

Study subjects offered to students of the Erasmus exchange program in the 2024/2025 academic year

Autumn/Fall semester

Faculty of Science and Engineering

Computer Science (Bachelor)

1st study semester (1st year students)

Code	Area	Title	ECTS
DatZPB16	Computer Science/ Programming languages and systems	Module "Software development" I: Programming	7,5
ETelPA01	Electronics and telecommunications/ Electronic elements	Introduction to Electronics	3
ETehPA10	Electronics and electrical engineering/ Systems analysis, modeling and design	Module "Software development" I: Introduction to Software Engineering	3
Mate3025	Mathematics	Linear Algebra	3
MatePB56	Mathematics	Discrete Mathematics I	6
			22,5

2nd study semester (1st year students)

ValoPB81	Linguistics/ Applied linguistics	Technical English	3
DatZPB36	Computer Science/ Programming languages and systems	Module "Software development II": Algorithms and data structures	3
ETehPA19	Computer Science/ Data processing systems and computer networks	Module "Software development II": Web programming	3
DatZPB19	Computer Science/ Programming languages and systems	Module "Software development II": Programming language Python	3
DatZPA75	Electronics and electrical engineering/ Electronic technologies and automation	Internet of Things I	3
DatZPB08	Electronics and electrical engineering/ Data processing systems and computer networks	Module "Software development II": Introduction to data processing systems	3
MatePB59	Mathematics	Mathematics II	3
MatePB60	Mathematics	Probabilities and Mathematical Statistics	3
			24

*Available only in Spring semester

3rd study semester (2nd year students)

ETehPA27	Electronics and electrical engineering/ Systems analysis, modeling and design	Module "Software engineering I": Software Project Management	3
DatZ2003	Computer Science/ Data processing systems and computer networks	Module "Software engineering I": Databases	6
DatZP077	Computer Science/ Programming languages and systems	Module "Software engineering I": Object-Oriented Programming	3
MatePB59	Mathematics	Mathematics II	3
MatePB60	Mathematics	Probabilities and Mathematical Statistics	3
			18

*Available only in Fall semester

4th study semester (2nd year students)

DatZ3893	Computer Science/ Programming languages and systems	Programming Languages	3
DatZPA78	Computer Science/ Data processing systems and computer networks	Data processing systems	3
DatZ3929	Computer Science and Informatics	Operating Systems	3
ETehPA49	Electronics and electrical engineering/ Systems analysis, modeling and design	Module "Software engineering II": Quality of software projects and data	3
Inft2001	Electronics and electrical engineering/ Systems analysis, modeling and design	Module "Software engineering II": Information systems analysis and design	6
DatZ6011	Computer Science and Informatics	Module "Software engineering II": Software testing	3
DatZPB18	Computer Science/ Programming languages and systems	Mobile application development	3
			24

5th study semester (3rd year students)

ETehPA12	Electronics and electrical engineering/ Data processing systems and computer networks	Module "Computer's systems and network": Internet technologies	3
DatZPB17	Electronics and electrical engineering/ Data processing systems and computer networks	Module "Computer's systems and network": Computer Networks	6
ETehPA13	Electronics and electrical engineering/ Data processing systems and computer	Module "Computer's systems and network": Cloud Computing	3
			12

6th study semester (3rd year students)

DatZPA22	Computer Science/ Programming languages and systems	Java Programming	3
ETehPA14	Electronics and electrical engineering/ Systems analysis, modeling and design	Module "Software engineering III": Requirements Engineering	3
ETehPA15	Electronics and electrical engineering/ Systems analysis, modeling and design	Module "Software engineering III": Software system architecture and design	3
ETehPA16	Electronics and electrical engineering/ Systems analysis, modeling and design	Module "Software engineering III": Human-computer interaction	3
			12

IT (Master)

1st study semester (1st year students)

Code	Area	Title	ECTS
InfTPA05	Information technology	Total Quality Managementa	6
ETehPA28	Information technology / Systems analysis, modeling and design	IT Project Management	3
ETehPA33	Information technology / Systems analysis, modeling and design	IT Project Work Management	3
ETehPA29	Information technology / Systems analysis, modeling and design	Big Data	3
ETehPA30	Information technology / Systems analysis, modeling and design	Advanced software development methodologies	3
ETehPA31	Information technology / Systems analysis, modeling and design	User behavior aspects	3
ETehPA32	Information technology / Systems analysis, modeling and design	Software systems prototyping	3
ETehPA34	Information technology	Business Analysis	3
			27

***If chosen from diferent semesters, course times can overlap**

Descriptions

Subject title	Module: Software development I Programming
Level (Basic studies, master)	Basic studies

Subject aim	Develop student's algorithmic thinking and create an understanding of programming.
Subject content	In this course students are introduced to the basic constructs of algorithms and programming languages (conditions, loops, arrays, etc.). The theoretical knowledge acquired during the course is strengthened by creating solutions to various tasks in the programming language C++.
Expected results	<p>Completing the course students will:</p> <ul style="list-style-type: none"> • know the programming language C ++; • know the basics of software development; • know the basic data structures and design elements of algorithms; • be able to write software code encode and to debug it; • be able to use software development tools; • be able to analyse the program code; • be able to implement the user interface; • be able to write the software code analysing the input and output data; • be able to write the software code designing algorithms; • be able to write the software code reading and analysing the code written by others; • be able to write the software code designing user interface; • be able to write the software code debugging it and implementing its unit tests; • be able to write the software code analysing the runtime and managing its optimisation; • be able to write the software code documenting it.
Requirements to gain ECTS	The final assessment is calculated as a relation between the sum of student's obtained points and the maximum amount of points that a student can obtain by completing and presenting all homework during the semester, successful completion of three tests and group work, if the grade is 7 or higher. In case of a lower grade, student must pass the exam which consists of two parts - theoretical and practical.

Subject title	Introduction to electronics
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To provide in-depth knowledge of the regularities of electrical engineering and electronics and the processes required to know the principles of operation of equipment.
Subject content	<p>Boolean algebra. Formal representation of logic functions in two-level logic (SoP and PoS). Analysis and synthesis of combinatorial digital systems. Simplification of the logic function by using Karno maps.</p> <p>Logical elements. Analysis and synthesis of sequential logic systems. Situation coding and simplification. Synthesis of synchronous sequential circuits with different triggers.</p> <p>Current and voltage. Kirchhoff's laws. Methods of analysis of linear static circuits. Resistance, capacitance, inductance. Diodes. Bipolar transistor. MOSFET transistor as a switch. Operational amplifiers.</p>
Expected results	<p>After completing the course students:</p> <ul style="list-style-type: none"> • will know the principles of operation of electronic equipment used at the level of understanding; • will be able to analyse the operating modes of the scheme elements and choose adequate means to solve tasks; • will be able to use virtual simulation software;

	<ul style="list-style-type: none"> • will be able to determine the type and location of possible damage.
Requirements to gain ECTS	Successfully completed homework assignments in time, participation in seminars, laboratory works, must pass an exam

Subject title	Module: Software development I Introduction to Software Engineering
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	Develop a basic understanding about IT project development, versioning, software requirement definition and documentation.
Subject content	In the theoretical part of the course, students gain knowledge and understanding of IT project development, software development life cycle, methodologies used in IT projects. In the practical part, students acquire the skills to use and use in practice the project management tool JIRA or any other similar project management tool. Students gain knowledge about Git, GitHub and will know how to use basic Git commands to be able to use them in practice, on a daily basis.
Expected results	<p>Upon course completion students:</p> <ul style="list-style-type: none"> • have knowledge about IT project definitions and their development models; • have knowledge of software lifecycle; • have knowledge of the activities in different states of development lifecycle; • have knowledge of the definitions of user stories and requirements; • have knowledge of project management tools; <p>have knowledge of to use project management tools for project management;</p> <ul style="list-style-type: none"> • have knowledge of software versioning; • have knowledge of Git tool; • have knowledge of GitHub tool; • have ability to define requirements and write user stories; • have ability to use Git or other comparable versioning tool; • have ability to basic commands of Git to clone, merge code, create merge request, create branches and move among them; • have ability to create projects in project management tool Jira or other comparable project management tool; • have ability to create work tasks in accordance to project requirements in the project management tool Jira or other comparable project management tool.
Requirements to gain ECTS	Practical assignments submitted and presented within certain deadlines. (60%) Passed test (40%)

Subject title	English for engineers
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	The course aims at developing a) scientific English communication skills (speaking, reading, writing) using natural & engineering science as the working topic; b) skills on research papers writing and presentation in English.
Subject content	An advanced course in English targeted on non-native English speakers who are studying natural sciences and engineering at Bachelor`s level. Course is related to development of scientific English communication skills – both oral and writing; emphasis is put on practical project work – preparing research paper.

Expected results	<p>Knowledge on English grammar, scientific terminology, structure & style of research articles written in English</p> <p>Knowledge on research papers` quality criteria as well as on algorithms for successful paper in English</p> <p>Technical English reading (understanding), writing, oral communication / presentation and editorial skills</p> <p>Competencies: advanced level of technical English proficiency; critical & creative English writing abilities; English Rhetoric`s competency</p>
Requirements to gain ECTS	Students are expected to participate in all the course activities (students are not allowed to miss more than 20% of the class time) and to complete all the course assignments (see “Students` Independent Work”: Course examination comprises these 4 assignments). Every student must be an active learner who is fully responsible for his/her own English fluency and technical English competencies development.

Subject title	Module: Software development II
	Algorithms and Data Structures
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	Create an understanding about commonly used dynamic data structures, their usage in C++ programming language.
Subject content	In this course students are introduced to the most common dynamic data structures (linked list, queue, stack, tree), their manipulation algorithms. The theoretical knowledge is reinforced by applying it to practical tasks in C++ programming language.
Expected results	<p>Upon the course completion, students:</p> <ul style="list-style-type: none"> ● have knowledge of commonly used data structures and their manipulation algorithms; ● have ability to code, by constructing algorithms; ● have ability to code, by reading and analysing third party code; ● have ability to design and construct algorithms; ● have ability to use good programming style; ● have ability to code and debug simple programs; ● have ability to complete assignments individually.
Requirements to gain ECTS	<p>Independently created, within deadlines submitted and reviewed practical assignments (40%)</p> <p>Successfully passed two tests (50% completion is considered a pass) (40%)</p> <p>Successfully passed exam (50% completion is considered a pass) at the end of semester (20%)</p>

Subject title	Module: Software development II.
	Web programming
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To promote the student's understanding of web technologies and web page development fundamentals and examples of good practice
Subject content	In the current course, students are introduced to the basics of web technologies (client-side application, server-side application, HTML, CSS, Javascript, etc.). The theoretical knowledge acquired during the course is reinforced by developing a web page.
Expected results	Completing the course students will:

	<p>know about Internet technologies and languages HTML, CSS and Javascript;</p> <ul style="list-style-type: none"> <input type="checkbox"/> know how to select the means and tools for the development of client-side web pages that are suitable for the solution of the given problem; <input type="checkbox"/> know how to use software development tools for the development of client-side web pages; <input type="checkbox"/> know how to implement user interfaces on web pages; <input type="checkbox"/> know how to use a good programming style; <input type="checkbox"/> be able to write program code by analyzing input and output data; <input type="checkbox"/> be able to write program code and design user interface
Requirements to gain ECTS	<p>Independently completed tasks, submitted and defended their solutions within the specified deadlines (40%)</p> <p>Successful completion of two tests (40%)</p> <p>Successfully passed the final test 20%</p>

Subject title	Module: Software development II. Programming language Python
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To provide knowledge of the programming language Python and improve students' programming skills.
Subject content	During the current course, students learn the programming language Python as a second programming language. In the classes, students improve their knowledge and skills for using the basic constructions of algorithms (branches, cycles, functions, etc.) and data structures (lists, queues, stacks, binary trees, etc.) to solve various programming tasks. Separate lessons are devoted to data visualization using Python.
Expected results	<p>Completing the course students will:</p> <ul style="list-style-type: none"> <input type="checkbox"/> know the programming language Python; <input type="checkbox"/> know the basic constructions of data structures and algorithms and their implementation in Python <input type="checkbox"/> know how to write program code and debug programs in Python; <input type="checkbox"/> be able to analyze Python program code; <input type="checkbox"/> know how to select suitable tools for the solution of the given tasks; <input type="checkbox"/> know how to use information search and selection tools; <input type="checkbox"/> be able to design, construct and describe algorithms; <input type="checkbox"/> be able to design by analyzing various technical solutions and selecting the most suitable one <input type="checkbox"/> be able to write program code by analyzing input and output data; <input type="checkbox"/> be able to write program code, constructing algorithms; <input type="checkbox"/> be able to write program code, reading and analyzing program texts; <input type="checkbox"/> be able to write program code and debug programs; <input type="checkbox"/> be able to write program code, analyzing the execution time of the program and optimizing it; <input type="checkbox"/> be able to write program code by documenting the code.
Requirements to gain ECTS	The final grade of the current course has calculated as average rating of the solution of all independent tasks

Subject title	Internet of Things I
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To give students an idea of mobile and stationary systems for data acquisition, processing, transmission and storage.

Subject content	<p>The course provides knowledge and skills in the development of controlled and controllable devices, data acquisition and transmission.</p> <p>Students make sensor connections based on the programmable controller Arduino Uno, programming takes place using the programming language C ++; organizes data transmission via WiFi and Bluetooth modules, gains experience in program library applications for connections of various actuators (LEDs, LED strings, DC and stepper motors, etc.). At the end of the semester, students develop a prototype of a controlled device with organized data acquisition, transmission, processing, and storage.</p>
Expected results	<p>Students will know the basic principles of creating controllable and manageable devices.</p> <p>will be able to create sensor connections with several types of programmable controllers, organize data transmission channels (WiFi, Bluetooth, LoRa, ZigBee, etc.), send data to a remote server and receive processed information.</p> <p>knows the application of program libraries for the corresponding controller, software structure, development of work functions (subprograms), program debugging.</p>
Requirements to gain ECTS	80% attendance, submissions submitted and defended.

Subject title	Modulis: Software development II. Introduction to data processing systems
Level	Basic studies
<i>(Basic studies, master)</i>	
Subject aim	To provide basic knowledge and understanding of databases, the basic principles of database management systems, database technologies and the basics of the SQL language. To acquire basic skills in using database management systems, designing databases and creating SQL queries, data processing using programming languages Python, Javascript, and PHP.
Subject content	The aim of the course is to provide knowledge and understanding of database access technologies, basic principles of database management systems and SQL language. The course provides basic skills of writing SQL queries, designing databases and using database management systems, as well as data processing using programming languages Python, JavaScript and PHP.
Expected results	<p>Completing the course students will:</p> <ul style="list-style-type: none"> <input type="checkbox"/> be able to analyze and choose the most appropriate database management system for database creation; <input type="checkbox"/> be able to design databases and implement them using the selected database management system; <input type="checkbox"/> be able to write queries according to requirements; <input type="checkbox"/> be able to use database technologies; <input type="checkbox"/> know how to use data request languages; <input type="checkbox"/> be able to compile data processing programs using Python and PHP; <input type="checkbox"/> be able to develop an applications with a database; <input type="checkbox"/> know database technologies.
Requirements to gain ECTS	Class attendance 75%. Successfully passed tests, successfully completed all assignments.

Subject title	Module: Software engineering I
----------------------	---------------------------------------

	Software Project Management
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To develop understanding of the software project development process, to develop practical skills in preparing a software project plan, to develop understanding of the management of the software project, the role of the project leader and the team in the implementation of the project, to develop an understanding of the specifics of computer systems and software development projects
Subject content	The objective of the course is to create understanding on software project development, to develop skills necessary for software project management and promote skills of team-working
Expected results	Completing the course students will: <ul style="list-style-type: none"> • know about a software development project management; • be able to participate in project management • be able to work in a team and manage other team members; • be able to plan the necessary activities and to define their priorities; • be able to design software system according to the software design description; • be able to design software system analyzing different technological solutions and selecting the most suitable one; • be able to deliver the software system customizing its environment; • be able to deliver the software system providing user consultation; • be able to plan software project estimating the development effort and time; • be able to plan software project through individual work planning and control; • be able to plan software project participating in discussions of the implementation progress of the project; • be able to plan software project developing programming guidelines
Requirements to gain ECTS	Timely individual work assignment submissions. Creating and presentation of software project management plan.

Subject title	Module: Software Engineering I Databases
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To provide knowledge and understanding of databases, database management systems operating framework, database technology and SQL language. To acquire skills in the administration and use of database management systems, database design and in-depth creation of SQL queries. Gain knowledge and understanding of access methods to databases. Acquire skills for storing and retrieving data using different types of database access. To acquire skills to create a client interface for data processing in databases.
Subject content	The aim of the course is to provide knowledge and understanding of database access technologies, advanced principles of database management systems and

	SQL language. The course acquires skills in designing databases, creating SQL queries and administering database management systems.
Expected results	<p>After completing the course, students:</p> <ul style="list-style-type: none"> • Will be able to analyze and choose the most appropriate database management system for database creation. • Will be able to design databases and implement them in the selected database management system. • Will be able to compose queries according to requirements. • Will be able to use database technologies. • Will be able to perform user administration in the DBMS system. • Will be able to administer access rights in the DVBS system. • Will be able to index data in both new and existing databases. • Will be able to perform database maintenance and administration. • Will be able to maintain and administer database tables. • Knows how to use data request languages. • Will be able to analyze and choose the most appropriate access technology for the database. • Will be able to develop an app with a database. • Knows database technologies. <p>• • Knows how to use data request languages.</p>
Requirements to gain ECTS	Class attendance 75%. Successfully passed tests, successfully completed all practical tasks.

Subject title	Module: Software engineering I. Object-Oriented Programming
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To provide knowledge on object-oriented programming common features and paradigms of object-oriented programming and train skills to develop software applications applying object-oriented approach
Subject content	The course provides knowledge about the object-oriented programming approach and the main principles of development. The theoretical part of the course deals with the basic concepts of object-oriented programming - inheritance, encapsulation and polymorphism. Insights into friend functions, abstract classes, and templates are also given. Practical lessons are dedicated to developing programming skills in C++ and C# and system modeling skills in UML.
Expected results	<p>Completing the course students will:</p> <ul style="list-style-type: none"> <input type="checkbox"/> know the basic principles of object-oriented programming; <input type="checkbox"/> know how to code and debug programs; <input type="checkbox"/> know how to write program code using the object-oriented approach; <input type="checkbox"/> be able to apply design schemes and diagrams; <input type="checkbox"/> know how to use a good programming style; <input type="checkbox"/> know how to select suitable tools for solution of the given task; <input type="checkbox"/> know how to do work independently; <input type="checkbox"/> be able to write program code, reading and analyzing program code written by others; <input type="checkbox"/> be able to write program code, reading and understanding software design descriptions; <input type="checkbox"/> be able to write program code according to design and coding guidelines; <input type="checkbox"/> be able to write program code and its documentation; <input type="checkbox"/> be able to design, create and describe software architecture;

	<input type="checkbox"/> be able to design software system, creating an implementation model (i.e., hierarchy of classes and/or functions); <input type="checkbox"/> be able to design, preparing a software design description document; <input type="checkbox"/> be able to take part of projects where the object-oriented programming approach has been used.
Requirements to gain ECTS	All tasks of independent work must be completed; UML models of the software system must be built, and C++ and C# programs must be written for them

Subject title	Programming Languages
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	The aim of the course is to provide knowledge on different programming languages, their common and different features and components
Subject content	to provide knowledge on different programming languages, their common and different features and components. The content of the current course consists of the following topics: syntax of programming languages, context-free grammars, finite automata and their applications in theory of formal languages, imperative, functional and logical programming languages, and regular expressions.
Expected results	<p>Completing the course students will:</p> <p>know about programming languages, their main components and structures;</p> <input type="checkbox"/> know how to select suitable programming languages and tools for solving the given problem; <input type="checkbox"/> be able to write the code by analyzing input and output data; <input type="checkbox"/> be able to design software system, analyzing various technical solutions and selecting the most suitable one
Requirements to gain ECTS	Successful completion of all assignments, Exam

Subject title	Data processing systems
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	The aim of the course is to provide knowledge about the possibilities of data extraction, storing, selection and processing.
Subject content	The aim of the course is to provide knowledge about the possibilities of data extraction, storing, selection and processing.. During the course, students learn about statistical methods of data processing and train the skills how to implement these methods in programming and database tasks.
Expected results	<p>Completing the course students will:</p> <input type="checkbox"/> be able to analyze the problem and choose the most suitable data processing system and techniques; <input type="checkbox"/> be able to process data stored in different storage formats; <input type="checkbox"/> be able to design data processing and analysis systems and implement them by the selected software development system; <input type="checkbox"/> be able to compile data processing queries according to requirements; <input type="checkbox"/> be able to use data processing and analysis technologies.
Requirements to gain ECTS	<p>Successfully passed homework tasks.</p> <p>It is necessary to get a positive evaluation of the solutions of each task</p>

Subject title	Operating Systems
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	<p>The goal of the course is to provide knowledge about operating systems concepts, processing and management basic principles; improve student's skills and abilities in practical operating system problem solving.</p> <p>The aim of the course is to provide knowledge about principles of computer processes and techniques in hardware level, teach about hardware programming principles and skills.</p>
Subject content	<p>The course gives understanding about operating systems concepts, architecture and processing principles: process management, file systems, input/output organization and memory management. The course develops knowledge and skills in Linux operating system administration.</p> <p>The course offers to obtain understanding about architecture and processing principles of computer processors, about input/output organization, random access memory and external memory organization and their types. Practical skills are developed in machine-oriented language Assembles programming.</p>
Expected results	<p>After course completion students:</p> <ul style="list-style-type: none"> • will have basic understanding about operating systems classification and their usage, operating systems concepts, processing principles and main algorithms; • will be able to use Linux operating system; • will be able to perform basic administration functions in Linux operating system; • will have basic understanding about computer architecture and functioning • will have skills in coding and debugging in hardware-oriented Assembler programming language
Requirements to gain ECTS	Successful completion of theoretical test works and practical tasks.

Subject title	Module: Software engineering II Software project and data quality
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	<p>To acquire knowledge and understanding of the principles and methods of software project and data quality assurance. Gain knowledge about the basic principles of software project and data quality assurance, its goals and understand the basic concepts of quality assurance.</p> <p>To acquire skills in the organization of the software project and data quality assurance process and the preparation of the corresponding documentation. To acquire skills in project quality assurance and data quality assessment.</p>
Subject content	The course provides knowledge and understanding of software project quality assurance and data quality evaluation principles and methods. Students acquire the skills for software project quality process organization, and documentation; skills for data quality evaluation.
Expected results	<p>After completing the course students:</p> <ul style="list-style-type: none"> • Will be able to analyze and choose the most appropriate quality assurance and evaluation methods. • Will be able to prepare quality assurance documentation

	<ul style="list-style-type: none"> • Will be able to prepare user documentation by familiarizing himself with user transaction terminology. • Will be able to prepare user documentation by writing and designing user documentation text. • Will be able to prepare user documentation. • Will be able to plan a software project by developing programming guidelines. • Will know how to draw up business documents. • Will know how to use information technology industry standards. • Will know how to use a good programming style. • Will be able to manage the national language. • Knows the principles of software development project management. • Knows software engineering. • Will be able to prepare a software testing review document. • Will know how to draw up business documents. • Will know how to use information technology industry standards. • Will be able to manage the national language. • Knows the principles of software development project management. • Knows software engineering.
Requirements to gain ECTS	<p>• Knows professional terms in the national language and in a foreign language.</p> <p>Successfully developed and presented software documentation: Software Requirements Specification (PPS), Software Design Description (PPA) and User Documentation.</p> <p>Successfully developed and presented project management plan, including testing procedure, configuration management and quality assurance measures in the project.</p> <p>Successfully performed testing and presented testing documentation (test examples, testing log, problem reports, problem message registration log, testing summary) (see section "Student's independent work", requirements</p>

Subject title	Module: Software engineering II Information systems analysis and design
Level (Basic studies, master)	Basic studies
Subject aim	To give knowledge and skills in software engineering analysis and design and give insight on current trends in industry in these topics
Subject content	The course provides knowledge and skills in software engineering in general, requirements analysis and design including design patterns, in industry adapted practice methods and principles for development of units and system architecture. Alternative design options are compared to select most appropriate.
Expected results	<p>Completing the course, the student will:</p> <ul style="list-style-type: none"> • will have basic understanding in subfields of software engineering: system analysis and design • will be able to choose the right tools to solve concrete task and perform software design task; • will be able to design system by reading software requirements description; • will be able to design by constructing and describing algorithms; • will be able to design by constructing and describing software architecture; • will be able to design analysing various technical solutions and choosing the best ones; • will be able to design making conceptual and physical data model; • will be able to design describing software design description; • will be able to design software with good user interface in mind.
Requirements to gain ECTS	Student must do all individual work tasks and pass exam.

Subject title	Module: Software engineering II Software Testing
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To give knowledge and understanding of software testing and its role in the software development life cycle. To give knowledge and understanding about the test case creation process, defects finding, the understand how to implement testing strategies in information technology projects, establishing testing documentation, reporting test results. Acquire the skills to use and use in practice the TestRail testing tool and the project management tool JIRA.
Subject content	In the theoretical part of the course, students acquire knowledge and understanding of the basis for software testing and their role in the life cycle of software development, testing types and techniques, establishing a testing strategy for information technology projects, establishing testing documentation, In the practical part, students acquire the skills to use and use in practice the testing tool TestRail and project management tool JIRA. Skills to build testing examples, test hikes, problem reporting.
Expected results	Completing the course, the student will: <ul style="list-style-type: none"> <input type="checkbox"/> Will know what software testing is and why testing is needed; <input type="checkbox"/> Will know the role of testing in software development lyfe cycle; <input type="checkbox"/> Will know what testing and quality is; <input type="checkbox"/> Will know how much testing is sufficient; <input type="checkbox"/> Will know what software tester certification is; <input type="checkbox"/> Will know what software testing standards are; <input type="checkbox"/> Will know what debugging is; <input type="checkbox"/> Will know what seven test principles are; <input type="checkbox"/> Will know what a set of testing activities are; <input type="checkbox"/> Will know what the life cycle of software development is; <input type="checkbox"/> Will know what software development models (V-model, Waterfall model, Iterative incremental model); <input type="checkbox"/> Will know what Agile, Scrum, Kanban methodology is; <input type="checkbox"/> Will know what test levels are, i.e. component testing, integration testing, system testing, acceptance testing; <input type="checkbox"/> Will know what functional testing is; <input type="checkbox"/> Will know what non-functional testing is; <input type="checkbox"/> Will know what white box testing and its techniques are; <input type="checkbox"/> Will know what black-box testing and its techniques are; <input type="checkbox"/> Will know what retesting and regression testing is; <input type="checkbox"/> Will know what maintenance testing is; <input type="checkbox"/> Will know what static testing and its techniques are; <input type="checkbox"/> Will know what dynamic testing and its techniques are; <input type="checkbox"/> Will know what experience-based testing techniques are; <input type="checkbox"/> Will know what testing management is; <input type="checkbox"/> Will know what is an example of testing, passage of tests, problem reporting; <input type="checkbox"/> Will know what the life cycle of the defect is. <input type="checkbox"/> Will be able to create test cases; <input type="checkbox"/> Will be able to design and execute test runs; <input type="checkbox"/> Will be able to create bug reports; <input type="checkbox"/> Will be able to use the TestRail test tool; <input type="checkbox"/> Will be able to use THE JIRA Project Management Tool; <input type="checkbox"/> Will be able to create testing documentation; <input type="checkbox"/> Will be able to use the standards set by the computer science industry and the Latvian State in relation to software testing; <input type="checkbox"/> Will be able to apply traceability between initial requirements in the IT project, test cases and bug reports.

Requirements to gain ECTS	The theoretical knowledge test accounts for 30% of the final assessment, the other 70% being the practical work with the deliverable and presentation. The traversal final work shall be composed of a deliverable on test cases, executed test runs, the test summary, the bug reports, traceability.
----------------------------------	--

Subject title	Mobile application development
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To provide knowledge on creating mobile applications using the Android mobile operating system.
Subject content	Introduction of creating mobile applications. Defining and using JAVA class. Practical application development. Creating interface based on XML. Remote connection development
Expected results	After completion of given course, the students will: <ul style="list-style-type: none"> • Have basic knowledge about creating Android applications • Know major principles of Android operating system, to correctly create applications • Have basic knowledge about JAVA programming language • Be able to solve given tasks • Be able to read Android specific Java programming code • Be able to create interface based on XML language • Have additional knowledge about object-oriented programming • Have basic understanding of programming language - Kotlin
Requirements to gain ECTS	End of semester average mark, which is based on summing individual tasks, two individually gradable tasks, if average mark is above 7, If not there is exam

Subject title	Module: Computer systems and computer networks Internet technologies
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	Give knowledge about computer network technologies and increase skills about computer network administration and shared computer system creation.
Subject content	Practically learns how to install and configure a server in a virtual machine. Being able to install PHP, MySql and Apache or Nginx software environments to create remotely accessible websites using a linux operating system. Be able to install and configure a Python programming language server to be used as a website maintenance service.
Expected results	After course student will be able: <ol style="list-style-type: none"> 1. Will know basic usage of internet technologies 2. Will understand computer network technologies. 3. Will be able fulfill different tasks and choose corresponding solutions, 4. Will be able to configurate work place and work tools. 5. Able to use software development tools 6. Will know basic OS classification and use case 7. Will know basic computer system structure and functional abilities 8. Will be able to configurate Linux environment, using newest technologies 9. Will be able configure cloud-based services using terminal 10. Will be able to configurate and setup webservices to create public websites
Requirements to gain ECTS	Attendance in seminar, positive mark about individual work, and laboratory work.

Subject title	Module: Computer systems and network Computer Networks
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	To provide knowledge about the computer networks structure, its basic operation principles, technology and develop skills in computer network administration and maintenance. Prepare students for the MTCNA - MikroTik Certified Network Associate certification exam.
Subject content	The course introduces students to the principles of computer networks. It is based on computer network terminology, OSI and TCP/IP models and the implementation of protocols, routers, router administration, IP addressing and other computer networking standards. The study course includes the topic required for MTCNA - MikroTik Certified Network Associate certification, which is divided into 9 modules. Each of the modules contains a theory lesson and several practical works using MikroTik RouteBOARD computer network devices. At the end of the course, students take the MTCNA - MikroTik Certified Network Associate certification exam.
Expected results	<p>Completing the course students will:</p> <ul style="list-style-type: none"> • be able to choose adequate means to solve tasks; • be able to configure the workplace and tools; • be able to use software development tools; • have an understanding level of computer network technology; • know the level of use of Internet technologies; • Have competences to deliver the software, providing consultations during the delivery of the software. • know OSI and TCP IP models, protocols for the layers included and their usage; • know about IP addressing and to be able to identify subnets by the given IP address; • understand the classification and concepts of computer networks division and the importance of their building; • understand network management principles, meaning and possibilities; • be able to perform network management; • be able to install and configure network hardware, i.e., a network adapter (PCI, USB, wireless) and router; • be able to create a project for network installation or upgrade; • have competences to design the computer system analysing various technological solutions and selecting the most suitable one; • be able to configure, manage and do basic troubleshooting of a MikroTik RouterOS device; • be able to provide basic services to clients using MikroTik RouterOS device.
Requirements to gain ECTS	Student must do all individual work tasks. Students take the MTCNA Certification Exam.

Subject title	Module: Computer systems and network Cloud Computing
Level <i>(Basic studies, master)</i>	Basic studies

Subject aim	To create an understanding of the cloud computing architecture, its system design principles, their security and service and development models, as well as to provide knowledge about the application of cloud computing solutions in software development projects.
Subject content	The current course provides knowledge on Cloud Computing (CC) architecture, design and maintenance of CC systems, and cloud services. During the course, the students will learn how to administrate IaaS (Infrastructure-as-a-Service) system. Students will practically learn how to use CC systems in software development projects.
Expected results	Completing the course, the student will: <ul style="list-style-type: none"> 34. be able to choose adequate means to solve tasks; 35. be able to configure the workplace and tools; 36. be able to use software development tools; 37. have an understanding level of computer network technology; 38. know the level of use of Internet technologies; 39. know on cloud computing project management tasks and their solutions; 40. be able to estimate cloud computing development efforts; 41. be able to develop a cloud computing project configuration management plan; 42. be able to evaluate cloud computing project status; 43. be able to manage cloud computing project risks; 44. be able to argue personal view on applications of cloud computing services for management in software development projects; 45. be able to search, analyse, and concludes the information that is essential for development and maintenance of cloud computing systems.
Requirements to gain ECTS	Students need to complete the practical work assigned during the lessons. Students need a positive mark in the report and exam test.

Subject title	Module: Software engineering III. Enterprise resource planning
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	Create an understanding of IT applications in the management of company resources, the main principles of their operation, the advantages and problems of implementation, maintenance and use
Subject content	The aim of the course is to create an understanding of IT applications in enterprise resource management, the basic principles of their operation, the advantages and problems of implementation, maintenance and use.
Expected results	Completing the course students will: <ul style="list-style-type: none"> • know about business and management information systems; • be able to analyze alternative solutions; • know how to use office software; • be able to describe processes and account the working time used; • be able to prepare a report on the used resources; • be able to identify problems, set priorities and make decisions; • be able to perform a cost-benefit analysis; • be able to draw up a continuous operation plan; • be able to conduct project progress inspections; • be able to inspect deliveries; • be able to control the implementation of project plans; • be able to assess the customer's satisfaction
Requirements to gain ECTS	Successful completion of all assignments, Exam

Subject title	Java programming
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	Provide knowledge and skills in java programming language and show how to use some of the java supporting tools
Subject content	The course offers the knowledge about Java programming language and its capabilities in object oriented programming, multithreading support, generic for data types, etc. Students are introduced to some Java development tools. Skills are strengthened by solving various practical programming tasks.
Expected results	After successful completion of the course students will know the basic of Java programming language and its capabilities. They will be able to solve practical tasks by using Java programming language.
Requirements to gain ECTS	Fulfilment of practical tasks, passing of the final test.

Subject title	Module: Software engineering III Requirements Engineering
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	Provide knowledge of requirements engineering processes and models; requirements quality assurance; requirements modelling and management; Learn requirements modelling and management techniques and tools
Subject content	In the theoretical part, students will gain knowledge about requirements engineering processes and models; requirements quality assurance methods and requirements modeling and management methods; In the practical part, students will learn requirements modeling and management tools. Will be able to develop a requirements module for the Study Project software.
Expected results	Completing the course students will: <ul style="list-style-type: none"> <input type="checkbox"/> know requirement engineering processes and models; <input type="checkbox"/> know requirement quality assurance methods; <input type="checkbox"/> know requirement modeling and management methods; <input type="checkbox"/> be able to use requirement modeling and management tools <input type="checkbox"/> be able to prepare requirement model for software
Requirements to gain ECTS	Prepared and presented a report on current methods and tools of requirements engineering. (30 %). The Software Requirements Model of the Study Project was developed (70%)

Subject title	Module: Software engineering III Software system architecture and design
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	Create an understanding of software system architecture and its documentation
Subject content	The aim of the course is to create an understanding of software system architecture and its documentation. The course improves modeling skills in designing software systems using the modeling language UML. Certain tasks of the practical work are related to the tasks of the course "Study project (computer systems)".
Expected results	After completing the course, students will: <ul style="list-style-type: none"> <input type="checkbox"/> know the software architecture at the level of understanding; <input type="checkbox"/> know the unified modelling language (UML) diagrams at the level of use;

	<input type="checkbox"/> know the methods and techniques of recording and visualizing software systems and algorithms; <input type="checkbox"/> know the system design stages; <input type="checkbox"/> know the methods of process description visualization; <input type="checkbox"/> know the Unified Modelling Language (UML); <input type="checkbox"/> be able to write software documentation; <input type="checkbox"/> be able to document the architecture and design of the software system; <input type="checkbox"/> be able to develop a system model using the common modeling language; <input type="checkbox"/> be able to read and analyze visualized process descriptions; <input type="checkbox"/> be able to navigate the stages of system design while working in a team.
Requirements to gain ECTS	Successful completion of all course assignments, submit the first version of the introduction to the bachelor's thesis

Subject title	Module: Software engineering III Human-computer interaction
Level <i>(Basic studies, master)</i>	Basic studies
Subject aim	Create an understanding of UX (user experience) and its core principles and applications in UI (user interface) design. Apply gained theoretical knowledge by building a high fidelity software prototype.
Subject content	The "User Experience Design" course should: <ul style="list-style-type: none"> ● give the necessary knowledge and understanding in the field of development of user-centred software applications, including development with the most commonly used software frameworks, evaluation and testing of their usability; ● put knowledge and skills into practice by working with prototyping software; ● expanded knowledge and formed new skills of students, including approaches to conduct user research and interviews, and conduct user testing.
Expected results	Upon the course completion, students: <ul style="list-style-type: none"> ● have knowledge of user centric approach and its application in user experience design and user interface design; ● have knowledge of user experience design principles; ● have knowledge of and ability to design efficient communication with the end user; ● have knowledge of common performance optimization methods; ● have knowledge of different prototyping methods; ● have ability to design user interface; ● have ability to design in-built user assistance; ● have ability to design and create interface wireframes; ● have ability to design and create high fidelity prototypes; ● have ability to analyse and evaluate usability of a user interface.
Requirements to gain ECTS	Completed prototype deliverable. (35%) Completed prototype description. (25%) Completed prototype peer review. (25%) Completed workshop assignments (15%)

Subject title	Module: Project Management. Total Quality Management
Level <i>(Basic studies, master)</i>	Master
Subject aim	To provide knowledge about the quality concept, measurement concept and Total Quality Management (TQM). To create understandings about software quality providing principles, its objectives and

	software development understanding, implementation and maintenance of quality providing.
Subject content	To give understanding about principles of software quality assurance, its aims and to understand issues for ensuring software's quality development, establishment and maintaining. Students have to master skills and abilities in estimating the quality of process and product and in organising Total Quality Management. Acquire notions of software's quality and master skills to work out documentation-ensuring software's quality.
Expected results	Know: <ul style="list-style-type: none"> • IT project quality management; • IT project management documentation; • IT standards and regulations. Will be able: <ul style="list-style-type: none"> • to provide IT project management process smooth operation and coordination; • to develop and maintain IT project's documents for the project development and implementation process; • to use professional terminology.
Requirements to gain ECTS	Individual works are developed and presented. Passed tests.

Subject title	Module: Project management. IT Project Management
Level <i>(Basic studies, master)</i>	Master
Subject aim	To develop understanding about project development, develop practical skills in preparing a project plan. To develop understanding about project management, project manager and team's role in project implementation. To develop understanding about specifics of software development and IT projects.
Subject content	The goal of the course is to raise awareness of the project development process and to develop skills necessary for development of software project plan. The current course provides possibilities to acquire knowledge and develop skills and abilities required for the design, implementation and documenting of software projects.
Expected results	Completing the course, the student will be able: <ul style="list-style-type: none"> • to estimate the project costs and efforts; • to write project proposal; • to develop operational plan; • to make task decomposition; • to evaluate status of the project; • to manage risks; • to read professional literature; • to build a team; • to plan and conduct interviews; • to organise and manage meetings; • to analyse alternative solutions; • to develop project schedule; • to develop project budget; • to describe project processes.
Requirements to gain ECTS	Students must complete all tasks of the independent work. Starting the current course, the students establish teams by 3 to 4 members. The task for the each team is to develop application of the ICT-based project. At the end of the current course, the project application must be presented to the course teacher and other students. Each student needs to present his/her individual contribution, too. The current course ends with presentation of the study project (see "Study project (IT project planning)").

Subject title	Module: IT Project Management.
----------------------	--------------------------------

IT project work management	
Level <i>(Basic studies, master)</i>	Master
Subject aim	<p>Develop an understanding of project work management, build and develop skills and abilities to manage project work performance processes and work tasks To provide knowledge, build skills and develop abilities:</p> <ul style="list-style-type: none"> • manage the software project according to the organization's defined goals and development strategy • monitor, measure and control how software project implementation complies with project objectives • manage software project work changes, • collaborate with third parties to plan and manage procurement and contracts
Subject content	The aim of the course is to develop an understanding on IT project work management, to develop and improve skills and abilities to manage project work execution processes and tasks. The course acquires the knowledge, skills and competences required in IT project management, performance management and delivery management.
Expected results	<p>Completing the current course students will have the abilities to:</p> <ul style="list-style-type: none"> • evaluate project work execution processes, identify improvements and manage their implementation; • plan and determine the necessary physical resources, costs, deadlines, volumes; • plan, manage and manage the supply and efficient use of physical resources; • organize the procurement process and contract control; • cooperate with the interested parties involved in the organization and implementation of the procurement process; • coordinate project work changes in accordance with business priorities to achieve project goals; • evaluate and analyse project experience, lessons learned and ensure knowledge transfer; • organize the dissemination of information about the results achieved in the project; • lead and manage the achievement of project goals and the delivery of value to the organization's products and services; • manage compliance of project deliveries, plan and coordinate corrective actions in case of non-compliance; • understand, evaluate the quality criteria and measurements of the project result; • timely identify deviations from the quality acceptance criteria of deliveries and initiate their elimination; • timely and clearly communicate the necessary changes to the project team; • timely communicate and coordinate changes in quality requirements with interested parties • accurately, unambiguously assess the progress of project execution; • make modifications to the project plan in case of deviations and communicate them to interested parties; • provide quality support in the delivery acceptance and objectively evaluate the result of the acceptance; • evaluate the realization of project benefits; • identify, plan and coordinate improvements to project deliverables; • find out, analyse and summarize the experience and results of the project for reuse; • identify the baseline of project performance management, define the basic performance indicators of the project and their measurements; • to define the management processes of the key performance indicators of the project; • monitor, measure and control project performance indicators; • assess the compliance of the project results with the planned benefits and customer acceptance criteria; • analyse the causes of performance discrepancies, plan and take corrective measures.
Requirements to gain ECTS	At the beginning of the course, master's students form teams of 3-4 members. Each team uses the results of the project developed in the "Project Management" module to complete course tasks. Students' independent work during the course is related to the performance of both individual and team tasks. The results of some of the tasks completed in the

	course are expected to be included in the project deliverables of the "IT project management" module.
--	---

Subject title	Module: Software engineering. Big data
Level <i>(Basic studies, master)</i>	Master
Subject aim	To gain knowledge about big data in information technology; Data retrieval and processing methods and tools; data quality assessment methods. Acquire skills in big data analytics, in big data visualization in data quality assessment and quality assurance.
Subject content	The study course envisages creating an understanding of big data processing methods, data quality assurance. In the practical part, students acquire skills in big data analytics, data quality assessment and learn data processing and visualization tools.
Expected results	Knows big data processing methods. Knows data quality assurance and evaluation methods. Will be able to use data processing and visualization tools. Will be able to assess the quality of data. Competence to choose the most appropriate methods and tools for data processing and quality assurance
Requirements to gain ECTS	Part I. An information processing concept has been developed (70% of the assessment). Prepared a report on the current affairs of information processing (10% of the evaluation). Completed theory test (20% of the grade). Part II. Developed data quality assessment method (70%) Completed theory test (20%). Prepared report on data quality issues (10% of the grade). The final grade consists of the average grade from Part I and Part II.

Subject title	Module: Software engineering. Advanced software development methodologies
Level <i>(Basic studies, master)</i>	Master
Subject aim	Learn software development methods. To master software development process monitoring methods Master software integration and implementation methods
Subject content	The course acquires knowledge of software development methods and risks, monitoring software development processes and software integration and implementation methods so that a leading programming engineer can plan and manage software module and/or system planning, development, testing, implementation and maintenance.
Expected results	Will know the ability of software development and risks Will be able to ensure monitoring of software development processes Will be able to ensure the integration and implementation of software Will be able to plan and manage software planning, development, testing, implementation and maintenance.
Requirements to gain ECTS	All the theory tests are passed. All practical works have been developed and presented.

Subject title	Module: Software engineering. User behavior aspects
Level <i>(Basic studies, master)</i>	Master
Subject aim	Create an understanding of UX (user experience) and its core principles and applications in UI (user interface) design. Apply gained theoretical knowledge by building a high fidelity software prototype.
Subject content	The "User Experience Design" course should:

	<ul style="list-style-type: none"> ● give the necessary knowledge and understanding in the field of development of user-centred software applications, including development with the most commonly used software frameworks, evaluation and testing of their usability; ● put knowledge and skills into practice by working with prototyping software; ● expanded knowledge and formed new skills of students, including approaches to conduct user research and interviews, and conduct user testing.
Expected results	<p>Upon the course completion, students:</p> <ul style="list-style-type: none"> ● have knowledge of user centric approach and its application in user experience design and user interface design; ● have knowledge of user experience design principles; ● have knowledge of and ability to design efficient communication with the end user; ● have knowledge of common performance optimization methods; ● have knowledge of different prototyping methods; ● have ability to design user interface; ● have ability to design in-built user assistance; ● have ability to design and create interface wireframes; ● have ability to design and create high fidelity prototypes; ● have ability to analyse and evaluate usability of a user interface.
Requirements to gain ECTS	<p>Completed prototype deliverable. (35%) Completed prototype description. (25%) Completed prototype peer review. (25%) Completed workshop assignments (15%)</p>

Subject title	Module: Software engineering. Software Systems prototyping
Level <i>(Basic studies, master)</i>	Master
Subject aim	The goal of the course is to provide knowledge and skills in software engineering processes: software system prototyping and inform about industry trends of the field.
Subject content	The course provides knowledge of software system prototyping methods and tools; Industry examples, patterns and trends in these issues are discussed. To learn theory of software engineering processes: analysis and design, to investigate actual questions of the subject and discuss about them in seminars.
Expected results	<p>After successful finishing this course students:</p> <ul style="list-style-type: none"> ● after completing the course, students: ● will know how to choose adequate means for solving tasks and perform system design; ● will be able to plan software algorithm and operational scenarios ● will be able to use software system prototyping tools;
Requirements to gain ECTS	Task completion and exam passed

Subject title	Module: IT project management. Business analysis
Level <i>(Basic studies, master)</i>	Master
Subject aim	Gain knowledge about business analysis processes and methodologies. Gain skills in business analysis, business process modelling and tools use.
Subject content	The study course provides an understanding of business analysis processes and methodologies; business process modelling methods and tools. In the practical part, students acquire skills in business analysis and business process modelling. Will learn business analysis and business process modelling tools.
Expected results	Knows business analysis processes and methodologies. Will know how to perform business analysis using tools.

	<p>Will know how to develop a business process model using BPM tools</p> <p>Will be able to evaluate and apply business analysis methods and tools relevant and suitable for project implementation.</p>
<p>Requirements to gain ECTS</p>	<p>A business analysis process corresponding to the project has been developed (40% of the evaluation).</p> <p>A business process model corresponding to the project has been developed (40% of the assessment).</p> <p>Completed theory test (20% of the grade).</p>