



# UNIVERSITÀ DI PAVIA

Anno Accademico 2021/2022

## PROGETTO DI SISTEMI DI TELEMEDICINA

<b>Anno immatricolazione</b>	2021/2022
<b>Anno offerta</b>	2021/2022
<b>Normativa</b>	DM270
<b>SSD</b>	ING-INF/06 (BIOINGEGNERIA ELETTRONICA E INFORMATICA)
<b>Dipartimento</b>	DIPARTIMENTO DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE
<b>Corso di studio</b>	BIOINGEGNERIA
<b>Curriculum</b>	Sanita' digitale
<b>Anno di corso</b>	1°
<b>Periodo didattico</b>	Secondo Semestre (07/03/2022 - 17/06/2022)
<b>Crediti</b>	9
<b>Ore</b>	66 ore di attività frontale
<b>Lingua insegnamento</b>	ITALIANO
<b>Tipo esame</b>	SCRITTO
<b>Docente</b>	LANZOLA GIORDANO - 9 CFU
<b>Prerequisiti</b>	<p>La condizione necessaria per seguire proficuamente il corso consiste nell'aver frequentato il modulo propedeutico "Fondamenti di Telemedicina" erogato nel semestre precedente. In particolare si richiede di avere acquisito le conoscenze di base impartite da tale corso e di averne sostenuto con successo la relativa prova di appello.</p> <p>Il corso "Progetto di Sistemi di Telemedicina" richiede infatti una solida conoscenza e una discreta pratica relativa ai concetti fondamentali connessi con la programmazione (i.e. variabili, espressioni, istruzioni, blocchi, funzioni/metodi, espressioni logiche e strutture di controllo) abbinate alla capacità di saper codificare algoritmi e realizzare applicazioni Object Oriented declinate nel contesto Java. Il corso ha una forte componente applicativa e fa uso dello stesso ambiente di sviluppo precedentemente introdotto durante il modulo "Fondamenti di Telemedicina".</p>

Si richiede infine una conoscenza di base relativa al funzionamento degli elaboratori (sistemi operativi, processi, gestione della memoria e memoria virtuale, canali di I/O, connettività di rete, etc...).

#### Obiettivi formativi

Il corso fornisce una panoramica generale su metodologie e tecniche per la realizzazione di sistemi distribuiti multi componente per la elaborazione delle informazioni, con particolare riguardo al contesto della Telemedicina.

Nell'ambito del corso si introduce un semplice ambiente che supporta lo scambio di messaggi asincroni fra sistemi distribuiti e si forniscono alcune applicazioni per illustrare come sia possibile ottenere l'integrazione di moduli in esecuzione su sistemi distinti. Si illustrano inoltre le tecnologie a supporto del lavoro di gruppo il cui uso e' da tempo entrato in ogni realta' professionale che si occupa di sviluppo del software.

Si richiede quindi agli Studenti che frequentano il corso di utilizzare queste metodologie e tecniche cimentandosi nello sviluppo di un prototipo di applicazione in linguaggio Java. Gli Studenti in previsione dell'appello dovranno formare dei gruppi di lavoro che adottano un approccio "agile" emulando quanto avviene correntemente in ambito professionale.

#### Programma e contenuti

Il corso, collocato al primo anno della Laurea Magistrale in Bioingegneria riunisce sapientemente aspetti metodologici e tecnologici. Esso prevede, sin dal suo inizio, una stretta alternanza fra lezioni in cui vengono esposti concetti metodologici (i.e. Object Orientation, Team Work, Software Design Patterns, Modellazione UML, Elaborazione Concorrente, Sistemi Distribuiti, Agenti, etc...) ed esercitazioni in cui lo Studente e' chiamato ad applicarli elaborando gli esempi proposti e sviluppando un proprio progetto.

##### Sistemi di Versioning e di Team Work

Si fornisce una generica introduzione ai sistemi di versioning e di supporto al lavoro di gruppo motivandone la necessita'.

Successivamente si illustra nel dettaglio il modello adottato da GIT che ormai rappresenta uno standard in ambito professionale attraverso diversi esempi di funzionamento svolti in classe. GIT viene utilizzato costantemente nel corso dell'intero semestre sia come sistema per la distribuzione degli esercizi e del codice su cui lavorare, sia come strumento per la condivisione del lavoro svolto all'interno dei gruppi.

##### Ambiente di Sviluppo

Si fornisce una illustrazione dell'ambiente software appositamente realizzato per il corso e che viene utilizzato dagli studenti per la realizzazione dei loro prototipi di Sistemi Distribuiti Multi Agente. Sono previsti anche alcuni richiami alla Programmazione ad Oggetti con particolare riferimento al linguaggio Java volti ad approfondire aspetti funzionali alla successiva realizzazione del progetto.

##### Programmazione Concorrente

Vengono illustrati i principi di Elaborazione Concorrente, riprendendo

quanto eventualmente già visto nel corso della parte propedeutica di "Fondamenti di Telemedicina". Viene introdotto il concetto di Processo e confrontato con quello di Thread. Infine vengono illustrate le primitive che devono essere utilizzate per coordinare l'esecuzione dei vari Thread all'interno di una applicazione con elaborazione concorrente realizzata in Linguaggio java.

#### Esercitazioni in Aula

Per ogni argomento trattato è prevista l'illustrazione di diversi esempi. In particolare si forniranno anche diverse applicazioni di sistemi con il duplice scopo di analizzarne il codice ed illustrare le funzionalità della libreria applicativa che consente di interagire con un broker per lo scambio dei messaggi di interazione. Tuttavia esse costituiranno anche la base per un richiamo all'uso del linguaggio di programmazione Java e ai principi di buona progettazione del software.

#### Metodi didattici

Il corso è organizzato secondo il seguente schema di massima:  
45 ore di lezione frontale (6 CFU)  
15 ore per esercitazioni in aula (1 CFU)  
45 ore per lo svolgimento del progetto in gruppo (2 CFU)

#### Testi di riferimento

Durante il corso viene fornito materiale didattico costituito da dispense, pubblicazioni open source e codice di esempio. Si segnalano i seguenti testi sia come riferimento che per ulteriori approfondimenti:

Cay Horstmann & Gary Cornell. Core Java, Volume 1 - Fundamentals (8th edition). Prentice Hall ISBN: 978-0132354769.

Cay Horstmann. Object Oriented Design and Patterns. John Wiley & Sons, Inc. ISBN: 978-0-471-74487-0 (450 Pg. 2006).

Martin Fowler. UML Distilled, Third Edition. Addison Wesley. ISBN: 0-321-19368-7 (192 Pg, Settembre 2003).

Craig Larman. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition). Prentice Hall. ISBN: 978-0131489066 (736 Pg).

#### Modalità verifica apprendimento

Gli Studenti formano dei gruppi per realizzare un progetto in linguaggio Java basato sulle metodologie e tecnologie Object Oriented apprese nel propedeutico modulo "Fondamenti di Telemedicina", integrato con gli aspetti relativi alla elaborazione concorrente e al Team work che sono oggetto del presente modulo "Progetto di Sistemi di Telemedicina".

Il gruppo deve fornire inizialmente le specifiche del proprio progetto descrivendole tramite una relazione che faccia uso di diagrammi UML. Successivamente i vari componenti del gruppo si suddividono i compiti di sviluppo e ciascuno realizza individualmente la propria parte che deve inserirsi nel disegno unitario del progetto fornito tramite il documento di specifiche.

In sede di valutazione finale viene analizzato il documento di specifiche prodotto, il codice realizzato e si chiede di illustrare il funzionamento del

sistema tramite una dimostrazione. Si deve quindi sostenere una prova pratica comparativa nel corso della quale a ciascuno Studente viene chiesto individualmente di apportare alcune modifiche di funzionalità su una copia del progetto.

#### Esito Conclusivo

Il voto viene assegnato al termine della prova pratica individuale in cui viene richiesto a ciascuno Studente di modificare alcune funzionalità del progetto realizzato in gruppo. Consiste in una valutazione complessiva dei risultati ottenuti nei due moduli Fondamenti di Telemedicina (6 CFU) e Progetto di Sistemi di Telemedicina (9 CFU) per un totale di 15 CFU.

#### Altre informazioni

#### Obiettivi Agenda 2030 per lo sviluppo sostenibile

[\\$lbl legenda sviluppo sostenibile](#)