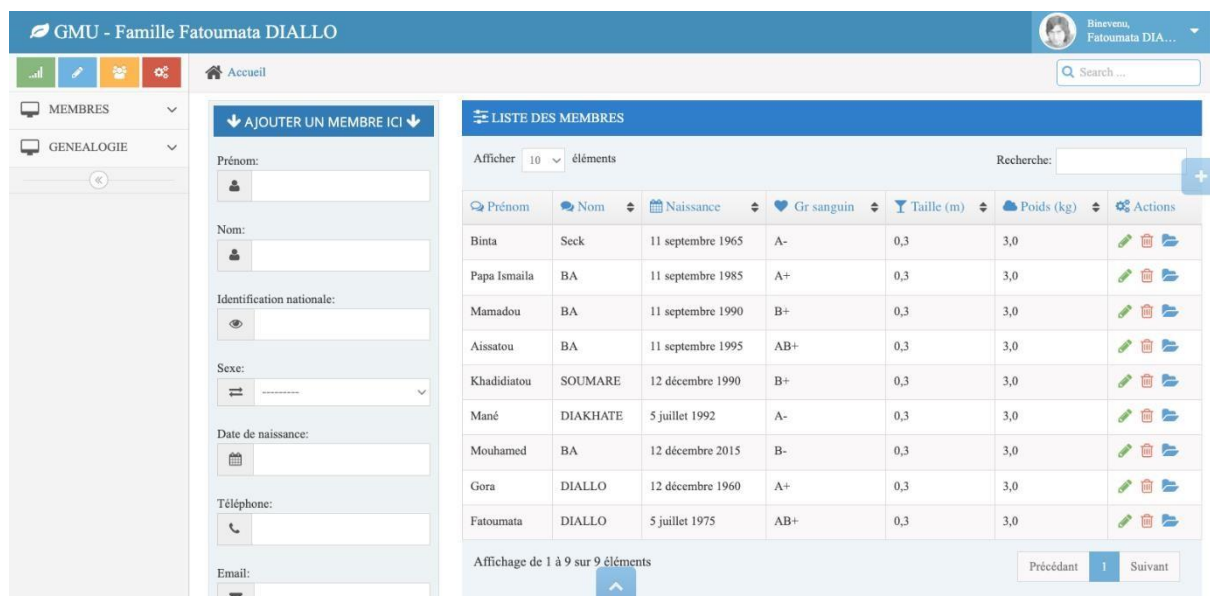


Figure 16-Fenêtre de visualisation et d'ajout des membres d'une famille

La partie gauche de cette fenêtre comporte un formulaire permettant de renseigner les informations sur un membre et de l'ajouter. Les membres ajoutés sont visualisés sur la partie droite de la fenêtre. A côté de chaque membre, figurent trois boutons permettant d'effectuer les actions suivantes :

- Mettre à jour un membre (le « crayon »). En cliquant sur ce bouton, les informations sur le membre sont chargées dans le formulaire de droite pour permettre des modifications.
- Retirer un membre (la « poubelle »). En cliquant sur ce bouton, le membre concerné est retiré. Cette fonctionnalité est surtout utile quand un membre a été ajouté par erreur par exemple.
- Ouvrir le dossier médical d'un membre (le bouton sous forme de « dossier »). En cliquant sur ce bouton, le dossier médical du patient est ouvert et l'historique de ses consultations est présenté sous forme de tableau comme dans la figure suivante :

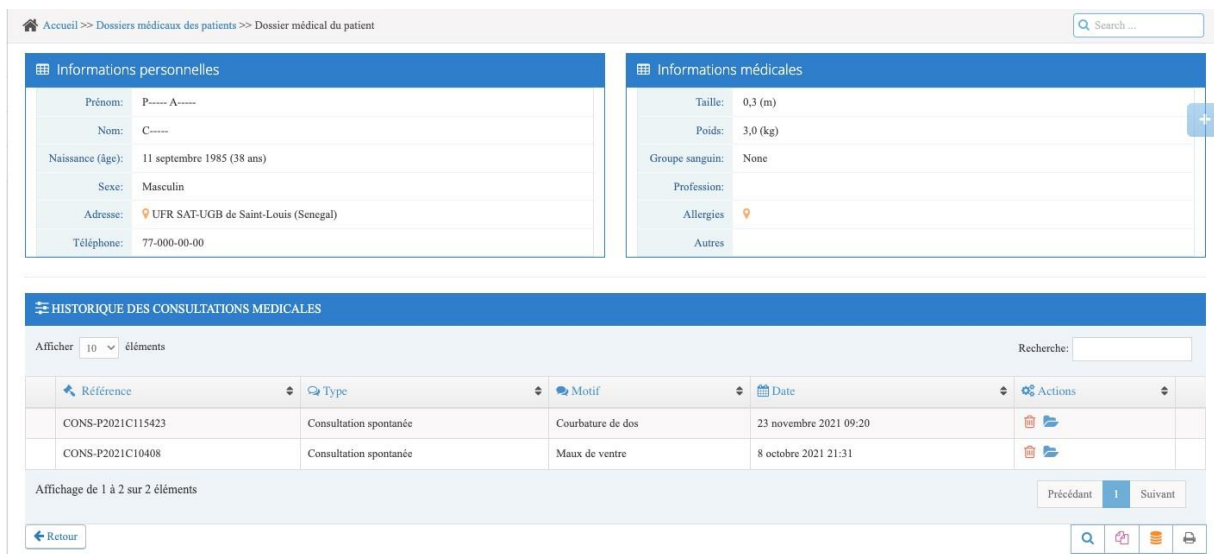


Figure 17-Fenêtre de visualisation du dossier médical d'un membre

Cette fenêtre permet de visualiser les informations personnelles et les constantes médicale d'un membre d'une part, d'autre part, l'historique de ses consultations médicales. En cliquant sur le bouton sous forme de « dossier » à coté de chaque consultation, les détails de la consultation sont ouverts comme dans la figure suivante :

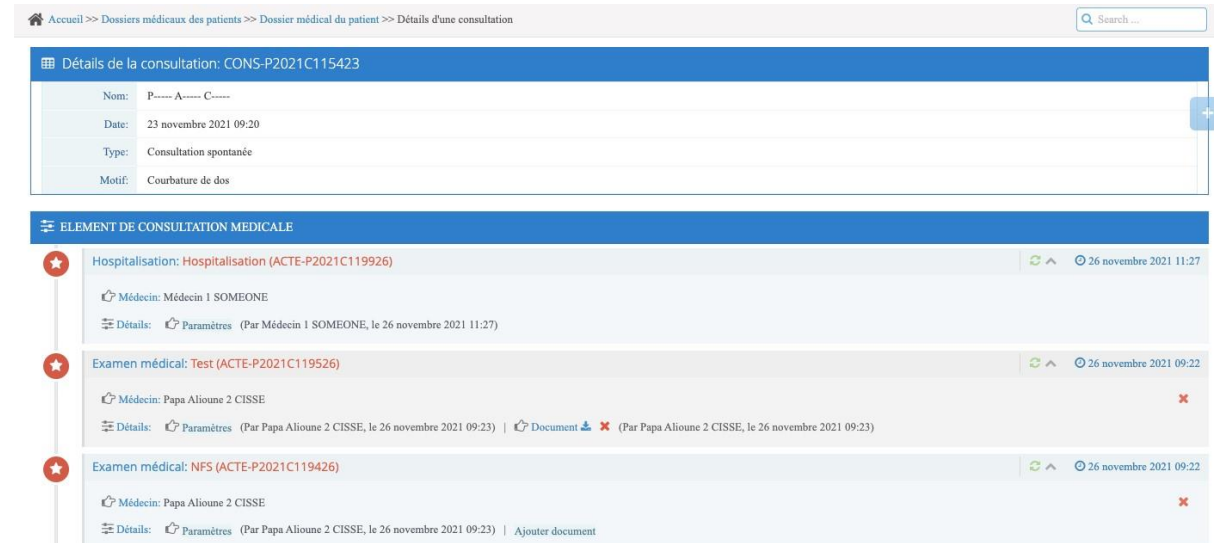


Figure 18-Fenêtre de visualisation des détails d'une consultation médicale

Cette fenêtre montre tous les actes médicaux relatifs à une consultation médicale.

III.3.2 Gestion de la généalogie

La page d'accueil de cette fonctionnalité permet d'ajouter un couple et de visualiser, sous forme de tableau, la liste des couple ajoutés.

LISTE DES UNIONS DANS LA FAMILLE

| Union | Date de l'union | Actions |
|---|------------------------|----------|
| Couple (Gora DIALLO et Binta Seck) | 19 décembre 1084 00:00 | [Icones] |
| Couple (Gora DIALLO et Aissatou BA) | 19 décembre 1995 00:00 | [Icones] |
| Couple (Papa Ismaïla BA et Khadidiatou SOUMARE) | 19 décembre 2013 00:00 | [Icones] |
| Couple (Mamadou BA et Mané DIAKHATE) | 19 décembre 2017 00:00 | [Icones] |

ARBRE GENEALOGIQUE DE LA FAMILLE

```

    graph TD
      Gora --- Aissatou
      Gora --- Binta
      Aissatou --- PapaIsmaila[Papa Ismaïla]
      Aissatou --- Khadidiatou
      Binta --- Mamadou
      Binta --- Mané
      Binta --- Fatoumata
      Khadidiatou --- Mouhamed
  
```

Figure 19-Fenêtre de gestion des couples dans une famille

La partie supérieure de cette fenêtre comporte à sa gauche un formulaire permettant d'ajouter un couple et dans sa partie droite la liste des couples ajoutés. La partie inférieure de cette fenêtre permet de visualiser l'arbre généalogique généré à partir des informations ajoutées.

En cliquant sur le bouton sous forme de « dossier » à côté d'un couple, les détails (descendances, situations évolutives) sur ce couple sont affichés sur la fenêtre suivante :

SITUATIONS FAMILIALES DE : Couple (Gora DIALLO et Binta Seck)

| Situation | Date | Actions |
|---------------|------------------------|----------|
| Union - Union | 19 décembre 1084 00:00 | [Icones] |

ENFANTS DIRECTS DE : Couple (Gora DIALLO et Binta Seck)

| Enfant | Date de naissance | Actions |
|------------------|-------------------|----------|
| Fatoumata DIALLO | 5 juillet 1975 | [Icones] |
| Papa Ismaïla BA | 11 septembre 1985 | [Icones] |
| Mamadou BA | 11 septembre 1990 | [Icones] |
| Aissatou BA | 11 septembre 1995 | [Icones] |

Figure 20-Fenêtre de gestion d'un couple

C'est aussi à partir de cette fenêtre que les descendances d'un couple sont ajoutées.

III.3.3 Fenêtres d'authentification

Les Fenêtre de connexion à la plateforme permettent aux membres de la famille de s'authentifier, de changer et/ou de récupérer le mot de passe (fig. 21-22-23).



The screenshot shows the login interface for 'GMU Gestion Utilisateurs'. At the top, the logo 'GMU' is followed by 'Gestion Utilisateurs' and '© GMU'. In the top right corner, there are links for 'Dark / Blur / Light'. The main content area features a white box with a green key icon and the text 'Entrez vos informations de connexion !'. Below this, there are two input fields: 'Adresse électronique :' and 'Mot de passe :'. A red message 'Merci de vous être connecté' is displayed above a blue 'Se connecter' button. At the bottom of the white box, there is a blue link '← Mot de passe oublié ?'.

Figure 21-Fenetre de connexion au GMU



The screenshot shows the password recovery interface for 'GMU Gestion Utilisateurs'. At the top, the logo 'GMU' is followed by 'Gestion Utilisateurs' and '© GMU'. In the top right corner, there are links for 'Dark / Blur / Light'. The main content area features a white box with a green key icon and the text 'Recupération de mot de passe: entrez votre email !'. Below this, there is a single input field for the email address. At the bottom of the white box, there is a blue 'Modifier' button.

Figure 22- Fenêtre de changement de mot de passe

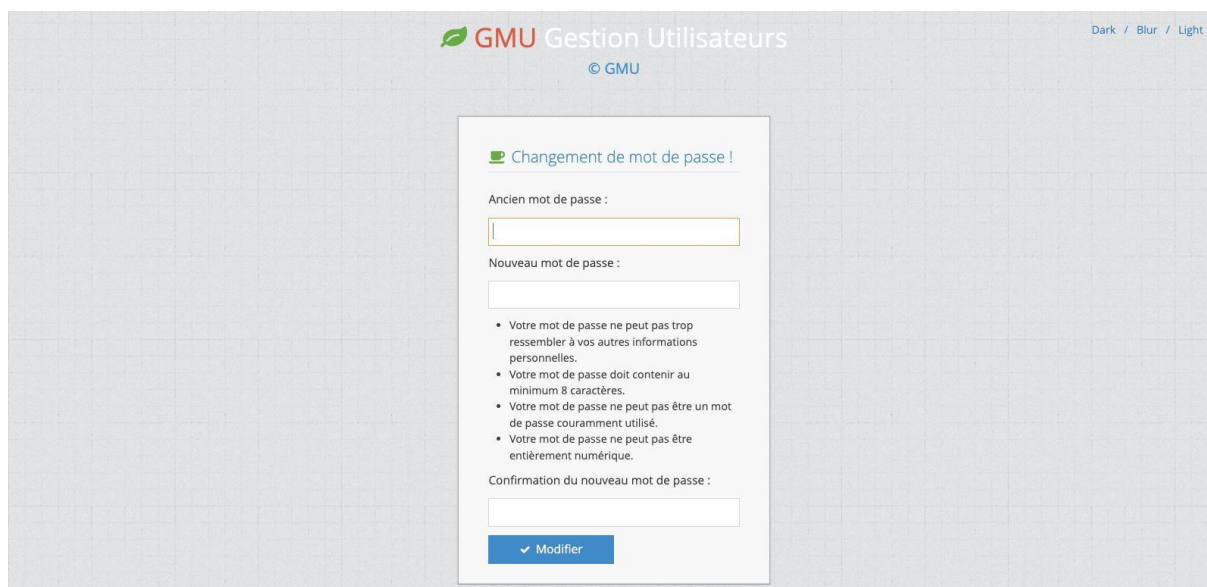


Figure 23-fenêtre de récupération de mot de passe

CONCLUSION ET PERSPECTIVE

A terme de ce travail de mémoire, nous rappelons que l'objectif de était de concevoir le module d'espace familial de l'application de gestion des données médicale partagées (GMU). La mise en place de ce module permettra au patient d'avoir une vue plus large de sa situation sanitaire et de son ascendance. Il permet aussi aux personnels de santé d'analyser les informations sanitaires d'une famille dans le but de comprendre la prévenance d'une maladie et/ou d'en prévenir. Pour parvenir à un tel résultat, plusieurs questions sont traitées au préalable. Ce qui nous a permis, dans une logique de structuration de diviser notre travail en trois chapitres après avoir dégagé le contexte, la problématique et les objectifs. Dans le premier chapitre, nous avons montré l'existant et défini de manière détaillé les exigences du projet. Dans le deuxième chapitre, il a été question de choix sur la méthodologie de travail à adopter entre l'agile et le traditionnelle, ainsi que d'effectuer un cadrage de notre module. Pour finir, le dernière chapitre a permis de faire une étude comparative des technologies utilisés, de montrer une partie du code et enfin de faire la présentation de notre solution.

La mise en œuvre de ce module constitue l'innovation dans cette application car elle prend en considération les liens de parenté au sein d'une famille. Elle ouvre de nouvelles perspectives

pour la prévention des maladies génétiques, la personnalisation des soins de santé et la recherche médicale, contribuant ainsi à une approche plus éclairée et individualisée de la santé. Tout d'abord, la collecte de données devient plus complète et significative en intégrant les antécédents médicaux familiaux dans l'application. Cela permet de créer un tableau exhaustif de la santé génétique de la famille, identifiant les schémas et les tendances qui peuvent être cruciaux pour anticiper les risques génétiques. L'analyse des risques génétiques est grandement améliorée, car l'application facilite la reconnaissance des facteurs de risque héréditaires. En comprenant les liens familiaux, il devient possible d'anticiper les maladies génétiques et de mettre en place des mesures préventives avant même l'apparition de symptômes. Sur le plan de la recherche médicale, une telle base de données enrichie en informations familiales permet d'approfondir la compréhension des mécanismes génétiques sous-jacents aux maladies. Les chercheurs peuvent utiliser ces données longitudinales pour identifier des corrélations, comprendre l'héritabilité des conditions médicales et développer des thérapies plus ciblées répondant à une problématique énumérée dans la première partie. La personnalisation des plans de prévention constitue un autre avantage notable, en utilisant au-delà des données génétiques, les professionnels de la santé peuvent élaborer des plans de prévention spécifiques à chaque individu, en tenant compte de ses prédispositions génétiques avec la géolocalisation de chaque patient. Cela ouvre la voie à une médecine préventive plus précise et efficace.

Le conseil génétique bénéficie également de cette approche intégrée. Les informations sur les liens de parenté permettent aux conseillers génétiques de fournir des recommandations plus précises et adaptées aux risques spécifiques d'une famille.

Enfin, cette application offre la possibilité de suivre l'évolution des données médicales sur le long terme. En établissant une base de données, il devient possible de suivre l'évolution des maladies génétiques à travers les générations, d'ajuster les plans de prévention en fonction des nouvelles découvertes médicales et d'adapter les stratégies de soins en conséquence.

Bibliographie et Webographie

- [1] P.-M. David, « La santé : un enjeu de plus en plus central dans les politiques publiques de développement international? », *Socio-logos . Revue de l'association française de sociologie*, n° 6, Art. n° 6, mai 2011, doi: 10.4000/socio-logos.2550.
- [2] « Système d'information dans l'entreprise : définition - Expert Linux », Syloe. Disponible sur: <https://www.syloe.com/glossaire/systeme-dinformation/>
- [3] « [4bc86eb39.pdf](https://www.unhcr.org/sites/default/files/legacypdf/4bc86eb39.pdf) ». Disponible sur: <https://www.unhcr.org/sites/default/files/legacypdf/4bc86eb39.pdf>
- [4] « TIC et santé numérique », ITU. Disponible sur <https://www.itu.int:443/fr/mediacentre/backgrounders/Pages/icts-digital-health.aspx>
- [5] Auteur Feriel GHARBI, « Tout savoir sur la M-santé », Faculté Privée des Sciences de la Santé en Tunisie. Disponible sur: <https://upsat.tn/culture/non-classifiee/tout-savoirusur-la-m-sante/>
- [6] « E-santé & m-santé », Haute Autorité de Santé. Disponible sur: https://www.hassante.fr/jcms/p_3106128/fr/e-sante-m-sante
- [7] « Généalogie médicale », *Wikipédia*. 3 décembre 2023. Disponible sur: https://fr.wikipedia.org/w/index.php?title=G%C3%A9n%C3%A9alogie_m%C3%A9dicale&oldid=210241113
- [8] « Dossier médical ». Disponible sur: <https://www.servicepublic.fr/particuliers/vosdroits/F12210>
- [9] « Le contenu du dossier médical - Qu'est-ce que c'est ? », Figaro Santé. Disponible sur: <https://sante.lefigaro.fr/social/droit/contenu-dossier-medical/quest-ce-que-cest>
- [10] C. Gekiere et S. Soudan, « Dossier patient informatisé et confidentialité : évolution des modèles et des pratiques. "Le diable gît dans les détails" », *L'information psychiatrique*, vol. 91, n° 4, p. 323-330, 2015, doi: 10.1684/ipe.2015.1336.
- [11] P. Mielotte et B. Paque, « " Etude sur le partage des données patients dans les dossiers médicaux en Belgique à travers le plan européen e-santé », Disponible sur: https://dial.uclouvain.be/downloader/downloader.php?pid=thesis%3A15463&datastream=PDF_02
- [12] « Guichet unique », *Wikipédia*. 10 mars 2023. Disponible sur: https://fr.wikipedia.org/w/index.php?title=Guichet_unique&oldid=202169778
- [13] « Artificial Intelligence (AI): What Is AI and How Does It Work? | Built In ». Disponible sur: <https://builtin.com/artificial-intelligence> [14] « [plsantedigitale.pdf](https://www.sante.gouv.sn/sites/default/files/plsantedigitale.pdf) ». Disponible sur: <https://www.sante.gouv.sn/sites/default/files/plsantedigitale.pdf>
- [15] « La CSSDOS », Portail ESANTE - CSSDOS. Disponible sur: <https://www.esante.sn/lacssdos-sante-digitale/>
- [16] « Structures Archive », Portail ESANTE - CSSDOS. Disponible sur: <https://www.esante.sn/structure/>
- [17] « Together let's digitize your business ». Disponible sur: <https://eyone.net/en/>

- [18] « A Propos – SIMENS ». Disponible sur: <https://www.simens.sn/a-propos/>
- [19] « Cifrecom – Cifrecom ». Disponible sur: <http://cifrecom.com/>
- [20] « District Health Information System 2 (DHIS2) | Open Health News ». Disponible sur: <https://www.openhealthnews.com/resources/district-health-information-system-2-dhis2>
- [21] « DHIS2 », HISP-SA.. Disponible sur: https://www.hisp.org/?page_id=5091
- [22] « 1-UML_Besoins.pdf ». Disponible sur: https://home.mis.u-picardie.fr/~furst/docs/1UML_Besoins.pdf
- [23] Unknown, « View: Merise (informatique) », View. Disponible sur: <https://webindep.blogspot.com/2013/08/merise-informatique.html>
- [24] « Comparaison MERISE UML | PDF | Scrum (développement) | Architecture de logiciel ». Disponible sur: <https://fr.scribd.com/document/319995380/Comparaison-MERISE-UML>
- [25] « Cisse - Conception et Implémentation d'un Système de Gesti.pdf ». Disponible sur: https://rivieresdusud.uasz.sn/bitstream/handle/123456789/1766/ciss%c3%a9_memoire_2023.pdf?sequence=1&isAllowed=y
- [26] « What is Unified Modeling Language (UML)? ». Disponible sur: <https://www.visualparadigm.com/guide/uml-unified-modeling-language/what-is-uml/>
- [27] « Qu'est-ce qu'un diagramme de classes UML ? », Lucidchart. Disponible sur: <https://www.lucidchart.com/pages/fr/diagramme-de-classes-uml>
- [28] « Qu'est-ce qu'un diagramme d'activité UML ? », Lucidchart. Disponible sur: <https://www.lucidchart.com/pages/fr/diagramme-dactivite-uml>
- [29] B. Wicht, « Implémentation du pattern MVC- », Developpez.com. Disponible sur: <https://baptiste-wicht.developpez.com/tutoriel/conception/mvc/>
- [30] « MVC Design Pattern », GeeksforGeeks. Consulté le: 16 décembre 2023. [En ligne]. Disponible sur: <https://www.geeksforgeeks.org/mvc-design-pattern/>
- [31] W. S. Vincent, *Django for Beginners: Build websites with Python and Django*. WelcomeToCode, 2022.
- [32] J. Forcier, P. Bissex, et W. J. Chun, *Python Web Development with Django*. AddisonWesley Professional, 2008.
- [33] « Scaffold (programming) », *Wikipédia*. 2 juillet 2023. Disponible sur: [https://en.wikipedia.org/w/index.php?title=Scaffold_\(programming\)&oldid=1162988578](https://en.wikipedia.org/w/index.php?title=Scaffold_(programming)&oldid=1162988578)
- [34] « What is PostgreSQL? – PostgreSQL Database Explained - AWS », Amazon Web Services, Inc . Disponible sur: <https://aws.amazon.com/rds/postgresql/what-ispostgresql/>