

Déclaration de Travail d’Architecture

*Projet : Nouvelle architecture logicielle Client : Foosus*

Table des matières

[Information sur le document 4](#_Toc206612235)

[Objet de ce document 5](#_Toc206612236)

[Déclaration de travail d’architecture 6](#_Toc206612237)

[Requête du projet et contexte 6](#_Toc206612238)

[Description du projet 6](#_Toc206612239)

[Vue d’ensemble 6](#_Toc206612240)

[Alignement stratégique 6](#_Toc206612241)

[Objectifs et périmètre 7](#_Toc206612242)

[Objectifs 7](#_Toc206612243)

[Périmètre 7](#_Toc206612244)

[Parties prenantes, préoccupations, et visions 8](#_Toc206612245)

[Approche managériale 9](#_Toc206612246)

[Procédures de changement de périmètre 9](#_Toc206612247)

[Rôles et responsabilités 10](#_Toc206612248)

[Structure de gouvernance 10](#_Toc206612249)

[Process du projet 10](#_Toc206612250)

[Réunions régulières 10](#_Toc206612251)

[Comités de pilotage 10](#_Toc206612252)

[Répertoire de documents 11](#_Toc206612253)

[Management de la configuration 11](#_Toc206612254)

[Assurance qualité 11](#_Toc206612255)

[Procédure en cas d’escalade 11](#_Toc206612256)

[Procédure en cas de changement 11](#_Toc206612257)

[Rôles et responsabilités (RACI) 12](#_Toc206612258)

[Approche architecturale 13](#_Toc206612259)

[Process d’architecture 13](#_Toc206612260)

[Contenu de l’architecture 15](#_Toc206612261)

[Plan de travail 18](#_Toc206612262)

[Élément de travail 1 : Préparation et vision d’architecture (Phases Préliminaire et A) 18](#_Toc206612263)

[Activités : 18](#_Toc206612264)

[Livrables : 19](#_Toc206612265)

[Élément de travail 2 : Définition des architectures métier, données, applications, et technologie (Phases B, C, D) 19](#_Toc206612266)

[Livrables : 19](#_Toc206612267)

[Élément de travail 3 : Planification de la migration et préparation du prototype (Phases E, F) 20](#_Toc206612268)

[Livrables : 20](#_Toc206612269)

[Produits de travail globaux : 20](#_Toc206612270)

[Plan de communication 21](#_Toc206612271)

[**Plan et calendrier du projet** 22](#_Toc206612272)

[Risques et facteurs de réduction 23](#_Toc206612273)

[Analyse des risques 23](#_Toc206612274)

[Hypothèses 24](#_Toc206612275)

[Critères d’acceptation et procédures 25](#_Toc206612276)

[Métriques et KPIs 25](#_Toc206612277)

[Procédure d’acceptation 25](#_Toc206612278)

[Approbations signées 26](#_Toc206612279)

## **Information sur le document**

|  |  |
| --- | --- |
|  |  |
| *Nom du projet* | Projet FOOSUS |
| *Préparé par :* | Julien GUTTER |
| *N° de version du document :* | 0.1 |
| *Titre :* | *Déclaration de travail d’architecture* |
| *Date de version du document :* | Juillet 2025 |
| *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 |

# **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**

## **Requête du projet et contexte**

L'objectif commercial de Foosus est de soutenir la consommation de produits alimentaires locaux et de mettre en contact les clients avec des producteurs et artisans locaux pour satisfaire tous leurs besoins. Une nouvelle plateforme d'e-commerce est nécessaire afin d'améliorer sa compétitivité par rapport aux grandes entreprises d'e-commerce internationales.

## **Description du projet**

Le but du projet est de mettre en place une nouvelle architecture permettant aux équipes techniques de laisser libre cours à leur créativité et d’expérimenter. Cette architecture devra permettre à Foosus d’atteindre un million d’utilisateurs inscrits, pour cela plusieurs objectifs ont été identifiés et seront décrits dans le document.

## **Vue d’ensemble**

La plateforme actuelle de Foosus a atteint un point au-delà duquel elle ne peut plus soutenir les projets de croissance et d'expansion de l'entreprise. Après plusieurs années de développement, la solution technique complexe actuelle n'évolue plus au rythme de l'activité et risque d'entraver la croissance de l’entreprise. Les études de marché et les analyses commerciales montrent que nos clients souhaitent acheter local et soutiennent les producteurs locaux.

## **Alignement stratégique**

Les concurrents de Foosus n'ont pas ciblé le marché de niche d’acteurs locaux. Foosus veut s’appuyer sur les connaissances acquises ces trois dernières années et créer une plateforme qui mettra en contact des consommateurs avec des producteurs et des artisans locaux dans toutes les catégories de besoins.

# **Objectifs et périmètre**

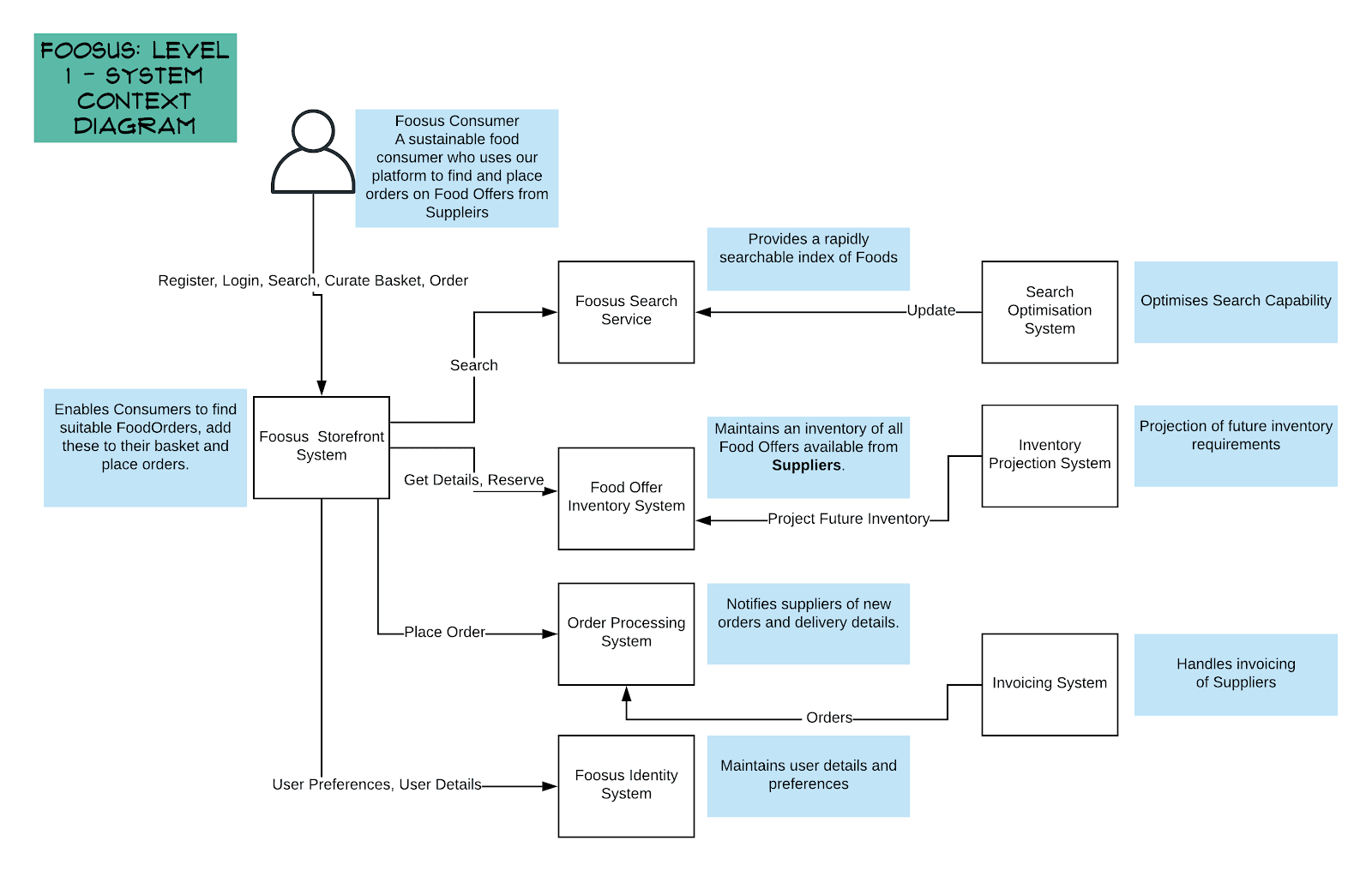
## **Objectifs**

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

|  |
| --- |
| ***Objectif Business*** |
| Tirer parti de la géolocalisation pour relier des fournisseurs et des consommateurs et pour proposer des produits disponibles à proximité des lieux de résidence de ces derniers. |
| Disposer d’une architecture évolutive pour permettre de déployer les services sur diverses régions à travers des villes et des pays donnés. |
| Les améliorations et autres modifications apportées aux systèmes de production devront limiter ou supprimer la nécessité d'interrompre le service pour procéder au déploiement |
| La solution doit être disponible pour nos fournisseurs et nos consommateurs, où qu'ils se trouvent. Cette solution doit être utilisable avec des appareils mobiles et fixes. Elle doit tenir compte des contraintes de bande passante pour les réseaux cellulaires et les connexions Internet haut débit. |
| Différents types d'utilisateurs peuvent être paramétrés (par exemple, fournisseurs, back-office, consommateurs) avec des fonctionnalités et des services spécifiques pour chaque catégorie |

## **Périmètre**

Le périmètre actuel est présenté sur le diagramme de contexte ci-dessous :



## **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 prenante** | **Préoccupation** | **Vision** |
| Ash Callum  CEO | Nombre d’utilisateurs inscrits | Rentabilité |
| Daniel Anthony  CPO | Innovation  Evolutivité  Visibilité | Disponibilité, performance, stabilité, sécurité |
| Christina Orgega  CMO | Visibilité de l’application  Innovation | Satisfaction des clients, performance du système, stabilité du système |
| Jo Kumar  CFO | Nombre d’utilisateurs inscrits  Innovation | Rentabilité, performance, qualité |
| Natasha Jarson  CIO | Faisabilité  Innovation  Flexibilité | Stabilité, fiabilité, performance, disponibilité, sécurité |
| Pete Parker  Engineering Owner | Développement | Adaptabilité, performance |
| Jack Harkner  Operations Lead | Déploiement | Disponibilité, sécurité |

## **Approche managériale**

Un comité d’architecture sera constitué, celui-ci aura pour responsabilité la conformité des implantations vis-à-vis des principes et des décisions relevant de l’architecture d’entreprise.

Il assurera également la gestion du référentiel d’architecture.

Ses interactions se feront dans une culture Agile et Lean.

## **Procédures de changement de périmètre**

En cas de changement de périmètre, le comité d’architecture sera réuni et il procédera à la révision de l’ensemble des pièces du référentiel d’architecture.

# **Rôles et responsabilités**

## **Structure de gouvernance**

L’organigramme ci-dessous présente les liens hiérarchiques ainsi que les fonctions des parties prenantes :

Une image contenant prise

Le contenu généré par l’IA peut être incorrect.

## **Process du projet**

Cette partie décrit comment va se dérouler le processus de définition d’architecture pour les aspects suivants :

### Réunions régulières

Des réunions hebdomadaires de l’équipe d’architecture auront lieu pour suivre l’avancement des livrables, identifier les risques et assigner des actions. Des ateliers techniques bihebdomadaires avec les développeurs clarifieront les exigences.

### Comités de pilotage

Le comité de gouvernance, composé du CEO, CPO, CFO, CMO, CIO et de l’architecte principal, se réunira chaque mois ou sur demande pour valider les livrables et résoudre les escalades.

### Répertoire de documents

Tous les livrables seront stockés sur le dépôt architectural hébergé sur git. Les documents seront versionnés.

### Management de la configuration

Les modèles ArchiMate et les documents seront gérés via Git. Chaque élément de configuration sera versionné, avec un journal des modifications. Des audits mensuels vérifieront la cohérence des versions.

### Assurance qualité

Des revues deux fois par semaine des livrables seront effectuées à l’aide de check-lists basées sur les principes d’architecture. Les évaluations de conformité dans la Phase G identifieront les écarts, qui seront escaladés si nécessaire.

### Procédure en cas d’escalade

Les problèmes non résolus par l’équipe de développement (ex. : non-conformité d’une technologie) seront escaladés au comité de gouvernance. Le comité évaluera les impacts et rendra une décision documentée.

### Procédure en cas de changement

Les nouvelles exigences ou contraintes déclencheront une Demande de Changement. L’architecte principal analysera l’impact, et les changements majeurs seront soumis au comité de gouvernance. Les livrables mis à jour seront déposés sur le répertoire de documents.

## **Rôles et responsabilités (RACI)**

Le tableau ci-dessous présente les responsabilités de chaque partie prenante, R(éalise) A(pprouve) C(onsulté) I(nformé), pour les différentes étapes du processus du projet.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | CEO | CPO | CMO | CFO | CIO | Engineering Owner | Operations Lead | Entreprise Architecture Owner |
| **Réunions** | | | | | | | | |
| Organisation et animation | I | I | I | I | I | A | C | R |
| Rédaction des comptes rendus | I | I | I | I | C | R | C | A |
| C**omité de pilotage** | | | | | | | | |
| Préparation des réunions | I | A | I | I | I | C | C | R |
| Validation des décisions | A | I | I | I | I | C | C | C |
| **Répertoire de document** | | | | | | | | |
| Création et maintenance | C | C | C | C | C | R | C | A |
| Accès et versionnage | I | I | I | I | I | R | I | A |
| **Management de la configuration** | | | | | | | | |
| Identification des éléments de configuration | I | I | I | I | I | R | R | A |
| Gestion des version | I | I | I | I | I | R | R | A |
| **Assurance qualité** | | | | | | | | |
| Revue des livrables | I | I | I | I | I | R | R | A |
| Evaluation de la conformité | C | C | C | C | C | R | I | A |
| **Procédure d’escalade** | | | | | | | | |
| Identification et documentation du problème | I | I | I | I | I | A | C | R |
| Escalade au comité de gouvernance | A | I | I | I | I | C | C | R |
| **Procédure de changement** | | | | | | | | |
| Soumission d’une demande de changement | A | C | C | I | C | C | C | R |
| Analyse et mise à jour des livrables | I | R | R | R | R | R | R | A |
| Validation des changements | A | I | I | I | I | I | I | R |

# **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 | Objectifs commerciaux : géolocalisation, évolutivité  Contraintes : 50 000 USD, 6 mois  Directives : open source, cohérence technologique | Cadre de gouvernance  principes d’architecture  référentiel initial  organisation de l’équipe. | Établissement du comité de gouvernance. Définition des principes (évolutivité, accessibilité, sécurité).  Définition des outils : LeanIX, Jira, ArchiMate |
| A —Vision de l’architecture | Objectifs commerciaux  Besoins des parties prenantes  KPI (10 % d’augmentation des adhésions). | Vision de l’Architecture  Déclaration de travail d’architecture  Carte des parties prenantes. | Définition d’une vision axée sur la géolocalisation et l’évolutivité.  Deux itérations pour valider avec les parties prenantes.  Focus sur une architecture cloud et open source. |
| B —Architecture business | Vision de l’Architecture  Processus actuels  Besoins utilisateurs. | Modèles de processus métier (recherche géolocalisée, gestion des commandes)  Architecture existante  Architecture cible | Analyse des processus actuels (recherche inefficace, 48 % d’abandons)  Définition de processus cibles (matching géolocalisé). Une itération pour valider avec les fournisseurs. |
| C — Architecture des systèmes d’information | Vision de l’Architecture  Exigences fonctionnelles (multi-utilisateurs, multi-appareils)  Inventaire des systèmes existants. | Modèles de données (produits, clients)  Modèles d’applications (recherche, panier)  Architecture existante  Architecture cible | Analyse des bases actuelles et proposition d’une architecture cible.  Deux itérations pour les interfaces UX (fournisseurs, consommateurs). |
| D — Architecture technologique | Vision de l’Architecture  Normes open source  Contraintes de bande passante. | Modèles d’infrastructure (cloud),  Standards technologiques  Architecture existante  Architecture cible | Analyse de l’infrastructure actuelle (serveurs locaux) et définition d’une architecture cloud.  Optimisation pour faible bande passante.  Focus sur la sécurité (ISO 27001). |
| E —Opportunités et solutions | Modèles d’architecture (B, C, D)  Contraintes budgétaires  Directives open source. | Catalogue des solutions  Analyse coûts-avantages  Plan initial de mise en œuvre. | Évaluation des solutions open source vs commerciales. |
| F —Planning de migration | Architectures cibles (B, C, D),  Catalogue des solutions  KPI (délai de parution <1 semaine). | Plan de transformation de l’architecture  Matrice de transition. | Plan de migration pour un déploiement pilote dans une région (par exemple, une grande ville).  Respect du délai de 6 mois.  Coexistence des plateformes existante et nouvelle. |
| G —Gouvernance de l’implémentation | Plan de transformation de l’architecture  Spécifications techniques  Contrats fournisseurs. | Rapports de conformité  Plans d’implémentation validés. | Supervision des équipes de développement pour le prototype.  Revues bihebdomadaires pour vérifier la conformité (géolocalisation, performance). |
| H —Management du changement d’architecture | Rapports de conformité, nouvelles exigences. | Demandes de changement approuvés  Architectures mises à jour. | Gestion des changements réglementaires ou des retours utilisateurs.  Préparation du projet de suivi pour le prototype. Focus sur l’expérimentation. |
| Management des conditions requises | Exigences initiales (géolocalisation, multi-appareils)  retours des parties prenantes  KPI. | Spécification des Besoins d’Architecture mise à jour. | Collecte continue des exigences.  Priorisation des fonctionnalités (recherche géolocalisée, panier).  Intégration des besoins des utilisateurs (fournisseurs, consommateurs, back-office). |

# **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** |
| **Principes, Vision, et Conditions requises de l’Architecture** | Objectifs commerciaux : géolocalisation, évolutivité  Contraintes : budget, délai  Directives : open source | Principes d’architecture  Vision de l’Architecture  Spécification des Besoins d’Architecture | Sous-catégories : Principes (évolutivité, accessibilité, sécurité, coût-efficacité), vision (plateforme e-commerce géolocalisée), exigences (recherche géolocalisée, multi-appareils). Les principes guident la sélection de solutions open source et microservices. |
| **Architecture Business** | Vision de l’Architecture  Processus actuels (recherche, commande)  besoins des utilisateurs (fournisseurs, consommateurs). | Modèles de processus (recherche géolocalisée, gestion des commandes)  diagrammes de flux  matrice des rôles. | Sous-catégories :  Processus métier (inscription fournisseur, recherche géolocalisée, panier),  Flux de valeur (client-fournisseur).  Focus sur la réduction des abandons (48 % actuellement). |
| **Architecture des systèmes d’information — Données** | Exigences de données (produits, clients, géolocalisation). | Modèles de données (conceptuel, logique, physique)  Schémas de base de données. | Sous-catégories :  Modèle de données pour produits (localisation, disponibilité),  Données géolocalisées (distances).  Base cloud envisagée. |
| **Architecture des systèmes d’information — Applications** | Exigences fonctionnelles (interfaces fournisseurs, consommateurs, back-office)  Besoins UX/CX. | Modèles d’applications  Diagrammes d’interfaces API  Matrice d’intégration | Sous-catégories :  Modules applicatifs (recherche géolocalisée, panier, back-office),  Focus sur l’UX (recherche priorisant la proximité). |
| **Architecture technologique** | Normes open source  Contraintes de bande passante  Besoins d’évolutivité. | Modèles d’infrastructure  Diagrammes de déploiement  Standards technologiques. | Sous-catégories :  Infrastructure cloud, microservices, optimisation pour faible bande passante, sécurité.  Cohérence de la pile technologique. |
| **Réalisation de l’architecture** | Architectures cibles (B, C, D)  Plan de migration  KPI (délai de parution, incidents) | Plan de transformation de l’architecture  Plans d’implémentation  Rapports de conformité | Sous-catégories : Plan de migration (déploiement pilote), matrice de transition, rapports de conformité pour le prototype. Suivi via revues bihebdomadaires pour minimiser les incidents. |

**Méthodologies pertinentes et normes de l’industrie**

La conception de l’architecture de la plateforme d’e-commerce Foosus s’appuie sur des méthodologies et normes de l’industrie reconnues pour répondre aux objectifs commerciaux (géolocalisation, évolutivité, accessibilité multi-appareils) et aux contraintes du projet (budget de 50 000 USD, délai de 6 mois). Cette section détaille les approches méthodologiques et les standards qui guideront le développement de l’architecture, en respectant les directives du projet (solutions open source, cohérence technologique, support continu) et en visant les KPI (augmentation de 10 % des adhésions, réduction des incidents P1 à <1/mois).

Méthodologies pertinentes

* TOGAF ADM : Le cycle ADM de TOGAF est la méthodologie principale pour structurer le développement de l’architecture. Une approche semi-agile est adoptée, avec des itérations dans les phases A à D (Vision, Architectures métier, données, applications, technologie) pour intégrer les retours des parties prenantes et tester les fonctionnalités clés, comme la recherche géolocalisée. Les phases E et F se concentreront sur un déploiement pilote dans une région cible, avec une coexistence des plateformes existante et nouvelle pour minimiser les interruptions.
* Approche lean : Une approche lean est intégrée pour préserver l’autonomie des équipes techniques et produits, tout en accélérant les cycles de livraison (objectif : délai de parution <1 semaine). Cela inclut des expérimentations, comme des tests A/B pour optimiser la recherche géolocalisée, et une priorisation des fonctionnalités critiques.
* Gestion de projet agile : La méthode Scrum sera utilisée pour coordonner les équipes, avec des sprints de deux semaines pour les revues UX et les validations techniques.
* Gestion des exigences : Un processus continu de collecte et de priorisation des exigences sera mis en place, avec des ateliers réguliers impliquant les parties prenantes pour aligner les besoins (par exemple, interfaces spécifiques pour fournisseurs et back-office).

Normes de l’industrie

* REST et OpenAPI : Les API REST seront utilisées pour l’interopérabilité entre les modules (recherche géolocalisée, panier, back-office) et les systèmes externes (par exemple, prestataires de paiement tiers). Les spécifications OpenAPI garantiront une documentation standardisée.
* NoSQL (MongoDB, ElasticSearch) : Une base de données open source sera adoptée pour gérer les données des produits, clients, et localisations, avec des schémas optimisés pour la géolocalisation (par exemple, indexation spatiale).
* Microservices : Une architecture de microservices sera utilisée pour permettre des déploiements continus sans interruption, réduisant les incidents P1 et facilitant l’expérimentation (tests A/B).

Directives spécifiques au projet

* Solutions open source : Les technologies open source comme PostgreSQL (base de données), Node.js (back-end), React (front-end), et Keycloak (authentification) seront privilégiées pour minimiser les coûts et respecter le budget de 50 000 USD. Les solutions seront évaluées en fonction de leur support continu (communautés actives, documentation robuste).
* Cohérence de la pile technologique : Une pile technologique cohérente (par exemple, Node.js, PostgreSQL, React) sera adoptée pour réduire les coûts de maintenance et faciliter l’intégration des équipes.
* Architecture cloud : Une infrastructure cloud (AWS ou Azure) sera utilisée pour garantir l’évolutivité multi-régions et la résilience face aux pics de trafic (objectif : supporter les campagnes marketing).
* Déploiement continu : L’architecture de microservices permettra des mises à jour fréquentes et transparentes, réduisant le délai de parution à moins d’une semaine et minimisant les interruptions de service (objectif : plateforme active 24/7).

# **Plan de travail**

Le plan de travail décrit les activités et livrables nécessaires pour concevoir l’architecture de la plateforme d’e-commerce Foosus dans un délai de 6 mois, avec un budget de 50 000 USD. Le travail est structuré en éléments alignés sur les phases ADM de TOGAF, avec une approche semi-agile pour intégrer les retours des parties prenantes et répondre aux KPI.

Le tableau ci-dessous récapitule les grandes étapes du projet :

|  |  |  |
| --- | --- | --- |
| **N°** | **Phase du cycle ADM** | **Description** |
| 1 | Préliminaire et Phase A | Définir à quels besoins doit répondre la nouvelle architecture et mettre en place le cadre qui permettra de réaliser les phases suivantes |
| 2 | Phase B, C, D | Définition précise des modèles de données, applicatifs et quelles sont les technologies retenues |
| 3 | Phase E et F | Définir les étapes de transition pour passer de l’architecture actuelle à l’architecture cible |

## Élément de travail 1 : Préparation et vision d’architecture (Phases Préliminaire et A)

### Activités :

* Mettre en place le cadre de gouvernance (comité de gouvernance, rôles des parties prenantes).
* Identifier les principes d’architecture (évolutivité, accessibilité, coût-efficacité).
* Organiser des ateliers avec les parties prenantes pour définir les besoins.
* Rédiger et valider la vision d’architecture avec le comité de gouvernance.
* Configurer le référentiel d’architecture.

### Livrables :

* Cadre de gouvernance : Document définissant le comité de gouvernance, les processus d’escalade, et les rôles (RACI).
* Principes d’architecture : Liste des principes.
* Vision de l’Architecture : Document décrivant les objectifs (géolocalisation, multi-appareils) et la portée.
* Déclaration de travail d’Architecture : présent document.

## Élément de travail 2 : Définition des architectures métier, données, applications, et technologie (Phases B, C, D)

Activités :

* Analyser les processus métier actuels (recherche, commande) et définir les processus cibles (matching géolocalisé, panier optimisé).
* Modéliser les architectures métier, données (produits, clients), applications, et technologie (infrastructure cloud) avec ArchiMate.
* Effectuer une itération pour tester les wireframes UX de la recherche géolocalisée avec les équipes produit.
* Évaluer les solutions open source (par exemple, MongoDB, Node.js, React) et cloud (AWS, Azure) pour la cohérence technologique.
* Vérifier la conformité ISO 27001 (sécurité).

### Livrables :

* Document de Définition d’Architecture : Modèles ArchiMate pour les architectures métier (processus de recherche), données (schémas clients/produits), application, et technologie (infrastructure cloud).
* Spécification des Besoins d’Architecture : Exigences fonctionnelles (géolocalisation, multi-utilisateurs) et non fonctionnelles (performance, sécurité).
* Rapport d’analyse des solutions : Comparaison des options open source vs commerciales.

## Élément de travail 3 : Planification de la migration et préparation du prototype (Phases E, F)

Activités :

* Identifier les opportunités et solutions (par exemple, microservices pour déploiements continus).
* Élaborer un plan de migration pour un déploiement pilote dans une région cible.
* Définir une matrice de transition pour la coexistence des plateformes existante et nouvelle.
* Préparer les spécifications pour le projet de suivi (prototype).
* Valider le plan avec le comité de gouvernance.

### Livrables :

* Plan de transformation de l’architecture : Plan stratégique pour le déploiement pilote et l’évolutivité multi-régions.
* Plan de migration : Étapes détaillées pour le pilote, incluant la coexistence des plateformes.
* Matrice de transition : Description des étapes de migration (par exemple, migration des données clients, déploiement des modules de recherche).

## Produits de travail globaux :

* Vision de l’Architecture : Document définissant la portée et les objectifs de la plateforme (géolocalisation, évolutivité, accessibilité).
* Document de Définition d’Architecture: Modèles détaillés des architectures métier, données, applications, et technologie, intégrant les principes open source.
* Spécification des Besoins d’Architecture : Liste des exigences fonctionnelles (recherche géolocalisée, panier) et non fonctionnelles (performance, sécurité).
* Plan de transformation de l’architecture : Plan stratégique pour le déploiement pilote et la scalabilité future.
* Plan de migration : Document pour guider le projet de suivi (prototype) dans une région cible.

## **Plan de communication**

Le plan de communication vise à assurer une collaboration fluide entre les parties prenantes pour aligner le développement de l’architecture sur les objectifs commerciaux et les KPI de Foosus. Les communications seront régulières, transparentes, et adaptées aux besoins des différentes audiences.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Événement** | **Canaux** | **Formats** | **Contenu** | **Durée et effort** | **Collaboration** |
| Réunions hebdomadaires de l’équipe d’architecture | Microsoft Teams | Présentation PowerPoint, tableau de bord Jira | Progrès des livrables (par exemple, modèles ArchiMate), suivi des KPI (adhésions, incidents), identification des risques | 1 heure/semaine, 10 participants | Architecte principal (animation), équipe d’architecture (contributeurs), chef de projet (validation), équipes produits (feedback) |
| Ateliers UX avec parties prenantes | Présentiel, Microsoft Teams, Miro | Wireframes, prototypes Figma | Validation des interfaces (recherche géolocalisée, panier), retours UX/CX | 2 heures/2x par semaine, 12 participants | Équipes produits (animation), fournisseurs/consommateurs (feedback), architecte principal (validation) |
| Réunions trimestrielles du comité de gouvernance | Microsoft Teams ou présentiel | Rapport PDF, présentation PowerPoint | Validation des livrables (Vision de l’Architecture, Roadmap), décisions sur escalades, suivi des KPI | 2 heures/trimestre, 8 participants | Comité de gouvernance (décisions), DSI (validation), architecte principal (présentation) |
| Rapports de progrès mensuels | E-mails, SharePoint | Rapport PDF, tableau de bord Grafana | Résumé des progrès, métriques de performance (temps de réponse, taux d’erreurs), avancement des KPI | 30 minutes/mois (rédaction), diffusion aux parties prenantes | Architecte principal (rédaction), DSI et équipes produits (lecteurs) |
| Revues de conformité | Présentiel,  Microsoft Teams | Rapport PDF | Conformité de livrables à la vision d’architecture | 1 heure/mois, 6 participants | Équipe conformité (animation), architecte principal (validation), équipes techniques (implémentation) |
| Communication externe | Réseaux sociaux | Posts | Nouvelles fonctionnalités définies | 1 heure/mois, 6 participants | Equipe marketing (animation), Équipe produit (réalisation), architecte principal (validation), |

Collaboration :

* Les réunions hebdomadaires impliqueront l’équipe d’architecture et les équipes produits pour garantir l’alignement sur les exigences (par exemple, recherche géolocalisée).
* Les ateliers UX incluront des représentants des fournisseurs et consommateurs pour valider les interfaces.
* Le comité de gouvernance validera les livrables clés et résoudra les escalades.
* Les rapports mensuels seront partagés via SharePoint pour assurer la transparence et la traçabilité.

### **Plan et calendrier du projet**

Le calendrier suivant détaille les activités et livrables du projet Foosus sur 6 mois alignés sur les phases ADM de TOGAF. Les jalons clés garantissent le respect du délai et du budget, avec des itérations pour intégrer les retours des parties prenantes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Phase ADM** | **Activités** | **Livrables** | **Durée** | **Dépendances** |
| Préliminaire | Mettre en place la gouvernance, définir les principes, configurer le référentiel | Cadre de gouvernance, principes d’architecture | 2 semaines | Aucune |
| A — Vision | Ateliers avec parties prenantes, rédaction et validation de la vision | Vision de l’Architecture, Statement of Architecture Work | 4 semaines | Phase Préliminaire |
| B — Architecture business | Analyse des processus actuels, modélisation des processus cibles, itération UX | Modèles de processus (recherche géolocalisée, panier) | 4 semaines | Phase A |
| C — Architecture des systèmes d’information | Modélisation des données (clients, produits) et applications (modules CRM), itération UX | Modèles de données et d’applications, Spécification des Besoins d’Architecture | 4 semaines | Phase B |
| D — Architecture technologique | Analyse de l’infrastructure actuelle, définition de l’architecture cloud, sélection des solutions open source | Modèles technologiques (cloud, microservices) | 4 semaines | Phase C |
| E — Opportunités et solutions | Évaluation des solutions open source vs commerciales, analyse coûts-avantages | Catalogue des solutions, plan initial | 2 semaines | Phases B, C, D |
| F — Planning de migration | Élaboration du plan de migration, validation du déploiement pilote | Plan de transformation de l’architecture, plan de migration | 4 semaines | Phase E |
| Management des exigences | Collecte et priorisation continues des exigences via Jira | Spécification des Besoins d’Architecture mise à jour | 6 mois | Toutes les phases |

# **Risques et facteurs de réduction**

## **Analyse des risques**

Les risques suivants ont été recensés :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Risque** | **Gravité** | **Probabilité** | **Facteur de réduction** | **Propriétaire** |
| 1. | Dépassement du budget (50 000 USD) en raison de difficultés imprévues dans la définition de certains livrables | Élevée | Moyenne | - Prioriser les besoins  - Réaliser une analyse coûts-avantages dans la phase E pour comparer les solutions open source (par exemple, PostgreSQL, Node.js) et cloud (AWS vs Azure).  - Prioriser les composants avec un support communautaire actif pour éviter les coûts de maintenance.  - Suivre les dépenses via un tableau de bord mensuel. | Architecte principal |
| 2. | Non-respect du délai de 6 mois en raison de retards dans les itérations UX ou la validation des parties prenantes | Élevée | Moyenne | - Adopter une approche semi-agile avec des sprints de 2 semaines pour les phases A à D.  - Organiser des ateliers UX bihebdomadaires avec les fournisseurs et consommateurs pour accélérer les retours.  - Valider les livrables clés (Vision de l’Architecture, Roadmap) lors des réunions trimestrielles du comité de gouvernance. | Chef de projet |
| 3. | Performances insuffisantes de la recherche géolocalisée sur les connexions à faible bande passante | Moyenne | Moyenne | - Tester les API de géolocalisation (par exemple, OpenStreetMap) dans des conditions de faible bande passante lors de la phase C.  - Optimiser les interfaces React pour une charge rapide sur mobile.  - Utiliser Prometheus/Grafana pour surveiller les performances. | Équipe d’architecture |
| 4. | Difficultés d’intégration des équipes en raison d’une pile technologique non cohérente | Moyenne | Faible | - Définir une pile technologique cohérente (Node.js, PostgreSQL, React) dans la phase D.  - Documenter les standards (REST, OpenAPI) dans l’Architecture Definition Document.  - Former les équipes techniques sur les outils open source choisis. | Architecte principal |

## 

## **Hypothèses**

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

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Hypothèse** | **Impact** | **Propriétaire** |
| 1 | La plateforme existante peut être maintenue en mode maintenance pendant la conception de la nouvelle architecture, sans développement de nouvelles fonctionnalités. | Si fausse, des ressources supplémentaires seront nécessaires pour maintenir la plateforme existante, dépassant le budget de 50 000 USD. | CIO |
| 2 | La coexistence des plateformes existante et nouvelle est réalisable pendant le déploiement pilote, avec une migration progressive des utilisateurs. | Si fausse, des interruptions de service pourraient survenir, augmentant les incidents P1 et nuisant aux KPI (adhésions, abandons). | Architecte principal |
| 3 | Les solutions open source (par exemple, Node.js, PostgreSQL, React) offrent un support communautaire suffisant pour garantir la stabilité et l’évolutivité. | Si fausse, des coûts de maintenance imprévus pourraient compromettre le budget et la cohérence technologique. | Équipe d’architecture |
| 4 | La géolocalisation, modélisée tôt dans l’architecture (phase C), permettra des innovations futures (par exemple, recommandations basées sur la localisation). | Si fausse, la plateforme pourrait ne pas répondre aux attentes des utilisateurs pour la recherche géolocalisée, augmentant les abandons (48 % actuellement). | Équipe d’architecture |
| 5 | Les équipes techniques et produits resteront autonomes grâce à une approche lean et des outils agiles. | Si fausse, des retards dans les itérations UX ou les validations pourraient compromettre le délai de 6 mois. | CIO |

# **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** |
| Nombre d'adhésions d'utilisateurs par jour | Comptage du nombre d’utilisateurs dans la base de données | + 10% / vs actuel | Mesure le développement commercial de la plateforme auprès des clients |
| Adhésion de producteurs alimentaires | Comptage du nombre d’utilisateurs catégorisés en producteurs dans la base de données | 4/mois | Mesure le développement commercial de la plateforme auprès des fournisseurs |
| Délai moyen de parution | Comptage du temps passé entre le démarrage du sprint et le déploiement | <1 semaine | Mesure l’adaptabilité de la nouvelle architecture |
| Taux d'incidents de production P1 | Comptage du nombre de tickets de support technique | < 1/mois | Mesure la fiabilité améliorée de la nouvelle architecture |

## **Procédure d’acceptation**

Lors du déroulement du cycle ADM, les KPI seront rebalayés afin de s’assurer que les solutions proposées restent alignées avec l’atteinte des KPI définis précédemment.

# **Approbations signées**

|  |  |  |
| --- | --- | --- |
| **Nom - Fonction** | **Date de signature** | **Signature** |
| Ash Callum - CEO |  |  |
| Daniel Anthony - CPO |  |  |
| Christina Orgega - CMO |  |  |
| Jo Kumar - CFO |  |  |
| Natasha Jarson - CIO |  |  |
| Pete Parker - Engineering Owner |  |  |
| Jack Harkner - Operations Lead |  |  |
| Ash Callum - CEO |  |  |