REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

OFFRE DE FORMATION
INGENIEUR D'ETAT

Intégration du module Lean Startup

Domaine	Filière	Spécialité
Mathématiques et Informatique	Informatique	Systèmes d'information

Le 16/09/2026

رئيس اللجنة البيداغوجية الوطنية
لميدان الرياضيات والإعـــلام الآلـي
أ.د بابا حنيني محمد شوقي

Année universitaire : 2026/2027

الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

عرض تكوين مسار مهندس دولة

إضافة مقياس المؤسسات الناشئة اللينة

الميدان	الشعبة	التخصص
رياضيات و إعلام آلي	الإعلام الآلي	نظم المعلومات

Le 16/09/2026

رئيس اللجنة البيداغوجية الوطنية
لميدان الرياضيات والإعلام الآلي
أ.د بابا حنيني محمد شوقي

السنة الجامعية: 2027/2026

II- Fiche d’organisation semestrielle des enseignements

Semestre 5 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14sem	C	TD	TP	Autre			Continu	Examen
UE fondamentales									
UEF1(O/P) : SI5.1	105H	3H	3H	1H30		6	8		
Bases de données avancées	63H	1H30	1H30	1H30		3	4	40%	60%
Génie logiciel	42H	1H30	1H30			3	4	40%	60%
UEF2(O/P) : SI5.2	105H	3H	3H	1H30		6	8		
Systèmes d'information avancés 1	42H	1H30	1H30			3	4	40%	60%
Compilation	63H	1H30	1H30	1H30		3	4	40%	60%
UE Méthodologique									
UEM(O/P) : SI5.3	105H	3H	1H30	3H		4	8		
Réseaux et protocoles	42H	1H30		1H30		2	4	40%	60%
Représentation des connaissances et raisonnement	63H	1H30	1H30	1H30		2	4	40%	60%
UE Transversale									
UET(O/P): SI5.4	42H	1H30	1H30			2	3		
Recherche opérationnelle	42H	1H30	1H30			2	3	40%	60%
UE Découverte									
UED(O/P): SI5.5	42H	1H30		1H30		2	3		
Recherche d'information	42H	1H30		1H30		2	3	40%	60%
Total Semestre 5	399H	12H	9H	7H30		20	30		

Semestre 6

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14sem	C	TD	TP	Autre			Continu	Examen
UE fondamentales									
UEF1(O/P) : SI6.1	105H	3H	1H30	3H		6	8		
Bases de données NoSQL	42H	1H30		1H30		3	4	40%	60%
Conception de logiciels	63H	1H30	1H30	1H30		3	4	40%	60%
UEF2(O/P) : SI6.2	126H	3H	3H	3H		6	8		
Systèmes d'information avancés 2	63H	1H30	1H30	1H30		3	4	40%	60%
La transformation digitale	63H	1H30	1H30	1H30		3	4	40%	60%
UE Méthodologique									
UEM(O/P) : SI6.3	84H	3H		3H		4	8		
Réseaux avancés	42H	1H30		1H30		2	4	40%	60%
Web sémantique	42H	1H30		1H30		2	4	40%	60%
UE Transversale							3		
UET(O/P): SI6.4	42H	1H30	1H30			2	3		
Analyse de Données	42H	1H30	1H30			2	3	40%	60%
UE Découverte									
UED(O/P): SI6.5	42H	1H30		1H30		2	3		
Systèmes de recommandation	42H	1H30		1H30		2	3	40%	60%
Total Semestre 6	399H	12H	6H	10H30		20	30		

Semestre 7

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14sem	C	TD	TP	Autre			Continu	Examen
UE fondamentales									
UEF1(O/P) : SI7.1	**126H**	**3H**		**6H**		**6**	**9**		
Entreprise Resource Planning	63H	1H30		3H		3	4	40%	60%
Développement Web	63H	1H30		3H		3	5	40%	60%
UEF2(O/P) : SI7.2	**105H**	**3H**	**3H**	**1H30**		**6**	**9**		
Ingénierie des exigences	42H	1H30	1H30			3	4	40%	60%
Interopérabilité des systèmes d'information	63H	1H30	1H30	1H30		3	5	40%	60%
UE Méthodologique									
UEM(O/P) : SI7.3	**126H**	**3H**	**3H**	**3H**		**6**	**8**		
Sécurité informatique	63H	1H30	1H30	1H30		3	4	40%	60%
Fondements de l'Intelligence Artificielle	63H	1H30	1H30	1H30		3	4	40%	60%
UE Transversale									
UE Découverte									
UED (O/P): SI7.4	**42H**	**1H30**	**1H30**			**2**	**4**		
Ingénierie des systèmes	42H	1H30	1H30			2	4	40%	60%
Total Semestre 7	**399H**	**10H30**	**7H30**	**10H30**		**20**	**30**		

Semestre 8

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14sem	C	TD	TP	Autre			Continu	Examen
UE fondamentale									
UEF 8.1	**105H**	**3H**	**1H30**	**3H**		**6**	**9**		
Architecture et Gestion des Systèmes d'Information Avancés	**63H**	**1H30**	**1H30**	**1H30**		**3**	**5**		
Développement mobile	**42H**	**1H30**		**1H30**		**3**	**4**	**40%**	**60%**
UEF 8.2	**105H**	**3H**	**1H30**	**3H**		**6**	**9**		
Architectures logicielles	**63H**	**1H30**	**1H30**	**1H30**		**3**	**5**	**40%**	**60%**
Big data analytics	**42H**	**1H30**		**1H30**		**3**	**4**	**40%**	**60%**
UE Méthodologique									
UEM 8.1	**105H**	**3H**	**1H30**	**3H**		**5**	**9**		
Sécurité des Systèmes d'information	**63H**	**1H30**	**1H30**	**1H30**		**3**	**5**	**40%**	**60%**
Machine and Deep Learning	**42H**	**1H30**		**1H30**		**2**	**4**	**40%**	**60%**
UE Transversale									
UET 8.1	**42H**	**1H30**		**1H30**		**2**	**2**		
Lean Startup	**42H**	**1H30**		**1H30**		**2**	**2**	**80%**	**20%**
UE Découverte									
UED 8.1	**63H**	**1H30**	**1H30**	**1H30**		**1**	**1**		
Fondements de la Data Science	**63H**	**1H30**	**1H30**	**1H30**		**1**	**1**	**40%**	**60%**
Total Semestre 8	**420H**	**12H**	**6H**	**12H**		**20**	**30**		

Semestre 9

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14sem	C	TD	TP	Autre			Continu	Examen
UE fondamentales									
UEF1(O/P) : SI9.1	**84H**	**6H**		**3H**		**6**	**10**		
Interfaces Homme Machine (IHM)	63H	3H		1H30		3	5	40%	60%
Architectures et technologies blockchain	63H	3H		1H30		3	5	40%	60%
UEF2(O/P) : SI9.2	**126H**	**3H**	**3H**	**3H**		**6**	**10**		
DevOps et Cloud Computing	63H	1H30	1H30	1H30		3	5	40%	60%
Internet of Things (IoT) : Concepts et développement	63H	1H30	1H30	1H30		3	5	40%	60%
UE Méthodologique									
UEM(O/P) : SI9.3	**105H**	**3H**		**3H**		**6**	**8**		
Business Intelligence (BI)	42H	1H30		1H30		3	4	40%	60%
Gestion de projets informatiques	63H	1H30		1H30		3	4	40%	60%
UE Transversale									
UE Découverte									
UET(O/P) : SI9.4	**21H**	**1H30**				**2**	**2**		
Aspect Juridique pour les technologies de l'information	21H	1H30				2	2		100%
Total Semestre 9	**357H**	**13H30**	**3H**	**9H**		**20**	**30**		

Semestre 10 :

Domaine : Mathématiques et Informatique

Filière : Informatique

Spécialité : Systèmes d'information

Stage en entreprise sanctionné par un mémoire et une soutenance en présentiel ou à distance.

	VHS	Coeff	Crédits
Projet de fin d'étude	420H	20	30
Stage en entreprise	/	/	/
Séminaires	/	/	/
Autre (préciser)	/	/	/
Total Semestre 10	420H	20	30

7- Récapitulatif global de la formation :(indiquer le VH global séparé en cours, TD, **pour les 6 semestres d'enseignement**, pour les différents types d'UE)

UE / VH	UEF	UEM	UET	UED	Total
Cours	33H	15H	3H	7H30	58H30
TD	19H30	7H30	3H	3H	33H
TP	25H30	15H	3H	4H30	48H
Travail personnel					
Autre (PFE S10)	30H				30H
Total	108H	37H30	9H	15H	**169H30**
Crédits	118	38	10	14	**180**
% en crédits pour chaque UE	64,44%	21,11%	6,67%	7,78%	**100%**

III - Programme détaillé par matière

Semestre : S5
Unité d'enseignement : UEF1
Matière : Bases de données avancées
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

Ce cours fournit aux étudiants des connaissances théoriques et des compétences pratiques sur des sujets avancés dans les systèmes de bases de données et le big data. L'objectif principal de ce cours est d'élargir la vision des étudiants et d'introduire des sujets avancés, notamment les extensions orientées objet, relationnel-objet, les bases de données déductives et les langages d'interrogation appropriés ainsi que les bases de données distribuées. Les sujets supplémentaires couverts dans ce cours aideront les étudiants à devenir plus compétents dans le choix des technologies appropriées pour les projets et élargiront leur base de connaissances afin qu'ils aient une meilleure compréhension du domaine.

A la fin de ce cours les étudiants devront avoir une solide connaissance en :

Volet 1 : Modélisation des bases de données

- Modélisation de données orientée Objet.
- Modélisation des bases de données relationnelles-Objet.
- Modélisation des bases de données déductives et autres types de bases de données

Volet 2 : Mise en œuvre de bases de données

- Utilisation de SGBD classique dans un environnement distribué (concept de SGBD distribué, transaction distribuée, requête distribuée).

Volet 3 : Travail collaboratif sous forme d'ateliers

Connaissances préalables recommandées :

BDD : Modélisation et exploitation (S5) et BDD : Administration et Architecture (S6).

Contenu de la matière :

Chapitre I : Introduction aux SGBD orientés Objet

1. Evolution des SGBD
2. Limite des SGBDR
3. Généralités - Historique
4. Concepts fondamentaux des modèles objets
   4.1 Objet, Type et Classe
   4.2 Association
   4.3 Autres concepts de l'approche OO
   Agrégation, composition, héritage, liaison dynamique
5. Modélisation Objet avec UML:Unified Modeling Language
   5.1 Diagrammes de classes
   5.2 Diagrammes d'Etat/Transition
6. Caractéristiques des SGBD orientés objet
   6.1 Manifeste du système de BD OO
   6.2 Approches de développement de SGBDOO
   6.3 Avantages et inconvénients des SGBDOO

Chapitre II : Le Relationnel Objet : le Langage SQL3

L'apport des modèles orientés objet

1. Modèle Relationnel-Objet-SQL3
   1.1 Types abstraits de données et tables objets
   1.2 Identification d'objet OID
   1.3 Méthodes
   1.4 Héritage
   1.5 Large Object-LOB
   1.6 Vues Objet
   1.7 Association en SQL3

Chapitre III : Généralités sur les bases de données distribuées

Introduction

1. Les bases de données distribuées (réparties)
   1.1 Définitions
   1.2 Caractéristiques et avantages de la répartition
2. Le Système de Gestion de BD distribué : SGBDD
3. Construction d'une base de données distribuée : les techniques de répartition de données
   3.1 Fragmentation (horizontale, verticale, mixte)
4. Architecture et fonctions d'un SGBDD
5. Transparences dans un SGBDD : 12 règles de « DATE »
6. Classification des approches de conception d'une BD distribuée : systèmes multi bases et systèmes fédérés

Chapitre VI : Gestion de Requêtes distribuées

1. Requêtes distribuées
2. Optimisation de requêtes
   2.1 Optimisation globale
   2.2 Optimisation locale
3. Stratégies d'évaluation de requêtes
   3.1 Algorithme d'optimisation R* (System R)
   3.2 Stratégies de jointure

Chapitre VII : Gestion de Transactions distribuées

1. Gestion de transactions : Définitions
2. Gestion de la concurrence : sérialisation distribuée, protocoles de verrouillage et d'estampillage
3. Validation et reprise
   3.1 Validation à deux phases 2PC,
   3.2 Validation à trois phases 3PC

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- C. Delobel et M. Adiba, Bases de données et systèmes relationnels, Dunod 1982.
- Georges Gardarin : Les BD systèmes et leurs langages Eyrolles 2003
- G. Gardarin et P. Valduriez : Bases de données relationnelles: analyse et comparaison des systèmes Eyrolles, 1985
- C. J. Date 6ième édition traduit par Frédéric Cuppens : Introduction aux bases de Données. Internationale Thomson Publishing, 1998.
- T. Connoly et CorolynBegg. Systèmes de bases de données : approche pratique de conception de l'implémentation et de l'administration. Eyrolles 2005

Semestre : S5
Unité d'enseignement : UEF1
Matière : Génie logiciel (GL)
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

- ✓ Acquérir les concepts de base du génie logiciel en termes d'objectifs de qualité, de coût et de délai.
- ✓ Maitriser les étapes d'un cycle de vie de logiciels
- ✓ Comprendre le processus de spécification et d'analyse des besoins en utilisant les diagrammes de cas d'utilisation UML
- ✓ Maîtriser les aspects de conception de logiciels
- ✓ Maîtriser la conception d'un diagramme de classes et de séquences UML.
- ✓ Acquérir les bases de la conception de qualité
- ✓ Exploiter les patrons de conception
- ✓ Maîtrise les techniques de test dans le but de détecter la présence d'erreurs dans un programme vis-à-vis de sa spécification.

Connaissances préalables recommandées :

- ✓ *Maîtrise des concepts algorithmiques et de la programmation*
- ✓ *Notions de base en Programmation Orientée Objet.*

Contenu de la matière :

- ✓ **Chapitre 1 : Problématique et définitions (4 séances)**
  - Objectifs du génie logiciel
  - Définition d'un produit logiciel
  - Qualités d'un logiciel
  - Notion de cycle de vie d'un logiciel
  - Les étapes d'un cycle de vie de logiciels
  - Les différents modèles de cycle de vie (cascade, V, prototypage, spirale, incréments)

- ✓ **Chapitre 2 : Analyse & spécification des besoins (3 séances)**
  - Structure d'un cahier des charges
  - Modèles conceptuels
  - Besoins fonctionnels et non fonctionnels
  - Validation des besoins
  - Introduction à UML : Modélisation unifiée des systèmes
  - Définition des besoins à l'aide du diagramme de cas d'utilisation

- ✓ **Chapitre 3 : Conception des Logiciels (3 séances)**
  - Notions de base et principes de la conception de logiciels
    - conception architecturale
    - conception détaillée
  - Qualité de la conception et mesures : Couplage & Cohésion, Documentation
  - Stratégies et méthodes de conception :
    - Conception Orientée Fonctions
    - Conception Orientée Données
    - Conception Orientée Objets

- ✓ **Chapitre 4 : Modélisation orientée objet à l'aide d'UML / Aspect Statique (2 séances)**
  - Diagrammes de classes et d'objets.

- ✓ **Chapitre 5 : Modélisation orientée objet à l'aide d'UML / Aspect Dynamique (5 séances)**
    - Les diagrammes d'interactions : de séquences et communication
    - Les diagrammes d'états-transitions et d'activités.
- ✓ **Chapitre 6 : Les patrons de conception (3 séances)**
    - Introduction aux patrons de conception
    - Exemples : Singleton, Composition, Observateur …
- ✓ **Chapitre 7 : Tests des Logiciels (4 séances)**
    - Introduction (origine et problématique
    - Vérification statique (revue et inspection)
    - Techniques de test
        - Test dynamique (structurel et fonctionnel)
        - Test statistique, de surcharge, de robustesse
        - Test Mutationnel (évaluer jeu de test)
    - Automatisation des tests
- ✓ **Chapitre 8 : Concepts avancés (2 séances)**
    - Conception Orientée Composants : Diagrammes de composants …
    - Introduction aux méthodes Agiles (Méthodes XP, Méthodes SCRUM
    - DevOps
    - Référentiels de GL …

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- ✓ Ian Sommerville. Software Engineering, Pearson Edition, 2015.
- ✓ Roger S. Pressman, Bruce Maxim. Software Engineering: A Practitioner's Approach 8th Edition, McGraw Hill, 2014.
- ✓ Stephens, Rod. Beginning software engineering, Wrox, a Wiley Brand, 2015.
- ✓ Systems Analysis and Design: An Object-Oriented Approach with UML 6th Edition, Alan Dennis, Barbara Wixom, David Tegarden, Wiley, 2020.
- ✓ Kim, G., Humble, J., Debois, P., Willis, J., & Forsgren, N. The DevOps handbook: How to create world-class agility, reliability, & security in technology organizations. IT Revolution. 2021
- ✓ UML 2 : Modélisation des objets. Laurent Debrauwer et Fien Van der Heyde. Editeur ENI, Janvier 2013
- ✓ UML 2 par la pratique : Etude de cas et exercices corrigés. Pascal Roques. Editeur Eyrolles, Octobre 2011.
- ✓ Test logiciel en pratique Vuibert informatique John Warkins 2002.
- ✓ Object-Oriented Software Engineering: Using UML, Patterns and Java. Bernd Bruegge and Allen. H. Dutoit. Pearson International Edition, Prentice Hall, Second Edition, 2004.
- ✓ Architecture logicielle : Concevoir des applications simples, sûres et adaptables. Jacques Printz, Editeur Dunod, Juin 2012.
- ✓ Design Patterns pour Java : Mise en œuvre des modèles de conception en Java - Exercices et corrigés. Naouel Karam et Laurent Debrauwer, Editeur Eni, Novembre 2010.

Semestre : S5
Unitéd'enseignement : UEF2
Matière : Systèmes d'information avancés I
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
L'objectif de ce cours est de définir les concepts et les techniques avancés et de présenter les différentes stratégies de développement dans le domaine des systèmes d'information. Il permet aux étudiants de se familiariser avec les techniques et les méthodes modernes.

Connaissances préalables recommandées :

Une maîtrise des concepts de bases sur les systèmes d'informations.

Contenu de la matière

1. Introduction
2. Panorama des méthodes de conception des SI
3. Systèmes d'information distribués et répartis
4. Systèmes d'information géographiques

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :

1. Reix R., Fallery B., Kalika M., Rowe F., Systèmes d'information et management des organisations, Vuibert, 6ème édition, 2011.
2. Annick Fron, Architectures réparties en Java : RMI, CORBA, JMS, sockets, SOAP, services Web, Paris, Dunod, 2007, 208 p.
3. J. Denegre& F. Salge, '' Les systèmes d'information géographique'', PUF QSJN° 3122, 1996
4. H. Pornon, ''Systèmes d'information géographique, pouvoir et organisations géomatiques et stratégies d'acteurs'', Harmattan, 1998

Semestre : S5
Unité d'enseignement : UEF1
Matière : Compilation
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

Connaissances préalables recommandées :
Théorie des langages, Architecture des ordinateurs, Algorithmique, Système d'exploitation, Logique Mathématique.

Contenu de la matière :

Introduction générale
1- Structure d'un compilateur
- 1.1 Phases d'analyse
  - 1-1-1 Analyse lexicale
  - 1.1.2 Analyse Syntaxique
- 1-1-3 Analyse sémantique
- 1.2 Phases de production
  - 1.2.1 Génération de code
  - 1.2.2 Optimisation de code
- 1.3 Phases parallèles
  - 1.3.1 Gestion de la table de symboles
  - 1.3.2 Gestion des erreurs

2- Compilateurs, pré-compilateurs, traducteur et interpréteur
3- Rappels sur la théorie des langages
- 3.1 Les expressions régulières
- 3.2 Les automates
- 3.3 Equivalence entre expressions régulières et automates
- 3.4 Les grammaires

Chapitre I : Analyse lexicale (10%)
1- Introduction
2- Mise en œuvre d'un analyseur lexical
   - 2.1 Spécification des entités lexicales
   - 2.2 Correspondance entre expressions régulières et automates
   - 2.3 Algorithme d'analyse
   - 2.4 Création et gestion de la table des symboles
3- Gestion des erreurs lexicales

Chapitre II : Analyse syntaxique (45%)
1- Introduction
2- Les concepts de base
   - 2-1 Les formes normales des Grammaires
   - 2-2 Grammaire ε-libres et sans cycles
   - 2-3 Elimination de la récursivité à gauche directe et indirecte dans une grammaire
   - 2-4 Factorisation d'une Grammaire
3- Les méthodes d'analyse descendantes
   - 3.1 Les méthodes d'analyse non déterministes
   - 3.2 Les méthodes d'analyse déterministes
     - 3.2.1 La méthode d'analyse LL(1)
       - 3.2.1.1 Calcul des débuts et des suivants
       - 3.2.1.2 Construction la table d'analyse LL(1)

3.2.1.3 Algorithme d'analyse LL(1)
3.2.2 La Descente récursive
4- Les méthodes d'analyse Ascendantes déterministes
4.1 Analyse LR(1) par les contextes et les items
4.1.1 Construction de l'ensemble des contextes et des items LR(1)
4.1.2 Construction de la table d'analyse LR(1)
4.1.3 Algorithme d'analyse LR(1)
4.1.4 Analyse LR(K)
4.2 Analyse Simple LR(1) par les contextes et les items
4.2.1 Construction de l'ensemble des items LR(0)
4.2.2 Construction de la table d'analyse SLR(1)
4.2.3 Algorithme d'analyse SLR(1)
4.3 Analyse LALR(1) par les contextes et les items
4.3.1 Construction de l'ensemble des contextes et des items LR(1)
4.3.2 Construction de la table d'analyse LALR(1)
4.3.3 Algorithme d'analyse LALR(1)
5- Mise à jour de la table des symboles
6- Gestion des conflits dans le cas des tables d'analyse multi-définies
7- Relations entre les méthodes de l'analyse syntaxique
8- Gestion des erreurs syntaxiques
8-1 mode panique
8-2 Récupération au niveau syntagme

Chapitre III : Les formes intermédiaires (13%)

1- Introduction
2- La forme post-fixée
2.1 Evaluation des expressions arithmétiques et logiques
2.2 L'instruction d'affectation
2.3 Le branchement inconditionnel
2.4 Le branchement conditionnel
2.5 Les Instructions répétitives
2.6 Déclaration d'un tableau multidimensionnel
2.7 Référence à un élément d'un tableau multidimensionnel
3- Les quadruplets
3.1 L'instruction d'affectation
3.2 Le branchement inconditionnel
3.3 L'instruction conditionnelle
3.4 Les Instructions répétitives
3.5 Déclaration d'un tableau multidimensionnel
3.6 Référence à un élément d'un tableau multidimensionnel
4- Les triplets directs et indirects
4.1 L'instruction d'affectation
4.2 Le branchement inconditionnel
4.3 L'instruction conditionnelle
4.4 Les instructions répétitives
4.5 Déclaration d'un tableau multidimensionnel
4.6 Référence à un élément d'un tableau multidimensionnel
5- Les Arbres abstraits
4.1 L'instruction d'affectation
4.2 Le branchement inconditionnel
4.3 L'instruction conditionnelle
4.4 Les Instructions répétitives
4.5 Bloc d'instructions BEGIN-END

4.6 Déclaration d'un tableau multidimensionnel
4.7 Référence à un élément d'un tableau multidimensionnel

Chapitre IV : Traduction dirigée par la syntaxe (30%)

1- Schéma de traduction dirigée par la syntaxe dans le cas de l'analyse descendante
   1.1 Les expressions arithmétiques
   1.2 Les expressions logiques
   1.3 Les structures de contrôle
      1.3.1 Instruction conditionnelle
      1.3.2 Instructions répétitives
2- Schéma de traduction dirigée par la syntaxe dans le cas de l'analyse ascendante
   2.1 Découpage de la grammaire
   2.2 Les expressions arithmétiques
   2.3 Les instructions répétitives
   2.4 Instruction du branchement inconditionnel

Travaux Dirigés :

Les travaux dirigés sont composés de 5 séries d'exercices permettant de couvrir l'ensemble du programme vu en Cours.

Série 1 :Grammaires/Automateset Analyse lexicale (1séance)
Série 2 : Analyse syntaxique descendante (3 séances)
Série 3 : Analyse syntaxique ascendante (3 séances)
Série 4 : Génération du code intermédiaire (3 séances)
Série 5 : Traduction dirigée par la syntaxe (3 séances)

Travaux Pratiques :

Les Travaux pratiques consistent à réaliser un compilateur d'un mini langage. Les séances seront organisées en 2 parties.

- La première partie sera dédiée à développer l'analyseur lexical(15%),
- La seconde partie sera consacrée à la mise en œuvre de la traduction dirigée par la syntaxe qui englobe (85%):
  - l'analyse syntaxique,
  - L'analyse sémantique,
  - La génération du code intermédiaire.

Le développement du compilateur pourra se faire de deux manières :

- soit en utilisant les générateurs des analyseurs lexicaux (tel que Flex) et les analyseurs syntaxico-sémantiques avec la génération du code intermédiaire (tel que Bison)
- soit en implémentant les algorithmes de l'analyse lexicale et de la traduction dirigée par la syntaxe (Analyse syntaxique, Analyse sémantique et Génération du code intermédiaire).

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- Compilateurs: principes, techniques et outils - A. Aho, R. Sethi, J. Ullman – InterEditions.
- Compilateurs - D. Grune, H. Bal, C. Jacobs, K. Langendoen - Dunod.
- Principles of Compilers- V. Aho, J.D. Ullman - édition 2006.
- Modern compiler implementation in ML- A.W.Appel - Cambridge University Press 1998.
- The theory and practice of compiler writing - P.Q G. Sorenson - McGraw-Hill computer science series 1985.
- Flex & Bison - J.Levin- O'Reilly Media 2009.

Semestre : *S5*
Unité d'enseignement : UEM
Matière : Réseaux et protocoles
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
 Maitriser les protocoles de routage et les protocoles applicatifs
 Réussir la transition de votre réseau vers IPv6
 Simplifier la gestion des réseaux et l'attribution de configurations IP
 Construire un inter réseau IPv6 à l'aide de tunnels établis sur le réseau IPv4 existant
 Déployer la version modernisée et adaptée à IPv6 des protocoles de routage
Connaissances préalables recommandées :
Module réseau en S4, connaissances de TCP/IP .
Contenu de la matière :
1. Architectures Tcp/Ip
 Rappel (protocoles Tcp/Ip)
 Sous-adressage (statique/dynamique)
2. Protocoles de routage
 Routage statique (rappel)
 Routage dynamique (RIP, OSPF, EIGRP, BGP)
3. Réseaux locaux
 Réseaux locaux Virtuels (Vlan), Protocoles VTP, DTP, Routage inter-Vlan
 Redondances dans les LANs (STP, EtherChannel, Protocole HSRP)
4. Réseaux étendues
 Protocole PPP
 VPN (Protocole GRE)
5. Protocoles applicatifs
 Protocoles web (HTTP et HTML)
 Protocoles de messagerie électronique (SMTP, POP et IMAP)
 Services de partage de fichiers (FTP et SMB)
 DHCP et DNS
 Telnet et SSH
6. Protocole IPv6
 Adressage IPv6
 Transition IPv4/IPv6
 Services IPv6
 Sécuriser la communication IPv6 en utilisant IPsec.
7. Fibres optiques
 Introduction à la fibre optique
 Concept de bases
 Fonctionnement
Mode d'évaluation :
Contrôle continu (40%), Examen (60%)
Références :
1. Jeff Doyle, Jennifer DeHaven Carroll, « Professional DevelopmentRouting TCP/IP », Edition Cisco PressCCIE,2015
2. James Aweya, « IP Routing Protocols: Link-State and Path-Vector Routing Protocols », 1er Édition, Format Kindle, 2021c
3. Rick Graziani, « IPv6 Fundamentals: A Straightforward Approach to Understanding IPv6 »2e Édition, Format Kindle, 2017

Semestre : 5
Unité d'enseignement : UEM
Matière : Représentation des connaissances et raisonnement
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
- faire une correspondance entre le monde extérieur et un système souvent symbolique manipulable par un ordinateur,
- raisonner sur ces connaissances.

Ainsi, les objectifs de ce module s'insèrent dans cette optique. Différentes théories et formalismes (logiques et graphiques) seront étudiés afin de représenter et de raisonner sur des connaissances de natures multiples.

Connaissances préalables recommandées.

Logique mathématique, les fondements de l'Intelligence Artificielle.

Contenu de la matière :

CHAPITRE I : LES LOGIQUES NON CLASSIQUES
1- La logique modale
1-1 Notion de modalité
1-2 syntaxe
1-3 règles de déduction
1-4 Sémantique : mondes possibles, sémantique de Kripke.
1-5 Représentation des connaissances
1-6 Logique modale et les systèmes multi-agents
2- La logique des défauts
2-1 Notion de non monotonie
2-2 Concepts de base de la logique des défauts 2-3 Calcul des extensions
2-4 La logique des défauts et le data-mining
3- Les logiques de description
3-1 les bases des logiques dedescription
3-2 Mécanismes deraisonnement
3-3 les logiques de description et le traitement desimages
3-4 Les logiques de description et lesontologies

CHAPITRE II- LES MODELES DE L'INCERTAIN
1- La théorie de Dempster &Shafer
1-1 Concepts de base
1-2 Mesures de croyance et deplausibilité
1-3 Règles de combinaisonEtablissement :
1-4 La prise de décision
1-5 La fusion en traitement d'image
2- Les réseaux Bayésiens : Modèle Graphique del'incertain
2-1 Rappel sur la théorie des probabilités
2-2 Concepts de base
2-3 Algorithmes d'Inférence exacte
2-4 Domaines d'application des réseaux bayésiens

CHAPITRE III- LA THÉORIE DES SOUS-ENSEMBLES FLOUS

1- Concepts de base
2- Contrôleur flou
3- Domaines d'application de la théorie des sous-ensembles flous

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références:

- I. Bloch, R. Clouard, M. Revenu and O. Sigaud, in A Guided Tour of Artificial Intelligence Research (P. Marquis,
O. Papini, H. Prade, Eds), Springer, chap. Artifical Intelligence and Pattern Recognition, Vision, Learning, 2017.
-Yifan Yang, Jamal Atif, Isabelle Bloch: Raisonnement abductif en logique de description exploitant les domaines concrets spatiaux pour l'interprétation d'images. Revue d'IntelligenceArtificielle 31(1-2) 11-39, 2017.
- Van Benthem J., Bezhanishvili G. Modal Logics of Space. In: Aiello M., Pratt-Hartmann I., Van Benthem J. (eds) Handbook of Spatial Logics. Springer, Dordrecht, 2007.
- F. Baader. The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge UniversityPress, 2003.
- B.Bouchon-Meunier et C.Marsala, Logique floue, principes, aide à la décision, Hermès Sciences, 2002.
- D. Kayser : La représentation des connaissances. Hermes 1997.
- F. V. Jensen. An Introduction to Bayesian Networks. University College London: UCL Press, 1996.
- D.Poole, Default logic. In Handbook of Logic in Artificial Intelligence and Logic Programming: Nonmonotonic Reasoning and Uncertain Reasoning (vol. 3), D. M. Gabbay, C, 1994.
- J. Hogger, and J. A. Robinson, Eds. Oxford University Press, Inc., New York, NY, 189–215
- M. Alliot et T.Schiex, Intelligence Artificielle et Informatique Théorique, Cépaduès Editions, 1993.
- L.Sombe. Raisonnement sur des informations incomplètes en intelligence artificielle. Teknea, Marseille, 1989.
- P. Besnard, An Introduction to Default Logic. Springer-Verlag, Vienna, Austria, 1989.
- J.Pearl, Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann Publications, 1988.
- P.Smets, E.H.Mamdani, D.Dubois and H.Prade: Non-standard logics for automated reasoning. 1988.
-D. Dubois and H. Prade. Possibility Theory. An Approach to Computerized Processing of Uncertainty. Plenum Press, New York and London, 1988.
- R.Reiter, A logic for default reasoning. Artificial . Intelligence. 13, 81–132, 1980.
-D. Dubois and H. Prade. Fuzzy Sets and Systems - Theory and Applications. Academic Press, New York, 1980.
-L.A. Zadeh. Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems, 1978.

Semestre :*S5*
Unité d'enseignement : UET
Matière : Recherche opérationnelle
Crédits : 3
Coefficient : 3
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
La première partie de ce cours se propose de rendre compte des trois composantes qui s'entremêlent en théorie des graphes : Résolution des problèmes, mathématiques discrètes et algorithmiques. La deuxième partie de ce cours s'intéresse à la programmation linéaire qui est un des domaines les plus utilisés de la RO. Elle permet de traiter un vaste ensemble de problèmes d'optimisation dans des contextes divers comme la gestion de stocks, flux de transport, distribution de tâches à des personnels, recherche de plans de fabrication etc. . . La modélisation de ces problèmes débouche sur des équations ou inéquations linéaires (exprimant les différentes contraintes) dont on cherche les solutions permettant d'optimiser une fonction économique elle-même linéaire.

Connaissances préalables recommandées :
Algèbre Linéaire, Analyse matricielle

Contenu de la matière :
I. Introduction à la Recherche Opérationnelle et à la modélisation
1. Introduction à la recherche opérationnelle
2. Méthodologie de résolution d'un problème de RO
3. Modélisation et validation de modèle
4. Choix de la méthode de résolution
II. Notions fondamentales de la théorie des graphes
1. Définitions et généralités
2. Chaînes, cycles et connexité
3. Représentation matricielle d'un graphe
4. Problème de coloration (algorithmes Welch et Powel ;DSatur)
5. Problème de l'arbre couvrant de poids minimum (algorithmes kruskal et Prim)
III. Problème de cheminement
1. Parcours eulériens et hamiltoniens
2. Position du problème du plus court chemin
3. Propriétés des plus courts chemins
4. Algorithmes du plus court chemin : Djikstra, Bellman, Ford et algorithme de Floyd.
IV. Problème du flot maximum
1. Position du problème
2. Flots compatibles, complets
3. Amélioration de flots
4. Algorithme de Ford et Fulkerson
V. Problème d'ordonnancement
1. Position du problème
2. Réseau associé à un projet
3. Méthode MPM
4. Méthode PERT
VI. Programmation Linéaire
1. Problématique de la programmation Linéaire
2. Modélisation et résolution graphique
3. L'algorithme du Simplexe
4. Obtention d'une solution de base réalisable : Algorithme du simplexe de deux phases
Intitulé : Systèmes d'information et science de données Année: 2024/2025
VII. La dualité dans la Programmation Linéaire

1. La dualité et son interprétation
2. Propriétés de la dualité
3. Du problème dual au problème primal
4. L'algorithme dual du Simplexe

Travail personnel : TP optionnel

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

• L. R. Ford et D. R.Fulkerson, "Flows and networks", Princeton UniversityPress.
• M. Gondron et M. Minoux, " Graphs and Algorithms" WileyInterscience, 1984.
• R. Bronson, "Operations Research " Série Shaum, 1982.
• Dantzig G. Linear programming and extensions. Princeton university press; 2016 Aug 10.

Semestre : S6

Semestre :*S5*
Unité d'enseignement : UED
Matière : Recherche d'information
Crédits : 3
Coefficient : 3
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
L'objectif de ce cours est de dresser un panorama et un bilan des recherches en Recherche d'Informations (RI). Il s'agit de décrire les méthodes, modèles et algorithmes. L'apport des travaux de la communauté « Web Sémantique » pour la recherche d'information y est également détaillée.

Connaissances préalables recommandées
L'étudiant doit acquérir préalablement des connaissances relatives à l'algorithmique, structure de données, algèbre linéaire et la théorie des probabilités, Web Sémantique, Données semi-structurée.

Contenu de la matière :
1. Introduction à la recherche d'information
2. Modèles de RI
3. Évaluation des systèmes de recherche d'information
4. RI sur le Web (Moteur de recherche)
5. Recherche d'information semi-structurée
6. Recherche d'information sémantique : Indexation sémantique, désambiguïsation, mesures de similarité sémantique.
7. Recherche d'information multilingue.

Mode d'évaluation :
Contrôlecontinu (40%), Examen (60%)

Références :
1. Baeza-Yates, Ricardo and Berthier Ribeiro-Neto (1999) Modern Information Retrieval. Addison-Wesley, New York City, NY, ACM Press.
2. Cristopher D. M., Prabhaker R., Hinrich S. (2009) An Introduction to Information Retrieval. Cambridge UniversityPress.

Semestre : ***S6***
Unité d'enseignement : UEF1
Matière : Bases de données NoSQL
Crédits : 4
Coefficient :4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Les étudiants seront capables de maîtriser des concepts avancés des bases de données et auront une connaissance sur les différentes architectures des SGBD reposants sur les nouveaux modèles des bases de données.

Connaissances préalables recommandées :
Les notions sur les bases de données : conception, création et gestion d'une base de données relationnelle, PL/SQL.

1. **Contenu de la matière :**
2. Bases de données NoSQL : Concepts de bases
3. Bases de données orientées documents : MongoDB
4. Bases de données orientées graphe : Neo4j
5. Base de données géographiques
6. NewSQL : Concepts de base

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références *:*

1. *Georges Gardarin, Bases De Donnees(Objet/Relationnel), 1999*
2. *A. Foucret, « Livre blanc sur NoSQL », par Smile (http://www.smile.fr/Livres-blancs/Culture-du-web/NoSQL).*
3. *A.-C. Caron, « NoSQL », Université de Lille 1.*
4. *S-K. Gajendran, « A Survey on NoSQL Databases ».*
5. *Chrisment C., Pujolle G., Ravat F., Teste O., Zurfluh G., « Entrepôts de données », Dossier Techniques de l'Ingénieur, N°H3870, 2005.*
6. *Kimball R., Ross, M., « Entrepôts de données : guide pratique de modélisation dimensionnelle », 2°édition, Ed. Vuibert, 2003, ISBN : 2-7117-4811-1.*

Semestre : S6
Unité d'enseignement : UEF1
Matière : Conception de logiciels
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

Ce module vise à donner aux étudiants les compétences indispensables pour concevoir des logiciels de qualité. Il a pour but d'offrir une vision complète de la conception de logiciels, en couvrant à la fois les principes et les méthodes de base ainsi que les perspectives émergentes dans la matière. Le module s'article principalement autour de points suivants :

- Comprendre les principes de conception des gros logiciels.
- Savoir concevoir une expérience utilisateur intuitive.
- Comprendre les enjeux présents et futurs de la conception de logiciels.
- Savoir utiliser les principaux patrons de conception pour résoudre des problèmes de conception récurrents.
- Savoir utiliser des patrons d'architecture pour structurer une solution logicielle.

Connaissances préalables recommandées :

Génie Logiciel, Programmation orientée Objet

Contenu de la matière :

- **Chapitre I : Introduction**

Concepts clés de la conception de logiciels
Les enjeux présents de la conception de logiciels
Conception pour la scalabilité
Conception pour l'expérience utilisateur (UX)

- **Chapitre II : Patrons de conceptions courants**

Rappel sur les avantages et inconvénients des patrons, la description d'un patron, la classification
Réutilisation de patrons de conception
Exemples de patrons de conception

- **Chapitre III : Patterns architecturaux courants**

Le patrons MVC
Le patron PAC

- **Chapitre IV : Ingénierie Dirigée par les Modèles et conception de logiciels**

La conception dans l'IDM
La conception dans MDA (Model driven architecture)

- **Chapitre V : Les principes de conception**

Principe de responsabilité unique

Principe d'ouverture-fermeture

Principe de substitution de Liskov

Principe de séparation des interfaces

Principe d'inversion des dépendances

- **Chapitre VI : Tendances futures de la conception de logiciels (optionnel)**

Discussion sur les tendances futures de la conception de logiciels
Impacts sur les processus de développement

- **Chapitre VII : Étude de cas d'un processus logiciel : 2TUP**

Conception générique
Conception préliminaire
Conception détaillée

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références:

- Dennis, Alan, Barbara Wixom, and David Tegarden. Systems analysis and design: An object-oriented approach with UML. John wiley& sons, 2015.
- Kleppmann, Martin. "Designing Data-Intensive Applications: The Big Ideas Behind Reliable." Scalable, and Maintainable Systems 1st Edition–O'Reilly Media (2017).
- Freeman, Eric, et al. Head First Design Patterns: A Brain-Friendly Guide. " O'Reilly Media, Inc.", 2004.
- Freeman, Adam, and Adam Freeman. "The MVC Pattern, Projects, and Conventions." Pro ASP. NET Core MVC 2 (2017): 53-65.
- Rubis, Ruslan. Patterns for Enterprise Application Design and Development. Diss. Florida Atlantic University, 2017.
- Gamma, Erich, et al. "Elements of Reusable Object-Oriented Software." Design Patterns (1995).

Semestre : S6
Unitéd'enseignement : UEF2
Matière : Systèmes d'information avancés 2
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Présenter les résultats fondamentaux et les plus récents constituant une suite de spécialisation dans le domaine des systèmes d'information.

Connaissances préalables recommandées

Systèmes d'informations avancés I.

Contenu de la matière

1. Systèmes d'information multi-médias
2. Systèmes d'information coopératifs et systèmes multi-agents
3. Systèmes d'information pervasifs
4. Méthodes formelles pour la conception des systèmes d'information

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :

1. Catherine Garbay, Jean Charley, Jacques Le Maitre « le document multimédia », 191 pages, parution le 01/09/2000, Eyrolles
2. Jean-Pierre Briot, Yves Demazeau « Principes et architecture des systèmes multi-agents », 272 pages, 2001, Eyrolles
3. M.Weiser , « The computer for the 21st century »,1991

Semestre : S6
Unité d'enseignement : UEF2
Matière : La transformation digitale
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

La transformation digitale des organisations n'est pas une simple valeur ajoutée. Elle est une évolution culturelle qui modifie le mode de fonctionnement global des organisations. Tous les secteurs sont concernés :

- L'énergie, les transports, les télécommunications, la production industrielle, les soins de santé,
- L'agroalimentaire, etc. La digitalisation vise une amélioration de la performance en travaillant plus, plusrapidement et avec moins d'efforts. Objectif réalisable en appliquant les dernière techniques del'intelligence artificielle.

Ce cours permet aux étudiants d'apprendre comment assurer une transformationdigitale pour une organisation en se basant sur les connaissances, de l'intelligence artificielle, acquiseshdurant la formation.

Connaissances préalables recommandées :
Connaissances acquises durant le cursus de formation de la licence : Systèmes informatiques (SI)

Contenu de la matière :
•Introduction
•Numérisation ou digitalisation
•Les avantages de la transformation digitale
•Rôle stratégique des TIC dans la Transformation digitale
•La transformation digitale généralisée
•Les éléments clés de la transformation digitale
•Processus de transformation digitale (étapes et outils)
•Les bâtiments intelligents et les villes intelligentes
•E-gouvernance, E-éducation, E-sante, E-agriculture
•Digitalisation et intelligence artificielle
•L'éthique de la digitalization

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :
DELORME, Pascal et DJELLALIL, Jilani. La transformation digitale: Saisir les opportunités du numérique pour l'entreprise. Dunod, 2015.
Brennen, J. S., &Kreiss, D. (2016). Digitalization. The international encyclopedia of communication theory and philosophy, 1-11.
Schallmo, D. R., & Tidd, J. (2021). Digitalization. Springer International Publishing.

Semestre :*S6*
Unité d'enseignement : UEM
Matière : Réseaux avancé 2
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Ce module permet d'initier les étudiants aux concepts de base du réseau SDN et de la virtualisation des fonctions réseau. En particulier, ce module consiste à :

- Découvrir comment SDN et NFV font évoluer les réseaux traditionnels, conduisent l'évolution de la 5G et facilitent la gestion des réseaux sans fil.
- Créer, configurer et déployer les technologies SDN et NFV les plus populaires pour créerdes réseaux, contrôler le routage.
- Intégrer ces composants à un environnement physique émulé et effectuer des tests devérification.

Connaissances préalables recommandées :
Connaissances acquises dans le module « Réseaux et protocoles » du S5

Contenu de la matière :
1. Introduction au réseau SDN
1. Réseau traditionnel
1.1 Plan de contrôle
1.2 Plan de données
1.3 Plan de gestion
1.4 Limites des réseaux traditionnels
2. Le réseau SDN
3. SDN vs réseaux traditionnels
4. Historique et évolution du SDN
5. L'architecture du réseau SDN
6. Le découpage du plan de transmission-contrôle
7. Les couches d'abstraction
8. Les contrôleurs physiques et virtuels
9. Les SDN dans les réseaux locaux
10. Les contrôleurs LAN applicatifs
11. Les acteurs du SDN open networking foundation IETF
12. Les contrôles de l'infrastructure (centralisé ou distribué)
13. L'automatisation de la gestion du réseau
14. La migration de réseaux virtuels
15. Gestion de sécurité, les contrôles d'accès et équilibrage de charge

2. OpenFlow
1. Concepts de base
2. Ports openflow
3. Tables openflow
4. Messages openflow
5. Le protocole de commutation openflow
6. Architecture de commutateur compatible OpenFlow
7. Catégorisation des messages échangés entre un commutateur et un contrôleur
8. Connexion entre les hôtes sur l'environnement OpenFlow
9. Implémentation Openflow dans les périphériques réseau

3. Contrôleurs SDN
1. Concepts généraux

2. Contrôleur centralisé
3. Contrôleur distribué
4. Placement et redondance des contrôleurs
5. Présentation de contrôleurs existantsOnos, Nicira: NVP – NEC: Trema – Big Switch Networks: Floodlight/BNC – Juniper/Contrail – CiscoXNC

4. Virtualisation des fonctions réseau (NFV)

1. Introduction
2. Qu'est-ce que la virtualisation des fonctions réseau ?
2.1 Terminologies principales
2.2 Activités de normalisation
3. Couche d'infrastructure de virtualisation des fonctions réseau
4. Couche de gestion et d'orchestration NFV
5. Couche de fonction de réseau virtuel
6. Algorithmes VNF
6.1 Placement VNF
6.2 Ordonnancement VNF
6.3 Migration VNF
6.4 Chaînage VNF
6.5 Multidiffusion VNF
7. NFV et SDN cas d'utilisation
7.1 Mobilité avec l'aide du SDN
7.2 Communications machine à machine (M2M)
7.3 Réseauxcoopératifs

5.SD-WAN (Software-defined Wide Area Network)

1. Concepts du réseau WAN
2. Défis du réseau WAN
3. Réseau SD-WAN
3.1 Définition
3.2 Objectifs
3.3 Solutions
4. L'architecture SD-WAN
5. Déploiement SD-WAN
6. SD-WAN contre MPLS

6 .W-WAN (Wireless Software-defined Network)

1. Concepts du réseau W-WAN
2. Virtualisation du réseau core mobile
3. W-WAN et réseau cellulaire
4. W-WAN et réseau de capteurs
5. W-WAN et le réseau mech

7.SDN, programmation de réseau

1. Présentation de langage P4
2. Présentation des fonctionnalités et de l'architecture de programmation P4
3. Programmation avec le langage P4
4. Configuration du commutateur logiciel P4
5. Compilation du programme P4
6. Exécution du programme P4
7. P4Runtime

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

1. Huawei Technologies Co., Ltd..(2023). SDN and NFV. In: Data Communications and Network Technologies. Springer, Singapore. https://doi.org/10.1007/978-981-19-3029-4_17
2. https://opennetworking.org/wp-content/uploads/2014/10/openflow-spec-v1.4.0.pdf
3. IsraatTanzeena Haque, Nael B. Abu-Ghazaleh: Wireless Software Defined Networking: A Survey and Taxonomy. IEEE Commun. Surv. Tutorials 18(4): 2713-2737 (2016)
4. Abbas Bradai, Mubashir Husain Rehmani, Israat Haque, Michele Nogueira, Syed Hashim Raza Bukhari: Software-Defined Networking (SDN) and Network Function Virtualization (NFV) for a Hyperconnected World: Challenges, Applications, and Major Advancements. J. Netw. Syst. Manag. 28(3): 433-435 (2020)
5. P4_D2_East_2018_01_basics.pdf (opennetworking.org). https://opennetworking.org/wpcontent/uploads/2020/12/P4_D2_East_2018_01_basics.pdf

Semestre : S6
Unité d'enseignement : UEM
Matière : Web sémantique
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Comprendre les différentes facettes relatives à l'ingénierie ontologique, Comprendre et concevoir une architecture orientée service et Mise en oeuvre et déploiement d'applications à base de Web services.

Connaissances préalables recommandées :
Connaissances acquises durant le cursus de formation de la licence : Systèmes informatiques (SI) Programmation Java, Bases de données, XML, XSLT, XPATH, XQuery, SAX et DOM

Contenu de la matière :
 Rappels sur XML, XSLT, XPATH, XQuery, SAX et DOM
 Définition de l'ontologie
 Différents types d'ontologies (thesaurus, ontologies descriptives, …) et introduction à l'ingénierie ontologique
 Formalisation d'ontologies en RDF (Ressource Description Framework), son extension aux schémas (RDFS).
 Un modèle ontologique basé sur la Logique de Description pour le Web sémantique: le Projet du W3C Consortium groupe « WebOnt» : OWL (Ontology Web Language)
 Architecture orientée services : concepts et mise en oeuvre.
 Technologies standards des services Web
 Interactions entre services : échanges de messages, composition
 Conception et réalisation de services Web

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :

1. *Le web sémantique, Comment lier les données et les schémas sur le web ? ; Fabien L. Gandon, Olivier Corby, Catherine Faron-Zucker ; Editeur(s) : Dunod ; 2012.*
2. *Programming the Semantic Web ; Build Flexible Applications with Graph Data; Toby Segaran, Colin Evans, Jamie Taylor ; Editeur(s) : O'Reilly ; 2009.*

Semestre : *S6*
Unité d'enseignement : UET
Matière : Analyse de Données
Crédits : 3
Coefficient : 3
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
L'Analyse des Données forme le domaine des méthodes exploratoires qui permettent de révéler et montrer les principales liaisons statistiques présentes dans une base de données. Elle a pour objectif de donner une vision synthétique et interprétable de l'organisation naturelle de données.
Ce module est une introduction à l'analyse des données en présentant les différents types de tableaux et les méthodes d'analyse multidimensionnelle. Ensuite on présente les techniques par réduction des dimensions, telles que l'analyse par composantes principales, l'analyse factorielle des correspondances et l'analyse des correspondances multiples. Enfin, on termine par les méthodes de classification et d'analyse discriminante.

Connaissances préalables recommandées :

Contenu de la matière :
1. Présentation de l'analyse des données (types de tableaux analysables)
2. Analyse en composantes principales (analyse du nuage des individus et de l'espace des variables, interprétation, qualité des résultats)
3. Analyse des correspondances (analyse factorielle des correspondances, analyse des correspondances multiples)
4. Analyse discriminante (aspects descriptif et décisionnel)
5. Régression linéaire
6. Méthodes de classification (classification supervisée et non supervisée méthodes de partitionnement, classification ascendante hiérarchique)

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :
1. Brigitte Escofier, Jérôme Pagès. Analyses factorielles simples et multiples. 5e édition. DUNOD, 2016. · Michel Volle. Analyse des données. 4e édition, Economica, 1999.
2. · Michel Crucianu. Méthodes factorielles pour l'analyse des données, Hermes, 2008.
3. · Adresses de sites Web à consulter https://www.univdocs.com/2020/05/analyse-de-donnees.html http://eco.um1.free.fr/doc/Semestre_6/Analyse_de_donnees/cours/bb2b9646e3772dfc142fe67c994fa6d9.pdf

Semestre : *S6*
Unité d'enseignement : UED
Matière : Systèmes de recommandation
Crédits : 3
Coefficient : 3
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
In the current age of information overload, recommender systems offer personalized access for users to efficiently search information and make choices online. This course introduces recommender systems' major concepts and algorithms. A variety of real-world applications are included, such as those deployed in e-commerce sites and social networks

Connaissances préalables recommandées :
Connaissances acquises dans le module « Recherche d'information » du S5

Contenu de la matière :
I. Basic concepts of recommendersystems
A. User preference and ratings
B. Prediction
C. Taxonomy of recommenders
II. Recommendation algorithms
A. Collaborative filtering
B. Matrix Factorization
C. Content-based recommendation
D. Others
III Applications
A. E-commerce websites
B. Social networks
C. Others (e.g., news, healthcare, etc.

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :
1-Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possibleextensions. IEEE Transactions on Knowledge & Data Engineering,(6):734–749, 2005.

2- Yilmaz Ar and Erkan Bostanci. A genetic algorithm solution to the collaborative filtering problem. Expert Systems with Applications, 61:122–128,2016

3-Hendrik Drachsler, Katrien Verbert, Olga C Santos, and Nikos Manouselis. Panorama of recommender systems to support learning. In Recommendersystemshandbook, pages 421–451. Springer, 2015

Semestre : S7
Unité d'enseignement : UEF1
Matière : Entreprise Resource Planning
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

Ce module a pour objectif de fournir un aperçu sur les fonctionnalités d'un ERP, sa configuration de base et le développement d'un nouveau module. Les TP se baseront sur L'ERP open source « Odoo ».

Connaissances préalables recommandées :

Système d'informations, Algorithmique.

Contenu de la matière :

- Installation de Odoo ;
- Présentation fonctionnelles ;
- Gestion des droits d'accès ;
- Création d'un module ;
- Respect de l'architecture MVC (Model Vue Controler) ;
- Interaction avec l'ORM (Object Relational mapping) ;
- Réalisation de rapports ;
- Actions automatisées (envoi d'un email, dépôt de fichiers, ...) ;
- Interfacer Odoo avec une autre application.

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- Odoo Development Essentials, 2015, Daniel Reis, PacktPublishing;
- http://sebastien.nameche.fr/supports/Le_framework_Odoo_v0.1_20151125.pdf
- https://www.odoo.com/documentation/9.0/

Semestre : S7
Unité d'enseignement : UEF1
Matière : Développement Web
Crédits : 5
Coefficient : 5
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Ce cours vise à doter les étudiants des connaissances et des compétences nécessaires pour construire des applications Web complexes utilisant des technologies avancées front-end et back-end. Pour la technologie frontend, ce cours se concentre sur React, où les étudiants acquerront une compréhension approfondie des Principes et techniques de React, y compris les composants, la gestion de l'état et le routage. De plus, le deuxième objectif principal de ce cours est de comprendre le développement back-end. Ici, les étudiants découvriront les technologies côté serveur et en particulier comment créer des API RESTful pour gérer le stockage, la récupération et la manipulation des données. De plus, les étudiants apprendront et appliqueront les meilleures pratiques modernes en matière de modularisation, de test, de documentation, de sécurité et de déploiement du code. À la fin du cours, les étudiants seront capables de créer des applications full-stack intégrant des technologies front-end et back-end pour créer des applications Web complètes avec des interfaces utilisateur interactives et une gestion robuste des données.

Connaissances préalables recommandées :
Concepts fondamentaux du développement web enseignés en 3ème année (Développement d'Applications Web)

Contenu de la matière :
Chapitre 0 : Concepts de base (1 semaine) :
Présentation générale du World Wide Web (WWW)
Types d'applications Web, composants et caractéristiques clés
Quelques concepts importants : Types de rendu (SSR, CSR, Hybride, Pré-rendu),
Optimisation des moteurs de recherche (SEO), application à page unique (SPA), Web progressif Applications (PWA),...
Comment fonctionne le Web : DNS, hébergement et serveurs

Chapitre 1 : Sémantique HTTP (1 semaine)
Mécanisme de demande/réponse
Méthodes HTTP
En-têtes HTTP
Statut des codes
Mécanisme de mise en cache
Négociation de contenu

Chapitre 2 : Développement Front-End avec React (4 semaines)
Concepts JavaScript avancés :
JavaScript asynchrone : rappels, promesses et asynchrone/attente
Fonctionnalités JavaScript ES6 et au-delà
Fondamentaux de React :
JSX, composants, état et accessoires,…
Cycle de vie des composants, hooks (useState, useEffect, etc.), …
Construire des composants réutilisables et des bibliothèques de composants
Concepts de réaction avancés :
Routage côté client avec React Router
Formulaires et validation des formulaires
Gestion de l'état avec Redux ou Context API

Test des composants React (Jest, Enzyme)
Création d'applications à page unique avec React
Introduction àReactContext et aux hooks avancés
Optimisation des performances.

Chapitre 3 : Développement Back-End avec React (4 semaines)
Introduction aux technologies et frameworks côté serveur
Style REST : Comprendre le transfert d'état représentatif (REST), les principes et contraintes de l'architecture RESTful
Principes de conception d'API RESTful
Construction et consommation d'API REST : opérations CRUD, méthodes et outils de test.
Authentification et autorisation
Documentation API
Bonnes pratiques de sécurité pour les applications Web
Déploiement et gestion d'applications Web

Chapitre 4 : Création d'applications Full-Stack (3 semaines)
Intégration du front-end de React avec les API back-end
Mise en oeuvre de l'authentification et de l'autorisation des utilisateurs
Création d'interfaces utilisateur complexes avec interaction avec les données
Implémentation de fonctionnalités temps réel.
Meilleures pratiques :
Outils de gestion de projet et de collaboration,
Test et débogage d'applications full-stack
Déploiement et mise à l'échelle d'applications Web
Introduction au framework MERN Stack.
Contenu des travaux pratiques et recommandations :
Présentation de quelques concepts HTTP en utilisant Node.js.
Développement front-end : JavaScript avancé et React. Développement back-end : Node.js et npm, Express Framework, Intégration de bases de données
(Utilisation de MongoDB avec Mongoose) et authentification avec des jetons Web JSON et bcrypt.
Tests et déploiement : intégration de tests pour React et Node.js.
Présentation du framework MERN Stack.
Étude de cas pour montrer les compétences full stack.

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références:
· R. Fielding, M. Nottingham, J. Reschke, E "RFC9110: HTTP Semantics", 2022,RFC 9110: HTTP Semantics (rfc-editor.org)
· Daniel Tanner:"React JS: A Beginner's Guide to Building Dynamic User Interfaces", 2023.
· Greg Lim:"Beginning Node.js, Express & MongoDB Development", 2019.
· Leonard Richardson and SAM Ruby:"RESTful Web Services", O'Reilly Media,2007.
· Roy Thomas Fielding:"Architectural Styles and the Design of Network-based Software Architectures, Representational State Transfer REST », University of California, Irvine, 2000.
"How to use MERN Stack Framework,
https://www.mongodb.com/languages/mernstack-tutorial

Semestre : S7
Unité d'enseignement : UEF2
Matière : Ingénierie des exigences
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Au terme de l'enseignement de cette matière, l'étudiant sera en mesure d'analyser les besoins des logiciels et d'en rédiger un cahier de charge (document de spécifications) pour le développement ou l'acquisition d'un logiciel.

Connaissances préalables recommandées :
L'étudiant de cette matière devrait avoir les connaissances de base en analyse et conception des logiciels acquises avant.

Contenu de la matière :
1. Ingénierie des exigences: Concepts de base
2. Ingénierie des exigences : Typologie d'exigences
3. Ingénierie des exigences : Niveaux d'exigences
4. Ingénierie des exigences : Processus
5. Ingénierie des exigences : Styles de spécification
6. Ingénierie des exigences : Documentation des exigences
7. Ingénierie des exigences : Conflits entre exigences
8. Ingénierie des exigences : Gestion du changement des exigences

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références:
Lauesen S, Software Requirements: Styles and Techniques, Addison-Wesley (2002)
Sommerville I, Software Engineering, Pearson Education (2013)

Semestre : 7
Unité d'enseignement : UEF2
Matière : Interopérabilité des Systèmes d'information
Crédits : 5
Coefficient : 5
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Des applications émergentes telles que le e-gouvernement, le e-learning, les bibliothèques électroniques, etc.nécessitent l'accès à diverses sources d'information. Ces dernières sont généralement hétérogènes, que cesoit au niveau syntaxique ou sémantique. En effet, des conflits sémantiques surviennent puisque lessystèmes n'utilisent pas la même interprétation de l'information qui est définie différemment d'uneorganisation à l'autre.
Connaissances préalables recommandées :
Contenu de la matière :
1. Définition de l'interopérabilité (1h)
2. Évolution de l'interopérabilité(2h)
3. Propriété d'un système pour l'interopération (2h)
4. Problèmes d'hétérogénéité des données : (3h)
- Conflits syntaxiques
- Conflits structurels
- Conflits sémantiques
5. La sémantique dans le processus d'interopérabilité (2h)
- Les Méta données
- L'Ontologie
- le contexte
6. Les approches d'interopérabilité (3h)
- Approche fédérée
- Approche à base de médiation
- Approche basée sur le contexte
-…
7. Application : (2h)
-Interopérabilité de SIG
- Interopérabilité des services Web

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :

- D. Benslimane, et al. Interopérabilté de SiG : un Etat de l'art, revue internationale de géomatique , Vol9,3 1999
- Euzenat, J. (2001). An infrastructure for formally ensuring interoperability in a heterogeneous semanticweb. In Cruz, I. F., Decker, S.,
- Visser, P. R. S., Jones, D. M., Bench-Capon, T. J. M., and Shave, M. J. R. (1997). An analysis ofontological mismatches: Heterogeneity versus interoperability. In AAAI 1997 Spring Symposium onOntological Engineering, Stanford, USA.

Wache, H. and Stuckenschmidt, H. (2001). Practical context transformation for information systeminteroperability. In Akman, V., Bouquet, P., Thomason, R., and Young, R. A., editors, Modeling andUsing Context, Third international and Interdisciplinary Conference, CONTEXT 2001, number 2116 inLecture Notes in Artificial Intelligence,Springer

Semestre : S7
Unité d'enseignement : UEM
Matière : Sécurité informatique
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Ce cours vise à :

- Sensibiliser l'étudiant aux problèmes de sécurité informatique.
- Présenter les aspects fondamentaux de la sécurité informatique.
- Savoir réaliser des analyses de risque.
- Familiariser l'étudiant avec les aspects de la cryptographie.
- Savoir utiliser quelques outils cryptographiques pour réaliser un service de sécurité.
- Identifier et corriger les failles possibles aussi bien au niveau utilisation d'un système d'exploitation qu'au niveau construction d'un logiciel.

Connaissances préalables recommandées :

Contenu de la matière :
I. Concepts de base (5h)
1. Motivation
- Sensibilisation des étudiants aux problèmes de sécurité par les chiffres
- Sensibilisation des étudiants aux problèmes de sécurité par les exemples : virus, ver, cheval de Troie, spyware, spam, etc.
2. Généralités
- Définition de la sécurité informatique
- Objectifs de la sécurité informatique
- Les menaces/ Les niveaux de vulnérabilités
3. Analyse de risque
TD : faire des tableaux d'analyse des risques suivant des scénarios donnés.
II. Introduction à la cryptographie (10h)
1. Objectifs de la cryptographie (confidentialité, intégrité, authentification, etc.)
2. Définition cryptographie/cryptanalyse
3. Chiffrement/Déchiffrement/Clé de chiffrement et notion d'entropie
4. Chiffrement symétrique (DES, AES, RC4)
5. Chiffrement asymétrique (RSA, ElGamal, EC)
6. Autres primitives cryptographiques
- Hachage cryptographique et intégrité
- MAC/HMAC et authentification
- Signature électronique
7. Principe de gestion de clés
- Présentation du problème
- Echange de clé par Diffie-Hallman
- Infrastructure à clés publics
Modèle décentralisé
Modèle hiérarchique et certificats
8. Méthodes de cryptanalyse de base et protection des clés
- Quelques protocoles cryptographiques
- Types d'attaques possibles
- Protocoles d'authentification de l'origine
- Protocoles d'authentification forte par défi/réponse

TD/TP du chapitre II : Atelier OpenSSL pour utiliser la cryptographie au profit de la sécurité des données et des échanges.

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

References :

- W. Talligs, « Sécurité des réseaux : Applications et Standards », Vuibert, 2002.
- B. Schneier, « Cryptographie appliquée : Algorithmes, protocoles et codes source en C », Vuibert, 2002.
- G. Dubertret, « Initiation à la cryptographie », Vuibert 1998.
- « Les principes de la sécurité informatique : Guide d'audit », IFACI, PARIS.

Semestre : ***S7***
Unité d'enseignement : UEM
Matière : ***Fondements de l'IA***
Crédits : 4
Coefficient : 3
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
L'objectif de ce module est de présenter les fondements du domaine de l'intelligence artificielle. Dans ce cours, les principes de base des différents éléments de l‟intelligence artificielle seront présentés : Logiques et Raisonnement, Résolution de problèmes, Méta heuristiques, Problème de satisfaction de contraintes, Apprentissage automatique et réseaux de neurones, Systèmes multi-agents.

Connaissances préalables recommandées :

Contenu de la matière :
Chapitre 1 : Logiques et Raisonnement
1.1 Logique propositionnelle
1.2 Logique du 1er ordre
1.3 Introduction aux logiques non classiques
Chapitre 2. Résolution de problèmes de planification
2.1. Représentation d‟un problème par un espace d‟états
2.2. Méthodes de recherche de solution dans les espaces d‟états
Chapitre 3. Algorithmes pour les jeux
3.1 Algorithme Min Max
3.2 Algorithme Alpha Beta
Chapitre 4. Les méta-heuristiques
4.1 La recherche locale
4.2 Méthode SimulatedAnnealing (Recuit Simulé)
4.3 Algorithme génétique
Chapitre 5. Problème de satisfaction de contraintes (CSP)
5.1 Exemples de CSP
5.2 Recherche en arrière pour les CSPs (BackTracking)
5.3 Algorithme AC-3
Chapitre 6. Apprentissage automatique et réseaux de neurones
6.1 Algorithmes d‟apprentissage automatique
6.2- Méthode des K plus proches voisins
6.3- Les réseaux de neurones
Chapitre 7 : Les systèmes multi-agents
7.1 Concepts de base
7.2 Les systèmes multi agents. (SMA) : Interaction, Communication, Négociation et Architecture

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références *:*
[1] Principles of Artificial Intelligence par Nils J. Nilson, 1982, Springer-Verlag Berlin Heidelberg.
[2] Essentials of Artificial Intelligence par Nils J. Nilson, 1993, Copyright: © Morgan Kaufmann
[3] ArtificialIntelligence: A new synthesis, Nils Nilson, 1997, Elsevier.
[4]Artificial Intelligence: A Modern Approach par Stuart Russell and Peter Norvig, Ed. Pearson, 2016
[5] Rapport de synthèse - France Intelligence Artificielle. Ministère de l'Enseignement supérieur et de la Recherche, 2017.

Semestre : S7
Unité d'enseignement : UED
Matière : Ingénierie des systèmes
Crédits : 4
Coefficient : 2
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
L'objectif de ce module est de présenter les principes de base de l'ingénierie des systèmes. Ce module initie les étudiants à toutes les étapes du cycle de vie d'un système complexe et pluridisciplinaire de l'émergence du besoin jusqu'au retrait du système.

Connaissances préalables recommandées :
Génie logicielle.

Contenu de la matière :
- Définitions de base ;
- Système à faire : Le système étudié ;
- Système pour faire : Le projet ;
- Définitions des besoins et exigences ;
- Conception du système ;
- Intégration, vérification, validation et qualification ;
- Maitrise des risques.

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :
- Découvrir et comprendre l'ingénierie système, 2012, Serge Fiorèse et Jean-PierreMeinadier, CépaduèsÉditions;
- Systems Engineering and Analysis, 2013, Benjamin Blanchard et Wolter Fabrycky,Pearson.
- INCOSE systems engineering handbook, 2016, International Council on SystemsEngineering (INCOSE)

Semestre : S8
Unité d'enseignement : UEF1
Matière : Architecture et gestion des Systèmes d'information avancés
Crédits : 5
Coefficient : 5
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

Les systèmes d'information modernes ou avancés ont su tirer profit des évolutions tant sur le plan des technologies que celui des modes d'organisation et de management. Tout système d'information est perceptible sous quatre dimensions que sont l'informationnel, l'organisationnel, le technologique et le managérial. Parmi les objectifs de cette première partie du module :

- Expliquer le principe de l'évolution des SI selon les 04 dimensions ;
- Comprendre le passage d'une culture verticale vers celle horizontale ;
- Différencier entre la vision orientée fonction et celle orientée processus d'entreprise ;
- Découvrir les enjeux de la cartographie des processus pour l'amont (stratégique) et l'aval (opérationnel) d'une organisation.
- Apprendre à manipuler les formalismes et les outils de modélisation d'entreprises et de processus ;
- Avoir une idée sur l'architecture des systèmes de gestion de workflow ou BPMS ;
- … etc.

De plus, les systèmes d'information ont connu des avancées technologiques qui font que le marché en la matière offre divers choix de solutions pour l'automatisation allant de la simple suite logicielle passant par les progiciels de gestion intégrés (ERP) jusqu'aux architectures urbanisées offrant une vision plus large et plus flexible pour la gestion des SI généralement complexes. Dans cette partie, nous verrons entre autres :

- Découvrir des concepts avancés comme l'agilité des SI ;
- Distinguer entre les visions intégrées et fragmentées dans les solutions d'automatisation de SI ;
- Comment sont apparus les premières formes d'ERP (ou MRP) ;
- Les ERP(s) modernes et leurs diverses classifications ;
- Les ERP(s) Open-Source : leurs forces et leurs faiblesses ;
- Comment configurer et exploiter un ERP ;
- Découvrir ce qu'est l'urbanisation en SI, ses fondements et principes ;
- Comprendre le projet d'urbanisation et sa gestion ;
- Manipuler quelques outils ;

Connaissances préalables recommandées :

Systèmes d'information, Bases de données, Génie logiciel

Contenu de la matière :

Chapitre I : Les Quatre Dimensions d'un SI

1. Notions et Rappels sur les SI
2. Dimensions de Reix
    a. Informationnelle
    b. Organisationnelle
    c. Technologique
    d. Managériale

Chapitre II : Le Business Process Management (BPM)

1. Modalités de la coopération/collaboration
2. Changement de culture
    a. Orienté donné vers orienté information
    b. Orienté fonction vers orienté processus
3. Evolution vers le BPM
    a. Visions W3C vs WFMC
    b. Autres consortiums
4. Concepts de base et classifications
    a. Processus
    b. Taxonomie des processus
5. Cycle de vie du BPM
    a. Modèles de cycle de vie
    b. CPI (ContinuousProcessImprovement)
    c. BPR (Business ProcessReingineering)
6. Modélisation des processus métier
    a. Objectifs de la modélisation graphique
    b. Quelques formalismes graphiques
    c. Le langage BPMN
7. Concepts et enjeux de la cartographie
    a. Cartographie du niveau stratégique
    b. Cartographie du niveau opérationnel
8. Le projet BPM (démarche, méthodologies et outils)

Chapitre III : La Technologie Workflow

1. Quelques définitions de base (processus, activité, acteur, rôle, contrôle de flux, instance, …)
2. Les modèles de référence et organismes de standardisation
    a. Modèle aux cinq interfaces
    b. Exemples d'outils conformes (Staffware, ProcessMaker)
3. Interopérabilité des systèmes de gestion de workflow
4. Architecture de workflow Inter-organisationnel
    a. Modèle en « sous-traitance »
    b. Modèle « faiblement couplé »
    c. Modèle « chaîné »
5. Workflow à base de services
    a. Eléments de SOA et services Web
    b. Le langage BPEL
6. Système de gestion de WF (WFMS)
    a. Architecture d'un WFMS
    b. Quelques outils : Bonita, Oracle BPM et Yawl/Woped

Chapitre IV : Agilité des SI

1. Concepts et niveaux d'agilité
2. Architecture d'entreprise agile
3. Gouvernance d'un SI agile
4. Alignement stratégique du SI comme outil

Chapitre V : Évolution technologique des SI

1. Vision intégrée vs vision fédérée des SIQ (SAI)
2. L'EAI
3. L'ERP
    a. Architecture des premiers ERP : MRP

b. ERP classique
c. ERP vertical vs horizontal
d. Les ERP et la vision processus : ERP vs BPMS
Combinaison ERP et BPM
e. Typologie et classification des ERP sur le marché
Présentation de quelques ERP (Open-Concerto, Odoo, SAP, …)
f. Le projet de mise en place d'un ERP dans une entreprise
(Objectifs, coûts, couverture fonctionnelle, cahier des charges, conduite du changement, …)

Chapitre VI : Urbanisation des SI Complexes

1. Complexité des SI : un résultat de l'évolution d'entreprise
   a. Notion de valeur ajoutée : chaîne de Porter
2. Urbanisation : principes et fondements
   a. Bases et origines
   b. Terminologie
   c. Plan d'occupation du sol (POS)
3. Concepts d'une architecture urbanisée
   a. Processus
   b. Fonctions
   c. Applications
   d. Infrastructures
4. Le projet d'urbanisation : Équipes, processus et organisation

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

1. Robert Reix, Robert Fallery, Michel Kalika, Frantz Rowe (2016). "Systèmes d'information et ***management***". 7ème edition. Editeur : Vuibert.
2. Will Van Der Aalst, Kees Max Van Hee (2004). "***Workflow Management: Models, Methods, and Systems***". The MIT Press.
3. Will Van Der Aalst, (2016). "***Process Mining. Data Science in Action***". Editeur : Springer.
4. SetragKhoshafian, Marek Buckiewics (1998). "***Groupware et Workflow***". Editions Eyrolles.
5. SetragKhoshafian (2006). "***Service Oriented Enterprise***". Editions Taylor & Francis.
6. Jean-Louis Tomas, Yossi Gal (2011). "***ERP et conduite des changements : Alignement, sélection et déploiement***". Edieur : DUNOD.
7. Ellen Monk, Bret Wagner (2012). "***Concepts in Enterprise Resource Planning***". 4th Edition. Editor : Cengage Learning.
8. Jacques Sassoon (1998). "***Urbanisation des systems d'information***". Editor : Hermes Science Publication.
9. Philippe Eynaud, Daniel Alban, Julien Malaurent, Jean-Loup Richet, Claudio Vitari (2019). "***Information Systems Management : Governance, Urbanisation and Alignement***". Editeur : Wiley-ISTE.
10. CRIPP : Cadre de Référence International des Pratiques Professionnelles de l'audit interne. Etablit par l'IIA (Institute of InternalAuditors).

Semestre : 8
Unité d'enseignement : UEF1
Matière : Développement mobile
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Ce cours fournit une base de référence pour créer des applications mobiles à l'aide de Flutter et Dart. Les participants apprendront les bases du langage de programmation Dart et comment développer des applications mobiles multiplateformes avec Flutter.

Connaissances préalables recommandées :
• Compréhension de base des concepts de programmation
• La connaissance des langages de programmation orientés objet est bénéfique mais pas obligatoire
Contenu de la matière :
Chapitre 1 : Introduction à Flutter
• Présentation du framework Flutter
• Avantages de Flutter pour le développement d'applications mobiles
• Comprendre l'architecture de Flutter
• Mise en place de l'environnement de développement Flutter
Chapitre 2 : Commencer avec Dart
• Introduction au langage de programmation Dart
• Variables, types de données et opérateurs
• Flux de contrôle : instructions if-else, boucles
• Fonctions et paramètres
Chapitre 3 : Créer des interfaces utilisateur avec Flutter
• Widgets et mise en page Flutter
• Gestion des entrées et des gestes de l'utilisateur
• Styliser et thématiser les applications Flutter
• Navigation et routage dans les applications Flutter
Chapitre 4 : Gestion de l'état dans Flutter
• Comprendre les widgets avec et sans état
• Gestion de l'état avec setState()
• Utilisation d'un fournisseur pour la gestion de l'état
• Introduction au modèle BLoC
Chapitre 5 : Travailler avec des données dans Flutter
• Récupération de données à partir des API REST
• Analyse des données JSON dans Flutter
• Stockage des données localement à l'aide de SQLite ou SharedPreferences
• Travail avec les formulaires et validation
Projet final
• Les participants travaillent sur un projet final pour appliquer les concepts appris tout au long du cours. Le projet consistera à concevoir et développer une application mobile multiplateforme fonctionnelle utilisant Flutter et Dart.
Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références:

TEXT BOOKS:

1. T1. Lauren Darcey and Shane Conder, “Android Wireless Application Development”, Pearson Education, 2nd ed. (2011)

REFERENCE BOOKS:

1. R1. Reto Meier, “Professional Android 2 Application Development”, Wiley India Pvt Ltd
2. R2. Mark L Murphy, “Beginning Android”, Wiley India Pvt Ltd
3. R3. Android Application Development All in one for Dummies by Barry Burd, Edition: I

Semestre : S8
Unité d'enseignement : UEF2
Matière : Architectures logicielles
Crédits : 5
Coefficient : 5
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

- ✓ Maîtriser l'ensemble des compétences indispensables au métier de développeur et d'architecte logiciel.
- ✓ Apprendre à utiliser les techniques de conception, de modélisation et d'implémentation selon l'approche composants, services et architectures logicielles.
- ✓ Apprendre à réfléchir au développement en termes d'architectures logicielles.

Connaissances préalables recommandées :

Génie logiciel, programmation orientée objet

Contenu de la matière :

1. **Introduction et généralités**
   - Introduction.
   - Les architectures distribuées et les architectures centralisées.
   - Les architectures n-tiers.
   - Les modèles de structuration des systèmes et applications.
   - Présentation et comparaison des types d'architectures : multi-niveaux, clients/serveurs, extensibles, dynamiques,…etc.
2. **Les composants et les modèles de composants logiciels**
   - Introduction et définitions.
   - Object vs Composant.
   - Organisation logique d'un composant.
   - L'aspect structurel et comportemental.
   - L'approche par composant : principe et concepts de base.
   - Avantages et inconvénients.
   - Cycle de vie.
   - Présentation et études de quelques modèles de composants (EJB3, UML, .Net…etc.).
3. **Les architectures logicielles**
   - Introduction.
   - Principe et objectif.
   - Architecture vs. Conception.
   - Les modèles d'architectures.
4. **Les langages de description d'architecture logicielle (ADLs)**
   - Introduction.
   - Les concepts de base des langages de description d'architecture (Les composants, les connecteurs et les configurations).
   - Description des principaux ADLs.
   - Classification des ADLs.
   - L'aspect statique dans les ADLs.
   - L'aspect dynamique dans les ADLs.
   - Vérification et validation.
5. **Conception architecturale**
   - Les bases de la conception architecturale.
   - Les approches de conception architecturale.
   - Développer un modèle architectural.
   - Exemple d'application.

6. **Les styles architecturaux**
7. **Les architectures orientées Services (SOA)**
    - Introduction.
    - Les concepts de base (service, annuaire de services, …etc).
    - L'architecture orientée services.
    - Les objectifs de l'architecture orientée services.
    - Les principes de l'architecture l'orienté service.
    - L'approche orientée services.
    - Technologies basées sur les principes de SOA.
    - Les services web : définitions et caractéristiques.
    - Les applications des services web.
    - Le fonctionnement des services web.
    - Architecture des services web.
    - Protocoles et normes.
    - Les architectures micro services.
8. **Le déploiement**
    - Introduction.
    - Déploiement sur des infrastructures centralisées.
    - Déploiement sur des infrastructures distribuées.
    - Déploiement sur le Cloud.
    - Autres solutions et technologies pour le déploiement.

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- M. OUSSALAH. Ingénierie des composants: Concepts, techniques et outils. Éditeur Vuibert (20 juin 2005). ISBN-10 : 2711748367.
- Projet ACCORD (Assemblage de composants par contrats en environnement ouvert et réparti). Juin 2002.
- N. Medvidovic and R. N. Taylor, A Classification and Comparison Framework for Software Architecture Description Languages, IEEE Transactions on Software Engineering, Vol 26, no1, pp70-93, Janvier 2000.
- Jean-Michel Doudoux. Ensemble des cours de Jean-Michel Doudoux sur la technologie Java (Java SE, Java EE, Java ME) et Eclipse (https://www.jmdoudoux.fr/java/dej/titre.htm). Version 2.30 du 15/06/2022
- A. TASSO. Le livre de Java premier langage- Avec 109 exercices corrigés. 592 pages. ISBN-13 : 978-2212138542. ISBN-10 : 2212138547. Éditeur : Eyrolles; 9e édition (31 octobre 2013). Langue: Français.
- J. LONCHAMP. Conception d'applications en Java/JEE - 2e édition. Principes, patterns et architectures. Collection : InfoSup, Dunod. Parution : janvier 2019.
- J. ARLAT. Composants logiciels et sûreté de fonctionnement - intégration de COTS. Juin 2000.
- J. Fontanel, F. Lacomme, L. Ren. Les web services - Concevoir et utiliser des applications 2.0 C#, Java, PHP, API JavaScript, Android SDK, iOS SDK. Éditeur:Ellipses. Avril 2013.

Semestre : 8
Unité d'enseignement : UEF1
Matière : Big data analytics
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Connaissances sur les différents types d'analyses de données massives (Big Data Analytics) que les organisations peuvent utiliser. Comment l'analyse de données massives peut aider, par exemple, à faire de la prédiction, à comprendre des pratiques, à opérer des analyses à grande échelle, etc.

Connaissances préalables recommandées :
Algèbre linéaire / concepts de base en apprentissage automatique Statistiques, machine learning, data mining, outils analytiques, bases de données

Contenu de la matière :

- Définir le concept de Big Data (Volume, Variété, Vélocité, etc.)
- Comprendre et illustrer la problématique de passage à l'échelle
- Comprendre le concept de valorisation de la donnée
- Comprendre les approches d'intégration de données massives
- Comprendre la problématique de traitement de données massives à des finsd'analyse distribuer (partitionner) des données sur plusieurs noeuds de calcul
- Comprendre et appliquer le paradigme Map/Reduce
- Configurer des systèmes Map/Reduce et déployer des applications
- Découvrir la palette d'outils d'analyse de données massives
- Identifier les risques et opportunités du Big Data pour les individus (vieprivée, protection des données personnelles…)
- Conduire un projet Big Data en entreprise (étapes projets, bonnes pratiques,schémas organisationnels et gouvernance des données)
- Situer les notions connexes sur la visualisation de données, les objetsconnectés, l'Open Data etc. vis-à-vis de l'analyse de données massives

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)
Références :

- Kleppmann, Martin. Designing data-intensive applications: The bigideas behind reliable,
- scalable, and maintainable systems. " O'Reilly Media, Inc.", 2017.
- Davenport, Thomas H., Paul Barth, and Randy Bean. How'bigdata'isdifferent. MIT Sloan
- Management Review, 2012.
- Marz, Nathan, and James Warren. Big Data: Principles and bestpractices of scalable realtime data systems. New York; ManningPublications Co., 2015.
- Reynolds, Vince. Big Data for Beginners: Understanding SMART BigData, Data Mining
- & Data Analytics for Improved Business Performance, Life Decisions &More!.
- CreateSpace Independent Publishing Platform, 2016.

Semestre : 8
Unité d'enseignement : UEM
Matière : Sécurité des Systèmes d'information
Crédits : 3
Coefficient : 3
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Permettre aux étudiants de maîtriser les concepts liés à l'ingénierie et au management de la sécurité des systèmes d'information et de participer à la mise en place pertinente d'une politique de sécurité au sein de l'entreprise.
Connaissances préalables recommandées :
La connaissance du module « Sécurité informatique » en S7
Contenu de la matière :
I. Introduction (2h)
1. Les défis du SI d'aujourd'hui
2. La culture « Sécurité » dans l'entreprise
II. Panorama des vulnérabilités, des menaces et des risques (4 h)
III. Acteurs de la sécurité (2h)
IV. Architecture de sécurité (1h)
V. Manager la sécurité (3h)
1. Maturité des entreprises vis-à-vis de la sécurité
2. Politique de sécurité
3. L'organisation de la sécurité et les ressources humaines affectées
VI.TD : Méthodes et référentiels de sécurité existants (15h)
1. Méthodes françaises
- La méthode EBIOS (Expression des besoins et Identification des Objectifs)
- La méthode MEHARI (Méthode Harmonisée d'Analyse de Risques)
- La méthode Marion
- La méthode Melissa
2. Autres méthodes
Allemande (www.bsi.de),
canadiennes (www.cse-cst.gc.ca),
américaines (www.ansi.org)
3. Normes internationales
- Série ISO 2700X
VII. RSSI : Rôles et moyens d'action (processus de sécurité, roadmap sécurité, externalisation...) (3h)

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références :

1. M.Bennasar, A.Champenois, P. Arnould et T.Rivat, « Manager la sécurité du SI, Planifier, déployer, contrôler, améliorer » , DUNOD, 2007.
2. B.Foray, La fonction RSSI, Guide des pratiques et retours d'expérience, DUNOD, 2007.
3. T.Harlé et F.Skrabacz, Clés pour la sécurité des SI,Hermès, 2004, www.clusif.asso.fr

Semestre : 8
Unité d'enseignement : UEM
Matière : Machine et Deep Learning
Crédits : 3
Coefficient : 3
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Les étudiants apprendront les différentes techniques et algorithmes d'apprentissage automatiquequ'il soit supervisé (régression, classification), non supervisé (clustering) ou par renforcement. Ilsapprendront la récente évolution accélérée par la puissance de calcul et la disponibilité de donnéesmassives qu'est le deeplearning et ses différentes architectures.
Connaissances préalables recommandées :
Programmation python, algèbre linéaire(opérations sur matrices, vecteurs), calcul différentiel (dérivées, dérivées partielles).

Contenu de la matière :

Chapitre 1 : Introduction au Machine Learning
- Types de machine learning (supervisé, non supervisé, par renforcement)
- Processus d'apprentissage
- Evaluation de modèles (mesures de performance, validation croisée)
- Algorithme KNN (K Nearest neighbor)
- Arbres de décision : Algorithme ID3

Chapitre 2 : Apprentissage supervisé et non supervisé
- Méthodes de régression (uni et multivariée) et descente du gradient
- Le perceptron
- Classification (régression logistique)
- Algorithme K-means

Chapitre 3 : Support Vector Machine (SVM)
- Classification SVM linéaire
- Classification SVM non linéaire
- Régression SVM

Chapitre 4 : Introduction au Deep Learning
- Rappel sur les réseaux de neurones : Propagation, rétro-propagation ;
- Réseau de neurones profond
- Propagation et rétro-propagation dans les réseaux de neurones profonds
- Paramètres et hyper-paramètres
- Optimisation des réseaux de neurones profonds
- Apprentissage par transfert et fine-tuning

Chapitre 5 : Apprentissage profond
- Présentation des réseaux de neurones convolutifs (CNNs)
- Apprentissage par transfert dans les CNNs
- Augmentation des données
- Résolution du problème de déséquilibre des données
- Limites des CNNs

Chapitre 6 : Modèles de séquence
- Architecture et optimisation des réseaux de neurones récurrents (RNNs)
- Encodage du texte dans les RNNs (Word Embedding, Word2Vec)
- Solutions au problème de disparition de gradient dans les RNNs:GatedCell (LSTM,bidirectional LSTM, GRU)
- Exemple d'un modèle de séquence Many-to-Many: Traduction automatique
- Limite des réseaux de neurones récurrents

● Mécanisme d'attention
● Apprendre le self-attention avec les réseaux de neurones
● Les Transformeurs

Chapitre 7 : Modèles génératifs profonds

Pourquoi les modèles génératifs ?
Présentation des auto-encodeurs
Réseaux antagonistes génératifs
Auto-encodeurs variationnels (VAEs)

Chapitre 8 : Apprentissage profond avancé

Introduction aux réseaux convolutifs pour graphe (GCN)
● Présentation des GCNs
● Exemples d'applications des GCNs Présentation des modèles de diffusion
● Principe du forward-noising et reverse-denoising
● Exemples d'applications des modèles de diffusion

Chapitre 9 : Apprentissage par renforcement

● Introduction à l'apprentissage par renforcement
● Introduction à la programmation dynamique, le cas discret
● Processus de décision markovien
● Algorithmes de programmation dynamique
● Algorithmes d'Apprentissage par renforcement
- Méthodes de Monte-Carlo,
- Algorithme différence temporelle
- Algorithme Q-Learning.

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références *:*

- Principles of Artificial Intelligence par J. Nilson
- Essentials of Artificial Intelligence par Morgan Kaufmann,
- Artificial Intelligence : A new synthesis par Morgan Kaufmann,
- ArtificialIntelligence: A Modern Approach par Stuart Russell et Peter Norvig
- Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep Learning. www.deeplearningbook.org
- Richard S. Sutton and Andrew G. Barto. Reinforcement Learning: An Introduction. Second Edition, MIT Press, Cambridge, MA, 2018
- François Chollet*, Deep Learning with Python, Ed* M A N N I N G , 2018SHELTER ISLAND
- Aurélien Géron, Hands-On Machine learning with scikit-Learn, Keras and tensorflow – Concepts, Tools and
techniques to build Intelligent systems, 2nd Ed., O.reilly, 2019.
- Aurélien Géron, Machine learning avec scikit-learn – Mise en oeuvre et cas concrets, 2ième ed., Dunod, 2019.

Semestre : 8
Unité d'enseignement : UET
Matière : Projet pluridisciplinaire
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

Ce cours se déroule durant le second semestre de la deuxième année. Il consiste en la conception et la réalisation d'un projet en informatique qui se déroule dans les mêmes conditions que celles d'une entreprise. Il est décrit à travers un cahier des charges précis il peut porter sur des thèmes très variés. Il est proposé par un ou plusieurs enseignants qui jouent le rôle de « client». Il est encadré par un enseignant qui peut être en même temps le « client».

Le groupe de projet, composé de 4 étudiants au minimum et de 6 au maximum, selon la taille du projet et sous la responsabilité d'un chef de projet nommé parmi les membres du groupe, doit se comporter comme une véritable équipe. Outre le contenu technique, qui consistera en l'application des connaissances acquises pour la mise en œuvre du cycle de développement d'un petit logiciel, l'accent sera mis sur l'acquisition et l'application des aspects organisationnels et relationnels entre les membres du groupe, de l'encadreur et du« client» :

- analyse et découpage du travail,
- répartition des charges de travail entre les membres du groupe par le chef de projet, circulation de l'information entre les membres du groupe,
- mise en place d'un planning de travail,
- exposé hebdomadaire de l'avancement du projet,
- délivrance des livrables fixés dans la fiche de projet,
- Rédaction d'un rapport final et exposé du travail réalisé.

Connaissances préalables recommandées :

Connaissances des modules antérieurs

Mode d'évaluation :

Contrôle continu (100%)

Références :

Semestre : 8
Unité d'enseignement : UED
Matière : Fondements de la Data Science
Crédits : 2
Coefficient : 2
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Ce cours offre aux étudiants une introduction succincte au domaine de la science des données. L'objectif principal est d'amener l'étudiant à comprendre et maîtriser les notions théoriques relatives à ce domaine et de lui donner une idée sur les types de problèmes à résoudre et une vue générale sur la démarche globale d'un projet data science. L'étudiant aura une idée sur les matières enseignées dans cette option, les outils et les approches qu'il aura à maitriser ainsi que des exemples de scénarios d'application. Les notions seront abordées sans rentrer dans les détails dans un but de compréhension générale.
Connaissances préalables recommandées :
Connaissances acquises durant les deux premières années du cursus de formation Ingénieur.
Contenu de la matière :
Chapitre 1 : Introduction à la Science des Données
- Définition et objectifs
- Historique de la discipline
- Cycle de vie d'un projet Data Science

Chapitre 2 : Les données
- Donnée – Information – Connaissance
- Les types de données (quantitatives – qualitatives)
- Les données structurées – Semi-structurées – non structurées
- Les différentes sources de données
- Acquisition- stockage et analyse des données
- Les formats d'échange de données
- Le cycle de vie de la donnée
- Hétérogénéité des données
- Intégration de données
- Visualisation des données

Chapitre 3 : Les disciplines intervenant dans la science des données
- Les bases de données (Relationnelles – Orientées-objets - .XML - NoSQL….)
- Entrepôts de données
- Big Data
- Analyse de données
- Intelligence Artificielle et Machine Learning
- Mathématiques appliquées
- Optimisation
- Outils de programmation

Chapitre 4 : Domaines d'application de la science de données et études de cas
- La santé
- L'industrie
- Le transport
- Le commerce électronique
- La grande distribution
- Les réseaux sociaux

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références:

- "Introduction to Data Science", Rick Mewis, Independently published (31 juillet 2023).
- "Data Science and Analysis Guide for Beginners", Sam Campbell , Independently published (21 janvier 2024).
- "Data Science for Beginners: Comprehensive Guide to Most Important Basics in Data Science", Alex Campbell, Independently published (12 janvier 2021).

Semestre : 9
Unité d'enseignement : UEF1
Matière : Interfaces Homme-Machine (IHM)
Crédits : 5
Coefficient : 5
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
- Introduire les concepts de base de l'interaction Homme-Machine et donner une vision complète de tous les aspects liés.
- Maîtriser la conception/la modélisation, le développement et l'évaluation des IHMs
- Acquérir les compétences nécessaires pour le développement des IHMs

Connaissances préalables recommandées :

Contenu de la matière :

I. Introduction à l'IHM (4h)
a. Historique et définitions
i. Pourquoi IHM ?
ii. Interface, Interaction, Communication et Dialogue
iii. Facteurs humains : ergonomie et psychologie cognitive
iv. Design : design industriel, arts graphiques, typographie
b. Évolution des IHMs
i. Navigation, indexation et annotation
ii. Interfaces graphiques
iii. What you see is what you get
iv. Interface multimodale
v. Informatique ubiquitaire

II. Ergonomie et sciences cognitives pour l'IHM (4h)
a. Notions de base sur les sciences cognitives
i. Le sous-système sensoriel
ii. Le sous-système cognitif
iii. Le sous-système moteur
iv. Les limites
v. L'approche cognitive pour les IHMs
vi. L'approche pour les utilisateurs
b. Ergonomie et guide de style
i. Définition
ii. Critères ergonomiques
1. Compatibilité
2. Guidage
3. Homogénéité
4. Souplesse
5. Contrôle explicite
6. Gestion des erreurs
7. Concision

III. Modélisation des IHMs (3h)
a. Les besoins de modélisation en IHM
b. Les modèles en IHM
i. Modèles d'architecture
ii. Modèles linguistiques
iii. Modèles à agent

iv. Modèles d'interaction

IV. Développement et évaluation des IHMs (4h)

a. Programmation événementielle

i. Définition

ii. Bibliothèque et environnements (JavaFX, QT, etc.)

b. Boîtes à Outils

i. Interface web (Ink, TCL/tk, etc.)

ii. Interface mobile (Android, iOS, etc.)

iii. Limitations

Établissement : ESI de Sidi Bel Abbes Intitulé de l'offre : Système d'Information et Web Page 74 c. Génération automatique des interfaces (MIMIC, GABI, etc.)

d. Évaluation des IHMs

i. Objectifs de l'évaluation

ii. Les outils de l'évaluation

iii. Limitations

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- Ludovie Cinquin, Erika Duriot, EricGroise, Olivier Mallassi, André Nedelcoux, David Rousselie, Vanessa Vimond « les dossiers de l'écran ; Utilisabilité et technologie IHM », Edition OCTO, Technologiue 2010
- J. Coutaz, « Interface Homme-Ordinateur » Dunod, 19990.
- N. Elouali, « Approche à base des modèles pour construction d'applications mobiles multimodales », PHD (2014).

Semestre 9
Unité d'enseignement : UEF1
Matière : Architectures et technologies blockchain
Crédits : 5
Coefficient : 5
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

Ce module vise principalement à fournir aux étudiants une compréhension des concepts, des technologies et des applications de la blockchain dans le but d'explorer ses implications à long terme pour les entreprises et sa relation avec d'autres technologies émergentes telle que l'IA. L'objectif est donc de permettre aux étudiants de proposer des solutions basées sur la blockchain pour des problèmes réels.

Connaissances préalables recommandées :

Programmation orienté objet, Systèmes distribués, Cryptographie

Contenu de la matière :

Chapitre 1 : introduction à la blockchain

1. Définition et concepts fondamentaux de la blockchain (Définitions, Caractéristiques, Composants clés, les types de réseaux Blockchain (Blockchains publiques, privées, permissionnées, hybrides, etc.).
2. Utilisations et cas d'usage réels (finance, supplychain, gouvernance, etc.)
3. Smart contracts: fonctionnement, langages de programmation (Solidity, Vyper, etc.)
4. Dapps (applications décentralisées) : architecture, développement, défis
5. DAOs (organisations décentralisées autonomes) : gouvernance, prise de décision collective

Chapitre 2 : Structure de données pour la blockchain

1. Blocs : en-tête, transactions, merkle-tree
2. Chaîne de blocs : structure de données, propriétés de sécurité
3. Transactions : structure, signatures numériques, UTXO vs comptes
4. Adresses : génération, gestion des clés publiques/privées

Chapitre 3 : fonctionnement de la blockchain

1. Emission d'une transaction
2. Validation de la transaction
3. Constitution d'un bloc candidat
4. Obtention d'un consensus décentralisé
5. Validation du nouveau bloc

Chapitre 4 : Protocoles de consensus

1. Preuve de travail (PoW) : fonctionnement, minage
2. Preuve d'enjeu (PoS) : différentes variantes (Casper, Ouroboros, etc.)
3. Autres protocoles : preuve d'autorité, preuve d'espace, etc.
4. Tolérance aux pannes byzantines et propriétés recherchées
5. Attaques contre le consensus : étude de cas et parades

Chapitre5 : Distributed Ledger Technology (DLT)

1. Définition
2. Différence entre blockchain et DLT.
3. Avantages de la technologie DLT.
4. Domaines d'application : finance, identités, IoT, etc.
5. Une comparaison des avantages/inconvénients par rapport aux blockchains

Chapitre 6 : Plateformes de blockchain

1. Blockchains publiques (Bitcoin : La première blockchain, focalisée sur la crypto-monnaie, Ethereum : Plateforme de contrats intelligents et dapps, protocole proof-of-work, Solana, Avalanche : Nouvelles blockchains hautes performances, Cardano, Polkadot : Prometteuses blockchains de 3e génération, Autres : Litecoin, Bitcoin Cash, etc.)

2. Blockchains d'entreprises permissionnées : (HyperledgerFabric (Linux Foundation) : Modulaire, adaptée aux consortiums, R3 Corda : Optimisée pour les applications financières, Quorum (JP Morgan) : Fourche privée d'Ethereum avec confidentialité
3. Distributed Ledgers non-blockchain (Hashgraph (Hedera) : Basée sur les graphes hashésgossiping, IOTA Tangle : DLT pour l'Internet des objets sans blocs, Holochain: Framework d'applications distribués pair-à-pair
4. Plateformes hybrides/interopérabilité (Cosmos : Écosystème de blockchains interconnectées, Polkadot : Protocole de transfert de données cross-chain, Ondulation: Passerelle de liquidité entre blockchains)

Chapitre 7: processus de conception des application blochckain,

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

1. "Building Decentralized Blockchain Applications: Learn How to Use Blockchain as a Backbone for Distributed Applications" par Shahid Shaikh
2. "Blockchain Basics: A Non-Technical Introduction in 25 Steps" par Daniel Drescher
3. "Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World" par Don Tapscott et Alex Tapscott

Semestre : S9
Contenu de la matière :
'enseignement : UEF2
Matière : DevOps et Cloud Computing
Crédits : 5
Coefficient : 5
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

- acquérir les fondamentaux liés au concept de la virtualisation et à son application dans le domaine du Cloud computing, à la gestion des infrastructures et au déploiement d'applications scalables.
- Identifier les principes fondamentaux de DevOps et sa fonction dans le développement de logiciels contemporains.
- Découvrir les technologies et les processus DevOps, y compris le contrôle de version, l'infrastructure en tant que code (IaC), l'intégration continue/livraison continue (CI/CD) et la conteneurisation.
- Découvrir comment effectuer des tâches d'analyse, de test, de déploiement et de surveillance du code.

Connaissances préalables recommandées

Contenu de la matière :

Chapitre 1 : Virtualisation (3 Séances)

1- Principes de la virtualisation
2- Objectifs et bénéfices
3- Historique
4- Définitions, concepts et techniques de virtualisation
   - les différentes architectures de la virtualisation de système d'exploitation
   - Virtualisation dans différents domaines
5- Performances et rendement

Chapitre 2 : Environnement Cloud Computing (3 Séances)

1- Rappel sur les systèmes distribués : Concepts et architectures, Concepts d'objet, composant, service, Architecture SOA
2- Définitions liées à l'environnement Cloud Computing
3- Caractéristiques
4- Modèles de déploiement
5- Modèles de services (SAAS, PAAS,IAAS)
6- Architectures du cloud
7- Entreprises face au cloud
8- Fournisseurs /clients de services
9- Environnement multi-cloud
10- Virtualisation et cloud

Chapitre 3 : Applications de Cloud computing (3 Séances)

1- Sécurité
2- Gestion
3- Stockage
4- Allocation de ressources, etc
1- Outils de mise en œuvre : Cloudsim, Openstack

Chapitre 4 : Introduction à DevOps (1 Séance)

• Qu'est-ce que DevOps ?

- Principes et pratiques DevOps.
- Avantages du DevOps : rapidité, qualité, collaboration
- DevOps vs développement de logiciels traditionnel

Chapitre 5 : Contrôle de version avec Git (2 Séances)

- Introduction à Git et aux systèmes de contrôle de versions distribués
- Commandes Git de base : cloner, ajouter, valider, pousser, tirer, créer une branche
- Workflows de collaboration avec Git : stratégies de branchement, pull request
- Utilisation des plateformes d'hébergement Git (par exemple, GitHub, GitLab)

Chapitre 6 : Intégration continue et livraison continue (CI/CD) (2 Séances)

- Comprendre les principes et les avantages du CI/CD
- Créer des pipelines CI/CD avec des outils comme Jenkins, GitLab CI/CD
- Tests automatisés et contrôles de qualité du code
- Stratégies de déploiement : bleu-vert, canari, mises à jour progressives

Chapitre 7 : Infrastructure as Code (IaC) (2 Séances)

- Gérer l'infrastructure avec du code : avantages et outils
- Introduction aux outils IaC : Terraform, Ansible, AWS CloudFormation
- Définir et provisionner les ressources de l'infrastructure
- Gestion et orchestration des configurations

Chapitre 8 : Conteneurisation avec Docker (4 Séances)

- Introduction à la conteneurisation et à Docker
- Créer et exécuter des conteneurs Docker
- Images Docker, registres et référentiels
- Orchestration de conteneurs avec Kubernetes (Concepts de base)

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

1. K. C. Gouda, AnuragPatro, Dines Dwivedi, NagarajBhat, Virtualization Approaches in Cloud Computing, in International Journal of Computer Trends and Technology (IJCTT), vol 12 Issues 4, (Juin 2014).
2. Tonglin Hawk, Ioan Raicu, and Lavanya Ramakrishnan. Scalable state management for scientific applications in the cloud. IEEE International Congress on Big Data, 2014.
3. Ali Belgacem, KaddaBeghdad Bey, Hassina Nacer, Sofiane Bouznad: Efficient dynamic resource allocation method for cloud computing environment. Clust. Comput. 23(4): 2871-2889 (2020)
4. KaddaBeghdad Bey, Sofiane Bouznad, Farid Benhammadi, Hassina Nacer: Improved Virus Optimization algorithm for two-objective tasks scheduling in Cloud Environment. FedCSIS (Communication Papers) 2019: 109-117
5. Linda Ouchaou, Hassina Nacer, Chahrazed Labba. Towards a distributed SaaS management system in a multi-cloud environment**.** Clust. Comput. 25(6): 4051-4071 (2022)
6. Guillaume Plouin "Cloud Computing et SaaS", Dunod,Paris, 1-ere _edition 2009, 2-eme édition 2012
7. Christopher M. Moyer, "Building Applications in the Cloud : Concepts, Patterns, and Projects" Addison-Wesley, 2011, (Pearson 2011, en français) pour les développeurs
8. Eric A. Marks, Bob Lozano "Executive's Guide to Cloud Computing", Wiley, 2010
9. Charles Babcock "Management Strategies for the Cloud Revolution", McGraw-Hill, 2010

Semestre : S9
Unité d'enseignement : UEF2
Matière : Internet of Things (IoT) : Concepts et développement
Crédits : 5
Coefficient : 5
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

L'objectif de ce cours est de transmettre à l'étudiant les concepts fondamentaux et technologies nécessaires pour comprendre, concevoir et développer une solution IoT (Internet des objets). A travers ce cours l'étudiant va apprendre à concevoir une solution IoT, fabriquer et programmer des objets IoT et l'intégrer dans un système IoT.

Connaissances préalables recommandées :

Systèmes d'exploitation, Réseaux, Programmation web, Python

Contenu de la matière :

Chapitre 1 : Introduction à l'Internet des objets

- 1.1 Définitions
  - 1.1.1 Internet des objets (IoT)
  - 1.1.2 Objets connectés
- 1.2 Caractéristiques d'IoT
- 1.3 IoT vs M2M
- 1.4 Domaines d'application d'IoT
- 1.5 IoTenabling technologies
  - 1.5.1 Réseaux de capteurs sans fil (WSN)
  - 1.5.2 Cloudcomputing
  - 1.5.3 Big data analytics
  - 1.5.4 Protocoles de communication
  - 1.5.5 Systèmes embarqués
    - 1.5.5.1 Définition et caractéristiques
    - 1.5.5.2 Modèles d'interaction avec ou sans contrainte de temps
    - 1.5.5.3 Approches de conception
    - 1.5.5.4 Co-conception (Codesign)
- 1.6 Challenges

Chapitre 2 : Architecture d'IoT et protocoles

- 2.1 Architectured'IoT
- 2.1.1 Architecture Three and Five layers
- 2.1.2 Architecture basée Cloud and Fog
- 2.2 Couche physique d'IoT
  - 2.2.1 Objets IoT (IoT devices)
  - 2.2.2 Architecture typique d'objet IoT
- 2.3 Protocoles de la couche liaison de données
  - 2.3.1 IEEE 802.3 - Ethernet
  - 2.3.2 IEEE 802.11 - WiFi
  - 2.3.3 IEEE 802.15.1 - Bluetooth
  - 2.3.4 IEEE 802.15.4 - LR-WPAN
  - 2.3.5 IEEE 802.16 - WiMax
  - 2.3.6 2G/3G/4G/5G - Communication mobile
  - 2.3.7 LoRaWAN
  - 2.3.8 Technologies de communication sans contact : RFID, NFC
- 2.4 Protocoles de la couche Réseaux/Internet
  - 2.4.1 IPv4

2.4.2 IPv6
2.4.3 6LoWPAN
2.5 Protocoles de la couche Transport
2.4.1 TCP
2.4.2 UDP
2.6 Protocoles de la couche application
2.6.1 HTTP
2.6.2 CoAP
2.6.3 WebSocket
2.6.4 MQTT
2.6.5 XMPP
2.6.6 DDS
2.6.7 AMQP
2.7 Conception logique d'IoT
2.7.1 Blocs fonctionnels d'IoT
2.7.2 Modèles de communication d'IoT
2.7.3 APIs de communication d'IoT

Chapitre 3 : Conception et développement d'un système IoT

3.1 Composants d'un système IoT
3.1.1 Objets IoT
3.1.2 Ressources
3.1.3 Contrôleur de service
3.1.4 Base de données
3.1.5 Web service
3.1.6 Composant d'analyse
3.1.7 Applications IoT
3.2 Méthodologie de conception d'un système IoT
3.2.1 Spécification des objectifs et exigences
3.2.2 Spécification du modèle de processus (cas d'utilisation)
3.2.3 Spécification du modèle de domaine (concepts, objets,...)
3.2.4 Spécification du modèle d'information
3.2.5 Spécification de services
3.2.6 Spécification des composants du système IoT
3.2.7 Spécification de la vue fonctionnelle
3.2.8 Spécification de la vue opérationnelle
3.2.9 Intégration des composants et objets
3.2.10 Développement d'application IoT
3.3 Etude de cas d'un système IoT

Chapitre 4 : Développement d'une solutions IoT

4.1 IoT and Python
4.1.1 Motivation
4.1.2 Packages Python pour IoT
4.2 Objets IoT (IoT physicaldevices)
4.2.1 Exemples d'objets IoT
4.2.2 Programmation d'objets avec Python
4.3 Serveurs IoT et cloud
4.3.1 Modèles de stockages cloud et APIs de communication
4.3.2 Web Application Messaging protocol (WAMP)
4.3.3 Xively cloud pour IoT

4.3.4 Développement d'applications avec le Framework Django (Python web application framework)
4.3.5 Développement des services web REST
4.4 Cas d'étude illustratifs

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- ArshdeepBahgaandVijayMadisetti. Internet of Things, A Hands on Approach, 2015.
- Pallavi Sethi1 and Smruti R. Sarangi. Internet of Things: Architectures, Protocols, and Applications. Journal of Electrical and Computer Engineering. 2017.
- P.P. Ray. A survey on Internet of Things architectures. King Saud University Journal of King Saud University –Computer and Information Sciences. 2016.
- Sridipta Misra Muthucumaruand Maheswaran Salman Hashmi. Security Challenges and Approaches in Internet of Things. SpringerBriefs in Electrical and Computer Engineering. 2017.
- Colin Dow . Internet of Things Programming Projects: Build modern IoT solutions with the Raspberry Pi 3 and Python. 2018.
- Olivier Hersent , David Boswarthick and Omar Elloumi. L'internet des Objets : Les principaux protocoles M2M et leur évolution vers IP. Edition DUNOD. 2014.

Semestre : S9
Unité d'enseignement : UEM
Matière : Business Intelligence
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :
Ce cours vise la découverte des différentes facettes du processus décisionnel et des modèles et outils associés et la maitrise de la technologie des Data Warehouses comme application d'un système décisionnel**.**

Connaissances préalables recommandées :
Bases de données, Système d'information
Contenu de la matière :
Introduction aux systèmes d'aide à la décision
- Concepts de base
- Historique
- Architectures décisionnelles

Modélisation multidimensionnelle :
- OLAP : concepts et règles
- Niveau conceptuel
- Niveau logique
- Niveau physique
- Méthode de conception multidimensionnelle

Intégration et administration
- Extraction
- Transformation
- Chargement

Exploitation des données décisionnelles
- Manipulation des données multidimensionnelles
- SQL analytique
- Tableaux de bord

Optimisation
- Les vues matérialisées
- Les index
- La fragmentation

Mode d'évaluation :
Contrôle continu (40%), Examen (60%)

Références:

- W. Inmon. *Building the Data Warehouse*. John Wiley& Sons Publishers, 2002
- Thomas Connolly, Carolyn Begg : Systèmes de bases de données : approche pratique de la conception, de l'implémentation et de l'administration, Eyrolles 2005
- R. Kimball, L. Reeves, M. Ross, W. Thornthwaite, *Concevoir et déployer un data warehouse*, Editions Eyrolles, (2000).
- Advanced Data Warehouse Design: From Conventional to Spatial and Temporal Applications (Data-Centric Systems and applications) by Elzbieta Malinowski, Esteban Zimányi.

Semestre : S9
Unité d'enseignement : UEM
Matière : Gestion de projets informatiques
Crédits : 4
Coefficient : 4
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

Les principaux objectifs de la matière « Management de projets informatiques » sont d'acquérir une vision globale des projets, la gestion et l'animation d'une équipe, les méthodes de gestion et de contrôle du projet, les différents logiciels de gestion de projet qui permettent de planifier les projets et les ressources tout en assurant le suivi de sa réalisation.

Connaissances préalables recommandées :

- Principes du génie logiciel
- Principes des systèmes d'information

Contenu de la matière :

- Introduction
- Méthodes traditionnelles vs. Méthodes agiles
- Planification du projet
- Estimation du coût de développement du projet (Modèle COCOMO)
- Pilotage et suivi du projet
- Adopter une méthode agile (Scrum)
- Les outils de gestion de projet.

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- CNRS, DSI (http://www.dsi.cnrs.fr/conduite-projet/Default.htm)
- Association Francophone de la gestion de projet (http://www.afitep.fr/Default.htm)
- Project Management Institute (PMI) (http://www.pmi.org/)
- Software Engineering Institute (SEI) (http://www.sei.cmu.edu/)
- IEEE Software Engineering Group (http://standards.ieee.org/software/)
- Guide to the Software Engineering Body of Knowledge (http://www.swebok.org/)
- Cost estimation tools • http://www.retisoft.com/SCEPFeatures.html
- http://www.construx.com/estimate
- FAQ on Function Points http://ourworld.compuserve.com/homepages/softcomp/fpfaq.htm
- Choosing a project management tool
- http://www.4pm.com/articles/selpmsw.html
- http://www.infogoal.com/pmc/pmcswr.htm
- Project management tools http://www.startwright.com/project1.htm

Semestre : S9
Unité d'enseignement : UED
Intitulé de la matière : Aspect Juridique pour les technologies de l'information
Crédits : 2
Coefficients : 2
Mode d'enseignement : Présentiel
Objectifs d'enseignement :

- Connaitre l'aspect juridique et la réglementation de la cybersécurité en Algérie
- **Connaissances préalables recommandées :**

Contenu de la matière :

- Rappels sur la Cybersécurité
- Géopolitique du Cyber Espace et Cyber Attaques
- Cybersécurité pour les Systèmes d'Information
- Cybersécurité à l'ère des réseaux sociaux
- Souveraineté Numérique et respect de la vie privée
- Politiques de Cybersécurité à travers le Monde et coopération internationale
- Politiques de Cybersécurité à travers le Monde et coopération internationale
- Stratégie de lutte contre Incidents Cybernétiques

Mode d'évaluation :

Contrôle continu (40%), Examen (60%)

Références :

- Compilation des décrets, arrêtés (Réglementations en vigueurs)