NFA019 - Projet systèmes d'information : mise en pratique avec Java

Mis à jour le 19-12-2025

Code : NFA019

Unité d'enseignement de type mixte

6 crédits

Volume horaire de référence (+/- 10%) : **50 heures**

Responsabilité nationale :
EPN05 - Informatique / Nada MIMOUNI

Contact national :

EPN05 - Informatique

292 rue saint Martin
33.1.13B
75003 Paris
01 40 27 22 64
Florian Gau
florian.gau@lecnam.net

Présentation

Prérequis

UE de niveau fin BAC+2 en Informatique, option Système d'Information.
Destiné aux auditeurs préparant la licence et/ou le diplôme RNCP Analyste-Programmeur option Système d'Information.
Il est conseillé d'avoir fait les UE NFA031, NFA032 et NFA035.

Objectifs pédagogiques

Ce module permet d'acquérir, de consolider et de mettre en pratique un ensemble de compétences permettant de réaliser un système d'information : spécification d'un cahier des charges, conception UML de l'architecture d'une solution et de ses différents modules, mise en place d'une stratégie de test, utilisation des outils de gestion des versions, intégration de patrons de conception, gestion des données et de la persistance, conception des interfaces graphiques constituant l'IHM du système développé, notions de base pour le développement d'une application Web.

A l'issue de ce module, l'étudiant sera capable de réaliser une solution logicielle complète et fonctionnelle pour répondre à un besoin donné (cahier des charges). Le langage support de ce module est le langage Java. L'environnement de développement est Eclipse. La gestion des données s'appuiera sur MySQL (XAMPP). Les tests s'appuient sur l'environnement JUnit. Et enfin, la gestion des versions se fait avec GIT.

Compétences

Analyste et développeur de projets en Systèmes d'Information avec :

- Conception de diagramme UML.
- Maitrise des relations de classes et paradigmes orientés objets (encapsulation, héritage, polymorphisme, classes abstraites, interfaces).
- Maitrise du langage Java (concepts orientés objets, exceptions, collections, etc.).
- Réalisation de solutions respectant l'architecture MVC (Modèle-Vue-Contrôleur).
- Gestion des données avec l'API JDBC, maitrise des patterns Singleton et DAO (Data Access Object).
- Conception d'interfaces graphiques avec Swing.
- Développement d'une application Web Java avec Servlets et JSP (notions de base).
- Définition de scénarios et stratégies de tests avec JUnit.
- Gestion de versions de programmes avec GIT.

Programme

Contenu

Cette formation est basée sur la réalisation d'un projet sur plusieurs étapes :

- Rappel des notions de base de la POO.
- Présentation du langage de conception UML (en particulier le diagramme de classes), utilisation de l'outil StarUML avec l'extension Java.
- Rédaction d'un cahier de charges à partir d'un ensemble de spécifications.
- Création des packages et implémentation des classes Java correspondants au cahier de charges, prise en compte de l'architecture MVC (Modèle-Vue-Contrôleur).
- Test des fonctionnalités des classes Java avec les Test unitaires JUnit.
- Export du projet sur un serveur de gestion de versions : outil GIT, plugin EGIT sur Eclipse.
- Gestion des données avec l'API JDBC (Java Data Base Connectivity) et selon les modèles Singleton et DAO (Data Access Object). Utilisation du serveur XAMPP et MySQL.
- Création d'interfaces graphiques avec Swing pour une application desktop.
- Développement d'une application Web Java avec Servlets et JSP (introduction).
- Finalisation du projet et démonstration de l'application réalisée.

Modalités de validation

- Contrôle continu
- Projet(s)

Bibliographie

Titre	Auteur(s)
Programmer en Java - (Eyrolles, 2016)	Claude Delannoy
Design Patterns: Elements of Reusable Object-Oriented Software - (Addison-Wesley Professional Computing Series, 1994)	Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
Git - Maîtrisez la gestion de vos versions (concepts, utilisation et cas pratiques) - (Epsilon, 2019)	Samuel Dauzon
JUnit in Action, Second Edition - (Manning Publications, 2010)	Petar Tahchiev, Felipe Leme, Vincent Massol, Gary Gregory