



## Déclaration de Travail d'Architecture - FOOSUS

Version : 1.0

Auteur : Clément Hindié

Date : 20 janvier 2026

Document : Déclaration de Travail d'Architecture

Projet : Nouvelle architecture

## *Historique des modifications*

## Tables des matières

1.	Objet du document .....	4
2.	Vision d'Architecture .....	4
2.1	Intention.....	4
2.2	Principes structurants .....	5
2.3	Adaptation de l'ADM au contexte FOOSUS.....	5
3.	État Cible de l'Architecture.....	6
3.1	Architecture cible.....	6
3.1.1	Diagramme d'Architecture .....	6
3.1.2	Composants de l'architecture cible .....	7
3.2	Domaines fonctionnels.....	8
3.3	Coexistence avec l'existant.....	8
4.	Portée du Travail d'Architecture .....	9
4.1	Plan de mise en œuvre de haut niveau .....	9
5.	Parties Prenantes.....	9
6.	Approche d'Architecture .....	10
6.1	Limites responsables.....	10
6.2	Gestion du changement et migration .....	10
7.	Livrables .....	11
8.	Critères de réussite.....	11
9.	Risques et Atténuations .....	12
10.	Engagements .....	13

## 1. Objet du document

Ce document définit le cadre de travail architectural pour la nouvelle plateforme FOOSUS.

Il présente la vision, les objectifs, les responsabilités et l'approche retenue pour construire une architecture cohérente, évolutive et adaptée à la croissance de l'entreprise.

Il constitue une adaptation sur mesure du cadre TOGAF, appliquée de manière pragmatique et lean : seules les activités, vues et décisions réellement utiles à FOOSUS sont retenues. L'objectif est de fournir une base technique saine et durable, capable de soutenir l'innovation produit tout en garantissant la stabilité opérationnelle.

## 2. Vision d'Architecture

### 2.1 Intention

FOOSUS doit disposer d'une plateforme géo-centrée, modulaire et fiable, capable de connecter efficacement consommateurs et producteurs locaux.

Cette plateforme doit permettre aux équipes produit d'expérimenter rapidement, tout en assurant que l'architecture soutienne la croissance plutôt que de la freiner.

Cette vision correspond à la phase Architecture Vision du cadre TOGAF, adaptée à la culture lean de FOOSUS.

## 2.2 Principes structurants

Les principes suivants guident toutes les décisions architecturales :

- **Responsabilités claires** : chaque service couvre un périmètre fonctionnel limité et non ambigu.
- **Normalisation pragmatique** : des normes existantes, mais ne brident pas la créativité.
- **Évolutivité horizontale** : la plateforme doit absorber les pics de charge sans dégrader l'expérience utilisateur.
- **Livraisons continues** : versions petites, peu risquées, déployées sans interruption.
- **Sécurité intégrée dès la conception** : aucun compromis entre facilité d'usage et protection des données.
- **Observabilité native** : logs structurés, métriques, traces et alertes dès le départ.
- **Réversibilité** : les décisions doivent être faciles à inverser tant que possible.
- **Coexistence maîtrisée** : migration progressive depuis l'existant, sans bascule brutale.

Ces principes constituent le framework d'architecture sur mesure de FOOSUS.

## 2.3 Adaptation de l'ADM au contexte FOOSUS

FOOSUS adopte une version allégée et pragmatique de l'ADM de TOGAF, adaptée à une start-up Lean :

- Phase Vision → Définition de la direction stratégique et des capacités prioritaires (géolocalisation en premier incrément).
- Phase Business/Data/Application/Technologie → Fusionnées en un cycle court centré sur la livraison incrémentale.
- Phase Exigences → Formulée dans la SCRA, servant de base à la conformité.
- Phase Opportunités & Solutions → Définition des incréments successifs.
- Phase Migration Planning → Approche progressive par Strangler pattern.
- Phase Implémentation Gouvernance → Revues légères et continues.
- Phase Architecture Change Management → Ajustements rapides basés sur l' apprentissage produit.

Cette ADM personnalisée garantit un cadre structurant tout en préservant la rapidité d'exécution.

### 3. État Cible de l'Architecture

#### 3.1 Architecture cible

L'architecture cible repose sur un ensemble de micro-services organisés par domaines métier et exposant des API standardisées.

Elle doit :

- Fonctionner dans un environnement cloud.
- Intégrer la géolocalisation comme capacité native.
- Supporter plusieurs régions géographiques.
- Fonctionner en continu sans interruption lors des déploiements.
- Coexister temporairement avec la plateforme actuelle.
- Offrir une expérience fluide sur mobile même avec une faible bande passante.

Cette cible servira de référence pour les phases ultérieures du cadre TOGAF (Business, Data, Application et Technologie d'Architecture), qui seront détaillées au fil des itérations.

##### 3.1.1 Diagramme d'Architecture

La plateforme FOOSUS repose sur une architecture modulaire organisée autour de domaines métiers clairement définis.

Le diagramme qui suit, illustre les principaux composants, leurs responsabilités et leurs interactions avec les parties prenantes internes et externes.



Figure 3-1 Vue logique de l'architecture cible FOOSUS

Cette vue présente l'architecture cible FOOSUS structurée en couches selon le cadre TOGAF.

Elle distingue les utilisateurs, les canaux d'accès, les services métiers, les services techniques transverses, les données et l'infrastructure cloud.

Cette structuration clarifie les responsabilités, les interactions et les dépendances entre les composants.

### 3.1.2 Composants de l'architecture cible

L'architecture cible s'appuie sur un ensemble de composants spécialisés, chacun responsable d'un périmètre fonctionnel précis.

Ces composants sont conçus pour être faiblement couplés, observables et déployables indépendamment.

- **Search Service :**
  - Rôle : recherche géolocalisée des offres et des producteurs
  - Capacités : indexation, filtrage, système de notation, géo-distance
  - Interfaces : API REST /search
  - Justification : composant critique pour l'expérience utilisateur
  
- **Catalog Service :**
  - Rôle : gestion des offres alimentaires et métadonnées associées
  - Capacités : CRUD des offres, disponibilité, catégorisation
  - Interfaces : API REST /catalog
  - Justification : cœur de métier de FOOSUS
  
- **User & Supplier Service :**
  - Rôle : gestion des comptes consommateurs et producteurs
  - Capacités : profils, préférences, vérifications
  - Interfaces : API REST /users et /suppliers
  - Justification : séparation claire des responsabilités
  
- **Back-Office Service :**
  - Rôle : supervision, qualité, support, modération
  - Capacités : tableau de bord, alertes, gestion des incidents
  - Interfaces : API REST /admin
  - Justification : indispensable pour l'exploitation
  
- **Experimentation Service :**
  - Rôle : gestion des feature flags et variantes
  - Capacités : activation ciblée, tests A/B, suivi des métriques
  - Interfaces : API REST /experiments
  - Justification : support à l'innovation produit
  
- **Geo Service :**
  - Rôle : géolocalisation, calculs de distance, normalisation d'adresses

- Capacités : géocodage, reverse-géocodage, clustering
  - Interfaces : API REST /geo
  - Justification : capacité native de la plateforme
- 
- **External Systems :**
    - Rôle : intégration avec services tiers (paiements, cartographie, notifications)
    - Justification : modularité et évolutivité

## 3.2 Domaines fonctionnels

Les domaines concernés incluent :

- Recherche géolocalisée.
- Catalogue et offres alimentaires.
- Gestion des fournisseurs et des consommateurs.
- Opérations de back-office.
- Capacités d'expérimentation (feature flags, variantes).
- Paiements (phase ultérieure).

## 3.3 Coexistence avec l'existant

La plateforme actuelle sera maintenue en mode maintenance.

La migration vers la nouvelle architecture se fera progressivement, fonctionnalité par fonctionnalité, selon un strangler pattern.

Cette approche réduit les risques, évite les interruptions de service et permet une montée en charge progressive.

## 4. Portée du Travail d'Architecture

### 4.1 Plan de mise en œuvre de haut niveau

La transformation se déroule en incrément successifs :

Incrément	Capacité livrée	Description
1	Géolocalisation	Calcul de distance, normalisation d'adresses, fondations GEO
2	Recherche locale	Service Search + intégration GEO, priorisation des résultats locaux
3	Catalogue & Offres	Consultation, filtrage, disponibilité, structuration des données d'offres
4	Comptes utilisateurs & fournisseurs	Création de comptes, préférences, gestion des profils
5	Back-Office & supervision	Modération, qualité, alertes, supervision opérationnelle
6	Expérimentation	Feature flags, variantes, tests A/B, activation progressive

Chaque incrément est autonome, testable, déployable indépendamment et introduit progressivement les capacités de l'architecture cible.

## 5. Parties Prenantes

Partie prenante	Rôle et responsabilités
Direction générale (CEO)	Porte la vision business, arbitre les priorités stratégiques et valide les orientations structurantes.
CPO	Définit les capacités métier, pilote la priorisation produit et garantit l'alignement avec la feuille de route.
CIO	Assure la gouvernance technique, l'alignement organisationnel et la cohérence des pratiques.
Responsable Ingénierie	Encadre les standards de développement, garantit la qualité d'implémentation et soutient les équipes techniques.
Équipes de développement	Conçoivent, implémentent et testent les services ; apportent un retour d'expérience continu.
Fonction Architecture	Porte la vision globale, définit les garde-fous, documente les décisions et accompagne les équipes.

## 6. Approche d'Architecture

### 6.1 Limites responsables

L'architecture FOOSUS impose des limites responsables permettant d'innover rapidement tout en préservant la cohérence globale du système. Ces limites définissent le cadre dans lequel les équipes peuvent évoluer en autonomie sans compromettre la stabilité ou la maintenabilité de la plateforme.

- Frontières de services strictes : chaque service couvre un périmètre fonctionnel clair et non ambigu.
- Découplage systématique : aucune dépendance directe entre services, uniquement via des API explicites.
- Réversibilité : les décisions structurantes doivent pouvoir être ajustées ou remplacées à faible coût.
- API stables et documentées : versionnement obligatoire, schémas contractuels, compatibilité ascendante.
- Observabilité obligatoire : logs structurés, métriques et traces pour chaque service.
- Déploiements sans interruption : CI/CD, feature flags, activation progressive.
- Simplicité : éviter les solutions complexes ou difficiles à maintenir dans une petite équipe.

Ces limites garantissent un cadre clair, soutenant l'innovation rapide sans générer de dette technique incontrôlée.

### 6.2 Gestion du changement et migration

La transformation de la plateforme FOOSUS repose sur une gestion du changement progressive, continue et compatible avec les pratiques Lean de l'entreprise. L'objectif est de réduire les risques, d'éviter les bascules brutales et de permettre une montée en maturité incrémentale.

- Migration progressive (Strangler Pattern) : remplacement des capacités existantes par incrément successifs.
- Coexistence contrôlée : l'ancienne plateforme reste en maintenance jusqu'à substitution complète.
- Feature flags : activation progressive des nouvelles fonctionnalités sans déploiement risqué.
- Déploiements continus : pipelines CI/CD automatisés, sans interruption perceptible.
- Validation continue : chaque incrément est mesuré via les SLO/SLI définis dans la SCRA.
- Apprentissage rapide : les décisions sont ajustées en fonction des retours utilisateurs et des métriques.
- Documentation légère : mise à jour continue des artefacts dans le dépôt d'architecture.

Cette approche garantit une transition fluide, maîtrisée et compatible avec les contraintes opérationnelles de FOOSUS.

Étape	Objectif	Description
1	Coexistence initiale	Maintien de l'ancienne plateforme en mode maintenance
2	Introduction des nouveaux services	Déploiement des services GEO, Search, Catalogue
3	Activation progressive	Feature flags pour basculer les flux réels
4	Substitution	Remplacement complet des fonctionnalités existantes
5	Retrait de l'ancien système	Extinction progressive de la plateforme historique

## 7. Livrables

Avant de pouvoir engager l'ensemble des équipes dans une trajectoire commune, il est essentiel de rendre l'architecture visible, compréhensible et partageable. Les livrables suivants constituent la base documentaire qui permettra à chacun — direction, produit, design et développement — de s'aligner sur une vision claire et d'avancer de manière cohérente.

- Déclaration de Travail d'Architecture
- Spécification des Conditions Requises pour l'Architecture
- Contrats architecturaux (business, design, développement)
- Vues d'architecture :
  - Vue de contexte
  - Vue des domaines
  - Vue micro-services
  - Vue de coexistence ancien/nouveau
  - Vue de déploiement
- Diagrammes UML locaux
- Modèle de gouvernance

## 8. Critères de réussite

Avant de pouvoir affirmer que l'architecture proposée remplit son rôle, il est nécessaire de définir des repères concrets et mesurables. Ces critères permettront d'évaluer objectivement les progrès réalisés et de valider que la plateforme soutient réellement la croissance et l'innovation attendues.

- Réduction des incidents critiques à moins d'un par mois.
- Lead time inférieur à une semaine.
- Mise en production d'une recherche géolocalisée opérationnelle dès le premier trimestre.
- Validation de l'architecture par les parties prenantes.
- Adoption des standards par les équipes.
- Premiers micro-services déployés sans interruption de service.

## 9. Risques et Atténuations

Avant de détailler les mesures d'atténuation, il est utile d'évaluer chaque risque selon une échelle simple. Le système de notation permet de prioriser les efforts et de rendre explicite le niveau d'attention requis pour chaque point sensible.

Échelle de notation :

- Probabilité : 1 (faible) → 5 (très élevée)
- Impact : 1 (faible) → 5 (très élevé)
- Score global = Probabilité × Impact
- 1–8 : risque maîtrisable
- 9–15 : risque significatif
- 16–25 : risque critique

Risque	Probabilité (1–5)	Impact (1–5)	Score	Stratégie d'atténuation
<b>Fragmentation technologique</b>	4	5	20	Standards clairs, accompagnement des équipes, revue régulière des pratiques.
<b>Résistance au changement</b>	3	5	15	Communication transparente, cadre non contraignant, implication progressive.
<b>Dette technique persistante</b>	5	5	25	Migration progressive, priorisation conjointe, visibilité sur la dette.
<b>Décisions difficiles à inverser</b>	3	5	15	Patterns réversibles, expérimentation contrôlée, documentation des choix.
<b>Charge de travail excessive</b>	3	4	12	Découpage en phases, priorisation stricte, synchronisation régulière.

## 10. Engagements

La réussite de cette initiative repose sur un engagement mutuel. La Fonction Architecture doit fournir un cadre clair et soutenant, tandis que les parties prenantes doivent contribuer activement à la cohérence et à la discipline nécessaires pour faire évoluer la plateforme.

La Fonction Architecture s'engage à :

- Fournir une vision claire et cohérente.
- Documenter les décisions structurantes.
- Accompagner les équipes dans la montée en maturité.
- Garantir la cohérence globale de la plateforme.

Les parties prenantes s'engagent à :

- Participer aux ateliers.
- Respecter les contrats architecturaux.
- Contribuer à la normalisation.
- Prioriser les travaux nécessaires à la santé du système.