

REFONTE SI DE LA GESTION DES CADEAUX FIDÉLITÉ

FROMAGERIE DIGICHEESE

OBJECTIF DU PROJET

[15/01/2025]

CONTEXTE DU PROJET

1. Historique et origine du projet



La Fromagerie DIGICHEESE, une entreprise familiale centenaire, utilise une application développée il y a plus de 20 ans avec Access Office 2000. Cette application est devenue obsolète, instable et difficile à maintenir. Elle ne répond plus aux besoins croissants de l'entreprise en matière de gestion de la fidélité et de traitement des commandes.

2. Situation actuelle



La situation actuelle de l'entreprise repose sur une application de gestion interne qui utilise Microsoft Access, avec une implémentation en VBA, pour répondre aux besoins de gestion des données et des processus métiers. Cependant, cette solution, bien que fonctionnelle à court terme, présente plusieurs limites notables. D'une part, l'application souffre d'une instabilité fréquente avec des bugs réguliers qui nuisent à la productivité des utilisateurs. D'autre part, sa maintenance est complexe et chronophage, et la possibilité d'ajouter de nouvelles fonctionnalités ou d'adapter l'application à l'évolution des besoins de l'entreprise est limitée. En outre, le manque de fluidité et d'accessibilité rend l'outil difficile à utiliser pour les collaborateurs, ce qui impacte la performance et la réactivité des équipes. Ces problèmes sont exacerbés par la taille du fichier, qui atteint entre 50 et 60 Mo, ce qui engendre une gestion inefficace des ressources système.

Dans un contexte économique où la digitalisation des processus métiers est cruciale pour gagner en efficacité, cette application vieillissante ne permet plus de répondre aux exigences modernes d'agilité et de performance. Par ailleurs, dans un environnement social où les attentes des utilisateurs en matière de simplicité et d'accessibilité des outils sont élevées, l'application actuelle ne répond plus aux standards de qualité et d'ergonomie attendus. Cette situation nécessite donc une refonte complète pour répondre aux enjeux de stabilité, de maintenance simplifiée, de flexibilité et d'optimisation des performances.

3. Problématique(s) identifiée(s)



- *Application obsolète qui ne convient pas aux besoins client.*
- *Maintenance : difficile à maintenir et à mettre à jour.*
- *Instabilité : des bugs fréquents perturbent son fonctionnement.*
- *Gestion des stocks nécessitant une simplification et amélioration des performances.*
- *Évolutivité : peu de possibilités d'ajout de nouvelles fonctionnalités.*
- *Accessibilité : manque de fluidité et de visibilité pour les utilisateurs.*
- *Base de données: potentiel corruption des données*

ENJEUX & OBJECTIFS DU PROJET

4. Enjeux du projet



- *Réduction des erreurs humaines grâce à l'automatisation.*
- *Amélioration de l'expérience utilisateur avec une interface moderne et intuitive.*
- *Augmentation de la performance et de la fiabilité du système.*
- *Mise en place d'une infrastructure technologique durable.*

5. Exigences principales



- *Qualité : Développement d'une solution performante et conforme aux attentes.*
- *Délai : Finalisation du projet dans les délais fixés (voir chronologie).*
- *Coûts : Respect du budget global de 39 460 €.*
- *Stratégie : Alignement avec les objectifs à long terme de l'entreprise.*

6. Présentation de la cible



La cible est un environnement applicatif en ligne permettant une gestion fluide et performante des programmes de fidélité et des processus associés (gestion des stocks, commandes, communication client, extraction de statistiques).

***DIGICHEESE** est une entreprise familiale centenaire spécialisée dans la production et la distribution de produits laitiers, principalement des fromages. Dans le contexte du projet mentionné, **DIGICHEESE** gère un **programme de fidélité** destiné à ses clients particuliers et distributeurs. Ce programme repose sur un système où les clients accumulent des points qu'ils peuvent échanger contre des cadeaux (ou goodies).*

Cependant, leur infrastructure technologique actuelle, basée sur une application développée il y a plus de 20 ans avec Access Office 2000, est devenue obsolète. L'objectif principal du projet est donc de moderniser leur système pour répondre aux besoins croissants de l'entreprise et améliorer la gestion de la fidélité des clients.

Le projet vise également à améliorer les processus internes, tels que la gestion des commandes, des stocks, et la communication avec les clients.

PÉRIMÈTRE DU PROJET

7. Périmètre et limites du projet



Inclut :

- Gestion des utilisateurs, commandes, stocks, statistiques.
- Développement des API pour les fonctionnalités
- Gestion de la base de données
 - CRUD Utilisateurs
 - CRUD Communes
 - CRUD Objets
 - CRUD Conditionnements
- Création de commandes
- Communication Client
 - Mailing Client
 - Envoi de Newsletter
- Gestion des stocks
 - Mise à jour automatique des stocks
 - Mise à jour de l'inventaire
 - Imprimer rapports stocks
- Extraction de statistiques
- Mise en place d'une architecture moderne (FastAPI, SQLAlchemy, MariaDB)

Exclut :

- Fonctionnalité à mettre à jour

8. Cartographie des acteurs (parties prenantes et affectées)



- **Chef de projet :** Christophe Germain
- **Développeurs :** Olivier Koenig, Alexandre Sacks, Maxime Godineau, Sylvie Kelonkeu
- **Utilisateurs finaux :** Employés de Digicheese: Administrateurs, opérateurs colis, opérateurs stocks.
- **Parties impactées :** Clients de DIGICHEESE bénéficiant du programme de fidélité.

9. Livrables projet



- Nouvelle application web (front-end et back-end).
- Documentation utilisateur et technique.
- Rapports d'avancement et statistiques personnalisés.
- Maquettes d'interface utilisateur.

10. Moyens mis à disposition



Moyens humains :

- **1 Chef de projet :** Responsable de la coordination globale, de la planification et du suivi des jalons clés.
- **4 Développeurs concepteurs :**
 - 2 spécialisés en Front-end pour développer l'interface utilisateur (React.js).
 - 2 spécialisés en Back-end pour gérer les API REST et l'intégration des bases de données (Node.js, Sequelize).

Moyens techniques :

- **Serveurs physiques et virtuels :**
 - **Serveur Dell PowerEdge T550**
 - Processeurs : 3.0Ghz 8 Coeurs 16 Threads
 - RAM : 128 Go .
 - Stockage : 1 To SSD + 10 To HDD pour les sauvegardes locales.
 - OS : Proxmox
 - Alimentation redondante pour garantir la disponibilité.
 - **Machines virtuelles :**
 - Déployées sur VMware ESXi pour isoler les environnements de développement, test, et production.

- **Outils logiciels :**
 - **Développement :**
 - **FastAPI** : Framework pour la création des APIs REST rapides et sécurisées.
 - **SQLAlchemy** : ORM pour simplifier l'interaction avec la base de données.
 - **Python 3.10** : Langage principal pour les composants back-end.
 - **Uvicorn** : Serveur ASGI pour exécuter les applications FastAPI.
 - **MariaDB** : Base de données
 - **Tests :**
 - **Pytest** : Framework pour les tests unitaires et d'intégration.
 - **Swagger** : Outil pour tester les endpoints API.
 - **Gestion des versions :**
 - **Git** avec intégration GitHub pour le contrôle des versions et la collaboration.
- **Sécurisation des données :**
 - Bases de données protégées par chiffrement AES-256.
- **Outils de gestion de projet :**
 - **Jira** : Suivi des tâches et gestion Agile des sprints.

Moyens financiers :

- **Budget global estimé : 39 460 €.**
- **La répartition détaillée des coûts est visible dans le Tableau des coûts.**

Tableau des coûts :

- **Salaires :**
 Calcul : $4 \times (28 \text{ jours} \times 350 \text{ €/jour}) + 5 \text{ jours} \times 700 \text{ €/jour} = 32900 \text{ €}$
 Raison : Salaires de l'équipe projet. (voir détails)
- **Maintenance logicielle 2 ans :**
 Montant : 500 €
 Raison : Permet la maintenance du code sans frais supplémentaires.
- **Maintenance corrective infrastructure 2 ans :**
 Montant : 500 €
 Raison : Permet la résolution des incidents d'infrastructure sans frais supplémentaires.
- **Serveur Dell PowerEdge T550 :**
 Montant : 2 000 €
 Raison : Infrastructure.
- **Licence Proxmox VE Standard 2 ans :**
 Montant : 1 060 €
 Raison : Outils et licences.

- **Débordement de budget:**

Nous avons prévu une marge de 15% sur le budget du projet qui équivaut à 5919 €.
Le budget Maximal du projet est donc de 45379 €

Equipes :

- **Analyse des besoins :**

Durée : 5 jours/homme
Responsable : Chef de projet (Christophe)
Coût : 700 €/jour HT

- **Développement Front-end :**

Durée : 56 jours/homme
Responsable : Développeur Full-stack junior
Coût : 350 €/jour HT

- **Intégration API :**

Durée : 28 jours/homme
Responsable : Développeur Full-stack junior
Coût : 350 €/jour HT

- **Tests et validation :**

Durée : 28 jours/homme
Responsable : Développeur Full-stack junior
Coût : 350 €/jour HT

RISQUES, CONTRAINTES ET DÉPENDANCES

11. Risques identifiés



Inadéquation avec les besoins du client

Description: Si les exigences des utilisateurs finaux ne sont pas bien comprises ou sont mal définies dès le départ, cela pourrait entraîner des retards, des révisions coûteuses, et un produit final non conforme aux attentes.

Niveau d'impact : Élevé

Niveau de survenance : Moyen

Criticité: 6

Solution de remédiation:

- Impliquer les utilisateurs finaux dès les premières phases du projet pour co-construire les nouvelles fonctionnalités et l'interface.
- Recueillir des retours réguliers grâce à des tests utilisateurs et des phases de validation intermédiaires.
- Introduire progressivement la nouvelle application pour permettre une adaptation en douceur.
- Mettre en place un mécanisme de collecte de feedback post-déploiement.
- Prévoir des mises à jour rapides pour corriger les éventuelles incohérences ou lacunes identifiées.

Échec de la migration:

Description: La migration de l'ancienne application vers le nouveau système présente des risques significatifs susceptibles d'affecter le projet et les opérations. Tout d'abord, il existe un risque de perte ou de corruption de données critiques, pouvant survenir en raison d'erreurs de manipulation, de défaillances techniques ou de problèmes de compatibilité, ce qui compromettrait la fiabilité et l'intégrité des informations transférées. Ensuite, une indisponibilité prolongée du service durant la migration pourrait perturber les utilisateurs, ralentir les activités opérationnelles et engendrer des pertes financières importantes.

Niveau d'impact: Moyen

Niveau de survenance: Moyen

Criticité: 4

Solution de remédiation:

- Effectuer un audit complet de l'ancienne application et définir clairement les objectifs de la migration.
- Procéder à une migration progressive.
- Impliquer les parties prenantes tout au long du processus pour assurer une transition en douceur.
- Prévoir un plan B pour revenir temporairement à l'ancienne application en cas de problème majeur.

Sous performance de la solution

Description: La nouvelle application risque de ne pas atteindre les objectifs de performance attendus, notamment en matière de rapidité d'exécution, de stabilité sous diverses conditions d'utilisation, ou de capacité à supporter les charges de travail actuelles et futures. Cela pourrait entraîner des ralentissements, des

interruptions de service ou une expérience utilisateur dégradée, compromettant ainsi l'efficacité opérationnelle et la satisfaction des utilisateurs finaux.

Niveau d'impact: Élevée

Niveau de survenance: Faible

Criticité: 3

Solution de remédiation:

- **Effectuer des tests de charge et de performance avant la migration complète.**
- **Prévoir des ajustements et optimisations basés sur les retours des tests.**
- **Implémenter des solutions scalables pour anticiper une augmentation future de la charge.**

Risque de sécurité:

Description: Risque d'exposition, de perte ou d'accès non autorisé aux données sensibles des clients, des commandes, et des stocks, pouvant entraîner des sanctions légales, une perte de confiance des utilisateurs, et un préjudice financier si les protocoles de sécurité ne sont pas correctement configurés, maintenus ou mis à jour.

Niveau d'impact: Élevé

Niveau de survenance: Faible

Criticité: 3

Solution de remédiation: Pour remédier au risque de sécurité, il est essentiel de mettre en œuvre des protocoles de sécurité robustes, incluant le chiffrement des données sensibles, une gestion stricte des accès, et des pare-feu avancés. Il convient d'effectuer des audits réguliers pour identifier et corriger les vulnérabilités, ainsi que de maintenir à jour les systèmes et logiciels de sécurité. Une sensibilisation des employés aux bonnes pratiques de cybersécurité et la mise en place d'un plan de réponse aux incidents garantiront une gestion rapide en cas de violation. Enfin, une conformité stricte avec les réglementations (RGPD, etc.) renforcera la protection et la confiance des parties prenantes.

Non-Adhésion des Utilisateurs à la Nouvelle Application

Description: Une adoption limitée de la nouvelle application pourrait réduire les bénéfices attendus du projet, entraîner des résistances au changement, et augmenter les coûts liés au support ou aux ajustements supplémentaires pour répondre aux retours des utilisateurs.

Niveau d'impact : Moyen

Niveau de survenance : Moyen

Criticité: 4

Solution de remédiation: Pour remédier à la non-adhésion des utilisateurs à la nouvelle application, il est crucial de mettre en place un plan de gestion du changement en sensibilisant et en formant les utilisateurs dès les premières phases du projet. Il faut intégrer les retours des utilisateurs via des tests pilotes pour adapter l'application à leurs besoins réels. Une communication claire sur les avantages et les impacts positifs de l'application doit être renforcée, accompagnée de sessions de formation et de supports adaptés. Enfin, un suivi post-déploiement, incluant un support réactif et des améliorations continues basées sur les retours, garantira une meilleure adoption.

Retard dans la livraison:

Description: Un retard dans la livraison pourrait entraîner une insatisfaction des parties prenantes, des coûts supplémentaires et un impact négatif sur les objectifs stratégiques de l'entreprise.

Niveau d'impact: Moyen

Niveau de survenance: Moyen

Criticité: 4

Solution de remédiation: Pour remédier aux retards de livraison, il est essentiel d'identifier et d'analyser les causes principales, puis de renforcer la planification en intégrant des marges de sécurité et en adoptant une méthode de gestion de projet adaptée. Une communication régulière avec les parties prenantes est cruciale pour anticiper les décalages. Parallèlement, il convient d'optimiser les processus de livraison, d'automatiser certaines étapes si possible et de renforcer les contrats avec les fournisseurs pour garantir le respect des délais. Enfin, la mise en place d'un suivi des risques et d'indicateurs de performance permettra de prévenir les problèmes futurs et d'améliorer continuellement les opérations.

Changements dans les Besoins du Client:

Description: Les modifications des besoins du client en cours de projet peuvent entraîner des retards, une augmentation des coûts et une complexité accrue dans la gestion des priorités et des ressources.

Niveau d'impact: Moyen

Niveau de survenance: Moyen

Criticité: 4

Solution de remédiation:

- Documenter les besoins initiaux et les valider avec les clients pour réduire les risques d'ambiguïté.

- *Prévoir une marge dans la planification pour intégrer des ajustements mineurs sans perturber l'ensemble du projet.*
- *Adopter une approche agile avec des livraisons incrémentales permettant d'intégrer les ajustements progressivement.*
- *Maintenir une communication continue avec le client pour identifier rapidement les changements nécessaires et évaluer leur impact.*

Manque de stockage/ressources

Description: Une insuffisance de stockage ou de ressources (matérielles ou logicielles) pourrait entraîner des interruptions du système, une dégradation des performances, et un échec dans la gestion des données, ce qui affecterait directement la continuité des opérations.

Niveau d'impact: Élevé

Niveau de survenance: Faible

Criticité: 3

Solution de remédiation:

- *Réaliser une évaluation détaillée des besoins en stockage et ressources dès les phases initiales du projet.*
- *Mettre en place un système de surveillance pour identifier les signes précoces de saturation comme alertes pour seuils critiques.*
- *Effectuer des tests de charge réguliers pour anticiper les éventuels besoins en capacité.*
- *Prévoir des sauvegardes et des plans de secours pour garantir la continuité du service en cas de saturation temporaire.*

Baisse de productivité en cas d'absence d'un membre clé de l'équipe de développement

Description: L'absence imprévue d'un collaborateur clé de l'équipe de développement, en raison de maladie ou d'autres circonstances, pourrait perturber l'avancement du projet, entraîner des retards et affecter la qualité du livrable final. Cette situation pourrait également nécessiter des ajustements dans la répartition des tâches ou un recrutement temporaire pour compenser la perte de compétences, augmentant ainsi les coûts et la complexité du projet.

Niveau d'impact: Élevé

Niveau de survenance: Faible

Criticité: 3

Solution de remédiation: Pour remédier à l'impact d'une absence imprévue d'un membre clé de l'équipe de développement, il est essentiel de mettre en place un plan de gestion des ressources, incluant une documentation des processus pour faciliter la prise en charge. Renforcer l'équipe avec des membres polyvalents et surveiller le bien-être des collaborateurs aidera à prévenir les absences. La planification de

scénarios alternatifs et l'ajout de marges de sécurité dans les délais permettront de minimiser les perturbations, avec la possibilité de recourir à des freelances si nécessaire.

12. Contraintes identifiées



[Présentez de manière succincte les contraintes à respecter sur ce projet (contraintes de temps, contraintes métiers, etc.)

Par exemple : Ne pas faire de migration informatique entre le 25 et le 31 du mois, car clôture comptable.]

Plusieurs contraintes ont été identifiées pour ce projet afin d'assurer son bon déroulement. Ces contraintes couvrent divers aspects du projet, notamment les délais, les impératifs métiers, techniques et organisationnels.

Contraintes de Temps :

Le client souhaite avoir les nouvelles fonctionnalités en production sous 4 semaines. CODESECURE s'engage à respecter ces délais

Contraintes Techniques:

Aucun travail de migration informatique ne doit être effectué entre le 25 et le 31 du mois, en raison de la clôture comptable.

La solution devra respecter les normes de sécurité les plus strictes, notamment en ce qui concerne la protection des données sensibles, telles que les informations des clients.

Le système devra être conforme au règlement RGPD (Règlement Général sur la Protection des Données) pour assurer la légalité du traitement des données personnelles.

Contraintes Budgétaires :

Le projet représente une charge humaine de 89 jours-homme répartis entre l'analyse des besoins (5 jours-hommes), le développement front-end (28 jours-hommes), l'intégration des API (28 jours-hommes) et les tests/validation (28 jours-hommes).

Selon la charge humaine du projet, voici le calcul du coût du projet en fonction des taux horaires des collaborateurs. Ils s'élèvent à une valeur de 350€/j pour un développeur junior, 400€/j pour un développeur Full-stack junior, et 350€/j pour un testeur junior et 700€/j pour chef de projet.

Le budget du projet est donc estimé à 39 460 € doit couvrir les coûts de développement, les licences logicielles, la maintenance et les formations nécessaires, avec un suivi régulier des dépenses.

Contraintes Métiers :

Aucun changement dans les processus métiers existants ne doit être effectué sans validation préalable du Product Owner et des parties prenantes.

Certaines fonctionnalités doivent être considérées comme prioritaires conformément aux priorités définies par le Product Owner.

Contraintes Organisationnelles:

Les ressources humaines affectées au projet doivent être disponibles à temps plein.

Les réunions avec le Product Owner devront être planifiées régulièrement pour éviter tout retard dû à des décisions manquées.

Toute modification du périmètre du projet devra être validée par toutes les parties prenantes afin de maintenir une bonne gestion des attentes.

Cf Matrice des risques en annexe

13. Dépendances



La fin de ce projet marquera le début du projet de MCO (Maintenance en conditions opérationnelles) d'une durée de 2 ans. Ce projet consiste à assurer le fonctionnement intégral de l'infrastructure applicative et la base de données.

Le projet de MCO sera entamé à condition que le projet de refonte du SI soit réalisé avec succès. Sinon, le projet de MCO sera reporté.

ORGANISATION DU PROJET

14. Comitologie



Pour le suivi du projet, nous organiserons une réunion de pilotage hebdomadaire tous les lundis en distanciel sur Teams. Elles seront utiles pour superviser le projet en évaluant l'avancement des objectifs imposés.

Ces points seront importants pour résoudre les problèmes rencontrés puis valider les jalons importants afin de passer aux prochaines phases du projet. Les réunions importantes seront la validation du recueil du besoin, la validation de l'outil et des scripts en préparation au déploiement puis pour finir, la validation de la documentation du programme.

La réunion de pilotage de projet est essentielle pour évaluer l'avancement global, prendre des décisions stratégiques et résoudre les problèmes. Elle permet de vérifier si les objectifs sont atteints, d'analyser les informations disponibles et de définir des mesures correctives si nécessaire. En résumé, cette réunion assure la supervision et la réussite du projet en maintenant le cap, en répondant aux exigences des parties prenantes et en assurant la conformité avec les délais, les coûts et la qualité prévus.

Réunion de copilotage sur site client à la fin première phase de développement et de tests pour valider l'outil et autoriser son exécution depuis les serveurs du client.

Nous pourrions alors effectuer des tests en conditions réelles ce qui va soit valider la solution pour un déploiement, soit l'invalidé et nous renvoyer vers une phase de développement pour préparer la validation de l'outil à la prochaine réunion de suivi.

Réunion go live après la première de développement et de documentation pour valider l'application et autoriser son déploiement.

La solution sera alors testée et documentée pour limiter les erreurs d'exécution et pour que le client ait la procédure d'utilisation.

Le tableau Kanban de CODESECURE sera utilisé pour planifier et organiser les différentes phases et réunions entre les parties prenantes.

Christophe Germain de Tecken effectuera la gouvernance de ce projet à travers la validation des tâches notamment.

Le compte-rendu des réunions sera rédigé par un des membres de l'équipe différent à chaque réunion. Il sera revu et validé par toute l'équipe.

15. Organisation globale du projet



Sprint 1: Analyse du besoin

Ce sprint comprend l'étude de la documentation technique, l'identification des contraintes et des priorités, la création d'un cahier des charges détaillé

Il se terminera par une réunion avec les parties prenantes pour clarifier les objectifs et valider le plan de travail

Sprint 2: Intégration API

Ce sprint comprend le choix des API à intégrer, la configuration des environnements, le développement de scripts pour consommer les API, la mise en place de tests unitaires pour vérifier la connectivité, ainsi que la documentation des processus d'intégration.

Sprint 3: Test et validation API

Ce sprint comprend la mise en place d'un plan de test complet (tests de performance, de sécurité, de compatibilité), l'exécution des tests sur les différentes fonctionnalités de l'API, la résolution des bugs identifiés lors des tests, ainsi que la validation finale avec les parties prenantes.

Sprint 4: Développement front end

Ce sprint comprend la conception des maquettes (wireframes et prototypes), le développement des interfaces utilisateur (UI), l'intégration des appels API dans les composants front-end, ainsi que les tests d'interaction utilisateur (UX).

Sprint 5+: Maintenance

Ce sprint comprend la surveillance des performances et la correction des anomalies, la mise à jour régulière de l'API et des dépendances, la documentation pour faciliter la maintenance future, ainsi que la formation ou le transfert de connaissances à l'équipe de support.

16. Chronologie & Planification



Le projet se déroulera du 23/12/2024 au 24/01/2025 pour les phases 1 à 4 (Modélisation puis Développement).

La phase 5 de maintenance débutera le 27/01/2025 et se terminera le 27/01/2027

Notre chemin critique sera le suivant:

Sprint 1 → Sprint 2 → Sprint 3 → Sprint 4 → Sprint 5

cf Planning en annexe

17. Plan de mise en œuvre



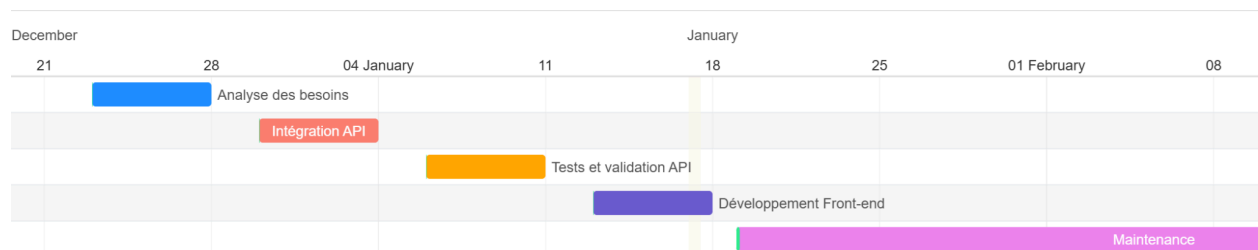
[Ajoutez des recommandations pour la solution proposée. Faites une synthèse de vos propositions et des moyens mis en œuvre pour atteindre les objectifs.

Ces informations seront développées dans le plan d'action du projet et le planning détaillé.]

ANNEXES

	Olivier Koenig	Alexandre Sacks	Maxime Godineau	Sylvie Kelonkeu	Christophe Germain	Digicheese
Deliverable or Task	Développeurs				CDP	Client
Sprint 1						
Analyse du besoin	R	R	R	R	A	R
Sprint 2						
Intégration API	R	R	C	R	A	A
Sprint 3						
Tests et validation API	C	R	R	C	A	A
Sprint 4						
Développement Front-end	R	C	C	R	A	A
Sprint 5+						
Maintenance	R	R	R	R	C	A

Annexe 1: RACI



Annexe 2: Planning

		Impact		
		Faible	Moyen	Elevé
Probabilité	Elevé	3 - Moyen	6 - Elevé	9 - Elevé
	Moyen	2 - Faible	4 - Moyen	6 - Elevé
	Faible	1 - Faible	2 - Faible	3 - Moyen

Annexe 3: Matrice des risques

APPROBATION ET AUTORISATION

Nous approuvons le projet tel qu'il est décrit ci-dessus, et autorisons l'équipe à le mettre en œuvre.

Nom	Titre	Date
Christophe GERMAIN	Chef de projet	17/01/2025
KOENIG Olivier	Développeur concepteur	17/01/2025
KELONKEU Sylvie	Développeur concepteur	17/01/2025
SACKS Alexandre	Développeur concepteur	17/01/2025
GODINEAU Maxime	Développeur concepteur	17/01/2025

Approuvé par Alexandre
Sacks

Date
17/01/2025

Approuvé par Sylvie
Kelonkeu

Date
17/01/2025

Signature



Signature




Approuvé par Koenig Olivier

Date
17/01/2025

Approuvé par Maxime

Date

Signature



Signature