

# Guida rapida all'insegnamento

## Ingegneria del software 2

Docente: Giuseppe Scollo

Università di Catania, sede di Comiso (RG)

Facoltà di Scienze Matematiche, Fisiche e Naturali  
Corso di Studi in Informatica applicata, AA 2006-7

### Indice

1. Guida rapida all'insegnamento
2. Organizzazione dell'insegnamento
3. Obiettivi formativi
4. Attività formative
5. Relazione con altri insegnamenti
6. Modalità di valutazione
7. Programma delle lezioni
8. Testi consigliati
9. Strumenti per l'interazione formativa
10. Siti di riferimento
11. Strumenti open-source

## Organizzazione dell'insegnamento

- **modalità:** blended e-learning
  - come per Ingegneria del software 1, però: meno Forum, più Wiki!
- sviluppo progettuale cooperativo
  - non solo di progetti di laboratorio, ma anche dello stesso progetto formativo dell'insegnamento
- condivisione di obiettivi formativi
- "imparare a non programmare" - 2

## Obiettivi formativi

Acquisizione di:

- metodi di progettazione e specifica di architetture software
  - con particolare attenzione alla progettazione orientata agli oggetti
- competenze professionali nell'impiego dei linguaggi standard UML e OCL
  - per la modellazione, specifica e sviluppo di tali architetture
- competenze professionali nell'uso di strumenti di laboratorio
  - di supporto a tali attività

## **Attività formative**

L'organizzazione didattica dell'insegnamento prevede

24 ore di lezione

L'acquisizione di metodi e competenze professionali nella disciplina è sostenuta dallo

studio delle note delle lezioni, e di eventuali altre fonti di approfondimento,

e dalla sua applicazione nella

sperimentazione dell'uso di strumenti di laboratorio

per la

realizzazione di progetti autogestiti

## **Relazione con altri insegnamenti**

Quale bagaglio preliminare è

- raccomandato
  - significa: nessun vincolo di propedeuticità sugli esami, ma seguire i due insegnamenti nella sequenza naturale è ottimale per il conseguimento degli obiettivi formativi

l'insegnamento di Ingegneria del software 1

## **Modalità di valutazione**

La verifica del profitto avviene con un colloquio individuale sugli argomenti del programma,

- a partire dal lavoro svolto dallo studente in un progetto,
- preferibilmente realizzato in collaborazione con altri studenti.

Il superamento della prova porta all'acquisizione di 3 crediti.

Gli studenti che hanno scelto entrambi gli insegnamenti di Ingegneria del software nel piano di studio possono, se lo trovano conveniente, sostenere entrambi gli esami nello stesso giorno, in contiguità temporale dei due colloqui.

## **Programma delle lezioni**

1. Modellazione di architetture software, introduzione a UML e MDA
2. Diagrammi delle classi: concetti fondamentali
3. Diagrammi delle classi: concetti avanzati
4. Strumenti per lo sviluppo di modelli UML e di software da essi
5. Costruzione di un modello di rete di servizi
6. Diagrammi di interazione
7. Diagrammi di macchine a stati
8. Diagrammi di attività e di interazione generale
9. Altri diagrammi strutturali
10. Specifica di vincoli in modelli UML con OCL
11. Costruzione di modelli con OCL
12. Tipi e costrutti avanzati di OCL

## Testi consigliati

M. Fowler

UML Distilled, Terza Edizione

Addison-Wesley, Pearson Education Italia (2004)

E. Damiani, M. Madravo, A., Böhm

UML pratico, con elementi di ingegneria del software, Seconda Edizione

Addison-Wesley, Pearson Education Italia (2007)

J.B. Warmer & A.G. Kleppe

The Object Constraint Language, Second Edition

Addison-Wesley (2003)

## Strumenti per l'interazione formativa

meno Forum, più Wiki: perché?

- meno Forum: non ci sono esercizi specifici per le singole lezioni, mentre restano le discussioni di
  - argomenti delle lezioni
  - idee di progetto
  - uso di strumenti di laboratorio
  - organizzazione dell'insegnamento
- più Wiki: è incoraggiata la documentazione in corso d'opera dello sviluppo di modelli e specifiche, facendo pieno uso della funzionalità Wiki per il controllo di versione e per la gestione di discussioni

## **Siti di riferimento**

**Object Management Group (OMG):** <http://www.omg.org>

**OMG Unified Modeling Language (UML):** <http://www.uml.org>

**UML Object Constraint Language (OCL):** <http://www.klasse.nl/ocl>

### **Altri siti rilevanti**

**Objects by Design:** <http://www.objectsbydesign.com>

**Precise UML:** <http://www.cs.york.ac.uk/puml>

## **Strumenti open-source**

**AndroMDA:** <http://www.andromda.org>

**ArgoUML:** <http://argouml.tigris.org>

**BOUML:** <http://bouml.free.fr>

**Gaphor:** <http://gaphor.sourceforge.net>

**Dresden OCL Compiler:** <http://dresden-ocl.sourceforge.net>

**USE - A UML-based Specification Environment:**  
<http://www.db.informatik.uni-bremen.de/projects/USE>