

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

ЗАТВЕРДЖУЮ:

Декан факультету комп'ютерних
інформаційних технологій
Ігор ЯКИМЕНКО

“29” 2025 р.

ЗАТВЕРДЖУЮ:

Проректор з
науково-педагогічної роботи
Віктор ОСТРОВЕРХОВ

“29” 2025 р.

РОБОЧА ПРОГРАМА

з дисципліни

«Розробка клієнт-серверних застосунків»

ступінь вищої освіти – бакалавр
галузь знань – 12 «Інформаційні технології»
спеціальність – 122 «Комп'ютерні науки»
освітньо-професійна програма «Штучний інтелект»

Кафедра інформаційно-обчислювальних систем і управління

Форма навчання	Курс	Семестр	Лекції (год.)	Лабор. (год.)	ІРС (год.)	Тренінг (год.)	СРС (год.)	Разом (год.)	Екзамен (сем.)
Денна	3	5	30	30	4	8	78	150	5

Тернопіль – ЗУНУ
2025

29.08.2025
[Підпис]

Робоча програма складена на основі освітньо-професійної програми «Штучний інтелект» підготовки бакалавра галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки», затвердженої Вченою радою ЗУНУ (протокол № 10 від 23.06.2023 р.).

Робочу програму склав доцент кафедри інформаційно-обчислювальних систем і управління, к.т.н. Биковий Павло Євгенович

Робоча програма затверджена на засіданні кафедри інформаційно-обчислювальних систем і управління, протокол № 1 від 26 серпня 2025 р.

В.о. завідувача кафедри



Надія ВАСИЛЬКІВ

Гарант освітньо-професійної програми "Штучний інтелект",
канд. техн. наук, доцент



Василь КОВАЛЬ

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «РОЗРОБКА КЛІЄНТ-СЕРВЕРНИХ ЗАСТОСУВАНЬ»

1. ОПИС ДИСЦИПЛІНИ

«РОЗРОБКА КЛІЄНТ-СЕРВЕРНИХ ЗАСТОСУВАНЬ»

Дисципліна – «Розробка клієнт-серверних застосувань»	Галузь знань, спеціальність, СВО	Характеристика навчальної дисципліни
Кількість кредитів ECTS - 5	Галузь знань – 12 «Інформаційні технології»	Статус дисципліни – обов’язкова дисципліна циклу професійної підготовки Мова навчання - українська
Кількість залікових модулів – 5	Спеціальність – 122 «Комп’ютерні науки»	Рік підготовки: 3 Семестр: 5
Кількість змістових модулів – 5	Освітньо-професійна програма: «Штучний інтелект»	Лекції: 30 год. Лабораторні заняття: 30 год.
Загальна кількість годин - 150	Ступінь вищої освіти – бакалавр	Самостійна робота: 78 год., Тренінг: 8 год. Індивідуальна робота – 4 год.
Тижневих годин – 10 год., з них аудиторних – 4 год.		Вид підсумкового контролю: екзамен

2. МЕТА ТА ЗАВДАННЯ ВИВЧЕННЯ ДИСЦИПЛІНИ «РОЗРОБКА КЛІЄНТ-СЕРВЕРНИХ ЗАСТОСУВАНЬ»

2.1. Мета вивчення дисципліни

Мета вивчення дисципліни «Розробка клієнт-серверних застосувань» – дати студентам систематизовані відомості про основні принципи розробки клієнтських та серверних застосувань, забезпечити розуміння сервісної архітектури, знати призначення та використання контейнерів, вміти працювати з API, використовувати хмарні технології.

2.2. Завдання вивчення дисципліни

- оволодіння основними поняттями клієнт-серверних технологій;
- ознайомлення з сучасними програмними засобами створення клієнт-серверних застосувань;
- ознайомлення із основними алгоритмами розробки клієнт-серверних застосувань;
- ознайомлення із методами передачі інформації між клієнтами і серверами;
- набуття практичних навичок роботи з сервісом Postman;
- ознайомлення з WSDL та набуття практичних навичок з його використанням;
- ознайомлення з SOAP та набуття практичних навичок з його використанням;
- оволодіння основними поняттями сервіс та мікросервіс;
- набуття практичних навичок по роботі з Rest API/ RESTfull веб-сервісами;
- ознайомлення із основними компонентами CRM систем на прикладі Salesforce;
- Знати принципи та архітектуру web-сервісів;
- набуття практичних навичок по роботі з інструментом Docker;
- набуття практичних навичок по роботі з технологією Kubernetes;
- оволодіння основними поняттями хмарних технологій та хмарних моделей IaaS, PaaS, SaaS;
- набуття практичних навичок використання Docker контейнерів при розробці хмарних веб-сервісів;
- набуття практичних навичок роботи з Google App Engine;
- набуття практичних навичок з використання технологій HTML та JavaScript компонент для роботи з WebRTC;
- набуття практичних навичок із роботою з Telnet клієнтом.

2.3. Найменування та опис компетентностей, формування котрих забезпечує вивчення дисципліни:

СК9. Здатність 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.

СК13. Здатність до розробки мережевого програмного забезпечення, що функціонує на основі різних топологій структурованих кабельних систем, використовує комп'ютерні системи та мережі передачі даних та аналізує якості

роботи комп'ютерних мереж.

2.4. Передумови для вивчення дисципліни:

Вивчення курсу „Розробка клієнт-серверних застосувань” передбачає наявність систематичних та ґрунтовних знань із дисциплін: «Алгоритми та структури даних», «Сучасні парадигми програмування», «Бази і сховища даних».

2.5. Результати навчання:

ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

3. Програма навчальної дисципліни

«РОЗРОБКА КЛІЄНТ-СЕРВЕРНИХ ЗАСТОСУВАНЬ»

Змістовий модуль 1. Методи взаємодії між клієнтами і серверами

Тема 1. Клієнт-серверна модель та розробка програмного забезпечення

Основні поняття клієнта і сервера. Привілеї та складності. Стандартне та нестандартне клієнтське програмне забезпечення. Орієнтовані та неорієнтовані на з'єднання сервери.

Тема 2. Методи передачі інформації між серверами

Методи POST, GET, PUT, DELETE. Коди статусу відповідей, структури запиту та відповіді. Методи візуальної діагностики запитів і відповідей сервера за допомогою сервісу Postman.

Тема 3. Веб-сервіси SOAP

Принципи та архітектура web-сервісів. Основи WSDL та принципи його використання. Основи SOAP та принципи його використання. Специфікації web-сервісів SOAP. Принципи створення web-сервісів SOAP.

Змістовий Модуль 2. Поняття сервісів та мікросервісів

Тема 4. Поняття сервісів та мікросервісів

Переваги та недоліки мікросервісної архітектури. Міграція та реалізація мікросервісів. Створення програми на основі мікросервісів. Підхід на основі служб.

Тема 5. Основи проектування інтерфейсу програмування API.

Концепція та структура REST API. Створення API проекту. Формування API методів.

Тема 6. CRM Системи.

Основні компоненти CRM систем. Інтеграція з сайтами, маркетплейсами, поштовими системами, чатботами. Архітектура CRM системи Salesforce.

Змістовий модуль 3. Технології контейнеризації у системах керування потоками даних

Тема 7. Інструмент Docker

Призначення контейнерів. Архітектура Docker. Основні команди. Використання Dockerfile. Створення образів. Взаємозв'язок з контейнером. Управління даними в Docker контейнері.

Тема 8. Технологія Kubernetes

Історія створення та призначення. Завантаження/перезавантаження контейнерів. Pod-оболонки. Планувальник Kubernetes. Адреси сервісів. Робота з kubectl. Засоби моніторингу

Змістовий модуль 4. Хмарні технології

Тема 9. Основи хмарних технологій

Основні поняття та концепції хмарних технологій. Ахітектура хмарних обчислень. Основні класи хмарних систем. Моделі IaaS, PaaS, SaaS, їх основні властивості та приклади.

Тема 10. Засоби розробки програмного забезпечення в хмарних системах

Створення API для хмарного додатку. Взаємодія з базою даних. Використання Docker контейнерів розробці хмарних веб-сервісів

Тема 11. Обчислювальні кластери в хмарних системах.

Основи функціонування центрів обробки даних, їх характеристики та використання. Робота з Google App Engine.

Змістовий модуль 5. Потокова передача даних між клієнтом та сервером

Тема 12. Проект WebRTC

Передача звуку і відео в браузерях. Основи роботи з WebRTC. Огляд HTML та JavaScript компонент для роботи з WebRTC.

Тема 13. Робота з Telnet клієнтом

Термінал користувача. Потокова модель для Telnet клієнта. Алгоритм роботи Telnet клієнта.

Тема 14. Реалізація Telnet - клієнта

Відправка інформації. Відображення даних. Записи блоку даних на сервер. Взаємодія з місцевими клієнтами. Реалізація сценаріїв.

4. Структура залікового кредиту з дисципліни «РОЗРОБКА КЛІЄНТ-СЕРВЕРНИХ ЗАСТОСУВАНЬ»

Денна форма навчання

Тема		Кількість годин				
	Лекції	Лабора торні	ІР	Тренінг	Самостій на робота	Контрольні заходи
Змістовий модуль 1. Методи взаємодії між клієнтами і серверами						
Тема 1. Клієнт-серверна модель та розробка програмного забезпечення	2	-	-	2	4	Опитування під час заняття
Тема 2. Методи передачі інформації між серверами	2	6			6	Опитування під час заняття
Тема 3. Веб-сервіси SOAP	2	-			6	Опитування під час заняття
Змістовий Модуль 2. Поняття сервісів та мікросервісів						
Тема 4. Поняття сервісів та мікросервісів	2	-	1	4	4	Опитування під час заняття
Тема 5. Основи проектування інтерфейсу програмування API.	2	6			6	Опитування під час заняття
Тема 6. CRM Системи	2	4			6	Опитування під час заняття
Змістовий модуль 3. Технології контейнеризації у системах керування потоками даних						
Тема 7. Інструмент Docker	2	6	1	2	6	Опитування під час заняття
Тема 8. Технологія Kubernetes	2	-			6	Опитування під час заняття
Змістовий модуль 4. Хмарні технології						
Тема 9. Основи хмарних технологій	2	-	1	2	6	Опитування під час заняття
Тема 10. Засоби розробки програмного забезпечення в хмарних системах	2	-			6	Опитування під час заняття
Тема 11. Обчислювальні кластери в хмарних системах	4	2			6	Опитування під час заняття
Змістовий модуль 5. Потокова передача даних між клієнтом та сервером						
Тема 12. Проект WebRTC	2	6	1	2	4	Опитування під час заняття
Тема 13. Робота з Telnet клієнтом	2	-			6	Опитування під час заняття
Тема 14. Реалізація Telnet клієнта	2	-			6	Опитування під час заняття
ВСЬОГО	30	30	4	12	78	

5. Тематика лабораторних занять

1. Огляд Rest Api/ Rest Full систем
2. Робота з Salesforce Rest API
3. Docker основи
4. Створення Rest API серверу
5. Перенесення локального проекту в «хмарне середовище» Google Cloud
6. Перенесення локального проекту на VPS сервер

6. Самостійна робота

Студенти проводять дослідження на обрану з поданого нижче переліку тему та готують результати дослідження у вигляді доповіді (презентація на 15-20 слайдів, 10-15 хв). Практично реалізувати приклад використання наведеної технології та представити результат. Студенти також можуть самостійно запропонувати тему дослідження, відповідно до власних вподобань та в рамках даного предмету і погодити її з викладачем.

Перелік тем:

- | | |
|-------------------------------------|-------------------------------|
| 1. Google Cloud Storage. | 18. AWS Transcribe. |
| 2. Google Cloud Translation API. | 19. AWS Step functions. |
| 3. Google Cloud Vision API. | 20. AWS Translate. |
| 4. Google Cloud Speech-to-Text API. | 21. AWS Textract. |
| 5. Google Cloud Text-to-Speech API. | 22. AWS Rekognition. |
| 6. Google Video Intelligence API. | 23. AWS IoT Core. |
| 7. Google Colab for ML. | 24. AWS S3. |
| 8. Google Cloud Firestore. | 25. AWS CodeGuru. |
| 9. Google Firebase. | 26. AWS Lex. |
| 10. Google Vertex AI. | 27. AWS Serverless API. |
| 11. Google Cloud CDN. | 28. AWS Comprehend. |
| 12. Google Big Query | 29. Amazon DynamoDB. |
| 13. Google Cloud SQL | 30. Dialogflow API. |
| 14. Azure Custom Vision | 31. MongoDB Atlas. |
| 15. Automated ML in Azure. | 32. Docker. |
| 16. Azure Databricks. | 33. Google Kubernetes Engine. |
| 17. Azure Synapse Analytics | |

7. Тренінг з дисципліни

№ з/п	Вид роботи	Порядок проведення тренінгу
1.	Вступна частина	Ознайомлення студентів з темою тренінгового заняття «Розміщення клієнт-серверної технології в мережі Інтернет та організація її роботи» і видача завдання
2.	Практична частина	1. Здійснити тестування клієнт серверного додатку на локальній мережі перед тим як розмістити його на сервері в Інтернеті, а саме: 2. Ознайомитись з умовами та можливостями розміщення клієнт серверного додатку на безкоштовному гостьовому Інтернет-сервері. 3. Представити результати у вигляді короткого звіту, що повинен містити копії розміщеного додатку в інтернеті. Описати хід виконання кожного завдання.
3.	Підведення підсумків	Обговорення результатів виконаних завдань

8. Засоби оцінювання та методи демонстрування результатів навчання

У процесі вивчення дисципліни «Розробка клієнт-серверних застосувань» використовуються наступні засоби оцінювання та методи демонстрування результатів навчання:

- поточне тестування та опитування;
- оцінювання результатів лабораторних робіт;
- оцінювання виконання завдань тренінгу;
- оцінювання виконання завдань самостійної роботи;
- екзамен.

9. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни «Розробка клієнт-серверних застосувань» визначається як середньозважена величина, в залежності від питомої ваги кожної складової залікового кредиту:

Модуль 1		Модуль 2		Модуль 3	Модуль 4	Модуль 5
10 %	10 %	10 %	10 %	5 %	15%	40 %
Поточне оцінювання	Модульний контроль 1	Поточне оцінювання	Модульний контроль 2	Тренінг	Самостійна робота	Екзамен
Оцінка визначається як середнє арифметичне з оцінок, отриманих за виконання і захист 3 лабораторних робіт	Модульна контрольна робота (15 тестових завдань та 1 практичне)	Оцінка визначається як середнє арифметичне з оцінок, отриманих за виконання і захист 3 лабораторних робіт	Модульна контрольна робота (15 тестових завдань та 1 практичне)	Виконання завдань під час тренінгу (3 завдання)	Виконання завдання для самостійної роботи	2 теоретичні питання з розгорнутими відповідями та 2 практичні завдання

Модуль 1 та Модуль 2 включають по дві складові за якими оцінюють знання студента:

1. **“Поточне оцінювання”** - оцінка визначається як середнє арифметичне оцінок, отриманих за виконання та захист 4 лабораторних робіт.

ECTS, бали	Критерії оцінювання
A 90 – 100	Виконані всі завдання лабораторної роботи. Реалізований код програми є працездатним, оптимізованим та структурованим. Правильно виконані всі етапи роботи (налаштування середовища, програмування, відлагодження, перевірка результатів). Звіт оформлено акуратно з поясненням принципів роботи програми. Студент впевнено відповідає на запитання викладача, надає повні пояснення, вміє узагальнити результати та відповідає на додаткові запитання. Звіт оформлено акуратно з поясненням принципів роботи програм та без помилок.
B 85 – 89	Виконані всі основні завдання та частина додаткових. Результати працездатні, хоча присутні дрібні неточності у кодї чи описі. Студент демонструє добре розуміння теми та пояснює ключові рішення. Звіт містить незначні недоліки в оформленні.
C 75 – 84	Виконані всі основні завдання, але без додаткових. Програма працює, але з обмеженнями (відсутні додаткові перевірки, код програми не оптимізований, відсутня обробка виняткових ситуацій). Частково зроблений аналіз отриманих результатів. У звіті є недоліки в оформленні або опис.
D 65 – 74	Виконана лише частина основних завдань. Представлені рішення містять суттєві недоліки в кодї або логіці роботи, але студент демонструє базове розуміння теми. Робота виконана за зразком, без власних пояснень. Аналіз та висновки неповні або неточні. У звіті є помилки чи пропуски.
E 60 – 64	Виконано мінімальну частину базових завдань з допомогою викладача. Отримані результати обмежено працездатні. Аналіз та висновки відсутні або поверхневі. Звіт має значні недоліки.
FХ 35 – 59	Більшість завдань не виконано. Код програми або відсутній, або непрацездатний. Студент не може пояснити свої дії та не розуміє суті лабораторної роботи. Звіт відсутній або не відповідає вимогам.

2. **“Модульний контроль”** - оцінка визначається за результатами складання тесту. Тест включає 15 питань теоретичного характеру, які охоплюють основні теми дисципліни та одне завдання практичного характеру. Загальне число балів, що отримав студент, показує наскільки він має глибокі, міцні і системні знання із дисципліни.

Виконання тестових завдань оцінюється за 60-бальною шкалою, тому одна правильна відповідь на тестове завдання оцінюється кількістю балів, яка розраховується наступною формулою:

$$K = 100/N,$$

де: K - кількість балів за 1 питання; N - кількість питань у тесті.

Загальна кількість балів за весь тест оцінюється наступною формулою:

$$Z = K * P,$$

де: Z - загальна кількість балів; K - кількість балів за 1 запитання; P - кількість правильних відповідей на тестові завдання.

Практичне завдання оцінюється у 40 балів.

Модуль 3 - “Тренінг” – оцінка визначається, як середнє арифметичне оцінок, отриманих за виконання окремих завдань тренінгу, відповідно п. 7 даної робочої програми.

ECTS, бали	Критерії оцінювання
A 90 – 100	Завдання тренінгу виконані в повному обсязі та реалізовані на високому рівні з урахуванням всіх вимог. Показана креативність або оптимізація, що покращує результат. Студент упевнено та аргументовано пояснює прийняті рішення, відповідає на запитання. Демонструє впевнене володіння матеріалом. Продемонстровано якісний кінцевий програмний продукт/прототип.
B 85 – 89	Виконано всі основні завдання. Присутні незначні недоліки у деталях або оформленні. Результат відповідає вимогам та є цілком працездатним.
C 75 – 84	Виконано більшість завдань. Деякі вимоги реалізовані поверхово. Представлений результат працездатний, але має обмежену функціональність.
D 65 – 74	Завдання виконано частково. Присутні суттєві недоліки в реалізації або якості результату. Досягнуто мінімально необхідного рівня. Реалізована основна ідея, але без глибокої деталізації.
E 60 – 64	Виконано лише базову частину завдань. Робота відповідає мінімальним вимогам та демонструє обмежений рівень знань і практичних умінь.
FX 35 – 59	Завдання виконано частково, значна частина ключових етапів відсутня. Результат неповний або не відповідає встановленим вимогам. Демонстрація роботи програми/прототипу обмежена або неможлива.

Модуль 4 - “Самостійна робота” – оцінка визначається, як середнє арифметичне оцінок, отриманих під час виконання завдання самостійної роботи, а саме: 1) представлення проведеного дослідження з обраної теми; 2) аналіз отриманих результатів.

ECTS, бали	Критерії оцінювання
A 90 – 100	Робота виконана повністю, тема розкрита всебічно. Використано достатню кількість джерел, є чітка структура (вступ, основна частина, висновки). Аналіз результатів глибокий та аргументований, зроблено власні висновки. Оформлення відповідає вимогам.
B 85 – 89	Тема досліджена на високому рівні, використано якісні джерела. Присутня логічна структура та аналіз, проте глибина висновків або критичний огляд обмежені. Невеликі недоліки в оформленні.
C 75 – 84	Виконано основний обсяг роботи, тема розкрита частково. Аналіз поверховий, висновки загальні. Використано обмежену кількість джерел, але робота залишається цілісною та

ECTS, бали	Критерії оцінювання
	зрозумілою.
D 65 – 74	Тема досліджена неповністю, значна частина матеріалу подана описово без належного аналізу. Висновки слабкі або нечіткі. Є помітні недоліки в структурі та оформленні.
E 60 – 64	Представлено лише базову інформацію по темі, відсутній достатній аналіз. Висновки формальні. Робота відповідає мінімальним вимогам.
FX 35 – 59	Виконано частину завдання. Тема розкрита поверхово, аналіз практично відсутній. Структура нечітка, джерела використані обмежено. Робота не відповідає більшості вимог

Модуль 5 - “Екзамен” – оцінка визначається за результатами екзаменаційного білету, який складає 2 теоретичні питання з розгорнутими відповідями по 30 балів та 2 практичні завдання по 20 балів.

Шкала оцінювання

За шкалою університету	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)

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

№	Найменування	Номер теми
1	Сервіс Postman	2, 3
2.	CRM Salesforce	6
3	Інструмент Docker	7, 10
4	OS Linux	7
5	База даних MySQL	5
6	Сервіс WebRTC	12
7	Платформа Google App Engine	9, 10, 11
8	Клієнт Telnet	13, 14
9	Редактор коду Visual Studio Code	5, 6, 8, 12

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

1. Arnaud Lauret. The Design of Web APIs. Publisher: Manning Publications, 2019, 400 p.
2. Dave Westerveld. API Testing and Development with Postman: A practical guide to creating, testing, and managing APIs for automated software testing. Publisher: Packt Publishing, 2021, 340 p.
3. Ian Miell, Aiden Hobson Sayers. Docker in Practice. Publisher: Manning Publications, Year: 2019, 384 p.
4. Rui Costa, Drew Hodun. Google Cloud Cookbook: Practical Solutions for Building and Deploying Cloud Services. Publisher: O'Reilly Media, 2021, 282 p.
5. Paul Goodey. Salesforce CRM - The Definitive Admin Handbook: Build, configure, and customize Salesforce CRM and mobile solutions, 5th Edition. Publisher: Packt, 2019, 782 p.
6. Електронний курс з дисципліни «Розробка клієнт-серверних застосувань» для студентів галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки» на платформі Moodle ЗУНУ /Биковий П.Є. - Тернопіль, 2021. <https://moodle.wunu.edu.ua>.
7. Биковий П.Є. Опорний конспект лекцій з дисципліни «Розробка клієнт-серверних застосувань» для студентів спеціальності 122 «Комп'ютерні науки». - Тернопіль: ЗУНУ, 2021. - 80 с.
8. Биковий П.Є., Осолінський О.Р. Методичні вказівки до виконання лабораторних робіт з дисципліни «Розробка клієнт-серверних застосувань» для студентів спеціальності «Комп'ютерні науки». - Тернопіль: ЗУНУ, 2021. - 68 с.