

Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Кафедра прикладної математики

“ЗАТВЕРДЖУЮ”

Декан факультету  
математики і інформатики

Григорій ЖОЛТКЕВИЧ

“ 29 ” 08

2023 р.



## РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### Основи web-програмування

(назва навчальної дисципліни)

рівень вищої освіти перший (бакалаврський)

галузь знань 11- Математика та статистика

спеціальність 113 – Прикладна математика  
(шифр і назва)

освітня програма прикладна математика  
(шифр і назва)

спеціалізація \_\_\_\_\_  
(шифр і назва)

вид дисципліни за вибором  
(обов'язкова / за вибором)

факультет математики і інформатики

2023 / 2024 навчальний рік

Програму рекомендовано до затвердження вченою радою факультету математики і інформатики

“29” серпня 2023 року, протокол № 8

РОЗРОБНИКИ ПРОГРАМИ: **Бєбія Максим Отарійович**, кандидат фіз.-мат. наук, доцент кафедри прикладної математики

Програму рекомендовано до затвердження вченою радою факультету математики і інформатики на засіданні кафедри прикладної математики

“28” серпня 2023 року, протокол № 10

Завідувач кафедри Прикладної математики



(підпис)

Валерій КОРОБОВ

(ім'я та прізвище)

Програму погоджено з гарантом освітньо-професійної програми (керівником проектної групи) Прикладна математика

назва освітньої програми

Гарант освітньо-професійної програми

(керівник проектної групи) Прикладна математика



(підпис)

Сергій ПОСЛАВСЬКИЙ

(ім'я та прізвище)

Програму погоджено науково-методичною комісією факультету математики і інформатики Харківського національного університету імені В.Н. Каразіна

назва факультету, для здобувачів вищої освіти якого викладається навчальна дисципліна

Протокол від “29” серпня 2023 року, протокол № 1

Голова науково-методичної комісії факультету математики і інформатики



(підпис)

Ольга АНОЩЕНКО

(ім'я та прізвище)

## ВСТУП

Програма навчальної дисципліни **“Основи web-програмування”** складена відповідно до освітньо-професійної програми підготовки

бакалавр  
(назва рівня вищої освіти)

Спеціальності (напряму) 113 - Прикладна математика

спеціалізації \_\_\_\_\_

### 1. Опис навчальної дисципліни

1.1. Мета викладання навчальної дисципліни “Основи web-програмування” полягає у формуванні у слухачів фундаментальних знань та навичок, необхідних для створення web-сайтів і web-додатків. А також – у створенні основ щодо подальшого більш поглибленого вивчення інформаційних технологій.

1.2. Основні завдання вивчення дисципліни “Основи web-програмування” полягають у оволодінні найпоширенішими технологіями web-програмування, що необхідні для створення фронтенду та бекенду.

1.3. Кількість кредитів – 4

1.4. Загальна кількість годин 120

1.5. Характеристика навчальної дисципліни	
За вибором	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
3-й	–
Семестр	
5-й	–
Лекції	
32 год.	–
Практичні, семінарські заняття	
22 год.	–
Лабораторні заняття	
год.	–
Самостійна робота	
66 год.	–
у тому числі індивідуальні завдання	
–	–

1.6. Заплановані результати навчання

У результаті вивчення даного курсу студент повинен

**знати:**

- структуру та синтаксис HTML коду;
- основи CSS та адаптивної верстки;
- основні елементи мови JavaScript;
- способи маніпуляції DOM за допомогою JavaScript;
- як взаємодіяти з API;

- основи розробки веб-серверів та обробки запитів;

**вміти:**

- створювати web-сторінки за допомогою HTML;
- стилізувати сторінки за допомогою CSS;
- реалізовувати інтерактивність web-сторінок за допомогою JavaScript;
- використовувати мову JavaScript для запитів до API;
- реалізовувати взаємодію між клієнтом та сервером;
- працювати з базами даних;
- використовувати інструменти для відладки web-додатків;

## 2. Тематичний план навчальної дисципліни

### **Розділ 1. Введення у веб-розробку та основи оформлення web-сторінок.**

#### *Тема 1.* Верстка та оформлення web-сторінок.

Загальний огляд технологій клієнтської сторони веб-додатку. Основні елементи (теги) мови розмітки HTML (заголовки, параграфи, таблиці, списки, посилання, зображення, форми введення даних). Створення простих web-сторінок. Стилізування елементів за допомогою каскадних таблиць стилів CSS (кольори, шрифти, розташування, анімація і ін.). Адаптивний дизайн для оптимізації вигляду на різних пристроях.

#### *Тема 2.* Використання мови JavaScript для надання інтерактивності web-сторінкам.

Введення у мову програмування JavaScript. Типи змінних, оператори, цикли, функції, робота зі строками у JavaScript. Робота з масивами та об'єктами. Об'єктна модель документа DOM (document object model). Маніпулювання веб-документом за допомогою JavaScript: пошук потрібного елемента, отримання чергового дочірнього елемента, додавання та видалення елементів, зміна стилю елементів.

### **Розділ 2. Введення у розробку серверної частини web-додатку.**

#### *Тема 3.* Взаємодія із серверною частиною web-додатку.

Протокол HTTP і методи GET, POST, PUT, DELETE. Взаємодія з RESTful API для обміну даними між клієнтом і сервером. Використання інструментів веб-розробника для відладки коду.

#### *Тема 4.* Основи створення серверної частини web-додатків.

Огляд роботи серверної частини web-додатку. Створення серверів і обробка запитів (за допомогою Node.js ). Основи роботи з реляційними та NoSQL базами даних. Мова структурованих запитів до баз даних SQL. Взаємодія з базами даних на стороні сервера. Створення з'єднання з базою даних, виконання CRUD операції (Create, Read, Update, Delete).

## 3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб.	інд.	с. р.		л	п	лаб.	інд.	с. р.
1	2	3	4	5	6	7	8	9	10	11	12	13
<b>Розділ 1. Введення у веб-розробку та основи оформлення web-сторінок.</b>												
Тема 1. Верстка та оформлення web-сторінок.	28	8	4			16						

Тема 2. Використання мови JavaScript для надання інтерактивності web- сторінкам.	31	8	6			17						
Разом за розділом 1	59	16	10			33						
<b>Розділ 2. Введення у розробку серверної частини web-додатку.</b>												
Тема 3. Взаємодія із серверною частиною web-додатку	28	8	4			16						
Тема 4. Основи створення серверної частини web- додатків.	31	8	6			17						
<i>Контрольна робота</i>	2		2									
Разом за розділом 2	61	16	12			33						
<b>Усього годин</b>	<b>120</b>	<b>32</b>	<b>22</b>			<b>66</b>						

#### 4. Темі семінарських (практичних, лабораторних) занять

№ з/п	Назва теми	Кількість годин
1	Основні елементи (теги) мови розмітки HTML, а саме <h1> - <h6>, <p>, <a>, <img>, <div>, <span>. Атрибути елементів. Списки і таблиці (теги <ul>, <ol>, <li>, <table>). Елементи форми – теги <input>, <select>, атрибути id, type, name.	2
2	Стилізація web-сторінок за допомогою CSS. Селектори CSS. Налаштування параметрів тексту. Блочна верстка, flex-верстка. Адаптивний дизайн, правило @media.	2
3	Огляд мови JavaScript. Розгалуження, цикли, функції, робота зі строками.	2
4	Зміна DOM за допомогою JavaScript: пошук елементів, додавання та видалення елементів, динамічне додавання стилів.	4
5	Знайомство із інструментами DevTool браузера. Події, обробники подій.	2
6	Взаємодія зі сторонніми API за допомогою JavaScript.	2
7	Контрольна робота.	2
8	Створення простого Node.js серверу.	2
9	Основи баз даних. З'єднання з базою даних за допомогою Node.js Виконання базових CRUD операцій.	2
10	Створення власного RESTful API.	2
	<b>Разом</b>	<b>22</b>

#### 5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
1	Детальне знайомство із новими можливостями HTML 5. Верстання за допомогою технології css 3 – display: grid. Ознайомлення з фреймворками jquery, lodash, bootstrap, tailwind.	21

2	Розгляд архітектурних шаблонів, таких як MVC (Model-View-Controller) або MVVM (Model-View-ViewModel).	21
3	Детальне знайомство з Node.js, Express.js. Впровадження методів аутентифікації та авторизації в додатках.	24
<b>Разом</b>		<b>66</b>

## 6. Індивідуальні завдання

*Не передбачені*

## 7. Методи навчання

Пояснювально-ілюстративні лекції, репродуктивні і частково-пошукові методи при проведенні практичних занять, індивідуальні завдання. Студенти опановують значну частину теоретичного матеріалу шляхом самостійного написання комп'ютерних програм.

## 8. Методи контролю

- облік відвідування аудиторних або дистанційних занять;
- перевірка виконання домашніх завдань;
- усне опитування;
- перевірка контрольної роботи;
- підсумкова залікова робота.

## 9. Схема нарахування балів

Поточний контроль, самостійна робота, індивідуальні завдання					Сума		
Розділ 1		Розділ 2		Контрольна робота, передбачена навчальним планом	Індивідуальні завдання	Залікова робота	
T1	T2	T3	T4	20	-	40	100
10	10	10	10				

T1, T2 ... – теми розділів.

Для допуску до складання підсумкового контролю (заліку) здобувач вищої освіти повинен набрати не менше 10 балів з навчальної дисципліни під час поточного контролю, самостійної роботи, індивідуального завдання.

### Шкала оцінювання: дворівнева

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка
50-100	зараховано
1-49	не зараховано

### Критерії оцінювання

**Поточний контроль** враховує активність під час практичних занять та правильність виконання домашніх завдань, за кожне з яких разом можна отримати до 4 балів.

**Контрольна робота** оцінюється максимум у 20 балів. При оцінюванні враховується не лише правильність результату роботи програми, а і ефективність написаного програмного коду. Неefективний розв'язок може призвести до зняття 1-2 балів.

**Залікове завдання** складається з двох теоретичних питань та однієї задачі. **Перше теоретичне питання** оцінюється максимум у 15 балів та пов'язано із основними загальними концепціями створення клієнтської частини web-додатку. Максимальну кількість балів можна отримати, якщо студент демонструє розуміння основних концепцій та може навести приклади їх практичного застосування. Якщо студент не може навести прикладів, він отримує максимум 10 балів. Якщо студент має поверхове розуміння базових понять і не може навести приклади їх використання, він отримує максимум 5 балів. **Друге теоретичне питання** пов'язано із знанням стандартних можливостей мови JavaScript. В тому числі для програмування на стороні сервера. Максимальна оцінка 10 балів. Студент отримує максимальну оцінку, якщо він розуміє та вміє використовувати стандартні можливості. Якщо студент має часткове розуміння стандартних можливостей, але не до кінця вміє ними користуватися, він отримує від 1 до 9 балів. **Практичне завдання** пов'язано із розв'язанням базових задач web-розробки. Максимальна оцінка 15 балів. Якщо студент обрав нераціональний шлях розв'язання або не має повного розуміння, то він отримує 10-14 балів. При частковому розв'язанні студент отримує 1-9 балів.

## 10. Рекомендована література

### Основна література

1. D. DuRocher HTML and CSS QuickStart Guide, ClydeBank Media LLC. – 2021. – 352 p.
2. D. Flanagan, JavaScript: The Definitive Guide, O'Reilly Media, 2020. –1153 p.
3. M. Casciaro, L. Mammino, Node.js Design Patterns: Design and Implement Production-grade Node.js Applications Using Proven Patterns and Techniques. – Packt Publishing, 2020. – 664 p.

### Допоміжна література

4. Цеслів О.В. Основи програмування та веб-дизайн. – К. : КПІ ім. Ігоря Сікорського, 2020. – 149 с.
5. Мельник Р. Програмування веб-застосувань (фронт-енд та бек-енд). – Видавництво: «Львівська політехніка», 2018 – 248 с.

## 11. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. <http://www-library.univer.kharkov.ua/ukr/>
2. <https://uk.wikipedia.org/wiki/Веброзробка>
3. <https://developer.mozilla.org/en-US/docs/Web>
4. <https://www.w3schools.com/html/>
5. <https://www.w3schools.com/css/>
6. <https://www.w3schools.com/js/>
7. [https://www.tutorialspoint.com/internet\\_technologies/websites\\_development.htm](https://www.tutorialspoint.com/internet_technologies/websites_development.htm)