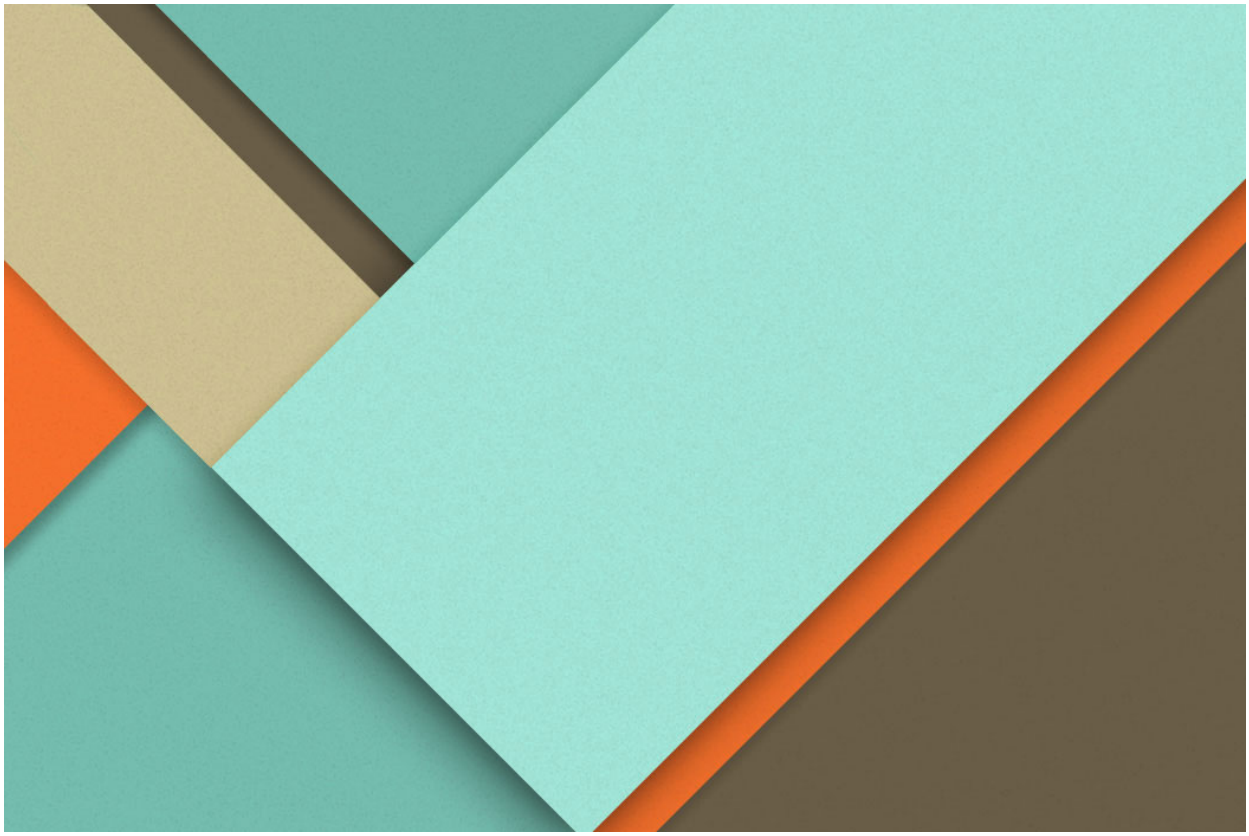

Architecture WEB

29.11.2023

Membres du Groupe

Jiad ABDUL

GLIN-DAYI Faithgot

Paul Charbel

Yassine HAMIL

Arthur ZAJAC

Vue d'ensemble

Le module d'authentification que nous concevons a pour objectif de permettre aux utilisateurs de se connecter aux sites des compagnies membres en utilisant l'identifiant enregistré sur notre plateforme. Il jouera un rôle crucial dans la gestion des accès sécurisés aux différents services offerts par ces compagnies.

Objectifs

1. Centralisation de l'authentification : Offrir une solution centralisée où les utilisateurs peuvent s'authentifier une fois et avoir accès à plusieurs sites membres sans inscription.
2. Sécurité : Mettre en place des mécanismes robustes de sécurité pour protéger les informations d'identification des utilisateurs.
3. Facilité d'utilisation : Concevoir une interface conviviale pour que les utilisateurs puissent se connecter facilement sans complications.

Caractéristiques

Nous les subdivisons en fonction de nos deux principales entités que sont: les compagnies et les utilisateurs.

Company

1. Gestion des compagnies : Enregistrement, connexion (uniquement via notre interface), modification, et suppression des comptes utilisateurs.
2. Changement de mot de passe en deux processus : Génération du token unique et validation du token.
3. Protocoles de sécurité : Utiliser des protocoles de sécurité standards tels que HTTPS, cryptage des mots de passe, et éventuellement l'authentification via une interface visuelle.
4. Authentification unique (SSO) : Permettre aux utilisateurs de se connecter une fois et d'accéder à tous les sites membres sans avoir à se reconnecter. (Idéale non concrétisée)
5. Journalisation des activités : Enregistrer les activités liées à l'authentification pour des raisons de sécurité et de suivi (Idéale non concrétisée) .

Users

1. Gestion des utilisateurs : Enregistrement,connexion (via notre interface et via la plateforme de la compagnie), modification et suppression des comptes utilisateurs.
2. Changement de mot de passe en deux processus : Génération du token unique et validation du token.
3. Protocoles de sécurité : Utiliser des protocoles de sécurité standards tels que HTTPS, cryptage des mots de passe, et éventuellement l'authentification via une interface visuelle.
4. Authentification unique (SSO) : Permettre aux utilisateurs de se connecter une fois et d'accéder à tous les sites membres sans avoir à se reconnecter. (Idéale non concrétisée)
5. Journalisation des activités : Enregistrer les activités liées à l'authentification pour des raisons de sécurité et de suivi (Idéale non concrétisée) .

Stack

MERN (MongoDB, Express, React, Node.js)

I. Frontend

A. React

B. Tailwind CSS

II. Backend

A. NodeJS

B. MongoDB

III. Ops et DB

A. NoSQL - [MongoDB] (ORM - [Mongoose])

B. SandBox Openshift

C. Netlify

IV. Syntaxe, libraries et concepts

A. JSON

B. Markdown
C. Tailwind CSS
D. Bcrypt JS
E. Cookie-Parser
F. CORS
G. Debug
H. Express
I. Express-async-handler
J. JsonWebToken
K. Mongoose
L. Morgan

V. Outils

A. Github & Git
B. IDE
C. POSTMAN

Grandes étapes

1. **Formation de l'équipe**
2. **Analyse des besoins** : Comprendre les exigences spécifiques des compagnies membres et des utilisateurs.
3. **Conception de l'architecture** : Définir l'architecture du module, les flux d'information, et les interactions avec les sites membres.
4. **Partage des rôles**: Définition des rôles entre les membres du groupe en fonction du centre d'intérêt de chacun.
5. **Implémentation** : Développer le module en suivant les meilleures pratiques de développement web et respectant les principes de l'agilité qui est notre méthodologie appliquée..
6. **Tests** : Effectuer des tests manuels approfondis pour garantir la sécurité et la fonctionnalité du module (L'idéal (pas concrétisé) est de faire aussi des tests automatisés si on pense maintenir le projet à long terme). L'API en Backend a été testé grâce à Postman.
7. **Déploiement** : Mettre en place le module sur une plateforme pour permettre intégration des sites membres. Le Backend a été tourné en ligne grâce au service

d'un mois gratuit de *Red Hat OpenShift* et tourne ce lien. Le Front quant à lui est déjà relié au back et tourne sur ce lien.

8. **Documentation, formation et support** : Mise à disposition sur Github d'une documentation appropriée aux utilisateurs et assurance d' un support technique continu via Discord ou par assistance/interaction physique.

Difficultés et Objectifs non atteints

1. **Délais insuffisant** : Notre première difficulté est liée au temps imparti pour le projet et dont la majorité est passée en entreprise sur d'autres projets.
2. **Sécurité** : On aurait beaucoups plus protéger certaines roots que la façon dont nous l'avions fait. Nous l'avions implémenter mais compte tenu du trafic à ces dernières heures nous n'avons pas ajouté le snippet de code qui restreint les routes protégées.
3. **Non finition de certaines vues front** : On a fini avec l'API Backend mais certaines pages du frontend, compte tenu du temps, n'ont pas pu être terminées. Il s'agit entre autres des pages de modifications des infos de comptes et de réinitialisation de mot de passe.