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	Réseaux

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	الشبكات

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 de la spécialité

1- Semestre 5 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff.	Crédits	Mode d'enseignement		Mode d'évaluation	
	14 sem	C	TD	TP	Autres			A distance	En présentiel	Continu	Examen
UE fondamentales											
UEF1(O/P)		**4 H 30**	**3 H 00**	**3 H 00**		**8**	**12**				
Système d'exploitation : (Synchronisation et communication)	**84 H 00**	3 H 00	1 H 30	1 H 30		4	6		**X**	**40**	**60**
Réseaux avancés 1	**63 H 00**	1 H 30	1 H 30	1 H 30		4	6		**X**	**40**	**60**
UEF2(O/P)		**3 H 00**	**3 H 00**	**3 H 00**		**6**	**10**				
Bases de données avancées	**63 H 00**	1 H 30	1 H 30	1 H 30		3	5		**X**	**40**	**60**
Compilation	**63 H 00**	1 H 30	1 H 30	1 H 30		3	5		**X**	**40**	**60**
UE méthodologie											
UEM1(O/P)		**3 H 00**	**1 H 30**	**1 H 30**		**4**	**6**				
IHM	**42 H 00**	1 H 30		1 H 30		2	3		**X**	**40**	**60**
Modélisation des systèmes d'information	**42 H 00**	1 H 30	1 H 30			2	3				
UE découverte											
UED1(O/P)		**1 H 30**		**1 H 30**		**2**	**2**				
Technologie Web	**42 H 00**	1 H 30		1 H 30		2	2		**X**	**40**	**60**
Total Semestre 5	**399H**	**12 H00**	**7 H30**	**9 H00**		**20**	**30**				

Volume Horaire : 28 H30
Nbr Modules : 7

2- Semestre 6 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff.	Crédits	Mode d'enseignement		Mode d'évaluation	
	14 sem	C	TD	TP	Autres			A distance	En présentiel	Continu	Examen
UE fondamentales											
UEF1(O/P)		**3 H 00**	**3 H 00**	**3 H 00**		**8**	**12**				
Systèmes distribués	**63 H 00**	1 H 30	1 H 30	1 H 30		4	6		**X**	**40**	**60**
Réseaux avancés 2	**63 H 00**	1 H 30	1 H 30	1 H 30		4	6		**X**	**40**	**60**
UEF2(O/P)		**3 H 00**	**3 H 00**	**3 H 00**		**6**	**10**				
Algorithmique et complexité avancées	**63 H 00**	1 H 30	1 H 30	1 H 30		3	5		**X**	**40**	**60**
Génie logiciel	**63 H 00**	1 H 30	1 H 30	1 H 30		3	5		**X**	**40**	**60**
UE méthodologie											
UEM1(O/P)		**3 H 00**	**3 H 00**	**3 H 00**		**5**	**6**				
Introduction à l'IA	**63 H 00**	1 H30	1 H 30	1 H 30		3	3		**X**	**40**	**60**
Analyse numérique	**63 H 00**	1 H 30	1 H 30	1 H 30		2	3		**X**	**40**	**60**
UE transversales											
UET1(O/P)		**1 H 30**				**1**	**2**				
Economie numérique et veille stratégique	**21 H 00**	1 H 30				1	2	**X**	**X**		**100**
Total Semestre 6	**399 H**	**10 H30**	**9 H00**	**9 H00**		**20**	**30**				

Volume Horaire : 28 H30
Nbr Modules : 7

3- Semestre 7 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff.	Crédits	Mode d'enseignement		Mode d'évaluation	
	14 sem	C	TD	TP	Autres			A distance	En présentiel	Continu	Examen
UE fondamentales											
UEF1(O/P)		**3 H 00**	**1 H 30**	**6 H00**		**7**	**11**				
Administration et gestion des systèmes et des réseaux	**84 H 00**	1 H 30	1 H 30	3 H 00		3	5		**X**	**40**	**60**
Virtualisation des réseaux et des systèmes	**63 H 00**	1 H 30		3 H 00		4	6		**X**	**40**	**60**
UEF2(O/P)		**3 H 00**	**3 H 00**	**3 H 00**		**7**	**11**				
Cryptographie	**63 H 00**	1 H 30	1 H 30	1 H 30		4	6		**X**	**40**	**60**
Modélisation et évaluation des performances des réseaux	**63 H 00**	1 H 30	1 H 30	1 H 30		3	5		**X**	**40**	**60**
UE méthodologie											
UEM(O/P)		**1 H 30**	**1 H 30**	**1 H 30**		**3**	**3**				
Apprentissage automatique	**63 H 00**	1 H 30	1 H 30	1 H 30		3	3		**X**	**40**	**60**
UE découverte											
UED1(O/P)		**1 H 30**	**1 H 30**			**2**	**3**				
Traitement du signal	**42 H 00**	1 H30	1 H30			2	3		**X**	**40**	**60**
UE transversale											
UET1(O/P)		**1 H 30**				**1**	**2**				
Soft-Skills : Amélioration des compétences comportementales	**21 H 00**	1 H 30				1	2	**X**	**X**		**100**
Total Semestre 7	**399 H**	**10 H30**	**7 H 30**	**10 H30**		**20**	**30**				

Volume Horaire : 28 h30
Nbr Modules : 7

4- Semestre 8 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff.	Crédits	Mode d'enseignement		Mode d'évaluation	
	14-16sem	C	TD	TP	Autres			A distance	En présentiel	Continu	Examen
UE fondamentales											
UEF1(O/P)		**4H 30**	**3H**	**4H 30**		**5**	**8**				
Cloud Computing	**84H00**	1H30	1H30	3H00		3	4		**X**	**40**	**60**
Technologies des réseaux sans fil	**84H00**	3H 00	1H30	1H30		2	4		**X**	**40**	**60**
UEF2(O/P)		**4H 30**	**3H**	**1H 30**		**5**	**8**				
Introduction à la sécurité des réseaux et systèmes	**63H00**	1H 30	1H30	1H30		3	4		**X**	**40**	**60**
Techniques d'optimisation pour les réseaux	**63H00**	3H 00	1H30			2	4		**X**	**40**	**60**
UE méthodologie											
UEM1(O/P)		**3H 00**		**6H 00**		**8**	**10**				
Développement mobile	**42H00**	1H30		1H 30		2	3		**X**	**40**	**60**
Système embarqués	**42H00**	1H30		1H30		2	3	**X**		**100**	
Projet pluridisciplinaire	**42H00**			3H00		4	4	**X**	**X**		**100**
UE transversales											
UET1(O/P)		**1H 30**		**1H 30**		**2**	**4**				
Lean Startup	42H00	01H30		01H30		2	2	X	X	80	20
Total Semestre 8	**462H**	**13H30**	**6H 00**	**13H30**		**20**	**30**				

Volume Horaire : 28 h 30
Nbr Modules : 7

5- Semestre 9 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff.	Crédits	Mode d'enseignement		Mode d'évaluation	
	14 sem	C	TD	TP	Autres			A distance	En présentiel	Continu	Examen
UE fondamentales											
UEF1(O/P)		**3 H 00**	**1 H 30**	**4 H 30**		**8**	**10**				
Internet d'objets (IoT) et ses applications	**63 H 00**	1 H 30		3 H00		4	5		**X**	**40**	**60**
Sécurité des systèmes et réseaux	**63 H 00**	1 H 30	1 H 30	1 H 30		4	5		**X**	**40**	**60**
UEF2(O/P)		**3 H 00**	**1 H 30**	**3 H 00**		**6**	**10**				
Multimédia et QoS	**63 H 00**	1 H 30	1 H 30	1 H 30		3	5		**X**	**40**	**60**
Architectures micro-services	**42 H 00**	1 H 30		1 H 30		3	5				
UE méthodologie											
UEM1(O/P)		**3 H 00**	**1 H 30**	**3 H 00**		**5**	**8**				
Gestion de projets (GP)	**63 H 00**	1 H 30	1 H 30	1 H 30		3	4	**X**	**X**	**40**	**60**
Business Intelligence	**42 H 00**	1 H 30		1 H 30		2	4		**X**	**40**	**60**
UE découverte											
UET1(O/P)		**1 H 30**				**1**	**2**				
Aspect Juridique pour les technologies de l'information	**21 H 00**	1h30				1	2	**X**	**X**		**100**
Total Semestre 9	**357 H**	**10 H30**	**4 H30**	**10 H30**		**20**	**30**				

Volume Horaire : 25 h30
Nbr Modules : 7

6- Semestre 10 :

Domaine : Mathématiques et Informatique
Filière : Informatique
Spécialité : Réseaux

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

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

5- 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	UED	UET	Total
Cours	34 H 30	13 H 30	3 H	4 H 30	55 H 30
TD	25 H 30	7 H 30	1 H 30	0	34 H 30
TP	33 H	12	1 H 30	3	49.5
Travail personnel					
Autre (PFE S10)	30				30
Total	123 H	33	6	7 H 30	169 h 30
Crédits	137	30	5	8	**180**
% en crédits pour chaque UE	72.57 %	19.47 %	3.54 %	4.42 %	100%

III - Programme détaillé par matière

Semestre : *S5*
Unité d'enseignement : UEF1

Matière : Système d'Exploitation : Synchronisation et Communication

Crédits : 6

Coefficient : 4

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Introduire la problématique du parallélisme dans les systèmes d'exploitation, étudier la mise en œuvre de la coopération à travers les outils de synchronisation et de communication et les problèmes associés.

Recommandation : Illustrer les différents concepts à l'aide d'outils (création de processus, signaux, sémaphores, mémoire partagée, files de messages, tubes) du système UNIX notamment dans les séances de TP.

Connaissances préalables recommandées (*les deux modules : introduction aux systèmes d'exploitation 1 et 2).*

Contenu de la matière :

Chapitre 1 : NOTION DE PARALLELISME/CONCURRENCE (10 %)

- Systèmes de tâches
- Parallélisme et langages évolués
- Déterminisme d'un système de tâches
- Parallélisme maximal

Chapitre 2 : SYNCHRONISATION DES PROCESSUS (20 %)

- Partage de ressources et des données
- Exclusion mutuelle
- Mécanismes d'exclusion mutuelle : variables d'états, solutions matérielles, les sémaphores, autres mécanismes de synchronisation

Chapitre 3 : COMMUNICATION ENTRE PROCESSUS (15 %)

- Communication par partage de variables : Modèle du producteur/ consommateur, modèle des lecteurs/ rédacteurs.
- Communication par échanges de messages : Désignation (directe : CSP, indirecte : les boites aux lettres)

Chapitre 4: MONITEURS ET REGIONS CRITIQUES (15%)

- Les moniteurs : Structure, implémentation, illustrations, variantes de moniteurs
- Régions critiques (Simples, Conditionnelles)

Chapitre 5: INTERBLOCAGE (20%)

- Caractéristiques, Modèles, Formalisation, Représentations graphiques
- Traitement de l'interblocage : Prévention, Détection- Guérison, Evitement

Chapitre 6: SYSTEMES DE FICHIERS (20%)

- Interface d'un système de fichiers : Attributs d'un fichier, opérations, types, structures, méthodes d'accès, répertoires.
- Implémentation d'un système de fichiers : Structure, organisation physique des fichiers et répertoires, implémentation, gestion des fichiers actifs et de l'espace libre, partage de fichiers, protection.
- Systèmes de fichiers multimédias
    - Types, structures et compression de fichiers multimédias
    - Opérations sur les fichiers multimédias
    - Organisation des fichiers multimédias

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

Références bibliographiques

- Thomas Anderson and Michael Dahlin Operating Systems: Principles and Practice (Second Edition) Volume II: Concurrency ISBN : 978-0-9856735-4-3 Publisher: Recursive Books, Ltd.,. 2011-2015.
- M. J. Bach, traduit par G. Fellah, ''Conception du Système UNIX,'' Masson et Prentice Hall,1990.
- J. Beauquier, B. Berard,'' Systèmes d'exploitation : concepts et algorithmes,'' McGraw-Hill, 1990.
- Crocus,'' Systèmes d'exploitation des ordinateurs'', Dunod 1975.
- Patrick Cegielski, Conception de systèmes d'exploitation : le cas linux, Éditions Eyrolles, 16 Octobre 2003.
- N. B. Fontaine, P. Hammes, ''UNIX système V: Système et environnement'' Masson 1989.
- S. Krakowiak, ''Principes des systèmes d'exploitation des ordinateurs,'' Dunod 1987.
- J-L.Peterson, A.Silbershartz ''Operating Systems Concepts,''Addison-Wesly Publishing Company,Inc, 1983.
- Silberschatz, P. B. Galvin '' Principes des systèmes d'exploitation,'' 4 e Edition, Addison Wesley, 1994.
- William Stallings, Operating Systems: Internals and Design Principles (8th Edition), Publisher : Pearson; 8th edition (February 2, 2014).
- Tanenbaum, '' Modern Operating Systems, '' Third Edition Prentice Hall, 2009.
- Tananbaun, ''Systèmes d'exploitation,'' 3°édition, Pearson Edition. 2008.
- J-P. Verjus et al,''Synchronisation des programmes parallèles : Expression et mise en oeuvre dans les systèmes centralisés, édition 1983 Dunod.

Semestre : S5

Unité d'enseignement : UEF1

Matière : Réseaux avancés 1

Crédits :6

Coefficient :5

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Cette matière présente aux étudiants des aspects avancés sur les réseaux de communication selon l'approche top-down. Le contenu de la matière se focalise sur les couches supérieures : Application, Transport, Réseaux (IP et routage) où les protocoles réseaux de ces trois niveaux sont à étudier de façon approfondie.

Connaissances préalables recommandées : Notions de bases liées aux réseaux informatiques.

Contenu de la matière :

Chapitre 1 : Paradigmes de communication réseau

- Client/Serveur
- Peer-to-Peer
- Publish/Subscribe …

Chapitre 2 : Couche Application

- Les protocoles HTTP (HyperText Transfer Protocol) et HTTP streaming,
- Le protocole FTP (File transfer protocol) et TFTP (Trivial File Transfer Protocol),
- Le protocole SIP (Session Initiation Protocol)
- Le protocole SMTP (Simple Mail Transfer Protocol)
- Le protocole DNS (Domain Name Server)
- Les protocoles CoAP (Constrained Application Protocol) et MQTT (Message Queuing Telemetry Transport)

Chapitre 3 : Applications réseaux en vedette

- VoIP (Voice over IP) / RoIP (Radio over IP)
- Messagerie instantanée et Messagerie basée sur le model Publish/subscribe.

Chapitre 4 : Couche Transport

- Techniques de multiplexage et démultiplexage.
- Les protocoles TCP (Transmission Control Protocol), et UDP (User Datagram Protocol)
- Le protocole RTP (Real-time Transport Protocol)
- Le protocole SCTP (Stream Control Transmission Protocol)

Chapitre 5 : Couche Réseau

- Rappel sur IPv4
- Le routage dans les réseaux IP :
  - Routage statique et dynamique
  - Protocoles à vecteur de distance : RIP, EIGRP
  - Protocoles à état de lien : OSPF, IS-IS
  - Routage inter AS (Autonomous system) : BGP
  - Routage multicast
- Les protocoles IPV6 et ICMPv6

- La gestion de la coexistence des réseaux IPv4 et IPv6
    - La technique de tunneling (encapsulation entre les protocoles IPv6 et IPv4)
    - La coexistence basée sur les passerelles.
- La mobilité dans les réseaux IP avec le protocole MIP (Mobile IP).

Mode d'évaluation : Examen 60% + Contrôle Continu 40%

Références *(Livres et polycopiés, sites internet, etc).*

- Kurose, J. F., & Ross, K. W. (2021). Computer Networking A Top-Down Approach. Peasron Editions. 2021.
- Bonaventure, O., Networking: Principles, Protocols and Practice, 3rd Edition, 2021.
- Lannone, E., Telecommunication networks, O'Reilly Edition, 2017.
- James Aweya, « IP Routing Protocols: Link-State and Path-Vector Routing Protocols », 1er Édition, Format Kindle, 2021c
- Rick Graziani, « IPv6 Fundamentals: A Straightforward Approach to Understanding IPv6 »2e Édition, Format Kindle, 2017

Semestre : S5

Unité d'enseignement : UEF2

Matière : Bases de données avancées

Crédits : 5

Coefficient : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Le module a pour objectif de présenter les notions avancées en BD.

Connaissances préalables recommandées

Les notions approfondies en « base de données ».

Contenu de la matière :

Chapitre 1 PL-SQL

- Identificateurs (variable, curseur, exception, etc.)
- Les types de variables (Scalaires, Composites, Référence, LOB)
- Les fonctions SQL en PL/SQL (fonctions sur les nombres, sur les chaînes de caractères, de conversion de type de données, sur les dates, etc)
- Structure de contrôle dans PL/SQL (structures conditionnelles, Les boucles, etc)
- Les instructions SQL dans PL-SQL (select, insert, delete, update)
- Les curseurs (implicites et explicites, les curseurs et les records, les curseurs et la boucle FOR, etc)
- Les exceptions (prédéfinies, non-prédéfinies, utilisateurs)
- Les sous programmes (procédures/fonctions déclarées/stockées)
- Les packages (spécification, implémentation)
- Les triggers (ECA, before, after, for each row, when, inserting, deleting, updating, etc)

Chapitre 2 Les bases de données Objet-Relationnel

- Passage d'un schéma relationnel à un schéma Objet-relationnel (exemple de motivation)
- Les types objet dans les bases de données (création, suppression, modification, instanciation, etc)
- Gestion de la valeur null (Objet à NULL ou ses valeurs à NULL)
- Tables objets (création, instanciation, mise-à-jour, suppression, interrogation, etc)
- Gestion des références (utilisation des références pour l'insertion et la modification, les références perdues, etc)
- Tables imbriquées (création, insertion, modification, suppression, interrogation, etc)
- Tableaux pré-dimensionnés (VARRAY) (création, insertion, modification, suppression, interrogation, etc)

Chapitre 3 : Optimisation des bases de données

- Introduction à l'optimisation du schéma interne (Schéma interne et performances des applications, Évaluation des besoins de performance, Indexation, Dénormalisation, etc)
- Mécanismes d'optimisation des moteurs de requêtes (Calcul du coût d'une requête, Principe de l'optimisation de requêtes, Techniques pour l'optimisation de requêtes, Collecte de

statistiques pour le moteur d'optimisation, Jointures et optimisation de requêtes
- Analyse et optimisation manuelle des requêtes (Analyse de coûts de requêtes (EXPLAIN), Optimisation manuelle des requêtes)

Chapitre 4 : Administration de bases de données
- Structure physique d'une base de données : fichiers de données (Data Files), fichiers de reprise (Redo Log Files), fichiers de contrôle (Control Files)
- Structure logique d'une base de données : les tablespaces, de segments, les extensions (extents), les blocs, ...
- Sauvegarde et restauration d'une base de données : les différents modes de sauvegarde, les différents modes restauration,
- Administration d'une BD d'Oracle : Création, modification et suppression de la BD, Démarrage et Arrêt de la BD, ...

Chapitre 5 : Les bases de données XML
- Présentation et syntaxe de XML (notion d'élément, d'attribut, document bien formé, etc)
- Un document XML valide (DTD, XML Schéma)
- Interrogation de document XML (XPath, XQuery, etc)
- Présentation des données XML (XSL, CSS, XSLT, etc)

Mode d'évaluation : 40% contrôle continu 60% examen écrit

Références bibliographiques

1. Bases de données Architectures, modèles relationnels et objets, SQL3, Serge Miranda, InfoPro, Dunod/01 Informatique, 2002.
2. Bases de données, Georges Gardarin , Eyrolles, 2003.

Semestre : S5

Unité d'enseignement : UEF2

Matière : Compilation

Crédits : 5

Coefficient : 3

Mode d'enseignement : Présentiel

Connaissances préalables recommandées (*descriptif succinct des connaissances requises pour pouvoir suivre cet enseignement – Maximum 2 lignes).*

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/Automates et 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 bibliographiques

- 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 : UEM1

Matière : Interface Homme-machine

Crédits : 3

Coefficient : 2

Mode d'enseignement : Présentiel

Objectifs de l'enseignement : permettre aux étudiants d'acquérir des compétences pour confectionner des interfaces graphiques visuelles en respectant les critères ergonomiques et les standards du design des interfaces interactives et conviviales.

- Connaissances des règles ergonomiques
- Connaissance d'une méthode de développent d'IHM
- Couplage avec la méthode de développement par objets
- Mise en œuvre de ces méthodes dans un projet

Connaissances préalables recommandées : Algorithmique et structure de données, génie logiciel

Contenu de la matière :

Chapitre I : Notions d'interaction

I.1 - Définitions : Interaction, Interactivité, ...
I.2 – Causes de rejet de certaines applications.
I.3- Enjeux : gisements d'économie, mutation du poste de travail, conséquences d'une interface
négligée,
I.4- Difficultés : la variété des utilisateurs, fabrication difficile, lien entre concepteur et développeur
I.5- Définition d'une IHM.
I.6- Historique des IHMs.

Chapitre II : Méthodologie de construction d'une IHM

II.1- Méthodologie Classique**.**
II.2- Etape d'identification : identification des domaines fonctionnels, Définition du modèle de l'utilisateur (notion de profil de l'utilisateur), Définition du modèle des taches (types des tâches) et environnement technique.
II.3- Etape d'analyse des taches (notion de séquence actions-objectifs).
II.4- Etape de modélisation (nécessite de choisir un modèle et une architecture). II**.**5- Etape de spécification (cahier de charges)

- Etude des besoins pour l'IHM
- Spécification conceptuelle
- Spécification fonctionnelle
- Spécification syntaxique
- Spécification lexicale

Chapitre III : Modèles & architectures

III.1- Le Contrôleur de dialogue (définition & rôle).
III.1- Présentation du modèle Seeheim
III.2- Présentation du modèle PAC III.3- Présentation du modèle MVC
III.4- Présentation des modèles à agents.

Chapitre IV : Régles ergonomiques dans les IHMs

IV.1- heuristiques de Nielsen.
IV.2- critères ergonomiques de Bastien et Scapin
IV.3- règles d'or de Coutaz

Chapitre V : Conception d'interfaces multi utilisateurs

V.1- Etude comparative entre IHM mono utilisateur et multi utilisateur.
V.2- La méthode CCU (conception centrée sur l'utilisateur).
V.3- Exemples d'interfaces multi utilisateurs.

Chapitre VI : Interfaces adaptativesLe Modèle de Vaudry.

- Etude d'un exemple : Modèle à agents.

Chapitre VII : Les interfaces multimodales et les interfaces futures

- Techniques d'interactions avancées, Réalité Augmentée, Interface Tangible, projection 3D, Analyse du mouvement)
- Elements de Programmation Visuelle.

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

Références:

- Ménadier Jean-Paul, l'interface utilisateur : Pour une informatique conviviale, DUNOD, Informatique et Stratégie, 1991
- Coutaz Joelle, Interface homme-ordinateur : conception et réalisation Dunod-Informatique 1990
- Kolski, C, Ezzedine, H et Abed, M, « *Développement du logiciel : des cycles classiques aux cycles enrichis sous l'angle des IHM* », ouvrage collectif, Analyse et conception de l'IHM, Interaction homme-machine pour les systèmes d'information Vol 1, Hermès, 2001, 250 p, ISBN 2-7462-0239-5, p. 23-49.
- Drouin, A, Valentin, A et Vanderdonckt, J, « *Les apports de l'ergonomie à l'analyse et à la conception des systèmes d'information* », in Christophe KOLSKI, (ed.), Analyse et conception de l'IHM, Interaction homme-machine pour les systèmes d'information Vol 1, Hermès, 2001, 250 p, ISBN 2- 7462-0239-5, p. 51-83.
- David Benyon, Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design,Pearson; 3 edition, 2013
- Yvonne Rogers, Helen Sharp & Jenny Preece, Interaction Design: beyond human- computerinteraction (3rd edition), Wiley, 2011
- Norman DA, The Design of Everyday Things, Basic Books, 2002. Serengul Smith-Atakan The FastTrack to Human-Computer Interaction, (Paperback) Thomson Learning, 2006.

Semestre : *S5*

Unité d'enseignement : UEM1

Matière : Modélisation des systèmes d'information

Crédits : 3

Coefficient : 2

Mode d'enseignement : Présentiel

Objectifs de l'enseignement :

Ce cours vise à inculquer à l'étudiant une démarche générique à base du langage UML pour l'analyse et conception des systèmes d'information (y compris les SIs d'entreprise et les Sis orientés web)

A travers cette démarche l'étudiant apprendra : i) à mettre en point un cahier de charge, ii) à analyser et concevoir un système d'information (informatisé) iii) développer et déployer le logiciel adéquat.

Connaissances préalables recommandées : Algorithmique et structure de données, génie logiciel

Contenu de la matière :

I. Introduction

II. Systèmes Organisationnel et Système de gestion d'entreprise

- Entreprise et ses Fonctions
  - o Définition Entreprise
  - o Grandes Fonctions Entreprise
  - o Système organisationnel
  - o Structures et Organisation
  - o Théorie d'organisation

III. Ingénierie des besoins d'un système d'information

- Notions de besoins (Exigences)
  - o Définitions
- Exigences fonctionnelles et non fonctionnelles
  - o Cahier des charges : Le document définissant les exigences logicielles
- Spécification des besoins
  - o Façon d'écrire des spécifications des besoins
  - o Spécification Structurée
  - o Spécification formatée
- Processus d'ingénierie des besoins
  - o Elicitation des exigences
  - o Analyse des exigences
  - o Validation des exigences
  - o Management des exigences
- Etude de cas

IV. Analyse et modélisation d'un système d'information

- Modèles contextuels
- Rappel sur le langage UML
- Modèles d'interaction
  - o Modélisation des cas d'utilisation
  - o Diagramme de séquence
- Modèles Structurels
  - o Diagramme de Classes

- Diagramme d'Objets
- Modèles comportementaux (dynamiques)
  - Diagramme d'Etat et de Transition
  - Diagramme d'activité
- Langage OCL
  - Topologie des Contraintes
  - Types et opérations utilisables dans les expressions OCL
- Etude de Cas

V. Architecture et Conception d'un Système d'information (8h)

- Rappel sur les styles architecturaux
- Modéliser l'architecture avec UML
  - Vue Logique : diagramme de Paquetages
  - Vue Réalisation: diagramme de composants
  - Vue de déploiement : Diagramme de déploiement
- Méthode de Conception Générique (UP/XP)
  - Maquette IHM et Diagramme d'activité de navigation
  - Diagramme de classes participantes (Classes de : dialogue, contrôle et entité)
  - Diagramme de classes de conception
    - Concevoir les algorithmes pour implémenter les opérations
    - Affectation des opérations aux classes correspondantes
    - Optimiser les chemins d'accès aux données
  - Etude de cas
- Conception de systèmes d'information orientés Web (ou Applications Web) :
  - Qualité de systèmes d'information orientés web
  - Buts de conception
  - Pyramide de conception :
    - Conception de l'interface
    - Conception de de l'aspect l'esthétique
    - Conception du contenu
    - Conception de la navigation
    - Conception de l'architecture
    - Conception de composant
  - Etude de Cas

VI. Implantation et déploiement (2h)

- Réutilisation
- Gestion de la configuration
- Développement sur les machines hôte et cible
- Plateformes de développement de logiciels

.

VII. Maintenance et Evolution de Logiciel (2h)

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

BIBLIOGRAPHIE

- Fowler, Martin. UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition), Addison-Wesley, 2003.
- Jackson, Michael. Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices, Addison-Wesley 1995.
- Muller, Pierre-Alain & Gaertner, Nathalie. Modélisation objet avec UML (2e édition), Eyrolles, 2003.
- Hainaut, Jean-Luc. Bases de données : Concepts, utilisation et développement, Dunod, 2ème Edition, 2012

- Warmer, Jos & Kleppe, Anneke. The Object Constraint Language: Getting Your Models Ready for MDA, 2003, Addison-Wesley Professional
- Harel, David & Politi, Michal. Modeling Reactive Systems with Statecharts, the STATEMATE Approach. McGraw-Hill, 1996.
- Milner, Robin. Communication and Concurrency. Prentice Hall, 1989.
- Micskei, Zoltán & Waeselynck, Hélène. The Many Meanings of UML 2 Sequence Diagrams: A Survey. Software & Systems Modeling, 10(4), 489-514. 2011.
- Object Management Group website : http://www.omg.org/spec/
- UML Diagrams : http://www.uml-diagrams.org

Semestre : S5

Unité d'enseignement :UED1

Matière : Technologies Web

Crédits :2

Coefficient :2

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Les objectifs principaux : Comprendre les fondements des technologies web, notamment les langages HTML, CSS et JavaScript, et leur fonctionnement ensemble pour créer des pages web interactives et dynamiques, apprendre les bonnes pratiques de codage pour développer des pages web performantes, sécurisées et accessibles à tous les utilisateurs, comprendre les concepts de base de la programmation côté serveur, y compris les bases de données et les langages de programmation tels que PHP et SQL et apprendre les technologies émergentes, telles que les frameworks de développement web tels que Angular, React et Vue, pour créer des pages web plus rapidement et efficacement

Connaissances préalables

Algorithmique

Contenu de la matière :

- Introduction aux technologies web
  - Introduction aux technologies web
  - Comprendre le fonctionnement d'un site web
  - Les différentes technologies web et leur rôle
- HTML
  - Comprendre la structure d'une page web en HTML
  - Utilisation des balises HTML courantes
  - Utilisation des balises HTML avancées
  - Les formulaires HTML
- CSS
  - Comprendre la syntaxe CSS
  - Appliquer des styles à un document HTML
  - Utilisation de sélecteurs
  - Utilisation de sélecteurs CSS avancés
  - Utilisation de feuilles de style CSS externes
- JavaScript
  - Comprendre la syntaxe JavaScript
  - Les variables
  - les conditions
  - les boucles
  - les tableaux
  - Les fonctions
  - Manipulation du DOM en JavaScript
  - Gestion d'événements
  - o
- AJAX
  - Introduction à AJAX

    - Utilisation de l'objet XMLHttpRequest en JavaScript
    - Utilisation de jQuery pour manipuler le DOM et ajouter des effets
- jQuery
    - Utilisation des sélecteurs jQuery
    - Manipulation du DOM avec jQuery
    - Utilisation de XMLHttpRequest ; $.Post ; $.GET
    - Gestion des erreurs et des réponses
    - Utilisation d'AJAX pour la mise à jour dynamique d'une page web
    - Utilisation d'AJAX pour la mise à jour dynamique d'une page web sans rechargement de la page entière
    - Les événements
    - Les animations
- Bootstrap
    - La structure d'un site web Bootstrap
    - Création d'un design responsive avec Bootstrap
    - o
- PHP
    - Comprendre la syntaxe PHP
    - Les variables
    - les conditions
    - les boucles
    - les tableaux
    - Les fonctions
    - Utilisation de PHP pour interagir avec une base de données avec PDO
- Mysql
    - Présentation de la base de données MySQL
    - Utilisation de MySQL pour stocker des données utilisateur
- Node.js
    - Introduction à Node.js
    - Utilisation de Node.js pour le développement web
    - Utilisation de npm pour installer des packages
- Express.js
    - Introduction à Express.js
    - Utilisation de routes et de middleware
    - Gestion des formulaires avec Express.js
- React
    - Introduction à React
    - Composants React et props
    - État et cycle de vie des composants
    - D'autres framework Présentation du framework Vue.js et Angular.js
- Introduction à MongoDB
    - Utilisation de MongoDB pour stocker des données
    - Utilisation de Mongoose pour interagir avec MongoDB
- GraphQL
    - Introduction à GraphQL
    - Création d'un schéma GraphQL
    - Utilisation de GraphQL avec Express.js
- XML
    - Présentation de la technologie XML
    - Utilisation de XML pour la création
    - Présentation de la syntaxe XML et de ses avantages par rapport à d'autres formats de données

    - Utilisation de XML pour l'échange de données entre différents systèmes
- API REST
    - Présentation de l'API REST
    - Utilisation de l'API REST pour l'échange de données entre différents systèmes
- Développement Backend avec Django
    - Introduction à Django
    - Architecture de Django : modèles, vues et templates
    - Utilisation de l'ORM de Django pour interagir avec la base de données
    - URLs dynamiques : utilisation des expressions régulières pour gérer les URLs
    - Système de templates de Django : utilisation des templates pour générer des pages HTML dynamiques
    - Vues : création de vues pour gérer les requêtes HTTP
    - Administration automatique : utilisation de l'interface d'administration de Django pour gérer les données
    - Développement d'API RESTFul avec Django Rest Framework

Mode d'évaluation : (type d'évaluation et pondération) 40% contrôle continu 60% examen écrit)

Références bibliographiques

- Anik Acharjee Apprendre les technologies du web de zéro: Expérience pratique sur les technologies du web (French Edition) Editions Notre Savoir 2020
- Roxane Anquetil Les bases du web : Html5, Css3, JavaScript , 2019

Cyprien Accard Apprendre le développement Web au lycée - avec les langages HTML5, CSS3, JavaScript, PHP et MySQL – 2018

Semestre : S6

Unité d'enseignement :UEF1

Matière : Systèmes distribués

Crédits :6

Coefficient :4

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Ce cours a pour objectif de fournir aux étudiants une compréhension approfondie des concepts, des architectures, des protocoles et des algorithmes utilisés dans les systèmes distribués. Les étudiants seront exposés à la conception et à l'implémentation de systèmes distribués, tout en abordant les défis de la synchronisation, de la tolérance aux pannes, et de la sécurité.

Connaissances préalables recommandées

Réseaux informatique, Systèmes d'exploitation, techniques de synchronisation, programmation concurrente.

Contenu de la matière

Chapitre1. Introduction aux Systèmes Distribués

1) Définition et caractéristiques.
2) Critères de Conception d'un S.E.D
3) Exemples de systèmes distribués
4) Modèles d'architecture (client-serveur, peer-to-peer, et architectures hybrides)
5) Avantages et défis des systèmes distribués

Chapitre 2. Communication dans les systèmes distribués

1) Couches de protocoles (Étude des protocoles utilisés pour la communication dans SD, comme TCP/IP, UDP, RPC, etc)
2) modèle client-serveur
3) appels de procédures à distance
4) communication de groupe
5) Quelques approches pratiques

Chapitre 3. Synchronisation dans les systèmes distribués

1) Gestion du temps.
2) Synchronisation d'horloge.
3) Algorithmes d'élection.
4) Exclusion-mutuelle.
5) Transactions atomiques.
6) Détection de terminaison.

7) Interblocage dans les systèmes distribués.

Chapitre 4. Tolérance aux pannes et sécurité

1) **Tolérance aux pannes**
    - Techniques de détection et de récupération
    - Protocoles de redondance et de réplication
2) **Sécurité**
    - Authentification et autorisation
    - Chiffrement des communications
    - Gestion des clés et infrastructures à clé publique (PKI)

Chapitre 5. Applications et études de cas

1) **Applications pratiques des systèmes distribués**
    - Cloud computing
    - Services web et microservices
    - Bases de données distribuées
2) **Études de cas:**Analyse de systèmes distribués réels (ex. : Google File System, Amazon DynamoDB)

Chapitre 6. Nouvelles tendances et recherches en systèmes distribués

1) **Edge computing et fog computing**
    - Concepts et applications
    - Différences avec le cloud computing
2) **Blockchain et systèmes distribués**
    - Concepts de blockchain
    - Applications de la blockchain dans les systèmes distribués
3) **Internet des objets (IoT)**
    - Intégration des systèmes distribués et IoT
    - Défis et solutions.

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

Références

- Van Steen, M., & Tanenbaum, A. (2002). Distributed systems principles and paradigms. *Network*, *2*(28), 1.
- Coulouris, G., J. Dollimore et T. Kindberg. Distributed Systems – Concepts and Designs, 5th edition, Addison-Wesley, 2012.
- M. Raynal, J-M. Helary,'' Synchronisation et contrôle des systèmes et des programmes répartis'', Eyrolles 1988.

Andrew, S. T., & Herbert, B. (2015). *Modern operating systems*. Pearson Education.

Semestre : S6

Unité d'enseignement : UEF1

Matière : Réseaux avancés 2

Crédits :6

Coefficient :4

Objectifs de l'enseignement.

L'objectif de cette matière est de permettre aux étudiants d'étudier des concepts avancés sur les réseaux informatiques, en mettant l'accent sur :

- Une panoplie des protocoles importants fonctionnant au niveau de la couche liaison de données,
- Les réseaux locaux virtuels
- Les principaux protocoles des réseaux étendus.
- Les réseaux programmables appelés réseaux définis par logiciel (SDN)
- La virtualisation des fonctions réseau (NFV).
- Les réseaux étendus définis par logiciel (SDWAN)

Connaissances préalables recommandées

Les connaissances acquises des modules réseaux avancés 1

Contenu de la matière :

Chapitre 1 : Les réseaux locaux

- Les réseaux locaux virtuels VLANs (protocoles VTP, DTP et Routage inter-VLANs)
- La Redondances dans les réseaux locaux (les protocoles STP, Rapid STP, MSTP, EtherChannel, HSRP).
- Le protocol MPLS (Multi-Protocol Label Switching)
- Les techniques NAT (Network Address Translation) et PAT (Port Address Translation)

Chapitre 2 : Les réseaux étendus

- Les protocoles PPP, PPPoE
- Les réseaux privés virtuels VPN

Chapitre 3 : La couche liaison de données

- Les protocoles ARP (Address Resolution Protocol) et RARP (Reverse Address Resolution Protocol)
- Le protocole HDLC (High-Level Data Link Control),
- Le protocole CSMA/CD (Carrier Sense Multiple Access with Collision Detection)
- Le protocole LLDP (Link Layer Discovery Protocol)

Chapitre 4 : Les réseaux définis par logiciel (SDN : Software-Defined Networking)

- Les concepts d'un SDN
- Architecture d'un SDN
- Le protocole OPENFLOW
- Les contrôleurs SDN

Chapitre 5 : Virtualisation des fonctions réseau (NFV : Network Functions Virtualization)

- Introduction à virtualisation des fonctions réseau

- Architecture NFV (VNF, NFVi et MANO)
- Algorithmes VNF (Placement, Ordonnancement, Migration, Chaînage, multidiffusion)

Chapitre 6 : Réseau étendu défini par logiciel (SD-WAN: Software-defined Wide Area Network)

- Concepts du réseau WAN
- Introduction au réseau SD-WAN
- Architecture SD-WAN
- SD-WAN contre MPLS

Mode d'évaluation : Examen 60% + Contrôle Continu 40%

Références : *(Livres et polycopiés, sites internet, etc).*

- Kurose, J. F., & Ross, K. W. (2021). Computer Networking A Top-Down Approach. Peasron Editions. 2021.
- Bonaventure, O., Networking : Principles, Protocols and Practice, 3rd Edition, 2021.
- Lannone, E., Telecommunication networks, O'Reilly Edition, 2017.
- Réseaux logiciels - Volume 1, Virtualisation, SDN, 5G et sécurité, Guy Pujolle, 2015.
- 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
- 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)

Semestre : S6

Unité d'enseignement : UEF2

Matière : Algorithmiques et complexité avancées

Crédits :5

Coefficient :3

Objectifs de l'enseignement (*Décrire ce que l'étudiant est censé avoir acquis comme compétences après le succès à cette matière – maximum 3 lignes).*
L'objectif de ce cours est d'insister sur les critères de qualité d'un algorithme, notamment la validité et la complexité. L'étudiant ayant déjà un pré-requis sur la notion de complexité des algorithmes (module en S3), apprend à mieux analyser (en termes de validité et de complexité) les algorithmes des plus simples aux plus complexes, utilisant aussi les structures de données avancées (graphes, arbres, tas, tables de hachage). A l'issue de ce cours, L'étudiant devra savoir classer les problèmes (classes N, NL, P, NP, NP-complet), avoir une idée sur les problèmes les plus difficiles (les 21 problèmes NP-complets de Karp), savoir montrer la NP-complétude d'un problème, en utilisant la technique de réduction polynomiale).
Le cours est appuyé par des exercices de TD ainsi que des implémentations en TP, permettant d'évaluer une complexité expérimentale afin de justifier la complexité théorique établie.
Connaissances préalables recommandées (*descriptif succinct des connaissances requises pour pouvoir suivre cet enseignement – Maximum 2 lignes).*
Notions d'algorithmique, structures de données simples et avancées, éléments d'introduction à la complexité.

Contenu de la matière

Chapitre 1. Rappel sur la Complexité des algorithmes
- Introduction à la théorie de la complexité
- Définition et analyse de la complexité d'un algorithme (complexité spatiale et temporelle, Exemples, complexité théorique vs pratique).
- Types d'analyse (pire cas, cas moyen).

Chapitre 2. Complexité des problèmes
1) Classes de complexité des problèmes (Définition des classes P et NP, Problèmes de décision et problèmes d'optimisation, Relation entre les classes P et NP (problème P vs NP)
2) Algorithmes déterministes et non déterministes
3) La classe NP-complet (Définition de la classe NP-complet et de ses propriétés)
4) Exemples de problèmes NP-complets (SAT, clique, couverture de sommet,cycle hamiltonien, voyageur de commerce)
5) Réductions polynomiales

Chapitre 3. Preuve d'algorithmes
1) Tests et preuve théorique
2) Propriété de terminaison
   - Terminaison d'un algorithme itératif
   - Terminaison d'un algorithme récursif
   - Exemples
3) Propriété de validité (correction)
   - Validité d'un algorithme itératif (invariant de boucle)
   - Preuve par induction
   - Validité d'un algorithme récursif
   - Validité d'un algorithme récursif (relation de récurrence)

Chapitre 4. Paradigme « Diviser pour régner » et complexité

1) Introduction au paradigme « Diviser pour régner »
   - (Définition et principe du paradigme « Diviser pour régner »
   - Avantages et inconvénients du paradigme « Diviser pour régner »
   - Champs d'application du paradigme « Diviser pour régner »)
2) Stratégies de résolution de problèmes par « Diviser pour régner »
   - (Décomposition du problème en sous-problèmes plus simples, Résolution récursive des sous-problèmes, Combinaison des solutions des sous-problèmes)
3) Exemples d'algorithmes utilisant le paradigme « Diviser pour régner » & complexité
   - Tri par fusion (MergeSort)
   - Tri rapide (Quicksort)
   - Recherche linéaire vs recherche dichotomique
   - Suite de Fibonacci
   - Algorithme de Strassen pour la multiplication de matrices
   - Tours de Hanoi
4) Complexité d'algorithmes récursifs
   - Équation de récurrence pour les algorithmes récursifs
   - Les différentes variantes de master théorème pour la résolution d'équations de récurrence
5) Méthodes de résolution des équations de récurrence
   - Master théorème
   - Méthode par substitution
   - Méthode par deviner et tester (induction mathématique)
   - Arbre des appels récursifs

Chapitre 5. La programmation dynamique

1) Introduction à la programmation dynamique
2) Différence entre programmation dynamique et recherche exhaustive
3) Exemples simples de problèmes résolus par programmation dynamique (ex:suite de Fibonacci)
4) Programmation dynamique vs Diviser pour régner
5) Approches de la programmation dynamique
   - Approche Top-Down (Memorization), Approche Bottom-Up (Tabulation)
6) Exemples d'algorithmes utilisant la programmation dynamique
   - Problème du sac à dos
   - Chemin le plus court
   - Séquence de nombres la plus longue croissante
7) Limites de la programmation dynamique

Chapitre 6. Algorithmes gloutons

1) Introduction aux algorithmes gloutons
2) Exemples d'algorithmes gloutons
3) Analyse de la performance des algorithmes gloutons
4) Limites des algorithmes gloutons

Chapitre 7. Heuristiques

1) Le problème de la couverture de sommet
   - Heuristique
   - Exemple d'utilisation
   - Garantie de performances
2) Le problème du voyageur de commerce

- Heuristique
- Exemple d'utilisation
- Garantie de performances

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

Références

- Cormen, Leiserson, Rivest Stein, Algorithmique, cours exercices et problèmes, 3ème édition, Dunod, 2010.
- Ivan Lavallée, Complexité et algorithmique avancée, une introduction, édition Hermann 2008.
- Sylvain Perifel, Complexité algorithmique, Ellipses, 2014, 432 p. (ISBN - 9782729886929)
- Nicolas Hermann et Pierre Lescanne, Est-ce que P = NP ? Les Dossiers de La Recherche, 20:64–68, août-octobre 2005

Semestre : *S6*

Unité d'enseignement : UEF2

Matière : Génie logiciel

Crédits : 5

Coefficient : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement (*Décrire ce que l'étudiant est censé avoir acquis comme compétences après le succès à cette matière – maximum 3 lignes).*

- ✓ 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 (*descriptif succinct des connaissances requises pour pouvoir suivre cet enseignement – Maximum 2 lignes).*

- ✓ *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 : (type d'évaluation et pondération)

Contrôle continu (40%), Examen (60%)

Références bibliographiques *(Livres et polycopiés, sites internet, etc)* :

Citer au moins 3 à 4 références classiques et importantes.

✓ 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 *:* 6

Unité d'enseignement : UEM1

Intitulé de la matière : Introduction à l'AI

Crédits : 3

Coefficients : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement :

Le module Apprentissage Machine se justifie par la demande accrue de l'utilisation des algorithmes d'apprentissage machine en informatique et ses domaines. Ce cours est conçu pour fournir aux étudiants une compréhension approfondie des algorithmes et outils de Data science leurs permettant de résoudre des problèmes pratiques de l'intelligence artificielle.

Connaissances préalables recommandées

Statistiques et probabilités, Algèbre linéaire, Calcul numérique et la programmation Python.

Objectifs de l'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.

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 Simulated Annealing (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 bibliographiques :

- Principles of Artificial Intelligence par Nils J. Nilson, 1982, Springer-Verlag Berlin Heidelberg.
- Essentials of Artificial Intelligence par Nils J. Nilson, 1993, Copyright: © Morgan Kaufmann
- Artificial Intelligence: A new synthesis, Nils Nilson, 1997, Elsevier.
- Artificial Intelligence: A Modern Approach par Stuart Russell and Peter Norvig, Ed. Pearson, 2016

Rapport de synthèse - France Intelligence Artificielle. Ministère de l'Enseignement supérieur et de la Recherche, 2017.

Semestre : S6

Unité d'enseignement : UEM1

Matière : Analyse numérique

Crédits :3

Coefficient :2

Mode d'enseignement : Présentiel

Objectifs de l'enseignement : Le module d'analyse Numérique est un module de base dans le domaine du calcul scientifique. Le calcul scientifique étant défini comme la branche qui permet d'implémenter une solution d'un problème sur un calculateur. L'objectif du module est l'acquisition des méthodes de discrétisation, des algorithmes de résolution des problèmes spécifiques posés dans le domaine de l'ingénierie informatique, l'analyse des algorithmes (rapidité, précision et souplesse), et enfin l'estimation des erreurs commises dans les modèles mathématiques employées ainsi que dans leur implémentation. Le domaine d'application est très vaste ; ça peut aller de la résolution d'un problème réel donné aux méthodes élaborées de l'apprentissage profond. Il est indispensable dans la formation d'ingénieur informaticien.

Connaissances préalables recommandées : Algèbre1, Algèbre 2, Analyse Mathématique 1, Analyse Mathématique 2, Analyse Mathématique 3

Contenu de la matière :

Chapitre 1. Analyse d'erreurs

1.1 Introduction générale
1.2 Erreurs de Modélisation
1.3 Représentation des nombres sur un ordinateur :
1.3.1 Représentation des entiers signés
1.3.2 Représentation des réels.
1.3.3 Erreurs dues à la représentation.
1.4 Arithmétique Flottante
1.4.1 Opérations élémentaires
1.4.2 Opérations risquées
1.4.3 Evaluation des polynômes
1.5 Erreurs de Troncatures

Chapitre 2. Equations non linéaire

2.1 Introduction générale
2.2 Méthode de la bissection
2.3 Méthodes des points fixes :
2.3.1 Convergence de la méthode
2.3.2 Interprétation géométrique
2.3.3 Théorème d'Ostrovski.
2.3.4 Ordre de Convergence
2.3.5 Extrapolation d'Aitken
2.4 Méthode de Newton

2.4.1 Interprétation géométrique
2.4.2 Analyse de convergence
2.4.3 Théorème de Convergence Locale
2.5 Méthode de la sécante.
2.6 Applications

Chapitre 3. Systèmes d'équations linéaires

3.1 Introduction générale
3.2 Résolution des Systèmes linéaires triangulaires

Partie I Méthodes Directes

3.3 Elimination de Gauss
3.4 Décomposition LU ou l'interprétation matricielle de la méthode de Gauss
3.4.1 Décomposition de Crout
3.4.2 Décomposition LU et permutation de lignes
3.5 Décomposition de Cholesky
3.6 Méthode de GAUSS avec permutation
3.6.1 Pivot Partiel
3.6.2 Pivot Total
3.6.3 Pivot de la Toure (Rook Pivoting)
3.7 Systèmes tridiagonaux
3.8 Calcul de la Matrice inverse A^{-1}
3.9 Effets de l'arithmétique flottante
3.10 Conditionnement d'une matrice
3.8 Applications

Partie II Méthodes Itératives

3.9 Principe
3.10 Condition générale de Convergence
3.11 Applications :
3.11.1 Méthode de Jacobi
3.11.2 Méthode de Gauss-Seidel
3.11.3 Evaluation de l'erreur
3.11.4 Méthodes de relaxation et condition suffisante de convergence

Chapitre 4. Interpolation

4.1 Introduction
4.2 Matrice de Vandermonde
4.3 Interpolation de Lagrange
4.4 Polynôme de newton
4.5 Erreurs d'interpolation
4.6 Splines Cubiques :
4.6.1 Courbes de la forme $y = f(x)$
4.6.2 Splines paramétrées
4.7 Transformés de Fourier Discrètes.
4.8 Introduction au Nurbs :
4.8.1 B-splines
4.8.2 Génération de courbes

Chapitre 5. Différentiation et Intégration Numériques

1.1 Introduction
1.2 Différentiation Numérique
1.2.1 Différentiation d'ordre 1

1.2.2 Différentiation d'ordre supérieur

1.3 Intégration Numérique :

1.3.1 Formules de Newton -côtes simples et Composées

1.3.2 Méthode de Romberg

1.3.3 Quadratures de Gauss-Legendre

1.3.4 Intégration en utilisant les Splines

1.3.5 Applications

Chapitre 6. Calcul numérique des valeurs et vecteurs propres :

1.1 Introduction

1.2 Localisation des valeurs propres : Théorème de Gershgorin

1.3 Méthode de la puissance

1.4 Méthode de la puissance Inverse

1.5 Méthode QRApplications

Chapitre 7. Equations différentielles

3.1 Introduction

3.2 Méthode d'Euler Explicite

3.3 Méthode de Taylor

3.4 Méthode de Runge-Kutta

3.4.1 Méthode de Runge-Kutta d'ordre 2

3.4.2 Méthode de Runge-Kutta d'ordre 4

3.4.3 Analyse de l'erreur

3.5 Méthodes à pas multiples

Mode d'évaluation : (type d'évaluation et pondération)

Contrôle continu (40%), Examen (60%)

Références bibliographiques *(Livres et polycopiés, sites internet, etc)* :

Citer au moins 3 à 4 références classiques et importantes.

[1] Rappaz, J. et M. Picasso: *Introduction a l'analyse numerique*. Presses polytechniques et universitaires romandes, Lausanne, 1998.

[2] Reddy, J.N.: *An Introduction to the Finite Element Method*. McGraw-Hill, New York, deuxieme edition, 1993.

[3] Asher, U. M. et Petzold L. R.: Computer Methods for Ordinary Differential Equations and Differential-Agebraic Equations. SIAM, Philadelphia, USA,1998.

[4] A. Fortin Analyse Numérique pour ingénieurs. Presses Internationales Polytechniques, 4ieme édition, 2011.

[5] F.Filbet Analyse Numérique-Algorithme et Etude Mathématique. Sciences Sup. Dunod, 2013.

[6] M., Benhamadou, A. Jeribi, Analyse Numérique Matricielle –Méthodes et Algorithmes, exercices et Problèmes corrigés . Edition Ellipses.2020

Semestre : S6

Unité d'enseignement : UET1

Matière : Economie numérique et veille stratégique

Crédits :2

Coefficient : 1

Objectifs de l'enseignement : l'ère de l'information s'est imposé en ce XXIème siècle. L'objectif de cette matière et de doter l'étudiant de connaissances sur deux concepts de l'avenir proche et lointain, il s'agit de l'économie numérique et la veille stratégique.

Contenu de la matière

Chapitre 1 – Economie numérique

- Définition et historique
- Le commerce électronique
- Le contrat électronique et la signature électronique
- La prospection et la publicité par voie électronique
- La propriété intellectuelle et commerciale

Chapitre 2 : Veille stratégique

- Concepts liés à la veille et types de veille
- Les modèles de processus de veille
- Les étapes détaillées du processus de veille
- Panorama de plusieurs outils de veille gratuits pour la collecte, la gestion et la diffusion d'information

Chapitre 3 : La veille et les réseaux sociaux

- Planifier, collecter et organiser l'information
- La veille réputation
- La curation de contenus
- Particularités de la veille sur les réseaux sociaux
- Se créer un (bon) profil LinkedIn et bâtir son réseau
- **Atelier pratique : Twitter, LinkedIn, Mention,**

Tweetdeck Chapitre 4 : Gestion d'un projet de veille

- Gérer un projet de veille
- Travailler en équipe efficacement
- Identifier des sources d'information

Mode d'évaluation : Examen (100%)

Références

- Système d'information pour le management stratégique : l'entreprise intelligente. Ed. McGraw Hill , Paris, 146 p Ce livre a reçu le Prix Harvard
- Francine Séguin, Taïeb Hafsi et Christiane Demers, Le management stratégique, de l'analyse à

l’action, Les
Éditions Transcontinental, 2008.

- Veille stratégique : Comment ne pas être noyé sous les informations. Économies et Sociétés, Série Sciences de Gestion, n°2/1998, p.159-177. LESCA, H. (2001)
- Veille stratégique : passage de la notion de signal faible à la notion de signe d’alerte précoce. Colloque V““T
  2001, Barcelone oct., Actes du colloque, tome 1. LESCA, H. CARON, M-L (1995) - Veille stratégique : créer une intelligence collective au sein de l'entreprise. Revue Française de Gestion, sept. - oct. , p.58-68. LESCA, H RAYMOND, L. (1993)
- Expérimentation d'un système expert pour l'évaluation de la Veille Stratégique dans les PME. Revue Internationale PME, vol.6 n°1 p.49-65. (Québec, Canada) Site internet http://www.veille-strategique.org

Semestre : S7

Unité d'enseignement : UEF1

Matière : Administration et gestion des systèmes et des réseaux

Crédits :5

Coefficient :3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

A la fin de cette matière, les étudiants auront la capacité de déployer, configurer et d'administrer les serveurs et les réseaux et tous les concepts liés aux deux domaines

Contenu de La Matière :

Chapitre 1 : Introduction à l'administration des systèmes et des réseaux

- Introduction à l'administration des systèmes et des réseaux
- Les systèmes d'exploitation serveurs SES (la notion de « rôles »)
- Présentation des outils d'administration graphique: webmin et Cockpit

Chapitre 2 : Automatisation et Administration des systèmes

- Introduction à l'automatisation de l'administration
- Présentation des objectifs et des enjeux de l'automatisation
- Introduction aux outils d'automatisation : Ansible Vs Puppet Vs Chef
- Automatisation de la gestion des systèmes (Utilisateurs, Groupes , Fichiers et Dossiers)
- Automatisation de la gestion des packages
    - o Automatisation de l'installation et la désinstallation de packages
    - o Automatisation de la mise à jour des packages
    - o Automatisation de la gestion des dépôts de packages
- Automatisation de la gestion des services
    - o Automatisation du démarrage, l'arrêt et la vérification des services
    - o Automatisation de la gestion des configurations des services
    - o Automatisation de la gestion des logs des services
- Automatisation de la gestion des sauvegardes et des restaurations
    - o Automatisation de la sauvegarde et la restauration de données
    - o Automatisation de la sauvegarde et la restauration de configurations
- Automatisation de la gestion des mises à jour et des correctifs

Chapitre 3 : Administration des réseaux

- Les notions de Proxy, Reverse proxy, Load balacer et la haute disponibilité.
- La sécurité des systèmes et des réseaux
    - o La notion de DMZ
    - o Gestion des pare-feux
- La surveillance des systèmes et des réseaux
- Le protocole de gestion des réseaux SNMP (Simple Network Management Protocol)

Travaux pratiques :

- TP : Installation d'un système d'exploitation serveur et première configuration
- TP : Installation et configuration d'Ansible + Configuration d'une tâche automatisée avec Ansible
- TP : Automatisation du démarrage, l'arrêt et la vérification des services
- TP : Installation et configuration d'un contrôleur de domaine
- TP : Gestion des utilisateurs, des groupes et des permissions sur un serveur
- TP : Configuration d'un serveur DNS et d'un serveur DHCP
- TP : Déploiement, configuration et administration d'un serveur de messagerie
- TP : Configuration d'un serveur de partage NFS et SAMBA
- TP : Déploiement et configuration d'une solution de surveillance des systèmes et des réseaux (Nagios, Zabbix et Prometheus …)

Mode d'évaluation : 40% contrôle continu 60% examen écrit

Références bibliographiques

- Jason Edelman, Scott S. Lowe, Matt Oswalt, Network Programmability and Automation; O'Reilly Media Publisher , February 2018 , ISBN: 9781491931318

- Negus, Christopher. Linux Bible. John Wiley & Sons, 2020

Semestre : S7

Unité d'enseignement : UEF1

Matière : Virtualisation des réseaux et des systèmes et des réseaux

Crédits :6

Coefficient :4

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Le but de cette matière est la compréhension et la mise en œuvre de la virtualisation système et réseaux

Contenu De La Matière :

1. Introduction à la virtualisation

- o Les avantages de la virtualisation
- o Types de virtualisation : virtualisation de serveur, de bureau et de réseau
- o Types de virtualisation : virtualisation lourde et légère
- o Hyperviseurs de type 1 et hyperviseurs 2
- o Exemples de solutions de virtualisation : VirtualBox, proxmox, Qemu-KVM

2. Virtualisation de serveurs

- o Introduction à la virtualisation de serveurs
- o Configuration des paramètres de la machine virtuelle
- o Gestion des ressources : CPU, RAM, stockage
- o Sauvegarde et restauration de machines virtuelles

3. Virtualisation de stockage

- o Introduction à la virtualisation de stockage
- o Configuration de disques virtuels
- o Configuration de stockage partagé
- o Configuration de disques durs virtuels redondants

4. Conteneurs

- o Introduction aux conteneurs
- o Différence entre la virtualisation lourde et les conteneurs
- o Exemples de technologies de conteneurs : LXC , LXD, Docker, Podman

5. Orchestration de conteneurs

- o Introduction à l'orchestration de conteneurs
- o Utilisation de Kubernetes pour la gestion de conteneurs
- o Configuration de services et de réplicas
- o Configuration des mises à jour et des rollbacks

6. Haute disponibilité et reprise après panne

- o Introduction à la haute disponibilité et à la reprise après panne

o Configuration de clusters de serveurs

o Configuration de failover automatique

o Configuration de sauvegardes et de restauration

7. Virtualisation de réseaux

o Introduction à la virtualisation des réseaux

- Définition de la virtualisation des réseaux
- SDN Vs virtualisation des réseaux
- Avantages et inconvénients de la virtualisation des réseaux

o Les différentes technologies de virtualisation des réseaux

- Virtualisation de réseau basée sur des hyperviseurs
- Virtualisation de réseau basée sur des commutateurs logiciels
- Virtualisation de réseau basée sur des overlays
- Virtualisation de réseau basée sur des conteneurs

o Les architectures de virtualisation des réseaux

- Architecture centralisée
- Architecture distribuée

8. Sécurité et virtualisation

o Risques de sécurité associés à la virtualisation

o Sécurité des machines virtuelles

o Sécurité des réseaux virtuels

o Sécurité des conteneurs

9. Migration vers la virtualisation

o Introduction à la migration vers la virtualisation

o Évaluation des besoins de l'entreprise

o Planification de la migration

o Exécution de la migration

Travaux personnels :

- TP : Installation et configuration de VirtualBox et Proxmox et création d'une machine virtuelle + programme python de création de machine virtuelle libvirt

- TP : Installation et configuration de LXC, LXD, Docker et Podman et création et manipulation d'un conteneur + python open virtualision

- TP : Configuration de Kubernetes (Kublet) pour la gestion de conteneurs + python kubernetes

- TP : virtualization de réseaux avec Open vSwitch and OVN

Mode d'évaluation : 40% contrôle continu 60% examen écrit

Références bibliographiques

- Pierre-Yves Cloux, Thomas Garlot, Johann Kohler , Docker et conteneurs ; Architectures, développement, usages et outils - 3e édition ; Dunod 2022
- Kelsey Hightower, Brendan Burns, Joe Beda, Kubernetes Maîtrisez l'orchestrateur des infrastructures du futur ;Dunod 2019

Semestre : *S7*

Unité d'enseignement : UEF2

Matière : Cryptographie

Crédits : 5

Coefficient : 4

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Formation aux principe essentiels de la cryptographie et ces applications en sécurité informatique (cryptographie appliquée). Apprendre les différents modes d'utilisation des outils cryptographique pour assurer les principaux services de la sécurité : confidentialité, intégrité, authentification et non-répudiation. La première partie de cette matière couvre la cryptographie symétrique essentiellement et les méthodes de chiffrement authentifiées.

Connaissances préalables recommandées /

Contenu de la matière :

Chapitre 1 : Introduction et historique de la cryptographie

1. Définitions élémentaires
2. Objectifs et services de la cryptographie
3. Cryptographie classique : historique, évolution et attaques d'ordre générale.
4. Cryptographie moderne : nouvelle aspects et objectifs

Chapitre 2 : Chiffrement symétrique (Par bloc et parflot)

1. Principe de chiffrement symétrique et scénarios d'utilisations
2. OTP et sécurité algorithmique
3. Principe de sécurité sémantique, et aspects fondamentaux de la définition de sécurité
4. PRNG's et chiffrement par flots : sécurité sémantique des PRNG, exemples d'algorithmes (LFSR,RC4,Slasa ….)
5. Chiffrement par blocs : définitions, principes, historique
6. Building blocks du chiffrement par bloc : PRF, PRP, sécurité sémantique, Réseaux de feistel ;
7. Exemples d'algorithmes :AES,DES,3DES
8. Extension de la problématique de la réutilisation de la clé : MTK, notions de chiffrement aléatoires, par nonce…
9. CPA-security et modes opératoires de chiffrement.

Chapitre 3. Authentification des messages (MAC)

1. Problème des attaques actives
2. Nécessité de protection d'intégrité des messages (par rapport à la confidentialité)
3. Notions de MAC : construction et sécurité sémantique d'un MAC
4. MAC à travers une PRF/PRP : construction et sécurité
5. MAC pour des messages de tailles arbitraires : CBC-MAC, NMAC,PMAC, HMAC
6. Falsification existentielle et conditions de sécurité.
7. Fonction de hachages : définitions, conditions de sécurité, méthodes de construction (Ad-hoc, mathématiques…)
8. Constructions Merklee Damgrade, Par PRF/PRP et conditions de sécurité.
9. Attaques sur les MAC et les fonctions de hachage (exemples)

Chapitre 4 . Chiffrement authentifiée

1. Problème de l'attaque à texte chiffré choisie : Attaques passives/actives
2. Rappelles sur les MAC et les fonctions de hachages

3. Définition de la CCA-sécurité (sécurité contre l'attaque à texte chiffré choisie),
4. Définition de la cryptographie authentifiée.
5. Sécurité sémantique du chiffrement authentifié.
6. Modes de chiffrement authentifié : Mac-then-encrypt, Mac-and-encrypt, encrypt-then-mac GCM,CCM, EAX.
7. Exemples d'attaques : TLS, CBC padding, SSH

Mode d'évaluation : 40% contrôle continu 60% examen écrit

Références bibliographiques

- Cryptography: Theory and Practice, Third Edition. Douglas R. Stinson, 2005.
  Handbook of Applied Cryptography Alfred Menezes, Paul van Oorschot, , 1997.
- Introduction to Modern Cryptography: Principles and Protocols, Jonathan Katz, 2007. Introduction to Cryptography with Coding Theory Wade Trappe and Lawrence C. Washington, 2005.
- William Stallings "Cryptography and Network Security", Fifth Edition ; 2009

Semestre : ***S7***

Unité d'enseignement : UEF2

Matière : Modélisation et évaluation des performances des réseaux

Crédits : 5

Coefficient : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Les objectifs de ce module sont multiples. Le premier objectif serait de familiariser les étudiants avec les principes de la modélisation et de l'évaluation des performances des systèmes informatiques de manière générale et plus particulièrement des réseaux et leurs différents protocoles avec et sans fil. Un autre objectif consiste à sensibiliser l'étudiant au fait que le déploiement de tout système ou réseau ne doit aboutir à un système sous-dimensionné tout en évitant le surdimensionnement. Pour cela, la conception d'un système ou d'un protocole réseau adapté, en respectant le plus possible les objectifs du cahier des charges, est une démarche qui passera obligatoirement, par une étape de modélisation en choisissant le formalisme le plus adéquat, suivie d'une étape d'analyse et d'évaluation des performances."

Connaissances préalables recommandées :

Contenu de la matière :

1. Principes de la modélisation et de l'évaluation des performances des réseaux filaires et mobiles
2. Processus stochastiques
3. Chaînes de Markov et leur application aux réseaux
    - 3.1. Chaînes de Markov à temps discret
    - 3.2. Chaînes de Markov à temps continu
4. Files d'attente
    - 4.1. Files d'attente mono-serveur
    - 4.2. Files d'attente multi-serveurs
    - 4.3. Files d'attente à capacité limitée de clients
    - 4.4. Les réseaux de Petri
    - 4.5. Les réseaux de Petri stochastiques

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

Références bibliographiques :

- Randal Douc, Eric Moulines, Pierre Priouret, Philippe Soulier, Markov Chains, Springer, 2018.
- Annie Choquet-Geniet, Les réseaux de Petri - Un outil de modélisation : Cours et exercices corrigés - Licence, Master, écoles d'ingénieur, Editeur Dunod, 2006 - B. Baynat, Théorie des files d'attente, Hermes 2000.

Semestre : ***S7***

Unité d'enseignement : UEM1

Matière : Apprentissage automatique

Crédits : 3

Coefficient: 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement: "This course provides students with a strong background in machine learning, focusing on deep learning and Q-learning techniques applied to network security. They will learn the primary and recent components of machine learning in this 14-week class, including the data preprocessing methods and the implementation of neural networks. Reinforcement learning, specifically Q-learning, is also contemplated. Applications of such models to real-life network security problems are discussed with practical projects and case studies"

Contenu de la matière :

Chapitre 1: Introduction:

1. Turing test
2. Definition and types of learning: supervised, unsupervised, reinforcement learning
3. Applications in networks and security

Chapitre 2: Heuristics and problem resolution techniques

1. Astar, Branch and Bound
2. Game theory
3. Evolutionary algorithms

Chapitre 3 : Data Preprocessing and Basic Algorithms

1. Data Preprocessing (cleaning, normalization)
2. Unsupervised techniques
    a. Clustering techniques
    b. Outlier detection
3. Supervised techniques
    a. Evaluation metrics (Training, validation and testing, Metrics (Accuracy, Sensitivity, Confusion matrix …)
    b. Regression
    c. Classification (SVMs, random forests…)

Chapitre 4: Deep learning:

1. Deep Learning Fundamentals
    a. Artificial neural networks: perceptrons, backpropagation
    b. Activation, loss, and optimization (gradient descent, Adam)
    c. Advanced methods
        i. Dropout, batch norm…
        ii. Hyperparameter optimization
        iii. Transfer learning
2. Convolution Neural networks
3. Recurrent neural networks

Chapitre 5 : Reinforcement Learning and Q-Learning

1. Introduction to Reinforcement Learning
    a. Basic concepts: (Markov decision process) agent, environment, reward, policy
    b. Differences between reinforcement learning and other ML types
2. Model-based vs model-free
3. Value-based , Policy-based and actor -critic based.

4. Q-Learning
    a. Q-learning: algorithm, Q-table, exploration vs exploitation
    b. Q-deep learning
    c. Practical applications: network security simulations, optimal routing

Semestre : S7

Unité d'enseignement : UED1

Matière : Traitement du signal

Crédits :3

Coefficient :2

Mode d'enseignement : Présentiel

Les objectifs de la matière : Le cours "Traitement de signal" a pour objectif de fournir aux étudiants une compréhension approfondie des signaux et de leur traitement. Les objectifs principaux du cours sont de comprendre les bases théoriques du traitement de signal, d'analyser des signaux en utilisant des techniques mathématiques, de concevoir des filtres numériques, de résoudre des problèmes de traitement de signal, et de maîtriser la programmation en Python pour implémenter des algorithmes de traitement de signal.

Les connaissances préalables recommandées : il est recommandé d'avoir des connaissances préalables en mathématiques telles que le calcul différentiel et intégral, l'algèbre linéaire et la transformée de Fourier

Contenu de la matière :

Chapitre 1. Généralités sur les signaux

Introduction
Définitions
Classifications des signaux
Signaux usuels

Chapitre 2. Analyse de Fourier

Notions générales
Séries de Fourier
Transformée de Fourier

Chapitre 3. Systèmes linéaires invariants dans le temps (SLIT)

Généralités sur les systèmes
Produit de convolution
Transformée de Laplace

Chapitre 4. Corrélation des signaux

Notions de corrélation
Propriétés de la fonction de corrélation

Chapitre 5. Transformée en Z

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

Références bibliographiques :

- J.P. DELMAS, Éléments de théorie du signal : les signaux déterministes, Ellipses, Paris, 1991.
- P. DUVAUT, Traitement du signal, concepts et applications, Hermès, Paris, 1991.
- J. MAX et J.L. LACOUME Méthodes et techniques de traitement du signal et applications aux mesures physiques, Tome 1 : Principes généraux et méthodes classiques, Masson, Paris, 1996

Semestre : S7
Unité d'enseignement : UET1

Intitulé de la matière : Soft-Skills : Amélioration des compétences comportementales

Crédits : 2

Coefficients : 1

Objectifs de l'enseignement :

- Compléter vos savoir-faire métiers par la maîtrise des savoir-être aujourd'hui incontournables pour être un professionnel efficace
- Identifier vos besoins, vos limites, vos croyances négatives, en termes de comportements au travail
- Identifier les principales dimensions des compétences comportementales et développer les éléments fondamentaux de ces attitudes : la relation positive, la maîtrise temporelle, l'attitude résiliente

Contenu de la matière :

1. Introduction
- Présentation de chacun
- Attentes et objectifs visés de chaque participant
- Présentation de la formation

2. Les nouveaux contextes de travail
- Incertitude, instabilité, complexité, ambiguïté, paradoxes
- Les points de tension, le temps, la décision, les modes d'organisation
- Les effets négatifs de la perte de contrôle : stress, usure, fatigue, démotivation
- La digitalisation et les nouvelles organisations

3. Faire le point sur sa cartographie compétences
- Les 3 dimensions des soft skills
- Evaluer ses besoins, ses points faibles, points forts
- Evaluer ses modèles, ses styles d'intervention, de communication, de relation
- Evaluer son leadership
- Des besoins en compétence

4.**Développer des relations positives**
- Gagner en aisance relationnelle
- Identifier ses modèles et ses styles personnels
- Développer les postures et les positionnements efficaces
- Pratiquer la communication inclusive, le nudge, l'influence douce
- Des clés de la communication non violente
- Développer des comportements stables et positifs face aux oppositions, aux confrontations, aux différences

5.Intelligence émotionnelle
- Approfondir la connaissance des émotions et leurs rôles dans les relations interpersonnelles
- Percevoir que jugements, opinions et ressentis ne sont pas des faits
- Développer son ouverture émotionnelle

- Inhibition, réaction, réponse : gérer ses émotions
- Identifier les mécanismes et ses propres sources de stress
- Développer ses leviers d'adaptation et de résilience

5 .**Se placer dans une dynamique positive**

- Se donner toutes les chances de cultiver sa confiance en soi
- Mesurer l'impact de l'éthique et des valeurs sur l'efficacité au quotidien
- Mettre en œuvre des comportements stimulants
- Comprendre les dynamiques d'engagement, d'implication

6.**Stabiliser sa relation au temps**

- Mieux comprendre son rapport au temps
- Développer des comportements stables et une organisation structurée
- Mieux s'organiser personnellement et avec les autres
- Lutter contre les urgences, gérer les priorités

7.S'outiller en mode digital

- Utiliser les outils digitaux de travail et de coopération
- Gérer le rapport vie privée et vie professionnelle

8.Mieux vivre les transitions

- Création, innovation, rupture, comportements novateurs
- Comprendre les leviers de la créativité
- Sortir des limitations, points de vue et croyances personnelles
- Développer des solutions, agir dans une logique de solutions

9.Maintenir ses capacités décisionnelles dans la pression

- Agir et décider sous pression
- Construire dans l'incertain, proposer, trouver des réponses, résoudre les problèmes
- Développer des comportements stables, cadrés et rassurants

Mode d'évaluation : 100 % examen

Références bibliographiques :

Semestre : S8

Unité d'enseignement : UEF1

Matière : Cloud Computing

Crédits : 5

Coefficient : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

L'objectif de cette matière est de fournir aux étudiants une compréhension approfondie des concepts, des technologies et des outils utilisés dans le domaine du Cloud Computing. Comprendre les concepts clés du Cloud Computing, tels que les modèles de service, les modèles de déploiement et les architectures Cloud, d'utiliser des outils de gestion de l'infrastructure Cloud et de déployer des applications sur les plateformes Cloud telles OpenStack.

Connaissances préalables recommandées

Administration des réseaux et systèmes, Virtualisation des réseaux et des systèmes

Contenu de la matière :

- **Introduction au Cloud computing :**
  - o Définitions
  - o Les bases du Cloud computing
  - o Historique et évolution du Cloud Computing
  - o Les avantages et inconvénients
  - o Les types de Cloud computing
    - privé /publique/hybride
  - o stockage /calcul/ réseau
- Les clouds open source **:** Openstack , CloudStack
- Les fournisseurs de services Cloud publique: AWS, Azure, Google Cloud Platform ; Linode
- Déploiement d'un environnement de test
- Infrastructure-as-a-Service (IaaS) :
  - o Les principes de l'IaaS
    - Composants
    - Fonctionnalités
    - Architectures
  - o Les outils de gestion de l'infrastructure : AWS CloudFormation, Azure Resource Manager, TERRAFORM
- Platform-as-a-Service (PaaS) :
  - o Les principes du PaaS : définitions, avantages et inconvénients
  - o Les principaux fournisseurs de PaaS
  - o Déploiement d'une application PaaS
- Software-as-a-Service (SaaS) :
  - o Les principes du SaaS : définitions, avantages et inconvénients

- Les principaux fournisseurs de SaaS

- Stockage et bases de données :
  - le stockage dans le Cloud Computing
  - Les services de stockage Cloud
  - Les services de bases de données Cloud
  - Déploiement d'une base de données Cloud
- Analyse et traitement de données
  - services d'analyse de données disponibles dans le Cloud
  - Utilisation d'outils d'analyse de données tels que Tableau ou Power BI
- Calcul distribué dans le Cloud
- Réseaux dans le Cloud
- Sécurité du Cloud Computing
  - Les risques de sécurité associés au Cloud Computing
  - Les enjeux de sécurité et de conformité dans le Cloud computing
  - Les outils de sécurité Cloud
- Monitoring et gestion des performances
  - Les outils de monitoring et gestion des performances Cloud
  - Configuration d'un monitoring et gestion des performances Cloud
- Migration vers le Cloud
  - les différents scénarios de migration vers le Cloud
    - rehosting
    - refactoring
    - Rewriting

- Le Mobile Cloud Computing

Mode d'évaluation : 40% contrôle continu 60% examen écrit

Travaux personnels :

- Créer un compte AWS / linode /google cloud + Déployer une machine virtuelle
- Déploiment d'Openstack ou CloudStack
- Déployer une infrastructure IaaS : exemple TERRAFORM
- Déploiement d'une infrastructure IaaS avec AWS
- Déployer une application PaaS
- Déployer une Application SaaS de gestion de projet
- Configuration et déploiement d'un cluster Hadoop dans le Cloud

Références bibliographiques:

o Souvik Pal , Dac-Nhuong Le , Prasant Kumar Pattnaik :Cloud Computing Solutions: Architecture, Data Storage, Implementation, and Security , ISBN: 978-1-119-68202-8 May 2022

o Barrie Sosinsky :Cloud Computing Bible ISBN: 978-1-118-02399-0 December 2010

o Judith S. Hurwitz, Daniel Kirsch :Cloud Computing For Dummies, 2nd Edition, ISBN: 978-1-119-54677-1 July 2020

Semestre : S8

Unité d'enseignement : UEF1

Matière : technologies des réseaux sans fil

Crédits :5

Coefficient :4

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Les réseaux sans fil concernent l'ensemble des systèmes de communication qui utilisent les voies hertziennes. L'enseignement présente les réseaux mobiles qui sont à leur 5eme génération 5G. Ensuite les réseaux sans fil détaillés dans leurs différentes typologies. Certaines normes, plus particulièrement les normes 802.11 et 802.16, sont étudiées.

Connaissances préalables recommandées : Réseaux filaires, traitement du signal, Protocoles réseaux

Contenu de la matière

Chapitre 1: Les réseaux sans fil

- Introduction
- Catégorie des réseaux sans fil selon le périmètre géographique
- Principale technologies

Chapitre 2: La technologie IEEE 802.11

- Techniques de modulation du signal
- CSMA/CA
- Routage adhoc (réactif, proactif, hybride, géographique)
- Cas pratiques (Manet, Vanet, WSN)

Chapitre 3: La technologie IEEE 802.16

- Worldwide Interoperability for Microwave Access (WIMAX)
- Objectif, version et Déserte
- Les normes
- Les couches protocolaires
- La couche physique (OFDM et OFDMA)
- La couche MAC
- Service Specific Convergence Sublayer (SSCS)
- MAC Common Part Sublayer (CPS)
- PrivacySublayer (PS)
- La norme IEEE 802.21

Chapitre 4 : Gestion de mobilité

- IEEE 802.11 et Handover
- La macro-mobilité et la micro-mobilité
- Mobile IP (MIP) , - Hiérarchique mobile IP(HMPI)
- Cellular IP et HAWAII

Chapitre 5 : Les approche Cross-layers

- Enjeux des réseaux Adhoc
- Principe du cross-layer
- Communications Cross layer
- Différentes architectures cross-layers
- Méthodes de signalisation
- Cross-layer et notion d'adaptation

Chapitre 6: Introduction aux réseaux cellulaires

- Plan de fréquence
- Les différentes technologies
- Architecture de réseaux cellulaires

Mode devaluation : 40% Contrôle continu, 60 % Examen

Références bibliographiques :

- C. Siva Ram Murthy, B. S. Manoj, Ad Hoc Wireless Networks: Architectures and Protocols, Prentice Hall, 2004
- Alan Holt , Chi-Yu Huang, 802.11 Wireless Networks Security and Analysis, springer, 2010

Semestre : S8

Unité d'enseignement : UEF2

Matière : Introduction à la sécurité des Systèmes et réseaux

Crédits : 5

Coefficient : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

- Identifier les enjeux de la sécurité des systèmes d'information, ainsi que ses acteurs et ses limites
- Identifier les vulnérabilités, menaces des réseaux informatiques
- Utiliser différents outils permettant de détecter une intrusion sur un réseau.

Contenu de la matière

Partie 1 : Operations de sécurité

- Durcissement des systèmes
    - Registres windows
    - Durcicement des systèmes
    - Structure des fichiersæ
    - Processus systèmes
    - Architecture hardware

- Gestion de l'identité et des accès
    - Authentification à Multi-facteurs (MFA)
    - Single sign-on (SSO)
    - Fédération
    - Gestion des accès privilégiés (PAM)
    - Passwordless
    - Cloud access security broker (CASB)

- Infrastructure à clé publique
    - PKI
    - Secure Socket Layer inspection
        - Protection des données sensibles
    - Données à caractère personnel (DCP)
    - Données de cartes de paiement
    - Data Loss Prevention (DLP)
- Indicateurs de compromission
    - Réseaux
    - Hôte
    - Application
    - Autres
- Threat-intelligence et Threat hunting
    - Threat-actor

- Tactiques, techniques et procédures (TTP)
- Niveaux de confiances
- Méthodes de collecte et sources
- Partage de renseignement sur les menaces
- Recherche de menaces

Partie 2 : Gestion des vulnérabilités

- Les concepts liés à la réponse aux vulnérabilités, à la gestion, et le management.
- Implémentation des méthodes d'analyse des vulnérabilités et notions.
- Analyser les résultats des outils d'évaluation de la vulnérabilité
- Analyser les données pour hiérarchiser les vulnérabilités
- Recommander des contrôles pour atténuer les attaques et vulnérabilités logicielles.

TP : Déterminer les outils et techniques pour la détection d'activités malicieuses

- Capture de paquets : wireshark, tcpdump
- EDR : windows events, sysmon, osquery, wazuh
- Analyse de logs et correlation (analyse de comportement suspicieux) : SIEM (splunk ou ELK)
- Réputation DNS et IP : whois, abuseIPDB
- Analyse de fichier : strings, virusTotal, Hashing
- Analyse d'email

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

Références bibliographiques :

- Benjamin Strout, « The Vulnerability Researcher's Handbook: A comprehensive guide to discovering, reporting, and publishing security vulnerabilities », Broché, 2023

- Wenliang Du, « Computer Security: A Hands-on Approach », Broché –2022

- Rebecca Blair Aligning Security Operations with MITRE ATT&CK Framework: An effective guide to leveling up your SOC for better security Broché – 5 mai 2023

Semestre *:* 8

Unité d'enseignement : UEF2

Intitulé de la matière : Techniques d'optimisation pour les réseaux

Crédits : 5

Coefficients : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement (*Décrire ce que l'étudiant est censé avoir acquis comme compétences après le succès à cette matière – maximum 3 lignes).*

- Connaître les enjeux, critères et paramètres d'optimisation dans les réseaux
- Comprendre et maitriser des outils mathématiques utilisés pour modéliser et optimiser les réseaux.

Connaissances préalables recommandées (*descriptif succinct des connaissances requises pour pouvoir suivre cet enseignement – Maximum 2 lignes).*

Connaissances de base en recherche opérationnelle (Théorie des graphes, …), Connaissances de base en réseaux informatique (Modèles OSI & TCP-IP, …) Notions de base en algorithmique.

Contenu de la matière

Chapitre 1. Modélisation des problèmes d'optimisation

- Formalisation des problèmes d'optimisation dans le contexte des réseaux
- Définition des objectifs et des contraintes.
- Quelques problèmes d'optimisation
  - Problème du voyageur de commerce.
  - Problème du sac à dos.
  - Problème de routage.

Chapitre 2. Algorithmes et problèmes issus de la théorie des graphes pour les réseaux

6) Algorithme de Dijkstra pour le plus court chemin.
7) Algorithme de Bellman-Ford pour les plus courts chemins avec des coûts négatifs.
8) Algorithme de Ford-Fulkerson pour le flot maximal dans les réseaux.
9) Arbres couvrants.
10) Coloration d'un graphe
    - Nombre chromatique
    - Algorithme de Welsh-Powell
11) Le problème d'affectation

Chapitre 3. Méthodes de Résolution en Optimisation Combinatoire

1) Optimisation combinatoire
2) Classification des méthodes de résolutions
3) Les méthodes de résolution exactes
   - La méthode séparation et évaluation (Branch and Bound)
   - La méthode de coupes planes (Cutting-Plane)
   - La méthode Branch and Cut
   - Programmation dynamique
4) Les méthodes de résolution approchées
   - Heuristiques
   - Métaheuristiques
   - Classification des métaheuristiques

Métaheuristiques à solution unique
- La méthode de descente
- Recuit Simulé (Simulated Annealing)
- La recherche Tabou (Tabu Search)

Métaheuristiques à population de solutions
- Colonies de fourmis
- Les algorithmes génétiques
- L'optimisation par essaim de particules
- La recherche dispersée

- Intensification et diversification

Chapitre 4. Optimisation multi-objectif

4) Comprendre les problèmes d'optimisation avec plusieurs objectifs
5) Techniques de résolution pour les problèmes d'optimisation multi-objectif dans les réseaux (exemple Plus Court Chemin MultiObjectif).

Chapitre 5. Modélisation et résolution de quelques problèmes d'optimisation dans les réseaux

1) Problème de planification de capacité
2) Méthodes d'optimisation spécifiques aux réseaux sans fil (par exemple, optimisation de la couverture, allocation de ressources).
3) Problème d'optimisation de la topologie du réseau.
4) Problème de routage multi-objectif.

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

Références

- Papadimitriou, Christos H., and Kenneth Steiglitz. *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998..
- Mariño, Pablo Pavón. *Optimization of computer networks: modeling and algorithms: a hands-on approach*. John Wiley & Sons, 2016.
- Bertsekas, Dimitri. *Network optimization: continuous and discrete models*. Vol. 8. Athena Scientific, 1998.
- DE, C. E. S. Cours des *Méthodes de Résolution Exactes Heuristiques* et *Métaheuristiques*.
- H. Mülhenbein. *Evolutionary Algorithms : Theory and Applications*. Wiley,1993.

Semestre : S8

Unité d'enseignement : UEM1

Matière : Développement mobile

Crédits :4

Coefficient :3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement : la finalité de cette matière est d'apporter à l'étudiant des connaissances en matière de développement d'application et système informatique dans des environnements mobile. Avec l'arrivée des smartphones les applications mobiles sont omniprésente que l'on soit client (BtoC), fournisseur (BtoB) ou collaborateur (BtoE). Le but de ce cours est aussi d'apprendre la programmation sous Android, sa plate-forme de développement et les spécificités du développement embarqué sur *smartphone*.

Connaissances requises : algorithmique, connaissance sur le web, POO

Contenu de la matière :

- Introduction au développement mobile
  - Les principaux types de technologies mobiles
  - Introduction à l'architecture des applications mobiles
  - Introduction à la plateforme Android
  - Installation et configuration de l'environnement de développement Android
  - Premier programme Android : création d'une application simple
- Interface utilisateur dans les applications Android
  - Les vues et les ressources d'interface utilisateur Android
  - Création d'interfaces utilisateur simples à l'aide de XML et Java
  - Composants d'interface utilisateur mobile (boutons, champs de texte, listes, etc.)
  - Utilisation de contrôles de disposition et de widgets pour améliorer l'interface utilisateur
- Activités et cycles de vie de l'application
  - Concept d'activités dans les applications Android
  - Cycle de vie des activités
  - Création et utilisation de plusieurs activités
- Intentions et fragments
  - Introduction aux Intentions
  - Utilisation des Intentions pour interagir avec d'autres applications
  - Introduction aux fragments et à leur utilisation pour la création d'interfaces utilisateur réutilisables
- Stockage de données sur Android
  - Utilisation de SQLite pour stocker des données dans des applications Android
  - Utilisation de l'API RESTful
  - Utilisation de SharedPreferences pour stocker des préférences de l'utilisateur
- Utilisation de services
  - Introduction aux services Android
  - Utilisation de services pour effectuer des tâches en arrière-plan
- Notifications et alarmes
  - Introduction aux notifications Android
  - Création de notifications pour informer l'utilisateur
  - Introduction aux alarmes et à leur utilisation pour déclencher des actions
- Réseaux et connectivité

- Utilisation de HTTP pour communiquer avec des serveurs distants
- Utilisation de la connectivité Internet pour accéder à des ressources externes
- Connectivité avec un middlware PHP/Mysql

- Géolocalisation
  - Introduction à la géolocalisation sur Android
  - Utilisation du GPS et de l'emplacement pour suivre la position de l'utilisateur
- Utilisation de capteurs
  - Introduction aux capteurs sur Android
  - Utilisation de capteurs tels que l'accéléromètre, le gyroscope et le capteur de proximité
  - Traitement des données de capteur pour créer des interactions utilisateur avancées

- Multimédia
  - Utilisation des fonctionnalités multimédia d'Android, y compris l'enregistrement audio et vidéo et la lecture de fichiers audio et vidéo
- Conception et développement d'applications en équipe
  - Méthodologies de développement agile
  - Travail en équipe pour concevoir et développer une application Android
- Projet de développement d'application
  - Conception et développement d'une application Android fonctionnelle en utilisant les compétences acquises au cours des semaines précédentes
- Semaine 13-14: Déploiement et publication
  - Introduction aux processus de déploiement et de publication d'applications

mobiles Travaux pratiques suggérés:

- Création d'une application simple avec une interface utilisateur
- Utilisation d'un service pour exécuter une tâche en arrière-plan
- Stockage et récupération de données dans une application Android à l'aide de SQLite
- Utilisation de notifications pour informer l'utilisateur des événements importants
- Création d'une application basée sur la géolocalisation pour suivre la position de l'utilisateur
- Utilisation de capteurs pour collecter des données

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

Références:

- Créez des applications pour Android - OpenClassrooms
  https://openclassrooms.com/courses/creez-des-applications-pour-android
- Développement Android - Jean-Francois Lalande
  http://www.univ-orleans.fr/lifo/Members/Jean-Francois.Lalande/enseignement/android/cours-android.pdf

Semestre *:* 8

Intitulé de l'UE : UEM1

Intitulé de la matière : Système embarqués

Crédits : 3

Coefficients : 2

Objectifs de l'enseignement :

Objectifs de l'enseignement (*Décrire ce que l'étudiant est censé avoir acquis comme compétences après le succès à cette matière – maximum 3 lignes).*
Les systèmes embarqués sont des dispositifs combinant électronique et informatique, opérant de manière autonome. Ils représentent des entités intelligentes et communicantes, dotées de capacités de calcul croissantes et capables d'interagir avec leur environnement grâce à des moyens de communication avancés. Ces systèmes sont souvent équipés de systèmes d'exploitation temps réel. Ce cours vise à familiariser les étudiants avec les caractéristiques spécifiques de ces systèmes, à étudier en profondeur leurs systèmes d'exploitation, et à les initier aux méthodes et outils de programmation nécessaires pour leur développement et leur déploiement efficace.
Connaissances préalables recommandées (*descriptif succinct des connaissances requises pour pouvoir suivre cet enseignement – Maximum 2 lignes).*
Connaissances acquises durant le cursus de formation de la licence (Architecture des ordinateurs, Systèmes d'exploitation)**,** Programmation et langages informatiques (C, C++,.. etc.)

Contenu de la matière

Chapitre1. Introduction aux systèmes embarqués

1) Définition et caractéristiques des systèmes embarqués.
2) Importance et domaines d'application (automobile, aérospatial, appareils médicaux, IoT).

Chapitre 2. Architecture matérielle des systèmes embarqués

- Processeurs embarqués : architectures (ARM, MIPS, etc.)
- Mémoires embarquées : types, hiérarchie et gestion.
- Périphériques d'entrée/sortie et interfaces de communication.

Chapitre 3. Systèmes d'exploitation embarqués

- Critères de choix des systèmes d'exploitation (RTOS, RT-Linux, VxWorks, etc.).
- Gestion des ressources : planification temps réel, gestion de la mémoire, gestion de l'énergie.
- Synchronisation et communication entre tâches.

Chapitre 4. Développement logiciel pour systèmes embarqués

1) Langages de programmation adaptés aux systèmes embarqués (C, C++, Rust, Java Embarqué: KVM, J2ME, RT-java etc.).
2) Développement de drivers et de services bas-niveau.
3) Techniques de débogage et de test spécifiques aux systèmes embarqués.

Chapitre 5. Conception matérielle et prototypage

1) Conception de circuits imprimés (PCB) pour systèmes embarqués.
2) Outils de modélisation et de simulation (ex. : MATLAB/Simulink).
3) Techniques de prototypage rapide (prototypage FPGA, plateformes de développement).

Chapitre 6. Réseaux et connectivité pour systèmes embarqués

1) Protocoles de communication sans fil (WiFi, Bluetooth, Zigbee, LoRa, etc.).
2) Réseaux de capteurs sans fil (WSN) et IoT.
3) Sécurité et gestion des données dans les systèmes embarqués connectés.

Chapitre 7. Applications

1) Études de cas sur des applications réelles de systèmes embarqués (automobiles connectées, drones, dispositifs médicaux, etc.).
2) Analyse de performances et optimisation des systèmes embarqués.

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

Références

- Kamal. (2006). *Embedded Systems: Architechture, Programming and Design*. McGraw-Hill Science/Engineering/Math. Mariño, Pablo Pavón. *Optimization of computer networks: modeling and algorithms: a hands-on approach*. John Wiley & Sons, 2016.
- Berger, A. (2001). *Embedded systems design: an introduction to processes, tools, and techniques*. CRC Press..
- Barr, M., & Massa, A. (2006). *Programming embedded systems: with C and GNU development tools*. " O'Reilly Media, Inc.".
- Chattopadhyay, S. (2023). *Embedded System Design*. PHI Learning Pvt. Ltd..

Semestre *:* 8

Unité d'enseignement : UET1

Intitulé de la matière : Projet pluridisciplinaire

Crédits : 4

Coefficients : 3

Objectifs de l'enseignement

Le but de cette matière est l'immersion des étudiants dans le milieu socio-économique en les plaçant dans des stages dans les entreprises. Le projet se déroule durant le second semestre de la quatrième année. Il consiste en la conception et la réalisation d'un petit projet informatique qui se déroule dans une entreprise.

Stage : un stage de 15 jours dans une entreprise pendant les vacances du printemps.

Déroulement du projet :

Le projet est décrit à travers un cahier des charges précis et peut porter sur des thèmes très variés. Il est proposé et encadré par un enseignant du département et il doit couvrir au moins deux disciplines.
Le groupe de projet est composé de 4 étudiants au maximum. 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 de l'entreprise d'accueil en respectant les points suivants :

- ✓ Analyse et découpage du travail,
- ✓ Répartition des charges de travail entre les membres du groupe par l'encadreur.
- ✓ Circulation de l'information entre les membres du groupe,
- ✓ Mise en place d'un planning de travail,
- ✓ Exposés périodiques de l'avancement du projet,
- ✓ Délivrance des livrables fixés dans la fiche de projet,
- ✓ Rédaction d'un rapport de stage (entre 20 et 30 pages).
- ✓ Exposé du travail réalisé devant une commission d'examen.

Modalités d'évaluation du projet

L'évaluation du projet aura la forme d'une note sur vingt et repose sur les critères suivants :
Le groupe remet un rapport de stage et le logiciel accompagnés d'une lettre de présence dans l'entreprise d'accueil.
Une commission d'examen composée de l'encadreur, d'un enseignant du département et éventuellement d'un représentant de l'entreprise d'accueil examinera le dossier en présence du groupe d'étudiants.
La note finale est délivrée à chaque étudiant du groupe (note globale attribuée à l'équipe ou individuelle au cas où il est fait constat que le volume de travail fourni par les membres est inégal) selon le barème suivant :

- Le rapport de stage est noté sur 6 points
- Le logiciel est noté sur 6 points
- La présentation et les réponses aux questions sont notées sur 6 points.

(La note attribuée sur 18 est égale à la moyenne des notes octroyées par les membres de la commission d'examen).

- Une note de travail continu (sur 2 points) est donnée par l'encadreur. Cette note validera en quelque sorte l'assiduité des étudiants aux réunions périodiques et le respect des objectifs fixés.

Mode d'évaluation : Soutenance (100%)

Semestre : ***S9***

Unité d'enseignement : UEF1

Matière : Internet d'objets (IoT) et ses applications

Crédits : 5

Coefficient : 4

Mode d'enseignement : Présentiel

Objectifs de l'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-Edge. A travers ce cours l'étudiant va apprendre à concevoir une solution IoT-Edge.

Connaissances préalables recommandées : Systèmes d'exploitation, Réseaux, Programmation web, Python.

Contenu de la matière :
Partie I
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 IoT enabling technologies
  - 1.5.1 Réseaux de capteurs sans fil (WSN)
  - 1.5.2 Cloud computing
  - 1.5.3 Big data analytics
  - 1.5.4 Protocoles de communication
  - 1.5.5 Systèmes embarqués
- 1.6 Challenges

2.Architecture d'IoT et protocoles

- 2.1 Architecture d'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

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

3.3 Etude de cas d'un système IoT

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 physical devices)
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
4.5 Cas d'étude illustratifs

5. Technologie Edge pour les applications IoT

5.1. Introduction
5.2. Concept et fonctionnement

5.3. Gestion de ressources
5.4. Problèmes d'allocation de ressources
5.5. Modèles de communication IoT-Edge
5.6. Algorithme de distribution de Requêtes-Noeud Edge

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

Références bibliographiques :

- ArshdeepBahga and Vijay Madisetti. 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 Muthucumaru and 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 : UEF1

Matière : Sécurité des systèmes et réseaux

Crédits : 5

Coefficient : 4

Mode d'enseignement : Présentiel

Objectifs de l'enseignement : maitriser les techniques de sécurisation de réseaux

Contenu de la matière :

Partie 1 : Réponses aux incidents

- Plateforme de cyber-défense
- La pyramide de la douleur (pyramid of pain)
- La cyber kill chain
- La kill chain unifiée
- Le model en diamant
- Mitre

Partie 2 : Sécurité des serveurs

- Attaques des serveurs web
- Méthodologies
- Outils
- Contre-mesures
- Gestion des correctifs

Partie 3 : Sécurité des réseaux et des réseaux sans fil

- Sécurité des réseaux IP
  - Sécurité de la couche Liaison de données
  - Sécurité de la couche routage
  - Sécurité de la couche transport
  - Sécurité de la couche application

- Sécurité des réseaux sans fil
  - Chiffrement des réseaux wifi
  - Menaces sur les réseaux wifi
  - Méthodologies et outils
  - Attaques bluetooth
  - Contre-mesures

Partie 4 : Sécurité IOT & OT

- IOT et OT attaques
- Méthodologies et outils

- Contre-mesures

Partie 5 : Sécurité cloud

- Gestion des identités et des accès
- Mettre en œuvre la protection de la plate-forme
- Sécuriser les données et les applications
- Sécuriser les services et micro-services
- Gérer les opérations de sécurité

Partie 6 : Blockchain

- Historique, Principe de fonctionnement
- Fonctionnement des transactions
- Le stockage de la blockchain
- Les avantages et les inconvénients
- Les enjeux de la blockchain
- Blockchain privée / publique
- Les applications décentralisées (DApp)
- Les smart contracts

Mode d'évaluation : (type d'évaluation et pondération)

Contrôle continu (40%), Examen (60%)

Références bibliographiques :

- William Stalling, "5G Wireless: A Comprehensive Introduction", Broché, 2016
- Charlie Kaufman, "Network Security: Private Communication in a Public World", 2020
- J. Michael Stewart, Denise Kinsey, " Network Security, Firewalls, and VPNs (Issa)" 3e Édition, Format Kindle, 2020

Semestre : ***S9***

Unité d'enseignement : UEF2

Matière : Multimédia et QoS

Crédits : 5

Coefficient : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Étudier la problématique de l'intégration des applications multimédias dans un réseau dit de moindre effort, en occurrence l'Internet. Dans cette UE, on lit et on analyse des papiers scientifiques qui décrivent cette problématique et des papiers qui étaient influents dans l'évolution des normes qui encadrent la transmission des flux multimédia dans l'internet.

Contenu de la matière :

- Introduction au multimédia dans un réseau dit de moindre effort.
- Les pertes dans le réseau : caractéristiques et techniques pour en faire face. La voix sur IP (VoIP). Signalisation SIP.
- Les délais dans le réseau : caractéristiques et techniques pour en faire face. La lecture en transit (streaming). Signalisation RTSP.
- Partage de la bande passante dans un réseau dit de moindre effort. Modélisation avancée de la formule TCP-Friendly. Protocole DCCP, avec TFRC.
- Réservation de ressources. IntServ, DiffServ, MPLS, et RSVP.

Mode d'évaluation : 40% contrôle continu 60% examen écrit

Références

- J. Rosenberg, H. Schulzrinne. "Timer Reconsideration for Enhanced RTP Scalability". In Proc. Infocom. 1998.
- J. Padhye, V. Firoiu, D. Towsley, J. Kurose. "Modeling TCP Throughput: A Simple Model and its Empirical Validation". In Proc. Sigcomm. 1998.
- S. B. Moon, J. Kurose, D. Towsley. "Packet Audio Playout Delay Adjustment: Performance Bounds and Algorithms". Multimedia Systems, Volume 6, Issue 1. Janvier 1998.
- W. Jiang, H. Schulzrinne. "Perceived Quality of Packet Audio under Bursty Losses". In Proc. IEEE Infocom, 2002.

Semestre : S9

Unité d'enseignement : UEF2

Matière : Architectures micro-services

Crédits : 5

Coefficient : 3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement : À l'issue de cette formation Microservices, vous aurez acquis les **connaissances et les compétences nécessaires**

- Comprendre l'intérêt et le fonctionnement d'une architecture microservices
- Avoir une vue d'ensemble de Spring, Spring Data et Spring Cloud dans le cadre des microservices
- Connaître les différentes architectures REST à l'œuvre avec les microservices
- Envisager les microservices dans le cadre d'une base de données NoSQL traditionnelle
- Connaître les méthodes pour router, monitorer, journaliser et déployer des microservices

Contenu de la matière :

1. Introduction aux différents types d'architecture
    a. Vision d'ensemble
    b. Les applications monolithiques
    c. Les architectures SOA/WOA/ROA
    d. Présentation de l'architecture en microservices
2. Présentation du projet
    a. Architecture du projet
    b. Place des outils dans le projet
3. Spring
    a. Présentation du framework
    b. Fonctionnement global
    c. Injection de dépendances / Inversion de contrôle
    d. Spring MVC : principe
    e. JPA/Hibernate et Spring Data
4. REST
    a. Principes
    b. HATEOAS
    c. Tests automatisés
    d. Versionning
    e. Documentation
    f. Exemple avec : Spring Boot, Spring MVC, Spring HATEOAS, Swagger, Spring REST Docs
5. Création de microservices avec Spring Data et Spring Cloud
    a. Configuration d'une architecture microservices
    b. Externalisation et centralisation de la configuration
    c. Environnements et profils

d. Sécurité
e. Configuration des serveurs avec Spring Cloud Config

6. Les bases de données NoSQL
    a. Présentation des bases de données NOSQL
    b. Comparaison avec les bases de données relationnelles
    c. Les différentes familles de NOSQL (document, graph, clés/valeurs, colonnes)
    d. Présentation des différentes bases de données NOSQL
7. Les APIs GraphQL Du REST au GraphQL
8. Communication entre microservices
    a. Les « Message Oriented Middleware » (MOM)
    b. Présentation de Kafka
    c. Mise en place de Kafka et Zookeeper
    d. Mise en place d'un microservice Node.js
9. Enregistrement et découverte des services
    a. Les registres de services : présentation
    b. Consul
    c. Eureka
    d. Kubernetes
    e. Zookeeper
    f. Mise en place du registre de services
10. Le routage
    a. Load balancing
    b. Mise en application: Load balancing avec Zuul
11. Monitoring
    a. Gestion des ressources
    b. Récupération des métriques
    c. Mise en application : Actuator Spring Boot
12. Journalisation
    a. Gestion des logs
    b. ELK
    c. Graylog
    d. Identifiant de corrélation
    e. Mise en place de la gestion des logs
13. Déploiement des microservices
    a. Cloud (Heroku, AWS, Azure)
    b. Mise en application : packaging et déploiement d'une application microservices

Mode d'évaluation : 40% contrôle continu 60% examen écrit

Références bibliographiques

- Martin L. Abbott, Michael T. Fisher: The Art of Scalability – Scalable Web Architecture, Processes, and Organizations for the Modern Enterprise; refer to: http://theartofscalability.com.
- Blog: Adopting Microservices at Netflix – Lessons for Architectural Design; refer to: www.nginx.com/blog/microservices-at-netflix-architectural-best-practices.

- Melvin E. Conway: Conway's Law: How Do CommitteesInvent? ; refer to: www.melconway.com/research/committees.html.
- Eric Evans: Domain-Driven Design – Tackling Complexity in the Heart of Software, Addison-Wesley, 2003
- Blog: James Hughes on Micro Service Architecture; refer to: http://yobriefca.se/blog/2013/04/28/micro-service-architecture.

Semestre : S9

Unité d'enseignement : UEM1

Matière : Gestion de projets

Crédits :4

Coefficient :3

Mode d'enseignement : Présentiel

Objectifs de l'enseignement

Décrire ce que l'étudiant est censé acquérir pour maitriser le bon déroulement d'un projet et assurer les enjeux aussi bien temporels et financière

- Permettre à l'étudiant de comprendre l'enjeu majeur de la gestion de projet.
- Initier l'étudiant au processus d'organisation et de planification.
- Entrainer l'étudiant à l'application de processus, méthodes et outils de planification.
- Initier l'étudiant aux environnements de gestion projet.

Connaissances préalables recommandées

Contenu de la matière :

1. **Introduction**
   - Définition de concepts de base.
   - Notions de projet et gestion de projet.

2. **Les modèles de gestion de projet**
   - Les modèles de basés sur les livrables.
   - Les modèles basés sur le risque.

3. **Les éléments de Gestion de projet**
   - Les enjeux de gestion de projet.
   - Les activités de gestion de projet.
   - La structure de gestion de projet.
   - Les risques et la gestion de projet.

4. **L'organisation des équipes de programmation**
   - L'organisation de base.
   - Les outils de support.

5. **Les éléments de la planification**
   - La productivité du programmeur.
   - Echéance et jalonnement d'un projet.

6. **Le processus de planification**
   - Découpage et coordination des activités.
   - Les outils de planification (ordonnancement des activités et affectation des ressources).
   - Les environnements de planification (ex : MSPROJECT).

7. **Estimation des charges, délais et coût**

- Les options alternatives : méthodes.
- La précision de la taille des programmes.
- Modèle d'estimation algorithmique.

8. Approche Agile
- Principes et méthodes Agiles.
- Présentation : Méthode Scrum. Méthode XP.

9. Techniques d'analyse d'options et de décisions.
- Graphe polaire.
- Matrice des bénéfices

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

Références bibliographiques :

- Principles of software engineering management by Tom GILB Edition Lavoisier.
- Software Engineering: A Practitioner's Approach by Roger S Pressman.
- Software Project Management in Practice by Pankaj Jalote.
- Génie logiciel: principes, méthodes et techniques by Alfred Strohmeier et Didier Buchs.

Semestre : S9

Unité d'enseignement : UEM1

Intitulé de la matière : Business Intelligence

Crédits : 4

Coefficients : 2

Objectives:
Business Intelligence (BI) is a crucial aspect of modern business strategy, focusing on the utilization of data-driven insights to make informed decisions and gain a competitive advantage. This course introduces students to the concepts, technologies, and practices of BI, covering topics such as data warehousing, data mining, analytics, visualization, and decision support systems. Through lectures, case studies, and hands-on projects, students will learn how to collect, analyze, and interpret data to support organizational decision-making and improve business performance.

Content:

- Chapter 1: Introduction to Business Intelligence
- Chapter 2: Data Warehousing and ETL Processes
- Chapter 3: Data Modeling and Dimensional Design
- Chapter 4: Data Mining and Predictive Analytics
- Chapter 5: BI Tools and Technologies
- Chapter 6: Advanced Analytics and Big Data
- Chapter 7: Business Intelligence Applications and Case Studies

Evaluation: 40 % Continuous assessment + 60 % exam.

References:

- MÜLLER, Roland M. et LENZ, Hans-Joachim. *Business intelligence*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2013.
- HOWSON, Cindi. *Successful business intelligence*. Emeryville : McGraw-Hill Professional Publishing, 2007.
- MICHALEWICZ, Zbigniew, SCHMIDT, Martin, MICHALEWICZ, Matthew, *et al.* *Adaptive business intelligence*. Springer Berlin Heidelberg, 2006.

Semestre : S9

Unité d'enseignement : UET1

Intitulé de la matière : Aspect Juridique pour les technologies de l'information

Crédits : 2

Coefficients : 1

Objectifs de l'enseignement

- Connaitre l'aspect juridique et la réglementation de la cybersécurité en Algérie

Connaissances préalables recommandées (*descriptif succinct des connaissances requises pour pouvoir suivre cet enseignement – Maximum 2 lignes).*

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 : (type d'évaluation et pondération)
Contrôle continu (40%), Examen (60%)

Références

- Compilation des décrets, arrêtés (Réglementations en vigueurs)