

Contrat de développement d'architecture business



Projet : Approvisionnement alimentaire géographiquement responsable

Client : Foosus

Préparé par : Emrys Callait

Table des matières

Contrat de développement d'architecture	1
Projet : Approvisionnement alimentaire géographiquement responsable	1
Client : Foosus	1
Préparé par : Emrys Callait	1
Objet de ce document	3
Introduction et contexte	4
La nature de l'accord	4
Objectifs	4
Objectifs stratégiques et commerciaux.....	4
Croissance et expansion	6
Efficacité opérationnelle	6
Expérience et accès utilisateur.....	7
Périmètre.....	8
Parties prenantes, préoccupations et visions	9
Description de l'architecture, principes stratégiques et conditions requises.....	11
Description	11
Principes stratégiques	12
Référence aux conditions requises pour l'architecture	13
Livrables architecturaux	15
Plan de travail commun priorisé	15

Plan de travail.....	15
Initialisation et Vision (Phases Préliminaire & A)	15
Définition de l'Architecture Cible (Phases B, C & D).....	16
Planification de la Transition (Phases E & F)	17
Gouvernance de l'Implémentation (Phase G)	17
Plan de communication.....	17
Évènements.....	17
Canaux	18
Formats.....	18
Rythme de communication	19
Risques et facteurs de réduction.....	20
Analyse des risques	20
Hypothèse	21
Critères d'acceptation et procédures.....	22
Métriques Business de l'architecture (KPI)	22
Métriques de livraison de l'architecture et du business.....	22
Procédure d'acceptation	23
Procédures de changement de périmètre	24
Conditions requises pour la conformité	24
Développement et propriété de l'architecture	25
Calendrier	26
Personnes approuvant ce plan.....	27

Objet de ce document

Les Contrats d'Architecture sont les accords communs entre les partenaires de développement et les sponsors sur les livrables, la qualité, et la correspondance à l'objectif d'une architecture. L'implémentation réussie de ces accords sera livrée grâce à une gouvernance de l'architecture efficace (voir TOGAF Partie VII, Gouvernance de l'architecture). En implémentant une approche dirigée du management de contrats, les éléments suivants seront garantis :

- *Un système de contrôle continu pour vérifier l'intégrité, les changements, les prises de décisions, et l'audit de toutes les activités relatives à l'architecture au sein de l'organisation.*
- *L'adhésion aux principes, standards et conditions requises des architectures existantes ou en développement*
- *L'identification des risques dans tous les aspects du développement et de l'implémentation des/de l'architecture(s), y compris le développement interne en fonction des standards acceptés, des politiques, des technologies et des produits, de même que les aspects opérationnels des architectures de façon à ce que l'organisation puisse poursuivre son business au sein d'un environnement résilient.*
 - *Un ensemble de processus et de pratiques qui garantissent la transparence, la responsabilité et la discipline au regard du développement et de l'utilisation de tous les artefacts architecturaux*
 - *Un accord formel sur l'organe de gouvernance responsable du contrat, son degré d'autorité, et le périmètre de l'architecture sous la gouvernance de cet organe*

Ceci est une déclaration d'intention signée sur la conception et le développement de l'architecture d'entreprise, ou de parties significatives de celles-ci, de la part d'organisations partenaires, y compris les intégrateurs système, fournisseurs d'applications, et fournisseurs de service.

De plus en plus, le développement d'un ou plusieurs domaine(s) d'architecture (business, données, application, technologie) peut être externalisé, avec la fonction d'architecture de l'entreprise fournissant une vue d'ensemble de l'architecture d'entreprise globale, ainsi que la coordination et le contrôle de l'effort total. Dans certains cas, même ce rôle de supervision peut être externalisé, bien que la plupart des entreprises préfèrent conserver cette responsabilité clé en interne.

Quelles que soient les spécificités des dispositions d'externalisation, les dispositions elles-mêmes seront normalement gouvernées par un Contrat d'Architecture qui définit les livrables, la qualité, et la correspondance à l'objectif de l'architecture développée, ainsi que les processus de collaboration pour les partenaires du développement de l'architecture.

Introduction et contexte

Foosus est une startup spécialisée dans la mise en relation entre les consommateurs et les producteurs alimentaires locaux. Sa mission est de soutenir la consommation de produits locaux et durables en offrant une plateforme qui connecte directement les clients avec des agriculteurs et artisans de leur région.

La base d'utilisateurs de Foosus comprend à la fois des consommateurs à la recherche de produits frais et locaux, et des fournisseurs alimentaires cherchant à vendre leurs produits à une clientèle de proximité.

L'entreprise est actuellement en pleine évolution technologique pour soutenir son expansion dans de nouvelles régions géographiques et sa croissance.

La nature de l'accord

Ce contrat de conception et de développement établit les principes de collaboration et les responsabilités partagées entre la fonction d'architecture, les équipes de conception et de développement de Foosus. Il a pour but de fournir des standards et un cadre de travail cohérent pour accélérer l'innovation tout en préservant l'autonomie des équipes, conformément à la culture agile de l'entreprise. Ce document est évolutif, il sera révisé et amélioré en continu avec l'accord des parties prenantes.

Objectifs

Objectifs stratégiques et commerciaux

Soutenir l'économie locale

L'objectif principal est de soutenir la consommation de produits alimentaires locaux et de mettre en relation les clients avec des producteurs et artisans locaux.

Compétitivité

Développer une nouvelle plateforme d'e-commerce est nécessaire pour améliorer la compétitivité de Foosus face aux grandes entreprises internationales du secteur

Passer à l'étape supérieure

Créer une plateforme de commerce électronique polyvalente pour permettre à l'entreprise de franchir un nouveau cap de croissance

Innovation rapide

La nouvelle plateforme doit permettre aux équipes produit d'innover rapidement, d'expérimenter et de réorienter des solutions existantes. L'un des buts est de revenir à un niveau où l'entreprise peut innover rapidement pour attirer de nouveaux utilisateurs

Croissance et expansion

Expansion géographique

L'architecture doit être évolutive pour permettre le déploiement des services dans diverses régions, villes et pays. La nouvelle architecture doit pouvoir prendre en charge un nouvel emplacement géographique

Augmentation des adhésions

- Augmenter le nombre d'adhésions d'utilisateurs de 10%/j
- Faire passer l'adhésion de producteurs alimentaires de 1,4 par mois à 4 par mois

Scalabilité

La pile technologique doit pouvoir évoluer au même rythme que la base de clientèle pour absorber les pics d'utilisation et les campagnes marketing. L'objectif est de pouvoir supporter le prochain million d'utilisateurs inscrits

Efficacité opérationnelle

Réduction du time to market

Réduire le délai moyen de parution de 3,5 semaines à moins d'une semaine

Fiabilité et stabilité

- Réduire le taux d'incidents de production P1 de plus de 25 par mois à moins de 1 par mois
- Fournir la fiabilité nécessaire aux clients, fournisseurs et consommateurs

Déploiement sans interruption

Les modifications apportées aux systèmes de production devront supprimer la nécessité d'interrompre le service. Il n'est plus acceptable de désactiver la plateforme pour chaque nouvelle version

Expérience et accès utilisateur

Utilisation de la géolocalisation

Tirer parti de la géolocalisation pour connecter fournisseurs et consommateurs et proposer des produits à proximité

Multi-plateforme

La solution doit être accessible et utilisable aussi bien sur des appareils mobiles que fixes, en prenant en compte les contraintes de bande passante, que ce soit pour les réseaux cellulaires ou les connexions haut débit

Haute disponibilité

Les services doivent être disponibles 24/7 et accessibles pour les fournisseurs et consommateurs où qu'ils se trouvent

Sécurité

L'approche doit garantir la sécurité pour éviter tout risque pour l'image de marque de l'entreprise

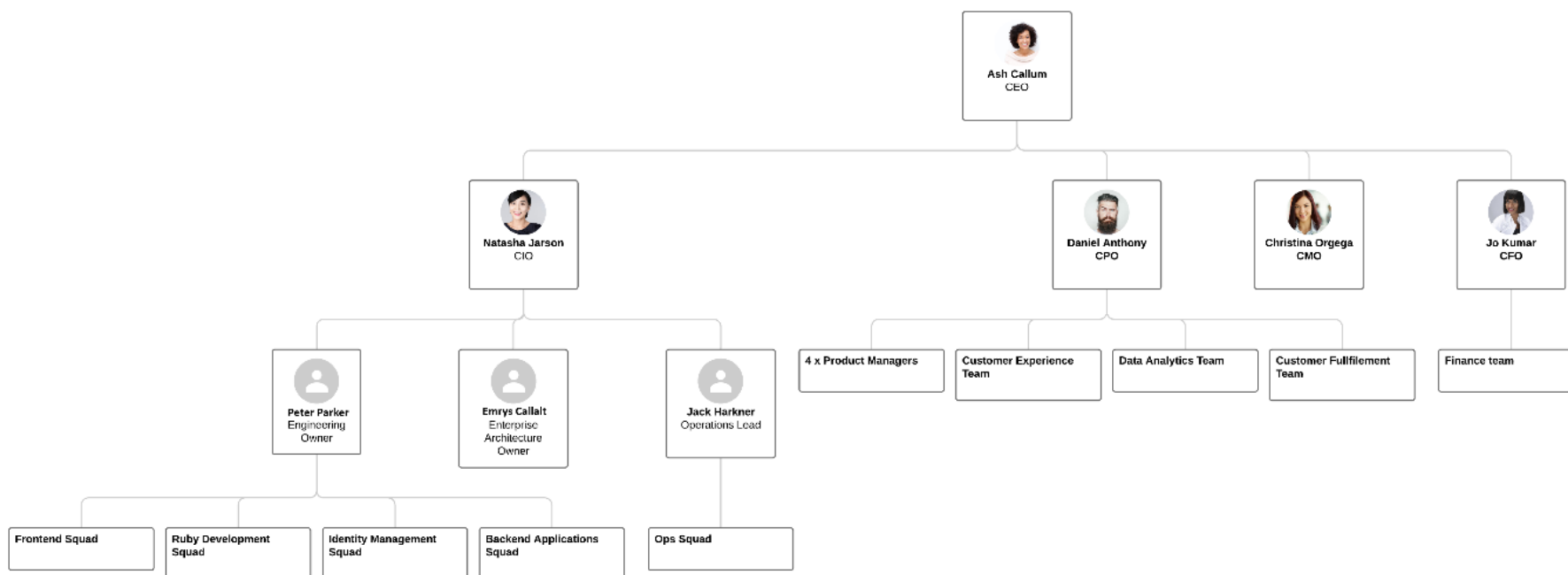
Périmètre

Foosus ambitionne de déployer des campagnes marketing dans le monde entier, tout en garantissant que sa plateforme reste disponible, performante et offre une expérience utilisateur optimale

La plateforme devra ainsi être opérationnelle 24/7, quelles que soient les conditions de bandes passantes, disponibilité, les langues et devra être capable d'absorber les pics de charge sans dégradation de service

Parties prenantes, préoccupations et visions

Lien Organigramme : <https://github.com/EmrysC/Concevez-une-nouvelle-architecture-afin-de-soutenir-le-d-veloppement-de-votre-entreprise/blob/master/images/org-chart.png>



Partie prenante	Préoccupation	Vision
Equipe de direction	Natasha Jarson : CIO (co-fondatrice) Supervise les aspects informatiques du projet, s'assure de l'alignement entre les équipes techniques et les objectifs business	Stratégique globale de l'entreprise, budget, produit et vente
Responsable du projet et de la stratégie	Daniel Anthony : CPO Sa responsabilité est de s'assurer que la nouvelle plateforme répond aux besoins des équipes produit et des utilisateurs finaux, en définissant les exigences fonctionnelles et les indicateurs de succès (KPIs) Emrys Callait : Enterprise Architecture Owner Est un acteur central de la phase de conception. Sa mission est de définir l'architecture technique cible, de créer un plan de migration et de s'assurer que la solution est évolutive, fiable et rentable, en collaboration avec toutes les autres parties prenantes	Vision produit, architecture, gestion de la donnée
Equipe technique	Peter Parker : Engineering Owner Responsable des équipes de développement. Il est chargé de transformer l'architecture définie en un plan d'action concret et de gérer les développeurs qui construiront la nouvelle plateforme Jack Harkness : Operation Lead Evalue la faisabilité opérationnelle de la nouvelle architecture et s'assure qu'elle pourra être maintenue et exploitée de manière efficace une fois en production Equipe technique Acteurs de la réalisation, elles sont responsables de la construction, du test et du déploiement de la nouvelle plateforme. Leur adhésion et leur expertise sont cruciales pour le succès de l'implémentation	Gestion des ressources, gestion de la donnée, réalisation, technique, maintenabilité et résilience

Description de l'architecture, principes stratégiques et conditions requis

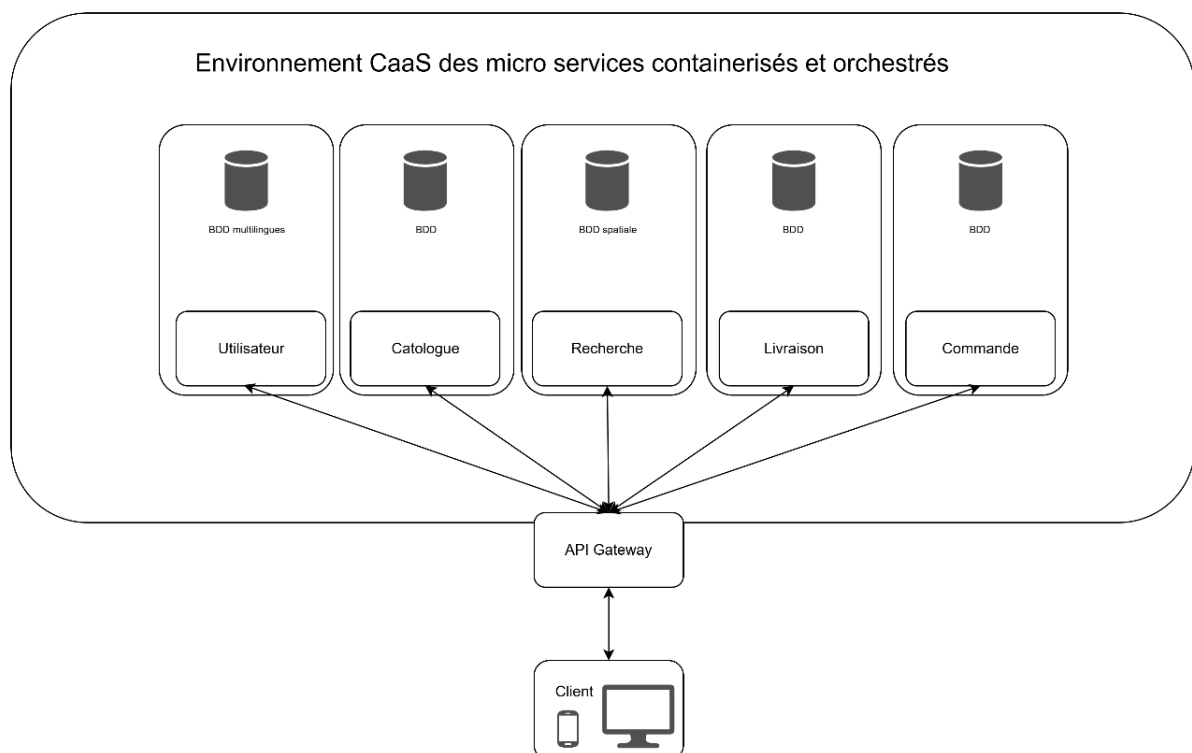
Description

Une architecture basée en micro services avec communication APIs est le modèle choisi, elle consiste à décomposer l'application en un ensemble de petits services indépendants, chacun responsable d'un domaine métier spécifique (utilisateurs, produits, recherche, etc.).

Ce choix répond directement à nos défis. Il permet aux équipes de développer, déployer et faire évoluer leurs services de manière autonome, ce qui augmente l'agilité et donc le « Time to Market ».

La défaillance d'un service n'entraîne pas la panne de toute la plateforme, garantissant une meilleure fiabilité.

Enfin, avec la containerisation et l'orchestration, chaque service peut être dimensionné indépendamment pour répondre aux besoins de scalabilité, tout en améliorant la sécurité grâce au principe de privilège de ressource des conteneurs.



Principes stratégiques

Continuité de Service

Ce principe garantit que la plateforme Foosus est constamment disponible et fiable pour ses utilisateurs. L'entreprise étant désormais active 24/7, il n'est plus acceptable de désactiver la plateforme pour effectuer des déploiements ou des modifications. Les mises à jour devront donc être transparentes pour les utilisateurs. La fiabilité est également un pilier de ce principe, avec un objectif clair de réduire les incidents de production critiques (P1) de plus de 25 par mois à moins de 1 par mois. Même en cas de forte charge, le système doit pouvoir continuer de fonctionner de manière dégradée plutôt que de subir une panne complète.

Réactivité au changement

La nouvelle architecture doit permettre à Foosus de s'adapter rapidement aux évolutions du marché et aux retours des clients. Pour cela, elle doit permettre aux équipes produit d'innover rapidement, d'expérimenter et de réorienter des solutions existantes facilement. Un aspect fondamental de ce principe est la réversibilité : les décisions d'architecture doivent pouvoir être inversées à faible coût afin de minimiser les risques liés à l'incertitude du secteur. Cette agilité se traduira par une réduction du délai de mise sur le marché, qui passera de 3,5 semaines à moins d'une semaine

Scalabilité

La scalabilité est la capacité de l'architecture à évoluer pour soutenir la croissance de l'entreprise. La plateforme doit être conçue pour évoluer naturellement avec l'augmentation de la base de clientèle et pour gérer sans difficulté les pics d'utilisation dus aux campagnes marketing. Ce principe est essentiel pour permettre le déploiement des services Foosus sur de nouvelles régions géographiques, que ce soit dans différentes villes ou différents pays

Sécurité

La sécurité n'est plus une option mais une exigence fondamentale pour protéger l'image de marque de Foosus. Par le passé, des choix priorisant la facilité d'utilisation sur la sécurité ont failli nuire à la réputation de l'entreprise. La nouvelle approche doit donc garantir la sécurité à chaque étape de l'élargissement de la plateforme, afin d'éviter tout risque pour la marque

Standardisation et maîtrise des coûts

Ce principe vise à mettre fin à l'ensemble hétérogène de technologies résultant d'expérimentations passées. En réponse à une demande des équipes de développement elles-mêmes, une standardisation sera mise en place pour faciliter la maintenance. Pour maîtriser les coûts, les solutions open source seront préférées aux solutions payantes et les composants devront, dans la mesure du possible, faire partie d'une même pile technologique pour réduire les frais de support continus

Référence aux conditions requises pour l'architecture

Exigences applicatives et de conception

- Modularité des Services : L'architecture doit être décomposée en services indépendants (microservices), chacun responsable d'un domaine métier unique, afin de supporter le principe de réactivité au changement
- Communication par APIs : Toute communication entre les services doit se faire via des APIs standardisées et versionnées pour garantir l'interopérabilité et le découplage
- Déploiement indépendant : Chaque service doit pouvoir être déployé de manière autonome, sans interruption de service pour les autres composants (continuité de service)

Exigences non fonctionnelles

- Haute Disponibilité : Le système doit garantir une disponibilité de service 24/7, avec pour objectif la réduction des incidents critiques (P1) à moins de 1 par mois
- Scalabilité : L'architecture doit être conçue pour gérer la croissance prévue (prochain million d'utilisateurs, pics de charge marketing) par une mise à l'échelle horizontale et automatisée des services
- Performance : La plateforme doit offrir un temps de réponse optimal, y compris sur des connexions à faible bande passante (mobile), pour garantir une expérience utilisateur fluide.

- Fiabilité : En cas de défaillance d'un service non critique, le système global doit rester opérationnel en mode dégradé plutôt que de subir une panne complète

-

Exigences technologiques

- Containerisation : Tous les services applicatifs devront être packagés dans des conteneurs (ex : Docker) pour assurer la portabilité et la standardisation des déploiements
- Orchestration : Un orchestrateur de conteneurs (ex : Kubernetes) doit être utilisé pour gérer automatiquement la mise à l'échelle, le déploiement et la résilience des services

Exigences de sécurité

- **Sécurité dès la conception ("Security by Design")** : Les mécanismes de sécurité (authentification, autorisation, chiffrement des données) doivent être intégrés nativement dans l'architecture et non ajoutés après coup (**principe de sécurité**)
- **Isolation des services** : La compromission d'un service ne doit pas permettre d'accéder aux autres services ou à leurs données.

Exigences de conformité

- **Conformité RGPD** : La gestion des données personnelles des utilisateurs et des producteurs doit être en stricte conformité avec le Règlement Général sur la Protection des Données (RGPD)

Livrables architecturaux

Livrables architecturaux qui satisfont aux conditions requises pour le business.

Développement de l'architecture : Document Technique d'Application DTA

Détail les spécifications techniques, les composants, les interfaces, les flux de données, les contraintes et les exigences fonctionnelles et non fonctionnelles d'une application

Mesures de l'architecture cible : Spécification des conditions requises

Livraison de l'architecture et métriques business : Contrat, KPI

Phases de livraison définies : Roadmap

Plan de travail commun priorisé

Cette section décrit toutes les activités et tous les livrables pour le travail d'architecture.

Plan de travail

Initialisation et Vision (Phases Préliminaire & A)

Objectif : Établir une compréhension commune des objectifs et définir la vision et les principes directeurs de la nouvelle architecture.

Tâche	Livrables associé
Analyse et entretien initiaux	Compte-rendu de synthèse des attentes et des contraintes
Élaboration de la Vision et des Principes	Section "Vision" et "Principes Stratégiques" de la Déclaration de Travail
Rédaction de la Déclaration de Travail d'Architecture	Version finale du document, incluant la vision, les objectifs, les principes et le périmètre
Spécification des exigences d'architecture	Document des Spécifications (techniques, de conformité, ANS)

Définition de l'Architecture Cible (Phases B, C & D)

Objectif : Concevoir l'architecture cible et formaliser la collaboration avec les équipes via des contrats clairs.

Tâche	Livrables associé
Conception de l'architecture cible	Diagrammes d'architecture (micro services), description des composants, choix technologiques
Revue de l'architecture	Compte-rendu d'ateliers, ajustements de la conception
Rédaction des contrats d'architecture	Contrat d'architecture avec le business, contrat avec le développement
Validation et signature des contrats	Contrats d'architecture validés et signés

Planification de la Transition (Phases E & F)

Tâche	Livrables associé
Mise en place du référentiel d'architecture	Dépôt GitHub structuré avec modèles, directives et documentation finalisée, technologies
Présentation	Support de présentation pour l'équipe de direction et aux équipes de développement
Suivi	Roadmap / Gant

Gouvernance de l'Implémentation (Phase G)

Tâche	Livrables associé
Mise en place du système de ticket	Système de ticketing
Documentation technique et fonctionnel	Support de formation

Plan de communication

Évènements

Pour assurer une communication fluide et un suivi efficace, le projet sera rythmé par trois types de réunions principales :

Atelier de Lancement (Kick-off)

Une fois l'architecture validée, un atelier de lancement réunira l'équipe de développement, le responsable produit et l'architecte. L'objectif sera de présenter en détail le périmètre du prototype, de revoir les lignes directrices de l'architecture et de définir clairement les rôles et responsabilités de chacun pour la phase d'implémentation.

Daily

En accord avec les pratiques Agiles de Foosus, des réunions de suivi fréquentes (quotidiennes ou hebdomadaires) seront organisées au niveau de la direction du projet. Elles permettront de suivre l'avancement, de s'assurer du respect des contraintes techniques et architecturales et d'ajuster le plan d'action de manière réactive pour lever les éventuels blocages

Démonstration et revue du prototype

À la fin de la phase de développement, une session de démonstration sera organisée avec les responsables et commanditaires du projet. Cette réunion de clôture aura pour but de présenter le prototype fonctionnel, de valider sa conformité aux objectifs initiaux et de décider des prochaines étapes pour le déploiement de la solution

Canaux

La communication et la collaboration entre les équipes seront assurées via la suite Microsoft Office 365 : Teams sera utilisé pour les réunions et la messagerie instantanée, Outlook pour les échanges par e-mail, et Planner pour le suivi des tâches en mode Kanban. Un dépôt GitHub partagé fera office de référentiel unique pour l'ensemble des artefacts architecturaux

Un système de ticketing avec la priorité, les impacts, le temps de prise en charge, le temps de résolution et les cause doit être mise en place pour les remonter des incident et leur analyse

Formats

Les documents produits (rapports, spécifications, comptes rendus, etc.) seront rédigés aux formats natifs de la suite Office (Word, Excel, PowerPoint) afin de garantir une compatibilité maximale

Rythme de communication

Daily meetings (Kanban)

(15 minutes chaque jour) pour faire le point sur les tâches en cours, les éventuels blocages et les priorités du jour

Point d'avancement hebdomadaire (Sprint)

(Réunion d'équipe ou comité projet) pour faire un suivi global de l'état d'avancement, des livrables, des risques et des jalons

Risques et facteurs de réduction

Analyse des risques

ID	Risque	Gravité	Probabilité	Facteur de Réduction	Propriétaire
1	Interruption des services	Majeure	Possible	Utiliser des outils de monitoring, utiliser une architecture en micro services	Responsable Ingénierie / Equipe de développement
2	Surcharge des services	Majeure	Probable	Mise à l'échelle horizontale des serveurs, dockerisation et orchestration, utilisation de Caas scalable	Responsable Ingénierie / Architecte logiciel
3	Faible de sécurité	Majeure	Peu probable	Mettre en place un système de double authentification, ne pas utiliser des technologies avec des failles connues	RSSI / Responsable Ingénierie / Architecte logiciel
4	Capacité des serveurs (volumétrie)	Forte	Forte	Validation en phase de test.	Equipe de développement / testeur
5	Les équipes de développement rencontrent des difficultés dans l'utilisation de la nouvelle plateforme	Forte	Moyenne	Plan de formation	Responsable d'équipe
6	Les équipes support maîtrisent mal la nouvelle application	Forte	Moyenne	Réunions de suivi, formation	Responsable d'équipe
7	Non-respect du budget ou du délai	Majeure	Possible	Prévoir des marges, modifier le planning si besoin, augmenter ou diminuer l'effectif de l'équipe de développement	CIO / Responsable Ingénierie / Equipe de développement
8	L'utilisation de 2 plateformes en parallèle peut mettre l'équipe support en difficulté	Moyenne	Faible	Mettre en place une procédure	Responsable d'équipe

Hypothèse

ID	Risque	Impact	Propriétaire
1	Interruption des services	Dégradation de l'expérience utilisateur, perte de confiance des clients et fournisseurs, incapacité à soutenir les campagnes marketing, ce qui freine l'acquisition de nouveaux utilisateurs	Responsable Ingénierie / Equipe de développement
2	Surcharge des services	Panne des services lors des pics d'utilisation, mauvaise réactivité de la plateforme et impossibilité d'atteindre les objectifs de croissance en termes d'utilisateurs et de programmes marketing	Responsable Ingénierie / Architecte logiciel
3	Faible de sécurité	Atteinte grave à la réputation et à l'image de marque de l'entreprise, entraînant une perte de confiance des utilisateurs et des partenaires	RSSI / Responsable Ingénierie / Architecte logiciel
4	Capacité des serveurs (volumétrie)	Limitation de la croissance du nombre d'utilisateurs et de l'expansion géographique. Le système ne pourrait pas supporter le niveau d'engagement attendu pour les futurs programmes marketing	Equipe de développement / testeur
5	Les équipes de développement rencontrent des difficultés dans l'utilisation de la nouvelle plateforme	Ralentissement du développement, augmentation du délai de mise sur le marché des nouvelles fonctionnalités et risque de ne pas atteindre les objectifs d'innovation rapide	Responsable d'équipe
6	Les équipes support maîtrisent mal la nouvelle application	Augmentation du temps de résolution des problèmes clients, baisse de la satisfaction des utilisateurs et surcharge de travail pour les équipes de support	Responsable d'équipe
7	Non-respect du budget ou du délai	Pour la phase initiale, un retard dans la définition de l'architecture repousserait le début du développement du prototype. Sur le projet global, cela signifie des coûts accrus et un retard dans la livraison de valeur à l'entreprise	CIO / Responsable Ingénierie / Equipe de développement
8	L'utilisation de 2 plateformes en parallèle peut mettre l'équipe support en difficulté	Complexité opérationnelle accrue, confusion pour les équipes de support devant maîtriser deux systèmes, et potentiellement des temps de réponse plus longs pour les clients durant la phase de transition	Responsable d'équipe

Critères d'acceptation et procédures

Métriques Business de l'architecture (KPI)

Le projet sera considéré comme réussi si tu les indicateurs sont validés.

Indicateur	Changement souhaité pour l'indicateur
Nombre d'adhésions d'utilisateurs par jour	Augmentation de 10 %
Adhésion de producteurs alimentaires	Passer de 1,4/mois à 4/mois
Délai moyen de parution	Réduit de 3,5 semaines à moins d'une semaine
Taux d'incidents de production P1	Pour commencer : réduit de >25/mois à moins de 1/mois.

Métriques de livraison de l'architecture et du business

Métrique	Technique de mesure	Valeur cible	Justification
Fiabilité et Haute Disponibilité	Monitoring du temps de disponibilité	P1 -> -1%	L'application doit être disponible en tout temps
Sécurité dès la Conception ("Security by Design")	Audit de sécurité	Aucun problème critique détecté	L'application ne doit pas avoir de problème de sécurité
Rentabilité et Maîtrise des Coûts	Monitoring coût hébergement	Le coût / utilisateur ne doit pas augmenter voire diminuer	Si le coût par utilisateur augmente la rentabilité de l'entreprise va être impactée
Agilité et Innovation / Time to Market	Calcul du temps entre la proposition de l'idée et le déploiement (Monitoring par Jira)	3 semaines	La caractéristique de différenciation de l'entreprise est son innovation
Scalabilité	Analyse et retour des incident	0 incident lié à l'augmentation d'utilisateur	Avec un objectif de croissance de 10%/j la scalabilité doit être prévue au moment du design et donc ne pas impacter la production

Procédure d'acceptation

Une fois les phases 1 (Cadrage et Vision) et 2 (Définition de l'architecture cible) achevées, une procédure d'acceptation formelle est mise en place. Le document de « Contrat de développement d'architecture » est présenté lors d'une réunion dédiée réunissant :

La direction :

- Natasha Jarson en tant que **CIO** et cofondatrice qui supervise l'alignement stratégique du projet et validera la cohérence globale de la proposition avec les objectifs métiers.

Le responsable du projet et de la stratégie :

- Daniel Anthony en tant que **CPO**, évaluera quant à lui l'adéquation de l'architecture proposée avec les besoins des équipes produit et des utilisateurs finaux, en s'appuyant notamment sur les exigences fonctionnelles et les KPIs définis.

L'équipe projet :

- Jack Harker en tant que operation **lead** complétera cette revue en s'assurant que la future solution s'intègre correctement dans les processus de production et d'exploitation, en tenant compte des impératifs de performance, de disponibilité et de maintenabilité
- Pete Parker en tant que **responsable ingénierie** interviendra pour garantir la faisabilité technique de la solution au regard des capacités des équipes de développement et des contraintes opérationnelles. Il apportera une vision terrain sur l'impact de l'architecture cible en matière d'implémentation, d'organisation et de charge de travail
- Emrys Callait, en tant qu'**architecte**, défendra les choix techniques retenus

À l'issue de cette réunion, chaque membre de l'équipe pourra formuler ses remarques éventuelles. Après prise en compte des retours, le document sera soumis à une validation finale afin d'acter officiellement l'acceptation de cette étape clé du projet.

Procédures de changement de périmètre

Notre processus vise à intégrer le changement de manière agile et responsable, en alignant l'innovation rapide avec une architecture robuste.

Soumission : Toute demande de changement doit être formulée via un "ticket de demande de changement", justifiant le besoin métier ou technique

Évaluation : L'Architecte, le Responsable Ingénierie et le Directeur Produit évaluent l'impact sur l'architecture, les coûts et les délais

Décision : Les demandes sont revues chaque semaine pour statuer sur leur acceptation en fonction du rapport valeur/impact

Approbation : Le trio (Architecte, Ingénierie, Produit) approuve les changements mineurs, la validation de la CIO est requise pour les changements majeurs

Implémentation : Une fois approuvée, la modification est priorisée et intégrée dans le backlog de l'équipe de développement

Documentation : Toute modification de l'architecture est consignée dans le dépôt GitHub pour garantir la transparence et la cohérence

Conditions requises pour la conformité

Pour être validée, la nouvelle architecture doit respecter deux points essentiels :

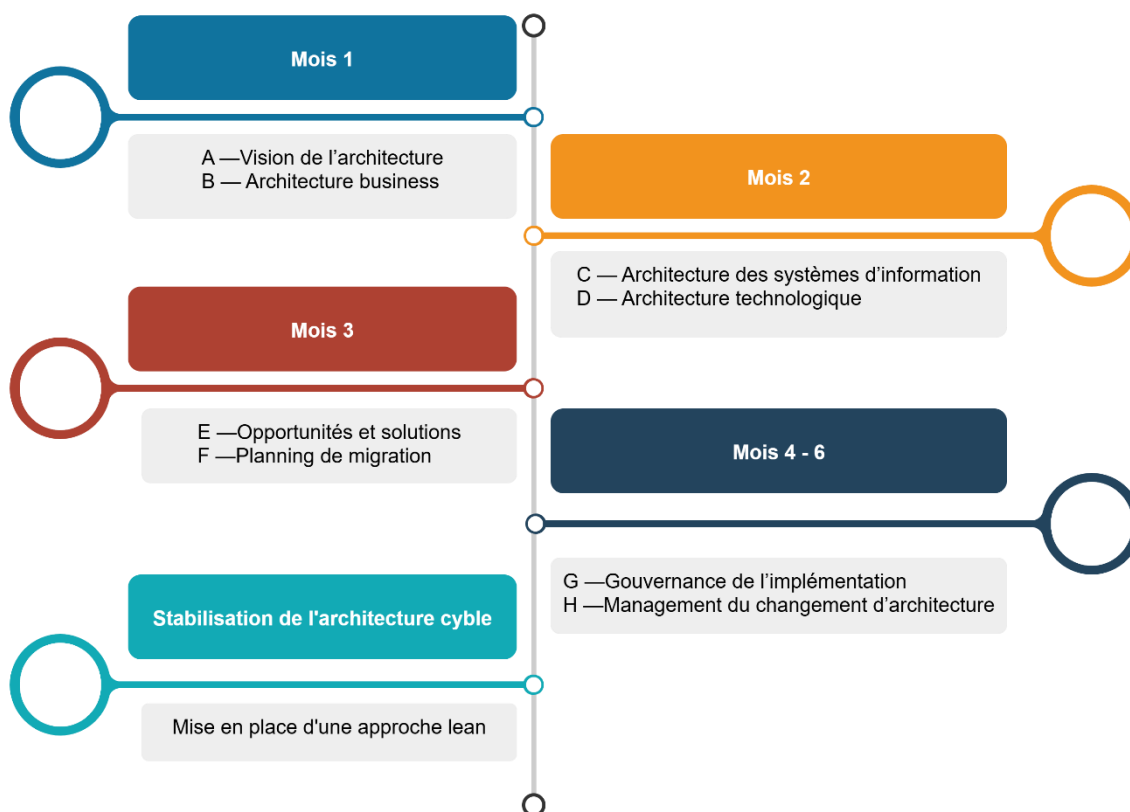
Légalité : Elle doit être conforme aux lois (RGPD)

Objectifs : Elle doit répondre à tous les objectifs business et techniques listés dans ce document

Développement et propriété de l'architecture

Nom	Rôle	Responsabilité
Natasha Jarson	CIO	Supervise l'alignement stratégique du projet et validera la cohérence globale de la proposition avec les objectifs métiers
Daniel Anthony	CPO	Évalue quant à lui l'adéquation de l'architecture proposée avec les besoins des équipes produit et des utilisateurs finaux, en s'appuyant notamment sur les exigences fonctionnelles et les KPIs définis
Jack Harker	Operation Lead	S'assure que la future solution s'intègre correctement dans les processus de production et d'exploitation, en tenant compte des impératifs de performance, de disponibilité et de maintenabilité
Pete Parker	Responsable Ingénierie	Intervient pour garantir la faisabilité technique de la solution au regard des capacités des équipes de développement et des contraintes opérationnelles. Il apportera une vision terrain sur l'impact de l'architecture cible en matière d'implémentation, d'organisation et de charge de travail
Équipe de développement	Développeur	A pour rôle de collaborer à la définition d'une architecture de micro services bien structurée et de construire la nouvelle plateforme, en équilibrant l'innovation rapide avec des pratiques de réutilisation stratégique pour éviter de ralentir le rythme de livraison

Calendrier



Personnes approuvant ce plan

Natasha Jarson

Date : __ / __ / ____

Signature : _____

Daniel Anthony

Date : __ / __ / ____

Signature : _____

Jack Harker

Date : __ / __ / ____

Signature : _____

Pete Parker

Date : __ / __ / ____

Signature : _____

Emrys Callait

Date : __ / __ / ____

Signature : _____