

L'entreprise **Tangerois**, spécialisée dans la vente d'électroménager, souhaite **moderniser sa plateforme e-commerce** pour offrir une meilleure expérience utilisateur et une interface plus claire, adaptée aux appareils mobiles et aux nouveaux standards du design UX.

L'objectif est de **concevoir et développer un MVP** (Minimum Viable Product) de la nouvelle version du site **Tangerois.ma** à l'aide de **React.js**, en suivant un processus complet de **conception UI/UX et UML**.

Objectifs

Ce projet vise à évaluer la capacité de l'étudiant à :

- Identifier les **besoins fonctionnels et utilisateurs**.
- Réaliser les **étapes de conception UI/UX** : zoning → wireframes → maquettes → prototype interactif.
- Modéliser le système à l'aide des **diagrammes UML**.
- Réaliser un **MVP fonctionnel avec React.js** simulant les principales fonctionnalités du site.

Enoncé du projet

Vous travaillez comme **UX/UI Designer et Développeur Front-end React** dans une agence web chargée de la refonte du site **Tangerois.ma**.

Votre mission est de concevoir et développer un **prototype fonctionnel (MVP)** d'un site e-commerce d'électroménager, comportant les fonctionnalités principales suivantes :

Partie 1 : Analyse et conception UX

1. Définir les personas

- Exemple :
 - **Fatima**, 32 ans, mère de famille à Casablanca, achète souvent en ligne pour éviter les déplacements.
 - **Youssef**, 27 ans, jeune salarié à Tanger, compare les prix avant d'acheter une TV.
 - **Administrateur**, responsable de la mise à jour des produits et promotions.

2. Identifier les besoins et parcours utilisateurs

- Parcours 1 : consulter la liste des produits.
- Parcours 2 : filtrer les produits par catégorie ou prix.
- Parcours 3 : ajouter un produit au panier.
- Parcours 4 : passer la commande.
- Parcours 5 : administrer les produits.

3. Créer les zonings

- Page d'accueil : bannière + catégories principales.
- Page liste produits : grille de cartes (image, prix, bouton "Voir détails").
- Page produit : détails + bouton "Ajouter au panier".

- Page panier : liste des articles ajoutés + bouton “Commander”.
4. **Créer les wireframes**
(Esquisses en noir et blanc des pages principales, avec hiérarchie visuelle claire).
 5. **Créer les maquettes (mockups)**
 - Utiliser les codes couleurs et style inspirés de Tangerois (rouge, blanc, gris).
 - Ajouter des éléments réels (images produits, logo, typographie).
 6. **Créer le prototype interactif (ex. sur Figma)**
 - Simuler la navigation entre les pages : “Accueil → Liste → Détails → Panier → Commander”.
 7. Appliquer au moins quatre lois de l’UX .

Partie 2 : Conception UML

a. Diagramme de cas d'utilisation

Acteurs :

- Client
- Administrateur

Cas principaux :

- Parcourir les produits
- Filtrer les produits
- Ajouter au panier
- Passer commande
- Gérer les produits (Admin)

b. Diagramme de séquence (scénario “Ajouter un produit au panier”)

- Le client clique sur “Ajouter au panier”.
- Le composant React envoie la requête au store (useContext avec useReducer)
- Le panier se met à jour et affiche la quantité totale.

c. Diagramme de classes

Classes principales :

- Product (id, nom, prix, catégorie, image, description)
- CartItem (productId, quantity)
- User (id, nom, rôle, email)
- Order (id, userId, items[], total, date)

Relations :

- Un User peut avoir plusieurs Orders.
- Un Order contient plusieurs CartItem.

- Un CartItem correspond à un seul Product.

d. Diagramme d'états et transitions

Exemple : État d'un produit

e. diagramme d'activité

À partir du parcours utilisateur suivant, réalisez le **diagramme d'activité UML** qui illustre les différentes étapes nécessaires pour **commander un produit** sur le site.

L'utilisateur (client) :

1. Accède à la **liste des produits**.
2. **Sélectionne** un produit.
3. Consulte la **fiche détaillée** du produit.
4. Clique sur "**Ajouter au panier**".
5. Accède à la **page panier** pour vérifier son contenu.
6. Clique sur "**Commander**".
7. **S'identifie** ou crée un **compte** s'il n'en a pas.
8. Choisit le **mode de livraison** et le **mode de paiement**.
9. Confirme la **commande**.
10. Reçoit une **confirmation de commande**.

Le diagramme doit représenter :

- Les **actions principales** de l'utilisateur et du système.
- Les **conditions** (ex. : "Compte existant ?") sous forme de **nœuds de décision**.
- Les **activités parallèles** éventuelles (ex. : vérification du paiement + envoi de mail).
- Le **début** et la **fin** du processus.

Partie 3 : MVP avec React.js

Créer un **MVP fonctionnel** avec les pages principales suivantes :

Page	Composants React	Fonctionnalité
/	<Home />	Affiche les catégories principales
/products	<ProductList />, <ProductCard />	Affiche la liste des produits
/products/:id	<ProductDetails />	Affiche les détails d'un produit
/cart	<Cart />	Gère le panier
/admin	<AdminPanel />	Gère les produits (ajouter, modifier, supprimer)

Technologies :

- React.js
- useContext avec useReducer(gestion de l'état global)
- React Router (navigation entre pages)
- Fake API ou JSON local pour simuler les produits

Partie 4 : Livrables attendus

1. Dossier UX complet (personas, zoning, wireframes, maquettes, prototype).
2. Dossier UML complet (use case, séquence, classe, état).
3. Code source du MVP (React.js).
4. Documentation du projet (guide d'installation + résumé des choix UX et UML).