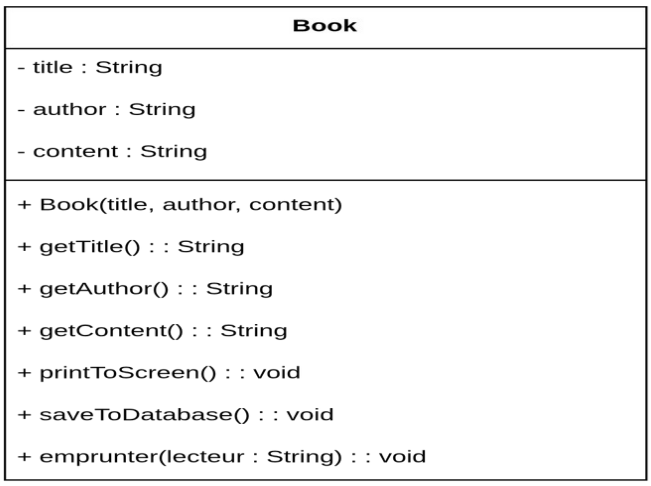
diagrammes de classes UML des differents principes SOLID :

SRP - Single Responsibility Principle

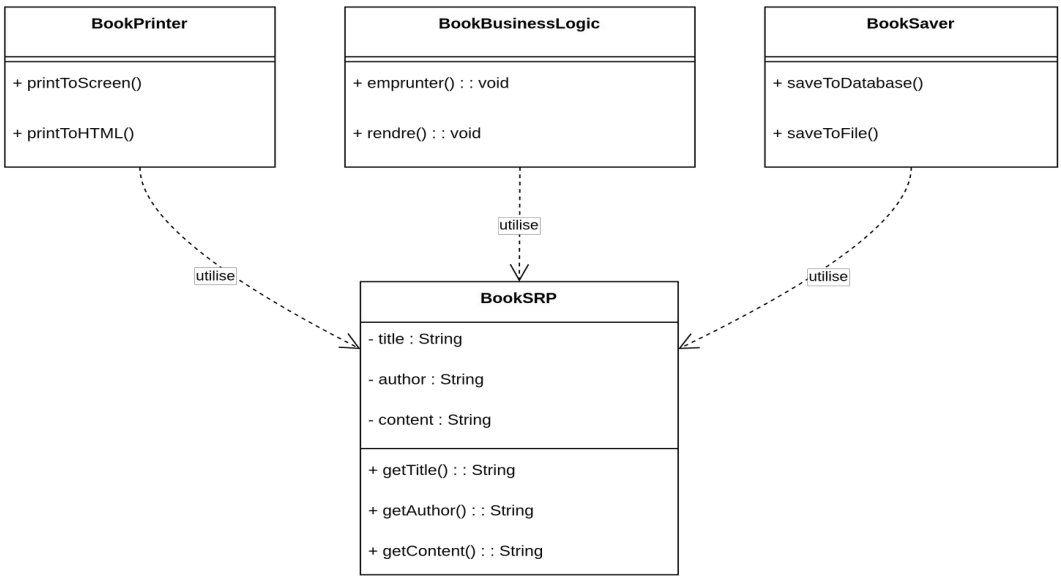
AVANT Refactoring

Problème : 4 responsabilités mélangées (données + affichage + persistance + métier)



APRÈS Refactoring

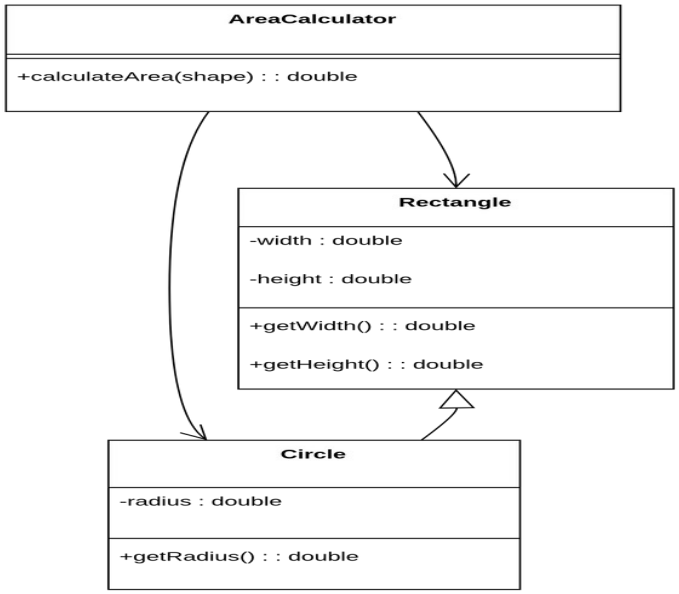
Solution : Chaque classe a une responsabilité unique.



OCP - Open/Closed Principle

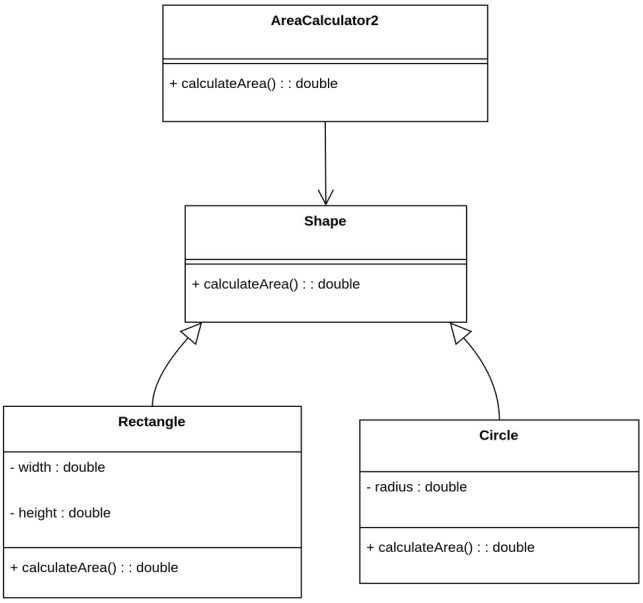
AVANT Refactoring (Page 25 du cours)

Problème : `calculateArea()` contient des `if (shape instanceof Rectangle)` et `if (shape instanceof Circle)`.



APRÈS Refactoring

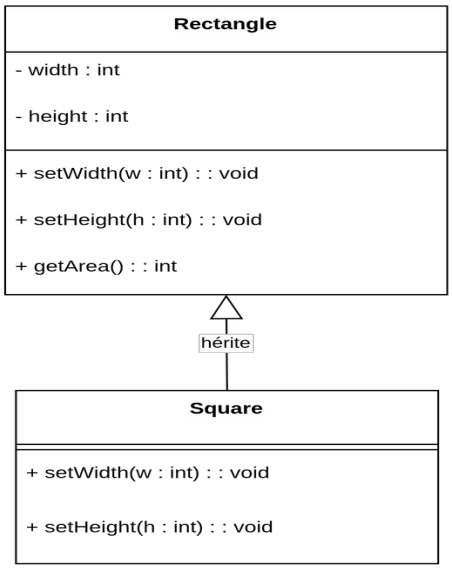
Solution : Interface `Shape` avec polymorphisme.



LSP - Liskov Substitution Principle

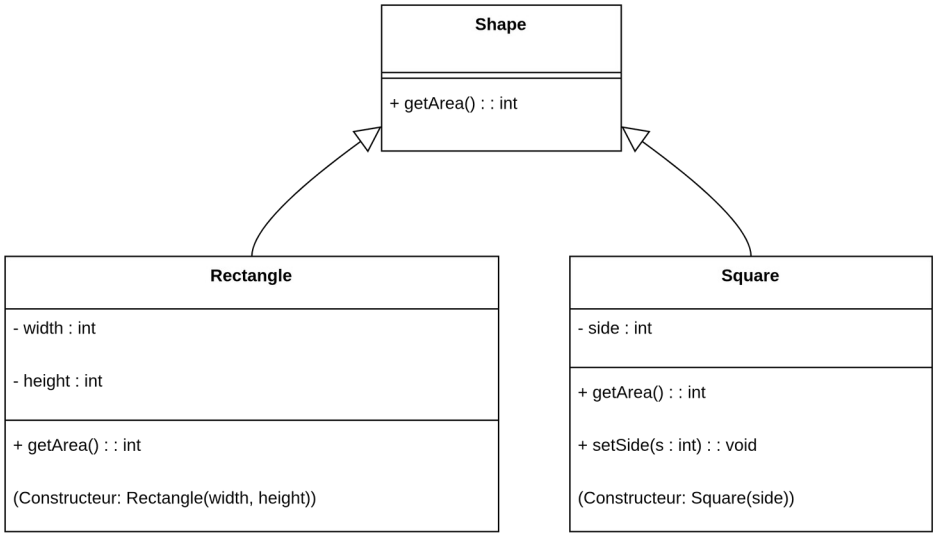
AVANT Refactoring

Problème : Square.setWidth modifie aussi la hauteur → violation du comportement attendu d'un Rectangle.



APRÈS Refactoring

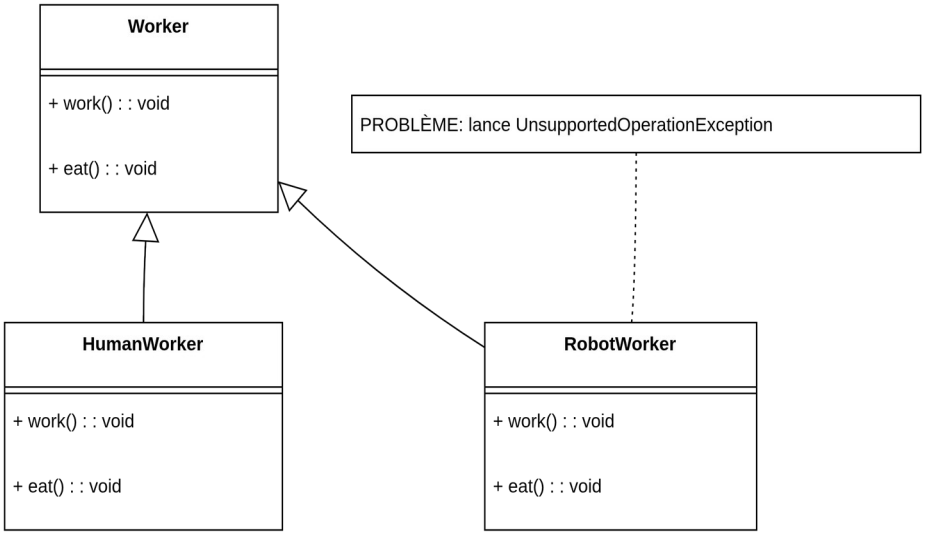
Solution : Square et Rectangle implémentent Shape mais n'héritent plus l'un de l'autre.



ISP - Interface Segregation Principle

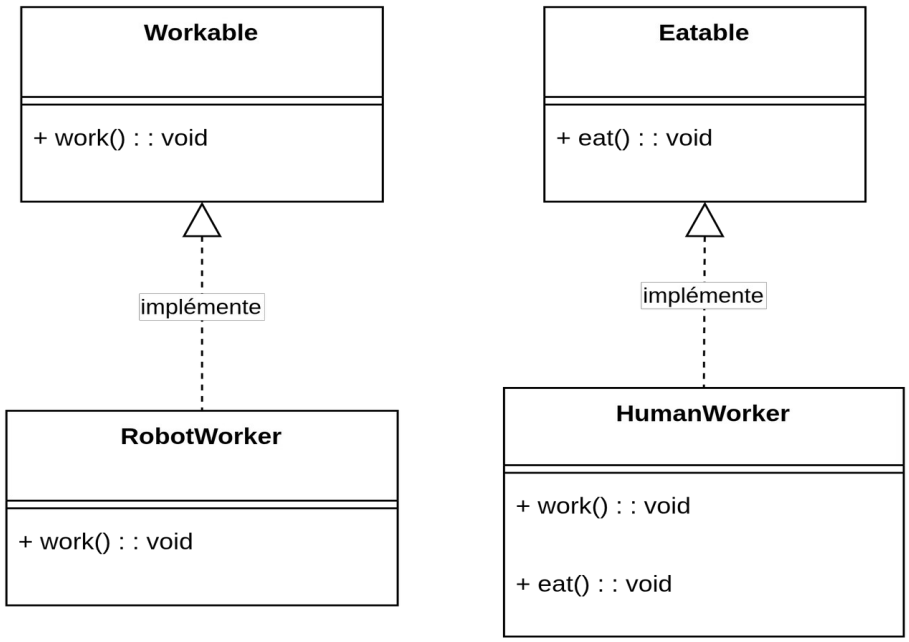
AVANT Refactoring

Problème : RobotWorker doit implémenter eat() mais ne devrait pas.



APRÈS Refactoring

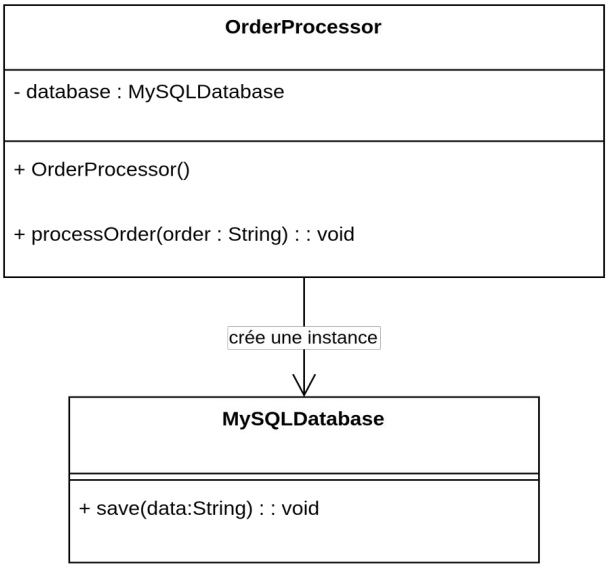
Solution : Séparation des interfaces.



DIP - Dependency Inversion Principle

AVANT Refactoring

Problème : Couplage fort avec MySQLDatabase.



APRÈS Refactoring

Solution : Injection de dépendance via l'interface Database.