


En raison de la crise du COVID-19, les informations ci-dessous sont susceptibles d'être modifiées, notamment celles qui concernent le mode d'enseignement (en présentiel, en distanciel ou sous un format comodal ou hybride).

5 crédits	30.0 h + 15.0 h	Q1
-----------	-----------------	----

Enseignants	Mens Kim ;
Langue d'enseignement	Anglais
Lieu du cours	Louvain-la-Neuve
Thèmes abordés	<ul style="list-style-type: none"> • Les "bonnes pratiques" de la programmation orienté-objet; • Les techniques de réutilisation, application frameworks; • Les métriques et mesures de logiciels; • La compréhension et rétro-ingénierie de logiciels; • La ré-ingénierie et la restructuration et réusinage de logiciels; • L'utilisation de certains outils qui supportent les taches mentionnées ci-dessus.
Acquis d'apprentissage	<p>Eu égard au référentiel AA du programme « Master ingénieur civil en informatique », ce cours contribue au développement, à l'acquisition et à l'évaluation des acquis d'apprentissage suivants :</p> <ul style="list-style-type: none"> • INFO1.1 , INFO1.3 • INFO2.5 • INFO5.5 <p>Eu égard au référentiel AA du programme « Master [120] en sciences informatiques », ce cours contribue au développement, à l'acquisition et à l'évaluation des acquis d'apprentissage suivants :</p> <ul style="list-style-type: none"> • SINF1.M3 • SINF2.5 • SINF5.5 <p>1</p> <p>Les étudiants ayant suivi avec fruit ce cours seront capables de</p> <ul style="list-style-type: none"> • Comprendre les difficultés de développement de code dans un contexte de changement, par opposition au développement à partir de rien • Évaluer l'impact d'une demande de modification d'un produit existant de taille moyenne. • Décrire les techniques, les idiomatiques de codage et d'autres mécanismes de mise en 'uvre des conceptions qui sont plus maintenables. • Comprendre comment les modèles de conception peuvent améliorer la conception d'un système de logiciel. • Refactoriser une implémentation logicielle existante pour améliorer certains aspects de sa conception. • Identifier les principaux problèmes liés à l'évolution d'un logiciel et expliquer leur impact sur le cycle de vie du logiciel. • Discuter des avantages et inconvénients des différents types de réutilisation de logiciels. <p>-----</p> <p><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></p>
Modes d'évaluation des acquis des étudiants	<p>En raison de la crise du COVID-19, les informations de cette rubrique sont particulièrement susceptibles d'être modifiées.</p> <p>ÉVALUATION DU COURS</p> <ul style="list-style-type: none"> • [10%] Participation active lors des travaux pratiques • [40%] 2 missions intermédiaires liées aux travaux pratiques • [10%] démo d'une application en fin de semestre • [50%] pendant la session d'examen <ul style="list-style-type: none"> • [25%] examen écrit • [25%] présentation d'une mission finale <p>En cas de doute sur la note finale le titulaire du cours se réserve le droit de demander à l'étudiant de passer un examen oral complémentaire.</p>

Méthodes d'enseignement	<p>En raison de la crise du COVID-19, les informations de cette rubrique sont particulièrement susceptibles d'être modifiées.</p> <p>ORGANISATION DU COURS :</p> <p>Cours magistraux couvrant les différents sujets du cours</p> <p>Travaux pratiques pour appliquer les concepts de manière pratique</p> <ul style="list-style-type: none"> • développer et faire évoluer un système logiciel maintenable et réutilisable <p>Missions pour compléter et faire évoluer une application développée lors des travaux pratiques</p>
Contenu	<p>Le cours couvre une variété de techniques, outils et méthodologies pour aider la construction de systèmes logiciels qui sont plus faciles à comprendre, à maintenir, réutiliser et d'évoluer.</p> <ul style="list-style-type: none"> • La compréhension d'un logiciel : comment concevoir et structurer un logiciel afin de pouvoir le comprendre plus facilement; • La maintenance d'un logiciel : comment construire un logiciel qui est plus maintenable, ou comment améliorer un logiciel existant pour le rendre plus maintenable; • La réutilisation d'un logiciel et son évolution: comment construire des logiciels réutilisables et comment réutiliser les logiciels existants.
Ressources en ligne	<p>Site du cours Moodle</p> <p>Les transparents de cours ainsi que d'autres informations pertinentes et pratiques relatives au cours seront accessibles sur Moodle. La même plate-forme sera également le moyen de communication entre l'enseignant et les étudiants.</p>
Bibliographie	<p>French</p> <p>Compte tenu de la variété des sujets abordés, ce cours ne suivra pas un seul livre de référence, mais sera basé sur du matériel provenant de nombreuses sources différentes. Les slides de cours seront le matériel de référence principale pour ce cours et des pointeurs vers des lectures supplémentaires seront fournis par la plate-forme de cours en ligne.</p> <p>English</p> <p>Given the variety of topics covered, this course will not follow a single textbook but is based on material from many different sources. As such, the course slides will be the main reference material for this course and pointers to additional reading material will be provided through the online course platform.</p>
Autres infos	<p>Même si un logiciel de bonne qualité peut être plus facile à maintenir et à faire évoluer, les techniques d'assurance de la qualité des logiciels ne seront pas abordées explicitement dans ce cours car ils font l'objet d'un cours distinct sur l'assurance qualité du logiciel [LINGI2251]</p> <p>Préalables:</p> <ul style="list-style-type: none"> • Avoir une bonne connaissance et expérience avec les concepts de la programmation orientée objet, les algorithmes et les structures de données. • Avoir une expérience préalable ou simultanée avec le développement d'un logiciel de taille moyenne à grande.
Faculté ou entité en charge:	INFO

Programmes / formations proposant cette unité d'enseignement (UE)				
Intitulé du programme	Sigle	Crédits	Prérequis	Acquis d'apprentissage
Master [120] : ingénieur civil en informatique	INFO2M	5		
Master [120] en sciences informatiques	SINF2M	5		