"Alikhan Bokeikhan University"

Faculty of Information Technology and Economics

Department of "Information – technologies science»

6B06124 Computational technology and software

CATALOGUE OF ELECTIVE COURSES

Year of admission – 2024

Semey, year 2024

Discussed and approved at a meeting of educational and methodological council of the faculty
Protocol № 4 15.03.2024
 Chairman of the faculty EMS _____ Shoybakova E. O.
Approved at the meeting of the educational and methodological council of the university
Protocol № 5   28.05.2024
Chairman of the EMS _____ Zharykbasova K. S.

№	Discipline	Amount of credits		Prerequisites	Postrequisites	Prerequisites Postrekvizity Brief description indicating the purpose of the study, executive summary, and expected results of the study (knowledge, skills, competences)
		PK	ECTS			
BASIC DISCIPLINES						
Be sure to select(BSS)						
1	Introduction to Blockchain		6	Scool of Informatics Course	Object-oriented programming Delphi Operating systems	**Aim:** this course on mastering various aspects of blockchain technology. Students will study related topics of cryptography, wallets, nodes, smart contracts and tokens **Content**: Blockchain fundamentals, consensus algorithms, understanding cryptocurrencies and smart contracts, using blockchain technology in the real world. **Expected result:** **Know:** − basic blockchain concepts, advantages and limitations of blockchain technologies; − key differences between blockchain and other technological systems; **Able to:** − understand the technical underpinnings of blockchain technology deeply enough to conduct impact analysis of certain implementation solutions in proposals; − understand relevant legal, ethical and confidential blockchain-related matters; **Possess skills:** − influence on the policies and actions of organizations or individuals; − identifying a real-world problem that blockchain technologies can help solve; or explain why they won't help.
1	Fundamentals of Blockchain technology		6	Scool of Informatics Course	Object-oriented programming Delphi Operating systems, environments and shells	**Aim:** the study of blockchain technology (distributed ledger) with an emphasis on its mathematical and technical foundations, as well as applied aspects. **Content:** The architecture of the blockchain. The order of operation of the blockchain transaction. The purpose of the blockchain. Limitations of blockchain technology. Using blockchain technology to organize distributed databases **Expected result:** **Know:** - fundamentals of blockchain technology; - cryptographic fundamentals of blockchain technology on the platform .Net; - technologies for creating blockchain applications on the .Net platform **Able to:** - use blockchain technologies;

					- apply the cryptographic foundations of blockchain technology on the platform .Net; - create blockchain applications on the .Net platform **Possess skills:** - skills in using blockchain technology; - skills in applying cryptographic blockchain technologies on the platform .Net; - skills in creating blockchain applications on the .Net platform
2	Setting up, repair, optimization and maintenance of computer systems	5	Programming languages and technologies Electronics	Software development technology	**Aim:** Development of professional competences of students necessary for realization of professional activity, formation of abilities and skills on performance of the works connected with maintenance and repair of computer systems and complexes. **Content:** Configuring the computer equipment. Test the hardware with the debugger. Testing of hardware with the help of diagnostic programs. Create bootable media. Determining the parameters of the power supply. Testing the video system. Testing the CPU. Testing the hard drive. Restore the operating system to a healthy state. Keyboard and mouse maintenance. Test ports on the motherboard. Maintenance of the local network. **Expected result:** **Know:** - features monitoring and diagnostic devices hardware and software systems; - main diagnostic methods; - hardware and software functional control and diagnosis of computer systems capabilities and applications of standard and special test equipment to locate the ground fault SWT; - use of service tools and built-in test programs; - hardware and software configuration of computer systems and complexes; - installation, configuration and configuration of the operating system, - drivers, resident programs; methods of ensuring stable operation of computer systems and complexes. **Able to:** - to control, diagnose and restore the performance of computer systems and systems; - to carry out system technical maintenance of computer systems and complexes; - take part in debugging and technical testing of computer systems and systems; - installation, configuration and configuration of the operating system, drivers, resident programs. **Possess skills:** - carrying out of control, diagnostics and restoration of working capacity of computer systems and complexes; - systems engineering computer systems and complexes; - debugging of hardware-software systems and complexes; - installation, configuration and configuration of the operating system, drivers, resident programs.

2	Technics computer and communication systems	5	Programming languages	Software development process	**Aim**: the study of the discipline is the development of professional competencies of students necessary for the implementation of professional activities, the formation of skills and abilities to perform work related to the maintenance and repair of computer systems and complexes. **Contents**: Control and diagnostics of computer systems and complexes. Structure of computer systems and complexes Generalized structure of computer systems and complexes. Hardware of functional control and diagnostics of computer systems and complexes, their application. Software for functional control and diagnostics of computer systems and complexes, their application. **Expected result:** **Know**: - to control, diagnose and restore the performance of computer systems complexes; - carry out system maintenance of computer systems and complexes; - take part in debugging and technical testing of computer systems and complexes, installation, configuration and configuration of the operating system, drivers, resident programs; - comply with safety regulations; **Able to:** - features of control and diagnostics of devices of hardware and software systems; main diagnostic methods; - hardware and software for functional control and diagnostics of computer systems and complexes, possibilities and applications of standard and special control and measuring equipment for localization of fault locations of SVT; - application of service tools and built-in test programs; - hardware and software configuration of computer systems and complexes; - installation, configuration and configuration of the operating system, drivers, resident programs, methods of ensuring the stable operation of computer systems and complexes; - rules and norms of labor protection, safety, industrial sanitation and fire protection. **Own skills:** - monitoring, diagnostics and recovery of computer systems and complexes;
3	Theory of information processes	5	Introduction to Programming, Information and Communication Technologies	Computer networks and telecommunications	**Aim:** formation of students' theoretical knowledge about modern information systems and technologies, models, methods and means of solving functional problems and organizing information processes, studying the organizational, functional and physical structure of information systems and basic information processes, considering the prospects of using information technologies in the context of the transition to an information society**.** **Contents:** Introduction. Basic concepts and definitions. Signals as information carriers. Forms of representation of deterministic signals. Representation of message signals by basic functions. **Expected result.** **Know:**

					- Fundamentals of information culture; principles and structure of information processes and systems; - purpose and classification of software for digital information processing;. **Be able to:** − to use IP theory, information processes and computer technology in solving specific practical problems; − to assess the problems of the relationship between the individual, human society and nature; − to identify the effect of physical laws in the processes and phenomena of nature; to develop proposals for the organization of information processes and systems when using the information space using modern technologies, digital assets; − choose network technologies and automated document management tools of the organization;. **Possess skills:** − methods of organizing and using database management systems; methods of organizing and evaluating the effectiveness of the information space of the organization; − methods of working with application software
3	Information technologies	5	Algorithmization and programming, Information and communication technologies (in English.language)	Technics of computer and communication systems, Programming language, Data protection	**Aim:** Introduction to the basic concepts of information and information technologies, classification of information technologies by fields of application. **Content:** The Information. Information technology. Information technologies in the fields of application. Information security. The main types of threats to information security. Means of information protection. Identification and authorization of network users and resources **Expected result:** **Know:** - basic concepts: information and information technology; - technologies for collecting, storing, transmitting, processing and providing information; - classification of information technologies by fields of application: processing of text and numerical information, hypertext methods of storage and presentation of information, document markup languages; - general information about computers and computer networks: the concept of information system, data, databases, personal computer, server; - assign the computer a logical and physical structure of computer, hardware and software; - processor, RAM, disk and video subsystems; - peripherals: interfaces, cables and connectors; - personal computer (PC) operating system, file systems, file formats, file management programs; - local networks: protocols and standards of local networks; topology of networks, structured cabling systems, network adapters, hubs, switches, logical network structuring; - identification and authorization of users and network resources; - General information on global computer networks(Internet), addressing, domain names,

					data protocols, hypertext presentation, WorldWideWeb network (WWW), e-mail, server and client software;
					- information security: main types of threats, ways to counteract threats
					Able to:
					- work with graphical operating systems of a personal computer (PC): enable, disable, manage sessions and tasks performed by the operating system of a personal computer;
					- work with file systems, different file formats, file management programs;
					- work in applications: text and table editors, presentation editor, use information from technical documentation and help files.
					Possess skills:
					- presentation of information;
					- search for files, computers, and network resources;
					methods and means of determining the amount of information.
4	Decentralized applications	3	Introduction to Blockchain	Architecture of the blockchain system, Architecture of smart contracts, Introduction to Web 3	**Aim**: This course is aimed at learning the technical skills necessary to create decentralized applications on public blockchains. Also, the development of applications that perform business transactions without the participation of a trusted third party. **Contents**: the main components of a decentralized application (App), social and design problems preventing the implementation of DApp, smart contracts written in the Solidity programming language, the development environment necessary for writing, testing and deploying Ethereum DApp. **Expected result:** **Know**: – planning, designing, implementing and testing end-to-end dApps with proper configuration of the blockchain network on an enterprise scale **Able to**: – understand the architecture and components of DApp, including the external interface and internal processing supported by blockchain and smart contracts. **Possess skills**: – critically evaluating new blockchain standards and architectures and applying them in various use cases
4	LMS technologies	3	Fundamentals of Blockchain technology	Ethereum, Web3 and Truffle development environments, Working with Etherea, Introduction to Internet Marketing	**Aim**: The aim is to form a system of knowledge, skills and abilities among students in the field of using information and communication technologies in the process of their education, as well as in further practical activities in the context of organizing e-learning using distance learning technologies in an educational institution.. **Contents**: The concept of LMS? LMS MOODLE. Getting a username and password. Creating an e-learning course in Moodle. Tools (modules) Moodle. Resources. Interactive elements. **Expected result:** **Know**: – The basics of conducting practical classes in the chat

					– registration of electronic documents and import of documents from alternative formats **Able to**: – The use of test technologies (development of test tasks (tests), analysis of test results). **Possess skills**: – Formation of skills for setting up an e - course assessment; – Formation of skills for working with reports and e - course logs
5	Programming No Code/Lo Code	5	Introduction to Programming	Database programming, Programming languages	**Aim:** developing products without using code and understanding how an IT product fundamentally works. In the course, we will look at how to make high-quality projects using NoCode tools, what things you need to know before starting your path in No-Code development**.** **Content:** The concept of No-code and Low-code. The difference between No-Cod and Low-Code technology. Features of Low-code and No-code. Advantages and disadvantages of No-code. Advantages and disadvantages of Low-code. The main No-Code services. Low-Code and No-Code perspectives **Expected result:** **Know:** - the main content of the discipline has been mastered, allowing to assess the temporal and capacitive complexity of the software, but when presenting there are difficulties or inaccuracies that the student corrects with leading questions from the teacher; the main content of the discipline has been mastered, allowing to assess the temporal and capacitive complexity of the software, however, when presenting there are difficulties or inaccuracies that the student corrects independently - the content has been assimilated, which makes it possible to assess the time and capacity complexity of the software, the presentation is clear and competent, without difficulties and inaccuracies **Able to:** - performs all operations that allow you to evaluate the time and capacity complexity of the software, but makes mistakes that are not always able to eliminate without leading questions from the teacher - performs all operations that allow you to evaluate the time and capacity complexity of the software, but allows inaccuracies that can be eliminated without leading questions from the teacher - performs all operations that allow you to evaluate the time and capacity complexity of the software; the actions are thoughtful and error-free; **Possess skills:** - demonstrates an uncertain command of the required skills, the available experience is fragmented, but sufficient to complete the task - demonstrates mastery of the required skills, the available experience is sufficient to complete the task

					- demonstrates confident mastery of the required skills, the task is solved quickly and in an optimal way
5	Object-oriented programming	5	Introduction to specialty Introduction to computer science	Functional programming Fundamentals of microprocessor technics	**Aim:** Introduction to modern approach to programming in objects, acquisition of skills of writing programs in object-oriented languages. **Content:** Introduction to OOP. Structural features of object-oriented languages. Inheritance and composition. Fundamentals of object-oriented analysis and design. **Expected result:** **Know:** - what is a class and object; - the basic principles of object-oriented programming; - principles classes; - criteria for checking the correctness of the construction of classes; - main trends in the development of object-oriented programming technologies. **Able to:** - use modern methods of object-oriented programming in coding software systems of different complexity levels; **Possess skills:** - work with the environment of visual programming Delphi; - basics of algorithmization.
6	Probabilities theory and Math statistics	3	Scool of Mathematics	Computer modeling 3D graphics and animation	**Aim**: to obtain generalized knowledge of any probabilistic and statistical systems, to identify common patterns of their construction and operation. Identification of objects of application of the acquired knowledge with the use of modern information technologies. **Contents**: the Subject of probability theory and mathematical statistics. Basic concepts of probability theory. Trials and events. Actions on events. Random event. Types of random events. Basic formulas of combinatorics. The classical definition of probability. The theorem of adding the probabilities of incompatible events. Full group of events. Opposite events. Independent and dependent events. Multiplication theorem for independent events. Conditional probability. The solution of problems on conditional probability. **Expected result:** **Know:** - regularities in random and information processes (type of distribution, numerical characteristics, accumulation, processing, distribution, etc.)) **Able to:** - create mathematical and computer models of random phenomena in various fields of human activity; **Own skills:** - information about the main scientific achievements in the theory of probability and mathematical statistics;

| 6 | Discrete mathematics | 3 | Scool of Mathematics Course | Mathematical and computer modeling Interactive graphics systems | **Aim:** Acquaint students with the most important sections of discrete mathematics and its application in computer science.
Content: Set, element of set, subset. Operation on sets and their properties. Binary relations and their properties. The equivalence relation and split into classes. Types of functions: injections, surjections and bijections, inverse and compositions. Dirichlet principle. Construction of the truth table of logical formulas. Methods of proof: direct, inverse, negative, mathematical induction. Combinatorics.
Expected result:
Know:
- basic concepts of sets;
- algebraic methods model description;
- elementary functions of logic algebra, properties and their analytical representation;
- foundations of the logical calculus of propositional and predicate;
- methods for solving classical problems formulated in terms of combinatorics**.**
Able to:
- to apply combinatorial configuration for solving problems to determine the type of binary relations and its properties, perform operations on sets to represent graphs in different ways, to perform operations on graphs, finding shortest path graph, construct the truth table Boolean function, perform the identity transformation, find SDNF, SCNF to determine the minimum DNF.
Possess skills:
- use of basic tools of discrete mathematics for solving applied problems;
method of construction, analysis and application of discrete models in professional activity. |
| 7 | Operating systems | 5 | Information and communication technologies | Artificial intelligence systems Industrial practice (III) | **Aim:** Training in knowledge and skills of using modern software, obtaining knowledge about modern operating systems, their functional architecture, the resources and methods implemented by them, management of resources of computer complexes. To teach knowledge and skills in the use of modern software, to familiarize with the effective algorithms for solving various scientific and technical problems.
Content: General information about operating systems. History of operating systems. The architecture of the operating system. The basic functions of the OS. Processes and flows. Memory management. File system. Input and output management. The management of real memory. Configure network settings and share resources on local networks. Programming with system calls on the Windows operating system in the Linux System shell
Expected result:
Know:
- the concept, principles, types and functions of operating systems; operating environment;
- machine-independent properties of operating systems.
Able to:
- install and maintain operating systems; |

					- take into account the peculiarities of work in a particular operating system, organize support for applications of other operating systems; - use the tools of the operating system. **Possess skills:** - security and fault tolerance of operating systems; - principles of construction of operating systems; ways of organizing device support, hardware drivers, network operating systems.
7	Operating systems, environments and shells	5	Information and communication technologies	Functional programming Intelligent animation Industrial practice (III)	**Aim:** Study the theoretical principles and algorithms underlying the development of modern operating systems and shells, the development of problems in this area, a review of research areas, obtaining skills of installation, configuration and administration of operating systems Win32 and UNIX families. **Content:** Introduction. History of operating systems and shells. The basic functions of the OS. Processes and flows. Memory management. File system. Input / output control. **Expected result:** **Know:** - current state of the level and directions of development of computer technology and software; - main stages, methods, means and standards of software development; - main types of operating systems, operating system resource management principles; - features of operation in specific operating environments and shells; - service software tools; - methods of organizing, storing and processing information on the computer (technology of processing information on the computer). **Able to:** - to work in a chosen environment; - to learn a new operating system or shell program; - to obtain information about users, processes, directories, reference on system commands; - to perform a message exchange with other users; - create and view directories, copy, move and delete files, manage file access mode; - to create, view and merge text files, search pattern, search file according to the specified parameters, to use pipelines and redirection input / output. **Possess skills:** - security and fault tolerance of operating systems; - principles of construction of operating systems and shells; ways of organizing device support, hardware drivers, network operating systems.
8	Programming languages	5	Information technologies	Web programming Technics computer and communication systems	**Aim:** Consider the basics of building languages and programming methods, the study of the basic types and structures of data and algorithms for their processing, teaching students the basics of programming based on C++programming language. **Content:** Basic concepts of programming languages. Lexical analysis. Semantic analysis of program code. Object-oriented programming (OOP). Programming in language C **Expected result:** **Know:** - terminology of discipline;

					- basic structures and tools that are used in programming languages such as C++: - main structures and types of C++ data; - main methods in the development of algorithms (recursion, backward, branch and boundary methods, analysis of arithmetic expressions); basic algorithms; dialects C++, including used in programming microcontrollers; library of standard programs. **Able to:** - to apply programming techniques in the development of information systems; - determine data structures in the design of algorithms in the process of solving problems; - break down the solution of a complex problem into a sequence of more simple tasks. **Possess skills:** - use the library of standard programs that are included in the programming language C++; - self-settling in the programming language that you must use when solving problems.
8	Programming languages and technologies	5	Information theory Algorithmization and programming	Programming on PHP Setting up, repair, optimization and maintenance of computer systems	**Aim:** Teach students the basic principles and methods of building programs in programming languages, to familiarize with the semantics of programming languages, formal languages specifications, object-oriented specifications. **Content:** Structural, modular, object - oriented programming. Basic concepts and mechanisms of the environment of input and execution of programs. Base data type. Basic principles of organization and structuring of programs. Key concepts and linguistic means to describe software objects. Operating personnel. The main means of data processing. Preprocessor tools. Algorithmic basis for writing effective programs. Basic principles and means of organization of the software interface. Functions. Basic principles of program development. **Expected result:** **Know:** - programming methods and technologies; - basic data processing algorithms; - about modern programming languages; - about the structure of computing systems; **Able to:** - develop algorithms; - to implement algorithms in the programming language high-level; - implement the methods of analyzing and processing data; - work in programming environments. **Possess skills:** - methods and technology development of algorithms; - high-level programming language; - work in various programming environments.
9	Computer networks and telecommunications	5	Theory of information processes	Distributed systems technologies	**Purpose**: to Expand the theoretical base in the subject area, and to instill in students practical skills to work with special information support capabilities. **Contents**: definition of local networks. Local network topologies. The main components of the network. Types of Ethernet communication lines.

					The oldest standards of the network. Ethernet, TokenRing, FDDI **Expected results**: as a result of the study of the discipline the student must **know**: - The main components of the network, types of communication lines - IP address types - Methods and means of network protection - PHP syntax - SQL syntax - Types of domain and types of hosting **Able to:** - Create schemes HP - Clean your PC from viruses - Apply a digital signature - To apply the principles of encryption - Create PHP applications - Create websites with DB - Create a database using phpmyadmin and SQL - To process form data **Own skills :** - Create a LAN scheme - Perform network configuration and administration - Create applications in PHP - Creation and maintenance of websites - Publication of web-sites on the Internet - - System and network administration
9	Technics of computer and communication systems	5	Information technologies	Technology for the development of distributed information systems	**Aim:** Development of professional competences of students necessary for realization of professional activity, formation of abilities and skills on performance of the works connected with maintenance and repair and communication systems. **Content:** Control diagnostics and recovery of computer and communication systems. Systematic maintenance of computer and communication systems. Debugging and technical testing of computer and communication systems. Installation, configuration of the software. **Expected result:** **Know:** - features monitoring and diagnostic devices hardware and software systems; - main diagnostic methods; - hardware and software functional control and diagnosis of computer systems capabilities and applications of standard and special test equipment to locate the ground fault SWT; - use of service tools and built-in test programs; - hardware and software configuration of computer systems and complexes; - installation, configuration and configuration of the operating system, drivers, resident programs; methods to ensure the stable operation of computer systems and complexes; - rules and norms of labor protection, commercial - safety, industrial sanitation and fire protection **Able to:** - monitor, diagnose and restore the performance of computer and communication systems; - carry out system maintenance of computer and communication systems;

					- take part in debugging and technical testing of computer and communication systems; - installation, configuration and configuration of the operating system, drivers, resident programs; - to perform the safety procedures. **Possess skills:** - carrying out of control, diagnostics and restoration of working capacity of computer and communication systems; - system engineering services of computer and communication systems; - debugging of hardware-software systems and complexes; - installation, configuration and configuration of the operating system, drivers, resident programs.
10	Object-oriented programming in C++	5	Programming languages	Programming on PHP Modern methods and means of Java programing	**Aim:** In-depth study and development of programming languages based on object-oriented and generalized (using a standard library) programming technology, as the base language is used high-level programming C++. **Content:** Classes. Encapsulation. The design of conventional classes. Reference type. The creation and destruction of objects. Constructors and Destructors. Copy constructor. Hopscotch. Inheritance. The Programming Methods. Override operators (operations). Sorting of arrays. Virtual methods of classes, destructors. Abstract methods and classes. The use of virtual functions. Hopscotch. Encapsulation. Class member. Constructors and destructors. Friends classes. Overloading of operations. Templates. Name space. Expected result: **Know:** - the concept of object-oriented programming, its basic concepts (class, object), properties (encapsulation, inheritance, polymorphism); - method of analysis and design of object-oriented programs; - the basic concepts, the syntax and semantics of the constructs of the programming language C++; - methods of drawing up object-oriented programs in C++programming language; - features of the integrated programming environment in C++. **Able to:** - debug and test programs written in C++; - formulate the problem statement; perform a formalized description of the task, its algorithmization; - based on the existing algorithm to build a computer program in algorithmic languages and C++. **Possess skills:** - object-oriented design; - development of object-oriented software code in modern operating systems.
10	Functional programming	5	Programming languages and technologies	Web programming Modern methods and means of NET programing	**Aim:** Formation of students ' General methodological foundations and practical skills of developing software systems using a functional approach to programming **Content**: Introduction to functional programming. Introduction to the course. The paradigm of functional programming. A comparison of

					imperative and functional programming. The characteristic features of functional programming. Fundamentals of lambda calculus. Reasons for the use of lambda calculus formalization. The concept of lambda expression. Currying. Free and connected variables, expressions. **Expected result:** **Know:** - features of artificial intelligence problems and the role of functional programming as methodologies for solving these problems; - trends and prospects of functional programming tools development; - fundamentals of lambda calculus theory and practice. **Able to:** - develop software applications for solving the tasks in the functional programming language; - develop algorithms for solving problems for functional programming **Possess skills:** - work with the software application for solving of the tasks in a functional programming language; - development of algorithms for solving problems for functional programming.
11	Database design	5	Programming No Code/Lo Code	Preparing theses	**Aim:** Formation of students ' deep theoretical knowledge in the field of management, data storage and processing, as well as practical skills. **Content:**Introduction to the database. Database technology, basic concepts and definitions. DBMS, architecture of DBMS. Hierarchical, network, and relational data models. Relational systems, classification, client. Stages of database design. **Expected result:** **Know:** - features of the relational model and their impact on database design, visual AIDS used in ER modeling; - basics of relational algebra; principles of database design, ensuring the consistency and integrity of data; design database structures; - SQL query language. **Able to:** - design a relational database; - use SQL to programmatically retrieve information from databases. **Possess skills:** - searching and structuring information; modern techniques and technologies for the development and support of technical systems.
11	1C programming	5	Software in business	Preparing theses	**Purpose**: to Teach future specialists a complex of special knowledge and skills in the field of design and organization of highly efficient automated production processes of mechanical Assembly production in mechanical engineering. **Contents**: Basic concepts: configuration, configuration objects. Work options. The main and auxiliary application window. The set and properties of information storage objects. Constants. Reference books. The configuration branch is "General". Programming event handlers of configuration objects. Documents. Forms. Modules. Accumulation registers, data registers, transfers.

					Turnover Registers Reports. Programming the optimization of the document. Temporary table manager. **Expected result:** **Know**: - principles of construction of automatic machine systems and fundamental theories of automation of production processes; - features of automation of Assembly processes; - target mechanisms of automatic machines and automatic lines; **Be able to:** - design separate target mechanisms of automatic machines and automatic lines; - to design an automatic machine tool systems; - perform calculations of the performance and reliability of automatic equipment; **Possess skills:** - analysis of the performance, reliability and cost-effectiveness of automatic lines; - processing and analysis of statistical information on the reliability, performance and efficiency of automatic systems operation
12	Computer-modeling	5	Probabilities theory and Math statistics	Preparing theses	**Aim:** Development by students of methodology and technology of modeling (first of all computer) at research, design and operation of information systems. **Content:** Basic concepts of system modeling. Tools for modeling systems. Operation of the system. Formalization and algorithmization of processes of functioning of systems. Methods of planning experiments. Modeling of systems using typical machine schemes**.** **Expected result:** **Know:** - main concepts of modeling theory, classification of models and their use, modeling problems; - main modeling tools used in the process of designing systems at different stages of project detail; - methods of modeling and analysis of systems; - principles of construction of models. **Able to:** - perform an analysis of the system or process under study; reasonably choose a modeling method; - to build an adequate model of the system or process using modern computer tools; - to interpret and analyze the simulation results. **Possess skills:** - the main criterion of evaluation of the obtained simulation results; - experience of work and use in simulation of scientific and technical information.
12	3D graphics and animation	4	Discrete Math	Preparing theses	**Aim:** Mastering the knowledge of three-dimensional modeling, creating three-dimensional animation and visual effects. **Content:** Introduction to 3D. User interface with 3D StudioMax and Windows projections. Working with basic objects. Object transformation. Modifiers. Modeling of simple and complex forms. Toning. Animation. Visualization. **Expected result:** **Know:**

					- EN basic concepts of three-dimensional graphics; - features of 3D Studio max; - principles of creation, modification, texturing and lighting of objects on the subject plane, types of lighting, features of color rendering; - the principles and methods of transmitting motion in animation; - General principles for the development of the project in 3D Studio max; - the steps of creating a project in 3D Studio MAX. **Able to:** - create a fixed three-dimensional scene in accordance with the rules of artistic and technical design taking into account color-package solutions; - to create a simple animated three-dimensional scene using 3D Studio max; - to export and import image files into the 3D Studio MAX; - develop and submit to the defense your project created by the program in 3D Studio MAX. **Possess skills:** – create 3D graphics in 3D Studio max, Autodesk 3ds Max, and AutodeskMaya 3d. – Develop graphic and multimedia design
13	Modern methods and Java software	3	Object-oriented programming in C++ Programming languages	Preparing theses	**Aim:** Development Of methods and tools, as well as the basics of programming for Windows on Java and prepare for their active use in solving problems selected specialties. **Content:** The data structure and operations that apply to them. The control statements. Data entry and output. Arrays. Edit the arrays. Work with files. String manipulation. Treatment of special cases. Object-oriented programming in Java. Properties of the target programming. Packages and interfaces. Graphic primitive. Java integrated environment. Simple Java applications. **Expected result:** **Know:** - types, the characteristics of the data operations, and language operators; - principles of object-oriented programming; - fundamentals of computer networks and associations of networks, the internet, concepts, programming environment Java. **Able to:** - use classes to process applications; - work with files; use the principles of building a graphical interface, graphical primitive; convert applets. **Possess skills:** - work with operators, with arrays of application processing; - create classes, methods, publications, objects; - creating client components and applications; - work with Java network technologies.
13	Modern methods and software NET	3	Programming languages and technologies Functional programming	Preparing theses	**Aim:** Mastering methods and tools, as well as the basics of programming for Windows on NET and preparing for their active use in solving the problems of the selected specialties.

					Content: Basics of computer networks and networking. Internet services. The concept of the world wide web (world wide web). **Expected result:** **Know:** - types, the characteristics of the data operations, and language operators; - principles of object-oriented programming; - basic principles of computer networks and networking, internet services, concepts, programming environment NET. **Able to:** - use classes to process applications; - work with files; use the principles of building a graphical interface, graphical primitives; convert applets. **Possess skills:** - work with operators, with arrays of application processing; - create classes, methods, publications, objects; - creating client components and applications; - work with NET network technologies.
14	The use of data in machine learning	3	Artificial intelligence systems	Fundamentals of robotics and artificial intelligence	**Purpose:** this academic discipline is implemented as a series of lectures and practical classes that introduce students to the theoretical foundations and algorithms of machine learning, their practical implementation and use in solving specific problems. In this course, students should have an understanding of the principles of constructing some of the main classifiers and the problems to be solved using the theory in question. **Contents**: Introduction to machine learning. Statistical evaluation and hypothesis testing. Machine learning as mathematical modeling. Introduction to linear models and the problem of regression. Linear models and classification problems **Expected result:** **Know**: - principles for constructing feature vectors, decision rules, and classification; - main types of classifiers; - principles of construction of linear classifiers; - principles of constructing nonlinear classifiers; - selection of classification features and features of pre - processing of data. **Able to:** - selecting the appropriate type of classifier depending on the problem being solved; - selecting a set of features for classification and pre-processing data; - ability to use algorithms for training and compiling a classifier for selection; - Performing calculations related to the study and operation of the classifier in the MATLAB environment **Skills**: - skills for selecting, creating, training, and using basic classifiers problem solving

14	Introduction to machine learning and data analysis	3	Artificial intelligence systems	Robotic systems and complexes	**Aim**: to master the skills of working in Python, knowledge and understanding of data management tasks, including data loading, data conversion, and preliminary data analysis and visualization, familiarization with the main tasks and models of machine learning, knowledge of methods for evaluating the quality of various machine learning models, understanding the process of combining machine learning models within the tasks facing potential customers. increasing students ' interest in further deepening their knowledge in the field of data processing and machine learning **Contents**: discrete analysis and probability theory. Introduction to machine learning. Neural network. Criteria for selecting models and methods for selecting features. Logical classification methods. Clusterization methods **Expected result:** **Know**: - the main methods of data transformation; - know the main tasks of machine learning; the main models of machine learning; - Main stages of the machine learning project **Able to:** - these works massivement; - Formalizing a business task as a machine learning task - solving machine learning tasks in specific business tasks **Skills**: - Loading, converting, clearing, and visualizing data in Python - Applying machine learning models in Python - quality assessment and interpretation of the results obtained
15	Architecture of the blockchain system	3	Smart Contract Architecture	Preparing theses	**Aim:** This course is aimed at understanding and developing the architecture of blockchain systems. **Content:** Blockchain architecture, blockchain and trust architecture, database architectures and blockchain architectures (Explanation of the types of blockchain architectures), algorithmic management using smart contracts, hierarchical and alternative blockchain structures, from blockchain to distributed ledger technologies (DLT). **Expected result:** **Know:** - The inner workings of smart contracts as a means to develop decentralized applications; - interaction between the closed network of smart contracts and the outside world, - The further implications of these interactions are to understand the set of technologies that support the network's core decentralized storage network (e.g. IPFS, Swarm, Filecoin) **Able to:** - identify the key characteristics of the blockchain (i.e. decentralization, permanence, anonymity, verifiability, etc.);

					- explain the different levels of components that make up the architecture of a blockchain-based system; - understand the problems of consensus algorithms at a high level; - understand the algorithmic execution in DLT, their consensus model, code execution, its network operation, storage options, and the main actors who participate in each protocol; - Understand the underlying incentive and management models;. **Possess skills:** - forecasting the development and implementation of DLT in the future based on various use cases; - Understanding how other emerging technologies (for example, IoT and AI) can be used in combination with blockchain
15	Ethereum, Web3 and Truffle development environments	3	Working with Ethereum	Preparing theses	**Aim**: create a node of your own private Ethereum blockchain for further work within the framework of this course on the Ubuntu and Debian server **Contents**: Introduction to working with Truffle and installation instructions. A project in the Truffle environment. contracts on the Ethereum network **Expected result:** **Know**: - information about the Ethereum network node **Able to:** - rent a virtual or cloud server from one of the providers **Skills**: - working with the Geth node
16	Fundamentals of robotics and artificial intelligence	4	The use of data in machine learning 1C programming	Preparing theses	**Aim** of the course: to familiarize students with the basics of robotics, training programs for mobile robots **Contents**: Fundamentals of robotics. Physical fundamentals of robotics.Information in modulating, information processes.Design basics.Mobile work. From simple to complex.Algorithmization. Programming mobile robots.The decision of applied problems. Education robotics.Educational robotics. **Expected result:** **Know**: mathematical models of systems of automation and robotization of production processes using modern software data; analyze and evaluate mathematical models of systems of robotization and automation of production processes using modern software data. **Be able to**: design automation and robotization systems; comparative analysis with the use of modern software products for the robotization of technological complexes and systems for automating production processes in various industries, as well as artificial intelligence methods.; **Possess the skills** to form modern trends in the development of robotic systems and automation of production processes

16	Robotic systems and complexes	4	Introduction to machine learning and data analysis Database design	Preparing theses	**Aim** is to develop the ability to Express oneself creatively in the process of creating robotic systems by acquiring design skills. **Contents**: Robot actuators.Computing devices in the control system for robots and flexible production modules. Software control systems for industrial robots. Adaptive robot control systems. Robot sensitivity systems.Remote-controlled robots and manipulators. Solving software problems of applying robotic systems. **Expected result:** **Know**: industrial robot control systems; about remotely controlled robots; **Be able to**: solve programming problems using robotic systems **Possess the skills:** formation of work on the organization of processing; organization of work on the collection, storage and processing of information used in the field of professional activity
MAJORS					
Be sure to select (MSS)					
1	Fundamentals of component technologies	5	Programming languages	Programming on PHP	**Aim:** Training in modern methods and means of component programming. **Content:** Extensions to the C++ language environment C++ Builder. Additional types of data. Additional scopes. Model PME. Properties. **Expected result:** **Know:** - basic concepts of technology of component-oriented programming; - mechanisms for the implementation of the technology component programming in the library of visual components VCL; - the principles of event-driven programming; - technology user interface design of applications using a component library VCL; - the hierarchy of base class library of visual components VCL, their properties and methods; - purpose, properties, methods, usage characteristics, components, general purpose; - ways of organizing the application's user interface. **Able to:** - to analyze a subject area and choose the library classes required for the solution of applied problems; - to use the tools of the integrated development environment of C++ Builder for visual development of applications. **Possess skills:** - development of user interfaces of applications based on generic and specialized components, library, VCL; - the implementation of the application with different user interface types.
1	Component technologies and distributed software development	5	Programming languages and technologies	Web programming	**Aim:** Acquaintance with the concepts of distributed information system, distributed information processing, as well as the principles and problems of this subject area. **Content:** Main mechanisms of distributed object technologies. The problem of integration in

					distributed applications. Internet technologies when creating distributed applications. **Expected result:** **Know:** - main types of distributed applications; - modern development technologies and development of distributed applications; - main distributed object technologies and architectures (service-oriented architecture, component architecture, agent architecture, CORBA architecture). **Able to:** - develop distributed applications using socket technologies, remote procedure calls, component models, CORBA, web services; - select the development technology based on the specifics of the application. **Possess skills:** - development of distributed applications of different types; - the use of object-oriented programming in distributed systems.
2	Information security and information safety	5	Probabilities theory and Math statistics	Technologies of distributed systems	**Aim:** Formation of students ' knowledge system in the field of information security and practical application of methods of information security. **Content:**Information security of computing systems, a multilevel protection of corporate networks; protection of information in networks; the requirements of the information security systems **Expected result:** **Know:** - about protection of information of computer systems, the main subsystems of the computer, which cover concepts such as system highways, internal and external memory; - requirements for information security systems; - on the protection of corporate networks, the principles of security of information processing systems; - main characteristics of cryptographic methods of information protection. **Able to:** - in practice, to use means of information protection against unauthorized access and destructive software actions. **Possess skills:** - access to electronic information resources, databases, libraries, archives; - adaptation information resources and information technology; - work with documents containing restricted information.
2	Data protection	5	Discrete Math	Technologies of development of distributed information systems	**Aim:** Formation of students ' knowledge system in the field of information security and practical application of methods of information security. **Content:**Information security of computing systems, a multilevel protection of corporate networks; protection of information in networks; the requirements of the information security systems **Expected result:** **Know:** - about protection of information of computer systems, the main subsystems of the computer,

					which cover concepts such as system highways, internal and external memory; - requirements for information security systems; - on the protection of corporate networks, the principles of security of information processing systems; - main characteristics of cryptographic methods of information protection. **Able to:** - in practice, to use means of information protection against unauthorized access and destructive software actions. **Possess skills:** - access to electronic information resources, databases, libraries, archives; - adaptation information resources and information technology; - work with documents containing restricted information.
3	Microcontrollers and microprocessor systems	5	Digital circuit desing	Internet of things Production (IV)	**Aim:** Teaching students the principles of construction, functionality and architectural solutions of modern microprocessor systems (MPs), microcontrollers (MC) and personal computers, as well as the development of techniques for designing microprocessor systems. **Content:** Basics of organization and design of microprocessor systems (MPs). Architecture of microprocessors, ICS and microcontrollers (MC). Management of peripheral equipment in IPU. Data processing, management. The organization of interfaces in MPs and MK. Design of MPs Expected result: **Know:** - program-logic model of microprocessor 1810BM86; - modes of operation of the microprocessor 1810 BM86; - principles of construction of microprocessor systems; - program-logic model MCU series 1816; - modes of operation of micro-computer 1816 WE48; - features of the organization of system interrupts microprocessor and microcontroller 1810BM86 1816BE48; - organization of memory of 1816 series microcontrollers**.** **Able to:** - to build microprocessor systems on the basis of sets of 1816 and 1810; - to test the microprocessors in computers **Possess skills:** - composing electronic circuits for the operation of microprocessors and how to incorporate
3	Fundamentals of microprocessor technics	5	Digital circuit desing	Design of Distributed Control Systems Production (IV)	**Aim:** Familiarize students with the classification of microprocessor systems (MPs), basic architectures of MPs, functional units and the principle of the processor, by studying the architecture, command systems, the order of work with the main peripherals and subsystems of a particular single-chip RISC microcontroller, to consolidate the basic theoretical provisions

					Content: Overview of MK families AVR. The basics of programming in assembler, AVR MC. Familiarity with peripheral devices in the MC AVR. **Expected result:** **Know:** - principles of construction of electronic devices on the basis of modern element base and MPs; - principles of functioning of electronic devices on the basis of modern element base and MPs; - main technical parameters, performance characteristics and application fields of the main devices and functional units of electronics and MPs; - the basic principles of designing circuits on the basis of the IPU. **Able to:** - to perform the design and calculation of standard units of MEAs; - to make a choice of MPs to the required task. **Possess skills:** - perform analysis and synthesis of electronic circuits with MPs; - of design and analysis of electronic devices with the help of computers.
4	Smart Contract Architecture	3	Decentralized applications	Architecture of the blockchain system	**Aim:** The course aims to understand the many possibilities of creating decentralized applications using the Web3 stack and the Solidity language on the Ethereum Virtual Machine (EVM).. **Content:** Introduction to blockchain and Ethereum, introduction to smart contracts, blockchain technology and support for full Turing languages, virtual machines, introduction to the DApp development pipeline, deep immersion in Solidity, global variables and functions, expressions and control structures. **Expected result:** **Know:** - The Ethereum model, consensus model, execution code, network operation, data storage options and the main actors who participate in its protocol; - The inner workings of smart contracts as a means to develop decentralized applications **Able to:** - develop smart contracts using the Solidity programming language (including a deep understanding of the libraries provided); - to understand the interaction between the closed network of smart contracts and the outside world, to realize the further consequences of these interactions for the aspect of decentralization;. **Possess skills:** - using the smart contract development lifecycle (contract implementation, testing, deployment and contract migration); - a set of technologies that support a backbone decentralized data storage network (for example, IPFS, Swarm).

4	Working with Ethereum	3	LMS technologies	Ethereum, Web3 and Truffle development environments	**Aim:** develop your own application for working with the Ethereum-based blockchain. At the end of the course, you will create a wallet program for storing cryptocurrency with the ability to access multiple people. **Content:** The Ethereum blockchain. Creation and testing of the simplest smart contracts. Data storage and processing in distributed registries. Implementation of gaming smart contracts. **Expected result:** **Know:** - algorithmic programming languages, operating systems and shells, modern software development environments. **Able to:** - create algorithms, write and debug codes in a programming language, test the performance of a program, integrate software modules. **Possess skills:** - programming language; debugging and testing skills of the program.
5	Internet of things	3	Microcontrollers and microprocessor systems,	The use of data in machine learning	Aim: to study the organization of automated systems based on the Arduino microcontroller, application of these systems in the automation of scientific experiments, practical skills of working with modern controllers, solving automation problems. **Contents**: Applied electronics. The case of "computer vision". Software development. The "game console" case. Web-technology. State of the smart home. Design basics. The case of "smart mirror". **Expected result:** **Know:** - principles of organization and functioning of the "Internet of things" - History of the origin and development of the "Internet of things" - main factors in the development of the " Internet of things" - Existing technologies in the Internet of things industry" - Main trends and directions in the field of "Internet of things". **Be able to:** - work with microcontrollers and main repair plates (Arduino and Raspberry Pi)) - understand existing IoT technologies and their application to specific scenarios - design of integrated IoT systems (including end devices, network connections, data exchange, cloud platforms, data analysis). **Possess skills:** - terminology - basic knowledge of programming end devices - basic skills for connecting end devices to the network - basic cloud technologies for developing software solutions for data processing and storage.
5	Design of Distributed Control Systems	3	Fundamentals of microprocessor technics	Introduction to machine learning and data analysis	**Aim:** Development of the concept and methodology of analysis and synthesis of complex systems, design principles of information systems

					Content: Phases and stages of designing SU. Sketch design. Analysis and synthesis of data processing systems. Synthesis and analysis of the control system. Measures to assess the quality of the designed system. Synthesis of XOIU. **Expected result:** **Know:** - properties, characteristics and architecture (structure and topology) of distributed control and automation systems (DCS), - types of support {methodical, technical, software, informational, metrological, ergonomic, organizational and legal); - functional objectives and performance criteria of DCS. **Able to:** - to carry out projects of means of automation, systems of automation of technological processes: - perform automation of research and testing: - design and implement algorithms for preprocessing information (compression, filtering, improving the accuracy of conversion, etc.).), - Build modern control algorithms (modal, neuro-fuzzy, network - centric, etc.). - to determine the section of the network with the maximum transmission delay of IP packets; - to form HTTP requests and parse the fields in the HTTP response; - develop hypertext documents. **Possess skills:** - implementation of formal construction and transformation of analytical and simulation models of DCS; - the application of methods and techniques for the analysis and synthesis of RSU architectures; - development and use of analytical and simulation models of DCS for evaluation of design solutions; - implementation of the sequence of design stages of control and automation systems.
6	Programming on PHP	5	Programming languages Object-oriented programming in C++	Preparing theses	**Aim**: currently, - one of the most popular languages for the implementation of web applications. This course is devoted to the study of its basics. The emphasis is on the practical application of the acquired skills. PHP language was created to solve a specific practical problem in the Internet environment. Familiarity with the PHP language, development of skills in design and programming of web applications. **Contents**: Discusses how to separate statements, create comments, variables, constants and data types, operators. Conditional statements (if, switch), working with loops (while, for, foreach) and using include, require functions. **Expected result:** **Know:** - principles of Internet services; **Able to:** - create static and dynamic pages. **Possess skills:** - programming and client-server technologies.

6	Web programming	5	Programming languages and technologies Functional programming	Preparing theses	**Aim**: the Discipline "Web-programming" aims to learn the basics of Java programming and basic concepts that allow students to get a basic understanding of effective ways to develop Web applications, along with the acquisition of practical skills **Contents**: introductory lecture. The subject of the course, the emergence and development of Java. Java virtual machine. Algorithmic tools of the Java language. Vocabulary of the language. Data type. Operations. Control structures. Naming rules. Packages. Objects and classes. **Expected result:** **Know:** - static web-site development technologies; - methods of using multimedia (graphics, video, animation) on web-pages; - client-side software tools used to create web pages; **Able to:** - design and develop the structure of the site; - use HTML hypertext markup language and cascading style sheets (CSS) to create web pages; - develop scripts in the JavaScript programming language; **Possess skills:** - creation of web-sites; - placement of the web-site on the server and its maintenance; - registration of the site in search engines.
7	Blockchain Business Models	3	Introduction to Web3	Preparing theses	**Aim:** This course is aimed at analyzing entrepreneurship and innovation management based on blockchain technology **Content:** The importance of innovation, the innovative nature of digital currencies / blockchain / DLT, the management and dissemination of blockchain innovations, the transformation of the idea of blockchain into a business plan, the application of design thinking and strategy in blockchain projects, risk analysis and management **Expected result:** **Know:** - business opportunities, design and develop new blockchain-based services, as well as create and develop a successful business; **Able to:** - Develop ideas and innovative strategies; **Possess skills:** - Development of a business model consistent with the principles of digital currencies, decentralization and the growth of peer-to-peer transactional relationships between producers and consumers.
7	The basics of blockchain and cryptocurrencies	3	Introduction to Internet Marketing	Preparing theses	**Aim:** the study of the features of blockchain technology and the use of cryptocurrencies, the formation of skills and abilities of investing in cryptocurrencies and the use of the ICO mechanism for financial support of an innovative project **Content:** Cryptocurrency and markets. Mining and cryptocurrencies — how to use and interact. Cryptocurrencies and Smart contracts **Expected result:** **Know:** - features and principles of functioning of cryptocurrencies, their types;

					- The importance of blockchain technology in the functioning of cryptocurrency instruments; - The main types of consensus in blockchain technology: PoW (Proof-of-work) and PoS (Proofof-stake); - The main ways to invest in cryptocurrencies; - features of existing cryptocurrency trading platforms;. **Able to:** - to develop the specifics of the legal regulation of transactions in the cryptocurrency market in different countries of the world; - to solve the problems of state regulation in the field of blockchain technology and the use of cryptocurrencies;. **Possess skills:** - raising funds through the mechanism of initial coin issuance (ICO).
8	Distributed systems technologies	5	Information security and information safety Computer networks and telecommunications	Preparing theses	**Aim:** Disclosure of the essence of distributed computing technology, principles and technologies of distributed databases, to characterize the technologies and models of "Client-server" used in modern enterprises. **Content:** Distributed computing technologies. Distributed databases Client-server technologies and models Object data binding technologies Data replication technologies **Expected result:** **Know:** - Principles of building distributed information processing systems; - database distribution; - Client-server network technology and models; - object data binding technologies. **Able to:** - to use technologies for the construction and operation of distributed information systems **Possess skills:** - s Work with modern distributed systems design and development systems.
8	Technologies for the development of distributed information systems	5	Information security Technics of computer and communication systems	Preparing theses	**Aim:** Theoretical and practical training of students in the field of information technology to such an extent that they can choose the necessary technical, algorithmic, software and technological solutions, be able to explain the principles of their functioning and use them correctly. **Content:** Communication in distributed systems. Remote procedure call. Safety. Types of connections. Distributed transactions. The concept of a transaction. The ACID principle. Nested transactions. Distributed transactions **Expected result:** **Know:** - Principles of building distributed information processing systems; - communication in distributed systems; - types of links; - the concept of a transaction. **Able to:** - to use technologies for the construction and operation of distributed information systems. **Possess skills:**

					- working with modern distributed systems design and development systems.
9	Software development technology	6	Distributed systems technologies	Preparing theses	**Aim:** Study of software classification, tools and methods of software tools, tools and methods of detection, processing tools and compilation, download, installation of software characteristics of devices, their information support, support and implementation of software, the practical application of modern processing tools. **Content:** Introduction. State and foreign normative documents, determination of treatment composition. RUP. Processing of applications. DC. The tools and techniques of logical design. UML. Description of processing functionality. The tools and techniques. Create a cluster diagram. Methods, technologies, tools. Define language processing, determine the propagation medium and determine the processing tools. Physical design procedure-order, tool, resource, documentation Tools for visual programming with MS VisualStudio, BorlandDelphi and others. Selection and editing of components, machining of components. Open the API TOOL. The repair software. Tooling. Repair method. Testing. Variants and examples of tests. Selection and editing of components, machining of components. Open the API TOOL. The creation of a software interface. The principles of processing tool. Methods for creating interfaces and tools. Optimize processing time and size. The tools and techniques. Create help. The tools and techniques. Principles of software development and protection. Principles of software development. **Expected result:** **Know:** - modern trends in computer science, computer technology; - basis of creation of information systems and use of new information technologies of information processing; - life cycle of the software; object-oriented programming; - theories and methods of classification; - elements of complexity theory. - test the software **Able to:** - use mathematical methods, physical laws and computational techniques to solve practical problems; - program in one of algorithmic languages; - to apply algorithms of information retrieval IN software development. **Possess skills:** - drafting of projects for the development of modern software; - technologies of data collection, processing, transmission and storage. - software development

					- software development and testing, WEB and multimedia applications
9	Software development process	6	Technologies for the development of distributed information systems	Preparing theses	**Aim:** Study the classification, life cycle, technology rapid software development. **Content:** Life cycle of the software. Identification of requirements for the software system. Work with customers. Review of software design methodologies. Fast software development technologies. Object-oriented design of a software system. Information support tools for software projects and products (CALS) technologies. Testing and debugging of software systems. Assessment of the quality of the software. Implementation and maintenance of software products **Expected result:** **Know:** - theoretical basis of software tools; - classical and modern methods of building the information structure and interface of the tool. **Able to:** - select tools when creating software; - to apply the standards of construction of the software; - to assess the effectiveness of tools and the analysis of qualitative characteristics; - realize the economic efficiency of the software; - to apply object-oriented and structured methods of distribution in control and measuring instruments. **Possess skills:** - software development hard; comparative analysis of selection tools.
10	Introduction to Web3	2	Decentralized applications	Blockchain Business Models	**Aim:** This course aims to understand the basic principles of decentralization and the use of web 3 to create various types of decentralized applications. **Content:** the difference between web2 and web3, the basic principles of decentralization, distributed IPFW and Swarm systems, Ethereum protocols, creation of smart contracts for voting tasks, asset management and identification **Expected result:** **Know:** - smart contract deployment; **Able to:** - apply protocols in web 3 applications **Possess skills:** - applications of decentralization in applications; - web 3 application development;
10	Introduction to Internet Marketing	2	LMS technologies	The basics of blockchain and cryptocurrencies	**Aim:** The purpose of studying the discipline is to form students' solid knowledge in the field of applying the capabilities of Internet technologies in promoting an enterprise to the market, as well as the practical application of knowledge gained in the marketing course, the acquisition by students of skills and abilities necessary for successful work as marketing specialists in enterprises of any form of ownership. **Content:** An overview of Internet marketing tools. Development of an Internet marketing strategy. E-mail marketing. Social media and New media

					Marketing
					Expected result:
					Know:
					- key concepts, goals and objectives of Internet marketing; the specifics of the work of projects in the Internet space; typical mistakes that should be avoided in Internet projects;
					- how to competently build business communication;
					- the theoretical foundations of business communications, their main types and materials;
					- business communication, its structure and principles;
					- psychological characteristics and types of subjects of the communicative process
					Able to:
					- to compose and describe the entire life cycle of an Internet project, from the origin of an idea to its extinction;
					- to analyze any segment of the Internet business, to evaluate the effectiveness, monetization, and work with the audience;
					- prepare analytical reports on all areas of the project's life (market analysis, competition assessment, promotion methods, approaches to working with the audience, quantitative indicators, sources of financing);
					- formulate questions, critically evaluate the selection of promotion tools
					Possess skills:
					– information in global computer networks and corporate information networks; the use of modern tools used in carrying out a range of Internet marketing activities;
					– business communication technologies, a wide range of communication techniques and techniques for establishing contact with the interlocutor, creating an atmosphere of trusting communication;
					– organization of feedback in order to use them effectively in professional activities;
					– methods of cognition of the personality of the communication partner

LIST OF COMPONENTS BY CHOICE
B057- Information technologies
«6B06124 «Computational technology and software»

DISCIPLINE LIST
elective courses

Form of training: Full-time
Training period: 4 years

Name of the discipline	discipline code	Credits	Semester
Basic disciplines			
Component on a choice 1			
Introduction to Blockchain	IB 2212	5	3
Fundamentals of Blockchain technology	FBT 2212		
Component on a choice 2			
Setting up, repair, optimization and maintenance of computer systems	SROMCS 2213	5	3
Maintenance and repair of computer systems and complexes	MRCSC 2213		
Component on a choice 3			
Theory of information processes	TIP 2214	5	3
Information technologies	IT 2214		
Component on a choice 4			
Decentralized applications	DA 2215	3	4
LMS technologies	LMST 2215		
Component on a choice 5			
Programming No Code/Lo Code	PNC/LC 2216	5	4
Object-oriented programming	OOP 2216		
Component on a choice 6			
Probabilities theory and Math statistics	PTMS 1217	3	2
Discrete Math	DM 1217		
Component on a choice 7			
Operating systems	OS 3218	5	5
Operating systems, environments and shells	OSES 3218		
Component on a choice 8			
Programming languages	PL 3219	5	5
Programming languages and technologies	PLT 3219		
Component on a choice 9			
Computer networks and telecommunications	CNT 3220	5	6
Technics of computer and communication systems	TCCS 3220		
Component on a choice 10			
Object-oriented programming in C++	OOPC 3221	5	6
Functional programming	FP 3221		
Component on a choice 11			
Database programming	DP 4222	3	7
Programming on 1C	P1C 4222		
Component on a choice 12			
Computer modeling	CM 4223	5	7
3D graphics and animation	3DGA 4223		
Component on a choice 13			
Modern methods and means of Java programing	MMMJP 4224	3	7
Modern methods and means of NET programing	MMMNP 4224		
Component on a choice 14			
The use of data in machine learning	TUDML 4225	3	7
Introduction to machine learning and data analysis	IMLDA 4225		
Component on a choice 15			
Architecture of the blockchain system	ABS 4226	3	7

Ethereum, Web3 and Truffle development environments	EWTDE 4226		
Component on a choice 16		4	8
Fundamentals of robotics and artificial intelligence	FRAI 3224		
Robotic systems and complexes	RSC 3224		
ProfilingDiscipline			
Component on a choice 1		5	6
Fundamentals of component technologies	OKT 3305		
Component technologies and distributed software development	CTDSD 3305		
Component on a choice 2		5	5
Information security and information safety	ISIS 3306		
Information security	IS 3306		
Component on a choice 3		3	5
Microcontrollers and microprocessor systems	MMS 3307		
Fundamentals of microprocessor technics	FMT 3307		
Component on a choice 4		3	6
Smart Contract Architecture	SCA 3308		
Working with Ethereum	WE 3308		
Component on a choice 5		5	6
Internet of things	IT 3309		
Design of Distributed Control Systems	DDCS 3309		
Component on a choice 6		5	7
Programming on PHP	PPHP 4310		
Web programming	WP 4310		
Component on a choice 7		3	7
Blockchain Business Models	BBM 4311		
The basics of blockchain and cryptocurrencies	TBBC 4311		
Component on a choice 8		5	7
Distributed systems technologies	DST 4312		
Technologies for the development of distributed information systems	TDDIS 4312		
Component on a choice 9		6	8
Software development technology	SDT 4311		
Software development process	SDP 4311		
Component on a choice 10		2	5
Introduction to Web3	IW 3314		
Introduction to Internet Marketing	IIM 3314		