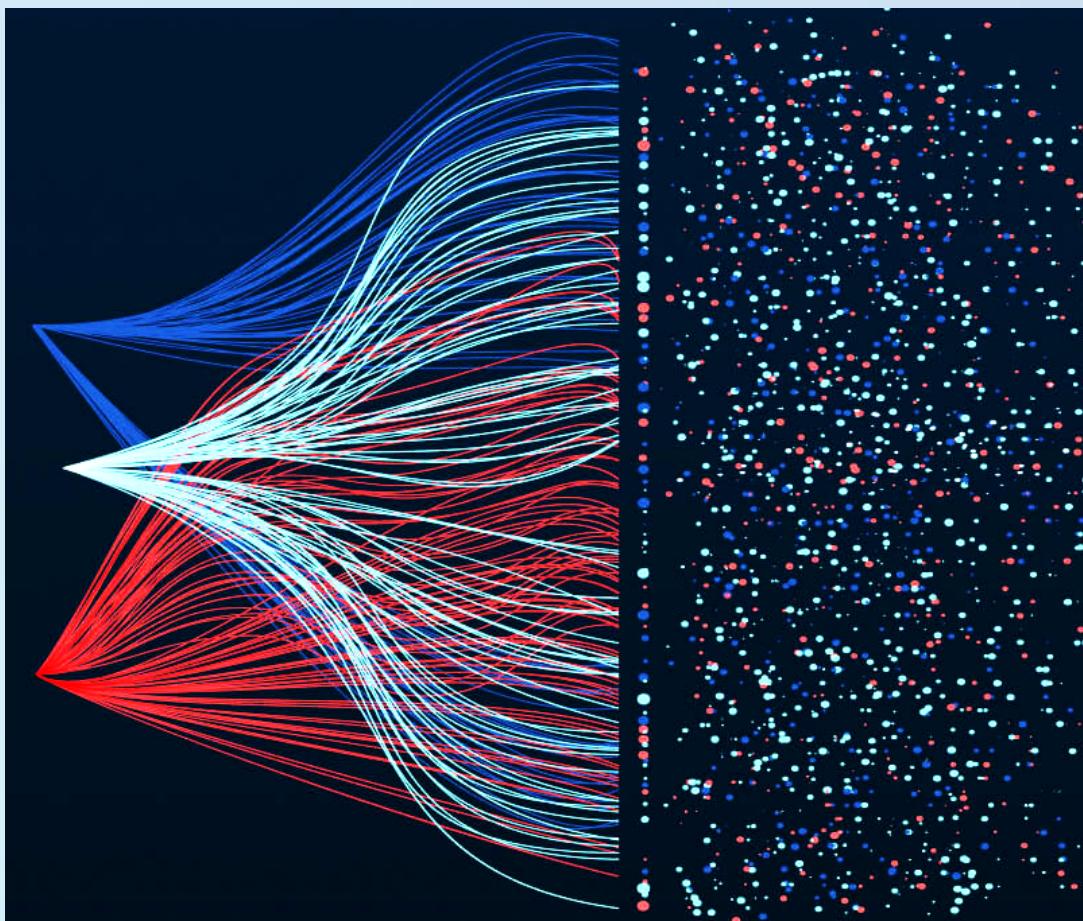


# **RAPPORT DE PROJET**

Outil d'augmentation de la donnée  
**L3Q1-2022**



# **RAPPORT DE PROJET**

## **PROJET DATA AUGMENTATION**

### **Identification du document**

**Référence du document : L3Q1.F**

**Version du document : 1.0**

**Date du document : 16/04/2023**

#### **Auteurs :**

Abou Assaf Mahyr-Florian

Daumont-Ouk Ilan'

Ngo Alexandre

Chemla Ethan

# Sommaire

<b>1. Remerciements</b>	<b>4</b>
<b>2. Contexte</b>	<b>5</b>
<b>3. Introduction</b>	<b>5</b>
<b>4. Objectif</b>	<b>6</b>
<b>5. Gestion du projet</b>	<b>6</b>
5.2 L'équipe	6
5.2 La planification	7
5.3 Outil de gestion	9
5.4 Organisation interne	10
5.5 Chef de projet	10
5.6 Environnement de développement et de production	10
<b>6. Développement</b>	<b>11</b>
6.1 Lexique	11
6.2 Architecture logicielle	12
6.3 Technologie Back-end	14
6.4 Technologie Front-end	15
6.5 Description des modules	16
6.6 Affichage pour l'utilisateur	17
6.7 Les difficultés rencontrées	19
<b>7. Annexes</b>	<b>19</b>

# 1. Remerciements

Nous tenons tout d'abord à remercier chaleureusement notre encadrant M. Elie El Debs, qui a su être à notre écoute, répondre à nos interrogations et garder un cadre tout au long du projet afin que nous restions rigoureux et vigilant face aux délais impartis. Nous remercions l'UFR Paris Cité et son responsable M. David Janiszek, pour nous avoir donné l'opportunité d'étendre nos compétences et connaissances en informatique.

Grâce à ce module, nous avons pu apprendre à travailler en groupe, améliorer notre communication afin de se répartir les tâches et donc à avancer plus vite et de manière efficace. De plus, travailler sur un sujet auquel toutes les grandes entreprises font face actuellement a été enrichissant et sera sans doute un levier pour la suite de nos études en Master.

## 2. Contexte

Ce projet est réalisé dans le cadre de l'Unité d'Enseignement fondamentale " Projet tutoré " de l'Université Paris Cité d'Informatique. Quatre étudiants sont sous la tutelle d'un encadrant auquel ils font un compte rendu hebdomadaire sur l'avancement de leur projet. Ils doivent également rendre différents documents afin de faciliter la compréhension du projet et son éventuelle utilisation par un tiers.

Lors d'une soutenance orale organisée en mai, les étudiants présentent à la fin des 12 semaines de travail, le projet devant un jury, composé de M. Janiszek, notre encadrant M. Elie El Debs, et au moins un autre enseignant.

## 3. Introduction

L'augmentation de la donnée ou data augmentation en anglais, est un processus de collecte et d'enrichissement des données réelles, provenant de diverses sources afin de créer des informations plus globales, utiles et importantes pour la prise de décision. En augmentant les ensembles de données existants avec de nouvelles informations les entreprises peuvent mieux comprendre leurs clients, leurs marchés, leurs concurrents et leurs opérations.

Cela peut également s'inscrire dans une optique de donnée dirigée pour améliorer la qualité des résultats des modèles d'apprentissage automatique, en l'occurrence cela sera la direction prise pour notre projet.

L'augmentation de la donnée permet aux entreprises d'accroître leur capacité à extraire et intégrer efficacement des volumes importants de données à partir de sources variées afin d'obtenir une image pertinente et proche du monde réel pour alimenter leurs projets d'intelligence artificielle.

Il est important de noter que ces données obtenues ne sont pas réelles, il s'agit seulement de données synthétiques.

De nombreuses zones géographiques sont concernées par la data augmentation, comme l'Amérique du Nord, l'Europe, et l'Asie. Ce sont généralement les secteurs de la Tech, de la finance ou du médical qui l'utilisent. Les sociétés utilisant des outils de data augmentation sont nombreuses, comme Amazon Web Service, IBM Watson, Google Query, ou encore Azure Machine Learning. Mais il y a également des librairies open-source telles que TensorFlow ou PyTorch. À noter l'arrivée de nouvelles technologies d'intelligence artificielle telles que DALL-E 2, OpenAI, JasperAI, WriteSonic, etc... qui insufflent un nouvel élan à l'industrie de l'IA.

## 4. Objectif

Dans le cadre de ce projet nous proposons de développer une application web capable de collecter un ensemble de données d'images, puis de générer des données légèrement modifiées de manière quantitative. Ces méthodes de génération de données se feront au moyen de l'implémentation d'algorithmes d'augmentation de la donnée du côté back-end tels que le bruit, la luminosité, le contraste, le recadrage, le flopping, le zoom.

L'objectif est de retourner à l'utilisateur un ensemble de données augmenté d'images, ainsi beaucoup plus important que celui initialement fourni à l'application. Il pourra par la suite télécharger son dataset augmenté et/ou le supprimer de son espace personnel de l'application web. Notre projet vise ainsi à aider les entreprises à améliorer leurs modèles d'Intelligence Artificielle en leur fournissant un dataset plus significatif et en réduisant les coûts liés à la collecte de données.

## 5. Gestion du projet

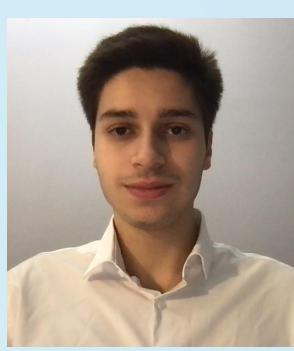
La gestion de projet permet de planifier au mieux le bon déroulement de ce dernier, avec la façon la plus optimale possible.

### 5.2 L'équipe

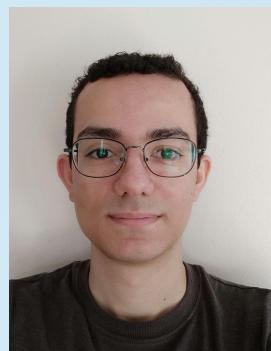
Dans ce projet encadré par M. Elie El Debs consultant en informatique chez CGI, l'équipe L3Q1 est composée de 4 étudiants en 3ème année de Licence Informatique à l'Université Paris Cité :



**NGO  
Alexandre**



**CHEMLA  
Ethan**



**ABOU ASSAF  
Mahyr-Florian**



**DAUMONT-OUK  
Ilan'**

## 5.2 La planification

Le projet sur l'augmentation des données s'étend sur une durée de conception et de développement de 12 semaines. Cela inclut des réunions hebdomadaires avec notre encadrant pour structurer notre avancement, mais aussi le rendu régulier de documents, tels que l'étude de marché, le cahier des charges, le cahier de recettes, le plan de tests, l'architecture logicielle et les manuels d'installation & d'utilisation pour permettre au client la prise en main du projet réalisé.

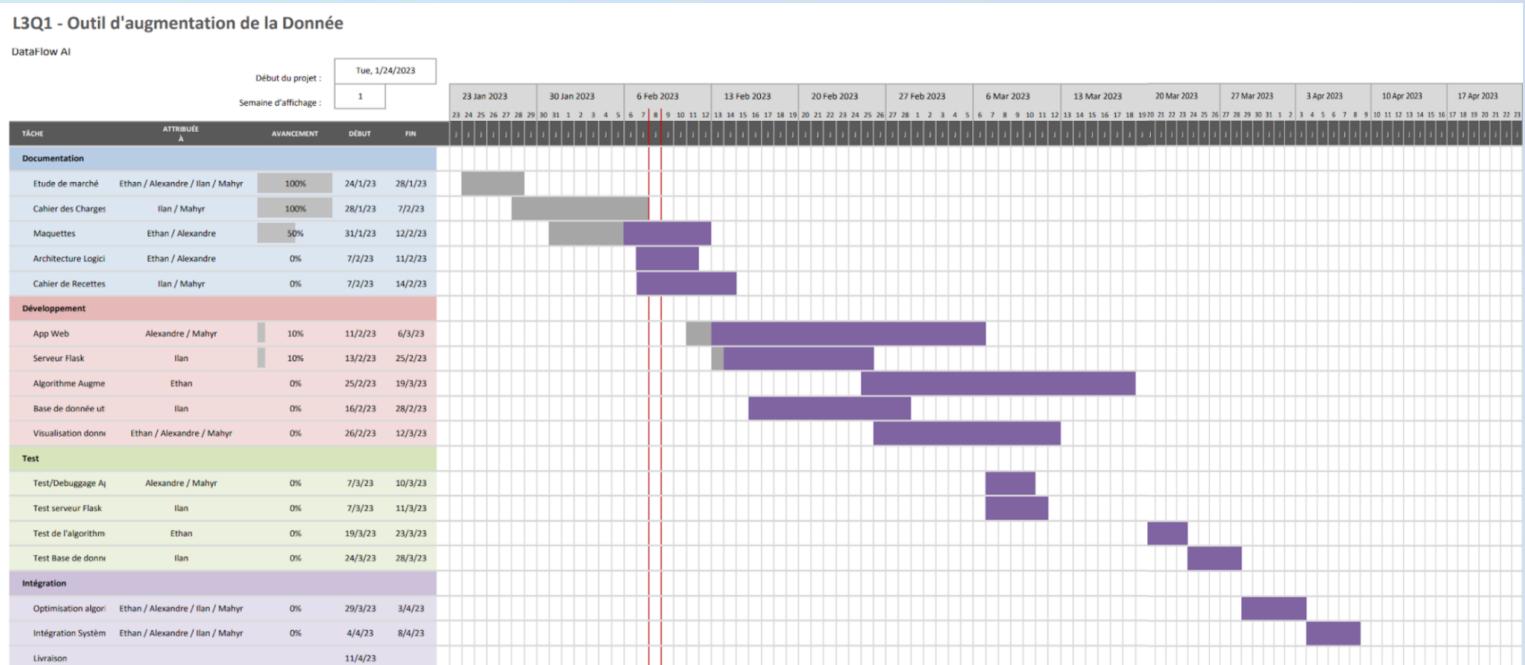


Diagramme de Gantt sur la planification du projet

Ce diagramme de Gantt présente les différents documents rendus, ainsi que le temps passé à les rédiger. Par la suite, nous décrirons les documents que nous avons édité.

Nous avons passé plusieurs jours à faire des recherches liées à cet outil et à comprendre le marché actuel dans le monde, lié à la data augmentation. S'en est suivi la rédaction des différentes fonctionnalités que nous trouvons pertinentes après discussion avec l'encadrant, mais également de mettre en œuvre les différents tests qui feront appel aux principaux concepts utilisés. L'architecture logicielle, un document clé pour le passage de la phase de conceptualisation du projet à l'implémentation de nos idées, celui-ci, comme tous les autres documents, nous ont permis de simplement et rapidement traduire nos concepts en code fonctionnel. Finalement, nous avons proposé des manuels qui contiennent pas à pas les étapes à suivre pour l'installation et l'utilisation de l'application web.

## Documents réalisés :

**Étude de marché** (semaine 1) : permet de mieux connaître sur quels sujets est utilisée la data augmentation, de cibler les offres concurrentes et d'évaluer la demande actuelle sur le marché mondiale.

**Cahier des charges** (semaine 2) : permet d'avoir une vue globale sur la façon dont le projet va être mené, avec la description de la demande, les contraintes et l'établissement de notre travail en tant que programmeurs.

**Cahier de recettes** (semaine 4) : permet de lister les scénarios à vérifier tout au long du projet et lors de la soutenance.

**Maquette** (semaine 4) : première visualisation du rendu de notre application et de l'agencement de ses différentes pages web.

**Cahier de tests** (semaine 4) : définit les tests effectués afin de valider les différentes fonctionnalités du programme sous ses angles multiples.

**Conception détaillée** (semaine 5) : permet de garantir une bonne compréhension des aspects techniques et une description complète de l'architecture de l'application.

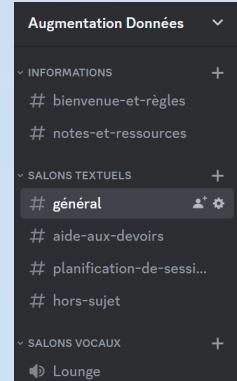
**Manuel d'installation** (semaine 11) : décrit les étapes de déploiement de l'application sur la data augmentation pour la mettre en place sur un ordinateur avec la base de données.

**Manuel d'utilisation** (semaine 11) : décrit l'ensemble des fonctionnalités de l'application web, et des actions à réaliser pour l'utiliser.

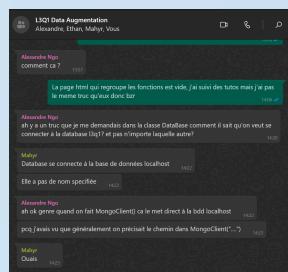
## 5.3 Outil de gestion

Les principaux outils de gestion de l'organisation, de l'environnement de travail ainsi que l'espace de partage des documents sont les suivants :

**Discord** : Logiciel d'appel et de partage des ressources entre étudiants, cette application permet la création de différents salons de discussions en fonction des ressources partagées. Nous avons créé deux salons de discussions vocaux pour la répartition des tâches en groupe de deux lors des phases de conception et de développement, ainsi que cinq salons de discussions textuels.



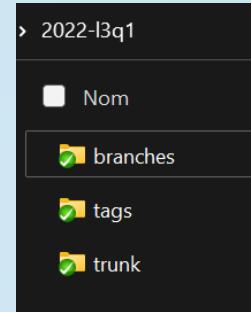
**WhatsApp** : Application mobile qui permet de s'informer et de recevoir les notifications plus rapidement sur nos téléphones / ordinateurs. Nous avions également le choix de pouvoir faire un appel de groupe.



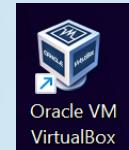
**Forge** : Environnement de partage et de communication avec notre encadrant. Tous les documents tels que les cahiers, les plans et les manuels ont pu être envoyés sur la Forge. Les chefs de projet avaient pour consigne de communiquer sur le forum dédié avec l'encadrant, notamment sur les comptes rendus de chaque réunion.



**SVN** : Environnement de partage de la partie informatique, les scripts python, java et les paramètres du dashboard ont pu être communiqués dans un dossier "branches" lieu où le code est envoyé par groupe (selon les groupes et leurs tâches) puis un dossier "trunk" où nous avons pu envoyer la finalité des dossiers avec les scripts bien ordonnancés et le projet fonctionnel.



**Oracle VM VirtualBox** : Environnement de travail virtuel. Toutes les installations, les paramétrages et le développement du code ont pu être générés sur un système d'exploitation Linux.



## **5.4 Organisation interne**

En dehors du temps des réunions hebdomadaires avec notre encadrant M. Elie El Debs, l'équipe des quatre étudiants s'est réuni chaque soir - ou lendemain - de réunion afin de rédiger le compte rendu. De plus, nous commençons à se répartir le travail hebdomadaire à effectuer. Nous nous réunissions également au moins deux fois par semaine pour se mettre à jour sur l'avancement de la conception ou du développement à réaliser, ainsi que les possibles difficultés rencontrées pour échanger et trouver une solution ensemble.

## **5.5 Chef de projet**

Toutes les trois semaines, un étudiant est désigné en tant que chef de projet qui se voit attribuer le rôle d'organisateur au sein de l'équipe et d'envoyer le compte-rendu sur la forge. Celui-ci avait également pour devoir de gérer les communications avec l'encadrant. En fin de projet, cette responsabilité basculait à une semaine par personne afin d'atteindre une répartition équitable des responsabilités sur 12 semaines.

## **5.6 Environnement de développement et de production**

Pour le développement de notre projet, nous avons utilisé une machine virtuelle Kali Linux. Cela nous a permis de tous travailler sur le même environnement et de rester en accord avec le cahier des charges établi en début de projet. Nous avons bien entendu testé ce que nous avons développé directement sur l'application web qui sera livrée le jour de la soutenance.

# 6. Développement

## 6.1 Lexique

**Application web:** Programme ou logiciel qui est accessible depuis différents navigateurs web.

**Data Cleaning :** Ensemble de processus ayant pour but d'améliorer la qualité des données en corrigeant les données inexactes dans un ensemble de données.

**Dataset :** Ensemble de données cohérentes qui peuvent se présenter sous différents formats (textes, chiffres, images, audios, vidéos...).

**Données non structurées :** Données non organisées en base de données, c'est-à-dire la messagerie, les images, les vidéos, etc...

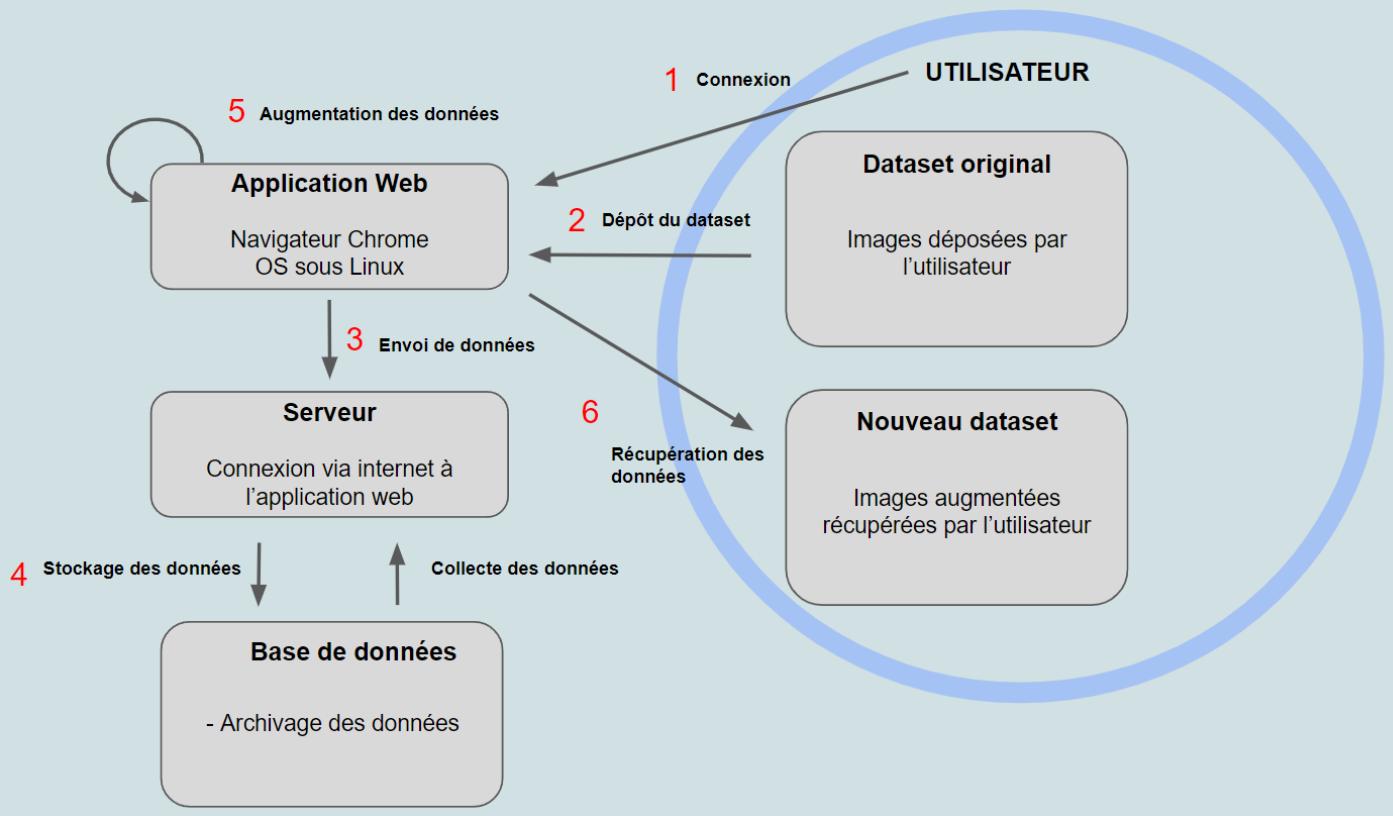
**Machine learning :** Le Machine Learning ou apprentissage automatique est un domaine scientifique, et plus particulièrement **une sous-catégorie de l'intelligence artificielle**. Elle consiste à laisser des algorithmes découvrir des “patterns” , à savoir des motifs récurrents, dans les ensembles de données. Ces données peuvent être des chiffres, des mots, des images, des statistiques...

**Deep learning :** Il s'agit d'un type de machine learning, version plus améliorée. L'apprentissage profond utilise une technique lui conférant une aptitude supérieure à détecter les patterns même les plus subtiles.

**Python :** Langage de programmation orienté objet, notamment très utile dans le domaine de l'intelligence artificielle.

**Flask :** Permet de créer des applications web en Python. Flask est un framework web léger qui offre une grande flexibilité et une facilité de développement pour la création d'applications web.

## 6.2 Architecture logicielle



L'architecture en question peut être divisée en deux grandes catégories :

- Le côté utilisateur
- Le côté back-end et front-end de l'application web

Concernant le côté utilisateur, le client se connecte à l'application web via un navigateur tel que Google Chrome (1). Il accède à son espace personnel où il peut déposer son ensemble de données d'images (2) et le configurer. L'ensemble de données déposé est directement transmis à la base de données MongoDB via un serveur local pour être stocké. L'utilisateur accède ensuite à une autre page web où il sélectionne les configurations souhaitées pour son futur ensemble de données augmenté (5). Les données sont automatiquement récupérées et téléchargées sur son ordinateur (6). L'utilisateur peut également télécharger son ensemble de données augmenté ultérieurement.

Le client est représenté par le navigateur web Google Chrome, qui permet à l'utilisateur d'interagir avec l'application en naviguant sur les différentes pages et en réalisant diverses actions. Le serveur, quant à lui, héberge l'application et fournit les ressources nécessaires pour son bon fonctionnement. Les requêtes HTTP sont

employées pour assurer la communication entre le client et le serveur, avec des requêtes POST et GET pour transmettre les données.

Le middleware constitue l'interface entre les différentes couches de l'application, garantissant la transmission des données aux parties appropriées du système et gérant les erreurs et exceptions. L'API fournit une interface standardisée pour accéder aux différentes fonctionnalités de l'application, composée de modules contenant des fonctions atomiques, c'est-à-dire des fonctions ne renvoyant qu'un seul résultat pour une entrée donnée.

La stratification facilite la maintenance et l'évolution de l'application en permettant de modifier une couche sans toucher aux autres. Elle favorise également la collaboration entre les différents développeurs responsables des diverses parties de l'application. Les fonctions atomiques simplifient le code et facilitent la maintenance en autorisant la réutilisation des fonctions dans différentes sections de l'application.

Le serveur gère les demandes des clients et fournit les réponses appropriées. À chaque fois qu'un utilisateur souhaite effectuer une action, comme déposer un ensemble de données ou appliquer un algorithme d'augmentation de données, une requête est envoyée au serveur qui effectue les actions nécessaires et renvoie une réponse. La base de données joue un rôle crucial dans l'application car elle permet de stocker les ensembles de données des utilisateurs ainsi que les informations d'authentification et de connexion.

Le processus de dépôt d'un ensemble de données débute par l'authentification de l'utilisateur. Les informations d'authentification sont conservées dans la base de données et servent à vérifier l'identité de l'utilisateur. Si l'utilisateur est authentifié avec succès, il est redirigé vers son espace personnel. Dans cet espace, l'utilisateur peut déposer son ensemble de données en envoyant une requête HTTP au serveur. Le serveur reçoit les données et les stocke dans la base de données. Une fois l'ensemble de données stocké, l'utilisateur peut appliquer les algorithmes d'augmentation de données de son choix en sélectionnant les options correspondantes. Le serveur reçoit les choix de l'utilisateur et applique les algorithmes sélectionnés.

Les algorithmes d'augmentation de données sont gérés par des scripts Python qui communiquent avec le serveur pour fournir les données nécessaires à l'application et effectuer les traitements requis. Une fois que les algorithmes d'augmentation de données ont été appliqués, l'utilisateur peut télécharger l'ensemble de données augmenté depuis son espace personnel.

En somme, cette architecture en plusieurs couches permet non seulement de faciliter la maintenance et l'évolution de l'application, mais aussi de favoriser la collaboration entre les différents développeurs responsables des diverses parties de

l'application. De plus, elle garantit une expérience utilisateur fluide et sécurisée, en assurant la gestion des demandes et des réponses entre le client et le serveur, ainsi que la sécurisation des informations d'authentification et de connexion.

### 6.3 Technologie Back-end

La partie back-end d'un site web constitue un élément essentiel de l'architecture d'une application web, étant donné qu'elle assume la responsabilité de la gestion des données et de la logique de l'application. Cette composante de l'architecture prend en charge la gestion des bases de données, l'authentification des utilisateurs, la gestion des sessions, la sécurité de l'application et la manipulation des données circulant entre le client et le serveur. Dans ce contexte, il est primordial de sélectionner les technologies les plus appropriées afin d'assurer un rendement optimal et une expérience utilisateur de qualité.

Pour la gestion du serveur, le choix s'est porté sur Flask, un framework open-source écrit en Python qui facilite la construction d'applications web. Flask propose une structure de base pour le développement d'applications web en utilisant le protocole HTTP et autorise la création de serveurs web pour des applications web grâce au langage de programmation Python. Cette solution présente de multiples avantages, notamment sa flexibilité et sa facilité d'utilisation, qui en font une option privilégiée pour des applications web de petite et moyenne envergure.

Lorsqu'un utilisateur soumet une requête HTTP, le serveur Flask écoute cette requête à une adresse IP et un port spécifiques, puis la traite avant de renvoyer une réponse HTTP. Cette architecture permet de garantir une expérience utilisateur fluide et réactive, tout en assurant la sécurité des données transitant entre le client et le serveur.

En résumé, le choix de Flask comme framework pour la gestion du serveur constitue une décision judicieuse, car il offre une solution solide, flexible et facile à utiliser pour le développement d'applications web. Cette technologie permettra d'assurer des performances optimales et une expérience utilisateur satisfaisante pour les utilisateurs du site web.

### 6.4 Technologie Front-end

Le front-end d'un site web se réfère à l'interface utilisateur visible et interactive, qui est cruciale pour une expérience utilisateur optimale. Il est constitué principalement de trois langages de programmation, à savoir HTML, CSS et JavaScript, qui permettent de façonner la structure, l'apparence et la fonctionnalité d'une page web.

Le langage de balisage HTML est utilisé pour organiser le contenu de la page en établissant la hiérarchie des divers éléments tels que les titres, les paragraphes, les images, les vidéos, les formulaires et les tableaux. Le langage de style CSS, quant à lui, est employé pour déterminer l'aspect visuel de la page web, en appliquant des règles de style spécifiques aux éléments concernés, contrôlant ainsi leur apparence et leur agencement. En l'absence de CSS, la page HTML présenterait uniquement un contenu brut, dépourvu de mise en forme.

Le langage de programmation JavaScript sert principalement à insuffler de l'interactivité et de la dynamique aux pages web grâce à la création de scripts exécutés côté client. Il permet de manipuler le contenu et le comportement de la page, tout en capturant et en gérant les événements utilisateur, tels que les clics de souris et les pressions de touche, ajoutant ainsi une couche de dynamisme à une page HTML statique.

Le code front-end est exécuté côté client, c'est-à-dire dans le navigateur de l'utilisateur. Il peut comprendre une multitude d'éléments interactifs tels que des formulaires, des boutons, des menus, des images, des vidéos et des animations. L'emploi du langage CSS est indispensable pour conférer un attrait visuel et une esthétique soignée à ces éléments, permettant un contrôle total sur la présentation de la page web. Sans CSS, la page HTML consisterait en une simple collection de textes et d'images sans mise en forme.

Afin de développer efficacement la partie front-end d'un site web, la maîtrise de ces trois langages est primordiale. De surcroît, des frameworks tels que Bootstrap facilitent le développement en proposant des modèles de conception et des éléments prédéfinis pour construire des pages web de manière rapide et aisée. Le front-end est d'une importance capitale pour la conception de sites web contemporains, car il offre aux utilisateurs la possibilité d'interagir avec les informations présentées de manière ergonomique et accessible.

## 6.5 Description des modules

Notre application web dédiée à l'augmentation de données se compose de plusieurs modules, chacun ayant une fonction spécifique.

**La page d'accueil** présente notre application et offre un accès aisément à diverses pages telles que la page de connexion, la page d'inscription, la page de contact et la page de présentation.

**La page d'inscription** comporte un formulaire permettant la création d'un espace personnalisé pour chaque utilisateur. Les données entrées dans ce formulaire sont transmises au serveur, qui les traite et met à jour la base de données pour les nouveaux comptes utilisateurs.

**La page de connexion** s'adresse aux utilisateurs existants, leur permettant d'accéder à leur compte en saisissant leur adresse e-mail, leur nom, leur prénom et leur mot de passe. La page de réinitialisation du mot de passe offre la possibilité de rétablir le mot de passe d'un utilisateur en cas d'oubli.

**L'espace personnel** est conçu pour permettre aux utilisateurs de déposer des images et de personnaliser leur apparence. Cette page comprend plusieurs boutons pour modifier divers aspects de l'image, tels que la luminosité, le contraste, le bruit, le zoom et le recadrage.

**La page de visualisation** des images présente l'ensemble des ensembles de données d'images déposées, qu'elles aient été augmentées ou non.

**La page de documentation** fournit des informations détaillées sur les fonctionnalités de notre site et les traitements applicables aux ensembles de données.

Enfin, **la page présentant l'équipe de développeurs** permet aux utilisateurs de mieux connaître les personnes responsables du développement de notre application. Pour se déconnecter, les utilisateurs peuvent simplement cliquer sur le bouton "Se déconnecter" situé sur la page de déconnexion.

## 6.6 Affichage pour l'utilisateur

L'application web se présente comme ci-dessous :

La page d'accueil :

The screenshot shows the DataFlow AI application's landing page. At the top, there is a navigation bar with links for 'Dashboard', 'Data Augmentation', 'Documentation', 'Team', and a 'Log In' button. Below the navigation is a large, artistic profile of a person's head and shoulders, rendered in green and blue, with data points and lines emanating from it, symbolizing AI and data flow. To the left of the profile, the 'DataFlow AI' logo is displayed, with the tagline 'Amplifiez vos données, découpez vos performances'. A brief welcome message explains the purpose of the platform: to help users augment their datasets quickly and effectively using AI models. Below the message are three buttons: 'S'inscrire' (Sign Up), 'Ou' (Or), and 'Se connecter' (Log In). A section titled 'Pourquoi utiliser DataFlow AI ?' (Why use DataFlow AI?) contains two columns of text. The left column discusses the evolution of AI and ML, noting that while they play a key role, obtaining reliable results often requires a large dataset. The right column describes the application's mission to assist users in easily augmenting their datasets using advanced methods. At the bottom of the page, there is a footer with the DataFlow AI logo and the text 'Amplifiez vos données, découpez vos performances.', followed by logos for Université Paris Descartes and Université de Paris.

**La page de connexion :**

The screenshot shows a login form titled "Identification". At the top left is a link "Retour à la page d'accueil". The form contains fields for "Adresse Mail" (with "nom@example.com" entered) and "Mot de Passe" (with "Mot de passe" entered). There is a checked checkbox "Se souvenir de moi" and a blue "Se connecter" button. Below the form are links for password recovery ("Mot de passe oublié ?") and registration ("Vous n'avez pas de compte ? S'inscrire").

**La page d'inscription :**

**La page de mot de passe oublié :**

**La page espace personnel :**

**La page dashboard :**

**La page de dépôt des datasets d'images :**

**La page de documentation :**

**La page de contact :**

**La page de présentation de l'équipe :**

## 6.7 Difficultés rencontrées

L'un des principaux problèmes concernait l'interface web, qui n'était initialement pas conçue pour être responsive.

Cette situation a nécessité une réécriture du code HTML pour assurer une expérience utilisateur optimale sur différents écrans.

Nous avons également rencontré des difficultés lors de la mise en place du routage dans le serveur Flask, nécessitant une révision et un déboggage minutieux pour garantir un fonctionnement correct.

L'intégration de la base de données MongoDB a également posé des problèmes, notamment en termes de configuration et de communication avec le serveur, qui ont dû être résolus pour assurer la cohérence et la stabilité de l'application.

Enfin, nous avons dû remplacer le serveur Werkzeug par Gevent pour passer à un serveur asynchrone utilisant le module Socket.IO, afin de permettre aux utilisateurs de continuer à interagir avec l'application pendant le processus d'augmentation des données. Cela a notamment permis la mise en place de la barre de progression avec l'application web qui communique en continu avec le serveur flask durant l'augmentation des données.

## 6.8 Améliorations :

Concernant les améliorations à apporter à notre application, plusieurs aspects pourraient être considérés pour optimiser davantage ses performances et son efficacité.

Tout d'abord, il serait judicieux d'ajouter de nouvelles techniques d'augmentation de données afin de couvrir un éventail plus large de cas d'utilisation et de répondre aux besoins diverses de l'entraînement de modèles d'intelligence artificielle dans différents domaines et applications.

De plus, l'intégration d'un système d'apprentissage automatique pour adapter automatiquement les paramètres d'augmentation en fonction des besoins spécifiques de chaque utilisateur pourrait permettre d'améliorer la personnalisation et l'expérience utilisateur.

En outre, la mise en place d'un système de suivi des performances et d'analyse de l'impact des augmentations de données sur la qualité des modèles entraînés contribuerait à une meilleure compréhension de l'efficacité des méthodes utilisées.

De plus, nous souhaiterions intégrer notre modèle de machine learning dans l'application, permettant ainsi aux utilisateurs de tester directement leur donnée augmentées et vérifier leur résultats.

Enfin, élargir les domaines de données augmentées pris en charge par notre application permettrait d'attirer un public plus large et de s'adapter aux besoins de diverses industries et domaines d'application.

## 7. Annexes

Introduction générale à la data augmentation :

<https://neovision.fr/data-augmentation-solutions-manque-donnees/>

[https://en.wikipedia.org/wiki/Data\\_augmentation](https://en.wikipedia.org/wiki/Data_augmentation)

<https://blog.datumize.com/the-five-most-common-data-quality-issues-and-how-to-overcome-them>

<https://www.cnil.fr/fr/definition/augmentation-de-donnees>

<https://lbourdois.github.io/blog/nlp/Data-augmentation-in-NLP/>

Présentation machine learning et deep learning :

<https://datascientest.com/machine-learning-tout-savoir>

[https://inside-machinelearning.com/pourquoi-et-comment-normaliser-ces-donnees-pytorch-une-etape-essentielle-du-deep-learning-partie-1/#Normaliser\\_les\\_donnees](https://inside-machinelearning.com/pourquoi-et-comment-normaliser-ces-donnees-pytorch-une-etape-essentielle-du-deep-learning-partie-1/#Normaliser_les_donnees)