



COURSE UNIT DESCRIPTION

Course unit title	Course unit code
Blockchain Technologies	

Lecturer(s)	Department where the course unit is delivered
Koordinuojantis: Saulius Grigaitis Kitas (-i):	Department of Software Engineering Institute of Computer Science Vilnius University

Cycle	Type of the course unit
1 st (BA)	Optional

Mode of delivery	Semester or period when the course unit is delivered	Language of instruction
Face-to-face	5 th , 7 th semester	Lithuanian, English

Prerequisites
Prerequisites:

Number of credits allocated	Student's workload	Contact hours	Individual work
5	133	66	67

Purpose of the course unit: programme competences to be developed		
<p>Purpose of the course unit: to provide students with basic knowledge of distributed ledger technologies by focusing on one of their types - blockchain technologies, their operation principles, algorithms used. Introducing to the development of such distributed systems using Rust.</p>		
<p>Generic competences:</p> <ul style="list-style-type: none"> • Communication and collaboration (<i>GC1</i>). <ul style="list-style-type: none"> ◦ An ability to organize their own work independently (<i>GC1.3</i>). • Life-long learning (<i>GC2</i>). <ul style="list-style-type: none"> ◦ An ability to undertake literature searches and analysis, and to use data bases and other sources of information (<i>GC2.2</i>). ◦ An ability independently to acquire new knowledge, methodologies, and tools and to apply them in practice (<i>GC2.3</i>). 		
<p>Specific competences:</p> <ul style="list-style-type: none"> • Knowledge and skills of underlying conceptual basis (<i>SC4</i>). <ul style="list-style-type: none"> ◦ An ability to apply mathematical foundations, knowledge of science and engineering, computer science theory, and algorithmic principles in software systems development (<i>SC4.2</i>). • Software development knowledge and skills (<i>SC5</i>). <ul style="list-style-type: none"> ◦ An ability to become familiar with new software engineering applications, to appreciate the extent of domain knowledge, to evaluate the complexity of the problems and the feasibility of their solution (<i>SC5.1</i>). 		
Learning outcomes of the course unit: students will be able to	Teaching and learning methods	Assessment methods
<ul style="list-style-type: none"> • Understand the principles of distributed ledger technologies. • Understand how the principles of distributed systems are used in distributed ledger technologies. • Understand the unique principles of 	Lectures, problem-oriented teaching, case studies, literary reading, individual work, tutorials, laboratory work.	Laboratory works and results presentation, written exam (open, semi-open and close-ended questions and tasks).

blockchain technologies that are not specific to the other types of distributed ledger technologies.								
• Use blockchain frameworks and libraries.								
• Use blockchain technologies to develop applications.								
• Use Rust programming language and instrumental tools to create blockchain systems.								
Course content: breakdown of the topics	Contact hours						Assessment methods	
	Lectures	Tutorials	Seminars	Practice	Laboratory work	Practical training	Contact hours	Individual work
Introduction.	2						2	2
Principles of distributed ledger technologies.	2				2		4	4
Principles of blockchain technologies. Cryptocurrencies. Smart Contracts.	4				4		8	8
Distributed systems theory. FLP impossibility. Synchronous, asynchronous and partially synchronous network models. Fault types. Fault detectors. Leader election. Time in distributed systems. Consensus properties.	4				6		10	8
Byzantine generals problem. PoW, PoS, pBFT, dBFT and other BFT consensus algorithms.	4				4		8	7
Cryptography in blockchain technologies.	2				2		4	5
Architectures of blockchain technologies.	2				2		4	5
Frameworks and libraries for blockchain development.	2				2		4	5
Introduction to Rust programming and ecosystem.	2				2		4	5
Rust functional programming.	2				2		4	4
Rust memory model.	2				2		4	4
Rust parallel programming.	2				2		4	4
Automated testing with Rust.	2				2		4	4
Tutorials during the semester		2					2	
Final exam (written)								
Total	32	2			32		66	67

Individual reading.
Laboratory works.
Self-control tasks.

Assessment strategy	Weight %	Deadline	Vertinimo kriterijai
Laboratory works	60	During the semester	During the semester, a group of students is required to complete a single project. The assessment of the project is divided into no more than 5 stages. The stages are of different complexity with the corresponding maximum scores for each stage. The maximum score for all stages is 6 points, which corresponds to 60% of the final score. It is required to get at least 2 points to be allowed to take the exam.
Exam (written)	40	Exam session	During the exam, it is possible to get at most 4 points, which are equivalent to 40% of the final score. During the examination, the student must answer various questions of diverse complexity (0-4 points). The student must provide answers to multiple different open, semi-open and closed type questions of different complexity (0-4 points).

Author	Publishing year	Title	Number or volume	Publisher or URL
Required reading				
Satoshi Nakamoto	2008	Bitcoin: A Peer-to-Peer Electronic Cash System		https://bitcoin.org/bitcoin.pdf
Vitalik Buterin, Virgil Griffith	2019	Casper the Friendly Finality Gadget		https://arxiv.org/abs/1710.09437
Steve Klabnik and Carol Nichols	2018	The Rust Programming Language		California, USA. No Starch Press
Recommended reading				
Miguel Castro, Barbara Liskov	1999	Practical Byzantine Fault Tolerance		http://pmg.csail.mit.edu/papers/osdi99.pdf
Maarten van Steen, Andrew S. Tanenbaum	2017	Distributed Systems (3 rd Edition)		California, USA. CreateSpace Independent Publishing Platform.



MODULIO APRAŠAS

Modulio pavadinimas	Kodas
Blokų grandinių technologijos	

Dėstytojas	Padalinys
Koordinuojantis: partnerystės doc. Saulius Grigaitis Kitas (-i):	Informatikos institutas Matematikos ir informatikos fakultetas Vilniaus universitetas

Studijų pakopa	Dalyko tipas
Pirmoji	Pasirenkamas

Igyvendinimo forma	Vykdyto laikotarpis	Vykdyto kalbos
Auditorinė	5, 7 semestrai	Lietuvių, anglų

Reikalavimai studijuojančiajam
Išankstiniai reikalavimai:

Modulio apimtis kreditais	Visas studento darbo krūvis	Kontaktinio darbo valandos	Savarankiško darbo valandos
5	130	66	64

Modulio tikslas: studijų programos ugdomos kompetencijos		
<p>Modulio tikslas: suteikti studentams bazines žinias apie išsklaidytų operacijų knygų technologijas fokusuojantis į vieną iš jų tipą - blokų grandinių technologijas, jų veikimo principus, naudojamus algoritmus. Supažindinti su tokių paskirstytų sistemų kūrimu naudojant Rust kalbą.</p>		
<p>Bendrosios kompetencijos:</p> <ul style="list-style-type: none">• Bendravimas ir bendradarbiavimas (BK1).<ul style="list-style-type: none">◦ Gebės savarankiškai efektyviai organizuoti savo darbą (BK1.3).• Nuolatinis mokymasis (BK2).<ul style="list-style-type: none">◦ Gebės atlikti literatūros paiešką ir analizę, naudoti duomenų bazes ir kitus informacijos šaltinius (BK2.2).◦ Gebės savarankiškai įsisavinti naujas žinias, metodus ir įrankius bei taikyti juos praktikoje (BK2.3).		
<p>Dalykinės kompetencijos:</p> <ul style="list-style-type: none">• Konceptualių pagrindų žinios ir gebėjimai (DK4).<ul style="list-style-type: none">◦ Gebės taikyti matematikos pagrindų, mokslo, inžinerijos, kompiuterių mokslo teorines žinias ir algoritminius principus programų sistemų kūrimo (DK4.2).• Programų sistemų kūrimo žinios ir gebėjimai (DK5).<ul style="list-style-type: none">◦ Gebės išžvelgti naujas programų sistemų taikymo galimybes, įvertinti taikomosios srities žinių poreikį, problemų kompleksumą bei jų sprendimų būdų įgyvendinamumą (DK5.1).		
Modulio studijų siekiniai	Studijų metodai	Vertinimo metodai
<ul style="list-style-type: none">• Suprasti išsklaidytų operacijų knygų technologijų veikimo principus.• Suprasti kaip išskirstytų sistemų veikimo principai yra panaudojami kuriant išsklaidytų operacijų knygų sistemas.• Suprasti blokų grandinių technologijų veikimo principus, kurie nėra būdingi kitų tipų išsklaidytų operacijų knygų sistemoms.• Naudoti blokų grandinių technologijoms	Paskaitos, probleminis dėstymas, atvejų analizė, literatūros skaitymas, savarankiškas darbas, pavyzdžių analizė, konsultacijos, laboratoriniai darbai.	Laboratorinių darbų atlikimas bei rezultatų gynimas, egzaminas raštu (atvirojo, pusiau atvirojo bei uždarojo tipo klausimai ir užduotys).

kurti skirtus karkasus ir bibliotekas <ul style="list-style-type: none"> Gebėti taikomiosiose programose panaudoti tinkamas blokų grandinių technologijas Naudoti Rust programavimo kalbą ir instrumentines priemones kuriant blokų grandinių sistemas. 								
Temos	Kontaktinio darbo valandos						Savarankiškų studijų laikas ir užduotys	
	Paskaitos	Konsultacijos	Seminarai	Pratybos	Laboratoriniai darbai (LD)	Praktika	Visas kontaktinis darbas	Savarankiškas darbas
Įvadas.	2					2	2	Savarankiškas literatūros skaitymas Laboratoriniai darbai. Savikontrolės užduočių atlikimas.
Išsklaidytų operacijų knygų technologijų veikimo principai.	2				2	4	4	
Blokų grandinių technologijų veikimo principai. Kriptovaliutos. Išmanieji kontraktai.	4				4	8	8	
Išskirstytų sistemų teorija. FLP negalimumas. Sinchroninis, asinchroninis ir dalinai sinchroninis modeliai. Trikių tipai. Trikių detektoriai. Lyderio rinkimas. Laikas išskirstytose sistemose. Konsensuso sąvybės.	4				6	10	8	
Bizantijos generolų problema. PoW, PoS, pBFT, dBFT ir kiti BFT konsensuso algoritmai.	4				4	8	7	
Kriptografija blokų grandinių technologijose	2				2	4	5	
Blokų grandinių technologijų architektūros.	2				2	4	5	
Blokų grandinių kūrimui skirti karkasai ir bibliotekos.	2				2	4	4	
Rust programavimo kalbos pagrindai ir ekosistema.	2				2	4	4	
Funkcinė paradigma Rust programavimo kalboje.	2				2	4	4	
Atminties modelis Rust programavimo kalboje.	2				2	4	4	
Lygiagretumo modeliai Rust programavimo kalboje.	2				2	4	4	
Automatizuotas testavimas su Rust.	2				2	4	3	
Konsultacijos semestro eigoje		2				2		
Egzaminas (raštu)								
Total	32	2			32	66	64	

Vertinimo strategija	Svoris proc.	Atsiskaitymo laikas	Vertinimo kriterijai
Laboratoriniai darbai	60	Semestro metu	Semestro metu studentų grupė turi atlikti vieną projektą. Projektas skaidomas į ne daugiau nei 5 etapus. Etapai yra skirtingo sudėtingumo su atitinkamais maksimaliais vertinimo balais už kiekvieną etapą. Maksimalus visų etapų įvertinimas yra 6 balai, kurie atitinka 60% galutinio įvertinimo. Minimalus įvertinimas leidžiantis laikyti egzaminą yra 2 balai.
Egzaminas (raštu)	40	Egzaminų sesijos metu	Egzamino metu galima surinkti iki 4 balų, kurie atitinka 40% galutinio įvertinimo. Egzamino metu studentas turi pateikti atsakymus į skirtingus įvairaus sudėtingumo atvirojo, pusiau atvirojo bei uždarojo tipo klausimus (0-4 balai).

Autorius	Leidimo metai	Pavadinimas	Periodinio leidinio Nr. ar leidinio tomas	Leidimo vieta ir leidykla ar internetinė nuoroda
Privalomoji literatūra				
Satoshi Nakamoto	2008	Bitcoin: A Peer-to-Peer Electronic Cash System		https://bitcoin.org/bitcoin.pdf
Vitalik Buterin, Virgil Griffith	2019	Casper the Friendly Finality Gadget		https://arxiv.org/abs/1710.09437
Steve Klabnik and Carol Nichols	2018	The Rust Programming Language		California, USA. No Starch Press
Papildoma literatūra				
Miguel Castro, Barbara Liskov	1999	Practical Byzantine Fault Tolerance		http://pmg.csail.mit.edu/papers/o/sdi99.pdf
Maarten van Steen, Andrew S. Tanenbaum	2017	Distributed Systems (3 rd Edition)		California, USA. CreateSpace Independent Publishing Platform.