## Course Title
**Verification and Reliability for Dependable Systems**

### Course Code
76058

### Scientific Sector
INF/01

### Degree
Software Engineering for Information Systems (LM-18)

### Semester
2

### Year
1

### Credits
6

### Modular
No

### Total Lecturing Hours
40

### Total Exercise Hours
20

### Attendance
Not compulsory

### Prerequisites
Students are familiar with practices and methods of modern software product development and statistics and are able to develop software programs in autonomy.

Pre-requisite material can be taught in the following courses:
- Contemporary Software Development
- Requirements and Design for Dependable Systems

### Course Page
https://ole.unibz.it/

### Specific Educational Objectives
The course belongs to the type caratterizzanti – discipline informatiche.

The course defines principles and practices of verification and reliability of software systems that have dependable characteristics. Verification methods aim at checking that the system meets prescribed software specifications. Reliability aims at observing and predicting the capability of a system to operate according to its specifications over a given period of time.

The goal of the course is to prepare the students to recognize the quality characteristics of a system and to develop and maintain a system accordingly.

### Lecturer
Barbara Russo

### Contact
Piazza Domenicani 3, Room 1.16, barbara.russo@unibz.it, 0471-016170

### Scientific Sector of Lecturer
INF/01

### Teaching Language
English

### Office Hours
During the lecture time span, Wednesdays 14:00 - 16:00, arrange beforehand by email.

### Lecturing Assistant (If any)
TBD

### Contact LA
Matteo Camilli, Matteo.Camilli@unibz.it

### Office Hours LA
By previous appointment via e-mail.

### List of Topics
- Dependable properties of systems
- Software and software systems testing
- Techniques for verification of software systems
- Advances in test design and implementation
- Search Based testing
- HW and SW reliability models
### Teaching format
Frontal lectures, inverted classroom model, exercises and discussions in the lab, and work in team is welcome.

### Learning outcomes

| Knowledge and understanding | 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 |
| Applying knowledge and understanding | D2.5 To be able to extend and modify in an original way an existing technical solution or a formal model taking into account changed conditions, requirements and evolution of the technology |
| Making judgments | D3.1 To be able to autonomously select documentation from a variety of sources, including technical books, digital libraries, technical scientific journals, web portals or open source software and hardware tools |
| Communication skills | D4.5 To be able to prepare and conduct technical presentations in English |
| Learning skills | 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 |

### Assessment
The assignments consist of different implementation exercises as well as reading and presentation tasks. Each assignment requires a submission. The assignments will be evaluated periodically and they are a prerequisite for attending the written exam. The assignments aim at assessing to what extent the student has achieved the above-mentioned learning outcomes related to: applying knowledge and understanding, making judgments and communication skills. The written exam will assess to what extent the student has achieved above-mentioned learning outcomes related to: knowledge and understanding, applying knowledge and understanding, ability to learn. In case the project work assessment is positive but the final written exam is not positive, the assignments grade is valid for all three regular exam sessions.

### Assessment language
English

### Assessment typology
Monocratic

### Evaluation criteria and criteria for awarding marks
Final grade: 50% lab assessment + 50% written exam
Lab assessment must be positive (i.e., 18 or higher) to access the written exam.
Final grade pass: 18 or higher.
Relevant for the assessment:
Lab assessment: ability to apply in autonomy and develop further instruments introduced during the lectures/labs and needed to accomplish tasks and perform little studies with data.
Ability to report in a professional manner also using the appropriate terminology and concepts of the course.

Written exam: ability to use the appropriate terminology and concepts of the course and to apply them in different context.
Ability to understand the assumptions under which different techniques/methods can better perform or be used.
Ability to analyze a problem and determine the causes.
Ability to synthetize the results and interpret them in a specific context also using mathematical instruments to compare and evaluate models shaping software systems in testing or reliability.

Required readings
Lecture notes and papers will be handed out during the course.

Supplementary readings
- Laurie Williams et al. http://openseminar.org/se/modules/7/index/screen.do
- Kent Beck: Test Driven Development by Example, Addison-Wesley Verlag

Software used
- R/Python
- Notebook
- Software needed for project development (e.g., testing frameworks)
  In case is needed, students will develop their own tools to mine software reliability data