



Ingegneria del Software

corso A-L

a.a. 2023-2024

Lunedì	10:10 - 13:30
Mercoledì	10:40 - 13:10

Aula A piano terra

Ricevimento: su appuntamento via email o MS Teams
filippo.lanubile@uniba.it

Prof. Filippo Lanubile



Ingegneria del Software

corso M-Z

a.a. 2023-2024

Martedì	9:20 - 12-40
Venerdì	9:20 - 11-50

Aula B piano terra

Ricevimento: su appuntamento via email o MS Teams
filippo.lanubile@uniba.it



Contenuti di insegnamento

Lezioni

- Introduzione
- Processi software e sviluppo agile
- Scrum
- Controllo di versione
- Automazione dello sviluppo e del rilascio
- Modellazione con UML
- Analisi dei requisiti
- System design
- OO design
- Testing
- Manutenzione ed Evoluzione

Esercitazioni

- CLI
- GitHub
- Git, GitHub Flow
- Gradle
- Docker
- Checkstyle, SpotBugs
- Strumenti UML
- JUnit



Testi di riferimento

Craig Larman. *Applicare UML e i pattern*. Pearson.

Altri libri consigliati (per approfondimenti):

- Bernd Bruegge, Dutoit Allen. *Object-Oriented Software Engineering Using UML, Patterns, and Java*. Pearson New Int. Ed. (Inglese)
- Titus Winters, Tom Manshreck, Hyrum Wright. *Software Engineering at Google*. O'Reilly Media. ISBN: 9781492082798. March 2020.
<https://abseil.io/resources/swe-book/html/toc.html>
- Martina Seidl, Marion Scholz, Christian Huemer, Gerti Kappel. *UML @ Classroom: An Introduction to Object-Oriented Modeling*, Springer.
- Martin Fowler. *UML Distilled*. Pearson.



Dispense e materiale didattico

Su piattaforma di e-learning:

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

Chiave di iscrizione: **IS-INF2324**



Registrazione su MS Teams

Codice del team: **qmaqyoj**



Esame senza esonero

- Prova di Laboratorio
 - Prove individuali in laboratorio:
git, GitHub, GitHub Flow, UML, testing
- Prova Orale
- Le due prove devono essere sostenute e superate nella stessa data di appello
- È necessario avere tutte le propedeuticità per poter registrare l'esame nella data di appello



Esame con esonero

- Esonero dalle due prove di esame
- Progetto assegnato dal docente:
 - Gruppi di 4-7 studenti
 - Da formare entro 2-3 settimane
- Non è necessario avere tutte le propedeuticità
 - Il risultato è conservato per l'intero anno accademico
 - Prenotazione su ESSE3 con nota «verbalizzazione esame»
- È consigliato
 - frequentare Metodi avanzati di programmazione
 - aver seguito Programmazione, Laboratorio di Informatica, Linguaggi di Programmazione

Calendario e voti dell'esame con esonero



	Inizio	Fine	Review/Feedback
Sprint 0	31 marzo o settimana 2-5 aprile	12 aprile	Settimana 15-19 aprile
Sprint 1	2-3 maggio	17 maggio	30-31 maggio
Sprint 2	Settimana 3-7 giugno	max 13 giugno	dal 13 al 27 giugno Pubblicazione voto gruppo: 27 giugno Verbalizzazione: 28 giugno

- Sprint 0 di familiarizzazione con l'ambiente di lavoro: **5 punti**
- Sprint 1 di sviluppo con relazione tecnica parziale: **10 punti**
- Sprint 2 di sviluppo con relazione tecnica finale: **15 punti**
- Voto individuale: **[-5, +5]**

Ambiente di lavoro per il progetto



Categoria	Scelta
Linguaggio di programmazione	Java
Modellazione UML	A scelta (es. GUI-based: StarUML; markup: Mermaid, PlantUML; web-based: draw.io, Visual Paradigm Online)
IDE	A scelta (es. VS Code, IntelliJ IDEA, NetBeans, Eclipse)
Gestione dipendenze e build automation	Gradle
Controllo di versione	Git
Cloud-based hosting	GitHub (Classroom)
Unit test automatizzato	JUnit
Analisi statica per QA	Checkstyle, SpotBugs
Continuous integration e deployment (CI/CD)	GitHub CI
Containerization	Docker
Coordinamento	Issue tracking e project board di GitHub
Comunicazione	Microsoft Teams



Progetto assegnato: Ataxx

<https://en.m.wikipedia.org/wiki/Ataxx>

- Videogioco di strategia del 1988
- Due giocatori competono su una griglia quadrata sette per sette
- Lo scopo del gioco è che un giocatore abbia la maggioranza dei pezzi sul tabellone alla fine del gioco, convertendo il maggior numero possibile di pezzi dell'avversario

I due giocatori si alterneranno su un unico device **Command Line Interface (CLI)**

