



Déclaration de Travail d'Architecture

- *Projet : Approvisionnement alimentaire géographiquement responsable v 1.1 pour FOOSUS*
- *Préparé par : Maxime Cansell - Architecte Logiciel FOOSUS*

Information sur le document

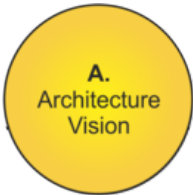
Livrable issu de la Phase TOGAF :	
Nom du projet	Approvisionnement alimentaire géographiquement responsable v 1.1
Préparé par :	Maxime Cansell - Architecte logiciel
N° de version du document :	0.1
Titre :	Déclaration de travail d'architecture
Date de version du document :	20/03/2021
Revu par :	Ash Callum - CEO
Date de révision :	
Liste de distribution :	
De :	
Date :	
Email :	
Pour Action :	
Date de rendu :	
Email :	
Types d'action :	Approbation
Historique de versions du document	https://github.com/SergeMax/OCP5.git

Table des Matières

1. Objet de ce document	4
2. Déclaration de travail d'architecture	5
Requête du projet et contexte	5
3. Objectifs et périmètre	6
Objectifs	6
Périmètre	7
Parties prenantes, préoccupations et visions	7
Tableau des vues TOGAF	9
Approche managériale	10
Procédures de changement de périmètre	11
4. Rôles et responsabilités	12
Structure de gouvernance	12
RACI	15
Process du projet	16
5. Approche architecturale	19
Vision architecturale cible	19
Pattern architecturale : Micro front-end et Micro-service	19
Business architecture baseline and target	20
Applicative architecture target	21
Data architecture target	22
Technology architecture target	22
Process d'architecture	24
Contenu de l'architecture	27
6. Plan de travail	29
Éléments de travail	29
Plan de communication	30
Plan et calendrier du projet	31
7. Risques et facteurs de réduction	32
Analyse des risques	32
Hypothèses	34
8. Critères d'acceptation et procédures	35
Métriques et KPIs	35
Procédure d'acceptation	36
9. Approbations signées	37
10. Annexes	38
11. Table des illustrations	39

1. Objet de ce document

Ce document est une Déclaration de travail d'architecture pour le projet "Approvisionnement alimentaire géographiquement responsable" de Foosus.

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 permet de mesurer la réussite de l'exécution du projet d'architecture et formera la base de l'accord contractuel entre le fournisseur et le consommateur de services d'architecture. Les informations de ce document se situent à un haut niveau.

La Déclaration de travail d'architecture peut être documentée sur le dépôt Git.

Ce document se base sur une implémentation TOGAF.

2. Déclaration de travail d'architecture

Requête du projet et contexte

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, notre solution technique complexe n'évolue plus au rythme de l'activité et risque d'entraver notre croissance. Les études de marché et les analyses commerciales montrent que nos clients souhaitent acheter localement et soutiennent les producteurs locaux.

Nos concurrents n'ont pas ciblé cette niche. Nous voulons nous appuyer sur les connaissances acquises ces trois dernières années et créer une plateforme basée sur la géolocalisation et qui mettra en contact des consommateurs avec des producteurs et des artisans locaux dans toutes les catégories de besoins.

Pour répondre à cet objectif, une nouvelle architecture SI est nécessaire. La déclaration de travail architectural en fait la description.

3. Objectifs et périmètre

Objectifs

Le tableau suivant liste les objectifs business de FOOSUS.

	Objectif Business	Notes
1	Améliorer sa compétitivité par rapport aux grandes entreprises d'e-commerce alimentaire internationales.	Créer une nouvelle plateforme d'e-commerce alimentaire
2	Soutenir la consommation de produits alimentaires locaux en mettant en contact les clients avec des producteurs et artisans de leur secteur.	Grâce à la géolocalisation
3	Augmenter le taux d'inscriptions utilisateurs	
4	Innover dans le périmètre d'une Architecture d'Entreprise	
5	Soutenir l'innovation technique rapide et l'expérimentation	
6	Améliorer la réputation de Foosus sur le marché	Diminuer les interruptions grâce à la stabilité du système



Tableau 1 : "Tableau des objectifs business"

Périmètre

Le périmètre de travail concerne la conception, la réalisation, la migration, la mise en production et la gouvernance de l'architecture de la nouvelle plateforme d'e-commerce alimentaire géolocalisée de Foosus.

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. Il se base sur le tableau (n°3) des différentes visions délivrées par TOGAF. Celui-ci est visible sur la page suivante. Il fournit la légende des différentes couleurs de chaque vision des parties prenantes au tableau 2.

Partie prenante	Préoccupation	Vision
CEO, CFO, le consortium d'investisseurs	Augmenter le taux d'inscriptions utilisateurs.	 Vue des fonctions commerciales Vue des personnes (organigramme) Vue stratégie commerciale et objectifs Vue des services commerciaux Vue de la mobilité d'entreprise Vue des objectifs commerciaux (objectifs SMART du scénario d'entreprise) Vue des entités de données (modèle de classe)
CIO, CPO, CFO	Innover dans le périmètre d'une Architecture d'Entreprise	Vue de la manageabilité de l'entreprise  Vues de l'architecture d'entreprise Vues d'architecture de données


		Vues de l'architecture des applications Vues de l'architecture technologique
CMO, CIO, CPO, CFO	Soutenir l'innovation technique rapide et l'expérimentation	Vue de l'ingénierie système 
CMO, CPO, Directeur des Opérations	Visibilité de la plateforme	Vue de la manageabilité de l'entreprise  Vue de la qualité des services 

Tableau 2 : "Tableau des vues des parties prenantes"

Tableau des vues TOGAF

To address the concerns of the following stakeholders...			
Users, Planners, Business Management	Database Designers and Administrators, System Engineers	System and Software Engineers	Acquirers, Operators, Administrators, & Managers
... the following views may be developed			
Business Architecture Views	Data Architecture Views	Applications Architecture Views	Technology Architecture Views
Business Function View	Data Entity View	Software Engineering View	Networked Computing/ Hardware View
Business Services View			
Business Process View			
Business Information View			
Business Locations View			
Business Logistics View	Data Flow View (Organization Data Use)	Applications Interoperability View	Communications Engineering View
People View (Organization Chart)			Processing View
Workflow View			
Usability View			
Business Strategy and Goals View			Logical Data View
Business Objectives View	Standards View		
Business Rules View			
Business Events View			
Business Performance View			
	System Engineering View		
Enterprise Security View			
Enterprise Manageability View			
Enterprise Quality of Service View			
Enterprise Mobility View			

Tableau 3 : "Tableau des vues TOGAF"

<https://pubs.opengroup.org/architecture/togaf8-doc/arch/chap31.html>

Approche managériale

L'approche managériale est une approche Lean. Nous reportons ici les principes de cette approche, tels que dans la section "Vision d'architecture" du chapitre : "Principes à refléter durant un processus d'architecture Lean"

Principes à refléter durant un processus d'architecture Lean

Principes généraux

- Décisions pilotées par le feed-back et l'apprentissage.
- Faire des choix qui soutiennent les objectifs à long terme.
- Accepter le fait que les erreurs se produisent.
- Nous assurer que nous concevons l'architecture pour échouer vite et nous améliorer.

Principes Business

- Soutenir l'innovation et l'agilité du business grâce à l'extensibilité
- Soutenir la réputation de la marque grâce à la stabilité

Principes 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.

Principes 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.

Principes technologiques

- Faire des choix ouverts et aisés à modifier.
- Les choix de construction versus achat doivent être raisonnés et toujours pris en compte.
- Les choix technologiques doivent s'aligner sur la capacité et la correspondance avec le business.
- Soutenir les sorties logiciel dès que possible.
- S'assurer que tous les composants de l'architecture sont conçus pour être faciles à cataloguer et à ne pas perdre de vue.
- Privilégier la prévisibilité et la répétabilité plutôt que le non-déterminisme.

Procédures de changement de périmètre

Si le périmètre du projet d'architecture venait à changer, il est nécessaire de respecter les étapes suivantes :

- 1) **Analyser la nature du changement** et sa portée sur les différentes vues architecturales.
- 2) **Lister les parties prenantes** intervenant dans ces vues.
- 3) **Effectuer un conseil** composé des parties prenantes visées au point précédent et déterminer si ce changement nécessite un nouveau tour complet de l'ADM ou si une adaptation (reteilloring) est suffisante. La décision finale appartient au CEO.
- 4) **Mettre en œuvre la modification ou débiter l'ADM.**
- 5) **Mettre à jour la documentation impactée.**

4. Rôles et responsabilités

Structure de gouvernance

Les cadres de l'entreprise sont représentés par le diagramme suivant avec à leur tête Ash Callum. Le diagramme complet est visible en annexe.

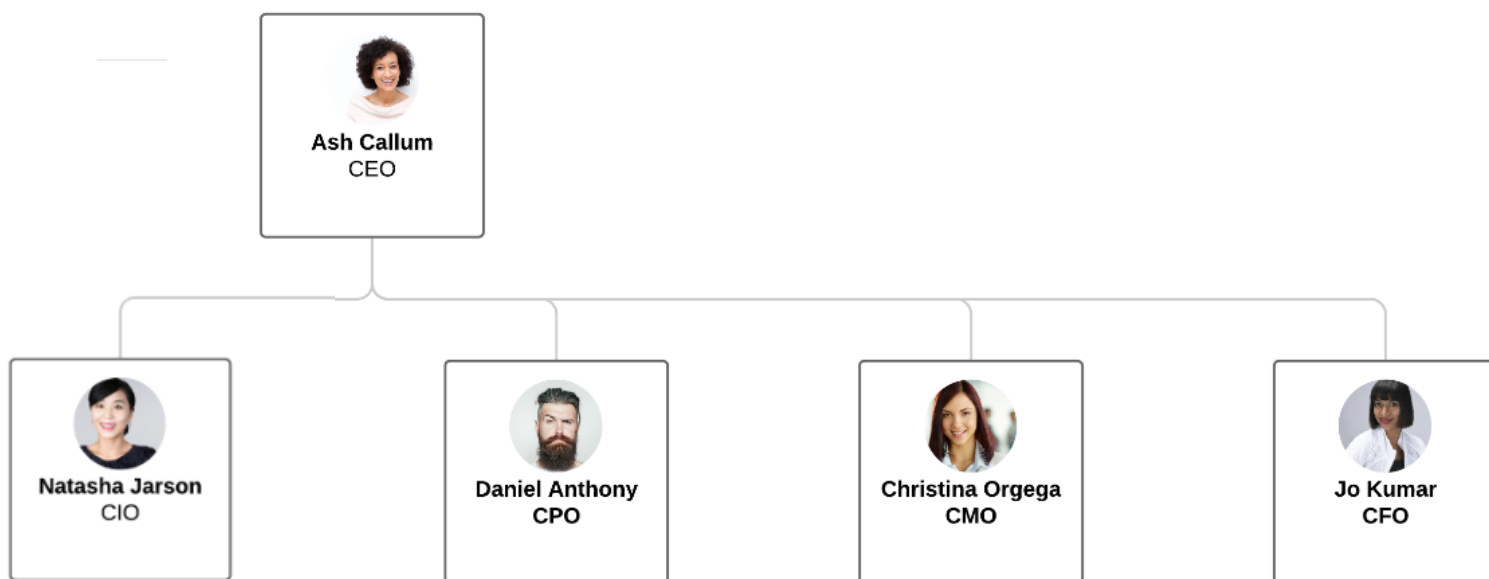


Diagramme 1 : “Cadres FOOSUS”



Ash Callum
CEO

Chief Executive Officer :

Ash Callum est la dirigeante de l'entreprise.

C'est elle qui établit les stratégies d'évolution et de développement de la structure, tant au point de vue comptable, financier, managérial que technique. Elle est en tête de l'ensemble des opérations et en donne les orientations à long et court termes.

Responsabilités :

- Représentant légal de l'entreprise.
- Préside le conseil d'administration.
- Conçoit et anime la stratégie de l'entreprise : définit ses objectifs généraux et décide en dernier ressort (arbitrage) des moyens financiers, matériels et humains à mettre en œuvre dans le cadre des orientations et des décisions du conseil d'administration ou de la maison mère.
- Anime le comité de direction de l'entreprise et est responsable de ses résultats.
- Rend compte au consortium d'investisseurs.



Natasha Jarson
CIO

Chief Information Officer

Natasha Jarson est chargée de choisir la structure informatique en fonction des besoins de l'entreprise. Elle et son équipe doivent comprendre, articuler et planifier l'utilisation des technologies existantes et émergentes pour élaborer un plan d'utilisation serein et ambitieux des technologies.

La CIO doit prévoir et adapter la stratégie numérique, tout en répondant aux besoins commerciaux.



Daniel Anthony
CPO

Chief Product Officer

Daniel Anthony accompagne, structure et accélère la croissance économique de l'entreprise. Il est notamment en charge de la stratégie du produit et supervise tous les éléments de celui-ci, de sa conceptualisation à ses performances de lancement.

Le CPO est spécialiste du Product Management. Il se charge de structurer et professionnaliser cette pratique. Il manage dans l'objectif de consolider son équipe et de la faire monter en compétence.



Jo Kumar
CFO

Chief Product Officer

Jo Kumar, directrice financier, est en charge de la planification économique et financière de l'entreprise. Décider quand et où investir, évaluer les risques, tout cela afin d'augmenter la valeur de l'entreprise. Apporte des connaissances financières, comptables et fournit un regard général et analytique sur l'entreprise. Elle est également le conseiller en affaires stratégiques du PDG.



Christina Orgega
CMO

Chief Marketing Officer

Christina Orgega directrice marketing est responsable des activités de marketing, qui comprennent la gestion des ventes, le développement de produits, la publicité, les études de marché et le service à la clientèle. Elle met l'accent sur le maintien d'un service de qualité par la communication, et aligne les intérêts de tous les départements pour optimiser les activités de marketing.

RACI

Le tableau RACI suivant attribue des rôles et des responsabilités pour chaque phase de la conception architecturale. Les **R** (responsables) sont les seules personnes habilitées au contrôle final et à la validation des différents livrables de la phase correspondante.



Ash Callum
CEO



Natasha Jarson
CIO



Daniel Anthony
CPO



Jo Kumar
CFO



Christina Ortega
CMO

Préliminaire	RA	A	A	A	A
A —Vision de l’architecture	RA	A	A	A	A
B —Architecture business	RA	A	A	A	A
C — Architecture des systèmes d’information	CI	R	A	CI	I
D — Architecture technologique	CI	R	CI	I	CI
E —Opportunités et solutions	R	A	A	A	I
F —Planning de migration	RA	A	A	A	I
G —Gouvernance de l’implémentation	RA	A	I	I	I
H —Management du changement d’architecture	RA	A	A	A	A
Management des conditions requises	R	A	A	A	A

R : Responsable.

A : Acteur.

C : Consulté

I : Informé

Tableau 4 : “RACI”

Process du projet

- Agile :

Le développement suivra une méthodologie dite agile qui permet de mieux prendre en compte les évolutions du besoin, et limite l'impact du changement sur le coût et le temps du projet. Elle repose sur une meilleure implication de Product Owner (client) dans le processus de création, ainsi que sur un découpage du projet par fonctionnalités livrées régulièrement. Pour plus d'informations, nous recommandons la lecture du manifeste agile : <https://manifesteagile.fr/>

- Présence aux réunions



Réunions planification	X	X	-	X	-
Validations post conception	X	X	X	X	X
Phases de tests	-	X	X	-	-
Réunions démo	X	X	X	X	-
Livraison fonctionnalité globale = réunion démo + 2j	X	X	X	X	X

Tableau 5 : "Tableau des réunions"

- Répertoire de documents

Le répertoire de document se trouvera sur un dépôt Git déployé sur un serveur local privé. Dans un souci de confidentialité et d'espionnage industriel les documents sensibles ne devront pas être stockés dans le cloud sauf exception si indispensable ; dans un cloud de l'Union européenne.

Url : <https://github.com/SergeMax/OCP5.git>

- Management de la configuration :

Nous adopterons une démarche DevOps grâce au pipeline CI/CD dont voici les phases :

- Build : compilation de l'application.
- Test : test du code. L'automatisation permet ici d'épargner du temps et des efforts.
- Lancement : distribution de l'application au référentiel.
- Déploiement : déploiement du code en production.
- Validation et conformité : ces étapes de validation sont à adapter en fonction des besoins de votre entreprise. Des outils d'analyse de la sécurité des images tels que Clair peuvent garantir la qualité des images en les comparant à des vulnérabilités (CVE) connues.

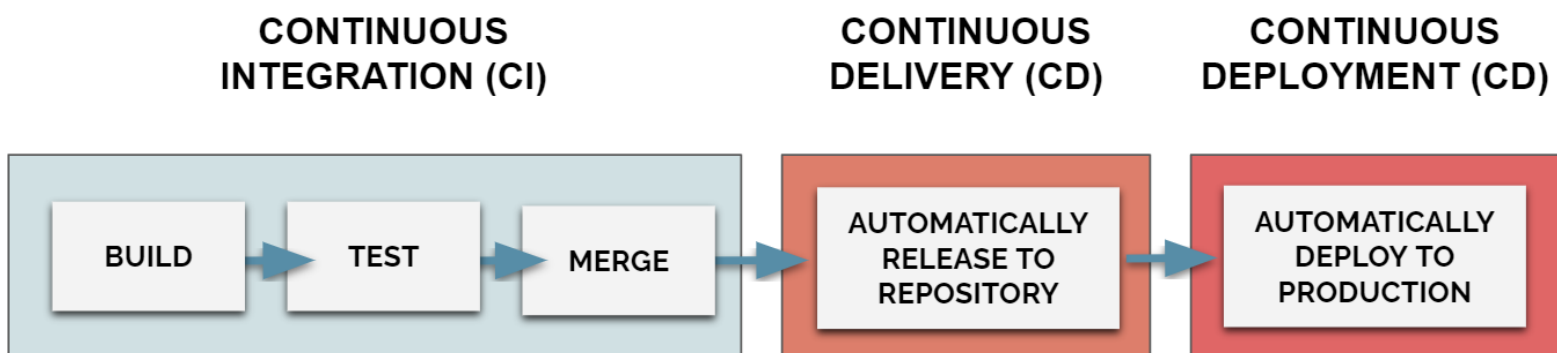


Diagramme 2 : "Pipeline CI-CD"

- **Assurance qualité :**

Les tests automatisés et d'intégrités seront réalisés par une équipe indépendante.

- **Procédure en cas d'escalade :**

En cas d'escalade, le premier niveau, N1, sera composé de l'équipe de support.

N2 sera composé du CIO et de l'architecte logiciel et du CPO si besoin.

N3 réunira les niveaux précédents avec le CEO.

- **Procédure en cas de changement :**

Se référer à la partie 3. "Procédure de changement de périmètre"

5. Approche architecturale

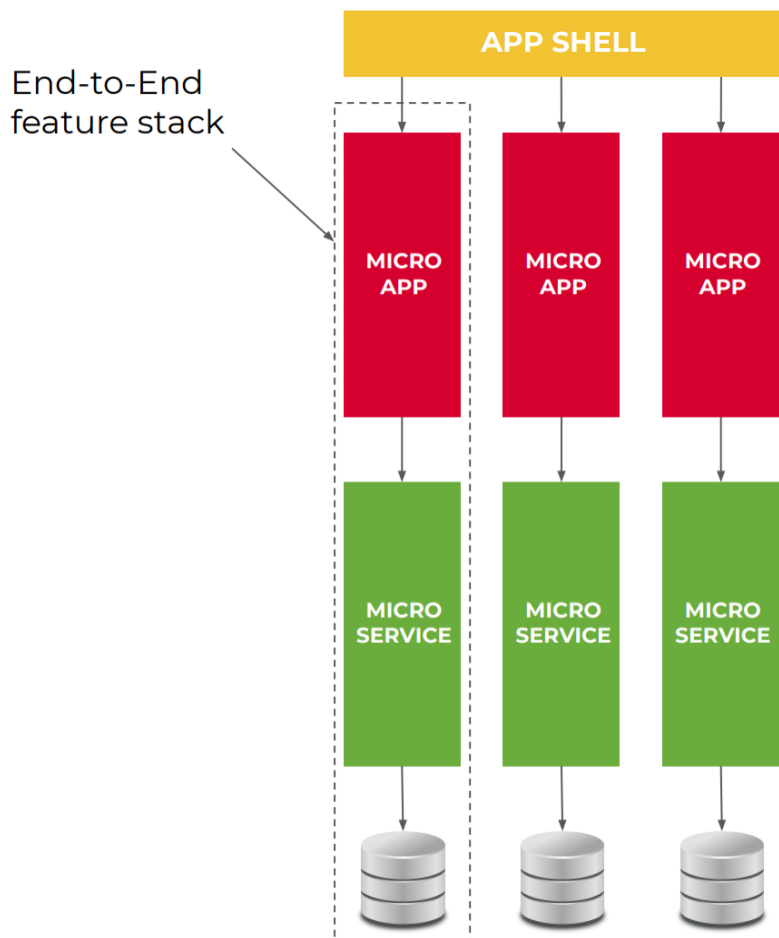
Vision architecturale cible

- Pattern architectural : Micro front-end et Micro-service

L'objectif business d'innovation technique rapide et d'expérimentation nécessite d'avoir une architecture facilement évolutive. La phase d'étude préalable de notre équipe a établi que l'architecture devra s'orienter vers le pattern micro-services (cf. Brief business de haut niveau).

Nous poussons cette recommandation au front-end avec une architecture micro-front end. Les avantages du micro-frontend sont les mêmes que ceux du micro-service ;

- Base de code réduite pour chaque micro-frontend.
- Tests unitaires plus aisés.
- Développement parallèle facilité (équipe verticale pour chaque fonctionnalité).
- Déploiement rapide et facilité.
- Tests parallèles de comparaison de nouvelles fonctionnalités en production, validation ou retrait simplifié.



Dans un souci de compatibilité et de débogage il est recommandé, pour le micro-frontend, de garder la même stack technologique. Notre équipe souhaitant harmoniser ses pratiques nous utiliseront également la même stack technologique pour les micro-services.

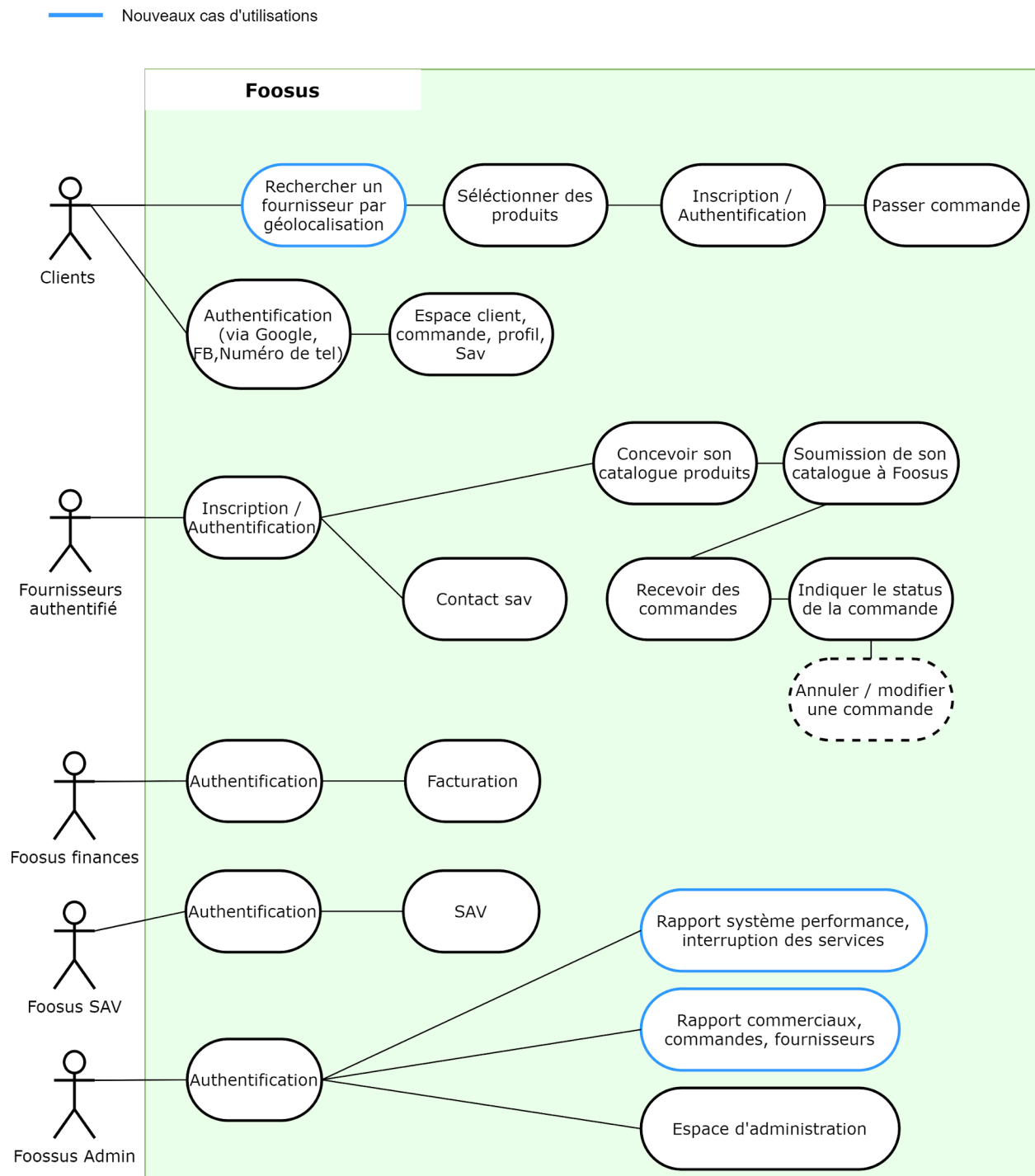
L'architecture cible Business, Applicative, Data et technologique sera précisée et définie dans le document "Définition d'architecture" résultant des phases B, C et D.

Diagramme n°3 : "Vision Architecturale"

- Business architecture baseline and target :

L'architecture métier cible (nouveaux cas indiqués en bleu) disposera des mêmes fonctionnalités que celle actuelle en y ajoutant la recherche des fournisseurs par géolocalisation ainsi que les rapports. Voici la version préliminaire de haut niveau :

Diagramme n°4 : "Cas d'utilisations métiers"



- Applicative architecture target :

Voici le diagramme des composants applicatifs.

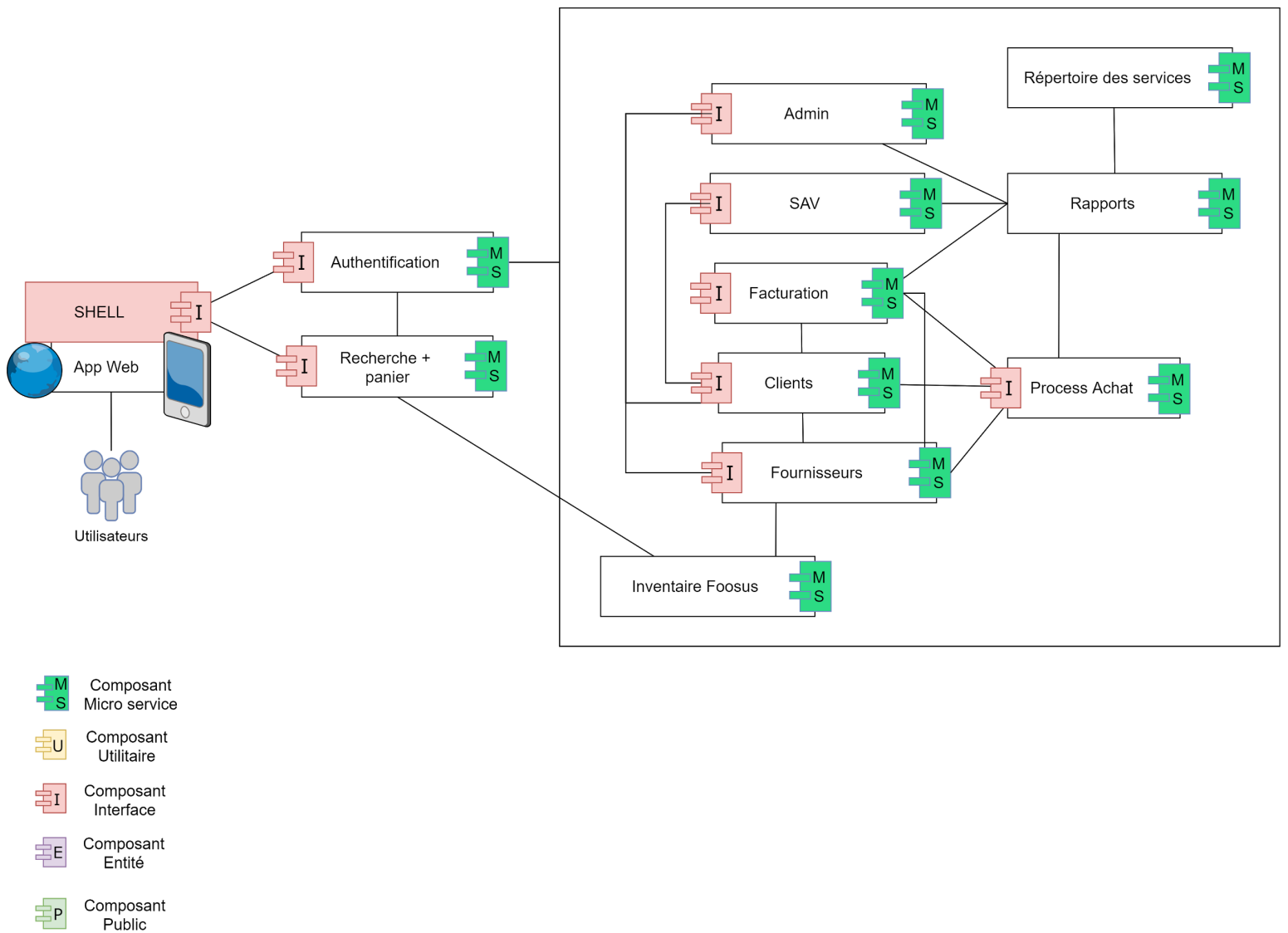


Diagramme n°5 : “Diagrammes des composants”

- Data architecture target :

Nous ne disposons pas à ce stade du dictionnaire de données baseline nous permettant d'établir une vision préliminaire de l'architecture Data. Nous devons récolter ces informations pour réaliser l'architecture des données lors de la phase de définition d'architecture.

Nous pouvons cependant prévoir l'ajout de ces catégories de données :

Localisation :

- Adresse et coordonnée GPS des fournisseurs et clients.
- Périmètre de recherche

Données de rapport :

- Différentes données commerciales souhaitées.
- Taux de disponibilité des services
- Temps de réponse moyen des API

- Technology architecture target :

Frontend :

Micro front en Angular 11 (à compléter justifier?)



Angular est un des leaders parmi les framework Frontend. Il utilise le langage Typescript (du Javascript typé) et permet de créer des "Single Page Application" (SPA).

Une SPA est une application web accessible via une page web unique. Le but est d'éviter le chargement d'une nouvelle page à chaque action demandée, et de fluidifier ainsi l'expérience utilisateur. Il se base sur les web-component et modules. Le développement parallèle, en équipe et micro-front end en est facilité.

Backend :

Spring Boot



Spring Boot

Spring boot est un framework Backend en langage JAVA. Les applications Spring Boot sont une avancée majeure dans le développement Java et l'utilisation du framework Spring.

Spring Boot permet de concevoir, de façon facilitée, une architecture évolutive logicielle dite de "Micro-services".

Pour la géolocalisation lors des recherches de fournisseurs par le client nous pouvons conseiller l'utilisation de l'API MaxMind GeoIP2 Java.

Base de données :



SQL

Le système reposera sur une base de données SQL avec comme système de gestion MySQL.

Une base de donnée NoSQL comme MongoDB a l'avantage d'être rapide sur des bases contenant plusieurs dizaines de millions de données déstructurées (avec peu de relation entre elles) mais dès qu'il y a des requêtes SQL complexes avec des tables ayant diverses relations cela s'avère plus compliqué.

- Infrastructures :

Les options de déploiement doivent encore être analysées avec le CEO, le CIO et le CFO pour choisir la meilleure option autant sur le plan technique que financier.

A ce stade nous pouvons recommander un déploiement dans le cloud avec AWS et Kubernetes. Duo très largement majoritaire dans l'IT.

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	H	A	
A —Vision de l'architecture	Préliminaire	B	
B —Architecture business	A	C	Baseline + Target architecture
C — Architecture des systèmes d'information	B	D	Baseline + Target architecture Baseline + Target architecture
D — Architecture technologique	C	F	Target architecture
E —Opportunités et solutions			
F —Planning de migration	D	G	
G —Gouvernance de l'implémentation	F	H	
H —Management du changement d'architecture	G	Préliminaire	
Management des conditions requises		Préliminaire,	

Tableau 6 : “Process d'architecture”

- Itérations

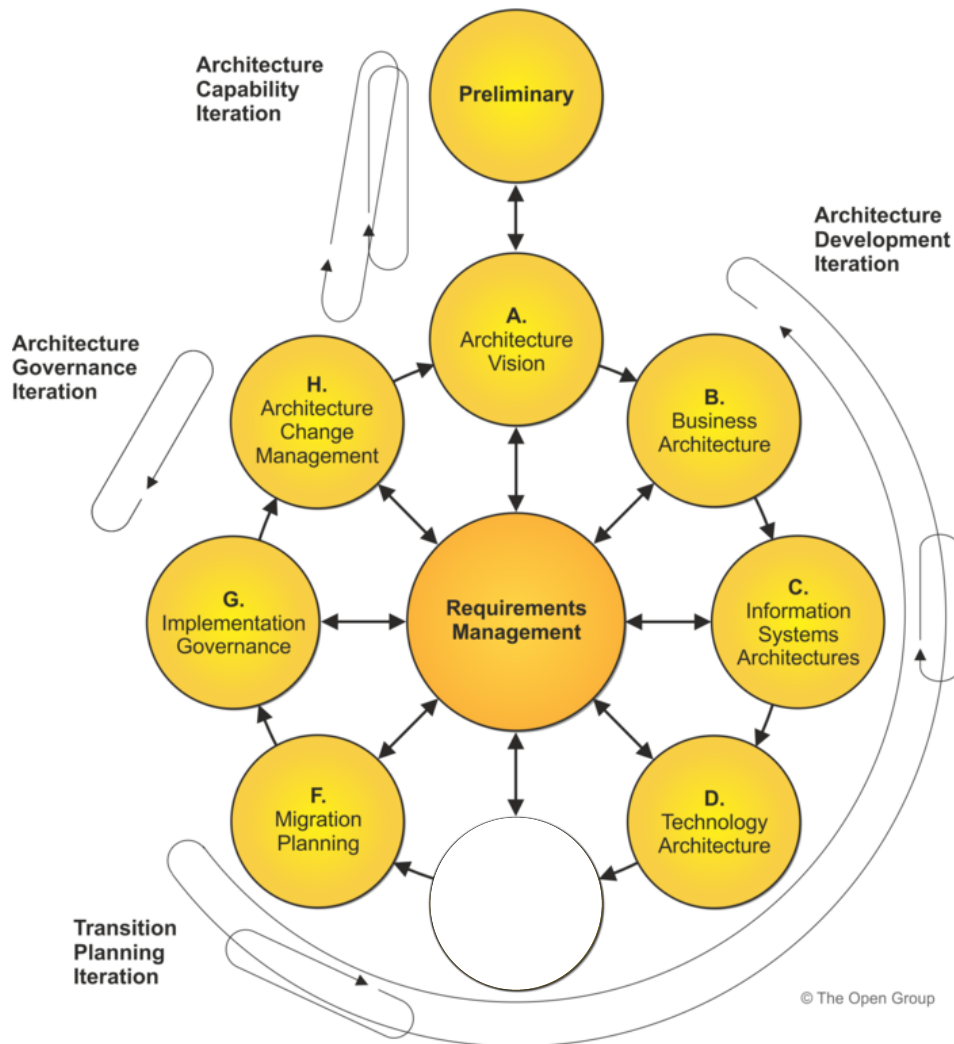


Diagramme 6 : “Itération ADM TOGAF”

<https://pubs.opengroup.org/architecture/togaf9-doc/arch/chap18.html>

Des itérations seront possible entre les phases suivantes (cf diagramme ci-dessus) :

Les itérations de capacité d'architecture :

Entre Préliminaire et A et entre H et Préliminaire.

Les itérations de la capacité d'architecture soutiennent la création et l'évolution de la capacité d'architecture requise.

Cela comprend la mobilisation initiale de l'activité d'architecture pour un objectif ou un type de mission d'architecture donné en établissant ou en ajustant l'approche, les principes, la portée, la vision et la gouvernance de l'architecture.

Les itérations de développement d'architecture :

Sur F.

Elles permettent la création de contenu d'architecture en parcourant ou en intégrant les phases Business, de systèmes d'information et d'architecture technologique.

Ces itérations garantissent que l'architecture est considérée comme un tout. Au fur et à mesure que les itérations convergent vers une cible, des extensions dans la phases Planification de la migration garantissent que l'implémentabilité de l'architecture est prise en compte au fur et à mesure que l'architecture est finalisée.

Les itérations de planification de la transition :

Entre E et F.

Les itérations de planification de la transition permettent de créer des feuilles de route formelles pour une architecture définie.

Les itérations de gouvernance d'architecture :

Entre G et H.

Les itérations de gouvernance d'architecture soutiennent la gouvernance de l'activité de changement progressant vers une architecture cible définie.

Contenu de l'architecture

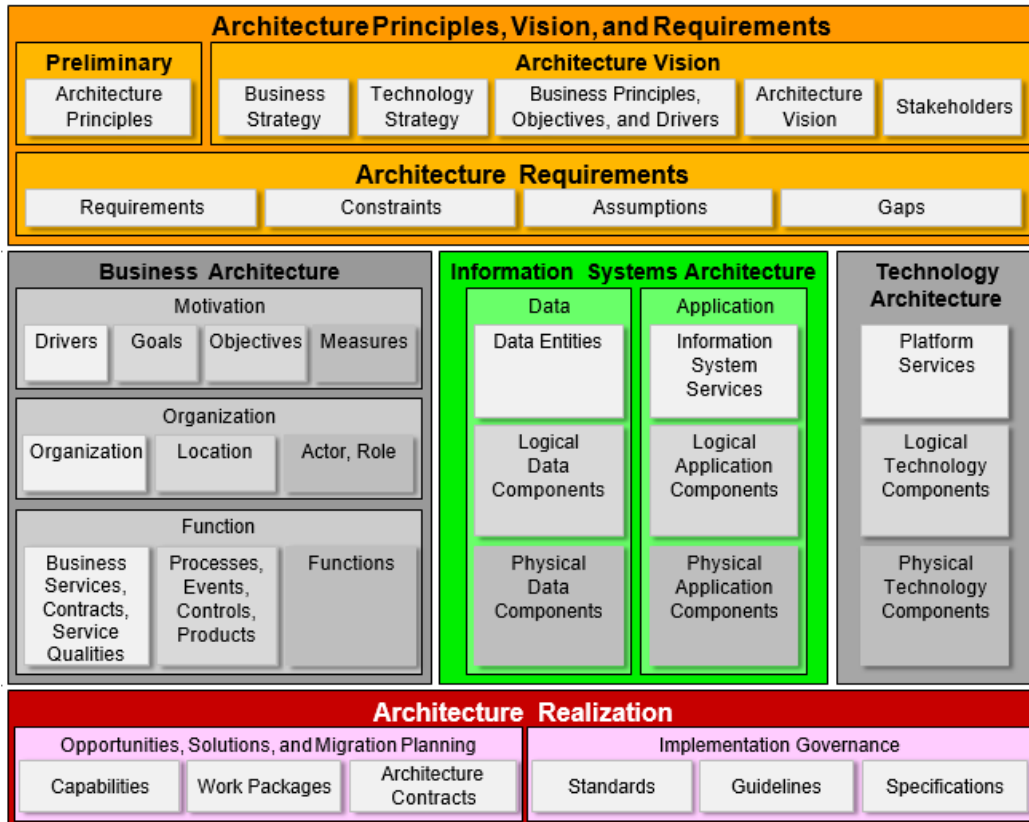


Tableau 7 : “Contenu d’architecture TOGAF”

<https://www.vanharen.net/blog/using-the-togaf-9-1-framework-with-the-archimate-3-0-modeling-language/>

Le cadre de contenu d’architecture TOGAF (ou ACF pour « Architecture Content Framework ») visible ci-dessus, 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 (n°8) sur la page suivante décrit les zones de contenu pertinentes pour ce projet spécifique.

Zone de contenu	Contenu personnalisé
Principes, Vision, et Conditions requises de l'Architecture	<p>Préliminaire : Principes d'architecture</p> <p>Vision d'architecture : Stratégie métier, stratégie technologie, Principes métiers, objectifs et conduite, Vision d'architecture, Partie prenante</p> <p>Exigences : Exigences, contraintes, hypothèses, écarts.</p>
Architecture Business	<p>Motivation : But, Objectifs, Mesure</p> <p>Organisation : Organisation des unités, Acteurs et rôles</p> <p>Fonctionnalités : Business, Services, contrats, service qualité, Processus, Événements, Contrôles, Produits</p>
Architecture des systèmes d'information — Données	<p>Data : Entités, Composants logique de données, Composant physique de données</p>
Architecture des systèmes d'information — Applications	<p>Application : Service des systèmes d'information, Composants logique d'application, composant physique d'application.</p>
Architecture technologique	<p>Service des plateforme (infrastructure), composants technologique logique, composants technologique physique. Infrastructures.</p>
Réalisation de l'architecture	<p>Opportunités, solution et planning de migration : Capacités, ensemble des travaux, contrats d'architecture</p> <p>Gouvernance de l'implémentation : Standards, Guidelines, spécifications</p>

Tableau 8 : “Contenu d'architecture TOGAF sur mesure”

6. Plan de travail

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

Éléments de travail

	Livrable	Réalisé	Sortie de ...	Entrée de ...
1	Architecture Repository	X	Préliminaire	Préliminaire, A, B, C, D, E, F, G, H, Gestion des exigences
2	Requirements impact analysis		Gestion des exigences	Gestion des exigences
3	Request for Architectural Work	X	Préliminaire, F, H	A, G
4	Architecture Vision	X	A, E	B, C, D, E, F, G, H, Gestion des exigences
5	High Level Business Requirements Brief	X	Préliminaire, F, H	A
6	Statement of Architecture Work	X	A, B, C, D, E, F, G, H	B, C, D, E, F, G, H, Gestion des exigences
7	Architecture Contract Development	X	-	-
8	Architecture Contract With Business	X	-	-
9	New Value Stream Map		B	C, D, E, F, G, H
10	Architecture Définition		B, C, D, E, F	C, D, E, F, G, H
12	Évaluation de la conformité		G	H
13	Plan de mise en œuvre et de migration		E, F	F
14	Modèle de gouvernance de la mise en œuvre		F	G, H

Tableau 9 : "Éléments de travail"

Plan de communication

Évènements	Canaux	Formats
Décisions sur pilotage du projet	Tableau de bord de pilotage Trello Orale + Mail(si besoin)	-
Suivi budgétaire	Plan prévisionnel et comptable	.Doc
Suivi du planning	Burndown chart	.Doc
Suivi des livrables	Git par CIO	Dépot git
Retards et autres difficultés rencontrées	Mail	Ecrit
Mise à jour des risques	Git - Ce document	.Doc
Prises de décisions		-

Tableau 10 : “Plan de communication”

Plan et calendrier du projet

Le diagramme présenté ici est le plan de mise en œuvre de haut niveau. Une fois en production, la nouvelle solution devra coexister avec le système actuellement en place le temps de s'assurer que tout est fonctionnel. Après la migration des derniers éléments, il faudra arrêter le système actuel.

Une période de 6 mois est prévue pour définir l'architecture (analyse et conception).

La problématique d'intégrité des données résultant de deux systèmes fonctionnant en parallèle doit être traitée dans le document définition d'architecture chapitre architecture DATA. Nous ne pouvons traiter actuellement cette problématique faute de données.

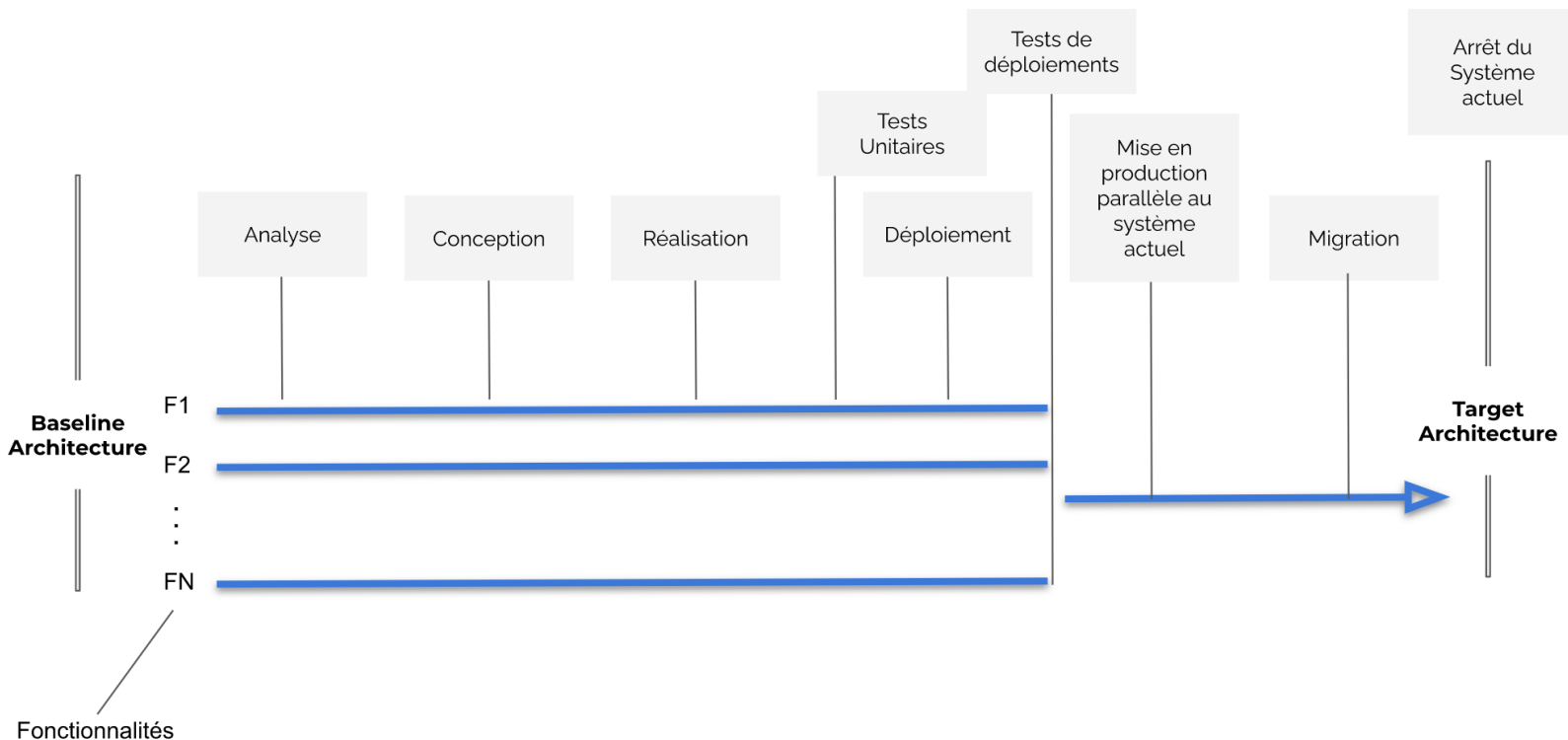


Diagramme 7 : "Plan calendrier projet"

7. Risques et facteurs de réduction

Analyse des risques

Description	Gravité 1-4	Fréquence 1-4	Criticité	Resp	Prévention	Réparation / Plan B	Veille / Détection anticipée
Mauvaise planification : Budget et délais insuffisants	4	3	12	CEO	<ul style="list-style-type: none"> - Validation du budget et des livrables lors de la phase découverte pré-projet. 	<ul style="list-style-type: none"> - Améliorer la communication et la gestion des parties prenantes. - Redéfinir le projet : réduire sa portée. - Justifier de nouveau son budget. - Ajouter et/ou supprimer des ressources. (Bien mesurer car ajouter accroît le retard "Loi de Brooks") - Résoudre les problèmes techniques difficiles. - Mise en place du plan de redressement. - Consulting. 	<ul style="list-style-type: none"> - Suivi de l'avancement grâce à une Burndown Chart. - Surveiller l'avancement des livrables à chaque réunion. - Surveiller la maîtrise du budget.
Spécifications ambiguës	3	3	9	CEO, CIO	<ul style="list-style-type: none"> - Phase de découverte pré-projet et contact étroit avec les parties prenantes. - Validation à chaque début de réunion de Planning. 	<ul style="list-style-type: none"> - Réajustement à chaque Sprint Planning. 	<ul style="list-style-type: none"> - Faire valider par le CTO les maquettes et les Users Stories dès qu'elles sont réalisées.
Fuite d'information / Manque de sécurité	4	2	8	CIO, Architecte Logiciel	<ul style="list-style-type: none"> - Consulting d'expert en sécurité dès la conception. - Mise à jour régulière des systèmes. 	<ul style="list-style-type: none"> - Identifier ou faire identifier les failles qui ont conduit à la fuite. - Identifier les données qui ont fuité. - Corriger les failles. 	<ul style="list-style-type: none"> - Se tenir informé des failles détectées par les concepteurs des technos utilisées.
Demandes de changements au cours du projet	2	3	6	CEO	<ul style="list-style-type: none"> - Phase de découverte avec contact étroit avec les parties prenantes entre le 1er et le 4 décembre. - Validation à chaque début de réunion de Planning. 	<ul style="list-style-type: none"> - Réajustement à chaque réunion de Planning. 	<ul style="list-style-type: none"> - Faire valider par le CTO les maquettes et les Users Stories dès qu'elles sont réalisées.

Absence du responsable aux jalons de validation	4	1	4	CEO	<ul style="list-style-type: none"> - Engagement sur les dates dans le cahier des charges. 	<ul style="list-style-type: none"> - Prévenir au plus tôt en cas d'absence. - Reporter la réunion. - Décaler les livrables. 	<ul style="list-style-type: none"> - Planifier les réunions.
Mauvaise utilisation des TIC (complexité technique)	2	2	4	Architecte Logiciel	<ul style="list-style-type: none"> - S'assurer de l'expérience des professionnels sur les technologies qu'ils auront à utiliser. - Plan de formation sur les lacunes identifiées. 	<ul style="list-style-type: none"> - Paire programming en cas de blocage. - Inviter à la formation. - Consulting si besoin. - Réaffecter le personnel. 	<ul style="list-style-type: none"> - Suivi des réalisations. - Revue de code. - Réunion quotidienne avec le Scrum Master pour identifier les problèmes.
Absence des Testeurs aux réunions, Meeting post conception, réunion de démo	2	1	2	CEO, CIO	<ul style="list-style-type: none"> - Engagement sur les dates dans le cahier des charges. 	<ul style="list-style-type: none"> - Prévenir l'équipe en charge du projet au plus tôt en cas d'absence. - Faire un compte rendu par les personnes de Velocity présentes. - Décaler les livrables. 	<ul style="list-style-type: none"> - Planifier les réunions.
Mauvaise installation	2	1	2	Architecte Logiciel	<ul style="list-style-type: none"> - Analyser les outils nécessaires aux technologies utilisées lors de la phase de conception. 	<ul style="list-style-type: none"> - Corriger les installations défectueuses. 	<ul style="list-style-type: none"> - Questionner, identifier et faire remonter les problèmes lors des réunions.

Tableau 11 : "Analyse des risques"

Hypothèses

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

Tableau 12 : "Hypothèses"

ID	Hypothèse
1.	Plutôt que d'investir davantage dans la plateforme existante, nous la conserverons en mode de maintenance. Aucune nouvelle fonctionnalité ne sera développée.
2.	La nouvelle architecture sera construite en fonction des technologies actuelles et avec la capacité de s'adapter à de nouvelles technologies lorsque celles-ci seront disponibles.
3.	Les équipes étant attachées à la plateforme existante, les dirigeants devront éviter de prendre de faux raccourcis en intégrant un nouveau comportement dans le système existant.
4.	L'offre initiale impliquera la coexistence de deux plateformes et la montée en puissance empirique du volume d'utilisateurs qui migreront vers la nouvelle plateforme à mesure que le produit évoluera. Cette augmentation sera proportionnelle à l'évolution des fonctionnalités.
5.	Par exemple, les utilisateurs précoces pourront choisir d'utiliser les nouvelles fonctionnalités de recherche intégrées au processus de paiement existant.
6.	La géolocalisation, si elle est modélisée suffisamment tôt dans la nouvelle plateforme, permettra d'introduire d'autres innovations en fonction de l'emplacement de l'utilisateur ou du fournisseur alimentaire.
7.	L'élaboration sur mesure d'une approche architecturale de type « lean » pourra contribuer à la réalisation de cette feuille de route, ce qui évitera de priver les équipes de leur autonomie et de compromettre la rapidité des cycles de versions.

8. Critères d'acceptation et procédures

Métriques et KPIs

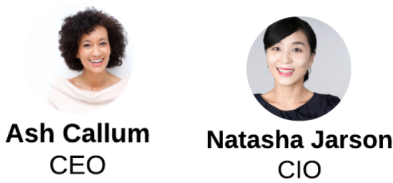
Les métriques suivantes seront utilisées pour déterminer le succès de ce travail d'architecture :

Métrique	Technique de mesure	Valeur cible	Justification
Nombre d'adhésions d'utilisateurs par jour	Rapport BDD	Augmentation de 10%	Indice de bonne santé de l'entreprise
Nombre d'adhésion de producteurs alimentaires	Rapport BDD	Passer de 1,4/mois à 4/mois	Indice de la réputation de l'entreprise auprès des producteurs
Délai moyen de parution d'une nouvelle fonctionnalité	Calcul par CIO	Réduit de 3,5 semaines à moins d'une semaine	Indice d'agilité de l'architecture
Taux d'incidents de production P1	A définir	Pour commencer : réduit de >25/mois à moins de 1/mois	Indice de stabilité de la plateforme. Importante pour l'image de l'entreprise

Tableau 13 : "KPIs"

Procédure d'acceptation

Chaque livrable appartient à une phase Togaf (cf page 2 “Information sur le document”). La partie prenante responsable de la phase est la seule habilitée au contrôle final et à la validation du livrable en question.



Préliminaire	X	
A —Vision de l’architecture	X	
B —Architecture business	X	
C — Architecture des systèmes d’information		X
D — Architecture technologique		X
E —Opportunités et solutions	X	
F —Planning de migration	X	
G —Gouvernance de l’implémentation	X	
H —Management du changement d’architecture	X	
Management des conditions requises	X	

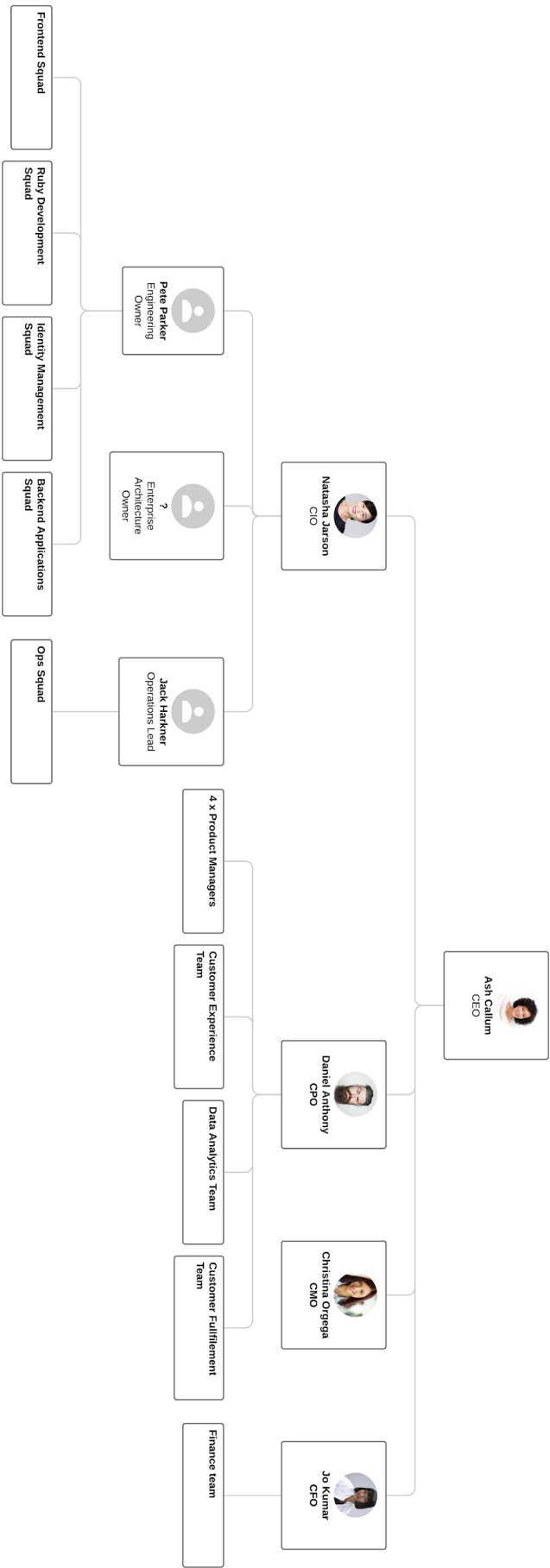
Tableau 14 : “Validation des livrables par phase”

9. Approbations signées

Valideur	Domaine de responsabilité	Date
Ash Callum	CEO	

10. Annexes

Diagramme du personnel



11. Table des illustrations

Tableaux

- Tableau 1 : “Tableau des objectifs business” p.5
- Tableau 2 : “Tableau des vues des parties prenantes” p.6
- Tableau 3 : “Tableau des vues TOGAF” p.8
- Tableau 4 : “RACI” p.14
- Tableau 5 : “Tableau des réunions” p.15
- Tableau 6 : “Process d’architecture” p.23
- Tableau 7 : “Contenu d’architecture TOGAF” p.26
- Tableau 8 : “Contenu d’architecture TOGAF sur mesure” p.27
- Tableau 9 : “Éléments de travail” p.28
- Tableau 10 : “Plan de communication” p.29
- Tableau 11 : “Analyse des risques” p.32
- Tableau 12 : “Hypothèses” p.33
- Tableau 13 : “KPIs” p.34
- Tableau 14 : “Validation des livrables par phase” p.35

Figures

- Figure 1 - Diagramme 1 : “Cadres FOOSUS” p.11
- Figure 2 - Diagramme 2 : “Pipeline CI-CD” p.16
- Figure 3 - Diagramme 3 : “Vision Architecturale” p.18
- Figure 4 - Diagramme 4 : “Cas d’utilisations métiers” p.19
- Figure 5 - Diagramme 5 : “Diagrammes des composants” p.20
- Figure 6 - Diagramme 6 : “Itération ADM TOGAF” p.24
- Figure 7 - Diagramme 7 : “Plan calendrier projet” p.30