

Licence Informatique

Programmation et conception

Responsables	Descriptions	Informations
Basile COUETOUX basile.couetoux@univ-amu.fr	Code : S04IN5A1	Composante : Faculté des Sciences
Nicolas PRCOVIC nicolas.prcovic@univ-amu.fr	Nature : Domaines : Sciences et Technologies	Nombre de crédits :

LANGUE(S) D'ENSEIGNEMENT

Français

CONTENU

La conception consiste à donner à un projet de programmation une organisation souple mais structurante. Une telle organisation permettra de garder un code clair au cours des différentes évolutions que le code subira. Ces évolutions devront pouvoir être faites facilement, et le code pourra à tout moment être testé facilement. La modularité intelligente est une clé de voûte de la conception. Les principes de cette modularité sont énoncés dans les principes SOLID. Le savoir-faire de cette modularité est résumé dans les patrons de conceptions qui permettent d'éviter les pièges habituels de conception en suivant des modèles sous la forme de diagrammes UML.

- Documentation, spécification, automatisation des tests (20%)
- Principes SOLID : SRP, OCP, LSP, ISP, DIP (20%)
- Diagramme de classes UML (20%)
- Présentation et utilisation de patrons de conceptions sur des exemples concrets (40%)

COMPÉTENCES À ACQUÉRIR

- Produire une solution technique à partir d'une modélisation.
- Détecter un problème de conception en se référant aux principes SOLID et le corriger en utilisant des patrons de conception.
- Concevoir une solution technique en mettant en place une stratégie de tests pour garantir les spécifications.
- Utiliser les outils et techniques de gestion de projet informatique : environnement de développement, contrôle de version, automatisation de la production.
- Utiliser une bibliothèque ou un framework en consultant sa documentation.

MODALITÉS D'ORGANISATION

- Le langage de programmation pour illustrer ce cours est Java.
- Le travail des étudiants sera stocké sur GIT, ce qui permettra une bonne prise en main de cet outil de gestion de version.

BIBLIOGRAPHIE, LECTURES RECOMMANDÉES

- Clean Code: A Handbook of Agile Software Craftsmanship, Robert C. Martin
- Code Complete, Steve McConnell
- Refactoring, Martin Fowler
- Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides

PRÉREQUIS RECOMMANDÉS

- Projet : initiation génie logiciel

VOLUME HORAIRE

- Volume total: 59 heures
- Cours magistraux: 15 heures
- Travaux pratiques: 44 heures

CODES APOGÉE

- SIN5U02A [ELP]
- SIN5U02L [ELP]

- SIN5U03C [ELP]
- SIN5U03C [ELP]

M3C

Aucune donnée M3C trouvée

POUR PLUS D'INFORMATIONS

[Aller sur le site de l'offre de formation...](#)



Dernière modification le 13/12/2023