

## СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «ТЕХНОЛОГІЇ DEVOPS»



Ступінь освіти	магістр
Освітня програма	Інформаційні системи та технології
Тривалість викладання	1, 2 чверть
Заняття:	Осінній семестр
лекції:	1 година
лабораторні заняття:	2 години
Мова викладання	українська

Сторінка курсу в СДО НТУ «ДП»: <https://do.nmu.org.ua/course/view.php?id=3286>

Кафедра, що викладає Інформаційних технологій та комп'ютерної інженерії



**Викладач:**

**Гаркуша Ігор Миколайович**  
Доцент, к.т.н., доцент кафедри

**Персональна сторінка**

[http://it.nmu.org.ua/ua/HR\\_staff/prepods/garkusha.php](http://it.nmu.org.ua/ua/HR_staff/prepods/garkusha.php)

**Е-mail:**

[garkusha.i.m@nmu.one](mailto:garkusha.i.m@nmu.one)

### 1. Анотація до курсу

**DevOps** – це культурний рух, що змінює ставлення людей до роботи і до її результатів. Завдяки впровадженню DevOps в організації формуються інтенціональні процеси, що прискорюють ефективність бізнесу. Це сприяє швидкій появі результату від соціальних і технічних нововведень. Завдяки новим способам мислення і роботи окремі співробітники та організації можуть розвивати і підтримувати стійкі робочі практики. DevOps це культурний субстрат, що прискорює формування емпатії між колегами і полегшує обмін досвідом. В результаті закладається міцний фундамент, що забезпечує ефективне докладання зусиль в процесі роботи з боку окремих співробітників і команд.

**DevOps** – це процес безперервного вдосконалення програмних продуктів за допомогою прискорення циклів релізів, глобальної автоматизації інтеграційних і розгорточних конвеєрів та тісної взаємодії між командами. Метою DevOps є скорочення часу і вартості втілення ідеї в продукт, яким користуються клієнти. DevOps застосовує багато автоматизованих процесів для прискорення розробки та розгортання.

*Дисципліна “Технології DevOps”* є комплексом знань та практичних навичок, які направлені на засвоєння певних DevOps-практик та найвідоміших інструментів для прискорення розробки та розгортання в проектах.

## 2. Мета та завдання курсу

**Мета дисципліни** – формування компетентностей щодо процесу швидкого розгортання, супроводу, моніторингу та загальної автоматизації певних задач в хмарних середовищах на Linux-сумісних платформах у тому числі на базі мікросервісів.

### Завдання курсу:

- ознайомити здобувачів вищої освіти з певними DevOps-практиками;
- закріпити знання та навички програмування мовами Bash, Python та Go;
- вивчити особливості використання інструменту Git;
- навчити здобувачів вищої освіти будувати docker-контейнери та простіші мікросервіси.

## 3. Результати навчання

Основні результати навчання:

- управляти інформаційними ресурсами та системами, організовувати та підтримувати виконання комплексу заходів з впровадження інформаційних систем та сервісів;
- обґрунтовувати вибір програмних продуктів, їх взаємодію та потенційний вплив на вирішення організаційних проблем, здійснювати їх впровадження та використання з дотриманням професійних і етичних стандартів;
- використовувати мови програмування, опису інформаційних ресурсів, специфікацій, інструментальні засоби, крос-платформні бібліотеки, фреймворки та технології під час проектування та створення інформаційних систем, продуктів і сервісів інформаційних технологій.

## 4. Структура курсу

### ЛЕКЦІЇ

#### 1. Bash та Python для DevOps

- 1.1. Коротка історія Bash та основні можливості
- 1.2. Спеціальні файли
- 1.3. Використання лапок. Формати команд
- 1.4. Функції. Змінні та підставка змінних. Використання declare. Масиви в Bash
- 1.5. Операції. Використання for та if
- 1.6. Основні елементи мови Python та особливості використання
- 1.7. Оператори мови. Використання циклічних процесів та відповідних структур даних
- 1.8. Використання в скриптах

#### 2. Швидке введення в мову Go

- 2.1. Коротка історія мови. Підготовка коду, компіляція та запуск
- 2.2. Використання пакетів

2.3. Оператори мови Go. Функції

2.4. Реалізація простого HTTP-серверу

### 3. Інструмент Git

3.1. Коротка історія Git. Призначення

3.2. Встановлення та налаштування

3.3. Створення нового репозиторія та клонування існуючого

3.4. Команди status, add, commit, rm, mv, log, tag, show, branch, checkout, merge, push, pull

### 4. Інструмент Docker

4.1. Призначення контейнерів. Коротка історія Docker

4.2. Суть мікросервісів. Архітектура Docker

4.3. Основні команди

4.4. Використання Dockerfile

4.5. Створення образів. Взаємозв'язок з контейнером

4.6. Управління даними

### 5. Технологія Kubernetes

5.1. Історія створення та призначення

5.2. Завантаження/перезавантаження контейнерів. Pod-оболонки

5.3. Планувальник Kubernetes

5.4. Адреси сервісів

5.5. Робота з kubectl

5.6. Засоби моніторингу

### 6. Jenkins, Ansible, Zabbix

6.1. Використання Jenkins, Ansible, Zabbix

## ЛАБОРАТОРНІ ЗАНЯТТЯ

**TDO-1** – Використання Bash та Python при автоматизації задач.

**TDO-2** – Використання Go при автоматизації задач.

**TDO-3** – Робота з Git.

**TDO-4** – Створення та обслуговування Docker-контейнерів.

### 5. Технічне обладнання та/або програмне забезпечення

№ роботи (шифр)	Назва роботи	Інструменти, обладнання та програмне забезпечення, що застосовуються при проведенні роботи
TDO-1	Використання Bash та Python при автоматизації задач	Персональний комп'ютер Платформа Ubuntu (через мережеве з'єднання або через середовище Oracle VM VirtualBox) Редактор nano, mcedit або geany

№ роботи (шифр)	Назва роботи	Інструменти, обладнання та програмне забезпечення, що застосовуються при проведенні роботи
		Bash 4.4.x Python 3.x PyCharm MS Office або LibreOffice
TDO-2	Використання Go при автоматизації задач	Персональний комп'ютер Платформа Ubuntu (через мережеве з'єднання або через середовище Oracle VM VirtualBox) Редактор nano, mcedit або geany GoLang 1.10.x LiteIDE або сумісний MS Office або LibreOffice
TDO-3	Робота з Git	Персональний комп'ютер Платформа Ubuntu (через мережеве з'єднання або через середовище Oracle VM VirtualBox) Розгорнутий GitLab-сервер Редактор nano, mcedit або geany Python 3.x PyCharm MS Office або LibreOffice
TDO-4	Створення та обслуговування Docker-контейнерів	Персональний комп'ютер Платформа Ubuntu (через мережеве з'єднання або через середовище Oracle VM VirtualBox) Розгорнутий GitLab-сервер Редактор nano, mcedit або geany Python 3.x PyCharm Docker MS Office або LibreOffice

## 6. Система оцінювання та вимоги

**6.1. Навчальні досягнення здобувачів вищої освіти за результатами вивчення курсу оцінюватимуться за шкалою, що наведена нижче:**

Рейтингова шкала	Інституційна шкала
90 – 100	відмінно
75-89	добре
60-74	задовільно
0-59	незадовільно

**6.2.** Здобувачі вищої освіти можуть отримати **підсумкову оцінку** з навчальної дисципліни на підставі поточного оцінювання знань за умови, якщо набрана кількість балів з поточного тестування та самостійної роботи складатиме не менше 60 балів.

Максимальне оцінювання:

Теоретична частина	Лабораторна частина		Бонус	Разом
	При своєчасному складанні	При несвоєчасному складанні		
65	30	20	5	<b>100</b>

Лабораторні роботи приймаються за контрольними запитаннями до кожної з роботи.

Теоретична частина оцінюється за результатами здачі білету диференційного заліку, який містить 2 питання.

### **6.3. Критерії оцінювання підсумкової роботи**

Робота повинна містити розгорнуті відповіді на два питання білету. Якщо робота виконується у дистанційному режимі, то видача номеру білета проходить через систему MS Teams у зазначеній викладачем групі спілкування. В такому режимі виконана робота пишеться вручну, фотографується та відсилається не електронну пошту викладача у впродовж встановленого викладачем часу. За виконану роботу нараховуються бали:

**65 балів** – дана розгорнута відповідь на два питання.

**50 балів** – дана розгорнута відповідь на одне питання, але є помилки при розгляді іншого питання, або є несуттєві помилки у відповідях на два питання.

**30 балів** – два повна відповідь на одне питання або на два питання зі значними помилками.

**20 балів** – відповідь на одне питання із значними помилками.

**0 балів** – відповіді на питання відсутні або повністю невірні, або робота здана несвоєчасно.

### **6.4. Критерії оцінювання лабораторної роботи**

З кожної лабораторної роботи здобувач вищої освіти отримує запитання з переліку контрольних запитань до роботи. Кількість вірних відповідей визначають кількість отриманих балів.

## **7. Політика курсу**

### **7.1. Політика щодо академічної доброчесності**

Академічна доброчесність здобувачів вищої освіти є важливою умовою для опанування результатами навчання за дисципліною і отримання задовільної оцінки з поточного та підсумкового контролів. Академічна доброчесність базується на засудженні практик списування (виконання письмових робіт із залученням зовнішніх джерел інформації, крім дозволених для використання), плагіату (відтворення

опублікованих текстів інших авторів без зазначення авторства), фабрикації (вигадування даних чи фактів, що використовуються в освітньому процесі). Політика щодо академічної доброчесності регламентується положенням "Положення про систему запобігання та виявлення плагіату у Національному технічному університеті "Дніпровська політехніка". [http://www.nmu.org.ua/ua/content/activity/us\\_documents/System\\_of\\_prevention\\_and\\_detection\\_of\\_plagiarism.pdf](http://www.nmu.org.ua/ua/content/activity/us_documents/System_of_prevention_and_detection_of_plagiarism.pdf).

У разі порушення здобувачем вищої освіти академічної доброчесності (списування, плагіат, фабрикація), робота оцінюється незадовільно та має бути виконана повторно. При цьому викладач залишає за собою право змінити тему завдання.

## **7.2. Комунікаційна політика**

Здобувачі вищої освіти повинні мати активовану університетську пошту.

Усі письмові запитання до викладачів стосовно курсу мають надсилатися на університетську електронну пошту.

## **7.3. Політика щодо перескладання**

Роботи, які здаються із порушенням термінів без поважних причин оцінюються на нижчу оцінку. Перескладання відбувається із дозволу деканату за наявності поважних причин (наприклад, лікарняний).

## **7.4 Політика щодо оскарження оцінювання**

Якщо здобувач вищої освіти не згоден з оцінюванням його знань він може опротестувати виставлену викладачем оцінку у встановленому порядку.

## **7.5. Відвідування занять**

Для здобувачів вищої освіти денної форми відвідування занять є обов'язковим. Поважними причинами для неявки на заняття є хвороба, участь в університетських заходах, академічна мобільність, які необхідно підтверджувати документами. Про відсутність на занятті та причини відсутності здобувач вищої освіти має повідомити викладача або особисто, або через старосту.

За об'єктивних причин (наприклад, міжнародна мобільність) навчання може відбутись в он-лайн формі за погодженням з керівником курсу.

## **7.6. Бонуси**

Наприкінці вивчення курсу та перед початком сесії здобувача вищої освіти буде запропоновано анонімно заповнити електронні анкети (Microsoft Forms Office 365), які буде розіслано на ваші університетські поштові скриньки. Заповнення анкет є важливою складовою вашої навчальної активності, що дозволить оцінити дієвість застосованих методів викладання та врахувати ваші пропозиції стосовно покращення змісту навчальної дисципліни «Технології DevOps». За участь у анкетуванні здобувач вищої освіти отримує **5 балів**.

## 8 Рекомендовані джерела інформації

### Базові

- 1 Mikael Krief. Learning DevOps. – Packt Publishing, 2019. – 489 p. – ISBN 978-1-83864-273-0.
- 2 Scott Chacon, Ben Straub. Pro Git. – Apress, 2014. – 441 p. Online ISBN 978-1-4842-0076-6.
- 3 Чакон С., Штрауб Б. Git для професіонального програміста. – СПб.: Питер, 2016. – 496 с.: ил. – (Серия «Библиотека программиста»). ISBN 978-5-496-01763-3.
- 4 Моуэт Э. Использование Docker / пер. с англ. А.В. Снастина; науч. ред. А.А.Маркелов. – М.: ДМК Пресс, 2017. – 354 с.: ил. ISBN 978-5-97060-426-7.
- 5 Арундел Джон, Домингус Джастин. Kubernetes для DevOps: развертывание, запуск и масштабирование в облаке. – СПб.: Питер, 2020. – 384 с.: ил. – (Серия «Бестселлеры O'Reilly»). ISBN 978-5-4461-1602-7.
- 6 Далле Вакке А. Zabbix. Практическое руководство / пер. сангл. А.Н. Киселева. – М.: ДМК Пресс, 2017. – 356 с.: ил. ISBN 978-5-97060-462-5.
- 7 Brent Laster. Jenkins 2. Приступаем к работе. / пер. с англ. Д. А. Беликова. – М.: ДМК Пресс, 2019. – 652 с.: ил. ISBN 978-5-97060-711-4.
- 8 Хохштейн Л., Мозер Р. Запускаем Ansible / пер. с англ. Е.В. Филонова, А.Н. Киселева. – М.: ДМК Пресс, 2018. – 382 с.: ил. ISBN 978-5-97060-513-4.
- 9 Джульен Вехен Безопасный DevOps. Эффективная эксплуатация систем. – СПб.: Питер, 2020. – 432 с.: ил. – (Серия «Для профессионалов»). ISBN 978-5-4461-1336-1.
- 10 Форсгрэн Н., Хамбл Д., Ким Д. Ускоряйся! Наука DevOps: Как создавать и масштабировать высокопроизводительные цифровые организации / Николь Форсгрэн, Джек Хамбл, Джин Ким; Пер. с англ. А. Техненко. – М.: Интеллектуальная Литература, 2020. – 180 с. ISBN 978-5-9072-7433-4.
- 11 Роббинс, Арнольд. Bash. Карманный справочник системного администратора, 2-е изд.: Пер. с англ. – СПб.: ООО "Альфа-книга", 2017. – 152 с.: ил. – Парал. тит. англ. ISBN 978-5-9909445-4-1.
- 12 Тейлор Дейв, Перри Брендон. Сценарии командной оболочки. Linux, OS X и Unix. 2-е изд. – СПб.: Питер, 2017. – 448 с.: ил. – (Серия «Для профессионалов»). ISBN 978-5-496-03029-8.

- 13 Свейгарт, Эл. Автоматизация рутинных задач с помощью Python: практическое руководство для начинающих.: Пер. с англ. – М.: ООО “И.Д. Вильямс”, 2017 год. – 592 с.: ил. – Парал. тит. англ. ISBN 978-5-8459-2090-4.
- 14 Марк Саммерфильд. Программирование на Go. Разработка приложений XXI века: пер.с англ.: Киселёв А.Н. – М.: ДМК Пресс, 2013. – 580 с.: ил. ISBN 978-5-94074-854-0.
- 15 Newmarch, Jan. Network Programming with Go: Essential Skills for Using and Securing Networks. – Apress, 2017. eBook ISBN 978-1-4842-2692-6.
- 16 Ричардсон Крис. Микросервисы. Паттерны разработки и рефакторинга. – СПб.: Питер, 2019. – 544 с.: ил. – (Серия «Библиотека программиста»). ISBN 978-5-4461-0996-8.

### Додаткові

- 17 Джин Ким, Кевин Бер, Джордж Спаффорд. Проект «Феникс». Роман о том, как DevOps меняет бизнес к лучшему. Пер. с англ. Агеев В. – ООО «Издательство «Эксмо», 2015. – 384 с.
- 18 Ким, Джин. Руководство по DevOps. Как добиться гибкости, надежности и безопасности мирового уровня в технологических компаниях / Джин Ким, Патрик Дебуа, Джон Уиллис, Джек Хамбл; пер. с англ. И. Лейко и И. Васильева; [науч. ред. Н. Корытко]. – М.: Манн, Иванов и Фербер, 2018. – 512 с. ISBN 978-5-00100-750-0.
- 19 Дженнифер Дэвис, Кэтрин Дэниелс. Философия DevOps. Искусство управления ИТ. — СПб.: Питер, 2017. – 416 с. ISBN 978-5-496-02555-3.
- 20 Лимончелли Т., Хоган К., Чейлап С. Системное и сетевое администрирование. Практическое руководство, 2-е издание. – Пер. с англ. – СПб: Символ-Плюс, 2009. – 944 с., ил. ISBN: 978-5-93286-130-1.
- 21 Айвалиотис Д. Администрирование сервера NGINX. – М.: ДМК Пресс, 2013. – 288 с.: ил. ISBN 978-5-94074-961-5.
- 22 Kozyra N. Mastering Go Web Services. – Packt Publishing, 2015. – 264 p. – ISBN-10: 178398130X, ISBN-13: 978-1-78398-130-4.
- 23 Команды LINUX «от А до Z». Обзор с примерами. – Академия Cisco, LPI, SEDICOMM, 2017. – 50 с.