

# **Spécification des Conditions requises pour l’Architecture**

*Projet :* nouvelle application

*Client :* Foosus

*Préparé par :*

*N° de Version du Document :* 0.1

*Titre :* Spécification des Conditions requises pour l’Architecture

*Date de Version du Document :*

*Revu par :*

*Date de Révision :*

Table des matières

[**Spécification des Conditions requises pour l’Architecture** 1](#_Toc108196378)

[**Objet de ce document** 2](#_Toc108196379)

[**Mesures du succès** 2](#_Toc108196380)

[Indicateurs de réussite 3](#_Toc108196381)

[**Conditions requises pour l’architecture** 3](#_Toc108196382)

[**Contrats de service business** 4](#_Toc108196383)

[**Accords de niveau de service** 4](#_Toc108196384)

[**Contrats de service application** 4](#_Toc108196385)

[**Objectifs de niveau de service** 4](#_Toc108196386)

[**Indicateur de niveau de service** 5](#_Toc108196387)

[**Lignes directrices pour l’implémentation** 5](#_Toc108196388)

[**Spécifications pour l’implémentation** 6](#_Toc108196389)

[**Standards pour l’implémentation** 6](#_Toc108196390)

[**Conditions requises pour l’interopérabilité** 6](#_Toc108196391)

[**Contraintes** 7](#_Toc108196392)

# **Objet de ce document**

*La 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 l’architecture.*

*Une Spécification des Conditions requises pour l’Architecture constitue généralement un composant majeur du contrat d’implémentation, ou du contrat pour une Définition de l’Architecture plus détaillée.*

*Comme mentionné ci-dessus, la Spécification des Conditions requises pour l’Architecture accompagne le Document de Définition de l’Architecture, avec un objectif complémentaire : le Document de Définition de l’Architecture fournit une vision qualitative de la solution et tâche de communiquer l’intention de l’architecte.*

*La Spécification des Conditions requises pour l’Architecture fournit une vision quantitative de la solution, énumérant des critères mesurables qui doivent être remplis durant l’implémentation de l’architecture.*

# **Mesures du succès**

## Indicateurs de réussite

Nous avons défini plusieurs indicateurs permettant de mesurer le succès de la nouvelle architecture. En effet, en atteignant les différents objectifs cités, nous pourrons considérer le projet comme étant un succès total.

|  |  |
| --- | --- |
| Indicateur | Changement souhaité pour l'indicateur |
| Nombre d'adhésions d'utilisateurs par jour | Augmentation de 10 % |
| Adhésion de producteurs alimentaires | Passer de 1,4/mois à 4/mois |
| Délai moyen de parution | Réduit de 3,5 semaines à moins d'une semaine |
| Taux d'incidents de production P1 | Pour commencer : réduit de >25/mois à moins de 1/mois. |

# **Conditions requises pour l’architecture**

Nous avons plusieurs de conditions à respecter afin de pouvoir concevoir une nouvelle architecture, viable, fiable et efficace. En effet la nouvelle architecture se doit de remplacer l’ancienne, donc fournir une qualité de service égale voir supérieure.

La nouvelle architecture se doit aussi de suivre l’une des lignes directrices majeures de notre entreprise Foosus, qui oriente vers le green IT.

Nous avons décidé de créer un tableau pour regrouper les différentes conditions requises liées à la nouvelle architecture.

|  |  |
| --- | --- |
| Conditions | Détails |
| Logique éco-responsable | Mise en place d’une architecture comportant des éléments légers, des fonctionnalités nécessaires et limitant les requêtes inutiles. |
| Logique évolutif | La nouvelle architecture sera capable de s’adapter aux technologies actuelle et futur |
| Politique de Transparence | Mise en place d’une politique de transparence avec les utilisateurs |

# **Contrats de service business**

## **Accords de niveau de service**

Afin d’efficacement définir les différents accords de niveau de service et respecter les normes en vigueur, nous nous devons de définir certains termes :

SLI (Indicateur de niveau de service) : Le SLI est la mesure clé des performances

SLO (Objectif de niveau de service) : Les SLO sont les valeurs cibles ou les objectifs de performances du système. Ils représentent un engagement permanent.

SLA (Accord de niveau de service) : Le SLA définit les conséquences du non-respect des engagements SLI/SLO

|  |  |  |
| --- | --- | --- |
| SLI | SLO | SLA |
| Les clients peuvent accéder à l’application quand ils le souhaitent | Les serveurs d’application doivent être disponibles 99,9% du temps | Un remboursement sera effectué en compensation des pertes induites par la non-disponibilité |
| Nous pouvons accéder à tout moment aux serveurs afin d’effectuer des modifications | L’accès aux différentes interfaces doit être disponible 7 jours sur 7 et 24/24H | Des pénalités peuvent être appliquées vis-à-vis de notre fournisseur |
| Nous avons une panne matérielle ou logiciel | Un technicien se mobilise sur site en moins de 4H et une assistance à distance disponible en moins de 10mins | Des pénalités peuvent être appliquées vis-à-vis de notre fournisseur |

# **Contrats de service application**

## **Objectifs de niveau de service**

Les ambitions de notre entreprise Foosus nous poussent à avoir des objectifs de haut niveau pour nos services proposés.

En effet notre nouvelle application propose une multitude de services. Pour chacun d’entre eux, divers objectifs ont été fixés.

|  |  |
| --- | --- |
| Service | Objectif |
| Géolocalisation | Tirer parti de la géolocalisation pour relier des fournisseurs et des consommateurs et pour proposer des produits disponibles à proximité des lieux de résidence de ces derniers. |
| Adaptabilité géographique | L'architecture devra être évolutive pour permettre à nos services de se déployer sur diverses régions à travers des villes et des pays donnés. |
| Compatibilité | Cette solution doit être utilisable avec des appareils mobiles et fixes. Elle doit tenir compte des contraintes de bande passante. |
| Gestion de type de compte | Elle doit pouvoir prendre en charge différents types d'utilisateurs |

## **Indicateur de niveau de service**

Afin de pour mesurer l’efficacité de nos niveaux de service, nous avons décidé d’utiliser comme indicateur un formulaire client sous forme de QCM.

En effet le QCM est sur 10pts, nous allons distribuer le QCM à la majorité de nos clients. La moyenne de tous les résultats obtenus nous servira d’indicateur de niveau de service.

Voici l’échelle de notation.

|  |  |
| --- | --- |
| Note |  |
| Inférieur à 5 | Les services proposés sont désastreux |
| Compris en 5 et 7 | Le niveau de service est acceptable |
| Supérieur à 7 | Le niveau de service est convenable |

# **Lignes directrices pour l’implémentation**

Notre ligne directrice pour l’implémentation de la nouvelle solution comprend 7 concepts à respecter, afin de pouvoir implémenter parfaitement la nouvelle solution.

* Les objectifs se doivent d’être planifiés
* Toute implémentation se doit d’être planifiée
* Toutes les parties prenantes se doivent d’être sollicitées où être tenues informées
* Les critères d’évaluation et succès se doivent d’être déterminés à l’avance
* Les différents livrables d’utilisation et de conception se doivent d’être clairs
* La gestion de risque et d’opportunité doit être effectuée avant toute implémentation
* Chaque étape se doit de répondre à un critère d’évaluation

# **Spécifications pour l’implémentation**

Nous avons plusieurs spécifications à respecter, cela est dû au langage de programmation, et à l’environnement qui l’accompagne.

Nous avons décidé de regrouper, dans un tableau, les caractéristiques majeures du projet. Afin de plus facilement détailler les différentes spécificités liées à celle-ci par la suite.

|  |  |
| --- | --- |
|  |  |
| Langage | Java |
| Framework | Sprint boot |
| Architecture type | Micro-service |
| Méthode | Agile |

Les machines actuellement disponibles possèdent les prérequis nécessaires pour développer en java, néanmoins l’installation d’un IDE tel que **IntelliJ IDEA** est nécessaire (version communautaire) et la mise en place d’un dépôt **git lab** sera nécessaire.

# **Standards pour l’implémentation**

Nous avons opté pour une architecture micro-service développé en java, nous allons donc devoir respecter les standards de développement et de conception lier à nos choix techniques.

Nous avons mis à disposition un guide de bonnes pratiques pour le développement java sous sprint boot, disponible sur le drive ou encore accessible depuis [ici](https://spring.io/microservices).

# **Conditions requises pour l’interopérabilité**

Afin de faciliter l’interopérabilité, nous avons décidé d’implémenter plusieurs techniques basées au tour de java. Nous avons décidé de les consigner dans un tableau afin de faciliter la lecture.

|  |  |
| --- | --- |
| Technique | Explication |
| Mise en place des paquets pour les différents OS | Pour les OS les plus présents sur le marché des paquets seront disponibles afin de garantir le fonctionnement |
| Création d’une maquette universelle | Nous allons développer l’application de sorte qu’elle soit interprétée par les différents OS de la même façon |
| Création d’une version web responsive | Elle pourra être utilisée sur tout OS qui possède un navigateur web. |

# **Contraintes**

Nous nous retrouvons avec différentes contraintes, les contraintes techniques et celle de production.

Comme contraintes de production, nous retrouvons :

● 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.

Comme contraintes techniques nous avons :

●L’application se doit de tenir compte des contraintes de bande passante

●Elle se doit d’être légère et évolutive

●L’application doit fonctionner sur les différents supports disponibles sur le marché