

# Guide de Restitution du Projet MLOps

Vous pouvez restituer votre projet jusqu'au **30 décembre 2025 à 23h59** sur la plateforme Bootcamp. Votre projet doit être restitué sous la forme d'un **document PDF** avec les sections suivantes :

1. Informations du groupe

- Indiquez votre identifiant de groupe **GX-MGXX**
- Listez tous les membres de votre groupe avec (Nom et Prénom)
- Le titre de votre projet

2. Informations du projet

- Fournissez l'URL complète de votre dépôt GitHub public. Assurez-vous que votre dépôt est bien **PUBLIC** et qu'il est accessible. Un dépôt privé ne pourra pas être évalué.
- Description du projet : ajouter une description succincte :
  - Ce que fait votre application
  - Le problème qu'elle résout
  - Les technologies principales utilisées (Python, FastAPI, TensorFlow, etc.)

3. Services AWS utilisés

- Listez **tous** les services AWS que votre projet utilise avec une brève description de leur rôle.

| Service AWS | Nom de la ressource | Rôle  |
|-------------|---------------------|---|
| S3          | s3-g0mg00           | Stockage des modèles ML et des données      |
| ECR         | ecr-g0mg00          | Stockage des images Docker de l'application |
| ECS         | ecs-g0mg00          | Orchestration et exécution des conteneurs   |
| App Runner  | apprunner-g0mg00    | Déploiement et hébergement de l'API         |

- Si vous n'avez **pas** utilisé le format de nommage avec votre identifiant de groupe (gXmgXX), vous **devez** lister les noms exacts que vous avez utilisés pour chaque

service.

4. URL de l'interface / API : si votre projet possède une interface web accessible publiquement ou une API déployée, fournissez son URL.
5. Architecture des microservices : si votre projet est composé de plusieurs microservices, décrivez l'architecture.

## Critères d'évaluation

### 1. Complétude de l'infrastructure (30%)

- Services AWS correctement configurés
- Infrastructure as Code avec Terraform
- Respect des conventions de nommage

### 2. Pipeline CI/CD (30%)

- Automatisation complète de l'intégration et du déploiement
- Tests unitaire/intégration intégrés dans le pipeline
- Robustesse du workflow

### 3. Qualité du code (20%)

- Structure du projet et des pipelines (data, modèles, API)
- Documentation (README, commentaires)
- Bonnes pratiques de développement

### 4. Fonctionnalités (10%)

- Application fonctionnelle
- Cas d'usage pertinent
- Complexité technique

### 5. Documentation (10%)

- Clarté du document de restitution
- Qualité des explications
- Diagrammes et schémas