

АРХІТЕКТУРА ТА ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Тип дисципліни	Обов'язкова
Рівень вищої освіти	Перший (бакалаврський)
Мова викладання	Українська, Англійська
Семестри	шостий
Обсяг кредитів ЄКТС	5
Форми здобуття освіти	Очна (денна)

Результати навчання. Відповідно до Стандарту вищої освіти та освітньої програми дисципліна має забезпечити:

– **компетентності:** Здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій; здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування; здатність розробляти архітектури, модулі та компоненти програмних систем; здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя; здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення; здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення; здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

– **програмні результати навчання:** аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки; знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення; проводити передпроектне обстеження предметної області, системний аналіз об'єкта проектування; вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання; застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення; застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

Зміст навчальної дисципліни. Вступ до архітектури програм. Моделі, каркаси та зразки проектування. Типи архітектур та їх моделі. Архітектура: нотація, стандарти та інструментальні засоби. Архітектурні шаблони і стилі. Контроль якості при виборі архітектури. Вступ в детальне проектування. Діаграми послідовності та діаграми потоків даних в детальному проектуванні. Специфікація алгоритмів, класів та функцій. Зразки проектування. Бібліотеки стандартних шаблонів. Стандарти, нотація та інструментальні засоби проектування. Вплив детального проектування на проект

Запланована навчальна діяльність: лекції – 34 год, лабораторні заняття – 34 год, самостійна робота – 142 год, разом – 210 год.

Форми (методи) навчання: лекції (з використанням методів візуалізації); лабораторні заняття (з використанням методів комп'ютерного моделювання), самостійна робота

Форми оцінювання результатів навчання: захист лабораторних робіт, тестування, іспит, захист курсового проекту

Форми семестрового контролю: іспит, захист курсового проекту

Навчальні ресурси:

1. Форкун Ю. В., Яшина О.М. Методичні вказівки до виконання курсового проекту з дисципліни «Архітектура та проектування програмного забезпечення» для студентів спеціальності 121 «Інженерія програмного забезпечення»/ Ю. В. Форкун, О.М.Яшина. – Хмельницький: ХНУ, 2023. – 42 с.
2. Конспект лекцій з дисципліни «Архітектура та проектування програмного забезпечення» для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Інженерія програмного забезпечення» із спеціальності 121 – «Інженерія програмного забезпечення» / Укл. В.В.Завгородній, К.М.Ялова. – Кам'янське: ДДТУ, 2019.– 144с.

3. Мартін Р. Чиста архітектура /Роберт Мартін. – Харків: Фабула, 2019. – 416 с.
4. Фрімен Е. Head First. Патерни проектування /Ерік Фрімен, Елізабет Робсон. – Харків: Фабула, 2020. – 672 с.
5. Richards M. Fundamentals of Software Architecture: An Engineering Approach / Mark Richards, NealFord. – Sebastopol, California: O'Reilly Media – 1st edition, 2020. – 419 p.
6. Lanciaux R. Modern Front-end Architecture: Optimize Your Front-end Development with Components, Storybook, and Mise en Place Philosophy / Ryan Lanciaux – New York: Apress, 2021 – 144 p.
7. Frighi V. Smart Architecture – A Sustainable Approach for Transparent Building ComponentsDesign / Valentina Frighi – Berlin: Springer – 1st edition, 2022. – 293 p.
8. Модульне середовище для навчання MOODLE. Доступ до ресурсу : <https://msn.khmnu.edu.ua>
9. Електронна бібліотека університету Доступ до ресурсу : <https://lib.khnu.km.ua>
10. Репозитарій ХНУ. Доступ до ресурсу: <http://elar.khmnu.edu.ua/jspui/?locate=uk>

Викладачі: канд. техн. наук, доцент Яшина О.М., канд. техн. наук, доцент Форкун Ю.В.

SOFTWARE ARCHITECTURE AND DESIGN

Type of discipline	Mandatory
Level of higher education	First (bachelor's)
Language of instruction	Ukrainian
Semesters	sixth
ECTS credits	5
Course study mode	Full-time (full-time)

Learning outcomes. According to the Standard of Higher Education and Educational Program, the discipline must ensure:

Integral Competence (IC): Ability to solve complex, specialised tasks or practical problems in software engineering, characterised by complexity and uncertainty of conditions, using information technology theories and methods.

Special (Professional) Competencies (PC): Ability to participate in software design, including modelling (formal description) of its structure, behaviour, and operational processes. Ability to develop architectures, modules, and components of software systems. Ability to accumulate, process, and systematise professional knowledge regarding the creation and maintenance of software and recognise the importance of lifelong learning. Ability to implement phases and iterations of the life cycle of software systems and information technologies based on relevant software development models and approaches. Ability to execute the system integration process and apply standards and change management procedures to maintain the integrity, overall functionality, and reliability of the software. Ability to reasonably choose and master the toolkit for software development and maintenance.

Program learning outcomes. To analyse, purposefully search for, and select the necessary information, reference resources, and knowledge for solving professional tasks, considering modern scientific and technical achievements. To understand the software lifecycle's leading processes, phases, and iterations. To conduct a pre-project survey of the subject area and system analysis of the design object. To select initial data for design, guided by formal methods of requirement descriptions and modelling. To apply effective software design approaches in practice. To use instrumental software tools in practice for domain analysis, design, testing, visualisation, measurement, and software documentation

The content of the discipline. Introduction to Application Architecture. Models, wireframes, and design samples. Types of architectures and their models. Architecture: Notation, Standards, and Tools. Architectural templates and styles. Quality control when choosing an architecture. Introduction to detailed design. Sequence diagrams and data flow diagrams in detailed design. Specification of algorithms, classes, and functions. Design samples. Libraries of standard templates. Standards, notation, and design tools. The Impact of Detailed Design on the Project.

Planned educational activities: lectures – 34 hours, laboratory classes – 34 hours, individual work – 142 hours, total – 210 hours.

Forms (methods) of training: lectures (using visualization methods); laboratory classes (using computer modeling methods), independent work

Forms of assessment of learning outcomes: defense of laboratory work, testing, exam, defense of a course project

Formi of semester control: exam, defense of the course project

Learning Resources:

1. Форкун Ю. В., Яшина О.М. Методичні вказівки до виконання курсового проекту з дисципліни «Архітектура та проектування програмного забезпечення» для студентів спеціальності 121 «Інженерія програмного забезпечення»/ Ю. В. Форкун, О.М.Яшина. – Хмельницький: ХНУ, 2023. – 42 с.
2. Конспект лекцій з дисципліни «Архітектура та проектування програмного забезпечення» для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Інженерія програмного забезпечення» із спеціальності 121 – «Інженерія програмного забезпечення» / Укл. В.В.Завгородній, К.М.Ялова. – Кам'янське: ДДТУ, 2019.– 144с.
3. Мартін Р. Чиста архітектура /Роберт Мартін. – Харків: Фабула, 2019. – 416 с.
4. Фрімен Е. Head First. Патерни проектування /Ерік Фрімен, Елізабет Робсон. – Харків: Фабула, 2020. – 672 с.

5. Richards M. Fundamentals of Software Architecture: An Engineering Approach / Mark Richards, Neal Ford. – Sebastopol, California: O'Reilly Media – 1st edition, 2020. – 419 p.
6. Lanciaux R. Modern Front-end Architecture: Optimize Your Front-end Development with Components, Storybook, and Mise en Place Philosophy / Ryan Lanciaux – New York: Apress, 2021 – 144 p.
7. Frighi V. Smart Architecture – A Sustainable Approach for Transparent Building Components Design / Valentina Frighi – Berlin: Springer – 1st edition, 2022. – 293 p.
8. Modular learning environment MOODLE. Access to the resource: <https://msn.khmnu.edu.ua>
9. Electronic Library of the University Access to the resource: <https://lib.khnu.km.ua>
10. KhNU Repository. Access to the resource: <http://elar.khmnu.edu.ua/jspui/?locate=uk>

Lecturers: Ph.D. Tech. Doctor of Physical Sciences, Associate Professor Yashina O.M., Ph.D. Tech. Doctor of Physical Sciences, Associate Professor Forkun Y.V.