UNIVERSITÀ DI PISA

LABORATORY FOR INNOVATIVE SOFTWARE

CHIARA BODEI

Academic year	2023/24
Course	INFORMATICA
Code	652AA
Credits	6

Modules	Area	Type	Hours	Teacher(s)
LABORATORY FOR INNOVATIVE SOFTWARE	INF/01	LABORATORI	48	CHIARA BODEI GIAN-LUIGI FERRARI

Obiettivi di apprendimento

Conoscenze

By the end of this class, students will have developed skills in three distinct competency areas
Reliable coding:
Writing code that is well organized at a high level; exploiting the best programming language features appropriately and avoiding troublesome ones; applying sophisticated idioms to structure code elegantly; using innovative toolkits to check program properties including automatable unit tests in the code base; preventing security attacks.
Design:
Analyzing problems to understand what the tricky aspects are; identifying key design issues, and analyzing their tradeoffs; selecting features for a minimal viable product.
Professionalism:
Constructing and delivering presentation of the deployed software; collaborating with team members; making constructive critiques.

Modalità di verifica delle conoscenze

Ongoing assessment to monitor and evaluate the progress will be carried out in the form of meetings between the lecturer and a group of students developing the programming projects.

Capacità

By the end of the course:

- Students will know how to eploit advanced programming toolkis in the design and development of innovative software applications.
- Students will be able to conduct to analysis of code source developed by third parties
- Students will be able to present, in a written report the results of their software design.

Modalità di verifica delle capacità

Practical lab activities will be carried out. During the lab session, students will be free to work on the lab exercises on their own or with others, and the course lecturer and lab demonstrator will be available to answer any questions.
Students will have to prepare and present a written report that documents the results of the project activity

Comportamenti

Students will acquire an awareness of the issues associated with the choice of spoecific advannced toolkits in the desiogn and development of innovative software
Students will be able to manage the responsibility of working within a team project
Students will acquire accuracy and precision when desiging, programming, analysing and delivering software apps.

Modalità di verifica dei comportamenti

During the lab sessions, the accuracy and precision of the activities carried out will be evaluated
During group work, the methods of assigning responsibility, management and organisation during the project phases will be evaluated

Prerequisiti (conoscenze iniziali)

The fundamentals of the software curriculum.

UNIVERSITÀ DI PISA

Indicazioni metodologiche

Cooperative leaning method taking into account the characteristics of the group of students and the activity to be worked on

Programma (contenuti dell'insegnamento)

Practical development of software requires an understanding of successful methods for bridging the gap between a problem to be solved and a working reliable software system. This course will train the student to develop large software systems working in real projects by exploiting the techniques and the skills acquired in the fundamental courses of the curriculum.
The main novelty of the course is the attempt to balance traditional lectures and experimental activities with technical meetings with software architects of innovative software enterprises. During the course students will face and deal with the up-to-date issues of software design, implementation and testing of real projects. In this way students will also learn how to inspect actively software solutions.
Each time the course is offered the design and implementation of a new innovative software artifact will be addressed, however the main underlying theme will always be building reliable code. To this purpose the course experiments modern techniques for making software more robust. These techniques include, but are not limited to:

- *Ad hoc static code analyses and tools.*
- *Model checkers.*
- *Code verification.*
- *Machine learning techniques applied to code analysis.*
- *Undefined behavior detectors.*
- *Testing frameworks.*
- *Language-based security frameworks.*

Modalità d'esame

The exam is made up of of an oral discussion of a lab project.

Altri riferimenti web

Ultimo aggiornamento 10/12/2023 19:58