

# CONTRAT DE CONCEPTION ET DE DEVELOPPEMENT DE L'ARCHITECTURE

---



**Projet** : Foosus géoconscient

**Client** : Foosus

# Information sur le document

---

<b>Nom du projet</b>	Foosus géoconscient (titre provisoire)
<b>Préparé par</b>	Noémie BARRAL
<b>N° de version</b>	1.0
<b>Titre</b>	Contrat de Conception et de Développement de l'Architecture
<b>Date de version</b>	14 / 11 / 2022
<b>Revu par</b>	N/A
<b>Date de révision</b>	N/A
<b>Historique de version</b>	v. 1.0 (14 / 11 / 2022)

# Table des matières

---

Information sur le document.....	2
Table des matières.....	3
Objet de ce document .....	5
Introduction et Contexte .....	6
La Nature de l'accord .....	6
Objectifs et périmètre .....	7
Objectifs .....	7
Périmètre .....	7
Parties prenantes, préoccupations et visions .....	8
Description de l'architecture.....	10
Description .....	10
Principes stratégiques .....	12
Références aux conditions requises pour l'architecture .....	12
Livrables architecturaux .....	14
Développement de l'architecture.....	14
Mesures de l'architecture cible .....	14
Livraison de l'architecture et métriques business .....	14
Phases de livraison définies .....	14
Plan de travail commun priorisé.....	18
Plan d'architecture .....	18
1. Phase préliminaire.....	18
2. Phase A – Vision de l'architecture .....	18
3. Phase B – Architecture Business.....	19
4. Phase C – Architecture des systèmes d'information.....	20
5. Phase D – Architecture technologique .....	22
6. Phase E – Opportunités et solutions .....	23
7. Phase F – Planning de Migration.....	24
8. Phase G – Gouvernance de l'implémentation .....	25
9. Phase H – Management du changement d'architecture .....	26
10. Phase N – Management des conditions requises .....	27
Plan de communication .....	28
1. Création ou mise à jour d'un document ou d'un artefact.....	28
2. Réunion du groupe projet.....	28
3. Réunion des sponsors .....	28
Risques et facteurs de réduction .....	29

Structure de gouvernance .....	29
Analyse des risques .....	29
Hypothèses.....	29
Critères d'acceptation et procédures .....	31
Métriques et KPI de l'état cible de l'architecture .....	31
Métriques de livraison de l'architecture et du business .....	31
Procédure d'acceptation.....	31
Procédures de changement de périmètre.....	32
Conditions requises pour la conformité.....	32
Développement et propriété de l'architecture .....	32
Calendrier .....	34
Phases de livrables définies.....	35
Approbations signées.....	36

# 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 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 entreprise récente dans le secteur de l'alimentation durable dont l'objectif principal est de soutenir la consommation de produits alimentaires locaux. Pour cela, l'entreprise a déployé une solution qui permet notamment de mettre en contact les clients avec des producteurs et artisans locaux.

Les études de marché et les analyses commerciales montrent en effet que les clients souhaitent acheter local et soutenir les producteurs locaux, confortant Foosus dans sa stratégie commerciale. L'entreprise cherche à améliorer sa solution en proposant une plus grande variété de produit, dans un environnement géographique en extension, avec notamment la possibilité de se baser sur la géolocalisation de l'utilisateur pour qu'il puisse à tout endroit accéder aux offres proches de lui.

La plateforme actuelle a cependant atteint un point au-delà duquel elle ne peut plus soutenir les projets de croissance et d'expansion de l'entreprise. En effet, la solution technique n'évolue plus au rythme de l'activité et risque à terme d'entraver la croissance de l'entreprise.

Foosus souhaite donc créer une nouvelle plateforme de e-commerce pour soutenir sa compétitivité face aux grandes entreprises d'e-commerce internationales et sa capacité d'expansion.

## La Nature de l'accord

---

Ce contrat vise à spécifier les engagements et les attentes que doivent respecter la conception et le développement de la nouvelle plateforme de Foosus. Il s'adresse notamment à l'ensemble des équipes informatiques impliquées sur le projet.

# Objectifs et périmètre

## Objectifs

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

Objectif business	Notes
La solution doit pouvoir évoluer avec le nombre d'utilisateurs (mise à l'échelle)	<ul style="list-style-type: none"><li>• Absorber les pics d'utilisation</li><li>• Permettre la croissance du nombre d'utilisateurs</li><li>• Rester accessible même en cas de surcharge (mode dégradé)</li></ul>
La solution doit être sécurisée en tout lieu et toute circonstance	<ul style="list-style-type: none"><li>• S'adapter aux particularités locales (ex : RGPD)</li><li>• Garantir la sécurité en cas d'élargissement de la plateforme</li></ul>
La solution doit être accessible en tout lieu et tout moment	<ul style="list-style-type: none"><li>• Supprimer la nécessité d'interrompre le service en cas de déploiement</li><li>• Garantir l'accès à la plateforme aux connexions lentes</li></ul>
La solution doit être fonctionnellement évolutive	<ul style="list-style-type: none"><li>• Garantir la capacité à intégrer de nouvelles fonctionnalités</li><li>• Fonctionner en mode agile (cycles courts et retours utilisateurs rapides)</li></ul>
La solution doit favoriser l'innovation	<ul style="list-style-type: none"><li>• Mettre en place des "environnements d'expérimentation"</li></ul>

## Périmètre

Le périmètre du projet peut s'observer sous différents angles.

D'une part, le périmètre métier va s'étendre sur l'ensemble des process de l'activité d'e-commerce de Foosus, à commencer par l'inventaire des produits jusqu'à leur vente et leur facturation.

Ensuite, le périmètre utilisateurs va comprendre tous les acteurs intervenant sur les process énoncés ci-dessus, notamment les consommateurs, les fournisseurs ou encore les back office. Certains services pourront n'être déployés que pour certains utilisateurs spécifiques.

Il convient aussi d'observer le périmètre géographique du projet. Foosus ayant une ambition internationale avec une stratégie d'expansion forte, la solution doit pouvoir être utilisée quel que soit la position géographique de l'utilisateur.

Enfin, le périmètre technique sera influencé par la nécessité d'avoir une solution compatible au plus grand nombre de supports, mobiles et fixes, tenant compte des contraintes de bande passante pouvant aller d'un réseau cellulaire limité à une connexion internet très haut débit.

## 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
<b>Ash CALLUM</b> Chief Executive Officer (CEO)	La nouvelle solution répond aux enjeux stratégiques	Approbation de la vision stratégique au niveau du projet
<b>Natasha JARSON</b> Chief Information Officer (CIO)	Les choix technologiques de développement et de déploiement répondent aux enjeux stratégiques	Approbation de la vision technologique et de la vision infrastructure en accord avec la vision stratégique
<b>Pete PARKER</b> Engineering Owner	Les choix technologiques de développement répondent aux enjeux technologiques et business	Application et proposition de la vision technologique en accord avec la vision business
Frontend Squad	Le développement du frontend répond aux enjeux technologiques et business	Application de la vision technologique selon la vision business
Ruby Development Squad	Le développement Ruby répond aux enjeux technologiques et business	Application de la vision technologique selon la vision business
Identity Management Squad	Le développement de la stratégie d'autorisation répond aux enjeux technologiques et business	Application de la vision technologique selon la vision business
Backend Application Squad	Le développement du backend répond aux enjeux technologiques et business	Application de la vision technologique selon la vision business
Enterprise Architecte Owner	Les choix d'architecture répondent aux enjeux business et technologiques	Définition de l'architecture business, technologique et infrastructure selon la vision correspondante
<b>Jack HARKNER</b> Operations Lead	Les choix technologiques de déploiement répondent aux enjeux technologiques et infrastructurels	Application et proposition de la vision infrastructure en accord avec la vision technologique
Ops Squad	Le déploiement répond aux enjeux infrastructurels	Application de la vision infrastructure
<b>Daniel ANTHONY</b> Chief Product Officer (CPO)	Les choix business répondent aux enjeux stratégiques de l'entreprise	Approbation de la vision business en accord avec la vision stratégique
Product Managers x4	L'évolution du produit répond aux enjeux business	Définition de la vision business



Customer Experience Team	L'évolution du produit répond aux enjeux des clients et impact l'expérience client	Participation à la vision business selon les attentes clients et application de la vision business
Data Analytics Team	L'évolution du produit répond aux enjeux des clients	Participation à la vision business selon les attentes clients
Customer Fulfillement Team	L'évolution du produit impacte l'expérience client	Application de la vision business
<b>Christina ORGEA</b> Chief Marketing Officer (CMO)	Les choix stratégiques business répondent à la stratégie marketing et au positionnement de l'entreprise	Approbation de la vision stratégique au niveau du positionnement stratégique
<b>Jo KUMAR</b> Chief Financial Officer (CFO)	La nouvelle solution respecte la stratégie financière	Approbation de la vision stratégique au niveau des coûts et ROI du projet
Finance Team	L'application de la solution respecte la stratégie financière	Application de la vision stratégique dans le contrôle financier

# Description de l'architecture

---

## Description

### Standardisation technologique

Java étant la compétence clé des équipes et pouvant parfaitement répondre aux exigences de l'application, il sera privilégié sur la partie back-end de notre solution.

Les autres choix technologiques seront à définir à l'aide de la définition d'architecture mais devront autant que possible faire partie d'une même pile technologique, autant pour les technologies de développement que pour celles d'infrastructure.

### Mise à l'échelle

Kubernetes est une solution de gestion de conteneur qui nous permet de mettre en place un scaling automatique de nos conteneurs, les multipliant lorsque nécessaire et les détruisant lorsqu'ils ne sont plus utilisés, permettant ainsi d'absorber nos pics utilisateurs.

### Disponibilité géographique

Kubernetes en plus de répliquer les conteneurs pour mieux gérer la charge, peut répartir la charge à travers différents serveurs situés à travers le monde.

### Interruptions de service

L'autoscaling apporté par Kubernetes peut aussi se faire de manière à supporter temporairement les deux versions en cours de l'application lors d'un déploiement, évitant ainsi toute interruption. Les nouvelles demandes seront progressivement envoyées vers des conteneurs de la nouvelle version tandis que les conteneurs de l'ancienne version seront fermés progressivement lorsque les utilisateurs termineront leur en-cours.

### Rollback

Dans le but d'assurer un historique lisible, il est conseillé de fonctionner avec une branche principale de référence (one trunk base) sur laquelle on viendra merge en fast forward sans commit de merge (ou rebase) afin d'éviter les commits superflus, les historiques illisibles et la duplication de commit. Cette méthode permettra aussi de limiter la gestion des conflits lors des étapes de merge. Dans le but d'assurer des merges request lisibles et une séparation de concern cohérente, il conviendra de bien effectuer une merge request par feature ou bugfix. Les versions seront identifiées par le biais d'un tag permettant de les retrouver facilement.

Cette lisibilité, associée à une démarche d'infra-as-code, permettant de versionner la configuration de l'infrastructure, et à une CI/CD automatisé permettra d'atteindre l'objectif de pouvoir rollback facilement en cas de livraison défectueuse.

### Environnements d'expérimentation

#### **Environnement de développement**

L'environnement de développement va permettre de déployer les modifications en cours avec l'ensemble des composants de l'application afin de tester les modifications de manière indépendante les unes des autres. Chaque merge request dispose donc de son propre environnement de développement.

## **Environnement d'intégration**

L'environnement d'intégration va permettre de déployer la branche principale afin de tester l'intégration des features et des bugfix ensembles.

## **Environnement de QA**

L'environnement de QA permet le déploiement des releases candidates. Ces dernières seront identifiées sur la branche principale par le biais de tag avec une nomenclature spécifique propre aux release candidate. Cet environnement permet d'effectuer la phase de recette et de soumettre les services au tests d'acceptance avant le passage en production.

## **Environnement de production**

Une fois la release candidate validée, un tag de release est ajouté sur la branche principale afin de l'identifier, afin de pouvoir rollback vers cette version ou les précédentes en cas de dysfonctionnement. Cet environnement est destiné à l'utilisateur final.

## **Livraisons rapides, stables et fiables**

Afin de garantir la fiabilité des livraisons, une CI/CD avec d'indicateurs spécifiques à respecter pour valider son passage doit être instaurée, tel que par exemple le taux de couverture du code.

En parallèle, chaque version doit faire l'objet de tests d'acceptance dans l'environnement de QA afin de s'assurer que les modifications apportées fonctionnent et correspondent aux attentes métier.

Les livraisons rapides et régulières garantissent par ailleurs d'obtenir des retours utilisateurs efficaces afin de pouvoir réagir rapidement en cas de dysfonctionnement ou de mauvaise définition du besoin.

## **Évolutivité**

La mise en place d'une architecture microservice va pouvoir se faire autour du Gateway Pattern qui va nous permettre de développer nos services de manière indépendante puis de gérer les appels à ces services par le biais d'une Gateway centralisant l'ensemble de ses appels et les redirigeant vers les bons services.

## **Gestion des droits**

Tenant compte de l'utilisation majoritaire de Java pour le développement de back-end, une solution telle que Spring Security permettant la gestion de rôle et de droits avec un certain niveau de granularité permettrait de répondre au besoin de rendre disponible des services selon le type d'utilisateur.

En parallèle, Kubernetes peut être configurer pour déployer certains services selon certaines zones géographiques spécifiques par le biais de la configuration des Pods.

## **Sécurité**

De manière générale, mais plus particulièrement pour les éléments ayant trait avec la sécurité, il conviendra de toujours s'appuyer sur les dernières normes et version en cours pour garantir

Actuellement, le protocole de communication sécurisée à respecter dans le web est le protocole HTTPS.

# Principes stratégiques

Les principes stratégiques identifiées jusqu'à présent sont les suivants :

- La solution doit permettre une standardisation des technologies au regard d'une réflexion stratégique.
- La solution doit être fonctionnellement évolutive en fonction du rythme de l'entreprise, en facilitant l'ajout ou la modification de services.
- La solution doit permettre la création et le déploiement de services pour des régions ou des utilisateurs spécifiques.
- La solution doit être sécurisée en tout lieu et toute circonstance.
- La solution doit pouvoir évoluer avec le nombre d'utilisateurs en absorbant les pics d'utilisateurs et permettant la croissance du nombre d'utilisateur grâce à un mécanisme de mise à l'échelle.
- La solution doit être disponible et performante quelque soit la zone géographique.
- La solution doit permettre de supprimer (ou du moins limiter) les interruptions de services lors des déploiements avec la possibilité de rollback facilement en cas de livraison défectueuse.
- La solution doit favoriser l'innovation.
- La solution doit permettre d'effectuer des livraisons régulières stables et fiables

## Références aux conditions requises pour l'architecture

Plusieurs conditions ont été retenus comme bonnes pratiques afin d'avancer sur le travail d'architecture.

### Choix technologiques

- Les solutions open source sont préférables aux solutions payantes.
- Le support continu des composants doit être pris en compte lors de leur sélection ou lors des prises de décision de création ou d'achat.
- Toutes les solutions du commerce ou open source doivent, dans la mesure du possible, faire partie d'une même pile technologique afin de réduire les coûts de maintenance et de support continus.

### Principe data

- Toujours modéliser comme si vous n'aviez pas encore la vision d'ensemble.
- Toujours protéger les données permettant l'identification personnelle.
- Concevoir pour l'accès aux données ou la mutabilité en fonction du problème.
- Appliquer la cohérence en fonction du scénario pour satisfaire au mieux le besoin business. (Ne partez pas du principe que toutes les données doivent être cohérentes immédiatement ou même à terme.)
- Refléter le modèle de domaine au sein d'un contexte délimité de façon appropriée.

### Principe d'application

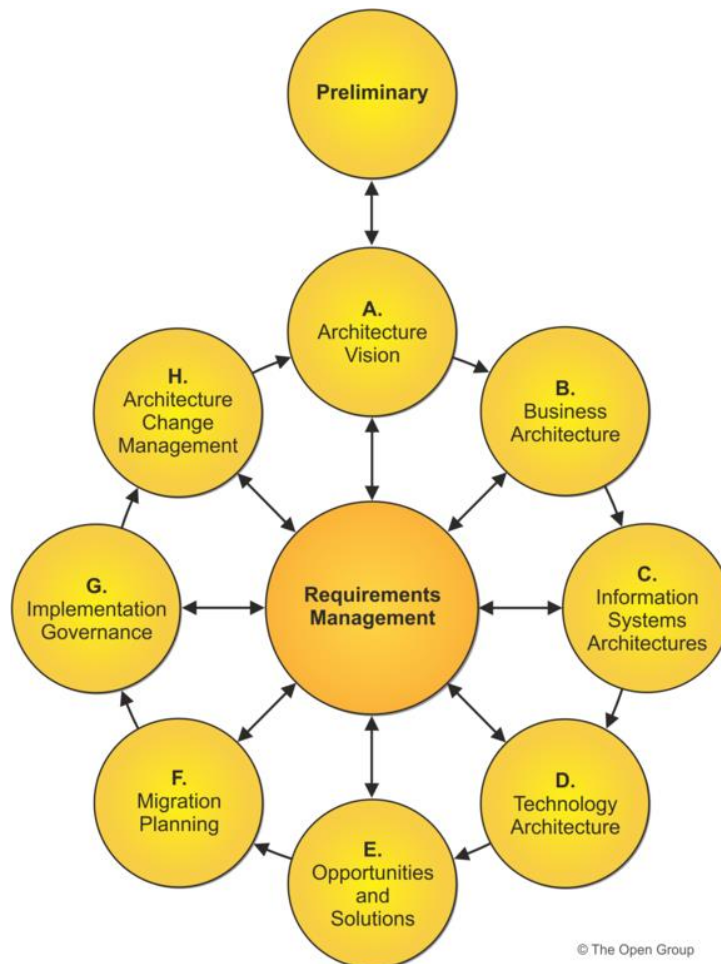
- Responsabilité unique et couplage faible des applications.
- Concevoir des interfaces ouvertes et extensibles en systèmes, sur lesquelles il est facile d'itérer.

- Appliquer une approche pilotée par le contrat client, où les interfaces entre les systèmes reflètent uniquement les données et opérations nécessaires à leur intégration.
- Éviter les dépendances cycliques entre les systèmes.

# Livrables architecturaux

## Développement de l'architecture

Le développement de l'architecture se base sur le déroulement d'un cycle ADM tel que défini dans le cadre de TOGAF.



## Mesures de l'architecture cible

Les métriques de l'architecture cible sont décrits plus bas dans le chapitre « [Critères d'acceptation et procédure – Métrique et KPI de l'état cible de l'architecture](#) ».

## Livraison de l'architecture et métriques business

Les métriques de l'architecture cible sont décrits plus bas dans le chapitre « [Critères d'acceptation et procédure – Métriques de livraison de l'architecture et du business](#) »

## Phases de livraison définies

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.

Livrables exclus de la démarche :

- Organization Model for Enterprise Architecture : Inexistant à l'origine. Scope trop réduit pour justifier un intérêt.
- Tailored Architecture Framework : Inexistant à l'origine. Scope trop réduit pour justifier un intérêt.
- Solution Building Blocks : Si présents, ils seront intégrés directement à d'autres livrables.
- Capability Assessment : Viabilité technique validée (solution déjà existantes et reproduites par d'autres), moyens nécessaires déjà validés.

ID de phase	Phase
<b>P</b>	Préliminaires
<b>A</b>	Vision de l'architecture
<b>B</b>	Architecture business
<b>C</b>	Architecture des systèmes d'information
<b>D</b>	Architecture technologique
<b>E</b>	Opportunités et solutions
<b>F</b>	Planning de migration
<b>G</b>	Gouvernance de l'implémentation
<b>H</b>	Management du changement d'architecture
<b>N</b>	Management des conditions requises

Phase	Entrée	Version	Sortie	Version
<b>P</b>	Architecture Repository	V.0	Architecture Repository	V.P
			Request for Architecture Work <i>includ Business Principles, Goals and Drivers</i>	V.P
<b>A</b>	Architecture Repository	V.P	Architecture Vision	V.A
	Request for Architecture Work	V.P	Statement of Architecture Work <i>includ Communication Plan et Architecture RoadMap</i>	V.A
	Architecture Building Blocks	V.0	Architecture Definition Document	V.A
<b>B</b>	Architecture Building Blocks	V.0	Architecture Definition Document	V.B
	Architecture Repository	V.P	Architecture Requirement Specification	V.B
	Architecture Vision	V.A	Statement of Architecture Work	V.B
	Statement of Architecture Work	V.A		
	Architecture Definition Document	V.A		
<b>C</b>	Architecture Building Blocks	V.0	Architecture Definition Document	V.C
	Architecture Repository	V.P	Architecture Requirement Specification	V.C
	Architecture Vision	V.A	Statement of Architecture Work	V.C
	Architecture Definition Document	V.B		
	Architecture Requirement Specification	V.B		
	Statement of Architecture Work	V.B		
<b>D</b>	Architecture Building Blocks	V.0	Architecture Definition Document	V.D
	Architecture Repository	V.P	Architecture Requirement Specification	V.D
	Architecture Vision	V.A	Statement of Architecture Work	V.D
	Architecture Definition Document	V.C		
	Architecture Requirement Specification	V.C		
	Statement of Architecture Work	V.C		
<b>E</b>	Architecture Building Blocks	V.0	Architecture Vision	V.E
	Architecture Repository	V.P	Architecture Definition Document	V.E
	Architecture Vision	V.A	Architecture Requirement Specification	V.E
	Architecture Definition Document	V.D	Statement of Architecture Work	V.E
	Architecture Requirement Specification	V.D	Implementation and Migration Plan	V.E
	Statement of Architecture Work	V.D		
<b>F</b>	Architecture Repository	V.P	Architecture Requirement Specification	V.F



	Architecture Vision	V.E	Implementation and Migration Plan <i>includ Implementation Governance Model</i>	V.F
	Architecture Definition Document	V.E	Architecture Definition Document	V.F
	Architecture Requirement Specification	V.E	Request for Architecture Work	V.F
	Statement of Architecture Work	V.E	Statement of Architecture Work	V.F
	Implementation and Migration Plan	V.E		
<b>G</b>	Architecture Repository	V.P	Architecture Contract	V.G
	Architecture Requirement Specification	V.F	Compliance Assessment	V.G
	Implementation and Migration Plan	V.F	Statement of Architecture Work	V.G
	Architecture Definition Document	V.F		
	Request for Architecture Work	V.F		
	Statement of Architecture Work	V.F		
	Architecture Vision	V.E		
	Architecture Contract	V.0		
<b>H</b>	Architecture Repository	V.P	Change Request	V.H
	Architecture Definition Document	V.F	Request for Architecture Work	V.H
	Implementation and Migration Plan	V.F	Statement of Architecture Work	V.H
	Architecture Requirement Specification	V.F		
	Architecture Vision	V.E		
	Architecture Contract	V.G		
	Compliance Assessment	V.G		
	Statement of Architecture Work	V.G		
<b>N</b>	Requirement Impact Assessment	V.0	Requirement Impact Assessment	V.X
	Architecture Repository	V.P	Architecture Requirement Specification	V.X
	Architecture Requirement Specification	V.X		
	Architecture Vision	V.X		
	Statement of Architecture Work	V.X		

# Plan de travail commun priorisé

---

Cette section reprend toutes les activités et tous les livrables du travail d'architecture en spécifiant le rôle du design et du développement dans celles-ci. Elle peut être amenée à évoluer selon les éléments mis en avant dans chaque phase.

Il est à noter que l'architecte d'entreprise est concerné par l'ensemble de ses phases, étant responsable du bon déroulement de la définition d'architecture.

## Plan d'architecture

### 1. Phase préliminaire

#### a. Activités

La phase préliminaire permet de définir les attentes et les objectifs à atteindre avec la nouvelle architecture et encadre le lancement du projet.

#### b. Livrables

##### Architecture Repository

Créer sous Git et consultable sur GitHub, il permet de répertorier et versionner le travail de définition d'architecture tout au long du déroulement du projet.

- L'architecture repository devra être accessible aux équipes de développement afin qu'ils puissent y consulter les documents de travail d'architecture qui les concerne.

##### Request for Architecture Work

Ce document est produit par l'organisation sponsor pour déclencher le cycle de développement d'architecture. Il permet de confirmer le contexte et les objectifs du projet. Il inclut également des spécifications concernant les principes business et les objectifs généraux de Foosus.

- Il est produit conjointement avec la CIO qui valide la faisabilité technique du projet.

### 2. Phase A – Vision de l'architecture

#### a. Activités

La vision de l'architecture fournit une première vision sur les changements attendus grâce à la mise en place de l'architecture cible et encadre le scope et les moyens mis en place atteindre cette architecture.

#### b. Livrables

##### Architecture Vision

Elle permet de produire une vision sommaire des changements apportés à l'entreprise grâce au déploiement de l'architecture cible et cadre le résultat attendu pour les parties prenantes. Une matrice des parties prenantes (en : **Stakeholder Map Matrix**) permet d'identifier ces parties prenantes et leur niveau d'application. Une **Value Stream Map** permet d'identifier la chaîne de fonctionnement du produit et les étapes sur lesquelles se concentrent la création de valeur afin de se concentrer sur les détails nécessaires au bon fonctionnement de l'entreprise.

L'Architecture Vision se veut être une version résumée de la définition d'architecture à destination de l'ensemble des parties prenantes.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

### **Statement of Architecture Work**

La déclaration de travail d'architecture (en : Statement of Architecture Work) définit le scope et l'approche qui seront utilisé pour compléter le cycle de développement de l'architecture, ainsi que le plan de communication et le calendrier associé. Le calendrier pourra au fil des phases être de plus en plus précis afin de permettre la définition d'une Architecture RoadMap listant l'ensemble des packages réalisés par l'architecture cible et les liant à une timeline pour montrer la progression de l'existant vers la cible.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

### **Architecture Definition Document**

La définition d'architecture contient les principaux artefacts architecturaux crée pendant le projet et les informations importantes qui y sont liées. A cette étape, nous retrouvons la définition du scope du projet ainsi que la description des objectifs et des contraintes déjà identifiés.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

## **3. Phase B – Architecture Business**

### **a. Activités**

La phase d'architecture business doit permettre d'établir une définition de l'architecture business attendue ainsi que l'ensemble des spécifications que l'implémentation se doit de respecter afin de s'y conformer.

### **b. Livrables**

### **Architecture Definition Document**

Pendant cette phase, la définition d'architecture doit permettre de définir les principes et le modèle d'architecture business cible afin de faire l'analyse des écarts avec l'architecture business actuelle, puis proposer une première vision business des étapes de transitions nécessaires. Afin de produire ce document, il est important de :

Comprendre le produit de l'entreprise et son cycle de vie en s'appuyant sur les artefacts suivants :

- **Driver / Goal / Objective Catalog**, afin d'identifier les objectifs qui permettent de répondre à la stratégie de l'entreprise
- **Business Service / Function Catalog**, afin d'identifier les capacités fonctionnelles actuelles et souhaitées par l'entreprise

- **Process / Event / Control / Product Catalog**, afin d'identifier et de hiérarchiser les process, les événements qu'ils traversent, leurs livrables et les éléments de contrôle d'exécution des process
- **Contract / Measure Catalog**, afin d'identifier les services pour lesquels l'entreprise s'est engagée
- **Value Stream Stages Catalog**, afin d'identifier les étapes qui permettent la création de la valeur pour arriver au produit final de l'utilisateur
- **Business Service / Information Diagram**, afin d'identifier les informations, données et sources nécessaires pour exécuter les services métiers et visualiser les dépendances fonctionnelles
- **Product Lifecycle Diagram**, afin d'identifier le cycle de vie du produit et assurer les contrôle et process nécessaires dans l'architecture
- **Business Use-Case Diagram**, afin d'identifier les relations entre les clients et les services métiers, en illustrant comment et quand une capacité fonctionnelle est utilisée
- **Value Stream Map**, afin d'identifier la chaîne de fonctionnement du produit et les étapes qui permettent la création de valeur

Comprendre les interactions des différentes parties prenantes à travers ses artefacts :

- **Organization / Actor Catalog**, afin d'identifier les différents participants qui interagiront avec les équipes informatiques
- **Role Catalog**, afin d'identifier les différents niveaux d'autorisation possibles
- **Actor / Role Matrix**, afin d'identifier quel acteur dispose de quel rôle et optimiser l'analyse des besoins de formations, les paramètres de sécurité utilisateur et anticipé les changements d'organisation et de management

Comprendre le scope concerné au-delà des parties prenantes grâce aux artefacts suivants :

- **Location Catalog**, afin d'identifier les lieux où l'entreprise exerce son activité, ainsi que tout autre élément relatif à l'architecture, tel que la présence de Data Center.

### **Architecture Requirement Specification**

Ce document permet de formaliser l'ensemble des éléments quantitatifs que le projet d'implémentation doit respecter pour se conformer à l'architecture business, énumérant des critères mesurables qui doivent être remplis durant l'implémentation de l'architecture.

Pour le réaliser, il est possible de se baser sur les mêmes artefacts que ceux énumérés pour la définition d'architecture qui propose une vision qualitative de l'architecture business.

### **Statement of Architecture Work**

La déclaration de travail doit être mise à jour au regard des éléments soulevés pendant le déroulement de la phase. L'architecture roadmap peut être développée en proposant une vision business du déroulement des étapes vers l'architecture cible.

## **4. Phase C – Architecture des systèmes d'information**

### **a. Activités**

La phase d'architecture SI doit permettre d'établir une définition de l'architecture SI attendue ainsi que l'ensemble des spécifications que l'implémentation se doit de respecter afin de s'y conformer.

## b. Livrables

### Architecture Definition Document

Pendant cette phase, la définition d'architecture doit permettre de définir les principes et le modèle d'architecture SI cible afin de faire l'analyse des écarts avec l'architecture SI actuelle, puis proposer une première vision SI des étapes de transitions nécessaires. Afin de produire ce document, il est possible de se baser sur les artefacts suivants :

- **Data Entity / Data Component Catalog**, afin d'identifier la liste de toutes les données présentes dans l'entreprise, leur usage et leur gouvernance
- **Conceptual Data Diagram**, afin d'identifier les relations entre les entités de données critique dans l'entreprise à destination des parties prenantes métier
- **Logical Data Diagram**, afin d'identifier les relations logiques entre les entités de données dans l'entreprise
- **Data Security Diagram**, afin d'identifier quel acteur peut accéder à quel donnée de l'entreprise ainsi que les règles applicables en matière de données
- **Data Migration Diagram**, afin d'identifier le chemin des données, de leur source jusqu'à leur cible ainsi que les transformation nécessaires
- **Application / Function Matrix**, afin d'identifier les relations entre application et fonction métier dans l'entreprise
- **Application Use-Case Diagram**, afin d'identifier les relations entre consommateurs et fournisseurs des services applicatifs en illustrant comment et quand cela est fonctionnellement utilisé
- **Software Engineering Diagram**, afin d'identifier les packages, modules, services et opérations dans l'application dans une perspective de développement
- **Application Migration Diagram**, afin d'identifier les étapes de migration de l'application de l'état actuel jusqu'à la cible
- **Software Distribution Diagram**, afin d'identifier comment le logiciel applicatif est structuré et distribué

Lors de l'élaboration de ce document :

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner fournit les éléments techniques nécessaires.
- Les équipes de développement peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

### Architecture Requirement Specification

Ce document permet de formaliser l'ensemble des éléments quantitatifs que le projet d'implémentation doit respecter pour se conformer à l'architecture SI, énumérant des critères mesurables qui doivent être remplis durant l'implémentation de l'architecture.

Pour le réaliser, il est possible de se baser sur les mêmes artefacts que ceux énumérés pour la définition d'architecture qui propose une vision qualitative de l'architecture SI.

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner fournit les éléments techniques nécessaires.
- Les équipes de développement peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

## **Statement of Architecture Work**

La déclaration de travail doit être mise à jour au regard des éléments soulevés pendant le déroulement de la phase. L'architecture roadmap peut être développée en proposant une vision SI du déroulement des étapes vers l'architecture cible.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

## **5. Phase D – Architecture technologique**

### **a. Activités**

La phase d'architecture technologique doit permettre d'établir une définition de l'architecture technologique attendue ainsi que l'ensemble des spécifications que l'implémentation se doit de respecter afin de s'y conformer.

### **b. Livrables**

#### **Architecture Definition Document**

Pendant cette phase, la définition d'architecture doit permettre de définir les principes et le modèle d'architecture technologique cible afin de faire l'analyse des écarts avec l'architecture technologique actuelle, puis proposer une première vision technologiques des étapes de transitions nécessaires. Afin de produire ce document, il est possible de se baser sur les artefacts suivants :

- **Technology Portfolio Catalog**, afin d'identifier la liste des technologie utilisé dans l'entreprise, y compris la gestion de leur cycle de vie et de leur versionning
- **Platform Decomposition Catalog**, afin d'illustrer les plateformes technologiques qui vont supporter les opérations de l'architecture des systèmes d'information
- **Processing Diagram**, afin d'identifier des unité de code et de configuration déployables et les technologies sur lesquels elles sont déployées

Lors de l'élaboration de ce document :

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner et l'Operations Lead fournissent les éléments techniques nécessaires.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

#### **Architecture Requirement Specification**

Ce document permet de formaliser l'ensemble des éléments quantitatifs que le projet d'implémentation doit respecter pour se conformer à l'architecture technologique, énumérant des critères mesurables qui doivent être remplis durant l'implémentation de l'architecture.

Pour le réaliser, il est possible de se baser sur les mêmes artefacts que ceux énumérés pour la définition d'architecture qui propose une vision qualitative de l'architecture technologique.

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner et l'Operations Lead fournissent les éléments techniques nécessaires.

- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

### **Statement of Architecture Work**

La déclaration de travail doit être mise à jour au regard des éléments soulevés pendant le déroulement de la phase. L'architecture roadmap peut être développée en proposant une vision technologique du déroulement des étapes vers l'architecture cible.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

## **6. Phase E – Opportunités et solutions**

### **a. Activités**

Cette phase doit permettre de dégager les opportunités et les solutions générées par l'architecture cible définit précédemment, afin de les concrétiser et si besoin de modifier les livrables précédemment fournis.

### **b. Livrables**

### **Architecture Definition Document**

La définition d'architecture va prendre en compte l'analyse d'impact et ses conclusions en compléments des éléments déjà évalués en considérant la création des artefacts suivants :

- **Project Context Diagram**, afin d'identifier les organisations, fonctions, service, process, applications, données et technologie qui vont être ajoutés, enlevés ou impactés par le projet
- **Benefits Diagram**, afin d'illustre les opportunités identifiées par la définition de l'architecture et les hiérarchiser selon leur taille, bénéfices et complexité.

Lors de l'élaboration de ce document :

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner et l'Operations Lead fournissent les éléments techniques nécessaires.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

### **Architecture Vision**

L'Architecture Vision peut être reprise pour résumer l'ensemble de la définition d'architecture qui est maintenant achevée et précise ainsi que les impacts issues de la modification de l'architecture, notamment les bénéfices qui en seront tirés.

- La CIO valide les décisions techniques prises dans le document avec le support de l'Engineering Owner et de l'Operations Lead.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

### **Architecture Requirement Specification**

Ce document permet de formaliser l'ensemble des éléments quantitatifs que le projet d'implémentation doit respecter pour se conformer à l'architecture définie et aux opportunités

soulevées pendant cette phase, énumérant des critères mesurables qui doivent être remplis durant l'implémentation de l'architecture.

Pour le réaliser, il est possible de se baser sur les mêmes artefacts que ceux énumérés pour la définition d'architecture qui propose une vision qualitative de l'architecture et des opportunités qui lui sont liées.

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner et l'Operations Lead fournissent les éléments techniques nécessaires.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

### **Statement of Architecture Work**

La déclaration de travail d'architecture (en : Statement of Architecture Work) définit le scope et l'approche qui seront utilisés pour compléter le cycle de développement de l'architecture, ainsi que le plan de communication et le calendrier associé.

Cette phase permet d'intégrer les opportunités soulevées et de mettre à jour avec les éléments qui ont pu être mis en avant pendant la définition d'architecture.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

### **Implementation and Migration Plan**

Ce document se concentre sur l'implémentation et la mise en place par étape de l'architecture cible. On y retrouvera le calendrier détaillé du projet avec l'exposé des risques, solutions, opportunités à chaque étape de la migration vers l'architecture cible.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

## **7. Phase F – Planning de Migration**

### **a. Activités**

Sur la base du travail fourni lors des précédentes phases et des décisions prises pour l'architecture cible, cette phase va permettre de définir les étapes de transition architecturale ainsi que tout élément qui entoure cette transition.

### **b. Livrables**

### **Architecture Definition Document**

La définition d'architecture va pouvoir finaliser les artefacts concernant la transition architecturale en se basant sur la vision d'ensemble offerte par les phases précédentes.

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner et l'Operations Lead fournissent les éléments techniques nécessaires.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.



## **Architecture Requirement Specification**

Ce document permet de formaliser l'ensemble des éléments quantitatifs que le projet d'implémentation doit respecter pour se conformer à l'architecture définie et aux étapes de migration définies pendant cette phase, énumérant des critères mesurables qui doivent être remplis durant l'implémentation de l'architecture.

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner et l'Operations Lead fournissent les éléments techniques nécessaires.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

## **Statement of Architecture Work**

La déclaration de travail d'architecture (en : Statement of Architecture Work) définit le scope et l'approche qui seront utilisés pour compléter le cycle de développement de l'architecture, ainsi que le plan de communication et le calendrier associé.

Cette phase permet d'intégrer la définition des étapes de migration de l'architecture, notamment le calendrier associé à son implémentation (Architecture RoadMap).

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

## **Request for Architecture Work**

Ce document est produit par l'organisation sponsor pour déclencher le cycle de développement d'architecture. Il permet de confirmer le contexte et les objectifs du projet. Il inclut également des spécifications concernant les principes business et les objectifs généraux de Foosus.

Pendant cette phase, ce document va permettre de valider la vision finale mise en avant pendant le travail d'architecture, notamment avec les sponsors, afin de pouvoir lancer le cycle de migration de l'architecture.

- Il est produit conjointement avec la CIO qui valide la vision finale technique du projet.

## **Implementation and Migration Plan**

Ce document se concentre sur l'implémentation et la mise en place par étape de l'architecture cible. On y retrouvera le calendrier détaillé du projet avec l'exposé des risques, solutions, opportunités à chaque étape de la migration vers l'architecture cible.

Pendant cette phase, en plus de la révision du planning d'implémentation, il sera important de planifier comment l'architecture de transition sera gouvernée pendant l'implémentation (**Implementation Governance Model**).

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- l'Engineering Owner et l'Operations Lead fournissent les éléments techniques nécessaires.
- Les équipes de développement et d'Ops peuvent être consulté sur le sujet pour leur expertise sur les aspects techniques.

## **8. Phase G – Gouvernance de l'implémentation**

### **a. Activités**

Cette phase permet de cadrer la gouvernance de l'implémentation en formalisant les engagements pris par les parties dans la gestion de l'implémentation et en assurant un cadre qualitatif à l'implémentation.

#### **b. Livrables**

##### **Architecture Contract**

Cet ensemble de documents permet d'établir des accords entre les partenaires et les sponsors sur les livrables à fournir en fin de cycle, la qualité attendue de ces livrables et la nécessité de l'architecture choisie pour répondre à ces questions.

- Les équipes de développement et d'Ops sont destinataires de l'un des contrats d'architecture (ici).

##### **Compliance Assessment**

Ce document formalise les éléments de suivi de l'implémentation afin de s'assurer qu'elle respecte l'Architecture Vision et que les retours d'expérience permettant l'évolution et l'amélioration de l'implémentation sont bien pris en compte.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

##### **Statement of Architecture Work**

La déclaration de travail d'architecture (en : Statement of Architecture Work) définit le scope et l'approche qui seront utilisés pour compléter le cycle de développement de l'architecture, ainsi que le plan de communication et le calendrier associé.

Cette phase permet d'intégrer la définition des éléments de gouvernance de l'implémentation, notamment les processus de suivi de la qualité de l'implémentation.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

## **9. Phase H – Management du changement d'architecture**

#### **a. Activités**

Au cours de l'implémentation, des modifications peuvent être souhaitables pour parvenir à une architecture répondant pleinement au besoin. Cette phase permet d'encadrer la mise en place de ces changements afin qu'ils soient correctement communiqués aux parties prenantes et pleinement intégrés dans l'implémentation de l'architecture.

#### **b. Livrables**

##### **Change Request**

Ce document permet de formaliser une demande de changement de l'architecture cible initiale dans l'hypothèse où de nouveaux éléments seraient à prendre en compte et viendrait impacter l'architecture prévue. Pour chaque modification souhaitée, il doit comprendre une description de ce changement, la raison de ce changement et faire une analyse de l'impact de ce changement.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

### **Request for Architecture Work**

Toute demande de changement doit faire l'objet d'une information et d'une validation auprès des sponsors. Aussi, la requête de travail doit être mise à jour en intégrant le changement et valider auprès des sponsors pour déclencher la mise en place du changement.

- Il est produit conjointement avec la CIO qui valide la vision technique du changement.

### **Statement of Architecture Work**

La déclaration de travail d'architecture (en : Statement of Architecture Work) définit le scope et l'approche qui seront utilisés pour compléter le cycle de développement de l'architecture, ainsi que le plan de communication et le calendrier associé.

Toute demande de changement pouvant avoir un impact sur le déroulé du projet, il convient d'actualiser la déclaration de travail afin de prendre en compte les modifications validées.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.

## **10. Phase N – Management des conditions requises**

### **a. Activités**

Cette phase est une phase intermédiaire à répéter entre chacune des autres phases afin de s'assurer que le travail effectué dans chaque phase n'impacte pas le travail précédemment fourni et de cibler à quelle phase il est nécessaire de retourner pour bien prendre en considération les impacts des nouveaux éléments fournis.

### **b. Livrables**

#### **Requirement Impact Assessment**

Ce document permet d'identifier les changements qui devraient être faits et l'impact de ses changements lors de la prise en compte de nouveaux éléments en s'appuyant sur le Requirement Catalog qui liste les éléments que l'entreprise doit faire pour atteindre ces objectifs. Les changements identifiés permettent de cibler à quelle phase du développement il faut remonter pour prendre en considération les impacts sur l'ensemble du cycle.

- La CIO définit et valide l'orientation stratégique technique énoncé dans le document avec le support de l'Engineering Owner et de l'Operations Lead.
- Les équipes de développement peuvent s'appuyer sur ce document pour apporter les corrections nécessaires dans leur travail.

#### **Architecture Requirement Specification**

Ce document permet de formaliser l'ensemble des éléments quantitatifs que le projet d'implémentation doit respecter pour se conformer à l'architecture, énumérant des critères mesurables qui doivent être remplis durant l'implémentation de l'architecture.

Cette phase impose de revoir les éléments quantitatifs sélectionnés pour éventuellement ajouter ce qui auront pu être mis en avant par l'étude d'impact.

- La CIO valide les décisions stratégiques techniques énoncées dans le document
- L'Engineering Owner fournit les éléments techniques nécessaires.
- Les équipes de développement peuvent être consultées sur le sujet pour leur expertise sur les aspects techniques.

# Plan de communication

## 1. Création ou mise à jour d'un document ou d'un artefact

Les équipes informatiques peuvent être concernées selon le document finalisé ou mis à jour.

<b>Public cible</b>	Parties prenantes concernées selon le sujet - Voir liste de distribution en en-tête des documents ou le RACI de la déclaration de travail
<b>Modalités</b>	Diffusion sur le repository GitHub à la finalisation ou validation du document Envoie par mail aux parties prenantes concernées. En cas de mise à jour, préciser les changements apportés
<b>Fréquence Durée</b>	Dès la finalisation / validation du document
<b>Auteur</b>	Auteur du document

## 2. Réunion du groupe projet

Le groupe projet intègre notamment l'Engineer Owner et l'Operation Lead.

D'autres membres des équipes informatiques peuvent être concernées selon le sujet en cours.

<b>Public cible</b>	Parties prenantes concernées selon le sujet - Voir liste de distribution en en-tête des documents ou le RACI de la déclaration de travail
<b>Modalités</b>	Réunion de travail en présentiel ou distanciel Revue des avancées, problèmes, besoins, objectifs de manière détaillée
<b>Fréquence Durée</b>	1 fois par semaine Minimum 2 heures, peut prévoir plus sur certains sujets spécifiques
<b>Auteur</b>	Chef du projet

## 3. Réunion des sponsors

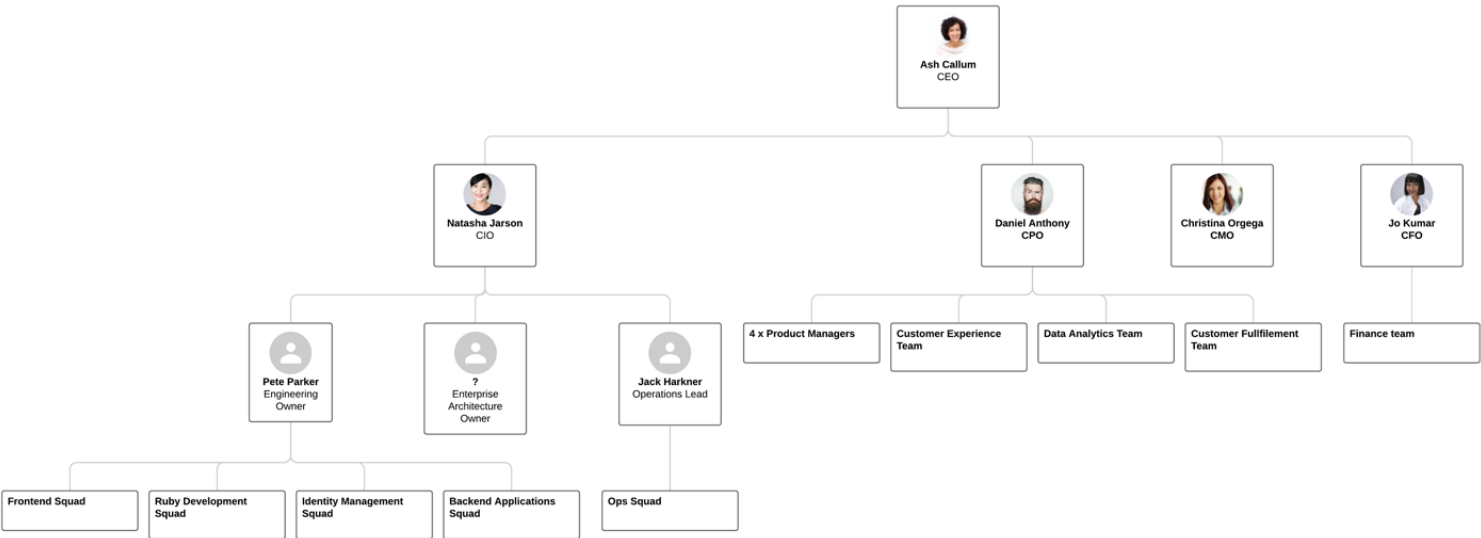
La CIO sera convoquée à cette réunion en tant que membre du CODIR

<b>Public cible</b>	Sponsors et membres du CODIR
<b>Modalités</b>	Réunion de travail en présentiel ou distanciel Revue des avancées, problèmes, besoins, objectifs de manière sommaire Validation finale des éléments qui ont besoin d'être validés
<b>Fréquence Durée</b>	1 fois par semaine 30 minutes, sauf sujet particulier à aborder
<b>Auteur</b>	Chef du projet

# Risques et facteurs de réduction

## Structure de gouvernance

Figure 1 : Organigramme hiérarchique



## Analyse des risques

P : Probabilité

D : Dangersité

C : Criticité

Risque	P	D	C	Actions
Non-réponse au besoin fonctionnel				AGILE, Critères d'acceptation
Non-réponse au besoin technique				AGILE, KPI
Tensions entre parties				Communication régulière et transparente

P : Probabilité		D : Niveau de Danger	
Peu probable		Très peu de conséquences, retards légers	
Probable		Peu de conséquences, retards	
Très probable		Conséquences non-négligeables, abandon de fonctionnalités	
Certains		Conséquences importantes, abandon du projet	

## Hypothèses

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

Hypothèses	Impact	Propriétaire
Le budget alloué pour la mise en place du projet est suffisant	Les décisions prises ne le sont pas au regard d'une contrainte budgétaire établie	CFO
Les compétences nécessaires pour la mise en place du projet sont disponibles en interne	Les décisions prises considèrent que la faisabilité technique est possible sans recrutement externe, ni prestataire	CIO

# Critères d'acceptation et procédures

## Métriques et KPI de l'état cible de l'architecture

Indicateur	Cible
Taux de disponibilité	> 99%
Disponibilité géographique	> 90%
Plateformes supportées	Android IOS Web : <ul style="list-style-type: none"><li>- Edge</li><li>- Chrome</li><li>- Firefox</li><li>- Safari</li></ul>
Type d'architecture	Microservice
Langage back-end	Java
Protocole de communication sécurisée	HTTPS

## Métriques de livraison de l'architecture et du business

Indicateur	Changement attendu
Nombre d'adhésions d'utilisateurs par jour	Augmenter de 10 %
Adhésion de producteurs alimentaires	Passer de 1,4 / mois à 4 / mois
Délai moyen de parution	Passer de 3,5 semaines à moins d'une semaine
Taux d'incident de production	Passer de > 25 / mois à < 1 / mois

## Procédure d'acceptation

Chaque élément à valider devra être validé avec le groupe projet puis passer en réunion des sponsors afin d'obtenir une validation finale. Une fois la validation obtenue, l'élément pourra être communiqué à l'ensemble des parties prenantes.

## Procédures de changement de périmètre

Tout changement de périmètre du projet doit faire l'objet d'une validation par les membres du comité de direction afin d'approuver la vision stratégique portée par le projet ainsi que d'une validation par les membres du comité de pilotage du projet afin de valider sa faisabilité.

Un tel changement doit au préalable de sa validation faire l'objet de la rédaction d'un document de requête faisant état notamment :

- Des changements proposés
- Des raisons de ce changement
- D'une analyse de l'impact de ce changement (humain, financiers logistique...)

Le changement de périmètre doit être communiquée à l'ensemble des parties prenantes afin qu'elles puissent considérer ce changement dans leurs actions.

À la suite d'un changement de périmètre, il conviendra de réitérer sur l'ensemble des documents du projet afin d'assurer leur cohérence avec ce changement.

L'ensemble des impacts devra être communiqué par la suite aux services de développement concernés.

## Développement et propriété de l'architecture

Le tableau ci-dessous résume le rôle de chaque partie prenante dans le cadre du process du projet. Chaque étape pourrait être détaillée en sous-étapes afin d'obtenir une granularité plus fine et ainsi une meilleure vision de l'implication de chacun dans le projet.

Figure 2 : Légende du RACI

<b>R</b>	<b>Réalisateur</b>	Réalise la tâche et est responsable de son achèvement
<b>A</b>	<b>Approbateur</b>	Approuve l'achèvement de la tâche
<b>C</b>	<b>Consultant</b>	Conseille, intervient avant une décision ou une action
<b>I</b>	<b>Informé</b>	Doit être informé après une décision ou une action

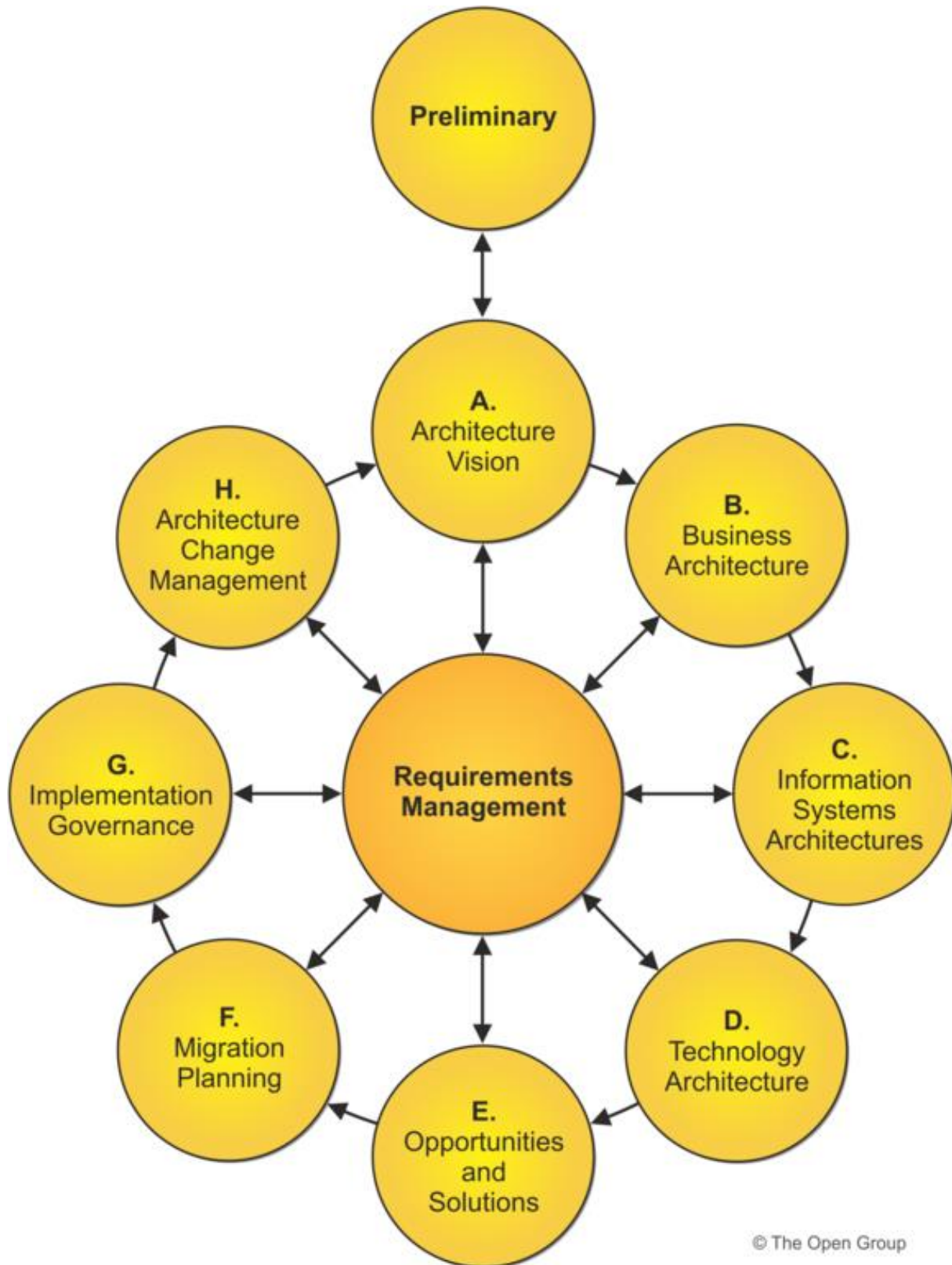


Figure 3 : RACI

Parties Prenantes		Statut		ID															
Dir.		Tâches		ID															
Chief Executive Officer		Ash CALLUM		Phase 1 : Lancement du projet															
Chief Information Officer		Natasha JARSON		P101															
Engineering Owner		Pete PARKER		P102															
Frontend Squad				Phase 2 : Planification															
Ruby Development Squad				P201															
Identity Management Squad				P202															
Backend Applications Squad				P203															
Enterprise Architecte Owner				P204															
Operations Lead		Jack HARKNER		Phase 3 : Réalisation															
Ops Squad				P301															
Chief Product Officer		Daniel ANTHONY		P302															
Product Managers x4				P303															
Customer Experience Team				Phase 4 : Contrôle															
Data Analytics Team				P401															
Customer Fulfillement Team				P402															
Chief Marketing Officer		Christina ORGEGA		Phase 5 : Production															
Chief Financial Officer		Jo KUMAR		P501															
Finance Team				P502															

# Calendrier

Le planning du projet suivra le cycle ADM tel que défini par TOGAF. Il conviendra de venir apporter des précisions à ce cycle par la suite.



## Phases de livrables définies

---

La phase de définition de l'architecture permettra d'aboutir sur la livraison d'un ensemble de document tel que définit précédemment, et notamment d'un document de définition d'architecture permettant la mise en application par les équipes de développement et de déploiement.

Une fois la phase de développement terminé, les livrables attendus sont d'une part la plateforme Foosus opérationnel selon les conditions définis par la définition d'architecture mais aussi un guide actualisé à destination des différents utilisateurs finaux.

La documentation recoupant l'ensemble des spécifications techniques de la plateforme à destination des équipes en charge de la maintenance et de l'évolution de la plateforme devra être fourni, sur la base du travail effectué lors de la définition d'architecture.

# Approbations signées

---

**Ash CALLUM**

Chief Executive Officer

**Natasha JARSON**

Chief Information Officer

**Daniel ANTHONY**

Chief Product Officer

**Christina ORGEGA**

Chief Marketing Officer

**Jo KUMAR**

Chief Financial Officer

Date de signature :