

Чернівецький національний університет імені Юрія Федьковича

(повне найменування закладу вищої освіти)

факультет математики та інформатики

(назва факультету/навчально-наукового інституту)

Кафедра

математичного моделювання

(назва кафедри)

“ЗАТВЕРДЖУЮ”

Декан

“ 25 ”

Мартинюк О.В.
2024 року



РОБОЧА ПРОГРАМА

навчальної дисципліни

Сучасні клієнтські веб-технології

(вказіть назву навчальної дисципліни (іноземною, якщо дисципліна викладається іноземною мовою))

вибіркова

(вказати: обов'язкова чи вибіркова)

Освітньо-професійна програма Системний аналіз

(назва програми)

Спеціальність 124 Системний аналіз

(вказати: код, назва)

Галузь знань 12 Інформаційні технології

(вказати: шифр, назва)

Рівень вищої освіти другий (магістерський)

(вказати: перший бакалаврський/другий магістерський)

факультет математики та інформатики

(назва факультету/навчально-наукового інституту, на якому здійснюється підготовка фахівців за вказаною освітньо-професійною програмою)

Мова навчання українська

(вказати: якою мовою викладається дисципліна)

Чернівці 2024 рік

Робоча програма навчальної дисципліни «Сучасні клієнтські веб-технології» складена відповідно до освітньо-професійної програми «Системний аналіз», затвердженої Вченою радою Чернівецького національного університету імені Юрія Федьковича « 29 » травня 2023 року, протокол № 5.

Розробники: Мельник Василь Сергійович, асистент, канд. фіз.-мат. наук

Погоджено з гарантом ОП і затверджено на засіданні кафедри математичного моделювання

Протокол № 18 від « 25 » червня 2024 року

Завідувач кафедри Черевко І.М.
(підпис) (прізвище та ініціали)

Схвалено методичною радою факультету математики та інформатики

Протокол № 11 від «25» червня 2024 року

Голова методичної ради факультету
математики та інформатики Сікора В.С.
(підпис) (прізвище та ініціали)

1. Мета навчальної дисципліни: Головною метою курсу “Сучасні клієнтські веб-технології” є вивчення основних прийомів і методів фулл-стак розробки, конструювання веб-аплікацій, освоєння важливих патернів розробки і ознайомлення з сучасними бібліотеками для розробки.

Цей курс розроблений з метою надання студентам глибокого розуміння та практичних навичок у сфері клієнтського веб-програмування. Студенти зможуть ознайомитися з останніми тенденціями у веб-розробці, включаючи HTML5, CSS3, та JavaScript ES6+. Студенти будуть навчатися створювати веб-додатки та інтерактивні інтерфейси з використанням сучасних клієнтських технологій.

Курс також охоплює адаптивний та мобільний дизайн, що є важливим аспектом сучасних веб-технологій. Курс надає можливість студентам реалізовувати власні проекти та вдосконалювати їхні навички веб-розробника.

2. Результати навчання: У результаті вивчення дисципліни студент повинен знати:

- Основні аспекти Full-Stack розробки;
- Основні патерни розробки;
- Найновіші клієнт-серверні технології;

У результаті вивчення дисципліни фахівець повинен вміти:

- Створювати веб-аплікації;
- Вміти проводити якісний рефакторинг існуючих аплікацій;
- Працювати з системою контролю версій git;

Компетентності за ОП:

- ЗК2. Здатність спілкуватися іноземною мовою.
- ЗК3. Здатність вчитися та оволодівати сучасними знаннями
- ЗК8. Здатність розробляти проекти та управляти ними.

СК:

- СК2. Здатність проектувати архітектуру інформаційних систем
- СК7. Здатність управляти робочими процесами у сфері інформаційних технологій, які є складними, непередбачуваними та потребують нових стратегічних підходів
- СК10. Здатність до самоосвіти та професійного розвитку.

РН:

- РН1. Спеціалізовані концептуальні знання, що включають сучасні наукові здобутки у сфері системного аналізу та інформаційних технологій і є основою для оригінального мислення та проведення досліджень.
- РН7. Розробляти інтелектуальні системи в умовах слабо структурованих даних різної природи.
- РН10. Зрозуміло і недвозначно доносити власні знання, висновки та аргументацію до фахівців і нефахівців, зокрема до осіб, які навчаються

- РН11. Вільно презентувати та обговорювати усно і письмово результати досліджень та інновацій, інші питання професійної діяльності державною та англійською мовами.

3. Опис навчальної дисципліни

3.1. Загальна інформація

Форма навчання	Рік підготовки	Семестр	Кількість		Кількість годин						Вид підсумкового контролю
			кредитів	годин	лекції	практичні	семінарські	лабораторні	самостійна робота	індивідуальні завдання	
Денна	1	2	3	90			15	15	60		залік

3.2. Структура змісту навчальної дисципліни

Назви змістових модулів і тем навчальних занять	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		лекц.	прак./семін.	лаб.	інд.	с.р.		лекц.	прак./сем.	лаб.	інд.	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Теми навчальних занять	Змістовий модуль 1. (Front-End and Back-End development)											
1. React basics. JSX, components, props, state.	4			1	1	2						
2. Components lifecycle. React Hooks. State management (local vs global, lifting state, useReducer).	4			1	1	2						
3: CSS preprocessors: Sass, Less. Variables, nesting, mixings, modular css.	4			1	1	2						
4: .Net core 8, ASP.NET and Blazor. Components in Blazor.	4			1	1	2						
5: Blazor routing and navigation, data binding, event handling. Dependency injection in Blazor. Forms and validation.	4			1	1	2						

6: .Net core 8, ASP.NET and Blazor;	4			1	1	2						
7: Blazor state management, components communication. JSinterop.	4			1	1	2						
8: Connecting frontend app to the backend, fetch and axios. Connecting application to the database. Error handling.	2			1	1	1						
Разом за 3М1	30			8	8	15						
Теми навчальних занять	Змістовий модуль 2. (Client-server technologies and best practices)											
9: REST and Api design.	4			1	1	1						
10: GraphQL basics, queries and mutations. GraphQL subscriptions.	4			1	1	2						
11: Authentication, JWT and OAuth2.0	4			1	1	2						
12: WebSockets and WebHooks. Integration with WebSocket servers.	4			1	1	2						
13: XSS, CSRF, CORS, security and OWASP top 10.	4			1	1	2						
14: Web performance optimization, lazy loading.	4			1	1	2						
15: Unit and integration testing. Tools for unit testing web applications.	4			1	1	2						
Разом за 3М 2	30			7	7	15						
Усього годин	60			15	15	30						

3.3. Тематика лабораторних занять

№	Назва теми	Кількість годин
1	Front-End application development	3
2	Backend application development	4
3	Rest Api and GraphQL. Authentication via Oauth2.	4
4	Unit tests.	4

3.4. Самостійна робота студента

№	Назва теми	кількість годин
1	React Hooks in Depth, Context API for Global State Management, React Router: Navigation in SPAs, State Management with Redux	6
2	Blazor WebAssembly and Progressive Web Apps, Service workers, caching strategies, and offline support; Blazor and SignalR for Real-Time Communication	8
3	Performance optimization techniques	8
4	Architecture patterns, GoF, SOLID	8

3.5. Освітні технології, методи навчання і викладання навчальної дисципліни

Методи навчання та викладання: лабораторні та семінарські заняття, електронне навчання з використанням системи Moodle, тестування, виконання завдань самостійної роботи.

3.6. Критерії та засоби оцінювання результатів навчання з навчальної дисципліни

5.1. Критерієм підсумкового оцінювання має бути досягнення студентом мінімальних порогових рівнів оцінок (балів) за кожним передбаченим результатом навчання.

Система оцінювання рівня навчальних досягнень ґрунтується на принципах ECTS та є накопичувальною. Протягом семестру студенти виконують 4 лабораторних робіт. Кожна лабораторна робота оцінюється в 15 балів. Виконуючи завдання лабораторної роботи, студент повинен оформити і завантажити для подальшої перевірки на сайт електронного навчання звіт разом із працездатними файлами програмної реалізації завдань ЛР (правила оформлення наведені на сторінці навчальної дисципліни на сайті). 50% балів, відведених на оцінювання ЛР, студент отримує за працюючий програмний продукт, в якому реалізовано всі завдання ЛР та оформлений звіт. Решта 50% балів виставляється після захисту студентом виконаного звіту. На захисті звіту з ЛР студент має відповісти на питання щодо постановки задачі та розробленого ним алгоритму реалізації кожного із завдань ЛР. При відповіді на теоретичні питання та питання щодо програмної реалізації алгоритму у випадку неістотної помилки знімається 10-20% балів, а у випадку істотної 20-40% балів, якщо ж студент не опанував теоретичний матеріал, плутається в означеннях, наводить логічно невірні твердження, то знімається до 50% балів від усієї суми балів за ЛР. Максимальна кількість, яку можна набрати на підсумковому модулі (тестування) – 40 балів. Підсумкова оцінка виставляється за результатами суми балів, набраних на змістових модулях під час семестру та підсумковому модулі згідно з нижченаведеною таблицею.

5.2. Шкала оцінювання: національна та ECTS

Оцінка за національною шкалою	Оцінка за шкалою ECTS	
	Оцінка (бали)	Пояснення за розширеною шкалою
Зараховано	A (90-100)	відмінно
	B (80-89)	дуже добре
	C (70-79)	добре
	D (60-69)	задовільно
	E (50-59)	достатньо
Не зараховано	FX (35-49)	(незадовільно) з можливістю повторного складання
	F (1-34)	(незадовільно) з обов'язковим повторним курсом

5.3. Засоби оцінювання

Засобами оцінювання та демонстрування результатів навчання є
- лабораторні роботи; стандартизовані тести; проекти.

6. Форми поточного та підсумкового контролю

Формою підсумкового контролю є залік.

7. Рекомендована література

1. Martin R. C. Design Principles and Design Patterns. Addison-Wesley Professional; 1 edition, 2000.
2. Burns B. Designing Distributed Systems: Patterns and Paradigms for Scalable, Reliable Services. O'Reilly, 2017. 166 p.
3. OWASP top 10. URL: <https://owasp.org/www-project-top-ten/>.
4. Peter Himschoot Microsoft Blazor: Building Web Applications in .NET 6 and Beyond. Third Edition, Melle, Belgium
5. Designing Interfaces: Patterns for Effective Interaction Design 3rd Edition. J. Tidwell, Ch. Brewer.
6. Web Application Architecture: Principles, Protocols and Practices by Shklar, Leon Published by Wiley 2nd (second) edition (2009)
7. Porcello E., Banks A. Learning GraphQL: Declarative Data Fetching for Modern Web Apps.