

## PROPOSITIONS

### Objectifs du Projet

Le projet vise à informatiser un service de ventes aux enchères, où les utilisateurs peuvent :

- Mettre en vente des produits.
- Acheter ou enchérir sur des produits.
- Participer à des ventes organisées par des salles de vente.

## ENTITÉS ET LEURS ATTRIBUTS

### Produit

- Attributs :
  - IdProduit (PK)
  - NomProduit
  - PrixRevient
  - Stock

### Catégorie

- Attributs :
  - NomCat (PK)
  - DescrCat

### SalleDeVente

- Attributs :
  - IdSalle (PK)
  - Catégorie

### Vente

- Attributs :
  - IdVente (PK)
  - TypeVente // offre montante ou descendante
  - PrixDépart
  - Révocable
  - LimiteOffres
  - Durée // Duree limitee ou durée libre
  - DateFinVente

### Utilisateur

- Attributs :
  - Email (PK)
  - Nom
  - Prenom
  - AdressePostale

### Caractéristiques

- Attributs :
  - NomCar (PK)
  - ValeurCar

## **Offre**

- Attributs :
- IdOffre (PK)
- PrixOffre
- DateOffre
- HeureOffre
- Quantité

## **RELATIONS (3FN) + Contraintes de Multiplicite**

### **Utilisateur (1..1) ----- (0..\*) Offre**

Cela signifie qu'un utilisateur doit exister pour soumettre une offre, mais qu'il peut soumettre plusieurs offres.

### **Offre (0..\*) ----- (1..1) Produit**

Un produit peut être proposé dans plusieurs offres, mais chaque offre doit être liée à un produit spécifique.

### **Produit (0..\*) ----- (1..1) Catégorie**

Chaque produit doit être classé dans une catégorie, mais une catégorie peut contenir plusieurs produits.

### **Produit (1..1) ----- (1..\*) Caractéristiques**

Chaque produit devrait avoir des caractéristiques pour décrire ses spécificités.

### **Produit (1..1) ----- (0..\*) Vente**

Un produit peut être vendu plusieurs fois, mais chaque vente doit référencer un produit.

### **Vente (1..\*) ----- (1..1) SalleDeVente**

Chaque vente doit être liée à une salle de vente.

### **SalleDeVente (0..,\*) ----- (1..1) Catégorie**

Une salle de vente peut être associée à une seule catégorie. Une catégorie peut avoir plusieurs salles.

## **Dépendances Fonctionnelles :**

### **Email → Nom, Prenom, AdressePostale**

- L'email détermine de manière unique le nom, le prénom et l'adresse postale d'un utilisateur.

### **IdProduit → NomProduit, PrixRevient, Stock, NomCat**

- L'IdProduit détermine de manière unique le nom, le prix de revient et le stock du produit, ainsi que la catégorie (NomCat).

**NomCat → DescrCat**

- Le NomCat détermine de manière unique la description de la catégorie.

**IdSalle → Catégorie**

- L'IdSalle détermine de manière unique la catégorie associée à la salle de vente.

**IdVente → TypeVente, PrixDépart, Révocable, LimiteOffres, Durée, IdProduit, IdSalle**

- L'IdVente détermine de manière unique le type de vente, le prix de départ, la révocabilité, les limites d'offres, la durée, ainsi que les références au produit et à la salle de vente.

**NomCar, IdProduit → ValeurCar**

- Le NomCar détermine de manière unique la valeur associée à une caractéristique.

**IdOffre → PrixOffre, DateOffre, HeureOffre, Quantité, Email, IdVente**

- L'IdOffre détermine de manière unique le prix de l'offre, la date et l'heure de dépôt, la quantité, l'email de l'utilisateur qui a fait l'offre, et l'IdVente concernée.

**Dépendances Fonctionnelles des Associations (1..1)**

**Utilisateur (1..1) - Offre (0..\*)**

- **IdOffre → Email**
  - Chaque utilisateur peut faire plusieurs offres, et chaque offre est liée à un utilisateur spécifique.

**Offre (0..\*) - Produit (1..1)**

- **IdOffre → IdProduit**
  - Chaque offre est associée à un produit, garantissant que chaque offre fait référence à un produit spécifique.

**Produit (0..\*) - Catégorie (1..1)**

- **IdProduit → NomCat**
  - Chaque produit appartient à une catégorie unique, établissant ainsi une relation de dépendance.

**Produit (1..1) - Caractéristiques (1..\*)**

- **IdProduit → NomCar**
  - Chaque produit peut avoir plusieurs caractéristiques, les reliant ainsi à un produit spécifique.

**Produit (1..1) - Vente (0..\*)**

- **IdVente → IdProduit**
  - Chaque vente fait référence à un produit spécifique.

### **Vente (1..\*) - SalleDeVente (1..1)**

- **IdVente → IdSalle**
  - Chaque vente doit se dérouler dans une salle de vente spécifique, établissant une relation de dépendance.

### **SalleDeVente (0..\*) - Categorie (1..1)**

- **IdSalle → NomCat**
  - Chaque salle de vente est associée à une catégorie, ce qui signifie que l'identifiant de la salle détermine de manière unique la catégorie.

### **Contraintes de valeur :**

Les contraintes de valeurs garantissent que les données saisies respectent certaines conditions de validité.

- $\text{PrixRevient} > 0$
- $\text{Stock} \geq 0$
- $\text{TypeVente} \in \{ \text{"montante"}, \text{"descendante"} \}$
- $\text{PrixDépart} > 0$
- $\text{Révocable} \in \{ \text{true}, \text{false} \}$  (booléen)
- $\text{LimiteOffres} \geq 0$
- $\text{PrixOffre} > 0$
- $\text{Quantité} \geq 0$

### **Contraintes Contextuelles**

Les contraintes contextuelles définissent des règles qui ne sont pas directement liées aux valeurs des attributs, mais qui sont importantes pour le fonctionnement logique de l'application.

- **Limite d'Offres :** Si  $\text{LimiteOffres}$  est !null, un utilisateur ne peut faire qu'un certain nombre d'offres sur un produit (par exemple, une seule offre par utilisateur si  $\text{LimiteOffres} = 1$ ).
- **Durée de Vente :** Une vente avec une durée limitée doit avoir une date et une heure de fin définies. Les ventes à durée libre doivent respecter un temps maximum entre deux offres (10 minutes).
- Pour une vente montante :  $\text{nouvelle\_offre.PrixOffre} > \text{offre\_precedente.PrixOffre}$
- Pour une vente descendante : première offre remporte la vente
- Si  $\text{vente.Duree} = \text{limitee}$ , la vente se termine à  $\text{DateFinVente}$  et toutes les offres après sont invalides.
- Si  $\text{Revocable} = \text{true}$  : la vente est annulée si  $\text{meilleure\_offre.PrixOffre} < \text{produit.PrixRevient}$ . (si le prix de revient du produit n'est pas atteint à la fin de la vente)
- La somme des quantités des offres gagnantes ne peut pas dépasser le stock du produit
- Une vente ne peut avoir lieu que si le stock du produit est  $> 0$

## **FONCTIONNALITÉS DU PROJET**

Le projet requiert des fonctionnalités spécifiques, telles que :

- Mise en place d'une salle de vente.
- Enchères sur des produits.
- Processus de fin d'enchère.