General Information	**A.A. 2020-2021**
Academic subject	Secure Software Engineering
Degree course	Computer Science (second-level degree in Computer Science)
Curriculum	
ECTS credits	9
Compulsory attendance	No
Language	English

Subject teachers	*Name Surname*	*Mail address*
	Danilo Caivano	danilo.caivano@uniba.it
	Maria Teresa Baldassarre	mariateresa.baldassarre@uniba.it
Office Hours	Location	Day and time
Department of Informatics	VI Floor Room 622	Monday 10:30 -12:30

ECTS credits details			
Basic teaching activities	Lectures Lab/Practical Sessions		

Class schedule	
Period	Second semester
Year	First
Type of Class	Lectures (7 ECTS) Lab/Practical Sessions (2 ECTS)

Time management	
Hours	86
Hours of Lecture	56 (7 credits)
Tutorials and lab	30 (2 credits)

Academic calendar	
Class begins	March 1st, 2021
Class ends	June 4th, 2021

Syllabus	
Prerequisites/requirements	There are not mandatory requirements
Expected learning outcomes (according to Dublin Descriptors) (it is recommended that they are congruent with the learning outcomes contained in the Didactic Regulation and Prospectus a.a. 2017-2018)	• *Knowledge and understanding* The main expected learning result is the knowledge of the most important aspects of secure software engineering, both in theory and practice: the ability to identify and model threats, and to identify and apply techniques and tools in order to avoid security vulnerabilities. Students acquire this knowledge both through lectures and possible participation in specific seminars, and through

	specific exercises, which allow them to put into practice and verify what they have learned, thus acquiring awareness of their ability to understand and how to improve them. • *Applying knowledge and understanding* In order to enable students to apply the acquired knowledge, they perform both individual and collaborative exercises. In addition, students are required to develop a small project in which they apply some of security techniques presented in class, having selected the most appropriate ones for the specific case. This project contributes to the student’s final assessment and thus to the final grade for the course. • *Making judgements* An important objective of the course is that the student achieves the ability to integrate knowledge, handle complexity and make decisions during the secure software development. The exercises performed during the course, which are discussed by teacher and students, are a means to train students to make judgements. This ability is evaluated by the teacher and contributes to the final grade, which also considers the active participation of the student to the discussions in class and the presentation of the project. • *Communication* Students are encouraged to work in groups and are often invited to study and present some course topics to the class (Flipped Classroom) or the outcome of exercises carried out individually or in groups, with the goal of developing their communication skills. Students are also required to develop a small project in which they apply some of the learned techniques for secure software development. The presentation of this project is part of the oral examination and allows the student to demonstrate his/her communicative abilities by illustrating the performed work using some slides previously prepared. • *Learning skills* In order to stimulate their own learning skills, students are solicited to deepen some topics not discussed in detail by the teacher, using books and/or other sources different from the textbook. Students will present these topics in class following the Flipped Classroom model. Students are also invited to attend seminars held by other lecturers, internal to the department or visiting researchers, and they will be asked to discuss in class the content of such seminars
Course Program	**Introduction** • Application vulnerability scenario • Security life cycle • Secure software application

	Key Elements • Vulnerability • Exploit • Risk • Threat • Privacy and Secure by Default • Defence in Depht **Secure Software Development** • Secure Software Development Life Cycle (SSDLC) o Threat modeling o Risk analysis o Architectural security o Secure coding o Secure configuration and deployment o Secure monitoring • Maturity Models **Privacy by Design in software development** • Principle of Privacy by Design • Privacy Design Strategies • Privacy Pattern • Privacy Enhancing Technologies • Use cases and scenario • POSD (Privacy Oriented Software Development) approach **Cyber attacks and techniques**
Bibliography	Gary McGraw, "Software Security: Building Security In". Addison-Wesley, 2006. ISBN-10: 9780321356703
Notes	
Teaching methods	Lectures in class with the support of slides prepared by the teacher. Lab/Practical sessions in order to help students experience a secure development life cycle first hand. A small project to be developed in group, with the supervision of the teacher
Assessment methods (indicate at least the type written, oral, other)	The assessment method used during the final exam includes: • An oral presentation which illustrates and discusses the project developed in a group. The project is assigned during the course semester. The project must be delivered 3 working days before the date of the exam. The positive evaluation of a project is valid for the current academic year. • A written test consisting in answering a questionnaire containing closed or open-ended questions.

Evaluation criteria (Explain for each expected learning outcome what a student has to know, or is able to do, and how many levels of achievement there are).	To ascertain the knowledge acquired by the student, and also his/her autonomy of judgment, communication skills and the ability to learn, it is planned to: • Evaluate, through an oral presentation, the project carried out (in a group) considering how it was structured, how the principles and methodologies were applied, the appropriateness of the techniques used, the originality of the solutions, clarity and of the synthesis capacity that results from the documentation produced (written report, oral and possible presentation through slides). The contribution of the single student to the group work will be evaluated. The evaluation is in thirtieths and is the same for the whole working group. Individual evaluation is therefore obtained with bonus or malus points in relation to: the contribution given to the group in the realization of the project; the ability to synthesis as well as clarity of exposure; the ability to make meaningful comparisons between different methodologies, techniques and technologies and to report one's own critical judgment, to mastering the technical terms. • Evaluate through a written test consisting in answering a questionnaire containing closed or open questions, the knowledge acquired during the course. The final mark is in thirtieths. The final grade will be obtained as the weighted average of the grade obtained for the oral presentation of the project (60%) and that of the written test (40%).
Further information	