IMMUNE TECHNOLOGY INSTITUTE

COMPUTER ENTREPRENEURSHIP BACHELOR

BLOCK NAME	**BIG DATA**
BLOCK CODE	**CS-L4B4**

COURSE	**2**
LEVEL	**4**
CREDITS	**5**
CLASS HOURS	**50**
HOMEWORK	**75**
TOTAL HOURS	**125**

DESCRIPTION

This block introduces the fundamentals of Big Data and its ecosystem.
We will face the challenge of using Apache Hadoop and Apache Spark to gather and show some KPIs for a hypothetical management team of a company. This company will have a huge customer database with information coming from a number of heterogeneous sources (so we will also need to perform ETL actions).

PRE-REQUISITES

Basic programming and advanced database skills are needed.
CS-L1B1, CS-L2B4, CS-L4B3

OBJECTIVES

The goal is for students to be familiar with the most commonly used Big Data tools and technologies.

SKILLS TO BE DEVELOPED

1 - Big Data fundamentals.
1.1 - Understand the fundamentals of Big Data technologies.
1.2 - Be able to identify when Big Data technologies are needed to solve a specific problem.
2 - Apache Hadoop.
2.1 - Know the basic components of Apache Hadoop and how they interact.
3 - Extract, transform, load.
3.1 - Understand the need for ETL actions when working with data from external sources.
3.2 - Be able to perform ETL actions.
4 - Map-Reduce.
4.1 - Understand how the Map-Reduce method works.
4.2 - Create simple Map-Reduce programs.
5 - Batch vs Streaming.
5.1 - Be able to differentiate between batch and streaming approaches and to identify which kind of problems each approach is appropriate for.
6 - Apache Spark.
6.1 - Know the fundamentals of Apache Spark.
7 - Big Data ecosystem.
7.1 - Identify different members of the Big Data ecosystem.

SYLLABUS

1 - Big Data fundamentals.
2 - Apache Hadoop.
3 - Extract, transform, load.
4 - Map-Reduce.
5 - Batch vs Streaming.
6 - Apache Spark.
7 - Big Data ecosystem.

METHODOLOGY

Resolution of practical activities supervised by the mentor. Compulsory attendance.

DEDICATION AND EVALUATION

The student must pass the mandatory activities (challenges/projects) that are covered in the block.
Each challenge/project produces its own score and has been designed to cover certain block percentages.
Such score is 80% objective (the program that solves the challenge/project works without errors and producing the expected results) and 20% subjective (solution elegance, how clean the code is, documentation).
Block scores are finally calculated by prorating individual activities with respect to their block coverage percentages.