

# ÉMERGENCE Déploiement Cloud Run (MAJ v4)

## RÉSUMÉ DÉPLOIEMENT CLOUD RUN ÉMERGENCE (MAJ v4)

Date: 18/08/2025 04:19 UTC

### AVERTISSEMENT ARBO-LOCK

- Référence architecture: arborescence\_synchronisée\_20250817.txt (dernière version validée).
- Création annoncée et validée : src/backend/features/dev\_auth/router.py.
- Aucune autre création/déplacement/suppression dans ce livrable.

## 1) CONTEXTE & CIBLE

- Projet: emergence-469005 | Service: emergence-app | Région: europe-west1
- Accès via HTTPS Load Balancer: <https://emergence-app.ch>
- Révisions impliquées: emergence-app-00040-rqc puis emergence-app-00041-6wl
- Sécurité: /api/health public ; endpoints métier (/api/documents, /api/debate, /api/memory, /api/chat) protégés par Google ID token.

## 2) ACTIONS DE CETTE PASSE

- Création du router FastAPI src/backend/features/dev\_auth/router.py servant /dev-auth.html.
- Patch de main.py pour monter ce router sans préfixe.
- Ajout du fichier src/frontend/dev-auth.html (outil de test GIS).
- Nettoyage : suppression d'un répertoire erroné src/frontend/features/dev\_auth/.
- Correction des variables Cloud Run : GOOGLE\_OAUTH\_CLIENT\_ID sans '=' final, AUTH\_DEV\_MODE=0.
- Déploiement révision emergence-app-00041-6wl.
- Tests avec dev-auth.html : génération ID token GIS → aud valide, iss = <https://accounts.google.com>.
- Tests avec curl.exe : 200 avec Authorization Bearer ID token ; 401 sans Authorization.
- Vérification logs Cloud Run : plus d'erreurs Auth failure après correctifs.

## 3) RÉSULTATS POST-DÉPLOIEMENT

- Authentification IAP supprimée (désactivée au niveau LB/backends).
- Authentification app-side Google via GIS opérationnelle en production.
- Endpoints /api/\* désormais protégés par ID token Google valide.
- Mode test (AUTH\_DEV\_MODE) désactivé en prod.
- Page /dev-auth.html opérationnelle et confirmée utile pour debug.
- Sécurité validée : accès public rejeté (401), accès avec ID token accepté (200).

## 4) PROCHAINES ÉTAPES

- Frontend : intégrer GIS directement dans l'UI de l'app (remplacer dev-auth manuel).
- Créer un wrapper JS fetchWithAuth pour injecter Authorization: Bearer .
- Stockage et rafraîchissement ID token (sessionStorage).
- Backend : optionnel - ajouter allowlist emails/domains via GOOGLE\_ALLOWED\_EMAILS / GOOGLE\_ALLOWED\_HD.

- Restreindre /dev-auth.html aux environnements debug/non-prod.
- Documentation : maintenir consignes Cloud Run & GIS, vérifier staging/prod avec plusieurs client\_id.

## **5) CHECKLIST DE SORTIE**

- ARBO-LOCK respecté (création router dev\_auth annoncée, snapshot mis à jour).
- Router intégré dans main.py.
- Fichier dev-auth.html livré et testé.
- Variables d'environnement corrigées (GOOGLE\_OAUTH\_CLIENT\_ID propre, AUTH\_DEV\_MODE=0).
- Tests manuels et Cloud Run : 200 avec ID token, 401 sans.
- Intégration frontend GIS → à faire.
- Durcissement backend (allowlist) → à faire.