

**Силабус курсу**  
**Фронт інжиніринг**

**Ступінь вищої освіти – бакалавр**  
**Спеціальність 051 «Економіка»**  
**Освітньо-професійна програма «Економічна кібернетика»**

**Рік навчання: 4, Семестр: 7**

**Кількість кредитів: 5 Мова викладання: українська**

**Керівник курсу**

**ППП**

PhD, Семененко Юрій Сергійович

**Контактна  
інформація**

[y.semenenko@wunu.edu.ua](mailto:y.semenenko@wunu.edu.ua), +380973631487

**Опис дисципліни**

Дисципліна «Front Engineering» належить до блоку вибірових дисциплін циклу загальної підготовки студентів за спеціальністю Системний аналіз.

Дисципліна спрямована на формування у здобувачів освіти системного розуміння принципів розробки веб-інтерфейсів, включаючи адаптивний дизайн, інтерактивні елементи та інтеграцію Front-End із Back-End системами. Студенти отримують фундаментальні знання про основи HTML, CSS та JavaScript, засвоюють сучасні підходи до створення зручних та ефективних веб-додатків. Вивчення дисципліни також сприяє розвитку навичок роботи з сучасними фреймворками та інструментами для автоматизації і оптимізації процесів розробки, забезпечує розуміння принципів UI та UX дизайну.

Навчальна дисципліна орієнтована на підготовку здобувачів до розв'язання реальних задач Front-End розробки, забезпечуючи інтеграцію теоретичних знань із практичними навичками в галузі створення веб-інтерфейсів. Студенти оволодівають методами оптимізації продуктивності веб-додатків, кросбраузерної сумісності, а також принципами забезпечення доступності та безпеки веб-ресурсів.

## Структура курсу

Години (лек. / сем.)	Тема	Результати навчання	Завдання
2/2	<b>Тема 1.</b> Вступ до Front-End розробки, основні концепції та роль у веб-розробці	Студенти володіють теоретичними основами Front-End інженерії та розуміють її значення у створенні веб-додатків. Вони також усвідомлюють основні етапи Front-End розробки та взаємодію з Back-End	Тести, питання, виконання практичних завдань
2/2	<b>Тема 2.</b> HTML та основи структурування веб-сторінок	Студенти демонструють знання синтаксису та семантики HTML, а також розуміють принципи структурування веб-сторінок для створення структурованого та доступного контенту.	Тести, питання, виконання практичних завдань
2/2	<b>Тема 3.</b> CSS та стилізація веб-сторінок	Студенти здатні використовувати CSS для візуального оформлення веб-сторінок, розуміють методи застосування стилів і володіють базовими принципами стилізації елементів, таких як шрифти, кольори та відступи	Тести, питання, виконання практичних завдань
2/2	<b>Тема 4.</b> Адаптивний дизайн, використання медіа-запитів	Студенти опанували концепцію адаптивного дизайну та здатні використовувати медіа-запити для створення веб-сторінок, що підлаштовуються під різні розміри екранів, забезпечуючи кросплатформену сумісність	Тести, питання, виконання практичних завдань
2/2	<b>Тема 5.</b> Flexbox та Grid Layout, сучасні методи компоновання	Студенти знають, як використовувати Flexbox та Grid Layout для компоновання елементів на сторінці. Вони можуть створювати гнучкі та складні макети, розуміють принципи роботи з одновимірними та двовимірними макетами	Тести, питання, виконання практичних завдань
2/2	<b>Тема 6.</b> Основи JavaScript, інтерактивність на веб-сторінці	Студенти володіють основами JavaScript, розуміють його синтаксис і принципи роботи. Вони можуть застосовувати JavaScript для створення інтерактивних елементів на веб-сторінці, що покращує користувацький досвід	Тести, питання, виконання практичних завдань

2/2	<b>Тема 7.</b> Робота з DOM, маніпуляція елементами сторінки	Студенти демонструють здатність працювати з Document Object Model (DOM), здійснювати маніпуляції елементами веб-сторінок у реальному часі, використовуючи JavaScript для динамічного оновлення контенту	Тести, питання, виконання практичних завдань
2/2	<b>Тема 8.</b> Використання бібліотек та фреймворків, приклад jQuery	Студенти розуміють значення бібліотек і фреймворків у JavaScript. Вони можуть використовувати jQuery для спрощення роботи з DOM, обробки подій та створення анімацій, а також усвідомлюють переваги та недоліки таких рішень	Тести, питання, виконання практичних завдань
2/2	<b>Тема 9.</b> Інтерактивність і анімація, основи CSS анімацій та переходів	Студенти здатні створювати інтерактивність та анімації на веб-сторінках за допомогою CSS. Вони розуміють основи роботи з переходами і ключовими кадрами для створення плавних анімацій, що підвищують візуальний досвід користувачів	Тести, питання, виконання практичних завдань
2/2	<b>Тема 10.</b> Основи роботи з API та асинхронні запити	Студенти володіють основами роботи з API та розуміють принципи асинхронних запитів. Вони можуть використовувати Fetch API для виконання HTTP-запитів і відображення даних із зовнішніх серверів на веб-сторінці	Тести, питання, виконання практичних завдань
2/2	<b>Тема 11.</b> Основи UI/UX дизайну: розробка зручних інтерфейсів	Студенти розуміють основи UI/UX дизайну та здатні проектувати зручні інтерфейси, що відповідають потребам користувачів. Вони усвідомлюють роль Front-End розробника у процесі створення інтуїтивно зрозумілих та зручних інтерфейсів	Тести, питання, виконання практичних завдань
2/2	<b>Тема 12.</b> Принципи інклюзивного дизайну та доступності	Студенти володіють принципами інклюзивного дизайну та можуть забезпечити доступність веб-додатків для всіх користувачів, включаючи людей із фізичними обмеженнями, шляхом використання спеціальних технік та інструментів	Тести, питання, виконання практичних завдань

2/2	<b>Тема 13.</b> Тестування Front-End: забезпечення якості коду	Студенти знають методи тестування Front-End додатків для забезпечення їх якості. Вони здатні використовувати інструменти автоматизації тестування та усвідомлюють принципи відлагодження й профілювання коду	Тести, питання, виконання практичних завдань
2/2	<b>Тема 14.</b> Інструменти автоматизації та збірки, Webpack, Gulp	Студенти демонструють вміння використовувати інструменти автоматизації та збірки, такі як Webpack і Gulp, для оптимізації процесу розробки, включаючи збірку модулів JavaScript, мініфікацію коду та інші завдання	Тести, питання, виконання практичних завдань
2/2	<b>Тема 15.</b> Кросбраузерна сумісність, основи забезпечення	Студенти розуміють проблеми кросбраузерної сумісності та володіють методами їх вирішення. Вони можуть забезпечити однакову роботу веб-додатків у різних браузерах і знають, як проводити тестування у різних середовищах	Тести, питання, виконання практичних завдань
2/2	<b>Тема 16.</b> Тенденції у Front-End розробці, сучасні підходи та технології	Студенти ознайомлені з новітніми тенденціями у Front-End розробці. Вони здатні застосовувати сучасні технології, такі як прогресивні веб-додатки (PWA) та AMP, у реальних проєктах, що дозволяє їм залишатися в курсі нових підходів і рішень	Тести, питання, виконання практичних завдань

### РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

1. Marijn Haverbeke. Eloquent JavaScript, 3rd Edition: A Modern Introduction to Programming — No Starch Press, 2018. — 472 p. Also available online <https://eloquentjavascript.net/>
2. David Flanagan. JavaScript: The Definitive Guide: Master the World's Most-Used Programming Language. — 7th Edition — O'Reilly Media, 2020. — 706 p.
3. Aristeidis Bampakos, Pablo Deeelman. Learning Angular: A no-nonsense beginner's guide to building web applications with Angular 10 and TypeScript. — 3rd Edition — Packt Publishing, 2020. — 430 p.
4. Alex Banks, Eve Porcello. Learning React: Modern Patterns for Developing React Apps. — 2nd Edition O'Reilly Media, 2020. — 310 p.
5. Heitor Ramon Ribeiro. Vue.js 3 Cookbook: Discover actionable solutions for building modern web apps with the latest Vue features and TypeScript. — Packt Publishing , 2020. — 562 p.
6. Яковенко О. І. Управління проєктами та ризиками : навч. посібник. Ніжин : Видавець ПП Лисенко М.М., 2019. 196 с
7. Б.Г. Основи програмування: Навчальний посібник.- Кондор,2018.-364 с.

## Політика оцінювання

*Політика щодо дедлайнів і перескладання.* Для виконання індивідуальних завдань і проведення контрольних заходів встановлюються конкретні терміни. Перескладання модулів відбувається з дозволу дирекції факультету за наявності поважних причин (наприклад, тимчасова втрата працездатності).

*Політика щодо академічної доброчесності.* Використання друкованих і електронних джерел інформації під час контрольних заходів та екзаменів заборонено.

*Політика щодо відвідування.* Відвідування занять є обов'язковим компонентом оцінювання. За об'єктивних причин (наприклад, карантин, воєнний стан, хвороба, закордонне стажування) навчання може відбуватись в он-лайн формі за погодженням із керівником курсу з дозволу дирекції факультету.

Підсумковий бал (за 100-бальною шкалою) з дисципліни «Моделі та технології програмування» визначається як середньозважена величина, в залежності від питомої ваги кожної складової залікового кредиту:

Модуль 1		Модуль 2		Модуль 3	Модуль 4
20%	20%	20%	20%	5%	15%
Поточне опитування	Модульний контроль	Поточне опитування	Модульний контроль	Тренінги	Самостійна робота
Оцінка визначається із середнє арифметичне з отриманих оцінок за перший змістовий модуль. (1-6 заняття)	Виконання модульного завдання, яке складається з однієї практичної задачі. (теми 1-4)	Оцінка визначається із середнє арифметичне з отриманих оцінок за другий змістовий модуль. (7-16 заняття)	Виконання модульного завдання, яке складається з однієї практичної задачі (5-7 теми)	Оцінка за виконання завдання (презентації)	Оцінка за виконання самостійного завдання (презентація або звіт)

### Шкала оцінювання:

За шкалою ЗУНУ	За національною шкалою	За шкалою ECTS
90–100	відмінно	A (відмінно)
85–89	добре	B (дуже добре)
75-84		C (добре)
65-74	задовільно	D (задовільно)
60-64		E (достатньо)
35-59	незадовільно	FX (незадовільно з можливістю повторного складання)
1-34		F (незадовільно з обов'язковим повторним курсом)