

COURSE DESCRIPTION – ACADEMIC YEAR 2020/2021

Course title	Requirements and Design for Dependable Systems
Course code	76052
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 exercise hours	20
Attendance	Not compulsory. Non-attending students have to contact the lecturer at the start of the course to agree on the modalities of the independent study.
Prerequisites	Basic courses in Programming and Software Engineering. Familiarity with UML and software modelling.
Course page	https://ole.unibz.it/

Specific educational objectives	<p>The course belongs to the type caratterizzanti – discipline informatiche and is part of the Foundations in Software Engineering.</p> <p>The course objective is to familiarize students with tools and techniques to acquire and analyze software requirements, and to define and design a software-based system with the focus on dependability as the key concern. Emphasis is given to traceability of requirements to architecture, to justification of design decisions based on collected requirements, to consequences of design decisions for dependable systems.</p> <p>Students will acquire skills and competencies resulting from the conception, negotiation, documentation and maintenance of dependability requirements in a specific domain and environment. Requirements analysis aims at reviewing, assessing, prioritizing, and balancing the requirements by developing technical specifications (functional and non-functional) for building a system that will meet the needs of the stakeholders. The design of dependable software-based systems aims at identifying or building software components that define the characteristics and quality of such a system. The students are exposed to problem-solving techniques that allow the synthesis of software solutions satisfying the requirements of the system.</p>
--	--

Lecturer	Martins Guerra Eduardo
Contact LA	Piazza Domenicani 3, eduardo.guerra@unibz.it
Scientific sector of lecturer	ING-INF/05
Teaching language	English
Office hours	During the lecture times, and by arrangement by email
Lecturing Assistant (if any)	Alton Stephan Ingenuin
Contact LA	Piazza Domenicani 3, StephanIngenuin.Alton@unibz.it
Office hours LA	During the lecture times, and by arrangement by email

<p>List of topics</p>	<ul style="list-style-type: none"> • Requirements Engineering • Software Architecture and Design • Dependable Systems • Reliability and Availability • Safety and Security
<p>Teaching format</p>	<p>Frontal lectures, exercises; team and/or individual projects.</p>
<p>Learning outcomes</p>	<p>Knowledge and understanding</p> <ul style="list-style-type: none"> • D1.1 To have a sound knowledge of both the fundamentals and the application aspects of the various core areas of information technology; • D1.2 To be able to analyze and solve even complex problems in the area of Software Engineering for Information Systems with particular emphasis on the use of studies, methods, techniques and technologies of empirical evaluation; • D1.3 To know in depth the scientific method of investigation applied to complex systems and innovative technologies that support information technology and its applications; <p>Applying knowledge and understanding</p> <ul style="list-style-type: none"> • D2.3 To know how to apply the principles of software engineering to domains of different complexity, both IT and non-IT, in which software technology is of great importance, such as, for example, in the transport sector or in the medical field; • D2.4 To be able to define an innovative technical solution to an application problem that meets technical, functional and organisational constraints and requirements; <p>Making judgments</p> <ul style="list-style-type: none"> • 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; • D3.3 To be able to define work objectives compatible with the time and resources available; <p>Communication skills</p> <ul style="list-style-type: none"> • D4.3 To be able to structure and draft scientific and technical documentation describing project activities; • D4.4 To be able to coordinate project teams and to identify activities to achieve project objectives; • D4.6 To be able to interact and collaborate during the implementation of a project or research with peers and experts; • D4.7 To be able to carry out research and projects in collaborative manner; <p>Learning skills:</p> <ul style="list-style-type: none"> • D5.3 In the context of a problem solving activity, to be able to extend knowledge, even if incomplete, taking into account the final objective of the project;
<p>Assessment</p>	<p>The assessment is based on the lab assessment and the final exam. The lab assessment is composed of a number of assignments. The assignments motivate the students to study throughout the semester.</p>

	<p>The final exam evaluates the students' understanding of the theoretical backgrounds and the ability of solving problems.</p> <p>The students will have the opportunity to perform optional activities during the course, such as:</p> <ul style="list-style-type: none"> • Programming challenges, where the student would need to perform a programming task • Tool or technology research, where the students would need to search information about a tool or technology • Quests, where the student will be challenged to search for real examples in local companies; • Judgements, where students will judge a technology or technique presenting both positive and negative points; • Participation and performance in educational games and quizzes performed in the classroom. <p>The optional activities completed will score points in a gamification system that will be used to give bonus points for the students in their final grade.</p>
Assessment language	English
Assessment typology	Monocratic
Evaluation criteria and criteria for awarding marks	<p>For both, attending and non-attending students, the grade is calculated based on (i) the lab assessment (50% weight) and (ii) the final exam (50% weight).</p> <p>Students, optional activities will be used for a score in a Gamification system. Based on students' position on ranking, they might get some bonus in their final grade.</p>
Required readings	The course will be based on lecture notes.
Supplementary readings	<p>Selected chapters from the following book/ebooks (available with unibz credentials from the library website):</p> <ul style="list-style-type: none"> • Ian Sommerville. Software Engineering, Addison Wesley. Available in library. <p>Open educational resources, representing alternative or supplementary materials, shall be linked to the course website.</p>
Software used	Software Modelling (e.g. Argo UML, Papyrus, StarUML), Java JDK, Java Programming IDE (e. g. Eclipse, IntelliJ)