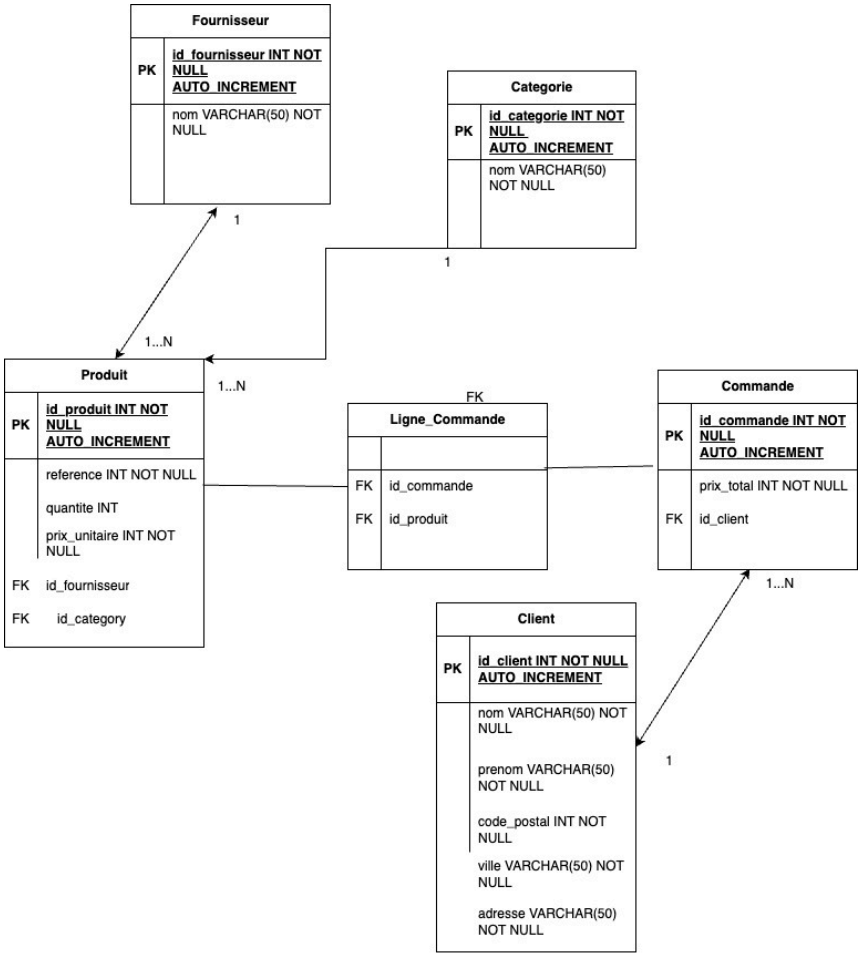
# Examen Base de données

---

## Audit V1 :

Ceci est un rapport des fonctionnalités de la base de données mise en place pour l'entreprise AvionPapier avec un focus sur les failles sécurité et manque d'intégrité, les risques qu'ils entraînent ainsi que les solutions possibles.

Ainsi le code écrit en JS permet la mise en place d'un serveur Express et la connexion à une BDD mySql. Voici une modélisation type MCD/MLD de la base.




En plus de la mise en place des différentes tables et des clés primaires/étrangères, des APIs ont également été créées. On y retrouve des APIs de type GET, POST, PUT et DELETE.

Il est important de noter que les APIs ont ici une faille intrinsèque à leur conception. En effet les requêtes SQL ont été formées par concaténation, ce qui ouvre la possibilité d'**injections SQL.**

Une injection SQL est une attaque où un utilisateur malveillant insère du code SQL dans une requête non sécurisée pour manipuler ou voler des données. Cela se produit lorsqu'une entrée utilisateur est directement concaténée dans une requête SQL au lieu d'être paramétrée.

Par exemple, l'endpoint :
**/client/:id/commandes** est vulnérable à l'injection SQL et construit la requête comme ceci :

**const query = 'SELECT * FROM Client WHERE id_client = ' + req.params.id;**

Un attaquant peut entrer cette valeur dans l'URL :
**/client/1 OR 1=1**

Ce qui transforme la requête en :

**SELECT * FROM Client WHERE id_client = 1 OR 1=1;**

**Résultat : tous les clients de la base de données sont retournés, exposant ainsi des données sensibles.**

En plus de ces failles de sécurité, il y a un manque d'intégrité des données, notamment dû à l'absence de validation des champs et de vérification des types de données envoyées. Par exemple, un utilisateur pourrait enregistrer une quantité négative en stock ou saisir du texte à la place d'un prix, ce qui entraînerait des incohérences dans la gestion des commandes.

Pour conclure, cette première version de la BDD répond aux attentes purement fonctionnelles mais contient des failles de sécurité critiques, interdisant sa mise en œuvre dans l'état actuel. Afin de pouvoir corriger les problèmes je vous propose une V2, complètement sécurisée, notamment par la mise en place de requêtes paramétrées et du PoLP (Principle of Least Privileges) Cependant, la mise en œuvre de cette V2 prendra du temps et beaucoup de recherche, je compte donc sur votre professionnalisme.