

Spécifications des conditions requises pour l'architecture

Projet : Conception d'une nouvelle architecture

Version du document	Date	Diffusion
v0.1	08/08/2021	Structure et plan
v0.2	08/08/2021	KPI
v0.3	15/08/2021	Contrats de service application
V0.4	22/08/2021	Lignes directrices implémentation
v0.5	29/08/2021	Conditions requises
v0.6	01/09/2021	Mise en forme
v0.7	08/09/2021	Finalisation du document

Table des matières

Table des matières	1
Objet de ce document	3
Mesures du succès	3
Métriques et KPIs de l'État Cible de l'Architecture	3
Métriques de livraison de l'architecture et du business	3
Métriques et KPIs	4
Conditions requises pour l'architecture	5
Contrats de service business	5
Accords de niveau de service	5
Tableau SLA	5
Contrats de service application	6
Description des erreurs services	6
Objectif de Niveau de Service	7
Indicateur de niveau de service	7
Lignes directrices pour l'implémentation	7
Standards pour l'implémentation	8
Standards d'écriture de code	8
Plate-forme de développement continu	9
Conditions requises pour l'interopérabilité	9
Normes de nommage et documentation	9
Normes techniques	10
Normes syntaxiques	10
Conditions requises pour le management du service IT	11
Gestion des services informatiques	11
Stratégie de service	11
Design de service	12
Transition de service	12
Fonctionnement du service	12
Amélioration continue du service	13
Inspection des services	13
Evaluation des processus	13
Définition des initiatives de ACS	13
Surveillance des initiatives ACS	13
Contraintes	13
Hypothèses	13

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

Voici les différentes mesures de succès requises pour la réussite de ce projet.

Métriques et KPIs de l'État Cible de l'Architecture

La définition des métriques est effectuée en utilisant le brief business haut niveau ainsi que la vision d'architecture. Ces documents permettent de cibler les aspects techniques et haut niveau à évaluer et améliorer.

Les métriques suivantes seront utilisées pour déterminer le succès de ce travail d'architecture :

Métriques de livraison de l'architecture et du business

Métrique	Valeur cible	Justification
Rapidité d'implémentation de la géolocalisation	Un mois	La géolocalisation est une feature phare de l'architecture cible et doit être disponible rapidement.
Rapidité de déploiement	Moins d'une demi-heure	La complexité du déploiement freine largement l'évolutivité du système.

Rapidité de modification du hardware	Moins d'une demi-heure	Il doit être facile d'ajouter ou de supprimer de l'architecture des serveurs, bases de données..
Accessibilité et résilience	Sur une semaine : accessible à 100% Rapidité de remise en service après perte d'un serveur : moins d'une heure	Le site doit être accessible tous les jours à toutes heures malgré la perte d'un élément de l'architecture
Réduction du nombre de bugs en environnement de production	Réduction du nombre de bugs en production par 100%	L'évolutivité du système requiert une stabilité importante face aux bugs.
Réduction des dépendances entre les services	Services touchés par la modification d'un service : 1 maximum (le service modifié)	L'évolutivité du système doit être améliorée par l'absence de dépendance entre les services

Métriques et KPIs applicatives

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

Métrique	Valeur cible	Justification	Notes supplémentaires
Nombre d'adhésions d'utilisateurs par jour	Augmentation de 10 % (du % d'adhésion par jour)	L'amélioration de la plate-forme doit se traduire par une augmentation des utilisateurs.	Il faudra également travailler la visibilité du produit
Adhésion de producteurs alimentaires	De 1,4 à 4 par mois	La géolocalisation doit assurer aux producteurs d'être facilement trouvés par les clients.	Il faudra planifier des fonctionnalités user-friendly rapidement

Délai moyen de parution	Réduit de 3,5 semaines à moins d'une semaine	Amélioration des workflow de déploiement	Ajout de différents environnements dans le cycle de développement continu
Taux d'incidents de production P1	Pour commencer : réduit de >25/mois à moins de 1/mois.	La plate-forme doit être accessible à tout moment pour les utilisateurs.	Ajout d'un environnement de tests et de recette

Conditions requises pour l'architecture

Contrats de service business

Accords de niveau de service

Un Accord de Niveau de Service (SLA) est un accord entre un client et un fournisseur de service sur des critères de performance. Le but est de s'accorder sur plusieurs critères de performance (résilience, disponibilité, responsabilité..) qui permettront ensuite d'évaluer le niveau de fonctionnement du service.

Tableau SLA

SLO/Microservice	Latence de réponse	Disponibilité	Charge
Rôle	99%	99,99%	99,99%
Utilisateur	99%	99,99%	99,99%
Fournisseur	99%	99,99%	99,99%
Commande	85%	99%	99%
Stock	90%	99%	99%
Vitrine	90%	95%	99%

Contrats de service application

Description des erreurs services

Niveau de sévérité	Description	Réponse attendue
1. Panne	SaaS serveur hors service	Immédiate
2. Critique	Risque élevé de panne de serveur	Dans les 10 minutes
3. Urgent	Impact sur les utilisateurs	Dans les 20 minutes
4. Important	Impact potentiel sur les performances	Dans les 30 minutes
5. Préoccupant	Problème traité mais potentiellement problématique à l'avenir	Dans les un jour ouvré
6. Informatif	Inquiry for information	Dans les 48h

Objectif de Niveau de Service

Un Objectif de Niveau de Service (SLO) spécifie le niveau de performances attendu pour un Indicateur de Niveau de Service (SLI) pendant une période de conformité.

SLO/Microservice	Latence de réponse	Disponibilité	Charge
------------------	--------------------	---------------	--------

Rôle	99%	99,99%	99,99%
Utilisateur	99%	99,99%	99,99%
Fournisseur	99%	99,99%	99,99%
Commande	85%	99%	99%
Stock	90%	99%	99%
Vitrine	90%	95%	99%

Indicateur de niveau de service

La latence de réponse sera mesurée selon le nombre de requêtes dépassant un temps de réponse de 300ms sur 30 jours.

La disponibilité sera mesurée sur le nombre de requêtes indiquant une réussite sur 30 jours.

La charge sera définie par le nombre de tâches par lot planifié dans la minute qui suit son heure de début cible.

Lignes directrices pour l'implémentation

Les lignes directrices suivantes devront être suivies afin de garantir une implémentation efficace du projet.

Lignes directrices haut niveau

1. Avant d'être délivré, le code sera systématiquement validé sur plusieurs environnements différents notamment sur un environnement de build, de test et de recette.
2. Les KPI et les accords de niveau de service seront évalués au moins toutes les semaines.

3. La stabilité du système sera garantie par l'utilisation du "fail-over", c'est-à-dire le remplacement dynamique des composants architecturaux en cas de panne. Cette technique sera également utilisée pour garantir le fonctionnement du service lors des mises à jour système.
4. Le système doit permettre un déploiement rapide et économiquement viable dans de nouvelles zones géographiques (nouveaux pays, nouvelles villes..)
5. L'identité d'un utilisateur sera garantie par un "token" unique dont la validité expirera au bout d'un laps de temps de moins d'une heure. Après cela, la vérification de l'identité de l'utilisateur sera de nouveau requise.
6. Chaque serveur et chaque base de données ne sera utilisée que par un et un seul micro service pour garantir une absence totale de dépendance.
7. Les microservices seront systématiquement décrits selon la norme OpenAPI et la documentation sera disponible à toutes les parties prenantes susceptibles d'utiliser les APIs.
8. Le temps moyen de commande sur un appareil mobile doit être équivalent à celui sur appareil fixe.

Lignes directrices bas niveau

1. La nouvelle interface de recherche de fournisseurs alimentaire devra faire apparaître la position relative du fournisseur par rapport au consommateur;
2. La proximité entre le fournisseur et le consommateur deviendra un critère majeur lors de la recherche
3. Les produits présentés seront détaillés lors du visionnage d'un certain nombre de statistiques avancées les concernant;

Standards pour l'implémentation

Standards d'écriture de code

Les différents standards d'écriture de code à mettre en place sont :

- Une convention de nommage pour les variables, classes, packages;
- Une convention d'organisation des fichiers et dossiers du projet;
- L'indentation et la présentation du code;
- Les commentaires et la documentation;
- Le test des API.

Il sera important d'avoir des standards communs établis pour tout le projet afin de réduire sa dette technique et garantir son évolutivité.

Plate-forme de développement continu

La mise en place d'une plate-forme de développement continu permet d'adopter un cycle de travail et de mise en production standardisé.

Les différentes étapes du développement continu classique sont décrites dans le diagramme ci-dessous :

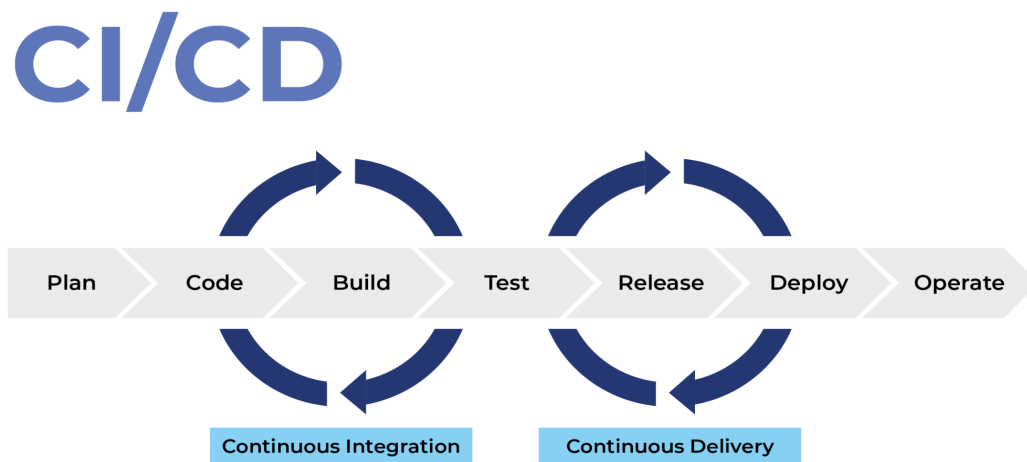


Diagramme d'intégration continue : <https://edge.siriuscom.com>

Ainsi, l'objectif est de suivre ce cycle en utilisant divers environnements de développement continu afin de limiter la mise en production d'erreurs et garantir une efficacité élevée de qualité de code.

Conditions requises pour l'interopérabilité

Normes de nommage et documentation

Les normes de nommage des variables, fonctions, classes, packages et dossiers ainsi qu'une documentation standard permettent de comprendre et d'exploiter plus facilement notre application.

Nous conseillons la norme de documentation OpenAPI pour les différents microservices du système. Cette norme repose notamment sur :

- La description des endpoints (/utilisateur, /fournisseur..) et des méthodes (get, post..);
- La descriptions des input et output des méthodes;
- Les manières de s'identifier;
- Les informations de contact, termes d'usage et autres informations.

La documentation en ligne devra être accessible en plusieurs langues et en anglais pour les utilisateurs non voyants. Elle devra être mise à jour et facilement parcourable.

Normes techniques

Catégorie	Sous-catégorie	Standards
Réseau		IPv6, IPSec
Transport		TCP, UDP, NTP, RTP, SRTP, RTCP, TLS
Session		SSH
Application	Transfert	HTTP, HTTPS, SFTP, CORS, AS2
Application	Accès	LDAP, LDAPS
Application	Authentification	OAuth2
Service	Géospatial	WMS, WFS, WCS, TJS, WMTS, CSW, WPS

Normes syntaxiques

Catégorie	Sous-catégorie	Standards
Encodage	Caractère	UTF-8
Encodage	Compression	Zip, TAR, 7z
Web		HTML, CSS, JS, CMIS
Données		XML, JSON, DOM,
Document		PDF, ODF
Multimédia	Conteneur Vidéo	MKV, MP4
Multimédia	Conteneur Audio	MP3, AAC, FLAC
Multimédia	Image	PNG, JPEG, SVG, GeoTIFF
Signature		CAdES, PAdES, XAdES, ASIC

Conditions requises pour le management du service IT

Gestion des services informatiques

La gestion des services informatiques propose d'étudier les services et les processus de l'entreprise en les identifiant, les mesurant et en essayant de les améliorer dans un cycle régulier. Le but étant de gérer les services de manière optimale afin qu'ils créent une valeur, un atout stratégique.

Pour cela, de nombreux framework ont été conçus afin de standardiser cette gestion de services et nous détaillerons dans la suite de cette partie le framework ITIL (Information Technology Infrastructure Library).

Stratégie de service

La stratégie de service forme la base de l'organisation des framework de gestion des services informatiques et pose les bases pour les prochaines étapes. Le but ici est de déterminer quels services seront offerts et quelles fonctionnalités devront être développées, en détaillant :

- La définition des marchés;

- Le développement des atouts stratégiques;
- La préparation du déploiement;
- La définition des offres.

Design de service

C'est dans cette partie qu'il faut planifier et établir les nouvelles stratégies IT ainsi que l'amélioration des stratégies existantes. Nous pouvons couvrir de nombreuses problématiques de service management :

- Listing des services;
- Capacité et disponibilité;
- Sécurité de l'information;
- Processus de management des fournisseur de services;

Transition de service

Dans cette partie, les organisations peuvent tester et implémenter leurs nouveaux designs. En suivant ces étapes et en rectifiant les problématiques qui apparaissent, la disponibilité et la résilience des services sera grandement augmentée. Ceci est particulièrement important lorsque nous modifions des services existants.

Le processus de transition de service doit détailler :

- La planification et le support de la transition;
- Le management de la release et du déploiement;
- La validation et le test du service;
- Les processus de management des enseignements acquis;

Fonctionnement du service

Même après la validation des services, ils sont ensuite intégrés dans un environnement réel et de nombreux problèmes apparaîtront éventuellement. Les organisations doivent donc surveiller attentivement la livraison des services et se préparer à :

- Répondre aux demandes clients;
- Résoudre les erreurs de services;
- Résoudre les différents problèmes;
- Opérer des tâches opérationnelles;

Amélioration continue du service

Les organisations ne devraient jamais se contenter de la qualité de leurs services car il y aura toujours des éléments optimisables ou améliorables. Le but de cette partie est d'apprendre à identifier de potentielles améliorations dans les parties précédentes. Lors du

déroulement de l'amélioration continue du service, les quatres étapes suivantes doivent être suivies :

Inspection des services

Ceci consiste à évaluer régulièrement les services, en cherchant notamment à améliorer la qualité du service ou des manières de rendre le service disponible d'un point de vue économique.

Evaluation des processus

Ceci consiste à inspecter l'état et la qualité des processus notamment lorsque que les objectifs du service ne sont pas atteints. Pour cela, il faut effectuer des benchmarks, des audit et des inspections régulières.

Définition des initiatives de ACS

La définition des initiatives d'Amélioration Continue des Services consiste à établir des initiatives précises permettant d'améliorer les services et les processus.

Surveillance des initiatives ACS

Il faut ici vérifier que les initiatives se déroulent comme prévues et procéder à des ajustements si besoin.

Contraintes

Les contraintes du projet au moment de la rédaction de ce document sont les suivantes :

- 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.

Hypothèses

ID	Hypothèse	Impact
1	Plateforme actuelle en mode maintenance	Arrêt de l'amélioration de la plateforme actuelle (jusqu'au changement de plateforme)
2	Coexistence des plateformes	Mise à disposition de nouvelle fonctionnalités et montée en charge progressive
3	Flexibilité des technologies utilisées	Utilisation de technologies actuelles et possibilité de s'adapter aux prochaines nouvelles technologies.
3	Approche lean	Efficacité et autonomie des équipes conservée