Міністерство освіти і науки України
Національний аерокосмічний університет
«Харківський авіаційний інститут»

Кафедра інформаційно-комунікаційних технологій
ім. О.О. Зеленського (№ 504)

ЗАТВЕРДЖУЮ

Гарант освітньої програми



_____ Олег ЄРЕМЕЄВ
(підпис) (ім'я та прізвище)

«31» серпня 2026 р.

СИЛАБУС
ОБОВ'ЯЗКОВОЇ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Адміністрування інформаційних систем»
(назва навчальної дисципліни)

Галузь знань: F «Інформаційні технології»
(шифр і найменування галузі знань)
Спеціальність: F6 «Інформаційні системи і технології»

Освітня програма: «Штучний інтелект та інформаційні системи»

Форма навчання: денна

Рівень вищої освіти: перший (бакалаврський)

Силабус введено в дію з 01.09.2026

Харків 2026 рік

Розробник: Зряхов М.С., доцент, канд. техніч. наук, доцент _____
(прізвище та ініціали, посада, науковий ступінь і вчене звання) (підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри №504 кафедра інформаційно-комунікаційних технологій імені О.О. Зеленського
(назва кафедри)

Протокол № 1 від 31 серпня 2026 р.

Завідувач кафедри д.т.н., професор _____ Володимир ЛУКІН
(науковий ступінь і вчене звання) (ім'я та прізвище)

Погоджено з представником здобувачів освіти:

здобувачка гр. 528 _____ Ангеліна ПОГРЕБНЯК
(підпис) (ім'я та ПРІЗВИЩЕ)

1. Загальна інформація про викладача



ПІБ: Зряхов Михайло Сергійович

Посада: доцент

Науковий ступінь: канд. техніч. наук

Вчене звання: доцент

Перелік дисциплін, які викладає:
«Хмарні технології»
«Технології неперервної інтеграції і розгортання інформаційних систем»
«UNIX-подібні операційні системи»
«Адміністрування інформаційних систем»

Напрями наукових досліджень:
«Автоматичне стиснення та обробка багатоканальних даних»
«Системи та технології хмарних обчислень»

Контактна інформація:
mail: m.zriakhov@khai.edu
tel/telegram: +38(050)9780367

2. Опис навчальної дисципліни

Форма здобуття освіти	*Денна*
Семестр	*4/2*[*]
Мова викладання	Українська
Тип дисципліни	*Обов'язкова*
Обсяг дисципліни: кредити ЄКТС/ кількість годин	*денна*: 4 кредити ЄКТС / 120 годин (64 аудиторних, з яких: лекції – 32, практичні – 32; СРЗ – 56)
Види навчальної діяльності	Лекції, практичні, самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль – іспит
Пререквізити	*Захист інформації в інфокомунікаціях, UNIX-подібні операційні системи*

[*] Для здобувачів, які навчаються за скороченим терміном.

3. Мета та завдання навчальної дисципліни, переліки компетентностей та очікуваних результатів навчання

Мета – даної програми навчання полягає в тому, щоб надати здобувачам глибокі знання та практичний досвід у створенні, розгортанні та управлінні додатками за допомогою Docker та Kubernetes. Завдяки цьому вони зможуть повноцінно взаємодіяти та ефективно працювати в сучасних розподілених та мікросервісних середовищах.

Завдання: дисципліни «Адміністрування інфокомунікаційних систем»: набуття фундаментальних знань Docker та контейнерних технологій, включаючи роботу з образами, мережами та Docker Compose; розвиток вмінь з впровадження та конфігурації Kubernetes кластерів, розуміння основних понять та компонентів; вивчення і розуміння процесу розгортання, масштабування та оновлення додатків у Kubernetes; навчання використання томів (persistent volumes) та конфігурацій (ConfigMaps) для забезпечення стійкості даних та функціональності додатків; розуміння ролі моніторингу, безпеки та аварійного відновлення в мікросервісних архітектурах.

Компетентності, які набуваються:

Інтегральна компетентність:

Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми в області інформаційних систем та технологій, або в процесі навчання, що характеризуються комплексністю та невизначеністю умов, які потребують застосування теорій та методів інформаційних технологій.

Загальні компетентності:

- Здатність абстрактно мислити, аналізуючи архітектуру контейнерних систем, синтезуючи оптимальні рішення для розгортання та масштабування додатків;
- Здатність застосовувати знання з контейнеризації та оркестрації у практичних ситуаціях, вирішуючи завдання розгортання, налаштування та оптимізації додатків;
- Здатність розуміти предметну область сучасних технологій контейнеризації та оркестрації, а також їх застосування у професійній діяльності ІТ-спеціаліста;
- Здатність вчитися і оволодівати сучасними знаннями у сфері DevOps, самостійно опановувати нові інструменти та підходи до роботи з Docker і Kubernetes;
- Здатність розробляти та управляти проєктами, використовуючи інструменти контейнеризації та оркестрації для організації ефективної командної роботи.

Здатність ухвалювати рішення та діяти, дотримуючись неприпустимості корупції та будь-яких інших проявів недоброчесності.

Фахові компетентності:

- Здатність аналізувати об'єкти проектування або функціонування контейнерних систем та їх інфраструктури;
- Здатність застосовувати стандарти у сфері контейнеризації та оркестрації (Docker, Kubernetes) при розробці, інтеграції та підтримці сучасних ІТ-систем і сервісів;
- Здатність до проектування, розробки, налаштування та вдосконалення контейнерних рішень, забезпечення їх інтеграції з програмно-апаратною інфраструктурою організації;
- Здатність проєктувати, розробляти та використовувати засоби автоматизації розгортання, масштабування та моніторингу додатків у контейнерних середовищах;
- Здатність використовувати сучасні методи аналізу даних, моніторингу та логування для підтримки функціонування контейнерних додатків і сервісів;
- Здатність до вибору, проєктування, розгортання, інтегрування, управління, адміністрування та супроводження контейнерних платформ, сервісів та інфраструктури організації на основі Docker і Kubernetes;
- Здатність управляти та користуватися сучасними засобами інфокомунікацій, технологіями контейнеризації та оркестрації для забезпечення ефективної роботи ІТ-систем (у тому числі з використанням хмарних рішень та Інтернету).

Програмні результати навчання:

- Аргументувати вибір програмних і технічних засобів для створення контейнерних платформ та оркестрації (Docker, Kubernetes) на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи та експлуатаційних умов; мати навички налаштування та тестування контейнерних рішень;
- Застосовувати правила оформлення проектних матеріалів щодо впровадження контейнерних технологій, знати склад та послідовність виконання проектних робіт з урахуванням вимог відповідних нормативно-правових документів для професійної діяльностіх;
- Здійснювати системний аналіз архітектури підприємства та його IT-інфраструктури, проводити розроблення та вдосконалення її елементної бази і структури із застосуванням сучасних технологій контейнеризації та оркестрації.

4. Зміст навчальної дисципліни

МОДУЛЬ 1

Змістовний модуль 1. Вивчення Docker

Тема 1. Вступ до контейнерів та оркестрації

Анотація: Ознайомлення з поняттями контейнеризації, віртуалізації, основними інструментами Docker та Kubernetes, їх роллю у сучасній розробці та деплойменті додатків.

Лекція 1: Вступ до контейнерів і віртуалізації. Огляд Docker та Kubernetes.

Самостійна робота здобувача освіти: Вивчення основних понять контейнеризації, перегляд офіційної документації Docker та Kubernetes.

Виконання практичного завдання: Розгортання додатку на Docker.

Тема 2. Знайомство з Docker

Анотація: Вивчення основ Docker, встановлення на різних ОС, знайо ство з Docker CLI та базовими командами для роботи з контейнерами.

Лекція 1: Вступний огляд Docker та контейнерів. Встановлення Docker на різних операційних системах. Ознайомлення з Docker CLI та основними командами.

Самостійна робота здобувача освіти: Встановлення Docker на власному комп’ютері, ознайомлення з офіційною документацією.

Виконання практичного завдання: Оптимізація Docker-образів.

Тема 3. Робота з Docker-образами

Анотація: Створення та оптимізація Docker-образів, вивчення основних та прихованих інструкцій Dockerfile, зменшення розміру образів.

Лекція 1: Створення Docker-образів за допомогою Dockerfile. Знайомство з базовими інструкціями Dockerfile (FROM, RUN, COPY, EXPOSE).

Лекція 2: Приховані інструкції (ENTRYPOINT, CMD, VOLUME, ARG). Основи оптимізації Docker-образів (скорочення розміру, скорочення кількості шарів тощо).

Самостійна робота здобувача освіти: Створення власного Dockerfile для простого додатку.

Виконання практичного завдання: Створення файлів YAML для композітних сервісів.

Тема 4. Використання репозиторіїв та Docker Hub

Анотація: Робота з публічними та приватними репозиторіями Docker, завантаження та публікація образів, налаштування власного реєстру.

Лекція 1: Робота з публічними та приватними репозиторіями. Завантаження та скачування Docker-образів. Регістр Docker на власному вузлі.

Самостійна робота здобувача освіти: Ознайомлення з Docker Hub, створення власного репозиторію.

Виконання практичного завдання: Управління ресурсами контейнерів і відстеження використання ресурсів.

Тема 5. Мережі в Docker

Анотація: Вивчення мережевих можливостей Docker, налаштування внутрішніх і зовнішніх мереж, використання різних драйверів для багатоконтейнерних додатків.

Лекція 1: Внутрішні і зовнішні мережі контейнерів. Docker мережеві драйвери та їх відмінності. Налаштування Docker мереж для багатоконтейнерних додатків.

Самостійна робота здобувача освіти: Вивчення документації щодо мереж Docker.

Виконання практичного завдання: Docker Networking.

Тема 6. Управління контейнерами та життєвим циклом контейнерів

Анотація: Ознайомлення з Docker Compose, створення та керування багатоконтейнерними додатками, використання зовнішніх сервісів та образів. Лекція 1: Огляд Docker Compose та його синтаксису. Створення файлів YAML для композітних сервісів.

Лекція 2: Включення посилань на зовнішні образи, сервіси та репозиторії у файл Docker Compose. Керування багатоконтейнерними додатками за допомогою Docker Compose.

Самостійна робота здобувача освіти: Створення власного docker-compose.yml для простого проекту.

Виконання практичного завдання: Встановлення та налаштування Kubernetes.

Тема 7. Docker Compose

Анотація: Управління користувачами та групами у контейнерах, зміна параметрів, робота з конфігураційними файлами, виконання команд від імені іншого користувача.

Лекція 1: Додавання користувачів. Зміна параметрів користувачів. Зміна пароля. Встановлення пустого пароля користувача. Отримання інформації про користувача. Знищення користувача. Створення, зміна та знищення групи. Файли конфігурації користувачів. Виконання команд від імені іншого користувача.

Самостійна робота здобувача освіти: Ознайомлення з best practices щодо безпеки користувачів у Docker.

Виконання практичного завдання: Docker Compose, Робота з Pod'ами.

Модульний контроль 1 Вивчення Docker (виконання тестових завдань)

Змістовний модуль 2. Вивчення Kubernetes

Тема 1. Основи Kubernetes

Анотація: Вступ до Kubernetes, основні поняття та компоненти, встановлення та базове налаштування кластера, робота з Kubectl.

Лекція 1: Основні поняття та компоненти. Вступ до управління кластером і Kubectl. Встановлення та налаштування Kubernetes.

Самостійна робота здобувача освіти: Ознайомлення з офіційною документацією Kubernetes, встановлення Minikube або іншого локального кластера.

Виконання практичного завдання: Розгортання та оновлення додатків за допомогою Deployment.

Тема 2. Kubernetes ресурси та розгортання

Анотація: Вивчення основних ресурсів Kubernetes, робота з Pod, ReplicaSet, Deployment, ConfigMap, Secret, Service, Ingress, томами та Persistent Volume.

Лекція 1: Поди, ReplicaSets та Deployments.

Лекція 2: Конфігурація, секрети та ConfigMaps.

Лекція 3: Services, Ingress та мережеві функціональність.

Лекція 4: Томи та Persistent Volume.

Самостійна робота здобувача освіти: Створення YAML-маніфестів для різних ресурсів Kubernetes.

Виконання практичного завдання: Робота з Services та Networking в Kubernetes, Конфігурація та секрети в Kubernetes, Застосування Persistent Volumes та StatefulSets.

Тема 3. Інтеграція Docker та Kubernetes

Анотація: Інтеграція Docker-образів у Kubernetes, типові сценарії розгортання контейнерів, використання приватних реєстрів.

Лекція 1: Застосування Docker в Kubernetes. Інтеграція Docker-образів в Kubernetes. Типові сценарії розгортання контейнерів на базі Docker.

Самостійна робота здобувача освіти: Підготовка власного Docker-образу для розгортання у Kubernetes.

Виконання практичного завдання: Моніторинг та логи в Kubernetes.

Тема 4. Глибше вивчення Kubernetes

Анотація: Вивчення розширених ресурсів Kubernetes, таких як DaemonSet, StatefulSet, Custom Resource Definitions та оператори.

Лекція 1: DaemonSets, StatefulSets. Custom Resource Definitions та оператори.

Самостійна робота здобувача освіти: Ознайомлення з прикладами використання CRD та операторів у реальних проектах.

Виконання практичного завдання: Застосування Persistent Volumes та StatefulSets.

Модульний контроль 2: Вивчення Kubernetes (виконання тестових завдань)

5. Індивідуальні завдання

Навчальним планом не передбачені

6. Методи навчання

Методи стимулювання і мотивації навчально-пізнавальної діяльності:

- переконання у значущості навчання;
- вимоги;
- створення ситуації зацікавленості.

Методи організації та здійснення навчально-пізнавальної діяльності:

- пояснювально-ілюстративний;
- словесний (розповідь, лекція, бесіда, пояснення);
- наочний (ілюстрація, демонстрація);
- практичний (вправи).

7. Методи контролю

1) модульний контроль (виконання тестових завдань);

2) поточний контроль (оцінювання роботи студентів на практичних та лекційних заняттях).

8. Критерії оцінювання та розподіл балів, які отримують здобувачі освіти

Таблиця 8.1 – Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість годин	Сумарна кількість балів
Змістовний модуль 1			
Робота на лекціях	0…1	7	0…7
Виконання практичних завдань	0…3	7	0…21
Модульний контроль	0…26	1	0…26
Змістовний модуль 2			
Робота на лекціях	0…1	4	0…4
Виконання практичних завдань	0…4	4	0…16
Модульний контроль	0…26	1	0…26
Усього за семестр			**0…100**

Семестровий контроль (іспит) проводиться у разі відмови студента від балів поточного тестування й за наявності допуску до заліку (виконання всіх завдань з практичних робіт). Під час складання семестрового заліку студент має можливість отримати максимум 100 балів

Білет для заліку складається з 50 тестових запитань. Максимальна кількість балів, яку можна отримати за вірну відповідь на одне запитання, складає 1 бал. Підсумкова оцінка за іспит обчислюється як сума оцінок, що отримана здобувачем за відповіді на білет та оцінок за виконання практичних завдань під час семестру (максимум 100 балів).

Таблиця 8.2 – Шкали оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційний залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

Критерії оцінювання роботи здобувача освіти протягом семестру

Задовільно (60-74). Показати мінімум знань та умінь. Захистити всі практичні роботи. Набрати не менше ніж 60% від максимуму під час проходження модульних контролів.

Добре (75-89). Продемонструвати середній рівень знань та умінь. Захистити всі практичні роботи. Набрати не менше ніж 75% від максимуму під час проходження модульних контролів. Відвідувати лекції.

Відмінно (90-100). Продемонструвати високий рівень знань та умінь. Захистити всі практичні роботи з високими балами. Набрати не менше ніж 90% від максимуму під час проходження модульних контролів. Відвідувати лекції й брати активну участь під час аудиторних занять.

9. Політика навчального курсу

Відвідування занять. Регуляція пропусків. Інтерактивний характер курсу передбачає відвідування практичних занять. Здобувачі освіти, які за певних обставин не можуть відвідувати практичні заняття регулярно, повинні протягом тижня узгодити із викладачем графік індивідуального відпрацювання пропущених занять. Окремі пропущені заняття мають бути відпрацьовані на найближчій консультації протягом одного-двох тижнів після їх пропуску. Відпрацювання занять здійснюється письмово у формі складання звіту за темою практичного заняття, визначеним темою заняття або дозволяється письмове відпрацювання пропущених занять шляхом виконання індивідуального письмового завдання.

Дотримання вимог академічної доброчесності здобувачами освіти під час вивчення навчальної дисципліни. Під час вивчення навчальної дисципліни здобувачі освіти мають дотримуватися загальноприйнятих морально-етичних норм і правил поведінки, вимог академічної доброчесності, передбачених Кодексом академічної доброчесності та забезпечення її дотримання Національного аерокосмічного університету «Харківський авіаційний інститут». Очікується, що роботи здобувачів освіти будуть їх оригінальними дослідженнями або міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших здобувачів освіти становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі здобувача освіти є підставою для її незарахування викладачем незалежно від масштабів плагіату чи обману.

Вирішення конфліктів. Порядок і процедури врегулювання конфліктів, пов’язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що

можуть виникнути під час навчання, а також правила етичної поведінки регламентуються Кодексом етики в Національному аерокосмічному університеті «Харківський авіаційний інститут».

10. Методичне забезпечення

1. Комплект слайдів презентацій з матеріалами лекцій: (https://mentor.khai.edu/course/view.php?id=7093)

11. Рекомендована література

Базова

1. Nigel Poulton. - Docker Deep Dive. - Independently published, 2020. - 432 pages.
2. James Turnbull. - The Docker Book: Containerization is the new virtualization. - James Turnbull, 2014. - 342 pages.
3. rendan Burns, Joe Beda, Kelsey Hightower. - Kubernetes: Up and Running: Dive into the Future of Infrastructure. - O'Reilly Media, 2017. - 228 pages.
4. Marko Luksa. - Kubernetes in Action. - Manning Publications, 2018. - 416 pages.

Допоміжна

1. Adrian Mouat. - Using Docker: Developing and Deploying Software with Containers. - O'Reilly Media, 2015. - 354 pages
2. Bilal Zafar, Roland Huß. - Kubernetes Patterns: Reusable Elements for Designing Cloud-Native Applications. - Red Hat, 2019. - 266 pages.

12. Інформаційні ресурси

1. Docker Hub [Електронний ресурс] / Режим доступу: https://hub.docker.com/
2. GitHub репозиторій Docker [Електронний ресурс] / Режим доступу: https://github.com/docker/docker
3. Docker Playground [Електронний ресурс] / Режим доступу: https://labs.play-with-docker.com/
4. Interactive Kubernetes Tutorials [Електронний ресурс] / Режим доступу: https://kubernetes.io/docs/tutorials/kubernetes-basics/
5. GitHub репозиторій Kubernetes [Електронний ресурс] / Режим доступу: https://github.com/kubernetes/kubernetes
6. Сайти кафедри: http://k504.khai.edu, http://k504.xai.edu.ua
7. Сайт бібліотеки ХАІ: library.khai.edu