

Pourquoi cette UE ?

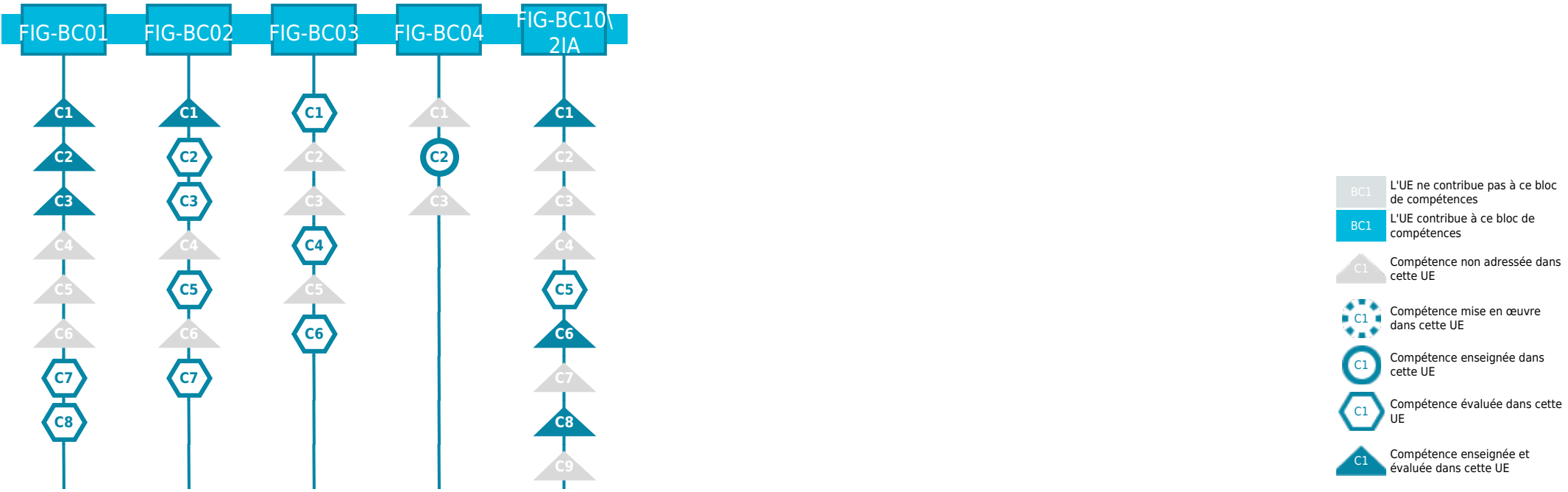
Quel que soit le profil visé par les étudiants du département IIA (IASD ou IAIL), il est nécessaire de connaître les différentes étapes du cycle de vie des projets logiciels, incluant l'ingénierie des exigences, l'utilisation de langages de modélisation et outils associés, de maîtriser les techniques 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.

Eléments constitutifs de l'UE

		coefficient
2IA_8_4-1 Ingénierie des exigences		0
2IA_8_4-2 Conception des logiciels		1
2IA_8_4-3 Spécification formelle		1
2IA_8_4-4 Cas d'étude		2
Volume d'heures d'enseignement encadré	Volume d'heures de travail personnel	Nombre d'ECTS
53	39	4

Alignement curriculaire

Parmi les compétences visées par la formation, lesquelles sont développées dans cette UE ?



Contexte et enjeux de l'enseignement

Ce cours présente la méthode KAOS d'ingénierie des exigences définie par Axel van Lamsweerde, guidée par les objectifs, afin d'établir des cahiers des charges. Trois notions principales sont présentées : définition des besoins (finalité) d'un système, définition des exigences fonctionnelles et non fonctionnelles, et définition des propriétés du domaine et attentes du système sur son environnement (hypothèses simplificatrices du système). L'outil support est Objectiver de la société Respect-it.

Prise en compte des dimensions socio-environnementales

ODD7 - Énergie propre et d'un coût abordable ODD8 - Travail décent et croissance économique ODD9 - Industrie, innovation et infrastructure

Prérequis

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	6
Cours intégré (cours + TD)	
TD	0
TP	0
Projets	0
Travail en autonomie encadré	0
Contrôles et soutenances	0
Travail personnel	4

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

Comprendre les distinctions entre besoins, exigences et hypothèses, ainsi que la différence entre la gestion des exigences et l'élicitation des exigences. Voir l'importance de la définition du contexte : connaissances du domaine, attendus du système sur son environnement, et non uniquement attendus des utilisateurs sur le système.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

Exemples et cours.
Exercices sur des systèmes nouveaux pour lesquels il est nécessaire d'identifier des exigences.
Application dans le cas d'étude principal.

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Evaluation au sein de l'étude de cas.

2IA_8_4 Ingénierie logicielle	FIG
2IA_8_4-1 Ingénierie des exigences	S8

Plan de cours

- 1. Introduction générale : constat, définitions, objectifs.
- 2. Notions d'exigences, besoins et hypothèses.
- 3. Exercice.
- 4. Exemple et déroulement de la méthode.

Ressources et références

Supports de présentation. Référence à un ouvrage fondateur.

2IA_8_4 Ingénierie logicielle	FIG
2IA_8_4-2 Conception des logiciels	S8

Contexte et enjeux de l'enseignement

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. Le cours présente les concepts et les diagrammes associés à une démarche de conception relevant du Domain-Driven Engineering et de l'Agile UML. Ils sont pratiqués en TD à l'aide de l'outil Visual Paradigm (AGL UML). Cette démarche et ces outils sont utilisés lors du cas d'étude proposé dans cette UE.

Prise en compte des dimensions socio-environnementales

ODD9 - Industrie, innovation et infrastructure

Prérequis

Les concepts de la programmation orientée-objets doivent être assimilés. Ceux de la conception de schéma de bases de données également.

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	9
Cours intégré (cours + TD)	
TD	0
TP	0
Projets	0
Travail en autonomie encadré	0
Contrôles et soutenances	1
Travail personnel	5

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

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.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

Les enseignements sont prévus pour 60 élèves. Les TP seront réalisés sur les ordinateurs personnels de ces derniers. Le découpage est prévu comme suit :

- 9h de cours/TD
- 1h de contrôle
- environ 5h de travaux personnel.

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Contrôle écrit, tous documents autorisés sauf électroniques. 2h.

Mise à disposition des copies et restitution 3 semaines après l'examen.

Plan de cours

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 classes
- 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).

Ressources et références

Copie des slides de cours. Enoncés de TDs. Corrigés de TDs.

2IA_8_4 Ingénierie logicielle	FIG
2IA_8_4-3 Spécification formelle	S8

Contexte et enjeux de l'enseignement

Les enseignements des options IASD (Intelligence Artificielle et Science des Données) et IAIL (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.

Prise en compte des dimensions socio-environnementales

ODD8 - Travail décent et croissance économique

Prérequis

Connaissances de base en logique classique de prédicats du premier ordre (notions de proposition, prédicat, assertion, ...). Connaissances des notions de classes et objet.

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	4
Cours intégré (cours + TD)	
TD	
TP	6
Projets	
Travail en autonomie encadré	
Contrôles et soutenances	2
Travail personnel	6

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

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.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

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
- 6h de TD/TP
- 2h de contrôle
- 6h de travail personnel

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Contrôle écrit, tous documents autorisés sauf électroniques. 2h.

Mise à disposition des corrections, consultation des copies 3 semaines après la dernière séance.

2IA_8_4 Ingénierie logicielle	FIG
2IA_8_4-3 Spécification formelle	S8

Plan de cours

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 (satisfiabilité) 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.

Ressources et références

Supports de cours. Enoncés et corrigés des exercices. Outil et base de jeu d'exercices.

Contexte et enjeux de l'enseignement

Les enseignements des options IASD (Intelligence Artificielle et Science des Données) et IAIL (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.

Prise en compte des dimensions socio-environnementales

ODD9 - Industrie, innovation et infrastructure

Prérequis

Programmation orientée-objets. Conception de bases de données. Langage UML.

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	
Cours intégré (cours + TD)	
TD	
TP	
Projets	24
Travail en autonomie encadré	
Contrôles et soutenances	1
Travail personnel	24

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

Savoir appréhender un projet informatique : l'analyser, le spécifier, le concevoir, le développer, le tester, l'ensemble de façon Agile

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

Les enseignements sont prévus pour 60 élèves répartis en trinômes . Les projets sont réalisés sur les ordinateurs personnels des étudiants (BYOD). ces derniers.
Le découpage est prévu comme suit :
- 5 séances de TP (encadrement et suivi de projet),
- une séance de recette (présentation et remise des livrables).

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Types d'épreuves et répartition des coef : projet (coef. 2)

Retours au moment de la soutenance. Notes diffusées 3 semaines après la soutenance

2IA_8_4 Ingénierie logicielle	FIG
2IA_8_4-4 Cas d'étude	S8

Plan de cours

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 étatstransitions), modèle conceptuel de données ; implantation : applications et bases de données ; test et qualité.

Nous essayons de proposer des sujets basés sur des cas d'études réels, avec participation des parties prenantes lors des phases d'ingénierie du besoin et de recette.

Ressources et références

Présentation du projet. Ressources de présentation du contexte, du domaine métier. Données de terrain.