

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE**

**Cahier des charges
d'habilitation d'une Formation à
recrutement national**

Master

- Génie Logiciel -

Université Abdelhamid Mehri – Constantine 2

الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

دفتـر الشـروط

لتأهيل تكوين ذات تسجيل وطني

ماسـتر

- هندسة البرمجيات -

جامعة عبد الحميد مهري – قسنطينة 2

L - Organisation de la formation

L.1 - Fiche d'organisation semestrielle des enseignements (Prière de présenter les fiches des 4 semestres)

Semestre 1:

Unités d'enseignement	Matières	Crédits	Coefficient	Volume horaire hebdomadaire			VHS (14-16 semaines)	Autre*	Mode d'évaluation	
	Intitulé			Cours	TD	TP			Contrôle Continu	Examen
UE Fondamentale Code : UEF 1.1.1 Crédits : 13 Coefficients : 8	Programmation orientée composants	7	4	3h		3h	84h		✓	✓
	Programmation Logique et Fonctionnelle	6	4	3h	1h.30	1h30	84h		✓	✓
UE Fondamentale Code : UEF 1.1.2 Crédits : 12 Coefficients : 7	Sécurité des Réseaux	5	3	1h30	1h30	1h30	6 3h		✓	✓
	Systèmes Distribués	7	4	3h	1h30		84h		✓	✓
UE Methodologique Code : UEM1.1.1 Crédits : 5 Coefficients : 3	Statistiques et Analyse de données	3	2	1h30	1h30		42h		✓	✓
	Anglais	2	1	1h.30			21h			✓
Total semestre 1		30	18	13h30	6h	6h	357h			

*Autres travaux supplémentaires

Semestre 2:

Unités d'enseignement	Matières	Crédits	Coefficient	Volume horaire hebdomadaire			VHS (14-16 semaines)	Autre*	Mode d'évaluation	
	Intitulé			Cours	TD	TP			Contrôle Continu	Examen
UE Fondamentale Code : UEF 1.2.1 Crédits : 13 Coefficients : 7	Spécification et vérification des systèmes	7	4	3h	1h30	1h30	84h		✓	✓
	Modélisation et Simulation des Systèmes Informatiques	6	3	1h30	1h30	1h30	63h		✓	✓
UE Fondamentale Code : UEF 1.2.2 Crédits : 11 Coefficients : 6	Exigences et Développement de Logiciels	7	4	3h		3h	42h		✓	✓
	Analyse de la qualité du logiciel	4	2	1h30h	1h30		84h		✓	✓
UE Méthodologique Code : UEM 1.2.1 Crédits : 4 Coefficients : 2	Techniques de Maintenance du Logiciel	4	2	1h30		1h30	42h		✓	
UE Découverte Code : UED 1.2.1 Crédits : 2 Coefficients : 1	Pratique Professionnelle du GL	2	1	1h30			21h		✓	
Total semestre 2		30	16	12h	4h30	7h30h	336h			

*Autres travaux supplémentaires

Semestre 3:

Unités d'enseignement	Matières	Crédits	Coefficient	Volume horaire hebdomadaire			VHS (14-16 semaines)	Autre*	Mode d'évaluation	
	Intitulé			Cours	TD	TP			Contrôle Continu	Examen
UE Fondamentale Code : UEF 2.1.1 Crédits : 10 Coefficients : 7	Ingénierie Dirigée par les Modèles	4	4	1h30		1h30	42h		✓	✓
	Architectures Logicielles	6	3	1h30		1h30	42h		✓	✓
UE Fondamentale Code : UEF 2.1.2 Crédits : 10 Coefficients : 7	SOA et Cloud Computing	6	4	1h30		1h30	42h		✓	✓
	Administration des BDs	4	3	1h30		1h30	42h		✓	✓
UE Méthodologique Code : UEM 2.1.1 Crédits : 10 Coefficients : 7	Programmation Orientée Agents	6	4	1h30	1h30	1h30	63h		✓	✓
	Méthodes Agiles	4	3	1h30		1h30	42h		✓	✓
Total semestre 3		30	21	9h	1h30	9h	273h			

*Autres travaux supplémentaires

Semestre 4 :

Stage en entreprise sanctionné par un mémoire et une soutenance.

	VHS	Coeff	Crédits
Travail Personnel	14 semaines	9	27
Stage en entreprise			
Séminaires à thèmes ouverts	1h30/semaine	1	3
Autre (préciser)			
Total Semestre 4	14 semaines	10	30

Récapitulatif global de la formation: *(indiquer le VH global séparé en cours, TD, pour les 04 semestres d'enseignement, pour les différents types d'UE)*

VH \ UE	UEF	UEM	UED	UET	Total
Cours	25h30	7h30	3h		36h
TD	9h	3h	0h		12h
TP	18h	4h30	0h		22h30
Travail personnel	24h	14h	4h		42h
Autre (préciser)					
Total	52h30	15h	3h		70h30
Crédits	95	18	7		120
% en crédits pour chaque UE	79,16%	15%	5,83%	25	100%

L.2 - Fiches d'organisation des unités d'enseignement

(Etablir une fiche par UE)

Semestre : 01

UE: UEF 1.1.1(Programmation des Systèmes répartis)

Répartition du volume horaire global de l'UE et de ses matières	Cours : 6h TD : 1.5h TP: 4.5 Travail personnel : 5h
Crédits et coefficients affectés à l'UE et à ses matières	UE : UEF 1.1 crédits 13 Matière 1 : Programmation Orientée Composants Crédits : 7 Coefficient : 4 Matière 2 : Programmation Logique et Fonctionnelle Crédits : 6 Coefficient : 4
Mode d'évaluation (continu ou examen)	Continu et examen
Description des matières	Programmation Orientée Composants: A l'issue de ce module, l'étudiant maîtrisera les éléments nécessaires pour la conception d'applications distribuées à base de composants et comprendra le rôle des intergiciels dans le développement et l'exécution d'applications distribuées. Il sera aussi initié quelques plateformes à base de composants. Programmation Logique et Fonctionnelle: En s'appuyant sur la programmation déclarative (programmation logique et fonctionnelle), l'objectif de ce cours est d'exposer les techniques de base de l'Intelligence Artificielle. Situer l'Intelligence Artificielle en tant que discipline scientifique, Maîtriser les concepts de base de l'intelligence artificielle.

Semestre : 01
UE : UEF 1.1.2

Répartition du volume horaire global de l'UE et de ses matières	Cours : 4h30 TD : 3h TP: 1h30 Travail personnel :
Crédits et coefficients affectés à l'UE et à ses matières	UE : UEF 1.1.2 crédits : 14 Matière 1 : Sécurité des Réseaux Crédits : 5 Coefficient : 3 Matière 2 : Systèmes Distribués Crédits : 7 Coefficient : 4
Mode d'évaluation (continu ou examen)	Continu et examen
Description des matières	Sécurité des réseaux : L'étudiant est censé acquérir une maîtrise globale de la sécurisation d'un réseau privé d'entreprise utilisant les technologies Internet/intranet, et de son interconnexion avec des réseaux extérieurs. Systèmes distribués: Acquérir les concepts de base concernant la structure et le fonctionnement d'un système distribué, tous les mécanismes de partage d'information et de communication doivent être étudiés.

Semestre : 01
UE : UEM 1.1.1

Répartition du volume horaire global de l'UE et de ses matières	Cours : 3h TD : 1h30 TP: Travail personnel :
Crédits et coefficients affectés à l'UE et à ses matières	UE : UEM 1.1 crédits : 5 Matière 1 : Statistiques et Analyse de données Crédits : 3 Coefficient : 2 Matière 2 : Anglais Crédits : 2 Coefficient : 1
Mode d'évaluation (continu ou examen)	Examen
Description des Matières	Statistiques et Analyse de données : Il est tout à fait courant qu'après un certain traitement ou fonctionnement d'un système on se trouve en face d'une masse importante de données recueillies qu'on a besoin d'analyser. L'analyse des données est un domaine scientifique très large qui comporte des techniques variées principalement empruntées au domaine de la statistique et l'apprentissage automatique. Outre la préparation et le nettoyage des données, elle vise à mettre en exergue les relations qui peuvent exister entre les données, à résumer les données et / ou à établir des prédictions. Cette matière a pour objectif de permettre à l'étudiant de se familiariser avec le domaine de l'analyse de données et d'acquérir une certaine maîtrise des techniques et outils permettant d'analyser les données. L'étudiant aura l'occasion d'apprendre comment appréhender un tableau de données et à choisir et appliquer la technique appropriée pour répondre à un besoin d'analyse précis compte tenu de la nature des données disponibles.
OBS	Il est préférable de faire des sorties sur terrains dans la matière Statistiques et Analyse de données

Semestre : 02

UE : UF 1.2.1

Répartition du volume horaire global de l'UE et de ses matières	Cours : 4h30 h TD : 3h TP: 3 h Travail personnel : 6h
Crédits et coefficients affectés à l'UE et à ses matières	UE : UF 1.2.1 crédits 13 Matière 1 : Spécification et vérification des systèmes Crédits : 7 Coefficient : 4 Matière 2 : Modélisation et Evaluation des Systèmes Informatiques Crédits : 6 Coefficient : 3
Mode d'évaluation (continu ou examen)	Continu et examen
Description des matières	Spécification et vérification des systèmes : Les systèmes critiques nécessitent le recours à certaines des méthodes formelles pour les phases amont et aval du Génie Logiciel : spécifications et vérification. Dégager l'essentiel des perspectives offertes par ces méthodes fut le challenge. L'objectif de ce cours est justement de faire le point sur les techniques formelles utilisables dans le contexte des systèmes répartis caractérisés par des composants distincts, chacun d'eux s'exécutant potentiellement sur un processeur différent. De plus, un panorama des techniques de vérification formelle les plus appropriées dans ce cadre est aussi donné. Modélisation et Evaluation des Systèmes Informatiques : L'objectif de ce module est de sensibiliser les étudiants aux problèmes de modélisation et d'évaluation des performances des systèmes réels. A l'issue de ce module, l'étudiant sera capable de poser un modèle représentant le

	système étudié et de sélectionner, parmi les méthodes analytiques et de simulation, la plus appropriée.
--	---

Semestre : 02

UE : UF 1.2.2

Répartition du volume horaire global de l'UE et de ses matières	Cours : 4.5 h TD : 1h30 TP: 3 h Travail personnel :
Crédits et coefficients affectés à l'UE et à ses matières	UE : UF 1.2.2 crédits 11 Matière 1 : Exigences et Développement de Logiciels Crédits : 7 Coefficient : 4 Matière 2 : Analyse de la qualité du logiciel Crédits : 4 Coefficient : 2
Mode d'évaluation (continu ou examen)	Continu et examen
Description des matières	Exigences et Développement de Logiciels: L'ingénierie des exigences est la partie du génie logiciel permettant de déterminer quel système sera développé. A l'issue de ce module, l'étudiant pourra mettre en œuvre les différents processus d'ingénierie des besoins (exigences) nécessaires à la confection d'un cahier des charges cohérent. L'étudiant mettra en œuvre les concepts théoriques liés à l'ingénierie des exigences et le cycle de développement d'un logiciel à travers un mini-projet. Analyse de la qualité du logiciel: La qualité en logiciel est une notion complexe et globale qui regroupe beaucoup de facteurs. Le rôle et l'importance de chacun de ces facteurs sont parfois difficiles à discerner et à isoler. De plus, la qualité est une notion souvent suggestive qui est perçue différemment selon qu'on la regarde dans la perspective du client-utilisateur ou de l'ingénieur logiciel. Ce module permettra aux étudiants de comprendre et de pouvoir utiliser les concepts et les méthodes d'ingénierie de la

	qualité du logiciel ou encore de prendre une part active au sein d'une équipe d'assurance qualité.
--	--

Semestre : 02

UE : UEM 1.2.1

Répartition du volume horaire global de l'UE et de ses matières	Cours : 1h30 TD : TP: 1h30 Travail personnel :
Crédits et coefficients affectés à l'UE et à ses matières	UE : UEM 1.2.1 crédits : 4 Matière 1 : Techniques de Maintenance du logiciel Crédits : 4 Coefficient : 2
Mode d'évaluation (continu ou examen)	Continu

Semestre : 02
UE : UED 1.2.1

Répartition du volume horaire global de l'UE et de ses matières	Cours : 1h30 TD : TP: Travail personnel :
Crédits et coefficients affectés à l'UE et à ses matières	UE : UED 1.2.1 crédits : 2 Matière 1 : Pratique Professionnelle du GL Crédits : 2 Coefficient : 1
Mode d'évaluation (continu ou examen)	Continu
Description de la matière	Pratique Professionnelle du GL : Le développement de logiciel nécessite la coopération d'un ensemble de partenaire (l'utilisateur final, le chef du projet et les membres de l'équipe de développement), l'objectif du module est d'initier l'étudiant au travail de groupe et la technique de communication et la rédaction de rapports.

Semestre : 03
UE : UEF 2.1.1

Répartition du volume horaire global de l'UE et de ses matières	Cours : 3 h TD : TP: 3h Travail personnel :
Crédits et coefficients affectés à l'UE et à ses matières	UE : UEF 1.2.2 crédits 10 Matière 1 : Ingénierie Dirigée par les Modèles. Crédits : 4 Coefficient :3 Matière 2 : Architectures logicielles Crédits : 6 Coefficient : 4
Mode d'évaluation (continu ou examen)	Continu et examen
Description des matières	Architectures Logicielles : Ce cours a pour objet de présenter la notion d'architecture logicielle ainsi que son rôle tout au long du cycle de vie du logiciel. Les concepts introduits seront illustrés à travers de nombreux exemples de langages de description d'architecture. Par ailleurs, après avoir étudié les principaux concepts manipulés par l'architecte, un ensemble de pratiques permettant de spécifier, concevoir et évaluer les architectures est proposé. Une étude de cas suivie sera appliquée aux architectures étudiées, et les outils méthodologiques permettront d'en évaluer les possibilités, avantages et inconvénients. Ingénierie Dirigée par les Modèles : Ce cours présente les principes de l'ingénierie logicielle dirigée par les modèles. En particulier, le cours aborde le processus de développement logiciel par transformations de modèles et les concepts de modèles indépendants des plateformes

	et modèles spécifiques aux plateformes. Ce cours couvre aussi les principes et les standards de modélisation et de méta-modélisation, les langages spécifiques aux domaines et l'architecture dirigée par les modèles MDA (Model-Driven Architecture) de l'OMG (Object management group).
--	---

Semestre : 03
UE : UEF 2.1.2

Répartition du volume horaire global de l'UE et de ses matières	Cours : 3h TD : TP: 3h Travail personnel :
Crédits et coefficients affectés à l'UE et à ses matières	UE : UEF 2.1.2 crédits : 10 Matière 1 : SOA et Cloud Computing Crédits : 6 Coefficient : 4 Matière 2 : Administration des BDs Crédits : 4 Coefficient : 3
Mode d'évaluation (continu ou examen)	Contrôle de connaissance et examen
Description des matières	SOA et cloudcomputing: L'essor des Services Web accompagne naturellement la montée en puissance de SOA, qui fait l'objet aujourd'hui d'un réel engouement. En effet, les applications reposant sur des architectures SOA, qui requièrent des Web Services, assurent flexibilité et réactivité du système informatique. Les étudiants comprendront ce que sont les Web Services et leur fonctionnement ainsi que les technologies mises en œuvre dans le cadre des Services Web Administration dans les bases de données : Préparer des futurs informaticiens orientés conception et administration de base de données. L'accent est mis sur l'utilisation d'une méthodologie de conception de base de données centralisée ou répartie, les fonctions d'administration d'une base de données, la démarche d'optimisation d'une base de données, les règles d'évaluation du coût des opérations. Les défis modernes en matière d'architecture et leurs impacts sur les SGBD seront abordés.
Obs	Il est préférable de faire des sorties sur terrain dans la matière SOA et Cloud computing

Semestre : 03
UE : UEM 2.1.1

Répartition du volume horaire global de l'UE et de ses matières	Cours : 3h TD : 1h30 TP: 3h Travail personnel :
Crédits et coefficients affectés à l'UE et à ses matières	UE : UED 2.1.1 crédits : 10 Matière 1 : Programmation Orientée Agents Crédits : 6 Coefficient : 4 Matière 2 : Méthodes Agiles Crédits : 4 Coefficient : 3
Mode d'évaluation (continu ou examen)	Examen
Description des matières	Programmation Orientée Agents : La programmation orientée agents est l'évolution de la programmation orientée objets. Le but de ce cours est d'introduire la conception à base d'agents logiciels. Il étudie des méthodologies de développement et de la programmation orientées agents mais aussi des différents protocoles d'interactions entre agents afin de maximiser leur efficacité. Méthodes Agiles : L'approche agile s'est imposée depuis une dizaine d'années comme une alternative aux méthodes classiques de développement de logiciels. L'agilité permet de mieux répondre aux évolutions des besoins des clients en cours de projet. Son approche itérative et incrémentale permet de disposer d'un produit opérationnel dès les premières phases du développement. L'objectif de ce cours est de donner aux étudiants les compétences, concepts et pratiques agiles. Il constitue également une introduction aux différentes méthodes agiles (SCRUM, XP, LEAN).

L.3 - Programme détaillé par matière (1 fiche détaillée par matière)

Semestre : 1

UE : UEF 1.1.1

Intitulé de matière: Programmation Orientée Composants

Objectifs de l'enseignement

Ce cours comprend deux parties. La première vise à introduire les applications à base d'objets distribués, leurs structures et leurs constituants de base, en insistant sur la notion d'interface de l'objet distribué, proxy et référence d'objets. Cette partie met en évidence le rôle des intergiciels dans les applications distribuées. La deuxième concerne le développement d'applications à base de composants distribués. Elle présente quelques technologies de composants.

Connaissances préalables

- Concepts de base du GL
- Programmation orientée objet

Contenu de la matière :

Chapitre 1: Introduction aux applications distribuées

- 1.1 Introduction
- 1.2 Construction d'applications réparties en mode message
- 1.3 Construction d'applications réparties dans un environnement OO
- 1.4 Environnements d'exécution à objets répartis

Chapitre 2 : Mécanisme RPC

- 2.1 Motivation
- 2.2 Principe de base
- 2.3 Eléments d'un RPC

Chapitre 3 : Introduction à la POC

- 3.1 Définition de la POC
- 3.2 Rôles et intérêts de la POC
- 3.3 Définition d'un composant
- 3.4 Différences entre la POO et la POC

Chapitre 4: Les composants .NET

- 4.1 Le cadre .NET
- 4.2 Le modèle composant de .NET
- 4.3 Le modèle de connexion de .NET
- 4.4 Le modèle de déploiement de .NET
- 4.5 Visual Studio .NET
- 4.6 Exemples pratiques

Chapitre 5 : Les composants OSGI

- 5.1 L'infrastructure OSGI
- 5.2 Le modèle composant des OSGI
- 5.3 Le modèle de connexion OSGI

- 5.4 Le modèle de déploiement OSGI
- 5.5 Exemples pratiques

Chapitre 6 : Les composants Fractal

- 6.1 Introduction à l'infrastructure Fractal
- 6.2 Le modèle composant de Fractal
- 6.3 Le modèle connexion de Fractal
- 6.4 Le modèle de déploiement de Fractal
- 6.5 Exemples pratiques

Chapitre 7 : Modèle de composants orienté service (SCA)

- 7.1 Le modèle de composant SCA
- 7.2 Le modèle de connexion SCA
- 7.3 Outils d'implémentation du modèle SCA
- 7.4 Exemples pratiques

Mode d'évaluation : TP, Examen écrit

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

1. Intergiciel et Construction d'Applications Réparties, (19 janvier 2007)
Distribué sous licence Creative Commons :
<http://creativecommons.org/licenses/by-nc-nd/2.0/fr/deed.fr>
2. COMPONENT-ORIENTED PROGRAMMING, ANDY JU AN WANG KAI QIAN
Southern Polytechnic State University, Marietta, Georgia
A JOHN WILEY & SONS, INC., PUBLICATION, 2008.
3. Tuscany SCA in action, simon Laws, Mark Combella, Raymond Feng, Halen Mahbod, Simon Nash. Manning Publications, 2011.

Semestre : 01

UE : UEF 1.1.2

Intitulé de matière: Programmation Logique et Fonctionnelle

Objectifs de l'enseignement

En s'appuyant sur la programmation déclarative, l'objectif de ce cours est d'exposer les techniques de base de l'Intelligence Artificielle. Situer l'Intelligence Artificielle en tant que discipline scientifique, Maîtriser les concepts de base de l'intelligence artificielle.

Connaissances préalables recommandées

Les connaissances algorithmiques et éléments de logiques acquises en cours de formation « licence »

Contenu de la matière :

Ce cours comprend deux parties.

Partie 1 : cette première partie vise à étudier les paradigmes de programmation fonctionnelle et logique. Pour chacun, on y présente d'abord les concepts de base (fonction, réduction, évaluation, pour la programmation fonctionnelle, logique de Horn, stratégie de sélection, négation) avant d'aborder des sujets plus avancés, comme les fonctions d'ordre supérieur, les types polymorphes, les monades ou encore quelques exemples de programmation non déterministe. Une attention particulière est aussi portée aux techniques de programmation.

Partie 2 : la deuxième partie aborde les techniques de base de l'Intelligence Artificielle. Après une introduction à ce domaine ainsi que l'exposé de concepts fondamentaux au travers d'un exemple simple de système expert, cette deuxième partie s'articule autour de trois parties:

- i. étude des fonctionnalités et des composants d'un système expert,
- ii. techniques de représentation des connaissances,
- iii. techniques de résolution de problèmes.

L'accent est, en particulier, mis sur l'utilisation de la logique et de la programmation par contraintes. On y développe enfin les bases de l'apprentissage automatique.

Références

H. Farreny, M. Ghallab. *Eléments d'Intelligence Artificielle*. Hermès, 1985.

J.-P. Haton, N. Bouzid, F. Charpillet, M.-C.Haton, B. et H. Lâasri, P. Marquis, T. Mondot, A. Napoli. *Le Raisonnement en Intelligence Artificielle - Modèles, Techniques et Architectures pour les Systèmes à Bases de Connaissances*. InterEditions, 1991.

J.-L. Laurière. *Intelligence Artificielle :Résolution de Problèmes par l'Homme et la Machine*. Eyrolles, Paris, 1986.

Robert W. Sebesta, " Concepts of Programming Languages", Addison Wesley 7th Edition, 2006.

Semestre :1

Unité d'enseignement : UEF 1.1.2

Matière : Sécurité des Réseaux

Objectifs de l'enseignement :

L'étudiant est censé acquérir une maîtrise globale de la sécurisation d'un réseau privé d'entreprise utilisant les technologies Internet/intranet, et de son interconnexion avec des réseaux extérieurs, ainsi cette matière doit offrir à l'étudiant:

- Initiation aux principes de sécurité des réseaux informatiques en termes de méthodes, pratiques et politiques.
- Présentation des dispositifs logiciels et matériels qui servent à assurer la sécurité des ressources en informatique.
- Évaluation des risques d'attaque, et estimation du niveau de vulnérabilité et proposition de solutions de protection du réseau.

Connaissances préalables recommandées :

- Principes fondamentaux des réseaux de communication TCP/IP,
- Principes fondamentaux de l'administration des réseaux en termes d'architectures de services.

Contenu de la matière :

1. Introduction à la sécurité des réseaux

- 1.1. Définition des principaux concepts de sécurité informatique.
- 1.2. Etat des lieux de la sécurité informatique.
- 1.3. Types de menaces et d'attaques Réseau
 - 1.3.1. Attaques "couches basses".
 - 1.3.2. Attaques applicatives.
 - 1.3.3. Déni de service
 - 1.3.4. Vers, virus, trojans, malwares et keyloggers.

2. Sécurisation d'un système ("Hardening")

- 2.1. Insuffisance des installations par défaut.
- 2.2. Critères d'évaluation
- 2.3. Sécurisation d'Unix-Linux.
- 2.4. Sécurisation de Windows.

3. Sécurisation des Données ou Cryptographie.

- 3.1. Cryptographie
 - 3.1.1. Introduction à la cryptographie (protocoles et techniques).
 - 3.1.2. Chiffrements symétrique et asymétrique.
 - 3.1.3. Fonctions de hachage.
 - 3.1.4. Services cryptographiques.
- 3.2. Authentification de l'utilisateur.

- 3.3. Protocole SSL/TLS,
- 3.4. Protocole SSH.
- 3.5. Signature électronique (protocole LDAP)

4. Politique et Architecture de sécurité (normes et standards).

- 4.1. Quelles architectures pour quels besoins ?
 - 4.1.1. Plan d'adressage sécurisé : RFC 1918.
 - 4.1.2. Translation d'adresses
 - 4.1.3. Rôle des zones démilitarisées (DMZ).
- 4.2. Virtualisation.
- 4.3. Firewall
 - 4.3.1. Actions et limites
 - 4.3.2. Evolution technologique des firewalls.
 - 4.3.3. Proxy serveur et relais applicatif,
 - 4.3.4. Relais SMTP.
- 4.4. IPSec
 - 4.4.1. Présentation du protocole.
 - 4.4.2. Modes tunnel et transport.
 - 4.4.3. ESP et AH.

5. Moyens d'Analyse et de Détection des Attaques

- 5.1. Tests d'intrusion : outils et moyens.
- 5.2. Détection des vulnérabilités (scanners, sondes IDS, etc.).
- 5.3. Outils de détection temps réel IDS-IPS, agent, sonde ou coupure.
- 5.4. Supervision et administration.
- 5.5. Veille technologique.

6. Mécanismes de la sécurité des réseaux sans fil (Facultatif)

- 6.1. Sécurité Wi-Fi.
- 6.2. Risques inhérents aux réseaux sans fil.
- 6.3. Types d'attaques.
- 6.4. Limites du WEP.
- 6.5. Protocole WPA et WPA2.
- 6.6. Attaque Man in the Middle avec le rogue AP.

Travaux pratiques (à titre indicatif)

- Installation et utilisation de l'analyseur réseau Wireshark ou autre à travers GNS3.
- Exemple de sécurisation d'un système Linux.
- Mise en œuvre d'une attaque type ARP.
- Mise en œuvre d'une attaque applicative.
- Mise en œuvre d'un proxy Cache/Authentification.
- Déploiement d'un relais SMTP et d'un proxy HTTP/FTP Antivirus.
- Mise en œuvre d'IPSec mode transport/PSK.

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

Contrôle de connaissances + contrôle TP + évaluation continue

Références bibliographiques :

- Solange Ghernaouti-Hélie, « Sécurité Informatique et Réseaux », 2006, Dunod ISBN-2100074482.
- Raymond Panko « Sécurité des Systèmes d'information et des Réseaux » , Pearson Education Stallings, W. Network Security Essentials, 2^{ième} edition. Prentice Hall, 2003
- Eric Maiwald, « Network Security A Beginner's Guide », 3^{ième} Edition, Publisher: McGraw-Hill Osborne Media. 2012.

Semestre : 01

UE : UEF 1.1.2

Intitulé de matière: Systèmes Distribués

Objectifs de l'enseignement

Acquérir les concepts de base concernant la structure et le fonctionnement d'un système distribué, tous les mécanismes de partage d'information et de communication doivent être étudiés.

Connaissances préalables recommandées

- Systèmes d'exploitation
- Systèmes centralisés
- Programmation Java

Contenu de la matière :

Chapitre I : Introduction aux systèmes distribués

1. Définition structurelle d'un système distribué
2. Motivation des systèmes distribués
 - 2.1. Avantages des systèmes distribués/systèmes centralisés
 - 2.2. Avantages des systèmes distribués/ordinateurs personnels
 - 2.3. Désavantage des systèmes distribués

Chapitre II : Systèmes Distribués : Définition et caractéristiques

1. Définition d'un système Distribué
2. Caractéristiques d'un système distribué
 - 2.1. Mécanisme de communication
 - 2.2. Structure globale de nommage
 - 2.3. Image unique
 - 2.4. Partage des organes de stockages
 - 2.5. Temps de réponse
3. Différences des systèmes distribués/ systèmes centralisés
 - 3.1. Recouvrement
 - 3.2. Sécurité
 - 3.3. Temps de réponse
 - 3.4. Croissance incrémentale

Chapitre III : Architecture des systèmes distribués

1. Notion de client/serveur
2. Familles des systèmes distribués
 - 2.1. Monolithique noyau
 - 2.1.1. Première approche
 - 2.1.2. Deuxième approche
 - 2.2. Micro noyau

Chapitre IV : Système de fichiers dans les systèmes distribués

1. Systèmes de fichiers distribués

- 2. Classe des systèmes de fichiers distribués
- 3. Composantes d'un système de fichiers distribués
 - 3.1. Service de répertoire
 - 3.2. Communication
 - 3.3. Service de fichiers
- 4. Service de répertoires
 - 4.1. Système hiérarchique
 - 4.1.1 Image unique
 - 4.1.2. Multiple images
 - 4.1.3. Systèmes hiérarchiques dans les systèmes d'exploitation distribués
 - 4.1.4. Systèmes hiérarchiques dans les systèmes distribués coopératifs
 - 4.2. Spécification des fichiers distants
 - 4.2.1. Transparence de réseau
 - 4.2.2. Transparence de location
 - 4.2.3. Indépendance de location
 - 4.3. Binding (liaison)
 - 4.3.1. Serveur de noms
 - 4.3.1.1. Distribution du serveur de noms
 - 4.3.1.2. Partitionnement
 - 4.3.1.3. Duplication
 - 4.3.1.4. Approche hybride

Chapitre V : Mécanismes de communication dans les systèmes distribués

- 1. Communication dans les systèmes distribués
- 2. Eléments de la communication
 - 2.1. Service Distant
 - 2.2. Caching
- 3. Service Distant
 - 3.1. Pourquoi pas OSI
 - 3.2. Simple protocole
 - 3.3. RPC
 - 3.3.1. Transparence
 - 3.3.2. Protocole de transfert de données
 - 3.3.3. Sémantique
- 4. Caching
 - 4.1. Objectif
 - 4.2. Localisation du cache
 - 4.3. Politique de propagation de modifications
 - 4.4. Validation du cache
 - 4.5. Sémantique de partageabilité

Chapitre VI : Besoins d'un service de fichiers

- 1. Service de fichiers
- 2. Caractéristiques désirables d'un service de fichiers
 - 2.1. Disponibilité
 - 2.2. Sécurité
 - 2.3. Croissance incrémentale
- 3. Structures de serveurs
- 4. Types de serveurs & tolérances aux pannes

Mode d'évaluation : *Examens , contrôle continu*

Contrôle * 2 + interrogation

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

- Distributed operating systems, Andrew S. Tanenbaum, Pearson Education, 1995.
- Distributed operating systems: concepts and design, Pradeep K. Sinha, PHI Learning, Pvt, Ltd, 1998.

Semestre : 01

UE : UEM 1.1.1

Intitulé de matière: Statistiques et Analyse de données

Objectifs de l'enseignement

Il est tout à fait courant qu'après un certain traitement ou fonctionnement d'un système on se trouve en face d'une masse importante de données recueillies qu'on a besoin d'analyser. L'analyse des données est un domaine scientifique très large qui comporte des techniques variées principalement empruntées au domaine de la statistique et l'apprentissage automatique. Outre la préparation et le nettoyage des données, elle vise à mettre en exergue les relations qui peuvent exister entre les données, à résumer les données et / ou à établir des prédictions. Cette matière a pour objectif de permettre à l'étudiant de se familiariser avec le domaine de l'analyse de données et d'acquérir une certaine maîtrise des techniques et outils permettant d'analyser les données. L'étudiant aura l'occasion d'apprendre comment appréhender un tableau de données et à choisir et appliquer la technique appropriée pour répondre à un besoin d'analyse précis compte tenu de la nature des données disponibles.

Connaissances préalables recommandées

Ce cours suppose que l'étudiant maîtrise le contenu du module probabilités et statistique vu en L1 licence.

Contenu de la matière

Chapitre 1 : Rappels des concepts de base de probabilités et statistiques

- 1.1** Introduction : tableau de données, attributs, type d'attributs, les données d'un point de vue algébrique et géométrique.
- 1.2** Statistique descriptive
- 1.3** Probabilité et variables aléatoires
- 1.4** Distributions de probabilités usuelles
- 1.5** Tests d'hypothèses
- 1.6** Outils de visualisation

Chapitre 2 : Modèles descriptifs pour l'analyse des données : clustering

- 2.1** La classification non supervisée (clustering) des données
- 2.2** L'algorithme k-means
- 2.3** Le clustering hiérarchique
- 2.4** Le clustering basé densité : DBSCAN
- 2.5** Outils pour le clustering

Chapitre 3 : Modèles descriptifs pour l'analyse des données : méthode de réduction de dimensionnalité

3.1 Introduction

3.2 Analyse en composantes principales

3.3 Analyse factorielle des correspondances

3.3 Analyse factorielle des correspondances multiples

3.4 L'analyse discriminante

Chapitre 4 : Modèles prédictifs pour l'analyse des données : régression

4.1 Introduction

4.2 Modèle de régression linéaire simple

4.3 Modèle de régression non linéaire simple

4.4 Modèle de régression multiple

4.5 Outils pour la régression

Chapitre 5 : Modèles prédictifs pour l'analyse des données : classification supervisée

5.1 Introduction à la classification supervisée

5.2 Classification Bayésienne

5.3 Réseaux de neurones

5.4 Classification par machines à vecteurs de support

5.5 Outils pour la classification supervisée

Mode d'évaluation :

$$\text{Contrôle} * 2 + (\text{Interrogation} + \text{TP})/2$$

Références bibliographiques :

1. Ludovic Lebart, Marie Piron et Alain Morineau, Statistique Exploratoire multidimensionnelle, Paris, Dunod, (7-049616-10-2-978ISBN) ,2006 .
2. Stéphane Tufféry, Data Mining et statistique décisionnelle, Paris, éditions Technip, (3-0946-7108-2-978ISBN) ,2010
3. Härdle, W. Simar L. (2007) Applied Multivariate Statistical Analysis, 2nd edition, Springer.
4. Bach, F.R., Jordan, M.I., (2006). A probabilistic interpretation of Canonical Correlation Analysis. Univ. California, Berkeley.
5. G. Saporta, (2005). Probabilités, analyses des données et statistiques.

Semestre : 2

Unité d'enseignement : UEF 1.2.1

Matière : Spécification et vérification des systèmes

Objectifs de l'enseignement

L'objectif de ce cours est de doter l'étudiant de techniques de spécification et de vérification formelles utilisables dans le contexte des systèmes répartis caractérisés par des composants distincts, chacun d'eux s'exécutant potentiellement sur un processeur différent.

Connaissances préalables recommandées

Génie logiciel (Licence), Systèmes Distribués (S1)

Contenu de la matière :

Chapitre1- Généralités sur les méthodes formelles

1. Utilité des méthodes formelles en GL
2. Concepts de base (ensemble, prédicat, fonction, foncteur, équivalence, abstraction...)

Chapitre2- Spécification des systèmes

1. Notion de spécification
2. Types de spécification (axiomatique, opérationnelle, ...)
3. Réseaux de Petri (simples, étendus, de haut niveau)
4. Algèbre de processus (CCS, π -calcul)
5. Systèmes de transitions étiquetés (LTS)

Chapitre 3- Propriétés à Vérifier

1. Types de Propriétés
Atteignabilité, Sureté, Vivacité, Équité, Absence de Blocage, Terminaison, ...
2. Spécification des propriétés
 - Logique des prédicats
 - Logique Temporelle

Chapitre4- Vérification des systèmes

1. Notion de Vérification/validation
2. Types de Vérification
 - Formelle : Basées sur les preuves de théorèmes, Basées sur les équivalences, Basées sur le model-checking
 - Par Simulation
 - Par émulation

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

(Interrogation +TP + 2*contrôle)/4

Références bibliographiques:

- Michael Huth, Mark Ryan, “Logic in Computer Science Modeling and Reasoning about Systems” (2nd Edition), 2004.
- Joost-Pieter Katoen, “Principles of model-checking”, Lecture Notes, 2004.
- Henri Habrias and Marc Frappier” *Software Specification Methods*”, ISTE Ltd, 2006, [ISBN 1-905209-34-7](#).
- Jean-François Monin “*Understanding Formal Methods*”, Springer, 2003, [ISBN 1-85233-247-6](#).

Semestre : 02

UE : UEF 1.2.1

Intitulé de matière: Modélisation et Evaluation des Systèmes Informatiques

Objectifs de l'enseignement

L'objectif de ce module est de sensibiliser les étudiants aux problèmes de modélisation et d'évaluation des performances des systèmes réels. A l'issue de ce module, l'étudiant sera capable de poser un modèle représentant le système étudié et de sélectionner, parmi les méthodes analytiques et de simulation, la plus appropriée.

Connaissances préalables recommandées

Ce cours suppose que l'étudiant maîtrise le contenu du module Statistiques et Analyse des données de M1.

Contenu de la matière

Chapitre 1 : Introduction à la modélisation et à l'évaluation de performances des systèmes

- 1.1 Introduction
- 1.2 La modélisation des réseaux
- 1.3 Problème de la caractérisation et prévision de la charge
- 1.4 La qualité de service dans la conception des réseaux
- 1.5 Les solutions possibles pour dimensionner un réseau multiservice

Chapitre 2 : Le formalisme Files d'attentes

- 2.1 La station de service de base
- 2.2 Processus des arrivées
- 2.3 Processus des services
- 2.4 Les chaînes de Markov
- 2.5 La file M/M/1 et la file M/M/C
- 2.6 Les réseaux de files d'attente

Chapitre 3 : La différenciation de services dans les réseaux

- 3.1 Introduction
- 3.2 Le modèle DiffServ
- 3.3 Le traitement différencié de paquet dans le routeur DiffServ
- 3.4 Intégration MPLS/Diffserv

Chapitre 4 : Evaluation de performance de DiffServ

- 4.1 Utilité des techniques de simulation
- 4.2 La simulation par événements
- 4.3 Estimation de la précision
- 4.4 Le simulateur NS
- 4.5 Implémentation de DiffServ dans NS
- 4.6 Résultats et interprétations : Equité de la bande passante et taux de perte

Chapitre 5 : Le simulateur NS (Travaux Pratiques)

5.1 Présentation générale du simulateur NS2

5.2 Téléchargement et installation de NS2

5.3 Exemples de simulation sous NS2 :

- Exemple 1 : Simulation d'un réseau à deux nœuds
- Exemple 2 : Simulation de la fonction de routage et visualisation de la file d'attente
- Exemple 3 : Simulation d'un réseau ad hoc pour l'évaluation des performances d'un protocole de routage

Mode d'évaluation : *TP + Examens écrits.*
(Contrôle * 2 + Interrogation + TP)/4

Références bibliographiques :

- Frikha, M., (2007). Planification et simulation des réseaux. Hermès Science Publication.
- Melin, J.-L., (2002). Qualité de services sur IP. Eyrolles.
- Baynat, B., (2000). Théorie des files d'attente. Hermès Science Publication.

Semestre : 02

UE : UEF 1.2.2

Intitulé de matière: Exigences et Développement de Logiciel

Objectifs de l'enseignement

L'ingénierie des exigences est la partie du génie logiciel permettant de déterminer quel système sera développé. A l'issue de ce module, l'étudiant pourra mettre en œuvre les différents processus d'ingénierie des besoins (exigences) nécessaires à la confection d'un cahier des charges cohérent. Ces processus seront appliqués à un projet logiciel

Connaissances préalables recommandées

- Connaissances sur UML et le processus unifié
- Bases de la gestion de projet

Contenu de la matière :

Introduction

Gestion de projet et exigences logicielles
Crise du logiciel et génie logiciel
Enjeux de l'ingénierie des exigences

Processus d'ingénierie des exigences

Analyse du domaine
Elucidation des exigences et évaluation des exigences
Documentation des exigences
Consolidation des exigences

Classification des exigences

Exigences fonctionnelles
Exigences non fonctionnelles

Techniques de l'ingénierie des exigences

Etude de documents, interview, questionnaire
Scénarios d'interaction
Prototypage
Revue des exigences
Validation des modèles
Tests d'acceptation
Gestion du changement

Modélisation des exigences

Modélisation des objectifs, des objets, des agents
Modélisation des opérations et des comportements

Mode d'évaluation : contrôle long +contrôle court

Références

www.standishgroup.com

www.swebok.org/index.html

Semestre : 02

UE : UEF 1.2.2

Intitulé de matière: Analyse de la qualité du logiciel

Objectifs de l'enseignement

La qualité en logiciel est une notion complexe et globale qui regroupe beaucoup de facteurs. Le rôle et l'importance de chacun de ces facteurs sont parfois difficiles à discerner et à isoler. De plus, la qualité est une notion souvent suggestive qui est perçue différemment selon qu'on la regarde dans la perspective du client-utilisateur ou de l'ingénieur logiciel.

Ce module permettra aux étudiants de comprendre et de pouvoir utiliser les concepts et les méthodes d'ingénierie de la qualité du logiciel ou encore de prendre une part active au sein d'une équipe d'assurance qualité.

Contenu de la matière

Chapitre 1: Concepts de base

- Exigences qualité
- Modèles de qualité
- Mesures de qualité
- Évaluation de qualité

Chapitre 2: Modèle de la structure de la qualité du produit logiciel

- Le chemin de décomposition des exigences haut niveau aux catégories des mesures qualité
- Analyse de la traçabilité
- Impact des exigences fonctionnelles sur les exigences qualité

Chapitre 3: Formalisation des exigences qualité

Chapitre 4: Modèle d'implantation de qualité

Chapitre 5: Processus d'ingénierie de qualité

Mode d'évaluation : contrôle long +contrôle court

Références

- Suryn W. "Maturing Usability: Quality in Software, Interaction and Value. Part 1: Software Quality Engineering – the leverage for gaining maturity". Springer Publisher, December 14th, 2007
- Lauesen S ., "Software Requirements". Addison-Wesley, 2002
- Vliet, H., "Software Engineering, Principles and Practice", Second Edition. John Wiley& Sons, 2002
- McGarry, J et al, "Practical Software Measurement". Addison-Wesley, 2002
- Kan S., H., "Metrics and Models in Software Quality Engineering", Second Edition, Addison-Wesley 2003

Semestre : 02

UE : UED 2.1.1

Intitulé de matière: Techniques de maintenance du logiciel

Objectifs de l'enseignement

Ce module vise à présenter des techniques de maintenance du logiciel : Compréhension de programmes, Reengineering, Reverse engineering, Migration, Retirement.

Contenu de la matière :

Ce cours présente les différentes techniques de maintenance du logiciel :

- Compréhension de programmes
- Reengineering
- Reverse engineering
- Migration
- Retirement

Mode d'évaluation : *TP + Examens écrits.*

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

Semestre : 02

UE : UEM 2.1.1

Intitulé de matière: Pratique Professionnelle du GL

Objectifs de l'enseignement :

Le développement de logiciel nécessite la coopération d'un ensemble de partenaire (l'utilisateur final, le chef du projet et les membres de l'équipe de développement), l'objectif du module est d'initier l'étudiant au travail de groupe et la technique de communication et la rédaction de rapports.

Connaissances préalables recommandées :

Connaissances basiques en techniques d'expression et de communication
Navigation sur Internet

Contenu de la matière :

- Problématique méthodologique de la recherche scientifique
- Etapes du processus de la recherche
- Méthodes et validité de la recherche
- Recherche de l'information scientifique
- Différentes étapes d'un projet de recherche
- Rédaction scientifique et techniques d'exposés
- Déontologie de la recherche

Mode d'évaluation : *examen écrit et/ou exposés*

Références

Sites

- aeris.11vm-serv.net
- csidoc.insa-lyon.fr/sapristi/
- etc....