



Технологія розробки Web-додатків

Робоча програма дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Другий (магістерський)
Галузь знань	F Інформаційні технології
Спеціальність	F7 Комп'ютерна інженерія
Освітня програма	ОПП Системне програмування та спеціалізовані комп'ютерні системи
Статус дисципліни	Вибіркова
Форма навчання	очна(денна)
Рік підготовки, семестр	1 курс, весняний семестр
Обсяг дисципліни	4 кредити (120 годин)
Семестровий контроль/ контрольні заходи	МКР, РГР, залік
Розклад занять	http://rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор і лабораторні: доцент кафедри СПіСКС Петрашенко Андрій Васильович, petrashenko@gmail.com
Розміщення курсу	Google Classroom

Програма навчальної дисципліни

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

Навчальна дисципліна "Технологія розробки Web-додатків" спрямована на вивчення технологічних аспектів створення прикладних Web-орієнтованих додатків, а також відповідних мов програмування та прикладних програмних інтерфейсів.

Метою кредитного модуля є формування у студентів здатності створювати різноманітні динамічні Web-орієнтовані додатки, аналізувати функціональні вимоги до Web-орієнтованих системи, а також виконувати автоматизоване тестування Web-додатків, використовуючи сучасні засоби модульного та приймального тестування.

Предмет навчальної дисципліни – процес розроблення та застосування Web-орієнтованих додатків.

Вивчення дисципліни «Технологія розробки Web-додатків» сприяє формуванню у студентів наступних компетентностей.

Компетентності, формуванню яких сприяє дана дисципліна:

ЗК 2 Здатність вчитися і оволодівати сучасними знаннями

ЗК 3 Здатність застосовувати знання у практичних ситуаціях

ЗК 7 Здатність виявляти, ставити та вирішувати проблеми

ЗК 8 Здатність працювати в команді

ФК 2 Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення

ФК 11 Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів

ФК 12 Здатність ідентифікувати, класифікувати та описувати роботу програмнотехнічних засобів, комп'ютерних та кіберфізичних систем, мереж та їхніх компонент шляхом використання аналітичних методів і методів моделювання.

ФК 15 Здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення

ФК 16 Здатність до алгоритмічного та логічного мислення

Формування зазначених компетентностей забезпечує досягнення наступних **програмних результатів навчання**:

ПРН 3 Знати новітні технології в галузі комп'ютерної інженерії.

ПРН 6 Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.

ПРН 7 Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності.

ПРН 8 Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.

ПРН 11 Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії.

ПРН 12 Вміти ефективно працювати як індивідуально, так і у складі команди.

ПРН 14 Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів.

ПРН 16 Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.

ПРН 18 Використовувати інформаційні технології та для ефективного спілкування на професійному та соціальному рівнях.

ПРН 19 Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.

ПРН 20 Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.

ПРН 21 Якісно виконувати роботу та досягати поставленої мети з дотриманням вимог професійної етики.

ПРН 25 Розробляти програмні засоби комп'ютерних систем та мереж з використанням сучасних технологій програмування

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

Для успішного засвоєння дисципліни «Технологія розробки Web-додатків» необхідно і достатньо мати навички алгоритмічного програмування і відповідних мов та інструментальних засобів, засвоїти дисципліни «Web-дизайн» та «Об'єктно-орієнтоване програмування», а також мати базовий рівень володіння англійською мовою не нижче B1.

Успішне оволодіння знаннями з даної дисципліни забезпечує вивчення надалі таких курсів, як «Програмування інтерфейсів користувача» та «Технологія програмування».

3. Зміст навчальної дисципліни

РОЗДІЛ 1. Протокол HTTP та прикладний програмний інтерфейс REST API

Тема 1.1 Вступ до курсу. Протокол HTTP

Тема 1.2. Складові протоколу HTTP. REST API.

РОЗДІЛ 2. Універсальний фреймворк розробки Web-додатків Django.

Тема 2.1. Фреймворк Django. Частина 1. Архітектура та функції.

Тема 2.2. Фреймворк Django. Частина 2. ORM.

Тема 2.3. Фреймворк Django. Частина 3. Мова шаблонів.

РОЗДІЛ 3. Нереляційні СУБД як інструментальні засоби створення Web-додатків.

Тема 3.1. Нереляційні бази даних у Web-технологіях.

Тема 3.2. Застосування нереляційної СУБД redis у типових задачах Web-додатків.

Тема 3.3. Асинхронне виконання задач Web-додатків. Бібліотека Celery.

Тема 3.4. Документні бази даних. MongoDB.

Тема 3.5. Графові бази даних. Neo4J.

РОЗДІЛ 4. Хмарні технології та DevOps у контексті Web-додатків

Тема 4.1. Хмарні технології

Тема 4.2. Хмарні технології AWS. Архітектура.

Тема 4.3. Контейнеризація та мікросервісна архітектура Web-додатків

Тема 4.4. CI/CD для Web-додатків.

4. Навчальні матеріали та ресурси

Базова література

1. Висоцька В.А., Оборська О.В. Python: алгоритмізація та програмування, Новий світ-2000,- 2021, 514с.
2. Зінченко О.В., Іщеряков С.М., Прокопов С.В., Серих С.О., Василенко В.В. Хмарні технології. – Навчальний посібник. – К: ФОП Гуляєва В.М., 2020. – 75с.

Допоміжна література

3. William S. Vincent. *Django for Beginners: Build Websites with Python and Django*. WelcomeToCode – 2020, 292p.
4. Arora, P., Gupta, A. *AWS Certified Solutions Architect Official Study Guide: Associate Exam*. Wiley, 2020, 416 pages.
5. Hightower, K., Burns, B., Beda, J. *Kubernetes: Up and Running: Dive into the Future of Infrastructure*. O'Reilly Media, 2019, 348 pages.

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Навчальний контент дисципліни складається з лекційних та лабораторних занять.

5.1. Лекційні заняття:

№ з/п	Назва теми лекції
1	Вступ. Протокол HTTP. Предмет курсу. Основні поняття. HTTP як основний протокол обміну між Web-браузером і Web-сервером.
2	Складові протоколу HTTP. REST API. Поняття API – прикладного програмного інтерфейсу.
3	Фреймворк Django. Частина 1. Основні підходи до автоматизації рутинних задач розробки Web-додатків.
4	Фреймворк Django. Частина 2. Шаблон MVT, автоматизація обробки Web-форм, об'єктно-орієнтована проєкція
5	Фреймворк Django. Частина 3. Інтерфейс адміністрування Web-додатку, мова шаблонів, подання (контролери).
6	Прикладний програмний інтерфейс. REST API. Бібліотека Django REST Framework: компоненти, методика створення API
7	Web-сокети. Бібліотека Django-Channels. Організація двостороннього обміну між клієнтом і сервером, спільний доступ до функцій Web-додатка.
8	Нереляційні бази даних у Web-технологіях. Проблеми масштабування Web-додатків. Типи нереляційних баз даних та сфери їх застосування.
9	Застосування нереляційної СУБД redis у типових задачах Web-додатків. Шаблон publish-subscribe, кеш веб-додатку, асинхронне виконання задач
10	Асинхронне виконання задач Web-додатків. Бібліотека Celery. Підходи до організації асинхронного виконання задач. Методика використання Celery.
11	Документні бази даних. MongoDB. Можливості MongoDB щодо використання у Web-додатках. Реплікація і шардинг.
12	Графові бази даних. Neo4J. Можливості Neo4J щодо використання у Web-додатках. Мова Cypher як інструмент доступу до сильнозв'язаних даних.

13	Хмарні технології як платформа для Web-додатків. Переваги використання хмарних технологій для розробки та розгортання веб-додатків. Порівняння інструментів для хмарного хостингу та масштабування веб-додатків: Google GCP, Amazon AWS, Microsoft Azure. Практичний приклад розгортання веб-додатку на Amazon AWS.
14	Контейнеризація та мікросервісна архітектура. Основи контейнеризації. Docker для розробки та тестування. Ідея мікросервісної архітектури, переваги та виклики. Використання Docker та Kubernetes для мікросервісів.
15	DevOps та CI/CD для Web-додатків. Основи DevOps. Інструменти CI/CD, такі як Jenkins, GitHub Actions, GitLab CI.

5.2. Лабораторні роботи:

№ з/п	Назва лабораторної роботи	Кількість ауд. годин
1	Розробка серверної частини Web-додатка	8
2	Організація спільної роботи користувачів Web-додатка	8

5.3. Розрахунково-графічна робота (РГР)

РГР передбачає виконання індивідуальних завдань згідно до встановлених варіантів. Метою РГР є поглиблення знань і напрацювання навичок створення програмних додатків, орієнтованих на роботу з нереляційними базами даних. Ваговий бал РГР – 20 балів.

В умовах дистанційного навчання 2025-2026 навчального року усі види занять, у тому числі контрольні заходи, проводяться з використанням сервісу Zoom.

6. Самостійна робота студента

Самостійна робота студентів полягає в наступному:

- підготовці до лекційних занять шляхом вивчення попереднього лекційного матеріалу а також літературних джерел, на яких базується матеріал попередніх лекцій (список джерел і перелік розділів надається разом з лекційним матеріалом);
- підготовці до лабораторних робіт шляхом з ознайомлення із завданням та методичними вказівками до виконання лабораторної роботи, в т. ч. вивченні теоретичного матеріалу, необхідного для відповіді на контрольні питання до лабораторних робіт;
- виконанні завдання лабораторних робіт і оформленні звіту, який висвітлює результати виконання лабораторної роботи (лістинг коду, аналіз результатів і т. і.).

Контроль знання на лабораторних заняттях здійснюється шляхом перевірки звіту з лабораторних робіт та опитування, а також через виконання модульних контрольних робіт.

Термін виконання лабораторних робіт визначається індивідуально для кожної лабораторної роботи і вказується в методичних вказівках.

7. Політика навчальної дисципліни (освітнього компонента)

Система вимог, які ставляться перед студентом:

- студент зобов'язаний відвідувати лекційні та лабораторні заняття й активно працювати над засвоєнням матеріалу, що викладається на них;
- на лекції викладач користується власним презентаційним матеріалом; відпрацьовує практичну частину на віртуальній машині;
- лабораторні роботи захищаються на лабораторному занятті, попередньо оформивши звіт, і надіславши його викладачу;
- модульна контрольна робота пишеться на лекційному занятті із застосуванням всіх доступних матеріалів;
- штрафні бали нараховуються за: невчасну здачу лабораторної роботи. Граничний термін захисту лабораторної роботи без нарахування штрафних балів визначається для кожної лабораторної роботи і надається студенту разом із завданням. Кількість штрафних балів не більше 6.
- **Інклюзивне навчання.** Засвоєння знань та умінь у ході вивчення дисципліни «Технологія розробки Web-додатків» може бути доступним для більшості осіб з особливими освітніми потребами, окрім здобувачів з серйозними вадами зору, які не дозволяють працювати з вебінтерфейсами та виконувати завдання за допомогою персональних комп'ютерів, ноутбуків або інших технічних засобів.
- **Визнання результатів інформальної/неформальної освіти.** У межах дисципліни передбачено можливість зарахування однієї лабораторної роботи на підставі результатів навчання, здобутого в неформальній або інформальній освіті. Зокрема, допускається проходження відповідного онлайн-курсу на освітніх платформах, таких як **Coursera, Udemy, Prometheus** або подібних. Студент має надати сертифікат про успішне завершення курсу, зміст якого відповідає тематиці лабораторної роботи, для розгляду та зарахування викладачем.

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

Види контролю та система оцінювання:

- 2 лабораторні роботи. Кожна лабораторна робота оцінюється наступним чином:
 - бездоганна робота – 20 балів;
 - є певні недоліки у оформленні роботи – 19-16 балів;
 - є певні недоліки у виконанні програми роботи – 15-10 балів;
 - за невчасне виконання і захист лабораторної роботи нараховуються штрафні бали (до 3-х балів);
 - замість будь-якої однієї ЛР дозволяється зарахувати пройдений студентом курс відповідної тематики на платформах Coursera, Udemy тощо (за наявності відповідного сертифіката) – 15-20 балів.
- 1 розрахунково-графічна робота (РГР) оцінюється наступним чином:
 - бездоганна робота – 20 балів;
 - є певні недоліки у оформленні роботи – 19-16 балів;
 - є певні недоліки у виконанні програми роботи – 15-10 балів;
 - за невчасне виконання і захист лабораторної роботи нараховуються штрафні бали (до 2-х балів);

- 4 модульні контрольні роботи (МКР) — 10 балів. Контрольне завдання цієї роботи складається з 10 тестових запитань. За кожну правильну відповідь на тестове запитання нараховується 1 бал. Тривалість кожної МКР – 20 хвилин.
- семестровий контроль — залік.

Система оцінювання

Оцінювання результатів навчання з дисципліни "Технологія розробки Web-додатків" відбувається у вигляді заліку і здійснюється за допомогою РСО першого типу (РСО-1), що передбачає оцінювання результатів навчання за підсумками заходів поточного контролю впродовж семестру або за результатами семестрового контролю.

При цьому розподіл балів здійснюється наступним чином:

Вид роботи	Кількість	Бали за одиницю	Всього балів
Лабораторна робота №1 (ЛР1)	1	20	20
Лабораторна робота №2 (ЛР2)	1	20	20
Розрахунково-графічна робота (РГР)	1	20	20
Модульна контрольна робота	4	10	40
Разом	—	—	100

Умови допуску до заліку

1. Обов'язкове виконання індивідуальних завдань:
 - **ЛР1, ЛР2, РГР** (максимум 60 балів).
 - Мінімальний рівень виконання кожного завдання – **не менше 60%**:
 - ЛР1 ≥ 12 балів
 - ЛР2 ≥ 12 балів
 - РГР ≥ 12 балів
 - Мінімальна сума для допуску: **36 балів**.
2. Модульні контрольні роботи рекомендуються, але **не є обов'язковими** для допуску.

Семестровий контроль (залік)

- Студент отримує **залік автоматично**, якщо виконав умови допуску та набрав сумарно **60 і більше балів**.
- Якщо студент набрав менше 60 балів (але ≥ 36 балів), або прагне підвищити підсумкову оцінку, він складає залік у вигляді підсумкового тесту.
- Форма контролю: **комплексний тест** (32 запитання по 2 бали кожне = 64 бали).
- Остаточна оцінка = **бали за обов'язкові роботи (до 36) + бали за залік (до 64)**.
- Використовується «м'який» підхід обчислення підсумкової оцінки: вона визначається як кращий результат між накопиченими балами та комбінацією «обов'язкові бали + залік».

Календарний контроль (атестація): провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу. Умовою першої атестації є отримання не менше 15 балів (на час атестації). Умовою другої атестації – отримання не менше 30 балів (на час атестації).

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

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

Робочу програму навчальної дисципліни (силабус):

Складено доцентом кафедри СПіСКС, к.т.н., Петрашенком А.В.

Ухвалено кафедрою СПСКС (протокол № 3 від 14.10.2025р.)

Погоджено Методичною комісією факультету ПМ (протокол № 2 від 16.10.2025р.)