

Nom : BENALI

Prénom : MANEL

# 

Table des matières

[1.Présentation personnelle en anglais 4](#_Toc139662374)

[2.Présentation du projet 5](#_Toc139662375)

[3.Cahier de charge : 7](#_Toc139662376)

[4.Gestion du projet 13](#_Toc139662377)

[5.UX/UI: 17](#_Toc139662378)

[23](#_Toc139662379)

[6.Conception de la base de donnée 25](#_Toc139662380)

[7.Conception de L’application : 30](#_Toc139662381)

[8-Conception multi-touches: 34](#_Toc139662382)

[9.SECURITE 35](#_Toc139662383)

[10.Démonstration de l’application 38](#_Toc139662384)

[11.Politique de test 38](#_Toc139662385)

[12.Veille technoligique 44](#_Toc139662386)

[13.Difficultés rencontrées 44](#_Toc139662387)

[14.CONSCLUSION 46](#_Toc139662388)

Liste des compétences du référentiel couvertes par le projet:

Une image contenant texte, capture d’écran, Police, document

Description générée automatiquement

# 1.Présentation personnelle en anglais

My name is Manel Ben Ali, and I am 33 years old.

I studied Human Resources and had a career in recruitment.

However, I decided to make a career change into the field of software development.

Currently, I am pursuing a diploma in Application Design and Development through an apprenticeship program while working at the Seine-Saint-Denis Departmental Council.

Today, I am here to present my end-of-year project, which involves creating a mobile application.

For this project, I have chosen an e-commerce website for selling makeup products.

I using Symfony 6.2 to sell beauty services. Visitors can add services to their cart and make payments through Stripe. However, there is currently no appointment management system, which I plan to add in the future.

Additionally, there is an admin panel for CRUD operations on all entities: Product (services), Category, Order, OrderItem (service quantity in an order), and User. These entities are stored in a MySQL relational database.

The frontend uses Bootstrap classes and CSS for styling, and HTML5 for website content structure with Twig as the default template engine for Symfony. No JavaScript framework was used for the frontend. The website is hosted on an Apache server with IONOS.

Figma was used for page mockups.

# 2.Présentation du projet

Le but de mon site internet est de permettre aux clients de commander des prestations de beauté en ligne. L'activité derrière ce site est celle d'un institut de beauté qui propose des prestations telles que des soins du visage, des massages, des épilations et des manucures. L'objectif est de rendre ces prestations plus accessibles et plus pratiques pour les clients qui peuvent désormais les commander en ligne, depuis leur domicile ou leur bureau.

Le choix de développer un site e-commerce pour cette activité est motivé par plusieurs facteurs. Tout d'abord, le marché des prestations de beauté est très concurrentiel et les instituts de beauté doivent se différencier pour attirer et fidéliser leur clientèle. Un site e-commerce permet de proposer une expérience utilisateur optimisée pour la commande de prestations en ligne, ce qui peut être un avantage concurrentiel pour l'institut de beauté.

Ensuite, le contexte sanitaire actuel a renforcé la nécessité pour les entreprises d'être présentes sur internet et de proposer des solutions de commande en ligne pour maintenir leur activité. Le site e-commerce que j'ai développé permet à l'institut de beauté de continuer à proposer ses prestations malgré les restrictions sanitaires en vigueur.

Enfin, le développement d'un site e-commerce pour cette activité offre également des avantages en termes de gestion. La fonctionnalité d'administration me permet de gérer facilement les produits, les commandes, les catégories et les utilisateurs, ce qui facilite la gestion au quotidien de l'institut de beauté. Le site peut également être utilisé pour générer des rapports sur les ventes, les clients et les produits, ce qui peut aider à optimiser l'activité de l'institut de beauté.

Les fonctionnalités clés de notre application de vente de prestations de beauté incluent :

1. Recherche et navigation des services : Les utilisateurs peuvent parcourir différentes catégories de services de beauté, afficher les détails des prestations, et consulter les tarifs des prestations ainsi que pouvoir payer les prestations directement en ligne via stripe.
2. Gestion des utilisateurs : Les utilisateurs peuvent créer un compte, se connecter, gérer leur profil, consulter leur historique de paiement.
3. Panier et paiement : Les utilisateurs peuvent ajouter des services à leur panier, vérifier les détails de leur commande et effectuer le paiement en utilisant des méthodes de paiement sécurisées.
4. Administration : Les administrateurs de l'application disposent d'un tableau de bord d'administration pour gérer les utilisateurs, les professionnels, les services, les paiements.

Structure de base du site :

1. Page d'accueil :
   * Présentation de l'application et de ses avantages.
   * Mise en avant des offres et promotions spéciales.
   * Liens vers les différentes catégories de services de beauté.
2. Pages de catégories de services :
   * Une page dédiée à chaque catégorie de services (soins du visage, massages, manucures, etc.).
   * Affichage des prestations disponibles dans la catégorie sélectionnée.
   * Filtres de recherche pour affiner les résultats en fonction des préférences des utilisateurs (prix, localisation, disponibilité, etc.).
3. Page de détails de prestation :
   * Affichage détaillé des informations sur une prestation spécifique (description, durée, prix, photos, avis des clients, etc.).
   * Possibilité pour les utilisateurs de sélectionner une date et un créneau horaire pour la réservation.
4. Panier d'achat :
   * Récapitulatif des prestations sélectionnées par l'utilisateur.
   * Calcul du prix total des prestations.
   * Possibilité de modifier les prestations sélectionnées ou de supprimer des articles du panier.
   * Bouton de paiement pour procéder au règlement.
5. Page de paiement :
   * Formulaire de paiement sécurisé où les utilisateurs peuvent saisir les informations de leur carte de crédit ou utiliser d'autres méthodes de paiement.
   * Validation des informations de paiement et traitement de la transaction.
6. Profil utilisateur :
   * Page permettant aux utilisateurs de gérer leurs informations personnelles (nom, adresse, coordonnées, etc.).
   * Historique des commandes passées.
   * Possibilité de laisser des avis et des évaluations pour les prestations reçues.

7.Tableau de bord administrateur :

* + Interface réservée aux administrateurs pour gérer les clients, les services et les paiements.

Les prestations attendues pour le site de vente de prestations de beauté peuvent inclure :

* Liste des prestations : Affichage d'une liste complète des prestations de beauté disponibles, telles que les soins du visage, les manucures, les pédicures, les coiffures, etc. Chaque prestation devrait être accompagnée d'une description détaillée.
* Recherche peut être effectuée à l'aide de mots-clés ou d'options de filtrage prédéfinies.
* Profils des prestataires : Afficher les profils détaillés des prestataires de services.
* Paiement en ligne : Intégrer un système de paiement sécurisé pour permettre aux utilisateurs de régler les prestations réservées en ligne

# 3.Cahier de charge :

Le but de mon site internet est de permettre aux clients de commander des prestations de beauté en ligne.

L’activité derrière ce site est celle d'un institut de beauté qui propose des prestations telles que des soins du visage, des épilations et des manucures. L'objectif est de rendre ces prestations plus accessibles et plus pratiques pour les clients qui peuvent désormais les commander en ligne, depuis leur domicile ou leur bureau.

Le choix de développer un site e-commerce pour cette activité est motivé par plusieurs facteurs. Tout d'abord, le marché des prestations de beauté est très concurrentiel et les instituts de beauté doivent se différencier pour attirer et fidéliser leur clientèle. Un site e-commerce permet de proposer une expérience utilisateur optimisée pour la commande de prestations en ligne, ce qui peut être un avantage concurrentiel pour l’institut de beauté.

Ensuite, le contexte sanitaire actuel a renforcé la nécessité pour les entreprises d'être présentes sur internet et de proposer des solutions de commande en ligne pour maintenir leur activité

Enfin, le développement d'un site e-commerce pour cette activité offre également des avantages en termes de gestion. La fonctionnalité d'administration me permet de gérer facilement les produits, les commandes, les catégories et les utilisateurs, ce qui facilite la gestion au quotidien de l’institut de beauté. Le site peut également être utilisé pour générer des rapports sur les ventes, les clients et les produits, ce qui peut aider à optimiser l’activité de l’institut de beauté.

L’objectif du site internet est de proposer des prestations de beauté de haute qualité à des prix compétitifs. Nous voulons offrir à nos clients des prestations de beauté de qualité supérieure à celles proposées par nos concurrents, tout en maintenant des prix compétitifs pour rester attractifs sur le marché.

Nous avons travaillé en étroite collaboration avec des professionnels de la beauté pour sélectionner les meilleures prestations de beauté à proposer sur notre site internet. Nous avons choisi des prestations de qualité supérieure pour offrir à nos clients une expérience de beauté inoubliable, tout en veillant à ce que les prix restent abordables pour nos clients.

Nous avons également mis en place une stratégie de prix compétitive pour nous démarquer de nos concurrents et offrir à nos clients un rapport qualité-prix optimal. Nous sommes convaincus que nos prix compétitifs nous permettront de nous différencier sur le marché et d'attirer de nouveaux clients.

L’objectif est de proposer des prestations de beauté de haute qualité à des prix compétitifs pour offrir à nos clients une expérience de beauté inoubliable tout en restant attractifs sur le marché. Nous avons travaillé en étroite collaboration avec des professionnels de la beauté pour sélectionner les meilleures prestations de beauté à proposer sur notre site internet et nous avons mis en place une stratégie de prix compétitive pour nous différencier de nos concurrents.

Genèse de l’idée

L'idée de créer un site e-commerce pour vendre des prestations de beauté m'est venue après avoir constaté la croissance du marché de la beauté et la demande croissante pour des prestations de qualité à des prix abordables. Je suis passionné par la beauté et j'ai réalisé que de nombreuses personnes avaient des difficultés à trouver des prestations de qualité à des prix abordables, surtout dans les grandes villes.

Après avoir mené une étude de marché approfondie et identifié un besoin clair sur le marché, j'ai décidé de lancer mon propre site e-commerce pour offrir des prestations de beauté de qualité à des prix compétitifs. Mon site e-commerce a été conçu pour offrir une expérience utilisateur optimisée pour la commande de prestations de beauté en ligne, en offrant une large gamme de prestations de qualité, une livraison rapide et un service client de haute qualité.

Population cible

La population cible de mon site internet est constituée de personnes qui cherchent à commander des prestations de beauté en ligne, de manière pratique et sécurisée. Nous ciblons en particulier les personnes qui ont un mode de vie actif et qui n'ont pas nécessairement le temps de se rendre dans un institut de beauté pour commander des prestations. Nous ciblons également les personnes qui recherchent des solutions pratiques pour commander des prestations de beauté à domicile ou sur leur lieu de travail.

Notre site e-commerce est conçu pour répondre aux besoins de cette population cible en proposant une expérience utilisateur optimisée pour la commande de prestations de beauté en ligne. Nous offrons une large gamme de prestations de beauté de haute qualité, avec des descriptions détaillées et des images pour chaque produit. Nous proposons également des prix compétitifs et une livraison rapide pour offrir une expérience client de haute qualité.

Nous ciblons également les personnes soucieuses de leur apparence physique et de leur bien-être. Notre site e-commerce propose des prestations de beauté de haute qualité qui répondent à ces besoins. Nous offrons des produits de soins du visage, des massages relaxants, des épilations et des manucures pour aider nos clients à prendre soin de leur apparence physique et de leur bien-être.

La population cible de notre site internet est constituée de personnes qui cherchent à commander des prestations de beauté en ligne, de manière pratique et sécurisée. Nous ciblons en particulier les personnes qui ont un mode de vie actif, qui cherchent des solutions pratiques pour commander des prestations de beauté à domicile ou sur leur lieu de travail, ainsi que les personnes soucieuses de leur apparence physique et de leur bien-être.

Besoin du marché:

Mon site internet comble un besoin important sur le marché de la beauté, celui d'offrir une plateforme en ligne simple et efficace pour commander des prestations de beauté de qualité à des prix abordables. Nous avons identifié que de nombreuses personnes avaient des difficultés à trouver des prestations de beauté de qualité à des prix abordables, surtout dans les grandes villes où les prix peuvent être élevés.

Notre site internet propose une large gamme de prestations de beauté de qualité, sélectionnées avec soin en collaboration avec des professionnels de la beauté, pour offrir à nos clients une expérience de beauté inoubliable. Nous avons travaillé dur pour garantir que nos prix restent compétitifs, tout en offrant des prestations de qualité supérieure à celles de nos concurrents.

Notre plateforme en ligne est facile à utiliser, intuitive et conviviale. Les clients peuvent parcourir notre sélection de prestations de beauté, les ajouter à leur panier et les commander en quelques clics seulement. Nous avons également mis en place une fonction de recherche pour permettre aux clients de trouver rapidement les prestions de beautés.

En outre, notre plateforme de paiement en ligne avec Stripe garantit la sécurité totale des transactions, offrant à nos clients une tranquillité d'esprit supplémentaire lorsqu'ils commandent sur notre site internet.

Notre site internet comble un besoin important sur le marché de la beauté en offrant une plateforme en ligne simple et efficace pour commander des prestations de beauté de qualité à des prix abordables. Nous avons travaillé en étroite collaboration avec des professionnels de la beauté pour sélectionner les meilleures prestations de beauté à proposer sur notre site internet, tout en offrant une expérience utilisateur optimale et une sécurité totale pour les transactions en ligne. beauté qu'ils recherchent.

Concurrence sur le marché

Le marché des prestations de beauté est très concurrentiel et il existe de nombreux instituts de beauté qui proposent des prestations similaires à celles que nous offrons sur notre site e-commerce. Le domaine de l'activité de notre site internet est donc celui de la beauté, avec des prestations telles que des soins du visage, des massages, des épilations et des manucures.

Il y a plusieurs raisons pour lesquelles il y a des concurrents sur ce marché. Tout d'abord, le marché de la beauté est en croissance constante, avec une demande croissante pour les prestations de beauté de haute qualité et abordables. Les consommateurs sont donc enclins à chercher des instituts de beauté qui proposent des prestations répondant à leurs besoins et à leur budget.

Ensuite, la concurrence sur ce marché est également liée à la croissance du commerce en ligne. De plus en plus de consommateurs recherchent des solutions pratiques pour commander des prestations de beauté en ligne, ce qui a conduit à une augmentation des sites e-commerce spécialisés dans la beauté. Les clients ont donc le choix entre différents sites pour commander des prestations de beauté en ligne.

Enfin, la concurrence sur ce marché est également liée à la différenciation entre les instituts de beauté. Les instituts de beauté qui réussissent sont souvent ceux qui proposent des prestations de haute qualité, un service client irréprochable, des prix compétitifs et des promotions attractives. Les clients sont donc enclins à choisir les instituts de beauté qui répondent le mieux à leurs besoins et à leurs attentes.

Le marché des prestations de beauté est très concurrentiel et il existe de nombreux instituts de beauté qui proposent des prestations similaires à celles que nous offrons sur notre site e-commerce. La concurrence sur ce marché est liée à la croissance de la demande pour les prestations de beauté, la croissance du commerce en ligne et la différenciation entre les instituts de beauté. Nous sommes donc conscients des défis liés à ce marché et nous mettons tout en œuvre pour offrir des prestations de haute qualité, un service client irréprochable et des prix compétitifs pour se démarquer de la concurrence.

Fonctionnalité du projet :

* Fonction primaire du projet

Le premier objectif de mon site internet est de proposer une plateforme conviviale et efficace pour permettre aux clients de commander des prestations de beauté en ligne en toute simplicité. Nous voulons offrir une expérience utilisateur optimale pour que les clients puissent facilement parcourir notre sélection de prestations de beauté, les ajouter à leur panier et les commander en quelques clics seulement.

Nous avons conçu notre site internet pour répondre aux besoins de notre population cible, à savoir les personnes qui cherchent à commander des prestations de beauté de manière pratique et sécurisée. Nous avons donc mis en place une interface conviviale et intuitive, avec une navigation facile et une fonction de recherche pour permettre aux clients de trouver rapidement les prestations de beauté qu'ils recherchent.

Nous avons également mis en place un système de commande en ligne simple et efficace, avec une fonction de panier pour permettre aux clients de visualiser les produits qu'ils ont choisis, les quantités, les prix et le montant total de leur commande. Les clients peuvent ensuite payer en ligne grâce à Stripe pour finaliser leur commande en toute sécurité et en toute confiance.

Notre premier objectif est donc d'offrir une expérience utilisateur optimale pour la commande de prestations de beauté en ligne, en proposant une plateforme conviviale, une large gamme de prestations de beauté de qualité, des prix compétitifs et une livraison rapide. Nous voulons que nos clients se sentent en confiance lorsqu'ils commandent sur notre site et que leur expérience de commande soit aussi agréable que possible.

Notre premier objectif est de proposer une plateforme conviviale et efficace pour permettre aux clients de commander des prestations de beauté en ligne en toute simplicité, tout en offrant une expérience utilisateur optimale et une sécurité totale. Nous avons conçu notre site internet en tenant compte des besoins de nos clients et nous sommes fiers de proposer une plateforme de qualité pour répondre à ces besoins.

Une des fonctionnalités les plus importantes de mon site e-commerce est la possibilité pour les clients de commander des prestations de beauté en ligne et de les ajouter à leur panier. Cette fonctionnalité permet aux clients de parcourir facilement les différentes prestations proposées sur notre site et de les ajouter à leur panier en quelques clics. Cela facilite grandement la commande de prestations de beauté pour nos clients, tout en offrant une expérience utilisateur intuitive et agréable.

La fonctionnalité de commande de prestations de beauté en ligne sur notre site est également accompagnée de la possibilité de visualiser les produits dans leur panier. Les clients peuvent facilement voir les produits qu'ils ont choisis, les quantités, les prix et le montant total de leur commande. Cela offre une transparence totale sur les produits choisis et le coût total, ce qui contribue à instaurer un climat de confiance avec nos clients.

Une autre fonctionnalité importante de notre site est la possibilité de payer en ligne grâce à Stripe. Les clients peuvent choisir leur mode de paiement préféré et entrer leurs informations de carte bancaire pour finaliser leur commande en toute sécurité. Nous utilisons la technologie de sécurité SSL pour garantir que les informations de paiement des clients sont protégées et sécurisées, offrant ainsi une expérience utilisateur sans souci et sans risque.

* Fonction Principale du projet

| **Rôle** | **Fonctionnalités** |
| --- | --- |
| Visiteur | - Accéder aux catégories de produits  - Effectuer des recherches  - Consulter les fiches produits  - Créer un compte utilisateur  - Contacter le service clientèle  -Ajouter les produits au panier |
| Client | - Suivre l'état de la commande  - Ajouter des produits  - Payer en toute sécurité  - Consulter l'historique des commandes |
| Administrateur | - Gérer les catégories de produits  - Gérer les moyens de paiement  - Supprimer ou modifier un compte client |

Les contraintes techniques

Pour assurer le bon fonctionnement de notre projet d'application de vente de prestations de beauté, il est essentiel de considérer les contraintes techniques.

**Voici quelques aspects à prendre en compte:**

Langage de programmation: Utiliser PHP pour le développement du back-end, en utilisant le framework Symfony.

Base de données : Utiliser MySQL comme système de gestion de base de données. Vous pouvez gérer votre base de données à l'aide de l'outil PHPMyAdmin, qui fournit une interface conviviale pour administrer la **base** de données MySQL.

**Sécurité** : Mettre en place des mesures de sécurité pour protéger les données sensibles, telles que les informations personnelles des utilisateurs et les données de paiement. Utiliser les fonctionnalités de sécurité fournies par Symfony, telles que l'authentification utilisateur, les autorisations d'accès, la validation des formulaires, etc. il faut également à échapper et à valider correctement les données entrantes pour prévenir les attaques par injection SQL.

**API RESTful** : Concevez l’ application pour fournir une API RESTful pour permettre aux clients d'accéder aux services de l’ application à partir de Angular.

Utiliser les composants Symfony pour créer et gérer API RESTful.

**Intégration de paiement:** Intégrez une passerelle de paiement sécurisée pour permettre aux utilisateurs de payer les prestations de beauté en ligne.

Des services populaires tels que Stripe offrent des API faciles à intégrer pour le traitement des paiements.

**Intégration avec Angular :** S’assurer d'avoir une API claire et documentée pour communiquer avec l’application Angular.

Développer les fonctionnalités frontales nécessaires, telles que l'affichage des prestations de beauté, le panier d'achat, le suivi des commandes, etc.

**Tests unitaires et fonctionnels :** Écrire des tests unitaires pour vérifier la qualité et le bon fonctionnement du code.

Utiliser des outils tels que PHPUnit pour tester les parties back-end, et des outils tels que Protractor ou Karma pour les tests front-end.

**Performance :** Optimiser les requêtes de base de données pour améliorer les performances de l’ application. Utiliser des index appropriés, mettez en cache les données fréquemment utilisées et utilisez des techniques de mise en mémoire tampon pour minimiser les temps de chargement.

**Déploiement:** Mettre en place un environnement de déploiement approprié pour l’ application, en utilisant des outils tels que Docker pour la conteneurisation et déployer l’application sur une plateforme d'hébergement adaptée à Symfony et Angular, comme AWS, Digital Ocean, Heroku, etc.

**Planification et gestion du projet:** Jai utilisé une methodologie de gestion de projet telles que Kanban pour une gestion agile du projet avec l’outil Trello.

**Contrôle de version :** J’ai utilisé un système de contrôle de version : Git pour gérer le code source du projet.

J’ai configuré un dépôt Git pour suivre les modifications, faciliter la collaboration et permettre le retour en arrière si nécessaire.

4.Gestion du projet

Planning et suivi

Pour la création de mon application de vente de prestations de beauté, j'ai choisi d'adopter la méthode Kanban pour la gestion de projet. J'ai utilisé l'outil Trello pour mettre en pratique cette approche.

L'utilisation de la méthode Kanban m'a permis de visualiser clairement les différentes étapes du projet et de gérer efficacement les tâches associées.

J'ai créé des tableaux Trello représentant les différentes phases du développement de l'application, telles que la conception, le développement, les tests et le déploiement.

Dans chaque tableau, j'ai créé des listes pour représenter les étapes spécifiques de chaque phase. Par exemple, j'ai eu des listes telles que "À faire", "En cours" et "Terminé" pour le développement des fonctionnalités en utilisant PHP Symfony et Angular.

J'ai créé des cartes pour chaque tâche à accomplir, en les décrivant de manière détaillée et en y ajoutant des informations telles que les dépendances, les priorités et les échéances. J'ai également utilisé des étiquettes pour catégoriser les tâches en fonction de leur nature, par exemple, "Front-end", "Back-end", "Intégration", etc.

En déplaçant les cartes à travers les listes, j'ai pu suivre l'évolution des tâches et visualiser facilement les tâches en cours, les tâches à venir et celles qui étaient terminées. Cela m'a aidé à prioriser les tâches, à gérer les délais et à maintenir une visibilité claire sur l'avancement du projet.

La méthode Kanban m'a également permis de collaborer efficacement avec mon équipe. Chaque membre de l'équipe avait accès aux tableaux Trello, ce qui leur permettait de visualiser les tâches assignées, de les mettre à jour et de communiquer sur l'avancement.

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

Environnement humain et technique

* CHOIX DES LANGAGES:

Les langages de base que j’ai choisis pour le projet sont:

* PHP 8.0
* HTML 5
* CSS 3
* CHOIX DES FRAMEWORKS:

J’ai décidé de partir sur trois frameworks :

* Un framework PHP pour le côté serveur : Slim (version Symfony6.2)
* Un framework Javascript pour le côté client : Angular JS (version 15.2.9)
* Un « framework » pour le design : Bootstrap (version 5.3.0)

**LOGICIELS ET AUTRES OUTILS**

* EDITEUR DE TEXTE : VS CODE
* SERVEUR WEB : APACHE 2.4
* BASE DE DONNEES : MYSQL 8.0.31

Concernant mon choix actuel des frameworks, voici quelques explications et avantages :

**Framework PHP - Symfony :**

Symfony est un framework PHP populaire et mature qui offre une architecture solide et modulaire. Il suit les meilleures pratiques de développement web et fournit une grande flexibilité et une abstraction puissante pour gérer les aspects techniques.

Symfony dispose d'une communauté active et d'une documentation exhaustive, ce qui facilite l'apprentissage et la résolution des problèmes.

Symfony dispose également de nombreux composants prêts à l'emploi pour gérer des tâches courantes, comme l'authentification, la gestion des formulaires, l'accès à la base de données, etc.

**Framework JavaScript - AngularJS :**

AngularJS est un framework JavaScript développé par Google, qui offre une approche structurée et complète pour le développement d'applications web complexes.

AngularJS utilise le concept de liaison de données bidirectionnelle, ce qui facilite la gestion des données entre les composants de votre application.

Il offre une architecture modulaire et encourage les bonnes pratiques de développement, telles que l'injection de dépendances, la séparation des préoccupations, etc.

AngularJS dispose également d'une large communauté de développeurs, de nombreuses ressources et d'une documentation abondante pour faciliter le développement et la résolution des problèmes.

Objectifs de qualité

Les exigences de performance pour l'application de vente de prestations de beauté peuvent inclure les éléments suivants :

1. Temps de chargement rapide : L'application doit être optimisée pour offrir un temps de chargement rapide des pages et des fonctionnalités. Cela implique d'optimiser le code, de minimiser les requêtes réseau et d'utiliser des techniques de mise en cache appropriées.
2. Réactivité de l'interface utilisateur : L'application doit offrir une expérience utilisateur fluide et réactive, en évitant les délais et les ralentissements lors de l'interaction avec les différentes fonctionnalités. Cela peut être réalisé en optimisant les performances du frontend avec Angular, en utilisant des techniques de chargement asynchrone et en optimisant les requêtes réseau.
3. Scalabilité : L'application doit être capable de gérer un nombre croissant d'utilisateurs et de charges de travail sans compromettre ses performances. Cela peut être réalisé en utilisant des techniques de mise en cache, en optimisant les requêtes de base de données et en mettant en place une architecture évolutive et extensible.
4. Gestion efficace de la base de données : La base de données doit être optimisée pour gérer efficacement les requêtes et les opérations de lecture/écriture, afin d'éviter les goulots d'étranglement et les temps de latence élevés. Cela peut inclure l'indexation appropriée des tables, l'optimisation des requêtes SQL et l'utilisation de techniques de mise en cache.
5. Temps de réponse des API : Les API utilisées pour la communication entre le frontend et le backend doivent offrir des temps de réponse rapides, afin de minimiser les délais d'attente pour les requêtes et les mises à jour des données. Cela peut être réalisé en optimisant les performances du backend avec Symfony, en utilisant des techniques de mise en cache et en optimisant les requêtes et les réponses JSON.
6. Gestion de la charge : L'application doit être capable de gérer efficacement les pics de charge, tels que les périodes de forte affluence ou les moments de promotion spéciale. Cela peut être réalisé en utilisant des techniques de mise en cache, en mettant en place une infrastructure de serveurs évolutifs et en optimisant les performances du code.
7. Suivi des performances : Il est important de mettre en place des outils de suivi des performances tels que des journaux, des outils de surveillance et des mesures de performances pour identifier les goulots d'étranglement, les problèmes de performances et les opportunités d’optimisation.

En respectant ces exigences de performance, l'application de vente de prestations de beauté pourra offrir une expérience utilisateur fluide, réactive et sans délai, même lors de périodes de forte charge.

# 5.UX/UI:

Design du site

Le design du site : de vente de prestations de beauté devrait être à la fois attractif et fonctionnel pour offrir une expérience utilisateur agréable. Voici quelques éléments clés à prendre en compte pour le design du site :

1. Interface utilisateur intuitive : Assurez-vous que l'interface utilisateur est conviviale et facile à naviguer. Utilisez une structure de menu claire et des liens bien organisés pour permettre aux utilisateurs de trouver facilement les informations dont ils ont besoin.
2. Mise en page équilibrée : Créez une mise en page équilibrée et aérée qui permet aux visiteurs de se concentrer sur les informations importantes. Utilisez des espaces vides stratégiquement pour améliorer la lisibilité et la clarté du contenu.
3. Choix des couleurs : Sélectionnez une palette de couleurs attrayante et cohérente qui représente l'identité visuelle de votre marque. Les couleurs doivent être harmonieuses et mettre en valeur les éléments clés du site.
4. Typographie lisible : Utilisez des polices de caractères lisibles et adaptées au contenu. Assurez-vous que la taille de police est suffisamment grande pour une lecture facile sur tous les appareils.
5. Images et visuels de qualité : Intégrez des images de haute qualité qui reflètent l'univers de la beauté et des prestations proposées. Utilisez des visuels attrayants pour illustrer les produits ou les services offerts.
6. Responsive Design : Assurez-vous que le site est responsive, c'est-à-dire qu'il s'adapte automatiquement à différentes tailles d'écran. Cela garantit une expérience utilisateur optimale sur les ordinateurs de bureau, les tablettes et les smartphones.
7. Appels à l'action clairs : Utilisez des boutons et des appels à l'action clairs pour inciter les visiteurs à effectuer des actions spécifiques, comme s'inscrire, acheter un produit ou prendre rendez-vous.
8. Optimisation des performances : Veillez à ce que le site se charge rapidement en optimisant les images, en utilisant une structure de code propre et en minimisant les requêtes au serveur.
9. Accessibilité : Assurez-vous que le site est accessible à tous les utilisateurs, y compris ceux ayant des besoins spécifiques. Respectez les normes d'accessibilité Web pour garantir une expérience inclusive pour tous.
10. Cohérence et branding : Maintenez une cohérence visuelle et de branding sur l'ensemble du site. Utilisez les mêmes éléments de design tels que le logo, les couleurs et les polices pour renforcer l'identité de votre marque.

En combinant ces éléments de design, vous pouvez créer un site attrayant, convivial et professionnel pour la vente de prestations de beauté. N'hésitez pas à prendre en compte les préférences et les attentes de votre public cible pour personnaliser davantage le design du site.

Description de la charte graphique

* Création du logo

Une image contenant écriture manuscrite, Police, typographie, conception

Description générée automatiquement

Le logo se compose du texte "Gystesy" stylisé avec une police de caractères élégante et lisible.

Le texte est de couleur "#C29E5B", un ton chaud et doré qui ajoute une touche de sophistication et de chaleur au logo.​

Au-dessus ou à côté du texte, se trouve un cercle délicat. Le cercle est dessiné de manière à être lisse et équilibré. Il est de la meme couleur que le texte.​

À l'intérieur du cercle,les petites fleurs sont délicatement dessinées. Ces fleurs sont stylisées avec des pétales détaillés et des tiges fines. Elles sont représentées avec des couleurs harmonieuses qui s'accordent avec la couleur du texte, Les fleurs ajoutent une touche de féminité, de beauté et de nature au logo.​

L'ensemble du design du logo vise à transmettre une image d'élégance, de douceur et de raffinement. La combinaison du texte stylisé avec la couleur dorée et du cercle avec les fleurs crée un équilibre visuel et attire l'attention de manière subtile. Ce design s'est particulièrement concentré sur une marque dans l'industrie de la beauté, de la mode ou d'autres domaines où une esthétique féminine et délicate est recherchée

* Typographie :

Pour notre site internet, j'ai choisi une typographie simple et facile à lire pour offrir une expérience utilisateur agréable et cohérente. J'ai utilisé la police de caractères "Open Sans" pour le texte principal, qui est une police sans serif propre et élégante qui offre une grande lisibilité.

J'ai également utilisé la police de caractères "Montsserat" pour les titres et les en-têtes de page. Cette police sans serif offre un design moderne et élégant tout en étant facile à lire et à naviguer pour les visiteurs.

En utilisant ces deux polices de caractères, j'ai créé une présentation cohérente à travers toutes les pages de notre site internet. Cela permet aux visiteurs de naviguer facilement sur le site et de trouver les informations qu'ils cherchent sans difficulté.

Enfin, j'ai utilisé des tailles de police variées pour mettre en valeur les éléments de la page. J'ai utilisé des tailles plus grandes pour les titres et les en-têtes de page, et des tailles plus petites pour le texte principal et les descriptions. Cela permet de créer une hiérarchie visuelle claire pour les visiteurs et de les guider à travers le site de manière intuitive.

La typographie simple et facile à lire utilisée sur notre site internet offre une expérience utilisateur agréable et cohérente, tout en permettant aux visiteurs de naviguer facilement sur le site et de trouver les informations qu'ils cherchent sans difficulté.

* Palette de couleur et identité

Les couleurs principales utilisée sur le site sont le blanc et le gris, qui est utilisé pour mettre en avant les éléments importants tels que les boutons de commande et les liens hypertexte. J'ai également utilisé cette couleur pour souligner les bordures et les blocs de texte, créant ainsi une présentation cohérente à travers toutes les pages.

La palette de couleurs douces et harmonieuses utilisée sur notre site internet permet de créer une ambiance agréable et relaxante pour les visiteurs, tout en offrant une présentation esthétique et cohérente à travers toutes les pages.

* Images et médias :

Utilisez les classes d'images de Bootstrap pour styliser les images de produits, les photos de profil ou les bannières.

Utilisez les classes de Bootstrap pour intégrer des vidéos de manière responsive et les rendre compatibles avec différents appareils.

* Mise en page et hiérarchie visuelle :

Utilisez le système de grille de Bootstrap pour créer une mise en page responsive et harmonieuse.

Utilisez les classes de Bootstrap pour définir la taille, la couleur et l'espacement des titres, des boutons et des autres éléments visuels afin de créer une hiérarchie visuelle claire et attrayante.

* Images et médias :

Utilisez les classes d'images de Bootstrap pour styliser les images de produits, les photos de profil ou les bannières.

Utilisez les classes de Bootstrap pour intégrer des vidéos de manière responsive et les rendre compatibles avec différents appareils.

* Consistance et cohérence :

Respectez les conventions de Bootstrap pour garantir une cohérence visuelle dans l'ensemble du site.

Évitez de surcharger le site avec trop de styles personnalisés qui pourraient compromettre la cohérence et la convivialité offertes par Bootstrap.

Zoning

Une image contenant capture d’écran, électroménager, texte, Rectangle

Description générée automatiquement

Wireframe

Une image contenant texte, capture d’écran, Police, conception

Description générée automatiquement

Maquette :

Pour maquetter les pages de notre site internet, j'ai utilisé l'outil Figma, qui est une plateforme de conception collaborative basée sur le cloud. J'ai commencé par créer des maquettes brutes de chaque page, en utilisant les éléments de design de base tels que des blocs de texte, des images de remplissage et des espaces réservés pour les images réelles.

Une fois les maquettes brutes créées, j'ai travaillé à affiner le design en ajoutant des éléments de style tels que des couleurs, des polices de caractères et des icônes. J'ai également utilisé des grilles pour aligner les éléments de la page et assurer une présentation cohérente à travers toutes les pages.

Enfin, j'ai partagé les maquettes avec l'équipe pour obtenir des commentaires et des suggestions, et j'ai apporté les modifications nécessaires pour améliorer la convivialité et l'expérience utilisateur globale.

Dans l'ensemble, l'utilisation de Figma pour maquetter nos pages a été très bénéfique car cela nous a permis de créer des designs cohérents et professionnels, tout en permettant une collaboration facile entre les membres de l'équipe. Les maquettes ont également aidé à visualiser le site internet final avant même que le développement ne commence, ce qui a permis d'optimiser le processus de développement et d'assurer la cohérence entre les designs et la mise en œuvre finale.

Une image contenant texte, Visage humain, capture d’écran, sourire

Description générée automatiquement

## Une image contenant texte, Visage humain, capture d’écran, femme Description générée automatiquement

Responsive design

Pour réaliser l'interface web de notre site internet, j'ai utilisé le framework Symfony 5.4 pour PHP, qui a fourni des fonctionnalités avancées pour le développement web et la création de sites web adaptables. En utilisant les composants de Symfony tels que le moteur de templates Twig, j'ai pu générer des pages web HTML5 dynamiques et adaptables pour nos visiteurs.

Pour rendre notre site internet adaptable, j'ai utilisé les classes de Bootstrap, qui fournissent un système de grille flexible pour organiser les éléments de la page en fonction de la taille de l'écran. Cela permet aux visiteurs d'avoir une expérience utilisateur optimale, quel que soit l'appareil qu'ils utilisent pour accéder au site internet.

En plus de Bootstrap, j'ai également utilisé des techniques CSS pour rendre notre site internet adaptable. J'ai utilisé des media queries pour détecter la taille de l'écran du visiteur et appliquer les styles appropriés en fonction de la taille de l'écran. J'ai également utilisé des techniques CSS pour cacher ou afficher certains éléments en fonction de la taille de l'écran, afin de garantir une expérience utilisateur optimale.

Enfin, j'ai également utilisé des outils de développement pour tester notre site internet sur une variété d'appareils et de navigateurs, afin de m'assurer que l'interface était adaptative sur tous les types d'appareils et de navigateurs.

Dans l'ensemble, la combinaison de Symfony, Twig, Bootstrap et des techniques CSS a permis de créer une interface web dynamique et adaptative pour notre site internet. Cela a permis d'offrir une expérience utilisateur optimale pour nos visiteurs, quel que soit l'appareil qu'ils utilisent pour accéder au site.

Le Responsive Design est une approche de conception Web qui vise à créer des sites Web qui s'adaptent de manière fluide à différents appareils et tailles d'écran. Les media queries sont un élément clé du Responsive Design, car ils permettent de définir des règles CSS spécifiques en fonction des caractéristiques de l'appareil sur lequel le site est affiché.

Les media queries sont utilisées pour définir des points de rupture (breakpoints) à partir desquels les styles CSS spécifiques seront appliqués. Cela signifie que vous pouvez créer des règles CSS différentes pour différentes tailles d'écran.

Voici un exemple de media query en CSS :

css

Copy code

@media (max-width: 768px) { /\* Styles à appliquer pour les écrans de largeur maximale de 768px \*/ } @media (min-width: 768px) and (max-width: 1024px) { /\* Styles à appliquer pour les écrans dont la largeur est comprise entre 768px et 1024px \*/ } @media (min-width: 1025px) { /\* Styles à appliquer pour les écrans dont la largeur est supérieure à 1024px \*/ }

Dans cet exemple, nous avons défini trois media queries. La première s'appliquera aux écrans ayant une largeur maximale de 768px, la deuxième s'appliquera aux écrans dont la largeur est comprise entre 768px et 1024px, et la troisième s'appliquera aux écrans dont la largeur est supérieure à 1024px.

À l'intérieur de chaque media query, vous pouvez définir les styles CSS spécifiques que vous souhaitez appliquer pour cette taille d'écran particulière. Cela peut inclure des ajustements de la mise en page, des tailles de police, des marges, des images ou tout autre élément que vous souhaitez modifier en fonction de la taille de l'écran.

En utilisant les media queries, vous pouvez créer une expérience utilisateur optimale en adaptant votre site Web à différentes résolutions et tailles d'écran, que ce soit pour les ordinateurs de bureau, les tablettes ou les téléphones mobiles. Cela permet d'offrir une expérience cohérente et conviviale, quel que soit l'appareil utilisé pour accéder au site.

6.Conception de la base de donnée

Accès aux données

Dans le cadre de mon projet, j'ai utilisé des composants d'accès aux données pour interagir avec la base de données et récupérer les informations nécessaires pour afficher les pages du site. J'ai utilisé Doctrine, un ORM (Object Relational Mapping) pour la communication entre la base de données et le code PHP.

Doctrine est un outil puissant qui permet de simplifier l'accès à la base de données en utilisant des objets au lieu d'écrire des requêtes SQL brutes. Avec Doctrine, j'ai défini des entités qui correspondent aux tables de la base de données et j'ai utilisé ces entités pour créer, lire, mettre à jour et supprimer des données. Cela m'a permis de simplifier considérablement le code et de rendre le processus de gestion des données plus efficace.

J'ai également utilisé des repositories pour gérer les opérations de lecture et d'écriture de données. Les repositories sont des classes qui encapsulent la logique d'accès aux données et fournissent une interface claire et cohérente pour interagir avec la base de données. Grâce à l'utilisation de ces repositories, j'ai pu maintenir un code clair et facilement compréhensible.

Enfin, j'ai également utilisé des services pour gérer les opérations complexes liées aux données. Les services sont des classes qui encapsulent la logique métier de l'application et fournissent des fonctionnalités spécifiques. J'ai utilisé des services pour effectuer des opérations telles que la création de nouvelles entités, l'envoi d'e-mails et la validation des données entrées par l'utilisateur.

L'utilisation de composants d'accès aux données tels que Doctrine, repositories et services a grandement simplifié le développement de mon projet en permettant de gérer les données de manière claire et efficace.

* Modèle de base de données

Pour la création de la base de données de mon site internet, j'ai suivi une méthodologie en plusieurs étapes. Tout d'abord, j'ai défini les entités qui seraient nécessaires pour mon site, à savoir Product (les prestations), Category (une catégorie de produits), Order (commande), OrderItem (produit dans une commande avec leur quantité) et User. Ensuite, j'ai créé les schémas de chaque entité en utilisant le langage de modélisation UML pour définir les champs et les relations entre les différentes tables.

Après cela, j'ai utilisé l'outil Doctrine, fourni par Symfony, pour générer les classes d'entité correspondantes à partir de mes schémas UML. Ces classes ont ensuite été utilisées pour créer les tables de la base de données MySQL.

Ensuite, j'ai défini les contraintes d'intégrité de la base de données pour garantir l'intégrité des données stockées. J'ai également créé des index pour optimiser les requêtes sur la base de données.

Une fois la structure de la base de données établie, j'ai rempli les tables avec des données d'exemple pour tester le fonctionnement de mon site. J'ai également utilisé des requêtes SQL pour extraire et analyser des données de la base de données pendant le développement de mon site.

Enfin, j'ai effectué des tests pour vérifier que toutes les fonctionnalités du site liées à la base de données fonctionnaient correctement. J'ai également effectué des tests de performance pour m'assurer que la base de données pouvait gérer de grandes quantités de données sans affecter les performances du site.

Ma méthodologie de création de la base de données a suivi une approche structurée et étape par étape, qui a permis d'assurer l'intégrité des données stockées et de garantir la performance de la base de données pour mon site internet.

Mon site internet utilise une base de données relationnelle MySQL pour stocker plusieurs entités, chacune représentée par une table dans la base de données.

Tout d'abord, il y a l'entité Product, qui représente les différentes prestations de beauté que les visiteurs peuvent commander sur le site. Cette entité est stockée dans une table appelée product, qui contient des informations telles que le nom de la prestation, sa description, son prix, etc.

Ensuite, il y a l'entité Category, qui représente les différentes catégories de produits auxquelles les produits peuvent être associés. Cette entité est stockée dans une table appelée category, qui contient des informations telles que le nom de la catégorie, sa description, etc. Chaque produit est associé à une seule catégorie.

L'entité Order représente une commande passée par un utilisateur, et est stockée dans une table appelée order. Cette table contient des informations sur la commande, telles que la date de commande, l'adresse de livraison, le statut de la commande, etc.

L'entité OrderItem représente un produit commandé dans une commande, et est stockée dans une table appelée order\_item. Cette table contient des informations sur chaque produit commandé, telles que la quantité commandée, le prix unitaire, etc.

Enfin, l'entité User représente un utilisateur de mon site internet et est stockée dans une table appelée user. Cette table contient des informations sur chaque utilisateur, telles que leur nom, leur adresse e-mail, leur mot de passe haché, etc.

Chacune de ces entités est liée à d'autres entités par des clés étrangères, ce qui permet de représenter des relations entre les différentes tables. Par exemple, chaque OrderItem est associé à un seul produit (Product).

* Schéma :

Une image contenant texte, Police, nombre, diagramme

Description générée automatiquement

Methode Merise

MCD

Le Modèle Conceptuel des Données (MCD) est une étape essentielle dans la conception d'une base de données. Il représente les entités, les associations et les attributs nécessaires pour modéliser les données d'un système d'information. Le MCD utilise des concepts tels que les entités, les relations, les clés primaires et les clés étrangères pour décrire la structure logique des données.

Dans notre projet, nous avons utilisé le MCD pour définir les entités principales, telles que "Product", "Category", "Order", "OrderItem" et "User". Chaque entité est caractérisée par des attributs qui décrivent les propriétés spécifiques de cette entité. Par exemple, l'entité "Product" possède des attributs tels que "ID", "Name", "Description" et "Price", qui représentent respectivement l'identifiant du produit, son nom, sa description et son prix.

Les associations entre les entités sont également représentées dans le MCD. Par exemple, nous avons établi une relation "1 à N" entre les entités "Category" et "Product", ce qui signifie qu'une catégorie peut contenir plusieurs produits, tandis qu'un produit est associé à une seule catégorie. Cette relation est matérialisée par l'ajout d'une clé étrangère "CategoryID" dans l'entité "Product", qui fait référence à l'identifiant de la catégorie correspondante.

Une image contenant texte, diagramme, capture d’écran, ligne

Description générée automatiquement

MLD

Une fois que le MCD est créé, nous pouvons passer à la modélisation logique des données (MLD). Le MLD consiste à transformer le MCD en un modèle logique spécifique à un système de gestion de base de données (SGBD) particulier, tel que MySQL. Cette transformation implique de définir les tables, les colonnes, les types de données, les contraintes d'intégrité et les relations entre les tables.

Dans notre projet, nous avons utilisé le MLD pour créer les tables de la base de données MySQL. Chaque entité du MCD correspond à une table dans le MLD, avec des colonnes représentant les attributs de l'entité. Par exemple, l'entité "Product" du MCD est représentée par une table "product" dans le MLD, avec des colonnes telles que "ID", "Name", "Description" et "Price".

Les relations entre les entités dans le MCD sont traduites en relations entre les tables dans le MLD. Par exemple, la relation "1 à N" entre les entités "Category" et "Product" se traduit par l'ajout d'une clé étrangère "CategoryID" dans la table "product", qui fait référence à l'identifiant de la catégorie correspondante dans la table "category".

Une image contenant texte, diagramme, capture d’écran, Police

Description générée automatiquement

# 7.Conception de L’application :

UML:

Pour réaliser cette partie nous avons utilisé StarUML, cet outil permet de créer des profils utilisateurs, des diagrammes, il nous a permis d’établir de nombreux éléments de conception que nous allons aborder.

Au niveau de la conception du projet API-BWM trois cas principaux d’utilisation sont présents (Admin et visiteur et client)

*Diagramme de cas généralisé pour un Administrateur*

*Diagramme de cas généralisé pour un visiteur*

*Diagramme de cas généralisé pour un client*

Diagrame de cas d’utilisation

Avant de commencer la réalisation du projet je me sers du logiciel StartUML pour schématiser les cas d’utilisation de l’application.

Je représente trois cas différents :

* Visiteur : Le visiteur peut consulter les prestations et contacter l’administrateur, Il a aussi la possibilité de s’inscrire.
* Utilisateur : L’utilisateur peut faire la même chose que le visiteur, après connexion il peut gère son profil, et ses commandes et également il peut ajouter des prestations à son panier et passer au paiement.

• Administrateur : Il peut faire la même chose que l’utilisateur, la différence qu’il a accès à tous les comptes des utilisateurs inscrit et à leurs commandes. Et il peut également intervenir sur la gestion de produits et catégories (ajout - suppression- modification)

III.Shéma:

* *Diagramme de cas généralisé pour un Administrateur*

Une image contenant texte, diagramme, dessin, ligne

Description générée automatiquement

* *Diagramme de cas généralisé pour un visiteur*

Une image contenant texte, capture d’écran, diagramme, croquis

Description générée automatiquement

* Une image contenant texte, diagramme, dessin, Police

  Description générée automatiquement*Diagramme de cas généralisé pour un client*

III. Diagrame de classe

Ce diagramme de classes permet de visualiser les principales entités du système et leurs relations. Il constitue une base solide pour la conception du système et peut être étendu avec d'autres classes et attributs spécifiques en fonction des besoins du site de vente de prestations de beauté.

* Shéma:

Une image contenant texte, diagramme, Parallèle, capture d’écran

Description générée automatiquement

Dans ce diagramme, nous avons plusieurs classes représentant les entités principales du système :

1. **Utilisateur** : Cette classe représente les utilisateurs enregistrés sur le site. Elle peut avoir des attributs tels que l'identifiant, le nom, l'adresse email, le mot de passe, etc.
2. Administrateur : Cette classe représente l'administrateur du site. Elle peut avoir des attributs similaires à la classe Utilisateur, ainsi que des fonctionnalités spécifiques à l'administration du site.
3. **Visiteur** : Cette classe représente les visiteurs non connectés sur le site. Elle peut avoir des attributs similaires à la classe Utilisateur, mais avec des fonctionnalités limitées.
4. Produit : Cette classe représente les différentes prestations de beauté disponibles à la vente sur le site. Elle peut avoir des attributs tels que l'identifiant, le nom, la description, le prix, etc.
5. **Catégorie** : Cette classe représente les catégories auxquelles les produits peuvent être associés. Elle peut avoir des attributs tels que l'identifiant, le nom, la description, etc.
6. Commande : Cette classe représente une commande passée par un utilisateur. Elle peut avoir des attributs tels que l'identifiant, la date de commande, le statut de la commande, etc.
7. **Panier** : Cette classe représente le panier d'un utilisateur où les produits sélectionnés sont temporairement stockés avant de passer à la commande. Elle peut avoir des attributs tels que l'identifiant, la liste des produits, les quantités, etc.

**Les associations entre les classes sont représentées par des lignes avec des flèches :**

* L'utilisateur est associé à toutes les autres classes, car il interagit avec le système en tant qu'acheteur ou administrateur.
* Les produits sont associés à la classe Catégorie, car chaque produit est lié à une catégorie spécifique.
* Le panier est associé à l'utilisateur, car chaque utilisateur a son propre panier.
* La commande est associée à l'utilisateur, car chaque commande est effectuée par un utilisateur spécifique.

# 8-Conception multi-touches:

Architecture MVVMImage

L'architecture MVVM (Modèle-Vue-VueModèle) est un modèle de conception qui sépare les responsabilités de l'interface utilisateur et de la logique métier d'une application. Il est couramment utilisé avec les frameworks de développement d'applications modernes tels que Angular (pour le front-end) et Symfony (pour le back-end). Voici comment l'architecture MVVM peut être appliquée à une application de vente de prestations avec un backend PHP Symfony et un front-end Angular :

1. Modèle (Model) :
   * Dans le backend Symfony, le modèle représente la logique métier et la gestion des données. Il peut inclure des entités, des services et des repositories qui interagissent avec la base de données.
   * Le modèle est responsable de la persistance des données, des opérations de lecture/écriture, des règles métier et des validations.
2. Vue (View) :
   * Dans le front-end Angular, la vue est responsable de l'interface utilisateur et de l'affichage des données.
   * La vue est généralement constituée de composants Angular qui affichent les données et interagissent avec l'utilisateur.
3. VueModèle (ViewModel) :
   * Le VueModèle est une couche intermédiaire qui agit comme une liaison entre la vue et le modèle.
   * Dans le front-end Angular, le VueModèle représente les données et le comportement associés à une vue spécifique.
   * Il récupère les données du modèle via des appels aux API REST exposées par le backend Symfony et les prépare pour l'affichage dans la vue.
   * Il gère également les interactions utilisateur, comme les formulaires, les actions de bouton, etc.
   * Le VueModèle peut également inclure des fonctions de validation des données et des opérations spécifiques à la vue.

Pour mettre en œuvre cette architecture, j’ai utilisé Angular pour développer le front-end de votre application de vente de prestations et Symfony pour créer le backend. Les services API REST exposés par Symfony seront consommés par les VueModèles Angular pour récupérer et manipuler les données. Les composants Angular seront responsables de l'affichage des données et de l'interaction avec l'utilisateur.

# 9.SECURITE

La sécurité est une considération essentielle pour toute application, y compris une application de vente de prestations de beauté. Voici quelques exigences de sécurité à prendre en compte :

1. Authentification et gestion des utilisateurs : Mettez en place un système d'authentification sécurisé pour permettre aux utilisateurs de s'inscrire, de se connecter et de gérer leurs comptes. Utilisez des techniques de hachage et de salage appropriées pour stocker les mots de passe des utilisateurs de manière sécurisée.
2. Autorisations et contrôle d'accès : Mettez en place des mécanismes de contrôle d'accès pour définir les permissions des utilisateurs et restreindre l'accès aux fonctionnalités sensibles. Assurez-vous que seuls les utilisateurs autorisés peuvent effectuer des opérations telles que la modification des informations de compte, la gestion des réservations, etc.
3. Protection contre les attaques par injection : Utilisez des requêtes préparées ou des ORM (Object-Relational Mapping) pour interagir avec la base de données et éviter les vulnérabilités d'injection SQL. Assurez-vous que toutes les données entrantes sont correctement validées et filtrées pour prévenir les attaques par injection.
4. Protection contre les failles de sécurité web : Appliquez les bonnes pratiques de sécurité web, telles que la validation et l'échappement des données de l'utilisateur pour prévenir les attaques XSS (Cross-Site Scripting) et CSRF (Cross-Site Request Forgery). Utilisez des mécanismes de protection tels que les jetons CSRF pour renforcer la sécurité de l'application.
5. Sécurité des transactions financières : Lors du traitement des paiements en ligne, utilisez des passerelles de paiement sécurisées et conformes aux normes de sécurité, telles que PCI DSS (Payment Card Industry Data Security Standard). Assurez-vous que les informations sensibles telles que les détails de carte de crédit sont cryptées lors de leur transmission.
6. Sécurité des API : Protégez vos API en mettant en place une authentification et une autorisation appropriées. Utilisez des clés API ou des jetons d'accès pour vérifier l'identité et les autorisations des utilisateurs qui accèdent aux API. Limitez également l'accès aux fonctionnalités sensibles uniquement aux utilisateurs autorisés.
7. Protection contre les attaques par force brute : Mettez en place des mécanismes de protection contre les attaques par force brute, telles que le verrouillage des comptes après un certain nombre de tentatives infructueuses de connexion.
8. Gestion des mises à jour de sécurité : Assurez-vous de maintenir tous les composants logiciels à jour, y compris les frameworks, les bibliothèques et les modules tiers utilisés dans l'application. Suivez les mises à jour de sécurité et appliquez-les régulièrement pour prévenir les vulnérabilités connues.

En respectant ces exigences de sécurité, vous pourrez protéger les données sensibles des utilisateurs, prévenir les attaques et assurer la confidentialité et l'intégrité des informations dans votre application de vente de prestations de beauté.

Sécurité et gestion des accès

Pour assurer la sécurité et la gestion des accès sur votre site de vente de prestations de beauté, voici quelques mesures et fonctionnalités à prendre en compte :

* Authentification des utilisateurs :
  + Mettez en place un système d'authentification robuste pour permettre aux utilisateurs de créer des comptes et de se connecter en toute sécurité.
  + Utilisez des algorithmes de hachage pour stocker les mots de passe de manière sécurisée dans la base de données.
  + Mettez en œuvre des mécanismes de récupération de mot de passe sécurisés, tels que l'envoi d'un lien de réinitialisation par e-mail.
* Gestion des rôles et des autorisations :
  + Définissez différents niveaux d'accès pour les utilisateurs en fonction de leur rôle, tels que les administrateurs, les employés et les clients.
  + Accordez des autorisations spécifiques à chaque rôle pour contrôler les fonctionnalités et les actions auxquelles ils ont accès.
  + Utilisez des middleware ou des annotations pour restreindre l'accès à certaines parties de l'application en fonction du rôle de l’utilisateur.
* Protection contre les attaques courantes :
  + Mettez en place des mesures de protection contre les attaques courantes telles que les attaques par injection SQL, les attaques par cross-site scripting (XSS) et les attaques par falsification de requête intersite (CSRF).
  + Utilisez des requêtes préparées ou des ORM pour éviter les injections SQL.
  + Appliquez l'échappement des caractères spéciaux et la validation des données utilisateur pour prévenir les attaques XSS.
  + Utilisez des jetons CSRF pour protéger les formulaires et les requêtes contre les attaques CSRF.
* Sécurité des transactions et des données :
  + Utilisez des protocoles de communication sécurisés tels que HTTPS pour crypter les données échangées entre le navigateur et le serveur.
  + Stockez les informations sensibles, telles que les informations de paiement, de manière sécurisée en utilisant des méthodes de chiffrement appropriées.
  + Assurez-vous que les informations d'identification, telles que les mots de passe, ne sont jamais stockées en clair et sont protégées de manière adéquate.
* Journalisation et suivi des activités :
  + Implémentez un système de journalisation pour enregistrer les activités des utilisateurs et les événements importants.
  + Surveillez les journaux pour détecter les comportements suspects ou les tentatives d'accès non autorisées.
  + Mettez en place des alertes de sécurité pour être informé en cas d'activité suspecte ou de violation potentielle.
* Mises à jour régulières et sécurité du serveur :
  + Maintenez les composants du serveur (système d'exploitation, serveur web, base de données, etc.) à jour avec les dernières mises à jour de sécurité.
  + Appliquez les bonnes pratiques de configuration de serveur pour renforcer la sécurité, tels que la désactivation des fonctionnalités inutiles et la sécurisation des paramètres de sécurité du serveur web

# 10.Démonstration de l’application

# 11.Politique de test

Test unitaire et test fonctionnel

Tests utilisateurs

Les tests utilisateurs sont une méthode pour évaluer l'expérience utilisateur d'un site internet. Ils permettent de collecter des données sur la façon dont les utilisateurs interagissent avec le site et d'identifier les problèmes ou les points à améliorer.

Dans le cadre de notre projet, nous avons mené des tests utilisateurs pour évaluer la convivialité de notre site et recueillir des commentaires de nos utilisateurs. Nous avons recruté des utilisateurs représentatifs de notre population cible et les avons invités à tester notre site en effectuant des tâches spécifiques.

Nous avons utilisé une variété de méthodes de test, y compris des tests de convivialité, des tests de navigation, des tests de fonctionnalité, et des tests de rétroaction. Nous avons également enregistré les sessions de test pour pouvoir analyser les résultats plus tard.

Les commentaires que nous avons reçus de nos utilisateurs ont été extrêmement précieux pour améliorer l'expérience utilisateur de notre site. Nous avons apporté des modifications en fonction de leurs commentaires, tels que des améliorations de la navigation, des ajustements de l'interface, et des changements de fonctionnalité pour mieux répondre à leurs besoins.

Les tests utilisateurs nous ont permis d'identifier les problèmes et d'améliorer l'expérience utilisateur de notre site en nous donnant une perspective extérieure et en nous fournissant des commentaires directs de nos utilisateurs. Cela nous a permis de créer un site plus convivial et plus efficace pour notre population cible.

* Exemple de test effectuer sur notre application :

Tester nos entités au sein de notre base de donnée pour voir si elle se comporte comme prévu :

**Entité Product :**

Une image contenant texte, capture d’écran, logiciel, affichage

Description générée automatiquement

Une image contenant texte, Appareils électroniques, capture d’écran, logiciel

Description générée automatiquement

**Tester nos pages et les fonctionnalités de notre application :**

->faire la requette

->Interagir avec la page

->Tester la réponse

**Une image contenant texte, capture d’écran, Police

Description générée automatiquementPage d’acceuil :**

* Intégrité des données

Pour vérifier que les données enregistrées en base de données sont bonnes et que l'intégrité des données y est assurée, j'ai mis en place plusieurs mesures.

Tout d'abord, j'ai utilisé des requêtes SQL pour récupérer les données stockées dans la base de données et les comparer à celles que j'ai entrées dans l'application. Cela m'a permis de m'assurer que les données étaient correctement stockées et que l'intégrité des données était maintenue.

Ensuite, j'ai utilisé des fonctions de validation côté serveur pour m'assurer que les données entrées dans les formulaires étaient correctes et qu'elles satisfaisaient aux contraintes de la base de données. Cela a permis de garantir que seules les données valides étaient enregistrées en base de données.

* Responsivité du site internet

Pour tester la responsivité de notre site, j'ai utilisé plusieurs outils de développement web tels que l'outil de simulation de responsive design dans les navigateurs web et l'outil d'inspection des éléments.

Je me suis assuré que les pages de notre site s'affichent correctement sur tous les types d'appareils tels que les ordinateurs de bureau, les ordinateurs portables, les tablettes et les smartphones. J'ai également vérifié la compatibilité avec les différents navigateurs web populaires tels que Google Chrome, Mozilla Firefox et Safari.

Sécurisation par l’HTTPS

Le protocole HTTPS est une extension du protocole HTTP qui utilise une couche de chiffrement SSL/TLS pour sécuriser les données échangées entre le navigateur de l'utilisateur et le serveur Web. Cela permet de garantir que les informations transmises, telles que les identifiants et les mots de passe, ne peuvent pas être interceptées et lues par des tiers malveillants.

Dans le cadre de mon projet, j'ai utilisé un hébergeur qui propose le protocole HTTPS de manière native. Cela signifie que toutes les communications entre le navigateur de l'utilisateur et le serveur sont automatiquement chiffrées, sans que je n'aie besoin de faire quoi que ce soit de particulier pour le mettre en place.

Cependant, j'ai tout de même pris soin de bien configurer mon site pour qu'il fonctionne correctement avec le protocole HTTPS. J'ai notamment dû m'assurer que toutes les ressources externes (images, scripts, stylesheets, etc.) étaient également accessibles via une connexion chiffrée, afin d'éviter tout avertissement de sécurité dans le navigateur de l'utilisateur.

Le protocole HTTPS est un élément clé de la sécurité et de la confidentialité sur le Web, et j'ai veillé à ce qu'il soit correctement mis en place sur mon site pour garantir la meilleure expérience utilisateur possible.

Actions sur le référencement naturel

* Description du SEO

Le SEO (Search Engine Optimization) est une technique qui permet d'optimiser le positionnement d'un site web sur les résultats des moteurs de recherche. Le but du SEO est d'augmenter la visibilité et le trafic du site, en le rendant plus facilement trouvable pour les internautes.

Le SEO repose sur plusieurs techniques telles que la recherche de mots-clés, l'optimisation de la structure du site, la qualité et la pertinence du contenu, la création de liens entrants, etc. En travaillant ces différents aspects, on peut améliorer la pertinence du site pour les moteurs de recherche, ce qui aura pour effet d'améliorer son positionnement sur les résultats de recherche.

Pour notre projet, le SEO est important car cela permettra d'augmenter la visibilité de notre site et donc d'attirer davantage de clients potentiels. En effet, un bon positionnement sur les moteurs de recherche peut générer un trafic important et de qualité vers le site, ce qui peut se traduire par une augmentation des ventes.

Pour optimiser le SEO de notre site, nous avons travaillé sur la structure des pages, l'optimisation des balises meta, la qualité et la pertinence du contenu, ainsi que la création de liens entrants. Nous avons également veillé à respecter les bonnes pratiques en matière de SEO pour maximiser nos chances de réussite.

* Actions relatives au SEO

Pour améliorer la visibilité de mon site sur les moteurs de recherche, j'ai mis en place plusieurs éléments de SEO. Tout d'abord, j'ai choisi des mots-clés pertinents pour mon domaine d'activité et je les ai inclus dans le contenu de mon site, en particulier dans les titres et les descriptions des pages. J'ai également utilisé des balises méta pour spécifier les mots-clés et les descriptions de chaque page de mon site.

Ensuite, j'ai optimisé la structure de mon site en créant une arborescence claire pour les pages, en utilisant des liens internes pour faciliter la navigation et en ajoutant des balises de titre et de description à mes images.

J'ai également veillé à la qualité et à la pertinence de mon contenu, en utilisant des mots-clés liés à mon domaine d'activité et en créant des textes informatifs pour chaque page de mon site. Cela permet non seulement de mieux répondre aux attentes des utilisateurs, mais aussi de fournir du contenu pertinent pour les moteurs de recherche.

Enfin, j'ai utilisé des outils tels que Google Analytics pour surveiller les performances de mon site, en analysant le trafic, les comportements des utilisateurs et les sources de trafic pour identifier les zones à améliorer. J'ai également créé un fichier sitemap.xml pour faciliter l'indexation de mon site par les moteurs de recherche.

En mettant en place ces éléments de SEO, j'espère améliorer la visibilité de mon site sur les moteurs de recherche et augmenter le nombre de visiteurs qualifiés sur mon site. Cela peut conduire à une augmentation des ventes et à une amélioration de la notoriété de ma marque

* II. Hébergement du site internet

J'ai choisi d'héberger mon site web sur IONOS, anciennement 1&1 IONOS, un fournisseur d'hébergement web et de noms de domaine basé en Allemagne. IONOS est l'un des plus grands hébergeurs d'Europe et propose une gamme de services d'hébergement web pour les particuliers et les entreprises.

Fondée en 1988, l'entreprise a connu une croissance rapide dans les années 90 et a été l'une des premières entreprises à offrir des services d'hébergement web en Europe. Aujourd'hui, IONOS sert plus de 8 millions de clients dans le monde entier et propose une gamme complète de services d'hébergement web, y compris des plans d'hébergement partagé, des serveurs dédiés et des solutions d'hébergement cloud.

L'un des avantages de l'utilisation d'IONOS est la facilité d'utilisation de leur panneau de contrôle, qui est conçu pour être convivial même pour les utilisateurs débutants. Ils offrent également une assistance technique 24 heures sur 24 et 7 jours sur 7, ce qui est essentiel pour moi en cas de problème avec mon site web.

En fin de compte, j'ai choisi IONOS pour héberger mon site web en raison de leur réputation, de leur fiabilité et de leur facilité d'utilisation, ainsi que de leur excellent service client.

# 12.Veille technoligique

Ressources utilisées

En tant que développeur web, je suis très sensible à la sécurité informatique et je suis régulièrement en veille pour suivre les dernières tendances en matière de sécurité. Dans cette optique, j'ai identifié plusieurs sites intéressants qui m'ont permis de me tenir informé des dernières menaces et des bonnes pratiques à adopter pour protéger les sites web que je développe.

Le premier site que je suis régulièrement est celui de l'OWASP (Open Web Application Security Project). L'OWASP est une organisation à but non lucratif qui vise à améliorer la sécurité des applications web. Leur site web propose une mine d'informations, notamment des guides de bonnes pratiques pour sécuriser les applications web, des outils de test de sécurité et des rapports sur les dernières tendances en matière de sécurité. J'utilise également leur Top 10 des vulnérabilités de sécurité des applications web pour m'assurer que mes applications sont conformes aux normes de sécurité.

Le deuxième site que je suis régulièrement est celui de la communauté de sécurité informatique Reddit. Bien que cette communauté ne soit pas spécifiquement dédiée à la sécurité des applications web, elle propose un grand nombre de ressources sur la sécurité informatique en général, y compris des conseils pour protéger les sites web. Je suis particulièrement intéressé par les fils de discussion sur les dernières failles de sécurité et les techniques de piratage, qui me permettent de comprendre les dernières tendances en matière de sécurité et de prévoir les risques.

# 13.Difficultés rencontrées

Lorceque j’essayais de me connecter j’avais une erreur : **invalid credentials**

J’ai commencer par regarder dans les logs  et j’ai trouvé cette erreur:

**Authenticator does not support the request.** {"firewall\_name":"main","authenticator":"App\\Security\\UserAuthenticator"} []

L'erreur mentionnée: "Authenticator does not support the request" indique que l'authentificateur ne prend pas en charge la requête actuelle. Cela peut se produire lorsque l'authentificateur est configuré pour un pare-feu spécifique dans Symfony, mais la requête actuelle est envoyée à un autre pare-feu ou que l'authentificateur n'est pas correctement configuré pour le pare-feu actuel.

**Voici les étapes à suivre pour résoudre ce problème :**

Assurez-vous que votre fichier **security.yaml** est correctement configuré. Vérifiez que le pare-feu **main** est configuré et qu'il utilise l'authentificateur **App\Security\UserAuthenticator**.

Vérifiez que vous utilisez le pare-feu **main** dans votre configuration de sécurité. Cela peut être spécifié dans le fichier **config/packages/security.yaml** ou dans un autre fichier de configuration lié à la sécurité.

Assurez-vous que votre route de connexion, généralement **app\_login**, est correctement configurée dans votre fichier **security.yaml**. Vérifiez que cette route correspond à celle utilisée dans la méthode **getLoginUrl()** de votre **UserAuthenticator**.

Si vous utilisez plusieurs pare-feux dans votre application Symfony, assurez-vous que la requête est envoyée au bon pare-feu correspondant à votre authentificateur. Vous pouvez spécifier le pare-feu dans la configuration du pare-feu de votre application ou en utilisant les annotations **@Security** dans vos contrôleurs.

Vérifiez que votre authentificateur est correctement enregistré et configuré dans votre application Symfony. Assurez-vous que vous avez bien déclaré le service **App\Security\UserAuthenticator** et que vous l'avez correctement lié à votre pare-feu dans votre fichier **services.yaml** ou **security.yaml**.

.Assurez-vous que la méthode **supports()** dans votre authentificateur **UserAuthenticator** est correctement implémentée pour prendre en charge la requête de connexion (**GET /login**). Par défaut, l'authentificateur ne devrait pas prendre en charge les requêtes **GET**, car il est conçu pour traiter les requêtes **POST** provenant du formulaire de connexion. Vous pouvez mettre à jour la méthode **supports()** pour retourner **false** pour les requêtes **GET**. Par exemple :

public function supports(Request $request): bool { return $request->isMethod('POST') && $request->attributes->get('\_route') === 'app\_login'; }

7.Vérifiez que votre formulaire de connexion envoie une requête **POST** plutôt qu'une requête **GET**. Assurez-vous que votre formulaire est configuré pour soumettre les données via la méthode **POST**.

**Solution :**

Rep à la question 7  
Le formulaire semble correct et soumet les données via la méthode **POST**, ce qui est conforme aux attentes de votre authentificateur **UserAuthenticator**. Cependant, il y a une petite erreur dans votre formulaire. L'attribut **name** de votre champ de mot de passe est défini sur "Mot de passe" au lieu de "password". Cela peut entraîner des problèmes lors de la soumission du formulaire. Pour corriger cela, modifiez l'attribut **name** du champ de mot de passe comme suit

**<input type="password" name="password" id="inputPassword" class="form-control" autocomplete="current-password" required>**

# 14.CONSCLUSION

Pour l'avenir de notre site, nous avons plusieurs projets à venir. Tout d'abord, nous souhaitons améliorer l'expérience utilisateur en proposant une version mobile plus conviviale. Nous voulons également ajouter une fonctionnalité de chat en direct pour permettre aux utilisateurs de poser des questions et de recevoir une assistance en temps réel. Nous prévoyons également de mettre en place une stratégie de référencement plus solide pour attirer plus de trafic sur le site. Enfin, nous travaillons sur l'intégration d'une plateforme de formation en ligne pour permettre aux utilisateurs de développer leurs compétences en ligne. Nous sommes conscients que cela représente un travail important, mais nous sommes confiants dans notre capacité à continuer à développer et à améliorer notre site pour fournir une expérience utilisateur de qualité à tous nos utilisateurs.