**PAGE DE GARDE**

**Remerciements**

**Dédicaces**

**Table des matières**

[CHAPITRE 1 : INTRODUCTION 1](#_Toc195179330)

[1. Introduction : 2](#_Toc195179331)

[1.1 Contexte et objectifs du stage : 2](#_Toc195179332)

[1.2 Présentation du sujet : 2](#_Toc195179333)

[1.3 Méthodologie adoptée : 2](#_Toc195179334)

[CHAPITRE 2 : PRESNETATION DE L’ENTREPRISE 3](#_Toc195179335)

[2. Présentation de l’Entreprise : 4](#_Toc195179336)

[2.1 Historique et mission : 4](#_Toc195179337)

[2.2 Services et produits : 4](#_Toc195179338)

[2.3 Organisation interne : 5](#_Toc195179339)

[CHAPITRE 3 : ANALYSE DES BESOINS 6](#_Toc195179340)

[3. Analyse des Besoins : 7](#_Toc195179341)

[3.1 Problématique actuelle : 7](#_Toc195179342)

[3.2 Objectifs du projet : 7](#_Toc195179343)

[3.3 Cahier des charges : 8](#_Toc195179344)

[CHAPITRE 4 : CONCEPTION DU CMS 10](#_Toc195179345)

[4. Conception du CMS : 11](#_Toc195179346)

[4.1 Architecture globale : 11](#_Toc195179347)

[4.2 Schéma UML et wireframes : 11](#_Toc195179348)

[4.2.1 Diagramme de cas d'utilisation : 11](#_Toc195179349)

[4.2.2 Diagramme de classes : 15](#_Toc195179350)

[4.2.3 Diagramme de séquence : 17](#_Toc195179351)

[4.2.4 Diagramme d’objets : 17](#_Toc195179352)

[4.3 Technologies utilisées : 19](#_Toc195179353)

[CHAPITRE 5 : Développement et Implémentation 21](#_Toc195179354)

[5. Développement et Implémentation : 22](#_Toc195179355)

[5.1 Frontend : interfaces et fonctionnalités : 22](#_Toc195179356)

[5.2 Backend : API et base de données : 22](#_Toc195179357)

[5.3 Sécurité et optimisation des performances : 23](#_Toc195179358)

[CHAPITRE 6 : TESTS ET VALIDATIONS 24](#_Toc195179359)

**Liste des FIGURE**

**Figure 1:** xxxxxxxxxxxxxxxxxxxxxxx 12

**Figure 2:** xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 13

**Figure 3:**xxxxxxxxxxxx 14

**Figure 4:**xxxxxxxxx 15

**Figure 5:** xxxxxxxx 16

**Figure 6:** xxxxxxx 17

**Figure 7:** xxxxxxx 18

# CHAPITRE 1 : INTRODUCTION

# 1. Introduction :

## 1.1 Contexte et objectifs du stage :

Dans un contexte où la digitalisation devient un élément clé pour la visibilité des entreprises, la gestion de contenu efficace est un enjeu majeur. Mon stage s'inscrit dans cette dynamique en se focalisant sur l'amélioration d'un site vitrine à travers l'intégration d'un CMS (Content Management System). L'objectif principal est de rendre la gestion du site plus intuitive, flexible et performante pour les utilisateurs finaux.

## 1.2 Présentation du sujet :

Ce projet vise à optimiser l’interface utilisateur et l’ergonomie du site vitrine en mettant en place un CMS moderne et fonctionnel. Grâce à des fonctionnalités avancées telles que le glisser-déposer, la gestion des médias et la prévisualisation en temps réel, le site deviendra plus accessible aux administrateurs non techniques. De plus, des améliorations en termes de performance et de compatibilité multi-appareils seront intégrées.

## 1.3 Méthodologie adoptée :

Pour mener à bien ce projet, une approche méthodique a été suivie :

* **Analyse des besoins :** Étude du site existant et des attentes des utilisateurs.
* **Conception :** Développement des schémas UML et des wireframes pour structurer la nouvelle interface.
* **Développement :** Implémentation du CMS avec des technologies modernes telles que React.js, Node.js et MySQL.
* **Tests et validation :** Vérification des performances et de la facilité d’utilisation du CMS.
* **Déploiement et documentation :** Mise en production et rédaction d’un guide utilisateur.

Cette méthodologie garantit une amélioration significative du site vitrine tout en assurant une expérience utilisateur optimale.

# CHAPITRE 2 : PRESNETATION DE L’ENTREPRISE

# 2. Présentation de l’Entreprise :

## 2.1 Historique et mission :

Créée en septembre 2023, Verdanova Solutions est une startup technologique tunisienne qui s’est rapidement distinguée par son engagement dans la transformation numérique, notamment dans le secteur agricole. Labellisée "Startup Tunisie" en mars 2024 et lauréate du prix Air1 en juillet de la même année, l’entreprise a participé à plusieurs programmes d’incubation comme Entrepreria, TalentHub / Coworky, CIES Tunisie, Startup10 et Medianet.

La mission principale de Verdanova Solutions est de concevoir des solutions numériques innovantes, accessibles et performantes, permettant aux entreprises de surmonter leurs défis digitaux. L’approche adoptée est centrée sur l’utilisateur, combinant expertise technique, créativité et méthodologie agile pour proposer des produits fonctionnels, ergonomiques et adaptés aux besoins spécifiques de chaque client.

## 2.2 Services et produits :

Verdanova Solutions propose une large gamme de services dans le domaine du développement informatique, notamment :

* **Développement Web s Mobile :** Création d’applications et de plateformes personnalisées en utilisant des technologies modernes telles que Node.js, React.js, Vue.js, Flutter et React Native.
* **Développement de plateformes E-commerce et Marketplace :** Conception de solutions complètes pour la gestion des ventes en ligne, favorisant les échanges entre entreprises et clients.
* **Conseil en transformation digitale :** Accompagnement stratégique, audit technologique, élaboration de roadmaps digitales.
* **Intelligence Artificielle s Data Science :** Mise en œuvre de solutions basées sur l’IA pour l’analyse prédictive, la reconnaissance d’images et la gestion intelligente des données.

Le produit phare de l’entreprise est WeeFarm, une solution AgriTech complète dédiée au secteur agricole. Cette plateforme intègre des fonctionnalités de gestion des exploitations, une marketplace agricole et des outils prédictifs basés sur l’intelligence artificielle. WeeFarm permet ainsi de :

* Digitaliser la gestion agricole.
* Améliorer la productivité.
* Faciliter l’accès aux marchés.
* Optimiser les décisions grâce à la data science.

Les fonctionnalités clés incluent la gestion des exploitations, la prédiction des prix agricoles, une base de données agricole vulgarisée, un flux de communication directe avec les experts, et des tableaux de bord interactifs.

## 2.3 Organisation interne :

L’équipe de Verdanova Solutions est constituée de professionnels passionnés par l’innovation technologique, avec des compétences complémentaires en développement logiciel, design, gestion de projet, data science et accompagnement stratégique. La structure de l’entreprise repose sur une organisation agile et collaborative, où chaque membre contribue activement à la réussite des projets. Le travail en équipe, la transparence, l’excellence technique et l’écoute client sont des valeurs fondamentales qui guident les décisions et les actions de la startup.

# CHAPITRE 3 : ANALYSE DES BESOINS

# 3. Analyse des Besoins :

## 3.1 Problématique actuelle :

Dans le secteur agricole, de nombreuses sociétés rencontrent des difficultés pour gérer efficacement leur présence en ligne et leurs processus internes. Ces défis incluent la complexité de la gestion de contenu de leurs sites vitrine, une faible automatisation des processus, et des outils peu adaptés à leurs besoins spécifiques. La gestion manuelle des produits et des services, ainsi que l'absence de synchronisation avec des plateformes de marketplace, ralentit la compétitivité et l'efficacité des entreprises agricoles. De plus, les solutions existantes ne permettent pas une personnalisation optimale, ce qui complique l'adaptation aux besoins variés des sociétés agricoles.

Il existe donc un besoin crucial d’une solution numérique permettant aux sociétés agricoles de créer, personnaliser et gérer facilement leurs sites vitrine tout en intégrant leurs produits et services à une marketplace spécifique. Une telle solution devrait simplifier la gestion des contenus, automatiser certaines tâches et offrir une interface intuitive même pour les utilisateurs non techniques.

## 3.2 Objectifs du projet :

L’objectif principal de ce projet est de développer un module CMS (Content Management System) destiné aux sociétés agricoles de la plateforme Weefarm. Ce CMS vise à répondre aux besoins suivants :

* + **Création et personnalisation des sites vitrine :** Offrir aux sociétés agricoles la possibilité de créer et personnaliser facilement leur site vitrine en ligne, en choisissant des modèles de design, des couleurs, et des mises en page adaptées.
  + **Gestion simplifiée du contenu :** Permettre une gestion intuitive du contenu (texte, images, vidéos, documents) grâce à un éditeur WYSIWYG (What You See Is What You Get).
  + **Synchronisation avec la marketplace :** Assurer l'intégration automatique des produits et services des sociétés agricoles dans la marketplace Weefarm, facilitant ainsi les transactions en ligne.
  + **Gestion des abonnements et des autorisations d'accès** : Offrir un système de gestion des abonnements et des droits d'accès pour une administration flexible du site.
  + **Optimisation de l'expérience utilisateur :** Garantir une expérience fluide, intuitive et adaptée à tous types d'utilisateurs (administrateurs, éditeurs, visiteurs).

Ce projet vise donc à faciliter la gestion des sites vitrine, à améliorer l'automatisation des processus et à fournir une interface qui simplifie l'administration des contenus tout en intégrant les fonctionnalités de la marketplace.

## 3.3 Cahier des charges :

Le Cahier des charges du projet définit les spécifications détaillées pour le développement du module CMS. Il comprend les éléments suivants :

* Architecture du système : Le module CMS doit s’intégrer dans l’architecture globale de la plateforme Weefarm, comprenant la gestion centralisée des accès, des abonnements, des statistiques, ainsi que l’intégration avec la marketplace Weefarm.
  + **Fonctionnalités principales :**
* **Création et personnalisation du site :** Interface permettant aux sociétés agricoles de personnaliser leur site (choix de design, ajout et modification de contenu).
* **Gestion des produits et services :** Synchronisation avec la marketplace Weefarm, ajout et gestion des produits agricoles, et gestion des stocks.
* **Gestion des utilisateurs :** Authentification et gestion des rôles (administrateurs, éditeurs, visiteurs).
* **Abonnements et facturation :** Gestion des abonnements avec des notifications pour le renouvellement, ainsi que le suivi des paiements.
* **Statistiques et reporting :** Tableau de bord interactif pour suivre les performances du site (trafic, conversions, etc.).
  + **Exigences techniques :**

Le module CMS doit être développé avec les technologies suivantes :

* **Frontend :** HTML5, CSS3, JavaScript, React.js, Tailwind CSS.
* **Backend :** Node.js, Express.js, Swagger.
* **Base de données :** MySQL.
* **Sécurité :** Mise en place de mécanismes de sécurité robustes pour protéger les données sensibles et garantir l’intégrité des informations.
  + **Contraintes fonctionnelles et ergonomiques :**

L’interface doit être responsive, facile à utiliser, et compatible avec les différents appareils (mobiles, tablettes, ordinateurs de bureau). Le CMS doit être performant et capable de gérer un grand volume de données sans altérer l’expérience utilisateur.

* + **Livrables attendus :**
* Prototype fonctionnel du CMS.
* Documentation utilisateur et technique.
* Code source et déploiement sur le serveur.

# CHAPITRE 4 : CONCEPTION DU CMS

# 4. Conception du CMS :

## 4.1 Architecture globale :

L'architecture du CMS Weefarm est conçue pour une gestion optimale des sites vitrines des sociétés agricoles et une intégration fluide avec la marketplace. Elle repose sur plusieurs composants clés :

* + Frontend : L'interface utilisateur est construite avec React.js et Tailwind CSS, garantissant une navigation fluide et responsive. Elle permet aux utilisateurs de gérer facilement le contenu du site, avec un accès différencié selon leur rôle (administrateur, éditeur, visiteur).
  + Backend : Le serveur est développé avec Node.js et Express.js, gérant les requêtes et la logique métier via des API RESTful, assurant une communication fluide entre le frontend, la base de données et la marketplace.
  + Base de données : MySQL est utilisé pour stocker toutes les données relatives aux utilisateurs, produits, services et abonnements, permettant une gestion centralisée et performante.
  + Marketplace Weefarm : Le CMS est intégré à la marketplace pour synchroniser les produits et services des sociétés agricoles, facilitant ainsi les transactions et la gestion des ventes.
  + Sécurité et gestion des rôles : Le système inclut une gestion des utilisateurs et des rôles, avec des permissions d'accès spécifiques, et un protocole de sécurité renforcé, y compris l'authentification à plusieurs facteurs (MFA).

## 4.2 Schéma UML et wireframes :

* + **Schéma UML :**

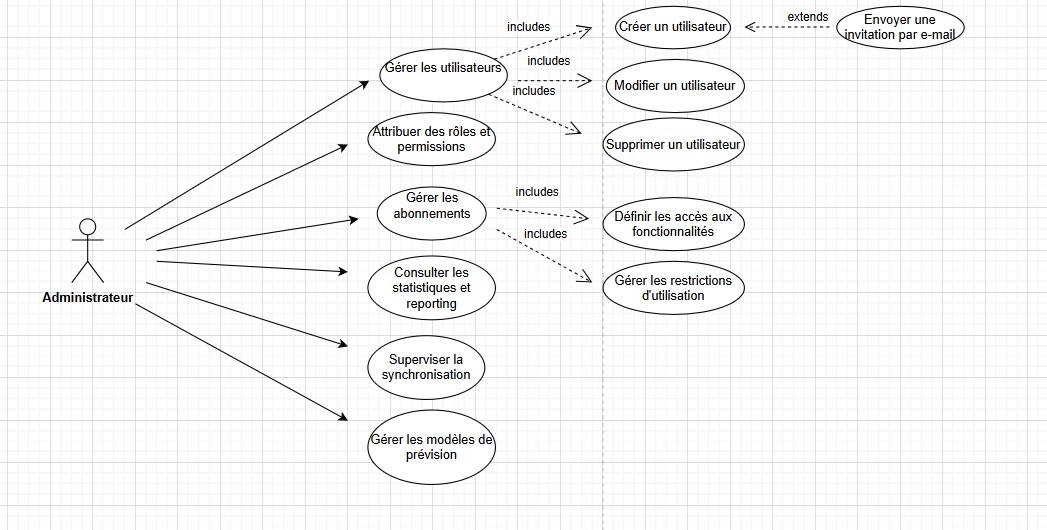
Dans cette partie, nous présentons les principaux diagrammes UML qui illustrent l'architecture du CMS Weefarm. Ces diagrammes montrent l'interaction entre les différents composants du système, les cas d'utilisation des utilisateurs, et les relations entre les entités.

### 4.2.1 Diagramme de cas d'utilisation :

Ce diagramme représente les différentes actions des utilisateurs dans le CMS. Il montre les interactions entre les utilisateurs (administrateurs, éditeurs, etc.) et les fonctionnalités principales du système (ajout de contenu, gestion des produits, configuration des paramètres).



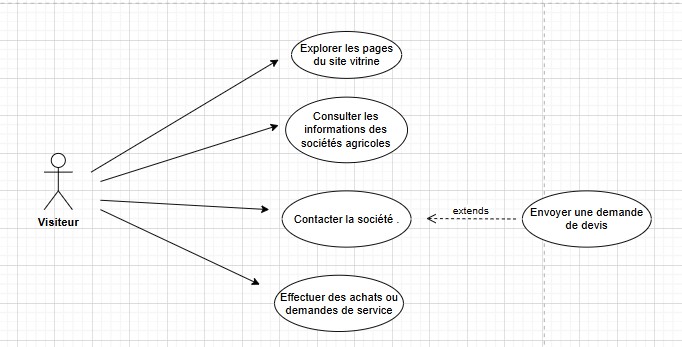
**Figure 1:** xxxxxxxxxxxxxxxxxxxxxxx



**Figure 2:** xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx



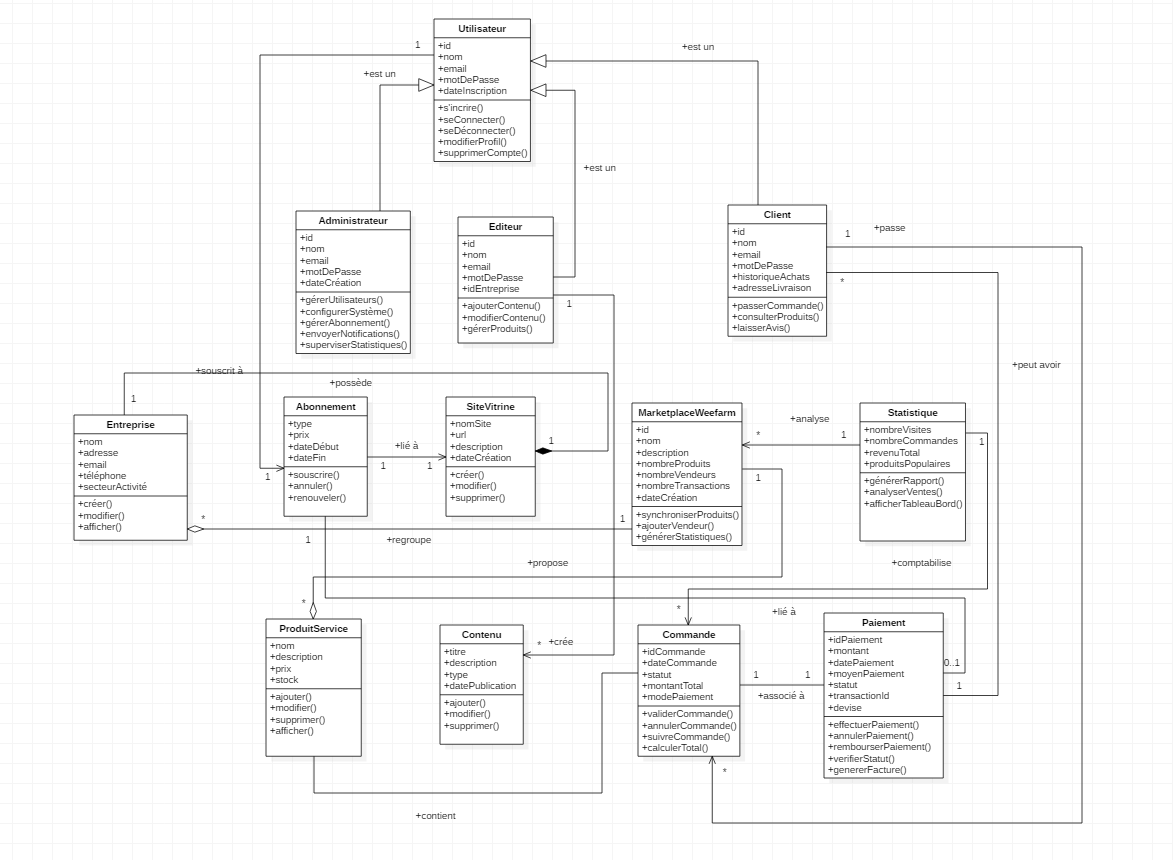
**Figure 3:**xxxxxxxxxxxx



**Figure 4:**xxxxxxxxx

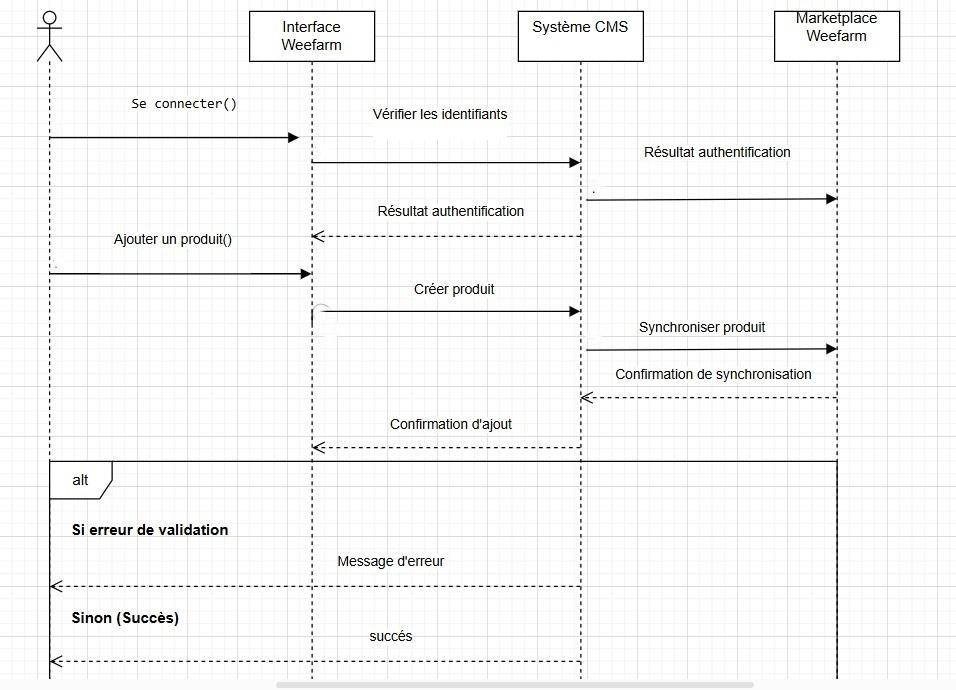
### 4.2.2 Diagramme de classes :

Le diagramme de classes décrit les principales entités du CMS et leurs relations. Il inclut des entités telles que "Utilisateur", "Produit", "Service", "Abonnement" et "Page", ainsi que les associations entrent-elles. Ce diagramme permet de mieux comprendre la structure des données et la façon dont les informations sont liées dans le système.



**Figure 5:** xxxxxxxx

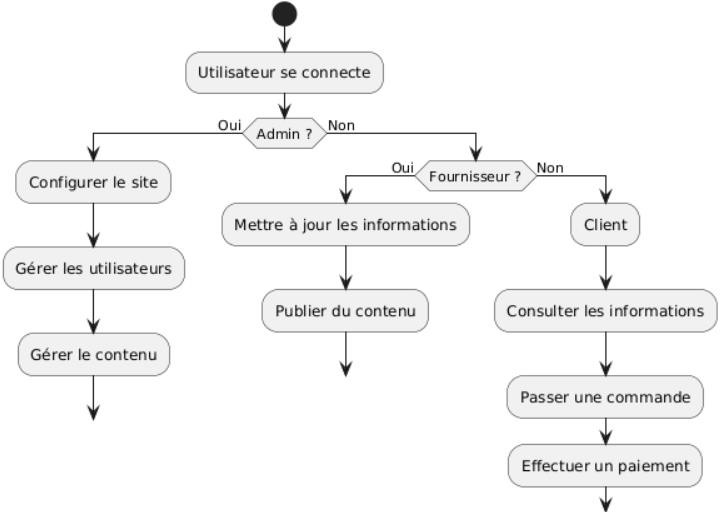
### 4.2.3 Diagramme de séquence :

Ce diagramme montre l'enchaînement des interactions entre le frontend, le backend et la base de données lors de certaines actions, comme l'ajout d'un produit ou la modification d'un abonnement. Il permet de visualiser le flux des données et des appels de méthode au sein du système.

**Figure 6:** xxxxxxx

### 4.2.4 Diagramme d’objets :

Le diagramme d‘objets représente les différents modules du système (frontend, backend, base de données, etc.) et comment ils interagissent. Ce diagramme aide à visualiser l'architecture technique du CMS, en détaillant les dépendances entre les composants.



**Figure 7:** xxxxxxx

* + **Wireframes :**

Les wireframes sont des maquettes simplifiées qui montrent la disposition des éléments sur les pages principales du CMS. Ils permettent de visualiser l'interface utilisateur avant le développement et de s'assurer que l'expérience utilisateur (UX) est optimale.

**A) Page d'accueil :**

Cette page présente le tableau de bord avec les principales actions disponibles pour les utilisateurs. Les administrateurs peuvent y accéder aux statistiques et gérer les utilisateurs, les produits, et les abonnements.

* Insérer ici le wireframe de la page d’accueil.

**B) Page de gestion du contenu :**

Cette page permet aux administrateurs et éditeurs de créer, modifier et supprimer du contenu (textes, images, vidéos). Elle inclut un éditeur WYSIWYG et un aperçu en temps réel du contenu.

* Insérer ici le wireframe de la page de gestion du contenu.

**C) Page de gestion des produits :**

Cette page permet de gérer les produits, y compris l'ajout, la modification et la suppression des produits, ainsi que la gestion des stocks et des prix. Elle inclut également une synchronisation automatique avec la marketplace.

* Insérer ici le wireframe de la page de gestion des produits.

## 4.3 Technologies utilisées :

Le développement du module CMS de Weefarm repose sur une stack technologique moderne, choisie pour sa robustesse, sa performance et sa compatibilité avec les standards actuels du développement web. Ces technologies ont permis d’assurer une architecture modulaire, évolutive et responsive, tout en garantissant une bonne expérience utilisateur.

* + **Frontend :**
    - **React.js :** Framework JavaScript utilisé pour construire des interfaces utilisateur dynamiques et réactives. Il permet une mise à jour rapide de l’affichage grâce à son DOM virtuel.
    - **Tailwind CSS :** Framework CSS utilitaire qui facilite la conception d’interfaces cohérentes et modernes avec une grande souplesse de personnalisation.
    - **HTML5 s CSS3 :** Technologies de base pour la structuration et la mise en forme des pages web.
    - **JavaScript :** Langage de programmation client utilisé pour gérer les interactions et la logique côté navigateur.
  + **Backend :**
    - **Node.js :** Environnement d’exécution JavaScript côté serveur, utilisé pour gérer les requêtes, le logique métier et les opérations serveur.
    - **Express.js :** Framework minimaliste pour Node.js, permettant de créer des APIs REST performantes et bien structurées.
  + **API & Documentation :**
    - **Swagger (OpenAPI) :** Outil de documentation et de test d’API REST, facilitant la communication entre le frontend et le backend ainsi que l’intégration avec d’autres systèmes (comme la marketplace Weefarm).
  + **Base de données :**
    - **MySQL :** Système de gestion de base de données relationnelle choisi pour sa fiabilité, sa compatibilité avec Node.js, et sa capacité à gérer des données structurées complexes (utilisateurs, contenus, abonnements, produits, etc.).
  + **Outils complémentaires :**
    - **VS Code :** Environnement de développement principal utilisé pour coder, tester et déployer l'application.

# CHAPITRE 5 : Développement et Implémentation

# 5. Développement et Implémentation :

## 5.1 Frontend : interfaces et fonctionnalités :

Le développement de la partie frontend du CMS a été réalisé en utilisant React.js avec Tailwind CSS pour garantir une interface moderne, responsive et accessible sur tous types de supports (desktop, tablette, mobile). L’objectif principal était de proposer une interface utilisateur intuitive et modulaire permettant aux sociétés agricoles de gérer leur site vitrine en toute autonomie.

Parmi les interfaces clés développées :

* **Tableau de bord personnalisé** : affichage des statistiques du site vitrine (visites, produits, services, etc.).
* **Gestion des pages :** création, modification et suppression de sections telles que “À propos”, “Services”, “Actualités”, etc.
* **Éditeur WYSIWYG :** permet aux utilisateurs de formater leurs contenus facilement (textes, images, vidéos, documents).
* **Personnalisation graphique :** choix des couleurs, typographies, thèmes, et logos.
* **Formulaires dynamiques :** ajout et gestion de contenus structurés (formulaires de contact, témoignages, FAQ…).
* **Prévisualisation en temps réel :** aperçu direct des modifications avant publication.

Chaque interface a été conçue dans une logique de modularité et de réutilisabilité, ce qui facilite la maintenance et l’évolution du système.

## 5.2 Backend : API et base de données :

Le backend repose sur Node.js avec Express.js, permettant une architecture RESTful bien structurée. Les différentes routes API gèrent toutes les opérations CRUD (Create, Read, Update, Delete) relatives aux utilisateurs, contenus, produits, abonnements, etc.

Les fonctionnalités implémentées côté serveur incluent :

* **Gestion des rôles et permissions :** contrôle d’accès selon les profils (admin, éditeur, visiteur…).
* **Gestion du contenu** : endpoints pour la création, l’édition et la suppression de contenus dynamiques.
* **Synchronisation avec la marketplace :** intégration automatisée des produits/services dans la marketplace Weefarm.
* **Gestion des abonnements et paiements :** suivi des formules d’abonnement, génération des factures et notifications de renouvellement.
* **Statistiques et tableaux de bord :** agrégation de données pour analyse des performances.

La base de données MySQL a été structurée en plusieurs tables relationnelles couvrant l’ensemble des modules du CMS (utilisateurs, rôles, sites, pages, contenus, médias, produits, etc.). Des relations claires ont été établies pour assurer la cohérence et la rapidité des requêtes.

## 5.3 Sécurité et optimisation des performances :

La sécurité du CMS a été abordée de manière proactive pour protéger les données des utilisateurs et prévenir les attaques courantes. Les principales mesures mises en place sont :

* Authentification sécurisée via JWT (JSON Web Token).
* Hachage des mots de passe avec bcrypt.
* Validation des entrées côté client et serveur pour éviter les injections SQL/XSS.
* Contrôle d’accès par rôle pour sécuriser les endpoints sensibles.

En ce qui concerne les performances, plusieurs optimisations ont été effectuées :

* Lazy loading des composants React pour améliorer le temps de chargement.
* Mise en cache des requêtes les plus fréquentes.
* Pagination et filtrage sur les appels API pour éviter les surcharges serveur.
* Optimisation des requêtes SQL et utilisation d’index sur les champs critiques.
* Compression des fichiers (images, CSS, JS) avant envoi.

Ces mesures ont permis d’obtenir un CMS rapide, sécurisé, et capable de gérer efficacement un grand volume de contenus et d'utilisateurs.

# CHAPITRE 6 : TESTS ET VALIDATIONS