



Guide pédagogique

Module « Ingénierie Logicielle »

Options IASD et IL – 8.4 (4 crédits ECTS)

Place du module et enjeux

Quel que soit le profil visé par les étudiants du département IIA (IASD ou IL), il est nécessaire de connaître les différentes étapes du cycle de vie des projets logiciels et les langages de modélisation et outils associés, de maîtriser les techniques de gestion de projet informatique ainsi que de connaître les bonnes pratiques de la conception de logiciels. Ce module est dédié à ces thématiques et détaillera à la fois les fondements théoriques de la spécification de logiciels, les méthodes de conception et les appliquera sur un cas d'étude.

Teaching guide and syllabus

« Software Engineering » module

DSAI and SE options – 8.4 (4 ECTS credits)

Subject matter importance and associated issues

For both DSAI and SE options, we require students to know Software life cycle stages, modelling languages and associated tools, project management techniques as well as software design good practices. This module concerns these aspects and will detail theoretical backgrounds of software specification and design methods. This will be applied on a case study.

Responsable : Thomas Lambolais

Téléphone : 04 34 24 62 60

Courriel : thomas.lambolais@mines-ales.fr



IMT Mines Alès
École Mines-Télécom

| ENSEIGNEMENTS ACADÉMIQUES | Volume horaire | Détail des coefficients | Crédits |
|------------------------------|----------------|-------------------------|---------|
| Ingénierie Logicielle | 50 h | | |
| ○ Conception des logiciels | 10 | 1 | 4 |
| ○ Spécification formelle | 12 | 1 | |
| ○ Cas d'étude | 28 | 2 | |

Matière 1

| | |
|---|---|
| Titre de la matière : Conception des logiciels | |
| Code : 2IA-8.4.1 | Titre du module : « Ingénierie Logicielle » |
| Semestre : S8 | Cursus de rattachement : Département 2IA, options IASD et IL |

| Heures présentiel | Heures total | Cours | TD | TP | Projet | Contrôles | Travail personnel | Coef /module | ECTS |
|-------------------|--------------|-------|----|----|--------|-----------|-------------------|--------------|------|
| 10 | 15 | 4 | | 5 | | 1 | 5 | 1 | 1 |

| | |
|---------------|---|
| Résumé | Ce cours présente une démarche de développement logiciel, ainsi que le langage de modélisation UML. |
|---------------|---|

| | |
|---------------------------|--|
| Responsable | Sylvain Vauttier – LGI2P/IMT Mines Alès |
| Équipe enseignante | Sylvain Vauttier – LGI2P/IMT Mines Alès Doctorant en appui pour l'encadrement de TP |

| | |
|------------------|---|
| Mots-clés | Analyse, Conception, Modélisation, UML, Logiciel. |
| Prérequis | La notion de classe et d'objets en programmation doit être assimilée. |

Contexte et objectif général :
 Les enseignements des options IASD (Intelligence Artificielle et Science des Données) et IL (Ingénierie Logicielle) nécessitent une bonne maîtrise de la modélisation, de l'analyse et de la conception de logiciels. Les langages de modélisation standardisés comme UML sont incontournables.

Programme et contenu :
 Les notions d'analyse et de conception d'un système informatique sont introduites, à travers le langage UML. Les diagrammes UML présentés sont :
 - Les diagrammes de classe ;
 - Les diagrammes d'activité ;
 - Les diagrammes de cas d'utilisation ;
 - Les diagrammes de package ;
 - Les diagrammes de séquence ;
 - Les diagrammes d'états.
 Plusieurs exemples sont donnés et la méthode associée est présentée (modèle de domaine, modèle des usages, architecture globale, architecture détaillée). Plusieurs séances d'exercices sont proposées.

Méthode et organisation pédagogique :
 Les enseignements sont prévus pour 60 élèves. Les TP seront réalisés sur les ordinateurs personnels de ces derniers.
 Le découpage est prévu comme suit :
 - 4h de cours
 - 5h de TD/TP
 - 1h de contrôle
 - environ 5h de travaux personnel.

Acquis d'apprentissage visés :
 Savoir connaître et utiliser les concepts des langages de modélisation orientés-objets : classe, objet, relation de spécialisation, relation d'association, notions d'activité, d'état, d'événement, d'acteur, de cas d'utilisation.

Évaluation :
 Contrôle d'une heure sur table. Documents autorisés. Sans machine.

Retour sur l'évaluation fait à l'élève :
 Mise à disposition des copies et restitution 3 semaines après l'examen.

Support pédagogique et références :
 Copie des slides, énoncés de TP, corrigés de TP.

Matière 2

| | |
|---|---|
| Titre de la matière : Spécification formelle | |
| Code : 2IA-8.4.2 | Titre du module : « Ingénierie Logicielle » |
| Semestre : S8 | Cursus de rattachement : Département 2IA, options IASD et IL |

| Heures présentiel | Heures Total | Cours | TD | TP | Projet | Contrôles | Travail personnel | Coef /module | ECTS |
|-------------------|--------------|-------|----|----|--------|-----------|-------------------|--------------|------|
| 12 | 18 | 4 | | 6 | | 2 | 6 | 1 | 1 |

| | |
|---------------|---|
| Résumé | Introduction au langage de spécification formelle Alloy |
|---------------|---|

| | |
|---------------------------|---|
| Responsable | Thomas Lambolais – LGI2P/IMT Mines Alès |
| Équipe enseignante | Thomas Lambolais – LGI2P/IMT Mines Alès |

| | |
|------------------|--|
| Mots-clés | Spécification formelle, notation logique et ensembliste, vérification. |
| Prérequis | Connaissances de base en logique classique de prédicats du premier ordre (notions de proposition, prédicat, assertion, ...). Connaissances des notions de classes et objet (diagrammes de classes UML) |

| | |
|--|--|
| Contexte et objectif général : | |
| Les enseignements des options IASD (Intelligence Artificielle et Science des Données) et IL (Ingénierie Logicielle) nécessitent une bonne maîtrise de la modélisation, de l'analyse et de la conception de logiciels. Les langages de modélisation standardisés comme UML sont incontournables. Les bases théoriques de la modélisation formelle sont nécessaires. | |
| Programme et contenu : | |
| Ce cours donne une introduction au langage de spécification formelle Alloy défini au MIT. Le premier objectif est de comprendre l'intérêt et le positionnement d'un langage de spécification, qui vise à énoncer un problème de façon rigoureuse, mais sans chercher à le résoudre. Le second objectif est de comprendre l'utilité d'un langage de spécification formelle. Alloy est un langage de spécification formelle, de type déclaratif. Alloy est défini sur des bases théoriques solides (théorie des ensembles, logique du premier ordre) et utilise un moteur de vérification automatique permettant d'étudier la cohérence (satisfaisabilité) des modèles et la validité des assertions. De nombreux exemples et exercices sont étudiés : paradoxe du Barbier, affirmation de Peterson, Système de gestion de fichiers. Sur chacun de ces exemples, plusieurs propriétés sont formellement étudiées. | |
| Méthode et organisation pédagogique : | |
| Les enseignements sont prévus pour 60 élèves. Les TP seront réalisés sur les ordinateurs personnels de ces derniers. Le découpage est prévu comme suit : <ul style="list-style-type: none"> - 4h de cours - 6h de TD/TP - 2h de contrôle - 6h de travail personnel | |
| Acquis d'apprentissage visés : | |
| Savoir ce qu'est une spécification. Savoir ce qu'est un langage formel. Modéliser formellement les aspects statiques d'un problème simple. Savoir distinguer assertions, faits et prédicats et leurs représentations en Alloy. Savoir utiliser les outils associés. Connaître les limites du langage Alloy. | |
| Évaluation : | |
| Contrôle écrit, tous documents autorisés sauf électroniques. 2h. | |
| Retour sur l'évaluation fait à l'élève : | |
| Mise à disposition des corrections, consultation des copies 3 semaines après la dernière séance. | |
| Support pédagogique et références : | |
| Supports de cours. Énoncés et corrigés des exercices. Outil et base de jeu d'exercices. | |

Matière 3

| | |
|--|---|
| Titre de la matière : Cas d'étude | |
| Code : 2IA-8.4.3 | Titre du module : « Ingénierie Logicielle » |
| Semestre : S8 | Cursus de rattachement : Département 2IA, options IASD et IL |

| Heures présentiel | Heures total | Cours | TD | TP | Projet | Contrôles | Travail personnel | Coef /module | ECTS |
|-------------------|--------------|-------|----|----|--------|-----------|-------------------|--------------|------|
| 28 | 30 | | | | 24 | 4 | 10 | 2 | 2 |

| | |
|---------------|--|
| Résumé | Développement d'un projet de génie logiciel de type « système d'information ». |
|---------------|--|

| | |
|---------------------------|---|
| Responsable | Sylvain Vauttier – LGI2P/IMT Mines Alès |
| Équipe enseignante | Sylvain Vauttier – LGI2P/IMT Mines Alès |

| | |
|------------------|---|
| Mots-clés | Projet de génie logiciel, système d'information, méthode Agile. |
| Prérequis | Module d'approfondissement |

| |
|---|
| <p>Contexte et objectif général : Les enseignements des options IASD (Intelligence Artificielle et Science des Données) et IL (Ingénierie Logicielle) forment des ingénieurs devant maîtriser la gestion des projets de développement logiciels. Cette mise en pratique permet de s'assurer que ces notions sont acquises.</p> |
| <p>Programme et contenu : Traiter un projet informatique dans sa globalité : suivre un cycle de développement depuis la conception jusqu'à la réalisation ; mise en œuvre de méthodes et d'outils de génie logiciel Démarche : ingénierie du besoin : modèle de domaine, modèle des usages ; conception : architectures (diagrammes de package, diagrammes de classes), comportement (diagrammes de séquence, diagrammes états-transitions), modèle conceptuel de données ; implantation : applications et bases de données ; test et qualité.</p> |
| <p>Méthode et organisation pédagogique : Les enseignements sont prévus pour 60 élèves répartis en deux groupes de 23 élèves. Les TP seront réalisés sur les ordinateurs personnels de ces derniers. Le découpage est prévu comme suit : - 5 séances de TP - travail en équipes de 3.</p> |
| <p>Acquis d'apprentissage visés : Savoir appréhender un projet informatique : l'analyse, le spécifier, le concevoir, le développer, le tester, l'ensemble de façon Agile.</p> |
| <p>Évaluation : Types d'épreuves et répartition des coef : projet (coef. 2)</p> |
| <p>Retour sur l'évaluation fait à l'élève : Retours au moment de la soutenance. Notes diffusées 3 semaines après la soutenance.</p> |
| <p>Support pédagogique et références : 1 Polycopié</p> |

Méthode et organisation pédagogique

Il s'agit d'un enseignement relativement classique avec une partie réalisée en cours interactifs et une partie appliquée au travers de TD/TP et projets. De nombreux outils sont utilisés.

Modalité d'évaluation

Le niveau d'acquisition des compétences sera évalué selon les exigences suivantes :

| N° indicateur | Indicateur |
|---------------|---|
| 1 | connaître les savoirs formels et pratiques du socle des fondamentaux |
| 2 | Exploiter les savoirs théoriques et pratiques |
| 3 | Analyser, interpréter, modéliser, émettre des hypothèses, et résoudre |

Répartition

| Matière | Contrôle | Coefficients | Type de notation | Indicateurs évalués | Chapitres |
|--------------------------|---|--------------|------------------|---------------------|-----------|
| Conception des logiciels | Interrogation écrite | 1 | Individuelle | 1 | Tous |
| Conception des logiciels | Sur projet | 2 | En binôme | 3 | Tous |
| Spécification formelle | Interrogation écrite Évaluation projet | 1 | Individuelle | 1, 2, 3 | Tous |
| Cas d'étude | Évaluation projet | 1 | En binôme | 3 | Tous |

Engagement de l'étudiant, éthique et professionnalisme

La démarche éthique est définie dans le règlement intérieur de l'établissement. Chaque étudiant s'engage à en prendre connaissance et à la respecter.

Obligation des cours : Présence obligatoire pour tous à chaque séance

Nombre d'heures estimées de travail personnel : pour acquérir les compétences demandées, il est nécessaire que l'étudiant consacre minimum 45 min de travail personnel de compréhension et d'approfondissement par séance de cours.

25h de travail personnel sont estimées, principalement dédiées à la définition et réalisation du projet.

Nombre d'heures estimées de préparation aux travaux dirigés (TD) : 1 à 2h

Pénalité pour retard : Tout travail remis en retard sans motif valable peut être pénalisé de 1 point par jour de retard.

Équipe enseignante

| Nom | Domaine d'expertise | Courriel/Téléphone |
|-------------------|--|--|
| Christelle URTADO | Génie Logiciel et Programmation Orientée Objet | christelle.urtado@mines-ales.fr 04 34 24 62 89 |
| Thomas LAMBOLAI | Génie logiciel, spécification formelle et vérification | thomas.lambolais@mines-ales.fr 04 34 24 62 60 |
| Sylvain VAUTIER | Génie Logiciel et Programmation Orientée Objet | sylvain.vautier@mines-ales.fr 04 34 24 62 85 |

| ACADEMIC TEACHING | Teaching hours | Coefficients | Credits |
|-----------------------------|----------------|--------------|---------|
| Software engineering | 50 h | | |
| ○ Software design | 10 | 1 | 4 |
| ○ Formal specification | 12 | 1 | |
| ○ Case study | 20 | 2 | |

Course 1

| | |
|-------------------------------|---|
| Title: Software design | |
| Code : 2IA-8.4.1 | Title of the module : « Software engineering » |
| Semester : S8 | Associated Cursus : CSAI Department, DSAI and SE options |

| Hours of presence | Total hours | Lectures | Seminar | Labs | Project | Testing | Personal work | Coef /module | ECTS |
|-------------------|-------------|----------|---------|------|---------|---------|---------------|--------------|------|
| 10 | 15 | 4 | | 5 | | 1 | 5 | 1 | 1 |

| | |
|----------------|--|
| Summary | This course presents a generic software design process based on UML (Unified Modeling Language). |
|----------------|--|

| | |
|----------------------|---|
| Head | Sylvain Vauttier – LGI2P ¹ /IMT Mines Alès |
| Teaching Team | Sylvain Vauttier – LGI2P/IMT Mines Alès |

| | |
|----------------------|--|
| Keywords | Software analysis, Software design, Software modeling, UML |
| Prerequisites | Object-oriented programming |

| |
|--|
| <p>Context and general objective: Courses in the CSAI department require good software engineering skills, including software analysis and design. As an industrial standard, UML modelling is an essential design ability.</p> <p>Programme and contents: A generic software analysis and design process is presented, based on the UML modelling language. Different kinds of UML diagrams are used as support tools:</p> <ul style="list-style-type: none"> - class diagrams, - activity diagrams, - use case diagrams, - package diagrams, - sequence diagrams, - statechart diagrams. <p>The outputs of the process (domain model, usage model, global and detailed architecture) are exemplified by a study case. Different labs, using a CASE tool (Visual Paradigm), are proposed to experiment each step of the process.</p> <p>Method and pedagogic organisation: Activities are organised for a full group of 46 students max. Labs are realised on the personal computer of the students (BYOD). Activities are organized as follows:</p> <ul style="list-style-type: none"> - courses (4h) - labs (5h) - exam (1h) - personal work (5h) <p>Targeted skills or knowledge: To master the analysis and design of a software using essential UML diagrams</p> <p>Evaluation: 1 hour written exam (no computer; no tool). Documents allowed.</p> <p>Feedback made to the student: Consulting copies of exams Deadlines for giving correction of exams: (a maximum of 3 weeks is allowed for a correction of exams)</p> <p>Teaching material and references : Textbook. – Lab topic and corrections</p> |
|--|

¹ Laboratoire de Génie Informatique et d'Ingénierie de Production (Computer Science and Systems Engineering lab).

Course 2

| | |
|-------------------------------------|--|
| Title : Formal specification | |
| Code : 2IA-8.4.2 | Title of the module : « Software engineering » |
| Semester : S8 | Associated Coursus : CSAI Department, DSAI and SE options |

| Hours of presence | Total hours | Lectures | Seminar | Labs | Project | Testing | Personal work | Coef /module | ECTS |
|-------------------|-------------|----------|---------|------|---------|---------|---------------|--------------|------|
| 12 | 18 | 4 | | 6 | | 2 | 6 | 1 | 1 |

| | |
|----------------|--|
| Summary | Introduction to the Alloy formal specification language. |
|----------------|--|

| | |
|----------------------|---|
| Head | Thomas Lambolais – LGI2P/IMT Mines Alès |
| Teaching Team | Thomas Lambolais – LGI2P/IMT Mines Alès |

| | |
|----------------------|---|
| Keywords | Formal specification, logic notations, overall rating, verification. |
| Prerequisites | Basic knowledge of first-order predicates logic (notions of proposal, predicate, assertion, etc.). Class and object concepts (UML class diagrams) |

Context and general objective:
 The lessons of the DSAI (Data Science and Artificial Intelligence) and SE (Software Engineering) options require a good command of software modelling, analysis and design. Standardized modelling languages such as UML are essential. The theoretical foundations of formal modelling are necessary.

Programme and contents:
 This course provides an introduction to the formal specification language Alloy defined at MIT. The first objective is to understand the interest and positioning of a specification language, which aims to state a problem in a rigorous way, but without seeking to solve it. The second objective is to understand the usefulness of a formal specification language.
 Alloy is a formal specification language, of a declarative type.
 Alloy is defined on solid theoretical bases (set theory, first-order logic) and uses an automatic verification engine to study the consistency (satisfiability) of models and the validity of assertions.
 Many examples and exercises are studied: Barber Paradox, Peterson's statement, File Management System. On each of these examples, several properties are formally studied.

Method and pedagogic organisation:
 The lessons are planned for 60 students. The labs will be performed on the latter's personal computers.
 The course will be organized as follows:
 - 4 hours of lectures
 - 6 hours of seminars and labs
 - 2 hours of written exam
 - 6 hours of personal work

Targeted skills or knowledge:
 Know what a specification and a formal language are.
 Formally model the static aspects of a simple problem. To be able to distinguish assertions, facts and predicates and their representations in Alloy. Know how to use the associated tools. Know the limits of the Alloy language.

Evaluation:
 Written exam, with documents allowed except numerical ones.

Feedback made to the student :
 Exam copies within three weeks after the exam.

Teaching material and references :
 Lecture slides, statements and exercises corrections, Tool and exercise set base

Course 3

| | |
|---------------------------|---|
| Title : Case study | |
| Code : 2IA-8.4.3 | Title of the module : « Software engineering » |
| Semester : S8 | Associated Cursus : CSAI Department, DSAI and SE options |

| Hours of presence | Total hours | Lectures | Seminar | Labs | Project | Testing | Personal work | Coef /module | ECTS |
|-------------------|-------------|----------|---------|------|---------|---------|---------------|--------------|------|
| 28 | 30 | | | | 24 | 4 | 10 | 2 | 2 |

| | |
|----------------|--|
| Summary | Small-sized information system development project |
|----------------|--|

| | |
|----------------------|--|
| Head | Sylvain Vauttier – LGI2P/IMT Mines Alès |
| Teaching Team | Sylvain Vauttier – LGI2P/IMT Mines Alès Doctorant en appui pour l'encadrement de TP |

| | |
|----------------------|---|
| Keywords | Software engineering project, Information system, Agile development process |
| Prerequisites | Module 7.3 |

| |
|--|
| <p>Context and general objective: DSAI (Data Science and Artificial Intelligence) and SE (Software Engineering) options train engineer that should master software engineering development project management.</p> |
| <p>Programme and contents:</p> <p>Realizing a complete software engineering process: from software analysis to programming; application of software engineering best practices and tools.</p> <p>Approach:</p> <ul style="list-style-type: none"> - requirement engineering (domain model, usage model), - design (architecture, behaviour, data model) - programming (applications and databases) - test, quality assessment, versioning |
| <p>Method and pedagogic organisation: Activities are organised for a full group of 60 students max. Labs are realised on the personal computer of the students (BYOD). Activities are organized as 5 labs (3h each), for teams of 3 students.</p> |
| <p>Targeted skills or knowledge : Managing software engineering projects: requirement analysis, software design, software implementation, agile development process.</p> |
| <p>Evaluation : project (2)</p> |
| <p>Feedback made to the student : Feedbacks just after the presentation of the project.</p> |
| <p>Teaching material and references : 1 Photocopied material.</p> |

Method and teaching organisation

Teachings will consist of lectures, seminars and labs.

Testing procedures

The student's level of knowledge acquisition will be evaluated according to the following points:

| N° Indicator | Indicator |
|--------------|---|
| 1 | To know the formal and practical knowledge constituting the foundation of a given field |
| 2 | Exploit theoretical and practical knowledge |
| 3 | Analyse, interpret, model, hypothesize and solve problems |

Grading scheme:

| Class | Exam | Coefficients | Administration mode | Evaluated Indicators | Chapters |
|----------------------|-----------------|--------------|---------------------|----------------------|----------|
| Software design | MCQ | 1 | Individual | 1 | all |
| Software design | Project | 2 | Duo | 3 | all |
| Formal specification | Exam Project | 1 | Individual Duo | 1, 2, 3 | all |
| Case study | Project | 1 | Duo | 3 | all |

Student commitments, ethics and professionalism

Expectations concerning ethics are defined in the establishment's code of conduct. Each student is expected to know and respect the code of conduct.

Obligatory presence in classes: Students must attend all courses, seminars and labs

Estimated hours of personal study: in order to acquire the required learning level, the student is expected (must) to spend a minimum of 45 min of personal study time per hour spent in class. 25h of personal work are estimated for the project.

Estimated hours of preparation required for labs/Work Shop: 1 to 2h

Late penalties: Late works are subject to penalties as follows: 1 points per day (ratings are between 0 and 20)

Teaching team

| Name | Field of expertise | Email/phone |
|-------------------|---|--|
| Christelle URTADO | Software Engineering and Object-oriented programming | christelle.urtado@mines-ales.fr 04 34 24 62 89 |
| Thomas LAMBOLAIS | Software Engineering, Formal specification and validation | thomas.lambolais@mines-ales.fr 04 34 24 62 60 |
| Sylvain VAUTTIER | Software Engineering and Object-oriented programming | sylvain.vauttier@mines-ales.fr 04 34 24 62 85 |

Approbation

Ce guide pédagogique entre en vigueur à compter du 7 janvier 2019

Il est porté à la connaissance des élèves par une publication sur le site de l'école

| Rédaction | Vérification | Validation |
|--|---|--|
| L'enseignant responsable du module : Thomas Lambolais | Le responsable d'UE / de département : Sylvie Ranwez | Le directeur de l'école, Pour le directeur et par délégation, Le directeur de la DFA / de la DE : Michel Ferlut |