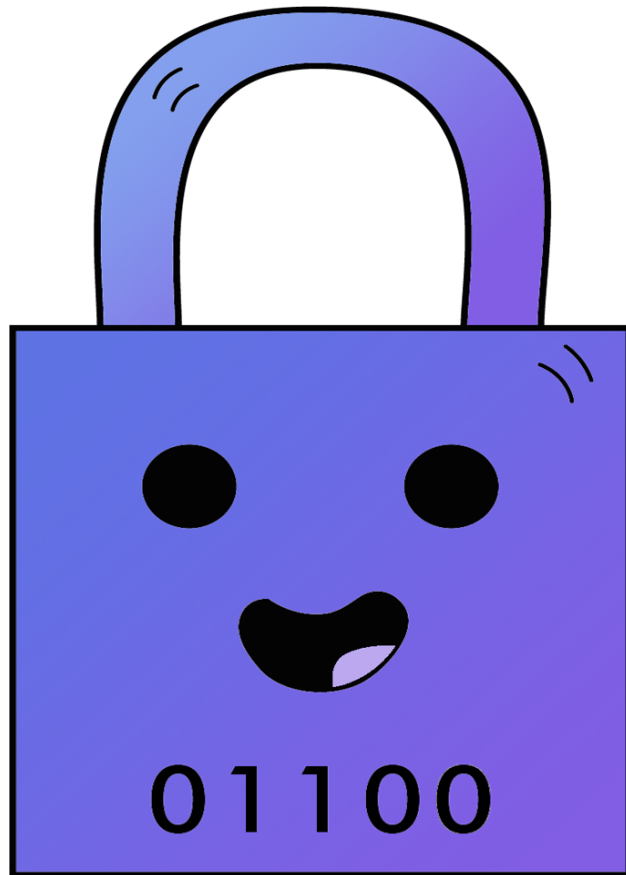


NotPassito



BOULLET Nicolas
GUAFFI Sandrine
NDOLO Adonis

4A-CFA

URL DU SITE

<http://54.38.189.228:8080/>

Pour se connecter:

- Soit créer un compte
- Soit se connecter avec les ID suivants :
 - email : adonis@lebg.com
 - password : leFrontMaVie

DESCRIPTION DU PROJET

NotPassito est un gestionnaire de mots de passe en ligne qui permet, en s'identifiant, d'accéder à ses différents mots de passe sauvegardés ou bien de créer une base et d'y sauvegarder un ensemble de mots de passe dans une base de données.

RESPECT DES CONTRAINTES TECHNIQUES

- VueJS
- Node.js
- CRUD
- Authentification utilisateur
- Persistance des données

ÉTAPES DE CONCEPTION

1. Définition des différentes fonctionnalités sur des cartes dans un Trello
2. Réalisation de mockup de l'application pour pouvoir réaliser une première version du site
3. Réalisation d'une convention de code. Grâce à cela il sera plus facile de trouver les noms de variables utilisés dans le projet et mais également de collaborer car le code est écrit à plusieurs
4. Réalisation d'une roadmap qui indique les principales versions du projet avec leurs dates de début et de fin pour pouvoir ne pas prendre de retard dans le planning

RÉALISATION

Nous avons d'abord commencé par faire une première version de test. Notamment en essayant de bien comprendre comment fonctionnait le template, les éléments que l'on devait supprimer ainsi que ceux que l'on devait garder. Pendant ce temps, il fallait également faire un CRUD en MySQL, pour créer et gérer les données. Mais également utiliser Axios pour communiquer avec une API en faisant des requêtes HTTP pour récupérer et sauvegarder les données. Nous nous sommes ensuite réunis pour mettre en commun ce que l'on avait fait et ainsi appeler les routes du back avec le front et récupérer le résultat pour afficher les données.

La deuxième étape était d'avoir une interface web conforme à nos attentes avec uniquement ce qui nous intéressait, donc il fallait supprimer des pages du template qui nous paraissaient inutiles puis ajouter les pages qui étaient manquantes tel que la page de connexion ou celle pour créer une nouvelle base.

Enfin la dernière étape de réalisation concerne le SGBD. Nous avons également jugé la criticité d'un mot de passe. Des fonctions back ont été créées pour permettre de juger de la complexité d'un mot de passe en fonction d'une norme. Pour le moment l'unique norme utilisée est la norme de l'ANSSI. À terme, d'autres standards seront implémentés. Les fonctions du back ont été prévues pour analyser la composition d'un mot de passe, puis en fonction d'un standard donnée, ressortir un jugement.

DÉPLOIEMENT

Nous n'avons pas réussi à porter notre projet sur Glitch, étant donné que notre projet est le Back et le Front séparés, chaque dossier possède 2 fichiers package.json propre. Bien qu'en ayant build le front avec la commande `npm run build`, Glitch ne pouvait que charger l'arborescence de notre répertoire et non pas le front.

DIFFICULTÉS RENCONTRÉES

Nous avons eu un problème de base de données. Cette dernière étant hébergée en ligne. La base a fermé il a donc fallu rouvrir l'accès, après avoir entamé la procédure, on a dû attendre que ça soit fait au niveau de l'hébergeur. On a également eu d'autres problèmes avec la base de données notamment cette dernière se déconnecte de temps en temps.

A cause d'une mauvaise décision sur la conception de la base de données, lorsque l'on supprime un mot de passe d'une table, le lien entre la base, qui contient les tables, et la table est supprimé pour la table ciblée. On a donc dû trouver une astuce temporaire pour rendre une première version à temps. Cette astuce est d'ajouter une condition sur la suppression : si la table contient d'autres mot de passe, alors on la re créera par la suite après suppression du mot de passe. Sinon on supprime tout simplement le mot de passe ainsi que la table. Nous n'avons pas réussi à mettre notre projet sur Glitch, à cause de la configuration de notre projet.