Informatique
iut Nord Franche-Comté

SAE 2.03 - BUT INFORMATIQUE - GROUPE 1

Dossier de définition des besoins pour l'hébergement de l'application meuuuhble

Auteur :
Florian HEGELE(*A1*)
Yahia KHERZA(*A1*)
Olivier MARAVAL(*A1*)
Manon LAMBLOT(*A1*)

Client :
Michel SALOMON
Référent :
Olivier MARAVAL

2 avril 2024

Table des matières

Chapitre 1

Description de l'application développée

Meuuuhble entend développer un nouveau canal de distribution en investissant dans le commerce en ligne, ce qui permettra de toucher une clientèle plus large et de faciliter l'expansion à l'international. La plateforme digitale sera conçue pour offrir une expérience utilisateur intuitive et engageante, renforçant ainsi la présence en ligne de Meuuuhble et son image de marque. C'est en cela que consistera la boutique en ligne développée pour Meuuuhble.

1.1 Description non technique de l'application

L'application développée est une plateforme de commerce en ligne dédiée à la vente de meubles. Elle permet aux utilisateurs de parcourir un catalogue de meubles, de les ajouter à leur panier, de passer commande, et de laisser des commentaires sur les produits qu'ils ont achetés.

Voici les points clés définissant l'application éponyme de l'entreprise Meuuuhble :

- **Achat d'articles :** Le site permettra aux utilisateurs d'acheter des meubles en ligne, avec un processus de paiement sécurisé et une interface claire pour faciliter la transaction.
- **Recherche avancée d'articles :** Une fonctionnalité de recherche avec des filtres dynamiques sera intégrée, permettant aux clients de trouver facilement des articles selon différents critères tels que la catégorie, le prix, la couleur, le matériau, etc. Les articles pourront se décliner en différentes variantes, offrant ainsi un choix plus large.
- **Commentaires et évaluations :** Les utilisateurs auront la possibilité de commenter et de noter les articles qu'ils ont achetés, ce qui fournira des retours précieux à la fois pour Meuuuhble et pour les futurs acheteurs.
- **Gestion des adresses d'expédition :** Les clients pourront enregistrer plusieurs adresses d'expédition et sélectionner l'adresse préférée lors du processus de commande, simplifiant ainsi les achats répétés.
- **Liste d'envies et historique :** Le site offrira la possibilité de créer une liste d'envies où les clients pourront sauvegarder leurs articles préférés pour un achat futur. Un historique des derniers articles consultés sera également disponible pour faciliter la reprise du shopping.

Ces fonctionnalités sont conçues pour améliorer l'expérience client en ligne, augmenter l'engagement et favoriser la fidélisation. Elles soutiendront également les objectifs commerciaux de Meuuuhble en augmentant le taux de conversion et en élargissant sa clientèle.

1.2 Description technique de l'application

L'application utilise le framework web Flask pour le backend, écrit en Python. Flask est choisi pour sa simplicité et sa flexibilité, permettant de développer rapidement des applications web avec une structure claire et maintenable. L'architecture logicielle repose sur une séparation entre la logique de présentation (templates HTML) et la logique métier (contrôleurs).

1.2.1 Langages de Programmation

- **Frontend : JavaScript**, **HTML5**, **CSS3**
- **Backend : Python 3**, avec **Flask** comme framework web et **Jinja** comme moteur de template

1.2.2 Bibliothèques et Frameworks

- **Bootstrap** pour la construction de l'interface utilisateur
- **Flask :** Framework web en Python pour la création d'applications web et notamment les fonctionnalités suivantes :
  - **request :** Permet d'accéder aux données de la requête entrante.
  - **render_template :** Utilisé pour rendre des modèles Jinja2 pour générer des pages HTML.
  - **redirect :** Redirige l'utilisateur vers une autre URL.
  - **url_for** : Génère une URL pour une fonction de vue spécifique.
  - **abort :** Arrête le traitement d'une requête et renvoie une réponse d'erreur.
  - **flash :** Utilisé pour afficher un message flash à l'utilisateur.
  - **session :** Permet de stocker des informations utilisateur entre les requêtes.
  - **g :** Objet global qui est utilisé pour stocker des données pendant toute la durée d'une requête.
  - **blueprint :** Permet de créer des modules Flask réutilisables et modulaires, regroupant des routes, des vues et d'autres fonctionnalités liées.
- **SQLAlchemy : Utilisé comme ORM pour faciliter l'interaction avec la base de données.**

1.2.3 Base de Données

- **mysql** pour le stockage des données relationnelles

1.2.4 Infrastructure et Déploiement

- **Git** pour le contrôle de version
- **Python-Anywhere** pour la présentation d'une maquette fonctionnelle au client

1.2.5 Outils utilisés pour le développement

- **Visual Studio Code** et **Pycharm** pour le développement de l'application web.
- **Datagrip** pour assister la gestion de la base de données mysql.
- **Looping** pour l'élaboration du MCD et du MLD.

1.2.6 Arborescence des Fichiers

Voici la représentation de l'arborescence des fichiers pour l'application meuuuhble :

```
meuuuhble/
├── controllers/
│   └── Contient les fichiers python assurant le fonctionnement du site
├── static/
│   ├── assets/
│   │   └── Contient les scripts js et les feuilles de style css
│   ├── images/
│   │   └── Contient les photos des produits vendus
│   └── images_asset/
│       └── Contient les images nécessaire a l'affichage du site
└── templates/
    ├── admin/
    │   └── Contient les templates jinja des pages côté administrateur
    ├── auth/
    │   └── Contient les templates jinja des pages d'authentifications
    └── client/
        └── Contient les templates jinja des pages côté client
```

Chapitre 2

Description des besoins

2.1 Besoins contraints

2.1.1 Matériel

— Processeur multi-cœur (par exemple, Intel Xeon)
— Mémoire RAM suffisante pour gérer le trafic web (recommandé : 8 Go ou plus)
— Espace de stockage SSD pour des performances rapides
— Au moins 10 Go d'espace disque disponible pour le stockage de la base de données, des fichiers statiques (images, CSS, JS), et des logs.

2.1.2 Système d'exploitation

L'application a été développé et testé sur un système **Linux**. Ce sera ce système d'exploitation qui devra être installé sur votre serveur afin d'assurer le fonctionnement du site.

2.1.3 Langages et Bibliothèques

— **Python (version 3.8 ou supérieure) :** Nécessaire pour exécuter l'application Flask. Python doit être installé sur le serveur de production.
— **Flask :** Le framework web utilisé pour développer l'application. Flask doit être installé via pip, le gestionnaire de paquets Python.
— **SQLAlchemy :** Utilisé comme ORM pour faciliter l'interaction avec la base de données. À installer via pip.
— **Jinja :** Moteur de template par défaut de Flask, pour la génération dynamique des pages HTML.

2.1.4 Système de Gestion de Base de Données (SGBD)

MySQL ou **mariaDB** doivent être installés et configurés sur le serveur de production.

2.1.5 Comptes utilisateurs

Comptes utilisateurs avec accès `sudo` pour la gestion du serveur et de l'application. Ces comptes doivent avoir les permissions nécessaires pour installer des logiciels, gérer la base de données, et accéder aux fichiers de l'application.

2.1.6 Administration à Distance

— **SSH :** Pour l'administration à distance du serveur de production, l'accès SSH doit être configuré. Cela permet une connexion sécurisée au serveur pour la maintenance et le déploiement.
— **Outils de déploiement :** Utilisation de Git pour le contrôle de version et de scripts de déploiement automatisés pour faciliter la mise à jour de l'application sur le serveur de production.

2.2 Besoins ouverts

2.2.1 Distribution Linux

Une distribution Linux recommandée pour héberger le site web serait **Ubuntu Server**, largement utilisée pour les serveurs web en raison de sa stabilité, de sa sécurité et de sa facilité d'utilisation. Voici quelques raisons pour lesquelles nous vous proposons ce choix :

— **Stabilité :** Ubuntu Server bénéficie de mises à jour régulières et d'un support à long terme (LTS), assurant la stabilité du système d'exploitation.
— **Sécurité :** Ubuntu Server est réputé pour sa sécurité renforcée et ses correctifs de sécurité réguliers.
— **Facilité d'installation :** L'installation et la configuration d'Ubuntu Server sont relativement simples, ce qui facilite la mise en place du serveur.
— **Support de la communauté :** Ubuntu dispose d'une vaste communauté en ligne prête à fournir de l'aide et des conseils en cas de besoin.

Cependant, vous pourriez aussi choisir d'installer une distribution **Debian** si les points suivants vous semblent pertinents :

— Bien qu'Ubuntu Server bénéficie également de mises à jour régulières, Debian est souvent considéré comme plus conservateur en termes de stabilité, ce qui peut être un avantage pour les environnements critiques.
— Certains utilisateurs estiment que Debian est plus strict en matière de sécurité.
— Debian offre une grande flexibilité en termes de choix de logiciels et de configurations. Il permet une personnalisation approfondie pour répondre aux besoins spécifiques de l'utilisateur.

2.2.2 Serveur web

Flask intègre déjà un serveur de développement. Cependant, pour la mise en production, un serveur web robuste comme Nginx ou Apache est recommandé pour servir l'application. Le choix entre Nginx et Apache peut dépendre des préférences de l'administrateur système ou de considérations spécifiques liées à la performance et à la sécurité.

2.2.3 Serveur de messagerie

Si l'application nécessite l'envoi d'emails (par exemple, pour la récupération de mots de passe ou les notifications), un serveur de messagerie est nécessaire. Des services comme SendGrid, Mailgun, ou l'utilisation d'un serveur SMTP local peuvent être envisagés.

2.2.4 Nom de domaine

— **www.meuuuhblesinnovants.com**

  - **Extension** : .com, populaire et facilement mémorisable.
  - **Coût estimé** : Environ 8,50 € à 12,75 € par an.
- **www.espacemeuuuhble.shop**
  - **Extension** : .shop, parfait pour un site de commerce en ligne.
  - **Coût estimé** : Environ 25,50 € à 34 € par an.
- **www.maisonmeuuuhble.net**
  - **Extension** : .net, une alternative populaire au .com.
  - **Coût estimé** : Environ 8,50 € à 12,75 € par an.
- **www.luxeameuuuuhblement.store**
  - **Extension** : .store, spécifiquement destiné aux boutiques en ligne.
  - **Coût estimé** : Environ 42,50 € à 51 € par an.

2.3 Sécurisation / Utilisation de TLS

2.3.1 Pourquoi sécuriser un site web ?

Sécuriser un site web est crucial pour protéger les données sensibles des utilisateurs contre les interceptions, les modifications non autorisées et autres cyberattaques. Cela aide à maintenir la confidentialité, l'intégrité et la disponibilité des informations.

2.3.2 Objectif de TLS

TLS (Transport Layer Security) vise à sécuriser les communications entre le navigateur de l'utilisateur et le serveur web. Il assure la confidentialité et l'intégrité des données échangées en chiffrant les informations transmises, empêchant ainsi les écoutes indiscrètes et les modifications non autorisées.

2.3.3 Mise en place de TLS

Pour mettre en place TLS, vous avez besoin :
- D'un certificat SSL/TLS, qui peut être obtenu gratuitement via Let's Encrypt ou acheté auprès d'une autorité de certification.
- De configurer le serveur web pour utiliser ce certificat et activer HTTPS.