



Déclaration de Travail d'Architecture

Projet :

**État Cible de l'Architecture et implémentation d'une nouvelle
architecture**

Rédaction : Nicolas Cervantes
Valideur : Eric Legba



Table des Matières

- 1. Objet de ce document**
- 2. Déclaration de travail d'architecture**
 - 2.1 Requête du projet et contexte
 - 2.2 Description du projet et périmètre
 - 2.3 Vue d'ensemble
 - 2.4 Alignement stratégique
- 3. Objectifs et périmètre**
 - 3.1 Objectifs business
 - 3.2 Parties prenantes, préoccupations et visions
 - 3.3 Approche managériale
 - 3.4 Procédures de changement de périmètre
- 4. Rôles et responsabilités**
 - 4.1 Structure de gouvernance
 - 4.2 Process du projet
- 5. Approche architecturale**
- 6. Contenu de l'architecture**
- 7. Procédures d'acceptation**
- 8. Approbations signées**



Information sur le document

1. Objet de ce document

Ce document est une Déclaration de travail d'architecture pour l'état Cible de l'Architecture et implémentation d'une nouvelle architecture pour l'entreprise 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 constitue habituellement le document qui permet de mesurer la réussite de l'exécution du projet d'architecture et peut former la base de l'accord contractuel entre le fournisseur et le consommateur de services d'architecture. En général, toutes les informations de ce document doivent se situer à un haut niveau.

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

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

2. Déclaration de travail d'architecture

Requête du projet et contexte

Foosus, startup spécialisée dans le domaine de l'alimentation durable, l'intérêt principale de l'entreprise est de pouvoir intégrer un contact plus direct entre les consommateurs et les artisans et producteurs locaux

Une limite sur les capacités de la plateforme a été atteinte et ne pourra pas permettre d'arriver aux objectifs métier visés pour les prochaines étapes de croissance. L'architecture actuelle est beaucoup trop complexe et difficile à maintenir, à l'état actuel elle ne pourra pas répondre à la croissance.

Afin de récupérer et reprendre le temps investi dans cette solution, les équipes techniques de Foosus devront trouver et mettre en place une architecture qui permettra l'innovation et le bon développement des futurs projets d'expansion de l'entreprise.



Description du projet et périmètre

Architecture

1. L'architecture devra être capable de s'adapter aux demandes techniques et du métier, celle-ci se réalisera au niveau de la scalabilité des requêtes faites par le client et fournisseurs.
2. La pile technologique évoluera au même rythme que les besoins des clients.

Maintenance

3. Privilégier la suppression d'interruption du service au moments des déploiements et migrations.

Accès

4. Les fournisseurs et clients (consommateurs) ne devront faire face dans aucun cas d'une limitation d'accès aux services, celle-ci devra être accessible dans tout type de réseaux et différences au niveau de la bande passante.

Profils

5. Plusieurs types de profils devront être mis en place afin de soutenir la variété d'utilisateurs, voir fournisseurs, métier ou consommateurs.

Sécurité

6. La plateforme et services feront face à une utilisation majeur de la part des utilisateurs, des couches de sécurité devront être implémentés afin d'établir la confiance, sécurité et l'image de l'entreprise.

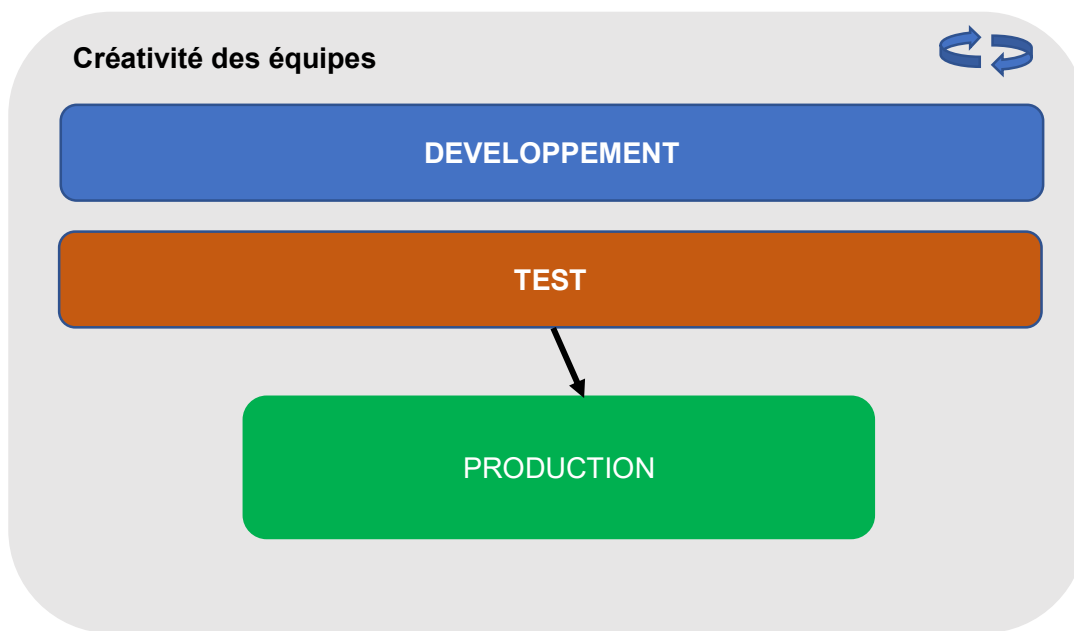
Equipes

7. Pour permettre aux équipes de travailler de façon indépendante et en ayant une liberté d'innovation, les systèmes devront avoir une facilité de modifications et intégrations entre partenaires.

Documentation

8. Les livrables seront produits en intervalles réguliers, cela permettra d'avoir un système à jour et innovant.

Alignement stratégique



3. Objectifs et périmètre

2.1 Objectifs business

1. Soutenir la consommation de produits alimentaires locaux
2. Mettre en contact les clients avec des producteurs et artisans locaux
3. Etablir un nouveau point de contact avec les clients en créant une nouvelle plateforme d'e-commerce

Parties prenantes, préoccupations et visions

Déclaration de Travail d'Architecture

V.D: 0.1

Client : Foosus



Date rev: 15/09/2021

Parties prenantes, préoccupations et visions		
Partie prenante	Préoccupation	Vision
Ash Callum CEO	Business	1. Innovation 2. Croissance 3. Communication 4. Améliorer les résultats et maximiser la rentabilité à long terme
	Data	
	Innovation technique rapide et expérimentation	
Natasha Jarson CIO	Business	1. Innovation 2. Enthousiasmer les équipes 3. Différencier la position sur le marché 4. Déploiement et gestion de systèmes maintenables, scalable, performant et sûr.
	Applicative	
	Technologique	
	Data	
	Innovation dans le périmètre d'une architecture	
	Innovation technique rapide et expérimentation	
Daniel Anthony CPO	Data	1. Innovation 2. Améliorer l'expérience du produit. 3. Stratégie du produit 4. Management de l'équipe produits 5. Rapports basés sur les données
	Business	
	Visibilité de la plateforme	
	Innovation dans le périmètre d'une architecture	
	Innovation technique rapide et expérimentation	
Christina Ortega CMO	Business	1. Fixer des buts et objectifs marketing 2. Planifier, mettre en œuvre et gérer la stratégie marketing 3. Déterminer les KPI pour le service marketing 4. Créer et présenter le plan et la stratégie marketing annuels 5. Utiliser des données et des rapports pour prendre des décisions fondées sur des preuves
	Visibilité de la plateforme	
	Data	
	Applications	
Jo Kumar CFO	Taux d'inscription utilisateurs	1. Assurer le leadership et la gestion du département financier 2. Soutenir le service financier dans l'élaboration d'objectifs financiers à court et à long terme 3. Créer des plans financier 4. Superviser le processus des devis annuel. 5. Gérer la liquidité de l'entreprise. 6. Créer un système de contrôle des coûts
	Business	
	Innovation dans le périmètre d'une architecture	
	Innovation technique rapide et expérimentation	
	Data	

Déclaration de Travail d'Architecture

V.D: 0.1

Client : Foosus



Date rev: 15/09/2021

Pete Parker Engineering Owner	2	Applicative	1. Examens d'évaluation, de faisabilité et de planification du projet 2. Revues de conception technique et de planification 3. Analyse et optimisation des horaires 4. L'analyse coûts-avantages
		Data	
Jack Harner Operations Lead	3	Visibilité de la plateforme	1. Faciliter les interactions entre l'équipe interne et les plateformes hébergées pour le succès global de l'organisation 2. Encadrer les membres de l'équipe 3. Identifier les manquements 4. Organiser une formation pour doter le personnel des connaissances sur les tendances en développement afin de créer un service hautement qualifié
		Applicative	
		Business	
Nicolas Cervantes Solution Architect	4	Business	1. Créer une vision technique pour montrer comment le logiciel couvrira les besoins de l'entreprise. 2. Transmettre la vision technologique à l'équipe de développement 3. Assurer l'interface entre l'équipe business et développement. 4. Décider d'une pile (stack) technologique.
		Technologique	
		Applicative	
		Data	
Product Team	2	Business	1. Améliorer l'expérience du produit. 2. Stratégie du produit

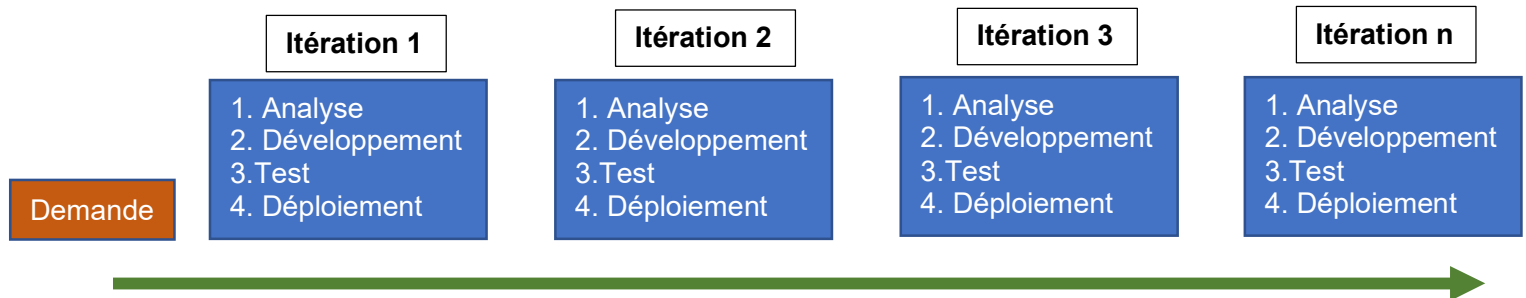
Approche managériale

Foosus utilise deux approche managériales :

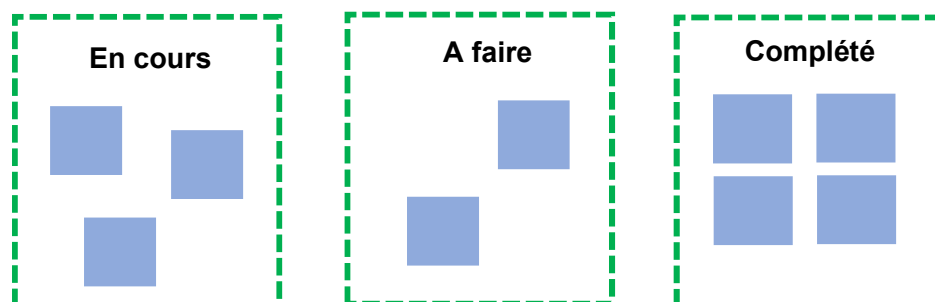
Nicolas Cervantes | Architecte Logiciel



La méthode Agile



La méthode Kanban



Procédures de changement de périmètre

1. Etudier les changement à réaliser
2. Définir la nouvelle architecture
3. Connaître les éléments existant afin de réaliser la migration des données.
4. Assurer la stabilité de la nouvelle implémentation en réalisant des test avant production.
5. Echanger avec les parties prenantes afin de connaitre et signaler de défaillance.
6. Corriger les défaillance.
7. Alimenter la documentation avec les résultats de la mise en production et futurs changements.



4. Rôles et responsabilités

Structure de gouvernance

	Nom / Equipe	Rôle / Responsabilités	Niveau hiérarchique
DG	Ash Callum	Directeur General	DRG_T1
SI	Natasha Jarson	Directeur des Systèmes d'Information	DR_T2
SI	Pete Parker	Engineering Owner	MG
SI	Frontend Squad	Développement d'interfaces web	CL_01
SI	Ruby Dev Squad	Développement d'interfaces Ruby	CL_01
SI	Backend Applications Squad	Développement d'interfaces Backend	CL_01
SI	Identity Management Squad	Gestion des accès	CL_01
SI	Nicolas Cervantes	Enterprise Architecture Owner	CL_01
SI	Jack Harkner	Management des opérations	MG
SI	Equipe Opérations	Operations	CL_01
PR	Daniel Anthony	Directeur Produits	DR_T2
PR	4x Managers Produit	Management produit	CL_01
PR	Equipe Expérience Client	Expérience Client	CL_01
PR	Data Analytics Team	Analyse de données	CL_01
PR	Equipe d'Accomplissement Client	Relation Client	CL_01
MK	Christina Orgega	Directeur Marketing	DR_T2
FN	Jo Kumar	CFO	DR_T2
FN	Equipe Finance	Finances	CL_01

DRG_T1 = Direction générale, **DR_T2** = Direction, **MG** = Manager, **CL_01** = Collaborateur



Process du projet

Communication

Trello

Trello repose sur les logiciels de **gestion de projet/missions**.

Permet de personnaliser et ajouter d'autres fonctionnalités à mesure que le travail de l'équipe s'élargit, également à organiser des tâches et développer l'esprit d'équipe. Permet aussi aux équipes de configurer et de personnaliser rapidement des flux de production

Sauvegarde et gestion de la documentation

Git

Logiciel de suivi des modifications dans n'importe quel ensemble de fichiers, généralement utilisé pour coordonner le travail entre les programmeurs développant en collaboration le code source pendant le développement du logiciel. Ses objectifs incluent la vitesse, l'intégrité des données et la prise en charge des flux de travail distribués et non linéaires.

Cet outil permettra aux développeur et parties prenantes du projet à avoir un suivi des changements de la base du code et également à l'architecte et gestionnaires de la documentation à appliquer les mis à jour réalisés.

GitHub

À un niveau élevé GitHub repose sur Git, GitHub est un site Web et un service basé sur le cloud qui aide les développeurs à stocker et à gérer leur code, ainsi qu'à suivre et contrôler les modifications apportées à leur code.

Le contrôle de version aide les développeurs à suivre et à gérer les modifications apportées au code d'un projet logiciel. Au fur et à mesure qu'un projet logiciel grandit, le contrôle de version devient essentiel.

GitHub se compose principalement de trois piliers :

- **Référentiel GitHub :**
 - Un référentiel GitHub peut être utilisé pour stocker un projet de développement.
 - Il peut contenir des dossiers et tout type de fichiers (HTML, CSS, JavaScript, Documents, Données, Images).
 - Un référentiel GitHub doit également inclure un fichier de licence et un fichier README sur le projet.
 - Un référentiel GitHub peut également être utilisé pour stocker des idées ou toutes les ressources que vous souhaitez partager.
- **Branche GitHub :**
 - Une branche GitHub est utilisée pour travailler avec différentes versions d'un référentiel en même temps.
 - Par défaut, un référentiel a une branche master (une branche de production).

Déclaration de Travail d'Architecture

V.D: 0.1

Client : *Foosus*



Date rev: 15/09/2021

- Toute autre branche est une copie de la branche principale (telle qu'elle était à un moment donné).
- Les nouvelles branches sont destinées aux corrections de bogues et aux fonctionnalités distinctes de la branche principale. Lorsque les modifications sont prêtes, elles peuvent être fusionnées dans la branche master. Si vous apportez des modifications à la branche principale tout en travaillant sur une nouvelle branche, ces mises à jour peuvent être récupérées.
- **Commits GitHub :**
 - Dans GitHub, les modifications sont appelées commits.
 - Chaque commit (changement) a une description expliquant pourquoi un changement a été effectué.
- **Pull Requests :**
 - Les Pull Requests sont au cœur de la collaboration GitHub.
 - Avec une demande d'extraction, vous proposez que vos modifications soient fusionnées (tirées) avec le maître.
 - Les demandes d'extraction affichent les différences de contenu, les modifications, les ajouts et les soustractions en couleurs (vert et rouge).
 - Dès que vous avez un commit, vous pouvez ouvrir une pull request et démarrer une discussion, avant même que le code ne soit terminé.



Rôles et responsabilités (RACI)

Le tableau qui suit indique le niveau de pouvoir, intérêt et intégration des parties prenantes

Cas: Changement d'Architecture et système	
<p>Responsable</p>	<p>Approbateur</p>
<p>Consulté</p>	<p>Informé</p>
Ash Calum – CEO	Natalia Jarson - CIO
Daniel Anthony - CPO	Christina Orgega - CMO
Jo Kumar CFO	Nicolas Cervantes - AL
Pete Parker - Resp. Ingénierie	Jack Harner – Dir. Opérations
Equipe Commerciale	Equipe Développement
Producteurs	Clients



Parties prenantes, préoccupations et visions		
Partie prenante	Préoccupation	Vision
Ash Callum CEO	3	1. Innovation 2. Croissance 3. Communication 4. Améliorer les résultats et maximiser la rentabilité à long terme
	Business	
	Data	
Natasha Jarson CIO	Innovation technique rapide et expérimentation	1. Innovation 2. Enthousiasmer les équipes 3. Différencier la position sur le marché 4. Déploiement et gestion de systèmes maintenables, scalable, performant et sûr.
	Business	
	Applicative	
	Technologique	
	Data	
Daniel Anthony CPO	Innovation dans le périmètre d'une architecture	1. Innovation 2. Améliorer l'expérience du produit. 3. Stratégie du produit 4. Management de l'équipe produits 5. Rapports basés sur les données
	Innovation technique rapide et expérimentation	
	Business	
	Visibilité de la plateforme	
Christina Ortega CMO	4	1. Fixer des buts et objectifs marketing 2. Planifier, mettre en œuvre et gérer la stratégie marketing 3. Déterminer les KPI pour le service marketing 4. Créer et présenter le plan et la stratégie marketing annuels 5. Utiliser des données et des rapports pour prendre des décisions fondées sur des preuves
	Data	
	Applications	
	Taux d'inscription utilisateurs	
Jo Kumar CFO	4	1. Assurer le leadership et la gestion du département financier 2. Soutenir le service financier dans l'élaboration d'objectifs financiers à court et à long terme 3. Créer des plans financier 4. Superviser le processus les devis annuel. 5. Gérer la liquidité de l'entreprise. 6. Créer un système de contrôle des coûts
	Business	
	Innovation dans le périmètre d'une architecture	
	Innovation technique rapide et expérimentation	
Pete Parker Engineering Owner	2	1. Examens d'évaluation, de faisabilité et de planification du projet 2. Revues de conception technique et de planification 3. Analyse et optimisation des horaires 4. L'analyse coûts-avantages
	Applicative	
	Data	



Date rev: 15/09/2021

<p>Jack Harner Operations Lead</p>	3	Visibilité de la plateforme	<p>1. Faciliter les interactions entre l'équipe interne et les plateformes hébergées pour le succès global de l'organisation 2. Encadrer les membres de l'équipe 3. Identifier les manquements 4. Organiser une formation pour doter le personnel des connaissances sur les tendances en développement afin de créer un service hautement qualifié</p>
		Applicative	
		Business	
<p>Nicolas Cervantes Solution Architect</p>	4	Business	<p>1. Créer une vision technique pour montrer comment le logiciel couvrira les besoins de l'entreprise. 2. Transmettre la vision technologique à l'équipe de développement 3. Assurer l'interface entre l'équipe business et développement. 4. Décider d'une pile (stack) technologique.</p>
		Technologique	
		Applicative	
		Data	
<p>Product Team</p>	2	Business	<p>1. Améliorer l'expérience du produit. 2. Stratégie du produit</p>



5. Approche architecturale

Process d'architecture

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

Phase	Entrée/Sortie	Notes
Préliminaire	Contexte et recueil des besoins / Résultats d'analyse	
A —Vision de l'architecture	Besoins technique + nouvelle architecture	Répertorier les besoins et éléments nécessaires pour la nouvelle architecture
B —Architecture business	Demandes du métier / proposition de fonctions	Apporter une vision sur les besoins du métier
C — Architecture des systèmes d'information	Eléments techniques + support	Phase de définition
D — Architecture technologique	Analyse du stack technologique	Approuver les besoins techniques pour l'implémentation



E —Opportunités et solutions	Plateforme scalable et modulable	Les nouvelles fonctionnalités permettront à la plateforme de recevoir plus d'utilisateurs
F —Planning de migration	Utilisation des services de la plateforme d'hébergement	La migration se fera en plusieurs étapes afin d'éviter couper les services.
G —Gouvernance de l'implémentation	L'équipe technique et de direction	Vation de la nouvelle plateforme
H —Management du changement d'architecture	Les besoins du métier / Les offres techniques	Les changements se feront sous un protocole stricte et ouvert aux améliorations.
Management des conditions requises	Les besoins commerciales / Les offres techniques	Les conditions sont définis pas les équipes du métier, puis communiqués aux équipes techniques.

6. Contenu de l'architecture

Modélisez les services autour du domaine de l'entreprise.
 Décentralisez tout. Des équipes individuelles sont chargées de concevoir et de créer les services. Évitez le partage de code ou de schémas de données.



Le stockage de données doit être accessible uniquement au service auquel les données appartiennent. Utilisez le meilleur stockage pour chaque type de service et de données. Les services communiquent via des API bien conçues. Évitez les fuites des détails de la mise en oeuvre. Les API doivent modéliser le domaine et non la mise en oeuvre interne du service.

18

Évitez le couplage entre les services. Le couplage peut notamment être dû à des schémas de base de données partagés et à des protocoles de communication rigides.

Confiez les responsabilités transversales, telles que l'authentification et la terminaison SSL, à la passerelle.

Gardez les connaissances du domaine hors de la passerelle. La passerelle doit gérer et acheminer les requêtes de clients sans aucune connaissance des règles métier ou de la logique de domaine. Dans le cas contraire, la passerelle devient une dépendance et peut causer un couplage entre les services.

Les services doivent présenter un couplage faible et une forte cohésion fonctionnelle. Les fonctions qui sont susceptibles de changer en même temps doivent être empaquetées et déployées ensemble. S'ils résident dans des services distincts, ces services se retrouvent fortement couplés, car un changement de l'un d'eux exigera la mise à jour de l'autre. Une communication trop importante entre deux services peut être un signe de couplage fort et de faible cohésion.

Isolez les défaillances. Utilisez des stratégies de résilience afin d'empêcher que les défaillances au sein d'un service n'entraînent une réaction en chaîne. Pour plus d'informations, consultez Conception d'applications résilientes pour Azure.

7. Procédure d'acceptation

L'acceptation de la nouvelle plateforme :

1. Lister les besoins métier
2. Offres et propositions technique
3. Validation de l'offre technique par les directions
4. Validation des tests
5. Validation de la migration
6. Validation de l'implémentation
7. Analyse des besoins
8. GoLive



8. Approbations signées

Valideurs : Natasha Jarson
La Direction Générale
Date de signature : 13/10/2021