

Nom : ALLALI

Prénom : Med Tadjedine

Section : Software enginner 3

Matricule : 222231707810

# Documentation du site E-commerce "Tech Bazaar"

## 1. HTML & CSS

---

Le site "Tech Bazaar" utilise une structure HTML générée dynamiquement par PHP. Cette approche permet de créer des pages web personnalisées selon les actions de l'utilisateur et les données de la base.

### Structure HTML

- Le HTML est généré par des fichiers PHP qui incluent des éléments communs comme l'en-tête (`header.php`) et le pied de page (`footer.php`)
- La navigation change selon que l'utilisateur est connecté (`navconnected.php`) ou non (`nav.php`)
- Les pages principales incluent l'accueil, les catégories de produits, les détails des produits, le panier et le processus de paiement

### Styles CSS

- Les styles sont organisés dans plusieurs fichiers CSS dans le dossier `src/css`
- Le fichier principal `style.css` définit l'apparence générale du site
- Des styles spécifiques sont utilisés pour les animations et les effets visuels
- Le site utilise la police Roboto pour une meilleure lisibilité
- La mise en page est responsive pour s'adapter aux différentes tailles d'écran

### Éléments visuels

- Le site utilise des icônes pour représenter les catégories de produits

- Des images de produits sont stockées dans le dossier `src/img`
- Des effets d'animation sont appliqués aux éléments interactifs pour améliorer l'expérience utilisateur

## 2. JavaScript

---

Le JavaScript est utilisé pour rendre le site interactif et améliorer l'expérience utilisateur.

### Fonctionnalités principales

- Validation des formulaires (inscription, connexion, paiement)
- Mise à jour dynamique du panier sans rechargement de page
- Effets visuels et animations lors des interactions
- Autocomplétion dans la barre de recherche
- Défilement fluide entre les sections de la page

### Fichiers JavaScript

- `main.js` : contient les fonctions personnalisées pour le site
- `jquery-2.1.4.js` : utilisé pour simplifier les manipulations du DOM
- `nprogress.js` : affiche une barre de progression lors du chargement
- `smoothscroll.js` : permet un défilement fluide
- `typed.js` : crée des effets de texte tapé à la machine
- `wow.min.js` : gère les animations au défilement

## 3. PHP

---

PHP est le langage principal du site, gérant à la fois l'affichage et la logique métier.

### Structure du code

- Séparation en modules : partie publique et administration
- Utilisation de sessions pour gérer l'état de l'utilisateur
- Inclusion de fichiers communs pour éviter la répétition de code

## Fonctionnalités principales

- Gestion des utilisateurs (inscription, connexion, profil)
- Affichage des produits par catégorie
- Système de panier d'achat
- Processus de commande et paiement
- Génération de factures PDF
- Panneau d'administration pour gérer les produits et les commandes

## Sécurité

- Validation des données utilisateur
- Protection contre les injections SQL
- Gestion des sessions sécurisée

## 4. Base de données (MySQL)

---

La base de données stocke toutes les informations nécessaires au fonctionnement du site.

### Structure de la base

- Connexion établie via le fichier `db.php`
- Base nommée `ecommerce_db` avec plusieurs tables relationnelles

### Tables principales

- `users` (id int, email varchar(100), firstname varchar(1000), lastname varchar(1000), password varchar(1000), address varchar(1000), city varchar(1000), country varchar(1000), role varchar(1000))
- `category` (id int, name varchar(1000), icon varchar(1000))
- `product` (id int, **id\_category** int, name varchar(1000), description varchar(1000), price int, **id\_picture** int, thumbnail varchar(1000), promo varchar(1000), stock int)
- `command` (id int, **id\_product** int, quantity int, dat timestamp, statut varchar(1000), **id\_user** int)
- `details_command` (id int, product varchar(1000), quantity int, price int, **id\_command** int, **id\_user** int, user varchar(1000), address varchar(1000), country varchar(1000), city varchar(1000), statut varchar(1000))
- `pictures` (id int, picture varchar(1000), **id\_product** int\*)

- cancelled\_orders (id int, **id\_command** int, **id\_product** int, quantity int, **id\_user** int\*, cancel\_date timestamp, reason varchar(1000))

## Fonctionnalités de la base et opérations associées

1. **Affichage des détails de commande et montant total**
2. Opération : Procédure stockée GetOrderDetails
3. Description : Affiche tous les détails d'une commande et calcule le montant total
4. **Finalisation de commande et vidage du panier**
5. Opération : Procédure stockée FinalizeOrder
6. Description : Finalise une commande, transfère les articles du panier vers les commandes payées et met à jour le stock
7. **Historique des commandes d'un client**
8. Opération : Procédure stockée GetOrderHistory
9. Description : Affiche l'historique complet des commandes d'un client
10. **Mise à jour du stock après validation de commande**
11. Opération : Trigger update\_stock\_after\_order
12. Description : Décrémente automatiquement le stock lorsqu'une commande passe au statut "payé"
13. **Vérification de disponibilité du stock**
14. Opération : Trigger check\_stock\_before\_order
15. Description : Empêche la création d'une commande si la quantité demandée dépasse le stock disponible
16. **Restauration du stock après annulation de commande**
17. Opération : Trigger restore\_stock\_after\_cancel
18. Description : Réincrémente le stock lorsqu'une commande est annulée
19. **Suivi des commandes annulées**
20. Opération : Trigger track\_cancelled\_orders
21. Description : Enregistre les informations des commandes annulées dans une table dédiée pour analyse