GLG105 - Génie Logiciel et Architecture Cloud Native

Mis à jour le 16-02-2026

Code : GLG105

Unité d'enseignement de type cours

6 crédits

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

Responsabilité nationale :
EPN05 - Informatique / Ghislain ROCHETEAU

Contact national :

EPN05 - Informatique

2 rue Conté
33.1.4A
75003 Paris
01 40 27 22 40
Agnès Lapierre
agnes.lapierre@lecnam.net

Présentation

Prérequis

Pour tirer le meilleur parti de ce cours il est souhaitable de connaître les connaissances de base en informatique, dispensées en L1, L2.

Une connaissance générale, acquise dans d'autres cours ou dans la pratique, de la programmation, des systèmes d'exploitation et de l'architecture sera un plus.

Objectifs pédagogiques

Cette UE est destinée à donner un panorama complet des principales techniques et méthodes de base permettant de mettre en oeuvre et de déployer des logiciels dont le principe est qu'ils fonctionnent correctement conformément aux besoins de ses utilisateurs.

Compétences

Maîtriser le cycle de vie complet d'un logiciel et appliquer les bonnes pratiques du génie logiciel pour produire, déployer et maintenir des logiciels de qualité.

Programme

Contenu

Ce cours est articulé sur la compréhension en profondeur du processus de développement logiciel de façon qu'un auditeur puisse se couler sans difficulté dans les différentes techniques et méthodes qu'il rencontrera dans sa vie professionnelle.

Il couvre l'ensemble du cycle de vie d'un produit logiciel, depuis l'expression du besoin jusqu'à son exploitation en environnement Cloud, en intégrant les dimensions architecturales, organisationnelles et opérationnelles. L'objectif est de permettre aux étudiants de maîtriser les pratiques actuelles du génie logiciel.

Cette UE est organisée selon le cycle de vie d'un produit logiciel moderne :

- De l'idée au produit (Engineering & Pilotage) : Importance et rôle du génie logiciel dans l'industrie, les cycles de vie (cascade, V, itératif, agile), la gestion de projet IT, Scrum + Agilité à l'échelle
- Concevoir pour durer (Architecture & Design) : Architecture logicielle et conception orientée qualité dans les systèmes distribués modernes. Rôle et responsabilités de l'architecte logiciel. Styles architecturaux modernes, Architecture Cloud Native.
- Cloud Native Engineering : Approche, intérêt, Cloud privée, Cloud publique, Cloud hybride, modèles de déploiement IaaS, PaaS et SaaS, gouvernance, aspects financiers, move to cloud, sécurité, FinOps (coûts).
    - Aspect technologique avec la virtualisation, l'approche conteneur avec Docker et leur orchestration avec Kubernetes ainsi que l'architecture sous-jacente, le Datacenter. Les différents notions, machine virtuelle, mise à l'échelle, Load Balancing, IoT seront présentées à travers le Cloud public Microsoft Azure.
    - Travaux pratiques sur Docker et Kubernetes pour mettre en œuvre des conteneurs et orchestrer des applications.
- Construire et industrialiser (DevOps). La gestion de configuration/version avec Git/GitHub, l'approche CI/CD intégration continue / déploiement continue, les modèles de déploiement. Concepts de sécurité intégrée.

Il sera souligné l'importance de l'architecture et des concepts architecturaux. Une attention sera portée à la prise en compte des aspects non fonctionnels du logiciel dans leur développement.
A ces modules, pour compléter le panorama, s'ajoutent un module de sensibilisation : Exploiter et maintenir (Qualité & Observabilité), concernant l'assurance qualité logicielle. Concept transverse indispensable à prendre en compte dans tous les projets informatiques.

Modalités de validation

- Examen final

Description des modalités de validation

Examen 3h.

Bibliographie

Titre	Auteur(s)
Software Engineering Standards Collection	IEEE
Le Génie logiciel (Que sais-je n°2956, PUF).	J. PRINTZ
Productivité des programmeurs, Hermès-Lavoisier	J. PRINTZ
Puissance et limites des systèmes informatisés, Hermès-Lavoisier	J. PRINTZ
Architecture logicielle, Dunod	J. PRINTZ
Estimation des projets de l'entreprise numérique	J. Printz, B. Mesdon, N. Treves
DevOps Hanbook	Gene Kim, Patrick Debois, John Willis, Jez Humble
The Phoenix project	Gene Kim, Kevin Behr, George Spafford