



Solution Building Blocks

Les blocs de construction (Building Block) représentent des composantes (potentiellement réutilisables) de la capacité métier, informatique ou architecturale qui peut être combinée avec d'autres blocs de construction pour fournir des architectures et des solutions.

Project Name:	ERS, Emergency Responder System (allocation de lits d'hôpital pour les urgences)		
Prepared By:	Damien Senechal	Document Version No:	0.1
Title:	Solution Building Blocks	Document Version Date:	
Reviewed By:		Review Date:	25/07/22

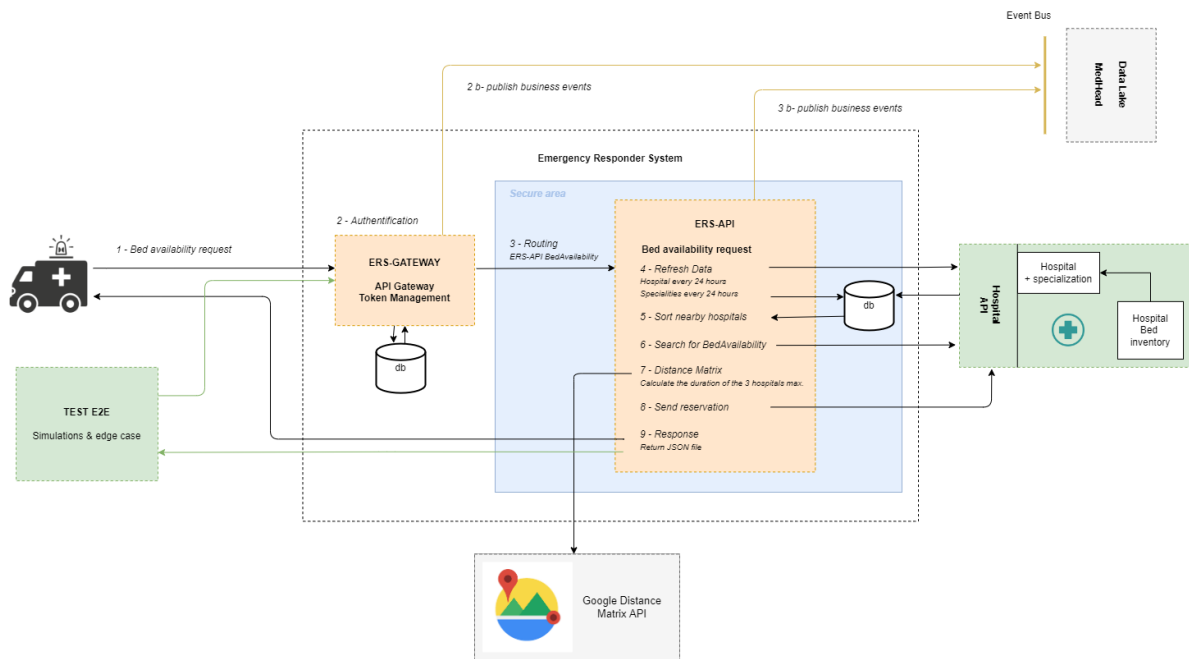
Table des matières

Table des matières	2
Vision globale de l'architecture cible	3
ERS Gateway avec authentification	4
ERS API logique métier	5
FAKE API HOSPITAL	7
TEST End to End	8

Documents liés :

- [Statement of Architecture Work](#)
- [Définition de l'architecture](#)
- [Registre des risques](#)
- [Stratégie de test](#)
- [Hypothèse de validation de principe](#)

Vision globale de l'architecture cible



Le projet ERS (Emergency Responder System), l'allocation de lits d'hôpital pour les urgences, se base sur une API RESTful proposant les solutions suivantes (les blocs de construction sont représentés en orange et en vert sur ce diagramme) :

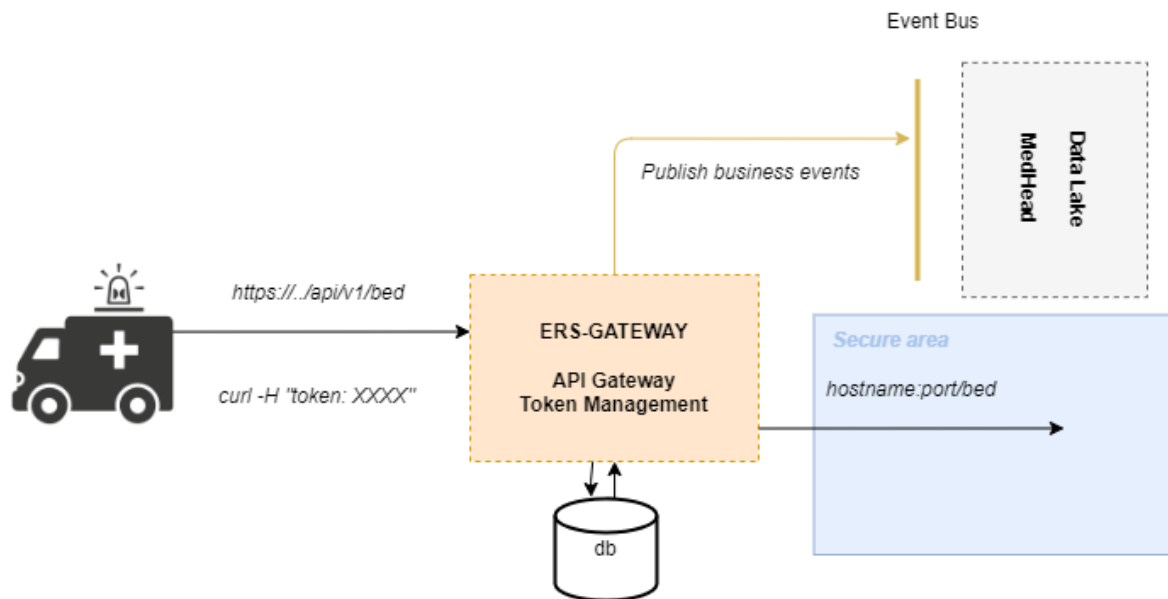
- ERS Gateway avec authentification
- ERS API logique métier
- FAKE API HOSPITAL
- TEST End to End

Nous devons envoyer tous nos événements métier sur un bus d'événement, pour ainsi remplir le lac de données de MedHead. Seulement *ERS Gateway avec authentification* et *ERS API logique métier* (blocs en orange) doivent publier leurs événements. Le détails des données à publier est décrit dans les sections suivantes.

Toutes les gestion de bases de données doivent être faites avec une bibliothèque s'occupant du suivi, de la gestion et de l'application des changements de schéma de base de données et cela indépendamment du SGBD.

Nous utiliserons la bibliothèque [LiquiBase](#).

ERS Gateway avec authentification



Nom du module de construction : ERS-GATEWAY

Fonctionnalités fournies : Système d'authentification de la plateforme. Celle-ci s'assure que toutes les requêtes soient accompagnées d'un token valide et les routent vers les services enregistrés (applications s'exécutants dans la zone sécurisée).

Lien vers des exemples d'implémentation ou d'interfaces : N/A

Travail supplémentaire pour terminer ce module de construction :

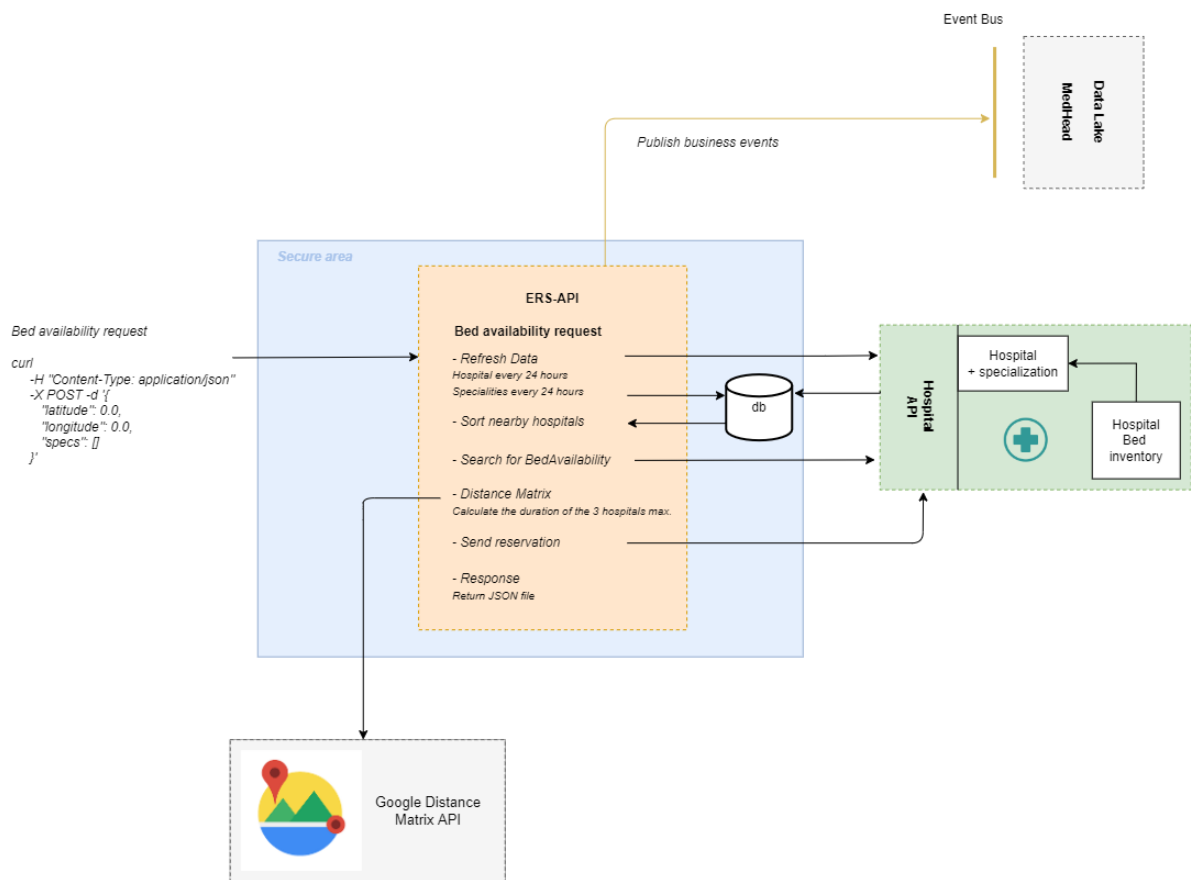
- Une base de données doit accompagner ce module avec une table contenant :
 - token (string)
 - routes valides (array)
- A chaque connexion d'un utilisateur, un événement sera publié sur le bus d'événement avec :
 - la date et heure
 - l'ip de l'expéditeur
 - la route utilisé
 - l'id du token utilisé

Alignement architectural :

Ce SBB permet l'atteinte des objectifs et principes suivants :

- Se connecter au service
- Isoler la logique métier derrière une API Gateway

ERS API logique métier



Nom du module de construction : ERS-API

Fonctionnalités fournies : Fonction principale du projet (CF. document [Hypothèse de validation de principe](#)). Le sous-système d'intervention d'urgence en temps réel est destiné à recevoir une ou plusieurs spécialités médicales (voir les [Données de référence sur les spécialités](#)) et une banque de données d'informations récentes sur les hôpitaux afin de suggérer l'hôpital le plus proche offrant un lit disponible, associé à une ou plusieurs spécialisations correspondantes. Le lieu de l'incident d'urgence doit également être fourni.

Lien vers des exemples d'implémentation ou d'interfaces : N/A

Travail supplémentaire pour terminer ce module de construction :

- Mettre dans une base de données et tenir à jour la liste des spécificités NHS
 - id
 - groupe (ex. 'Anesthésie')
 - spécialité (ex. 'Soins intensifs')
- Mettre dans une base de données et tenir à jour la liste des hopitaux et connaitre leurs positions géographiques
 - id

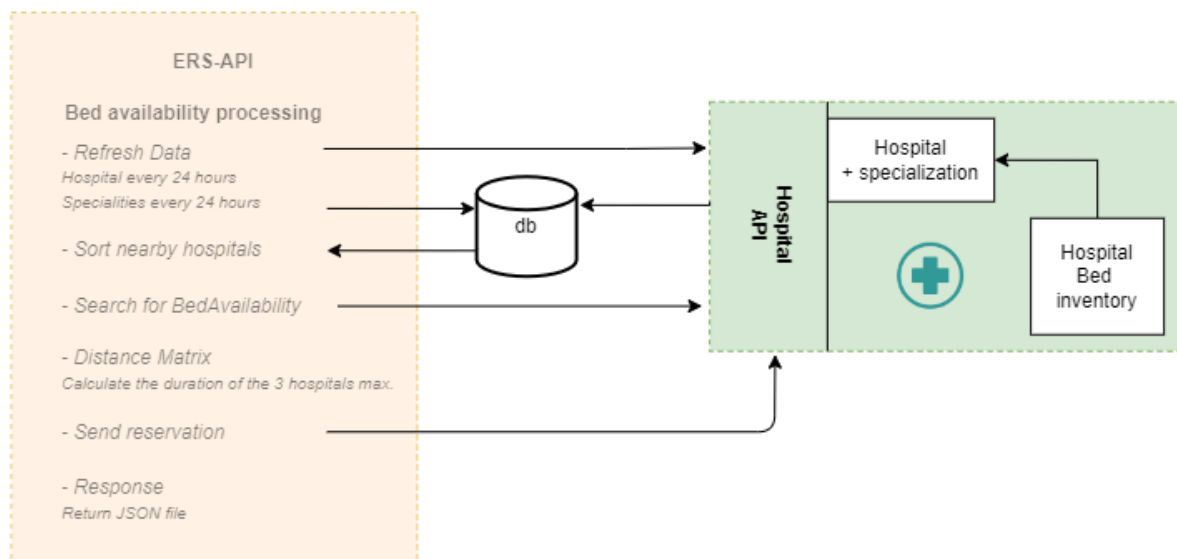
- Nom de l'hôpital
 - latitude / longitude
 - spécialités (array(id spécialité))
- Extraire les hôpitaux par spécificités NHS
- Etre capable de calculer la distance entre le lieu de la demande et la position des hôpitaux (à vol d'oiseau) pour obtenir les résultat les plus pertinents
- Envoyer une demande de disponibilité de lit à l'API HOSPITAL
- Exécution d'une API externe pour calculer la distance et le temps de route
- Envoyer une confirmation de réservation de lit à l'API HOSPITAL
- A chaque confirmation de réservation, un événement sera publié sur le bus d'événement avec :
 - la date et heure
 - l'ip de l'expéditeur
 - le lieu de l'incident
 - les spécialités médicales requises
 - l'hôpital retenu
 - les infos d'id de la réponse de confirmation de réservation (afin de traçabilité et suivi)

Alignement architectural :

Ce SBB permet l'atteinte des objectifs et principes suivants :

- Attribuer un cas spécifique à l'établissement de soins approprié.
- Déterminer où les ambulances doivent se rendre, en fonction de la distance, des besoins en soins et de la disponibilité du personnel.

FAKE API HOSPITAL



Nom du module de construction : FAKE API HOSPITAL.

Fonctionnalité fournie : Pour palier aux contraintes du projet (voir document [Définition de l'architecture](#)) à savoir : *Les systèmes et processus existants ne doivent pas être significativement entravés pendant les phases du projet ET Les données des patients doivent être anonymisées ou nous devons utiliser des données factices.* Nous devons mettre en place un environnement de développement fiable et sécurisé en disposant d'un clone (très basique) de cette API HOSPITAL, qui réponde de la même façon que le ferait l'original afin de pouvoir lancer des tests sans impacter quoique ce soit.

Lien vers des exemples d'implémentation ou d'interfaces : N/A

Travail supplémentaire pour terminer ce module de construction :

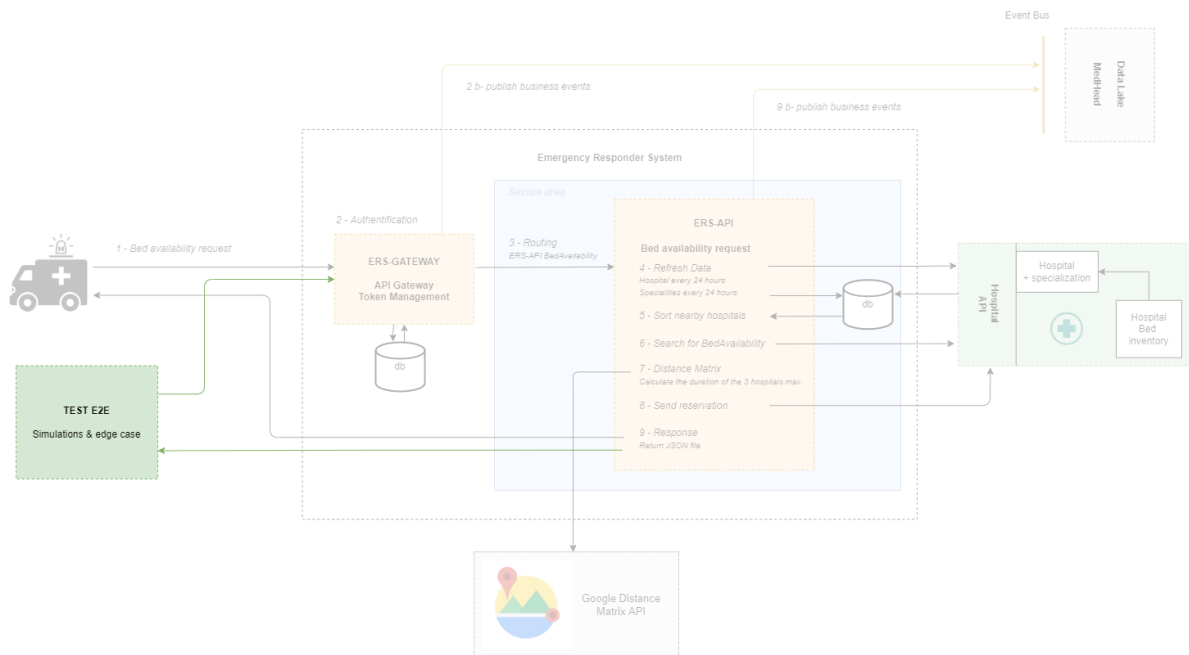
- Une correspondance entre la réponse de l'API original et celle du FAKE doit être mis en place
- le but est de fournir des requêtes afin de :
 - lister les spécificités NHS
 - lister les hôpitaux
 - obtenir la quantité de lit disponible pour un hôpital donné
 - réserver un lit dans un hôpital donné

Alignement architectural :

Ce SBB permet l'atteinte des objectifs et principes suivants :

- Cautionne les contraintes du projet
- rend l'environnement de développement robuste

TEST End to End



Nom du module de construction : TEST E2E

Fonctionnalité fournie : Les tests de bouts en bouts sont les tests très simples à mettre en œuvre mais qui donnent fiabilité et robustesse à nos applications. Ce module n'est pas un réel projet, écrit en java pour la concordance avec le code de l'application principale, il devra uniquement être capable de compiler et de passer les tests j-unit. Il devra toutefois exécuter le test principal et se faire passer pour un utilisateur lambda qui se connecte à la gateway pour effectuer une demande de disponibilité de lit.

Lien vers des exemples d'implémentation ou d'interfaces : N/A

Travail supplémentaire pour terminer ce module de construction :

- Mise en place d'un projet opérant dans notre environnement de développement
- dans la phase de build, exécuter une requête de demande de dispo de lit en passant par la gateway en livrant un token valide. Le résultat doit être celui attendu (en fonction du modèle de données)

Alignement architectural :

Ce SBB permet l'atteinte des objectifs et principes suivants :

- Le respect des meilleures pratiques
- Atténue les risques par un Test de performance précoce d'une preuve de concept représentative