National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute"

Computer Systems Software Department

BIG DATA TECHNOLOGIES

Syllabus

Requisites of the Course

Cycle of Higher Education	***First cycle of higher education (Bachelor's degree)***
Field of Study	*F Information Technologies*
Speciality	*F2 Software engineering*
Education Program	*Software Engineering of Multimedia and Information Retrieval Systems*
Type of Course	*Selective*
Mode of Studies	*full-time*
Year of studies, semester	*3 year (5 semester)*
ECTS workload	*4 credits (ECTS). Time allotment - 120 hours, including lectures: 36 hours, practicalwork: 18 hours, independent work: 66 hours.*
Testing and assessment	*Final test, modular test , calendar control*
Course Schedule	*Classes by the timetable http://roz.kpi.ua/*
Language of Instruction	*English*
Course Instructors	*Lecturer: PhD, Associate Professor, Liubov Oleshchenko, oleshchenkoliubov@gmail.com* *Practical work: PhD, Associate Professor, Liubov Oleshchenko, oleshchenkoliubov@gmail.com*
Access to the course	*Google classroom: Access is given to registered students.*

Outline of the Course

1. Course description, goals, objectives, and learning outcomes

The study of the discipline "Big Data Technologies" allows students to form the competencies necessary to solve practical problems of professional and scientific activities related to the preparation and analysis of big data.

*The **purpose** of studying the discipline "Big Data Technologies" is to form students' ability to use software methods and data processing tools for big data analysis and management decisions in various fields of science and business.*

*The **subject** of the discipline "Big Data Technologies" is the general principles and approaches to big data processing, capabilities and technologies of Python and R programming languages for processing, analysis and visualization of big data.*

*The study of the discipline "Fundamentals of computer systems and networks" contributes to the formation of the following **professional competence (PC)** in students according to the educational program:*

***PC8** Ability to apply fundamental and interdisciplinary knowledge to successfully solve software engineering problems.*

*The study of the discipline "Fundamentals of computer systems and networks" contributes to the formation of the following **program learning outcomes (PLO)** for students according to the educational program:*

PLO01 *To analyze, purposefully search and select the necessary information and reference resources and knowledge to solve professional problems, taking into account modern advances in science and technology.*

PLO07 *To know and to apply in practice the fundamental concepts, paradigms and basic principles of the functioning of language, instrumental and computational tools of software engineering.*

PLO18 *To know and be able to apply information technology of processing, storage and transmission of data.*

2. Prerequisites and post-requisites of the course

(the place of the course in the scheme of studies in accordance with curriculum)

Successful study of the discipline "Big Data Technologies" is preceded by the study of disciplines "Programming" and "Databases" of the curriculum for bachelors in the specialty F2 "Software Engineering". The theoretical knowledge and practical skills obtained during the mastering of the discipline "Big Data Technologies" are necessary for studying the discipline "Network Software Design and Development" of the curriculum for master's degree in specialty F2 "Software Engineering".

To successfully master the discipline requires a basic level of English not less than A2.

3. Content of the course

Discipline "Big Data Technologies" involves the study of the following topics:

Topic 1. Sources and types of big data. Preparation and data analysis in Python.

Topic 2. Programming language R capabilities. Architectural models of Big Data.

Modular test.

Final test.

4. Coursebooks and teaching resources

Basic references:

1. *Oleshchenko L.M. Big Data Technologies: Lecture Notes [Electronic resource]: tutorial is aimed at students of the speciality F2 "Software Engineering" (educational program «Software Engineering of Multimedia and Information Retrieval Systems») / Igor Sikorsky Kyiv Polytechnic Institute; Oleshchenko Liubov M. – Electronic text data. – Kyiv: Igor Sikorsky Kyiv Polytechnic Institute, 2025. – 225 p. https://ela.kpi.ua/handle/123456789/77498*

Additional references:

1. Big Data and Big Data Analytics: Concepts, Types and Technologies / https://www.researchgate.net/publication/328783489_Big_Data_and_Big_Data_Analytics_Concepts_Types_and_Technologies/related

2. Understanding Big Data. Analytics for Enterprise Class. Hadoop and Streaming Data / https://www.immagic.com/eLibrary/ARCHIVES/EBOOKS/I111025E.pdf?__cf_chl_jschl_tk__=pmd_224e99ec9955bf0d1a8e4ab864b26016568b9271-1628851175-0-gqNtZGzNAfijcnBszQdi

3. Programming with Databases – Python / https://swcarpentry.github.io/sql-novice-survey/10-prog/index.html

4. Pandas / https://www.w3schools.com/python/pandas/pandas_intro.asp

5. Plotly Python Open Source Graphing Library / https://plot.ly/python/

6. Getting Started with Plotly in Python / https://plot.ly/python/getting-started/

7. Microsoft R Application Network / https://mran.microsoft.com/documents/what-is-rDecision Tree / https://www.geeksforgeeks.org/decision-tree/

8. Apache Hadoop / http://hadoop.apache.org/

9. HDFS / https://www.ibm.com/analytics/hadoop/hdfs

10. About the Cassandra File System (CFS) – deprecated / https://docs.datastax.com/en/dse/5.1/dse-dev/datastax_enterprise/ analytics/ cfsAbout.html

11. Extract, transform, and load (ETL) / https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl

Educational content

5. Methodology

№	Type of training session	Description of the training session	Number of hours
Topic 1. Sources and types of big data. Data preparation and analysis in Python.			
1	*Lecture 1. Sources of big data. Internet of Things. Definition of Big Data.*	*Internet of Things and the growth of data. Kaggle Platform. DrivenData. Big Data Definition. Real-world examples of big data. Open data. Data privacy. Structured and unstructured data. Cloud and fog computing. Data at rest and data in motion. Big data infrastructure. Distributed data and its processing.*	*2*
2	*Practical work 1. Researching open data sources. Loading a dataset and saving data in csv format. Working in the Google Colab cloud environment.*	*Lesson 1. Researching open data sources Task: to get acquainted with the concept of open data and the main platforms that provide access to them (Kaggle, Data.gov, Google Dataset Search, Unified Open Data Portal of Ukraine). Find a set of open data on the selected topic (ecology, transport, education, healthcare, etc.). Preview and analyze the structure of the selected dataset: determine the format, size, number of variables, data types and content of each column. Assess the suitability of the dataset for further analytical processing.*	*2*
		Lesson 2. Loading, processing, and saving data in Google Colab Task: learn how to work with the Google Colab cloud environment to perform analytical tasks in Python. Import the necessary libraries (pandas, numpy). Download the selected data set from your local computer, Google Drive, or via the link. Check the correctness of the download: view the first rows of data, table dimensions, and the presence of gaps. Save the processed data set in CSV format in your own environment or on Google Drive. Draw conclusions about the advantages of using Google Colab for processing and analyzing big data.	*2*

3	*Lecture 2. Developing software for analyzing websites that provide open data using Python Pandas. Open data, its formats and processing tools.*	*Capabilities of data analysis tools. The role of Python in data analysis. Traditional big data analytics and next-generation analytics.* *Data analysis life cycle. Open data, its formats and processing tools. Web scraping. Data extraction, transformation and loading.*	*2*
4	*Practical work 2. Data analysis and visualization in Python.*	*Lesson 1. Data preparation and preliminary analysis* *Task: load the selected dataset into the development environment. Import the necessary libraries (pandas, numpy, matplotlib, seaborn). Perform an initial data review: view the first rows, table dimensions, variable types, number of gaps, and presence of duplicates. Perform basic data cleaning — remove or process missing values, eliminate duplicates, change data types as needed. Prepare the dataset for further analysis.*	*2*
		Lesson 2. Descriptive statistics and primary analysis of big data *Task: perform descriptive statistics for numeric and categorical variables using the pandas library. Calculate basic statistical indicators (mean, median, minimum, maximum, standard deviation). Construct frequency tables for categorical variables. Draw conclusions about the nature of the data distribution, the presence of outliers or unevenness in the values of the features.*	*2*
		Lesson 3. Visualization of numerical and categorical variables, generation of a final report *Tasks: master the basic methods of data visualization in Python. Build histograms to study the distribution of numerical variables, column charts for categorical data, and boxplots to detect outliers. Customize styles, colors, axis labels, and legends to improve the visual presentation of information. Make brief analytical conclusions based on the constructed graphs. Build combined graphs (bar + line), pie charts, and develop one generalized dashboard and interactive visualization using the seaborn or plotly library. Make a final conclusion about the structure and features of the studied dataset, prepare a report in the form of a distributed notebook.*	*2*
5	*Lecture 3. Formatting time and date data, reading and writing files in Python.*	*Formatting time and date data in Python.* *Reading and writing files in Python.* *Interaction with external applications.*	*2*
6	*Lecture 4. Python and SQLite programming. Purpose of the csvsql utility.*	*Basic SQL operations. Working with SQLite using Python.* *Purpose of the csvsql utility. The execute() method.*	*2*

7	*Lecture 5. Procedure for importing data from files into Pandas. Importing data from the Internet using Pandas. Tools for correlation analysis in Pandas.*	*Statistical approaches to big data analytics. Using Pandas. Importing data from files. Importing data from the Internet. Descriptive statistics in Pandas. Tools for correlation analysis in Pandas.*	*2*
8	*Practical work 3. Correlation analysis in Python.*	*Lesson 1. Preliminary data analysis to identify correlations. Task: demonstrate skills in primary analysis and data preparation for further correlational research. It is necessary to download the data set, check it for omissions, perform descriptive statistics and visually assess possible relationships between variables using graphical methods (pairplot, scatterplot). The result should be a preliminary conclusion about which variables can potentially be correlated.*	*2*
		Lesson 2. Calculating and visualizing correlations between variables Task: demonstrate skills in performing correlation analysis of data using Python. You will need to calculate correlation coefficients (Pearson or Spearman) between selected variables, construct a correlation matrix in the form of a heat map, and visualize the relationships between variables using scatter plots. Based on the results, formulate conclusions about the strength and direction of correlations in the data set.	*2*
9	*Lecture 6. Handling missing data. Converting data types and manipulating date frames.*	*Handling missing data. Converting data types. Manipulating date frames in Python .*	*2*
10	*Lecture 7. Data Regression Analysis in Python.*	*Regression analysis. Types of regression analysis. Applying regression analysis to analyze data.*	*2*
11	*Practical work 4. Regression data analysis in Python.*	*Lesson 1. Familiarization with the data and preparation for constructing a regression . Task: Import the necessary Python libraries (pandas, numpy, matplotlib, seaborn, scikit-learn). Load the sales dataset. Perform preliminary data analysis: examine the table structure, identify gaps and anomalies, construct descriptive statistics. Visualize the relationship between the number of stores in the area and annual net sales (scatterplot, correlation analysis).*	*2*
		Lesson 2. Constructing and evaluating linear regression. Task: split the data into training and test sets. Build a linear regression model (LinearRegression with scikit-learn). Train the model and obtain regression	*2*

		coefficients. Forecast annual net sales. Evaluate the accuracy of the model using metrics (MAE, MSE, R^2). Plot the regression line and interpret the results.	
12	*Lecture 8. Errors in data analysis and predictive analytics. Estimating regression errors using Python.*	*Errors in data analysis and predictive analytics. Estimating regression errors using Python. Purpose of the scikit-learn library.*	*2*
13	*Lecture 9. Data classification algorithms. Applications and problems of classifications.*	*Classification problems. Classification algorithms. Visualization of classifications. Application and validation of classifications. Decision tree classifier model.*	*2*
14	*Lecture 10. Pyplot module. Plotly tool. Types of data visualization. Anomaly visualization. Using Folium and Leaflet.js libraries to build maps.*	*Pyplot module. Plotly tool. Types of data visualization. Anomaly visualization. Using Folium and Leaflet.js libraries to build maps .*	*2*
Topic 2. R language capabilities. Big Data architectural models.			
15	*Lecture 11. Data analysis in R. Factors, lists, frames and actions on them.*	*History of the development of the R language. Features of the R language. Objects, packages, functions. Vectors, matrices and operations on them in R. Factors, lists, frames and operations on them.*	*2*
16	*Lecture 12. Exporting, importing, and processing data in R.*	*Exporting and importing data in R. Using R for time series analysis. Processing data in R.*	*2*
17	*Lecture 13. Basic tools for data analysis and visualization in R.*	*The plot () function and its parameters. Managing common parameters - arguments of graphical functions. Types of graphs in R.*	*2*
18	*Lecture 14. Architectural models of Big Data. Virtualization technologies. Hypervisors. Container technology for executing program code on the server. SaaS, PaaS and IaaS.*	*Architectural models of Big Data engineering. Virtualization technologies. Layers of abstraction. Hypervisors. Container technology for executing program code on the server. Data engineering.*	*2*
19	*Lecture 15. Hadoop Big Data Technologies. Distributed MapReduce Processing. HDFS.*	*Scalability through big data. Data storage and processing in distributed file systems. Distributed databases. Hadoop Distributed File System (HDFS).*	*2*

20	*Lecture 16. Kafka distributed streaming platform. Cassandra advantages.*	*Data ingestion problem. Kafka distributed streaming platform. Cassandra advantages.*	*2*
21	*Lecture 17. Apache Spark Platform. Lambda and Kappa architectures for big data processing.*	*The problem of computational function. Spark technology. Comparison of Spark and MapReduce. Spark and sparklyr for working with big data in R. Lambda architecture. Advantages and disadvantages of Lambda architecture. Kappa architecture. Advantages and disadvantages of Kappa architecture.*	*2*
22	*Lecture 18. Final lesson.*	*Modular test.*	*2*

6. Self-study

The discipline "Big Data Technologies" is based on independent preparation for classroom classes on theoretical and practical topics.

№	*The name of the topic that is submitted for independent study*	*Hours of study*	*References*
1	*Preparing for Lecture 1.*	*2*	*[1], pp. 9–27*
2	*Preparing for Practical work 1.*	*3*	*[1], pp. 9–39*
3	*Preparing for Lecture 2.*	*2*	*[1], pp. 28–39*
4	*Preparing for Practical work 2.*	*3*	*[1], pp. 54–64, 65–71, 103–120*
5	*Preparing for Lecture 3.*	*2*	*[1], pp. 40–46*
6	*Preparing for Lecture 4.*	*2*	*[1], pp. 47–53*
7	*Preparing for Lecture 5.*	*2*	*[1], pp. 54–64*
8	*Preparing for Practical work 3.*	*3*	*[1], pp. 54–64*
9	*Preparing for Lecture 6.*	*2*	*[1], pp. 65–71*
10	*Preparing for Lecture 7.*	*2*	*[1], pp. 72–82*
11	*Preparing for Practical work 4.*	*3*	*[1], pp. 72–93*
12	*Preparing for Lecture 8.*	*2*	*[1], pp. 83–93*
13	*Preparing for Lecture 9.*	*2*	*[1], pp. 94–102*
14	*Preparing for Lecture 10.*	*2*	*[1], pp. 103–120*
15	*Preparing for Lecture 11.*	*2*	*[1], pp. 121–143*
16	*Preparing for Lecture 12.*	*2*	*[1], pp. 144–158*
17	*Preparing for Lecture 13.*	*2*	*[1], pp. 159–169*
18	*Preparing for Lecture 14.*	*2*	*[1], pp. 170–182*
19	*Preparing for Lecture 15.*	*2*	*[1], pp. 183–189*
20	*Preparing for Lecture 16.*	*2*	*[1], pp. 190–200*

21	*Preparing for Lecture 17.*	*2*	*[1], pp. 201–225*
22	*Preparing for Final Test.*	*20*	*[1], pp. 9–225, [1-11 add.]*

Policy and Assessment

7. Course policy

Forms of organizing the educational process, types of training sessions and assessment of learning outcomes are regulated by the Regulations on the Organization of the Educational Process at the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".

***Class attendance policy**. Attendance at lecture classes is mandatory. Attendance at practical classes may be occasional and required for the defense of practical work. The presence or absence of a student at a class is not assessed by awarding or deducting points. If a student cannot attend classes, he or she is still responsible for studying the theoretical material and completing practical assignments.*

***Policy on ethical norms in the classroom**: discipline; compliance with subordination; honesty; responsibility; respect for those present, turning off phones.*

***Policy on assessing learning outcomes**. The policy on assessing learning outcomes is regulated by the Regulations on the system of assessing learning outcomes at Igor Sikorsky Kyiv Polytechnic Institute. According to the Regulations, each grade is given in accordance with the criteria developed by the teacher and announced to students in advance. If a student fails to complete all four practical tasks, he/she will not be allowed to take the test. Failure to pass the current control measure (modular test) without good reason is assessed as 0 points.*

The policy and principles of academic integrity are regulated by the norms set of the Code of Honor of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" (https://kpi.ua/files/honorcode_en.pdf); Regulations on the Organization of the Educational Process, Regulations on the System for Preventing Academic Plagiarism, Regulations on the Commission on Ethics and Academic Integrity. Plagiarism and other forms of violation of the principles of academic integrity are unacceptable. The student must complete all practical tasks independently using open sources of information and acquired knowledge and skills.

Plagiarism and other forms of violation of the principles of academic integrity are unacceptable. All work for current and semester tests must be completed independently by the student using open sources of information and the acquired knowledge and skills.

All works that violate the principles of academic integrity (the program code does not match the assignment option, the identity of the program code among different works, etc.) are evaluated at 0 points. To gain access to the test, the student must independently complete the practical work (without changing the current rating). In the case of semester control work that violates the principles of academic integrity, the semester control report is marked "Eliminate".

Policy on appealing the results of assessment of control measures. According to the "Regulations on resolving conflict situations at Igor Sikorsky Kyiv Polytechnic Institute" (https://osvita.kpi.ua/sites/default/files/downloads/regulations_resolving_conflict_situations_2020.pdf) , students have the right to appeal the results of control measures with arguments, explaining which criterion they disagree with according to the assessment. A student can raise any issue related to the procedure of control measures and expect that it will be considered in accordance with predetermined procedures.

Policy on assigning incentive points. According to the "Regulations on the system of assessing learning outcomes at Igor Sikorsky Kyiv Polytechnic Institute", incentive points are not included in the main RSO scale, and their sum cannot exceed 10 points.

Incentive points are awarded for a creative approach in performing practical work (the maximum number of points for all work is 10 points), as well as for participation in scientific projects and conferences related to the topic of this course.

8. Monitoring and grading policy

Semester and current control in the discipline is evaluated by a scientific and pedagogical employee in accordance with the rating system for assessing learning outcomes and the policy of the academic discipline.

During the semester, students complete 4 practical assignments. The maximum number of points for each practical assignment is 15 points.

Points for practical work are awarded for:

- quality of practical work: 0-9 points;

- presentation of practical work and answer during defense: 0-2 points;

- completion of a practical report: 0-4 points.

Criteria for evaluating the quality of practical work:

9 points – the work was performed qualitatively, fully in accordance with the task;

6-8 points – the work is completed in full, but has shortcomings;

4-5 points – the work is completed in full, but contains minor inaccuracies in accordance with the task;

1-3 points – the work is not completed in full, contains significant inaccuracies in accordance with the task;

0 points – the work was not completed or was completed in violation of the principles of academic integrity.

Criteria for evaluating the presentation of work and answers:

2 points – the presentation is complete, the answers are well-reasoned;

1 point – the presentation was overall successful, the answers have shortcomings or minor errors;

0 points – no presentation, answers have significant errors.

Report evaluation criteria:

4 points – the report is completed in full;

2-3 points – the report was not completed in full;

0-1 point – the report is missing or the report has significant shortcomings.

The practical work report includes a file in .pdf, .docx, etc. format with a detailed description of the order of actions performed, as well as a file in .mp4 format with a demonstration of the results of the practical work.

Maximum number of points for performing and defending practical work:

15 points × 4 lab. works = 60 points.

The task for the modular test consists of 4 theoretical questions. The maximum number of points for the answer to a theoretical question is 10 points.

Criteria for evaluating modular test answers:

10 points – the answer is correct, complete, and well-reasoned;

7-9 points – the answer is correct, detailed, but not very well argued;

5-6 points – the answer is generally correct, but has shortcomings;

3-4 points – there are minor errors in the answer;

1-2 points – the answer contains significant errors;

0 points – no answer or incorrect answer.

Maximum number of points for the modular test:

10 points × 4 theoretical questions = 40 points.

Calendar control is carried out twice a semester as a monitoring of the current status of the implementation of the syllabus requirements.

At the first certification (7th week), the student receives a "pass" if his current rating is at least 7 points (50% of the maximum number of points that a student can receive before the first certification).

At the second certification (13th week), the student receives a "pass" if his current rating is at least 22 points (50% of the maximum number of points that a student can receive before the second certification).

Semester control: ***credit test***

Conditions for admission to semester control: upon passing all four practical works, the student is admitted to take the test.

The student's rating R is determined as the sum of the student's semester rating (accumulated rating for performing practical work R_{pr} and completing the modular test R_{MT}). The grade is set according to the value of the Table of correspondence of rating points to grades on the university scale.

The rating scale for the discipline is:

$R = R_{pr} + R_{MT}$ = 60 points + 40 points = 100 points.

The final performance score or the results of Final test the Fail/ Pass are adopted by university grading system as follows:

Score	Grade
100-95	Excellent
94-85	Very good
84-75	Good
74-65	Satisfactory
64-60	Sufficient
Below 60	Fail
Course requirements are not met	Not Graded

9. Additional information about the course

According to the Regulations on the recognition of learning outcomes acquired in non-formal/informal education at Igor Sikorsky Kyiv Polytechnic Institute https://osvita.kpi.ua/node/179, during the study of the discipline, it is possible to enroll in an online course or other element of non-formal education in the discipline.

Having a certificate of completion of a similar course is evaluated as 15 points (if the course topic corresponds to the topic of the practical work), writing articles or participating in conferences/projects on the relevant topic is also evaluated as an additional 5 points.

For example, one or more labs can be automatically credited by sending a certificate of completion to the classroom for a course such as:

Introduction to Data Science. https://www.netacad.com/courses/introduction-data-science?courseLang=en-US,

or, for example, Computer Workshop can also be automatically credited as 10 points by sending a certificate of completion of the following course to the classroom:

Google Data Analytics Professional Certificate (Google) A series of 8 courses — from data analysis to practical projects. https://www.coursera.org/professional-certificates/google-data-analytics

etc.

The student must inform the teacher about the completed or planned course of study and specify the results and prospects (conditions) of crediting the learning outcomes obtained in non-formal/informal education.

Syllabus of the course

Is designed by teacher PhD, Associate Professor, Liubov Oleshchenko

Adopted by Computer Systems Software Department (protocol № 12, 11.02.2026)

Approved by the Faculty Board of Methodology (protocol № 7, 23.02.2026)