Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Computer Systems Software Department

BLOCK CHAIN TECHNOLOGY

Syllabus

1. Requisites of the Course

Cycle of Higher Education	***First cycle of higher education (Bachelor's degree)***
Field of Study	*12 Information Technologies*
Speciality	*121 Software engineering*
Education Program	*Software Engineering of Multimedia and Information Retrieval Systems*
Type of Course	*Elective*
Mode of Studies	*full-time*
Year of studies, semester	*4 year (8 semester)*
ECTS workload	*Time allocation: 36 hours for lectures, 18 hours for lab classes, 66 hours for self-study*
Testing and assessment	*Credit, Modular test*
Course Schedule	*According to rozklad.kpi.ua*
Language of Instruction	*English*
Course Instructors	*Lecturer: teaching assistant, Yuri Zhykin, yzhykin@protonmail.com,* *Lab classes: teaching assistant, Yuri Zhykin, yzhykin@protonmail.com*
Access to the course	*GitHub: https://github.com/rodentrabies/nobsbitcoin*

2. Outline of the Course

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

The study of the discipline "Block Chain Technology" allows students to form the competencies necessary to solve practical problems of professional activities related to interaction with block chain systems and distributed consensus systems.

*The **purpose** of studying the discipline "Block Chain Technology" is to introduce the students to the modern technologies and instruments for development and interaction with cryptocurrency systems and form practical skills necessary to interact with block chain technologies and understand their underlying concepts, such as cryptography, computer networks, information theory and economics.*

*The **subject** of the discipline "Block Chain Technology" is theoretical and practical principles of development and analysis of the source code necessary for interaction with cryptocurrency protocol.*

*Study of the discipline "Block Chain Technology" contributes to the formation of **professional competencies (PC)** in students, necessary for solving practical tasks of professional activity, related to the development and use of information retrieval methods in software.*

***PC06** Ability to analyze, select and use information security methods and tools (including cybersecurity).*
***PC07** Knowledge of information data models, the ability to create software for data storage, retrieval and processing.*
***PC08** Ability to use fundamental interdisciplinary knowledge for successful solving of software engineering problems.*
***PC14** Ability to algorithmic and logical thinking.*

*Study of the discipline "Block Chain Technology" contributes to the formation in students of the following **program learning outcomes (PLO)** according to the educational program:*

PLO01 *Analyze, research and select information and documentation resources and knowledge necessary for solving professional tasks relevant in terms of modern science and technology.*
PLO05 *To know and apply relevant mathematical concepts, domain methods, system and object-oriented analysis and mathematical modeling for software development.*
PLO13 *To know and apply methods of developing algorithms, designing software and data and knowledge structures.*
PLO15 *To choose programming languages and development technologies to solve the problems of creating and maintaining software.*
PLO19 *To know and be able to apply software verification and validation methods.*

2. Prerequisites and post-requisites of the course (the place of the course in the scheme of studies in accordance with curriculum)

Successful study of the discipline "Block Chain Technology" is preceded by the study of normative disciplines of the curriculum for Bachelors in 121 Software Engineering.

The theoretical knowledge and practical skills obtained during the mastering of the discipline "Block Chain Technology" ensure the successful implementation of course and diploma projects in the specialty 121 Software engineering.

3. Content of the course

Topic 1. Historical and economic context of the invention and development of cryptocurrency technology

Topic 2. Bitcoin block chain protocol

Topic 3. Open problems of Bitcoin cryptocurrency protocol

Topic 4. Alternative cryptocurrency technologies, their advantages and drawbacks

Modular test

4. Coursebooks and teaching resources

Main resources:

1. Bitcoin and cryptocurrency technology. Lecture notes [Online]. – 2023 – Available: https://github.com/rodentrabies/nobsbitcion.

Additional resources:

2. Saifedean Ammous. Bitcoin Standard. The Decentralized Alternative to Central Banking. Wiley; 1st edition, 2018, 304 p.
3. Mastering Bitcoin / Andreas Antonopoulos. [Online]. Available: https://github.com/bitcoinbook/bitcoinbook.
4. Mastering Ethereum / Andreas Antonopoulos, Gavin Wood. [Online]. Available: https://github.com/ethereumbook/ethereumbook.
5. Mastering the Lightning Network / Andreas Antonopoulos, Olaoluwa Osuntokun, Rene Pickhardt. [Online]. Available: https://github.com/lnbook/lnbook.
6. Reference Bitcoin Implementation. [Online]. Available: https://github.com/bitcoin/bitcoin.
7. Jimmy Song. Programming Bitcoin: Learn How to Program Bitcoin from Scratch. O'Reilly Media; 1st edition, 2019, 322 p.

3. Educational content

5. Methodology

№	Type of class	Class description
Topic 1. Historical and economic context of the invention and development of cryptocurrency technology		
1	Lecture 1	Introduction. Economic and ideological reasons for Bitcoin invention. Money. Primitive forms of money. Metallic money. Fiat money. Stable money.
2	Lecture 2	What is Bitcoin. History of Bitcoin. Who uses Bitcoin.
3	Lecture 3	Why do we need Bitcion. Preservation of purchasing power. Individual autonomy.
Topic 2. Bitcoin block chain protocol		
4	Lecture 4	How does Bitcoin work. Transactions. Blocks. Mining. Block chain.
5	Laboratory class 1	Overview and installation of the reference Bitcoin implementation 1.
6	Laboratory class 2	Overview and installation of the reference Bitcoin implementation 2.
7	Laboratory work 1	Perform assigned task.
8	Lecture 5	Bitcoin client. Reference implementation. Interaction with the Bitcoin network node. Alternative clients.
9	Lecture 6	Keys, addresses, wallets. Public and private keys and cryptography. Elliptic curve cryptography. Bitcoin addresses.
10	Laboratory class 3	Setting up Bitcoin wallet on Linux, Android and iOS.
11	Lecture 7	Transactions. Transaction lifecycle. Structure, inputs and outputs of a transaction. Sctripts and script language.
12	Lecture 8	Scripts and script language.
13	Laboratory class 4	Development of a program for working with Bitcoin scripts. Static analysis of scripts.
14	Lecture 9	Bitcoin network. Network architecture. Types and roles of Bitcoin nodes. Transaction pool.
15	Lecture 10	Block chain. Block structure. Block header. Block identifiers: block hash and block height.
16	Laboratory class 5	Development of a program for synchronization and storing of the block header database.
17	Laboratory work 2	Perform assigned task.
18	Lecture 11	Chaining of blocks. Merkle trees.
19	Lecture 12	Mining and consensus. Decentralized consensus. Independent verification of transactions. Aggregation of transactions into blocks.
20	Laboratory class 6	Development of the program for reading, parsing and simplified validation of block chain data.
21	Laboratory work 3	Perform assigned task.
22	Lecture 13	Block rewards. Transaction fees. Proof-of-work algorithm. Proof-of-work complexity and tuning. Block validation.
Topic 3. Open problems of Bitcoin cryptocurrency protocol		
23	Lecture 14	Mining and hash power race. Consensus attacks. Bitcoin security. Security recommendations.
24	Laboratory class 7	Development of the program for interaction with peer-to-peer Bitcoin network.

25	Lecture 15	Potential improvements and new developments in Bitcoin protocol. Potential problems and attack vectors. Lightning network. Payment channels.
Topic 4. Alternative cryptocurrency technologies, their advantages and drawbacks		
26	Lecture 16	Alternative block chains and their use.
27	Laboratory class 8	Work with software for interaction with second-layer protocols on top of Bitcoin.
28	Lecture 17	Ethereum.
29	Lecture 18	Monero.
30	Laboratory class 9	Work with Ethereum and Monero software.

6. Self-study

The discipline "Block Chain Technology" is based on independent preparation for classroom classes on theoretical and practical topics.

№	Title	Number of hours	Resources
1	Preparation to the lecture 1	1	1;2
2	Preparation to the lecture 2	1	1;2
3	Preparation to the lecture 3	1	1;2
4	Preparation to the lecture 4	1	1;2
5	Preparation to the laboratory class 1	3	1;3;6;7
6	Preparation to the laboratory class 2	3	1;3;6;7
7	Laboratory work 1	7	1;3;6;7
8	Preparation to the lecture 5	1	1;3;6;7
9	Preparation to the lecture 6	1	1;3;6;7
10	Preparation to the laboratory class 3	3	1;3;6;7
11	Preparation to the lecture 7	1	1;3;6;7
12	Preparation to the lecture 8	1	1;3;6;7
13	Preparation to the laboratory class 4	3	1;3;6;7
14	Preparation to the lecture 9	1	1;3;6
15	Preparation to the lecture 10	1	1;3;6
16	Preparation to the laboratory class 5	3	1;3;6;7
17	Laboratory work 2	7	1;3;6;7
18	Preparation to the lecture 11	1	1;3;7
19	Preparation to the lecture 12	1	1;3;7
20	Preparation to the laboratory class 6	3	1;3;6;7
21	Laboratory work 3	7	1;3;6;7
22	Preparation to the lecture 13	1	1;3;7

23	*Preparation to the lecture 14*	*1*	*1;5;6;7*
24	*Preparation to the laboratory class 7*	*3*	*1;3;6;7*
25	*Preparation to the lecture 15*	*1*	*1;5;6;7*
26	*Preparation to the lecture 16*	*1*	*1;4*
27	*Preparation to the laboratory class 8*	*3*	*1;3;6;7*
28	*Preparation to the lecture 17*	*1*	*1;4*
29	*Preparation to the lecture 18*	*1*	*1;4*
30	*Preparation to the laboratory class 9*	*3*	*1;3;5;6;7*

4. Policy and Assessment

7. Course policy

Attendance at lectures is mandatory.

Attendance at laboratory classes can be sporadic and if you need to defend a laboratory work.

Rules of conduct in the classroom: activity, respect for those present.

Adherence to the policy of academic integrity.
Rules for protecting the laboratory work: the work should be done according to the option of the student, which is determined by his number in the group list.
The rules for assigning penalty points are as follows.

8. Monitoring and grading policy

During the semester, students complete 3 laboratory works. Maximum number of points for each work: 20 points

Points are awarded for:
- quality of the work: 0-5 points;
- answer during the work defense of: 0-3 points;
- timely submission of work to the defense: 0-2 points.

Criteria for evaluating the quality of work:
10 points – the work is done perfectly;
8 points – the work is done completely and well, but has shortcomings;
6 point – the work is done completely, but has some issues;
4 points – the work is done completely, but has serious issues;
2 points – the work is incomplete but shows an certain amount of effort;
0 points – the work is not done.

Answer evaluation criteria:
6 points – complete answer, well-reasoned;
4 points – complete answer, with minor issues;
2 point – complete answer, with major issues;
0 points – not complete or incorrect answer.

Timely submission of work to the defense:
4 points – the work is submitted for defense no later than the specified deadline;
2 points – the work is submitted for defense later than the specified deadline.

Maximum number of points for performing and defending laboratory works:
20 points × 3 laboratory works = 60 points.

The task for the midterm test consists of 4 test questions. Each question is valued at 10 points.

Answer evaluation criteria for a question:

10 points – complete, correct answer confirmed with references;
8-9 points – complete, correct answer, no confirming references;
6-7 points – correct but incomplete answer;
4-5 points – slightly incorrect answer, corrected during discussion;
1-3 points – incorrect answer, unable to correct during discussion;
0 points – no answer.

Maximum number of points
20 points × 3 laboratory works + 10 points × 4 questions = 100 points.

During the semester's laboratory classes the students will be asked questions on the topics of all the lectures studied before the class and may be rewarded additional points for correct answers to those questions.

The rating scale for the discipline is: R = R_C = 100 points

Semester assessment: final test.
If a student has R_C of 60 and more points, they are eligible for automatic grading according to the table below. Otherwise, it will be necessary to answer 3-5 questions on the topics reviewed in the lecture material.

The table of compliance between overall points and the final grade:

Points	*Grade*
100-95	Excellent
94-85	Very good
84-75	Good
74-65	Satisfactory
64-60	Fair
Less than 60	Unsatisfactory
Course requirements are not met	Not graded

Syllabus of the course

Is designed by teaching assistant, Yuri Zhykin

Adopted by Computer Systems Software Department (protocol № 8 from 25.01.23)

Approved by the Faculty Board of Methodology (protocol № 6 from 27.01.23)