

Projet Redac

[Présentation du projet](#)

[Fonctionnalités principales](#)

[Technologies utilisées](#)

[Structure algorithmique](#)

[Rôle de chaque partie](#)

[Capture d'écran](#)

Présentation du projet

Un site web de customisation de produits de mobilité. L'idée originale derrière le site est d'acheter en s'amusant, c'est-à-dire sélectionner un produit le modifier à sa guise en y ajoutant par exemple des couleurs. C'est le principe de **Gamefication**. Les choix des produits s'est porté sur des trottinettes électrique. La seconde idée est de simplifier la navigation côté utilisateur et simplifier la gestion du site côté administrateur.

Fonctionnalités principales

Les fonctionnalités principales comprennent :

1. L'utilisateur n'a pas besoin de compte pour choisir un produit et le customiser.

La customisation comprend :

1. Changement de la couleur d'un produit
 2. Ajout d'un produit
2. L'utilisateur a la possibilité d'ajouter un produit au panier et le payer.
 3. L'utilisateur retrouve ses produits dans le panier même après une absence d'activité sur le site.

4. L'administrateur à la possibilité de gerer les produits (CRUD) proposés à l'utilisateur.

Technologies utilisées

Pour un projet de petite envergure comme le nôtre, toutes technologies aurait été adaptée. Parmi les membres du groupe, le langage Javascript étant le langage le plus maîtrisé , nous souhaitons continuer à parfaire nos compétences tout en apprenant à utiliser une technologie qui nous fournirait une souplesse de developpement. Nous nous sommes mis d'accord afin d'utiliser un framework, une "boite à outils" qui serait capable de gérer la partie front et la partie back dans un seul et même projet. NextJs est un bon candidat. La fonctionnalité de routage basée sur des fichiers nous a permis de rapidement construire l'interface utilisateur et la partie API nous a permis de structurer nos données et de facilement organiser notre code. Finalement, la simplicité de déploiement que confère Vercel nous a permis de déployer le site en quelques minutes.

Afin d'organiser nos données, nous avons choisi d'utiliser MySQL. Nous avons créé 3 tables comportant des données utiles au site. Voici la liste des tables créé :

- Produit
- Client
- Transaction

Pour communiquer de façon simple avec la base de donnée, nous utilisons un ORM appelé Sequelize. Il nous permet notamment de simuler une base orienté objet.

Par la suite et tout au long du projet, nous avons mis en place Firebase authentication pour gerer les utilisateur. C'est donc la base de donnée Firestore fournit par Firebase que nous utilisons pour stocker les informations clients.

Pour gerer le state dans notre application web, nous utilisons Redux.

Ensuite, nous avons choisi d'utiliser des dépendances telles que Mantine pour les composants stylés et ThreeJS pour animer du contenu 3D dans le site.

Enfin, nous utilisons des API tel que Imgur pour le hosting d'images des produits vendus et Paypal API pour la vente/paiement des produits choisi par l'utilisateur.

Structure algorithmique

Pour ce projet, nous souhaitons pouvoir suivre l'état d'un utilisateur connecté ou non et l'état du panier tout au long de l'utilisation de l'application web. Nous utilisons donc Redux. La gestion de l'état d'un utilisateur connecté est un booléen qui est tester avec des ternaires dans les fichiers ou la connection est requise. La gestion du panier est un tableau qui se met à jour et met à jour le stockage local du navigateur pour que l'utilisateur puisse retrouver les informations du panier meme lorsque l'onglet du navigateur est fermé.

Sur la partie interface utilisateur, nous avons crée un Layout qui permet de simplifier davantage la création de nouvelles pages au site.

Nous utilisons le plus possible des interfaces pour suivre et typer nos props passées en paramètres à nos composants.

Rôle de chaque partie

Nous sommes 3 developpeur dans l'équipe : Loïc BOZON, Kenji NGUYEN et Mickaël Brunet

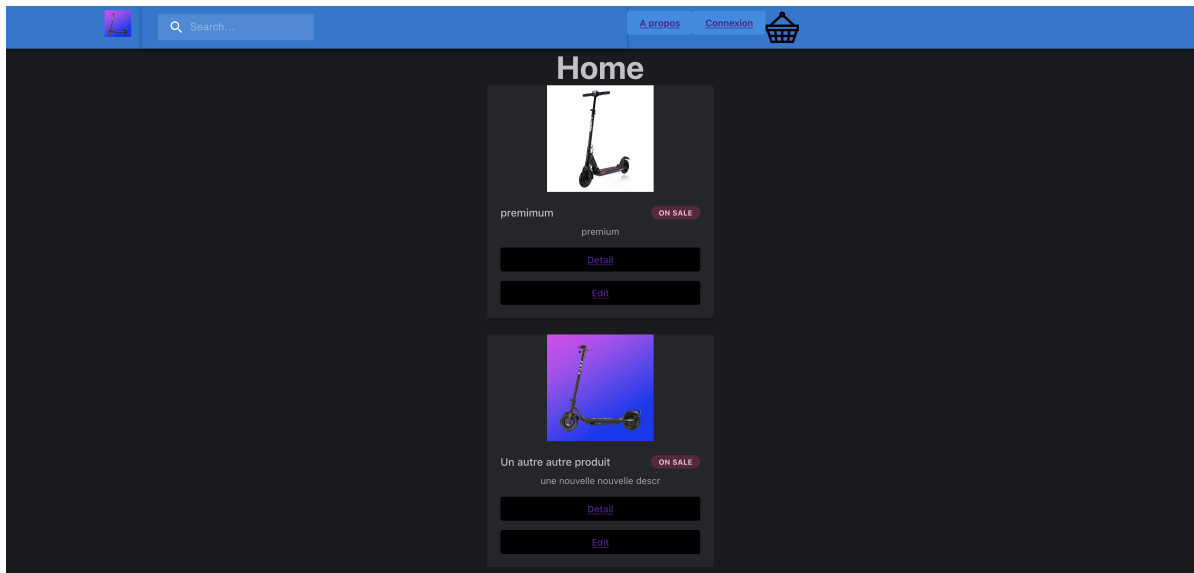
Kenji s'est occupé de la partie base de donnée. Il nous a crée la structure de la base.

Loïc s'est occupé de la partie interface utilisateur en créant les pages utiles au projet. De la page d'accueil à la page de description d'un produit.

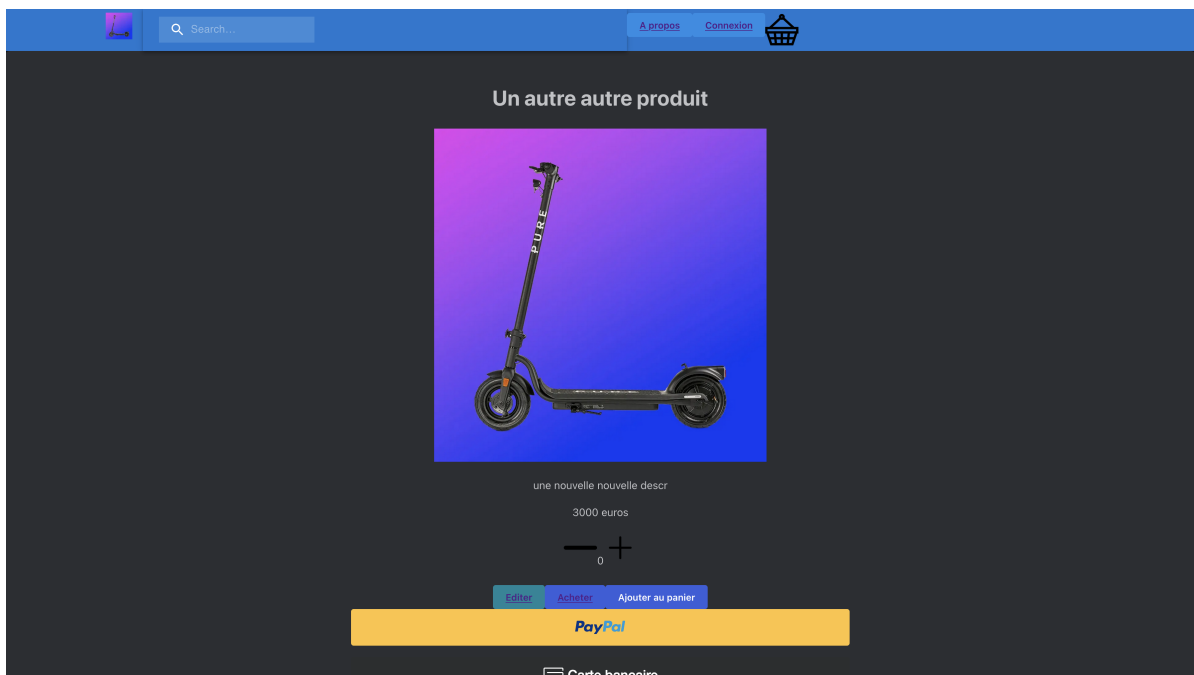
Mickaël s'est occupé de générer le rendu 3D, la gestion du panier et l'implémentation des API Imgur et PayPal.

Capture d'écran

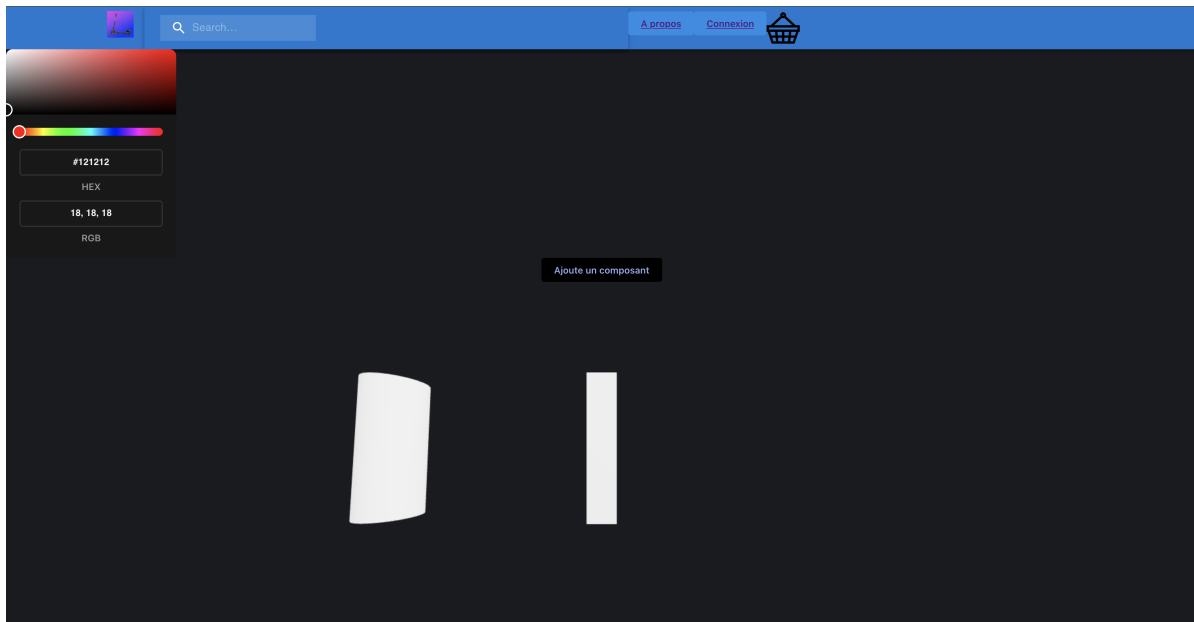
1. Accueil



2. Detail produit





3. Edition d'un produit



4. Achat


Test Store


  3 000,00 EUR


Bonjour John [Déconnexion](#)

Adresse de livraison
John Doe
Av. de la Pelouse, 75002 Paris
[Modifier](#)

Payer avec

☒  Solde PayPal 3 000,00 EUR
☐ Définir comme source d'approvisionnement préférée

☐  Visa
Crédit ****9195

☐  Rabobank Nederland
Courant ****1478

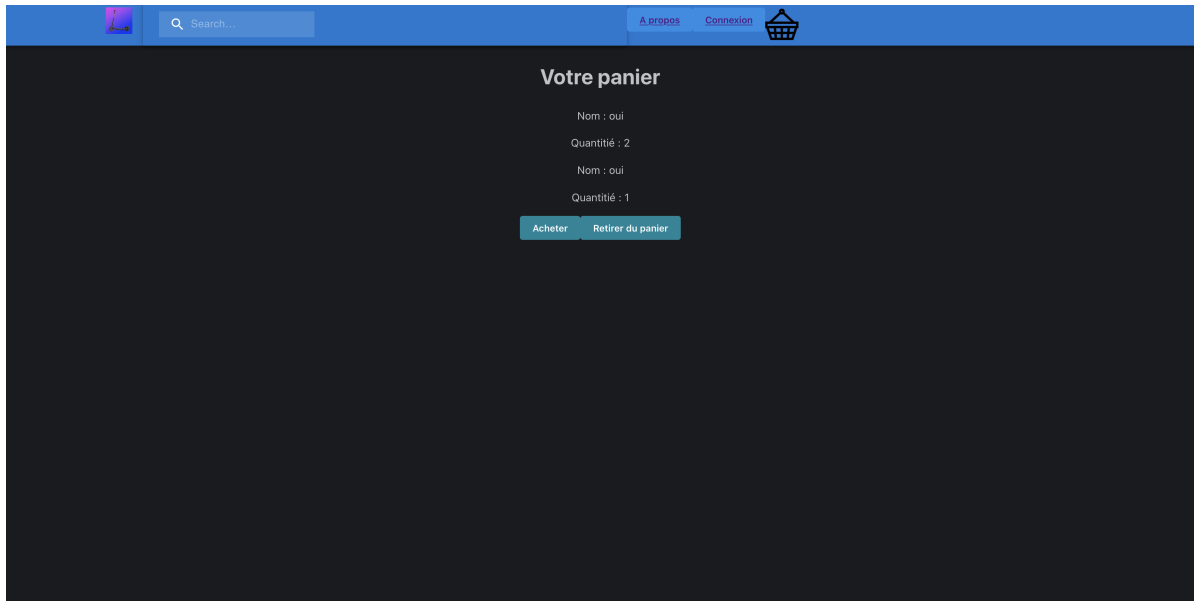
[+ Enregistrer une carte bancaire](#)

[Payer](#)

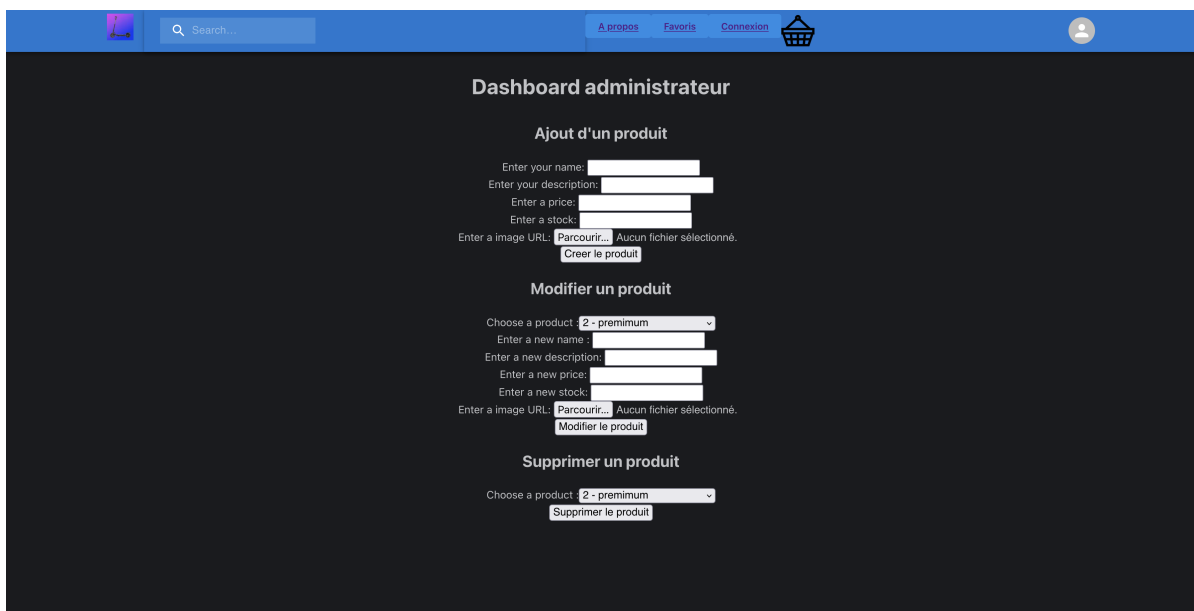
[Annuler et retourner sur Test Store](#)
© 1999 - 2022 [Contrats d'utilisation](#) [Conditions d'utilisation](#) [Respect de la vie privée](#)

5. Mise au panier

6. Page panier



7. Page Dashboard



8.