

Faculty of Information Technology and Cybersecurity

Department of «Computer Engineering»

APPROVED BY

Vice-rector for academic affairs,
International Information
Technology University JSC
Mustafina A.K.



“ 14 ” 03 2023

6B06110

(Code of Academic Program)

Software Engineering

(Name of Academic Program)

CATALOGUE OF ELECTIVE DISCIPLINES

2023 entry year

2023

The catalogue of elective disciplines for the specialty/AP 6B06110 Software Engineering is developed on the basis of the working curriculum of the specialty/AP.

The catalogue of elective disciplines was discussed at a meeting of the Computer Engineering department

minutes No. _____ from “ _____ ” _____ 2023

Head of Department

CED compiler



Chinibayeva T.T.

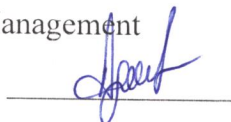
Yemberdiyeva A.B.

The catalogue of elective disciplines was approved at a meeting of the Academic Council of JSC IITU

minutes No. _____ from “ _____ ” _____ 2023

Head of the Management

Department



Adzhibayeva A.SH.

1 TERMS AND ABBREVIATIONS

1.1 Academic program is a single set of basic characteristics of education, including goals, results and content of training, the organization of educational process, ways and methods for their implementation and criteria for assessing learning outcomes. The content of academic program of higher education consists of three cycles of disciplines - general education disciplines (hereinafter - GED), basic disciplines (hereinafter - BD) and core disciplines (hereinafter - CD). The cycle of GED includes disciplines of the compulsory component (hereinafter - CC), the university component (hereinafter - UC) and (or) the component of choice (hereinafter - COC). BD and CD include disciplines of UC and COC.

1.2 Catalogue of elective disciplines (CED) is a systematic annotated list of all COC disciplines, for the entire training period, containing a brief description indicating the purpose of study, a summary of main sections and expected learning outcomes. CED reflects the prerequisites and postrequisites of each academic discipline. It should provide the students with the possibility of an alternative choice of elective disciplines for the formation of an individual educational trajectory.

On the basis of academic program and CED, the students develop individual curricula with the help of advisers.

1.3 Individual curriculum (IC) is a curriculum formed by the students independently with the help of an adviser for each academic year on the basis of the academic program, the catalogue of elective disciplines or modules;

IC defines an individual educational trajectory of each student separately. It includes disciplines and types of educational activities (internship, experimental research, forms of final certification) of the compulsory component (CC), the university component (UC) and the component of choice (COC).

1.4 Advisor is a teacher who performs the functions of an academic mentor of a student (according to the appropriate academic program), and assists in choosing a learning path (creating an individual curriculum) and mastering the academic program during the training period.

1.5 The university component is a list of compulsory educational disciplines determined by the university independently for the mastering of the academic program.

1.6 The component of choice is a list of academic disciplines and the corresponding minimum amounts of academic credits offered by the university and independently chosen by students in any academic period, taking into account their prerequisites and postrequisites.

1.7 Elective disciplines are educational disciplines that are a part of the university component and the component of choice in the framework of established academic credits, introduced by organizations of education reflecting the individual preparation of students and taking into account the specifics of socio-economic development, the needs of a particular region and established scientific schools.

1.8 Postrequisites are the disciplines and (or) modules and other types of academic work, the study of which requires knowledge, skills and competencies acquired at the end of the study of this discipline and (or) modules;

1.9 Prerequisites are the disciplines and (or) modules and other types of educational work containing knowledge, abilities, skills and competencies necessary for the mastering of the studied discipline and (or) modules;

1.10 Competencies are the ability of the practical use of acquired knowledge and skills in professional activities.

2 ELECTIVE DISCIPLINES

| № | Cycle of discipline | Code of discipline | Name of discipline | Semester | Number of credits | Prerequisites |
|---------------|---------------------|--------------------|---|----------|-------------------|------------------|
| <i>3 year</i> | | | | | | |
| 1 | PD | SFT6309 | UX/UI development | 5 | 5 | ICT6001 |
| 2 | PD | SFT6310 | Web-component development (Java EE) | 6 | 7 | SFT6002 |
| 3 | PD | MIN601 | Minor 1 | 5 | 5 | |
| 4 | PD | NET6310 | Linux Operating System | 5 | 5 | SFT6002 |
| 5 | PD | SFT6313 | Mobile technologies and applications(Android) | 5 | 7 | SFT6002 |
| 6 | PD | SFT6328 | Development of mobile applications on IOS | 5 | 7 | SFT6307 |
| 7 | PD | SFT6311 | Front-end development | 6 | 5 | SFT6002 |
| 8 | PD | MIN602 | Minor 2 | 6 | 5 | |
| <i>4 year</i> | | | | | | |
| 9 | PD | SFT6314 | Full stack development | 7 | 5 | SFT6311, SFT6307 |
| 10 | PD | MIN603 | Minor 3 | 7 | 5 | |
| 11 | PD | SFT6376 | Microsoft .NET Framework | 7 | 5 | SFT6002 |
| 12 | PD | SFT6321 | QA testing | 7 | 6 | SFT6002, SFT6307 |
| 13 | PD | SFT6319 | Blockchain technology | 7 | 6 | SFT6307 |
| 14 | PD | SFT6315 | DevOps | 7 | 7 | SFT6002, SFT6307 |
| 15 | PD | SFT6323 | Programming on the QT platform | 7 | 7 | SFT6002, SFT6311 |

3 DESCRIPTION OF ELECTIVE DISCIPLINES

| Description of discipline | |
|----------------------------------|---|
| Code of discipline | SFT6309 |
| Name of discipline | UX/UI development |
| Number of credits (ECTS) | 5 |
| Course, semester | 3, 5 |
| Department | CE |
| Prerequisites | Information and Communication Technology |
| Postrequisites | Diploma project |
| Brief course description | <p>This course introduces students to the concept of designing systems that are able to interact effectively with humans. The field of Human Computer Interaction involves understanding and creating methods and artifacts that improve human lives, tasks, goals, and social environments through education in design, computer science, and behavioral and social sciences. In this course, students will learn principles of design and human behavior and the empirical research methods used to solve real problems in the design and use of technology. The course also provides students with opportunities to work on their own as well as in small teams to solve design problems and use HCI methods and principles to model the problems, create solutions, and study the impact of their designs.</p> |
| Expected learning outcomes | <p>After successful completion of the course students will be able to:</p> <p>Define and Discuss:</p> <ul style="list-style-type: none"> – the concept of usability engineering, why and when to use it, why and when usage is justified, and its underlying benefits and principles; – the standard usability tools and methods such as personas and scenarios, competitive analysis, flow diagrams, generalized transition networks, site maps, storyboards, wireframes and mockups; – usability-testing methods. This includes understanding the process of planning and preparing a user test, determining and recruiting participants, designing test tasks, scripts, and scenarios, executing a user test, and recording and analyzing user-test data. <p>Use and Design:</p> <ul style="list-style-type: none"> – HCI tools, methods and concepts to design systems that are able to interact effectively with humans; – the principles of design and human behavior, computer science, and the empirical research methods used to solve real problems in the design and use of technology; – user interfaces from the perspective of the user, creating a design that supports its intended users' existing beliefs, attitudes, and behaviors as they relate to the tasks that the system is being designed to support; – an iterative design process to design interfaces that provide more efficient and satisfying experiences for the user; – design, plan, and conduct usability test and use the results of |

| | |
|--|---|
| | the test to create recommendations for design improvements and implement those recommendations. |
|--|---|

| Description of discipline | |
|----------------------------------|--|
| Code of discipline | SFT6310 |
| Name of discipline | Web-component development (Java EE) |
| Number of credits (ECTS) | 7 |
| Course, semester | 3,6 |
| Department | CE |
| Prerequisites | Object-Oriented Programming |
| Postrequisites | Business Component and Web-services Development (Java EE) |
| Brief course description | During the course, students will learn front-end development from defining functionality of web sites to layout of web pages and back-end development using modern web technologies. Content of the course covers following web technologies: REST API, JAX RS, Hibernate ORM, Spring MVC Framework, Spring Security, Tomcat Server, Servlet API, JSP. During the semester, students will learn development of corporate systems in Java programming language, and correct using of Servlets and JSP. Students must understand MVC pattern, developing secured web application. Students will be able to practice on real web projects and tasks. In addition, students will be able to study new trend technologies by researching. |
| Expected learning outcomes | After successful completion of the course students will be able to: <ul style="list-style-type: none"> - analyze advanced web technologies for solving various types of tasks; - explain and justify of using java web development tools for certain purposes; - know Java Programming Language; - explain the basics of Servlets and JSP (Java Server Pages); - hibernate ORM library; - develop secured corporate server-client web applications. |

| Description of discipline | |
|----------------------------------|---|
| Code of discipline | MIN601 |
| Name of discipline | Minor1 |
| Number of credits (ECTS) | 6 |
| Course, semester | 3,6 |
| Department | CE |
| Prerequisites | - |
| Postrequisites | - |
| Brief course description | Additional educational program (minor) - a set of disciplines and (or) modules and other types of educational work, determined by students for study in order to form additional competencies |

| | |
|----------------------------|---|
| Expected learning outcomes | - |
|----------------------------|---|

| Description of discipline | |
|----------------------------------|---|
| Code of discipline | NET6310 |
| Name of discipline | Linux Operating System |
| Number of credits (ECTS) | 5 |
| Course, semester | 3, 6 |
| Department | CE |
| Prerequisites | Programming technologies; Algorithms and data structures; |
| Postrequisites | Course work |
| Brief course description | The purpose of the discipline "Linux Operating System" is to teach students the basics of working and managing the Linux operating system. Upon completion of the course, students should have an understanding of the core concepts of Linux and be able to use it effectively in a variety of scenarios. |
| Expected learning outcomes | As a result of studying the discipline, the student must: know: - architecture of operating systems Windows 7/8, Linux, functional purpose of their logical components; - a set of basic commands and interfaces for operating systems Windows 7/8, Linux; be able to apply knowledge and understanding: - install Windows 7/8, Linux operating systems, configure and debug operating system components, connect standard and non-standard equipment; - use utilities and utilities for their intended purpose to ensure the safe operation of system and application software; |

| Description of discipline | |
|----------------------------------|--|
| Code of discipline | SFT6313 |
| Name of discipline | Mobile technologies and applications |
| Number of credits (ECTS) | 7 |
| Course, semester | 3, 6 |
| Department | CE |
| Prerequisites | Object-Oriented Programming |
| Postrequisites | Diploma project |
| Brief course description | Android programming concepts are reinforced through a set of thematic programming exercises that introduce these topics and incrementally allow the student to build a complex application; that is, programming labs form a set of components that collectively implement a continuous sensing application. The resulting phone app allows user to log their exercises (e.g., walks, runs) and display them on Google maps. |

| | |
|----------------------------|--|
| Expected learning outcomes | After successful completion of the course students will be able to: <ul style="list-style-type: none"> – characterize the architecture of mobile applications; – analyze the requirements of mobile applications; – design and develop mobile applications using one application development framework. |
|----------------------------|--|

| Description of discipline | |
|----------------------------------|--|
| Code of discipline | SFT6314 |
| Name of discipline | Full stack development |
| Number of credits (ECTS) | 5 |
| Course, semester | 4, 7 |
| Department | CE |
| Prerequisites | Web Technologies |
| Postrequisites | Diploma project |
| Brief course description | A full stack developer is an engineer who can handle all the work of databases, servers, systems engineering, and clients. Depending on the project, what customers need may be a mobile stack, a Web stack, or a native application stack. In fact, “full stack” refers to the collection of a series of technologies needed to complete a project. “Stack” refers to a collection of sub-modules. These software sub-modules or components combined together to achieve the established function while without the need for other modules. |
| Expected learning outcomes | After successful completion of the course students will be able to: <ul style="list-style-type: none"> – analyze the software requirements; – distribute the tasks between the stack components; – develop the stack components; – test the stack components. |

| Description of discipline | |
|----------------------------------|--|
| Code of discipline | SFT6328 |
| Name of discipline | Development of mobile applications on IOS |
| Number of credits (ECTS) | 7 |
| Course, semester | 3,5 |
| Department | CE |
| Prerequisites | Application Development Foundation |
| Postrequisites | Full Stack Development, Diploma project |
| Brief course description | The student will learn the features of databases and information assurance applications in operation systems iOS; will utilize enterprise information systems to support information security applications; to have basic skills in database administration of enterprise information systems. |
| Expected learning outcomes | - |

| Description of discipline | |
|----------------------------|---|
| Code of discipline | SFT6311 |
| Name of discipline | Front-end development |
| Number of credits (ECTS) | 7 |
| Course, semester | 3,6 |
| Department | CE |
| Prerequisites | Application Development Foundation |
| Postrequisites | Full Stack Development, Diploma project |
| Brief course description | In this course, students will study in detail the process of creating the client side of the site, namely the layout of the site template and the development of the user interface. |
| Expected learning outcomes | Upon successful completion, students will be able to: - create modern websites using HTML, CSS and JavaScript; - develop web applications from scratch; - write more efficient web code; |

| Description of discipline | |
|----------------------------|---|
| Code of discipline | MIN602 |
| Name of discipline | Minor2 |
| Number of credits (ECTS) | 5 |
| Course, semester | 3,5 |
| Department | CE |
| Prerequisites | - |
| Postrequisites | - |
| Brief course description | Additional educational program (minor) - a set of disciplines and (or) modules and other types of educational work, determined by students for study in order to form additional competencies |
| Expected learning outcomes | - |

| Description of discipline | |
|----------------------------|---|
| Code of discipline | MIN603 |
| Name of discipline | Minor3 |
| Number of credits (ECTS) | 5 |
| Course, semester | 3,5 |
| Department | CE |
| Prerequisites | - |
| Postrequisites | - |
| Brief course description | Additional educational program (minor) - a set of disciplines and (or) modules and other types of educational work, determined by students for study in order to form additional competencies |
| Expected learning outcomes | - |

| Description of discipline | |
|---------------------------|---------|
| Code of discipline | SFT6321 |

| | |
|----------------------------|--|
| Name of discipline | QA testing |
| Number of credits (ECTS) | 6 |
| Course, semester | 4,7 |
| Department | CE |
| Prerequisites | - |
| Postrequisites | Diploma project |
| Brief course description | This course includes theoretical and practical classes on the following topics: main types of testing; basics and classification of testing; testing principles; WEB-product testing; software development methodology; test design techniques; work with Requirements for the tester; compiling and working with checklists in practice; compiling and working with test cases in practice; compiling and working with bug reports in practice; compiling and working with test sets; work in the JIRA system, etc. |
| Expected learning outcomes | You will gain an understanding of the theory of testing. You will practice writing tests for a variety of quality intent, including code coverage, defect finding, and statistical testing. You will develop test plans to guide the testing stage of the software development lifecycle. You will create defect reports to provide transparency and understanding to supervisors, colleagues, and users. |

| Description of discipline | |
|----------------------------------|---|
| Code of discipline | SFT6319 |
| Name of discipline | Blockchain technology |
| Number of credits (ECTS) | 6 |
| Course, semester | 4, 7 |
| Department | CE |
| Prerequisites | - |
| Postrequisites | - |
| Brief course description | The Blockchain course is for those who want to learn more about blockchain technology and its applications. The course will look at how blockchain works, what its advantages and disadvantages are, what cryptocurrencies and tokens use blockchain, how to create and use smart contracts, and what are the examples of blockchain applications in various fields such as finance, logistics, medicine, etc. others |
| Expected learning outcomes | According to the goal and objectives of the course by the end of this course the students will be able to understand the technical underpinnings of blockchain technology, and write code for smart contracts and build blockchain networks. |

| Description of discipline | |
|----------------------------------|--|
| Code of discipline | SFT6315 |
| Name of discipline | DevOps |
| Number of credits (ECTS) | 7 |
| Course, semester | 4, 7 |
| Department | CE |
| Prerequisites | Introduction to Programming, Mathematical analysis |

| | |
|----------------------------|---|
| Postrequisites | Writing and defending a thesis, diploma project or preparing and passing a state exam, Applied robotics |
| Brief course description | The course examines the key concepts and principles of DevOps, organizational factors and automation tools in the development of software products using this method. |
| Expected learning outcomes | After completing the course, the students will be able to: <ul style="list-style-type: none"> - Define the key principles and concepts of DevOps, including automation, collaboration, and continuous integration and continuous delivery (CI/CD). - Develop effective communication and collaboration skills to bridge the gap between development and operations teams. - Learn how to set up and manage CI pipelines to automatically build and test code changes as they are committed to a version control system. - Understand the importance of integrating security into the DevOps process (DevSecOps) and learn about security practices and tools. - Develop skills in implementing automated testing practices, including unit testing, integration testing, and end-to-end testing. |

| Description of discipline | |
|----------------------------------|--|
| Code of discipline | SFT6323 |
| Name of discipline | Programming on the QT platform |
| Number of credits (ECTS) | 7 |
| Course, semester | 4, 7 |
| Department | CE |
| Prerequisites | Algorithmization and programming, Introduction to programming |
| Postrequisites | Course project |
| Brief course description | The course "Programming on the QT platform" is intended for those who want to master the development of cross-platform graphical applications in C ++ using QT - one of the most popular libraries for creating GUI applications. As part of the course, students will learn the basic concepts of QT, learn how to work with controls, create layouts and customize their appearance. |
| Expected learning outcomes | After completing the course, students can: <ul style="list-style-type: none"> - have a good understanding of the main components of the QT Framework, such as widgets, data models, events and signals, as well as other basic elements of the library; - ability to develop graphical user interfaces using various widgets and Qt components; - Skills in using integrated development environments such as Qt Creator to optimize the programming and debugging process; |

| Description of discipline | |
|----------------------------------|---|
| Code of discipline | SFT6376 |
| Name of discipline | Microsoft .NET Framework |
| Number of credits (ECTS) | 7 |
| Course, semester | 4, 7 |
| Department | CE |
| Prerequisites | C# programming language, VB.NET, introduction to SQL databases, |

| | |
|----------------------------|--|
| | web technologies |
| Postrequisites | Course project |
| Brief course description | <p>The discipline "Microsoft .NET Framework - Application Development" is included in the university educational program and is intended for students interested in developing software on the Microsoft .NET Framework platform. The course includes learning the basics of technologies and tools used to create modern applications that run on the .NET platform.</p> <p>Within this discipline, students learn the basics of programming on the .NET platform, including the programming languages C# and Visual Basic.NET, as well as the basics of working with the Visual Studio integrated development environment. Students also learn how to build and debug applications that use various .NET components such as Windows Forms, ASP.NET, ADO.NET, WPF (Windows Presentation Foundation) and others.</p> |
| Expected learning outcomes | <p>After completing the course, students will have:</p> <ul style="list-style-type: none">- Understanding of the architecture and core components of the .NET Framework, such as the Common Language Runtime (CLR), base classes and libraries;- Understanding of basic OOP concepts such as classes, inheritance, polymorphism and encapsulation;- Knowledge of the basics of using the Visual Studio integrated development environment, including creating projects, debugging and using tools; |