

# ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ



**ЗАТВЕРДЖУЮ**

Декан факультету інформаційних технологій  
проф. Тетяна ГОВОРУЩЕНКО

2024 р.

## РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

### WEB-технології та WEB-дизайн


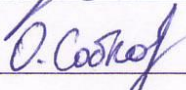
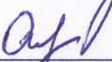
Назва дисципліни

**Галузь знань** – 12 Інформаційні технології  
**Спеціальність** – 122 Комп'ютерні науки  
**Рівень вищої освіти** – Перший бакалаврський  
**Освітньо-професійна програма** – Комп'ютерні науки  
**Обсяг дисципліни** – 11 кредитів ЄКТС  
**Шифр дисципліни** – ОПП.02  
**Мова навчання** – українська  
**Статус дисципліни** – обов'язкова (цикл професійної підготовки)  
**Факультет** - Інформаційних технологій  
**Кафедра** – Комп'ютерних наук

Курс	Семестр	Обсяг дисципліни		Кількість годин							Форма семестрового контролю		
		Кредити ЄКТС	Години	Аудиторні заняття				Індивідуальна робота студента	Самостійна робота, у т.ч. ІРС	Курсовий проект	Курсова робота	Залік	Іспит
				Разом	Лекції	Лабораторні роботи	Практичні заняття						
3	5	5	150	68	34	34			82				+
3	6	6	180	51	17	34			129	+			+
<b>Разом</b>		<b>11</b>	<b>310</b>	<b>119</b>	<b>51</b>	<b>68</b>			<b>211</b>	<b>1</b>			<b>2</b>

Робоча програма складена на основі освітньо-професійної програми «Комп'ютерні науки» та стандарту вищої освіти зі спеціальності 122 – Комп'ютерні науки першого (бакалаврського) рівня вищої освіти.

Програма складена

к.т.н., доц. Руслан БАГРІЙ

асистент каф. КН Олена СОБКО

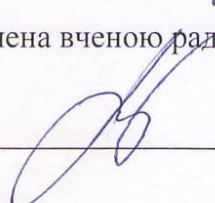
асистент каф. КН Олександр ОВЧАРУК

Схвалено на засіданні кафедри комп'ютерних наук

Протокол від 30 серпня 2024 р. № 1. Зав. кафедри  проф. Олександр БАРМАК

Робоча програма розглянута та схвалена вченою радою факультету інформаційних технологій

Голова вченої ради факультету



проф. Тетяна ГОВОРУЩЕНКО

## WEB-ТЕХНОЛОГІЇ ТА WEB-ДИЗАЙН

Тип дисципліни	Обов'язкова професійної підготовки
Рівень вищої освіти	Перший (бакалаврський)
Мова викладання	Українська
Семестр	п'ятий, шостий
Кількість встановлених кредитів ЄКТС	11
Форма здобуття освіти	Очна денна

**Результати навчання.** Студент, який успішно завершив вивчення дисципліни, повинен: вміло використовувати понятійний апарат та інструментальні засоби розробки веб-орієнтованих систем; вміти створювати веб-застосунки з використання веб-фреймворків та сучасних технологій розробки. Використовуючи раніше набуті знання по системах управління базами даних, об'єктно-орієнтованому проектуванню та програмуванню студенти повинні вміти застосувати ці знання та використовуючи сучасні платформи навчитися складати програмне забезпечення веб-застосунків. Володіти специфікою архітектурних шаблонів проектування та особливостями розробки серверної частини веб-орієнтованих систем; проводити повний цикл розробки програмного забезпечення для веб: від проектування архітектури до розгортання, тестування та оптимізації.

**Зміст навчальної дисципліни.** Вступ до Web. Протокол HTTP та HTTPS. Теоретичні основи веб-дизайну. Веб-технології створення сайту. Основи мови HTML. Теги та їх використання. Семантика та доступність. Технологія CSS. Використання стилів для форматування HTML. Верстка. Адаптивний дизайн. CSS-методології та препроцесори. Синтаксис мови PHP. Передача даних та обробка запитів на сервері. Основи роботи з базами даних. Взаємодія PHP і MySQL. Концепція MVC. PHP-фреймворк Laravel. Контролери та Шаблони. ORM Eloquent та міграції. Маршрутизація та контролер ресурсів. Робота з файлами. Валідація даних. Відношення між моделями. Отримання записів за зв'язком. Авторизація та аутентифікація. Сесії.

**Пререквізити** – об'єктно-орієнтоване програмування, системне та мережеве програмне забезпечення, проектування баз даних, основи програмної інженерії та тестування програмного забезпечення.

**Кореквізити** – проектування баз даних (курсове проектування), інформаційні технології хмарних обчислень.

**Запланована навчальна діяльність:** лекції – 51 год., лабораторні заняття – 68 год., самостійна робота – 211 год., разом – 330 год.

**Методи навчання:** лекції (з використанням методів візуалізації); лабораторні заняття (з використанням тренінгів, майстер-класів), самостійна робота (індивідуальні завдання).

**Форми оцінювання результатів навчання:** захист лабораторних робіт; підсумковий контрольний захід, захист курсового проекту.

**Форма семестрового контролю:** іспит – 5 семестр, курсовий проект – 6 семестр

**Навчальні ресурси:**

1. Основи проектування та розробка веб-додатків: навчальний посібник / Манзюк Е.А, Лищук О.А., Мазурець О.В, Багрій Р.О., Петровський С.С. – Хмельницький: ХНУ, 2019. – 184 с.
2. Frain, Ben. Responsive Web Design with HTML5 and CSS: Develop Future-Proof Responsive Websites Using the Latest HTML5 and CSS Techniques, 3rd Edition. Packt Publishing, 2020.
3. DuRocher, David. HTML and CSS QuickStart Guide, ClydeBank Media LLC, 2021.
4. Duckett, Jon. PHP & MySQL: Server-Side Web Development. 1st edition, Wiley, 2022.
5. Stauffer, Matt. Laravel: Up & Running: A Framework for Building Modern PHP Apps. 2nd edition, O'Reilly Media, 2019.
6. Correa, Daniel, и Paola Vallejo. Practical Laravel: Develop Clean MVC Web Applications. Independently published, 2022
7. Вебтехнології: методичні рекомендації до курсового проектування для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки» / Р. О. Багрій, О. В. Собко, О. О. Тищенко, С. С. Петровський. Хмельницький : ХНУ, 2024. 35 с.
8. Модульне середовище для навчання MOODLE. Режим доступу: <https://msn.khnu.km.ua>

**Викладач:** кандидат технічних наук, доцент Руслан БАГРІЙ, асистент кафедри комп'ютерних наук Олена СОБКО, асистент кафедри комп'ютерних наук Олександр ОВЧАРУК

## Пояснювальна записка

Дисципліна «Курс «WEB-технології та WEB-дизайн» дозволяє сформувати вміння і навички використання сучасних WEB-технологій необхідних для створення HTML-сторінок і WEB-сайтів.

Курс призначений для підготовки бакалаврів з галузі знань 12 "Інформаційні технології" за спеціальністю 122 "Комп'ютерні науки " і ставить за мету оволодіння навиками роботи з веб-технологіями та практичною реалізацією таких технологій у вигляді веб-застосунків, застосування фреймворків, що реалізують WEB-технології та WEB-дизайн та програмного інструментарію для реалізації таких технологій.

**Пререквізити** – об'єктно-орієнтоване програмування, системне та мережеве програмне забезпечення, проектування баз даних, основи програмної інженерії та тестування програмного забезпечення.

**Кореквізити** – проектування баз даних (курсове проектування), інформаційні технології хмарних обчислень.

Відповідно до Стандарту вищої освіти із спеціальності 122 Комп'ютерні науки та освітньої програми «Комп'ютерні науки», дисципліна має забезпечити:

### **компетентності:**

**ФК 8.** Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

### **програмні результати навчання:**

**ПРН 10.** Використовувати інструментальні засоби розробки клієнт-серверних застосунків, проектувати концептуальні, логічні та фізичні моделі баз даних, розробляти та оптимізувати запити до них, створювати розподілені бази даних, сховища та вітрини даних, бази знань, у тому числі на хмарних сервісах, із застосуванням мов веб-програмування.

**ПРН 13.** Застосовувати мови системного програмування та методи розробки програм, що взаємодіють з компонентами комп'ютерних систем, знати мережні технології, архітектури комп'ютерних мереж, мати практичні навички технології адміністрування комп'ютерних мереж та їх програмного забезпечення

**Мета дисципліни** – опанування базових знань з сучасних веб-технологій та основ веб-програмування.

**Предмет дисципліни.** Методи та підходи до розробки динамічних веб-сайтів; механізми та принципи роботи веб-застосунків.

**Завдання дисципліни.** Формування професійних якостей спеціалістів, що знають технічні засоби та програмне забезпечення для реалізації веб-технологій, набуття компетенції, знань, умінь та навиків із застосування веб-технологій та розробки веб-орієнтованих систем, відповідно до кваліфікації фахівця з інформаційних технологій.

**Результати навчання:** студент, який успішно завершив вивчення дисципліни, повинен: вміло використовувати понятійний апарат та інструментальні засоби розробки веб-орієнтованих систем; вміти створювати веб-застосунки з використання веб-фреймворків та сучасних технологій розробки. Використовуючи раніше набуті знання по системах управління базами даних, об'єктно-орієнтованому проектуванню та програмуванню студенти повинні вміти застосувати ці знання та використовуючи сучасні платформи навчитися складати програмне забезпечення веб-застосунків. Володіти специфікою архітектурних шаблонів проектування та особливостями розробки серверної частини веб-орієнтованих систем; проводити повний цикл розробки програмного забезпечення для веб: від проектування архітектури до розгортання, тестування та оптимізації.

### Структура залікових кредитів дисципліни

Назва розділу (теми)	Кількість годин, відведених на:		
	лекції	лабор. роботи	СРС
<b>П'ятий семестр</b>			
Тема 1. Вступ до Web. Веб-технології створення сайту.	6	4	6
Тема 2. Основи мови HTML.	4	4	12
Тема 3. Технологія CSS. CSS та верстка.	8	8	12
Тема 4. Основи мови PHP. Передача даних та обробка запитів на сервері.	8	8	12
Тема 5. Основи роботи з базами даних. Взаємодія PHP і MySQL.	4	4	12
Тема 6. Концепція MVC.	4	6	28
<b>Разом за семестр:</b>	<b>34</b>	<b>34</b>	<b>82</b>
<b>Шостий семестр</b>			
Тема 7. PHP-фреймворк Laravel.	1	4	16
Тема 8. Маршрутизація. Контролери та Шаблони.	2	4	16
Тема 9. Робота із базами даних в Laravel. ORM Eloquent та міграції.	4	8	32
Тема 10. Обробка запиту та відповіді сервера. Маршрутизація та контролер ресурсів.	2	4	16
Тема 11. Робота з файлами. Валідація даних.	2	4	16
Тема 12. Відношення між моделями. Отримання записів за зв'язком	4	6	24
Тема 13. Авторизація та аутентифікація. Сесії.	2	4	9
<b>Разом за семестр:</b>	<b>17</b>	<b>34</b>	<b>129</b>
<b>Разом:</b>	<b>51</b>	<b>68</b>	<b>211</b>

### Програма навчальної дисципліни Зміст лекційного курсу\*

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
	<b>П'ятий семестр</b>	
1	<b>Введення в Web-дизайн</b> Web-дизайн: визначення поняття. Введення в Web-дизайн і принципи дизайну. Проектування сайтів; план сайту; класифікація сайтів; структура сайту; класифікація моделей сайтів; порівняння сайтів; теорія навігації. Динамічне та статичне компонування сайту. [2] с. 10–18,	2
2	<b>Проектування Web-сайту.</b> Постановка завдання на розробку. З'ясування цілей та задач Web-публікації. Оцінка аудиторії (круга користувачів), її специфічних особливостей. Створення концепції. [2] с. 19–27	2
3	<b>Розробка дизайну сторінок</b> Відбір матеріалу, вибір структури сайту і особливостей розміщення матеріалів. Вимоги до інтерфейсу. Вимоги до оформлення Web-сторінок. Вибір дизайну публікації в цілому. Створення макету. [2] с. 89–123	2
4	<b>Основи мови HTML.</b> Основи мови розмітки гіпертексту – HTML. Структура HTML-документа. Оформлення тексту. Гіперпосилання. Вставка зображень.	2

	[1] с. 15 – 20, [3] с. 27 – 30, [4] с. 90 – 104,	
5	<b>Оформлення HTML документа.</b> Блочні елементи, Рядкові елементи. Створення таблиць в HTML. Верстка сайту. Форми в HTML. Семантика та доступність. [1] с. 21 – 29, [3] с. 31 – 43, [4] с. 105 – 140, с. 250 – 285	2
6	<b>Технологія CSS.</b> Технологія CSS. Під'єднання до Html. Наслідування та каскадність стилів. Синтаксис опису стилів. Види селекторів. Одиниці виміру в CSS. Видимість елементів CSS. CSS-властивості. [3] с. 155 – 183, [4] с. 140 – 163, с. 181 – 226	2
7	<b>CSS-методології та препроцесори.</b> Основні принципи, популярні методології (BEM, OOCSS, SMACSS, Atomic CSS). Визначення препроцесорів (Sass, Less та Stylus). [13] с. 21 – 35, с. 95 – 123	2
8	<b>Блочна модель.</b> Положення елементів. Відступи. Границі. Плаваючі елементи. Розміри елементів. Адаптивна розмітка веб-сторінки. [4] с. 163 – 180	2
9	<b>Блочна структура сторінки.</b> Позиціонування. Модульна сітка - GRID. Контейнер Flexbox. Використання бібліотеки Bootstrap. [3] с. 77 – 155, с. 327 – 382	2
10	<b>Основи мови PHP.</b> Історія створення PHP. Основи синтаксису. Змінні. Типи даних. Вирази порівняння. Робота з масивами. Оператори в PHP. [1] с. 39 – 60, [5] с. 29 – 103	2
11	<b>Функції в PHP. Об'єктно-орієнтований підхід.</b> Функції в PHP. Об'єкти і класи. [5] с. 143 – 177	2
12	<b>Протокол HTTP.</b> Веб інтерфейс. Підпорядкованість протоколів. Робота TCP та IP при передачі даних. Методи HTTP. Структура URL. Заголовки запиту HTTP. HTTP- відповідь сервера. [1] с. 171 – 192, [8] с. 41 – 65	2
13	<b>Обробка форм. Взаємодія із сервером.</b> Використання HTML-форм для передачі даних на сервер. Метод GET і POST. Обробка запитів за допомогою PHP. Робота з файловою системою. Завантаження файлу на сервер. Робота з сесіями та cookies. Обробка помилок. Робота з датами. [1] с. 68 – 83, [5] с. 231 – 285, 285 – 309	2
14	<b>Серверна база даних MySQL.</b> Формування запитів до бази даних. Оператор вибору Select. Застосування агрегатних функцій. [1] с. 94 – 131, [5] с. 381 – 433	2
15	<b>Взаємодія PHP та MySQL.</b> Встановлення з'єднання. Вибір бази даних. Відображення даних, що зберігаються в MySQL. [1] с. 83 – 92, [5] с. 433 – 521	2
16	<b>Концепція MVC.</b> Типи архітектури веб-додатків. MVC для веб-додатків. Шаблони на базі підходу модель-вид-контролер. Маршрутизація і URL-адреси. [1] с. 131 – 154	2
17	<b>Реалізація MVC.</b> Використання шаблонного підходу для роботи	2

	з базами даних. Використання MVC та CRUD у додатку. [1] с. 154 – 171	
<b>Разом за семестр:</b>		<b>34</b>
<b>Шостий семестр</b>		
18	<b>PHP-фреймворк Laravel.</b> PHP Frameworks. Шаблони проектування. Composer. Структура Laravel. Робоче середовище. Artisan: основні команди [6] с. 1 – 22, с. 201 – 205	1
19	<b>Маршрутизація та Контролери. Шаблонізатор Blade.</b> Реєстрація маршрутів. Параметри маршрутів. Групування маршрутів. Контролери. Види. Перенаправлення маршрутів. Передача даних у види. Керуючі конструкції. Наслідування шаблонів. Компоненти. [6] с. 23 – 86	2
20	<b>Робота із базами даних в Laravel.</b> Міграції. Конструктор запитів до БД. Колекції. Seeding. [6] с. 87 – 116	2
21	<b>Модель Eloquent</b> Робота з ORM Eloquent. Active Record. [6] с. 117 – 154	2
22	<b>Обробка запиту та відповіді сервера. Маршрутизація та контролер ресурсів.</b> Обробка запиту: отримання URL, отримання даних. Перевірка даних. Робота за файлами. Робота з маршрутами. Іменовані маршрути. Ресурс Контролер. Спуфінг. Відповідь сервера. Статус відповіді. Заголовки відповіді. [6] с. 253 – 278	2
23	<b>Робота з файлами. Валідація даних.</b> Storage. Робота з файлами. Основні методи. Генерація URL. Валідація в Laravel. Правила валідації. Повідомлення про помилки. [6] с. 181 – 200, с. 377 – 381,	2
24	<b>Відношення між моделями. Отримання записів за зв'язком.</b> Види відношень Eloquent. Один до одного. Один до багатьох. Багато до багатьох. [9] с. 79 – 90	2
25	<b>Відношення між моделями. Отримання записів за зв'язком.</b> Види відношень Eloquent. До багатьох через. Поліморфні зв'язки. Жадібне завантаження даних. [9] с. 90 – 113	2
26	<b>Авторизація та аутентифікація. Сесії.</b> Стартові пакети аутентифікації. Поля авторизації. Фасад Auth. Сесії в Laravel: налаштування, основні команди. [6] с. 221 – 253, с. 381 – 386,	2
<b>Разом за семестр:</b>		<b>17</b>

**Примітка.** \*Лекції плануються по 2 год. Якщо у навчальному плані в непарних семестрах запланована 1 год. аудиторних занять на тиждень, то залежно від розкладу занять фактична кількість годин становитиме 36 – по чисельнику, 34 – по знаменнику.

### Зміст лабораторних занять Перелік лабораторних занять для студентів денної форми навчання

№ з/п	Тема лабораторного заняття	Кількість годин
<b>П'ятий семестр</b>		
1	Розробка вебсторінок із використанням мови HTML. [2] с. 10 – 123;	4

2	Використання CSS для стилізації вебсторінок. [1] с. 15 – 20, [3] с. 27 – 30, [4] с. 90 – 104,	4
3	Адаптивна верстка. CSS-Flex. CSS-Grid. [2] с. 121 – 155, [3] с. 155 – 183, [4] с. 140 – 163	8
4	Основи мови PHP. Функції та масиви. [1] с. 39 – 60, [5] с. 29 – 103	4
5	Обробка форм. Робота з файлами. [1] с. 60 – 67, с. 171 – 192, [8] с. 41 – 65	4
6	Взаємодія PHP та MySQL. [1] с. 83 – 92, с. 94 – 131, [5] с. 381 – 521	4
7	Реалізація MVC. Використання MVC для роботи з базою даних. [1] с. 131 – 171	6
<b>Разом за семестр:</b>		<b>34</b>
<i><b>Шостий семестр</b></i>		
8	Встановлення та налаштування фреймворка Laravel. Маршрутизація та шаблони. [7] с. 12 – 24	4
9	Використання контролерів та передача даних у шаблони. Створення контролера. Передача даних у макет представлення View. [7] с. 25 – 34	4
10	Робота із базами даних. ORM Eloquent та міграції. [7] с. 35 – 48	8
11	Html-форми та редагування даних. Маршрутизація та контролер ресурсів. [7] с. 49 – 68	4
12	Валідація даних. Повідомлення про помилки. [7] с. 69 – 70	4
13	Зв'язки між відношеннями. [7] с. 85 – 108	6
14	Аутентифікація та авторизація користувачів. Захист маршрутів. [7] с. 71 – 84	2
15	Підсумкове заняття, захист курсового проекту.	2
<b>Разом за семестр:</b>		<b>34</b>

### **Зарахування результатів неформальної освіти**

Окремі результати вивчення курсу можуть бути зараховані у випадку отримання студентом результатів навчання у неформальній освіті, що підтверджені відповідним документом (сертифікат, свідоцтво, освітня програма тощо).

Як результат виконання лабораторних робіт №1-3 зараховуються онлайн-курси:

- HTML5/ CSS3/JAVASCRIPT FUNDAMENTALS

<https://career.softserveinc.com/uk-ua/technology/course/html5-css3-javascript-fundamentals>

- Front-End Self-Paced

<https://training.epam.ua/ua/training/3474>

- Основи Web UI розробки 2023

[https://prometheus.org.ua/course/course-v1:LITS+114+2022\\_T2](https://prometheus.org.ua/course/course-v1:LITS+114+2022_T2)

- The World Wide Web Consortium (W3C): CSS Basics

<https://www.edx.org/learn/css/the-world-wide-web-consortium-w3c-css-basics>

Як результат виконання лабораторних робіт №4-6 зараховуються онлайн-курси:

- Building Web Applications in PHP

<https://www.coursera.org/learn/web-applications-php>

- Building Database Applications in PHP

<https://www.coursera.org/learn/database-applications-php>

Як результат виконання лабораторних робіт №7-14 зараховуються онлайн-курси:

- Getting Started with Laravel (PHP Framework) 7 - The Basics  
<https://www.pluralsight.com/courses/laravel-php-framework-getting-started-the-basics>
- Laravel : Make six projects with PHP and Laravel  
<https://www.udemy.com/course/laravel-masterclass-build-web-apps-with-laravel-7-php/>

### **Зміст самостійної (у т.ч. індивідуальної) роботи**

Самостійна робота студентів денної форми навчання полягає у систематичному опрацюванні програмного матеріалу з відповідних джерел інформації, підготовці до виконання і захисту лабораторних робіт, виконанні курсового проекту, тощо.

Самостійна робота студентів денної форми навчання полягає у систематичному опрацюванні програмного матеріалу, підготовці до виконання і захисту лабораторних робіт, виконанні курсового проекту, тощо. Керівництво самостійною роботою та контроль за виконанням індивідуального завдання здійснює викладач згідно з розкладом консультацій у позаурочний час.

Самостійна робота дисципліни «WEB-технології та WEB-дизайн» також передбачає виконання Індивідуальної самостійної роботи (ІСР) до 4 індивідуальних завдань, що має на меті закріплення та розвиток вмінь і навичок, набутих при виконанні циклу лабораторних робіт. Оцінка, яка виставляється за виконання ІСР, виводиться як середня з оцінювання таких елементів (опис критеріїв наведено вище):

- якість та рівень розв'язання задач роботи;
- якість наповнення й оформлення звіту.

### **Зміст самостійної роботи студентів денної форми навчання**

Номер тижня	Вид самостійної роботи <i>П'ятий семестр</i>	Кількість годин
1-2	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи. Огляд засобів створення дизайну веб-сторінок. Основи мови HTML. Підготовка до виконання лабораторної роботи 1.	6
3-4	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи. Використання CSS для стилізації вебсторінок. Підготовка до виконання лабораторної роботи 2. Виконання завдання для самостійної роботи №1 – використання можливостей CSS3 для стилізації вебсторінок	12
5-6	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи. Адаптивна верстка. Підготовка до виконання лабораторної роботи 3. Виконання завдання для самостійної роботи №2 – використання CSS-фреймворку Bootstrap для розробки вебінтерфейсів	12
7-8	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи. Основи мови PHP. Функції і масиви. Підготовка до виконання лабораторної роботи 4.	6
9-10	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи. Обробка форм. Взаємодія із сервером. Підготовка до виконання лабораторної роботи 5.	6
11-12	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи. Серверна база даних MySQL. Формування запитів до бази даних. Підготовка до виконання лабораторної роботи 6. Виконання завдання для самостійної роботи №3 – реалізація авторизованого доступу за допомогою механізму сесій в PHP.	12
13-14	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи. Концепція MVC. Підготовка до виконання лабораторної роботи 7. Виконання завдання для самостійної роботи №4 – використання MVC для розробки вебсайту.	12



15-18	Опрацювання лекційного матеріалу. Підготовка до захисту лабораторної роботи. Виконання завдання для самостійної роботи №4 – використання MVC для розробки вебсайту. Підготовка до підсумкового контролю. Презентація результатів виконання індивідуальних завдань.	16
	<b>Разом:</b>	<b>82</b>
<b>Шостий семестр</b>		
9	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи 1. Вибір теми курсового проєкту (далі – КП).	16
10	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи 2. Складання плану КП.	16
11	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи 3. Виконання КП.	32
12	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи 4. Виконання КП.	16
13	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи 5. Виконання КП.	16
14	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи 6. Виконання КП.	24
15	Опрацювання лекційного матеріалу. Опрацювання матеріалу лабораторної роботи 7. Оформлення курсового проєкту та підготовка до захисту.	9
	<b>Разом:</b>	<b>129</b>

### **Технології навчання**

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій, зокрема: лекції (з використанням методів візуалізації); лабораторні заняття (з використанням майстер-класів), самостійна робота (індивідуальні завдання; курсове проєктування).

Необхідні інструменти, обладнання, програмне забезпечення: комп'ютер (надається для використання в лабораторіях кафедри КН), Google Chrome (безкоштовне), VS Code (безкоштовне), локальний вебсервер: XAMPP (безкоштовне) або інші безкоштовні рішення.

### **Методи контролю**

Поточний контроль здійснюється під час лекційних та лабораторних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком навчального процесу. При цьому використовуються такі методи поточного контролю:

- усне опитування перед допуском до лабораторного заняття та захист лабораторних робіт;
- презентація і захист індивідуальних завдань.

При виведенні підсумкової семестрової оцінки враховуються результати як поточного контролю, так і підсумкового контрольного заходу, який проводиться у вигляді іспиту, що складається шляхом надання письмових відповідей на питання екзаменаційних білетів. Студент, який набрав позитивний середньозважений бал за поточну роботу і не здав підсумковий контрольний захід (іспит), вважається невстигаючим.

### **Оцінювання результатів навчання студентів у семестрі**

Оцінювання академічних досягнень здобувача вищої освіти здійснюється відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ». Кожний вид роботи з дисципліни оцінюється за інституційною чотирибальною шкалою і виставляється в електронному журналі обліку успішності.

Засвоєння студентом теоретичного матеріалу з дисципліни оцінюється під час захисту лабораторних робіт та підсумковому контрольному заході. Поточний контроль передбачає захист лабораторних робіт. Поточний контроль здійснюється під час лабораторних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком

навчального процесу.

Політика проведення лабораторних робіт з навчальної дисципліни «WEB-технології та WEB-дизайн»:

1. Студент має бути присутнім на кожному занятті.
2. Студент може бути відсутнім на занятті, якщо здав відповідну лабораторну роботу або з поважної причини: хвороба, тощо.
3. В разі потреби при виконанні лабораторної роботи студент може скористатися засобами зв'язку для пошуку інформації в інтернеті, тощо.
4. На одному занятті захищається тільки одна лабораторна робота. В разі бажання студента здати ще одну роботу, студент має стати в кінець черги. За одне заняття можна здати не більше 2 робіт.
5. Перескладання лабораторної роботи можливе на наступному занятті.
6. Можливе перезарахування деяких лабораторних робіт елементами неформальної освіти згідно Робочої програми дисципліни.

При виконанні лабораторних робіт студент має дотримуватись положень академічної доброчесності; використання напрацювань інших студентів (компоненти програмного забезпечення та програмні коди, графічні матеріали лабораторних робіт та фрагменти звітів з лабораторних робіт) є підставою для зниження оцінки чи перескладання лабораторної роботи.

Студент до виконання лабораторної роботи має бути підготовлений, зокрема: ознайомлений із темою, порядком виконання та відповідними теоретичними відомостями. Для встановлення рівня готовності студента до виконання роботи викладач здійснює опитування.

Пропущене лабораторне заняття студент зобов'язаний відпрацювати не пізніше, ніж за два тижні до кінця теоретичних занять у семестрі.

Семестрова підсумкова оцінка визначається як середньозважена з усіх видів навчальної роботи, виконаних і зданих студентом позитивно, з урахуванням коефіцієнта вагомості і розраховується в автоматизованому режимі за відповідною програмою. Вагові коефіцієнти змінюються залежно від структури дисципліни і важливості окремих видів її робіт. При виведенні підсумкової семестрової оцінки враховуються результати як поточного контролю, так і підсумкового контрольного заходу, який проводиться письмово з усього матеріалу дисципліни. Студент, який набрав позитивний середньозважений бал за поточну роботу і не здав підсумковий контрольний захід, вважається невстигаючим.

Оцінювання знань студентів здійснюється за такими критеріями:

Оцінка за інституційною шкалою	Узагальнений критерій оцінювання
Відмінно	Студент глибоко і у повному обсязі опанував зміст навчального матеріалу, легко в ньому орієнтується і вміло використовує понятійний апарат; вміє пов'язувати теорію з практикою, вирішувати практичні завдання, впевнено висловлювати і обґрунтовувати свої судження. Відмінна оцінка передбачає, логічний виклад відповіді державною мовою (в усній або у письмовій формі), демонструє якісне оформлення роботи і володіння спеціальними інструментами. Студент не вагається при видозміні запитання, вміє робити детальні та узагальнюючі висновки. При відповіді допустив дві–три несуттєві <i>похибки</i> .
Добре	Студент виявив повне засвоєння навчального матеріалу, володіє понятійним апаратом і фаховою термінологією, орієнтується у вивченому матеріалі; свідомо використовує теоретичні знання для вирішення практичних завдань; виклад відповіді грамотний, але у змісті і формі відповіді можуть мати місце окремі неточності, нечіткі формулювання закономірностей тощо. Відповідь студента будується на основі самостійного мислення. Студент у відповіді допустив дві–три <i>несуттєві помилки</i> .
Задовільно	Студент виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, справляється з виконанням практичних завдань, передбачених програмою.

	Як правило, відповідь студента будується на рівні репродуктивного мислення, студент має слабкі знання структури курсу, допускає неточності і <i>суттєві помилки</i> у відповіді, вагається при відповіді на видозмінене запитання. Разом з тим, набув навичок, необхідних для виконання нескладних практичних завдань, які відповідають мінімальним критеріям оцінювання і володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у відповіді.
Незадовільно	Студент виявив розрізнені, безсистемні знання, не вміє виділяти головне і другорядне, допускається помилок у визначенні понять, перекручує їх зміст, хаотично і невпевнено викладає матеріал, не може використовувати теоретичні знання при вирішенні практичних завдань. Як правило, оцінка "незадовільно" виставляється студенту, який не може продовжити навчання без додаткової роботи з вивчення дисципліни.

### Структурування дисципліни за видами робіт і оцінювання результатів навчання студентів денної форми навчання у семестрі за ваговими коефіцієнтами

Аудиторна робота								Самостійна, індивідуальна робота	Семестровий контроль, іспит
<i>п'ятий семестр</i>									
Лабораторні роботи №:								Індивідуальне завдання	Підсумковий контрольний захід
1	2	3	4	5	6	7			
ВК: 0,5								0,1	0,4
<i>шостий семестр</i>									
Лабораторні роботи №:								Курсове проектування	Підсумковий контрольний захід
1	2	3	4	5	6	7			
ВК: 0,6								+	0,4

\*Умовні позначення: ВК – ваговий коефіцієнт.

### Курсове проектування

Індивідуальна робота студентів у шостому семестрі відповідно до навчального плану передбачає виконання курсового проєкту.

Метою курсового проєкту є формування практичних навичок у розробці та реалізації вебзастосунків. Основна увага зосереджується на використанні вебтехнологій для створення інтуїтивно зрозумілих інтерфейсів, оптимізації взаємодії з користувачами, а також роботи з базами даних. Завдяки виконанню проєкту, здобувачі зможуть застосувати теоретичні знання на практиці, вирішуючи конкретні задачі та проблеми, що виникають у процесі створення сучасних вебсайтів і вебзастосунків, що стане важливим кроком у підготовці до професійної діяльності в галузі інформаційних технологій і комп'ютерних наук.

На початку семестру керівник курсового проєкту проводить загальну консультацію, на якій висвітлюються мета та завдання курсового проектування, етапи і терміни його виконання, календарний план, тематика, форми та критерії оцінювання. Розглядаються загальні вимоги щодо написання й оформлення курсового проєкту, вимоги до розроблення вебзастосунку, розклад індивідуальних консультацій, список рекомендованої навчальної й довідкової літератури для виконання завдань курсового проєкту. Оголошується перелік тем курсових проєктів, що пропонуються кафедрою студентам для вибору. За узгодженням з лектором (керівником проєкту) тема також може бути запропонована здобувачем самостійно, заміненна або відкоригована.

Курсовий проєкт студенти виконують згідно із методичними рекомендаціями до курсового проектування.

Пояснювальна записка є текстовим документом, що містить аналіз, обґрунтування й опис прийнятих у курсовому проєкті рішень щодо розроблення вебзастосунку, а також документування процесу програмного проектування та реалізації вебзастосунку. Рекомендована кількість сторінок пояснювальної записки курсового проєкту (без завдання,

анотації та додатків) 35–45 с.

Консультації з курсового проекту здійснює керівник проекту – викладач кафедри відповідно до графіку, затвердженого деканом факультету. Захист курсового проекту здійснюється після його перевірки керівником перед комісією, яка складається з 2–3 викладачів кафедри. До складу комісії обов’язково входить керівник проекту.

### Орієнтовна тематика курсових проектів

1. Вебсайт для організації волонтерської діяльності
2. Інтернет-сервіс для обговорення комп’ютерних ігор
3. Вебсервіс для контролю особистих фінансів
4. Вебсайт кулінарних рецептів
5. Вебсервіс для організації туристичних подорожей
6. Вебсервіс для системи керування 3D-моделями
7. Вебплатформа для тематичних вікторин
8. Вебсайт для онлайн замовлень піцерії
9. Вебсайт притулку для бездомних тварин
10. Вебсайт пошуку роботи

Оцінювання курсового проекту здійснюється за інституційною чотирибальною шкалою «відмінно», «добре», «задовільно», «незадовільно» і шкалою ЄКТС.

Критерії оцінювання курсового проекту:

– відмінно – розроблений вебзастосунок виконує всі необхідні функції, що були поставлені у завданні, використано сучасні засоби розроблення, пояснювальна записка повністю відповідає вимогам оформлення та розкриває всі положення курсового проекту, на всі запитання на захисті дано вичерпні й коректні відповіді;

– добре – розроблений вебзастосунок виконує основні необхідні функції, що були поставлені у завданні, використано сучасні засоби розроблення, пояснювальна записка у достатній мірі відповідає вимогам оформлення і розкриває ключові положення проекту, на всі запитання на захисті дано коректні відповіді;

– задовільно – розроблений вебзастосунок виконує більшість необхідних функцій, що були поставлені у завданні, пояснювальна записка переважно відповідає вимогам оформлення й розкриває більшість положень курсового проекту, на більшість запитань на захисті дано коректні відповіді;

– незадовільно – розроблений вебзастосунок виконує недостатню кількість функцій, що були поставлені у завданні, або не в повній мірі відповідає темі курсового проекту, пояснювальна записка не відповідає вимогам оформлення або недостатньо розкриває положення курсового проекту, на більшість запитань на захисті дано некоректні відповіді;

### Структурування оцінювання курсового проекту за компонентами та ваговими коефіцієнтами

№	Компоненти оцінювання	Коеф.
1	Пояснювальна записка – відповідність структури та вмісту, рівень розкриття мети проектування, оформлення (нормоконтроль)	0,3
2	Вебзастосунок – функціонал, коректність результатів, складність реалізації, використані вебтехнології.	0,3
3	Захист курсового проекту (доповідь, відповіді на питання)	0,4

### Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Інституційна інтервальна шкала балів	Інституційна оцінка, критерії оцінювання	
A	4,75–5,00	5	Зарах <i>Відмінно</i> – глибоке і повне опанування навчального матеріалу і виявлення відповідних умінь та навичок <i>Добре</i> – повне знання навчального матеріалу з кількома
B	4,25–4,74	4	

				незначними помилками
C	3,75–4,24	4		<i>Добре</i> – в загальному правильна відповідь з двома-трьома суттєвими помилками
D	3,25–3,74	3		<i>Задовільно</i> – неповне опанування програмного матеріалу, але достатнє для практичної діяльності за професією
E	3,00–3,24	3		<i>Задовільно</i> – неповне опанування програмного матеріалу, що задовольняє мінімальні критерії оцінювання
FX	2,00–2,99	2	Незараховано	<i>Незадовільно</i> – безсистемність одержаних знань і неможливість продовжити навчання без додаткових знань з дисципліни
F	0,00–1,99	2		<i>Незадовільно</i> – необхідна серйозна подальша робота і повторне вивчення дисципліни

### Перелік питань для самоконтролю

1. Що таке Web та як він працює?
2. Як відбувається обмін даними між браузером та сервером?
3. Що таке протоколи HTTP та HTTPS? Як вони використовуються для передачі даних між клієнтом та сервером?
4. Як працює клієнт-серверна архітектура?
5. Що таке елемент HTML та як він записується?
6. В чому полягає особливість порожніх елементів?
7. Що таке атрибут HTML та як він записується?
8. Як визначаються логічні атрибути?
9. Як записати власний атрибут?
10. Як створюються гіперпосилання?
11. Як задати ціль гіперпосилання?
12. Яким чином розмінюються зображення на веб-сторінці?
13. Що таке фрейм та як його додати в HTML документ?
14. Які є основні елементи розмітки в HTML?
15. Що таке теги та їх атрибути? Як вони використовуються для створення різноманітного контенту?
16. Що таке блочні та інлайнні теги? Як вони використовуються?
17. Які теги використовуються для створення списків, таблиць та форм?
18. Які атрибути використовуються для налаштування елементів та їх правильного відображення на сторінці?
19. Як семантика впливає на доступність та SEO-оптимізацію вебсторінок?
20. Які теги використовуються для створення заголовків, абзаців, списків тощо?
21. У чому різниця між рядковим, блоковим та рядково-блоковим елементами (inline, block, inline-block)?
22. Які теги та атрибути обов'язково мають бути в HTML-документі?
23. Що таке метатеги?
24. Яка роль CSS в програмуванні вебсторінок?
25. Який синтаксис має CSS?
26. Як працюють селектори, властивості та значення?
27. Яку проблему вирішує специфічність?
28. Що таке псевдоелементи та псевдокласи?
29. Які формати кольорів використовуються в CSS?
30. Які основні принципи стилізації тексту? Що таке сімейство шрифтів, наслідування, line-height тощо?

31. Що таке блокова модель документа? Як використовувати властивості для зміни розмірів та положення елементів на сторінці?
32. Як формувати списки за допомогою CSS?
33. Як формувати форми за допомогою CSS? Які особливості має стилізація форм (стани елементів, валідація тощо)?
34. Навіщо потрібні індикатори завантаження (спінери)?
35. Що краще використовувати для вставки зображень на вебсторінку: CSS-властивість 'background-image' чи HTML-тег 'img'? Які плюси та мінуси кожного з них?
36. Що таке CSS Grid Layout та як побудувати Layout з його допомогою?
37. Що таке CSS Flexbox Layout та як побудувати Layout з його допомогою?
38. Коли варто використовувати Flexbox?
39. Які переваги має Grid перед Flexbox?
40. Які є принципи позиціонування елементів в CSS? Як вони взаємодіють між собою?
41. Який сенс поняття Viewport в контексті вебдизайну? Яку роль відіграють медіазапити в адаптивному дизайні?
42. Як правильно стилізувати посилання в CSS, використовуючи принцип «LoVe HAte»? Як працює цей принцип?
43. Що таке каскадна таблиця стилів CSS?
44. Як додати внутрішній стиль CSS?
45. Як додати вбудовану таблицю стилів CSS?
46. Як додати посилання на зовнішню таблицю стилів CSS?
47. Як використовувати селектори класів для стилізації елементів?
48. Як використовувати селектори ідентифікаторів для стилізації елементів?
49. Які ще селектори можна використовувати?
50. Як виконується каскадування стилів CSS?
51. Як враховується специфічність стилів?
52. Як виконується спадкування стилів CSS?
53. Яким чином задається колір в CSS?
54. Як задати параметри шрифту в CSS?
55. Які є підходи до побудови Layout з використанням CSS?
56. Які принципи дизайну та його структури необхідно розуміти для роботи з макетами та дизайнами?
57. Які основні принципи стоять за адаптивною та резиновою версткою?
58. Які є сучасні підходи до підключення стилів до вебсторінки та чому їх використовують?
59. У яких випадках використовуються вбудовані та внутрішні стилі CSS?
60. Як вертикально та горизонтально центрувати елементи на вебсторінці?
61. Як використовувати одиниці виміру в CSS: %, vw, vh, em, rem?
62. Як використовувати CSS-анімації та переходи для створення ефектів руху, зміни форми та положення елементів?
63. Які є особливості використання властивостей анімацій position та transform? Які їх плюси та мінуси?
64. Яка основна відмінність між CSS-властивостями transition та animation?
65. Як використовувати media queries, Flexbox та Grid для створення адаптивної верстки?
66. Які розміри можна використовувати для досягнення потрібної адаптивності та зручності відображення сторінки на різних пристроях та браузерах?
67. Які ключові аспекти потрібно враховувати для створення зручного для користувача інтерфейсу?
68. Який сенс мають CSS-методології для сучасного розробника?
69. Як препроцесори допомагають покращити процес розробки стилів?
70. Коли варто використовувати Sass-змінні, а коли CSS-змінні?
71. Які методи дозволяють прискорити завантаження вебсторінки?
72. Серверна мова PHP. Синтаксис. Включення PHP-сценарію в HTML-документ.
73. Серверна мова PHP. Організація розгалужень.
74. Серверна мова PHP. Поняття класу. Основні компоненти класу.
75. Серверна мова PHP. Абстрагування, інкапсуляція, модульність і ієрархія

76. Серверна мова PHP. Зберігання й використання даних користувача. Способи зберігання. Зберігання даних у файлах.
77. Серверна мова PHP. Зберігання даних у файлах. Відкриття файлу. Функція fopen(). Режими файлу. Читання файлу. Запис у файл.
78. Серверна мова PHP. Організація лічильника відвідувань.
79. Серверна мова PHP. Обробка форм.
80. Серверна мова PHP. Масиви, наслідування.
81. Аутентифікація користувачів засобами PHP.
82. Аутентифікація користувачів засобами WEB-сервера.
83. Назвіть відомі Вам способи створення БД у MySQL.
84. Назвіть відомі Вам способи створення таблиць у MySQL.
85. Які типи даних існують у MySQL?
86. Чим відрізняється тип CHAR від типу VARCHAR?
87. Які формати типу дати існують і від чого залежать?
88. Чим відрізняється SQL, який використовується у MySQL, від стандартного SQL?
89. Які функції MySQL використовуються в SQL-запитах для організації повнотекстового пошуку в таблиці?
90. Які режими повнотекстового пошуку існують в MySQL?
91. Як створити індекс для повнотекстового пошуку?
92. Які особливості використання логічного режиму повнотекстового пошуку?
93. Як підключитися за допомогою PHP-скрипта до екземпляра сервера MySQL?
94. Яким чином обрати базу даних у підключеному екземплярі сервера MySQL?
95. Яким чином виконати SQL-запит в обраній базі даних MySQL?
96. У якому вигляді вертається результат запиту до бази даних MySQL?
97. Яким чином можна обробляти результат запиту до БД MySQL?
98. Як перевірити чи успішно було виконано запит з PHP до БД MySQL?
99. Чому дуже важливо виконувати перевірку успішності запиту?
100. Який метод передачі змінних форми до оброблювача більш надійний, особливо при авторизації користувачів?
101. Існуючі об'єктно-орієнтовані технології web-програмування.
102. Який основний зміст паттерна MVC?
103. MVC архітектура Laravel Framework. Типова послідовність роботи додатку на основі Laravel framework.
104. За що відповідає компонент додатку „модель”?
105. За що відповідає компонент додатку „контролер”?
106. За що відповідає компонент додатку „представлення”?
107. Перерахуйте переваги застосування паттерна MVC при розробці веб-додатків?
108. Опишіть структуру MVC-додатку та його складові частини.
109. Надайте опис схеми розробки веб-додатку з використанням Laravel Framework.
110. Що таке синтаксис Blade. Опишіть його характерні риси та властивості.
111. Назвіть синтаксичні особливості Blade.
112. Наведіть приклад застосування синтаксису Blade для реалізації умовного оператора.
113. Яким чином здійснюється поєднання Blade із статичними елементами HTML?
114. Що таке Модель веб-додатку?
115. Що таке об'єктно-реляційна модель веб-додатку?
116. Які механізми об'єктно-реляційних моделей додатку застосовуються в Laravel Framework?
117. Опишіть спосіб переходу від таблиці реляційної бази даних до програмного класу веб-додатку.
118. Що таке концептуальна модель Eloquent ORM?
119. Що таке модель збереження даних Eloquent ORM?
120. Шаблон Active Record.
121. Назвіть головні принципи побудови прошарку доступу до даних веб-додатку.
122. Назвіть переваги розподілу веб-додатку на прошарки та створення прошарку доступу до даних.

123. Як працює механізм Database First?
124. Як працює механізм Code First?
125. Як працює механізм Model First?
126. Що таке контролер веб-додатку Laravel Framework.
127. За що відповідають контролери веб-додатку Laravel Framework?
128. Назвіть вимоги, які пред'являються при створенні контролера у веб-додатку Laravel Framework.
129. Що таке метод-дія контролера?
130. Які типи значень, що повертають методи-дії, можна використовувати в Laravel Framework?
131. Опишіть порядок виклику контролерів із веб-додатку.
132. Що таке представлення веб-додатку?
133. Опишіть схему виклику представлення у додатку Laravel Framework.
134. Що таке code-behind-файл?
135. Що таке шаблонна сторінка?
136. За що відповідає типізація даних представлення?
137. Що таке часткове представлення?
138. Чим відрізняється представлення, часткове представлення та шаблонна сторінка?
139. Що таке маршрутизація веб-додатку Laravel Framework?
140. Що таке таблиця маршрутизації?
141. Опишіть загальний синтаксис створення маршруту.

### **Методичне забезпечення**

Навчальний процес з дисципліни «WEB-технології та WEB-дизайн» повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою. Зокрема, викладачами кафедри підготовлені і видані такі роботи:

1. Основи проектування та розробка веб-додатків: навчальний посібник / Манзюк Е.А., Лищук О.А., Мазурець О.В., Багрій Р.О., Петровський С.С. – Хмельницький: ХНУ, 2019. – 184 с.
2. Вебтехнології: методичні рекомендації до курсового проектування для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки» / Р. О. Багрій, О. В. Собко, О. О. Тищенко, С. С. Петровський. Хмельницький : ХНУ, 2024. 35 с.

### **Рекомендована література**

#### **Основна**

1. Основи проектування та розробка веб-додатків: навчальний посібник / Манзюк Е.А., Лищук О.А., Мазурець О.В., Багрій Р.О., Петровський С.С. – Хмельницький: ХНУ, 2019. – 184 с.
2. Miller, Brian D., и Jason Ackerman. Principles of Web Design. Allworth, 2022, 280 p.
3. Frain, Ben. Responsive Web Design with HTML5 and CSS: Develop Future-Proof Responsive Websites Using the Latest HTML5 and CSS Techniques, 3rd Edition. Packt Publishing, 2020.
4. DuRocher, David. HTML and CSS QuickStart Guide: The Simplified Beginners Guide to Developing a Strong Coding Foundation, Building Responsive Websites, and Mastering ... of Modern Web Design. ClydeBank Media LLC, 2021.
5. Duckett, Jon. PHP & MySQL: Server-Side Web Development. 1st edition, Wiley, 2022.
6. Duane Noonan. Advanced CSS and Sass: Flexbox, Grid, Animations and More!, Kindle Edition, 2023
7. Laravel – The PHP Framework For Web Artisans. <https://laravel.com/docs/10.x/eloquent-relationships>
8. Drishti Jain. Ultimate Laravel for Modern Web Development. Orange Education Pvt Ltd, 2024
9. Stauffer, Matt. Laravel: Up & Running: A Framework for Building Modern PHP Apps. 2nd edition, O'Reilly Media, 2019



10. PHP BLUEPRINT: Tips and Tricks for Building Modern PHP Apps Kindle Edition, David Mitchell; 1st edition, 2023

11. Вебтехнології: методичні рекомендації до курсового проектування для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 122 «Комп'ютерні науки» / Р. О. Багрій, О. В. Собко, О. О. Тищенко, С. С. Петровський. Хмельницький : ХНУ, 2024. 35 с.

#### **Допоміжна**

12. Vickler, Andy. PHP: 3 Books in 1 : PHP Basics for Beginners + PHP Security and Session Management + Advanced PHP Functions.

13. Tatroe, Kevin, и Peter MacIntyre. Programming PHP: Creating Dynamic Web Pages. 4th edition, O'Reilly Media, 2020.

14. Blum, Richard. PHP, MySQL, & JavaScript All-in-One For Dummies. 1st edition, For Dummies, 2018.

15. Nixon, Robin. Learning PHP, MySQL & JavaScript. 6th edition, O'Reilly Media, 2021

16. Prettyman, Steve. Learn PHP 8: Using MySQL, JavaScript, CSS3, and HTML5. 2nd edition, Apress, 2020.

17. Christopher Pitt. Pro PHP 8 MVC: Model View Controller Architecture-Driven Application Development, Apress; 2nd edition, 2021

18. Stauffer, Matt. Laravel: Up & Running: A Framework for Building Modern PHP Apps. 2nd edition, O'Reilly Media, 2019

#### **Інформаційні ресурси**

1. Модульне середовище для навчання. Доступ до ресурсу: <https://msn.khnu.km.ua>.

2. Електронна бібліотека університету. Доступ до ресурсу: <http://library.khmnu.edu.ua/>

3. Репозитарій ХНУ. Доступ до ресурсу: <http://elar.khmnu.edu.ua/jspui/?locale=uk>