



**CENTRE HOSPITALIER
INTERCOMMUNAL**
CASTRES-MAZAMET

Projet Tutoré
Sujet : PECM CHIC

RAPPORT SEMESTRE 2

Erreurs médicamenteuses :
Comment optimiser la gestion des risques ?

Rédigé par :

Séraphie MAURY

Salomé CHEVAILLER

Sarah GROS

Années universitaires 2021/2022

REMERCIEMENTS

Nous tenions tout d'abord à remercier l'ensemble du CHIC de Castres-Mazamet de nous avoir proposé ce projet tutoré.

Nous souhaitons remercier plus particulièrement le Dr CLEOSTRATE et le Dr CUFI pour le temps qu'elles nous ont accordé tout le long du projet ainsi que leur aide précieuse.

Nous remercions bien évidemment Mme THABET pour son accompagnement et son implication du début à la fin.

Pour finir, nous remercions nos camarades Gauthier POUS et Nelson ROGERS pour le travail qu'ils ont fourni avec Séraphie MAURY au premier semestre.

GLOSSAIRE

CHIC : Centre hospitalier intercommunal

Comité de retour d'expérience (CREX) : groupe pluriprofessionnel ayant pour missions de réduire les risques et d'augmenter la sécurité des soins

Diagramme de Gantt : outil utilisé en gestion de projet permettant de visualiser dans le temps les diverses tâches composant un projet

Erreur médicamenteuse (EM) : omission ou réalisation d'un acte non intentionnel impliquant un médicament durant le processus de soins

HTML (Hypertext Markup Language) : langage de balisage conçu pour représenter les pages web

Java : langage de programmation orienté objet

JavaScript : langage de programmation de scripts employé dans les pages web interactives

Méthode BPRIM : méthode de gestion intégrée des risques et des processus métiers

Microsoft SQL Server : système de gestion de base de données en langage SQL

Modèle relationnel de données : manière de modéliser les relations existantes entre plusieurs informations et de les ordonner entre elles

MySQL : système de gestion de base de données relationnelles

Never event : événement qui ne devrait jamais se produire

PHP (PHP Hypertext Preprocessor) : langage de scripts spécialement conçu pour le développement d'applications web

Prise en charge médicamenteuse (PECM) : processus complexe comprenant de nombreuses étapes et impliquant de nombreux acteurs

SQL (Structured Query Language) : langage informatique servant à exploiter des bases de données relationnelles

Système de gestion de base de données (SGBD) : logiciel système servant à stocker, manipuler et partager des données dans une base de données

Trello : outil de gestion de projet en ligne basé sur les méthodes agiles

UML : langage de modélisation graphique à base de pictogrammes conçu dans les domaines du développement logiciel et en conception orientée objet

RÉSUMÉ

Notre projet tutoré s'intégrait au sein du Centre Hospitalier Intercommunal de Castres Mazamet, dans lequel un outil basé sur la méthode BPRIM, une méthode de gestion intégrée des risques et des processus métiers, est utilisé afin de maîtriser les risques liés aux erreurs médicamenteuses au cours de la Prise En Charge Médicamenteuse (PECM) du patient hospitalisé. L'objectif principal de ce projet était de développer un outil informatique permettant de déclarer, recenser et analyser suivant des critères spécifiques les différentes erreurs médicamenteuses ayant lieu.

Ainsi, un travail d'analyse des besoins, de priorisation et de rédaction a été réalisé au premier semestre. Lors du second semestre, le développement d'une application web répondant aux différents besoins a été produit. Une documentation technique et une documentation utilisateur ont également été créées.

Mots clés :

Prise en charge médicamenteuse ; Erreur médicamenteuse ; Never event ; Déclaration ; Analyse ; CHIC de Castres Mazamet ; Application web ; PHP ; HTML ; Microsoft SQL Server ; SQL

SOMMAIRE

REMERCIEMENTS.....	2
GLOSSAIRE.....	3
RÉSUMÉ.....	4
INTRODUCTION.....	6
I. CONTEXTE DU PROJET.....	7
A. Présentation du projet.....	7
B. Présentation du CHIC de Castres-Mazamet.....	9
C. Enjeux du projet.....	10
II. MISE EN APPLICATION.....	11
A. Modèle de données.....	11
B. Structure de l'application.....	13
III. ETAPES DU PROJET.....	17
A. Gestion de projet.....	17
B. Problèmes rencontrés.....	18
C. Prolongements possibles.....	19
CONCLUSION.....	22
BIBLIOGRAPHIE.....	23
ANNEXES.....	24

INTRODUCTION

L'un des principaux défis des établissements de santé, aujourd'hui, est de maîtriser les risques liés aux erreurs médicamenteuses au cours de la Prise En Charge Médicamenteuse (PECM) du patient hospitalisé. La PECM est un des processus de soins parmi les plus complexes, transversal, pluridisciplinaire et qui implique de nombreux professionnels (médecins, infirmières, pharmaciens, préparateurs, aides-soignantes, etc.) sur plusieurs étapes.

Le CHIC de Castres-Mazamet a déjà développé en interne un outil basé sur la méthode BPRIM, une méthode de gestion intégrée des risques et des processus métiers. L'apprentissage mutuel au cours de cette expérimentation avec l'équipe du CHIC, a permis de tracer de nouvelles perspectives.

Ce projet consiste à implanter des techniques de cartographie des risques liés au processus de la PECM du patient hospitalisé au CHIC.

La cartographie des risques est un outil d'analyse et de communication graphique qui aide à recueillir les failles identifiées par les opérationnels, et évaluer avec eux la pertinence des actions à mettre en place pour atteindre les objectifs fixés. La matrice de criticité est une manière assez classique de synthétiser les connaissances sur ces risques. En complément de cela, ce projet consiste à proposer d'autres techniques pour construire un tableau de bord qui permettraient de faire un suivi de l'évolution dans le temps des risques.

Vous trouverez dans ce rapport une première partie présentant le projet et son contexte. Nous aborderons ensuite comment nous avons mis en application les différents objectifs fixés, en présentant le modèle de données et la structure de l'application. Enfin, nous ferons un point sur la gestion de projet, les problèmes que nous avons rencontrés et les prolongements possibles.

I. CONTEXTE DU PROJET

A. Présentation du projet

1. Les parties prenantes du projet

Maîtres d'ouvrage : Dr CLEOSTRATE, pharmacienne, et Dr CUFI, médecin au service gériatrie, docteurs au CHIC de Castres Mazamet.

Tuteurs école : Rafika THABET et Elyes LAMINE

Etudiants : Séraphie MAURY, Sarah GROS et Salomé CHEVAILLER

2. Les Erreurs Médicamenteuses (EM)

D'après le dictionnaire français de l'erreur médicamenteuse (cf. Annexe 7), rédigé sous l'égide de la Société Française de Pharmacie Clinique¹, l'erreur médicamenteuse représente l'écart par rapport à ce qui aurait dû être fait au cours de la prise en charge thérapeutique médicamenteuse du patient. L'erreur médicamenteuse est l'omission ou la réalisation non-intentionnelle d'un acte relatif à un médicament, qui peut être à l'origine d'un risque ou d'un événement indésirable pour le patient. Par définition, l'erreur médicamenteuse est évitable, car elle manifeste ce qui aurait dû être fait et qui ne l'a pas été au cours de la prise en charge thérapeutique médicamenteuse d'un patient.

L'erreur médicamenteuse peut concerner une ou plusieurs étapes du circuit du médicament, telles que : sélection au livret du médicament, prescription, dispensation, analyse des ordonnances, préparation galénique, stockage, délivrance, administration, information, suivi thérapeutique, mais aussi ses interfaces, telles que les transmissions ou les transcriptions.

3. Les besoins du CHIC de Castres-Mazamet

a. Les difficultés rencontrées

Les difficultés actuellement rencontrées concernant les risques liés aux EM au cours de la prise en charge médicamenteuse (PECM) du patient hospitalisé sont au niveau de la maîtrise et de la gestion de ces derniers.

Désormais, le CHIC expérimente avec un outil basé sur la méthode BPRIM, méthode de gestion des risques et des processus métiers et souhaite le développer avec de nouvelles fonctionnalités.

Actuellement, les erreurs médicamenteuses survenues au CHIC de Castres-Mazamet sont recensées sur des fiches manuscrites remplies par le personnel (cf. Annexe 6). Or, la méthode de déclaration n'est pas idéale puisqu'il faut se rendre à un endroit spécifique de l'hôpital pour effectuer la déclaration. Ainsi, certains ne le font pas ou oublient de le faire et ces EM demeurent non traitées. De plus, la description est faite sous forme de champ de texte libre ce qui ne favorise pas le traitement des EM.

La consultation des événements déclarés peut également s'avérer difficile. En effet, puisque la liste des EM est sur papier et que la description est un texte libre, il est impossible de les trier de manière efficace.

b. La réponse au besoin

Les événements recensés sont tous proposés pour analyse lors des comités de retour d'expérience (CREX). Un CREX est une démarche collective, organisée, rétrospective et systémique de recueil et d'analyse des défaillances produites lors de la prise en charge d'un patient. À chaque CREX, il n'y a qu'un seul événement choisi pour l'analyse. Il est alors très important de pouvoir classer les différents éléments selon divers critères afin de faciliter le choix des événements.

Les objectifs fixés par le CHIC sont les suivants :

- Enrichir l'outil dédié à la méthode BPRIM avec des techniques pour construire un tableau de bord.
- Faire évoluer l'outil par l'intégration de nouvelles connaissances issues de l'observation sur terrain.
- Tester et évaluer l'outil et l'améliorer avec les parties prenantes.

Dans le cadre de ce projet, il s'agit donc de développer un outil informatique avec une interface permettant de déclarer un événement avec davantage de précision, de consulter la liste des événements déclarés mais aussi d'avoir un tableau de bord qui recense et analyse les différentes erreurs médicamenteuses selon des critères spécifiques (période d'occurrence, médicament impliqué, récurrence...). De plus, ce service doit être accessible depuis tous les services de l'hôpital à tout moment.

B. Présentation du CHIC de Castres-Mazamet

1. Qu'est-ce qu'un CHIC ?

Un CHIC (Centre Hospitalier InterCommunal) est un centre hospitalier réparti sur plusieurs communes, résultant généralement de la fusion de plusieurs établissements de santé. Aujourd'hui, il existe 17 CHIC en France dont 2 en Occitanie: celui de Castelsarrasin-Moissac et celui de Castres-Mazamet avec lequel nous travaillons sur ce projet.

2. Le CHIC de Castres-Mazamet

Le Centre Hospitalier Intercommunal de Castres-Mazamet est un regroupement de 10 établissements répartis sur les communes de Castres et Mazamet :

- L'Hôpital du Pays d'Autan : Court Séjour (médecine, chirurgie et obstétrique) - L'Hôpital des Monges : Centre de Soins de Suite et de Réadaptation (SSR) et de Long Séjour
- La Villégiale Saint-Jacques : Résidence de retraite avec soins courants ou de cure médicale
- L'Institut de Formation en Soins Infirmiers (IFSI)
- Le Dispensaire, Centre de Vaccination International

- Le Centre de Planification et d'Education Familiale (CPEF)
- L'Hôpital Général de Mazamet : SSR et Unité Cognitivo-Comportementale (UCC)
- La Résidence du Midi : Résidence de retraite avec soins courants ou de cure médicale
- L'Unité de Soins de Longue Durée (USLD)
- La Blanchisserie

Pour ce projet, nous travaillons donc avec le Dr CLEOSTRATE et le Dr CUFI qui travaillent à l'Hôpital du Pays d'Autan à Castres.

C. Enjeux du projet

1. Enjeux sanitaire et organisationnel

Les erreurs médicamenteuses sont encore beaucoup trop présentes au sein de nos établissements de santé et restent malheureusement souvent non corrigées. En effet, la communication quant à la survenue des EM peut s'avérer difficile : difficulté de communication entre les services de l'hôpital, déclaration peu précise ou erronée d'une EM, etc. Les EM ne peuvent donc pas être traitées de manière optimale et les récurrences sont encore trop fréquentes.

Notre projet a pour objectif de répondre à cet enjeu sanitaire en informatisant les données de l'hôpital. Ceci permet un accès facilité aux données (peu importe le service dans lequel on se trouve), une déclaration plus rapide et plus précise des événements d'EM et l'analyse concrète des divers événements ainsi que leur évolution au cours du temps. Nous répondons donc à un enjeu à la fois sanitaire et organisationnel du CHIC de Castres-Mazamet.

2. Enjeu éthique

Au cours de ce projet, nous serons amenés à travailler dans un milieu où le secret professionnel est primordial et avec de nombreuses données confidentielles. Il s'agira alors de respecter cette confidentialité.

II. MISE EN APPLICATION

A. Modèle de données

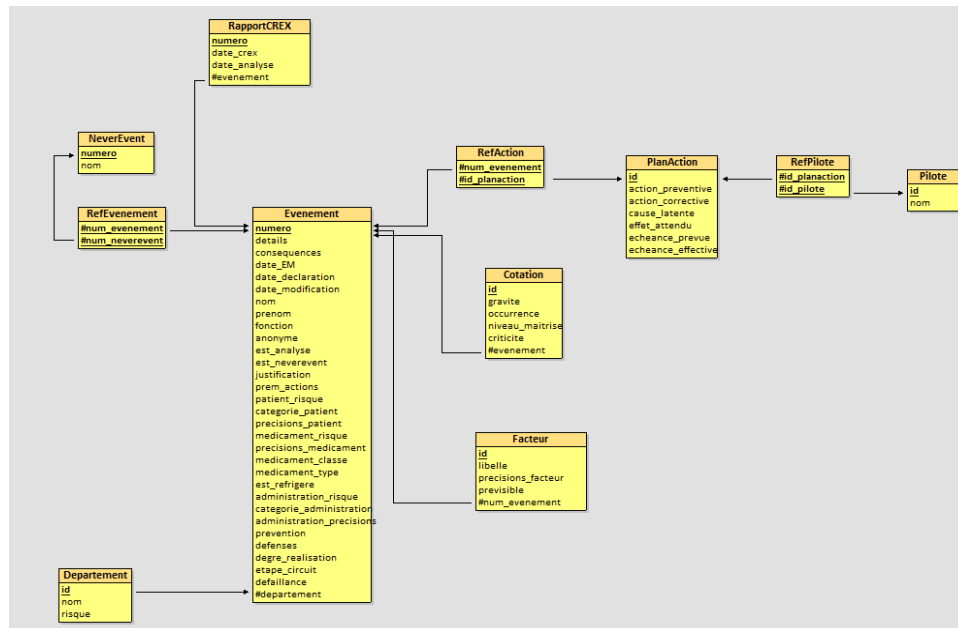


Fig.1. Modèle relationnel de données

Pour concevoir le modèle relationnel de données (cf. Annexe 2), nous sommes d'abord parties du diagramme UML fait au premier semestre (que vous pouvez voir ci-dessous). La plupart des classes devenaient des entités et celles-ci étaient liées entre elles à l'aide de clés étrangères selon l'association définie dans le diagramme UML (cf. Annexe 1).

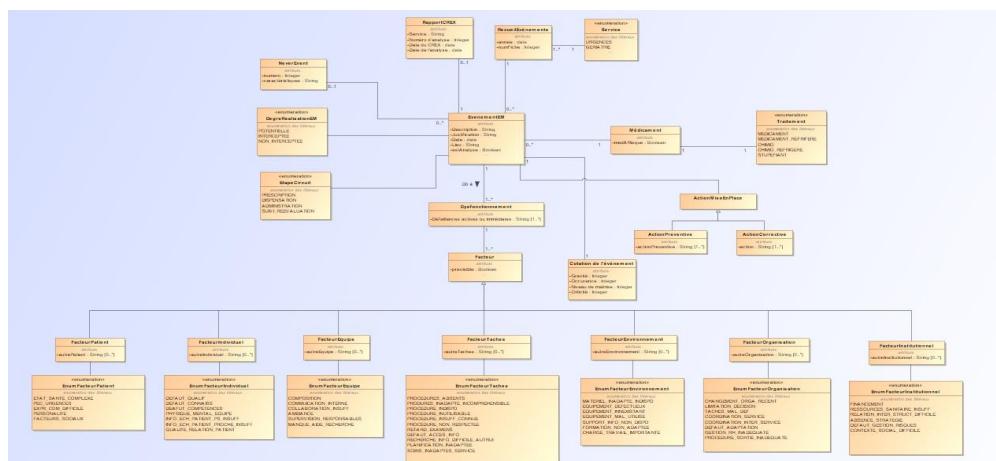


Fig.2. Diagramme UML

Cependant, lorsque nous avons commencé à programmer l'application, nous avons eu besoin de faire des ajustements et ce tout au long du projet. En effet, certaines entités n'étaient pas nécessaires et il était préférable de les intégrer directement dans l'entité "Evenement" pour simplifier la programmation.

C'est par exemple le cas des entités "Degre_realisation", "Etape_circuit", "Dysfonctionnement" et "Medicament". Nous avons simplement créé trois attributs "degre_realisation", "etape_circuit" et "defaillances" dans l'entité "Evenement" pour remplacer les entités du même nom. Quant à l'entité "Medicament", nous avons créé des attributs dans l'entité "Evenement" représentant ceux qui étaient dans l'entité "Medicament".

Nous avons également supprimé l'entité "Recueil_Evenements" qui était une collection d'événements, car en passant à des déclarations faites informatiquement, il n'était plus nécessaire d'avoir l'équivalent des fiches listant les événements qui ont eu lieu puisqu'une simple requête SQL suffit à les récupérer.

Concernant l'entité "Facteur", nous avons ajouté les attributs "libelle" et "precisions_facteur" pour que chaque facteur puisse être défini par son libellé et les précisions qui lui seront apportées. Nous avons bien évidemment laissé l'attribut "previsible" pour savoir si le facteur était prévisible ou non.

Il a aussi fallu créer trois tables de référence : "RefEvenement" reliant les entités "Evenement" et "NeverEvent", "RefAction" reliant les entités "Evenement" et "PlanAction", et "RefPilote" reliant les entités "PlanAction" et "Pilote". Ces tables de référence étaient nécessaires, car un événement peut avoir plusieurs never events, plusieurs plans d'action et un plan d'action peut lui-même avoir plusieurs pilotes. C'est aussi pour cette dernière raison que nous avons choisi de séparer le pilote de l'entité "PlanAction" et d'en faire une entité.

B. Structure de l'application

1. Choix des technologies

Au niveau du choix des technologies à utiliser pour le projet, nous avons une seule obligation : utiliser Microsoft SQL Server pour l'implémentation de la base de données puisque la plateforme ADOxx du CHIC de Castres-Mazamet utilise le SGBD (Système de Gestion de Base de Donnée) de Microsoft SQL Server.

Après réflexion, nous avons choisi de développer l'application en PHP (PHP Hypertext Preprocessor) qui est un langage de programmation spécialement conçu pour le développement d'applications web. Salomé et Séraphie ont eu l'opportunité de découvrir ce langage lors de leur stage de 3ème année et ont trouvé plus approprié d'utiliser ce langage plutôt que Java, que nous apprenons lors de notre formation à ISIS, pour sa facilité d'utilisation.

Ce langage peut également être facilement intégré au HTML (HyperText Markup Language). En effet, au milieu des balises HTML, il est possible d'intégrer des balises PHP exécutant une commande.

Un autre avantage est que le code est directement exécuté sur le serveur, générant ainsi le HTML, qui est ensuite envoyé au client. Le client ne reçoit que le résultat du script, sans aucun moyen d'avoir accès au code qui a produit ce résultat.

Pour communiquer avec la base de données, nous utilisons le SQL (Structured Query Language). C'est un langage informatique très utilisé pour communiquer avec les données d'un site Web.

2. Pages PHP

Notre application repose donc sur des pages PHP, composées de HTML pour définir leur structure.

Nous avons donc 8 pages, pour lesquelles nous allons vous expliquer le contenu :

- **accueil.php** : c'est la page d'accueil de l'application. Elle est composée de trois boutons permettant d'accéder à la déclaration d'une erreur médicamenteuse, à la consultation des erreurs déclarées ou à la consultation des erreurs analysées.
- **ajoutEM.php** : c'est la page de déclaration d'une erreur médicamenteuse. Le déclarant peut compléter tous les champs et valider sa déclaration, ajoutant l'erreur dans la base de données.
- **analyseEM.php** : c'est la page d'analyse d'une erreur médicamenteuse. Lors d'un CREX, les professionnels peuvent compléter les champs d'analyse et revenir sur les champs remplis lors de la déclaration.
- **bdd.php** : c'est la page permettant de connecter la base de données aux interfaces.
- **consultationEM.php** : c'est la page permettant de consulter les données d'une erreur médicamenteuse déclarée. On y retrouve tous les champs qui ont été complétés par le professionnel de santé déclarant.
- **consultationEManalyse.php** : c'est la page permettant de consulter les données d'une erreur médicamenteuse analysée. On y retrouve tous les champs qui ont été complétés lors de l'analyse de l'erreur.
- **listeEM.php** : c'est la page listant les erreurs médicamenteuses déclarées. Il y a la possibilité de filtrer les erreurs selon différents critères définis. On peut également accéder à la consultation ou à l'analyse de l'erreur choisie.
- **listeAnalyses.php** : c'est la page listant les erreurs médicamenteuses analysées. Il y a également la possibilité de filtrer les erreurs selon différents critères. On peut consulter l'erreur choisie.

Vous trouverez le visuel de ces différentes pages en annexes.

3. Connexion de la base de données aux interfaces

Il a fallu connecter nos différentes interfaces PHP avec notre base de données pour qu'elles puissent communiquer ensemble. Pour cela, nous avons utilisé les fonctions sqlsrv qui permettent de lier le PHP à une base de données Microsoft SQL Server.

Nous avons créé deux variables : `serverName` et `connectionInfo`. La première, `serverName`, précise sous forme de chaîne de caractères le nom du serveur que nous souhaitons utiliser. On peut retrouver celui-ci lorsque nous ouvrons l'application de Microsoft SQL Server, comme ci-dessous :

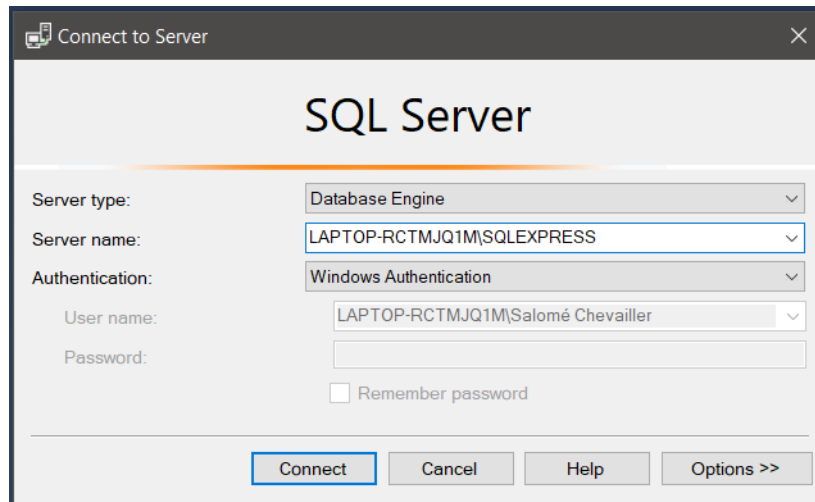


Fig.3. Interface de connexion de Microsoft SQL Server

La deuxième, `connectionInfo`, comprend les informations de connexion à la base de données sous forme de tableau. Nous avons donc spécifié le nom de la base de données. Notre base ne spécifiant pas d'identifiant utilisateur ni de mot de passe, nous n'avons pas eu besoin d'ajouter d'autres paramètres. La connexion utilisera donc l'authentification Windows.

Nous avons ensuite passé ces deux variables en paramètres de la fonction `sqlsrv_connect()` qui permet de connecter la base de données aux pages PHP. Cette fonction renvoie une valeur booléenne en fonction de sa réussite d'exécution.

Voici le code associé :

```
$serverName = "LAPTOP-RCTMJQ1M\SQLEXPRESS";  
// Since UID and PWD are not specified in the $connectionInfo array,  
// The connection will be attempted using Windows Authentication.  
$connectionInfo = array( "Database"=>"master");  
$conn = sqlsrv_connect( $serverName, $connectionInfo);
```

Fig.4. Code pour la connexion à la base de données

Pour tester que la connexion fonctionne bien, sans avoir à faire des requêtes SQL pour manipuler des données, nous avons utilisé le code suivant, qui affiche "Connection established" si la connexion a bien été établie ou "Connection could not be established" suivie des erreurs de connexion le cas échéant.

```
if( $conn ) {  
    echo "Connection established.<br />";  
}else{  
    echo "Connection could not be established.<br />";  
    die( print_r( sqlsrv_errors(), true));  
}
```

Fig.5. Code testant si la connexion a été établie

III. ETAPES DU PROJET

A. Gestion de projet

En ce qui concerne la gestion de projet, nous avons établi un diagramme de Gantt (cf. Annexe 3) afin de visualiser dans le temps les différentes tâches que nous avons à faire tout au long du projet.

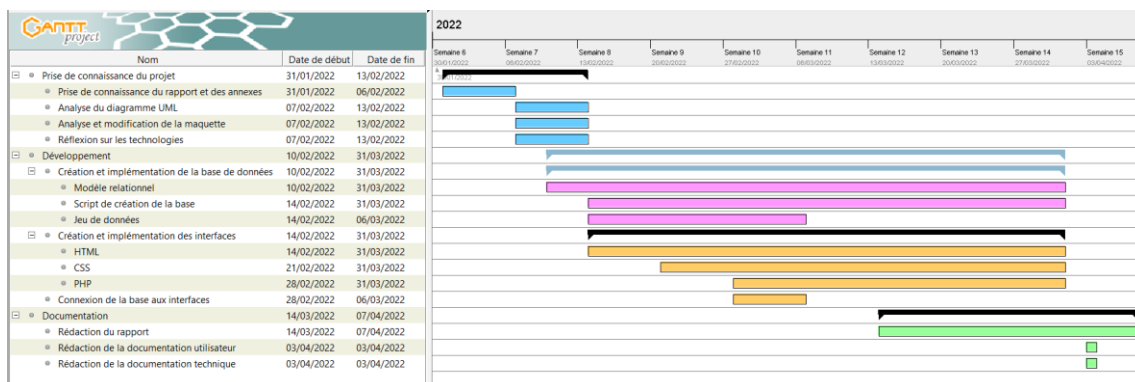


Fig.6. Diagramme de Gantt

Nous avons également utilisé Trello, un outil de gestion de projet en ligne inspiré des méthodes Agiles, qui permet de répartir les différentes tâches dans des colonnes. Nous avons donc réparti nos tâches dans trois colonnes : “À faire”, “En cours” et “Fait”, afin de gérer visuellement et collectivement l’état d’avancement du travail.

De plus, nous avons des réunions régulières avec notre tutrice (toutes les semaines), Rafika THABET, afin de faire le point sur l’avancement du projet et de poser nos éventuelles questions. Nous avons eu deux réunions (le 11/03 et le 25/03) avec nos maîtres d’ouvrage, Dr CLEOSTRATE et Dr CUFI, pour leur présenter l’application et prendre en compte leurs différentes remarques afin d’apporter des améliorations.

Concernant la répartition des tâches, Salomé s’est occupée du back-end, du front-end et des rédactions de documents (rapport, documentations utilisateur et technique). Séraphie s’est chargée du front-end ainsi que de la rédaction du rapport. Et enfin, Sarah s’est occupée du front-end.

B. Problèmes rencontrés

1. Problèmes humains

Parmi les problèmes rencontrés, le principal fut le manque de temps. En effet, le nombre d'heures prévues dans l'emploi du temps ne suffisait pas à mener à bien ce projet. Il a donc fallu trouver du temps en dehors de ces créneaux, ce qui fut compliqué étant donné la charge de travail que nous avions dans les autres matières.

De plus, le fait que le groupe ait quasiment totalement changé entre le premier semestre et le deuxième semestre fut compliqué à gérer. Salomé n'étant rentrée de son semestre à Chypre que début février, nous avons donc pris du retard sur le commencement du projet pour ce deuxième semestre. Il a également fallu passer un long temps à prendre connaissance du travail qui avait été fait au semestre précédent, du sujet du projet et du travail devant être effectué.

Nous souhaitions revenir sur le fait que le sujet n'était pas assez défini. Les besoins n'étaient pas assez explicités ni priorisés. Nous avons eu des indications générales sur le travail à effectuer mais de nombreuses informations étaient manquantes. Cela est aussi dû au fait que nous avons rencontré les maîtres d'ouvrage que le 11 mars. Avant cette date, nous avons dû avancer en ayant connaissance des grandes lignes, avec ce que nous avons compris et avec ce que nous estimions être pertinent. De ce fait, lorsque nous avons eu ce premier rendez-vous, beaucoup de nouvelles fonctionnalités ont été ajoutées et ce fut difficile par la suite de pouvoir tout prendre en compte dans les délais restants.

2. Problèmes logiciels

Comme mentionné précédemment, nous avons dû relier notre base de données à nos pages PHP pour qu'elles puissent communiquer ensemble. PHP fonctionne généralement avec MySQL, un autre système de gestion de base de données. Il a donc fallu trouver une solution pour relier nos pages à Microsoft SQL Server. Après plusieurs recherches sur le sujet, nous avons dû installer un driver Microsoft SQL Server à notre serveur web. Ce driver permet la lecture et l'écriture de données SQL Server à partir de scripts PHP. De ce fait, nous avons pu manipuler les données comme nous le souhaitions. Néanmoins, il a également fallu se familiariser avec les différentes méthodes sqlsrv liées à ce driver, ce qui a pris un peu de temps.

Toujours concernant Microsoft SQL Server, il n'est téléchargeable que sur les OS Windows et Linux. De ce fait, pour Mac OS ce n'est pas directement téléchargeable, il faut utiliser SQL Server on Linux Docker Image. Il faut donc télécharger Docker pour Mac OS, puis télécharger SQL Server on Linux Docker image. Il faut ensuite installer homebrew pour installer ODBC Driver et SQL Command Line Utility pour SQL Server.

C. Prolongements possibles

Malheureusement, nous n'avons pas eu le temps de finaliser notre application. De nombreuses fonctionnalités sont manquantes et certains points abordés avec les maîtres d'ouvrage n'ont pas pu être ajoutés.

Nous allons donc d'abord aborder les fonctionnalités manquantes et par la suite énoncer les nouvelles requêtes du CHIC.

1. Fonctionnalités manquantes

Premièrement, sur les pages listant les erreurs médicamenteuses déclarées et celles analysées (listeEM.php et listeAnalyses.php), nous avons créé des filtres pour que les utilisateurs puissent choisir les erreurs médicamenteuses qu'ils souhaitent afficher. Cependant, ces filtres ne sont actuellement pas fonctionnels. Il faudra donc les faire fonctionner, en prenant en compte le fait de pouvoir les combiner les uns avec les autres.

Ensuite, lors de la déclaration et de l'analyse, il y a la possibilité de spécifier les never-events concernés et les différents facteurs liés à l'événement. Les données sont alors bien enregistrées dans la base de données, mais ne sont jamais récupérées. Il faudrait donc faire les requêtes SQL nécessaires et afficher les données récupérées aux endroits dédiés.

Concernant la phase d'analyse, il est pour l'instant possible de renseigner les champs sauf pour les deux dernières étapes (étapes 4 et 5). Il sera donc nécessaire de programmer l'enregistrement dans la base des différentes données complétées lors de ces étapes.

Un point sur lequel il faudra absolument travailler concerne le design de l'application. En effet, il faudrait séparer les différents champs pour que ce soit plus aéré et lisible, penser également à ajouter de la couleur pour que ce soit plus plaisant à l'utilisation.

Pour faciliter l'utilisation, il faudrait également ajouter du JavaScript, un langage de programmation utilisé dans les pages web interactives. Effectivement, cela pourrait être intéressant de faire apparaître certains champs seulement lorsque nous en avons besoin. Par exemple, si la personne déclarant une erreur médicamenteuse a coché qu'elle voulait déclarer de manière anonyme, alors les champs pour entrer le nom, le prénom et la fonction ne devraient pas être visibles. Et inversement, si elle a coché qu'elle ne voulait pas rester anonyme, les champs devraient apparaître.

Une des requêtes faites était de pouvoir afficher le nombre d'erreurs médicamenteuses suivant certains critères. Pour cela, il serait envisageable de créer une nouvelle interface dans laquelle on afficherait les données sous forme de diagrammes et de courbes pour que ce soit plus représentatif.

2. Nouvelles requêtes du CHIC

Nous allons maintenant aborder les nouvelles demandes de nos maîtres d'ouvrage.

Tout d'abord, le CHIC souhaiterait que lorsque nous consultons une erreur médicamenteuse déclarée, il y ait trois boutons. Le premier bouton, appelé "Valider l'événement tel qu'il a été déclaré" permettrait de valider les informations qui ont été entrées lors de la déclaration. Il faudrait qu'en validant, il y ait la possibilité d'ajouter le nom, le prénom et la fonction du professionnel de santé ayant validé cette déclaration. Le deuxième bouton, appelé "Modifier l'événement" permettrait de débloquer la modification des différents champs de l'erreur médicamenteuse, pour correction ou changement. En validant les modifications, il faudrait également qu'on puisse préciser le nom, le prénom et la fonction du professionnel de santé ayant modifié les données initiales. Le troisième bouton, appelé "Mettre l'événement en attente" permettrait de déplacer l'événement dans une liste d'événements mis en attente parce qu'il ne serait pas exploitable tel qu'il a été déclaré et que la modification de ses données ne serait pas non plus possible.

Il faudrait que pour chacune de ces trois actions, on garde une trace de l'action qui a été faite. Par exemple, si l'événement a été validé tel qu'il a été déclaré, que ce soit écrit quelque part lors de sa consultation ou quand il est affiché dans la liste des erreurs déclarées.

Un point découlant directement de cette nouvelle requête est qu'il faut absolument pouvoir garder une trace de l'événement tel qu'il a été déclaré au départ. Juridiquement, il est obligatoire de garder la déclaration initiale. Il serait peut-être possible de placer ces déclarations dans une archive.

Une autre demande qui nous a été formulée est le fait de pouvoir extraire les lignes concernant les actions mises en place suite à l'erreur médicamenteuse dans un tableur Excel. En effet, ces données sont importantes et utilisées par un autre service du CHIC. Cela leur ferait donc gagner du temps.

Concernant les actions qui ont été menées, il serait intéressant de pouvoir les consulter sans avoir accès à toutes les erreurs médicamenteuses. Ainsi, il faudrait créer une nouvelle interface accessible depuis l'accueil par un bouton "Je consulte les actions mises en place". Dans cette interface, on retrouverait une liste des actions en fonction de la date du CREX ainsi que les résultats d'analyse. Il faudrait également pouvoir filtrer ces actions en fonction du service et du pilote.

CONCLUSION

Pour conclure, nous pensons avoir en majeure partie répondu aux besoins initiaux du CHIC de Castres Mazamet. En effet, il était primordial de passer d'un processus papier à un processus entièrement informatisé pour faciliter la déclaration, l'analyse et la prise en main des erreurs médicamenteuses. Pour cela, nous avons produit une application web proposant une interface qui permet de déclarer une erreur médicamenteuse avec des critères supplémentaires, en comparaison avec la fiche manuscrite de déclaration, pour que cette dernière soit bien plus précise et affinée. Cette application permet également de consulter la liste des événements déclarés, de passer à leur analyse et d'avoir un récapitulatif de celle-ci. Néanmoins, tous les objectifs n'ont pas pu être traités dans les délais mais ont été abordés à plusieurs reprises avec les professionnels pour qu'ils puissent être envisagés par la suite.

Nous espérons donc que le travail que nous avons fourni satisfera nos maîtres d'ouvrage et qu'il permettra de faciliter leur travail au quotidien. Nous espérons également que ce projet puisse être poursuivi par d'autres étudiants pour que toutes les demandes du CHIC soient traitées et qu'une version complète de l'outil puisse voir le jour.

BIBLIOGRAPHIE

1. SFPC. Dictionnaire français de l'erreur médicamenteuse. http://optimiz-sih-circ-med.fr/Documents/Dictionnaire_SFPC_EM.pdf
2. Microsoft SQL Server. <https://www.microsoft.com/en-us/sql-server>
3. PHP. <https://www.php.net/manual/fr/intro-what-is.php>
4. Java. https://www.java.com/fr/download/help/what-is_java.html
5. HTML. <https://developer.mozilla.org/en-US/docs/Web/HTML>
6. SQL. <https://www.w3schools.com/sql/>
7. MySQL. <https://www.mysql.com/>
8. JavaScript. <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
9. Never event. <https://www.omedit-paysdelaloire.fr/qualite-securite-et-vigilances/never-events/>

ANNEXES

Annexe 1 : Diagramme UML

Annexe 2 : Modèle relationnel de données

Annexe 3 : Diagramme de Gantt

Annexe 4 : Documentation utilisateur

Annexe 5 : Documentation technique

Annexe 6 : Dossier de fiches CREX

Annexe 7 : Dictionnaire SFPC