



REPOBLIKAN’I MADAGASIKARA

**Fitiavana-Tanindrazana-Fandrosoana**

-----------------------------------------

**MINISTERE DE L’ENSEIGNEMENT SUPERIEUR ET DE LA**

**RECHERCHE SCIENTIFIQUE**

--------------------------------

**UNIVERSITE DE MAHAJANGA**

------------------------

**Institut Supérieur des Sciences et Technologies de Mahajanga**

**(ISSTM)**

---------------------

**Mention: Sciences et Technique du Numérique et Physiques Appliquées**

**Parcours: Génie Informatique**

**-----------------------------**

Mémoire de fin d’étude en vue de l’obtention du Diplôme de

Licence Professionnelle

----------------

**THEME: CREATION D’UNE APPLICATION DE GESTION DE DISPENSAIRE**

Présenté par:

**Mr RAKOTONIAINA Tefy Nambinintsoa**

Président du Jury :

Examinateur :

Encadreur Pédagogique :

Encadreur Professionnelle  :

ANNEE UNIVERSITAIRE : 2020 - 2021

# REMERCIEMENTS

Nous remercions, tout d’abord Dieu tout puissant et miséricordieux de nous avoir donné de la chance et le privilège de suivre le chemin de connaissance et du savoir.

Ensuite, nos remerciements, les plus sincères vont à toute personne ayant eu la bonté et la patience de satisfaire notre curiosité et de nous aider dans notre travail par leurs précieux conseils, réponses et recommandations.

En particulier ;

* Monsieur **TIANDAZA Dinaraly Odilon**, Professeur titulaire, Président de l’université de Mahajanga pour sa collaboration et son soutien tout au long de nos formations ;
* Monsieur **MAXWELL Djaffard**, Professeur, Directeur de l’Institut Supérieur des Sciences et Technologies de Mahajanga (ISSTM), qui n’a pas ménagé son temps et ses efforts pour la bonne marche de notre formation au niveau de l’institut ;
* Monsieur **TSANGANDRAZANA Judicaël**, Chef de parcours de génie Informatique et notre encadreur pédagogique, qui malgré ses lourdes responsabilités, n’a jamais cessé de nous prodiguer des conseils visant à nous garantir une carrière professionnelle honorable.
* Nous tenons à remercier toute l’équipe pédagogique de l’Institut Supérieur des Sciences et Technologies de Mahajanga en particulier le corps professoral intervenant dans toute les mentions pour l’effort fournit à la réussite de notre formation et mieux atteindre tous les objectifs attendus des différentes disciplines.

Je tiens à remercier mes très chers parents qui ont toujours été là pour moi, vous avez tout sacrifié pour vos enfants n’épargnant ni santé ni efforts. Vous m’avez donné un magnifique modèle de labeur et de persévérance. Je suis redevable d’une éducation dont je suis fière.

Sans oublier d’exprimer notre reconnaissance envers les amis et collègues qui m’ont apporté leur soutien moral et intellectuel tout au long de ma démarche.

À toutes et à tous, soyez assurés, de mes respects et de mes gratitudes.

# Présentation de L’ISSTM

**Historique**

Depuis la création du Centre Universitaire Régional de Mahajanga en 1977 et aussi après la création des six Universités de Madagascar en 1988, l’évolution de l’offre de formation a été relativement faible. Depuis l’année universitaire 2010/2011, un bon nombre d’offres de formations a été créé à l’Université de Mahajanga. L’institut Supérieur de Sciences et Technologies de Mahajanga en abrégé par ISSTM est l’un parmi des nouveaux centres de formation de techniciens supérieurs de haut niveau au sein de cette Université. Cet institut est créé au cours de l’année universitaire 2011/2012.

Les exigences du développement économique régional, les insuffisances des cadres techniciens intermédiaires dans les entreprises, les collectivités décentralisées et dans le but de réduire les dépenses des parents qui sont obligés d’envoyer leurs enfants hors de la province de Mahajanga, sont les principales raisons de la mise en place de la licence professionnelle en Sciences et Technologies en Génies Civil et Industriel au sein de l’ISSTM. Les étudiants sortant de l’ISSTM sont opérationnels dans les domaines du génie thermique, du génie industriel, du génie électrique, du génie civil et du génie informatique après avoir fini leur cursus.

L’institut est devenu un établissement à part entière comme l’IOSTM, la Faculté de Médecine et la FSTE, selon le décret N°2017-418 du 06 Juin 2017.

Les contenus de ces formations scientifiques sont accompagnés d’une ouverture au sein des milieux professionnels garantie dans les visites d’imprégnations, les stages en entreprises et les interventions de professionnels dans les domaines spécialisés du programme.

La formation est payante, est suit le système LMD et délivre à la fin du cursus un diplôme de licence professionnelle dans les mentions suivantes :

* Sciences et Technologies Industrielles (STI) composée de trois parcours :
* Génie Thermique (*option froid et énergie*)
* Génie Industriel (*maintenance et électromécanique*)
* Génie Electrique
* Génie Chimique (*chimie industrielle et extraction des huiles essentielles*)
* Sciences et Techniques du Génie Civil (STGC), parcours :
* Bâtiment et Travaux Publics
* Hydraulique
* Architecture
* Sciences et Techniques du Numérique et Physiques Appliquées (STNPA)
* Génie Informatique
* Génie Electronique informatique

**L’organigramme des personnels au sein de L'ISSTM**

La figure mentionnée dans la page suivante illustre l’organigramme du fonctionnement de l’ISSTM.

*Figure 1 : Organigramme de l’ISSTM*

# LISTE DES ABRÉVIATIONS

IDE: Integrated Development Environment

VsCode: Visual Studio Code

PHP: Hypertext Preprocessor

HTML: HyperText Markup Language

PSR: PHP Standards Recommendations

FIG: Framework Interop Group

MVC: Modèle-vue-contrôleur

XSS: cross-site scripting

CSRF: cross-site request forgery

SQL: Structured Query Language

CLI: Command Line Interface

HTTPS: HyperText Transfer Protocol Secure

DAO: Decentralized Autonomous Organization

CSS: Cascading Style Sheet

Ajax: Asynchronous JavaScript And XML

MCD: Modèle Conceptuel des Données

MLD: Modèles logiques de données

SGBD: Système de Gestion de Base de Données

Mysql: My Structured Query Language

XAMPP: **X** (cross) **A**pache **M**ariaDB **P**erl **P**HP

# LISTE DES FIGURES

Figure 1 : Organigramme de l’ISSTM

**Figure 2 : Onglet d’extension**

**Figure 3 : Extension dans Visual Studio Code**

**Figure 4 : Changer en mode prod**

**Figure 5 : extrait du code de la requête**

**Figure 6 : Déroulement d’un site en symfony**

**Figure 7 : Modèle conceptuel des Données**

**Figure 8 : Modèle Logique des Données**

**Figure 9 : Mode de traitement des données**

***Figure 10 :* Différents systèmes de gestion de base de données**

**Figure 11 : Fenêtre principale**

**Figure 12 : Fenêtre d’Authentification**

**Figure 13 : Extrait du code pour créer l’Authentification**

**Figure 14 : Extrait du code de redirection d’Authentification**

**Figure 15 : Ajout d’un patient**

**Figure 16 : Prise de charge**

**Figure 17 : Liste des patients**

**Figure 18 : Page d’accueil du docteur**

**Figure 19 : Page du diagnostique**

**Figure 20 : Ajout de la maladie**

**Figure 21 : Ajout du médicament**

**Figure 21 : Ajout du médicament**

**Figure 23 : Accueil d’administration et ajout d’un personnel**

**Figure 24 : Liste des personnels**

# LISTE DES TABLEAUX

Tableau des commandes pour l’installation

**Table des matières**

[REMERCIEMENTS i](#_Toc75188342)

[Présentation de L’ISSTM ii](#_Toc75188343)

[LISTE DES ABRÉVIATIONS iv](#_Toc75188344)

[LISTE DES FIGURES v](#_Toc75188345)

[LISTE DES TABLEAUX vi](#_Toc75188346)

[A- GENERALITE 1](file:///C:\Users\TEFY\Desktop\Mémoire%20de%20fin%20d'étude.docx#_Toc75188347)

[Introduction générale 2](#_Toc75188348)

[B- MATERIELS ET METHODES 4](file:///C:\Users\TEFY\Desktop\Mémoire%20de%20fin%20d'étude.docx#_Toc75188349)

[I. Matériels 4](#_Toc75188350)

[1. Environnement de développement 5](#_Toc75188351)

[a. Les outils nécessaires dans un IDE: 5](#_Toc75188352)

[b. Le Visual Studio Code: 6](#_Toc75188353)

[2. Utilisation du FRAMEWORK PHP 8](#_Toc75188354)

[a. C’est quoi un FRAMEWORK? 8](#_Toc75188355)

[b. Le Framework SYMFONY 10](#_Toc75188356)

[3. Utilisation d’une bibliothèque BOOTSRAP [XI] 15](#_Toc75188357)

[ Bootstrap.css 15](#_Toc75188358)

[ Bootstrap.js 16](#_Toc75188359)

[ Glyphicons 16](#_Toc75188360)

[II. Méthodes 17](#_Toc75188361)

[1. Merise [XII] 17](#_Toc75188362)

[ Méthode d'analyse et de conception 17](#_Toc75188363)

[ Niveau conceptuel 17](#_Toc75188364)

[2. Base de données [XIV] 20](#_Toc75188365)

[Système de Gestion de Base de Données: 20](#_Toc75188366)

[Les différents systèmes de gestion de base de données 21](#_Toc75188367)

[C- REALISATION ET RESULTAT 23](file:///C:\Users\TEFY\Desktop\Mémoire%20de%20fin%20d'étude.docx#_Toc75188368)

[I. Réalisation 24](#_Toc75188369)

[D- Discussion 33](file:///C:\Users\TEFY\Desktop\Mémoire%20de%20fin%20d'étude.docx#_Toc75188370)

[1. Difficultés rencontrés 34](#_Toc75188371)

[CONCLUSION 35](#_Toc75188372)

# A- GENERALITE

Introduction générale:

La propagation des maladies ingérable et contagieuse oblige beaucoup de personne à consulter les hôpitaux ou les cabinets médicaux. Grâce à ce situation, ces établissements de santé sont débordées de patient en consultation ce qui augmente le taux de contamination en vue des regroupements. Au niveau technique, l’enregistrement de ses patients s’annonce difficile sur le fait qu’il y manque de technologie approprié.

L’usage d’application web est l’un des moyens envisageable pour combler ses manques. Une **application web** désigne un logiciel applicatif hébergé sur un serveur et accessible via un navigateur web. Contrairement à un logiciel traditionnel, l’utilisateur d’une application web n’a pas besoin de l’installer sur son ordinateur. Il lui suffit de se connecter à l’application à l’aide de son navigateur favori. La tendance actuelle est d’offrir une expérience utilisateur et des fonctionnalités équivalentes aux logiciels directement installés sur les ordinateurs [I]. C’est de là que s’inspire le développement de ce thème intitulé « CREATION D’UNE APPLICATION DE GESTION DE DISPENSAIRE » durant le stage de la fin d’étude en Licence qu’on a effectué à la société KAROKA. Une application qui permet d’avoir un moyen rapide de consulter les patients, de mieux organiser le travail et d’enregistrer rapidement les activités.

**PROBLEMATIQUE:**

Au sein de la société KAROKA, la conquête de nouvelles activités bénéficières est primordiale. Chaque année, le chef de service de KAROKA recrute des personnes possédant des talents pouvant bénéficier la société; le recrutement des développeurs fait partie de ces critères.

A part la conception des projets interne, ils proposent aussi des idées pouvant développer l’informatisation des autres établissements. Le développement du projet de création d’une application de gestion de dispensaire fait partis de ces idées. Dans le développement de ce projet, la problématique de l’étude s’annonce comme suit: face au plusieurs enregistrement des patients au sein des établissements de santé, comment enregistrer rapidement les données, avoir une bonne organisation et éviter les pertes de données qui pourraient amener à la mal traitement des patients au sein des dispensaires?

**Objectifs spécifique:**

Pour pallier à ces problèmes, nous allons implanter un système informatique qui va assurer la gestion de toutes les branches existantes dans les cabinets médicales. Ainsi, on va introduire un système qui permet de traiter automatiquement les données. Un système qui permet de gérer les diagnostics des patients, de gérer leur maladies y compris leur médicaments.

Au finale, ce système informatique a pour rôle de gérer la plus part des fonctions existant au sein des établissements médicales y compris les traitements et recherches des fichiers des patients.

Ces solutions seront apportées par notre application et tout en assurant la sécurité et la fiabilité des documents et des informations stockées dans les cabinets.

# B- MATERIELS ET METHODES

I. Matériels

1. Environnement de développement

**Connue sous l’abréviation IDE**, c’est un logiciel qui rassemble des outils permettant de développer d’autres logiciels tels que des applications mobiles, des logiciels pour ordinateur ou consoles de jeux, des sites web… [II].

En programmation informatique, il fournit un ensemble d'outils qui permet d'augmenter la productivité des programmeurs qui développent des logiciels. Il comporte un éditeur de texte, un simulateur, un compilateur ainsi qu'un débogueur. [III].

### a. Les outils nécessaires dans un IDE:

Comme nous avons ci-dessus, un bon environnement de développement doivent avoir ces quatre fonctions: un éditeur de texte, un simulateur, un compilateur ainsi qu’un débogueur.

* Editeur de texte: Elle est destinée à la programmation, en outre, à insérée notre code au moment du développement. Son éditeur de texte peut avoir une fonctionnalité de code intelligent, c’est-à-dire qu’il change automatiquement la coloration, génère une mise en forme et s’auto complète en fonction du code en entré.
* Simulateur: c’est un logiciel à l’intérieur d’un environnement de développement qui permettant de tester l’exécution de son logiciel.
* Compilateur: c’est celui-là qui va transformer le code source rédigé par le développeur en code binaire, composé de 0 et de 1 uniquement pour que l’ordinateur le comprenne.
* Débogueur: c’est une fonctionnalité qui nous aide à corriger les bugs. Dans le contexte Visual Studio, quand on débogue notre application, cela signifie généralement que nous exécutons notre application en y ayant attaché le débogueur (c’est-à-dire en mode Débogueur). Quand on faites cela, le débogueur fournit de nombreuses façons de voir ce que fait notre code pendant qu’il s’exécute. On peut parcourir notre code et examiner les valeurs stockées dans des variables, définir des espions sur des variables pour voir quand les valeurs changent, et d’examiner le chemin d’exécution de notre code [IV].

Il existe de nombreux IDE. Certains permettent de développer pour un système d’exploitation spécifique, d’autres sont polyvalents. Voici quelques exemples d’IDE: Scilab , Visual Studio, Eclipse, NetBeans, Delphi, Code::Blocks, MonoDevelop, SharpDevelop, KDevelop, Access, C++ Builder, MATLAB, LabVIEW, Dev-C++, Qt Creator, XCode, Lazarus, WinDev, 4D, Dreamweaver, Zend Studio,…

### b. Le Visual Studio Code:

VS Code est un éditeur de code **léger** et **rapide** qui permet d’ajouter des **extensions**. Ce qui signifie que vous pouvez intégrer de nombreux langages et fonctionnalités.

Par rapport à un éditeur de texte basique, voici quelques exemples de fonctionnalités :

* **la coloration syntaxique du code** : en PHP et dans tous les langages, les mots écrits obéissent à une certaine grammaire : des mots clés propres au langage (en Symfony, par exemple, class, public, …), des variables (un libellé qui contient une valeur), des valeurs (nombres, chaînes de caractères), etc. Utiliser des couleurs différentes en fonction de cette grammaire permet d'améliorer grandement la lisibilité du code ;
* aide à **l'indentation du code** : en Java, le code est formé par des blocs qui se suivent et qui s'imbriquent. Pour mieux les distinguer, on utilise des espaces en début de ligne, un peu comme les paragraphes d'un roman. VS Code permet de reformater un code pour que ces espaces soient cohérents et mieux lisibles ;
* **l'autocomplétion** : VS Code peut deviner ce que vous souhaitez coder et le proposer, un peu comme sur un moteur de recherche Internet ;
* d'autres facilités diverses pour éditer du code, par exemple l'affichage de l'arborescence des fichiers à côté du code.
* **Fonctionnalités supplémentaires de Visual Studio Code**

Jetons maintenant un œil à quelques-unes des fonctionnalités qui font de Visual Code Studio un outil puissant et utilisable gratuitement pour développer des logiciels sur votre ordinateur personnel !

Le côté gauche du logiciel contient un **explorateur de fichiers**, une **barre de recherche**, un **débogueur** et une **option d’extension** ; ce dernier point étant particulièrement intéressant et utile.

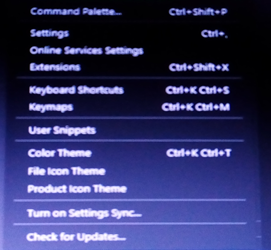
**L’onglet Extensions**

Les extensions sont de petits programmes que vous pouvez ajouter à Visual Studio Code pour **améliorer** ses fonctionnalités et le **personnaliser**. En ayant saisi du code HTML, VS Code propose de **télécharger** des extensions liées au HTML.

Commencez par cliquer sur l’icône du bas dans le panneau de gauche, appelée "Extensions". Elle mène à la marketplace des extensions qui peuvent être installées sur Visual Studio Code.



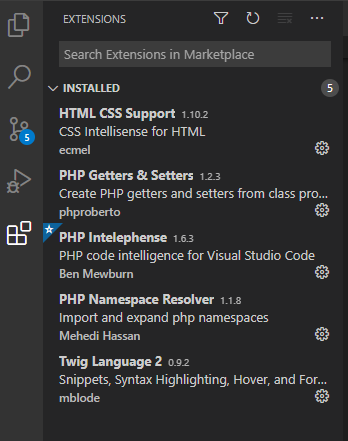
1



2

*Figure 2 : Onglet d’extension*

À gauche, on voit les extensions qui sont, on peut trouver des extensions en les tapant dans la barre de recherche en haut de la fenêtre. C’est comme pour chercher une application à installer sur notre téléphone.



Pour la recherche d’extension

Les listes des extensions qu’on a installées

*Figure 3 : Extension dans Visual Studio Code*

### 2. Utilisation du FRAMEWORK PHP

### a. C’est quoi un FRAMEWORK?

Un **Framework** est, comme son nom l’indique en anglais, un “**cadre de travail**“. L’objectif d’un Framework est généralement de **simplifier le travail des développeurs informatiques**, en leur offrant une **architecture “prête à l’emploi”** et qui leur permette de ne pas repartir de zéro à chaque nouveau projet [V].

Le Framework nous apporte beaucoup d’avantage, nous allons expliquer quelque notion en utilisant un FRAMWORK peut import si on travaille seul ou en équipe.

* **Une organisation à notre projet:**

En développant sur les FRAMEWORKS, nos projets vont être faits suivant ces étapes:

* **découpage** logique du code source
* **factorisation** de composants communs, réutilisabilité  du code
* ajout de couche **métier**
* séparation des logiques techniques / métiers (développeurs) et logiques de présentation (designeurs / intégrateurs)
* maintenance et **évolutivité**
* **Des composants et bibliothèques réutilisables [VI]:**

Bien qu’un Framework comme Symfony 2 se positionne comme **chef d’orchestre** plutôt que comme  simple agrégat de libraires, il n’empêche que les Frameworks viennent systématiquement avec leur lot de **composants réutilisables**. Que ce soit des modules, des plugins, des bundles ou tout ce qui nous fera plaisir.

* **Une incitation aux bonnes pratiques [VII]:**

C’est surtout vrai avec les **versions 2** des frameworks les plus connus (Zend, Symfony, Cake). Jusqu’à présent, chacun codait dans son coin, sans trop s’occuper des autres. Depuis quelques années, le monde PHP s’organise, se professionnalise et les développements **s’industrialisent**. Avec l’arrivée des recommandations PSR et du FIG, le code source se standardise. Les frameworks PHP actuels ne cessent de mettre en avant ces bonnes pratiques afin de rendre compatible le code issu de différents frameworks.

Ces bonnes pratiques sont le garant d’un code **lisible et compréhensible** par tous les développeurs avertis.

* **Une base régulièrement mise à jour [VIII]:**

En choisissant un framework, on choisit également une **communauté** active qui détectera et corrigera des failles ou des manques du framework. En quelque sorte, on mutualise les moyens de développements. Vous profiterez ainsi des mises à jour du framework avec le lot d’améliorations (et de nouveaux bugs) qu’elles comportent.

### b. Le Framework SYMFONY

**Symfony** est un ensemble de composants PHP ainsi qu'un Framework MVC libre écrit en PHP. Il fournit des fonctionnalités modulables et adaptables qui permettent de faciliter et d’accélérer le développement d'un site web [IX].

La technologie Symfony intègre des mesures de sécurité innées afin de lutter contre des failles et attaques XSS, CSRF et SQL. Ce sont des failles permettant d’injecter du code HTML dans des variables mal protégées [X].

Pour mieux développer en symfony, on a besoin d’en avoir quelque outil pour y aller rapidement:

**Terminal**: système d’exécution de tâche en ligne de commande et facilitant la génération de plusieurs autre composants;

**Git:** système pour enregistrer chaque modification de notre projet;

**Composer**: système pour gérer les dépendances;

**Symfony CLI:** système qui nous permet de pouvoir installé la commande symfony, de pouvoir utiliser HTTPS localement (site sécuriser) en exécutons la commande:

$ symfony server:ca:install

**Installation de Symfony:**

Pour installer symfony, il y a deux solutions possibles: soit avec un installer de symfony, soit avec composer. Mais avant de l’installer, il faut vérifier qu’on a la version récente de PHP (7 par exemple); sinon installons un serveur de notre choix, le lancer et de démarrer les serveurs.

* Installer symfony avec un installer symfony: on télécharge un fichier exécutable dans le site de symfony: <https://get.symfony.com/cli/setup.exe> puis l’installé (sous WINDOWS).

Sous Linux de même pour le Mac, cela se fait avec une commande:

wget https://get.symfony.com/cli/installer -O - | bash

Linux:

curl -sS https://get.symfony.com/cli/installer | bash

Mac :

* Installer symfony via composer: on télécharge un fichier exécutable dans le site: <https://getcomposer.org/Composer-Setup.exe>, on lance l’installation de  « composer » depuis le fichier télécharger. Ensuite, on exécute la commande:

composer create-project symfony/website-skeleton project\_name

Voici quelques commandes assez importantes à l’installation de symfony:

|  |  |
| --- | --- |
| Commandes | Utilisation |
| composer create-project symfony/skeleton project\_name | Pour créer un projet console ou microservice Symfony 5 en utilisant Composer. |
| symfony -V | Pour vériﬁer la version utilisée de Symfony. |
| symfony php -v | Pour connaitre la version de PHP utilisée par Symfony. |
| * cd project\_name * symfony server:start ou symfony serve | Pour lancer un projet Symfony 5 créé avec la commande symfony. |
| git clone lien\_vers\_repository\_github | cloner un projet existant |
| cd nom\_projet | se positionner dans le projet |
| composer install | installer les dépendances composer dans vendor |

Tableau des commandes pour l’installation

Sous Visual Studio Code, l’installation de l’extension PHP Namespace Resolver nous permet d’importer les namespaces nécessaires pour certains classes.

**Structure et fonctionnement de symfony:**

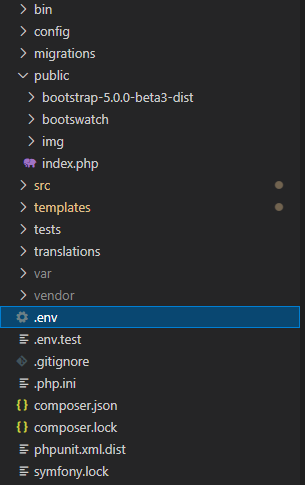
* bin/ : contenant deux exécutables, la console de Symfony et PHPUnit
* config/ : pour les ﬁchiers de conﬁguration (routes, ORM...)
* migrations/ : pour enregistrer les données de migration de la base de données
* public/ (appelé web dans la version 3) : seul dossier accessible de l’extérieur (contenant le contrôleur frontal index.php)
* src/ : pour les ﬁchiers sources de type classes (contrôleurs, entités, formulaires, DAO...)
* templates/ : contenant les vues (vue partielle) de l’application
* tests/ : pour les ﬁchiers de test
* translations/ : pour les ﬁchiers de l’internationalisation
* var/ (utilisé pendant l’exécution) : contenant les données de cache, le log, les sessions...
* vendor/ : contenant les packages nécessaires pour Symfony (mentionnés dans composer.json)

Dans le fichier « src » se trouve un fichier appelé **Kernel.php,** c’est le noyau de symfony; il est utilisé par le contrôleur frontal pour désigner le contrôleur adéquat pour répondre à la requête HTTP reçue. Le contrôleur frontal: c’est le point d’entrée d’une application symfony; il est défini dans le dossier **public/index.php.**

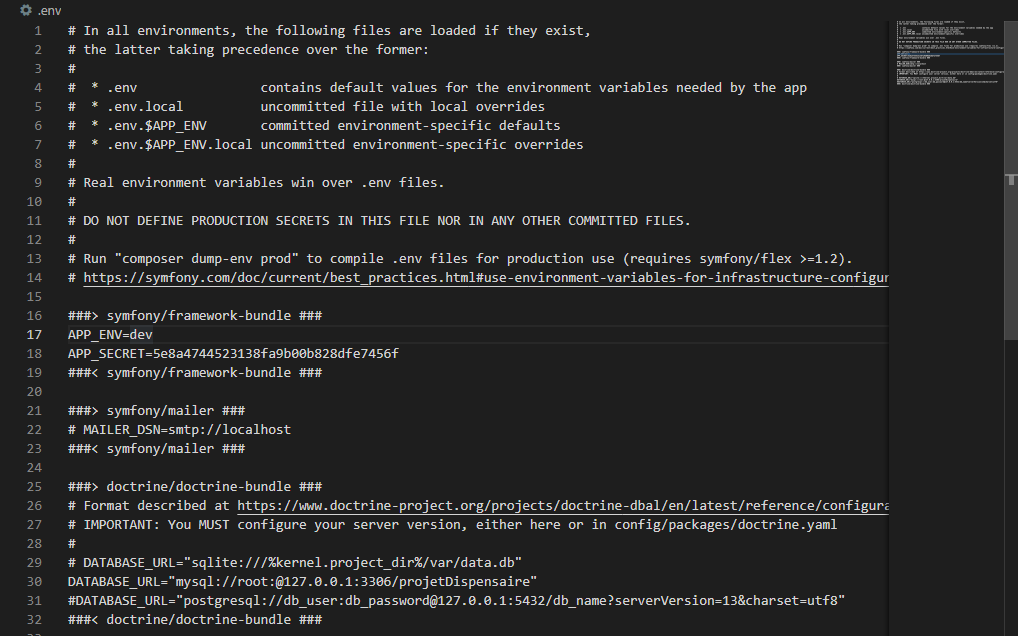
En développement symfony, il y a *deux environnements de travail*:

* **prod**: qui est destiné aux utilisateurs finaux de l’application.
* montrant l’application telle qu’elle sera visible par les visiteurs.
* rapide à exécuter
* n’affichant pas les messages d’erreur.
* **dev**: qui est destinés aux développeurs
* Plus lent que la version de production
* Environnement de débogage complet
* Possibilité d’ajouter des nouvelles fonctionnalités

Par défaut, une application en symfony est configurée à l’environnement **dev**; pour changer d’environnement, on va dans le fichier **.env** et modifier la valeur de **APP\_ENV** à **prod**.



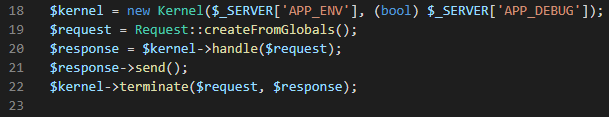
1



Changer en **prod**

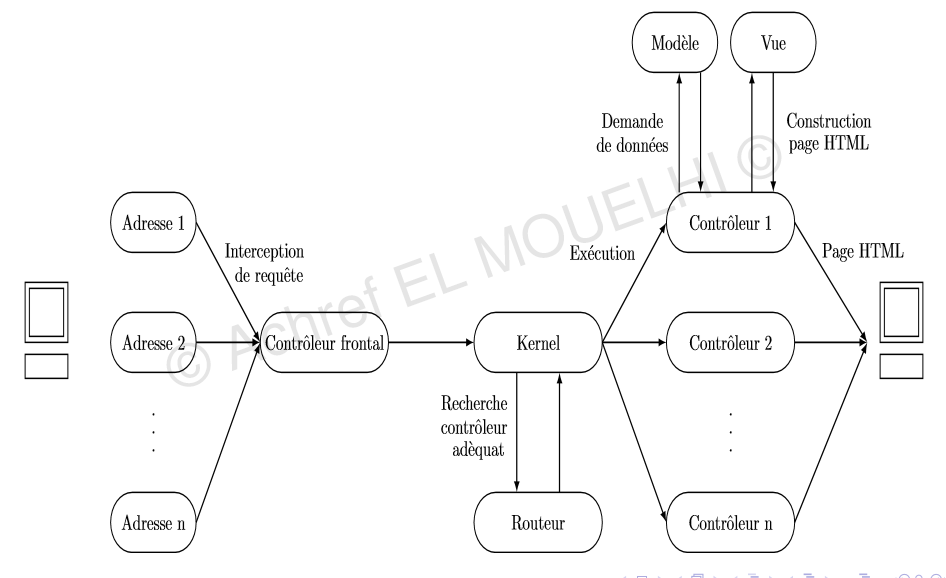
*Figure 4 : Changer en mode prod*

Voici un extrait d’index.php d’un projet web de symfony:



*Figure 5 : extrait du code de la requête*

Le contrôleur frontal charge le *kernel* selon l’environnement précisé dans **.env,** récupère la requête et utilise le *kernel* pour retourner une réponse.



*Figure 6 : Déroulement d’un site en symfony*

1. L’utilisateur saisit l’adresse du site
2. Le contrôleur frontal intercepte la requête et il la transmet au Kernel
3. Le Kernel demande au Routeur le contrôleur adéquat à la page demandée
4. A la réception d’une réponse, le Kernel exécute le contrôleur
5. Le contrôleur communique avec le modèle pour récupérer ou stocker certaines données
6. Ensuite il renvoie ces données à la vue pour qu’elle construise la page HTML et la lui retourne.
7. Enﬁn le contrôleur envoie à l’utilisateur la réponse (page HTML).

### 3. Utilisation d’une bibliothèque BOOTSRAP [XI]

Définition

Bootstrap un framework frontale à code source ouvert qui a pour but d’accélérer et faciliter le développement de sites web frontaux. Il contient toutes sortes de modèles de conception basés sur HTML et CSS pour diverses fonctions et composants tels que la navigation, le système de grille, les carrousels d’images et les boutons.

Bien que Bootstrap fasse gagner du temps aux développeurs en leur évitant de devoir gérer les modèles de façon répétitive, son objectif premier est de créer des sites réactifs. Il permet à l’interface utilisateur d’un site web de fonctionner de manière optimale sur toutes les tailles d’écran, que ce soit sur des téléphones à petit écran ou des ordinateurs de bureau à grand écran.

Comme **Bootstrap** est constitué d’une collection de syntaxes qui remplissent des fonctions spécifiques, il est logique que le framework contienne différents types de fichiers. Voici les trois principaux fichiers qui gèrent l’interface utilisateur et les fonctionnalités d’un site web.

### Bootstrap.css

Bootstrap.css est un framework CSS qui organise et gère la mise en page d’un site web. Alors que le HTML gère le contenu et la structure d’une page web, le CSS s’occupe de la mise en page du site. Pour cette raison, les deux structures doivent coexister pour effectuer une action particulière.

Grâce à ses fonctions, le CSS nous permet de créer un aspect uniforme sur autant de pages web que nous le souhaitons.

Avec le CSS, il suffit de renvoyer les pages web au fichier CSS. Toute modification nécessaire peut être effectuée dans ce seul fichier.

Les fonctions du CSS ne se limitent pas aux seuls styles de texte car elles peuvent être utilisées pour formater d’autres aspects de la page web tels que les tableaux et les mises en page d’images.

### Bootstrap.js

Ce fichier est la partie centrale de Bootstrap. Il est constitué de fichiers [**JavaScript**](https://www.hostinger.fr/tutoriels/javascript-introduction-basique) qui sont responsables de l’interactivité du site web.

Pour gagner du temps en évitant d’écrire de nombreuses fois des syntaxes JavaScript, les développeurs ont tendance à utiliser jQuery. Il s’agit d’une bibliothèque JavaScript multiplateforme à code source ouvert très répandue qui permet d’ajouter diverses fonctionnalités à un site web.

Voici quelques exemples de ce que jQuery peut faire :

* Effectuer des requêtes Ajax comme la soustraction dynamique de données d’un autre emplacement
* Créer des widgets à l’aide d’une collection de plugins JavaScript
* Créer des animations personnalisées en utilisant les propriétés CSS
* Dynamiser le contenu du site

Alors qu’un Bootstrap avec des propriétés CSS et des éléments HTML peut fonctionner parfaitement, il a besoin de jQuery pour créer un design réactif.

### Glyphicons

Les icônes font partie intégrante de la partie frontale d’un site web. Elles sont souvent associées à certaines actions et données dans l’interface utilisateur. Bootstrap utilise des glyphes pour répondre à ce besoin.

## II. Méthodes

### 1. Merise [XII]

MERISE est une méthode d'analyse et de conception des systèmes d'information basée sur le principe de la séparation des données et des traitements.

## Méthode d'analyse et de conception

La méthode Merise d'analyse et de conception propose une démarche articulée simultanément selon 3 axes pour *hiérarchiser* les préoccupations et les questions auxquelles répondre lors de la conduite d'un projet:

* **Cycle de vie** : phases de conception, de réalisation, de maintenance puis nouveau cycle de projet.
* **Cycle de décision** : des grands choix, la définition du projet (étude détaillée) jusqu'aux petites décisions des détails de la réalisation et de la mise en œuvre du système d'information. Chaque étape est documentée et marquée par une prise de décision.
* **Cycle d'abstraction** : niveaux conceptuels, logique/organisationnel et physique/opérationnel (du plus abstrait au plus concret) L'objectif du *cycle d'abstraction* est de prendre d'abord les grandes décisions métier, pour les principales activités (Conceptuel) sans rentrer dans le détail de questions d'ordre organisationnel ou technique.

La méthode Merise, très analytique, distingue nettement les données et les traitements, même si les interactions entre les deux sont profondes et s'enrichissent mutuellement (validation des données par les traitements et réciproquement).

### Niveau conceptuel

L'étude *conceptuelle* Merise s'attache aux *invariants* de l'entreprise ou de l'organisme du point de vue du métier : quels sont les activités, les métiers gérés par l'entreprise, quels sont les grands processus traités, de quoi parle-t-on en matière de données, quelles notions manipule-t-on ?... et ce indépendamment des choix techniques (comment fait-on ?) ou organisationnels (qui fait quoi ?) qui ne seront abordés que dans les niveaux suivants.

* MCD: schéma représentant la structure du système d'information, du point de vue des données, c'est-à-dire les dépendances ou relations entre les différentes données du système d'information.

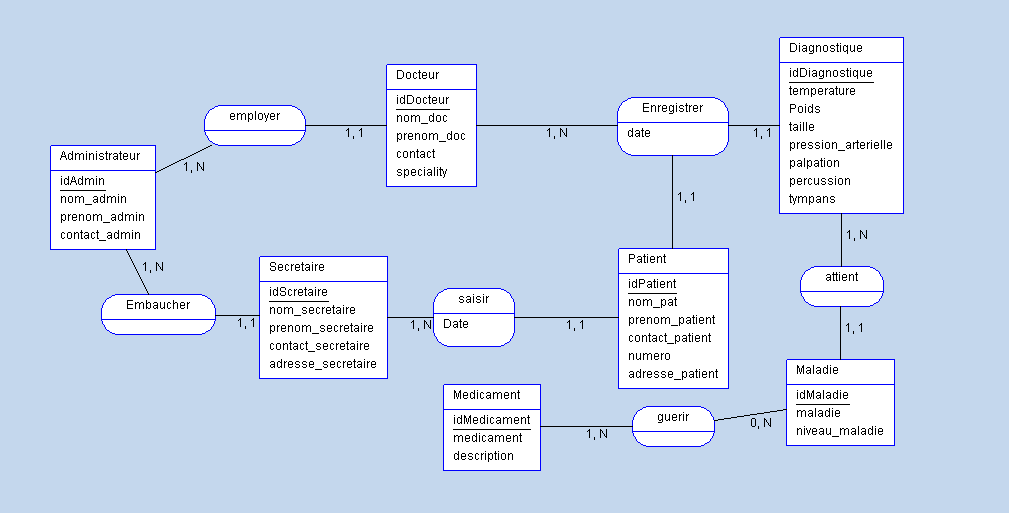
Le MCD repose sur les notions d'entité et d'association et sur les notions de relations.

**L'entité** est définie comme un *objet de gestion* considéré d'intérêt pour représenter l'activité à modéliser et chaque entité est porteur d'une ou plusieurs propriétés simples, dont l'une, unique et discriminante, est désignée comme identifiant.

##### L'association ou relation

L'association est un lien sémantique entre une ou plusieurs entités : l'association peut être réflexive, de préférence binaire, parfois ternaire, voire de dimension supérieure. Elle peut également être porteuse d'une ou plusieurs propriétés.

Cette description sémantique est enrichie par la notion de cardinalité, celle-ci indique le nombre minimum (0 ou 1) et maximum (1 ou n) de fois où une occurrence quelconque d'une entité peut participer à une association.

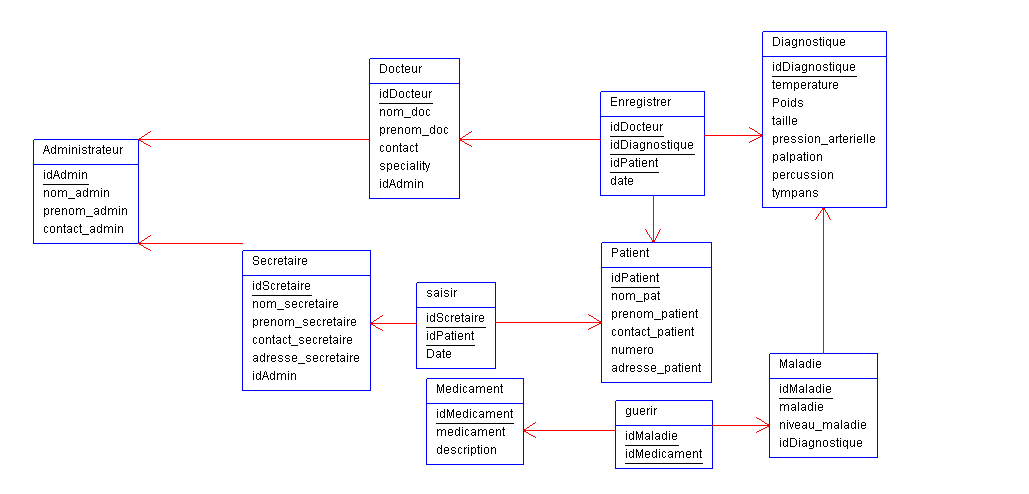
Dans notre projet, le MCD de base est présenté à la figure ci-dessous:

*Figure 7 : Modèle conceptuel des Données*

On la conçut avec l’outil AnalyseSI.jar, cet outil est réalisé avec le langage Java et son extension donc est .jar (java archive), ce qui nécessite d’installer une version de Java sur notre ordinateur. AnalyseSI est un simple outil qui nous permet de construire des MCD et de générer ensuite les MLD, ainsi que les scripts SQL.

En outre, ce modèle conceptuel de données nous a permis d’avoir le modèle physique de données. Cette étape permet de **construire la structure finale de la base de données** avec les différents liens entre les éléments qui la composent. Pour la peine, on change aussi de vocabulaire :

* Les entités se transforment en tables ;
* Les propriétés se transforment en champs (ou attributs) ;
* Les propriétés se trouvant au milieu d’une relation génèrent une nouvelle table ou glissent vers la table adéquate en fonction des cardinalités de la relation ;
* Les identifiants se transforment en clés et se retrouvent soulignés. Chaque table dispose d’au minimum 1 clé dite primaire ;
* Les relations et les cardinalités se transforment en champs parfois soulignés : il s’agit de créer des « clés étrangères » reliées à une « clé primaire » dans une autre table [XIII].



*Figure 8 : Modèle Logique des Données*

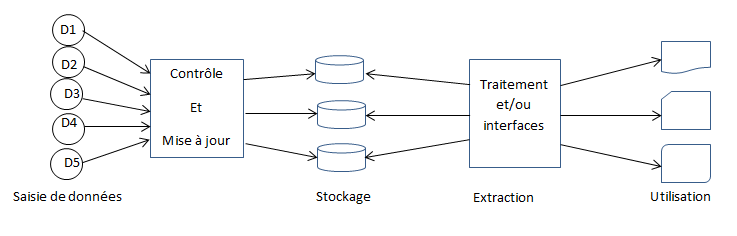
### 2. Base de données [XIV]

Une base de données (que nous nommerons BDD par commodité) est une **collection d’informations organisées afin d’être facilement consultables, gérables et mises à jour**. Au sein d’une database, les données sont organisées en lignes, colonnes et tableaux. Elles sont indexées afin de pouvoir facilement trouver les informations recherchées à l’aide d’un logiciel informatique. Chaque fois que de nouvelles informations sont ajoutées, les données sont mises à jour, et éventuellement supprimées.

# Système de Gestion de Base de Données:

Un **Système de Gestion de Base de Données** (SGBD) est un logiciel qui permet de stocker des informations dans une base de données. Un tel système permet de lire, écrire, modifier, trier, transformer ou même imprimer les données qui sont contenus dans la base de données.

La manipulation de données est une des utilisations les plus courantes des ordinateurs. Les bases de données sont par exemple utilisées dans les secteurs de finance, des assurances, des écoles, de l’administration publique et des médias.



*Figure 9 : Mode de traitement des données*

### Les différents systèmes de gestion de base de données



*Figure 10 :* Différents systèmes de gestion de base de données

MySQL est un serveur de bases de données relationnelles SQL développé dans un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mises à jour fréquentes et fortement sécurisées. Il est multi-thread et multi-utilisateur.

C'est un logiciel libre, open source, développé sous double licence selon qu'il est distribué avec un produit libre ou avec un produit propriétaire. Dans ce dernier cas, la licence est payante, sinon c'est la licence publique générale GNU (GPL) qui s'applique. Un logiciel qui intègre du code MySQL ou intègre MySQL lors de son installation devra donc être libre ou acquérir une licence payante. Cependant, si la base de données est séparée du logiciel propriétaire qui ne fait qu'utiliser des API tierces, alors il n'y a pas besoin d'acquérir une licence payante MySQL.

**XAMPP [XV]:**

XAMPP est un **ensemble de logiciels libres**. Ce dernier réunit le serveur Web **A**pache, la base de données relationnelle et système d’exploitation **M**ySQL ou **M**ariaDB ainsi que les langages scripts **P**erl et **P**HP.

* **Apache:** le serveur Web open source Apache est utilisé mondialement et permet de délivrer des contenus Web. L’application de serveur est mise à disposition en open source par l’*Apache Software Foundation*.
* **MySQL/MariaDB:** avec MySQL, XAMPP se compose de l’un des systèmes de gestion de base de données relationnelle les plus populaires au monde. En combinaison avec le serveur Web Apache et le langage script PHP, MySQL sert à l’enregistrement de données pour des services Web. Les versions

actuelles de XAMPP favorisaient MariaDB à l’insu de MySQL comme gestionnaire de base de données, marquant un détachement avec ce dernier.

* **PHP:** Il s’agit d’un langage script côté serveur permettant de créer des pages Web ou applications dynamiques. PHP peut être mis en place sur toutes les plateformes possibles et est compatible avec divers systèmes de base de données.
* **Perl:** le langage script Perl est utilisé pour l’administration système, le développement Web et la programmation en réseau. De plus, des applications Web dynamiques peuvent être programmées de la même manière que PHP.

Un **système de test local XAMPP** s’installe rapidement et facilement avec un seul fichier exécutable sous Linux, Windows et Mac OS X. L’ensemble de logiciels contient les mêmes composants que ceux aussi utilisés sur des serveurs Web courants. Les développeurs ont donc la possibilité de tester des projets localement et de les transmettre à des systèmes productifs confortablement. XAMPP n’est cependant pas idéal en tant que serveur public car il comporte des **restrictions en matière de sécurité**.

# C- REALISATION ET RESULTAT

## I. Réalisation

La réalisation consiste à la description de l’application, comment ça fonctionne, quelles sont les fonctions disponibles et de les décrire étape par étape.

Pour commencer, la page d’accueil de notre application se présente comme suit:



*Figure 11 : Fenêtre principale*

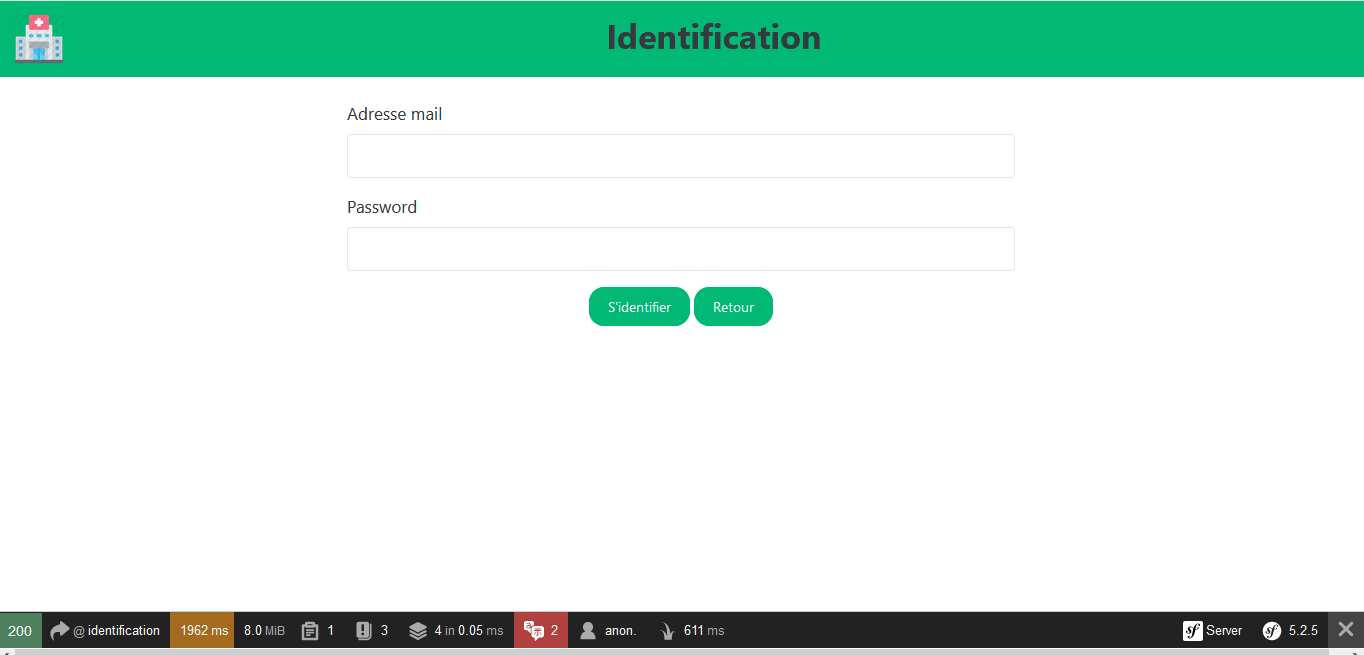
On a conçus la page d’accueil en admettant que personne ne peut effectuer une action sans se connecter. C’est un page simple qui énumère les fonctionnements du logiciel.

* **L’Utilisateur**

En termes d’utilisation, il y a deux catégories d’utilisateur dans cette application:

* Administrateur: l’administrateur est comme le chef d’orchestre, c’est celui qui a le pouvoir de gérer les personnels, d’y ajouter, modifier ou supprimer.
* Simples utilisateurs: ce sont les personnels ajouté par l’administrateur; ses derniers n’ont aucun accès à l’application que si et seulement l’administrateur leur réserve une fonction. Ils ne peuvent pas ajoutés des nouveaux personnels, par contre, l’ajout d’un patient est disponible selon la fonction définie.
* **L’Authentification**

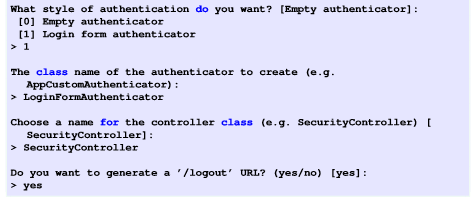
En appuyant sur le bouton « **Se connecter** », on sera redirigé à une page d’authentification.



*Figure 12 : Fenêtre d’Authentification*

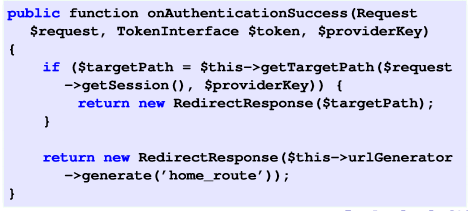
En remplissant correctement les champs, l’utilisateur sera redirigé à la page qui est convenu à son adresse email et son mot de passe. C’est-à-dire que si les données entrées corresponds à l’adresse email et mot de passe d’un docteur, l’utilisateur sera dirigé à la page des docteurs. Ce concept est fait en symfony avec deux cas possible:

Première cas: en exécutant la commande **php bin/console make:auth**  depuis le terminal, et de répondre au choix proposé comme l’exemple suivante:



*Figure 13 : Extrait du code pour créer l’Authentification*

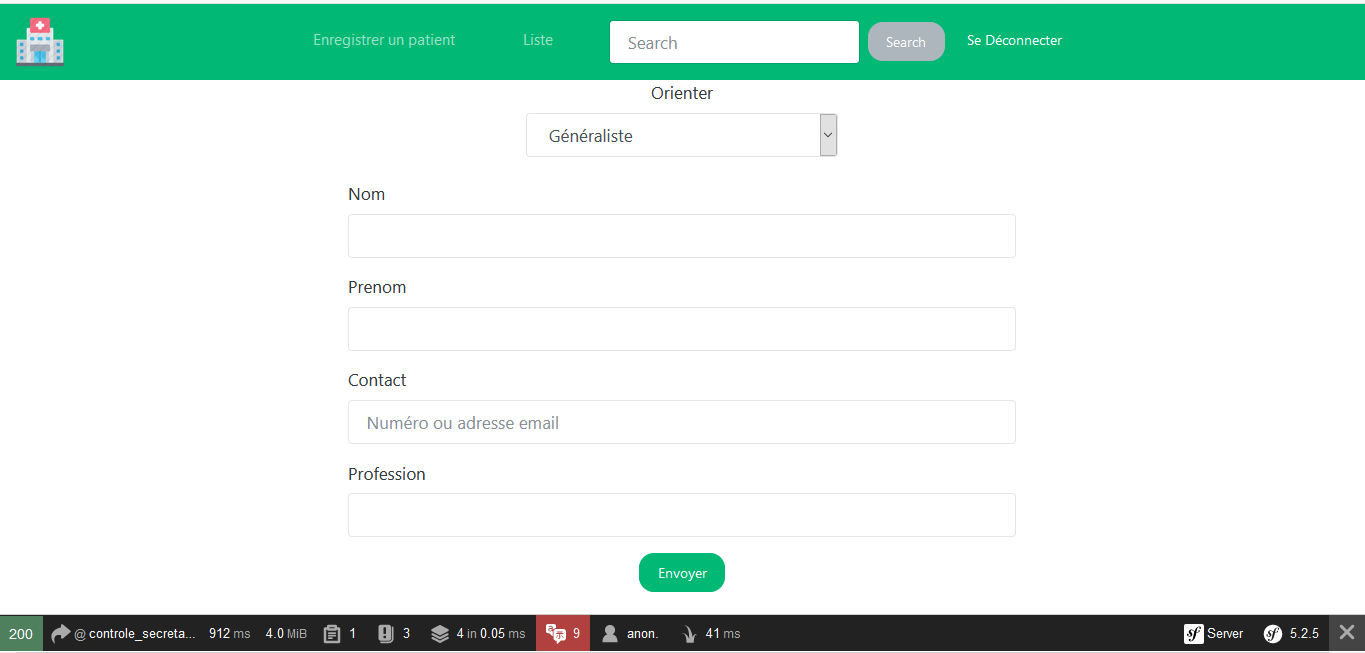
Ensuite, de modifier la méthode **onAuthenticationSuccess** défini dans security/LoginFromAuthenticator pour être rediriger vers la route convenable:



*Figure 14 : Extrait du code de redirection d’Authentification*

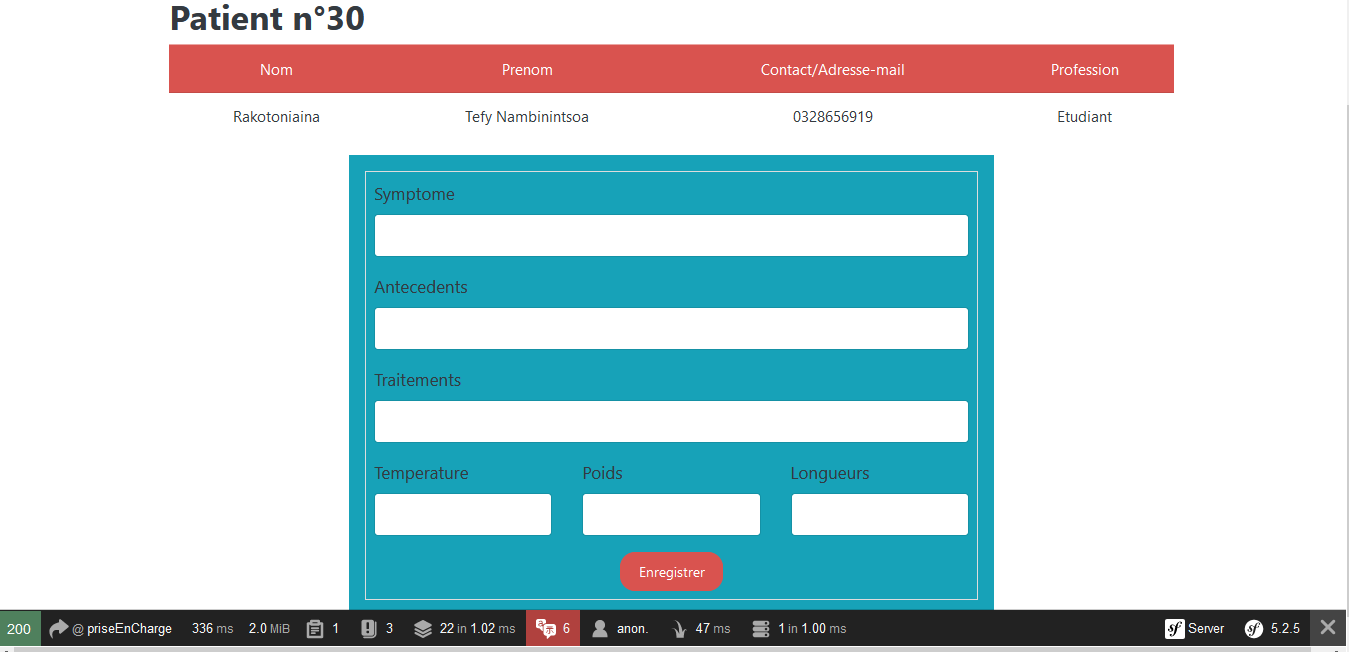
Deuxième cas: en créant notre propre formulaire d’authentification, on crée un formulaire avec la commande **php bin/console make:form**; ensuite de créer une route pour la redirection. Pour créer une route, on exécute la commande **php bin/console make:controller,** ensuite dans le fichier Controller qui a été créé on génère notre route sous forme de méthode.

* **Espace Utilisateur**
* Espace secrétaire: pour y accédé, l’utilisateur doit avoir l’adresse email et le mot de passe convenable. En ayant entrée les données correspondant, l’utilisateur sera redirigé sur la page d’accueil des secrétaires représenter comme suit:



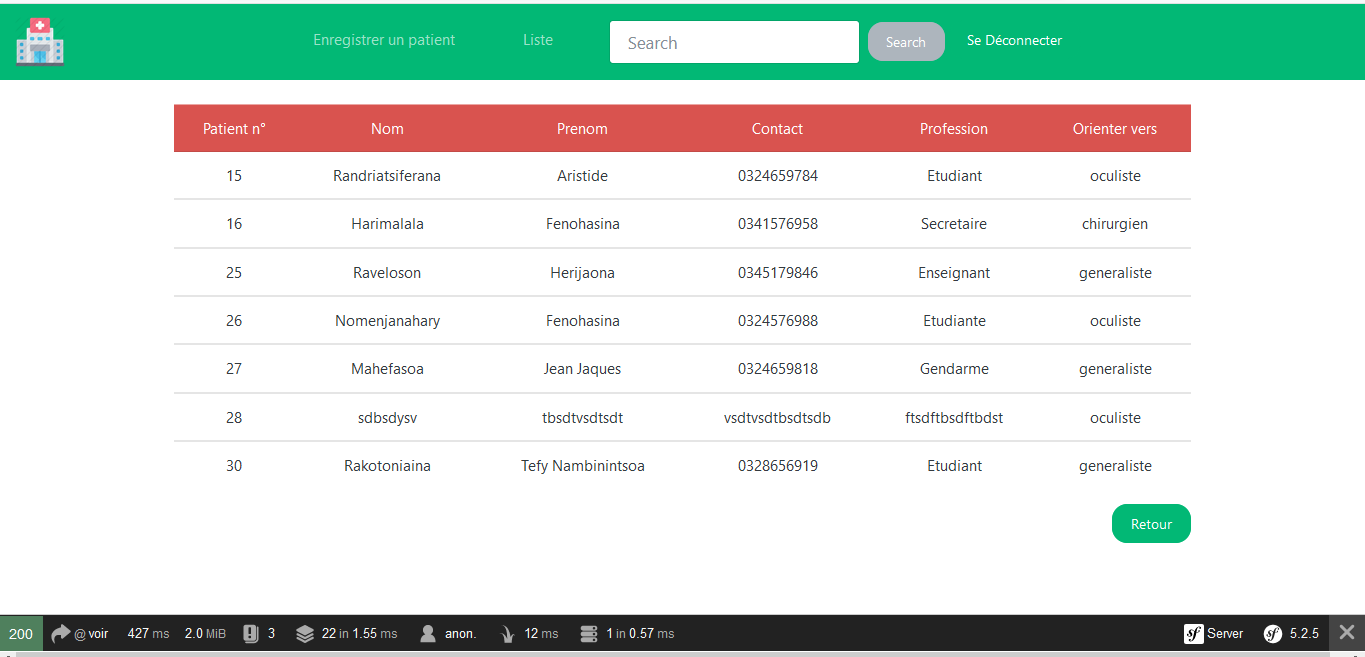
*Figure 15 : Ajout d’un patient*

Son accueil possède des liens qui ont pour fonction de voir la liste des patients enregistrer, d’enregistré un nouveau patient, de faire une recherche sur un patient enregistré et un lien pour la déconnexion. L’accueil du secrétaire se positionne directement sur l’enregistrement d’un nouveau patient, ce dernier oriente le patient vers la spécialité des docteurs et ensuite de remplir le formulaire de renseignement concernant le patient. En enregistrant les renseignements, l’utilisateur est redirigé vers une page demandant plus de détail sur le patient afin de facilité le travail des docteurs concernant le diagnostic.



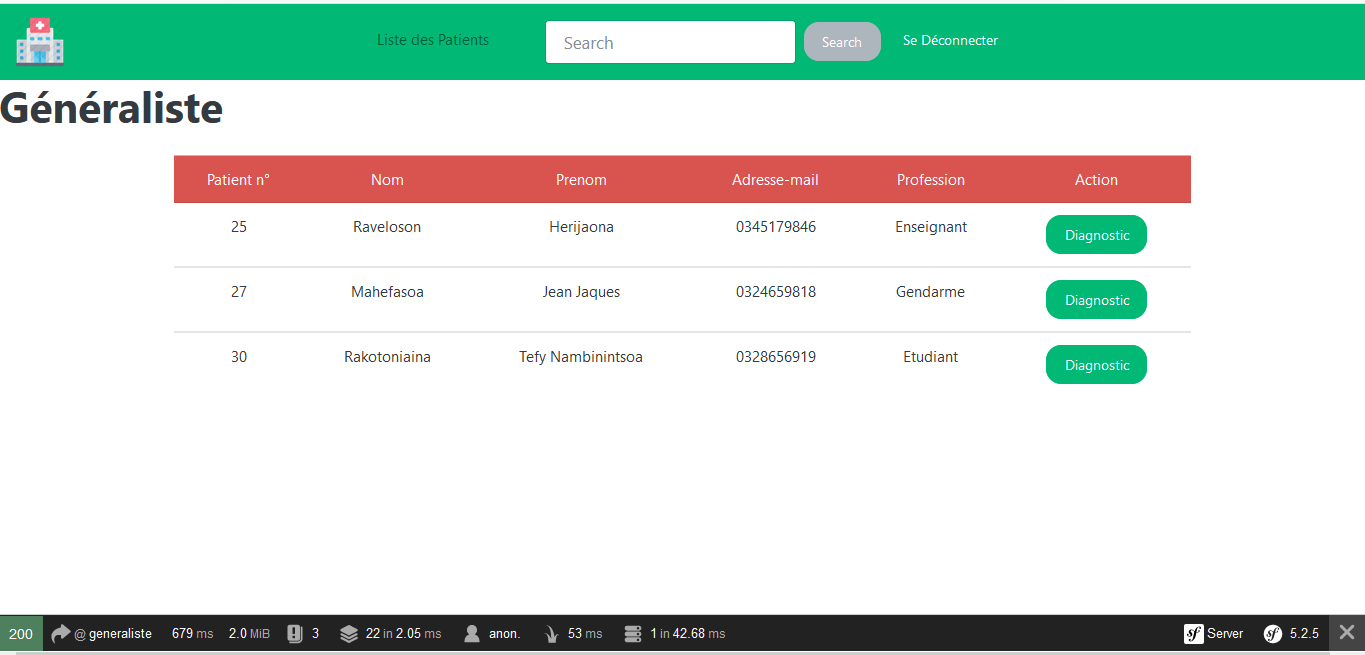
*Figure 16 : Prise de charge*

En remplissant cette prise de charge, l’utilisateur est dirigé vers la liste des patients.



*Figure 17 : Liste des patients*

* Espace docteur: comme tous les utilisateurs, les docteurs doivent avoir le bon adresse email et le mot de passe pour accédés à son page. Pour plus d’info, les données entrées par l’utilisateur sont traitées et comparées à ceux enregistré dans la base de données afin de rediriger l’utilisateur vers la bonne page. L’accueil des docteurs se présente comme suit:



*Figure 18 : Page d’accueil du docteur*

Une page simple représentant la liste des patients qui ont été enregistré par le secrétaire, ces patients sont listés dans un tableau avec leur information et une action de plus pour enregistrer le diagnostic. En cliquant sur le bouton **Diagnostic**, l’utilisateur sera redirigé vers une page contenant un formulaire pour que le docteur puisse introduire les informations à enregistré.



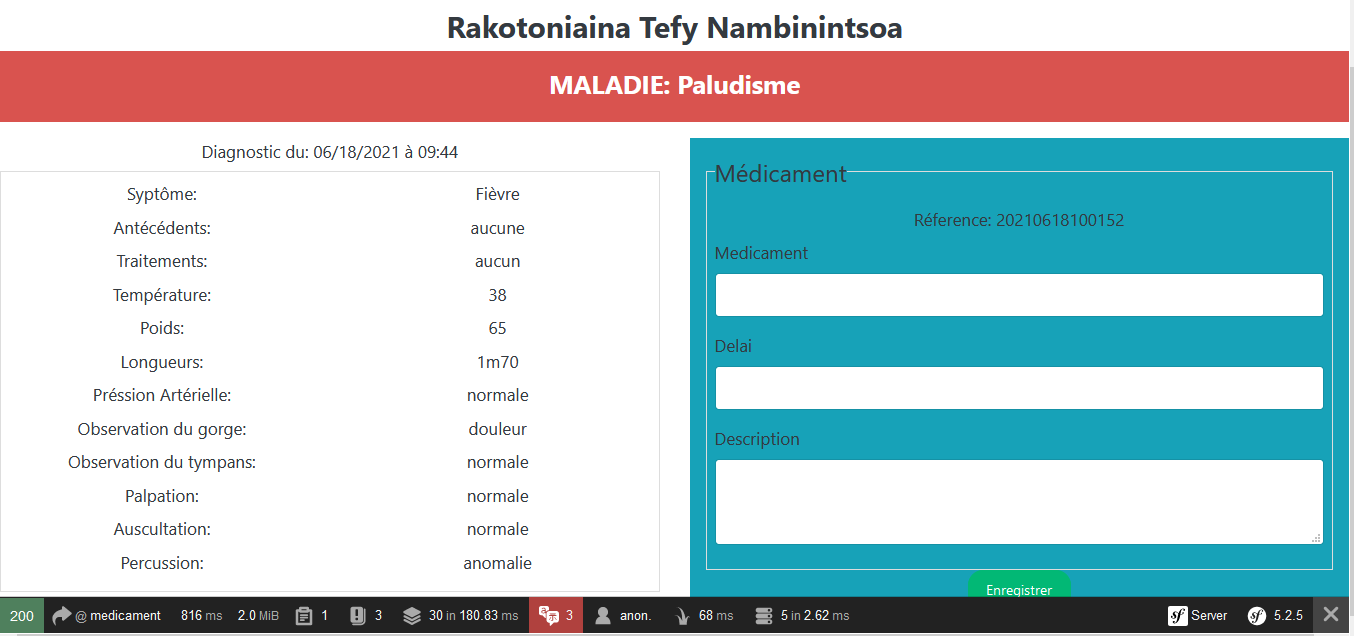
*Figure 19 : Page du diagnostique*

En enregistrant ces informations à propos du diagnostic, l’utilisateur sera dirigé vers une page montrant toutes les informations qui sont enregistrées par les personnels concernant le patient. Afin que le docteur puisse deviner la maladie et le niveau d’intervention sur le patient.



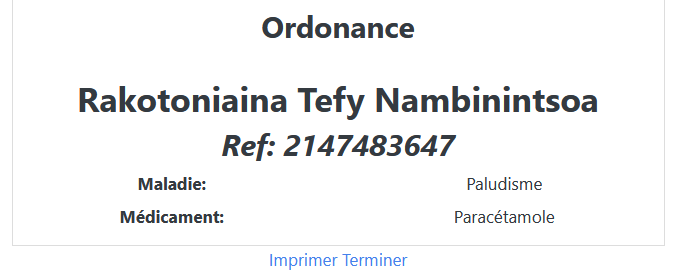
*Figure 20 : Ajout de la maladie*

Après l’ajout de la maladie, l’utilisateur accède à la formulaire pour souscrire la médicament.

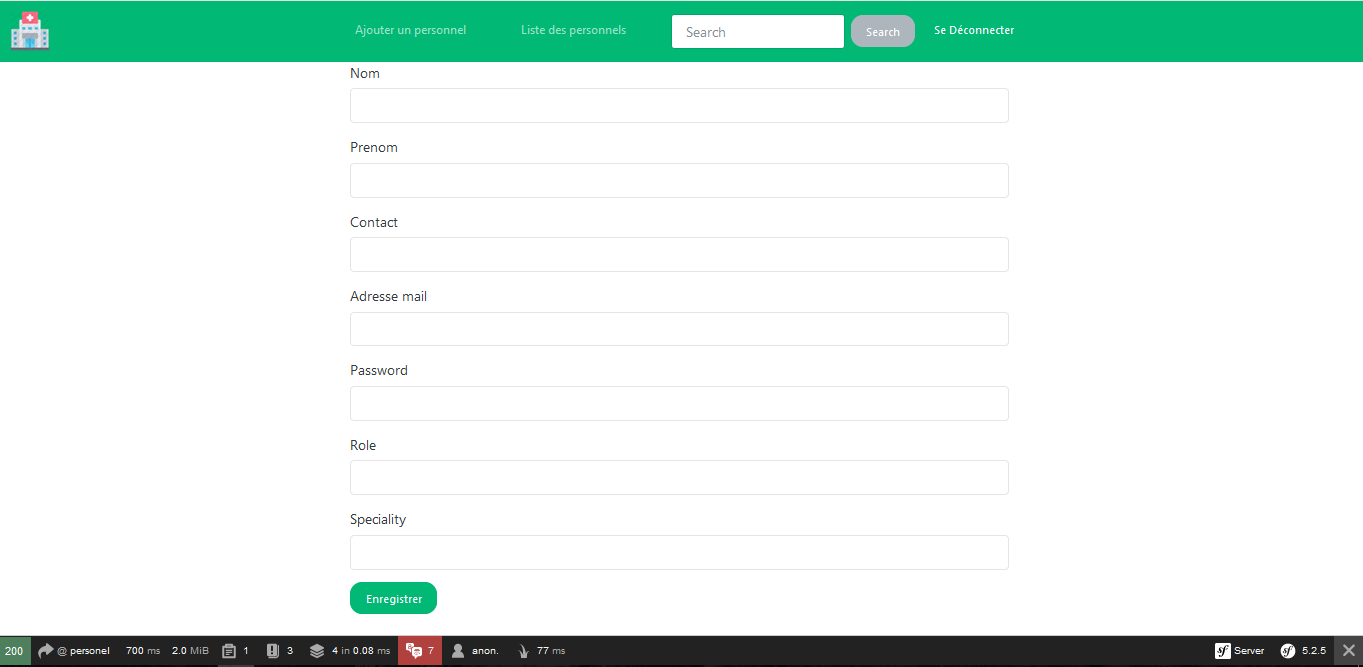


*Figure 21 : Ajout du médicament*

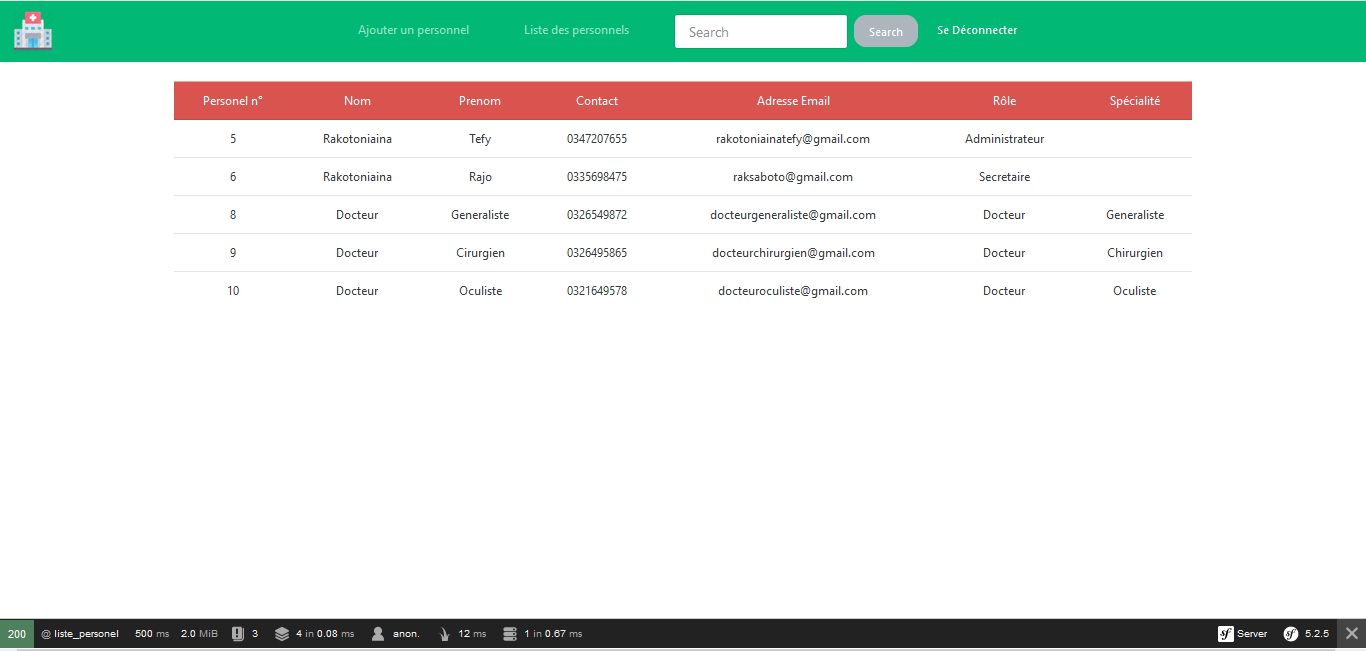
La référence sera un moyen pour la pharmacie de savoir et de vérifier si la personne est bel et bien un patient. Pour finir sur les fonctions disponibles sur la page du docteur, en enregistrant le médicament, il est dirigé vers une page d’ordonnance qui sera ensuite imprimer pour l’usage à la pharmacie.

 *Figure 22 : Ordonnance*

* Espace administrateur: l’accueil de l’administrateur se présente à l’ajout d’un nouveau personnel, des champs pour inscrire des renseignements concernant le personnel a ajouté, un champ pour mettre son rôle et un autre champ pour ajouter son spécialité qui est facultatif sauf au docteur.



*Figure 23 : Accueil d’administration et ajout d’un personnel*

En ayant enregistré un nouveau personnel, l’utilisateur sera rediriger vers la liste de tous les personnels. 

*Figure 24 : Liste des personnels*

# D- Discussion

## 1. Difficultés rencontrés

Durant le développement de ce projet, il y avait eu quelques difficultés que ce soit sur le plan matériel ou sur le développement du logiciel.

Sur le plan matériel: durant notre période de stage, il y avait eu un problème sur mon ordinateur. La recherche de NAP de rechange nous a fait perdre beaucoup de temps. En outre, on a décidé d’acheté un autre ordinateur car la recherche de la pièce manquante s’avère introuvable, vue que l’ordinateur que j’avais utilisé était du modèle très ancienne.

Sur le développement: le choix de la technologie a utilisé était un peu difficile, du coup, la demande de conseil auprès des collègues et à l’encadreur professionnel nous a permis de choisir symfony et en plus vue sa force, son efficacité et ses plusieurs documentations, développé le projet avec serait motivant. Mais apprendre la technologie et développé le projet en même temps n’était pas facile et demandait beaucoup de temps.

**Les points d’amélioration:**

Pour le meilleur fonctionnement de cette application, il y a quelque amélioration à ajouter. Voici quelques améliorations dont on doit finir après:

* La mise au point d’une gestion de rendez-vous pour limité les regroupements.
* Créer une page pour les patients à distance.
* Ajouter des autres spécialités des docteurs et généré des formulaires de diagnostic selon les normes.

# CONCLUSION

Dans le début de la présentation de ce thème, on a mentionné quelques problèmes rencontrés au sein des dispensaires et que l’utilisation de l’application web était un des moyens disponible pour combler ses besoins. La question qui a été posé est: « comment enregistrer rapidement les données, avoir une bonne organisation et éviter les pertes de données qui pourraient amener à la mal traitement des patients au sein des dispensaires?»

D’après les résultats obtenues en analysant le fruit du projet, et qui ont déjà été présentées dans la partie réalisation dans cette présentation, les objectifs sont atteints malgré quelques améliorations qu’on devrait encore y faire pour le meilleur fonctionnement de l’application.

Pour conclure, nous pouvons dire que le travail a été bien réalisé en analysant les résultats obtenues lors du développement du projet. Et face à des problèmes qui pourraient subvenir, il serait garanti qu’il aura une maintenance que nous devrions faire pour satisfaire ces besoins et aussi ajouter au plus ample possible les fonctionnalités de l’application.

**WEBIOGRAPHIE**

[I]: Définition d’une application web. https://www.ideematic.com/dictionnaire-digital/application-web/

[II]: Définition: c’est quoi un IDE. https://www.mobizel.com/definition-cest-quoi-un-ide/

[III]: Environnement de développement- Wikipédia https://fr.wikipedia.org/wiki/Environnement\_de\_d%C3%A9veloppement

[IV]: Présentation du débogueur. https://docs.microsoft.com/fr-fr/visualstudio/debugger/debugger-feature-tour?view=vs-2019

[V]: Framework: définition simple et objectif du terme. https://www.1min30.com/dictionnaire-du-web/framework

[VI]: Pourquoi utiliser un Framework PHP https://blog.nicolashachet.com/developpement-php/pourquoi-utiliser-un-framework-php/

[VII]: https://blog.nicolashachet.com/developpement-php/pourquoi-utiliser-un-framework-php/

[VIII]: https://blog.nicolashachet.com/developpement-php/pourquoi-utiliser-un-framework-php/

[IX]: Symfony-Wilipédia. https://fr.wikipedia.org/wiki/Symfony

[X]: Tous ce qu’il faut savoir sur la technologie symfony. https://www.benjaminyeurch.com/blog/developpement-web/tout-savoir-sur-symfony/

[XI]: c’est quoi Bootstrap. https://www.hostinger.fr/tutoriels/cest-quoi-bootstrap/#Crsquoest-quoi-Bootstrap

[XII]: Merise. https://www.techno-science.net/glossaire-definition/Merise-informatique.html

[XIII]: MPD ou modèle physique de données. https://www.base-de-donnees.com/mpd/

[XIV]: Base de données: qu’est-ce que c’est? https://www.lebigdata.fr/base-de-donnees

[XV]: Tutoriel XAMPP. https://www.ionos.fr/digitalguide/serveur/outils/tutoriel-xampp-creer-un-serveur-de-test-local/