



Національний
транспортний
університет

ВК Сучасні технології розробки мережевих застосувань

Рівень вищої освіти – *другий (магістерський)*

Дні занять, час занять, аудиторія: будуть доступні згідно розкладу за посиланням <https://kizt.ntu.edu.ua/to-the-student/rozklady-zanyat-i-sesij/>

Кафедра інформаційних технологій та штучного інтелекту

Лекції проводить

к.ф.-м.н., доц. Ткаченко Ольга Іванівна

Контактна інформація	oitkachen@gmail.com +38063-282-43-60
----------------------	--

Адреса, номер аудиторії	м. Київ, вул. Івана Огієнка, 19, каб. 601a
-------------------------	--

Час консультацій	За попередньою домовленістю з викладачем
------------------	--

Практичні заняття проводить

к.ф.-м.н., доц. Ткаченко Ольга Іванівна

Контактна інформація	oitkachen@gmail.com +38063-282-43-60
----------------------	--

Аудиторія	м. Київ, вул. Івана Огієнка, 19, каб. 601a
-----------	--

Час консультацій	За попередньою домовленістю з викладачем
------------------	--

Анотація дисципліни: «Сучасні технології розробки мережевих застосувань» - вибіркова навчальна дисципліна про створення прикладних програм, що взаємодіють через мережу. Курс охоплює HTTP, клієнт-серверну взаємодію, серверну розробку мовою PHP, проєктування прикладних API, валідацію даних, обробку помилок, тестування та оцінювання надійності мережевих застосувань.

Предметом вивчення навчальної дисципліни є принципи та інструменти розробки прикладних мережевих застосувань: взаємодія клієнта і сервера за протоколом HTTP, серверна обробка запитів, прикладні програмні інтерфейси, перевірка даних, тестування, журналювання та оцінювання якості. Обсяг дисципліни - 4 кредити ЄКТС (120 годин): денна форма - 18 годин лекцій, 18 годин практичних занять і 84 години самостійної роботи; заочна форма - 2 години лекцій, 4 години практичних занять і 114 годин самостійної роботи.

Мета дисципліни: сформувати здатність проєктувати, реалізовувати, тестувати та документувати прикладні мережеві застосування з використанням сучасних протоколів, серверних технологій і програмних інтерфейсів.

Очікувані результати навчання: після завершення дисципліни здобувач освіти зможе: 1) пояснювати взаємодію клієнта і сервера та життєвий цикл HTTP-запиту; 2) розробляти серверні обробники й прикладні API мовою PHP; 3) перевіряти вхідні дані, формувати коректні відповіді та обробляти помилки; 4) застосовувати принципи маршрутизації, автентифікації та керування станом у межах прикладного застосування; 5) писати модульні та інтеграційні тести, використовувати

журналювання й відтворювати дефекти; 6) оцінювати продуктивність, надійність і супроводжуваність мережевого застосування та обґрунтовувати технічні рішення.

Міждисциплінарні зв'язки: дисципліна пов'язана з навчальними дисциплінами «Методологія та організація наукових досліджень», «Моделі управління IT-інфраструктурою», «Проектування програмного забезпечення інтелектуальних систем», «Проектування систем з розподіленими базами даних», «Хмарні технології», «Інформаційна безпека та гібридні загрози» та «Доменна інженерія».

Програма навчальної дисципліни складається з таких модулів:

Модуль 1.

Тема 1. Архітектура мережевих застосувань і протокол HTTP

Моделі клієнт–сервер і багаторівневої прикладної системи. URI, методи HTTP, заголовки, коди стану, представлення ресурсів і принципи ідемпотентності. Життєвий цикл запиту та відповіді, маршрутизація, кешування на рівні протоколу. Аналіз обміну повідомленнями засобами браузера та мережевого журналу.

Тема 2. Серверна розробка та прикладні програмні інтерфейси

Структура серверного застосування мовою PHP, обробники запитів і маршрути. Проектування ресурсів і операцій API, параметри запитів, формати JSON і HTML, версіювання інтерфейсу. Формування відповідей, обробка винятків і єдиний формат помилок. Документування API за допомогою OpenAPI без прив'язки до конкретної інфраструктури розгортання.

Тема 3. Обробка даних, форми та керування станом

Передавання даних формами й API, кодування, серіалізація та десеріалізація. Валідація типів, діапазонів і бізнес-правил на межі застосування. Сесії, cookies і токени як механізми прикладного керування станом. Захист від некоректного введення розглядається на рівні програмної обробки даних, без поглиблення в засоби інформаційної безпеки.

Модуль 2.

Тема 4. Доступ до даних через прикладний шар

Принципи розділення прикладної логіки та доступу до даних. Параметризовані запити, транзакційна межа прикладної операції та обробка помилок взаємодії з джерелом даних. Побудова репозиторію або сервісного шару, перетворення даних у DTO та перевірка узгодженості відповідей API. Проектування схем, реплікація й розподілене зберігання баз даних до змісту теми не належать.

Тема 5. Тестування, спостережуваність і якість мережевих застосувань

Модульне, інтеграційне та контрактне тестування прикладних API. Тестові дані, ізоляція залежностей, відтворення помилок і перевірка граничних випадків. Журналювання подій, кореляційний ідентифікатор запиту, базові метрики часу відповіді та частоти помилок. Профілювання, контроль продуктивності та підготовка технічного звіту про якість і супроводжуваність застосування.

Для розвитку дослідницьких і творчих навичок здобувачам освіти можуть пропонуватися додаткові завдання: тестування API, відтворення дефекту, порівняння форматів відповідей або профілювання прикладного запиту. Результати таких робіт зараховуються в межах установлених балів поточного контролю.

Тематика індивідуальної роботи погоджується з викладачем і має містити опис мережевого сценарію, програмної реалізації, тестів, вимірних показників якості та висновків.

Методи поточного контролю: виконання та захист практичних робіт, тематичні тести, перевірка API за сценаріями, аналіз журналів і метрик, усне обґрунтування технічних рішень.

Методи модульного контролю: для денної форми - індивідуальне завдання на 20 балів; для заочної форми - контрольна робота на 20 балів. Підсумковий контроль - 40 балів; форма визначається після встановлення коду вибіркового компонента.

Джерела для вивчення дисципліни:

1. Методичні матеріали, приклади програм і завдання до практичних робіт, розміщені у віртуальному середовищі навчання.

2. Електронний ресурс бібліотеки НТУ: <http://lib.ntu.edu.ua/catalog/login.html>

Віртуальне середовище навчання (Google Classroom): посилання на курс надається викладачем на першому занятті.

Оцінювання

Контроль протягом семестру (60 балів)					Підсумковий контроль	Сума балів
Модуль 1		Модуль 2		Модуль 3 (ІЗ)		
Тема 1	Тема 2	Тема 3	Тема 4-5			
Для денної форми: за кожною з тем 1–5 - до 5 балів за виконання і захист практичної роботи та до 3 балів за тематичний тест або усне обґрунтування. Разом за теми - 40 балів.					40	100
Для заочної форми: за кожною з тем 1–5 - до 8 балів за виконання і захист практичного завдання. Разом за теми - 40 балів.						

Самостійна робота

Самостійна робота охоплює опрацювання документації, підготовку середовища, завершення практичних експериментів, тестування API, аналіз журналів і оформлення звітів. Додаткові види робіт оцінюються лише в межах 60 балів поточного і модульного контролю.

Підготовка до практичних занять і контрольних заходів здійснюється під час самостійної роботи з можливістю консультування з викладачем у визначений час або засобами електронного зв'язку.

Результати неформальної або інформальної освіти можуть бути визнані замість відповідних завдань за процедурою НТУ в межах 60 балів і не збільшують максимальну оцінку понад 100 балів.

Критерії оцінювання

Критерії оцінювання: коректність програмної реалізації; повнота тестів; правильність обробки помилок і даних; відтворюваність сценаріїв; якість журналювання, вимірювань та технічного звіту. Підсумкові бали визначаються як сума балів, отриманих протягом семестру та за підсумковий контроль, відповідно до Додатка 1 до Положення про організацію освітнього процесу в НТУ.

Політика несвоєчасного проходження контрольних заходів.

Поточний та підсумковий контролю проводяться згідно з графіком освітнього процесу та затвердженими графіками. У разі неявки на підсумковий контроль з поважної причини контроль

може бути проведений індивідуально в погоджений строк за наявності дозволу дирекції ННКІЗТ.

Повторне складання підсумкового контролю в разі отримання незадовільної оцінки допускається не більше двох разів: один раз - викладачеві, другий - комісії, створеній директором ННКІЗТ відповідно до чинних нормативних документів НТУ.

У разі подання завдання пізніше встановленого строку без поважної причини оцінка знижується на 10 %. Документовані збої електронного середовища навчання, аварійні відключення або інші обставини, що об'єктивно унеможливили своєчасне подання, розглядаються викладачем індивідуально.

Політика перескладання. Упродовж тижня після оголошення результатів поточного контролю здобувач освіти може звернутися до викладача за роз'ясненням або з обґрунтованою незгодою. Апеляція щодо підсумкового контролю подається в порядку та строки, визначені чинними нормативними документами НТУ. Повторне проходження підсумкового контролю з метою підвищення позитивної оцінки не допускається.

Політика відвідування та активності. Здобувач освіти має брати участь у заняттях і виконувати передбачені дисципліною завдання. Пропущені практичні роботи та контрольні заходи виконуються за погодженням індивідуальним графіком до завершення підсумкового контролю.

Академічна доброчесність. Дотримання академічної доброчесності регулюється Кодексом етики академічних взаємовідносин та доброчесності НТУ, Положенням про систему забезпечення академічної доброчесності та іншими чинними нормативними документами університету.

Кодекс етики академічних взаємовідносин та доброчесності Національного транспортного університету

Положення про систему забезпечення академічної доброчесності педагогічними, науково-педагогічними, науковими працівниками та здобувачами вищої освіти в Національному транспортному університеті

Порушеннями академічної доброчесності є:

- академічний плагіат;
- фальсифікація;
- списування;
- обман;
- неправомірна вигода;
- хабарництво.

Порушення Кодексу етики академічних взаємовідносин та доброчесності Національного транспортного університету є серйозним порушенням, навіть якщо воно є ненавмисним.

Під час поточного або підсумкового контролю дозволено використовувати лише ті джерела й засоби, які прямо визначив викладач. Ознаки списування, недозволеної допомоги або підміни авторства розглядаються за процедурами академічної доброчесності НТУ.

Поведінка в аудиторії. Ноутбуки та портативні пристрої використовуються з навчальною метою відповідно до завдань дисципліни. Не допускається використання пристроїв, яке перешкоджає заняттю або порушує академічну доброчесність. Обмеження не поширюються на погоджені засоби доступності та допоміжні технології.

Підтримка здобувачів з особливими освітніми потребами погоджується з дирекцією ННКІЗТ.

За проблем зі здоров'ям, що перешкоджають навчанню, здобувач звертається до медичного закладу та повідомляє дирекцію ННКІЗТ.

Скарги, пропозиції, зауваження та повідомлення про конфліктні ситуації в межах освітніх програм можна надсилати на електронну адресу general@ntu.edu.ua або через скриньку довіри,

розміщену при вході до університету.

Електронна скринька для звернень до психологічної служби: philosophy@ntu.edu.ua.

Канали зв'язку з розробником силабуса: e-mail oitkachen@gmail.com; тел.: +380 (63) 282-43-60.

Рекомендована література:

1. Mozilla Developer Network. HTTP documentation. URL: <https://developer.mozilla.org/en-US/docs/Web/HTTP>.
2. PHP Documentation. URL: <https://www.php.net/docs.php>.
3. Fielding R. T. Architectural Styles and the Design of Network-based Software Architectures. 2000. URL: <https://ics.uci.edu/~fielding/pubs/dissertation/top.htm>.
4. IETF. HTTP Semantics (RFC 9110). URL: <https://www.rfc-editor.org/rfc/rfc9110>.
5. OpenAPI Initiative. OpenAPI Specification. URL: <https://spec.openapis.org/oas/latest.html>.
6. PHP-FIG. PSR-7: HTTP Message Interface. URL: <https://www.php-fig.org/psr/psr-7/>.
7. PHPUnit Documentation. URL: <https://phpunit.de/documentation.html>.
8. OWASP. API Security Top 10. URL: <https://owasp.org/API-Security/>.
9. PHP-FIG. PSR-15: HTTP Server Request Handlers. URL: <https://www.php-fig.org/psr/psr-15/>.
10. Apache Software Foundation. Apache HTTP Server Documentation. URL: <https://httpd.apache.org/docs/>.
11. RFC Editor. Problem Details for HTTP APIs (RFC 9457). URL: <https://www.rfc-editor.org/rfc/rfc9457>.
12. Mesbah A., van Deursen A. An Architectural Style for Web Applications. 2007.
13. Google. Web performance fundamentals. URL: <https://web.dev/learn/performance/>.
14. ISO/IEC 25010:2023. Systems and software quality models.

Додаткова інформація

Детальнішу інформацію щодо методів навчання, практичних занять, форм оцінювання, самостійної роботи та повного списку літератури наведено у Робочій програмі навчальної дисципліни.