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

Description générée automatiquement

**Pilotage et management des équipes projet**

**Préparation d’un plan de changement de périmètre avec un dossier de planification**

*Youssef Berhayla et Matthieu Roignant*

*Dernière mise à jour : 20/01/2025*

Avec un peu plus d’un siècle d’existence, la fromagerie DIGICHEESE est une entreprise familiale d’environ 130 salariés. Ses produits sont essentiellement vendus aux grands distributeurs ainsi que dans la boutique d’usine pour les clients se rendant physiquement sur place. Depuis plus de 20 ans, une application de gestion développée sur Microsoft Access 2000 est utilisée comme outil de gestion. Au fil des années, Microsoft a arrêté le développement et les mises à jour d’Access 2000 entraînant l’obsolescence des outils de gestion de DIGICHEESE et l’impossibilité d’améliorations et de modernisation.

La demande de DIGICHEESE est de développer une nouvelle application de gestion à des fins de modernisation, d’ergonomie, d’efficacité et de pouvoir envisager de nouvelles fonctionnalités et une expansion de leur activité. Une modernisation des bases de données actuelle est également attendue.

1. **Pilotage et management des équipes projets**
2. **DIGICHEESE - Présentation du client**

Pour ce projet, l’entreprise DIGICHEESE est composée de :

* Christophe Germain, Directeur DSI
* Administrateur
* Opérateur Stock
* Opérateur colis

Par l’intermédiaire de son Directeur DSI, DIGICHEESE demande une refonte de son système de gestion. DIGICHEESE définit les besoins et les attentes en matière de fonctionnalités et de performance de la nouvelle application. L’entreprise fournit également les accès aux ressources nécessaires pour le bon développement de l’application.

Le Directeur DSI de DIGICHEESE, en la personne de Christophe Germain, supervise l’avancée du projet de refonte de l’application au nom de DIGICHEESE. Le directeur gère la relation avec l’équipe de développement des nouveaux outils et s’assure du respect des exigences techniques demandées ainsi que les objectifs stratégiques et de l’adéquation de l’application avec les besoins des utilisateurs finaux.

Les rôles utilisateur finaux de l’application sont au nombre de trois : Opérateur stock, Opérateur colis et Administrateur

L’administrateur a pour rôle de gérer les utilisateurs, de définir les paramètres de l’application et de mettre à jour certaines bases de données (poids, poidsV, objet, conditionnement et commune). L’administrateur a la possibilité d’ajouter, de modifier et de supprimer dans chacune de ses affectations ainsi que de réaliser des impressions au format papier.

L’opérateur colis a comme tâche principale la gestion des clients, des commandes et du conditionnement. Il a accès à la visualisation de la liste client, de la fiche client, les colis en cours, la liste des emballages, la relation poids/colis, la relation poids-vignette/colis et les différentes statistiques. Le mailing client, l’envoi d’emails personnalisé et l’impression au format papier font également parti de ses affectations.

L’opérateur stock gère et met à jour annuellement les stocks de l’entreprise. Il peut également générer des impressions au format papier.

1. **DIGIDEV - Equipe de développement**

L’équipe de DIGIDEV est composée :

* Robin Hotton, Product Owner
* Développeur FrontEnd
* Développeur BackEnd
* Développeur Full-Stack
* Equipe QA/Tests

DIGIDEV, représenté par le Product Owner, est l’entreprise chargée du développement de l’application. L’entreprise fournit les ressources techniques nécessaires au développement de la nouvelle application de DIGICHEESE

Robin Hotton est le Product Owner en charge du projet de refonte des outils de gestion de DIGICHEESE. Le Product Owner élabore et gère le backlog du produit. Il s’assure que la qualité et des performances de la nouvelle application soit en adéquation avec les besoins des utilisateurs finaux. Un de ses rôles est de communiquer régulièrement avec le Directeur DSI de DIGICHEESE pour s’assurer de la satisfaction du client et de la bonne avancée du projet en validant des livrables à chaque étape. Enfin, il s’assure que la livraison du projet se fasse dans les délais et budget convenu avec DIGICHEESE.

L’équipe DIGIDEV est composée d’un développeur FrontEnd, d’un développeur BackEnd, d’un développeur Full-Stack et d’une équipe QA/Tests.

Le développeur FrontEnd a la charge du développement de la partie design, esthétique et ergonomique de l’interface de l’application à l’aide de React.

Le développeur BackEnd s’occupe de la conception de l’application via Django et de la mise en place des nouvelles bases de données à l’aide de MySQL.

Le développeur Full-Stack met en place les nouveaux outils de gestion sur les serveurs de l’entreprise.

L’équipe QA/Tests rédige et exécute les plans de tests afin d’identifier et documenter les anomalies rencontrées lors des tests de la nouvelle application. Après retour aux développeurs pour correction, elle s’assure de la validation des corrections avant le déploiement de l’application.

1. **Définition des objectifs**

L’objectif de ce projet est de réaliser un site interne à l’entreprise remplaçant les outils de gestion actuels obsolètes et d’améliorer les bases de données existantes.

Le nouveau site de l’entreprise sera divisé en quatre parties :

* Authentification : page de connexion au site, accessible à tous les utilisateurs
* Administration : accessible pour les rôles « Administrateur »
* Gestion des colis : pour la gestion des colis, accessible pour les rôles « OP-colis »
* Gestion des stocks : pour la gestion des stocks, accessible pour les rôles « OP-stock »

Les utilisateurs pourront cumulés les rôles dans la nouvelle application.

1. Authentification

Via cette page, les utilisateurs pourront se connecter au site interne de DIGICHEESE en utilisant leur identifiant pour accéder au session correspondante (Administrateur, OP-colis ou OP-stock).

Cette authentification renforcera la sécurité des données relatives à l’entreprise et aux clients.

1. Administration

Cet espace permettra aux utilisateurs autorisés de gérer (ajout, modification, suppression) les utilisateurs, les paramètres métiers (poids, poidsV, objet, conditionnement et commune) et l’impression au format papier.

1. Gestion des colis

Dans cette session, les utilisateurs pourront gérer les clients, les commandes et le conditionnement, d’avoir accès à diverses listes (client, colis, emballage, relation poids/vignette, relation poids-vignette/colis) et d’avoir la possibilité d’envoi d’emails aux clients. Ils pourront également imprimer au format papier.

1. Gestion des stocks

Les utilisateurs OP-stock auront la charge de gérer le stock et de le mettre à jour annuellement.

*Une importance sera apportée sur la praticité et l’esthétique pour la nouvelle application de gestion, le site interne et la mise à jour des bases de données existantes.*

1. **Organisation du projet**
2. Compétences nécessaires et répartition des rôles

Afin de mener à bien ce projet, une répartition claire des responsabilités entre les différents profils de développeurs a été définie :

Développeur FrontEnd

* Prise en charge du développement de l’esthétique et de l’interface utilisateur pour le nouveau site interne de DIGICHEESE
* Conception et développement de l’application de gestion en React, en respectant les exigences en termes d’expérience utilisateur (UX) et de design (UI).
* Collaboration avec les autres membres de l’équipe pour intégrer les aspects visuels aux fonctionnalités BackEnd
* Assurer la compatibilité et la performance de l’interface sur les différents navigateurs et plateformes

Développeur BackEnd

* Développement de la logique métier et des fonctionnalités BackEnd pour la nouvelle application de gestion, en utilisant le framework Django
* Intégration des exigences spécifiques de DIGICHEESE dans les processus métiers et la gestion des données
* Mise à jour et migration des anciennes bases de données vers des versions modernisées, tout en assurant la compatibilité avec les nouveaux systèmes

Développeur Full-Stack

* Collaboration avec le développeur BackEnd dans les phases initiales du projet, notamment pour la création et l’intégration des fonctionnalités de l’application
* Supervision du processus de déploiement de l’application sur le site interne de DIGICHEESE
* Rédaction de la documentation technique complète (architecture et spécifications techniques, processus d’installation et de maintenance, bonnes pratiques pour l’utilisation et l’évolution de l’application)

Equipe de testeurs

* Mise en œuvre des phases de tests
  + Tests unitaires pour vérifier la qualité du code
  + Tests d’intégration pour garantir une communication fluide entre les différentes parties de l’application
  + Tests de performance et de charge pour évaluer la robustesse de la solution
* Identification et documentation des bugs ou problèmes détectés durant les tests
* Collaboration avec l’équipe de développement pour corriger les erreurs et valider les modifications apportées

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tâche | Product Owner | Développeur FrontEnd | Développeur BackEnd | Développeur FullStack | Equipe QA/Tests | DIGICHEESE |
| Conception du visuel de l’application | C | R | C | I | I | A |
| Développement de l’application | C | I | R | I | I | A |
| Mise à niveau des bases de données | C | I | R | I | I | A |
| Aide au développement BackEnd | I | I | I | R | I | I |
| Déploiement de l’application sur site | C | I | I | R | I | A |
| Tests de l’application | C | I | I | I | R | A |
| Correction du code | C | I | R | C | C | A |
| Rédaction de la documentation technique | C | I | C | R | I | A |

1. Matrice RACI pour l’organisation du projet

|  |  |
| --- | --- |
| Réalisateur (R) : celui qui effectue le travail | Approbateur (A) : celui qui valide les décisions |
| Consulté (C) : celui qui fournit des informations | Informé (I) : celui qui est tenu au courant |
|  |  |

1. Suivi et motivation de l’équipe

JIRA

JIRA est l’outil principal utilisé pour organiser et suivre l’avancement des tâches du projet. Il permet de visualiser la progression de chaque membre de l’équipe, de s’assurer du respect des délais et d’identifier rapidement les éventuels retards. Chaque phase du projet est structurée en sprints d’une durée de deux semaines, ce qui favorise une gestion itérative et flexible. L’outil centralise également les priorités, les dépendances et les objectifs des tâches assignées.

Suivi de l’équipe

Afin de maintenir une dynamique positive et de garantir la qualité des livrables, plusieurs méthodes de suivi et d’évaluation réguliers seront mis en place.

1. *Réunion de planification de sprint (Sprint planning)*

Une session sera organisée au début de chaque sprint pour définir les objectifs à atteindre et les tâches à réaliser.

L’équipe participera à l’estimation des efforts nécessaires pour chaque tâche, en se basant sur les priorités définies par le product owner.

Cette réunion permet de clarifier les attentes et de garantir que chaque membre sait précisément ce qu’il doit accomplir.

1. *Daily Scrum*

Une réunion quotidienne (15 min) sera tenue entre le product owner et tous les membres impliqués dans le projet.

L’objectif est de permettre à chacun de partager l’état d’avancement de ses tâches, de signaler d’éventuels blocages et de synchroniser les efforts de l’équipe.

Ces échanges favorisent la transparence et permettent d’anticiper les ajustements nécessaires au planning.

1. *Revue hebdomadaire de code*

Une réunion dédiée sera organisée chaque semaine pour évaluer les contributions de chaque développeur (1 heure).

Les critères évalués incluront la qualité du code, les respects des standards définis et l’intégration fluide avec les autres parties du projet.

Cette revue permettra également de partager des bonnes pratiques et de renforcer les compétences techniques des membres de l’équipe.

1. *Revue de sprint (Sprint Review)*

A la fin de chaque sprint, une réunion récapitulative sera organisée (1 à 2 heures).

Tous les membres de l’équipe participeront pour discuter des résultats obtenus, partager des retours constructifs et identifier les points d’amélioration pour les prochaines itérations.

Un compte-rendu précis de cette revue sera rédigé, incluant les progrès réalisés, les obstacles rencontrés et les décisions prises. Ce document sera ensuite transmis à DIGICHEESE pour assurer un suivi transparent avec les parties prenantes externes.

1. *Rétrospective de sprint (Sprint retrospective)*

Une rétrospective sera réalisée après chaque sprint pour analyser ce qui a bien fonctionné, ce qui doit être amélioré et les leçons à tirer pour les prochains sprints.

L’équipe participera activement à identifier les forces et faiblesses du processus de développement.

Des actions concrètes seront décidées pour optimiser l’efficacité et la collaboration. Un compte rendu sera rédigé et transmis à DIGICHEESE.

1. Plan de montée en compétences

Développeur FrontEnd

Compétences acquises :

* Maitrise de React pour le développement d’interfaces utilisateur
* Connaissance des bonnes pratiques en matière de conception UX/UI

Objectifs d’amélioration (formations certifiantes et wébinaires) :

* Amélioration des compétences en React
* Intégration de nouvelles technologies

Développeur BackEnd

Compétences acquises :

* Maitrise de Django pour le développement de l’application
* Connaissance approfondie de MySQL pour la gestion des bases de données

Objectifs d’amélioration (formations certifiantes et wébinaires) :

* Maitrise avancée de Django, approfondissement des connaissances
* Se former sur les techniques modernes de migrations de données

Développeur Full-Stack

Compétences acquises :

* Compétences combinées en développement Front-End (React) et Back-End (Django)
* Expérience dans le déploiement et la gestion des applications sur les serveurs

Objectifs d’amélioration (formations certifiantes et wébinaires) :

* Approfondir la connaissance des architectures Cloud
* Se former sur les pratiques DevOps pour automatiser les processus de déploiement et de gestion des versions

Equipe QA/tests

Compétences acquises :

* Connaissance des principes de base des tests unitaires et d'intégration
* Expérience dans l'utilisation d'outils de gestion des tests

Objectifs d’amélioration (formations certifiantes et wébinaires) :

* Formation approfondie sur les frameworks de tests automatisés
* Mise en œuvre de stratégies de tests automatisés pour accélérer les cycles de tests

1. **Communication**
2. Au sein de l’équipe

Une communication fluide et constante entre les membres de l’équipe est essentielle pour garantir l’efficacité et la réussite du projet.

1. Echanges entre le product owner et les développeurs

Le product owner communique régulièrement avec chaque développeur pour :

* Suivre l’état d’avancement des tâches en cours
* Identifier rapidement les obstacles rencontrés par chacun et apporter un soutien ou des solutions
* Valider les fonctionnalités livrées au fil des sprints pour s’assurer qu’elles respectent les besoins des utilisateurs finaux

Ces échanges se font au travers des Daily Scrum et des points individuels lorsque nécessaire.

1. Echanges entre les développeurs FrontEnd, BackEnd et Full-Stack

Les développeurs travaillent en étroite collaboration pour garantir une intégration harmonieuse des différentes couches de l’application.

1. Echanges entre les développeurs et l’équipe QA/Tests

En fin de projet, l’équipe QA/Tests travaillera étroitement avec le développeur BackEnd pour :

* Rapporter les anomalies et bugs détectés durant les tests
* Discuter des solutions possibles pour corriger ces problèmes
* Valider les corrections appliquées et s’assurer que les modifications n’introduisent pas de nouveaux dysfonctionnements.

Cette collaboration s’étendra également au développeur Full-Stack pour la validation finale de l’application.

1. Avec DIGICHEESE

Une communication claire et régulière entre DIGICHEESE et DIGIDEV est essentielle pour assurer le succès du projet.

Le product owner est le principal point de contact entre les deux entités :

* Transmet les besoins et priorités définis par DIGICHEESE à l’équipe de développement
* Recueille les retours de DIGICHEESE à chaque étape clé et valide les livrables en conséquence
* Organise des réunions avec le directeur DSI de DIGICHEESE, Christophe Germain, pour faire le point sur l’avancement du projet, les défis rencontrés et les ajustements nécessaires.

Des réunions régulières entre DIGICHEESE et DIGIDEV seront organisées :

* Réunion initiale pour définir les objectifs globaux, les priorités fonctionnelles, les contraintes techniques et les attentes de DIGICHEESE entre le product owner, le directeur DSI et les responsables techniques de DIGIDEV
* Réunion hebdomadaire entre le product owner et le directeur DSI pour suivre l’avancement du projet et définir les priorités à court terme
* Réunion de validation finale afin de vérifier que l’ensemble des livrables respecte les exigences initiales avant le déploiement. Cette réunion se fera en présence du product owner, des développeurs et du directeur DSI.

DIGIDEV fournira des rapports détaillés après chaque sprint incluant les fonctionnalités développées, les problèmes rencontrés et résolus et les tâches à venir et les ajustements nécessaires.

DIGICHEESE peut proposer des ajustements ou des changements prioritaires auprès du product owner qui analysera les demandes et évaluera leur impact sur le planning et le budget.

1. **Ressources matérielles et architecture logiciels**

Matériel

* Ordinateurs performants pour les développeurs équipés d’environnement de développement intégrés adaptés (VS Code, Linux)
* Serveurs de tests et de production pour héberger les environnements applicatifs et les bases de données (Nginx)
* Connexion internet stable et sécurisée

Logiciels

*Outils de développement*

* FrontEnd : React, Node.js, TypeScript
* BackEnd : Django, Python, MySQL
* Tests : Jest, Pytest

*Outils de collaboration*

* Gestion de projet : JIRA
* Communication : Slack, Teams

*Contrôle de version*

* Git
* GitHub

1. **Macroplanning**

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

Description générée automatiquement

1. **Dossier de planification d’un projet informatique**
2. **Les phases**
3. Développement

Objectif principal

La phase de développement a pour but de concevoir et de mettre en œuvre une application moderne et fonctionnelle en utilisant Django pour le backend et React pour le frontend.

Django a été choisi pour sa robustesse, sa sécurité intégrée et ses fonctionnalités complètes facilitant le développement rapide d’applications complexes. React a été retenu pour sa capacité à créer des interfaces utilisateur dynamiques, modulaires et réactives, améliorant ainsi l’expérience utilisateur.

Cette phase inclut la création des modules clés, le développement des API nécessaires à la communication entre les deux couches, et la réalisation de tests unitaires pour garantir la qualité du code.

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

Description générée automatiquementModules à développer

**Module Administrateur**

* **Fonctionnalités** :
  + Gestion des utilisateurs : création, modification, suppression.
  + Gestion des bases de données (communes, objets, conditionnements, etc.).
  + Accès aux statistiques, notamment les connexions des utilisateurs.
  + Exportation et impression des données administratives.
* **Tâches spécifiques** :
  + Développement des API Django REST Framework pour les opérations CRUD.
  + Intégration des fonctionnalités d’exportation et d’impression.

**Module Gestion des Colis**

* **Fonctionnalités** :
  + Gestion des commandes, clients et conditionnements.
  + Suivi des commandes et envoi d’e-mails personnalisés aux clients.
  + Accès aux statistiques des ventes et des goodies.
* **Tâches spécifiques** :
  + Création d’API REST pour gérer les données clients, commandes et conditionnements.
  + Mise en place d’un système d’envoi d’e-mails via le backend Django.

**Module Gestion des Stocks**

* **Fonctionnalités** :
  + Consultation et mise à jour des stocks.
  + Impression de listes d’inventaire et de fiches de produits.
* **Tâches spécifiques** :
  + Développement des API pour gérer les inventaires dans Django.
  + Intégration de mises à jour en temps réel via WebSockets ou des requêtes asynchrones.

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

Description générée automatiquement

Création du Frontend et Intégration Frontend-Backend

**Fonctionnalités** :

* Développement d’une interface utilisateur réactive et dynamique avec React.
* Création d’espaces dédiés pour chaque équipe :
  + **Équipe Administration** : Tableau de bord pour gérer les utilisateurs, les bases de données, et accéder aux statistiques.
  + **Équipe Colis** : Interface intuitive pour gérer les commandes, les clients et envoyer des emails personnalisés.
  + **Équipe Stock** : Vue dédiée pour la gestion des stocks, leur mise à jour et l’impression des inventaires.
* Communication fluide entre le frontend et le backend via des API RESTful.
* Gestion des flux de données en temps réel pour améliorer l’interactivité.

**Tâches spécifiques** :

* Création et documentation des endpoints nécessaires avec Django REST Framework pour des opérations comme la récupération des détails d’une commande et la mise à jour des utilisateurs.
* Implémentation des appels asynchrones dans React avec Axios ou Fetch API, intégrant des indicateurs de chargement pour une meilleure expérience utilisateur.
* Réalisation de tests d’intégration complets pour vérifier la cohérence et la fiabilité des interactions entre le frontend et le backend.

Une image contenant texte, diagramme, ligne, Police

Description générée automatiquement

Architecture Global de l’Application

Une image contenant texte, capture d’écran, conception

Description générée automatiquement

Chronologie

La phase de développement est prévue sur **11 semaines**, réparties comme suit :

* **Semaines 1 à 4** : Développement du module administrateur
* **Semaines 5 à 7** : Développement des modules gestion des colis
* **Semaines 8 à 9 :** Développement des modules gestion des stocks
* **Semaines 9 à 11 :** Création du Frontend et intégration complète du frontend et du backend, suivi de tests approfondis pour garantir la stabilité.

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

Description générée automatiquement

Livrables attendus

* Interfaces fonctionnelles pour chaque module
* API backend entièrement opérationnelles
* Tests unitaires validés pour chaque fonctionnalité

Outils et technologies

* **Outils de développement** : VS Code, Git, Docker
* **Tests** : Jest pour le frontend, Pytest pour le backend

Points critiques à surveiller

* Synchronisation des équipes frontend et backend pour garantir une intégration fluide, en utilisant des outils tels que Jira pour le suivi des tâches, des réunions hebdomadaires pour coordonner les avancées, et Git pour la gestion centralisée des versions du code.
* Respect des délais pour chaque sprint afin d’éviter un effet domino sur les phases suivantes.

1. Tests et Déploiement

Objectif principal

Garantir que l'application répond aux exigences fonctionnelles et techniques, qu'elle est stable et sécurisée, et qu'elle peut être déployée avec succès dans l'environnement de production. Cette phase inclut des tests approfondis, la préparation de l'infrastructure, et la mise en production.

Étape de Tests

1. Types de tests réalisés :

**Tests fonctionnels :**

* Vérification de chaque fonctionnalité en fonction des spécifications (CRUD, gestion des utilisateurs, commandes, stocks).
* Tests basés sur des cas d'utilisation réels pour garantir que chaque module répond aux besoins des utilisateurs finaux.

**Tests de performance :**

* Analyse des temps de réponse pour les opérations critiques (connexion, requêtes API, chargement des pages).
* Simulation de scénarios de forte charge pour vérifier la capacité du système à supporter un grand nombre d’utilisateurs simultanés.

**Tests de sécurité :**

* Validation des mécanismes d’authentification (JWT) et des protections contre les attaques (XSS, CSRF, injection SQL).
* Analyse de la robustesse face aux tentatives d’accès non autorisées.

**Tests d'intégration :**

* Vérification de la cohérence des flux de données entre le frontend et le backend via les APIs.
* Simulation d’interactions complètes entre les modules pour détecter les éventuelles incompatibilités.

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

Description générée automatiquement

1. Livrables des tests :

* Rapport détaillé des anomalies détectées, classées par priorité.
* Plan de correction avec un calendrier pour chaque anomalie.
* Validation finale par les parties prenantes avant le déploiement.

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

Description générée automatiquement

Étape de Déploiement

**Préparation de l'infrastructure :**

* **Serveurs :** Configuration des serveurs sous Linux (Ubuntu) avec NGINX pour l’hébergement.
* **Conteneurs :** Déploiement des composants backend et base de données dans des conteneurs Docker pour une gestion simplifiée et une meilleure isolation.
* **Sécurité :** Mise en place de certificats SSL pour activer HTTPS et garantir des échanges sécurisés.

**Migration des données :**

* Transfert des données critiques (utilisateurs, commandes, stocks) depuis l'ancien système vers la nouvelle base de données MySQL.
* Double vérification de l’intégrité et de la cohérence des données migrées.

**Mise en production :**

* Déploiement progressif pour limiter les risques.
* Surveillance active pour détecter rapidement les anomalies.

**Formation des utilisateurs :**

* Organisation de sessions pratiques adaptées aux administrateurs et opérateurs.
* Mise à disposition de guides détaillés pour garantir une prise en main rapide.

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

Description générée automatiquement

Maintenance et Améliorations

**Objectif principal**

Assurer la stabilité de l’application après le déploiement, répondre aux besoins émergents des utilisateurs, et mettre en œuvre des améliorations basées sur les retours.

1. Maintenance

**Suivi post-déploiement :**

* Surveillance des performances en temps réel (temps de réponse, utilisation des ressources).
* Détection et correction immédiates des bugs ou dysfonctionnements.

**Support utilisateur :**

* Mise en place d’un système de Ticketing pour collecter les retours et signalements.
* Résolution des problèmes signalés dans des délais définis (ex. : 24h pour les problèmes critiques).

**Sauvegardes et sécurité :**

* Planification de sauvegardes régulières des données pour éviter toute perte.
* Mise à jour des mécanismes de sécurité en fonction des nouvelles menaces identifiées.

1. Améliorations

* Envoi de questionnaires et organisation de réunions pour collecter les suggestions d’amélioration.

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

Description générée automatiquement

1. **Les ressources**
2. Budget prévisionnel

Une image contenant texte, ligne, diagramme, nombre

Description générée automatiquement

1. Le diagramme de GANTT

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

Description générée automatiquement

1. Ressources humaines

Pour ce projet, un développeur FrontEnd, un développeur BackEnd, un développeur Full-Stack et un ingénieur test seront utilisés.

Le développeur FrontEnd aura la charge du développement de l’esthétique et de l’interface utilisateur du nouveau site de DIGICHEESE en utilisant React tout en collaborant avec les autres membres de l’équipe pour intégrer les aspects visuels aux fonctionnalités BackEnd.

Le développeur BackEnd s’occupera du développement de la logique métier et des fonctionnalités BackEnd pour la nouvelle application de gestion, en utilisant le framework Django et en respectant les exigences spécifiques de DIGICHEESE. Il aura également la charge de la mise à jour et migration des anciennes bases de données vers des versions modernisées.

Le développeur Full-Stack sera en collaboration avec le développeur BackEnd dans les phases initiales du projet notamment pour l’intégration des fonctionnalités de l’application et il aura la charge du déploiement de l’application sur le site de DIGICHEESE. Il s’occupera de la rédaction de la documentation technique complète.

L’ingénieur test mettra en œuvre les phases de tests, identifiera et documentera les bugs rencontrés et vérifiera les corrections apportées par les développeurs.

1. **Bilan prévisionnel**
2. Objectifs et résultats attendus

**Objectifs principaux :**

* **Modernisation de l’application** : Remplacer l’ancien système par une solution moderne et performante, utilisant des technologies récentes (React, Django, MySQL).
* **Amélioration de l’efficacité** : Réduire les temps de traitement des opérations courantes.
* **Sécurisation des données** : Mettre en place une authentification robuste (via JWT), un chiffrement des données, et des protections contre les attaques courantes (XSS, SQL Injection).
* **Réduction des erreurs humaines** : Automatiser les processus critiques (gestion des stocks, commandes, utilisateurs) pour minimiser les risques d’erreurs.
* **Évolutivité intégrée** : Concevoir une architecture modulaire permettant l’ajout de nouvelles fonctionnalités sans impact sur les performances existantes.

**Résultats escomptés :**

* Une application performante, intuitive et rapide, répondant aux besoins des utilisateurs finaux.
* Une infrastructure stable et sécurisée, adaptée aux exigences actuelles et futures de l’entreprise.
* Une gestion simplifiée des processus internes, augmentant la productivité des équipes.
* Une architecture évolutive, permettant une adaptation aux besoins émergents sans effort excessif.

1. Bilan technique prévisionnel

**Forces anticipées :**

* **Technologies modernes** : Le choix de React pour le frontend et Django pour le backend assurera une base technologique robuste, performante et évolutive.
* **Intégration fluide** : Les différents modules (administration, gestion des colis, gestion des stocks) seront connectés via des APIs RESTful bien structurées.
* **Base de données optimisée** : MySQL sera configuré pour gérer efficacement les données critiques tout en garantissant leur sécurité et leur cohérence.
* **Infrastructure solide** : L’utilisation de Docker et NGINX permettra une gestion simplifiée et une stabilité accrue pour le déploiement et la maintenance.
* **Tests rigoureux** : La mise en place de tests fonctionnels, de performance, de sécurité et d’intégration garantira une validation complète avant le déploiement.

**Défis identifiés :**

* **Migration des données** : La transition des données de l’ancien système vers la nouvelle base de données pourrait nécessiter des efforts supplémentaires pour garantir leur intégrité.
* **Gestion des retours utilisateurs** : L’intégration des retours des différents profils d’utilisateurs (administrateurs, opérateurs colis et stocks) devra être anticipée pour ajuster l’interface et les fonctionnalités.
* **Contrainte de temps** : Le respect des délais sera un défi, notamment lors de la phase de correction des anomalies.

1. Bilan méthodologique prévisionnel

**Points forts prévus :**

* **Méthode Agile** : L’utilisation de sprints, de réunions régulières et d’une gestion itérative permettra de s’adapter rapidement aux besoins et aux imprévus.
* **Communication étroite** : Une coordination continue entre les équipes frontend, backend et QA assurera un déroulement fluide des phases de développement et de tests.
* **Documentation complète** : Une documentation technique et utilisateur sera créée tout au long du projet pour faciliter la prise en main et la maintenance de l’application.

**Améliorations anticipées :**

* **Planification des tests** : Une organisation minutieuse des tests permettra de réduire les délais associés à la correction des anomalies.
* **Formation des utilisateurs** : Des ateliers approfondis et des guides détaillés faciliteront l’adoption rapide du nouvel outil par les utilisateurs finaux.
* **Analyse des risques** : Une évaluation rigoureuse des risques en amont minimisera les imprévus pendant le développement et le déploiement.

**Conclusion**

Le projet prévoit de répondre aux besoins critiques de modernisation, de sécurisation et d’optimisation des processus internes. Grâce à une approche méthodique et à l’utilisation de technologies avancées, l’application prévue sera performante, évolutive et adaptée aux objectifs de l’entreprise. Les enseignements tirés de l'exécution de ce projet permettront également d'améliorer les pratiques pour les futurs développements.

1. **Annexe - Changement de planification**
2. **Changement de planning**

**Demande de DIGICHEESE :** La Fromagerie DIGICHEESE désire que la livraison soit décalée de 4 mois et qu’elle soit réalisée sur site en 45 jours avec un budget de 45 K € HT.

**Modifications :**

Initialement prévue à fin mai 2025, la livraison finale de la nouvelle application de gestion sera décalée à septembre comme le souhaite la Fromagerie DIGICHEESE.

1. **Changement de lot de livraison**

**Demande de DIGICHEESE :** La Fromagerie DIGICHEESE désire que la Gestion des colis et du profil « Administrateur » soient développés en priorité.

**Modifications :**

La Gestion des colis et le profil « Administrateur » sont les deux fonctionnalités principales déjà priorisées. Le module Gestion des stocks sera mis de côté dans un premier temps.

1. **Nouveau planning et budget**

Une image contenant texte, ligne, nombre, Police

Description générée automatiquement

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

Description générée automatiquement