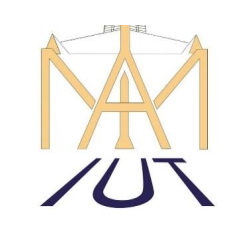
****

République du Cameroun  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
Paix-Travail-Patrie  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
Ministère des enseignements supérieur  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
Université de Ngaoundéré  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Republic of Cameroon  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
Peace-Work-Fatherland  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
Ministry of Higher Education  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
University of Ngaoundéré  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

****

**Encadreur Académique**

M. KANI

Enseignant à L’IUT de Ngaoundéré

**Encadreur Professionnel**

M. TSOBENG TSAFACK HERBIN

CEO MODAFA

Année Universitaire 2021 / 2022

**Mention :** Génie Informatique

**Parcours :** Génie Informatique

**Sujet : Application web pour la génération des attestions et certificats**

***Effectué du 11 Avril au 28 Aout à MODAFA***

Par

EYEBE ELOUNDOU JOEL CERDIC

**Matricule** : 20IO77IU

**INSTITUT UNIVERSITAIRE DE TECHNOLOGIE DE**

**NGAOUNDERE**

**BP : 455 NGAOUNDERE**

***Division de la formation permanente et des relations avec les milieux professionnels***

***TEL : 677512108/677112217/674916057***

***E-mail : Dstages2015@gmail.com***

**MEMOIRE DE FIN D’ETUDE EN VUE DE L’OBTENTION DU DIPLOME UNIVERSITAIRE DE TECHNOLOGIE**

**A tous ceux qui me furent chers**

**et qui ne sont plus**

# Remerciements

Qu’il me soit permis de remercier :

* Le DIEU tout puissant pour m’avoir permis de suivre ce stage sans incident majeur.
* Mes Parents en particulier ma mère pour les sacrifices consentis pour mon éducation.
* Monsieur TSOBENG Herbin CEO de The Modern Application Factory.
* La Direction de l’institut universitaire de technologie de l’université de **NGAOUNDERE** qui a jugé bon de mettre à la disposition de ses étudiants un stage pour peaufiné notre apprentissage et mieux nous familiariser avec le monde entrepreneurial.
* Monsieur KANI Djoude mon encadreur académique.

# *Résumé*

Ce mémoire a été réalisé dans le cadre du stage des agent maitrise ayant pour but de peaufiner notre formation, nous familiariser avec le monde socio-professionnel et préparer dans des conditions idoines notre soutenance qui se déroulera dans un avenir proche. Ce projet a été effectué au sein de l’entreprise The Modern Application Factory (MODAFA).

Le but de notre travail est de concevoir et développer une application web pour la génération des attestions et certificats qui permettra à l’entreprise de gérer ses différents stagiaires, tout en leur permettant de pouvoir imprimer leurs attestations de stages, ailleurs cette application pourra être une source d’informations. La réalisation de cette solution a été effectuée en utilisant plusieurs outils et technologies parmi lesquels : Visual Studio Code, Win Design, MySQL Workbench, les langages de programmations à savoir PHP, JavaScript, le langage de modélisation UML pour la conception et le Framework Symfony.

# *Abstract*

This dissertation was produced as part of the master's agent internship with the aim of refining our training, familiarizing ourselves with the socio-professional world and preparing in suitable conditions for our defense which will take place shortly. This project was carried out within the company The Modern Application Factory (MODAFA).

The purpose of our work is to design and develop a web application for the generation of certificates and certificates which will allow the company to manage its various trainees and has the latter to be able to print their training certificates, moreover this application can be a source of information. The realization of this solution was carried out using several tools and technologies among which we quote: Visual Studio Code, Win Design, MySQL Workbench, the programming languages ​​​​namely PHP, JavaScript, the modeling language UML for the design and the Framework Symphony.

**TABLE DES MATIERES**

[Remerciements 2](#_Toc107555839)

[*Résumé* 3](#_Toc107555840)

[*Abstract* 4](#_Toc107555841)

[INTRODUCTION 7](file:///C:\Users\HP\OneDrive\Documents\Stage\Rapports\Memoire.docx#_Toc107555842)

[Introduction 8](#_Toc107555843)

[Première partie : Présentation du cadre du projet 10](#_Toc107555844)

[I. Présentation de l’entreprise 10](#_Toc107555845)

[1. Historique 10](#_Toc107555846)

[2. Services 10](#_Toc107555847)

[3. Les réalisations de MODAFA 11](#_Toc107555848)

[4. Fonctionnement et organigramme 11](#_Toc107555849)

[II. Plan de localisation 13](#_Toc107555850)

[Deuxieme partie : Specification des besoins 15](#_Toc107555851)

[I. Contexte et objectifs 15](#_Toc107555852)

[1. Objectifs 15](#_Toc107555853)

[2. Planning prévisionnel 15](#_Toc107555854)

[II. Les besoins fonctionnels 15](#_Toc107555855)

[1. Les besoins propres à l’utilisateur. 16](#_Toc107555856)

[2. Les besoins de l’administrateur 16](#_Toc107555857)

[III. Les besoins non fonctionnels 16](#_Toc107555858)

[IV. Diagramme de cas d’utilisation(Besoins) 16](#_Toc107555859)

[1. Définition 16](#_Toc107555860)

[2. Les acteurs 17](#_Toc107555861)

[3. Cas d’utilisation de l’internaute 17](#_Toc107555862)

[4. Cas d’utilisation de l’utilisateur 18](#_Toc107555863)

[5. Cas d’utilisation de l’administrateur 19](#_Toc107555864)

[V. Raffinement des cas d’utilisations (Use Case) 20](#_Toc107555865)

[1. Use Case ‘‘manage profile ’’ et ‘‘manage template’’ (par user) 20](#_Toc107555866)

[2. Description textuelle de quelques cas d’utilisations. 21](#_Toc107555867)

[3. Quelques diagramme de séquences 25](#_Toc107555868)

[Troisieme partie : La conception 28](#_Toc107555869)

[I. Définition 28](#_Toc107555870)

[II. Diagramme De Classes 28](#_Toc107555871)

[1. Les règles de gestion 28](#_Toc107555872)

[2. Diagramme de Classes 29](#_Toc107555873)

[III. Diagramme de séquences 30](#_Toc107555874)

[1. Définition 30](#_Toc107555875)

[2. Diagramme de séquences 31](#_Toc107555876)

[Quatrième partie : Réalisation 34](#_Toc107555877)

[I. Environnement de travail 34](#_Toc107555878)

[1. Environnement matériel 34](#_Toc107555879)

[2. Environnement logiciel 34](#_Toc107555880)

[3. Architecture de développement : Architecture MVC (Model-View-Controller) 36](#_Toc107555881)

[4. Langages de programmations et technologies 36](#_Toc107555882)

[II. Présentation des interfaces. 38](#_Toc107555883)

[Conclusion 43](#_Toc107555884)

[Liste de figures 44](#_Toc107555885)

[Liste de tableaux 44](#_Toc107555886)

# INTRODUCTION

# Introduction

Dans le cadre de la formation de chaque étudiant, les cours théoriques sont complétés par l’analyse des cas pratiques, des visites d’entreprises et surtout des stages en entreprises qui permettent aux apprenants d’affronter les réalités concrètes et de mieux se préparer à l’éventualité future d’une insertion socio-professionnelle réussite. C’est dans cette optique que, pendant ma formation à l’IUT de NGaoundéré en Génie Informatique j’ai côtoyé le monde professionnel durant une période 10 semaines au travers de l’entreprise The Modern Application Factory. The Modern Application Factory est une entreprise qui fait dans les offres de services informatiques au vue de la place stratégique qu’elle occupe dans notre environnement, elle est amenée à accueillir de nombreux stagiaires et à travailler avec de nombreux employés. Au vue de cette état des choses, il pourrait se pose un souci de gestion efficace de ses personnes très souvent sur le départ. Donc pourquoi ne pas avoir une application dédie à cela qui permettra de réaliser toute ces taches de façons optimale et d’avoir une meilleur vue d’ensemble. Notre projet consiste à concevoir et développer une application web de génération des attestions et certificats pour The modern Application Factory. Pour ma part ce mémoire va s’architecturer sur quatre pilier : La présentation du cadre du projet, où on va présenter The Modern Application Factory; La spécifications des besoins ; La conception détaillée, avec les différents diagrammes UML réalisés ; La réalisation, où on présentera tous les outils matériels et logiciels utilisés pour le développement de la solution.

PREMIERE PARTIE :

*PRESENTATION DU*

*CADRE DU PROJET*

# Première partie : Présentation du cadre du projet

L’entreprise est une entité économique qui a pour objectif la production des biens et services. Quelle que soit le domaine d’activité de cette dernière, elle a pour objectif principale la croissance et le développement. Cependant, compte tenu de son environnement et de son activité, elle est structurer d’une manière spécifique qui lui est propre. Dans le cadre de l’entreprise qui nous intéresses c’est-à-dire MODAFA : entreprise de service numérique, au sein de laquelle nous avons eu l’opportunité d’effectuer notre stage et où nous avons chaleureusement été accueillis, nous nous efforcerons de faire une brève et claire présentation de celle-ci.

## Présentation de l’entreprise

Ici, nous allons retracer la genèse de l’entreprise, les services, les réalisations et son organigramme.

### Historique

C’est le 24 octobre 2019 que le projet de création de cette agence web vois le jour. C’est à l’initiative d’un jeune ingénieur informaticien ayant le souci de répondre au challenge de la digitalisation des services en Afrique et au Cameroun en particulier. MODAFA (The Modern Application Factory), se positionne comme une entreprise de services numérique offrant des prestations de haute qualité. Comme son nom l’indique, MODAFA est une usine d’application moderne spécialisée dans le développement logiciel offshore (la notion d’offshore renvoyait alors à l’externalisation de l’activité en dehors du territoire), onshore (sur site) ou nearshore (hors des frontières). Elle offre un large panel de solutions digitales très intéressantes pour toute entreprise ou organisation qui souhaite se démarquer positivement de l’ère du tout numérique.

MODAFA est située au quartier Mvog-Ada (au lieu-dit Fanta Citron), à Yaoundé. Elle travaille avec des startups, des sociétés de développement logiciel, des agences numériques et des entreprises dans le but de simplifier leur expérience d’externalisation informatique et de réduire le rapport couts délais de commercialisation.

### Services

L’entreprise MODAFA offre une multitude de services varier parmi lesquelles.

* Le développement d’application Web/Mobile : Elle fournit des services de développement de logiciels de haute qualité pour le Web et le mobile, en tirant parti des meilleures capacités de la plate-forme et en offrant une expérience utilisateur haut de gamme.
* La maintenance des applications tierces : Maintenance d’applications tierces (TMA) – Que votre application informatique ait été créée et mise en œuvre par notre société ou non, les équipes de MODAFA sont en mesure de vous fournir leurs services de maintenance, en assumant des responsabilités professionnelles liées aux mesures correctives et de développement nécessaires.
* Le conseil informatique : MODAFA met à disposition de ses clients une équipe de consultants expérimentés pour vous aider à définir clairement vos objectifs et les résultats souhaités, à développer une approche personnalisée de vos besoins, en équilibrant les meilleures pratiques avec vos exigences de rapidité, l’efficacité et la flexibilité.
* Modernisation des applications : Transforme les applications héritées en technologies de pointe et en architecture moderne pour rester pertinentes et offrir la plus grande valeur commerciale.
* Services de révision de code : Le code des produits existants des clients est soigneusement vérifié pour s’assurer qu’il respecte les meilleures pratiques en termes de fiabilité du code, de maintenabilité, d’efficacité et de sécurité.
* Les équipes dédiées : En tant que société d’externalisation informatique, MODAFA fournit des équipes de développement de logiciels dédiées de manière fiable qui ont une vaste expérience dans les dernières technologies de développement. Dans le cadre d’un personnel informatique dédié, Le chef de projet travaille directement avec le développeur de logiciels. Très rapidement, le développeur de logiciels dédié fait ainsi partie de l’équipe locale du client.

### Les réalisations de MODAFA

MODAFA en si peu de temps à réaliser une multitude de projet dans des domaines varier parmi lesquelles nous pouvons citer :

* AGRI IO Web and Mobile App : qui est un projet réaliser avec les technologies PHP, MySQL, Symfony, flutter et les API REST pour une structure agricole Allemande ayant des filiales en côte d’Ivoire et au Burkina Faso.
* Odoo ERP
* EFI Services
* EK lagacy
* Zmart entretien groupe qui est projet pour une entreprise d’entretien.
* Africa Market Insights
* Ivoire Organics est une solution Intranet avec la solution Microsoft SharePoint.
* HOMECM : qui est une plateforme modern conçue pour servir les professionnels de l’immobilier et les particuliers désirant louer, acheter ou vendre des biens immobiliers.

### Fonctionnement et organigramme

#### Fonctionnement

Pour mieux présenter l’organigramme de MODAFA, nous allons énumérer les différentes fonctions de responsabilités existantes au sein de cette entreprise.

* CEO (Chief executive officer) il est chargé de définir les orientations et de l’entreprise et de les mettre sur pied. Il s’occupe aussi de la ressource humaine et des différentes stratégies.
* CTO (Chief Operating Officer) Son rôle consiste à déterminer des innovations pertinentes. Celles qui seront déployées pourront concerner n’importe quel service de l’entreprise. En charge d’étudier de nouvelles opportunités technologiques, le CTO participe au déploiement de celles-ci et travaille au sein du groupe.
* CFO (Chief Financial Officer) il a pour mission de garantir un parfait équilibre financier, administratif et juridique, comprenant parfois les RH, les achats et l’office management, fonction extrêmement importante en entreprise. Il accompagne l’entreprise dans le processus de lever des fonds.

#### Organigramme

Le schéma ci-dessous représente l’organigramme hiérarchique de **MODAFA.**

Requirements elicitation, analysist, specification, validation, design

Development Manager

Employee Name

Assistant project foreman

Employee Name

Quality Assurance Manager

Employee Name

Leads Projects, keeps timeframes on target, assist others in probleming solving, software construction, unit testing

Business Analyst Assistant

Employee Name

Requirements elicitation, analysist, specification, validation, design

Supervisor, analyst, Review Master, team leader, method & tools

Hiring, Marketing, Strategy, Evaluation

Administration, Accounting, budget

Business Analyst Manager

Employee Name

CFO

Employee Name

CTO

Employee Name

**CEO**

TSOBENG TSAFACK HERBIN

**Figure 1 : Organigramme de MODAFA**

Project C Team Member

Employee Name

Project C Quality Assurance Manager

Employee Name

Project C Analyst Assistant

Employee Name

Project B Quality Assurance Manager

Employee Name

Project B Team Member

Employee Name

Project B Analyst Assistant

Employee Name

Project A Analyst Assistant

Employee Name

Project A Team Member

Employee Name

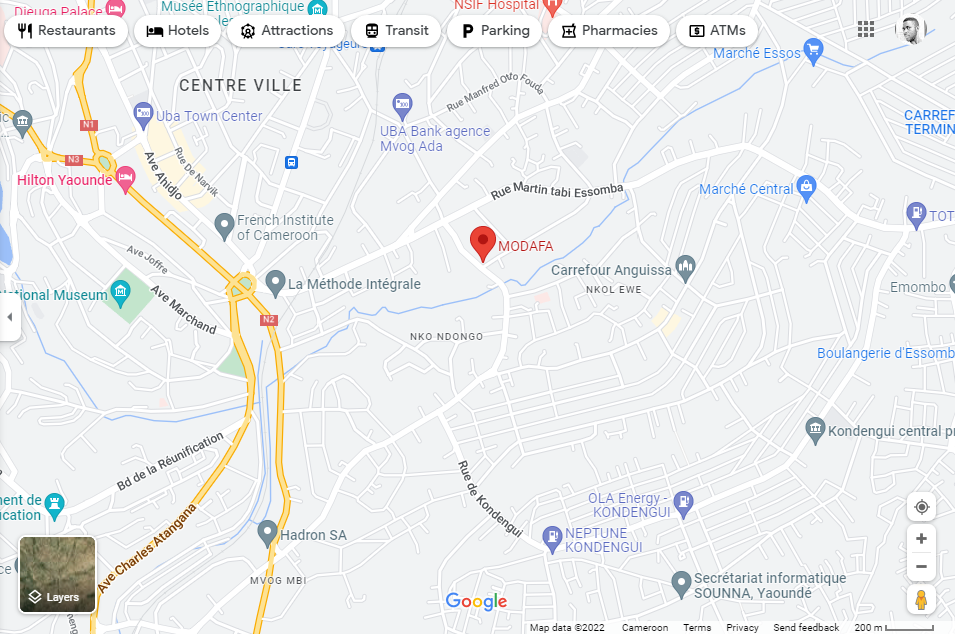
Assistant Quality Assurance Manager

Employee Name

Project A Quality Assurance Manager

Employee Name

## Plan de localisation

Le schéma ci-dessous illustre le plan de localisation de MODAFA.

**Figure 2 : Plan de localisation**

La présentation du cadre du projet nous a permis de présenter la genèse de l’entreprise, les services qu’elle offre, certaines réalisations, son fonctionnent et son plan de localisation. Tout ceci nous a permis de planter le décor. Dans la deuxième partie nous mettrons en sur pied le document des spécifications.

DEUXIEME PARTIE :

*DOCUMENT DES SPECIFICATIONS*

# Deuxieme partie : Specification des besoins

Cette application AWGAC « Application web pour la génération des attestions et certificats » qui a pour cible principale MODAFA et éventuellement toutes les structures ayant un besoin similaire. Car cette solution est une alternative à l’archivage. Dans cette partie, on va présenter les besoins fonctionnels et non fonctionnels de la solution.

## Contexte et objectifs

### Objectifs

Ce projet est suscité par le fait que MODAFA désirait une application qui lui permettrait de gérer de génère des certificats et attestations de stage. Pour pouvoir solutionner ce problème, cette thématique nous a été attribuer comme sujet de réflexion durant le stage. Pour mener à bien ce travail nous devrons concevoir de bout en bout cette application. Nous envisageons donc avoir à la fin ce travail des livrables comme :

* Le document d’analyse
* Le document de conception

### Planning prévisionnel

Le diagramme ci-dessous décrit le planning prévisionnel à suivre pour la réalisation du projet :

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Avril 2022 | | | | | | Mai 2022 | | | | | | | | | Juin 2022 | | | | | | |
| 11 | 15 | 18 | 22 | 25 | 29 | 2 | 6 | 9 | 13 | 16 | 20 | 23 | 27 | 30 | 3 | 6 | 10 | 13 | 17 | 20 | 24 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Insertion |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Etude préalable |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Conception |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Codage |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tests |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Rapport |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Figure 3 : Diagramme de Grantt**

## Les besoins fonctionnels

Les besoins sont fonction des profils dans notre cas nous les avons regroupés en deux profils qui sont : L’utilisateur (USER) et l’administrateur (ADMIN).

### Les besoins propres à l’utilisateur.

Les besoins fonctionnels de l’utilisateur sont :

* L’administration des modèles : Cette tâche consiste à l’ajout d’un nouveau modèles, la visualisation des modèles existants et la suppression des modèles qui serait devenu indésirable.
* L’administration des stagiaires : Cette tâche consiste à créer un stagiaire au besoin, modifier les informations d’un stagiaire en cas d’informations erroné, afficher le détail des informations d’un stagiaire et le supprimer pour une raison ou une autre.
* Administrer les attestions de stage : Cette tâche consiste à créer une attestation au besoin, le visualiser, l’imprimer et même le supprimer au besoin.
* Vérifier l’authenticité d’une attestation : Cette opération consiste à rechercher une attestation à partir d’un code unique.

### Les besoins de l’administrateur

Les besoins fonctionnels de l’administrateur sont :

* La gestion des utilisateurs : Cette tâche consiste créer un utilisateur, consulter son profil et l’historique d’activité, le supprimer si on éprouve le besoin.
* La gestion des certificats : Cette tâche consiste à créer un certificat de travail pour un employé, visualiser le certificat et le supprimer pour une raison quelconque.

## Les besoins non fonctionnels

Notre système doit répondre aux critères de qualités suivants :

* Securité : Les comptes d’utilisateurs doivent être protéger donc une connexion avec email et mot de passe est primordiale. La visibilitéé des contenus fonction des rôles doit aussi être pris en compte.
* L’intégrité : L’intégrité et la cohérence des données doivent être assurées à chaque mise à jour et insertion dans la base de données.
* La rapidité : Rapidité d’exécution des traitements (code optimisé).
* L’ergonomie : Les interfaces doivent être conviviales et ergonomiques. Elles doivent être simples, lisibles, confortables à l’œil et faciles à utiliser.
* La maintenabilité : Le code doit être compréhensible afin d’assurer son évolution et son extensibilité pour répondre aux besoins sans cesse croissant des utilisateurs.

## Diagramme de cas d’utilisation(Besoins)

### Définition

Un diagramme de cas d'utilisation définit le comportement d'un système, comme vu par un utilisateur extérieur. Il décompose la fonctionnalité du système en unités cohérentes, à savoir les cas d'utilisation, ayant un sens pour les acteurs. Ceux-ci permettent d’exprimer les besoins des utilisateurs d'un système

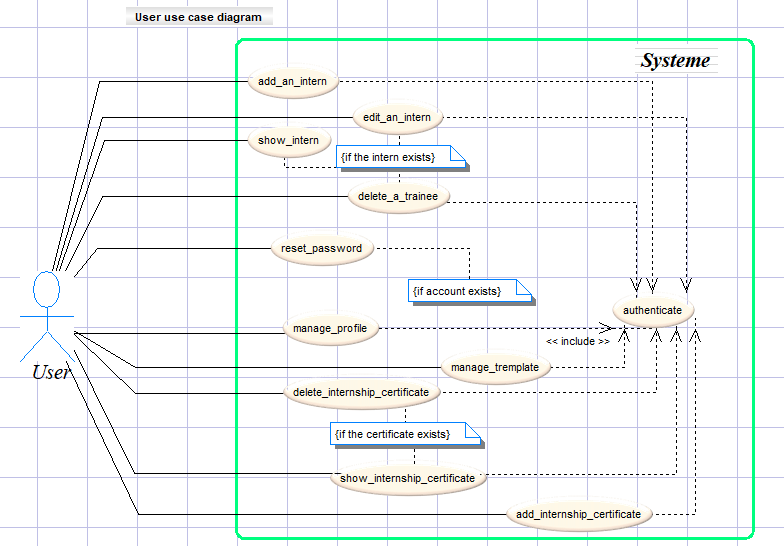
### Les acteurs

* L’internaute : il peut vérifier la véracité de l’authenticité d’une attestation.
* L’utilisateur / encadreur : il s’occupe de la gestion des stagiaires, des attestions de stage.
* L’administrateur : C’est celui qui gère le système, il s’occupe de la gestion des utilisateur / encadreurs et peut aussi réalise aussi toutes les opérations propres à l’utilisateur.

### Cas d’utilisation de l’internaute

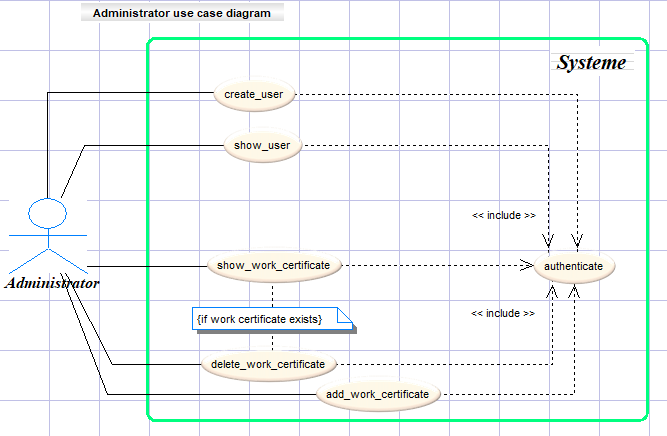
**Figure 4** **:** **Diagramme de cas d’utilisation de L’internaute.**

### Cas d’utilisation de l’utilisateur

Les tâches citées plus haut sont présentés par le diagramme de cas d’utilisation de l’utilisateur suivant.

**Figure 5** **: Diagramme de cas d’utilisation de l’utilisateur.**

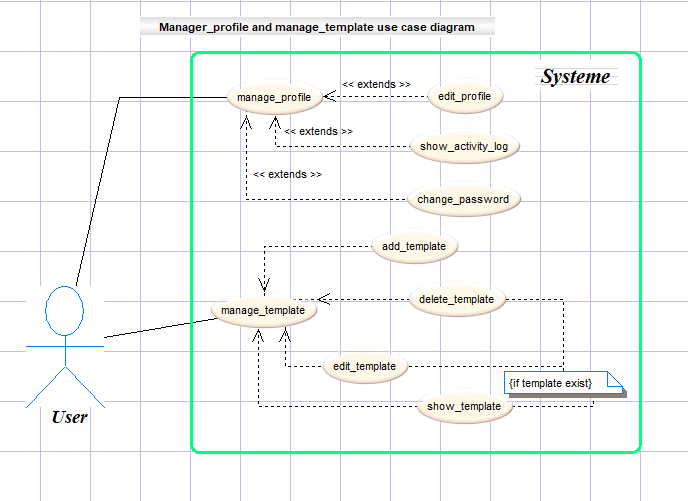
### Cas d’utilisation de l’administrateur

Les tâches citées plus haut sont présentés par le diagramme de cas d’utilisation de l’administrateur suivant.

**Figure 6 : Diagramme de cas d’utilisation de l’admin**

## Raffinement des cas d’utilisations (Use Case)

### Use Case ‘‘manage profile ’’ et ‘‘manage template’’ (par user)

Le diagramme suivant est le raffinement de l’UC ‘‘manage profile’’ et ‘‘manage template’’

**Figure 7 : Raffinement des use case.**

### Description textuelle de quelques cas d’utilisations.

Pour une meilleure compréhension de certain cas d’utilisation lister plus haut, une description textuelle de ceux-ci est nécessaire.

**SOMMAIRE D’IDENTIFICATION**

**Titre** : Authenticate

**Résumé** : un utilisateur souhaite se connecter. Il entre ses informations (email et mot de passe), valide et le système effectue une vérification. Au le cas où les informations fournit seraient correctes l’utilisateur sera admis sur l’application, au cas contraire un message d’erreur lui sera renvoyer.

**Acteurs :** user, administrator

**Description des scénarios**

**Préconditions**

La page de connexion doit être accessible.

**Scénario nominal**

|  |  |
| --- | --- |
| 1. Ce cas d’utilisation commence quand l’utilisateur, l’administrateur se rend sur la page de connexion du site. |  |
| 1. Il entre ses identifiants |  |
| 1. Il clique sur le bouton ‘‘Sign in’’ pour soumettre ses informations. | 1. Le système vérifie puis le redirige a la page correspondante à son role. |

Tableau 1 : Scénario nominal du cas d’utilisation ‘‘authenticate’’

**Enchainement alternatif**

**A1** : identifiants incorrectes.

L’enchainement A1 démarre au point 4 du scénario nominal.

* 1. Le système indique à l’utilisateur que ‘‘Invalid credentials.’’.

Le scénario nominal reprend au point 2.

**Post condition**

L’utilisateur est connecté et sera redirigé vers le listing des stagiaires.

**SOMMAIRE D’IDENTIFICATION**

**Titre** : Add an intern

**Résumé** : un utilisateur ou l’administrateur souhaite enregistrer un nouveau stagiaire. Il saisit les informations du stagiaire (nom, prénom, sexe, date de naissance…), valide et le système effectue les vérifications. Si les informations sont correctes il sera redirigé vers le listing des stagiaires et un message de succès lui sera notifier ; au cas contraire un message d’erreur lui sera renvoyer.

**Acteurs :** user, administrator

**Description des scénarios**

**Préconditions**

L’utilisateur doit être connecter.

**Scénario nominal**

|  |  |
| --- | --- |
| 1. Ce cas d’utilisation commence quand l’utilisateur ou l’administrateur souhaite enregistrer un nouveau stagiaire. |  |
| 1. Il entre les informations du stagiaire |  |
| 1. Il clique sur le bouton ‘‘Save’’ pour soumettre ses informations. | 1. Le système vérifie la conformité des informations et le redirige au listing des stagiaires. |

Tableau 2 : Scénario nominal du cas d’utilisation ‘‘add an intern’’

**Enchainement alternatif**

**A1** : Les dates ne seraient pas cohérentes (il est âgé de moins de 15 ans, les date de début et de fin ne respecteraient un ordre chronologique).

L’enchainement A1 démarre au point 4 du scénario nominal.

* 1. Le système indique à l’utilisateur que ‘‘The dates are inconsistent’’.

Le scénario nominal reprend au point 2.

**Post condition**

**L**e stagiaire est ajouté en base de donnes et l’utilisateur sera redirigé sur la vers la page de listing des stagiaires.

**SOMMAIRE D’IDENTIFICATION**

**Titre** : Reset password

**Résumé** : Un utilisateur ou l’administrateur aurait oublier son mot de passe et veux le réinitialiser. Il clique sur ‘‘Password forget ?’’ puis il est invité à renseigner son adresse email ; ensuite un email de vérification lui sera envoyer par le system à partir de celui-ci il pourra réinitialiser son mot de passe et en entrer un nouveau.

**Acteurs :** user, administrator

**Description des scénarios**

**Préconditions**

L’utilisateur doit avoir un compte et avoir oublier son mot de passe.

**Scénario nominal**

|  |  |
| --- | --- |
| 1. Ce cas d’utilisation commence quand l’utilisateur ou l’administrateur souhaite réinitialiser son mot de passe. |  |
| 1. Il clique sur le lien ‘‘Password forget ?’’ pour enclencher le processus. | 1. Le système lui demande de saisir son adresse email. |
| 1. Il entre son adresse email et la soumet. | 1. Le système vérifie si cette adresse email a un compte et lui envoie un email de réinitialisation de mot de passe. |
| 1. Il clique sur le bouton réinitialisé qui se trouve dans le mail pour confirmer la de demande de modification du mot de passe. | 1. Le système vérifie la validité du jeton de réinitialisation et lui envoie un formulaire de réinitialisation. |
| 1. Il entre son nouveau mot de passe clique sur le bouton ‘‘Register ’’. | 1. Le système vérifie si le mot de passe est conforme et est différent du précédent puis lui envoi la notification ‘‘Password has ben change as successfuly.’’. Il est par la suite rediriger sur la page de connexion |

Tableau 3 : Scénario nominal du cas d’utilisation ‘‘Reset password’’

**Enchainement alternatif**

**A1** : L’adresse email saisi pour la récupération n’aurait pas de compte.

L’enchainement A1 démarre au point 5 du scénario nominal.

* 1. Le système indique à l’utilisateur que ‘‘There was a problem.’’.

Le scénario nominal reprend au point 4.

**A2** : Le nouveau mot de passe n’est pas conforme ou est identique au précèdent.

L’enchainement A2 démarre au point 9 du scénario nominal.

* 1. Le système indique à l’utilisateur qu’une erreur s’est produite.

Le scénario nominal reprend au point 8.

**Enchainement d’erreur**

**E1** : Le jeton à expirer ou est invalide.

L’enchainement E1 démarre au point 7 du scénario nominal.

* 1. Le système indique à l’utilisateur qu’une erreur s’est produite.

Le cas d’utilisation se termine en échec.

**Préconditions**

Le mot de passe est modifier.

**SOMMAIRE D’IDENTIFICATION**

**Titre** : create\_user

**Résumé** : L’administrateur souhaite crée un nouvel utilisateur.

**Acteurs :** administrator

**Description des scénarios**

**Préconditions**

L’administrateur doit être connecter.

**Scénario nominal**

|  |  |
| --- | --- |
| 1. Ce cas d’utilisation commence quand l’administrateur décide de crée un nouvel utilisateur. |  |
| 1. Il entre les identifiants du nouvel utilisateur(le nom et son email). |  |
| 1. Il clique sur le bouton ‘‘Register’’ pour soumettre les informations. | 1. Le système vérifie les informations ensuite envoie un email à l’utilisateur avec son mot de passe ; et l’administrateur reçoit à son tour un message de succès. |

Tableau 4 : Scénario nominal du cas d’utilisation ‘‘create user’’

**Enchainement alternatif**

**A1** : L’adresse email est déjà utiliser par un compte.

L’enchainement A1 démarre au point 4 du scénario nominal.

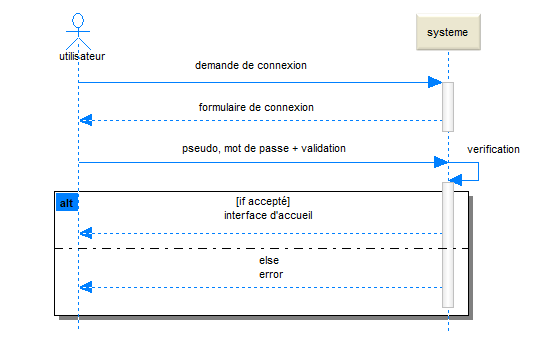
* 1. Le système indique à l’administrateur que ‘‘There is already an account with this email.’’.

Le scénario nominal reprend au point 2.

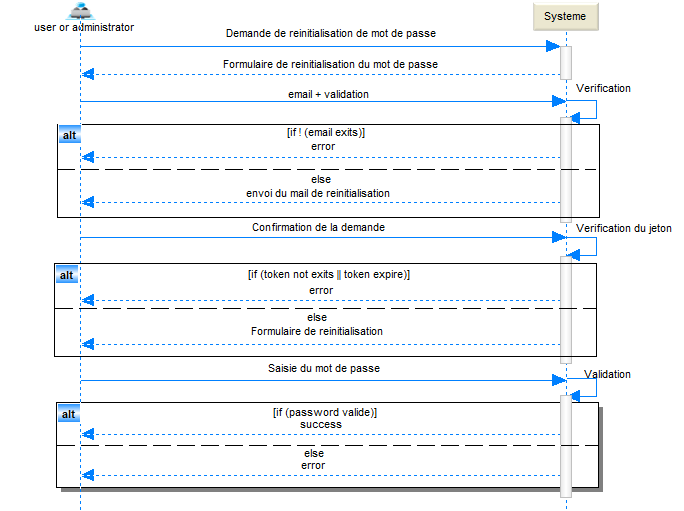
**Post condition**

L’utilisateur est ajouté en base de donne et administrateur reçoit un message de succès.

### Quelques diagramme de séquences

* Le digramme de séquence du cas d’utilisation ‘‘authenticate’’

**Figure 8 : Diagramme de séquence ‘‘authenticate’’**

* Le diagramme de séquence du cas d’utilisation ‘‘reset\_password’’

**Figure 9 : Diagramme de séquence ‘‘reset\_password’’**

Dans cette partie, on a bien spécifié tous les besoins fonctionnels et non fonctionnels de notre application, et on a présenté également les diagrammes des cas d’utilisation ainsi que leurs raffinements. Dans la partie suivant on va entamer l’étape suivante du projet, à savoir la conception détaillée, en présentant les diagrammes UML nécessaires à la phase du développement.

TROISIEME PARTIE :

*DOCUMENT DE CONCEPTION*

# Troisieme partie : La conception

La conception permet de formaliser les étapes préliminaires du développement afin de le rendre plus fidèle aux besoins. L’étape d'analyse permet d’identifier les résultats attendus, en termes de fonctionnalités tandis que l’étape de conception décrit, en utilisant un langage de modélisation, le fonctionnement du système, dans le but d'en faciliter le développement. Dans ce chapitre nous présentons les différents diagrammes UML (Unified Modeling Language) créés en utilisant l’outil de conception Win Design.

## Définition

UML est un langage graphique permettant de représenter et de communiquer les différents aspects d'un système d'information, au lieu de définir une méthode. Des textes sont associés aux schémas graphiques expliquant leur contenu. Il est donc un métalangage, car il fournit les éléments permettant de construire le modèle qui sera le langage du projet.

## Diagramme De Classes

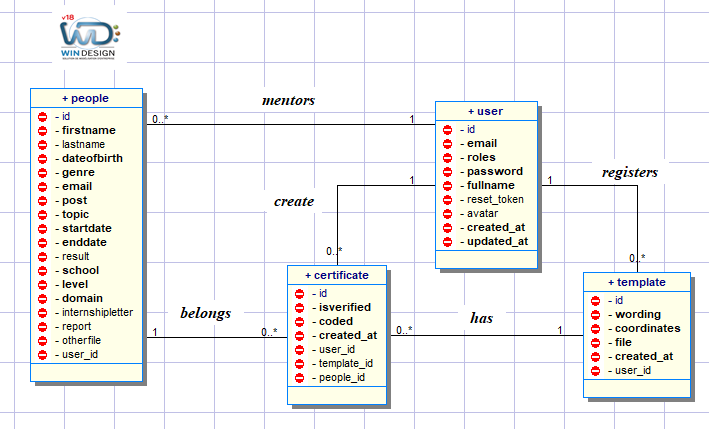
Lors de la modélisation orientée objet, ce diagramme est obligatoire et le plus important. Il montre la structure interne du système en représentant d'une façon abstraite les objets qui vont interagir pour réaliser les cas d'utilisation.

Il représente une vue statique, où le facteur temporel n'est pas pris en compte dans le comportement du système. Ce diagramme permet de modéliser les classes et les relations du système.

### Les règles de gestion

* Un internaute pourra faire une vérification de certificat ou d’attestation.
* Un utilisateur pourra gérer les stagiaires (créer, mettre à jour, supprimer…) ; gérer les attestations (créer, supprimer, avoir un aperçue) ; gérer les modèles (créer, mettre à jour, supprimer…) ; effectuer une vérification de certificat.
* L’administrateur peut réaliser toutes les taches propres de l’utilisateur et de l’internaute.
* L’administrateur enregistre un utilisateur / encadreur
* L’administrateur s’occupe de la gestion des certificats
* Pour créer un certificat il faut que celui-ci appartienne à un utilisateur et fait à partir d’un modèle. Ceci est également valable pour les attestations.

### Diagramme de Classes



**Figure 10** **: Diagramme de classes**

Dans ce diagramme, on trouve les différentes classes et les relations qui représentent la structure de la base de données :

* User : est la classe qui représente un utilisateur, elle contient toutes ces informations personnelles.
* People : est la classe qui représente un stagiaire, elle contient toutes les informations nécessaires à la gestion des stagiaires et une clé étrangère provenant (user\_id) de la table User qui représente dans ce cas le lien entre l’encadreur et le stagiaire.
* Template : est la classe qui représente les modèles de certificat et attestations. Cette table en plus de ses attributs contient aussi une clé étrangère (user\_id) provenant de la table User qui représente l’utilisateur qui à enregistrer le modèle.
* Certificate : est la classe qui représente les certificats et attestations. Cette table contient les informations sur un certificat ou une attestation elle porte trois clés étrangères des trois classes avec lesquelles elle est en relation. user\_id provient de la table User qui représente l’utilisateur qui a créer le certificat ; template\_id qui provient de la table Template qui représente le modèle choisit pour le certificat ; people\_id qui provient de la table People qui est le stagiaire titulaire du certificat.

## Diagramme de séquences

### Définition

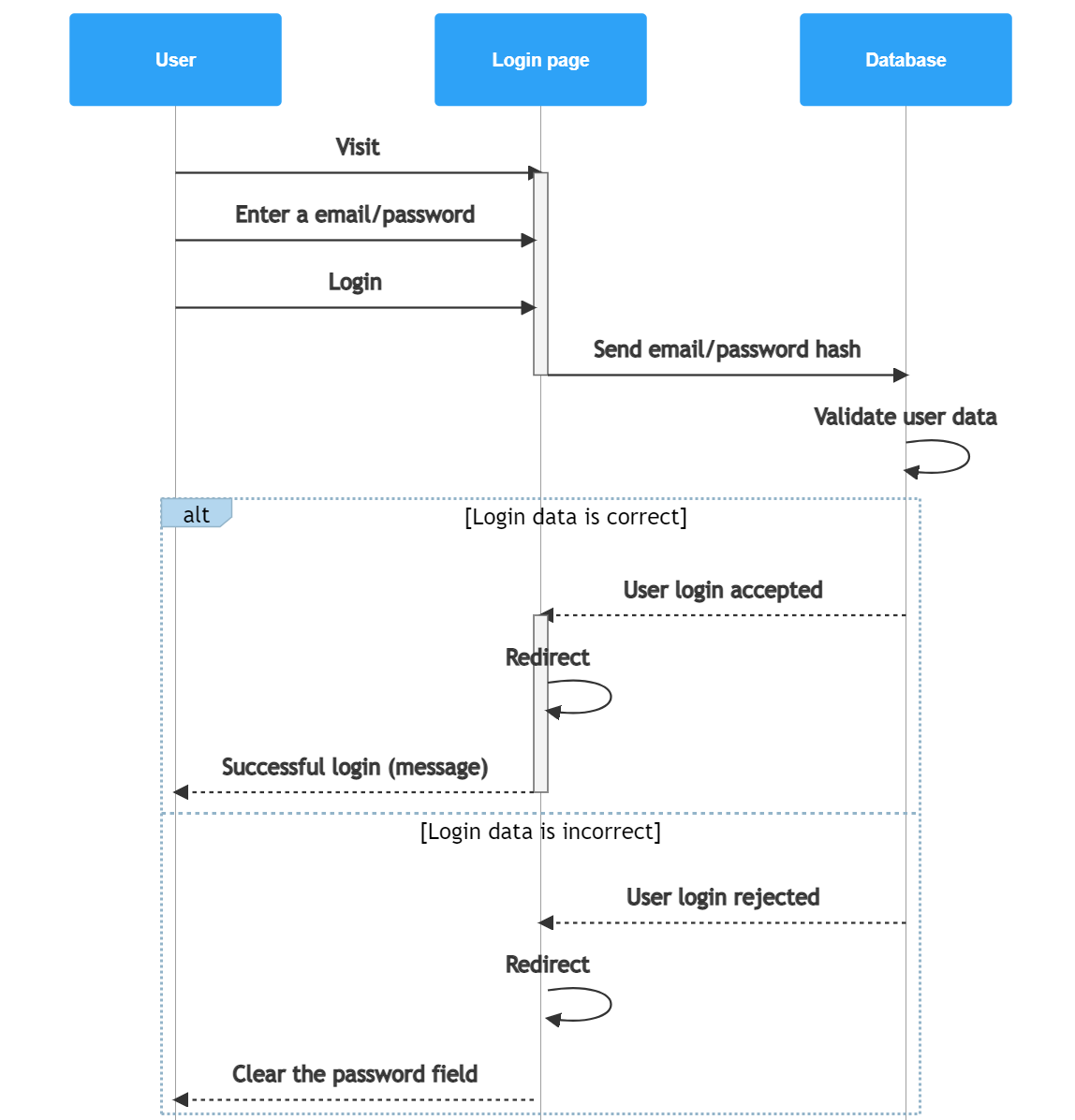
« C’est un diagramme temporel qui montre les échanges séquentiels de messages et le passage du temps. C’est un diagramme temporel. Les principaux concepts sont les objets participants à la séquence, le temps, les messages. »

On a présenté dans la partie précédent quelques diagrammes de séquences globaux montrant l’interaction entre l’utilisateur et le système. Les diagrammes qu’on va présenter montrent en détails l’interaction entre les différentes composantes du système, en partant de l’utilisateur, jusqu’au contrôleur. Cette interaction est représentée par des flèches portant des messages descriptifs, on trouve par exemple des noms de méthodes appelés avec ses paramètres, des messages d’erreurs ou validation, des messages décrivant la tâche réalisée comme par exemple vérification, ou remplissage du formulaire, ou bien des messages de navigation permettant la redirection vers d’autres vues ou composants graphiques de l’application.

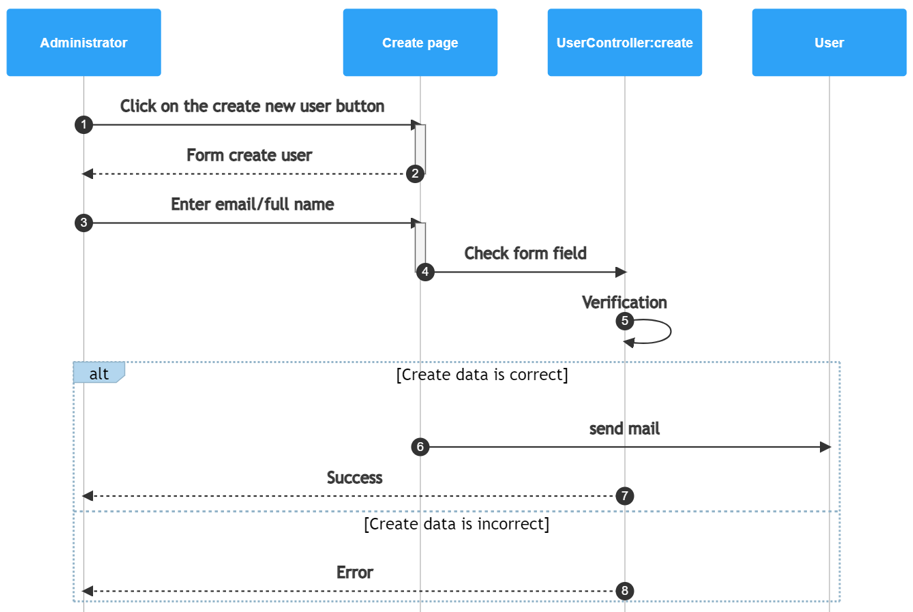
### Diagramme de séquences

On va représenter les diagrammes de séquence détaillés de l’authentification et la création d’un utilisateur.

* Diagramme de séquence de “authenticate”



**Figure 11 : Diagramme de séquence de ‘‘authenticate’’**

* Diagramme de séquence de ‘‘create\_user’’

**Figure 12 : Diagramme de séquence de ‘‘create\_user’’**

Dans cette partie on a présenté les diagrammes UML nécessaires à la réalisation de notre solution, ce qui nous permet de commencer le développement en ayant une idée claire et détaillée sur l’aspect statique et dynamique de la solution. Il reste à coder ce qu’on a conçu. Le chapitre suivant présentera l’environnement de travail à savoir les outils et les méthodes utilisés lors du développement.

QUATRIEME PARTIE :

*REALISATION*

# Quatrième partie : Réalisation

Pour mener à bien notre projet, il est nécessaire de choisir les outils et technologies adéquats qui permettent de simplifier le développement. Dans ce chapitre, on présentera l’environnement matériels et logiciels et à la fin l’état de réalisation.

## Environnement de travail

Dans ce qui suit, nous présenterons les différents outils matériels et logiciels utilisés pour la réalisation de notre solution.

### Environnement matériel

Pour le développement de notre solution, on a utilisé un ordinateur portable de marque HP ayant les caractéristiques suivantes :

|  |  |
| --- | --- |
| *ELEMENTS* | *CARACTERISTIQUES* |
| Processeur | Intel(R) Core(TM) i5-6300U CPU@  2.40 GHz, 2.496 MHz, Core(s), 4 Logical Processor(s) |
| RAM | 8 Go |
| Disque Dur | 500 Go |
| Type du système | 64 bits |
| Système | Windows 10 |

### Environnement logiciel

Les différents logiciels et technologies utilisés pour le développement de notre application sont :

* Visual studio code



**Figure 13 : Logo Visual studio code**

Visual Studio Code est un éditeur de code source léger mais puissant qui s’exécute sur votre bureau et est disponible pour Windows, macOS et Linux. Il est livré avec un support intégré pour JavaScript, TypeScript et Node.js et dispose d’un riche écosystème d’extensions pour d’autres langages et runtimes (tels que C ++, C #, Java, Python, PHP, Go, .NET).

* Wampserver

**Figure 14 : Logo wampserver**

WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l’aide du serveur Apache2, du langage de scripts PHP et d’une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement vos bases de données.

* Win Design



**Figure 15 : Logo win Design**

Dédié aux entreprises, WinDesign est un environnement permettant la modélisation de système d'information informatiques et organisationnels. Il est composé de trois modules complémentaires Database, Object et Business Process afin de couvrir tous vos besoins, et ce, que vous soyez administrateur de bases de données, chef de projet ou développeur.

Le premier module, Database, offre la possibilité de concevoir, générer et documenter les différentes bases de données.

Le module Object permet quant à lui de spécifier les points de vue dynamiques et statiques avec l'appui de diagrammes et de langages suivant les référentiels UML.

Enfin, le module Businness Process donne les moyens de réaliser des cartographies métier, applicative et fonctionnelle. Ceci peut s'avérer pratique pour mesurer les impacts d'un changement dans votre système d'information.

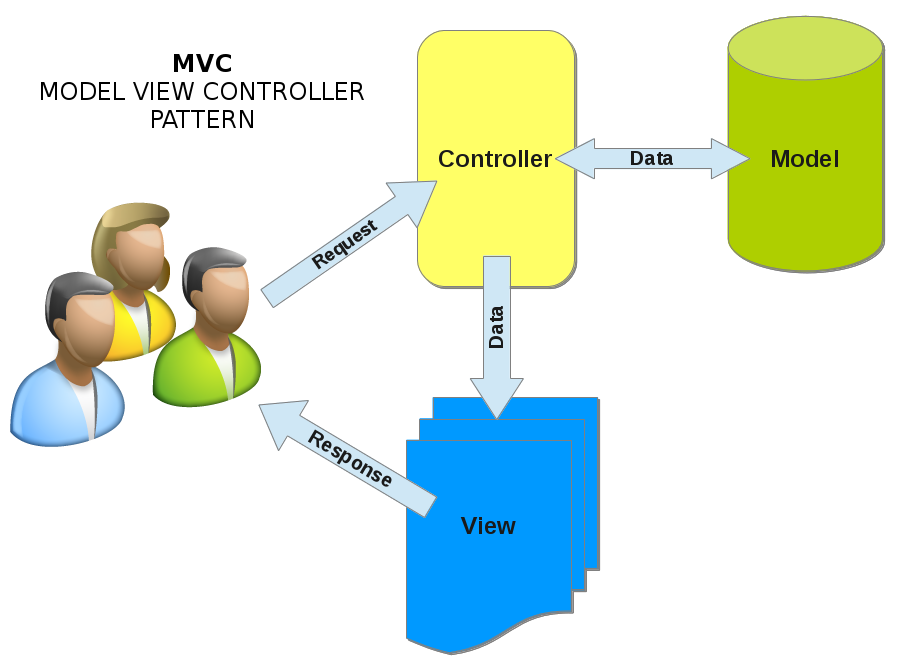
### Architecture de développement : Architecture MVC (Model-View-Controller)

Cette architecture est composée de 3 couches :

**Modèle** : cette couche est chargée de récupérer et gérer les données de la base de données. On y trouve donc les requêtes SQL. Ensuite elle les organise et les rassemble pour être traitées par le contrôleur.

**Contrôleur** : c’est l’intermédiaire entre la vue et le modèle, il reçoit les requêtes des utilisateurs, elle les envoie au modèle, puis notifie la vue pour récupérer les données du modèle.

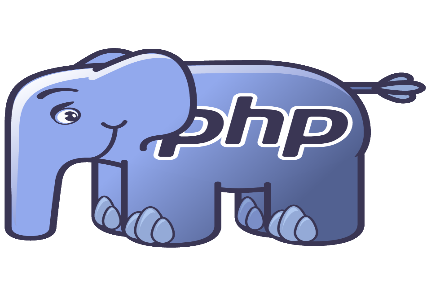
**Vue** : c’est la couche qui s’occupe de la représentation des interfaces graphiques et l’affichage des données traitées et reçues par les autres couches.

On a utilisé cette architecture pour la réalisation de notre application, et pour un souci de maintenabilité.

**Figure 16 : Schéma du modèle MVC**

### Langages de programmations et technologies

Les langages de programmation et technologie utilisés pour développer notre application sont :

* PHP

**Figure 17 : Le langage PHP**

Le PHP, pour HyperText Preprocessor, est un langage de script, utilisé pour concevoir des sites web dynamiques. Il est sous licence libre et est souvent associé aux serveurs MySQL (base de données) et Apache (serveur web).

Ce langage a été utilisé dans notre application pour la partie Back end, créer les interactions avec la base de données.

* HTML

**Figure 18 : Le langage HTML**

HTML est l’abréviation du mot « HyperText Markup Language », qui est un langage de balisage utilisé pour concevoir les pages web. On a utilisé ce langage pour structurer les vues de notre application.

* CSS

**Figure 19 : CSS**

CSS est l’abréviation de « Cascading Style Sheets » est un langage utilisé pour mettre en forme les pages web. Nous l’avons utilisé pour styliser les vues, aider de l’outil Bootstrap pour concevoir des pages responsives.

* Bootstrap

**Figure 20 : Bootstrap**

C’est un ensemble d’outils permettant la conception du design des applications et sites web. En terme technique c’est un Framework css.

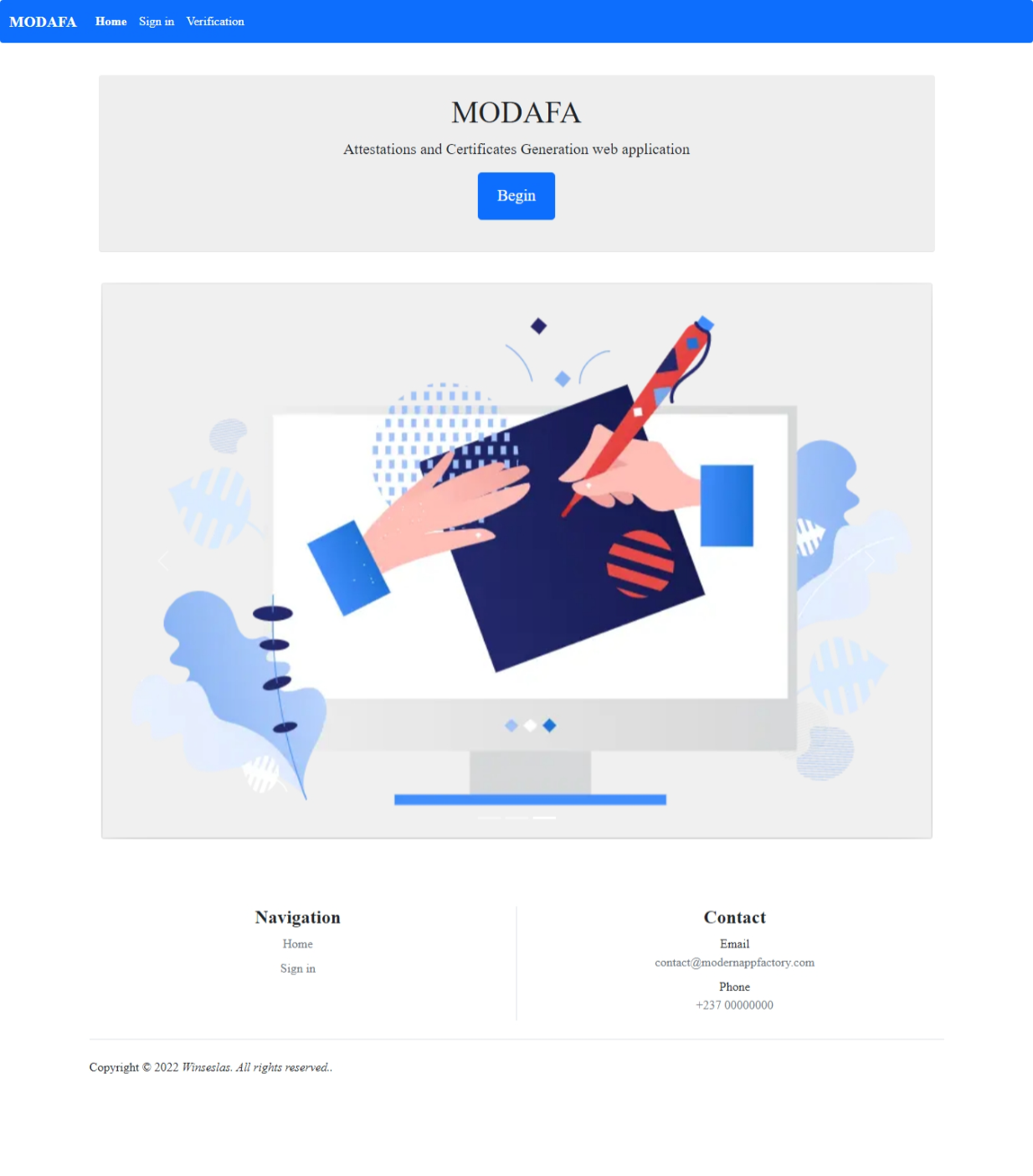
* Symfony

**Figure 21: Logo du Framework Symfony.**

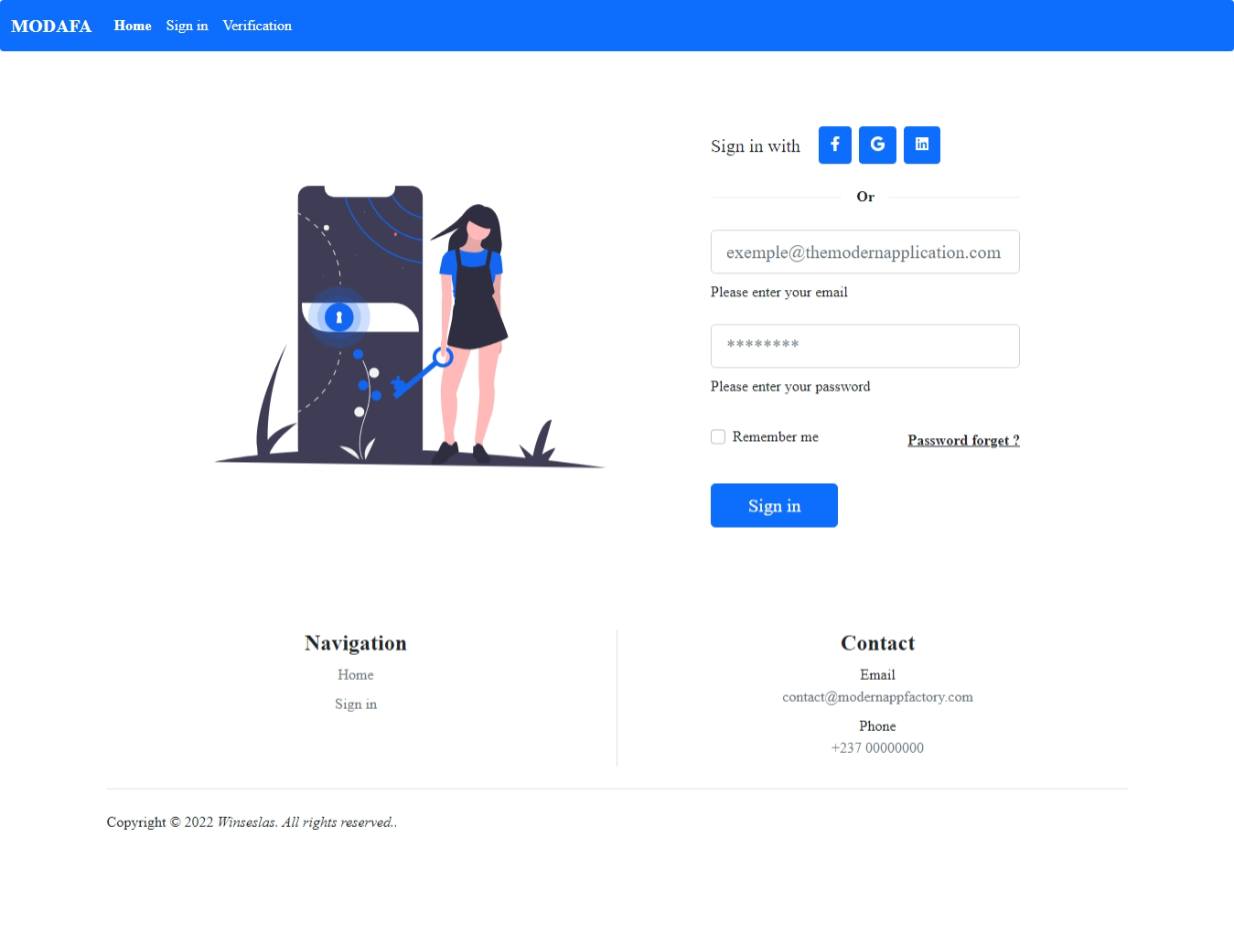
Symfony est un ensemble de composants PHP, un Framework d’application Web, une philosophie et une communauté, tous travaillant ensemble en harmonie. Pour notre solution nous avons utilisé la version 6.\*.

## Présentation des interfaces.

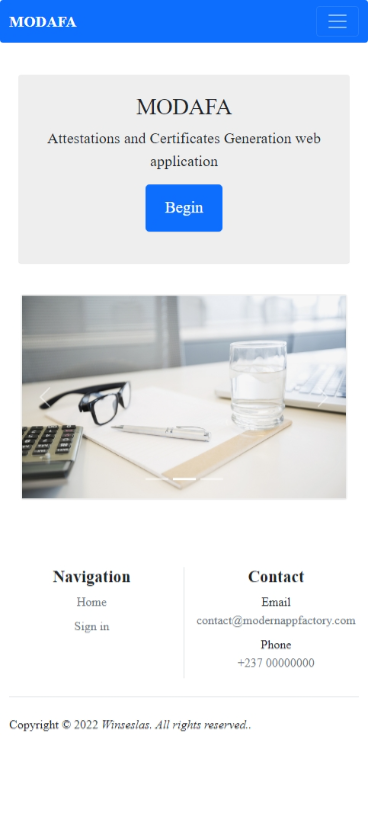
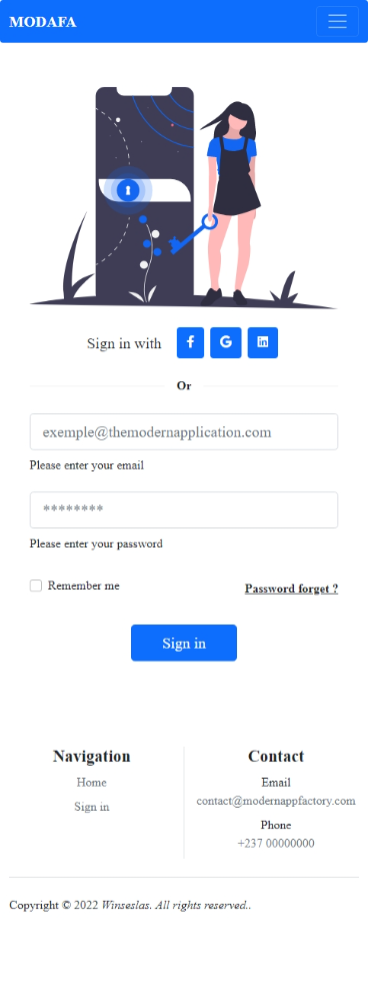
* Page d’accueil



**Figure 22 : Page d’accueil**

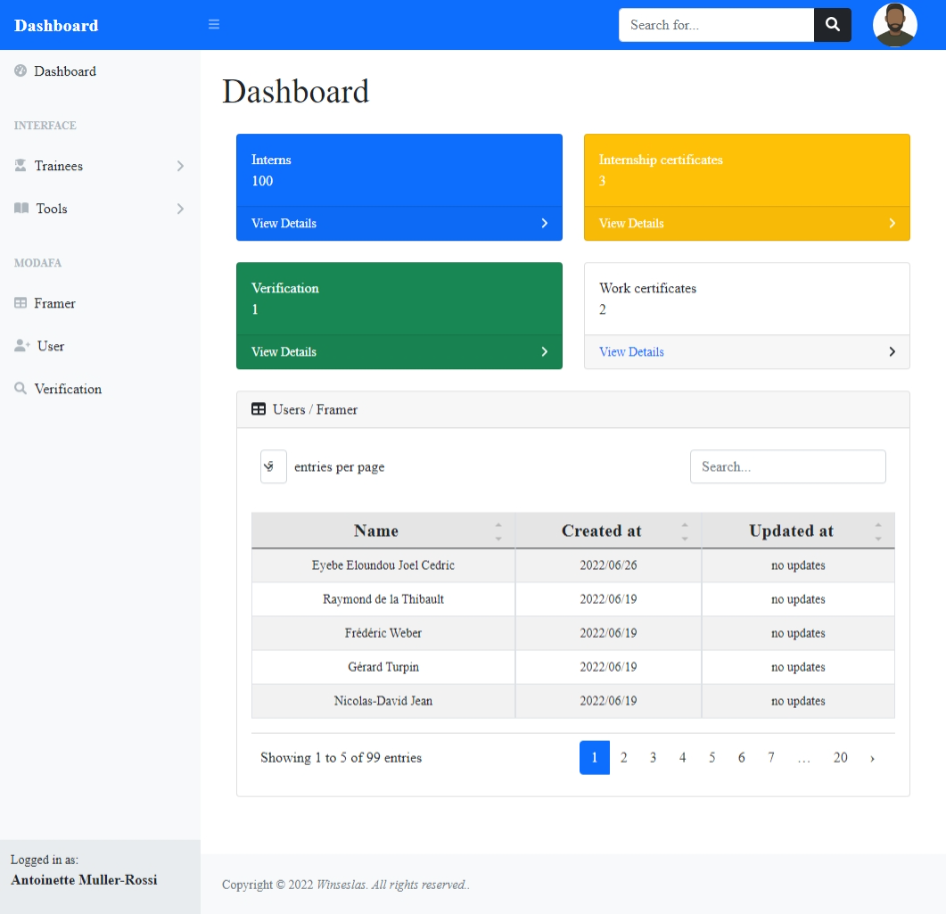
* Page de connexion

**Figure 23 : Page de connexion**

* Pages responsives

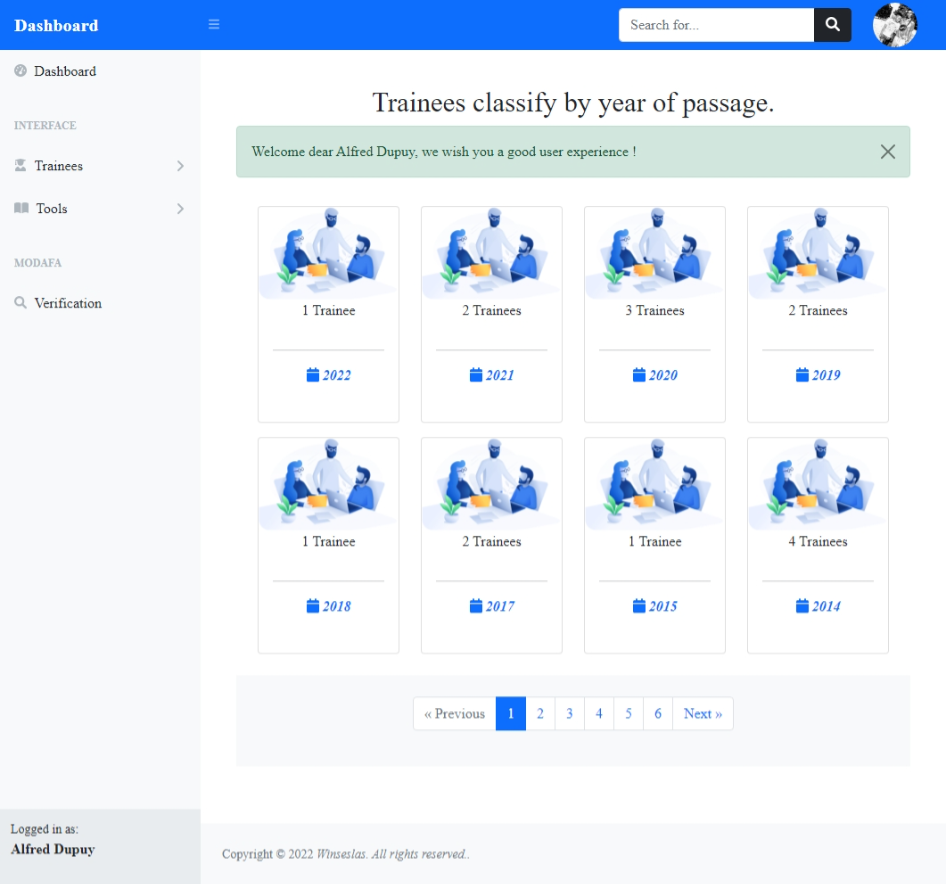
**Figure 24 : Rendu responsive**

* Espace d’admin



**Figure 25 : Espace d’admin**

* Espace user



**Figure 26 : Espace User**

# Conclusion

Dans ce projet qui tire à sa fin, nous devions réaliser une application web de génération des attestions et certificat. La première phase à consister à présenter le cadre du projet. A la deuxième phase nous délimiter le périmètre de notre projet en spécifiant les besoins, représenter pas les diagrammes de cas d’utilisations et de séquences. A la phase suivante, on a réalisé la conception de notre système par UML, ou on a présenté quelques diagrammes permettant d’expliciter la structure statique (diagramme de classes) ainsi que celle dynamique (diagramme de séquences) du système. Enfin, nous avons réalisé l’application, en utilisant des outils et technologies innovants et très répandus dans le domaine de développement et du génie logiciel. Ce travail a été une occasion pour enrichir de plus nos connaissances théoriques et pratiques en développement et en conception, et de pouvoir mieux manipuler des outils tels que Win Design, ainsi que divers langages de programmations tels que PHP, SQL et de découvrir le Framework Symfony… Cependant, c’est la première version de notre solution, qui restera ouverte à des améliorations dans l’avenir.

# Liste de figures

[*Figure 1 : Organigramme de MODAFA 12*](file:///C:\Users\HP\OneDrive\Documents\Stage\Rapports\Memoire.docx#_Toc107555160)

[*Figure 2 : Plan de localisation 13*](#_Toc107555161)

[*Figure 3 : Diagramme de Grantt 15*](#_Toc107555162)

[*Figure 4 : Diagramme de cas d’utilisation de L’internaute. 17*](#_Toc107555163)

[*Figure 5 : Diagramme de cas d’utilisation de l’utilisateur. 18*](#_Toc107555164)

[*Figure 6 : Diagramme de cas d’utilisation de l’admin 19*](#_Toc107555165)

[*Figure 7 : Raffinement des use case. 20*](#_Toc107555166)

[*Figure 8 : Diagramme de séquence ‘‘authenticate’’ 25*](#_Toc107555167)

[*Figure 9 : Diagramme de séquence ‘‘reset\_password’’ 26*](#_Toc107555168)

[*Figure 10 : Diagramme de classes 29*](#_Toc107555169)

[*Figure 11 : Diagramme de séquence de ‘‘authenticate’’ 31*](#_Toc107555170)

[*Figure 12 : Diagramme de séquence de ‘‘create\_user’’ 32*](#_Toc107555171)

[*Figure 13 : Logo Visual studio code 34*](#_Toc107555172)

[*Figure 14 : Logo wampserver 35*](#_Toc107555173)

[*Figure 15 : Logo win Design 35*](#_Toc107555174)

[*Figure 16 : Schéma du modèle MVC 36*](#_Toc107555175)

[*Figure 17 : Le langage PHP 36*](#_Toc107555176)

[*Figure 18 : Le langage HTML 37*](#_Toc107555177)

[*Figure 19 : CSS 37*](#_Toc107555178)

[*Figure 20 : Bootstrap 37*](#_Toc107555179)

[*Figure 21: Logo du Framework Symfony. 38*](#_Toc107555180)

[*Figure 22 : Page d’accueil 38*](#_Toc107555181)

[*Figure 23 : Page de connexion 39*](#_Toc107555182)

[*Figure 24 : Rendu responsive 40*](#_Toc107555183)

[*Figure 25 : Espace d’admin 41*](#_Toc107555184)

[*Figure 26 : Espace User 42*](#_Toc107555185)

# Liste de tableaux

[*Tableau 1 : Scénario nominal du cas d’utilisation ‘‘authenticate’’ 21*](#_Toc107555752)

[*Tableau 2 : Scénario nominal du cas d’utilisation ‘‘add an intern’’ 22*](#_Toc107555753)

[*Tableau 3 : Scénario nominal du cas d’utilisation ‘‘Reset password’’ 23*](#_Toc107555754)

[*Tableau 4 : Scénario nominal du cas d’utilisation ‘‘create user’’ 24*](#_Toc107555755)