**PROJET**

**BASES DE DONNEES RELATIONNELLES 2**

**L2 MIASHS, parcours MIAGE, Université Paris Nanterre**

**Semestre 4**

**Année scolaire 2022/2023**

**Romain THOMAS - 41005717**

**Kévin SOARES - 41001212**

**1/ Choix du scénario**

Notre base de données décrit le fonctionnement d’un site Web de vente de chaussures.

Un client possède un identifiant unique ainsi qu’un nom, prénom, date de naissance, adresse, email, numéro de téléphone et un mot de passe.

Un produit possède un identifiant unique, un nom, une marque, un prix, et une description.

Chaque client possède un seul panier et un panier n’appartient qu’à un seul client.

Chaque modèle différent d’un même produit est stocké dans une table et possède une couleur et l’emplacement mémoire de l’image correspondante. Un produit peut avoir plusieurs modèles mais un modèle n’appartient qu’à un seul produit.

Une table Articles\_panier contient la taille et la quantité des articles présents dans chaque panier, ainsi que des références aux tables Paniers, Produits et Modèles.

Un produit peut être présents dans plusieurs paniers mais un article d’un panier ne fait référence qu’à un seul produit. Un panier peut contenir plusieurs articles.

Un panier est créé automatiquement dès qu’un nouveau client est enregistré, et est associé à ce client automatiquement.

**2/ Modèles E/A et en Relation :**

**Modèle E/A :**

**Une image contenant diagramme

Description générée automatiquement**

**Modèle en relation :**

Une image contenant diagramme

Description générée automatiquement

**3/ Normalisation :**

On voit que les relations de ce modèle sont normalisées en FN1 car chaque attribut n’est présent qu'une seule fois par table et n'est pas décomposable en d'autres attributs. De plus, aucun attribut n'est lui-même une relation.

Les relations sont également normalisées en FN2 car tous les attributs des tables dépendent de la totalité de la clé de leur table :

Clients : ID\_Client -> {Nom, Prénom, Date\_N, Adresse, Email, Tel, Mdp, Ref\_panier}

Articles\_panier : -> {Taille, Quantité, Ref\_produit, Ref\_modele, Ref\_panier}

Produits : ID\_Produit -> {Nom, Marque, Prix, Description}

Modeles : ID\_modele -> {Couleur, Emplacement\_img, Ref\_produit}

Pour finir on peut montrer qu'elles sont normalisées en FN3 car aucun attribut des tables ne dépend de la clé primaire par dépendance fonctionnelle transitive, ils y dépendent tous directement.

**4/ SQL :**

**Création des tables :**

CREATE DATABASE Vente\_chaussures;

USE Vente\_chaussures;

CREATE TABLE IF NOT EXISTS Clients(

ID\_clients INT AUTO\_INCREMENT PRIMARY KEY,

Nom VARCHAR(50),

Prenom VARCHAR(50),

Date\_n date,

Adresse VARCHAR(300),

Email VARCHAR(300),

Tel INT,

Mdp VARCHAR(50),

Ref\_panier INT REFERENCES Paniers(ID\_panier)

);

CREATE TABLE IF NOT EXISTS Paniers(

ID\_panier INT AUTO\_INCREMENT PRIMARY KEY

);

CREATE TABLE IF NOT EXISTS Articles\_panier(

ID\_article INT AUTO\_INCREMENT PRIMARY KEY,

Taille INT,

Quantite INT,

Ref\_produit INT REFERENCES Produits(ID\_produit),

Ref\_modele INT REFERENCES Modeles(ID\_modele),

Ref\_panier INT REFERENCES Paniers(ID\_panier)

);

CREATE TABLE IF NOT EXISTS Produits(

ID\_produit INT AUTO\_INCREMENT PRIMARY KEY,

Nom VARCHAR(300),

Marque VARCHAR(300),

Prix INT,

Descrip VARCHAR(500)

);

CREATE TABLE IF NOT EXISTS Modeles(

ID\_modele INT AUTO\_INCREMENT PRIMARY KEY,

Couleur VARCHAR(300),

Emplacement\_img VARCHAR(300),

Ref\_produit INT REFERENCES Produit(ID\_produit)

);

**Insertion des données :**

USE Vente\_chaussures;

INSERT INTO Clients (`Nom`, `Prenom`, `Date\_n`, `Adresse`, `Email`, `Tel`, `Mdp`, `Ref\_panier`) VALUES

(‘Soares’, ‘Kevin’, ‘2003-08-07’, ’1 rue de Paris’, ‘ksoares@gmail.com’, ’0689442211’, ’motdepasse’, ‘1’);

INSERT INTO Clients (`Nom`, `Prenom`, `Date\_n`, `Adresse`, `Email`, `Tel`, `Mdp`, `Ref\_panier`) VALUES

(‘Thomas, ‘Romain, ‘2004-01-21’, ’12 rue des Champs Elysees, ‘romainthomas@gmail.com’, 0755448822, passwrd123, ‘2’);

INSERT INTO Clients (`Nom`, `Prenom`, `Date\_n`, `Adresse`, `Email`, `Tel`, `Mdp`, `Ref\_panier`) VALUES

(‘Dupont, ‘Jean, ‘2001-04-03’, ’3 avenue des quais, ‘jeandupont@orange.fr’, ’0124560486’, ’Jdup01’, ‘3’);

INSERT INTO Paniers (`ID\_panier`) VALUES (‘1’);

INSERT INTO Paniers (`ID\_panier`) VALUES (‘2’);

INSERT INTO Paniers (`ID\_panier`) VALUES (‘3’);

INSERT INTO Articles\_panier ( `Taille`, `Quantite`, `Ref\_produit`, `Ref\_modele`, `Ref\_panier`) VALUES (‘42’, ‘2’, ‘1’, ‘1’, ‘2’),

(‘43’, ‘5’, ‘1’, ‘2’, ‘1’),

(‘40’, ‘1’, ‘3’, ‘2’, ‘7’),

(‘39’, ‘3’, ‘2’, ‘3’, ‘1’);

INSERT INTO Produits (`Nom`,`Marque`,`Prix`,`Description`) VALUES

(‘Air Force 1’, ‘Nike’, ‘120’, ‘Chaussure iconique depuis des années…’),

(‘Dunk Low, ‘Nike’, ‘100, ‘Inspiré des Air Force…’),

(‘Stan Smith, ‘Adidas, ‘70, ‘Mondialement connu, ce modèle…’) ;

INSERT INTO Modeles (`Couleur`, `Emplacement\_img`, `Ref\_produit`) VALUES

(‘Vert’, ‘IMG/SHOES/modele1.png’, ‘1’),

(‘Bleu, ‘IMG/SHOES/modele2.png’, ‘1’),

(‘Vert’, ‘IMG/SHOES/modele3.png’, ‘2’),

(‘Rouge, ‘IMG/SHOES/modele4.png’, ‘3’),

(‘Violet, ‘IMG/SHOES/modele5.png’, ‘3’);

**5/ Vues :**

**1ère vue : Cette vue permet de récupérer les informations des produits qui ont déjà été commandés par quelqu’un. Elle pourra servir, par exemple, à connaitre les noms des clients qui ont commandé un certain produit :**

CREATE VIEW vue1 AS

SELECT \* FROM Produits P

INNER JOIN Articles\_panier A ON P.ID\_produit=A.Ref\_produit

WHERE Quantite > 0 ;

**2ème vue : Cette vue permet de récupérer toutes les couleurs de chaussures existantes pour chaque marque. Elle peut, par exemple, à déterminer quelle couleur de produit est la plus commandée :**

CREATE VIEW vue2 AS

SELECT DISTINCT Couleur FROM Modeles M

INNER JOIN Produits P ON M.Ref\_produit=P.ID\_produit

GROUP BY P.Marque;

**6/ Transactions:**

Il peut y avoir des problèmes de cohérence dans notre base de données.

Le 1er peut arriver pendant la commande d’un produit. Le scénario est le suivant :

Un client sélectionne un produit présent sur le site. Pendant qu’il est en train de sélectionner les informations nécessaires à sa commande (taille, quantité, paiement…), le produit choisit est retiré du site. Le client termine sa commande, mais le produit n’est plus disponible. Le client a payé pour quelque chose qui n’est plus là. Incohérence.

Le 2ème problème peut survenir lors de l’inscription d’un nouveau client. Scénario :

Deux clients créent un compte en même temps. Le panier du premier est créé automatiquement mais le commit n’a pas encore eu lieu. Le panier du second est créé aussi, mais le commit n’ayant pas eu lieu, il prend le même identifiant que le premier. Les deux commit sont faits, il y a un problème de cohérence car le premier client va posséder deux paniers et le second n’en aura pas, alors que chaque client doit avoir absolument un panier, pas 0 ni 2.

**7/ Utilisateurs :**

**Création d’un utilisateur avec un rôle « administrateur » :**

CREATE ROLE administrateur;

GRANT ALL PRIVILEGES ON \*.\* TO 'admin\_user'@'localhost'

IDENTIFIED BY 'motdepasse1';

GRANT administrateur TO 'admin\_user'@'localhost';

FLUSH PRIVILEGES;

**Création d’un utilisateur avec un rôle « utilisateur\_classique » auquel on n’accorde l’accès qu’au commandes SELECT, INSERT, UPDATE, DELETE sur la table « ma\_table » :**

CREATE ROLE utilisateur\_classique;

GRANT SELECT, INSERT, UPDATE, DELETE ON ma\_table.\* TO 'standard\_user'@'localhost'

IDENTIFIED BY 'motdepasse2';

GRANT utilisateur\_classique TO 'standard\_user'@'localhost';

FLUSH PRIVILEGES;