

Fil Rouge CDA

III. Développements

Pour développer son système de vente, l'entreprise souhaiterait mettre en place un site web (e-commerce) accessible aux particuliers et aux professionnels. Cette solution doit offrir les services suivants:

- Front-Office (partie publique du site)
 - Consultation du catalogue / Ajout des produits dans le panier
 - Inscription d'un nouvel utilisateur sur le site (pour les particuliers)
 - Connexion/Deconnexion d'un utilisateur pour accéder à son profil
 - Validation du panier pour créer une nouvelle commande
- Back-Office (réservé à l'administrateur)
 - Ajout/Suppression de produit dans le panier
 - Visualisation des anciennes commandes

La liste de produits doit s'afficher avec un système de pagination.

Vous devez prévoir un module de recherche de produits.

Le site doit bien évidemment être lisible sur les principaux navigateurs et il doit s'adapter aux navigateurs mobiles.

A. Maquetter une application

1. Construire la maquette de l'application

Avec un outil de maquettage (mockflow.com ou Figma [tuto video](#)), réalisez la maquette de l'application Web. Cette maquette doit comporter :

- La fenêtre de démarrage de l'application affichant la liste des catégories
- Une page affichant la liste des sous catégories d'une catégorie
- Une page affichant la liste des produits d'une sous catégorie
- Une page affichant le contenu du panier
- Une page d'inscription
- Une page permettant de se connecter

Représentez le graphe de navigation de votre application.

Ce graphe doit mettre en évidence les différents écrans de votre application ainsi que les liens permettant de passer d'un écran à un autre.

Vous pouvez vous inspirer de [cet exemple](#).

Pour l'application Desktop, la maquette doit comporter:

- L'écran d'accueil incluant le menu
- Une interface permettant de naviguer dans la catalogue (Catégorie/Sous catégorie)
- Une interface permettant de gérer le CRUD sur le catalogue
- Une interface permettant de sélectionner un client et d'afficher les commandes
- Le tableau de bord

Pour l'application Mobile, la maquette doit comporter:

- L'écran de démarrage de l'application
- Une page affichant la liste des sous catégories d'une catégorie
- Une page affichant la liste des produits d'une sous catégorie

B. Développer une interface de type desktop et des composants d'accès aux données

C. Développer la partie Front-End

a) Développer des pages web statiques (HTML/CSS)

Réaliser un page d'accueil pour votre projet. Vous devez réaliser l'intégration HTML/CSS de votre page d'accueil à partir des éléments qui vous sont fournis dans la charte graphique.

Le site est divisé en deux parties.

- Font Office : contient la partie publique du site (dont la page d'accueil) et un accès à la liste de produits et accès au formulaire d'inscription. Vous devez intégrer au mieux les éléments de la charte graphique.
- Back Office : c'est la partie privée du site, réservée à l'administrateur, elle permet de gérer les produits, les commandes ...

Reprenez les différents écrans de votre maquette et réalisez l'intégration HTML/CSS. Le résultat de votre travail doit comporter au moins :

- La fenêtre de démarrage de l'application affichant la liste des catégories
- Une page affichant la liste des sous catégories d'une catégorie
- Une page affichant la liste des produits d'une sous-catégorie
- Une page affichant le contenu du panier
- Une page d'inscription
- Une page permettant de se connecter

b) Intégrer des scripts clients (Javascript)

Votre application WEB doit comporter un formulaire d'inscription pour le client.

Vous devez empêcher l'utilisateur d'envoyer des informations erronées et lui indiquer les erreurs.

D. Développer la partie Back-End

a) Développer des composants web d'accès aux données

Pour la partie Front Office:

Votre site doit permettre à l'utilisateur de créer une commande (ajout des produits au panier, inscription, connexion, validation du panier et enregistrement de la commande)

Pour la partie Back Office:

Vous devez mettre en œuvre la gestion CRUD sur une table de votre choix. Ces pages devront être accessibles à partir de votre menu d'accueil.

Votre interface doit permettre d'afficher la liste des éléments, l'ajout, la modification et la suppression.

Vous devez utiliser une architecture MVC pour réaliser l'ensemble de ce travail.

2. Publier l'application

Publier le résultat de votre travail sur l'espace de publication qui vous a été attribué. Le site publié doit s'exécuter sans erreur.

a) C# / composants graphiques

Pour son usage interne, l'entreprise souhaite s'équiper d'un logiciel de gestion commerciale. Ce logiciel sera déployé sur les postes informatiques de l'entreprise

1. Un module de gestion des commandes réservé au service commercial :
 - Connaître l'état de la commande (saisie, annulée, en préparation, expédiée, facturée, retard de paiement, soldée)
2. Un module de gestion des produits réservé au service de gestion des produits doit permettre :
 - D'ajouter des produits
 - D'en supprimer
 - D'en modifier les caractéristiques (libellé, caractéristique, tarif)
3. Un tableau de bord de gestion permettra d'afficher des indicateurs de performance
 - Chiffre d'affaires mois par mois pour une année sélectionnée
 - Chiffre d'affaires généré pour un fournisseur
 - TOP 10 des produits les plus commandés pour une année sélectionnée (référence et nom du produit, quantité commandée, fournisseur)
 - TOP 10 des produits les plus rémunérateurs pour une année sélectionnée (référence et nom du produit, marge, fournisseur)
 - Top 10 des clients en nombre de commandes ou chiffre d'affaires
 - Répartition du chiffre d'affaires par type de client
 - Nombre de commandes en cours de livraison.

À partir de la maquette réalisée précédemment, concevez (au minimum) :

- L'écran d'accueil
- L'interface permettant de sélectionner un client et d'afficher la liste des commandes.
- L'interface permettant de gérer le CRUD sur la table des produits.
- L'interface permettant d'afficher le tableau de bord.

Les zones saisies doivent être contrôlées et les erreurs de saisie gérées.

En utilisant le pattern DAO, créez des composants d'accès aux données vous permettant de fournir des données à vos formulaires. Une première version utilisera une couche d'accès aux données C#. Une deuxième version utilisera une couche d'accès via l'API que vous développerez plus tard.

Votre couche d'accès aux données doit être testée par le framework de tests unitaires.