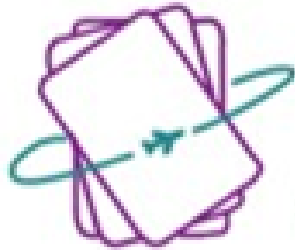


Documentation technique

# Fresqu'AéroMaps

21 septembre 2025



**Auteur :** Le Ber Tom

**Rôle :** Développeur du projet (Stagiaire)

**E-mail :** [tom.leber.toulouse@gmail.com](mailto:tom.leber.toulouse@gmail.com)

**Téléphone :** PRIVATE

## Table des matières

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Présentation générale</b>  | <b>2</b> |
| 1.1      | Présentation de la <i>Fresqu'Aéro</i> . . . . .                             | 2        |
| 1.2      | Présentation de <i>Fresqu'AéroMaps</i> . . . . .                            | 2        |
| 1.3      | Utilisation de <i>Fresqu'AéroMaps</i> . . . . .                             | 2        |
| <b>2</b> | <b>Architecture de <i>Fresqu'AéroMaps</i></b>                               | <b>3</b> |
| 2.1      | Arborescence de <i>Fresqu'AéroMaps</i> . . . . .                            | 3        |
| 2.2      | Description détaillée de l'architecture de <i>Fresqu'AéroMaps</i> . . . . . | 3        |
| <b>3</b> | <b>Installation et Lancement de <i>Fresqu'AéroMaps</i></b>                  | <b>5</b> |
| 3.1      | Installation de <i>Fresqu'AéroMaps</i> . . . . .                            | 5        |
| 3.2      | Lancement de <i>Fresqu'AéroMaps</i> en local . . . . .                      | 6        |
| 3.3      | Lancement de <i>Fresqu'AéroMaps</i> en ligne . . . . .                      | 6        |
| <b>4</b> | <b>Tâches restantes</b>   | <b>7</b> |
| <b>5</b> | <b>Contacts &amp; Liens pratiques</b>                                       | <b>8</b> |
| 5.1      | Contacts . . . . .  | 8        |
| 5.2      | Liens pratiques . . . . .   | 8        |

# 1 Présentation générale

## 1.1 Présentation de la *Fresqu'Aéro*

La *Fresqu'Aéro*, inspirée de la *Fresque du Climat*, est un atelier ludique, scientifique et collaboratif qui vise à sensibiliser et questionner le plus grand nombre sur la place du transport aérien dans la société.

Cet atelier s'organise sous la forme de petits groupes de discussion (encadrés par un animateur), dans lesquels les participants sont amenés à échanger sur les enjeux sociétaux et environnementaux, les vulnérabilités et les leviers d'actions liés au transport aérien.

La *Fresqu'Aéro* est un atelier créé par des membres du personnel ENAC et ISAE-Supaéro. Pour tout complément d'informations, contactez Isabelle Laplace (voir [Section 5](#)).

## 1.2 Présentation de *Fresqu'AéroMaps*

L'application *Fresqu'AéroMaps* intervient lors de la dernière phase de la *Fresqu'Aéro*, consistant en l'exploration de différents leviers d'actions permettant de réduire l'impact environnemental du transport aérien.

Cette application, basée sur l'outil AéroMAPS ([Site Web](#)), implémente plusieurs graphiques modélisant l'évolution de la trajectoire des émissions de  $CO_2$  ainsi que le pourcentage du budget mondial des ressources consommées par le transport aérien entre aujourd'hui et 2050, en fonction des différents leviers d'actions choisis par chaque groupe.

## 1.3 Utilisation de *Fresqu'AéroMaps*

*Fresqu'AéroMaps* possède actuellement deux supports (l'application est hébergée sur deux sites web distincts), chacun ayant différents avantages :

- **Support Render** : Très peu d'étapes de mise en place et plus intuitif mais de longs temps de chargement.
- **Support Binder** : Courts temps de chargement mais mise en place plus complexe et moins intuitive.

Vous trouverez un guide d'utilisation détaillé pour chaque support sur le [Google Drive](#) du projet.

## 2 Architecture de *Fresqu'AéroMaps*

### 2.1 Arborescence de *Fresqu'AéroMaps*

L'arborescence du projet *Fresqu'AéroMaps* est la suivante :

|                        |  |
|------------------------|--|
| fresqu-aeromaps/       |  |
| ├─ data/               |  |
| │ └─ graphs_json/      | Fichiers <i>JSON</i> de description des cartes implémentées et des paramètres <i>AéroMAPS</i> utilisés dans le logiciel. |
| │ └─ ...               | Autres fichiers guide d'utilisation et logo du logiciel.   |
| ├─ src/                |  |
| │ └─ core/             |  |
| │ │ └─ aeromaps_utils/ | Fichiers <i>Python</i> contenant toutes les fonctions d'interaction avec <i>AéroMAPS</i> .                               |
| │ └─ bqplot_figures/   | Fichiers <i>Python</i> définissant les graphiques simulant les scénarios de chaque groupe.                               |
| │ └─ crud/             | Fichiers <i>Python</i> de gestion des fichiers <i>JSON</i> .   |
| │ └─ ui/               | Fichiers <i>Python</i> créant l'interface utilisateur de l'application.  |
| │ └─ ...               | Autres fichiers <i>Python</i> utilitaires.   |
| ├─ .env                | Fichier d'environnement définissant le <i>PYTHONPATH</i> .   |
| ├─ requirements.txt    | Fichier listant les dépendances <i>Python</i> du projet.   |
| ├─ Dockerfile          | Fichier de configuration <i>Docker</i> .   |
| ├─ README.md           | Documentation principale du projet.  |
| ├─ app.py              | Fichier racine de la version "web" de l'application.   |
| └─ app.ipynb           | Fichier racine de la version "Jupyter Notebook" de l'application.  |

### 2.2 Description détaillée de l'architecture de *Fresqu'AéroMaps*

Vous trouverez ci-dessous une description détaillée de chaque composante de *Fresqu'AéroMaps*, triée par thématique.

#### 2.2.1 Interaction entre *Fresqu'AéroMaps* et *AéroMAPS*

L'interaction entre *Fresqu'AéroMaps* et *AéroMAPS* est gérée par les fichiers situés dans le dossier `./src/core/aeromaps_utils/`.

La classe `ProcessEngine`, située dans le fichier `./src/core/aeromaps_utils/process_engine.py`, initialise un objet `AeroMAPSProcess` pour chaque groupe et modifie via la fonction `compute_process` du même fichier différents paramètres d'entrée en fonction des cartes choisies et par défaut (indépendamment de la sélection des cartes (ex : renouvellement de la flotte)).

### 2.2.2 Génération des graphiques résultats

Il existe trois types de graphiques permettant de simuler les résultats des scénarios choisis par chaque groupe.

Chaque type de graphique est créé à partir de la classe mère abstraite BaseGraph, située dans le fichier `./src/bqplot_figures/base_graph.py`.

Les trois types de graphiques créés par le logiciel sont les suivants :

- Les graphiques ProspectiveScenario, créés à partir de la classe du même nom, située dans le fichier `./src/bqplot_figures/prospective_scenario_graph.py`.
  - Pour chaque groupe, ils simulent en fonction des cartes sélectionnées la trajectoire des émissions de  $CO_2$  émises par le transport aérien entre 2019 et 2050 par rapport à une croissant annuelle de 3%.
- Les graphiques ProspectiveScenarionGroupComparizon, créés à partir de la classe du même nom, située dans le fichier `./src/bqplot_figures/prospective_scenario_graph.py`.
  - Ils comparent les trajectoires des émissions de  $CO_2$  émises par le transport aérien entre 2019 et 2050 de chaque groupe par rapport à une croissant annuelle de 3%.
- Les graphiques MultidisciplinaryGraph, créés à partir de la classe du même nom, située dans le fichier `./src/bqplot_figures/multidisciplinary_graph.py`.
  - Pour chaque groupe, ils estiment (en pourcentage) en fonction des cartes sélectionnées la consommation nécessaire en émissions totales, de  $CO_2$ , de biomasse et d'électricité entre 2019 et 2050 et les comparent à leur budget mondial respectif.

Les fichiers situés dans le dossier `./src/bqplot_figures/utils/` contiennent les fonctions utilitaires permettant à chaque type de graphique de récupérer les données nécessaires à leur tracé :

- Chaque type de graphique possède un fichier JSON associé dans un sous-dossier de `./data/graphs_json/` contenant les différentes formules de calcul du graphe à tracer en utilisant des variables de sortie d'AéroMAPS.
  - **Pour comprendre la raison d'un tel choix d'implémentation, veuillez regarder la section 2.1 du document listant les missions réalisées par Le Ber Tom.**
- Chaque formule est interprétée via la fonction `evaluate_expression_aeromaps`, située dans le fichier `./src/bqplot_figures/utils/evaluate_expression.py`.
  - Cette fonction extrait les données de sortie d'AéroMAPS via les fonctions utilitaires contenues dans le fichier `./src/bqplot_figures/utils/extract_processed_data.py`.
- Les autres données (contenues dans les JSON) associées à chaque type de graphe sont récupérées via un fichier Python associé dans le sous-dossier `./src/crud/`.

### 2.2.3 Création et mise en place de l'interface utilisateur

L'interface utilisateur de *Fresqu'AéroMaps* est créée via la librairie `ipywidgets` (car supportée par Panel et Jupyter Notebook).

La classe FresqueAeroMapsUI, située dans le fichier `./src/ui/fresque_aeromaps_UI.py`, crée l'interface utilisateur principale de l'application en assemblant les différentes constantes, figures et widgets créés dans les fichiers utilitaires du dossier `./src/ui/utils/`.

*Fresqu'AéroMaps* possède deux interfaces principales, une version "web" et une version "Jupyter Notebook", chacune ayant son propre fichier racine :

- **Version "web"** : Lancée via le fichier `./app.py`. Utilise la librairie Panel.
- **Version "Jupyter Notebook"** : Lancée via le fichier `./app.ipynb`.

## 3 Installation et Lancement de *Fresqu'AéroMaps*

### 3.1 Installation de *Fresqu'AéroMaps*

Afin de pouvoir facilement installer, modifier et utiliser l'application *Fresqu'AéroMaps* sur votre machine, il est fortement recommandé d'utiliser un environnement virtuel Python au sein d'un système Linux ou WSL (Windows Subsystem for Linux).

Vous trouverez ci-dessous une procédure d'installation simple, organisée par étapes :

1. Cloner le dépôt GitHub (<https://github.com/fresqu-aeromaps/fresqu-aeromaps>) :
  - `git clone https://github.com/fresqu-aeromaps/fresqu-aeromaps`
  - `cd fresqu-aeromaps`
2. Installer les outils de compilation et les headers Python :
  - `sudo apt update`
  - `sudo apt install build-essential python3.12-dev`
3. Créer un environnement virtuel :
  - `python3.12 -m venv .venv`
  - `source .venv/bin/activate`
4. Installer les dépendances du projet :
  - `pip install -upgrade pip setuptools wheel`
  - `pip install -r requirements.txt`
5. Créer le fichier d'environnement :
  - Créer le fichier `.env` à la racine du projet.
  - Copier le contenu du template de fichier d'environnement suivant :  
`PYTHONPATH=./src`

## 3.2 Lancement de *Fresqu'AéroMaps* en local

L'application *Fresqu'AéroMaps* possède deux interfaces, une version "web" et une version "Jupyter Notebook". Vous trouverez ci-dessous un tutoriel de lancement en local pour chaque version.

### 3.2.1 Tutoriel de lancement de la version "web"

- Via le Dockerfile, version utilisée par [Render](#) [VERSION RECOMMANDÉE] :
  - `docker build -t fresque-aeromaps .`
    - *Cette commande est un peu longue à s'exécuter, il y en aura pour environ 10-15 minutes.*
  - `docker run -rm -p 8888:8888 fresque-aeromaps`
- Via le fichier racine `app.py` :
  - `panel serve app.py -address=0.0.0.0 -port=8888 -allow-websocket-origin="*" -prefix="" -index="app"`
    - *Lors du debug, il est également recommandé d'ajouter l'option `-autoreload` afin de ne pas avoir à relancer l'application à chaque modification du code source.*
- L'application sera alors accessible à l'adresse <http://localhost:8888/app> (et <http://localhost:8888>).

### 3.2.2 Tutoriel de lancement de la version "Jupyter Notebook"

- Via le fichier racine `app.ipynb` :
  - `jupyter notebook app.ipynb`
    - *Si vous utilisez Visual Studio Code, veuillez bien à choisir le Kernel correspondant à l'environnement `.venv` en haut à droite du notebook.*

## 3.3 Lancement de *Fresqu'AéroMaps* en ligne

L'application *Fresqu'AéroMaps* est hébergée en ligne sur 2 sites :

1. Sur [render.com](https://fresqu-aeromaps.onrender.com), à l'adresse suivante : <https://fresqu-aeromaps.onrender.com>.
  - Le temps de chargement initial et de mise à jour de l'application peut être un peu long.
2. Sur [binder.org](https://fresqu-aeromaps-binder.onrender.com), à l'adresse suivante : <https://fresqu-aeromaps-binder.onrender.com>.
  - Cette version est beaucoup plus rapide que celle hébergée sur Render, mais elle est moins intuitive.

## 4 Tâches restantes

Peu après votre arrivée en tant que nouveau développeur de *Fresqu'AéroMaps*, essayez de prendre rendez-vous avec l'ancien développeur :

Tom Le Ber  
tom.leber.toulouse@gmail.com  
PRIVATE

Cela vous permettra de mieux comprendre le projet ainsi que ses différents enjeux (comment l'installer, son architecture, le fonctionnement des JSON, ext...), de répondre à d'éventuelles questions de votre part tout en pouvant prendre rapidement le relais sur les différentes tâches restantes à implémenter.

Liste des principales implémentations / améliorations restantes à réaliser sur *Fresqu'AéroMaps* :

- Transformer / Ré-implémenter l'interface "Panel" actuelle en un vrai site web :
  - Cela permettrait d'avoir plusieurs pages, avec par exemple, l'interface actuelle comme page principale.
  - Cela permettrait également d'implémenter une page secondaire, verrouillée derrière un mot de passe permettant d'ajouter, modifier et supprimer la liste différentes variables d'AéroMAPS utilisées par le logiciel lors du choix d'une carte / de la génération des différents graphiques. Voir le document sur les missions réalisées par Le Ber Tom, section "Interaction avec AeroMAPS" (2.1) pour plus de détails.
- Trouver un moyen d'utiliser le PYTHONPATH sur le Binder.
  - En effet, actuellement, Binder ne prend pas en compte le PYTHONPATH défini dans le fichier `.env`, ce qui nécessite de créer une branche parallèle spécifique au Binder dans le dépôt GitHub. La configuration actuelle est donc susceptible de créer une désynchronisation entre la version principale (utilisée par Render) et celle de Binder (sur sa propre branche).
- Régler le bug d'affichage des graphiques des nouveaux groupes sur le support Render :
  - Actuellement, lors du changement du nombre de groupes réalisant l'activité sur le support Render, les nouveaux graphiques ne s'affichent que sur un tiers de la largeur de la page. Ce bug étant dû à un problème interne de la librairie Panel (ajout dynamique d'éléments sur le menu mal géré), il vous faudra donc trouver votre propre solution (ne cherchez pas de bug(s) dans le code actuel, le problème ne vient pas de là).
  - Conditions pour reproduire le bug : Lancer la version du support Render et choisir un nouveau nombre de groupe supérieur au nombre par défaut de groupes (2).

Liste des recommandations d'améliorations / fonctionnalités supplémentaires à implémenter sur *Fresqu'AéroMaps* :

- Dû à de longs temps de chargements sur la plateforme utilisée pour le moment ([Render](#)), il est recommandé de passer à un plan supérieur sur le site web actuel si plus de budget est alloué. Alternativement, si une autre plateforme plus adaptée est trouvée, il serait judicieux d'y transférer l'application.



## 5 Contacts & Liens pratiques

### 5.1 Contacts

Contacts relatifs à *Fresqu'AéroMaps* :

- Responsable de la *Fresqu'Aéro* :  
Laplace Isabelle ([isabelle.laplace@enac.fr](mailto:isabelle.laplace@enac.fr))
- Développeur de *Fresqu'AéroMaps* :  
Le Ber Tom ([tom.leber.toulouse@gmail.com](mailto:tom.leber.toulouse@gmail.com), PRIVATE)

### 5.2 Liens pratiques

Liens relatifs à *Fresqu'AéroMaps* :

- Dépôt GitHub de *Fresqu'AéroMaps* :  
<https://github.com/fresqu-aeromaps/fresqu-aeromaps>
- Support Render de *Fresqu'AéroMaps* :  
<https://fresqu-aeromaps.onrender.com>
- Support Binder de *Fresqu'AéroMaps* :  
<https://fresqu-aeromaps-binder.onrender.com>
- Google Drive de *Fresqu'AéroMaps* :  
<https://drive.google.com/drive/folders/1yX4AJeh1WWywaGyqcGTwCMvwyP8OfEEh?usp=sharing>
- Site web d'AéroMAPS :  
<https://aeromaps.isae-superaero.fr>