



**Приватний вищий навчальний заклад  
«Буковинський університет»  
Факультет інформаційних технологій та економіки  
Кафедра комп'ютерних систем і технологій**

---

СХВАЛЕНО  
на засіданні науково-методичної  
ради факультету  
протокол № 1 від 26 серпня 2025 р.

ЗАТВЕРДЖУЮ  
Декан факультету ІТЕ  
\_\_\_\_\_/ Тетяна ШТЕРМА/  
«\_\_\_» \_\_\_\_\_ 2025 р.

## СИЛАБУС

**обов'язкової навчальної дисципліни  
«Розробка клієнт-серверних застосувань»**

<b>Освітньо-професійна програма:</b>	<u>Комп'ютерні науки</u>
<b>Спеціальність:</b>	<u>Комп'ютерні науки</u>
<b>Галузь знань:</b>	<u>Інформаційні технології</u>
<b>Рівень вищої освіти:</b>	<u>перший (бакалаврський)</u>
<b>Факультет:</b>	<u>Інформаційних технологій та економіки</u>
<b>Мова навчання:</b>	<u>українська</u>
<b>Розробник:</b>	Артеменко Ольга Іванівна – кандидат технічних наук, доцент
<b>Профайл викладача:</b>	<a href="https://bukuniver.edu.ua/university/faculties-and-departments/ite-faculty/department-of-computer-systems-and-technologies/">https://bukuniver.edu.ua/university/faculties-and-departments/ite-faculty/department-of-computer-systems-and-technologies/</a>
<b>E-mail:</b>	<a href="mailto:olgaartemenko@bukuniver.edu.ua">olgaartemenko@bukuniver.edu.ua</a>
<b>Консультації:</b>	четвер з 10.00 до 16.00.

## 1. Анотація (призначення навчальної дисципліни).

Освітній компонент «Розробка клієнт-серверних застосунків» є складовою навчального плану підготовки фахівців першого (бакалаврського) рівня вищої освіти галузі знань **Інформаційні технології** спеціальності **Комп'ютерні науки**, яка побудована відповідно до вимог Європейської кредитної трансферно-накопичувальної системи (ECTS) і містить 5 кредитів. Форма підсумкового контролю – екзамен.

Предметом вивчення курсу «Розробка клієнт-серверних застосунків» є методи, технології та інструменти проєктування, розроблення й інтеграції клієнтських і серверних компонентів програмних систем, а також принципи організації взаємодії між ними через мережеві протоколи, API та служби обробки даних. До предметної області належать підходи до побудови архітектури клієнт-серверних систем, особливості роботи з базами даних, забезпечення безпеки, масштабованості, продуктивності та тестування розподілених застосунків.

## 2. Мета та завдання вивчення дисципліни.

*Мета курсу* – формування у здобувачів системи теоретичних знань та практичних навичок щодо архітектурних принципів побудови клієнт-серверних систем, методів розроблення та інтеграції клієнтських і серверних компонентів, а також використання сучасних технологій та інструментальних засобів для створення масштабованих і безпечних застосунків клієнт-серверної архітектури.

*Завдання дисципліни* – засвоєння здобувачами основних моделей та рівнів архітектури клієнт-серверної взаємодії, оволодіння методами проєктування та реалізації інтерфейсів користувача й серверної логіки, форматів обміну даними та принципів побудови API, а також набуття практичних навичок створення та розгортання клієнт-серверних застосунків, налагодження роботи із серверними базами даних, організацію безпеки передачі інформації та тестування працездатності програмних систем клієнт-серверної архітектури.

*Основні знання та вміння, яких набуває студент після опанування цієї дисципліни*

*Основні знання:*

- архітектурні моделі та рівні взаємодії клієнт-серверних систем (2-tier, 3-tier, N-tier);
- мережеві протоколи передачі даних (HTTP/HTTPS, WebSockets) та принципи їх функціонування;
- методи проєктування та архітектурні стилі побудови API (REST, GraphQL);
- сучасні формати обміну даними (JSON, XML) та способи їх серіалізації/десеріалізації;
- принципи організації взаємодії застосунку із серверними базами даних;
- методи автентифікації, авторизації та стандарти безпечної передачі інформації;
- інструментальні засоби та технології для розгортання, контейнеризації та тестування клієнт-серверних систем.

*Основні вміння:*

- проєктувати та реалізовувати адаптивні інтерфейси користувача (Frontend частина);
- розробляти серверну логіку, обробку запитів та бізнес-правил (Backend частина);
- створювати, документувати та інтегрувати програмні інтерфейси (API);
- реалізовувати обмін даними між клієнтом і сервером у реальному часі;
- налагоджувати стійкі з'єднання та виконувати CRUD-операції з базами даних;
- впроваджувати механізми захисту даних на рівні клієнта та сервера;
- розгортати клієнт-серверні застосунки на хостингових майданчиках або хмарних платформах;
- проводити комплексне тестування працездатності, продуктивності та безпеки системи.

*Предметом вивчення навчальної дисципліни* є архітектурні засади організації клієнт-серверної взаємодії та інженерні практики розроблення і розгортання програмних компонентів для реалізації функціональних можливостей систем на рівні інтерфейсу користувача та серверної логіки.

## 3. Пререквізити:

- ОК18 Організація баз даних та знань;
- ОК22 Об'єктно-орієнтоване програмування;
- ОК24 Технології створення програмних продуктів;
- ОК25 Веб-технології та веб-дизайн.

## 4. Компетентності та результати навчання.

Під час вивчення дисципліни, відповідно до освітньо-професійної програми, формуються компетентності:

**Інтегральна компетентність**

Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов і вимог.

### **Загальні компетентності (ЗК)**

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК6. Здатність вчитися і оволодівати сучасними знаннями.

ЗК11. Здатність приймати обґрунтовані рішення.

### **Спеціальні (фахові, предметні) компетентності**

СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальнення, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

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 і мережі передачі даних та аналізує якість роботи комп'ютерних мереж.

СК14. Здатність застосовувати методи та засоби забезпечення інформаційної безпеки, розробляти й експлуатувати спеціальне програмне забезпечення захисту інформаційних ресурсів об'єктів критичної інформаційної інфраструктури.

СК16. Здатність реалізовувати високопродуктивні обчислення на основі хмарних сервісів і технологій, паралельних і розподілених обчислень при розробці й експлуатації розподілених систем паралельної обробки інформації.

### **Програмні результати навчання**

РН9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

РН10. 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.

РН13. Володіти мовами системного програмування та методами розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення.

РН15. Розуміти концепцію інформаційної безпеки, принципи безпечного проектування програмного забезпечення, забезпечувати безпеку комп'ютерних мереж в умовах неповноти та невизначеності вихідних даних.

РН16. Виконувати паралельні та розподілені обчислення, застосовувати чисельні методи та алгоритми для паралельних структур, мови паралельного програмування при розробці та експлуатації паралельного та розподіленого програмного забезпечення.

## **5. Зміст навчальної дисципліни**

### **Змістовий модуль 1. Архітектурні основи та мережева взаємодія**

Тема 1. Архітектурні моделі клієнт-серверної взаємодії та мережеві протоколи.

Тема 2. Формати обміну даними та механізми їх оброблення в клієнт-серверних системах.

Тема 3. Проектування та розроблення програмних інтерфейсів на основі архітектурного стилю REST.

Тема 4. Організація взаємодії серверної логіки з базами даних та забезпечення цілісності даних.

### **Змістовий модуль 2. Безпека, тестування та розгортання систем**

Тема 5. Технології обміну даними в реальному часі та альтернативні підходи до побудови API.

Тема 6. Методи забезпечення безпеки, автентифікації та авторизації в клієнт-серверних застосунках.

Тема 7. Стратегії тестування, налагодження та аналізу продуктивності взаємодії компонентів.

Тема 8. Контейнеризація, розгортання та хмарна інтеграція клієнт-серверних систем.

## **6. Система контролю та оцінювання.**

### **Методи навчання:**

- словесні методи (лекція, дискусія, пояснення, розповідь);
- практичні методи (практичні заняття, практичні завдання);
- наочні методи (демонстрація, ілюстрація, презентація);
- робота з інформаційними ресурсами: з навчально-методичною, науковою, нормативною літературою та інтернет-ресурсами;
- самостійна робота над індивідуальним завданням або за програмою навчальної дисципліни;
- дистанційне навчання з використанням відповідних онлайн-платформ.

#### **Форми та методи оцінювання:**

- усне опитування;
- тестування;
- презентація результатів виконаних завдань;
- виконання вправ;
- контрольні роботи;
- підсумковий контроль –екзамен.

*Поточний контроль* проводиться на кожному практичному занятті.

*Об'єктом поточного контролю* знань студентів є:

- систематичність, якість та своєчасність виконання і захисту практичних робіт;
- систематичність та своєчасність виконання завдань самостійної роботи студента;
- якість виконання модульних контрольних робіт.

*Підсумковий контроль знань* проводиться у формі екзамена.

На екзамен виносяться вузлові питання дисципліни та типові задачі, що потребують творчої відповіді та уміння синтезувати отримані знання і застосувати їх при вирішенні практичних задач. Максимально можлива оцінка за екзамен – 30 балів.

Семестрова кількість балів може становити від 0 до 100 балів і визначається як сума балів: отриманих за всі види роботи на практичних заняттях; за виконання самостійної роботи; модульних контрольних робіт, результат підсумкового контролю у формі екзамена.

Оцінювання здійснюється за національною шкалою – «відмінно», «добре», «задовільно», «незадовільно» та за шкалою ECTS.

#### **Шкала оцінювання: національна та ECTS**

Оцінка за шкалою ECTS	Оцінка за шкалою, що використовується у закладі вищої освіти та фахової передвищої освіти	Оцінка за національною шкалою
A	90-100	5 (відмінно)
B	80-89	4 (добре)
C	70-79	
D	60-69	3 (задовільно)
E	50-59	
FX	35-49	2 (незадовільно) з можливістю повторного складання
F	1-34	2 (незадовільно) з обов'язковим повторним вивченням дисципліни

#### **Розподіл балів з навчальної дисципліни**

Поточний контроль (аудиторна та самостійна робота)		Іспит	Загальна кількість балів
Модуль I	Модуль II		
35	35	30	100

#### **Політика академічної доброчесності**

Студент зобов'язаний ознайомитися з Положенням про забезпечення академічної доброчесності у ПВНЗ «Буковинський університет» та неухильно його дотримуватися. Текст документа розміщено у відкритому доступі на офіційному сайті університету. В освітньому процесі студент має виявляти

дисциплінованість, ввічливість, доброзичливість, чесність і відповідальність.

Дотримання академічної доброчесності здобувачами освіти передбачає: самостійне виконання навчальних завдань, завдань підсумкового контролю результатів навчання (для осіб з особливими освітніми потребами ця вимога застосовується з урахуванням їхніх індивідуальних потреб і можливостей); посилення на джерела інформації у разі використання ідей, розробок, тверджень, відомостей; дотримання норм законодавства про авторське право і суміжні права; надання достовірної інформації про результати власної навчальної (наукової, творчої) діяльності, використані методики досліджень і джерела інформації. Списування (копіювання тексту) під час виконання письмових робіт заборонені. Самостійні роботи у вигляді рефератів, доповідей, презентацій повинні мати коректні текстові посилення на використані інформаційні джерела. Дозволяється використання інструментів штучного інтелекту за умови дотримання принципів академічної доброчесності.

## 7.Рекомендована література

1. Бунке О. С. *Серверні WEB-технології : навч. посіб.* – Київ : Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», 2023. – 120 с. – URL: [https://ela.kpi.ua/bitstream/123456789/57288/1/Serverni\\_2023.pdf](https://ela.kpi.ua/bitstream/123456789/57288/1/Serverni_2023.pdf) (дата звернення: 29.07.2025).
2. Піскунов О. Г. *Розробка клієнт-серверних додатків засобами мови C# та їхнє документування : навч. посіб.* – Київ : Персонал, 2024. – 358 с.
3. Sharma S. *Modern API Development with Spring 6 and Spring Boot 3 : design modern scalable and maintainable web APIs with REST, gRPC, GraphQL and reactive paradigm.* – Birmingham : Packt Publishing, 2023. – 494 с. – ISBN 978-1804613276.
4. Varga M., de Andrade P. H. P., Bomfim T., de Moraes S., Fraga I. A. *Mastering RESTful Web Services with Java : practical guide for building secure and scalable production-ready REST APIs.* – Birmingham : Packt Publishing, 2025. – 432 с. – ISBN 978-1835466100.
5. Madden N. *API Security in Action.* – Shelter Island, NY : Manning Publications, 2021. – 576 с. – ISBN 978-1617296022.
6. Walls C. *Spring in Action.* – Shelter Island, NY : Manning Publications, 2021. – 6-е вид. – ISBN 978-1617294945.
7. Крилов Є. В., Анікін В. К. *Технології проєктування No SQL баз даних : навч. посібник / Є. В. Крилов, В. К. Анікін.* – Електронні текст. дані (1 файл). – Київ : КПІ ім. Ігоря Сікорського, 2023. URL: <https://ela.kpi.ua/handle/123456789/63578> (дата звернення: 29.07.2025).
8. Радченко К. О., Петрашенко А. В. *Бази даних NoSQL. Конспект лекцій : навч. посіб. / К. О. Радченко, А. В. Петрашенко.* – Електронні текст. дані (1 файл). – Київ : КПІ ім. Ігоря Сікорського, 2023. URL: <https://ela.kpi.ua/handle/123456789/54819> (дата звернення: 29.07.2025).
9. Артеменко О., Полтавський Д. Methods for improving the performance and speed of cross-platform mobile applications. *Наука і техніка сьогодні.* 2025. № 5(46). DOI:10.52058/2786-6025-2025-5(46).
10. Artemenko, O., Kaidyk, O. L., Terletskyi, T. V., Pugach, S. O., & Uhryn, D. I. (2025). *Modeling and improving the reliability of access subject identification in access control systems based on hybrid PCA algorithms.* COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION, (61), 82–90. <https://doi.org/10.36910/6775-2524-0560-2025-61-12>
11. Козуб Г. О., Сурма Ю. Ю., Артеменко О. І. Роль вебкомпонентів у побудові сучасних інтерфейсів: переваги та обмеження / Вісник Херсонського національного технічного університету. – 2025. – № 2(93), ч. 2. – С. 175-180. – DOI: 10.35546/kntu2078-4481.2025.2.2.21

### Інформаційні ресурси:

1. MongoDB : офіц. документація [Електронний ресурс]. – URL: <https://www.mongodb.com/docs/> (дата звернення: 29.07.2025).
2. PostgreSQL : офіц. документація [Електронний ресурс]. – URL: <https://www.postgresql.org/docs/> (дата звернення: 29.07.2025).
3. W3C. Extensible Markup Language (XML) [Електронний ресурс]. – URL: <https://www.w3.org/XML/> (дата звернення: 29.07.2025).
4. W3Schools. XML Tutorial [Електронний ресурс]. – URL: <https://www.w3schools.com/xml/> (дата звернення: 29.07.2025).
5. W3C. JSON-LD 1.1 (Linked Data) [Електронний ресурс]. – URL: <https://www.w3.org/TR/json-ld11/> (дата звернення: 29.07.2025).
6. W3Schools. PostgreSQL Tutorial [Електронний ресурс]. – URL: <https://www.w3schools.com/postgresql/> (дата звернення: 29.07.2025).

7. Docker. Документація [Електронний ресурс]. – URL: <https://docs.docker.com/> (дата звернення: 29.07.2025).
8. Kubernetes. Документація [Електронний ресурс]. – URL: <https://kubernetes.io/docs/> (дата звернення: 29.07.2025).
9. Spring Boot. Офіційна документація [Електронний ресурс]. – URL: <https://spring.io/projects/spring-boot> (дата звернення: 29.07.2025).
10. Spring Guides. Tutorials for Spring Boot [Електронний ресурс]. – URL: <https://spring.io/guides> (дата звернення: 29.07.2025).