

Parcours Aigle, LMD4

Architecture et Ingénierie du loGicieL et du wEb

Clémentine Nebut, Pierre Pompidor

9 avril 2015

Cette formation vise à former des cadres en informatique dans le domaine du développement logiciel ce qui cible principalement les métiers suivants :

- Architecte logiciel
- Chef de projet logiciel
- Architecte des systèmes d'informations

Par ailleurs, la formation, adossée au LIRMM (Laboratoire d'Informatique, de Micro-électronique et de Robotique de Montpellier), permet également une poursuite d'étude en doctorat dans l'optique de devenir enseignant-chercheur ou chercheur, dans l'enseignement supérieur et la recherche, ou dans la R&D d'entreprise.

1 Semestre 1

- Anglais. Equipe pédagogique : le SCEL (contact : Anne Heaps).
- 3 UEs obligatoires :
 - Bases de données avancées. Equipe pédagogique : Anne-Muriel Chifolleau.
 - Compilation et interprétation. Equipe pédagogique : Roland Ducournau, Mathieu Lafourcade.
 - Ingénierie Logicielle. Equipe pédagogique : Christophe Dony et Clémentine Nebut.
- 2 UEs parmi :
 - Présentation des données du web. Equipe pédagogique : Pierre Pompidor.
 - Réseaux et communication. Equipe pédagogique : Hinde Bouziane.
 - ou 1 UE d'ouverture parmi la liste d'UEs suivante :
 - Algorithmique du texte
 - Complexité algorithmique
 - Intelligence Artificielle
 - Interaction Homme Machine
 - Programmation Orientée Agents

2 Semestre 2

- TER
- 3 parmi 4 :
 - Architectures n-tiers. Equipe pédagogique : Clémentine Nebut, Djamel Seriai.
 - Conduite de projets. Equipe pédagogique : Eric Bourreau, Clémentine Nebut.
 - Développement logiciel pour mobiles. Equipe pédagogique : Abdelhak-Djamel Seriai.
 - Spécifications formelles, vérification, validation. Equipe pédagogique : Clémentine Nebut, Marianne Huchard, intervenants extérieurs.
- 2 UE d'ouverture parmi une liste d'UEs parmi la liste d'UEs suivante :
 - une UE coeur AIGLE
 - Algorithmique d'exploration et de mouvement
 - Algorithmique distribuée
 - Epistémologie de l'informatique
 - Extraction de Connaissances dans les Données

- Graphes et structures
- Ingénierie des connaissances
- Technologies de la langue
- Web sémantique et social

3 Semestre 3

- 5 parmi 8 :
 - Composants et réutilisation. Equipe pédagogique : Christophe Dony, Chouki Tibermacine.
 - Développement logiciel pour l'embarqué. Equipe pédagogique : Abdelhak-Djamel Seriai.
 - E-applications. Equipe pédagogique : Anna-Sophie Fiston.
 - Evolution et restructuration. Equipe pédagogique : Djamel Seriai, Marianne Huchard, Chouki Tibermacine.
 - Ingénierie des modèles. Equipe pédagogique : Clémentine Nebut, Marianne Huchard.
 - Métaprogrammation et réflexivité. Equipe pédagogique : Roland Ducournau, Christophe Dony.
 - Modélisations formelles pour le génie logiciel. Equipe pédagogique : Marianne Huchard, Thomas Lambolais, Clémentine Nebut.
 - Spécification et implémentation des langages à objets et à composants. Equipe pédagogique : Christophe Dony, Roland Ducournau, Djamel Seriai, Chouki Tibermacine.
- 1 UE d'ouverture parmi une liste d'UEs parmi la liste d'UEs suivante :
 - une UE coeur AIGLE
 - Administration de bases de données
 - Extraction de Connaissances Avancées
 - Gestion des connaissances pour l'aide à la décision
 - Jeux sérieux
 - Nouvelles approches pour la persistance des données
 - Traitement de données à grande échelle
 - Son et Musique

4 Semestre 4

- enjeux juridiques et sociétaux de l'informatique
- Orientation pro
 - Stage industriel
- Orientation recherche
 - étude bibliographique
 - stage académique

5 Contenu des UEs proposées par AIGLE

5.1 Semestre 1

Compilation et interprétation. (16.5h CM ; 16.5h TD ; 16.5h TP) Cette UE étudie de façon pratique les divers procédés qui peuvent être appliqués à un programme informatique pour le rendre exécutable sur une machine. Seront particulièrement étudiés : la compilation (transformation d'un langage de haut-niveau dans un langage de bas niveau), l'interprétation (exécution dynamique d'un programme) et les machines virtuelles. Les étudiants réaliseront en TP et projet la construction progressive d'un compilateur ou interprète complet du langage Scheme ou Lisp.

Ingénierie Logicielle. (16.5h CM ; 16.5h TD ; 16.5h TP) Ce module étudie d'une part les schémas de réutilisation par spécialisation, par composition et par fonctions d'ordre supérieur en programmation par objet. Sont abordées également les notions de framework, et de schéma de conception (étude des principaux schémas). Ce module aborde d'autre part la conception complète d'un logiciel, des spécifications (avec plusieurs vues) au test, en passant par la cristallisation de l'architecture par applications de patrons de conception. Ce module introduit de plus les contrats en programmation.

Présentation des données du web. (16.5h CM ; 16.5h TD ; 16.5h TP) Ce module aborde le tiers présentation sous différents angles :

- XML (schémas...) et transformations XSL (pré-requis du module Web sémantique et social) ;
- bases de données XML ;
- fonctionnalités/librairies avancées de HTML5 (stockage de données sur le client, web sockets ...) ;
- médiatisation de données avec HTML5 (2D et 3D) (en aval d'importat. de fichiers XML/JSON). (cette dernière partie se fera en coordination avec l'enseignement dispensé en IHM).

Réseaux et communication (15h CM ; 15h TD ; 19.5h TP) L'objectif de cette UE est la maîtrise de quelques concepts et techniques de communication intra-système et dans les systèmes distribués. Les étudiants mettent en application ces concepts à travers la conception et la mise en œuvre d'applications intégrant le volet réseau, programmation concurrente, communication et synchronisation (en C/C++). Le contenu se divise en deux parties :

- Communication et synchronisation entre processus et processus légers : mémoires partagées, files de messages, sémaphores, etc.
- Programmation distribuée : client-serveur (utilisation avancée des sockets, serveurs itératifs et concurrents), RPC, systèmes de fichiers distribués, etc.

5.2 Semestre 2

Architectures n-tiers. (15h CM ; 15h TD ; 15h TP) Les applications actuelles deviennent de plus en plus complexes, et intègrent souvent plusieurs parties communicantes. Les architectures n-tiers ont pour but de gérer ce type d'application (communication et échange de données entre les systèmes, persistance et partage des données, ...). Ce module donne les principes des architectures n-tiers, et différentes technologies sous-jacentes, en Java et en .NET :

- objets distribués et intergiciels à objets (RMI, .NET remoting, Corba)
- services web (SOAP, WSDL)

Conduite de projets. (15h CM ; 15h TD ; 15h TP) Ce module aborde différents enjeux et techniques de gestion de projet. Ce module aborde également la vision qualité appliquée au monde logiciel et insiste sur la diversité de la fonction de tests, et sur la gestion des exigences. Une introduction de CMMI est donnée. Les méthodes agiles sont présentées.

Développement logiciel pour mobile. (15h CM ; 15h TD ; 15h TP) L'objectif de ce cours est de présenter les bases permettant de comprendre et de maîtriser le développement logiciel sur des plateformes mobiles (téléphone, tablette, etc.) de manière générale, et ceux liés aux systèmes Android en particulier. Parmi les aspects abordés : Les propriétés et catégories des systèmes embarqués, principes et architectures des OS mobiles, les grands principes des langages de programmation pour mobile, l'architecture et les principaux composants des applications mobiles, les modèles de structuration des interfaces utilisateurs sur mobiles, la gestion des événements, la gestion des ressources embarquées, la gestion des processus et du multi-tâches, la gestion de la connectivité, la gestion des capteurs, la gestion de la persistance des données sur mobile, etc. En plus du cours magistral et des travaux pratiques, et dans la cadre d'un mini-projet par équipe, il est demandé aux étudiants de développer une application mobile permettant de répondre à un besoin réel.

Spécifications formelles, vérification, validation. (15h CM ; 15h TD ; 15h TP) L'objectif de ce module est de fournir des bases théoriques pour le génie logiciel : spécifications formelles, techniques de vérification, fondements du test de logiciels.

5.3 Semestre 3

e-applications (15h CM ; 15h TD ; 15h TP) Ce module aborde les aspects liés au e-commerce et e-business. Sont abordés notamment :

- Architecture d'un site marchand
- Paiement électronique
- Conception d'un site vendeur
- Marketing électronique
- Intelligence économique et veille
- Conception d'applications vocales

- Intégration de contenu (mashup)

Composants et réutilisation (15h CM; 15h TD; 15h TP) Compétences : Comprendre et mettre en œuvre les principes de réutilisation et d'assemblage de composants. Connaissances :

- Frameworks , Schémas de Conception, Aspects
- Composants assemblables (Java Beans)
- Composants Distribués EJB (Architecture J2EE) – Grid Services
- Intégration rapide de Composants.

Développement Logiciel pour l'embarqué. (15h CM; 15h TD; 15h TP) Ce cours constitue une suite du cours "Développement Logiciel pour Mobile" (du Master 1). C'est un cours pro/recherche divisé en deux parties. La première partie présente quelques problématiques recherche liées au développement de logiciels sur mobiles telles que les systèmes sensibles au contexte et le besoin de l'adaptation dynamique, la programmation multi-plateformes pour mobiles et les solutions proposées telles que la programmation à base de scripts, IDM/DSL, ligne de produits, etc. La seconde partie est constituée d'un projet.

Evolution et restructuration. (15h CM; 15h TD; 15h TP) Cette UE aborde les aspects techniques et gestionnaires essentiels de la maintenance et de l'évolution des logiciels. Une vue d'ensemble sera présentée des principes, des techniques, des outils et des meilleures pratiques pour l'évolution de logiciel. A titre d'exemples seront abordés les systèmes patrimoniaux et leur migration, la rétro-ingénierie et la compréhension des programmes, la réingénierie des logiciels, l'évolution transformationnelle, la refactorisation et la restructuration, la propagation des changements et l'analyse des impacts, etc.

Ingénierie des modèles. (15h CM; 15h TD; 15h TP) L'Ingénierie dirigée par les Modèles (IDM) est une approche de développement logiciel qui consiste à mettre les modèles et leurs métamodèles au centre du processus. Dans ce module, nous montrons les enjeux liés à l'Ingénierie des modèles (capitalisation du métier, indépendance vis à vis des technologies, définition de transformations de modèles). Nous positionnons l'IDM par rapport à l'initiative MDA (Model Driven Architecture) de l'OMG. Ces concepts sont mis en oeuvre au travers de l'étude du métamodèle UML, de l'utilisation du langage de contraintes OCL pour la spécification de métamodèles, et de l'implémentation de transformations de modèles avec des langages spécifiques. Les compétences acquises lors de ce modules sont :

- maîtrise des différents niveaux de modélisation et métamodélisation,
- capacité à comprendre un métamodèle existant et en concevoir un nouveau,
- capacité à développer des transformations de modèles.

Métaprogrammation et réflexivité. (15h CM; 15h TD; 15h TP) Le module donne les bases de la méta-modélisation objet qui est à la base d'UML et de l'Ingénierie Dirigée par les Modèles. Application aux classes et propriétés pour l'analyse de l'héritage multiple. Introduction aux méta-classes et à la méta-programmation dans différent contextes : Smalltalk, CLOS, Python, Java (OpenJava, Javassist), C++ (OpenC++), Scala.

Modélisations formelles pour le génie logiciel. (13h CM; 26h TD) Une partie de ce module s'intéresse aux spécifications des systèmes réactifs. Les systèmes logiciels ou à composante logicielle intègrent des aspects comportementaux. Ce sont souvent des systèmes réactifs (en interaction constante avec leur environnement, à une vitesse imposée par l'environnement) et parallèles. Ils doivent accepter certaines opérations dans certaines circonstances (vivacité), mais, pour des raisons de sûreté, en refuser à d'autres moments. Ce module s'intéresse à la modélisation de tels systèmes, en phases de spécification puis de conception. Il s'intéresse également à la vérification des modèles obtenus. La seconde partie de ce module variera en fonction des intervenants extérieurs et abordera des thèmes comme la vérification de l'assemblage de composants, l'usage de l'Analyse Formelle de Concepts en génie logiciel, etc (à définir plus précisément).

Spécification et implémentation des langages à objets et à composants (13h CM; 26h TD) Ce module se compose de deux parties, l'une concerne les objets, et l'autre les composants. La partie Objets de ce module a pour objet de dresser un panorama des techniques utilisées pour implémenter les langages à objets en typage statique, ainsi que d'étudier certains mécanismes spécifiques — comme l'héritage multiple et les modules — pour en faire des spécifications correctes et en examiner l'implémentation. Le module s'intéresse autant à des langages classiques (C++, JAVA, EIF-

FEL) qu'à des langages de laboratoires comme SCALA développé à l'EPFL ou PRM développé au LIRMM.