



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Вебтехнології

Назва


Галузь знань 12 – Інформаційні технології
Спеціальність 123 – Комп'ютерна інженерія очна денна форма здобуття освіти
Освітня програма Комп'ютерна інженерія та програмування
Статус дисципліни: обов'язкова, дисципліна професійної підготовки
Факультет – інформаційних технологій
Кафедра – Комп'ютерної інженерії та інформаційних систем

Форма здобуття освіти	Курс	Семестр	Загальне навантаження		Кількість годин						Курсовий проєкт	Курсова робота	Залік	Іспит
			Кредити ЄКТС	Години	Аудиторні заняття				Індивідуальна робота студента	Самостійна робота, в т.ч. ІРС				
					Разом	Лекції	Лабораторні роботи	Практичні заняття						
ОД	1	2	5	150	72	18	36	18		78				+
Разом			5	150	72	18	36	18		78				1

Робоча програма складена на основі стандарту вищої освіти зі спеціальності 123 Комп'ютерна інженерія, освітньо-професійної програми та навчального плану.

Програма складена  Ольга ПАВЛОВА
Підпис Ініціали, прізвище викладача(ів)

Схвалена на засіданні кафедри комп'ютерної інженерії та інформаційних систем.
 Протокол № 1 від 12 серпня 2022 р.

Зав. кафедри комп'ютерної інженерії та інформаційних систем  Тетяна ГОВОРУЩЕНКО
Підпис Ініціали, прізвище

Робоча програма розглянута та схвалена Вченою радою факультету інформаційних технологій

Голова Вченої ради  Олег САВЕНКО
Підпис Ініціали, прізвище

ВСТУП

Мета викладання дисципліни. Дисципліна "Веб-технології" є однією зі спеціальних профільюючих дисциплін і тому займає провідне місце у підготовці бакалаврів комп'ютерної інженерії.

Метою дисципліни " Веб-технології " є: 1) формування здатності розв'язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності в комп'ютерній галузі або навчання, що передбачає застосування теорій та методів комп'ютерної інженерії і характеризується комплексністю та невизначеністю умов; 2) вироблення вмінь аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати та захищати прийняті рішення; 3) формування здатності забезпечувати проектування та розроблення якісних програмних і технічних засобів комп'ютерних систем та мереж; 4) набуття навичок розробки бізнес-рішень та оцінювання нових технологічних пропозицій.

Предмет дисципліни. Методи та основи розробки статичних та динамічних веб-сайтів; механізми та принципи роботи веб-додатків.

Завдання дисципліни. Надати студентам знання із розуміння механізмів роботи веб-додатків, а також умінь та практичні навички для розробки власних веб-сервісів.

Після вивчення дисципліни "**Вебтехнології**" студент має досягти таких результатів навчання (сукупність знань, умінь, навичок, компетентностей).

знати:

- об'єкт, предмет, задачі, проблематику дисципліни та її основні розділи;
- основи веб-розробки та управління веб-проектами;
- інновації у галузі веб-розробки

уміти:

- використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення; розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем мовою PHP;
- використовувати навички програмування, технології розроблення алгоритмів і комп'ютерних програм, методи створення інтернет-ресурсів мовами високого рівня для розв'язання задач проектування і використання програмно-технічних засобів комп'ютерних систем та мереж;
- ефективно працювати як індивідуально, так і у складі команди;
- системно мислити та застосовувати творчі здібності до формування нових ідей, удосконалювати креативне мислення.

бути здатним:

- застосовувати законодавчу та нормативно-правову базу, а також державні та міжнародні вимоги, практики і стандарти з метою здійснення професійної діяльності в галузі комп'ютерної інженерії;
- використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення;
- здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів;
- аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати та захищати прийняті рішення;
- розробляти бізнес-рішення та оцінювати нові технологічні пропозиції.

Компетентності, на формування яких спрямовано ОК:

Інтегральна – Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності в комп'ютерній галузі або навчання, що передбачає застосування теорій та методів комп'ютерної інженерії і характеризується комплексністю та невизначеністю умов;

ЗК2. Здатність вчитися і оволодівати сучасними знаннями

ЗК3. Здатність застосовувати знання у практичних ситуаціях

ЗК4. Здатність спілкуватися державною мовою як усно, так і письмово

ЗК6. Навички міжособистісної взаємодії

ЗК7. Вміння виявляти, ставити та вирішувати проблеми

ЗК8. Здатність працювати в команді

ЗК11. Здатність до розуміння предметної галузі та професійної діяльності

ЗК12. Здатність використовувати інформаційні та комунікаційні технології

ЗК13. Здатність розв'язувати поставлені задачі та приймати відповідні рішення

ЗК14. Здатність розробляти та управляти проектами, оцінювати та забезпечувати якість виконуваних робіт

ФК1. Здатність застосовувати законодавчу та нормативно-правову базу, а також державні та міжнародні вимоги, практики і стандарти з метою здійснення професійної діяльності в галузі комп'ютерної інженерії.

ФК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

ФК3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.

ФК11. Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів.

ФК15. Здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати та захищати прийняті рішення. Спеціальні компетентності, визначені за освітньою програмою

ФК17. Здатність забезпечувати проектування та розроблення якісних програмних і технічних засобів комп'ютерних систем та мереж

ФК18. Здатність розробляти бізнес-рішення та оцінювати нові технологічні пропозиції

Програмні результати навчання, на забезпечення яких спрямовано ОК:

ПРН3. Знати новітні технології в галузі комп'ютерної інженерії.

ПРН5. Мати знання основ економіки та управління проектами.

ПРН6. Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.

ПРН7. Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності.

ПРН8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.

ПРН10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати типове для спеціальності обладнання.

ПРН11. Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії.

ПРН12. Вміти ефективно працювати як індивідуально, так і у складі команди.

ПРН15. Вміти виконувати експериментальні дослідження за професійною тематикою.

ПРН16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.

ПРН17. Спілкуватись усно та письмово з професійних питань українською мовою та однією з іноземних мов (англійською, німецькою, італійською, французькою, іспанською).

ПРН18. Використовувати інформаційні технології для ефективного спілкування на професійному та соціальному рівнях.

ПРН19. Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.

ПРН20. Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.

ПРН21. Якісно виконувати роботу та досягати поставленої мети з дотриманням вимог професійної етики.

ПРН23. Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання програмно-технічних засобів комп'ютерних систем та мереж.

ВЕБТЕХНОЛОГІЇ

Тип дисципліни	Обов'язкова
Рівень вищої освіти	Перший (бакалаврський)
Мова викладання	Українська
Семестр	2
Кількість встановлених кредитів ЄКТС	5
Форми здобуття освіти	Очна денна

Результати навчання. Студент, який успішно завершив вивчення дисципліни, повинен: знати новітні технології в галузі комп'ютерної інженерії; мати знання основ економіки та управління проєктами; вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей; вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності; вміти системно мислити та застосовувати творчі здібності до формування нових ідей; вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати типове для спеціальності обладнання; вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії; вміти ефективно працювати як індивідуально, так і у складі команди; вміти виконувати експериментальні дослідження за професійною тематикою; вміти оцінювати отримані результати та аргументовано захищати прийняті рішення; спілкуватись усно та письмово з професійних питань українською мовою та однією з іноземних мов (англійською, німецькою, італійською, французькою, іспанською); використовувати інформаційні технології для ефективного спілкування на професійному та соціальному рівнях; адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення; усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення; якісно виконувати роботу та досягати поставленої мети з дотриманням вимог професійної етики.

Зміст навчальної дисципліни. Основні положення та означення веб-програмування. Поняття клієнт-серверної архітектури: взаємодія веб-сервера та браузера. Мова гіпертекстової розмітки. Мови сценаріїв (серверні та скриптові мови). Протокол HTTP. Система контролю версій GitHub. Поняття крос-браузерної верстки із застосуванням технології Bootstrap. Основи роботи з БД MySQL. Основи ООП: класи та об'єкти. Поняття MVC-архітектури. Фреймворки Laravel та Yii. Поняття системи керування контентом. Розгляд найпоширеніших шаблонів програмування.

Запланована навчальна діяльність: лекції – 18 год., лабораторні заняття – 36 год., практичні заняття – 18 год., самостійна робота – 78 год.; разом – 150 год.

Методи навчання: словесні, наочні, проблемно-пошукові (лекції); пояснювально-ілюстративні, практичні, частково-пошукові (практичні та лабораторні заняття), дослідницькі, частково-пошукові (самостійна робота: індивідуальні завдання).

Форми оцінювання результатів навчання: захист лабораторних та практичних робіт, тестовий контроль, підсумковий контрольний захід.

Форма семестрового контролю: іспит

Навчальні ресурси:

1. Павлова О.О., Денисюк Д.О., Гнатчук Є.Г., Кисіль Т.М. Методичні вказівки до виконання лабораторних робіт з навчальної дисципліни «Веб-технології» – Хмельницький: ХНУ, 2020. - 99 с.
2. Steve Prettyman, Learn PHP 8 using MySQL, JavaScript, CSS3 and HTML5. Apress Media, USA, 2021. – 442 р.
3. Модульне середовище для навчання MOODLE. Доступ до ресурсу: <https://msn.khmnu.edu.ua/>
4. Електронна бібліотека університету. Доступ до ресурсу: <http://library.khmnu.edu.ua/>

Викладач: доктор філософії, старший викладач Павлова О. О.

1. СТРУКТУРА ЗАЛІКОВИХ КРЕДИТІВ ДИСЦИПЛІНИ

Назва теми	Кількість годин, відведених на:			
	Лекції	Практичні роботи	Лабораторні роботи	СРС
<i>Другий семестр</i>				
Тема 1. Поняття клієнт-серверної архітектури. Мови сценаріїв (серверні та скриптові мови).	2	2	4	8
Тема 2. Мова розмітки тексту веб-сторінок HTML	2	2	4	8
Тема 3. Мова стилізації веб-сторінок. Каскадні таблиці стилів CSS	2	2	6	8
Тема 4. Мова сценаріїв веб-сторінок JavaScript +jQuery	2	2	4	11
Тема 5. Адаптивний дизайн веб-сторінок. Технологія Bootstrap	2	2	4	8
Тема 6. Мова програмування PHP	2	2	4	8
Тема 7. Робота з формами в PHP. Суперглобальні змінні та функції для роботи з файлами та масивами	2	2	4	8
Тема 8. Робота з базою даних MySQL в PHP	2	2	4	11
Тема 9. Застосування отриманих знань для роботи над веб-проектом	2	2	2	8
Разом:	18	18	36	78

2. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

2.1. Зміст лекційного курсу

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
	<i>Другий семестр</i>	
1	Поняття клієнт-серверної архітектури. Мови сценаріїв (серверні та скриптові мови). Вступ. Поняття «клієнт-сервер». Поняття локального сервера. Поняття серверних та скриптових мов програмування. Літ.: [1, 2, 3]	2
2	Мова розмітки тексту веб-сторінок HTML Поняття шаблону розмітки веб-сторінки. Види HTML-тегів. Рядкові та блокові елементи. Літ.: [1-3,6,7]	2
3	Мова стилізації веб-сторінок. Каскадні таблиці стилів CSS Елементи та класи в CSS. Стилзація Веб-сторінки за допомогою елементів CSS. Блокова-модель CSS. Літ.: [3, 12-17]	2
4	Мова сценаріїв веб-сторінок JavaScript +jQuery Знайомство з синтаксисом, основними функціями та подіями в JavaScript. Основи програмування сценаріїв веб-сторінки за допомогою мови JavaScript Літ.: [1-3, 6, 7]	2
5	Адаптивний дизайн веб-сторінок. Технологія Bootstrap Поняття адаптивного дизайну та розмірної сітки Bootstrap. Стилі готових компонентів веб-сайту. Літ.: [1-3]	2
6	Мова програмування PHP Змінні та синтаксис PHP. Функції користувача та вбудовані функції. Літ.: [1-3,6,7]	2
7	Робота з формами в PHP. Суперглобальні змінні та функції для роботи з файлами та масивами Створення форм користувача. Функції для роботи з файлами та масивами та їх застосування у веб-сайтах. Літ.: [1-7]	2
8	Робота з базою даних MySQL в PHP Функції для роботи з базою даних. Робота з PHP MyAdmin Літ.: [1-5]	2
9	Підсумкове заняття	2
	Разом:	18

2.2 Зміст лабораторних занять

№ з/п	Тема лабораторного заняття	Кількість годин
<i>Другий семестр</i>		
1	Встановлення та налаштування локального сервера (APACHE +MySQL) та середовища розробки PHP Storm Літ.: [1, 2, 3]	4
2	Знайомство зі структурою HTML-документу. Header веб-сторінки Літ.: [1-3,6,7]	4
3	Знайомство зі структурою HTML-документу. Header веб-сторінки Літ.: [3, 12-17]	
4	Основна частина веб-сторінки. Активне застосування HTML+CSS Літ.: [1-3]	4
5	Основна частина веб-сторінки. Активне застосування HTML+CSS Літ.: [18-20]	
6	Футер веб-сторінки. Створення форми підписки Літ.: [1-3, 6,7]	4
7	Футер веб-сторінки. Створення форми підписки Літ.: [1-7]	4
8	Створення випадаючого меню та форми авторизації. Прив'язка додаткових сторінок до веб-сайту Літ.: [1-7]	4
9	Підсумкове заняття	4
	Разом	36

2.3 Зміст практичних занять

№ з/п	Тема практичного заняття	Кількість годин
<i>Другий семестр</i>		
1	Ознайомлення зі структурою HTML-документу. Створення найпростішої Веб-сторінки (створення власної сторінки-візитки)	2
2	Літ.: [1-3, 6, 7] Створення та стилізування таблиці за допомогою елементів HTML та CSS	2
3	Літ.: [1-3, 6, 7] Стилізація Веб-сторінки за допомогою елементів CSS. Блокова-модель CSS.	2
4	Літ.: [1-3, 6, 7] Знайомство з синтаксисом, основними функціями та подіями в JavaScript	2
5	Літ.: [3, 12-17] Основи програмування сценаріїв веб-сторінки за допомогою мови JavaScript. Взаємодія з користувачем. Обробка подій і форми	2
6	Літ.: [1-3, 6, 7] Адаптивний дизайн веб-сторінок. Технологія Bootstrap (редизайн власної сторінки-візитки)	2
7	Літ.: [1-3, 6, 7] Ознайомлення із мовою програмування PHP	2
8	Літ.: [1-3, 6, 7] Підсумкове заняття	2
	Разом	18

2.4 Зміст самостійної (індивідуальної) роботи

Самостійна робота студентів денної форми навчання полягає у систематичному опрацюванні програмного матеріалу, підготовці до виконання і захисту практичних та лабораторних робіт, тестування з теоретичного матеріалу, виконанні індивідуальних завдань тощо.

№ п/п	Теми самостійної роботи	Кількість годин
1	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи №1 та до практичної роботи №1.	4
2	Опрацювання лекційного матеріалу. Підготовка до практичної роботи №2.	4
3	Опрацювання лекційного матеріалу. Підготовка до захисту лабораторної роботи №1. Підготовка до захисту практичної роботи №1.	4
4	Опрацювання лекційного матеріалу. Підготовка до практичної роботи №3.	4
5	Опрацювання лекційного матеріалу. Підготовка до захисту лабораторної роботи №2. Підготовка до захисту практичної роботи №2.	4
6	Опрацювання лекційного матеріалу. Підготовка до практичної роботи №4.	4
7	Опрацювання лекційного матеріалу. Підготовка до захисту лабораторної роботи №3. Підготовка до захисту практичної роботи №3.	4
8	Опрацювання лекційного матеріалу. Підготовка до практичної роботи №5.	4
9	Опрацювання лекційного матеріалу. Підготовка до захисту лабораторної роботи №4. Підготовка до захисту практичної роботи №4.	4
10	Опрацювання лекційного матеріалу. Підготовка до практичної роботи №6. Підготовка до тестування за темами 1-5.	4
11	Опрацювання лекційного матеріалу. Підготовка до захисту лабораторної роботи №5. Підготовка до захисту практичної роботи №5.	4
12	Опрацювання лекційного матеріалу. Підготовка до практичної роботи №7.	4
13	Опрацювання лекційного матеріалу. Підготовка до захисту лабораторної роботи №6. Підготовка до захисту практичної роботи №6.	4
14	Опрацювання лекційного матеріалу. Підготовка до практичної роботи №8.	4
15	Опрацювання лекційного матеріалу. Підготовка до захисту лабораторної роботи №7. Підготовка до захисту практичної роботи №7.	4
16	Опрацювання лекційного матеріалу. Підготовка до захисту практичної роботи №8. Підготовка до практичної роботи №8. Підготовка до тестування за темами 6-10.	6
17	Опрацювання лекційного матеріалу. Підготовка до підсумкових лабораторного та практичного занять	6
18	Опрацювання лекційного матеріалу. Підготовка до підсумкового контрольного заходу.	6
Загалом:		78

Керівництво самостійною роботою та контроль за виконанням індивідуального завдання здійснює викладач згідно з розкладом консультацій у позаурочний час.

3. МЕТОДИ НАВЧАННЯ

Лекції проводяться, в основному, з використанням словесних, наочних, проблемно-пошукових методів; практичні та лабораторні заняття проводяться пояснювально-ілюстративними методами, практичними та частково-пошуковими методами; самостійна робота передбачає виконання індивідуальних завдань із залученням практичних, дослідницьких, частково-пошукових методів.

4. ФОРМИ І МЕТОДИ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Поточний контроль здійснюється під час лекційних, практичних та лабораторних занять, а також у дні проведення контрольних заходів, встановлених робочим планом дисципліни.

Семестровий контроль проводиться у формі іспиту. При цьому при виведенні остаточної оцінки враховуються результати поточного контролю.

Кожний вид роботи з дисципліни оцінюється за чотирибальною шкалою. Семестрова підсумкова оцінка визначається як середньозважена з усіх видів навчальної роботи, виконаних і зданих позитивно з врахуванням коефіцієнта вагомості. Вагові коефіцієнти змінюються залежно від структури дисципліни і важливості окремих її видів робіт. Студент, який набрав позитивний середньозважений бал за поточну роботу і не здав підсумковий контрольний захід (іспит), вважається невстигаючим.

При оцінюванні знань студентів використовуються різні засоби контролю, зокрема: усне опитування перед допуском до виконання лабораторної та практичної робіт – здійснюється на їх початку; засвоєння теоретичного матеріалу з тем перевіряється тестовим контролем; якість виконання, набуття теоретичних знань і практичних навичок перевіряється шляхом захисту кожної лабораторної та практичної роботи, та індивідуального завдання згідно з робочою програмою дисципліни і робочим навчальним планом.

При оцінюванні знань студентів викладач керується такими критеріями. Оцінку „відмінно” за шкалою ECTS – A (див. шкалу оцінок) отримує студент за глибоке і повне опанування змісту навчального матеріалу з Вебтехнологій, в якому він легко орієнтується, понятійного апарату, за уміння зв'язувати теорію з практикою, вирішувати практичні завдання, висловлювати і обґрунтовувати свої судження і конструктивні рішення. Відмінна оцінка передбачає грамотний, логічний виклад відповіді (як в усній, так і в письмовій формі), якісне зовнішнє оформлення. Оцінка "відмінно" виставляється студенту, який глибоко засвоїв конструктивні особливості Вебтехнологій, та вміє їх раціонально застосувати, знає методики та вміє ними користуватися при проектуванні ПЗ. Студент не повинен вагатися при видозміні запитання, повинен робити детальні та узагальнюючі висновки.

Оцінку „добре” за шкалою ECTS – B, отримує студент за повне засвоєння навчального матеріалу, володіння понятійним апаратом Вебтехнологій, орієнтування в вивченому матеріалі, свідоме використання знань для вирішення практичних завдань, грамотний виклад відповіді, але у змісті і формі відповіді мали місце окремі неточності (похибки), нечіткі формулювання закономірностей тощо. Відповідь студента має будуватись на основі самостійного мислення. Оцінку „добре” за шкалою ECTS – C, отримує студент за правильну відповідь з двома-трьома суттєвими помилками.

Оцінки "задовільно" за шкалою ECTS – D, заслуговує студент, який виявив знання основного навчально-програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, що справляється з виконанням практичних завдань, передбачених програмою. Як правило, відповідь студента будується на рівні репродуктивного мислення, студент слабо знає структуру курсу, допускає помилки у відповіді, засвоїв і набув практичні навички з проектування ПЗ, але допустив неточності. Вагається при відповіді на видозмінене запитання, разом з тим студент володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у відповіді. Оцінки "задовільно" заслуговує студент за неповне опанування програмного матеріалу, але отримані знання і набуті практичні навички з розроблення ПЗ відповідають мінімальним критеріям оцінювання.

Оцінка „незадовільно” за шкалою ECTS – E, виставляється, коли студент має 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 знання при вирішенні практичних завдань. Як правило, оцінка "незадовільно" за шкалою ECTS – FХ, виставляється студенту, який не може продовжити навчання без додаткових знань з курсу.

Як результати навчання, отримані у неформальній освіті, зокрема онлайн-курси, може бути зараховано виконання двох лабораторних робіт.

На основі результатів поточного контролю і підсумкового контрольного заходу виставляється підсумкова семестрова оцінка. На основі аналізу контролю знань викладач удосконалює курс лекцій, звертаючи особливу увагу на ті розділи, чи теми, з яких було найбільше неточних відповідей, що свідчить про методичні чи інші недоліки при висвітленні вказаних тем або розділів. Аналогічно вносяться корективи в методичні посібники для лабораторних та практичних робіт, детальніше розглядаються принципові питання при виконанні лабораторних та практичних робіт та їх захисті.

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

Аудиторна робота								Самостійна, індивідуальна робота	Форма семестрового контролю
III семестр									
Лабораторні та практичні роботи №:								Тестовий контроль:	іспит
1	2	3	4	5	6	7	8	Т 1-16	1
ВК: 0,4								0,2	0,4

Примітка: Т – тема дисципліни; ВК – ваговий коефіцієнт;

Для переходу від вітчизняної оцінки до оцінки за шкалою ECTS необхідно знайти середньоарифметичну оцінку за вітчизняною шкалою, помножити її на відповідний ваговий коефіцієнт і, додавши всі складові, отримаємо суму балів, які визначають конкретну оцінку ECTS.

Оцінювання тестових завдань

Тематичний тест для кожного студента складається з двадцяти тестових завдань, кожне з яких оцінюється одним балом. Максимальна сума балів, яку може набрати студент, складає 25. Оцінювання здійснюється за чотирибальною шкалою. Відповідність набраних балів за тестове завдання оцінці, що виставляється студенту, представлена у нижченаведеній таблиці.

Сума балів за тестове завдання	1–13	14–16	17–22	23–25
Оцінка	2	3	4	5

Тестування проводиться з використанням модульного середовища для навчання MOODLE. Правильні відповіді студент реєструє в он-лайн режимі в модульному середовищі MOODLE. Викладач виставляє результати тестування згідно журналу оцінок модульного середовища MOODLE. Підсумкова семестрова оцінка за національною шкалою і шкалою ECTS встановлюється в автоматизованому режимі після внесення усіх оцінок до електронного журналу. Співвідношення вітчизняної шкали оцінювання і шкали оцінювання ECTS наведені у наступній таблиці.

Співвідношення вітчизняної шкали оцінювання і шкали оцінювання ECTS

Оцінка ECTS	Інституційна шкала балів	Інституційна оцінка	Критерії оцінювання	
A	4,75-5,00	5	Зараховано	
B	4,25-4,74	4		Відмінно – глибоке і повне опанування навчального матеріалу і виявлення відповідних умінь та навиків.
C	3,75-4,24	4		Добре – повне знання навчального матеріалу з кількома незначними помилками.
D	3,25-3,74	3		Добре – в загальному правильна відповідь з двома-трьома суттєвими помилками.
E	3,00-3,24	3		Задовільно – неповне опанування програмного матеріалу, але достатнє для практичної діяльності за професією.
			Задовільно – неповне опанування програмного матеріалу, що задовольняє мінімальні критерії оцінювання	

FX	2,00-2,99	2	Незараховано	Незадовільно – безсистемність одержаних знань і неможливість продовжити навчання без додаткових знань з дисципліни
F	0,00-1,99	2		Незадовільно – необхідна серйозна подальша робота і повторне вивчення дисципліни.

5. ПИТАННЯ ДЛЯ САМОКОНТРОЛЮ СТУДЕНТІВ

1. З яких частин складається документ HTML?
2. Що таке дескриптор (тег)?
3. Що таке контейнер?
4. Для чого потрібен заголовок документа і за допомогою яких дескрипторів він вводиться?
5. Для чого потрібен контейнер BODY?
6. Що визначають атрибути елемента BODY? Наведіть приклади.
7. Яким чином і за допомогою якого елемента здійснюється розбиття тексту на абзаци в HTML?
8. Яким чином і за допомогою якого елемента в HTML здійснюється розбиття тексту на рядки?
9. Для чого потрібне структурування тексту?
10. Назвіть два способи структурування тексту.
11. Яким чином і за допомогою якого елемента в HTML додаються заголовки?
12. Яким чином і за допомогою якого елемента в HTML додаються горизонтальні лінії?
13. Назвіть атрибути дескриптора <HR> та вкажіть їх призначення.
14. Вкажіть основне призначення дескриптора <DIV>.
15. Які два види форматування символів тексту ви знаєте?
16. Який дескриптор HTML служить для керування зовнішнім виглядом тексту Web-сторінки?
17. Яке посилання називають відносним? Наведіть його приклади.
18. Яке посилання називають абсолютним? Наведіть приклади.
19. Як створити посилання на інші документи? Поясніть на прикладі.
20. Що таке внутрішнє посилання і для чого воно потрібне? Наведіть його приклади.
21. Для чого потрібен елемент <LINK>? Чим він відрізняється від елемента <A>? Поясніть на прикладах.
22. Як створити посилання на адресу електронної пошти? Наведіть приклади посилань на адресу електронної пошти.
23. Яким чином і за допомогою якого елемента можна додати зображення в документ HTML?
24. Назвіть атрибути дескриптора та вкажіть їх призначення.
25. Як і за допомогою яких атрибутів елемента можна змінювати розміри зображення?
26. За допомогою якого атрибута елемента і яким чином можна вставити рамку, яка охоплює зображення?
27. Як додати вільний простір навколо зображення?
28. В яких випадках необхідно створювати альтернативний текст?
29. За допомогою якого дескриптора в код html-сторінки включаються елементи JavaScript
30. Як включити оператори JavaScript в документ HTML? Наведіть приклади.
31. Як встановити зв'язок HTML-документу із зовнішнім файлом JavaScript ? Наведіть приклади.
32. Яке призначення мають коментарі? Чим відрізняються однорядкові коментарі від багаторядкових? Поясніть на прикладах.
33. Які засоби налагодження сценаріїв ви знаєте?
34. Які види помилок сценаріїв вам відомі?
35. Для чого призначені функції alert й prompt? Наведіть приклади.
36. Використанням діалогового вікна confirm? Наведіть приклади.
37. Які математичні операції підтримуються в мові JavaScript?

38. Яка послідовність виконання математичних операцій у математичних виразах прийнята у мові JavaScript?
39. Які тригонометричні методи об'єкта Math ви знаєте і яке їхнє призначення?
40. Для чого призначені оператори for й if ?
41. Чим відрізняються конструкції оператора if та if ...else?
42. Чим відрізняються оператори if та switch?
43. Для чого призначене ключове слово default у конструкції оператора switch?
44. Для чого наприкінці рядка для кожного варіанта умови в операторі switch використовується ключове слово break? Що буде, якщо опустити це слово?
45. У чому складається відмінність операторів for, while та do...while? Як коректно замінити оператор for оператором while?
46. Як замінити оператор switch комбінацією операторів if ...else?
47. Яке призначення функцій parseInt і parseFloat?
48. З якою метою в HTML-документах використовують форми?
49. Як звернутися до форми в поточному HTML-документі?
50. Як можна довідатися кількість форм у поточному HTML-документі?
51. Що визначають атрибути ACTION та METHOD дескриптора <FORM>?
52. Для чого використовується дескриптор <SELECT>? Наведіть приклади його використання?
53. Які основні атрибути містить дескриптор <SELECT> і що вони визначають?
54. Для чого використовується дескриптор <TEXTAREA>? Наведіть приклади його використання?
55. Які основні атрибути містить дескриптор <TEXTAREA> і що вони визначають?
56. Для чого використовується дескриптор <INPUT>? Наведіть приклади його використання?
57. Які основні атрибути містить дескриптор <INPUT> і що вони визначають?
58. Які елементи форми можна додавати за допомогою атрибута TYPE дескриптора <INPUT>? Наведіть приклади.
59. Як одержати список всіх елементів форми?
60. Як можна одержати посилання на поточну форму з оброблювача події її елементів?
61. Як визначити URL сервера, куди будуть відправлені дані з поточної форми при виклику методу submit?
62. Які події в JavaScript Ви знаєте? Перерахуйте їх.
63. Вкажіть основне призначення оброблювачів подій onmouseover та onmouseout. Наведіть приклади їх використання.
64. Вкажіть основне призначення оброблювачів подій onclick та ondblclick. Наведіть приклади їх використання.
65. Вкажіть основне призначення оброблювачів подій onfocus та onblur. Наведіть приклади їх використання.
66. Вкажіть основне призначення оброблювача події onload. Наведіть приклади його використання.
67. Вкажіть основне призначення оброблювача події onchange. Наведіть приклади його використання.
68. Що таке PHP. У чому відмінність PHP від Javascript.
69. Які теги можна використати для визначення початку й кінця PHP-коду.
70. Для чого використовується конструкція echo()?
71. Як оформляються коментарі в PHP коді
72. Поясніть роботу оператора присвоєння
73. Які типи змінних існують в PHP?
74. Як оголошуються змінні в PHP?
75. Що таке визначені змінні?
76. Як одержати значення змінних, переданих у скрипт методами POST й GET
77. У чому відмінність використання одинарних лапок від подвійних?
78. Що таке конкантинація рядків, і яким образом вона виконується в PHP?

79. Назвіть основні типи даних, над якими здійснюються операції в PHP.
80. Які оператори вибору використовуються в PHP?
81. Наведіть приклади використання умовного оператора if...else.
82. Наведіть приклади використання перемикача switch.
83. Наведіть приклади використання операторів while та do...while.
84. Наведіть приклади використання оператора for.
85. Наведіть приклади ініціалізації одномірних масивів.

6. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

Навчальний процес з дисципліни забезпечений необхідними навчально-методичними розробками в модульному середовищі. Зокрема, викладачами кафедри було розроблено:

Павлова О.О., Денисюк Д.О., Гнатчук Є.Г., Кисіль Т.М. Методичні вказівки до виконання лабораторних робіт з навчальної дисципліни «Веб-технології» – Хмельницький: ХНУ, 2020. - 99 с.

7. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Steve Prettyman, Learn PHP 8 using MySQL, JavaScript, CSS3 and HTML5. Apress Media,USA, 2021. – 442 p.
2. Спірінцев В.В., Гнатушенко В.В., Волковський О.С. WEB-ТЕХНОЛОГІЇ ТА WEB-ДИЗАЙН: HTML, CSS Навчальний посібник. Дніпро. 2017.
3. Antonio Lopez, Learning PHP 7, Packt Publishing Ltd, Birmingham B3 2PB, UK, 2016 – 415p.
4. Vikram Vaswani, PHP: A Beginner's Guide, The McGraw-Hill Companies, 2016. – 478 p.
5. Mike McGrath, PHP & MySQL in easy steps, 2018. – 192 p.
6. Luke Welling, Laura Thomson, PHP and MySQL Web Development (5th Edition), 2016. – 688 p.
7. Sau Sheong Chang, Go Web Programming +code, 2016. – 312 p.
8. Kasun Indrasiri, Prabath Siriwardena, Microservices for the Enterprise: Designing, Developing, and Deploying, Apress Media,USA, 2018. – 422 p.
9. Evelina Buiciag, HTML5 applications development manual, 2016.- 190 p.
10. Micheal Knapp, HTML and CSS: Learn The Fundamentals In 7 days, 2017. – 98 p.
11. Thoriq Firdaus, Ben Frain, Benjamin LaGrone, HTML5 and CSS3: Building Responsive Websites, 2016. – 709 p.
12. Peter A. Carter, SQL Server Advanced Data Types: JSON, XML, and Beyond, 2018. – 392 p.
13. Gregory Blake, SQL Server 2017: A Practical Guide for Beginners, 2017. – 60 p.
14. Owen Kriev, SQL: The Comprehensive Beginners Guide to Learn SQL with Practical Examples, 2017. – 116 p.
15. Ikram Hawramani, Computer Programming for Complete Beginners: A Quick Course for Mastering the Basics of Coding through Interactive Steps and Visual Examples, 2018. -199 p.
16. Phil Ballard, JavaScript in 24 Hours, Sams Teach Yourself (7th Edition), 2018. – 432 p.
17. Jennifer Harder, Graphics and Multimedia for the Web with Adobe Creative Cloud: Navigating the Adobe Software Landscape, 2018. – 1017 p.

11. ІНФОРМАЦІЙНІ РЕСУРСИ

Електронний університет:

1. Модульне середовище для навчання (розміщені усі необхідні матеріали з дисципліни, в тому числі тестові завдання для поточного та семестрового контролю знань).
2. Електронна бібліотека університету