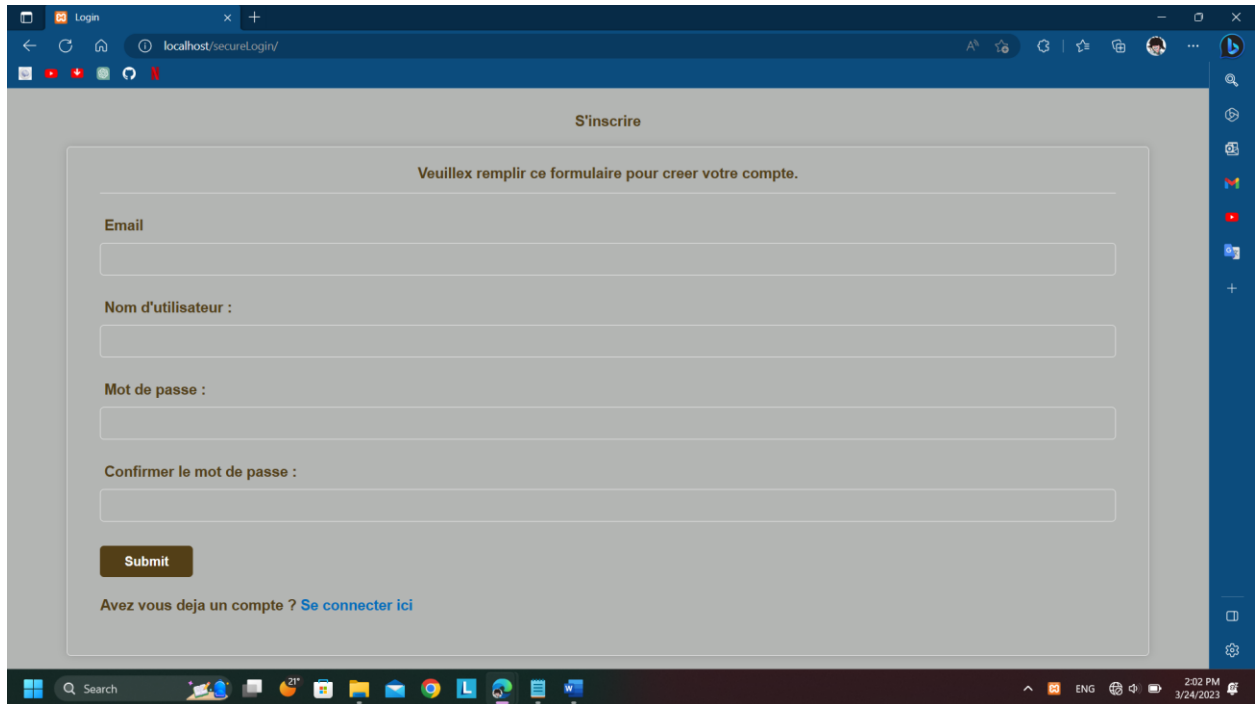
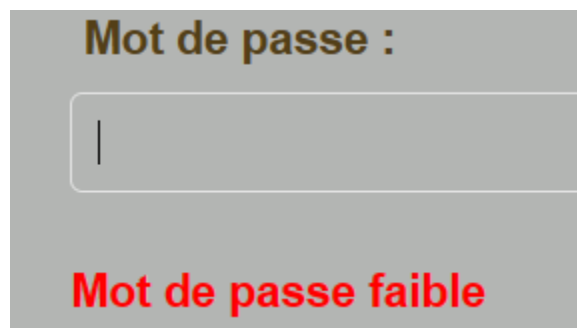


# Rapport de réalisation d'un projet de connexion sécurisée :

## 1. Au niveau de la page d'inscription :



- On peut créer un nouvel utilisateur avec un mot de passe haché qui doit contenir au moins 8 caractères (Une lettre minuscule et une majuscule et un chiffre au minimum), et on a une fonctionnalité dans notre application qui sert à guider l'utilisateur au moment d'écritures de mot de passe :



Mot de passe :

.....

Mot de passe moyen

Mot de passe :

.....

Mot de passe fort

- L'email doit être valide.
- On ne peut pas créer un nouvel utilisateur avec un email déjà crée dans la base de données.
- Le mot de passe doit être identique eu champs de la confirmation de mot de passe.
- Des fonctions qui enlèvent les espaces et les caractères spéciaux saisies au niveau des champs.

## 1. La page de connexion :

**S'identifier**

Veuillez remplir ce formulaire pour accéder a votre compte

Nom d'utilisateur

Mot de passe

Submit

Vous n'avez pas de compte ? [Créer ici](#)

[Mot de passe oublié ?](#)

## 2. La page de réinitialisation du mot de passe :

## Réinitialisation du mot de passe

Veuillez entrer votre email afin que nous puissions vous envoyer un nouveau mot de passe

Email

Submit

[Retour à la page de connexion ?](#)

- Ici on doit saisir l'email dont on a créé le compte, après un nouveau code est envoyé a votre boîte mail pour que pouvez se connecter avec.



- Une fois on est connecté on peut se déconnecter
- On peut même supprimer notre compte

## Bienvenue

Vous etes sur votre compte !

[Se déconnecter](#)  
[Supprimer compte](#)

**Conclusion :**

Notre application permet de gérer des attaques qu'on a étudié au niveau du module de sécurité informatique tel que :

- **Les injections SQL** : Toutes les requêtes SQL sont préparées pour empêcher les injections SQL, les données fournies par l'utilisateur sont filtrées pour empêcher les injections SQL.
- **Protection contre les attaques par force brute** : Les mots de passe sont stockés dans la base de données sous forme d'hachage à l'aide de la fonction de hachage PHP password\_hash, les tentatives de connexion infructueuses sont limitées à un nombre défini pour empêcher les attaques par force brute.
- **Protection contre les attaques XSS** : Les données fournies par l'utilisateur sont filtrées pour empêcher les attaques XSS, les données sensibles stockées dans la base de données sont échappées pour empêcher les attaques XSS.
- **Hachages des mots de passes.**

	user_id	username	password	email
<input type="checkbox"/> Edit Copy Delete	22	omar	\$2y\$10\$zGV1pscd/0LTa4mQVXaT.FYAN9kYcvTEZU7O8Hi0p8...	ellhotriomar@gmail.com