

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ



РОБОЧА ПРОГРАМА
з дисципліни
"Front Engineering"

Ступінь вищої освіти: бакалавр

Галузь знань – 12 Інформаційні технології

Спеціальність – 124 Системний аналіз

Освітньо-професійна програма Системний аналіз

Кафедра економічної кібернетики та інформатики

Форма навчання	Курс	Семестр	Лекцій (год)	Практ (год.)	Інд.робота (год)	Тренінг (год)	Самост. роб. (год)	Разом (год)	Залік
Денна	4	7	32	14	3	6	95	150	7

30.08.2024
[Signature]

Тернопіль – ЗУНУ

2024

Робоча програма розроблена доктором філософії (PhD), старшим викладачем кафедри економічної кібернетики та інформатики Юрієм СЕМЕНЕНКОМ

Робоча програма затверджена на засіданні кафедри економічної кібернетики та інформатики, протокол № 1 від 28.08.2024 р.

Завідувач кафедри
д.е.н., професор



Леся БУЯК

Розглянуто та схвалено групою забезпечення спеціальності Системний аналіз, протокол № 1 від 30.08.2024 р.

Голова групи
забезпечення спеціальності
д.т.н., професор



Роман ПАСІЧНИК

Гарант ОП
д.т.н., професор



Роман ПАСІЧНИК

Структура робочої програми навчальної дисципліни

«Front engineering»

1. Опис дисципліни «Front engineering»

Дисципліна «Front engineering»	Галузь знань, спеціальність, СВО	Характеристика навчальної дисципліни
Кількість кредитів ECTS - 5	Галузь знань – 12 Інформаційні технології	Статус дисципліни: Вибіркова дисципліна Мова викладання: Українська
Кількість залікових модулів – 3	Спеціальність 124 Системний аналіз програма «Системний аналіз»	<i>Денна:</i> Рік підготовки: 4 Семестр: 7
Кількість змістових модулів – 2	Ступінь вищої освіти - Бакалавр	Лекції <i>Денна – 32 год.</i>
Загальна кількість годин - 150		Самостійна робота <i>Денна – 95 год.</i> Тренінг <i>Денна – 6 год.</i> Індивідуальна робота <i>Денна – 3 год.</i>
Тижневих годин – 10 З них аудиторних -3		Вид підсумкового контролю - залік

2. Мета та завдання дисципліни «Front engineering»

2.1. Мета вивчення дисципліни

Мета вивчення дисципліни «Front Engineering» полягає в тому, щоб забезпечити студентів теоретичними знаннями та практичними навичками, необхідними для розробки, проектування та підтримки сучасних веб-інтерфейсів. Це включає вивчення принципів створення користувацьких інтерфейсів, адаптивного дизайну, взаємодії з користувачами, а також інтеграції Front-End з Back-End системами.

Для досягнення цієї мети в процесі навчання поставлені такі основні завдання:

- Ознайомлення студентів з основами HTML, CSS, та JavaScript, як ключовими технологіями Front-End розробки.
- Вивчення сучасних підходів до адаптивного дизайну та створення інтерфейсів, орієнтованих на користувача.
- Набуття навичок у використанні інструментів та фреймворків для оптимізації та автоматизації розробки інтерфейсів.
- Формування здатності до аналізу і створення зручних, продуктивних та доступних веб-додатків.
- Розвиток практичних навичок у впровадженні інтерактивних елементів та забезпеченні кросбраузерної сумісності.
- Забезпечення розуміння взаємодії Front-End з Back-End і ролі Front-End розробника у процесі створення повноцінних веб-додатків.

2.2. Найменування та опис компетентностей, формування котрих забезпечує вивчення дисципліни «Front Engineering»:

- Вміння розробляти адаптивні та інтерактивні веб-інтерфейси.
- Вміння застосовувати сучасні технології та фреймворки для Front-End розробки.
- Вміння працювати з інструментами автоматизації та оптимізації коду.
- Вміння забезпечувати кросбраузерну сумісність та продуктивність веб-додатків.
- Вміння інтегрувати Front-End з Back-End системами.
- Вміння застосовувати принципи UI та UX дизайну.
- Вміння забезпечувати доступність веб-інтерфейсів для різних категорій користувачів.
- Вміння впроваджувати сучасні стандарти безпеки у веб-розробці.

2.3. Результати навчання дисципліни

В результаті навчання студенти отримають базові теоретичні знання та практичні навички у розробці сучасних веб-інтерфейсів. Вони зможуть створювати адаптивні та інтерактивні веб-сторінки, застосовувати сучасні технології та фреймворки Front-End розробки, забезпечувати продуктивність та кросбраузерну сумісність веб-додатків. Також студенти навчатимуться інтегрувати

Front-End з Back-End системами, застосовувати принципи UI та UX дизайну, та впроваджувати стандарти доступності й безпеки у веб-розробці.

2.4. Завдання вивчення дисципліни:

У результаті вивчення дисципліни «Front Engineering» студент повинен знати:

- Основні технології розробки веб-інтерфейсів, такі як HTML, CSS, та JavaScript.
- Принципи адаптивного дизайну та компоновання елементів за допомогою Flexbox і Grid Layout.
- Основи роботи з сучасними фреймворками Front-End розробки (наприклад, React, Angular, Vue.js).
- Принципи інтеграції Front-End з Back-End, включаючи роботу з API та асинхронні запити.
- Основи UI та UX дизайну, включаючи розробку зручних та доступних інтерфейсів.
- Підходи до забезпечення кросбраузерної сумісності та оптимізації продуктивності веб-додатків.
- Основи безпеки у веб-розробці, включаючи захист від поширених загроз.

2.5. Завдання лекційних занять

Мета проведення лекцій з дисципліни «Front Engineering» полягає у:

- Ознайомленні студентів з основними технологіями та підходами до розробки сучасних веб-інтерфейсів.
- Викладенні основних питань курсу, таких як адаптивний дизайн, інтерактивні елементи, та інтеграція Front-End з Back-End.
- Формуванні цілісної системи теоретичних знань про принципи створення зручних, продуктивних та доступних веб-додатків.

2.6. Завдання практичних занять

Мета проведення практичних занять з дисципліни «Front Engineering» полягає у:

- Оволодінні навичками створення адаптивних та інтерактивних веб-інтерфейсів.
- Розвитку навичок роботи з сучасними інструментами та фреймворками Front-End розробки.
- Отриманні практичного досвіду в інтеграції Front-End з Back-End системами.
- Застосуванні принципів UI та UX дизайну для розробки зручних та доступних веб-додатків.
- Формуванні вмінь оптимізувати продуктивність та забезпечувати кросбраузерну сумісність веб-додатків.

3. Зміст дисципліни «Front Engineering»

Змістовий модуль 1. Вступ до Front Engineering

Тема 1. Вступ до Front-End розробки, основні концепції та роль у веб-розробці

Визначення Front-End інженерії, її значення у створенні веб-додатків. Основні етапи розробки Front-End. Взаємодія з Back-End.

Тема 2. HTML та основи структурування веб-сторінок

Огляд основ HTML, його синтаксис і семантика. Структурування веб-сторінок за допомогою тегів. Роль HTML у створенні контенту.

Тема 3. CSS та стилізація веб-сторінок

Основи каскадних таблиць стилів (CSS), їх роль у візуальному оформленні веб-сторінок. Розгляд синтаксису CSS, різних способів застосування стилів (вбудовані, внутрішні, зовнішні таблиці стилів). Вивчення принципів стилізації елементів, включаючи кольори, шрифти, відступи, рамки та інші атрибути, що впливають на вигляд сторінки.

Тема 4. Адаптивний дизайн, використання медіа-запитів

Ознайомлення з концепцією адаптивного дизайну, який дозволяє створювати веб-сторінки, що автоматично підлаштовуються під різні розміри екранів. Вивчення медіа-запитів у CSS як засобу для реалізації адаптивного дизайну. Практичні приклади налаштування стилів для різних пристроїв, таких як мобільні телефони, планшети та десктопи.

Тема 5. Flexbox та Grid Layout, сучасні методи компоновання

Дослідження Flexbox та Grid Layout як інструментів для гнучкого компоновання елементів на веб-сторінці. Flexbox, принципи роботи з одновимірними макетами, управління напрямком, вирівнюванням і порядком елементів. Grid Layout, робота з двовимірними макетами, створення складних структур і позиціонування елементів в сітці. Порівняння обох підходів і їх застосування у різних сценаріях.

Тема 6. Основи JavaScript, інтерактивність на веб-сторінці

Введення у JavaScript як мову програмування, що додає динамічності та інтерактивності веб-сторінкам. Основи синтаксису JavaScript: змінні, типи даних, оператори, функції. Застосування JavaScript для створення інтерактивних елементів, таких як форми, кнопки, випадаючі меню, а також для реалізації базової логіки на стороні клієнта.

Тема 7. Робота з DOM, маніпуляція елементами сторінки

Ознайомлення з Document Object Model (DOM) як структурою, що представляє HTML-документ у вигляді дерева об'єктів. Вивчення методів JavaScript для маніпуляції DOM, додавання, видалення, зміна елементів на сторінці в

реальному часі. Практичні приклади обробки подій, таких як натискання на кнопку, зміна тексту, додавання класів і стилів до елементів.

Тема 8. Використання бібліотек та фреймворків, приклад jQuery

Огляд бібліотек та фреймворків для спрощення розробки на JavaScript. Використання jQuery як прикладу популярної бібліотеки для роботи з DOM, обробки подій, створення анімацій та AJAX-запитів. Переваги та недоліки використання бібліотек у порівнянні з чистим JavaScript.

Тема 9. Інтерактивність і анімація, основи CSS анімацій та переходів

Дослідження методів створення анімацій та переходів за допомогою CSS. Основи ключових кадрів (keyframes) та властивостей переходів (transitions). Створення плавних анімацій для покращення користувацького досвіду, включаючи ефекти при наведенні, натисканні та зміні станів елементів на сторінці.

Тема 10. Основи роботи з API та асинхронні запити

Вивчення концепції Application Programming Interface (API) як засобу для взаємодії між Front-End та зовнішніми сервісами. Основи роботи з Fetch API для виконання асинхронних HTTP-запитів та обробки відповіді. Створення прикладів використання API для отримання даних із серверів та їхнього відображення на веб-сторінці.

Тема 11. Основи UI/UX дизайну: розробка зручних інтерфейсів

Ознайомлення з поняттями User Interface (UI) та User Experience (UX) дизайну. Основи проектування інтерфейсів, орієнтованих на потреби користувачів. Принципи створення зручних та інтуїтивних інтерфейсів, що забезпечують позитивний користувацький досвід. Роль Front-End розробника у процесі розробки UI та UX.

Тема 12. Принципи інклюзивного дизайну та доступності

Вивчення принципів інклюзивного дизайну, що передбачає створення інтерфейсів, доступних для всіх користувачів, незалежно від їх фізичних можливостей. Ознайомлення з техніками та інструментами для забезпечення доступності веб-додатків, включаючи підтримку клавіатури, екранних читалок, кольорових контрастів та інших аспектів, важливих для користувачів з особливими потребами.

Тема 13. Тестування Front-End: забезпечення якості коду

Ознайомлення з методами тестування Front-End додатків для забезпечення їхньої якості та стабільності. Вивчення основних типів тестування, таких як юніт-тести, інтеграційні тести, тестування інтерфейсів. Використання інструментів для автоматизації тестування, а також принципів відлагодження та профілювання коду.

Тема 14. Інструменти автоматизації та збірки, Webpack, Gulp

Дослідження інструментів для автоматизації та оптимізації процесів розробки Front-End. Ознайомлення з Webpack як засобом для збірки модулів JavaScript та оптимізації ресурсів, таких як зображення та стилі. Використання Gulp для автоматизації завдань, таких як об'єднання файлів, мініфікація коду, та живе оновлення під час розробки.

Тема 15. Кросбраузерна сумісність, основи забезпечення

Вивчення проблем кросбраузерної сумісності, що виникають під час розробки веб-додатків для різних браузерів. Ознайомлення з методами перевірки та забезпечення однакового вигляду та поведінки додатків у різних середовищах. Використання інструментів для тестування у різних браузерах та рекомендації щодо найкращих практик кросбраузерної розробки.

Тема 16. Тенденції у Front-End розробці, сучасні підходи та технології

Огляд новітніх тенденцій та інновацій у сфері Front-End розробки. Вивчення сучасних підходів, таких як прогресивні веб-додатки (PWA), технології AMP, і розподілені інтерфейси. Ознайомлення з новими інструментами та фреймворками, що стають популярними в індустрії, і їхнє застосування у реальних проєктах.

4. Структура залікового кредиту з дисципліни

«Моделювання бізнес-процесів»

Тема	Кількість годин				
	Лекції	Практичні	Самост робота	Тренінг	Індивідуальна робота
Тема 1. Вступ до Front-End розробки, основні концепції та роль у веб-розробці	2	2	5	6	
Тема 2. HTML та основи структурування веб-сторінок	2		5		1
Тема 3. CSS та стилізація веб-сторінок	2	2	5		
Тема 4. Адаптивний дизайн, використання медіа-запитів	2		6		1
Тема 5. Flexbox та Grid Layout, сучасні методи компоновання	2	2	6		
Тема 6. Основи JavaScript, інтерактивність на веб-сторінці	2		6		1
Тема 7. Робота з DOM, маніпуляція елементами сторінки	2	2	6		
Тема 8. Використання бібліотек та фреймворків, приклад jQuery	2		6		
Тема 9. Інтерактивність і анімація, основи CSS анімацій та переходів	2	2	6		
Тема 10. Основи роботи з API та асинхронні запити	2		6		
Тема 11. Основи UI/UX дизайну: розробка зручних інтерфейсів	2	2	6		
Тема 12. Принципи інклюзивного дизайну та доступності	2		6		
Тема 13. Тестування Front-End: забезпечення якості коду	2		6		
Тема 14. Інструменти автоматизації та збірки, Webpack, Gulp	2		6		
Тема 15. Кросбраузерна сумісність, основи забезпечення	2	2	6		
Тема 16. Тенденції у Front-End розробці, сучасні підходи та технології	2		8		
Разом	32	14	95		3

5. Тематика практичних занять

Практичне заняття №1. Створення та структурування веб-сторінки за допомогою HTML

Практичне завдання №2. Стилзація веб-сторінки з використанням CSS: базові стилі, шрифти, кольори, відступи.

Практичне завдання №3. Адаптивний дизайн: створення веб-сторінки, що підлаштовується під різні розміри екранів за допомогою медіа-запитів.

Практичне завдання №4. Використання Flexbox для компонування елементів на веб-сторінці.

Практичне завдання №5 Застосування Grid Layout для створення складної сіткової структури на веб-сторінці.

Практичне завдання №6. Додавання інтерактивності на веб-сторінку за допомогою JavaScript: обробка подій і маніпуляція DOM.

Практичне завдання №7. Реалізація базових анімацій та переходів за допомогою CSS для покращення користувацького досвіду.

6. Самостійна робота

Самостійна робота з дисципліни «Front Engineering» виконується кожним студентом протягом семестру. Її виконання є однією із обов'язкових складових навчальної дисципліни.

Протягом семестру студентам потрібно розробити веб-сторінку, яка буде представляти їхнє персональне портфоліо. Мета завдання полягає в тому, щоб застосувати всі отримані знання з тем курсу (HTML, CSS, JavaScript, Flexbox, Grid, медіа-запити, інтерактивність) і створити функціональний і візуально привабливий веб-сайт.

- Сайт повинен мати принаймні такі розділи: головна сторінка, про себе, проекти, навички, контакти. Використати семантичні теги HTML для побудови логічної структури сторінки.

Адаптивний дизайн:

- Використати Flexbox і Grid Layout для побудови макету.
- Використати медіа-запити для адаптації сторінки під різні розміри екранів (мобільні пристрої, планшети, десктопи).

Стилізація:

- Використати CSS для оформлення сторінки: кольорові схеми, шрифти, відступи, анімації.
- Забезпечити кросбраузерну підтримку.

Інтерактивність:

- Додати прості елементи JavaScript для створення інтерактивності (наприклад, форму зворотного зв'язку, динамічний контент, навігацію, яка приховується на мобільних пристроях тощо).

Оптимізація:

- Забезпечити швидке завантаження сторінки та оптимізувати зображення.
- Перевірити сайт на відповідність принципам доступності (інклюзивний дизайн).

Контактна форма:

- Додати просту форму для контакту з полями (ім'я, електронна пошта, повідомлення), яка надсилає дані (в рамках завдання це може бути простий макет без бекенду).

Наприкінці семестру студенти повинні представити своє портфоліо, яке відповідає зазначеним вимогам. Вони також повинні написати короткий опис того, як кожен з розділів курсу був застосований у їхньому проекті.

7. Організація та проведення тренінгу

Тематика: Розробка інтерактивних та адаптивних веб-додатків, від концепції до реалізації

Тренінг охоплює основні етапи створення сучасних веб-додатків з акцентом на інтерактивність та адаптивність. Студенти починають з концептуального проєктування користувацьких інтерфейсів, де визначаються ключові вимоги до зручності та доступності. На наступних етапах тренінг зосереджується на розробці за допомогою HTML, CSS, JavaScript, з використанням Flexbox і Grid Layout для побудови адаптивних макетів. Особлива увага приділяється інтерактивності, де студенти вчаться додавати динамічні елементи та взаємодії за допомогою JavaScript. Завершується тренінг інтеграцією додатків із зовнішніми API та забезпеченням кросбраузерної сумісності, що гарантує безперебійну роботу на різних пристроях.

8. Методи навчання

У навчальному процесі застосовуються: лекції, практичні та індивідуальні заняття, консультації, самостійна робота, метод опитування, підготовка і презентація тренінгу.

9. Методи оцінювання

В процесі вивчення дисципліни використовуються наступні методи оцінювання навчальної роботи студента:

- поточне опитування;
- підсумкове тестування по кожному змістовому модулю;
- залік.

10. Критерії, форми поточного та підсумкового контролю

Evaluation policy

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 можливістюповторного складання)
1-34		F (незадовільно з обов'язковимповторним курсом)

11. Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

№	Найменування	Номер теми
1.	Персональний компютер та інтернет	1-7

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 Deeleman. 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.