

SYLLABUS COURSE DESCRIPTION

COURSE TITLE	Software Engineering
COURSE CODE	76215
SCIENTIFIC SECTOR	INF/01
DEGREE	Bachelor in Computer Science
SEMESTER	2nd Semester
YEAR	2nd year
CREDITS	6

TOTAL LECTURING HOURS	40
TOTAL LAB HOURS	20
ATTENDANCE	Generally, attendance is not compulsory, but non-attending students have to contact the lecturer at the start of the course to agree on the modalities of the independent study. Non-attending students will still be assessed under the same modalities as regularly attending students regarding the project organization/assessment and final exam.
PREREQUISITES	Students should have done the following courses: Computer Programming, Programming Project
COURSE PAGE	https://ole.unibz.it/

SPECIFIC EDUCATIONAL OBJECTIVES	<ul style="list-style-type: none"> • Type of course: caratterizzanti • Scientific area: discipline informatiche <p>The course introduces the state-of-the-art in software engineering. It aims to demonstrate how this is transferred into practically applicable knowledge and skills for software development.</p>
--	--

LECTURER	<u>Claus Pahl</u>
SCIENTIFIC SECTOR OF THE LECTURER	INF/01
TEACHING LANGUAGE	German

OFFICE HOURS	After the lecture/lab times or by appointment (email). Faculty of CS, Piazza Domenicani 3, Office 1.11
TEACHING ASSISTANT	Claus Pahl
OFFICE HOURS	-
LIST OF TOPICS COVERED	<ul style="list-style-type: none"> • Software life-cycle: principles and methodologies • Software processes and software project management • Requirements engineering: elicitation and modeling • System modeling and construction: UML, design patterns • Software testing: principles and techniques • Software management and evolution
TEACHING FORMAT	Frontal lectures, exercises, projects.

<p>LEARNING OUTCOMES</p>	<p>Knowledge and understanding:</p> <ul style="list-style-type: none"> • D1.8 To have a thorough knowledge of the main fundamentals techniques and methods of software design, development and maintenance <p>Applying knowledge and understanding:</p> <ul style="list-style-type: none"> • D2.5 Be able to apply the own knowledge to the analysis, design, development and testing of information systems which satisfy given requirements • D2.10 Be able to solve typical problems in computer science based on software engineering methodologies, such as the definition of requirements, the analysis of possible methods for a solution, the selection of the most appropriate methods and tools as well as their application • D2.11 Be able to evaluate the quality of information systems and to identify critical aspects • D2.18 Be able to apply the own knowledge in different working contexts <p>Ability to make judgments</p> <ul style="list-style-type: none"> • D3.1 Be able to collect and interpret useful data and to judge information systems and their applicability • D3.2 Be able to work autonomously according to the own level of knowledge and understanding • D3.3 Be able to take the responsibility for development of projects or IT consulting. <p>Communication skills</p> <ul style="list-style-type: none"> • D4.1 Be able to use one of the three languages English, Italian and German, and be able to use technical terms and communication appropriately. • D4.3 Be able to negotiate with a customer for the definition of the pre-requisites and features of information systems. • D4.4 Be able to structure and write scientific documentation. • D4.5 Be able to work in teams for the realization of IT systems. <p>Ability to learn</p> <ul style="list-style-type: none"> • D5.1 Have developed learning capabilities to pursue further studies with a high degree of autonomy. • D5.2 Have acquired learning capabilities that enable to carry out project activities in companies, public institutions or in distributed development communities. • D5.3 Be able to follow the fast technological evolution and to learn cutting edge IT technologies and innovative aspects of last generation information systems.
---------------------------------	---

<p>ASSESSMENT</p>	<p>Written and project work: written exam with verification questions and written project report done in groups. In case of a positive mark the project will count for all 3 regular exam sessions. Projects have to be submitted BEFORE the final exam at the end of the semester, otherwise the exam cannot be registered.</p>
--------------------------	---

ASSESSMENT LANGUAGE	German
EVALUATION CRITERIA AND CRITERIA FOR AWARDING MARKS	<p>Weighting of parts:</p> <ul style="list-style-type: none"> - 60% written exam - 40% exercises/project. <p>Criteria:</p> <p>Relevant for assessment of project and exam:</p> <ul style="list-style-type: none"> - clarity of answers, - mastery of language, - skills in critical thinking, - ability to summarize, evaluate, and establish relationships between topics, - technical competence <p>Relevant for project assessment:</p> <ul style="list-style-type: none"> - ability to work in a team, - creativity, - development skills

REQUIRED READINGS	The course will be based on lecture notes
SUPPLEMENTARY READINGS	I. Sommerville. Software Engineering. Addison Wesley.
SOFTWARE USED	Software Modelling (e.g. Argo UML, Papyrus, StarUML, UMLet)