Remerciements

Résume du projet

Mon projet consiste à développer un site web pour Élégance Couture, une marque de mode masculine qui met l'accent sur l'élégance intemporelle, la qualité exceptionnelle et le service client attentif. Dans le cadre de mon ECF, j'ai entrepris la mission passionnante de créer un site de e-commerce de vente de prêt-à-porter, de A à Z. Pour commencer, j'ai entamé mes réflexions en concevant des diagrammes pour planifier la structure de la base de données, suivi d'une étape de maquettage (wireframe) afin de visualiser le design et l'agencement de l'interface utilisateur. Je souhaite créer une plateforme en ligne qui incarne pleinement les valeurs de ma marque, offrant une expérience d'achat raffinée et esthétique pour mes clients.

Sur mon site web, les utilisateurs seront accueillis par une page d'accueil mettant en avant les produits "favoris", suivie d'une page E-shop permettant de naviguer à travers les différentes collections. Une page concept partagera ma vision et mes valeurs, tandis qu'une page boutique présentera les emplacements physiques de mes magasins pour offrir une expérience omnicanale. (Assurer une expérience omnicanale signifie offrir une interaction fluide et cohérente avec l'entreprise, quel que soit le canal utilisé par le client (boutique, site web….).

Pour la réalisation de l'interface utilisateur, j'ai opté pour une approche statique et responsive en utilisant HTML, CSS et Bootstrap. Mon objectif était de créer une interface esthétique et intuitive, qui s'adapte de manière transparente à tous les types d'appareils et de tailles d'écran. En outre, j'ai intégré des éléments dynamiques grâce à JavaScript, offrant ainsi une expérience interactive et engageante aux utilisateurs. Les animations et les transitions ajoutent une touche de dynamisme à l'interface, améliorant ainsi l'expérience globale de l'utilisateur.

Le back-end de mon site permettra la présentation détaillée des produits, la gestion des menus, le traitement des commandes et le processus d'inscription des utilisateurs. Je prévois également d'ajouter des fonctionnalités supplémentaires telles que la recherche et le filtrage des produits, la gestion des commentaires et avis des clients, ainsi que l'intégration d'une passerelle de paiement en ligne.

Sur le plan technique, j'ai structuré mon application selon le modèle MVC avec PHP, assurant ainsi une séparation claire entre la logique métier, la présentation et la gestion des données. Cette approche garantit un code bien organisé et maintenable, facilitant ainsi le développement et la maintenance de l'application à long terme. Pour la gestion des données, j'ai opté pour MySQL comme système de gestion de base de données, offrant une grande fiabilité et une bonne performance. En ce qui concerne l'hébergement, j'utilise Hostinger comme service d'hébergement en ligne, offrant une plateforme stable et fiable pour héberger mon application web.

En résumé, ce projet a été une expérience extrêmement enrichissante, me permettant de mettre en pratique mes compétences en conception, développement et déploiement d'applications web. De la conception initiale à la réalisation finale, j'ai été impliqué dans toutes les étapes du processus de développement, ce qui m'a permis d'acquérir une compréhension approfondie des défis et des opportunités associés à la création d'une application web complète. Je suis fier du résultat final et convaincu que cette expérience m'aidera à progresser dans ma carrière de développeur web.

Summary of the project:(Résumé du projet en anglais )

My project involves developing a website for Élégance Couture, a men's fashion brand focusing on timeless elegance, exceptional quality, and attentive customer service. As part of my ECF, I undertook the exciting mission of creating a full-fledged e-commerce website from scratch. I began by conceptualizing the database structure through diagrams and wireframing to visualize the user interface design. My goal is to create an online platform that fully embodies the values of my brand, offering a refined and aesthetic shopping experience for customers.

The functionality of my application includes allowing users to browse product collections, place orders, add items to their cart, and authenticate their accounts. Additionally, I've considered another type of user, referred to as the "host," for whom I've implemented a content management solution to autonomously manage site data. This approach ensures an optimal user experience, providing a responsive and user-friendly platform for customers while equipping the administrator with tools to efficiently manage site content.

On the website, users will be greeted with a homepage showcasing featured products, followed by an E-shop page for navigating through various collections. A concept page will share my vision and values, while a boutique page will showcase physical store locations to offer an omnichannel experience.

For the user interface, I utilized HTML, CSS, and Bootstrap for a static and responsive approach, aiming to create an aesthetic and intuitive interface adaptable to all devices. Additionally, I integrated dynamic elements with JavaScript to enhance interactivity and engagement. Animations and transitions were added to inject dynamism into the interface, improving the overall user experience.

The backend of my site will handle detailed product presentations, menu management, order processing, and user registration. I plan to incorporate additional features such as product search and filtering, customer review management, and online payment integration.

Technically, I structured my application using the MVC model with PHP, ensuring a clear separation of business logic, presentation, and data management. MySQL was chosen as the database management system for its reliability and performance. As for hosting, I utilize Hostinger for stable and reliable hosting of my web application.

In summary, this project has been an incredibly enriching experience, allowing me to apply my skills in web design, development, and deployment. From initial conception to final implementation, I've been involved in every stage of development, gaining a deep understanding of the challenges and opportunities associated with creating a comprehensive web application. I'm proud of the final result and confident that this experience will contribute to my career progression as a web developer.

Compétences du référentiel couvertes par le projet

**Activités types**

1. **Développer la partie front-end d’une application web ou web mobile sécurisée** :
   * Maquêter des interfaces utilisateur web ou web mobile.
   * Réaliser les interfaces utilisateur statiques pour un site ou une application web.
   * Développer la partie dynamique d’un site ou une application web en intégrant les recommandations de sécurité.
2. **Développer la partie back-end d’une application web ou web mobile sécurisée** :
   * Installer et configurer son environnement de travail en fonction du projet web.
   * Mettre en place une base de données relationnelle et/ou NoSQL.
   * Développer des composants d’accès aux données SQL et NoSQL.
   * Intégrer des composants dans une page web mobile.

Le cahier des charges

**1.Contexte**

Le projet vise à créer un site e-commerce pour répondre aux besoins croissants des consommateurs en ligne. Avec l'essor du commerce électronique, il est essentiel de proposer une plateforme conviviale, sécurisée et attrayante pour attirer et fidéliser les clients.

Le futur site web d'Élégance Couture incarnerait pleinement les valeurs de ses créateurs : l'élégance intemporelle, la qualité exceptionnelle et le service client attentif. Chaque aspect de l'interface serait soigneusement conçu pour offrir une expérience d'achat en ligne raffinée et esthétique, mettant en valeur chaque produit avec des descriptions détaillées et des images de haute qualité. Nous utiliserions également notre plateforme pour partager notre passion pour la mode masculine et pour maintenir une communication transparente et respectueuse avec nos clients.

En somme, notre site web serait l'expression numérique de l'identité et des valeurs d'Élégance Couture, offrant une expérience d'achat en ligne empreinte d'élégance et de qualité supérieure.

**Objectif**

L'objectif principal du projet est de développer un site e-commerce fonctionnel et intuitif qui offre une expérience d'achat agréable et sécurisée. Le site devrait permettre aux utilisateurs de parcourir facilement les produits, de passer des commandes en toute sécurité et de bénéficier d'un service client efficace.

**Cibles**  
Le site vise principalement les hommes et les jeunes hommes, avec une concentration particulière sur ceux qui sont sensibles à la mode et qui ont déjà un intérêt pour cet univers. Cependant, il s'adresse également à ceux qui souhaitent découvrir le monde de la mode masculine de manière accessible et attrayante. En termes de géographie, bien que le site soit basé dans la région parisienne, il vise à toucher les hommes de toute la France. L'objectif est d'offrir une gamme variée de produits qui répondent aux différents besoins et préférences des consommateurs masculins, en proposant des vêtements et des accessoires de qualité supérieure tout en reflétant l'élégance intemporelle et le style distinctif d'Élégance Couture.

**Analyse de la concurrence**

L'objectif est de se démarquer des concurrents en offrant une expérience utilisateur supérieure. Une analyse de la concurrence sera réalisée pour étudier les sites e-commerce similaires déjà présents sur le marché. Cela permettra d'identifier les forces et les faiblesses de la concurrence, ainsi que les opportunités à exploiter pour se démarquer. Les principaux domaines d'analyse incluent la convivialité, la conception visuelle, la clarté des informations, la facilité de navigation et les fonctionnalités offertes.

**Planification**  
En qualité de développeur unique sur le projet, la planification sera alignée sur ma disponibilité et mon rythme de travail.

**Spécifications techniques**

* Le site sera développé en utilisant les langages de programmation HTML, CSS, JavaScript pour le front-end et PHP pour le back-end.
* La base de données sera gérée à l'aide de MySQL pour stocker les informations sur les produits, les utilisateurs et les commandes.
* Le site sera responsive, offrant une expérience utilisateur optimale sur les appareils mobiles, les tablettes et les ordinateurs de bureau.
* Des fonctionnalités telles que l’authentification et le panier d'achat et le suivi des commandes seront intégrées.

**Critères de réussite**

1. Lancement réussi du site dans les délais impartis : Le site doit être lancé conformément au calendrier établi, sans retards significatifs.
2. Augmentation du temps passé par les utilisateurs sur le site : Ce qui démontrera ainsi l'engagement et l'intérêt pour le contenu proposé. Amélioration de la convivialité et de la pertinence du contenu, optimisation des performances du site pour réduire les temps de chargement, création de contenus engageants et attrayants.
3. Retours positifs des utilisateurs sur l'expérience d'achat: Les utilisateurs doivent exprimer leur satisfaction à travers des retours positifs sur leur expérience d'achat, y compris la facilité de navigation, la clarté des informations produit, la rapidité du processus de commande, et la qualité des produits reçus.
4. Positionnement concurrentiel sur le marché : L'objectif est de garantir que notre site e-commerce se positionne de manière compétitive par rapport à d'autres acteurs du marché. Cela implique d'analyser les fonctionnalités et les services offerts par nos concurrents, tout en identifiant les domaines où nous pouvons nous démarquer, que ce soit par l'innovation, la qualité du service ou une proposition de valeur unique. En comprenant le paysage concurrentiel, nous pourrons adapter notre stratégie pour rester compétitifs et attractifs pour nos clients."
5. Référencement : Améliorer le classement du site dans les résultats des moteurs de recherche pour augmenter la visibilité et le trafic organique. Amélioration de la structure du site et de la qualité des liens entrants, utilisation de bonnes pratiques en matière de référencement technique.

**Contraintes & Risques**

1. **Limitations de temps :** En tant que développeur unique sur mon projet, je suis conscient que ma disponibilité peut être affectée par d'autres engagements professionnels ou personnels, ce qui pourrait entraîner des retards dans le développement et le lancement du site.
2. **Compétences techniques :** J'admets que mes compétences techniques peuvent être limitées dans certains domaines en tant que développeur unique. Cela peut poser des défis lors du développement de fonctionnalités complexes ou de l'intégration de certaines technologies.
3. **Dépendance des ressources externes :** J'ai conscience que si mon projet nécessite l'utilisation de ressources externes telles que des bibliothèques de code, des plugins ou des API, je pourrais rencontrer des problèmes liés à leur disponibilité, leur fiabilité ou leur compatibilité.
4. **Stabilité de l'hébergeur gratuit :** J'ai choisi d'héberger mon site sur une plateforme gratuite, mais je suis conscient des risques potentiels en termes de fiabilité, de performances et de sécurité. Je dois surveiller attentivement les performances du site et être prêt à migrer vers un hébergement payant si nécessaire.
5. **Sécurité :** Je reconnais que la sécurité des données des utilisateurs et des transactions en ligne est une préoccupation majeure dans le commerce électronique. Je dois mettre en place des mesures de sécurité appropriées pour protéger mon site contre les cyberattaques, les failles de sécurité et les violations de données.

En identifiant et en évaluant ces contraintes et risques potentiels dès le début de mon projet, je peux mettre en place des mesures d'atténuation appropriées pour minimiser leur impact sur la réussite de mon projet.

**Fonctionnalités**

**2. Front-end**

L’application est composée de plusieurs pages :

* **Page d'accueil :** Cette page présente les produits "favoris" des clients et offre des liens vers les autres pages du site. Les produits mis en avant seront sélectionnés en fonction des tendances actuelles et des préférences des clients.
* **Page E-shop :** Permet à l’utilisateur de choisir la collection qu’il souhaite visiter. Cette page affichera les différentes collections disponibles.
* **Page concept :** Présente la vision et les valeurs d'Élégance Couture pour partager la passion de l'entreprise pour la mode masculine.
* **Page boutique :** Permet à l’utilisateur de savoir où nous retrouver et lui montre l’ambiance qui règne dans nos commerces. Cette page inclura des informations sur les emplacements physiques des magasins, des photos des boutiques.
* **Page de connexion :** Permet aux utilisateurs de se connecter à leur compte pour accéder à des fonctionnalités personnalisées et gérer leurs informations personnelles.
* **Page Panier :** Affiche le contenu du panier de l'utilisateur et lui permet de modifier les quantités, de supprimer des articles ou de passer à la caisse.
* **Page d’inscription :** Permet aux nouveaux utilisateurs de créer un compte
* **Page d’administration :** Permet à l'équipe d'administration de gérer le contenu du site, y compris les produits, les commandes, les utilisateurs et les articles de blog.

Les horaires d’ouverture du restaurant sont affichés au bas de toutes les pages pour informer les clients des heures d'ouverture.

**3. Back-end**

* **Connexion :** Il y a deux utilisateurs sur l’application : "l’hôte d’accueil" (l’administrateur) et le client. Les deux ont le même formulaire de connexion, avec pour identifiants une adresse email et un mot de passe. La sécurité des mots de passe est assurée par des techniques de hachage.
* **Présentations des produits :** Chaque produit est présenté avec une photo, un titre, une description et un prix. Tous ces éléments peuvent être ajoutés, modifiés ou supprimés par l'administrateur.
* **Inscription :** Le formulaire d'inscription permet aux utilisateurs de créer un profil sur le site en fournissant des informations de base telles que leur nom, leur adresse email et leur mot de passe.
* **Panier :** Le panier permet aux utilisateurs de sélectionner des produits et de passer à l’achat en fournissant leurs informations de paiement.

**Base de données**

La base de données du site sera conçue pour stocker et gérer toutes les informations pertinentes nécessaires au bon fonctionnement du site e-commerce. Voici les principales caractéristiques de la base de données :

1. **Structure de la base de données :** La base de données sera organisée de manière à pouvoir stocker les informations sur les produits, les utilisateurs, les commandes, les transactions, etc. Une structure de base de données relationnelle sera utilisée pour garantir l'intégrité des données et faciliter les requêtes complexes.
2. **Tables et relations :** Plusieurs tables seront créées pour représenter les différentes entités du système, telles que les produits, les catégories, les utilisateurs, les commandes, les paiements, etc. Des relations appropriées seront établies entre ces tables pour permettre la récupération efficace des données et maintenir la cohérence des informations. (A adapter)
3. **Sécurité des données :** Des mesures de sécurité robustes seront mises en place pour protéger les données sensibles des utilisateurs, telles que les informations personnelles et les détails de paiement. Cela inclura l'utilisation de techniques de hashage pour stocker les mots de passe et la mise en œuvre de contrôles d'accès pour limiter l'accès aux données sensibles.

En concevant une base de données robuste et sécurisée, je m'assurerai que le site e-commerce dispose d'un fondement solide pour stocker, gérer et récupérer efficacement toutes les informations nécessaires à son bon fonctionnement.

**Idées de fonctionnalités supplémentaires potentielles**

1. **Recherche et filtrage des produits :** Ajout d'une fonctionnalité de recherche pour aider les utilisateurs à trouver rapidement les produits qu'ils recherchent. Des options de filtrage par catégorie, prix, taille, etc., seront également disponibles.
2. **Gestion des commentaires et des avis :** Ajout d'une section où les clients peuvent laisser des commentaires et des avis sur les produits. Les commentaires seront modérés avant d'être publiés sur le site.
3. **Paiement en ligne :** Intégration d'une passerelle de paiement sécurisée pour permettre aux utilisateurs de payer leurs commandes en ligne de manière sécurisée.

GESTION DE PROJET

Pour mener à bien mon projet de développement web, j'ai adopté une approche méthodique et organisée de gestion de projet. Voici les principales étapes que j'ai suivies :

1. **Définition des Objectifs** : La première étape de mon projet a été la définition précise de ses objectifs. Pour ce faire, j'ai procédé à une analyse approfondie des besoins spécifiques de mon client (moi-même) et de son produit : Élégance Couture. J'ai ensuite identifié les fonctionnalités essentielles que le site web devra offrir pour répondre aux attentes de ses clients. Cette démarche m'a permis de poser des bases solides pour le développement du projet, en garantissant une alignement clair sur les attentes du client et une orientation cohérente tout au long du processus.
2. Haut du formulaire
3. **Planification** : Ensuite, j'ai élaboré un plan détaillé pour le projet, en décomposant les tâches en étapes réalisables et en attribuant des délais à chaque étape. J'ai également identifié les ressources nécessaires, telles que les compétences requises et les outils à utiliser.
4. **Suivi et Gestion des Tâches** : Pour assurer une progression efficace du projet, j'ai utilisé des outils de gestion de projet comme Trello. Cela m'a permis de garder une vue d'ensemble des activités en cours et de surveiller les progrès réalisés.
5. **Communication :** Dans ce projet, la communication formelle était moins essentielle du fait que le client et le développeur étaient une seule et même personne. Cette situation a éliminé le besoin d'établir des canaux de communication externes, permettant un accès direct à mes propres besoins, idées et exigences. Malgré cela, maintenir un dialogue interne était crucial pour garantir la cohérence et la clarté tout au long du processus de développement. J'ai donc veillé à prendre du recul régulièrement pour évaluer mes propres progrès, à documenter mes décisions et à réfléchir aux ajustements nécessaires pour atteindre les objectifs fixés. Cette approche a offert une plus grande flexibilité, me permettant d'adapter rapidement mes attentes et exigences en fonction des évolutions du projet. En somme, bien que la communication avec un client externe n'était pas nécessaire, maintenir une communication interne organisée et réfléchie était fondamentale pour assurer le succès du projet.
6. **Flexibilité et Adaptabilité** : Tout au long du projet, j'ai été prêt à m'adapter aux changements de priorités ou aux exigences nouvelles émergentes. J'ai gardé un état d'esprit agile, ce qui m'a permis de réagir rapidement aux défis et aux opportunités rencontrés en cours de route.

Spécifications techniques

1.Environnement de travail

Visual Studio Code – J'ai utilisé VsCode comme mon éditeur de code principal pour développer mon projet. C'est un éditeur léger et puissant qui offre de nombreuses fonctionnalités pratiques telles que la coloration syntaxique, l'autocomplétion, le débogage intégré et une grande variété d'extensions pour personnaliser mon environnement de développement.

Git & GitHub – C’est l’outil de versioning que j’ai choisi, il me permet de déposer mon travail sur un dépôt distant, et surtout de pouvoir travailler sur de nouvelles fonctionnalités de mon code en sécurité en utilisant une branche différente. J’utilise uniquement des lignes de commandes sur le terminal Git Bash de mon éditeur Visual Studio Code. J'ai utilisé des commandes Git telles que commit, push, pull et merge pour gérer les versions de mon code et travailler efficacement

Trello – Un outil essentiel pour ma productivité, qui me permet de suivre ma progression mais surtout d’organiser mon travail par tâche et par priorité (à faire, en cours, terminé), on appelle également cette méthode la « Méthode Kanban ».

Figma – J'ai utilisé Figma comme outil de conception pour élaborer les maquettes de mon projet. Avec Figma, j'ai pu partager mes conceptions en temps réel avec des collègues et des formateurs, ce qui m'a permis de recevoir leurs retours instantanément. De plus, Figma m'a offert la possibilité de créer des designs interactifs pour concevoir l'interface utilisateur de mon application de manière dynamique.

Draw.io – Pour créer des diagrammes et des schémas pour mon projet, j'ai utilisé Draw.io. C'est un outil en ligne simple et intuitif qui m'a permis de créer des diagrammes de séquence, de classe et d’utilisation et d'autres types de schémas pour représenter, la structure des bases de données. J'ai trouvé Draw.io très utile pour communiquer des idées complexes de manière claire et concise.

InfinityFree-- Une plateforme de déploiement qui permet aux développeurs de déployer, gérer et héberger des applications web. Dans mon projet, j'ai choisi d'héberger mon application sur InfinityFree pour sa simplicité de déploiement. J'ai utilisé les outils fournis par InfinityFree, tels que le tableau de bord web, pour déployer et gérer mon application en ligne.

XAMMP-- J'ai utilisé XAMPP comme environnement de développement local pour mon projet. Avec XAMPP, j'ai pu installer un serveur web Apache, une base de données MySQL, et d'autres outils comme PHP et phpMyAdmin sur ma machine locale. Cela m'a permis de développer et de tester mon application localement avant de la déployer sur un serveur en production. XAMPP offre une solution pratique et complète pour créer un environnement de développement web sur votre propre ordinateur, ce qui facilite le processus de développement et de débogage de votre projet.

1. Modèle utilisé

Dans mon projet, j'ai opté pour une architecture basée sur le modèle MVC (Model-View-Controller). Pour commencer, j'ai placé tout mon frontend, comprenant HTML, CSS et JavaScript, dans la partie Vue (View). Cela inclut l'interface utilisateur visible par les utilisateurs de mon application.

Ensuite, j'ai décidé de placer le PDO dans la partie Modèle (Model), et je l’ai utilisé pour interagir avec la base de données. PDO m'a permis de fournir une interface générique pour accéder et manipuler les données.

Enfin, j'ai organisé la logique de traitement des requêtes et des réponses dans la partie Contrôleur (Controller). C'est là que j'ai géré les entrées de l'utilisateur, appelé les méthodes du Modèle via PDO pour récupérer ou manipuler des données, puis préparé les données à envoyer à la Vue pour affichage.

En suivant cette approche, j'ai pu séparer clairement les différentes préoccupations de mon application, ce qui l'a rendue plus modulaire, maintenable et extensible. En respectant les principes du modèle MVC, j'ai pu concevoir une application bien organisée et facile à comprendre.

3.Langages utilisés

1. Front-end

HTML - J'ai utilisé HTML pour structurer le contenu de mes pages web. Cela inclut la création de balises pour définir la structure des éléments tels que les en-têtes, les paragraphes, les liens, les images, etc.

CSS / Bootstrap - Pour le style et la mise en page de mon application, j'ai alterné entre CSS pur et le framework CSS Bootstrap en fonction de mes besoins. CSS m'a permis de personnaliser le style de mes éléments HTML, tandis que Bootstrap m'a fourni des composants prêts à l'emploi et des grilles pour rendre mon application responsive..

JavaScript - J'ai utilisé JavaScript pour ajouter de l'interactivité et de la dynamique à mon application côté client. Cela inclut la validation des formulaires, les animations, les requêtes AJAX pour les mises à jour asynchrones, etc.

2.3 Back-end

PHP – Un langage back-end complet, que j’ai utilisé pour gérer la logique côté serveur de mon application web. Cela inclut le traitement des formulaires, l'accès à la base de données, la gestion des sessions utilisateur, etc.

SQL -- (Structured Query Language) est un langage de programmation utilisé pour gérer et manipuler des bases de données relationnelles. Dans mon projet, j'ai utilisé SQL pour créer et gérer la structure de ma base de données, ainsi que pour effectuer des opérations telles que l'insertion, la mise à jour, la suppression et la récupération de données.

MySQL -- Un système de gestion de base de données relationnelle open-source largement utilisé. Dans mon projet, j'ai choisi MySQL comme SGBDR pour stocker et gérer les données de mon application. J'ai utilisé des requêtes SQL pour interagir avec la base de données MySQL, notamment pour créer des tables, insérer des données

1)Installation et Configuration

1. Installation de l’environnement de travail

Tout d'abord, je vais configurer mon environnement de développement en installant les outils nécessaires tels qu’un éditeur de code, ici Visual Studio Code, un outil de conception graphique, ici Figma, un outil de création de graphique, ici Draw.io et enfin le gestionnaire de version, ici Git avec son terminal dédié (Git Bash).

Avant de commencer le développement de mon application, il est nécessaire d’avoir les prérequis suivants installés sur mon ordinateur :

PHP 7.1 ou version ultérieure - - -

MySQL ou un autre serveur de base de données compatible avec PHP

Un serveur Web (par exemple Apache ou Nginx).

Je recommande l’utilisation de Xampp qui contient l’ensemble de ces trois technologies nécessaires au démarrage de mon projet ainsi que PhpMyAdmin qui dispose d’une interface intéressante pour la gestion de BDD.

Il faut maintenant initialiser un projet Git avec la commande git init, ensuite on doit créer un repository distant sur GitHub et faire la connexion avec le projet local.

Enfin, après avoir veillé à être positionné sur le répertoire de notre projet on peut lancer l’application via Xampp en activant le serveur web Apache et en y accédant sur notre navigateur via l’URL <http://localhost:8000/>

2 Installation et configuration de la base de données

Le type de base de données que j’ai choisi est le MySQL, le langage backend utilisé pour interagir avec est le SQL.

J’ai choisi d’utiliser dans un premier temps l’interface PHPMyAdmin pour seulement avoir une vue sur ma base de données, je n’ai en aucun cas utilisé ses fonctionnalités pour créer, lire, mettre à jour, ou supprimer des données (CRUD). Nous verrons plus tard comment j’ai procédé pour cela.

L’installation d’un serveur local (ici Xampp) est également nécessaire pour prendre en charge notre application et en même temps avoir accès directement à l’interface PHPMyAdmin.

Une fois ceci fait, l’outil qui va nous permettre dans PHP d’interagir avec la base de données est le PDO qui fera le lien avec le SGBDR (ici MySQL)

2. Développement de l’application

**1.Maquette**

Étape 1 : **Analyse des Besoins**

Dans un premier temps, je m'attache à comprendre les besoins du projet en collaborant étroitement avec les parties prenantes. Cette phase implique une analyse approfondie des fonctionnalités requises et des objectifs spécifiques de l'interface utilisateur.

Étape 2 : **Recherche et Inspiration**

Je m'engage dans une recherche approfondie pour recueillir des inspirations et rester à jour avec les tendances actuelles en matière de conception. J'explore divers sites web, applications mobiles et autres sources pour identifier des éléments de design pertinents.

Étape 3 : **Création d'un Zonning Initial avec Figma**

En utilisant l'outil de conception Figma, je conçois un wireframe initial. Mon objectif est de me concentrer sur la disposition des éléments principaux sans entrer dans les détails visuels. Figma facilite cette étape en permettant une création rapide et collaborative.

Étape 4 : **Wireframing et Structuration de l'Information**

J'organise l'information de manière logique en utilisant des groupes, des sections et des menus si nécessaire. Je veille à ce que la navigation soit intuitive en respectant les meilleures pratiques en UX (expérience utilisateur).

Étape 5 : **Hiérarchisation des Éléments**

Je définis une hiérarchie visuelle pour les éléments en fonction de leur importance. J'utilise les fonctionnalités de mise en forme de Figma pour guider l'attention de l'utilisateur à l'aide de différences de taille, de couleur et de style.

Etape 6 : **Charte Graphique**

Pour la mise en place de la charte graphique, je veille à choisir une palette de couleurs et des polices en harmonie avec l'identité de marque et le public cible. En parallèle, j'identifie les éléments visuels tels que les logos et les icônes pour renforcer cette identité. Ces décisions sont consignées dans un style guide, garantissant ainsi la cohérence visuelle à travers le projet. Enfin, je m'assure que la charte graphique offre une expérience utilisateur intuitive et conforme aux meilleures pratiques en matière de navigation web.

ETAPE 7 : **Maquette fonctionnelle :**

A partir du wireframe, je crée une maquette fonctionnelle. J'utilise des outils de conception tels que des formes, des icônes et des couleurs pour rendre la maquette plus attrayante. J'ajoute des éléments réels, tels que le logo et les liens de navigation.

Étape 8 : **Réajustement dans Figma**

Grâce aux fonctionnalités de Figma, je peux itérer et réajuster ma maquette en fonction des retours des tests utilisateurs ou des discussions avec les parties prenantes. La collaboration en temps réel facilite le travail d'équipe.

En suivant ces étapes avec Figma, je parviens à créer une maquette d'interface utilisateur web et web mobile bien pensée, alignée sur les besoins du projet et offrant une expérience utilisateur optimale.

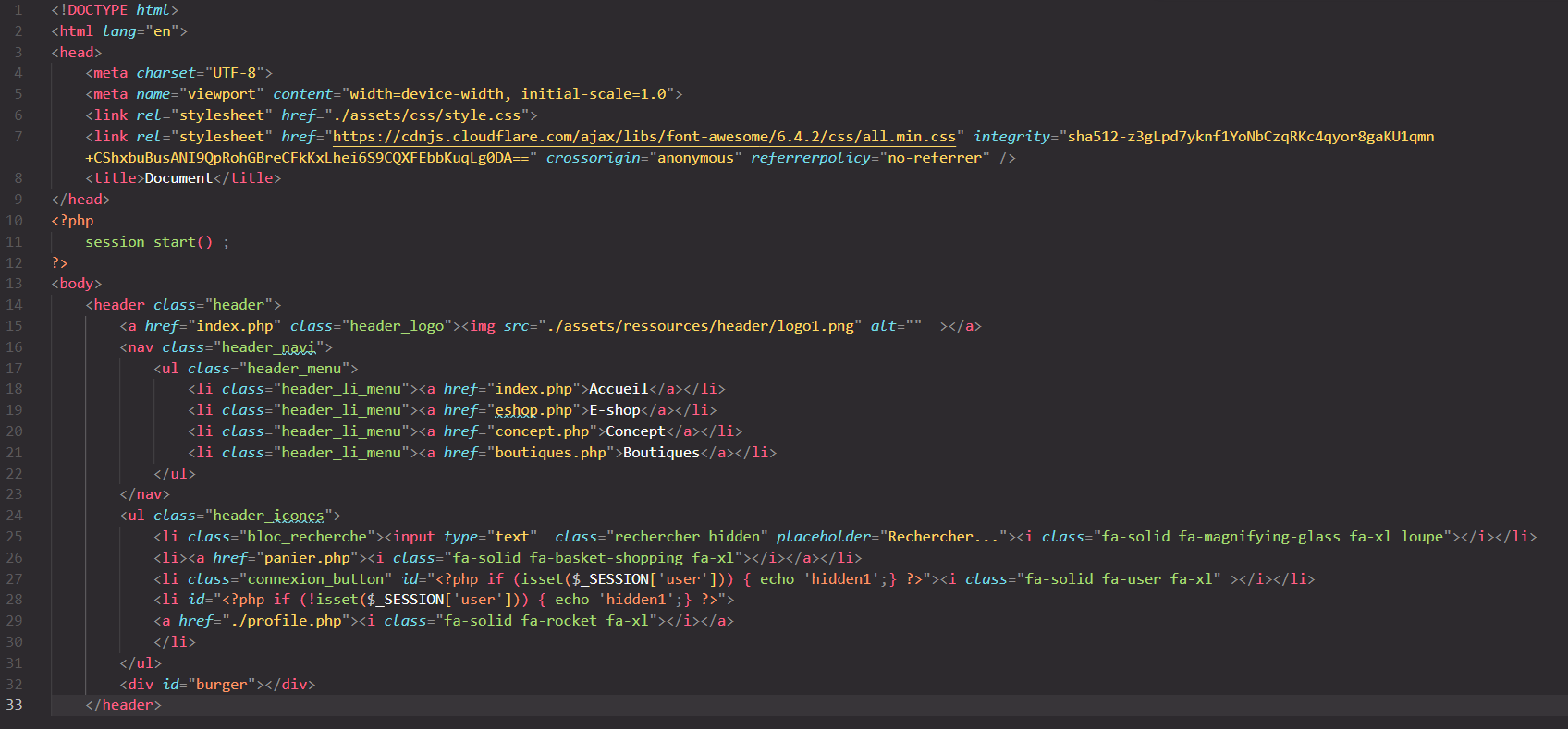
Développement de l’application

1)Front-end

**1)Analyse des maquettes :**

Dans cette phase, après m’être assurer de la bonne correspondance de la maquette avec les besoins du projet, j'analyse attentivement les maquettes pour comprendre les fonctionnalités et l'expérience utilisateur. J'identifie les éléments à traduire en fonctionnalités techniques et de quelle manière procéder. Je veille à ce que l'application réponde aux attentes des utilisateurs finaux tout en respectant mes contraintes techniques et mes objectifs.

**2)Développement de la structure HTML :**

****

Je commence par déclarer le type de document avec **<!DOCTYPE html>**. C'est une étape importante pour indiquer au navigateur qu'il s'agit d'un fichier HTML5. Ensuite, je commence ma balise **<html>**, qui enveloppera tout mon contenu HTML.

Dans la balise **<head>**, je place des informations supplémentaires sur ma page, comme les métadonnées et les liens vers des fichiers externes. Les métadonnées définissent des informations telles que l'encodage de caractères et les paramètres d'affichage. Les balises **<link>** permettent de référencer des fichiers externes, comme mes feuilles de style CSS et les icônes de Font Awesome. Enfin, avec la balise **<title>**, je spécifie le titre de ma page qui apparaîtra dans l'onglet du navigateur.

Maintenant, j'entre dans le corps de ma page avec la balise **<body>**. C'est là que tout le contenu visible de ma page sera placé.

La structure principale de ma page est divisée en trois parties distinctes :

Mon en-tête est encapsulé dans la balise **<header>**. À l'intérieur, je trouve mon logo et ma barre de navigation principale. La navigation est contenue dans une balise **<nav>**, où je liste les liens vers différentes pages de mon site. En plus de cela, j'ai une zone de recherche et des icônes supplémentaires comme un panier et des icônes de connexion.

* Le **<main>** : C'est là que je place mon contenu principal.
* Le **<footer>** : C'est la section qui contient les informations de bas de page, telles que les conditions d'utilisation et les liens utiles.

Chaque section est organisée à l'aide de la balise **<div>**, qui est une division générique utilisée pour grouper du contenu dans une page HTML. J'utilise des classes pour identifier les éléments, ce qui me permet de les styliser à ma guise via mon fichier CSS.

Mais ici intéressons-nous à la structure spécifique de cette page « header.php ».

À l'intérieur de la balise **<nav>**, j'utilise **<ul>** pour créer une liste non ordonnée de liens de navigation.Dans cette balise **<ul>**, chaque élément **<li>** représente un élément de la liste, c'est-à-dire un lien de navigation. À l'intérieur de chaque **<li>**, j'utilise la balise **<a>** pour créer un lien hypertexte vers une page spécifique de mon site. Cela permet aux utilisateurs de naviguer facilement entre les différentes pages. Dans la balise **<a>**, la balise **<i>** est utilisée pour intégrer des icônes de Font Awesome. Ces icônes ajoutent des éléments visuels attrayants et informatifs à mes liens de navigation. Enfin j'utilise la balise **<input>** pour créer un champ de recherche dans ma barre de navigation.

Passons maintenant aux attributs :

**class** : J'attribue des classes à plusieurs éléments HTML pour les identifier et les styliser avec du CSS. Par exemple, j'utilise la classe "header" pour styliser mon en-tête, et "header\_menu" pour styliser ma liste de navigation.

**id** : J'utilise les attributs **id** pour cibler des éléments spécifiques avec du JavaScript ou du CSS. Par exemple, j'ai des identifiants comme "hidden1" qui contrôlent l'affichage conditionnel d'éléments, comme le bouton de connexion.

**alt** : J'utilise cet attribut avec la balise **<img>** pour fournir un texte alternatif pour mes images. Cela est utile si l'image ne peut pas être chargée ou pour les utilisateurs qui utilisent des lecteurs d'écran et représente une bonne pratique SEO appréciée des navigateurs.

**href** : Cet attribut est utilisé avec la balise **<a>** pour spécifier l'URL de destination du lien. Par exemple, **href="index.php"** pointe vers la page d'accueil de mon site.

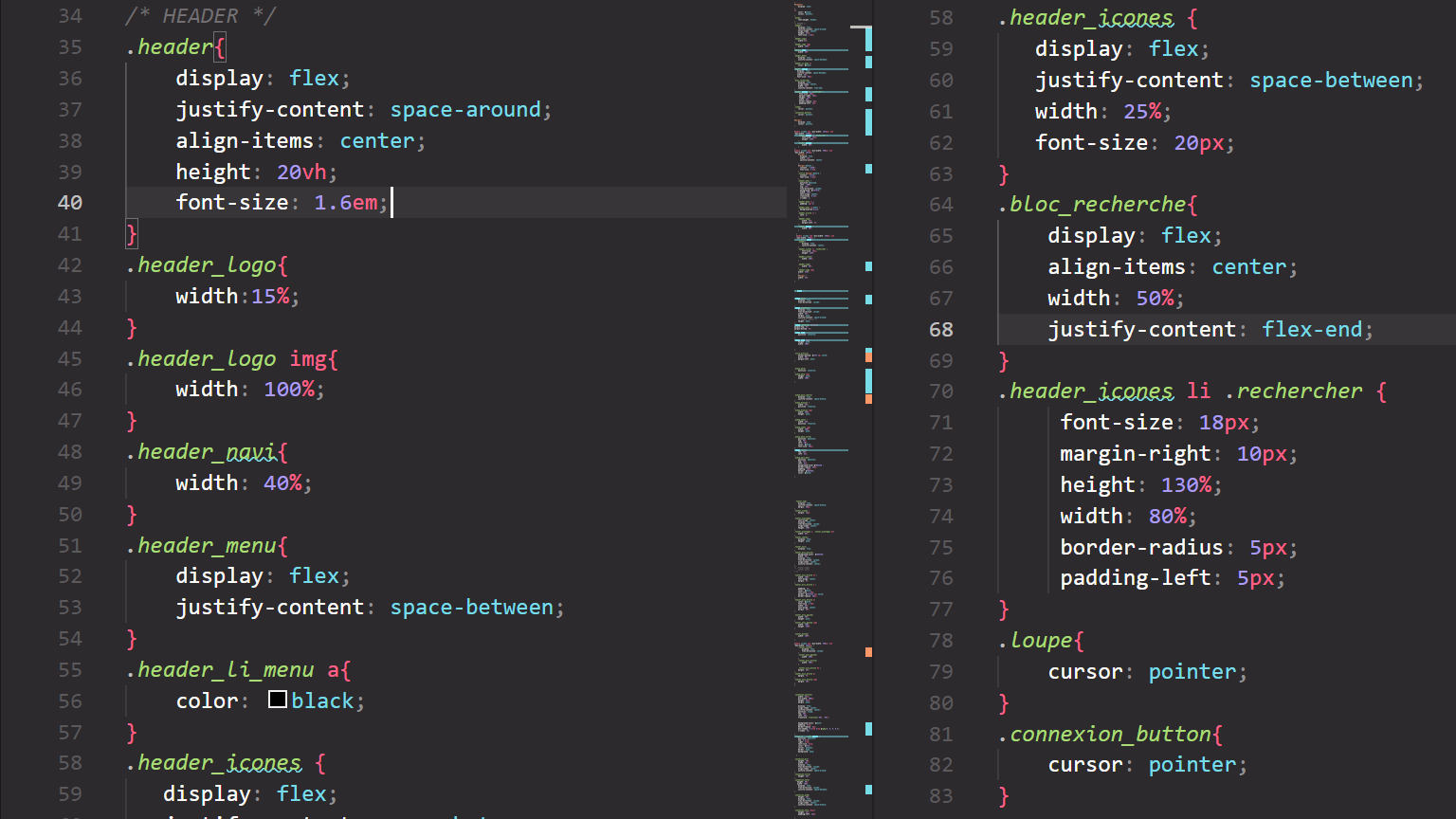
Ainsi, en combinant ces balises et attributs, je structure ma page web et fournis une expérience de navigation intuitive et interactive pour les utilisateurs.

De plus, j'ai une gestion conditionnelle avec PHP, par exemple pour afficher certains éléments uniquement si l'utilisateur est connecté, mais nous reviendrons dessus un peu plus tard dans ce dossier.

En résumé, mon code HTML est structuré de manière logique et organisée, suivant les bonnes pratiques de développement web.

.

**3)Stylisation avec CSS :**

****

Dans ce code CSS, j'organise l'apparence de l'en-tête de ma page web. Avec la classe ".header", je définis les bases en utilisant "display: flex;" pour créer une mise en page flexible. Ainsi, avec "justify-content: space-around;", je répartis les éléments uniformément autour de l'en-tête et "align-items: center;" pour les aligner verticalement au centre. En fixant "height: 20vh;", je détermine la hauteur de mon en-tête à 20% de la hauteur de la fenêtre du navigateur, et avec "font-size: 1.6em;", je spécifie la taille de la police.

Ensuite, avec ".header\_logo", je m'occupe du logo en lui attribuant une largeur de 15% de son conteneur parent. Pour que le logo s'affiche correctement, j'applique également "width: 100%;" à l'image elle-même dans ".header\_logo img".

La barre de navigation est stylisée avec ".header\_navi", à laquelle je donne une largeur de 40% de l'en-tête. À l'intérieur, ".header\_menu" organise les liens de navigation avec "display: flex;" pour créer un modèle de boîte flexible, et "justify-content: space-between;" pour les répartir uniformément.

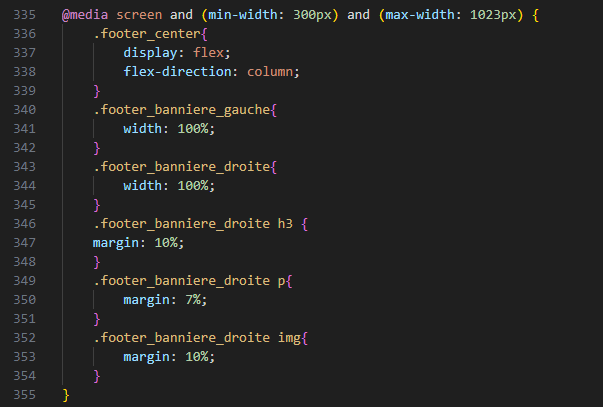
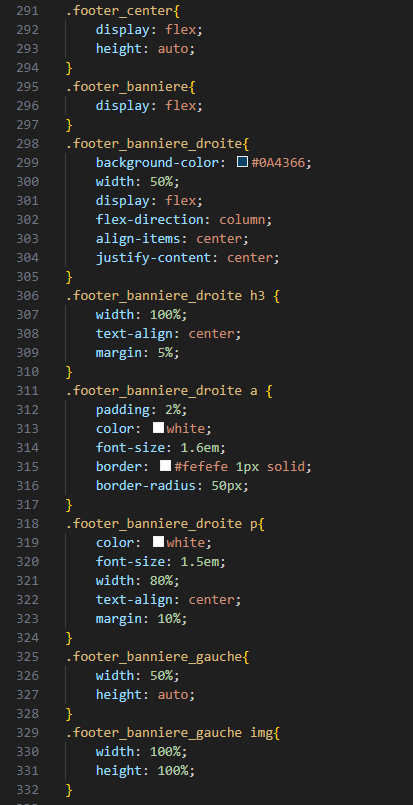
Les liens eux-mêmes sont stylisés par ".header\_li\_menu a", avec "color: black;" pour définir leur couleur de texte à noir.

Les icônes dans l'en-tête sont alignées et espacées de manière uniforme avec ".header\_icones", qui utilise également "display: flex;" et "justify-content: space-between;". Sa largeur est définie à 25% de l'en-tête.

La zone de recherche est organisée par ".bloc\_recherche", qui utilise "display: flex;" pour un alignement flexible et "justify-content: flex-end;" pour le positionner à droite. La taille du champ de recherche est affinée avec ".header\_icones li .rechercher", ajustant la taille de la police et les marges. Les classes ".loupe" et ".connexion\_button" ajoutent des interactions utilisateur. le curseur changera de forme en une « main pointeur » pour indiquer que le bouton est cliquable.

**4)Responsive design :**

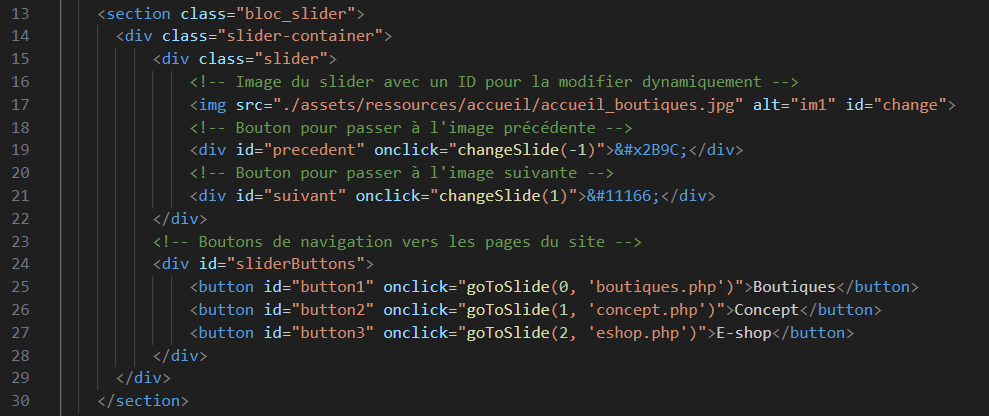
Dans cette partie du CSS, je configure le style d'une section du footer de mon site. J'utilise Flexbox pour aligner les éléments horizontalement dans la classe **.footer\_center**. C’est une bannière horizontale qui affiche **.footer\_banniere\_gauche** une image qui prend 50% de la largeur de son parent en laissant **.footer\_banniere\_droite**, un bloc avec un arrière-plan coloré occupé les 50% restant. Les éléments à l'intérieur de ce bloc sont centrés verticalement et horizontalement grâce au règles flex.

  
Dans la version responsive du code CSS, j'apporte plusieurs ajustements pour améliorer l'expérience utilisateur sur des écrans plus petits, tels que les tablettes. Le changement le plus important se situe au niveau de la disposition des éléments à l'intérieur du footer. J'utilise la propriété **flex-direction: column** dans la classe **.footer\_center** pour que les éléments à l'intérieur adoptent une disposition verticale plutôt qu'horizontale, ce qui optimise l'utilisation de l'espace sur des écrans plus restreints.

Ensuite, je redimensionne les parties gauche et droite de la bannière en leur attribuant une largeur de 100% au lieu de 50% grâce à la règle **width: 100%** dans les classes **.footer\_banniere\_gauche** et **.footer\_banniere\_droite**. Cela signifie qu'elles occuperont toute la largeur de l'écran, offrant ainsi une présentation plus adaptée à des appareils de taille réduite.

Pour les éléments à l'intérieur de la partie droite de la bannière, tels que les titres, les paragraphes et les images, j'ajuste les marges pour créer un espacement plus confortable entre ces éléments et les bords de l'écran. J'utilise les règles **margin** pour augmenter les espaces dans les classes **.footer\_banniere\_droite h3**, **.footer\_banniere\_droite p** et **.footer\_banniere\_droite img**. Cette modification vise à garantir une meilleure lisibilité et une disposition plus équilibrée sur des écrans de taille réduite.

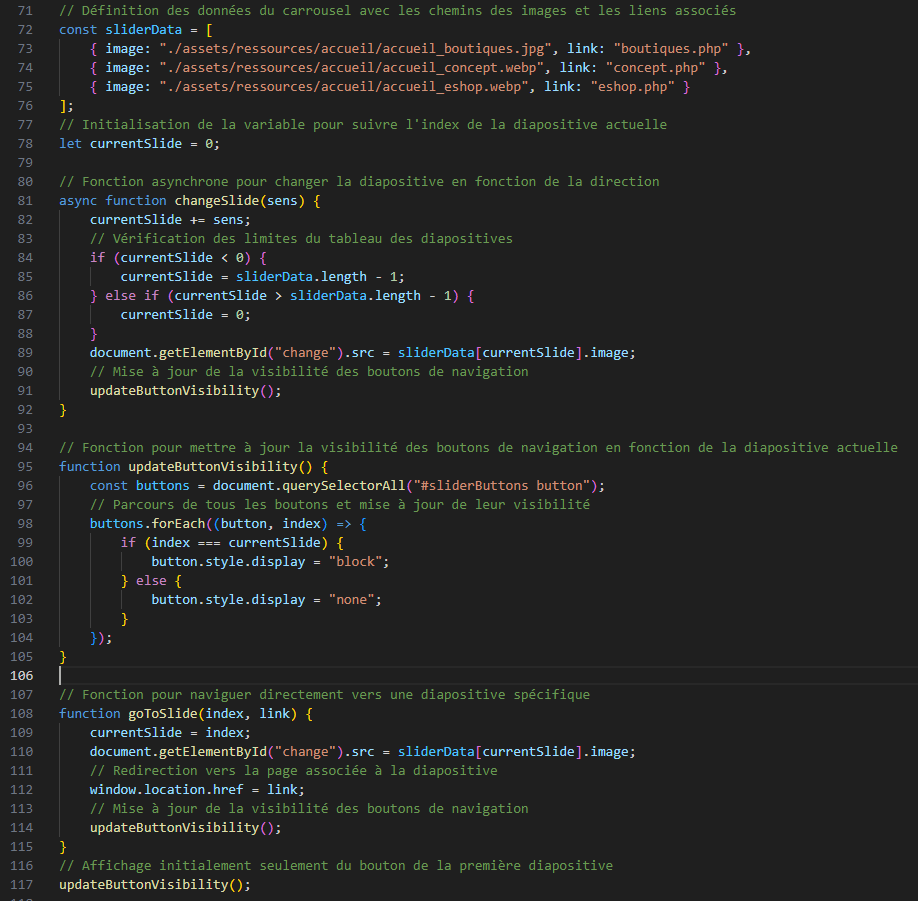
**5)Interactivité avec JavaScript :**

Dans cette section HTML, je crée un carrousel d'images avec des boutons de navigation vers différentes pages de mon site. La classe "bloc\_slider" est utilisée pour styliser cette partie de la page. À l'intérieur, j'ai un conteneur "slider-container" qui englobe tous les éléments du carrousel.

Dans le carrousel lui-même, j'ai un conteneur "slider" qui contient l'image principale à afficher. Chaque image est représentée par une balise **<img>** avec un ID "change", ce qui me permet de la modifier dynamiquement via JavaScript.

En dessous de l'image, j'ai deux divs qui servent de boutons de navigation. L'un est pour passer à l'image précédente et l'autre pour passer à l'image suivante. Ces boutons sont stylisés avec des flèches et ils ont des attributs **onclick** qui appellent les fonctions JavaScript correspondantes pour changer les diapositives du carrousel.

Enfin, j'ai un autre conteneur "sliderButtons" qui contient des boutons pour naviguer vers différentes pages du site. Chaque bouton est associé à une page spécifique et possède un attribut **onclick** qui déclenche une fonction JavaScript pour afficher la diapositive correspondante et rediriger l'utilisateur vers la page associée.

****

Dans ce code, je crée un carrousel d'images avec des liens vers différentes pages.

Je commence par définir un tableau nommé **sliderData**. Chaque élément de ce tableau est un objet contenant l'URL d'une image et le lien vers une page associée. Ces données seront utilisées pour afficher les images dans le carrousel et pour rediriger l'utilisateur vers les pages correspondantes.

Ensuite, j'initialise une variable **currentSlide** à zéro. Cette variable va suivre l'index de la diapositive actuellement affichée dans le carrousel.

La fonction **changeSlide(sens)** est la fonction principale qui permet de changer de diapositive dans le carrousel. Elle prend en paramètre un argument **sens**, qui indique la direction dans laquelle nous voulons naviguer dans le carrousel (avancer ou reculer).

Dans cette fonction, j'incrémente ou je décrémente la variable **currentSlide** en fonction de la direction donnée. Ensuite, je vérifie si l'index de la diapositive dépasse les limites du tableau. Si c'est le cas, je réinitialise l'index pour assurer une navigation continue dans le carrousel.

Enfin, j'actualise l'image affichée dans le carrousel en modifiant l'attribut **src** de l'élément HTML correspondant. Je m'assure également que les boutons de navigation sont mis à jour pour refléter la diapositive actuelle.

La fonction **updateButtonVisibility()** gère la visibilité des boutons de navigation. Elle parcourt tous les boutons et affiche uniquement celui correspondant à la diapositive actuelle, cachant ainsi les autres.

La fonction **goToSlide(index, link)** permet de naviguer directement vers une diapositive spécifique en utilisant son index. Elle met à jour l'image affichée et redirige l'utilisateur vers la page associée à cette diapositive.

En résumé, ce code crée un carrousel d'images avec des liens interactifs, permettant à l'utilisateur de naviguer facilement entre les différentes pages associées aux images.

Haut du formulaire

2)Back-end

**1)Base de données**

1. Conception de la base de données

Comme il est essentiel pour la partie front d’établir une maquette (wireframe et/ou MockUp) afin de matérialiser notre réflexion avant de coder, il est encore plus indispensable de modéliser notre base de données sous forme de diagrammes, avant de créer notre système.

Pour se faire j’ai conçu les trois diagrammes suivants qui chacun ont un rôle différent.

1.1 Le diagramme de classe (ou la méthode Merise)

La méthode Merise est une méthode de conceptualisation des données basée sur le modèle MCD (Modèle Conceptuel de Données), son but est de faciliter la communication entre tous les acteurs du processus de développement d’un projet, bien qu’ils soient étrangers au domaine du développement informatique.

Pour ma part elle m’a surtout servi à organiser mon système d’information et éviter de basculer dans un hors sujet. Grâce à cette méthode j’ai pu déterminer les éléments suivants : - - -

Les entités, qui représentent chacune un ensemble d’information (propriétés). Elles possèdent toute un identifiant « \_id ».

Exemple : l’entité « Carte » possède un identifiant « carte\_id » et plusieurs propriétés comme par exemple « titre »

Les relations, qui sont les liaisons des entités.

Exemple : un Utilisateur « prend » une réservation, ou une réservation est « prise » par un utilisateur. Les cardinalités, c’est le nombre de fois que l’entité participe à une relation, elle est représentée sous la forme « 1,1 », « 1, n » ou « 0, n »

Exemple : un Utilisateur « est » au minimum « 1 » client et au maximum plusieurs (« n ») clients. PHOTO DIAGRAMME DE CLASSE

1.2 Le diagramme de cas d’utilisation

Le diagramme de cas d’utilisation donne une description globale du comportement de l’application, il identifie les acteurs et leur interaction avec celle-ci.

Exemple : l’acteur « visiteur » peux consulter la carte, le menu, les horaires, et la galerie. Mais il peut aussi Réserver sans s’inscrire, ou s’inscrire avant de réserver. PHOTO DIAGRAMME DE CAS D’UTILISATION

1.3 Le diagramme de séquence

Comme son nom l’indique ce diagramme présente une séquence en particulier, mais cette fois ci il est décrit sur une ligne de temps (de haut en bas), grâce à cela on peut prendre une action de l’utilisateur et présenter toutes les interactions qu’elle déclenche les unes après les autres.

La lecture du diagramme de séquence ci-dessous décrivant la fonctionnalité « Réservation » se fait donc simplement de haut en bas.

Exemple : Le client clique sur le bouton réservation, ce qui ouvre un formulaire de réservation, peut ensuite remplir le formulaire, qui communique à son tour avec la base de données etc.. PHOTO DIAGRAMME DE SEQUENCE

**2)Intégration Communication avec la base de données :**

La section de connexion à la base de données du code PHP utilise l'extension PDO (PHP Data Objects) pour établir une connexion sécurisée avec une base de données MySQL. Cette connexion utilise les informations d'hôte, de nom de base de données, d'utilisateur et de mot de passe qui sont définies dans les variables $db\_host, $db\_name, $db\_user et $db\_password.

Dans la partie "try" du script, la logique est centrée sur le traitement des données soumises via un formulaire. Lors de la détection d'une requête POST, les valeurs des champs genre, nom, prénom, mail et password sont extraites de la variable $\_POST. Ensuite, une instruction SQL préparée est utilisée pour insérer ces données de manière sécurisée dans une table nommée "users" dans la base de données. Cette requête est préparée avec des paramètres de substitution pour éviter les attaques par injection SQL. Les paramètres de substitution sont des espaces réservés dans une requête SQL préparée qui sont remplacés par les valeurs réelles lors de l'exécution. Ils sont utilisés pour sécuriser les requêtes en évitant les injections SQL. En PHP PDO, ils sont représentés par des **?** ou des noms commençant par **:** dans la requête SQL. Les valeurs des champs du formulaire sont ensuite passées à la méthode execute() pour effectuer l'insertion dans la base de données.

Le mécanisme "try-catch" est associé à la gestion des erreurs. En cas d'exception, généralement causée par une erreur dans l'exécution de la requête SQL, le script intercepte l'erreur et la gère de manière appropriée. Cela permet d'afficher des informations détaillées sur l'erreur tout en garantissant la stabilité de l'application.

En résumé, ce script établit une connexion sécurisée à la base de données et traite les données soumises via un formulaire, les insérant de manière sécurisée dans une table spécifiée. La gestion des erreurs est intégrée pour garantir un fonctionnement fluide et identifier rapidement tout problème éventuel.

**3)Développement des fonctionnalités**

**4)Sécurité :** Je commence par une fonction nommée **verif\_pass**, qui s'occupe de vérifier la complexité des mots de passe fournis par les utilisateurs. Pour cela, j'utilise une expression régulière qui définit les critères de complexité : au moins une lettre majuscule, une lettre minuscule, un chiffre, un caractère spécial et une longueur minimale de 8 caractères. Si le mot de passe respecte ces critères, la fonction renvoie vrai, sinon elle renvoie faux.

Ensuite, je m'assure que tous les champs requis du formulaire sont remplis et contiennent des données valides. Ces champs sont le genre, le nom, le prénom, l'adresse e-mail et le mot de passe. Pour cela, j'utilise les fonctions **isset** et **empty** pour vérifier si les champs sont définis et non vides.

Une fois que je m'assure que les données du formulaire sont correctes, je les nettoie en utilisant la fonction **htmlspecialchars**. Cela convertit les caractères spéciaux en entités HTML, ce qui rend les balises HTML inoffensives. Cela aide à prévenir les attaques XSS, où un attaquant tenterait d'insérer du code malveillant dans les champs du formulaire.

Enfin, si le mot de passe satisfait aux critères de complexité, je le hash avec la fonction **password\_hash**. Le hashage sécurise le mot de passe en le transformant en une chaîne de caractères aléatoire et irréversible. Cela garantit que même si la base de données est compromise, les mots de passe des utilisateurs restent sécurisés.

**5)Tests unitaires et tests d'intégration :**

**6)Gestion des erreurs et des exceptions :**

Déploiement

Pour déployer mon site e-commerce simple, "Elegance Couture", j'ai choisi d'utiliser InfinityFree comme plateforme d'hébergement et FileZilla comme client FTP. Voici le processus que j'ai suivi :

Tout d'abord, je me suis inscrit sur InfinityFree pour obtenir un espace d'hébergement gratuit. J'ai configuré mon domaine gratuit, elegancecouture.infinityfreeapp.com, pour y héberger mon site.

Ensuite, j'ai téléchargé et installé FileZilla sur mon ordinateur. Une fois installé, j'ai ouvert FileZilla et j'ai saisi les informations de connexion fournies par InfinityFree, notamment l'hôte, le nom d'utilisateur et le mot de passe.

Après m'être connecté à mon compte InfinityFree via FileZilla, j'ai navigué sur mon ordinateur pour localiser tous les fichiers de mon site e-commerce, y compris les fichiers PHP, HTML, CSS, JavaScript et les ressources telles que les images.

Je n'ai pas besoin de configurer des paramètres spécifiques dans les fichiers de configuration de mon application pour l'environnement de production, ni d'installer des dépendances externes.

Cependant, j'ai besoin d'une base de données pour mon site e-commerce. Je vais donc importer ma base de données sur InfinityFree.

Je ne sais pas quel serveur web est utilisé sur InfinityFree, ni si je dois configurer des certificats SSL pour mon domaine. Je vais donc également demander ces informations au support d'InfinityFree.

Il n'y a pas de mesures de sécurité spécifiques, de surveillance ou de journalisation, de procédures de sauvegarde et de restauration, ni d'intégration avec des outils de déploiement continu pour ce déploiement, car c'est ma première expérience de déploiement.

Je n'ai pas non plus rencontré de problèmes courants lors de déploiements précédents, et je ne sais pas où trouver les points de contact pour obtenir de l'aide en cas de problème.

Enfin, il n'y a pas de notes de version ou de mises à jour spécifiques à inclure dans la documentation, car il s'agit d'un déploiement initial.

Haut du formulaire

SEO

Le SEO, acronyme de Search Engine Optimization (Optimisation pour les moteurs de recherche), englobe un ensemble de tactiques visant à améliorer la visibilité d'un site web dans les résultats organiques des moteurs de recherche tels que Google. Son objectif premier est d'accroître le flux de visiteurs vers un site en favorisant son classement élevé dans les pages de résultats de recherche.

Pour évaluer la pertinence et la qualité d'un site web par rapport à une requête spécifique, les moteurs de recherche utilisent des algorithmes sophistiqués. Ces algorithmes prennent en compte divers critères, notamment la pertinence des mots-clés, la qualité du contenu, la structure du site, ainsi que la quantité et la qualité des liens externes pointant vers celui-ci.

Il est primordial de noter que les moteurs de recherche mettent régulièrement à jour leurs algorithmes. Il est donc crucial de rester à jour avec les meilleures pratiques de SEO afin de maintenir une visibilité optimale dans les résultats de recherche.

Le SEO joue un rôle crucial dans la stratégie de marketing en ligne de toute entreprise, car un classement élevé dans les moteurs de recherche peut entraîner un flux de trafic significatif. Cependant, il est important de souligner que le SEO demande du temps et des efforts continus, et les résultats ne sont généralement pas instantanés. Il est donc essentiel de persévérer dans la mise en œuvre de techniques SEO efficaces pour maximiser les avantages à long terme.

Ajouter le travail personnel sur le SEO

RGPD :

CONCLUSION

En conclusion, la création de ce site web pour mon site e-commerce est le résultat d'un engagement profond et d'une réflexion méticuleuse. Les fonctionnalités discutées, telles que la page d'accueil attrayante, la section de catalogue de produits, l'espace dédié aux clients, le blog informatif, les outils interactifs, ainsi que la galerie de photos et de vidéos, ont été soigneusement élaborées pour offrir une expérience complète et engageante aux visiteurs.

Envisager l'évolution potentielle de ce site vers un système de gestion pour mon e-commerce représente une opportunité significative de croissance et d'efficacité. La gestion intégrée des produits, des stocks, des commandes et des paiements en ligne simplifiera considérablement les opérations quotidiennes de l'entreprise, contribuant ainsi à améliorer la satisfaction des clients.

Ce projet a été une véritable exploration, me permettant d'acquérir de nouvelles compétences et de me familiariser avec des outils et technologies innovants tels que HTML, CSS, JavaScript et PHP et d'autres. Ces compétences nouvellement acquises sont précieuses et constitueront une base solide pour mes projets futurs dans le domaine du développement web.

Je tiens à exprimer ma gratitude envers toutes les personnes qui m'ont soutenu tout au long de ce projet, en particulier mon entourage, mes amis, et les ressources en ligne qui m'ont aidé à surmonter les défis et à concrétiser ce site web.

En résumé, ce projet marque une étape significative dans mon parcours de développement web, et je suis fier du résultat obtenu. J'espère sincèrement que ce site web ajoutera de la valeur à mon entreprise et offrira une expérience positive et enrichissante aux clients.