

# МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ПРИРОДОКОРИСТУВАННЯ

Факультет механіки, енергетики та інформаційних технологій  
Кафедра інформаційних технологій



## РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### *Методологія DevOps*

ОПШ «Комп'ютерні науки»  
спеціальність: 122 «Комп'ютерні науки»  
першого (бакалаврського) рівня вищої освіти

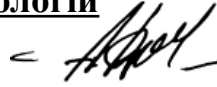
Львів 2024 р.

Робоча програма навчальної дисципліни **Методологія DevOps** для студентів спеціальності 122 «Комп'ютерні науки», першого (бакалаврського) рівня вищої освіти

Розробники: к.ф.-м.н. Чухрай Л.В., к.т.н., доц. Луб П.М.

Робочу програму схвалено на засіданні кафедри **Інформаційних технологій**  
Протокол № 1 від 12 серпня 2024 року

Завідувач кафедри **Інформаційних технологій**



(підпис)

(Тригуба А.М.)

(прізвище та ініціали)

Робочу програму схвалено на засіданні методичної комісії факультету механіки, енергетики та інформаційних технологій Протокол № 1 від 29 серпня 2024 року.

Голова методичної комісії факультету механіки, енергетики та інформаційних технологій.



(підпис)

(Ковалишин С.Й.)

(прізвище та ініціали)

## 1. Опис навчальної дисципліни

Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень

Освітній ступінь: Бакалавр

Галузь знань 12 Інформаційні технології

(шифр і назва)

Спеціальність 122 «Комп'ютерні науки»

Характеристика навчальної дисципліни:

Вибіркова

Кількість кредитів 3

Загальна кількість годин – 90

Індивідуальне науково-дослідне завдання \_\_\_\_\_

(назва)

Вид контролю: Залік

Тижневих аудиторних годин для денної форми навчання – 3

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить (%):

для денної форми навчання – 53,3%;

для заочної форми навчання – 22,2%

## 2. Програма навчальної дисципліни

### Тема 1. Вступ до DevOps:

Історія та філософія DevOps

Основні принципи та практики

### Тема 2. Інструменти та технології:

Огляд ключових інструментів (наприклад, Jenkins, Docker)

Автоматизація процесів

### Тема 3. Управління версіями та співпраця:

Git і системи контролю версій

Стратегії гілкування та співпраця в команді

### Тема 4. Неперервна інтеграція (CI) та неперервна доставка (CD):

Процеси CI/CD

Практичне впровадження CI/CD

### Тема 5. Тестування та якість коду:

Автоматизація тестування

Стратегії забезпечення якості

### Тема 6. Інфраструктура як код (IaC):

Принципи та практики IaC

Інструменти, такі як Terraform та Ansible

**Тема 7. Моніторинг та логування:**

Системи моніторингу та логування

Важливість моніторингу в DevOps

**Тема 8. Культура та практика DevOps:**

Розвиток культури DevOps

Взаємодія команд розробки та експлуатації

**3. Структура навчальної дисципліни**

Назви тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	пр	лаб.	інд.	с.р.		л	пр	лаб.	інд.	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
	Рік підготовки 3 Семестр 6						Рік підготовки 3 Семестр 6					
<i>Семестр 6</i>												
Тема 1.	11	2	4			5	14	2	2			10
Тема 2.	11	2	4			5	14	2	2			10
Тема 3.	11	2	4			5	10					10
Тема 4.	11	2	4			5	12	2	2			8
Тема 5.	11	2	4			5	8					8
Тема 6.	11	2	4			5	12	2	2			8
Тема 7.	12	2	4			6	8					8
Тема 8.	12	2	4			6	12	2	2			8
Іспит	0	-	-			0	0					0
<b>Усього годин</b>	<b>90</b>	<b>16</b>	<b>32</b>	<b>0</b>	<b>0</b>	<b>42</b>	<b>90</b>	<b>10</b>	<b>10</b>	<b>0</b>	<b>0</b>	<b>70</b>
<b>Індивідуальні завдання</b>												
КР	0					0	0					0
<b>Усього годин</b>	<b>90</b>	<b>16</b>	<b>32</b>	<b>0</b>	<b>0</b>	<b>42</b>	<b>90</b>	<b>10</b>	<b>10</b>	<b>0</b>	<b>0</b>	<b>70</b>

**4. Перелік лабораторно-практичних занять**

№ з/п	Назва теми	Кількість, год.
<i>Семестр 6</i>		
1	<b>Створення базового DevOps робочого потоку</b> Задача: Створити схематичне зображення робочого потоку DevOps для невеликого проекту, включаючи етапи розробки, тестування, інтеграції, доставки та впровадження.	2
2	<b>Аналіз кейсу</b> Задача: Аналізувати реальний кейс використання DevOps в індустрії, обговорюючи виклики, рішення та результати.	2
3	<b>Робота з Docker</b> Задача: Створити Docker контейнер для простого веб-додатку.	2
4	<b>Використання Jenkins</b>	2

	Задача: Налаштувати Jenkins для автоматичного розгортання коду з GitHub репозиторію.	
5	<b>Вправи з Git</b> Задача: Виконати серію завдань з використанням Git, включаючи гілкування, злиття та вирішення конфліктів.	2
6	<b>Симуляція командної роботи</b> Задача: Працювати в групах, використовуючи Git для спільної розробки міні-проекту.	2
7	<b>Налаштування CI з Jenkins</b> Задача: Створити CI пайплайн в Jenkins для автоматичного тестування коду.	2
8	<b>Реалізація CD</b> Задача: Розгортання веб-додатку на хмарному сервісі через CD пайплайн.	2
9	<b>Автоматизація тестування</b> Задача: Написання автоматизованих тестів для вже існуючого додатку.	2
10	<b>Аналіз якості коду</b> Задача: Використання інструментів для аналізу якості коду та вдосконалення кодової бази.	2
11	<b>Введення в Terraform</b> Задача: Створення простої інфраструктури за допомогою Terraform.	2
12	<b>Робота з Ansible</b> Задача: Автоматизація налаштувань серверу за допомогою Ansible.	2
13	<b>Налаштування системи моніторингу</b> Задача: Встановлення та налаштування Prometheus для моніторингу веб-додатку.	2
14	<b>Логуювання та аналіз даних</b> Задача: Використання ELK Stack для збору та аналізу логів додатку.	2
15	<b>Розвиток DevOps культури</b> Задача: Розробка плану впровадження DevOps культури в організації.	2
16	<b>Кейс-стаді з DevOps</b> Задача: Створення детального звіту про інтеграцію DevOps практик у вигадану компанію.	2

### 5. Теми винесені на самостійне вивчення

№ з/п	Назва теми
1	<b>Основи Lean методології:</b> Студенти досліджують принципи Lean методології та її вплив на ефективність процесів розробки ПЗ.
2	<b>Знайомство з Kubernetes:</b> Вивчення основ Kubernetes як системи оркестрації контейнерів та її ролі в масштабуванні додатків.
3	<b>Роль автоматизації у DevOps:</b> Дослідження різних аспектів та переваг автоматизації в процесах DevOps.
4	<b>Основи неперервного вивчення в DevOps:</b> Ознайомлення з концепцією неперервного вивчення та її значенням для постійного вдосконалення в DevOps.
5	<b>Безпека в DevOps (DevSecOps):</b> Вивчення інтеграції практик безпеки в DevOps процеси та принципи DevSecOps.
6	<b>Мікросервісна архітектура:</b> Дослідження основ мікросервісної архітектури та її переваг для гнучкої розробки та впровадження.

7	<b>Знайомство з облічними хмарами:</b> Ознайомлення з базовими концептами хмарних обчислень та їх роллю в DevOps.
8	<b>Культурні зміни при впровадженні DevOps:</b> Вивчення важливості культурних змін у організаціях для успішного впровадження DevOps.

## 6. Індивідуальні завдання

### 7. Методи навчання

**1. Словесні методи** (розповідь, пояснення, бесіда, лекція.)

**2. Наочні методи:**

- ілюстрація (презентації, таблиці, моделі, муляжі, малюнки тощо);  
 - демонстрування методики, об'єкту вивчення: діюча методика; модель; база даних; інтерактивний ресурс тощо;

**3. Практичні методи:** практичні та самостійні роботи.

### 8. Методи контролю:

**1. Усне опитування** (фронтальне, індивідуальне, детальний аналіз відповідей студентів).

**2. Письмова аудиторна та поза аудиторна перевірка** (розв'язок поставленого завдання, вирішення задачі, підготовка виступу, реферату, контрольні роботи (з конкретних питань тощо)).

**3. Практична перевірка** (виконання практичної роботи, аналіз виробничої інформації, рішення професійних завдань і т.д.).

**4. Стандартизований контроль** (письмовий іспит).

*Види контролю:* Поточний контроль, проміжна та семестрова атестація.

### 9. Очікувані результати навчання з дисципліни:

Очікуваними результатами навчання з дисципліни «Методологія DevOps» є набуття студентами *інтегральних компетентностей* – здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов.

*Загальних компетентностей* – ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності. ЗК9. Здатність працювати в команді. ЗК11. Здатність приймати обґрунтовані рішення. ЗК12. Здатність оцінювати та забезпечувати якість виконуваних робіт.

*Спеціальних (фахових) компетентностей* – СК10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника. СК12. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників ефективності функціонування операційних систем і системного програмного забезпечення. СК17. Здатність проектувати та реалізовувати програмне забезпечення для вирішення різноманітних прикладних задач інтелектуальної комп'ютеризації у сфері природокористування із використанням різних мов програмування, сучасних бібліотек

крос-платформного програмування, проводити тестування на різних апаратних платформах, впроваджувати і підтримувати роботу інформаційних систем на сучасних платформах.

**Програмні результати** – ПРН11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вмінні розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт). ПРН13. Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем.

## 10. Розподіл балів, які отримують студенти

### Семестр 6

Поточне тестування та самостійна робота (разом 100 балів)										Сума
Модуль 1					Модуль 2					
T1	T2	T3	T4	M1	T5	T6	T7	T8	M2	100
10	10	10	10	10	10	10	10	10	10	

T1, T2 ... T8 – теми, M1, M2 – модулі.

## 11. Методичне забезпечення

Навчально-методичні матеріали до лабораторно-практичних занять; підручники і навчальні посібники; контрольні роботи; текстові та електронні варіанти тестів для поточного і підсумкового контролю, методичні матеріали для самостійної роботи студентів та виконання завдань.

## 12. Рекомендована література

### Базова

1. Michael Hüttermann. DevOps for Developers. Apress, 2012. – 184 p.
2. Viktor Farcic. The DevOps 2.0 Toolkit. Viktor Farcic, 2016 – 404 p.
3. Len Bass, Ingo Weber, Liming Zhu. DevOps A Software Architect's Perspective. Addison-Wesley, 2015. – 338p.
4. DevOps For Dummies. IBM Limited Edition, 2014 – 51 p.
7. Humble, Jez, and Farley, David. Continuous Delivery . Upper Saddle River, NJ: Addison-Wesley, 2010.
8. Poppendieck, Mary, and Thomas David Poppendieck. Implementing Lean Software Development . Upper Saddle River, NJ: Addison-Wesley, 2007.
9. Walls, Mandi. Building a DevOps Culture . Sebastopol, CA: O'Reilly Media, 2013.
10. Dekker, Sidney. Field Guide to Understanding Human Error . Farnham, UK: Ashgate Publishing, 2006.
11. DevOps Revealed 3rd edition. International DevOps Certification Academy.- 94 p. [Electronic resource]. – Access mode <https://www.devops-certification.org/>

### Допоміжна

12. Greaves, Karen, and Samantha Laing. Collaboration Games from the Growing Agile Toolbox . Victoria, BC: Leanpub/Growing Agile, 2014.
13. Cowie, Jon. Customizing Chef . Sebastopol, CA: O'Reilly Media, 2014.
14. Dixon, Jason. Monitoring with Graphite . Sebastopol, CA.: O'Reilly Media, 2015.

### 13. Інформаційні ресурси

1.Бібліотечно-інформаційні ресурси – [книжковий фонд](#), періодика та фонди на [електронних носіях](#) бібліотеки ЛНУП, державних органів науково-технічної інформації, наукових, науково-технічних бібліотек та інших наукових бібліотек України.

2.Електронні інформаційні ресурси мережі Інтернет з переліком сайтів:

1.Міністерство освіти і науки, молоді та спорту України <http://www.mon.gov.ua>, [www.osvita.com](http://www.osvita.com).

2. Allspaw, John. «A Mature Role for Automation: Part 1». KitchenSoap.com, September 21, 2012. <http://bit.ly/allspaw-automation>

3. Caum, Carl. «Continuous Delivery vs. Continuous Deployment: What's the Diff?» Puppet blog, August 30, 2013. <http://bit.ly/cd-vs-cd>

4. Coutinho, Rodrigo. «In Support of DevOps: Kanban vs. Scrum». DevOps.com, July 29, 2014. <http://bit.ly/kanban-v-scrum>

5. Humble, Jez. «Deployment pipeline anti-patterns». <http://bit.ly/humbleantipatterns>

6. Kim, Gene. Kanbans and DevOps: «Resource Guide for Phoenix Project (Part 2).» IT Revolution Press, N.d. <http://bit.ly/kanbans-devops>

7. Arrested DevOps (<https://www.arresteddevops.com/>)

8. DevOps Cafe Podcast with John Willis and Damon Edwards (<http://devopscafe.org/>)