



Ecole Nationale  
Supérieure  
d'Ingénieurs  
de Tunis

Département de l'Informatique



الجمهورية التونسية  
وزارة التعليم العالي والبحث العلمي  
جامعة تونس  
المدرسة الوطنية العليا للمهندسين بتونس  
قسم الإعلامية

# ENSIT

Ecole Nationale Supérieure des Ingénieurs de Tunis

Département De l'Informatique

## PROGRAMME DE LA FORMATION D'INGENIEURS EN GENIE INFORMATIQUE

*Extrait de quelques Fiches modules*

2021-2022



## Table des matières

<b>1. Programme détaillé : Semestre 1 .....</b>	<b>1</b>
INF1-1-1 Mathématiques de l'ingénieur .....	2
INF1-1-2 Analyse numérique .....	4
INF1-2-2 Composants & systèmes logiques .....	9
INF1-2-3 Bases de données .....	11
INF1-3-1 Algorithmique et structure de données (ASD) .....	13
INF1-3-2 Programmation 1 .....	16
INF1-4-1 Introduction aux réseaux informatiques .....	19
INF1-4-2 Système d'exploitation 1 .....	22
INF1-5-1 Economie générale .....	24
INF1-6-1 Techniques de communications écrites 1 .....	26
 <b>2. Programme détaillé : Semestre 2 .....</b>	 <b>28</b>
INF1-1-5 Recherche opérationnelle .....	29
INF1-2-4 Technologies Web 2 .....	31
INF1-2-5 Architecture des Ordinateurs .....	33
INF1-2-6 Théorie des Langages et des Automates .....	35
INF1-2-7 Atelier Bases de Données .....	38
INF1-3-3 Théorie des Graphes .....	40
INF1-3-4 Programmation 2 .....	43
INF1-4-3 Réseaux Locaux .....	46
INF1-4-4 Système d'exploitation 2 .....	49
INF1-6-3 Techniques de communications orales 1 .....	51
 <b>3. Programme détaillé : Semestre 3 .....</b>	 <b>53</b>
INF2-1-1 Algorithmique Avancé .....	54
INF2-1-2 Compilation .....	56
INF2-1-3 Programmation 3 (Java) .....	59
INF2-3-1 Sécurité des réseaux .....	62
INF2-3-2 Les réseaux TCP/IP .....	64
INF2-3-3 Administration Système .....	66
INF2-4-2 Génie Logiciel .....	68
 <b>4. Programme détaillé : Semestre 4 .....</b>	 <b>70</b>
INF2-1-4 Programmation Mobile .....	71
INF2-1-5 Interface Homme-Machine .....	74
INF2-1-6 Intelligence Artificielle .....	76
INF2-2-2 Atelier JEE .....	78
INF2-2-3 Programmation système sous Unix .....	80
INF2-4-4 Architecture Orientée Services (AOS) .....	82
 <b>5. Programme détaillé : Semestre 5 .....</b>	 <b>85</b>
INF3-X-1-1 Nouvelles Architectures .....	86

<b>INF3-X-1-2 Linux embarqué .....</b>	<b>89</b>
<b>INF3-X-1-3 Programmation temps réel .....</b>	<b>92</b>
<b>INF3-X-1-4 Cloud Computing .....</b>	<b>94</b>
<b>INF3-X-2-4 Gestion de projet .....</b>	<b>96</b>
 <b>Option 1 : GL et Informatique Décisionnelle (GLID) .....</b>	 <b>98</b>
<b>INF3-GLID-3-1 Data mining .....</b>	<b>98</b>
<b>INF3-GLID-4-3 Big Data.....</b>	<b>101</b>
 <b>Option 2 : Nouvelles Technologies et Sécurité (NTS) .....</b>	 <b>104</b>
<b>INF3-NTS-3-3 Internet of Things (IoT) .....</b>	<b>104</b>
<b>INF3-NTS-4-1 Réseaux et applications Multimédia .....</b>	<b>106</b>

## 1.PROGRAMME DETAILLE : SEMESTRE 1

Code	Module	Volume horaire
INF1-1-1	Mathématiques de l'ingénieur	30
INF1-1-2	Analyse numérique	30
INF1-1-3	Logique formelle	30
INF1-2-1	Technologies Web 1	30
INF1-2-2	Composants et Systèmes logiques	30
INF1-2-3	Base de données	30
INF1-3-1	Algorithmique et structure de données	60
INF1-3-2	Programmation 1 (langage C)	60
INF1-4-1	Introduction aux réseaux informatiques	45
INF1-4-2	Systèmes d'exploitation 1	45
INF1-5-1	Economie générale	15
INF1-5-2	Semaine Logiciels 1	15
INF1-6-1	Techniques de communications écrites 1	15
INF1-6-2	Anglais 1	15

# INF1-1-1 Mathématiques de l'ingénieur

Groupe de module	INF1-1		Intitulé du module	Mathématiques de l'ingénieur		Code	INF1-1-1	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	x	Régime	CC		Semestre	1	x
	TD	x					2	
	TP			Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		30h 22,5hC et 7,5h TD			Coefficient	2		
					Crédits	2		
Responsable du Module				Intervenants	Dorra Drissi			
<b>Acquis d'apprentissage</b>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : Intégrer au sens de Lebesgue et Riemann Calculer des transformées de Fourier et de Laplace. Calculer les origines Résoudre des équations différentielles en utilisant les transformées de Laplace et de Fourier.						
<b>Contenu</b>		<b>Cours</b> <ul style="list-style-type: none"><li>• Intégrale de Lebesgue</li><li>• Espace <math>L^p</math></li><li>• Convolution et régularisation</li><li>• Transformée de Fourier</li><li>• Transformée de Laplace (Application : transformée en Z)</li><li>• Espace de Hilbert</li><li>• Base Hilbertiennes</li><li>• Distributions</li><li>• Dérivées des distributions</li><li>• Formulation variationnelle</li></ul>						
<b>Méthode d'enseignement et d'apprentissage</b>		face à face (présentiel)						
<b>Technique d'enseignement</b>		Cours 22,5h et TD 7,5 h						
<b>Méthodes d'évaluation</b>		Ecrit Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.  L'épreuve écrite de mi-parcours sera consacrée à l'évolution du niveau d'atteinte des acquis d'apprentissage 1 et 2.  L'épreuve écrite finale est consacrée à l'évolution du niveau d'atteinte des acquis d'apprentissage 2,3 et principalement 4.						
<b>Critères d'évaluation</b>		Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à intergrer au sens de Lebesgue et calculer des transformées de Laplace et de Fourier. Au terme du test final, les étudiants devront résoudre des EDP, écrire des formulations variationnelles et étudier l'existence et l'unicité de la solution de quelques problèmes.						
<b>Mesure d'évaluation</b>		Attribution d'une moyenne finale						

<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi parcours (D.S) :1/3</p> <p>Test écrit final (Examen): 2/3</p>
<b><u>Pré requis</u></b>	Une bonne maîtrise de programme et d'analyse d'algèbre des années préparatoires
<b><u>Acquis de formation visés</u></b>	PLO1
<b><u>Références Bibliographiques</u></b>	<p>Initiation aux distributions et aux transformations intégrales</p> <p>Par D. Euvrard avec collaboration de M-Minoux. ENSTA Cours IA 14</p> <p>EDITION 1980</p> <p>Intégration</p> <p>Exercices et corrigés en complément du cours</p> <p>De Gilles Pagés – Jacque Fréjoz Université de pierre et Marie Curie – 2004-2005</p> <p>Techniques mathématiques pour la physique</p> <p>Guy Auliac, Jean Avignant, Elie Azoulay.</p> <p>Ellipses édition 2000</p> <p>ISBN 2 – 7298 -0041 -7</p> <p>Mathématiques Appliquées pour la maîtrise</p> <p>PG – Ciarlet, J –L Lions. Masson, Paris 1983</p> <p>ISBN 2- 2225 - 771987</p>

## INF1-1-2 Analyse numérique

Groupe de module	INF1-1		Intitulé du module	Analyse Numérique		Code	INF1-1-2	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	x	Régime	CC		Semestre	1	x
	TD	x					2	
	TP			Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		30h 22,5h C, 7,5h TD			Coefficient	2		
					Crédits	2		
Responsable du Module				Intervenants	Dorra Drissi			
Acquis d'apprentissage		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : 1. Résoudre des systèmes linéaires et non linéaires 2. Résoudre numériquement des EDP 3. Faire des interpolations polynomiales 4. Intégrer numériquement.						
Contenu		<b>Cours</b> <ul style="list-style-type: none"><li>• Introduction et rappel d'algèbre linéaire</li><li>• Méthodes directes</li><li>• Méthodes itératives</li><li>• Méthode du gradient</li><li>• Calcul des valeurs propres</li><li>• Résolution d'une équation non linéaire</li><li>• Interpolation</li><li>• Intégration</li><li>• Résolution numérique des équations différentielles linéaires</li><li>• Résolution numérique des équations non-linéaires</li></ul>						
Méthode d'enseignement et d'apprentissage		face à face (présentiel)						
Technique d'enseignement		Cours 22.5 h et TD 7.5h						
Méthodes d'évaluation		Ecrit et pratique. Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus. <ul style="list-style-type: none"><li>• L'épreuve écrite de mi-parcours sera consacrée à l'évolution du niveau d'atteinte des acquis d'apprentissage 1 et 2.</li><li>• L'épreuve écrite finale est consacrée à l'évolution du niveau d'atteinte des acquis d'apprentissage 2,3 et principalement 4.</li></ul>						
Critères d'évaluation		Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à résoudre des systemes linaires et non linaires, ecrire des algorithms. Au terme du test final, les étudiants devront résoudre numeriquement des EDP linaires et non linaires et écrire des schemas numériques.						



<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum ) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Test écrit à mi parcours (D.S) : 1/3 Test écrit final (Examen) : 2/3
<b><u>Pré requis</u></b>	Une bonne maitrise de programme et d'analyse d'algèbre des années préparatoires.
<b><u>Acquis de formation visés</u></b>	PLO1
<b><u>Références Bibliographiques</u></b>	<ol style="list-style-type: none"> <li>1. Analyse numérique linéaire Cours ENIT élaboré par : H.Ateb , H.Bouhafa, H.Cheker 2008-2009</li> <li>2. Analyse numérique non linéaire Cours ENIT élaboré par : H.Ateb , H.Bouhafa, H.Cheker 2008-2009</li> <li>3. Analyse numérique avec MATLAB Exercice et problème Jean -louis- Merrien. DUNOD ISBN – 6647747</li> <li>4. Calcul scientifique Cours, exercices corrigés et illustration en MATLAB et OCTAVE Alifio Quarteroni, Fausto Saleri, Paola Gervasio 2eme édition – springer 2010 ISBN – 978 -88 -470 -1675 -0</li> </ol>

## INF1-2-1 Technologies Web 1

Groupe de Module	INF1-2		Module	Technologie web 1		Code	INF1-2-1	
Département	Informatique		Filière	Génie Info				
Type d'enseignement	C	x	Régime	CC	x	Semestre	1	x
	TD						2	
	TP	x		Mixte			3	
	Proj						4	
							5	
Volume horaire/semaine		Présentiel : C : 15h, TP:15h Non présentiel : 10			Coefficient		2	
					Crédit		2	
Responsable du Module		BEN RGAYA Houda		Enseignant(e)s intervenants	BEN RGAYA Houda			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : 1. Comprendre l'évolution du web et des langages de marquage 2. Étudier les aspects de structure, de style et traitement dans un site web 3. Se familiariser avec les langages HTML 4. Comprendre et utiliser les feuilles de style CSS 5. Se familiariser avec les outils d'intégration des objets multimédia 6. Développer des sites Web statiques grâce à aux technologies HTML 7. Étudier les éléments importants de programmation du côté du client à l'aide de langages de scripts tels que JavaScript 8. Analyser et manipuler le Document Model Object (DOM) 9. Appliquer les concepts appris dans des cas pratiques afin d'introduire de la dynamicité au site web en utilisant la librairie jQuery 10. Développer et réaliser des jeux simples pour le Web						
<u>Contenu</u>		<b>Cours</b> 1. Introduction au Web : • Historique d'Internet et du Web Internet, Le modèle client- serveur. 2. Structure d'un document HTML : • Généralités • Principe de balisage • Les principales balises HTML (Formatage de texte, Mise en page, tableaux) • Les notions de url et de liens • Les Frames • Les Formulaires 3. Les feuilles de styles CSS • Principe des sélecteurs et des règles • Différents types de sélecteur • Notion de mise en page • Utilisation de Bootstrap 4. Les langages de script côté client • Introduction • Console						

	<ul style="list-style-type: none"> <li>• Les bases de Javascript (Syntaxe et Variables, Nombres et Chaînes de caractères, Opérateurs, Commentaires)</li> <li>• Les structures conditionnelles et itératives</li> <li>• Fonctions</li> <li>• Les Objets : les propriétés et méthodes prédéfinies (Date, Array, String,...)</li> </ul> <p>5. Le Document Object Model DOM</p> <ul style="list-style-type: none"> <li>• <u>Présentation du DOM HTML et de ses APIs accessibles en JavaScript</u></li> <li>• <u>Accéder aux éléments dans un document avec JavaScript et modifier leur contenu</u></li> <li>• <u>Naviguer ou se déplacer dans le DOM</u></li> <li>• <u>Ajouter, modifier ou supprimer des éléments du DOM</u></li> <li>• <u>Manipuler les attributs et les styles des éléments via le DOM</u></li> <li>• <u>La gestion d'événements en JavaScript et la méthode addEventListener</u></li> <li>• <u>La propagation des événements</u></li> <li>• <u>Empêcher un événement de se propager et annuler son comportement par défaut</u></li> </ul> <p>6. jQuery</p> <ul style="list-style-type: none"> <li>• Introduction</li> <li>• Les bases de jQuery : Les Sélecteurs, les Événements et les Effets</li> <li>• Les Animations</li> <li>• jQuery et HTML</li> <li>• jQuery et CSS</li> <li>• jQuery et l'arborescence des éléments</li> </ul> <p><b>Travaux pratiques</b></p> <ol style="list-style-type: none"> <li>1. Tp1 : HTML</li> <li>2. Tp2 : CSS</li> <li>3. Tp3 : Programmation javascript</li> <li>4. Tp4 : Manipulation du DOM</li> <li>5. Tp5 : jQuery</li> </ol> <p><b>Projets à réaliser</b></p> <ol style="list-style-type: none"> <li>1. Mini projet 1 : Réalisation de site en utilisant HTML et CSS</li> <li>2. Mini projet 2 : réalisation d'un jeu en utilisant jQuery</li> </ol>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 30 heures et 30h non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours : 15h et TP : 15h
<b><u>Méthodes d'évaluation</u></b>	<p><b>Pratique</b> : Des mini-projets et une évaluation finale (Examen TP) sont prévus.</p> <ul style="list-style-type: none"> <li>- Les mini-projets seront consacrés à l'évaluation du niveau d'atteinte des acquis d'apprentissage de 1 à 4.</li> <li>- L'évaluation finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage de 1 à 6.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors des mini-projets l'élève doit montrer la bonne maîtrise et l'utilisation des notions de base de chacun des cours :HTML, CSS, Javascript, jQuery</p> <p>Lors de l'examen final, l'élève est amené à montrer ses facultés à utiliser tous les acquis du cours afin de réaliser un site web statique attractif et conforme au cahier des charges, en faisant appel à sa créativité, son imagination et son gout artistique.</p>

<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Mini-projets : 50% Evaluation du Mini-projet final (Examen ) : 50%
<b><u>Pré requis</u></b>	Notion de programmation peu importe le langage
<b><u>Acquis de formation visés</u></b>	PLO1, PLO6 et PLO8
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>• Mozilla Developer Network : – <a href="https://developer.mozilla.org/fr/docs/Web/HTML">https://developer.mozilla.org/fr/docs/Web/HTML</a> – <a href="https://developer.mozilla.org/fr/docs/Web/CSS">https://developer.mozilla.org/fr/docs/Web/CSS</a> – <a href="https://developer.mozilla.org/fr/docs/Web/JavaScript">https://developer.mozilla.org/fr/docs/Web/JavaScript</a></li> <li>• Reg Braithwaite – JavaScript Allongé, the “Six” Edition</li> <li>• David Sklar – Learning PHP, A Gentle Introduction to the Web’s Most Popular Language O’Reilly Media, 2016. (Disponible en ligne, après un abonnement gratuit, à BANQ)</li> <li>• FELKE-MORRIS, Terry – Web Development and Design Foundations with XHTML – 5/E. ADDISON-WESLEY, 2010.</li> <li>• SEBESTA, Robert – Programming the World Wide Web – 5/E. ADDISON-WESLEY, 2010.</li> <li>• LEMAINQUE, Fabrice – HTML, XHTML, CSS, Scripts, Le guide Complet – EDITIONS MICRO APPLICATION, 2008.</li> <li>• C. Porteneuve - Bien developers pour le Web 2.0 - Groupe Eyrolles, 2007</li> <li>• P.B. Naigeon - Tutos «Notion de base, CSS» - 2 Mai 2006</li> <li>• NEBRA, Mathieu – Réussir son site Web avec XHTML et CSS – EYROLLES, 2010.</li> </ul>

## INF1-2-2 Composants & systèmes logiques

Groupe de module	INF1-2		Intitulé du module	Composants et systèmes logiques		Code	INF1-2-2	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	x
	TD	X					2	
	TP			Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		30h 22.5hC , 7.5 hTD			Coefficient	2		
					Crédits	2		
Responsable du Module		Meriem Riahi		Intervenants	Sameh Kazis			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  <div><div>1.</div><div>Comprendre le fonctionnement des composants électroniques de base, en particulier : la diode et le transistor.</div><div>2.</div><div>Maitriser la mise en oeuvre des composants dans les circuits élémentaires.</div><div>3.</div><div>Traiter en détails les différents systèmes de numération et codage (codes BCD, GRAY et ASCII )</div><div>4.</div><div>Se familiariser avec l'algèbre de Boole, les fonctions et les portes logiques</div><div>5.</div><div>Etudier et réaliser des systèmes combinatoires</div></div>						
<u>Contenu</u>		<div><div>Chapitre 1 : Introduction aux Composantes électroniques</div><div><div>•</div><div>Semi-conducteurs et jonctions PN.</div><div>•</div><div>Diodes.</div><div>•</div><div>Transistors bipolaires.</div><div>•</div><div>Transistors MOS.</div></div><div>Chapitre 2 : Systèmes de Numération</div><div><div>•</div><div>Codage binaire (BCD, GRAY et ASCII)</div><div>•</div><div>Changement de base (binaire, octal, hexadécimal et décimal)</div><div>•</div><div>Représentation des entiers, des réels et des caractères</div></div><div>Chapitre 3 : Algèbre de Boole</div><div><div>•</div><div>Théorèmes et propriétés de l'Algèbre de Boole:</div><div>•</div><div>Théorème de DeMorgan généralisé</div></div><div>Chapitre 4 : Les fonctions logiques et les portes logiques</div><div><div>•</div><div>Fonctions logiques élémentaires :AND, OR, NAND, NOR, EXOR, EXNOR,</div><div>•</div><div>Différentes formes des fonctions logiques : Forme numérique d'une fonction logique, Table de Vérité, Logigramme, Chronogramme</div><div>•</div><div>Simplification algébrique</div><div>•</div><div>Simplification par tableaux de Karnaugh</div></div><div>Chapitre 5 : Introduction aux Circuits logiques combinatoires</div><div><div>•</div><div>Afficheur 7 segments,</div><div>•</div><div>Circuits de transcodages (encodeur, décodeur ...),</div></div></div>						
<u>Méthode d'enseignement et d'apprentissage</u>		face à face (présentiel) 30 heures et 10 H non présentiel						

<b><u>Technique d'enseignement</u></b>	Cours 22.5 heures et TD 7.5 heures
<b><u>Méthodes d'évaluation</u></b>	Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus. L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3. L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3, 4 et 5.
<b><u>Critères d'évaluation</u></b>	Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à : - Appliquer correctement les théorèmes généraux de l'électricité - Analyser des circuits conçus avec des diodes, savoir identifier correctement le schéma équivalent de la diode dans un circuit, formuler les équations correspondantes et les résoudre à l'aide de méthodes analytiques. Au terme du test final, les étudiants devront être capable de traduire le comportement d'un système logique par des équations logiques et les simplifiant grâce aux règles de l'algèbre booléenne et aux tables de Karnaugh.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Test écrit à mi-parcours (D.S) : 1/3 Test écrit final (Examen): 2/3
<b><u>Pré requis</u></b>	Avoir des connaissances élémentaires en arithmétique, Théorème généraux de l'électricité.
<b><u>Acquis de formation visés</u></b>	PLO1, PLO3
<b><u>Références Bibliographiques</u></b>	1. Adel SAID, ELECTRONIQUE GENERALE, 2013/2014 2. Valkov S., Electronique analogique, 1994 Casteilla. 3- Boittiaux B., Cours d'électronique, 1995 Lavoisier. 4- Circuits logiques, Eric Carioul 5- Logique Combinatoire et Technologie, Marcel Gindre et Denis Roux, Editeur : BELIN, 1984. ISBN : 2-7011-0857-8. 6- Systèmes Numériques, Jaccob Millman et Arvin Grabel, Editeur : MCGRAW-HILL, 1989. ISBN : 2-7042-1182-5. 7- Circuits logiques de base, Adrian Daerr, 2016/2017

## INF1-2-3 Bases de données

Groupe de module	INF1-2		Intitulé du module	Bases de données		Code	INF1-2-3	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	x
	TD	X					2	
	TP			Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		Présentiel : 30h (22.5hC , 7.5 hTD) Non présentiel : 20 h			Coefficient		2	
					Crédits		2	
Responsable du Module		Ines Bayoudh Saâdi		Intervenants	Ines Bayoudh Saâdi /Narjes Hacheni			

<b><u>Acquis d'apprentissage</u></b>	<p>A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :</p> <ol style="list-style-type: none"> <li>1. Comprendre l'ensemble des concepts sous-jacents aux bases de données.</li> <li>2. Connaître les différents niveaux d'abstraction de modélisation des données</li> <li>3. Connaître et maîtriser les concepts de conception, d'implémentation et de manipulation de la base de données.</li> <li>4. Mettre en pratique les concepts théoriques pour concevoir une base de données en respectant les règles et les normes des modèles de données.</li> <li>5. Traduire un schéma conceptuel en un schéma logique (relationnel) cohérent.</li> <li>6. Comprendre la notion de dépendance fonctionnelle et ses différents types</li> <li>7. Appliquer les opérations de l'algèbre relationnelle pour interroger une base de données.</li> <li>8. Comprendre et Appliquer le langage SQL pour Oracle</li> </ol>
<b><u>Contenu</u></b>	<p><b>Cours :</b></p> <p><b>Chp1 : Introduction aux Bases de Données</b></p> <p><b>Chp2 : Modèle Conceptuel (Modèle Entité-Association)</b></p> <p><b>Chp3 : Modèle relationnel</b></p> <p><b>Chp4 : Dépendances Fonctionnelles &amp; Normalisation</b></p> <p><b>Chp5 : SQL pour Oracle pour la Définition et Manipulation de Données</b></p> <p><b>Chp6 : SQL pour Oracle pour l'Interrogation des Données</b></p> <p><b>Projet de Fin d'année:</b> Dans ce type d'activités, les étudiants sont répartis en groupes de 4 apprenants et ils traitent des études de cas de conception et de gestion de base de données en utilisant les logiciels : PowerAMC pour la conception et Oracle pour la gestion de la base de données. Ainsi, les étudiants travaillent et reçoivent des <b>conseils / feedbacks par leurs pairs et de leur tuteur</b>. Ceci leur permet d'être à la fois acteur de leur projet, mais également de développer des compétences pour pouvoir évaluer les projets des autres.</p>

	Un espace de cours sur la plateforme moodle est créé sur la plateforme d'enseignement à distance de l'UVT ( <a href="http://ent.uvt.rnu.tn">ent.uvt.rnu.tn</a> ) pour supporter le déroulement du projet et ses livrables.
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 30 heures et 20h non présentiel (travail à la maison avec des séances de validation présentielle)  Un espace de cours intitulé "Base de données" sur la plateforme d'enseignement à distance de l'UVT ( <a href="http://ent.uvt.rnu.tn">ent.uvt.rnu.tn</a> ), comportant un ensemble de ressources pédagogiques et d'activités d'apprentissage est mis à la disposition des apprenants.
<b><u>Technique d'enseignement</u></b>	Cours et TD = 30h (22,5h et 7,5h)
<b><u>Méthodes d'évaluation</u></b>	Écrit et pratique.  Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus. <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5 et 6.</li> </ul> Evaluation de Projet : le PFA 1 sera comptabilisé à part.
<b><u>Critères d'évaluation</u></b>	Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à maîtriser les concepts sous-jacents à la base de données, Interpréter une description de problème pour construire un modèle Entité/Association et générer le modèle relationnel correspondant.  Au terme du test final, les étudiants devront être capables d'écrire un script de création de base de données, d'alimenter et de manipuler cette base de données, d'exprimer en SQL des requêtes d'interrogation en utilisant SQL pour oracle.  Lors de projet, les étudiants doivent démontrer leur connaissance et leur compréhension des acquis du cours de base de données à travers les différentes études de cas traités par équipes dans le projet de Fin d'années.  Enfin, la capacité des étudiants à participer aux discussions pour justifier leurs choix de modélisation et d'implémentation est évaluée.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Test écrit à mi-parcours (D.S) : 1/3 Etude de cas : 10% de la note du DS Test écrit final (Examen) : 2/3
<b><u>Pré requis</u></b>	Aucun
<b><u>Acquis de formation visés</u></b>	PLO1, PLO7
<b>Références Bibliographiques</b>	<ul style="list-style-type: none"> <li>• J.L, Hainaut. Bases de données : Concepts, utilisation et développement. Dunod, 2015</li> <li>• C.Soutou. SQL pour oracle. Eyrolles, 2010.</li> <li>• A. Rmon et al. Introduction aux bases de données relationnelles. EdiScience, 2002.</li> </ul>



## INF1-3-1 Algorithmique et structure de données (ASD)

Groupe de module	INF1-3		Intitulé du module	Algorithmique et structure de données		Code	INF1-3-1	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	x
	TD	X					2	
	TP			Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		Présentiel : 60h (30h C, 30h TD) Non présentiel : 15 h			Coefficient	3		
					Crédits	4		
Responsable du Module		Besma Fayeche		Intervenants	Hajer Abid			
<u>Acquis d'apprentissage</u>			A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <ul style="list-style-type: none"><li>Comprendre les fondements de base des différentes structures de contrôle.</li><li>Définir correctement l'environnement d'un algorithme.</li><li>Utiliser les structures de données simples.</li><li>Connaître les principaux algorithmes de recherche et de tri.</li><li>Définir des algorithmes récursifs.</li><li>Savoir définir de façon abstraite un type.</li><li>Implémenter des structures de données abstraites (liste chaînée, pile, file, arbre et arbre binaire de recherche et les algorithmes associés</li><li>Evaluation d'une expression arithmétique</li></ul>					
<u>Contenu</u>			<b>Cours</b> <ol style="list-style-type: none"><li>Rappel sur les fondements de base :<ul style="list-style-type: none"><li>Les structures de contrôle (conditionnel et itératif)</li><li>Les procédures et les fonctions (paramètres d'entrée, de sortie et d'entrée/sortie ; passage par valeur, passage par variable)</li><li>Les tableaux (Vecteurs et Matrice) Et Les chaines de caractères</li></ul></li><li>Les enregistrements et Les Fichiers :<ul style="list-style-type: none"><li>Les enregistrements (type énumérer, définition, déclaration, tableaux d'enregistrements)</li><li>Les Fichiers (définition, Organisation et accès, déclaration, manipulation,)</li></ul></li><li>Les algorithmes de tris et de recherches et La récursivité :<ul style="list-style-type: none"><li>Rappel sur les algorithmes de tris et de recherches (Tri : à bulles, par sélection par insertion, Rapide, Fusion, Recherche : séquentielle, dichotomique, ...)</li><li>La récursivité (définition, principe, forme générale de l'algorithme récursif et les types de récursivités)</li></ul></li><li>Pointeurs et Allocation dynamique et Listes chaînées :<ul style="list-style-type: none"><li>Pointeurs et Allocation dynamique (déclaration, utilisation, pointeur et enregistrement).</li></ul></li></ol>					

	<p>- Listes chaînées (Introduction, Les listes linéaires (simple, double, circulaire), Représentation physique, Opérations de base, Représentation récursive, Avantages et inconvénients,)</p> <p>5. Les Piles et files et les Expressions arithmétiques :</p> <p>- Les Piles et files (opérations de base, représentation contiguë, représentation chaînée)</p> <p>- Les Expressions arithmétiques (évaluation des expressions postfixé, préfixé infixé, transformation d'une représentation a une autre)</p> <p>6. Arbres (binaires et généraux) : (Introduction, Arbre général, Arbre Binaire (illustration, parcours, quelques arbres binaires particuliers), Exemple d'implémentation d'arbres binaires, Opérations de base sur les arbres binaires,).</p>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 60 heures et 15H non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 30H et TD=30H
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit</p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3, 4, 5 et principalement 6</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient manipuler les structures permettant d'organiser et d'accéder efficacement aux données, analyser et résoudre des problèmes algorithmiques ayant un aspect récursif en appliquant les concepts fonction et procédure. L'évaluation portera sur leur capacité à identifier correctement la structure adéquate et à résoudre les problèmes avec des méthodes simples et en utilisant le moins d'instructions.</p> <p>Au terme du test final, les étudiants devront résoudre un problème complexe (Les structures de données linéaires (liste chaînées) et structures arborescences (les arbres). Le problème nécessitera également l'interprétation des figures afin de choisir les structures adéquates. Un intérêt particulier sera donné aux problèmes de structures arborescentes. L'évaluation portera sur la capacité des étudiants à bien cerner le problème, à identifier les étapes de résolution et à appliquer correctement les notions étudiées en classe qui sont le mieux adaptées.</p> <p>L'appréciation tiendra compte de la participation des étudiants aux différentes discussions en classe avec leurs enseignant et camarades dans des travaux dirigées, et on donnera une importance particulière aux meilleurs solutions proposées par les étudiants</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 1/3</p> <p>Test écrit final (Examen) : 2/3</p>
<b><u>Prérequis</u></b>	

<b><u>Acquis de formation visés</u></b>	PLO1
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>- Aho, Hopcroft &amp; Ullman, Data Structures and Algorithms, Addison Wesley</li> <li>- M. C. Gaudell &amp; all, Types de données et Algorithmes, Mc Graw Hill</li> <li>- Tenenbaum, Data Structures Using Pascal, Prentice Hall</li> <li>- N. Wirth, Algorithms + Data Structures = Programs, Prentice Hall</li> <li>- Hapcraft, The Design and Analysis of Computer Algorithms AHO, Ullman.</li> <li>- Fraidenaux, M. –C. Gaudel &amp; M. Sonia, Types de données et Algorithmes, McGRAW–HILL,1990</li> <li>- Pair, R. Mohr &amp; R. Schott, Construire les algorithmes, Dunod Informatique, 1988</li> <li>- Algorithmes D.E Knuth CSLI Publications 2011</li> <li>- Cours et exercices corrigés d’algorithmique- J. Julliand Ed Vuibert Fev 2010</li> </ul>

# INF1-3-2 Programmation 1

Groupe de module	INF1-3		Intitulé du module	Programmation 1		Code	INF1-3-2	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	x
	TD	X					2	
	TP	X		Mixte	X		3	
	Pr	X					4	
							5	
Volume Horaire		Présentiel : 60h (22.5 h C/ 7.5 h TD/ 30h TP) Non présentiel : 30h			Coefficient	4		
					Crédits	4		
Responsable du Module		Olfa Temimi		Intervenants	Olfa Temimi/ Hajer Abid			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  <div><div>1.</div><div>Connaître les caractéristiques du langage C et saisir le sens de ses composantes</div></div> <div><div>2.</div><div>Analyser et implémenter des programmes à l'aide des instructions de base</div></div> <div><div>3.</div><div>Construire des fonctions et écrire des programmes modulaires</div></div> <div><div>4.</div><div>Expérimenter les types pointeurs, tableaux et chaînes de caractères.</div></div> <div><div>5.</div><div>Comprendre l'allocation dynamique de la mémoire à l'aide des pointeurs</div></div> <div><div>6.</div><div>Développer des applications avancées utilisant des types de données énumérés, des pointeurs de fonction, des structures imbriquées et des listes chaînées.</div></div> <div><div>7.</div><div>Identifier les bases du mécanisme de traitement des fichiers.</div></div>						
<u>Contenu</u>		<b>Cours</b>  <div><div>1.</div><div>Aspect général d'un programme C</div></div> <div><div>2.</div><div>Instructions élémentaires</div></div> <div><div>3.</div><div>Structures de contrôle<div><div>3.1</div><div>Instructions conditionnelles</div></div><div><div>3.2</div><div>Instructions répétitives</div></div><div><div>3.3</div><div>Instructions de branchement</div></div></div></div> <div><div>4.</div><div>Fonctions<div><div>4.1</div><div>Déclaration de fonctions</div></div><div><div>4.2</div><div>Définition de fonctions</div></div><div><div>4.3</div><div>Appel de fonction et passage de paramètres</div></div><div><div>4.4</div><div>Fonctions récursives</div></div></div></div> <div><div>5.</div><div>Pointeurs et allocation dynamique</div></div> <div><div>6.</div><div>Tableaux à 1 et 2 dimensions</div></div> <div><div>7.</div><div>Chaînes de caractères</div></div> <div><div>8.</div><div>Structures et listes chaînées</div></div> <div><div>9.</div><div>Manipulation des fichiers (textes et binaires)</div></div>						

	<p><b>Travaux pratiques</b></p> <ul style="list-style-type: none"> <li>- Présenter l'environnement de travail</li> <li>- Réaliser une expérimentation directe pour concrétiser les concepts acquis en cours, à travers un ensemble d'exercices variés, pour comprendre le fonctionnement du compilateur et pouvoir détecter et corriger les erreurs syntaxiques et sémantiques.</li> </ul> <p><b>Projets à réaliser par compétences :</b>  Un projet sera effectué durant la deuxième moitié du semestre qui permet d'approfondir les connaissances acquises et mettre le point sur la capacité de l'étudiant à bien exploiter les outils dont il dispose et à montrer ses facultés dans le travail collaboratif.</p>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 60 heures et 30 heures non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	<b>Cours</b> = 22.5 H et <b>TD</b> = 7.5H <b>TP</b> = 30H
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.  Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2, 3, 4 et 5.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 6,7, 8 et 9.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<ul style="list-style-type: none"> <li>- Lors du test à mi-parcours l'étudiant doit montrer la bonne maîtrise des concepts syntaxiques, l'analyse et le découpage d'un problème d'une certaine taille en sous problèmes plus simples et la résolution de ces derniers d'une manière efficace en choisissant les structures de données les plus appropriées.</li> <li>- Lors du test final, les étudiants sont amenés à montrer leurs facultés à utiliser les concepts avancés de la programmation C à savoir les structures, les listes chaînées et la manipulation des fichiers pour résoudre des problèmes plus complexes.</li> </ul> <p><b>Evaluation des travaux pratiques</b>  L'évaluation portera sur la capacité de l'étudiant à analyser, découper, implémenter, compiler, corriger les erreurs puis exécuter un programme en réponse à un énoncé portant sur une partie bien déterminée du cours.</p> <p><b>Evaluation du Mini projet</b>  Pour l'évaluation du mini projet, plusieurs critères seront pris en considération à savoir le respect de l'énoncé et l'originalité du travail, La qualité de programmation (efficacité algorithmique, choix de structures, . . .) et l'organisation du code (commentaires et nommage des données manipulées)</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale

<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux pratiques : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	
<b><u>Acquis de formation visés</u></b>	PLO1, PLO10
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>• B.W. Kernighan, D. M. Ritchie, "The C Programming Language 2nd Ed", Prentice-Hall, 1988.</li> <li>• S.P. Harbison, G.L. Steele Jr, "C A Reference Manual 5th Ed", Prentice-Hall 2002.</li> <li>• C. Delannoy - "Programmer en langage C : Cours et exercices corrigés". Editions Eyrolles, 2009.</li> </ul>

## INF1-4-1 Introduction aux réseaux informatiques

Groupe de module	INF1-4		Intitulé du module	Introduction aux réseaux informatiques		Code	INF1-4-1	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	x
	TD	X					2	
	TP			Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		45h 30hC, 15h TD			Coefficient	3		
					Crédits	3		
Responsable du Module		Lamia Ben Azouz		Intervenants	Yassine Faleh			
Acquis d'apprentissage		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  <div><div>1.</div><div>Se familiariser avec la terminologie de base de la télécommunication et des réseaux téléinformatique</div></div> <div><div>2.</div><div>S'initier aux équipements d'interconnexion;</div></div> <div><div>3.</div><div>Connaître les caractéristiques des réseaux;</div></div> <div><div>4.</div><div>Connaître et maîtriser le modèle OSI</div></div> <div><div>5.</div><div>Connaître les techniques de transmission et des signaux (Numériques, analogiques, Modulations, Synchronisations ...)</div></div> <div><div>6.</div><div>Connaître les principaux services et protocoles des couches basses du modèle OSI;</div></div> <div><div>7.</div><div>Connaître les réseaux de transmission de données.</div></div>						
Contenu		<b>Cours :</b> <b>Chapitre 1 Généralités</b> <div><div>1-</div><div>Introduction</div></div> <div><div>2-</div><div>Architecture des systèmes de communication</div></div> <div><div>3-</div><div>Les réseaux informatiques</div></div> <div><div>4-</div><div>Les réseaux de transmission de données</div></div> <div><div>5-</div><div>Classification des réseaux informatiques (Portée, nature, débit)</div></div> <div><div>6-</div><div>Différents types de réseaux téléinformatiques (industriels, locaux...)</div></div> <div><div>7-</div><div>Eléments d'un système téléinformatique (Equipements terminaux, Equipements d'interconnexion).</div></div> <b>Chapitre 2 Topologie et câblage</b> <div><div>1-</div><div>Topologie en bus</div></div> <div><div>2-</div><div>Topologie en étoile</div></div> <div><div>3-</div><div>Topologie en anneau</div></div> <div><div>4-</div><div>Topologie en arbre</div></div> <div><div>5-</div><div>Topologie Maillée (WAN)</div></div> <div><div>6-</div><div>Supports de transmission (Coaxial, Paires torsadées, Fibre optique)</div></div> <b>Chapitre 3 Fonctionnement théorique des réseaux téléinformatiques</b> <div><div>1-</div><div>Structure en couches</div></div> <div><div>2-</div><div>Notion de protocole</div></div> <div><div>3-</div><div>Notion de service</div></div>						

	<p>4- Modèle de référence OSI</p> <p><b>Chapitre 4 La couche physique</b></p> <p>1- Le Codage de l'information</p> <p>2- La Transmission</p> <p>3- Structure de la liaison physique (ETCD, ETDD)</p> <p>4- Les Caractéristiques d'une voie de transmission (bande passante, impédance)</p> <p>5- Le Débit binaire</p> <p>6- L'erreur et le taux d'erreur</p> <p>7- Le Temps de transfert</p> <p>8- Le Rapport signal bruit</p> <p>9- Les Techniques de transmission (Bande de base, large bande)</p> <p>10- Les Codes de transmission (Manchester, NRZ, Miller...)</p> <p>11- La modulation et les techniques de modulation</p> <p>12- Les Modems (V21, V22, V90)</p> <p>13- Les Modes d'exploitation d'un support (Simplex, half duplex, full duplex)</p> <p>14- La transmission en série et en parallèle</p> <p>15- Les modes de communication (Synchrone, Asynchrone)</p> <p>16- Le Multiplexage et le partage d'une voie de transmission</p> <p>17- Les Liaisons Hertziennes</p> <p><b>Chapitre 5 La couche Liaison de données</b></p> <p>1- Construction des trames</p> <p>2- Codes détecteurs et correcteurs d'erreurs</p> <p><b>Chapitre 6 Les réseaux de transmission de données</b></p> <p>1- Le réseau RNIS</p> <p>2- Le réseau Frame relay</p> <p>3- Les réseaux des LS point à point</p> <p>4- Le protocole HDLC</p>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 45 heures et 10H non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 30 heures et TD 15 H TP= 0 H
<b><u>Méthodes d'évaluation</u></b>	<ul style="list-style-type: none"> <li>• Un devoir surveillé en mi période de formation</li> <li>• Un examen final à la fin de la formation</li> <li>• Un examen d contrôle (rattrapage)</li> </ul>
<b><u>Critères d'évaluation</u></b>	Lors du devoir surveillé de mi-parcours les étudiants doivent maîtriser les notions étudiées au niveau des trois premiers chapitres. En examen final l'étudiant est évalué et tester sur la totalité du cours pour s'assurer de acquis.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale



<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de <b>0</b> (minimum) à <b>20</b> (maximum)</p> <p>La moyenne minimale de validation est <b>10</b></p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : <b>1/3</b></p> <p>Test écrit final (Examen) : <b>2/3</b></p>
<b><u>Pré requis</u></b>	Connaissances de base en réseau
<b><u>Acquis de formation visés</u></b>	PLO1, PLO4
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>• G. Pujolle, O. Salvatory, Les Réseaux, Eyrolles.</li> <li>• Claude Servin Réseaux et Télécoms Dunod Sciences SUP</li> <li>• G. Pujolle, E.Horlait Architecture des réseaux Informatiques Les outils de communication Eyrolles.</li> <li>• TCP/IP Architectures et protocoles applications Dougla Coher.</li> <li>• Shaim's Réseaux ED TITTEL.</li> <li>• M. Rolin, Réseaux locaux, normes et protocoles, Hermès.</li> <li>• Intreconnecting Cisco Network Device - Préparation CCNA Stève Mc Query</li> </ul>

# INF1-4-2 Système d'exploitation 1

Groupe de module	INF1-4		Intitulé du module	Système d'exploitation 1		Code	INF1-4-2	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	x
	TD	X					2	
	TP	X		Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		45h 15hC, 15h TDet 15hTP			Coefficient	3		
					Crédits	3		
Responsable du Module		Zouel Kouki		Intervenants	Zouel Kouki /Faiza Ben Nasr			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <div><div>1.</div><div>Appréhender les concepts généraux des systèmes d'exploitation (SE) et leur historique d'évolution.</div><div>2.</div><div>Comprendre le rôle et les tâches attribuées aux SEs.</div><div>3.</div><div>Distinguer les différents types des SEs et classifier les SE populaires.</div><div>4.</div><div>Comprendre la multiprogrammation et le besoin de partage des mémoires et du processeur qu'elle induit.</div><div>5.</div><div>Comprendre la programmation système et les notions généraux d'interruptions et de processus.</div><div>6.</div><div>Maîtriser les manipulations simples des processus via le SHELL UNIX</div><div>7.</div><div>Comprendre la gestion des entrées/sorties et connaître les modes de communication machine/périphériques</div><div>8.</div><div>Comprendre l'organisation des données et de l'espace libre sur les disques de stockage.</div><div>9.</div><div>Maîtriser le SGF UNIX.</div><div>10.</div><div>Maîtriser le langage SHELL.</div><div>11.</div><div>créer des programmes SHELL visant des comportements spécifiques.</div></div>						
<u>Contenu</u>		<b>Cours</b> <div><div>1.</div><div>Ordinateurs et systèmes d'exploitation : le tour d'horizons</div><div>2.</div><div>Notions de base des systèmes d'exploitation</div><div>3.</div><div>Multiprogrammation et notions de processus (cas des SEs UNIX)</div><div>4.</div><div>Systèmes de gestion de fichiers (SGF)</div><div>5.</div><div>Gestion des entrées/sorties</div></div> <b>Travaux pratiques</b> <div><div>•</div><div>TP1 : Commandes UNIX de base.</div><div>•</div><div>TP2 : Gestion des Processus</div><div>•</div><div>TP3 : Manipulations des fichiers, Redirection et filtres</div><div>•</div><div>TP4 : Programmation SHELL UNIX</div></div>						
<u>Méthode d'enseignement et d'apprentissage</u>		face à f ace (présentiel) 45 heures et 10 non présentiel (travail à la maison)						

<b><u>Technique d'enseignement</u></b>	Cours 15 heures, TD 15 heures et TP 15 heures.
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus pour évaluer les acquis théoriques de l'étudiant.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5 et 6</li> <li>• Un test pratique à la fin du parcours est prévu pour évaluer le niveau d'atteinte des acquis d'apprentissage de travaux pratiques 1, 2, 3 et 4.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à décrire une vue globale du système d'exploitation et de son fonctionnement. Au terme du parcours, les étudiants seront en mesure de rapporter une vision modulaire du système d'exploitation. Il sauront décrire un à un les modules essentiels du SE.</p> <p>Le test pratique final est une occasion pour les étudiants pour démontrer leur compréhension des concepts de base du SE et leur capacité de communiquer avec la machine à travers la ligne de commande shell</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante:</p> <p>Test pratique final : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	Notions élémentaires d'algorithmique et de programmation, et d'architecture des ordinateurs
<b><u>Acquis de formation visés</u></b>	PLO1, PLO3
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>• «A. TANNENBAUM, Les Systèmes d'Exploitation, Conception et Mise en Oeuvre, InterEdition, 1987</li> <li>• A. TANNENBAUM, Systèmes d'Exploitation, PEARSON Education, 2003</li> <li>• M. GRIFFITHS &amp; al, Architecture des Systèmes d'Exploitation, Hermès 1988</li> <li>• J. BEAUQUIER, B. BRENARD, Systèmes d'Exploitation, Concepts et Algorithmes, MC Graw Hill, 1990</li> <li>• S. KRAKOWIAK, Principes des Systèmes d'Exploitation des Ordinateurs, DUNOD, 1985</li> <li>• Neil Mattiew &amp; Richard Stones, Programmation Linux, EYROLLES, 2000</li> </ul>

## INF1-5-1 Economie générale

Groupe de module	INF1-5		Intitulé du module	Economie générale		Code	INF1-5-1	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	x
	TD						2	
	TP			Mixte			3	
	Pr	X					4	
							5	
Volume Horaire		15hC			Coefficient		1.5	
					Crédits		1	
Responsable du Module		Téja Zribi		Intervenants	Téja Zribi			
Acquis d'apprentissage		1- Connaître des outils de base de l'économie générale et à la façon dont se comportent des entreprises : croissance, chômage, inflation, balance commerciale, etc 2- Acquérir la culture et les connaissances nécessaires à la compréhension du monde économique 1- Utiliser des techniques économétriques de modélisation et de prévision.						
Contenu		<u>Chapitre 1</u> : Définition, objet et méthode en économie. a-Définition de l'économie. b-Objet de l'économie c-La différente méthode en économie. <u>Chapitre 2 : les différents penseurs en économie.</u> a- L'école classique : Adam Smith. b- L'école néoclassique : Menger, Walras.. c- Le Marxisme <u>Chapitre 3 : La science économique : méthodes et préoccupations.</u> a- La microéconomie. b- La macro économie. c- Les donnes et mesure économique.						
Méthode d'enseignement et d'apprentissage		Présentiel (le volume horaire est de 1 heure par semaine).						
Technique d'enseignement		Cours = 15h par semestre : utilisation du tableau						
Méthodes d'évaluation		Écrit et pratique. 1- un test écrit à mi-parcours. 2- mini-projet. Les apprenants sont répartis en groupe de trois à quatre. Certains groupes choisissent leur sujet.						
Critères d'évaluation		* Lors de chaque test, les apprenants devraient démontrer leur capacité à maîtriser les acquis visés par ce module. * Lors du mini-projet, les apprenants doivent démontrer leur capacité de travail en équipe ainsi que la maîtrise de leur connaissance et leur compréhension des acquis du cours d'économie générale à travers le sujet traité.						
Mesure d'évaluation		Attribution d'une moyenne finale						
Critère d'attribution de la moyenne finale		La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Moyenne des tests écrits : 70% Note du mini-projet : 30%						
Pré requis								

<b><u>Acquis de formation visés</u></b>	<b>PLO10</b>
<b>Références bibliographiques</b>	

## INF1-6-1 Techniques de communications écrites 1

Groupe de module	INF1-6		Intitulé du module	Techniques de communications écrites 1		Code	INF1-6-1		
Département	Informatique		Filière	Génie Info.	Option				
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	x	
	TD			Mixte				2	
	TP							3	
	Pr							4	
								5	
Volume Horaire		15hC			Coefficient	1.5			
					Crédits	1			
Responsable du Module		Mohsen Dhaouadi		Intervenants	Mohsen Dhaouadi, Kamel Barghouthi				
<u>Acquis d'apprentissage</u>		<div>- Acquérir les compétences rédactionnelles indispensables pour produire les principaux écrits techniques</div> <div>- Accompagner les étudiants sur la plateforme « <b>Voltaire</b> » pour améliorer leurs performances orthographiques.</div> <div>- Développer des aptitudes à rédiger des textes surtout scientifiques à partir de situations de communication précises.</div>							
<u>Contenu</u>		<div><b>Séance 1</b> : -caractériser un texte par rapport à la situation de communication et à la typologie des textes.</div> <div>-Lecture de textes avec analyse à travers la variation des exercices d'identification, de réécriture et de production écrite.</div> <div><b>Séance 2</b> : Texte et fonctions du langage</div> <div>-Rédiger et maitriser les constituants spécifiques du discours pour les restituer par rapport à la situation de communication.</div> <div><b>Séance 3</b> : Les caractéristiques du texte scientifique et technique :</div> <div>-Maitriser les outils syntaxiques et lexicaux appropriés.</div> <div>-Varier les exercices d'entraînement à la production écrite.</div> <div><b>Séance 4</b> : Réaliser la synthèse de textes et de documents.</div> <div>Rechercher et exploiter les informations afin d'en restituer efficacement l'essentiel.</div> <div><b>Séance 5</b> : Exploiter les techniques de synthèse de texte pour préparer les étudiants à la rédaction d'un rapport.</div> <div><b>Séance 6</b> : Rédiger le rapport de stage comme illustration du discours scientifique et technique.</div> <div>Connaitre les techniques de rédaction de ce type d'écrit.</div> <div><b>Séance 7</b> : Insérer le rapport de stage dans une situation de communication professionnelle</div> <div>Rédiger les séquences fondamentales constituanes d'un rapport de stage</div>							
<u>Méthode d'enseignement et d'apprentissage</u>		cours magistral et pratique							
<u>Technique d'enseignement</u>		cours théoriques par PowerPoint+débats oraux							

<b><u>Méthodes d'évaluation</u></b>	Produire des textes de genre scientifique
<b><u>Critères d'évaluation</u></b>	Maîtrise des spécificités grammaticales et discursives de ce genre de textes.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	moyenne DS1+DS2
<b><u>Pré requis</u></b>	Manipulation du code grammatical usuel du Français
<b><u>Acquis de formation visés</u></b>	PLO9
<b><u>Références Bibliographiques</u></b>	« Guide de communication orale et écrite » Benoit Belon et Ginette Levac. Coll Module 2 édition 2015 « La lettre de motivation pour les filières professionnelles » Fabien Lemerrier. F Marol « Entretien d'embauche et tests de recrutement (Sabine Duhamel, Valérie Lachenaud)

## 2.PROGRAMME DETAILLE : SEMESTRE 2

Code	Module	Volume Horaire
INF1-1-4	Probabilité et statistique	30
INF1-1-5	Recherche opérationnelle	30
INF1-2-4	Technologies web 2	30
INF1-2-5	Architecture des ordinateurs	30
INF1-2-6	Théorie des langages et automates	30
INF1-2-7	Atelier bases de données	15
INF1-3-3	Théorie des graphes	60
INF1-3-4	Programmation 2 (langage C++)	60
INF1-4-3	Réseaux locaux	45
INF1-4-4	Systèmes d'exploitation 2	45
INF1-5-3	Principes de gestion	15
INF1-5-4	Activités socio-culturelles	15
INF1-5-5	Projet de Fin d'Année 1 (PFA1)	15
INF1-6-3	Techniques de Communication orales 1	15
INF1-6-4	Anglais 2	15



## INF1-1-5 Recherche opérationnelle

Groupe de module	INF1-1		Intitulé du module	Recherche Opérationnelle		Code	INF1-1-5	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignant	C	X	Régime	CC	X	Semestre	1	x
	TD	X		2				
	TP			3				
	Pr			4				
				5				
Volume Horaire		30h 22.5hC, 7.5hTD			Coefficient	2		
					Crédits	2		
Responsable du Module		Besma Fayech		Intervenants	Ali Balma			
<u>Acquis d'apprentissage</u>			A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <div><div>1.</div><div>Connaître quelques définitions de la programmation linéaire, son historique et savoir la situer parmi les disciplines de l'informatique ainsi que son utilité à résoudre des problèmes concrets dans l'entreprise.</div><div>2.</div><div>Savoir formuler un programme linéaire à partir d'un descriptif du problème à modéliser et optimiser.</div><div>3.</div><div>Appréhender la notion de dualité et écrire le programme dual d'un programme linéaire donné.</div><div>4.</div><div>Connaître les relations entre le problème primal et son dual et l'utilité de la dualité dans la résolution de problèmes ainsi que de l'interprétation des résultats</div><div>5.</div><div>Analyser la sensibilité des résultats par rapport à des changements dans les paramètres du processus à optimiser</div></div>					
<u>Contenu</u>			<b>Cours</b> <div><div>1.</div><div>Introduction à la programmation linéaire</div><div>2.</div><div>Modélisation et résolution graphique</div><div>3.</div><div>Algorithme du simplexe</div><div>4.</div><div>Théorie de la Dualité</div><div>5.</div><div>Analyse de sensibilité</div><div>6.</div><div>/Modélisation de quelques problèmes réseaux (routage par plus court chemin, arbre de poids minimal, flot maximum,...)</div></div>					
<u>Méthode d'enseignement et d'apprentissage</u>			face à face (présentiel) 30 heures					
<u>Technique d'enseignement</u>			Cours 22,5 heures et TD 7,5H					
<u>Méthodes d'évaluation</u>			Écrit et pratique. Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus. <div><div>•</div><div>L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3.</div></div>					

	<ul style="list-style-type: none"> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5 et 6.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors DS, les élèves doivent démontrer leur aptitude à modéliser des problèmes d'optimisation sous la forme d'un programme linéaire et représenter un problème à deux variables et le résoudre graphiquement.</p> <p>Au terme de l'examen, les étudiants doivent savoir résoudre un programme linéaire par la méthode du simplexe, écrire son dual et interpréter les résultats en faisant intervenir la théorie de dualité. En outre, ils doivent savoir analyser la sensibilité des résultats d'optimisation par le simplexe quand les paramètres du problème changent</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 1/3</p> <p>Test écrit final (Examen) : 2/3</p>
<b><u>Pré requis</u></b>	Algèbre linéaire
<b><u>Acquis de formation visés</u></b>	<u>PLO1</u>
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>▪ J. Teghem, Recherche Opérationnelle : Méthodes d'optimisation. Ed. Ellipses, 2012</li> <li>▪ K. G. Murty, Operations Research : deterministic optimization models, Prentice Hall, 1995</li> <li>▪ Roseaux : exercices et problèmes résolus de la recherche opérationnelle, Elsevier-Masson, 1993</li> </ul>

## INF1-2-4 Technologies Web 2

Groupe de module	INF1-2		Intitulé du module	Technologies Web 2		Code	INF1-2-4	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD				2		x	
	TP	X		Mixte			3	
	Pr				4			
					5			
Volume Horaire		30h 15hC, 15hTP			Coefficient	2		
					Crédits	2		
Responsable du Module	Adel ESSAFI			Intervenants	Adel ESSAFI & Mustapha Touzri			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : 1. Analyser un cahier de charge pour la réalisation d'un site web 2. Concevoir une application WEB 3. Réaliser un site web dynamique avec le langage PHP						
<u>Contenu</u>		<b>Cours</b> 1. Appondissent de JS (ajax) 2. Introduction à PHP et principe de fonctionnement 3. Programmation avec php : variable, structures conditionnelle, boucles, fonctions. 4. Se connecter aux bases de données avec php 5. Gestion des sessions  <b>Travaux pratiques</b> 1. TP 1 : Consommation de service WEB avec PHP 2. TP 2 : Réalisation d'un site web dynamique avec PHP 3. TP3 : Ajouter la couche persistance au site web 4. TP4 : Sécuriser l'accès au site web via les sessions						
<u>Méthode d'enseignement et d'apprentissage</u>		Présentiel TP Travail à la maison						
<u>Technique d'enseignement</u>		50% cours 50%TP Mooc fournies par l'enseignants						
<u>Méthodes d'évaluation</u>		Epreuve présentielle de durée 2H (console continue) Epreuve finale d'une durée de 4H. L'étudiant doit être capable de réaliser un noyau de sire web opérationnel						
<u>Critères d'évaluation</u>		Capacité de conception Capacité de réalisation						

<b><u>Pré requis</u></b>	Cours web statique
<b><u>Acquis de formation visés</u></b>	PLO1, PLO6, PLO8
<b><u>Références Bibliographiques</u></b>	Olivier Rollet, Apprendre à développer un site web avec PHP et MYSQL. Editions eni, aout 2018. Damien Seguy , Philippe Gamache. Sécurité PHP 5 et MySQL. Editions eyrolles. 2007. 250 pages

## INF1-2-5 Architecture des Ordinateurs

Groupe de module	INF1-2		Intitulé du module	Architecture des ordinateurs		Code	INF1-2-5	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X					2	x
	TP			Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		30h 22.5hC, 7.5hTD			Coefficient	2		
					Crédits	2		
Responsable du Module		Sonda Chtourou		Intervenants	Sonda Chtourou			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  1. Comprendre l'architecture matérielle de base des ordinateurs, ses différents composants et leurs rôles. 2. Comprendre les circuits logiques (combinatoires et séquentiels) qui constituent le fondement du matériel des ordinateurs. 3. Savoir modéliser n'importe quelle fonction spécifique niveau portes logiques et en déduire son équation. 4. Identifier les différents types de mémoires et leurs caractéristiques. Savoir hiérarchiser les mémoires selon la vitesse et la capacité de stockage. 5. Comprendre le mode de fonctionnement de l'unité centrale de traitement (processeur) des ordinateurs et leurs composants. 6. Étudier comment les composants des architectures matérielles interagissent entre eux pour exécuter un programme donné. 7. Acquérir des connaissances de programmation assembleur. Développer des applications avec la programmation assembleur et comprendre les différents cycles d'exécution d'une instruction.						
<u>Contenu</u>		<b>Cours</b>  1. Architecture de base d'un ordinateur 2. Circuits combinatoires (Circuits d'aiguillages (multiplexeurs, démultiplexeurs), Circuits de comparaisons (comparateurs....) et circuits Arithmétiques (additionneurs, soustracteurs...) 3. Circuits séquentiels 4. Mémoire 5. Unité centrale de traitement (processeur) 6. Unités d'entrée/sortie et périphériques 7. Programmation assembleur  <b>Travaux dirigés :</b>  TD1. Architecture de base d'un ordinateur TD2. Circuits combinatoires TD3. Circuits séquentiels						

	TD4. Programmation Assembleur
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 30 heures
<b><u>Technique d'enseignement</u></b>	Cours 22.5 heures TD 7.5 heures
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5, 6 et 7.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test écrit à mi-parcours, les étudiants devront maîtriser l'architecture de base des ordinateurs (Von Neumann), ses différents composants et leurs rôles. Ils doivent aussi être capables de modéliser n'importe quelle fonction spécifique donnée (additionneur, soustracteur, diviseur ...) en portes logiques et donner le circuit résultant. Ils doivent aussi maîtriser les bascules et les différents types de registres.</p> <p>Lors du test écrit final (Examen), les étudiants devront démontrer des connaissances solide en terme de l'unité centrale de traitement (processeur) et mémoire. Ils doivent aussi développer des applications avec une programmation assembleur et comprendre les différents cycles d'exécution d'une instruction.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 1/3</p> <p>Test écrit final (Examen) : 2/3</p>
<b><u>Pré requis</u></b>	<p>Avoir des bonnes connaissances en: circuit logique, portes logiques, formulations des équations des sorties des circuits logiques et leurs simplifications.</p> <p>Avoir des bonnes connaissances niveau les systèmes de numération, le changement de base et les opérations arithmétiques (binaires et hexadécimales)</p>
<b><u>Acquis de formation visés</u></b>	PLO1, PLO3
<b><u>Références Bibliographiques</u></b>	<p>Andrew TANENBAUM, « Architecture de l'ordinateur cours et exercice », 4ème édition DUNOD.</p> <p>N. P. CARTER, "Architecture de l'ordinateur", Shaum's ediscience.</p> <p>Bernard GOOSSENS, « Architecture et micro-architecture des processeurs ».</p> <p>Emmanuel Viennet, " Architecture des ordinateurs ".</p>

## INF1-2-6 Théorie des Langages et des Automates

Groupe de module	INF1-2		Intitulé du module	Théorie des langages et des automates		Code	INF1-2-6	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X					2	x
	TP			Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		30h 22.5hC, 7.5hTD			Coefficient	2		
					Crédits	2		
Responsable du Module		Afef Kacem		Intervenants	Afef Kacem			
Acquis d'apprentissage		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de comprendre la notion de langage formel et définir les langages réguliers, les langages hors-contextes et les langages contextuels, en étudiant les mécanismes générateurs (expressions régulières, grammaires hors-contexte) et les mécanismes de reconnaissance (automates, machine de Turing) associés.						
Contenu		<p><b>Cours</b></p> <p>Chapitre I : GÉNÉRALITÉS</p> <ol style="list-style-type: none"><li>1. Introduction</li><li>2. Langages</li><li>3. Définitions</li><li>4. Opérations sur les langages</li><li>5. Grammaires</li><li>6. Dérivations</li><li>7. Ambiguïté</li><li>8. Hiéarchie de Chomesky</li></ol> <p>Chapitre II : LANGAGES RATIONNELS (RÉGULIERS)</p> <ol style="list-style-type: none"><li>1. Introduction</li><li>2. Définition et Propriétés</li><li>3. Expressions régulières</li><li>4. Automates d'états finis</li><li>5. Grammaires régulières</li><li>6. Traduction d'une expression régulière en grammaire régulière</li><li>7. Traduction d'une grammaire régulière en automate</li><li>8. Traduction d'un automate à une grammaire régulière</li></ol> <p>Chapitre III : LES LANGAGES ALGÈBRIQUES (HORS CONTEXTES)</p> <ol style="list-style-type: none"><li>1. Introduction</li><li>2. Grammaires hors contexte</li><li>3. Propriétés des langages hors-contexte</li><li>4. Automates à pile</li><li>5. Grammaire algébrique vers automate à pile</li><li>6. Automate à pile vers grammaire algébrique</li></ol>						

	Chapitre IV : LES LANGAGES CONTEXTUELS 1. Langages contextuels 2. Machines de Turing
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 30 heures
<b><u>Technique d'enseignement</u></b>	Cours 30 heures ( 22,5 cours et 7.5 TD)
<b><u>Méthodes d'évaluation</u></b>	<u>Ecrit et pratique</u> Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus. <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis des chapitres I et II.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis des chapitres III et IV.</li> </ul>
<b><u>Critères d'évaluation</u></b>	Lors du test à mi-parcours, les étudiants devraient démontrer leur : <ul style="list-style-type: none"> <li>• Niveau d'assimilation des différents concepts et propriétés des langages artificiels : notions de vocabulaire, de mots et de langages, opérations de base sur les langages, types de langages.</li> <li>• Niveau de compréhension de la classe des langages réguliers ainsi que des expressions et grammaires régulières et les automates d'états finis et leur problème de l'indéterminisme et minimisation.</li> <li>• Capacité de passer d'une représentation d'un langage régulier à une autre.</li> </ul> Au terme du test final, les étudiants devront démontrer leur <ul style="list-style-type: none"> <li>• Niveau d'assimilation de la classe des langages algébriques et contextuels</li> <li>• Capacité de représenter un langage hors contexte par un automate à pile</li> <li>• Capacité de représenter un langage contextuel par une machine de Turing.</li> </ul>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Test écrit à mi-parcours (D.S) : 1/3 Test écrit final (Examen) : 2/3
<b><u>Pré requis</u></b>	
<b><u>Acquis de formation visés</u></b>	<u>PLO1</u>
<b><u>Références Bibliographiques</u></b>	[1] Jean-Michel Autebert Théorie des langages et des automates, 1997, Amazon. [2] Elise Bonzon, Université Paris Descartes, <a href="http://www.math-info.univ-paris5.fr/vbonzon/">http://www.math-info.univ-paris5.fr/vbonzon/</a> [3] Marie-Paule, Muller <a href="http://les.mathematiques.free.fr/pdf/cours_info_automate_grammaire.pdf">http://les.mathematiques.free.fr/pdf/cours_info_automate_grammaire.pdf</a> [4] Olivier carton, Université Paris, <a href="https://www.irif.fr/~carton/Enseignement/Complexite/MasterInfo/Cours/turing.html">https://www.irif.fr/~carton/Enseignement/Complexite/MasterInfo/Cours/turing.html</a> 1





## INF1-2-7 Atelier Bases de Données

Groupe de module	INF1-2		Intitulé du module	Atelier Base de données		Code	INF1-2-7	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C		Régime	CC	X	Semestre	1	
	TD				2		x	
	TP	X		Mixte			3	
	Pr				4			
					5			
Volume Horaire		15h TP Non présentiel : 10 h			Coefficient	1		
					Crédits	1		
Responsable du Module		Ines Bayoudh		Intervenants	Hajer Abid/ Olfa Temimi			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  <div><div>1.</div><div>Etablir et vérifier un Modèle Conceptuel de Données (Modèle Entité-Association) à l'aide de l'atelier de génie logiciel PowerAMC.</div><div>2.</div><div>Générer le Modèle Physique de Données (MPD) et le script de création d la BD avec PowerAMC</div><div>3.</div><div>Pratiquer la Retro-conception avec PowerAMC</div><div>4.</div><div>Pratiquer le Langage de Définition de Données (LDD) sous Oracle</div><div>5.</div><div>Pratiquer le Langage de Manipulation de Données (LMD) sous Oracle</div><div>6.</div><div>Interroger la BD avec des requêtes SQL de différents niveaux de complexité</div></div>						
<u>Contenu</u>		<b>Travaux pratiques :</b>  TP1 : Exploration de l'atelier de génie logiciel PowerAMC TP2 : Création du Modèle Conceptuel de Données à l'aide du PowerAMC TP3 : Génération du MLD, MPD et du script SQL de création de la BD. TP4 : Création et Alimentation de la BD sous Oracle TP5 : Manipulation de la BD sous Oracle TP6 : Interrogation de la BD : Sélection, Restrictions et Jointures TP7 : Interrogation de la BD : Fonctions, Groupements, Opérateurs-Ensemblistes et Sous-Interrogations  Les travaux pratiques sont réalisées sous l'environnement SQL*PLUS du SGBD Oracle.						
<u>Méthode d'enseignement et d'apprentissage</u>		face à face (présentiel) 15 heures et 10h non présentiel (travail à la maison)						
<u>Technique d'enseignement</u>		TP = 15H						

<b><u>Méthodes d'évaluation</u></b>	<p>Pratique.</p> <p>Des comptes-rendus et un test pratique final (Examen TP) sont prévus.</p> <ul style="list-style-type: none"> <li>• Les comptes-rendus seront consacrés à l'évaluation du niveau d'atteinte des acquis d'apprentissage de 1 à 7.</li> <li>• L'épreuve pratique finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 5, 6 et 7.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors des comptes-rendus, les étudiants devraient démontrer leur capacité à maîtriser la construction d'un Modèle Entité-Association à l'aide du PowerAMC, Générer à partir de ce modèle, le MPD et le script de création de la base correspondant. Ils doivent également démontrer leur capacité à écrire et exécuter des scripts SQL de création, de manipulation et d'interrogation de base de données sous Oracle.</p> <p>Au terme du test final, les étudiants devront être capables d'écrire un script SQL complet de gestion de base de données. L'évaluation portera sur la capacité des étudiants à exprimer et exécuter correctement les requêtes SQL et distinguer les différents types de requêtes d'interrogation de la BD.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Comptes-rendus : 50%</p> <p>Test pratique final (Examen TP) : 50%</p>
<b><u>Pré requis</u></b>	Avoir une bonne connaissance des bases de données relationnelles et du langage SQL
<b><u>Acquis de formation visés</u></b>	PLO1, PLO7
<b>Références Bibliographiques</b>	<ul style="list-style-type: none"> <li>• J.L, Hainaut. Bases de données : Concepts, utilisation et développement. Dunod, 2015</li> <li>• C.Soutou. SQL pour oracle. Eyrolles, 2010.</li> <li>• A. Rmon et al. Introduction aux bases de données relationnelles. EdiScience, 2002.</li> </ul>

## INF1-3-3 Théorie des Graphes

Groupe de module	INF1-3		Intitulé du module	Théorie des Graphes		Code	INF1-3-3	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X			2		x	
	TP	X		X	Mixte		3	
	Pr				4			
					5			
Volume Horaire		60h (22.5hC, 7.5hTD, 30hTP) <b>Non présentiel : 10h</b>			Coefficient	3		
					Crédits	4		
Responsable du Module		Besma Fayeche		Intervenants	Besma Fayeche, Olfa Temimi, Hajer Abid			
<u>Acquis d'apprentissage</u>		<ul style="list-style-type: none"><li>- S'initier au calcul de la complexité algorithmique pour pouvoir évaluer la complexité des algorithmes liés aux graphes</li><li>- Maîtriser les notions de base liées aux graphes orientés et non orientés</li><li>- Appliquer quelques méthodes de calcul et quelques algorithmes pour la recherche de l'existence de chemin ou de chemin de longueur minimale ou maximal dans un graphe orienté valué</li><li>- Maîtriser les méthodes MPM et Pert pour le calcul de la durée minimal d'un projet à ordonnancer</li><li>- Maîtriser deux heuristiques de recherche d'arbres couvrants de poids minimal</li><li>- Etudier les problèmes de flots et appliquer l'algorithme de ford fulkerson pour la recherche de flot de valeur maximale</li></ul>						
<u>Contenu</u>		<ol style="list-style-type: none"><li>Introduction à la complexité algorithmique<ul style="list-style-type: none"><li>- Ressources, tailles des données et opérations</li><li>- Estimations asymptotiques</li><li>- Calcul de la complexité temporelle</li><li>- Evaluation de la complexité des algorithmes itératifs</li></ul></li><li>Eléments de la théorie des graphes<ul style="list-style-type: none"><li>- Concepts de bases</li><li>- Modes de représentation</li><li>- Connexité</li><li>- Parcours eulériens et hamiltoniens</li><li>- Graphes particuliers (Graphes sans circuits (calcul des rangs), arbres, ...)</li></ul></li><li>Cheminement dans les graphes<ul style="list-style-type: none"><li>- Recherche de chemins (matrice des fermetures transitives, reche. des comp.fort.connexes)</li><li>- Algorithmes de recherche de plus court chemins (Ford, Dijkstra, ...)</li></ul></li><li>Méthodes d'ordonnancement de projets (MPM et PERT)</li><li>Algorithmes de recherche d'arbre couvrant minimal<ul style="list-style-type: none"><li>- Enumération d'arbres</li></ul></li></ol>						

	<ul style="list-style-type: none"> <li>- Arbre couvrant de poids minimum : algorithmes Kruskal, Prim</li> </ul> <p>6. Problèmes de flots dans les réseaux</p> <ul style="list-style-type: none"> <li>- Flot de valeur maximale (Alg Ford Fulkerson)</li> <li>- Flot maximal à coût minimal</li> </ul>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 60 heures et 10 heures non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	bien Cours et TD = 30H <b>TP</b> = 30H
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5, 6.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<ul style="list-style-type: none"> <li>- Lors du test à mi-parcours l'étudiant doit montrer la bonne maîtrise des concepts syntaxiques, l'analyse et le découpage d'un problème d'une certaine taille en sous problèmes plus simples et la résolution de ces derniers d'une manière efficace en choisissant les structures de données les plus appropriées.</li> <li>- Lors du test final, les étudiants sont amenés à montrer leurs facultés à utiliser les concepts avancés de la programmation C à savoir les structures, les listes chaînées et la manipulation des fichiers pour résoudre des problèmes plus complexes.</li> </ul> <p><b>Evaluation des travaux pratiques</b> L'évaluation portera sur la capacité de l'étudiant à analyser, découper, implémenter, compiler puis exécuter un programme en réponse à un énoncé portant sur une partie bien déterminée du cours.</p> <p><b>Evaluation du Mini projet</b> Pour l'évaluation du mini projet, plusieurs critères seront pris en considération à savoir le respect de l'énoncé et l'originalité du travail, La qualité de programmation (efficacité algorithmique, choix de structures, . . .) et l'organisation du code (commentaires et nommage des données manipulées)</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>TP : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	
<b><u>Acquis de formation visés</u></b>	PLO1, PLO2, PLO3
<b><u>Références Bibliographiques</u></b>	

--	--

## INF1-3-4 Programmation 2

Groupe de module	INF1-3		Intitulé du module	Programmation 2		Code	INF1-3-4	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD				2		x	
	TP	X		Mixte	X		3	
	Pr				4			
					5			
Volume Horaire		60h (22.5hC, 7.5hTD, 30hTP) Non présentiel : 30h			Coefficient	4		
					Crédits	4		
Responsable du Module		Olfa Temimi		Intervenants	Olfa Temimi/ Hajer Abid			
Acquis d'apprentissage		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <div><div>1.</div><div>Identifier et exploiter les différentes particularités du langage orienté objets C++</div></div> <div><div>2.</div><div>Maitriser les notions de classe, d'objet et de référence.</div></div> <div><div>3.</div><div>Implémenter différents types de constructeurs et des fonctions membre de classe.</div></div> <div><div>4.</div><div>Surcharger des fonctions et utiliser les valeurs par défaut</div></div> <div><div>5.</div><div>Surcharger les opérateurs en C ++</div></div> <div><div>6.</div><div>Appréhender et exploiter les fonctionnalités de la classe string</div></div> <div><div>7.</div><div>Maitriser les fondements de la POO à savoir l'encapsulation, l'héritage, le polymorphisme et l'abstraction</div></div>						
Contenu		<b>Cours</b> <div><div>I.</div><div>Vers une Programmation Orientée Objet</div><div><div>○</div><div>Historique de la programmation (non structurée, procédurale, modulaire, orientée objet)</div></div><div><div>○</div><div>Programmation structurée vs Programmation orientée objet</div></div><div><div>○</div><div>Historique de la POO</div></div><div><div>○</div><div>Historique du C++</div></div></div> <div><div>II.</div><div>Nouvelles possibilités du C++</div></div> <div><div>III.</div><div>Orienté objet en C++</div><div><div>○</div><div>Définition d'une classe</div></div><div><div>○</div><div>Création et manipulation d'objets (ou instances) d'une classe</div></div><div><div>○</div><div>Constructeurs &amp; Destructeurs</div></div><div><div>○</div><div>Propriétés des fonctions membres</div></div><div><div>○</div><div>Attributs et méthodes statiques</div></div><div><div>○</div><div>Attributs et méthodes constantes</div></div><div><div>○</div><div>Fonctions et classes amies</div></div><div><div>○</div><div>Construction d'objets membres</div></div></div> <div><div>IV.</div><div>Surcharge des opérateurs en C++</div><div><div>○</div><div>Surcharge des opérateurs internes</div></div></div>						

	<ul style="list-style-type: none"> <li>○ Surcharge des opérateurs externes</li> </ul> <p>V. La classe string de la STL (Standard Template Library)</p> <p>VI. Héritage simple / multiple et polymorphisme en C++</p> <p>VII. Fonctions et classes template</p> <p><b>Travaux pratiques</b></p> <ol style="list-style-type: none"> <li>1. Présentation de l'environnement de développement</li> <li>2. Déclaration et implémentation de classes simples</li> <li>3. Illustration des propriétés des attributs et des fonctions membres</li> <li>4. Surcharge d'opérateurs</li> <li>5. Utilisation de la classe string de la STL</li> <li>6. Agrégation et héritage</li> <li>7. Polymorphisme</li> </ol>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 60 heures et 30 heures non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 22.5 heures et TD 7.5H TP= 30H
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage des chapitres 1, 2, 3 et 4.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage des chapitres 4, 5 et 6.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants doivent confirmer leur assimilation des concepts orienté objet et de prouver leur aptitude à implémenter des programmes objets qui mettent en évidence les notions de classes, d'objets, d'abstraction et d'encapsulation.</p> <p>La maîtrise des notions d'amitié et de surcharge font aussi partie des critères d'évaluation.</p> <p>Au terme du test final, les étudiants doivent montrer leur assimilation des concepts avancés de l'héritage simple et multiple, des fonctions virtuelles et des classes abstraites et mettre en évidence la réutilisation du code à travers la résolution de problèmes complexes.</p> <p>Pour l'évaluation des travaux pratiques, les étudiants doivent confirmer leur aptitude à concevoir, coder et tester un programme objet en réponse à un problème donné, en</p>



	<p>respectant les bonnes pratiques de la programmation objet (organisation du code et des fichiers d'entête)</p> <p>Lors de l'examen tp final, les étudiants seront amenés à prouver leur maîtrise des différents acquis de ce cours et leur faculté à choisir et appliquer les concepts les plus appropriés pour mener à terme le travail demandé.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale :</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux pratiques : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%.</p>
<b><u>Pré requis:</u></b>	
<b><u>Acquis de formation visés</u></b>	PLO1, PLO6
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>• C. Delannoy - "Programmer en langage C++". Editions Eyrolles, 2011</li> <li>• Jesse Liberty et Bradley Jones - "Le langage C++". Pearson, 2009.</li> <li>• G. P. Reich, "Initiation à C++", Revue Micro Système [Oct. 90, Fév. 91]</li> <li>• Jean Cédric Chappelier et Florian Seydoux, " C++ par la pratique, Recueil d'exercices et aide-memoire" 3<sup>ème</sup> Edition</li> </ul>

## INF1-4-3 Réseaux Locaux

Groupe de module	INF1-4		Intitulé du module	Réseaux Locaux		Code	INF1-4-3	
Département	Informatique		Filière	Génie info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X					2	x
	TP	X		Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		45h 22,5hC, 7,5hTD, 15h TP			Coefficient	3		
					Crédits	3		
Responsable du Module		Lamia Ben Azzouz		Intervenants	Yassine Faleh			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : 1. Maîtriser les techniques de réseaux 2. Connaître les différents types des réseaux locaux ; 3. Connaître les différents supports de transmission ; 4. Maîtriser les topologies des réseaux locaux 5. Maîtriser les réseaux sans fil 6. Maîtriser les différents équipements réseau locaux;						
<u>Contenu</u>		<b>Cours:</b> <b>Chapitre 1 Les réseaux locaux et distants</b> <b>I-</b> Généralités <b>II-</b> La transmission locale et la transmission distante <b>III-</b> Le réseau local LAN 1. Introduction 2. Les objectifs des réseaux locaux 3. L'historique des réseaux locaux <b>IV-</b> Les réseaux métropolitains MAN <b>V-</b> Les réseaux étendus WAN <b>Chapitre 2 La topologie des réseaux locaux</b> <b>I-</b> Introduction <b>II-</b> La topologie en bus <b>III-</b> La topologie en étoile <b>IV-</b> La topologie anneau <b>V-</b> La topologie maillée <b>VI-</b> Le choix d'une topologie <b>Chapitre 3 Les supports de transmission</b> <b>I-</b> Introduction aux supports de transmission <b>II-</b> Les câbles coaxiaux 1- Les types de câbles coaxiaux 2- Le matériel de la connexion coaxiale 3- Les caractéristiques de la norme IEEE 802.3 <b>III-</b> Les paires torsadées 1- Les paires torsadées non blindés (UTP) 2- Les paires torsadées blindés (STP) 3- Les paires torsadées semi blindés (FTP) 4- Les Catégories de câbles						

	<p>5- Les composants du câble paires torsadées</p> <p>6- La connexion des câbles aux connecteurs RJ45</p> <p><b>IV- Les fibres optiques</b></p> <p>1- Introduction</p> <p>2- Le principe de fonctionnement</p> <p>3- La fibre multimode</p> <p>4- La fibre monomode</p> <p>5- Les principaux avantages et inconvénients de la fibre optique</p> <p><b>V- Les transmissions en réseau sans fil (Wireless)</b></p> <p>1- Les utilités des réseaux sans fil</p> <p>2- Les types de réseaux sans fil</p> <p>3- Les points d'accès</p> <p>4- Les principales normes des réseaux sans fil 802.11</p> <p>5- Les différentes techniques de transmission</p> <p><b>VI- La comparaison des différents types de câbles</b></p> <p><b>Chapitre 4 Exemples de réseaux locaux</b></p> <p><b>I- Introduction : Les deux premières couches du modèle OSI</b></p> <p>1- La couche physique</p> <p>2- La couche liaison de données</p> <p><b>II- Le réseau Ethernet (Topologie, format de la trame, méthode d'accès au support)</b></p> <p><b>III- Le réseau Token ring (Topologie, format de la trame, méthode d'accès au support)</b></p> <p><b>IV- Le réseau FDDI (Topologie, format de la trame, méthode d'accès au support)</b></p> <p><b>V- Les normes des réseaux locaux</b></p> <p><b>Chapitre 5 Les composants d'un réseau local</b></p> <p><b>I- Les répéteurs</b></p> <p><b>II- Les hubs</b></p> <p><b>III- Les Switchs</b></p> <p><b>IV- Les cartes réseau local</b></p> <p><b>V- La méthode de configuration de la carte réseau local en IP</b></p> <p><b>VI- Les ponts</b></p> <p><b>VII- Les routeurs</b></p> <p><b>VIII- Les équipements et les couches du modèle OSI</b></p> <p><b>Chapitre 6 Les systèmes d'adressage réseau</b></p> <p><b>I- Introduction</b></p> <p><b>II- L'adressage physique</b></p> <p><b>III- L'adressage IP</b></p> <p><b>IV- Le protocole ARP</b></p>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 45 heures et 10H non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 22,5 heures et TD 7,5 H TP= 15 H
<b><u>Méthodes d'évaluation</u></b>	<ul style="list-style-type: none"> <li>• Un devoir surveillé en mi période de formation</li> <li>• Un examen final à la fin de la formation</li> <li>• Un examen d contrôle (rattrapage)</li> </ul>

<b><u>Critères d'évaluation</u></b>	Lors du devoir surveillé de mi-parcours les étudiants doivent maîtriser les notions étudiées au niveau des trois premiers chapitres. En examen final l'étudiant est évalué et tester sur la totalité du cours pour s'assurer de acquis.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de <b>0</b> (minimum) à <b>20</b> (maximum) La moyenne minimale de validation est <b>10</b> La moyenne finale est calculée selon la formule suivante : Travaux Pratiques : 25% Test écrit à mi-parcours (D.S) : 25% Test écrit final (Examen) : 50%.
<b><u>Pré requis</u></b>	Connaissances de base en réseau
<b><u>Acquis de formation visés</u></b>	PLO1, PLO4
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>• G. Pujolle, O. Salvatory, Les Réseaux, Eyrolles.</li> <li>• Claude Servin Réseaux et Télécoms Dunod Sciences SUP</li> <li>• G. Pujolle, E.Horlait Architecture des réseaux Informatiques Les outils de communication Eyrolles.</li> <li>• TCP/IP Architectures et protocoles applications Dougla Coher.</li> <li>• Shaim's Réseaux ED TITTEL.</li> <li>• M. Rolin, Réseaux locaux, normes et protocoles, Hermès.</li> <li>• Intreconnecting Cisco Network Device - Préparation CCNA Stève Mc Querry</li> </ul>

## INF1-4-4 Système d'exploitation 2

Groupe de module	INF1-4		Intitulé du module	Système d'exploitation 2		Code	INF1-4-4	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X					2	x
	TP	X		Mixte	X		3	
	Pr						4	
							5	
Volume Horaire		45h 22.5hC, 7.5hTD, 15hTP			Coefficient	3		
					Crédits	3		
Responsable du Module		Hajer Krichene		Intervenants	Hajer Krichene			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  <div><div>1.</div><div>Développer des connaissances théoriques les concepts de base caractérisant le noyau d'un système d'exploitation SE.</div></div> <div><div>2.</div><div>Se familiariser avec les services d'un SE liés principalement à la gestion des processus, les interactions entre processus en posant le problème d'inter-blocage et les solutions envisagées pour éliminer une telle situation d'inter-blocage.</div></div> <div><div>3.</div><div>Connaitre les principales techniques d'ordonnancement préemptif et celles d'ordonnancement non préemptif, et les différentes politiques de gestion de la mémoire.</div></div> <div><div>4.</div><div>Savoir calculer l'efficacité d'un ordonnanceur par rapport à au autre sur une configuration de processus en se basant sur certains critères de performances (temps de séjour, temps d'attente, temps de réponse, etc.).</div></div>						
<u>Contenu</u>		<b>Cours</b> <div><div>1.</div><div>Les processus</div></div> <div><div>2.</div><div>Ordonnancement des processus</div></div> <div><div>3.</div><div>Communication et synchronisation inter-processus</div></div> <div><div>4.</div><div>Gestion de la mémoire</div></div> <div><div>5.</div><div>Protection et sécurité</div></div> <b>Travaux pratiques</b> <div><div>▪</div><div>des TPs sur les processus (création des processus fils, attente de la terminaison des fils, mort du processus père), la communication interprocessus via les files de messages, la synchronisation des processus via sémaphores, la gestion des threads via les mutex.</div></div> <div><div>▪</div><div>des TDs sur les techniques d'ordonnancement des processus, sur la communication et synchronisation inter processus, et sur les politiques de gestion de la mémoire.</div></div>						
<u>Méthode d'enseignement et d'apprentissage</u>		face à face (présentiel) 45 H et 6H non présentiel (travail à la maison)						

<b><u>Technique d'enseignement</u></b>	Cours 22.5heures et TD 7.5H TP= 15H
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1et 2.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3, 4 et 5.</li> </ul> <p>Evaluation des travaux pratiques : des comptes rendus sont demandés et ainsi corrigés suite à chaque TP.</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test écrit à mi-parcours, les étudiants devront démontrer leur capacité à maîtriser le principe de chacune des principales techniques d'ordonnancement préemptif et d'ordonnancement non préemptif, et à programmer des processus avec les fonctions de manipulations des processus.</p> <p>Au terme du test écrit final, les étudiants devront démontrer leur connaissance et leur compréhension des différentes techniques de gestion de mémoire. Ils devront aussi démontrer leur capacité à utiliser correctement les outils de synchronisation et de communication pour résoudre tout problème d'incohérence de données et d'interblocage.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux pratiques et test pratique: 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test pratique final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	Systèmes d'exploitation 1
<b><u>Acquis de formation visés</u></b>	PLO1, PLO3, PLO8
<b><u>Références Bibliographiques</u></b>	<ol style="list-style-type: none"> <li>1. « Systèmes d'exploitation : systèmes centralisés et systèmes distribués », A. Tannenbaum</li> <li>2. « Systèmes d'exploitation des ordinateurs », Crocus, Dunod</li> <li>3. « Operating systems concepts », A. Silbershatz</li> <li>4. "Principe des systèmes d'exploitation », M.S. Ouerghi</li> </ol>

## INF1-6-3 Techniques de communications orales 1

Groupe de module	INF1-6		Intitulé du module	Techniques de communications orales 1		Code	INF1-6-3	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD			Mixte			2	x
	TP						3	
	Pr	X					4	
							5	
Volume Horaire		15hC			Coefficient		1.5	
					Crédits		1	
Responsable du Module		Mohsen Dhaouadi		Intervenants	Mohsen Dhaouadi, Kamel Barghouthi			
Acquis d'apprentissage		<div>- Développer les capacités et les compétences relationnelles pour communiquer efficacement.</div> <div>- Acquérir les habiletés de base en communication au sein d'un groupe et en animation de réunion.</div>						
Contenu :		<div>Séance 1 : performer son acte de communication avec l'analyse transactionnelle et la PNL (programmation neurolinguistique)</div> <div><div>- Maitriser son mental</div><div>- Gérer ses émotions et ses actes</div><div>- Simuler des situations de communication pour perfectionner ses interactions</div></div> <div>Séance 2 : distinguer et pratiquer les deux types d'oral : l'oral normé et l'oral spontané</div> <div><div>- Rédiger et prononcer un discours avec prise de position selon le contexte approprié</div><div>- Pratiquer la conversation pour améliorer son oral spontané</div><div>- les placer dans des situations de production orale spécifique</div></div> <div>Séance 3 : préparer et réussir son entretien d'embauche</div> <div><div>- savoir identifier ses points forts et valoriser sa candidature</div><div>- adopter les comportements efficients lors des entretiens</div></div> <div>Séance 4 : communiquer et animer efficacement une réunion</div> <div><div>- comprendre les enjeux et l'intérêt de la mise en place d'une réunion</div><div>- organiser, préparer et animer une réunion.</div></div> <div>*2ème prise de parole pour chaque étudiant :</div> <div><div>Séance 5 -----&gt;</div><div>Séance 6 -----&gt;</div><div>Séance 7-----&gt;</div><div><div>- présenter une communication scientifique</div><div>- le sujet : l'objectif et les outils de recherche du PFA 2</div><div>- entretien d'embauche</div></div></div>						

<b><u>Méthode d'enseignement et d'apprentissage</u></b>	cours magistral et TP
<b><u>Technique d'enseignement</u></b>	cours théoriques par PowerPoint+débats oraux
<b><u>Méthodes d'évaluation</u></b>	DS écrit + simulation d'entretien d'embauche
<b><u>Critères d'évaluation</u></b>	la structure de la phrase ;enchaînement des idées ;la richesse du répertoire ; la prononciation et l'articulation, la communication non verbale.
<b><u>Mesure d'évaluation</u></b>	
<b><u>Critère d'attributizon de la moyenne finale</u></b>	<b><u>moyenne DS1+DS2</u></b>
<b><u>Pré requis</u></b>	Maitriser les outils de base de la communication orale.
<b><u>Acquis de formation visés</u></b>	PLO9
<b><u>Références Bibliographiques</u></b>	



### 3.PROGRAMME DETAILLE : SEMESTRE 3

<b>Code</b>	<b>Module</b>	<b>Volume Horaire</b>
INF2-1-1	Algorithmique Avancé	30
INF2-1-2	Compilation	45
INF2-1-3	Programmation 3 (java)	60
INF2-2-1	Atelier de développement	30
INF2-3-1	Sécurité des réseaux	30
INF2-3-2	Réseaux TCP/IP	45
INF2-3-3	Administration Système	30
INF2-3-4	Préparation certif 1 (Cisco CCNA1)	30
INF2-4-1	SGBD	45
INF2-4-2	Génie Logiciel	30
INF2-4-3	Conception orientée objets	30
INF2-5-1	Management Leadership	15
INF2-5-2	Semaine Logiciels 2	15
INF2-6-1	Techniques communications écrites 2	15
INF2-6-2	Anglais 3	15

## INF2-1-1 Algorithmique Avancé

Groupe de module	INF2-1		Intitulé du module	Algorithmique Avancé		Code	INF2-1-1	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD						2	
	TP	X		Mixte	X		3	x
	Pr						4	
							5	
Volume Horaire		30h 22.5hC, 7.5hTP			Coefficient	2		
					Crédits	2		
Responsable du Module		Zouel Kouki		Intervenants	Zouel Kouki			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <ol style="list-style-type: none"><li>1. Appréhender les principes généraux de création d'algorithmes itératifs et récursifs en réponse à un problème donné</li><li>2. Concevoir, implémenter et analyser les algorithmes.</li><li>3. Savoir évaluer l'évolution du temps d'exécution d'un algorithme en fonction de l'évolution de la taille des données.</li><li>4. Formuler les problèmes et appliquer l'approche « Diviser pour régner ».</li><li>5. Savoir examiner les problèmes de décision et les classer selon leurs complexités.</li><li>6. Appréhender les approches gloutonnes et de programmation dynamique.</li><li>7. Savoir déterminer l'approche appropriée à chaque problème.</li><li>8. Savoir implémenter son algorithme dans les langages de programmation usuels.</li><li>9. Discuter, communiquer et partager ses idées avec ses collègues en classe.</li></ol>						
<u>Contenu</u>		<ol style="list-style-type: none"><li>1. Rappels mathématiques et Complexité des algorithmes itératifs</li><li>2. Complexité des algorithmes récursifs</li><li>3. Classification de problèmes</li><li>4. Approche diviser pour régner</li><li>5. Algorithmes gloutons</li><li>6. Programmation dynamique</li></ol>						
<u>Méthode d'enseignement et d'apprentissage</u>		face à face (présentiel) 30 heures et 20 non présentiel (travail à la maison)						
<u>Technique d'enseignement</u>		Cours 22,5 heures, TP 7,5 heures.						
<u>Méthodes d'évaluation</u>		Écrit et pratique. Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus pour évaluer les acquis théoriques de l'étudiant. <ul style="list-style-type: none"><li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2, 3 et 4.</li></ul>						

	<ul style="list-style-type: none"> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5 et 6.</li> <li>• Un projet est élaboré à la fin du parcours et est prévu pour évaluer le niveau d'atteinte des acquis concernant les chapitres 4,5 et 6.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test de mi-parcours, les étudiants devront savoir déterminer la complexité algorithmique et démontrer leur capacité à concevoir des algorithmes itératifs et récursifs corrects et efficaces.</p> <p>Lors du test final, les étudiants devront être en mesure d'identifier la classe du problème, le reformuler, savoir choisir la méthode de résolution et créer un algorithme de complexité minimale.</p> <p>Lors de projet, les étudiants doivent démontrer leurs connaissances et compréhension des principes de la classification des problèmes et des approches appropriés. Ils doivent être en mesure de concevoir les algorithmes et</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante:</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%.</p> <p>Projet final : 25%</p>
<b><u>Pré requis</u></b>	Algorithmique et structures de données, programmation, mathématiques de l'ingénieur
<b><u>Acquis de formation visés</u></b>	PLO1, PLO2, PLO3
<b><u>Références Bibliographiques</u></b>	<ol style="list-style-type: none"> <li>1. G. Brassard and P. Bratley. <i>Fundamentals of Algorithmics</i>. Prentice Hall, 1996.</li> <li>2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. <i>Introduction à l'algorithmique</i>. Dunod, 2002. 2e édition,</li> </ol>

## INF2-1-2 Compilation

Groupe de module	INF2-1		Intitulé du module	Compilation		Code	INF2-1-2	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X					2	
	TP	X		Mixte	X		3	x
	Pr						4	
							5	
Volume Horaire		45h 22.5hC; 7.5hTD; 15hTP			Coefficient	2		
					Crédits	2		
Responsable du Module		Afef Kacem		Intervenants	Afef Kacem			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable d'assimiler les principes de base à la réalisation de compilateurs (analyse lexicale, syntaxique et sémantique et génération de code) et les outils fondamentaux utilisés pour effectuer ces analyses (expressions régulières, automates, grammaires, ...).						
<u>Contenu</u>		<b>Cours</b> 1. Chapitre 1 : Généralités Introduction Rôles de la compilation Structure d'un compilateur Phases de la compilation Exemple de traduction d'une instruction 2. Chapitre 2 : Analyse lexicale Introduction Unités lexicales, modèles et lexèmes Attributs des unités lexicales Spécification des unités lexicales Reconnaissance des unités lexicales 3. Chapitre 3 : Analyse syntaxique Introduction Grammaires Arbre de dérivation Analyse descendante Analyse ascendante 4. Chapitre 4 : Analyse sémantique Introduction Définition dirigée par la syntaxe (DDS) Schéma de traduction dirigé par la syntaxe (STDS) Evaluation des attributs Contrôle de type 5. Chapitre 5 : Génération de code Introduction Code intermédiaire Code à 3 adresses simplifié Production de code à 3 adresses Optimisation du code intermédiaire						

	<b>Travaux pratiques</b> <ol style="list-style-type: none"> <li>Analyse lexicale avec l'outil (f)LEX           <ul style="list-style-type: none"> <li>Introduction</li> <li>Structure du fichier de spécification (f)lex</li> <li>Expressions régulières (f)lex</li> <li>Variables et fonctions prédéfinies</li> <li>Exemples de fichiers .l</li> </ul> </li> <li>Analyse syntaxique avec l'outil YACC-BISON           <ul style="list-style-type: none"> <li>Introduction</li> <li>Spécification du fichier YACC/BISON</li> <li>Communication entre LEX et YACC : Tockens</li> <li>Communication entre LEX et YACC : 'yylval'</li> <li>Variables, fonctions et actions prédéfinies</li> <li>Associativité et priorité des symboles terminaux</li> <li>Récupération des erreurs</li> </ul> </li> </ol>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 45 heures
<b><u>Technique d'enseignement</u></b>	Cours 22.5h TD 7.5h TP 15h
<b><u>Méthodes d'évaluation</u></b>	<b><u>Ecrit et pratique</u></b> Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus. <ul style="list-style-type: none"> <li>L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis des chapitres 1, 2 et 3.</li> <li>L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis des chapitres 4 et 5.</li> </ul> Evaluation des travaux pratiques : évaluation du contrôle continu des niveaux atteints d'apprentissage des outils (f)lex, et YACC-BISON
<b><u>Critères d'évaluation</u></b>	Lors du test à mi-parcours, les étudiants devraient démontrer leur : <ul style="list-style-type: none"> <li>Niveau d'assimilation des différents concepts, structure, étapes de compilation.</li> <li>Capacité de réaliser l'analyse lexicale à l'aide d'expressions ou grammaires régulières et automates d'états finis.</li> <li>Niveau de compréhension des étapes d'analyse syntaxique descendante et construction de l'arbre de dérivation.</li> </ul> Au terme du test final, les étudiants devront démontrer leur <ul style="list-style-type: none"> <li>Niveau d'assimilation des concepts et du principe de l'analyse syntaxique ascendante à l'aide des automates à pile.</li> <li>Niveau d'assimilation des concepts et du principe de l'analyse syntaxique ascendante à l'aide des automates à pile.</li> <li>Niveau d'assimilation des concepts et du principe de l'analyse sémantique</li> <li>Capacité de faire la génération et l'optimisation du code intermédiaire.</li> </ul>

	Le test et l'examen tp mettront en épreuve les connaissances acquises dans les séances de tp.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Travaux Pratiques : 25% Test écrit à mi-parcours (D.S) : 25% Test écrit final (Examen) : 50%.
<b><u>Pré requis</u></b>	Théorie des langages et automates
<b><u>Acquis de formation visés</u></b>	PLO1
<b><u>Références Bibliographiques</u></b>	[1] A. AHO, R. GETHI ET J. ULMAN, « COMPILATEURS : PRINCIPES, TECHNIQUES ET OUTILS », INTER EDITIONS 91 [2] N. SILVERIO, « REALISER UN COMPILATEUR », EYROLLES, 95 [3] R. WILHELM, D. MAURER, «LES COMPILATEURS, THEORIE : CONSTRUCTION, GENERATION », MASSON [4] J. LEVINE, T. MASSON, D. BROWN, «LEX _ YACC», EDITIONS O'REILLY INTERNATIONAL THOMSON [5] COMPILATION, THEORIE DES LANGAGES, UNIVERSITE DE BRETAGNE OCCIDENTALE, IUP INGENIERIE INFORMATIQUE, 2EME ANNEE.

## INF2-1-3 Programmation 3 (Java)

Groupe de module	INF2-1		Intitulé du module	Programmation 3 (Java)		Code	INF2-1-3	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	x	Régime	CC		Semestre	1	
	TD						2	
	TP	x		Mixte	X		3	x
	Pr						4	
							5	
Volume Horaire		60h (30h C, 30h TP) Non présentiel 30h			Coefficient		4	
					Crédits		4	
Responsable du Module		Ramzi Farhat		Intervenants	Ramzi Farhat/ Ines Eloued			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <div><div>1.</div><div>Connaitre le langage de programmation orienté objet Java, les principales API offertes par ce langage et les spécificités de ce langage.</div><div>2.</div><div>Développer des applications informatiques en utilisant le langage de programmation orienté objet Java.</div><div>3.</div><div>Choisir les éléments du langage les plus appropriés selon les besoins exprimés dans les cahiers de charges.</div></div>						
<u>Contenu</u>		<div>Cours<div><div>1.</div><div>Introduction au Langage Java</div><div>2.</div><div>Programmation Objet avec Java</div><div>3.</div><div>Gestion des Exceptions avec Java</div><div>4.</div><div>Tableaux en Java</div><div>5.</div><div>Chaînes de caractères</div><div>6.</div><div>Enumérations</div><div>7.</div><div>Accès aux bases de données</div><div>8.</div><div>Programmation générique</div><div>9.</div><div>Collections</div><div>10.</div><div>Programmation multithread en Java</div><div>11.</div><div>Introduction à la programmation graphique avec Java</div></div></div> <div>Travaux pratiques<div><div>1.</div><div>Premier programme java et Introduction à l'IDE Eclipse</div><div>2.</div><div>Création de classes et instanciation en objets</div><div>3.</div><div>Application d'Héritage</div><div>4.</div><div>Classes abstraites et interfaces</div><div>5.</div><div>Gestion d'exceptions</div><div>6.</div><div>Enumérations et collections de données</div><div>7.</div><div>Accès aux bases de données</div><div>8.</div><div>Création d'interfaces graphiques</div></div></div> <div>Projets à réaliser par compétences: Implémenter une application java, conforme à un diagramme de classes où l'étudiant est invité à implémenter des interfaces graphiques, créer une base de données, mettre en œuvres tous les concepts orienté objets java et utiliser des APIs java.</div>						

<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 60 heures et 30 non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 30heures TP= 30 heures
<b><u>Méthodes d'évaluation</u></b>	<p><u>Ecrit et pratique</u></p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1 et 2</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3.</li> </ul> <p>Evaluation du projet :Le projet est consacré à l'évaluation du niveau d'atteinte des acquis d'apprentissage de 1 à 8 et 10.</p> <p>Evaluation des travaux pratiques : évaluation du contrôle continu des niveaux atteints d'apprentissage 1à 5 à travers la correction des comptes rendus 25 %, Evaluation du projet 25% pour contrôler les niveaux d'acquis 1 à 8 et10 et un examen tp 50 % pour contrôler les acquis d'apprentissage 1à 8.</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à identifier les éléments du langage à utiliser pour répondre aux exigences d'un cahier de charges et à écrire du code Java permettant de répondre à l'énoncé donné. L'évaluation portera sur le choix des éléments du langage tels que les classes, les classes abstraites, les interfaces, l'héritage, le polymorphisme, les types et les structures de données et les exceptions.</p> <p>Au terme du test final, les étudiants devront donner le code source d'une application informatique complexe qui nécessite l'utilisation de plusieurs concepts avancés en langage Java, en particulier la programmation générique et les collections. Ils ont également à justifier certains choix conceptuels pour démontrer leur maîtrise des concepts avancés de la programmation orienté objet avec le langage Java.</p> <p>Dans l'évaluation des TPs les étudiants doivent démontrer leur capacité d'écrire du code java en réponse à un énoncé donné, de le compiler sans erreurs et de l'exécuter.</p> <p>Dans l'évaluation du projet les étudiants doivent pouvoir développer, tester et présenter une application java qui met en pratique les différents acquis du cours et des séances de tps mais qui montre aussi leur capacité à utiliser d'autres APIs java comme le javafx.</p> <p>Finalement, dans l'examen tp, les étudiants doivent montrer leur capacité de choisir les structures de données et les classes adéquates pour implémenter un programme java fonctionnel intégrant les différents acquis.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux Pratiques : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%</p>



<b><u>Pré requis</u></b>	Algorithmique Programmation Orienté Objet
<b><u>Acquis de formation visés</u></b>	PLO1, PLO6
<b><u>Références Bibliographiques:</u></b>	Kathy Sierra et Bert Bates, <i>OCA/OCP Java SE 7 Programmer I &amp; II Study Guide</i> , 2015, McGraw-Hill Education. Ben Evans, <i>Java: The Legend</i> , 2015, O'Reilly. Bruce Eckel, <i>Thinking in Java</i> , 2006, Pearson. Claude Delannoy, <i>Exercices en Java</i> , 2017, Eyrolles.

## INF2-3-1 Sécurité des réseaux

Groupe de module	INF2-3		Intitulé du module	Sécurité des réseaux		Code	INF2-3-1	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X					2	
	TP			Mixte	X		3	x
	Pr						4	
							5	
Volume Horaire		30h (22,5h C, 7.5hTD) Non présentiel : 6h			Coefficient	2		
					Crédits	2		
Responsable du Module		Lamia Ben Azouz		Intervenants	Lamia Ben Azouz			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  1. d' identifier les attaques sur un réseau de communication 2. Comprendre le fonctionnement des algorithmes de chiffrement symétrique/ asymétrique 3. Comprendre le fonctionnement de la signature numérique 4. Comprendre les Infrastructures à Clés Publiques 5. Comprendre la sécurité de niveau 2 (Wi-Fi) 6. Comprendre le fonctionnement des protocoles de sécurité IPsec, SSL						
<u>Contenu</u>		<b>Cours</b> <ul style="list-style-type: none"><li>• Besoins en sécurité.</li><li>• Les menaces</li><li>• Firewalls</li><li>• Cryptographie à clés secrète</li><li>• Cryptographie à clés Publiques</li><li>• MAC/Signature</li><li>• Public Key Infrastructure</li><li>• Sécurité 802.11</li><li>• Le protocole IPsec</li><li>• Le protocole SSL</li></ul>						
<u>Méthode d'enseignement et d'apprentissage</u>		face à face (présentiel) 30 heures et 6H non présentiel (travail à la maison)						
<u>Technique d'enseignement</u>		Cours 22.5h et TD 7.5h						

<b><u>Méthodes d'évaluation</u></b>	<p>Écrit</p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2, 3</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5 et 6.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à identifier les attaques sur un cas d'utilisation, définir les services de sécurité et mettre les mécanismes de sécurité nécessaires pour éviter les attaques identifiés .</p> <p>Au terme du test final, les étudiants devront connaître les vulnérabilité des réseaux 802.11 avec et sans infrastructure, les mécanismes mis en place pour les éviter et leurs insuffisances, savoir les appliquer dans d'autres environnement sans fils (véhicule connecté, Internet of things, etc). Ils devront montrer leur capacité à utiliser IPsec et SSL dans les usages adéquats.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 1/3</p> <p>Test écrit final (Examen) : 2/3</p>
<b><u>Pré requis</u></b>	Avoir des bonnes connaissances en Mathématique, réseaux locaux et TCP/IP
<b><u>Acquis de formation visés</u></b>	PLO8
<b><u>Références Bibliographiques</u></b>	<p>1- B. Schneier, <i>Applied Cryptography</i>( 2nd ed.), John Wiley,</p> <p>2-. A. Menezes, P. van Orschot, and S. Vanstone, <i>Handbook of Applied Cryptography</i>, CRC Press, 1996</p>

## INF2-3-2 Les réseaux TCP/IP

<i>Groupe de module</i>	INF2-3		Intitulé du module	Les réseaux TCP/IP		Code	INF2-3-2		
<i>Département</i>	Informatique		Filière	Génie Info.	Option				
<i>Type d'enseignement</i>	C	X	Régime	CC		Semestre	1		
	TD	X					2		
	TP			Mixte	X			3	x
	Pr							4	
								5	
<i>Volume Horaire</i>		45h 30h C, 15hTD			Coefficient	2			
					Crédit	2			
<i>Responsable du Module</i>	Lamia Ben Azouz			Intervenants	Lamia Ben Azouz 1 Khaled Ghorbel				
<u>Acquis d'apprentissage</u>	A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  1. Comprendre le fonctionnement du protocole IP 2. Construire un plan d'adressage IPv4/ IPv6 . 3. Comprendre les fonctions de fragmentation et réassemblage 4. Effectuer des tests et diagnostics du réseau.(ICMP) 5. Comprendre le fonctionnement des protocoles de transport TCP/UDP 6. Comprendre le fonctionnement du DNS et de DHCPv4/v6								
<u>Contenu</u>	<b>Cours</b> <ul style="list-style-type: none"><li>• Internet Protocol IPv4</li><li>• Le protocole de gestion de réseau ICMPv4</li><li>• IPv6</li><li>• ICMPv6 et L'auto-configuration sans états (SLAAC)</li><li>• Protocole de transport TCP/UDP</li><li>• Domain Name Server DNS</li><li>• DHCPv4/V6</li></ul>								
<u>Méthode d'enseignement et d'apprentissage</u>	face à face (présentiel) 45 heures et 6H non présentiel (travail à la maison)								
<u>Technique d'enseignement</u>	Cours et TD=45H								

<b><u>Méthodes d'évaluation</u></b>	<p>Écrit</p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1,2 (IPv4) ,3 et 4</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 2 (IPv6), et principalement 5 et 6</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à maîtriser le fonctionnement du protocole IP, élaborer un plan d'adressage IPv4 ( sous adressage fixe et variable ), maîtriser le procédé de fragmentation et de réassemblage.</p> <p>Au terme du test final, les étudiants devront pouvoir élaborer un plan d'adressage IPv6, maîtriser l'auto-configuration avec état et sans état, maîtriser TCP avec ses fonctionnalités de contrôle de flux, contrôle de congestion et maîtriser les concepts DNS.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 1/3</p> <p>Test écrit final (Examen) : 2/3</p>
<b><u>Pré requis</u></b>	Avoir des bonnes connaissances en transmission de données (couche physique) et en réseaux locaux .
<b><u>Acquis de formation visés</u></b>	PLO1, PLO4
<b><u>Références Bibliographiques</u></b>	<p>1. Andrew Tanenbaum, " Réseaux: Cours et exercices", Edition DUNOD</p> <p>2- François Laissus, "Cours d'introduction à TCP/IP", <a href="http://www.laissus.fr/pub/cours/cours.pdf">http://www.laissus.fr/pub/cours/cours.pdf</a></p> <p>3- Guy Pujolle, Réseaux, Edition Eyrolles</p> <p>4- Pierre Rolin, " Les réseaux: Principes Fondamentaux ", Lavoisier</p>

## INF2-3-3 Administration Système

Groupe de module	INF2-3		Intitulé du module	Administration Système		Code	INF2-3-3	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD						2	
	TP	X		Mixte			3	x
	Pr						4	
							5	
Volume Horaire		30h 22.5hC; 7.5hTP			Coefficient	2		
					Crédits	2		
Responsable du Module		Adel ESSAFI		Intervenants	Adel ESSAFI			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : 1. Travailler et administrer une station de travail Linux 2. Automatiser les tâches grâce à la programmation shell 3. Installer les services nécessaires à l'exploitation de la station de travail 4. Sécuriser le poste de travail 5. Planifier les taches d'administration sous linux						
<u>Contenu</u>		<b>Cours</b> 1. Shells, scripts et gestion de données a. Personnalisation et utilisation de l'environnement du shell b. Personnalisation ou écriture des scripts simples 2. Interface utilisateur et environnement graphique a. Installation et configuration de X11 b. Connaissance des bureaux graphiques Linux c. Accessibilité 3. Tâches d'administration a. Gestion des comptes utilisateurs et des groupes ainsi que des fichiers systèmes concernés b. Automatisation des tâches d'administration par la planification c. Paramètres régionaux et langues 4. Services système essentiels a. Gestion de l'horloge système b. Journaux systèmes c. Serveurs SMTP (MTA) d. Gestion de l'impression et des imprimantes 5. Notions élémentaires sur les réseaux a. Notions élémentaires sur les protocoles Internet b. Configuration réseau persistante c. Résolution de problèmes réseau d. Configuration DNS sur les postes de travail 6. Sécurité a. Tâches d'administration de sécurité b. Configuration de la sécurité du système c. Sécurisation des données avec le chiffrement						

	<b>Travaux pratiques</b> TP1 : programmation shell TP2 : Configurer et exploiter La station de travail linux TP3 : Installation des services de bases de linux (Cron, horloge, SMTP ...) TP4 : Connecter et sécuriser son poste de travail  <b>Remarque importante :</b> A la fin de ce module, l'étudiant doit être capable de passer l'examen de lpi 102
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Présentiel 50% / travaux pratique 50% / travail à la maison
<b><u>Méthodes d'évaluation</u></b>	Projet programmation shell Examen QCM
<b><u>Critères d'évaluation</u></b>	Projet programmation shell : capacité de conception de solution, Qualité de réalisation QCM : Attitude à choisir la réponse qui correspond à la situation posée
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Projet : 50% Examen QCM : 50%
<b><u>Pré requis</u></b>	Avoir des connaissances solides de linux (cours de base + semaine logiciel 3)
<b><u>Acquis de formation visés</u></b>	PLO3
<b><u>Références Bibliographiques</u></b>	Webographie : site officiel de LPI : <a href="http://www.lpi.org">www.lpi.org</a>

## INF2-4-2 Génie Logiciel

Groupe de module	INF2-4		Intitulé du module	Génie logiciel		Code	INF2-4-2	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X					2	
	TP			Mixte	X		3	x
	Pr						4	
							5	
Volume Horaire		30h 22.5hC, 7.5hTD			Coefficient		2	
					Crédits		2	
Responsable du Module		Meriem Riahi		Intervenants	Meriem Riahi			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <div><div>1.</div><div>Maitriser le processus de développement d'un logiciel informatique et les différentes étapes nécessaires et comprendre les enjeux liés à chacune de ces étapes.</div><div>2.</div><div>Maitriser les différentes techniques et outils de spécification de logiciels</div><div>3.</div><div>Connaître les bonnes pratiques de conception et de programmation</div><div>4.</div><div>Distinguer les différentes techniques de tests pour la vérification et la validation des logiciels informatiques</div></div>						
<u>Contenu</u>		<b>Cours</b> <b>Chapitre 1 : Introduction au génie logiciel</b> <ul style="list-style-type: none"><li>La crise du logiciel</li><li>Définition du génie logiciel</li><li>Classification des projets logiciels</li><li>Qualité logicielle</li><li>Eléments de base du génie logiciel</li></ul> <b>Chapitre 2 : Cycles de Vie d'un logiciel</b> <ul style="list-style-type: none"><li>Le cycle de vie d'un logiciel</li><li>Les activités de développement</li><li>Les modèles de cycle de vie (modèle cascade, en V, prototypage, etc..)</li></ul> <b>Chapitre3 Analyse &amp; Spécification des besoins</b> <ul style="list-style-type: none"><li>Problématique de l'analyse des besoins</li><li>Définition et types de spécification</li><li>Différents outils de spécifications</li><li>spécification semi formelle (RdP, Diagramme d'états, DFD, etc.)</li><li>Spécification formmelle (Langage Z)</li></ul> <b>Chapitre 4 Conception</b> <ul style="list-style-type: none"><li>Principes de la conception</li><li>Concepts pratiques de la conception</li><li>Critères d'évaluation de la conception</li><li>Architectures physiques</li></ul> <b>Chapitre 5 Codage et Tests</b>						



	<ul style="list-style-type: none"> <li>• Généralités sur les méthodes de tests et d'intégration : Qu'est ce qu'on teste? et Comment?</li> <li>• Tests unitaires et tests d'intégration</li> <li>• Types de tests : Tests boîte noire et tests boîte blanche</li> <li>• Automatisation des tests</li> </ul>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 30 heures en classe et 6 heures non présentiel à la maison (exercices de TD pour appliquer les différentes techniques de spécification et de tests)
<b><u>Technique d'enseignement</u></b>	Cours : 22.5 h TD : 7.5 h
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit.</p> <p>Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1 et 2.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 2,3 et 4.</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à maîtriser les différents cycles de vie de développement en GL, et à comprendre et savoir pratiquer les différentes techniques de spécification vues en cours (DFE, RdP, Z, etc...).</p> <p>Au terme du test final, les étudiants devront pouvoir maîtriser les pratiques d'une bonne conception d'un logiciel informatique et les différentes techniques de test et validation des logiciels informatiques.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 1/3</p> <p>Test écrit final (Examen) : 2/3</p>
<b><u>Pré requis:</u></b>	Aucun
<b><u>Acquis de formation visés</u></b>	PLO3, PLO7
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>• Alfred Strohmeier and Didier Buchs, « Génie logiciel : principes, méthodes et techniques », Presses polytechniques et universitaires romandes, 1996.</li> <li>• E.J. Braude et M.E. Bernstein, « Software Engineering, modern approaches », Wiley, 2010.</li> <li>• Bertrand Meyer, « Conception et Programmation Orientée Objet », Eyrolles, 2000</li> <li>• J. Printz et J.F. Pradat-Peyre, « Pratique des tests logiciels », Dunod, 2014.</li> <li>• Durand, « Maîtrise d'œuvre des projets informatiques », Dunod, 2004.</li> <li>• Leszek A. Maciaszek, « Requirements Analysis and System Design », Addison Wesley, 2007</li> <li>• René David, Hassane Alla, « Du grafct aux réseaux de Petri", Hermès - Lavoisier, 1992.</li> <li>• Yann Morère, « Cours de réseaux de Petri », Avril 2002.</li> </ul>

## 4. PROGRAMME DETAILLE : SEMESTRE 4

Code	Module	Volume Horaire
INF2-1-4	Programmation mobile	30
INF2-1-5	Interface Homme-Machine	30
INF2-1-6	Intelligence Artificielle	30
INF2-2-2	Atelier JEE	30
INF2-2-3	Atelier Programmation sous Unix/Linux	30
INF2-GLID-2-4	Option 1-1 : Datawarehouse et BI	30
INF2-GLID-2-5	Option 1-2 : ERP	30
INF2-NTS-2-4	Option 2-1 : Administration des réseaux	30
INF2-NTS-2-5	Option 2-2 : Sécurité des applications et BD	30
INF2-3-5	Préparation certif 2 (Cisco CCNA2)	30
INF2-3-6	Plateformes de sécurité	30
INF2-4-4	Architecture orientée services	30
INF2-4-5	Systèmes répartis	30
INF2-4-6	Conception avancée	30
INF2-5-4	Institutions et environnements économiques	15
INF2-5-5	Activités socio-culturelles <sup>2</sup>	15
INF2-5-6	Projet de Fin d'Année 2 (PFA2)	15
INF2-6-3	Techniques de Communication orales 2	15
INF2-6-4	Anglais 4	15

## INF2-1-4 Programmation Mobile

Groupe de module	INF2-1		Intitulé du module	Programmation Mobile		Code	INF2-1-4	
Département	Informatique		Filière	GénieInfo.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD			Mixte			2	
	TP	X					3	
	Pr						4	x
							5	
Volume Horaire		30h 22.5hC; 7.5hTP			Coefficient	2		
					Crédits	2		
Responsable du Module		Mehrez Boulaares		Intervenants	Mehrez Boulaares			

### Acquis d'apprentissage

- A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :
1. Comprendre le fonctionnement d'un système d'exploitation mobile Android et savoir son architecture et ses composants de base.
  2. Installer et configurer l'environnement Android Studio (SDK Manager et Device Manager)
  3. Concevoir et implémenter des interfaces graphiques avec le langage Android
  4. Créer des intentions pour naviguer dans une application Android
  5. Maîtriser les mécanismes de la persistance de données dans Android (Fichiers, préférences, base de données embarquées avec SQLite et base de données externe)
  6. Connaître les habiletés nécessaires et les particularités de développement mobile cross-platform

### Contenu

#### Cours :

#### Introduction à la programmation Mobile

- Situation du marché mobile.
- Les notions et les concepts fondamentaux de la programmation Mobile (Types d'applications, différentes plateformes OS et technologies associées).
- Les particularités de développement mobile sous Android.

#### Les interfaces graphiques sous Android

- La notion d'activité Android (Activity) et son cycle de vie.
- L'implémentation des interfaces graphiques avec différents modes (code Java, XML et assistant Design).
- Le mécanisme d'interaction entre la gestion des interfaces graphiques et la gestion logiques des applications Android.

#### Navigation entre applications sous Android (les intentions)

- Utiliser les intentions ou « intents » pour envoyer un utilisateur vers une autre application système/native (carte, contacts, Appel, etc) ou bien vers une autre activité de l'application.
- Distinguer entre les intentions implicites et explicites.
- Répondre à certaines intentions.

#### Persistance des données dans Android

- Gérer les données dans Android.
- Utiliser les préférences, les fichiers (stockage interne et externe).

	<ul style="list-style-type: none"> <li>▪ Etudier le format JSON.</li> <li>▪ Stocker de l'information structurée sous la forme d'une base de données SQLite.</li> <li>▪ Interagir avec une base de données externe hébergée sur le Cloud).</li> </ul> <p><b>Applications Cross-Platform</b></p> <ul style="list-style-type: none"> <li>▪ Rappel sur les technologies web de base</li> <li>▪ Initiation aux Frameworks mobiles basés sur JavaScript, HTML5 et CSS3.</li> <li>▪ Conception des interfaces web avec Ionic.</li> <li>▪ Gestion de la logique et des données avec Angular.</li> <li>▪ Génération des applications multi-plateformes avec le Framework Apache Cordova</li> </ul> <p><b>Travaux pratiques :</b></p> <ul style="list-style-type: none"> <li>▪ Interfaces Graphiques Android.</li> <li>▪ Intents.</li> <li>▪ Fichier, Format JSON et XML.</li> <li>▪ Base de données SQLite.</li> <li>▪ Android en mode Cloud (Firebase et Openshift)</li> <li>▪ Application hybrides avec Angular, Ionic et Cordova</li> </ul> <p><b>Projets à réaliser par compétences :</b> L'objectif du travail est de traduire toutes les compétences acquises en développement mobile native ou hybride pour réaliser une application.</p>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 30 heures et 20 heures non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 22,5 heures TP= 7.5H
<b><u>Méthodes d'évaluation</u></b>	<p><u>Pratique</u> Evaluation des comptes rendus des travaux pratiques, 2 tests pratiques et un mini projet sont prévus.</p> <ul style="list-style-type: none"> <li>• Corriger quotidiennement les comptes rendus du TP pour évaluer chaque acquis d'apprentissage</li> <li>• Le premier test pratique sera consacré à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3</li> <li>• Le deuxième test pratique sera consacré à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5 et 6</li> <li>• Le mini projet est consacré à l'évaluation du niveau d'atteinte de tous les acquis d'apprentissage</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du premier test pratique, les étudiants devraient démontrer leur :</p> <ul style="list-style-type: none"> <li>• Capacité de compiler, debugger et tester une application Android avec l'émulateur ou bien sur un appareil mobile.</li> <li>• Capacité de créer des interfaces graphiques et des fragments à l'aide de la classe Activity et Fragment</li> </ul> <p>Lors du deuxième test pratique, les étudiants devraient démontrer leur :</p> <ul style="list-style-type: none"> <li>• Niveau de compréhension des différents types d'intentions explicite et implicite</li> </ul>

	<ul style="list-style-type: none"> <li>• Capacité de naviguer dans une application Android en utilisant la classe Intent (gérer les Action, Category et BroadCast Receiver).</li> <li>• Capacité de lier une application Android avec une base de données embarquée (SQLite) et une base de données externe (hébergée sur le Cloud)</li> <li>• Niveau d'assimilation des concepts liés aux services web JSON, REST et la gestion de mécanisme Push Notification</li> <li>• Capacité de créer des interfaces responsives à l'aide des technologies mobile cross-platform</li> </ul> <p>Au terme du mini projet final, les étudiants devront démontrer leur</p> <ul style="list-style-type: none"> <li>• Capacité de créer une application mobile complète (à l'aide d'un design pattern MVP ou MVVM)</li> <li>• Capacité de faire le choix entre les deux types d'application mobile native ou cross-platform</li> </ul> <p>Les comptes rendus des travaux pratiques mettront en épreuve les connaissances acquises dans les séances d'apprentissage.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Tests et Evaluation des travaux pratiques : 25%</p> <p>Examen pratique final (mini-projet) : 75%.</p>
<b><u>Pré requis</u></b>	Avoir des bonnes connaissances en programmation orientée objet avec JAVA, en développement web et en services web.
<b><u>Acquis de formation visés</u></b>	PLO6, PLO8
<b><u>Références Bibliographiques</u></b>	<ol style="list-style-type: none"> <li>1. Cours de Bertrand Estellon (2016), Université Aix-Marseille.</li> <li>2. Cours de Jean-Francois Lalande (2016), INSA Centre Val de Loire.</li> <li>3. Cours de Olivier Le Goaer (2014).</li> <li>4. <u>Android Developer</u>.</li> </ol>

## INF2-1-5 Interface Homme-Machine

Groupe de module	INF2-1		Intitulé du module	Interface homme-machine		Code	INF2-1-5	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD			Mixte			2	
	TP	X					3	
	Pr	X					4	X
							5	
Volume Horaire		30h (15hC, 15h TP) Non présentiel : 10h			Coefficient	2		
					Crédit	2		
Responsable du Module		Meriem Riahi		Intervenants	Meriem Riahi			
Acquis d'apprentissage			A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <div><div>1.</div><div>Comprendre la problématique de l'interaction homme-machine et les problèmes que rencontrent les utilisateurs avec certains logiciels.</div><div>2.</div><div>Maîtriser les principes ergonomiques des IHM</div><div>3.</div><div>Pratiquer la conception et la réalisation d'un prototype interface ergonomique</div><div>4.</div><div>Introduire les nouveautés avec l'informatique ubiquitaire</div></div>					
Contenu			<div>Cours</div> <div>Chapitre 1 : Problématique de l'Interaction Homme-Machine</div> <div><div>1.</div><div>Notion de l'interaction Homme-Machine</div><div>2.</div><div>Historique</div><div>3.</div><div>Enjeux et difficultés</div></div> <div>Chapitre 2 : Apports de la psychologie cognitive</div> <div><div>1.</div><div>Les facteurs humains</div><div>2.</div><div>L'ergonomie cognitive</div><div>3.</div><div>Quelques recommandations ergonomiques</div></div> <div>Chapitre 3 : Conception et évaluation des IHM</div> <div><div>I.</div><div>Conception des IHM</div><div><div>1.</div><div>Modèles architecturaux d'IHM</div><div>2.</div><div>Cycles de vie GL et IHM</div><div>3.</div><div>De la conception centrée utilisateur à la conception centrée contexte</div></div><div>II.</div><div>Evaluation des IHM</div><div><div>1.</div><div>Quoi, quand et comment évaluer</div><div>2.</div><div>Critères d'évaluation</div><div>3.</div><div>Quelques guides de référence</div></div></div> <div>Chapitre 4 : Guide ergonomique pour les applications mobiles</div> <div><div>1.</div><div>Adapter l'ergonomie au contexte d'utilisation</div><div>2.</div><div>Quelques références de guide</div></div> <div>Travaux pratiques</div> <div>Activités pratiques pour distinguer l'apport de l'ergonomie aux interfaces développées</div> <div>Projets à réaliser par compétences:</div> <div>Mini projet de conception et développement d'une interface conviviale pour une application donnée, respectant les principes ergonomiques présentés au cours.</div>					

<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 30 heures en classe et 10 heures non présentiel à la maison (miniprojet en monôme ou en binôme)
<b><u>Technique d'enseignement</u></b>	Cours : 15 h TP : 15 h
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (DS1) et un test écrit final (DS2) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1 et 2.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3 et 4.</li> </ul> <p>Evaluation du projet : évaluation de niveau atteint des acquis d'apprentissage 2,3 et 4.</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur compréhension de la problématique de l'interaction homme-machine et des solutions proposées pour remédier à ces problèmes.</p> <p>Au terme du test final, les étudiants devront savoir résoudre une conception ergonomique d'une interface et savoir évaluer ergonomiquement une interface développée.</p> <p>Lors du mini-projet, les étudiants doivent démontrer leur connaissance et leur compréhension des principes ergonomiques présentés au cours et doivent les pratiquer sur une étude de cas à choisir pour développer un prototype d'interfaces ergonomique pour une application web ou mobile.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Mini projet : 50 %</p> <p>Test écrit à mi-parcours (DS 1) : 25%</p> <p>Test écrit final (DS 2) : 25%.</p>
<b><u>Pré requis</u></b>	Connaissances en Génie Logiciel et en programmation
<b><u>Acquis de formation visés</u></b>	PLO6, PLO7
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>• Jean-Yves Antoine, cours “ Ergonomie des Interfaces &amp; visualisation d'information”, 2014</li> <li>• Miratech ,” Android vs. iOS : un design différent”, (www.miratech.fr ), Livre Blanc, 2015</li> <li>• Nogier J.-F., Bouilllot T., Leclerc J, Ergonomie des interfaces. Dunod; 2011</li> <li>• Jean-François Nogier, <i>Ergonomie du logiciel et design web</i>, 3e édition, Dunod, 2005.</li> <li>• Ben Shneiderman, Catherine Plaisant, <i>Designing the User Interface</i>, 4e édition, Pearson Education / Addison Wesley, 2005.</li> <li>• Christopher D. Wickens, Justin G. Hollands, <i>Engineering Psychology and Human Performance</i>, 3e édition, Prentice Hall, 2000.</li> <li>• Jakob Nielsen, <i>Usability Engineering</i>, Elsevier / Morgan Kaufmann, 1993.</li> </ul>

## INF2-1-6 Intelligence Artificielle

Groupe de module	INF2-1		Intitulé du module	Intelligence Artificielle		Code	INF2-1-6	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD	X					2	
	TP			Mixte	X		3	
	Pr						4	x
							5	
Volume Horaire		30h (22.5hC, 7.5hTD) Non présentiel : 20h			Coefficient	2		
					Crédits	2		
Responsable du Module		Emna Souissi		Intervenants	Emna Souissi			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <div><div>1.</div><div>Connaître quelques définitions de l'IA, l'historique de l'intelligence artificielle et apprécier son importance et savoir situer l'IA dans l'évolution générale des idées sur l'intelligence et la technologie.</div><div>2.</div><div>Comprendre les modes de représentation de connaissance et savoir concevoir des modèles simples.</div><div>3.</div><div>Savoir décrire le fonctionnement général d'un système expert et de certaines méthodes d'inférence ou de contrôle.</div><div>4.</div><div>Simuler le fonctionnement général d'un mécanisme de déduction en programmation logique.</div><div>5.</div><div>Connaître un panorama de problèmes et d'algorithmes pour les résoudre.</div><div>6.</div><div>Analyser, concevoir et implémenter des algorithmes de recherches dans un espace d'états.</div><div>7.</div><div>Comprendre les algorithmes de jeux à deux adversaires.</div><div>8.</div><div>Connaître les agents intelligents.</div><div>9.</div><div>Distinguer les aspects syntaxiques, sémantiques et pragmatiques de l'analyse de phrases et découvrir les outils de base de traitement de langues.</div></div>						
<u>Contenu</u>		<b>Cours</b> <div><div>1.</div><div>L'intelligence Artificielle</div><div>2.</div><div>La représentation des connaissances</div><div>3.</div><div>La résolution de problèmes en IA</div><div>4.</div><div>Les stratégies de recherche dans les espaces d'états</div><div>5.</div><div>Introduction aux systèmes multi-agents</div><div>6.</div><div>Introduction au traitement des langues</div></div> <b>Projets à réaliser par compétences :</b> Implémenter un des algorithmes de recherche dans un espace d'états sur un problème réel.						
<u>Méthode d'enseignement et d'apprentissage</u>		face à face (présentiel) 30 heures et 20 non présentiel (travail à la maison) Un complément de cours est mis sur la plateforme d'enseignement à distance de l'UVT (ent.uvt.rnu.tn)						



<b><u>Technique d'enseignement</u></b>	Cours 22,5 heures et TD 7,5H
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (DS) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3, 4, 5 et 6.</li> </ul> <p>Evaluation du projet : évaluation de niveau atteint des acquis d'apprentissage 3 et 4.</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à savoir déterminer les différentes approches en IA et à identifier les modes de représentation des connaissances.</p> <p>Au terme du test final, les étudiants devront résoudre un problème lié au IA. Le problème nécessitera également la modélisation puis la résolution en utilisant l'un des algorithmes de recherche ou de jeux. Les étudiants devraient démontrer aussi leur capacité à distinguer les différents types d'agents intelligents et les différents niveaux de traitement de langues. L'évaluation portera sur la capacité des étudiants à bien cerner le problème, à identifier le mode de fonctionnement et en particulier, à identifier et appliquer correctement un algorithme.</p> <p>Lors de projet, les étudiants doivent démontrer leur connaissance et leur compréhension des principes fondamentaux des algorithmes de recherche dans un espace d'états ou les algorithmes de jeux. En outre, leur capacité à illustrer leurs connaissances et leur compréhension clairement, synthétiquement et correctement sera évaluée.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 1/3 dont 20% note Mini projet</p> <p>Test écrit final (Examen) : 2/3.</p>
<b><u>Pré requis:</u></b>	Logique formelle
<b><u>Acquis de formation visés</u></b>	PLO2
<b><u>Références Bibliographiques</u></b>	<ul style="list-style-type: none"> <li>▪ « Eléments de l'IA », Henri Farreny et Malik Ghallab, Hermes</li> <li>▪ « L'IA », Jean Paul Haton et Marie Christine Haton, Que sais-je</li> <li>▪ « L'IA et le langage », Gérard Sabah, Hermes</li> <li>▪ « L'IA », Jean Gabriel Gnasaa, Flammarion</li> <li>▪ « IA et informatique théorique », J.M. Alliot et T.Shiey, Cépaduès Editions</li> <li>▪ « L'INTELLIGENCE ARTIFICIELLE : L'EXPERTISE PARTOUT ACCESSIBLE À TOUS », Serge SOUDOPLATOFF, Fondation pour l'innovation politique, 2018</li> </ul>

## INF2-2-2 Atelier JEE

Groupe de module	INF2-2		Intitulé du module	Atelier JEE		Code	INF2-2-2	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	x	Régime	CC	X	Semestre	1	
	TD			Mixte			2	
	TP	x					3	
	Pr						4	x
							5	
Volume Horaire		30h (22.5hC, 7.5hTP) Non présentiel : 30h			Coefficient		2	
					Crédits		2	
Responsable du Module		Ramzi FARHAT		Intervenants		Dr. Ramzi FARHAT		
Acquis d'apprentissage		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : 1. Comprendre l'architecture client/serveur et son implémentation dans le cadre de la plateforme JEE 2. Connaître les principales spécifications de la plateforme JEE 3. Utiliser la plateforme JEE pour développer des applications réparties 4. Construire une application JEE selon le patron de conception MVC						
Contenu		<b>Cours</b> 1. Architectures applicatives 2. Vue d'ensemble de la plateforme JEE 3. Servlet 4. Java Server Pages 5. JSP Standard Tag Library et Expression Language 6. Java Persistence API 7. Enterprise Java Beans 8. Java Server Faces 9. Java Message Service  <b>Travaux pratiques</b> TP1 : Servlet – Communication Client/Serveur TP2 : Servlet – Sessions et Cookies TP3 : Java Server Pages TP4 : Java Persistence API TP5 : Enterprise Java Bean TP6 : Java Server Faces TP7 : Java Message Service  <b>Projets à réaliser par compétences :</b> Implémenter une application répartie conforme au patron de conception MVC utilisant différentes spécifications de la plateforme JEE.						
Méthode d'enseignement et d'apprentissage		face à face (présentiel) 30 heures et 30 heures non présentiel (travail à la maison)						

<b><u>Technique d'enseignement</u></b>	Cours 22.5 heures TP 7,5 heures
<b><u>Méthodes d'évaluation</u></b>	Écrit et pratique. Evaluation des travaux pratiques : évaluation du contrôle continue des niveaux atteints d'apprentissage 3 et 4 à travers la correction des comptes rendus. Evaluation du projet : évaluation de niveau atteint des acquis 3 et 4. Un test final est consacré à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1 et 2.
<b><u>Critères d'évaluation</u></b>	Dans l'évaluation des TPs les étudiants doivent démontrer la capacité d'utiliser les différentes spécifications JEE en les utilisant dans la réalisation d'application Client/Serveur. Dans l'évaluation du projet les étudiants doivent pouvoir concevoir, développer, tester et présenter une application JEE tout en respectant le patron de conception MVC. Ce travail nécessite l'analyse du cahier de charges, la conception d'une solution logicielle, son implémentation, sa documentation et faire une présentation orale pour mettre en valeur le travail réalisé. Finalement, dans le test écrit, les étudiants doivent montrer qu'ils ont compris les différents aspects architecturaux, technologiques et techniques relatives à la plateforme JEE.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Evaluation du projet : 25% Evaluation des TPs : 25% Test écrit : 50%.
<b><u>Pré requis</u></b>	Développement Web : HTML, CSS et JavaScript Langage de programmation Java
<b><u>Acquis de formation visés</u></b>	PLO1, PLO6
<b><u>Références Bibliographiques</u></b>	Antonio Goncalves, " <b>Beginning Java™ EE 6 Platform with GlassFish™ 3 From Novice to Professional</b> ", Second Edition, Apress, 2010.  Oracle, " <b>Java Platform, Enterprise Edition: The Java EE Tutorial</b> ", <a href="https://docs.oracle.com/javaee/7/tutorial/">https://docs.oracle.com/javaee/7/tutorial/</a>

## INF2-2-3 Programmation système sous Unix

Groupe de Module	INF2-2		Module	Programmation système sous Unix		Code	INF2-2-3	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	x	Régime	CC		Semestre	1	
	TD				2			
	TP	x		Mixte	X		3	
	Pr				4		x	
					5			
Volume Horaire		30h (22.5hC, 7.5hTP) Non présentiel : 10h			Coefficient	2		
					Crédits	2		
Responsable du Module		Hajer Krichene		Intervenants	Hajer Krichene			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  <div><div>1.</div><div>Maîtriser la ligne de commande et mémoriser les caractères spéciaux et la majorité des filtres de texte disponibles sur la ligne de commande.</div><div>2.</div><div>Maîtriser les bases de la programmation du Shell, et le principe des scripts Shell pour comprendre les scripts système, et savoir programmer des expressions logiques et arithmétiques ainsi que des boucles.</div><div>3.</div><div>Apprendre les fonctions permettant de manipuler les processus, leurs envoyer des signaux, les stopper, les tuer, les mettre en arrière-plan, et les ramener en avant-plan.</div><div>4.</div><div>Apprendre à gérer la synchronisation entre processus au moyen des signaux.</div><div>5.</div><div>Se familiariser avec les primitives de communication (les tubes sans nom, les tubes nommés, les files de messages, et les sockets) et celles de synchronisation (sémaphores et mutex) mises en œuvre sous Linux.</div></div>						
<u>Contenu</u>		<b>Cours</b> <div><div>1.</div><div>Shell et commandes de base sous LINUX</div><div>2.</div><div>Programmation Shell</div><div>3.</div><div>Contrôle des processus sous Linux</div><div>4.</div><div>Communication interprocessus sous Linux</div><div>5.</div><div>Synchronisation interprocessus sous Linux</div></div> <b>Travaux pratiques</b> <div><div>▪</div><div>Un TP pour se familiariser avec les lignes de commandes liées principalement aux Redirections, Tubes, et Filtres.</div><div>▪</div><div>Un TP sur la programmation des scripts Bash</div><div>▪</div><div>Des TP de programmation UNIX sur la création de processus – recouvrement, la synchronisation via les signaux, les tubes de</div></div>						

	communication, les sockets pour communiquer des processus distants, les sémaphores, les mutex, etc.
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 30 H et 10H non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 22,5 heures et TP 7,5H
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.  Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2, et 3.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3, 4 et 5.</li> </ul> <p>Evaluation des travaux pratiques : des comptes rendus sont demandés et ainsi corrigés suite à chaque TP.</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test écrit à mi-parcours, les étudiants devront démontrer leur capacité à rédiger des scripts en Bash et à programmer des processus avec les fonctions de manipulations des processus.</p> <p>Au terme des tests (pratique et puis écrit) finaux, les étudiants devront démontrer leur connaissance dans la programmation avec les principaux outils et services logiciels (de communication, de synchronisation) fournis par Unix.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux pratiques et test pratique: 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test pratique final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	Programmation en C, Les systèmes d'exploitation
<b><u>Acquis de formation visés</u></b>	PLO1, PLO4, PLO5
<b><u>Références Bibliographiques</u></b>	

## INF2-4-4 Architecture Orientée Services (AOS)

Groupe de module	INF2-4		Intitulé du module	Architecture Orientée services (AOS)		Code	INF2-4-4	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD						2	
	TP	X		Mixte	X		3	
	Pr						4	x
							5	
Volume Horaire		30h (22.5hC, 7.5h TP) Non présentiel : 20h			Coefficient		2	
					Crédits		2	
Responsable du Module		Ines Eloued		Intervenants	Ines Eloued			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <div><div>1.</div><div>Connaître les principes et les concepts de base de l'architecture Orientée Service</div><div>2.</div><div>Maîtriser les langages XML et JSON (Créer et manipuler des documents valides aux formats XML et JSON)</div><div>3.</div><div>Comprendre les concepts, les standards et le fonctionnement des web services SOAP et REST.</div><div>4.</div><div>Créer des applications web hébergeant et consommant des services web SOAP et REST.</div><div>5.</div><div>Saisir les différences entre les services web de type SOAP et REST et être capable de faire le choix selon le besoin.</div><div>6.</div><div>Saisir les concepts de l'Entreprise Service Bus (ESB) et du Business Process Modeling (BPM).</div><div>7.</div><div>Connaître les concepts de base de l'architecture micro services.</div></div>						
<u>Contenu</u>		<b>Cours</b> <div><div>1.</div><div>Introduction aux web services</div><div>2.</div><div>Formats de données : XML, JSON</div><div>3.</div><div>Standards des web services SOAP : WSDL, SOAP, UDDI</div><div>4.</div><div>Web Services RESTFUL</div><div>5.</div><div>Entreprise Service Bus (ESB), Business Process Modeling (BPM)</div><div>6.</div><div>Introduction à l'architecture Micro Services.</div></div> <b>Travaux pratiques</b> <div><div>1.</div><div>XML &amp; schémas XML</div><div>1.1.</div><div>Créer et valider des documents xml</div><div>1.2.</div><div>S'exercer sur les schémas xml</div><div>2.</div><div>Web Services SOAP</div><div>2.1.</div><div>Créer des services web SOAP en utilisant la technologie JAX-WS et la plateforme JEE</div><div>2.2.</div><div>Créer des applications web qui consomment des services web SOAP</div><div>3.</div><div>Web Services Restful</div><div>3.1.</div><div>Créer des services web REST en utilisant la technologie JAX-RS et la plateforme JEE</div><div>2.3.</div><div>Créer des applications web qui consomment des APIs REST</div></div>						

	<p>2.4. Créer des APIs Rest en utilisant les fonction Lambda du Cloud Amazon web service.</p> <p><b>Projets à réaliser par compétences:</b> Concevoir et réaliser une application web qui offre des services web et consomment d'autres.</p>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	présentiel 30 heures et 20 heures non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 22,5 heures TP= 7.5H
<b><u>Méthodes d'évaluation</u></b>	<p><u>Ecrit et pratique</u> Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5,6 et 7.</li> </ul> <p>Evaluation des travaux pratiques : évaluation du contrôle continu des niveaux atteints d'apprentissage 2 à travers la correction des comptes rendus 50% et un examen de synthèse 50 % pour contrôler les acquis d'apprentissage 3, 4 et 6.</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur :</p> <ul style="list-style-type: none"> <li>• Niveau d'assimilation des différents concepts et propriétés des architectures orientées services (tels que couplage, montée en charge, interopérabilité, portabilité et standardisation) et de situer l'architecture Orientée Services par rapport aux autres architectures,</li> <li>• Capacité de formater des données dans le langage XML suivant des schémas XML et aussi JSON.</li> <li>• Niveau de compréhension des différents standards relatifs aux web services SOAP tels que le WSDL, UDDI et SOAP</li> <li>• Capacité de créer les messages requêtes et réponse SOAP à échanger avec un web service à partir de son wsdl, de concevoir et créer la classe d'un service web en se basant sur son wsdl et vice-versa.</li> </ul> <p>Au terme du test final, les étudiants devront démontrer leur</p> <ul style="list-style-type: none"> <li>• Niveau d'assimilation des concepts et du principe des services web Restful</li> <li>• Capacité de créer des requêtes aux APIs REST et de concevoir et créer un web service REST à partir de son WADL ou à partir de sa description.</li> <li>• Capacité de faire le choix entre les deux types de web services SOAP et REST selon la situation et le contexte donnés.</li> <li>• Niveau d'assimilation des concepts liés à l'Entreprise Service Bus (ESB) et le Business Process Modeling (BPM)</li> </ul> <p>Le test et l'examen tp mettront en épreuve les connaissances acquises dans les séances de tp.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale

<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux Pratiques : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	Programmation orientée Objet Java, JEE.
<b><u>Acquis de formation visés</u></b>	PLO3, PLO6, PLO8
<b><u>Références Bibliographiques</u></b>	<p>[1] Libero Maesano, Christian Bernard, Xavier Le Galles. Services Web avec J2EE et .NET : Conception et implémentations, Eyrolles, 2003.</p> <p>[2] José ROUILLARD, Thomas VANTROYS, Vincent CHEVRIN. Les architectures orientées service : Une approche pragmatique des SOA, Vuilbert, 2007.</p> <p>[3] Jean-MarieChauvet, Web services avec SOAP, WSDL, UDDI, EbXML, Eyrolles, 2003</p> <p>[4] Cours architectures Réparties, Haythem Abbas, Maître Assistant à la faculté des sciences de Tunis .</p> <p>[5] Cours Introduction aux architectures orientées services, F.Baude, Maître de conférences à l'université sophia Antipolis, Nice.  <a href="http://deptinfo.unice.fr/~baude/WS/cours_SOA_AO+FB.pdf">http://deptinfo.unice.fr/~baude/WS/cours_SOA_AO+FB.pdf</a>.</p> <p>[6] Cours Web Services REST, Pierre Antoine Champin, Maître de conférences à l'université Claude Bernard Lyon. <a href="http://liris.cnrs.fr/~pchampin/enseignement/rest/">http://liris.cnrs.fr/~pchampin/enseignement/rest/</a></p> <p>[7] Mohamed Youssfi, cours Introduction aux web services SOAP et REST.  <a href="https://www.youtube.com/watch?v=V1HERlQdrCo">https://www.youtube.com/watch?v=V1HERlQdrCo</a></p>



## 5.PROGRAMME DETAILLE : SEMESTRE 5

Code	Tronc Commun (X pour GLID et NTS)	Volume Horaire
INF3-X-1-1	Nouvelles architectures	30
INF3-X-1-2	Linux embarqué	30
INF3-X-1-3	Programmation temps réel	30
INF3-X-1-4	Cloud computing	30
INF3-X-2-1	Techniques multimédia	30
INF3-X-2-2	Génie logiciel avancé	30
INF3-X-2-3	Machine learning	30
INF3-X-2-4	Gestion de projet	30
INF3-X-5-1	Semaine Logiciels 3	15
INF3-X-5-2	Management de l'innovation des projets	15
INF3-X-5-3	Droit Informatique	15
INF3-X-5-4	Anglais 5	15
<b>Option GL et Informatique Décisionnelle (GLID)</b>		
INF3-GLID-3-1	Datamining	30
INF3-GLID-3-2	Processus Stochastique	15
INF3-GLID-3-3	Soft Computing	15
INF3-GLID-4-1	BDA	30
INF3-GLID-4-2	Management de la qualité logicielle et Métriques	30
INF3-GLID-4-3	Big Data	30
<b>Option Nouvelles Technologies et Sécurité des Réseaux (NTS)</b>		
INF3-NTS-3-1	Sécurité Data center	15
INF3-NTS-3-2	Audit et outils de Sécurité	15
INF3-NTS-3-3	Internet of things (IoT)	30
INF3-NTS-4-1	Réseaux et applications multimédia	30
INF3-NTS-4-2	Réseaux des capteurs	30
INF3-NTS-4-3	Préparation Certification (Cisco CCNA)	30

## INF3-X-1-1 Nouvelles Architectures

Groupe de module	INF3-X-1		Intitulé du module	Nouvelles Architectures		Code	INF3-X-1-1	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD						2	
	TP	X		Mixte			3	
	Pr						4	
							5	x
Volume Horaire		30h 22.5hC, 7.5hTP			Coefficient		2	
					Crédits		2	
Responsable du Module		Mehrez Boulaares		Intervenants	Mehrez Boulaares			

### Acquis d'apprentissage

- A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :
1. Introduire les nouvelles architectures logicielles à base de la technologie conteneur Docker.
  2. Apprendre à manipuler Docker et conteneuriser des applications à travers des exemples pratiques.
  3. Introduire la méthodologie DevOps et ses bonnes pratiques pour le développement logiciel moderne.
  4. Savoir mettre en place une usine logicielle d'Intégration Continue et Livraison Continue.
  5. Savoir configurer, approvisionner et administrer une infrastructure par le code d'une manière automatisée.

### Contenu

#### **Cours :**

#### **Introduction aux nouvelles architectures logicielles**

- Applications monolithiques vs Microservices
- Evolution des infrastructures
- Modern IT et l'adoption des applications conteneurisées

#### **Administration Docker**

- Installation Docker
- Création et Administration de conteneurs
- Volumes et Réseaux
- Gestion et Fabrication des images
- Multi-conteneurs

#### **Orchestration des services**

- Cluster Docker
- Docker Cloud

#### **Méthodologie DevOps :**

- Origines, démarche et pile technologique de mise en œuvre
- Méthodes agiles (Scrum) vs DevOps
- Chaîne logicielle de DevOps

#### **Intégration et Livraison Continue**

- Automatisation de la construction logicielle
- Gestion de code source
- Automatisation des tests (unitaires, d'intégration, fonctionnels et de charge)
- Intégration Continue avec Jenkins

	<ul style="list-style-type: none"> <li>▪ Qualité de code</li> <li>▪ Gestion des livrables</li> <li>▪ Déploiement continu</li> <li>▪ Autoscaling des microservices</li> </ul> <p><b>Infrastructure par le code</b></p> <ul style="list-style-type: none"> <li>▪ Configuration et gestion par le code des serveurs virtuels reproductibles</li> <li>▪ Approvisionnement et gestion par le code des applications microservices</li> </ul> <p><b>Projets à réaliser par compétences :</b>  Dans le cadre de développement d'un projet Java JEE, l'objectif du travail est de traduire toutes les compétences acquises en technologie conteneur et DevOps pour mettre en place une plateforme DevOps d'Intégration Continue (CI) et de Livraison Continue (CD).</p>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 30 heures et 20 heures non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 22.5 heures TP= 7.5H
<b><u>Méthodes d'évaluation</u></b>	<p><u>Pratique</u>  Evaluation des comptes rendus des travaux pratiques, 2 tests pratiques et un mini projet sont prévus.</p> <ul style="list-style-type: none"> <li>• Corriger quotidiennement les comptes rendus du TP pour évaluer chaque acquis d'apprentissage</li> <li>• Le premier test pratique sera consacré à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1 et 2</li> <li>• Le deuxième test pratique sera consacré à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3, 4 et 5</li> <li>• Le mini projet est consacré à l'évaluation du niveau d'atteinte de tous les acquis d'apprentissage</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du premier test pratique, les étudiants devraient démontrer leur :</p> <ul style="list-style-type: none"> <li>• Manipulation et Administration Docker</li> <li>• Capacité de créer et administrer des conteneurs Docker en termes de réseaux, images et stockage de données</li> <li>• Capacité de créer un cluster Docker en utilisant Docker-Machine</li> <li>• Capacité de créer une stack multi conteneurs sous forme des microservices en utilisant Docker-Compose et Docker-Stack</li> </ul> <p>Lors du deuxième test pratique, les étudiants devraient démontrer leur :</p> <ul style="list-style-type: none"> <li>• Niveau d'assimilation des concepts liés à l'intégration continue et livraison continue (Gestion décentralisée de code source, automatisation de build, automatisation de test, Qualité logiciel et gestion de livrables, etc.).</li> <li>• Capacité de créer et approvisionner une infrastructure par le code</li> </ul> <p>Au terme du mini projet final, les étudiants devront démontrer leur</p> <ul style="list-style-type: none"> <li>• Capacité de mettre en place une plateforme DevOps basée sur un cluster Docker</li> <li>• Capacité de faire le choix entre les technologies DevOps.</li> </ul> <p>Les comptes rendus des travaux pratiques mettront en épreuve les connaissances acquises dans les séances d'apprentissage.</p>

<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : Tests et Evaluation des travaux pratiques : 25% Examen pratique final (mini-projet) : 75%.
<b><u>Pré requis</u></b>	Avoir des bonnes connaissances en administration linux, en réaux, en développement Java JEE, en architectures orientées services (SOA) et Cloud Computing.
<b><u>Acquis de formation visés</u></b>	PLO3, PLO6, PLO8
<b><u>Références Bibliographiques</u></b>	<ol style="list-style-type: none"> <li>1. <a href="#"><u>DevOps Institute</u></a></li> <li>2. <a href="#"><u>Docker Documentation</u></a></li> <li>3. <a href="#"><u>Documentation Docker Swarm</u></a></li> <li>4. <a href="#"><u>Documentation/Webinars de Red Hat (OpenShift)</u></a></li> <li>5. <a href="#"><u>Cloud Native Computing Foundation Webinars</u></a></li> </ol>

## INF3-X-1-2 Linux embarqué

Groupe de module	INF3-X-1		Intitulé du module	Linux embarqué		Code	INF3-X-1-2	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC		Semestre	1	
	TD						2	
	TP	X		Mixte	X		3	
	Pr						4	
							5	x
Volume Horaire		30h 22.5hC, 7.5hTP			Coefficient	2		
					Crédits	2		
Responsable du Module		Sonda Chtourou		Intervenants	Sonda Chtourou			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  1. Démontrer des connaissances solides en termes de système d'exploitation embarqué, architecture de linux embarqué, méthodologie de développement linux embarqué, composants du noyau linux et arborescence du le système de fichiers. 2. Configurer, construire et porter le noyau linux pour différentes plateformes (ordinateur et embarqués); 3. Analyser les problèmes de démarrage du noyau et proposer les éventuelles solutions. 4. Maitriser les différentes techniques de démarrage du noyau Linux. 5. Configurer et construire un noyau linux modulaire et développer des modules dynamiques. 6. Configurer, construire et porter un système de fichiers complet (répertoires, commandes, fichiers de configuration, scripts de démarrage ....) 7. Etudier les différentes approches temps réel (mou et dur) pour le noyau linux embarqué standard. 8. Développer des applications en utilisant les différents services du système d'exploitation (linux et linux temps réel) et estimer les performances.						
<u>Contenu</u>		<b>Cours</b>  1. Linux comme système embarqué 2. Méthodologie de création d'un système d'exploitation linux embarqué 3. Noyau et Pilotes 4. Système de fichiers 5. Linux embarqué temps réel 6. Linux embarqué et FPGA  <b>Travaux pratiques</b> TP1. Compilation et démarrage du noyau linux pour une architecture d'un ordinateur. TP2. Compilation croisée du noyau pour un processeur embarqué. TP3. Noyau linux modulaire et modules dynamiques. TP4. Techniques de démarrage de Linux. TP5. Configuration et construction d'un système de fichiers linux.						

	<p>TP6. Développement des applications en utilisant les différents services du système linux.</p> <p>TP7. Portage de linux embarqué temps réel dur sur un FPGA (<math>\mu</math>Clinux + Xenomai).</p> <p>TP8. Développement des applications en utilisant les différents services du système linux temps réel.</p>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 30 heures et 6 non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	<p>Cours 22.5 heures</p> <p>TP 7.5 heures</p>
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2, 3 et 4.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 5, 6 et 7.</li> </ul> <p>Evaluation des travaux pratiques : Livrables des travaux et des comptes rendus sont demandés suite à chaque TP (Acquis d'apprentissage 2 à 8).</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test écrit à mi-parcours, les étudiants devront démontrer des connaissances solides en termes de rôle, avantages et architecture du système d'exploitation linux embarqué. Ils devront aussi maîtriser les différentes étapes de construction d'un noyau linux pour n'importe quelle plateforme. En cas de problème de démarrage du noyau, les étudiants devront analyser la cause du problème et proposer des solutions. Ils devront aussi maîtriser les différentes techniques de démarrage du noyau linux et connaître leurs avantages, inconvénients et utilisation.</p> <p>Lors du test écrit final (Examen), les étudiants devront maîtriser les différentes étapes de construction d'un noyau linux modulaire et développer des modules dynamiques. Ils doivent aussi connaître les différents répertoires de l'arborescence du système de fichiers linux et leur rôles. Ils doivent aussi démontrer qu'ils ont compris les différentes approches pour rendre le noyau linux embarqué standard temps réel.</p> <p>Dans l'évaluation des TPs, les étudiants doivent démontrer la capacité de configurer un système d'exploitation linux embarqué complet selon différentes contraintes (minimaliste, modulaire, temps réel ...). Ils doivent de plus développer des applications en utilisant les différents services de synchronisation du système d'exploitation.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux Pratiques : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%</p>

<b><u>Pré requis</u></b>	Avoir des bonnes connaissances niveau fondement du système d'exploitation: rôle, processus, synchronisation interprocessus...; Avoir des bonnes connaissances en programmation shell (commandes Unix, Makefiles et scripts shell).
<b><u>Acquis de formation visés</u></b>	PLO3, PLO4, PLO5
<b><u>Références Bibliographiques</u></b>	Catherine Lingxia Wang, "Survey of Embedded OS". Pierre Ficheux, "Linux embarqué", 4ème édition. Patrice Kadionik, "Linux embarqué: le projet uClinux" et "Les Systèmes Embarqués Linux pour l'embarqué". Joel R.Williams "Embedding Linux in a Commercial Product".

## INF3-X-1-3 Programmation temps réel

Groupe de module	INF3-X-1		Intitulé du module	Programmation temps réel		Code	INF3-X-1-3	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD			2				
	TP	X		3				
	Pr			4				
				5			x	
Volume Horaire	30h 22.5hC, 7.5hTP Non présentiel : 10				Coefficient	2		
					Crédits	2		
Responsable du Module	Hajer Krichene			Intervenants	Hajer Krichene			
<u>Acquis d'apprentissage</u>	A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <ol style="list-style-type: none"><li>Développer des connaissances sur les concepts de base des systèmes temps réel, leurs caractéristiques spécifiques, et les défis de leur modélisation.</li><li>Connaître l'architecture type d'un système temps réel et faire la distinction entre la programmation conventionnelle et la programmation temps réel, ainsi qu'entre un système <b>embarqué temps réel</b> et un système <b>distribué temps réel</b>.</li><li>Donner les notions essentielles de la conception des applications embarquées temps réel pour comprendre les outils de développement d'applications des systèmes programmable sur puce (PSoC : <b>Programmable System on Chip</b>). Comme mini PC embarqués nous choisissons l'<b>ARDUINO UNO</b>.</li><li>Mettre l'accent sur l'importance de la présence d'un système d'exploitation temps réel <b>RTOS</b> pour structurer et simplifier la programmation de la partie logicielle d'un système temps réel, ainsi que garantir le comportement temps réel d'une application multitâche. Comme cas d'exemple, nous choisissons le RTOS le plus regardé sur le marché de l'embarqué temps réel « <b>freeRTOS</b> ».</li><li>Connaître les caractéristiques et les mécanismes qui composent le noyau temps réel freeRTOS (gestion de tâches et concurrence, gestion de temps, communication via les queues de messages, synchronisation au moyen les sémaphores, traitement des interruptions, attribution des priorités aux tâches, ordonnancement préemptif et coopératif, etc.).</li></ol>							
<u>Contenu</u>	<b>Cours</b> <ol style="list-style-type: none"><li>Introduction aux systèmes temps réel</li><li>Programmation temps réel sur ARDUINO</li><li>Programmation multitâche temps réel basée sur les RTOS : Le multitâche sous freeRTOS</li><li>Communication et synchronisation inter-tâches sous freeRTOS</li><li>Techniques d'ordonnancement pour le temps réel</li></ol> <b>Travaux pratiques</b> <ul style="list-style-type: none"><li>TD sur l'ordonnancement temps réel pour exploiter les techniques d'ordonnancement statique telles que RM et DM, les techniques d'ordonnancement dynamiques telles qu'EDF et LLF, et l'ordonnancement sous des contraintes de ressources. TDs sur les</li></ul>							



	<p>mécanismes de communication et de synchronisation inter-tâche dans un système temps-réel.</p> <ul style="list-style-type: none"> <li>• Afin de tirer profit des capteurs et actionneurs du monde des microcontrôleurs, et de se familiariser avec les appels système (de tâche, temps, communication, synchronisation, traitement interruptions,...) du noyau temps réel freeRTOS, divers TP de développement sur la carte Arduino UNO et de réalisation des montages associés ont été proposées.</li> </ul>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	face à face (présentiel) 30 H et 10H non présentiel (travail à la maison)
<b><u>Technique d'enseignement</u></b>	Cours 22,5 heures et TP 7,5H
<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2 et 3.</li> <li>• L'épreuve pratique finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3, 4 et 5.</li> </ul> <p>Evaluation du projet : évaluation de niveau atteint des acquis 3, 4 et 5.</p> <p>Evaluation des travaux pratiques : des comptes rendus sont demandés et ainsi corrigés suite à chaque TP.</p>
<b><u>Critères d'évaluation :</u></b>	Au terme des tests (pratique et puis écrit), les étudiants devront démontrer leurs connaissances et compétences théoriques et pratiques autour les techniques de développement d'applications multitâches temps réel sur des systèmes programmables sur puce (la carte programmable ARDUINO UNO est utilisée dans ce cours pour mettre en oeuvre ces techniques de développement et de réalisation), tout en s'appuyant sur les services et outils système fournis par le noyau temps réel freeRTOS.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux pratiques et projet : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test pratique final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	
<b><u>Acquis de formation visés</u></b>	PLO4, PLO5, PLO8
<b><u>Références Bibliographiques</u></b>	<p>[1] Pan, Tianhong, Zhu, Yi. “<i>Designing Embedded Systems with Arduino</i>”, Springer Singapore, 1st Edition, 2018.</p> <p>[2] David J.Russell, “<i>Introduction to Embedded Systems: Using ANSI C and the Arduino Development Environment</i>”, Morgan and Claypool Publishers series 2010.</p> <p>[3] Jeremy Blum, “<i>Exploring Arduino: Tools and Techniques for Engineering Wizardry</i>”, Wiley; 1 edition 2013.</p> <p>[4] <a href="http://Arduino.cc/en">http://Arduino.cc/en</a></p> <p>[5] Simon Landrault ,Hippolyte Weisslinger. “ <i>Premiers pas en informatique embarquée</i> ». Le blog d'Eskimon, Edition 2014.</p>

## INF3-X-1-4 Cloud Computing

Groupe de module	INF3-X-1		Intitulé du module	Cloud Computing		Code	INF3-X-1-4	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD			Mixte			2	
	TP	X					3	
	Pr						4	
							5	x
Volume Horaire		30h 15hC, 15hTP			Coefficient	2		
					Crédits	1		
Responsable du Module		Adel Essafi		Intervenants	Adel Essafi			
Acquis d'apprentissage		<div><div>1. A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :</div><div><div>2. Identifier la valeur ajoutée de l'adoption du cloud pour l'entreprise</div><div>3. Connaître les modèles de services, le modèle de déploiement et le modèle de coût offert par les acteurs cloud</div><div>4. Maitriser le services de stockage</div><div>5. Maitriser le service de base du cloud (le calcul)</div><div>6. Maitriser les bases de données sur le cloud</div><div>7. Surveiller et analyser les performances de l'architecture cloud déployée</div><div>8. Déployer une application découplée (avec un minimum de dépendance possible entre les services)</div><div>9. Optimiser le cout de déploiement de l'architecture cible</div><div>10. Construire une architecture hautement disponible et tolérante aux pannes</div></div></div>						
Contenu		<div><div>Cours</div><div><div>1. Introduction au cloud computing</div><div>2. Le service de stockage</div><div>3. Le service de base du cloud : le calcul</div><div>4. Le service de base de données</div><div>5. Le découplage dans le cloud</div><div>6. Construire une application hautement disponible</div></div></div> <div><div>Travaux pratiques</div><div><div>1. TP1 : création de compte, gestion des utilisateurs et</div><div>2. TP2 : Hébergement d'un site web statique sur le cloud</div><div>3. TP3 : Déploiement d'une application web dynamique sur le cloud</div><div>4. TP4 : Conception d'une application hautement disponible</div><div>5. TP5 : mise en place d'une architecture découplée</div></div></div> <div><div>Projets à réaliser par compétences :</div><div><div>1. Concevoir, réaliser et déployer un site web hautement et mondialement disponible sur le cloud AWS</div></div></div>						

<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Cours 50% TP 50% Travail à la maison, MOOC fournies par l'enseignant pour approfondir les connaissances
<b><u>Technique d'enseignement</u></b>	Cours et TP
<b><u>Méthodes d'évaluation</u></b>	QCM : évaluation des connaissances théorique Contrôle continue (CC): Evaluation TP présentielle Examen : réalisation de projet grandeur nature
<b><u>Critères d'évaluation</u></b>	La moyenne finale va de 0 (minimum) à 20 (maximum) La moyenne minimale de validation est 10 La moyenne finale est calculée selon la formule suivante : QCM : 25% CC : 25% Soutenance de projet : 50%.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une note finale comprise entre 0 et 20
<b><u>Pré requis</u></b>	Administration système et administration réseau Programmation web
<b><u>Acquis de formation visés</u></b>	PLO4, PLO8
<b><u>Références Bibliographiques</u></b>	Matthieu ZAROUK, Cloud Computing Maîtrisez la plate-forme AWS - Amazon Web Services, éditions eni. Janvier 2013, 316 pages Documentation officielle de AWS : <a href="https://docs.aws.amazon.com/">https://docs.aws.amazon.com/</a> Mooc : <a href="https://acloud.guru/learn/aws-certified-solutions-architect-associate">https://acloud.guru/learn/aws-certified-solutions-architect-associate</a>

## INF3-X-2-4 Gestion de projet

Groupe de module	INF3-X-2		Intitulé du module	Gestion de projet		Code	INF3-X-2-4	
Département	Informatique		Filière	Génie Info.	Option			
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD	X		Mixte			2	
	TP						3	
	Pr						4	
							5	x
Volume Horaire		Présentiel : C : 22,5 h TD: 7,5 h Non présentiel : 6 h			Coefficient	2		
					Crédits	2		
Responsable du Module	Ines Bayoudh Saâdi		Intervenants	Ines Bayoudh Saâdi				
<u>Acquis d'apprentissage</u>			A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  <div><div>1.</div><div>Connaître tant les aspects techniques de la gestion de projet (les boites à outils) que ses aspects humains et comportementaux (compétences et responsabilités des acteurs projet) et leurs interactions selon le corpus de connaissances en management de projet : PMBOK</div><div>2.</div><div>Comprendre et maîtriser les différentes démarches de gestion de projet (approches et référentiels)</div><div>3.</div><div>Monter, planifier, piloter et clôturer un projet selon le PMBoK</div></div>					
<u>Contenu</u>			<b>Cours :</b> <b>Chp1 : Introduction à la gestion de projet</b> <b>Chp2 : Organisation du projet</b> <b>Chp3 : Découpage en tâches/activités</b> <b>Chp4 : Estimation des charges</b> <b>Chp5 : Techniques de planification</b> <b>Chp6 : Suivi et contrôle du projet</b> <b>Chp7 : Management de projet selon le PMBOK (version 4)</b>  <b>Projets à réaliser par compétences :</b> les étudiants travaillent sur une étude de cas et en utilisant un logiciel de gestion de projet. Ils reçoivent des <b>conseils / feedbacks par leurs pairs et de leur tuteur</b> . Ceci leur permet d'être à la fois acteur de leur projet, mais également de développer des compétences pour pouvoir évaluer les projets des autres.  <b>Utilisation d'une moulinette de certification PMP :</b> les étudiants utilisent cette moulinette pour se préparer à certification PMP.					
<u>Méthode d'enseignement et d'apprentissage</u>			face à face (présentiel) 30 heures et 6h non présentiel (travail à la maison) Un ensemble de ressources pédagogiques et activités d'apprentissage est mis à la disposition des apprenants dans un espace de cours sur la plateforme d'enseignement à distance de l'UVT ( <a href="http://ent.uvt.rnu.tn">ent.uvt.rnu.tn</a> ), pour les supporter et les préparer à la certification PMP .					
<u>Technique d'enseignement</u>			Cours et TD = 30H					

<b><u>Méthodes d'évaluation</u></b>	<p>Écrit et pratique.</p> <p>Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1,2 et 3.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5, 6 et 7.</li> </ul> <p>Evaluation de l'étude de cas: sera comptabilisée avec le DS(10% de la note du DS)</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à maîtriser le vocabulaire associé à la gestion de projet, les acteurs et l'organisation du projet et les différents types de découpages . L'évaluation portera sur ces différentes parties.</p> <p>Au terme du test final, les étudiants devront résoudre un problème lié à l'estimation des charges, à la planification, au contrôle de projet en calculant et interprétant un ensemble d'indicateurs de performance, ils devront aussi savoir comment clôturer un projet.</p> <p>Lors de projet, les étudiants doivent démontrer leur connaissance et leur compréhension des principes fondamentaux de la gestion de projet. En outre, leur capacité à illustrer leurs connaissances et leur compréhension clairement</p> <p>Enfin, la capacité des étudiants à participer aux discussions en classe avec les enseignants et leurs collègues sera évaluée avec l'étude de cas.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 30% Etude de cas : 10% de la note du DS</p> <p>Test écrit final (Examen) : 70%.</p>
<b><u>Pré requis</u></b>	Aucun
<b><u>Acquis de formation visés</u></b>	PLO3, PLO6
<b>Références Bibliographiques</b>	<ul style="list-style-type: none"> <li>• Bellenger, L., &amp; COUCHAERE, M. J. (1997). Animer et gérer un projet. <i>ESF Editeur</i>.</li> <li>• Maders, H. P., &amp; Clet, É. (2007). <i>Comment manager un projet: les sept facettes du management de projet</i>. Editions Eyrolles.</li> <li>• Morley, C. (1998). <i>Gestion d'un projet système d'information</i>. Inter éditions.</li> <li>• Asquin, A., Falcoz, C., &amp; Picq, T. (2005). <i>Ce que manager par projet veut dire</i>. Éditions d'Organisation.</li> <li>• Guide, A. (2004). Project management body of knowledge (pmbok® guide). In <i>Project Management Institute</i>.</li> </ul>

## Option 1 : GL et Informatique Décisionnelle (GLID)

### INF3-GLID-3-1 Data mining

Groupe de module	INF3-GLID-3		Intitulé du module	Data mining		Code	INF3-GLID-3-1	
Département	Informatique		Filière	Génie Info.	Option	GLID		
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD						2	
	TP	X		Mixte			3	
	Pr						4	
							5	x
Volume Horaire		30h 22.5hC, 7.5hTP			Coefficient		2	
					Crédits		2	
Responsable du Module		Afef Kacem		Intervenants	Afef Kacem			
Acquis d'apprentissage		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable d'assimiler les principes de base de la fouille de données et particulièrement la classification de données et saisir les enjeux de la classification non-supervisée et supervisée.						
Contenu		<b>Cours</b> <ul style="list-style-type: none"><li>• CHAPITRE 1: GENERALITES<ol style="list-style-type: none"><li>1. Fouille de données<ol style="list-style-type: none"><li>1.1. Définition</li><li>1.2. Domaines d'application</li><li>1.3. Processus</li></ol></li><li>2. Donnée<ol style="list-style-type: none"><li>2.1. Définition</li><li>2.2. Types d'attributs</li><li>2.3. Bruit</li><li>2.4. Tâches d'extraction d'information</li></ol></li><li>3. Classification<ol style="list-style-type: none"><li>3.1. Objectif et définitions</li><li>3.2. Domaines d'application</li><li>3.3. Terminologie</li><li>3.4. Mise en œuvre d'une classification</li><li>3.5. Classification supervisée vs. non supervisée</li><li>3.6. Evaluation de la classification</li></ol></li></ol></li><li>• CHAPITRE 2: CLASSIFICATION NON SUPERVISEE<ol style="list-style-type: none"><li>1. Introduction</li><li>2. Méthodes par partitionnement<ol style="list-style-type: none"><li>2.1. Principe</li><li>2.2. K-moyennes</li><li>2.3. K-médoïdes</li></ol></li><li>3. Méthodes hiérarchiques<ol style="list-style-type: none"><li>3.1. Classification hiérarchique ascendante</li><li>3.2. Classification hiérarchique descendante</li></ol></li></ol></li></ul>						

	<ul style="list-style-type: none"> <li>• CHAPITRE 3: CLASSIFICATION SUPERVISEE               <ol style="list-style-type: none"> <li>1. K-PLUS PROCHES VOISINS</li> <li>2. CLASSIFICATION BAYESIENNE</li> <li>3. ARBRE DE DECISION</li> </ol> </li> </ul> <p><b>Travaux pratiques</b></p> <ul style="list-style-type: none"> <li>• INITIATION AU LANGAGE R</li> <li>• INTERFACE WEKA</li> </ul>
<b><u>Méthode d'enseignement et d'apprentissage</u></b>	Face à face (présentiel) 30 heures
<b><u>Technique d'enseignement</u></b>	Cours 22.5 heures TP= 7.5 heures
<b><u>Méthodes d'évaluation</u></b>	<p><u>Ecrit et pratique</u></p> <p>Un test écrit à mi-parcours et un test écrit final sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis des chapitres 1 et 2.</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis des chapitres 3.</li> </ul> <p>Evaluation des travaux pratiques : évaluation du contrôle continu des niveaux atteints d'apprentissage du langage R et de l'interface Weka</p>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur :</p> <ul style="list-style-type: none"> <li>• Niveau d'assimilation des différents concepts liés au datamining et particulièrement à la classification de données.</li> <li>• Niveau de compréhension d'un certain nombre de méthodes de classification non supervisée (k-moyennes, k-médoïdes, classification hiérarchique descendante et ascendante, etc.)</li> </ul> <p>Au terme du test final, les étudiants devront démontrer leur</p> <ul style="list-style-type: none"> <li>• Niveau d'assimilation des concepts et du principe de la classification supervisée (k-plus proches voisins, classification bayésienne, arbre de décision, etc.).</li> </ul> <p>Le test et l'examen TP mettra en épreuve les connaissances acquises dans les séances de travaux pratiques.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux Pratiques : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	Fondements de base en mathématiques
<b><u>Acquis de formation visés</u></b>	PLO2

<b><u>Références</u></b> <b><u>Bibliographiques</u></b>	<p>[1] Christophe Chesneau, « Eléments de classification », Université de Caen, <a href="http://www.math.unicaen.fr/~chesneau/classif-cours.pdf">http://www.math.unicaen.fr/~chesneau/classif-cours.pdf</a></p> <p>[2] Ph. Preux, « Fouille de données : notes de cours », université de Lille 3, 2011.</p> <p>[3] Maurice ROUX Professeur émérite, Algorithmes de classification, Université Paul Cézanne Marseille, France.</p> <p>[4] Lebart, L., Morineau, A., Piron M, Analyse exploratoire multidimensionnelle, Dunod, Paris, 2000.</p> <p>[5] <a href="http://www.fil.univ-lille1.fr/~decomite/ue/APE/tp/tp1/weka2009.pdf">http://www.fil.univ-lille1.fr/~decomite/ue/APE/tp/tp1/weka2009.pdf</a></p>
--	--



## INF3-GLID-4-3 Big Data

Groupe de Module	INF3-GLID-4		Module	BigData		Code	INF3-GLID-4-
Département	INFORMATIQUE		Filière	GINFO	Option		
Type d'enseignement	C	X	Régime	CC		Semestre	5
	TD	X					
	TP			Mixte	X		
	Proj						
Volume horaire/semaine		Présentiel : C : 22,5 TD: 7.5 TP:			Coefficient	2	
		Non présentiel : 6h			Crédit	2	
Responsable du Module		Ahlem Ben Younes		Enseignant(e)s intervenants			
<u>Acquis d'apprentissage</u>		<p>A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :</p> <p><b>Volet 1 :</b> Connaître les fondamentaux en matière de Big Data, le vocabulaire et les fonctionnalités d'une solution Big Data. Identifier les problématiques et les solutions liées au stockage, à la gestion et à l'analyse des environnements à très forte volumétrie. Évaluer l'architecture, les composants, le fonctionnement et les outils du Big Data</p> <p><b>Volet 2 : les Base données NoSQL</b> Connaître les fondamentaux en matière des bases NoSQL. Connaître l'écosystème de NoSQL et les caractéristiques des solutions du marché (MongoDB ,Cassandra, ...). Comprendre les concepts de systèmes distribués et de donnée : cohérence, théorème de CAP... Savoir choisir le modèle de base NoSQL qui répond le plus aux besoins Mettre en œuvre une solution Big Data</p>					
<u>Contenu :</u>		<p><b>Volet 1 : Bigdata</b> <b>Chapitre 1 : Introduction au Big Data</b> 1 Big Data : Faits, Intérêts, Sources et Challenges 2 Qu'est-ce que le Big Data ? 3 Problématique et termes clés 4 Chiffres clés et forces majeurs 5 Importance et défis du Big Data 6 Big Data et décisionnel 7 Approche traditionnelle versus approche Big Data 8 Mise en place du Big Data : des chantiers de grande ampleur</p> <p><b>Chapitre 2 : Hadoop, MapReduce et le Big Data</b> <b>Partie1</b> : Hadoop : Un standard de gestion des données et applications distribuées Problématiques du calcul distribué <b>Partie 2:</b> HDFS : Hadoop Distributed File System <b>Partie 3:</b> MapReduce <i>Travaux Pratique 1: Initialisation à Hadoop et HDFS</i></p> <p><b>Chapitre 3 : Programmation MapReduce sous Hadoop</b> <b>Partie 1</b> : Architecture fonctionnelle, Fonctionnement de MapReduce et Yarn</p>					

	<p><b>Partie 2 :</b> MapReduce du point de vue du développeur Java : Les entrées / Sorties          Programmation MapReduce Hadoop en java</p> <p><b>Partie 3 :</b> Hadoop Streaming  <i>Application 1 (Travaux Pratique 2 et Homework) : Initialisation a Hadoop et MapReduce (Hadoop Streaming)</i></p> <p><b>Chapitre 4 :</b> Hadoop query languages : Traitement des données Big Data avec hadoop  <i>Application2 (Travaux Pratique3) : Prise en main de Hive</i></p> <p><b>Chapitre 5 :</b> Spark          Spark vs Hadoop          Elément de l'API Spark          Exemple de programme pySpark  <i>Application 3 (Travaux Pratique4 et homework) : Initiation Spark</i>  <b>Homework :</b> Écrire et exécuter une application dans Spark</p> <p><b>Volet 2 : Bigdata et NoSQL</b></p> <p><b>Chapitre 6:</b> Big Data et NoSQL  <b>Partie1 :</b> Mouvement NoSQL  <b>Partie2 :</b> Taxonomie des bases NoSQL  <b>Partie3 :</b> Avantages / Inconvénients des bases NoSQL</p> <p><b>Chapitre 7 :</b> NoSQL orientées colonnes: Cassandra          Partitionnement dans Cassandra          Replication dans Cassandra          Consistance dans Cassandra          Gestion de données et des objets dans Cassandra</p> <p><b>Travaux Pratique 4 :</b> NoSQL avec Cassandra</p> <p><b>Chapitre 8 :</b> NoSQL – orientées Documents : MongoDB          Les Bases de données avec MongoDB          Définition d'un schéma de données avec MongoDB          Réplication dans MongoDB</p> <p><b>Travaux Pratique 5 :</b> Installation et Configuration de MongoDB</p> <p><b>Chapitre 9 :</b> NoSQL orientées graphe : Neo4j  <b>Travaux Pratique 6 :</b> NoSQL avec Neo4j</p> <p><b>Projets à réaliser par compétences:</b> les étudiants travaillent sur une étude de cas ( un jeu de données qu'ils devront choisir eux même dans l'un des sites qui proposent des datasets ) et en utilisant des outils du Big Data qui répond le plus aux besoins du problème choisi. Ils reçoivent des <b>conseils / feedbacks par leurs pairs et de leur tuteur</b>. Ceci leur permet d'être à la fois acteur de leur projet, mais également de développer des compétences pour pouvoir évaluer les projets des autres.</p>
<b>Méthode d'enseignement et d'apprentissage:</b>	face à face (présentiel) 30 heures et 10 H non présentiel
<b>Technique d'enseignement :</b>	Cours 22.5 heures et TD 7.5H
<b>Méthodes d'évaluation :</b>	Écrit et pratique. Un test écrit à mi-parcours (D.S) et un test écrit final (Examen) sont prévus. L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1-9

	<p>L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 4, 5, 6 et 7.</p> <p>Evaluation de l'étude de cas: sera comptabilisée avec le DS (20% de la note du DS)</p>
<b><u>Critères d'évaluation :</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité : à maîtriser les fondamentaux associé aux Bigdata,, à évaluer l'architecture, les composants, le fonctionnement et les outils du Big Data, à utiliser Hadoop et développer des programmes MapReduce. L'évaluation portera sur ces différentes parties.</p> <p>Au terme du test final, les étudiants devraient démontrer leur capacité : à maîtriser les fondamentaux en matière du BigData et des bases NoSQL à évaluer les différentes solutions NoSQL du marché (MongoDB, Cassandra, ...). Ils devront être capables : de choisir le modèle de base NoSQL qui répond le plus aux besoins, et de maîtriser l'utilisation d'une base NoSQL (MongoDB, Cassandra)</p> <p>Lors du projet, les étudiants doivent démontrer leurs connaissances et leurs compréhensions des principes fondamentaux du Bigdata et NoSql. En outre, leur capacité à illustrer leurs connaissances et leur compréhension clairement dans la mise en œuvre d'une solution Bigdata.</p> <p>Enfin, la capacité des étudiants à participer aux discussions en classe avec les enseignants et leurs collègues sera évaluée avec l'étude de cas.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale :</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Test écrit à mi-parcours (D.S) : 1/3 Etude de cas : 20% de la note du DS</p> <p>Test écrit final (Examen) : 2/3</p>
<b><u>Pré requis:</u></b>	Avoir des bonnes connaissances en : programmation, BI (informatique décisionnelle), Cloud, Base de données
<b><u>Acquis de formation visés</u></b>	PLO3, PLO4, PLO7, PLO8
<b><u>Références Bibliographiques</u></b>	<p><b>Cours Big Data Analytics – Lesson 1: What is Big Data, IBM, Big Data University</b></p> <p><b>Cours «D'énormes gisements de création de valeurs » Xavier Dalloz</b></p> <p><b>Cours des Bases de Données à Big Data, Professeur Serge Miranda</b></p> <p><b>Département Informatique Université de Nice Sophia Antipolis</b></p> <p><b>Livres Blancs :</b></p> <p>Top 5 Considerations when evaluating NOSQL Databases, MongoDB White Paper, Juin 2013.</p> <p>Big data et machine learning ; Manuel du la data science , Pirmin Lemberger, Marc Batty, Médéric Morel, Jean-Luc Raffaëlli, <i>Dunod 2016</i></p> <p><b>Sites:</b></p> <p>Why NOSQL – Part 1 – CAP Theorem : <a href="http://bigdatanerd.wordpress.com/2011/12/08/why-nosql-part-1-cap-theorem/">http://bigdatanerd.wordpress.com/2011/12/08/why-nosql-part-1-cap-theorem/</a></p> <p>DATANERD</p> <p>NOSQL : 5 minutes pour comprendre : <a href="http://blog.neoxia.com/nosql-5-minutes-pour-comprendre/">http://blog.neoxia.com/nosql-5-minutes-pour-comprendre/</a> NEOXIA</p> <p>NOSQL Europe : Bases de données orientées colonnes et Cassandra : <a href="http://blog.xebia.fr/2010/05/04/nosql-europe-bases-de-donnees-orientees-colonnes-et-cassandra/">http://blog.xebia.fr/2010/05/04/nosql-europe-bases-de-donnees-orientees-colonnes-et-cassandra/</a> XEBIA</p> <p>Une base Cassandra : <a href="http://www-igm.univ-mlv.fr/~dr/XPOSE2010/Cassandra/">http://www-igm.univ-mlv.fr/~dr/XPOSE2010/Cassandra/</a></p> <p>Documentation officielle MongoDB: <a href="http://docs.mongodb.org/MongoDB">http://docs.mongodb.org/MongoDB</a></p> <p>Documentation officielle Neo4j: <a href="https://neo4j.com/docs/">https://neo4j.com/docs/</a></p>

## Option 2 : Nouvelles Technologies et Sécurité (NTS)

### INF3-NTS-3-3 Internet of Things (IoT)

Groupe de module	INF3-NTS-3		Intitulé du module	Internet of Things (IoT)		Code	INF3-NTS-3-3	
Département	Informatique		Filière	Génie Info.	Option	NTS		
Type d'enseignement	C	X	Régime	CC	X	Semestre	1	
	TD				2			
	TP	X		Mixte			3	
	Pr						4	
							5	X
Volume Horaire		30h 15hC, 15hTP			Coefficient		2	
					Crédits		2	
Responsable du Module		Lamia Ben Azouz		Intervenants	Lamia Ben Azouz			
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de :  <div><div>1.</div><div>Connaître les domaines d'application IoT</div></div> <div><div>2.</div><div>Connaître les architectures IoT</div></div> <div><div>3.</div><div>Choisir les modèles IoT adéquats pour les types d'application</div></div> <div><div>4.</div><div>Choisir les technologies courte distance adéquates</div></div> <div><div>5.</div><div>Choisir les technologies longues distance adéquates</div></div>						
<u>Contenu</u>		<b>Cours</b> <ul style="list-style-type: none"><li>• Introduction</li><li>• Les Applications IoT</li><li>• Les capteurs</li><li>• Architectures</li><li>• Modèle de communication</li><li>• Technologies de communication courte distance</li><li>• 6LowPan</li><li>• Technologies de communication Longue distance</li></ul> <b>Travaux Pratique</b> <ul style="list-style-type: none"><li>•     Modèle Device to Gateway: Smart Home</li><li>•     Modèle Device to cloud:     Smart Home   Véhicule connecté</li><li>•     Modèle Device to Device: Smart Garage</li></ul>						
<u>Méthode d'enseignement et d'apprentissage</u>		face à face (présentiel) 30 heures et 12 H non présentiel (travail à la maison)						
<u>Technique d'enseignement</u>		Cours =15 H TP=15 H						

<b><u>Méthodes d'évaluation</u></b>	<p>Écrit, pratique et projet</p> <p>Un test écrit à mi-parcours (mini projet) et un test écrit final (pratique + écrit) sont prévus.</p> <ul style="list-style-type: none"> <li>• L'épreuve écrite de mi-parcours sera consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 1, 2, 3 ,4</li> <li>• L'épreuve écrite finale est consacrée à l'évaluation du niveau d'atteinte des acquis d'apprentissage 3, 4 et .5</li> </ul>
<b><u>Critères d'évaluation</u></b>	<p>Lors du test à mi-parcours, les étudiants devraient démontrer leur capacité à concevoir et implémenter sur l'environnement de simulation Packet Tracer des applications IoT qui s'adaptent à l'environnement Tunisien .</p> <p>Au terme du test final, les étudiants doivent démontrer leurs capacité à configurer les objets, les équipements réseau et à programmer sur les micro contrôleur pour répondre à un use case. Ils doivent montrer leur capacité à choisir les modèles IoT et technologies adéquats pour des use cases spécifiques.</p>
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	<p>La moyenne finale va de 0 (minimum) à 20 (maximum)</p> <p>La moyenne minimale de validation est 10</p> <p>La moyenne finale est calculée selon la formule suivante :</p> <p>Travaux Pratiques : 25%</p> <p>Test écrit à mi-parcours (D.S) : 25%</p> <p>Test écrit final (Examen) : 50%.</p>
<b><u>Pré requis</u></b>	Avoir des bonnes connaissances en réseaux locaux ,TCP/IP, certif 1 et certif 2
<b><u>Acquis de formation visés</u></b>	PLO1, PLO4, PLO5, PLO8
<b><u>Références Bibliographiques</u></b>	<p>1- Dave Evans, "The Internet of Things How the Next Evolution of the Internet Is Changing Everything", White Paper, Cisco Internet Business Solutions Group , 2011</p> <p>2- . Mats Andersson , "Short range low power wireless devices and Internet of Things (IoT) ", White paper,</p> <p>3- RFC 7452, " Architectural Considerations in Smart Object Networking ", 2015.</p> <p>4.. <a href="https://www.sigfox.com/en/sigfox-iot-technology-overview">https://www.sigfox.com/en/sigfox-iot-technology-overview</a></p>

## INF3-NTS-4-1 Réseaux et applications Multimédia

Groupe de Module	INF3-NTS-4		Module	Réseaux des applications multimédia		Code	INF3-NTS-4-1
Département	INFORMATIQUE		Filière	GINFO	Option	NTS	
Type d'enseignement	C	22,5 h	Régime	CC		Semestre	5
	TD	7,5 h		Mixte	X		
	TP	0h					
	Proj						
Volume horaire/semaine		Présentiel : C : 1,5 TD : 0,5 TP : 0 <b>Non présentiel :</b>			Coefficient	2	
					Crédit	2	
Responsable du Module		Lamia Ben Azouz		Enseignant(e)s intervenants	Mohamed Aymen Messaoudi		
<u>Acquis d'apprentissage</u>		A l'issue de ce module d'enseignement, l'étudiant(e) sera capable de : <ul style="list-style-type: none"><li>- Se familiariser avec les technologies Multimédia</li><li>- S'initier aux codecs et protocoles multimédia ;</li><li>- Connaître les caractéristiques et les infrastructures des réseaux Multimédia</li><li>- Apprendre à concevoir une solution de plateforme Multimédia sur un réseau IP.</li><li>- Savoir dimensionner et choisir les équipements nécessaires</li><li>- Sécuriser une plateforme multimédia connaissant les menaces et les vulnérabilités.</li></ul>					
<u>Contenu</u>		<b>Chapitre 1 : La communication téléphonie classique</b> Le Réseau téléphonique commuté <b>Chapitre 2 : La voix sur IP et la téléphonie sur IP</b> La Voix sur IP (Voice over IP, VoIP) Les Codecs (G711, G722, G729..... ) L'infrastructure VoIP Avantage et inconvénients de la VoIP La téléphonie sur IP (ToIP) Normes et protocoles Multimédia <b>Chapitre 3 : Conception et proposition d'une solution VoIP</b> Analyse d'une infrastructure générale du réseau Conception d'une solution VoIP Sécurisation de l'infrastructure multimédia					
<u>Méthode d'enseignement et d'apprentissage</u>		face à face (présentiel) 30 heures					

<b><u>Technique d'enseignement</u></b>	Cours 22,5 heures et TD 7,5 H
<b><u>Méthodes d'évaluation</u></b>	<ul style="list-style-type: none"> <li>• Un devoir surveillé en mi période de formation</li> <li>• Un examen final à la fin de la formation</li> </ul>
<b><u>Critères d'évaluation</u></b>	Lors du devoir surveillé de mi-parcours les étudiants doivent maîtriser les notions étudiées au niveau des deux premiers chapitres. En examen final l'étudiant est évalué et tester sur la totalité du cours.
<b><u>Mesure d'évaluation</u></b>	Attribution d'une moyenne finale
<b><u>Critère d'attribution de la moyenne finale</u></b>	La moyenne finale va de <b>0</b> (minimum) à <b>20</b> (maximum) La moyenne minimale de validation est <b>10</b> La moyenne finale est calculée selon la formule suivante : D.S à mi-parcours : 1/3 Examen final : 2/3
<b><u>Pré requis</u></b>	Connaissances de base en réseau
<b><u>Acquis de formation visés</u></b>	PLO1, PLO4, PLO8
<b><u>Références Bibliographiques</u></b>	<a href="http://cisco.com">http://cisco.com</a> <a href="http://www.avaya.fr">http://www.avaya.fr</a> . <a href="http://www.en.voipforo.com">http://www.en.voipforo.com</a> <a href="http://www.frameip.com">http://www.frameip.com</a> <a href="http://www.testeur-voip.com">http://www.testeur-voip.com</a> .