

République Française
Ministère de l'Enseignement Supérieur,
de la Recherche et de l'Innovation

UNIVERSITÉ CLERMONT AUVERGNE
INSTITUT UNIVERSITAIRE DE TECHNOLOGIE
BUT 3 Sciences des données parcours Visualisation, conception
d'outils décisionnel

Module : SAE conception d'un outil décisionnel

CAHIER DES CHARGES FONCTIONNELLES & TECHNIQUES

Réaliser par :

- Firdaouss Kehailou
- Noah Cantin
- Amadou Moussa Sow
- Illyes Maouda

Encadrer par:

- Lagarde Bruno

Table de matière

1. Description du Projet ou du Produit	4
1.1 Couche de Présentation :	4
1.2 Couche de Traitement :	4
1.3 Couche de Stockage :	5
2. Objectifs et Résultats Attendus.....	5
2.1 Conception du projet.....	6
2.2 Méthodologie de Mise en Œuvre.....	7
2.3 Gestion du Changement.....	7
2.4 Plan de Communication.....	7
2.4.1 Fréquence des Réunions.....	7
2.4.2 Canaux de Communication.....	8
3. Missions Commanditées (Rapports F1 à F8 et L1 à L5)	8
4. Contraintes Techniques, Économiques, Réglementaires, Environnementales et Documentation.....	12
5. Spécifications Fonctionnelles, Techniques et de Performance.....	12
5.1 Systèmes d'Exploitation :	12
5.2 Fonctionnelles :	12
5.3 Techniques :	12
5.4 Performance :	13
6. Exigences de Qualité et de Sécurité.....	13
6.1 Qualité des Données :	13
6.1.1 Sources de Données :	13
6.1.2 Exactitude des Données :	13
6.1.3 Cohérence des Données :	14
6.1.4 Fiabilité des Données :	14
6.1.5 Contrôle de Validation :	14
6.2 Sécurité des Données.....	14
6.2.1 Gestion des Accès Utilisateurs :	14
6.2.2 Droit d'Accès :	15
6.2.3 Authentification :	15
6.2.4 Session et Déconnexion :	15
6.2.5 Protection contre les Attaques Externes :	15
6.2.6 Confidentialité des Informations Sensibles :	16
7. Performance et Scalabilité :	16
8. Délais et Coûts Prévus.....	16
9. Responsabilités et Rôles des Acteurs Impliqués.....	17

Table de figures :

Figure 1: MCD de la base de données du stock	5
Figure 2: Diagramme de classe du projet	6
Figure 3: Diagramme de déploiement du projet.....	6
Figure 4: Diagramme de Gantt.....	7
Figure 5 : Processus d'ETL.....	13

Cahier des Charges

1. Description du Projet ou du Produit

L'objectif de ce projet est de concevoir et de mettre en place un entrepôt de données pour centraliser, stocker et analyser les informations relatives aux ventes et aux livraisons de l'entreprise de vente en ligne. L'entrepôt de données devra permettre de générer des rapports détaillés sur les performances commerciales et logistiques afin de soutenir la prise de décision pour les services finances et logistique de l'entreprise. Pour cela il faudra un développement en IOS pour les tablettes de l'entreprise et pour les PC il faut un développement sur les systèmes d'exploitation présent dans l'entreprise (Windows ou Linux au choix). Cette application comprendra aussi un compte utilisateur accessible via le mail professionnel cela sera précisé. L'architecture technique de l'entrepôt de données sera fondée sur un modèle en couches, permettant une séparation claire des préoccupations et favorisant la modularité. Cette structure permettra d'optimiser la gestion des données, le stockage et l'accès tout en assurant une performance élevée.

1.1 Couche de Présentation :

- **Interfaces Utilisateurs** : L'accès aux données sera facilité par des interfaces web et des applications mobiles conçues pour être intuitives. Ces interfaces permettront aux utilisateurs de naviguer aisément à travers les tableaux de bord, les visualisations et les rapports générés.
- **Technologies Utilisées** (au choix). Les technologies de développement telles que Python, Angular, React ou Vue.js seront explorées pour concevoir des interfaces réactives, tandis que des outils de BI (Business Intelligence) comme Tableau ou Power BI pourraient être intégrés pour des visualisations avancées.

1.2 Couche de Traitement :

- **Moteur Analytique** : Au cœur de cette couche, un moteur de traitement des données sera mis en place pour exécuter des requêtes complexes et effectuer des analyses en temps réel. Cela peut inclure des outils comme Apache Spark ou un entrepôt de données comme Snowflake ou Amazon Redshift à votre choix.
- **Chargement des Données** : Les données provenant des sources seront traitées à l'aide d'un processus ETL (Extract, Transform, Load) robuste, permettant une transformation et une normalisation des données avant qu'elles ne soient stockées.

1.3 Couche de Stockage :

- **Base de Données Relationnelle** : Utilisation d'une base de données relationnelle, comme PostgreSQL ou MySQL, va permettre le stockage et la gestion des données avec la possibilité d'effectuer des requêtes complexes.
- **Stockage des Données Non Structurées** : Pour les données non structurées (documents, images, etc.), une solution de stockage telle qu'**Amazon S3** qu'utilise l'entreprise sera intégrée. Cela permettra de gérer efficacement le Big Data.

2. Objectifs et Résultats Attendus

Service des Finances : Générer des rapports détaillant les chiffres d'affaires, les remises, les frais de port, les commandes, les employés responsables, les fournisseurs, et les transporteurs.

Service Logistique : Optimiser la gestion des livraisons, les stocks, les réapprovisionnements, et le suivi des transporteurs et des expéditions.

Ces objectifs possèdent à Fournir des informations permettant une gestion optimisée des stocks et des livraisons pour améliorer l'efficacité de l'entreprise.

Les tables suivantes représentent les différents besoins pour la gestion des stocks. Dans ces tables, nous avons les différentes relations, les clés primaires.

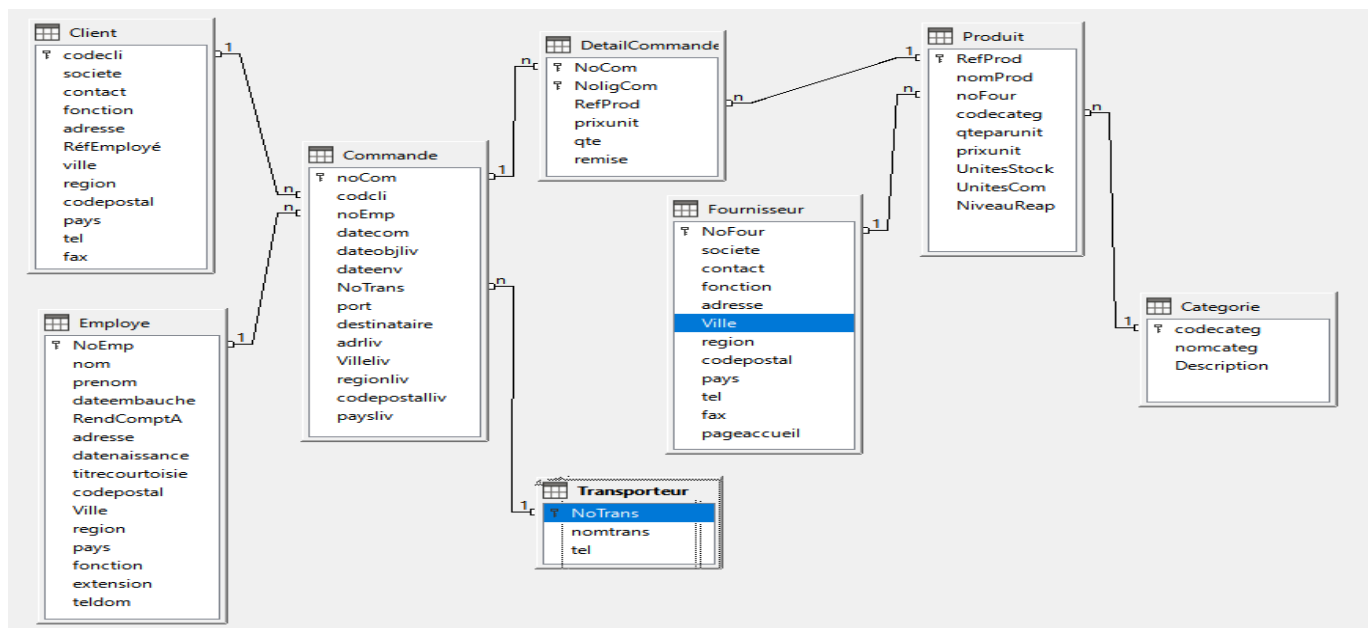


Figure 1: MCD de la base de données du stock

2.1 Conception du projet

Dans le cadre de la conception de notre projet, nous avons travaillé sur deux types de diagrammes UML : le diagramme de classes et le diagramme de déploiement. Ces outils nous ont permis de structurer et de modéliser efficacement les différentes composantes du projet :

- **Diagramme de classes** : Ce diagramme a servi à représenter les entités principales du système, leurs attributs, leurs méthodes, ainsi que les relations entre elles. Il constitue la base pour comprendre et implémenter la structure logique du projet.

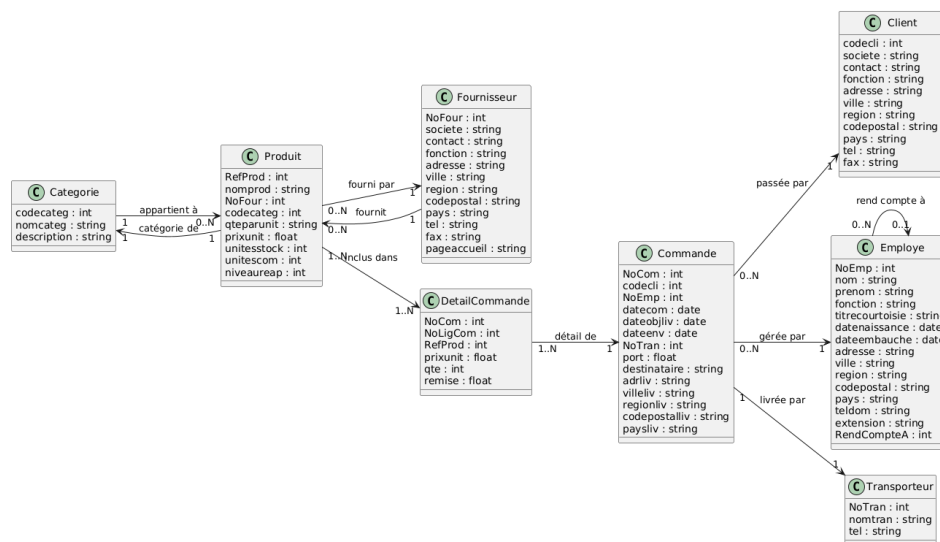


Figure 2: Diagramme de classe du projet

- **Diagramme de déploiement** : Ce diagramme a permis de visualiser l'architecture physique du système, en mettant en évidence les composants matériels, les nœuds, et la répartition des logiciels.

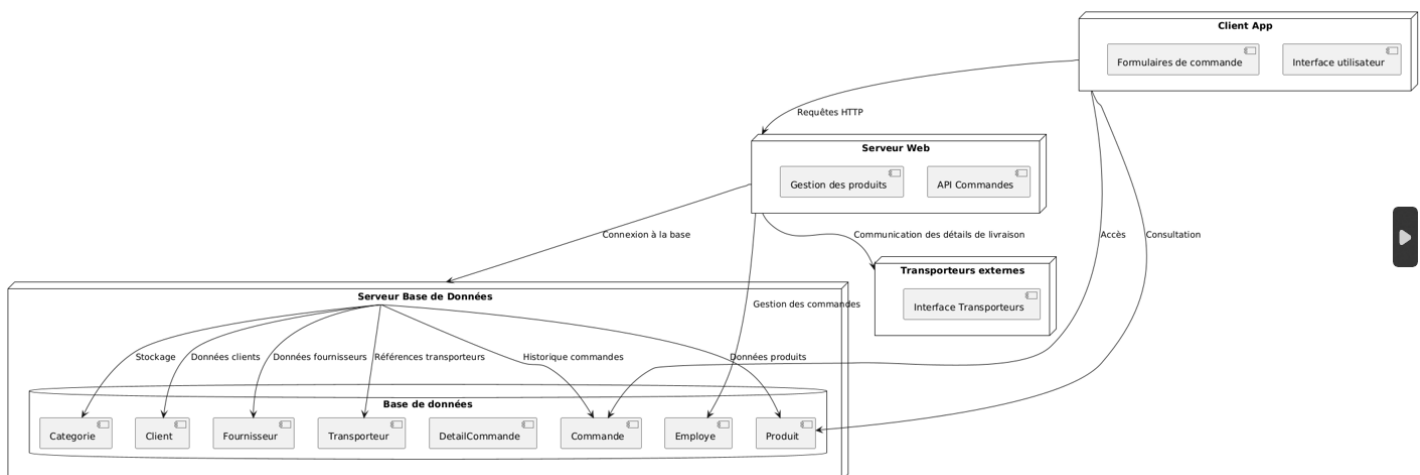


Figure 3: Diagramme de déploiement du projet

2.2 Méthodologie de Mise en Œuvre

- **Approche projet** : Méthode agile avec une série de sprints pour permettre des ajustements basés sur des retours fréquents.
- **Phases de livraison** :
 - **Phase 1 : Analyse des besoins** (Durée : 1 mois)
 - **Phase 2 : Conception** (Durée : 2 mois)
 - **Phase 3 : Développement** (Durée : 3 mois)
 - **Phase 4 : Tests et Validation** (Durée : 2 mois)
 - **Phase 5 : Déploiement** (Durée : 1 mois)



Figure 4: Diagramme de Gantt

2.3 Gestion du Changement

Etablir un processus formel de gestion des changements :

- **Demande de changement** : Formulaire à soumettre pour toute demande de modification.
- **Évaluation** : Une équipe dédiée évaluera l'impact de tout changement proposé.
- **Validation** : Les changements seront validés par les parties prenantes avant d'être mis en œuvre.

2.4 Plan de Communication

2.4.1 Fréquence des Réunions

- Réunions hebdomadaires pour mise à jour et suivi des progrès.
- Réunions mensuelles avec toutes les parties prenantes pour évaluation et direction du projet.

2.4.2 Canaux de Communication

- Utilisation d'e-mails pour les communications formelles.
- Plateformes de messagerie instantanée avec Microsoft Teams de l'entreprise pour des échanges rapides.
- Outil de gestion de projet GitHub utilisé par l'entreprise pour la traçabilité des tâches et des progrès.

3. Missions Commanditées (Rapports F1 à F8 et L1 à L5)

Les codes suivants sont présents à titre indicatif pour illustrer de manière grossière ce qui est attendue dans chaque partie des missions.

Pour le Service des Finances (Rapports F1 à F8)

F1 - Liste décroissante des produits avec les plus grosses remises

Ce rapport doit fournir une liste des produits ayant les remises les plus élevées, triée par montant de remise. Il doit inclure le client concerné et l'employé responsable de la commande.

Code SQL :

```
SELECT p.nomprod, SUM (dc.remise) AS remise_totale, c.societe AS client, e.nom AS employe
FROM DetailCommande as dc
JOIN Produit p ON dc.RefProd = p.RefProd
JOIN Commande cmd ON dc.NoCom = cmd.NoCom
JOIN Client c ON cmd.codecli = c.codecli
JOIN Employe e ON cmd.NoEmp = e.NoEmp
GROUP BY p.nomprod, c.societe, e.nom
ORDER BY remise_totale DESC ;
```

F2 - Liste du chiffre d'affaires par produit avec sous-totaux par catégorie de produitCe rapport doit fournir le chiffre d'affaires pour chaque produit, avec des sous-totaux par catégorie de produit.

Code SQL :

```
SELECT c.nomcateg, p.nomprod, SUM (dc.qte * dc.prixunit) AS chiffre_affaire
FROM DetailCommande as dc
JOIN Produit p ON dc.RefProd = p.RefProd
JOIN Categorie c ON p.codecateg = c.codecateg
GROUP BY c.nomcateg, p.nomprod;
```

F3 - Liste du chiffre d'affaires par pays avec le total des frais de port
Ce rapport doit fournir le chiffre d'affaires généré par chaque pays, ainsi que le total des frais de port associés à chaque commande.

Code SQL :

```
SELECT cli.pays, SUM (dc.qte * dc.prixunit) AS chiffre_affaire, SUM (cmd.port) AS frais_port
FROM Commande as cmd
JOIN Client cli ON cmd.codecli = cli.codecli
JOIN DetailCommande dc ON cmd.NoCom = dc.NoCom
GROUP BY cli.pays;
```

F4 - Liste du chiffre d'affaires transporté par transporteur
Ce rapport doit indiquer le chiffre d'affaires total par transporteur, avec la répartition des commandes expédiées par chaque transporteur.

Code SQL :

```
SELECT t.nomtran, SUM (dc.qte * dc.prixunit) AS chiffre_affaire
FROM Commande as cmd
JOIN Transporteur t ON cmd.NoTran = t.NoTran
JOIN DetailCommande dc ON cmd.NoCom = dc.NoCom
GROUP BY t.nomtran;
```

F5 - Liste des commandes par employé avec son nom et les remises moyennes qu'il accorde par client
Ce rapport doit fournir la liste des commandes traitées par chaque employé, en incluant les remises moyennes accordées à chaque client par cet employé.

Code SQL :

```
SELECT e.nom, cmd.NoCom, c.societe, AVG (dc.remise) AS remise_moyenne
FROM Commande as cmd
JOIN Employe e ON cmd.NoEmp = e.NoEmp
JOIN Client c ON cmd.codecli = c.codecli
JOIN DetailCommande dc ON cmd.NoCom = dc.NoCom
GROUP BY e.nom, cmd.NoCom, c.societe;
```

F6 - Liste des chiffres d'affaires et quantités réalisées avec les produits, par fournisseur

Ce rapport doit fournir le chiffre d'affaires généré et les quantités vendues pour chaque produit, regroupés par fournisseur.

Code SQL :

```
SELECT f.societe, p.nomprod, SUM (dc.qte) AS quantite_vendue, SUM (dc.qte * dc.prixunit) AS  
chiffre_affaire  
FROM DetailCommande as dc  
JOIN Produit p ON dc.RefProd = p.RefProd  
JOIN Fournisseur f ON p.NoFour = f.NoFour  
GROUP BY f.societe, p.nomprod;
```

F7 - Cumul des frais de port par transporteur avec le nombre d'expéditions

Ce rapport doit fournir le total des frais de port générés par chaque transporteur, ainsi que le nombre d'expéditions réalisées par chaque transporteur.

Code SQL :

```
SELECT t.nomtran, SUM (cmd.port) AS frais_port, COUNT (cmd.NoCom) AS nombre_expeditions  
FROM Commande as cmd  
JOIN Transporteur t ON cmd.NoTran = t.NoTran  
GROUP BY t.nomtran;
```

F8 - Rapport supplémentaire utile pour le service des finances

Ce rapport supplémentaire pourrait être conçu pour répondre à un besoin spécifique non couvert par les rapports précédents. Voici un exemple qui fournit des informations sur les remises moyennes par pays.

Code SQL :

```
SELECT cli.pays, AVG (dc.remise) AS remise_moyenne  
FROM Commande as cmd  
JOIN Client cli ON cmd.codecli = cli.codecli  
JOIN DetailCommande dc ON cmd.NoCom = dc.NoCom  
GROUP BY cli.pays;
```

Pour le Service Logistique (Rapports L1 à L5)**L1 - Liste des plus grands écarts entre la date d'objectif de livraison et la date d'envoi de la commande**

Ce rapport doit lister les commandes où l'écart entre la date d'objectif de livraison et la date d'envoi de la commande est significatif

.

Code SQL :

```
SELECT NoCom, DATEDIFF (dateenv, dateobjliv) AS ecart_jours
FROM Commande
WHERE DATEDIFF (dateenv, dateobjliv) > 0 ;
```

L2 - Liste des produits dont le niveau de réapprovisionnement est déclenché

Ce rapport doit lister les produits dont le niveau de réapprovisionnement est atteint, soit lorsque l'unité en stock moins l'unité en commande est inférieure au niveau de réapprovisionnement.

Code SQL :

```
SELECT RefProd, nomprod, unitesstock, unitescom, niveaureap
FROM Produit
WHERE (unitesstock - unitescom) < niveaureap;
```

L3 - Quantités de produits livrées par région et par ville de livraison

Ce rapport doit fournir des informations détaillées sur les quantités de produits livrées dans chaque région et ville.

Code SQL :

```
SELECT cmd.villeliv, cmd.regionliv, SUM (dc.qte) AS qte_livree
FROM Commande as cmd
JOIN DetailCommande dc ON cmd.NoCom = dc.NoCom
GROUP BY cmd.villeliv, cmd.regionliv;
```

L4 - Liste des commandes à livrer ayant un fournisseur dans la même région

Ce rapport doit lister les commandes dont les fournisseurs sont situés dans la même région que la destination de la commande, pour optimiser les coûts de transport.

Code SQL :

```
SELECT cmd.NoCom, f.societe AS fournisseur, cmd.regionliv
FROM Commande as cmd
JOIN Produit p ON cmd.NoCom = p.RefProd
JOIN Fournisseur f ON p.NoFour = f.NoFour
WHERE cmd.regionliv = f.region;
```

L5 - Rapport supplémentaire utile pour le service logistique

Ce rapport supplémentaire pourrait être conçu pour répondre à un besoin spécifique non couvert par les rapports précédents. Voici un exemple qui fournit des informations sur les produits livrés en retard.

Code SQL :

```
SELECT NoCom, DATEDIFF (dateenv, dateobjliv) AS ecart_jours, cmd.villeliv
FROM Commande as cmd
WHERE DATEDIFF (dateenv, dateobjliv) > 0 ;
```

4. Contraintes Techniques, Économiques, Réglementaires, Environnementales et Documentation

Techniques : Le système doit être conçu pour traiter un grand volume de données et pour générer des rapports rapidement. Les données doivent être accessibles en temps réel ou presque.

Économiques : Le budget pour le projet doit inclure la conception, la mise en œuvre, les ressources humaines, les outils nécessaires et la maintenance.

Réglementaires : Le système doit respecter les normes relatives à la protection des données personnelles et à la sécurité des informations (conformité au RGPD).

Environnementales : Le projet doit être conçu avec une gestion optimale des ressources matérielles et énergétiques, en tenant compte des impératifs environnementaux.

Documentation : Création de guides d'utilisation distribués à l'ensemble des utilisateurs après la conception. Cela va permettre aux personnes qui ont accès de bien savoir se servir du logiciel.

5. Spécifications Fonctionnelles, Techniques et de Performance

5.1 Systèmes d'Exploitation :

- **Systèmes Linux** : Les distributions Linux telles que Ubuntu seront utilisées pour les serveurs de base de données et de traitement avec **PostgreSQL ou MySQL**.
- **Systèmes Windows** : Les serveurs de web et les applications métiers pourront être déployés sur des systèmes Windows, en fonction des spécifications des applications requises.

Étant donné que Windows est le système d'exploitation le plus utilisé dans les entreprises, il faudra une version Windows du logiciel.

5.2 Fonctionnelles :

L'entrepôt de données devra centraliser et organiser les données relatives aux commandes, aux produits, aux clients, aux employés, aux fournisseurs et aux transporteurs.

Il devra permettre de générer les rapports financiers et logistiques mentionnés ci-dessus, avec une capacité de filtrage et d'agrégation des données selon différentes périodes (mois, trimestres, années). Ces données devront pouvoir être exportées au format **DataFrame** (df), CSV, TXT, XML ou vers une autre base de données.

5.3 Techniques :

Le système utilisera une base de données relationnelle pour le stockage des données (Nosql).

Un processus ETL (Extract, Transform, Load) sera mis en place pour intégrer et transformer les données provenant des différentes sources.

En guise d'illustration d'un processus ETL.

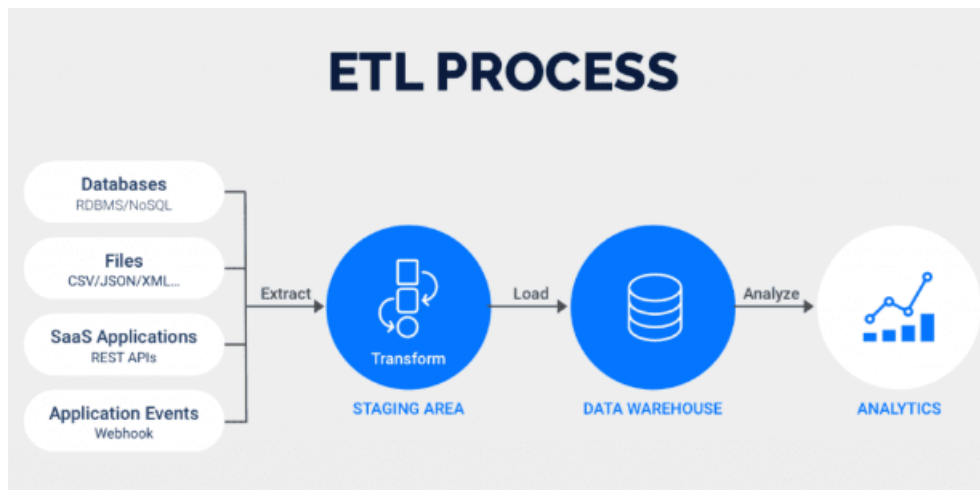


Figure 5 : Processus d'ETL

5.4 Performance :

Le temps de réponse pour générer les rapports doit être inférieur à 10 secondes pour les rapports standard, et le système doit être capable de gérer un volume important de données en croissance. Le système devra être scalable pour s'adapter à l'évolution de l'entreprise (augmentation du volume de données, nouveaux besoins en rapports, etc.).

6. Exigences de Qualité et de Sécurité

6.1 Qualité des Données :

La qualité des données est cruciale pour assurer l'intégrité des analyses et des décisions basées sur l'entrepôt de données.

Voici les exigences spécifiques en matière de qualité :

6.1.1 Sources de Données :

- **Intégration des Données** : Un ensemble de connecteurs ETL sera développé pour intégrer des données de sources variées telles que des systèmes ERP (comme SAP) et des CRM (comme Salesforce). L'Outil Talend peut être utilisé pour automatiser le chargement des données.
- **Gestion de la Qualité des Données** : Un processus de validation des données sera mis en place pour garantir leur précision, leur intégrité et leur cohérence avant leur chargement dans l'entrepôt.

6.1.2 Exactitude des Données :

- **Validation à l'Entrée** : Les données doivent être validées lors de leur saisie ou intégration dans l'entrepôt pour garantir leur précision. Ceci inclut des vérifications de format (ex : formats de date, adresse e-mail valide etc..) et des contrôles de cohérence.

- **Erreurs et Anomalies** : Des alertes doivent être mises en place pour signaler les erreurs de saisie, comme des quantités négatives ou des prix inexacts.

6.1.3 Cohérence des Données :

- **Contrôles entre Systèmes** : Les données doivent être comparées régulièrement entre plusieurs systèmes d'origine (ERP, CRM) pour s'assurer qu'elles correspondent.
- **Règles de Transformation** : Développer et documenter des règles ETL (Extract, Transform, Load) qui définissent comment les données doivent être transformées et intégrées pour éviter des incohérences lors de l'intégration.

6.1.4 Fiabilité des Données :

- **Historique des Modifications** : Maintenir un historique des versions des données pour permettre un audit complet et une traçabilité. Cela implique la capture des modifications apportées aux données critiques.
- **Rapports de Qualité** : Émettre des rapports réguliers sur la qualité des données comprenant des statistiques sur les taux d'erreur et des suggestions d'amélioration.

6.1.5 Contrôle de Validation :

- **Tests Réguliers** : Effectuer des tests de qualité des données à intervalles réguliers afin de détecter les erreurs, les doublons et les valeurs aberrantes.
- **Processus de Révision** : Établir un processus de révision des données par les utilisateurs concernés pour valider les données critiques utilisées dans les rapports décisionnels.

6.2 Sécurité des Données

La sécurité des données est essentielle pour protéger les informations sensibles contre les accès non autorisés et les cybermenaces. Les exigences en matière de sécurité incluent :

6.2.1 Gestion des Accès Utilisateurs :

- **Gestion des Groupes Utilisateurs** : Créer des groupes d'utilisateurs définis (ex : Administrateurs, Utilisateurs Financiers, Utilisateurs Logistiques) avec des droits d'accès spécifiques. Documenter les rôles assignés pour les utilisateurs au sein de chaque groupe afin de garantir la **traçabilité** des permissions.

6.2.2 Droit d'Accès :

- **Niveau Société** : Déterminer qui a accès aux données pour une société spécifique (ex : un utilisateur ne peut accéder qu'aux données de sa propre société).
- **Niveau Fonction** : Définir des permissions basées sur les fonctions pour limiter l'accès aux fonctionnalités spécifiques du système (ex : génération de rapport, modification de données).
- **Niveau Administrateur** : Les administrateurs doivent avoir des droits de paramétrage complets, leur permettant de configurer les accès et les paramètres du système. Ils doivent également être responsables de la gestion des rôles et des droits d'accès.

6.2.3 Authentification :

- **Mots de Passe Sécurisés** : Imposer des règles strictes en matière de mots de passe (longueur minimale, complexité) pour garantir une protection robuste contre les accès non autorisés. Les mots de passe doivent être stockés de manière cryptée et ne doivent pas être accessibles en clair.

6.2.4 Session et Déconnexion :

- **Inactivité** : Implémenter une fonctionnalité de déconnexion automatique au bout d'une période d'inactivité déterminée (ex : 15minutes) pour éviter tout accès non autorisé lorsque l'utilisateur s'absente.
- **Maintenance** : Permettre la déconnexion sécurisée des utilisateurs pendant les opérations de maintenance sans affecter les données ou l'intégrité du système.

6.2.5 Protection contre les Attaques Externes :

- **Pare-feu et systèmes de détection d'intrusion** : Mettre en place des pare-feux pour protéger le réseau de l'entrepôt de données. Utiliser des systèmes de détection et de prévention des intrusions pour surveiller les activités suspectes en temps réel.
- **Chiffrement des Données Sensibles** : Chiffrer les données sensibles, tant au repos (stockées dans la base de données) qu'en transit (lors de la transmission sur des réseaux), pour garantir la confidentialité des informations des clients, des employés et des produits.
- **Audits et Tests de Sécurité** : Réaliser des audits de sécurité réguliers et des tests d'intrusion pour identifier et corriger les vulnérabilités dans le système.

6.2.6 Confidentialité des Informations Sensibles :

- **Politique de Confidentialité** : Établir une politique claire de confidentialité des données, spécifiant comment et pourquoi les données sont stockées et utilisées, ainsi que les droits des utilisateurs.
- **Responsabilisation des Utilisateurs** : Formation des utilisateurs sur l'importance de la sécurité des données, la gestion de l'accès et les meilleures pratiques pour garantir la sécurité des informations sensibles.

7. Performance et Scalabilité :

L'architecture technique devra être conçue pour garantir une performance optimale et une évolutivité durable.

- **Optimisation des Requêtes** :
Des techniques avancées telles que l'indexation appropriée, l'utilisation de vues matérialisées et le partitionnement des tables seront mises en œuvre pour accélérer les temps de réponse lors des requêtes.
- **Mise en Cache** : Un système de mise en cache, via Redis, sera intégré pour stocker temporairement les résultats des requêtes fréquentes, réduisant ainsi la charge sur la base de données.
- **Scalabilité Horizontale** : L'infrastructure sera conçue pour permettre l'ajout de nœuds supplémentaires facilement, que ce soit pour le traitement ou le stockage, en utilisant la technologie de cloud computing AWS, permettant ainsi de répondre à une augmentation soudaine des demandes d'analyse des données.

8. Délais et Coûts Prévus

Délais : Le projet doit être achevé dans un délai de 9 mois, incluant les phases de conception, développement, test et déploiement.

Coûts : Le budget total du projet sera défini en fonction des besoins en ressources humaines, matérielles et logicielles nécessaires. Il comprendra également les coûts de formation, de maintenance et de support après le déploiement.

9. Responsabilités et Rôles des Acteurs Impliqués

Chef de Projet : Supervision générale du projet, gestion des délais, du budget et de la communication entre les équipes.

Équipe Technique : Conception, développement et mise en œuvre de l'entrepôt de données, des processus ETL et des rapports.

Services Finances et Logistique : Fourniture des spécifications et validation des rapports. Ils sont responsables de l'acceptation des livrables.

Équipe de Test : Test des rapports, de la performance du système, et validation des données.

Service IT et Maintenance : Assurer la maintenance, le support et la mise à jour continue du système après son déploiement.