

Antoine CALDICHOURY

Alexandra CHUVATIN

TD P

# Rapport de projet BDD : *VeloMax*

Ce rapport servira de note technique explicative sur notre projet pour l'entreprise *VeloMax* dont l'objectif était de créer une nouvelle base de données afin de mieux gérer les ventes et les clients de celle-ci.

Nous allons donc expliquer dans cette note nos choix techniques, notamment l'organisation de nos base de données, les fonctionnalités WPF qui permettent l'utilisation et la gestion totale de cette base de données ou encore les développements complémentaires que nous avons pu faire.

## I) Organisation de la base de données

Nous avons créé une base de données qui permet de stocker toutes les informations demandées par M. Legrand. Celle-ci contient un total de 10 tables listées dans le schéma relationnel ci-dessous :

### Schéma relationnel :

**Modele**(num\_modele : varchar(40), prix\_modele : float, ligne\_produit : varchar(40), nom\_modele : varchar(40), grandeur : varchar(40), date\_intro\_modele : datetime, date\_disc\_modele : datetime, cadre : varchar(40), guidon : varchar(40), freins : varchar(40), selle : varchar(40), derailleur\_avant : varchar(40), derailleur\_arriere : varchar(40), roue\_avant : varchar(40), roue\_arriere : varchar(40), reflecteurs : varchar(40), pedaliere : varchar(40), ordinateur : varchar(40), panier : varchar(40), stock\_modele : int)

**Piece**(num\_piece : varchar(40), description : varchar(40), nom\_fournisseur : varchar(40), num\_produit\_catalogue : varchar(40), prix\_piece : float, date\_intro\_piece : datetime, date\_disc\_piece : datetime, delai : varchar(40), stock\_piece : int)

**Commande**(num\_commande : int, date\_commande : datetime, adresse\_livraison : varchar(40), date\_livraison : datetime, #nom\_individu : varchar(40), #nom\_boutique : varchar(40))

**Contient\_modele**(#num\_comande : int, #num\_modele : varchar(40), quantite\_modele : int)

**Contient\_piece**(#num\_comande : int, #num\_piece : varchar(40), quantite\_piece : int)

**Boutique**(nom\_boutique : varchar(40), adresse\_boutique : varchar(40), tel\_boutique : varchar(40), mail\_boutique : varchar(40), nom\_contact : varchar(40), remise : float)

**Individu**(nom\_individu : varchar(40), prenom\_individu : varchar(40), adresse\_individu : varchar(40), tel\_individu : varchar(40), mail\_individu : varchar(40), date\_adhesion : datetime, #num\_programme : int)

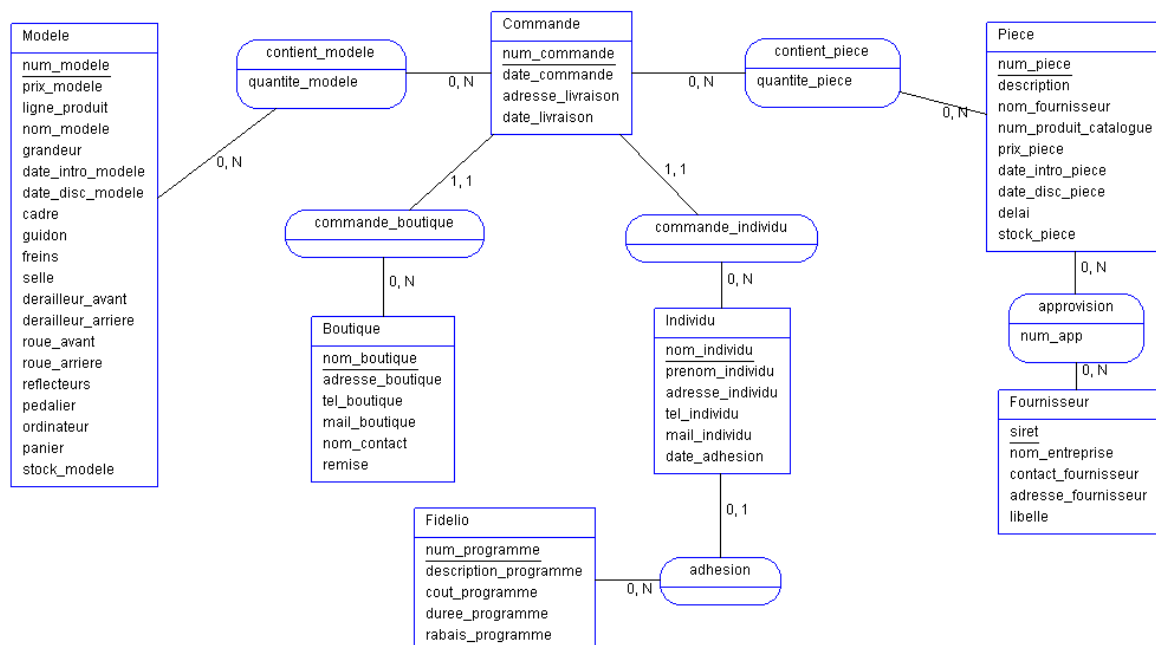
**Fournisseur**(siret : varchar(40), nom\_entreprise : varchar(40), contact\_fournisseur : varchar(40), adresse\_fournisseur : varchar(40), libelle : int)

**Fidelio**(num\_programme : int, description\_programme : varchar(40), cout\_programme : float, duree\_programme : varchar(40), rabais\_programme : float)

**Approvision**(#num\_piece : varchar(40), #siret: varchar(40))

Ce schéma relationnel a été créé à partir du modèle E/A ci-joint :

Modèle E/A :



Un des principaux points sur lequel nous avons beaucoup réfléchi a été la table **Modele**. En effet, il fallait pouvoir récupérer la liste des pièces nécessaires à l'assemblage de chaque modèle de bicyclette. Nous avons finalement opté de faire en sorte qu'un modèle soit caractérisé par les informations telles que le numéro de modèle (en clé primaire), le prix, ... mais aussi par l'ensemble des pièces qui peuvent constituer un vélo. Si par exemple un vélo ne contient pas de panier, celui-ci sera de type *null*.

Ensuite, pour la table **Individu**, nous avons une clé étrangère *num\_programme* qui désigne le programme auquel le client adhère (1, 2, 3 ou 4). Cependant, un client peut également choisir de ne pas adhérer à un programme. Dans ce cas, la valeur de *num\_programme* sera à 0.

La même situation apparaît pour la table **Commande** qui contient 2 clés étrangères : *nom\_boutique* et *nom\_individu* car une commande peut être effectuée par un client particulier ou bien par un client entreprise. Dans le cas où il s'agit d'une commande faite par une boutique, c'est *nom\_individu* qui sera *null*, et vice-versa.

Finalement, un autre point dont il est important de parler concerne également la table **Commande**. Pour connaître les pièces et/ou modèles qu'une commande contient, cette table est reliée par les associations **Contient\_modele** et **Contient\_piece** qui sont en fait des tables de notre BDD, identifiées par un numéro de commande, un numéro de pièce/modèle et une quantité. C'est donc grâce au numéro de commande que nous pouvons accéder au contenu de chaque commande.

## II) Projet WPF

Pour ce qui est du projet WPF qui permet l'utilisation de la BDD de *VéloMax* et sa gestion, nous allons dans cette partie présenter les principaux points nécessaires à la compréhension du fonctionnement de nos fenêtres WPF.

La première page qui s'ouvre, la **MainWindow**, permet de sélectionner l'utilisateur qui souhaite se connecter (M. Legrand qui aura accès à la BDD en root/root ou encore Bozo qui, comme son nom l'indique, se connectera en bozo/bozo). Cela mène ensuite à la fenêtre **Menu**. C'est la page principale. Via les différents boutons proposés, elle permet d'accéder à toutes les pages de gestion (Gestion Clients, Fournisseurs, Commandes, Stocks, Pièces et Modèles), ainsi qu'à la fenêtre Statistiques. Chaque fenêtre de gestion affiche le contenu de la table en question ainsi que des boutons permettant de modifier, supprimer ou créer l'élément voulu. Des boutons supplémentaires « Export JSON » et « Export XML » permettent l'export de la table dont il est question en JSON ou XML. La création ainsi que la modification d'un élément amènent à la fenêtre **Modif\_« nom de l'élément »** qui contient des *TextBox* permettant de modifier ou créer l'élément. Ce principe s'applique à toutes les fenêtres de gestion. Cependant **Modif\_Commande** (qui permet de modifier le contenu d'une commande) mène également à la fenêtre **Modif\_Contenu\_Commande**, qui permet d'ajouter, modifier, supprimer le contenu de **Contient\_Piece** et **Contient\_Modele** associés à la commande.

En ce qui concerne la fenêtre **Statistiques**, elle permet de récupérer les quantités vendues de chaque item de l'inventaire de *VéloMax*, la liste des membres par programmes d'adhésion, ainsi que les statistiques suivantes : moyennes de pièces et modèles par commande, total de quantités vendues et somme total des ventes de *VéloMax*. Il est également possible grâce au bouton « Export JSON » d'exporter les clients dont l'abonnement arrive à terme ou expire dans le format JSON.