Syllabus
Course Program

Efficiency and quality of architectural solutions of information systems

Specialty
122 – Computer sciences

Institute
Institute of Computer Modeling, Applied Physics and Mathematics

Educational program
Computer science. Modeling, design, and computer graphics

Department
Computer Modeling of Processes and Systems (162)

Level of education
Master's level

Course type
Special (professional), Mandatory

Semester
2

Language of instruction
English

Lecturers and course developers

Ivanchenko Kseniia Viktorivna

kseniia.ivanchenko@khpi.edu.ua
Candidate of technical sciences, associate professor, associate professor of the Department of Computer Modeling of Processes and Systems

Author and co-author of more than 30 scientific and methodical publications.
Courses: "Design of software systems", "Cross-platform programming", "Analysis and testing of software systems", "Programming of mobile devices".
More about the lecturer on the department's website

General information

Summary

This course provides students with a comprehensive understanding of core concepts, architectural styles, and patterns essential for designing robust, scalable, and secure information systems. Students will acquire the skills to assess efficiency metrics, ensure quality attributes, and implement security measures, preparing them for roles where they can contribute to the development of high-performance and resilient IT solutions.

Course objectives and goals

The purpose of this course is to equip students with the knowledge and practical skills required to design and evaluate information systems architectures that prioritize efficiency and quality. By focusing industry best practices, the course aims to prepare students for roles where they can contribute effectively to the development and maintenance of robust, scalable, and secure information systems. Students will gain an understanding of architectural solutions, enabling them to make informed decisions and design systems that meet contemporary standards of efficiency and quality in the rapidly evolving field of information technology.
Course Objectives:
Understand core concepts;

Integrate agile and architecture;
Assess efficiency metrics;
Ensure quality attributes;
Explore architectural styles and patterns;
Address scalability challenges;
Optimize information retrieval;
Implement security measures;
Evaluate system reliability.

Format of classes

Lectures, laboratory classes, consultations, self-study. Final control in the form of an exam.

Competencies

GC01. Ability to think abstractly, analyze and synthesize.
GC02. Ability to apply knowledge in practical situations.
GC03. Ability to communicate in the state language both orally and in writing.
GC05. Ability to learn and master modern knowledge.
GC06. Ability to be critical and self-critical.
GC07. Ability to generate new ideas (creativity).
PC01. Understanding of the theoretical foundations of computer science.
PC02. Ability to formalize the subject area of a particular project in the form of an appropriate information model.
PC04. Ability to collect and analyze data (including big data) to ensure the quality of project decision-making.
PC05. Ability to develop, describe, analyze and optimize architectural solutions for information and computer systems for various purposes.
PC06. Ability to apply existing and develop new algorithms for solving problems in the field of computer science.
PC07. Ability to develop software in accordance with the formulated requirements, taking into account available resources and constraints.
PC10. Ability to evaluate and ensure the quality of IT projects, information and computer systems for various purposes, apply international standards for assessing the quality of software for information and computer systems, models for assessing the maturity of information and computer system development processes.
PC11. Ability to initiate, plan and implement processes for the development of information and computer systems and software, including its design, analysis, testing, system integration, implementation and maintenance.

Learning outcomes

LO1. Have specialized conceptual knowledge, including modern
scientific achievements in the field of computer science and is the basis for original thinking and research, critical thinking of problems in the field of computer science and on the border of knowledge fields.
LO2. Have specialized skills/abilities to solve computer science problems necessary for research and/or innovation activities to develop new knowledge and procedures.
LO3. Clearly and unambiguously communicate own knowledge, conclusions and arguments in the field of computer science to specialists and non-specialists, including students.
LO10. Design architectural solutions for information and computer systems for various purposes.
LO11. Create new algorithms for solving problems in the field of computer science, evaluate their effectiveness and limitations on their application.
LO13. Evaluate and ensure the quality of information and computer systems for various purposes.
LO14. Test the software.
LO15. Identify the needs of potential customers for information processing automation.
LO17. Identify and eliminate problem situations during software operation, formulate tasks for its modification or reengineering.

LO18. Collect, formalize, systematize and analyze the needs and requirements for the information or computer system being developed, operated or maintained.
LO20. Create mathematical models, design software, collect and visualize data sets.

Student workload

The total volume of the course is 120 hours (4 ECTS credits): lectures - 16 hours, laboratory classes - 32 hours, self-study - 72 hours.

Course prerequisites

Programming

Features of the course, teaching and learning methods, and technologies

Lectures are conducted interactively using multimedia technologies. A project approach to learning is used in classes. Study materials are available to students through OneDrive.

Program of the course

Topics of the lectures

Topic 1. Introduction to Information Systems Architecture
Core concepts and definitions. Significance of architecture for efficiency and quality. Architecture disciplines. SA activities. Architecture contexts. Architectural structures. Architecture and Agile
Topic 2. Assessing the Efficiency of Architectural Solutions
Efficiency metrics. Performance analysis and evaluation.
Topic 3. Quality in Architectural Solutions
Maintainability / Modifiability. Performance. Scalability. Reliability. Availability. Security. Testing and validation of architectures. Automation of testing.
Topic 4. Principles of Solution Architecture Design
Scaling workload. Building resilient architecture. Design for performance. Using replaceable resources. Service not server. Using the right storage for the right need. Automating everything
Topic 5. Architectural styles
Common Principles. Architectural Style: Monolith. Architectural Style: Microservice. Architectural Style: Event-Driven. Architectural style: REST.
Topic 6 Architectural patterns
Base Patterns. Session State Patterns. Distribution Patterns. Object-Relational Structural Patterns. Object-Relational Behavioral Patterns. Data Source Architectural Patterns
Topic 7 Cloud Migration and Hybrid Cloud Architecture Design
Benefits of cloud-native architecture. Creating a cloud migration strategy. Steps for cloud migration. Creating a hybrid cloud architecture. Designing a cloud-native architecture. Popular public cloud choices.
Topic 8. Security and Reliability
Security measures in architecture. Ensuring system reliability

Topics of the workshops

Practical work within the discipline is not provided.

Topics of the laboratory classes

Topic 1.
Architectural Diagram Creation
Topic 2.
System Comparison of the architectures of two information systems.
Topic 3.
Performance Metrics Analysis
Topic 4.
Benchmarking Exercise.

Topics 5 - 6.
Testing Architectural Resilience. Automated Testing Implementation.
Topics 7-8.
Load Balancing Implementation.
Topics 9-10.
Implementing Architectural Patterns.
Topic 11-12.
Microservices vs. Event-Driven Architecture
Topics 13-14.
Designing a Hybrid Cloud Architecture.
Topic 15.
Presentation and defense of individual calculation tasks.
Topic 16.
Final test.

Self-study

The course involves the implementation of an individual calculation task on design the architecture of an information system that addresses specific efficiency and quality requirements outlined in the project scenario. Students are expected to incorporate relevant architectural styles, patterns, and best practices discussed during the course. The result of the development is made into a presentation, with a mandatory defense at the end of the course. The grade is assigned based on the result of the open defense of the calculation task before the commission, which consists of the teacher and other students of the group. Students are also recommended additional materials (videos, articles) for independent study and analysis

Course materials and recommended reading

1 Fowler M., Rice D., Foemmel M., Hieatt E. Mee R., Stafford R. Patterns of Enterprise Application Architecture, 2002.
2 Martin R.C. Clean Architecture: A Craftsman's Guide to Software Structure and Design, 2017.
3 Jones C. Measuring Software Quality: A Case Study Approach, 1991.
4 Vance S. Quality Code: Software Testing Principles, Practices, and Patterns, 2013.
5 Bass L., Clements P., Kazman R. Software Architecture in Practice, 2012.
6 Richardson C. Microservices Patterns: With examples in Java, 2018.
7 Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems, 2017.

Assessment and grading

Criteria for assessment of student performance, and the final score structure

100% of the final grade consists of assessment results in the form of final testing (20%) and ongoing assessment (80%).
Final test: 40 test questions, multiple choice questions and open questions are available.
Current assessment: performance of laboratory work (60%) and calculation task (20%).
In case of disagreement with the obtained grade, 100% of the final grade consists of the results of the assessment in the form of an exam. Completion of all laboratory work is a mandatory condition for admission to the exam.
Exam: written task (2 theory questions and a practical task) and an oral presentation.

Grading scale

Total points	National	ECTS
90–100	Excellent	A
82–89	Good	B
75–81	Good	C
64–74	Satisfactory	D
60–63	Satisfactory	E
35–59	Unsatisfactory (requires additional learning)	FX
1–34	Unsatisfactory (requires repetition of the course)	F

Norms of academic integrity and course policy

The student must adhere to the Code of Ethics of Academic Relations and Integrity of NTU "KhPI": to demonstrate discipline, good manners, kindness, honesty, and responsibility. Conflict situations should be openly discussed in academic groups with a lecturer, and if it is impossible to resolve the conflict, they should be brought to the attention of the Institute's management.
Regulatory and legal documents related to the implementation of the principles of academic integrity at NTU "KhPI" are available on the website: http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/

Approval

Approved by	Date, signature	Head of the department Dmitro BRESLAVSKY
	Date, signature	Guarantor of the educational program Oleksii VODKA