

# FORMATION : DataScientest / Data Engineer

**PROJET (Nov. 2024) : Analyse des Impacts Environnementaux Textiles, avec EcoBalyse v2.4.0**

**Temps Partiel (9 mois) - `jan24_continu_de / cde_projet_ecobalyse`**

# Sommaire

## 1. CONTEXTE

Ecobalyse  
GitHub / Equipe

## 2. SOLUTION

Synoptique  
Architecture

## 3. TECHNIQUE

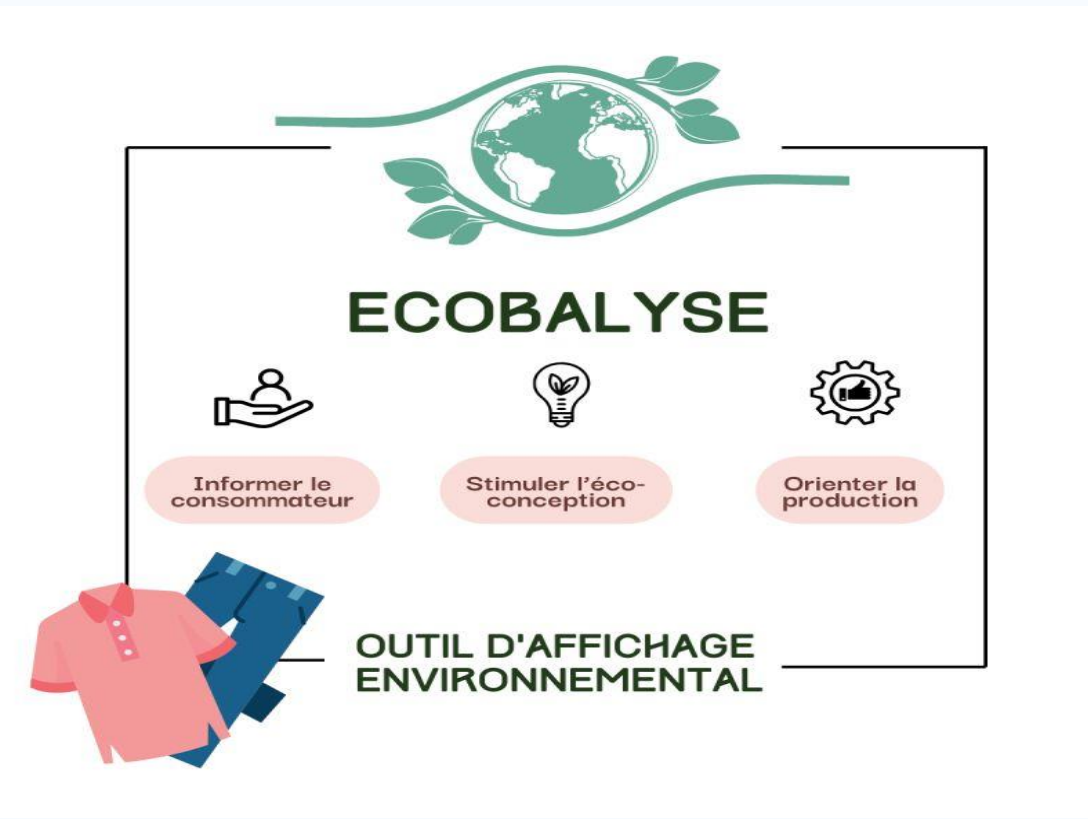
Scripts  
Machine Learning

## 4. DEMONSTRATION

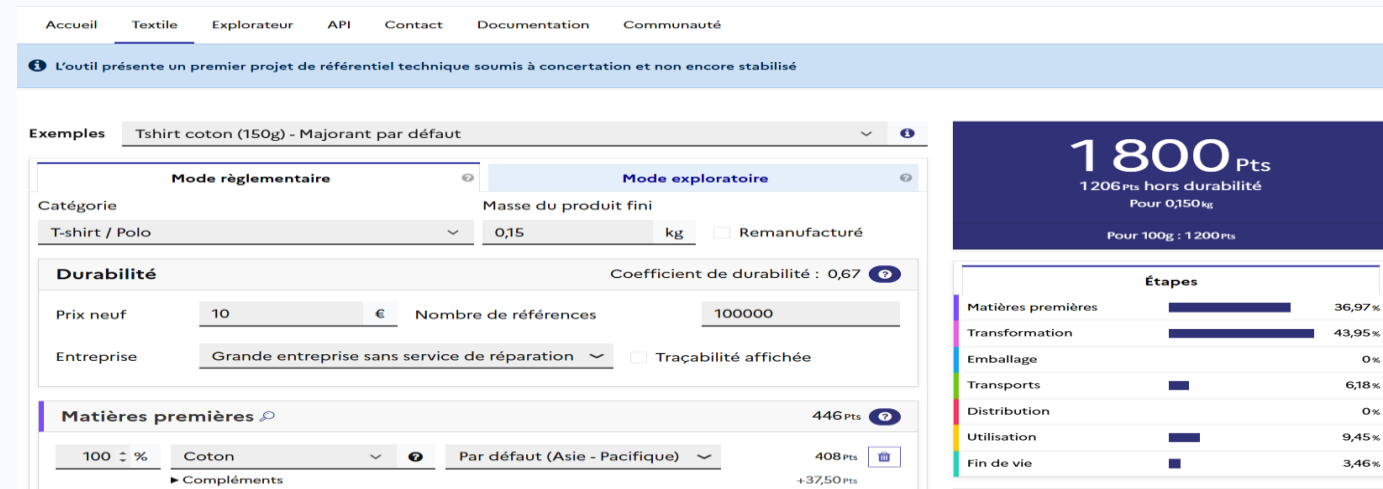
Flask  
Dash

# 1. CONTEXTE ECOBALYSE

Écobalyse est un outil développé par l'État français pour calculer l'impact écologique des produits textiles (*et alimentaires*) distribués en France.



## Présentation de l'interface et des Données Textiles Ecobalyse



The screenshot shows the Ecobalyse web interface. The top navigation bar includes links for Accueil, Textile, Explorateur, API, Contact, Documentation, and Communauté. A blue banner below the navigation bar states: "L'outil présente un premier projet de référentiel technique soumis à concertation et non encore stabilisé".

The main content area is titled "Exemples" and shows a simulation for "T-shirt coton (150g) - Majorant par défaut". The interface is divided into two modes: "Mode réglementaire" and "Mode exploratoire".

**Mode réglementaire:**

- Catégorie: T-shirt / Polo
- Masse du produit fini: 0,15 kg
- Coefficient de durabilité: 0,67
- Prix neuf: 10 €
- Nombre de références: 100000
- Entreprise: Grande entreprise sans service de réparation
- Traçabilité affichée: ☐
- Matières premières: 446 Pts
- 100 % Coton
- Par défaut (Asie - Pacifique)
- Compléments: 408 Pts
- +37,50 Pts

**Mode exploratoire:**

- Remanufacturé: ☐

**Summary:**

- 1800 Pts
- 1206 Pts hors durabilité
- Pour 0,150 kg
- Pour 100g : 1200 Pts

**Étapes (Steps):**

Étape	Pourcentage
Matières premières	36,97%
Transformation	43,95%
Emballage	0%
Transports	6,18%
Distribution	0%
Utilisation	9,45%
Fin de vie	3,46%

## Présentation de l'API Ecobalyse



The screenshot shows the Ecobalyse API documentation. It is organized into sections: "Commun" (Common) and "Textile".

**Commun:**

- Documentation de l'API multi-domaines au format OpenAPI
- GET /

**Textile:**

- Documentation de l'API pour le domaine textile au format OpenAPI
- GET /textile/countries: Liste des pays utilisables pour les simulations textiles.
- GET /textile/materials: Liste des matières textile
- GET /textile/products: Liste des types de produits textiles
- GET /textile/simulator: Calcul des impacts environnementaux d'un produit textile
- POST /textile/simulator: Calcul des impacts environnementaux d'un produit textile
- GET /textile/simulator/{impact}: Calcul d'un impact environnemental spécifique d'un produit textile
- POST /textile/simulator/{impact}: Calcul des impacts environnementaux d'un produit textile
- GET /textile/simulator/detailed: Calcul exhaustif des impacts environnementaux d'un produit textile
- POST /textile/simulator/detailed: Calcul des impacts environnementaux d'un produit textile

Pour en savoir plus, on peut visiter le site d'Écobalyse [ici](#).

A voir également :

- [GitBook Écobalyse](#)
- [Explorateur Écobalyse](#)
- [Documentation de API Écobalyse](#)
- [Ademe](#)

# 1. CONTEXTE

## GITHUB / EQUIPE

Afin de favoriser le travail en équipe, et suivre l'avancement du projet, un **GitHub** a été mis en place, avec différentes branches. Un moyen aussi de centraliser la documentation du projet.

The screenshot shows the GitHub repository page for 'DataScientest'. The README file is open, displaying the project title 'Datascientest: projet EcoBalyse (Nov. 2024)' and a description: 'Data Engineering End-to-End Project : AirFlow, Dash, Flask, Docker, Redis, MongoDB, Python'. The document was last updated on Mar. 12/11/2024, version v0.2.0. A 'Sommaire' (Table of Contents) section lists various links for context, presentation, mode d'emploi, solution technique, and details techniques. A branch switcher overlay is visible on the right side of the page, showing the current branch 'main' and a list of other branches: 'develop-aben', 'develop-aser', and 'develop-tdel'. The search bar in the branch switcher is empty, and the 'default' button is highlighted.

README MIT license

DataScientest

## Datascientest: projet EcoBalyse (Nov. 2024)

Data Engineering End-to-End Project : AirFlow, Dash, Flask, Docker, Redis, MongoDB, Python

Dernière Mise A Jour du Document : Mar. 12/11/2024 - Version : v0.2.0

### Sommaire

- [Contexte](#)
- [Présentation](#)
  - [Etapas du projet](#)
- [Mode d'emploi](#)
  - [Pré-requis \( ./info.sh | ./mode.sh | ./starter.sh \)](#)
  - [\(Ré\)Initialiser \( ./init.sh \)](#)
  - [\(Re\)Configurer \( ./setup.sh \)](#)
  - [\(Re\)Charger \( ./load.sh \)](#)
- [Solution technique](#)
  - [Schéma de Principe](#)
  - [Dossiers & Répertoires](#)
- [Détails techniques](#)
  - [Architecture](#)
  - [ETL](#)
  - [MongoDB](#)
  - [Redis](#)
  - [Flask](#)
  - [Dash](#)
  - [AirFlow](#)
- [A propos d'Ecobalyse](#)

main 4 Branches 0 Tags

Switch branches/tags

Find or create a branch...

Branches Tags

✓ main default

develop-aben

develop-aser

develop-tdel

View all branches

# Sommaire

## 1. CONTEXTE

Ecobalyse  
Méthodologie / Equipe

## 2. SOLUTION

Synoptique  
Architecture

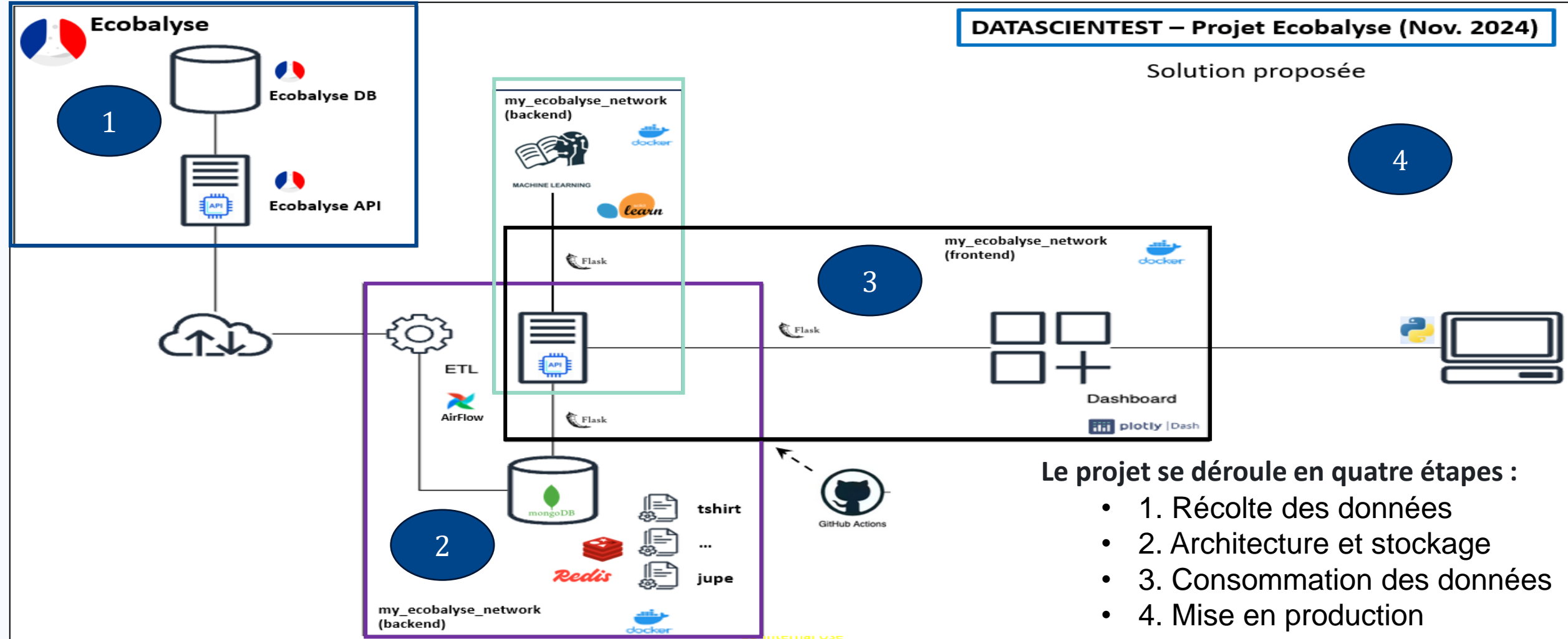
## 3. TECHNIQUE

Scripts  
Machine Learning

## 4. DEMONSTRATION

Flask  
Dash

## 2. SOLUTION SYNOPTIQUE





## 2. SOLUTION ARCHITECTURE

L'architecture et l'arborescence du projet sont structurées autour de **différents conteneurs Docker**, afin de faciliter les mises à jour, indépendamment et facilement.

```
prj_ECOBALYSE
├── data
│   ├── mongo
│   ├── redis
│   ├── params01_T-shirt.txt
│   ├── params02_Pull.txt
│   ├── params03_Pantalon.txt
│   ├── params04_Manteau.txt
│   ├── params05_Maillot-de-bain.txt
│   ├── params06_Jupe.txt
│   ├── params07_Jean.txt
│   ├── params08_Chemise.txt
│   ├── params09_Chaussettes.txt
│   ├── params10_Calecon.txt
│   └── params11_Slip.txt
├── dag
├── dash
├── etl
├── flask
├── img
├── logs
├── mongo
├── notebooks
├── redis
├── PRJ-ECOBALYSE-00-FICHE_PROJET.pdf
├── LICENSE
├── CONVENTIONS.md
├── README.md
├── docker-compose.yml
├── lisezMoi.txt
├── info.sh
├── load.sh
├── setup.sh
├── init.sh
└── starter.sh
```

```
etl
├── Dockerfile.etl
├── constants.py
├── extract1.py
├── get_constants.py
├── requirements.txt
├── utils01.py
├── utils02.py
├── utils03.py
```

```
flask
├── Dockerfile.flask
├── constants3.py
├── get_constants3.py
├── init_flask.sh
├── mongo_queries.py
├── redis_queries.py
├── requirements.txt
├── stylesheets
│   ├── listMongoBDD.css
│   └── styles.css
├── templates
│   ├── bienvenue.html
│   ├── index.html
│   └── listMongoBDD.html
├── test_flask.py
└── utils.py
```

```
mongo
├── Dockerfile.mongo
├── constants1.py
├── get_constants1.py
├── init_mongo.js
├── init_mongo.sh
├── mongo.conf
└── test_mongo.py

redis
├── Dockerfile.redis
├── constants2.py
├── get_constants2.py
├── init_redis.sh
├── redis.conf
└── test_redis.py
```

```
dash
├── Dockerfile.dash
├── assets
│   ├── info_countryDyeing.txt
│   ├── info_countryFabric.txt
│   ├── info_countryMaking.txt
│   ├── info_countrySpinning.txt
│   ├── info_masse.txt
│   ├── info_matiere.txt
│   ├── info_mode.txt
│   ├── info_pays.txt
│   ├── info_product.txt
│   ├── linear_regression_model_basic.pkl
│   ├── page0.jpg
│   └── styles.css
├── components.py
├── constants4.py
├── get_constants4.py
├── init_dash.sh
├── old_page13.py
├── page0.py
├── ...
├── page31.py
├── requirements.txt
└── test_dash.py
```



(Reste à faire)

# Sommaire

## 1. CONTEXTE

Ecobalyse  
Méthodologie / Equipe

## 2. SOLUTION

Synoptique  
Architecture

## 3. TECHNIQUE

Scripts  
Machine Learning

## 4. DEMONSTRATION

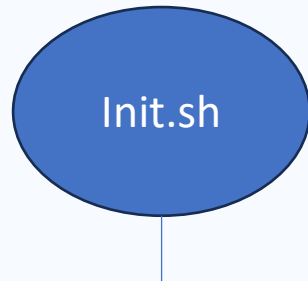
Flask  
Dash



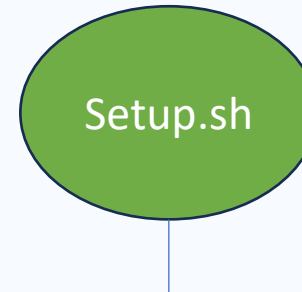
### 3. TECHNIQUE SCRIPTS

- [\(Ré\)Initialiser \( ./init.sh \)](#)
- [\(Re\)Configurer \( ./setup.sh \)](#)
- [\(Re\)Charger \( ./load.sh \)](#)

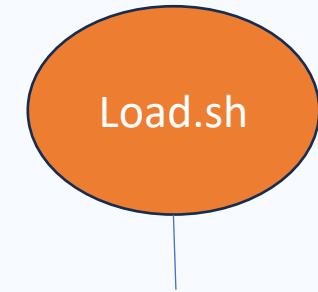
Trois scripts permettent de s'assurer du bon fonctionnement du projet :



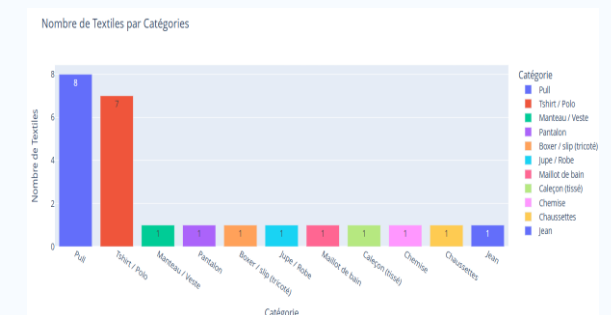
- Nettoie l'environnement pour une nouvelle exécution du setup
- Supprime les images des conteneurs préalablement créés
- Supprime les réseaux préalablement créés pour les échanges entre dockers
- Supprime les données préalablement créées lors de l'exécution du setup



- Supprime les fichiers logs si déjà existants
- Mets en place les conteneurs nécessaires à l'extraction et la transformation des données (*ecblwebscraping* , *ecblmongodb* , *ecblredis*)



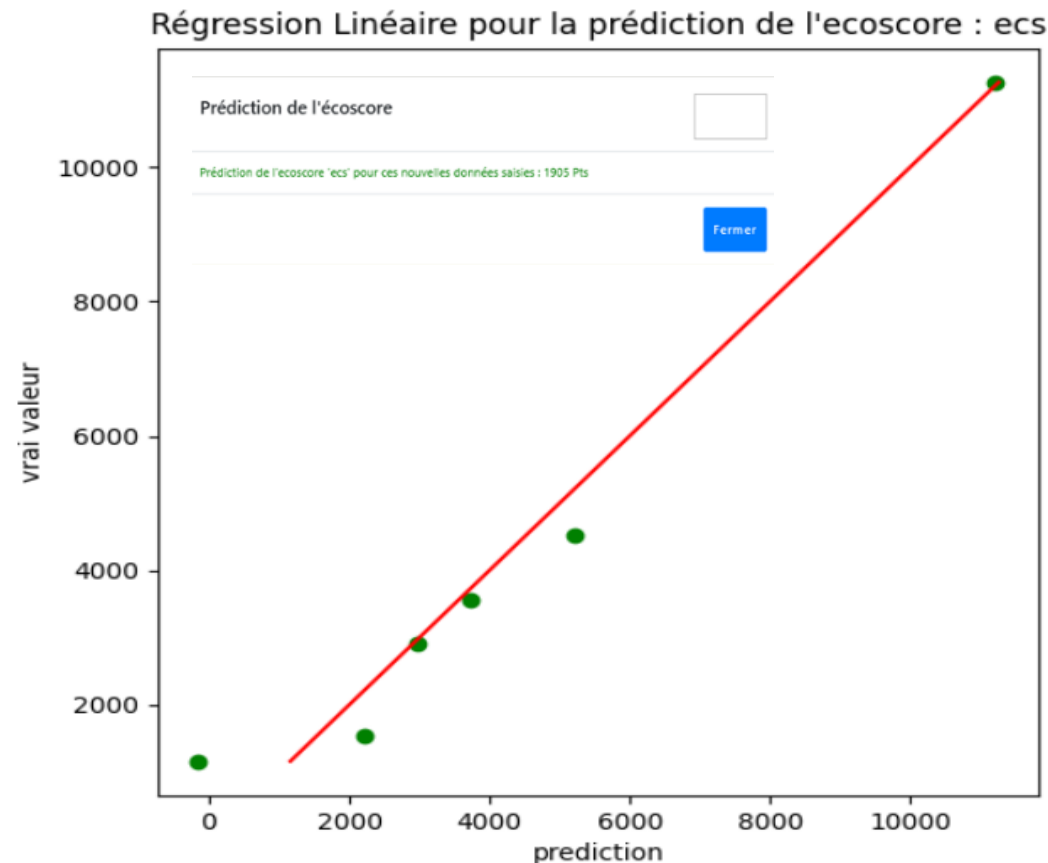
- Permet d'avoir une visualisation graphique des données traitées à l'étape précédente via le setup
- Fournit des résultats d'ecoscore basés sur le modèle de Machine Learning choisi pour tester les données fournies en entrée (27 échantillons)



# 3. TECHNIQUE MACHINE LEARNING

Objectif : reproduire le calcul de [l'API Ecobalyse](#), par prédiction

Suite à l'entrainement de plusieurs modèles de Machine Learning, la Régression Linéaire a été retenue, car les résultats fournis sont ceux qui se rapprochent le plus de ceux de [l'API Ecobalyse](#).



Visualisation Des Données : Listing & Description

Charger les données

Tshirt / Polo

Libelle	Categorie	ecs
Tshirt lin 150g France Mode ethique	Tshirt / Polo	446
Tshirt coton bio 150g France Mode ethique	Tshirt / Polo	507
Tshirt coton 150g France Mode traditionnelle	Tshirt / Polo	751
Tshirt coton 150g Chine Mode fast fashion	Tshirt / Polo	1118
Tshirt synthétique 150g Asie Mode ultra fast fashion	Tshirt / Polo	1547
Tshirt coton 150g Pakistan Mode traditionnelle	Tshirt / Polo	1714
Tshirt coton 150g Majorant par défaut	Tshirt / Polo	1855

Exemples

Tshirt coton (150g) - Majorant par défaut

Mode réglementaire

Mode exploratoire

Catégorie

T-shirt / Polo

Masse du produit fini

0,15 kg

Remanufacturé

Durabilité

Coefficient de durabilité : 0,67

Prix neuf

10 €

Nombre de références

100000

Entreprise

Grande entreprise sans service de réparation

Traçabilité affichée

Matières premières

100 %

Coton

Par défaut (Asie - Pacifique)

446 Pts

408 Pts

+37,50 Pts

1800 Pts

1206 Pts hors durabilité

Pour 0,150 kg

Pour 100g : 1200 Pts

Étapes

Étapes	Pourcentage
Matières premières	36,97%
Transformation	43,95%
Emballage	0%
Transports	6,18%
Distribution	0%
Utilisation	9,45%
Fin de vie	3,46%

# Sommaire

## 1. CONTEXTE

Ecobalyse  
Méthodologie / Equipe

## 2. SOLUTION

Synoptique  
Architecture

## 3. TECHNIQUE

Scripts  
Machine Learning

## 4. DEMONSTRATION

Flask  
Dash

# 4. INTERFACE

## FLASK (ACCUEIL)

### Flask

Bienvenue sur la page d'accueil de votre application Flask !

Pour vérifier le bon fonctionnement de votre application, cliquez sur les liens suivants :

[Lister les BDD MongoDB](#)

[Vérifier le contenu Ecobalyse de la BDD MongoDB](#)

[Vérifier le contenu Ecobalyse de la BDD Redis](#)

Pour obtenir quelques métriques sur les données récupérées, cliquez sur le(s) bouton(s) suivant(s) :

Métriques Environnementales

Métriques Économiques

Métriques de Durabilité

Métriques de Transport

Métriques de Traçabilité

Métriques de Transformation

(Reste à faire)

(Reste à faire)

(Reste à faire)

(Reste à faire)

(Reste à faire)

A propos des métriques disponibles :

Métriques Environnementales

Calculer le score environnemental moyen pour chaque catégorie de produit, mode, ou pays

Analyser la distribution des scores environnementaux pour identifier les produits avec des scores particulièrement élevés ou bas

Calculer l'impact moyen des matières premières sur le score environnemental

Métriques Économiques

Calculer le prix moyen des produits par catégorie

Calculer le nombre moyen de références pour chaque mode

Métriques de Durabilité

Calculer la durabilité moyenne des produits en fonction de la matière

Métriques de Transport

Calculer le ratio moyen de transport aérien pour chaque catégorie de produit

Métriques de Traçabilité

Calculer le pourcentage de produits pour lesquels la traçabilité est disponible

Métriques de Transformation

Calculer l'impact moyen de la transformation sur le score environnemental

Lancer Flask

- lancer le script `./load.sh -adm` pour lancer Flask via un *navigateur Web*.

## 4. INTERFACE DASH (ACCUEIL)

### Lancer Dash

- lancer le script `./load.sh` pour lancer Dash via un *navigateur Web*.

## Dashboard EcoBalyse (visiter le site EcoBalyse)

### A propos ...

#### Textiles & Environnement

L'industrie textile est l'une des plus polluantes au monde <sup>1, 2, 3, 4, 5</sup>.

En lien avec les préoccupations actuelles, et sur la base d'[Écobalyse](#), cet outil propose un comparatif de coûts environnementaux, en vue de favoriser un modèle de production plus durable, et de fournir des recommandations, ou des conseils, sur la manière de réduire l'impact écologique de textiles courants.

En savoir plus :

<sup>1</sup>[la-goose.com](#), <sup>2</sup>[oxfamfrance.org](#), <sup>3</sup>[ecologie.gouv.fr](#), <sup>4</sup>[climateseed.com](#), <sup>5</sup>[wwf.ch](#),

