

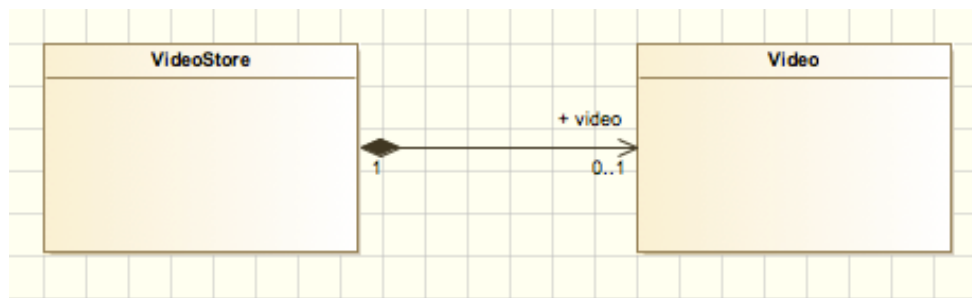
## UEF 7.3. Patrons De Conception

### TD/TP 1

### Patrons GRASP

#### Exercice 1 :

Considérons un magasin de vidéos qui comporte un ensemble de vidéos.



1. Quelle classe est responsable de l'instanciation d'une nouvelle vidéo dans le magasin? Quel patron GRASP avez vous appliqué?
2. A quelle classe peut on assigner la responsabilité de lister toutes les vidéos du magasin?

#### Exercice 2 :

Dans un système de gestion de vente, une facture comporte un tableau nommé `_LineItems`, où chaque `LineItem` comporte un item et la quantité du produit facturé (`Quantity`), et chaque item comporte l'identifiant du produit (`ProductID`), son nom (`ProductName`), et le prix unitaire (`UnitPrice`). Nous supposons que chaque item apparaît une seule fois: si un item est ajouté alors qu'il existe déjà dans le tableau `_LineItems` sa quantité est incrémentée.

Pour calculer le prix total, nous devons appliquer la formule suivante:

$$\text{Prix Total} = \text{Somme de (Prix Unitaire} \times \text{Quantité par LineItem)}$$

En plus de la classe facture, le système comporte les classes `Client` qui effectue les commandes, et `Paiement` qui décrit un mode de paiement (espèces, chèque, carte bancaire, à crédit, ...) où chaque Facture est associée à un mode de paiement. Notons que lorsqu'il s'agit d'un paiement par carte bancaire, le système doit se connecter au serveur bancaire associé à la carte pour valider la transaction. Si la facture est payée, elle est enregistrée dans la base de donnée avec l'état « payée », sinon elle est enregistrée avec l'état « en instance ». Le gestionnaire de vente peut mettre à jour son état ou la supprimer par la suite à travers l'interface du système qui lui permet d'effectuer les différents traitements (ajouter facture, ajouter une ligne d'items, etc.).

1. Donnez un diagramme de classe à ce système.
2. Sachant que deux items sont identiques s'ils ont des identifiants identiques, quel est l'objet qui est responsable de déterminer que deux items sont équivalents? Quel patron GRASP devez vous appliquer pour répondre à cette question?
3. A quelle(s) classe(s) devons nous assigner la responsabilité de calculer le total du prix ?
4. On ajoute une méthode `payer()` à la classe `Client`. Etudiez le couplage et la cohésion dans les deux cas suivants :
  - A. La méthode `payer()` crée une instance de `Paiement` et l'assigne à l'objet `Facture`.
  - B. La méthode `payer()` délègue l'action à la classe `Facture` qui crée une instance de `Paiement` et l'assigne.
5. Comment peut-on implémenter la vérification des paiements à l'aide de patrons GRASP? Justifiez.

## UEF 7.3. Patrons De Conception

6. Quel patron GRASP permet de prendre en charge les différents traitements émis par l'utilisateur à travers l'interface du système?
7. Modifiez votre diagramme de classe pour prendre en charge toutes ces spécifications.

**Exercice 3 : (CI 2016)** Soit le code suivant :

```
public class A {
    private ArrayList<B> listB ;
    private G g ;
    ..
    public void affich(C c, B b){
        if (b.getC().getD().getInf())>c.getD().getInf())
            System.out.print(b.getC().getD().getInf());
        else System.out.print (c.getD().getInf());
    }
    public boolean exist(G g){
        foreach (B b in listB)
            if (b.getG().equals(g)) return true;
        return false ;
    }
    public void addB(G g) {
        if ( ! exist(g))
            listB.add(new B(g));
    }
}

public class B implements IB {
    private C c;
    private G g ;
    public B(G g) {
        this.g = g;
    }
    ..
    // getG; getC;
    }
    public class C{
        private D d ;
        ...
        // getD; setD;
    }
    public class D{
        private int Inf ;
        ...
        // getInf()
    }
}
```

1. Ce code utilise déjà un patron GRASP, lequel ? Expliquez.
2. Ce code ne respecte pas des principes de conceptions et patrons GRASP, lesquels?
3. Quelle solution proposez vous? Donnez les modifications à apporter.

**Exercice 4 : (CI 2015)**

Un supermarché souhaite se doter d'un système de gestion des échanges d'articles. Pour bénéficier de ce service, le client doit présenter son ticket de caisse sur lequel sont inscrits un numéro unique, la date/heure des achats, le numéro de caisse, le montant total et l'ensemble des articles achetés. Un achat est caractérisé par l'article acheté et la quantité achetée de cet article. A chaque article sont associés un identifiant, un libellé, et un prix unitaire.

Certains articles du magasin ne sont pas échangeables. Pour les autres, le changement peut se faire dans un délai d'un mois à compter de la date de l'achat. Si un client désire échanger un produit il doit se présenter au service d'échange muni de son ticket. Le vendeur doit introduire le numéro du ticket et préciser les identifiants des articles concernés. Selon la date du ticket et le type de(s) l'article(s), une réponse sera fournie. Si l'opération est possible, la somme des articles à échanger peut être fournie comme un bon d'échange ou comme un solde qui sera ajoutée à sa carte de fidélité. Dans ce cas, une vérification de l'authenticité de la carte est déclenchée en invoquant le système de gestion de clients du supermarché.

1. Déterminez la (les) classe(s) à laquelle (aux quelles) on doit assigner la responsabilité de calculer la somme des articles à échanger. Expliquez le(s) patron(s) GRASP appliqué(s) pour répondre à cette question
2. Quel(s) patron(s) GRASP doit-on utiliser pour implémenter la vérification des cartes de fidélité? Expliquez.
3. Quel patron GRASP permet au vendeur d'interagir avec le système vendeur à travers l'interface du système?

### UEF 7.3. Patrons De Conception

---

4. Donnez le diagramme de classe de votre solution.

**Exercice 5 : (CI 2017)**

Nous souhaitons modéliser un annuaire téléphonique qui permet de gérer des contacts (ajouter, supprimer, modifier ou chercher un contact ou un numéro). Un contact possède un nom, une adresse, et une liste de numéros de téléphones. Les contacts peuvent être des personnes (ayant une civilité), ou des entreprises (avec adresse du site web et email). Chaque numéro de téléphone est caractérisé par son type (fixe professionnel, personnel, portable, fax, autre.) et le numéro en question. Lors de l'ajout, le formulaire doit vérifier si le numéro introduit est valide (données numériques en nombre variant de 10 à 15).

1. Nous proposons la classe ListNum pour implémenter la liste des numéros de téléphone d'un contact. Ce choix respecte-t-il les principes et patrons de conception ou ajoute une pure fabrication inutile ? Expliquez.
2. A quelle(s) classe(s) devons nous assigner la responsabilité de chercher un numéro de téléphone ? Quel patron GRASP avez vous appliqué?
3. Quelle classe est responsable de l'ajout d'un numéro de téléphone pour un contact existant ? Quel patron GRASP avez vous appliqué?
4. Donnez le diagramme de classe de l'annuaire en exposant les attributs et les méthodes. Quel autres patrons GRASP avez vous appliqué ?