

# Test Technique - Développement Front End

immediate

## ABSTRACT

Le but de ce test est de développer une application web de type e-commerce permettant l'ajout d'article dans un panier et la validation de celui-ci.

Le code est disponible à cette adresse: [https://github.com/ElieSol/Web\\_Store](https://github.com/ElieSol/Web_Store).

## 1 Matériel et Méthodes

---

### Langages de Programmation

Les langages de programmations Javascript (version ES6), HTML5 et CSS3 ont été utilisé lors de l'implémentation de ce projet

### Framework

Le framework ReactJS a été utilisé afin de faciliter l'organisation du code et d'améliorer l'expérience utilisateur.

### Librairies

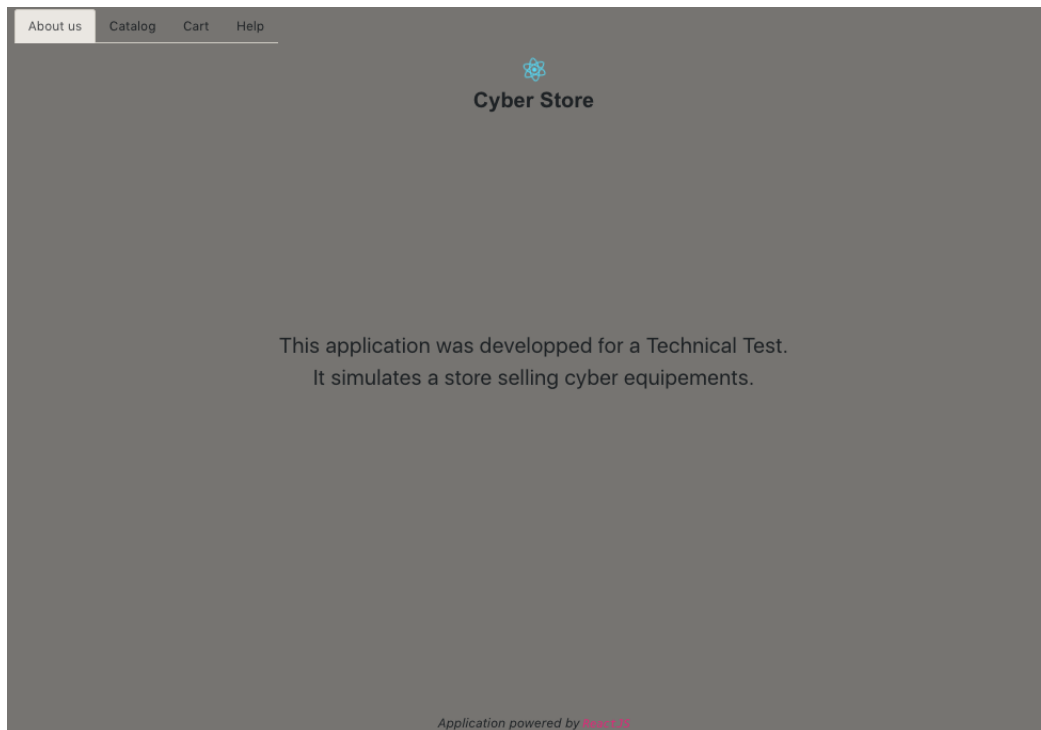
Différents librairies ont été utilisées dans ce projet:

- **Reactstrap**: Cette librairie permet d'utiliser des fonctionnalités de Bootstrap au sein du framework ReactJS. Elle a été utilisée dans ce projet principalement pour organiser les éléments de la page.
- **Jest**: Cette librairie est intégrée de base à ReactJS et permet de faire des tests.
- **Enzyme**: Une autre librairie permettant de faire des tests qui a été utilisé en complément de Jest

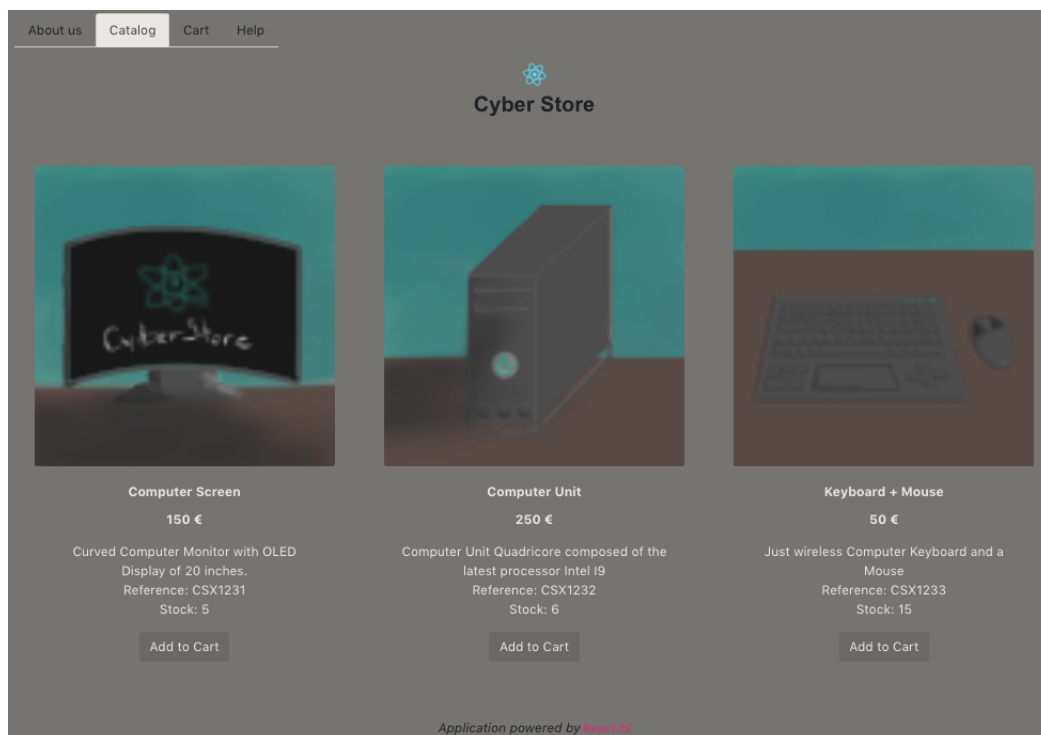
## 2 Rendu final

---

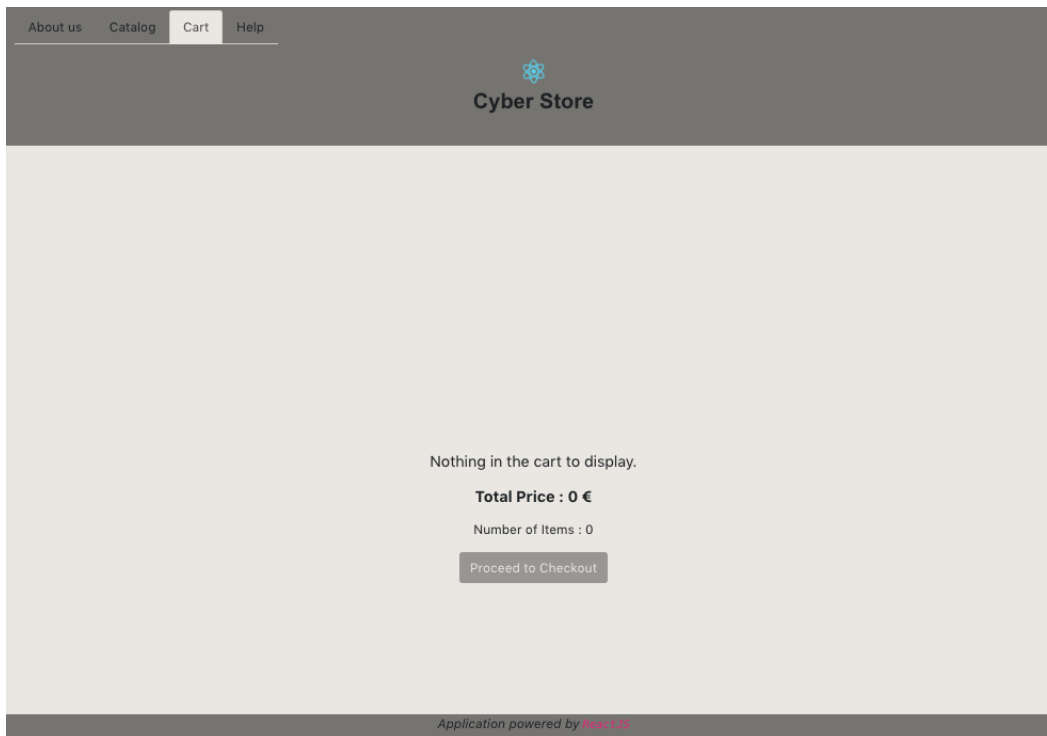
Une simulation d'application de e-commerce (côté front-end) a été implémentée. La page principale de l'application comporte 4 onglets, le premier étant ouvert par défaut et correspondant à l'onglet où se trouve le descriptif de l'application (fig [1]). Le deuxième onglet va correspondre à l'affichage du catalogue (fig [2]) et va faire appel au component correspondant `<CatalogPage/>`. Au niveau de l'affichage du catalogue, les items du magasin en ligne sont affichés sur un quadrillage par groupe de 3. Rattaché à chaque carte détaillant les items à vendre, un bouton permet d'ajouter l'item dans le panier client. Lorsque l'objet n'est plus en stock le bouton devient inactif et affiche "Soldout". Le troisième onglet va lui correspondre à l'affichage du panier client et va proposer de valider le panier et de procéder au paiement si et seulement si le panier n'est pas vide. Dans le cas où le panier est vide, le bouton permettant de procéder au paiement sera bloqué. Dans le cas contraire, si le panier est rempli, la liste des items que le client a sélectionnés sera affichée avec possibilité de les enlever. Le prix total du panier et le nombre total des items dans le panier sont aussi affichés (fig [3]). Le dernier onglet va quant à lui contenir un lien vers la page github du projet pour plus d'information.



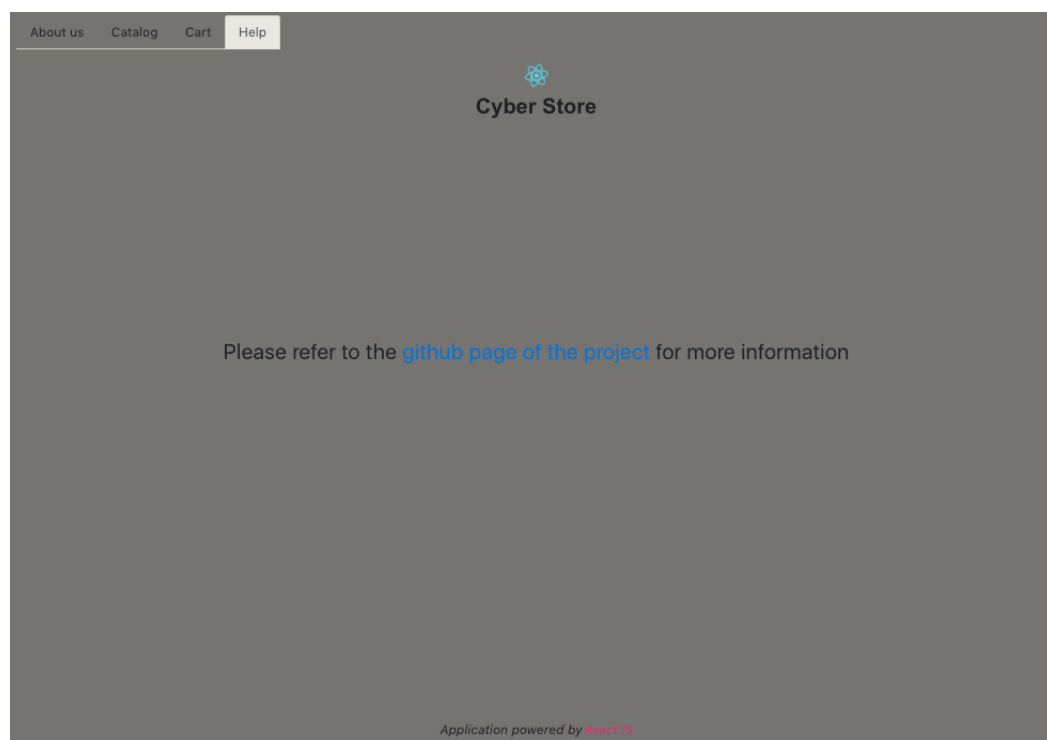
**Figure 1.** Page d'accueil de l'application web



**Figure 2.** Page Catalogue présentant les objets en ventes



**Figure 3.** Page Checkout présentant les objets mis dans le panier



**Figure 4.** Page d'aide de l'application

### 3 Perspectives et améliorations

---

Un outil fonctionnel au niveau front-end a été implémentée, cependant il pourrait être amélioré. En effet, l'interface utilisateur pourrait être améliorée et des fonctionnalités ajoutées comme par exemple l'affichage au niveau de l'onglet "Checkout" des articles en une seule fois pour chaque référence avec la quantité de chaque type d'articles. Au niveau des tests, ceux-ci pourraient être aussi améliorés. En effet, la couverture actuelle des tests du code est faible, d'autres tests pourraient être implémentés pour remédier à cela.