

Cloud Computing

Domaine	Ingénierie et Architecture
Filière	Informatique et systèmes de communication
Orientation	Réseaux et systèmes (ISCR)
Mode de formation	Plein temps

Informations générales

Nom	: Cloud Computing
Identifiant	: CLD
Années académiques	: 2021-2022, 2022-2023
Responsable	: Marcel Graf
Charge de travail	: 120 heures d'études
Périodes encadrées	: 64 (= 48 heures)

Semestre	E1	S1	S2	E2	S3	S4	E3	S5	S6
Cours						32			
Laboratoire						32			

Connaissances préalables recommandées

L'étudiant-e doit connaître et savoir utiliser les notions suivantes :

- programmation réseau.

L'unité d'enseignement API (Applications Internet) permet d'acquérir ces connaissances.

Objectifs

A l'issue de cette unité d'enseignement, l'étudiant-e sera capable de :

- évaluer si un service commercial comporte les caractéristiques essentielles du Cloud Computing ;
- analyser si une composante informatique se prête à être réalisée dans le cloud et choisir le modèle de service et le modèle de déploiement approprié ;
- concevoir un déploiement efficace d'application dans un service IaaS ;
- expliquer les avantages d'un service PaaS pour créer des services évolutifs et ses limitations ;
- expliquer les différences entre les bases de données single-tenant et les bases de données multi-tenant ;
- concevoir le déploiement d'une application à partir d'images conteneur dans le cloud ;
- concevoir un service formé à partir de plusieurs services cloud de base interconnectés avec des fonctions FaaS.

A l'issue des travaux pratiques en laboratoire, l'étudiant-e sera en outre capable de :

- mettre en œuvre une application multi-tier dans un service IaaS ;
- développer une application web sur un service PaaS ;
- utiliser une base de données multi-tenant ;
- exploiter une application sur un orchestrateur de conteneurs ;
- mettre en œuvre un service à partir de fonctions FaaS.

Contenu et formes d'enseignement

Répartition des périodes indiquée à titre informatif.

Cours: 32 périodes

- Introduction 2
- Concepts de base: Modèles de Service (IaaS/PaaS/SaaS), Modèles de Déploiement (cloud public/privé/communautaire/hybride), caractéristiques essentielles selon le NIST, forces économiques, principaux acteurs dans le marché 4
- Infrastructure-as-a-Service: instances (types, images d'instance, cycle de vie), pare-feux, stockage orienté bloc, réseaux virtuels, service de stockage de fichiers, principes des mesures d'utilisation et de la facturation 2
- Déploiement d'une application 2-tier évolutive sur IaaS: parallélisme des requêtes, réplication du serveur web, Load Balancing, gestion d'état (stateless/soft state/stateful), Sticky Sessions, images d'instance customisées 2
- Autoscaling d'une application en fonction de la charge: modèles de trafic, boucle de réconciliation, monitoring de la charge, Scaling Events, Launch Templates 2
- Déploiement de code applicatif sur Platform-as-a-Service: conteneur pour la logique d'application, services supplémentaires et leurs APIs, traitement sans état et avec état, mécanismes d'évolutivité fournis par la plateforme, services de persistance des données, modèle de facturation (quotas d'utilisation) 4
- Database-as-a-Service: Bases de données dans le cloud, modèle de données SQL/NoSQL, Tenancy (single-tenant/multi-tenant), distribution de la charge, sharding et hashage cohérent, réplication, accès concurrents et cohérence 2
- Container Orchestration (Kubernetes): architecture d'un cluster, déclaration de ressources par fichier, Pods, Services, Deployments, intégration avec un IaaS public pour Ingresses et stockage orienté bloc 4
- Automatisation IT: principes de Desired State Configuration et Infrastructure as Code, outils pour le déploiement des ressources cloud (CloudFormation, Terraform), outils pour le Configuration Management (Ansible) 4
- Serverless: Déploiement de code comme fonction (Function-as-a-Service), déclenchement par événement, facturation par milliseconde d'exécution, Auto Scaling, FaaS comme colle pour l'intégration des services cloud, problème du Cold Start 2
- Travaux écrits et discussion des corrections 4

Laboratoire: 32 périodes

- Déploiement d'une application GED 2-tier à partir de zéro sur un IaaS (Amazon Web Services) 6
- Réplication de l'application et Load Balancing pour la rendre évolutive, tests de montée en charge, Auto Scaling en fonction de la charge 6
- Déploiement du code source d'une application web depuis un IDE sur un PaaS (Google App Engine) et persistance des données évolutive dans un service de base de données NoSQL (Google Data Store) 4
- Déploiement des images conteneur d'une application 3-tiers sur Kubernetes (Minikube et Google Kubernetes Engine) 4
- Installation automatisée d'une application avec un outil de gestion de configuration (Ansible) 2
- Déploiement de code dans un service Function-as-a-Service et test de montée en charge 2
- Préparation du workshop 6
- Workshop sur des services cloud récents et/ou spécialisés 2

Bibliographie

- Thomas Erl, Zaigham Mahmood, Ricardo Puttini, Cloud Computing: Concepts, Technology & Architecture, 2013-05-16, Prentice Hall.
- Bill Wilder, Cloud Architecture Patterns, 2012-09, O'Reilly Media.
- Jurg van Vliet, Flavia Paganelli, Programming Amazon EC2, 2011-02, O'Reilly Media.
- Lucas Carlson, Programming for PaaS, 2013-07, O'Reilly Media.
- Mathieu Zarouk, Cloud Computing - Maîtrisez la plate-forme AWS - Amazon Web Services, 2013-01-09, Eni.

Contrôle de connaissances

Cours : l'acquisition des matières de cet enseignement sera contrôlée au fur et à mesure par des tests et des travaux personnels tout au long de son déroulement. Il y aura au moins 3 tests d'une durée totale d'au moins 2 périodes.

Laboratoire : ils seront évalués sur la base des rapports de manipulation, à 3 reprises au minimum.

Calcul de la note finale

Note finale = moyenne cours x 0.67 + moyenne laboratoire x 0.33