



# UNIVERSITÀ DI PAVIA

Anno Accademico 2018/2019

## TELEMEDICINA

Anno immatricolazione	2018/2019
Anno offerta	2018/2019
Normativa	DM270
Dipartimento	DIPARTIMENTO DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE
Corso di studio	BIOINGEGNERIA
Curriculum	Tecnologie per la salute
Anno di corso	1°
Periodo didattico	Annualità Singola (01/10/2018 - 14/06/2019)
Crediti	12
Lingua insegnamento	Italiano

### L'insegnamento è suddiviso

503190 - **FONDAMENTI DI TELEMEDICINA**

503194 - **PROGETTO DI SISTEMI DI TELEMEDICINA**



# UNIVERSITÀ DI PAVIA

Anno Accademico 2018/2019

## FONDAMENTI DI TELEMEDICINA

<b>Anno immatricolazione</b>	2018/2019
<b>Anno offerta</b>	2018/2019
<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>	Tecnologie per la salute
<b>Anno di corso</b>	1°
<b>Periodo didattico</b>	Primo Semestre (01/10/2018 - 18/01/2019)
<b>Crediti</b>	6
<b>Ore</b>	70 ore di attività frontale
<b>Lingua insegnamento</b>	Italiano
<b>Tipo esame</b>	SCRITTO E ORALE CONGIUNTI
<b>Docente</b>	LARIZZA CRISTIANA (titolare) - 6 CFU
<b>Prerequisiti</b>	Conoscenze e Competenze di Programmazione di base acquisite col corso di Fondamenti di Informatica I.
<b>Obiettivi formativi</b>	<p>Il corso introduce il paradigma a oggetti e la modellazione concettuale che verranno introdotti come strumenti potenti in tutte le fasi dello sviluppo software, dall'analisi, alla progettazione fino all'implementazione di applicazioni di media complessità. Le lezioni si alternano allo svolgimento di esercizi e discussione di elementi di codice.</p> <p>L'obiettivo del corso è quello di fornire i principi e le conoscenze necessarie a maturare buone capacità di sviluppo di software ad oggetti.</p>
<b>Programma e contenuti</b>	<p>Il modulo tratta i seguenti argomenti:</p> <ul style="list-style-type: none"><li>- Programmazione Orientata agli Oggetti (OOP).</li><li>- Il Linguaggio Java.</li></ul>

	- Notazione UML.
<b>Metodi didattici</b>	Lezioni (ore/anno in aula): 35 Esercitazioni (ore/anno in aula): 0 Attività pratiche (ore/anno in aula): 35
<b>Testi di riferimento</b>	Il linguaggio trattato a lezione è Java. Per apprendere la sintassi del linguaggio è consigliato l'uso di un manuale e della documentazione disponibile online. Per il linguaggio Java si consiglia uno dei seguenti testi: - Walter Savitch. Programmazione di base e avanzata con Java. PEARSON - Arnold Ken, Gosling James, Holmes David. Il linguaggio Java. Manuale ufficiale. Pearson Education Italia e del testo - Craig Larman. Applicare UML e i pattern. PEARSON
<b>Modalità verifica apprendimento</b>	La valutazione del primo modulo si basa su due prove individuali. La prima è una prova di teoria che ha l'obiettivo di verificare le conoscenze teoriche di base del paradigma Object Oriented e del linguaggio Java. Si compone di 15 domande e viene superata se vengono fornite almeno 9 risposte corrette. La seconda, alla quale si accede solo dopo il superamento della prima, è una prova pratica di laboratorio individuale, nella quale viene richiesto di risolvere un problema di media complessità. La soluzione consiste nello sviluppo di una applicazione in linguaggio Java in grado di soddisfare i requisiti indicati. La valutazione della prova pratica tiene conto sia della capacità di utilizzare in modo adeguato il linguaggio Java, che della qualità del modello concettuale utilizzato nella progettazione e realizzazione dell'applicazione. La valutazione finale è ottenuta come media pesata della prova di teoria (1/3) e della prova pratica (2/3).
<b>Altre informazioni</b>	
<b>Obiettivi Agenda 2030 per lo sviluppo sostenibile</b>	<a href="#">\$Ibl_legenda_sviluppo_sostenibile</a>



### PROGETTO DI SISTEMI DI TELEMEDICINA

<b>Anno immatricolazione</b>	2018/2019
<b>Anno offerta</b>	2018/2019
<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>	Tecnologie per la salute
<b>Anno di corso</b>	1°
<b>Periodo didattico</b>	Secondo Semestre (06/03/2019 - 14/06/2019)
<b>Crediti</b>	6
<b>Ore</b>	52 ore di attività frontale
<b>Lingua insegnamento</b>	
<b>Tipo esame</b>	SCRITTO E ORALE CONGIUNTI
<b>Docente</b>	LANZOLA GIORDANO (titolare) - 6 CFU
<b>Prerequisiti</b>	<p>Il corso richiede una conoscenza elementare (ma solida) relativa ai concetti fondamentali connessi con la programmazione dei computer (i.e. variabili, istruzioni, funzioni/metodi e strutture di controllo) abbinata alla capacità di sapere codificare semplici algoritmi. Ulteriori requisiti riguardano la conoscenza delle metodologie e tecnologie per l'analisi e la progettazione di database relazionali e la capacità di sapere scrivere correntemente semplici interrogazioni in linguaggio SQL. Tutti questi prerequisiti vengono acquisiti attraverso moduli precedenti. Completano il profilo dello Studente una conoscenza basilare relativa all'uso del PC e della navigazione su Web, oltre a quel mix di capacità progettuale, abilità logica, acume e senso critico che dovrebbero peraltro rappresentare un requisito essenziale per potersi collocare proficuamente in una Scuola di Ingegneria.</p>
<b>Obiettivi formativi</b>	Il corso fornisce un quadro generale su metodologie, tecnologie ed

architetture per la realizzazione di sistemi distribuiti per l'elaborazione delle informazioni, con particolare riguardo all'ambito della Telemedicina.

#### B) Progetto di Sistemi di Telemedicina:

Prevede la presentazione di un ambiente per lo sviluppo di applicazioni distribuite e si forniscono piccole applicazioni come base per illustrare l'integrazione di moduli distinti. Esso richiede altresì agli Studenti di sviluppare di un prototipo di applicazione da realizzare in un piccolo gruppo di lavoro.

### 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, Design Patterns, UML Modeling, Elaborazione Concorrente e Sistemi Distribuiti, Agenti, etc) ed esercitazioni in cui lo Studente è chiamato ad applicarli elaborando gli esempi proposti e sviluppando un proprio progetto.

#### Parte B: Sistemi di Versioning e Team Work

Si fornirà una introduzione generica ai sistemi di versioning, e di supporto al lavoro di gruppo. Successivamente verrà illustrato nel dettaglio GIT attraverso esempi di funzionamento svolti in classe. GIT verrà utilizzato costantemente nel corso del semestre sia come sistema per la distribuzione degli elementi di codice su cui lavorare, sia come strumento a supporto del lavoro svolto all'interno dei singoli gruppi.

#### Parte B: Ambiente di Sviluppo

Si fornirà una illustrazione dell'ambiente software appositamente realizzato per il corso e che verrà utilizzato dagli Studenti per sviluppare i loro prototipi di Sistemi Distribuiti Multi Agente. Sono altresì previsti alcuni richiami alla programmazione ad oggetti con particolare riferimento al Linguaggio Java volti ad approfondire aspetti funzionali alla successiva realizzazione del progetto.

#### Parte B: Programmazione Concorrente

Verranno illustrati i principi di Programmazione Concorrente, riprendendo quanto eventualmente già visto nella parte A). Verrà introdotto il concetto di Processo e confrontato con quello di Thread. Infine verranno illustrate le primitive che devono essere utilizzate per coordinare l'esecuzione dei vari Thread presenti all'interno di un sistema con elaborazione concorrente.

#### Parte B: Esercitazioni in Aula

E' previsto l'uso di alcune applicazioni di esempio con il duplice scopo di analizzarne il codice e illustrare le funzionalità della libreria applicativa che consente di interagire con il 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 progettazione del software.

### Metodi didattici

Il corso prevede (Mod A + Mod B):  
40(A) + 15(B) ore di lezione frontale (5.3(A) + 2(B) CFU)

10(A) + 30(B) ore di esercitazione assistita (0.7(A) + 2(B) CFU)  
45(B) ore per lo svolgimento di un progetto (2 CFU Solo Mod. B)

#### Testi di riferimento

Durante il corso viene fornito materiale didattico costituito da dispense e codice. E' tuttavia opportuno acquisire alcuni dei seguenti testi sia come riferimento che per ulteriori approfondimenti.

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

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

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

James Gosling, Bill Joy, Guy Steele, Gilad Bracha. The Java Language Specification (Third Edition). Addison Wesley. ISBN 0321246780 (688 Pagine, Giugno 2005).

#### Modalità verifica apprendimento

Parte B: "Progetto di Sistemi di Telemedicina"

Gli Studenti formano gruppi per realizzare un progetto che sfrutti metodologie e tecnologie apprese nella parte A integrate con quanto previsto dalla parte B. Le specifiche sono fornite dall'intero gruppo tramite una relazione ed utilizzando diagrammi UML. Il lavoro viene quindi diviso tra i componenti e ciascuno sviluppa la propria parte. In sede di valutazione puo' essere richiesto di apportare modifiche al progetto.

Esito

Il voto viene assegnato al termine della prova a progetto e consiste in una valutazione complessiva dei risultati ottenuti dal candidato su tutte le prove previste dai due moduli.

#### Altre informazioni

Parte B: "Progetto di Sistemi di Telemedicina"

Gli Studenti formano gruppi per realizzare un progetto che sfrutti metodologie e tecnologie apprese nella parte A integrate con quanto previsto dalla parte B. Le specifiche sono fornite dall'intero gruppo tramite una relazione ed utilizzando diagrammi UML. Il lavoro viene quindi diviso tra i componenti e ciascuno sviluppa la propria parte. In sede di valutazione puo' essere richiesto di apportare modifiche al progetto.

Esito

Il voto viene assegnato al termine della prova a progetto e consiste in una valutazione complessiva dei risultati ottenuti dal candidato su tutte le prove previste dai due moduli.

#### Obiettivi Agenda 2030 per lo

