

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ
Кафедра комп'ютерних наук

Соловська І.М., Русу О.П.

ВЕБТЕХНОЛОГІЇ ТА ВЕБДИЗАЙН

Методичні рекомендації до практичних занять
та самостійної роботи здобувачів
галузі знань – 12 Інформаційні технології, спеціальності 121 «Інженерія
програмного забезпечення», 122 «Комп'ютерні науки», 123 «Комп'ютерна
інженерія», 125 «Кібербезпека та захист інформації»,
галузі знань 17 – Електроніка, автоматизація та електронні комунікації,
спеціальність 172 «Електронні комунікації та радіотехніка»,
галузі знань 01 – Освіта/Педагогіка, спеціальність 014 «Середня освіта.
Інформатика»

Одеса – 2025 рік

Рекомендовано Вченою радою Міжнародного гуманітарного університету
(Протокол № 5 від 20.01.2025 р.)

Соловська І.М., Русу О.П.

Методичні рекомендації до практичних занять та самостійної роботи з навчальної дисципліни «Вебтехнології та вебдизайн» для здобувачів вищої освіти першого (бакалаврського) рівня усіх спеціальностей денної та заочної форми навчання [Електронне видання] / Соловська І.М., Русу О.П. Кафедра комп'ютерних наук Міжнародного гуманітарного університету. Одеса. 2025. – 45 с.

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
Кількість кредитів – 9, загальна кількість годин – 270	Галузь – 12 Інформаційні технології Спеціальність – 122 Комп'ютерні науки	Денна форма	Заочна форма
		Обов'язкова	
		Рік підготовки:	
		2,3	
		Семестр:	
Мова навчання – українська	Рівень вищої освіти – перший (бакалаврський)	4,5	
		Лекційні заняття:	
		52	20
		Практичні заняття:	
		66	16
		Лабораторні роботи	
		0	0
		Самостійна робота	
		152	234
		Вид контролю:	
		залік, екзамен	

Дисципліна **«Вебтехнології та вебдизайн»** формує у здобувачів необхідний обсяг теоретичних знань та професійних практичних навиків з ефективного використання основних компонентів Вебтехнологій та Вебдизайну для створення ресурсів в інформаційному просторі і формуванню навичок практичної роботи з пакетами прикладних програм для створення вебсайтів.

Вивчення дисципліни **«Вебтехнології та Вебдизайн»** забезпечує володіння знаннями та практичними навичками щодо мови гіпертекстової розмітки HTML та CSS стилів, базових понять розміщення вебсайту в Інтернет (хостингу та домену), основ безпеки Веб-сайту та основних понять Веб-дизайну, включно з використанням графіки та мультимедіа.

Знання, отриманні при вивченні дисципліни **«Веб-технології та Веб-дизайн»** дозволять бакалаврам використовувати можливості мови HTML та технології CSS для створення вебсторінок, вміти розробляти інформаційні ресурси у вебсередовищі та інтерактивні вебсторінки з використанням засобів графічного моделювання та дизайну.

МЕТА ДИСЦИПЛІНИ. Метою викладання дисципліни є надання знань, умінь та навичок необхідних для пошуку та рішення проблем, генерування гіпотез щодо створення привабливого і функціонального вебсайту, засвоєння сучасних вебтехнологій і суміжних галузей знань та використання інноваційних підходів і творчого мислення під час виконання професійних та творчих вебпроектів.

ПРЕРЕКВІЗИТИ. Даний курс базується на знаннях, отриманих під час вивчення основних дисциплін «Вступ до фаху», «Об'єктно-орієнтоване програмування», «Теорія інформації та кодування» та інших.

ПОСТРЕКВІЗИТИ. Знання, уміння та навички, набуті під час вивчення дисципліни, використовуються при подальшому вивченні дисциплін бакалаврської підготовки, практики та написанні кваліфікаційної роботи.

2. ОЧІКУВАНІ КОМПЕТЕНТНОСТІ, ЯКІ ПЛАНУЄТЬСЯ СФОРМУВАТИ, ТА ДОСЯГНЕННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

У процесі реалізації програми дисципліни «Вебтехнології та вебдизайн» формуються наступні компетентності із передбачених освітньою програмою «Комп'ютерні науки»:

Інтегральна компетентність

Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов.

Загальні компетентності (ЗК)

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК8. Здатність генерувати нові ідеї (креативність).

ЗК9. Здатність працювати в команді.

Спеціальні (фахові) компетентності

СК9. Здатність реалізувати багаторівневу обчислювальну модель на основі архітектури клієнт-сервер, включаючи бази даних, знань і сховища даних, виконувати розподілену обробку великих наборів даних на кластерах стандартних серверів для забезпечення обчислювальних потреб користувачів, у тому числі на хмарних сервісах.

СК11. Здатність до інтелектуального аналізу даних на основі методів обчислювального інтелекту включно з великими та погано структурованими

даними, їхньої оперативної обробки та візуалізації результатів аналізу в процесі розв'язування прикладних задач.

Навчальна дисципліна «Вебтехнології та вебдизайн» забезпечує досягнення програмних результатів навчання (РН), передбачених освітньою програмою:

ПР10. Використовувати інструментальні засоби розробки клієнт-серверних застосувань, 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.

ПР11. Володіти навичками управління життєвим циклом програмного забезпечення, продуктів і сервісів інформаційних технологій відповідно до вимог і обмежень замовника, вміти розробляти проектну документацію (техніко-економічне обґрунтування, технічне завдання, бізнес-план, угоду, договір, контракт).

ПР12. Застосовувати методи та алгоритми обчислювального інтелекту та інтелектуального аналізу даних в задачах класифікації, прогнозування, кластерного аналізу, пошуку асоціативних правил з використанням програмних інструментів підтримки багатовимірного аналізу даних на основі технологій DataMining, TextMining, WebMining.

ЗАПЛАНОВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ ЗА НАВЧАЛЬНОЮ ДИСЦИПЛІНОЮ

Студенти повинні знати:

- основні принципи Вебдизайну, вибір кольорів, побудови композиції та основи типографіки;
- принципи розробки Вебсайту за моделлю AIDA;
- базові Вебтехнології та сучасні засоби, які використовуються при створенні Вебсайту;
- технічні аспекти та послідовність розробки Вебсайту;
- принципи створення сайту за допомогою он-лайн конструктора Вебсайтів;
- функціонування системи управління контентом CMS;
- принципи вибору хостингу Вебсайту та домену;
- основні заходи щодо забезпечення безпеки Вебсайту.

Студенти повинні вміти:

- виконувати дизайн Веб-сайту, використовуючи необхідні засоби кольорової палітри, типографіки та композиції ;
- вміти роботи семантично грамотну розмітку інтернет сторінок за допомоги мови верстки гіпертексту HTML та використовувати сучасні засоби Веб-технологій для розробки Веб-сайтів;
- вміти розробляти зовнішній функціональний дизайн інтернет сторінок за допомоги каскадної таблиці стилю CSS 3;
- проводити розробку макету дизайну Веб-сайту в Figma;
- виконувати розробку Веб-сайту за допомогою конструктора Веб-сайтів (Wix, Weblium, Webflow);
- використовувати бібліотеки, фреймворки, CSS-фреймворк Bootstrap. JS-фреймворки Angular.js, React.js, Vue.js;
- вирішувати завдання хостингу, вибору домену та забезпечити безпеку Веб-сайту,
- вміти підключати зовнішні фреймворки для додання додаткових функціональних можливостей проекту;
- вміти користуватися системи управління контентом CMS;
- вміти встановлювати локальні Веб-сервери для тестування і відладки Веб-проектів;
- вміти встановлювати сучасні системи керування змістом для комерційних і творчих потреб.

3. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Тема 1. Основні поняття Web. URI. URL. URN. Домени. Механізм взаємодії клієнта та веб-сервера. Концепція Web 2.0. Введення в клієнт-серверні технології. Web-сервери та принципи їх роботи з користувачем. Протокол HTTP. Вебсервіси, вебдодатки, вебсервери, WebMining. Протокол HTTP/HTTPS. Забезпечення безпеки передачі даних HTTP.

Тема 2. Базові вебтехнології. HTML. DHTML. CSS. XML. XSLT. DOM. JavaScript. Серверні мови програмування. PHP. ASP. MySQL. Ajax. Сучасні засоби веб-технологій. Бібліотеки. Фреймворки. Інструменти. CSS-фреймворк Bootstrap. JS-фреймворки. Angular.js. React.js. Vue.js. Ember.js. Meteor.js. JS-бібліотеки. Бібліотека jQuery. Використання баз даних, знань і сховищ даних, хмарних сервісів.

Тема 3. Основи Вебдизайну

Використання графічних файлів для Веб-сайтів (векторні SVG та растрові GIF, JPEG, PNG-8, 24 файли). Анімаційні Веб-формати (APNG, анімований PNG), WebP, FLIF. Формати PSD, TIFF, BMP, PDF, WMF, 6AI, CRD. Зображення, що створюються програмним шляхом CSS, SVG, Canvas, JS-анімація.

Тема 4. Основи HTML/CSS

Структура DOM. Редактори коду. Блокова верстка. Основні семантичні елементи HTML. Розмітка сторінки з точки зору семантики. HTML і CSS код. Вимоги до HTML та CSS-коду.

Тема 3. Основні етапи розробки вебсайту.

Система управління контентом CMS. Впровадження CMS. Огляд безкоштовних CMS (WordPress, Joomla, Drupal, TYPO3, MODX, Opencart, Magento). Огляд комерційних CMS («Управління сайтом», CS Cart, DataLife Engine, NetCat). Розробка технічного завдання, бізнес-плану, угоди, договору для створення вебсайту. Програмні інструменти на основі технологій DataMining, TextMining, WebMining.

Тема 4. Хостинг, домен та безпека вебсайту.

Використання сховищ та вітрин даних, хмарних сервісів. Загрози конфіденційності, загрози цілісності, загрози доступності. Види атак на веб-додатки: Метод ін'єкцій, Міжсайтовий скриптинг XSS (Cross-Site Scripting), Відмова в обслуговуванні DoS (Denial of Service), DDoS – (Distributed Denial of Service).

4. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Семестр 4

Назви змістових модулів і тем	Кількість годин									
	Денна форма					Заочна форма				
	Заг	У тому числі				Заг	У тому числі			
		Лек	Пра к	Ла б	Сам роб.		Лек	Пра к	Ла б	Сам роб
Тема 1. Основні поняття Web. URI. URL. URN. Домени. Механізм взаємодії клієнта та веб-сервера. Концепція Web 2.0. Введення в клієнт-серверні технології. Web-сервери та принципи їх роботи з користувачем. Протокол HTTP. Вебсервіси, вебдодатки, вебсервери, WebMining. Протокол HTTP/HTTPS. Забезпечення безпеки передачі даних HTTP.	40	6		10	24	40	4			36
Тема 2. Базові вебтехнології. HTML. DHTML. CSS. XML. XSLT. DOM. JavaScript. Серверні мови програмування. PHP. ASP. MySQL. Ajax. Сучасні засоби веб-технологій. Бібліотеки. Фреймворки. Інструменти. CSS-фреймворк Bootstrap. JS-фреймворки. Angular.js. React.js. Vue.js. Ember.js. Meteor.js. JS-бібліотеки.	40	6		10	24	40	2			38

Семестр 5

Назви змістових модулів і тем	Кількість годин									
	Денна форма					Заочна форма				
	Заг.	У тому числі				Заг.	У тому числі			
		Ле к	Пр ак	Ла б	Са м. Роб .		Ле к	Пр ак	Ла б	Са м. Роб .
Тема 3. Основні етапи розробки вебсайту. Система управління контентом CMS. Впровадження CMS. Огляд безкоштовних CMS (WordPress, Joomla, Drupal, TYPO3, MODX, Opencart, Magento). Огляд комерційних CMS («Управління сайтом», CS Cart, DataLife Engine, NetCat). Розробка технічного завдання, бізнес-плану, угоди, договору для створення вебсайту. Програмні інструменти на основі технологій DataMining, TextMining, WebMining.	60	14	14		32	60	4	4		52
Тема 4. Хостинг, домен та безпека вебсайту. Використання сховищ та вітрин даних, хмарних сервісів. Загрози конфіденційності, загрози цілісності, загрози доступності. Види атак на веб-додатки: Метод	60	12	12		36	60	4	4		52

ін'єкцій, Міжсайтовий скриптинг XSS (Cross-Site Scripting), Відмова в обслуговуванні DoS (Denial of Service), DDoS – (Distributed Denial of Service).										
	120	26	26		68	120	8	8		104
ПІДСУМКОВИЙ КОНТРОЛЬ – ЕКЗАМЕН										

5. ПИТАННЯ ДО ПРАКТИЧНИХ ЗАНЯТЬ

Семестр 4

Назва теми	Кількість годин	
	Денна форма	Заочна форма
Тема 1. Основні поняття Web. Введення в клієнт-серверні технології. Web-сервери та принципи їх роботи з користувачем. Вебсервіси, вебдодатки, вебсервери, WebMining. Протокол HTTP/HTTPS. Забезпечення безпеки передачі даних HTTP.	24	36
Тема 2. Базові вебтехнології. Сучасні засоби веб-технологій. Бібліотеки. Фреймворки. Інструменти. CSS-фреймворк Bootstrap. JS-фреймворки.	24	38
Тема 3. Основи Вебдизайну Анімаційні Веб-формати (APNG, анімований PNG), WebP, FLIF. Формати PSD, TIFF, BMP, PDF, WMF, 6AI, CRD. Зображення, що створюються програмним шляхом CSS, SVG, Canvas, JS-анімація.	14	24
Тема 4. Основи HTML/CSS Структура DOM. Основні семантичні елементи HTML. Розмітка сторінки з точки зору семантики. HTML і CSS код. Вимоги до HTML та CSS-коду.	22	32
Загалом	84	130

Семестр 5

Назва теми	Кількість годин	
	Денна форма	Заочна форма
Тема 3. Основні етапи розробки Web-сайту. Система управління контентом CMS (WordPress, Joomla, Drupal, TYPO3, MODX, Opencart, Magento). Огляд комерційних CMS (CS Cart, DataLife Engine, NetCat).	32	52
Тема 4. Хостинг, домен та безпека Веб-сайту. Загрози конфіденційності, загрози цілісності, загрози доступності. Відмова в обслуговуванні DoS (Denial of Service), DDoS – (Distributed Denial of Service).	36	52
	68	104

6. САМОСТІЙНА РОБОТА

До самостійної роботи студентів відносяться наступні види діяльності:

1. Ознайомлення з науковою та навчальною літературою відповідно зазначених у програмі тем.
2. Опрацювання лекційного матеріалу.
3. Підготовка до практичних занять.
4. Консультації з викладачем протягом семестру.
5. Самостійне опрацювання окремих питань навчальної дисципліни.
6. Підготовка та виконання індивідуальних завдань у вигляді есе, рефератів тощо.
7. Підготовка до підсумкового контролю.

Семестр 4

Назва теми	Кількість годин	
	Денна форма	Заочна форма
Тема 1. Основні поняття Web. URI. URL. URN. Домени. Механізм взаємодії клієнта та веб-сервера. Концепція Web 2.0. Введення в клієнт-серверні технології. Web-сервери та принципи їх роботи з користувачем. Протокол HTTP.	24	36

Вебсервіси, вебдодатки, вебсервери, WebMining. Протокол HTTP/HTTPS. Забезпечення безпеки передачі даних HTTP.		
Тема 2. Базові вебтехнології. HTML. DHTML. CSS. XML. XSLT. DOM. JavaScript. Серверні мови програмування. PHP. ASP. MySQL. Аjax. Сучасні засоби веб-технологій. Бібліотеки. Фреймворки. Інструменти. CSS-фреймворк Bootstrap. JS-фреймворки. Angular.js. React.js. Vue.js. Ember.js. Meteor.js. JS-бібліотеки. Бібліотека jQuery.	24	38
Тема 3. Основи Вебдизайну Використання графічних файлів для Веб-сайтів (векторні SVG та растрові GIF, JPEG, PNG-8, 24 файли). Анімаційні Веб-формати (APNG, анімований PNG), WebP, FLIF. Формати PSD, TIFF, BMP, PDF, WMF, 6AI, CRD. Зображення, що створюються програмним шляхом CSS, SVG, Canvas, JS-анімація.	14	24
Тема 4. Основи HTML/CSS Структура DOM. Редактори коду. Блокова верстка. Основні семантичні елементи HTML. Розмітка сторінки з точки зору семантики. HTML і CSS код. Вимоги до HTML та CSS-коду.	22	32
Загалом	84	130

Семестр 5

Назва теми	Кількість годин	
	Денна форма	Заочна форма
Тема 3. Основні етапи розробки Web-сайту. Система управління контентом CMS. Впровадження CMS. Огляд безкоштовних CMS (WordPress, Joomla, Drupal, TYPO3, MODX, Opencart, Magento). Огляд комерційних CMS («Управління сайтом», CS Cart, DataLife Engine, NetCat).	32	52
Тема 4. Хостинг, домен та безпека Вебсайту. Загрози конфіденційності, загрози цілісності, загрози доступності. Види атак на веб-додатки: Метод ін'єкцій, Міжсайтовий скриптинг XSS (Cross-Site Scripting), Відмова в обслуговуванні DoS (Denial of Service), DDoS – (Distributed Denial of Service).	36	52
	68	104

7. ВИДИ ТА МЕТОДИ КОНТРОЛЮ

Види контролю	Складові оцінювання	
	Для екзамену	Для заліку
поточний контроль , який здійснюється у ході: проведення практичних (лабораторних) занять, виконання самостійної роботи, індивідуальних завдань, дослідна робота здобувача тощо.	50%	100%
підсумковий контроль	50%	

Методи діагностики знань (контролю)	фронтальне опитування; наукова доповідь, реферати, усне повідомлення, індивідуальне опитування; робота у групах; ділова гра, розв'язання ситуаційних завдань, кейсів, практичних завдань, залік
--	---

ПИТАННЯ ДЛЯ ПІДСУМКОВОГО КОНТРОЛЮ

1. Поясніть особливості Веб-дизайну та які заходи використовуються для забезпечення таких особливостей.
2. Вкажіть основні правила Веб-дизайну.
3. Наведіть приклади використання психології кольорів у Веб-дизайні.
4. Як складається колірна палітра для RGB та CMYK.
5. Як використовується типографіка у Веб-дизайні.
6. Поясніть поняття AIDA та його використання у Веб-дизайні.
7. Наведіть приклади Веб-дизайну з використанням запиту до дії Call to Action. Вкажіть необхідність використання схеми перегляду сторінки, візуальних напрямків.
8. Визначення та призначення HTML. Структура HTML-документа.
9. Форматування тексту засобами HTML та CSS. Наведіть приклад.
10. Виконайте порівняння фіксованого та еластичного макету.
11. Поясніть поняття: чуйний і адаптивний Веб-дизайн.
12. Наведіть основні етапи розробки сайту. Поясніть кожен етап.
13. Які функції виконує система керування контентом CMS. Надайте приклади.
14. Поясніть відмінності використання он-лайн конструкторів для розробки сайтів. Вкажіть основні переваги.
15. Порівняйте між собою різні види Веб-сайтів, приведіть приклади лендінга, e-commerce та інтернет-магазину.

16. Як виконується дослідження, які завдання використовуються для проведення дослідження.

17. Дайте визначення хостингу. Як обирається хостинг та за якими критеріями.

18. Надайте класифікацію видів хостингів.

19. Вкажіть, як обирається домен.

20. Інформаційне наповнення сайту. Використання копірайтингу та рерайтингу, оптимізовані SEO-тексти, копіпаст та плагіат.

21. Інфографіка (аналітична, новинна, реконструкція, презентація).

22. Веб-анімація. Створення, анімація процесів та даних.

23. Системи безпеки Веб-сайту. Захист від DDOS-атак.

24. Вкажіть основні способи захисту від атак на Веб-сайт.

25. Порівняйте види атак та способи захисту для методу ін'єкцій, міжсайтового скриптингу CSS та відмови в обслуговуванні DDoS.

26. Поясніть структуру Веб-сторінки з використанням блочних елементів.

27. Яке призначення HTML-тегів?

28. Чим відрізняються парні HTML-теги від непарних?

29. Як виглядає базова структура HTML документа?

30. Які елементи Html-документа є обов'язковими?

31. Вкажіть, які спеціальні символи використовуються в HTML?

32. Які HTML-списки і як саме можна подавати в Html-документах?

33. Як вставити в Html-документ графічні зображення?

34. Як визначити розмір вертикального відступу від тексту до границі рисунка?

35. Як визначити розмір горизонтального відступу від тексту до границі рисунка?

36. Як визначити гіперпосилання на рисунок?

37. Як визначити гіперпосилання у середині HTML-документа?

38. Як створити HTML-нумерований та маркирований список?

39. Наведіть приклад вкладеного HTML-списку.

40. Як створити HTML-таблицю?

41. Навіщо використовуються фрейми, як визначити кількість фреймів?

42. Як відрізняються зовнішні та внутрішні каскадні таблиці стилів?

43. Поясніть принципи роботи селекторів CSS.

44. Селектори, специфічність і каскадування.

45. За яким принципом створюються таблиці CSS?

46. Як підключити CSS файл?

47. Які сучасні методи розмітки існують і в чому їх переваги?
48. Наведіть приклад використання CSS для тексту.
49. Наведіть приклад використання CSS для списків.
50. Переходи та трансформація в CSS. Анімація.
51. Блокові та рядкові елементи. Властивості блокових елементів.
52. Використання фреймворку Bootstrap.
53. Використання JS-фреймворів Angular.js, React.js, Vue.js.
54. Верстка веб-сайту. Елементи HTML5-макету. Правила верстки.
55. Візуальне подання HTML-елементів на веб-сторінці.
56. Види верстки веб-сайту. Принципи адаптивної верстки.
57. Система управління контентом CMS, принципи впровадження та переваги.
58. Безпека вебсайту. Види атак на веб-додатки, метод ін'єкцій, міжсайтовий скриптинг, відмова в обслуговуванні DDoS .
59. Хостинг вебсайту, особливості та обґрунтування.
60. Вибір домену вебсайту.

8. ОЦІНЮВАННЯ ПОТОЧНОЇ, САМОСТІЙНОЇ ТА ІНДИВІДУАЛЬНОЇ РОБОТИ ЗДОБУВАЧІВ

Види роботи	Планові терміни виконання	Форми контролю та звітності	Максимальний відсоток оцінювання
Систематичність і активність роботи на практичних заняттях			
1.1. Підготовка до практичних занять	Відповідно до робочої програми та розкладу занять	Перевірка обсягу та якості засвоєного матеріалу під час практичних занять	60
Виконання завдань для самостійного опрацювання			
1.2. Індивідуальне тестування	Відповідно до робочої програми та розкладу занять	Перевірка результатів індивідуального тестування, перевірка конспектів лекцій, перевірка рефератів	20
Виконання індивідуальних завдань (дослідна робота здобувача)			
1.3. Інші види індивідуальних завдань, в т.ч. підготовка наукових публікацій, участь у роботі круглих столів, конференцій тощо.	Відповідно до робочої програми та розкладу занять	Обговорення результатів проведеної роботи під час аудиторних занять, наукових конференцій та круглих столів.	20
Разом балів за поточний контроль			100
Загальний результат оцінювання			100

Поточний контроль знань здобувачів освіти при формі контролю залік (максимум 100 балів) оцінюється за шкалою («2», «3», «4», «5»), з обов'язковим переведенням балів за кожний вид роботи таким чином:

– за підготовку до аудиторних занять (максимум 60 балів) здобувачу освіти необхідно мати не менше 1/2 оцінок від загальної кількості практичних

(лабораторних) занять, для перевodu балів за підготовку до аудиторних занять середньоарифметична оцінка множиться на коефіцієнт 12;

– за виконання завдань для самостійного опрацювання (тестування, завдання, підготовка реферату (есе) за заданою тематикою) (максимум 20 балів) здобувачу освіти необхідно мати не менше $1/2$ оцінок від загальної кількості програмного матеріалу, що виноситься на самостійне вивчення, для перевodu зазначених балів середньоарифметична оцінка множиться на коефіцієнт 4;

– за інші види індивідуальних завдань, в т.ч. підготовку наукових публікацій, участь у роботі круглих столів, конференцій тощо (максимум 20 балів) здобувачу освіти необхідно мати не менше $1/2$ оцінок від загальної кількості інших видів індивідуальних завдань, для перевodu балів за підготовку до аудиторних занять середньоарифметична оцінка множиться на коефіцієнт 4.

Критерії оцінювання поточного контролю

– «2» – виставляється за неправильну відповідь (письмову роботу), яка не відповідає змісту теми та свідчить про нерозуміння її основних положень;

– «3» – виставляється за відповідь (письмову роботу), яка відповідає змісту теми, але є неповною і неточною у визначенні понять, свідчить про неповне розуміння основних положень навчального матеріалу, свідчить про те, що знання здобувача мають фрагментарний, поверховий характер;

– «4» – оцінюється правильна і обґрунтована відповідь (письмова робота), з якої видно, що здобувач знає й розуміє теоретичний матеріал в обсязі навчальної теми, володіє необхідними навичками, не допускає суттєвих помилок.

– «5» – оцінюється повна, правильна, ґрунтовна відповідь (письмова робота) на запитання теми, що передбачає логічність і послідовність викладу матеріалу, володіння термінологією, демонструє вміння повно й глибоко використовувати теоретичні знання у практичній площині.

9. ОЦІНЮВАННЯ ЗАГАЛЬНИХ РЕЗУЛЬТАТІВ ЗНАНЬ ЗДОБУВАЧІВ

Загальні результати знань здобувачів по даній дисципліні оцінюються наступним чином:

– «Відмінно»/«зараховано» (А) – від 90 до 100 балів. Здобувач виявляє особливі творчі здібності, вміє самостійно знаходити та опрацьовувати необхідну інформацію, демонструє знання матеріалу, проводить узагальнення і висновки. Був присутній на лекціях та семінарських заняттях, під час яких давав вичерпні, обґрунтовані, теоретично і практично правильні відповіді, має конспект з

виконаними завданнями до самостійної роботи, презентував реферат за заданою тематикою, проявляє активність і творчість у науково-дослідній роботі;

– **«Добре»/«зараховано» (В)** – від 82 до 89 балів. Здобувач володіє знаннями матеріалу, але допускає незначні помилки у формуванні термінів, категорій, проте за допомогою викладача швидко орієнтується і знаходить правильні відповіді. Був присутній на лекціях та семінарських заняттях, має конспект з виконаними завданнями до самостійної роботи, презентував реферат за заданою тематикою, проявляє активність і творчість у науково-дослідній роботі;

– **«Добре»/«зараховано» (С)** – від 74 до 81 балів. Здобувач відтворює значну частину теоретичного матеріалу, виявляє знання і розуміння основних положень, з допомогою викладача може аналізувати навчальний матеріал, але дає недостатньо обґрунтовані, невичерпні відповіді, допускає помилки. При цьому враховується наявність конспекту з виконаними завданнями до самостійної роботи, реферату та активність у науково-дослідній роботі;

– **«Задовільно»/«зараховано» (D)** – від 64 до 73 балів. Здобувач був присутній не на всіх лекціях та семінарських заняттях, володіє навчальним матеріалом на середньому рівні, допускає помилки, серед яких є значна кількість суттєвих. При цьому враховується наявність конспекту з виконаними завданнями до самостійної роботи та рефератів;

– **«Задовільно»/«зараховано» (E)** – від 60 до 63 балів. Здобувач був присутній не на всіх лекціях та семінарських заняттях, володіє навчальним матеріалом на рівні, вищому за початковий, значну частину його відтворює на репродуктивному рівні, на всі запитання дає необґрунтовані, невичерпні відповіді, допускає помилки, має неповний конспект з завданнями до самостійної роботи.

– **«Незадовільно з можливістю повторного складання»/«не зараховано» (FX)** – від 35 до 59 балів. Здобувач володіє матеріалом на рівні окремих фрагментів, що становлять незначну частину навчального матеріалу.

– **«Незадовільно з обов’язковим повторним вивченням дисципліни»/«не зараховано» (F)** – від 1 до 34 балів. Здобувач не володіє навчальним матеріалом.

ТАБЛИЦЯ ВІДПОВІДНОСТІ РЕЗУЛЬТАТІВ КОНТРОЛЮ ЗНАНЬ ЗА РІЗНИМИ ШКАЛАМИ

100-бальна шкала	Шкала за ECTS	Національна шкала	
		Екзамен	Залік
90-100	A	Відмінно	Зараховано
82-89	B	Добре	Зараховано
74-81	C		
64-73	D	Задовільно	Зараховано
60-63	E		
35-59	FX	Незадовільно	Не зараховано
1-34	F		

10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Соловська І.М., Русу О.П. Веб-технології та Веб-дизайн: конспект лекцій для здобувачів, які навчаються за першим (бакалаврським) рівнем вищої освіти в галузі знань – 12 Інформаційні технології, спеціальності 121 «Інженерія програмного забезпечення», 122 «Комп’ютерні науки», 123 «Комп’ютерна інженерія», 125 «Кібербезпека та захист інформації», галузі знань 17 – Електроніка, автоматизація та електронні комунікації, спеціальність 172 «Електронні комунікації та радіотехніка», 01 – Освіта/Педагогіка, спеціальність 014 «Середня освіта. Інформатика» [Електронне видання] / Соловська І.М., Русу О.П. Кафедра комп’ютерних наук Міжнародного гуманітарного університету. Одеса, 2025. – 65 с.

2. Соловська І.М., Русу О.П. Навчально-методичні рекомендації до практичних занять та самостійної роботи з навчальної дисципліни «Веб-технології та Веб-дизайн» для здобувачів вищої освіти першого (бакалаврського) рівня спеціальностей 121 «Інженерія програмного забезпечення», 122 «Комп’ютерні науки», 123 «Комп’ютерна інженерія», 125 «Кібербезпека та захист інформації», 172 «Електронні комунікації та радіотехніка», 014 «Середня освіта. Інформатика» денної та заочної форми навчання [Електронне видання] / Соловська І.М., Русу О.П. Кафедра комп’ютерних наук Міжнародного гуманітарного університету. Одеса. 2025. – 45 с.

3. Фрімен Е., Робсон Е. Head First Патерни проєктування. Харків: Фабула, 2020, 688 с.

4. Бородкіна І.Л., Бородкін Г.О. Веб-технології та Веб-дизайн. Ліра – К, 2020, 212 с.
5. Двірничук К.В., Вацек Д.О. Веб-програмування та веб-дизайн : навч. посіб. Чернівці : Чернівець. нац. ун-т ім. Ю. Федьковича, 2022. 472 с.
6. Веб-технології та веб-дизайн: навч. посібник. / Трофименко О.Г., Козін О.Б., Задерейко О.В., Плачінда О.Є. Одеса: Фенікс, 2019. 278 с.
7. Комп'ютерні мережі. Частина 1: навчальний посібник / Б. Ю. Жураковський, І. О. Зенів. Київ: КПІ, 2020. 336 с.
8. Лобода Ю. Г. Вебтехнології та вебдизайн: навчально-методичний посібник для здобувачів вищої освіти галузі знань 12 «Інформаційні технології» [Електронне видання] / уклад.: Ю. Г. Лобода, А. А. Толокнов ; НУ «ОЮА». – Одеса : Фенікс, 2023. – 260 с.
9. Kyrnin J., Meloni J. HTML, CSS, and JavaScript All in One, Sams Teach Yourself: Covering HTML5. London: Pearson, 2018. 800 p.
10. Sean McManus Web Design in Easy Steps 6th Edition, In Easy Steps, 2020.
11. Duckett Jon. HTML and CSS: Design and Build Websites. 2021. 512 p.

Допоміжна

1. Стрелковська І.В., Соловська І.М. Сплاین-апроксимація в 3D-моделюванні VIII Всеукраїнська науково-практична конференція студентів, аспірантів та молодих учених «Гуманітарний і інноваційний ракурс професійної майстерності: Пошуки молодих вчених»: матеріали конф., 18 листопада 2022 р.: тези доп. – Одеса: МГУ, 2022. – С. 390-394. <https://doi.org/10.36059/978966-397-266-4/116>
2. Стрелковська І. В., Соловська І. М., Снігур Н., Малюга В., Параметричні сплайни в 3D-моделюванні. Міжнародна конференція «Передові технології в інформаційно-комунікаційній інженерії»: матеріали конф., 17-20 липня 2023р.: тези доп. – Одеса: МГУ, 2023. – С. 22-26.
3. Веб-технології та веб-дизайн: навчальний посібник: у 2-х частинах /О.М. Яцько, А.Я. Довгунь, Ю.Я. Томка. – Чернівці: Чернівецький нац. ун-т ім. Ю.Федьковича, 2018. – Частина 1. – 296 с.
4. Юскович-Жуковська В. І., Богут О. М. Ю-89 Web-програмування : Підручник. – Рівне : Волин. обереги, 2023. – 384 с.

Інформаційні ресурси

1. <http://www.w3schools.com/html/default.asp> – HTML Tutorial.
2. http://www.w3schools.com/html/html_colors.asp – HTML Colors.
3. <https://www.w3.org/TR/html5> – W3C. HTML5. A vocabulary and associated APIs for HTML and XHTML
4. <https://clockwise.software/blog/web-development-trends/> – Web Development Trends and the Latest Web Technology Stacks
5. <http://uk.wikipedia.org/wiki/HTML> – HTML: Вікіпідручник.
6. <http://www.w3schools.com/css/> – CSS Tutorial.

ДОДАТОК А. КОНСПЕКТ ЛЕКЦІЙ

Тема 1. Основні поняття Web

Всесвітня павутина WWW (World Wide Web) є глобальною інформаційною системою, що функціонує в мережі Інтернет і забезпечує доступ до різноманітних інформаційних ресурсів. Вона складається з великої кількості взаємопов'язаних документів і сервісів, які можуть містити текст, графіку, відео, аудіо та інтерактивні елементи. Основою функціонування вебу є взаємодія клієнтських програм, які називаються веб-браузерами, та серверів, на яких зберігаються вебресурси. Користувач отримує доступ до вебсторінок через спеціальні програми – браузери, які відправляють запити до серверів та відображають отриманий контент.

Одним із ключових понять є URI (Uniform Resource Identifier), універсальний ідентифікатор ресурсу. Він використовується для ідентифікації будь-якого ресурсу в мережі Інтернет. URI може мати дві основні форми: URL та URN. URL (Uniform Resource Locator) є адресою ресурсу, що визначає його місцезнаходження в мережі. Така адреса містить інформацію про протокол доступу, доменне ім'я сервера та шлях до файлу або сторінки. Наприклад, адреса вебсайту може містити протокол HTTPS, доменне ім'я та шлях до конкретної сторінки. URN (Uniform Resource Name), на відміну від URL, використовується для ідентифікації ресурсу за його унікальним ім'ям незалежно від фізичного розташування.

Важливим елементом функціонування вебзастосунків є доменні імена. Домен – це текстове представлення адреси вебсайту, яке використовується замість числових IP-адрес. Домени мають ієрархічну структуру. Найвищий рівень займають домени верхнього рівня, такі як .com, .org, .net або національні домени, наприклад .ua. Домен другого рівня зазвичай є назвою організації або бренду. Завдяки системі доменних імен DNS (Domain Name System) відбувається перетворення доменних імен у відповідні IP-адреси, що дозволяє комп'ютерам знаходити один одного в мережі.

Функціонування вебзастосунків базується на моделі клієнт-сервер. У цій моделі клієнтська програма, зазвичай браузер користувача, надсилає запит до вебсервера. Сервер обробляє цей запит і надсилає відповідь у вигляді вебсторінки або іншого ресурсу. Після отримання відповіді браузер відображає інформацію у зрозумілому для користувача вигляді. Така архітектура дозволяє централізовано зберігати дані на сервері та забезпечувати доступ до них великій кількості користувачів.

Сучасний розвиток Інтернету пов'язаний із концепцією Web 2.0. Вона характеризується переходом від статичних вебсторінок до інтерактивних вебсервісів, де користувачі можуть не лише споживати інформацію, а й створювати власний контент. Саме завдяки Web 2.0 з'явилися соціальні мережі, блоги, вікі-проекти та онлайн-платформи для спільної роботи. У таких системах користувачі можуть коментувати, публікувати інформацію, спілкуватися та взаємодіяти між собою.

Для роботи вебресурсів використовуються вебсервери. Вебсервер є програмним забезпеченням або комп'ютером, який приймає запити від клієнтів і надає необхідні ресурси. Найбільш поширеними вебсерверами є Apache, Nginx та Microsoft IIS. Основною функцією вебсервера є обробка HTTP-запитів, передача вебсторінок, а також взаємодія із серверними програмами та базами даних.

Передача інформації у Веб здійснюється за допомогою протоколу HTTP (HyperText Transfer Protocol). Цей протокол визначає правила обміну даними між клієнтом і сервером. HTTP працює за принципом «запит-відповідь». Коли користувач відкриває вебсторінку, браузер надсилає запит на сервер, після чого сервер надсилає відповідь, що містить HTML-код сторінки. У запитах можуть використовуватися різні методи, наприклад GET для отримання інформації або POST для передачі даних на сервер.

Для підвищення безпеки передачі даних використовується протокол HTTPS. Він є розширенням HTTP і використовує криптографічні протоколи SSL або TLS для шифрування інформації. Завдяки цьому дані, що передаються між браузером і сервером, не можуть бути перехоплені або змінені сторонніми особами. HTTPS широко використовується на сучасних вебсайтах, особливо там, де передаються конфіденційні дані, такі як паролі або банківська інформація.

Сучасний веб також включає вебсервіси та вебдодатки. Вебсервіси являють собою програмні інтерфейси, які дозволяють різним програмам обмінюватися даними через Інтернет. Вебдодатки, у свою чергу, є повноцінними програмами, що працюють у браузері та виконують складні функції, наприклад редагування документів або управління даними. Окрім цього, важливим напрямком є Web Mining – аналіз інформації, що міститься у WWW. Він дозволяє виявляти закономірності у поведінці користувачів, аналізувати структуру вебсайтів та обробляти великі обсяги даних.

Тема 2. Базові вебтехнології

Основою створення вебсайтів є низка базових технологій, які забезпечують структуру, оформлення та функціональність вебсторінок. Найважливішою з них є мова розмітки HTML (HyperText Markup Language). Вона використовується для створення структури вебсторінок і визначає, які елементи будуть відображатися на екрані. HTML складається з тегів, що описують різні частини сторінки, такі як заголовки, абзаци, зображення та посилання.

Для оформлення зовнішнього вигляду сторінок використовується мова стилів CSS (Cascading Style Sheets). CSS дозволяє задавати кольори, шрифти, відступи, розміщення елементів та інші параметри дизайну. Завдяки CSS можна створювати адаптивні вебсайти, які коректно відображаються на різних пристроях, зокрема на смартфонах, планшетах і комп'ютерах.

DHTML (Dynamic HTML) є сукупністю технологій, що дозволяють створювати динамічні вебсторінки. Він поєднує HTML, CSS, JavaScript та DOM і дозволяє змінювати структуру сторінки без її повного перезавантаження. Однією з важливих технологій для роботи з даними є XML — мова розмітки, призначена для зберігання та передачі структурованої інформації. XML широко використовується для обміну даними між різними системами.

Для перетворення XML-документів у різні формати застосовується технологія XSLT. Вона дозволяє змінювати структуру даних і формувати з них, наприклад, HTML-сторінки. Важливою складовою роботи вебсторінок є DOM (Document Object Model). Це програмна модель документа, яка представляє HTML або XML у вигляді дерева об'єктів. Завдяки DOM програмісти можуть змінювати структуру сторінки за допомогою JavaScript.

JavaScript є однією з основних мов програмування для вебу. Вона використовується для створення інтерактивних елементів, перевірки введених даних у формах, реалізації анімацій та інших динамічних функцій. JavaScript працює безпосередньо у браузері та дозволяє швидко реагувати на дії користувача.

Для обробки даних на сервері застосовуються серверні мови програмування, серед яких найбільш поширеними є PHP та ASP. Вони дозволяють створювати динамічні вебсторінки, працювати з базами даних та виконувати складні обчислення. Однією з найпопулярніших систем керування базами даних у веброзробці є MySQL. Вона використовується для зберігання інформації про користувачів, товари, статті та інші дані.

AJAX є технологією, що дозволяє здійснювати асинхронний обмін даними між браузером і сервером. Це означає, що вебсторінка може оновлювати окремі елементи без повного перезавантаження. Завдяки цьому сучасні вебдодатки працюють швидше та зручніше для користувача.

Для спрощення розробки використовуються спеціальні бібліотеки та фреймворки. Бібліотеки містять набір готових функцій, які допомагають швидше реалізувати необхідну функціональність. Однією з найвідоміших бібліотек є jQuery. Фреймворки, у свою чергу, надають повноцінну структуру для створення вебдодатків. Серед популярних JavaScript-фреймворків можна виділити Angular, React, Vue, Ember та Meteor.

Ще одним важливим інструментом є CSS-фреймворк Bootstrap. Він містить готові стилі, компоненти та адаптивну сіткову систему, що значно спрощує створення сучасних вебінтерфейсів. Окрім цього, сучасні вебсистеми активно використовують хмарні сервіси, які дозволяють зберігати дані, масштабувати ресурси та забезпечувати стабільну роботу вебдодатків.

Тема 3. Основи вебдизайну

Вебдизайн є важливою складовою створення вебсайтів, оскільки саме від нього залежить зручність використання та привабливість ресурсу для користувачів. Одним із ключових елементів вебдизайну є використання графічних файлів. У вебі застосовуються як растрові, так і векторні формати зображень. Растрові зображення складаються з пікселів і використовуються для фотографій та складної графіки. До таких форматів належать JPEG, PNG та GIF. Формат JPEG забезпечує хорошу компресію фотографій, PNG підтримує прозорість і високу якість, а GIF часто використовується для простих анімацій.

Векторна графіка представлена форматом SVG. Вона описується математичними формулами, тому може масштабуватися без втрати якості. Це робить SVG ідеальним для логотипів, іконок та інших графічних елементів інтерфейсу.

Для створення анімації у вебі використовуються різні формати, зокрема анімований GIF, APNG та WebP. Вони дозволяють створювати динамічні зображення, що використовуються у рекламі, інтерфейсах та інтерактивних елементах. Окрім цього, у веброзробці застосовуються формати, створені графічними редакторами, такі як PSD, TIFF, BMP, PDF, AI або CDR. Вони використовуються на етапі дизайну і підготовки макетів.

Сучасні вебсайти також можуть створювати графіку програмним шляхом. Для цього використовуються технології CSS, SVG, Canvas та JavaScript-анімації. Canvas дозволяє малювати графіку безпосередньо у браузері, що відкриває можливості для створення ігор, інтерактивних візуалізацій та складних анімацій.

Тема 4. Основи HTML та CSS

При створенні вебсторінок важливу роль відіграє структура DOM. DOM представляє HTML-документ у вигляді дерева об'єктів, де кожен елемент сторінки є окремим вузлом. Така структура дозволяє програмам взаємодіяти зі сторінкою, змінювати її елементи та реагувати на події користувача.

Для написання коду використовуються різноманітні редактори. Серед популярних редакторів можна виділити Visual Studio Code, Sublime Text та WebStorm. Вони надають інструменти для підсвічування синтаксису, автодоповнення та перевірки помилок.

Одним із основних підходів до створення макетів вебсторінок є блокова верстка. Вона базується на використанні блокових елементів HTML, які дозволяють організувати структуру сторінки у вигляді окремих секцій. Семантичні елементи HTML, такі як header, nav, section, article, footer, допомагають описати зміст сторінки та покращують її доступність і оптимізацію для пошукових систем.

Якісний HTML та CSS код повинен відповідати певним вимогам. Він має бути структурованим, зрозумілим і відповідати стандартам веброзробки. Важливо використовувати правильну семантику, уникати зайвих елементів та дотримуватися принципів чистого коду.

Тема 5. Основні етапи розробки вебсайту

Процес створення вебсайту складається з кількох послідовних етапів.

На першому етапі відбувається визначення цілей створення сайту та проведення досліджень за тематикою сайту. Складання технічного завдання є другим етапом розробки сайту, технічне завдання відображає всі поставлені завдання. У процесі створення такого проекту проводиться поетапне узгодження виконуваних робіт, що дозволяє зробити сайт, який відповідає всім вимогам клієнта. На цьому етапі розробник спільно з замовником визначаються з умовами щодо реалізації проекту.

Технічне завдання має містити такі пункти:

- тип сайту (лендинг, візитка, каталог, корпоративний, магазин тощо);

- структура вебсайту, сторінки, переходи та файлова структура;
- функціонал вебсайту (пошук, каталог, стрічка новин);
- стиль дизайну (витриманий, авангардний, в темних тонах), колірна гама, потреба у створенні нового бренду (логотип, слогани, фірмові кольори та шрифти).

Серед технічних аспектів проектування вебсайту виконується вибір архітектури сайту, файлової та логічної структури сторінок.

Архітектура вебсайту є основною організацією клієнтської та серверної частин вебсайту, їх взаємозв'язок та зв'язок з кінцевим користувачем. Створення архітектури вебсайту є першочерговим етапом в його проектуванні.

Файлова структура - це чітко оформлена система організації файлів. Продумана і зручна файлова структура допомагає розробнику оптимізувати роботу, а також буде зрозумілою для інших фахівців, що працюють над проектом.

Логічна структура вебсайту - це внутрішній устрій вебсайту, його «кістяк», розташування сторінок, розділів, підрозділів, додаткових матеріалів. І першочерговим завданням розробників є створення стрункого порядку з хаотичного скупчення інформації.

На четвертому етапі виконується розробка макета дизайну вебсайту. Цей етап поділяється на кілька підетапів:

- генерація ідей дизайну, розробка макету у вигляді ескізів з текстовими поясненнями;
- розробка попереднього макета дизайну головної сторінки вебсайту;
- виправлення зауважень замовника (якщо є), доробка макета до завершеного вигляду;
- розробка внутрішніх сторінок за аналогічним алгоритмом.

У макеті повинні бути промальовані всі блоки, які будуть на сайті. Якщо до моменту розробки дизайну текстові та графічні матеріали ще не готові, можна використовувати довільний текст, але не можна залишати в макеті порожні місця.

HTML-CSS верстка. Залежно від цілей і завдань сайту, верстка повинна задовольняти наступним вимогам. Зазвичай, ці вимоги такі:

- кросбраузерність - сторінки повинні однаково відображатися в популярних браузерах (Google Chrome, Opera, Internet Explorer, Safari);
- адаптивність під різні розміри екрану та мобільні пристрої;
- гнучкість верстки - можливість легко додавати/видаляти блоки інформації на сторінки;
- швидкість обробки коду браузером;
- валідність – відповідність до різних стандартів;

– семантична коректність - логічне і правильне використання елементів HTML.

На шостому етапі відбувається програмування і встановлення системи керування контентом CMS (Content Management System), на якому реалізується весь функціонал сайту. Вимоги до цього етапу визначаються в технічному завданні.

На сьомому етапі відбувається заповнення сайту контентом (інформацією), всі матеріали сайту, будь то тексти чи графіка повинні вписуватися в загальну концепцію сайту, відповідати його цілям і задачам.

Тестування вебсайту і виправлення помилок є наступним етапом розробки, тестування працездатності вебсайту може проводити як розробник, так і замовник. Публікація вебсайту в Інтернеті потребує вибору доменної адреси, хостингової площадки, розгортання адміністративної частини або системи управління контентом, фізичного перенесення інформації.

Після остаточного завершення робіт над вебсайтом потрібно залучати на нього відвідувачів. Для просування та популяризації вебсайту можна скористатися контекстною або банерною рекламою, пошуковою оптимізацією, просуванням в соціальних мережах та іншими методами.

Спочатку визначаються цілі проєкту та вимоги до майбутнього вебсайту. На цьому етапі розробляється технічне завдання, у якому описуються функції сайту, структура сторінок, дизайн та інші параметри.

Після цього створюється бізнес-план і укладається договір між замовником і виконавцем. Наступним етапом є розробка дизайну та програмної частини вебсайту. У сучасній веброботі часто використовуються системи управління контентом CMS, які дозволяють легко створювати та редагувати вебсайти без глибоких знань програмування.

Серед безкоштовних CMS найпопулярнішими є WordPress, Joomla, Drupal, TYPO3, MODX, Opencart та Magento. Вони широко застосовуються для створення блогів, корпоративних вебсайтів та інтернет-магазинів. Існують також комерційні CMS, наприклад CS Cart, DataLife Engine та NetCat, які мають додаткові функції та технічну підтримку.

Для аналізу інформації в сучасних вебсистемах використовуються технології Data Mining, Text Mining та Web Mining. Вони дозволяють обробляти великі масиви даних і знаходити корисні закономірності.

Тема 6. Хостинг, домен та безпека вебсайту

Після створення вебсайту його необхідно розмістити на сервері, щоб користувачі могли отримати доступ через Інтернет. Для цього використовується хостинг – послуга з надання серверного простору для зберігання файлів сайту. Вебсайт також повинен мати доменне ім'я, яке дозволяє користувачам легко знаходити його в мережі.

Сучасні вебсистеми часто використовують хмарні сервіси та сховища даних. Вони забезпечують високу доступність, масштабованість та надійність роботи вебдодатків.

Безпека вебсайтів є важливою частиною веброзробки. Основні загрози можна поділити на загрози конфіденційності, цілісності та доступності інформації. Однією з найпоширеніших атак є ін'єкції, коли зловмисник вводить спеціально сформований код у запити до бази даних. Іншою поширеною атакою є міжсайтовий скриптинг (XSS), який дозволяє вставляти шкідливий код у вебсторінки.

Ще одним типом загроз є атаки відмови в обслуговуванні (DoS), метою яких є перевантаження сервера великою кількістю запитів. У випадку DDoS атаки запити надходять з великої кількості комп'ютерів одночасно, що значно ускладнює захист. Для протидії таким загрозам використовуються різні методи безпеки, включаючи фільтрацію трафіку, системи захисту та шифрування даних.

ДОДАТОК Б. ІНДИВІДУАЛЬНІ ЗАВДАННЯ

1. **Завдання 1.** Робота з форматуванням HTML-тексту. Для заданого варіанту тесту необхідно виконати форматування, використовуючи тільки HTML подання.

Номер варіанту	Контент Веб-сторінки
1	HTML (збільшити розмір) – скорочення від "HyperText Mark-up Language" - перекладається як "<u>Мова розмітка гіпертексту</u>" (виділення напівжирним, виділення терміну, підкреслення) <u>https://w3schoolsua.github.io/</u> (підсвітити посилання) <u>Гіпертекст - це текст, що не послідовно зв'язаний з іншими документами, дозволяючи з першої сторінки документу перейти на останню</u> (підкреслення, текст на розмір менше) HTML (напівжирний) впроваджує засоби для: створення структурованого документу (курсив) шляхом позначення структурного складу тексту: <u>заголовки, абзаци, списки, таблиці, цитати та інше</u> (підкреслення); отримання інформації (курсив) із <u>Всесвітньої мережі</u> (підкреслення, текст більш не актуальний) через гіперпосилання (напівжирний); створення інтерактивних (напівжирний) форм; (текст більш не актуальний) включення зображень, звуку, відео, та інших об'єктів до тексту (курсив, використати перенос кожного слова). «HTML – це мова гіпертекстової розмітки документів (напівжирний), яка працює через систему тегів і допомагає структурувати та компонувати елементи Веб-сторінок» (цитата)
2	Браузер (напівжирний) – це програмний застосунок, який дає можливість користувачеві взаємодіяти з текстом, <u>рисунками або іншою інформацією на гіпертекстовій Веб-сторінці</u> (виділення напівжирним, виділення терміну, підкреслення) Як працює браузер (напівжирний) Браузер виконуючи програму (HTML-текст) (напівжирний, використати перенос кожного слова) формує на екрані документ, який бачить користувач (курсив). Важливою властивістю виконання є: (текст більш не актуальний)

	послідовна побудова елементів на екрані зліва-направо (напівжирний) і згори-донизу (напівжирний) у відповідності з порядком проходження (<i>текст більш не актуальний</i>) тегів в <u>HTML-документі</u>. (<i>підсвітити</i>) <u>не можна написати програму так, щоб браузер спочатку намалював лінію в нижній частині екрану (підкреслення), а потім вивів текст у верхній.</u> <i>Браузер заповнює екран зліва-направо і згори-донизу. Він не може повернутися до тієї частини екрану, яку уже побудував.</i> <i>(підкреслення, текст на розмір менше, цитата)</i>
3	Форматування тексту (збільшити розмір) – засоби його зміни, такі як вибір стилю шрифту і використання ефектів, <u>що дозволяють змінювати вигляд тексту</u>. (<i>виділення напівжирним, виділення терміну, підкреслення</i>) Текст добре читається (<i>напівжирний</i>), якщо він <i>структурований, тобто, розбитий на окремі смислові частини</i> (курсив, використати перенос кожного слова).і в ньому застосовується <u>єдиний спосіб оформлення окремих елементів</u> (<i>надати підсвітлення</i>) по всьому документу. Структуру тексту (збільшити розмір) (напівжирний)краще всього <u>оформлювати</u> у наступному форматуванні: (<i>текст більш не актуальний</i>) Запишіть заголовок для всього документу (напівжирний, перенесення наступного рядка). Він повинен бути найбільшим (тег <H₁>) і вирівняний по центру в рядку. Розділіть документ на головні частини (<i>глави</i>). <i>Заголовки глав вирівнюються по лівому краю і робляться трохи менше заголовку документа (тег <H₂>).</i>
4	Програмування виведення тексту (збільшити розмір) (текст більш не актуальний) Перейдемо від загальних рекомендацій до питання про те, як їх можна виконати засобами HTML (напівжирний, перенесення кожного слова) Відкривають теги можуть містити <i>атрибути</i>, тобто додаткову інформацію про властивості відображуваного на екрані елемента (<i>напівжирний</i>): Атрибути записуються всередині відкриваючого <i>тегу після його імені</i> у вигляді окремого ключового слова, знаку “=” і <i>параметра (значення атрибута)</i> (курсив, напівжирний, підсвітлення)

	<u>Порядок слідування атрибутів у тегу не важливий.</u> (напівжирний, підкреслення, текст на розмір менше) Атрибут діє від відкриваючого тегу, в якому він заданий, до закриваючого або тільки всередині тегу, якщо тег не має парного. (виділення напівжирним, виділення терміну, підкреслення) «HTML – це мова гіпертекстової розмітки документів (напівжирний), яка працює <u>через систему тегів</u> () і <u>допомагає структурувати</u> (підкреслення) та компонувати елементи Веб-сторінок <u>підкреслення»</u> (цитата)
5	Форматування тексту (збільшити розмір) – <u>засоби його зміни</u>, такі як вибір стилю шрифту і використання ефектів, <u>що дозволяють змінювати вигляд тексту.</u> (виділення напівжирним, виділення терміну, підкреслення) Зміна розміру шрифту Виділяти елементи тексту можна не тільки за допомогою жирності, а й зміною розміру шрифту. (виділення напівжирним, курсив, підсвітлення тексту) Команда <BIG>...</BIG> збільшує розмір шрифту по відношенню до поточного, а команда <SMALL>...</SMALL> – зменшує його. (напівжирний, підкреслення, текст на розмір менше) Команди можуть бути вкладеними, <u>що дозволяє збільшувати</u> (або зменшувати) розмір у кілька разів. В HTML розмір шрифту вимірюється <u>в умовних одиницях від 1 до 7</u>, середній розмір тексту, <u>що використовується за замовчуванням</u>, прийнятий 3. (цитата, перенести кожне слово) «Додавання тегу <big> збільшує текст <u>на одну умовну одиницю</u>. Допускається застосування вкладених тегів <big>, при цьому розмір шрифту буде більше з кожним рівнем». (цитата)
6	Спеціальні символи HTML - це символи, які використовуються для розміщення на веб сторінках, <u>а також стилях</u>. виділення напівжирним, виділення терміну, підкреслення) Спеціальні символи (збільшити розмір) (текст більш не актуальний) Оскільки символи “<” і “>” (напівжирний) сприймаються браузером як початок і кінець тегу, <u>виникає питання: а як показати ці символи на екрані?</u> (підкреслення, курсив) Ще питання: чи можна вставити в тексті абзацу примусову групу пробілів (<u>адже браузер ігнорує зайві пропуски</u>)? (напівжирний, підкреслення, перенесення наступного рядка) <u>Для цих символів, а також для деяких інших передбачено спеціальне кодування</u> (курсив, підсвітлення тексту)

	«Код будується з символу “&” (амперсанд), фіксованого імені та завершального символу “; &im’я (напівжирний)» (цитата, перенести кожне слово)
7	Формат GIF (збільшити розмір) (Graphics Interchange Format) - формат для обміну зображеннями, це растровий <u>формат графічних зображень</u>. (виділення напівжирним, виділення терміну, підкреслення) GIF (напівжирний) здатний зберігати стислі дані без втрати якості не більше 256 кольорів. У більшості випадків цього цілком достатньо для отримання якісних ілюстрацій. (напівжирний, текст більш не актуальний) GIF-формат має <u>три приємні додаткові можливості</u>: (підкреслення, підсвітлення тексту) Мультиплікація. Використовуючи <u>анімаційний GIF-редактор</u>, легко зібрати простий мультик: намалювати окремі кадри, налаштувати час затримки в <u>послідовності показу</u>, “зациклити кіно” або показувати <u>кадри тільки один раз</u>. (підкреслення, перенесення наступного рядка, текст на розмір менше) Анімація. Коли анімація готова, вона записується на диск як звичайний GIF-файл. Хоча в цьому файлі не одна, а ціла серія GIF-картинок і різна настроювальна інформація. (тематичний розділ тексту, текст на розмір менше)
8	Рисунок в тексті (збільшити розмір) – розміщення <u>картинки в тексті в HTML</u> виконується за допомогою тегу <code></code>. (виділення напівжирним, виділення терміну, підкреслення) Вставлення зображення на веб-сторінку відбувається через <u>тег <code></code> HTML-файлу і розміщується всередині тега <code><body></code></u>. (напівжирний, текст більш не актуальний) Елемент <u><code></code> є порожнім тегом, йому не потрібно закриває тег</u>, оскільки він не містить жодного додаткового вмісту, крім його атрибутів. (підкреслення, підсвітлення тексту) Команда не має парного закриваючого тегу, але має багато атрибутів. (перенесення наступного рядка) Атрибут <code>src</code> = і’мя файла Найголовнішим атрибутом команди є атрибут <code>src</code>, за допомогою якого можна задати ім’я файлу з рисунком. (перенесення кожного слова) (тематичний розділ тексту, текст на розмір менше)

	«Наприклад, команда змусить браузер відобразити на екрані графічний файл img1.gif з поточного каталогу» (цитата).
9	Рисунок, як гіперпосилання (збільшити розмір) Посилання в HTML (Hypertext Markup Language) <u>відіграють важливу роль у створенні мережі взаємопов'язаного контенту</u>, що дозволяє користувачам безперешкодно переміщатися (напівжирний) між різними сторінками та Веб-сайтами. (виділення напівжирним, виділення терміну, підкреслення) Для формування гіперпосилання (напівжирний) досить вкласти команду IMG всередину команди A: (напівжирний, текст більш не актуальний) Рисунок поводитьься так само, як текст. (курсив, підсвітлення тексту) Курсор при попаданні на картинку-посилання міняє свою форму. (курсив, перенос кожного слова) <i>Відмінність тільки в тому, що рисунок-посилання не підкреслюється, як текст, і не виділяється кольором, а обрамляється в рамочку.</i> (підкреслення, перенесення наступного рядка, текст на розмір менше)
10	HTML (збільшити розмір) – <i>це мова гіпертекстової розмітки документів</i> (напівжирний), яка працює <u>через систему тегів</u> (підкреслення) і <u>допомагає структурувати</u> (підкреслення) та компонувати елементи Веб-сторінок. (виділення напівжирним, виділення терміну, підкреслення) <u>Гіпертекст</u> - це текст, що не послідовно зв'язаний з іншими документами, дозволяючи з першої сторінки документу перейти на останню (підкреслення, текст на розмір менше) текст більш не актуальний) Більшість Веб-сторінок створюються за допомогою мови HTML (підсвітлення тексту) HTML-документ оброблюється браузером та відтворюється на екрані у звичному для людини вигляді. (напівжирний) HTML разом із каскадними таблицями стилів та вбудованими скриптами — це три основні технології побудови веб-сторінок. (тематичний розділ тексту, текст на розмір менше)

2. Завдання 2. HTML-зображення

1. Оберіть **готове зображення** (наприклад, пейзаж або фотографію тварини) та виконайте вставку зображення на HTML-сторінку. Дайте назву сторінці за обраним зображенням. Виконайте наступні вимоги:

Використовуйте тег **** для вставляння рисунку

Використайте атрибути **<alt>**, **width** і **height** для рисунку

Виконати центрування рисунку на сторінці з використанням атрибута **align**

3. Створіть слайд-шоу зображень

Створіть слайд-шоу за допомогою **5** тематично підібраних рисунків (фото), де кожне зображення відображається в новому рядку. Дайте назву кожному зображенню та сторінці.

Поставте кілька рисунків на сторінку за допомогою тегу ****

Використайте атрибути **<alt>**, **width** і **height** для рисунку

Будь-яке зображення повинно знаходитися в новому рядку (використовуйте тег **
** для переносу рисунку на нову сторінку).

4. Створіть інтерактивну карту на зображенні

Створіть інтерактивну карту на зображенні, використовуючи теги **<map>** і **<area>**. Наприклад, на зображенні карти України можна виділити різні області.

Вставте зображення карти .JPG за допомогою тегу ****

Додайте карту зображення за допомогою тега **<map>**

Для кожної області зображення створіть інтерактивні зони з тегом **<area>**

5. Вставка зображення із заголовком

Вставте зображення (наприклад, пейзаж або фотографію тварини) на HTML-сторінку та додайте під ним заголовок або підпис.

Вставте зображення з допомогою тегу ****

Використайте атрибути **<alt>**, **width** і **height** для зображення

Під зображенням додайте текст за допомогою заголовків **<h2>** або **<p>**.

Заголовок повинен описувати зображення (наприклад, "Море").

6. Фон сторінки із зображенням

Встановіть фонове зображення для всієї HTML-сторінки. Оберіть зображення на безкоштовному сайті зображень.

Використовуйте тег **<body>** з атрибутом **background** для задання фонового зображення.

Зображення має бути у форматі .JPG.

Переконайтеся, що зображення правильно відображається на всій HTML-сторінці.

7. Створення альбому зображень

Створіть веб-сторінку, яка являє собою фотоальбом, що складається з кількох сторінок, будь-яка з яких містить одне зображення та опис.

На кожній сторінці повинно бути одне зображення з тегом ****

Під кожним зображенням повинен бути текст з описом зображення.

Додайте навігацію між сторінками альбому за допомогою тега **<a>**, щоб користувач міг переходити з однієї сторінки на іншу.

Завдання 3. Розробка HTML-таблиці

1. Створити HTML-таблицю згідно варіанту - друга цифра (XY) номера за списком в журналі.

1	 <table border="1"> <thead> <tr> <th colspan="2">Застосування хмарних технологій</th></tr> <tr> <th>Етап уроку (заняття)</th><th>Додаток що використовується</th></tr> </thead> <tbody> <tr> <td>Перевірка домашнього завдання</td><td>Google Презентація, Google Документ, Google Таблиці, Google Форми, он-лайн дошка.</td></tr> <tr> <td>Перевірка знань і умінь учнів для підготовки до нової теми</td><td>Google Рисунок (інфографіка, схеми, графіки)</td></tr> <tr> <td>Первинна перевірка розуміння</td><td>Google Рисунок, Google Документ</td></tr> <tr> <td>Організація засвоєння способів діяльності шляхом відтворення інформації в її застосуванні за зразком</td><td>Google Рисунок, Google Документ, Google Таблиці</td></tr> <tr> <td>Творче застосування та добування знань, освоєння способів діяльності шляхом вирішення проблемних завдань, побудованих на основі засвоєних знань і умінь.</td><td>Google Презентація, Google Документ</td></tr> <tr> <td>Домашнє завдання</td><td>Google Таблиці, Google Форми, он-лайн дошка.</td></tr> </tbody> </table>	Застосування хмарних технологій		Етап уроку (заняття)	Додаток що використовується	Перевірка домашнього завдання	Google Презентація, Google Документ, Google Таблиці, Google Форми, он-лайн дошка.	Перевірка знань і умінь учнів для підготовки до нової теми	Google Рисунок (інфографіка, схеми, графіки)	Первинна перевірка розуміння	Google Рисунок, Google Документ	Організація засвоєння способів діяльності шляхом відтворення інформації в її застосуванні за зразком	Google Рисунок, Google Документ, Google Таблиці	Творче застосування та добування знань, освоєння способів діяльності шляхом вирішення проблемних завдань, побудованих на основі засвоєних знань і умінь.	Google Презентація, Google Документ	Домашнє завдання	Google Таблиці, Google Форми, он-лайн дошка.
Застосування хмарних технологій																	
Етап уроку (заняття)	Додаток що використовується																
Перевірка домашнього завдання	Google Презентація, Google Документ, Google Таблиці, Google Форми, он-лайн дошка.																
Перевірка знань і умінь учнів для підготовки до нової теми	Google Рисунок (інфографіка, схеми, графіки)																
Первинна перевірка розуміння	Google Рисунок, Google Документ																
Організація засвоєння способів діяльності шляхом відтворення інформації в її застосуванні за зразком	Google Рисунок, Google Документ, Google Таблиці																
Творче застосування та добування знань, освоєння способів діяльності шляхом вирішення проблемних завдань, побудованих на основі засвоєних знань і умінь.	Google Презентація, Google Документ																
Домашнє завдання	Google Таблиці, Google Форми, он-лайн дошка.																

2

PHY	Bandwidth (as number of data subcarriers)	Data bits per subcarrier	Time per OFDM symbol (800ns GI)	1 SS	3 SS	4 SS	8 SS
802.11ac	234 (80 MHz)	$5/6 \times \log_2(256)$ ≈ 6.67	4 μ s	390 Mbps	1.17 Gbps	1.56 Gbps	-
	2 x 234 (160 MHz)	X	/	= 780 Mbps	-	3.12 Gbps	-
802.11ax	980 (80 MHz)	$5/6 \times \log_2(1024)$ ≈ 8.33	13.6 μ s	600 Mbps	1.8 Gbps	2.4 Gbps	4.8 Gbps
	2 x 980 (160 MHz)			1.2 Gbps	3.6 Gbps	4.8 Gbps	-

3

Standard name	Branded name	Year Introduced	Supported Bands
802.11a	Wi-Fi 2*	1999	5 GHz
802.11b	Wi-Fi 1*	1999	2.4 GHz
802.11g	Wi-Fi 3*	2003	2.4 GHz
802.11n	Wi-Fi 4	2008	2.4 and 5 GHz
802.11ac	Wi-Fi 5	2013	5 GHz
802.11ax	Wi-Fi 6, Wi-Fi 6E	2020	2.4, 5 and 6 GHz

3

 MEREHEAD	Web 1.0	Web 2.0	Web 3.0
Роки	3 1991 по 2004 рік	3 2004 року по теперішній час	У найближчому майбутньому
Можливості	Пасивне поглинання контенту на відстані	Поглинання і створення контенту	Поглинання, створення та володіння контентом
Технології	HTML, ASP, PHP, JSP	Dynamic HTML, Javascript, CSS, Python, Node.js, MySQL та інші	Web 2.0 + блокчейн, Інтернет речей, штучний інтелект, VR
Віртуальне середовище	Ні	Базове 3D	3D, VR, AR
Реклама	Статична (банери тощо)	Інтерактивна, контекстна	Інтерактивна, вузькотаргетована, контекстна
Власник даних	Творець сайту	Технологічні гіганти	Творці контенту
Зберігання даних	PC, приватні невеликі сервери	Централізовані сховища	Розподілені мережі

4

Action	Дія	WU
Loading page HTML	Завантаження HTML сторінки	0.15
Each call to a server-side plugin action	Кожен виклик дії плагіна на стороні сервера	0.2
Each millisecond spend executing a server-side plugin action	Кожна мілісекунда виконує дію плагіна на стороні сервера	0.0005
1 sec	Кожна секунда виконує дію плагіна на стороні сервера	0.5
Performing a database search	Виконання пошуку в базі даних	0.3
Each character of data returned from the database	Кожен символ даних, що повертається з бази даних	0.000003
Each result returned from the database	Кожен результат повертається з бази даних	0.015
Each data trigger workflow initiated	Ініціюється робочий цикл кожного тригера даних	0.05
Each call to watch a search for real-time updates	Кожен дзвінок для перегляду пошуку оновлень у реальному часі	0.005
Each item deleted from the database	Кожен елемент видалено з бази даних	0.1

5	Company	Q1	Q2	Q3	Q4
	Microsoft	20.3	30.5	23.5	40.3
	Google	50.2	40.63	45.23	39.3
	Apple	25.4	30.2	33.3	36.7
	IBM	20.4	15.6	22.3	29.3

Завдання 4. Використання CSS

1. Виконати стилізацію заголовків

Створіть HTML-сторінку з трьома різними заголовками (h1, h2, h3). Використовуючи CSS, змініть такі властивості кожного заголовка:
 Шрифт (наприклад, Arial, Georgia)
 Розмір шрифту (наприклад, h1 – 40px, h2 – 30px, h3 – 20px)
 Колір тексту для кожного заголовка різний (наприклад, червоний, зелений, синій).

2. Оформлення абзацу

Створіть абзац тексту (тег <p>) з будь-яким текстом.
 Використовуючи CSS, налаштуйте такі властивості:
 Вирівнювання тексту (ліворуч, по центру, праворуч, за шириною).
 Відступи всередині абзацу (поля padding) – по 20px з усіх сторін.
 Встановіть інтерліньяж (line-height) - 1.5.
 Додайте тінь до тексту (text-shadow) будь-якого кольору.
 Зробіть межу обраного кольору.

3. Створення декоративного тексту

Створіть блок з текстом (тег <div>).
 Застосуйте наступні стилі:
 Встановіть розміри marding, padding, border для загальної ширини 400px, поля, межі та відступи зробити різним кольором

Додайте будь-який текст

Зробіть текст жирним за допомогою властивості font-weight.

Перетворіть текст на всі великі літери (text-transform: uppercase).

Додайте підкреслення до тексту (text-decoration: underline).

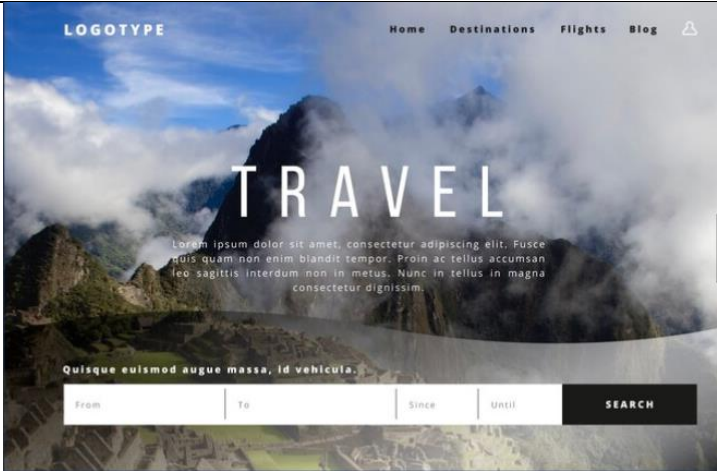
Налаштуйте відстань між літерами (letter-spacing) на 2px.

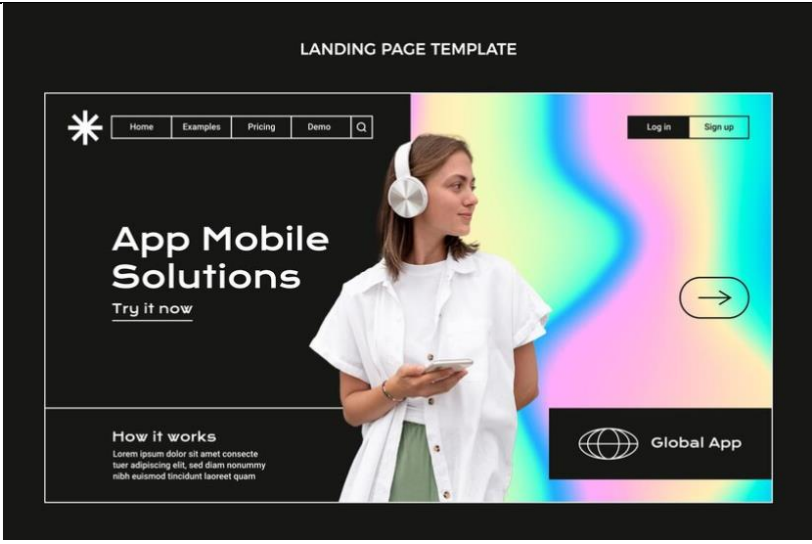
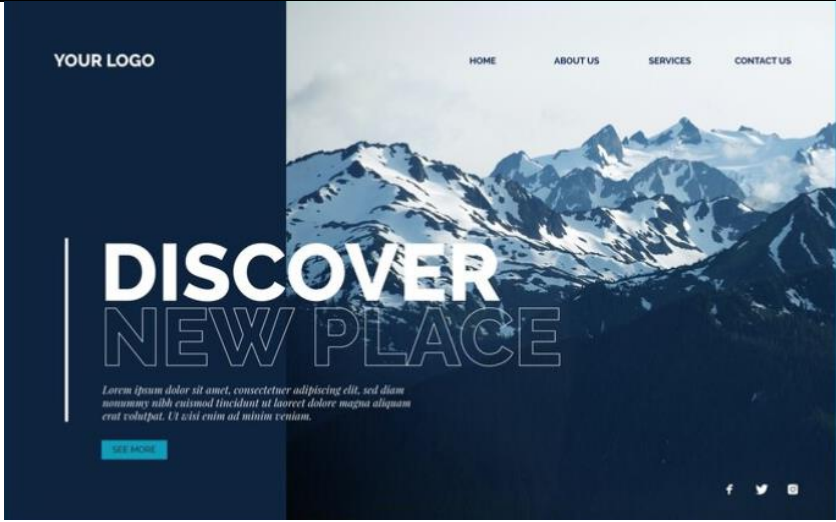
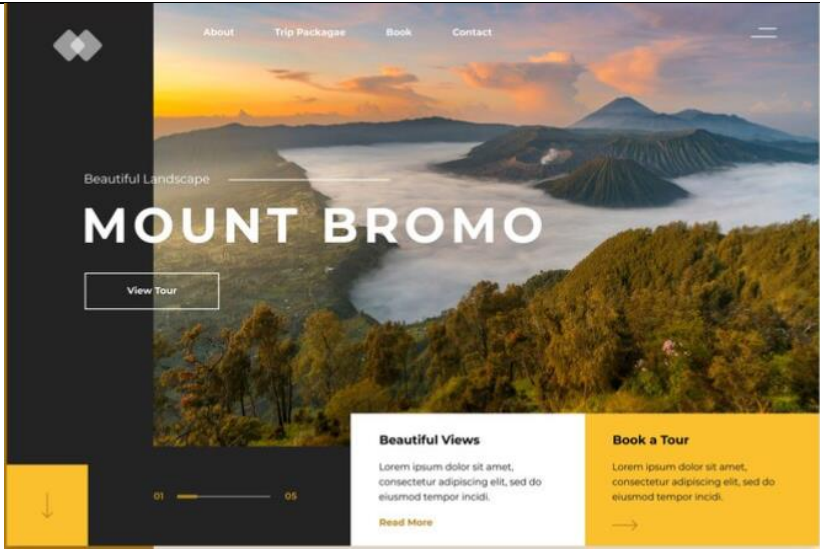
Завдання 4. Розробка фрагменту Веб-сайту

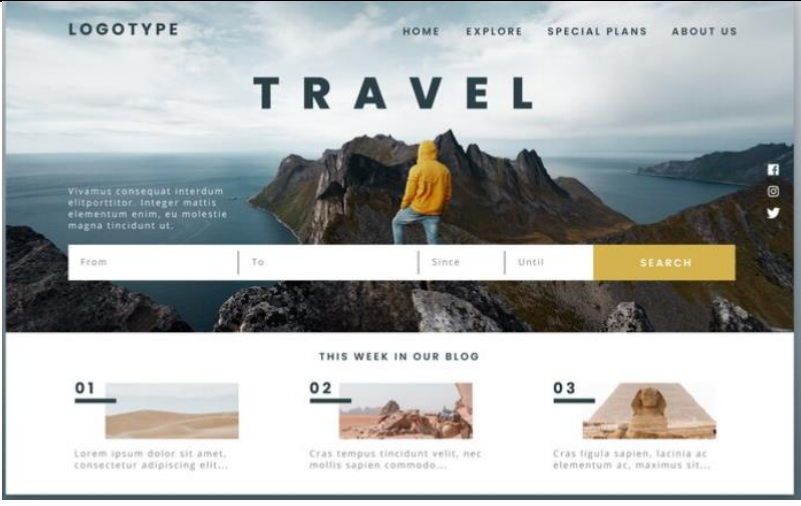
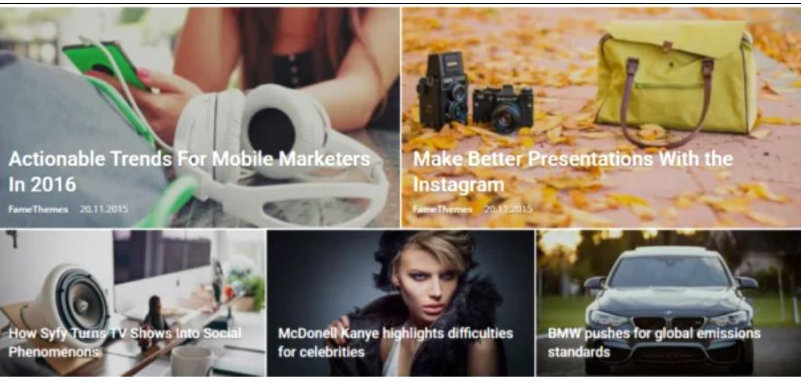
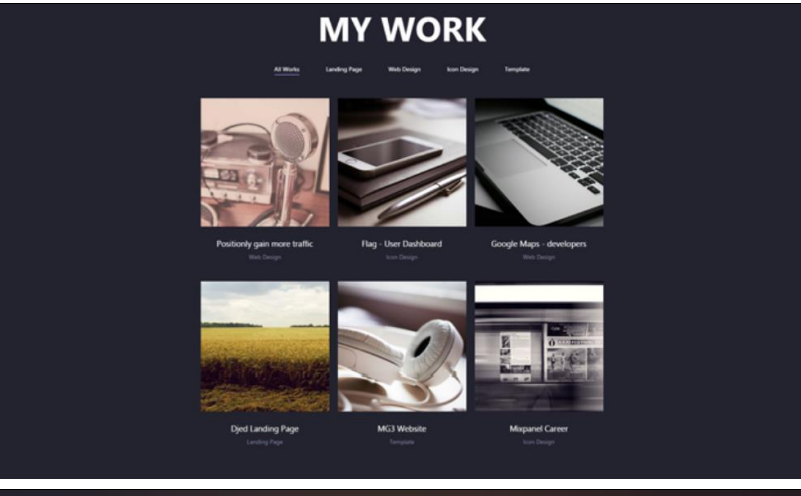
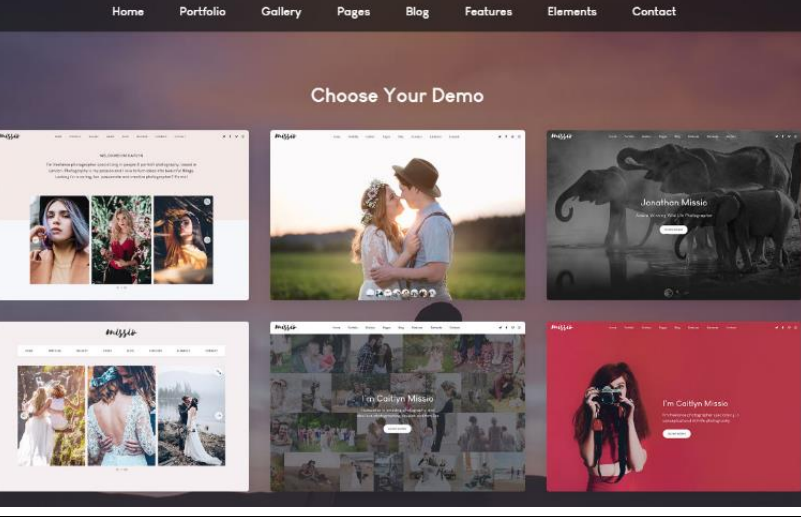
Розробити фрагмент Веб-сайту за заданим зображенням, на сайті передбачити:

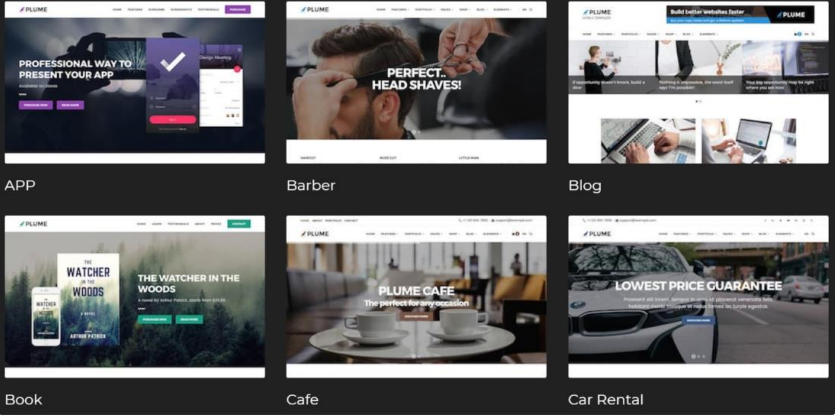
1. Назву сайту та повний перелік метаданих.
2. Підібрати фото або рисунки (за допомогою ресурсів www.freepik.com, unsplash.com/, www.pexels.com).
3. Перевірити фото або рисунки за допомогою ресурсу <https://www.metadata2go.com/view-metadata>.
4. Повністю повторити зовнішній вигляд пропонованого Вебсайту.
5. Використати посилання.

Варіанти завдань

№ варіанту	Зовнішній вигляд фрагменту сайту
1	

2	 A landing page template for mobile app solutions. The header includes a navigation menu with 'Home', 'Examples', 'Pricing', and 'Demo', along with a search icon and 'Log in'/'Sign up' buttons. The main section features a woman wearing headphones and holding a smartphone, with the text 'App Mobile Solutions' and 'Try it now'. A 'How it works' section follows with placeholder text. The footer includes a 'Global App' logo.
3	 A landing page template for discovering new places. The header includes a 'YOUR LOGO' placeholder and a navigation menu with 'HOME', 'ABOUT US', 'SERVICES', and 'CONTACT US'. The main section features a large image of snow-capped mountains with the text 'DISCOVER NEW PLACE' and a 'SEE MORE' button. The footer includes social media icons for Facebook, Twitter, and Instagram.
4	 A landing page template for Mount Bromo. The header includes a navigation menu with 'About', 'Trip Packages', 'Book', and 'Contact', along with a hamburger menu icon. The main section features a large image of Mount Bromo at sunset with the text 'Beautiful Landscape' and 'MOUNT BROMO', and a 'View Tour' button. The footer includes a 'Beautiful Views' section with placeholder text and a 'Read More' button, and a 'Book a Tour' section with placeholder text and a 'Book a Tour' button.

5	 A travel website template featuring a hero section with a large image of a person in a yellow jacket standing on a rocky cliff overlooking the ocean. The word "TRAVEL" is prominently displayed in the center. Below the hero section is a search bar with fields for "From", "To", "Since", and "Until", followed by a "SEARCH" button. The bottom section is titled "THIS WEEK IN OUR BLOG" and contains three blog entries, each with a small image and a title.
6	 A blog website template with a grid of six article cards. Each card features a large image, a title, and a date. The articles are: "Actionable Trends For Mobile Marketers In 2016", "Make Better Presentations With the Instagram", "How Syfy Turns TV Shows Into Social Phenomenons", "McDonell Kanye highlights difficulties for celebrities", "BMW pushes for global emissions standards", and "Google Maps - developers".
7	 A portfolio website template titled "MY WORK". It features a grid of six project cards, each with a large image and a title. The projects are: "Positionally gain more traffic", "Flag - User Dashboard", "Google Maps - developers", "Dynd Landing Page", "MCI Website", and "Mapanel Career".
8	 A photography website template titled "Choose Your Demo". It features a grid of six demo images, each with a title and a "View" button. The demos are: "Jonathan Missio", "I'm Caitlyn Missio", "Caitlyn Missio", "Jonathan Missio", "Jonathan Missio", and "Jonathan Missio".

9	 The image displays six website mockups for a brand named 'PLUME'. The mockups are arranged in a 2x3 grid. The top row includes 'APP' (showing a mobile app interface), 'Barber' (featuring a man's head being shaved), and 'Blog' (showing a person working on a laptop). The bottom row includes 'Book' (showing a book cover for 'THE WATCHER IN THE WOODS'), 'Cafe' (showing coffee cups on a table), and 'Car Rental' (showing a car with the text 'LOWEST PRICE GUARANTEE'). Each mockup has a dark header with the 'PLUME' logo and navigation links.
10	 The image shows a website mockup for 'Your Logo'. The header includes navigation links: HOME, PARTY, ADDRESS, GUESTS, ABOUT, and a search bar. The main content area features a large image of a woman with glasses, overlaid with a green grid pattern. To the right of the image, the text reads 'START THE PARTY! UNLEASH YOUR INNER WILD' in large, bold, black letters. Below this text is a 'JOIN NOW' button. At the bottom right, there are social media icons for WhatsApp, Instagram, Twitter, and Facebook.

ЗМІСТ

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ.....	3
2. ОЧІКУВАНІ КОМПЕТЕНТНОСТІ, ЯКІ ПЛАНУЄТЬСЯ СФОРМУВАТИ, ТА ДОСЯГНЕННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ	4
3. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ.....	7
4. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ.....	8
5. ПИТАННЯ ДО ПРАКТИЧНИХ ЗАНЯТЬ	11
6. САМОСТІЙНА РОБОТА.....	12
7. ВИДИ ТА МЕТОДИ КОНТРОЛЮ	14
8. ОЦІНЮВАННЯ ПОТОЧНОЇ, САМОСТІЙНОЇ ТА ІНДИВІДУАЛЬНОЇ РОБОТИ ЗДОБУВАЧІВ	17
9. ОЦІНЮВАННЯ ЗАГАЛЬНИХ РЕЗУЛЬТАТІВ ЗНАНЬ ЗДОБУВАЧІВ ..	18
10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА.....	20
ДОДАТОК А. КОНСПЕКТ ЛЕКЦІЙ.....	23
ДОДАТОК Б. ІНДИВІДУАЛЬНІ ЗАВДАННЯ.....	31
ЗМІСТ.....	44

Соловська Ірина Миколаївна

Русу Олександр Петрович

ВЕБТЕХНОЛОГІЇ ТА ВЕБДИЗАЙН

Методичні рекомендації до практичних занять

та самостійної роботи здобувачів

галузі знань – 12 Інформаційні технології, спеціальності 121 «Інженерія програмного забезпечення», 122 «Комп'ютерні науки», 123 «Комп'ютерна інженерія», 125 «Кібербезпека та захист інформації»,

галузі знань 17 – Електроніка, автоматизація та електронні комунікації, спеціальність 172 «Електронні комунікації та радіотехніка»,

галузі знань 01 – Освіта/Педагогіка, спеціальність 014 «Середня освіта. Інформатика»

Українською мовою