



PRINCIPES DE L'ARCHITECTURE

Imamou Chaima
Architecte logiciel MedHead+

Table des matières

I.	Principes d'architecture métier.....	2
A.	Principe A2 : Maximiser les avantages pour l'entreprise	2
B.	Principe A3 : Conformité aux lois et aux règlements	3
C.	Principe A4 : Adhésion au serment d'Hippocrate à tous les niveaux.....	3
II.	Principes de l'architecture informatique	3
A.	Principe B1 : Continuité des activités des systèmes critiques pour les.....	3
	Patients.....	3
B.	Principe B2 : Clarté grâce à une séparation fine des préoccupations.....	3
C.	Principe B3 : Intégration et livraison continues	4
D.	Principe B4 : Tests automatisés précoces, complets et appropriés.....	4
E.	Principe B5 : Sécurité de type « shift-left »	4
F.	Principe B6 : Possibilité d'extension grâce à des fonctionnalités.....	4
	déclenchées par les événements	4
III.	Méthodologie architecturale et principes de processus.....	5
A.	Principe C1 : Personnalisation de l'ADM TOGAF 9.2	5
B.	Principe C2 : Référentiel d'architecture centralisé et organisé comme.....	5
	source de référence	5
C.	Principe C3 : Normes ouvertes pour garantir des normes élevées.....	5
D.	Principe C4 : Favoriser une culture de “learning” avec des preuves de concept, des prototypes et des Spike	6

I. Principes d'architecture métier

A. Principe A2 : Maximiser les avantages pour l'entreprise

Nous avons mis en place dans le PoC, les éléments suivants qui vont permettre de maximiser les avantages pour l'entreprise :

Actions /fonctionnalité mise en place	Avantage
<ul style="list-style-type: none">- Architecture micro-service	<ul style="list-style-type: none">- les équipes de développement chargées de cette PoC sont en mesure de l'utiliser comme un jeu de modules de construction pour d'autres modules.- PoC facilement intégrée dans le développement futur- Le partage des informations a permis de gagner en temps sur la redirection des urgences- scalabilité des services
<ul style="list-style-type: none">- Mise en place des API REST	<ul style="list-style-type: none">- Partage d'information entre les services
<ul style="list-style-type: none">- API Gateway	<ul style="list-style-type: none">- Agrégation des requêtes sera faite directement au travers de l'API Gateway, cela permet de réduire la charge réseau et la multiplication des appels.
<ul style="list-style-type: none">- Recherche d'un hôpital le plus proche en fonction de la spécialisation	<ul style="list-style-type: none">- Gain de temps pour les personnels hospitaliers
<ul style="list-style-type: none">- Automatisation des tests	<ul style="list-style-type: none">- L'application va être régulièrement testée de manière automatique pour éviter tout risque que l'activité soit impactée par des erreurs et problèmes

B. Principe A3 : Conformité aux lois et aux règlements

Les données réelles des patients doivent rester conformes aux réglementations européennes, notamment le RGPD. Pour le respect de la confidentialité des données, nous avons utilisé des données factices pour la mise en place du PoC et pour la réalisation de ce projet.

La confidentialité des données des patients va être respectée et les prototypes non-conformes à la production vont anonymiser les données ou utiliser des données factices.

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

Sur le PoC mise en place nous avons respecté le serment d'Hippocrate à savoir, nous pouvons citer quelques exemples :

- Pas de discrimination (sexe, âge, ...) sur le site
- Une réponse à chaque question dans le but de sauver des vies
- Les données personnelles saisies seront protégées et seront inaccessibles aux personnes non habilitées.
- ...

II. Principes de l'architecture informatique

A. Principe B1 : Continuité des activités des systèmes critiques pour les Patients

La mise en place de l'application ne va pas entraver le déroulement de l'activité. Les modules vont être développés indépendamment des services actuels de l'entreprise. Des tests vont être mis en place au moment du déploiement afin d'éviter tout risque de problème qui engendrera l'arrêt de l'activité.

B. Principe B2 : Clarté grâce à une séparation fine des préoccupations

Nous avons mis en place les micro-services afin d'éviter la centralisation des données. Chaque micro-service aura sa propre base de données. De plus la scalabilité des micro-services va permettre d'adapter les services en augmentant la mise à l'échelle s'il y en a besoin.

C. Principe B3 : Intégration et livraison continues

Nous avons mis en place l'intégration et la livraison continue dans le projet pour les tests. Ce dernier va permettre de réduire les risques d'erreurs, de permettre une meilleure qualité de code. En automatisant les tests, on s'assure que les tests sont effectués de manière cohérente, ce qui rend les résultats plus fiables.

Les pipelines CI/CD vont être déclenchés à chaque mise à jour du code. Les exécutions sont liées à une livraison de fonctionnalité donnée. A chaque fois que la fonctionnalité est mise à jour, les tests seront lancés. Des rapports seront générés afin de faciliter les analyses des fonctionnalités (exemple rapport de couverture des tests avec Jacoco).

D. Principe B4 : Tests automatisés précoces, complets et appropriés

Des tests unitaire et d'intégrations ont été mise en place afin de tester les fonctionnalités pour garantir la fiabilité fonctionnelle et non-fonctionnelle. Des rapports pourront être consultés pour plus de précision et d'analyse sur les tests.

E. Principe B5 : Sécurité de type « shift-left »

Dans notre projet, nous avons mis en place l'exécution de tests précoces et fréquents. En automatisant le test du code, les développeurs sont régulièrement informés du moindre problème de sécurité, ce qui leur permet d'y remédier bien avant la mise en production du logiciel.

F. Principe B6 : Possibilité d'extension grâce à des fonctionnalités déclenchées par les événements

Les services mises en place vont pouvoir s'échanger des informations entre eux. En effet, les API des différents services vont pouvoir communiquer et déclencher des événements en cas de besoin.

A l'exemple, le service hôpital (qui regroupe les hôpitaux et leurs spécialisations) et le service de réservation de lit communiquent entre elles. Une fois qu'un hôpital a été sollicité et qu'une chambre est disponible, un événement est déclenché pour aller réserver un lit et qui ce dernier va ensuite déclencher un événement pour aller réduire le nombre de lits dans l'hôpital.

III. Méthodologie architecturale et principes de processus

A. Principe C1 : Personnalisation de l'ADM TOGAF 9.2

Durant tout le long du projet, nous nous sommes basés sur l'architecture TOGAF. La méthode TOGAF est devenue ces dernières années un cadre de référence pour l'architecture d'entreprise.

En effet, TOGAF ne prescrit pas les modèles à utiliser pour représenter l'architecture, mais guide le processus lors de la création de l'architecture. Il permet de :

- Apporter une vision d'ensemble
- Permettre ainsi d'avoir un fil directeur et les pratiques sont harmonisées
- Permettre de rendre beaucoup plus simple la collaboration entre services métiers et techniques

C'est un recueil de bonnes pratiques qui apporte un cadre complet pour l'architecture de l'entreprise.

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

Nous avons mis, en place un service d'enregistrement. Ce service permet d'enregistrer la localisation de chaque micro-service. C'est un annuaire qui publie l'ensemble des informations des micro-services (nom, adresse IP, numéro de port). Ce service permet à chaque micro-service de se connecter au *service d'enregistrement* pour publier son nom, son adresse IP et son numéro de port.

En cas de besoin par un client, le répertoire est en mesure de donner l'emplacement du micro-service ainsi permettre les échanges de données.

C. Principe C3 : Normes ouvertes pour garantir des normes élevées

Pour garantir l'organisation et respecter au mieux les normes, nous avons utilisé une architecture et des outils comme :

- Swagger pour la documentation des API Rest
- Utilisation du principe TOGAF
- Utilisation du modèle Agile
- L'architecture micro-service
- Les rapports de couverture des tests
- La mise en place des tests CI/CD

D. Principe C4 : Favoriser une culture de “learning” avec des preuves de concept, des prototypes et des Spike

Pour diminuer les risques, respecter les normes et garantir l’atteinte aux objectifs fixés, nous avons mis en place et fourni :

- Une validation des hypothèses du PoC
- Le PoC a été isolé des données réelles des patients (confidentialité des données)
- Le respect de la confidentialité des données des clients en utilisant des données factices pour la mise en place du projet
- plans de test comme outils de communication des exigences