



# Projet de fin d'études

*Bloc 3 : Elaborer et piloter un projet data*

## LogiOps360

Outil intelligent de supervision de la chaîne logistique

**Etudiante** : Milissa ZEBOU DJ

**Formation** : Mastère Data Scientist à Paris Ynov Campus

**Diplôme** : Expert en science des données spécialité Data Scientist

**Année universitaire** : 2024/2025

**Entreprise (fictive)** : LogiOps



**LogiOps  
360**

## Table des matières :

1. Le cadrage du projet
  - a. Contexte et problématique
  - b. Présentation de LogiOps
  - c. Objectifs et livrables du projet
  - d. Le cadre réglementaire et enjeux RSE
  - e. Contraintes et solutions
2. Le dimensionnement du projet
  - a. Ressources nécessaires
  - b. Chiffrage du projet
  - c. Analyse de la faisabilité
3. Documentation du projet
4. Méthodologie et planning
5. Outils pour le pilotage du projet
6. Indicateurs de performances
7. Management d'équipe
  - a. Constitution et animation de l'équipe
  - b. Plan de développement des compétences
  - c. Gestion des risques
8. Veille, éthiques et gouvernance des données
  - a. Méthodologie de veille
  - b. Réglementation et normes
  - c. Plan de gouvernance des données
9. Synthèse
10. Annexes

# 1. Le cadrage du projet :

## a. Contexte et problématique :

La chaîne logistique industrielle est aujourd'hui au cœur des enjeux de performance et de compétitivité des entreprises. Longtemps considérée comme un simple levier d'exécution, la logistique est désormais perçue comme un facteur stratégique, capable d'influencer à la fois les coûts, les délais, la satisfaction client et l'empreinte environnementale. Cette évolution s'est accélérée au cours des dernières années sous l'effet de plusieurs facteurs.

D'une part, les entreprises industrielles font face à une grande complexité logistique : multiplication des fournisseurs, diversification des canaux de distribution, exigences en matière de traçabilité, d'optimisation des stocks et de réactivité. Les processus doivent être pilotés de manière agile, fiable et en temps réel, ce qui nécessite des outils modernes et flexibles.

Par ailleurs, les enjeux environnementaux (réduction de l'empreinte carbone, logistique durable, économie circulaire) viennent ajouter une nouvelle dimension au pilotage logistique. Les industriels doivent non seulement être efficaces, mais aussi responsables.

Dans ce contexte, un constat s'impose : malgré l'abondance des données générées dans la chaîne logistique (commandes, livraisons, stocks, incidents, délais), ces données sont souvent dispersées, dans des outils hétérogènes (ERP, tableurs Excel, emails, outils transport) et sous-exploitées. De nombreuses entreprises ne disposent pas d'outils décisionnels centralisés permettant de visualiser et d'anticiper les événements logistiques de manière dynamique.

L'intelligence artificielle et la valorisation des données jouent un rôle de plus en plus central. L'IA permet non seulement d'automatiser certaines tâches (comme la prévision de la demande ou la détection d'anomalies), mais aussi d'optimiser les décisions complexes dans un environnement incertain. Grâce aux technologies de machine learning, il devient possible de prédire les retards de livraison, d'anticiper les saturations d'entrepôt ou de recommander le transporteur le plus performant selon le contexte. Combinée à une architecture data robuste, l'IA transforme ainsi la logistique traditionnelle en une logistique prédictive et intelligente au service de la performance globale de l'entreprise.

La problématique peut ainsi être formulée de la manière suivante :

Comment permettre à une entreprise industrielle de mieux superviser ses flux logistiques, d'identifier les points de blocage, de prendre des décisions plus rapides et plus éclairées, et de prédire les perturbations, à partir des données qu'elle possède déjà ?

Le projet LogiOps s'inscrit dans cette problématique : il vise à proposer une solution concrète, innovante et accessible pour accompagner les entreprises industrielles dans la centralisation et l'optimisation de leur pilotage logistique.

## b. Présentation de LogiOps :

C'est dans ce contexte de transformation digitale de la supply chain que s'inscrit LogiOps, une startup française fondée par des professionnels issus des secteurs de la logistique industrielle et de la data science. Sa vocation est simple mais ambitieuse : mettre la puissance des données et de l'intelligence artificielle au service d'une logistique plus réactive et plus durable, accessible même aux entreprises qui ne disposent pas d'un système complexe ou d'équipes data spécialisées.

LogiOps développe une solution logicielle appelée LogiOps360, une plateforme de supervision logistique intelligente conçue pour aider les entreprises industrielles à reprendre le contrôle de leurs opérations, depuis les fournisseurs jusqu'au dernier kilomètre. L'outil permet de centraliser toutes les données critiques (commandes, stocks, transport, etc.), de les visualiser sous forme de tableaux de bord dynamiques, de détecter les anomalies et surtout de produire des recommandations fondées sur des modèles prédictifs.

En résumé, LogiOps propose bien plus qu'un outil de reporting : c'est un copilote logistique intelligent, conçu pour transformer la donnée brute en décisions opérationnelles, dans un monde où chaque heure compte et chaque retard peut se transformer en perte de chiffre d'affaires.

## c. Objectifs et livrables :

L'objectif principal de ce projet est de développer une démonstration fonctionnelle de la solution LogiOps360, en s'appuyant sur des jeux de données réels issus de la logistique et du transport. Il s'agit de modéliser un système de supervision logistique cohérent, couvrant la chaîne de valeur de bout en bout, de la réception de commandes jusqu'à la livraison, en passant par le stockage et le suivi de performance opérationnelle.

Grâce aux données à disposition, issues de contextes logistiques variés (commandes clients, modes de transport, affectation d'entrepôts, historique d'expéditions), le projet vise à restituer un environnement réaliste dans lequel la plateforme LogiOps360 pourra apporter une valeur métier concrète. L'approche consiste à centraliser, structurer et croiser ces données pour permettre une analyse globale, une visualisation claire des flux, et une interprétation intelligente des indicateurs clés.

### Objectifs principaux :

- Centraliser les différentes sources de données logistiques disponibles : commandes clients, stockage, données de transport et séries temporelles de performance.
- Concevoir un tableau de bord interactif regroupant les modules suivants :
  - Suivi des commandes clients : statut, quantité, date de création, affectation.
  - État du stockage par zone et type d'allocation.
  - Analyse des expéditions et des moyens de transport utilisés.
  - Évolution temporelle de la performance logistique (volumes, délais).

- Mettre en place des indicateurs clés de performance : taux de ponctualité des livraisons, délai moyen de traitement, taux de saturation des zones de stockage, nombre de commandes en attente.
- Implémenter un modèle de prédiction des retards de livraison à partir des données disponibles.
- Simuler un environnement métier réaliste à travers la création de profils utilisateurs afin de démontrer la valeur métier du tableau de bord pour un responsable logistique ou un chef d'entrepôt.

Livrables attendus :

- Une interface de visualisation présentant les modules de suivi des commandes, de stockage et de transport.
- Des scripts Python documentés permettant le traitement des données, le calcul des indicateurs et la génération de visualisations.
- Un modèle prédictif permettant d'estimer le risque de retard sur les livraisons.
- Un modèle de prédiction de la saturation d'un type de stockage
- Une documentation projet complète : choix techniques, méthode de pilotage, limites, et propositions d'amélioration.

Ce projet constitue à la fois une preuve de concept technique de la plateforme LogiOps360, et un exercice de structuration d'un projet data logistique du point de vue fonctionnel, technique et métier.

#### d. Le cadre réglementaire et enjeux RSE :

Dans le cadre du développement de la solution LogiOps360, deux dimensions structurantes doivent être prises en compte dès la phase de conception : le respect des réglementations applicables en matière de données et les enjeux environnementaux et sociétaux associés à la logistique.

##### Le cadre réglementaire : protection des données et conformité

Même si les données utilisées dans le cadre de ce projet sont issues de sources ouvertes ou anonymisées, la conception de la plateforme LogiOps360 s'inscrit dans une logique de conformité au Règlement Général sur la Protection des Données (RGPD). Dans une version déployée en entreprise, la solution serait amenée à manipuler des données sensibles (clients, transporteurs, localisations, historiques de commandes, etc.). Il est donc essentiel d'anticiper dès aujourd'hui les bonnes pratiques en matière de :

- Sécurisation des données à caractère personnel (authentification, droits d'accès, chiffrement si nécessaire).
- Minimisation des données collectées et traitées (principe de « privacy by design »),
- Traçabilité des traitements effectués sur les données (logs, historique des modifications),
- Documentation des flux de données et des finalités associées.

Par ailleurs, toute intégration de la solution dans un système d'information existant devra respecter les règles internes de l'entreprise en matière de cybersécurité, de sauvegarde, de traçabilité et d'architecture logicielle.

#### Les enjeux RSE :

La logistique est un poste majeur d'impact environnemental pour les entreprises industrielles : elle représente une part significative des émissions de gaz à effet de serre, des coûts énergétiques et du gaspillage lié à la surproduction ou aux ruptures. Dans ce contexte, LogiOps360 se positionne comme un outil d'aide à la décision permettant de concilier performance opérationnelle et responsabilité environnementale.

En fournissant une meilleure visibilité sur les flux, des indicateurs de performance précis et des leviers d'optimisation, la solution permet notamment de :

- Réduire les kilomètres parcourus inutilement (en optimisant les tournées ou en anticipant les ruptures)
- Diminuer les stocks dormants ou en surcapacité (via la visualisation des taux de saturation),
- Mieux répartir la charge entre les sites et les entrepôts pour limiter les transports intersites

À terme, un module spécifique dédié au suivi des émissions carbone liées au transport pourrait être intégré à la plateforme, en lien avec les objectifs de décarbonation de la supply chain fixés par les entreprises et les régulations européennes.

Ainsi, bien que réalisé dans un contexte académique, le projet LogiOps360 a été pensé dès le départ pour être conforme aux exigences futures en matière de protection des données, de cybersécurité et de responsabilité numérique.

### e. Contraintes techniques et temporelles :

Le projet LogiOps360, bien que fictif dans sa mise en œuvre, repose sur des conditions réalistes de développement, proches de celles rencontrées dans un contexte professionnel. Sa réussite nécessite de prendre en compte un certain nombre de contraintes, qu'elles soient techniques, temporelles, réglementaires ou liées au périmètre du projet.

#### Contraintes techniques :

L'une des premières contraintes techniques réside dans la nature des données utilisées. Le projet repose sur des jeux de données hétérogènes, publics ou simulés, provenant de différentes sources. Ces données sont parfois incomplètes, non normalisées ou sujettes à des erreurs (doublons, colonnes manquantes, formats incohérents), ce qui nécessite un important travail de préparation (nettoyage, standardisation, agrégation) avant toute phase d'analyse.

De plus, certains éléments du système cible ne peuvent pas être reproduits intégralement dans le cadre de ce projet, comme l'intégration en temps réel de données issues de capteurs IoT, de GPS de camions ou d'ERP propriétaires. Le tableau de bord développé fonctionnera donc en environnement local et non en production cloud, ce qui limite certains cas d'usage comme les alertes en temps réel ou l'automatisation complète des flux.

Enfin, le choix technologique doit rester cohérent avec les ressources disponibles : Python, Streamlit, pandas, scikit-learn, PostgreSQL donc des outils open source, accessibles et adaptés à un

développement académique, mais limités pour le déploiement industriel à grande échelle sans adaptation supplémentaire.

### **Contraintes temporelles :**

Le projet s'inscrit dans un horizon de 3 mois, ce qui permet une organisation en plusieurs phases distinctes : cadrage, développement, tests et documentation. Ce délai reste court au regard de la complexité potentielle d'un projet logistique intégrant des modules d'analyse, de prédiction et de visualisation. Il impose donc une forte rigueur dans la planification et la priorisation des fonctionnalités.

Cette durée permet néanmoins d'aller au-delà d'un simple prototype statique : plusieurs itérations peuvent être réalisées, des tests d'évaluation des modèles peuvent être mis en place, et l'interface peut être structurée pour simuler une version quasi professionnelle de la solution LogiOps360.

### **Contraintes réglementaires :**

Même si le projet ne traite pas directement de données personnelles identifiables, certaines exigences réglementaires doivent être prises en compte, notamment si la solution venait à être utilisée dans un cadre professionnel réel.

- Respect du RGPD : bien que les données soient anonymisées, la conception du projet suit les principes du « privacy by design » :
  - Aucune donnée nominative n'est traitée,
  - Pseudonymisation des identifiants clients ou transporteurs,
  - Structuration des données par usage limité et légitime.
- Sécurité des systèmes : le projet anticipe les exigences de base en matière de cybersécurité :
  - Conception modulaire avec séparation des données,
  - Structure compatible avec une gestion d'accès différenciée par rôle (utilisateur, superviseur)
  - Stockage local simulé pouvant évoluer vers des solutions souveraines.

### **Contraintes budgétaires (simulation) :**

Aucune dépense n'est prévue dans le cadre de ce projet : tous les outils utilisés sont open source, et les données sont publiques. En entreprise, un tel projet représenterait un investissement conséquent, à la fois en développement (3 à 6 mois), en infrastructure (hébergement sécurisé), en maintenance et en formation. Le coût d'un pilote pour une ETI pourrait être estimé entre 25 000 € et 60 000 €, selon la complexité des flux à modéliser.

### **Contraintes liées au périmètre :**

Pour garantir la faisabilité, le projet a été volontairement recentré sur trois modules principaux :

- Commandes clients,
- Gestion des stocks par zone,

- Suivi des expéditions et délais.

Des dimensions complémentaires comme l'optimisation des tournées, la visualisation des flux en temps réel ou l'intégration carbone ont été mises de côté dans cette version. Elles pourront faire l'objet d'évolutions dans une phase 2.

## 2. Le dimensionnement du projet :

### a. Ressources nécessaires :

Dans le cadre du développement de la solution LogiOps360, la réussite du projet repose sur une mobilisation adaptée des ressources humaines, matérielles et logistiques, en cohérence avec les objectifs visés et les contraintes identifiées en amont. Cette section vise à dimensionner le projet de manière concrète, sans encore aborder la question du coût, afin d'assurer la cohérence entre charge de travail, ressources disponibles et délais.

#### **Ressources humaines :**

Le projet mobilise une équipe technique répartie entre deux grands pôles :

- Pôle ingénierie des données (data engineering) :
  - 2 data engineers assurent la collecte, le nettoyage, la normalisation, la structuration et la mise en forme des données logistiques issues de sources multiples. Ils développent les pipelines d'ingestion, automatisent les traitements et assurent la robustesse des jeux de données.
- Pôle modélisation & analyse (data science) :
  - 2 data scientists interviennent sur l'analyse exploratoire des données, la production des indicateurs avancés, la mise en place des visualisations dynamiques et l'entraînement de modèles de machine learning (prédiction de retard, saturation entrepôt, charge future).
- Chef de projet (Project Owner) : Il assure le cadrage fonctionnel du projet, la planification des étapes, l'animation des points d'équipe, la gestion des risques.

L'ensemble de l'équipe travaille en coordination selon une logique agile, avec des points de synchronisation réguliers et des objectifs par sprint. En cas de déploiement commercial, l'équipe pourrait être renforcée par un(e) UX designer, un(e) chef de projet ou un(e) intégrateur API.

#### **Ressources matérielles :**

La plateforme LogiOps360 repose sur une stack technique volontairement légère, adaptée à un développement rapide et décentralisé. Les ressources matérielles nécessaires sont les suivantes :

- Postes de travail équipés d'un environnement Python (Jupyter, terminal, éditeur de code)
- Connexion internet fiable pour le téléchargement de jeux de données et le partage de code
- Serveur local (simulé en local pour cette version MVP) pour exécuter l'application finale



- Disque dur avec capacité suffisante pour stocker les fichiers CSV/XLSX (entre 500 Mo et 2 Go)

Dans une version professionnelle, ces ressources seraient complétées par un hébergement cloud sécurisé, un accès API aux données réelles des clients, et un système de gestion des utilisateurs.

### **Ressources logistiques :**

La gestion du projet est organisée en mode hybride (présentiel / distanciel), avec des outils collaboratifs permettant la coordination fluide entre membres de l'équipe. Aucun besoin physique particulier n'est requis, hormis :

- Une infrastructure de partage cloud (GitHub).
- Un calendrier partagé pour les sprints et les revues de fin de phase.
- Des templates standardisés pour la documentation, le suivi qualité, les indicateurs.

## **b. Chiffrage du projet :**

L'objectif est de garantir la faisabilité du projet dans un cadre opérationnel, tout en respectant les contraintes identifiées (délai, périmètre, réglementation, disponibilité des données et outils). Cette estimation est menée à partir d'un scénario de développement en environnement startup technologique, dans une logique de MVP industrialisable.

### **Évaluation de la charge de travail :**

Le développement du noyau fonctionnel de LogiOps360 s'étale sur un cycle de 6 mois, correspondant à une première version stable, modulaire et démontrable auprès de clients industriels. Ce délai permet non seulement de concevoir un MVP, mais également d'engager une phase de consolidation, de tests utilisateurs et d'itérations produit.

Le projet est découpé en 7 phases successives, chacune mobilisant l'ensemble de l'équipe projet (chef de projet, data engineers et data scientists), selon une approche agile. Ce découpage permet de structurer le travail de manière itérative, tout en garantissant une montée en complexité progressive, une qualité technique soutenue et une capacité d'adaptation aux retours métier.

<b>Sprint</b>	<b>Phase</b>	<b>Durée estimée</b>	<b>Activités principales</b>
<b>Sprint 1</b>	Phase 1 : Etude de faisabilité et recherche de données	3 semaines	Exploration du marché, identification des sources de données, évaluation des contraintes techniques
<b>Sprint 1</b>	Phase 2 : Cadrage et planification du projet	3 semaines	Définition du périmètre, rédaction du cahier des charges, choix méthodologique et planification prévisionnelle

<b>Sprint 2</b>	Phase 3 : Développement du module 1 (Commandes)	3 semaines	Pipeline de traitement des commandes, visualisations, indicateurs clients
<b>Sprint 2</b>	Phase 4 : Développement du module 2 (Stock)	3 semaines	Structuration des données de stockage, alertes de saturation, Dashboard dédié
<b>Sprint 2</b>	Phase 5 : Développement du module 3 (livraison)	3 semaines	Intégration des données d'expédition, cartes et performances
<b>Sprint 3</b>	Phase 6 : Modélisation prédictive & intégration	5 semaines	Prédiction des retards, intégration dans l'interface, tests croisés
<b>Sprint 4</b>	Phase 7 : Documentation et déploiement	3 semaines	Tests utilisateurs, documentation complète, insertion des livrables, préparation soutenance

#### Estimation des jours-hommes :

Le projet LogiOps360 s'étale sur une période de 6 mois (environ 24 semaines). Sur cette durée, l'équipe projet est mobilisée de manière régulière, selon un rythme estimé à 3 jours par semaine et par personne. Ce volume permet à l'équipe de fonctionner en mode itératif tout en laissant place à la recherche, aux ajustements techniques et aux itérations fonctionnelles.

- Chef de projet (x1) : 2 j/semaine × 24 semaines = 48 jours-hommes
- Data engineers (x2) : 3 j/semaine × 24 semaines × 2 personnes = 144 jours-hommes
- Data scientists (x2) : 3 j/semaine × 24 semaines × 2 personnes = 144 jours-hommes

→ Charge totale estimée : 336 jours-hommes

Cette estimation inclut : le cadrage fonctionnel, le développement technique des modules, la construction des modèles prédictifs, la création des visualisations, les tests internes, la rédaction de la documentation, ainsi que la gestion du backlog et des itérations produit.

#### Évaluation budgétaire :

Dans un contexte d'entreprise, une estimation budgétaire peut être faite en se basant sur les TJM (tarifs journaliers moyens) observés pour des profils freelances ou consultants spécialisés.

Ressource	TJM estimé	Jours	Coût estimé
<b>Chef de projet</b>	500 €/j	48	24 000 €
<b>Data engineers (x2)</b>	400 €/j	144	57 600 €
<b>Data scientists (x2)</b>	450 €/j	144	64 800 €

⇒ Coût total estimé (hors hébergement cloud) : 146 400 € HT

Ce budget correspond à la conception et au développement complet d'un MVP opérationnel, incluant les trois modules métiers (commandes, stock, livraisons), les modèles prédictifs, une interface interactive prête à être présentée à des prospects, ainsi qu'une documentation technique et fonctionnelle. Il ne couvre pas encore l'industrialisation complète du produit (intégration API client, hébergement cloud, déploiement multi-utilisateurs, certification RGPD, support client), qui ferait l'objet d'un investissement complémentaire.

#### **Faisabilité :**

La planification sur 6 mois permet de maîtriser la complexité croissante du projet sans compromettre la qualité. Grâce à une équipe complémentaire, une stack technique éprouvée et une organisation agile, la solution LogiOps360 peut être livrée dans le délai imparti avec un niveau de professionnalisme et de maturité suffisant pour lancer une première phase de commercialisation pilote auprès de clients industriels.

### **3. Documentation du projet :**

Dans une logique de structuration professionnelle et conforme aux bonnes pratiques en gestion de projet data, LogiOps a mis en place une documentation claire, centralisée et adaptée à ses parties prenantes. Celle-ci a pour objectif de formaliser les attentes, de faciliter la collaboration entre les différents profils impliqués (techniques et non techniques), et de garantir la traçabilité de toutes les décisions, évolutions et fonctionnalités du projet LogiOps360.

La documentation est rédigée dans un vocabulaire accessible, à la fois pour les profils techniques (data engineers, data scientists), que pour les profils fonctionnels (chefs de projet, responsables logistiques, utilisateurs finaux). Elle est structurée autour de trois livrables principaux, en cohérence avec le cadrage établi lors de la phase 1.

#### **1. Cahier des charges fonctionnel (Annexe 1):**

Ce document présente les besoins métiers, les cas d'usage visés, les modules attendus (suivi des commandes, supervision des stocks, visualisation des expéditions) ainsi que les utilisateurs ciblés (responsable logistique, superviseur entrepôt, planificateur transport). Il clarifie également les indicateurs clés à mettre à disposition dans l'outil (taux de retard, taux de saturation, délai moyen, etc.). Ce cahier des charges permet d'aligner l'équipe sur les finalités de la solution LogiOps360 et les attentes fonctionnelles réelles.

#### **2. Spécifications techniques (Annexe 2) :**

Les spécifications techniques détaillent l'architecture de la solution, les jeux de données exploités, la structure des fichiers, les flux de traitement de données, ainsi que les choix technologiques (Python, scikit-learn, PostgreSQL). Un schéma du pipeline de données est joint dans le rapport du bloc 1 afin d'explicitier les étapes de traitement. Chaque module technique est documenté par un README clair et un diagramme de structure ou de flux.

#### **3. Charte éthique et gouvernance des données (Annexe 3) :**

Dans un souci de responsabilité, une charte éthique a été rédigée pour encadrer l'usage des données au sein de la plateforme. Elle rappelle l'importance de la protection des données, le

respect des principes RGPD (anonymisation, minimisation, traçabilité), et les choix réalisés pour garantir la conformité du projet, même en environnement simulé. Cette charte comprend également une première ébauche de plan de gouvernance des données applicable dans une phase de déploiement client réel.

La documentation projet est donc pleinement en adéquation avec le cadrage défini en amont. Elle permet à l'équipe de maintenir une vision claire et partagée, tout en offrant aux parties prenantes un support structuré pour comprendre, tester, ou évaluer la solution développée.

## 4. Méthodologie et planning :

Pour structurer le projet LogiOps360, j'ai choisi une approche agile hybride, inspirée de la méthode Scrum, mais adaptée à un contexte de projet technique orienté data. Cette méthodologie me permet de découper le projet en phases logiques tout en intégrant de la flexibilité à travers des sprints définis par blocs de travail fonctionnels.

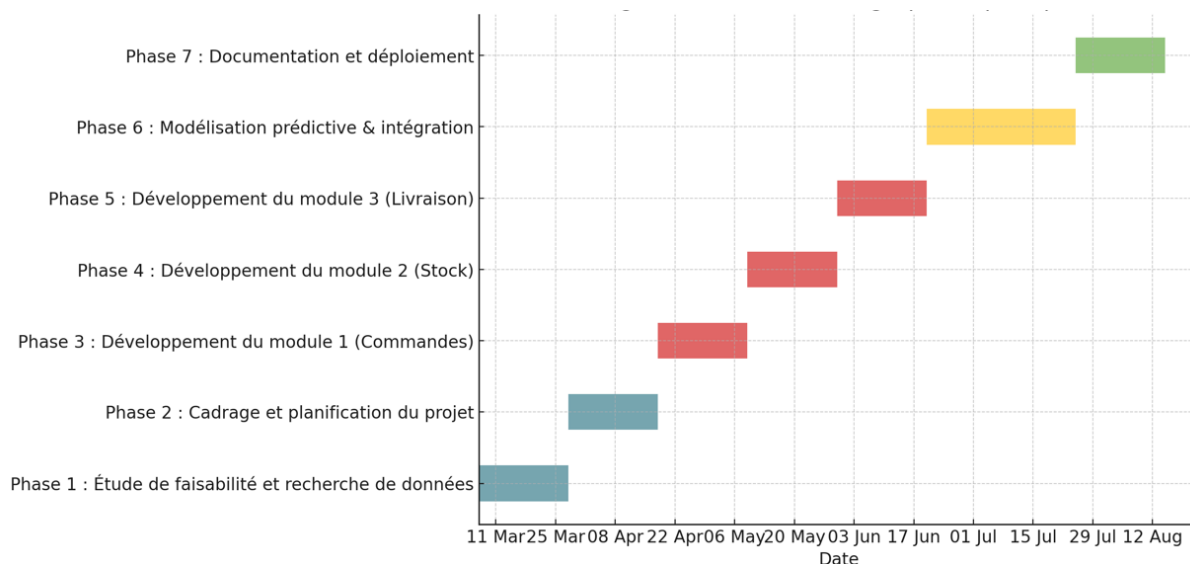
Dès le départ, j'ai souhaité ne pas me précipiter dans la partie technique. Le projet a donc été découpé en quatre grands sprints. Chaque sprint regroupe une ou plusieurs phases du projet, avec des objectifs clairs, des livrables attendus, et une logique de montée en complexité progressive. Cette structuration m'a permis d'alterner conception, développement et validation de manière fluide tout au long des six mois de réalisation.

Voici la structure retenue :

- Sprint 1 : Étude de faisabilité + cadrage du projet (6 semaines)
- Sprint 2 : Développement des modules métiers (commandes, stock, livraisons) – 9 semaines
- Sprint 3 : Modélisation prédictive & intégration dans l'interface – 5 semaines
- Sprint 4 : Documentation, tests utilisateurs et préparation de la soutenance – 3 semaines

Chaque sprint a été planifié avec un objectif principal et des tâches internes plus détaillé (répartis sur Trello). J'ai utilisé Trello pour le suivi des tâches quotidiennes, avec des colonnes de type Kanban (À faire / En cours / Terminé), et Notion pour structurer la documentation fonctionnelle, le backlog des indicateurs, et le suivi des décisions

J'ai également réalisé un diagramme de Gantt global pour visualiser l'enchaînement des sprints et des phases. Ce Gantt m'a permis de vérifier la cohérence du calendrier, d'anticiper les points critiques (comme l'entraînement des modèles prédictifs ou l'intégration technique), et d'ajuster les charges de travail en fonction des délais.



*Diagramme de Gantt pour la planification des phases par sprint  
Projet LogiOps360 – Février à Août 2025*

Enfin, une matrice RACI a été utilisée pour répartir les rôles dans l'équipe projet (deux data engineers, deux data scientists, un chef de projet). Bien que nous n'ayons pas eu de contrainte liée au handicap dans l'équipe, j'ai veillé à rendre les interfaces accessibles (Dashboard simple, navigation par clavier, etc.), et à documenter le projet de manière claire, accessible à tout nouveau collaborateur.

	Chef de projet	Data engineer 1	Data engineer 2	Data scientist 1	Data scientist 2
Étude de faisabilité	A	R	R	C	C
Cadrage & cahier des charges	A/R	C	C	I	I
Traitement des commandes (Module 1)	I	R	R	A	C
Traitement des stocks (Module 2)	I	R	R	C	A
Livraison & cartes (Module 3)	I	C	R	A	R
Modélisation prédictive	I	C	C	R/A	R
Intégration & interface	A	R	R	C	C
Documentation finale & soutenance	A/R	I	I	C	C

### Matrice RACI

- R – Responsable : réalise concrètement la tâche
- A – Accountable : valide et porte la responsabilité
- R/A – Responsable et valideur (double rôle)
- C – Consulted : consulté pour expertise
- I – Informed : informé de l'avancement

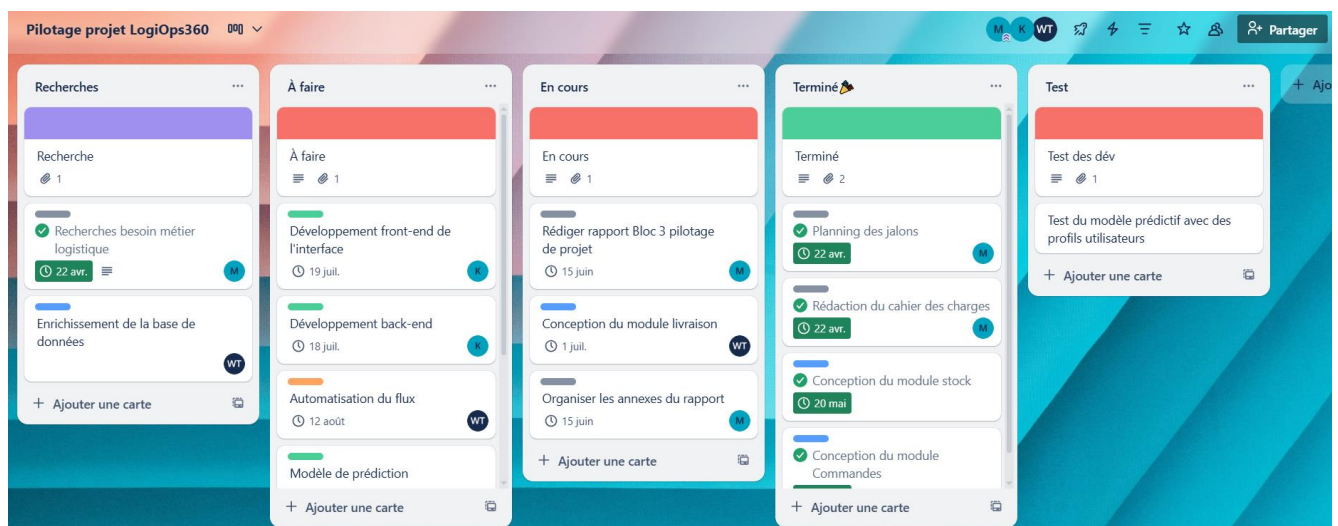
L'approche agile m'a permis en tant que chef de projet de rester réactive, de produire un livrable exploitable dès les premiers mois, et d'itérer jusqu'à une version stable et prête à être présentée à un client ou un jury professionnel.

## 5. Outils pour le pilotage du projet :

Afin d'assurer une gestion efficace et structurée du projet LogiOps360, j'ai eu recours à plusieurs outils numériques qui m'ont permis de suivre l'avancement, d'organiser le travail en équipe, de documenter les décisions prises et de gérer les différentes itérations du projet.

### Gestion des tâches : Trello

Pour piloter l'exécution des tâches au quotidien, j'ai utilisé Trello. Ce tableau Kanban a été divisé en colonnes représentant les différents états d'avancement. Cet outil m'a permis de prioriser les actions à mener, de suivre la progression de chacun et d'avoir une vue globale sur l'état du projet.



### Documentation : Notion

J'ai utilisé Notion comme outil centralisé de documentation tout au long du projet. Il m'a permis de conserver une trace claire de chaque étape : les objectifs de chaque sprint, le cahier des charges fonctionnel et les comptes rendus de réunions internes. Cet espace collaboratif a facilité la communication entre les membres de l'équipe et a servi de support pour structurer les livrables finaux.

### Suivi du code et versioning : GitHub

L'ensemble du code source du projet a été géré via un dépôt GitHub. Chaque fonctionnalité a été développée dans une branche dédiée. Cette méthode m'a permis de versionner le projet proprement, d'assurer un bon suivi des évolutions du code, et de collaborer efficacement avec les autres membres de l'équipe. Un fichier README est associé à chaque module pour expliquer son fonctionnement.

**Planification** : diagramme de Gantt

Un diagramme de Gantt a été construit en début de projet afin de planifier les phases et les sprints sur une durée totale de six mois. Il permet de visualiser l'enchaînement des tâches.

**Répartition des responsabilités** : matrice RACI

Pour clarifier les rôles de chacun dans l'équipe, j'ai établi une matrice RACI. Cette matrice m'a permis de mieux organiser la collaboration au sein du groupe et d'éviter les confusions.

Dans l'ensemble, cette combinaison d'outils m'a permis d'avoir une gestion de projet structurée et collaborative du projet LogiOps360.

## 6. Les indicateurs de performances (KPIs) :

Des indicateurs de pilotage de projet ont été définies pour mener à bien le projet LogiOps360 :

**Indicateurs par jalon :**

Jalon	Indicateur de suivi	Type	Fréquence
Étude de faisabilité	Taux d'avancement des recherches métier	Quantitatif	Hebdo
Cadrage et planification	Cahier des charges	Qualitatif	Fin phase
Développement module Commandes	Taux de tâches Trello terminées sur total du sprint	Quantitatif	Hebdo
Développement modules Stock & Livraison	Respect de la date de fin planifiée (écart en jours)	Quantitatif	À chaque fin de sprint
Modélisation prédictive	Modèle produit et testé	Qualitatif	Fin sprint
Documentation et déploiement	Intégration des livrables finaux dans le rapport (%)	Quantitatif	Hebdo

**Indicateurs transverses de pilotage**

En complément de ces indicateurs par phase, des indicateurs globaux ont été utilisés tout au long du projet :

- Taux de complétion global du projet (nombre de tâches réalisées / prévues)
- Nombre de livrables validés vs attendus
- Suivi du planning prévisionnel (écarts en jours sur les jalons clés)
- Suivi de la charge estimée vs réelle (en jours-hommes par profil)

- Nombre de points de blocage majeurs identifiés et résolus

### Suivi régulier et reporting :

Un reporting synthétique a été tenu à jour tous les quinze jours. Il reprend les indicateurs principaux, les écarts éventuels, les points de blocage rencontrés, ainsi que les décisions prises lors des réunions d'avancement. Ces suivis réguliers permettent d'anticiper certains et d'adapter la charge par profil.

En résumé, les indicateurs permettent de maintenir le cap sur les délais initiaux, et de garantir la qualité des livrables produits au fil des phases.

Le détail des réunions d'avancement (synthèses, décisions prises) est accessible via un espace Notion dédié, dont l'URL figure en annexe 4

## 7. Management d'équipe :

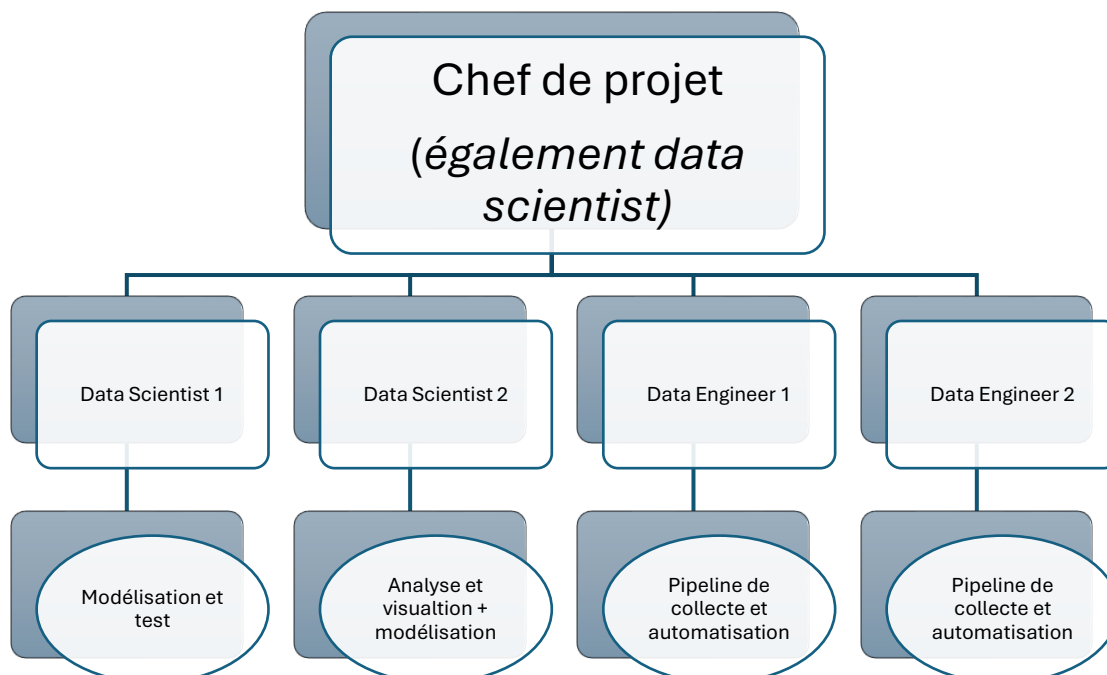
Le management d'équipe a été pensé de manière collaborative, structurée et agile dès la phase de cadrage :

### a. Constitution et animation de l'équipe :

L'équipe projet était composée de cinq membres aux profils complémentaires :

- 1 chef de projet (pilotage, organisation, suivi documentaire)
- 2 data engineers (structuration des données, pipelines, intégration)
- 2 data scientists (modélisation, analyses prédictives, indicateurs métiers)

Dès le démarrage, une matrice RACI a été construite pour répartir clairement les responsabilités de chacun sur les différentes phases du projet (cf. section précédente). Cette répartition a permis d'identifier facilement les référents sur chaque bloc technique.





Le travail collectif a été rythmé par :

- Des réunions d'équipe hebdomadaires pour faire le point sur l'avancement et ajuster les priorités.
- Des revues de sprint toutes les trois semaines pour valider les livrables réalisés et planifier les tâches suivantes.
- Une coordination via Trello et Notion, qui a facilité le suivi au quotidien et la documentation continue des décisions.

### b. Plan de développement des compétences :

Un diagnostic initial des compétences disponibles dans l'équipe a été réalisé afin d'identifier les besoins de montée en compétence nécessaires à la réussite du projet.

Sur la base des enjeux du projet, j'ai établi un plan de développement des compétences, ce plan s'est structuré autour de deux axes :

#### **1. Renforcement technique individuel :**

Chaque membre de l'équipe a été invité à suivre des formations (cours en ligne, documentation, tutoriels ciblés) en fonction de son rôle :

- Data engineers : optimisation de pipelines avec pandas, apprentissage de dbt et bonnes pratiques en ingestion de données logistiques.
- Data scientists : interprétabilité des modèles, métriques de performance pour classification déséquilibrée ainsi que l'utilisation des outils de visualisations et statistiques sur les données.
- Chef de projet : planification agile (Scrum, Kanban), conception de livrables projet (RACI, Gantt, reporting).

#### **2. Mutualisation des compétences :**

Des sessions internes de partage ont été mises en place tout au long du projet :

- Revue croisée de code entre ingénieurs et data scientists.
- Démonstrations de visualisation ou de modélisation à l'équipe entière.
- Retour d'expérience sur les difficultés rencontrées et résolution collect

### c. Gestion des risques humains et organisationnels :

Plusieurs risques ont été identifiés dès la phase de cadrage :

- Risque de désalignement entre la planification et l'exécution réelle du projet ;
- Risque de désengagement en cas de blocage technique prolongé ou de difficulté à structurer les données.

Pour y faire face, plusieurs leviers ont été mis en place :

- Une planification réaliste avec des marges entre les phases critiques.
- Des points de synchronisation réguliers permettant d'anticiper les retards.

- Une entraide active entre profils : les data scientists ont par exemple soutenu les ingénieurs dans le nettoyage, et inversement pour la phase de tests.

Enfin, même si l'équipe ne comprenait pas de personne en situation de handicap, une attention particulière a été portée à l'accessibilité du code, des interfaces, et de la documentation (structures claires, navigation simple, commentaires explicites).

#### **Exemple de réajustement en cours de projet :**

Au cours de la phase de développement du module Stock (sprint 2), l'équipe a rencontré une difficulté technique qui a nécessité un arbitrage. Lors de la tentative d'agrégation des volumes de stockage par type de produit, nous avons constaté que la structuration du fichier source ne permettait pas une correspondance immédiate avec les données de commandes. L'absence d'un identifiant commun exploitable compliquait l'analyse croisée entre ce qui était en stock et ce qui était prévu en commande.

Après un point rapide entre les deux data engineers et les data scientists, une décision a été prise :  
 → Plutôt que de forcer un joint complexe entre fichiers hétérogènes, nous avons décidé de dissocier temporairement le module Stock du module Commandes.  
 → Le module Stock a été recentré sur les taux de saturation par zone, les volumes globaux et les produits critiques, sans croisement avec les données de commande.  
 → Une mise en relation plus fine avec les commandes a été reportée à un sprint ultérieur, une fois la logique de mapping clarifiée..

En parallèle, les efforts sur le pipeline de traitement des livraisons ont été anticipés de quelques jours, afin d'optimiser la charge de l'équipe.

Cet ajustement, discuté avec d'équipe et validé collectivement, illustre notre capacité à arbitrer de manière pragmatique, sans compromettre les objectifs finaux du projet.

## 8. Veille, éthiques et gouvernance des données :

Nous avons porté une attention particulière à la qualité, à la conformité et à l'éthique du traitement des données.

### A. Méthodologie de veille :

Une veille technique et réglementaire a porté sur :

- Les outils open source pour la visualisation, le traitement des données (pandas, scikit-learn) et l'hébergement local.
- Les bonnes pratiques de gestion des flux logistiques, notamment sur la modélisation des retards.
- Les réglementations en vigueur en matière de données (RGPD, principes de minimisation des données, stockage local sécurisé).

## B. Conformité et cadre éthique :

Même si le projet repose sur des jeux de données publics, j'ai veillé à limiter les risques :

- Pas de traitement de données personnelles ou nominatives.
- Aucune remontée directe de données sensibles côté utilisateur.
- Traitement local des fichiers sans envoi vers des services tiers.

## C. Gouvernance des données :

Une gouvernance simple mais structurée a été mise en place :

- Stockage organisé des données par module dans des dossiers versionnés.
- Nommage standardisé des fichiers.
- Séparation claire entre les données brutes, transformées et utilisées en modélisation.
- Documentation technique des transformations réalisées.

L'ensemble de ces pratiques contribue à renforcer la fiabilité des analyses produites et à garantir une utilisation responsable des données, en cohérence avec les principes d'un projet professionnel respectueux des enjeux éthiques et réglementaires.

## 9. Synthèse :

LogiOps360 est un projet pensé pour répondre à un besoin réel : mieux superviser la chaîne logistique dans les entreprises industrielles. Trop de données, peu de lisibilité, des retards peu anticipés. L'outil vise à centraliser, analyser, prédire. Simple à utiliser, mais fondé sur une architecture solide.

Depuis le lancement, le projet avance par étapes. Étude de faisabilité, cadrage, premiers développements : les bases sont posées. L'équipe est mobilisée. Les modules sont construits un à un. Les données sont traitées, les interfaces se dessinent, la logique métier prend forme.

Le pilotage suit une méthode agile. Trello pour les tâches, Notion pour les notes, GitHub pour le code. Chaque sprint apporte des résultats concrets. Les rôles sont clairs. Les compétences se complètent. Chacun monte en expertise, et le projet progresse.

Techniquement, les pipelines sont en cours de déploiement. Les tableaux de bord s'assemblent. Le modèle de prédiction est en préparation. L'interface, elle, sera testée bientôt. Tout converge vers un outil cohérent, utile, et évolutif.

Au fil du projet, j'ai développé une vraie posture de gestion de projet data. J'ai cadré, planifié, suivi. J'ai aussi pris des décisions techniques et collaboré de manière structurée avec l'équipe. Ce projet m'aide à construire une approche professionnelle, concrète et valorisable.

## 10. Annexes :

### **Annexe 1 : Cahier des charges**

Le projet LogiOps360 répond à un besoin opérationnel. Dans l'industrie, la logistique est complexe. Les données sont dispersées, les outils limités. L'objectif : centraliser, visualiser, anticiper. L'utilisateur doit accéder simplement aux indicateurs clés, sans compétence technique.

Le contexte est clair. Les entreprises industrielles gèrent des volumes de commandes importants, répartis sur plusieurs sites, avec des contraintes de délais et des risques de rupture. L'idée de LogiOps360, c'est de proposer une vision globale, synthétique et interactive.

Les objectifs fonctionnels sont les suivants :

- Permettre à un utilisateur non technique de visualiser les données logistiques à travers une interface claire.
- Offrir un suivi par module : Commandes, Stock, Livraisons.
- Intégrer des indicateurs clés de performance, calculés automatiquement à partir des fichiers importés.
- Afficher des alertes simples en cas de dépassement de seuils (retard, saturation, rupture).
- Proposer un modèle de prédiction des retards, avec une restitution accessible (score, probabilité, explication).

L'interface est un point central. Elle doit permettre de naviguer entre les modules via un menu. Elle doit afficher les données sous forme de graphiques, de cartes ou de tableaux. Les filtres (par date, par entrepôt, par client) doivent être faciles à utiliser.

L'expérience utilisateur est conçue pour être fluide. Pas d'installation complexe, pas de manipulations techniques. L'utilisateur ouvre l'interface, importe un fichier CSV, et consulte ses indicateurs.

Du côté des contraintes, plusieurs éléments sont fixés dès le début :

- Le code doit être open source et documenté.
- L'outil ne doit pas nécessiter de base de données externe.
- L'ensemble doit être modulaire : chaque bloc fonctionnel doit pouvoir être testé séparément.

Enfin, le livrable attendu est un prototype fonctionnel, exploitable par un utilisateur métier dans un contexte semi-réel. Ce prototype servira de démonstrateur, en vue d'une industrialisation potentielle.

## **Annexe 2 : Spécifications techniques**

Le projet LogiOps360 repose sur une architecture simple, modulaire et reproductible. Elle a été pensée pour fonctionner localement, sans dépendre d'un cloud ou d'un serveur externe.

L'objectif : rendre l'outil accessible, même dans un environnement contraint.

Le langage utilisé est Python. C'est un choix cohérent. Il est maîtrisé par tous les membres de l'équipe. Il dispose d'un écosystème riche. Il permet de traiter les données, de modéliser, et de créer une interface.

Le traitement des données est assuré par pandas. Pour les opérations mathématiques, NumPy suffit. Les modèles prédictifs sont construits avec scikit-learn et tensorflow pour les modèles les plus avancés. Ces trois bibliothèques couvrent l'ensemble des besoins techniques du projet.

L'interface utilisateur repose sur Streamlit. Ce framework permet de transformer un script Python en une application web interactive. Il est léger. Il est adapté aux projets internes. Il offre un rendu visuel propre, sans nécessiter de connaissance en frontend.

Le code est versionné avec Git. Le dépôt est hébergé sur GitHub. Chaque module du projet est isolé. Il existe un script pour le traitement des commandes. Un autre pour le stock. Un troisième pour les livraisons. Enfin, un script gère la modélisation. Chacun peut être exécuté séparément.

L'architecture du projet est organisée ainsi :

- /data/raw : fichiers bruts.
- /data/cleaned : données nettoyées, prêtes à l'analyse.
- /notebooks : explorations, tests, validations intermédiaires.
- /scripts : scripts principaux, modulaires, commentés.
- /models : fichiers du modèle de prédiction (format .pkl).
- /dashboard : fichier Streamlit principal (app.py), avec les fonctions associées.

L'environnement technique est volontairement léger. Un ordinateur classique avec Python installé suffit. L'utilisateur lance simplement la commande streamlit run app.py. L'interface s'ouvre dans le navigateur.

Ce choix permet plusieurs choses. D'abord, le test rapide du prototype. Ensuite, l'autonomie pour un utilisateur technique. Enfin, la portabilité sur d'autres machines.

Le code est commenté. Chaque fonction a un objectif clair. Les noms de variables sont explicites. Les scripts sont courts, lisibles, testables.

Cette structure technique garantit la robustesse du projet, tout en restant suffisamment souple pour permettre des évolutions futures : ajout de modules, adaptation à d'autres jeux de données, ou intégration dans un environnement cloud si nécessaire.

### **Annexe 3 : Charte éthique et gouvernance des données**

Le projet LogiOps360 s'appuie sur des données logistiques publiques. Aucun fichier ne contient de données personnelles ou sensibles. Cette orientation a été volontaire. Elle vise à garantir un cadre de développement éthique, respectueux des utilisateurs et des principes fondamentaux de protection des données.

Dès la phase de cadrage, les choix techniques et organisationnels ont intégré des préoccupations de confidentialité. Les traitements sont réalisés localement. Aucun envoi vers un service tiers n'a lieu. Les fichiers sont analysés en mémoire, puis transformés et agrégés dans un dossier structuré. Rien n'est conservé en ligne. Aucun identifiant client, employé ou fournisseur réel n'est manipulé.

Les principes suivants ont guidé la gestion des données tout au long du projet :

1. Minimisation : seules les données strictement nécessaires à l'analyse sont utilisées. Si une variable est redondante, ou inutile à la modélisation, elle est exclue.
2. Transparence : chaque transformation de données (filtrage, nettoyage, agrégation) est documentée dans les scripts ou dans Notion. Tout est traçable.
3. Séparation des flux : les données brutes, les données nettoyées, et les données utilisées pour la visualisation sont stockées dans des dossiers distincts. Cette séparation renforce la lisibilité et la sécurité.
4. Stockage local : l'ensemble du prototype fonctionne en local. Aucun service cloud, aucune API externe, aucun transfert vers l'extérieur n'est requis.
5. Contrôle utilisateur : à tout moment, l'utilisateur peut consulter les fichiers chargés, les modifier ou les supprimer. L'interface ne masque pas les opérations réalisées sur les données.

Du point de vue réglementaire, ce projet ne relève pas du RGPD au sens strict, car il n'implique aucune donnée à caractère personnel. Toutefois, les bonnes pratiques inspirées de ce règlement ont été appliquées par précaution et par cohérence avec les standards actuels.

Si l'outil venait à être déployé dans un environnement réel, une politique plus complète de gouvernance serait nécessaire. Elle inclurait la gestion des habilitations, le chiffrage des fichiers, la traçabilité des accès, et la définition d'un rôle DPO (Data Protection Officer). Ces aspects relèveraient alors de la responsabilité de l'entreprise utilisatrice.

Dans sa version actuelle, LogiOps360 respecte un cadre éthique clair, cohérent, et proportionné à son usage. Il constitue ainsi un projet techniquement solide et éthiquement responsable.

### **Annexe 4 : Liens des outils de pilotage**

**Compte rendu des réunions d'équipe :**

[https://www.notion.so/Compte-rendu-des-reunions-d-equipe-2102e3d2649a80baa693fe493fd9d81d?source=copy\\_link](https://www.notion.so/Compte-rendu-des-reunions-d-equipe-2102e3d2649a80baa693fe493fd9d81d?source=copy_link)

**Dashboard Trello pour le suivi des tâches :**

<https://trello.com/invite/b/684a9d87de3f47d0736daf3b/ATT1b5490701ff2e928034470e62bb4cd278F9BF6F63/pilotage-projet-logiops360>