

Choix de conception

- Utilisation d'une interface

L'interface Vendable est introduite pour permettre à différents types d'objets vendables (comme Disque et Nourriture) d'avoir un comportement commun : la capacité à retourner un prix unitaire via la méthode getPrix(). Cela permet d'éviter l'héritage direct et on assure l'extensibilité (on peut facilement ajouter d'autres types de produit)

- La classe Produit

Une classe Produit est introduite pour encapsuler la notion de quantité et permettre le calcul du prix total d'un produit en multipliant son prix unitaire par sa quantité.

Chaque classe gère son propre calcul de prix, ce qui favorise une meilleure maintenance et une évolution facilitée.

- Relation et cardinalités

Produit – Panier -> Relation 1 à N

Un Panier peut contenir 1 ou plusieurs Produit : 1..*

Un Produit ne peut exister qu'une seule fois dans un panier : 0..1

Un Produit dépend faiblement d'un panier : agrégation

Produit – vendable -> Relation 1 à 1

Un Produit référence exactement un objet Vendable

Diagramme de séquence

Le Panier appelle getPrixTotal().

Pour chaque Produit :

- Le Panier appelle getPrixProduit().
- Le Produit appelle getPrix() sur son Vendable (Disque ou Nourriture).
- Le Vendable retourne son prix unitaire.
- Le Produit calcule quantité * prix unitaire et retourne le prix total.

Une fois la boucle terminée, le Panier additionne tous les retours des produits et renvoie directement la somme totale.