



# Notre application de ventes aux enchères

Helain Zimmermann, Abel Prud'homme Lacroix,  
Mateo Bedel, Nour El Hamidi, Nathan Guerin



# Sommaire

- **Introduction** : Les exigences de Baie-électronique
- La **base de donnée** produite :
  - les différentes tables
  - les relations et contraintes dans la base
- **L'interface homme machine**
  - la mise en place du menu
  - les fonctionnalités de l'application
- **Bilan**



# Dépendances Fonctionnelles

**Utilisateur** : EmailUtilisateur  $\rightarrow$  {Nom, Prenom, AdressePostale}

**Produit** : IdProduit  $\rightarrow$  {NomProduit, PrixRevient, Stock , NomCategorie}

**Catégorie** : NomCategorie  $\rightarrow$  Description

**Caractéristique** : {IdProduit, NomCaracteristique}  $\rightarrow$  ValeurCaracteristique

**Salle** : IdSalle  $\rightarrow$  {NomCategorie}

**Vente** : IdVente  $\rightarrow$  {IdProduit, IdSalle, PrixDepart, Sens, NbOffres, Revocabilite}

**Vente Limitée** : IdVente  $\rightarrow$  {Debut, Fin}

**Offre** : {EmailUtilisateur, IdVente, DateHeure}  $\rightarrow$  {PrixAchat, Quantite}

# Contraintes

Dépendances Fonctionnelles	Contraintes de Valeur	Contraintes de Multiplicité	Contraintes Contextuelles
EmailUtilisateur → {Nom, Prenom, AdressePostale}			
IdProduit → {NomProduit, PrixRevient, Stock, NomCategorie}	PrixRevient > 0	Email → Offre	Une vente à durée libre implique un délai maximal de 10 minutes entre deux offres
NomCategorie → Description	PrixDepart > 0	Produit → caractéristique	Ventes par défaut : Montantes, Non révocables, Sans limite de temps, Permettant plusieurs enchères par utilisateur
{IdProduit, NomCaracteristique} → ValeurCaracteristique	PrixAchat > 0	Vente → Offres	
IdSalle → {NomCategorie}	Quantite > 0	Salle → Produit	
IdVente → {IdProduit, IdSalle, PrixDepart, Sens, NbOffres, Revocabilite}	Fin > Debut	Catégorie → Produit	Vente montante: Le prix d'achat doit être plus grand que le prix d'achat de l'offre précédente (et que le prix de départ si c'est la première offre)
IdVente → {Debut, Fin}	DateHeure > Debut	Produit → Vente	
{EmailUtilisateur, IdVente, DateHeure} → {PrixAchat, Quantite}	Quantite > Stock	Vente → Salle	
		Vente → Produit	Utilisateur: peut être limité à une offre uniquement, ou en faire autant qu'il veut



# Normalisation

- FN1 : les attributs ont une valeur atomique
- FN2 : les attributs non clé dépendent entièrement de la clef
- FN3 : les attributs non clefs dépendent directement de cette clef
- FN3BCK : ils ne sont pas sources de dépendance fonctionnelle pour une partie de la clef

**Utilisateur** : EmailUtilisateur → {Nom, Prenom, AdressePostale}

**Produit** : IdProduit → {NomProduit, PrixRevient, Stock , NomCategorie}

**Catégorie** : NomCategorie → Description

**Caractéristique** : {IdProduit, NomCaracteristique} → ValeurCaracteristique

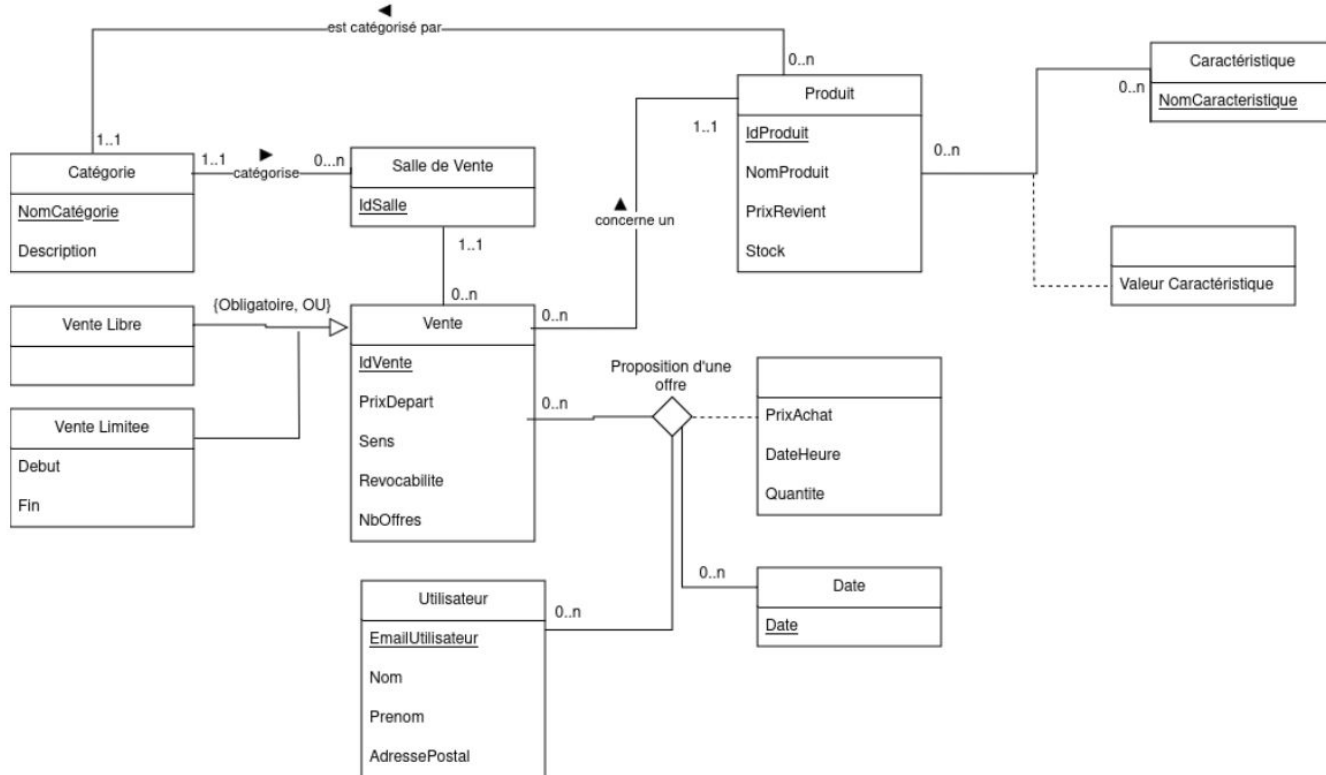
**Salle** : IdSalle → {NomCategorie}

**Vente** : IdVente → {IdProduit, IdSalle, PrixDepart, Sens, NbOffres, Revocabilite}

**Vente Limitée** : IdVente → {Debut, Fin}

**Offre** : {EmailUtilisateur, IdVente, DateHeure} → {PrixAchat, Quantite}

# Schéma entité-association





# Schéma relationnel

*Produit* : (**IdProduit**, NomProduit, PrixRevient, Stock, ...NomCategorie...)

*Salle de Vente* : (**IdSalle**, ...NomCategorie...)

*Vente* : (**IdVente**, PrixDepart, Sens, Revocabilité, NbOffres, ...IdSalle..., ...IdProduit...)

*Catégorie* : (**NomCategorie**, Description)

*Offre* : (...**EmailUtilisateur**..., ...**IdVente**..., ...**DateHeure**..., PrixAchat, Quantite)

*Date de l'offre* : ( **DateHeure** )

*VenteLimite* : (...**IdVente**..., DateDebut, DateFin)

*VenteLibre* : (...**IdVente**...)

*Utilisateur* : (**EmailUtilisateur**, Nom, Prenom, AdressePostale)

*Caracteristique* : (**NomCaracteristique**)

*Valeur\_Caracteristique* : (...**IdProduit**..., ...**NomCaracteristique**..., Valeur Caracteristique)

# Les fonctionnalités de l'application de ventes

Créer et mettre en place une salle de vente :

- Associer une catégorie de produit
- Associer des ventes du même type de vente que celles présentes dans la salle
- Vérifier que les produits dans la salle sont de la bonne catégorie
- Afficher la salle de vente créer dans les menus





# Les fonctionnalités de l'application de ventes

Faire une offre :

- Vérifier que la vente visée est toujours ouverte (en comparant les dates)
- Vérifications sur le prix d'achat
- Vérifier que l'utilisateur n'a pas dépassé le nombre d'offres limite
- Ajouter ou non l'offre à la vente



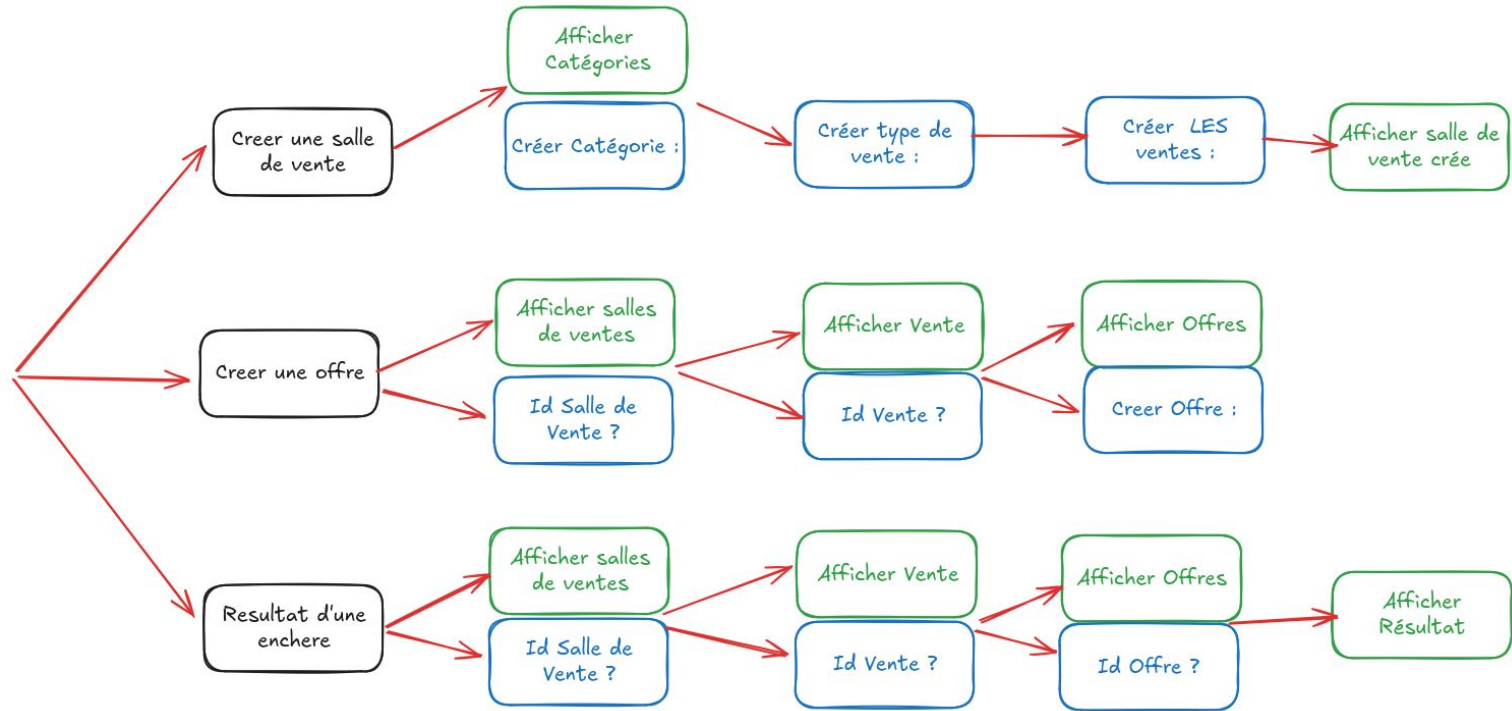
# Les fonctionnalités de l'application de ventes

Afficher le résultat d'une vente :

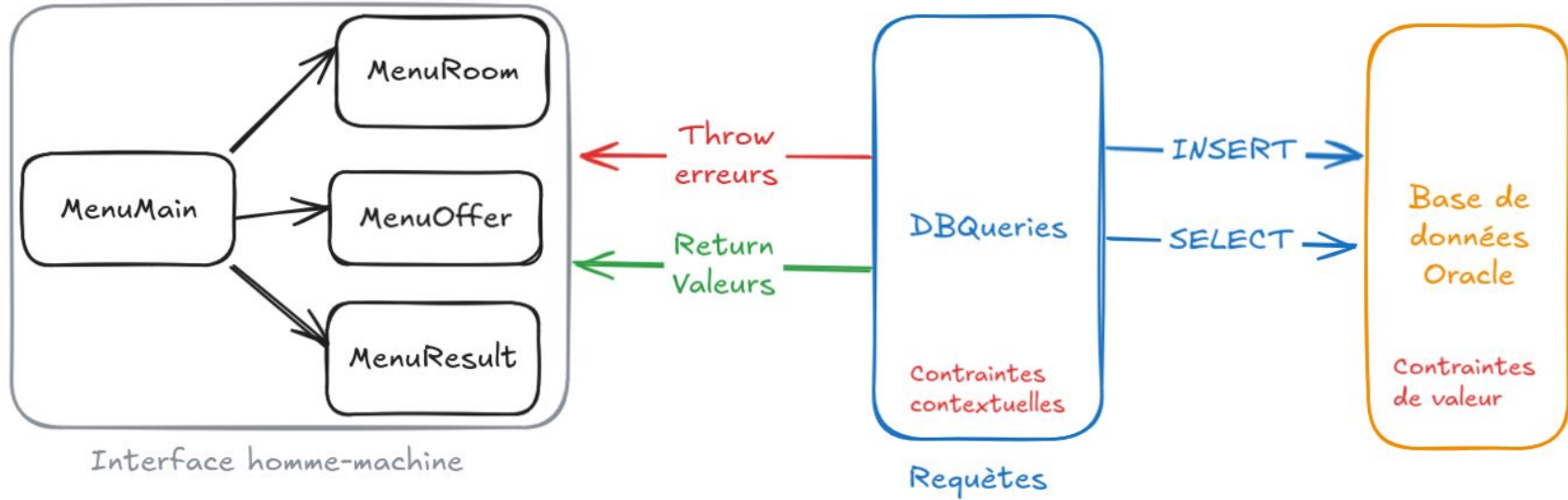
- Vérifier que l'enchère soit bien finie
- Verifie que des ventes ont eux lieux
- Récupérer la dernière offre faite sur la vente (ou l'unique offre si descendante)
- Vérifier la révocabilité de la vente



# L'interface Homme-machine



# Implémentation java





# Concurrence

- Transactions avec commits et rollbacks
- Isolation des transactions ( Niveau SERIALIZABLE )
- Gestion des conflits via retry pour salle de vente, produit et vente
- Verrouillage implicite pour les Sélections



# Limite de l'application

- Pas de réduction automatique des ventes descendantes toute les minutes
- Pas d'ajouts de caractéristique & valeurs depuis l'interface.
- Pas de problème du sac à dos pour avoir la combinaison d'offre optimale pour les ventes montantes (on se contente de prendre la plus grande).
- Contrainte sur le nombre d'offre maximum pas implémenté.

A blue parallelogram and a light green parallelogram are positioned in the upper-left corner of the slide. The blue shape is on the left, and the green shape is to its right, partially overlapping it. Both shapes are oriented diagonally, with their longer sides running from the top-left towards the bottom-right.

# Bilan