



Etude de cas Amélioration de solution

Expert en ingénierie du logiciel – Niveau 7 – GROUPE ESIEA INTECH Bloc de compétences : Conception avancée de l'architecture de logiciel Mastère Expert en Ingénierie Logicielle

Etude de cas

Proof Of Concept





Etude de cas Amélioration de solution

Expert en ingénierie du logiciel – Niveau 7 – GROUPE ESIEA INTECH Bloc de compétences : Conception avancée de l'architecture de logiciel Mastère Expert en Ingénierie Logicielle

Scénario

Votre client est un regroupement d'institutions médicales. Actuellement, les organisations membres du consortium utilisent une grande variété de technologies et d'appareils. Ils souhaitent une nouvelle plateforme pour unifier leurs pratiques. La technologie Java est pour eux un socle technique fiable pour ce projet.

Le comité s'attend à ce que la plateforme pallie les risques liés au traitement des recommandations de lits d'hôpitaux dans des situations d'intervention d'urgence, la grosse problématique actuelle.

BRIEF

La preuve de concept peut restreindre sa portée en se concentrant principalement sur la réponse aux préoccupations énoncées dans le document des exigences de la PoC. Voici des exemples de simplifications potentielles :

- Utiliser uniquement des interfaces tampons ou des implémentations factices de systèmes auxiliaires
- Faire des hypothèses simplifiées sur les contrats de ces systèmes en aval
- Documenter les lacunes d'un prototype et un plan pour y remédier dans tout système résultant.

La méthode de développement de l'architecture TOGAF (ADM) définit les meilleures pratiques pour le développement de l'architecture.

Le principe d'Architecture est défini dans un document Annexe complet, l'annexe 1.

Un document de définition de l'architecture est disponible en Annexe 2.





Etude de cas Amélioration de solution

Expert en ingénierie du logiciel – Niveau 7 – GROUPE ESIEA INTECH Bloc de compétences : Conception avancée de l'architecture de logiciel Mastère Expert en Ingénierie Logicielle

Instructions

Dans la mesure du possible, la PoC devrait fournir ou faciliter la création de modules de construction (Solution Building Blocks) pour l'architecture et la solution.

Plus précisément, nous souhaitons que la PoC fournisse :

- une stratégie de test pour la validation de principe
- la réalisation d'un pipeline CI/CD (même rudimentaire) qui pourra être utilisée pour d'autres projets

Il faudra avoir des accès à

Un repo du code contenant :

- o le code de la PoC versionné;
- o le(s) fichier(s) de configuration de la chaîne de build d'intégration continue ;
- le code des tests implémentés ;
- o un fichier README.

2. Un repo d'architecture contenant :

- o tous les documents du repo fourni (voir ci-dessus);
- un nouveau document de reporting.

Compétences évaluées

- C.14. Concevoir une architecture adéquate, à partir des exigences et attributs de qualité en réalisant des diagrammes d'architecture et en les formalisant dans un support technique à destination de l'équipe de développement afin de faciliter son usage, son adoption, sa robustesse et son évolutivité.
- C.15. Développer les fonctionnalités d'un logiciel existant, en s'appropriant l'historique de la solution à faire évoluer, et prenant en compte les évolutions futures dont elle pourra faire l'objet à long terme dans le cadre d'une reprise par un tiers afin de garantir la pérennité du logiciel et s'assurer de son évolutivité dans le temps.
- C.16. Implémenter un logiciel de qualité, en choisissant des structures de données adaptées et des algorithmes pertinents afin d'assurer la robustesse du logiciel.
- C.17. Tester le logiciel et l'application à plusieurs niveaux en utilisant les méthodologies de test éprouvées afin de garantir la conformité du logiciel au regard des spécifications et la non-régression des fonctionnalités déjà développées.
- C.18. Concevoir une application d'analyse de données massives en intégrant un programme d'apprentissage automatique (machine learning) au développement du logiciel et en utilisant des réseaux de neurones, des algorithmes d'optimisation et de recommandation afin de faire ressortir les tendances utilisateurs.

Principes de l'architecture

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	2
Principe A1 : Primauté des principes	2
Principe A2 : Maximiser les avantages pour l'entreprise	2
Principe A3 : Conformité aux lois et aux règlements	3
Principe A4 : Adhésion au serment d'Hippocrate à tous les niveaux	3
B. Principes de l'architecture informatique (système, données, solutions, sécurité et	
opérations)	4
Principe B1 : Continuité des activités des systèmes critiques pour les patients	4
Principe B2 : Clarté grâce à une séparation fine des préoccupations	5
Principe B3 : Intégration et livraison continues	6
Principe B4 : Tests automatisés précoces, complets et appropriés	6
Principe B5 : Sécurité de type « shift-left »	7
Principe B6 : Possibilité d'extension grâce à des fonctionnalités pilotées par les	
Événements	8
C. Méthodologie architecturale et principes de processus	9
Principe C1 : Personnalisation de l'ADM TOGAF 10	9
Principe C2 : Référentiel d'architecture centralisé et organisé comme source de référence	9
Principe C3: Normes reconnues pour garantir les meilleures pratiques	10
Principe C4 : Favoriser une culture de "learning" avec des preuves de concept et des	
prototypes	11

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 dégraderaient 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 générale 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 sur investissement 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, formations 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 modifications dans nos processus ou applications.

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

Déclaration :

En tant qu'entreprise à visée médicale 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

en découler 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 au principe 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 développement, de leur déploiement 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 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 performance, 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 d'activité 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 transparent, 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 des architectures de microservices. Cela implique un partenariat actif entre les équipes techniques et métiers 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 traitées dans le contexte métier et dans le monde réel. 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 cycles 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 claires 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 :

- unitaire;
- intégration;
- 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 culture de la sécurité d'abord, qui diminue le risque de non-respect des réglementations et de perte de 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 matière 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 10

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 10.

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 standard TOGAF d'OpenGroup fournit un cadre centré sur la gestion des exigences.

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.

Le comité d'architecture du consortium devra collaborer avec les parties prenantes médicales, métiers et techniques pour convenir d'un cadre architectural, qui pourra être modifié selon les projets et les différents contextes métiers.

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 géré en permanence par le comité d'architecture 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 reconnues pour garantir les meilleures pratiques

Déclaration :

L'application de normes reconnues et donc des meilleures pratiques 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. L'utilisation des normes associées à ces pratiques peut permettre de mieux tirer parti des avantages découlant des principes avec lesquels nous nous alignons.

Implications:

Nous encourageons et soutenons, au moins, les normes et les pratiques architecturales listées ci-dessous.

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

- Architectures pilotées par les événements :
 - Source des événements.
- Architectures microservices :
 - 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,
 - o 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 exécutables sur JVM en raison des directives du Consortium;
- Documentation :
 - Devrait favoriser Javadoc pour le code source et Markdown 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 et des prototypes

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 preuves de concepts 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 preuves 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 lorsque ce risque de nuire au patient ou à l'entreprise est présent.
Par exemple, l'apprentissage peut être mené de manière isolée dans un environnement artificiel afin d'éviter l'impact sur les systèmes de production.

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 devraient utiliser des données anonymisées ou factices lorsque cela est possible.

III) 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 secondaires de 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 performance 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.

IV) 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.
- V) 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 ;
 - 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.
- VI) 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 hypothèses en lien avec le principe B3.

Document de définition de l'architecture

Sommaire

Ce document détaille :

- le contexte métier et parties prenantes (page 1);
- les objectifs et contraintes du projet (page 4);
- les principes architecturaux (page 7);
- l'architecture de base (page 8) ;
- l'architecture cible (page 10);
- l'analyse de l'écart (page 14).

Objectif du document

Le document de définition de l'architecture est le livrable définissant les principaux artéfacts architecturaux créés au cours du projet. Il couvre tous les domaines de l'architecture (métier, données, application et technologie) et examine également tous les états pertinents de l'architecture (ligne de base, état(s) intermédiaire(s) et cible).

Public cible

Ce livrable est destiné à toutes les personnes impliquées dans le projet. Il est particulièrement utile au comité d'architecture et aux parties prenantes liées à la réalisation du projet.

Contexte métier et parties prenantes clés

Un consortium de quatre sociétés de premier plan s'est réuni pour consolider les efforts, les données, les applications et les feuilles de route de chacune afin de développer une plateforme de nouvelle génération centrée sur le patient et capable d'améliorer les soins de base proposés, tout en étant réactive, opérationnelle en temps réel et capable de

prendre des décisions dans les situations d'urgence, en prenant en compte l'ensemble des données.

Les organisations suivantes apportent des données spécialisées et une expertise centrée sur le patient dans chacun des domaines ci-dessous.

Organisation	Domaine spécialisé	Motivations métier	Remarques
Ursa Major Health	Planification de rendez-vous pour la médecine générale (GP)	Amélioration des processus de travail dans le cadre de l'intégration manuelle avec d'autres systèmes.	Consolidation des rendez-vous et réduction des problèmes d'intégration avec les systèmes de prise de rendez-vous hospitaliers.
Jupiter Scheduling	Prise de rendez-vous à l'hôpital	Amélioration des processus de travail dans le cadre de l'intégration manuelle avec d'autres systèmes.	Synergie avec d'autres systèmes de prise de rendez-vous et suppression des intégrations manuelles avec les systèmes de planification GP.
Emergency Expert Systems	Recomman- dations de l'hôpital en temps réel en fonction de la disponibilité des lits	Intégration et validation des données requises pour améliorer les décisions.	Réduire le taux de mauvaises recommandations en pondérant les décisions en fonction des antécédents médicaux connus des patients.
Schedule Shack (désormais Schedule Shed)	Liste du personnel médical et planification	Amélioration des processus de travail.	Dépend actuellement de mises à jour manuelles peu fiables des systèmes de prise de rendez-vous lorsque le personnel n'est pas disponible.

Toutes les parties prenantes souhaitent réduire les doublons au niveau des fonctions actuelles, tout en développant de nouvelles fonctionnalités susceptibles d'améliorer les soins dispensés aux patients grâce à une meilleure intégration et un partage des données entre les systèmes.

L'objectif principal du projet est de développer une plateforme de services dont les composants communiquent grâce à des événements. Ces services ont une responsabilité unique. Ils sont également découplés entre eux et tolérants aux pannes. En résumé, ils respectent les principes d'une architecture de microservices distribués.

Objectifs et contraintes du projet

Objectifs

L'objectif final, à savoir améliorer *les soins dispensés aux patients*, sera atteint grâce au consortium en combinant les forces de chacun pour créer les bases d'une plateforme partagée soutenue par des équipes compétentes dans le domaine des soins aux patients.

À son tour, cela soutiendra l'innovation des solutions centrées sur le patient, conformément à la stratégie métier visant à améliorer la qualité globale des soins. Ce socle devrait permettre au consortium de fournir de meilleurs soins aux patients au moyen des activités quotidiennes clés (voir annexe 1).

Contraintes

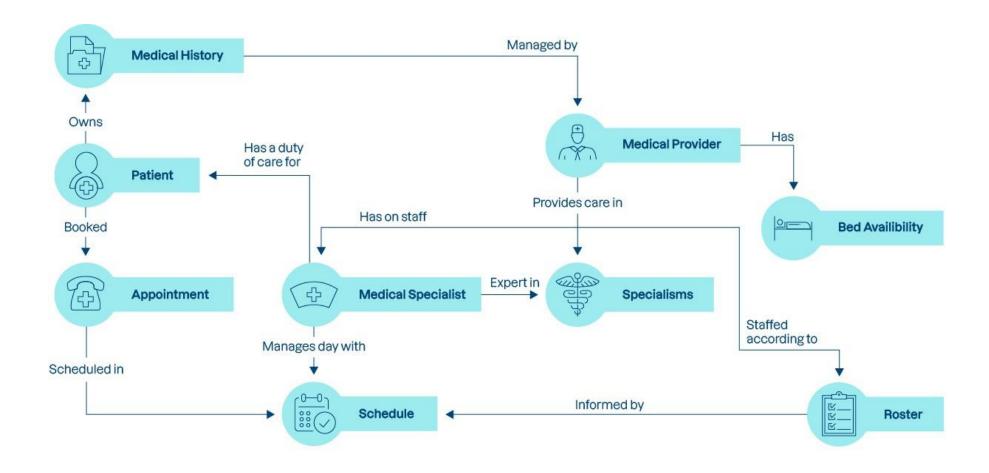
- Les systèmes et processus existants ne doivent pas être *significativement* entravés pendant les phases du projet.
- La feuille de route architecturale et les orientations alimentant les exigences fonctionnelles doivent respecter le <u>Cadre des normes numériques et</u> technologiques du NHS.
- Les données réelles des patients doivent à tout moment rester conformes aux réglementations européennes, notamment le RGPD.
- Les phases initiales du projet devraient viser la création de :
 - modules de construction réutilisables qui respectent les meilleures pratiques convenues;
 - ou de modèles pour des modules de construction futurs qui respectent les meilleures pratiques convenues.

Limites

- Le niveau de conformité peut être assoupli pour les prototypes.
- La confidentialité des données des patients doit être respectée et tous les prototypes non conformes à la production doivent anonymiser les données ou utiliser des données factices.

Limites du domaine

L'orientation architecturale se limite aux solutions, processus et systèmes affectant le modèle de domaine suivant, qui a été choisi par des experts en la matière impliqués dans le projet.



Contextes limités

Informations sur le patient : intervenant principal et identité d'un patient.

Information sur le spécialiste médical : intervenant principal et identité du responsable de la prestation des soins aux patients. Par exemple : un médecin, une infirmière ou un physiothérapeute.

Gestion des prestataires médicaux : organisation chargée de fournir une gamme de soins de santé à un patient, par le biais de médecins spécialistes affiliés.

Gestion des spécialités : capacités médicales fournies par les médecins spécialistes au prestataire.

Planification de la gestion des informations : programmation médicale à court et à long terme des activités quotidiennes et de la disponibilité.

Gestion des rendez-vous : prise de rendez-vous convenus entre le patient et le médecin spécialiste.

Gestion du registre : plan prévisionnel de dotation du personnel dans un proche avenir par un prestataire médical.

Gestion des données sur les antécédents médicaux des patients : données détenues par les patients et gérées par les centres médicaux.

Principes architecturaux

L'architecture doit être construite selon ce qui est énoncé dans le document Principes de l'architecture.

Notez que ces principes sont assouplis pour les validations de principes et les mises en œuvre destinées à l'apprentissage.

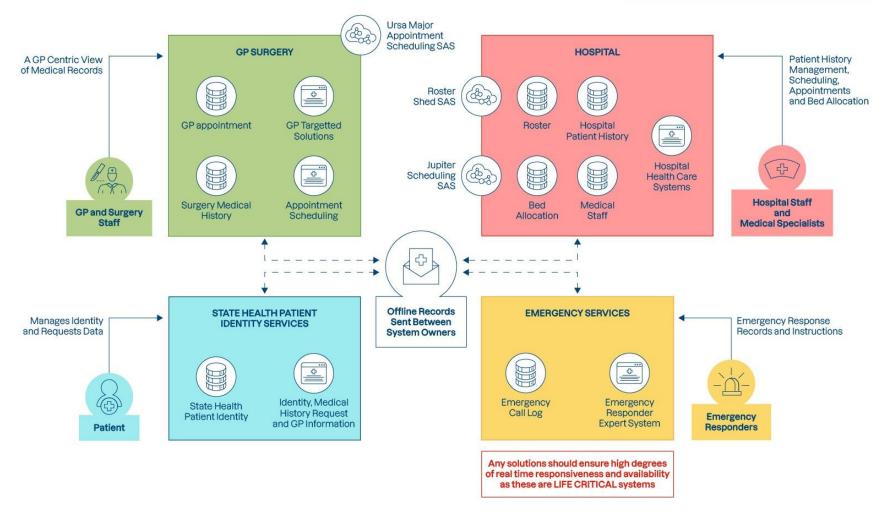
Architecture de base

L'architecture de base se compose de systèmes disparates selon les cas d'utilisation et les contextes médicaux spécifiques :

- Chirurgie (médecine générale);
- Hôpitaux;
- Intervenants d'urgence;
- Systèmes de gestion de l'identité médicale gérés par l'État.

Un aperçu de l'architecture est disponible ci-dessous :





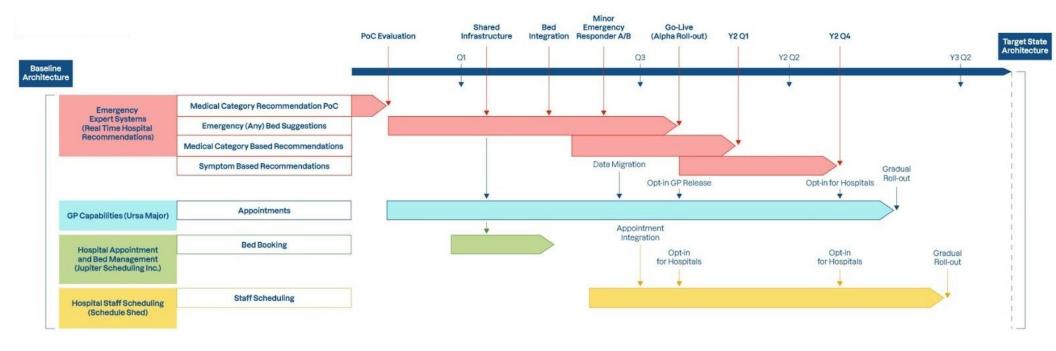
- La séparation des systèmes et le processus manuel important impliqué dans la synchronisation des informations pertinentes entre les systèmes ne parviennent manifestement pas à fournir une assistance en temps réel aux services d'urgence afin de faciliter les échanges entre les hôpitaux pour :
 - attribuer un cas spécifique à l'établissement de soins approprié ;
 - éviter que les équipes soient débordées dans un hôpital donné.
- Dans les cas qui nécessitent plusieurs hôpitaux, le système doit déterminer où les ambulances doivent se rendre, en fonction de la distance, des besoins en soins et de la disponibilité du personnel.

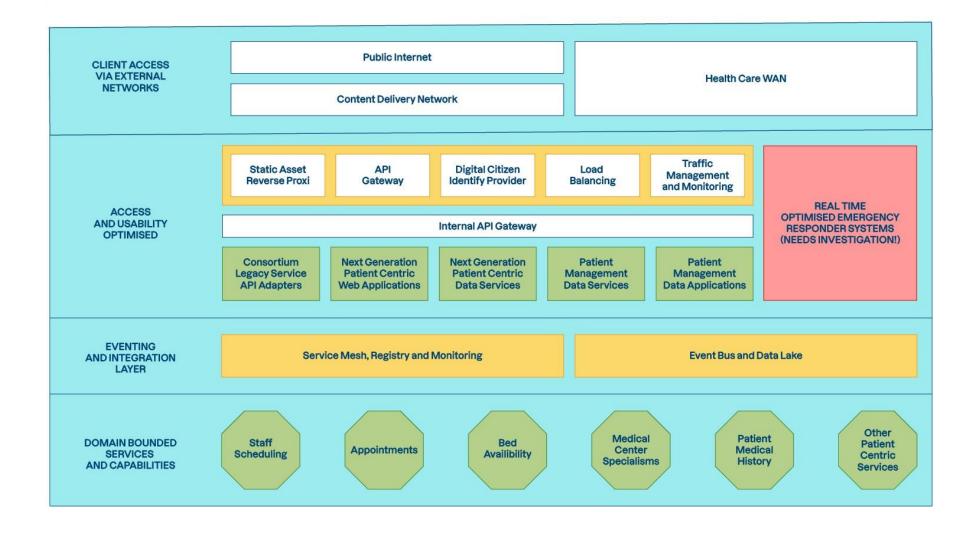
Architecture cible

Ce projet possède une gouvernance architecturale commune au sein du consortium, par le biais d'un groupe de gestion de la plateforme et des systèmes détenus conjointement.

Cette gouvernance, assurée par le comité d'architecture, a pour but de réaliser des économies d'échelle et de respecter une feuille de route stratégique pour la livraison d'une plateforme qui offre aux membres du consortium à la fois une agilité et une mise sur le marché plus rapide, avec des niveaux de sécurité des données et de tolérance aux pannes exigés de la part des institutions du secteur de la santé.

Ce qui suit décrit la feuille de route actuelle du consortium MedHead, partant de l'architecture métier existante pour aboutir à l'architecture cible finale.





L'architecture cible est divisée en quatre niveaux :

Accès client via des réseaux externes

Les clients externes utilisés par les patients et le personnel médical accèdent à la plateforme via cette couche. Ce niveau comprend les routages physiques et logiques pour accéder numériquement à la plateforme MedHead. Des contenus à faible risque et souvent renouvelés peuvent être disponibles via les réseaux de diffusion de contenu (CDN) approuvés par le projet.

Couche optimisée pour l'accès et l'utilisation

- Ce niveau fournit des applications et des services hautement optimisés pour l'accès aux données et leur utilisation par les patients et le personnel médical. Les systèmes devraient être découplés et inclure une tolérance aux pannes en suivant un modèle de microservice avec le pattern Database per service.
- Il contient également les systèmes d'intervention d'urgence optimisés en temps réel : les systèmes hautement optimisés pour des cas d'utilisation en temps réel et des niveaux élevés de tolérance aux pannes seront hébergés dans cette couche. Ce sont des systèmes critiques pour la sécurité des patients. Le projet nécessite une validation anticipée du modèle et des *modules de construction de solution* pouvant être utilisés pour développer des systèmes hautement réactifs.

• Couche d'événements et d'intégration

- Maillage de services : Les fonctions et services métiers de base seront accessibles via une mise en œuvre de type maillage de services dans la couche d'intégration. Cela fournira également l'observabilité par le biais de modèles « side-car proxy » et de découverte des services.
- Bus d'événements et lac de données : Tous les services devront publier les principaux événements métier sur un bus d'événements commun, ce qui entraînera également l'agrégation des événements dans un lac de données à accès optimisé. Le lac de données est destiné à permettre aux applications de reconstruire les magasins de données en fonction de l'historique des applications. Le lac de données est destiné à assister les systèmes en temps réel (entre autres) afin qu'ils fournissent des comportements réactifs au changement.

- Services et capacités limités au domaine
 - Ce niveau contient les services métier de base séparés par des contextes opérationnels et liés au domaine. Les systèmes de cette couche doivent suivre un modèle d'architecture hexagonal/ports et adaptateurs et être construits pour des interfaces et des abstractions qui les rendent faciles à modifier selon les nouvelles tendances et besoins métiers.

Notez que l'objectif de ce projet est de migrer progressivement les capacités existantes et nouvelles des silos des organisations membres vers une architecture cohérente qui appartient conjointement au groupe de gestion de la plateforme et des systèmes et adhère à son architecture, ses principes et ses normes.

Analyse de l'écart

Le tableau ci-dessous compare les fonctions connues de la plateforme DE BASE et l'architecture cible prévue. L'intersection entre une ligne et colonne peut contenir l'une de ces deux valeurs, *consolidé* ou *éliminé*, selon que l'architecture cible renforce ou supprime une fonctionnalité présente dans l'architecture de base. Remarquez que les fonctionnalités supprimées ont été regroupées à droite du tableau dans la colonne "Fonctionnalité éliminée". Une intersection sans indication signifie que la fonctionnalité de l'architecture de base est conservée.

	Fonctionnalités de l'architecture CIBLE	Historique du centre médical	Prise de rendez-vous	Planification du personnel	Attribution des lits	INTERVENTION D'URGENCE fondée sur les données	Fonctionnalité de l'architecture de base éliminée
Fonctionnalités de l'architecture de BASE							
Pré-réservation de l'hôpital local pour intervention d'urgence †							
Pré-réservation d'un hôpital national pour intervention d'urgence ††							Absent au niveau national
Tri selon les symptômes pour intervention d'urgence							Absent
Accès aux données d'historique - médecine générale		Consolidé					
Rendez-vous médecine générale			Consolidé				
Planification des médecins spécialistes de l'hôpital				Consolidé			
Disponibilité des lits d'hôpital					Consolidé		
Gestion des médicaments hospitaliers							Absent

- † Jusqu'à 12 hôpitaux de 2 000 lits chacun
- †† Environ 2 000 hôpitaux au Royaume-Uni et environ 200 000 lits

Lacunes identifiées

Pré-réservation d'un hôpital national pour intervention d'urgence

Bien que le système actuel permette de rechercher des lits d'hôpital au niveau national, dans seulement 0,005 % des cas (source non disponible), les patients sont transportés hors du district considéré comme local pour répondre à l'urgence et prendre en charge l'intervention. Sur cette base, tout système devrait pouvoir inclure des extensions permettant cela (par exemple un modèle de prise en charge), or cela n'est pas requis dans le système initial.

Tri selon les symptômes pour intervention d'urgence

Le système actuel utilise les symptômes et s'en sert pour déterminer la gravité de l'urgence.

Cela est actuellement considéré comme hors-projet et peut continuer à être assuré par les anciens systèmes.

Gestion des médicaments hospitaliers

Le système actuel permet de suivre les stocks de médicaments et de passer des commandes. Cela est considéré comme en dehors du cadre de ce projet par les parties prenantes et peut continuer d'être assuré à l'aide des anciens systèmes.

Risques

La principale préoccupation est actuellement le système d'intervention d'urgence en temps réel, notamment : sa capacité à fournir une allocation de lit en temps opportun, dans le service hospitalier approprié (cf FT)

Ces risques doivent être éliminés à l'aide d'une preuve de concept. Voir ci-dessous.

Annexes

Annexe 1 : Support aux patients

