

3.0 credits

30.0 h

2q

Teacher(s) :	Van Lamsweerde Axel ; Mens Kim ;
Language :	Anglais
Place of the course	Louvain-la-Neuve
Prerequisites :	-- know and have experimented techniques and concepts associated with software development of large-size applications (as taught in the course INGI2251)
Main themes :	<p>Current topics of software engineering such as:</p> <ul style="list-style-type: none"> <li>-- modeling, analysing, and improving software processes,</li> <li>-- reusing software products and processes,</li> <li>-- engineering secure applications,</li> <li>-- designing and analysing software architectures,</li> <li>-- domain-specific architectures,</li> <li>-- agent-oriented software,</li> <li>-- software testing,</li> <li>-- software engineering environments,</li> <li>-- formal methods,</li> <li>-- knowledge-based software engineering,</li> <li>-- software reengineering,</li> <li>-- software engineering techniques for specific types of systems: open systems, web services, reactive systems, real-time systems, safety-critical systems, mobile systems, ubiquitous systems, hybrid systems, etc.</li> </ul>
Aims :	<p>Students completing successfully this course will be able to</p> <ul style="list-style-type: none"> <li>-- Understand and explain recent results from software engineering research.</li> <li>-- Assess the current state of the art in specific areas, and discuss open issues.</li> <li>-- Interrelate current issues of software engineering.</li> </ul> <p>Students will have developed skills and operational methodology. In particular, they have developed their ability to</p> <ul style="list-style-type: none"> <li>-- self study (search for relevant material, assimilate/understand, evaluate/compare, summarise/illustrate)</li> <li>-- explain to others using efficient modern supports (slides, ...)</li> <li>-- participate actively to discussions in a research group</li> </ul> <p><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></p>
Evaluation methods :	<p>Presentation of ±2 hours Report of ±20 pages</p> <ul style="list-style-type: none"> <li>-- as a reporter ... provides a personal synthesis of another session</li> <li>-- as an illustrator ... shows an interesting instantiation of the patterns explain in some session, on an application of your choice</li> </ul>
Teaching methods :	<p>Each student will play 3 different roles (in different sessions) :</p> <ul style="list-style-type: none"> <li>-- presenter at one session</li> <li>-- reporter for another session</li> <li>-- illustrator for yet another session</li> </ul> <p>The last two involve to write of an individual report.</p>
Content :	-- architectural design patterns
Bibliography :	<ul style="list-style-type: none"> <li>-- D. Schmidt, M. Stal, H. Rohnert and F. Buschmann. Pattern-Oriented Software Architecture ' Patterns for Concurrent and Networked Objects. Wiley, 2001.</li> <li>-- F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad and M. Stal. Pattern-Oriented Software Architecture ' A System of Patterns. Wiley, 1996.</li> <li>-- E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns ' Elements of Reusable Object-Oriented Software. Addison-Wesley, 1995.</li> </ul>
Cycle and year of study :	<p>&gt; <a href="#">Master [120] in Computer Science and Engineering</a> &gt; <a href="#">Master [120] in Computer Science</a></p>
Faculty or entity in charge:	INFO

