GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

Subject name and code	Big Data, PG_00068194		
Field of study	Automatic Control, Cybernetics and Robotics		
Date of commencement of studies	October 2025	Academic year of realisation of subject	2028/2029
Education level	first-cycle studies	Subject group	Optional subject group Subject group related to scientific research in the field of study
Mode of study	Full-time studies	Mode of delivery	at the university
Year of study	4	Language of instruction	Polish
Semester of study	7	ECTS credits	3.0
Learning profile	general academic profile	Assessment form	assessment
Conducting unit	Department of Decision Systems and Robotics -> Faculty of Electronics Telecommunications and Informatics -> Faculties of Gdańsk University of Technology		
Name and surname of lecturer (lecturers)	Subject supervisor	dr inż. Jakub Wszołek	
	Teachers	dr inż. Jakub Wszołek	

Lesson types

Lesson type	Lecture	Tutorial	Laboratory	Project	Seminar	SUM
Number of study hours	15.0	0.0	15.0	15.0	0.0	45

E-learning hours included: 0.0

Learning activity and number of study hours

Learning activity	Participation in didactic classes included in study plan	Participation in consultation hours	Self-study	SUM
Number of study hours	45	3.0	27.0	75

Subject objectives

The course introduces students to the fundamental concepts, architectures, and tools used for processing, storing, and analyzing large-scale distributed datasets. The course covers modern data processing paradigms, including batch and streaming architectures, and provides an in-depth overview of platforms such as Apache Spark, Hadoop, and Lakehouse systems. Students gain practical experience in designing scalable data pipelines, working with columnar data formats, and implementing analytical workflows in distributed environments. The course combines theoretical foundations with hands-on labs, preparing students to build and optimize data-intensive applications used in contemporary enterprise and cloud ecosystems.

Learning outcomes

Course outcome	Subject outcome	Method of verification
[K6_U08] while identifying and formulating specifications of engineering tasks related to the field of study and solving these tasks, can:n- apply analytical, simulation and experimental methods,n- notice their systemic and non-technical aspects,n- make a preliminary economic assessment of suggested solutions and engineering work n	The student is able to apply analytical methods and Big Data tools to formulate and solve engineering problems, and evaluate the effectiveness and limitations of proposed solutions.	[SU4] Assessment of ability to use methods and tools
[K6_U07] can apply methods of process and function support, specific to the field of study	The student is able to use Big Data tools and platforms to implement analytical workflows, streaming data processing, and data pipelines, in accordance with the specifics of the study program.	[SU3] Assessment of ability to use knowledge gained from the subject
[K6_W21] knows and understands the basic methods of decision making as well as methods and techniques of design and operation of automatic regulation and control systems, computer applications for controlling and monitoring dynamic systems.	The student knows and understands the fundamental methods for designing and implementing large-scale data processing systems, including distributed system architectures, parallel computation models, and ETL/ELT processing techniques.	[SW3] Assessment of knowledge contained in written work and projects

Subject contents

Course content – lecture

1. Introduction to Big Data definitions, characteristics of large datasets, the 5V model.
2. Distributed systems architecture computation models, scalability, fault tolerance.
3. The Hadoop ecosystem HDFS, MapReduce, YARN.
4. Apache Spark architecture, execution model, RDD vs DataFrame API.
5. Spark SQL query optimization, Catalyst optimizer, Tungsten engine.
6. Data formats in Big Data systems Parquet, ORC, Avro; compression and columnar storage.
7. Delta Lake ACID transactions, time travel, metadata and table management.
8. Stream processing Spark Structured Streaming, sources, sinks, windowing.
9. Apache Kafka architecture, partitioning, message queues, integration with Spark.
10. Designing and building ETL/ELT data pipelines.
11. Data architectures: Data Lake, Data Warehouse, Lakehouse, introduction to Data Mesh.
12. Data security and governance access control, auditing, data quality, lineage.
13. Scaling and optimization of Big Data systems partitioning, caching, shuffle management.
14. Modern data platforms (cloud and on-premise) overview and comparison.

Course content – laboratory

1. Setting up the development environment local or cluster-based Spark configuration.
2. Loading and exploring large datasets (CSV, JSON, Parquet).
3. Working with RDDs, DataFrames, and Spark SQL.
4. Data transformations, aggregations, and window functions.
5. Query optimization and execution plan analysis (explain).
6. Implementing ETL/ELT pipelines in Apache Spark.
7. Working with columnar data formats and Delta Lake table creation, ACID operations.
8. Stream processing basics building a simple streaming pipeline.
9. Integrating Spark Structured Streaming with Apache Kafka.
10. Data quality validation and pipeline monitoring.

Course content – project

1. Designing a Big Data processing system for a selected problem (requirements + architecture).
2. Implementing an ETL/ELT pipeline using a chosen Big Data platform (e.g., Spark + Delta Lake).
3. Preparing and configuring the project environment (local or cloud-based).
4. Integrating data from multiple sources (batch and/or streaming).
5. Implementing analytical logic aggregations, reports, statistical models or ML (optional).
6. Preparing documentation of the architecture, metadata, processing steps, and results.
7. Final presentation and defense of the project.

Prerequisites and co-requisites

The student should have basic knowledge of programming (preferably Python or Java), fundamentals of algorithms, and an understanding of operating systems and computer networks. Basic familiarity with databases (SQL) and the ability to work in a Linux environment are recommended. Prior experience with data analysis or working with larger datasets is considered an advantage.

Assessment methods and criteria

Subject passing criteria	Passing threshold	Percentage of the final grade
Evaluation of a group project involving the design and implementation of a Big Data pipeline (ETL/ELT), data integration, and presentation of results.	50.0%	30.0%
Assessment of laboratory assignments involving Spark programming, data analysis, columnar formats, and streaming data processing.	50.0%	30.0%
Written exam assessing theoretical knowledge of Big Data architectures, distributed processing, and tools (Hadoop, Spark, Delta Lake).	50.0%	40.0%

Recommended reading

Basic literature

1. Tom White, Hadoop: Definitywne wprowadzenie (Hadoop: The Definitive Guide), OReilly.
2. Jules S. Damji, Brooke Wenig, Tathagata Das, Denny Lee, Spark: Definitywne wprowadzenie (Spark: The Definitive Guide), OReilly.
3. Bill Chambers, Matei Zaharia, Learning Spark: Lightning-Fast Big Data Analysis, OReilly.
4. Dokumentacja projektów Apache Spark, Hadoop, Delta Lake (materiały online).

Supplementary literature

1. Martin Kleppmann, Designing Data-Intensive Applications, OReilly.
2. Tyler Akidau, Slava Chernyak, Streaming Systems: The What, Where, When, and How of Large-Scale Data Processing, OReilly.
3. Ben Stopford, Designing Event-Driven Systems, OReilly.
4. Andrew Psaltis, Streaming Data: Understanding the Real-Time Pipeline, Manning.
5. Public datasets (e.g., Kaggle, UCI Machine Learning Repository) recommended for projects.
6. Official documentation: Apache Kafka, Apache Airflow, Delta Lake, Spark Structured Streaming.

eResources addresses

Example issues/ example questions/ tasks being completed	• Describe the architecture of Apache Spark and explain the roles of the driver, executors, and cluster manager. • Explain the differences between RDD, DataFrame, and Dataset in Spark. • Compare Parquet, ORC, and Avro data formats. When is each most effective? • Explain how Apache Kafka works, including partitions, offsets, and message flow. • Describe the Spark SQL optimization process (Catalyst optimizer, Tungsten engine). • Implement an ETL pipeline using the Spark DataFrame API. • Perform analytics on a large dataset using window functions. • Build a simple streaming application using Spark Structured Streaming and Kafka. • Perform ACID operations in Delta Lake and demonstrate the use of time travel. • Design a Big Data system architecture for a provided business scenario.
Practical activites within the subject	Not applicable

Document generated electronically. Does not require a seal or signature.