

Module Développement Web

Projet BookStore

Travail Pratique 1
version ASP.NET Core 6

Novembre 2022

1 Objectifs

Le but de ce travail pratique est de vous familiariser avec HTML5.

Plus particulièrement, vous allez élaborer la structure de pages web tout en respectant la structure sémantique apportée par HTML5. De plus, vous allez mettre en place l'interaction entre les différentes pages à l'aide des hyperliens.

2 Introduction

Tout au long du module, vous aurez à réaliser un site web transactionnel permettant l'achat de produits électronique. Pour y arriver, les différents travaux pratiques à réaliser dans le cadre du cours intégreront plusieurs notions pour obtenir un site web complètement fonctionnel à la fin de la session. Ainsi, le code que vous écrierez sera réutilisé lors des travaux pratiques suivants. Si vous n'êtes pas satisfait de votre remise ou que vous souhaitez partir du bon pied, vous pouvez utiliser le corrigé qui sera fourni au début du prochain travail pratique.

Un site web d'achat en ligne comporte plusieurs composants classiques. En effet, ce genre de site comporte systématiquement une page pour afficher les produits disponibles pour une certaine catégorie, une page qui détaille un produit en particulier, une page pour afficher les produits qui ont été ajoutés au panier d'achats, ainsi qu'une page pour commander les produits sélectionnés. Le site web de commerce en ligne [Amazon](#) intègre d'ailleurs toutes les pages qui ont été décrites. Par ailleurs, ce genre de site intègre également une partie administrative (CMS) qui permet de gérer les produits à afficher et les commandes reçues.

3 Travail à réaliser

Pour ce travail pratique, vous devez élaborer la structure des pages HTML5 qui composeront votre site web transactionnel. La structure attendue pour une page est d'ailleurs illustrée à la figure 1. Cette structure devra être définie à l'aide des balises sémantiques de HTML5, soient les balises "header", "nav", "main", et "footer". Les sous-sections qui suivent décrivent les différentes pages à réaliser.

Il est à noter que seule la structure HTML du site sera évaluée. Les fichiers HTML ne doivent contenir aucun élément de mise en page (classe CSS, etc.) ni de logique (JS / C#). Tâchez donc de vous assurer que les balises correspondent à la fonction de leur contenu.

Avant de débuter, assurez-vous d'avoir récupéré l'archive associée à ce travail pratique sur le partage "[fichiers tp1](#)". Celle-ci contient les fichiers HTML à compléter ainsi que des images à inclure dans les pages.

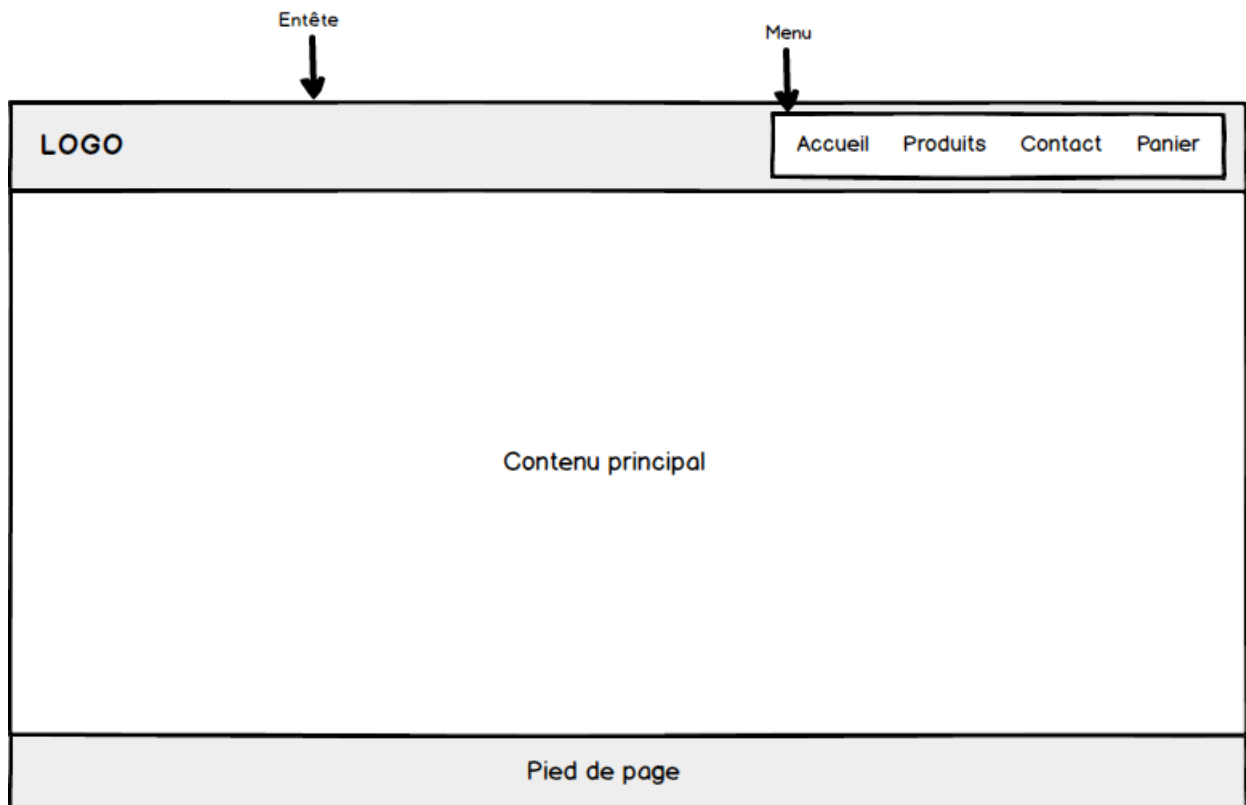


FIGURE 1 – Maquette d'une page web

3.1 Entête

Chacune des pages du site web doit contenir un entête. L'entête doit être composé du logo du site (`/images/logo.svg`) qui comporte un lien vers la page d'accueil (`Index.cshtml`) et d'un menu de navigation qui inclut les éléments suivants :

- Accueil (`Index.cshtml`);
- Produits (`Products.cshtml`);
- Contact (`Contact.cshtml`);
- Panier (`ShoppingCart.cshtml`).

3.2 Pied de page

Tout comme pour l'entête, chacune des pages du site web doit contenir un pied de page. Ce dernier doit contenir les informations du site, soit les nom du site et autres.

3.3 Page d'accueil (Index.cshtml)

La page d'accueil est la page d'entrée de votre site web. Celle-ci comprend un titre, un petit paragraphe décrivant le site ainsi qu'un lien vers la page qui liste les produits disponibles (Products.cshtml). Également, cette page doit inclure une image (/images/home.png).

3.4 Page des produits (Products.cshtml)

La page des produits affiche les différents produits disponibles dans l'inventaire de la boutique en ligne. Cette page est composée d'une barre latérale (sidebar) ainsi que d'un panneau principal. Par souci de clarté, la figure 2 illustre les différents éléments de la page qui sont discutés ci-dessous. Il est à noter que cette figure ne représente pas le rendu que vous allez obtenir, puisque vous n'avez aucune feuille de style associée avec votre page.

3.4.1 Barre latérale

La barre latérale contient deux groupes de boutons (voir ce [lien](#)). Un des groupes de boutons permet de sélectionner une catégorie de produits à afficher, alors que l'autre permet de trier les produits selon un ordre particulier. Le groupe de boutons permettant de sélectionner une catégorie doit posséder l'identifiant "product-categories" (id="product-categories"). De plus, celui permettant de trier les produits selon un critère particulier doit posséder l'identifiant "product-criteria" (id="product-criteria").

Les catégories de produits sont les suivantes :

- Appareils photo;
- Consoles;
- Écrans;
- Ordinateurs;
- Tous les produits (valeur par défaut).

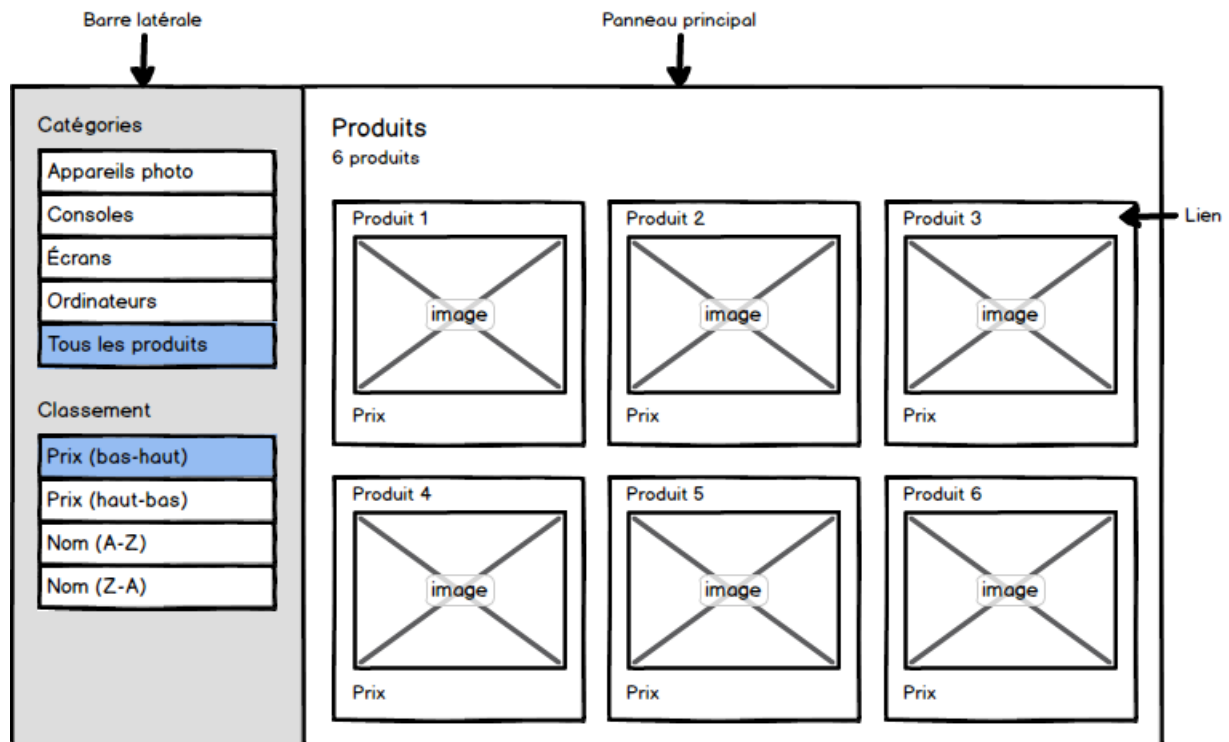


FIGURE 2 – Maquette de la page des produits

Les critères de tri des produits sont quant à eux les suivants :

- Prix (bas-haut) (valeur par défaut) ;
- Prix (haut-bas) ;
- Nom (A-Z) ;
- Nom (Z-A).

3.4.2 Panneau principal

Le panneau principal de la page permet l’affichage des produits. Chacun des produits listés sur cette page doit afficher son nom (doit être un élément de titre), son image associée ainsi que son prix. Également, les produits listés doivent être cliquables, c’est-à-dire qu’ils doivent contenir un lien vers la page d’un produit (`product.html`). Ainsi, le lien doit englober le titre, l’image et le prix de chacun des produits. De plus, la liste des produits doit posséder l’identifiant `"products-list"` (`id="products-list"`) ; assurez-vous donc de définir un élément qui permet de regrouper les différents produits listés.

En plus de lister les produits, un titre doit être présent dans le panneau principal. Également, un élément doit être inclus afin d'indiquer le nombre de produits actuellement affichés sur la page (p. ex. "6 produits").

⚠ Notez bien

Afin de vous simplifier la tâche, une liste de produits vous est fournie dans le fichier `products.json` (voir le dossier `/data`). Pour ce travail, il vous est demandé de lister uniquement six produits sur cette page. Utilisez également les images des produits qui vous sont fournies dans le dossier `/images`.

3.5 Page d'un produit (`Product.cshtml`)

Cette page affiche un produit en particulier. En ce sens, le nom (titre), l'image, la description (paragraphe), les caractéristiques (liste non ordonnée) ainsi que le prix sont affichés pour un produit spécifié.

Cette page contient également un formulaire permettant d'ajouter le produit courant au panier d'achats. Ce formulaire contient un champ pour spécifier la quantité de produits à ajouter au panier, ayant comme valeur par défaut un et ayant comme contrainte un nombre entier positif supérieur à zéro, ainsi qu'un bouton "Ajouter" pour réaliser l'action.

⚠ Notez bien

Afin de vous simplifier la tâche, choisissez un produit en particulier dans la liste de produits qui vous est fournie (`data/products.json`) pour réaliser la page web. Ainsi, il n'est pas nécessaire de dupliquer cette page pour chacun des produits disponibles pour ce travail.

3.6 Page de contact (`Contact.cshtml`)

Cette page affiche les différentes informations nécessaires pour contacter l'entreprise du site web. Ainsi, cette page doit contenir un titre et des liens vers une adresse courriel et un numéro de téléphone (voir l'attribut [href](#) et le tag-helper [asp-action](#)). Également, cette page doit afficher l'adresse de l'entreprise. Il est à noter que ces informations peuvent être fictives.

3.7 Page du panier d'achats (ShoppingCart.cshtml)

Cette page permet d'afficher les produits qui sont actuellement dans le panier d'achats.

Un tableau affiche les éléments contenus dans le panier. Ce tableau doit comporter cinq colonnes et chaque ligne doit correspondre à un élément du panier (sauf la première qui correspond à l'entête). Chaque ligne, en excluant la première, affiche le nom, le prix unitaire, la quantité ainsi que le prix total pour un des produits se trouvant dans le panier. De plus, le nom d'un produit dans le tableau comporte un lien vers la page d'un produit (Product.cshtml). Également, chacune des lignes du tableau contient plusieurs boutons pour interagir avec un item du panier. En ce sens, un premier bouton (« × ») permettra de supprimer l'item qui lui est associé, alors que deux autres boutons (« - » et « + ») permettront de contrôler la quantité du produit dans le panier. Par souci de clarté, la figure 3 montre le rendu de la page attendu. Il est à noter que le rendu peut différer en réalité puisque la page n'utilise pas de feuille de style.

En plus du tableau, cette page dispose d'un titre et d'un élément permettant d'afficher le total (en frs) des items contenus dans le panier. De plus, cette page possède un lien appelé « Commander » qui pointe vers la page de commande (Order.cshtml). La page possède également un bouton appelé « Vider le panier » qui permettra de vider les produits du panier.

| Panier | | | | |
|-------------------------------------|---------------------------|---------------|----------|----------|
| | Nom | Prix unitaire | Quantité | Prix |
| <input checked="" type="checkbox"/> | Produit 1 | 2.00frs | - 1 + | 2.00frs |
| <input checked="" type="checkbox"/> | Produit 2 | 4.00frs | - 2 + | 8.00frs |
| <input checked="" type="checkbox"/> | Produit 3 | 5.00frs | - 2 + | 10.00frs |

Total: 20.00frs

[Commander](#)

FIGURE 3 – Maquette de la page du panier d'achats

⚠ Notez bien

Puisque ce travail ne met en place aucune logique pour la gestion du panier d'achats, il vous est demandé de simuler l'affichage de trois produits qui ont été ajoutés au panier. Vous pouvez ajouter les produits de votre choix à partir de la liste des produits (voir data/products.json).

3.8 Page de commande (Order.cshtml)

Cette page contient un titre et un formulaire pour commander les produits se trouvant dans le panier. Les champs du formulaire doivent être validés avant d'être soumis à l'aide des attributs de HTML5 (p. ex. `required`). De plus, les champs du formulaire doivent utiliser les identifiants (`id="..."`) qui sont spécifiés ci-dessous et doivent posséder une étiquette (voir l'élément [label](#)).

La première section du formulaire, nommée « Contact », demande des informations pour être en mesure de contacter le client. En ce sens, cette partie doit contenir les champs suivants et ceux-ci sont obligatoires pour la soumission du formulaire :

- Prénom (`id="first-name"`);
- Nom (`id="last-name"`);
- Adresse courriel (`id="email"`);
- Numéro de téléphone (`id="phone"`).

La deuxième section du formulaire, appelée « Paiement », demande les informations nécessaires pour effectuer la facturation. Ainsi, cette partie doit posséder les champs suivants et ces mêmes champs sont obligatoires pour pouvoir soumettre le formulaire :

- Numéro de carte de crédit (`id="credit-card"`);
- Date d'expiration de la carte (`id="credit-card-expiry"`);

Finalement, le formulaire possède un bouton appelé « Payer » qui permet de soumettre les informations. Lorsque le formulaire est envoyé et est complet, la page de confirmation (`confirmation.html`) est affichée.

3.9 Page de confirmation (Confirmation.cshtml)

Cette page est affichée une fois qu'une commande est complétée. Celle-ci comprend un titre et un message de confirmation afin d'indiquer à l'utilisateur que sa commande est complétée.

Conseils pour la réalisation du travail pratique

1. Assurez-vous d'ajouter des titres aux sections importantes des pages.
2. Jetez un coup d'oeil aux différentes [balises sémantiques](#) de HTML5 afin de vous assurer que les éléments que vous utilisez soient bien adaptés.
3. Assurez-vous que la sémantique de votre page est correcte en la faisant valider par l'[outil de W3C](#).
4. Assurez-vous que votre plan de document (outline) soit valide. Voir ce [site web](#).
5. Effectuez une validation de vos formulaires grâce aux attributs de HTML5 (voir la section « Input Restrictions » de la page [HTML Input Types](#)).
6. Soyez consistant dans vos conventions de codage (voir [guide de codage de Mark Otto](#)).

4 Remise

Voici les consignes à suivre pour la remise de ce travail pratique ..