

Documentation du système d'authentification et de chat

Introduction

Dans cette partie du projet, nous avons mis en place un système d'authentification et de chat général pour une application web. L'objectif principal était de permettre aux utilisateurs de se connecter à l'application et de pouvoir participer à un chat général. Les technologies utilisées pour réaliser ce projet sont PHP, JavaScript, HTML, CSS, SQL (pour interagir avec la base de données), et WebSocket pour la communication en temps réel.

1. Mise en place du système d'authentification

La première partie consistait à mettre en place un système d'authentification pour les utilisateurs. Pour cela, nous avons utilisé PHP et SQL.

1.1. Connexion à la base de données

Nous avons utilisé PDO (PHP Data Objects) pour se connecter à notre base de données MySQL.

1.2. Authentification

Lorsqu'un utilisateur soumet le formulaire de connexion, le serveur récupère les informations envoyées (nom d'utilisateur et mot de passe), puis vérifie si ces informations correspondent à un utilisateur dans la base de données. Si c'est le cas, une nouvelle session est créée et les informations de l'utilisateur sont stockées dans cette session. L'utilisateur est ensuite redirigé vers la page d'accueil de l'application.

2. Mise en place du système de chat général

La seconde partie était la mise en place d'un chat général en utilisant la technologie WebSocket pour permettre une communication en temps réel.

2.1. Mise en place du serveur WebSocket

Nous avons utilisé la bibliothèque Ratchet en PHP pour mettre en place notre serveur WebSocket. Ce serveur permet d'écouter les nouvelles connexions, les messages reçus, les déconnexions et les erreurs. Lorsqu'un message est reçu, le serveur le renvoie à tous les clients connectés.

2.2. Mise en place du client WebSocket

Nous avons utilisé JavaScript pour mettre en place notre client WebSocket. Ce client permet de se connecter au serveur WebSocket, d'envoyer des messages et de recevoir des messages du serveur. Lorsqu'un message est reçu, il est affiché dans la zone de chat de la page HTML.

3. Intégration du nom d'utilisateur dans le chat

Pour finir, nous avons modifié notre client WebSocket pour intégrer le nom de l'utilisateur dans le chat. Pour cela, nous avons récupéré le nom de l'utilisateur à partir de la session PHP, puis nous l'avons intégré dans le message envoyé au serveur WebSocket.

Conclusion

Cette partie du projet nous a permis de mettre en place un système d'authentification et un chat général en utilisant diverses technologies telles que PHP, JavaScript, HTML, CSS, SQL et WebSocket. Cela a permis aux utilisateurs de se connecter à l'application et de participer à un chat général.