

Faculté d'Ingénierie et Management de la Santé

M1 – Data Science en Santé

**Dashboard**

-

**Rapport de projet**

Présenté par

AGODE Yawo Moevi Moïse – BETE Eden – DENIS Théo – MIEMO  
Martinien Borgia – SORO Yele Christelle

Le

15 février 2025 - Lille

# Sommaire

<b>Introduction.....</b>	<b>3</b>
<b>Méthodologie.....</b>	<b>3</b>
<b>Résultats.....</b>	<b>5</b>
<b>Discussion.....</b>	<b>6</b>
<b>Conclusion.....</b>	<b>7</b>

## I. Introduction

Les infections respiratoires virales représentent une préoccupation majeure de santé publique mondiale en raison de leur forte prévalence, leur potentiel de transmission rapide et leur impact sur les systèmes de santé. L'Organisation mondiale de la santé (OMS) estime que les infections respiratoires aiguës, dont les principales causes incluent les virus grippaux et les coronavirus, sont responsables de plus de 2,5 millions de décès chaque année dans le monde (OMS, 2022). En France, selon Santé Publique France, la grippe saisonnière seule entraîne chaque année environ 10 000 décès, principalement parmi les populations vulnérables, telles que les personnes âgées ou atteintes de comorbidités chroniques (Santé Publique France, 2021).

En outre, la pandémie de COVID-19 a illustré de manière saisissante la nécessité de systèmes robustes de surveillance et de visualisation des données épidémiologiques. Cette pandémie a mis en évidence l'importance de surveiller la propagation des virus à l'échelle mondiale pour prévenir les flambées et guider les décisions de santé publique (Johns Hopkins University, 2021). Dans ce contexte, l'analyse et la représentation visuelle des données jouent un rôle important pour permettre aux décideurs, aux chercheurs et au grand public de comprendre les dynamiques complexes des infections respiratoires.

Notre projet s'inscrit dans cet effort global, avec pour objectif de concevoir un dashboard interactif et intuitif permettant d'explorer les données épidémiologiques des infections respiratoires virales.

## II. Méthodologie

Pour mener à bien notre projet, nous avons premièrement dû recueillir les données qui nous intéressaient sur le site internet de l'Organisation Mondiale de la Santé - OMS (<https://www.who.int/>), plus précisément sur la page où étaient accessibles les données concernant les gripes (<https://www.who.int/tools/flunet>).

Evidemment, ayant pour objectif d'avoir un *dashboard* dynamique qui recueillait automatiquement les nouvelles données que l'OMS publiaient, nous ne pouvions nous contenter de télécharger une seule fois le fichier *.csv* que nous proposait le site internet. Aussi nous fallait-il *web-scrap* ces dernières informations de manière automatisée. Nous avons alors écrit un algorithme grâce au langage de programmation *Python*, afin de *scrap* la page *.html* du site internet et de récupérer le lien de téléchargement du fichier *.csv* pour l'ouvrir dans un dataframe *Python*, en utilisant la bibliothèque *Pandas*.

Les données du fichier *.csv* étaient ensuite traitées et formatées avant d'être enregistrées dans un fichier *.csv* qui nous servira de base de données. En effet, pour les données des diverses gripes, l'OMS utilise des dénominations de pays, ou des réunions de pays, qui ne sont pas celles utilisées par l'Organisation des Nations Unies - l'ONU, et qui ne sont pas reconnues par les cartes interactives que nous voulions intégrer à notre *dashboard*. Ainsi, par exemple, avons-nous dû changer les termes de l'OMS "Lao People's Democratic Republic" et "Serbia and Montenegro (2003-2006)" en, respectivement, "Laos" et "Serbie", faisant ainsi le choix, dans le second cas, de concentrer les

données en Serbie et non au Monténégro, ce qui en fait ainsi une des premières limites de notre *dashboard*.

Une fois formatées, les données étaient donc enregistrées sous un fichier *.csv* pour être ré-utilisées pour le *dashboard*. Avec l'automatisation du *web-scraping* pour récupérer les nouvelles données de l'OMS, nous ne voulions pas écraser le fichier *.csv* où étaient stockées nos données, ainsi nous fallait-il créer une fonction *Python* qui, dans le fichier *.csv* mis à jour téléchargé de l'OMS, ne récupérerait que les derniers enregistrements qui avaient été entrés postérieurement à la date du dernier enregistrement que nous avions, et qui ajoutait ces nouvelles données à notre *.csv*. Cela nous permettait également, pendant que le *web-scraping* se déroulait, de continuer à alimenter la *dashboard* en données et qu'il n'utilise pas une base de données vide.

Il ne nous restait plus qu'à créer une fonction d'automatisation du *web-scraping* reprenant les fonctions décrites ci-dessus. Nous avons choisi d'automatiser cette fonction pour qu'elle s'effectue tous les jours à l'heure arbitraire de 10:00. En effet, nous avons remarqué que le fichier *.csv* de l'OMS se mettait à jour environ toutes les semaines, mais une exécution du programme chaque semaine pouvait nous amener à avoir jusqu'à six jours de retard sur les données actuelles, ce qui ne correspondait pas avec la dynamique du *dashboard* que nous voulions proposer.

Concernant l'élaboration du *dashboard*, nous avons tous voulu participer au codage. Nous avons alors créé un répertoire *GIT* avec plusieurs branches de développement afin de faciliter le travail collaboratif, ainsi chacun pouvait voir l'avancée des travaux de ses camarades et les utiliser pour mieux faire correspondre ses propres travaux.

La conception du *dashboard* et sa disposition se sont d'abord imaginées sur papier afin de ne pas se précipiter aveuglément dans le code. Une fois établi ce que nous voulions avoir comme résultat final pour répondre aux différentes questions que nous nous sommes posées, nous nous sommes répartis les éléments du *dashboard* que chacun avait à coder : diagramme circulaire de la part des tests positifs et négatifs de la semaine actuelle, top dix des pays ayant recensé le plus grand nombre de cas d'infections virales, carte interactive de la répartition des cas d'infections virales dans le monde, graphique de l'évolution des différents recensement d'infections virales en fonction des pays et des virus, et enfin le nombre total de cas reportés sur l'année en cours avec le classement des dix pays les plus touchés. Pour chacun des éléments, nous avons utilisé plusieurs bibliothèques *Python* dont *Dash*, *Plotly* et *Datetime*.

Il était également important pour nous d'informer les utilisateurs du *dashboard* quant à son utilisation : en effet, nous avons pris connaissance des limites de notre *dashboard* décrites ci-après, comme le fait par exemple que la lecture du graphique de l'évolution des virus n'était pas optimale, et que l'information quant au possible zoom sur le graphique n'était pas correctement indiquée. Nous avons alors choisi d'inclure un bouton "*How does it work ?*" dans l'en-tête de notre *dashboard* qui ouvre une fenêtre modale indiquant la description du *dashboard*, comment l'utiliser et ce qu'il est possible de faire. Cela avait également pour objectif d'éviter à l'utilisateur non-habitué de se diriger vers le *GitHub* pour lire le *README*.

Plusieurs réunions ont également eu lieu entre nous pour discuter des avancées, se répartir les nouvelles tâches, et être sûrs que nous allions ensemble vers un objectif commun. Une fois les codes de chaque élément du *dashboard* établis ou presque aboutis, nous avons discuté ensemble du style à adopter pour présenter notre *dashboard* et mettre en avant les éléments que nous trouvions les plus

importants. Nous avons également discuté de divers petits détails mais non des moindres comme la police d'écriture et sa couleur, les principales couleurs, les tailles des éléments... Nous avons pour cela créé une feuille de style .css pour faciliter la mise en place du style.

Enfin, une fois notre *dashboard* abouti, il nous fallait le déployer sur un serveur pour le rendre accessible à tous. Pour cela, nous avons opté pour cPanel et son gestionnaire de fichier où l'ensemble de notre dossier se trouve. Nous avons alors configuré une "application *Python*" pour permettre l'hébergement du *dashboard* ainsi que le fichier *scraping.py*, mettant à jour les données directement dans le fichier *data\_flunet.csv* présent dans le gestionnaire de fichier. Plusieurs fichiers annexes ont dû être ajoutés pour permettre l'exécution et l'hébergement du *scraping* et du *dashboard*, comme *passenger\_wsgi.py*, *app.wsgi* ou encore *requirements.txt*.

Pour finir notre projet et le conclure, nous nous sommes répartis la rédaction du *README* disponible sur le *GitHub* du projet pour décrire notre *dashboard*, ainsi que la rédaction du présent rapport de projet.

### III. Résultats

Le *dashboard* final offre une interface interactive permettant aux utilisateurs d'explorer les tendances épidémiologiques des infections respiratoires virales à travers plusieurs visualisations :

Nous avons tout d'abord une répartition des tests positifs et négatifs représentée par un diagramme circulaire, et qui présente le taux de positivité par pays pour la semaine en cours.

Nous avons ensuite un classement des dix pays les plus touchés représenté par un diagramme en barres, qui met en évidence les pays ayant enregistré le plus grand nombre de cas au cours de la dernière semaine.

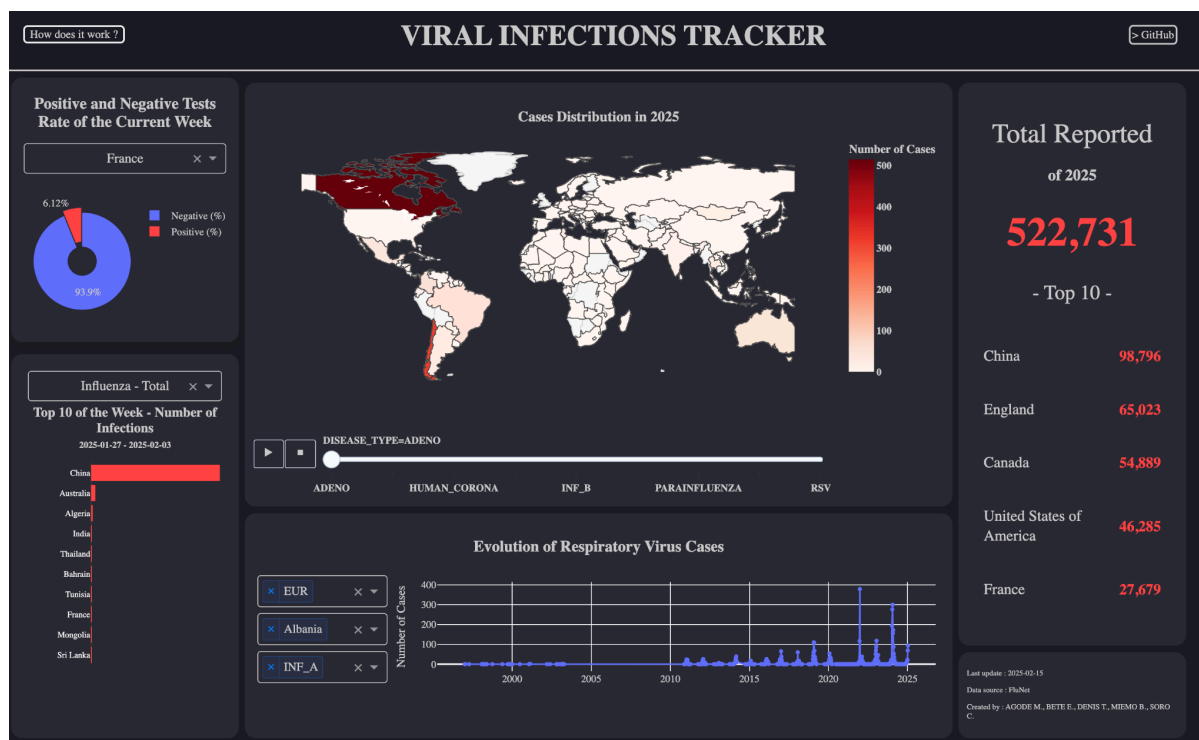
Le *dashboard* présente également une carte choroplète qui illustre la répartition géographique des infections en fonction du nombre de cas recensés. L'utilisation d'un dégradé de couleur rouge-blanc facilite l'identification des régions les plus affectées.

Se trouve également une évolution temporelle des infections représentée par un graphique linéaire, qui permet de comparer l'évolution des infections dans plusieurs pays et pour différents types de virus.

Nous avons enfin le nombre total de cas recensés pour l'année en cours, ainsi que le classement des pays les plus touchés et leur nombre de cas pour cette même année.

Grâce à ces visualisations, le *dashboard* permet aux chercheurs, décideurs et au grand public d'explorer les dynamiques de propagation des virus respiratoires et d'identifier les tendances épidémiologiques en temps réel.

Le *dashboard* a donc la visualisation voulue ci-dessous, disponible à l'adresse suivante : <http://virustracker.financeforge.fr/>. Le lien vers le *dashboard* n'est pas sécurisé, il faut donc, pour le moment, autoriser l'accès avec votre navigateur.



## IV. Discussion

Le développement de ce *dashboard* a permis de relever plusieurs défis techniques et méthodologiques. L'un des principaux enjeux a été la gestion des données issues de l'OMS, qui nécessitait une standardisation en raison des différences de nomenclature entre les pays et des formats de données utilisés. Cette phase de prétraitement des données s'est avérée essentielle pour garantir l'exactitude et la cohérence des visualisations.

Un autre défi majeur concernait l'automatisation du *web-scraping*. L'intégration d'un script permettant de mettre à jour les données quotidiennement a nécessité une réflexion sur la fréquence d'exécution et la gestion des ajouts incrémentaux sans écraser les données précédentes. La planification de l'extraction à une fréquence quotidienne a permis d'assurer que le dashboard reste constamment à jour avec les dernières informations de l'OMS, tout en évitant des surcharges inutiles.

D'un point de vue technique, nous avons rencontré des limitations liées aux bibliothèques de visualisation utilisées (*Dash* et *Plotly*). Si ces outils offrent une grande flexibilité, nous avons dû adapter certains graphiques pour améliorer leur lisibilité, notamment pour la carte choroplèthe et le graphique d'évolution des infections. Une autre contrainte a concerné la sélection dynamique des pays et virus, car les performances de l'interface peuvent être affectées par la grande quantité de données à afficher simultanément.

De plus, une de nos principales limites, et non des moindres, a été le déploiement du *dashboard* sur un hébergeur, ainsi que l'automatisation du *web-scraping* sur un serveur. Voulant un hébergeur gratuit pour que tout le monde puisse y avoir accès sans payer mensuellement, nous avons opté pour plusieurs solutions qui présentaient chacune une nouvelle limite à chaque fois : limite de taille des fichiers, période d'essai gratuite, pas d'automatisation du *web-scraping*... Nous avons alors

essayé sur *Plomber*, *AWS*, *Render*, *Railway*, *Heroku*... avant de finalement opter pour *cPanel*, service payant mais qui avait déjà été payé par l'un d'entre nous. Là encore, plusieurs limites se sont imposées à nous quant au déploiement, puisqu'il a fallu créer nombre de fichiers annexes pour permettre l'hébergement, ce qui dépassait légèrement nos capacités et les expériences que nous avions déjà eu.

Enfin, l'aspect collaboratif du projet a été optimisé grâce à l'utilisation de *Git* et *GitHub*. L'adoption d'une approche basée sur des branches de développement a permis une gestion efficace du code et a facilité la fusion des contributions de chaque membre de l'équipe.

## V. Conclusion

Ce projet a permis de développer un outil interactif et informatif, facilitant l'analyse des tendances des infections respiratoires virales à l'échelle mondiale. Grâce à l'automatisation des mises à jour et à une interface intuitive, le *dashboard* offre une vue d'ensemble claire et accessible des données épidémiologiques.

Toutefois, certaines limites subsistent. Le manque de granularité des données de l'OMS pour certaines régions peut affecter la précision des visualisations, et l'absence d'intégration avec d'autres sources de données (comme les hospitalisations ou la mortalité) limite l'analyse des impacts sanitaires des infections virales.

Pour les perspectives d'amélioration, nous pourrions envisager : l'intégration d'analyses prédictives basées sur des modèles statistiques ou d'intelligence artificielle pour anticiper les tendances, un enrichissement des sources de données, en croisant les informations de l'OMS avec d'autres bases de données épidémiologiques.

Ce projet constitue en somme une première étape vers un outil de veille épidémiologique efficace, contribuant à une meilleure compréhension et prévention des infections respiratoires virales à l'échelle mondiale.