Principes de l’architecture

Client : Consortium MedHead

Projet : Preuve de concept (l'allocation de lits d'hôpital pour les urgences)

Afin d'orienter les efforts, les membres du Consortium ont collaboré à la définition d'un ensemble de principes architecturaux par domaine, que nous privilégions. Comme pour tout principe, ceux-ci devraient être appliqués à tous les projets, et toute disparité évidente devrait être clairement justifiée par le contexte.

Résumé des principes

A. Principes d'architecture métier....3
Principe A1 : Primauté des principes....3
Principe A2 : Maximiser les avantages pour l'entreprise....3
Principe A3 : Conformité aux lois et aux règlements....4
Principe A4 : Adhésion au serment d'Hippocrate à tous les niveaux....4
B. Principes de l'architecture informatique (système, données, solutions, sécurité et opérations)...5
Principe B1 : Continuité des activités des systèmes critiques pour les patients....5
Principe B2 : Clarté grâce à une séparation fine des préoccupations....6
Principe B3 : Intégration et livraison continues....6
Principe B4 : Tests automatisés précoces, complets et appropriés....7
Principe B5 : Sécurité de type « shift-left »....8
Principe B6 : Possibilité d'extension grâce à des fonctionnalités pilotées par les événements....9
C. Méthodologie architecturale et principes de processus....9
Principe C1 : Personnalisation de l'ADM TOGAF 9.2....9
Principe C2 : Référentiel d'architecture centralisé et organisé comme source de référence....10
Principe C3 : Normes ouvertes convenues pour garantir des normes élevées....10
Principe C4 : Favoriser une culture de "learning" avec des preuves de concept, des prototypes et des Spike....12
i) Fournir une hypothèse pour chaque apprentissage....12
ii) Isoler les preuve de concept des données et des systèmes de production....12
iii) Utiliser des données factices ou anonymisées....13
iv) Assouplir la conformité, mais tenir compte des conséquences....13
v) Les principes de base de l'ingénierie, de la livraison et des tests ne doivent pas être assouplis pour l'architecture de la PoC....13
vi) Plans de test comme outils de communication des exigences....13
vii) Tester les rapports d'exécution pour documenter le comportement pris en charge....14

A. Principes d'architecture métier

Principe A1 : Primauté des principes

Déclaration :

Les principes énoncés ici s'appliquent à tous les membres du Consortium, que nous appellerons collectivement l'entreprise.

Raisonnement :

La seule façon de fournir aux décideurs un niveau cohérent et mesurable d'informations de qualité est que toutes les organisations respectent ces principes.

Implications :

Sans ce principe, des exclusions, du favoritisme et des incohérences mineraient rapidement la gestion et la pertinence des décisions concernant l'architecture.

Les initiatives ne débuteront pas tant que leur conformité aux principes n'aura pas été examinée. Un conflit avec un principe sera résolu en modifiant le cadre de l'initiative.

Principe A2 : Maximiser les avantages pour l'entreprise

Déclaration :

Les décisions d'architecture et de conception général sont prises pour fournir un avantage maximum à l'entreprise dans son ensemble, dans le cadre des efforts entrepris pour améliorer les soins dispensés aux patients touchés par ces décisions.

Raisonnement :

Ce principe incarne « l'engagement sans faille à servir autrui ». Les décisions prises selon la perspective de l'entreprise ont une plus grande valeur à long terme que les décisions prises dans une perspective organisationnelle particulière. Un retour surinvestissement maximal nécessite des décisions architecturales et de conception pour respecter les moteurs et les priorités à l'échelle de l'entreprise. Les intérêts d'aucun groupe minoritaire ne porteront atteinte aux intérêts de l'entreprise. Cependant, ce principe n'empêchera aucun groupe minoritaire de faire son travail.

Principe A3 : Conformité aux lois et aux règlements

Déclaration :

Le système d'information, les processus métier et les livrables doivent être conformes à toutes les lois, politiques et réglementations pertinentes.

Raisonnement :

La politique de l'entreprise exige le respect des lois, politiques et réglementations. Cela n'exclut pas les améliorations des processus métier qui conduisent à des changements de politiques et de réglementations.

Implications :

L'entreprise doit être attentive à se conformer aux lois, réglementations et politiques externes concernant la collecte, la conservation et la gestion des données, formation et accès aux réglementations. L'efficacité, le besoin et le bon sens ne sont pas les seuls moteurs. Les changements au niveau des lois et des réglementations peuvent entraîner des changements dans nos processus ou applications.

Principe A4 : Adhésion au serment d'Hippocrate à tous les niveaux

Déclaration :

En tant qu'entreprise à visé médical dont le but est d'améliorer les soins dispensés aux patients, toutes les décisions organisationnelles doivent adhérer au serment d'Hippocrate (« d'abord ne pas nuire, ensuite soigner ») en ce qui concerne les soins prodigués par tous les membres du Consortium et leur personnel interne.

Raisonnement :

La politique d'entreprise consiste à respecter les principes de soins aux patients et à reconnaître que les décisions organisationnelles peuvent avoir un impact sur leur vie.

Implications :

À tous les niveaux, l'entreprise doit être attentive à prendre des décisions visant à apporter de la valeur (économique et thérapeutique) au patient ainsi qu'aux organisations membres. Des conséquences financières et liées à la réputation peuvent s'en suivre directement si le patient subit un préjudice, intentionnellement ou par négligence.

B. Principes de l'architecture informatique (système, données, solutions, sécurité et opérations)

Principe B1 : Continuité des activités des systèmes critiques pour les patients

Déclaration :

Les opérations essentielles à la santé des patients, ainsi que les autres pratiques de soin, doivent être assurées malgré les interruptions du système.

Raisonnement :

Étant donné que les soins aux patients sont considérés comme une priorité, tous les systèmes critiques doivent être construits conformément aux principes de tolérance aux pannes, de telle sorte que la priorité soit accordée à la fiabilité de ces systèmes tout au long de leur conception, de leur déploiement, de leur développement et de leur utilisation. Les partenaires médicaux, les fonctions métiers et techniques de l'entreprise doivent être en mesure de remplir leurs tâches indépendamment des événements externes. Les pannes matérielles, les attaques ciblées, les catastrophes naturelles et la corruption des données ne doivent pas perturber ou à arrêter les activités de l'entreprise.

Implications :

La dépendance vis-à-vis des applications système partagées exige que les risques d'interruption des activités soient établis à l'avance et traités lorsqu'ils se présentent. La gestion comprend, sans s'y limiter :

- Principes SRE (Site Reliability Engineering) qui surveillent et mesurent en continu les SLI cibles (Service Level Indicators).
- Examens périodiques de la santé et des risques du système.
- Tests incrémentiels de performances, de vulnérabilité et d'exposition pour chaque incrément de la plateforme technique.
- Services critiques conçus pour assurer la continuité des fonctions de l'entreprise grâce à des capacités redondantes ou alternatives.
- La récupérabilité, la redondance et la maintenabilité doivent être prises en compte au moment de la conception.
- Les demandes doivent être évaluées selon leur criticité et leur impact sur la mission de l'entreprise, laquelle est d'assurer les soins aux patients.

- Des plans de reprise doivent exister pour tous les systèmes critiques.

Principe B2 : Clarté grâce à une séparation fine des préoccupations

Déclaration :

Il faut éviter de regrouper ensemble des responsabilités disparates. Il faut éviter les systèmes centralisés.

Raisonnement :

Par entropie naturelle, les architectures complexes ont tendance à évoluer au fil du temps vers des réseaux régis par des dépendances complexes et difficiles à définir, et des responsabilités mal placées. Les composants d'une telle architecture sont souvent étroitement et fortement couplés.

Cela peut, au fil du temps, entraîner une perte des fonctions de l'architecture qui limite l'agilité d'une plateforme à répondre à l'évolution des besoins de l'entreprise ou des patients. Il faut connaître les limites du système. Il faut rendre le système transparents, c'est à dire :

- tout est bien découpé et on sait ce qu'il fait exactement
- on connaît les dépendances entre chaque fonction.

Implications :

Les décisions architecturales doivent suivre les principes et les meilleures pratiques de la conception pilotée par le domaine et desarchitectures de microservices. Cela implique un partenariat actif entre les équipes techniques et métier pour fournir des capacités à l'entreprise en utilisant un modèle partagé et un langage qui reflète le domaine des soins aux patients. Les dépendances étroites entre les capacités techniques doivent être identifiées et, dans la mesure du possible, doivent apporter une réponse aux situations problématiques traités dans le contexte métier et dans le monderéel. Les solutions techniques doivent toutes être justifiées et modélisées en fonction de leur contribution globale aux scénarios de soins aux patients.

Principe B3 : Intégration et livraison continues

Déclaration :

L'intégration et la livraison continues de petits changements incrémentiels sont favorisées par rapport aux temps de cycle lents et aux intégrations majeures.

Raisonnement :

L'intégration continue de petites fonctions et pipelines jusqu'à la production réduit les risques et

permet d'avoir un retour précoce au sein des grandes équipes en cas de problèmes d'intégration. Une cadence de livraison rapide et régulière encourage également les équipes à réduire les risques en proposant des tests plus approfondis et de meilleurs résultats.

Implications :

Les pipelines CI/CD doivent être facilement (ou automatiquement) déclenchés par des événements appropriés dépendant de l'état du code poussé sur le répertoire. Pour faciliter cela, les points suivants sont également à considérer :

- Les fonctionnalités doivent être clairement traçables dans le contrôle de version en utilisant des techniques d'étiquetage appropriées.
- Les exécutions CI/CD doivent être liées à une livraison de fonctionnalité donnée.
- Les exécutions CI/CD génèrent des journaux ou des sorties clairs qui peuvent être analysés pour isoler les builds en échec ou les erreurs dans les étapes de build, de test et de livraison.

Principe B4 : Tests automatisés précoces, complets et appropriés

Déclaration :

Les applications doivent être construites à l'aide de tests automatisés qui garantissent la fiabilité à la fois fonctionnelle et non fonctionnelle de la mise en œuvre.

Raisonnement : Les bogues logiciels sont inévitables et peuvent être causés par des erreurs de code ou d'analyse. Des tests précoces garantissent que le logiciel est construit selon les spécifications et que chaque spécification est validée avant d'investir dans de mauvaises solutions.

Implication :

Ce principe encourage l'utilisation de techniques de développement dirigé par des tests (TDD pour Test-Driven Development en anglais). Afin de valider rapidement les exigences, il est recommandé d'utiliser le langage du domaine métier lors des tests.

Les premières exigences devraient être rédigées sous une forme qui facilite les tests.

Implications :

Les équipes devraient suivre la pyramide des tests et mettre en œuvre un niveau de test approprié pour chacune des catégories de tests suivantes :

- Integration continue
- E2E

Lorsque les services sont interdépendants, il est également conseillé d'envisager des tests centrés sur le consommateur.

Principe B5 : Sécurité de type « shift-left »

Déclaration :

Le risque global de sécurité de la plateforme est réduit en spécifiant et en respectant les exigences de sécurité dès le début de chaque incrément.

Raisonnement :

Il a été démontré que l'omission de problèmes de sécurité lors de la conception et de la mise en œuvre d'une solution entraîne souvent un coût et un risque plus élevés pour l'entreprise, car ces problèmes ne sont détectés que plus tard. Les problèmes de sécurité non identifiés dans de tels scénarios présentent un risque plus élevé pour l'entreprise s'ils ne sont pas détectés ou s'ils deviennent des vulnérabilités exploitées ou connues.

Implications :

En considérant, par incréments, les exigences de sécurité de chaque plateforme et chaque itération logicielle, ce risque est compensé et peut se traduire par une culturede la sécurité d'abord, qui diminue le risque de non-respect des réglementations et de perte de la confiance des patients et des médecins.

Les pratiques suivantes devraient être examinées et adaptées pour permettre une culture de la sécurité de type « shift-left » :

- Utiliser les ressources de sécurité actuellement limitées du Consortium (et du secteur dans son ensemble) comme des *catalyseurs* pour encourager une sécurité de type « shift-left ».
- Utiliser des méthodes pour prendre en compte les exigences non fonctionnelles liées à la sécurité, en fonction du risque, lors de la définition précoce des scénarios et des exigences.
- Tests de sécurité continus et automatisés pour réduire le risque dû à une erreur ou à une omission humaine.
- Sensibiliser le personnel à la sécurité et l'encourager à suivre les bonnes pratiques à l'échelle de l'entreprise.

Principe B6 : Possibilité d'extension grâce à des fonctionnalités pilotées par les événements

Déclaration :

Tous les composants techniques doivent être conçus pour publier en continu les événements métiers, dont l'apparition déclenche d'autres fonctions métiers.

Raisonnement :

Les systèmes initialement conçus pour assumer une seule responsabilité peuvent au fil du temps s'étendre à de nouveaux comportements, qui ne sont pas toujours directement liés à la responsabilité d'origine. De telles extensions peuvent à la fois ralentir le système d'origine, brouiller sa responsabilité et violer le principe de la responsabilité unique.

Implications :

Les architectures pilotées par les événements simplifient l'extension des systèmes existants avec de nouvelles capacités qui réagissent aux événements métiers qui se produisent ailleurs sur la plateforme. Cela peut également présenter des avantages en termes de performances, grâce à une mise à l'échelle horizontale des abonnés aux événements métiers.

C. Méthodologie architecturale et principes de processus

Principe C1 : Personnalisation de l'ADM TOGAF 9.2

Déclaration :

L'architecture métier sera façonnée par la personnalisation et l'amélioration continue d'un cadre d'architecture adapté à partir de l'ADM de TOGAF 9.2.

Raisonnement :

Afin de fournir un langage et une lisibilité communs pour l'architecture, il est nécessaire de partir d'une base bien définie et offrant plusieurs options. Le TOGAF d'OpenGroup fournit un cadre centré sur la gestion des exigences.

Ce TOGAF comprend la gouvernance et les conseils qui soutiennent la spécialisation d'un cadre et d'une méthodologie permettant de déterminer quels niveaux de rigueur sont requis pour les fonctionnalités liées à la sécurité des patients, à la confidentialité des données, à la sécurité globale des informations et au respect de l'exactitude des informations.

Implications :

L'ADM de TOGAF comprend la gouvernance et les protections nécessaires pour garantir une architecture capable de répondre aux exigences éthiques, métier et d'état concernant les logiciels centrés sur le patient.

L'architecte logiciel du Consortium devra collaborer avec les parties prenantes médicales, métier et techniques pour convenir d'un cadre architectural, qui pourra être modifié selon les projets et les différents contextes métier.

Principe C2 : Référentiel d'architecture centralisé et organisé comme source de référence

Déclaration :

Toutes les informations pertinentes sur le plan architectural devraient être disponibles dans un répertoire d'architecture central géré en permanence par la fonction d'architecture métier, qui en sera responsable.

Raisonnement :

Lorsque les artéfacts d'architecture sont dispersés sur plusieurs systèmes, il devient difficile, au fil du temps, pour tous les partenaires d'avoir une vision claire et *à jour* de l'état de l'architecture.

Implications :

Un répertoire centralisé simplifie le problème de la consolidation et de la conservation de tous les artéfacts, décisions et contenus actuels relatifs à l'architecture dans un paysage d'exigences métier et techniques en constante évolution.

Principe C3 : Normes ouvertes convenues pour garantir des normes élevées

Déclaration :

L'application de normes ouvertes et de meilleures pratiques convenues peut soutenir l'organisation en lui apportant les connaissances et l'expertise du secteur.

Raisonnement :

Les principes décrits ici s'appuient sur les meilleures pratiques du secteur qui ont évolué par des normes, des méthodes éprouvées et des directives. L'utilisation des normes associées peut permettre de mieux tirer parti des avantages découlant des principes avec lesquels nous nous alignons.

Implications :

Nous encouragerons et soutiendrons, au moins, les normes ouvertes et les meilleures pratiques architecturales listées ci-dessous. Toutes les conceptions et architectures doivent être conçues, le cas échéant, pour prendre en charge des extensions.

Il est conseillé de documenter la manière dont les extensions prennent en charge ces normes ou sont conçues pour être étendues à cette fin.

- Architectures pilotées par les événements
    - Source des événements
- Architectures microservicesO Spécification OpenAPI des contrats de service
    - Maillages de services
        - Observabilité des services
        - Surveillance des services
        - Découverte des services
        - Visibilité de l'intégration des services.
    - Déploiement via une infrastructure conteneurisée, immuable et reproductible
- Conception pilotée par le domaine
- Développement centré sur le comportement
    - Pour garantir l'exactitude des résultats attendus centrés sur le patient.
    - Pour soutenir un développement aligné avec un langage omniprésent.
- Tolérance aux pannes
- Intégration d'OpenID Connect avec les fournisseurs d'identité des patients gérés par l'État.
- Choix de la technologie :
    - Devrait favoriser les langages JVM en raison des directives du Consortium.
- Documentation :
    - Devrait favoriser Javadoc ou NDoc pour le code source et les milestones ou ASCIIdoc

pour la documentation au niveau du projet.

Comme cela définit un état cible, il est acceptable de faire des compromis, mais ces derniers doivent être documentés et justifiés.

Principe C4 : Favoriser une culture de "learning" avec des preuves de concept, des prototypes et des Spike

Déclaration :

L'entreprise encourage les implémentations centrées sur l'apprentissage qui réduisent les risques, valident les hypothèses et investissent dans l'apprentissage nécessaire pour faire évoluer la plateforme de manière responsable.

Raisonnement :

Le Consortium encourage collectivement l'utilisation de validations de principes, de Spike et de prototypes, ainsi que d'autres moyens d'enquête pour atteindre *un état d'échec sans danger* dans les zones où les informations disponibles sont insuffisantes pour comprendre le risque lié à la prise de décisions de conception ou de mise en œuvre spécifiques au niveau de la production. Le coût de l'investissement dans les efforts d'apprentissage pour réduire les risques est encouragé dans toute l'entreprise afin de protéger les intérêts des patients, des partenaires et de l'entreprise elle-même.

Implications :

Les partenaires du Consortium conviennent collectivement de stimuler une culture de prise de décision fondée sur des preuves et centrée sur l'apprentissage.

Ce faisant, les exceptions et considérations suivantes devraient s'appliquer :

i) Fournir une hypothèse pour chaque apprentissage

- Toutes les implémentations liées à l'apprentissage doivent être accompagnées d'une *hypothèse* définissant l'apprentissage souhaité et permettant de mesurer si ce résultat d'apprentissage a été atteint.

ii) Isoler les preuve de concept des données et des systèmes de production

- Des mesures doivent être prises pour atténuer ou éliminer le risque d'impact sur les patients lorsqu'il existe un risque de nuire au patient ou à l'entreprise. Par exemple, l'apprentissage peut être mené de manière isolée dans un environnement artificiel afin d'éviter l'impact sur les systèmes de production.

iii) Utiliser des données factices ou anonymisées

- Les données des patients utilisées pour les activités d'apprentissage à haut risque doivent être protégées, afin d'éviter un impact sur la sécurité des données ou les soins aux patients. Les PoC devrait utiliser des données anonymisées ou factices lorsque cela est possible.

iv) Assouplir la conformité, mais tenir compte des conséquences

- Les normes de gouvernance et les niveaux de conformité peuvent être assouplis lorsque des mesures sont prises pour protéger les systèmes de production et les données des patients. Les PoC isolées des données réelles des patients et des systèmes de production ne sont pas régies par des normes externes ou une quelconque gouvernance d'entreprise en matière de séparabilité.
- Lorsque les normes et la gouvernance ne sont pas pleinement respectées, les responsables de la mise en œuvre et les concepteurs devraient réfléchir à la manière dont ces prototypes ou ces mises en œuvre centrés sur l'apprentissage peuvent fournir des leçons dans les mises en œuvre finales en production.
- Il est fortement déconseillé de produire directement des prototypes. Il convient plutôt de veiller à ce que les conceptions tiennent compte des effets secondairesde la production qui peuvent invalider tout apprentissage. Par exemple, l'omission de problèmes de sécurité ou la mauvaise estimation du volume de données attendu peut entraîner des problèmes de performances qui invalident les apprentissages tirés d'un tel prototype non évolutif.
- Les tests de performance des prototypes et des implémentations centrées sur l'apprentissage devraient valider les algorithmes clés faisant partie de cette échelle d'apprentissage.

v) Les principes de base de l'ingénierie, de la livraison et des tests ne doivent pas être assouplis pour l'architecture de la PoC.

La validation de principe doit viser spécifiquement à respecter les principes suivants :

- Principe B1 : Continuité des activités des systèmes critiques pour les patients
- Principe B2 : Clarté grâce à une séparation fine des préoccupations
- Principe B3 : Intégration et livraison continues
- Principe B4 : Tests automatisés précoces, complets et appropriés

vi) Plans de test comme outils de communication des exigences

- Les livrables avec des plans de test autodocumentés sont préférables aux plans de test documentés en externe.

- La preuve de concept doit comporter des plans de test décrivant comment le produit doit se comporter.
- Les plans de test doivent utiliser des scénarios BDD (behaviour-driven development – voir C3) pour décrire les critères d'acceptation métier qui sont dans la portée.
- Les plans de test doivent utiliser le langage commun de l'entreprise et être compréhensibles par les partenaires techniques et non techniques.

vii) Tester les rapports d'exécution pour documenter le comportement pris en charge

Pour prendre en charge la visibilité des comportements attendus, l'apprentissage continu et la transparence concernant l'état du logiciel :

- Les PoC devraient avoir des pipelines CI qui exécutent des tests et produisent des rapports d'exécution des tests
- Les environnements CI doivent permettre aux propriétaires des logiciels d'inspecter les exécutions passées et les dégradations de la build qui peuvent affecter les l'hypothèse, en ligne avec le principe B3.