



# Spécification des Conditions requises pour l'Architecture

---

*Projet : Approvisionnement alimentaire géographiquement responsable*

*Client : Foosus*

*Préparé par : Geoffrey Menon*

*N° de Version du Document : 0.1*

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

*Date de Version du Document : 30 Novembre 2023*

*Revu par :*

*Date de Révision :*

# Table des Matières

---

1. Objet de ce document
2. Mesures du succès
3. Conditions requises pour l'architecture
4. Contrats de service business
5. Contrats de service application
6. Lignes directrices pour l'implémentation
7. Spécifications pour l'implémentation
8. Standards pour l'implémentation
9. Conditions requises pour l'interopérabilité
10. Conditions requises pour le management du service IT
11. Contraintes
12. Hypothèses

## 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 remplir quantitative de la solution, énumérant des critères mesurables qui doivent être remplis durant l'implémentation de l'architecture.*

# Mesures du succès

---

Indicateur	Description	Méthode de Mesure	Cible
<b>Taux d'Adoption</b>	Le pourcentage d'utilisateurs qui utilisent la nouvelle plateforme.	Suivi des connexions et des interactions.	30% d'augmentation dans les 3 premiers mois après le lancement.
<b>Satisfaction Utilisateur</b>	Mesure de la réaction des utilisateurs face à la nouvelle fonctionnalité.	Enquêtes de satisfaction, scores NPS, et revues.	Score Satisfaction supérieur à 80 % ou amélioration de 20% par rapport à la plateforme existante.
<b>Impact sur l'Engagement</b>	Changements dans la fréquence et la durée d'utilisation de l'application.	Analyse des métriques d'engagement avant et après l'introduction de la fonctionnalité.	Augmentation de 10% du temps passé sur l'application ou de la fréquence d'utilisation.
<b>Performance Technique</b>	Temps de réponse et fiabilité de la nouvelle fonctionnalité.	Monitoring de la performance et des temps de réponse via des outils d'analyse technique.	Temps de réponse inférieur à 2 secondes, disponibilité de 99.95%.
<b>Réduction de la dette technique</b>	Diminution mesurable de la dette technique existante avec la mise en œuvre de la nouvelle architecture.	Définir les éléments dépréciés et faire en sorte qu'il soit remplacé par un équivalent maintenu.	Suppression de toutes les bibliothèques ou technologies qui ne sont plus maintenues ou dépassées.
<b>Capacité à s'ajuster</b>	Capacité de la nouvelle architecture à résister à la charge	Test de charges et évaluation de la fiabilité de l'application avec 10 000 utilisateurs connectés.	Ajustement au besoin et optimisation des coûts d'infrastructure.

---

# Conditions requises pour l'architecture

Exigence	Description
Scalabilité	L'architecture doit soutenir une augmentation de 100% de la charge utilisateur sans dégradation significative des performances.
Mobilité	Capacité à fonctionner sur divers appareils tels que mobiles, ordinateurs, tablettes et autres périphériques.
Maintenabilité	Capacité du système à être maintenable dans le temps grâce à une structure de projet efficace et l'utilisation de standard de développement tel que SOLID.
Accessibilité	Plateforme utilisable par des lecteurs d'écran et autres systèmes rendant le site accessibles à des personnes en situation de handicap
Sécurité	L'architecture doit se conformer aux dernières normes de sécurité, incluant le chiffrement des données et la protection contre les intrusions.
Interopérabilité	Capacité de l'architecture à intégrer des services tiers et des API sans besoin de changement majeurs.
Évolutivité	Système permettant des mises à jour sans interruptions du service pour l'ajout fréquent de nouvelles fonctionnalités.

# Contrats de service business

## Accords de niveau de service

Service	Exigence
<b>Disponibilité de la plateforme Foosus</b>	Disponibilité garantie mensuelle. Maintenance programmée hors des heures de forte affluence, avec préavis de 2 jours.
<b>Performance de l'application Foosus</b>	Temps de réponse moyen inférieur à 2 seconde pour les recherches géociblées. Support de pic d'usage de 20% au-dessus de la moyenne mensuelle sans dégradation.
<b>Support technique Foosus</b>	Réponse en 15 minutes pour les incidents critiques affectant l'accès utilisateur. Résolution garantie dans les 2 heures pour les problèmes affectant plus de 5% de la base d'utilisateurs.
<b>Sécurité et conformité des données</b>	Audit de sécurité biannuel et conformité à la RGPD pour la protection des données utilisateur. Notifications immédiates aux utilisateurs en cas de violation de données.
<b>Stratégie de sauvegarde et de reprise après sinistre</b>	Sauvegardes quotidiennes des données utilisateur avec un RPO de 24 heures. RTO de 4 heures pour les services critiques en cas de panne majeure.
<b>Rapports d'analyse de données</b>	Rapports de performance des fonctionnalités géociblées fournis chaque trimestre. Analyses de tendances des utilisateurs disponibles pour les équipes de développement et de marketing.
<b>Cycle de mise à jour de l'application</b>	Mises à jour de fonctionnalités tous les 2 mois avec zéro temps d'arrêt. Notifications des mises à jour aux utilisateurs 14 jours à l'avance.
<b>Formation et support des utilisateurs</b>	Ressources d'auto-assistance disponibles via l'application avec mise à jour pour chaque nouvelle version. Webinaires de formation pour les grandes mises à jour fonctionnelles.
<b>Intégration avec les services tiers</b>	Intégrations avec les services de livraison tiers maintenues avec 99% de disponibilité. Documentation API pour partenaires mise à jour avec chaque nouvelle version.
<b>Gestion des changements</b>	Évaluations des risques pour chaque modification majeure du système avec des plans d'atténuation.

# Contrats de service application

## Objectifs de niveau de service

Service (SLI)	Description de l'Indicateur
Taux de réussite des requêtes de localisation	Pourcentage de requêtes de localisation qui se complètent sans erreurs.
Latence des requêtes de recherche	Temps moyen nécessaire pour obtenir une réponse à une requête de recherche.
Disponibilité des services de paiement	Pourcentage du temps où le service de paiement est opérationnel et accessible.
Succès des mises à jour de l'application	Pourcentage de mises à jour de l'application réalisées avec succès sans incident.
Compatibilité des versions de l'application	Pourcentage de dispositifs sur lesquels l'application fonctionne correctement.
Réponse aux vulnérabilités de sécurité	Délai moyen de réponse aux vulnérabilités de sécurité identifiées.
Résolution des incidents non critiques	Temps moyen pris pour résoudre les incidents non critiques.
Fiabilité de l'application	Temps de fonctionnement de l'application sans erreurs significatives.
Traitement des demandes de données utilisateur	Temps moyen pour répondre aux demandes des utilisateurs concernant leurs données.
Capacité d'évolutivité de l'application	Pourcentage de l'augmentation de charge que l'application peut gérer sans perte de performance.

# Indicateurs de niveau de service

---

Service (SLO)	Objectif Spécifique
Taux de réussite des requêtes de localisation	99.5% des requêtes doivent être réussies mensuellement.
Latence des requêtes de recherche	95% des requêtes doivent répondre en moins de 1 seconde.
Disponibilité des services de paiement	Le service doit être disponible 99.9% du temps chaque mois.
Succès des mises à jour de l'application	99% des mises à jour doivent être effectuées sans interruption de service.
Compatibilité des versions de l'application	La compatibilité doit être assurée pour 99% des appareils éligibles.
Réponse aux vulnérabilités de sécurité	95% des vulnérabilités doivent être adressées dans les 24 heures suivant leur découverte.
Résolution des incidents non critiques	90% des incidents doivent être résolus dans un délai de 48 heures.
Fiabilité de l'application	Un taux de fiabilité de 99.5% doit être maintenu mensuellement.
Traitement des demandes de données utilisateur	95% des demandes doivent être traitées dans un délai de 7 jours ouvrables.
Capacité d'évolutivité de l'application	L'application doit maintenir sa performance lors d'une augmentation de trafic jusqu'à 100% sans dégradation.

---

# Lignes directrices pour l'implémentation

Directive	Description
Gestion de la dette technique	Prioriser la refonte et la refactoration des composants avec la plus haute dette technique pour améliorer la maintenabilité et la performance.
Approche modulaire	Utiliser une architecture de microservices pour faciliter l'évolutivité et l'indépendance des fonctionnalités de l'application.
Automatisation des tests	Mettre en place un cadre d'automatisation des tests pour couvrir les tests unitaires, d'intégration, de charge, et de sécurité.
Gestion des configurations	Utiliser des outils de gestion de configuration pour tracer et gérer les changements dans l'environnement de développement et de production.
Principes SOLID pour la conception logicielle	Appliquer les principes SOLID pour créer des systèmes logiciels bien conçus, flexibles et maintenables.
Sécurité dès la conception	Intégrer les considérations de sécurité dès les premières étapes du développement pour prévenir les vulnérabilités et protéger les données utilisateur.
Responsive et adaptatif	Concevoir des applications qui offrent une expérience utilisateur cohérente sur divers appareils et tailles d'écran.
Surveillance et journalisation	Mettre en œuvre une surveillance robuste et des capacités de journalisation pour détecter et répondre rapidement aux problèmes opérationnels.
Documentation et commentaires	Maintenir une documentation complète et à jour pour chaque module et fonctionnalité, y compris les commentaires dans le code.
Approche axée sur l'utilisateur	Impliquer les feedbacks des utilisateurs dans le cycle de développement pour garantir que l'application répond aux besoins réels des clients.
Gestion des dépendances	Utiliser des outils de gestion des dépendances pour assurer la compatibilité et la mise à jour des bibliothèques et frameworks externes.



Directive	Description
<b>Planification de la scalabilité</b>	Évaluer et planifier la capacité des systèmes pour s'assurer qu'ils peuvent gérer la croissance des utilisateurs et des données.

## Spécifications pour l'implémentation

Spécification	Description
<b>Architecture basée sur les microservices</b>	Développer des services indépendants qui répondent à des fonctions commerciales spécifiques, permettant une évolutivité et une maintenance facilitées.
<b>Système de gestion de base de données</b>	Utiliser un système de gestion de base de données relationnelle ou NoSQL qui supporte les transactions géociblées et l'analyse de grandes quantités de données.
<b>Infrastructure Cloud</b>	Mettre en œuvre une infrastructure en tant que service (IaaS) ou plateforme en tant que service (PaaS) pour une mise à l'échelle dynamique et une gestion des ressources optimisée.
<b>Frontend réactif et mobile-first</b>	Concevoir une interface utilisateur réactive qui s'adapte à différents appareils, en priorisant les performances et l'expérience utilisateur sur mobile.
<b>API RESTful ou GraphQL</b>	Définir des interfaces de programmation d'applications (API) pour faciliter la communication entre les microservices et les clients externes de l'application.
<b>Authentification et autorisation</b>	Implémenter des mécanismes d'authentification robustes, tels que OAuth 2.0 ou OpenID Connect, pour sécuriser l'accès utilisateur.
<b>Protocoles de chiffrement</b>	Chiffrer les données en transit et au repos avec des protocoles tels que TLS et AES pour protéger la confidentialité.
<b>Cadre de travail frontend et backend</b>	Adopter des frameworks modernes et bien soutenus pour le développement frontend (ex: React, Vue.js) et backend (ex: Node.js, Spring Boot, Django).

Spécification	Description
<b>Containerisation et orchestration</b>	Utiliser des conteneurs Docker pour emballer et déployer des applications, et Kubernetes pour l'orchestration et la gestion des conteneurs.
<b>Intégration continue / Déploiement continu (CI/CD)</b>	Mettre en place un pipeline CI/CD pour automatiser les tests et le déploiement, réduisant les erreurs manuelles et accélérant la mise sur le marché.
<b>Plan de récupération d'urgence</b>	Établir un plan de récupération après sinistre pour garantir la continuité des opérations en cas de défaillance majeure du système.
<b>Conformité et normes</b>	Assurer la conformité avec les normes en vigueur tel que la RGPD et ISO/IEC 27001 pour la sécurité de l'information.
<b>Déploiement et maintenance</b>	Créer des processus standardisés pour le déploiement, la mise à jour, et la maintenance des applications pour minimiser les interruptions de service.

## Standards pour l'implémentation

Standard	Description
<b>Codage propre et maintenable</b>	Adhérer aux principes de codage propre, comme une nomenclature cohérente, des fonctions claires et des modules bien organisés, pour faciliter la maintenance et l'évolution du code.
<b>Conventions de codage</b>	Suivre les conventions de codage spécifiques au langage utilisé (par exemple, PEP 8 pour Python, Java Coding Standards pour Java) pour garantir la cohérence à travers l'équipe de développement.
<b>Utilisation de contrôle de version</b>	Utiliser Git pour le contrôle de version avec des branches dédiées pour les fonctionnalités, les corrections et les releases pour gérer efficacement le code source.
<b>Normes API</b>	Suivre les normes REST ou GraphQL pour les API, y compris des conventions pour les endpoints, les réponses HTTP et la gestion des erreurs.

Standard	Description
<b>Sécurité des données</b>	Respecter les normes de sécurité des données comme OWASP pour les applications web, en s'assurant que toutes les mesures de sécurité sont en place pour protéger contre les menaces courantes.
<b>Tests unitaires et d'intégration</b>	Écrire des tests unitaires pour chaque composant et des tests d'intégration pour les interactions entre composants, visant une couverture de test substantielle du code.
<b>Responsive Web Design (RWD)</b>	Suivre les principes de RWD pour que les applications web soient accessibles et fonctionnelles sur tous les appareils et tailles d'écran.
<b>Accessibilité</b>	S'assurer que l'interface utilisateur est conforme aux standards d'accessibilité WCAG pour que les applications soient utilisables par tous, y compris les personnes handicapées.
<b>Documentation du code et des API</b>	Maintenir une documentation à jour du code et des API, utilisant des outils comme Swagger ou JSDoc pour générer automatiquement la documentation.
<b>Surveillance des applications</b>	Utiliser des outils de surveillance tels que Prometheus ou New Relic pour surveiller la santé et les performances des applications en production.
<b>Gestion des dépendances</b>	Utiliser des outils comme Maven ou npm pour gérer les dépendances du projet et s'assurer qu'elles sont à jour et sécurisées.

## Conditions requises pour l'interopérabilité

---

<b>Intéropérabilité</b>	<b>Description</b>
<b>Compatibilité API</b>	Assurer que les API internes et externes sont conformes aux standards RESTful ou GraphQL, facilitant l'intégration avec d'autres systèmes et services.
<b>Formats de données standardisés</b>	Utiliser des formats de données universels tels que JSON ou XML pour l'échange de données, garantissant la compatibilité entre différents systèmes.
<b>Protocoles de communication sécurisés</b>	Mettre en œuvre des protocoles de communication sécurisés comme HTTPS, TLS pour sécuriser les échanges de données entre les services.
<b>Authentification et autorisation unifiées</b>	Adopter des mécanismes d'authentification et d'autorisation standard comme OAuth2 ou OpenID Connect pour une gestion cohérente des accès.
<b>Interopérabilité des bases de données</b>	Assurer que les systèmes de gestion de bases de données (relationnelles et NoSQL) supportent des standards ouverts pour l'interopérabilité avec d'autres outils et systèmes.
<b>Adaptabilité des services de paiement</b>	Intégrer des services de paiement flexibles capables de s'interfacer avec diverses plateformes de paiement et banques.
<b>Conformité avec les standards du cloud</b>	Respecter les standards et les API des principaux fournisseurs de cloud pour garantir la compatibilité et la portabilité des solutions cloud.
<b>Support multiplateforme</b>	Garantir que l'application est fonctionnelle sur diverses plateformes (iOS, Android, Web) avec une expérience utilisateur cohérente.
<b>Échange de données en temps réel</b>	Mettre en place des capacités d'échange de données en temps réel, comme les WebSockets, pour les fonctionnalités interactives comme le suivi géolocalisé.
<b>Internationalisation et localisation</b>	Prévoir une prise en charge de multiples langues et formats régionaux pour assurer l'accessibilité et la pertinence dans différents marchés.

# Conditions requises pour le management du service IT

---

Condition	Description
Sécurité et gestion des risques	Adopter une stratégie proactive de sécurité, incluant la cybersécurité, la protection des données, et la conformité.
Gestion de la capacité	Planifier et optimiser les ressources pour soutenir la croissance et l'évolutivité des services de Foosus.
Gestion des changements	Implanter un processus structuré pour la gestion des changements, minimisant les risques et perturbations.
Formation et développement des compétences	Prioriser la formation continue et le développement des compétences, en se concentrant sur les technologies et pratiques émergentes.
Continuité des opérations	Développer et maintenir un plan de continuité opérationnelle robuste pour assurer la résilience et la récupération rapide.
Surveillance et gestion des incidents	Mettre en place des systèmes de surveillance pour une détection et réponse rapide aux incidents.
Évaluation de la qualité des services	Effectuer des évaluations régulières pour améliorer la qualité des services et assurer la satisfaction des utilisateurs.

# Contraintes

---

Contrainte	Description
<b>Gestion de la dette technique</b>	Foosus doit faire face à une dette technique importante due à des choix d'architecture antérieurs, nécessitant une attention particulière pour la réduction de cette dette avec le développement de la nouvelle plateforme.
<b>Limites de budget</b>	Des restrictions budgétaires qui peuvent limiter les choix en termes d'outils, de technologies, ou de recrutement de nouvelles compétences.
<b>Compatibilité avec les systèmes existants</b>	Nécessité de maintenir l'interopérabilité avec les systèmes et les technologies existants, ce qui peut limiter les options pour les nouvelles implémentations.
<b>Contraintes de performances</b>	La nécessité de fournir des services géociblés de haute performance impose des exigences élevées en termes de temps de réponse et de traitement des données.
<b>Sécurité des données</b>	Obligation de respecter des normes de sécurité strictes, notamment pour la protection des données utilisateur.
<b>Contraintes de scalabilité</b>	La nécessité d'une architecture qui puisse évoluer avec la croissance rapide de l'entreprise, tout en gérant efficacement l'augmentation du trafic et des données.
<b>Dépendances technologiques</b>	Dépendances envers certaines technologies, plateformes ou frameworks, qui peuvent limiter la flexibilité dans le choix de nouvelles solutions.
<b>Exigences réglementaires</b>	Nécessité de se conformer à diverses réglementations tel que la RGPD ou autres normes qui peuvent influencer les choix de conception et d'implémentation.

# Hypothèses

ID	Hypothèse	Impact	Concerne
1	<b>Foosus dispose du savoir-faire et des compétences nécessaires à la réalisation du projet</b>	Positif : Permet à l'entreprise de mener à bien le projet et d'éviter d'avoir à embaucher des intervenants externes.	Équipe Technique
2	<b>Les parties prenantes internes et externes s'impliqueront pour permettre l'ajustement de la nouvelle solution.</b>	Positif : Assure une cohérence et une qualité, permet d'éviter les erreurs en s'adaptant au mieux aux besoins métiers.	Parties Prenantes
3	<b>Le budget nécessaire pour la réalisation du projet est correctement estimé</b>	Positif : Permet de maintenir les coûts sous contrôle et d'éviter les dépassements de budget qui pourrait mettre en péril la réalisation du projet.	PDG (Ash Callum)
4	<b>Les fournisseurs locaux et les clients sont enclin à s'intégrer à la plateforme.</b>	Positif : Crucial pour le succès de la plateforme et la satisfaction des utilisateurs finaux.	Directeur des Produits (Daniel Anthony)