



DevOps

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	122 Комп'ютерні науки
Освітня програма	Цифрові технології в енергетиці
Статус дисципліни	Вибіркова
Форма навчання	Очна (денна)
Рік підготовки, семестр	3 курс, осінній семестр
Обсяг дисципліни	4 кредити ЕКТС / 120 год.: лекції – 36 год., лабораторні роботи – 18 год., самостійна робота – 66 год.
Семестровий контроль / контрольні заходи	м.к.р., залік
Розклад занять	http://rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: доцент, к.т.н., доц. Лабжинський Володимир Анатолійович, email: labzhynskiy.volodymyr@lll.kpi.ua Лабораторні роботи: доцент, к.т.н., доц. Лабжинський Володимир Анатолійович, email: labzhynskiy.volodymyr@lll.kpi.ua
Розміщення курсу	Лекції: https://do.ipk.kpi.ua/course/view.php?id=298 (Увійдіть як гість) Лабораторні роботи: https://do.ipk.kpi.ua/course/view.php?id=259 (Увійдіть як гість)

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Метою вивчення навчальної дисципліни є формування у студентів таких загальних і фахових компетентностей:

- здатності до абстрактного мислення, аналізу та синтезу (ЗК1);
- здатності застосовувати знання у практичних ситуаціях (ЗК2);
- знання та розуміння предметної області та розуміння професійної діяльності (ЗК3);
- здатності генерувати нові ідеї (креативність) (ЗК8);
- здатності працювати в команді (ЗК9);
- здатності діяти на основі етичних міркувань (ЗК13);
- здатності реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, знань і сховища даних, виконувати розподілену обробку великих наборів даних на кластерах стандартних серверів для забезпечення обчислювальних потреб користувачів, у тому числі на хмарних сервісах (ФК9).

Згідно з вимогами освітньо-професійної програми студенти після засвоєння навчальної дисципліни мають продемонструвати такі результати навчання:

ЗНАННЯ:

- життєвий цикл програмного забезпечення та сучасні методології його розроблення;
- парадигми програмування, сучасних мов програмування;

- принципи побудови, склад, структуру та функції сучасних UNIX-подібних операційних систем;
- принципи та протоколи взаємодії основних мережних служб мережі Інтернет;
- основні інструментальні програмні засоби DevOps-інженера.

вміння:

- застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній галузі комп'ютерних наук (ПР1);
- використовувати довідкову літературу, технічну документацію;
- встановлювати, налаштовувати та обслуговувати системне та прикладне програмне забезпечення в хмарних середовищах;
- правильно застосовувати основні інструментальні програмні засоби DevOps-інженера.

2. Пререквізити та постреквізити дисципліни

Вивчення дисципліни спирається на знання, отримані за програмою попередніх років навчання за спеціальністю 122 «Комп'ютерні науки».

В структурно-логічній схемі навчання дисципліна "DevOps" розташована в 7 семестрі, тобто тоді, коли студенти вже прослухали курси "Алгоритмізація та програмування", "Комп'ютерна схемотехніка та архітектура", засвоїли принципи побудови алгоритмів, ознайомилися з архітектурою сучасних комп'ютерів та набули певного досвіду в галузі розроблення програмного забезпечення. З іншого боку, матеріал цього кредитного модуля може бути використаний під час вивчення таких курсів, як "Системи баз даних", "Технології паралельних обчислень в енергетичних комплексах", що подаються в наступних семестрах.

3. Зміст навчальної дисципліни

В дисципліні вивчаються такі теми:

Розділ 1. Загальні відомості про методологію DevOps

Тема 1.1. DevOps – сучасний підхід до розроблення та експлуатації програмного забезпечення

Розділ 2. Взаємодія та адміністрування ОС Linux

Тема 2.1. Взаємодія ОС Linux з користувачем

Тема 2.2. Адміністрування ОС Linux

Розділ 3. Хмарна інфраструктура та інструменти DevOps-інженера

Тема 2.1. AWS – приклад хмарної інфраструктури

Тема 2.2. Інструменти DevOps-інженера

4. Навчальні матеріали та ресурси

Базова література

1. Kim G., J. Humble, P. Debois, J. Willis, and N. Forsgren. 2021. *The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations*. 2nd ed. Portland, OR: IT Revolution Press.
2. Kernighan, B. W., and R. Pike. 1984. *The UNIX Programming Environment*. Englewood Cliffs, NJ: Prentice-Hall.
3. Nemeth E., G. Snyder, T. Hein, B. Whaley, and D. Mackin. 2017. *UNIX and Linux System Administration Handbook*. 5th ed. Boston, MA: Addison-Wesley Professional.
4. Wittig A., and M. Wittig. 2023. *Amazon Web Services in Action: An in-depth guide to AWS*. 3rd ed. Shelter Island, NY: Manning Publications.
5. Chacon S., and B. Straub. 2014. *Pro Git*. 2nd ed. New York, NY: Apress.

Додаткова література

1. Mouat A. 2016. *Using Docker: Developing and Deploying Software with Containers*. Sebastopol, CA: O'Reilly Media.
2. Domingus D., and J. Arundel. 2022. *Cloud Native DevOps with Kubernetes: Building, Deploying, and Scaling Modern Applications in the Cloud*. 2nd ed. Sebastopol, CA: O'Reilly Media.
3. Morris K. 2021. *Infrastructure as Code: Dynamic Systems for the Cloud Age*. 2nd ed. Sebastopol, CA: O'Reilly Media.
4. Freeman J., and J. Keating. 2021. *Mastering Ansible: Automate configuration management and overcome deployment challenges with Ansible*. 4th ed. Birmingham: Packt Publishing.
5. Tanenbaum A. S., and H. Bos. 2023. *Modern Operating Systems*. 5th ed. Upper Saddle River, NJ: Prentice-Hall PTR.

Навчальний контент

5. Методика опанування навчальної дисципліни

Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань	Кільк. ауд.год.
Розділ 1. Загальні відомості про методологію DevOps		
Тема 1.1. DevOps – сучасний підхід до розроблення та експлуатації програмного забезпечення		
1	Життєвий цикл програмного забезпечення. Моделі життєвого циклу. Порівняльний аналіз різних методологій розроблення програмного забезпечення.	2
2	Вступ до DevOps. Загальний огляд методології DevOps. Огляд основних інструментів і практик.	2
Розділ 2. Взаємодія та адміністрування ОС Linux		
Тема 2.1. Взаємодія ОС Linux з користувачем		
3	Оболонка – основний інструмент взаємодії ОС Linux з користувачем. Оболонка (shell) – інтерактивний оброблювач команд користувача. Огляд можливостей оболонки. Написання сценаріїв для оболонки Bash, змінні оточення, розгалуження, цикли.	2
4	Пошук даних у файлах. Регулярні вирази. Програми-фільтри: grep, sed, awk.	2
Тема 2.2. Адміністрування ОС Linux		
5	Процедура початкового завантаження ОС Linux. Стадії завантаження ОС Linux, процес init, огляд функцій ядра, програми ps, top, htop, atop. Демон Systemd, робота з юніт-сервісами та таймерами, аналіз та пошук помилок за допомогою journalctl.	2
6	Налаштування мережі в ОС Linux. Налаштування мережних інтерфейсів, netplan, ufw, SSH, робота з ключами, scp, rsync, NTP. Створення та конфігурування середовища з допомогою Vagrant.	2
7	Налаштування DNS-сервера в ОС Linux. Первинні та вторинні сервери. Зони відповідальності. Ресурсні записи.	2
8	Налаштування Web-сервера в ОС Linux. Apache та Nginx. Протоколи http та https. Розуміння архітектури Apache та Nginx. Стек LAMP (Linux, Apache, MySQL, PHP). Балансування навантаження за допомогою Nginx. Списки доступу та журналювання запитів.	2
Розділ 3. Хмарна інфраструктура та інструменти DevOps-інженера		
Тема 3.1. AWS – приклад хмарної інфраструктури		

9	Створення хмарної інфраструктури в середовищі AWS. Знайомство з хмарною інфраструктурою та основними сервісами. Робота з сервісами IAM, VPC, EC2, S3, Route53. Створення ресурсів в AWS за допомогою Terraform. Моніторинг CloudWatch. Керування сервісами та мінімізація витрат.	2
10	Інструменти для створення застосунків у середовищі AWS. Огляд можливостей засобів створення ПЗ: CodePipeline, CodeBuild, CodeDeploy, CodeStar.	2
Тема 3.2. Інструменти DevOps-інженера		
11	Git та GitLab. Основи роботи з git. Push, fetch, pull. Git flow. Cloning repo, branches, merging branches, pull requests. Робота з Gitlab.	2
12	Керування конфігураціями з допомогою Ansible. Автоматизація процесів налаштування та розгортання програмного забезпечення. Templates, handlers, dynamic inventory, vault, tags. Ролі, цикли, умови, фільтри.	2
13	Docker. Вступ до контейнеризації. Пристрій Docker. Dockerfile, docker registry, docker-compose. Безпека контейнерів.	2
14	Continuous Integration & Continuous Delivery. Вступ до CI/CD. Пристрій Gitlab CI. Безперервне постачання ПЗ. Побудова конвеєрів (pipeline) в GitLab. Інструмент Jenkins.	2
15	Автоматизація розгортання та масштабування контейнерних застосунків з допомогою Kubernetes. Вступ до Kubernetes, архітектура та основні компоненти. Розгортання кластера Kubernetes. Взаємодія з кластером за допомогою API.	2
16	Автоматизація керування контейнерними застосунками з допомогою Kubernetes. Робота з kubectl. Знайомство та операції з просторами імен. Створення та керування Pods, Deployments, ReplicaSets. Створення та управління Services, Ingress, Controllers. Вступ до Helm Charts. Використання Secrets для роботи з конфіденційною інформацією.	2
17	Моніторинг та журналювання. Моніторинг сервісів з допомогою Prometheus. Збирання та аналіз даних журналів з допомогою Elastic Stack (ELK). Розсилання попереджень з допомогою AlertManager. Візуалізація даних з допомогою Grafana.	2
18	Моніторинг обладнання та сервісів з допомогою Zabbix. Схема "менеджер-агент". Опитування пристроїв за протоколом SNMP.	2

Лабораторні роботи

№ з/п	Тема лабораторної роботи	Кільк. ауд.год.
Розділ 2. Взаємодія та адміністрування ОС Linux		
Тема 2.1. Взаємодія ОС Linux з користувачем		
1	Робота з оболонкою. Встановлення змінних оточення. Створення власного сценарію реєстрації	2
Тема 2.2. Адміністрування ОС Linux		
2	Налаштування DNS-сервера	2
3	Налаштування Web-серверів Apache та Nginx	2
Розділ 3. Хмарна інфраструктура та інструменти DevOps-інженера		
Тема 3.1. AWS – приклад хмарної інфраструктури		
4	Розроблення застосунків у середовищі AWS	2

Тема 3.2. Інструменти DevOps-інженера		
5	Робота в середовищах Git та GitLab	2
6	Створення контейнерів для застосунків з допомогою утиліти Docker	2
7	Робота в середовищі Kubernetes	2
8	Моніторинг працездатності мережних сервісів та пристроїв	2

Контрольні роботи

В п'ятому семестрі проводять одну модульну контрольну роботу. Для її проведення виділяють 2 навчальні години рахунок часу, передбаченого для виконання лабораторних робіт.

Метою модульної контрольної роботи є перевірка засвоєння студентами теоретичних відомостей з основних принципів налаштування класичної багатозадачної ОС Linux, а також перевірка знань та практичних навичок студентів, необхідних DevOps-інженеру в повсякденній діяльності.

6. Самостійна робота

Самостійна робота студента (66 годин) передбачає підготовку до аудиторних занять та контрольних заходів, засвоєння базової та ознайомлення з додатковою літературою.

Розподіл годин СРС: підготовка до лекції – 2 години; підготовка до лабораторної роботи – 3 години; підготовка до МКР – 2 години; підготовка до заліку – 4 години.

Політика та контроль

7. Політика навчальної дисципліни

- відвідування лекцій, а також відсутність на них, не оцінюється. Відвідування лабораторних і практичних занять є обов'язковою складовою вивчення матеріалу;
- під час захисту лабораторних робіт студент має продемонструвати розроблений програмний код та результати його виконання на тестах, як заздалегідь підготованих, так і запропонованих викладачем. У випадку дистанційної форми навчання захист відбувається на відповідній конференції шляхом демонстрації екрана;
- політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>;
- норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Розмір шкали рейтингових оцінок – 100 балів. Оцінка з дисципліни виставляється за рейтинговою системою з подальшим переведенням до традиційної 4-бальної.

Загальна рейтингова оцінка студента після завершення семестру складається з балів, отриманих за:

- виконання та захист лабораторних робіт;
- написання однієї модульної контрольної роботи.

Лабораторні роботи:

- кількість лабораторних робіт згідно з робочою програмою – 8;
- максимальна кількість балів за виконання лабораторних робіт – $r_{лр}=76$;
- внесок окремих лабораторних робіт до семестрового рейтингу студента подано у таблиці.

Номер лабораторної роботи	Внесок до семестрового рейтингу, балів
1.	5

2.	8
3.	8
4.	11
5.	11
6.	11
7.	11
8.	11
Всього	76

Критерії оцінювання виконання та захисту лабораторних робіт:

- якщо студент виконав роботу, але не відповів на контрольні запитання до неї, то за роботу нараховується 50% від максимальної кількості балів;
- якщо робота виконана невчасно, то знімається 10–30% від максимальної кількості балів (кількість процентів залежить від терміну запізнення);
- якщо студент виконав роботу не самостійно та не розбирається в коді програми, то бали за роботу не нараховуються;
- якщо в програмі не витримані основні правила створення програмних продуктів (модульність, дружній інтерфейс, наявність коментарів тощо) знімається по 5% від максимальної кількості балів за кожний з цих недоліків.

Модульна контрольна робота (МКР):

- тема МКР “Програмні засоби DevOps-інженера”;
- МКР містить два теоретичні та два практичні завдання;
- максимальна кількість балів за написання МКР $r_{МКР}=24$, в тому числі за теоретичне завдання – 8 балів, за практичне завдання – 4 бали;
- термін проведення МКР – 17-й тиждень семестру.

Критерії оцінювання написання МКР:

- якщо на питання подані повні та чітко аргументовані відповіді, контрольна робота виконана охайно, з дотриманням основних правил оформлення, то виставляється 90-100% від максимальної кількості балів;
- якщо методика виконання запропонованого завдання вірна, але допущені неprincipові помилки у теоретичному описі або чисельних розрахунках, то виставляється 75-90% від максимальної кількості балів;
- від 50% до 75% від максимальної кількості балів нараховується, якщо методика виконання завдання в основному вірна, але допущені деякі з наступних помилок: помилки у поданні вихідних даних, не обґрунтовані теоретичні рішення, є принципові помилки у чисельних розрахунках;
- менше 50% від максимальної кількості балів нараховується, якщо завдання не виконане або студент припустився грубих помилок при його виконанні.

Умови позитивної проміжної атестації

Поточні індивідуальні рейтинги студентів періодично доводяться викладачем до студентів безпосередньо або через старосту групи.

Поточна атестація студентів базується на поточній рейтинговій оцінці. Умовою позитивної атестації є значення поточного рейтингу студента не менше 50% від максимально можливого на час атестації.

Для отримання позитивної оцінки з першої проміжної атестації рейтинг студента повинен бути не менше, ніж 16 балів (за умови, що за 8 тижнів семестру згідно з календарним планом контрольних заходів студент може отримати максимум 32 бали).

Для отримання позитивної оцінки з другої проміжної атестації рейтинг студента повинен бути не менше, ніж 26 балів (за умови, що за 14 тижнів семестру згідно з календарним планом контрольних заходів студент може отримати максимум 52 бали).

Семестрова атестація (залік)

Залік проводиться на останній лабораторній роботі. Умови допуску студента до заліку:

- виконання всіх обов'язкових видів робіт, передбачених робочою програмою;
- студент отримує залік, якщо сума балів, набраних протягом семестру, не менша 60.

Розрахунок шкали (R) рейтингу:

Сума вагових балів контрольних заходів протягом семестру складає:

$$R = r_{\text{ЛР}} + r_{\text{МКР}} = 76 + 24 = 100 \text{ балів}$$

Таким чином, рейтингова шкала з кредитного модуля складає 100 балів.

Необхідною умовою допуску до складання заліку є стартовий рейтинг r_c , що дорівнює $0,4 \cdot 100 = 40$ балів.

Студенти, які набрали протягом семестру рейтинг з кредитного модуля $0,4R \leq r_c < 0,6R$, зобов'язані виконувати залікову контрольну роботу (пройти співбесіду).

Відповідність між рейтинговими та традиційними оцінками наведена в таблиці.

Кількість балів	Оцінка
95–100	Відмінно
85–94	Дуже добре
75–84	Добре
65–74	Задовільно
60–64	Достатньо
40–59	Незадовільно
0–39	Не допущений

9. Додаткова інформація з дисципліни

Для кращого засвоєння матеріалу рекомендується проводити лабораторні роботи на комп'ютерах, що підключені до мережі Інтернет. Завдання на розроблення алгоритму роботи паралельного сервера слід сформулювати заздалегідь з урахуванням бажання студентів та їх пропозицій.

Робочу програму навчальної дисципліни (силабус):

Складено доцентом, к.т.н., доц. Лабжинським Володимиром Анатолійовичем

Ухвалено кафедрою ЦТЕ (протокол № 20 від 10.05.2023)

Погоджено Методичною комісією НН ІАТЕ КПІ ім. Ігоря Сікорського (протокол № 9 від 26.05.2023)