# INGEGNERIA DELL'INFORMAZIONE (LB08)

(Lecce - Università degli Studi)

## Insegnamento PRINCIPI DI PROGETTAZIONE DEL SOFTWARE

GenCod A002443

**Docente titolare** LUCA MAINETTI

**Insegnamento** PRINCIPI DI PROGETTAZIONE DEL SOFTWARE

**Insegnamento in inglese** SOFTWARE DESIGN PRINCIPLES

**Settore disciplinare** ING-INF/05

**Anno di corso** 2

**Lingua** ITALIANO

**Percorso** PERCORSO COMUNE

**Corso di studi di riferimento** INGEGNERIA DELL'INFORMAZIONE

**Tipo corso di studi** Laurea

**Sede** Lecce

**Crediti** 6.0

**Periodo** Primo Semestre

**Ripartizione oraria** Ore Attività frontale: 54.0

**Tipo esame** Orale

**Per immatricolati nel** 2018/2019

**Valutazione** Voto Finale

**Erogato nel** 2019/2020

**Orario dell'insegnamento** https://easyroom.unisalento.it/Orario

### BREVE DESCRIZIONE DEL CORSO

Dopo il corso lo studente sarà in grado di:

- Comprendere i requisiti funzionali e non funzionali di un sistema software interattivo.
- Progettare l'architettura software e l'architettura dei dati di un sistema interattivo secondo principi ingegneristici mappando i requisiti in artefatti software.
- Pianificare le fasi dello sviluppo di un sistema software interattivo secondo il metodo agile "Scrum" suddividendo il lavoro da svolgere con un collega.
- Sviluppare un sistema software interattivo in linguaggio Java e la sua base di dati.
- Utilizzare operativamente ambienti di sviluppo Java (IntelliJ, Eclipse) e di basi di dati relazionali (MySQL).

### PREREQUISITI

Si richiedono conoscenze di Fondamenti di Informatica

### OBIETTIVI FORMATIVI

Il corso fornisce agli studenti una visione introduttiva ma globale sulla disciplina sull'ingegneria del software per una piena comprensione dei processi che guidano la progettazione, lo sviluppo e il testing di un sistema software complesso e della sua base di dati. L'approccio didattico è teorico-pratico: ogni concetto esposto è oggetto di applicazione pratica, in particolare grazie allo sviluppo di un elaborato software in linguaggio Java. Si illustrano moderne tecniche e strumenti di progettazione, implementazione e verifica dell'ingegneria del software. Si analizzano i differenti processi di sviluppo del software e si mettono in pratica i metodi agili.

### METODI DIDATTICI

Lezioni frontali, esercitazioni pratiche, elaborazione di un progetto software da svolgersi in coppie.

| | |
|---|---|
| MODALITA' D'ESAME | L'esame prevede due prove, uno scritto volto a verificare l'apprendimento dei concetti teorici, che contribuisce a un massimo di 10 punti su 30; un elaborato software volto a verificare la capacità di applicazione dei concetti teorici, che contribuisce a un massimo di 20 punti su 30. Ai fini dell'elaborato software, in gruppi di due persone, gli studenti devono realizzare un sistema software. Il sistema deve essere progettato con UML. Il sistema deve essere realizzato tramite un linguaggio di programmazione orientata agli oggetti (Java). Il sistema deve accedere dinamicamente ad una base di dati progettata con E/R e gestita da un DBMS relazionale (MySQL). Il gruppo deve lavorare secondo un processo "agile" (Scrum) e documentare il procedimento di lavoro adottato. Un mese prima del termine del corso è pubblicata la traccia dell'elaborato software, una traccia nuova per ogni anno, che rimane valida fino alla pubblicazione della traccia dell'anno accademico successivo. L'esito della prova scritta ha la medesima validità temporale della traccia dell'elaborato software.<br>**COVID19 E MODALITA' D'ESAME**: in osservanza delle indicazioni ricevute dal Magnifico Rettore, durante il periodo di restrizioni governative gli esami saranno svolti esclusivamente on line con uso della piattaforma Microsoft Teams, alla quale tutti gli iscritti agli appelli si devono registrare con le credenziali istituzionali. Sempre in tale periodo, per i candidati che ancora non hanno sostenuto positivamente la prova parziale, lo scritto è sostituito da due domande fatte prima della presentazione del progetto software durante la prova orale. Le domande riguarderanno esercizi di design e saranno del tutto analoghe a quelle comprese nelle prove scritte tradizionali. Agli studenti potrà essere chiesto di rispondere abbozzando diagrammi o specifiche testuali, per le quali potranno utilizzare comuni editor quali word e powerpoint. |
| APPELLI D'ESAME | Vedi www.ing.unisalento.it. |
| ALTRE INFORMAZIONI UTILI | Laurea Ingegneria dell'Informazione, Dipartimento di Ingegneria dell'Innovazione, Settore Scientifico Disciplinare 09-H1, 6 CFU, 48 ore di attività frontale, 44 ore di lavoro individuale svolto in coppie, 2° anno di corso, 1° semestre, lingua italiana.<br>**Orario di ricevimento**: lunedì 15:00-19:00 presso lo studio del docente, Edificio La Stecca, 1° piano, scala centrale fronte bar, Campus Ecotekne. |

PROGRAMMA ESTESO

- I processi di sviluppo del software (4 ore):
  - Processo a cascata, processi iterativi, processi agili.
- Progettazione dell'architettura software (8 ore):
  - UML: requisiti, casi d'uso;
  - UML: introduzione operativa ai diagrammi delle classi, sequenza, stato;
  - UML: diagrammi delle classi, diagrammi di sequenza, attività, stato, fisici.
- Progettazione delle classi (2 ore):
  - Le linee guida: incapsulamento, coesione, coerenza, pre-condizioni, asserzioni, eccezioni, post-condizioni, invarianti, test di unità.
- Fondamenti di basi di dati (6 ore):
  - Il modello Entità-Relazioni;
  - Dal modello Entità-Relazioni al modello Relazionale.

- Un corso veloce di Java (7 ore):
  - Oggetti, classi, tipi di dati fondamentali;
  - Strutture di controllo, riferimenti a oggetti, passaggio parametri;
  - Interfacce, polimorfismo, ereditarietà, eccezioni.
- Creare un'applicazione Java (21):
  - Predisporre l'ambiente di sviluppo;
  - Progettare le classi;
  - Implementare le classi;
  - Accedere a dati esterni;
  - Verificare le classi.

TESTI DI RIFERIMENTO

1. C. S. Horstmann, Progettazione del Software e Design Pattern in Java, Apogeo.
2. C. S. Horstmann, Concetti di Informatica e Fondamenti di Java, Apogeo.
3. M. Fowler, UML Distilled, Pearson Addison-Wesley, 2004.