**CAHIER DES CHARGES**

Mélanie ZHU

# **Remerciements**

Je tiens à remercier toutes les personnes qui m’ont suivi, accompagné et soutenue durant cette période importante de ma formation.

Je tiens à témoigner de ma reconnaissance à mes camarades de classe pour leur soutien moral en organisant des sessions de travail.

**SOMMAIRE**

**Introduction et Analyse**

1. Compétences du référentiel couvertes par le projet
2. Présentation du site web
3. Objectifs

**Description graphique**

1. Charte graphique
2. Maquette sur Figma

**Description fonctionnelle et technique**

1. Arborescence du site
2. Spécifications fonctionnelles
   1. USER
   2. Administrateur
3. Technologies utilisées :
   1. Front end
   2. Back end
4. Spécifications techniques :
   1. Outils de développement
   2. Réalisation

**Introduction & Analyse**

**Compétences du référentiel couvertes par le projet**

* Développer la partie front end d’une application web ou web mobile en intégrant les recommandations de sécurité :
  + Maquetter une application
  + Réaliser une interface utilisateur web statique et adaptable
  + Développer une interface utilisateur web dynamique
  + Réaliser une interface utilisateur avec une solution de gestion de contenu
* Développer la partie back end d’une application web ou web mobile en intégrant les recommandations de sécurité
  + Créer une base de données
  + Développer les composants d’accès aux données
  + Développer la partie back end d’une application web ou web mobile
  + Elaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce

**Présentation du site**

Le site web est un site e-commerce qui vends des câbles de chargeurs de téléphone pour une marque appelée « E-rigation » qui a établi son empreinte en tant que leader incontesté sur le marché depuis 5 années consécutives.

Aujourd’hui, l’entreprise souhaite se lancer dans la vente en ligne afin d’épandre leur stratégie marketing à tous les consommateurs.

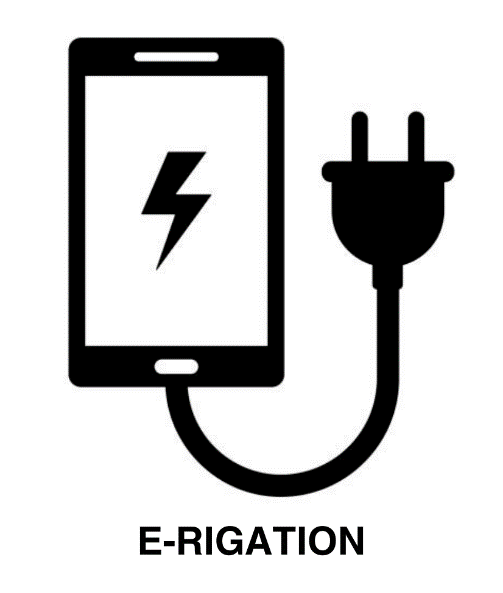
**Objectifs**

* Création d’un site web ergonomique, simple d’utilisation et adapté aux besoins de l’entreprise
* Fonctionnalités d’accès restreint : inscription et connexion obligatoire pour les utilisateurs autorisés
* Gestion des produits
* Panier
* Création d’un espace administrateur pour la gestion globale du système

**Description graphique**

**CHARTE GRAPHIQUE**

**LOGO**

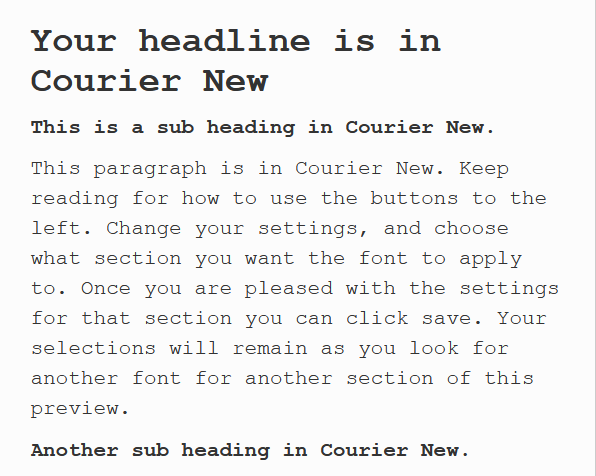


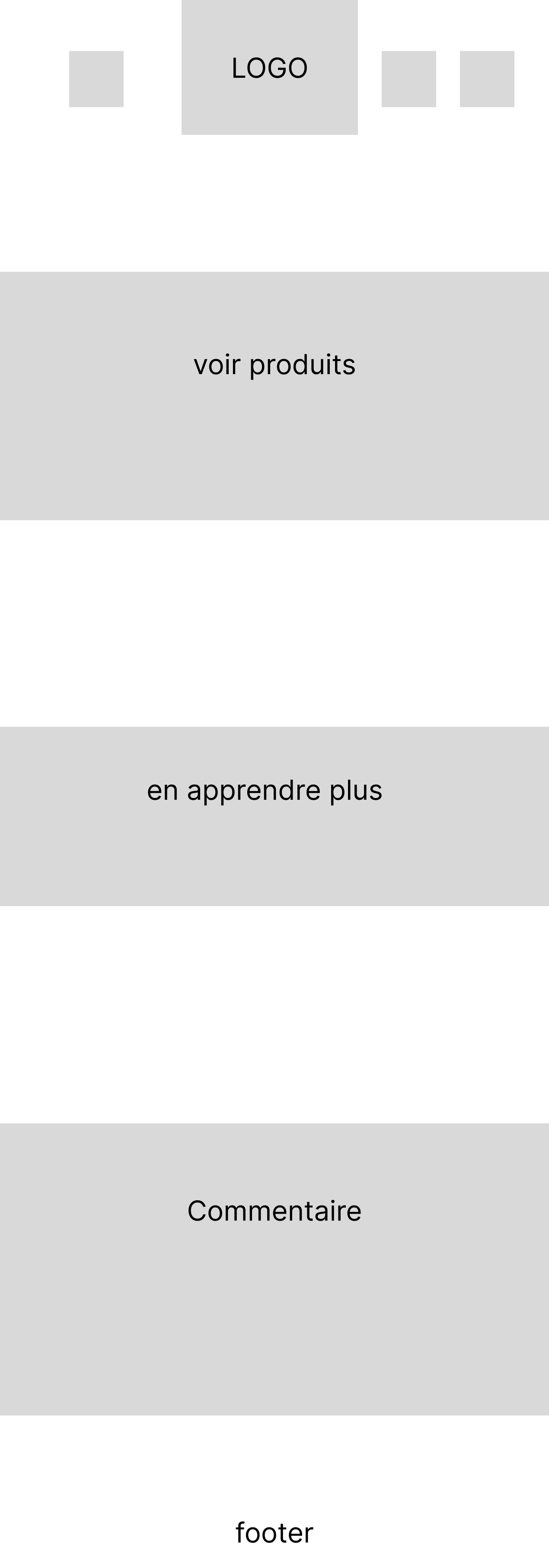
**Palette de couleur**

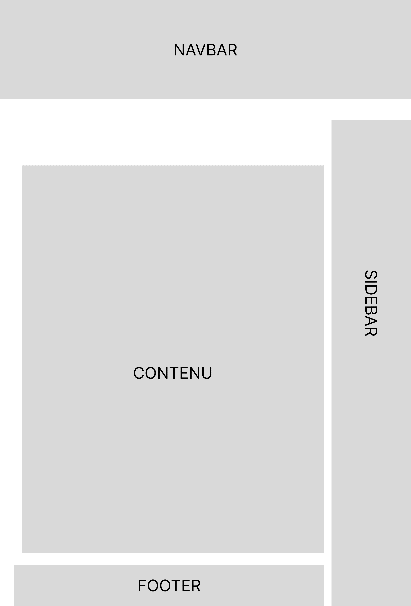


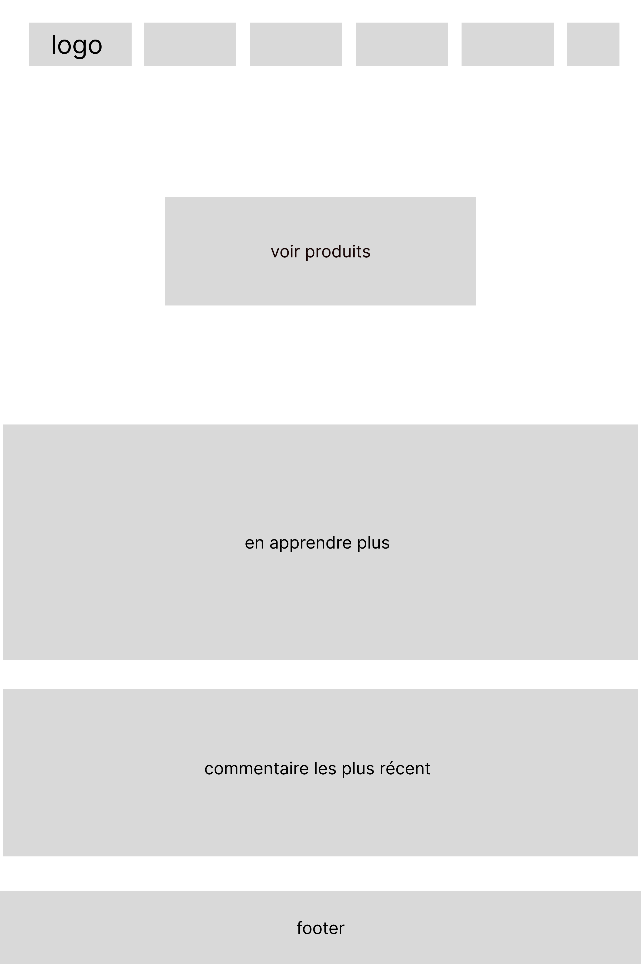
**Police d’écriture**

Courier New



**MAQUETTE FIGMA**

****

****

**DESCRIPTION FONCTIONNELLE ET TECHNIQUE**

**Arborescence du site**

Le site se divise en deux accès :

* Utilisateurs qui achètent les produits et peuvent modifier les produits
* Administrateurs qui gèrent les produits et gère les utilisateurs

Les fichiers sont séparés en plusieurs fichiers séparant chacune des pages et leurs CSS.

Ajouté à cela, un fichier components contient tous les composants ajoutés à leurs pages respectives.

**Spécifications Fonctionnelles**

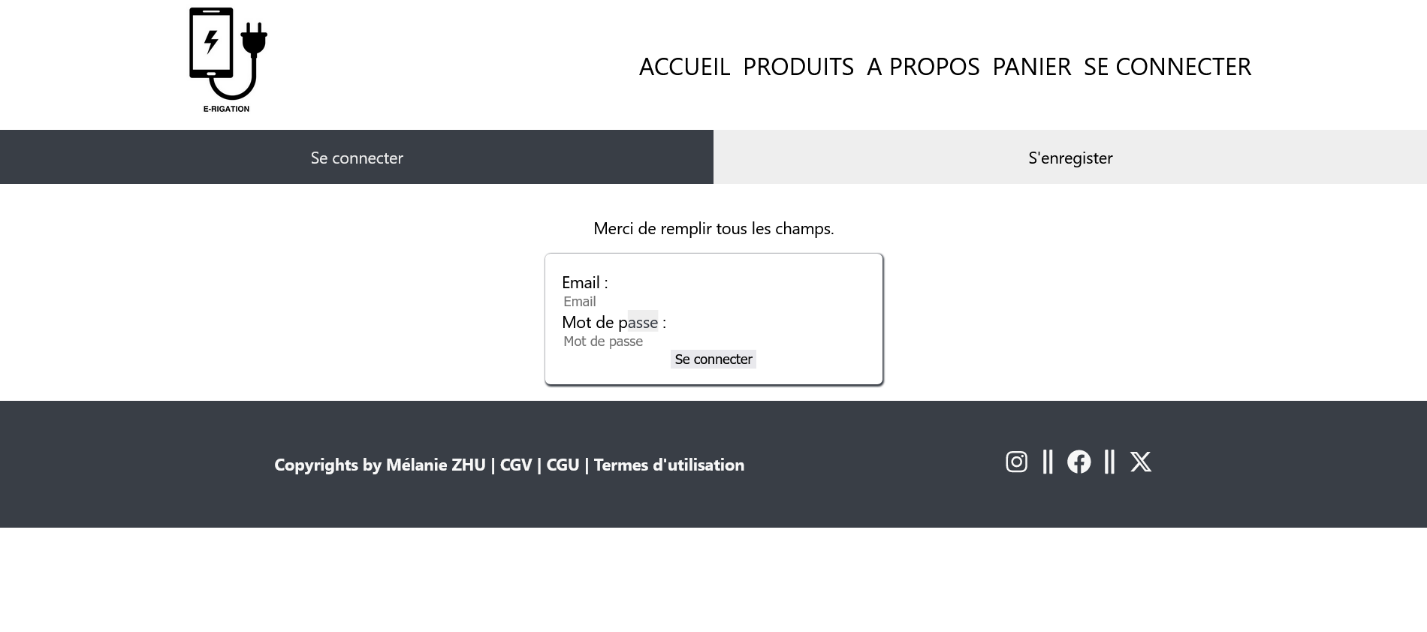
## **User**

**Page de connexion**

La page de connexion qui permet à l’utilisateur de s’enregistrer ou de se connecter, donne accès à des fonctionnalités supplémentaires du site.

Le formulaire de connexion demande deux champs obligatoire : mail et mot de passe.

En cas d’erreur de mot de passe ou de mail, un message d’erreur s’affiche « Wrong email or password ».

Si l’utilisateur n’a pas de compte, celui-ci a la possibilité d’accéder au formulaire d’inscription en cliquant sur « S’enregistrer ».

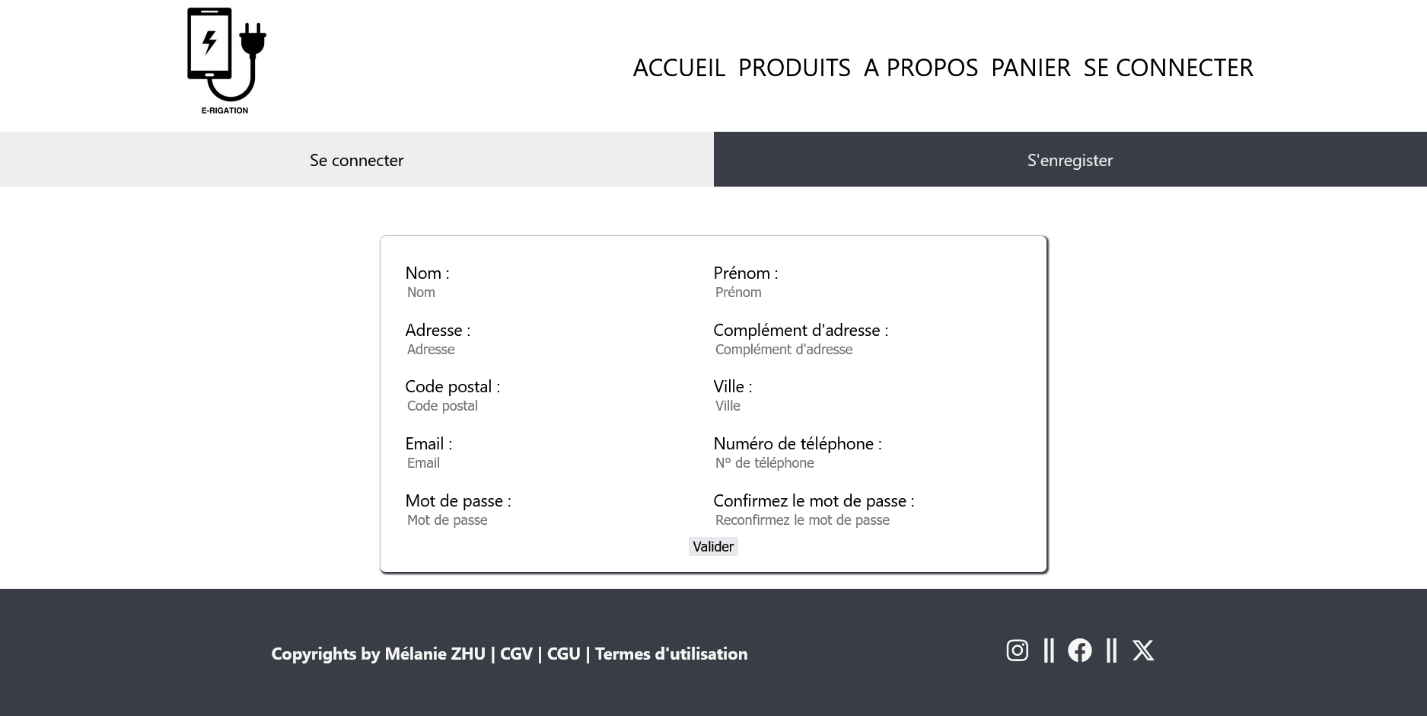
L’inscription a été sécurisé du côté front par un RegExp. Le Regular Expression communément appelée RegExp, est une chaîne de caractères qui impose une syntaxe précise à un champ.

Les utilisateurs doivent remplir tous les champs du formulaire pour s’inscrire. En cas de non-respect de ces consignes, des messages d’erreurs apparaissent.

Une fois l’inscription terminée et réussie, un message indique que le compte a été créer avec succès. L’utilisateur peut donc se connecter avec ce nouveau compte créer.

Les champs obligatoires à remplir dans le formulaire sont les suivants :

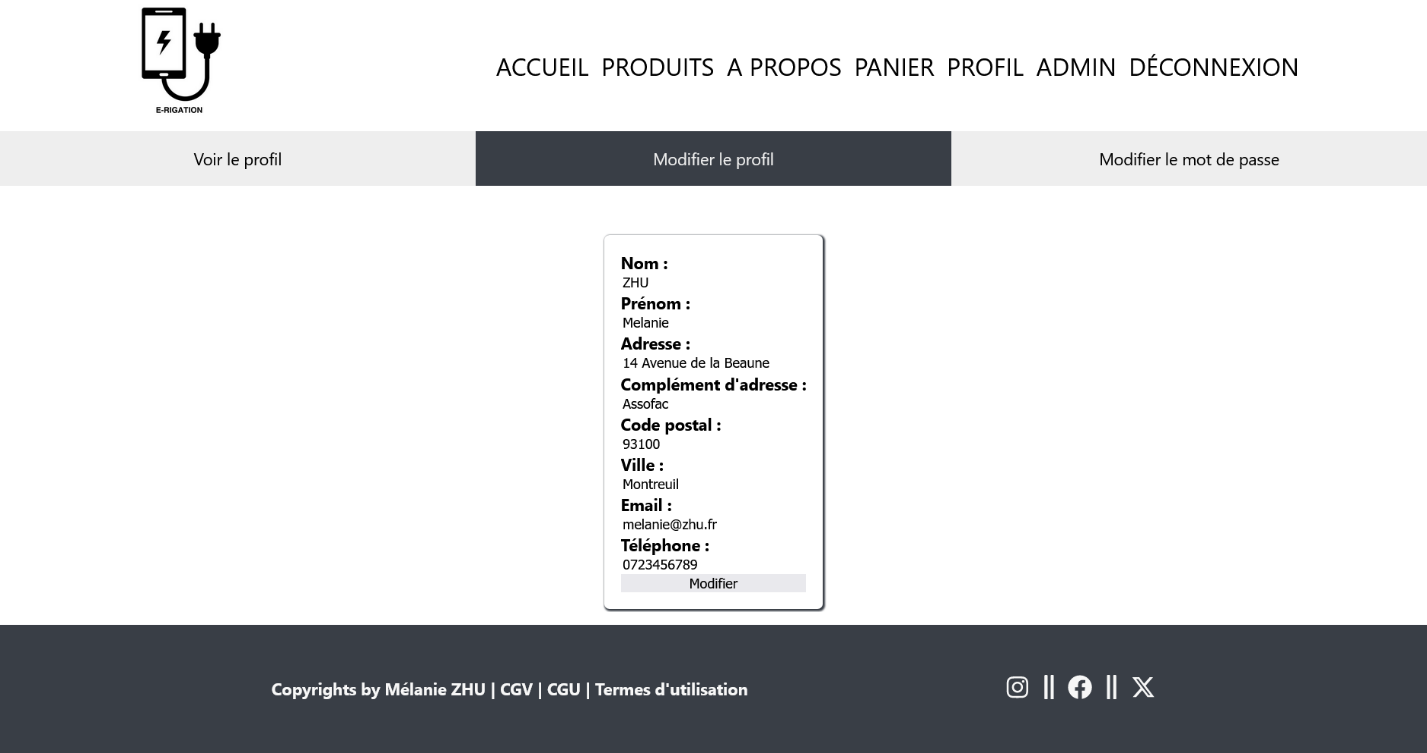
* Nom
* Prénom
* Email
* Adresse
* Téléphone
* Code postal
* Mot de passe
* Confirmation du mot de passe

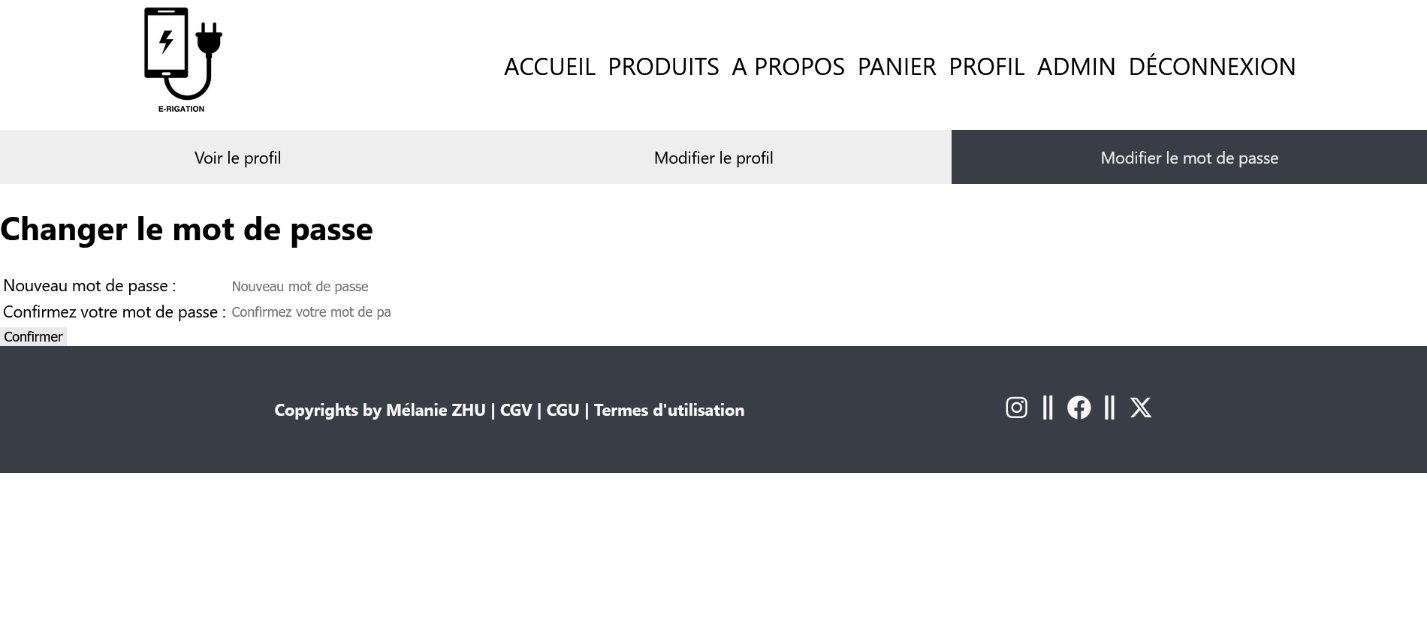
Tous les champs sont des inputs avec un placeholder ainsi que des boutons de type « button ».

**Page de profil**

La page profil permet d’avoir accès à nos informations personnelles (nom, prénom, email, numéro de téléphone, adresse). L’utilisateur peut changer ses informations et valider cette modification par le bouton « Modifier le profil ».



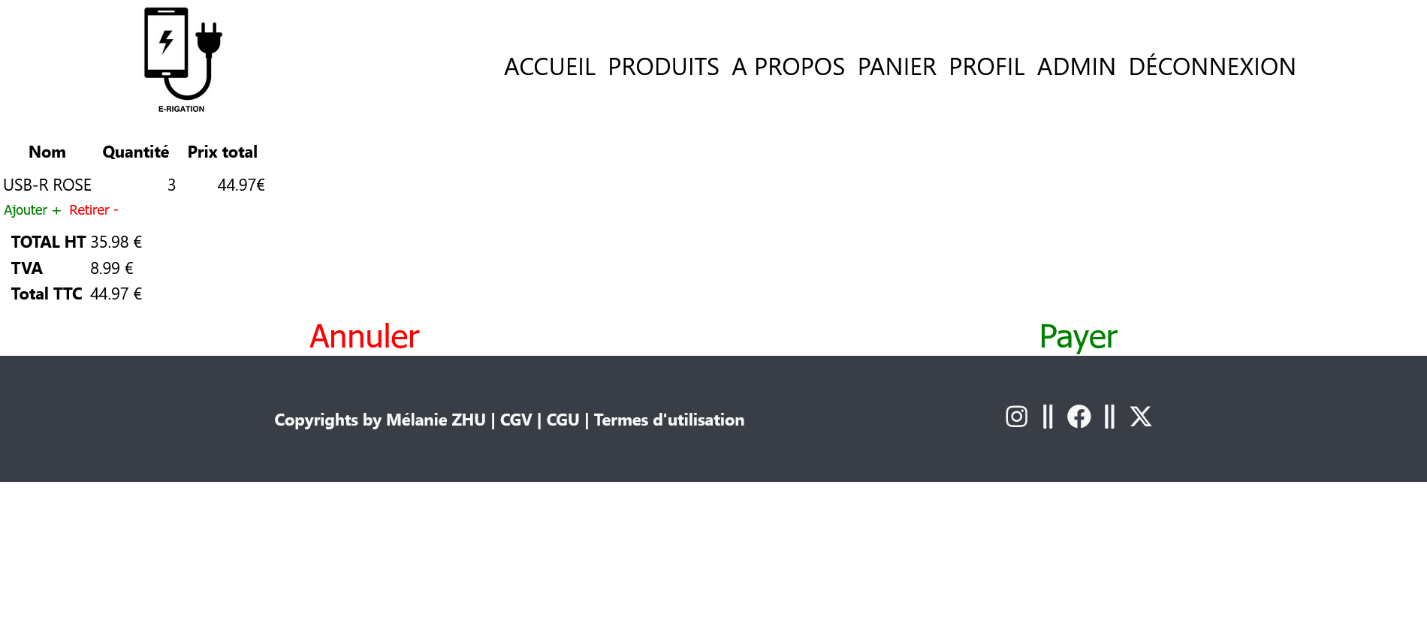
****

****

**Page panier**

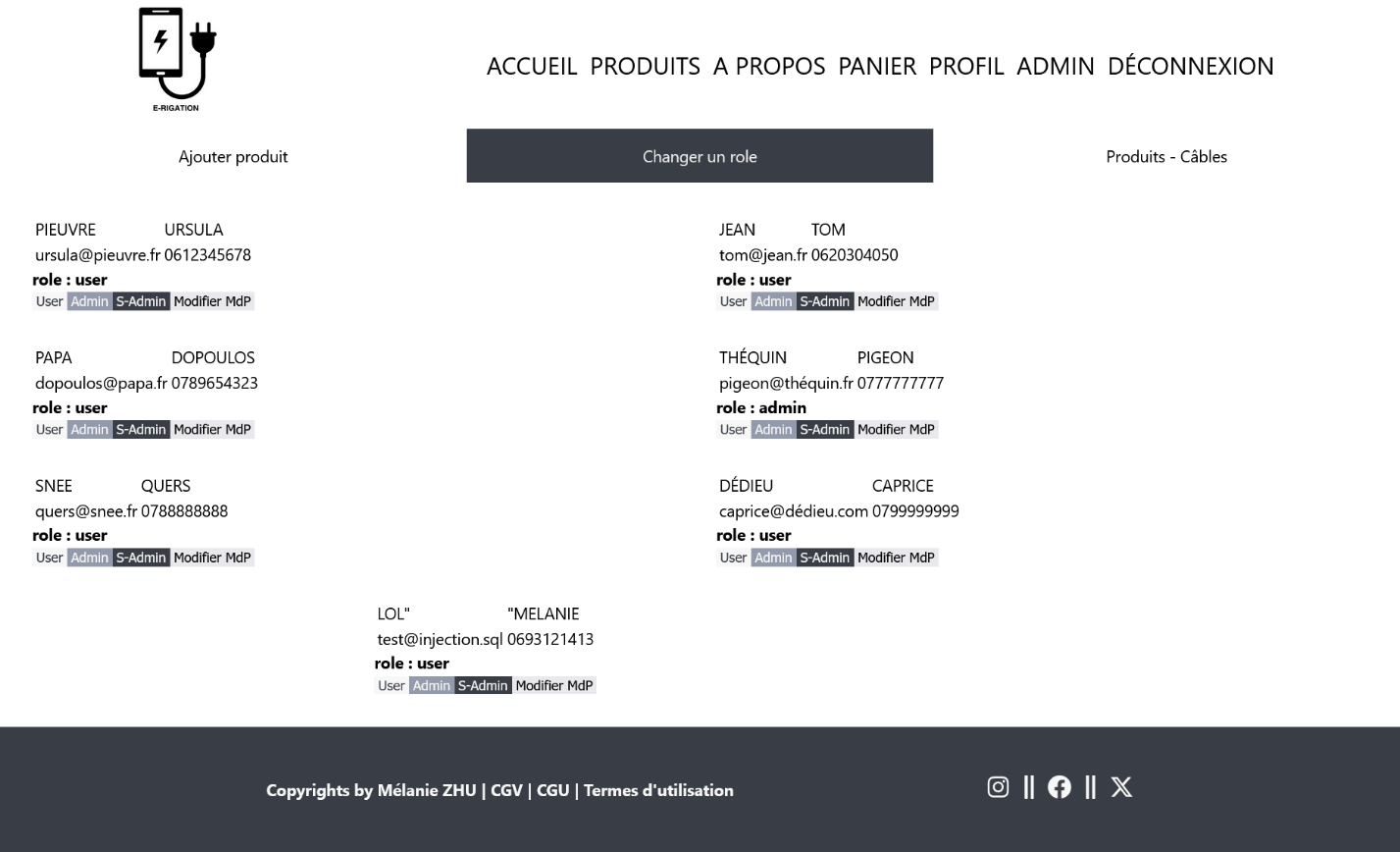
Permet de valider les produits ajoutés dans le panier et de passer une commande.



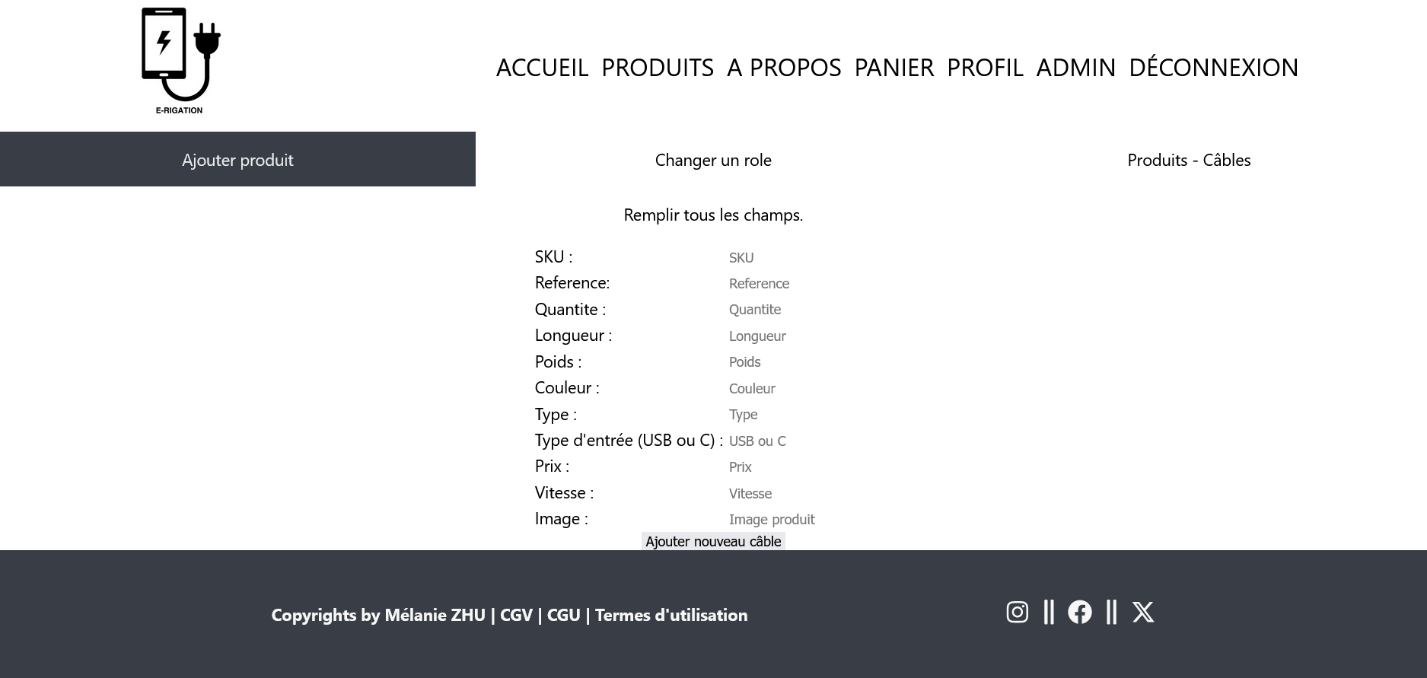
****

**Admnistrateur**

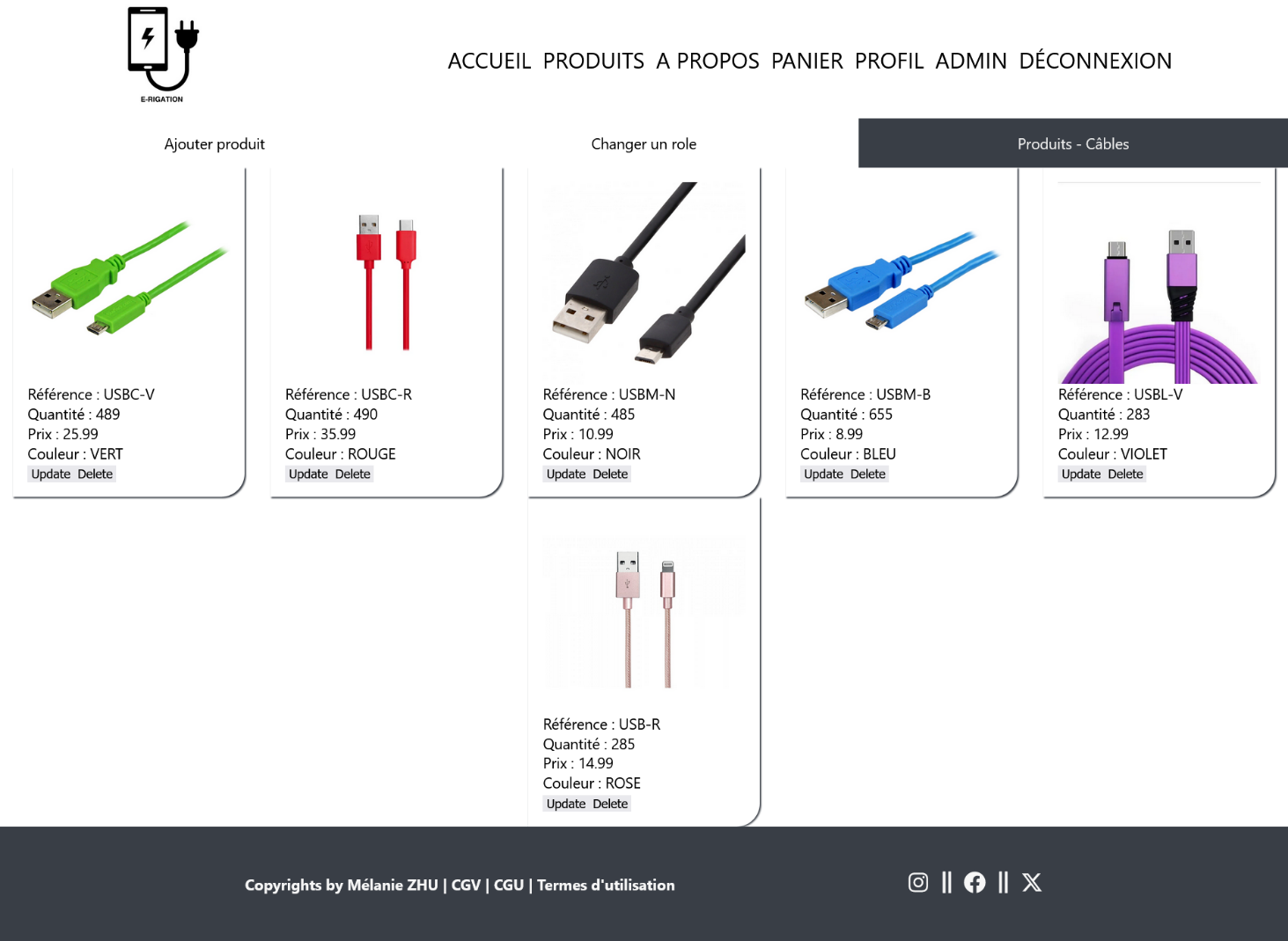
**Page gestion des utilisateurs**

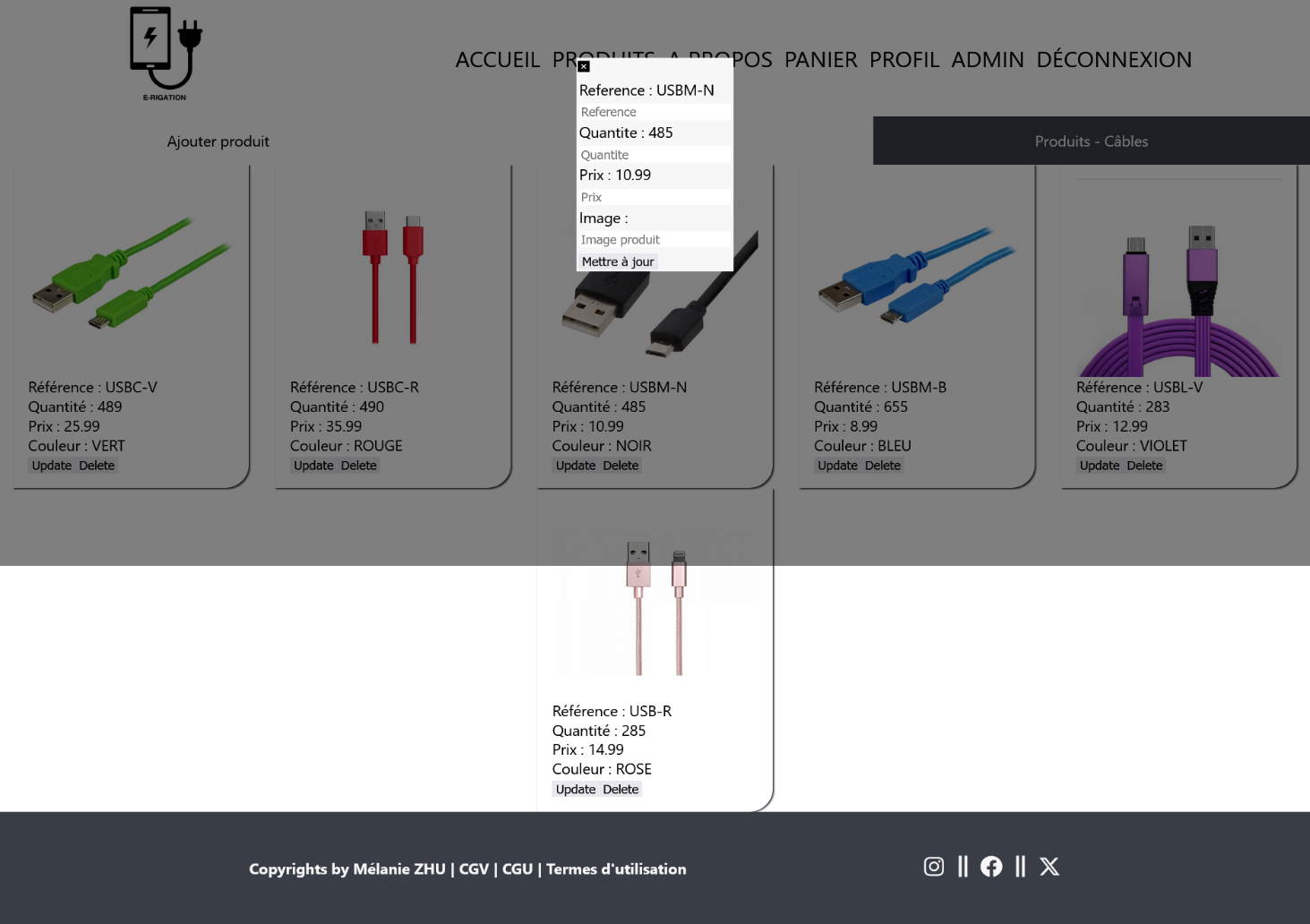
Permet de modifier le CRUD utilisateur ou produits. Sur cette page, l’administrateur peut choisir de modifier ou supprimer des produits, changer le mot de passe.

**Page ajouter un produit**



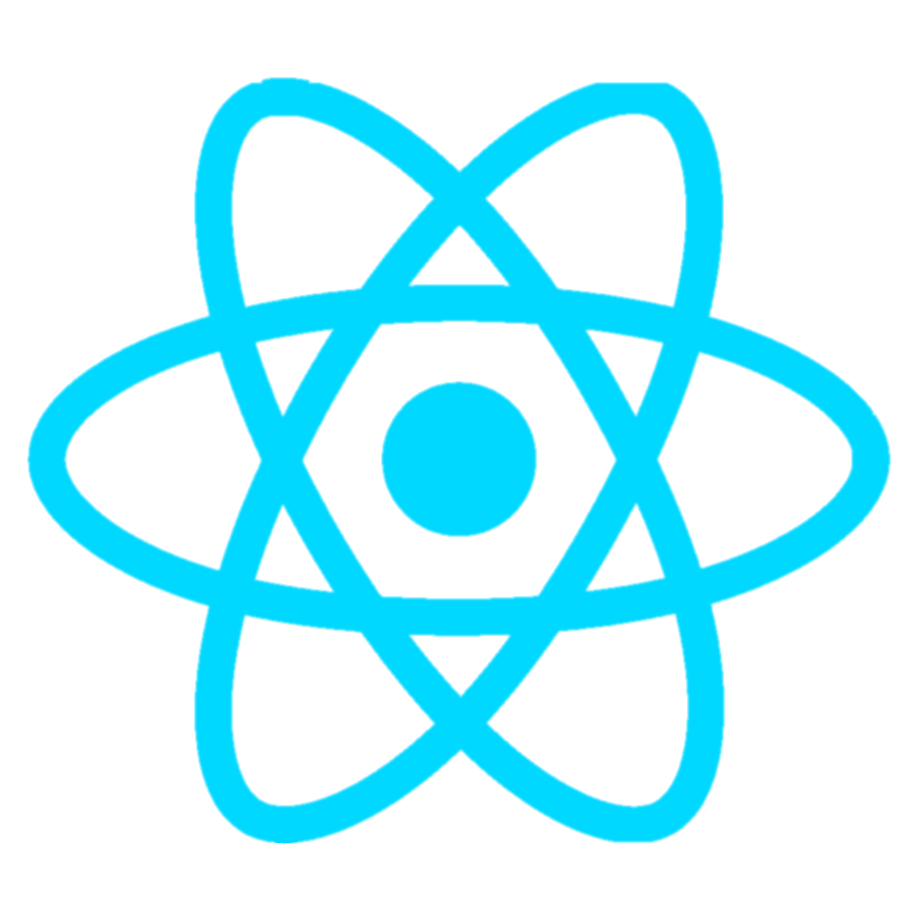
**Page modifier et supprimer un produit**



****

**Technologies utilisées**

**Front end**



ReactJS est une bibliothèque JavaScript open-source crée pas Facebook, utilisée pour créer des interfaces utilisateurs interactives et dynamiques. Il offre un modèle de composants réutilisables et facilite le développement d’applications web complexes.

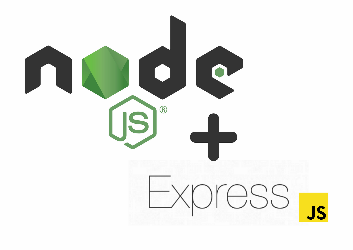
NodeJS est un logiciel qui permet d’exécuter du code JavaScript en dehors des navigateurs, principalement sur des serveurs. C’est très utile pour créer des sites web interactifs, applications côté serveur et d’autres services en ligne.

Axios prends en charge les requêtes http GET, POST, PUT et DELETE. Gère également les retours des réponses et la récupération des données JSON. Elle protège contre les attaques CSRF (Cross-Site Request Forgery) lors de l’envoie des requêtes avec les cookies.

LocalStorage : Permet de stocker des données utilisables à travers tout le site. C’est ainsi qu’un panier peut être stocké.

**Back end**

 MySQL est un système de gestion de base de données SQL (aka relationnelles). Cela signifie qu’il est conçu pour stocker et gérer des données sous forme de tables, avec des relations entre les différentes tables.

Facilite la création des serveurs WEB et la gestion des routes. Permet de gérer les erreurs qui se produisent lors du traitement des requêtes, ce qui permet de capturer et gérer les erreurs de manière centralisée.

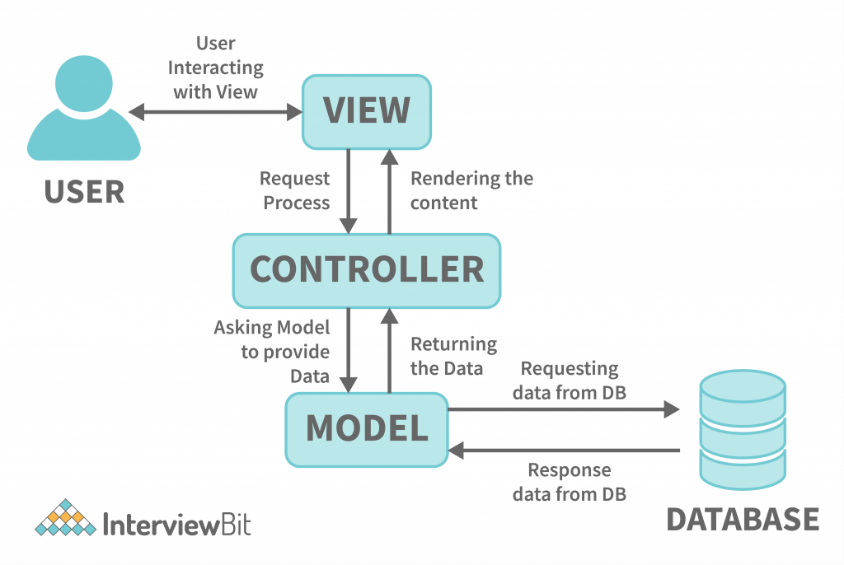
Mécanisme de sécurité qui contrôle l’accès aux ressources d’un site web depuis un domaine différent. Permet d’autoriser une requête provenant d’un autre domaine que celui sur lequel la ressource est hébergée.

**Outils de développement**



**GitHub** est crucial pour surveiller l’avancement du projet et garder une copie dans le cloud. Utilisé pour le back ou le front, c’est un outil nécessaire à tous les développeurs

**API Restfull :** J’ai développé une API en Node.js, elle permet d’effectuer des opérations telles que la récupération, l’envoi, la mise à jour et la suppression des données dans la base de données en utilisant des méthodes POST, GET, UPDATE et DELETE, ce qui corresponds au CRUD.

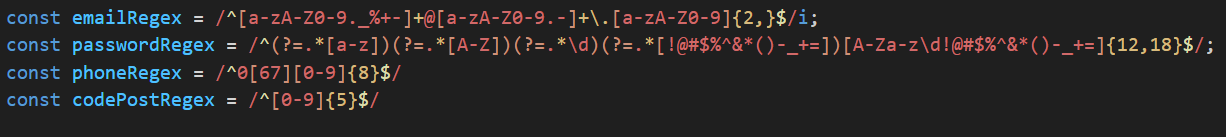
**Base de données :**J’utilise une base de données relationnelle MySQL pour stocker les données des formulaires. Cette base de données est ensuite connectée à l’API, permettant à l’application de stocker, récupérer et gérer ces données de manières organisée et efficace.

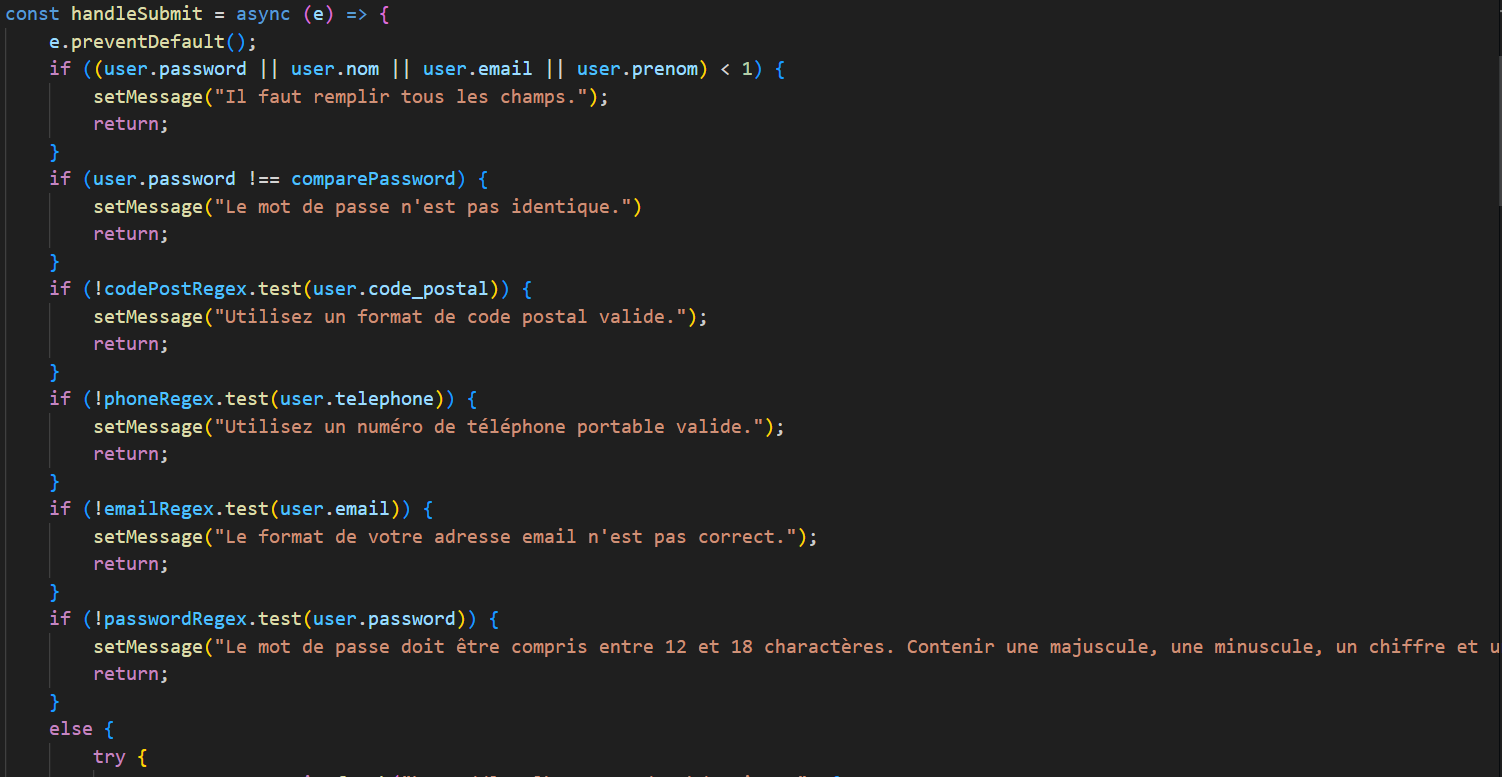
L’architecture **Modèle-Vue-Contrôleur(MVC)** est une méthode de conception logicielle qui divise une application en trois composants principaux : le modèle gère les données, la vue gère l’interface utilisateur et le contrôleur gère les interactions entre les deux.

**La sécurité**

**REGEX**

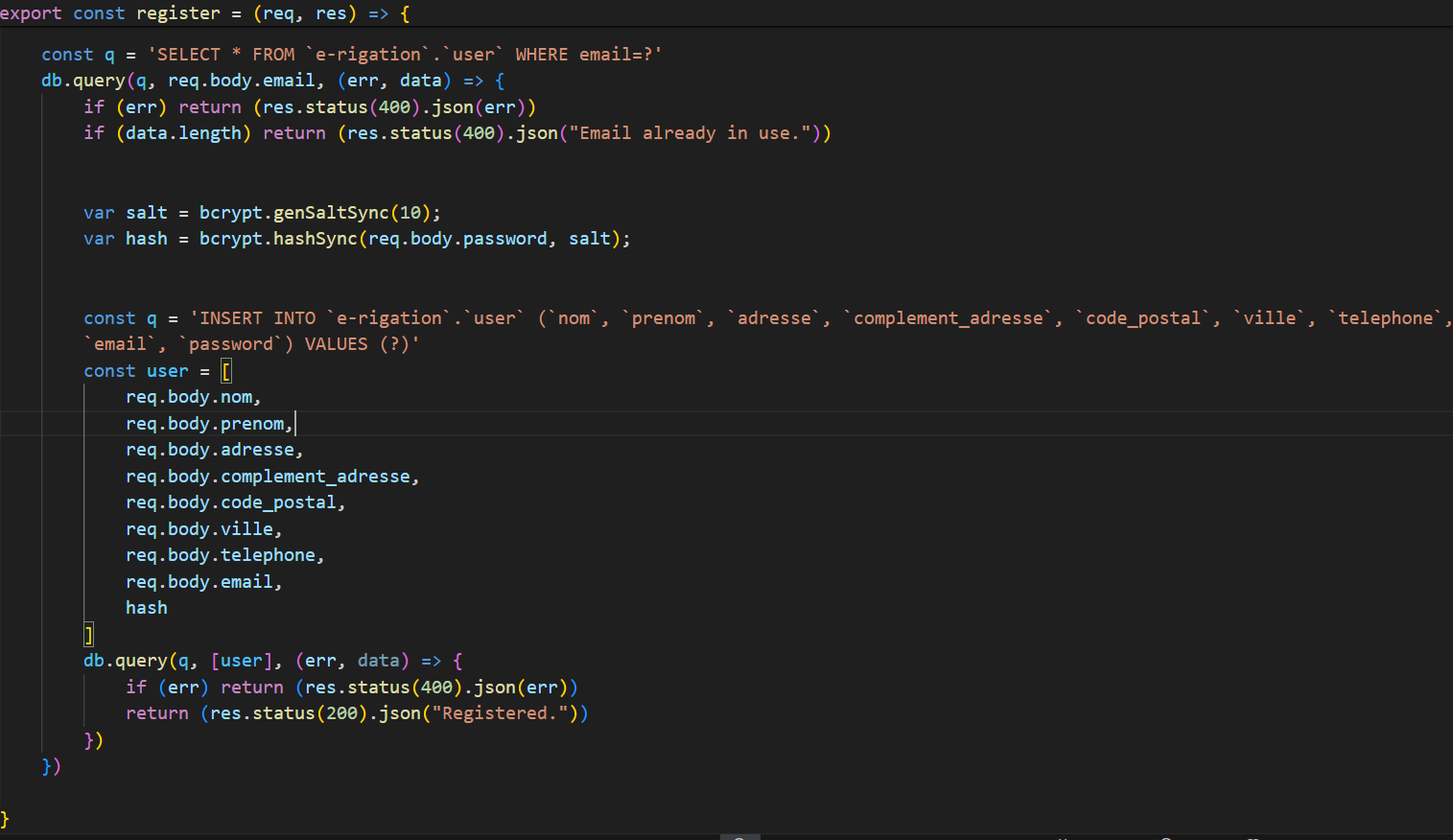
RegEx ou Expressions Régulières, permets de vérifier les champs, notamment les mots de passes, afin de garantir que l’utilisateur dispose d’un mot de passe sécurisé et conforme aux normes de sécurités recommandées par la CNIL.





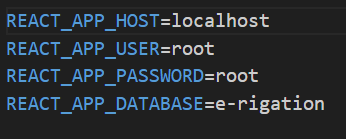
**Bcryptjs**

Permet de crypter/hasher le mot de passe de l’utilisateur selon le salt passé en paramètre.



**Dotenv**

Fichier d’environnement de l’ordinateur dans lequel on rends invisible des données sensibles.



**Réalisation**

**Conception de la base de données**

MCD (Modèle Conceptuel des Données)

Représentation graphique de ce que le système doit stocker et comment les différentes entités sont liées les unes aux autres.

