

Master informatique parcours AIGLE

Architectures et InGénierie du Logiciel et du wEb

Semestre 1

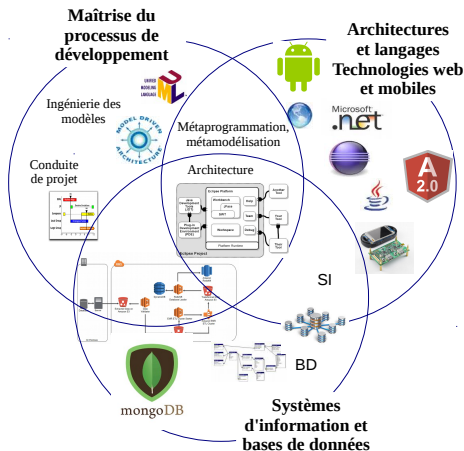
Clémentine Nebut (nebut@lirmm.fr)
Pierre Pompidor (pompidor@lirmm.fr)

Faculté des sciences UM

2 Septembre 2020



La spécialité AIGLE



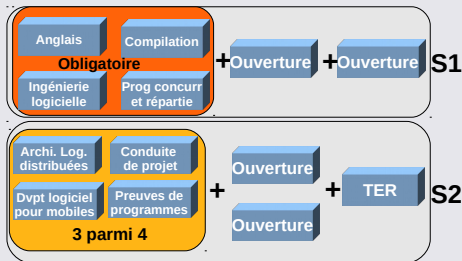
2 ans, 4 semestres

- 30 ECTS à acquérir par semestre
- non compensation annuelle des semestres
- compensation semestrielle des UEs
- redoublement possible (une fois par année) mais non souhaitable !

Les unités d'enseignement

- Renseignements disponibles sur l'offre de formation
- Les modalités de contrôle des connaissances varient d'une UE à l'autre (examen, TP, contrôle continu)

M1



M2

- Début de la nouvelle habilitation : master GL !

Objectifs

- Conception et développement d'architectures logicielles
 - architectures orientées services
 - architectures web
 - urbanisation (architectures des systèmes d'informations)
- Maîtrise de la conception et du développement par objets
- Maîtrise des technologies web
- Maîtrise de l'informatique embarquée (pour mobiles...)

Métiers visés

- Architecte des systèmes d'informations
- Architecte logiciel
- Chef de projet logiciel
- Enseignant-chercheur / chercheur

Objectifs

- Conception et développement d'architectures logicielles
 - architectures orientées services
 - architectures web
 - urbanisation (architectures des systèmes d'informations)
- Maîtrise de la conception et du développement par objets
- Maîtrise des technologies web
- Maîtrise de l'informatique embarquée (pour mobiles...)

Métiers visés

- Architecte des systèmes d'informations
- Architecte logiciel
- Chef de projet logiciel
- Enseignant-chercheur / chercheur

2 ans, 4 semestres

- 30 ECTS à acquérir par semestre
- non compensation annuelle des semestres
- compensation semestrielle des UEs
- redoublement possible (une fois par année) mais non souhaitable !

Les unités d'enseignement

- Renseignements disponibles sur l'offre de formation
- Les modalités de contrôle des connaissances varient d'une UE à l'autre (examen, TP, contrôle continu)
- Attention, mise en place de blocs d'UE avec des moyennes minimales requises.

- Anglais (HMIN101L)
- Principes de la programmation concurrente et répartie (HMIN105M)
- Compilation et Interprétation (HMIN104)
- Ingénierie logicielle (HMIN102)
- et 2 modules parmi :
 - Données du web (HMIN103)
 - Entrepôts de données et big-data (HMIN122M)
 - Interactions Homme-Machine (HMIN110)
 - Programmation orientée agents (HMIN108)
 - Algorithme du texte (HMSN116M)

Blocs pour ce semestre

- Bloc 1 : note ≥ 8
 - Anglais
- Bloc 2 : note ≥ 10
 - Principes de la programmation concurrente et répartie
 - Compilation et interprétation
 - Ingénierie Logicielle

Dates importantes

- Aujourd'hui : Choix des UEs : remplissez le formulaire d'ici ce soir !
- Lundi 7 septembre, 8h : Début des enseignements
- Lundi 14 septembre, 11h30 : Choix **définif** des UEs

- Merci aux étudiants salariés de le signaler
- Consigne pour les rapports remis lors du cursus
- Rappel des règles de courtoisie dans les courriels et les forums
- Séparation cursus Pro/Recherche
- Délégués : des volontaires (ou tirage au sort) → on y revient en fin de réunion

Emploi du temps

AIGLE - emploi du temps S1					
	lundi	mardi	mercredi	jeudi	vendredi
08:00 - 09:30	HMIN101L - Anglais	HMIN102 - Ingénierie logicielle	HMIN104 - Compilation et interprétation	HMIN110 - Interaction homme-machine	HMIN103 - Données du web
09:45 - 11:15					
11:30 - 13:00					
13:15 - 14:45	HMIN107 - Intelligence artificielle	HMIN105M - Principes de la programmation concurrente et répartie	HMIN108 - Programmation orientée agents	HMIN121M - Algorithmique du texte	HMIN122M - Entrepôts de données et big-data
15:00 - 16:30					
16:45 - 18:15					
18:30 - 20:00					

Description

- Conception logicielle avancée
- Réutilisation et de réutilisabilité du logiciel
- Validation par le test

Objectifs

- Schémas de réutilisation objet.
- Design patterns : analyse et conception UML, compréhension et mise en oeuvre.
- Frameworks, plugins, lignes de produits.
- Bases du test de logiciels.

Description

- Conception logicielle avancée
- Réutilisation et de réutilisabilité du logiciel
- Validation par le test

Objectifs

- Schémas de réutilisation objet.
- Design patterns : analyse et conception UML, compréhension et mise en oeuvre.
- Frameworks, plugins, lignes de produits.
- Bases du test de logiciels.

Compilation et Interprétation (HMIN104)

David Delahaye et Mathieu Lafourcade

Articulation avec le cours de L3

Les cours de compilation s'intéressent en général à l'analyse lexico-syntaxique des langages. Ce cours-ci prend le relais en se focalisant sur l'exécution des programmes.

Transformation de programmes

- transformation d'un langage source à un langage cible.
- évaluation directe de programmes.
- transformation du langage LISP en :
 - un langage intermédiaire (évaluation efficace)
 - un langage de machine virtuelle (processeurs physiques)
 - un langage de machine virtuelle à pile

Projet

Un projet en groupe permet aux étudiants de programmer de façon réaliste la totalité des chaînes d'exécution (évaluation de LISP ou du langage intermédiaire, compilation vers les langages intermédiaires et de machine virtuelle, évaluation des machines virtuelles).

Compilation et Interprétation (HMIN104)

David Delahaye et Mathieu Lafourcade

Articulation avec le cours de L3

Les cours de compilation s'intéressent en général à l'analyse lexico-syntaxique des langages. Ce cours-ci prend le relais en se focalisant sur l'exécution des programmes.

Transformation de programmes

- transformation d'un langage source à un langage cible.
- évaluation directe de programmes.
- transformation du langage LISP en :
 - un langage intermédiaire (évaluation efficace)
 - un langage de machine virtuelle (processeurs physiques)
 - un langage de machine virtuelle à pile

Projet

Un projet en groupe permet aux étudiants de programmer de façon réaliste la totalité des chaînes d'exécution (évaluation de LISP ou du langage intermédiaire, compilation vers les langages intermédiaires et de machine virtuelle, évaluation des machines virtuelles).

Compilation et Interprétation (HMIN104)

David Delahaye et Mathieu Lafourcade

Articulation avec le cours de L3

Les cours de compilation s'intéressent en général à l'analyse lexico-syntaxique des langages. Ce cours-ci prend le relais en se focalisant sur l'exécution des programmes.

Transformation de programmes

- transformation d'un langage source à un langage cible.
- évaluation directe de programmes.
- transformation du langage LISP en :
 - un langage intermédiaire (évaluation efficace)
 - un langage de machine virtuelle (processeurs physiques)
 - un langage de machine virtuelle à pile

Projet

Un projet en groupe permet aux étudiants de programmer de façon réaliste la totalité des chaînes d'exécution (évaluation de LISP ou du langage intermédiaire, compilation vers les langages intermédiaires et de machine virtuelle, évaluation des machines virtuelles).

Principes de la programmation concurrente et répartie (HMIN101M)

- Cours 15h, TD 15h, TP 19.5h

Contenu : programmation concurrente et communications intra et inter systèmes

- Communications inter-processus (IPC) : Files de messages, mémoire partagée et ensembles de sémaphores
- Activités dans les processus (threads), parallélisme et synchronisation
- Programmation client-serveur (utilisation avancée des sockets et serveurs itératifs et concurrents)
- RPC (appel de procédure à distance)



Principes de la programmation concurrente et répartie (HMIN101M)

A quoi ça sert ?

- Applications multi-média, jeux et traitement d'images / vidéos
- Application distribuées (P2P, réseaux sociaux, etc.)
- Simulations scientifiques (météo, etc.)
- Administration système et réseaux

Pré-requis

- Cours système L3 (notions de processus et sémaphores)
- Cours réseaux L3 (sockets et client-serveur)
- Langage C;

Programmation par agents (HMIN108)

- Concepts de base de la programmation par acteurs et agents : asynchronisme, parallélisme, distribution massive
- Usage des continuations locales vs gestion des tâches en cours
- Architectures classiques d'agents (réactives, BDI, subsomption)
- Langages de programmation d'agents
- Protocoles classiques de coordination et de négociation
- Utilisation de rôles et de groupes pour la réalisation d'applications
- Gestion de la distribution d'applications multi-agents
- Utilisation des techniques multi-agents pour faciliter la programmation d'applications ouvertes, évolutives et distribuées

Entrepôts de données et big-data

- Partie 1 : Fondements des langages de données semi-structuées du Web
 - Une suite de langages pour le Web : XML, DTD, XPath, XQuery.
 - Exportation et publication de données relationnelles en XML.
 - Stockage et interrogation de données XML dans les bases de données relationnelles.
- Partie 2 : Présentation des données du Web
 - médiatisation de données XML via des transformations XSLT
 - présentation du framework applicatif Angular au sein d'une architecture MEAN

- Responsable : Sèverine Bérard (MCF Info/ISEM)
- Programme : principalement centré autour de la recherche de motifs
 - Recherche de motifs exacts : KMP et Boyer-Moore
 - Recherche de motifs approchés et multiples : Aho-Corasick
 - Structures d'indexation des textes : arbre des suffixes, table des suffixes, dictionnaires
- MCC : contrôle continu 35 % et examen final 65 % (règle du max)
- Plus d'info : Severine.Berard@umontpellier.fr

- Courriels enseignants vers étudiants : via votre adresse institutionnelle
- Courriels étudiants vers enseignants : via une adresse où apparaît votre nom
- Beaucoup d'informations sur les pages web des enseignants
- La plupart des enseignants ont un bureau au LIRMM (campus Saint-Priest) : <http://www.lirmm.fr/xml/fr/0003-27.html>
- Pour faciliter la communication : délégués étudiants

- Espace numérique de travail (ENT) : site UM, ENT-intranet
- Emploi du temps ADE : ENT, onglet planning
- Site du département informatique :
<http://informatique.edu.umontpellier.fr/>

S'informer sur la spécialité Aigle

- Contact

- Clémentine Nebut (nebut@lirmm.fr)
- Pierre Pompidor (pompidor@lirmm.fr)
- Secrétariat (fds-info-secrmaster@umontpellier.fr)