



DESIGN OF TELEMEDICINE SYSTEMS

Enrollment year	2021/2022
Academic year	2021/2022
Regulations	DM270
Academic discipline	ING-INF/06 (ELECTRONIC AND INFORMATION BIOENGINEERING)
Department	DEPARTMENT OF ELECTRICAL, COMPUTER AND BIOMEDICAL ENGINEERING
Course	BIOENGINEERING
Curriculum	Sanita' digitale
Year of study	1°
Period	2nd semester (07/03/2022 - 17/06/2022)
ECTS	9
Lesson hours	66 lesson hours
Language	Italian
Activity type	WRITTEN TEST
Teacher	LANZOLA GIORDANO - 9 ECTS
Prerequisites	In order to proficiently attend this class the Student should have already attended the preliminary class "Foundations of Telemedicine" administered during the first semester of the Academic Year. More specifically the Student must have acquired all the basic skills taught by that class and should also have successfully passed the final assessment test. The class "Design of Telemedicine Systems" requires a basic and solid background knowledge concerning the fundamental concepts of computer programming (i.e. variables, expressions, statements, blocks, functions/methods, logic expressions and control structures). The Student's should also have developed a strong ability to code simple algorithms and develop applications according to the Object Oriented paradigm in the Java Language. This class has a strong operational commitment and makes use of the same software development

environment introduced during the preliminary module "Foundations of Telemedicine".

The Student should also have a basic knowledge concerning computer architecture and internals (operative system, processes, memory management and virtual memory, I/O channels, networking basics, etc...).

Learning outcomes

The course provides a basic introduction to architectures, methodologies and technologies for the development of applications that process information adopting a distributed model particularly focused on the Telemedicine context.

During the semester a simple environment supporting the exchange of asynchronous messages among distributed systems will be introduced. Several applications will be also presented as examples of software modeling and for illustrating how to properly exploit message queues to send/receive messages between systems.

The current methodologies and tools supporting team work will also be introduced considering that since long they have been adopted by every company concerned with software development and represent a requirement in that context.

The course also requires that Students team up in groups collaborating to the design and development of an application prototype, adopting an "agile" approach emulating the activities of a small business unit of a company.

Course contents

The course scheduled in the second semester of the first year of the Master Degree in Bioengineering combines architectural, methodological and technological topics. The lectures will first introduce new methodological issues of interest for the implementation of telemedicine systems (i.e. Object Orientation, Team Work, Software Design Patterns, UML Modeling, Concurrent Programming, Distributed Systems, Agents, etc...). Those will be better presented through short coding sessions where code chunks will be delivered to the class.

Versioning Systems and Team Work

Lectures will provide a short introduction to the purpose and functionality of Versioning Systems and the support they provide for Team Work. Then GIT will be analyzed to a greater extent due to its widespread diffusion and because it will be used as the tool for distributing code chunks and projects to the class. GIT will also be used throughout the semester as a means of supporting Team Work among the Students participating to the development of the group project for the final course assessment.

Developing Environment

Some lectures will be devoted to the illustration of the development environment supporting message passing among the Multi Agent System prototypes to be developed by the Students. Some insights on Object Oriented Design and Programming will also be provided with

specific reference to the Java Language that will be exploited for the group projects.

Concurrent Programming

A key issue of almost every ICT project involves Concurrent Programming. Processes and Threads will be introduced and analyzed with several coding examples in the Java context. Those lectures will also encompass a discussion of the basic primitives provided by the Java Language for controlling and coordinating code execution in a multi threaded application.

Class Work

Throughout the semester a lot of projects will be illustrated through an internal GIT server hosted by the laboratory with a twofold aim of learning how to implement and master distributed programming techniques and better understand how the broker platform that will be used for message passing actually works.

Teaching methods

The class is shaped according to the following format:

Lectures: 45 hours (6 ECTS)

Practical Class: 15 hours (1 ECTS)

Project: 45 hours (2 ECTS)

Reccomended or required readings

The course will provide slides, lecture notes, open source textbooks and all the code chunks illustrated in the examples considered during the semester. However to increment the level of awareness and expertise of the Students the following additional references are suggested:

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).

Assessment methods

Students will team up in groups during the course semester for designing and implementing a project. Projects will exploit methodologies and technologies already learned during the preliminary course "Foundations of Telemedicine" mostly concerning Object Orientation basic, and will be complemented with the specific topics delivered in this course addressing Team Work and Concurrent Programming.

Teams will first prepare a written specification illustrating the features of their project using both narratives and UML diagrams. Then they will partition the work among all group members and separately proceed to

the actual implementation of the project that must adhere to the original design provided through the specification.

The final assessment will consider both the written specification document and the code produced. Each group will then be asked to illustrate the functionality of the system produced during a joint demonstration session. Finally all the members of each group will undergo a final test during which they will be separately asked to provide a set of minor functional changes to their project to assess the level of proficiency reached.

Final Outcome

The final outcome of the course will take into account both the score achieved in the class "Foundations of Telemedicine" (6 ECTS) and the present one "Design of Telemedicine Systems" (9 ECTS) for a total of 15 ECTS.

Further information

Sustainable development goals - Agenda 2030

[\\$lbl_legenda_sviluppo_sostenibile](#)