

SCHEDA DELL'INSEGNAMENTO (SI)
"SOFTWARE ARCHITECTURE DESIGN"
SSD ING-INF/05

DENOMINAZIONE DEL CORSO DI STUDIO: INGEGNERIA INFORMATICA

ANNO ACCADEMICO 2023-2024

INFORMAZIONI GENERALI – DOCENTE

DOCENTE: ANNA RITA FASOLINO

TELEFONO: 081 7683906

EMAIL: ANNARITA.FASOLINO@UNINA.IT

INFORMAZIONI GENERALI – ATTIVITÀ

INSEGNAMENTO INTEGRATO (EVENTUALE): SOFTWARE ARCHITECTURE DESIGN

MODULO (EVENTUALE):

CANALE (EVENTUALE):

ANNO DI CORSO (I, II, III):

SEMESTRE (I, II): I

CFU: 6

INSEGNAMENTI PROPEDEUTICI (se previsti dall'Ordinamento del CdS)

Nessuno

EVENTUALI PREREQUISITI

Ingegneria del Software

OBIETTIVI FORMATIVI

L'insegnamento si propone di introdurre il tema delle architetture software e di fornire agli studenti conoscenze e competenze specialistiche relative alla progettazione, documentazione e sviluppo di tali architetture.

RISULTATI DI APPRENDIMENTO ATTESI (DESCRITTORI DI DUBLINO)

Conoscenza e capacità di comprensione

Lo studente deve dimostrare di conoscere e comprendere le problematiche relative alla disciplina delle architetture software ed il loro ruolo nell'ambito dei moderni processi di sviluppo software; il percorso formativo intende fornire agli studenti le conoscenze necessarie per comprendere le caratteristiche dei diversi modelli, stili e pattern architettonici e gli strumenti metodologici necessari per progettare, documentare, e sviluppare una architettura software. Tali conoscenze e strumenti consentiranno agli studenti di comprendere le principali relazioni che sussistono fra progetto architettonico e conseguenti caratteristiche di qualità del software e di cogliere le implicazioni delle diverse scelte di progetto sui requisiti posseduti dall'architettura risultante.

Capacità di applicare conoscenza e comprensione

Lo studente deve dimostrare di saper utilizzare le conoscenze e gli strumenti metodologici appresi per riuscire a progettare, documentare ed implementare architetture software in grado di realizzare tutti i requisiti funzionali e di qualità richiesti, implementando un processo di sviluppo ispirato ai moderni processi iterativi, evolutivi ed agili.

PROGRAMMA-SYLLABUS

Architetture Software: concetti fondamentali.

Definizioni di Architettura Software. Architettura come insieme di Strutture e di Viste: Strutture Modulari, Strutture dei Componenti e Connettori, Strutture di Allocazione. Ruolo ed Importanza dell'architettura nei processi di sviluppo software.

Qualità delle Architetture Software.

Attributi di qualità di una architettura osservabili a run-time (performance, security, availability, functionality, usability), Attributi non osservabili a run-time (modifiability, portability, reusability, integrability, e testability), attributi di qualità intrinseca dell'architettura (conceptual integrity, correctness, and completeness, buildability). Specificare gli attributi di qualità attraverso la tecnica degli Scenari di qualità. (CENNI)

Progettazione Architettonica.

Principi di progettazione architettonica, Progettazione basata sul riuso di esperienza, attraverso Pattern, Stili e Tattiche Architettoniche.

Pattern Architettonici.

I Pattern Architettonici State-logic-Display, Sense-Compute-Control e Model-View-Controller (MVC). Esempi.

Stili Architettonici

Stile procedurale e object-oriented, Stile a Livelli (Macchine Virtuali e Client-Server), Data-Flow (Batch e Pipe-and-Filter), a Memoria Condivisa (Blackboard e Rule Based), basato su Interprete (Codice Mobile), ad Invocazione Implicita (Publish-Subscribe e Event-Based), Stile ad Oggetti Distribuiti, Stile Service Oriented (SOA), Stile a Microservizi.

Tattiche per realizzare gli attributi di qualità di una architettura

Tattiche per l'Availability, Deployability, Modifiability, Energy Efficiency, Performance, Security, Safety, Testability, Usability.

Soluzioni Architetture

Accesso alle risorse di una architettura attraverso le Interfacce; progettazione delle Interfacce e Stili di interazione ; la rappresentazione e la struttura delle risorse scambiate (XML, JSON e Binary); Virtualizzazione ed il Cloud Computing; le applicazioni Mobili

Documentare le Architetture Software.

Documentare le diverse viste di una architettura software: stili di documentazione per la vista modulare, dei componenti e connettori, per la vista di distribuzione. Documentare il Contesto, il comportamento, le decisioni di progetto e le interfacce dei componenti di una architettura. Cenni allo Standard ISO/IEC 42010 (Systems and Software Engineering- Architecture Description).

Sviluppo di architetture Software con processi Iterativi ed Incrementali.

Lo sviluppo iterativo ed Incrementale: differenze rispetto allo sviluppo waterfall. Unified Process (UP): le Fasi di Ideazione, Elaborazione, Costruzione e Transizione. Le discipline e le pratiche di UP.

Sviluppo Agile. I Principi dello Sviluppo Agile. Manifesto Agile e pratiche Agili. Il Test Driven Development (TDD) e il Refactoring. Continuous Integration (CI). Il framework SCRUM: i ruoli in Scrum, i Backlog, le pratiche di Scrum.

Sviluppo Iterativo ed i Processi di Gestione della Configurazione del Software

La gestione delle Versioni del software. Modello di Version Control Centralizzato e Distribuito. Il processo di Build e l'Agile Building, Continuous Integration, DevOps. I processi di gestione delle Richieste di cambiamento del Software. Gestione delle Release.

MATERIALE DIDATTICO

- Len Bass, Paul Clements, Rick Kazman, Software Architecture in Practice, Fourth Edition, Addison Wesley, 2022
- P. Clements, F. Bachmann, Bass, etc. Documenting Software Architectures- Views and Beyond, Addison Wesley, Second Ed., 2010.
- N. Taylor, N. Medvidovic, E. Dashofy. Software Architecture -foundations, theory and practice, Wiley 2010.

Copia delle Slide proiettate durante le lezioni
Articoli scientifici e materiale integrativo dal Web

MODALITÀ DI SVOLGIMENTO DELL'INSEGNAMENTO

L'insegnamento verrà erogato prevalentemente attraverso lezioni frontali e sarà integrato da esercitazioni pratiche e da seminari di approfondimento. Le esercitazioni verteranno sullo studio di architetture software reali e l'applicazione dei metodi di progettazione e documentazione delle architetture presentati a lezione. Lo svolgimento delle esercitazioni potrà richiedere l'uso di ambienti integrati e software specialistico per la progettazione, la documentazione ed implementazione di architetture software.

Il docente utilizzerà:

- a) lezioni frontali per 30 ore (pari a circa il 60% delle ore totali),
- b) esercitazioni per approfondire praticamente aspetti teorici e applicare le conoscenze apprese per 14 ore (pari a circa il 30% delle ore totali),
- c) seminari specialistici per 4 ore

VERIFICA DI APPRENDIMENTO E CRITERI DI VALUTAZIONE

a) Modalità d'esame:

L'esame si articola in prova	
scritta e orale	
solo scritta	
Solo orale	X
discussione di elaborato progettuale	
altro	

In caso di prova scritta i quesiti sono (*)	A risposta multipla	
	A risposta libera	
	Esercizi numerici	

La prova orale finale tende ad accertare le conoscenze apprese nel campo delle architetture software, nonché dei moderni metodi e processi di progettazione e sviluppo di architetture software. La prova orale potrà prendere spunto da un elaborato di approfondimento svolto dagli studenti su uno dei temi trattati a lezione.

b) Modalità di valutazione: