

Objectifs de l'ECUE en termes de compétences et d'acquis d'apprentissage visés**A l'issue de cette UE, l'apprenant aura progressé sur les compétences suivantes du référentiel de la formation :**

- BC2.1 : Modéliser un problème ou un besoin fonctionnel exprimé par un client et spécifier une solution informatique
- BC2.2 : Étudier, comparer et sélectionner les outils et méthodes nécessaires à la conception, au développement et au test d'une solution informatique
- BC2.3 : Concevoir et développer les applications informatiques : web, mobiles, logicielles
- BC2.4 : Documenter une étude et une solution informatique
- BC2.5 : Analyser et identifier les aspects réglementaires et techniques
- BC4.3 : Déployer une solution informatique
- BC4.4 : Administrer une solution informatique

Plus précisément, il sera capable de :

- Modéliser des architectures orientées services
- Comprendre la notion de dynamisme d'application et de dynamisme architectural
- Concevoir des architectures orientées services : architectures micro-services et Event Driven Architecture
- S'appuyer sur et mettre en œuvre des paradigmes de communications synchrones et asynchrones au travers des bus à messages et des web services
- Décrire et documenter une solution architecture de système d'information
- Appréhender les problématiques de déploiements et d'adaptation des applications à chaud
- S'appuyer sur et mettre en œuvre des paradigmes de communications synchrones et asynchrones au travers des bus à messages et des web services
- Appréhender les problématiques de développement ServerLess (dit FaaS) en environnement conteneurisé, notion sur le Cold Start et Warm Start
- Administrer un système d'information à base de services

Description de l'ECUE

Les principes des Architecture Orientées services, micro-services, Architecture EDA.

Création d'applications dynamiques et adaptables en Java, utilisation des composants du framework SpringBoot pour la construction d'applications robustes.
Déploiement et assemblage à chaud des applications dans des conteneurs avec la solution docker, docker Swarm, portainerIO.
Conteneurisation et instanciation à la demande, notion de services en ServerLess (FaaS).
Passage à l'échelle des applications, scalabilité et distributivité des applications via des systèmes
Notion de qualité et sécurité dans l'assemblage des applications.
Mise en œuvre dans le monde Java, Framework SpringBoot, composants cloud, découverte à la volée de services, etc...
Communication asynchrone et synchrone et files de messages au travers du bus à message Kafka et de web services REST.

Prérequis

- > Connaître le langage JAVA version 11 ou plus
- > Maîtriser un IDE pour JAVA (Netbeans ou IntelliJ ou Eclipse)
- > Connaître le fonctionnement d'un outil de build automation comme MAVEN ou GRADLE
- > Connaître le framework Java SpringBoot et quelques notions sur la création de webservice sous SpringBoot.
- > Connaître les principes des Web Services REST.
- > Quelques notions sur la sécurité des Web Services Basic Auth, JWT, OAuth, OpenId (sans être un expert du domaine).
- > Des notions sur les formats standard d'échanges, json, xml, csv (les classiques).
- > Des notions sur la virtualisation par superviseurs.
- > Des notions sur les bases de données relationnelles (Postgresql) et +/- distribuée nosql (MongoDB).
- > Des notions sur les infrastructures informatiques et réseaux en général.
- > Des notions sur les architectures distribuées.
- > Maîtriser un environnement client (Windows ou Linux ou les deux, un peu de script bash ou script shell)

Références