

Projet : Librairie PHP pour les interfaces de paiement en ligne

Objectif

L'objectif de ce projet est de concevoir et de développer une librairie PHP robuste et extensible qui permettra d'interagir facilement avec différentes interfaces de paiement en ligne. Les étudiants devront mettre en pratique les principes SOLID et DRY, ainsi que des design patterns appropriés pour résoudre les problèmes rencontrés.

Contexte

Les paiements en ligne sont devenus monnaie courante dans le monde du commerce électronique. Il existe de nombreuses interfaces de paiement en ligne (comme PayPal, Stripe, etc.) et chacune d'entre elles a sa propre API et son mode de fonctionnement. Il peut donc être fastidieux pour un développeur de gérer plusieurs interfaces de paiement dans une même application.

Fonctionnalités attendues

La librairie devra fournir une interface unifiée pour effectuer des paiements avec différentes interfaces de paiement en ligne. Voici les fonctionnalités de base que la librairie devra proposer :

- Ajout et suppression d'interfaces de paiement prises en charge.
- Initialisation d'une interface de paiement avec ses identifiants de connexion.
- Création d'une transaction de paiement avec un montant, une devise et une description.
- Exécution d'une transaction de paiement et récupération de son résultat (succès, échec, etc.).
- Annulation d'une transaction de paiement en cours.

La librairie devra également proposer une fonctionnalité permettant de gérer le statut d'une transaction de paiement (en cours, réussi, échoué, annulé, etc.) et les transitions autorisées pour chaque état.

Les étudiants devront proposer un exemple d'utilisation de la librairie avec une interface de paiement de leur choix, par exemple Stripe.

Principes et design patterns à mettre en pratique

Les étudiants devront mettre en pratique les principes SOLID et DRY pour concevoir et développer la librairie. Ils devront également utiliser des design patterns appropriés pour résoudre les problèmes rencontrés. Voici quelques pistes à explorer :

- Pouvoir sélectionner dynamiquement l'interface de paiement à utiliser.
- Créer des transactions de paiement de manière flexible.
- Gérer le statut d'une transaction de paiement.
- Unifier les API des différentes interfaces de paiement.
- Notifier des tiers (comme le service de facturation, le service de gestion des stocks, etc.) de l'état d'une transaction de paiement.

Livrables

Les étudiants devront fournir les livrables suivants à la fin du projet :

- Le code source de la librairie et l'exemple d'implémentation.

Critères d'évaluation

Les étudiants seront évalués sur les critères suivants :

- La qualité de la conception et de l'architecture de la librairie.
- La qualité du code (respect des principes SOLID et DRY, utilisation de design patterns, etc.).
- Le respect des délais et des consignes.

Bon courage à tous pour ce projet !

Ressources

- <https://refactoring.guru/design-patterns>
- [https://fr.wikipedia.org/wiki/SOLID_\(informatique\)](https://fr.wikipedia.org/wiki/SOLID_(informatique))
- https://en.wikipedia.org/wiki/Don%27t_repeat_yourself