Projet DatAnalystFlow360

Objectif global

Développer un pipeline complet permettant à un **Data Analyst** de collecter, transformer et analyser des données, tout en automatisant les workflows grâce au CI/CD, afin de produire des **insights exploitables (BI, tableaux de bord, KPIs)**.

Étapes du pipeline Data Analyst avec CI/CD intégré

Étape 1 – Collecte et intégration des données

Objectifs :

- Récupérer des données issues de sources variées (fichiers CSV/Excel, APIs, bases SQL/NoSQL).
- Intégrer ces données dans un **Data Lake / Data Warehouse**.

Outils :

- Python (pandas, requests)
- PostgreSQL (Data Warehouse)
- Google BigQuery (Cloud)
- GitHub Actions (CI/CD pour automatiser ingestion + tests de qualité des données)

Étape 2 – Nettoyage et Transformation (ETL)

Objectifs :

- Automatiser le nettoyage (valeurs manquantes, doublons, outliers).
- Standardiser et enrichir les datasets.
- Mettre en place des **pipelines de transformation automatisés**.

Outils :

- Python (pandas, PySpark pour big data)
- dbt (ETL cloud-oriented)
- Airflow (orchestration des pipelines)
- GitHub Actions (tests unitaires sur transformations + déploiement automatisé)

Étape 3 – Stockage et Structuration des données

Objectifs :

- Centraliser les données nettoyées dans un **Data Warehouse**.
- Créer des **Data Marts** thématiques (ex : ventes, clients, opérations).
- Assurer la gouvernance et la sécurité.

Outils :

- PostgreSQL / Snowflake (cloud DWH)
- Google BigQuery (Data Warehouse cloud)
- Docker (containerisation pour reproductibilité)
- dbt tests + GitHub Actions (CI/CD sur le schéma des données)

Étape 4 – Visualisation et Business Intelligence

Objectifs :

- Créer des dashboards interactifs pour suivre KPIs et insights.
- Automatiser la mise à jour des dashboards à partir des pipelines CI/CD.
- Comparer plusieurs outils BI.

Outils :

- Power BI (on-premise / cloud)
- Tableau (BI avancée)
- Looker Studio (Google Cloud)
- Python (Dash / Streamlit pour visualisations customisées)

Étape 5 – Automatisation et Monitoring

Objectifs :

- Suivre la qualité des données et l'exécution des pipelines.
- Mettre en place du monitoring pour détecter erreurs et dérives.
- Automatiser les livraisons (CI/CD complet).

Outils :

- ELK Stack (Elasticsearch, Logstash, Kibana) pour logs
- Grafana (monitoring dashboards)
- GitHub Actions + Jenkins (CI/CD complet du pipeline)
- Docker + Docker Compose (déploiement reproductible)

Résultat final attendu (comme DataFlow360 classique)

- ✓ Données intégrées et nettoyées dans un **Data Warehouse / Data Lake**.
- ✓ Des **dashboards BI** clairs et interactifs pour aider à la décision.
- ✓ Un pipeline **automatisé et déployé via CI/CD**, reproductible et maintenable.

Livrables attendus

1. **Code source sur GitHub** :
    - Pipelines ETL
    - Docker Compose
    - Config CI/CD (GitHub Actions ou Jenkins)
2. **Résultats analytiques** :
    - Data Warehouse / Data Marts prêts à l'exploitation
    - Dashboards BI (Power BI, Tableau, Looker Studio, Streamlit)
3. **Documentation et communication** :
    - **Un rapport PDF (mémoire)** de 30-50 pages structuré (problématique, méthodologie, résultats, limites, perspectives).
    - **Un PowerPoint** de 15-20 slides pour présentation synthétique.
    - **Une vidéo de démonstration** (5-8 minutes) montrant :
        - Le pipeline automatisé,
        - Le résultat (dashboard BI),
        - Le déploiement CI/CD.

En résumé :
DataFlow360 version Data Analyst est un projet de bout en bout qui :

- Automatise les workflows avec CI/CD,
- Aboutit aux mêmes livrables techniques (pipeline, dashboards, monitoring),
- Se conclut par une **présentation académique et professionnelle** (rapport, slides, vidéo).