

Cahier de charge

Sommaire

1. Présentation Générale
 2. Contexte
 3. Présentation du Projet
 4. Objectifs
 5. Exigences Fonctionnelles
 6. Arborescence
 7. Exigences Techniques
 8. Outils
 9. Hébergement
 10. Structure de Base de Données
-

2. Contexte

Le projet EasyCMS est une application web développée dans le cadre d'une réalisation professionnelle. Elle a pour but de fournir une interface de gestion de contenu (CMS) permettant à des utilisateurs d'administrer des pages, contenus, utilisateurs, et menus de navigation d'un site web de manière simplifiée et structurée.

3. Présentation du Projet

Définition

Développement d'une application CMS (Content Management System) qui permet de gérer dynamiquement les contenus d'un site web sans connaissances techniques avancées.

Objectifs

- Création de comptes utilisateurs avec droits d'accès (admin, éditeur.)

- Authentification sécurisée (hashage des mots de passe avec sodium)
 - Création et édition de pages web via une interface intuitive
 - Gestion des contenus associés à une page (texte, description, types de contenu)
 - Gestion des menus de navigation du site
 - Visualisation dynamique des pages publiées
-

5. Exigences Fonctionnelles

- **Connexion / Inscription**
 - Interface de login avec vérification du mot de passe sécurisé
 - Formulaire d'inscription avec gestion des doublons et hachage du mot de passe
 - **Gestion des utilisateurs**
 - Ajout, modification, suppression
 - Gestion des droits (via id_right)
 - **Gestion des pages**
 - Création de pages (nom, statut publié ou non, page d'accueil)
 - Association de contenus aux pages
 - Définition d'une page comme page d'accueil
 - **Gestion des contenus**
 - Ajout de contenu avec type, description, position
 - Édition/suppression
 - Publication ou non d'un contenu
 - **Navigation (menu)**
 - Gestion des menus liés aux pages
 - Publication du menu
 - Organisation dans le site
-

6. Arborescence

- **Page d'accueil** : Vue utilisateur avec les contenus de la home page

- **Back Office**

- Tableau de bord
 - Gestion des utilisateurs
 - Gestion des pages
 - Gestion des contenus
 - Gestion des menus/navigation
-

7. Exigences Techniques

- Architecture MVC (Modèle – Vue – Contrôleur)
 - Utilisation du moteur de templates Twig pour la partie vue
 - Gestion des routes via index.php et des contrôleurs dynamiques
 - Sécurité : Validation des entrées, protection des sessions, hashage sodium
 - PDO pour la gestion des requêtes SQL
-

8. Outils

- Éditeur : VS Code ou PHPStorm
 - Langage : PHP 8.1
 - Base de données : MySQL 5.7
 - Front : HTML5, CSS3, Bootstrap 5, Twig
 - Librairies : Composer, Autoloading PSR-4
 - Débogage : Xdebug, Logs
 - Versionnage : Git / GitHub
-

9. Hébergement

- **Local** : WAMP pour développement
- **Production** : Hébergeur distant configuré via .env

- Paramètres : DB_HOST, DB_NAME, DB_LOGIN, DB_PASSWORD
-

10. Structure de Base de Données (simplifiée)

- **users** (id, login, password, id_right)
- **rights** (id, name)
- **pages** (id, page_name, is_home_page, is_published)
- **contents** (id, content_name, description, id_page, id_position, id_content_type, id_user, is_published)
- **content_types** (id, name, description)
- **positions** (id, position_number, id_page)
- **navigations** (id, name, id_page, is_published)