

Application de surveillance de seuil d'humidité et de température dans une salle de serveurs

Fonctionnalités de l'application :

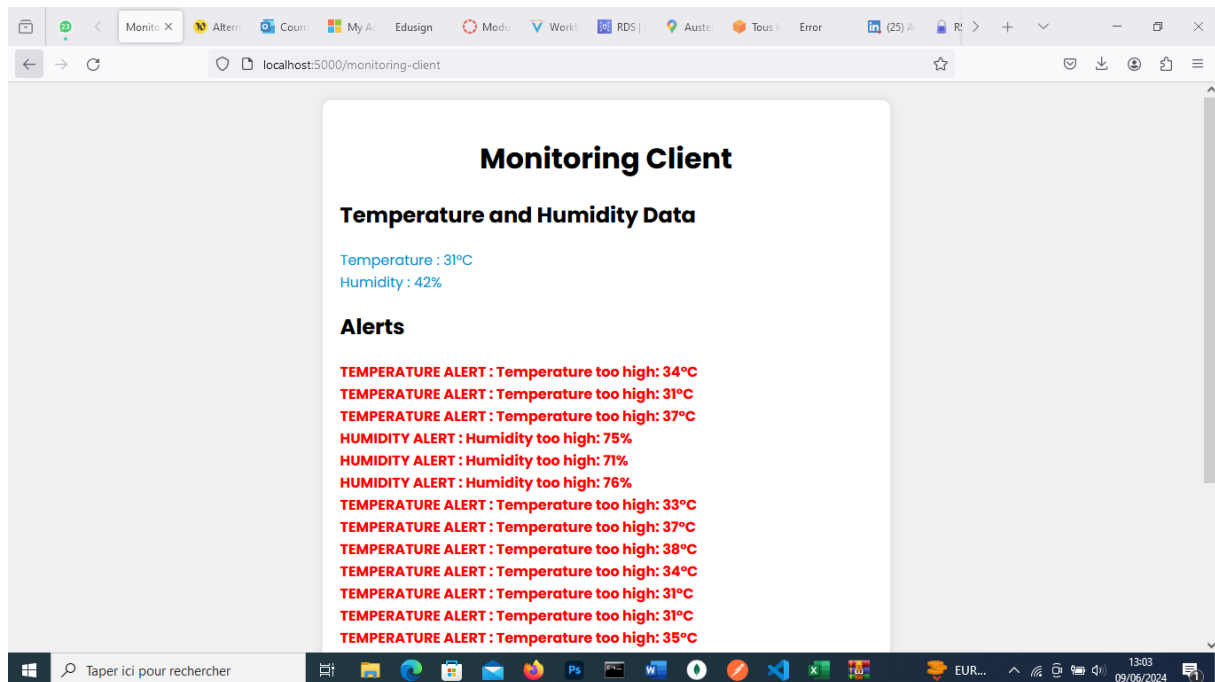
- Inscription, authentification et déconnexion des utilisateurs : Un utilisateur aura la possibilité de s'inscrire et de s'authentifier grâce aux routes /register, /login, et /logout implémentés. Les champs seront validés avant la création de l'utilisateur et son stockage dans mongoDB.

- Surveillance en temps réel des données : les données seront surveillées grâce aux objets data définis pour contenir chaque température et humidité à un instant donné. Tout ceci possible par les socketIO et la classe SensorSimulator qui simule les données aléatoirement

- Alertes en cas de lectures anormales des capteurs : L'utilisateur dès qu'il est connecté s'abonne aux événements et donc reçoit chaque 5s des alertes lui notifiant les changements de valeurs

- Authentification sécurisée des utilisateurs basée sur des jetons à l'aide de JWT signés avec RSA : tout les ressources de l'application sont protégées à l'exception de la route /register qui permet de créer un utilisateur. L'authentification se fait juste après la connexion de l'utilisateur en créant un token JWT signé avec l'algorithme RSA et garantit le maintien de la session. Le token sera détruit lorsque l'utilisateur va se déconnecter.

- Intégration de MongoDB pour le stockage des données comme le mot de passe qui est haché et salé avant son enregistrement dans la DB : Les données de l'utilisateur créé sont stockées dans mongoDB pour lui permettre de se connecter facilement. Par ailleurs afin d'éviter que quelqu'un de mal intentionné récupère les informations des utilisateurs, les mots de passes ne sont pas stockés en clair dans la DB. Elles ont été hachées et salées avant leur stockage



Capture d'écran montrant les alertes émis chaque 5s lorsque l'utilisateur s'est connecté et a écouté les évènements

Afin de relancer automatiquement l'application en cas de crash, l'outil PM2 a été installé. C'est un outil de gestion de processus installé en faisant `npm install pm2 -g`. PM2 offre la possibilité de redémarrer automatiquement une application en cas de crash en utilisant l'option `--watch`: `pm2 start server.js --watch`. Cette option permettra à PM2 de surveiller les fichiers de l'application et de redémarrer le processus si l'un d'eux est modifié. Si elle plante pour une raison quelconque, PM2 la redémarrera automatiquement, ce qui permettra de limiter les temps d'arrêt.

