

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Волинський національний університет імені Лесі Українки
Факультет інформаційних технологій і математики
Кафедра комп'ютерних наук та кібербезпеки

СИЛАБУС
нормативного освітнього компонента
ТЕХНОЛОГІЇ ПРОЄКТУВАННЯ ТА РОЗРОБКИ МЕРЕЖЕВИХ
РЕСУРСІВ
підготовки другого (магістерського) рівня вищої освіти
спеціальності F3 Комп'ютерні науки
освітньо-професійної програми Комп'ютерні науки та інформаційні
технології

Силабус нормативного освітнього компонента «Технології проектування та розробки мережевих ресурсів» підготовки магістрів, галузі знань F Інформаційні технології, спеціальності F3Комп'ютерні науки, за освітньою програмою Комп'ютерні науки та інформаційні технології.

Розробник: Ройко Олександр Юрійович, кандидат технічних наук, старший викладач кафедри комп'ютерних наук та кібербезпеки

Погоджено

Гарант освітньо-професійної програми:



Булатецький В.В.

Силабус освітнього компонента затверджено на засіданні кафедри комп'ютерних наук та кібербезпеки
протокол № 2 від 17.09.2025 р.



Завідувач кафедри:

Гришанович Т. О.

I. Опис освітнього компонента

Найменування показників	Галузь знань, спеціальність, освітньо-професійна програма, освітній рівень	Характеристика освітнього компонента
Денна форма навчання	F Інформаційні технології F3 Комп'ютерні науки Комп'ютерні науки та інформаційні технології магістр	Нормативна
Кількість годин/кредитів 120 / 4		Рік навчання 2
		Семестр 3-ій
		Лекції 14 год.
		Лабораторні 20 год.
ІНДЗ: є		Самостійна робота 78 год.
		Консультації 8 год.
		Форма контролю: екзамен
Мова навчання: українська		

II Інформація про викладача

ППП: Ройко Олександр Юрійович

Науковий ступінь: кандидат технічних наук

Посада: старший викладач кафедри комп'ютерних наук та кібербезпеки

Контактна інформація: Roiko.Oleksandr@vnu.edu.ua

Дні занять: <https://ps.vnu.edu.ua/cgi-bin/timetable.cgi>

III Опис освітнього компонента

- Анотація освітнього компонента.** Силабус освітнього компонента «Технології проектування та розробки мережевих ресурсів» складено відповідно до освітньо-професійної програми «Комп'ютерні науки та інформаційні технології» другого рівня вищої освіти, галузі знань F Інформаційні технології, за спеціальністю F3 Комп'ютерні науки. Освітній компонент «Технології проектування та розробки мережевих ресурсів» належить до переліку нормативних освітніх компонент. Освітній компонент охоплює сучасні технології та інструменти, необхідні для успішного створення мережевих ресурсів. Здобувачі матимуть можливість працювати над спільними проектами та виконувати практичні завдання, що дозволить їм засвоїти теоретичний матеріал та отримати практичний досвід.
- Пререквізити:** Методологія розробки алгоритмів; Проектування та супровід баз даних та знань; Адміністрування комп'ютерних систем та мереж.
Постреквізити: Архітектура та реінжиніринг інформаційних систем; Переддипломна практика; Кваліфікаційна робота.
- Мета і завдання освітнього компонента:** надання здобувачам освіти знань, навичок та практичного досвіду у галузі проектування та розробки сучасних мережевих застосунків. Вивчення освітнього компонента спрямовано на розширення розуміння процесу розробки вебдодатків, оволодіння архітектурними патернами, мережевими технологіями та інструментами для створення production-ready рішень.
- Компетентності. Програмні результати навчання.**
ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

- ЗК02. Здатність застосовувати знання у практичних ситуаціях.
- ЗК03. Здатність спілкуватися державною мовою як усно, так і письмово.
- ЗК05. Здатність вчитися й оволодівати сучасними знаннями.
- ЗК06. Здатність бути критичним і самокритичним.
- ЗК07. Здатність генерувати нові ідеї (креативність).
- СК01. Усвідомлення теоретичних засад комп'ютерних наук.
- СК02. Здатність формалізувати предметну область певного проєкту у вигляді відповідної інформаційної моделі.
- СК04. Здатність збирати і аналізувати дані (включно з великими), для забезпечення якості прийняття проєктних рішень.
- СК05. Здатність розробляти, описувати, аналізувати та оптимізувати архітектурні рішення інформаційних та комп'ютерних систем різного призначення.
- СК10. Здатність оцінювати та забезпечувати якість ІТ-проєктів, інформаційних та комп'ютерних систем різного призначення, застосовувати міжнародні стандарти оцінки якості програмного забезпечення інформаційних та комп'ютерних систем, моделі оцінки зрілості процесів розробки інформаційних та комп'ютерних систем.
- СК11. Здатність ініціювати, планувати та реалізовувати процеси розробки інформаційних та комп'ютерних систем та програмного забезпечення, включно з його розробкою, аналізом, тестуванням, системною інтеграцією, впровадженням і супроводом
- РН5. Оцінювати результати діяльності команд та колективів у сфері інформаційних технологій, забезпечувати ефективність їх діяльності.
- РН6. Розробляти концептуальну модель інформаційної або комп'ютерної системи.
- РН9. Розробляти алгоритмічне та програмне забезпечення для аналізу даних (включно з великими).
- РН10. Проєктувати архітектурні рішення інформаційних та комп'ютерних систем різного призначення.
- РН15. Виявляти потреби потенційних замовників щодо автоматизації обробки інформації.
- РН17. Виявляти та усувати проблемні ситуації в процесі експлуатації програмного забезпечення, формулювати завдання для його модифікації або реінжинірингу.

3. Структура освітнього компонента

Назви змістових модулів і тем	Кількість годин					Форма контролю / бали
	Усього	В тому числі				
		Лек.	Лаб.	Сам. роб.	Конс.	
Змістовий модуль 1. Розробка вебзастосунків за допомогою FastAPI						
Тема 1. Архітектура сучасних вебзастосунків. Проєктування REST API	13	2	2	8	1	Звіт. по лаб. роботі / 7
Тема 2. Python для веброзробки та основи FastAPI. Реалізація FastAPI проєкту	17	2	2	12	1	Звіт. по лаб. роботі / 7
Тема 3. Робота з базами даних у FastAPI. Інтеграція бази даних	9	2	2	4	1	Звіт. по лаб. роботі / 7

Назви змістових модулів і тем	Кількість годин					Форма контролю / бали
	Усього	В тому числі				
		Лек.	Лаб.	Сам. роб.	Конс.	
Тема 4. Архітектурні патерни та Clean Architecture. Service Layer та бізнес-логіка. Repository Pattern та тестування. Контейнеризація проєкту в Docker	18	2	6	8	2	Звіт. по лаб. роботі / 21
Разом за змістовим модулем 1	57	8	12	32	5	42
Змістовий модуль 2. Мережеві технології та розподілені системи						
Тема 5. WebSocket та real-time комунікація. Реалізація WebSocket застосунку. WebSocket Chat з кімнатами	13	2	4	6	1	Звіт. по лаб. роботі / 14
Тема 6. HTTP клієнти та інтеграція з зовнішніми API. Інтеграція зовнішніх API та Webhooks	17	2	2	12	1	Звіт. по лаб. роботі / 7
Тема 7. Message Queues, Celery та мікросервіси. Реалізація фонових задач	13	2	2	8	1	Звіт. по лаб. роботі / 7
Разом за змістовим модулем 2	43	6	8	26	3	28
Види підсумкових робіт						Бали
Модульне тестування 1						10
Модульне тестування 2						10
ІНДЗ	20			20		10
Всього годин / Балів	120	14	20	78	8	100

Завдання для самостійного опрацювання

№ з/п	Тема	Кількість годин
1	Підготовка до лабораторних робіт	20
2	Опрацювання лекційного матеріалу	14
3	Виконання ІНДЗ	20
4	JWT автентифікація та авторизація	4
5	Структуроване журналювання та моніторинг	4
6	Версіювання та документування API	4
7	GraphQL з FastAPI через Strawberry	4
8	Обмеження частоти запитів та регулювання навантаження	4
9	Моніторинг з Prometheus/Grafana	4
	Разом	78

IV Політика оцінювання

Політика викладача щодо здобувача освіти

Усі учасники освітнього процесу повинні дотримуватись вимог чинного законодавства України, Статуту і Правил внутрішнього розпорядку ВНУ імені Лесі

Українки, загальноприйнятих моральних принципів, правил поведінки та корпоративної культури; підтримувати атмосферу доброзичливості, відповідальності, порядності й толерантності. Атмосфера на заняттях повинна бути творчою, відкритою до конструктивної критики. Недопустимі запізнення на заняття; користування мобільним телефоном, планшетом чи іншими мобільними пристроями під час заняття; списування. Очікується, що всі здобувачі освіти відвідають усі лекції і лабораторні заняття курсу. Кожен здобувач освіти повинен бути учасником дистанційного курсу «Технології проектування та розробки мережеских ресурсів» (<https://moodle-cs.vnu.edu.ua/course/view.php?id=267>). Завдання для практичного виконання (лабораторні роботи, ІНДЗ, самостійні роботи), завдання підсумкового контролю (тести, контрольні роботи, що передбачають розробку програм) здаються із використанням засобів дистанційного курсу.

Політика щодо академічної доброчесності

Під час навчання учасники освітнього процесу зобов'язані дотримуватися академічної доброчесності: етичних принципів та визначених законом правил, якими мають керуватися учасники освітнього процесу під час навчання, викладання та провадження наукової діяльності.

Дотримання академічної доброчесності здобувачами передбачає: самостійне виконання навчальних завдань, завдань поточного та підсумкового контролю (для осіб з особливим освітніми потребами ця вимога застосовується з урахуванням їх індивідуальних потреб і можливостей); посилення на джерела інформації у разі використання ідей, тверджень, відомостей; дотримання норм законодавства про авторське право; надання достовірної інформації про результати власної навчальної (наукової, творчої) діяльності.

Під час оцінювання результатів навчання здобувачі освіти не користуються забороненими засобами (мобільний телефон, планшет, конспект, навчальна література, інші джерела інформації, в тому числі Інтернет-ресурси), самостійно виконують запропоновані завдання. При виконанні лабораторних робіт з курсу здобувачі мають право використовувати власні ноутбуки, якщо вони підтримують необхідне програмне забезпечення.

Комунікаційна політика

Здобувачі вищої освіти повинні мати активовану університетську корпоративну пошту. Усі письмові запитання до викладачів стосовно курсу мають надсилатися на корпоративну електронну пошту, можливе інше (додаткове) джерело комунікації, визначене викладачем для більш оперативного зв'язку зі здобувачами.

Політика щодо дедлайнів та перескладання

Перескладання контрольних робіт та тестувань, відбувається із дозволу лектора і тільки за наявності поважних причин (наприклад, лікарняний).

Політика щодо оскарження оцінки. Якщо здобувач вищої освіти не згоден з оцінюванням його знань він може опротестувати виставлену викладачем оцінку у встановленому порядку згідно «Положення про порядок і процедури вирішення конфліктних ситуацій у Волинському національному університеті імені Лесі Українки»

Політика щодо відвідування занять

Для здобувачів вищої освіти денної форми відвідування занять є обов'язковим. Поважними причинами для неявки на заняття є хвороба, академічна мобільність, які необхідно підтверджувати відповідними документами. Про відсутність на занятті та причини відсутності здобувач вищої освіти має повідомити викладача або особисто, або через старосту. За об'єктивних причин навчання може відбуватись в он-лайн формі за погодженням з керівником курсу та деканом факультету. Посилання на дистанційний курс: <https://moodle-cs.vnu.edu.ua/course/view.php?id=267>.

Поєднання навчання та досліджень. Здобувачі вищої освіти мають можливість додатково отримати бали за виконання індивідуальних завдань дослідницького

характеру, зокрема, написання та опублікування наукових тез та статей з тематики дисципліни. За рішенням кафедри здобувачам, які брали участь у роботі конференцій, підготовці наукових публікацій, конкурсах студентських наукових робіт за тематикою ОК й досягли значних результатів, може бути присуджено додаткові (бонусні) бали, які зараховуються як результати поточного контролю. Систему бонусних балів погоджує науково-методична комісія факультету інформаційних технологій і математики. При цьому загальна кількість балів, що вноситься до відомості за поточну роботу.

Можливість визнання результатів навчання, отриманих у формальній, неформальній та інформальній освіті

Під час вивчення освітнього компонента можливе визнання результатів навчання отриманих у формальній, неформальній та/або інформальній освіті. Порядок визнання результатів навчання для здобувачів вищої освіти, набутих у: формальній освіті (академічна мобільність студентів на території України чи поза її межами, для студентів, які переводяться, поновлюються з інших ЗВО (вітчизняних чи іноземних); неформальній та/або інформальній освіті здійснюється згідно «Положення про визнання результатів навчання, отриманих у формальній, неформальній та/або інформальній освіті у Волинському національному університеті імені Лесі Українки».

V Підсумковий контроль

Оцінювання здійснюється за 100-бальною шкалою. Оцінка включає в себе поточний контроль (нараховується за якісне виконання лабораторних робіт) та підсумковий модульний контроль (нараховується за виконання модульних контрольних робіт та модульних тестових робіт, до лекційних матеріалів курсу). Максимальна кількість балів, яку може отримати здобувач під час поточного оцінювання за семестр – 70 балів. Підсумковий модульний контроль за семестр включає в себе оцінки за всі модульні контрольні роботи (тестові завдання), ІНДЗ і складає 30 балів.

Якщо за результатами семестру накопичено не менше 75 балів і здобувач погоджується із цим результатом, то оцінка за семестр може виставлятися без складання іспиту. В іншому випадку здобувач складає іспит; максимальна кількість балів, яку можна отримати на іспиті – 30 балів при цьому бали за підсумковий модульний контроль анулюються.

Екзамен проходить в усній формі. Оцінка за семестр у випадку складання іспиту є сумою балів поточного контролю та балів, отриманих під час іспиту.

Питання, які виносяться на екзамен

1. Опишіть будову HTTP запиту та відповіді, вкажіть основні складові.
2. Поясніть призначення та семантику HTTP методів GET, POST, PUT, PATCH, DELETE.
3. Охарактеризуйте класифікацію кодів стану HTTP та призначення основних діапазонів (2xx, 3xx, 4xx, 5xx).
4. Сформулюйте основні принципи та обмеження архітектурного стилю REST.
5. Поясніть принципи ресурсно-орієнтованого проектування REST API.
6. Порівняйте стратегії версіонування API (через URI, заголовки, узгодження вмісту).
7. Обґрунтуйте необхідність стандартизації відповідей про помилки в API.
8. Розкрийте призначення об'єктів передачі даних (DTO) та опишіть їх використання.
9. Поясніть концепцію впровадження залежностей (Dependency Injection) та наведіть переваги.
10. Опишіть синтаксис анотацій типів у Python (type hints) та їх інтеграцію з IDE.
11. Сформулюйте основні концепції асинхронного програмування у Python.
12. Поясніть призначення ключових слів async та await, наведіть правила використання.
13. Визначте поняття співпрограм (coroutines) у Python та опишіть їх виконання.

14. Поясніть принцип роботи циклу подій (event loop) та керування асинхронними задачами.
15. Порівняйте фреймворки FastAPI, Flask та Django, вкажіть особливості кожного.
16. Опишіть механізм автоматичної перевірки даних у FastAPI та наведіть переваги.
17. Поясніть створення моделей Pydantic для перевірки та серіалізації даних.
18. Продемонструйте оголошення та перевірку параметрів шляху (path parameters) у FastAPI.
19. Опишіть роботу з параметрами запиту (query parameters) у FastAPI, включаючи необов'язкові параметри.
20. Поясніть роботу з тілом запиту (request body) у FastAPI через Pydantic моделі.
21. Порівняйте SQLAlchemy Core та ORM, наведіть сценарії використання кожного підходу.
22. Опишіть роботу з асинхронним SQLAlchemy через AsyncSession та AsyncEngine.
23. Поясніть визначення моделей через декларативну базу у SQLAlchemy.
24. Охарактеризуйте призначення Alembic та опишіть основні команди для роботи з міграціями.
25. Оцініть можливості та обмеження автоматичного створення міграцій в Alembic.
26. Розкрийте призначення шаблону Сховище (Repository pattern) та опишіть структуру реалізації.
27. Поясніть процес керування сесіями бази даних: створення, використання та закриття.
28. Опишіть концепцію пулу з'єднань (connection pooling) та його налаштування в SQLAlchemy.
29. Поясніть роботу з транзакціями в SQLAlchemy: фіксація, відкат, точка збереження.
30. Розшифруйте властивості ACID транзакцій та поясніть значення кожної властивості.
31. Сформулюйте основну ідею та принципи Чистої архітектури (Clean Architecture).
32. Опишіть багатошарову архітектуру (Layered Architecture) та призначення кожного шару.
33. Визначте відповідальність шару представлення (presentation layer) та його компоненти.
34. Поясніть відповідальність шару бізнес-логіки та особливості реалізації.
35. Розкрийте призначення доменної моделі та її відмінність від моделі бази даних.
36. Опишіть структуру та відповідальність шаблону Службовий шар (Service layer).
37. Поясніть призначення шаблону Одиниця роботи (Unit of Work) та використання зі Сховищем.
38. Розшифруйте принципи SOLID та надайте пояснення кожного принципу.
39. Сформулюйте принцип єдиної відповідальності (Single Responsibility) та наведіть приклади.
40. Продемонструйте механізм впровадження залежностей у FastAPI через Depends().
41. Опишіть структуру протоколу WebSocket, процес встановлення з'єднання та кадри передачі даних.
42. Порівняйте протоколи HTTP та WebSocket, наведіть сценарії використання кожного.
43. Поясніть концепцію повнодуплексної комунікації та переваги WebSocket.
44. Продемонструйте створення WebSocket кінцевих точок у FastAPI та обробку з'єднань.
45. Опишіть призначення та реалізацію шаблону Менеджер з'єднань (ConnectionManager).
46. Поясніть реалізацію розсилання повідомлень (broadcasting) через Менеджер з'єднань.
47. Порівняйте події від сервера (SSE) з WebSocket та наведіть сценарії використання.
48. Обґрунтуйте необхідність підтримки активності з'єднання (heartbeat) у WebSocket.
49. Порівняйте бібліотеки httpx та requests, вкажіть переваги httpx.
50. Продемонструйте використання httpx.AsyncClient для асинхронних HTTP запитів.

51. Опишіть типи часу очікування (timeout) та їх налаштування при HTTP запитих.
52. Поясніть стратегії логіки повторних спроб (retry logic) при обробці помилок.
53. Опишіть алгоритм експоненційної затримки (exponential backoff) та його застосування.
54. Охарактеризуйте методи автентифікації у HTTP клієнтах: токени Bearer, ключі API, OAuth2.
55. Поясніть концепцію вебхуків (webhooks) та їх відмінність від опитування (polling).
56. Обґрунтуйте важливість перевірки підпису вебхука для безпеки та опишіть методи.
57. Розкрийте призначення шаблону Автоматичний вимикач (Circuit Breaker) та механізм роботи.
58. Опишіть реалізацію обмеження швидкості запитів (rate limiting) до API.
59. Охарактеризуйте шаблони черг повідомлень: черга, публікація-підписка, теми.
60. Оцініть можливості та обмеження фонових задач у FastAPI.
61. Опишіть використання Redis як брокера повідомлень та структури даних для обміну.
62. Поясніть механізм публікації-підписки у Redis (Pub/Sub) та його обмеження.
63. Порівняйте потоки Redis (Streams) з Pub/Sub та наведіть переваги.
64. Опишіть архітектуру Celery та охарактеризуйте основні компоненти.
65. Поясніть призначення робітників Celery (workers) та стратегії масштабування.
66. Опишіть планування періодичних задач через планувальник Celery (beat).
67. Оцініть переваги та виклики розподіленої обробки задач.
68. Розкрийте призначення черги недоставлених повідомлень (dead letter queue) та реалізацію.
69. Сформулюйте основні принципи та характеристики мікросервісної архітектури.
70. Порівняйте синхронний та асинхронний підходи до комунікації між сервісами.
71. Опишіть роль та відповідальність шаблону Шлюз API (API Gateway).
72. Поясніть механізми та інструменти виявлення сервісів (service discovery).
73. Охарактеризуйте типи перевірок стану (health checks) та їх реалізацію в мікросервісах.
74. Порівняйте підходи хореографії та оркестрації у шаблоні Saga (Saga pattern).
75. Поясніть концепцію подієво-орієнтованої архітектури та наведіть переваги.
76. Опишіть основні концепції Docker: образи, контейнери, томи.

Екзаменаційні білети складаються з комплексних завдань двох типів:

1. Тестові завдання, 20 запитань, всього 10 балів (тестові завдання охоплюють всі теми змістових модулів).
2. Два теоретичні запитання по 10 балів за повну відповідь (теоретичні запитання охоплюють всі теми, які виносяться на екзамен).

VI Шкала оцінювання (національна та ECTS)

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою
90-100	A	Відмінно
82-89	B	Добре
75-81	C	
67-74	D	Задовільно
60-66	E	
1-59	Fx	Незадовільно

VII Рекомендована література та інтернет-ресурси

1. Технології проектування та розробки мережевих ресурсів: електронний курс навчальної дисципліни. Луцьк: ВНУ ім. Лесі Українки, 2025. URL: <https://moodle-cs.vnu.edu.ua/course/view.php?id=267>.
2. FastAPI Documentation. Official documentation. URL: <https://fastapi.tiangolo.com/> (дата звернення: 21.08.2025)
3. SQLAlchemy 2.0 Documentation. Official documentation. URL: <https://docs.sqlalchemy.org/en/20/> (дата звернення: 20.08.2025)
4. Pydantic Documentation. Data validation using Python type hints. URL: <https://docs.pydantic.dev/> (дата звернення: 19.08.2025)
5. Ramalho L. Fluent Python: Clear, Concise, and Effective Programming. 2nd Edition. O'Reilly Media, 2022. 1014 p.
6. Docker Documentation. Containerization platform. URL: <https://docs.docker.com/> (дата звернення: 05.08.2025)
7. Redis Documentation. In-memory data structure store. URL: <https://redis.io/docs/> (дата звернення: 20.08.2025)
8. Celery Documentation. Distributed task queue. URL: <https://docs.celeryq.dev/> (дата звернення: 20.08.2025)
9. Чакон С., Страуб Б. Pro Git. 2024. 519 с. URL: <https://github.com/progit/progit2-uk/releases/download/2.1.12/progit.pdf> (дата звернення: 20.08.2025)
10. GitHub Documentation. Version control and collaboration. URL: <https://docs.github.com/> (дата звернення: 20.08.2025)