

Explications du diagramme de notre classe pour le système de l'AL2000

Nous avons découpé le système en 3 sous parties / package :

L'IHM : Ce package est responsable des interactions avec l'utilisateur. Que ce soit les entrées, via souris / clavier, ou les sorties, via l'affichage à l'écran. Il se compose d'une classe abstraite `InterfaceUtilisateur`, qui sert de couche d'abstraction permettant de spécifier plusieurs type d'IHM. Ici, par soucis de simplicité et parce que le projet se concentre sur le noyau fonctionnel, nous n'avons représenté qu'une classe concrète d'interface textuelle.

Persistance : Ce package s'occupe de rendre les données pérennes dans le temps. Il faut pouvoir retrouver un utilisateur créé durant une exécution précédente. De même que pour l'IHM, le package met à disposition une couche d'abstraction permettant le choix de l'implémentation à utiliser lors de la création des objets. Nous avons simplement représenté une gestion simple par fichier.

Noyau Fonctionnel : C'est le cœur du problème, il représente le système que nous avons mis en place pour faire fonctionner l'AL2000. Comme il est plus complexe, nous allons détailler ci-après les différentes parties le composant :

Le cœur de du package est sa façade, **ViéoClub** : C'est la classe principale qui va servir de point d'entrée dans le package. Elle est responsable de mettre en relation les packages d'IHM et de persistance avec le noyau fonctionnel. Pour eux, elle est l'interface unique de communication, ce qui facilite les échanges.

Cette classe contiendra toutes les instances permettant le fonctionnement du système.

La fabrique :

Elle permet de dissocier la création des objets des objets en eux-mêmes. Elle fournit une couche d'abstraction permettant de faire varier les types concrets à instancier et la façon dont ils sont créés.

La filmothèque :

C'est « l'annuaire » des films. Elle est responsable de stocker les instances de tous les films et de rendre accessible les différents sous-groupes de films choisis.

Les utilisateurs :

Nous avons dissocié 3 types d'utilisateurs concrets :

- les utilisateursLambda sont les utilisateurs n'ayant pas de compte et qui loue un film uniquement et avec une carte bleue seulement.

- les AbonnéAdulte sont les utilisateurs ayant créé un compte, ayant des enfants ou non. Nous n'avons pas séparé les parents des non parents car leurs rôles sont sensiblement les mêmes. L'abonné devra se connecter avec sa carte et pourra louer jusqu'à 3 films.

Nous avons mis en place 2 niveau d'abstraction : le premier séparant les abonnés (enfant ou adulte) des utilisateurs lambda, et le second séparant les utilisateurs enfants et adulte. Ces 2 niveaux se basent sur le patron stratégie nous permettent de généraliser le code de ces 2 rôle

distinct, tout en nous offrant la possibilité d'étendre facilement le code en créant de nouveaux sous-types.

Locations :

Elles représentent les locations ayant été effectuées. Elles font le lien entre les films et les utilisateurs. Comme ce sont les utilisateurs qui effectuent les locations et que les prix varient suivant le type de l'utilisateur, nous avons choisi de stocker les locations dans les utilisateurs.

Techniciens :

Nous les avons représentés pour qu'ils ne soient pas oubliés dans les différents rôles mais il nous a été spécifié qu'il n'est pas de notre ressort de gérer les fonctionnalités de maintenance, soit les options du technicien.