COURSE DESCRIPTION – ACADEMIC YEAR 2017/2018

Course title	**Software Process and Project Management**
Course code	76028
Scientific sector	INF/01
Degree	European Master in Software Engineering (LM-18)
Semester	1
Year	1
Credits	8
Modular	No

Total lecturing hours	48
Total lab hours	24
Total exercise hours	--
Attendance	Not compulsory
Prerequisites	Basic software engineering and programming skills.
Course page	https://ole.unibz.it/

Specific educational objectives	The course belongs to the type "caratterizzanti – discipline informatiche" (EMSE-SPPM). Building the capacity and ability to define, manage and improve software process and project are the main focus of this topic. The course includes techniques and methods for managing the process of development and coordinating project artefacts in all its stages. The students will also be introduced to techniques of decision-making for software processes. The course provides a comprehensive background for understanding and managing the development process in a software company including both SMEs and IT departments of large companies focusing on novel development approaches. The course describes the different development processes highlighting of the differences and the limitations that characterize each of them. In particular, the course focuses on the differences between traditional and agile approaches to make the students aware of the advantages and disadvantages that such approaches have in different contexts.

Lecturer	Davide Taibi
Contact	Piazza Domenicani 3, Room 3.13, davide.taibi@unibz.it +39 0471 016185
Scientific sector of lecturer	ING-INF/04
Teaching language	English
Office hours	Monday 09:00 - 11:00, arrange beforehand by email.
Lecturing Assistant (if any)	---
Contact LA	---
Office hours LA	---
List of topics	• The Plan-based approach • The Agile approach • Personal Software Process • Team Software Process • Process Improvement

	• Defining and implementing projects with Process Improvement Plan • Defining and implementing projects with Measurement Plan • Process Improvement in field
Teaching format	Frontal lectures and lab

Learning outcomes	Knowledge and understanding • Have solid knowledge in both foundations and applications of different areas of Computer Science: know the main methods and techniques for designing, creating • Know in detail principles, structures and use of elaboration systems: maintaining software products and services and know the main methods for (re)engineering, refactoring and optimizing software products and processes. • Know foundations, techniques and methods for designing and implementing both basic and applicative information systems, with special emphasis on engineering principles applied to conception, creation, development, test and maintenance of software systems. • Have an understanding of business and professional culture: know the main methods of team, resource management and risks analysis in software development and maintenance. Applying knowledge and understanding • Be able to apply engineering principles in different domains of different complexity, both those typically IT related and those non-IT related, where software technology has great relevance, such as, for example, in logistics and in medicine: be able to design and implement information systems in vertical sectors of applications according to technical, functional and organizational requirements. • Be able to define an innovative technical solution to an application problem, which respects constraints and requirements of technical, functional and organizational kind: be able to apply innovative methods for management and improvement of development processes in different application domains such web or mobile. • Ability to extend and modify in an original way an existing technical solution or a formal model, taking into consideration changed conditions and requirements: be able to use and adapt process modelling software tools for the development of information systems. Making judgments • Be able to plan and re-plan a technical project activity aimed at building an information system and to bring it to completion by meeting the defined deadlines and objectives. • Be able to identify reasonable work goals and estimate the resources required to achieve the objectives. Communication skills • Be able to present in a fixed time the content of a scientific / technical report in front of an audience also composed of non-specialists. • Be able to coordinate the work of a project team and to interact positively with members of the group.

	Learning skills • Be able to autonomously extend the knowledge acquired during the study course by reading and understanding scientific and technical documentation in Italian, German and English. • Be able to independently keep up to date with developments in the most important areas of Computer Science.

Assessment	The assessment of the course consists of two parts: • project (50%); • oral exam (50%).
Assessment language	English
Evaluation criteria and criteria for awarding marks	The project activity will be assessed based on how students approach the development of a software process improvement project, their ability to work in a team, creativity, skills in critical thinking, and the ability to summarize the contents of the lecture in their own words. The oral exam will be assessed based on the acquired knowledge and the understanding of the material presented during lectures, the clarity of answers, mastery of language (also with respect to teaching language), and the ability to summarize, evaluate, and establish relationships between topics.

Required readings	Lecture notes that will be distributed during the lecturing hours
Supplementary readings	• Janes, A., Succi, G. Lean Software Development In Action, Springer, 2014. • Fenton, N. E., Pfleeger, S. H. Software Metrics: a Rigorous and • Practical Approach, Thomson Computer Press, 1994. • Ruhe, G., Wohlin C. eds. Software Project Management in a Changing World, Springer, 2014.
Software used	Java, Eclipse, Visual Studio