**MOHAMMED NABI / DEVELOPPEUR WEB ET WEB MOBILE 1**

**RAPPORT DE STAGE**

**Rédigé par : Mohammed NABI**

2021 / 2022 MOHAMMED NABI / DEVELOPPEUR WEB ET WEB MOBILE 2

Table des matières

[*Remerciements* 1](#_Toc107299645)

[**1** **INTRODUCTION GENERALE** 2](#_Toc107299646)

[**2** **Présentation du ‘’Centre Guinot’’ :** 5](#_Toc107299647)

[**3** **Présentation de l’établissement ‘’Bioinformatique’’ :** 5](#_Toc107299648)

[**4** **Introduction sur mon parcours :** 6](#_Toc107299649)

[**5** **Liste des compétences du référentiel :** 7](#_Toc107299650)

[**6** **Guide d’emploi de l’application :** 8](#_Toc107299651)

[**7** **Projet de l’organisme de formation BIOGESOFT :** 9](#_Toc107299652)

[**8** **Présentation du projet :** 10](#_Toc107299653)

[**9** **Cahier des charges :** 12](#_Toc107299654)

[**10** **Techniques à adopter :** 18](#_Toc107299655)

[**11** **Analyse et conception :** 20](#_Toc107299656)

[**11.1** **Le diagramme de classes :** 24](#_Toc107299657)

[**11.2** **Le diagramme de cas d’utilisation :** 27](#_Toc107299658)

[**11.3** **Description des cas d’utilisations :** 28](#_Toc107299659)

[**12** **Choix des technologies :** 35](#_Toc107299660)

[**13** **Planning de stage  :** 39](#_Toc107299661)

[**14** **Base de données et jeux de données Sur Symfony :** 40](#_Toc107299662)

[**15** **Documentation en Anglais :** 49](#_Toc107299663)

[**15.1** **Mot du référent du stage :** 51](#_Toc107299664)

[**15.2** **Conclusion :** 52](#_Toc107299665)

# *Remerciements*

*Tout d'abord, c’est à Monsieur* ***Akli SAIDANI,*** *mon tuteur de stage et directeur des études, que va toute ma reconnaissance, pour m'avoir orienté et informé du déroulement du stage et des démarches à entreprendre, son assistance a été capitale.*

*Mon maître de stage, Monsieur* ***Igor PALAKOT****, a été d’une patience exemplaire et apaisante. Je tiens à le remercier vivement pour son accueil, le temps passé ensemble et le partage de son expertise au quotidien. Grâce également à sa confiance, j'ai pu m'accomplir totalement dans mes missions. Il fut d'une aide précieuse dans les moments les plus délicats.*

*Mes sincères remerciements vont à l’ensemble du personnel du centre de formation* ***Guinot****, pour leur accueil, collaboration et assistance.*

*Mes vifs remerciements à Monsieur* ***Gilles LEVY****, mon professeur d’informatique, qui m'a beaucoup aidé dans ma formation, qui m'a permis d’acquérir des connaissances fondamentales dans le domaine et d’avoir de la motivation pour mon futur projet professionnel.*

*Je n’oublie pas non plus ma famille et mes proches, qui m’ont sans cesse soutenu, encouragé et motivé, dans la réalisation et l’élaboration de mon projet professionnel et m’ont aidé à chaque étape de ce rapport de stage.*

*Mes remerciements spéciaux vont à mon relecteur et correcteur qui a contribué, grâce à ses conseils et recommandations, à l’élaboration et au bon déroulé de mon rapport de stage.*

*Ce stage m’a permis d’affiner certaines pistes pour bâtir mon Projet d’Orientation Professionnel et signe l’aboutissement de mon cursus de développeur web et mobile.*

# **INTRODUCTION GENERALE**

Dans le cadre de ma deuxième année de formation de développeur web et mobile, j’ai réalisé mon stage dans une entreprise, dont les principales caractéristiques sont en adéquation avec mon projet professionnel.

Ce stage vise à clôturer mon cursus. Il me permet d’être formé au sein d’une entreprise dans le but d’acquérir des connaissances sur un secteur d’activité, tout en me permettant de mettre en pratique les connaissances théoriques, que j’ai acquises lors de mes années d’études.

En première étape, n’ayant pas réussi à trouver un stage au sein d’une entreprise pour des raisons administratives et vue le contexte actuel du Covid-19. Mon référent, **Igor PALAKOT**, m’a suggéré de faire un stage en interne, tout en attendant que mes recherches de stage aboutissent. Pour cela, l’objectif a été de développer, une application web de messagerie privé pour les stagiaires du centre de formation et de rééducation professionnelle pour Malvoyants « **Paul GUINOT** », qui ont fréquenté l’établissement lui-même.

En deuxième étape, après la poursuite, en parallèle, de mes recherches de stage et grâce à mes efforts acharnés, qui ont finalement abouti à l’obtention d’une offre de stage au sein de l’entreprise « **BIOGESOFT** », avec en prime une promesse d’embauche dans le cas de l’accomplissement des objectifs visés. Compte tenu que la durée de stage restante, dont je disposais, était courte et après concertation, nous nous sommes entendus sur la réalisation d’une extension du projet déjà existant. Ma mission a été de mettre en place un système d’inscription des candidats dans différentes formations proposées par BIOGESOFT.

Mon stage, en entreprise, a débuté par une phase d'intégration, présentation des employés de l'entreprise, présentation du projet, auquel je participe actuellement, et installation des logiciels nécessaires au développement de mon projet.

A travers le présent rapport de stage, je vais donc vous présenter les différentes étapes de la réalisation du projet de création de l’une application web qui m’a été confié, non pas uniquement dans le cadre du stage pratique mais devant être réellement mise en ligne et exploitée professionnellement par mon entreprise d’accueil.

Pour commencer, la première partie du rapport de stage concernera la présentation de mon cadre de travail (thème du projet et cahier des charges), avec la description du projet et de ses objectifs, ensuite les méthodes de travail utilisées et enfin le choix des solutions techniques exploitées lors de la phase de développement. En deuxième partie, il reprendra le développement et ses sections correspondantes. Ensuite, pour les deux dernières parties, ledit rapport portera sur l’administration et l’hébergement du site web développé. Enfin pour conclure, le Bilan et mon ressenti sur ce stage ainsi que ses apports sur les plans personnel et professionnel.

Cette plateforme en ligne, offre un espace de rencontre, de souvenirs, d’apprentissage, d’échanges et de partages d’expériences professionnelles entre les anciens stagiaires qui ont fréquenté l’établissement et les nouveaux stagiaires qui le découvrent. Une messagerie interne, pour que les stagiaires puissent échanger leur savoir et renforcer leurs acquis en programmations, y est incluse.

L’application ‘STAGIAIRES-GUINOT’, établit en autre, un véritable lien générationnel, une continuité culturelle et éducative, riche en conseils, en expériences, principalement dans le domaine lié à l’accessibilité.

En raison du contexte actuel, le centre m’a proposé un équipement informatique pour que je puisse me mettre en télétravail. Pour des raisons de confort d’utilisation et de gain de temps, j’ai préféré travailler, en utilisant mon propre matériel, tout en intégrant les outils collaboratifs du centre de formation.

Monsieur Igor, mon formateur au CRP Paul Guinot, a toujours été disposé à répondre à toutes mes interrogations en termes de développement. Il a souvent été présent également pour débloquer des situations très complexes dans lesquelles je me trouvais parfois.

Je poursuivis en parallèles ma recherche de stage, en espérant trouver une offre pour enrichir mes connaissances en milieu externe au sein d’une entreprise. Mes efforts acharnés ont finalement abouti à un projet de stage, avec en prime une promesse d’embauche.

1. Projet Externe :

Module : Formation ‘BIOInformatique.com’.

La première possibilité consistait à démarrer un nouveau projet, en solo, en partant d’un simple descriptif et d’un fichier au format ‘Excel’ pour avoir une idée sur l’affichage souhaité.

La seconde option était de prendre part à un projet en cours.

Mon choix, pour la première option, était motivé par une crainte en termes de temps car je ne disposais que d’une période courte. En effet, le risque de perdre beaucoup de temps à concevoir un nouveau projet me semblait très élevé.

L’étape de développement a nécessité d’effectuer quelques manipulations afin de me permettre de personnaliser mon application, en y intégrant un nouveau module. En effet, ma mission au sein de Bioinformatique était d’intégrer, à mon projet en cours, le module formation qui consiste à proposer des cours en ligne, sur des langages de programmation, tels que (le PHP, JAVA, PYTHON, …).

**- Module Formations :**

Le module ‘’Formations’’ offre toute une gamme de formations en ligne dans le domaine du numérique.

Chaque formation débouche sur une liste de langages de programmation tels que le PHP, JAVA, PYTHON…

Le futur stagiaire doit choisir le langage de sa formation.

Chaque langage propose une liste de cours en ligne.

Une liste de cours en ligne est proposée avec des cases à cocher.

Une fois le stagiaire identifié et le choix validé, il peut accéder à la formation qu’il a choisie et commencer son premier cours en ligne.

**Remarque :**

Le cours proposé se compose de trois formes :

* Cours en fichier PDF ou Word ;
* Cours en fichier MP4 (Vidéo) ;
* Un quiz.

La première semaine était consacrée à la lecture des documents du projet. Une réunion hebdomadaire se tenait chaque lundi. Cette dernière avait pour objectif de faire le point sur le projet en cours.

Il s’agit de prendre la parole pour relever d’éventuelles difficultés ou pour demander des précisions sur le projet, auprès de mon référant, au sujet du développement de telle ou telle fonctionnalité. Cela avait été une occasion, pour moi, de découvrir le fonctionnement de ces réunions.

En échangeant régulièrement avec le tuteur de stage, j’ai pu nourrir ma propre réflexion en termes d’identification des besoins. Cela a abouti à la rédaction du cahier de charges, lequel avait été validé pour me permettre de commencer la modélisation des données.

Dans un second temps, il fallait configurer un dépôt distant, en utilisant GitHub.

Des détails sur cet outil sont à consulter dans ce document, dans la rubrique « Outils ».

# **Présentation du ‘’Centre Guinot’’ :**

L’Etablissement de Réadaptation Professionnelle ‘’Paul et Liliane Guinot’’ forme des personnes handicapées visuelles, en vue d’une intégration professionnelle en milieu ordinaire de travail, par le biais d’une formation professionnelle et d’un accompagnement social en milieu spécialisé.

Le centre accueille chaque année 125 personnes aveugles et malvoyantes, en primo-formation ou en reclassement professionnel.

Les personnes en situation d’handicap visuel et qui ont une orientation professionnelle peuvent s’inscrire à une des formations proposées.

**Formations proposées :**

Masseur-Kinésithérapeute ;

Conseiller Relation Client à Distance ;

Développeur Web et Web Mobile.

# **Présentation de l’établissement ‘’Bioinformatique’’ :**

L’établissement Bioinformatique est une école de Formation Professionnelle en Développement et en Informatique de Gestion, Software et Réseaux, référencée QUALIOPI.

Le centre propose de nombreuses formations numériques pour créer, valoriser ou développer des projets personnel ou professionnel. Avec des formateurs passionnés et expérimentés dans leurs domaines et attentifs à la satisfaction de chacun de leurs stagiaires, mettant tout en place, pour que la formation se déroule dans les meilleures conditions possibles.

L’attestation QUALIOPI indique une qualité de mise en œuvre des formations, pour le développement et l’acquisition des compétences.

Le centre dispose également d’un parc informatique à la pointe de la technologie pour les formations numériques.

Saidani Akli, Directeur des Etudes

Adresse : 70 Avenue du général De Gaulle, 94000 Créteil.

Tel : 06 13 54 10 16 Fax : +33 6 13 54 10 16 Email : akli.saidani2000@gmail.com

# **Introduction sur mon parcours :**

**A propos de NABI :**

J’ai exercé, pendant près de 20 ans, le métier d’enseignant en langue française.

Suite à l’évolution de mon état de santé, j’ai été dans l’obligation d’arrêter mon métier. En effet, après avoir perdu la vue, en 2012, suite à une tension oculaire (glaucome) et après une succession d’interventions chirurgicales, sans succès au Centre Hospitalier Universitaire (CHU) de Toulouse, ma cécité était devenue définitivement irréversible.

Devant ce fait accompli et après de longs moments de solitude, dont je vous épargne les détails, je devais, à tout prix, me relever et poursuivre mon chemin. Après tout, la vie ne s’arrête pas à la vue. Et quand on touche le fond, on ne peut que rebondir.

*‘’Assez de poésie !’’*

De 2014 à 2016, J’ai commencé par donner des cours d’alphabétisation au Secours Catholique, Site de Reynerie, Place André Abbal, 31100 Toulouse, et au centre d'hébergement social d'urgence où je résidais. C’est là, où j’ai découvert l’outil informatique.

En effet, il me permettait de préparer mes cours, de rédiger de petits textes et dialogues pour les jeux de rôle. J’en faisais ensuite des photos copies pour chaque élève et ne me restait plus qu’à expliquer la leçon et corriger la prononciation.

Afin d’apprivoiser cet outil, qui m’était devenu indispensable, je devais apprendre à utiliser les outils de compensation tels que le braille, les lecteur d’écran, les raccourcis-clavier, … Pour cela, j’ai intégré l’Association ‘’Valentin Haüy Toulouse’’.

Puis, de 2017 à 2019, j’ai intégré l’Institut des Jeunes Aveugles | Centre d'éducation spécialisé où j’ai séjourné plus de deux ans.

J’y ai appris la bureautique, l’utilisation de ‘’Word’’ et ‘’Excel’’, ainsi que l’utilisation de la plage braille et des lecteurs d’écrans. Et pour être plus autonome dans mes déplacements et mes tâches quotidiennes, j’ai intégré le centre de rééducation ‘’André Mathis’’, à Saint-Gaudens, Toulouse, où je suis resté quatre mois. J’en suis sortis plus confiant en moi et prêt à affronter le monde des ‘’voyants’’.

En 2019, la préfecture de Toulouse m’a accordé un titre de séjour d’un an (vie privée et familiale) et en ai fait bon usage.

En 2019/2020, j’ai intégré le centre CFRP AVH, où j’ai suivi des cours préparatoires en TAI (Technicien Assistant en Informatique) durant lesquels, j’ai appris les bases du langage python, le système linux, la virtualisation, l’algorithme, …

*‘’Il faut dire que j’avais été mordu par le python !’’.*

En 2020/2021, ma passion pour la réalisation et le développement de projets web, m’a tourné, tout naturellement, vers une carrière de développeur web. C’est alors que j’ai intégré le Centre de Formation Paul et Liliane Guinot, pour une formation de deux ans de développeur web et web mobile, en PHP, Symfony, HTML, JavaScript et du SQL.

Fort de ces connaissances, mon plus grand souhait était de donner une nouvelle dynamique à mon parcours professionnel. La formation, Développeur Web et Web Mobile a enrichi mes connaissances.

Ainsi, en première année on m’a initié aux fondamentaux de la programmation (PHP Objet, HTML, CSS, Javascript, Jquery, SQL/conception d'une base de données avec la méthode Merise...).

En seconde année, les objectifs pédagogiques se sont inscrits dans une logique professionnelle :

Apprentissage des Frameworks élémentaires PHP (Symfony 4 et Symfony 5) ;

Développement d'un thème, modification d'un thème avec la méthode de l'héritage, développement de plugins, etc.…)

De ce fait, mon désire était de mettre en œuvre, sur le terrain, tout ce que j’ai appris, afin de contribuer à l’amélioration de l’accessibilité numérique pour les non-voyants, qui sont dans mon cas et intégrer un jour une entreprise dynamique et jeune et faire partie d’une équipe de développeurs web.

Je sais désormais faire preuve d'esprit d'initiative et de rigueur. Fort de ma passion.

# **Liste des compétences du référentiel :**

**Activité type 1 :**

Développer la partie front-end d’une application web ou web mobile en intégrant les recommandations de sécurité.

Dans cette catégorie, il nous est demandé de :

 Maquetter une application ;

 Réaliser une interface utilisateur web statique et adaptable ;

 Développer une interface utilisateur web dynamique ;

Ou de :

 Maquetter une application ;

 Réaliser une interface utilisateur avec une solution de gestion de contenu ou de e-commerce.

**Activité type 2 :**

Développer la partie back-end d’une application web ou web mobile en intégrant les recommandations de sécurité.

Dans cette catégorie, il nous est demandé de :

 Développer les composants d’accès aux données ;

 Développer la partie back-end d’une application web ou web mobile ;

Ou de :

 Développer les composants d’accès aux données ;

 Elaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce.

# **Guide d’emploi de l’application :**

Pour un visiteur lambda, une fois sur le site, Il peut accéder à la page d’accueil de l’application. Il peut accéder au lien ‘’Introduction sur Guinot’’ ou il peut consulter l’historique de Guinot, le mot du président, les différentes formations proposées et les conditions pour intégrer le centre de formation.

Il y aura également un bouton pour s’inscrire, à travers lequel, le visiteur devra saisir son login et son mot de passe. Une fois le formulaire d’inscription validé, l’utilisateur a la possibilité :

* D’accéder à la messagerie publique de Guinot, où il pourra envoyer et recevoir des messages, des documents, des fichiers MP4, …
* D’accéder à son espace personnel, où il pourra consulter ses messages, son agenda, ses activités, …
* De s’inscrire à l’une des formations proposées par le centre.

Dans ce dernier cas, un formulaire d’inscription à la formation s’affiche et le stagiaire devra saisir :

* Son nom ;
* Son e-mail ;
* Le nom de la formation qu’il a choisi ;
* Le nom du langage qu’il souhaite apprendre ;
* Les prérequis pour la formation ;
* La période de formation.

Remarque : Une fois le stagiaire inscrit à la formation, il pourra accéder à la liste des cours du langage qu’il a choisi.

Il y aura également : - Un bouton ‘’Se déconnecter’’ ;

- Un bouton ‘’Revenir à la page d’accueil’’.

# **Projet de l’organisme de formation BIOGESOFT :**

Afin de compléter le projet de fin de stage de Mohammed NABI, le module ‘Formations’ a été ajouté. Cette extension du projet consiste à mettre en place un ‘’Système d’Inscription des Candidats dans différentes formations, proposées par BIOGESOFT, en fonction du processus suivant :

A partir de la page d’accueil de l’école, on a un catalogue, où on peut consulter toutes les formations existantes.

Pour chacune des formations, on retrouve :

Le nom de la formation ;

Sa durée ;

Le coût de la formation ;

Le titre de la formation ;

Son objectif ;

Ses prérequis.

Chacune des formations est organisée en sessions, avec deux sessions par année ;

Chaque formation est composée de plusieurs modules ;

Chacun des modules est composé de plusieurs cours, avec :

Des cours théoriques ;

Des travaux pratiques ;

Des exercices.

NB : Prévoir aussi des cours sous forme de vidéo, que le candidat peut visualiser sans interagir en direct avec son formateur (juste visualiser la vidéo).

Voici quelques contraintes :

Un cours quelconque (théorique, pratique, exercice ou vidéo) peut figurer dans un ou plusieurs modules de formation ;

Chaque module a un nom, une durée, un coefficient, un objectif ;

Chaque cours a un nom, une durée, un contenu de plusieurs pages, en PDF ou une Vidéo ;

Un candidat peut être dispensé de certains modules, compte tenu de son niveau de prérequis et cette dispense est déterminée par l’organisme de formation.

Le candidat peut seulement choisir une formation et la session.

La mise à jour du catalogue des formations, des modules et de leurs contenus est confiée à l’administration.

Cette extension va servir surtout aux élèves de BIOGESOFT, pour leur permettre de suivre des cours à distance, en éditant chez eux les cours ou les vidéos mis en ligne.

*Remarque : On peut toujours ajouter des attributs dans chaque classe, objets ou table relationnelle de la base de données selon les exigences et les besoins du client.*

Compte tenu du temps restant, nous ne nous intéresserons pas à la formulation des demandes d’inscription, avec l’automatisation des réponses positives ou refus. On suppose que le candidat peut s’inscrire directement en ligne, pour une seule formation de son choix et l’inscription est validée lorsque le candidat joint à son inscription les documents demandés (pièce d’identité, justificatif de son niveau de formation (si demandé), un RIB).

# **Présentation du projet :**

Le projet de messagerie «STAGIAIRES-GUINOT», se compose de deux parties distinctes :

- La partie backend : la gestion entière des différents modules mis en place ainsi que leurs accès par l’administrateur ou par un autre rôle, si celui-ci dispose des droits nécessaires.

- la partie frontend : elle correspond à la mise en place d’un design et plus globalement à l’interface de navigation disponible sur l’ensemble du site.

Cette interface, accessible à un public varié, devra prendre en compte certains points d’accessibilité, pour que son accès soit le plus aisé possible pour l’ensemble des utilisateurs. Il est important de scinder ces deux parties car elles requièrent un travail différent mais elle demande, également, la mise en place pour chacune d’entre-elles des points techniques différents.

L’application dispose de plusieurs modules que nous évoquerons brièvement :

- Un module ‘’Utilisateurs’’ qui permet la gestion complète d’un profil dit utilisateur.

Lorsque nous aborderons le terme ‘’utilisateur’’, précisons que celui-ci désigne un enregistrement quelconque effectué via le module ‘’Utilisateur’’ sur la table ‘’User’’. Il peut s’agir de tout type de rôle avec tout type d’utilisateur.

- Un module ‘’Messages’’ qui permet de gérer les messages existants et de modifier, supprimer ou en ajouter de nouveaux. Ce module nécessite un accès administrateur. En effet, seuls les membres ayant le statut ‘’administrateur’’ peuvent accéder à ce module.

*Remarque : Chaque stagiaire identifié peut également gérer les messages existants dans son espace personnel, modifier, supprimer ou en ajouter de nouveaux.*

- Un module ‘’Formations’’ qui permet de gérer les formations existantes et permet de modifier, supprimer ou en ajouter de nouvelles. Ce module nécessite un accès administrateur. En effet, seuls les membres ayant le statut ‘’administrateur’’ peuvent accéder à ce module.

- Un module ‘’Langages’’ permet de gérer les langages existants et permet de modifier, supprimer ou en ajouter de nouveaux. Ce module nécessite un accès administrateur. En effet, seuls les membres ayant le statut ‘’administrateur’’ peuvent accéder à ce module.

- Un module ‘’Cours’’ qui permet de gérer les cours existants et permet de modifier, supprimer ou en ajouter de nouveaux. Ce module nécessite un accès administrateur. En effet, seuls les membres ayant le statut ‘’administrateur’’ peuvent accéder à ce module.

- Un module ‘’Stagiaires’’ qui permet de gérer les stagiaires existants et permet de modifier, supprimer ou en ajouter des nouveaux. Ce module nécessite un accès administrateur. En effet, seuls les membres ayant le statut ‘’administrateur’’ peuvent accéder à ce module.

- Un module ‘’Activités’’ qui permet de gérer les activités existantes et permet de modifier, supprimer ou en ajouter de nouvelles. Ce module nécessite un accès administrateur. En effet, seuls les membres ayant le statut ‘’administrateur’’ peuvent accéder à ce module.

*Remarque : Chaque Stagiaire identifié peut également gérer les activités existantes dans son espace personnel, modifier, supprimer ou en ajouter de nouvelles.*

- Un module ‘’Médias’’ qui permet de gérer des médias. Comme pour le module ‘’Messages’’, il nécessite quelques droits d’accès. Pour y accéder, il faut être connecté donc être enregistré et disposer d’un rôle ‘’administrateur’’. En effet, seuls les administrateurs peuvent créer, éditer et plus globalement géré un média.

*Remarque : Chaque stagiaire identifié peut également gérer les médias existants dans son espace personnel, modifier, supprimer ou en ajouter de nouveaux.*

- Un module ‘’Contacts’’ qui permet de gérer des contacts. Comme pour le module ‘’Médias’’, il nécessite quelques droits d’accès. Pour y accéder il faut être connecté donc être enregistré et disposer d’un rôle ‘’administrateur’’. En effet, seuls les administrateurs peuvent créer, éditer et plus globalement géré un média.

*Remarque : Chaque stagiaire identifié peut également gérer les contacts existants dans son espace personnel, modifier, supprimer ou en ajouter de nouveaux.*

C’est donc l’ensemble des différents modules présentés synthétiquement ci-dessus qui forment l’application. « STAGIAIRES-GUINOT ».

L’animation et le design ont été réalisés grâce au Framework bootstrap ainsi qu’avec l’utilisation de JavaScript.

Il est important de préciser que l’application ‘‘STAGIAIRES-GUINOT’’ a été réalisée grâce au Framework Symfony, qui est un cadre de travail conçu en PHP. Ce Framework permet le développement de son application de manière très sécurisée avec un ensemble de bonnes manières, qui sont communes à l’ensemble des développeurs, on peut citer les PSR (PHP Standard Recommandation).

C’est un ensemble de recommandations pour la programmation PHP publiée par le FIG (Framework Interoperability Group). Il permet d'améliorer l'interopérabilité entre les Frameworks mais aussi d'uniformiser les syntaxes et le formatage du code.

Cela le rend certes plus compliqué à prendre en main mais c’est l’un des outils le plus utilisé dans le développement web.

Il m’a été demandé de ne pas utiliser des Frameworks lourds et de privilégier le CSS simple, couplé au JavaScript.

# **Cahier des charges :**

**Nom du stagiaire :** NABI Mohammed.

**Nom du projet :** Retrouvailles des Anciens Stagiaires de GUINOT.

Le site "ANCIENS RETROUVAILLES GUINOT’’ est une application en ligne, spécialisée dans la recherche de personnes qui ont fréquenté le même établissement : Centre de Formation et de rééducation Professionnelle pour Malvoyants Paul et Liliane Guinot, Villejuif (94814).

Cette plateforme, en ligne, offre un espace de rencontre, de souvenirs, d’apprentissage, d’échanges et de partage d’expériences professionnelles entre les anciens stagiaires qui ont fréquenté l’établissement et les nouveaux stagiaires qui le découvrent.

Cette application établit, entre-autres, un vrai lien générationnel, une continuité culturelle et éducative riche en conseils, en expériences, principalement dans le domaine professionnel et celui lié à l’accessibilité.

Page d’accueil de l’application :

L’application "ANCIENS RETROUVAILLES GUINOT » dispose d’un service de tchat public, accessible aux utilisateurs.

Au lancement de l’application, il y aura :

* Une présentation sur le Centre Guinot, le fondateur de l’établissement, avec un lien (En savoir plus …) ;
* Un bouton "s’inscrire’’ ;
* Un bouton ‘’se connecter" ;
* Un bouton ‘’rechercher’’ ;
* Un bouton ‘’commentaires sur l’application’’.
* Un formulaire d’inscription. La personne qui souhaite s’inscrire sur le site

"ANCIENS RETROUVAILLES GUINOT » devra saisir les informations suivantes :

• sa civilité ;

• son nom ;

• son prénom ;

• le pays dans lequel il ou elle vit ;

• son adresse ;

• son Code postal ;

• sa ville ;

• son numéro de téléphone ;

• son mail ;

• un mot de passe supérieur ou égal à 6 caractères ;

• la formation suivie ;

• sa profession s'il en a ;

• Il faudra aussi préciser l’année de la formation.

Pour effectuer une demande de recherche, il faut se connecter sur le site "ANCIENS RETROUVAILLES GUINOT ».

Pour ce qui est du login, si la personne a saisi un numéro de téléphone et un mail, ce dernier constitue son login, sinon ce sera son numéro de téléphone.

Une fois inscrit, on reçoit un mail ou un texto de confirmation.

Se connecter : Deux zones de saisie seront présentées à savoir le login et le mot de passe, un bouton de validation et une option "mot de passe oublié".

L'option ‘’mot de passe oublié’’ permet d’envoyer un mail ou un texto contenant un lien conduisant à la page où l’on crée un nouveau mot de passe.

Choix d'un nouveau mot de passe : L'abonné, une fois avoir cliqué sur le lien envoyé, arrive à une page où on lui demande de choisir un nouveau mot de passe et doit le confirmer avec le bouton de validation.

Connexion établie : Une fois connecté à son espace personnel, il retrouve la liste de ses contacts approuvés et un historique de ses discussions (les 5 derniers messages textes / album photos / vidéo) et agenda. Il peut créer ou supprimer ses messages textes, de même, la personne connectée, pourra envoyer des photos ou des vidéos.

Un bouton ‘’rechercher un ancien stagiaire’’, « modifier / supprimer mes données personnelles », un bouton « supprimer mon compte » et un bouton de "déconnexion" sont aussi présents.

Si la personne a des demandes venant d'autres personnes, déjà membres ou non, elle les retrouvera dans cet espace.

Si la personne qui a fait la demande n’a pas de compte, elle recevra la réponse dans sa boîte mail, avec un message l’invitant à s’inscrire dans l’application, sinon elle ne pourra pas contacter cette personne.

Le bouton de "Déconnexion" : Celui-ci renvoie l’abonné à l’écran d’accueil, tout en lui demandant d’abord, avec un message de confirmation, s'il veut réellement se déconnecter.

Modification des données personnelles

C'est une fenêtre qui affiche toutes les données personnelles de l'utilisateur, saisies lors de l'inscription. Chaque information personnelle est modifiable s'il le souhaite.

La recherche peut s'effectuer de plusieurs façons :

• Recherche par le nom de la personne ;

• Recherche par la période d'étude ;

• Recherche par Email, Fax, Téléphone ou compte Facebook ;

• Recherche par département ( ville, régions ou code-postale)...

Si une demande de recherche est entamée, alors un e-mail direct est initie.

Exemple de recherche : « Rechercher un stagiaire »

La recherche se fera par nom. Le résultat affichera tous les noms et prénoms. L'abonné sélectionnera la personne qu'il recherche.

Il y aura donc une zone d'édition nom plus un bouton "valider".

Une fois la recherche effectuée, la personne concernée est automatiquement avertie par courriel avec un lien pour se connecter, en plus des options suivantes :

• Cliquez sur ce lien pour vous connecter ;

• Accepter ;

• Refuser ;

• Bloquer ;

• Signaler.

Une fois le contact établi, la personne invitée, peut accéder au service de tchat public, accessible aux utilisateurs ou elle peut, si elle le désir, discuter, envoyer des documents, des photos, des vidéos…

La personne invitée a également la possibilité de créer son espace personnel en s’inscrivant sur le site "ANCIENS RETROUVAILLES GUINOT », dans le cas contraire, on peut uniquement accéder au service de tchat public en ligne, pour discuter.

En cliquant sur le bouton « S’inscrire » sur la page d’accueil, un formulaire d’inscription s’affiche avec les champs à remplir suivants :

Veuillez renseigner les champs suivants :

• Votre Nom ;

• Votre Prénom ;

• Votre mot de passe ;

• Votre Civilité ;

• L’année de votre formation ;

• La filière de votre formation ;

• Votre statut actuel ;

• Nom de votre entreprise actuelle ;

• Motif de votre visite (Recherche d’une personne, Offrir de l’aide, demander de l’aide (Document administratifs, Faire un don, Offre de stage, Offre d’emploi, Autres…),

• Description ( Veuillez préciser en quelques lignes l’objet de votre demande.).

Une fois inscrit sur le site "ANCIENS RETROUVAILLES GUINOT », la personne peut accéder à son espace personnel en entrant :

• Son login ( e-mail ),

• Son mot de passe.

Une fois sur son espace personnel, la personne connectée peut effectuer les tâches suivantes :

• Faire une recherche ;

• Accéder au service de son espace de tchat privé ;

• Accéder à son agenda ;

• Accéder à son album photos ;

• Accéder à ses contacts ;

• Consulter la liste de ses messages ;

• Consulter ses dix (10) derniers messages ;

• Ajouter un nouveau contacte ;

• Supprimer un contacte ;

• Modifier un message ;

• Envoyer un message ;

• Supprimer un message ;

• Envoyer (des documents, des fichiers MP3/MP4, des photos/images, des vidéos…) ;

• Recevoir (des documents, des fichiers MP3/MP4, des photos/images, des vidéos…) ;

• Supprimer (des documents, des fichiers MP3/MP4, des photos/Images, des vidéos…) ;

. Gérer ses activités ;

• Se déconnecter.

*Remarque : une fois déconnecté, toute modification est automatiquement mise à jour et enregistrée dans son historique.*

L’application fournie les services de gestion suivants (ils comprennent la création, la modification et la suppression) :

• Gestion des évènements ;

• Notation des activités ;

• Commentaires sur les activités ;

• Présentation interactive de l’utilisation du site aux utilisateurs.

*Remarque : un abonné peut ajouter ou supprimer un ou plusieurs contacts à son espace personnel.*

Pour améliorer la qualité des activités, les activités bien notées sont mises en avant sur le site.

Ils peuvent régler par carte ou PayPal.

SPECIFICATIONS POUR LE SITE INTERNET

Le site internet ‘’STAGIAIRES-GUINOT’’ constitue une application accessible par différents types d’utilisateurs, à savoir :

L’application permet de s’inscrire, en tant que :

•  Administrateur ;

•  Utilisateur ;

•  Anonyme.

Profil ANONYME : Il s’agit des utilisateurs non authentifiés qui peuvent accéder au site afin de :

• S’inscrire en tant que stagiaire ;

• Consulter les activités proposées sur le site ;

• Consulter ses messages personnels ;

• Consulter la présentation de l’utilisation de l’application ;

• Utiliser le tchat ;

• Faire un commentaire en ligne.

Profil UTILISATEUR : Il s’agit des stagiaires qui possèdent un compte personnel et qui se sont authentifiés. Ils peuvent donc :

• Se connecter ;

• Accéder à leur espace personnel ;

• Consulter la liste de leurs messages ;

• Entamer une discussion ;

• Faire un don par CB ou PayPal ;

• Télécharger des fichiers ;

• Modifier ses informations ;

• Consulter l’historique de leurs messages ;

• Communiquer par tchat ou par mails ;

• Accéder à la fiche de ses contacts ;

• Faire une consultation en ligne de ses dix derniers messages ;

Profil ADMIN : il s’agit de l’administrateur de l’application "ANCIENS RETROUVAILLES GUINOT ». Ce profil, en plus de tous ceux que font les autres profils, peut également effectuer les tâches suivantes :

• Gestion de l’ensemble des tables de la base de données ;

• Se connecter ;

•  Gestion des informations ;

• Gestion des comptes utilisateurs ;

• Consulter les revenues et les dépenses ;

• Gestion des finances ;

• Gestion des dons ;

• Organisation de conférences ;

• Proposition d’activités culturelles et sociales ;

• Suppression de contenus ou d’images compromettantes :

• Etablissement de statistiques.

# **Techniques à adopter :**

La technique à adopter est la suivante :

On aura deux entités principales.

• L’entité Utilisateurs, qui gère les utilisateurs connectés ;

• L’entité Messages, qui gère les messages des utilisateurs.

Dans l’entité Utilisateurs, on a deux profils actifs.

• Le profil Expéditeur, celui qui envoie le message ;

• Le profil Destinataire, celui qui reçoit le message ou la réponse.

Chaque stagiaire possède une liste d'abonnement d’utilisateurs.

Chaque abonnement est lié à une file de messages émis par un autre utilisateur.

Côté serveur : La gestion des listes d'abonnements s’effectue du côté du serveur.

Côté client : Les messages sont visibles par tous les stagiaires.

Une fois l’utilisateur abonné, le filtrage s’effectue par rapport au choix de l'utilisateur qui reçoit et donc seuls les messages choisis, lui seront transmis.

En résumé, l’application "STAGIAIRES-GUINOT.COM’’ dispose d’un service de tchat public, accessible aux utilisateurs de Guinot. La personne connectée, peut, si elle le désire, discuter, envoyer des documents, des photos, des vidéos… Elle dispose également d’un service de tchat privé entre deux personnes maximum, seuls les deux utilisateurs concernés peuvent voir la discussion ;

*Remarque : Un abonné peut ajouter ou supprimer un ou plusieurs contacts dans son espace personnel.*

• Les activités bien notées sont mis en avant sur le site.

• Ils peuvent régler les donations, par carte ou PayPal.

Projet de fin de formation : Développeur web et web mobile.

NABI M.

Stagiaire développeur web et web mobile.

Tel : 06 38 48 27 32.

Email : [nabiabib31@gmail.com](mailto:nabiabib31@gmail.com)

# **Analyse et conception :**

1) Conception du projet.

Pour notre projet ? nous utiliserons la méthode RUP (Rational Unified Process).

Elle est très efficace car elle répond plus précisément aux besoins du client.

La méthode RUP, de quoi s’agit-il ?

La méthode RUP (Rational Unified Process) est une des émanations de la méthode PU, qui s’attache à donner un cadre précis au développement du logiciel. C’est une méthode générique, itérative et incrémentale assez lourde mais qui s’adapte très facilement aux processus et aux besoins du développement.

Tout au long du développement, on est en contact avec le client.

Avec la méthode RUP, on n’est pas figé sur une analyse. Elle est très adaptable et flexible aux besoins du client et ce, à la différence de la méthode Merise.

Auparavant, lorsque l’on utilisait la méthode Merise, une fois l’analyse effectuée, on ne peut plus retourner en arrière.

Aujourd’hui, cette méthode est obsolète car les besoins du client sont très changeants et exigeants. Les coûts du projet sont très onéreux et une entreprise ne peut prendre ce genre de risques.

L’UML, c’est quoi ?

UML est l’acronyme anglais de « Unified Modeling Language ». On le traduit en français par « Langage de Modélisation Unifié ».

L’UML est donc un langage de modélisation graphique, permettant d’expliquer les modèles objets, à travers un ensemble de diagrammes. En effet, un diagramme d’UML est une représentation graphique, qui offre treize (13) types possibles de diagrammes et chaque type de diagramme offre une vue différente d’un système donné.

a) Diagramme.

Tableau des opérations d’un cas d’utilisation.

Ce tableau détaille les cas Graphiques sans nom d’opération. Il sert de base à la construction de support d’interface graphique.

Acteur : Visiteur ;

Evènement Déclencheur : Néant ;

Intérêt : Accéder au Serveur pour s’authentifier et accéder à l’interface réservée au compte en question ;

Précondition : Serveur Actif ;

Post-Condition : Connexion Etablie ;

Scenario Nominal :

1. L’utilisateur saisit l’adresse IP du Serveur ;

2. L’utilisateur demande une connexion au Serveur ;

3. Le Serveur répond ;

Scenario Alternatif :

o L’adresse IP n’est pas renseignée, retour au cas 1 ;

o Le Serveur retourne un cas d’erreur, retour au cas 1 ;

o Le Serveur ne répond pas, retour au cas 1 ;

Cas d’utilisation : Créer un compte utilisateur :

Acteur : Administrateur ;

Evènement Déclencheur : Néant ;

Intérêt : crée un conte utilisateur permettant à une personne choisis l’accès à certaines parties du site ;

Précondition : Serveur Actif ;

o authentification réussie

Post-Condition : Néant ;

Scenario Nominal :

4. L’utilisateur saisit l’adresse IP du Serveur ;

5. L’utilisateur demande une connexion au Serveur ;

6. Le Serveur répond ;

Scenario Alternatif :

o L’adresse IP n’est pas renseignée, retour au cas 1 ;

o Le Serveur retourne un cas d’erreur, retour au cas 1 ;

o Le Serveur ne répond pas, retour au cas 1.

Il existe 3 types de rôles utilisateurs différents :

L’administrateur : Il a tous les droits sur l’application ;

Le visiteur : Il peut consulter uniquement le site « STAGIAIRES-GUINOT.COM» ;

L’expéditeur : il peut accéder à un espace, qui lui est réservé, il peut envoyer un message et avoir un aperçu sur la liste des messages reçus, il peut également gérer ses messages, ses médias, ses contacts ainsi que ses activités (Ajouter, supprimer ou modifier).

Le destinataire : il peut, lui aussi, accéder à un espace, qui lui est réservé, il peut répondre à un message reçu et avoir un aperçu sur la liste de ses messages envoyés, il peut également gérer ses messages, ses médias, ses contacts ainsi que ses activités (Ajouter, supprimer ou modifier).

B. Module « authentification »

a. Authentification de l’administrateur :

- Pour accéder à « Espace administrateur » de la « Messagerie STAGIAIRES-GUINOT», l'administrateur doit s'authentifier.

- Le formulaire d'authentification est disponible via l'adresse URL : «-STAGIAIRES-GUINOT.COM/admin ». Pour information, tous les utilisateurs ne possédant pas le statut administrateur seront automatiquement redirigés vers la page d’accueil.

Éléments composant le formulaire :

- Champ « Adresse email » (type="email" / champ obligatoire) ;

- Champ « Mot de passe » (type="password" / obligatoire) ;

- Bouton « Se connecter » (pour valider le formulaire).

Cas d'utilisation :

* Nom : S'authentifier (module « Authentification Administrateur ») ;
* Acteur(s) : Administrateur ;
* Description : L'utilisateur doit pouvoir s'authentifier et accéder à la messagerie «STAGIAIRES-GUINOT.COM» ;
* Préconditions : Avoir un compte utilisateur créé et posséder le rôle Admin ;
* Démarrage : L'utilisateur a demandé la page de connexion.

Scénario nominal :

1. Le système affiche le formulaire de connexion ;

2. L'utilisateur saisit son adresse e-mail et son mot de passe ;

3. Le système vérifie les données saisies ;

4. Le système affiche le tableau de bord du back-office Admin.

Scénarios alternatifs :

3a Le système indique que le compte n'existe pas ;

3b Le système redirige vers la page d’accueil car l’utilisateur ne possède pas un compte ayant le statut Admin.

Post conditions :

Scénario nominal : reconnaissance de l'utilisateur / accès à la plateforme = affichage des menus de gestion et du back-office Admin25.

b. Authentification de l’utilisateur :

Pour accéder à l’espace utilisateur, ce dernier doit s’authentifier.

- Le formulaire d'authentification est disponible via l'adresse URL : «STAGIAIRES-GUINOT.COM». Pour information, par mesures de sécurité, tout visiteur qui tenterait d'accéder à la plateforme par l'URL «STAGIAIRES-GUINOT.COM /[chemin sur le serveur web vers la ressource] » sera redirigé sur le formulaire d'authentification.

- - Bouton « Se connecter » (pour valider le formulaire).

Cas d'utilisation :

Nom : S'authentifier (module « Authentification utilisateur ») ;

Acteur(s) : utilisateur ;

Description : L'utilisateur doit pouvoir s'authentifier et accéder à la messagerie «STAGIAIRES-GUINOT.COM» ;

Préconditions : Avoir un compte utilisateur créé ;

Démarrage : L'utilisateur a demandé la page de connexion ;

Scénario nominal :

1. Le système affiche le formulaire de connexion ;

2. L'utilisateur saisit son adresse e-mail et son mot de passe ;

3. Le système vérifie les données saisies 26 ;

4. Le système affiche le tableau de bord du back-office utilisateur ;

Scénarios alternatifs :

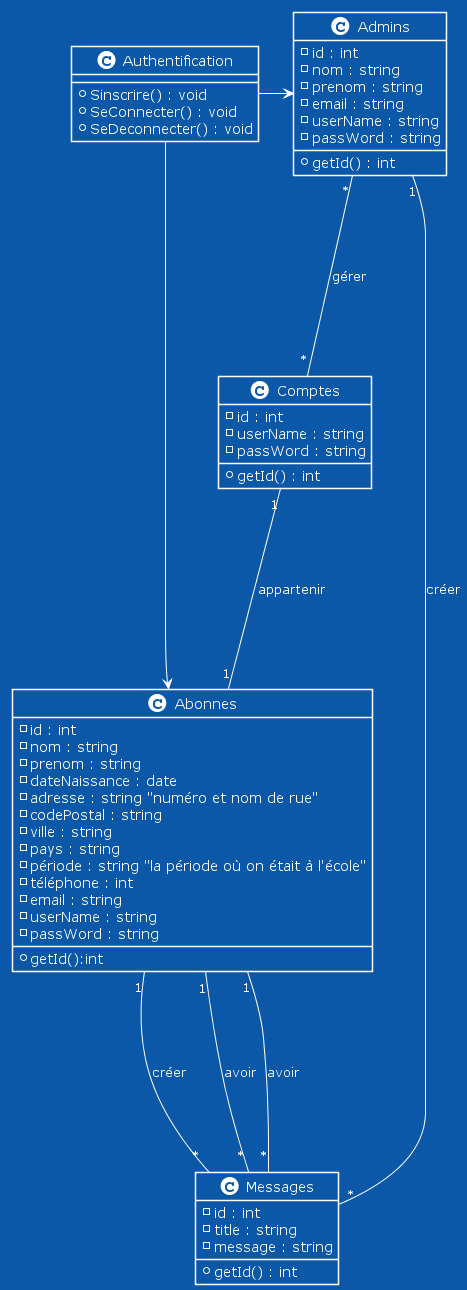
3a Le système indique que le compte n'existe pas 27 ;

Post conditions :

Scénario nominal : reconnaissance de l'utilisateur / accès à la plateforme = affichage des menus de gestion et du back-office utilisateur ;

Une fois l’utilisateur connecté, il a la possibilité d’accédé à son espace personnel 28.

## **Le diagramme de classes :**



Classe : Messages

--id

--nom

--Créer date

--expéditeur : association : (un utilisateur envoie plusieurs messages et chaque message ne peut avoir qu’un seul utilisateur (ManyToOne))

--destinataire : association : (un utilisateur reçoit plusieurs messages et chaque message ne peut avoir qu’un seul utilisateur (ManyToOne))

Classe : Médias

--id

-- titre

-- description

--utilisateurs : association : (un média appartient à un utilisateur et chaque utilisateur possède plusieurs médias (ManyToOne))

Classe : Activités

--id

-- titre

-- description

--utilisateurs : association : (une activité appartient à un utilisateur et chaque utilisateur possède plusieurs activités (ManyToOne))

Classe : Contacts

--id

--nom

--prénom

--téléphone

--adresse

--code\_postal

--ville

--département

--pays

--e-mail

--créer\_date

--utilisateurs : association : (un contact appartient à un utilisateur et chaque utilisateur possède plusieurs contacts (ManyToOne))

Classe : ResetPasswordRequest

--id

--user

--association : une demande de réinitialisation concerne un utilisateur et plusieurs demandes d'utilisateur peuvent être effectuées (ManyToOne)

Classe : Utilisateurs

--id

--email

--roles

--password

--isVerified

--nom

-- prenom

-- civilite

-- date

-- adresse

-- code\_postal

-- villes

-- departement

-- pays

--message\_envoyer: association: (un message ne peut être envoyé que par un seul utilisateur, et chaque utilisateur peut envoyer plusieurs messages (OneToMany))

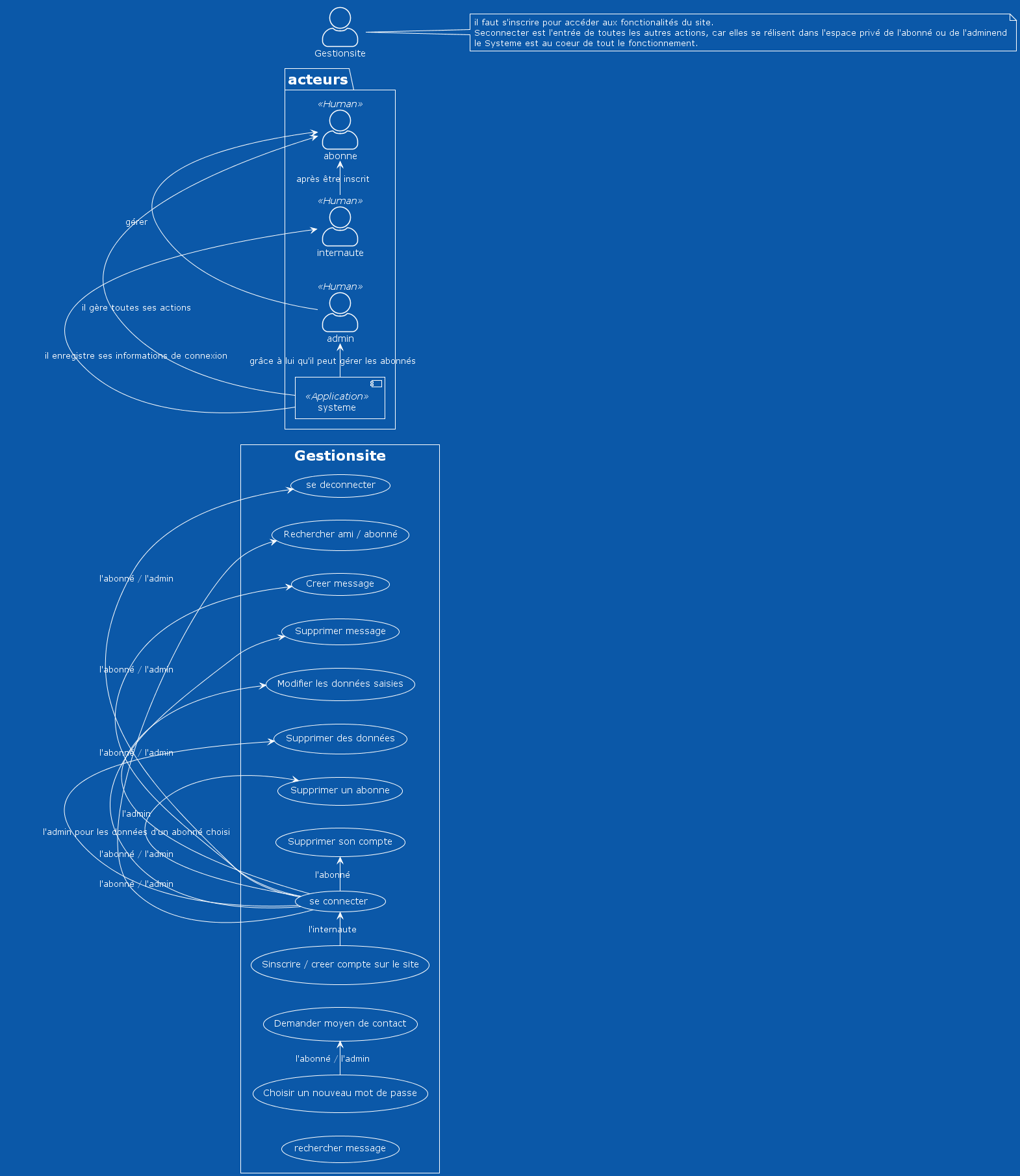
--message\_reçu: association: (un message ne peut être reçu que par un seul utilisateur, et chaque utilisateur peut recevoir plusieurs messages (OneToMany))

--Contacts : association : (un utilisateur peut avoir un ou plusieurs contacts et un contact concerne un seul utilisateur (OneToMany))

--médias : association : (un utilisateur peut avoir un ou plusieurs médias et un média concerne un seul utilisateur (OneToMany))

--activité : association : (un utilisateur peut avoir une ou plusieurs activités et une activité concerne un seul utilisateur (OneToMany))

## **Le diagramme de cas d’utilisation :**



## **Description des cas d’utilisations :**

**Description du cas Se déconnecter**

Cas d’utilisation : Se déconnecter de la plate-forme

Description du processus :

Au niveau du menu de navigation principal, tout utilisateur authentifié dispose d’un lien lui permettant de se déconnecter et donc de fermer sa session en toute sécurité. En validant ce lien, l’utilisateur est redirigé vers la page d’accueil et donc sur le formulaire d’authentification.

|  |  |
| --- | --- |
| Cas d’utilisation | Se déconnecter |
| Acteurs principal : | L’abonné et l’admin |
| Acteur secondaire | Système |
| Objectif | Quitter son espace personnel |
| Pré-condition | Etre déjà dans son espace personnel |
| Poste-condition | Session personnel fermée |
| Scénaris | 1. L’utilisateur clique sur le lien de déconnexion. 2. Le system demande à l’utilisateur s’il veut bien fermer sa session. 2. L’utilisateur clique sur « ok » pour confirmer. 4. Le système ferme son espace personnel et le redirige à la page d’accueil. |
| Scénaris alternatifs | * 1. L’utilisateur clique sur le lien « déconnexion » par erreur. 3.1 Il clique sur le bouton « annulé ». 4.1. Le système annule l’opération et le ramène à sa page personnel. |

**Cas d’utilisation : modification des données.**

|  |  |
| --- | --- |
| Cas d’utilisation | Modifier des données |
| Acteur | User ( abonné et l’admin ) |
| Objectif | Apporter des modifications sur ses propres données ou sur les données d’un abonné donné |
| Précondition | Etre déjà dans son propre espace personnel ou dans l’espace personnel d’un abonné en question |
| Postcondition | Mise à jour des données de l’user effectuée dans la base de données |
| Scénaris | 1. Etre déjà connecté dans l’espace personnel. 2. User clique sur « modifier mes données ». 3. Le système affiche la page des informations saisies au moment de l’inscription avec des zones modifiables sur les propriétés « email », « téléphone » et adresse. 4. L’user effectue ses modifications et clique sur « modifier mes données ». 5. Le système vérifie et enregistre les modifications, puis affiche un message de confirmation avec un bouton « ok ». 6. L’user clique sur « ok ». 7. Le système le renvoie à son espace personnel. |
| Scénario alternatif | 5.1. le système constate que certaines des données sont mal saisies, il renvoie l’user à l’étape précédent. |

Remarque :

Seul l’admin a le droit d’effectuer des modifications sur les comptes des utilisateurs.

**Cas d’utilisation : Choisir un nouveau mot de passe**

|  |  |
| --- | --- |
| Cas d’utilisation | Choisir un nouveau mot de passe |
| Acteur | User (l’abonné et l’admin) |
| Objectif | Changer son mot de passe suite à un oubli ou vouloir plus sécurisé |
| Précondition | Avoir cliqué sur le lien que l’on a reçu pour changer son mot de passe |
| Postcondition | Disposer d’un nouveau mot de passe |
| Scénaris | L’utilisateur reçoit un mail ou texto contenant un lien sur lequel il doit cliquer.  2. il clique sur ce lien.  3. Le système lui affiche la page de modification de mot de passe. 4. L’utilisateur remplit son nouveau mot de passe et le confirme, puis il valide. 5. Le système vérifie la conformité des informations et enregistre les données tout en lui affichant un message de confirmation avec un bouton « ok ». 6. L’utilisateur clique sur « ok ». 7. Le système le redirige à la page de connexion. |
| Scénario alternatif | 3.1. le système ne peut pas trouver la page demandée, il informe l’utilisateur avec un bouton « ok » «3.2. l’utilisateur clique sur ce « ok ». 3.3 le système le renvoit à la page « mot de passe oublié » où il lui demande ses moyens de contact. 5.1 Le système constate que le mot de passe tapé n’est pas conforme aux critères de mot de passe défini, il informe l’utilisateur et l’invite à cliquer sur « ok ». 5.2. l’utilisateur clique sur ce bouton. 5.3. le système le ramène à l’étap précédent. |

**Description du cas : créer un message**

|  |  |
| --- | --- |
| Cas d’utilisation | Créer un message |
| Acteur | User (l’admin et l’abonné) |
| Acteur secondaire | Système |
| Objectif | Ecrire un message |
| Préconditions | Etre inscrit dans l’application |
| Poste-conditions | Ecrire son message et pouvoir l’envoyer à son destinataire |
| Scenarios | 1. Se connecter à son espace. 2. Cliquer sur message. 3. Le système ouvre la page de message avec les options. 4. User clique sur « nouveau message ». 5. Le système lui affiche la page avec les zones d’éditions (à : CC : CCI : objet : corps  avec les boutons « envoyer » et « annuler ». 6. User compose son message et clique « envoyer ». 7. Le système envoie le message et lui donne l’information. 8. User clique sur « ok ». 9. Le système le redirige à l’accueil de son espace privé |

**Cas d’utilisation : faire une recherche**

Description du processus :

Une fois sur le site «Retrouvaille des Anciens Stagiaires De Guinot»,

Et sur son espace utilisateur dédié

L’abonné peut effectuer une recherche sur le site

|  |  |
| --- | --- |
| Cas d’utilisation | Faire une recherche |
| Acteur | User (l’internaute, l’abonné et l’admin) |
| Objectif | Retrouver un ancien stagiaire ou un abonné (pour l’admin) |
| Précondition | Etre dans l’application, mais, pour l’admin il faut qu’il se connecte dans son espace. |
| Poste-condition | Trouver un ancien stagiaire qui a fréquenté le centre de formation. |
| Scénario | 1. L’utilisateur clique sur « rechercher ». 2. Le système ouvre la zone de recherche. 3. L’utilisateur tape le nom de la personne qu’il recherche puis il clique sur « valider » 4. Le système lui donne une liste de nom avec leur prénom. 5. L’utilisateur clique sur le nom qu’il recherchait. 6. Le système affiche 2 zones d’édition « objet » et « corps du message » pour que la personne puisse lui envoyer un message. |
| Scénario  Alternatif | 4.1 le système ne trouve rien et lui informe avec un message. 4.2 user clique sur « ok ». 4.3 le système le renvoie à l’étape précédent. 6.1. le système trouve que l’user n’est pas dans sa session, il lui demande de se connecter à son espace avant de pouvoir envoyer un message à la personne. 6.2. user clique sur « ok ». 6.3. le système le renvoie à la page de connexion. |

**Cas d’utilisation : rechercher message**

|  |  |
| --- | --- |
| Cas d’utilisation | Rechercher un message |
| Acteur | User (l’abonné et l’admin) |
| Objectif | Retrouver un ou des message(s) précis |
| Précondition | Etre dans son espace privé et vouloir trouver un ou des messages |
| Poste-condition | Retrouver des messages dont on a besoin |
| Scénari | 1. Etre dans son espace privé. 2. User clique sur « rechercher un message ». 3. Le système lui ouvre une page de saisie où il doit taper sujet du message ou les mots qu’il retient dans le message recherché et il clique sur « rechercher ». 4. Le système lui donne les résultats trouvés |
| Scénari alternatif | 4.1 le système ne trouve rien et lui informe. 5. User clique sur « ok ». 6. Le système le ramène à la zone de recherche. |

**Cas d’utilisation : se connecter**

Description du processus

Après avoir entré l’url du site Retrouvaille.com, Au niveau du menu de navigation principal, tout utilisateur authentifié dispose d’un lien lui permettant de :

1. Affichage du formulaire de connexion au site Retrouvaille.com

2. Saisir les identifiants de connexion .

- Login (Email),

- Mots de passe.

|  |  |
| --- | --- |
| Cas d’utilisation | Se connecter |
| Acteurs principaux : | L’abonné et l’admin |
| Acteur secondaire | Système |
| Objectif | Accéder à son espace personnel |
| Pré-condition | Etre inscrit et que le compte soit actif |
| Poste-condition | Etre connecté à son espace personnel |
| Scénario | 1. L’admin ou l’abonné dispose un comte actif et dispose des éléments de connexions (login et mot de passe). 2. Il lance l’appli ou tape son URL dans un navigateur web, à la page d’accueil, on clique sur « se connecter ». 3. Le système ouvre la page de connexion avec les informations demandées. 4. La personne remplit les infos et clique sur « valider ». 5. Le système vérifie les informations et affiche l’espace privé de l’utilisateur. |
| Scénario alternatifs | 1. 5.1. Après vérification, le système constate que les informations saisies ne sont pas bonnes. Il informe l’utilisateur avec un message et l’invite à cliquer sur « ok » pour le renvoyer à l’étape 4. |

**Cas d’utilisation : Supprimer un abonne**

|  |  |
| --- | --- |
| Cas d’utilisation | Supprimerabonne |
| Acteur | L’admin |
| Objectif | Suprimmer un abonné de l’application |
| Précondition | Il faut que l’abonné existe dans l’application |
| Poste-condition | Se débarrasser d’un abonné qui n’a pas respecté les chartes de l’application ou un d’un abonné qui ne s’est pas connecté pendant très longtemps |
| Scénaris | 1. Se connecter à son espace privé en tant que admin. 2. Rechercher l’abonné en question, il le sélectionne et il clique sur « supprimer ». 3. Le système lui demande s’il veut vraiment supprimer cet abonné. 4. L’admin confirme en cliquant sur « ok ». 5. Le système supprime l’abonné et met à jour la base de données, puis informe l’admin avec un message. 6. L’admin clique sur « ok ». 7. Le système le ramène à son espace privé |
| Scénario alternatif | 3.1. le système n’a pas trouvé l’abonné recherché, il affiche l’information. 4. L’admin clique sur « ok ». 7. Le système le ramène à son espace privé |

**Cas d’utilisation : Supprimer un compte**

|  |  |
| --- | --- |
| Cas d’utilisation | Supprimer un compte |
| Acteur | L’abonné |
| Objectif | Ne plus être abonné de l’application |
| Précondition | Etre membre |
| Poste-condition | Supprimer toutes ses données de la plateforme |
| Scénarios | 1. Se connecter à son espace privé. 2. On clique sur « supprimer mon compte ». 3. Le système demande la confirmation. 4. User clique sur « ok ». 5. Le système demande à nouveau la confirmation. 6. l’abonné clique sur « je confirme » 7. Le système supprime le compte et le ramène à l’accueil |

**Cas d’utilisation : S’inscrire**

Description du cas\_creerCompte / s’inscrire

Après avoir entré l’url du site Retrouvaille.com, Au niveau du menu de navigation principal, tout internaute non authentifié dispose d’un lien lui permettant de :

1. Affichage du formulaire d’inscription au site Retrouvaille.com

2. Saisir les identifiants d’inscription.

|  |  |
| --- | --- |
| Cas d’utilisation | Créer compte / s’inscrire |
| Acteurs principaux | Internaute et Admin |
| Acteur secondaire | Système |
| Objectif | Avoir un espace personnel pour pouvoir retrouver les anciens stagiaires et disposer de nouvelles fonctionnalités ou créer un compte pour quelqu’un qui n’arrive pas à le faire. |
| Préconditions | Sauf l’admin, il ne faut pas disposer un compte auparavant dans l’application , vouloir devenir membre et trouver des amis et disposer d’un mail ou un numéro de téléphone |
| Poste-conditions | Disposer d’un compte avec son espace personnel, donner la possibilité à quelqu’un d’avoir un compte. |
| Scénarios | 1. Vouloir ouvrir un compte dans l’application. 2. Ouvrir l’application installée sur son mobile ou aller sur la toile et taper son URL, sur la page d’accueil, on clique sur s’inscrire. 3. Le système ouvre la page de connexion et vous demande des informations à saisir. 4. On remplit tous les champs avec les informations requises et on valide. 5. Le système vérifie et vous affiche la confirmation de la création de votre compte et vous redirige à la page de connexion. 6. L’admin reçoit une demande d’ouverture de compte venant de quelqu’un qui n’arrive pas tout seul à le faire. 7. Possédant toutes les informations de la personne pour créer son compte, il se connecte à son propre compte et clique sur « Créer un compte ». 8. Le système lui affiche la page d’inscription avec les informations demandées. 9. L’admin remplit le formulaire d’inscription et clique sur « valider ». 10. Le système lui affiche la confirmation de création de compte et lui invite à cliquer sur « ok » pour le renvoyer à son espace personnel. |
| Scénarios  alternatif | 1. 5.1 le système trouve que tous les champs ne sont pas remplis ou les informations fournies ne sont pas bonnes, il donne l’information et vous renvoie à l’étape 4. 7.1. L’admin n’a pas toutes les informations nécessaires pour créer le compte. Retour à l’étape 6 |

# **Choix des technologies :**

L’application a été développée sous Symfony 4.4 avec une architecture MVC.

Le choix de ce Framework PHP est dû à sa popularité mais surtout à sa puissance.

1.1) Symfony 4.4

Symfony est un Framework libre qui utilise le langage PHP avec une architecture MVC. J’ai choisi ce Framework notamment parce qu’il permet d’avoir une maintenance facile pour le site Web.

Les avantage de symfony 4.4 :

- Permet d’avoir une architecture MVC

- La rapidité des temps de réponse

- L’optimisation des URL permet d’avoir des URL distinctes pour chaque page

- Un système de configuration très facile avec langage de description YAML

- La documentation de Symfony est accessible

- L’utilisation de plugins

1.2) Bootstrap 5

J’ai utilisé le Framework de design Bootstrap pour rendre très attrayant le front office.

Bootstrap permet :

- D’avoir un large choix de thèmes

- Une feuille de style bien élaborée

- D’avoir des pages responsives.

1.3) HTML5 / TWIG

La combinaison du langage de balisage HTML le et moteur de Template pour le langage de programmation PHP TWIG m’ont permis d’avoir une bonne structure des pages Web.

1.4) JavaScript28

Le Javascript est très puissant et m’a donné la possibilité de rendre l’application plus attractive en jouant sur la dynamique des pages avec les animations mais aussi grâce à l’envoie et la récupération de données. Dans certains cas, des données s'actualisent en effet sans rechargement de la page grâce à des appels Ajax.

1.5) PostgreSQL

« PostgreSQL est un système de gestion de base de données relationnelle orienté objet puissant qui est en open source et qui est capable de prendre en charge en toute sécurité les charges de travail de données les plus complexes

1.6) Domaines et hébergement

Le site Web sera accessible à partir de l’URL https://www.Stagiaires-Guinot.com https://www.Stagiaires-Guinot.com et il y aura des sous domaines pour le backoffice. Le site sera hébergé dans la plateforme.

2. Environnement et architecture

2.1) Environnement

Pour la configuration du projet j’ai installé

- Php7.4

- Composer 2

- PostgreSQL

- Localhost

- Création d’un serveur web avec domaine https certifiée

- Serveur Apache v2.4

- Visual Code Studio

2.2) architecture

Le Framework Symfony repose sur la mise en place d'une architecture MVC.

Cette architecture définit un cadre d'organisation du code en accord avec sa nature.

Ce modèle permet une séparation du code en trois couches :

- La couche Modèle contenant le traitement logique des données (les accès à la base de données se trouvent dans cette couche).

- La Vue est la couche où interagit l'utilisateur (un moteur de Template fait partie de cette couche). Dans symfony, la couche vue est principalement faite de

Template PHP. Ces fichiers sont stockés dans les différents dossiers

Template/.

- Le Contrôleur est un morceau de code qui appelle le modèle pour obtenir certaines données qu'il passe à la Vue pour le rendu au client.

L'architecture Modèle/Vue/Contrôleur (MVC) est une façon d'organiser une interface

consiste à distinguer trois entités distinctes qui sont, le modèle, la vue et le contrôleur ayant chacun un rôle précis dans

l'interface. L'organisation globale d'une interface graphique est souvent délicate. L’architecture MVC est l’une des

architectures logicielles les plus utilisées pour les applications Web. Elle offre aussi un cadre pour structurer une

application. Elle se compose de 3 modules :

 Modèle : noyau de l’application qui gère les données, permet de récupérer les informations dans la base

de données, de les organiser pour qu’elles puissent ensuite être traitées par le contrôleur.

 Vue : composant graphique de l’interface qui permet de présenter les données du modèle à l’utilisateur.

 Contrôleur : composant responsable des prises de décision, gère la logique du code qui prend des

décisions, il est l’intermédiaire entre le modèle et la vue

(a) Présentation Framework Symfony

Symfony est un puissant Framework PHP respectant le design pattern MVC et utilisé pour développer des applications

web ou des sites web complexes, voire haut de gamme. Relativement simple d’accès, Symfony est un ensemble de

composants (ou « bibliothèques ») qui facilitent le développement web en réduisant de façon considérable le temps

et l’effort requis pour créer des composants génériques. Le langage de programmation PHP est celui qui est

majoritairement utilisé dans le cadre du développement web et des scripts de commande en ligne.

L’architecture MVC de Symfony :

Lorsqu'un client demande une URL, celle-ci est d'abord traitée par le contrôleur frontale de Symfony avant d'être

envoyée au noyaux.

Quand le noyaux reçois la demande d'une URL, il appelle le service de ROUTING, celui-ci va alors lui indiquer le

contrôleur à appeler pour l'URL que le client demande.

Le noyaux appel alors le contrôleur qui va si il en a besoin, appeler le modèle et générer la vue.

Quand le contrôleur a fini de générer la vue, il retourne le résultat au noyaux, celui-ci va alors transmettre la réponse

au client.

ORM : signifie Object Relationnel Mapping. Il permet de créer l'illusion d'une base de données orientée objet à partir

d'une base de données relationnelle, Il fournit la persistance transparente des objets PHP, et c’est l’interface qui permet

de faire le mapping entre nos objets et les éléments de la base de données. Il existe deux types d'ORM :

Data Mapped (Doctrine) : Permet une abstraction plus puissante avec la base de données, on envoie l’objet à

un manager pour le sauvegarder en base de données.

Un objet est une ligne de la base de données, chaque champs est une colonne, la

méthode save() est présente sur chaque objet.

L'arborescence d'un projet Symfony est la suivante :

Le backend admin, gestionnaire de notifications

EASYADMIN est un bundle de Symfony ou un module d’administration, c’est un outil qui permet de créer

des interfaces d’administration plus ou moins avancées. On peut réaliser des opérations SCRUD

(Search/Create/Read/Update/Delete) très facilement sur nos entités Doctrine (ORM seulement).

EASYADMIN est rapide et performant dans ses tâches, permettant une navigation fluide.

3. Outils de collaboration

On a utilisé différents outils de collaboration pour mieux réussir le travail d’équipe.

3.1) Git

« Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre créé par Linux ».

Git permet de créer différente branche pour une bonne répartition des tâches et des fonctionnalités de l’application,et pour une bonne gestion des versions.

Ce logiciel a permis également de suivre le niveau d’avancement du projet grâce aux Commits et aux Pull Request.

3.2) Trello

Trello est un outil de gestion de projet en ligne qui a facilité notre collaboration étant donné qu’il permet d’avoir un tableau de bord qui renseigne le travail réalisé, les contraintes, les dates limites, la répartition des taches Trello retrace le parcours du projet.

3.3) Skype

« Skype est un logiciel qui permet de discuter à travers le monde. Des millions d'individus et d'entreprises utilisent Skype pour effectuer des appels vidéo et des appels vocaux gratuits. »des réunions hebdomadaires ont était organisé grâce à Skype.

# **Planning de stage  :**

**AGENDA : Planning de stage Mars - Juin 2022.**

**Nom :** NABI M.

**Nom du projet :** APPLICATION DES ANCIENS RETROUVAILLES DE GUINOT.

**Planning de stage Mars - Juin 2022**

**Promotion : DWWM2 2022**

|  |  |  |  |
| --- | --- | --- | --- |
| **Mois** | **Date** | **Sem** | **Outils** |
| Février | Lundi 28 février 2022. | 01 | Mise en place des outils de travail |
| Mars | Lundi 07 mars 2022. | 02 | Recherche documentaire |
| Mars | Lundi 14 mars 2022. | 03 | Recherche documentaire |
| Mars | Lundi 22 mars 2022. | 04 | Descriptif + cahier des charges de l'application |
| Mars | Lundi 29 mars 2022. | 05 | **UML & Gestion de Projet**   * Analyse 1 (User Case) |
| Avril | Lundi 05 avril 2022. | 06 | **UML & Gestion de Projet**   * Analyse 2 (Classe/Objects) |
| Avril | Lundi 12 avril 2022. | 07 | **Analyse**   * Analyse 3 (Séquences & Others) |
| Avril | Lundi 19 avril 2022. | 08 | **Jeu d'essai**   * Mise en place du Socle * Création et test des Fixtures |
| Avril | Lundi 26 avril 2022. | 09 | **Jeu d'essai**   * Création et test des Fixtures (Essai Data Guinot) |
| Mai | Lundi 03 mai 2022. | 10 | Printemps |
| Mai | Lundi 10 mai 2022. | 11 | **Code et requêtes**   * SQL / * Entity / Test Unitaires |
| Mai | Lundi 17 mai 2022. | 12 | **Code et requêtes**   * Coding |
| Mai | Lundi 24 mai 2022. | 13 | **Code et requêtes**   * Coding |
| Mai | Lundi 31 mai 2022. | 14 | **Code et requêtes**   * Coding |
| Juin | Lundi 07 juin 2022. | 15 | **Codes + Rédaction du dossier technique**   * Coding, * Word |
| Juin | Lundi 14 juin 2022. | 16 | **Codes + Rédaction du dossier technique**   * Excel |
| Juin | Lundi 21 juin 2022. | 17 | Rédaction du dossier technique |

# **Base de données et jeux de données Sur Symfony :**

La base de données est assez simple à créer, on doit configurer le fichier ‘.emv’ à la ligne 28. On y ajoute le nom de la base de données.

Plus besoin de créer un script de création en sql long et fastidieux.

Après la configuration du fichier ‘.emv’ , il nous suffit d’utiliser la ligne de commande et tapez la commande ‘php bin/console doctrine :database :create’. Cette commande créera automatiquement une base de données qui sera accessible via le gestionnaire de base de données « php my admin ».

On souhaite mettre en place un jeu de donner pour pouvoir tester l’application.

Le jeu de données est un moyen factice pour tester notre application. Il nous permettra d’avoir de fausses données pour alimenter notre base de données et de tester l’application.

Pour commencer, on doit installer le bundle datafixtures.

Un bundle est une extension de Symfony qui nous permet d’utiliser une fonctionnalité, dont on a besoin, sans implémenter beaucoup de codes.

Pour l’installer, on devra taper en ligne de commande « Composer require --dev orm-fixtures ». Cela va nous créer un dossier datafixtures dans le Src.

Ensuite, on doit créer un fichier php DataFixtures qui reprendra l’ensemble des caractéristiques de chaque entité pour les remplir. Pour la créer, on doit taper la commande « php bin/console make :fixtures ».

Une fois nos fixtures créées, nous devrons installer le bundle faker qui nous génèrera des fixtures factices et réalistes dans la base de données.

Pour l’installer, on devra taper : « composer require fzaninotto/faker ».

Une fois le bundle faker installé, nous devons créer nos jeux de données, en ‘’settant’’ les différents champs de nos entités. Setter un champ, signifie que nous attribuons une valeur à un champ pour créer un enregistrement.

En utilisant, comme je l’ai fait, une boucle, on peut obtenir le nombre souhaité d’enregistrements.

Une fois fini, nous devons grâce à doctrine, pousser le tout vers la base de donné créé précédemment.

Qu’est-ce donc doctrine ?

Doctrine est un ORM (couche d'abstraction à la base de données) pour PHP, ...

Doctrine est l'ORM par défaut du framework Symfony (depuis la version 1.3 de ce framework). Cependant son utilisation dans le cadre d'un projet développé avec Symfony est option

C’est quoi ORM ?

ORM signifie Object-Relationnal Mapper. Un ORM sert à offrir une couche d’abstraction de connexion à toutes les BD relationnelles (comme PDO) mais aussi des facilités pour réaliser les requêtes courantes sans descendre au niveau des requêtes SQL et pour générer automatiquement des entités dans le langage utilisé avec les getters et setters correspondants.

Après, on devra faire la persistance des données. Cela veut dire que doctrine prépare et sauvegarde les données. Ensuite, on fait le flush pour envoyer les données de doctrine vers la base de données. Puis, on devra saisir en ligne de commande : « php bin/console doctrine :fixtures :load ».

a) Jeu de données.

Voici l’ensemble du code permettant de généré des fixtures

<?php

namespace App\DataFixtures;

use App\Entity\Messages;

use App\Entity\Utilisateurs;

use Faker;

use Faker\Factory;

use Doctrine\Persistence\ObjectManager;

use Doctrine\Bundle\FixturesBundle\Fixture;

use Symfony\Component\Security\Core\Role\Role;

use Symfony\Component\Validator\Constraints\DateTime;

class UtilisateursFixtures extends Fixture

{

    public function load(ObjectManager $manager): void

    {

        $faker = Faker\Factory::create('fr\_FR');

        // Liste des utilisateurs :

        for ($u = 1; $u < 20; $u++)

        {

            $utilisateurs = new Utilisateurs();

            $civilite = ["Femme", "Homme"];

            shuffle($civilite);

            $status = ['connecter', 'déconnecter', 'anonyme'];

            shuffle($status);

    $roles = ['ROLE\_ANONYMOUSE', 'ROLE\_USER', 'ROLE\_ADMIN', 'ROLE\_ABONNER'];

            shuffle($roles);

$email = $faker->email;

            $utilisateurs

                ->setCivilite($civilite[0])

                ->setNom($faker->lastName)

                ->setPrenom($faker->firstName)

                        ->setDateDeNaissance(new \DateTime())

        ->setAdresse($faker->address)

        ->setEmail($email)

        ->setIsVerified(true)

        ->setLogin($email)

        ->setPassword($faker->password())

                ->setVilles($faker->city)

                                ->setCodePostal($faker->postcode)

                ->setPays($faker->city)

                ->setTelephone($faker->phoneNumber)

                    ->setStatus($status[0])

                ->setRoles($roles[0]);

            $manager->persist($utilisateurs);

        // Liste des messages :

        for ($m=1; $m<20 ; $m++ )

    {

        $status = ['Nouveau Message', 'Ancien message', 'Message Privé', 'Message Annonyme', 'Message Archivé', 'Message Professionnel', 'Message Publicitaire'];

        shuffle($status);

        $pays = ['France', 'Espagne', 'Royaume-Uni', 'Italie', 'Allemagne', 'Russie', 'Nigeria', 'Danemark', 'Irlande', 'Argentine', ];

        shuffle($pays);

$utilisateurs = ['Gilles', 'Paul', 'Marie', 'Richard', 'Bernard', 'Julie', 'Omar', 'Sidi', 'Pierre', 'Caroline'];

shuffle($utilisateurs);

            $messages = new Messages();

            $messages

            ->setTitreMessage($faker->name)

            ->setContenuMessage($faker->sentence())

            ->setCreerDate(new \DateTime());

                        $manager->persist($messages);

                    }

       $manager->flush();

    }

    }

}

**Authentification :**

Création de l'entité User :

Php bin/console make:user User

Do you want to store user data in the database (via Doctrine)? (yes/no) [yes]:

> yes

Enter a property name that will be the unique "display" name for the user (e.g. email, username, uuid) [email]:

> email

Will this app need to hash/check user passwords? Choose No if passwords are not needed or will be checked/hashed by some other system (e.g. a single

sign-on server).

Does this app need to hash/check user passwords? (yes/no) [yes]:

> yes

**Explications des lignes :**

> Do you want to store user data in the database (via Doctrine)? (yes/no) [yes]

> - Est-ce que vous souhaitez enregistrer les utilisateurs en base de données avec Doctrine ? Oui bien sûr ! `yes`

> Enter a property name that will be the unique "display" name for the user (e.g. email, username, uuid) [email]:

> - Quelle sera la propriété de votre entité User qui serviva de login ?

On choisit `email`

> Will this app need to hash/check user passwords? Choose No if passwords are not needed or will be checked/hashed by some other system (e.g. a single sign-on server).

> Does this app need to hash/check user passwords? (yes/no) [yes]:

> - Est-ce que notre application a besoin de hasher les mots de passe (

c'est à dire les chiffrés) ? Évidemment oui ! `yes`

**Formulaire de création d'utilisateurs :**

Php bin/console make:registration-form

Creating a registration form for App\Entity\User

Do you want to add a @UniqueEntity validation annotation on your User class to make sure duplicate accounts aren't created? (yes/no) [yes]:

Do you want to automatically authenticate the user after registration? (yes/no) [yes]:

**Explication des lignes :**

> Do you want to add a @UniqueEntity validation annotation on your User class to make sure duplicate accounts aren't created? (yes/no) [yes]:

> - Est-ce que l'on souhaite que nos utilisateurs soient uniques sur le champ de login, leur e-mail donc ? Oui ! `yes`

> Do you want to automatically authenticate the user after registration? (yes/no) [yes]:

> - Est-ce que l'on veut que nos utilisateurs soient logués automatiquement après la création du compte ? Oui ! `yes`

Migration :

Maintenant que l'entité est créée, on peut faire une migration pour créer la table en base de données :

Php bin/console doctrine:migrations:migrate

bin/console make:migration

bin/console doctrine:migrations:migrate

Unable to generate a URL for the named route "" as such route does not exist.

// AbstractController->redirectToRoute('') in src/Controller/RegistrationController.php (line 39)

    $entityManager->persist($user);

    $entityManager->flush();

    // do anything else you need here, like send an email

    return $this->redirectToRoute('');

}

C'est en fait vraiment très clair : nous devons modifier `RegistrationController.php` (un des fichiers créés par `make:registration-form`) et lui indiquer, à la ligne qui nous est donnée, où nous devons rediriger l'utilisateur après qu'il se soit créé un compte.

Idéalement vers une page compte utilisateur par exemple, mais comme nous n'en avons pas, redirigeons-le vers la page d'accueil.

Remarque : le nom de la route (dans `Home` pour rappel) est `accueil\_index`.

Dans `RegistrationController.php` en ligne 39 :

return $this->redirectToRoute('');

Nous venons de créer notre premier compte utilisateur et nous l'avons rediriger vers la page d'accueil.

Remarque : Il faut bien prendre le temps d'étudier tous les nouveaux fichiers qui ont été créés par 'make:registration-form'. En effet, la création de User est plutôt complexe car beaucoup de choses entrent en compte (des sessions, le hashage des mots de passe...). Beaucoup de choses sont à apprendre des fichiers créés par Symfony.

**Gérer le login d'utilisateurs :**

Maintenant que nos utilisateurs peuvent créer un compte, il va falloir les loguer :

php bin/console make:auth

What style of authentication do you want? [Empty authenticator]:

[0] Empty authenticator

[1] Login form authenticator

> 1

The class name of the authenticator to create (e.g. AppCustomAuthenticator):

> LoginFormAuthenticator

Choose a name for the controller class (e.g. SecurityController) [SecurityController]:

Do you want to generate a '/logout' URL? (yes/no) [yes]:

**Explication des lignes :**

> What style of authentication do you want? [Empty authenticator]:

> - Symfony nous demande si il faut gérer l'authentification par un formulaire (c'est ce que nous voulons) ou par autre chose (login avec Google, Facebook, via une API...).

Ici, on remplit : `1`

> The class name of the authenticator to create (e.g. AppCustomAuthenticator):

> - Symfony va créer un service qui va gérer les méthodes d'authentification. Appelons-le : `LoginFormAuthenticator`

> Choose a name for the controller class (e.g. SecurityController) [SecurityController]:

> - Symfony va créer un controller qui va gérer les routes d'authentification (/login par exemple).

Apellons le avec le nom par défaut: `SecurityController`

> Do you want to generate a '/logout' URL? (yes/no) [yes]:

> - Est-ce que l'on veut une route /logout créée automatiquement ? Oui ! `yes`

Corrigeons maintenant l'erreur qui a été générée. Comme pour `/register`, l'erreur est qu’on n'a pas indiqué de page vers laquelle rediriger

un utilisateur qui vient d'être logué.

Dans `LoginFormAuthenticator.php`, il faut modifier la ligne 98 :

// For example : return new RedirectResponse($this->urlGenerator->generate('some\_route'));

throw new \Exception('TODO: provide a valid redirect inside '.\_\_FILE\_\_);

il faut remplacé par ce qui est indiqué en commentaire, avec une route valide... la page d'accueil par exemple : ‘accueil\_index’.

return new RedirectResponse($this->urlGenerator->generate('accueil\_index'));

Remarque : Il faut toujours actualiser, pour vérifier si on est connectés, il faut vérifier dans la barre de débug de Symfony : l'email de l'utilisateur devrait apparaître.

On allant sur '/logout' pour se déloguer,

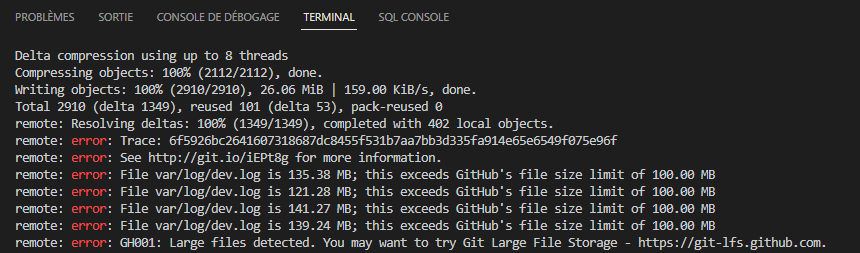
ça devrait remplacer l'adresse mail par "anon." (comme anonyme) .

Lors de mon énième push sur mon dépôt distant sur Git Hub, j’ai eu un message d’erreur.

remote:

error: File var/log/dev.log is 135.38 MB;

this exceeds GitHub's file size limit of 100.00 MB



GitHub limite la taille des fichiers autorisés dans les référentiels.

Pour suivre les fichiers au-delà de cette limite, nous pouvons utiliser Git Large File Storage. Avec Git LFS activé, nous pourrons récupérer, modifier et pousser des fichiers volumineux avec n'importe quel fichier géré par Git.

# **Documentation en Anglais :**

**An open source Git extension for versioning large files :**

Git Large File Storage (LFS) replaces large files such as audio samples, videos, datasets, and graphics with text pointers inside Git, while storing the file contents on a remote server like GitHub.com or GitHub Enterprise.

Download v3.2.0 (Windows)

Git LFS security update:

Windows users should update to 3.1.3 or newer.

**Getting Started :**

1. Download and install the Git command line extension. Once downloaded and installed, set up Git LFS for your user account by running:

git lfs install

You only need to run this once per user account.

2. In each Git repository where you want to use Git LFS, select the file types you'd like Git LFS to manage (or directly edit your .gitattributes). You can configure additional file extensions at anytime.

git lfs track "\*.psd"

Now make sure .gitattributes is tracked:

git add .gitattributes

Note that defining the file types Git LFS should track will not, by itself, convert any pre-existing files to Git LFS, such as files on other branches or in your prior commit history. To do that, use the git lfs migrate[1] command, which has a range of options designed to suit various potential use cases.

3. There is no step three. Just commit and push to GitHub as you normally would; for instance, if your current branch is named main:

git add file.psd

git commit -m "Add design file"

git push origin main

Check out our wiki, discussion forum, and documentation for help with any questions you might have!

Git LFS is an open source project

**Extension Git pour la gestion des versions de fichiers volumineux :**

lien : https://docs.github.com/en/repositories/working-with-files/managing-large-files

Git Large File Storage (LFS) remplace les fichiers volumineux tels que les échantillons audio, les vidéos, les ensembles de données et les graphiques par des pointeurs de texte dans Git, tout en stockant le contenu du fichier sur un serveur distant, tel que GitHub.com ou GitHub Enterprise.

Télécharger v3.2.0 (Windows)

install git-lfs

**Commencer**

1. Téléchargez et installez l'extension de ligne de commande Git.

Une fois téléchargé et installé, configurez Git LFS pour votre compte utilisateur en exécutant : Installation de git lfs

Vous n'avez besoin de l'exécuter qu'une seule fois par compte d'utilisateur.

1. Dans chaque référentiel Git où vous souhaitez utiliser Git LFS, sélectionnez les types de fichiers que vous souhaitez que Git LFS gère (ou modifiez directement vos attributs.git). Vous pouvez configurer des extensions de fichiers supplémentaires à tout moment.

git lfs track "\*.psd"

Assurez-vous maintenant que .gitattributes est suivi :

git add .gitattributes

Notez que la définition des types de fichiers que Git LFS doit suivre ne convertira pas, par elle-même, les fichiers préexistants vers Git LFS, tels que les fichiers sur d'autres branches ou dans votre historique de commits antérieur.

Pour ce faire, utilisez la commande : git lfs migrate[1], qui propose une gamme d'options conçues pour s'adapter à divers cas d'utilisations potentiels.

3. Il n'y a pas de troisième étape. Validez et poussez simplement vers GitHub comme vous le feriez normalement ; par exemple, si votre branche actuelle s'appelle main :

git add file.psd

git commit -m "Ajouter un fichier de conception"

git push origin principal

lien : https://docs.github.com/en/repositories/working-with-files/managing-large-files

**Remarque :**

Mise à jour de sécurité Git LFS : Les utilisateurs Windows doivent effectuer la mise à jour vers 3.1.3 ou une version plus récente

## **Mot du référent du stage :**

* Préface

Je suis heureux d'avoir connu monsieur NABI Mohamed qui m'a fait découvrir son centre de formation et en particulier sa salle des machines conçues spécialement pour les stagiaires non-voyants. J'ai éprouvé beaucoup de plaisir à lui proposer de rajouter dans son projet de fin d'études une plate forme LMS pour notre école BIOGESOFT qui est un organisme de formation professionnelle en bioinformatique et informatique de Gestion et Réseaux. Monsieur Nabi a beaucoup de volonté et surtout il a acquis une formation très solide au sein de son centre de formation à Villejuif. Il m'a été donné l'occasion de discuter avec ses formateurs et ses responsables qui sont très compétents et tous prêts à l'aider à aller jusqu'au bout de son projet.

Le savoir faire de monsieur NABI me sera certainement utile pour développer au sein de BIOGESOFT un pôle de formation Web spécialement pour les candidats ayant une déficience visuelle.

Je trouve en lui un collaborateur motivé et agréable.

Mr saidani. Directeur des études de BIOGESOFT.

## **Conclusion :**

étant stagiaire non-voyant j’ai intégré un ensemble de normes d’accessibilité qui

m’ont permis d’utiliser l’application comme tout autre public mais aussi qui ont permis à

de pouvoir avoir un complément d’accessibilité., cela m’a semblé

particulièrement important car j’ai pu utiliser mon handicap et mon expérience du quotidien

concernant mon utilisation de internet et plus particulièrement des différents sites web.

De ce fait,

Mon désire était de mettre en œuvre sur le terrain tout ce que j’ai appris

Afin de contribuer à l’amélioration de l’accessibilité numérique pour les non-voyants qui sont dans mon cas.