



# Déclaration de Travail d'Architecture

*Projet : Foosus géoconscient v2 Client : Foosus*



## Table des matières

Objet de ce document.....	5
Déclaration de travail d'architecture.....	5
Requête du projet et contexte.....	5
Description du projet et périmètre.....	6
Vue d'ensemble.....	6
Alignement stratégique.....	6
Objectifs et périmètre.....	7
Objectifs.....	7
Périmètre.....	7
Parties prenantes, préoccupations, et visions.....	8
Approche managériale.....	10
Procédures de changement de périmètre.....	10
Rôles et responsabilités.....	12
Structure de gouvernance.....	12
Process du projet.....	12
Rôles et responsabilités (RACI).....	15
Approche architecturale.....	16
Process d'architecture.....	16
Contenu de l'architecture.....	19
Plan de travail.....	21
Architecture de développement.....	21
Mise en place de l'environnement de production.....	22



## Foosus – Projet *géoconscient* V2

Itération de développement.....	22
Plan de communication.....	23
Évènements.....	23
Canaux.....	24
Formats et contenu.....	25
Rythme de communication.....	26
Collaboration.....	27
Risques et facteurs de réduction.....	29
Analyse des risques.....	29
Hypothèses.....	29
Critères d'acceptation et procédures.....	31
Métriques et KPIs.....	31
Procédure d'acceptation.....	32
Approbations signées.....	33



Foosus – Projet *géoconscient V2*

## Information sur le document

<i>Nom du projet</i>	Projet <i>Foosus</i> <i>géoconscient v2</i>
<i>Préparé par :</i>	Stiven GUILLAUME
<i>N° de version du document :</i>	0.1
<i>Titre :</i>	<i>Déclaration de travail d'architecture</i>
Types d'action :	Approbation, Révision, Information, Classement, Action requise, Participation à une réunion, Autre (à spécifier)
Historique de versions du document	Voir git



## Objet de ce document

---

Ce document est une Déclaration de travail d'architecture pour le projet *Foosus géoconscient* v2.

La Déclaration de travail d'architecture définit le périmètre et l'approche qui seront utilisés pour mener à bien un projet d'architecture. La Déclaration de travail d'architecture constitue habituellement le document qui permet de mesurer la réussite de l'exécution du projet d'architecture et peut former la base de l'accord contractuel entre le fournisseur et le consommateur de services d'architecture. En général, toutes les informations de ce document doivent se situer à un haut niveau.

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

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

## Déclaration de travail d'architecture

### Requête du projet et contexte

---

Foosus, start-up créée il y a 3 ans, est spécialisée dans le secteur de l'alimentation durable. Son objectif principal est de soutenir l'alimentation locale en mettant les consommateurs en contact avec des producteurs et des artisans locaux. Pour cela, Foosus s'appuie sur une plateforme e-commerce, développée en interne.

Après une forte progression, le nombre de nouveaux clients est stable, la plateforme actuelle de Foosus n'arrivant plus à soutenir les projets de croissance et d'expansion de l'entreprise. De ce fait, Foosus fait le choix de se lancer dans un projet de nouvelle architecture pour évoluer.



## Description du projet et périmètre

---

Le nouveau projet a pour objectif la création et le déploiement d'une nouvelle plateforme de commerce électronique, dont le but est de permettre à l'entreprise de renouer avec la croissance, en améliorant l'image de marque, et de pouvoir accueillir des nouveaux clients, mais aussi des nouvelles régions et services.

La plateforme devra répondre aux exigences techniques suivantes :

- Géolocalisation des clients pour leur proposer des produits disponibles près de leur lieu de résidences
- Une architecture évolutive pour pouvoir déployer les services dans diverses régions
- Réduire les interruptions de services en cas de modifications au système
- Solution utilisable sur périphériques mobiles
- Gestion de plusieurs types d'utilisateurs avec des fonctionnalités spécifiques en fonction des rôles
- Livrables pouvant être fournis à intervalles réguliers pour que le nouveau système soit rapidement opérationnel



# Objectifs et périmètre

## Objectifs

---

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

<b>Objectif Business</b>	<b>Notes</b>
Retrouver un taux d'inscription positif	Le système doit pouvoir supporter un nombre élevé de nouvelles inscriptions
Améliorer la réputation de Foosus sur le marché	L'instabilité de la plateforme et les interruptions de services ont provoqués une image de marque négative
Sortir de manière itérative et rapide un nouveau produit	Coexistence avec la plateforme existante, en mode maintenance.

## Périmètre

---

Dans le but d'éviter le risque de dérive des objectifs, le périmètre doit être défini au mieux.

L'utilisation de méthode AGILE pour la gestion et la mise en place du projet implique qu'une de facette du périmètre fonctionnel, au sens premier du terme, soit abandonné.

Les questions de délais de développement et de budgets ne sont plus impératifs, comme ils pourraient être dans le cas de gestion de projet avec la méthode de cycle en V ou de wartefall (cascade).

Néanmoins, des contraintes son présentes, au moins au niveau de la phase d'étude :



## Foosus – Projet *géoconscient* V2

- Projet initial approuvé pour un coût de 50 000\$
- Période de 6 mois pour la définition de l'architecture, et la préparation d'un projet de suivi afin de développer un prototype
- Une architecture qui doit permettre d'obtenir le meilleur rapport qualité-coût
- Une architecture pouvant inclure de nouveaux composants personnalisés ou des composants déjà existant pour favoriser la flexibilité, la stabilité et l'extensibilité

D'un point de vue plus technique, les directives suivantes doivent être si possible appliquées, ou a défaut s'en rapprocher le plus possible :

- Solution open source préférables aux solutions non open sources
- Support continu des composants à prendre en compte à la sélection
- rester dans la même pile technologique pour tous les modules composant l'application

## 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 (cible)
Ash Callum : CEO (Chief Executive Officer)	Business Visibilité de la plateforme	L'expansion au sein des marchés locaux et le fait de fournir du géo-ciblage sont vus comme des facteurs critiques pour toucher une gamme plus large d'utilisateurs.
Daniel Anthony : <b>CPO</b> (Chief Product Officer)	Architectural	Un périmètre clair est un besoin pour l'entreprise, pour





## Foosus – Projet *géoconscient V2*

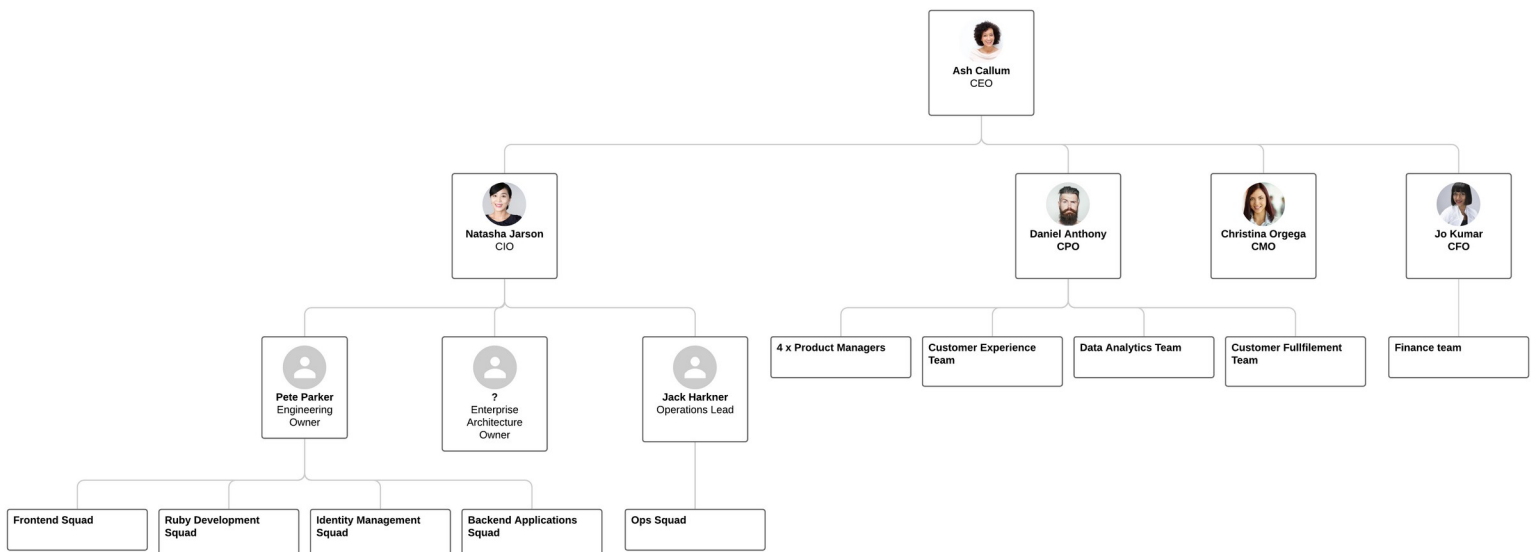
<p>Officer)</p> <p>Natasha Jarson : <b>CIO</b> (Chief Information Officer)</p> <p>Peter Parker : <b>CTO</b> (Chief Technology Officer)</p>	<p>Innovation technique rapide</p> <p>Visibilité de la plateforme</p>	<p>assurer que chaque incrément soit considéré selon son impact sur le fait de fournir les capacités business nécessaires et de soutenir la croissance à venir de Foosus.</p>
<p>Jack Harkness : <b>COO</b> (Chief Communications Officer)</p>	<p>Visibilité de la plateforme</p>	<p>Besoin de mettre en place un design d'architecture qui nous offre en temps réel des connaissances et une vision de la santé de la plateforme techniquement et d'un point de vue commercial</p>
<p>Jo Kumar : <b>CFO</b> (Chief Financial Officer)</p>	<p>Architectural</p> <p>Business</p> <p>Innovation technique rapide</p>	<p>Des process pour réduire le risque de sortir des solutions qui échouent ou qui soient de mauvaise qualité</p> <p>La capacité de sortir de nouvelles versions de notre plateforme sans impacter l'utilisateur par des interruptions de service</p>
<p>Christina Orgega : <b>CMO</b> (Chief Marketing Officer)</p>	<p>Innovation technique rapide</p> <p>Visibilité de la plateforme</p>	<p>L'apprentissage doit être au cœur de notre état cible de l'architecture</p> <p>La plateforme doit être conçue en gardant à l'idée l'extensibilité et la personnalisation des</p>



## Foosus – Projet *géoconscient* V2

		fonctionnalités
--	--	-----------------

## Approche managériale



## Procédures de changement de périmètre

Dans le cas où un changement de périmètre doit être effectué, pour tout ou partie du périmètre actuel une procédure doit être mis en place.

Nous préconisons de suivre le diagramme suivant.

Pour que la demande de changement de périmètre soit acceptée, il faut qu'elle réponde aux critères suivants :

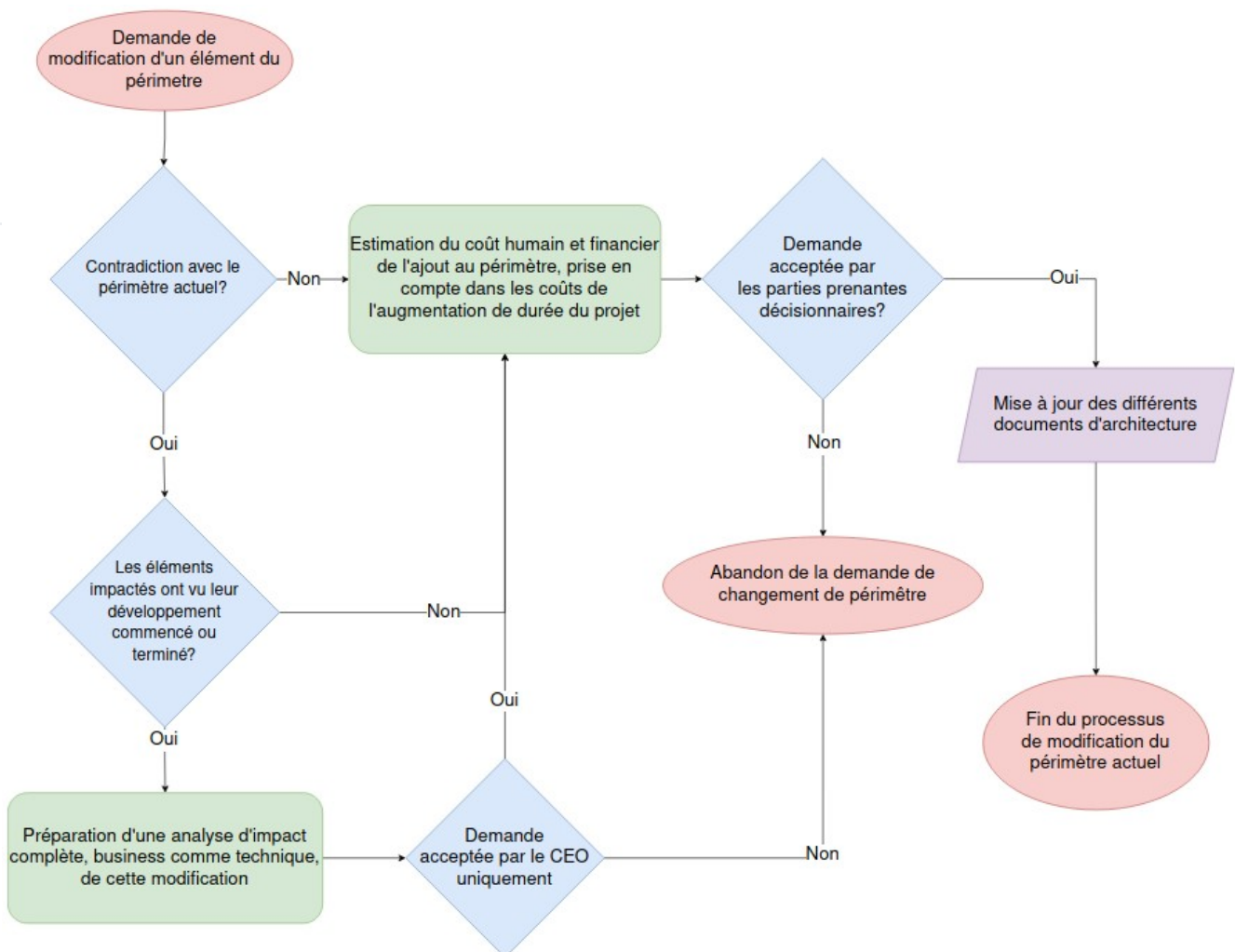
- Que la demande soit chiffrée, évaluée et analysée. Une estimation complète des coûts doit être faite.
- Que la présentation de la demande aux parties prenantes décisionnaires soit arbitrée par



## Foosus – Projet *géoconscient* V2

Ash Callum, le CEO, dans le cas où elle entre en contradiction avec le périmètre existant

- Qu'elle soit validée par les parties prenantes décisionnaires
- Qu'une analyse d'impact complète soit présentée au CEO dans le cas où il y aurait des impacts sur des éléments du périmètre actuel (développement déjà effectué,...)





# Rôles et responsabilités

## Structure de gouvernance

---

Dans le cadre du projet, il est nécessaire de préciser le rôle de chacun et la responsabilité des différents périmètres.

En tant que CEO, **Ash Callum** aura la responsabilité complète du projet, mais de manière indirecte. Il pourra faire jouer éventuellement son droit de veto, et participera à l'arbitrage en cas de conflit avec les différentes parties prenantes avec responsabilité.

Il aura aussi la responsabilité d'accepter les changements de périmètre dans le cas où des interaction sont présente.

En tant que CIO, **Natasha Jarson** devra s'appuyer sur Peter Parker, le CTO pour l'aspect purement technique. Ce dernier aura la responsabilité complète du développement, d'un point de vue technique, c'est à dire de l'application et du développement des différentes users stories.

**Daniel Anthony**, le CPO, aura la responsabilité de l'aspect « business ». de ce fait, Il sera responsable de la validation et de la non validation des livraisons, de la définition des plans de tests métier, et de la création des users stories, en coparticipation avec Peter Parker.

## Process du projet

---

Le projet du nouveau système de commerce électronique est un projet sur le long terme.

Pour cela, il doit avoir de bonnes bases techniques, mais aussi avoir un processus de développement fiable.

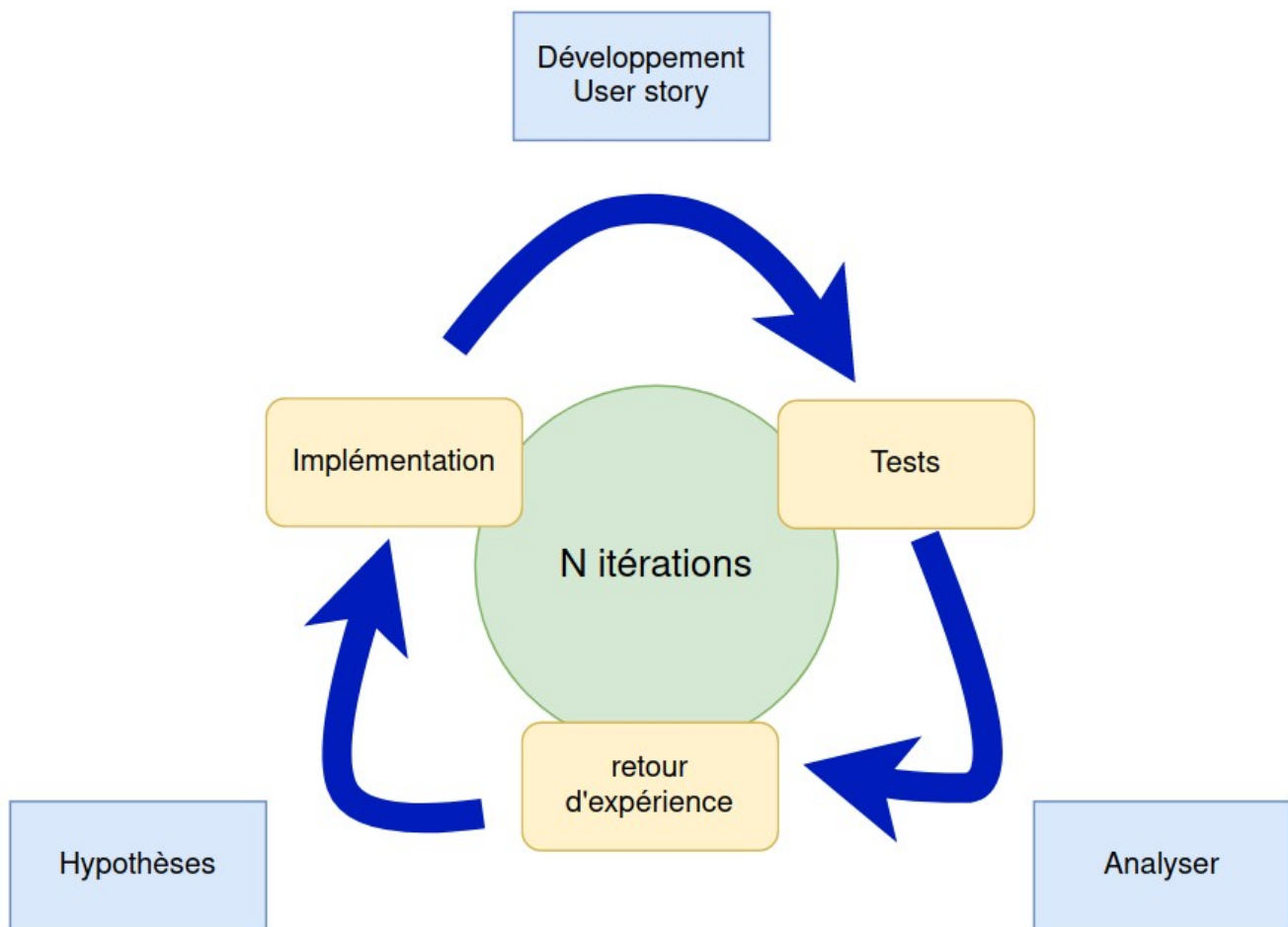
D'un point de vue « business », une des contrainte est que le nouveau système doit être conservé, d'abord en système principal, avec activation « au fil de l'eau » des nouvelles fonctionnalités du nouveau système, puis à terme en mode maintenance.

L'équipe interne de développeurs, au nombre de 15, à déjà une connaissance des



architectures type « micros services » et des méthodologies Agiles et Lean. De ce fait, les process et le cycle de développement auront pour base ces principes.

Enfin, l'utilisation d'un repository Git, comprenant les différentes branches de code, mais aussi la documentation nécessaire devra être utilisée.



Chaque itération sera décomposé de la forme suivante :

- Une phase d'implémentation : Cette phase devra répondre à une seule

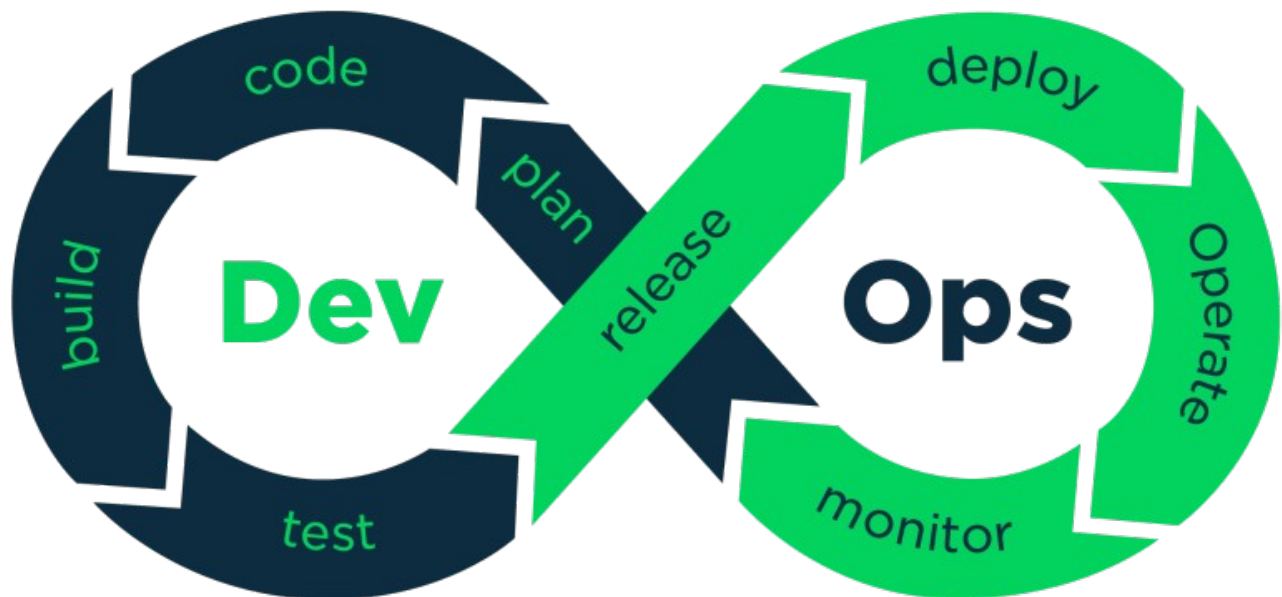


problématique : Développer la user story de cette itération.

- Une phase de tests : La fonctionnalité est développée, il faut vérifier sa conformité, est ce qu'elle répond au besoin business, tests fonctionnels et unitaires,...
- Une phase de retour d'expérience : Analyse de la non conformité éventuelle, hypothèse et mise en place de la prochaine user story, préparation de la prochaine itération

Pour chaque itération, la communication vers les différentes parties prenantes, et en fonction de leur niveau d'exigence, devra être effectuée.

De plus chaque itération devra, si la phase de tests est concluante, se conclure par la livraison de la fonctionnalité développée.





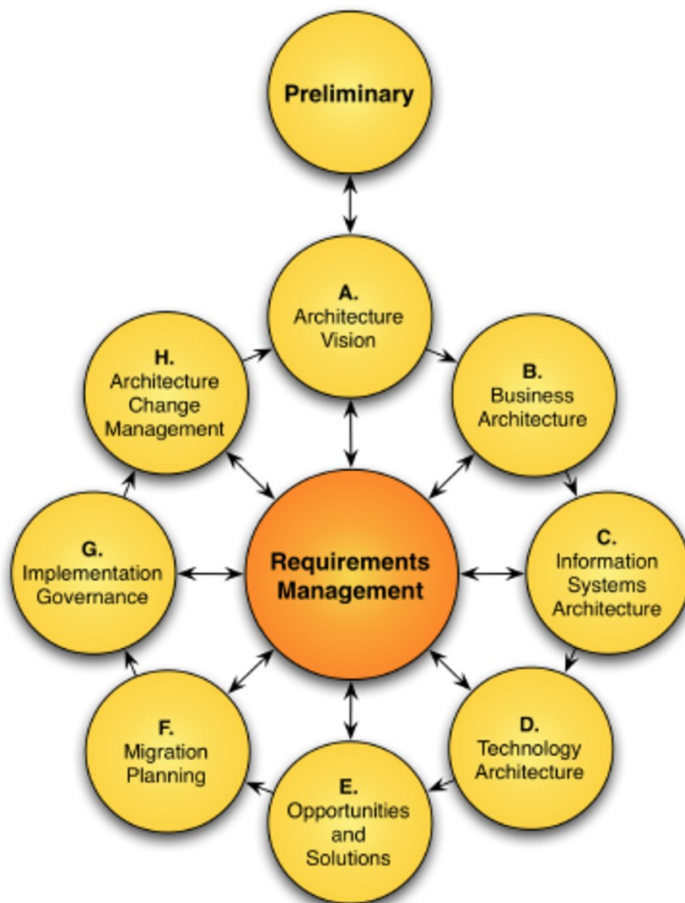
## Rôles et responsabilités (RACI)

	Ash Callum (CEO)	Jo Kumar (CFO)	Natasha Jarson (CIO)	Daniel Anthony (CPO)	Jack Harkness (COO)	Peter Parker (CTO)	Christina Ortega (CMO)	Equipe Dev	Architecte logiciel
Phase étude									
Définition des objectifs business	R		A				C		I
Rédaction des documents de haut niveau	I		R	R	A/C				I
Rédaction de la déclaration de travail d'architecture	A		I						R
Rédaction des contrats entre les parties prenantes	A/C	I	I	I	I	I	I		R
Choix des KPI/indicateurs de réussite	I		R	R	A/C		C		I
Phase de developpement									
Implémentation de l'architecture								R	C
« Animation » des itérations	I		I			A		R	C
Élaboration des tests fonctionnels	I		I	C			C	R	C
Élaboration des tests unitaires	I		I			A		R	C
Rédaction des users stories	I		I	C		A	C	A	C
Validation de la conformité des livraisons	I		A	R		A	I	C	C
	<p> <b>A - Accountable - il supervise et</b>  <b>R - Responsable - rend des comptes</b>  <b>C - Consulted - I - Informed</b>  <b>il conseille - il est informé</b> </p>								

# 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	+ TOGAF, ou tout autre framework d'architecture si besoin + Stratégie business/de gouvernance +Framework d'architecture actuel -Model d'organisation pour l'architecture d'entreprises -Cadre de gouvernance d'architecture -Un dépôt d'architecture	Assurer un engagement des différents acteurs impliqués ou bénéficiaires, définition des principes d'architecture, mettre en place un processus
A —Vision de l'architecture	<b>+Objectifs et principes métier</b> +Cadre d'architecture +Dépôt d'architecture avec documentation d'architecture existant de la phase préliminaire -Matrice des parties prenantes -Cadre d'architecture -Vision d'architecture	Définition du but de l'architecture, des méthodologies, et descriptions des parties prenantes
B —Architecture business	<b>+Architecture métier</b> +Plan de communication +Dépôt d'architecture avec documentation d'architecture existant de la phase préliminaire +Version préliminaire du document de définition d'architecture -Diagrammes du cycle de vie du produit -Résultats d'analyse du gap	Création d'une base de référence et d'architectures cibles Analyse du gap
C — Architecture des	+principes relatifs aux données et aux applications	Création d'une architecture de données et des



## Foosus – Projet *géoconscient* V2

<p>systèmes d'information</p>	<p>+Dépôt d'architecture avec documentation d'architecture existant de la phase préliminaire</p> <p>-Information sur les composants systèmes</p>	<p>applications</p>
<p>D — Architecture technologique</p>	<p>+Dépôt d'architecture avec documentation d'architecture existant de la phase préliminaire</p> <p>- Draft des spécification requises pour l'architecture</p> <p>- Draft de la déclaration de travail d'architectures</p> <p>- Composition des environnements</p>	<p>Élaboration d'une architecture technologique dans le but de créer la base de l'intégration dans l'entreprise</p>
<p>E — Opportunités et solutions</p>	<p>+Dépôt d'architecture avec documentation d'architecture existant de la phase préliminaire</p> <p>- Draft des spécification requises pour l'architecture</p> <p>- Draft de la déclaration de travail d'architectures</p> <p>- Roadmap d'architecture</p>	<p>Identification des paramètres du changement, des phases majeures et des projets les plus importants qui seront à réaliser dans l'optique de la migration de l'environnement actuel vers l'environnement cible</p>
<p>F —Planning de migration</p>	<p>+Dépôt d'architecture avec documentation d'architecture existant de la phase préliminaire</p> <p>+ Draft de la déclaration de travail d'architecture</p> <p>- Plan de migration</p>	<p>Classement par ordre de priorité des projets au sein de l'architecture technologique (phase D) et estimation des coûts</p>



	<ul style="list-style-type: none"> <li>- Déclaration de travail d'architecture terminée</li> <li>- Spécification requises pour l'architecture terminée</li> </ul>	
G — Gouvernance de l'implémentation	<ul style="list-style-type: none"> <li>+Dépôt d'architecture avec documentation d'architecture existant de la phase préliminaire</li> <li>- Contrats d'architecture signés</li> </ul>	Organiser chaque projets d'intégration
H — Management du changement d'architecture	<ul style="list-style-type: none"> <li>+ Requête de changements d'architecture - business</li> <li>+ Requête de changements d'architecture – Technologique</li> <li>- Mise à jours des architectures</li> </ul>	Gestion du changement pour arriver a l'élaboration d'une architecture
Management des conditions requises	<ul style="list-style-type: none"> <li>- Un énoncé des exigences structuré, comprenant : <ul style="list-style-type: none"> <li>• Exigences modifiées</li> <li>• Déclaration d'impact sur les exigences</li> </ul> </li> </ul>	Définir un processus défini par des exigences d'architecture d'entreprise, qui sont identifiées, stockées et mise en place dans et hors des phases ADM

## Contenu de l'architecture

Le cadre de contenu d'architecture TOGAF (ou ACF pour «Architecture Content Framework») fournit une catégorisation des meilleures pratiques pour le contenu de l'architecture. Néanmoins, tous les éléments ne sont pas également pertinents pour chaque projet. Le tableau ci-dessous décrit les zones de contenu pertinentes pour ce projet spécifique.



<b>Zone de contenu</b>	<b>Entrée/Sortie</b>
Principes, Vision, et Conditions requises de l'Architecture	Brief business de haut niveau
Architecture Business	Contrat d'architecture des utilisateurs business
Architecture des systèmes d'information — Données	Contrat de Conception et de Développement de l'Architecture
Architecture des systèmes d'information — Applications	Spécification des Conditions requises pour l'Architecture
Architecture technologique	Contrat de Conception et de Développement de l'Architecture
Réalisation de l'architecture	Déclaration de travail d'architecture



## Plan de travail

---

Dans le cadre de la mise en place du projet, différents items de travail doivent être mis en place, avec leur activité principale et un livrable, qui servira de base de travail pour les différentes équipes.

L'objectif final est obtenir un état cible du projet, autant d'un point de vue développement que d'un point de vue architecture.

## Architecture de développement

---

### Mise en place d'un environnement de développement

#### Item de travail : Livrables

##### *Infrastructure de développement fonctionnelle*

Une architecture complète basé sur le cloud devra être mise en place. Les différentes instances de serveur, load balancing, les groupes de sécurités ... doivent être fonctionnels et configurés.

##### *Outils de travail collaboratif et intégration continue*

Mise en place d'une forge avec github. Configuration des accès pour tous les développeurs. Activation et configuration des pipelines et des modules d'intégration continue pour un déploiement sur les environnements de développement après chaque opération de merge request

Mise en place des tests automatiques de livraisons

Mise en place d'un outil de gestion de projet en ligne acceptant la méthode KanBan, ex : Trello.



## Foosus – Projet *géoconscient* V2

Mise en place d'un outil de communication interne, avec automatisation de traitement. Ex : Discord, avec bot custom pour l'intégration des événements github dessus

## Mise en place de l'environnement de production

---

### Item de travail : Mise en place d'un environnement de production

#### Item de travail : Livrables

##### *Infrastructure de production fonctionnelle*

Une architecture complète basé sur le cloud devra être mise en place. Les différentes instances de serveur, load balancing, les groupes de sécurités ... doivent être fonctionnels et configurés.

Le multi AZ doit être configuré, la scalabilité fonctionnelle.

##### *Lien avec les outils de travail collaboratif, intégration continue...*

Connexion entre le github et l'environnement de production configurée pour le déploiement continue des nouvelles versions ou fonctionnalités.

## Itération de développement

---

### Item de travail : Hypothèse

#### Item de travail : Livrables

##### *User story*

Une user story devra être livré pour l'étape suivante de l'itération. De cette user story résultera un développement d'une fonctionnalité de la nouvelle plateforme



## **Item de travail : Développement**

### **Item de travail : Livrables**

#### ***Fonctionnalité développée***

La fonctionnalité a été développée en respectant la user story en entrée.

## **Item de travail : Analyse**

### **Item de travail : Livrables**

#### ***Résultats des différents tests techniques***

Le plan de test unitaire et technique doit être livré pour confirmer ou infirmer le bon développement de la fonctionnalité en vue d'une mise en production

#### ***Résultats des différents tests fonctionnels***

Les résultats issus d'un plan de test fonctionnel doivent être livrés pour confirmer ou infirmer le bon développement de la fonctionnalité en vue d'une mise en production

## **Plan de communication**

### **Événements**

---

Dans le cadre d'une communication optimale, divers événements doivent être mis en place tout au long du projet.

Dans le cas d'une démission d'un développeur :

- Le binôme restant se verra dispatché dans d'autres équipes de développeurs. La user story sera reportée à la prochaine itération

Dans le cas d'une embauche :

- Mettre en place une formation aux outils et aux processus de Foosus en immersion dans



## Foosus – Projet *géoconscient* V2

une des 5 équipes de développement. Cette équipe verra son itération doublée pour éviter une pression inutile et permettre une meilleure adaptation du nouveau développeur

Quotidiennement :

- Daily Scrum Meeting avec les équipes chargées du développement, pour partager ce qui a pu être fait la veille comme les éventuels points de blocage

Avant chaque itération :

- Réunion entre les parties prenantes concernées pour la rédaction des user-stories et des plans de tests fonctionnel associés
- Réunion de l'équipe de développement pour la définition des différents objectifs interne et l'utilisation de la méthode Kanban pour définir les tâches de chacun. Affectation des users story aux 5 équipes de développeurs

Après chaque itération :

- Réunion avec les équipes projets et les parties prenantes pour analyser les plans de tests techniques comme fonctionnels.
- Retour d'expérience de l'itération dans le cadre d'une amélioration continue

A la fin du projet :

- Retour d'expérience sur le projet avec les différentes parties prenantes
- Analyse du gap entre les KPI mesurée et les objectifs souhaités
- Une équipe R&D sera mise en place pour les évolutions/projets futurs

## Canaux

---

La communication vers les différentes parties prenantes sera effectuée via différents canaux :

- **Stand up meeting.** Toujours dans un but de développement plus rapide et de meilleure qualité, les Stand up meeting, ou réunions debout, seront à mettre en place, principalement du côté de l'équipe de développeur. Cette réunion a pour objectif de





## Foosus – Projet *géoconscient* V2

faire un point rapidement sur l'itération en cours, de dynamiser le groupe de développeur, de répondre à une problématique ou encore d'échanger des informations utiles.

- **E-mail** : Une communication électronique régulière sera effectuée. Un e-mail avec un récapitulatif, sous différents formats, sera transmis à chaque fin d'itération.
- Outils de travail collaboratif (échange libre, trello ? Discord?)

## Formats et contenu

---

**Parties prenante business** : Utilisation d'une petite infographie (user story, équipe de développeur, nombres clés (nombre de tests passés,...), picto validation fonctionnel ok/nok)

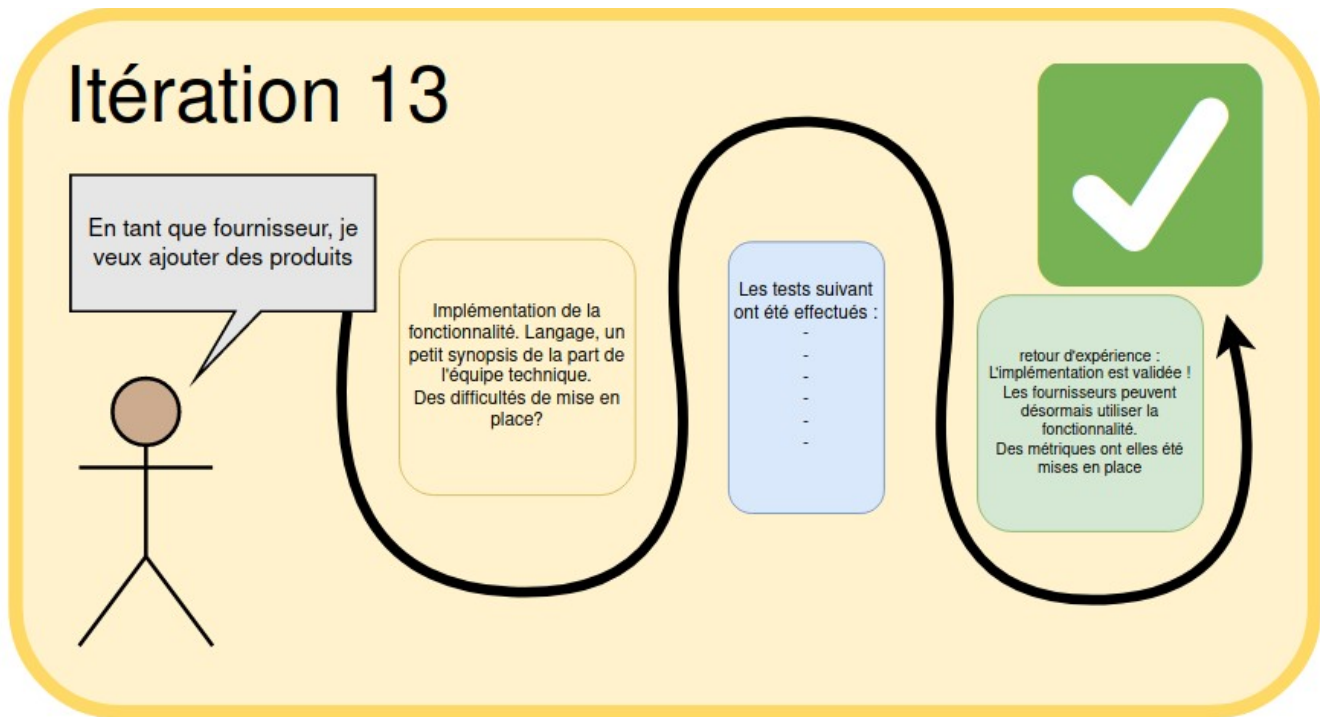


Figure 1: Exemple d'infographie de communication

## Rythme de communication

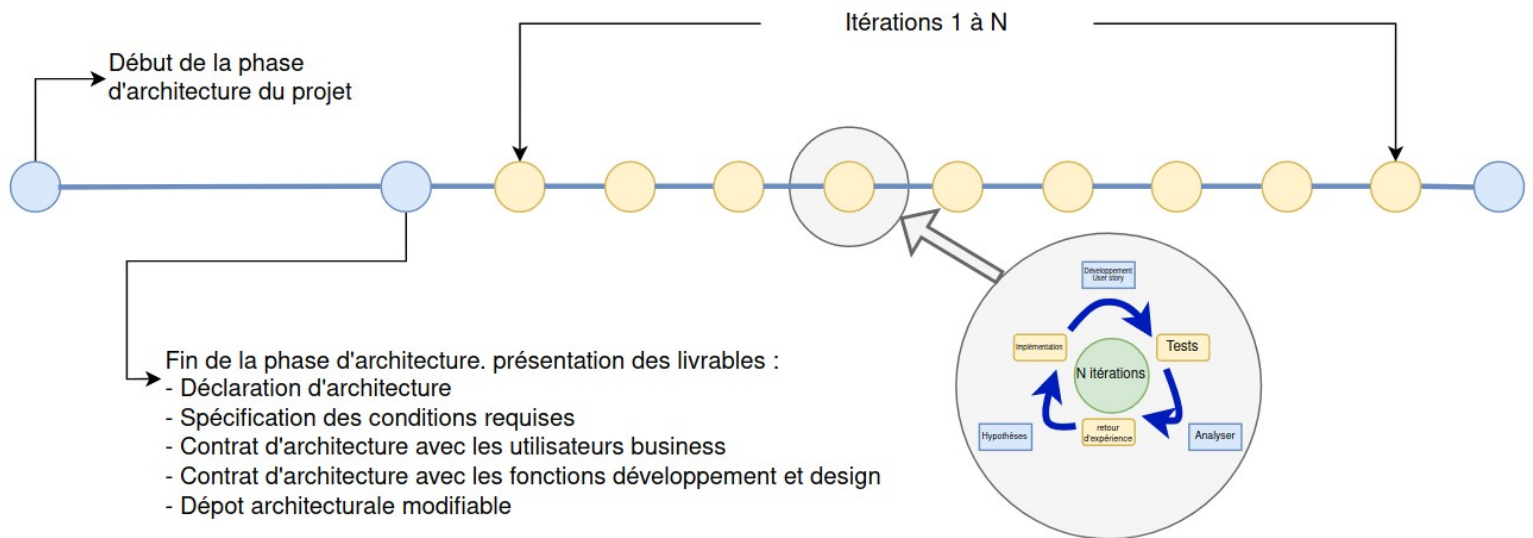
Dès la fin de la phase de déclaration de l'architecture, deux communications seront misent en place.

La première, exclusivement au niveau de l'équipe de développement, sera sous forme de réunions debout (stand-up meeting). Celle ci devra avoir lieu quotidiennement.

La seconde, par e-mail. La communication suivra le rythme des différentes itérations.



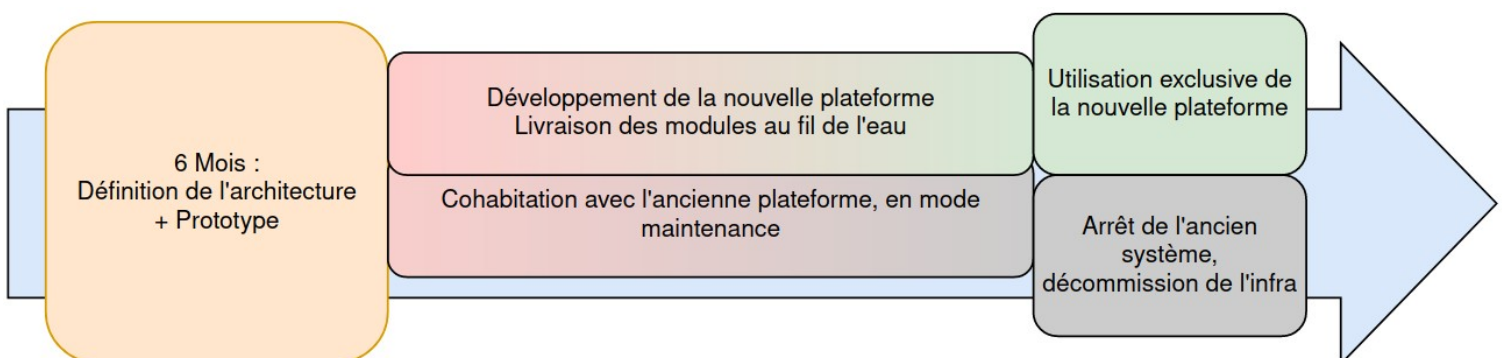
## Foosus – Projet *géoconscient V2*

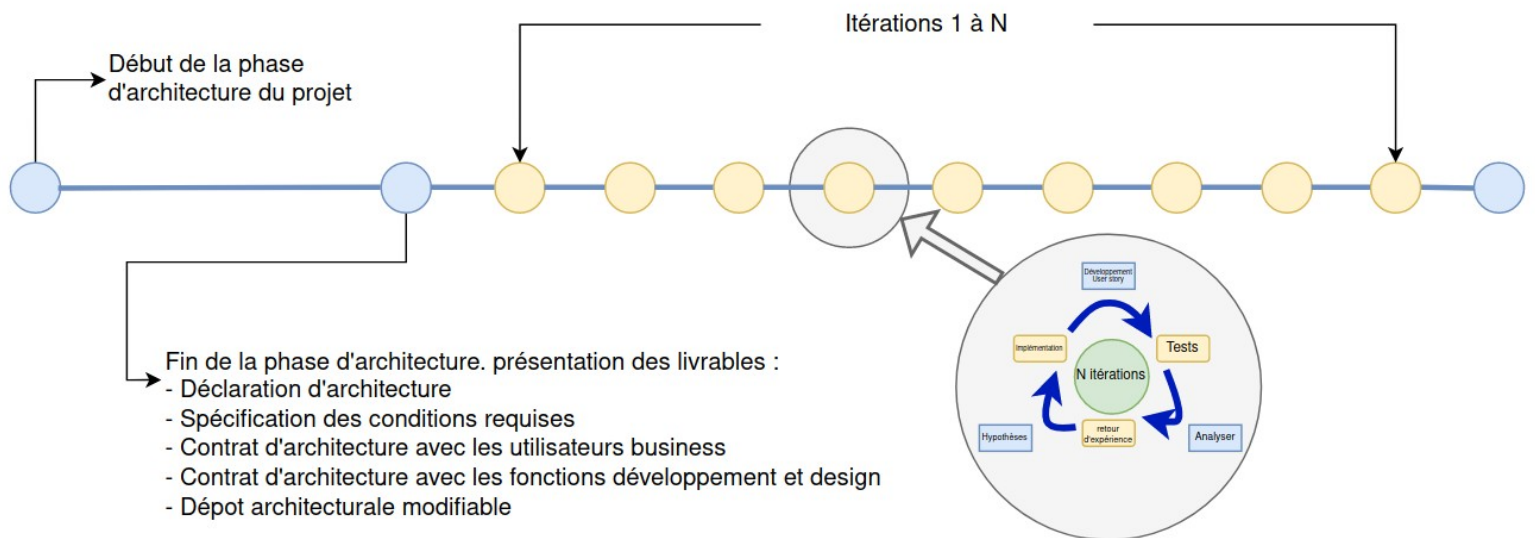


Chaque itérations, à la suite de la phase de retour d'expérience, devra se terminer par une communication adaptée aux différentes parties prenantes sous la forme d'un courrier électronique.

## Collaboration

### Plan et calendrier du projet







# Risques et facteurs de réduction

## Analyse des risques

Nature de risque	Description	Impact	Probabilité	Criticité	Conséquence si avéré	Responsable	Actions préventives	Actions correctrices
Projet qui n'aboutit pas	Le développement fini par être interrompu. Le nouveau projet n'aboutit pas	Catastrophique	Peu probable	Très critique	Perte financière Mise en risque de la société Nouveau projet à recréer de zéro	Architecte logiciel Ash Callum (CEO)	Supervision du projet par l'architecte logiciel Les parties prenantes n'ont pas un pouvoir décisionnel mais de consultation Animation de l'équipe de développeur quotidienne	Redefinition d'un projet en prenant en compte les éléments d'un retour d'expérience
Itérations trop courtes	Le découpage en user stories est correct, mais chaque US met plus d'une itération à être développée	Majeur	Peu probable	Critique	Perte de temps dans le projet, frustration des équipes de développeur et des équipes business	Architecte logiciel Peter Parker (CTO)	Bien dimensionner les itérations en avance de phase S'assurer que les US soient le plus simple possible L'animation de l'équipe de développeur doit détecter les risques au plus tôt	Augmentation de la durée de l'itération Passage de 5 équipes de 3 développeurs à 3 équipes de 5 développeurs Analyse des US
Équipe business insatisfaite	L'équipe business n'est pas satisfaite du travail livré par les développeurs	Grave	Peu probable	Critique	Perte de temps dans le projet, frustration des équipes de développeur et des équipes business	Peter Parker (CTO) Daniel Anthony (CPO) Natasha Jarson (CIO)	Bien réaliser les Users Stories La communication entre les parties prenantes doit être pertinente pour permettre à tous de comprendre les besoins Les plans de tests business doivent être pertinents	Revoir les plans de tests fonctionnels Augmenter la communication entre les parties prenantes
Baisse de motivation des développeurs	Les développeurs étaient habitués à beaucoup de libertés et de possibilité d'expérimentation	Catastrophique	Peu probable	Très critique	Difficulté à terminer les US durant l'itération Augmentation du risque de démission et de la perte de ressources humaines	Peter Parker (CTO)	Motivation via la culture d'entreprise Possibilité d'investissement des différents acteurs du développement Mise en avant de leurs qualités de développeurs Utilisation de méthodes AGILE et Lean	Création d'une équipe « POC », qui changerait de manière régulière dans le temps
Processus qualité mal défini	Les processus qualité ont mal été définis	Grave	Probable	Très critique	Plateforme ne correspondant pas aux besoins Retard dans le projet Perte définitive de la confiance client	Ash Callum (CEO) Architecte Logiciel Peter Parker (CTO) Daniel Anthony (CPO)	Bien identifier les processus en amont Mettre en place des principes issus de la qualité	Faire appel à un auditeur externe pour redéfinir les processus
Difficultés de mise en place de l'architecture micro-services	Les développeurs, non habitués à l'architecture micro-services, pourraient avoir des difficultés pour la mettre en place	Mineur	Peu probable	Modérée	Perte de temps dans les premières itérations Sentiment de frustration	Peter Parker (CTO)	S'assurer de la bonne formation des collaborateurs chargés du développement	Faire appel à la prestation externe Formation des collaborateurs
Incompréhension/manque de dialogue business/développeur	Un manque de communication ou un manque de dialogue entre les parties business et l'équipe chargée du développement	Grave	Probable	Très critique	Plateforme ne correspondant pas aux besoins Retard dans le projet Frustration des parties prenantes	Ash Callum (CEO) Architecte Logiciel Peter Parker (CTO) Daniel Anthony (CPO)	Rédaction des Users Stories avec l'équipe business, en étroite collaboration de l'équipe chargée du développement Rédaction et application des tests fonctionnels/métier par les équipes business	Arbitrage de la part du CEO

## Hypothèses



## Foosus – Projet *géoconscient V2*

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

I D	Hypothèse	Impact	Propriétaire
1	Aucune nouvelle fonctionnalité ne sera développée	Obligation de garder l'ancien système en mode de maintenance.  Chaque micro service ajoutés devront prendre le pas sur l'ancienne plateforme	Architecte logiciel Peter Parker (CTO)
2	Technologies actuelles et capacité à s'adapter aux technologies futures	L'architecture devra être en capacité d'accepter l'utilisation de technologies non existantes actuellement.  Là encore, l'usage ne micro service permet de séparer chaque fonctionnalités, qui deviens totalement indépendante technologiquement	Architecte logiciel Pater Parker (CTO)
3	Pas de faux raccourcis		Architecte logiciel
4	Montée en puissance empirique du volume d'utilisateur qui migrerons vers la nouvelle plateforme	Chaque micro service devra être en capacité de travailler avec l'ancien système  Les utilisateurs devront pouvoir être migrés fonctionnalité par fonctionnalité	Architecte logiciel Peter Parker (CTO) Natasha Jarson (CIO)



5	Nouvelles fonctionnalités introduites grâce à la géolocalisation	L'architecture en micro service rend l'ajout de nouvelles fonctionnalité totalement possible, sans interruption de service, et permet l'innovation	Architecte logiciel Pater Parker (CTO)
6	Approche architecturale type Lean	Une approche mêlant Lean et Agile, mais totalement sur mesure pour être adapté aux équipes et au projet sera mise en place	Architecte logiciel Pater Parker (CTO) Natasha Jarson (CIO)
7	Impact environnemental	Développement respectant les bonnes pratique et s'incluant dans une démarche Green IT	Architecte logiciel Pater Parker (CTO) Natasha Jarson (CIO)

## Critères d'acceptation et procédures

### Métriques et KPIs

De plus, les métriques suivantes seront utilisées pour déterminer le succès de ce travail d'architecture :

Métrique	Technique de mesure	Valeur cible	Justification	Notes supplémentaires
Nombre d'adhésion par jour		+10 %		
Adhésion de producteurs alimentaires		4 par mois	Actuel : 1,4 par mois	



Délais moyen de parution		<1 semaine	Actuel : 3,5 semaine	
Taux d'incident de prod P1		< 1 par mois	Actuel >25 par mois	

## Procédure d'acceptation

---

La signature du contrat de conception de de développement de l'architecture signifie pour toutes les parties prenantes signataires une acceptation complète de l'architecture.

Toute demande de changement du document doit faire appel à la procédure de changement de périmètre.

---





## Approbations signées

---

Valideur	Domaine de responsabilité	Date
<i>Ash Callum</i>	<i>Domaine business</i>	
<i>Natasha Jarson</i>	<i>Domaine technique</i>	
<i>Daniel Anthony</i>	<i>Domaine Business</i>	
<i>Peter Parker</i>	<i>Domaine technique</i>	