COURSE DESCRIPTION – ACADEMIC YEAR 2019/2020

Course title	**Contemporary Software Development**
Course code	76051
Scientific sector	INF/01
Degree	Master in Software Engineering for Information Systems (LM-18)
Semester	1
Year	1
Credits	6
Modular	No

Total lecturing hours	40
Total lab hours	20
Total exercise hours	--
Attendance	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. The exam modalities for non-attending students are indicated below, in the fields “Assessment” and “Evaluation criteria and criteria for awarding marks”.
Prerequisites	Students should be familiar with computer programming.
Course page	https://ole.unibz.it/

Specific educational objectives	The course belongs to the type caratterizzanti – discipline informatiche and is part of the Foundations in Software Engineering. The course is designed to give specific professional skills. In particular, students will learn how to apply software development techniques and tools that are used in contemporary working environments. These include choosing a software development process, choosing a testing and measurement strategy, high-level object-oriented designs and design patterns, and skill sets such as debugging, refactoring, integration, as well as critical aspects like software quality.

Lecturer	Andrea Janes
Contact LA	Office POS 1.16, andrea.janes@unibz.it, +39 0471 016132
Scientific sector of lecturer	INF/01
Teaching language	English
Office hours	By appointment, office POS 1.16
Lecturing Assistant (if any)	Michael Mairegger
Contact LA	Office POS 1.16, michael.mairegger@unibz.it
Office hours LA	By appointment, office POS 1.16
List of topics	• Software development environments (e.g., Visual Studio Code, IntelliJ IDEA) • Configuration management (e.g., GIT) • Software artifact management (e.g., Maven) • Design and programming techniques in practice (e.g., C4 model for software architecture, refactoring, object-oriented design patterns in practice, micro-service architectures) • Tools and techniques for process management and quality assurance (e.g., SonarQube, Gerrit, Kanbanflow, Jira) • Continuous integration (e.g., Gitlab CI, Jenkins, Microsoft Visual Studio Team Services)

Teaching format	Frontal lectures, lab exercises, and individual projects

Learning outcomes	Knowledge and understanding • D1.5 To know the fundamentals, techniques and methods of design, customization and implementation of software to support the automation of new generation information systems for industrial production and business; Applying knowledge and understanding • D2.2 To be able to design and perform experimental analyses of information systems in order to acquire measures related to their behaviour and to evaluate experimental hypotheses in different fields of application, such as business, industrial or research; • D2.4 To be able to define an innovative technical solution to an application problem that meets technical, functional and organisational constraints and requirements; Making judgments • D3.2 To be able to plan and re-plan a technical project activity and to carry it out in accordance with defined deadlines and objectives; Communication skills • D4.4 To be able to coordinate project teams and to identify activities to achieve project objectives; Learning skills • D5.2 To be able to keep up to date independently with developments in the most important areas of information technology;

Assessment	The assessment is based on the lab assessment and the final exam. The lab assessment is composed of weekly assignments. The weekly assignments motivate the students to study throughout the semester. The final exam evaluates the students' understanding of the theoretical backgrounds and solving smaller, individual programming tasks. Both, attending and non-attending students will be assessed through the lab assessment and the final exam. Also, both, attending and non-attending students can download the optional weekly assignments from the course web page.
Assessment language	English
Assessment typology	Monocratic commission
Evaluation criteria and criteria for awarding marks	For both, attending and non-attending students, the assessment is based on (i) the lab assessment (up to 15 points) and (ii) the final exam (up to 15 points). The lab assessment consists of weekly assignments. The final mark is the average between the lab assessment score and the final exam score. The lab assessment is a sum of the scores from the weekly assignments. The weekly assignments scores can be obtained only during the lectures period. Relevant for assessment of the weekly assignments is the solution of the given task and the ability to explain the adopted strategy to reach

	the solution. Relevant for the assessment of the final exam: clarity of answers, mastery of language, ability to summarize, evaluate, and establish relationships between topics.

Required readings	Lecture notes will be handed out during the course.
Supplementary readings	---
Software used	An IDE and language of choice of the student, see software mentioned in the “List of topics” above.