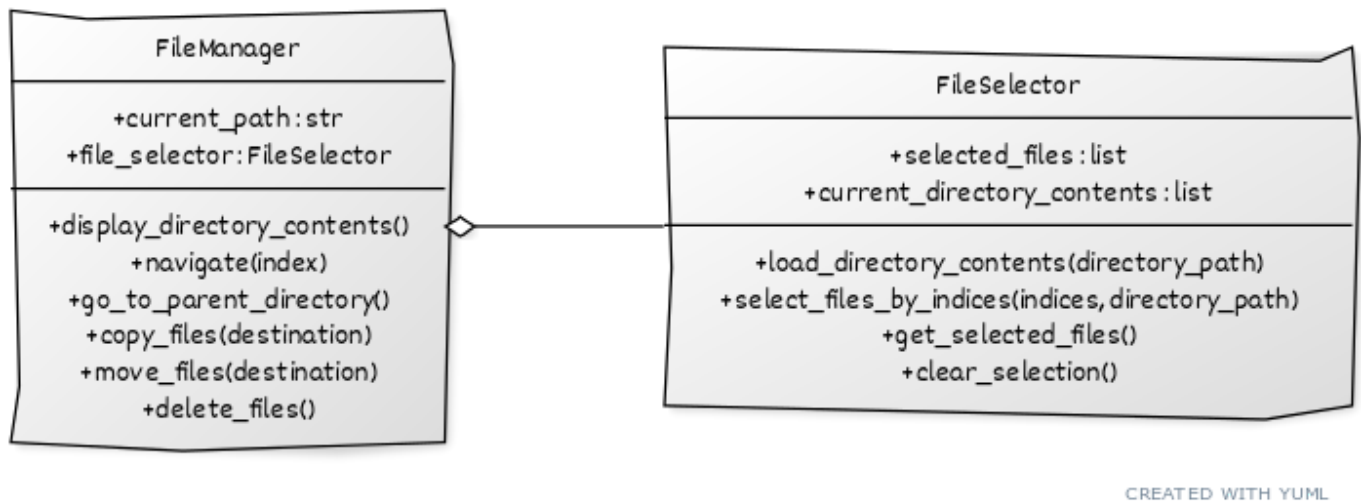


Atelier 1 - Testabilité d'une classe

Individuellement, vous rendez le projet généré fmgr.py testable.



Pour les personnes préférant un autre langage (C++, C#, Js, Java ou PHP acceptés), vous pouvez générer un programme similaire organisé avec les même classes que celui fourni.

Etape 1 - Appropriation par refacto.

La classe FileManager ne respecte pas le principe de responsabilité unique : Elle gère à la fois l'exploration et les traitements (copie, déplacement, suppression).

Sortir de la classe FileManager, les fonctions d'exploration pour les mettre dans une classe FileExplorer.

Etape 2 - Dépendances

La classe qui va subir de fortes évolutions est la classe FileManager. Elle va donc faire l'objet de tests unitaires qui garantiront :

- la qualité des fonctionnalités ajoutées,
- la non-régression des fonctionnalités existantes

Elle dépend de la classe FileExplorer, de l'interface utilisateur et de module (os, shutil).

Utiliser l'inversion de contrôle pour injecter via les paramètres du constructeur ou des méthodes ces dépendances sous la forme d'une ou plusieurs interfaces respectant les 3 derniers principes SOLID.

Remise

Lien du GitHub public du projet à sylvain.labasse@mail-formateur.net avant le 23/12/24.