

RAPPORT DE RESULTAT DU POC

Bilan des résultats obtenus et recommandations



Système d'allocations de lits d'urgence
Phase G - Gouvernance de la mise en œuvre

Historique

<i>Auteur</i>	<i>Remarques</i>	<i>Date</i>	<i>n° version</i>
Louis ZEPHIR	Création du documents, début de rédaction	01/12/2025	1.0
Louis ZEPHIR	Mise à jour, correction orthographe	03/01/2026	1.1

Auteurs

<i>Auteur</i>	<i>Fonction</i>	<i>Contact</i>
Louis ZEPHIR	Architecte Logiciel	louis.zephir@medhead.fr

Table des matières

Synthèse exécutive.....	3
Résultats clés.....	3
Verdict global	3
Rappel du contexte et des objectifs.....	4
Contexte.....	4
Hypothèses à valider.....	4
Périmètre de la PoC.....	4
Résultats par objectif.....	5
Analyse des résultats.....	5
Indicateurs clés de performance (KPIs).....	6
Fonctionnalités livrées	7
Métriques du projet.....	8
Répartition de l'effort	8
Apprentissages et retours d'expérience.....	9
Hypothèses validées	9
Points à améliorer	9
Découvertes	9
Écarts et points d'amélioration.....	10
Recommandations.....	11
Recommandations prioritaires	11
Recommandations générales	11
Conclusion et prochaines étapes.....	12
Conclusion	12
Prochaines étapes	12
Décision demandée	12
Approbations	12

Synthèse exécutive

Ce rapport présente les résultats de la Preuve de Concept (PoC) du système d'allocation de lits d'urgence développé dans le cadre du programme MedHead.

La PoC avait pour objectif de valider la faisabilité technique et de réduire les risques avant le développement du système complet.

Résultats clés

6/8 Objectifs atteints	79% Conformité architecturale	450ms Temps de réponse P95	82% Couverture de tests
----------------------------------	---	--------------------------------------	-----------------------------------

Verdict global

SUCCÈS : La PoC valide les hypothèses techniques critiques et démontre la viabilité de l'architecture proposée.

Les objectifs principaux ont été atteints : le système répond en temps réel (< 1 seconde), l'API d'allocation est fonctionnelle, et le pipeline CI/CD est opérationnel.

Les écarts identifiés (résilience, sécurité) sont documentés et planifiés dans la feuille de route.

Rappel du contexte et des objectifs

Contexte

Le consortium MedHead regroupe quatre organisations du secteur de la santé souhaitant consolider leurs systèmes pour améliorer les soins aux patients.

La composante critique identifiée est le système d'intervention d'urgence en temps réel, capable d'allouer un lit d'hôpital dans la spécialité appropriée, à l'hôpital le plus proche, en moins d'une seconde.

Hypothèses à valider

H1 : Une architecture microservices Spring Boot peut répondre aux exigences de performance temps réel.

H2 : Le modèle de données proposé (Hospital, Specialty, Bed) est adapté aux cas d'usage d'allocation.

H3 : L'algorithme de recherche géographique permet de sélectionner l'hôpital optimal.

H4 : Un pipeline CI/CD peut être mis en place pour assurer la qualité et la livraison continue.

Périmètre de la PoC

Inclus : API REST d'allocation, gestion des hôpitaux/spécialités/lits, calcul géographique, tests automatisés, pipeline CI/CD.

Exclus : Interface utilisateur, intégration systèmes externes réels, architecture événementielle, authentification OAuth2.

Résultats par objectif

Le tableau ci-dessous présente chaque objectif de la PoC avec la cible définie, le résultat obtenu et le statut.

Objectif	Cible	Résultat	Statut
Valider la capacité à répondre en temps réel (< 1 seconde)	< 1 000 ms	450 ms (P95)	Atteint
Implémenter l'API d'allocation de lits	API REST fonctionnelle	6 endpoints opérationnels	Atteint
Intégrer le référentiel NHS des spécialités	57 spécialités	57 spécialités chargées	Atteint
Mettre en place un pipeline CI/CD	Pipeline automatisé	Pipeline 8 étapes	Atteint
Atteindre une couverture de tests > 80%	> 80%	82%	Atteint
Documenter l'architecture et les API	Documentation complète	Swagger + Javadoc	Atteint
Implémenter les patterns de résilience	Circuit breakers	Non implémenté	Non atteint
Sécuriser l'API avec OAuth2/JWT	Authentification OAuth2	Authentification basique	Partiel

Analyse des résultats

- **6 objectifs atteints (75%)** : Les fonctionnalités cœur sont opérationnelles et performantes.
- **1 objectif partiellement atteint (12,5%)** : La sécurité est présente mais non conforme aux standards NHS.
- **1 objectif non atteint (12,5%)** : Les patterns de résilience n'ont pas été implémentés (décision de report).

Indicateurs clés de performance (KPIs)

Les KPIs suivants ont été mesurés tout au long du développement de la PoC.

Catégorie	KPI	Valeur	Cible	
Performance	Temps de réponse API (P50)	120 ms	< 500 ms	✓
Performance	Temps de réponse API (P95)	450 ms	< 1 000 ms	✓
Performance	Temps de réponse API (P99)	780 ms	< 1 000 ms	✓
Performance	Throughput maximal testé	250 req/s	100 req/s	✓
Qualité	Couverture de code globale	82%	> 80%	✓
Qualité	Tests unitaires passants	100%	100%	✓
Qualité	Tests d'intégration passants	100%	100%	✓
Qualité	Scénarios BDD passants	100%	100%	✓
Qualité	Bugs critiques ouverts	0	0	✓
Qualité	Dette technique (SonarQube)	2h	< 8h	✓
CI/CD	Durée du pipeline	12 min	< 15 min	✓
CI/CD	Taux de succès des builds	94%	> 90%	✓
Conformité	Principes architecturaux respectés	11/14	> 10/14	✓
Conformité	Vulnérabilités critiques	0	0	✓

Résultat : 14/14 KPIs atteignent leurs cibles (100%). Les performances mesurées dépassent les attentes, notamment sur le temps de réponse et le throughput.

Fonctionnalités livrées

La PoC a livré les modules fonctionnels suivants, tous testés et documentés.

API Hôpitaux

- ✓ Création d'un hôpital avec coordonnées GPS
- ✓ Consultation d'un hôpital par ID
- ✓ Liste des hôpitaux avec pagination
- ✓ Mise à jour des informations d'un hôpital
- ✓ Suppression d'un hôpital
- ✓ Recherche d'hôpitaux par ville

API Spécialités

- ✓ Liste des 57 spécialités NHS
- ✓ Consultation d'une spécialité par code
- ✓ Recherche par groupe de spécialité
- ✓ Association hôpital-spécialité

API Lits

- ✓ Gestion du parc de lits par hôpital
- ✓ Mise à jour du statut d'un lit
- ✓ Comptage des lits disponibles par spécialité

API Allocation

- ✓ Allocation de lit par spécialité
- ✓ Sélection de l'hôpital le plus proche (calcul géographique)
- ✓ Réservation atomique avec mise à jour du stock
- ✓ Gestion des cas sans lit disponible

Infrastructure

- ✓ Base de données PostgreSQL avec schéma JPA
- ✓ API REST documentée (OpenAPI/Swagger)
- ✓ Health checks et métriques Actuator
- ✓ Conteneurisation Docker

Métriques du projet

Les métriques suivantes caractérisent l'effort de développement de la PoC.

Durée du projet	8 semaines	Effort total	320 heures-homme
Lignes de code	4 500 lignes	Lignes de tests	3 200 lignes
Commits	156 commits	Pull Requests	34 PR mergées

Répartition de l'effort

- **Développement** : 45% (144h) - Code applicatif et API
- **Tests** : 25% (80h) - Tests unitaires, intégration, BDD
- **Infrastructure** : 15% (48h) - CI/CD, Docker, configuration
- **Documentation** : 15% (48h) - Architecture, API, rapports

Apprentissages et retours d'expérience

Hypothèses validées

- ✓ **Architecture microservices viable** : L'approche microservices avec Spring Boot permet d'atteindre les performances requises pour le temps réel.
- ✓ **Modèle de données adapté** : Le schéma JPA (Hospital, Specialty, Bed) répond aux besoins fonctionnels et permet des requêtes performantes.
- ✓ **Calcul géographique efficace** : L'algorithme de distance Haversine est suffisant pour la sélection de l'hôpital le plus proche.
- ✓ **Pipeline CI/CD reproductible** : Le pipeline Jenkins/GitHub Actions peut servir de modèle pour les autres services du consortium.

Points à améliorer

- ⚠ **Résilience insuffisante** : L'absence de circuit breakers expose le système aux défaillances en cascade. Priorité haute pour la Phase 1.
- ⚠ **Sécurité à renforcer** : L'authentification basique ne répond pas aux exigences NHS. Migration OAuth2/JWT nécessaire.
- ⚠ **Observabilité limitée** : Les logs applicatifs seuls ne permettent pas un diagnostic efficace en production.

Découvertes

- 💡 **Complexité de l'intégration NHS** : L'intégration avec NHS Identity sera plus complexe qu'anticipé. Prévoir des ressources supplémentaires.

Écarts et points d'amélioration

Les écarts suivants ont été identifiés entre les attentes initiales et les résultats obtenus.

Domaine	Écart identifié	Impact	Plan d'action
Résilience	Pas de circuit breakers	Élevé	Phase 1 - Resilience4j
Sécurité	Auth basique vs OAuth2	Critique	Phase 1 - OAuth2/JWT
Observabilité	Pas de monitoring centralisé	Moyen	Phase 3 - Prometheus/Grafana
Architecture	Pas de bus d'événements	Moyen	Phase 2 - Apache Kafka

Recommandations

Recommandations prioritaires

1. **Implémenter immédiatement la sécurité OAuth2/JWT** avant toute exposition externe du système.
2. **Ajouter les patterns de résilience (Resilience4j)** pour éviter les défaillances en cascade.
3. **Déployer la stack d'observabilité** pour permettre le diagnostic en production.
4. **Planifier les tests de charge à grande échelle** pour valider la scalabilité nationale.

Recommandations générales

- **Conserver l'équipe PoC** pour assurer la continuité des connaissances vers la Phase 1.
- **Utiliser la PoC comme référence** pour les autres microservices du consortium.
- **Documenter les décisions architecturales (ADR)** pour tracer les choix techniques.
- **Engager tôt les discussions avec NHS Identity** pour anticiper la complexité d'intégration.

Conclusion et prochaines étapes

Conclusion

La Preuve de Concept du système d'allocation de lits d'urgence MedHead est un succès. Les hypothèses techniques critiques ont été validées, notamment la capacité à répondre en temps réel et la viabilité du modèle de données.

Le taux d'atteinte des objectifs (75%) et la conformité architecturale (79%) démontrent la maturité de la solution proposée.

Les écarts identifiés (résilience, sécurité, observabilité) étaient anticipés et documentés dans les principes architecturaux. Ils font partie intégrante de la feuille de route et seront adressés dans les phases suivantes.

Prochaines étapes

Q1 2026 - Phase 1 : Consolidation et sécurisation (OAuth2, Resilience4j, réplication DB)

Q2 2026 - Phase 2 : Architecture événementielle (Apache Kafka)

Q2-Q3 2026 - Phase 3 : Observabilité et performance (Prometheus/Grafana, Jaeger)

Q3 2026 - Phase 4 : Pilote régional (5-10 hôpitaux)

Q4 2026 - Q1 2027 - Phase 5 : Déploiement national

Décision demandée

Le comité d'architecture est invité à valider les résultats de la PoC et à approuver le lancement de la Phase 1 conformément à la feuille de route établie.

Approbations

Rôle	Nom	Date	Signature
Sponsor	Kara Trace		
Architecte principal	Chris Pike		
Chef de projet			