
CONTRAT DE CONCEPTION ET DÉVELOPPEMENT DE L'ARCHITECTURE

GOSME ANTHONY

VERSION 1.0



TABLE DES MATIÈRES

1	INFORMATION SUR LE DOCUMENT	4
1.1	OBJET DE CE DOCUMENT	5
2	INTRODUCTION ET CONTEXTES	6
2.1	NATURE DE L'ACCORD	6
3	OBJECTIFS ET PÉRIMÈTRE	7
4	PARTIES PRENANTES	8
5	DESCRIPTION DE L'ARCHITECTURE, PRINCIPES STRATÉGIQUES ET CONDITIONS REQUISES	9
5.1	DESCRIPTION	9
6	LIVRABLE ARCHITECTURAUX	11
6.1	DÉVELOPPEMENT DE L'ARCHITECTURE	11
7	PLAN DE TRAVAIL COMMUN PRIORISE	12
7.1	APPLICATIONS À USAGE COMMUN	12
7.1.1	ORIENTATION SERVICE	12
7.2	PRINCIPES DES DONNÉES	13
7.2.1	LES DONNÉES SONT ACCESSIBLES	13
7.3	PRINCIPES D'APPLICATION	14
7.3.1	INDÉPENDANCE TECHNOLOGIQUE	14
7.4	PRINCIPES TECHNOLOGIQUES	14
7.4.1	MAÎTRISER LA DIVERSITÉ TECHNIQUE	14
7.4.2	INTEROPÉRABILITÉ	15
8	CONDITIONS REQUISES POUR L'ARCHITECTURE	16

9	PLAN DE COMMUNICATION	17
10	RISQUE ARCHITECTURE.....	18
11	PROCEDURE DE CHANGEMENT DE PÉRIMÈTRE	19
12	APPROBATIONS	20

1 INFORMATION SUR LE DOCUMENT

<i>Nom du projet</i>	<i>Nouvelle architecture de commerce en ligne v2</i>
Préparé par	<i>Anthony Gosme, architecte solution</i>
Version	<i>1.0</i>
Titre	<i>Contrat de conception et développement de l'architecture</i>
Courriel	<i>Anthonygosme@ocr.com</i>
Actions	<i>Approbation, révision, information, classement, action requise, participation à une réunion, autre (à spécifier)</i>

1.1 OBJET DE CE DOCUMENT

Les contrats d'architecture sont les accords communs entre les partenaires de développement et les sponsors sur les livrables, la qualité, et la correspondance à l'objectif d'une architecture. L'implémentation réussie de ces accords sera livrée grâce à une gouvernance de l'architecture efficace (voir togaf partie vii, gouvernance de l'architecture).

En implémentant une approche dirigée du management de contrats, les éléments suivants seront garantis :

- un système de contrôle continu pour vérifier l'intégrité, les changements, les prises de décisions, et l'audit de toutes les activités relatives à l'architecture au sein de l'organisation.
- l'adhésion aux principes, standards et conditions requises des architectures existantes ou en développement
- l'identification des risques dans tous les aspects du développement et de l'implémentation des/de l'architecture(s), y compris le développement interne en fonction des standards acceptés, des politiques, des technologies et des produits, de même que les aspects opérationnels des architectures de façon à ce que l'organisation puisse poursuivre son business au sein d'un environnement résilient.
 - un ensemble de processus et de pratiques qui garantissent la transparence, la responsabilité et la discipline au regard du développement et de l'utilisation de tous les artefacts architecturaux
 - un accord formel sur l'organe de gouvernance responsable du contrat, son degré d'autorité, et le périmètre de l'architecture sous la gouvernance de cet organe

Ceci est une déclaration d'intention signée sur la conception et le développement de l'architecture d'entreprise, ou de parties significatives de celles-ci, de la part d'organisations partenaires, y compris les intégrateurs système, fournisseurs d'applications, et fournisseurs de service.

De plus en plus, le développement d'un ou plusieurs domaine(s) d'architecture (business, données, application, technologie) peut être externalisé, avec la fonction d'architecture de l'entreprise fournissant une vue d'ensemble de l'architecture d'entreprise globale, ainsi que la coordination et le contrôle de l'effort total. Dans certains cas, même ce rôle de supervision peut être externalisé, bien que la plupart des entreprises préfèrent conserver cette responsabilité clé en interne.

Quelles que soient les spécificités des dispositions d'externalisation, les dispositions elles-mêmes seront normalement gouvernées par un contrat d'architecture qui définit les livrables, la qualité, et la correspondance à l'objectif de l'architecture développée, ainsi que les processus de collaboration pour les partenaires du développement de l'architecture.

2 INTRODUCTION ET CONTEXTES

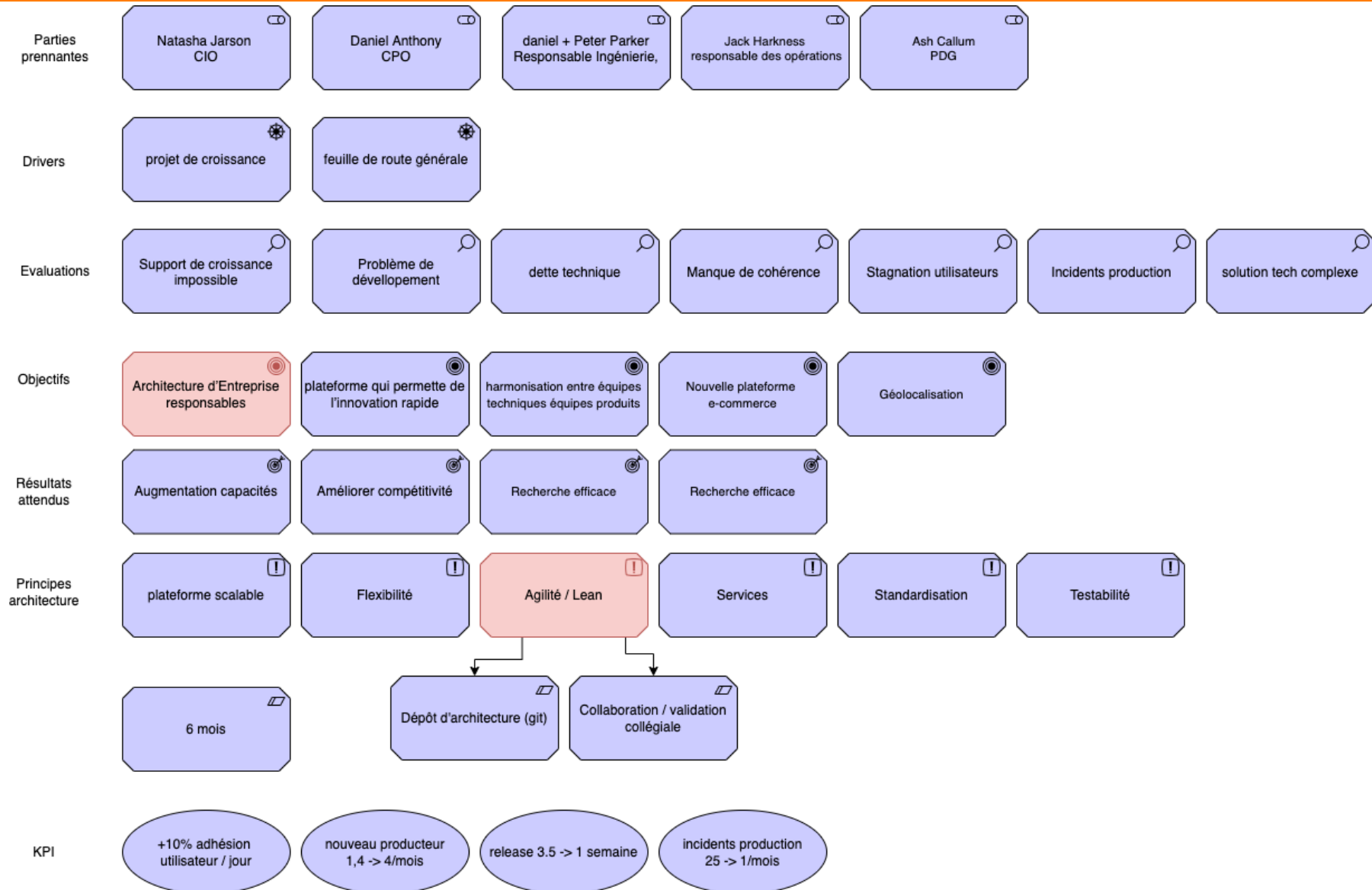
Foosus est une start-up fournissant un service de mise en relation producteur et acheteur de produit écologique. La solution en place accumule une dette technologique, des problèmes de qualité récurrent et des délais de développement trop importants. Afin de soutenir ses stratégies de croissance, Foosus souhaite créer une nouvelle plateforme géo ciblée définie sur de nouvelles bases architecturales.

2.1 NATURE DE L'ACCORD

Phase 1 → définition de l'architecture et construction d'une preuve de concept pour un budget de 50k€ sur 6 mois

Phases suivantes → implémentation et développement de la nouvelle architecture, budget non défini

3 OBJECTIFS ET PÉRIMÈTRE

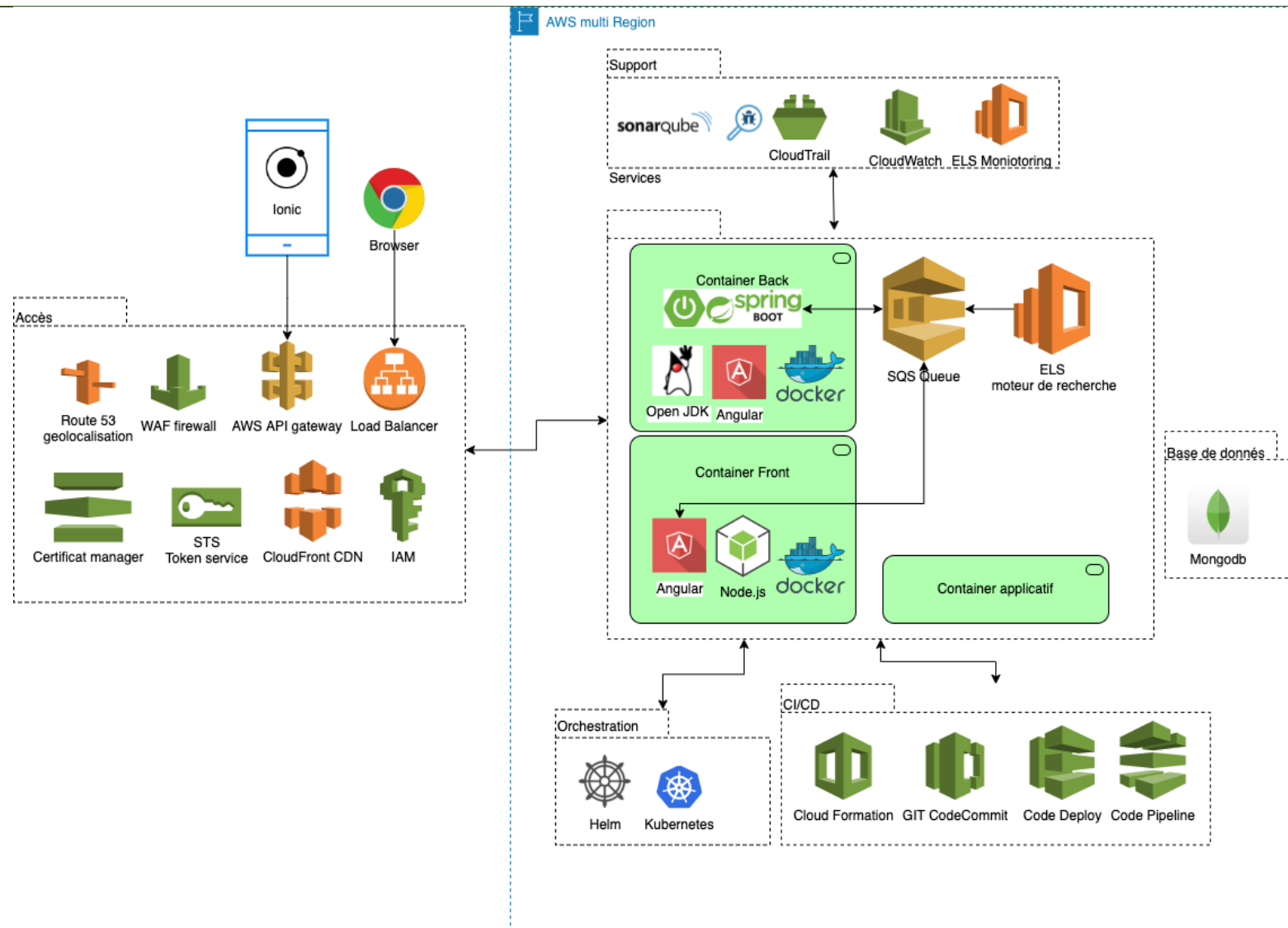


4 PARTIES PRENANTES

Nom	Rôle	Préoccupation	Vision
<i>Anthony Gosme</i>	Architecte solution	<ul style="list-style-type: none">• Qualité architecture• Alignement stratégique• Réponse au besoin	<ul style="list-style-type: none">• Toutes
<i>Natasha Jarson</i>	CIO	<ul style="list-style-type: none">• Innovation technologique• Qualité technologique	<ul style="list-style-type: none">• Pile applicative• Pile technique
<i>Ash callum</i>	CEO	<ul style="list-style-type: none">• Expansion du marché	<ul style="list-style-type: none">• Déclinaison stratégique
<i>Daniel Anthony</i>	CPO	<ul style="list-style-type: none">• Qualité et couverture fonctionnelle	<ul style="list-style-type: none">• Vue projet• Exigences
<i>Christina Orgega</i>	CMO	<ul style="list-style-type: none">• Tau d'engagement	<ul style="list-style-type: none">• Vision business haut niveau
<i>Jo Kumar</i>	CFO	<ul style="list-style-type: none">• Roi• Coût de la solution	<ul style="list-style-type: none">• Délais• Prix
<i>Jack Harkner</i>	Dir. Opération	<ul style="list-style-type: none">• SLA	<ul style="list-style-type: none">•

5 DESCRIPTION DE L'ARCHITECTURE, PRINCIPES STRATÉGIQUES ET CONDITIONS REQUISES

5.1 DESCRIPTION



Les stacks technologiques à utiliser sont et leur rôle définit par le graphique précédent :

- *Fonctionnalité standard (accès, BDD, CI/CD) en cloud native*
- *Orchestration des services avec Kubernetes*
- *Conteneurisation standardisée (docker)*
- *Stack logicielle des services avec langage standards*

6 LIVRABLE ARCHITECTURAUX

6.1 DÉVELOPPEMENT DE L'ARCHITECTURE

Livrables	Responsable	Validation	Mesure	Phases
<i>Document d'architecture</i>	Architecte	CIO	Nombre de phases TOGAF couvertes	1
<i>Poc</i>	Architecte	CIO	Approbation business	1
<i>Environnement de développement AWS</i>	DevOps	Architecte	% de services implémentés	2
<i>Environnement de test AWS</i>	Devops	Architecte	% de services implémentés	2
<i>Environnement de qualification AWS</i>	DevOps	QA	% de services implémentés	3
<i>Environnement de production AWS</i>				4
<i>Test de développement</i>	Dev	Resp. Prod	Score qualité de code, taux de couverture	2+
<i>Test technique</i>	Dev	Architecte	Kpi performance, erreurs	2+
<i>Test fonctionnel</i>	Qa	Resp. Ingé	# user stories couvert	2+
<i>Monitoring</i>	Prod	Resp. Ingé	Nombre d'indicateur	3
<i>Service back end</i>	Dev	Cpo	#user stories couvertes	2+
<i>Service front-end</i>	Dev	Cpo	#user stories couvertes	2+
<i>Application mobile</i>	Dev	Cpo	#user stories couvertes	2+
<i>Moteur de recherche</i>	Dev	Cpo	#user stories couvertes	2+
<i>Registre de qualité d'architecture</i>	Architecte	Architecte	ISO 25k	2+
<i>Registre de risque (projet, architecture, DevOps, qa, production, produit)</i>	Responsables	CIO	Iso 27k ou autre	2+

PHASES



7 PLAN DE TRAVAIL COMMUN PRIORISE

7.1 APPLICATIONS À USAGE COMMUN

Déclaration :

Le développement d'applications utilisées dans toute l'entreprise est préférable au développement d'applications similaires ou en double qui ne sont fournies qu'à une organisation particulière.

Raisonnement :

La capacité de duplication est coûteuse et prolifère des données contradictoires.

Conséquences :

- Les organisations qui dépendent d'une capacité qui ne sert pas l'ensemble de l'entreprise doivent passer à la capacité de remplacement à l'échelle de l'entreprise ; cela nécessitera l'établissement et le respect d'une politique
- Les organisations ne seront pas autorisées à développer des capacités pour leur propre usage qui sont similaires/dupliquées des capacités à l'échelle de l'entreprise ; de cette manière, les dépenses de ressources rares pour développer essentiellement la même capacité de manière légèrement différente seront réduites
- Les données et les informations utilisées pour soutenir la prise de décision de l'entreprise seront normalisées dans une bien plus grande mesure qu'auparavant

En effet, les capacités organisationnelles plus petites qui produisaient des données différentes (qui n'étaient pas partagées entre d'autres organisations) seront remplacées par des capacités à l'échelle de l'entreprise. L'impulsion pour ajouter à l'ensemble des capacités à l'échelle de l'entreprise peut bien provenir d'une organisation qui démontre de manière convaincante la valeur des données/informations précédemment produites par sa capacité organisationnelle, mais la capacité résultante fera partie du système à l'échelle de l'entreprise, et les données qu'il produit seront partagées dans toute l'entreprise.

7.1.1 ORIENTATION SERVICE

DÉCLARATION :

L'architecture est basée sur une conception de services qui reflètent les activités commerciales réelles comprenant les processus commerciaux de l'entreprise (ou interentreprises).

Raisonnement :

L'orientation service offre l'agilité de l'entreprise et un flux d'informations sans frontières.

Conséquences :

- La représentation de service utilise des descriptions métier pour fournir un contexte (c'est-à-dire, processus métier, objectif, règle, politique, interface de service et composant de service) et implémente des services à l'aide de l'orchestration de service
- L'orientation des services impose des exigences uniques à l'infrastructure, et les implémentations doivent utiliser des normes ouvertes pour réaliser l'interopérabilité et la transparence de l'emplacement
- Les implémentations sont spécifiques à l'environnement ; ils sont limités ou activés par le contexte et doivent être décrits dans ce contexte
- Une gouvernance solide de la représentation et de la mise en œuvre des services est requise
- Un "test décisif", qui détermine un "bon service", est requis

7.2 PRINCIPES DES DONNÉES

7.2.1 LES DONNÉES SONT ACCESSIBLES

Déclaration :

Les données sont accessibles aux utilisateurs pour exécuter leurs fonctions.

Raisonnement :

Un large accès aux données conduit à l'efficiency et à l'efficacité de la prise de décision et permet une réponse rapide aux demandes d'informations et à la prestation de services. L'utilisation des informations doit être considérée du point de vue de l'entreprise pour permettre l'accès à une grande variété d'utilisateurs. Le temps du personnel est économisé et la cohérence des données est améliorée.

Conséquences :

- Il s'agit de l'un des trois principes étroitement liés concernant les données : les données sont un atout ; les données sont partagées ; et les données sont facilement accessibles

L'implication est qu'il existe une tâche d'éducation pour s'assurer que toutes les organisations au sein de l'entreprise comprennent la relation entre la valeur des données, le partage des données et l'accessibilité aux données.

- L'accessibilité implique la facilité avec laquelle les utilisateurs obtiennent des informations
- La manière dont les informations sont accessibles et affichées doit être suffisamment adaptable pour répondre à un large éventail d'utilisateurs d'entreprise et leurs méthodes d'accès correspondantes
- L'accès aux données ne constitue pas une compréhension des données - le personnel doit veiller à ne pas mal interpréter les informations
- L'accès aux données n'accorde pas nécessairement à l'utilisateur des droits d'accès pour modifier ou divulguer les données

Cela nécessitera un processus d'éducation et un changement dans la culture organisationnelle, qui soutient actuellement une croyance en la « propriété » des données par les unités fonctionnelles.

7.3 PRINCIPES D'APPLICATION

7.3.1 INDÉPENDANCE TECHNOLOGIQUE

Déclaration :

Les applications sont indépendantes des choix technologiques spécifiques et peuvent donc fonctionner sur une variété de plates-formes technologiques.

Raisonnement :

L'indépendance des applications par rapport à la technologie sous-jacente permet aux applications d'être développées, mises à niveau et exploitées de la manière la plus rentable et la plus opportune. Sinon, la technologie, qui est sujette à une obsolescence continue et à la dépendance vis-à-vis des fournisseurs, devient le moteur plutôt que les besoins des utilisateurs eux-mêmes.

Sachant que chaque décision prise en matière d'informatique nous rend dépendants de cette technologie, l'intention de ce principe est de garantir que le logiciel d'application ne dépend pas de matériel et de logiciels de systèmes d'exploitation spécifiques.

Conséquences :

- Ce principe nécessitera des normes qui prennent en charge la portabilité
- Pour les applications commerciales sur étagère (cots) et gouvernementales sur étagère (gots), les choix actuels peuvent être limités, car bon nombre de ces applications dépendent de la technologie et de la plate-forme.
- Des interfaces de sous-système devront être développées pour permettre aux applications héritées d'interagir avec les applications et les environnements d'exploitation développés dans le cadre de l'architecture d'entreprise
- Le middleware doit être utilisé pour découpler les applications de solutions logicielles spécifiques
- À titre d'exemple, ce principe pourrait conduire à l'utilisation de java et de futurs protocoles de type java, qui accordent une grande priorité à l'indépendance de la plate-forme.

7.4 PRINCIPES TECHNOLOGIQUES

7.4.1 MAÎTRISER LA DIVERSITÉ TECHNIQUE

Déclaration :

La diversité technologique est contrôlée pour minimiser le coût non négligeable du maintien de l'expertise et de la connectivité entre plusieurs environnements de traitement.

Raisonnement :

Il existe un coût réel et non négligeable de l'infrastructure requise pour prendre en charge les technologies alternatives pour les environnements de traitement. D'autres coûts d'infrastructure sont encourus pour maintenir l'interconnexion et la maintenance de plusieurs constructions de processeurs.

Limiter le nombre de composants pris en charge simplifiera la maintenabilité et réduira les coûts.

Les avantages commerciaux d'une diversité technique minimale comprennent : un conditionnement standard des composants ; impact prévisible de la mise en œuvre ; des évaluations et des rendements prévisibles ; tests redéfinis ; statut d'utilité ; et une flexibilité accrue pour s'adapter aux progrès technologiques. La technologie commune à l'ensemble de l'entreprise apporte les avantages des économies d'échelle à l'entreprise. Les coûts d'administration technique et de support sont mieux maîtrisés lorsque des ressources limitées peuvent se concentrer sur cet ensemble de technologies partagées.

Conséquences :

- Les politiques, les normes et les procédures qui régissent l'acquisition de la technologie doivent être directement liées à ce principe
- Les choix technologiques seront limités par les choix disponibles dans le plan technologique

Des procédures visant à augmenter l'ensemble de technologies acceptables pour répondre aux exigences en constante évolution devront être élaborées et mises en place.

- La base technologique n'est pas gelée

Les avancées technologiques sont les bienvenues et modifieront le modèle technologique lorsque la compatibilité avec l'infrastructure actuelle, l'amélioration de l'efficacité opérationnelle ou une capacité requise aura été démontrée.

7.4.2 INTEROPÉRABILITÉ

Déclaration :

Les logiciels et le matériel doivent être conformes aux normes définies qui favorisent l'interopérabilité des données, des applications et de la technologie.

Raisonnement :

Les normes aident à assurer la cohérence, améliorant ainsi la capacité à gérer les systèmes et à améliorer la satisfaction des utilisateurs, et à protéger les investissements informatiques existants, maximisant ainsi le retour sur investissement et réduisant les coûts. Les normes d'interopérabilité aident en outre à assurer le support de plusieurs fournisseurs pour leurs produits et facilitent l'intégration de la chaîne d'approvisionnement.

Conséquences :

- Les normes d'interopérabilité et les normes de l'industrie seront suivies à moins qu'il n'y ait une raison commerciale impérieuse de mettre en œuvre une solution non standard
- Un processus pour établir des normes, les examiner et les réviser périodiquement et accorder des exceptions doit être établi
- Les plates-formes informatiques existantes doivent être identifiées et documentées

8 CONDITIONS REQUISES POUR L'ARCHITECTURE

Réf.	Qualité architecturale	Besoin non fonctionnel	Solution
Q1	Disponibilité	Mise à jour service sans interruption	Helm & Kubernetes
Q2	Disponibilité	Mise à jour bdd sans interruption	Mongodb managed multi region aws
Q3	Évolutivité	Version de taille réduite	Layers docker
Q4	Performance	Performance similaire dans différentes régions	Aws multi-région
Q5	Qualité	Fiabilité nouvelle version	Devops
Q6	Qualité	Problème intégrations	Microservice + DevOps + ci/cd
Q7	Qualité	Tests et qualification rapides	Env. Dev + test + qualification + ci/cd
Q8	Évolutivité	Architecture évolutive	Microservice
Q9	Disponibilité	Accessibilité téléphone et browser	Ionic
Q10	Performance	Comptabilité faible bande passante	Application simple page : Angular
Q11	Disponibilité	Géolocalisation des ressources	AWS CloudFront, DNS
Q12	Évolutivité	Cots open source et dev spécifiques	
Q13	Évolutivité	Pile technologique unique	Docker + openjdk+ spring boot + angular

9 PLAN DE COMMUNICATION

Évènements	Fréquence	Participants	Responsables	Contenue
<i>Daily scrum</i>	Quotidien	Développeur, qa, ux, DevOps	Scrum master	Suivit avancement/blocage
<i>Sprint planning</i>	Bimensuel	Développeur, qa, ux, DevOps	Scrum master & CPO & architecte	Planification des lots de livrable
<i>Sprint review</i>	Bimensuel	Développeur, qa, ux, DevOps	Scrum master & CPO & architecte	Validation fonctionnelle
<i>Sprint review architecture</i>	Bimensuel	Développeur, qa, ux, DevOps	Scrum master & architecte	Validation non fonctionnelle
<i>Sprint rétro</i>	Bimensuel	Développeur, qa, ux, DevOps	Scrum master	Amélioration processus
<i>COPIL - comité pilotage</i>	Bimensuel	CIO, CPO, resp. Ing., resp. Prod., architecte	CIO	Suivi planification & ressource
<i>COOP - comité opérationnel</i>	Bimensuel	CEO, CIO, CFO, CPO, resp ing., resp prod., architecte	Ceo	Suivi indicateurs qualité, projets, sécurité. Stratégie et arbitrage

10 RISQUE ARCHITECTURE

Catégorie	Prévention	Détection	Réponse	Proba.	Impact	Indéter.	Criticité
Changement périmètre	Meeting	Meeting	Cpo & archi	3	2	2	12
Planning	Analyse risque & meeting	Kpi	Cio	2	2	3	12
Gouvernance	Analyse risque & meeting	Meeting	Ceo	2	3	3	18
Complexité	Prototype	Review	Archi	3	2	2	12
Innovation	Prototype	Review	Archi	3	2	2	12
Qualité archi applicative	Plans archi	Iso 25k	Archi	2	2	2	8
Qualité archi donnée	Plans archi	Iso 25k	Archi	2	2	2	8
Qualité usage	Plans archi	Iso 25k	Archi	2	2	2	8
Budget	Discussion budget & scope	Négo scope & coût	Cfo	2	2	2	8

11 PROCÉDURE DE CHANGEMENT DE PÉRIMÈTRE

Type de changement	Impacts
<i>Changement non impactant</i>	Incorporé par les équipes dev. Ou prod.
<i>Changement besoin impactant la roadmap produit</i>	Backlog produit par le CPO COPIL -validation du CIO
<i>Changement technique impactant l'architecture</i>	Backlog architecture et document d'architecture par l'architecte Copil - validation du CIO

12 APPROBATIONS

<i>Nom</i>	<i>Poste</i>	<i>Date</i>	<i>Signature</i>
Ash callum	Chief executive officer		
Natasha Jaron	Chief information officer		
Daniel anthony	Chief product officer		
Christina Orgega	Chief marketing officer		
Jo kumar	Chief financial officer		
Anthony Gosme	Architecte logiciel		