

# Spécification des conditions requises pour l'architecture





## Table des matières

<b>Objet du document .....</b>	<b>3</b>
<b>Mesure du succès .....</b>	<b>4</b>
<b>Conditions requises pour l'architecture .....</b>	<b>5</b>
<b>Contrats de service .....</b>	<b>6</b>
<b>Accords de niveau de service .....</b>	<b>6</b>
<b>Diminution des incidents en production .....</b>	<b>6</b>
<b>Diminution du délai moyen de parution .....</b>	<b>6</b>
<b>Diminution des lourdes pannes systèmes succédant aux évolutions .....</b>	<b>7</b>
<b>Augmentation de la disponibilité .....</b>	<b>8</b>
<b>Augmentation des adhésions des utilisateurs .....</b>	<b>8</b>
<b>Augmentation des adhésions des producteurs .....</b>	<b>9</b>
<b>Lignes directrices de l'implémentation .....</b>	<b>10</b>
<b>Expressions de Foosus .....</b>	<b>10</b>
<b>Première phase : Définition du style architectural .....</b>	<b>10</b>
<b>Seconde phase : Migration des composants .....</b>	<b>11</b>
<b>Remarques .....</b>	<b>11</b>
<b>Spécifications d'implémentation .....</b>	<b>12</b>
<b>Principes généraux .....</b>	<b>12</b>
<b>Principes business .....</b>	<b>12</b>
<b>Principes data .....</b>	<b>12</b>
<b>Principes d'application .....</b>	<b>13</b>
<b>Principes technologiques .....</b>	<b>13</b>
<b>Standards d'implémentation .....</b>	<b>14</b>
<b>Intégration continue .....</b>	<b>14</b>
<b>Livraison continue .....</b>	<b>14</b>
<b>Déploiement continu .....</b>	<b>15</b>



<b>Interopérabilité .....</b>	<b>16</b>
<b>Communication .....</b>	<b>16</b>
<b>Format des données échangées .....</b>	<b>16</b>
<b>Compatibilité avec le système actuel .....</b>	<b>16</b>
<b>Contraintes et directives .....</b>	<b>17</b>
<b>Contraintes et directives générales .....</b>	<b>17</b>
<b>Taux d’inscriptions utilisateurs .....</b>	<b>17</b>
<b>Innover dans le périmètre d’une architecture d’entreprise .....</b>	<b>17</b>
<b>Soutenir l’innovation technique rapide et l’expérimentation .....</b>	<b>18</b>
<b>Visibilité de la plateforme .....</b>	<b>18</b>
<b>Améliorer la réputation de Foosus sur le marché grâce à la stabilité .....</b>	<b>18</b>
<b>Hypothèses .....</b>	<b>19</b>
<b>Approbations .....</b>	<b>20</b>



## Objet de ce document

Ce document, de spécification des conditions requises pour l'architecture, fournit un ensemble de déclarations quantitatives qui dessinent ce que doit faire un projet d'implémentation afin d'être conforme à une quelconque architecture.

Plus précisément dans ce document nous allons trouver les indicateurs clés de performance qui ont été émis par l'entreprise à l'origine du projet, Foosus, et qui sont primordiaux dans l'évaluation du succès du projet dans sa mise en œuvre architecturale.

Ce document nous permet également de prendre connaissances des conditions, fixées par Foosus, à prendre en compte dans la mise en place d'une solution et dont les objectifs, en découlant, doivent être obligatoirement atteints.

Par la suite, nous verrons les engagements, que doit remplir une quelconque solution, au travers des contrats de niveau de service. Section assez proche de la partie sur les indicateurs clés de performance, cependant il faut garder à l'esprit que les indicateurs clés de performance permettent de mesure le succès d'un système, les contrats de niveau de service sont, comme le nom l'indique, des contrats, donc des engagements devant être respectés avec des objectifs clairs et précis dont la mesure est nécessaire pour évaluer du succès de l'engagement.

Ce document permet également de mettre au clair les lignes directrices liées à l'implémentation, c'est à dire la manière dont va être mis en place le système, en fournissant une stratégie qui s'appuie sur les expressions émises par Foosus.

Ensuite, nous abordons les conditions d'implémentation comprenant l'ensemble des directives, quelle que soit leur nature, à prendre en compte dans le cadre de la réalisation de ce projet.

Une vue d'ensemble des standards d'implémentation est également proposée, ces standards regroupent l'ensemble des bonnes pratiques qui sont fortement conseillées de mettre en place pour une réalisation optimisée sur de nombreux niveaux.

Un approfondissement des conditions d'interopérabilités est également inscrit dans ce document, que ce soit au niveau de la communication, du format des données communiqués ou encore des éventuelles compatibilités de la nouvelle solution et du système actuel.

Puis nous verrons les différentes contraintes et les multiples directives imposés et, par conséquent, à prendre en compte dans la conception ainsi que l'implémentation de l'intégralité du système.

Enfin, nous verrons les hypothèses, quelle que soit leur nature, exprimées par Foosus ainsi que celles que nous émettons à notre tour concernant le style architectural à sélectionner pour répondre, au mieux, aux différentes exigences, contraintes, souhaits et autres objectifs impliqués dans ce projet.



## Mesures du succès

Bien que les engagements, mesurables, devant être respectés dans la réalisation de ce projet vont être davantage développés, dans ce même document, à la suite de cette partie, il reste important de traiter la section se rapportant aux indicateurs clés de performance qui serviront directement à Foosus pour évaluer le succès du projet dans sa globalité.

On pourra donc retrouver ces indicateurs clés de performance dans les accords de niveau de service afin de renforcer l'engagement à leur réalisation, rendant, alors, leur accomplissement prioritaire.

Pour résumer ces mesures de succès, on peut dire que ce sont des critères quantitatifs pour l'évaluation de la réussite globale de la mise en œuvre architecturale.

Ci-dessous, un tableau récapitulatif, de manière exhaustive, les indicateurs clés de performance qui sont spécifiés par Foosus. On peut remarquer 4 colonnes :

- Métrique, nous renseigne du titre d'un critère de performance
- Mesure, nous enseigne de la manière dont un critère va être mesuré
- Valeur initiale, nous informe de la valeur actuelle mesurée pour un critère
- Valeur cible, nous spécifie de la valeur cible à mesurer pour un critère, au terme de ce projet

Métrique	Mesure	Valeur initiale	Valeur cible
Adhésion journalière (utilisateur)	Requête SQL	NC	(+) 10%
Adhésion journalière (producteur)	Requête SQL	1,4/mois	4/mois
Délai de déploiement	Empirique	3,5 semaines	> 1 semaine
Incident de production	Empirique	> 25/mois	> 1/mois

### Mesures du succès

Dans ce tableau, nous pouvons remarquer que 4 indicateurs sont présents, de plus, il nous a été spécifié par Foosus que les inscriptions constituent une métrique clé aux yeux de leurs investisseurs.

Il est à noter que nous parlons des inscriptions des utilisateurs comme des producteurs et que ces inscriptions ne peuvent être améliorées que par l'agilité nécessaire permettant d'innover rapidement et d'expérimenter avec des variantes d'offres produit existantes.

Il est également important de mettre en exergue que le fait d'innover rapidement passe par un faible délai de déploiement, qui est, par ailleurs, la troisième métrique du tableau ci-dessus.

En ce qui concerne la diminution du taux d'incident de production, quatrième métrique, c'est un indicateur clé afin de ne pas perturber les utilisateurs utilisant le système et ainsi éviter toute frustration qui pourrait entraîner, à son tour, une fuite vers la concurrence et, par conséquent, réduire les deux premières métriques qui se rapportent aux adhésions.



## Conditions requises pour l'architecture

Afin de rendre son application la plus attractive possible et ainsi gagner un grand nombre d'utilisateurs, de tous bords, au travers d'un système sécurisé et fiable, Foosus a défini de multiples conditions avec plusieurs objectifs à atteindre qui en découlent.

En premier lieu, il souhaite tirer parti de la géolocalisation afin relier des fournisseurs et des consommateurs, ainsi ils proposeront des produits disponibles près des lieux de résidence de ces derniers. Un calculateur de distance devra être inclus pour permettre aux consommateurs de trouver les fournisseurs les plus proches.

Ils désirent également une architecture évolutive, scalable et résiliente, qui est, à la fois, capable de supporter un déploiement de leurs services sur diverses régions, pays et villes et des pays donnés, mais également en mesure de s'adapter à un grand nombre d'utilisateurs simultanément sans causer de quelconque problème. L'architecture doit pouvoir s'équiper d'outils permettant une stratégie de scalabilité avancée ainsi qu'un déploiement efficace, ayant la capacité d'être distribué au besoin.

De cette diversité géographique, un nouvel aspect entre en jeu, les arrêts du système volontaires doivent être supprimés, effectivement, des décalages horaires résultent le fait que le système est amené à être utilisé tout au long des 24 heures journalières, c'est pourquoi, Foosus a bien spécifié que les améliorations et autres modifications apportées aux systèmes de production doivent limiter ou supprimer la nécessité d'interrompre le service pour procéder au déploiement.

Il est aussi à mettre en exergue le fait que leurs fournisseurs et leurs consommateurs doivent pouvoir accéder à leur solution où qu'ils se trouvent sans perte de performance. Cette solution doit donc être utilisable avec des appareils fixes comme mobiles et elle doit tenir compte des contraintes de bande passante pour les réseaux cellulaires et les connexions Internet haut débit.

En d'autres termes, Foosus ne recherche pas une solution désuète mais plutôt tendancielle et bénéficiant d'une optimisation au niveau réseau, ce qui pourrait se traduire par de multiples manières tel que l'utilisation accrue de la mise en cache, la limitation des données superflues lors des échanges, des images de qualité standard ou encore l'utilisation, pour la partie web, d'une « Single Page Application », une page se chargeant une seule et unique fois, de façon à éviter les rechargements inutiles aux changements de page.

En outre, le nouveau système doit réaliser, au moins, les mêmes fonctionnalités que l'actuel, comme par exemple, pouvoir prendre en charge divers types d'utilisateurs, tel que des fournisseurs, des consommateurs, des administrateurs, et ceci avec des fonctionnalités et des services spécifiques pour ces catégories.

Enfin, les livrables doivent pouvoir être fournis à intervalles réguliers pour que le nouveau système soit rapidement opérationnel et puisse être doté de nouvelles fonctionnalités au fil du temps. Il faudrait donc, à minima, instaurer la livraison continue, si possible, le déploiement continu.



# Contrats de service

## Accords de niveau de service

Certains des éléments présents, ci-dessous, dans les accords de niveau de service ont été évoqués au cours de la partie détaillant les mesures du succès du système et ont été, pour rappel, directement imposés par Foosus dans le cadre du projet afin de satisfaire aux demandes de leurs investisseurs.

### 1. Diminution des incidents en production

Parmi les accords de niveau de service, nous allons retrouver l'engagement à la diminution des incidents survenant en phase de production. Le respect de cet engagement est sous la houlette du directeur informatique ainsi que plusieurs responsables, pouvant varier en fonction de la nature de l'incident, le responsable des ingénieurs, le responsable des infrastructures et l'architecte logiciel.

#### Objectif de niveau de service

Le taux mensuel d'incident est aujourd'hui inférieur à 25 et, dans le cadre de ce projet, doit être ramené à un taux mensuel inférieur à 1. On peut changer l'unité de cet objectif et ainsi dire qu'il faut passer de d'un taux annuel inférieur à 300 à un taux annuel inférieur à 12.

Cet accord de niveau de service, faisant partie des critères de réussite imposés par Foosus, démontre, de par son objectif, une attente considérable vis-à-vis de ce projet qui doit réussir à réduire les incidents en production de 96%.

#### Indicateurs de niveau de service

La mesure de cet objectif se fera sur l'observation des logs, c'est à dire de manière empirique, la mise en place d'un envoi automatique de courriel est nécessaire afin de prévenir les parties concernées lorsqu'un incident se produit. La comptabilisation ainsi que l'historique des incidents incombent à la partie concernée la plus haute dans la hiérarchie et ceci de la manière qu'elle désire.

### 2. Diminution du délai moyen de parution

Parmi les accords de niveau de service, nous allons retrouver l'engagement à la diminution du délai moyen de parution. Le respect de cet engagement est à la charge du directeur informatique ainsi que le responsable des ingénieurs, le responsable des infrastructures et l'architecte logiciel.



### **Objectif de niveau de service**

Le délai moyen de parution est aujourd'hui de 3,5 semaines et, dans le cadre de ce projet, doit être ramené à un délais moyen inférieur à 1 semaine.

Cet accord de niveau de service, faisant partie des critères de réussite imposés par Foosus, démontre, de par son objectif, une attente considérable vis-à-vis de ce projet qui doit réussir à réduire le délai moyen de parution d'environ 71,4%.

### **Indicateurs de niveau de service**

La mesure de cet objectif se fera sur l'observation et l'expérience, c'est à dire de manière empirique, des procédures seront mises en place et des outils spécialisés seront utilisés afin d'optimiser le délai moyen de parution.

Une constatation des résultats sera effectuée et consignée dans un registre tenu par la partie concernée la plus haute dans la hiérarchie, et ceci de la manière qu'elle dédire. Cette consignation entrainera une étude qui permettra de révéler la nécessité, ou non, de réviser la stratégie.

## **3. Diminution des lourdes pannes systèmes succédant aux évolutions**

Parmi les accords de niveau de service, nous allons retrouver l'engagement à la diminution des pannes conséquente du système survenant à la suite d'une quelconque évolution. Le respect de cet engagement est sous la coupe du directeur informatique ainsi que plusieurs responsables, pouvant varier en fonction de la nature de la panne, le responsable des ingénieurs, le responsable des infrastructures et l'architecte logiciel.

### **Objectif de niveau de service**

Le taux annuel dernier relatif aux pannes dues aux évolutions était de 12 et, dans le cadre de ce projet, doit être ramené à un taux annuel nul.

Cet accord de niveau de service, ne faisant pas partie des critères de réussite imposés par Foosus mais étant implicitement souligner lors des descriptifs du projet, démontre, de par son objectif, une grande attente en ce qui concerne ce projet qui doit permettre une réduction des pannes paralysant le système, suite à une mise à jour ou une évolution, de 100%.

### **Indicateurs de niveau de service**

La mesure de cet objectif se fera aussi de manière empirique, soit par la consultation de logs, soit par simple observation, effectivement, à contrario des incidents légers, il arrive que l'ensemble d'un système ne fonctionne plus suite à une évolution, ce qui n'est pas difficile à remarquer.





La mise en place d'un envoi automatique de courriel peut être envisagée afin de prévenir les parties concernées lorsqu'une panne se produit. La comptabilisation ainsi que l'historique des incidents incombent à la partie concernée la plus haute dans la hiérarchie et ceci de la manière qu'elle désire.

#### **4. Augmentation de la disponibilité**

Parmi les accords de niveau de service, nous allons retrouver l'engagement à l'augmentation de la disponibilité du système. Le respect de cet engagement est à la charge du directeur informatique ainsi que du responsable des ingénieurs, du responsable des infrastructures et de l'architecte logiciel.

##### **Objectif de niveau de service**

Bien que le taux de disponibilité moyen actuel ne soit pas connu, il est évident qu'avec le système présent qui est peu fiable, il ne doit pas s'en trouver très élevé. Avec ce projet, nous visons un taux moyen de disponibilité de 99%.

Cet accord de niveau de service, ne faisant pas partie des critères de réussite imposés par Foosus mais étant implicitement souligner lors des descriptifs du projet, démontre, de par son objectif, une large attente en ce qui concerne ce projet qui doit permettre d'atteindre un maximum de 63 heures et 36 minutes d'indisponibilité.

##### **Indicateurs de niveau de service**

La mesure de cet objectif se fera aussi de manière empirique, soit par la consultation de logs, soit par simple observation, bien que la consultation des logs soit préconisée pour un calcul bien plus fiable. Il est à noter que chacun des services proposés par le système a son propre taux de disponibilité et que le seuil de 99% à atteindre est une moyenne de l'ensemble des taux des différents services.

La mise en place d'un envoi automatique de courriel peut être envisagée afin de prévenir les parties concernées lorsqu'un service n'est plus disponible. La comptabilisation ainsi que l'historique des indisponibilités incombent à la partie concernée la plus haute dans la hiérarchie et ceci de la manière qu'elle désire.

#### **5. Augmentation des adhésions des utilisateurs**

Parmi les accords de niveau de service, nous allons retrouver l'engagement à l'augmentation des adhésions des utilisateurs. Le respect de cet engagement est sous le joug, pour la partie technique, du directeur informatique ainsi que du responsable des ingénieurs, du responsable des infrastructures et de l'architecte logiciel, mais également, pour la partie non technique, du directeur produit et du directeur marketing.



### **Objectif de niveau de service**

Le nombre d'adhésion journalière d'utilisateur n'a pas été communiqué, il est simplement spécifié un désire d'augmentation d'adhésion de 10%.

Cet accord de niveau de service, faisant partie des critères de réussite imposés par Foosus, démontre une attente précise de ce projet, en effet, c'est un argument de choix vis-à-vis des investisseurs.

### **Indicateurs de niveau de service**

La mesure de cet objectif se fera quotidiennement et automatiquement a grâce à la mise en place d'un script qui générera un rapport, en s'appuyant, notamment, sur l'utilisation de requêtes SQL.

La mise en place d'un envoi automatique de courriel doit être réalisé à l'issu de la génération du rapport afin de le transmettre à l'ensemble des parties investies.

## **6. Augmentation des adhésions des producteurs**

Parmi les accords de niveau de service, nous allons retrouver l'engagement à l'augmentation des adhésions des producteurs alimentaires. Le respect de cet engagement est sous la responsabilité, pour la partie technique, du directeur informatique ainsi que du responsable des ingénieurs, du responsable des infrastructures et de l'architecte logiciel, mais également, pour la partie non technique, du directeur produit et du directeur marketing.

### **Objectif de niveau de service**

Le nombre d'adhésion mensuel des producteurs alimentaires est de 1,4/mois. Foosus souhaite, à travers ce projet, une augmentation des adhésions à 4/mois.

Cet accord de niveau de service, faisant partie des critères de réussite imposés par Foosus, démontre une attente précise de ce projet, en effet, en plus d'être un argument de choix vis-à-vis des investisseurs, Foosus attend une augmentation de 286% de son nombre d'adhésion mensuel de producteurs alimentaires.

### **Indicateurs de niveau de service**

La mesure de cet objectif se fera mensuellement et automatiquement a grâce à la mise en place d'un script qui générera un rapport, en s'appuyant, notamment, sur l'utilisation de requêtes SQL.

La mise en place d'un envoi automatique de courriel doit être réalisé à l'issu de la génération du rapport afin de le transmettre à l'ensemble des parties investies.



## **Lignes directrices de l'implémentation**

### **Expressions de Foosus**

Les expressions de Foosus sont, pour rappel, de manière simplifiée, une arrivée rapide sur le marché avec un système ergonomique et fonctionnel étant doté des diverses fonctionnalités liées à la localisation ainsi que pouvant supporter de nombreux utilisateurs, quel que soit le support qu'ils utilisent, sans aucune perte de performance, quelle que soit la région où ils se trouvent.

Il est à ajouter à ces énumérations que Foosus souhaite la mise en place d'une nouvelle architecture, permettant de réaliser des évolutions rapides du système et devant remplacer l'intégralité de la solution utilisée actuellement par l'entreprise.

L'esprit de rapidité désiré par Foosus, visant à supplanter la concurrence sur la vente de produits entre clients et producteurs locaux, rentre d'une certaine manière en contradiction avec son souhait de migration de l'intégralité du système actuel.

Effectivement, ce système est fonctionnel, en revanche il ne prend simplement pas en charge les fonctionnalités liées à la localisation tel que la recherche et le tri de produits en fonction d'un rayon autour d'une adresse. Les fonctions liées à la localisation sont, parmi d'autres, au cœur des revendications exprimées par Foosus.

C'est pourquoi, au vu de ces constatations et dans un esprit de cohérence par rapport à ce qui est évoqué ci-dessus, nous proposons, comme lignes directrices pour l'implémentation d'une quelconque solution, un découpage en du projet en plusieurs phases.

### **Première phase : Définition du style architectural**

Une première phase définissant le nouveau style architectural qui va être sélectionné, ainsi que les premiers composants primordiaux qui le composent tel que le service de localisation ou le service de recherche. Cette première phase est tout à fait cohérente car elle permet bien de mettre en place un système de manière rapide afin de tout mettre en œuvre pour supplanter la concurrence et, de même temps, qui soit doté des fonctionnalités phares de l'entreprise.

Néanmoins, il est tout de même à préciser que pour que cette première phase produise un système fonctionnel, il faut que ce système s'appuie sur la solution existante afin d'élargir ses fonctionnalités en se reposant, pour le moment, sur ce qui est déjà implémenté actuellement tel que le système de gestion des utilisateurs ou encore le système de paiement.



## Seconde phase : Migration des composants

Une seconde phase de migration, succédant à la première phase, pour, comme son nom l'indique, migrer, un à un, les composants du système initial vers le nouveau style architectural défini. Les composants à migrer comporteront au moins les mêmes fonctionnalités une fois migrés.

On parle de migration de composants vers la nouvelle architecture, cependant, il serait, sans doute, plus approprié de parler de refonte de composants. En effet, les technologies employées dans le système actuel sont hétéroclites, ce qui engendre de nombreux problèmes ainsi qu'une augmentation significative des coûts, il est donc du désir de Foosus de nous voir y apporter une certaine standardisation.

## Remarques

Il est à noter que, quelle que soit la phase, Foosus souhaite vivement prioriser l'utilisation d'outils open source, il est important de rappeler que dans certains cas, ce n'est, hélas, pas possible. Afin qu'un outil gratuit soit fiable, il faut qu'il bénéficie d'un large support d'une communauté ou d'une entreprise qui le met à disposition à titre gracieux, ce qui est, évidemment, assez rare.

Il est également important de mettre en exergue, pour que notre approche soit comprise de tous dans sa globalité, que Foosus nous a spécifié que leurs équipes techniques passaient la majeure partie de leur temps à éteindre des incendies tant le système est désuet et mal optimisé, par conséquent, il ne souhaite pas qu'on le fasse évoluer, il doit être, impérativement, passé en mode maintenance, jusqu'à son remplacement total.



# Spécifications d'implémentation

Au cours de ce chapitre, nous allons énumérer les listes des différentes spécifications de l'implémentation dont nous devons tenir compte et ceci de la manière la plus générale possible, afin de couvrir le maximum de facettes qui y sont liées, plus précisément, aux différents aspects et principes devant être respectés dans la réalisation de ce projet.

## Principes généraux

Il y a 4 principes généraux à prendre en compte :

- Décisions pilotées par le feed-back et l'apprentissage.
- Faire des choix qui soutiennent les objectifs 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

Il y a 2 principes business à prendre en compte :

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

## Principes data

Il y a 5 principes data à prendre en compte :

- 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 partons 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

Il y a 4 principes d'application à prendre en compte :

- 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

Il y a 6 principes technologiques à prendre en compte :

- Faire des choix ouverts et aisés à modifier.
- Les choix de construction vs 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.



## Standards d'implémentation

Cette partie est consacrée aux standards à respecter pour l'implémentation d'une quelconque solution. On peut voir cette section comme les bonnes pratiques non techniques de développement. Il sera donc laissé de côté tout ce qui attrait directement au code source (style d'écriture, qualité, métriques, couplage des classes, instructions à éviter, etc.), à sa documentation ainsi qu'aux règles de conception.

### Intégration continue

Cette méthode repose souvent sur la mise en place d'une brique logicielle permettant l'automatisation de tâches : compilation, tests unitaires et fonctionnels, validation produit, tests de performances... À chaque changement du code, cette brique logicielle va exécuter un ensemble de tâches et produire un ensemble de résultats, que le développeur peut par la suite consulter. Cette intégration permet ainsi de ne pas oublier d'éléments lors de la mise en production et donc ainsi améliorer la qualité du produit final.

Pour appliquer cette technique, il faut d'abord que :

- Le code source soit partagé (en utilisant des logiciels de gestion de versions tels que CVS, Subversion, git, Mercurial, etc.) ;
- Les développeurs intègrent (commit) quotidiennement (au moins) leurs modifications ;
- Des tests d'intégration soient développés pour valider l'application ;
- Un outil d'intégration continue est ensuite nécessaire, tel que CruiseControl ou Jenkins. D'autres outils, comme SonarQube ou Jacoco, peuvent être mis en place afin de superviser la qualité du code.

Les principaux avantages d'une telle technique de développement sont :

- Le test immédiat des modifications ;
- La notification rapide en cas de code incompatible ou manquant ;
- Les problèmes d'intégration sont détectés et réparés de façon continue, évitant les problèmes de dernière minute ;
- Une version est toujours disponible pour un test, une démonstration ou une distribution.

### Livraison continue

La livraison continue est une approche d'ingénierie logicielle dans laquelle les équipes produisent des logiciels dans des cycles courts, ce qui permet de le mettre à disposition à n'importe quel moment. Le but est de construire, tester et diffuser un logiciel plus rapidement.



L'approche aide à réduire le temps d'évaluation des risques, et les risques associés à la livraison de changement en adoptant une approche plus incrémentielle des modifications en production. A terme, l'objectif est une réduction du coût. Un processus simple et répétable de déploiement est un élément clé

## Déploiement continu

Le déploiement continu, est une approche d'ingénierie logicielle dans laquelle les fonctionnalités logicielles sont livrées fréquemment par le biais de déploiements automatisés. Le déploiement continu diffère de la livraison continue, une approche similaire, détaillée ci-dessus, dans laquelle des fonctionnalités logicielles sont également livrées fréquemment et considérées comme pouvant potentiellement être déployées, mais qui ne le sont pas pour autant, effectivement le déploiement restant dans ce cas un processus manuel.





# Interopérabilité

Les conditions requises se rapportant à l'interopérabilité techniques ne seront que très brièvement détaillé dans cette section. Il reste évident qu'un système répondant aux différentes problématiques liées à l'interopérabilité sera produit. Il n'en reste pas moins que nous allons, tout de même, définir quelques normes devant être respectées.

## Communication

Les solutions monolithiques étant dépassées aujourd'hui, il est évident qu'un système découpé en plusieurs parties, devant communiquées entre elles, sera préconisée. Il devient alors important de nous établir sur le choix des protocoles permettant l'échange d'informations.

Les interactions internes comme externes devront se faire à l'aide du protocole de communication HTTPS, combinaison du célèbre protocole HTTP avec une couche de chiffrement SSL ou TLS, permettant alors de vivement amplifier la sécurité des données transitant à travers le système.

## Format des données échangées

Aujourd'hui un des formats les plus communément utilisé dans les échanges de données textuelles est le format JSON. Effectivement, bien que souffrant de quelques limitations comme, par exemple, dans la représentation des types généraux, il bénéficie, tout de même, d'une simplicité conséquente, d'une grande légèreté ainsi que de nombreux autres avantages.

Les types de données représentés par le JSON sont suffisamment génériques et abstraits pour, d'une part, pouvoir être représentés dans n'importe quel langage de programmation et d'autre part, pouvoir représenter n'importe quelle donnée concrète.

## Compatibilité avec le système actuel

La compatibilité entre la nouvelle solution et le système actuel est également au cœur des problématiques. Il faut veiller à bien étudier les différents protocoles de communication ainsi que les formats de données utilisés de manière à produire une solution qui en tient compte afin de mener avec succès et sans mauvaise surprise la première phase du projet.

Pour aller plus loin dans ce sens, il est également important de rappeler que la solution actuelle est très hétéroclite technologiquement, impliquant, potentiellement, une large variété dans ses protocoles de communication et ses formats de données.



## Contraintes et directives

### Contraintes et directives générales

Ci-après figure une liste des contraintes et directives générales relatives au projet :

- Le projet initial est approuvé pour un coût de 50 000 USD (45 190 €) et une période de 6 mois est prévue pour définir l'architecture et préparer un projet de-suivi afin de développer un prototype.
- L'architecture doit permettre d'obtenir le meilleur rapport qualité-coût.
- L'architecture peut inclure de nouveaux composants personnalisés ou des composants du commerce pour favoriser la flexibilité, la stabilité et l'extensibilité.
- L'objectif de cette phase du projet étant la définition de l'architecture, des projets de suivi seront créés pour compléter les détails avec les équipes internes.
- Les solutions open source sont préférables aux solutions payantes.
- Le support continu des composants doit être pris en compte lors de leur sélection ou lors des prises de décision de création ou d'achat.
- Toutes les solutions du commerce ou open source doivent, dans la mesure du possible, faire partie d'une même pile technologique afin de réduire les coûts de maintenance et de support continus.

### Taux d'inscriptions utilisateurs

Le consortium d'investisseurs actuels mesure notre valeur en fonction de notre capacité à maintenir un taux positif d'inscriptions de nouveaux utilisateurs. Cette métrique a chuté rapidement au cours des derniers mois et doit être améliorée en priorité.

L'expansion au sein des marchés locaux et le fait de fournir du géociblage sont vus comme des facteurs critiques pour toucher une gamme plus large d'utilisateurs.

Toute architecture doit être conçue pour se mettre à l'échelle en suivant notre base clients.

### Innover dans le périmètre d'une architecture d'entreprise

La plateforme historique de Foosus a naturellement évolué vers la complexité en raison du changement rapide et d'un manque de vision long terme.

Tout en préservant un sentiment de possession chez chaque ingénieur et partenaire impliqué dans la création de la nouvelle plateforme, nous avons besoin d'un périmètre clair pour 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.



## **Soutenir l'innovation technique rapide et l'expérimentation**

Le marché actuel voit nos concurrents directs prendre rapidement l'avantage en pivotant en réponse à de nouvelles informations apprises.

L'apprentissage doit être au cœur de notre état cible de l'architecture, étant donné que cela a été verrouillé par des solutions par le passé, d'une manière qui a généré davantage d'instabilité et de dette technique.

La plateforme doit être conçue en gardant à l'idée l'extensibilité et la personnalisation des fonctionnalités.

## **Visibilité de la plateforme**

Le comportement technique de la plateforme, non plus que sa performance d'un point de vue du business, n'est pas clair. Toutes les connaissances acquises actuellement nécessitent des analyses de registres et de feuilles de calcul, avant de pouvoir rechercher l'intelligence business.

Nous avons besoin d'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.

## **Améliorer la réputation de Foosus sur le marché grâce à la stabilité**

La marque Foosus doit être renforcée en réduisant les interruptions de service visibles par les utilisateurs. Cela implique, d'une part, des process pour réduire le risque de sortir des solutions qui échouent ou qui soient de mauvaise qualité, et d'autre part, la capacité de sortir de nouvelles versions de notre plateforme sans impacter l'utilisateur par des interruptions de service.

Les sorties à 3 h du matin semblaient fonctionner lorsque nos utilisateurs se trouvaient principalement dans la même zone géographique, mais nous ne devons plus dépendre de cela.



## Hypothèses

De manière à ce que la prise en main du projet soit optimale et que nous saisissons tous les tenants et aboutissants, Foosus nous a transmis la liste des hypothèses qu'il préconise dans le cadre de la réalisation de ce projet. Libre à nous d'en tenir, strictement, compte.

Néanmoins, il n'en reste pas moins intéressant de connaître leurs positions sur le sujet avant toute proposition de manière à avoir un second avis et ainsi pouvoir comparer nos réflexions.

Voici les 5 hypothèses de Foosus :

- Plutôt que d'investir davantage dans la plateforme existante, nous la conserverons en mode de maintenance. Aucune nouvelle fonctionnalité ne sera développée.
- 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.
- Les équipes étant attachées sentimentalement à la plateforme existante, les dirigeants devront éviter de prendre de faux raccourcis en intégrant un nouveau comportement dans le système déjà existant.
- 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 fonctions.
  - Par exemple, les utilisateurs précoces pourront choisir d'utiliser les nouvelles fonctionnalités de recherche intégrées au processus de paiement existant.
- 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.



## Approbations signées

Le présent document a été fait en 6 exemplaires, le ..... à .....

Signature des parties prenantes précédée de la mention « Lu et approuvé ».

**Monsieur Ayrton DE ABREU MIRANDA — Architecte logiciel**

*Signature*

**Monsieur Pete PARKER — Responsable ingénieur**

*Signature*

**Monsieur Jack HARKNER — Responsable infrastructure**

*Signature*

**Madame Natasha JARSON — Directeur informatique**

*Signature*

**Monsieur Daniel ANTHONY — Directeur produit**

*Signature*

**Madame Ash CALLUM — Directeur général**

*Signature*