

ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ



СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ  
ТЕХНОЛОГІЇ РОЗРОБКИ ПРОГРАМНИХ ПРОДУКТІВ

підготовки \_\_\_\_\_ бакалаврів \_\_\_\_\_

денної та заочної форми здобуття освіти

освітньо-професійна програма \_\_\_\_\_ Комп'ютерні науки \_\_\_\_\_

спеціальності \_\_\_\_\_ 122 Комп'ютерні науки \_\_\_\_\_

галузі знань \_\_\_\_\_ 12 Інформаційні технології \_\_\_\_\_

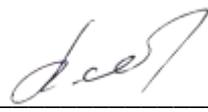
**ВИКЛАДАЧ:** Миронова Наталя Олексіївна, к.т.н., доцент, доцент кафедри комп'ютерних наук

Обговорено та ухвалено  
на засіданні кафедри комп'ютерних наук

Протокол № 1 від 25 серпня 2025 р.  
Завідувач кафедри комп'ютерних наук

Погоджено  
Гарант освітньо-професійної програми

  
\_\_\_\_\_ Г. М. Шило \_\_\_\_\_  
(підпис) (ініціали, прізвище)

  
\_\_\_\_\_ Н.В. Матвіїшина \_\_\_\_\_  
(підпис) (ініціали, прізвище)

2025 рік



**Зв'язок з викладачем: Миронова Наталя Олексіївна**

**E-mail:** [natali.myronova@gmail.com](mailto:natali.myronova@gmail.com)

**Сезн ЗНУ повідомлення:** <https://moodle.znu.edu.ua/course/view.php?id=17784>

**Телефон (кафедра):** 289-12-57

**Кафедра** комп'ютерних наук, ауд. №39, 1 корпус ЗНУ

## 1. Опис навчальної дисципліни

Дисципліна «Технології розробки програмних продуктів» є вибіркоким освітнім компонентом для здобувачів бакалаврського рівня спеціальності 122 Комп'ютерні науки, освітньо-професійної програми Комп'ютерні науки.

У даному курсі розглядаються сучасні методи, інструменти та технології, що використовуються при створенні програмних продуктів. Основна увага приділяється процесу розробки від аналізу вимог до проектування, прототипування, тестування та впровадження кінцевих систем. Вивчаються архітектурні стилі, шаблони проектування, сучасні платформи для командної розробки та підходи до автоматизації життєвого циклу програмного забезпечення. Компетентності, отримані під час вивчення освітньої компоненти «Технології розробки програмних продуктів», необхідні для написання розділів кваліфікаційної роботи бакалавра.

**Метою вивчення** освітньої компоненти «Технології розробки програмних продуктів» є формування у студентів цілісного уявлення про процес створення програмних продуктів, розвиток практичних навичок аналізу вимог, проектування архітектури, використання сучасних мов програмування, систем управління базами даних та інструментів командної роботи. Особливий акцент робиться на засвоєнні принципів прототипування користувацького інтерфейсу, використанні систем контролю версій, юніт-тестуванні та безперервній інтеграції (CI/CD).

У результаті вивчення навчальної дисципліни «Технології розробки програмних продуктів» здобувач освіти повинен **знати**:

- основні етапи життєвого циклу програмного забезпечення (аналіз вимог, проектування, розробка, тестування, розгортання та підтримка);
- сучасні підходи до розробки програмних систем (Waterfall, Agile, DevOps);
- принципи побудови архітектури програмних систем (монолітна, мікросервісна, клієнт-серверна);
- мови програмування та парадигми (об'єктно-орієнтовану, функціональну, процедурну);
- основні поняття та принципи роботи систем керування базами даних (реляційних і NoSQL);
- принципи роботи ORM (Object-Relational Mapping) та способи інтеграції з БД;
- підходи до проектування інтерфейсу користувача, правила UX/UI, інструменти прототипування (Figma);
- засади тестування програмних продуктів (ручне, автоматизоване, юніт-тестування);
- інструменти безперервної інтеграції та розгортання (CI/CD, GitHub Actions, Docker, Kubernetes);
- вимоги до безпеки програмних систем, зокрема аутентифікацію, авторизацію та захист даних.



**вміти:**

- аналізувати вимоги до програмного продукту та формувати технічне завдання;
- створювати UML-діаграми (Use Case, Activity, Class) для візуалізації логіки системи;
- проектувати архітектуру програмного забезпечення з використанням шаблонів (MVC, MVVM, Repository, Factory);
- розробляти прототипи користувацького інтерфейсу в Figma або подібних інструментах;
- створювати та оптимізувати бази даних, виконувати SQL-запити;
- реалізовувати зв'язок застосунку з базою даних за допомогою ORM-фреймворків;
- програмувати клієнтську та серверну частину застосунку з використанням сучасних мов та фреймворків (C#, Python, Java, JavaScript, Node.js, ASP.NET Core, Django);
- проводити тестування програмного продукту, складати тест-кейси, створювати автоматизовані тести;
- організовувати процес CI/CD, забезпечувати збірку, тестування і розгортання проекту;
- працювати в команді над спільним проектом, використовувати системи контролю версій (Git, GitHub).

### Паспорт навчальної дисципліни

Нормативні показники	денна форма здобуття освіти	заочна форма здобуття освіти
<i>1</i>	<i>2</i>	<i>3</i>
Статус дисципліни	<b>Вибіркова</b>	
Семестр	7-й	7-й
Кількість кредитів ECTS	6	6
Кількість годин	180 год.	180 год.
Лекційні заняття	28	10
Лабораторні заняття	28	8
Самостійна робота	124	162
Консультації	За розкладом; дистанційно: <a href="https://meet.google.com/ins-srkc-wyy">https://meet.google.com/ins-srkc-wyy</a>	
Вид підсумкового семестрового контролю:	<b>залік</b>	
Посилання на електронний курс у СЕЗН ЗНУ (платформа Moodle)	<a href="https://moodle.znu.edu.ua/course/view.php?id=17784">https://moodle.znu.edu.ua/course/view.php?id=17784</a>	



## 2. Методи досягнення запланованих освітньою програмою компетентностей і результатів навчання

Компетентності/ результати навчання	Методи навчання	Форми і методи оцінювання
<b>Компетентності</b>		
ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.	пояснення, обговорення, виконання завдань лабораторних та самостійних робіт	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування <b>Підсумковий контроль:</b> захист проєктів
ЗК2. Здатність застосовувати знання у практичних ситуаціях.	виконання завдань лабораторних та самостійних робіт, проєктні роботи	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.	лекції, аналіз літератури, виконання завдань самостійних робіт	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
ЗК4. Здатність спілкуватися державною мовою як усно, так і письмово.	усні презентації, виконання завдань лабораторних та самостійних робіт, проєктні роботи	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
ЗК6. Здатність вчитися й оволодівати сучасними знаннями.	робота з науковою та професійною літературою, самостійне опрацювання матеріалу, онлайн-курси	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.	робота з науковою та професійною літературою, самостійне опрацювання матеріалу, онлайн-курси	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
ЗК8. Здатність генерувати нові ідеї (креативність).	мозкові штурми, виконання завдань лабораторних та самостійних робіт, проєктні роботи	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація



		<p>проектів</p> <p><b>Підсумковий контроль:</b> захист проектів</p>
<p>ЗК9. Здатність працювати в команді.</p>	<p>групові проекти, командні завдання, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт, аналіз</p>	<p><b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проектів</p> <p><b>Підсумковий контроль:</b> захист проектів</p>
<p>СК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.</p>	<p>пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт</p>	<p><b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проектів</p> <p><b>Підсумковий контроль:</b> захист проектів</p>
<p>СК6. Здатність до системного мислення, застосування методології системного аналізу для дослідження складних проблем різної природи, методів формалізації та розв'язування системних задач, що мають суперечливі цілі, невизначеності та ризику.</p>	<p>лекції, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт</p>	<p><b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проектів</p> <p><b>Підсумковий контроль:</b> захист проектів</p>
<p>СК7. Здатність застосовувати теоретичні та практичні основи методології та технології моделювання для дослідження характеристик і поведінки складних об'єктів і систем, проводити обчислювальні експерименти з обробкою й аналізом результатів.</p>	<p>лекції, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт</p>	<p><b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проектів</p> <p><b>Підсумковий контроль:</b> захист проектів</p>
<p>СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування:</p>	<p>лекції, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт</p>	<p><b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проектів</p>



узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.		<b>Підсумковий контроль:</b> захист проєктів
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.	лекції, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт, проєктні роботи	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
СК10. Здатність застосовувати методології, технології та інструментальні засоби для управління процесами життєвого циклу інформаційних і програмних систем, продуктів і сервісів інформаційних технологій відповідно до вимог замовника.	лекції, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
<b>Результати навчання</b>		
ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.	пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
ПР8. Використовувати методологію системного аналізу об'єктів, процесів і систем для задач аналізу, прогнозування, управління та проєктування динамічних процесів в макроекономічних, технічних, технологічних і фінансових об'єктах.	лекції, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт	<b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проєктів <b>Підсумковий контроль:</b> захист проєктів
ПР9. Розробляти програмні моделі предметних	лекції, пояснення, демонстрування,	<b>Поточний контроль:</b> захист лабораторних та



<p>середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.</p>	<p>виконання завдань лабораторних та самостійних робіт</p>	<p>самостійних робіт, опитування, презентація проектів <b>Підсумковий контроль:</b> захист проектів</p>
<p>ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосунків, 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.</p>	<p>пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт</p>	<p><b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проектів <b>Підсумковий контроль:</b> захист проектів</p>
<p>ПР11 Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).</p>	<p>лекції, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт</p>	<p><b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проектів <b>Підсумковий контроль:</b> захист проектів</p>
<p>ПР14. Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктноорієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничотехнічних систем.</p>	<p>лекції, пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт</p>	<p><b>Поточний контроль:</b> захист лабораторних та самостійних робіт, опитування, презентація проектів <b>Підсумковий контроль:</b> захист проектів</p>



### 3. Зміст навчальної дисципліни

#### Змістовий модуль 1. Основи технологій розробки програмних продуктів

##### *Тема 1. Вступ до технологій розробки програмних продуктів*

Історія розвитку технологій розробки програмного забезпечення. Життєвий цикл програмного продукту: традиційні моделі (Waterfall), ітеративні та гнучкі методології (Agile, Scrum, Kanban). Основні етапи розробки: аналіз, проектування, програмування, тестування, впровадження, супровід. Види програмних продуктів: системне ПЗ, прикладне ПЗ, веб-застосунки, мобільні застосунки. Основи управління проектами та ролі учасників команди розробників.

##### *Тема 2. Аналіз вимог і проектування*

Поняття вимог до ПЗ: функціональні, нефункціональні, бізнес-вимоги. Етапи збору, аналізу й документування вимог. Методи збору вимог: інтерв'ю, опитування, аналіз сценаріїв використання, створення користувачьких історій (User Stories). Стандарти IEEE для вимог. Візуалізація вимог за допомогою UML: діаграми прецедентів (Use Case), діаграми діяльності (Activity Diagram), діаграми послідовностей (Sequence Diagram). Основи BPMN-нотації для моделювання бізнес-процесів. Приклади вимог до реальних систем (CRM, навчальні портали, e-commerce).

##### *Тема 3. Архітектурні стилі та шаблони проектування*

Основні архітектурні стилі: монолітна, клієнт-сервер, мікросервіси, серверless. Принципи модульності та компонентності в архітектурі. Архітектурні шаблони проектування програмних продуктів (MVC, MVVM, MVT). Популярні шаблони проектування (Design Patterns): одинак (Singleton), фабрика (Factory), фасад (Facade) спостерігач (Observer), стратегія (Strategy). Використання шаблонів для підвищення гнучкості та повторного використання коду. Практичні приклади застосування архітектурних стилів у сучасних інформаційних системах.

#### Змістовий модуль 2. Технології програмування та бази даних

##### *Тема 4. Технології програмування (мови та парадигми)*

Історія розвитку мов програмування та еволюція парадигм програмування. Основні типи мов: низькорівневі (асемблер), високорівневі (C, C++, Java, Python, C#), скриптові та функціональні. Парадигми програмування: процедурне, Об'єктно-орієнтоване, функціональне, логічне програмування. Критерії вибору мови та парадигми для розв'язання конкретних задач. Практичні приклади застосування різних парадигм у сучасних програмних продуктах.

##### *Тема 5. Системи керування базами даних (СКБД).*

Поняття СКБД, їх роль у зберіганні та обробці даних. Архітектура баз даних: клієнт-сервер, багаторівнева, розподілена. Реляційні СКБД (PostgreSQL, MySQL, MSSQL): створення таблиць, первинні та зовнішні ключі, зв'язки, індекси, транзакції. NoSQL-підходи (MongoDB, Redis): документоорієнтовані, графові, колоночні та ключ-значення сховища. Порівняння ACID і BASE моделей. Інструменти ORM (Entity Framework, Hibernate, SQLAlchemy): переваги, недоліки, приклади інтеграції з ПЗ.



### *Тема 6. Проектування баз даних*

Етапи проектування БД: концептуальне, логічне та фізичне моделювання. ER-моделі (сутність-зв'язок): сутності, атрибути, типи зв'язків (1:1, 1:N, M:N). Процес нормалізації: 1NF, 2NF, 3NF. Визначення ключів і залежностей. SQL-мовлення: створення, вставка, вибірка, оновлення, видалення (CREATE, SELECT, UPDATE, DELETE). Оптимізація запитів і використання індексів. Забезпечення цілісності даних, реплікація, резервне копіювання. Приклад побудови БД для навчального порталу або CRM-системи.

## **Змістовий модуль 3. Інтерфейси, серверна та клієнтська частини систем**

### *Тема 7. Прототипування та інструменти UI/UX*

Принципи проектування користувацького інтерфейсу: ергономіка, юзабіліті, доступність. Створення прототипів: паперові прототипи, цифрові прототипи у Figma, Adobe XD, Sketch. Wireframes і каркасні прототипи. Етапи створення інтерфейсу: аналіз користувача, сценарії, каркас, інтерактивний прототип. Принципи візуальної ієрархії, кольорова схема, адаптивність. UX-тестування прототипів. Тестування прототипів із користувачами, зворотний зв'язок та ітеративне покращення дизайну. Взаємодія UI/UX дизайну з бекендом і базами даних.

### *Тема 8. Технології реалізації серверної частини (Backend).*

Роль серверної частини у системі. Архітектура REST API. Framework-и: ASP.NET Core (C#), Spring Boot (Java), Django (Python), Node.js (JavaScript). Організація обробки запитів, middleware, маршрутизація, контролери. Робота з базами даних через ORM. Авторизація та аутентифікація користувачів. Логування, обробка помилок, unit-тестування backend-коду. Приклад створення REST-сервісу з базою PostgreSQL.

### *Тема 9. Технології реалізації клієнтської частини (Frontend).*

Основи HTML5, CSS3, JavaScript. Компонентна архітектура веб-застосунків. Framework-и: React, Angular, Vue.js. Робота з API, обробка подій, стан додатка (Redux, Vuex). Адаптивна верстка та оптимізація. Використання UI-бібліотек (Bootstrap, Material UI). Інтерактивні елементи, валідація форм, асинхронні запити (Axios, Fetch). Приклад побудови клієнтського інтерфейсу для системи замовлень із REST API.

### *Тема 10. Інтеграція та взаємодія компонентів системи.*

Поняття інтеграції. Методи інтеграції: монолітна інтеграція, сервіс-орієнтована інтеграція, мікросервісна архітектура. Способи обміну даними: REST, GraphQL, gRPC, WebSocket. Мікросервісна взаємодія через API-шлюзи. Інструменти: Postman, Swagger, RabbitMQ, Kafka. Формати передачі даних – JSON, XML, Protocol Buffers. Підключення зовнішніх сервісів (платіжні системи, аналітика, авторизація через OAuth). Контроль сумісності та версій API. Приклад інтеграції між клієнтською, серверною частинами і базою через REST.

## **Змістовий модуль 4. Забезпечення якості, безпека та підтримка програмних продуктів**



### *Тема 11. Тестування програмних продуктів*

Поняття тестування, його мета та види. Рівні тестування: модульне, інтеграційне, системне, приймальне. Тест-план, тест-кейс, звіт про дефекти. Ручне тестування: сценарії користувачів, перевірка функціоналу. Автоматизоване тестування: інструменти JUnit, NUnit, xUnit, Selenium. Методи перевірки продуктивності та навантаження. Створення тестів для API та UI. Приклад написання unit-тестів у C# або Python.

### *Тема 12. CI/CD та DevOps практики*

Поняття безперервної інтеграції (CI) та доставки (CD). Основи DevOps-культури. Інструменти автоматизації: GitHub Actions, GitLab CI/CD, Jenkins. Система контейнеризації Docker, оркестрація Kubernetes. Автоматичне тестування, збірка, розгортання. Структура пайплайну, змінні середовища, артефакти. Приклад: автоматизоване розгортання веб-застосунку на Heroku або AWS

### *Тема 13. Безпека програмних продуктів*

Основи інформаційної безпеки: конфіденційність, цілісність, доступність. Методи захисту серверів, баз даних та мережевих протоколів. Принципи безпечного програмування, захист від SQL-ін'єкцій, XSS, CSRF та інших вразливостей. Стандарти OWASP Top-10. Методи захисту: валідація вхідних даних, контроль доступу, шифрування, токени. Аутентифікація і авторизація (JWT, OAuth2). Захист REST API. Безпечне зберігання паролів і персональних даних. Приклад реалізації системи входу користувачів у веб-застосунку з токенами

### *Тема 14. Розгортання та підтримка програмних продуктів*

Методи розгортання програмних продуктів: локально, на сервері, у хмарі. Управління оновленнями та виправленням багів. Підтримка користувачів, резервне копіювання та відновлення даних. Хмарні сервіси: AWS, Azure, Google Cloud, Heroku. Моніторинг і логування (Grafana, Prometheus, ELK-Stack). Резервне копіювання, відновлення після збою, управління версіями. Підтримка і розвиток програмного продукту, DevOps-цикл.



#### 4. Структура навчальної дисципліни

Вид заняття /роботи	Назва теми	Кількість годин		Згідно з розкладом
		о/д. ф.	з.ф.	
Лекція 1	Вступ до технологій розробки програмних продуктів	2	0,5	тиждень 1
Лекція 2	Аналіз вимог і проектування ПЗ	2	0,5	тиждень 2
Лабораторна робота 1	Аналіз вимог та створення UML-діаграм	4	1	тиждень 1, 2
Лекція 3	Архітектурні стилі та шаблони проектування	2	0,5	тиждень 3
Лекція 4	Технології програмування (мови та парадигми)	2	0,5	тиждень 4
Лабораторна робота 2	Проектування архітектури застосунку	4	1	тиждень 3, 4
Лекція 5	Системи керування базами даних	2	0,5	тиждень 5
Лекція 6	Розробка бази даних	2	0,5	тиждень 6
Лабораторна робота 3	Розробка бази даних	4	1	тиждень 5, 6
Лекція 7	Прототипування інтерфейсу користувача	2	0,5	тиждень 7
Лекція 8	Технології реалізації серверної частини (Backend)	2	1	тиждень 8
Лабораторна робота 4	Прототипування інтерфейсу користувача	4	1	тиждень 7, 8
Самостійна робота	Програмна інженерія: моделювання програмних систем за допомогою UML URL: <a href="https://www.coursera.org/learn/software-engineering-modeling-software-systems-using-uml">https://www.coursera.org/learn/software-engineering-modeling-software-systems-using-uml</a> (23 години) ; User Interface Design and Prototyping URL: <a href="https://www.coursera.org/learn/user-interface-design-and-prototyping">https://www.coursera.org/learn/user-interface-design-and-prototyping</a> (17 годин)	40	60	тиждень 1-8
Лекція 9	Технології реалізації клієнтської частини (Frontend)	2	1	тиждень 9
Лекція 10	Інтеграція та взаємодія компонентів системи	2	0,5	тиждень 10
Лабораторна робота №5	Розробка серверної частини застосунку	4	2	тиждень 9, 10
Лекція 11	Тестування програмних продуктів	2	0,5	тиждень 11
Лекція 12	CI/CD та DevOps практики	2	0,5	тиждень 12
Лабораторна робота №6	Розробка клієнтської частини застосунку	4	2	тиждень 11, 12



Лекція 13	Безпека програмних продуктів	2	0,5	тиждень 13
Лекція 14	Розгортання та підтримка програмних продуктів	2	0,5	тиждень 14
Лабораторна робота №7	Тестування та розгортання	4	-	тиждень 13, 14
Самостійна робота	Вступ до Back-End розробки URL: <a href="https://www.coursera.org/learn/introduction-to-back-end-development">https://www.coursera.org/learn/introduction-to-back-end-development</a> (18 годин) Вступ до Front-End розробки URL: <a href="https://www.coursera.org/learn/introduction-to-front-end-development">https://www.coursera.org/learn/introduction-to-front-end-development</a> (18 годин)	36	60	тиждень 9-14
Підготовка проєкту	Підготовка проєктної командної презентації	48	44	тиждень 6-14

Методичні рекомендації до лабораторних занять розміщено СЕЗН ЗНУ Moodle на сторінці дисципліни «Технології розробки програмних продуктів».

### 5. Види і зміст контрольних заходів

Вид заняття/роботи	Вид контрольного заходу	Зміст контрольного заходу*	Критерії оцінювання та термін виконання	Усього балів
<b>Поточний контроль</b>				
Лабораторна робота	Захист лабораторної роботи 1	Розміщено в СЕЗН ЗНУ	Повне виконання завдання л/р оцінюється в 6 балів, за наявності помилок – 3 бали	6
Лабораторна робота	Захист лабораторної роботи 2	Розміщено в СЕЗН ЗНУ	Повне виконання завдання л/р оцінюється в 6 балів, за наявності помилок – 3 бали.	6
Лабораторна робота	Захист лабораторної роботи 3	Розміщено в СЕЗН ЗНУ	Повне виконання завдання л/р оцінюється в 6 балів, за наявності помилок – 3 бали	6
Лабораторна робота	Захист лабораторної роботи 4	Розміщено в СЕЗН ЗНУ	Повне виконання завдання л/р оцінюється в 6 балів, за наявності помилок – 3 бали.	6
Самостійна робота	Захист самостійної роботи 1	Розміщено в СЕЗН ЗНУ	Повне виконання завдання с/р оцінюється в 4 бали, за наявності помилок – 2 бали	4
Лабораторна робота	Захист лабораторної роботи 5	Розміщено в СЕЗН ЗНУ	Повне виконання завдання л/р оцінюється в 6 балів, за наявності помилок – 3 бали	6
Лабораторна робота	Захист лабораторної роботи 6	Розміщено в СЕЗН ЗНУ	Повне виконання завдання л/р оцінюється в 6 балів, за наявності помилок – 3 бали	6
Лабораторна робота	Захист лабораторної роботи 7	Розміщено в СЕЗН ЗНУ	Повне виконання завдання л/р оцінюється в 6 балів, за наявності помилок – 3 бали	6
Самостійна робота	Захист самостійної	Розміщено в СЕЗН ЗНУ	Повне виконання завдання с/р оцінюється в 4 бали, за	4



Вид заняття/ роботи	Вид контрольного заходу	Зміст контрольного заходу*	Критерії оцінювання та термін виконання	Усього о балів
	роботи 2		наявності помилок – 2 бали	
Самостійна робота	Презентація проекту	Розміщено в СЕЗН ЗНУ	Завантаження презентації проекту: – 10 балів – тема повністю розкрита, глибокий аналіз, є власні висновки, практичне значення; – 8 балів – тема розкрита, логічна структура, є аналітичні елементи; – 6 балів – поверхове розкриття, не вистачає прикладів або висновків; – 0-5 балів – зміст неповний або не відповідає темі.	10
<b>Усього за поточний контроль</b>				<b>60</b>
<b>Підсумковий контроль</b>				
<b>Залік</b>	Тестування	Розміщено в СЕЗН ЗНУ	10 тестових завдань правильна відповідь на кожне оцінюється в 2 бали	<b>20</b>
	Захист проекту	Розміщено в СЕЗН ЗНУ	Критерії оцінювання проектної презентації: 1. Усний виступ і комунікація – до 10 балів: – 10 балів – чітка, впевнена мова, добрий темп, контакт із аудиторією, уміння відповідати на запитання; – 7–9 балів – переважно впевнено, але є незначна невпевненість або зчитування з екрана; – 4–6 балів – мова невиразна, часті підглядання, слабкий контакт з аудиторією; – 0–3 бали – відсутня підготовка, виступ не структурований. 2. Креативність і самостійність – до 5 балів: 5 балів – нестандартний підхід, оригінальні ідеї, авторські рішення або прикладі;	<b>20</b>



Вид заняття/ роботи	Вид контрольного заходу	Зміст контрольного заходу*	Критерії оцінювання та термін виконання	Усього балів
			– 3–4 бали – є елементи творчості, частково використано власні підходи; – 2–3 бали – здебільшого шаблонне виконання; – 0–1 бал – відсутність самостійності, копіювання готових матеріалів. 4. Аргументованість і відповіді на запитання – до 5 балів: – 5 балів – чіткі, логічні відповіді, демонстрація глибокого розуміння теми; – 3–4 бали – відповіді здебільшого правильні, але не завжди достатньо обґрунтовані; – 1–2 бали – часткове розуміння теми, невпевнені відповіді; – 0 бали – не може пояснити власний проект або ухиляється від відповідей	
<b>Усього за підсумковий контроль</b>				<b>40</b>

### Шкала оцінювання ЗНУ: національна та ECTS

За шкалою ECTS	За шкалою університету	За національною шкалою	
		Екзамен	Залік
A	90 – 100 (відмінно)	5 (відмінно)	Зараховано
B	85 – 89 (дуже добре)	4 (добре)	
C	75 – 84 (добре)		
D	70 – 74 (задовільно)	3 (задовільно)	
E	60 – 69 (достатньо)		
FX	35 – 59 (незадовільно – з можливістю повторного складання)	2 (незадовільно)	Не зараховано
F	1 – 34 (незадовільно – з обов'язковим повторним курсом)		

## 6. Основні навчальні ресурси

### Рекомендована література

1. Алексенко О. В. Технології програмування та створення програмних продуктів: конспект лекцій для студ. напряму підготовки 6.050101 "Комп'ютерні науки" усіх форм навчання. Суми : СумДУ, 2019. 133 с.



2. Бородкіна І. Л., Бородкін Г. О. Інженерія програмного забезпечення: посібник для студентів вищих навчальних закладів. Київ: Національний університет біоресурсів та природокористування України, 2018. 251 с.

3. Дегтярєва Л.М., Гроза П.М., Сомов С.В., Навчальний посібник з дисципліни «Технології розробки програмного забезпечення» для студентів спеціальності 123 «Комп'ютерна інженерія». Полтава: ПолтНТУ, 2018. 218 с.

4. Трофименко О. Г., Манаков Д. Г., Ларін С. Ю. Основи програмної інженерії: навч.-метод. посіб. Одеса: Фенікс, 2022. 197 с.

5. Цибульник С. О. Технології розроблення програмного забезпечення. Частина 1. Життєвий цикл програмного забезпечення Київ: КПІ ім. Ігоря Сікорського, 2022. 270 с.

### **Інформаційні ресурси**

1. Handbook of Software Engineering Methods - 2nd Edition URL: <https://open.umn.edu/opentextbooks/textbooks/1557>

2. Fundamental Approaches to Software Engineering URL: <https://library.oapen.org/handle/20.500.12657/23333?>



## 7. Регуляції і політики курсу

### **Відвідування занять. Регуляція пропусків.**

Відвідування усіх занять є обов'язковим. Студенти зобов'язані дотримуватися усіх строків, визначених для виконання усіх видів робіт, передбачених даною дисципліною. Пропуски та запізнення на заняття є недопустимими.

### **Політика академічної доброчесності**

Кожний студент зобов'язаний дотримуватися принципів академічної доброчесності. Письмові завдання з використанням часткових або повнотекстових запозичень з інших робіт без зазначення авторства – це *плагіат*. Використання будь-якої інформації (текст, фото, ілюстрації тощо) мають бути правильно процитовані з посиланням на автора! Якщо ви не впевнені, що таке плагіат, фабрикація, фальсифікація, порадьтеся з викладачем. До студентів, у роботах яких буде виявлено списування, плагіат чи інші прояви недоброчесної поведінки можуть бути застосовані різні дисциплінарні заходи (див. посилання на Кодекс академічної доброчесності ЗНУ в додатку до силабусу). Неприпустиме складання роботи, виконаної іншою особою.

### **Використання комп'ютерів/телефонів на занятті**

Використання мобільних телефонів, ноутбуків та інших гаджетів під час лекційних та лабораторних занять дозволяється виключно у навчальних цілях (з активованим режимом «без звуку»).

### **Комунікація**

Комунікація викладача зі студентами здійснюється безпосередньо на заняттях та додатково за допомогою месенджерів (наприклад, Telegram), електронної пошти і в СЕЗН Moodle (форум курсу, приватні повідомлення)



## ДОДАТКОВА ІНФОРМАЦІЯ

**ГРАФІК ОСВІТНЬОГО ПРОЦЕСУ НА 2025-2026 н.р.** доступний за адресою: <https://surl.li/vlweoj>

**НАВЧАННЯ ТА ЗАБЕЗПЕЧЕННЯ ЯКОСТІ ОСВІТИ.** Перевірка набутих студентами знань, навичок та вмінь є невід'ємною складовою системи забезпечення якості освіти і проводиться відповідно до Положення про організацію та методику проведення поточного та підсумкового семестрового контролю навчання студентів Запорізького національного університету: <https://surl.li/wdzjrl>

**ПОВТОРНЕ ВИВЧЕННЯ ДИСЦИПЛІН.** Наявність академічної заборгованості до 6 навчальних дисциплін (у тому числі проходження практики чи виконання курсової роботи) за результатами однієї екзаменаційної сесії є підставою для надання студенту права на повторне вивчення зазначених навчальних дисциплін. Процедура повторного вивчення визначається Положенням про порядок повторного вивчення навчальних дисциплін та повторного навчання у ЗНУ: <https://surl.lu/hfjbya>

**ВИРІШЕННЯ КОНФЛІКТІВ.** Порядок і процедури врегулювання конфліктів, пов'язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що можуть виникнути під час навчання, регламентуються Положенням про порядок і процедури вирішення конфліктних ситуацій у ЗНУ: <https://surl.li/qgacqa>

Конфліктні ситуації, що виникають у сфері стипендіального забезпечення здобувачів вищої освіти, вирішуються стипендіальними комісіями факультетів, коледжів та університету в межах їх повноважень, відповідно до:

Положення про порядок призначення і виплати академічних стипендій у ЗНУ: <https://surl.li/unwzzm>

Положення про призначення та виплату соціальних стипендій у ЗНУ: <https://surl.lu/xkxmuz>

**ПСИХОЛОГІЧНА ДОПОМОГА.** Кабінет практичного психолога **Марі Ірини Вадимівни** – навч. корп. №4, каб. №235 (понеділок, середа, четвер 9.00-11.00, 13.00-15.00), навч. корп. №9 (ІННІ) каб.57 (п'ятниця 9.00-11.00, 13.00-15.00), гуртожиток №6 (вул. Добролюбова, 19, середа 9.00-11.00, 13.00-15.00). Попередній запис за тел.: 228-76-48, (099) 253-78-73 щоденно з 9 до 15.

**УПОВНОВАЖЕНА ОСОБА З ПИТАНЬ ЗАПОБІГАННЯ ТА ВИЯВЛЕННЯ КОРУПЦІЇ** Запорізького національного університету: **Банах Віктор Аркадійович**



Електронна адреса: [y\\_banakh@znu.edu.ua](mailto:y_banakh@znu.edu.ua)

Гаряча лінія: тел. (061) 227-12-76, факс 227-12-88

### **РІВНІ МОЖЛИВОСТІ ТА ІНКЛЮЗИВНЕ ОСВІТНЄ СЕРЕДОВИЩЕ.**

Центральні входи усіх навчальних корпусів ЗНУ обладнані пандусами для забезпечення доступу осіб з інвалідністю та інших маломобільних груп населення. Допомога для здійснення входу у разі потреби надається черговими охоронцями навчальних корпусів. Спеціалізована допомога: (061) 228-75-11 (начальник охорони). Порядок супроводу (надання допомоги) осіб з інвалідністю та інших маломобільних груп населення у ЗНУ: <https://surl.li/ivcwiH>

### **РЕСУРСИ ДЛЯ НАВЧАННЯ**

**НАУКОВА БІБЛІОТЕКА:** <http://library.znu.edu.ua>. Графік роботи абонементів: понеділок-п`ятниця з 08.00 до 16.00; вихідні дні: субота і неділя.

**СИСТЕМА ЕЛЕКТРОННОГО ЗАБЕЗПЕЧЕННЯ НАВЧАННЯ ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ (СЕЗН ЗНУ):**  
<https://moodle.znu.edu.ua>.

Посилання для відновлення паролю:  
<https://moodle.znu.edu.ua/mod/page/view.php?id=133015>.

**ЦЕНТР ІНТЕНСИВНОГО ВИВЧЕННЯ ІНОЗЕМНИХ МОВ:**  
<http://sites.znu.edu.ua/child-advance/>