

Design Patterns Implémentés :

Command

Gère les actions de création, validation et annulation des commandes de manière centralisée

+ Permet d'ajouter de nouvelles actions sans modifier les classes existantes, favorisant l'extensibilité et la centralisation des processus

Factory

Crée des commandes sans spécifier leur classe exacte

+ Simplifie la création d'objets et permet d'ajouter de nouveaux types de commandes sans modifier le code existant

Template

Définit un processus standard pour la génération de factures tout en permettant des implémentations spécifiques pour chaque format

+ Assure la réutilisation du code commun tout en permettant une personnalisation pour différents types de factures texte et PDF

State

Gère les différents états des commandes (brouillon, validée, annulée) et leur comportement

+ Facilite la gestion des transitions entre états, tout en maintenant des comportements spécifiques et séparés pour chaque état

Singleton

Assure qu'une seule instance d'une classe existe pour des ressources partagées ; le Stock

+ Garantit la cohérence des données en ayant une seule instance accessible

Observer

Notifie automatiquement les utilisateurs lorsque l'état du produit (quantité, prix) change

+ Permet un découplage entre les modules et une flexibilité pour ajouter de nouveaux observateurs

