



PPE GSB LOT

Liste des compétences du PPE

- A1.1.1 Analyse du cahier des charges d'un service à produire
 - C1.1.1.2 Identifier les fonctionnalités attendues du service à produire
- A1.4.1 Participation à un projet
 - C1.4.1.2 Rendre compte de son activité
- A1.4.2 Évaluation des indicateurs de suivi d'un projet et justification des écarts
 - C1.4.2.1 Suivre l'exécution du projet
 - C1.4.2.2 Analyser les écarts entre temps prévu et temps consommé
- A4.1.1 Proposition d'une solution applicative
 - C4.1.1.1 Identifier les composants logiciels nécessaires à la conception de la solution
- A4.1.2 Conception ou adaptation de l'interface utilisateur d'une solution applicative
 - C4.1.2.1 Définir les spécifications de l'interface utilisateur de la solution applicative
- A4.1.3 Conception ou adaptation d'une base de données
 - C4.1.3.2 Implémenter le schéma de données dans un SGBD
 - C4.1.3.4 Manipuler les données liées à la solution applicative à travers un langage de requête
- A4.1.6 Gestion d'environnements de développement et de test
 - C4.1.6.1 Mettre en place et exploiter un environnement de développement
- A4.1.7 Développement, utilisation ou adaptation de composants logiciels
 - C4.1.7.1 Développer les éléments d'une solution
 - C4.1.7.4 Utiliser des composants d'accès aux données

A1.1.1 Analyse du cahier des charges d'un service à produire

C1.1.1.2 Identifier les fonctionnalités attendues du service à produire

Définition du besoin

Le besoin est d'enregistrer et pouvoir consulter les lots et échantillons de médicaments depuis leur fabrication jusqu'à ce qu'ils soient laissés à un médecin.

Les informations à enregistrer sont :

- N° du lot
- N° d'échantillon : donnée numérotée de 1 à n pour un lot donné
- Médicament associé au lot (un seul médicament pour un lot)
- Date de fabrication du lot
- Date de sortie du stock d'un échantillon
- Visiteur médical à qui a été remis l'échantillon
- Date de visite chez le médecin à qui l'échantillon a été laissé
- Médecin à qui l'échantillon a été laissé

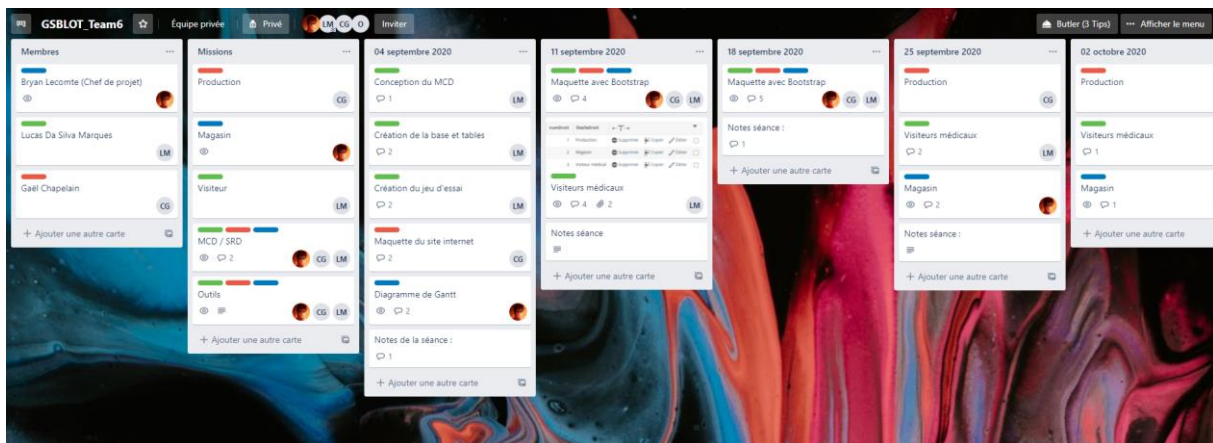
L'application est destinée à trois types d'utilisateur :

- La production :
 - o Elle enregistre les lots fabriqués
 - o Elle consulte les lots par médicament, par date de fabrication
- Le magasin :
 - o Il renseigne date et visiteur après attribution des échantillons
 - o Il consulte les échantillons
 - En stock
 - Sortis : par médicament, par date de sortie, par visiteur
- Le visiteur médical :
 - o Il renseigne date et médecin à qui l'échantillon a été laissé
 - o Il consulte les échantillons
 - par médicament, par date de visite, par médecin

Ceci se sont les fonctionnalités attendues de notre application. Tant en base de données que dans l'application elle-même en programmation.

A1.4.1 Participation à un projet

C1.4.1.2 Rendre compte de son activité



Pour une bonne organisation et pour savoir qui en est où, qui fait quoi et qui a des erreurs. Un Trello était obligatoire pour un bon suivi du projet. En cas de besoin, je venais fournir mon assistance.

A1.4.2 Évaluation des indicateurs de suivi d'un projet et justification des écarts

C1.4.2.1 Suivre l'exécution du projet

C1.4.2.2 Analyser les écarts entre temps prévu et temps consommé

Dans l'exécution du projet, il y a eu de nombreuses absences d'un de nos développeurs, ce qui a mis en retard l'ensemble de sa partie en développement, dans la globalité le projet à bien avancé malgré certains petits problèmes avec le second développeur qui contestait ce que son chef de projet lui demandait de faire.

Dans les écarts, suite à l'absence d'un de nos développeurs, le suivi de sa partie sur le projet était très difficile, il a dû faire un rattrapage de l'ensemble des heures qu'il n'a pas pu faire afin de ne plus être en retard malgré l'écart de temps par rapport au second développeur et du chef de projet.

A4.1.1 Proposition d'une solution applicative

C4.1.1.1 Identifier les composants logiciels nécessaires à la conception de la solution

Pour la mise en place des composants de l'application. Nous utilisons deux frameworks, l'un s'est Bootstrap pour l'interface utilisateur et l'autre c'est Smarty pour la séparation du PHP et du HTML.

A4.1.2 Conception ou adaptation de l'interface utilisateur d'une solution applicative

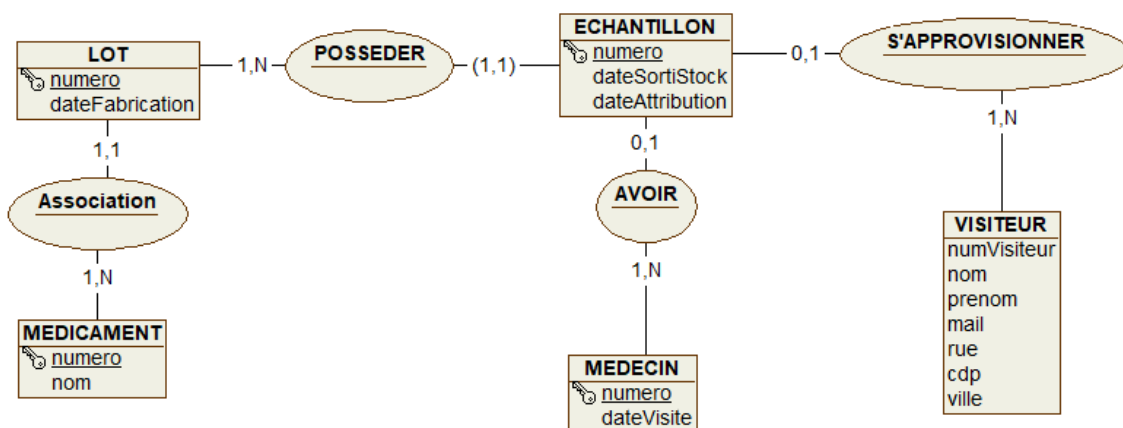
C4.1.2.1 Définir les spécifications de l'interface utilisateur de la solution applicative

L'interface utilisateur a été écrite à l'aide de Bootstrap avec quelques petits ajouts de CSS pour faire une personnalisation et avoir une application à la fois jolie sur un PC, mais aussi responsive donc accessible sans difficulté sur tablette et mobile.

A4.1.3 Conception ou adaptation d'une base de données

C4.1.3.2 Implémenter le schéma de données dans un SGBD

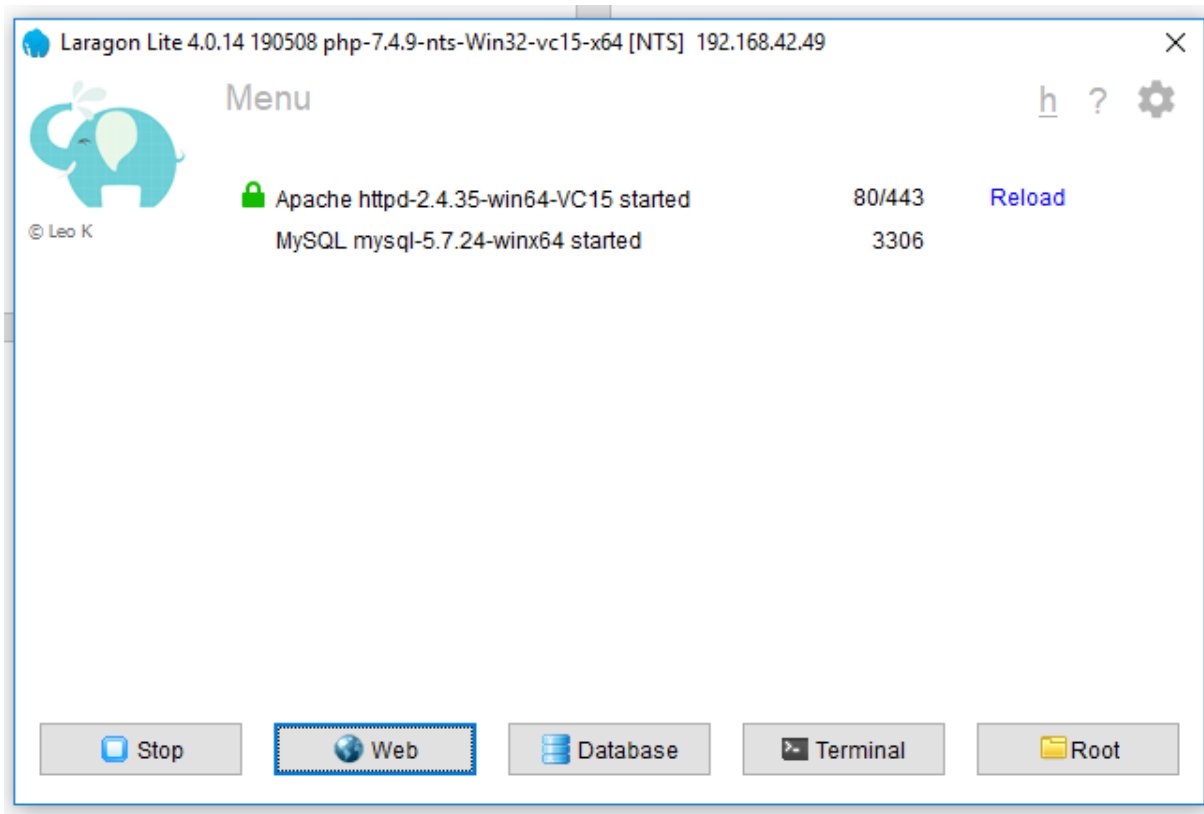
C4.1.3.4 Manipuler les données liées à la solution applicative à travers un langage de requête



Notre langage de requête dans notre application c'est du SQL et gérer par phpMyAdmin.

A4.1.6 Gestion d'environnements de développement et de test

C4.1.6.1 Mettre en place et exploiter un environnement de développement



Utilisation de Laragon qui est une application légère et portable que j'utilise sur mon disque dur externe, très pratique si je souhaite développer sur mon PC portable dans l'établissement et chez moi sur ma Surface Book 2. Laragon propose les dernières versions d'Apache, de MySQL, de phpMyAdmin et j'utilise le SSL pour une connexion cryptée.

A4.1.7 Développement, utilisation ou adaptation de composants logiciels

C4.1.7.1 Développer les éléments d'une solution

C4.1.7.4 Utiliser des composants d'accès aux données

```
<?php
// On essaye d'établir une connexion
// La connexion va être connectée à une base de données
try
{
    // On déclare une classe PDO qui permet la connexion
    // Cette classe demande l'hôte, la base de données, le nom d'utilisateur et le mot de passe
    // Ainsi dans l'exécution de la demande de connexion, on encode la connexion en UTF8
    $db = new PDO('mysql:host=localhost;dbname=bts_ppe_gsbplot', 'root', '');
    $db->exec("SET CHARACTER SET utf8");
}

// Si la connexion n'est pas établie
// Alors, notre catch nous donne l'erreur en $e
// Ainsi celui-ci affiche l'erreur et son numéro d'erreur
catch(Exception $e)
{
    echo 'Erreur : '.$e->getMessage().'\n';
    echo 'No : '.$e->getCode();
}
?>
```