

01 NOVEMBRE 2024



## Déclaration de Travail d'Architecture

LOUIS ZEPHIR  
FOOSUS

*Note : Ce document fournit un modèle générique. Il pourra nécessiter des modifications pour correspondre à un client et une situation de projet spécifiques.*

## Information sur le document

Nom du projet	Projet XXX
Préparé par :	
N° de version du document :	0.1
Titre :	<i>Déclaration de travail d'architecture</i>
Date de version du document :	
Revu par :	
Date de révision :	
Liste de distribution :	
De :	
Date :	
Email :	
Pour Action :	
Date de rendu :	
Email :	
Types d'action :	Approbation, Révision, Information, Classement, Action requise, Participation à une réunion, Autre (à spécifier)
Historique de versions du document	Voir git

## Table des matières

Information sur le document .....	1
Objet de ce document.....	4
Déclaration de travail d'architecture .....	4
Requête du projet et contexte .....	4
Description du projet et périmètre .....	5
Vue d'ensemble .....	5
Alignement stratégique.....	5
Objectifs et périmètre .....	6
Objectifs .....	6
Périmètre.....	6
Parties prenantes, préoccupations, et visions .....	7
Approche managériale .....	7
Procédures de changement de périmètre.....	8
Rôles et responsabilités .....	9
Structure de gouvernance .....	9
Process du projet .....	10
Rôles et responsabilités (RACI) .....	10
Approche architecturale .....	11
Process d'architecture .....	11
Contenu de l'architecture .....	13
Plan de travail.....	14
Élément de travail 1 : Définition de la vision architecturale .....	15
Activités.....	15
Livrables .....	15
Élément de travail 2 : Analyse et modélisation de l'architecture actuelle .....	15
Activités.....	15
Livrables .....	15
Élément de travail 3 : Définition de l'architecture cible .....	15
Activités.....	15
Livrables .....	16
Élément de travail 4 : Planification et mise en œuvre .....	16
Activités.....	16
Livrables .....	16
Élément de travail 5 : Tests et validation .....	16
Activités.....	16
Livrables .....	16
Plan de communication .....	17

<b>Évènements.....</b>	<b>17</b>
<b>Canaux.....</b>	<b>17</b>
<b>Formats .....</b>	<b>17</b>
<b>Contenu .....</b>	<b>18</b>
<b>Durée et effort.....</b>	<b>18</b>
<b>Collaboration .....</b>	<b>18</b>
<b>Plan et calendrier du projet.....</b>	<b>18</b>
<b>Risques et facteurs de réduction.....</b>	<b>19</b>
<b>Analyse des risques.....</b>	<b>19</b>
<b>Hypothèses .....</b>	<b>20</b>
<b>Critères d'acceptation et procédures .....</b>	<b>21</b>
<b>Métriques et KPIs .....</b>	<b>21</b>
<b>Procédure d'acceptation.....</b>	<b>21</b>
<b>Approbations signées .....</b>	<b>22</b>

# Objet de ce document

---

Ce document est une Déclaration de travail d'architecture pour le <<projet XXX>>.

La Déclaration de travail d'architecture définit le périmètre et l'approche qui seront utilisés pour mener à bien un projet d'architecture. La Déclaration de travail d'architecture constitue habituellement le document qui permet de mesurer la réussite de l'exécution du projet d'architecture et peut former la base de l'accord contractuel entre le fournisseur et le consommateur de services d'architecture. En général, toutes les informations de ce document doivent se situer à un haut niveau.

La Déclaration de travail d'architecture peut être documentée sur un wiki ou l'intranet plutôt que par un document texte. Pour faire encore mieux, vous pouvez utiliser un outil sous licence TOGAF pour restituer cette production.

Ce modèle montre les contenus « typiques » d'une Déclaration de travail d'architecture et peut être adapté pour être aligné sur toute adaptation TOGAF implémentée.

## Déclaration de travail d'architecture

---

### Vision architecturale

La déclaration de travail d'architecture vise à refondre la plateforme Foosus en une solution scalable, modulaire et géociblée, capable de soutenir l'innovation rapide et la croissance de l'entreprise.

### Engagements clés

1. Fournir une architecture répondant aux besoins des consommateurs et producteurs locaux.
2. Réduire la dette technique pour accélérer les cycles de développement.
3. Améliorer la performance et la disponibilité des services critiques.

## Requête du projet et contexte

---

Élément	Description
Contexte	Foosus est une start-up dans l'alimentation durable, visant à connecter les consommateurs aux producteurs locaux via une plateforme géociblée.
Problématique	L'architecture actuelle accumule une dette technique importante et manque de cohérence, ce qui freine l'innovation et la scalabilité.
Objectif de la	Concevoir une nouvelle architecture modulaire et évolutive pour soutenir

requête	l'innovation rapide et les besoins croissants de Foosus.
---------	--

## Description du projet et périmètre

Aspect	Description
Objectif du projet	Refonte de l'architecture pour répondre aux besoins business, réduire la dette technique, et améliorer la performance globale.
Périmètre	<ul style="list-style-type: none"> <li>- Services critiques : géolocalisation, gestion des utilisateurs, et mise en relation producteurs-consommateurs.</li> <li>- Intégrations avec des services tiers (paiements, cartographie).</li> <li>- Scalabilité et performance.</li> </ul>
Exclusions	Fonctionnalités non critiques ou secondaires, reportées à des phases ultérieures.

## Vue d'ensemble

Composant	Description
Architecture cible	Une architecture basée sur des microservices, intégrant des standards ouverts pour la modularité et l'interopérabilité.
Technologies prévues	Backend : Node.js, Python ; Frontend : React/Vue.js ; Bases de données : PostgreSQL, MongoDB.
Objectifs fonctionnels	<ul style="list-style-type: none"> <li>- Amélioration de l'expérience utilisateur.</li> <li>- Réduction des temps de réponse et augmentation de la disponibilité.</li> <li>- Gestion optimale des pics de trafic.</li> </ul>
Mécanismes de gouvernance	Utilisation de pipelines CI/CD, outils de monitoring (Prometheus, Grafana), et gestion des configurations (Ansible, Terraform).

## Alignement stratégique

Objectif stratégique	Contribution de l'architecture
Augmentation de l'engagement utilisateur	Amélioration des performances et de la fiabilité des services, assurant une expérience utilisateur optimale.
Réduction de la dette technique	Transition vers une architecture modulaire et standardisée, facilitant la maintenance et l'évolution.
Croissance de l'entreprise	Scalabilité accrue pour gérer les montées en charge et les nouveaux utilisateurs sans redéveloppement important.

# Objectifs et périmètre

## Objectifs

Les objectifs business de ce travail d'architecture sont les suivants :

Objectif Business	Notes	Commentaires
Améliorer l'engagement des utilisateurs	9	C'est une priorité stratégique pour attirer et fidéliser les utilisateurs finaux.
Réduire la dette technique et accélérer l'innovation	8	Essentiel pour garantir la maintenabilité et permettre une livraison rapide des nouvelles fonctionnalités.
Augmenter la scalabilité pour soutenir la croissance	7	Important pour accompagner la croissance à long terme, mais dépend d'autres prérequis techniques.

## Périmètre

Aspect	Inclusions	Exclusions
Services critiques	Développement des microservices pour la géolocalisation, la gestion des utilisateurs, et la mise en relation producteurs-consommateurs.	- Fonctionnalités secondaires ou non critiques (ex. systèmes de reporting avancés).
Technologies	- Adoption d'une architecture microservices. - Intégration des standards ouverts (REST, JSON, OAuth).	Intégration avec des outils non prioritaires ou obsolètes.
Performance et scalabilité	- Optimisation pour gérer $\geq 10\,000$ requêtes/minute. - Réduction des temps de réponse à $\leq 200$ ms.	Extensions nécessitant une refonte majeure en dehors du cadre du projet actuel.
Sécurité	Implémentation de mécanismes conformes RGPD (chiffrement, anonymisation, gestion des droits).	Gestion avancée de la sécurité pour des zones non critiques ou non accessibles publiquement.
Interopérabilité	- Compatibilité avec les services tiers existants (paiements, géolocalisation).	Intégrations futures non essentielles au lancement du projet.

<b>Gestion des données</b>	<ul style="list-style-type: none"> <li>- Sauvegardes automatisées.</li> <li>- Migration des données existantes vers des bases optimisées.</li> </ul>	Archivage ou suppression massive de données historiques non critiques.
----------------------------	--	--

## Parties prenantes, préoccupations, et visions

Le tableau suivant montre les parties prenantes qui utilisent ce document, leurs préoccupations, et la façon dont le travail d’architecture répondra à ces préoccupations par l’expression de plusieurs visions.

Partie pre-nante	Préoccupation	Vision
<b>CEO (Ash Cal-lum)</b>	Assurer l’alignement stratégique de l’architecture avec les objectifs de croissance de l’entreprise.	Une <b>vision stratégique</b> démontrant comment l’architecture cible soutient la mon-tée en charge et l’innovation rapide.
<b>CIO (Natasha Jarson)</b>	Réduire la dette technique et amé-liorer la maintenabilité et la stan-dardisation des pratiques.	Une <b>vision technique</b> détaillant les mé-canismes pour réduire les dépendances obsolètes et favoriser l’adoption de stan-dards.
<b>CPO (Daniel Anthony)</b>	Améliorer l’expérience utilisateur et accélérer le temps de mise sur le marché des nouvelles fonction-nalités.	Une <b>vision produite</b> illustrant l’impact de l’architecture sur la performance, la rapi-dité, et l’expérience utilisateur.
<b>Responsable Ingénierie (Pete Parker)</b>	Mettre en place une infrastructure fiable avec un monitoring robuste et des déploiements continus.	Une <b>vision opérationnelle</b> expliquant les pipelines CI/CD, les outils de monitoring, et les mécanismes de surveillance proactive.
<b>Équipe Pro-duit</b>	Faciliter l’ajout de nouvelles fonc-tionnalités et améliorer la collabo-ration avec l’équipe de développe-ment.	Une <b>vision fonctionnelle</b> décrivant les processus intégrés pour un développe-ment agile et collaboratif.
<b>Équipe de Dé-veloppement</b>	Intégrer de nouveaux standards tout en minimisant l’impact sur la productivité quotidienne.	Une <b>vision de développement</b> clarifiant les standards, les outils, et les pratiques nécessaires pour une transition harmo-nieuse.
<b>Clients finaux</b>	Garantir des performances opti-males, une disponibilité cons-tante, et une précision des résul-tats de géolocalisation.	Une <b>vision utilisateur</b> démontrant com-ment l’architecture répond aux attentes des utilisateurs finaux en termes de rapi-dité, fiabilité, et pertinence.

## Approche managériale

Aspect managérial	Approche adoptée	Description
<b>Gestion des parties</b>	Communication régulière avec	Garantit un alignement constant



<b>prenantes</b>	toutes les parties prenantes via des réunions hebdomadaires et des rapports mensuels.	entre les équipes business, techniques, et produit.
<b>Responsabilité et délégation</b>	Assignment claire des rôles et responsabilités à chaque membre clé de l'équipe d'architecture et de développement.	Assure une prise en charge fluide des tâches et une responsabilisation adaptée.
<b>Suivi des progrès</b>	Utilisation d'outils Agile tels que Kanban pour visualiser les tâches, priorités, et progrès en temps réel.	Permet un suivi précis des étapes et favorise une amélioration continue grâce à la transparence.
<b>Gestion des risques</b>	Identification proactive des risques avec un plan de mitigation associé pour chaque risque identifié.	Minimise les impacts négatifs potentiels sur le calendrier ou la qualité des livrables.
<b>Validation et itérations</b>	Validation fréquente des livrables par les parties prenantes via des revues régulières et des prototypes.	Garantit que les livrables répondent aux attentes avant d'avancer dans les étapes suivantes.
<b>Formation et montée en compétences</b>	Organisation de sessions de formation sur les outils et standards utilisés (ex. microservices, CI/CD).	Facilite l'adoption rapide des nouvelles pratiques et outils par les équipes.
<b>Documentation et traçabilité</b>	Centralisation de la documentation sur une plateforme collaborative (ex. GitHub, Confluence).	Assure une traçabilité complète et un accès simplifié à toutes les informations clés du projet.

## Procédures de changement de périmètre

Étape	Description	Responsable
<b>Identification du changement</b>	Toute demande de modification du périmètre doit être clairement identifiée et documentée.	Équipe concernée (Produit, Technique, ou Business)
<b>Soumission de la demande</b>	La demande doit être soumise via un formulaire standardisé, incluant la justification, l'impact attendu, et les délais associés.	Responsable de la partie prenante concernée (ex. Product Owner, Responsable IT)
<b>Évaluation de l'impact</b>	Analyse de l'impact du changement sur les délais, le budget, et les objectifs existants du projet.	Architecte logiciel et Comité d'architecture
<b>Validation de la demande</b>	La demande est examinée par le Comité d'architecture pour validation ou rejet.	Comité d'architecture (incluant les représentants business et techniques)
<b>Mise à jour du plan</b>	Si approuvé, le plan de travail, le calendrier, et les priorités sont ajustés en conséquence.	Architecte logiciel et Responsable de projet

<b>Communication du changement</b>	Une notification officielle est envoyée à toutes les parties prenantes, incluant les impacts et le nouveau plan.	Chef de projet
<b>Suivi et documentation</b>	Les modifications sont enregistrées dans une base de données des changements pour garantir la traçabilité.	Responsable de la gestion des changements (Change Manager)

# Rôles et responsabilités

## Structure de gouvernance

Rôle	Responsabilités	Lien hiérarchique
<b>CEO (Ash Callum)</b>	Supervision stratégique globale, validation des objectifs business, et arbitrage des priorités majeures.	Responsable de l'alignement global du projet avec la stratégie de Foosus.
<b>CIO (Natasha Jarson)</b>	Pilotage des initiatives architecturales et techniques, gestion de la dette technique, et coordination avec l'équipe IT.	Supervise directement les équipes techniques.
<b>CPO (Daniel Anthony)</b>	Définition des besoins produit, alignement des fonctionnalités avec les attentes des utilisateurs, et validation des livrables.	Collabore avec le CIO et le Responsable Ingénierie.
<b>Responsable Ingénierie (Pete Parker)</b>	Gestion des équipes de développement, implémentation des solutions techniques, et résolution des incidents critiques.	Rend compte au CIO et collabore avec le Comité d'architecture.
<b>Architecte Logiciel</b>	Conception de l'architecture cible, définition des standards techniques, et supervision des aspects d'interopérabilité.	Collabore avec le CIO, le Responsable Ingénierie, et l'équipe de développement.
<b>Product Owner</b>	Priorisation des fonctionnalités, suivi des besoins utilisateurs, et communication avec les parties prenantes.	Interagit avec l'équipe produit et les développeurs.
<b>Équipe de Développement</b>	Implémentation des microservices, intégration des solutions techniques, et résolution des anomalies.	Superviseurs : Responsable Ingénierie et Architecte Logiciel.
<b>Comité d'Architecture</b>	Validation des décisions stratégiques liées à l'architecture, gestion des risques techniques, et suivi de la gouvernance.	Inclus le CIO, l'Architecte Logiciel, et des représentants business et produit.

<b>Responsable DevOps</b>	Gestion des pipelines CI/CD, des outils de monitoring, et des environnements de développement et production.	Rend compte au Responsable Ingénierie.
---------------------------	--	--

**Structure de l'équipe et liens hiérarchiques**

- **CEO** supervise directement le **CIO** et le **CPO**, qui coordonnent respectivement les aspects techniques et business.
- Le **Responsable Ingénierie** et l'**Architecte Logiciel** sont les points focaux pour l'implémentation technique et collaborent étroitement avec les développeurs et les responsables DevOps.
- Le **Comité d'Architecture** assure la validation et le suivi des choix stratégiques, garantissant un alignement avec les objectifs globaux.

**Process du projet**

Process	Description	Responsable
<b>Réunions régulières</b>	Organisation de réunions hebdomadaires avec les parties prenantes pour suivre l'avancement, identifier les blocages, et prioriser les actions.	Chef de projet
<b>Comités de pilotage</b>	Réunions mensuelles du comité d'architecture pour valider les choix stratégiques et superviser les livrables clés.	Comité d'architecture
<b>Répertoire de documents</b>	Centralisation de la documentation (spécifications, livrables, décisions) dans un espace collaboratif (ex. GitHub, Confluence).	Responsable de documentation
<b>Management de la configuration</b>	Utilisation d'outils (ex. Git, Ansible) pour suivre et gérer les modifications des configurations et du code source.	Responsable DevOps
<b>Assurance qualité</b>	Implémentation de tests automatisés (unitaires, intégration, performance) et validation manuelle des livrables.	Équipe de développement et Architecte Logiciel
<b>Procédure en cas d'escalade</b>	Escalade des problèmes critiques vers les responsables appropriés (ex. CIO pour les décisions techniques majeures).	Chef de projet
<b>Procédure en cas de changement</b>	Documentation et évaluation des demandes de changement via un formulaire standardisé, suivi d'une validation par le comité d'architecture.	Comité d'architecture

**Rôles et responsabilités (RACI)**

Activité/Processus	Responsable (R)	Approbateur (A)	Consulté (C)	Informé (I)
Définition de la vision architecturale	Architecte Logiciel	Comité d'Architecture	CIO, Product Owner	CEO, Responsable Ingénierie
Validation des exigences business	Product Owner	CPO	Équipe Produit, Équipe Technique	CIO, Architecte Logiciel
Conception des microservices	Architecte Logiciel	Responsable Ingénierie	Développeurs, Product Owner	CIO, Responsable DevOps
Mise en œuvre des pipelines CI/CD	Responsable DevOps	Responsable Ingénierie	Développeurs, Architecte Logiciel	CIO, Product Owner
Validation des livrables techniques	Équipe de Développement	Comité d'Architecture	Product Owner, CIO	CEO, Équipe Produit
Gestion des incidents critiques	Responsable Ingénierie	CIO	Architecte Logiciel, Responsable DevOps	CEO, Product Owner
Évaluation des demandes de changement	Chef de Projet	Comité d'Architecture	Product Owner, Équipe Technique	CEO, CIO
Communication aux parties prenantes	Chef de Projet	CIO	CEO, Comité d'Architecture	Équipe Produit, Équipe Développement

# Approche architecturale

## Process d'architecture

La méthode de développement d'architecture TOGAF (ou ADM pour « Architecture Development Method ») décrit une méthodologie des meilleures pratiques pour le développement architectural. Néanmoins, toutes les phases ne sont pas également pertinentes pour chaque projet. Le tableau ci-dessous décrit l'utilisation de l'ADM pour ce projet spécifique.

Phase	Entrée/Sortie	Notes
Préliminaire	- Établir les principes architecturaux et le cadre.	Cette phase pose les bases du projet en définissant les attentes globales.

<b>A – Vision de l'architecture</b>	<ul style="list-style-type: none"> <li>- Description initiale de l'état cible.</li> <li>- Validation des parties prenantes.</li> </ul>	Crée une vision stratégique pour aligner les parties prenantes sur les objectifs.
<b>B – Architecture business</b>	<ul style="list-style-type: none"> <li>- Analyse des processus métier.</li> <li>- Modélisation des besoins business.</li> </ul>	Définit l'alignement entre les besoins business et les solutions proposées.
<b>C – Architecture des systèmes d'information</b>	<ul style="list-style-type: none"> <li>- Modélisation des flux de données.</li> <li>- Définition des structures d'information.</li> </ul>	Ligne de base pour les systèmes actuels et état cible pour les améliorations nécessaires.
<b>D – Architecture technologique</b>	<ul style="list-style-type: none"> <li>- Identification des technologies nécessaires.</li> <li>- Définition des composants techniques.</li> </ul>	Permet de sélectionner les technologies pour atteindre l'état cible de manière efficace.
<b>E – Opportunités et solutions</b>	<ul style="list-style-type: none"> <li>- Identification des opportunités d'amélioration.</li> <li>- Définition des priorités des solutions.</li> </ul>	Cette phase planifie les investissements nécessaires pour les livrables.
<b>F – Planning de migration</b>	<ul style="list-style-type: none"> <li>- Création de la feuille de route de migration.</li> <li>- Identification des étapes clés.</li> </ul>	Fournit un plan clair pour la transition vers l'état cible.
<b>G – Gouvernance de l'implémentation</b>	<ul style="list-style-type: none"> <li>- Suivi de l'exécution des projets.</li> <li>- Vérification de la conformité aux standards architecturaux.</li> </ul>	Assure que l'implémentation reste alignée avec la vision et les exigences définies.
<b>H – Management du changement d'architecture</b>	<ul style="list-style-type: none"> <li>- Évaluation continue des changements.</li> <li>- Ajustements nécessaires en fonction des évolutions business.</li> </ul>	Permet de maintenir la pertinence de l'architecture face aux évolutions des besoins.
<b>Management des conditions requises</b>	<ul style="list-style-type: none"> <li>- Suivi des exigences initiales et nouvelles.</li> <li>- Validation des livrables en conformité avec ces exigences.</li> </ul>	Assure que toutes les exigences critiques sont adressées tout au long du projet.

### Notes supplémentaires

- **Itérations**  
Certaines phases, comme la **vision d'architecture (Phase A)** et les **opportunités et solutions (Phase E)**, peuvent nécessiter plusieurs itérations en fonction des retours des parties prenantes.
- **Alignement avec les objectifs**  
Chaque phase est priorisée en fonction de l'impact sur la stratégie de Foosus, notamment l'innovation rapide, la scalabilité, et la réduction de la dette technique.

# Contenu de l'architecture

Le cadre de contenu d'architecture TOGAF (ou ACF pour « Architecture Content Framework ») fournit une catégorisation des meilleures pratiques pour le contenu de l'architecture. Néanmoins, tous les éléments ne sont pas également pertinents pour chaque projet. Le tableau ci-dessous décrit les zones de contenu pertinentes pour ce projet spécifique.

Zone de contenu	Entrée/Sortie	Notes
<b>Principes, Vision, et Conditions requises de l'Architecture</b>	<ul style="list-style-type: none"><li>- Entrée : Objectifs stratégiques et besoins business.</li><li>- Sortie : Vision cible et principes architecturaux définis.</li></ul>	Cette section décrit la direction stratégique et les principes qui guideront toutes les décisions architecturales.
<b>Architecture Business</b>	<ul style="list-style-type: none"><li>- Entrée : Modèle des processus métier actuels.</li><li>- Sortie : Modèle cible des processus métier.</li></ul>	Cette zone de contenu définit comment l'architecture soutient les besoins métier et améliore les flux.
<b>Architecture des systèmes d'information — Données</b>	<ul style="list-style-type: none"><li>- Entrée : Inventaire des données actuelles.</li><li>- Sortie : Modèle de données cible, stratégie de migration des données.</li></ul>	Spécifie comment les données seront structurées, migrées, et protégées pour répondre aux exigences business.
<b>Architecture des systèmes d'information — Applications</b>	<ul style="list-style-type: none"><li>- Entrée : Inventaire des applications existantes.</li><li>- Sortie : Catalogue des applications et interfaces cible.</li></ul>	Identifie les applications nécessaires pour supporter les processus métier et les interactions des utilisateurs.
<b>Architecture technologique</b>	<ul style="list-style-type: none"><li>- Entrée : Inventaire des infrastructures actuelles.</li><li>- Sortie : Description des composants technologiques nécessaires.</li></ul>	Détaille les technologies à utiliser pour supporter l'architecture cible, y compris les outils et frameworks.
<b>Réalisation de l'architecture</b>	<ul style="list-style-type: none"><li>- Entrée : Besoins des parties prenantes.</li><li>- Sortie : Roadmap détaillée et phases de déploiement.</li></ul>	Définit le plan d'action pour mettre en œuvre les éléments de l'architecture, en priorisant les livrables critiques.

Les préoccupations des parties prenantes de Foosus, telles que l'alignement stratégique, la réduction de la dette technique, l'amélioration de l'expérience utilisateur, et la fiabilité des services, sont traduites en visions spécifiques.

Ces visions incluent une architecture scalable, un cadre technique standardisé, une feuille de route produit agile, une infrastructure de monitoring automatisée, et une plateforme optimisée pour les performances et la fiabilité.

Ces éléments garantissent que l'architecture répond aux attentes stratégiques,

opérationnelles, et techniques de l'entrepris

## Méthodologies pertinentes et normes de l'industrie

- TOGAF (The Open Group Architecture Framework)

Utilisé comme cadre principal pour structurer et guider le développement de l'architecture, en s'appuyant sur l'Architecture Development Method (ADM) pour garantir une approche systématique.

- Standards d'interopérabilité

Adoption de normes ouvertes telles que REST, OAuth 2.0, et JSON pour assurer une compatibilité et une intégration fluide.

- Normes de sécurité

Conformité aux recommandations et respect du RGPD pour protéger les données personnelles des utilisateurs et garantir la sécurité de la plateforme.

- Meilleures pratiques Agile

Utilisation des pratiques Kanban et de l'amélioration continue pour aligner le projet avec les besoins changeants de l'entreprise.

- CI/CD :

Mise en œuvre de pipelines CI/CD pour accélérer les cycles de développement tout en minimisant les erreurs humaines.

### Points clés relatifs à l'approche architecturale

Aspect	Description
Niveau de détail	L'architecture cible se situe au niveau stratégique, en abordant des segments clés tels que la géolocalisation et les services de mise en relation.
Période	L'architecture cible couvrira une période de trois ans, avec une roadmap évolutive pour intégrer des améliorations continues.
Sujet	Domaine principal : solutions technologiques pour l'alimentation durable, notamment les services géociblés et les interactions utilisateur.
Niveau d'abstraction	Une architecture concrète focalisée sur des solutions spécifiques, complétée par des références à des architectures de référence abstraites.
Ligne de base vs cible	L'accent est mis sur la documentation de l'architecture cible, avec des étapes claires pour évaluer et migrer depuis la ligne de base actuelle.
Itération	L'approche est itérative, alignée sur l'ADM, permettant des ajustements réguliers en fonction des retours des parties prenantes.
Partitionnement	Le travail d'architecture est partitionné en plusieurs segments pour couvrir les systèmes d'information, l'architecture technologique, et les services métiers.

## Plan de travail

---



## Élément de travail 1 : Définition de la vision architecturale

---

### Activités

- Analyse des besoins business et techniques de Foosus.
- Organisation de réunions avec les parties prenantes clés pour valider la vision stratégique.
- Documentation de la vision cible et des principes directeurs de l'architecture.

### Livrables

Les produits de travail suivant seront créés en résultat de ce travail d'architecture :

- **Vision de l'architecture cible** : Un document décrivant les objectifs stratégiques, la structure de l'architecture cible, et son alignement avec les priorités business.
- **Liste des principes directeurs** : Un ensemble de règles pour guider les décisions architecturales.

## Élément de travail 2 : Analyse et modélisation de l'architecture actuelle

---

### Activités

- Audit des systèmes existants et identification de la dette technique.
- Modélisation de l'architecture actuelle (processus métiers, systèmes d'information, applications, et technologies).
- Évaluation des écarts entre la ligne de base actuelle et l'architecture cible.

### Livrables

- **Document de ligne de base** : Un état des lieux détaillé de l'architecture existante, y compris les points faibles et les opportunités.
- **Rapport d'analyse des écarts** : Un rapport identifiant les zones nécessitant des améliorations pour atteindre l'état cible.

## Élément de travail 3 : Définition de l'architecture cible

---

### Activités



- Conception des microservices pour répondre aux besoins de modularité et de scalabilité.
- Définition des interfaces d'API et des flux de données entre les composants.
- Sélection des technologies et outils pour la mise en œuvre.

## Livrables

- **Architecture cible** : Une documentation technique détaillée incluant les schémas de microservices, les API, et les technologies associées.
- **Catalogue des technologies** : Une liste des outils et technologies recommandés.

## Élément de travail 4 : Planification et mise en œuvre

---

### Activités

- Élaboration d'une roadmap pour la migration de l'architecture.
- Définition des étapes clés, des priorités, et des délais associés.
- Coordination avec les équipes de développement et produit pour le déploiement.

## Livrables

- **Feuille de route** : Une roadmap détaillée avec des phases de livraison définies.
- **Rapport de priorisation** : Une liste des activités classées par priorité et impact.

## Élément de travail 5 : Tests et validation

---

### Activités

- Mise en place de tests d'intégration et de performance pour les nouveaux composants.
- Validation des livrables avec les parties prenantes.
- Documenter les résultats des tests et les ajustements nécessaires.

## Livrables

- **Rapport de validation** : Documentation des tests réalisés et des résultats obtenus.
- **Certification de conformité** : Validation que les livrables répondent aux critères définis.

# Plan de communication

---

## Évènements

Évènement	Description	Fréquence
Réunions hebdomadaires	Suivi de l'avancement, gestion des obstacles, et réajustement des priorités avec les équipes et parties prenantes.	Chaque semaine
Revue de sprint	Présentation des progrès réalisés à la fin de chaque sprint et validation des livrables.	Toutes les deux semaines
Ateliers d'architecture	Résolution des questions d'architecture avec les équipes techniques pour aligner les décisions techniques.	Une fois par mois
Réunions trimestrielles	Évaluation stratégique de l'avancement global, ajustement de la feuille de route avec les dirigeants.	Tous les trimestres

## Canaux

Canal	Utilisation
Slack	Communication quotidienne pour la coordination rapide, partage des notifications importantes.
E-mail	Partage des documents officiels, rapports hebdomadaires, invitations et résumés des décisions.
GitHub	Gestion du code source, suivi des modifications, documentation technique, et suivi des tickets.
Google Meet/Zoom	Utilisé pour les réunions hebdomadaires, ateliers à distance et revues de sprint pour faciliter la collaboration.

## Formats

Format	Description
Présentations PowerPoint	Utilisées pour les revues de sprint, les ateliers et les réunions stratégiques trimestrielles.
Documents partagés (Google Docs)	Rapports d'avancement et comptes rendus de réunion, permettant la contribution et la validation en temps réel.
Tableaux Kanban (Jira, Trello)	Outil visuel pour suivre l'avancement des tâches, les priorités, et les blocages.
Rapports techniques	Détails des décisions architecturales, tests de performance et audits de sécurité.

# Contenu

Type de contenu	Description
Avancement du projet	Mise à jour sur l'état actuel des tâches, les réalisations et les fonctionnalités à venir.
Rapports de performance et KPI	Indicateurs clés de performance (KPI) : temps de réponse, nombre d'utilisateurs, incidents de production.
Décisions d'architecture	Documentation des décisions majeures en architecture, revue des choix techniques et compromis.
Risques et plans d'atténuation	Analyse des risques identifiés et plans d'atténuation pour minimiser les impacts potentiels.

# Durée et effort

Phase/Activité	Durée estimée	Effort requis (heures)	Ressources impliquées
Définition de la vision architecturale	2 semaines	80 heures	2 architectes logiciels, 1 product owner
Analyse de la ligne de base	3 semaines	120 heures	1 architecte logiciel, 1 responsable IT
Conception de l'architecture cible	4 semaines	160 heures	2 architectes logiciels, 1 responsable produit
Planification de la migration	2 semaines	80 heures	1 chef de projet, 1 architecte logiciel
Validation et tests	3 semaines	100 heures	Équipe technique, architectes, et product owner

# Collaboration

# Plan et calendrier du projet

Phase	Durée	Étapes clés
Phase 1 : Conception de l'architecture	Mois 1	<ul style="list-style-type: none"><li>- Validation des besoins business et techniques.</li><li>- Élaboration des spécifications d'architecture et des standards de microservices.</li><li>- Planification des ressources et outils.</li></ul>

<b>Phase 2 : Développement initial des microservices prioritaires</b>	Mois 2 à 4	<ul style="list-style-type: none"> <li>- Développement et tests unitaires des premiers microservices (gestion des utilisateurs, géolocalisation).</li> <li>- Mise en place des pipelines CI/CD pour déploiements automatisés.</li> <li>- Validation des premiers composants par le comité d'architecture.</li> </ul>
<b>Phase 3 : Tests d'intégration et optimisation de la performance</b>	Mois 5 à 6	<ul style="list-style-type: none"> <li>- Intégration des microservices dans un environnement de test complet.</li> <li>- Tests de performance et de sécurité pour garantir résilience et conformité.</li> <li>- Ajustements pour scalabilité et disponibilité.</li> </ul>
<b>Phase 4 : Déploiement en production des microservices initiaux</b>	Mois 7	<ul style="list-style-type: none"> <li>- Déploiement en production des composants validés (ex : gestion des utilisateurs et géolocalisation).</li> <li>- Surveillance post-déploiement.</li> <li>- Optimisation selon les retours utilisateurs.</li> </ul>
<b>Phase 5 : Élargissement des fonctionnalités et réduction de la dette technique</b>	Mois 8 à 10	<ul style="list-style-type: none"> <li>- Développement de nouvelles fonctionnalités (gestion des transactions, offres alimentaires).</li> <li>- Réduction progressive de la dette technique.</li> <li>- Formation continue pour les équipes sur les standards de microservices.</li> </ul>
<b>Phase 6 : Évaluation finale et validation complète de l'architecture</b>	Mois 11	<ul style="list-style-type: none"> <li>- Évaluation complète de l'architecture par rapport aux objectifs initiaux.</li> <li>- Audit final de conformité (sécurité, performance, scalabilité).</li> <li>- Validation et signature finale par le comité d'architecture.</li> </ul>
<b>Phase 7 : Documentation et transfert de propriété</b>	Mois 12	<ul style="list-style-type: none"> <li>- Finalisation de la documentation technique.</li> <li>- Transfert officiel de propriété aux équipes de maintenance.</li> <li>- Clôture du projet et évaluation finale selon les KPIs.</li> </ul>

## Risques et facteurs de réduction

### Analyse des risques

Risque	Gravité	Probabilité	Facteur de réduction	Propriétaire
Retards dans la validation des livrables	Élevée	Moyenne	Organiser des réunions hebdomadaires pour accélérer les validations.	Chef de projet
Manque de compétences sur les nouvelles technologies	Moyenne	Élevée	Organiser des formations sur les microservices et les pipelines CI/CD.	Responsable Ingénierie
Dépendances aux fournisseurs tiers	Élevée	Moyenne	Identifier des solutions alternatives en cas de défaillance des services.	Architecte Logiciel
Manque d'adoption des équipes produit et technique	Moyenne	Faible	Planifier des ateliers collaboratifs pour présenter les bénéfices du projet.	Product Owner
Risques de non-conformité au RGPD	Élevée	Faible	Mettre en place un audit de conformité et une validation juridique préliminaire.	Responsable IT Sécurité

## Hypothèses

Le tableau ci-dessous résume les hypothèses pour cette Déclaration de travail d'architecture :

Hypothèse	Impact	Propriétaire
Les fournisseurs tiers de services géolocalisés resteront disponibles et performants.	Toute indisponibilité pourrait entraîner des interruptions ou une baisse de performance.	Responsable Ingénierie
Les parties prenantes fourniront des validations dans les délais impartis.	Les retards de validation pourraient ralentir l'avancement global du projet.	Chef de Projet
Les utilisateurs finaux adopteront rapidement les nouvelles fonctionnalités.	Une faible adoption pourrait compromettre l'impact business de la nouvelle architecture.	Product Owner
L'équipe technique actuelle possède les compétences nécessaires pour la mise en œuvre initiale.	Des lacunes dans les compétences pourraient allonger les délais et nécessiter une formation.	Responsable IT
Le budget initial est suffisant pour couvrir les phases critiques du projet.	Un dépassement budgétaire pourrait entraîner une réduction de la portée du projet.	CEO (Ash Calum)

# Critères d'acceptation et procédures

## Métriques et KPIs

De plus, les métriques suivantes seront utilisées pour déterminer le succès de ce travail d'architecture :

Métrique	Technique de mesure	Valeur cible	Justification	Notes supplémentaires
Temps de réponse des services	Mesure via des outils de monitoring (ex. Grafana, New Relic).	$\leq 200$ ms	Garantir une expérience utilisateur fluide et rapide.	Applicable uniquement aux services critiques.
Disponibilité de la plateforme	Monitoring des uptimes via des outils comme Pingdom ou CloudWatch.	$\geq 99,9$ %	Réduire les interruptions pour assurer une continuité du service.	Inclut les pics de trafic.
Taux d'adoption des nouvelles fonctionnalités	Analyse des données utilisateur (Google Analytics, Amplitude).	$\geq 85$ % des utilisateurs actifs	Vérifier l'adoption et l'efficacité des fonctionnalités développées.	Suivi sur les trois premiers mois après le déploiement.
Scalabilité de l'architecture	Tests de charge via JMeter ou LoadRunner.	$\geq 10\,000$ requêtes/min	Assurer que la plateforme peut gérer les montées en charge sans dégradation des performances.	Tests réalisés avant chaque mise en production majeure.
Réduction de la dette technique	Analyse comparative des composants hérités et modernisés.	50 % des dépendances obsolètes retirées	Améliorer la maintenabilité et réduire les coûts de développement futurs.	Inclut les bases de code et les intégrations non critiques.

## Procédure d'acceptation

Étape	Nom	Description	Responsable
1.	Présentation des livrables	Les livrables sont présentés lors d'une réunion avec les parties prenantes. Les	Chef de projet

		documents sont partagés à l’avance pour revue.	
2.	Revue des livrables	Les parties prenantes examinent les livrables selon les critères d’acceptation (métriques et KPIs). Les ajustements nécessaires sont listés.	Parties prenantes concernées
3.	Résolution des retours	Les équipes responsables appliquent les corrections et ajustements demandés. Un suivi garantit que toutes les préoccupations sont adressées.	Équipe technique et Chef de projet
4.	Validation finale	Le comité d’architecture effectue une validation finale et approuve les livrables.	Comité d’architecture
5.	Signature et archivage	Une fois validés, les livrables sont approuvés via une signature officielle, puis archivés dans le répertoire centralisé.	Parties prenantes principales

# Approbations signées

Date de signature