# Plan du projet

Projet : Amazon Review Analysis

Auteur : Amara NAIT SAIDI

Date : 30 novembre 2025

Version 1.0

## Table des matières

1. Contexte du projet ..... 3
2. Architecture du projet (périmètre technique) ..... 3
3. Acteurs du projet ..... 6
3.1. Prise en compte des acteurs en situation d'handicap ..... 7
3.2. Accessibilité pour les utilisateurs de la solution finale : ..... 7
4. Spécifications fonctionnelles ..... 7
5. Spécifications techniques ..... 8
6. Planning – diagramme de Gantt ..... 8
7. Evaluation des charges JH (en jours/homme) ..... 9
8. Schéma de gouvernance du projet ..... 10
8.1. Sprint planning ..... 10
8.2. Daily Scrum (Daily Stand-up) ..... 10
8.3. Sprint review ..... 11
8.4. Sprint rétrospective ..... 11
8.5. Backlog Refinement ..... 12
9. Matrice RACI ..... 13
10. Facteurs clés de succès du projet ..... 14

# 1. Contexte du projet

Amazon souhaite renforcer sa capacité à exploiter efficacement les avis clients, un élément déterminant pour améliorer l'expérience utilisateur et optimiser la qualité des produits et services. Le volume massif d'avis déposés chaque jour rend impossible une analyse manuelle exhaustive, alors même que ces retours contiennent des informations critiques : qualité perçue des produits, problèmes de livraison, difficultés rencontrées avec le service client, ou encore signaux faibles liés à la satisfaction globale.

Dans ce contexte, le projet vise à mettre en place une solution interne permettant **d'analyser automatiquement les avis**, de **détecter les thématiques principales**, et d'identifier les avis véritablement utiles pour la prise de décision. Pour cela, la solution s'appuie sur deux briques complémentaires :

- **un algorithme de score de pertinence (pondération)** permettant de quantifier la valeur réelle d'un avis,
- **un modèle de classification Zero-Shot**, capable de catégoriser les avis sans entraînement préalable.

L'ensemble du traitement est intégré dans **Snowflake**, garantissant scalabilité, gouvernance et performance, tandis qu'une interface **Streamlit** offre aux équipes Business un accès simple et fluide aux résultats. Cette combinaison permet de générer des tableaux de bord dynamiques, de fiabiliser la base d'avis analysables et de fournir aux Business Analysts un outil opérationnel pour suivre l'évolution de la satisfaction client et détecter rapidement les points d'attention.

En résumé, cette solution vise à transformer un volume massif de données non structurées en **insights actionnables**, tout en améliorant la productivité des équipes et la qualité des décisions stratégiques.

# 2. Architecture du projet et périmètre technique

L'architecture globale repose sur une chaîne de traitement data moderne, scalable et supervisée. Elle s'organise de la manière suivante :

**1. PostgreSQL – Base de données source**

Base transactionnelle contenant les avis laissés par les clients (texte, notes, informations produits…). Elle constitue le point d'entrée du pipeline.

**2. Amazon S3 – Datalake (zone brute)**

Les données extraites depuis PostgreSQL sont transférées dans S3, où elles sont stockées en format brut. Cette zone permet l'archivage, la traçabilité et la réexécution du pipeline si nécessaire.

**3. ETL Python – Extraction & Transformation**

Un pipeline ETL développé en Python assure :

- l'extraction des données depuis PostgreSQL,
- les premières transformations (nettoyage, normalisation, parsing),
- le chargement vers S3 puis vers Snowflake.

Un système de gestion d'erreurs est intégré dans le code.

**4. MongoDB – Stockage des logs et données rejetées**

Les données incomplètes, invalides ou rejetées par les contrôles qualité sont envoyées dans MongoDB. Les logs d'exécution y sont également stockés afin de faciliter l'audit et le debugging du pipeline.

**5. Snowflake – Data Warehouse & stockage analytique**

Snowflake centralise les données propres et validées.
Après transformation, les avis reçoivent :

- un **score de pertinence** (algorithme de pondération),
- une **catégorie Zero-Shot** (qualité produit, livraison, service client, etc.).

Les données enrichies sont stockées dans des tables dédiées pour la visualisation et l'analyse métier.

### 6. Notebooks Python – Application des algorithmes

Deux algorithmes principaux sont exécutés :

- **Zero-Shot Classification**,
- **Score de pertinence / pondération**.

Cette étape transforme les avis textuels en données exploitables et priorisées.

### 7. Streamlit – Tableau de bord interactif

Les données enrichies présentes dans Snowflake sont visualisées dans une application **Streamlit**, permettant :

- des dashboards dynamiques,
- la consultation des avis pertinents,
- la navigation par produit, catégorie ou score.

### 8. Application métier – Interface des Business Analysts

Une application interne permet aux équipes métiers de consulter facilement :

- les avis enrichis,
- les métriques de qualité produit,
- les alertes sur les anomalies ou tendances critiques.

---

### 9. Système d'alertes

Le pipeline intègre un mécanisme d'alerting afin d'assurer la fiabilité de la chaîne de traitement :

- **Alerte en cas d'absence de nouvelles données extraites depuis PostgreSQL**,
- **Alerte si aucune donnée n'a été chargée dans S3**,
- **Alerte si aucune donnée n'est envoyée dans Snowflake**,
- **Envoi automatique par e-mail** détaillant l'origine du problème (erreur d'extraction, échec de transformation, credentials invalides, etc.).

Ce système garantit la supervision du pipeline et réduit le risque de rupture opérationnelle.

# 3. Acteurs du projet

| Rôle | Compétences clés | Missions principales |
|---|---|---|
| Sponsor (Amazon – Direction) | Vision stratégique, arbitrage, gouvernance | Valide les grandes orientations, le budget et les jalons majeurs du projet |
| Client métier : Business Analysts | Analyse métier, expression de besoin, compréhension des enjeux business | Fournissent les besoins fonctionnels, valident les livrables, utilisent les dashboards en production |
| Product Owner (PO) | Vision produit, priorisation, rédaction de user stories, gestion du backlog | Porte la vision produit, priorise le backlog, rédige les user stories, valide fonctionnellement les livrables et fait le lien entre le métier et la technique |
| Scrum Master | Méthodes Agile, facilitation, gestion des rituels | Organise les sprints, anime les cérémonies (daily, sprint planning, rétrospective), supprime les obstacles et protège l'équipe |
| Data Architect | Architecture data, cloud (AWS), modélisation | Conçoit l'architecture globale (PostgreSQL – S3 – ETL – MongoDB – Snowflake – BI) |
| Tech Lead Data | Expertise data, leadership technique, reviews de code | Définit les choix techniques, supervise la qualité du code, accompagne les Data Engineers et valide l'implémentation |
| Data Engineers (x2) | Python, ETL, Snowflake, AWS, SQL | Développent les pipelines ETL, gèrent l'ingestion, le nettoyage et le chargement dans Snowflake |
| Data Scientist | IA, NLP, machine learning, scoring | Conçoit et implémente l'algorithme zéro et l'algorithme de pondération pour classifier et scorer les avis |
| DevOps Engineer | CI/CD, sécurité, Docker, AWS, monitoring | Met en place l'infrastructure, automatise les déploiements, gère la sécurité et la supervision |
| Security / Compliance Officer | Sécurité des données, RGPD, conformité | Vérifie la protection des données, les accès, la conformité légale (RGPD, sécurité cloud) |
| Formateur | Pédagogie, communication | Conçoit les supports et forme les utilisateurs métiers |
| Support Client | Support technique et fonctionnel | Assure la maintenance, gère les incidents et accompagne les utilisateurs après la mise en production |

## 3.1. Prise en compte des acteurs en situation d'handicap

Le projet intègre une démarche d'inclusion afin de permettre à tout membre de l'équipe, y compris ceux pouvant être en situation de handicap, de participer pleinement aux activités du projet.
Les mesures prévues sont les suivantes :
Accessibilité des communications et réunions

- comptes rendus écrits systématiques.
- supports compatibles avec lecteurs d'écran.
- sous-titrage automatique des réunions visio si nécessaire.
- réunions accessibles (rythme adapté, ordre du jour clair).

Aménagements de poste si besoin

- outils collaboratifs compatibles accessibilité (Teams, Confluence, Jira).
- horaires flexibles pour les handicaps invisibles (TDAH, fatigue chronique…).
- possibilité de télétravail renforcé.

Accessibilité des documents du projet

- documents structurés (titres, listes, contrastes corrects).
- formats inclusifs (PDF accessibles, alternatives textuelles).

## 3.2. Accessibilité pour les utilisateurs de la solution finale :

La solution est déstinée principalement à des utilisateurs internes (Business Analystes), une attention particulière est portée à sa lisibilité :

- navigation simple et épurée
- icônes ou couleurs doublées d'un texte explicite
- documentation accessible (format texte lisible, structure claire)

# 4. Spécifications fonctionnelles

La solution permet l'importation automatique des avis clients. Elle identifie et élimine les doublons afin d'éviter toute prise en compte erronée, puis procède à un nettoyage des données avant leur transformation pour obtenir un modèle fiable et structuré. Des algorithmes sont ensuite appliqués afin d'identifier les principales catégories, et à chaque avis est attribué un score de confiance associé à la catégorie détectée ainsi qu'un score de confidence et grâce à ces deux indicateurs des seuils sont fixés afin de déterminer si un avis est pertinent ou pas. La solution procède à une mise à jour quotidienne des données et maintient une table de rejets. Elle garantit ainsi une historisation complète et traçable.

# 5. Spécifications techniques

L'architecture technique repose sur plusieurs technologies complémentaires.

La base de données transactionnelle PostgreSQL est utilisée comme base de données source, tandis que le stockage brut des données s'effectue sur AWS S3. Les processus ETL sont gérés avec Python via des librairies. Les logs et les données rejetées sont stockés dans MongoDB. Les données propres sont enregistrées dans Snowflake, et le traitement de machine learning s'effectue à travers un notebooks Python. La couche de visualisation est consultate dans une application Streamlit. L'orchestration des workflows est assurée par Airflow.

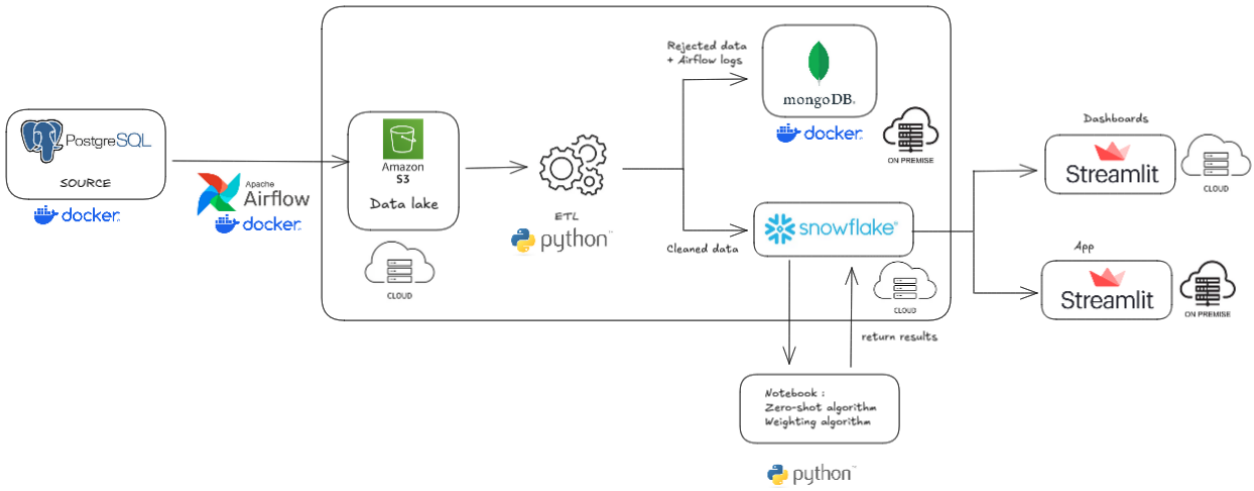


*Figure 1 : Architecture de la solution*

# 6. Planning – diagramme de Gantt

Le projet prendra en tout 18 semaines, avec 8 sprints de deux semaines (avec 2 semaines de marge en cas de retard). La livraison se fera à la fin du sprint 8 (phase 5).

Le projet sera découpé en 6 phases principales :
P1 : Cadrage des besoins
P2 : Conception et architecture
P3 : Développement et tests
P4 : Déploiement et formation
P5 : Support et stabilisation

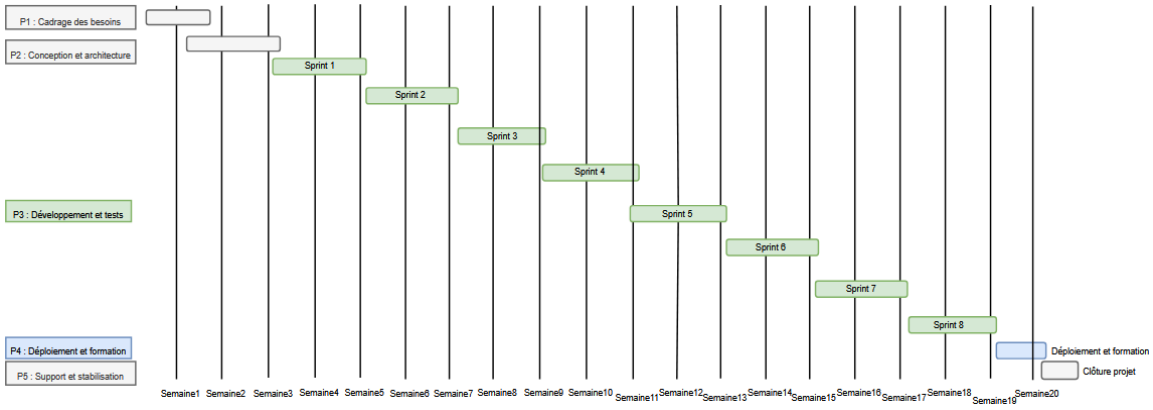


*Figure 2 : Planning du projet*

# 7. Evaluation des charges JH (en jours/homme)

Dans cette première vue, l'évaluation des charges en jours/hommes (J/H) est réalisée de manière globale, en prenant en compte l'ensemble des acteurs impliqués dans le projet, afin d'obtenir une vision complète de l'effort humain nécessaire à sa réalisation. Cette approche permet d'avoir une estimation macro du volume de travail à mobiliser sur toute la durée du projet, en intégrant les différents rôles.

Dans un second temps, et plus spécifiquement dans l'étape de calcul des coûts, l'analyse ne portera que sur les ressources réellement engagées dans la réalisation du projet. L'évaluation des charges sera alors recentrée uniquement sur les personnes participant activement aux sprints, afin d'obtenir une estimation financière plus précise et plus représentative de la réalité opérationnelle.

| Rôle | Durée d'intervention | Charge (JH) |
|---|---|---|
| Sponsor | 1 j par mois | 5 JH |
| Business Analysts | Très actifs P1 + P4 | 23 JH |
| Product Owner | 3 à 4 j / semaine | 40 JH |
| Scrum Master | 3 j / semaine (P2-P3) | 35 JH |
| Data Architect | P1 + P2 | 15 JH |
| Tech Lead Data | P2 à P3 | 25 JH |
| Data Engineers (x2) | P3 principalement (6 sem) | 50 JH (25 x 2) |
| Data Scientist | P3 | 15 JH |
| DevOps Engineer | P2 à P4 | 15 JH |
| Security / Compliance | Interventions ponctuelles | 7 JH |
| Formateur | P4 | 10 JH |
| Support Client | P5 | 10 JH |

Total du projet ~ 250 JH.

# 8. Schéma de gouvernance du projet

La gouvernance du projet s'articule autour de trois niveaux complémentaires : stratégique, tactique et opérationnel. Cette organisation permet d'assurer un pilotage efficace tout en maintenant l'agilité nécessaire à l'équipe de développement. Les instances mises en place garantissent l'alignement entre les objectifs métier d'Amazon et les réalisations concrètes, avec des points de synchronisation réguliers et des circuits de décision clairs. L'objectif est simple : garder une vision d'ensemble du projet et s'assurer de son bon avancement.

Voici une explication des instances :

## 8.1. Sprint planning

Au démarrage de chaque sprint, on planifie ce qu'on va développer pendant les deux semaines à venir. Le Product Owner présente les priorités, l'équipe estime la charge et on s'engage sur un objectif réaliste.

| Élément | Détails |
|---|---|
| Composition | - Product Owner<br>- Scrum Master<br>- Data Architect<br>- Tech Lead Data<br>- Data Engineers (x2)<br>- Data Scientist |
| Fréquence | Toutes les 2 semaines (début de sprint) |
| Durée | 2 heures |
| Objectifs | - Planification du sprint<br>- Définition des user stories<br>- Estimation des charges<br>- Engagement de l'équipe |

## 8.2. Daily Scrum (Daily Stand-up)

Le point rapide de 15 minutes chaque matin pour synchroniser l'équipe technique. Chacun dit ce qu'il fait, ce qui bloque, et on avance. Simple et efficace.

| Élément | Détails |
|---|---|
| Composition | - Scrum Master<br>- Data Engineers (x2)<br>- Data Scientist<br>- Tech Lead Data |
| Fréquence | Quotidienne |
| Durée | 15 minutes |
| Objectifs | - Point d'avancement journalier<br>- Identification des blocages<br>- Synchronisation de l'équipe |

## 8.3. Sprint review

La démo de fin de sprint où on montre ce qu'on a développé aux parties prenantes métier. Ça permet de valider qu'on est sur la bonne voie et d'ajuster si besoin.

| Élément | Détails |
|---|---|
| Composition | - Scrum team (PO, SM, Developers…)<br>- Parties prenantes (stekeholders, utilisateurs…). |
| Fréquence | Toutes les 2 semaines (fin de sprint) |
| Durée | 2h (par sprint) |
| Objectifs | - Démonstration des fonctionnalités<br>- Validation<br>- Collecte des feedbacks<br>- Ajustement du backlog |

## 8.4. Sprint rétrospective

Un moment pour l'équipe de prendre du recul : qu'est-ce qui a bien marché ? Qu'est-ce qu'on peut améliorer ? On en sort avec des actions concrètes pour être plus efficaces au prochain sprint.

| Élément | Détails |
|---|---|
| Composition | - Scrum team (PO, SM, Developers…) |
| Fréquence | À la fin de chaque sprint (juste après la review) |
| Durée | 1h30 |
| Objectifs | - Analyse de la performance d'équipe<br>- Identification des améliorations<br>- Actions correctives<br>- Renforcement de la collaboration |

## 8.5. Backlog Refinement

Ce n'est pas un événement officiel dans le guide, mais il est reconnu comme une pratique courante et fortement recommandée, son but est de préparer les prochains sprints.

| Élément | Détails |
|---|---|
| Composition | - Scrum team (PO, SM, Developers…) |
| Fréquence | 1 à 2 fois par sprint (en général) |
| Durée | 1 heure |
| Objectifs | - Clarification des user stories<br>- Découpage et estimation<br>- Repriorisation si besoin |

# 9. Matrice RACI

Voici la matrice RACI pour chaque phase du projet :

| Phase | Responsible (R) | Accountable (A) | Consulted (C) | Informed (I) |
|---|---|---|---|---|
| P1 – Cadrage des besoins | - Product Owner<br>- Business Analysts | - Sponsor (Amazon – Direction) | - Data Architect<br>- Scrum Master<br>- Tech Lead Data | - Data Engineers<br>- Data Scientist<br>- DevOps Engineer<br>- Security/Compliance<br>- Formateur<br>- Support Client |
| P2 – Conception & architecture | - Data Architect<br>- Tech Lead Data<br>- Product Owner | - Product Owner (PO) | - Business Analysts<br>- Scrum Master<br>- Security/Compliance<br>- Data Scientist | - Data Engineers<br>- DevOps Engineer<br>- Sponsor<br>- Formateur<br>- Support Client |
| P3 – Développement & tests | - Data Engineers<br>- Data Scientist<br>- DevOps Engineer<br>- Tech Lead Data | - Scrum Master | - Product Owner<br>- Data Architect<br>- Security/Compliance | - Sponsor<br>- Business Analysts<br>- Formateur<br>- Support Client |
| P4 – Déploiement & formation | - DevOps Engineer<br>- Tech Lead Data<br>- Data Engineers<br>- Formateur | - Product Owner (PO) | - Security/Compliance<br>- Scrum Master<br>- Support Client<br>- Business Analysts | - Sponsor<br>- Data Scientist<br>- Data Architect |
| P5 – Support & stabilisation | - Support Client<br>- DevOps Engineer | - Sponsor (Amazon – Direction) | - Product Owner<br>- Scrum Master<br>- Business Analysts | - Data Engineers<br>- Data Scientist<br>- Data Architect<br>- Tech Lead Data<br>- Security/Compliance<br>- Formateur |

**Légende :**

- Responsible : réalise la tâche.
- Accountable : porte la responsabilité finale de la tâche et valide le résultat.
- Consulted : Consulté pour avis ou expertise.
- Informed : Informé du déroulement ou du résultat de la tâche.

# 10. Facteurs clés de succès du projet

1. Inclusion et accessibilité : le projet doit permettre à tous les membres de l'équipe de travailler dans de bonnes conditions, y compris s'ils sont en situation de handicap. Cela passe par des supports accessibles, des réunions adaptées et la possibilité d'aménager le poste de travail si besoin.

2. Données fiables dès le début : pour que les algorithmes fonctionnent correctement, il est essentiel que les données sources soient complètes, propres et disponibles au bon moment. Si la base de départ n'est pas fiable, tout le reste du projet en pâtit.

3. Communication régulière avec les équipes métier : les échanges fréquents avec les Business Analysts sont indispensables pour vérifier que les traitements, les catégories et l'ensemble de la solution correspondent bien à leurs besoins.

4. Pipeline stable et bien surveillé : l'un des points clés est d'avoir un pipeline ETL solide, avec des logs clairs et un monitoring qui permet de détecter rapidement les problèmes.

5. Sécurité et conformité : le respect des règles de sécurité et du RGPD est non négociable, surtout dans un contexte avec des données clients. C'est un élément de réussite et aussi une obligation.

6. Adoption de la solution **:** le projet sera une réussite si la solution est réellement utilisée par les équipes métier, et si elle apporte une vraie aide à l'analyse des avis.

7. Bonne organisation du travail : la coordination entre les différents rôles (Data Engineer, Architecte, Data Scientist, DevOps…) est un point clé. Le pilotage du projet, les réunions régulières et un suivi clair aident à éviter les blocages.

8. Documentation simple et accessible : pour que tout le monde puisse suivre le projet facilement (y compris les personnes en situation de handicap), la documentation doit être claire, structurée et lisible.

9. Bonne gestion des sprints : des objectifs réalistes, une charge bien estimée et un rythme de travail stable contribuent beaucoup au bon déroulement du projet.