



SOLUTION BUILDING BLOCKS

**Projet : PoC pour le sous-système
d'intervention d'urgence en temps réel**

Client : MedHead

HISTORIQUE DES RÉVISIONS

N° de version	Description	Date	Auteur
1.0	Livraison initiale	15 janvier 2022	Nicolas Bertrand

TABLE DES MATIÈRES

Historique des révisions	2
Table des matières	3
Objectif du présent document	4
Blocs de construction de la preuve de concept	4
Module de la couche front	4
Modules de la couche service	5
Module de construction d'un pipeline CI	7

OBJECTIF DU PRÉSENT DOCUMENT

Ce document contient une description de tous les éléments de base réutilisables, obtenus lors de la conception et de la mise en œuvre de la preuve de concept pour le sous système d'intervention d'urgence souhaité par MedHead. Nous décrivons, pour chaque composant, les technologies et outils implémentés. En premier lieu, nous présentons les blocs de construction de la couche front puis de la couche service et terminons avec le pipeline d'intégration continue. L'élaboration du plan et du contenu de ce document est librement inspirée des recommandations du framework *TOGAF* (The Open Group Architecture Framework) et de son cycle *ADM* (Architecture Development Method), plus précisément du chapitre intitulé Building Blocks.

BLOCS DE CONSTRUCTION DE LA PREUVE DE CONCEPT

MODULE DE LA COUCHE FRONT

Nom de la brique : Front end

Fonctionnalité fournie : Une application web fournissant un formulaire de réservation de lit en urgence suivant trois critères (lieu, spécialité et id du patient).

Lien vers un exemple d'implémentation ou d'interfaces : L'interface utilisateur a été implémentée avec la librairie javascript React et un thème Bootstrap fourni par Bootswatch. Les liens suivants ont été utilisés afin d'implémenter la solution :

- [React](#)
- [Bootstrap](#)
- [Bootswatch](#)

Travaux à effectuer pour compléter ce module : Ajouter les composants hospital et patient à l'interface.

Alignement architectural : Cette brique permet ou reflète les objectifs et principes suivants :

Objectif 1 : Gagner la confiance des utilisateurs quant à la simplicité d'un tel système

Principe B2 : Clarté grâce à une séparation fine des préoccupations

Principe B4 : Tests automatisés précoces, complets et appropriés (E2E)

MODULES DE LA COUCHE SERVICE

Nom de la brique : Back end

Fonctionnalité fournie : L'application est structurée autour de trois microservices centraux (emergency, hospital et patient), responsables de la logique métier, et deux edge microservices (gateway et registry) pour l'orchestration.

Lien vers un exemple d'implémentation ou d'interfaces : L'architecture microservices est retenue pour ce projet. Les liens suivants ont été utilisés afin d'implémenter la solution :

- [Microservices](#)
- [Spring initializr](#)
- [Spring Cloud](#)

Travaux à effectuer pour compléter ce module : Ce module doit être complété avec l'implémentation ou le paramétrage d'autres edge microservices, comme par exemple config server, circuit breaker et load balancer ainsi que l'utilisation du protocole WebSocket pour l'utilisation d'un moyen de communication bi-directionnel.

Alignement architectural : Cette brique permet ou reflète les objectifs et principes suivants :

Objectif 1 : Améliorer la qualité des traitements d'urgence et sauver plus de vies

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

Nom de la brique : Emergency

Fonctionnalité fournie : Ce microservice permet de réserver un lit d'hôpital.

Lien vers un exemple d'implémentation ou d'interfaces : Un système de communications non-bloquantes est retenu pour ce module. Les liens suivants ont été utilisés afin d'implémenter la solution :

- [Spring Webflux](#)
- [Spring WebClient](#)
- [Reactor](#)

Travaux à effectuer pour compléter ce module : Ce module doit être complété avec l'implémentation de messages d'erreur personnalisés, de tests d'intégration automatisés et une prise en charge de la géolocalisation.

Alignement architectural : Cette brique permet ou reflète les objectifs et principes suivants :

Objectif 1 : Fournir une API RESTful qui tient les intervenants médicaux informés en temps réel

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

Nom de la brique : Hospital

Fonctionnalité fournie : Ce microservice gère les données relatives aux hôpitaux.

Lien vers un exemple d'implémentation ou d'interfaces : Un système de communications non-bloquantes est retenu pour ce module. Les liens suivants ont été utilisés afin d'implémenter la solution :

- [Spring Webflux](#)
- [Reactor](#)

Travaux à effectuer pour compléter ce module : Ce module doit être complété avec l'implémentation de messages d'erreur personnalisés et de tests d'intégration.

Alignement architectural : Cette brique permet ou reflète les objectifs et principes suivants :

Objectif 1 : Fournir une API RESTful qui tient les intervenants médicaux informés en temps réel

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

Nom de la brique : Patient

Fonctionnalité fournie : Ce microservice gère les données relatives aux patients.

Lien vers un exemple d'implémentation ou d'interfaces : Un système de communications non-bloquantes est retenu pour ce module. Les liens suivants ont été utilisés afin d'implémenter la solution :

- [Spring Webflux](#)
- [Reactor](#)

Travaux à effectuer pour compléter ce module : Ce module doit être complété avec l'implémentation de messages d'erreur personnalisés et de tests d'intégration.

Alignement architectural : Cette brique permet ou reflète les objectifs et principes suivants :

Objectif 1 : Fournir une API RESTful qui tient les intervenants médicaux informés en temps réel

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

Nom de la brique : Gateway

Fonctionnalité fournie : Ce microservice fournit un point d'entrée unique pour l'application afin d'accéder aux APIs.

Lien vers un exemple d'implémentation ou d'interfaces : Un système de communications non-bloquantes est retenu pour ce module. Les liens suivants ont été utilisés afin d'implémenter la solution :

- [Spring Webflux](#)
- [Reactor](#)
- [Spring Cloud Gateway](#)

Travaux à effectuer pour compléter ce module : Ce module doit être complété avec l'implémentation de messages d'erreur personnalisés et de tests d'intégration.

Alignement architectural : Cette brique permet ou reflète les objectifs et principes suivants :

Objectif 1 : Fournir une API RESTful qui tient les intervenants médicaux informés en temps réel

Principe B2 : Clarté grâce à une séparation fine des préoccupations

Principe C3 : Normes ouvertes convenues pour garantir des normes élevées

Nom de la brique : Registry

Fonctionnalité fournie : Ce microservice est un serveur de découverte des services. Les microservices s'enregistrent et découvrent les ports et adresses ip des autres microservices de la solution.

Lien vers un exemple d'implémentation ou d'interfaces : Les liens suivants ont été utilisés afin d'implémenter la solution :

- [Eureka Server](#)

Alignement architectural : Cette brique permet ou reflète les objectifs et principes suivants :

Objectif 1 : Fournir une API RESTful qui tient les intervenants médicaux informés en temps réel

Principe B2 : Clarté grâce à une séparation fine des préoccupations

Principe C3 : Normes ouvertes convenues pour garantir des normes élevées

MODULE DE CONSTRUCTION D'UN PIPELINE CI

Nom de la brique : Intégration continue

Fonctionnalité fournie : Solution d'intégration continue découpée en six étapes (build, tests unitaires, test coverage, code quality, package, container).

Lien vers un exemple d'implémentation ou d'interfaces : Les liens suivants ont été utilisés afin d'appréhender l'approche *DevOps* :

- [DevOps](#)
- [Manifeste Agile](#)
- [Test Driven Développement](#)
- [Docker](#)
- [GitLab](#)
- [Maven](#)
- [Jacoco](#)
- [Code Climate](#)
- [Sonar Java](#)

Travaux à effectuer pour compléter ce module : Le pipeline CI peut être amélioré en incluant l'exécution automatique des tests d'intégration des composants et du système.

Alignement architectural : Cette brique permet ou reflète les objectifs et principes commerciaux suivants :

Objectif 1 : Fournir un pipeline d'intégration continue

Principe B3 : Intégration continue