

Innopolis University

Course Info

Course: Software Architectures, Spring 2016
Lectures Time: Monday from 9AM to 10:30 AM, Room 108
Labs Time: Monday from 10:40 to 12:10, or
Monday from 13:20 to 14:50,
Depending on the group you have enrolled.

Instructor: Néstor Cataño
Email: n.catano@innopolis.ru
Office Hours: Tuesdays from 9AM to 11AM; otherwise, by email.
WEB: <http://poporo.uma.pt>

Summary

This course explores fundamental techniques for producing, deploying and analysing software systems. The course includes three components:

1. Software architecture in the strict sense: designing modular structures for reliability, extendibility and reusability. Software architectures address the overall organization of software systems and the techniques that make the development of large systems possible. Not all programming techniques that work on small programs scale-up; this course explores some that do.
2. Non-programming, non-design aspects of software engineering, such as lifecycle models, metrics, and quality assurance (including testing and other verification methods).
3. A strong practical content in the form of a collaborative project that applies the techniques discussed.

Course Objectives

Students will learn:

1. To be effective software engineers.
2. To manage complexity in large software systems.
3. To design, implement, verify and maintain efficient systems.
4. To work in a development team.
5. To use principles, methods, and good practices in software architectures and software design for developing software

Pre-requisite

- Previous exposure to Object Oriented Programming.

On-Line Course Materials

Lecture slides and course material will be reachable from the course web site.

Textbooks and Materials

1. Software Architecture in Practice, Third Edition, by Len Bass, Paul Clements and Rick Kazman.
2. Pattern-Oriented Software Architecture: A System of Patterns, by Frank Buschmann et al., WILEY.
3. Software Engineering, by Ian Sommerville, 9th Edition, Pearson Education.
4. Agile Principles, Patterns and Practices in C#, by Robert C. Martin.
5. Object-Oriented Software Construction, by Bertrand Meyer, Prentice-Hall, 1997.
6. Design Patterns: Elements of Reusable Object-Oriented Software, by E. Gamma, R. Helm, R. Johnson, and J. Vlissides.
7. A Discipline of Programming, by Edsger W. Dijkstra, Prentice-Hall, 1976.
8. Testing Computer Software, Second Edition, by Cem Kaner and Jack Falk, Wiley.
9. UML Distilled: A Brief Guide to the Standard Object Modelling Language, 3rd edition, by Martin Fowler.

Grading

Activity	%	When?
Project	25%	See project statement
Homework	25%	Throughout the term
Mid-term	20%	Week 08
Final exam	30%	Week 18

Plagiarism Policy

Although it is allowed to share ideas with other students so as to get feedback ideas one has in mind, and although it is allowed to search Internet for problems and solutions, copy-paste plagiarism is not allowed. Hence, you are not allowed, under any circumstances, to copy code or ideas from Internet, or from other students or sources, and to make these ideas (or the code) as they are yours.

Lectures, Midterm and Assignments Schedule

The following is subject to change and will likely change over the course of the semester. I will inform you about any changes in the syllabus.

Week	Topic	Instructor	Readings
Week 01 , Jan/18	Introduction	N. Cataño	Books 1, 2, 3, 4
Week 02 , Jan/25	Software Requirements	N. Cataño	Book 3, Chapter 4
Week 03 , Feb/01	Software Life Cycles	N. Cataño	Book 3, Chapter 2
Week 04 , Feb/08	Principles of Agile Software Development	N. Cataño	Book 4, Chapters 1-3
	Code Refactoring	N. Cataño	Book 4, Chapter 5
Week 05 , Feb/15	UML: Class Diagrams	N. Cataño	Book 9, Chapters 3, 5
	UML: State Machine Diagrams	N. Cataño	Book 9, Chapter 10
	UML: Activity Diagrams	N. Cataño	Book 9, Chapter 11
	UML: Sequence Diagrams	N. Cataño	Book 9, Chapter 4
Week 06 , Feb/22	Software Testing	N. Cataño	Book 8 Book 4, Chapter
	Test Driven Development (TDD)	N. Cataño	Book 4, Chapter 4

Week 07 , Feb/29	What's Software Architecture?	N. Cataño	Book 1, Chapters 1, 2
	Understanding Quality Attributes	N. Cataño	Book 1, Chapters 4-12
Week 08 , March/7	Architectural Tactics and Patterns	N. Cataño	Book 1, Chapter 13 Book 2, Chapter 2 Book 3, Chapter 6
	MVC	N. Cataño	Book 3, Chapter 6 Book 2, Section 2.4
Week 09 , March/14	Midterm	N. Cataño	
Week 10 , March/21	The SOLID principles for O-O Design General Responsibility Assignment Software Patterns (GRASP)	N. Cataño	Book 4, Chapters 8-12
Week 11 , March/28	Abstract Data Types	B. Meyer	Book 5, Chapter 6
	Modularity	B. Meyer	Book 5, Chapter 3
Week 12 , April/04	Design-by-Contract	B. Meyer	Book 5, Chapter 11
	Program Correctness	B. Meyer	Book 7
Week 13 , April/11	Creational Patterns: Abstract Factory, Builder, Factory Method, Prototype, Singleton	N. Cataño	Book 2, Chapter 3 Book 6, Chapter 3
Week 14 , April/18	Structural Patterns: Adapter, Bridge, Façade, Composite, Decorator	N. Cataño	Book 2, Chapter 3 Book 6, Chapter 4
Week 15 , April/25	Behavioural Patterns: State, Command, Interpreter, Iterator, Observer, Visitor	N. Cataño	Book 2, Chapter 3 Book 6, Chapter 5
Week 18 , May/09	Final Exam		