National Technical
University of Ukraine
“Igor Sikorsky
Kyiv Polytechnic Institute”

Апепс

Department of Design
Automation for Energy
Processes and Systems

Decentralized Applications Development
Work program of the discipline (Syllabus)

Course Requisites

Educational level	***Third (PhD)***
Field of knowledge	*12 Information technology*
Specialty	*121 Software Engineering*
Educational program	*Software Engineering*
Course status	*Elective*
Form of study	*Part-time*
Year of study, semester	*2 year, spring semester*
Number of ECTS credits	*5 ECTS credits (150 hours)*
End-of-semester control / control measures	*Final test*
Timetable	*http://rozklad.kpi.ua/*
Language of study	*English*
Information about course leader / teachers	Lecturer: PhD, Professor Havrylko Yevhen Volodymyrovych, gev.1964@ukr.net, tel. 067-506-91-85 Practical training: PhD, Professor Havrylko Yevhen Volodymyrovych, gev.1964@ukr.net, tel. 067-506-91-85
Course placement	http://campus.kpi.ua/

Curriculum

1. Description of the discipline, its purpose, subject of study and learning outcomes

The large number of computers connected by high-speed systems allows you to create computer networks, or in a new paradigm of computer science - decentralized or distributed systems (DS). A DS is a set of independent computers provided to their users by a single integrated system. Study of the discipline "Decentralized applications"

is dedicated to different approaches to the development and application of algorithms, computer programs and their new components, architecture, scaling and distribution to achieve maximum technical and commercial capabilities.

The purpose of the discipline is to form students' abilities to use the legislation of Ukraine, organizational, technical, algorithmic and other methods and tools, programs and data, legislation and standards in this area, modern DS, cryptosystems; hashing, the ability to use them in professional activities.

Task. The main tasks of studying the discipline "Decentralized Applications" are the acquisition of postgraduate competencies in order to participate in the design of decentralized information systems. As a result of studying the discipline, graduate students should form the following competencies:

common:

• ability to adhere to the ethics of research, as well as the rules of academic integrity in research and scientific and pedagogical activities (LC1),

• ability to scientific and abstract thinking, analysis and synthesis (LC2),

• Ability to find, process and analyze information needed to solve problems and make decisions (LC10).

professional:

• Ability to critically rethink existing software engineering technologies and track their development trends (FC3);
• Ability to think creatively, generate new progressive ideas in software engineering. (FC7);
• Ability to apply formal methods of design, development and research of software systems and technologies in research (FC6);
• mechanisms of architecture formation in decentralized systems (FC8);
• development of applications in decentralized systems (FC8).

After mastering the discipline, students must demonstrate the following program learning outcomes:

• develop decentralized systems (PRN3);
• work with decentralized systems (PRN5);
• the role and place of the blockchain in the design and configuration of information systems (PRN7);
• principles and methods of constructing standard blockchain solutions (PRN7);
• basics of functioning of the network blockchain consensus (PRN7);
• modern software technologies for the deployment of blockchain systems (PRN7).
• skills:
• analyze the known methods of blockchain design in accordance with the task and choose a specific method based on the purpose, objectives, real assumptions and limitations of blockchain development (PRN17);
• - choose specific methods of building a blockchain and build a prototype blockchain (PRN17).
• experience:
• - rational use of modern technologies, application packages and integrated blockchain programming environments (PRN19);
• - mastering modern blockchain programming software in selected environments (PRN21).

2. Prerequisites and postrequisites of the discipline (place in the structural and logical scheme of education according to the relevant educational program)

Prerequisites: mathematics, computer science, programming, operating system administration, information and coding theory, theoretical foundations of cryptography, database fundamentals, systems theory and systems analysis.

Postrequisites: knowledge gained in the study of this discipline is used in the following disciplines: design of distributed databases, design of information systems, design of computerized intelligent control systems and others.

3. The content of the discipline

Section 1. Basic model of decentralized systems.

Topic 1.1. Paradigm, principles, concepts of development of decentralized (distributed) systems.

Topic 1.2. Communication in computer and decentralized systems.

Section 2. Processes in decentralized systems.

Topic 2.1. Processes in decentralized systems.

Topic 2.2. Entity naming processes in decentralized systems.

Topic 2.3. Synchronization processes in decentralized systems.

Topic 2.4. Consistency formation and replication processes in decentralized systems.

Topic 2.5. Fault tolerance processes in decentralized systems.

Topic 2.6. Processes for protecting processes, computer systems, and networks in decentralized systems.

Section 3. Approaches to the formation of architecture in decentralized systems.

Topic 3.1. Approaches to the distribution of the system of objects in decentralized systems.

Topic 3.2. Approaches to the distribution of file systems in decentralized systems.

Topic 3.3. Approaches to the distribution of file systems in decentralized systems.

Topic 3.4. Approaches to the distribution of coordination systems in decentralized systems.

Section 4. Applications in decentralized systems.

4.1. Encryption in decentralized applications.

4.2. Hashing information for file systems in decentralized applications.

4.4. Blockchain-based applications.

4.5. Cryptosystems and cryptocurrencies.

4.6. Token.

4. Training materials and resources

Basic literature:

1. Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. ван Стеен. — СПб.: Питер, 2003. — 877 с: ил. — (Серия «Классика computer science»)

1. Венбо М. Современная криптография: теория и практика : пер. с англ. / М.Венбо – М. : Издательский дом "Вильямс", 2005. – 768 с.

2. Артем Генкин, Алексей Михеев Блокчейн: как это работает и что ждет нас завтра. – М.:Д, 2018.

3. Дрешер Д. Основы блокчейна: вводный курс для начинающих в 25 небольших главах. – М.:АльпинаПаблишер, 2018.

4. Вербіцький О. В. Вступ до криптології / О. В. Вербіцький. – Львів : ВНТЛ, 2011. – 248 с.

Additional literature:

1. Гребенчук В. Г. Цифровая стеганография / В. Г. Гребенчук, И. Н. Оков, И. В. Туринцев. – М. : СОЛОН-Пресс, 2012. – 272 с.

2. Зима В. М. Безопасность глобальных сетевых технологий / В. М. Зима, А. А. Молдовян, Н. А. Молдовян. – 2-е изд. – СПб. : БХВ-Петербург, 2013. – 368 с.

3. Кузнецов О. О. Захист інформації в інформаційних системах. Методи традиційної криптографії / О. О. Кузнецов, С. П. Євсеєв, О. Г. Король. – Х. : Вид. ХНЕУ, 2010. – 316 с.

4. Поповский В. В. Защита информации в телекоммуникационных системах : учебник : в 2 т. / В. В. Поповский, А. В. Персиков. – Х. : ООО "Компания СМИТ", 2006. – Т. 1. – 292 с.

5. Поповский В. В. Защита информации в телекоммуникационных системах : учебник : в 2 т. / В. В. Поповский, А. В. Персиков. – Х. : ООО "Компания СМИТ", 2006. – Т. 2. – 252 с.

6. Столлингс В. Криптография и защита сетей: принципы и практика. – 2-е изд. / В. Столлингс; пер. с англ. – М. : Издательский дом "Вильямс", 2001. – 672 с.

Educational content

5. Methods of mastering the discipline (educational component)

Section 1. Basic model of decentralized systems.

Topic 1.1. Paradigm, principles of development of decentralized (distributed) systems.

Lecture 1.1. Paradigm, principles, concepts of development of decentralized (distributed) systems.

Definition of DS. How a user connects to resources. DS tasks. Hardware solutions concept. Concepts of software solutions. Client-server model.

Topic 1.2. Communication in computer and decentralized systems.

Lecture 1.2. Communication in computer and decentralized systems.

Protocol levels. Remote procedure call. Access remote objects. Communication by messages. Communication based on data flows.

Section 2. Processes in decentralized systems.

Topic 2.1. Processes in decentralized systems.

Lecture 2.1. Processes in decentralized systems.

Execution threads. Customers. Servers. Code transfer. Software units.

Topic 2.2. Entity naming processes in decentralized systems.

Lecture 2.2. Entity naming processes in decentralized systems.

Entities and their names. Placement of mobile entities. Deleting entities.

Topic 2.3. Synchronization processes in decentralized systems.

Lecture 2.3. Synchronization processes in decentralized systems.

Clock synchronization. Logic clock. Global states. Voting algorithms. Mutual exceptions. Distribution of transactions.

Topic 2.4. Consistency formation and replication processes in decentralized systems.

Lecture 2.4. Consistency formation and replication processes in decentralized systems.

Models. Data-oriented consistency models. Distribution protocols. Consistency protocols.

Topic 2.5. Fault tolerance processes in decentralized systems.

Lecture 2.5. Fault tolerance processes in decentralized systems.

Understanding fault tolerance. Fault tolerance of processes. Client-server communication failure. Improper group mailing. Distributed confirmation. Restoration.

Topic 2.6. Processes for protecting processes, computer systems, and networks in decentralized systems.

Lecture 2.6. Processes for protecting processes, computer systems, and networks in decentralized systems.

Threats. Cryptography in DS. Channel security. Access control. Protection management.

Section 3. Approaches to the formation of architecture in decentralized systems.

Topic 3.1. Approaches to the distribution of the system of objects in decentralized systems.

Lecture 3.1. Approaches to the distribution of the system of objects in decentralized systems.

C0RBA.DCOM.Glode. Comparison of the mentioned systems.

Topic 3.2. Approaches to the distribution of file systems in decentralized systems.

Lecture 3.2. Approaches to the distribution of file systems in decentralized systems.

Sun Network File System. Example of file systems without servers. Scalable protection system.

Topic 3.3. Approaches to the distribution of file systems in decentralized systems.

Lecture 3.3. World Wide Web.

Topic 3.4. Approaches to the distribution of coordination systems in decentralized systems.

Lecture 3.4. TIB / Rendezvous.

Section 4. Applications in decentralized systems.

Topic 4.1. Encryption in decentralized applications.

Lecture 4.1. Digital signatures.

Scheme of digital signature application. Digital signature based on RSA cipher. Digital signature based on the code of El Gamal. Digital Signature Algorithm (DSA). Standard GOST R34.10-944.2. Hashing information for file systems in decentralized applications.

Topic 4.4. Blockchain-based applications.

Lecture 4.4. Blockchain-based applications.

Warhead scheme. Principles of development based on the relationship based on the warhead.

Topic 4.5. Cryptosystems and cryptocurrencies.

Lecture 4.5. Cryptocurrencies.

Review of cryptocurrencies. Application of cryptocurrencies

Topic 4.6. Tokens.

Lecture 4.5. Tokens as the essence of modern art exchange.

Tokens Review. Application of tokens.

6. Self-study

Section 1. Basic information security model.

Definition of DS. How a user connects to resources. DS tasks. Hardware solutions concept. Concepts of software solutions. Client-server model.

Protocol levels. Remote procedure call. Access remote objects. Communication by messages. Communication based on data flows.

Section 2. Processes in decentralized systems.

Execution threads. Customers. Servers. Code transfer. Software units.

Entities and their names. Placement of mobile entities. Deleting entities.

Clock synchronization. Logic clock. Global states. Voting algorithms. Mutual exceptions. Distribution of transactions.

Models. Data-oriented consistency models. Distribution protocols. Consistency protocols.

Understanding fault tolerance. Fault tolerance of processes. Client-server communication failure. Improper group mailing. Distributed confirmation. Restoration.

Threats. Cryptography in DS. Channel security. Access control. Protection management.

Section 3. Approaches to the formation of architecture in decentralized systems.

C0RBA.DCOM.Glode. Comparison of the mentioned systems.

Sun Network File System. Example of file systems without servers. Scalable protection system.

Section 4. Applications in decentralized systems.

Scheme of digital signature application. Digital signature based on RSA cipher. Digital signature based on the code of El Gamal. Digital Signature Algorithm (DSA). Standard GOST R34.10-944.2. Hashing information for file systems in decentralized applications.

Warhead scheme. Principles of development based on the relationship based on the warhead.

Review of cryptocurrencies. Application of cryptocurrencies

Tokens Review. Application of tokens.

Policy and control

7. Policy of academic discipline (educational component)

Attendance at lectures and practical classes is mandatory except for good reasons (illness, force majeure).

In case of skipping classes for good reasons, the teacher gives the student the opportunity to perform all or some laboratory tasks (except for the performance of some tasks in connection with the end of the educational process).

In case of skipping classes without good reason, as well as due to violation of the deadline deadline (student), the student may receive a reduced number of points from the maximum score for the task.

During the semester students:

- perform and defend laboratory work in a timely manner,

- write modular control work,

- should positively close two certifications (in late March and mid-May),

- at the end of the educational process make a test.

8. Types of control and rating system of assessment of learning outcomes (RSO)

System of rating (weight) points and evaluation criteria

The maximum number of points from the credit module is 100.

The student's rating in the discipline consists of points that he receives for:

- *performance and protection of laboratory work,*
- *modular test (MCR) lasting 1 acad. hour.*

1. Performing laboratory tasks

The task of laboratory work is an individual performance of work related to the solution on a computer of a given problem of computer modeling.

The weight points of the tasks are given in the table.

Tasks	*Contribution to the semester score rating*
Task №1. Symmetric cryptosystems. DES algorithm.	*20*
Task №2. Asymmetric cryptosystems. RSA algorithm.	*20*

Task №3. Electronic digital signature based on the RSA algorithm	*20*

The maximum number of points for all tasks is 60 points.

Evaluation criteria

Preparation for work (as a percentage of the maximum number of points for the relevant work):

- the protocol meets the requirements, neat - 20%;

- the protocol meets the requirements, but there are numerical corrections - 10%;

Execution of the task of laboratory work:

- the work is done completely and correctly within the allotted time - 50%;

- work performed after the specified deadline - 20%;

Quality of work protection:

- the student answered the question correctly and completely - 30%;

- the student admitted insignificant inaccuracies in the answer - 20%;

- the student admitted significant inaccuracies when answering the question, but corrected them on his own - 10%.

2. Modular control

Weight score - 40.

The test consists of 20 test tasks. For each correct answer to the question, 2 points are awarded.

The sum of weight points of control measures during the semester is:

R = 60 + 40 = 100 points.

A prerequisite for admission to the test is the enrollment of all laboratory work, as well as a starting rating (rc) of at least 40% of R, ie 40 points.

The sum of points is transferred to the credit score according to the table:

Table of correspondence of rating points to grades on the university scale:

Number of pints	*Grades*
100-95	Excellent
94-85	Very good
84-75	Good
74-65	Satisfactorily
64-60	Sufficiently
Less 60	Unsatisfactory
Admission conditions are not met	Not allowed

Work program of the discipline (Syllabus):

Developed by *Professor, Ph.D., Prof. Havrylko Yevhen Volodymyrovych*

Approved *by department ___________ (protocol № ___ from _____________)*

Resolved *by Methodical commission of the faculty (protocol № __ from ________)*