

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Кафедра \_\_\_\_\_ радіоелектронних  
і комп'ютерних систем \_\_\_\_\_

**“ЗАТВЕРДЖУЮ”**

Декан факультету \_\_\_\_\_  
доц. Юрій ФУРГАЛА  
“ ” 2022 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

спеціальність \_\_\_\_\_  
(шифр і назва навчальної дисципліни)  
121 – Інженерія програмного забезпечення  
(шифр і назва спеціальності)  
спеціалізація \_\_\_\_\_  
(назва спеціалізації)  
факультет \_\_\_\_\_  
електроніки та комп'ютерних технологій  
(назва інституту, факультету, відділення)

2022 – 2023 навчальний рік

Робоча програма “Архітектура програмного забезпечення” для студентів

(назва навчальної дисципліни)

галузі знань “12 – Інформаційні технології”

за спеціальністю “121 Інженерія програмного забезпечення”

Розробники: Ігор ОЛЕНИЧ (доктор фізико-математичних наук, професор,  
завідувач кафедри радіоелектронних і комп'ютерних систем)


Олег СІНЬКЕВИЧ (асистент, кафедри РКС)

(вказати авторів, їхні посади, наукові ступені та вчені звання)

Робочу програму схвалено на засіданні кафедри (циклової комісії) \_\_\_\_\_  
радіоелектронних і комп'ютерних систем

Протокол від “ 31 ” “ 08 ” 2022 року № 1/23

Завідувач кафедри радіоелектронних і комп'ютерних систем

 (Ігор ОЛЕНИЧ)

(підпис)

(прізвище та ініціали)

Ухвалено Вченою радою \_\_\_\_\_ факультету електроніки та комп'ютерних технологій

Протокол від “ 31 ” “ 08 ” 2022 року № 28/22

© Оленич І., 2022 рік

© Сінькевич О., 2022 рік

© ЛНУ ім. І. Франка, 2022 рік

### 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів <b>4,5</b>	Галузь знань <u>12 Інформаційні технології</u> (шифр і назва)	Нормативна	
Модулів – <i>немає</i>	Спеціальність: <u>121 Інженерія програмного забезпечення</u>	Рік підготовки	
Змістових модулів – <b>2</b>		<b>3-й</b>	
Індивідуальне науково-дослідне завдання <u>немає</u> (назва)		Семестр	
Загальна кількість годин – <b>135</b>		<b>6-й</b>	
Тижневих годин для денної форми навчання: аудиторних – <b>4</b> самостійної роботи студента – <b>4,5</b>	Освітній ступінь <u>бакалавр</u>	Лекції	
		<b>32 год.</b>	
		Практичні, семінарські	
		<i>немає</i>	
		Лабораторні	
		<b>32 год.</b>	
		Самостійна робота	
		<b>71 год.</b>	
		Індивідуальні завдання:	
<i>немає</i>			
Вид контролю:			
<i>екзамен</i>			

Примітка.

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить

для денної форми навчання – 0,9

для заочної форми навчання – немає

### 2. Мета та завдання навчальної дисципліни

**Мета:** надати студентам основні поняття про структуру, інформаційні моделі та системи побудови програмного забезпечення

**Цілі:** ознайомлення студентів з основними підходами, інформаційними моделями та системами побудови програмного забезпечення, мовами програмування та проектування програмного забезпечення.

У результаті вивчення навчальної дисципліни студент повинен

**знати:** методологію та технологію збору основних вимог до програмного забезпечення та методи їх аналізу; технології проектування програмного забезпечення; способи використання CASE засобів для аналізу предметного середовища та побудови архітектури програмного забезпечення;

**вміти:** виконувати аналіз вимог до програмного забезпечення, що розробляється; оцінювати трудомісткість і вибирати адекватні підходи до розробки програмного забезпечення; проектувати архітектуру програмного забезпечення з використанням засобів візуального моделювання; усвідомлено застосовувати методики випробувань і налагодження розроблюваного програмного забезпечення; проектувати компоненти архітектурного рішення.

У результаті вивчення курсу «Архітектура програмного забезпечення» здобувачі набудуть таких Загальних та Фахових компетентностей та Програмних результатів навчання:

**ЗК1.** Здатність до абстрактного мислення, аналізу та синтезу.

**ЗК2.** Здатність застосовувати знання у практичних ситуаціях.

**ФК19.** Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

**ФК20.** Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.

**ФК21.** Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки).

**ФК22.** Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

**ФК28.** Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

**ФК29.** Здатність до алгоритмічного та логічного мислення.

**ФК30.** Здатність використовувати для розробки програмного забезпечення перспективні технології, зокрема, системи штучного інтелекту, Fog/Edge-обчислення тощо.

**ФК31.** Володіння методами, серверними технологіями та інструментальними засобами проектування веб-застосувань.

**ФК32.** Здатність здійснювати розробку програмного забезпечення використовуючи різні парадигми програмування (в тому числі паралельне, об'єктно-орієнтоване, функціональне програмування тощо)

**ПРН16.** Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.

**ПРН19.** Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення.

**ПРН20.** Знати підходи щодо оцінки та забезпечення якості програмного забезпечення.

**ПРН21.** Знати, аналізувати, вибирати, кваліфіковано застосовувати засоби забезпечення інформаційної безпеки (в тому числі кібербезпеки) і цілісності

даних відповідно до розв'язуваних прикладних завдань та створюваних програмних систем.

### **3. Програма навчальної дисципліни**

#### **Змістовий модуль 1.**

Тема 1. *Основні відомості про архітектуру ПЗ.*

Особливості розробки великих програмних систем. Архітектура і дизайн. Мета архітектури. Функціональність та структура ПЗ. Роль архітектури та архітектора.

Тема 2. *Парадигми програмування.*

Структурне програмування. Об'єктно-орієнтоване програмування. Інкапсуляція, успадкування, поліморфізм. Функційне програмування.

Тема 3. *Застосування візуального моделювання в процесі розробки ПЗ.*

Цілі та засоби мови UML. Діаграми взаємодії, класів, станів, діяльності, компонентів, розміщення.

Тема 4. *Технології створення ПЗ.*

Загальні вимоги до технологій створення ПЗ. Правила та особливості розробки ПЗ. Проблеми, які виникають при розробці ПЗ.

Тема 5. *Моделі життєвого циклу ПЗ.*

Каскадна модель. Макетування. Інкрементна модель. Швидка розробка додатків. Спіральна та компонентно-орієнтована моделі. XP-процес.

Тема 6. *Класифікація архітектури ПЗ.*

Аналіз архітектури ПЗ з огляду на область застосування. Монолітна, сервісно-орієнтована, мікросервісна та безсерверна архітектура ПЗ.

#### **Змістовий модуль 2.**

Тема 7. *Мікросервісна архітектура в контексті веб-розробки.*

Архітектурний шаблон Модель-Вид-Контролер. Масштабування.

Тема 8. *Мікросервісна архітектура та контейнеризація.*

Мікросервісні патерни. Docker-контейнери. Restful web-сервіси. Безпека сервісів.

Тема 9. *Проектування мікросервісної архітектури.*

Оркестрування та відповідні сучасні засоби. Транзакції. Saga патерн.

Тема 10. *Просторова архітектура.*

Топологія та модулі обробки. Помпа та колізії даних. Хмарні та локальні реалізації. Характеристики. Віртуалізоване проміжне програмне забезпечення.

Тема 11. *Безсерверна архітектура.*

Хмарні нативні застосунки. Продуктивність. Кешування. AWS та безсерверна архітектура.

Тема 12. *Вибір архітектурного стилю.*

Критерії щодо монолітної та розподіленої архітектур. Побудова діаграми рішення.

Тема 13. *Аналіз якості та оцінка програмного дизайну.*

Атрибути якості. Методи аналізу архітектури. Комплексний підхід до оцінки архітектури. Метод аналізу вартості та ефективності.

#### 4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	денна форма						заочна форма					
	усьо го	у тому числі					усього	у тому числі				
		л	п	лаб	інд	с. р.		л	п	лаб	інд.	с. р.
1	2	3	4	5	6	7	8	9	10	11	12	13
<b>Змістовий модуль 1.</b>												
Тема 1. <i>Основні відомості про архітектуру ПЗ</i>		2		2		4,5						
Тема 2. <i>Парадигми програмування.</i>		2		2		4,5						
Тема 3. <i>Застосування візуального моделювання в процесі розробки ПЗ</i>		4		4		9						
Тема 4. <i>Технології створення ПЗ</i>		2		2		4,5						
Тема 5. <i>Моделі життєвого циклу ПЗ.</i>		2		2		4,5						
Тема 6. <i>Класифікація архітектури ПЗ</i>		4		4		8						
Разом за змістовим модулем 1		<b>16</b>		<b>16</b>		<b>35</b>						
<b>Змістовий модуль 2.</b>												
Тема 7. <i>Мікросервісна архітектура в контексті веб-розробки.</i>		2		2		4,5						
Тема 8. <i>Мікросервісна архітектура та контейнеризація.</i>		2		2		4,5						
Тема 9. <i>Проектування мікросервісної архітектури.</i>		4		4		9						

Тема 10. <i>Просторова архітектура.</i>	2	2	4,5						
Тема 11. <i>Безсерверна архітектура.</i>	2	2	4,5						
Тема 12. <i>Вибір архітектурного стилю.</i>	2	2	4,5						
Тема 13. <i>Аналіз якості та оцінка програмного дизайну.</i>	2	2	4,5						
Разом за змістовим модулем 2	<b>16</b>	<b>16</b>	<b>36</b>						
<b>Усього годин</b>	<b>32</b>	<b>32</b>	<b>71</b>						

### 5. Теми семінарських занять

### 6. Теми практичних занять

### 7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Вступне заняття. Формування команд, обрання теми проєкту.	2
2	Організація роботи в команді. Формати командної роботи	2
3	Розробка вимог до проєкту. Збір вимог до ПЗ та їх аналіз.	4
4	Проектування ПЗ. Оцінка тривалості задач проєкту.	2
5	Створення проєкту в спеціалізованих сервісах, додавання коду в проєкт, бази даних та розв'язання конфліктів.	4
6	Підсумкове заняття, Контрольний замір знань 1.	2
7	Реалізація проєкту. Обґрунтування архітектурного стилю. Проведення перегляду коду командою.	2
8	Реалізація проєкту. Дослідження використання мікросервісів та Restful.	2
9	Реалізація проєкту. Контейнеризація.	4
10	Тестування компонентів проєкту та оркестрування процесів.	2
11	Інтеграція та тестування розробки проєкту.	2
12	Підготовка та створення презентації проєкту	2
13	Підсумкове заняття, Контрольний замір знань 2.	2
	<b>Разом</b>	<b>32</b>

## 8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Основні відомості про архітектуру ПЗ	4,5
2	Парадигми програмування.	4,5
3	Застосування візуального моделювання в процесі розробки ПЗ	9
4	Технології створення ПЗ	4,5
5	Моделі життєвого циклу ПЗ.	4,5
6	Класифікація архітектури ПЗ	8
7	Мікросервісна архітектура в контексті веб-розробки.	4,5
8	Мікросервісна архітектура та контейнеризація.	4,5
9	Проектування мікросервісної архітектури.	9
10	Просторова архітектура.	4,5
11	Безсерверна архітектура.	4,5
12	Вибір архітектурного стилю.	4,5
13	Аналіз якості та оцінка програмного дизайну.	4,5
	Разом	<b>71</b>

## 9. Індивідуальні завдання

### 10. Методи навчання

Презентація, лекції, лабораторні роботи, обговорення, дискусія.

### 11. Методи контролю

Поточний контроль здійснюється шляхом проведення контрольних замірів знань у формі тестів та оцінювання виконання лабораторних робіт. У кінці курсу проводиться екзамен у формі захисту командних проєктів.

## 12. Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота													Підсумковий тест (екзамен)	Сума
Змістовий модуль 1						Змістовий модуль 2							50	100
T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13		
4	4	4	4	4	5	3	3	4	4	4	4	3		

T1, T2 ... T13 – теми змістових модулів.



### Шкала оцінювання: національна та ЄКТС

Оцінка ЄКТС	Сума балів за всі види навчальної діяльності	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
A	90 – 100	відмінно	зараховано
B	81-89	добре	
C	71-80		
D	61-70	задовільно	
E	51-60		
FX	21-50	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
F	0-20	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

\* кількість балів для оцінок «незадовільно» (FX і F) визначається Вченими радами факультетів (педагогічними радами коледжів).

### 13. Методичне забезпечення

1. Електронний курс «Архітектура програмного забезпечення». - <http://194.44.208.156/moodle/course/view.php?id=138>

### 14. Рекомендована література

#### Основна

1. Martin R.C. Clean Architecture: A Craftsman's Guide to Software Structure and Design / Robert C. Martin. - Pearson, 2017. - 432 p.
2. Clements P., Bachmann F., Bass L. Documenting Software Architectures: Views and Beyond (2nd Edition). - Boston: Addison-Wesley Professional, 2010.
3. Goma H. Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures. – NY: Cambridge University Press, 2011. – 578 p.
4. James Gough. Mastering API Architecture: Design, Operate, and Evolve API-Based Systems / J. Gough, D. Bryant, M. Auburn. O'Reilly Media; 1st edition, 2022. - 286 p.

### **Допоміжна**

5. Bewis T. C# Design Pattern Essentials. – NY: Ability First Limited, 2012. – 264 p.
6. Maciaszek L. Requirements analysis and systems design / L. Maciaszek. – Ontario: Pearson Education Canada, 2007. - 656 p.
7. Richards M. Fundamentals of Software Architecture. An Engineering Approach / M. Richards, N. Ford. - Sebastopol, CA: O'Reilly Media, 2020. - 422 p.
8. Brown S. Visualise, document and explore your software architecture. Software Architecture for Developers / Simon Brown. - Leanpub, 2019.
9. Eyskens S. Software Architecture for Busy Developers / S. Eyskens. - Birmingham: Packt Publishing, 2021. - 174 p.

### **15. Інформаційні ресурси**

1. Internet – джерела.
2. Наукова бібліотека Львівського національного університету імені Івана Франка (<https://www.lnulibrary.lviv.ua/to-users-2/paid-services/internet/> ).
3. Львівська національна наукова бібліотека України імені Василя Стефаника (<https://www.lsl.lviv.ua/index.php/uk/elektronni-resursy1/> ).