



Principali informazioni sull'insegnamento

Denominazione dell'insegnamento	Ingegneria del Software	
Corso di studio	Informatica e Comunicazione Digitale	
Anno Accademico	2023/24	
Crediti formativi universitari (CFU) / European Credit Transfer and Accumulation System (ECTS)	9 CFU	
Settore Scientifico Disciplinare	INF/01 - Informatica	
Lingua di erogazione	Italiano	
Anno di corso	Secondo	
Periodo di erogazione	2° semestre, le date esatte sono riportate nel manifesto/regolamento	
Obbligo di frequenza	La frequenza è fortemente raccomandata	
Sito web del corso di studio	https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/informatica-icd-taranto-270/laurea-triennale-in-informatica-e-comunicazione-digitale-sede-di-taranto-d.m.-270	

Docente/i	
Nome e cognome	Antonio Piccinno
Indirizzo mail	antonio.piccinno@uniba.it
Telefono	080-5442535
Sede	Dipartimento di Informatica, Via Orabona 4, 70125, Bari. Stanza n.619, VI piano.
Sede virtuale	Piattaforma ADA - https://elearning.uniba.it/
Sito web del docente	ivu.di.uniba.it/people/piccinno.htm
Ricevimento (giorni, orari e modalità, es. su appuntamento)	(da confermare) mercoledì 12.30 - 13-30 (previo appuntamento)



Syllabus	
Obiettivi formativi	L'insegnamento di Ingegneria del Software riguarda l'analisi, il progetto e la realizzazione di sistemi software applicando i principi dell'Ingegneria del Software, nonché metodologie e tecniche di sviluppo di sistemi software. Ciò include la costruzione di una applicazione d'impresa, dalla raccolta dei requisiti alla realizzazione della stessa.
Prerequisiti	Lo studente deve avere familiarità con almeno un linguaggio di programmazione e con le strutture di dati fondamentali. Le seguenti conoscenze preliminari facilitano ed accelerano la comprensione degli argomenti dell'insegnamento: <ul style="list-style-type: none">• da Programmazione: Linguaggi imperativi, capacità di sviluppo di programmi in un linguaggio di programmazione (es. C);• da Linguaggi di Programmazione: comprensione della relazione tra problemi, algoritmi, linguaggi formali e linguaggi di programmazione; sintassi e semantica di un linguaggio di programmazione;• da Laboratorio di Informatica: Algoritmi fondamentali, Progettazione modulare
Contenuti di insegnamento (Programma)	Introduzione all'Ingegneria del Software (ore 9 + 1 esercitazione) <ul style="list-style-type: none">- Visione d'insieme- I tipi di prodotti software- Processi di sviluppo software- Qualità dei prodotti- Problemi dell'ingegneria del software Principi dell'Ingegneria del Software (ore 6) <ul style="list-style-type: none">- Applicabilità dei principi- Rigore e formalità, Separazione degli interessi, Modularità, Astrazione, Generalità, Incrementalità Analisi dei requisiti (ore 8 + 2 esercitazione) <ul style="list-style-type: none">- Concetti generali- Specifiche dei Requisiti- Specifiche Software Processi Agili (ore 7) <ul style="list-style-type: none">- Sviluppo Agile del Software- Metodologia SCRUM Progetto Software (ore 4 + 2 esercitazione) <ul style="list-style-type: none">- Concetti Generali- Elementi di Base sui Processi- Linee guida di progetto (Information Hiding)- Processo di progettazione SW Linguaggio di modellazione di un sistema software – UML (ore 10 + 3 esercitazione) <ul style="list-style-type: none">- Overview- Diagramma dei casi d'uso: casi d'uso, scenari, relazioni- Diagramma delle classi: classi, oggetti, relazioni- Diagramma di sequenza- Diagramma delle componenti- Diagramma di deployment- UML per il web: Web Application Extension



	<ul style="list-style-type: none">- Stereotipi UML: Approccio BCE (Boundary – Control -Entity)- Esempi di modellazione UML <p>Stili Architettureali (ore 4 + 1 esercitazione)</p> <ul style="list-style-type: none">- Principi Generali- Stili Architettureali- Object Oriented <p>Design Pattern (ore 3)</p> <ul style="list-style-type: none">- Pattern di creazione- Pattern strutturali- Pattern comportamentali- Pattern J2EE (cenni) <p>Strumenti di Supporto allo sviluppo (ore 2 + 1 esercitazione)</p> <ul style="list-style-type: none">- Application Lifecycle Management (ALM)- Configuration Management <p>Caso di studio (ore 3 + 5 esercitazione)</p> <ul style="list-style-type: none">- Introduzione caso di studio- Verifica dei requisiti/user stories caso di studio- Esempi e best practices- Progettazione caso di studio
<p>Testi di riferimento</p>	<ul style="list-style-type: none">● Carlo Ghezzi, Medhi Jazayeri, Dino Mandrioli “Ingegneria del Software - Fondamenti e Principi, 2a edizione” Pearson Prentice Hall, 2004.<ul style="list-style-type: none">○ Capitolo 1: Ingegneria del Software: visione d’insieme○ Capitolo 2: Il software: natura e qualità○ Capitolo 3: Principi dell’ingegneria del software● Ian Sommerville “Ingegneria del Software”, 10a ed. Pearson, 2017<ul style="list-style-type: none">○ Capitolo 2: Processi Software○ Capitolo 3: Sviluppo Agile del Software○ Capitolo 5: Modelli di sistema○ Capitolo 6: Progettazione architetturale○ Capitolo 7: Progettazione e implementazione● Martin Fowler “UML distilled. Guida rapida al linguaggio di modellazione standard” (4 ed.). Pearson Addison Wesley, 2010.<ul style="list-style-type: none">○ Capitolo 1: Introduzione○ Capitolo 3: Diagramma delle classi: concetti fondamentali○ Capitolo 4: Diagramma di sequenza○ Capitolo 5: Diagramma delle classi: concetti avanzati○ Capitolo 8: Diagramma di deployment○ Capitolo 9: Casi d’uso○ Capitolo 14: Diagramma delle componenti <p>Gli studenti che lo desiderano possono ottenere i testi in prestito dalla Biblioteca. Può convenire verificarne la disponibilità mediante il Sistema Bibliotecario di Ateneo https://opac.uniba.it/easyweb/w8018/index.php? e contattare la biblioteca per concordare il prestito.</p>
<p>Note ai testi di riferimento</p>	<p>I testi di riferimento sono integrati con slide, dispense del docente e altro materiale didattico messi a disposizione degli studenti sulla piattaforma di e-learning usata dal CdS.</p> <p>Testi consigliati per approfondire specifici argomenti:</p> <ul style="list-style-type: none">● Jim Arlow, Ila Neustadt “UML 2 e Unified Process – Analisi e progettazione Object-Oriented, 2a ed.”, McGraw-Hill, 2014 (per diagrammi UML)



	<ul style="list-style-type: none"> Steven John Metsker, "Design pattern in Java: manuale pratico". Pearson: Addison Wesley, 2003 (per pattern software) 		
Organizzazione della didattica			
Ore			
Totali	Didattica frontale	Esercitazione guidate + Progetto	Studio individuale
225 ore	56 ore	15+25 ore di laboratorio ed esercitazioni guidate	129 ore
CFU/ETCS			
9 CFU	7 CFU	1 + 1P CFU	

Metodi didattici	
	<p>Lezioni frontali con l'ausilio di slide che riportano esempi per illustrare gli argomenti trattati.</p> <p>Esercitazioni pratiche sull'utilizzo dei vari principi e tecniche presentate a lezione attraverso esercizi da svolgere singolarmente.</p> <p>Un progetto da svolgere preferibilmente in gruppo utilizzando la piattaforma online Redmine quale strumento di Application Lifecycle Management.</p> <p>Utilizzo della piattaforma di e-learning del Dipartimento di Informatica per la distribuzione del materiale e per le interazioni tra docenti e studenti durante e dopo il corso.</p>

Risultati di apprendimento previsti	
Conoscenza e capacità di comprensione	<ul style="list-style-type: none"> Il principale risultato di apprendimento previsto è la conoscenza relativa a principi, paradigmi, metodologie, tecniche e tecnologie fondamentali per l'analisi e progettazione in team di sistemi software di medie-grandi dimensioni supportati da strumenti allo stato della pratica. Tali conoscenze mirano anche a fornire allo studente le competenze necessarie nella produzione e manutenzione di software applicativo per le applicazioni d'impresa. <p>Lo studente acquisisce tale conoscenza sia attraverso le lezioni frontali e la partecipazione a seminari tematici erogati durante il corso, sia attraverso esercitazioni che gli consente di mettere in pratica e verificare quanto appreso, acquisendo così consapevolezza della capacità di comprensione e di come migliorare l'applicazione delle tecniche apprese.</p>
Conoscenza e capacità di comprensione applicate	<ul style="list-style-type: none"> Per consentire allo studente di applicare le conoscenze per lo sviluppo (produzione e manutenzione) delle Applicazioni d'Impresa, si svolgono in aula sia esercitazioni individuali che collettive.



	<ul style="list-style-type: none">• Allo studente è richiesto di sviluppare un progetto, nel quale è necessario applicare i principi di ingegneria del software, le metodologie e le tecniche presentate a lezione, selezionando quelle più adeguate allo specifico caso. La valutazione di tale progetto contribuisce alla valutazione finale dello studente e quindi al voto conseguito all'esame di profitto.
Competenze trasversali	<p>Autonomia di giudizio</p> <ul style="list-style-type: none">• Acquisire una significativa autonomia nell'operare le opportune scelte durante l'analisi, la progettazione e lo sviluppo del sistema software oggetto del progetto.• Acquisire la capacità di lavorare in team per lo sviluppo del sistema software e verificare i risultati ottenuti. Le esercitazioni che si svolgono durante il corso contribuiscono al raggiungimento di tali competenze grazie anche alla discussione di tali scelte con il docente.• L'autonomia di giudizio è parte della valutazione finale dello studente e tiene conto delle discussioni avvenute durante le lezioni, delle esercitazioni e della presentazione del progetto. <p>Abilità comunicative</p> <ul style="list-style-type: none">• Illustrare il risultato di esercizi svolti, autonomamente o in gruppo, con l'obiettivo di sviluppare le sue abilità comunicative.• La presentazione e discussione del progetto sviluppato in gruppo è parte della prova orale d'esame e consente allo studente di mostrare le proprie abilità comunicative. <p>Capacità di apprendere in modo autonomo</p> <ul style="list-style-type: none">• Per stimolare la capacità di apprendere in modo autonomo, allo studente è richiesto di approfondire specifici argomenti oppure è invitato a partecipare a seminari tenuti da altri docenti, interni o in visita al dipartimento, sui quali lo studente deve poi presentare durante le lezioni, e riportare in sede d'esame.

Valutazione	
Modalità di verifica dell'apprendimento	<p>La verifica dei risultati formativi raggiunti avviene durante l'esame finale, che prevede:</p> <ul style="list-style-type: none">• Un colloquio orale in cui si presenta e si discute il progetto sviluppato in gruppo e si verificano le competenze acquisite durante il corso e le capacità espositive dello studente.• Una prova scritta con domande a risposta multipla, chiuse e/o aperte. <p>Il risultato di ciascuna prova superata è valido per l'intero anno accademico in corso (8 appelli d'esame).</p> <p>Per gli studenti frequentanti sono previste le seguenti facilitazioni:</p> <ul style="list-style-type: none">• Bonus punteggio a valere sulla valutazione del progetto per gli studenti che svolgono positivamente le esercitazioni sul progetto/caso di studio.• N° 2 prove intermedie in itinere esoneranti la prova scritta.• Le facilitazioni per lo studente frequentante sono valide solo per la sessione estiva (primi tre appelli) dell'anno accademico corrente.
Criteri di valutazione	<ul style="list-style-type: none">• Conoscenza e capacità di comprensione<ul style="list-style-type: none">○ Lo studente dovrà essere in grado di applicare correttamente principi e metodologie per lo sviluppo del progetto, validare l'appropriatezza delle tecniche usate, produrre una documentazione chiara ed esaustiva.



	<ul style="list-style-type: none"> ● Conoscenza e capacità di comprensione applicate <ul style="list-style-type: none"> ○ Si valuta la presentazione del progetto per verificare le competenze acquisite dallo studente e la sua capacità di sintesi nonché la chiarezza di esposizione, la capacità di fare confronti significativi tra metodologie, tecniche e tecnologie diverse adottate e riportare un proprio giudizio critico. ● Autonomia di giudizio <ul style="list-style-type: none"> ○ Lo studente dovrà essere in grado di applicare opportune soluzioni per lo sviluppo del sistema software. ○ Si valuta la presentazione del progetto per verificare le competenze acquisite dallo studente e la sua capacità di sintesi nonché la chiarezza di esposizione, la capacità di fare confronti significativi tra metodologie, tecniche e tecnologie diverse adottate e riportare un proprio giudizio critico. ○ Si valuta la prova scritta con domande a risposta multipla, chiuse e/o aperte, per accertare le conoscenze di base dello studente. ○ Il voto del progetto e la sua presentazione concorrono al 70% del voto complessivo dell'esame e la prova scritta al rimanente 30%. ● Abilità comunicative <ul style="list-style-type: none"> ○ Lo studente dovrà essere in grado di produrre una documentazione chiara e contenente le informazioni necessarie per il sistema software sviluppato. ● Capacità di apprendere <ul style="list-style-type: none"> ○ Lo studente dovrà essere in grado di tradurre autonomamente il progetto in un sistema software. 																
<p>Criteria di misurazione dell'apprendimento e di attribuzione del voto finale</p>	<table border="1"> <thead> <tr> <th>Voto</th> <th>Descrittori</th> </tr> </thead> <tbody> <tr> <td>< 18 insufficiente</td> <td>Conoscenze frammentarie e superficiali dei contenuti, errori nell'applicare i concetti, descrizione carente.</td> </tr> <tr> <td>18 - 20</td> <td>Conoscenze dei contenuti sufficienti ma generali, descrizione semplice, incertezze nell'applicazione di concetti teorici.</td> </tr> <tr> <td>21 - 23</td> <td>Conoscenze dei contenuti appropriate ma non approfondite, capacità di applicare i concetti teorici, capacità di presentare i contenuti in modo semplice.</td> </tr> <tr> <td>24 - 25</td> <td>Conoscenze dei contenuti appropriate ed ampie, discreta capacità di applicazione delle conoscenze, capacità di presentare i contenuti in modo articolato.</td> </tr> <tr> <td>26 - 27</td> <td>Conoscenze dei contenuti precise e complete, buona capacità di applicare le conoscenze, capacità di analisi, descrizione chiara e corretta.</td> </tr> <tr> <td>28 - 29</td> <td>Conoscenze dei contenuti ampie, complete ed approfondite, buona applicazione dei contenuti, buona capacità di analisi e di sintesi, descrizione sicura e corretta.</td> </tr> <tr> <td>30 30 e lode</td> <td>Conoscenze dei contenuti molto ampie, complete ed approfondite, capacità ben consolidata di applicare i contenuti, ottima capacità di analisi, di sintesi e di collegamenti interdisciplinari, padronanza di descrizione.</td> </tr> </tbody> </table>	Voto	Descrittori	< 18 insufficiente	Conoscenze frammentarie e superficiali dei contenuti, errori nell'applicare i concetti, descrizione carente.	18 - 20	Conoscenze dei contenuti sufficienti ma generali, descrizione semplice, incertezze nell'applicazione di concetti teorici.	21 - 23	Conoscenze dei contenuti appropriate ma non approfondite, capacità di applicare i concetti teorici, capacità di presentare i contenuti in modo semplice.	24 - 25	Conoscenze dei contenuti appropriate ed ampie, discreta capacità di applicazione delle conoscenze, capacità di presentare i contenuti in modo articolato.	26 - 27	Conoscenze dei contenuti precise e complete, buona capacità di applicare le conoscenze, capacità di analisi, descrizione chiara e corretta.	28 - 29	Conoscenze dei contenuti ampie, complete ed approfondite, buona applicazione dei contenuti, buona capacità di analisi e di sintesi, descrizione sicura e corretta.	30 30 e lode	Conoscenze dei contenuti molto ampie, complete ed approfondite, capacità ben consolidata di applicare i contenuti, ottima capacità di analisi, di sintesi e di collegamenti interdisciplinari, padronanza di descrizione.
Voto	Descrittori																
< 18 insufficiente	Conoscenze frammentarie e superficiali dei contenuti, errori nell'applicare i concetti, descrizione carente.																
18 - 20	Conoscenze dei contenuti sufficienti ma generali, descrizione semplice, incertezze nell'applicazione di concetti teorici.																
21 - 23	Conoscenze dei contenuti appropriate ma non approfondite, capacità di applicare i concetti teorici, capacità di presentare i contenuti in modo semplice.																
24 - 25	Conoscenze dei contenuti appropriate ed ampie, discreta capacità di applicazione delle conoscenze, capacità di presentare i contenuti in modo articolato.																
26 - 27	Conoscenze dei contenuti precise e complete, buona capacità di applicare le conoscenze, capacità di analisi, descrizione chiara e corretta.																
28 - 29	Conoscenze dei contenuti ampie, complete ed approfondite, buona applicazione dei contenuti, buona capacità di analisi e di sintesi, descrizione sicura e corretta.																
30 30 e lode	Conoscenze dei contenuti molto ampie, complete ed approfondite, capacità ben consolidata di applicare i contenuti, ottima capacità di analisi, di sintesi e di collegamenti interdisciplinari, padronanza di descrizione.																
<p>Altro</p>	<p>Si suggerisce agli studenti di affidarsi esclusivamente alle informazioni/comunicazioni fornite sui siti ufficiali del Dipartimento di Informatica, ovvero sui gruppi social solo se costituiti e amministrati esclusivamente dai docenti dei relativi insegnamenti:</p> <ul style="list-style-type: none"> ● https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/corsi-di-laurea 																



- <https://www.uniba.it/it/ricerca/dipartimenti/informatica>
- <https://elearning.uniba.it/>

I programmi degli insegnamenti sono disponibili qui:

- <https://elearning.uniba.it/>

Le informazioni che tutti gli studenti dovrebbero conoscere sono scritte nei Regolamenti didattici e manifesti degli studi disponibili nel sito:

- <https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/corsi-di-laurea>

Si suggerisce agli studenti di diffidare delle informazioni e dei materiali circolanti su siti o gruppi social non ufficiali, poiché spesso sono risultati non affidabili, non corretti o incompleti. Per ogni dubbio, chiedere un incontro al docente secondo le modalità previste per il ricevimento.

Link al corso sulla piattaforma e-learning del Centro E-Learning di Ateneo:

- <https://elearning.uniba.it/course/view.php?id=2050>