

# Présentation générale

Le but est d'utiliser un maximum de programmation orientée objet avancée ainsi qu'un peu de programmation fonctionnelle. Le but est de pouvoir savoir quel patron (orienté objet) utiliser pour résoudre telle ou telle situation de l'activité intégrative.

## Programmation fonctionnelle

Les fonctions sont vue comme des valeurs et peuvent être passée à d'autres fonctions. La programmation fonctionnelle est plus abstraite et permet d'être plus efficace au niveau de l'écriture car on peut faire plus de choses en beaucoup moins de lignes.

## Acquis visés par le cours (UE36)

- Concevoir une solution selon les principes POO
- Programmer en exploitant les spécificités d'un langage de programmation
- Exploiter les ressources externes (base de données, etc)
- Tester en intégration (capable de fonctionner dans son ensemble)
- Tester en isolation (extraire des bouts de code et leur faire passer des tests, exemples tests unitaires)

## Activité intégrative

L'activité intégrative consiste en deux projets, un en Java et un en C# qui dialogue via une base de donnée relationnelle. Le travail est découpé en itérations. Les différents cours (COO, Java, C# et SD) permettent de s'entraîner dans les différents acquis mais c'est bien l'AI qui valide les acquis.

## Itérations

L'AI se fait en 4 itérations

Itération	Durée	Status
1	2 semaines	Formative

Itération	Durée	Status
2	2 semaines	Formative
3	3 semaines	Certificative
4	3 semaines	Certificative

## Validation de l'UE

- Si tous les acquis sont validés, réussite d'office (très rare)
- $\leq 3$  acquis à observer, réussite via défense orale
- $> 3$  acquis à observer, deuxième session (même projet + nouveautés)

## Difficulté

La difficulté n'est pas dans la difficulté du projet ou des algorithmes mais dans l'organisation et la charge de travail. Il faut donc faire très attention à bien s'organiser et ne pas prendre de retard.

---

Revision #2

Created 19 September 2023 17:59:10 by SnowCode

Updated 3 October 2023 21:07:06 by SnowCode