

5.0 crédits

30.0 h + 30.0 h

1q

Enseignants:	Van Lamsweerde Axel ;
Langue d'enseignement:	Anglais
Lieu du cours	Louvain-la-Neuve
Ressources en ligne:	> http://www.info.ucl.ac.be/Enseignement/Cours/INGI2255/
Préalables :	-- logique mathématique telle qu'enseignée dans le cours INGI1101 -- maîtrise de la programmation orienté-objet, de l'algorithmique et des structures de données telle que visée par le cours SIN1121 -- avoir participé à la réalisation d'un projet logiciel de taille réduite (par exemple SIN1124).
Thèmes abordés :	-- Cycle de vie d'un projet logiciel: aspects statiques (les produits) et dynamiques (les procédés). -- Introduction à l'analyse des besoins: modélisation semi-formelle, spécification et analyse des exigences auxquelles doit répondre le logiciel. -- Introduction à la conception d'architectures logicielles: hiérarchisation, modularisation; styles et patterns architecturaux. -- Spécification d'un module vu comme unité de travail. -- Validation d'un logiciel: conception de jeux de test: black-box, white-box et d'intégration. -- Documentation des décisions prises au cours des différentes étapes du cycle de vie.
Acquis d'apprentissage	Les étudiants ayant suivi avec fruit ce cours seront capables de -- décrire les principaux problèmes rencontrés dans le contexte de gros projets logiciels -- discuter des différentes qualités qu'une application logicielle complexe et de grande taille doivent présenter -- modéliser les systèmes complexe faisant appel à d'important logiciels (génie logiciel guidé par modèle) -- maîtriser des approches systématiques de développement logiciel -- décrire les différents types de produits et les processus impliqués tout au long du cycle de vie du génie logiciel -- appliquer des méthodes classiques d'ingénierie des exigences logicielles et de modélisation, de spécification, de conception, de validation et de documentation de logiciels de haute qualité -- évaluer l'impact des décisions prises aux différentes étapes de ce cycle de vie. Les étudiants auront développé des compétences méthodologiques et opérationnelles. En particulier, ils auront développé leur capacité à -- modéliser de manière rigoureuse le produit final afin de pouvoir en faire une analyse rigoureuse dès la conception qui assurera la qualité du produit et du processus de développement, -- expliquer comment prendre en compte dès la conception d'un produit du volume de ressources (en particulier humaines) qui seront impliquées dans son développement. <i>La contribution de cette UE au développement et à la maîtrise des compétences et acquis du (des) programme(s) est accessible à la fin de cette fiche, dans la partie « Programmes/formations proposant cette unité d'enseignement (UE) ».</i>
Modes d'évaluation des acquis des étudiants :	Quizz individuel basé sur le mini-projet, après chaque étape -- pour vous aider à vous préparer avant de chef de projet de réunion -- pour l'auto-évaluation La participation individuelle lors des réunions de groupe hebdomadaires avec chef de projet -- connaissance des techniques nécessaires, le degré d'interaction, initiatives, idées, ... (20% ingi2255/sinf2255) La qualité des livrables: les rapports de projet et la documentation (40% ingi2255/sinf2255) Examens écrits (Janvier) -- INGI 2251: théorie (100% INGI2251) -- INGI2255: mini-projet individuel (40% ingi2255/sinf2255)
Méthodes d'enseignement :	Le cours est fortement couplé au développement d'un projet " en vraie grandeur ", par équipes, selon les techniques étudiées (voir INGI2255 : Projet de génie logiciel). L'enseignement est organisé de manière intensive, en début de quadrimestre, pour permettre un démarrage rapide du projet, puis de façon plus épisodique, en fonction des besoins des différentes étapes de développement. Certaines séances comportent des tests de connaissance (" quizz ") et des exposés d'études de cas illustrant les techniques à appliquer dans le projet. Les travaux pratiques (2 ECTS) sont totalement intégrés à INGI2255 (5 ECTS) ; cette introduction au génie logiciel correspond donc à une charge globale de 10 ECTS, soit un travail tiers-temps durant tout le quadrimestre.
Contenu :	-- Génie logiciel: quoi, pourquoi, comment -- Le cycle de vie d'un logiciel -- Cycle de vie statique

	<ul style="list-style-type: none"> -- Cycle de vie dynamique -- Estimation des coûts et échéancier -- Spécification dans le cycle de vie logiciel -- Ingénierie des exigences -- Elaboration de documents décrivant les exigences basées sur l'utilisation de modèles -- Spécification des exigences logicielles -- Conception de l'architecture d'un logiciel -- Structuration hiérarchique d'un logiciel -- Modularisation -- Styles architecturaux -- Test d'un logiciel -- boîte noire, boîte blanche, tests d'intégration
<p>Bibliographie :</p>	<ul style="list-style-type: none"> -- transparents disponibles en ligne -- A. van Lamsweerde, Requirements Engineering, Wiley, 2009 -- F. Brooks, The Mythical Man-Month, Addison-Wesley, 1995 -- S. Pfleeger, Software Engineering: Theory and Practice, 2nd Edition, Prentice-Hall, 2001
<p>Cycle et année d'étude: :</p>	<ul style="list-style-type: none"> > Master [120] : ingénieur civil en informatique > Master [120] en sciences informatiques > Master [60] en sciences informatiques > Master [120] : ingénieur civil biomédical
<p>Faculté ou entité en charge:</p>	<p>INFO</p>