



# Технології розробки серверного програмного забезпечення(backend)

## Робоча програма навчальної дисципліни (Силабус)

### ● 1. Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>123 Комп'ютерна інженерія</i>
Освітня програма	<i>Комп'ютерна інженерія</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>очна (денна)</i>
Рік підготовки, семестр	<i>3 курс (1 семестр)</i>
Обсяг дисципліни	<i>4 кредитів, 120 год.</i>
Семестровий контроль/ контрольні заходи	<i>Залік</i>
Розклад занять	<i>Згідно розкладу на осінній семестр поточного навчального року за адресою <a href="http://rozklad.kpi.ua">http://rozklad.kpi.ua</a></i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	Лектор: ас. кафедри ОТ, Валько Володимир Володимирович, <a href="mailto:valko.volodya@gmail.com">valko.volodya@gmail.com</a> Лабораторні: ас. кафедри ОТ, Валько Володимир Володимирович,, <a href="mailto:valko.volodya@gmail.com">valko.volodya@gmail.com</a>
Розміщення курсу	Платформа дистанційного навчання «Сікорський» в середовищі Google Workspace for Education: Технології розробки серверного програмного забезпечення(backend). <a href="https://classroom.google.com/c/NDg5Mzc5ODM4MTc1?cjc=7cmg2mo">https://classroom.google.com/c/NDg5Mzc5ODM4MTc1?cjc=7cmg2mo</a>

### ● 2. Програма навчальної дисципліни

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

Мета дисципліни «Технології розробки серверного програмного забезпечення(backend)» – вивчення теоретичних основ, принципів побудови серверного програмного забезпечення та набуття практичних навичок у побудові таких проектів. Предметом вивчення кредитного модуля «Технології розробки серверного програмного забезпечення(backend)» є архітектура та функціональні можливості сучасного серверного програмного забезпечення; взаємодія з базами даних на сервері; особливості програмування сучасного серверного програмного забезпечення а також забезпечення їх підтримки.

Вивчення дисципліни підсилює наступні загальні та фахові компетенції:

- ЗК2. Здатність вчитися і оволодівати сучасними знаннями.
- ФК1. Здатність застосовувати законодавчу та нормативно правову базу, а також державні та міжнародні вимоги, практики і стандарти з метою здійснення професійної діяльності в галузі комп'ютерної інженерії.
- ФК2 Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

- ФК5. Здатність використовувати засоби і системи автоматизації проектування до розроблення компонентів комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем тощо.
- ФК9. Здатність системно адмініструвати, використовувати, адаптувати та експлуатувати наявні інформаційні технології та системи.
- ФК13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій.
- ФК14. Здатність проектувати системи та їхні компоненти з урахуванням усіх аспектів їх життєвого циклу та поставленої задачі, включаючи створення, налаштування, експлуатацію, технічне обслуговування та утилізацію.

**У відповідності до вищезазначеного, підсилені загальні та фахові компетенції дадуть наступні результати навчання:**

- знання фундаментальних концепцій, парадигм і основних принципів функціонування серверного програмного забезпечення;
- здатність до використання теоретичних, логічних та архітектурних основ для розробки серверного програмного забезпечення і вміння їх застосовувати при вирішенні професійних завдань;
- здатність розробляти серверне програмне забезпечення для типових сфер використання, в тому числі, з використанням систем версіонування та сучасних фреймворків;
- здатність розробки та використання драйверів елементів архітектурозалежного рівня на основі знання загальних принципів організації та функціонування вбудованих систем;
- здатність брати участь у командній роботі по проектуванню серверного програмного забезпечення;
- здатність формувати та забезпечувати вимоги щодо надійності серверного програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

## **2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)**

При вивченні дисципліни «Технології розробки серверного програмного забезпечення(backend)» доцільно використовувати знання, отримані при вивченні попередніх дисциплін: «Програмування», «Об'єктно-орієнтоване програмування», «Алгоритми та методи обчислень», «Системне програмування», «Структури даних і алгоритми», «Дискретна математика», «Інженерія програмного забезпечення».

Дисципліна є базовою для курсів із Ф-каталога вибіркових дисциплін, що входять до складу сертифікатної програми: «Технології розробки серверного програмного забезпечення(backend)» (вибіркова), «Технології проектування інтелектуальних систем» (вибіркова), «Технології тестування (QA) вбудованих систем» (вибіркова), «Управління інфраструктурними ІТ проектами» (вибіркова), а також є корисною для вивчення нормативних дисциплін «Організація баз даних», «Системне програмне забезпечення. Курсова робота», «Системне програмування».

## **3. Структура кредитного модуля**

**Вступ. Роль систем контролю версій в розробці програмного забезпечення**

### **Розділ 1. Системи контролю версій**

Тема 1.1. VCS

Тема 1.2. Як працює git

Тема 1.3. Основні операції в git. Просунуте використання git

### **Розділ 2. Python як мова програмування серверного ПЗ. Теорія**

Тема 2.1. Інтерпретатор python

Тема 2.2. Представлення об'єктів в пам'яті

Тема 2.3. Типи в python

- Тема 2.4. ООП в python
- Тема 2.5. Поліморфізм
- Тема 2.6. Інкапсуляція
- Тема 2.7. Наслідування
- Тема 2.8. Магічні методи python

### **Розділ 3. Python як мова програмування серверного ПЗ. Практика**

- Тема 3.1. Базові операції
- Тема 3.2. Особливості роботи функцій в python
- Тема 3.3. Особливості ООП в python на практиці
- Тема 3.4. Реалізація декораторів

### **Розділ 4. Фреймворки, api**

- Тема 4.1. Поняття фреймворку
- Тема 4.2. Робота з JSON
- Тема 4.3. Поняття REST API
- Тема 4.4. Основні принципи роботи з Docker
- Тема 4.5. Основні принципи роботи з docker compose
- Тема 4.6. Поняття бібліотеки
- Тема 4.7. Практикум з docker

### **Розділ 5. Бази даних**

- Тема 5.1. Поняття бази даних
- Тема 5.2. Основні види баз даних та сфери використання
- Тема 5.3. Робота з PostgreSQL
- Тема 5.4. Особливості транзакцій в PostgreSQL
- Тема 5.5. Блокування в PostgreSQL
- Тема 5.6. Особливості join в PostgreSQL
- Тема 5.7. Індeksi в PostgreSQL
- Тема 5.8. Реплікація PostgreSQL

### **Розділ 6. Аутентифікація та авторизація**

- Тема 6.1. Поняття аутентифікації
- Тема 6.2. Поняття авторизації
- Тема 6.3. Мультифакторна авторизація
- Тема 6.4. Ролі

### **Розділ 7. Стандарти та фреймворки аутентифікації**

- Тема 7.1. Поняття JWT
- Тема 7.2. Будова JWT
- Тема 7.3. OAuth2

### **Розділ 8. Безпека серверного програмного забезпечення**

- Тема 8.1. Поняття вразливості програмного забезпечення
- Тема 8.2. Основні типи вразливостей
- Тема 8.3. Детальний розгляд найпоширеніших вразливостей — CSRF, XSS, SQLi

## **4. Навчальні ресурси та матеріали**

### **4.1. Базова література**

1. Основи програмування. Python. Частина 1 [Електронний ресурс]: підручник для студ. спеціальності 122 "Комп'ютерні науки", спеціалізації "Інформаційні технології в біології та медицині" / А. В. Яковенко ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 1,59 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018. – 195 с.
2. Основи програмування алгоритмічною мовою Python [Електронний ресурс]: навч. посіб. для студ. освітньої програми «Комп'ютерні системи та мережі» спеціальності 123 «Комп'ютерна інженерія» / М.А. Новотарський – Електронні текстові дані (1 файл: 18 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2022. – 701с.
3. Бази даних та засоби управління. Практикум. [Електронний ресурс] : навч. посіб. для студ. спеціальності 123 – Комп'ютерна інженерія. / В.І. Павловський, А.В. Петрашенко, Д.В.

Победа; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 7,7 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2021. – 112 с.

## 4.2. Додаткова література

4. L.Ramalho . Fluent Python: Clear, Concise, and Effective Programming 1st edition, O'Reilly Media. 2015
5. David Beazley, Brian K. Jones Python Cookbook, 3rd edition, O'Reilly Media. 2015
6. Zed Shaw Learn Python the Hard Way, 3rd ed. Addison-Wesley Professional 2017

## 4.3. Інформаційні ресурси

1. Дистанційний курс на платформі дистанційного навчання «Сікорський» в середовищі Google Workspace for Education: Технології розробки серверного програмного забезпечення(backend). <https://classroom.google.com/c/NDg5Mzc5ODM4MTc1?cjc=7cmg2mo>
2. Записи лекцій – YouTube. [https://www.youtube.com/playlist?list=PLDBW1r7eLmO96r\\_fzoF3QYI0fRGoDtBpw](https://www.youtube.com/playlist?list=PLDBW1r7eLmO96r_fzoF3QYI0fRGoDtBpw)

## 5. Лабораторні роботи

Мета лабораторних робіт – придбання вмінь та навиків застосування на практиці принципів проектування та розробки серверного програмного забезпечення. Для виконання лабораторних робіт використовуються інструменти для розробки програмного забезпечення (vim, Visual Studio Code, PyCharm, Docker, docker-compose, git).

### Тематика лабораторних робіт:

**Лабораторна робота 1.** Робота з git.

**Лабораторна робота 2.** Налаштування середовища, Розробка базового REST API

**Лабораторна робота 3.** Валідація, обробка помилок, робота з ORM.

**Лабораторна робота 4.** Реалізація аутентифікації, презентація проекту.

## 6. Самостійна робота студента

### Види самостійної роботи (66 години):

- підготовка до аудиторних занять (0,5 годин x 18 лекцій = 9 годин);
- виконання індивідуального завдання до лабораторних робіт, розв'язок задач, оформлення протоколу, розміщення результатів на GitHub (2 години x 8 лабораторних робіт = 16 годин);
- виконання модульної контрольної роботи (2 МКР x 4 = 8 годин);
- підготовка до експрес-тестів (4 години);
- опрацювання тем на самостійну роботу, завантаження та деплой програмного забезпечення для виконання лабораторних робіт (29 годин).

### 6.1. Теми на самостійне опрацювання (денна форма навчання)

#### Розділ 2. Python як мова програмування серверного ПЗ. Теорія

Тема 2.3. Типи в python

Тема 2.5. Поліморфізм

Тема 2.6. Інкапсуляція

#### Розділ 4. Фреймворки, api

Тема 4.2. Робота з JSON

Тема 4.4. Основні принципи роботи з Docker

Тема 4.5. Основні принципи роботи з docker compose

#### Розділ 5. Бази даних

Тема 5.1. Поняття бази даних

Тема 5.2. Основні види баз даних та сфери використання  
Тема 5.8. Реплікація PostgreSQL

## **7. Методика викладання дисципліни на заочній формі навчання**

### **7.1. Зміст лекцій та самостійної роботи**

#### **Лекція 1. Python як мова програмування серверного ПЗ. Теорія**

Тема 2.1. Інтерпретатор python  
Тема 2.2. Представлення об'єктів в пам'яті  
Тема 2.3. Типи в python  
Тема 2.4. ООП в python  
Тема 2.5. Поліморфізм  
Тема 2.6. Інкапсуляція  
Тема 2.7. Наслідування  
Тема 2.8. Магічні методи python

#### **Теми на самостійне опрацювання:**

Тема 1.3. Основні операції в git. Просунуте використання git

#### **Лекція 2. Бази даних**

Тема 5.1. Поняття бази даних  
Тема 5.2. Основні види баз даних та сфери використання  
Тема 5.3. Робота з PostgreSQL  
Тема 5.4. Особливості транзакцій в PostgreSQL  
Тема 5.5. Блокування в PostgreSQL  
Тема 5.6. Особливості join в PostgreSQL

#### **Теми на самостійне опрацювання:**

Тема 5.7. Індокси в PostgreSQL  
Тема 5.8. Реплікація PostgreSQL

#### **Лекція 3. Аутентифікація та авторизація**

Тема 6.1. Поняття аутентифікації  
Тема 6.2. Поняття авторизації  
Тема 6.3. Мультифакторна авторизація  
Тема 6.4. Ролі

#### **Лекція 4. Стандарти та фреймворки аутентифікації**

Тема 7.1. Поняття JWT  
Тема 7.2. Будова JWT  
Тема 7.3. OAuth2

#### **Теми на самостійне опрацювання:**

Тема 8.2. Основні типи вразливостей  
Тема 8.3. Детальний розгляд найпоширеніших вразливостей — CSRF, XSS, SQLi

### **7.2. Тематика лабораторних робіт для самостійного виконання**

**Лабораторна робота 1.** Робота з git.

**Лабораторна робота 2.** Налаштування середовища, Розробка базового REST API

**Лабораторна робота 3.** Валідація, обробка помилок, робота з ORM.

**Лабораторна робота 4.** Реалізація аутентифікації, презентація проекту.

### **7.3. Тематика лабораторних робіт для аудиторного виконання**

### **Аудиторне заняття 1:**

**Лабораторна робота 1.** Створення тестового проекту, робота з основними операціями в git — commit, push, merge. Створення тегів, знайомство з GitHub actions

### **Аудиторне заняття 2:**

Захист лабораторних робіт

### **Аудиторне заняття 3:**

Захист лабораторних робіт

### **Аудиторне заняття 4:**

Захист лабораторної роботи 4 та презентація всього проекту реалізованого в лабораторному курсі

### **Види самостійної роботи для здобувачів заочної форми навчання (104 години):**

- підготовка до аудиторних занять (1,5 години x 4 лекції = 6 годин);
- підготовка до експрес-тестів (4 години);
- самостійне виконання лабораторних робіт 1-5 (3,5 години x 5 лаб. робіт = 17,5 годин);
- підготовка до виконання лабораторних робіт (6 – 8), оформлення протоколів, розміщення результатів на GitHub (1,5 години x 3 лабораторні роботи = 4,5 години)
- виконання модульної контрольної роботи (4 години x 2 МКР = 8 годин);
- підготовка до заліку (4 години);
- самостійне опрацювання теоретичного матеріалу, завантаження та збирання програмного забезпечення для виконання лабораторних робіт (60 годин).

## **● 3. Політика та контроль**

### **8. Політика навчальної дисципліни (освітнього компонента)**

Оцінка, яку студент може отримати за виконання кожної лабораторної роботи та за кожну модульну контрольну роботу наведені в таблиці 1 оцінювання семестрових робіт, розділ 8 силабуса.

Виконання всіх лабораторних робіт є обов'язковими для допуску до семестрового контролю. Умовою допуску до семестрового контролю є зарахування всіх лабораторних робіт та стартовий рейтинг не менше 30 балів.

Для виконання лабораторних робіт встановлюються дедлайни. Виконання лабораторних робіт поза встановлених термінів супроводжується штрафними балами, які вираховуються із оцінки за протокол.

Штрафні бали та жорсткі дедлайни не запроваджуються у період військового стану.

Модульна контрольна робота виконується самостійно за індивідуальним завданням, МКР не приймається поза встановленим терміном. Модульні контрольні роботи не переписуються за умови негативної оцінки, негативна оцінка за МКР (менше ніж 9 балів (<60%)) прирівнюється до 0 балів, в цьому випадку МКР не зараховується.

Окремі теми лекційних занять супроводжуються короткими експрес-тестами (на 20 хвилин), які включають матеріал вивченої теми та питання, які задані на самостійне вивчення. Бали отримані за тест входять в семестрову рейтингову оцінку. Поточні тести не перескладаються.

Таким чином мінімальна оцінка, яку може отримати студент, щоб зарахувати навчальний курс = 60 балів, максимальна – 100 балів за виконання всіх поточних робіт за семестр.

Здобувачі, які виконали всі умови допуску до заліку (виконали всі лабораторні роботи) та мають рейтингову оцінку менше 60 балів, а також ті здобувачі, хто бажає підвищити свою рейтингову оцінку, на останньому за розкладом занятті мають можливість пройти семестровий контроль у вигляді залікової контрольної роботи.

У разі виконання залікової контрольної роботи рейтингова оцінка визначається як сума балів за залікову контрольну роботу та балів за індивідуальні семестрові завдання.

До індивідуальних семестрових завдань вноситься індивідуальна робота студента що стосується виконання лабораторних робіт. Максимальна кількість балів за індивідуальну роботу за семестр дорівнює 60 балів. Максимальна оцінка за залікову контрольну роботу дорівнює 40 балів. Таким чином здобувач має можливість підвищити свою рейтингову оцінку написавши залікову контрольну роботу і додавши додаткові бали до кількості балів, які отримані протягом семестру за індивідуальну семестрову роботу.

Після виконання залікової контрольної роботи, якщо оцінка за залікову контрольну роботу більша ніж за рейтингом, здобувач отримує оцінку за результатами залікової контрольної роботи. Якщо оцінка за залікову контрольну роботу менша ніж за рейтингом, попередній рейтинг здобувача (за винятком балів за семестрове індивідуальне завдання) скасовується і він отримує оцінку з урахуванням результатів залікової контрольної роботи. Цей варіант формує відповідальне ставлення здобувача до прийняття рішення про виконання залікової контрольної роботи, змушує його критично оцінити рівень своєї підготовки та ретельно готуватися до заліку.

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

Розподіл навчального часу за видами занять і завдань з дисципліни згідно з робочим навчальним планом. На кредитний модуль виділено 120 годин та 4 кредити.

Семестровий рейтинг студента з кредитного модуля розраховується, виходячи із 100-бальної шкали. Рейтинг складається з балів, що студент отримує за виконання 8 лабораторних робіт  $R_L$ , двох МКР  $R_{МКР}$  та двох експрес-тестів  $R_{ЕТ}$ .

Максимальна кількість балів за всі лабораторні роботи складає 60 балів, тобто  $R_L = 60$ .

### Критерії оцінювання лабораторних робіт наступні:

- своєчасність підготовки протоколу до лабораторного заняття, повнота виконання теоретичного або практичного завдання в протоколі, вчасне його завантаження: 0 – 2 бали;
- коректність функціонування розробленого програмного забезпечення, демонстрація власного репозиторію на GitHub з матеріалами готової лабораторної роботи: 0 – 4 бали;
- опитування за тематикою лабораторної роботи для зарахування практичної частини роботи, захист одержаних в роботі результатів, відповіді на додаткові теоретичні запитання викладача: 0 – 2 балів.

Максимальна кількість балів за МКР  $R_{МКР} = 2 \times 15 = 30$  балів.

### Оцінка за МКР знижується за:

- некоректне оформлення роботи на віддаленому репозиторії, помилки в файлах збірки;
- відсутність коментарів в програмному кодї та оформленні алгоритмів;
- відсутність коментарів та пояснень під час розрахунків.

Максимальна кількість балів за експрес-тести  $R_{ЕТ} = 2 \times 5 = 10$  балів, тести проводяться у вигляді автоматизованого тестування.

Таблиця 1. Деталізація балів за поточні роботи за семестр

Назва заняття	Форма контролю	Кількість балів	Допуск до заліку автомата м	Всього балів
Лабораторна робота 1	Виконання завдання	10	5	15
	Опитування	3		
	Протокол	2		
Лабораторна робота 2	Виконання завдання	10	5	15

	Опитування	3		
	Протокол	2		
Лабораторна робота 3	Виконання завдання	10	5	15
	Опитування	3		
	Протокол	2		
Лабораторна робота 4	Виконання завдання	10	5	15
	Опитування	3		
	Протокол	2		
Індивідуальна робота	-	-	-	60
Експрес-тести на лекціях	2 x 5	10	5	10
МКР	МКР1	15	9	15
	МКР2	15	9	15
<b>Всього балів</b>		100	60	100

Максимальна кількість балів за залікову роботу дорівнює  $R_3 = 40$  балів.

Заліковий білет містить 4 завдання (одне теоретичне і три практичних) за тематикою лекцій та лабораторних робіт, що виконувались під час семестру. Кожне запитання оцінюється від 0 до 10 балів.

### Критерії оцінювання кожного запитання за чотирма рівнями:

- правильна та змістовна відповідь: 9 – 10 балів;
- правильна відповідь, неповні пояснення: 6 – 8 балів;
- відповідь містить помилки: 3 – 5 балів;
- немає відповіді або відповідь невірна: 0 балів.

Оцінка за залікову роботу розраховується відповідно до таблиці 2. Оцінка незадовільно не дає права на зарахування додаткових балів. Додаткові бали до семестрової оцінки також не нараховуються, якщо оцінка за залікову роботу нижче ніж поточна семестрова оцінка студента (відповідно до таблиці 2). Семестрова оцінка може бути підвищена до мінімальної кількості балів семестрової оцінки (див. табл 3), яка відповідає оцінці, отриманій за залікову роботу (див. табл. 2).

Кількість балів	Оцінка
40-38	Відмінно
37-34	Дуже добре
33-30	Добре
29-26	Задовільно
25-24	Достатньо
Менше 24	Незадовільно

Календарна атестація студентів (на 8 та 14 тижнях семестрів) з дисципліни проводиться за значенням поточного рейтингу студента на час атестації. Якщо значення цього рейтингу не менше 50 % від максимально можливого на час атестації, студент вважається атестованим. В іншому випадку в атестаційній відомості виставляється «не атестовано».

Необхідною умовою допуску до заліку студента є виконання і захист всіх лабораторних робіт з сумою балів не менше ніж 30 балів.

Кількість балів, що отримує студент за семестр визначається формулою

$$RC = R_L + R_{МКР} + R_{ЕТ}$$

Максимальна кількість балів за семестр не перевищує  $RC = 100$ .

З урахуванням одержаної суми балів кінцева оцінка визначається таблицею 3.

Якщо студент пише залікову роботу, кількість балів, що отримує студент за семестр визначається формулою

$$RC = R_L + R_3$$



Максимальна кількість балів за семестр не перевищує  $RC = 100$ .

З урахуванням одержаної суми балів кінцева оцінка визначається таблицею 3.

Таблиця 3. Визначення семестрової оцінки

<i>Кількість балів</i>	<i>Оцінка</i>
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

Робочу програму навчальної дисципліни (силабус):

**Складено**, асистент, Валько Володимир Володимирович.

**Ухвалено** кафедрою обчислювальної техніки (протокол № 10 від 25.05.2022 р.).

**Погоджено** методичною комісією ФІОТ (протокол № 10 від 09.06.2022 р.).