



Spécification des Conditions requises pour l'Architecture

—
Projet : FoosusGeo2023

Client : Foosus

Préparé par : A.D

—
N° de Version du Document : VF

Titre : Spécification des Conditions requises pour l'Architecture

Date de Version du Document : 23 octobre 22

Revu par : DSI

Date de Révision : 24 oct. 22

—

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

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.

Mesures du succès

Les objectifs seront validés si les indicateurs ci-dessous sont atteints.

Indicateurs de réussite

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

L'architecture SI aura la charge d'analyser, définir et cadrer l'évolution des systèmes d'information en fonction de la stratégie d'entreprise, des processus métier et des innovations technologiques.

L'architecture à apportera une vue consolidée du SI et à proposera un plan cohérent pour aligner infrastructure, applications, modélisation fonctionnelle, besoins métiers et stratégie

d'entreprise, tout en prenant en compte l'existant !

La stratégie métier qui guidera les orientations SI. Les choix métiers ne devront ainsi être guidés par les contraintes techniques !

L'architecture apportera une vue consolidée du SI et à proposera un plan cohérent pour aligner infrastructure, applications, modélisation fonctionnelle, besoins métiers et stratégie d'entreprise, tout en prenant en compte l'existant

En prenant en compte les orientations stratégiques, l'architecture définira en conséquence la feuille de route et les différentes étapes pour passer du Système d'Information actuel au Système d'Information cible sur chacun des axes.

Une cartographie pour chacune des couches devra être présentée :

1. Cartographie des processus pour l'architecture métier
2. Cartographie fonctionnelle pour l'architecture fonctionnelle
3. Cartographie applicative pour l'architecture applicative
4. Cartographie technique pour l'architecture technique

La méthode d'architecture permettra la mise en œuvre fonctionnelle IT de la SI existant vers la solution cible afin de le faire évoluer le système d'information cible

La nouvelle architecture devra permettre :

1. Désiloter les applicatifs afin de faire communiquer les applications entre elles
2. Superviser les flux et les échanges afin de gagner en réactivité face aux erreurs et en capacité d'analyse
3. Faciliter le remplacement et le décommissionnement d'applications
4. Contrôler les impacts lors de la modification des interfaces ou des applications et éviter le cercle vicieux qui consiste à ajouter des couches car on ne maîtrise pas l'existant
5. Ouvrir votre Système d'Information vers vos partenaires, clients, fournisseurs

L'architecture doit permettre d'obtenir le meilleur rapport qualité-coût.

L'architecture pourra inclure de nouveaux composants personnalisés ou des composants du commerce pour favoriser la flexibilité, la stabilité et l'extensibilité.

En terme opérationnel les principaux objectifs de l'entreprise en matière d'architecture sont les suivants.

1. 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.
2. L'architecture devra être évolutive pour permettre à nos services de se déployer sur diverses régions à travers des villes et des pays donnés.
3. Notre solution doit être disponible pour nos fournisseurs et nos consommateurs, où qu'ils se trouvent. Cette solution doit être utilisable avec des appareils mobiles et

- fixes. Elle doit tenir compte des contraintes de bande passante pour les réseaux cellulaires et les connexions Internet haut débit.
4. Elle doit pouvoir prendre en charge différents types d'utilisateurs (par exemple, fournisseurs, back-office, consommateurs) avec des fonctionnalités et des services spécifiques pour ces catégories.

Contrats de service business

Accords de niveau de service

L'architecture doit être toujours disponible (vue client)

L'architecture doit être monitoré

Une communication doit être envoyée en cas de problème

Contrats de service application

Objectifs de niveau de service

Les principaux objectifs seront :

Une continuité de service de 99%

Une disponibilité de service :

1. Planification d'un cycle de maintenance tous les mois
2. Reprise de service sur incident (délai)
3. Test de DR

Indicateurs de niveau de service

Criticité	Délai de prise en compte	Délai de résolution
1	15 minutes	30 minutes
2	1 heures	4 heures
3	4 heures	8 heures
4	1 jour	3 jours

Lignes directrices pour l'implémentation

Les lignes directrices suivantes seront nécessaires afin d'assurer un service de qualité à l'ensemble de l'architecture :

Un **passage à l'échelle** afin d'assurer un redimensionnement et une mise à l'échelle du système selon les besoins des utilisateurs.

Il permettra une bonne disponibilité et de latence à l'architecture web en micro-services

La **maintenabilité** afin de permettre au système la capacité être simplement et rapidement réparé et ainsi à diminuer les temps et les coûts d'intervention.

La **sécurité** permettra une sécurisation des données et une protection et la disponibilité du service

Spécifications pour l'implémentation

Concernant le **passage à l'échelle** l'utilisation d'une orchestration des conteneurs permettra d'automatiser le déploiement, la gestion, la mise à l'échelle et la mise en réseau des conteneurs. Cette technologie est compatible avec tous les environnements qui exécutent des conteneurs.

Elle permettra de déployer la même application dans différents environnements sans modifier sa conception.

- Les tâches suivantes seront prises en compte :
- Provisionnement et déploiement
- Configuration et planification
- Allocation des ressources
- Disponibilité des conteneurs
- Mise à l'échelle ou suppression de conteneurs en fonction des charges de travail dans l'infrastructure
- Équilibrage de la charge et routage du trafic
- Surveillance de l'intégrité des conteneurs
- Configuration des applications en fonction du conteneur sur lequel elles vont s'exécuter
- Sécurisation des interactions entre les conteneurs

L'application sera hébergée sur un **cloud** qui permettra de garder une haute disponibilité de l'infrastructure et une gestion des coûts maîtrisés avec une facturation sur ce que Foosus consommera.

Afin de garder un niveau de maintenance optimal 3 niveaux de maintenance

La maintenance informatique préventive :

- Audit des performances
- Anti-virus
- Audit sécurité
- Mises à jour du système d'exploitation et des logiciels
- Sauvegarde de données

La maintenance informatique corrective.

- Corrective (Panne)
- Planifié (résolution ultérieure)

La maintenance informatique évolutive

- Maintenance qui visera à faire évoluer ou à adapter une ou plusieurs applications, afin d'intégrer de nouvelles fonctions, d'en améliorer le fonctionnement ou de prendre en compte de nouvelles dispositions.
-

Un outil de monitoring sera mis en place afin de remonter les alertes (panne, information etc...)

Les flux de communication seront chiffrés et une Gateway sera mise en place enfin d'assurer les liens entre les différents systèmes.

Un WAF est placé avant l'application Web.

Un anti-virus scanner 3 fois par jour les données qui seront, elle, sauvegardés chiffrés

Standards pour l'implémentation

La nouvelle architecture sera hébergée sur le **cloud AZURE** et utilisera le service Containers-as-a-Service et **Microsoft Azure SQL Database** pour la base de données.

L'utilisation de **Kubernetes** (open source) fournira une « plate-forme permettant d'automatiser le déploiement, la montée en charge et la mise en œuvre de conteneurs d'application sur les clusters de serveurs »

La performance des infrastructures et applicatives sera gérer par le logiciel **Dynatras** ainsi que le monitoring.

Jira Service Management assurera la gestion des tickets d'incident.

La communication des flux sera chiffrée en TLS (minimum 1.2) et l'application sera protégé avec le **WAF** Azure et Azure DDoS Protection Standard pour les dénies de service.

L'Antivirus Microsoft Defender détectera et éliminera et les virus.

La Gateway sera développée en interne avec **java**.

Les données stockées seront chiffrées avec algorithme AES **256 bits** disponible sur le **cloud Azure**.

Conditions requises pour l'interopérabilité

Le protocole de communication utilisé entre les applications sera HTTPS
Développement WEB sera via HTML ou CSS.

Pour l'application web PHP ou JavaScript seront privilégiés.
Pour l'application mobile le langage Java sera privilégié

Conditions requises pour le management du service IT

Le dossier d'architecture doit être validé.

Les règles de sécurité doivent être acceptées et respectées

Les normes et consignes doivent être diffusées et consultables

Les équipes doivent être formées

Contraintes

1. 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.
2. L'architecture doit permettre d'obtenir le meilleur rapport qualité-coût.
3. L'architecture peut inclure de nouveaux composants personnalisés ou des composants du commerce pour favoriser la flexibilité, la stabilité et l'extensibilité.

Hypothèses

1. La plateforme existante, sera conservée en mode de maintenance.
2. Aucune nouvelle fonctionnalité ne sera développée.
3. 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.
4. Les équipes étant attachées à la plateforme existante, les dirigeants devront éviter de prendre de faux raccourcis en intégrant un nouveau comportement dans le système existant.
5. 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.
6. Cette augmentation sera proportionnelle à l'évolution des fonctionnalités.
 - a. Par exemple, les utilisateurs précoces pourront choisir d'utiliser les nouvelles fonctionnalités de recherche intégrées au processus de paiement existant.
7. 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.