Une image contenant texte, clipart

Description générée automatiquement

Moktar TERKI

easLILLE | 27 RUE jEAN bART 59000 lille

Rapport de stage

21/02/2022 AU 13/05/2022

Table des matières

[Compétences couvertes 4](#_Toc104822782)

[Le Résumé du projet 5](#_Toc104822783)

[Le Résumé du projet (en anglais) 6](#_Toc104822784)

[Remerciements 7](#_Toc104822785)

[Introduction 8](#_Toc104822786)

[Présentation de l’association 9](#_Toc104822787)

[Qui Sommes-nous ? 9](#_Toc104822788)

[Ce que nous fesons 12](#_Toc104822789)

[Notre éthique 13](#_Toc104822790)

[Photos du stade et de l’association 14](#_Toc104822791)

[Cahier des charges 15](#_Toc104822792)

[Trello 17](#_Toc104822793)

[Diagramme de GANTT 17](#_Toc104822794)

[Git 18](#_Toc104822795)

[Technologie utilisée (développement) 19](#_Toc104822796)

[Spécifications fonctionnelles 22](#_Toc104822797)

[Interface Visiteur 24](#_Toc104822798)

[Interface Utilisateur 25](#_Toc104822799)

[Interface Administrateur 27](#_Toc104822800)

[Diagramme cas d’utilisation 28](#_Toc104822801)

[MCD 30](#_Toc104822802)

[MLD 30](#_Toc104822803)

[SQL 31](#_Toc104822804)

[Maquette 33](#_Toc104822805)

[Diagramme de classes 34](#_Toc104822806)

[RGPD 35](#_Toc104822807)

[**L’utilisation de cookies** 36](#_Toc104822808)

[Veille sécurité formulaire inscription 37](#_Toc104822809)

[Spécifications techniques 39](#_Toc104822810)

[HTML 39](#_Toc104822811)

[PHP 39](#_Toc104822813)

[JS 43](#_Toc104822814)

[CSHARP application bureau 49](#_Toc104822815)

[JAVA 51](#_Toc104822816)

[Recherche site anglophone 52](#_Toc104822817)

[Problème rencontrer 52](#_Toc104822818)

[Annexes Diagramme de flux 56](#_Toc104822819)

[Veille technologique 58](#_Toc104822820)

[**La veille sécurité pour protéger votre S.I** 58](#_Toc104822821)

[**Une veille délaissée par les pros** 59](#_Toc104822822)

[60](#_Toc104822823)

[60](#_Toc104822824)

[Déploiement de l’application 61](#_Toc104822825)

[Accès au serveur : 61](#_Toc104822826)

[Installation des fichiers : 61](#_Toc104822827)

[Installation de la base de données : 62](#_Toc104822828)

[Test de déploiement : 62](#_Toc104822829)

[Conclusion stage 63](#_Toc104822830)

[Conclusion formation 64](#_Toc104822831)

[Glossaire 65](#_Toc104822832)

# Compétences couvertes

* Maquetter une application.
* Concevoir une application.
* Développer une interface utilisateur de type desktop.
* Développer des composants d’accès aux données.
* Développer la partie front-end d’une interface utilisateur web.
* Développer la partie back-end d’une interface utilisateur web.
* Concevoir une base de données.
* Mettre en place une base de données.
* Développer des composants dans le langage d’une base de données.
* Collaborer à la gestion d’un projet informatique et à l’organisation de l’environnement de développement.
* Concevoir une application
* Développer des composants métier.
* Construire une application organisée en couches.
* Développer une application mobile.
* Préparer et exécuter les plans de tests d’une application.
* Préparer et exécuter le déploiement d’une application.

# Le Résumé du projet

Est Afrique Sport de Lille (EAS Lille) a été créé le 28/12/2020.

Cette association régie par la loi du 01/07/1901 a pour objet de : Solidariser, construire et d'entraide communautaire, qui regroupent les membres de la diaspora d’Afrique de l'Est autour de la pratique sportive hebdomadaire.

L’association avait un besoin d’une application web pour permettre à ses membres de se connecter à distance et de suivre les actualités de l’association.

Durant ma période de stage j'ai travaillé pour une association : EAS Lille, qui regroupe des gens d'Afrique de l'Est et qui a la pratique du football comme pilier.

Cette application vous permettra de gérer les différents événements de EAS LILLE comme les sorties sportives, un fil d'actualité, la création de nouveaux membres, la création d'un chat entre abonnés, la gestion des dons, le paiement des cotisations, la publication des CV des membres.

L'association devait accomplir différentes tâches et connecter ses membres à distance, pour cela il fallait une application web.

J'ai donc conçu puis développé une application web qui sert notamment pour l'inscription des membres, la gestion des contributions au sein de l'association, communication entre les membres authentifiées via un chat, la publication de nouvelles sur les événements et service associatif.

A l'issue de la phase de conception, j'ai choisi d'utiliser les technologies suivantes :

PHP/JS utilisé pour exécuter des requêtes sur la base de données, base de données MySQL pour stockage d'informations, C-Sharp pour l'application de bureau.

# Le Résumé du projet (en anglais)

Est Afrique Sport de Lille (EAS Lille) was created on 28/12/2020 This

Association governed by the law of 01/07/1901 aims to: Solidarize, build inter-community mutual aid, bring together members of the East African diaspora around the practice sports weekly.

During my internship period I worked for an association: EAS Lille, which brings together people from East Africa and which has the practice of football as a pillar.

This application will allow you to manage the various EAS Lille events

such as sports outings a news feed, the creation of new members, the

creation of a chat between subscribers, the management of monthly

contributions (online or cash), the publication of CVs of members.

The association needed to perform different tasks and connect its

members remotely, for that it needed a web application.

I therefore designed and then developed a web application which is used in particular for the registration of members, the management of

contributions within the association, communication between members

authenticated through a chat, the publication of news on events and

association services.

At the end of the design phase, I chose to use the following technologies PHP/JS used for executing requests on the database, MySQL database for storing information, C-Sharp for desktop application.

# Remerciements

Avant de commencer le développement de cette première expérience professionnelle, il me parait naturelle de commencer par remercier les personnes qui m’ont permis d’effectuer ce travail ainsi que ceux qui m’ont permis d’en faire un moment agréable et profitable.

Je tiens à remercier le conseil régional pour m’avoir financé cette formation ainsi que l’Afpa pour sa qualité de formation.

Je tiens à remercier ma formatrice Martine POIX de m’avoir accepté dans cette formation et de n’avoir cessé de m’encourager pendant la toute sa durée ainsi que pour sa générosité et son professionnalisme en matière de formation.

Cela m’a offert la possibilité de découvrir plus en profondeur ce métier qui me plaît beaucoup et ainsi de vivre cette expérience professionnelle.

Je tiens à remercier aussi mes collègues de formation pour leur sympathie et gentillesse ainsi aux conseils qu’ils ont pu me prodiguer au cours de cette formation.

Merci !

# Introduction

Suite à notre formation théorique de Concepteur développeur d’application, nous avons dû poursuivre notre enseignement en condition réelle en entreprise durant une période de 12 semaines de stages.

J’ai choisi d’effectuer mon stage auprès de l’associations EAS Lille qui est une association sportive nouvellement créer et ne possédant pas de site internet. Les raisons qui m’ont poussé à choisir EAS Lille fut l’attractivité du projet avec la mise en place d’un site web qui devaient allier ergonomie de l’interface, expérience utilisateur et besoins fonctionnels ainsi qu’une application bureau facilitant le travail de l’administrateur. J’avais aussi la conviction que j’allais évoluer dans un environnement de travail propice au développement de mes compétences.

Ce stage plus qu’enrichissant m’a permis de voir toutes les étapes d’un projet (analyse, conception, développement) au-delà d’enrichir mes connaissances en développement, ce stage m’a permis d’accroitre mon sens du travail en équipe.

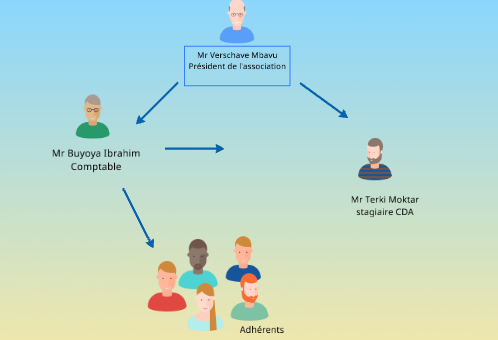
Je tiens à remercier Mr MBAVU (président de l’association) de m’avoir confié ce projet ainsi que Mr BUYOYA le secrétaire de EAS Lille (mon tuteur).

# Présentation de l’association

## Qui Sommes-nous ?

Est Afrique Sport de Lille a été créé le 28/12/2020 Cette association régit par la loi du 01/07/1901 a pour but : Solidariser, nouer l'entraide intercommunautaire, rassembler les membres de la diaspora est africaine autour de la pratique hebdomadaire du sport.

Mr MBAVU est le président de l’association et Mr BUYOYA a été mon tuteur de stage.



L’association organise des évènements pour partager et promouvoir les cultures et mœurs Est africaines (évènement sportif, exposition, concert, etc…).

Elle soutient l'émergence des projets visant la création d'activités locales et d'emplois dans les pays de l’Afrique.

Elle facilite l'intégration des membres de la communauté africaine dans la région lilloise.

Pour faire partie de cette association, il faut adhérer et s'acquitter de la cotisation annuelle.

Les mineurs peuvent adhérer à l'association sous réserve d'un accord de leurs parents ou tuteurs légaux. Ils sont membres à part entière de l'association.

Le siège social est fixe à la maison des associations au 27, rue jean Bart 59000 Lille.

Une image contenant carte

Description générée automatiquement

Les ressources de l'associations comprennent les dons, les subventions, cotisations annuelles, les recettes des ventes de prestations, en relation avec les activités de l’association.

## Ce que nous fesons

Le but de notre structure s’inscrit dans un cadre de dialogue, d’échanges et de dynamisme dans la synergie d’action.  
Les Principales activités de l’Association des étudiants africains du Nord Pas de Calais.

1- Créer un lien de solidarité entre les étudiants ;

2- Mise en place d’une structure d’information et d’accueil des nouveaux étudiants ;

3- Mise en place des cours de tutorats pour les étudiants en difficulté ;

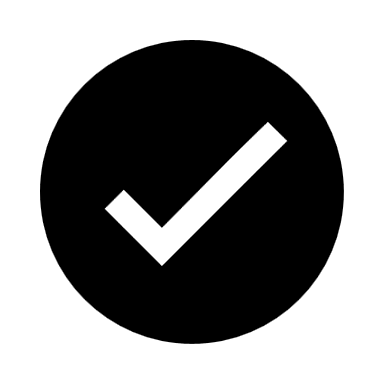
4- Organisation des manifestations sportives, culturelles et artistiques ;

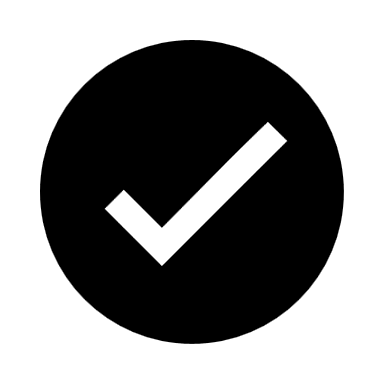
5- Mise en place d’un fond d’aide et de solidarité ;

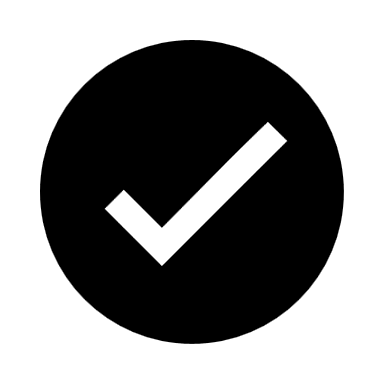
6- Coopération avec les autres associations.

Notre objectif, contribué à l’épanouissement de nos membres par un engagement commun qui fait un succès général.

## Notre éthique

Travail  
Nous réalisons nos ambitions par le travail, à travers lui nous façonnons notre environnement en l’imprimant notre idéal.  
C’est un levier important par lequel nos chances se multiplient et nos rêves se concrétisent.  
Nous œuvrons permanemment pour cultiver l’amour du travail bien accompli.

Solidarité  
Nous estimons que Tout groupe humain prend sa richesse dans la communication, l’entraide et la solidarité visant à un but commun :  
l’épanouissement de chacun dans le respect des différences.  
Nous enseignons à chacun de nos membres, la compréhension entre les humains qui est la condition et le garant de la solidarité intellectuelle et morale.

Transparence  
Notre credo réside dans notre capacité de gestion en toute sincérité.  
L’opacité sous toutes ses formes est bannie de notre mode de gestion.  
Nous estimons que notre intégrité, notre probité morale et notre crédibilité y dépendent tant pour nos élus que pour nos collaborateurs.  
S’investir pour plus de transparence est de mise au sein de notre association en général et du bureau exécutif en particulier.  
Le laxisme est notre ennemi commun pour toujours réussir ce pari.

## Photos du stade et de l’association

Une image contenant carte

Description générée automatiquement

Une image contenant arbre, extérieur, herbe, champ

Description générée automatiquement



# Cahier des charges

**Introduction :**

Projet de réalisation d’un site web. L’application aura une interface visuelle, facilement compréhensible et intuitive. Cette application permettra de gérer les différents évènements de EAS Lille tels que les sorties sportives, un fil d’actualité, la création des nouveaux membres, la création d’un tchat entre abonnés, la gestion des cotisations mensuels (en ligne ou espèce), la publication de cv des adhérents. Un aspect de connexion permet une interaction différente avec l’application suivant un rôle donné.

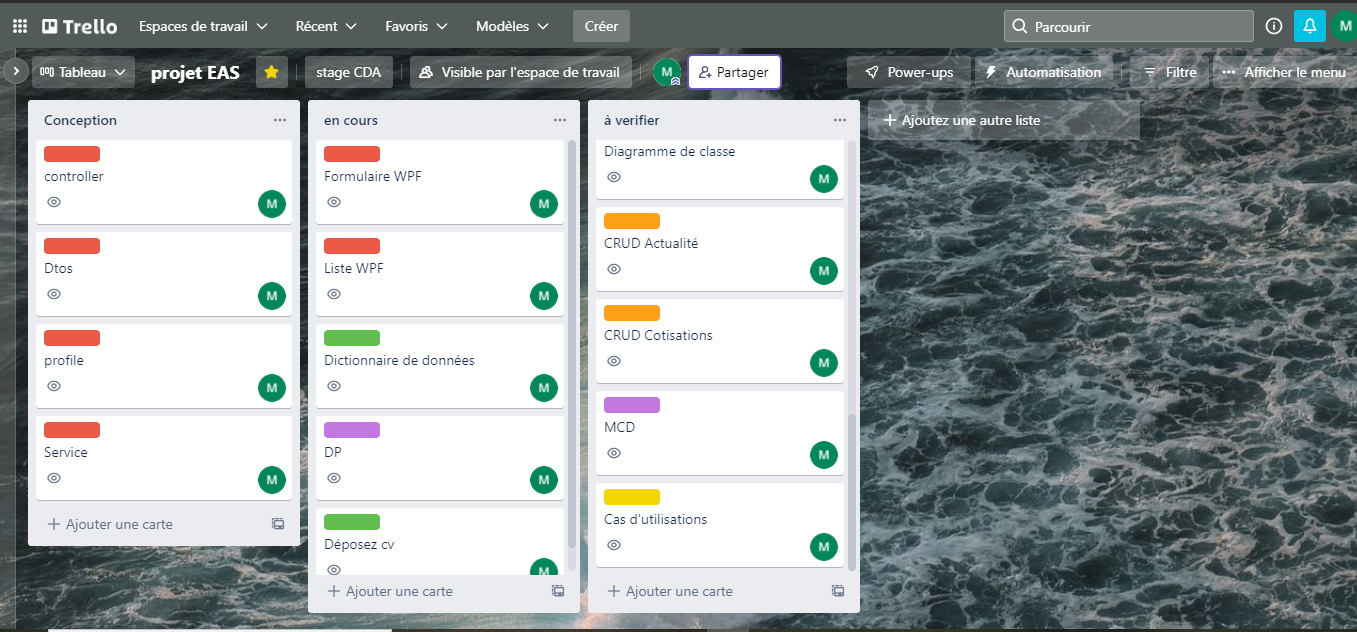
1. **Structure de l’application**

L’application aura une partie stockage de données via base de données dans un but de pérenniser les données et influer sur les fonctionnalités.

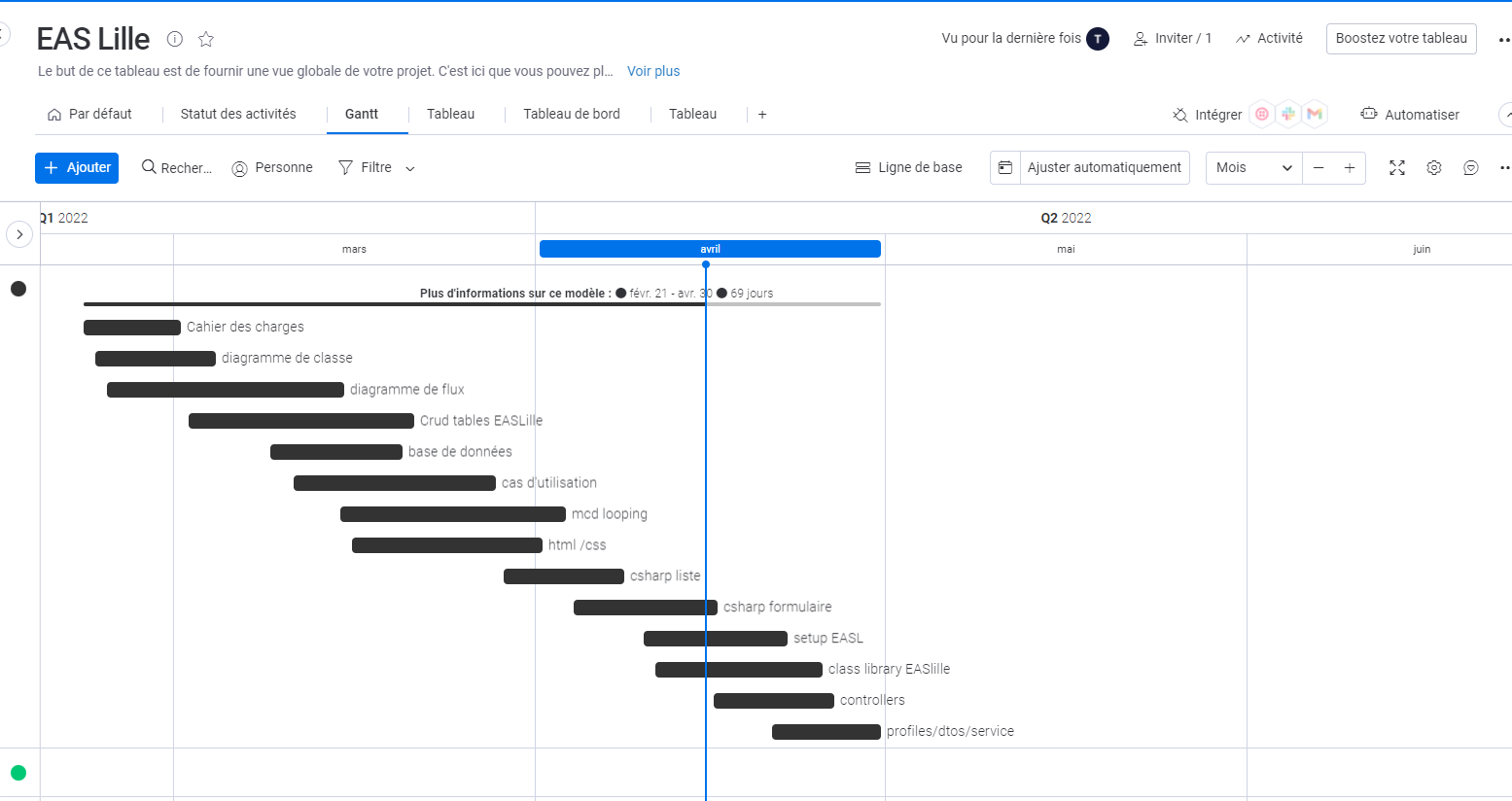
* Un résumé mensuel de l’état des cotisations.
* Historique de tous les paiements par adhérent.
* Un fil d‘actualité contenant des images et des vidéos.
* Une possibilité de faire des dons à l’association.
* Une messagerie instantanée réservée aux adhérents

1. **Une interface visuelle facilement accessible** 
   1. Un page d’accueil permettant de sélectionner directement le service désiré (cotisations, rencontre sportive, tchat, fil du jour, info sur l’association).
   2. Une interface interactive permettant d’avoir les infos du jour ainsi que plusieurs informations essentielles sur l’association.
   3. Un accès administrateur permettant de modifier, d’ajouter ou d’effacer des informations, des utilisateurs, de programmer des événements (back office).
   4. Une interface de connexion et d’inscription permettant soit de se connecter à l’application, soit de s’inscrire. Deux profils pourront interagir avec l’application, les utilisateurs ainsi que les administrateurs.
   5. Le fil d’actualité publie les événements de l’association en temps réel avec la page Facebook de EAS Lille.
   6. Une charte graphique ayant un code couleur (orange #ffa500, vert #008000, noir #000000) reprenant les couleurs du logo de l’association. Un système de cartes permettant l’accès à tous les thèmes du menu.

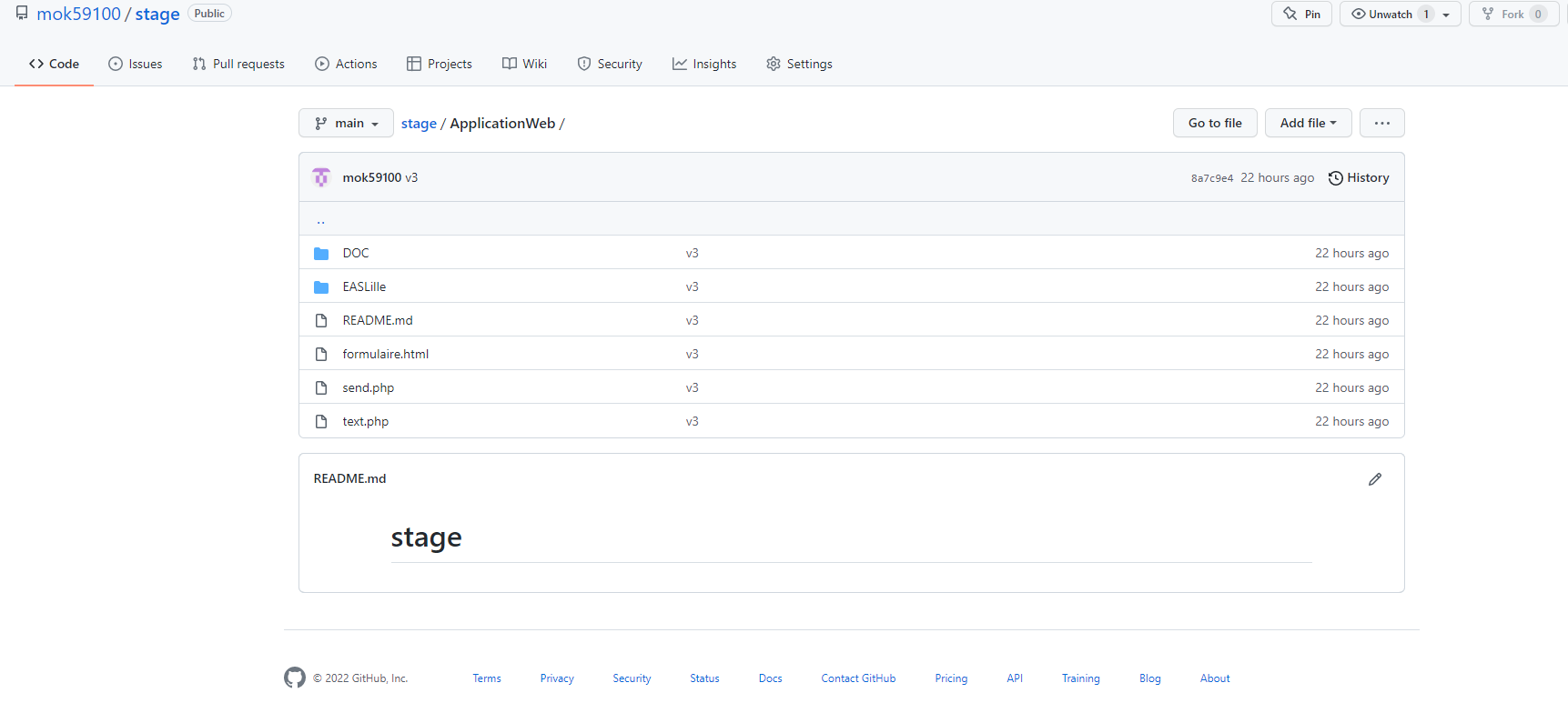
# Trello



# Diagramme de GANTT

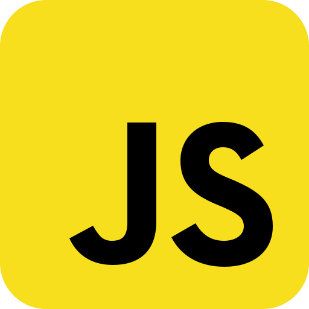


# Git



# Technologie utilisée (développement)

**Visual Studio** Code est un éditeur de code qui est disponible sur Windows, MacOs et Linux. Il est fourni avec un support pour JavaScript, type Script ou Node.js On a la possibilité de télécharger des extensions pour tout type de langage (c++, c#, Java, Python, PHP, etc..



**Javascript** souvent abrégé JS est un langage de programmation qui est conforme à la spécification ECMAScript. Il s’agit d’un standard concernant les langages de programmation de type script. Javascript est souvent utilisé sur les pages web afin de les rendre plus vivantes.

Une image contenant texte

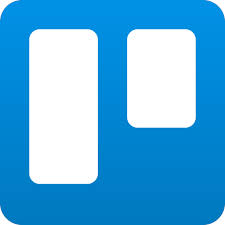
Description générée automatiquement

**PHP** signifiant HyperText Préprocesseur est un langage open source s’exécutant cote serveur ainsi que le HTML qui sera envoyer au client, de ce fait il permet de produire des pages web dynamique

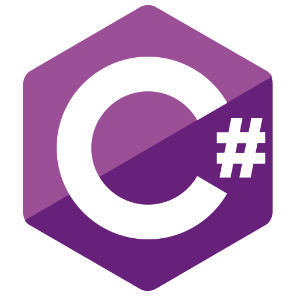
 J’utilise la plateforme de développement **WampServer** qui me permet de faire fonctionner mes scripts PHP en local grâce au serveur Apache. Pour fonctionner WAMP a donc besoin d’un système d’exploitation (Windows), d’un serveur Apache, d’un système de gestion de base de données (MySQL) et enfin d’un langage de script (PHP)

**AJAX** Signifie Asynchronous Javascript and XML. Il ne s’agit pas d’un langage ou d’une technologie en soi mais d’un ensemble de technologies qui permettent la mise à jour d’un contenu ciblé d’une page web de manière rapide et sans charger l’intégralité de la page.

***phpMyAdmin*** (PMA) est une [application Web](https://fr.wikipedia.org/wiki/Application_Web) de gestion pour les [systèmes de gestion de base de données](https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_base_de_donn%C3%A9es) [MySQL](https://fr.wikipedia.org/wiki/MySQL) et [Maria DB](https://fr.wikipedia.org/wiki/MariaDB), réalisée principalement en [PHP](https://fr.wikipedia.org/wiki/PHP). Il s'agit de l'une des plus célèbres interfaces pour gérer une base de données [MySQL](https://fr.wikipedia.org/wiki/MySQL) surun serveur [PHP](https://fr.wikipedia.org/wiki/PHP).



**Trello** c’est un moyen visuel d’organiser ce que l’on veut avec qui on veut. Ici pour l’entreprise il permet de définir les taches de chacun

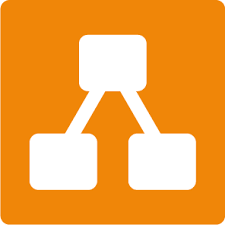
  
**C#** (*C Sharp* ) est un [langage](https://fr.wikipedia.org/wiki/Langage_de_programmation) de [programmation orientée objet](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet), commercialisé par [Microsoft](https://fr.wikipedia.org/wiki/Microsoft) depuis 2002[3](https://fr.wikipedia.org/wiki/C_sharp#cite_note-tata-3) et destiné à développer sur la plateforme [Microsoft .NET](https://fr.wikipedia.org/wiki/Microsoft_.NET). Il est dérivé du [C++](https://fr.wikipedia.org/wiki/C%2B%2B) et très proche du [Java](https://fr.wikipedia.org/wiki/Java_(langage)) dont il reprend la syntaxe générale ainsi que les concepts, y ajoutant des notions telles que la [surcharge des opérateurs](https://fr.wikipedia.org/wiki/Surcharge_des_op%C3%A9rateurs), les indexeurs et les [délégués](https://fr.wikipedia.org/wiki/C_sharp#Diff%C3%A9rences_avec_le_C++). Il est utilisé notamment pour développer des [applications web](https://fr.wikipedia.org/wiki/Applications_web) sur la plateforme [ASP.NET](https://fr.wikipedia.org/wiki/ASP.NET).

 **Java** est un **langage de programmation**et une **plate-forme informatique** créée par Sun Microsystems. Utilisé par 9 millions de développeurs dans le monde, ce langage est nécessaire pour le fonctionnement de beaucoup d’applications et de sites Web. Il est la base de la plupart des applications en réseau : des jeux, des téléphones portables à Internet, du contenu web et des logiciels entreprises**.**

**Looping**est un logiciel de modélisation conceptuelle de données qui vous permet d'organiser et de structurer vos données en diagrammes et représentations graphiques ordonnées. **Looping**vous permet notamment de modéliser des entités et des associations, de générer des diagrammes de classes UML.



**Android Studio** un [environ de développement](https://fr.wikipedia.org/wiki/Environnement_de_d%C3%A9veloppement) pour développer des applications mobiles [Android](https://fr.wikipedia.org/wiki/Android). Il est basé sur  [IntelliJ IDEA](https://fr.wikipedia.org/wiki/IntelliJ_IDEA) et utilise le [moteur de production](https://fr.wikipedia.org/wiki/Moteur_de_production) [Gradle](https://fr.wikipedia.org/wiki/Gradle" \o "Gradle). Il peut être téléchargé sous les systèmes d'exploitation [Windows](https://fr.wikipedia.org/wiki/Microsoft_Windows), [macOS](https://fr.wikipedia.org/wiki/MacOS" \o "MacOS), [Chrome OS](https://fr.wikipedia.org/wiki/Chrome_OS) et [Linux](https://fr.wikipedia.org/wiki/Linux).

**Drawio** est un logiciel de [dessin graphique](https://en.wikipedia.org/wiki/Graph_drawing) multiplateforme [gratuit et open source](https://en.wikipedia.org/wiki/Free_and_open-source_software) développé en [HTML5](https://en.wikipedia.org/wiki/HTML5) et [JavaScript](https://en.wikipedia.org/wiki/JavaScript) . Son interface peut être utilisée pour créer des [diagrammes](https://en.wikipedia.org/wiki/Diagram) tels que des [organigrammes](https://en.wikipedia.org/wiki/Flowchart) , des structures [filaires](https://en.wikipedia.org/wiki/Website_wireframe) , des diagrammes [UML](https://en.wikipedia.org/wiki/Unified_Modeling_Language) , [des organigrammes](https://en.wikipedia.org/wiki/Organizational_chart) et [des diagrammes de réseau](https://en.wikipedia.org/wiki/Computer_network_diagram).

# Spécifications fonctionnelles

**Introduction :**

Application d’une association sportive accessible sur le web (application web), desktop (accessible sur le pc de l’utilisateur sans connexion internet).

L’application aura pour but une connexion sécurisée à son compte utilisateur ainsi que visualiser les cotisations de EAS Lille, chatter depuis l’application, créer des actualités, les animer et/ou y participer.

1. **Espace connexion**

L’application sera pourvue d’un espace connexion permettant d’entrer dans l’application suivant un rôle donné. La connexion se fait à l’aide d’une adresse mail et d’un mot de passe.

1. **Espace Inscription**

L’application permettra une inscription, cette inscription demandera de nombreuses informations :

* + - * + Nom
        + Prénom
        + Adresse Mail
        + Mot de passe
        + Confirmation du mot de passe

L’inscription sera validée par l’administrateur. La personne contact l'association par le biais du formulaire disponible sur le site.

1. **Une application moderne et accessible**

Interface facile d’utilisation et de compréhension un design moderne

1. **Don**

Un don peut être fait par tous les membres (visiteurs ou utilisateurs) sans être forcément inscrit en tant qu’utilisateur. Les informations demandées seront son nom, prénom, libellé du don ainsi que le montant.

1. **Cotisation**

Elle est obligatoire pour tous les membres de l’association sauf dans le cas des personnes en difficultés (situation Professionnelle) ou celle-ci ne sera pas redevable avec l’accord de la direction au cas par cas. Ils auront accès à toutes les fonctions de l’application en tant qu’utilisateur.

1. **Adhésion**

Celle-ci permet le suivi de tous les utilisateurs en admettant qu’un utilisateur s’abonne pendant une période et quitte l’association avant de revenir à celle-ci. Pendant cette période d’intermittence les données de l’utilisateur seront gardées en base de données.

## Interface Visiteur

1. **Accueil**

Visibilité de toutes les infos de l’association avec les différents services, faire un don, voir les activités sportives ainsi que les actualités.

1. **Nos services**

Outil permettant de proposer tous les services de l’associations, celle-ci sera disponible sans être connecté grâce à un compte utilisateur. Il pourra néanmoins voir les photos et vidéos de l’association.

1. **Soutenir**

Le visiteur aura la possibilité de faire un don sans connexion préalable, par contre au moment de faire son don on demandera au visiteur des informations tel que (nom, prénom, montant, libellé du don).

1. **Actualité**

On trouvera toutes les actualités de l’association ainsi que le fil d’actualité Facebook. Il pourra voir les différents évènements en cours.

1. **Activité sportive**

On trouvera toutes les actualités sportives de l’association. Il pourra voir les différents évènements sportifs ainsi que les résultats.

1. **Inscription**

Le visiteur aura la possibilité de remplir un formulaire d’inscription.il devra remplir des informations tel que (nom, prénom, email, mot de passe, situation professionnelle, numéro de téléphone) il aura à sa disposition la possibilité de laisser un message pour expliquer sa situation. L’accord de son inscription se fera par retour de message par l’administrateur.

## Interface Utilisateur

1. **Accueil**

Visibilité de toutes les infos de l’association avec les différents services, faire un don, voir les activités sportives ainsi que les actualités.

1. **Nos services**

Outil permettant de proposer tous les services de l’associations, voir les photos et vidéos de l’association.

1. **Soutenir**

Le visiteur aura la possibilité de faire un don sans connexion préalable, cependant au moment de faire son don on demandera au visiteur des informations tel que (nom, prénom, montant, libellé du don).

1. **Actualité**

On trouvera toutes les actualités de l’association ainsi que le fil d’actualité Facebook. Il pourra voir les différents évènements en cours.

1. **Activité sportive**

On trouvera toutes les actualités sportives de l’association. Il pourra voir les différents évènements sportifs ainsi que les résultats

1. **Cotisation**

Il pourra payer ses cotisations en direct et voir le résumer de toute ses cotisations

1. **Chatter**

L’utilisateur pourra envoyer des messages en direct et en recevoir grâce à un tchat. Il pourra discuter avec les autres adhérents, poster des messages, voir des photos ainsi que suivre le fil d’actualité Facebook.

1. **Espace connexion**

Un espace connexion permettant d’entrer dans l’application suivant un rôle donné. La connexion se fait à l’aide d’une adresse mail et d’un mot de passe.

## Interface Administrateur

**1.** **Accueil**

Visibilité de toutes les infos de l’association avec les différents services de gestions réservés à l’administrateur

**2. CRUD actualité**

L’administrateur peut modifier ajouter, afficher, supprimer les actualités. Il pourra aussi mettre des nouvelles images et vidéos.

**3. CRUD Adhérent**

L’administrateur peut ajouter modifier, afficher, supprimer les adhésions. Il pourra aussi gérer les nouvelles adhésions, leur envoyer un message.

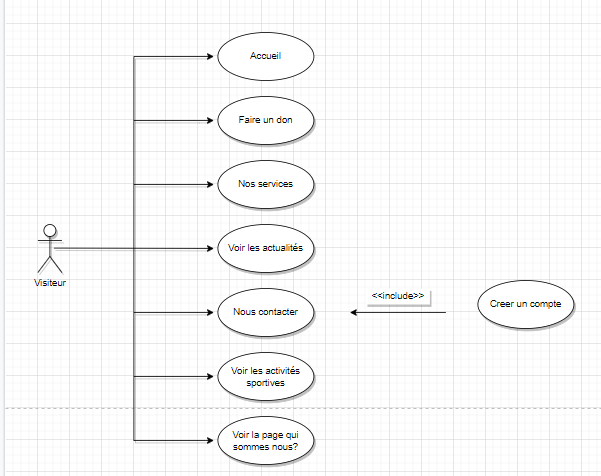
**4. CRUD Cotisation**

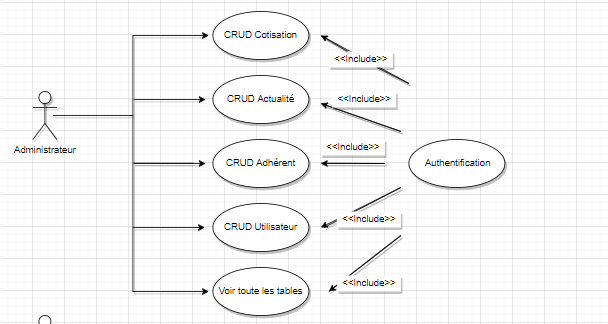
L’administrateur peut ajouter modifier, afficher, supprimer les cotisations. Il pourra avoir un tableau de bord sur toutes les cotisations et ainsi gérer les retards de paiement. Néanmoins il est leur seul a offrir les cotisations aux personnes ayant un statut professionnel faible ou ceux qui ne travaille pas.

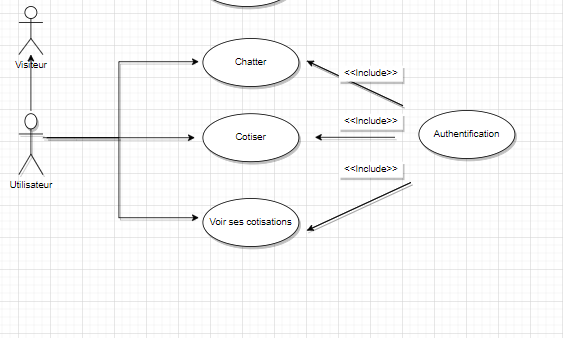
**5. CRUD Utilisateur**

L’administrateur peut ajouter modifier, afficher, supprimer les utilisateurs. Toutes les données des utilisateurs sont sauvegardées. Son profil sera soit activé ou désactivé, ainsi lors de son retour il pourra facilement se connecter

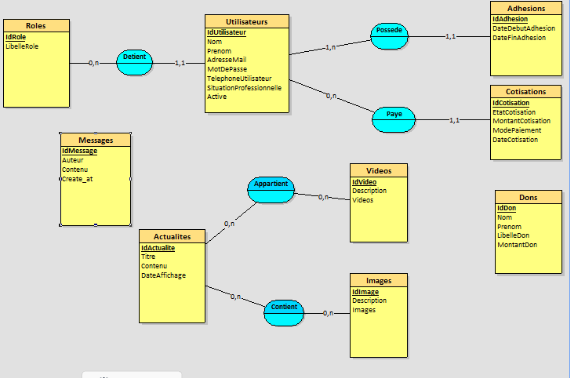
# Diagramme cas d’utilisation

20y%3D%22470%22%20width%3D%2230%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%E%3C%2Froot%3E%3C%2FmxGraphMode





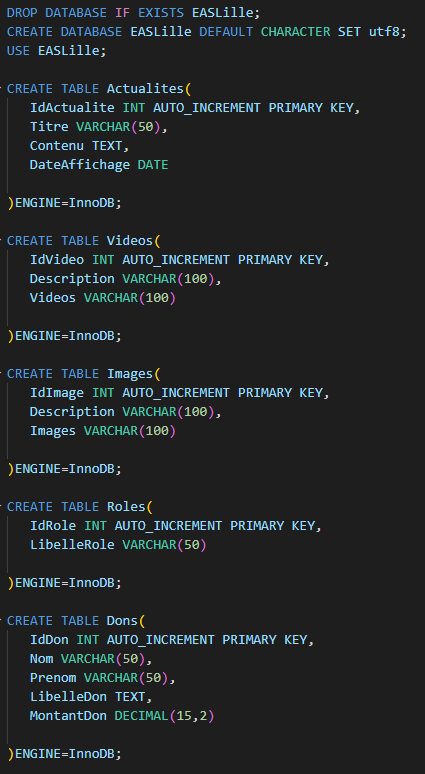
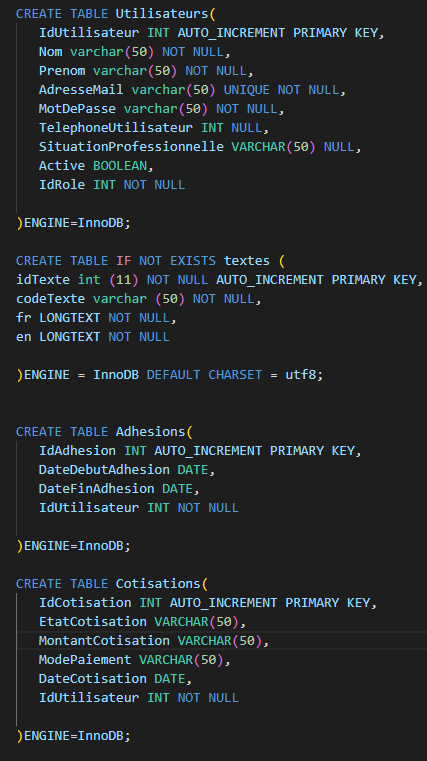
# MCD

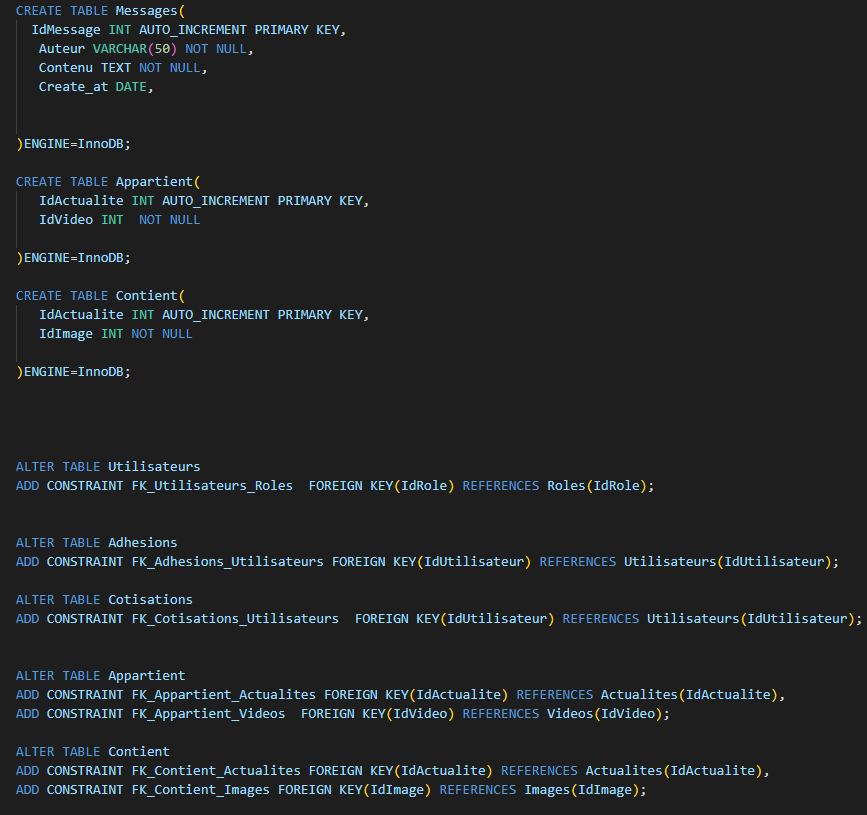
****

# MLD

****

# SQL

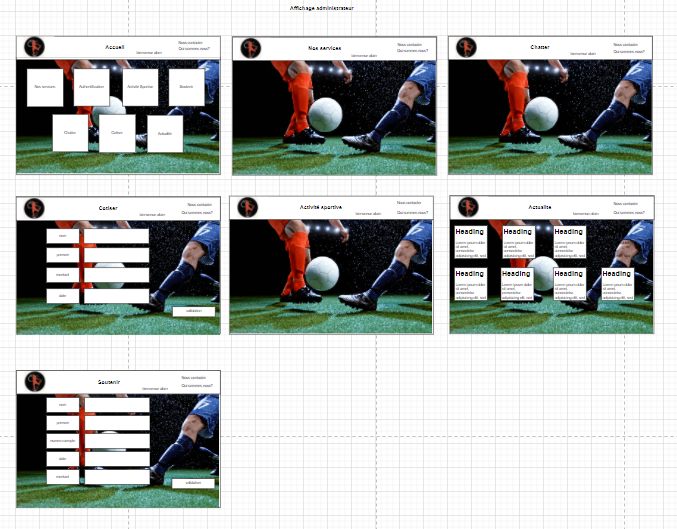
** **

****

# Maquette

**Une image contenant texte, intérieur

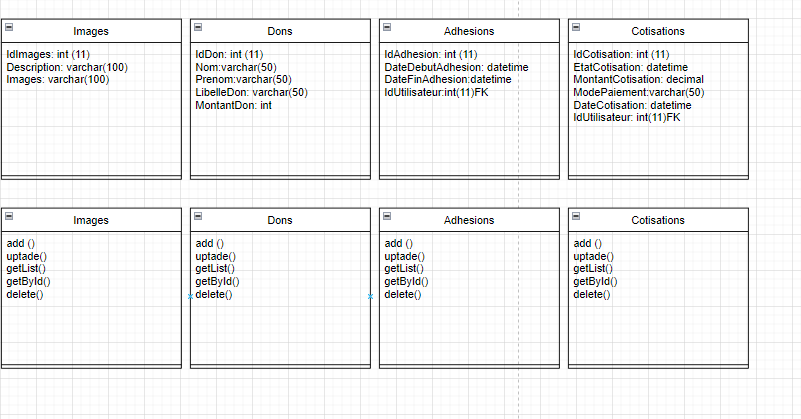
Description générée automatiquement**

****

# Diagramme de classes

**Une image contenant texte, intérieur, capture d’écran, ligné

Description générée automatiquement**

****

# RGPD

**Qu’est-ce que le RGPD ?**

**L’acronyme RGPD signifie Règlement général sur la protection des données. Le RGPD encadre le traitement des données personnelles sur le territoire de l’union européenne.**

**Le contexte juridique s’adapte pour suivre l’évolution des technologies et de nos sociétés (usages accrus du numérique, développement d’un commerce en ligne).**

**Le RGPD s’inscrit dans la continuité de la loi française Informatique et Libertés de 1978 et renforce le contrôle par les citoyens de l’utilisation qui peut être faite des données les concernant. Il harmonise les règles en Europe en offrant un cadre juridique unique aux professionnels. Il permet de développer leurs activités numériques au sein de l’UE en se fondant sur la confiance des utilisateurs.**

**Qui est concerné par le RGPD ?**

**Tout organisme quels que soit sa taille, son pays d’implantation et son activité, peut être concerné. Le RGPD s’applique à toute organisation, publique ou privé, qui traite les données personnelles pour son compte ou non.**

**Dès lors qu’elle est établie sur le territoire de l’union Européenne, ou que son activité cible directement des résidents européens.**

**Pour avoir un site internet conforme au RGPD, cela passe par** **la modification de ses formulaires de contact**. **Tout d’abord, vous devez désormais récolter les informations de manière justifiée par rapport à l’objectif de récolte. Autrement dit, vous devez recueillir uniquement les informations nécessaires à l’atteinte de l’objectif mais pas plus.**

**Concernant la durée il faut donner le temps que les données seront stockées. En l’occurrence, pour avoir un site internet conforme au RGPD, vous ne pouvez pas les conserver indéfiniment**. **Plus exactement, les données marketing peuvent se conserver maximum 3 ans**

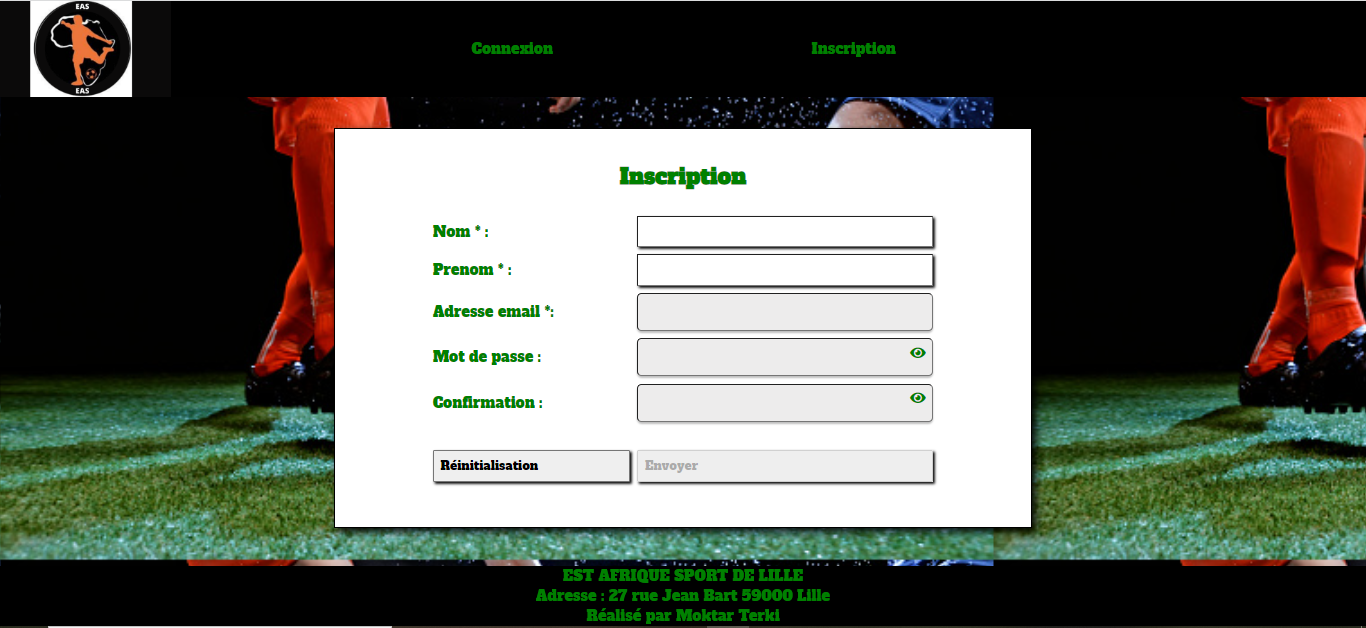
**En plus, vous devez demander le consentement de l’internaute par rapport à la récolte et à l’utilisation des données. Pour information, pour avoir un site internet conforme au RGPD, cela doit être un consentement volontaire. Pour cela, vous pouvez ajouter une case à cocher avant le bouton « envoyer » et avant le lien. Attention : pour avoir un site internet conforme au RGPD, cette case ne doit pas être pré-cochée**. **Et bien entendu, il faut accompagner cette case avec un texte approprié. Cela peut par exemple être : « J’ai lu et accepte la politique de confidentialité du site »**

### **L’utilisation de cookies**

**Ensuite, votre site peut utiliser des cookies. C’est par exemple le cas si vous utilisez Google Analytics. Dans ce cas, certaines modifications doivent être faites pour avoir un site internet conforme au RGPD. En l’occurrence, vous devez tout d’abord informer l’internaute de son suivi par un outil de tracking. En parallèle, vous devez préciser l’objectif de ce suivi. De plus, vous devez demander son consentement lors de sa première visite. A tout moment, vous devez également lui donner la possibilité de s’opposer au tracking. Et son opposition au tracking ne doit pas l’empêcher de naviguer normalement sur votre site.**

**Pour avoir un site web RGPD friendly, vous pouvez alors utiliser un bandeau. Autrement dit, un bandeau va apparaître pour informer le visiteur et demander son consentement.**

# Veille sécurité formulaire inscription



Fonction

 \* Vérifie que le pattern de l'input est respecté et que l'input n'est pas vide s’il est required.

 \* Change l'aspect des inputs en fonction de leur état

 \* Active ou pas le bouton submit

 \* @param {[object]} listInputs

 \* @param {[]} listInputsValidity

 \* @param {object} submit

 \* @param {object} formulaire

 \*/

function InputsCheckValidity(listInputs, listInputsValidity, submit, formulaire) {

    // Pour chaque input, on vérifie sa validité

    listInputs.forEach(element => {

        if (element.checkValidity()) {

            listInputsValidity[element.name] = true;

        } else {

            listInputsValidity[element.name] = false;

        }

    });

    // Vérification spéciale pour l'adresse mail à l'inscription

    verifEmail (listInputsValidity, formulaire);

    // Vérification spéciale pour le mot de passe

    verifPassword (listInputsValidity, formulaire);

    // Change l'aspect des inputs en fonction de leur état

    changeColor (listInputs, listInputsValidity);

    // Active ou pas le bouton submit

  revealSubmitButton(listInputsValidity, submit);

};

 \* Vérifie que l'adresse mail entrée à l'inscription n'existe pas

 \* @param {[]} listInputsValidity

 \*/

function verifEmail(listInputsValidity, formulaire) {

    let mailSaisi = formulaire.querySelector("#adresseMail");

    let infoMail = formulaire.querySelector("#infoEmail");

    if (mailSaisi != null && listemail.includes(mailSaisi.value)) { //modif

        listInputsValidity['adresseMail'] = false;

        infoMail.classList.remove('noDisplay');

    }

    else if (infoMail != null) {

        infoMail.classList.add('noDisplay');

    }

};

 \* Vérification du mot de passe et de la confirmation de mot de passe.

 \* @param {[]} listInputsValidity

 \* @param {object} formulaire

 \*/

function verifPassword(listInputsValidity, formulaire) {

    let listePassword = formulaire.querySelectorAll("input[type=password]");

    if (listePassword.length == 2 && listePassword[0].value != listePassword[1].value) { //modif

        listInputsValidity[listePassword[1].name] = false;

    }

};

# Spécifications techniques

## HTML

## 

## PHP

 <div *class*="user-inputs">

      <form *action*="Handler.php?task=write" *method*="POST">

        <div *class*="row">

<input *type*="text" *value*="<?php **echo**($\_SESSION['utilisateur']

->**getNom**(). " " . $\_SESSION['utilisateur']->**getPrenom**())?>" *disabled* *name*="auteur" *id*="auteur" *placeholder*="Pseudo ">

<input *type*="text" *id*="contenu" *name*="contenu" *placeholder*=" Message">

 </div>

<button *type*="submit" *id*="btnTchat">🔥 envoyer !</button>

<div><a *href*="index.php?page=Accueil"><button *class*="bouttonRetour" *type*="button"><i *class*="fas fa-sign-out-alt fa-rotate 180"></i></button></a></div>

*\* db = variable db Connexion simple à la base de données via PDO !*

*\*/*

$db = new **PDO**('mysql:host=localhost;dbname=easlille;charset=utf8', 'root', '', [

**PDO**::ATTR\_ERRMODE => **PDO**::ERRMODE\_EXCEPTION,

**PDO**::ATTR\_DEFAULT\_FETCH\_MODE => **PDO**::FETCH\_ASSOC

]);

Connexion à la base de données

Les paramètres requis sont les suivants : • host : adresse du serveur hébergeant la base de données (localhost ou votre serveur web) • dbname : nom de la base de données • charset : jeu de caractères utilisé • root : nom de l'utilisateur de la base de données, par exemple root • '' : le dernier argument précise le mot de passe ;

On doit analyser la demande faite via l'URL (GET) afin de déterminer si on souhaite récupérer les messages ou en écrire un.

*/\*\**

*\* On doit analyser la demande faite via l'URL (GET) afin de déterminer si on souhaite récupérer les messages ou en écrire un*

*\*/*

$task = "list";*// on creer une tache par defaut et on va l'appeler liste*

if(**array\_key\_exists**("task", $\_GET)){ *//si dans mon tableau get j'ai une cle qui s'appelle task*

  $task = $\_GET['task'];*//si on me presice une autre tâche alors on prend dans le get la task preciser*

}

if($task == "write"){ *//si la task = write*

**postMessage**();*//on appelle la fonction post message*

} else { *//*

**getMessages**();

}

On crée une tache par défaut et on va l'appeler liste

Si dans mon tableau get j'ai une clé qui s'appelle Task

Si on me précise une autre tâche alors on prend dans le get la Task préciser

Si la task = write

On appel la fonction post message Sinon dans tous les autres cas on fait un get message

*/\*\**

*\* Si on veut récupérer,  il faut envoyer du JSON*

*\*/*

function **getMessages**(){ *//function get message*

  global $db;*// pour dire que db est disponible a l'exterieur de la fonction*

*//  On requête la base de données pour sortir les 20 derniers messages*

  $resultats = $db->**query**("SELECT \* FROM messages ORDER BY Create\_at DESC LIMIT 20"); *//resultat = (db)la base de donnee*

*//et je créer une requête  qui sélectionne toute les données par descendant par contre je les limit a 20 par ordre descendant*

*//  On traite les résultats*

  $messages = $resultats->**fetchAll**();*// on crée une variable "messages" on va appliquer sur le résultat une méthode fetchAll*

*//fetchAll récupére toute les lignes de données et me les renvoyer sous forme de tableau dans la variable message*

*// On affiche les données sous format JSON*

**echo** **json\_encode**($messages); }

Fonction get message

Pour dire que db est disponible à l'extérieur de la fonction

On requête la base de données pour sortir les 20 derniers messages

Résultat = (db)la base de données

Je créer une requête qui sélectionne toutes les données par descendant par contre je les limite à 20 par ordre descendant

On traite les résultats

On crée une variable "messages" on va appliquer sur le résultat une méthode fetchAll

FetchAll récupère toutes les lignes de données et me les renvoyer sous forme de tableau dans la variable message

On affiche les données sous format JSON

*/\*\**

*\* Si on veut écrire au contraire, il faut analyser les paramètres envoyés en POST et les sauver dans la base de données*

*\*/*

function **postMessage**(){*//*

  global $db;

*//  Analyser les paramètres passés en POST (parametre: auteur, contenu)*

  if(!**array\_key\_exists**('auteur', $\_POST) || !**array\_key\_exists**('contenu', $\_POST)){

**echo** **json\_encode**(["status" => "error", "message" => "One field or many have not been sent"]);

    return;

  }

Si on veut écrire au contraire, il faut analyser les paramètres envoyés en POST et les sauver dans la base de données

Analyser les paramètres passés en POST (paramètre : auteur, contenu)

  $auteur = $\_POST['auteur']; *//on créer une variable auteur qui va être egal a post " auteur"*

  $contenu = $\_POST['contenu'];

On crée une variable auteur qui va être égal a post " auteur"

*// Créer une requête qui permettra d'insérer ces données*

  $query = $db->**prepare**('INSERT INTO messages SET Auteur = :auteur, Contenu = :contenu, Create\_at = NOW()');

*//on lui demande de preparer une requete d'insertion*

*//on lui dit que auteur est egale a une variable qui s'appel auteur, contenu qui sera egale a la variable contenu qu'on a declarer en bas et*

*// create at egale a la fonction msq now qui nous donnera la date et l'heure*

  $query->**execute**([*// execute la requete qui sera sous forme de tableau qui va lui donner les valeurs de auteur et contenu*

    "auteur" => $auteur, *//on veut que la variable auteur soit egal a la vrai variable php (ce qui est contenu dans la requette post)*

    "contenu" => $contenu

  ]);

*// Donner un statut de succes ou d'erreur au format JSON*

**echo** **json\_encode**(["status" => "success"]);

}

On crée une requête qui permettra d'insérer ces données.

On lui demande de préparer une requête d'insertion

 On lui dit qu’auteur est égale a une variable qui s'appelle auteur, contenu qui sera égale à la variable contenu qu'on a déclaré en bas et create at égale a la fonction MySql Now qui nous donnera la date et l'heure.

Execute la requête qui sera sous forme de tableau qui va lui donner les valeurs de auteur et contenu. On veut que la variable auteur soit égal a la vrai variable PHP (ce qui est contenu dans la requête post)

Donner un statut de succès ou d'erreur au format JSON

## JS

var btn = document.**getElementById**("btnTchat");

*//variable qui sert a faire un appel pour recuperer le bouton*

*/\*\**

Il nous faut une fonction pour récupérer le JSON des messages et les afficher correctement.

*\* Il nous faut une fonction pour récupérer le JSON des*

*\* messages et les afficher correctement*

*\*/*

function **getMessages**(){ *//fonction get message qui contacte le serveur et recupere*

*// les 20 derniers messages les traite et les affiche*

*// Elle doit créer une requête AJAX pour se connecter au serveur, et notamment au fichier handler.php*

  const requeteAjax = new **XMLHttpRequest**(); *//on creer une variable requete*

  requeteAjax.**open**("GET", "PHP/VIEW/GENERAL/Handler.php");*//quand on envoi la requetteAjax elle ira*

*//dans le get" handler.php"*

Fonction get message qui contacte le serveur et récupère les 20 derniers messages les traite et les affiche. Elle doit créer une requête AJAX pour se connecter au serveur, et notamment au fichier handler.php.

On crée une variable requête

Quand on envoi la requête Ajax elle ira dans le get" handler.php".

*// 2. Quand elle reçoit les données, il faut qu'elle les traite (en exploitant le JSON)*

*//et il faut qu'elle affiche ces données au format HTML*

  requeteAjax.**onload** = function(){*//quand elle aura changer la reponse du serveur*

    const resultat = JSON.**parse**(requeteAjax.responseText);

*//mettre dans une variable résultat (on le parse en json (n'est plus une chaine de caractère*

*//mais des objets )et on a la réponse de ce que le serveur a répondu (response text)*

*//affichage html*

    const html = resultat.**reverse**().**map**(function(message){ *//reverse =inverse le tableau objet*

*//resultat(tableau qui contient des objets)*

      return `

        <div class="message">

          <span class="date"> ${message.Create\_at.**substring**(11, 16)}</span>

          <span class="auteur"> ${message.Auteur}</span> :

          <span class="contenu">${message.Contenu}</span>

        </div>

      `

    }).**join**('');*//tu vas coller tous les éléments du tableau et en faire une grande chaine de caractère (html)*

    const messages = document.**querySelector**('.messages');*//*

    messages.innerHTML = html;

    messages.scrollTop = messages.scrollHeight;

  }

*// 3. On envoie la requête*

  requeteAjax.**send**();

}

 Quand elle reçoit les données, il faut qu'elle les traite (en exploitant le JSON) et il faut qu'elle affiche ces données au format HTML

Quand elle aura changé la réponse du serveur

 Mettre dans une variable résultat (on le parse en Json (n'est plus une chaine de caractère mais des objets) et on a la réponse de ce que le serveur a répondu (response text)

Reverse =inverse le tableau objet

Résultat (tableau qui contient des objets.

On va coller tous les éléments du tableau et en faire une grande chaine de caractère (html)

Ensuite on envoi la requête

*/\*\**

*\* Il nous faut une fonction post message  pour envoyer le nouveau*

*\* message au serveur et rafraichir les messages dans la base de données*

*\*/*

function **postMessage**(event){

*// Elle doit stopper le submit du formulaire*

  event.**preventDefault**();*//l'évènement doit s'arrêter*

*}*

Il nous faut une fonction post message pour envoyer le nouveau message au serveur et rafraichir les messages dans la base de données.

Elle doit stoper le submit du formulaire

L'évènement doit s'arrêter

*// Elle doit récupérer les données du formulaire*

  const auteur = document.**querySelector**('#auteur');*//constante auteur (champ auteur)*

  const contenu = document.**querySelector**('#contenu');*//constante contenu*

On déclare les constantes et on récupère toute les div de la page avec l’élément auteur et contenu

Elle doit conditionner les données

*// Elle doit conditionner les données*

 const data = new **FormData**();*// variable data = nouvel objet formdata*

  data.**append**('auteur', auteur.value);*//je cree un ensemble de donnee*

*// on ajoute une  données qui s'appel  auteur et qui va contenir mon champs auteur.value*

  data.**append**('contenu', contenu.value);

Variable data = nouvel objet form data

  requeteAjax.**onload** = function(){*//*

    contenu.value = '';*//contenu.valu = vide*

    contenu.**focus**();*//contenu mettre la souri dedans*

**getMessages**();*//reaficher les messages*

  }

  requeteAjax.**send**(data);*//on envoi les données recueilli dans le formulaire*

*//Ajoute l'attribut disable au boutton pour le rendre non cliquable*

On veut envoyer la requête en post vers handler.php?task=write (qui veut dire ne pas recevoir les messages et bien les envoyer).

contenu.valu = vide

Contenu : mettre la souri dedans.

Réafficher les messages.

On envoie les données recueillies dans le formulaire

  requeteAjax.**send**(data);*//on envoi les données recueilli dans le formulaire*

*//Ajoute l'attribut disable au boutton pour le rendre non cliquable*

  btn.**setAttribute**("disabled",true);

*//Attent 5sec et lance la fonction protect*

**setInterval**(**protect**, 5000);

}

function **protect**() {

*//Supprimer l'attribut du boutton pour permettre d'envoyer a nouveau des messages*

  btn.**removeAttribute**("disabled", false);

}

On ajoute l’attribut disable au button submit pour le rendre non cliquable.

On attend 5 secondes au bout de ces 5 secondes on fait appel a la fonction Protect pour le rendre accessible, donc l’envoi de message est possible.

Après l’envoie du message le submit redeviens disable.

Sélectionne le formulaire sur la page et j'ajoute un écouteur d'évènement sur l'évènement submit. On va appeler la fonction post message

document.**querySelector**('form').**addEventListener**('submit', **postMessage**);*// selectionne le formulaire sur la page*

*// et j'ajoute un ecouteur d'evenement sur l'evenement submit tu vas appeler la fonction post message*

*/\*\**

*\* Il nous faut une intervale qui demande le rafraichissement*

*\* des messages toutes les 3 secondes et qui donne*

*\* l'illusion du temps réel.*

*\*/*

const interval = window.**setInterval**(**getMessages**, 3000);*//on creer une interval*

*// ou tu vas appeler la fonction get message toute les 3000 milli secondes (3s)*

**getMessages**();*//affiche les message direct*

Il nous faut une intervalle qui demande le rafraichissement des messages toutes les 3 secondes et qui donne l'illusion du temps réel.

On crée un intervalle ou on va appeler la fonction get message toutes les 3000 milli secondes (3s) On affiche le message directement

# CSHARP application bureau



Une image contenant table

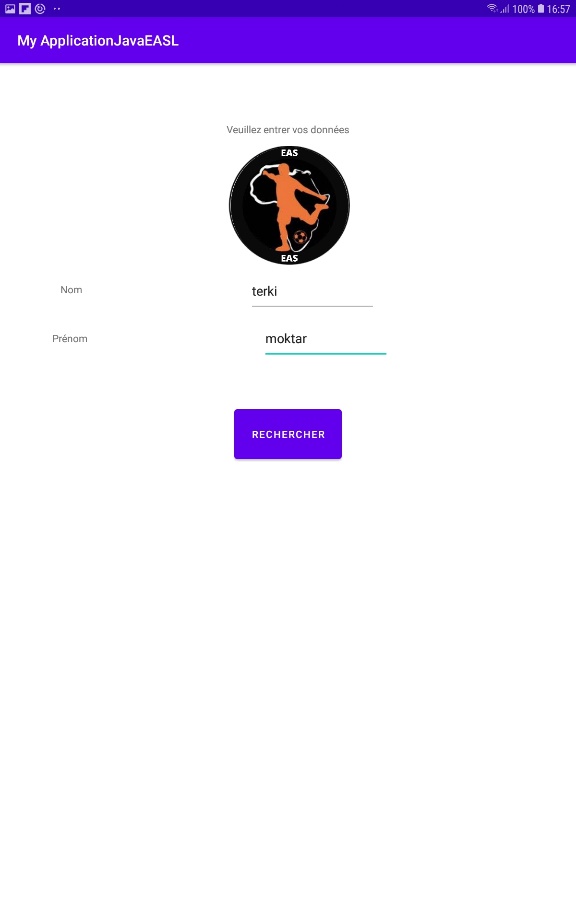
Description générée automatiquement







# JAVA (appli mobile)

Une image contenant texte

Description générée automatiquement

# Recherche site anglophone

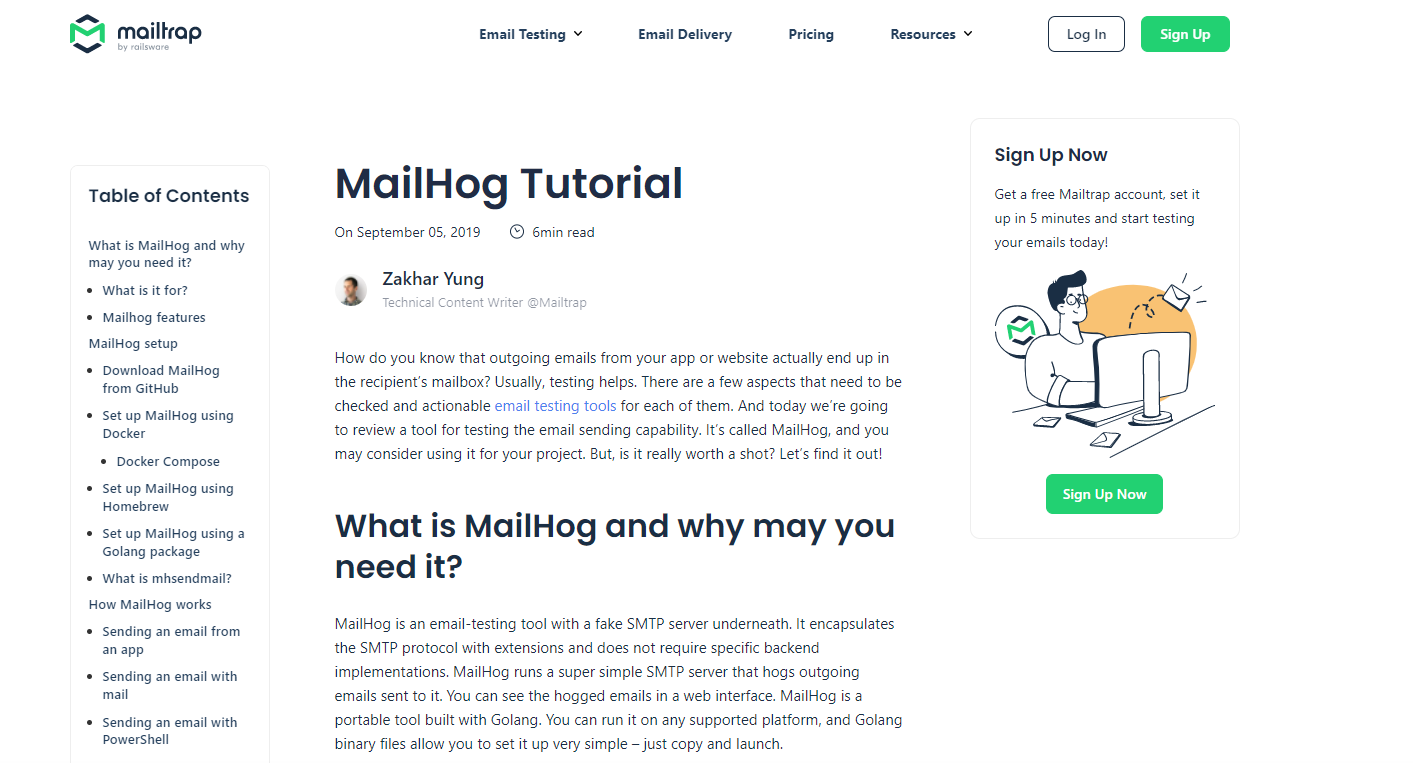
## Problème rencontrer

J’ai eu un problème pour envoyer mon formulaire contact, ne savant pas comment fonctionne l’envoi de mail.

Je voulais utiliser la fonction **mail()** en lui passant en argumants :

* *l’email du destinataire*
* *le sujet du courrier*
* *le texte du courrier*

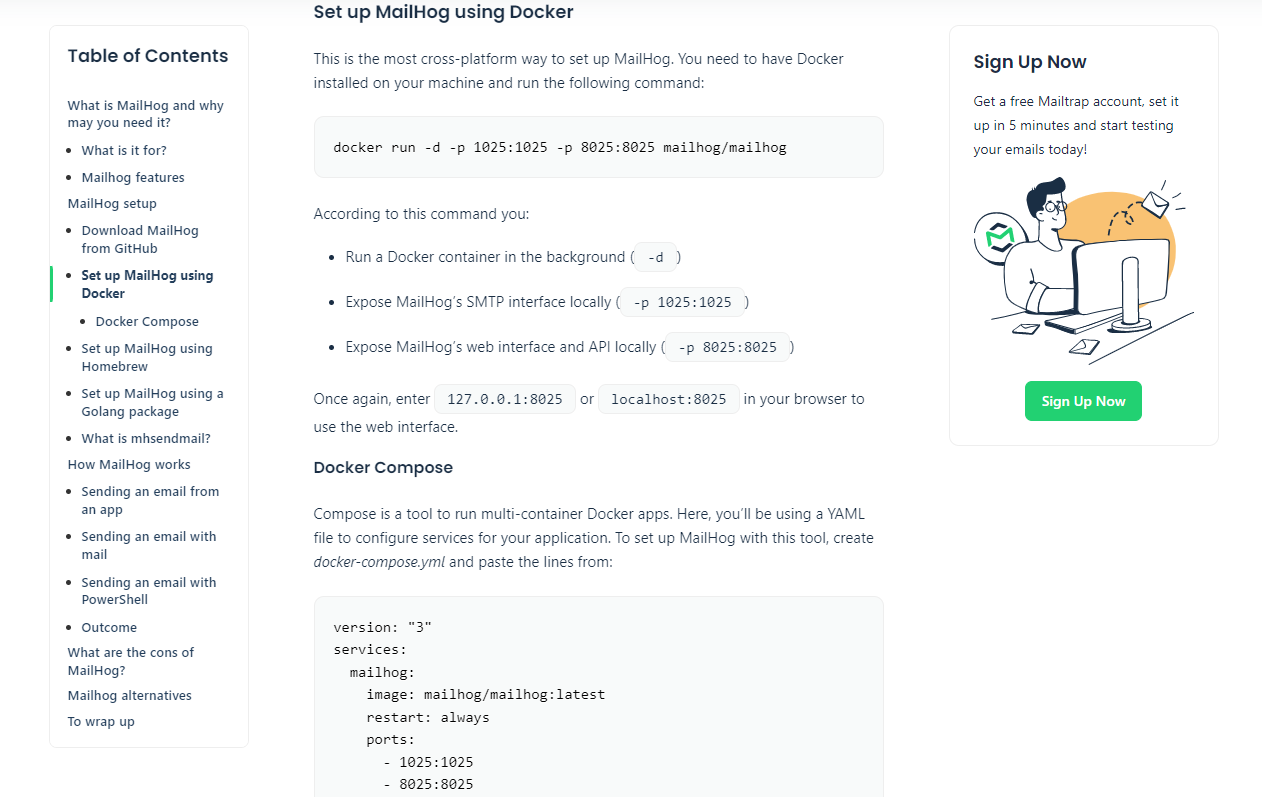
Après plusieurs recherche j’ai mis en place la solution ci-dessous.

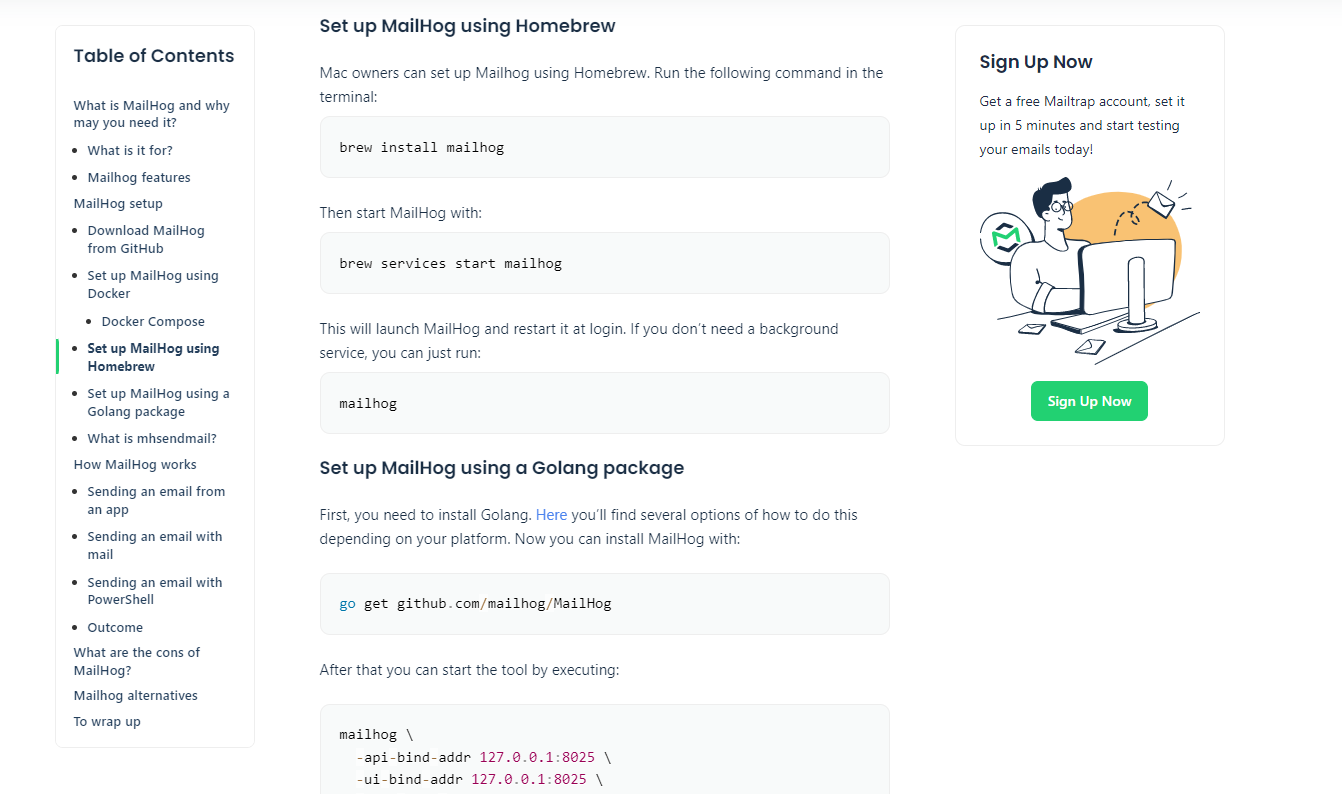


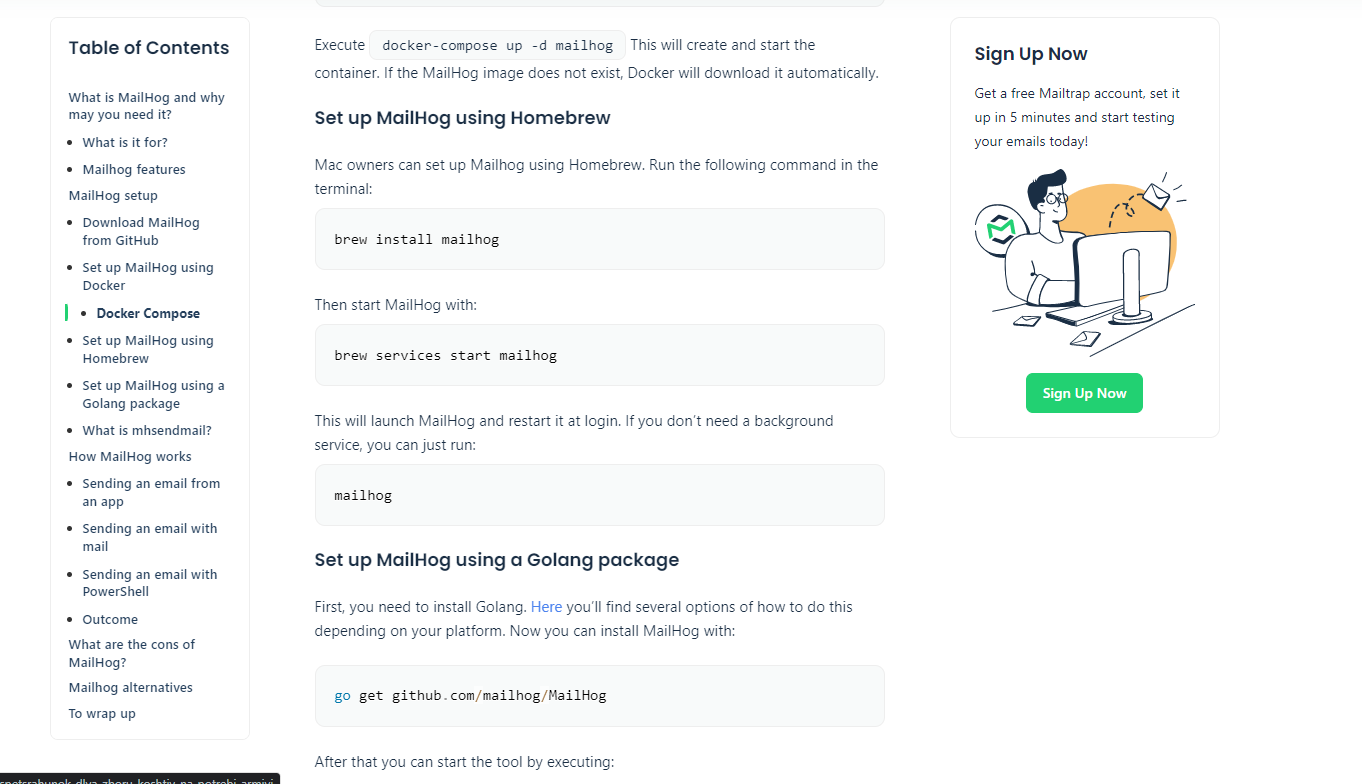
Une image contenant texte

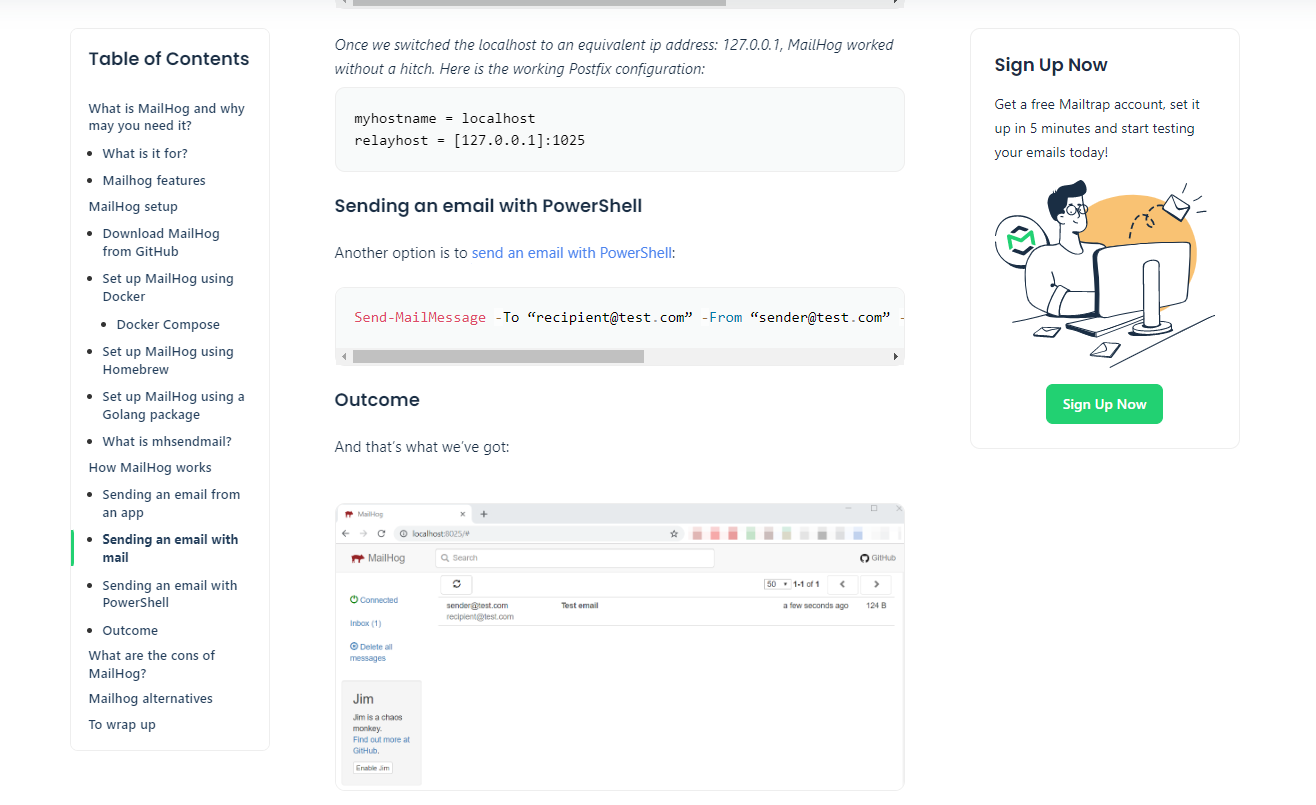
Description générée automatiquement



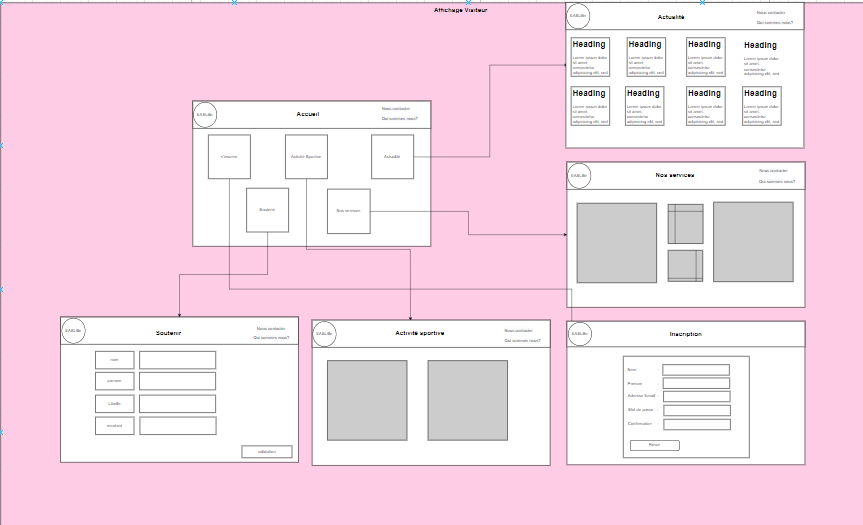


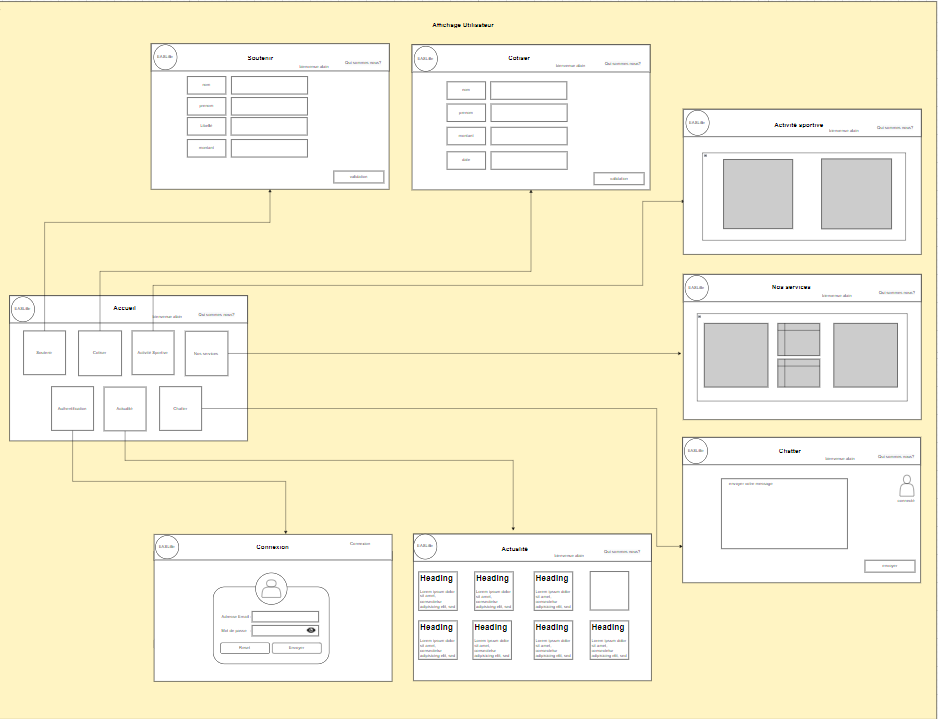


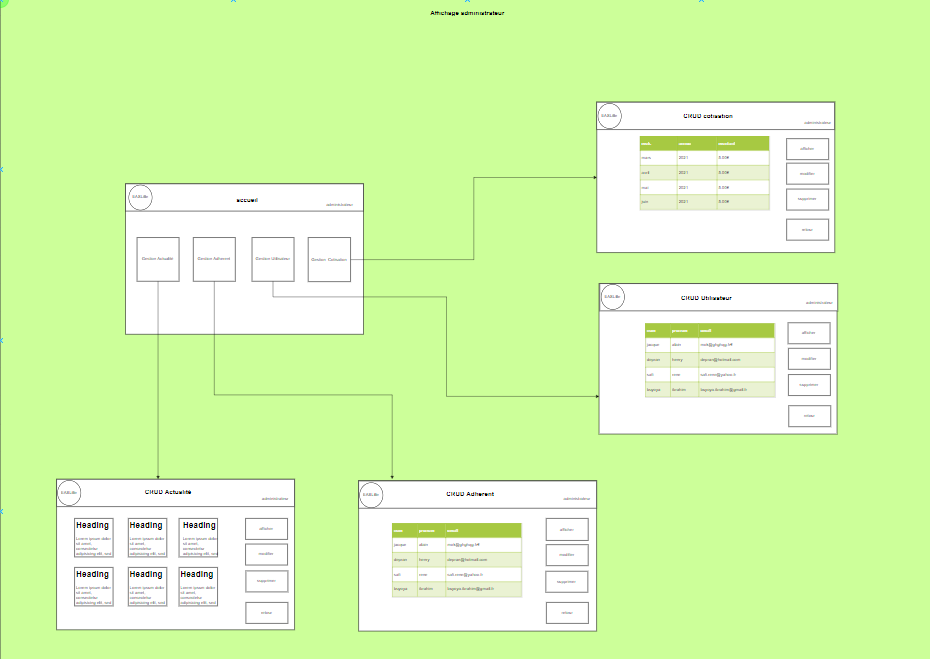




# Annexes Diagramme de flux



****



# 

# Veille technologique

**La veille en sécurité informatique est une activité indispensable, pour tous les professionnels. Elle vous permet de rester au fait des évolutions et tendances.**

**Mais, plus important, la veille vous donne la capacité d’anticiper les attaques informatiques et mieux vous préparer et donc de limiter le risque d’un incident. Tous les professionnels de l’informatique le savent… Pourtant selon un sondage que nous avons réalisé, peu sont les professionnels qui prennent/ont le temps de faire une veille active efficace.**

## **La veille sécurité pour protéger votre S.I**

**À la suite du**[**Mois Européen de la cybersécurité**](https://www.ssi.gouv.fr/agence/cybersecurite/cybermois-2020/)**, mois dédié à la sensibilisation, on avait envie de s’adresser aux professionnels et parler d’un sujet trop souvent délaissé, la veille cybersécurité.**

**Les technologies sont en évolutions permanentes, les attaques et les vulnérabilités également. C’est un vrai jeu du chat et de la souris que se livrent pirates et défenseurs.**

**Chaque semaine, des vulnérabilités sont découvertes, des entreprises sont victimes de piratages, entraînant des dommages importants, voire irrémédiables.**

**Pourtant malgré le buzz autour des**[**failles Zéro Day**](https://www.torii-security.fr/blog/meltdown-et-spectre-decryptage-des-deux-failles-critiques)**ou autres APT, c’est bien souvent les failles connues mais non corrigées qui sont exploités par les pirates… Les pirates profitent, entre autres, du temps entre la publication d’un correctif et le moment où vous appliquerez le patch pour agir.**

**Autant dire qu’il est indispensable :**

* **De se tenir au courant des vulnérabilités et des patchs,**
* **Des tendances d’attaques des pirates.**

**La veille en sécurité informatique, c’est donc l’outil indispensable pour s’éviter des frayeurs**.

## **Une veille délaissée par les pros**

**Selon notre sondage réalisé cet été, seul 27% d’entre vous font une veille sécurité quotidienne et 50 % pratiquent uniquement à l’occasion.**

**Le temps consacré à la veille, lui, est également faible au vu des enjeux: entre 10 et 30 minutes maximum pour 45 % d’entre vous.**

**Autre apprentissage, c’est les réseaux sociaux que vous préférez pour vous tenir au courant, comptant sur la viralité des infos pour faire émerger les infos importantes, et donc, optimiser le temps passer à faire de la veille.**

**Manque de temps, de compétences, barrière de la langue, les freins peuvent être nombreux, mais le constat est sans appel : Vous avez été 71% à déclarer que votre veille était inefficace.**

**Voici notre infographie regroupant vos pratiques de veille cybersécurité.**

# Une image contenant texte Description générée automatiquement

# 

# Déploiement de l’application

# 

## Accès au serveur :

Serveur de fichiers, serveur de base de données

Attendu : Les serveurs sont opérationnels, les droits sont correctement gérés.

**Obtenir les identifiants de connexion au serveur :**

Attendu : Avoir des identifiants suffisant pour travailler et pour faire tourner l'application.

**Obtenir les outils pour accéder au serveur :**

Installer Filezilla :

Attendu : Logiciel installé et paramétrage mis à jour.

**Vérifier les accès :**

Serveur de fichiers

Attendu : Connection et modification possible.

Serveur de base de données

Attendu : Connection et modification possible.

## Installation des fichiers :

**Déplacer les fichiers sur le serveur :**

Repérer les fichiers nécessaires sur le serveur

Attendu : Code nettoyé (plus de code en commentaire, plus de var dump,

Attendu : Fichiers inutiles effacés.

Transfert des fichiers de projet.

Modification des fichiers de projet.

Attendu : Les fichiers de paramétrage et de gestion des encryptages sont mis à jour.

## Installation de la base de données :

**Préparer les scripts :**

Préparer les scripts pour le nouvel environnement.

Epurer les scripts et organiser leur passage.

Passer les scripts sur le serveur.

Attendu : La base de données est créée et contient les données de départ de l'application.

## Test de déploiement :

**Test de fichiers :**

Lancer l'application et vérifier que les fichiers sont bien reconnus et placer au bon endroit.

Vérifier la casse des fichiers.

**Test de base de données :**

Vérifier l'accès aux données.

Vérifier la possibilité de modifier les données.

# Conclusion stage

Cette période de stage que j’ai effectué a été très enrichissante pour moi car cela a été l’occasion de découvrir dans le détail le secteur de l’informatique. Cela m’a aussi conforté dans ma conversion professionnelle.

Sur l’aspect professionnel cela m’a également servi à m’améliorer sur le travail en autonomie, en partant de la recherche jusqu’à la réalisation du projet qui m’a été confié.

Grace à ce stage j’ai pu approfondir mes connaissances sur le développement ainsi que la conception.

Sur un aspect personnel, ce stage m’a permis de mieux cerner mes aptitudes et de gagner en confiance. Cela m’a apporté comme bénéfice une meilleure gestion des situations quel que soit les circonstances.

Au-delà d’enrichir mes connaissances en développement, ce stage m’a aussi permis d’accroitre mon sens du travail en équipe ainsi que les techniques de communications écrites et orales. Il me semble essentiel de savoir communiquer dans ce domaine ou le travail en équipe est essentiel.

Je souhaiterais pour finir remercier toutes les personnes qui m’ont accompagné durant ce stage, qui m’ont fait confiance ainsi que ceux qui ont contribué au succès de mon stage.

# Conclusion formation

Cette période de formation a été très enrichissante pour moi car cela a été l’occasion de découvrir dans le détail le secteur de l’informatique. Cela m’a aussi conforté dans ma conversion professionnelle.

Votre enseignement de qualité m’a permis de développer mes connaissances compétences acquises et de découvrir de nouvelle notion.

J’ambitionne d’exercer le métier passionnant de développeur concepteur d’applications et la professionnalisation de cette formation représente un précieux atout pour une réinsertion réussie dans l’univers du travail.

La formation proposée par l’AFPA a été pour moi une aubaine pour me former aux métiers du numérique. Le centre de formation de dunkerque se distingue par :

* La qualité globale de la formation.
* Les conditions générales de déroulement de formation (locaux, hébergement, accueil, etc.).
* La qualité de la prestation de la formatrice Martine POIX.
* La qualité des méthodes, des supports pédagogiques utilisés et de la documentation remise.
* La conformité aux objectifs :
* L’adéquation de la formation à leurs besoins à leurs attentes.

Enfin je tiens à remercier le conseil régional ainsi que la région haut de France pour le financement de la formation.

Je remercie aussi tous les acteurs des différents services du centre de formation Afpa de dunkerque pour leur accompagnement durant cette période de formation réussi.

# Glossaire

**PHP** : (PHP HyperText Prepocessor) est un langage de scripts généraliste et open source, spécialement conçu pour le développement d’applications web. Il peut être facilement intégrer au html.

**HTML** : ( HyperText Markup Language) est le langage de balisage conçu pour représenter les pages web. C’est un langage permettant d’écrire de l’hypertexte.

**CSS** : (Cascading Style Sheets) est un langage informatique utilisé sur internet pour mettre en forme les fichier html ou xml. Ainsi les feuilles de style, aussi appelé les fichiers CSS, comprennent du code qui permet de gérer le design d’une page web.

**Template** : c’est un moyen de séparer le contenu, rédactionnel (contenu textuel) de la forme (la manière dont il est présenté). Un Template fait donc office de gabarit (modèle)ou seuls certains éléments sont modifiables (le contenu texte, les images, les couleurs et le fond).

**Cookies** : c’est un fichier qui est dépose par votre navigateur sur votre ordinateur lorsque vous surfez sur internet. Ce fichier est composé uniquement de texte.

**Encapsulation** : principe de l’orienté objet le fait de regrouper en un même objet, les attributs et les méthodes.

**Polymorphisme** : le fait de pouvoir avoir 2 méthodes qui portent le même nom, mais qui ont des signatures différentes (paramètre d’entrée et paramètre de sortie).

**Framework** : ensemble d’outils qui permettent une génération du code plus rapide.

**IDE** : environnement de développement qui peut intégrer un éditeur de texte, des compilateurs, des analyses de code, etc…

**MVC** : Modèle (lien à la base de données) – Vue (partie affichage de l’application) - Contrôleur (classe métier).

**Méthode agile** : méthode de travail optimisé qui permet de répartir efficacement les taches dans une équipe, livraison rapprochée en étroite collaboration avec le client.

**FTP** : File Transfer Protocol (protocole de transfert de fichier), ou FTP, est**un protocole de communication destiné au partage de fichiers sur un réseau TCP/IP.**

**Regex : Regular Expression (qui va obliger une certaine syntaxe sur un champ donné).**

**POO : programmation orienté objet** est une méthode de programmation dans laquelle les programmes sont organisés comme des ensembles d'objets. Chaque objet représente une instance d'une certaine classe, toutes les classes étant des membres d'une hiérarchie de classes unifiée par des relations d'héritage.

**PDO :** (PHP Data Object) est une extension PHP qui définit une interface d'accès à une base de données.

**API** : Application Programming Interface permet le transfert de donné d’une application à une autre, ou à elle-même.

**DAO** : **Data Access Object**.

**Concaténation** : Mettre bout à bout deux ou plusieurs chaines de caractère, pour en former une nouvelle.

**Objet** : instanciation de classe.

**Classe** : instancié par l’objet.

**CMS** : Content Management System est une solution de gestion de contenu, un système préprogrammé pour générer les applications.

**Json** : Java Objet Notation est un format de donnée.

**Clé primaire** : champ d’une table qui permet d’identifier de manière unique un enregistrement de la table (plusieurs champs pour une clé primaire).

**Clé secondaire** : résultat de la migration de la clé d’une clé primaire d’une table vers une autre table en vue d’établir une relation entre ses deux tables.

**Index** : référentiel de données d’une colonne ayant pour but d’améliorer la recherche des données.

**Vue** : table virtuelle permettant de regrouper plusieurs tables afin de faciliter l’écriture des requêtes.