

Introduction

L'information est devenue un élément essentiel pour assurer un avantage concurrentiel aux entités qui savent l'utiliser. C'est le constat qui explique la raison poussant les entreprises à mettre en place des systèmes de collecte et de traitement de données toujours plus performants. Elles intègrent de plus en plus des processus nécessaires à la rencontre de l'innovation technologique. Notamment par l'implantation accélérée de solutions accélérant la fluidité de l'information décisionnelle au sein de l'entreprise. C'est dans ce cadre que s'inscrit notre projet qui consiste à réaliser un système décisionnel pour aider les dirigeants à la prise des décisions tactiques et stratégiques.

Structure détaillée de la solution

1.1 Diagramme de classe

Un diagramme de classes fournit une vue globale d'un système en présentant ses classes, interfaces et collaborations, et les relations entre elles La figure 1.1 montre le diagramme de classe global relative à notre projet.

1.2 Architecture de l'application

L'architecture logicielle décrit d'une manière symbolique et schématique les différents éléments d'un ou de plusieurs systèmes informatiques, leurs interrelations et leurs interactions. Notre modèle d'architecture introduit par la figure 1.2 décrit l'architecture de notre système de gestion.

Notre système est composé de 3 parties principales :

- Power BI
- Power Apps
- Power Automate

Pour l'architecture logicielle du système d'aide à la prise de décision, nous avons utilisé l'outil d'ETL Open Talend pour mettre en scène et transformer les données source en un entrepôt de données central construit à l'aide d'un produit de base de données relationnelle SQL. La figure 1.3 décrit l'architecture de notre système de gestion.

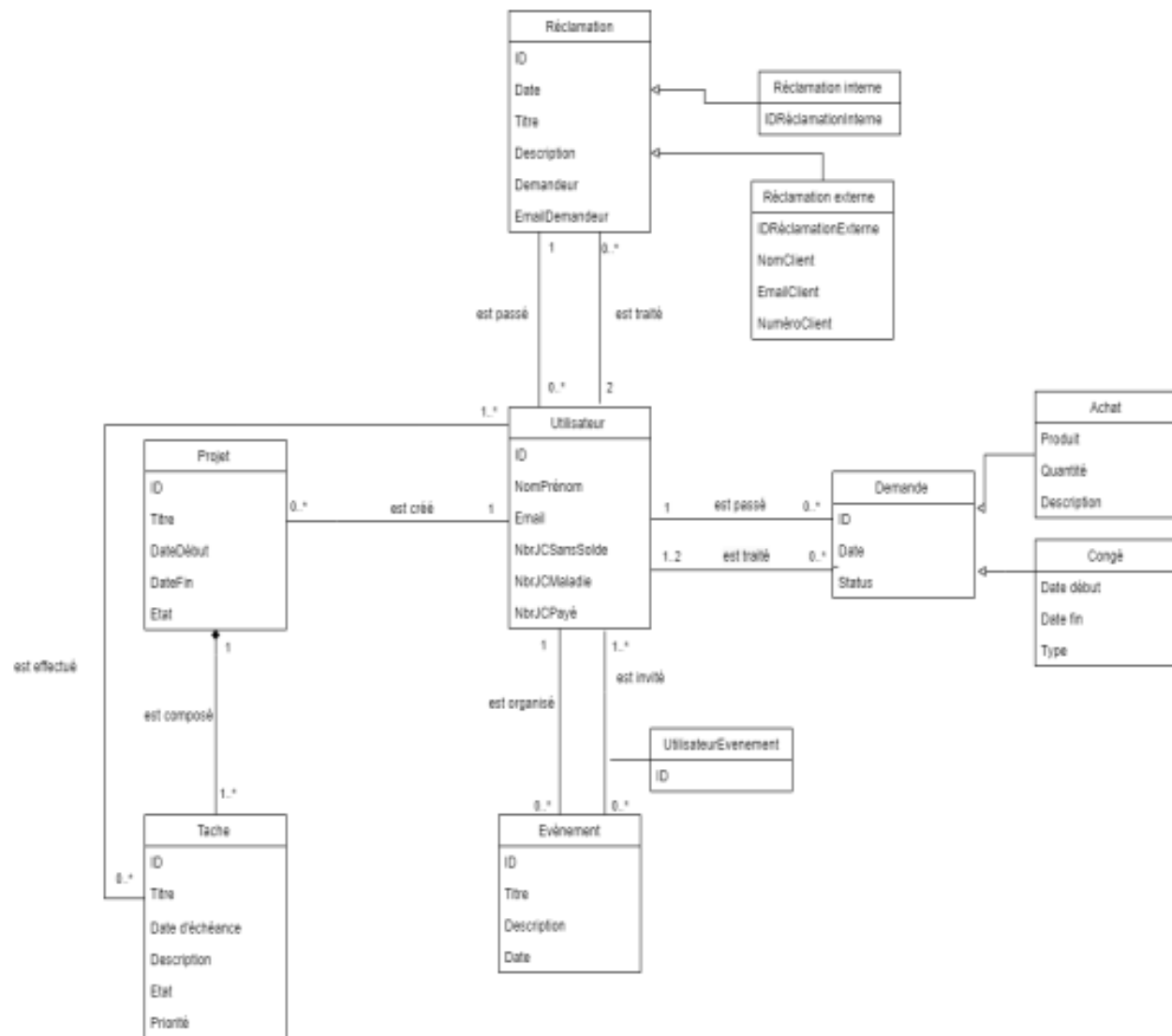


FIGURE 1.1 – Diagramme de classe

1.3 Présentation des concepts reliés au système de prise de décision

Un système de prise de décision est construit étape par étape, de l'extraction des données au déploiement. La figure 1.4 présente son architecture technique globale.

Cette architecture clarifie les différentes étapes d'un système de prise de décision depuis l'extraction des données des sources hétérogènes, puis la transformation et les chargements dans des entrepôts de données. Ces données sont unifiées à fin d'avoir un état cohérent. Grâce à ces derniers, nous créerons les tableaux de bords qui aideront les décideurs à prendre les bonnes décisions.



FIGURE 1.2 – Architecture du système de gestion d'entreprise

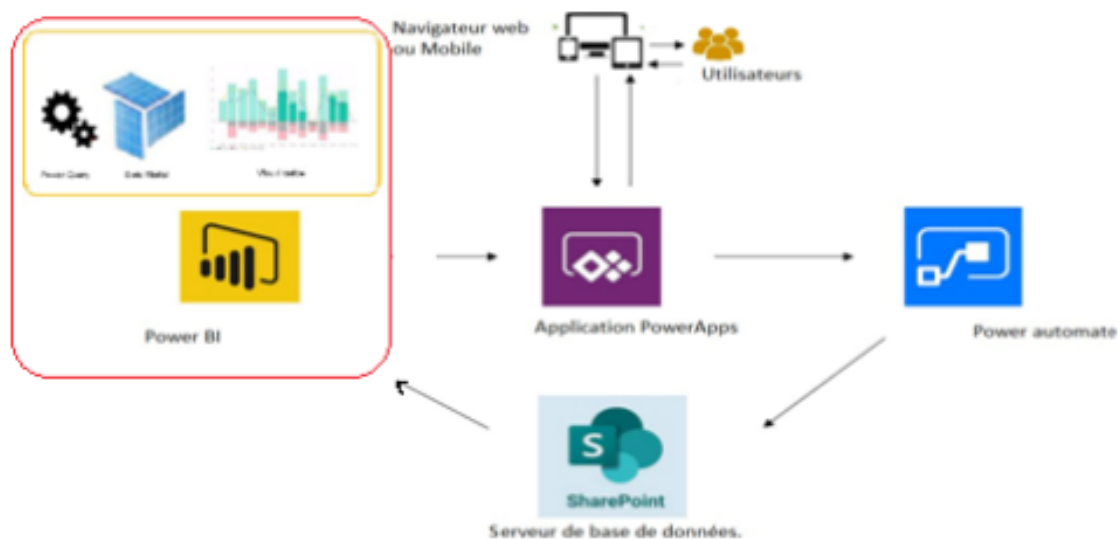


FIGURE 1.3 – Architecture du système d'aide à la prise de décision

1.3.1 Processus ETL

Les ETL désignent une séquence d'opération qui commence par l'extraction de données des différentes sources d'information et qui sont aussi de différents types, grâce à des connecteurs ou des APIs « Application Programming Interface ». L'étape suivante est la transformation de données brutes collectées pour les nettoyer, conformer, standardiser, documenter, corriger et dupliquer au besoin. C'est une étape cruciale qui va permettre de rendre l'ensemble de données compatibles entre elles et conforme au format cible, et elle consiste aussi à agréger les tables de données entre elles. La dernière étape c'est le chargement des données dans un Data Warehouse qui sont organisées et structurées sur la base d'un langage commun.

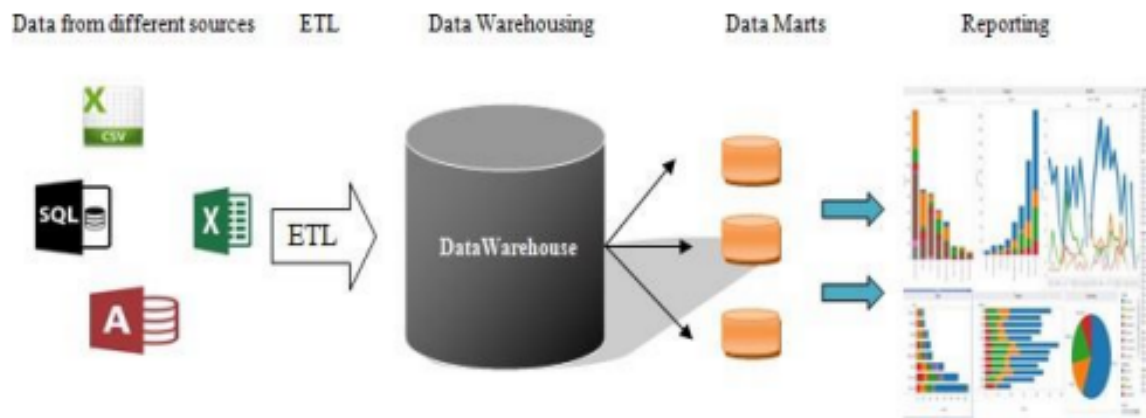


FIGURE 1.4 – Architecture technique d'un système de prise de décision

1.3.2 Entreposage de données

Le résultat d'un ETL sera chargé dans un entrepôt de donnée. Il sera utilisé par le système décisionnel puisque le DW est source d'information homogène, commune, normalisée et fiable. Le Data Warehouse doit suivre plusieurs principes clés : Oriente sur le sujet : La structure du data Warehouse doit être conçue en fonction des besoins de n'importe quels secteurs particuliers de l'entreprise. Non volatile : Les données ne doivent jamais être réécrites ou supprimées, elles sont statiques et on peut juste les lire. Intégré : le DW contient l'ensemble de données de l'entreprise qui doivent être fiables et cohérentes entre elles. Historisé : Les modifications ou les ajouts doivent être enregistrés et datés. Il existe deux types de stockage de données les entrepôts de données qui collecte toutes les données, ou bien dans des petites magasins appelées Datamart qui est un sous-ensemble de Data Warehouse, la différence entre les deux est que le Datamart répond à un besoin métier plus spécifique que le Data Warehouse.

1.3.3 Le Reporting

La restitution est la dernière étape d'un processus BI, elle est appelée aussi le reporting, qui nous permettons de voir les données stocker sous forme de tableau de bord ou bien des rapports, d'une manière claire et plus lisible dans le cadre de l'aide à la décision via des outils d'analyse décisionnels. Le reporting permet aux décideurs de :

- Sélectionner des données par période, production, par secteur etc.
- Trier, regrouper ou répartir ces données selon des critères de choix.
- Réaliser des calculs.
- Présenter les résultats de manière détaillée, généralement sous forme de graphiques ou bien des tableaux de bord.

Conception de la solution

2.1 Conception physique d'entrepôt de données

2.1.1 Préparation des dimensions

Les tables de dimensions contiennent le contexte textuel associé à un événement de mesure de processus métier. Ils décrivent le « qui, quoi, où, quand, comment et pourquoi » associé à l'événement. Différents types de dimensions existent, dans notre cas, nous allons utiliser les éléments suivants :

- Slowly-changing dimension (SCD) : Des dimensions qui changent lentement au fil du temps, plutôt que de changer selon un calendrier régulier. Des attributs comme noms d'emplacements géographiques, clients peut changer mais pas trop souvent.
- Static Dimension : Ces dimensions ne sont pas extraites de la source de données d'origine, mais sont générées à l'aide d'un SQL script ou une procédure stockée et sont chargés manuellement. La dimension temporelle est un classique exemple de cote statique
- Conformed dimension peut exister sous la forme d'une table à dimension unique qui se rapporte à plusieurs tables de faits dans le même entrepôt de données.
- définition des dimensions

2.1.2 Préparation des tables de fait

Une table de faits est utilisée dans le modèle dimensionnel dans la conception de l'entrepôt de données. Il se trouve au centre d'un schéma en étoile ou d'un schéma en flocon de neige entouré de tables de dimensions. Une table de faits se compose de deux types de colonnes. La colonne des clés étrangères permet de joindre des tables de dimensions et les colonnes de mesure contiennent les données en cours d'analyse. Dans notre cas, nous avons quatre table

Dimension	Type
Dim_Achat	Slowly-changing dimension
Dim_Conge	Slowly-changing dimension
Dim_produit	Slowly-changing dimension
Dim_Projet	Slowly-changing dimension
Dim_Reclamation	Slowly-changing dimension
Dim_Status	Static, Conformed dimension
Dim_Type	Static
Dim_Type_Reclamation	Static
Dim_Utilisateur	Slowly-changing dimension

FIGURE 2.1 – La description des dimensions

de faits qui sont « FactAnalyseConge », « FactAnalyseProjet », « FactAnalyseReclamation », « FactAnalyseAchat ». Ces tables de faits sont appelées « factless » puisque ils n'ont aucune mesure stockée, ces tables ne contiendront que des clés de différentes tables de dimension.

2.2 Conception d'entrepôt de données

Après avoir défini les dimensions et les colonnes nécessaires, nous avons défini les tables de faits. Ceci consiste à générer le modèle physique pour l'entrepôt de données à implémenter ensuite. La figure 2.2 montre la modélisation multidimensionnelle de l'entrepôt de données.

2.2.1 Conception et développement d'ETL

Après avoir créé les tables de fait et les tables des dimensions dans notre base de données SQL tout en affectant les relations entre eux nous passons à appliquer les étapes de l'ETL.

- Extract : c'est le processus de lecture de données à partir d'une base de données. À ce stade, les données sont collectées, souvent à partir de sources multiples et différentes.
- Transform : Dans une large mesure, la transformation dans ce contexte consiste à manipuler les données d'une manière qui répond aux besoins de l'entreprise. En d'autres

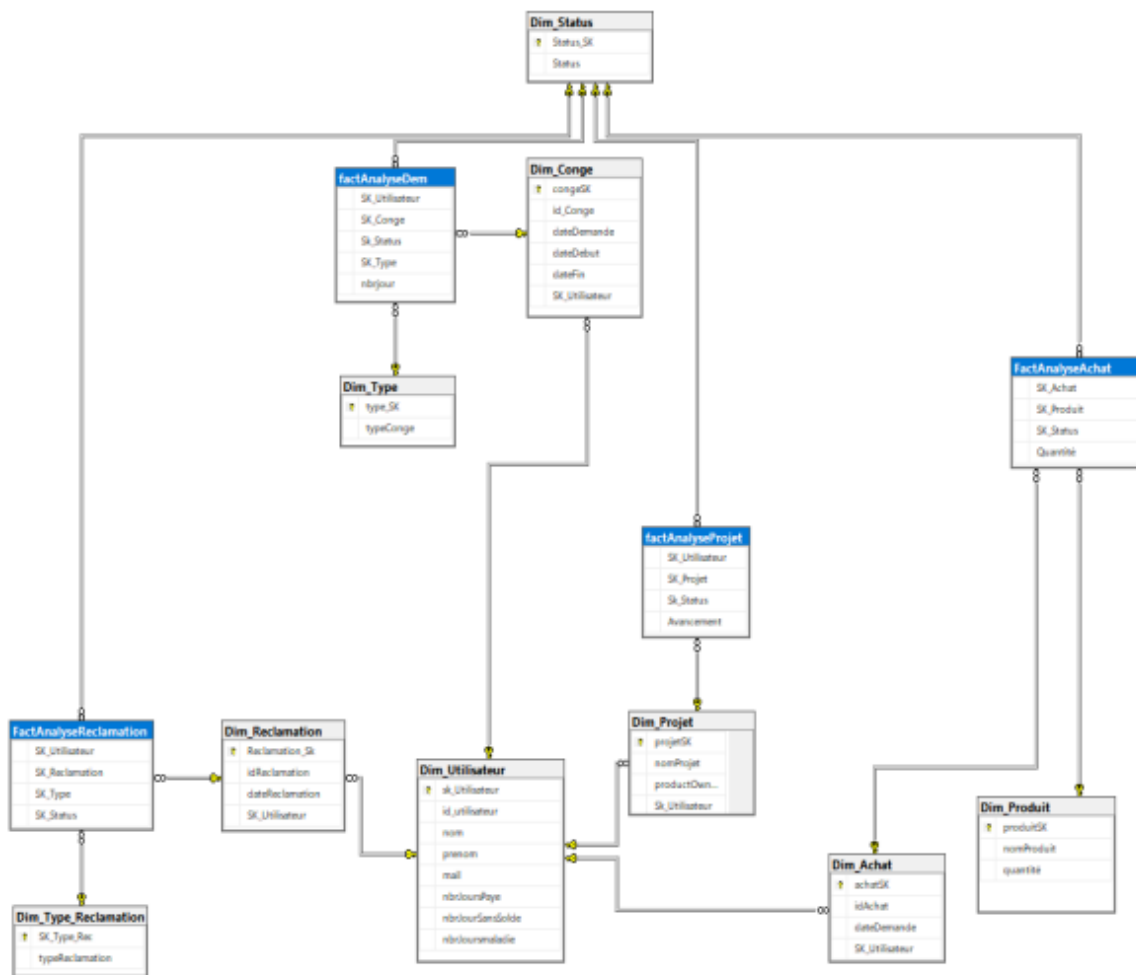


FIGURE 2.2 – La Modélisation multidimensionnelle de l’entrepôt de données

termes, les données brutes qui circulent dans l’organisation peuvent ne pas convenir à tout type d’utilisation, elles doivent donc être nettoyées, filtrées, etc. C’est la phase où les données génériques sont transformées en quelque chose qui peut être un atout précieux pour l’entreprise.

- Load : C’est le processus d’écriture des données dans la base de données cible. Ci-dessous, nous allons vous présenter toutes les étapes de l’ETL que nous avons appliqué grâce à l’outil Talend open studio for data integration par un exemple bien expliquer.

Chapitre 3

réalisation

3.1 connexion à la base de données :

Nous avons commencé par l'établissement de la connexion pour la source d'entrée qu'est un fichier csv et la source de sortie qu'est une base de données SQL déjà créer.

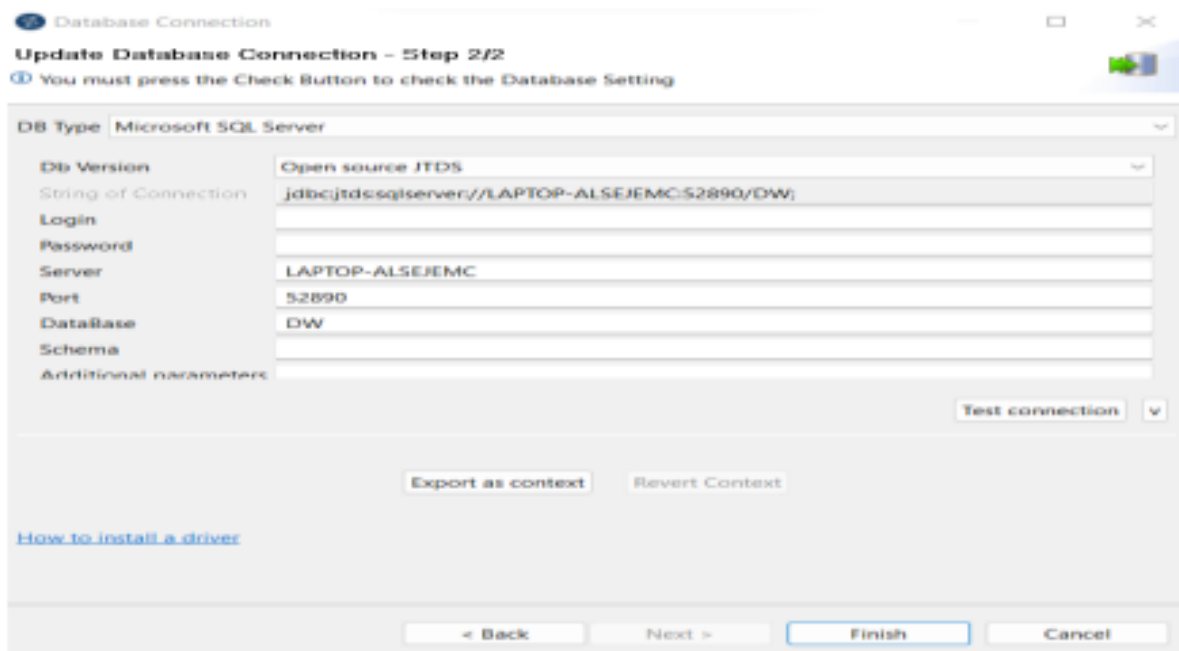


FIGURE 3.1 – Établissement de la connexion pour les données à extraire

Ensuite nous avons injecté dans notre job que nous avons déjà créés nos métadonnées en tant que « tFileInputDelimiter » puis nous avons utilisé le composant « tUniqueRow » pour supprimer les valeurs en double et le composant « tMap » pour faire le mapping entre tous les fichiers d'entré et la source table SQL que sera modélisée par le « tDBOutput ». Ensuite nous avons appliqué destransformationssur les données en choisissonsles colonnes nécessaires.

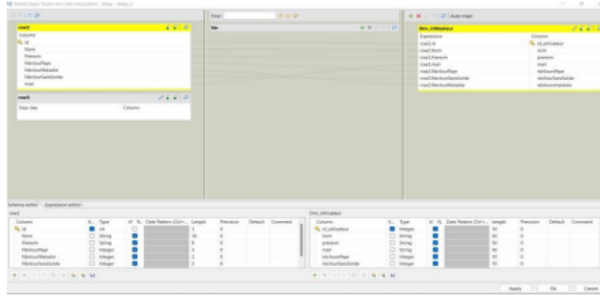


FIGURE 3.2 – Transformation de données pour les tables de dimensions

Dans La figure 3.3 nous montrons quelques jobs de chargement de des dimensions :

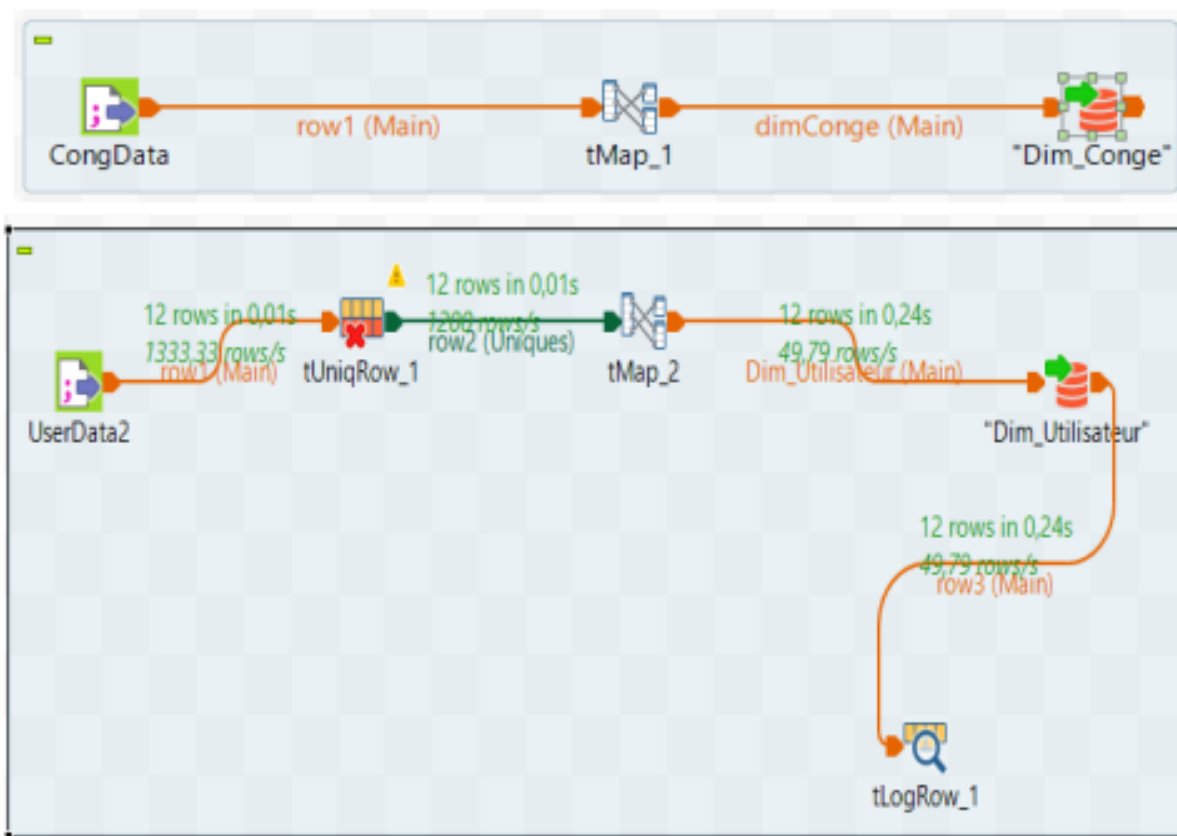


FIGURE 3.3 – Job de chargement des dimensions

Enfin à l'aide de « tLogRow » on trouve bien que la table de dimension a été bien charger dans notre entrepôt de données.

Après avoir préparé les dimensions, nous passons par la suite pour préparer les tables de fait.ci dessus un exemple, on fait nous avons injecté toute la dimension liée à la table de fait à partir de notre entrepôt de données on appliquant des requêtes SQL pour sélectionner les

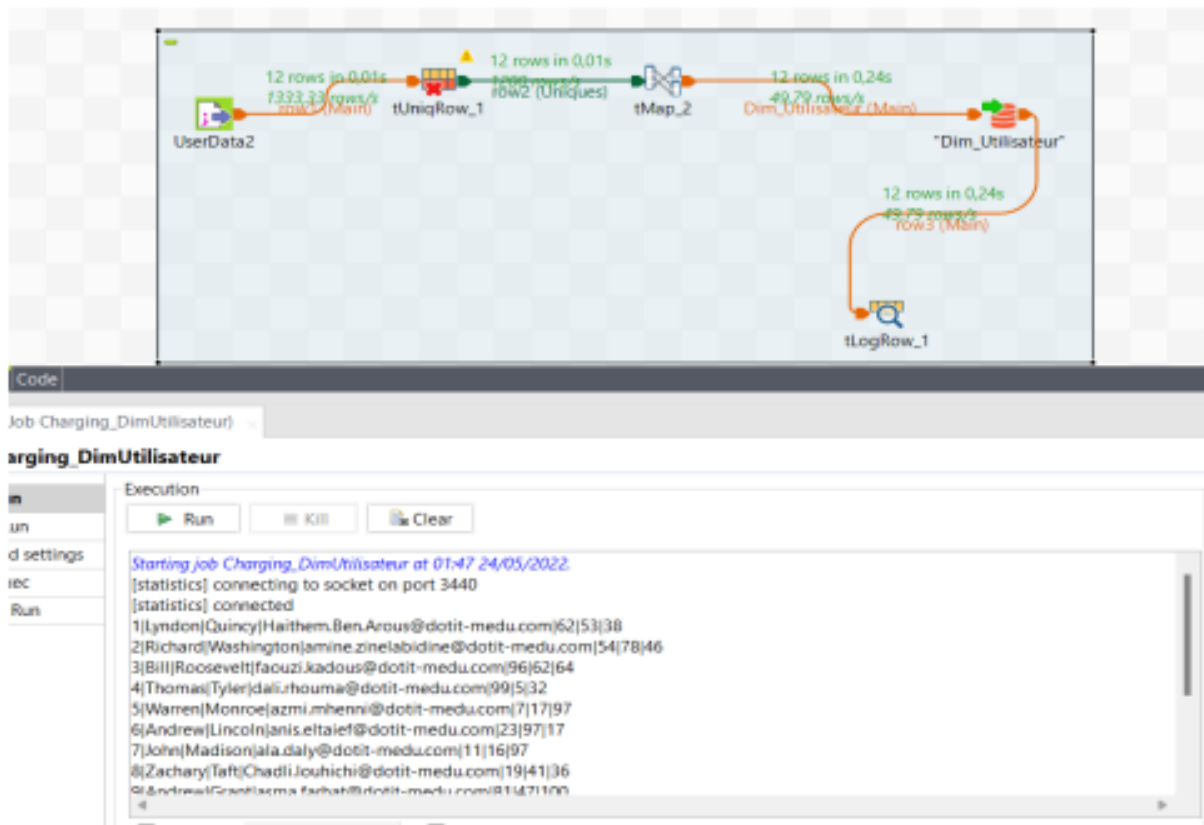


FIGURE 3.4 – La chargement de données dans la base de données SQL

colonnes des dimensions en éditant ses schémas pour effectuer des jointure avec la table de fait.

3.2 Le Reporting

A ce niveau, nous nous concentrons sur la modélisation des tableaux de bord avec Power bi en définissant comment un fait doit être enrourler. Comme nous nous montre les figures ci-dessus, Dans notre cas nous travaillerons sur une source de données SQL, Il nous faut également l'importer en sélectionnant les différentes tables de la base de données SQL que nous souhaitons intégrer dans notre rapport.

Comme nous montre la figure 3.9, si on rendre dans le modèle de vue dans power bi on trouve que les tables sont importées en concevant la modélisation multidimensionnelle qu'on a fait dans la base de données SQL.

Cependant, nous n'avons encore qu'à faire les modélisations de nos tableaux de bord. La

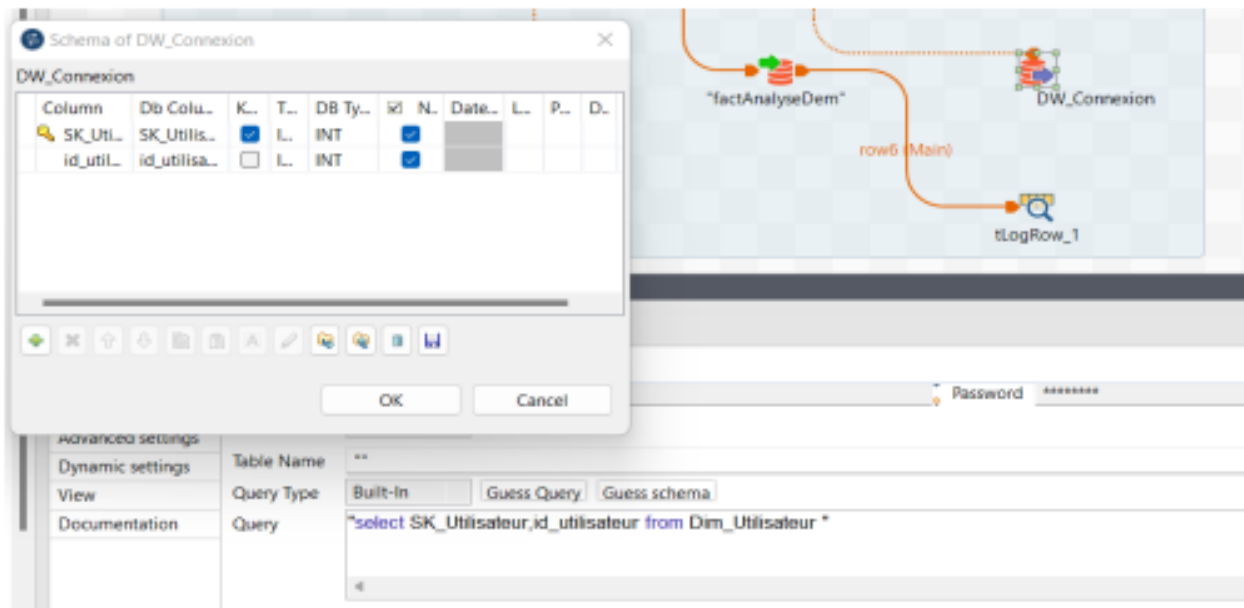


FIGURE 3.5 – Editions des schémas

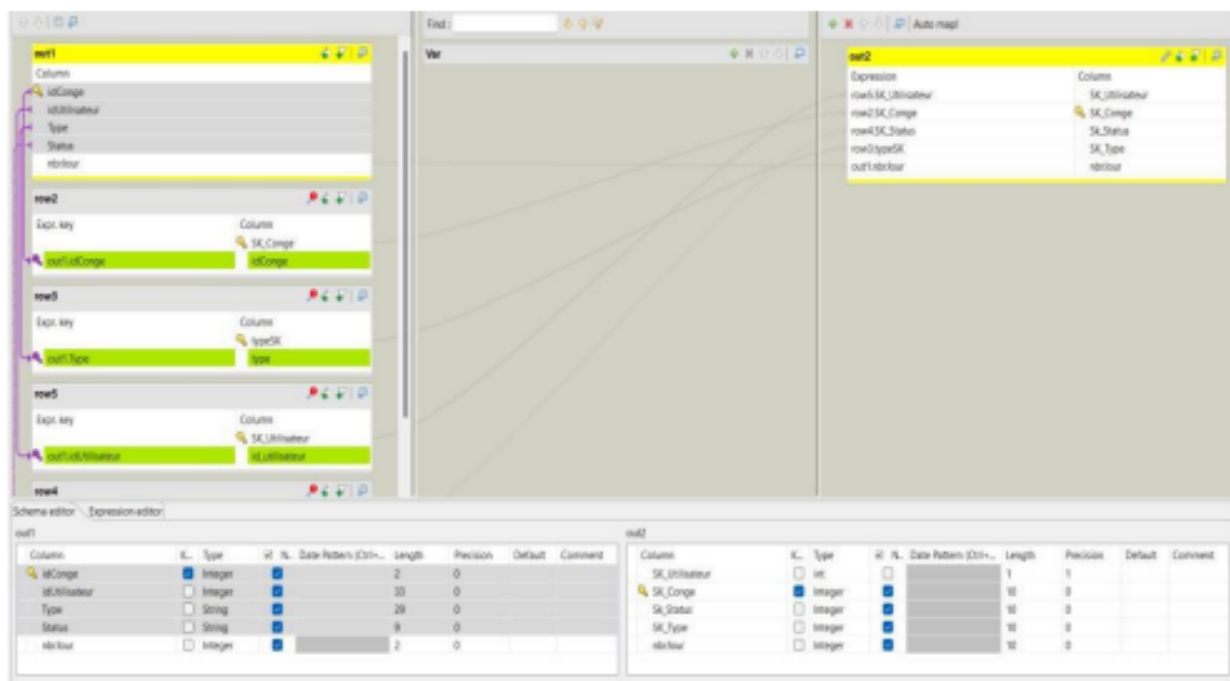


FIGURE 3.6 – La Transformation des données pour la table de fait

figure 3.10 illustre notre premier tableau de bord qui représente les nombres de demandes congelées par type, statuts, mois, et utilisateur, Afin d'aider les responsables RH de Détecter les utilisateurs, les mois qui ont le plus nombre de demande congelée, et comprendre les statuts et les types de ses demandes.

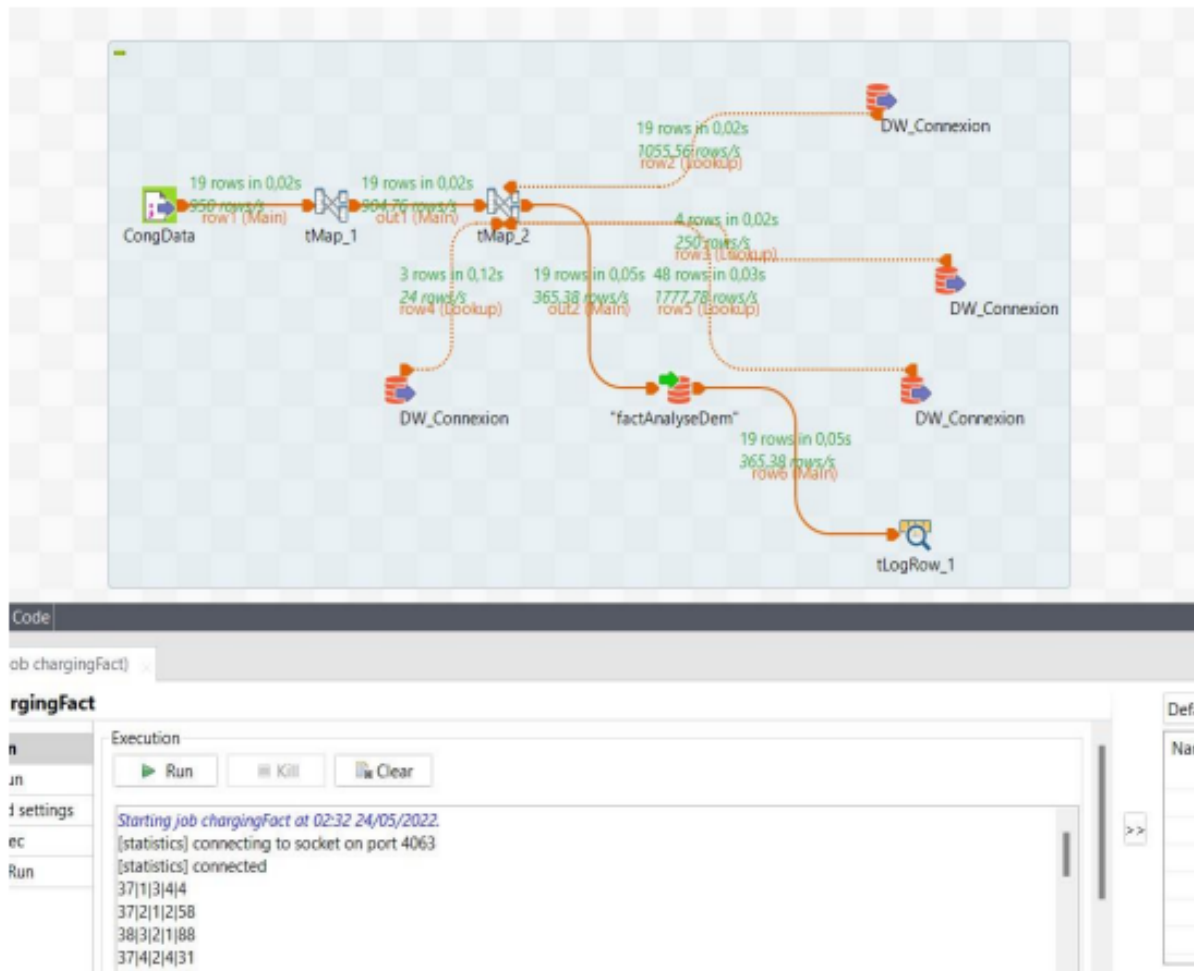
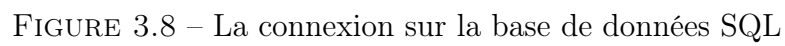


FIGURE 3.7 – Job chargement de table de fait

À partir de ce tableau de bord, nous pouvons avoir des informations sur l'état des projets, ses avancements, comprendre combien des projets ont été fait par chef de projet et par mois.

D'autres indicateurs sont également aussi importants que les précédents. Les employés au sein de l'entreprise rencontrent par fois des problèmes donc ils peuvent passer des réclamations. La figure 3.12 montre le tableau de bord des réclamations. Les dirigeant peut comprendre les utilisateurs, les mois qui ont le plus nombre des réclamations, et comprendre les statuts et les type de ses demandes.

Le dernier tableau de bord que nous allons implémenter est celui qui affiche le nombre de demande d'achat et la quantité totale d'achat par statuts, par mois, par employé et par produit afin d'aider le dirigeant de Détecter les employées, les mois qui ont le plus nombre de demande d'achat, et les quantités des produits les plus demandés.



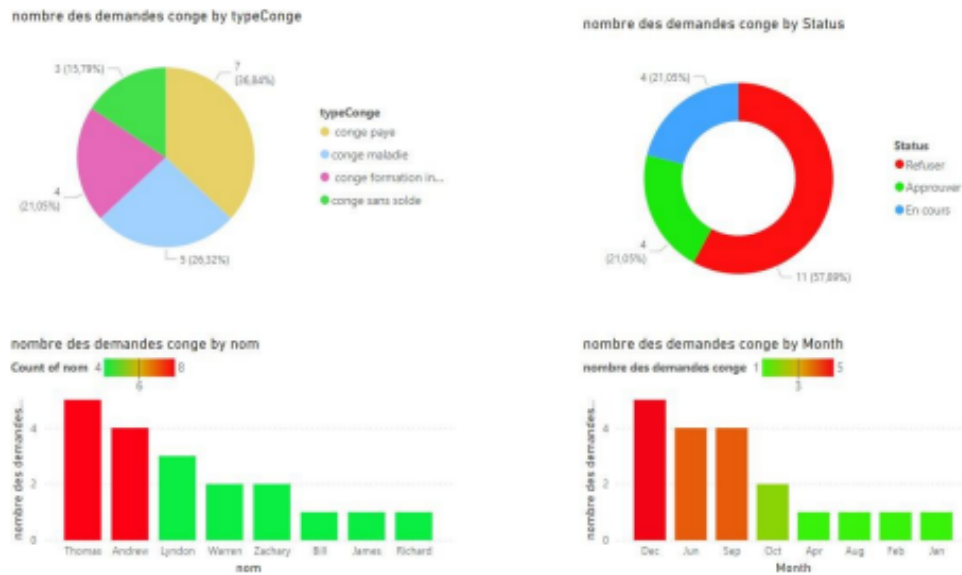


FIGURE 3.10 – Tableau de bord module demande de congé

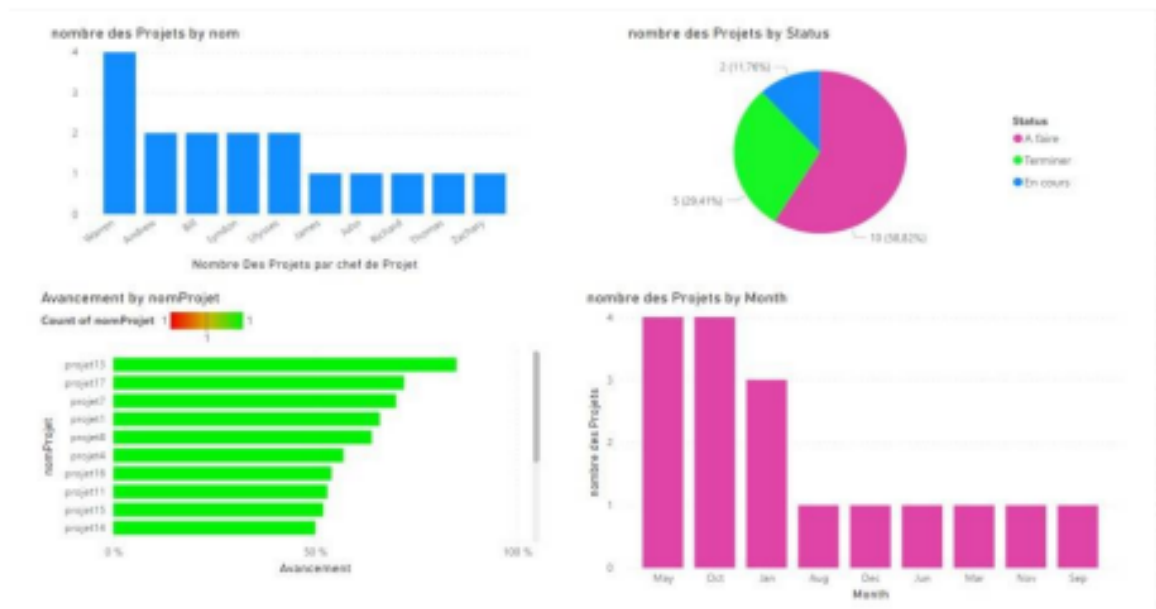


FIGURE 3.11 – Tableau de bord module Projet

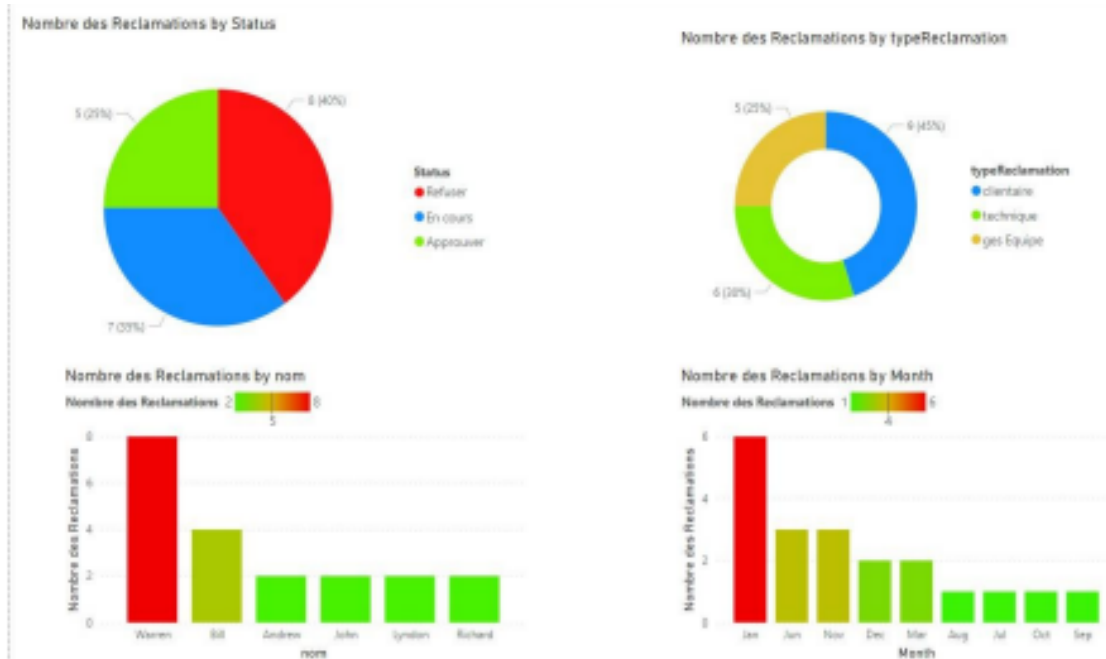


FIGURE 3.12 – Tableau de bord Module Réclamation

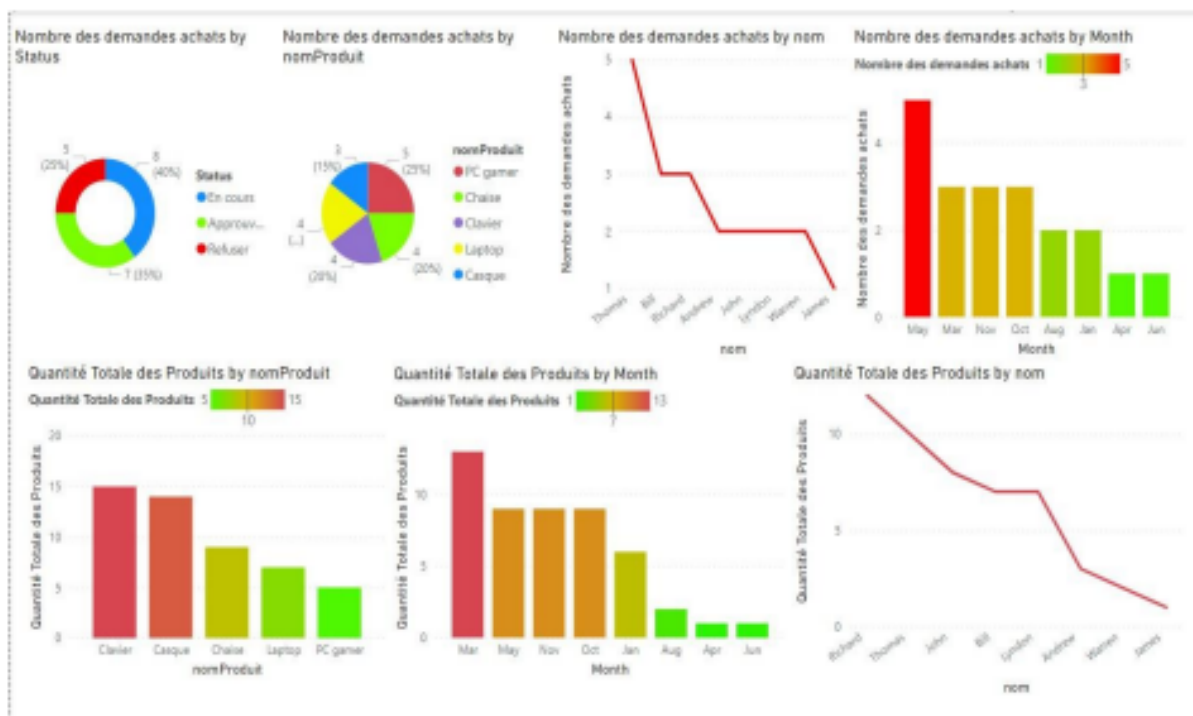


FIGURE 3.13 – Tableau de bord module demande achat