

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інформаційних систем та технологій



«ЗАТВЕРДЖУЮ»

Заступник декана

з навчально-виховної роботи

Наталія ТМЄНОВА

_____ 2025 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Технології програмування

для студентів

галузі знань
спеціальності
освітнього ступеня
освітньої програми
вид дисципліни

12 Інформаційні технології
126 Інформаційні системи та технології
бакалавр
Програмні технології інтернет речей
обов'язкова

Форма навчання	денна
Навчальний рік	2025/2026
Семестр	3
Кількість кредитів ECTS	5
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладачі: **Пирог Микола Володимирович**, асистент
Бондаренко Ольга Сергіївна, асистент

Пролонговано: на 20__/20__ н.р. _____ (_____) «__» 20__ р.
на 20__/20__ н.р. _____ (_____) «__» 20__ р.

КИЇВ – 2025

Розробник: **Пирог Микола Володимирович**, асистент кафедри
інформаційних систем та технологій

ЗАТВЕРДЖЕНО

Завідувач кафедри

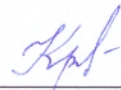
інформаційних систем та технологій

 проф. Володимир ДРУЖИНІН
(підпис)

Протокол №17_24/25 від 25 червня 2025 р

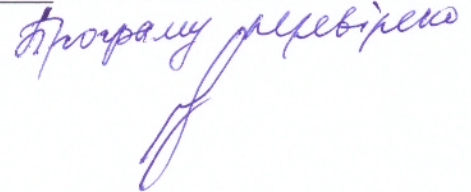
Схвалено науково-методичною комісією факультету інформаційних технологій
Протокол від «27» червня 2025 року № 10

Голова науково-методичної комісії



(підпис)

Ганна КРАСОВСЬКА



« ____ » _____ 2025 року

ВСТУП

1. Мета дисципліни полягає у формуванні у студента знань, вмінь і навичок з використання новітніх методів та інструментів для розроблення вебсайтів, розуміння структури та етапів створення вебсторінок, освоєння засобів інформаційних технологій щодо реалізації вебпроектів. Предметом навчальної дисципліни є теорія і практика застосування сучасних вебтехнологій для розроблення вебсайтів та застосунків.

2. Попередні вимоги до опанування або вибору навчальної дисципліни: успішне опанування дисциплін: «Основи програмування», «Веб-технології», «Алгоритми та структури даних».

3. Анотація навчальної дисципліни:

Програма дисципліни охоплює вивчення теоретичних основ розроблення вебсайтів, аналіз типів і категорій вебресурсів, динамічного програмування за допомогою мов JavaScript та TypeScript для вебсередовища. Також розглянуто методи оптимізації та підвищення продуктивності вебсайтів.

4. Завдання вивчення дисципліни полягає у формуванні у студентів вмінь і навичок з основ побудови вебсторінок з використанням стеку технологій HTML/CSS/JavaScript, формування вміння застосовувати ці знання в професійній діяльності, зокрема для розроблення сучасних веб-систем.

5. Результати навчання за дисципліною:

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
1.1	Знати основні типи вебсайтів і їх класифікацію, методології використання вебтехнологій для створення клієнт-серверних систем	Лекції	Модульна контрольна робота, іспит	15 %
1.2	Знати теорію побудови вебсистем, основи формування структури та взаємозв'язків їх компонент	Лекції	Модульна контрольна робота, іспит	15 %
2.1	Вміти проектувати та розробляти логіку алгоритмів мовою JavaScript	практичні заняття, лабораторні заняття	Виконання та захист практичних робіт	20 %
2.2	Вміти розробляти алгоритми оброблення вебсторінок і динамічного управління HTML-елементами та формами засобами мови JavaScript	практичні заняття, лабораторні заняття	Виконання та захист практичних робіт	15 %
2.3	Вміти перевіряти працездатність розроблених програм	практичні заняття, лабораторні заняття	Виконання та захист практичних робіт	10%
2.4	Вміти користуватися довідковою системою, бібліотечними функціями і структурами, що входять до складу обраних IDE	практичні заняття, лабораторні заняття	Виконання та захист практичних робіт	10%
3.1	Проведення дискусійного пошуку оптимальних рішень під час проектування інформаційних систем щодо поставлених задач.	лекція, практичні заняття, лабораторні заняття	Обговорення результатів дискусії	10%
4.1	Проявляти самостійність у здійсненні узагальнень, прийняття рішень і виконання самостійних дій при розробленні вебсистем	практичні заняття, лабораторні заняття	Виконання та захист практичних робіт	5%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	1.1	1.2	2.1	2.2	2.3	2.4	3.1	4.1
	1.1	1.2	2.1	2.2	2.3	2.4	3.1	4.1
Програмні результати навчання								
ПР 3. Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій.	+		+	+		+	+	+
ПР5. Аргументувати вибір програмних та технічних засобів для створення інформаційних систем та технологій на основі аналізу їх властивостей, призначення і технічних характеристик з урахуванням вимог до системи і експлуатаційних умов; мати навички налагодження та тестування програмних і технічних засобів інформаційних систем та технологій.		+	+		+			
ПР6. Демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп'ютерних систем та середовищ з метою їх запровадження у професійній діяльності.		+	+			+	+	+
ПР 16. Будувати ефективні обчислювальні алгоритми для розрахункових задач, визначати ефективність програм за допомогою програмного забезпечення комп'ютерів, розробляти комплексні інформаційні рішення для підприємств та фірм, включаючи проектування пристроїв Інтернету речей, здатність розробляти програмні продукти для процесів, які комп'ютеризуються, здатність використовувати можливості апаратного та програмного забезпечень для вирішення науково-технічних та практичних задач.								
ПР 17. Планувати та управляти проектами для розробки програмного забезпечення (в тому числі для пристроїв Інтернету речей) з урахуванням вимог клієнту обмежень по часу та ресурсах, здатність критично оцінювати і аналізувати складні проблеми, в тому числі за умов неповної інформації, приймати відповідні рішення при наявності обмежених ресурсів, виявлення та аналіз вимог до програмного забезпечення.								
ПР 19. Використовувати можливості апаратного забезпечення, використовувати можливості операційних систем, використовувати можливості мережних програмних систем, забезпечувати захищеність програм і даних від несанкціонованих дій.								
ПР 20. Документувати програмне забезпечення з дотриманням норм та стандартів.								

7. Схема формування оцінки.

7.1 Форми оцінювання студентів:

Рівень досягнення всіх запланованих результатів навчання визначається за результатами написання письмових контрольних робіт, виконання лабораторних та самостійних ї робіт. Дисципліна закінчується іспитом.

Питома вага результатів навчання у підсумковій оцінці за умови її опанування на належному рівні така:

результати навчання 1-(знання) РН 1.1, РН 1.2 - до 30%;

результати навчання 2-(вміння) РН 1.2., РН 1.4., РН 2.3.- до 55% ;

результати навчання 3-(комунікація) РН 3 - до 10%;

результати навчання 4-(автономність) РН 4 - до 5%.

7.2 Організація оцінювання:

Семестрове оцінювання: 60 балів (100%)

Оцінка за семестр формується шляхом успішного виконання та захисту лабораторних та практичних робіт, успішного написання модульної контрольної роботи.

Обов'язковим для проміжного модульного контролю є виконання та захист лабораторних та практичних робіт, передбачених робочою програмою навчальної дисципліни у визначені терміни.

Студенти виконують 6 лабораторних робіт та 6 практичних робіт, що оцінюються від 2,4 до 4 балів за роботу (сумарна кількість балів від 28 до 48 балів за лабораторні та практичні роботи). Якщо здобувач не вчасно виконує практичну роботу, то оцінка знижується на 1 бал за кожен тиждень протермінування.

Робота вважається зданою вчасно, коли захист відбувається на наступній парі після видачі завдання, якщо викладачем не повідомлено інші терміни. Терміни здачі викладач оголошує та викладає окремим повідомленням в класі системи Microsoft Teams (чи інших затверджених кафедрою системах навчання).

Робота вважається зданою та оціненою, якщо студент її захистив на занятті та завантажив звіт та архів з програмним кодом у відповідне завдання у Тімс та GitHub. За одне заняття приймається не більше 1 лабораторна робота, виняток для тих студентів, хто виконав завдання наперед та черга на захист робіт дозволяє здати всім охочим студентам групи.

Упродовж семестру, після завершення відповідних тем, проводиться письмова модульна контрольна робота у вигляді практичного завдання. Модульна контрольна робота оцінюється від 8 до 12 балів.

У випадку відсутності студента з поважних причин відпрацювання та перездачі здійснюються у відповідності до «Положення про порядок оцінювання знань студентів при кредитно-модульній системі організації навчального процесу» від 1 жовтня 2010 року.

Загальна сума балів набраних протягом семестру не перевищує 60 балів.

Рекомендований мінімум для допуску до іспиту – 36 балів, критично розрахунковий мінімум – 20 балів. При цьому обов'язковим є виконання всіх

лабораторних робіт та практичних робіт, а також отримання позитивної оцінки з контрольної роботи.

Для студентів, які упродовж семестру не набрали рекомендований мінімум обов'язковим є виконання запланованих лабораторних та практичних робіт, а також написання комплексної семестрової контрольної роботи, яка включає увесь пройдений матеріал за семестр і максимальна оцінка за яку не може перевищувати 40% підсумкової оцінки (до 40 балів за 100 – бальною шкалою).

Підсумкове оцінювання у формі іспиту: складає 40 балів (40% від загального рейтингу).

Студент не допускається до іспиту, якщо під час семестру набрав менше 36 балів. На іспиті студент отримує екзаменаційне завдання, що містить тест для перевірки теоретичних знань та практичну задачу. Тест і задача оцінюються по 20 балів максимально. Отже, сумарно на іспиті студент може набрати до 40 балів максимально. Якщо студент набрав менше 24 балів за іспит, то йому ставиться оцінка «незадовільно», після чого він має право на перескладання іспиту за встановленою процедурою в Київському національному університеті імені Тараса Шевченка.

Загальна оцінка за семестр не перевищує 100 балів.

7.3 Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59

8. Структура навчальної дисципліни.

Тематичний план лекційних занять, практичних та самостійних робіт

№ п/п	Номер і назва теми*	Кількість годин			
		лекції	прак тичні	лабора торні	Самості йна робота
Змістовий модуль 1. Основи розробки веб-сайтів, мова JavaScript					
1	Основи побудови веб-систем, поняття документації коду	2	-	-	6
2	JavaScript. Basics	2	-	6	6
3	JavaScript. Data types, typeof, type casting, arrays, date, object	2	-	6	6
4	JavaScript. Функції та методи функцій.	2	-	6	8
5	JavaScript. Поняття та керування DOM. Програмування та оброблення форм.	2	-	6	8
6	Використання JavaScript бібліотек.	2	-	8	8
	Модульна контрольна робота 1	-	2	-	-
Змістовий модуль 2. Сучасні технології веброзробки, TypeScript					
6	Основи мови TypeScript.	2	2		8
7	TypeScript – інтерфейси, класи, модулі.	2	2		8
8	Компоненти та їх життєвий цикл.	2	2		8
9	Проектування сучасних веб застосунків	2	2		8
10	Основи тестування застосунків.	2	4		4
	Модульна контрольна робота 2	-	2	-	-
	ВСЬОГО	22	16	32	78

Загальний обсяг 150 год, у тому числі:

Лекцій: **22 год.**

Практичні заняття: **16 год.**

Лабораторна робота: **32 год.**

Самостійна робота: **100 год.**

9. Рекомендовані джерела

Основні:

1. Фрімен Е., Робсон Е. Head First. Програмування на JavaScript / пер. з англ. Г. Якубовська. Харків : ВД "Фабула", 2022. 672 с.
2. Frisbie M. Professional JavaScript for Web Developers. Wiley & Sons, Incorporated, John, 2023. 1200 p.
3. Simpson J. How JavaScript Works. Berkeley, CA : Apress, 2023. URL: <https://doi.org/10.1007/978-1-4842-9738-4>.
4. Simon M. JavaScript for Web Developers. Berkeley, CA : Apress, 2023. URL: <https://doi.org/10.1007/978-1-4842-9774-2>.
5. Nicoara R. How to be a Web Developer. Berkeley, CA : Apress, 2023. URL: <https://doi.org/10.1007/978-1-4842-9663-9>.

Додаткові:

1. Сучасний підручник з JavaScript. URL: <https://uk.javascript.info/>.
2. MDN Web Docs. URL: <https://developer.mozilla.org/en-US/>.
3. W3Schools.com. W3Schools Online Web Tutorials. URL: <https://www.w3schools.com/>.
4. Larson Q. freeCodeCamp. URL: <https://www.freecodecamp.org/>.
5. ECMA-262 - Ecma International. URL: <https://ecma-international.org/publications-and-standards/standards/ecma-262/>.
6. Node.js – Run JavaScript Everywhere. URL: <https://nodejs.org/en>.
7. npm. URL: <https://www.npmjs.com/>.
8. Yarn. URL: <https://yarnpkg.com/>.
9. Vite. URL: <https://vite.dev/>.