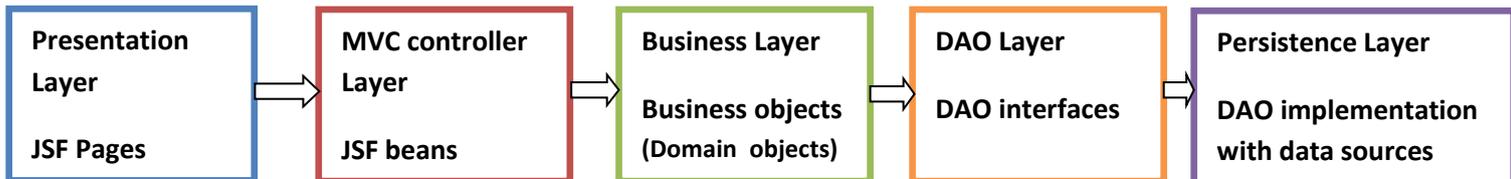


TD 7 : Simulation banque en architecture 5-tiers

1. Traitement en architecture 5-tiers

La couche « Presentation » est constituée des pages JSF. Celles-ci travaillent avec la couche « Controller » constituée des JSF beans qui s'occupe de la communication (interaction) avec la couche « Business », le noyau du système. La couche DAO (*Data Access Object*) reste générique, qui définit les interfaces permettant d'effectuer les opérations vers la couche « Persistence ». Les interfaces DAO peuvent être implémentées par différentes solutions indépendantes dans la couche « Persistence ».



2. Modifications nécessaires pour respecter l'architecture 5-tiers

A ce jour, nous avons eu différentes versions de notre programme « bank ». Notre programme évolue rapidement. Nous avons besoin de « prendre un peu de recul » et de restructurer notre programme afin de garantir toujours sa maintenabilité et son extensibilité, dans le cadre d'une architecture en 5-tiers. Voici une liste non exhaustive des parties du programme à « refactor » :

- ✓ Certains objets de domaine (business objects : BO) sont annotés et persistés directement, ce qui est contre le principe de DAO. Il faut les rendre en POJO (*Plain Old Java Object*) et gérer le travail de persistance dans la couche « Persistence ». A ce propos, les interfaces DAO (ex. *StatisticPersistence*) et leurs implémentations doivent être réorganisées.
- ✓ Vous devez réfléchir pour créer des classes persistance (à annoter) dans la couche « Persistence », afin de traiter les opérations CRUD, demandées par la couche « Business ».
- ✓ La classe « Simulation » est à présent bien complexe, vous pouvez essayer d'en extraire certains fonctionnements et les mettre dans d'autres nouvelles classes, en laissant uniquement l'algorithme de simulation.
- ✓ Regrouper toutes classes BO dans un package et des sous-packages.

Faire coupler/découpler les classes qui s'interagissent entre elles dans différentes couches, par la gestion des beans Spring IoC. Ceci permet de gérer la création des objets de ces classes de manière transparente et centralisée.

Utilisez la technique de Primefaces (<https://www.primefaces.org/showcase>) pour afficher le résultat de la simulation graphiquement (le camembert). Les autres résultats ne sont pas persistés et ne sont gérés qu'en mémoire par les classes métier.