

## Consignes du Mini-Projet : Docker & Deep Learning

### 1. Organisation du travail

Le mini-projet est réalisé en groupe de deux (02) étudiants. Chaque groupe est responsable de l'ensemble des phases du projet, depuis la compréhension du problème jusqu'au déploiement du modèle.

### 2. Objectif général du projet

L'objectif de ce mini-projet est de mettre en œuvre un projet d'Intelligence Artificielle complet et data-driven en exploitant Docker pour la reproductibilité, l'accélération de l'entraînement et le déploiement d'un modèle de Deep Learning.

### 3. Choix de la thématique et du modèle

Chaque groupe doit choisir une thématique libre en lien avec l'Intelligence Artificielle, selon ses centres d'intérêt.

- Exemples de thématiques possibles :

Vision par ordinateur (classification d'images, détection d'objets, OCR, etc.).

Traitement du langage naturel (NLP).

Reconnaissance vocale.

Séries temporelles et prédiction.

Le groupe doit ensuite sélectionner un modèle de Deep Learning approprié (CNN, RNN, LSTM, Transformer, etc.) et justifier ce choix.

### 4. Phases obligatoires du projet (Data-Driven Project)

#### 4.1 Compréhension du problème

Décrire la problématique à résoudre, définir les objectifs du modèle et préciser le contexte d'utilisation.

#### 4.2 Données

Présenter la source des données (dataset public ou construit), décrire les étapes de prétraitement et expliquer l'organisation des données à l'aide de volumes Docker.

#### 4.3 Entraînement du modèle

Implémenter et entraîner le modèle de Deep Learning en utilisant Docker.

L'utilisation du CPU ou du GPU doit être clairement indiquée. Une comparaison entre l'exécution locale et Dockerisée est obligatoire.

### 4.4 Conteneurisation avec Docker

Créer un ou plusieurs Dockerfile pour le projet, gérer les dépendances et utiliser Docker Compose si nécessaire. Les bonnes pratiques de conteneurisation doivent être respectées.

### 4.5 Déploiement du modèle

Déployer le modèle sous forme d'un service de prédiction via une API (Flask ou FastAPI), entièrement conteneurisée et testée.

### 4.6 Dimension MLOps

Mettre en évidence la reproductibilité des expériences, la structuration du projet et le rôle de Docker dans le cycle MLOps.

## 5. Livrables attendus

- Les livrables du projet comprennent :

Le code source complet du projet.

Les fichiers Dockerfile et docker-compose.yml.

Un fichier README décrivant les étapes d'exécution.

Un rapport écrit au format PDF.

Une présentation orale avec démonstration.

## 6. Évaluation

L'évaluation portera sur la pertinence du problème choisi, la qualité du modèle de Deep Learning, la bonne utilisation de Docker, la complétude du cycle data-driven, ainsi que la clarté du rapport et de la présentation orale.