

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

ЗАТВЕРДЖУЮ
В. о. декана факультету комп'ютерних
інформаційних технологій

_____ Ігор ЯКИМЕНКО

" ____ " _____ 2023 р.

ЗАТВЕРДЖУЮ
Директор навчально-
наукового інституту новітніх
освітніх технологій

_____ Святослав ПИТЕЛЬ

" ____ " _____ 2023 р.

ЗАТВЕРДЖУЮ
В. о. проректора з
науково-педагогічної роботи

_____ Віктор ОСТРОВЕРХОВ

" ____ " _____ 2023 р.

**РОБОЧА ПРОГРАМА
з дисципліни «Сучасні практики DevOps»**

Ступінь вищої освіти: бакалавр

Галузь знань – 12 «Інформаційні технології»

Спеціальність – 121 «Інженерія програмного забезпечення»

Освітньо-професійна програма – «Інженерія програмного забезпечення»

Кафедра комп'ютерних наук

Форма навчання	Курс	Семестр	Лекції (год.)	Лаб. (год.)	ІРС (год.)	Тренінг КПЗ (год.)	СРС (год.)	Разом (год.)	Залік (сем.)
денна	3	8	24	16	2	6	102	150	8
заочна	3	10,11	8	4	-	-	138	150	11

Робоча програма розроблена доцентом кафедри комп'ютерних наук,
к.т.н., Шпінталь Михайлом Ярославовичем.

Робоча програма затверджена на засідання кафедри комп'ютерних наук,
протокол № _____ від _____ 20____ р.

Завідувач кафедри д.т.н., професор _____ Андрій Пукас

Розглянуто та схвалено групою забезпечення спеціальності Інженерія
програмного забезпечення, протокол № _____ від _____ 20____ р.

Голова групи
забезпечення спеціальності,
д.т.н., професор _____ Микола Дивак

Гарант ОП _____

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Сучасні практики DevOps»

1. Опис дисципліни «Сучасні практики DevOps»

Дисципліна – «Сучасні практики DevOps»	Галузь знань, спеціальність, СВО	Характеристика навчальної дисципліни
Кількість кредитів – 5	Галузь знань - 12 Інформаційні технології»	Статус дисципліни: вибіркова Мова навчання: українська
Кількість залікових модулів - 3	Спеціальність :121 Інженерія програмного забезпечення	Рік підготовки: <i>Денна – 4</i> <i>Заочна – 4</i> Семестр: <i>Денна – 8</i> <i>Заочна – 10,11</i>
Кількість змістових модулів - 2	Ступінь вищої освіти : бакалавр	Лекції: <i>Денна–24 год.</i> <i>Заочна – 8 год.</i> Лабораторні заняття: <i>Денна – 16 год.</i> <i>Заочна – 4 год.</i>
Загальна кількість годин - 150		Самостійна робота: <i>Денна – 102 год.</i> тренінг – 6 год. <i>Заочна - 138 год.</i> Індивідуальна робота : (КПЗ) – 2 год.
Тижневих годин – 18 год., з них аудиторних – 5 год.		Вид підсумкового контролю – залік

2. Мета й завдання вивчення дисципліни

«Сучасні практики DevOps»

2.1. Мета вивчення дисципліни

Мета дисципліни «Сучасні практики DevOps» забезпечити теоретичну та практичну основу для розуміння та впровадження практик DevOps у розробці та експлуатації програмного забезпечення.

DevOps — це культурний зсув і набір практик, які наголошують на співпраці, комунікації, автоматизації та безперервній доставці в розробці та експлуатації програмного забезпечення. Набуті знання та навички дозволяють забезпечити системний підхід до впровадження цих практик. По суті devops – це спосіб мислення та роботи. Це свого роду каркас, який служить для того, щоб ділитися історіями та розвивати емпатію. Завдяки DevOps окремі люди та групи можуть ефективно та безперервно розвивати свої навички. Це частина культурної «тканини», яка формує способи виконання роботи, а також створює мотивацію для цієї роботи.

2.2. Завдання вивчення дисципліни

формування у студентів системи знань, необхідних при впровадженні концепції DevOps, а саме організації взаємодії між окремими співробітниками

та командами з розробки програмного забезпечення, застосування відповідних інструментів впровадження, масштабування організації.

2.3 Найменування та опис компетентностей, формування котрих забезпечує вивчення дисципліни

знання сучасних підходів до організації взаємодії між окремими співробітниками та командами з розробки програмного забезпечення, інструментів впровадження DevOps та існуючих практик-

2.4. Передумови для вивчення дисципліни.

- Володіння методами ООП.
- - Навики роботи у середовищі розробки програм.
- Володіння методами та засобами проектування.

2.5. Результати навчання

знання основ DevOps, основних засад співробітництва та близькості, особливостей застосування інструментів впровадження DevOps та масштабування організацій.

3. Програма навчальної дисципліни «Сучасні практики DevOps»

Тема 1. Введення в DevOps.

Ознайомлення з DevOps (термін, концепція, роль, структура, методологія). Методології розробки ПЗ: SDLC, waterfall, agile тощо. Інструменти: SCM, DB, CI/CD, Cloud, Containers, Monitoring, Security, Collaboration тощо. Процеси: автоматизація збірки, безперервна інтеграція та безперервне розгортання. Git

Література: 1-10.

Тема 2. Віртуалізація та основи хмарних сервісів.

Розуміння, які існують типи віртуалізації, типи та функції гіпервізора, популярні гіпервізори. VirtualBox (GUI, CLI, створення віртуальної машини, спільна папка, доступ до USB, знімки, налаштування мережі). Vagrant. Типи дата центрів, стандарти. Основні хмарні постачальники (AWS, Azure, GCP). Порівняння публічної, приватної та гібридної хмари. AWS і хмара. Розуміння AWS. Огляд контейнерів проти віртуальних машин. LXC, Docker, K8s. Інші технології: OpenStack, OpenShift. Основи облікового запису: безкоштовний рівень AWS, створення облікового запису AWS, навігація по консолі AWS. Вступ до основних служб AWS: VPC, EC2, RDS, S3, Route 53, ELB, Lambda, ECS

Тема 3. Адміністрування Linux за допомогою Bash. Мережі з використанням Linux

Вміти адмініструвати Linux-подібні OS. IP-маршрутизація та усунення несправностей. Основи безпеки мережі. Структура команди bash: прості, конвеєр, список. Метасимволи інтерпретатора команд.

Тема 4. Використання Python у DevOps.

Вміти застосовувати віртуальне середовище Python. Основні типи даних, змінні, типові математичні операції, оператори порівняння/умови, операції з файлами. Колекції та цикли. List, Dict, OrderedDict, tuple, namedtuple, set. Цикли: for, while, continue, break. Функції (процедури / vs методи). Знання для написання автоматизаційних скриптів.

Тема 5. CI/CD. Безперервна інтеграція і безперервна доставка і безперервне розгортання

Знання безперервного розгортання. Конвеєр CI/CD, Jenkins як структура CI/CD. Передумови для встановлення Jenkins. Кращі практики. Керування Jenkins: захист Jenkins, керування обліковими даними, керування плагінами, резервне копіювання Jenkins, створення підлеглої збірки Деталі збірки: ваша перша збірка, робота з Github, збірка з проекту Github, планування збірок, зв'язування збірок – вихідний і вихідний, проект Freestyle Maven, керування віддаленими системами за допомогою Jenkins, параметризовані збірки Плагін Pipeline і Jenkinsfile Розгортання з Jenkins.

Тема 6. Контейнеризація: Docker і Kubernetes.

Ознайомлення з Docker, розуміння різниці: контейнери проти віртуальних машин, архітектура Docker. Docker -hub. Встановлення та налаштування Docker. Життєвий цикл контейнера, зображення та керування контейнером Створення нашого першого зображення. Робота з кількома зображеннями. Упаковка індивідуального контейнера Запуск команд контейнера за допомогою Docker. Відкриття нашого контейнера з перенаправленнями портів і томами. Файл Docker. Створення та керування кластерами у Kubernetes.

Тема 7. Керування конфігурацією: Ansible.

Вміти автоматично розгортати ПЗ.Знайомство з Ansible. Ansible проти інших класичних інструментів, таких як Chef, Puppet, SaltStack. Введення в YAML. Приклади YAML. Ansible документація Встановлення та налаштування. Виконання команд Ansible. Збір фактів Заміна змінної. Заява про налагодження. Сповідення та обробники Ansible Playbooks. Структура Playbook з YAML. Наша перша п'єса Загальні модулі Ansible. Використання командного рядка Ansible.

Тема 8. Інфраструктура як код: Terraform.

Вміти розгорнути інфраструктуру за допомогою коду. Монтаж Terraform. Синтаксис HCL, провайдери Terraform. Ініціалізація, планування та застосування конфігурації інфраструктури Terraform віддалений стан і сервери; Неявні та явні залежності, вхідні та вихідні змінні. Огляд модулів Terraform.

4. Структура залікового кредиту дисципліни «Сучасні практики DevOps»

денна форма навчання	Кількість годин					
	Лекції	Лабораторна робота	СРС	ІРС	Тренінг КППЗ	Контрольні заходи
Змістовий модуль 1. Вступ в DevOps						
Тема 1. Введення в DevOps.	2	1	12	1	3	Усне опитування/тестування
Тема 2. Віртуалізація та основи хмарних сервісів..	2	1	12			Усне опитування/тестування
Тема 3. Адміністрування Linux за допомогою Bash. Мережі з використанням Linux	5	2	18			Усне опитування/тестування
Тема 4. Використання Python у DevOps.	4	2	12			Усне опитування/тестування
Змістовий модуль 2. Сучасні практики DevOps						
Тема 5. CI/CD. Безперервна інтеграція і безперервна доставка і безперервне розгортання	4	2	12	1	3	Усне опитування/тестування
Тема 6. Контейнерізація: Docker і Kubernetes.	4	1	12			Усне опитування/тестування
Тема 7. Керування конфігурацією: Ansible.	3	1	12			Усне опитування/тестування
Тема 8. Інфраструктура як код: Terraform.	4	2	12			Усне опитування/тестування
Разом	24	12	102	2	6	

заочна форма навчання	Кількість годин		
	Лекції	Лаборатор на робота	СРС
Тема 1. Введення в DevOps.	1		17
Тема 2. Віртуалізація та основи хмарних сервісів..	1	1	17
Тема 3. Адміністрування Linux за допомогою Bash. Мережі з використанням Linux	1		19
Тема 4. Використання Python у DevOps.	1		17
Тема 5. CI/CD. Безперервна інтеграція і безперервна доставка і безперервне розгортання	1	1	17
Тема 6. Контейнерізація: Docker і Kubernetes.	1		17
Тема 7. Керування конфігурацією: Ansible.	1	1	17
Тема 8. Інфраструктура як код: Terraform.	1		17
Разом	8	4	138

5. Тематика лабораторних робіт

5. Тематика лабораторних занять

1. Лабораторне заняття № 1 (4год.)

Тема: Віртуалізація та основи хмарних сервісів.

Мета:

Зрозуміти, які існують типи віртуалізації, типи та функції гіпервізора, популярні гіпервізори. VirtualBox (GUI, CLI, створення віртуальної машини, спільна папка, доступ до USB, знімки, налаштування мережі).

Лабораторне заняття № 2 (4год.)

Тема: Адміністрування Linux за допомогою Bash. Мережі з використанням Linux

Мета: Навчитися адмініструвати Linux-подібні OS. IP-маршрутизація та усунення несправностей. Основи безпеки мережі. Структура команди bash: прості, конвеєр, список. Метасимволи інтерпретатора команд.

Лабораторне заняття № 3(4год.)

Тема: Використання Python у DevOps

Мета: Навчитися застосовувати віртуальне середовище Python. Основні типи даних, змінні, типові математичні операції, оператори порівняння/умови, операції з файлами. Колекції та цикли.

Лабораторне заняття № 4 (4год.)

Тема: Контейнеризація: Docker і Kubernetes

Мета: Ознайомлення з Docker, розуміння різниці: контейнери проти віртуальних машин, архітектура Docker. Docker -hub. Встановлення та налаштування Docker. Життєвий цикл контейнера, зображення та керування контейнером

6. Комплексне практичне індивідуальне завдання

Індивідуальні завдання з дисципліни «Сучасні практики DevOps» виконується самостійно кожним студентом. КППЗ охоплює усі основні теми дисципліни

«Сучасні практики DevOps». Метою виконання КППЗ є оволодіння навичками застосування теоретичних знань. КППЗ оформлюється у відповідності з встановленими вимогами.

Виконання КППЗ є однією із обов'язкових складових модулів залікового кредиту з

дисципліни «Сучасні практики DevOps».

Варіанти КППЗ з дисципліни «Сучасні практики DevOps»:

Автоматизація процесів неперервної інтеграції/неперервного розгортання (CI/CD)

Завдання 1: Створення повного конвеєра CI/CD для мікросервісної архітектури з використанням Jenkins, Travis CI або GitLab CI.

Імплементация моніторингу та логування у комплексній інфраструктурі

Завдання 2: Розгортання стеку ELK (Elasticsearch, Logstash, Kibana) або Prometheus з Grafana для моніторингу роботи додатку та інфраструктури.

Контейнеризація додатків з Docker та Kubernetes

Завдання 3: Контейнеризація існуючого веб-додатку і його розгортання в кластері Kubernetes з використанням Helm чартів.

Автоматизація тестування за допомогою інфраструктури як коду (IaC)

Завдання 4: Розробка Terraform планів або Ansible плейбуків для автоматичного розгортання та тестування інфраструктури.

Оптимізація безпеки DevOps процесів (DevSecOps)

Завдання 5: Впровадження інструментів сканування вразливостей коду та контейнерів, наприклад, SonarQube та Clair, в CI/CD конвеєр.

Конфігурація і управління хмарними сервісами

Завдання 6: Використання AWS, Azure або GCP для розгортання інфраструктури та сервісів, що підтримують DevOps процеси.

Автоматизація та оптимізація процесів резервного копіювання та відновлення

Завдання 7: Створення стратегії резервного копіювання та плану відновлення після збою для многозонової хмарної інфраструктури.

Розробка та впровадження чатботів для оптимізації DevOps процесів

Завдання 8: Інтеграція Slack або Microsoft Teams з інструментами DevOps для надсилання сповіщень та керування задачами через чатботів.

Створення високодоступної (High-Availability) інфраструктури

Завдання 9: Планування та реалізація високодоступної інфраструктури з використанням кластерів, балансувальників навантаження та реплікації даних.

Впровадження політик управління конфігурацією та версіонування

Завдання 10: Розробка політик управління конфігурацією за допомогою GitOps та інструментів версіонування, таких як Git.

7. Самостійна робота

1. Що таке DevOps?

2. Ви набираєте google.com у браузері. Розкажіть якомога детальніше, що відбувається в цей час?

3. Як працює HTTPS?

4. Поясніть концепцію Infrastructure as Code, для чого це потрібно і які проблеми вирішує?

Linux

5. Опишіть загальну архітектуру операційної системи.

6. Опишіть основне призначення операційної системи.

7. Навіщо потрібні файлові системи? Які існують?

8. У чому різниця між віртуалізацією і контейнеризацією?

9. У чому переваги контейнерів?

10. Яка файлова структура у Linux (UNIX) систем, що розташовані в / etc, / dev, / proc, / sys, / lib, / var (кілька директорій на вибір)?

11. Що таке Load Average?

12. У чому різниця між soft та hard symlink?

13. Як працюють file permissions, навіщо директорії права виконання (+x)?

14. Що таке zombie process?

15. За допомогою чого можна зібрати інформацію про поточний стан процесора, пам'яті, диска, мережі?

16. Що таке swappiness?

17. Як подивитися вільне місце на диску?

18. Що таке inode?

19. Розкажіть поетапно процес завантаження Linux від моменту ввімкнення живлення комп'ютера.

20. Що станеться під час виконання команд:

1. cat file1 > file2
2. cat file1 >> file2

21. У чому різниця між Ctrl+C та Ctrl+Z?
22. Як перенаправити одночасно stderr та stdin?
23. Як вбити процес? Які є типи сигналів?
24. Що робить команда grep?
25. Що таке скрипт bash?
26. Які типи змінних використовують у bash?
27. Що виведуть команди:
 1. echo \${hostname};
 2. echo \$(hostname);

Networks

28. Що таке модель OSI, TCP/IP?
29. Для чого потрібні network masks?
30. Структура IP-пакета. З чого складається? Що таке фрагментація та чому вона відбувається?
31. Що таке колізія? Чому виникає?
32. Що таке проксі?
33. Що таке firewalls і навіщо вони потрібні?
34. Що таке NAT і навіщо він потрібен?
35. Які типи IP-адрес ви знаєте?
36. За яким портом і протоколом працюють Ping і Traceroute?

Clouds

37. У чому різниця між IaaS, PaaS та SaaS?
38. Що таке VPC і з яких компонентів має складатись?
39. Що таке cloud-init? init/systemd/upstart configs?

Automation

40. Що таке Iaas і навіщо він потрібний?
41. Що таке Terraform?
42. Які інструменти автоматизації ви знаєте?

Information Security

43. У чому різниця між аутентифікацією та авторизацією?
44. Сертифікати. Як працює HTTPS? Що таке certificate ciphers?
45. Як безпечно передати дані своєму колезі?
46. Що таке MFA, TOTP?

Віртуалізація

47. У чому різниця між віртуалізацією та контейнеризацією? У чому плюси та мінуси?

48. Як для при запуску Docker-контейнера «повісити» його з 80-го порту в контейнері на 8081 на хост?
49. Як передати у віртуальну машину USB device?
50. Docker-контейнер споживає багато SWAP. Що робити?
- CI/CD
51. Що таке Continuous Integration та Continuous Deployment? У чому різниця між Continuous Deployment і Continuous Delivery?
52. Опишіть основні етапи CI/CD.
53. Опишіть приклад процесу CI (та/або CD), який починається з моменту, коли розробник запушив зміни/PR до Git?
54. Розкажіть про різновиди тестів, які ми можемо використовувати в CI пайплайні.
55. Які інструменти CI ви використовували? Чи є досвід роботи з Jenkinsfile?
56. Які види тестів ви знаєте і навіщо вони потрібні?
- Development
57. Git. Як вирішити merge conflict? Що таке rebase, cherry-pick?
58. У чому різниця між git merge та git rebase?
59. Які UI використовували?
60. Яка різниця між GitLab/GitHub/Bitbucket?
61. Яка різниця між Git pull/Git fetch?
62. Що таке Git-flow?
63. Версіонування. Яка різниця між SemVer та CalVer?
64. Тестування. Які існують види? Як писати тести, TDD?
65. У чому різниця між компільованими та інтерпретаційними мовами програмування?

Monitoring/Logging

66. Які метрики треба збирати? Різниця між infrastructure та application monitoring.
67. Яка різниця між pull та push model у системах моніторингу?
68. Яка різниця між Black box та White box monitoring?
69. Розкажіть про підходи до збору application логів.

8. Тренінг з дисципліни

Тематика: Розгортання програмного забезпечення на базі Ansible та Chef, Puppet, SaltStack. Введення в YAML.

Порядок проведення:

1. Здійснити аналіз задачі.
2. Розробити структуру пакетів.
3. Розробити структуру класів.

4. Імплементувати методи класів.

9. Засоби оцінювання та методи демонстрування результатів навчання

У процесі вивчення дисципліни «Сучасні практики DevOps» використовуються наступні засоби оцінювання та методи демонстрування результатів навчання:

- поточне опитування;
- залікове модульне тестування та опитування;
- реферати;
- презентації результатів виконання завдань;
- оцінювання результатів КППЗ;
- ректорська контрольна робота;
- тренінги;
- залік.

10. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни «Сучасні практики DevOps» визначається як середньозважена величина, залежно від питомої ваги кожної складової залікового кредиту:

Заліковий модуль 1	Заліковий модуль 2 (РКР)	Заліковий модуль 3 (КППЗ)	Разом
30%	40%	30%	100 %
Виконання лабораторних робіт (6 робіт по 10 балів - 60 балів) Написання модульної роботи – 40 балів	Виконання лабораторних робіт (6 робіт по 10 балів - 60 балів) Написання ректорської контрольної роботи – 40 балів	Виконання завдань під час тренінгу (20 балів) Написання та захист КППЗ (80 балів)	100

Шкала оцінювання:

За шкалою ЗУНУ	За національною шкалою	За шкалою ECTS
90–100	відмінно	A (відмінно)
85–89	добре	B (дуже добре)
75-84		C (добре)
65-74	задовільно	D (задовільно)
60-64		E (достатньо)
35-59	незадовільно	FX (незадовільно з можливістю повторного складання)
1-34		F (незадовільно з обов'язковим повторним курсом)

1. Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

Найменування	Номер теми
--------------	------------

№		
1.	Мультимедійний проектор	1-8
2.	Проекційний екран	1-8
3.	Комунікаційне програмне забезпечення (Internet Explorer, Google Chrome, Firefox)	1-8
4.	Операційна система Windows, наявність доступу до мережі Internet	1-8
5.	Персональні комп'ютери	1-8
6.	Комунікаційне програмне забезпечення (Zoom) для проведення занять у режимі онлайн (за необхідності)	1-8
7.	Комунікаційна навчальна платформа (Moodle) для організації дистанційного навчання (за необхідності)	1-8
8.	Базове програмне забезпечення Microsoft Office	1-8
9.	Спеціалізоване програмне забезпечення For DevOps.	1-8

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

1. Doak J. Go for DevOps / J. Doak, D. Justice. – MUMBAI: Packt Publishing, 2022. – 634 p.
2. Arundel J. Cloud Native DevOps with Kubernetes / J. Arundel, J. Dominguez. – Boston: O'Reilly, 2020. – 384 p.
3. The Site Reliability Workbook / B.Beyer, N. Murphy, D. Rensin, K. Kawahara. – Boston: O'Reilly, 2018. – 508 p.
4. Site Reliability Engineer / B.Beyer, N. Murphy, C. Jones, J. Petoff. – Boston: O'Reilly, 2019. – 592 p.
5. Winters T. Software Engineering at Google / T. Winters, T. Manshreck, H. Wright. – Boston: O'Reilly, 2020. – 617 p.
6. Behr K. The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win / K. Behr, G. Kim, G. Spafford. – London: IT Revolution Press, 2020. – 345 p.
7. Brown C. Chatgpt prompts mastering / Christian Brown. – United States of America: Amazon.com, 2023. – 85 p.
8. Douglass B. Agile Model-Based Systems Engineering Cookbook / Bruce Powel Douglass. – MUMBAI: Packt Publishing, 2023. – 601 p.
9. Cowell C. Automating DevOps with GitLab CI/CD Pipelines / C. Cowell, N. Lotz, C. Timberlake. – BIRMINGHAM—MUMBAI: Packt Publishing, 2023. – 348 p.
10. Paul J. Distributed Serverless Architectures on AWS / Jithin Jude Paul. – Ernakulam, Kerala, India: Apress, 2023. – 154 p.

