



# Rapport projet JEE: Site de E-commerce intitulé "Electronic Corner".

Auteurs: EL ALOUT Ismail, HALILOUA Othmane, LAHMOUZ Zakaria  
et KARMAOUI Oussama.

Mai 2023

## Table des matières

<b>1</b>	<b>Introduction :</b>	<b>3</b>
<b>2</b>	<b>Fonctionnalités :</b>	<b>3</b>
<b>3</b>	<b>Architecture-Conception :</b>	<b>3</b>
<b>4</b>	<b>Difficultés rencontrés :</b>	<b>3</b>
4.1	FrontEnd : . . . . .	3
4.2	BackEnd : . . . . .	4
4.3	Liaison BackEnd et FrontEnd : . . . . .	4
<b>5</b>	<b>Conclusion :</b>	<b>4</b>

## 1 Introduction :

Le rapport de projet présente la création d'un site web de e-commerce "Electronic Corner" axé sur la vente de produits électroniques en utilisant ReactJS. Ce rapport examine en détail le processus de développement, les fonctionnalités clés du site, ainsi que les défis et les solutions rencontrés tout au long du projet. Le site web de commerce électronique offre une interface utilisateur conviviale, et une navigation fluide, tout en offrant aux utilisateurs des produits électroniques à prix convenable à choisir. Le rapport met en évidence les technologies utilisées, les décisions de conception prises et les résultats obtenus lors de la réalisation de ce projet ambitieux.

## 2 Fonctionnalités :

Le site s'ouvre sur une publicité présentant le type de produits commercialisés par le site, puis propose à l'utilisateur une possibilité d'inscription pour devenir membre de la communauté "Electronic Corner", pour qu'il puisse se connecter et accéder aux produits proposés à la vente, après connexion l'utilisateur peut choisir les produits qu'il souhaite acheter, les ajouter au panier, ayant la possibilité de modifier leur multiplicités, après validation du panier, l'utilisateur en entre ces coordonnées personnelles et bancaires pour passer au paiement. Autres fonctionnalités ont été programmés à être ajoutés, pourtant la plage de temps ne nous a pas aidé.

## 3 Architecture-Conception :

Le site a été créé par la framework React, pour profiter d'une modularité efficace avec des performance optimisées, une réutilisabilité des composants pour le maintien du code. Le styling des pages a été fait via CSS, l'acheminement et la transition des pages ont été assurés par les Routes. La dynamique des pages et leur réactivité ont été mises en place en exploitant le hook intégré dans React *useEffect()*. Et ce pour tout ce qui concerne le FrontEnd.

Quant au BackEnd, on s'est basé d'abord sur la création d'un ensemble de classes Java, celles les plus prépondérantes étant :

- Personne : Représente l'utilisateur, comprend ses données personnelles.
- Produit : Représente un produit à vendre.
- Commande : Représente un ensemble de produits composant une commande.
- Panier : Représente le panier de la commande.
- Login : Représentant une session de connexion.

La liaison entre BackEnd et FrontEnd s'est fait par l'utilisation d'URL des méthodes du BackEnd par des appels Ajax. Les API Web a été implémenté par l'API JAX-RS (Web Services RESTful) et ce pour exploitent les possibilités du protocole HTTP pour permettre au systèmes d'information de communiquer et de s'échanger des services.

## 4 Difficultés rencontrés :

### 4.1 FrontEnd :

On a rencontré des difficultés concernant la modularité des styles des page, la solution envisagée initialement était d'utiliser des fichiers *.module.css*, pourtant, il existait toujours des confusions, des styles non désirés ont été mis en place d'autres page que celles auxquelles les fichier *.module.css* ont été destinés. D'autre part, la vue non parfaite des images sur le site est due au fait de la non normalité des tailles des images utilisées, il était pénible de trouver des images de même taille.

## 4.2 BackEnd :

Le site supposé contenir les produits à vendre à la disposition du client, on voulait faire deux types d'utilisateurs, un administrateur et un acheteur, seuls les administrateurs ont la possibilité d'ajouter les produits proposés à la vente, or, vu le manque du temps, on s'est contenté de ne faire que le type d'utilisateur *acheteur*, l'ajout des produits se fait manuellement par les administrateurs (les membre du groupe).

## 4.3 Liaison BackEnd et FrontEnd :

- Problème de lancement JBoss sur windows.
- Problème de configuration du fichier persistence.xml.
- Difficulté : Choix de l'API RESTful.
- Problème de Cors Policy sur les navigateurs récents.

## 5 Conclusion :

En conclusion, ce projet d'application web développée avec ReactJS, Java Entities, JPA, CSS et HTML a été une expérience enrichissante. L'utilisation de ReactJS a permis de créer une interface utilisateur réactive et dynamique, offrant une expérience utilisateur fluide. Les Java Entities et JPA ont facilité la gestion des données, assurant une intégrité et une cohérence optimales. En combinant CSS et HTML, nous avons pu créer une interface esthétique et conviviale. En résumé, ce projet d'application web basée sur ReactJS, Java Entities, JPA, CSS et HTML a permis de créer une solution moderne, attrayante et fonctionnelle, offrant une expérience utilisateur agréable. Il a également permis d'acquérir une expérience précieuse dans le développement d'applications web en utilisant ces technologies.