P\_App



Senna Cruz – CID2A

B11 - Vennes

24 périodes

Gaël Sonney

Table des matières

[1 Conceptualisation 3](#_Toc192886472)

[2 Explications du code 3](#_Toc192886473)

[2.1 Hash et Sel 3](#_Toc192886474)

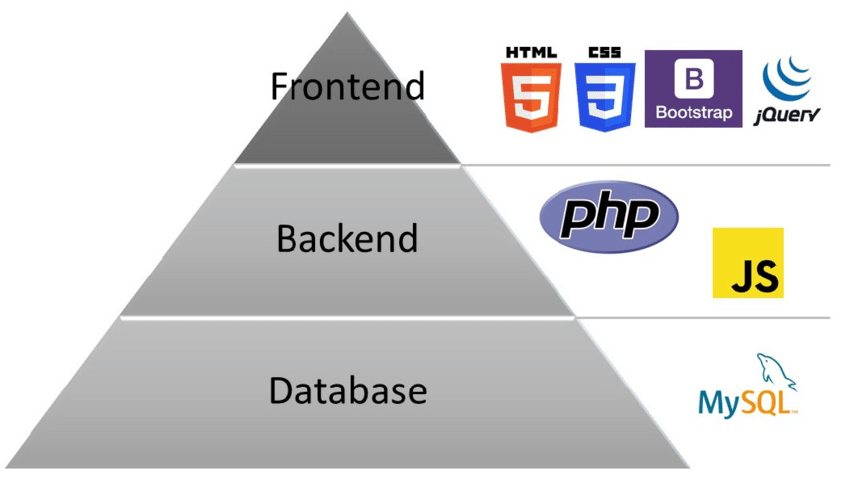
[2.2 Inscription 4](#_Toc192886475)

[2.3 Connexion 4](#_Toc192886476)

[3 Conclusion 6](#_Toc192886477)

[4 Recours à l’ia 6](#_Toc192886478)

# Conceptualisation



# Explications du code

## Hash et Sel

A computer screen shot of a program

AI-generated content may be incorrect.

Trois fonctions ont été implémentées pour gérer la sécurité du système, « generateSalt() », qui génère un sel aléatoire de 16 octets, « hashPassword() », qui s’agit d’une fonction qui hashe les mots de passe en utilisant sha256, et finalement « verifyPassword() », qui utilise la méthode précédente pour comparer le hash de la tentative de mot de passe avec le hash du mot de passe stocké dans la base de données.

## Inscription

L’intégralité du processus de la création de l’utilisateur est implémentée dans « register », qui est la fonction d’inscription du contrôleur d’authentification :

A screen shot of a computer program

AI-generated content may be incorrect.

Tout d’abord, une extraction de données du formulaire est effectuée suivie d’une vérification du nom d’utilisateur afin de savoir si celui-ci est unique en effectuant des requêtes SQL. Ensuite, un sel est généré via « generateSalt() » suivi d’un hashage du mot de passe avec « hashPassword(password, salt) ». Finalement, les données, qui s’agissent du nom d’utilisateur, du mot de passe haché, de l’email et du sel sont récupérées sont stockées de manière sécurisées dans la base de données.

## Connexion

Le processus d'authentification est implémenté dans la fonction login du contrôleur d'authentification :  
A screen shot of a computer program

AI-generated content may be incorrect.

La première étape concernant l’authentification s’agit du nom d’utilisateur et du mot de passe. Puis, une requête est effectuée afin d’essayer de trouver l’utilisateur soumis dans la base de données. Le sel qui est attribué à l’utilisateur est donc récupéré afin de l’utiliser pour vérifier l’utilisateur. Le hash généré est ensuite comparé avec celui de la base de données en utilisant la fonction « verifyPassword() » :

A computer screen with white text

AI-generated content may be incorrect.

Si la connexion est réussite, l’utilisateur est ensuite redirectionné à la page « dashboard » de l’admin si celui-ci s’agit d’un admin, cependant si ce n’est pas le cas il est redirecionné ver la page « dashboard » de « user ».

# Conclusion

Ce projet m’a permis de développer une application sécurisée d’où la focalisation était essentiellement l’aspect de sécurité de l’application. Lors de ce projet, j’ai pu accroitre mes compétences en :

* Implémentation d’authentification dans un système en utilisant un mécanisme de hashage et de sel.
* Protection contre les injections SQL.
* Gestion et manipulation des rôles.
* Configuration HTTPS.

En ce qui concerne mon attitude face au projet, celle-ci était passablement questionnable au début car les instructions n’étaient pas intégralement claires de mon point de vue. Cependant plus j’ai eu de temps, plus j’ai pu comprendre le but du projet et mon attitude a donc changée pour le côté positif.

# Recours à l’ia

L’IA a été utilisée pour générer la partie visuelle des pages web, cependant, la partie fonctionnelle des pages n’ont pas été générées et la partie visuelle n’était également pas la focalisation du projet, c’est donc pour cela que je me suis permis de faire en sorte que la partie visuelle soit plus fluide.