INGEGNERIA DELL'INFORMAZIONE (LB08)

(Lecce - Università degli Studi)

Insegnamento PRINCIPI DI INGEGNERIA DEL SOFTWARE

GenCod A005784

Docente titolare LUCA MAINETTI

Insegnamento PRINCIPI DI INGEGNERIA DEL SOFTWARE

Anno di corso 2

Insegnamento in inglese PRINCIPLE OF SOFTWARE ENGINEERING

Lingua ITALIANO

Settore disciplinare ING-INF/05

Percorso PERCORSO COMUNE

Corso di studi di riferimento INGEGNERIA DELL'INFORMAZIONE

Tipo corso di studi Laurea

Sede Lecce

Crediti 6.0

Periodo Secondo Semestre

Ripartizione oraria Ore Attività frontale: 54.0

Tipo esame Orale

Per immatricolati nel 2019/2020

Valutazione Voto Finale

Erogato nel 2020/2021

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

BREVE DESCRIZIONE DEL CORSO

Il corso intende fornire conoscenze di ingegneria del software e sviluppare le competenze necessarie per progettare e sviluppare programmi complessi. E' la naturale prosecuzione del corso di Fondamenti di Informatica II poiché richiede una buona padronanza del linguaggio Java, benché i contenuti teorici siano pienamente autonomi.
Nel concreto, il corso affronta in profondità le tecniche di analisi dei requisiti funzionali e non funzionali, progettazione con UML (Uniform Modeling Language) e sviluppo in Java di sistemi software organizzati secondo architetture standard (design patterns) e livelli di disaccoppiamento che ne facilitino l'evoluzione. Ogni concetto esposto è sperimentato in modo pratico insieme agli studenti, utilizzando il personal computer e strumenti di sviluppo moderni e ampiamente diffusi nel mondo industriale.

PREREQUISITI

Elementi di teoria dell'informatica. Principi e basi per la programmazione. Esecuzione di operazioni seguendo una sequenza codificata di istruzioni elementari. Concetto di algoritmo, algoritmi di base e strutture dati. Concetti di programmazione orientata agli oggetti. Linguaggio Java.
Non vi è alcuna propedeuticità ma sono suggerite le conoscenze di Fondamenti di Informatica II.

OBIETTIVI FORMATIVI	*Conoscenze e comprensione*. Al termine del corso gli studenti: (a) conosceranno i principi dell'ingegneria del software, in relazione alle caratteristiche e alle qualità del software; (b) conosceranno nel dettaglio le architetture software standard, i design patterns e i principi di riuso del software; (c) conosceranno e sperimenteranno i processi di sviluppo agili del software; (d) approfondiranno linguaggio Java in particolare per la costruzione di architetture software evolvibili; (e) comprenderanno le tecniche di condivisione del software e i sistemi di controllo delle versioni; (f) approfondiranno i principali ambienti di sviluppo Java anche in relazione alle necessità di sviluppo in coppia di applicazioni complesse. *Capacità di applicare conoscenze e comprensione*. Gli studenti saranno in grado di applicare le conoscenze acquisite in diversi ambiti applicativi e, in generale, per la codifica al computer in linguaggio Java di applicazioni software con principi di riuso ed evolvibilità. *Autonomia di giudizio*. Il corso favorisce l'autonomia di giudizio degli studenti attraverso l'analisi critica di problemi di modellazione del software da requisiti funzionali e non funzionali, per i quali trovare le soluzioni adeguate a risolverli in linguaggio Java con riferimento a noti design patterns. Diverse soluzioni proposte interattivamente dagli studenti saranno poste a confronto e valutate criticamente dagli studenti stessi. *Abilità comunicative*. Gli studenti apprenderanno come comunicare adeguatamente e con il corretto livello di formalismo le scelte di design adottate e le strategie di implementazione scelte. Il metodo di insegnamento interattivo e teorico/pratico favorirà momenti di confronto in cui mettere in pratica tali abilità comunicative. Lo sviluppo di prototipi software in coppia richiederà tecniche di comunicazione allo stato dell'arte per l'ingegneria del software *Capacità di apprendimento*. La materia in costante evoluzione (le tecniche di sviluppo orientate alle architetture software standard, i linguaggi che le implementano, gli strumenti di sviluppo) richiederà agli studenti la capacità di aggiornarsi e di ricercare materiale on-line, valutandone anche la qualità. Il metodo didattico favorirà l'approfondimento autonomo e in coppie da parte degli studenti, incuriosendoli su tecniche di sviluppo evolute (vedi il test driven development e le metriche del software) che saranno oggetto del corso magistrale di Software Engineering.
METODI DIDATTICI	Lezioni on-line o frontali, esercitazioni pratiche svolte preferibilmente con l'uso del personal computer, elaborazione a coppie di un'applicazione Java.
MODALITA' D'ESAME	L'esame prevede una prova orale per la verifica dell'apprendimento dei concetti teorici (verifica delle conoscenze) e della capacità di applicazione dei medesimi, in particolare per la codifica in coppie di applicazioni Java complesse (verifica delle competenze). Durante l'esame, che sarà svolto in coppie, gli studenti dovranno usare preferibilmente il loro personal computer, configurato con gli ambienti di sviluppo illustrati e utilizzati durante il corso.
APPELLI D'ESAME	Si veda www.ing.unisalento.it.
ALTRE INFORMAZIONI UTILI	www.unisalento.it/people/luca.mainetti

PROGRAMMA ESTESO

Presentazione dettagliata del corso e delle modalità d'esame (2 ore). Processi di sviluppo del software (2 ore). Processi agili (2 ore). UML: requisiti e casi d'uso (2 ore). UML: dai requisiti ai diagrammi delle classi, di sequenza e di stato (2 ore). UML: diagrammi delle classi, di sequenza, di stato, dei package e di deployment nel dettaglio (2 ore). Introduzione alle architetture software standard (2 ore). Design patterns (10 ore). Fondamenti di basi di dati: il modello Entità-Relazioni. (2 ore). Fondamenti di basi di dati: dal modello Entità-Relazioni al modello Relazionale. (2 ore). MySQL e SQL (4 ore). Presentazione dell'elaborato software (2 ore). Creare un'applicazione Java: predisporre l'ambiente di sviluppo e l'architettura software (4 ore). Creare un'applicazione Java: implementare le classi (4 ore). Creare un'applicazione Java: accedere a dati esterni (4 ore). Creare un'applicazione Java: realizzare l'interfaccia utente (6 ore). Gestire la sessione in Java (2 ore).

TESTI DI RIFERIMENTO

- Martin Fowler, "UML distilled. Guida rapida al linguaggio di modellazione standard", Quarta Edizione, Pearson, Maggioli Editore, 2010.
- Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Patterns: Elementi per il riuso di software a oggetti", Prima Edizione Italiana, Pearson, 2002.
- Il manuale di Java consigliato per il corso di Fondamenti di Informatica II.