

6.0 credits	15.0 h + 45.0 h	1q
-------------	-----------------	----

Teacher(s) :	Mens Kim ;
Language :	Anglais
Place of the course	Louvain-la-Neuve
Inline resources:	<a href="http://icampus.uclouvain.be/claroline/course/index.php?cid=SINFINFO2255">http://icampus.uclouvain.be/claroline/course/index.php?cid=SINFINFO2255</a>
Prerequisites :	-- mathematical logic as provided by the course INGI1101 -- master of object-oriented programming, algorithms and data structures as provided by the SINF1121 -- participating in the implementation of a small-size software project (for example SINF1124)
Main themes :	Depending on the topic of the project and the chosen software development methodology, which may vary from one year to another, the following themes may be addressed to some extent: ' Software development methodologies, static (products) and dynamic aspects (processes); ' Requirement analysis (goals, use cases), software architectures, architectural styles and patterns, model-driven engineering (MDE); ' Programming techniques, software development environments, refactoring; ' Software validation through unit tests, integration tests, functional and structural tests, and code reviews. ' Examples of kinds of systems to be developed are distributed systems, client/server systems, secure systems, mobile systems, adaptable systems, optimizations of existing systems or data-intensive systems. ' Project management, planning, resource estimation, reporting. ' Version management by using a version management tool.
Aims :	At the outcome of this course, the students will have acquired the necessary competences to build a large-scale software system under semi-professional working conditions. More specifically, students having completing this course with success will be able to: -- Complete, in a rigorous and systematic way, the different software life cycle phases (specification, architecture, design, implementation, validation, documentation); -- Apply a software development methodology currently practiced in industry; -- Put in practice different methods and techniques to assure the quality of the produced software; -- Estimate the time and resources needed to complete such a software development project, plan the tasks to be executed and the deliverables to be produced, and respect this planning; -- Use a project management tool to assign and follow the planned software development tasks; -- Understand the problems inherent to the development of large software systems having many different stakeholders and that consist of multiple components addressing different problems; -- Work in team and manage the coordination and communication between the different team members; -- Interact with a client to identify his requirements, to clarify imprecise specifications, and to take into account requested modifications throughout the development process; -- Build upon existing code that is poorly or badly documented. <i>The contribution of this Teaching Unit to the development and command of the skills and learning outcomes of the programme(s) can be accessed at the end of this sheet, in the section entitled "Programmes/courses offering this Teaching Unit".</i>
Evaluation methods :	-- quizzes during the semester (30%) -- intermediate reports (specifications, software architecture, formalization of the specifications and test sets) and final report, demonstration of the end product (40%) -- individual participation to the group meetings with the assistant (30%)

<p>Teaching methods :</p>	<p>The project is strongly coupled to INGI2251. It is to develop a large-scale software, in teams, according to the techniques discussed in the INGI2251 course .</p> <p>The INGI2251 course is organized intensively at the beginning of the semester, to allow a quick start of the project, then more episodically, depending on the needs of different stages of project development.</p> <p>The project is to develop a large-scale application, a typical software product from the industry, in conditions of semi-professional work</p> <p>--</p> <p>Teamwork of 6-8 developers (necessary to complete a big project), overseen by a project manager (investigator)</p> <p>--</p> <p>Traceability management between stages.</p> <p>--</p> <p>Exchange of units of work between developers (specification of a component A, design of this component by B design of test sets by C, implementation by D).</p> <p>--</p> <p>Management parallelism between developers (maximize) and interactions (minimize)</p> <p>--</p> <p>Weekly meeting with the project leader (a researcher in the institute): presentation of the progress and difficulties, assessment of alternative options proposed distribution of work within the team.</p> <p>For the main phases of the project, preliminary work in teams on a mini-project (application size reduced), followed by a correction with the teaching assistants.</p> <p>At various stages, individual quizzes based on this mini-project to ensure that every student has the tools necessary to effectively contribute to advancing the work of his team.</p>
<p>Content :</p>	<p>This project consists of the development of a realistic application, representative of a typical industrial software system, under semi-professional working conditions. The topic of the application to be constructed, as well as the development methodology to be used, will be proposed by an industrial Partner who participates in the organisation of this course. The project will be carried out by groups of 6 to 8 students.</p>
<p>Bibliography :</p>	<p>--</p> <p>slides of the INGI2251 course online</p> <p>--</p> <p>statements and schedules of the project online</p> <p>--</p> <p>partial and optional support: « Requirements engineering, From System Goals to UML Models to Software Specification », A. van Lamsweerde, Wiley, 2009</p>
<p>Cycle and year of study :</p>	<p>&gt; <a href="#">Master [120] in Computer Science and Engineering</a></p> <p>&gt; <a href="#">Master [120] in Biomedical Engineering</a></p>
<p>Faculty or entity in charge:</p>	<p>INFO</p>