|  |
| --- |
| **DIGICHEES**  **Cahier des charges**  *Rédigé le 09/04/2024 par*  *[Henri PIERRE, Angela CRUZ, Wasin ALSHAMI, David BILLON]*  *Dernière mise à jour : 15/04/2024* |

### 

## Sommaire

[Sommaire 2](#_Toc164074826)

[I. Contexte 4](#_Toc164074827)

[a. Origine du projet 4](#_Toc164074828)

[b. Etat des lieux 4](#_Toc164074829)

[c. Les exigences client 4](#_Toc164074830)

[d. Portée du projet 5](#_Toc164074831)

[II. Description du projet 6](#_Toc164074832)

[a. Objectifs globaux 6](#_Toc164074833)

[b. Descriptions des parties prenantes 7](#_Toc164074834)

[c. Contraintes et exigences spécifiques 8](#_Toc164074835)

[d. Lieux de réalisation du Projet 8](#_Toc164074836)

[III. Description fonctionnelle des besoins 8](#_Toc164074837)

[a. Profils utilisateurs 8](#_Toc164074838)

[b. Scénarios d’utilisation 9](#_Toc164074839)

[c. Fonctionnalités essentielles 13](#_Toc164074840)

[IV. Exigences fonctionnelles 16](#_Toc164074841)

[a. Description détaillée des fonctionnalités du système 16](#_Toc164074842)

[b. Cas d'utilisation 19](#_Toc164074843)

[c. Diagramme de séquence 20](#_Toc164074844)

[V. Exigences non-fonctionnelles 21](#_Toc164074845)

[a. Performance du système 21](#_Toc164074846)

[b. Sécurité des données 22](#_Toc164074847)

[c. Maintenance et évolutivité 22](#_Toc164074848)

[d. Indicateurs de suivi 22](#_Toc164074849)

[VI. Architecture technique 25](#_Toc164074850)

[a. Environnement de développement (DEV) 25](#_Toc164074851)

[c. Environnement de production (PROD) 26](#_Toc164074852)

[d. MPD et couche de persistance 27](#_Toc164074853)

[VII. Planning et Livrables 28](#_Toc164074854)

[a. Phases du projet 28](#_Toc164074855)

[b. Calendrier 29](#_Toc164074856)

[c. Livrables attendus à chaque étape 29](#_Toc164074857)

[d. Fréquence et durée des comités de pilotage 30](#_Toc164074858)

[VIII. Ressources 30](#_Toc164074859)

[a. Estimation des coûts 30](#_Toc164074860)

[b. Besoins en ressources humaines 31](#_Toc164074861)

[c. Matériels et logiciels nécessaires 32](#_Toc164074862)

[Matériel : 32](#_Toc164074863)

[Logiciels : 32](#_Toc164074864)

[IX. Risques / Contraintes / Dépendance 33](#_Toc164074865)

## Contexte

### Origine du projet

La TPE de la Fromagerie DIGICHEES souhaite moderniser son système d'information obsolète, visant à améliorer la performance, la maintenance, la collaboration et la sécurité. Le projet vise à concevoir une plateforme performante avec des temps de chargement rapides, une architecture modulaire utilisant Python pour faciliter les mises à jour. Il prévoit également des fonctionnalités de collaboration et d'intégration pour le partage d'informations et une extensibilité avec des outils externes. L'objectif est de garantir la confidentialité des données et de gérer efficacement le programme de point de fidélité.

### Etat des lieux

L'application actuelle, créée il y a plus de 20 ans via Access Office 2000, présente des limites majeures. Non pris en charge sur les versions récentes de Windows, elle a connu des mises à jour limitées avec des compétences restreintes. Le support pour Office 2000 a pris fin en 2009, et l'application est caractérisée par un client lourd peu performant, une interface dépassée et des possibilités d'amélioration limitées.

Cet état pose plusieurs problématiques, que le projet devra être amené à résoudre :

* Access Office 2000 n'est officiellement pas pris en charge sur Windows Vista ou les versions ultérieures de Windows
* Le support standard pour Office 2000 a pris fin le 30 juin 2004 et le support étendu a pris fin le 14 juillet 2009
* Interface rigide et dépassée
* Forte instabilité, dette technique importante qui engendre des bugs réguliers
* Maintenance compliquée à effectuer étant donné la technologie utilisée
* Client lourd peu performant en rapport aux technologies actuelles
* Peu/pas de modification et/ou amélioration possibles
* Design et graphisme mal structuré, inesthétique et peu ergonomique

L’objectif est de réaliser une refonte complète d’une application de gestion en prenant en compte les problèmes relevés ci-dessus.

### Les exigences client

* **Plateforme performante :**
  + Nous visons des temps de chargement des pages rapides et fluides pour une expérience utilisateur optimale. Nous travaillerons à optimiser les performances de l'application pour garantir des temps de réponse courts, en utilisant des bonnes pratiques de développement et des techniques d'optimisation de la vitesse de chargement des pages.
* **Maintenabilité de la solution :**
  + Nous privilégierons l'utilisation du langage Python pour le développement, afin de bénéficier de sa simplicité, de sa lisibilité et de sa large gamme de bibliothèques et de frameworks disponibles. Le code sera structuré de manière modulaire pour faciliter les mises à jour et les évolutions futures du système.
* **Collaboration :**
  + Nous concevrons le système avec des fonctionnalités de partage d'informations et de collaboration entre les utilisateurs, en veillant à ce que les permissions d'accès soient bien gérées. De plus, nous assurerons que le système soit prêt à intégrer facilement des outils externes lorsque cela sera nécessaire, en utilisant des normes d'intégration telles que les API RESTful.
* **Priorité :**
  + Le client “La Fromagerie Digichees” désire que la gestion des colis et du profil “administrateur” soient développés en priorité. Il désire que ce soit réalisé en 30 jours maximum à partir du lancement du projet.

### Portée du projet

Le projet de refonte du site intranet de la Fromagerie DIGICHEES se fera en suivant une approche agile et itérative. Il inclut la conception et le développement d'une nouvelle plateforme logicielle, ainsi que la migration des données et des fonctionnalités de l'ancienne application vers le nouveau système.

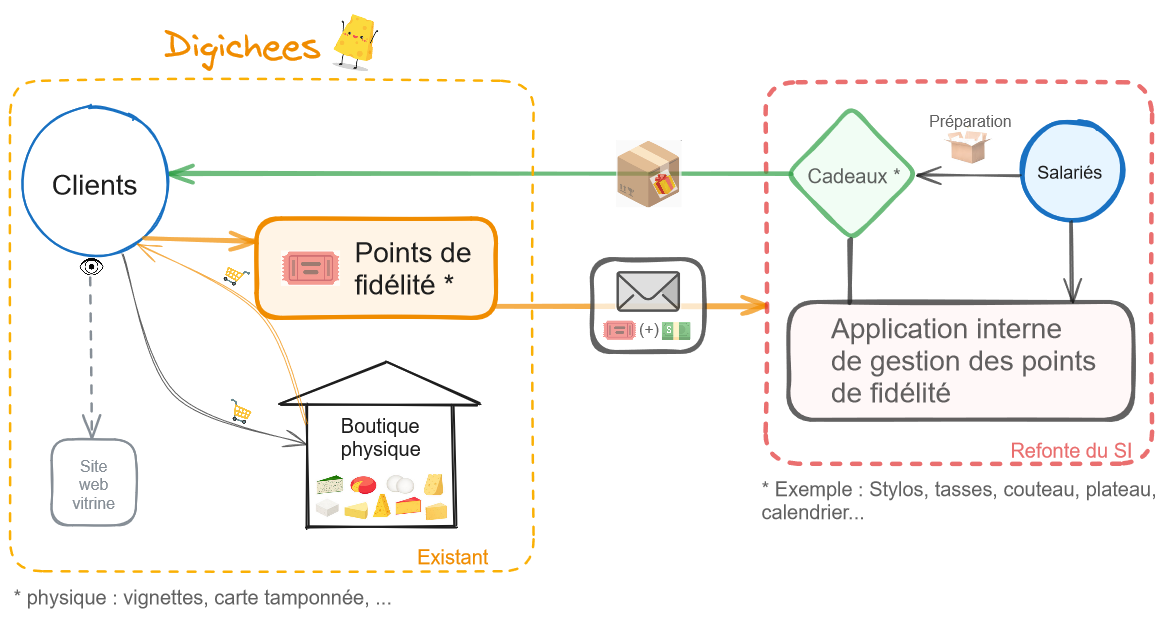
Ensuite, le projet prévoit la mise en place de fonctionnalités d’amélioration des process, d’une meilleure collaboration, la formation du personnel et la gestion du changement pour assurer une transition réussie vers le nouveau système.

La portée couvre donc l'ensemble du processus de modernisation du système d'information, depuis la conception et le développement jusqu'à la mise en œuvre et l'adoption par les utilisateurs, avec pour objectif d'améliorer les performances, la maintenabilité et l'ergonomie de l'application.

Le périmètre initial du projet se concentre exclusivement sur la création du site intranet pour la Fromagerie DIGICHEES. Par conséquent, le projet exclut toute intervention sur le site marketing de la fromagerie, qui reste en dehors de notre champ d'action.

## Description du projet

La section "Description du Projet" offre un aperçu détaillé de l'initiative de modernisation de l'intranet de gestion de la Fromagerie DIGICHEES.



### Objectifs globaux

Le projet de modernisation de l'intranet de gestion de la Fromagerie DIGICHEES vise à instaurer une transformation significative de son infrastructure informatique. L'objectif principal est de remplacer l'ancien système obsolète, développé il y a plus de 20 ans avec Access Office 2000, par une plateforme moderne et agile. Il s’agit de gérer les points de fidélité des clients de la fromagerie, qu’ils obtiennent en allant faire des achats dans la boutique physique. Ces mêmes points de fidélité sont utilisés pour acheter et commander des “goodies” (cadeaux, objets) en partie ou en totalité avec leurs points.

Cette initiative s'aligne avec la vision stratégique de l'entreprise, visant à renforcer sa compétitivité en offrant des outils informatiques performants, flexibles et évolutifs.

En se concentrant sur l'amélioration des performances, de la maintenabilité et de la convivialité, le projet vise à doter la Fromagerie DIGICHEES d'un intranet de gestion capable de soutenir ses opérations quotidiennes et de favoriser sa croissance future.

Nous pouvons les objectifs de manière globale comme suit :

* **Connexion en ligne :** Les utilisateurs pourront accéder à la plateforme via un navigateur web standard.
* **Authentification sécurisée :** L'accès nécessitera une authentification sécurisée avec des identifiants uniques et des niveaux d'accès définis par rôle. Aucun contenu ou fonctionnalité de la plateforme n'est accessible hors authentification.
* **Compatibilité multiplateforme :** La plateforme sera compatible avec différents systèmes d'exploitation et navigateurs web.
* **Adaptabilité aux appareils :** L'interface utilisateur sera adaptative pour une expérience optimale sur tous les appareils.
* **Accessibilité conforme aux normes :** La plateforme respectera les normes d'accessibilité Web pour garantir une expérience inclusive à tous les utilisateurs.

Ces modalités seront à compléter et à approfondir avec les parties prenantes lors de prochaines réunions avec le PO.

Les besoins fonctionnels seront soumis à évolution selon les besoins clients.

### Descriptions des parties prenantes

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom – Prénom** | **Rôle projet** | **Société** | **E-mail de contact** |
| Angela CRUZ | Product Owner proxy | Diginamic.fr | [acruz@diginamic-formation.fr](mailto:acruz@diginamic-formation.fr) |
| Henri PIERRE | Scrum Master | Diginamic.fr | [hpierre@diginamic-formation.fr](mailto:hpierre@diginamic-formation.fr) |
| Wasin ALSHAMI | Testeur | Diginamic.fr | [walshami@diginamic-formation.fr](mailto:walshami@diginamic-formation.fr) |
| David BILLON | Développeur | Diginamic.fr | [dbillon@diginamic-formation.fr](mailto:dbillon@diginamic-formation.fr) |
| Christophe GERMAIN | Directeur de projets / PO | Diginamic.fr | [cgermain@diginamic.fr](mailto:cgermain@diginamic.fr) |
| Robin HOTTON | PO | Diginamic.fr | [rhotton@diginamic-formation.fr](mailto:rhotton@diginamic-formation.fr) |
| Equipe DSI | Client | Digichees |  |

### Contraintes et exigences spécifiques

Le projet doit relever plusieurs défis et respecter plusieurs contraintes comme un budget défini pour optimiser l'utilisation des ressources et des exigences de sécurité strictes pour protéger les données de l'entreprise.

De plus, le nouvel intranet sous le nom de domaine digichees.com, doit être compatible avec les technologies actuelles, offrir une interface intuitive et flexible, ainsi qu'une architecture modulaire permettant une évolutivité future.

La conformité aux réglementations en vigueur, ainsi que la gestion efficace du changement pour assurer une adoption réussie du nouveau système, sont également des exigences à prendre en compte tout au long du projet.

### Lieux de réalisation du Projet

Le projet sera mené dans un environnement se déroulant à distance et lors de visites chez le client. Le télétravail sera encouragé pour les activités qui peuvent être réalisées de manière efficace à distance.

Des salles de réunion au bureau seront disponibles pour les réunions d'équipe, les séances de brainstorming et les présentations.

Puis, des visites chez le client seront planifiées selon les besoins du projet, permettant une interaction directe avec les parties prenantes.

## Description fonctionnelle des besoins

### Profils utilisateurs

**Il est possible d'accéder à la plateforme selon les rôles suivants :**

|  |  |  |
| --- | --- | --- |
| Rôles | Actions | Cumulable avec (optionnel) |
| Admin | Gère les users, communes, objets (articles), conditionnements, poids, poids vignette | ● Opérateur colis  ● Opérateur stock |
| Opérateur colis | Gestion de commande, sors le mailing, sort les stats, crée les clients, interface d’impression | ● Admin  ● Opérateur stock |
| Opérateur stock | Saisie de l’inventaire, gestion stock, interface d’impression | ● Admin  ● Opérateur colis |

### Scénarios d’utilisation

**Fonctionnalité [1] : Authentification**

Sous fonctionnalité : Redéfinir son mot de passe

Objectif : s’authentifier

Description : En tant que salarié possédant un identifiant, je peux m’authentifier en entrant mon nom et mon mot de passe, sur la page de connexion.

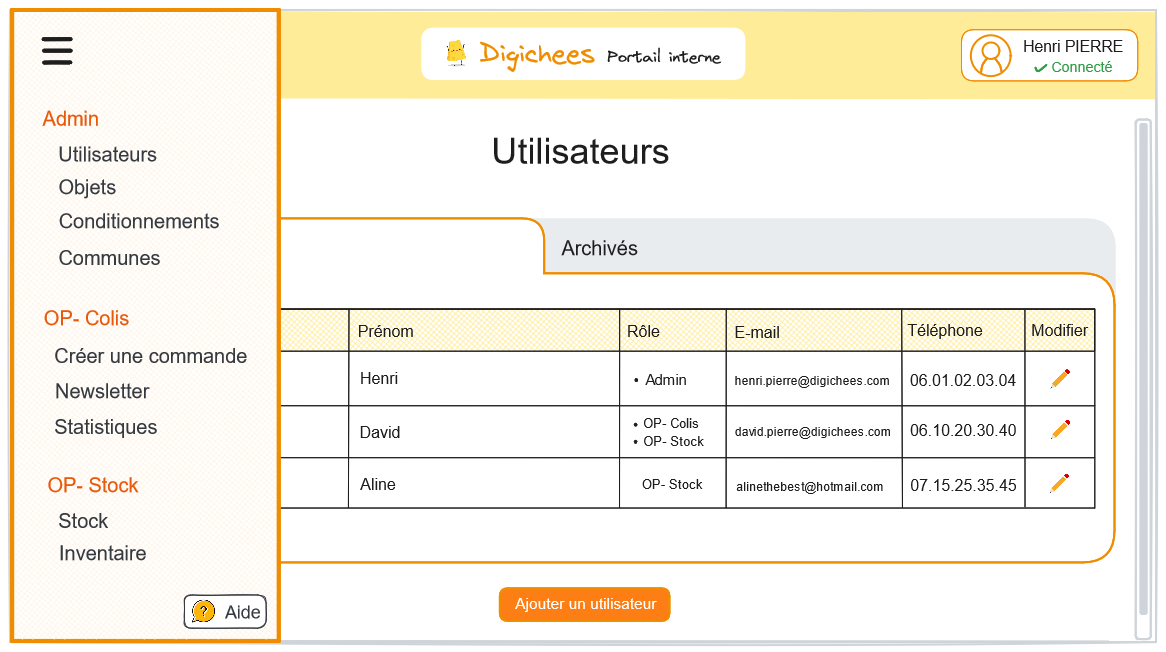
*Maquettes illustratives de la vue “Connexion”*

Contraintes et règles de gestion : le champ identifiant doit correspondre au Mail/Matricule.

Les deux champs doivent être sensible à la casse.

La connexion prend en compte le(s) rôle(s) de l’utilisateur : op-colis / op-stock /Administrateur

Niveau de priorité : Ultime

*Maquette illustrative : exemple de la vue d’un rôle Admin, OP-Colis et OP-Stock*

**Fonctionnalité [2] : Gérer les colis**

Sous fonctionnalité : gestion des clients/commandes/conditionnement,

Objectif :

Description :

En tant que Admin/op-colis je peux consulter la liste des commandes.

Je veux pouvoir traiter les commandes facilement, calculer les montants et suivre leur évolution pour assurer une bonne gestion des ventes.

Je suis informé si le client de la commande n’existe pas et je peux créer un client.

Je peux voir le montant final de la commande et les articles qui la compose.

Je veux pouvoir consulter les statistiques de nos ventes sur une période donnée pour mieux comprendre nos performances et identifier les tendances.

Je veux pouvoir choisir le conditionnement le plus adapté pour nos produits et consulter les options disponibles pour bien les protéger lors de l'expédition.

Je veux pouvoir connaître le poids moyen de nos colis pour mieux planifier les envois et éviter les problèmes de logistique.

J’ai la possibilité d’envoyer un email personnalisé.

Je veux pouvoir imprimer des documents au besoin pour disposer d'une copie papier de nos transactions ou de nos communications avec nos clients.

Contraintes et règles de gestion :

* **Traçabilité des Actions** : Toutes les actions effectuées dans le système doivent être enregistrées dans des journaux d'audit pour permettre la traçabilité et la vérification ultérieure.
* **Gestion des Autorisations** : Les utilisateurs doivent avoir des autorisations appropriées en fonction de leur rôle et de leurs responsabilités dans l'organisation.

Niveau de priorité : Haut

**Fonctionnalité [3] : Gérer les stocks**

Sous fonctionnalité : Aucune

Objectif : Gérer les stocks

En tant qu'OP-Stock, je peux consulter la liste des produits en stock pour voir ce que nous avons disponibles.

Je peux aussi actualiser ces informations en utilisant un bouton spécifique pour enregistrer les entrées et les sorties de produits.

Enfin, si besoin, je peux imprimer la liste des stocks sur papier pour l'avoir sous forme physique.

Contraintes et règles de gestion :

* **Traçabilité des Actions** : Toutes les actions effectuées dans le système doivent être enregistrées dans des journaux d'audit pour permettre la traçabilité et la vérification ultérieure.
* **Gestion des Autorisations** : Les utilisateurs doivent avoir des autorisations appropriées en fonction de leur rôle et de leurs responsabilités dans l'organisation.

Niveau de priorité : Haut

**Fonctionnalité [4] : Gérer les utilisateurs**

Sous fonctionnalité : Ajouter / modifier

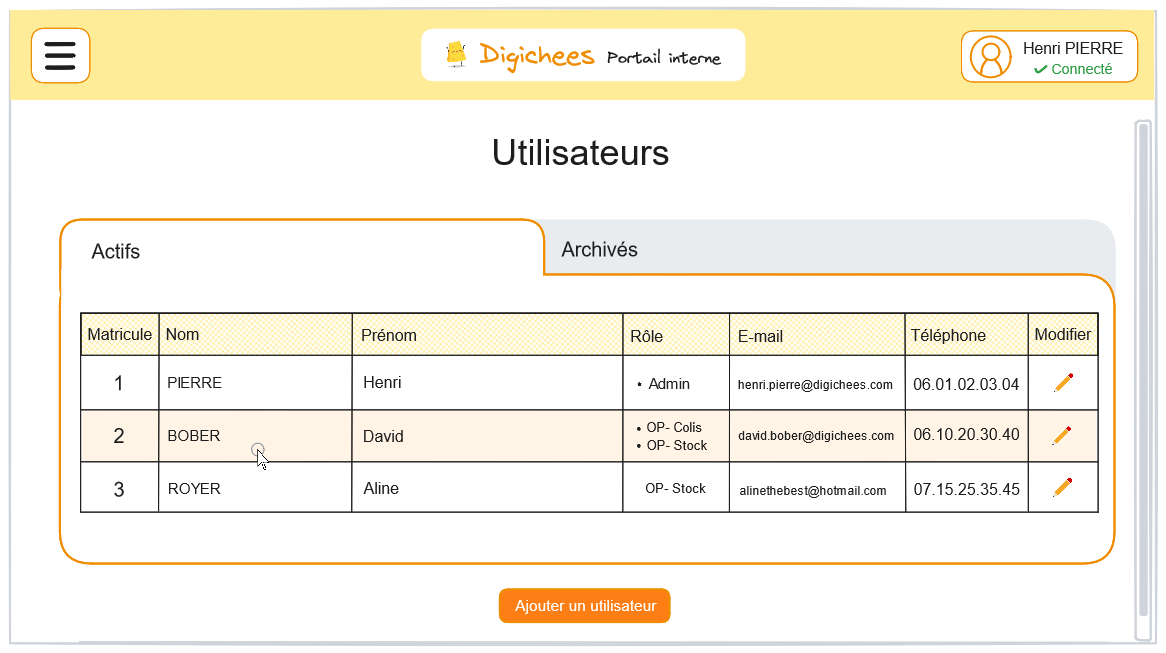
Objectif :

En tant qu'Administrateur je peux consulter la liste des utilisateurs de l’application.

Je peux accéder à la page d’édition d’un utilisateur en cliquant sur le bouton “Modifier”.

Je peux également accéder à la page de détail d’un utilisateur en cliquant sur la ligne idoine.

Je peux également ajouter un utilisateur en cliquant sur le bouton “Ajouter un utilisateur” qui me dirige vers le formulaire correspondant.

*Maquette illustrative : vue ”Gestion des utilisateurs”*

Contraintes et règles de gestion :

* **Traçabilité des Actions** : Toutes les actions effectuées dans le système doivent être enregistrées dans des journaux d'audit pour permettre la traçabilité et la vérification ultérieure.
* **Gestion des Autorisations** : Les utilisateurs doivent avoir des autorisations appropriées en fonction de leur rôle et de leurs responsabilités dans l'organisation.

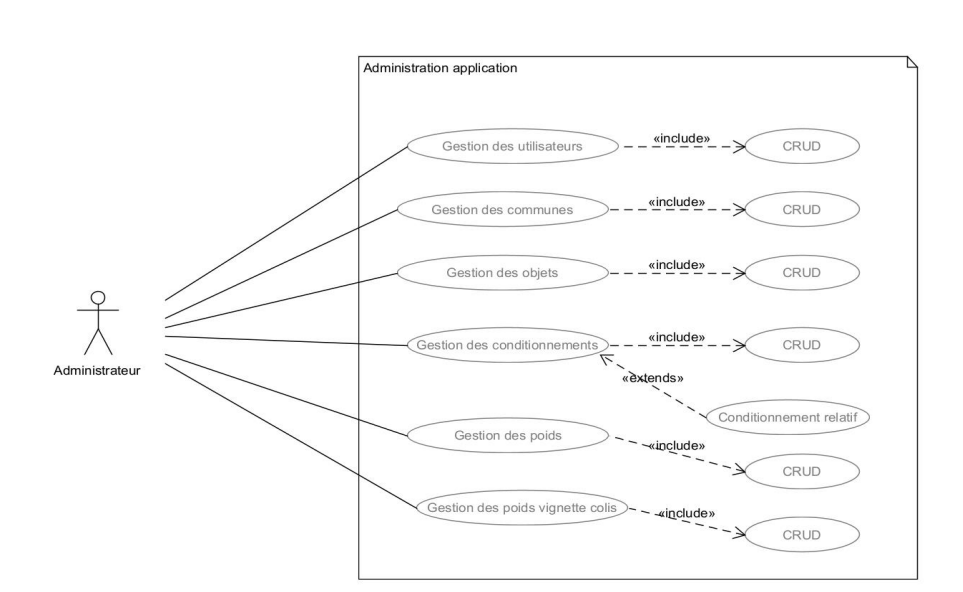
Niveau de priorité : Haut

### Fonctionnalités essentielles

Voici un aperçu des fonctionnalités essentielles pour chaque rôle dans le système de gestion :

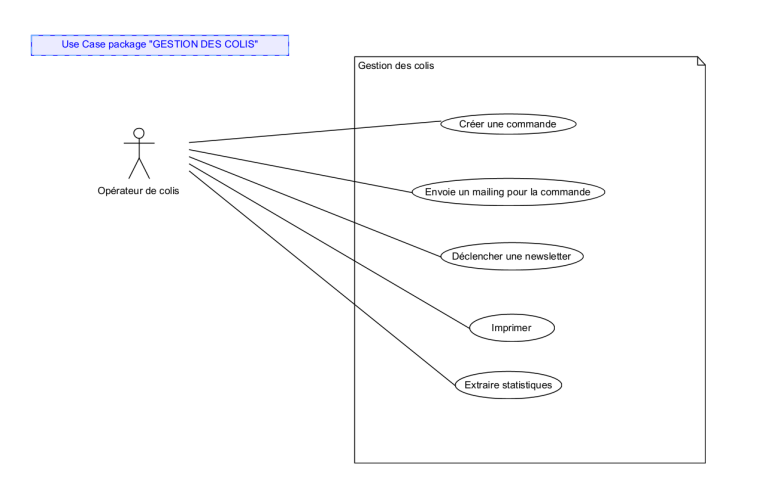
Pour le rôle administrateur :

* Gestion des utilisateurs et des paramètres métiers via des opérations CRUD.
* Possibilité de générer des impressions papier pour documenter les activités du système.



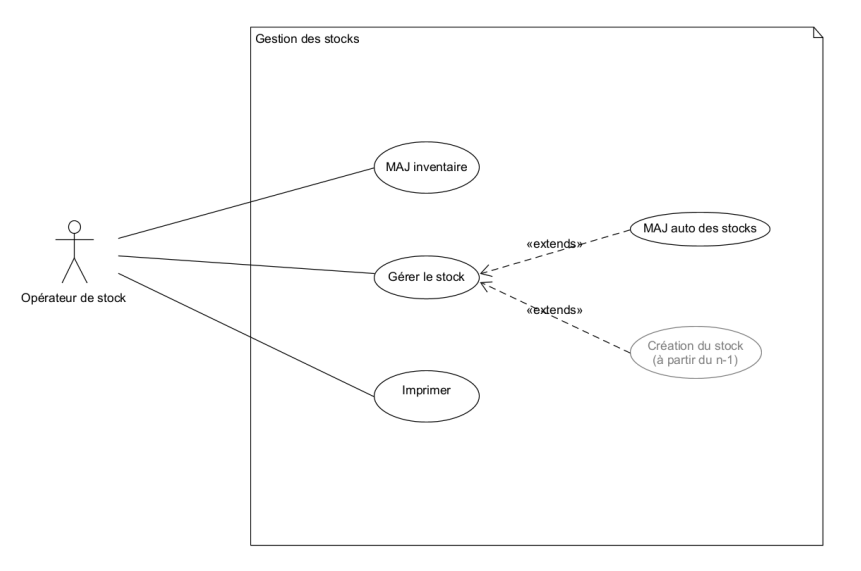
Pour le rôle OP-colis :

* Gestion des stocks pour une meilleure vue d'ensemble.
* Mise à jour annuelle des stocks facilitée par un bouton d'action dédié.
* Capacité à générer des impressions papier pour visualiser les stocks disponibles.



Pour le rôle OP-stock :

* Accès à un espace de gestion des stocks pour une vue globale.
* Possibilité de mettre à jour les stocks annuellement via un bouton d'action dédié.
* Capacité à générer des impressions papier pour documenter les stocks.



Ces fonctionnalités seront développées plus en détail dans la suite de la documentation pour fournir une compréhension approfondie de leur utilisation et de leur mise en œuvre.

## Exigences fonctionnelles

### Description détaillée des fonctionnalités du système

Pour le rôle administrateur :

1. Gestion des utilisateurs et des paramètres métiers :

* La fonctionnalité : Gestion des utilisateurs et des paramètres métiers via des opérations CRUD.
* Son objectif : Permettre à l'administrateur de contrôler les utilisateurs et de configurer les paramètres métiers du système.
* Son utilisation prévue : L'administrateur peut ajouter, modifier, supprimer et consulter les utilisateurs et les paramètres métiers selon les besoins.
* Les résultats attendus : Un contrôle complet sur les utilisateurs et les paramètres métiers du système, garantissant leur conformité avec les besoins de l'entreprise.

1. Génération d'impressions papier :

* La fonctionnalité : Possibilité de générer des impressions papier pour documenter les activités du système.
* Son objectif : Permettre à l'administrateur de créer des documents physiques pour archiver les activités et les données du système.
* Son utilisation prévue : L'administrateur peut générer des impressions papier à partir des rapports, des données utilisateur ou des paramètres métiers pour une documentation physique.
* Les résultats attendus : Disponibilité de documents physiques pour une documentation complète et une communication externe si nécessaire.

Pour le rôle OP-colis :

1. Gestion des commandes et historisation des mouvements :

* La fonctionnalité : Gestion des commandes et historisation des mouvements.
* Son objectif : Permettre à l'OP-colis de gérer les commandes et de suivre l'historique des mouvements.
* Son utilisation prévue : L'OP-colis peut consulter, ajouter, modifier et supprimer des commandes, ainsi que suivre l'historique des mouvements associés.
* Les résultats attendus : Une gestion efficace des commandes et un suivi précis des mouvements pour une meilleure traçabilité.

1. Visualisation de la liste client :

* La fonctionnalité : Visualisation de la liste client.
* Son objectif : Permettre à l'OP-colis de visualiser la liste des clients enregistrés dans le système.
* Son utilisation prévue : L'OP-colis peut consulter la liste des clients et accéder à leurs informations pertinentes.
* Les résultats attendus : Une vue claire et complète des clients enregistrés pour une meilleure gestion de la relation client.

1. Capacité à générer des impressions papier pour visualiser les stocks disponibles :

* La fonctionnalité : Capacité à générer des impressions papier pour visualiser les stocks disponibles.
* Son objectif : Offrir une option pour créer des documents physiques pour référence et communication sur les stocks.
* Son utilisation prévue : L'OP-colis peut générer des impressions papier pour obtenir une vue globale et détaillée des stocks disponibles.
* Les résultats attendus : Disponibilité de documents physiques pour une référence rapide et une communication externe si nécessaire.

1. Génération et envoie de mail personnalisé au client :

* La fonctionnalité : Génération et envoi de mail personnalisé au client.
* Son objectif : Permettre à l'OP-colis de communiquer de manière personnalisée avec les clients.
* Son utilisation prévue : L'OP-colis peut générer et envoyer des e-mails personnalisés aux clients pour des notifications, des confirmations de commande, etc.
* Les résultats attendus : Une communication efficace et personnalisée avec les clients pour améliorer l'expérience utilisateur et la satisfaction.

Pour le rôle OP-stock :

1. Accès à un espace de gestion des stocks pour une vue globale :

* La fonctionnalité : Accès à un espace de gestion des stocks pour une vue globale.
* Son objectif : Permettre à l'OP-stock d'avoir une vision d'ensemble sur les niveaux de stocks disponibles.
* Son utilisation prévue : L'OP-stock peut consulter les stocks actuels, vérifier les niveaux de disponibilité et surveiller les variations.
* Les résultats attendus : Une meilleure gestion des stocks grâce à une vue globale et en temps réel des stocks disponibles.

1. Possibilité de mettre à jour les stocks annuellement via un bouton d'action dédié :

* La fonctionnalité : Mise à jour annuelle des stocks facilitée par un bouton d'action dédié.
* Son objectif : Simplifier le processus de mise à jour des stocks à des fins de gestion et de reporting.
* Son utilisation prévue : L'OP-stock peut utiliser le bouton d'action dédié pour mettre à jour les niveaux de stocks à la fin de chaque année.
* Les résultats attendus : Des niveaux de stocks précis et à jour, permettant une planification efficace des activités futures.

1. Capacité à générer des impressions papier pour documenter les stocks :

* La fonctionnalité : Capacité à générer des impressions papier pour documenter les stocks.
* Son objectif : Offrir une option pour créer des documents physiques pour référence et communication sur les stocks.
* Son utilisation prévue : L'OP-stock peut générer des impressions papier pour obtenir une vue globale et détaillée des stocks disponibles.
* Les résultats attendus : Disponibilité de documents physiques pour une référence rapide et une communication externe si nécessaire.

### Cas d'utilisation

Voici un cas d’utilisation pour une fonctionnalité de chaque rôle :

Pour le rôle administrateur,

Scénario où l'administrateur doit ajouter un nouvel utilisateur au système :

1. L'administrateur se connecte à l'interface d'administration du système.
2. Il navigue jusqu'à la section de gestion des utilisateurs.
3. Il sélectionne l'option d'ajout d'un nouvel utilisateur.
4. Le système affiche un formulaire de saisie pour les informations de l'utilisateur.
5. L'administrateur remplit les champs requis tels que le nom, l'adresse e-mail et le rôle de l'utilisateur.
6. Une fois les informations saisies, l'administrateur valide le formulaire.
7. Le système vérifie les données saisies et les enregistre dans la base de données.
8. Une confirmation de réussite est affichée à l'administrateur, indiquant que le nouvel utilisateur a été ajouté avec succès.

Pour le rôle OP-colis,

Scénario où l'OP-colis doit générer et envoyer un e-mail personnalisé à un client pour confirmer sa commande :

1. L'OP-colis se connecte à l'interface de gestion des commandes.
2. Il sélectionne la commande pour laquelle il souhaite envoyer une confirmation au client.
3. Il choisit l'option d'envoi d'e-mail personnalisé dans le menu des actions disponibles.
4. Le système affiche un formulaire de composition d'e-mail prérempli avec les détails de la commande et les informations du client.
5. L'OP-colis vérifie et personnalise le contenu de l'e-mail au besoin.
6. Une fois satisfait du contenu, il envoie l'e-mail.
7. Le système traite la demande d'envoi d'e-mail et l'achemine au serveur de messagerie.
8. Une notification de confirmation est affichée à l'OP-colis, confirmant que l'e-mail a été envoyé avec succès au client.

Pour le rôle OP-stock,

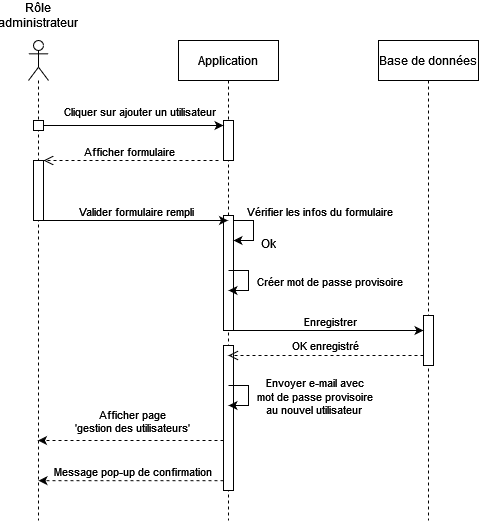
Scénario où l'OP-stock doit mettre à jour les niveaux de stock annuellement :

1. L'OP-stock accède à l'interface de gestion des stocks.
2. Il sélectionne l'option de mise à jour annuelle des stocks.
3. Le système affiche un formulaire pour saisir les nouvelles valeurs des stocks.
4. L'OP-stock saisit les nouvelles quantités de stock pour chaque produit.
5. Une fois les informations saisies, il valide le formulaire.
6. Le système vérifie les données et met à jour les niveaux de stock dans la base de données.
7. Une notification de confirmation est affichée à l'OP-stock, confirmant que les stocks ont été mis à jour avec succès pour l'année en cours.

Ces scénarios illustrent comment les utilisateurs interagissent avec le système pour accomplir des tâches spécifiques, telles que l'administration des utilisateurs, la gestion des commandes et la mise à jour des stocks. Chaque étape du processus est conçue pour être intuitive et efficace, garantissant une expérience utilisateur optimale.

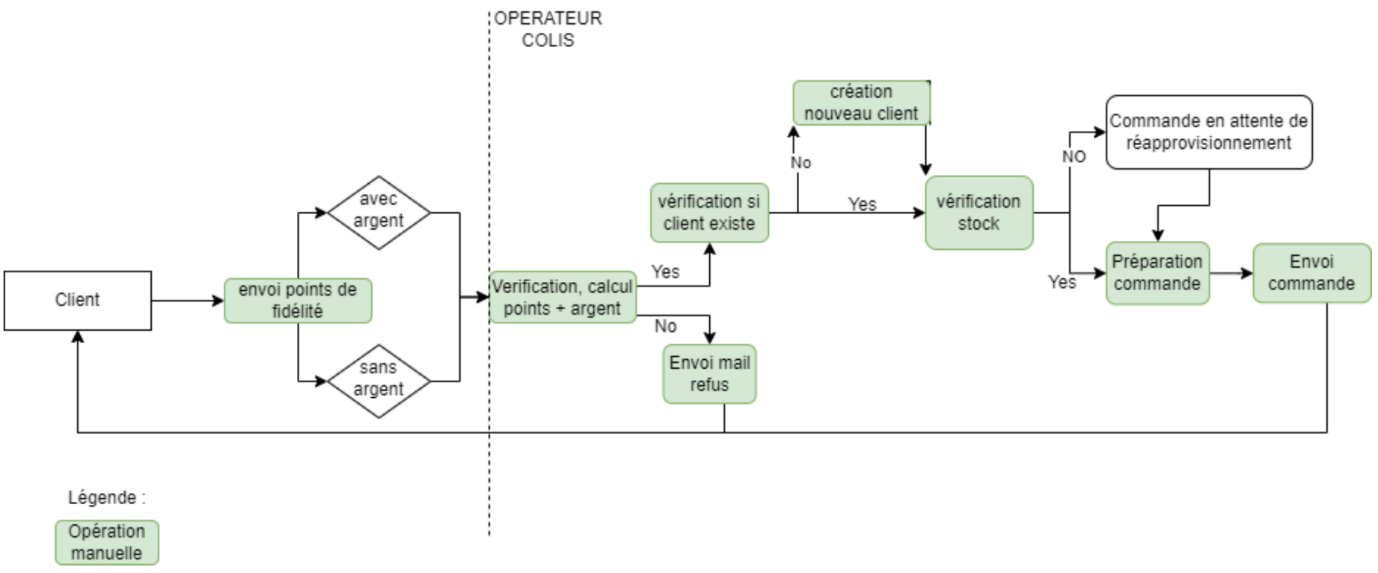
### Diagramme de séquence

Ajout d’un nouvel utilisateur :



### Diagramme de flux

Commande d’un client :



## Exigences non-fonctionnelles

### Performance du système

**Temps de réponse rapide** : Les utilisateurs s'attendent à ce que l'application réponde rapidement à leurs actions.

**Fiabilité** : Le système doit être fiable et disponible en permanence durant les heures d’activité de l’entreprise pour éviter les interruptions dans les processus opérationnels.

**Optimisation des ressources** : Le système doit utiliser efficacement les ressources matérielles disponibles, telles que la mémoire, le processeur et le stockage, pour garantir des performances optimales sans gaspillage de ressources.

### Sécurité des données

Dans le cadre de ce cahier des charges, un accent particulier sera mis sur la protection et la confidentialité des données sensibles. Cela implique la mise en place de mesures de sécurité robustes telles que le cryptage des données, la gestion des accès basée sur des rôles, l'authentification à deux facteurs et la surveillance continue des activités système pour détecter toute activité suspecte.

De plus, des protocoles de sauvegarde réguliers seront établis pour garantir la disponibilité des données en cas de sinistre ou de défaillance du système. En alignement avec les réglementations en vigueur telles que le RGPD, une attention particulière sera accordée à la protection de la vie privée des utilisateurs et à la conformité aux normes de sécurité de l'industrie.

### Maintenance et évolutivité

**Facilité de maintenance et de mise à jour** : La capacité à effectuer des mises à jour logicielles et à maintenir le système sans perturber les opérations est également importante pour assurer des performances continues.

**Évolutivité** : L'application doit être capable de gérer efficacement un grand nombre de commandes, de colis et d'utilisateurs à mesure que l'entreprise grandit. Cela signifie que le système doit être extensible et capable de s'adapter à une charge de travail accrue sans compromettre les performances.

### Indicateurs de suivi

Le projet inclura un ensemble d'Indicateurs de Performance Clés (KPI : Key Performance Indicators) sélectionnés pour évaluer la progression, la qualité et le succès global du projet. Ces KPI seront utilisés pour surveiller et évaluer la conformité aux délais, au budget et à la portée définis, ainsi que pour mesurer la satisfaction des parties prenantes et de l'équipe de projet. De plus, les KPI permettront d'identifier les risques, de suivre la productivité de l'équipe et de mesurer la qualité des livrables.

**Conformité aux délais :**

Pourcentage de tâches terminées dans les délais par rapport au total des tâches planifiées.

Nombre de jours de retard accumulés sur l'échéancier global du projet.

**Conformité au budget :**

Écart entre le coût réel du projet et le budget prévu, exprimé en pourcentage.

Pourcentage de dépenses par rapport au budget total alloué au projet à une étape spécifique.

**Portée du projet :**

Pourcentage de fonctionnalités ou de livrables achevés.

Nombre de changements apportés par rapport au plan initial.

**Qualité des livrables :**

Nombre de bugs déposés par le client ou nombre d'anomalies identifiés.

**Satisfaction des parties prenantes :**

Score de satisfaction des parties prenantes obtenu à partir de sondages ou d'évaluations régulières.

Par rapport à la qualité des livrables

**Productivité de l'équipe :**

Nombre de tâches traitées par l'équipe en fonction d’une temporalité à définir avec le client.

**Risque du projet :**

Nombre de risques résolus ou atténués par rapport au nombre total de risques identifiés.

**Taux de changement :**

Nombre de demandes de changement approuvées par rapport au nombre total de demandes reçues.

Impact financier des changements apportés à la portée, au calendrier ou au budget du projet.

**Statut du projet :**

Évaluation périodique de l’avancée du projet par rapport à un ensemble prédéfini de critères de réussite.

**Satisfaction de l'équipe :**

Score de satisfaction de l'équipe.

## Architecture technique

### Environnement de développement (DEV)

Objectif : Cet environnement est dédié à la phase de développement du projet. Il sert de bac à sable pour les développeurs afin d'écrire et de tester du code sans affecter l'environnement de production.

Configuration :

* Spécifications matérielles :
  + Serveur de Développement (prix total : 1342 €) :
    - Processeur : Intel Xeon E5-2620 2GHz
    - Mémoire (RAM) : Kingston HyperX Fury 16 Go DDR4 2666 MHz ECC
    - Stockage : Crucial MX500 1 To SSD NVMe
    - Système d'Exploitation : Ubuntu Server 20.04 LTS
    - Réseau : TP-Link TG-3468 Gigabit PCIe Network Adapter
    - Châssis de Serveur : Dell PowerEdge T340 Tower ServerOutils
* Logiciels utilisés :
  + Git
  + Pip
  + Apache Maven
  + Docker
  + MySQL

1. **Environnement de test (UAT)**

Objectif : L'UAT vise à garantir que le logiciel est prêt pour une utilisation en production en permettant aux utilisateurs finaux de valider ses fonctionnalités, sa convivialité et sa conformité aux exigences avant son déploiement final.

Configuration :

* Spécifications matérielles :
  + Serveur de Développement (prix total : Environ 1250€ à 1500 €) :
    - Processeur : Intel Core i7-11700K (8 cœurs, 16 threads, fréquence de base de 3,6 GHz, jusqu'à 5,0 GHz avec Turbo Boost)
    - Mémoire (RAM) : 32 Go de RAM DDR4 (2x16 Go, 3200 MHz)
    - Stockage : SSD NVMe Crucial P5 500 Go
    - Système d'Exploitation : Ubuntu Server 20.04 LTS
    - Réseau : Adaptateur Ethernet Gigabit intégré à la carte mère
    - Châssis de Serveur : Boîtier moyen tour Corsair Carbide Series 275R
    - Alimentation : Alimentation EVGA 600 W1, 80+ WHITE 600W
* Logiciels utilisés :
  + Git
  + Pip
  + Python runtime
  + MySQL
  + Apache HTTP Server

### Environnement de production (PROD)

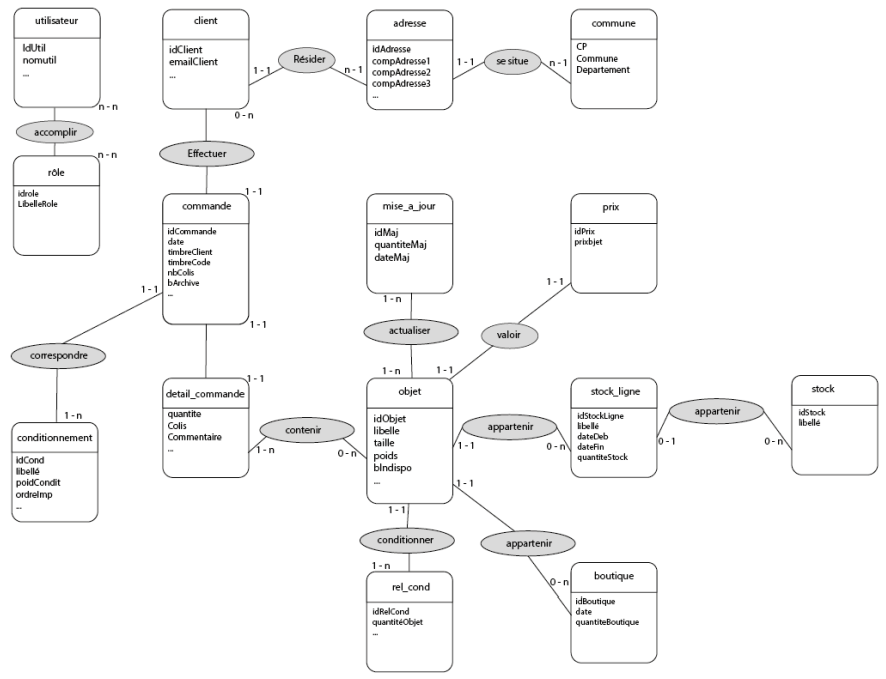
Objectif : Cet environnement est dédié à l’application déployée qui sera utilisable par les utilisateurs finaux (les salariés de Digichees). La configuration se veut plus performante pour coller aux exigences du client.

Configuration :

* Spécifications matérielles :
  + Serveur de production (prix total : Environ 2750€ à 3000€) :
    - Processeur : Intel Xeon E-2288G (8 cœurs, 16 threads, fréquence de base de 3,70 GHz, fréquence maximale de 5,00 GHz avec Turbo Boost)
    - Mémoire (RAM) : 64 Go DDR4 ECC (4x16 Go, 2666 MHz)
    - Stockage : SSD NVMe Samsung 970 EVO Plus 1 To
    - Système d'Exploitation : Ubuntu Server 20.04 LTS
    - Réseau : Adaptateur Ethernet Gigabit Intel intégré à la carte mère
    - Châssis de Serveur : Supermicro SuperServer SYS-5019S-L (format tour)
    - Alimentations Redondantes : Deux alimentations redondantes échangeables à chaud
    - Répartiteur de Charge : Répartiteur de charge matériel (par exemple, Kemp LoadMaster LM-2400)
* Logiciels utilisés :
  + MySQL
  + Python runtime
  + Apache HTTP Server
  + Git
  + Pip
  + Docker

### MPD et couche de persistance

Modèle physique des données :



## Planning et Livrables

### Phases du projet

* **Préconception / Conception**
  + Définir les objectifs et recueillir les exigences
  + Documentations légales (les cahiers des charges, Note de cadrage)
  + Considérations légales et de conformité et évaluation des risques
  + Définir le plan de projet
  + Créer des personas d'utilisateurs
  + Recherche et analyse (Sélection des technologies, préconception de l'architecture)
  + Storyboards et wireframes (maquettes)
  + Conception de l'architecture
  + Développement de prototypes
  + Documentation d’équipe (Wiki de projet, les readme)
* **Développement**
  + Configuration et mise en place des Environnements (développement, test, production)
  + Élaboration du Code
  + Intégration Continue
  + Tests Unitaires et Tests d'Intégration
  + Débogage et Correction d'Erreurs
  + Optimisation des Performances
  + Documentation
  + Révisions de Code
  + Collaboration
* **Testing**
  + Conception des Cas de Test
  + Tests d'Acceptation Utilisateur (UAT)
  + Tests de Performance
  + Tests de Sécurité
* **Déploiement**
  + Planification du déploiement
  + Livraison de versions
  + Exécution du déploiement
  + Gestion des configurations et des déploiements
  + Surveillance et maintenance
  + Tests après déploiement

### Calendrier

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Semaine 1** | **Semaine 2** | **Semaine 3** | **Semaine 4** | **Semaine 5** | **Semaine 6** |
| **Conception** | **Développement** | **Développement** | **Développement** | **Testing** | **Déploiement** |
| *Planning* | *Architecture* | *Backend* | *Front* | *Test client* | *Mise En Place* |
| *Documentation* | *BDD* | *Backend* | *TU* | *Test client / Debug* | *Mise En Place* |
| *Maquettage* | *Backend* | *Front* | *TU* | *Debug* | *Mise En Place* |

### Livrables attendus à chaque étape

Étapes de :

* **Préconception / Conception**
  + Document de vision du projet
  + Spécifications fonctionnelles
  + Wireframes ou maquettes
  + Document d'architecture
  + Plan de projet
* **Développement**
  + Code source
  + Documentation du code
  + Rapports de progression
  + Tests unitaires
* **Testing**
  + Plan de test
  + Cas de test
  + Rapports de test
  + Matrice de traçabilité des tests
* **Déploiement**
  + Package d'application
  + Plan de déploiement
  + Rapport de déploiement
  + Guide de maintenance
  + Notices pour les utilisateurs

### Fréquence et durée des comités de pilotage

Les comités de pilotage seront organisés régulièrement tout au long du projet. La fréquence des réunions sera déterminée en fonction des besoins spécifiques du projet, avec des réunions planifiées à des intervalles réguliers, telles que hebdomadaires, bihebdomadaires ou mensuelles.

La durée des réunions sera optimisée pour permettre une discussion approfondie des sujets à l'ordre du jour tout en respectant la disponibilité et l'attention des membres du comité. Les réunions seront préparées en amont avec des ordres du jour clairs et des documents pertinents partagés à l'avance pour maximiser l'efficacité du temps passé ensemble.

## Ressources

### Estimation des coûts

Le projet dispose d’un budget de 50 000€.

|  |  |
| --- | --- |
| **Coût** | **Gains** |
| Analyse des besoins initiaux -  3 000€ | Meilleure compréhension des exigences du projet |
| Rédaction du cahier des charges - 5 000€ | Clarification des attentes et objectifs du projet |
| Conception et développement -  20 000€ | Production d'un produit ou service conforme aux spécifications |
| Tests et validation –  7 000€ | Réduction des erreurs et des bugs |
| Formation du personnel - 4 000€ | Utilisation efficace du produit final |
| Support technique initial - 3 000€ | Résolution rapide des problèmes post-lancement |
| Réserve pour les coûts imprévus - 8 000€ | Flexibilité pour répondre aux changements ou aux défis imprévus |

### Besoins en ressources humaines

* Chef de Projet : Dirige l'équipe, organise les tâches et veille au bon déroulement du projet.
* Product Owner : Chargé de maximiser la valeur du produit en fonction des besoins et des attentes des utilisateurs finaux.
* Développeurs : Écrivent le code et construisent le logiciel.
* Testeurs : Trouvent et corrigent les problèmes dans le logiciel.
* Designers : Rendent le logiciel attrayant et facile à utiliser.
* Administrateurs Système : Assurent le bon fonctionnement technique.
* Analystes Métier : Comprennent les besoins du logiciel et les raisons derrière.
* Rédacteurs Techniques : Créent des guides pour aider les utilisateurs.
* Experts en Sécurité : Protègent le logiciel contre les hackers.
* Support Client : Aident les utilisateurs en cas de questions ou de problèmes.

### Matériels et logiciels nécessaires

#### Matériel :

* **Ordinateurs** : Chaque membre de l'équipe a besoin d'un ordinateur ou d'un portable pour coder, concevoir, tester et autres tâches.
* **Fournitures de bureau** : Stylos, carnets, tableaux blancs, post-it, etc., pour les sessions de brainstorming, la planification et la prise de notes lors des réunions.
* **Documentation** : Impressions ou copies numériques des exigences du projet, des histoires utilisateur, des documents de conception, etc., pour référence.

#### Logiciels :

* **Environnement de Développement Intégré (IDE)** : Des logiciels comme Visual Studio Code ou IntelliJ IDEA pour écrire et éditer le code
* **Système de Contrôle de Version** : Git et GitHub pour gérer et suivre les changements apportés à la base de code.
* **Outils de Gestion de Projet** : Jira pour attribuer du travail et suivre les progrès.
* **Outils de Communicatio**n : Microsoft Teams ou Discord pour la communication d'équipe, la collaboration et les réunions.
* **Outils de Conception** : Excalidraw ou Figma pour la conception des interfaces utilisateur et la création d'éléments visuels. Ainsi que les applications pour créer des diagrammes comme Draw.io.
* **Outils de Test** : Des frameworks comme Selenium, Jest ou JUnit pour les tests automatisés, et des outils comme Postman pour les tests d'API.
* **Système de Gestion de Base de Données** : Des plateformes comme MySQL, PostgreSQL ou MongoDB pour la gestion et l'interrogation des données.
* **Outils de Déploiement** : Des services comme Docker, Kubernetes ou AWS Elastic Beanstalk pour le déploiement et la gestion des applications dans les environnements de production.

## Risques / Contraintes / Dépendances

**Dépassements de budget :** La mauvaise estimation des coûts, des dépenses imprévues ou des changements de périmètre du projet peut entraîner un dépassement du budget de 50.000€ qui est alloué au projet.

➜ Solution : Réserve pour les coûts imprévus (environ 4 000€), suivi minutieux des dépenses.

**Retards de calendrier :** Des événements imprévus, tels que l'indisponibilité des ressources, des problèmes techniques ou des changements dans les exigences, peuvent entraîner des retards dans les délais du projet.

➜ Solutions : Planification, estimation le plus réaliste possible, suivi régulier, communication efficace, flexibilité.

**Défis techniques :** Des exigences techniques complexes ou des technologies non familières peuvent poser des défis lors de la mise en œuvre du projet, entraînant des retards ou des problèmes de qualité.

➜ Solution : Investir dans la recherche et la formation, recruter des experts spécialisés, prototyper et tester, favoriser la collaboration entre pair, envisager l’externalisation si nécessaire, adopter une planification flexible et maintenir une communication claire entre toutes les parties prenantes.

**Ruptures de communication :** Une communication inefficace au sein de l'équipe du projet ou avec les parties prenantes peut entraîner des malentendus, des retards ou des erreurs dans l'exécution du projet.

➜ Solution : Utiliser des outils de gestion de projet collaboratifs pour suivre les progrès, partager des documents et faciliter la communication.

Encourager un environnement où les membres de l'équipe se sentent à l'aise pour poser des questions et exprimer leurs préoccupations.

Mettre en place des réunions régulières pour assurer une communication transparente et ouverte au sein de l'équipe.

**Risques d'intégration :** Des problèmes de compatibilité ou des difficultés d'intégration de nouveaux systèmes ou processus avec l'infrastructure existante peuvent entraîner des perturbations ou des inefficacités.

➜ Solution : Réalisation d'une évaluation approfondie de la compatibilité, l'utilisation de normes ou de protocoles d'interopérabilité reconnus, la mise en place de tests d'intégration rigoureux, l'engagement de fournisseurs ou de partenaires expérimentés dans l'intégration de systèmes, l'adoption d'une approche incrémentielle pour l'intégration afin de minimiser les perturbations, ainsi que la communication transparente et la collaboration entre les équipes impliquées dans le processus d'intégration .