USSI8Y - Informatique S3

Présentation

Programme

Contenu

Positionnement : Semestre 3 **UE : Informatique** ECTS : 8

Nombre d'heures : 168h

Modalité : Présentiel (100%)

L'UE Informatique est constituée de 4 éléments constitutifs d'UE :

- Cloud Computing et Technologies Distribuées
- Electronique des objets
- Java avancé : Conception Avancée et Architecture Logicielle
- Langage Python – S3
- Projet

La note finale associée à cette UE est calculée en pondérant les notes des 5 ECUE avec les coefficients associés.

Mis à jour le 18-05-2026

Code : USSI8Y

Unité spécifique de type cours

8 crédits

Responsabilité nationale : EPN06 - Mathématique et statistique / Giorgio RUSSOLILLO

ECUE 1 : Cloud Computing et Technologies Distribuées (Coef : 2) - 56h (Cours, TD, TP) - 15h (Estimation temps de travail personnel)

Objectifs :

- L'objectif de ce cours et ses applications est d'acquérir les connaissances générales sur les systèmes distribués et les intergiciels pour l'urbanisation des systèmes d'information distribués, pour la conception d'applications distribuées large échelle.
- Compréhension des mécanismes sous-jacents du Cloud Computing.
- Être capable de différencier les avantages et inconvénients des systèmes distribués vs les systèmes « centralisés ».

Contenu :

1. Systèmes distribués : principes théoriques de fonctionnement
    a. Rappels sur les processus dans les systèmes centralisés : mécanismes de synchronisation, exclusion mutuelle, multi- threads, ordonnancement des processus, mémoire.
    b. Exigences des systèmes distribués et principes généraux.
    c. Gestion du temps : horloges logiques, ordre causal.
    d. Algorithmes utilisés dans les systèmes distribués : exclusion, synchronisation, élection, terminaison, coupure, état global, … (Quelques exemples)
    e. Gestion partagée des données : désignation/ transparence, cohérence mémoire, localisation, P2P, …
2. Architectures des systèmes distribués
    a. Infrastructures : data centers, multi-coeurs, clustering, architectures mémoires NUMA, UMA, …
    b. Méthodes de virtualisation et de containerisation.
    c. Exemple de l'écosystème Hadoop et de HDFS
3. Tolérance aux pannes dans les systèmes distribués :
    a. Définitions : pannes, erreurs, , sureté de fonctionnement, types de pannes, pannes byzantines, mécanismes de réplication : redondances matérielles, mécanismes de recouvrement, …
    b. Introductions aux bases NoSQL et à MongoDB
    c. Etude de cas et exemple de tolérance aux pannes avec MongoDB et Hadoop-HDFS
4. Architectures de type Cloud computing
    a. Notions de PaaS, SaaS …
    b. Exemples Google, Microsoft, AWS.
5. Traitements parallèles de masse - Map -reduce
    a. Paradigme Map reduce
    b. Exemple de programmation map reduce avec MongoDB et Hadoop HDFS -YARN
    c. Evolution du modèle Map reduce – Exemple de Apache Spark

Compétences visées :

A l'issue du cours, l'apprenti est capable de :

1. utiliser Map reduce sur MongoBD,
2. gérer la tolérance aux pannes dans les systèmes distribués,
3. passer à l'échelle sur le Cloud (AWS).

Modalités d'évaluation : Contrôle continu (rendus de TP)

ECUE 2 : Electronique des objets (Coef : 1) - 30h (Cours, TP, Projet) - 10h (Estimation temps de travail personnel)

Objectifs : Apprendre à utiliser les éléments de base du langage Arduino.

Contenu :

1. Introduction générale
    1. Présentation du matériel, des composants et vue d'ensemble des cartes
    2. Schéma d'une platine Arduino et exploration des différentes broches
    3. Explication de la platine d'expérimentation
    4. Installation de l'environnement de programmation IDE et des pilotes
2. Introduction à la programmation Arduino
    1. Présentation de l'environnement de la programmation IDE
    2. Structure d'un programme : entête déclarative, setup() et loop()
    3. Les différents types de variables et leur impact sur la mémoire
    4. Incrémentation d'une variable et conditions de sorties de boucles
3. Les fonctions du langage Arduino
    1. Utilisation des sorties analogiques (PWM)
    2. Pilotage des entrées et sorties numériques (protection par résistances)
    3. Les entrées analogiques
    4. Les fonctions de temps et de conversion (Mappage)
    5. Le potentiomètre
4. Communication entre des capteurs, des actionneurs et différents afficheurs
    1. Les protocoles de communication UART et I2C
    2. Contrôle et pilotage de servos moteurs
    3. Acquisition de données provenant de différents capteurs (température – humidité), visualisation par led, afficheur LCD
    4. Envoi et réception des données
5. Objets connectés via ESP8266 et visualisation sur terminal mobile
    1. Description de la carte électronique ESP8266
    2. Création d'objets connectés (IOT) et visualisation sur terminal mobile
6. Concevoir et réaliser des projets créatifs (Station météo, Oscilloscope, …)
7. Animer et construire des robots à base d'Arduino (mBot, Zumo, OTTO, …)

Compétences visées :

A l'issue de la formation, l'apprenant sera capable de :

- Connaitre les bases de la programmation Arduino
- Mettre en œuvre des capteurs et des actionneurs
- Réaliser un projet Arduino
- Acquérir des données et les enregistrer en temps réel sur un tableur
- Afficher ces données sur une tablette ou un smartphone

Modalités d'évaluation : Contrôle continu sous forme de TP

ECUE 3 : Java avancé : Conception Avancée et Architecture Logicielle (Coef : 1) - 30h (Cours, TP, Projet) - 10h (Estimation temps de travail personnel)

Objectifs : Ce cours est la suite de l'ECUE Informatique générale – Programmation algorithmique de première année. Ses objectifs sont de :

- Apprendre à mettre en place un projet JAVA complexe avec la gestion de dépendances correspondante.
- Maîtriser les concepts avancés de la programmation et conception objet en JAVA.

Contenu :

1. Les bases
2. Les entrées/sorties
3. Gestion du cycle de vie des applications
4. Tests unitaires
5. Réflexion
6. Concurrence
7. Modélisation OO

Compétences visées :

- Maîtriser le langage Java et les concepts objets avancés cités dans le programme.
- Pouvoir analyser et faire évoluer un projet JAVA existant.

Modalités d'évaluation : Contrôle continu.

ECUE 4 : Langage Python – S3 (Coef : 1) - 21h (TP) - 10h (Estimation temps de travail personnel)

Objectifs :

- Présentation des besoins d'intelligence artificielle dans le monde privé.
- Gérer un projet en Python sur un sujet d'IA.

Contenu :

1. Proposition de projet : un projet doit traiter au moins une des thématiques suivantes
    1. Machine Learning
    2. Deep Learning N
    3. LP (Natural Language Processing)
    4. Computer Vision
2. Identification des objectifs et problématiques correspondantes
3. Rédaction d'un plan de travail pendant 5 séances
    1. Préparation de dataset bruts
    2. Preprocessing de dataset
    3. Développement des méthodes d'IA
    4. Expérimentation

Compétences visées : Maitriser le langage Python et ses utilisations dans l'IA

Modalités d'évaluation : Projet : travail en groupe. Soutenance + démonstration (dernière séance). Notation du code et du rapport.

Modalités de validation

- Contrôle continu
- Projet(s)