



المدرسة العليا للتكنولوجيا - الفخيرة
École Supérieure de Technologie - Kénitra

PROJET DE STAGE D'INITIATION

FILIERE : GENIE INFORMATIQUE

ENTREPRISE PARTENAIRE : GEAR9

Développement d'une application informatisée de réseau social

Réalisé par : *Mohamed Khalil Assaddiki*

Encadré par : *M. Hassan Tlemcani*

Soutenu le 30/12/2023 devant le jury composé de :

Pr. Azouaoui Ahmed

Professeur à l'ESTK

Pr. El Ghayam Yassine

Professeur à l'ESTK

Année universitaire : 2023/2024

Dédicace

Avant de poursuivre avec cette présentation, je tiens à dédier cet effort à mes parents, mes frères et ma sœur, qui ont été une source constante de motivation et de soutien tout au long de ce stage, ainsi que tout au long de ma vie. Je souhaite également exprimer ma gratitude envers mes professeurs et mon encadrant ainsi que toutes les personnes qui ont contribué, de près ou de loin, à cette expérience et qui ont cru en moi. Votre encouragement et votre confiance ont été des éléments essentiels de cette réussite.

Remerciement

Je tiens à exprimer ma sincère gratitude envers Hassan Tlemcani, mon encadrant chez Gear9 à Casablanca, ainsi qu'envers toute l'équipe Gear9. Leur accueil chaleureux et leur soutien inestimable ont rendu mon stage enrichissant et stimulant. Cette expérience m'a permis de développer des compétences et d'apprendre un tas de nouvelles technologies.

Je souhaite également remercier mes professeurs à l'Ecole Supérieur de Technologie à Kenitra pour leur enseignement de qualité qui m'a bien préparé aux défis qui m'attendait. Ce stage représente une étape clé dans mon parcours, et je suis impatient d'appliquer ce que j'ai appris dans ma future carrière. Je suis reconnaissant envers tous ceux qui ont contribué à cette expérience mémorable et qui ont eu confiance en mon potentiel.

Sommaire

Liste des Figures	6
Liste des tableaux.....	8
Chapitre I : Contexte général du projet.....	9
1.Introduction	9
2.Presentation de l'entreprise.....	9
2.1. A propos de l'entreprise.....	9
2.2. Les Approches et Technologies Employées par Gear9.....	10
3. Contexte général du projet.....	12
3.1. Etude général du projet.....	12
3.2. Objectifs du projet.....	12
Chapitre II : Analyse et Conception.....	13
1. Méthode Merise.....	13
2. Modèle conceptuel de données.....	14
2.1.Le modèle	14
2.2.Lecture.....	14
3. Modèle Logique de données.....	16
4. Modèle conceptuel de traitement.....	17
5. Conclusion.....	18
Chapitre III : réalisation de l'application.....	18
1. Introduction.....	18
2. Langages et Frameworks utilisées.....	19
2.1. Front-end.....	19
2.2. Back-end.....	21

2.3. Version Control.....	21
3. Outils utilisées.....	22
4. Techniques utilisées	23
5. Création de l'application.....	24
5.1. Page de connexion (index.php).....	25
5.1.1. Cas de création du compte.....	26
5.1.2. Cas d'authentification.....	29
5.2. La page de profil.....	31
5.3. La page des utilisateurs (People)	37
5.4. Page du Chat.....	40
6. Conclusion.....	41
Conclusion General.....	41
Bibliographie.....	42

Liste des Figures

Figure 1 : Schéma explicatif de la méthode Agile.....	10
Figure 2 : Les étapes de la méthode Merise.....	13
Figure 3 : Modèle conceptuel de données (MCD).....	14
Figure 4 : Modèle logique de données (MLD).....	16
Figure 5 : MCT.....	17
Figure 6 : Logo HTML.....	19
Figure 7 : Logo CSS.....	19
Figure 8 : Logo JavaScript.....	20
Figure 9 : Logo PHP.....	21
Figure 10 : Logo Git.....	22
Figure 11 : Logo GitHub.....	22
Figure 12 : Logo VS Code.....	22
Figure 13 : Logo XAMPP.....	23
Figure 14 : arbre d projet.....	24
Figure 15 : Page de Connection au réseau social.....	25
Figure 16 : Extrait du formulaire HTML.....	26
Figure 17 : Fichiers SignUp selon MVC.....	27
Figure 18 : Extrait du code Connection.php.....	27
Figure 19 : Création du compte dans la DB.....	28
Figure 20 : Message de succès	28
Figure 21 : Fichiers SignIn selon MVC.....	29
Figure 22 : Page de réinitialisation du MDP.....	30
Figure 23 : Page de création du nouveau MDP.....	30

Figure 24 : Extrait du code SignIn.....	31
Figure 25 : Code PHP pour l'authentification.....	31
Figure 26 : Navbar version PC.....	32
Figure 27 : Navbar Tablet/Mobile (fermée).....	32
Figure 28 : Navbar Tablet/Mobile (ouvert).....	32
Figure 29 : Profile version PC.....	33
Figure 30 : Profile Tablet/Mobile (fermée).....	33
Figure 31 : Profile Tablet/Mobile (ouvert).....	33
Figure 32 : ChangeData Page.....	34
Figure 33 : Button pour ajouter un post.....	35
Figure 34 : Formulaire pour le partage d'un post.....	35
Figure 35 : Aperçu du post.....	36
Figure 36 : Détails du post	36
Figure 37 : Les utilisateurs du réseau.....	37
Figure 38 : Le profil d'utilisateur voulu.....	37
Figure 39 : Envoi d'une demande d'ami.....	38
Figure 40 : Alerte notification.....	38
Figure 41 : Page notification.....	38
Figure 42 : Réception de la demande.....	39
Figure 43 : Amitié crée.....	39
Figure 44 : Messages non lues	40
Figure 44 : Envoi Messages	40

Liste des tableaux

Tableau 1 : Tableau des technologies.....	10
---	----

Chapitre I : Contexte général du projet

1. Introduction :

Dans le cadre du stage d'observation et afin de mieux comprendre les pratiques et les méthodologies adoptées par les entreprises informatiques, j'ai eu l'opportunité de rejoindre **Gear9**, une société spécialisée dans les solutions digitales qui exploite les technologies de pointe pour satisfaire ses clients. Cette expérience a été extrêmement enrichissante, me permettant d'acquérir une vision plus approfondie du domaine informatique et de son fonctionnement.

2. Présentation de l'entreprise :

2.1. A propos de l'entreprise :

Gear9 est une entreprise dédiée aux solutions digitales, qu'il s'agisse de sites web, d'applications mobiles, de services cloud, ou d'autres innovations numériques. Fondée en 2018, cette société a su réunir une équipe coopérative et attentive aux besoins de ses clients, ce qui se traduit par la création d'expériences uniques et captivantes. Cette approche a permis à Gear9 de bâtir une réputation inégalée dans son domaine.

Gear9 se distingue également par son engagement constant dans l'adoption des nouvelles technologies. L'entreprise investit régulièrement dans les dernières avancées technologiques pour garantir à ses clients des solutions à la pointe de l'innovation. De plus, Gear9 privilégie l'utilisation de la méthodologie agile, ce qui se traduit par une meilleure performance et une plus grande productivité dans la réalisation de projets, contribuant ainsi à son succès continu sur le marché de la transformation digitale.

2.2. Les Approches et Technologies Employées par Gear9 :

- **Concernant les méthodologies**, Gear9 privilégie essentiellement la **méthode Agile**, une approche adaptée au développement logiciel qui favorise la flexibilité et la productivité. Plutôt que de planifier l'intégralité du projet dès le départ, cette méthode divise le travail en itérations appelées "sprints", généralement d'une durée de deux à quatre semaines. À la fin de chaque sprint, une version fonctionnelle du produit est livrée, permettant ainsi aux équipes de recueillir des retours et de s'adapter aux changements. L'Agilité met l'accent sur la communication continue avec les parties prenantes, la priorisation des tâches en fonction de la valeur pour le client, ainsi que la capacité à réagir rapidement aux besoins changeants du projet.

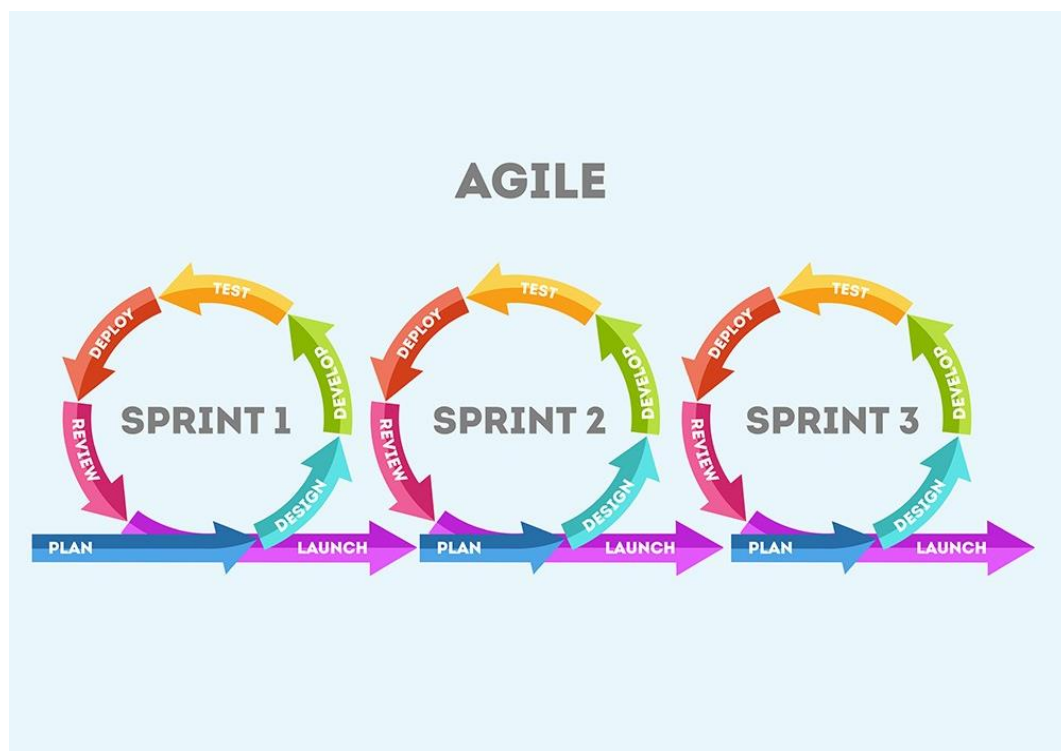


Figure 1 : Schéma explicatif de la méthode Agile

- **Pour la partie technique**, Gear9 exploite une variété de technologies, que ce soit en **front-end**, en **back-end**, en **développement mobile**, en DevOps ou en **gestion des données (Data)**.

Le tableau ci-dessous en donne un aperçu détaillé :

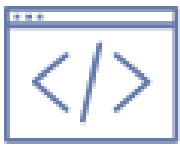




 Front-end	 Back-end	 Mobile	 Data	 DevOps
+ React JS + Vue JS + Next JS	+ PHP +Node JS +Java / JEE	+ React native + Swift + Kotlin	+ Python + BigQuery	+ Docker + Amazon Web Services

Tableau 1 : Tableau des technologies

3. Contexte général du projet :

3.1. Etude du projet :

Le projet se présente sous la forme d'une application de réseau social qui permet aux utilisateurs de créer leur propre compte personnel. Une fois le compte créé, les utilisateurs ont la possibilité d'ajouter des amis, d'envoyer des messages, de partager des photos dans leur espace personnel, et de les partager avec leurs amis.

3.2. Objectifs du projet :

Le projet consiste en la conception d'une application de réseau social destinée à offrir une expérience de communication en ligne conviviale et enrichissante.

Les objectifs de ce projet sont les suivants :

- ✓ Faciliter la création de liens sociaux.
- ✓ Permettre aux utilisateurs de rester connectés.
- ✓ Permettre le partage des photos avec les amis.
- ✓ Garantir la sécurité des données personnelles des utilisateurs.
- ✓ Offrir une expérience d'utilisation fluide et intuitive.
- ✓ Créer une interface attrayante.

L'objectif ultime est de fournir une plateforme qui favorise la confidentialité et la sécurité ainsi que des interactions sociales positives tout en offrant une expérience utilisateur exceptionnelle.

Chapitre II : Analyse et Conception

1. Méthode Merise :

La méthode MERISE (Méthode d'Étude et de Réalisation Informatique pour les Systèmes d'Entreprise) est une méthode de conception de systèmes d'information largement utilisée dans le domaine de l'informatique et de la gestion. Elle repose sur une approche structurée de la conception des systèmes d'information. Elle propose un ensemble d'outils, de techniques et de diagrammes pour modéliser les différentes étapes de la conception, de l'analyse des besoins à la réalisation concrète du système. Parmi les éléments clés de la méthode MERISE, on trouve les concepts de modèle entité-association pour la modélisation des données, les diagrammes de flux pour la modélisation des processus métier, et les diagrammes de structure pour la modélisation des structures de données.

La méthode Merise se pratique selon le schéma suivant :

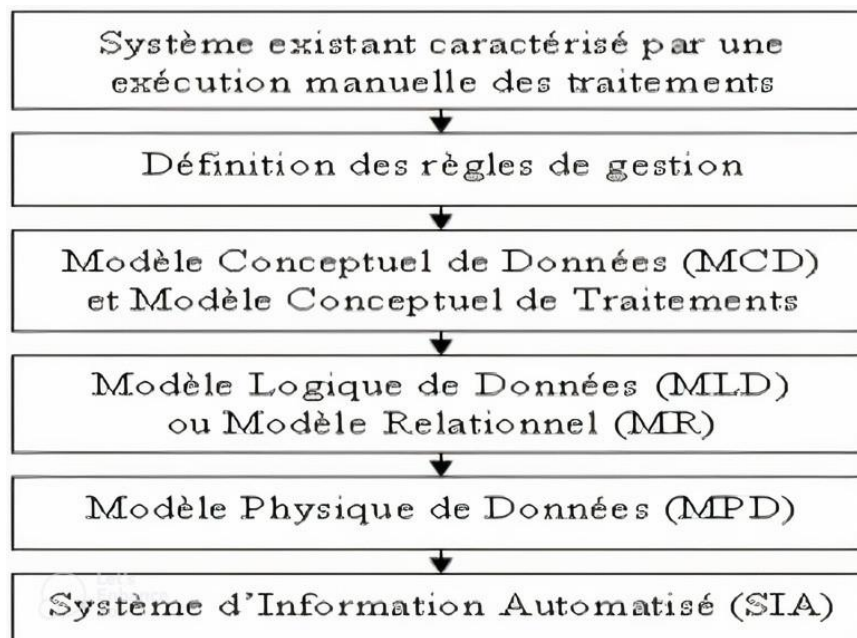


Figure 2 : Les étapes de la méthode Merise

2. Modèle Conceptuel de données :

2.1. Le modèle :

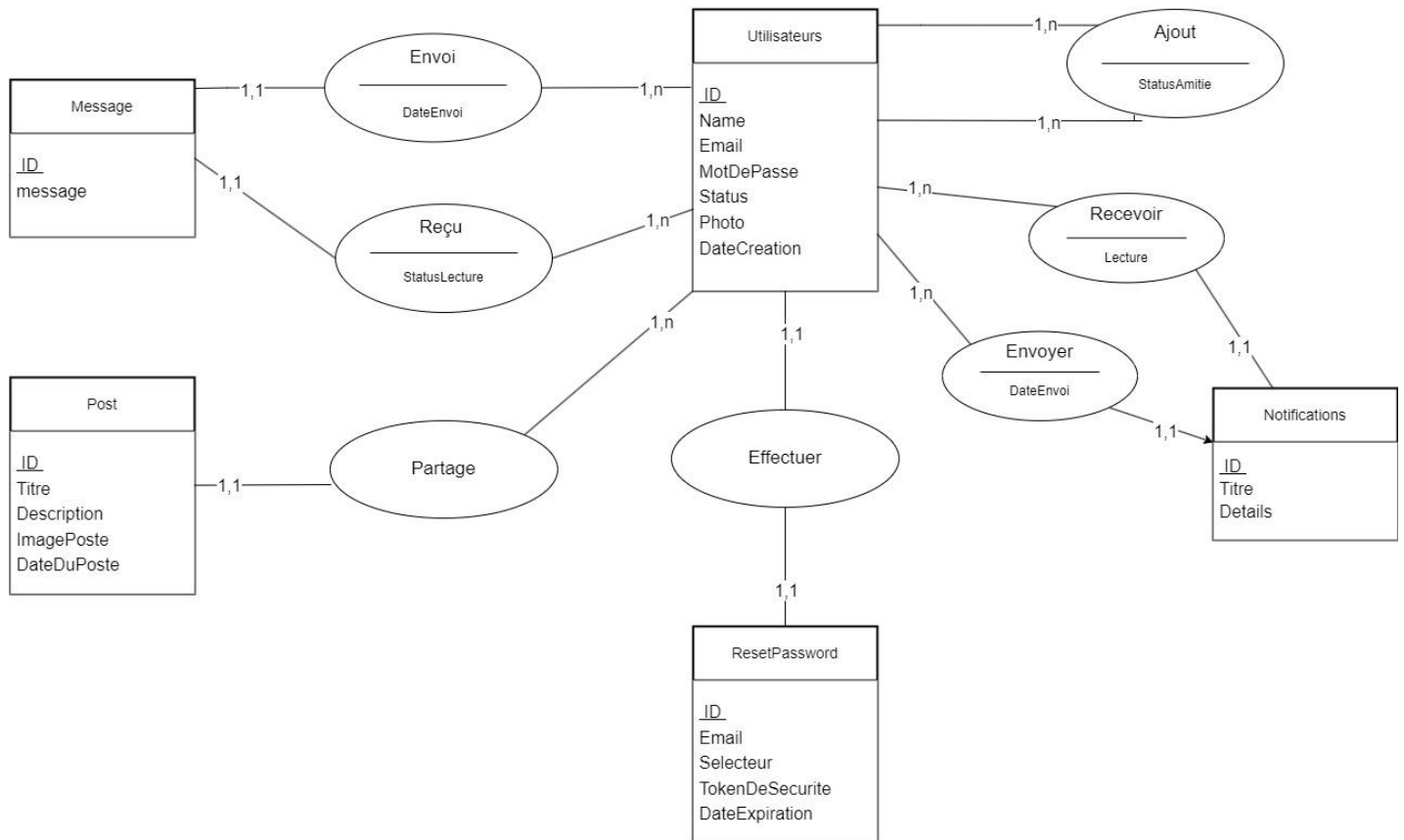


Figure 3 : modèle conceptuel de Données (MCD)

2.2. Lecture :

Association Envoi :

- Un message a été envoyé par un et un seul utilisateur durant la Date d'envoi
- Un utilisateur a envoyé un ou plusieurs messages durant la Date d'envoi

Association Reçu :

- Un message a été reçu par un et un seul utilisateur avec un statu lecture
- Un utilisateur reçoit un ou plusieurs messages un statu lecture

Association Partage :

- Un post a été partagé par un et un seul utilisateur
- Un utilisateur a partagé un ou plusieurs postes

Association Envoyer :

- Une notification a été envoyé par un et un seul utilisateur pendant une date d'envoi
- Un utilisateur a envoyé une ou plusieurs notifications pendant une date d'envoi

Association Recevoir :

- Une notification a été reçu par un et un seul utilisateur avec un statuts de lecture
- Un utilisateur a reçu une ou plusieurs notifications utilisateur avec un statut de lecture

Association Ajout :

- Un utilisateur a ajouté un ou plusieurs utilisateurs avec un statut déterminant la nature de l'amitié
- Un utilisateur a été ajouté par un ou plusieurs utilisateurs avec un statut déterminant la nature de l'amitié

Association Effectuer :

- Un utilisateur a effectué un et un seul reset du mot de passe
- Un reset du mot de passe a été effectué par un et un seul utilisateur

3. Modèle logique de données :

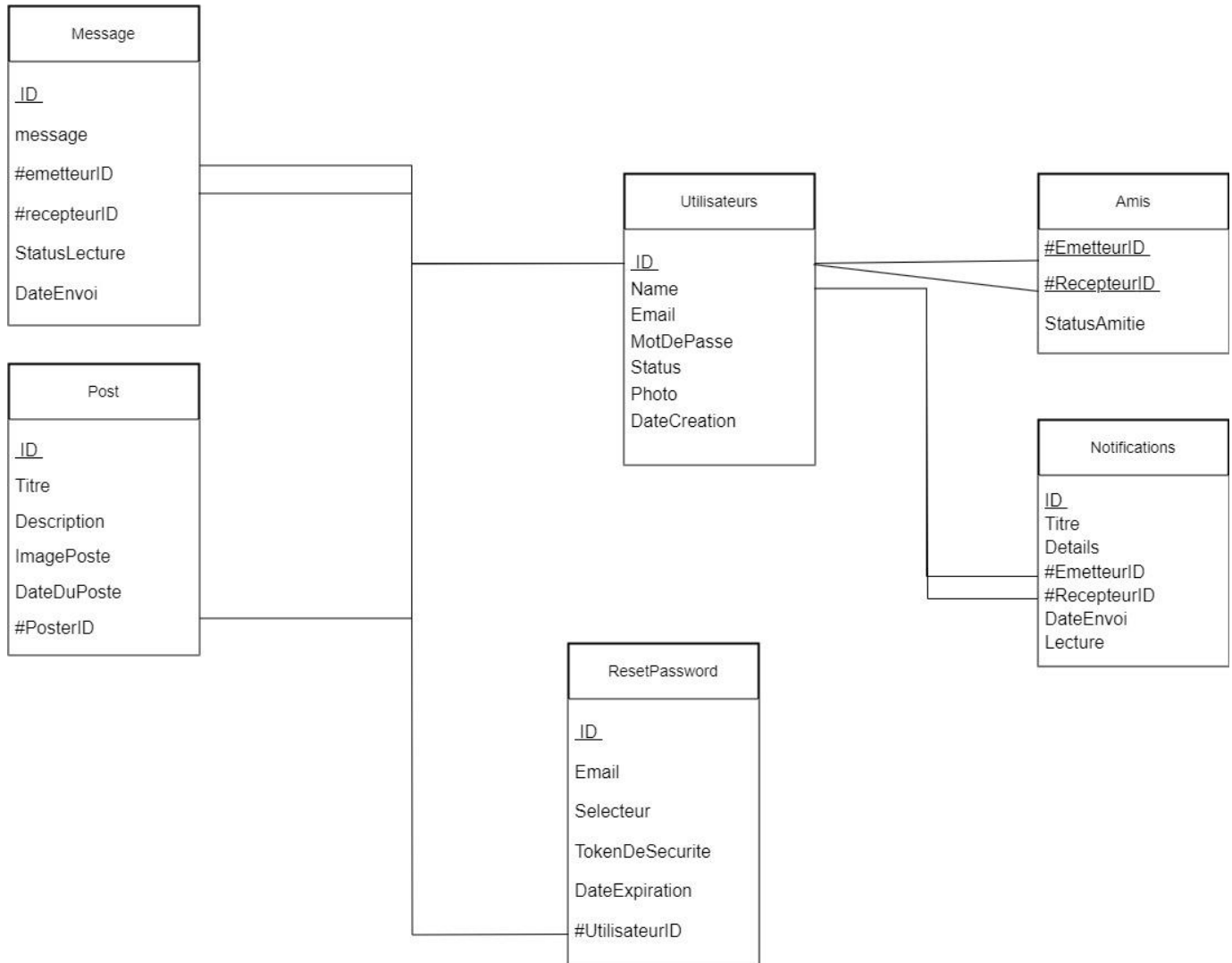


Figure 4 : Modèle Logique de données (MLD)

4. Modèle Conceptuel de traitement :

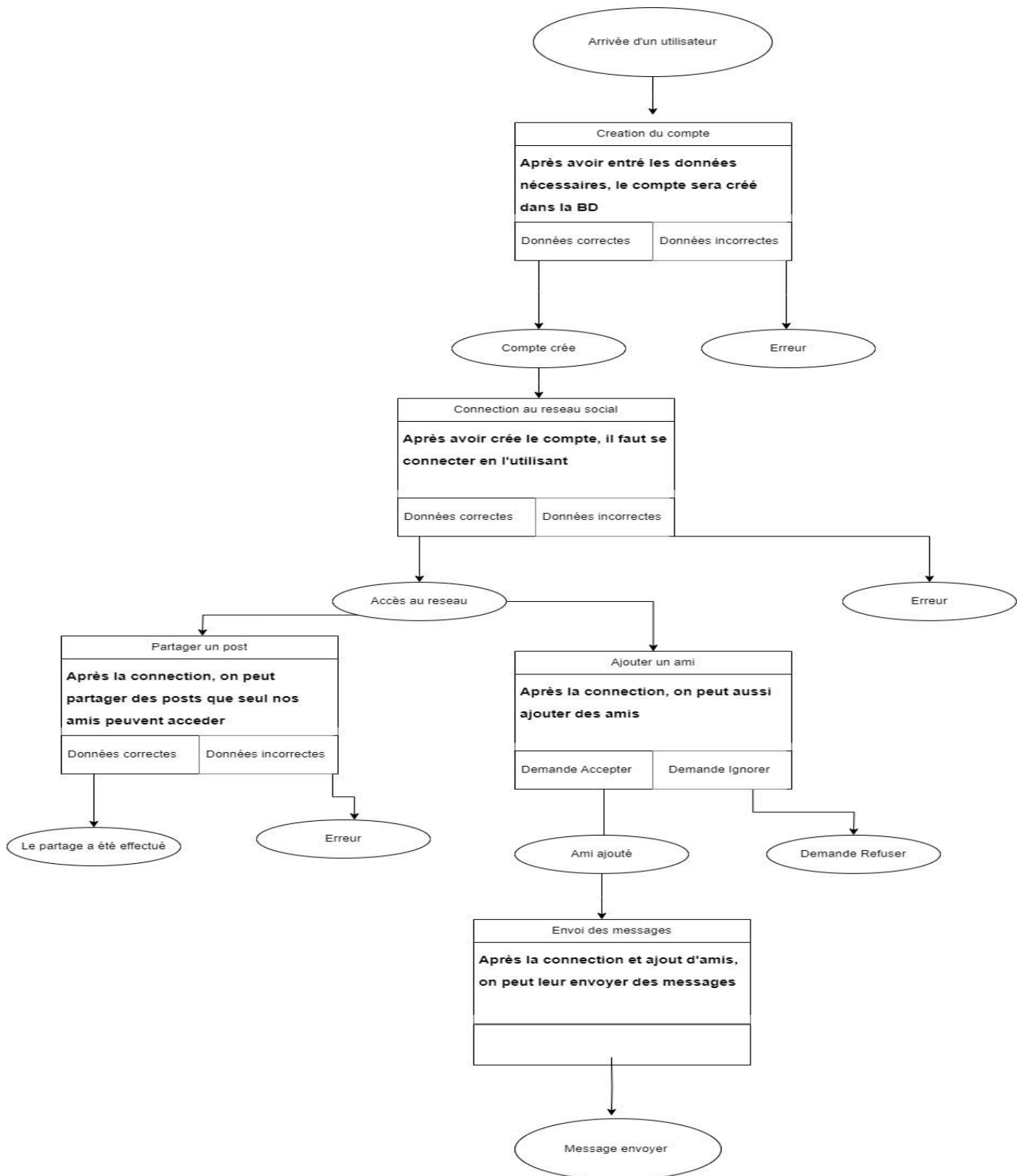


Figure 5 : MCT

5. Conclusion :

En concluant cette étape d'analyse et de conception **Merise**, on peut dire que nous avons bien saisi les besoins et les objectifs essentiels pour créer l'application de réseau social. Après une analyse approfondie, j'ai identifié les fonctionnalités clés telles que la gestion des profils, la messagerie, le partage de contenu, etc. On a également établi des règles et des contraintes pour garantir la sécurité et la convivialité de l'interface utilisateur.

J'ai créé des modèles conceptuels des données (MCD) et logiques (MLD) qui serviront de base solide pour la base de données. Ces modèles prennent en compte les relations complexes entre les entités, tout en assurant une cohérence optimale.

En résumé, cette phase d'analyse et de conception Merise représente une étape cruciale dans le développement du projet de réseau social. Elle nous permettra de passer à la phase de réalisation avec une vision claire et structurée de notre application, tout en veillant à respecter les principaux objectifs que nous avons définis.

Chapitre III : Réalisation de l'application

1. Introduction :

Dans ce chapitre, on va plonger au cœur de la réalisation de l'application. Cela implique de détailler les technologies spécifiques que j'ai sélectionnée pour leur adéquation avec nos objectifs, ainsi que les modèles de conception que nous avons judicieusement appliqués. L'ensemble de ces choix stratégiques constitue les fondations sur lesquelles repose le développement de notre application. Au fil de ce chapitre, nous allons vous dévoiler comment chaque décision a contribué à l'aboutissement de ce projet passionnant.

2. Langages et Frameworks utilisées :

2.1. Front-End :

- **HTML :**

HTML est le squelette d'une page Web, et il permet de définir la structure et le contenu de cette page, ce qui permet aux navigateurs Web de l'afficher correctement pour les utilisateurs.



Figure 6 : Logo HTML

- **CSS :**

CSS est utilisé pour contrôler l'apparence de ces éléments HTML. Grâce à CSS, les développeurs peuvent spécifier des styles tels que la couleur, la taille, la police, la mise en page, les marges et les bordures pour chaque élément d'une page Web.



Figure 7 : Logo CSS

- **JavaScript :**

JavaScript est un langage de programmation utilisé pour créer des sites Web interactifs. Il ajoute de la dynamique aux pages Web en permettant d'effectuer des actions telles que la validation de formulaires, la mise à jour du contenu en temps réel, l'animation d'éléments, et bien plus encore.



Figure 8 : Logo JavaScript

- **JQuery :**

JQuery est une bibliothèque JavaScript open-source populaire qui simplifie la manipulation du DOM (Document Object Model) et l'interaction avec des éléments HTML sur une page web. Elle offre une syntaxe simple et concise pour accomplir des tâches courantes telles que la manipulation d'éléments HTML, l'ajout ou la suppression de contenu, la gestion des événements, les requêtes AJAX pour communiquer avec un serveur, et bien plus encore.

2.2. Back-End :

- **PHP :**

PHP est un langage de programmation côté serveur utilisé principalement pour le développement web. Il est conçu pour générer du contenu dynamique sur les pages web en interagissant avec des bases de données, en manipulant des fichiers, en gérant des sessions utilisateur, et en effectuant diverses autres tâches côté serveur.



Figure 9 : Logo PHP

- **MySQL :**

MySQL est un système de gestion de base de données relationnelle (SGBDR) open source très populaire. Il est largement utilisé pour stocker, gérer et récupérer des données dans de nombreuses applications web et logicielles.

2.3. Version control :

La gestion de version est un système qui aide les développeurs de logiciels à gérer et à suivre les modifications apportées à leur code et à leurs fichiers. Elle conserve un historique de toutes les modifications effectuées, de qui les a effectuées et de quand elles ont été effectuées. Cela permet aux équipes de travailler efficacement ensemble, réduit le risque d'erreurs et permet de récupérer facilement les versions précédentes en cas de problème. Les systèmes de gestion de version populaires incluent **Git** et Subversion.

Dans ce projet, j'ai utilisé Git et GitHub. Leurs définitions sont les suivantes :

- **Git** : Git est un système de gestion de version décentralisé largement utilisé pour suivre les modifications apportées au code source et à d'autres fichiers. Il permet à plusieurs développeurs de travailler simultanément sur un projet, de fusionner leurs modifications en toute transparence et de conserver un historique complet des changements.

- **GitHub** : GitHub est une plateforme de développement collaborative basée sur Git. Elle fournit un espace où les développeurs peuvent héberger, partager et collaborer sur des projets Git. GitHub offre également des fonctionnalités telles que le suivi des problèmes, la gestion des tâches, la documentation et la possibilité de travailler en équipe sur des projets open source ou privés.



Figure 10 : Logo Git



Figure 11 : Logo GitHub

3. Outils Utilisées :

- **VS Code :**

Visual Studio Code (VS Code) est un éditeur de code source léger (IDE), gratuit et open source développé par Microsoft. Il est très populaire parmi les développeurs en raison de sa simplicité, de ses performances élevées et de sa grande flexibilité grâce à une variété d'extensions.

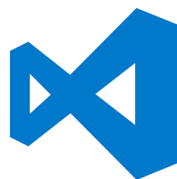


Figure 12 : Logo VS Code

- **XAMPP :**

XAMPP permet aux développeurs de créer et de tester des sites web en local avant de les déployer sur un serveur en ligne. Cela facilite le processus de développement et de débogage, car les développeurs peuvent travailler sur leurs projets sans avoir besoin d'une connexion internet ni de serveurs distants. Une fois que le site web est prêt, il peut être transféré vers un serveur en ligne pour être accessible au public.



Figure 13 : Logo XAMPP

4. Techniques utilisées :

- **AJAX :**

AJAX, qui signifie Asynchronous JavaScript and XML (JavaScript asynchrone et XML), est une approche de développement web qui permet aux pages web de communiquer avec un serveur en arrière-plan sans avoir à recharger la page entière. Cela signifie que les utilisateurs peuvent interagir avec une page web de manière fluide et dynamique, sans interruption.

- **Programmation Orientée Objet :**

OOP est un paradigme de programmation qui repose sur le concept d'objets. Il s'agit d'un style de programmation qui organise la logique d'un programme en utilisant des objets, qui sont des instances de classes.

- **MVC :**

MVC, ou Model-View-Controller, est un modèle de conception de logiciels largement utilisé dans le développement d'applications informatiques, notamment les applications web. Il divise une application en trois composants principaux, chacun selon un rôle spécifique :

5. Création de l'application :

Avant de plonger dans les détails, permettez-moi de vous présenter une vue d'ensemble des éléments du projet :

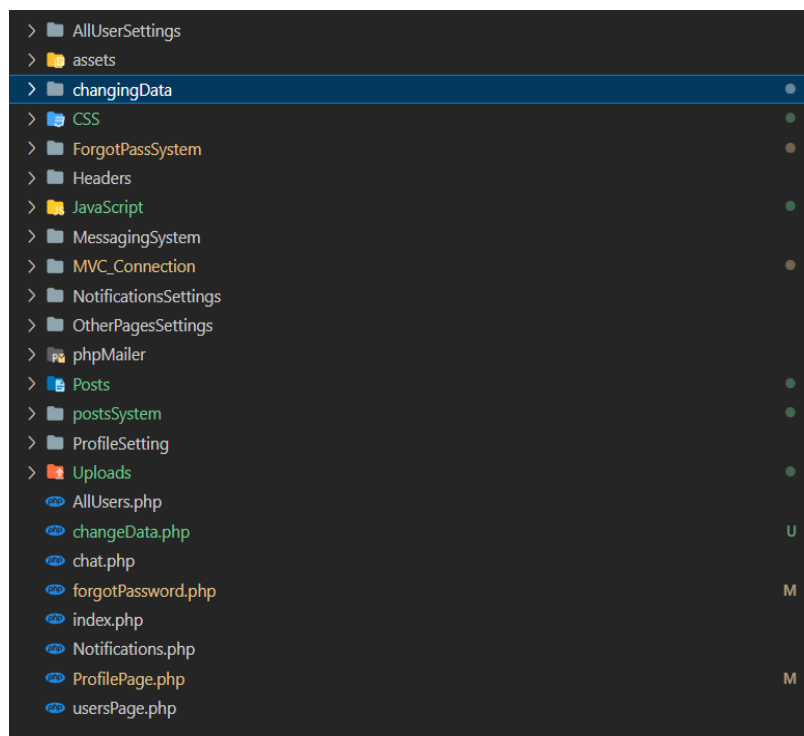


Figure 14 : arbre du projet

5.1. Page de Connection (index.php) :

- Cette page est la première page du projet, et son objectif est de permettre aux utilisateurs de créer un compte ou de se connecter s'ils en ont déjà un. Les données sont saisies dans un élément **<form>** qui a une action dirigée vers le fichier **Connection.php** situé dans le dossier **MVC_Connection**. Ce fichier est chargé de gérer le processus d'authentification et de création de compte des utilisateurs. En résumé, cette page d'accueil constitue le point de départ de l'interaction des utilisateurs avec l'application. Le site bascule entre les fonctions de création de compte et d'authentification en fonction du bouton sur lequel l'utilisateur clique.

Voici à quoi elle ressemble :

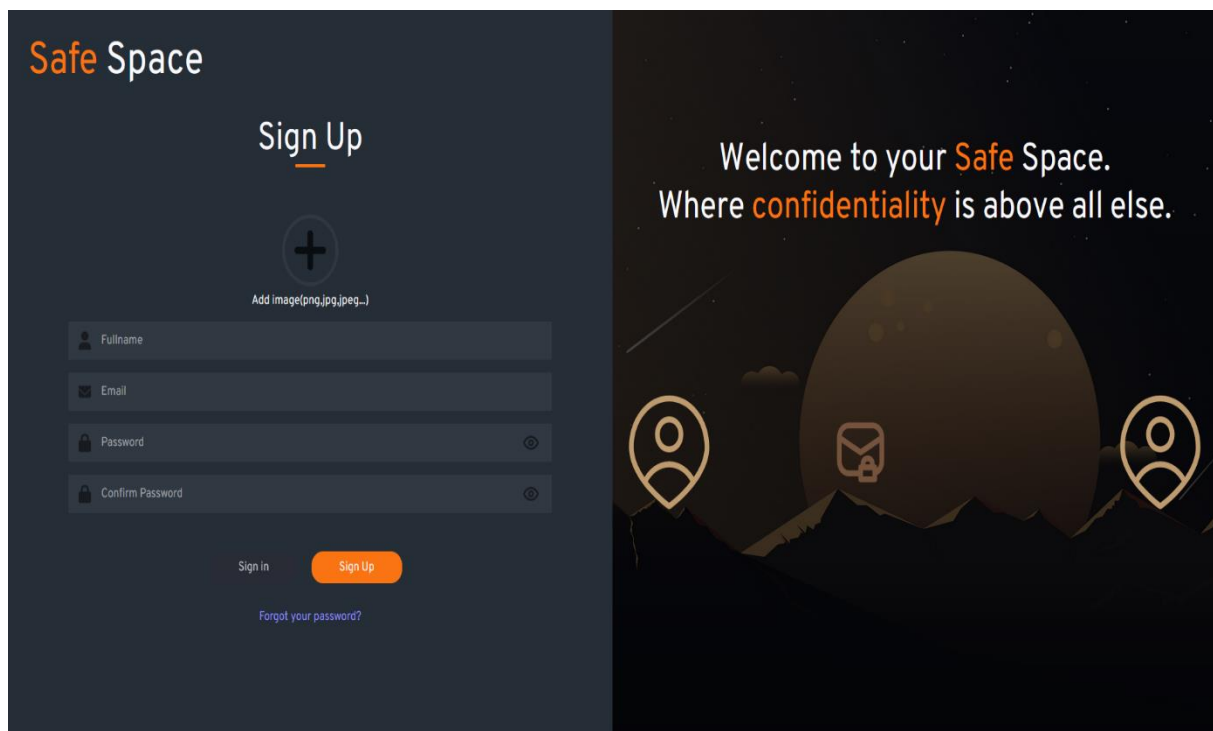


Figure 15 : Page de Connection au réseau social

Ci-dessous, on peut voir un extrait du code HTML :

```
<form id="form" action="MVC_Connection/Connection.php" method="post" enctype="multipart/form-data">
  <div class="inputs">
    <div class="image-field">
      <label for="img" id="image-upload"></label>
      <p id="img-para">Add image(png, jpg, jpeg...) </p> <?php imageErrors();?>
      <input id="img" class="input" type="file" name="Image" >
    </div>
    <div class="input-field signup-fields">
      <img class="icons" src = 'assets/user-icon.svg' alt='User Icon'>
      <input id='name-input' class='input' type='text' name= 'Fullname' maxlength='50' placeholder='Fullname' >
    </div>
    <div class="input-field">
      <img class="icons" src = 'assets/email.svg' alt='User Icon'>
      <input id='email-input' class='input' type='email' name= 'email' placeholder='Email' >
    </div>
    <div class="input-field">
      <img class="icons" src = "assets/lock-solid.svg" alt="Lock icon">
      <input class="input password1" type="password" name= "password1" maxlength="16" minlength="5" placeholder="Password" >
      <img class="icons eyes1" src = "assets/eye.svg" alt="eyes">
    </div>
    <div class="input-field signup-fields">
      <img class="icons" src = "assets/lock-solid.svg" alt="Lock icon">
      <input class="input password2" type="password" name= "password2" maxlength="16" minlength="5" placeholder="Confirm Password" >
      <img class="icons eyes2" src = "assets/eye.svg" alt="eyes">
    </div>
  </div>
  <div class="buttons-field">
    <input class="btn active n1" type="submit" name="SignIn" value="Sign in">
    <input class="btn inactive n2" type="submit" name="SignUp" value="Sign Up">
  </div>
</form>
```

Figure 16 : Extrait du formulaire HTML

5.1.1. Cas de création du compte :

- Dans le cas de la création de compte, l'utilisateur doit saisir toutes les informations requises, à l'exception de l'image, qui est facultative. En cas de données non valides, un message d'erreur s'affiche.
- Le code PHP chargé de la gestion des erreurs est organisé en classes, conformément à l'architecture MVC (Modèle-Vue-Contrôleur). La vue, nommée **signupView.php**, gère l'affichage des éléments visuels, le contrôleur, nommé **signupContr.php**, prend en charge les requêtes et interactions de l'utilisateur, tandis que le modèle, **signupModel.php**, gère les opérations liées à la base de données. Ces fichiers se trouvent dans le dossier MVC_Connection.

Les fichiers en question :

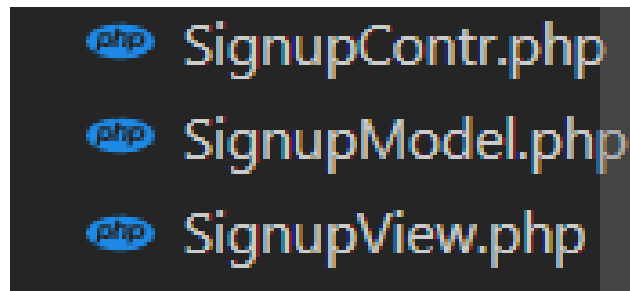


Figure 17 : Fichiers SignUp selon MVC

- Le fichier **Connection.php** est responsable de la coordination de toutes les informations transmises via index.php pour effectuer les opérations de gestion des erreurs, en utilisant ce qui précède.

Un extrait du code est comme suivant :

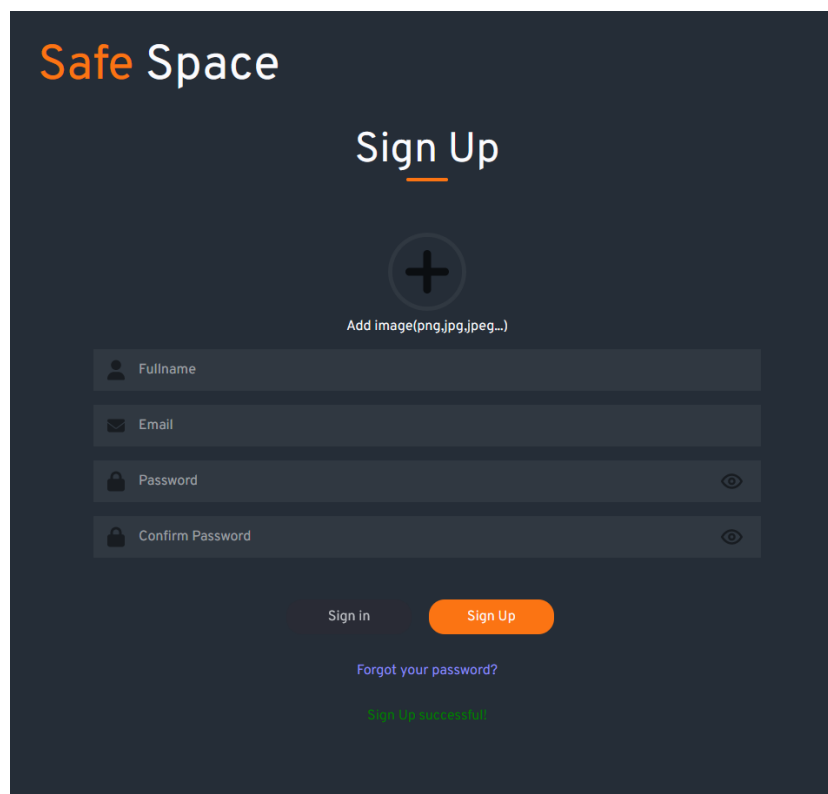
```
if($_SERVER["REQUEST_METHOD"] == "POST"){  
    // ! Signing up Configurations  
    if(isset($_POST["SignUp"])){  
        try{  
            require_once("SignupModel.php");  
            require_once("SignupContr.php");  
  
            /**Error Handling  
  
            $Controller = new CheckInput();  
            $errors = [];  
            $image_errors = [];  
            $imageFound = false;  
            if($Controller->isInputMissing($_POST["Fullname"], $_POST["email"], $_POST["password1"], $_POST["password2"])){  
                $errors["missingInput"] = "A Field is missing!";  
            }  
            if(!$Controller->validateEmail($_POST["email"])){  
                $errors["validateEmail"] = "Invalid Email!";  
            }  
            if ($Controller->isEmailTaken($_POST["email"])){  
                $errors["emailRegistered"] = "Email registered!";  
            }  
            if (!$Controller->typedPasswords($_POST["password1"], $_POST["password2"])) {  
                $errors["ConfirmPassword"] = "Passwords don't match!";  
            }  
            if (!$Controller->validatePassword($_POST["password1"])){  
                $errors["validatePass"] = "Password should contain at least one capital and special char!";  
            }  
        }  
    }  
}
```

Figure 18 : Extrait du code Connection.php

- En ce qui concerne les images, elles sont stockées dans un dossier nommé "**Uploads**" à la racine du répertoire, avec de nouveaux noms uniques qui sont également enregistrés dans la base de données pour les identifier plus tard.
- Après l'insertion des données sans erreurs, le compte sera créé en toute **sécurité** :

		id	Email	FullName	Passwords	LoginState	imageName	CreatedAt
<input type="checkbox"/>	Edit Copy Delete	1	test@gmail.com	Test	\$2y\$12\$ivBQq/sck2pfapwYr4JJ.yRIL8zdSyHTd7TP7.Hcil...	Offline	Default.png	2023-09-10 00:22:

Figure 19 : Création du compte dans la DB



The image shows a 'Sign Up' form for an application called 'Safe Space'. The form is set against a dark blue background. At the top, the 'Safe Space' logo is in orange and white. Below it, the text 'Sign Up' is centered in white. Underneath is a circular profile picture placeholder with a white plus sign and the text 'Add image(png,jpg,jpeg...)'. The form contains four input fields: 'FullName' (with a person icon), 'Email' (with an envelope icon), 'Password' (with a lock icon and a toggle eye icon), and 'Confirm Password' (with a lock icon and a toggle eye icon). At the bottom, there are two buttons: a dark blue 'Sign in' button and an orange 'Sign Up' button. Below the buttons, there are two links: 'Forgot your password?' in blue and 'Sign Up successful' in green.

Figure 20 : Message de succès

5.1.2. Cas d'authentification :

- Pour l'authentification et après la création du compte, vous pouvez vous connecter au réseau social en utilisant l'adresse e-mail et le mot de passe que vous avez précédemment saisis. En cas d'erreur, comme dans le cas d'un compte introuvable ou un mot de passe incorrect, un message d'erreur s'affichera.
- Le code destiné à la gestion des erreurs est structuré de la même manière que celui utilisé pour la création de compte, suivant le modèle MVC (Modèle-Vue-Contrôleur). Le contrôleur, **signinContr**, gère les requêtes, le modèle, **signinModel**, gère les requêtes à la base de données, et la vue, **signinView**, gère les éléments visuels.

Les fichiers d'authentification :

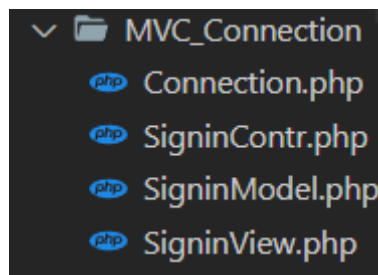


Figure 21 : Fichiers SignIn selon MVC

- Dans le cas où vous avez oublié **votre mot de passe**, vous avez la possibilité de le réinitialiser en cliquant sur le lien "Forgot your password ?". En cliquant sur ce lien, vous serez redirigé vers une page spéciale où vous pourrez suivre le processus de réinitialisation du mot de passe :

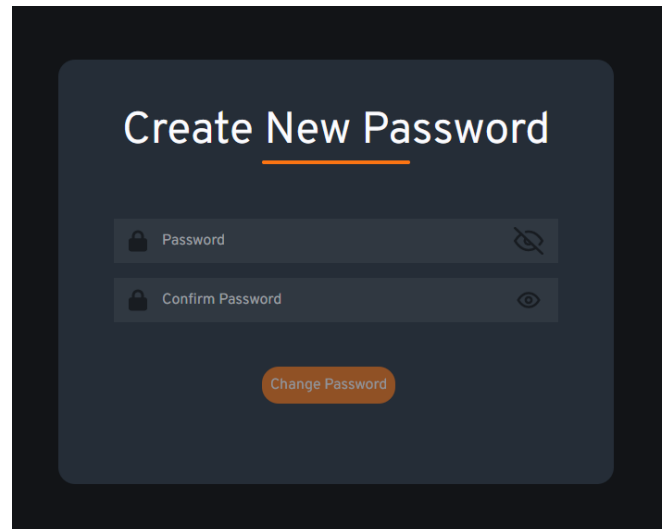


Figure 22 : Page de réinitialisation du MDP

- Après avoir saisi votre adresse e-mail et si celle-ci est associée à un compte, un e-mail contenant un lien vous sera envoyé en utilisant **la librairie PHP "PHPMailer"**. Ce lien contiendra un **token** ainsi qu'un **sélecteur** permettant d'identifier votre compte dans la base de données. Une fois que nous avons vérifié que votre compte et le token associé sont corrects, vous serez redirigé vers la page de modification de votre mot de passe.

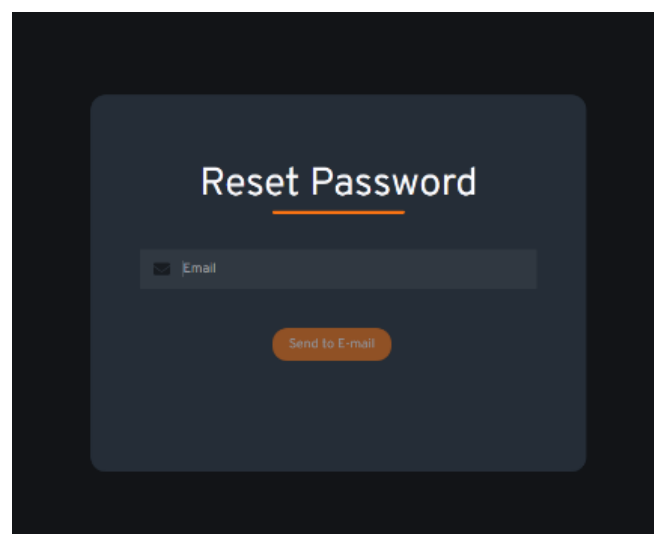


Figure 23 : Page de création du nouveau MDP

- Dans le fichier **Connection.php**, ces éléments (fichiers SignIn) sont appelés et appliqués aux données reçues de la partie front-end :

```

1  #!/ Signing in Config
2  else if (isset($_POST["SignIn"])){
3      $email = $_POST["email"];
4      $password = $_POST["password1"];
5      try {
6          require_once("SigninModel.php");
7          require_once("SigninContr.php");
8
9          /*Error Handling
10         $SignInController = new CheckSignInInput();
11         $errors = [];
12
13         if($SignInController->isInputMissing($email,$password)){
14             $errors["missingInput"] = "A Field is missing!";
15         } else {
16             if (!$SignInController->isEmailTaken($email)){
17                 $errors["emailRegistered"] = "Email not registered!";
18             } else {
19                 if (!$SignInController->passwordCheck($password, $email)){
20                     $errors["passwordCheck"] = "Incorrect Password!";
21                 }
22             }
23         }
24
25         if ($errors){
26             $_SESSION["SignInErrors"] = $errors;
27
28             header("Location: ../index.php");
29             die();
30         }
31     }
32 
```

Figure 24 : Extrait du code SignIn

- Si aucune erreur n'est détectée, l'authentification réussit et l'utilisateur est redirigé vers son profil :

```

1  /? Setting the necessary Variables
2  $_SESSION["userId"] = $userId;
3  $_SESSION["email"] = $result["Email"];
4  setcookie("user_id", $_SESSION["userId"], 0, "/");
5  $_SESSION["Fullname"] = $result["FullName"];
6  $_SESSION["imageName"] = $result["imageName"];
7  $_SESSION["LoggedIn"] = true;
8
9  if(isset($_SESSION["LoggedIn"]) && $_SESSION["LoggedIn"] ){
10     $SignInController->Online($_SESSION["email"]);
11 }
12
13 header("Location: ../ProfilePage.php?login=success&id=" . $_SESSION["userId"]);
14
15 die();
16 }
17 catch(PDOException $e){
18     return "SignIn Failed! Please try again!" . $e->getMessage();
19     die();
20 }
21 
```

Figure 25 : Code PHP pour l'authentification

5.2. La page de profil :

- La page de profil est accessible après une authentification réussie. Cela signifie que si un utilisateur essaie d'accéder à la page de profil sans être authentifié, il sera automatiquement redirigé vers la page d'index, où il peut soit se connecter, soit créer un nouveau compte

- La page de profil, tout comme les autres pages, est constituée d'une **navbar** contenant des liens vers les différents éléments de notre réseau social, comme illustré ci-dessous :

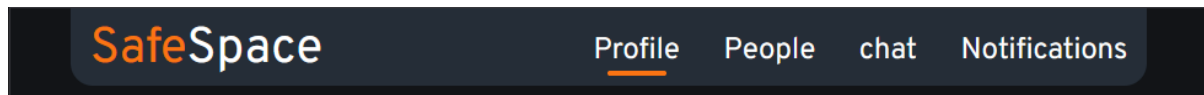


Figure 26 : Navbar version PC



Figure 27 : Navbar Tablet/Mobile (fermée)

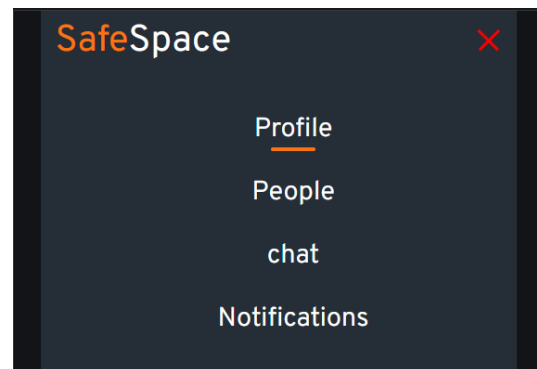


Figure 28 : Navbar Tablet/Mobile (ouvert)

- Elle permet également de **visualiser les données personnelles** obtenus en utilisant des déclarations SQL de type SELECT, et de **changer les données personnelles** en utilisant les commandes UPDATE. Pour la déconnexion, elle s'effectue en abandonnant la session de l'utilisateur et en mettant son statut à "Offline" dans la base de données :



Figure 29 : Profile version PC

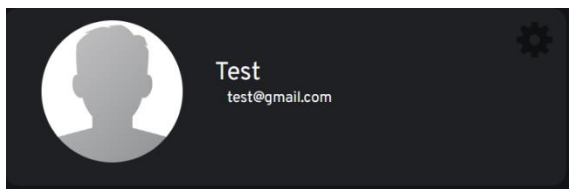


Figure 30 : Profile Tablet/Mobile (fermée)

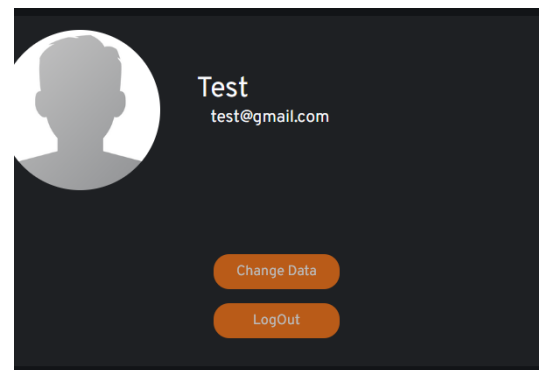


Figure 31 : Profile Tablet/Mobile (ouvert)

- En cliquant sur le bouton "Change Data", vous serez automatiquement dirigé vers une page dédiée à **la mise à jour de vos informations personnelles**. Cette page est conçue de manière à vous offrir un contrôle total sur les données associées à votre compte. Chaque élément présent sur cette page est soigneusement intégré pour faciliter la gestion de vos informations. L'ensemble du processus est pensé pour garantir une expérience utilisateur fluide et sécurisée :

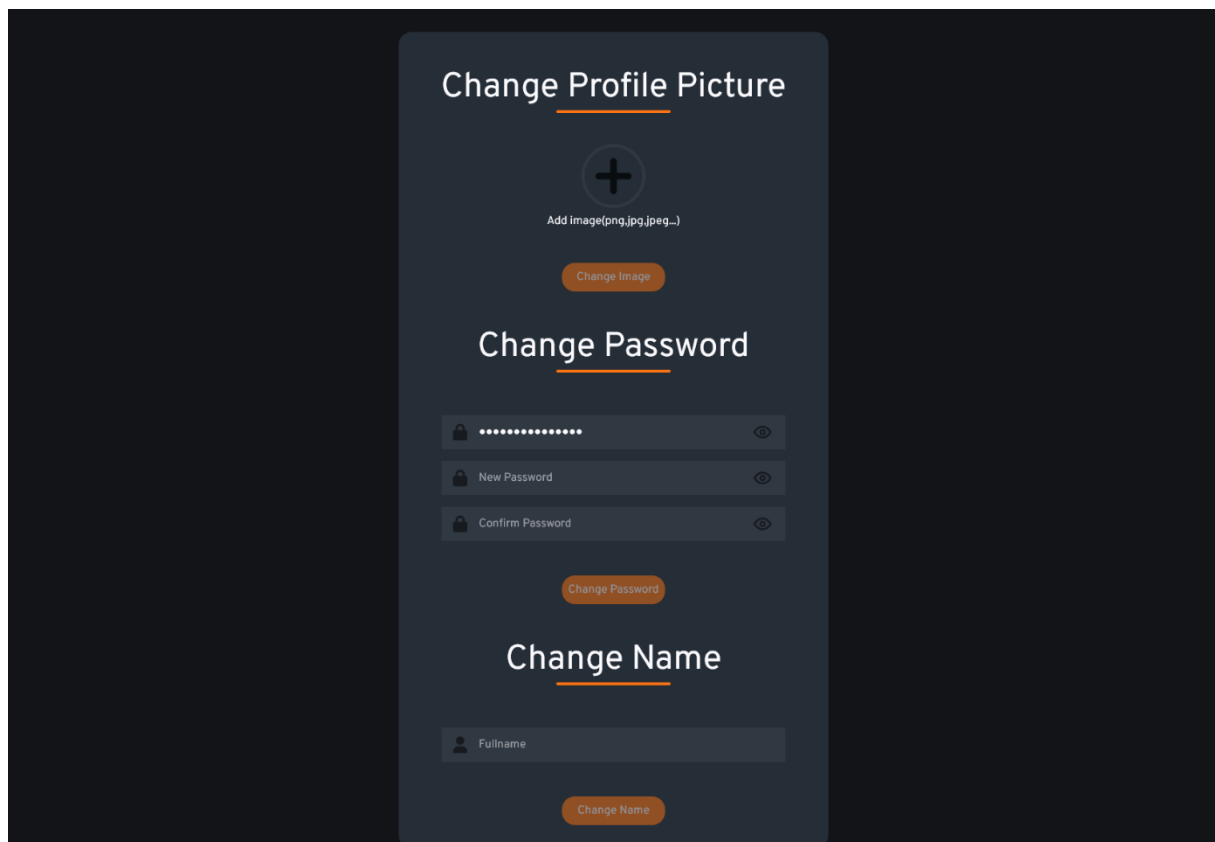


Figure 32 : ChangeData Page

- En cliquant sur le bouton « LogOut », vous serez automatiquement **déconnecté** de votre compte, et la session associée sera détruite pour assurer la sécurité du réseau.

- Cette page permet également de **partager des publications** en cliquant sur l'icône en forme de plus ci-dessous et en entrant les données nécessaires :



Figure 33 : Button pour ajouter un post

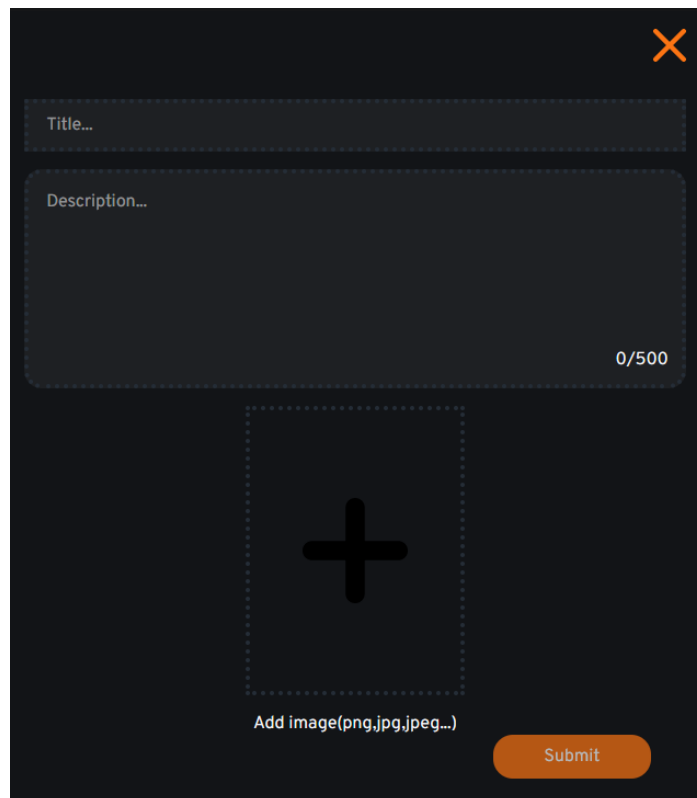


Figure 34 : Formulaire pour le partage d'un post

- Une fois le post partagé, il pourra être visualisé sur votre profil ou par vos amis :



Figure 35 : Aperçu du post

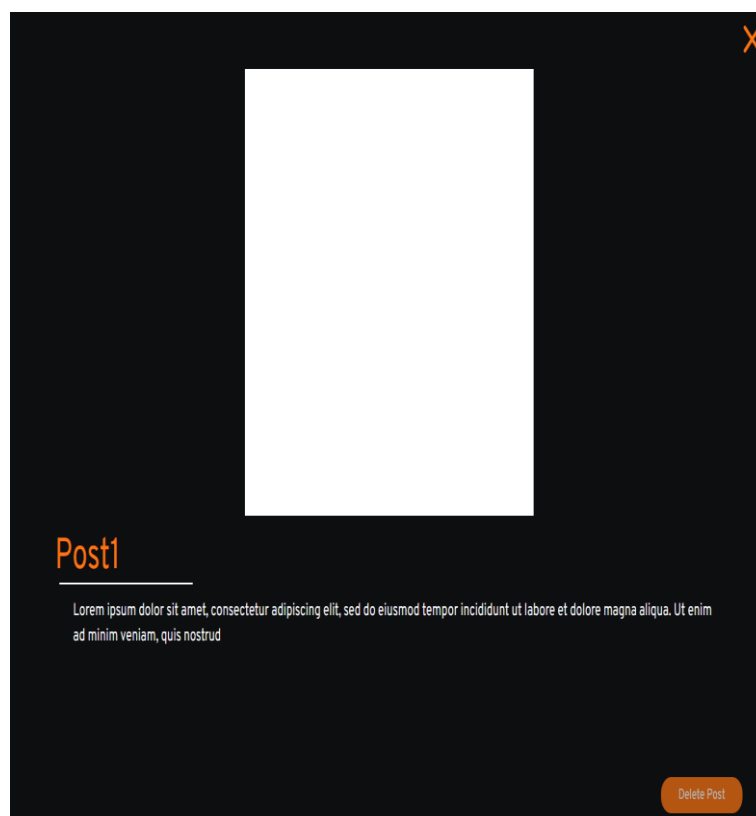


Figure 36 : Détails du post

- Le bouton « Delete Post » comme son nom l'indique, permet de supprimer le post complètement.

5.3. La page des utilisateurs (People) :

- En plus de la navbar mentionnée précédemment, cette page permet essentiellement l'interaction avec les autres utilisateurs. Elle affiche tous les autres utilisateurs, qu'ils soient en ligne ou non. De plus, elle permet d'accéder à leur profil pour voir leurs posts en cas d'amitié, ou de les ajouter comme amis :

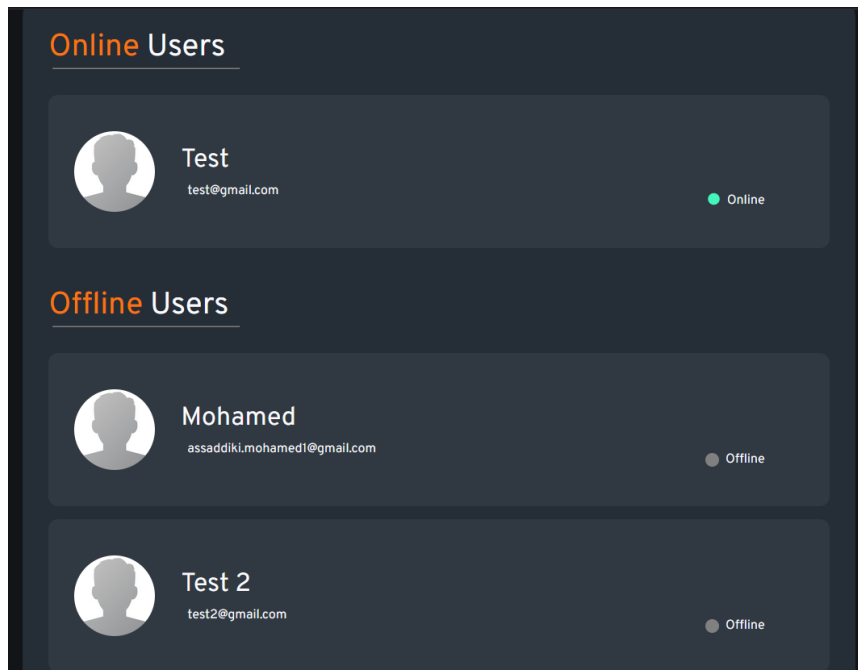


Figure 37 : Les utilisateurs du réseau

- En cliquant sur un des utilisateurs on accède à son profil :

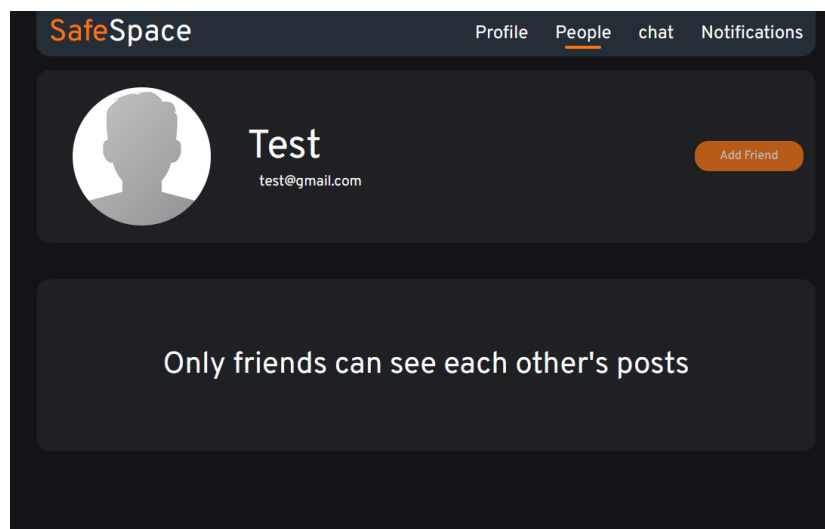


Figure 38 : Le profil d'utilisateur voulu

- Pour ajouter un ami, on clique sur le bouton « Add Friend » :

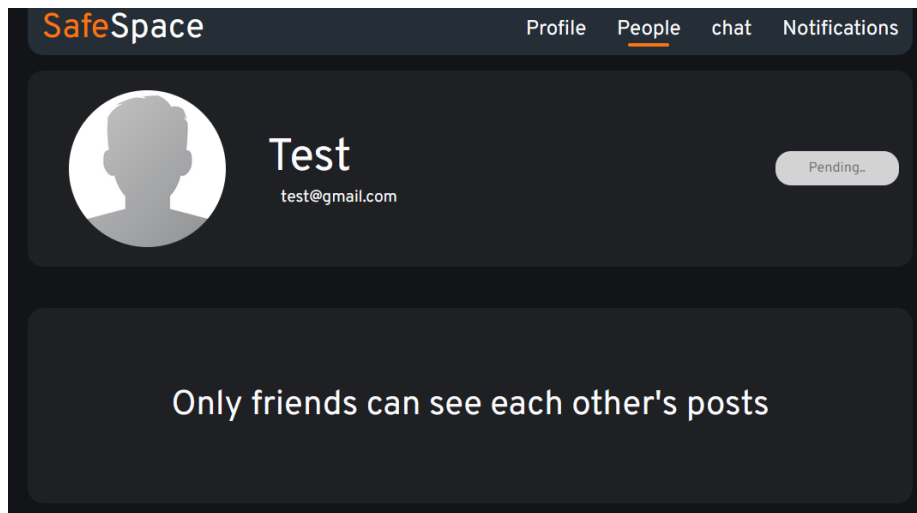


Figure 39 : Envoi d'une demande d'ami

- Une fois la demande envoyée, le destinataire reçoit une notification à ce sujet et peut désormais choisir de l'accepter ou de l'ignorer :



Figure 40 : Alerte notification

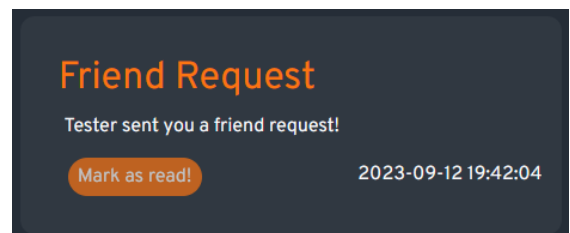


Figure 41 : Page de notification

- La notification peut être marquée comme lue sur la page de notifications, à laquelle vous pouvez accéder via la navbar ou en cliquant sur l'icône de la cloche

- Une fois que vous avez cliqué sur la notification, vous serez redirigé vers la page de l'émetteur de la demande d'amis, où vous pouvez choisir d'accepter cette demande ou de la rejeter en cliquant sur "Ignorer".

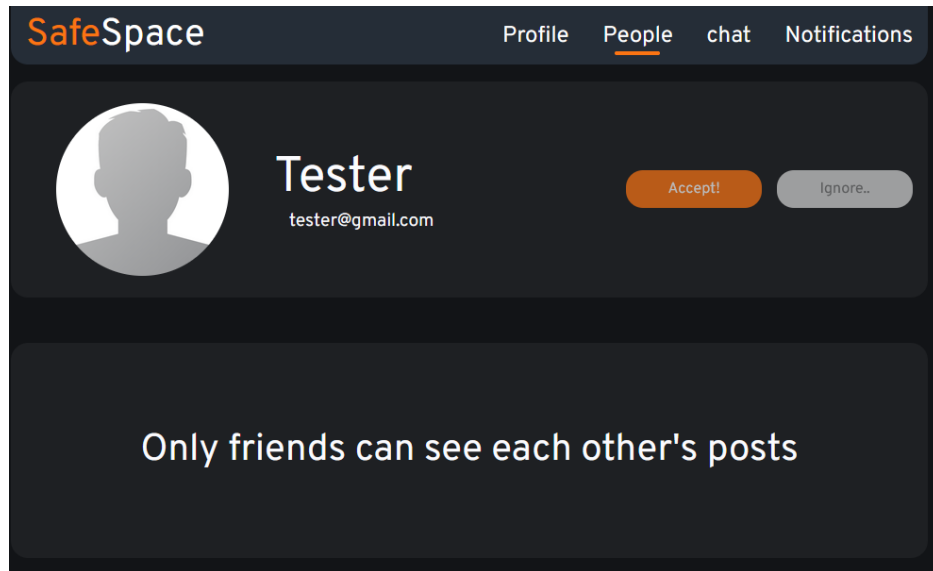


Figure 42 : Réception de la demande

- En cas de clic sur "Accepter", une amitié sera établie et aura un statut de 1 dans la base de données. Les posts de votre nouvel ami seront désormais accessibles.

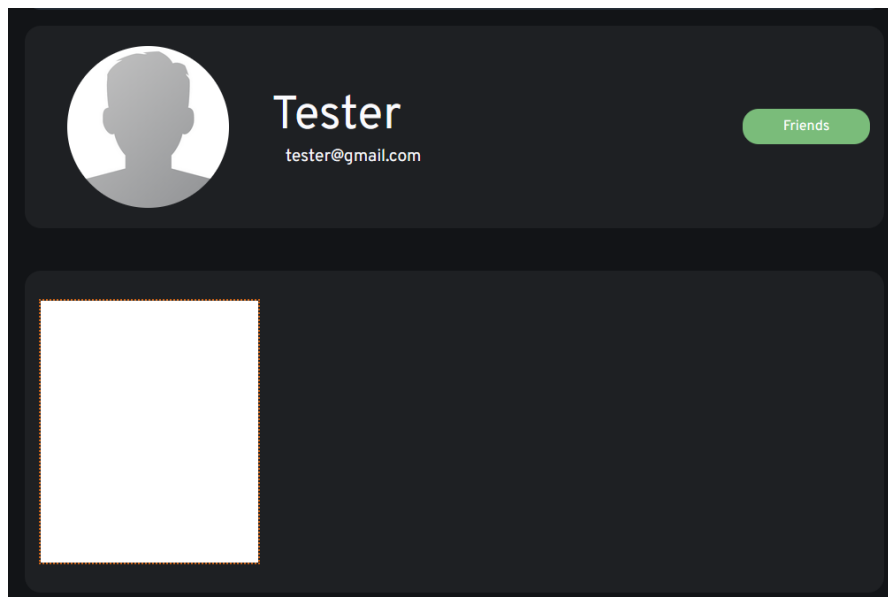


Figure 43 : Amitié créée

5.4. Page du Chat :

- Après l'amitié créée, vous aurez la possibilité de messenger vos amis en temps réel ou de leur laisser des messages pour plus tard.

Si on vous a envoyé des messages lorsque vous n'étiez pas en ligne, un cercle rouge apparaîtra à côté de l'ami qui vous a envoyé les messages non lus.

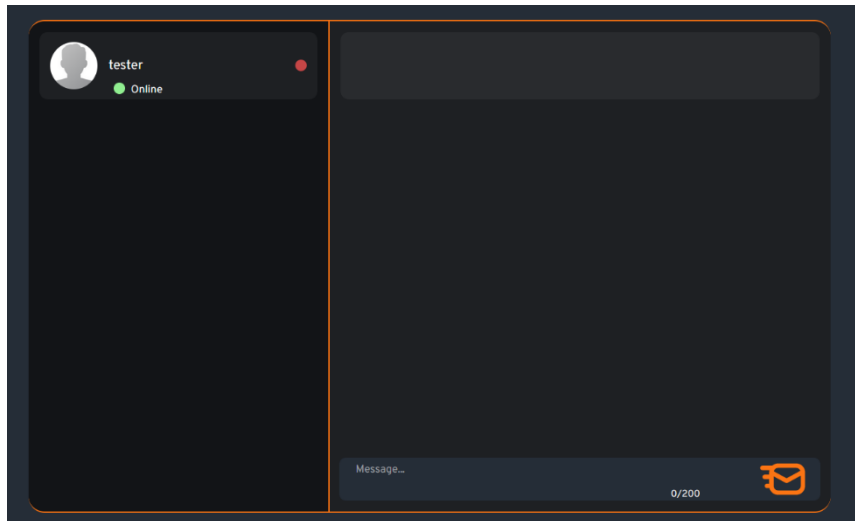


Figure 44 : Messages non lues

- Une fois que vous avez choisi l'ami avec lequel vous voulez discuter, vous pouvez vous envoyer des messages rapidement et en toute sécurité :

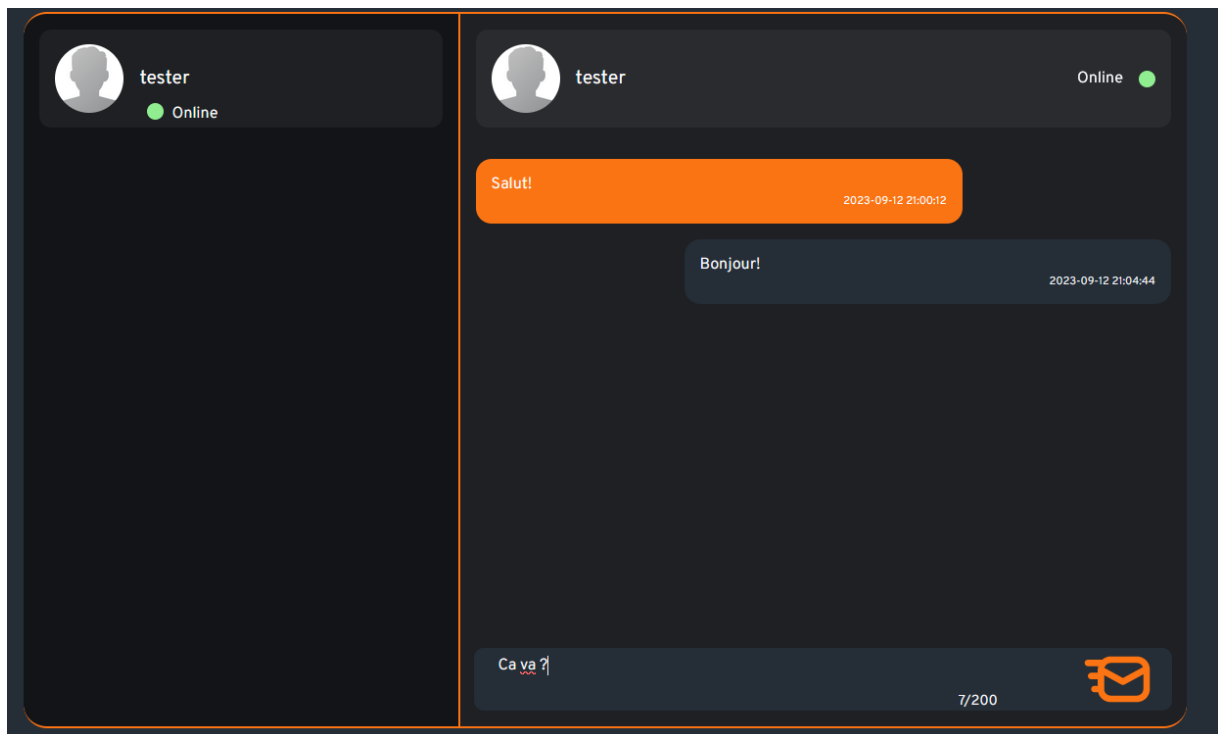


Figure 45 : Envoi des messages

6. Conclusion :

En conclusion, ce projet a abouti à la création d'une application de réseau social fonctionnelle et interactive. À travers la mise en œuvre de diverses technologies telles que PHP, HTML, CSS, JavaScript, et l'utilisation de concepts tels que l'architecture MVC, la gestion de bases de données, et l'authentification utilisateur, j'ai réussi à développer une plateforme permettant aux utilisateurs de créer des comptes, de se connecter, de partager des photos et des messages, d'établir des amitiés et de communiquer en temps réel.

Ce projet a permis d'acquérir des compétences précieuses en matière de développement web et d'applications, de renforcer la compréhension de l'architecture MVC, et de mettre en pratique les principes de la sécurité informatique.

En fin de compte, ce projet a été une expérience enrichissante qui m'a permis d'apprendre et de progresser significativement dans le domaine du développement web.

Conclusion Général :

En conclusion, ce stage a été une expérience extrêmement enrichissante. J'ai eu l'opportunité de mettre en pratique les connaissances acquises au cours de ma formation et d'apprendre de nouvelles compétences essentielles dans le domaine du développement web. J'ai découvert l'importance de l'architecture MVC, la gestion de bases de données, la sécurité informatique, et j'ai maîtrisé l'utilisation de nombreuses technologies telles que PHP, HTML, CSS, JavaScript, Git, et bien d'autres.

Ce stage m'a également permis de développer des compétences en gestion de projet, de travailler en équipe, et de comprendre l'importance de la communication et de la collaboration dans un environnement professionnel. J'ai été confronté à des défis stimulants, mais chaque obstacle surmonté m'a apporté une nouvelle compréhension et une expérience précieuse.

En somme, ce stage a été une étape cruciale dans mon parcours d'apprentissage et il a renforcé ma passion pour l'informatique. Je suis reconnaissant envers toute l'équipe de l'entreprise qui m'a soutenu et encadré tout au long de cette expérience. Je quitte ce stage avec une grande confiance en mes compétences et une motivation accrue pour continuer à élargir mes connaissances et ma carrière dans le vaste univers de l'informatique.

Bibliographie :

Afin de mettre en place ce rapport, j'ai utilisé plusieurs sources, essentiellement les suivantes :

<https://www.php.net/> : le site officiel de PHP

<https://www.w3schools.com/> : un site conçu pour aider les développeurs

<https://jquery.com/> : le site officiel de JQuery

<https://stackoverflow.co/> : forum pour les développeurs

<https://git-scm.com/> : site officiel du GIT

<https://developer.mozilla.org/en-US/> : Documentations pour les langages du web