

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об'єктно-орієнтовані технології розробки програмних систем
Назва

Статус дисципліни: вибіркова

Факультет – інформаційних технологій

Кафедра – Комп'ютерної інженерії та інформаційних систем

Форма навчання	Курс	Семестр	Загальне навантаження		Кількість годин						Форма семестрового контролю			
			Кредити ЄКТС	Години	Аудиторні заняття				Індивідуальна робота студента	Самостійна робота, в т.ч. ІРС	Курсовий проект	Курсова робота	Залік	Іспит
					Разом	Лекції	Лабораторні роботи	Практичні заняття						
Д		парний	8.0	240	108	36	36	36		132	-	-	+	-
		Разом	8.0	240	108	36	36	36		132			1	

Робоча програма складена на основі стандарту вищої освіти зі спеціальності 126 Інформаційні системи та технології, освітньо-професійної програми та навчального плану

Програма складена


Підпис

Лисенком С.М.
Ініціали, прізвище викладача(ів)

Схвалена на засіданні кафедри КІС

Протокол № 1 від 30.08.2023 р.

Зав. кафедри КІС


Підпис

Говорущенко Т.О.
Ініціали, прізвище

Робоча програма розглянута та схвалена Вченою радою факультету інформаційних технологій

Голова Вченої ради


Підпис

Савенко О.С.
Ініціали, прізвище

Хмельницький 2023

ВСТУП

Мета викладання дисципліни. Дисципліна «

Об'єктно-орієнтовані технології розробки програмних систем» є вибірковою з програмування та характеризується широким міждисциплінарним підходом.

Метою курсу є оволодіння студентами основними принципами і методами об'єктно-орієнтованого програмування, зокрема, шаблонів проектування програмного забезпечення, які в подальшому можуть сприяти їх успішному застосуванню в професійній діяльності.

Таким чином, мета і завдання дисципліни «Об'єктно-орієнтовані технології програмування» навчити застосовувати принципи об'єктно-орієнтованого програмування на всіх етапах життєвого циклу прикладної програмної системи, починаючи з аналізу вимог до програмної системи і її попереднього проектування, і закінчуючи її реалізацією, тестуванням і наступним супроводом.

Предмет дисципліни. Застосування шаблонів проектування при розробленні програмного забезпечення, а також розроблення веб-сервісів із застосуванням об'єктно-орієнтованих технологій програмування.

Завдання дисципліни. Навчити застосовувати принципи об'єктно-орієнтованого програмування, а також шаблони проектування при розробленні програмного забезпечення та при розробленні веб-сервісів та додатків із застосуванням об'єктно-орієнтованих технологій програмування.

Після вивчення дисципліни «Веб-орієнтовані технології програмування» студент має досягти таких результатів навчання (сукупність знань, умінь, навичок, компетентностей):

знати:

- об'єкт, предмет, задачі, проблематику дисципліни та її основні розділи;
- наукові і математичні положення, що лежать в основі функціонування програмних систем; базові поняття й визначення, використовувані у галузі комп'ютерної інженерії; основи проектування програмного забезпечення із застосуванням об'єктно-орієнтованих технологій програмування;
- інновації у галузі програмної інженерії;

уміти:

- застосовувати основи комп'ютерної інженерії для розв'язування задач синтезу веб-сервісів та програмного забезпечення на основі об'єктно-орієнтованого підходу із застосуванням об'єктно-орієнтованих технологій програмування;
- системно мислити та застосовувати творчі здібності до формування принципово нових ідей щодо розроблення веб-сервісів та програмного забезпечення засобами об'єктно-орієнтованого програмування із застосуванням об'єктно-орієнтованих технологій програмування;
- застосовувати знання принципів об'єктно-орієнтованого програмування із застосуванням об'єктно-орієнтованих технологій програмування для ідентифікації, формулювання і розв'язування технічних задач спеціальності;
- вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем застосовуючи об'єктно-орієнтований підхід до програмування із застосуванням об'єктно-орієнтованих технологій програмування;
- застосовувати теорію та методи системного аналізу, математичного і комп'ютерного моделювання при проектуванні та розробленні веб-сервісів та програмного забезпечення на основі об'єктно-орієнтованого підходу із застосуванням об'єктно-орієнтованих технологій програмування;
- оцінювати результати роботи, захищати, пояснювати та аргументувати розроблений веб-сервіс та програмного забезпечення із застосуванням об'єктно-орієнтованих технологій програмування, отримані результати;
- вміти здійснювати пошук інформації в різних джерелах для розв'язання задач побудови веб-сервісів та програмного забезпечення на основі об'єктно-орієнтованого підходу програмування із застосуванням об'єктно-орієнтованих технологій програмування;

бути здатним:

- розв'язувати складні задачі і проблеми в галузі проектування веб-сервісів та програмного забезпечення із застосуванням об'єктно-орієнтованих технологій програмування, що передбачає проведення досліджень та/або здійснення інновацій;
- абстрактно мислити, аналізувати і синтезувати веб-сервіси та програмного забезпечення на основі об'єктно-орієнтованого підходу програмування із застосуванням об'єктно-орієнтованих технологій програмування;
- застосовувати знання принципів об'єктно-орієнтованого програмування на практиці при реалізації веб-сервісів та програмного забезпечення;
- розробляти веб-сервіси та програмне забезпечення, програмні компоненти комп'ютерних систем та мереж, Інтернет додатки, кіберфізичних систем з використанням об'єктно-орієнтованих методів програмування із застосуванням об'єктно-орієнтованих технологій програмування;
- створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж із застосуванням об'єктно-орієнтованого підходу програмування із застосуванням об'єктно-орієнтованих технологій програмування;
- проектувати програмні системи та їхні компоненти із застосуванням об'єктно-орієнтованих методів програмування із застосуванням об'єктно-орієнтованих технологій програмування з урахуванням усіх аспектів їх життєвого циклу ПЗ, включаючи створення, налаштування, експлуатацію та обслуговування.

ОБ'ЄКТНО-ОРІЄНТОВАНІ ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ

Тип дисципліни	вибіркова
Цикл	Другий (магістерський)
Мова викладання	Українська
Рік навчання	1 курс
Семестр	2
Кредити ЄКТС	8,0
Форми навчання, для яких викладається дисципліна	денна

Результати навчання Студент, який успішно завершив вивчення дисципліни, повинен: вміло застосовувати основи комп'ютерної інженерії для розв'язування задач синтезу веб-сервісів та програмного забезпечення на основі об'єктно-орієнтованого підходу із застосуванням об'єктно-орієнтованих технологій програмування; системно мислити та застосовувати творчі здібності до формування принципово нових ідей щодо розроблення веб-сервісів та програмного забезпечення засобами об'єктно-орієнтованого програмування із застосуванням об'єктно-орієнтованих технологій програмування; застосовувати знання принципів об'єктно-орієнтованого програмування із застосуванням об'єктно-орієнтованих технологій програмування для ідентифікації, формулювання і розв'язування технічних задач спеціальності; володіти технологією розроблення програмного забезпечення із застосуванням патернів проектування, а також технологією розроблення веб-сервісів та додатків патернів проектування із застосуванням об'єктно-орієнтованих технологій програмування.

Зміст навчальної дисципліни. Застосування патернів проектування при розробці програмного забезпечення; розроблення веб-сервісів та додатків патернів проектування із застосуванням об'єктно-орієнтованих технологій програмування.

Запланована навчальна діяльність: лекцій 36 год., лабораторних робіт 36 год., практичних робіт 18 год., самостійної роботи 150 год.; разом 240 год

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

Форми оцінювання результатів навчання: контрольна робота, захисти лабораторних робіт

Форма семестрового контролю: залік

Рекомендована література

1. Технічна документація ASP.NET Core Microsoft. <https://docs.microsoft.com/uk-ua/aspnet/core/?view=aspnetcore-7.0>
2. Дженніфер Грін, Ендрю Стеллман. Head First C#: Посібник для навчання з програмування в реальному світі за допомогою C# та .NET Core 4th Edition. O'Reilly Media. 2021 рік с.800 (англ).
3. Джозеф Альбахарі, Бен Альбахарі. C# 8.0 Pocket Reference: миттєва допомога для програмістів C# 8.0, 1-е видання. O'Reilly Media. 2019 . с.248 (англ).
4. Джозеф Альбахарі, Ерік Йоганнсен. C# 8.0 in Nutshell: The Definitive Reference 1st Edition. O'Reilly Media. 2020. с. 1088(англ).
5. Ian Griffiths. Programming C# 8.0: Build Windows, Web, and Desktop Applications 1st Edition. O'Reilly Media. 2020. С.802 (англ).
6. Joseph Albahari. C# 10 in a Nutshell. The Definitive Reference. O'Reilly Media.2022. с.1058 (англ)
7. Джозеф Альбахарі, Бен Альбахарі. Кишеньковий довідник C# 10. Миттєва допомога для програмістів C# 10. 1-е видання. O'Reilly Media. 2022. С.240 (англ)
8. Голуб М.Б. C#. Концепція та синтаксиси. Навч. Посібн. Львів: Видавничий центр ЛНУ Івана Франка, 2018. – 136 с.
9. Електронна бібліотека університету. Доступ до ресурсу: http://lib.khnu.km.ua/asp/php_f/page_lib.php

Викладач: доктор техн. наук, доцент Лисенко С.М.

3. СТРУКТУРА ЗАЛІКОВИХ КРЕДИТІВ ДИСЦИПЛІНИ

Назва теми	Кількість годин відведених на:			
	Лекції	Лабораторні роботи	Практичні роботи	СРС
Тема 1. Основи патернів проектування. Введення в патерни проектування.	4	4	2	18
Тема 2. Породжуючі патерни	4	4	2	18
Тема 3. Патерни поведінки	8	8	4	34
Тема 4. Структурні патерни	8	8	4	34
Тема 5. Принципи SOLID	8	4	2	34
Тема 6. Додаткові патерни. Fluent Builder	4	4	2	12
Підсумкове заняття	-	4	2	-
Години	36	36	18	150
Разом	240 (8.0 кредити)			

4. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

4.1. Зміст лекційного курсу*

№ п/п	Перелік тем лекцій, їх анотація	Години
Мова об'єктно-орієнтованого програмування C#. Загальні відомості про програми на C#.		
1.	Тема 1. Основи патернів проектування. Лекція№1. Введення в патерни проектування. Класифікація патернів проектування. Вибір патернів. [1,2,4].	2
2.	Тема 1. Основи патернів проектування. Лекція№2. Відношення між класами і об'єктами. Інтерфейси або абстрактні класи [1,2,4].	2
3.	Тема 2. Породжуючі патерни. Лекція№3. Фабричний метод (Factory Method). Абстрактна фабрика (Abstract Factory). [2-4]	2
4.	Тема 2. Породжуючі патерни. Лекція№4. Сінглтон і багатопоточність. Прототип (Prototype). Будівельник (Builder). [2-4]	2
5.	Тема 3. Патерни поведінки Лекція№5. Стратегія (Strategy). Спостерігач (Observer). Команда (Command). [3,5]	2
6.	Тема 3. Патерни поведінки Лекція№6. Макрокоманди. Шаблонний метод (Template Method). [3,5]	2
7.	Тема 3. Патерни поведінки. Лекція№7. Ітератор (Iterator). Стан (State). Ланцюжок Обов'язків (Chain of responsibility). [4, 8, 9]	2
8.	Тема 3. Патерни поведінки. Лекція№8. Інтерпретатор (Interpreter). Посередник (Mediator). Зберігач (Memento). Відвідувач (Visitor). [4, 8, 9]	2
9.	Тема 4. Структурні патерни Лекція№9. Декоратор (Decorator). Адаптер (Adapter). [5, 10].	2
10.	Тема 4. Структурні патерни Лекція№10. Фасад (Facade). Компоновщик (Composite). [5, 10].	2
11.	Тема 4. Структурні патерни Лекція№11. Замісник (Проксі). Міст (Bridge). [3,6].	2
12.	Тема 4. Структурні патерни Лекція№12. Паттерн Пристосуванець (Flyweight). [3,6].	2
13.	Тема 5. Принципи SOLID Лекція№13. Single Responsibility Principle (Принцип єдиних обов'язків). Open/Closed Principle (Принцип відкритості / закритості). [2,11].	2
14.	Тема 5. Принципи SOLID Лекція№14. Liskov Substitution Principle (Принцип підстановки Лисків). [2,11].	2
15.	Тема 5. Принципи SOLID Лекція№15. Interface Segregation Principle (Принцип поділу інтерфейсів). [2,11].	2
16.	Тема 5. Принципи SOLID Лекція№16. Dependency Inversion Principle (Принцип інверсії залежностей) [2,11].	2

17.	Тема 6. Додаткові патерни. Лекція№17. Патерн проектування Fluent Builder. [7, 12].	2
18.	Тема 6. Додаткові патерни. Лекція№18. Патерн проектування Fluent Builder (продовження). [7, 12].	2
	Разом	36

4.2 Зміст лабораторних занять

Таблиця 4 – Перелік лабораторних занять для студентів *денної* форми навчання

№ п/п	Теми лабораторних робіт	Кількість годин
1	<i>Лабораторна робота №1.</i> Введення в ASP.NET Core. Основи ASP.NET Core [1,4].	4
2	<i>Лабораторна робота №2.</i> Сервіси і Dependency Injection. Конфігурація [1-4].	4
3	<i>Лабораторна робота №3.</i> Стан додатки. Куки. Сесії. Логування. Маршрутизація [2,6,7,10].	4
4	<i>Лабораторна робота №4.</i> ASP.NET Core MVC. Контролери. Представлення. Моделі [2,6,7,10].	4
5	<i>Лабораторна робота №5.</i> Маршрутизація в ASP.NET Core MVC [1-4,10].	4
6	<i>Лабораторна робота № 6.</i> Помічники HTML. Помічники тегів. View Component [1,9,10].	4
7	<i>Лабораторна робота №7.</i> Метадані та валідація моделі [1,2,6,7].	4
8	<i>Лабораторна робота №8.</i> Робота з даними в Entity Framework в MVC [6,9,10,11]	4
9	<i>Підсумкове заняття.</i>	4
<i>Всього</i>		36

4.3 Зміст практичних занять

Таблиця 4 – Перелік практичних занять для студентів денної форми навчання

№ п/п	Теми практичних робіт	Кількість годин
1	<i>Практична робота №1.</i> Введення в ASP.NET Core. Основи ASP.NET Core [1,4].	2
2	<i>Практична робота №2.</i> Сервіси і Dependency Injection. Конфігурація [1-4].	2
3	<i>Практична робота №3.</i> Стан додатки. Куки. Сесії. Логування. Маршрутизація [2,6,7,10].	2
4	<i>Практична робота №4.</i> ASP.NET Core MVC. Контролери. Представлення. Моделі [2,6,7,10].	2
5	<i>Практична робота №5.</i> Маршрутизація в ASP.NET Core MVC [1-4,10].	2
6	<i>Практична робота № 6.</i> Помічники HTML. Помічники тегів. View Component [1,9,10].	2
7	<i>Практична робота №7.</i> Метадані та валідація моделі [1,2,6,7].	2
8	<i>Практична робота №8.</i> Робота з даними в Entity Framework в MVC [6,9,10,11]	2
9	<i>Підсумкове заняття.</i>	2
Всього		18

4.4 Зміст самостійної (індивідуальної) роботи

Об'єм самостійної роботи з дисципліни “Об’єктно-орієнтовані технології програмування” становить 150 години. Він включає опрацювання лекційного матеріалу, підготовку до виконання лабораторних робіт і їх захисту, підготовку до поточного контролю, а також самостійну роботу студентів.

Таблиця 6 – Зміст самостійної роботи студентів денної форми навчання

Номер тижня	Вид самостійної роботи	К-ть годин
1-2	Опрацювання лекційного матеріалу. Підготовка до ЛР1	18
3-4	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР1. Підготовка до ЛР2	18
5-6	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР2. Підготовка до ЛР3	17
7-8	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР3. Підготовка до ЛР4	17
9-10	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР4. Підготовка до ЛР5	17
11-12	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР5. Підготовка до ЛР6	17
13-14	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР6. Підготовка до ЛР7	17
15-16	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР7. Підготовка до ЛР8	17
17-18	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР8. Підготовка до ТК	12
	Разом за семестр:	150

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

5. МЕТОДИ НАВЧАННЯ

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних методів. Зокрема, лекції проводяться в основному методами проблемного викладання, словесними, наочними з використанням інформаційних технологій. Лабораторні заняття проводяться з використанням методів пояснювально-ілюстративних, проблемного викладання, дослідницьких, частково-пошукових з використанням інформаційних технологій і мають за мету – набуття студентами практичних навичок. Самостійна робота передбачає виконання індивідуальних завдань, при розв'язанні яких застосовуються методи проблемного викладання, дослідницькі, частково-пошукові.

6. 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.

Процес оцінювання підготовленості студента можна розділити на етапи:

Перший етап оцінювання направлений на визначення знань інформаційного мінімуму. Якщо студент твердо засвоїв визначену навчальним планом суму формальних знань, то це означає, що він вміє використати їх при вирішенні різних питань предметної області, вміє розширити їх. При цьому необхідно встановити рівні та критерії сформованості знань щодо змісту навчальних елементів. Такими рівнями є:

Ознайомчо-орієнтовний (ОО) – особа має орієнтовне уявлення щодо понять, які вивчаються, здатна: знаходити оптимальні розв'язки, застосовуючи формальні методи; застосовувати підходи до формальної специфікації семантики критичних систем.

Понятійно-аналітичний (ПА) – особа має чітке уявлення щодо навчального об'єкту, здатна перенести раніше засвоєні знання на типові ситуації.

Продуктивно-синтетичний (ПС) – особа має глибоке розуміння щодо навчального об'єкту, здатна здійснювати синтез, генерувати нові ідеї та уявлення, переносити раніше засвоєні знання на нетипові, нестандартні ситуації.

Формами і методами оцінювання результатів навчання є контрольна робота та захисти лабораторних робіт.

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: письмова контрольна робота, усне опитування перед допуском до виконання лабораторної роботи – здійснюється на її початку; засвоєння теоретичного матеріалу з тем перевіряється під час проведення лабораторних занять; якість виконання, набуття теоретичних знань і практичних навичок перевіряється шляхом захисту кожної лабораторної роботи згідно з робочою програмою дисципліни і робочим навчальним планом.

Письмова контрольна робота проводиться на останньому практичному занятті і включає два теоретичні питання та одну практичну задачу.

Оцінка, яка виставляється за лабораторне заняття, складається з таких елементів: усне опитування студентів перед допуском до виконання лабораторної роботи; знання теоретичного матеріалу з теми; якість оформлення протоколу і графічної частини; вміння студента обґрунтувати прийняті конструктивні рішення; своєчасний захист лабораторної роботи. Для виконання програми дисципліни студент повинен отримати вісім оцінок за лабораторні роботи в семестрі і написати підсумкову контрольну роботу на позитивну оцінку.

Термін захисту лабораторної роботи вважається своєчасним, якщо студент захистив її на

наступному після виконання роботи занятті. За несвоєчасний захист лабораторної роботи з неповажної причини студент за позитивну відповідь отримує оцінку «задовільно».

Пропущене лабораторне заняття студент повинен відпрацювати в лабораторіях кафедри у встановлений викладачем термін з реєстрацією у відповідному журналі кафедри, але не пізніше, ніж за два тижні до закінчення теоретичних занять у семестрі.

При оцінюванні знань студентів філософії викладач керується такими критеріями.

Оцінка "відмінно" виставляється студенту, який глибоко засвоїв методи розв'язування практичних задач та вмів їх раціонально застосувати. Студент не повинен вагатися при видозміні запитання, повинен робити детальні та узагальнюючі висновки.

Оцінку „добре” отримує студент за повне засвоєння навчального матеріалу, володіння понятійним апаратом, орієнтування у 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 повинна будуватись на основі самостійного мислення.

Оцінку „добре” отримує студент за правильну відповідь з однією-двома суттєвими помилками.

Оцінки "задовільно" 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 практичних навичок застосування методів розв'язування практичних задач, але припустився неточностей. 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 застосування технологій розв'язування практичних задач.

Оцінка „незадовільно” виставляється, коли студент має розрізнені, безсистемні знання, не вмів 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" виставляється студенту, який не може продовжити навчання без додаткових знань з курсу.

Кожний вид роботи оцінюється за чотирибальною шкалою. Семестрова підсумкова оцінка визначається як середньозважена з усіх видів робіт.

Таблиця 7– Структурування дисципліни за видами робіт і оцінювання результатів навчання студентів у семестрі за ваговими коефіцієнтами

Аудиторна робота		Самостійна, індивідуальна робота		Семестровий контроль (залік)
<i>II семестр</i>				
Лабораторні роботи:		Практичні роботи:	Оцінювання проектів	за рейтингом
1-8		1-8	контрольна робота	
ВК:	0,4	0,2	0,4	0

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 п'яти 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

На тестування відводиться 25 хвилин (для закритої форми тестів – по одній хвилині на кожне завдання). Тестування студент проходить в он-лайн режимі в модульному середовищі MOODLE.

Якщо студент отримав негативну оцінку, то він має перездати її в установленому порядку, але обов'язково до терміну наступного контролю. У випадку, коли студент не виконав індивідуальний план з дисципліни у заплановані терміни без поважних причин, то під час відпрацювання заборгованості при позитивній відповіді йому виставляється оцінка „задовільно”.

Підсумкова семестрова оцінка за національною шкалою і шкалою ЄКТС встановлюється в автоматизованому режимі після внесення усіх оцінок до електронного журналу. Співвідношення вітчизняної шкали оцінювання і шкали оцінювання ЄКТС наведені у табл. 9.

Залік виставляється, якщо середньозважений бал, який отримав студент з дисципліни, знаходиться в межах від 3,00 до 5,00 балів. При цьому за вітчизняною шкалою ставиться «зараховано», а за шкалою ЄКТС – буквене позначення оцінки, що відповідає набраній студентом кількості балів.

Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ECTS	Інституційна шкала балів	Інституційна оцінка	Критерії оцінювання	
A	4,75-5,00	5	Зараховано	Відмінно – глибоке і повне опанування навчального матеріалу і виявлення відповідних умінь та навиків.
B	4,25-4,74	4		Добре – повне знання навчального матеріалу з кількома незначними помилками.
C	3,75-4,24	4		Добре – в загальному правильна відповідь з двома-трьома суттєвими помилками.
D	3,25-3,74	3		Задовільно – неповне опанування програмного матеріалу, але достатнє для лабораторної діяльності за професією.
E	3.00-3,24	3		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.

8. ПИТАННЯ ДЛЯ САМОКОНТРОЛЮ СТУДЕНТІВ

1. Особливості ASP.NET CORE
2. Особливості створення проекту ASP.NET CORE
3. Створення контролера і представлень
4. Стилізація додатків
5. Основи контролерів
6. Методи дій і їх параметри
7. Результати дій
8. ViewResult і генерація представлень
9. Переведення і відправка кодів статусу і помилок
10. Відправка файлів в ASP.NET CORE 5
11. Контекст запиту HttpContext. Куки сесії
12. Асинхронні методи в ASP.NET CORE 5
13. Строго типізовані представлення
14. Майстер-сторінки
15. Часткові представлення

16. HTML-хелпери
17. Робота з формами
18. Строго типізовані хелпери
19. Моделі і БД
20. Підключення до бази даних
21. Шаблонові хелпери
22. Редагування моделі
23. Додавання і видалення моделі
24. Шаблони формування
25. Моделі зі складною структурою
26. Робота зі складними моделями
27. Моделі зі зв'язком багато-до-багатьох
28. Робота з моделями зі зв'язком багато-до-багатьох
29. Передача масивів і складних даних в контролер
30. Міграція баз даних
31. Створення пагінацію
32. Перевизначення шаблонів формування
33. Перевизначення шаблонів відображення і редагування
34. Фільтрація даних
35. Domain Model і View Model
36. Управління ієрархічними даними
37. Визначення маршрутів
38. Робота з маршрутами
39. Створення обмежень для маршрутів
40. Генерація вихідних адрес URL
41. Області в ASP.NET CORE 5
42. Створення власного обробника маршрутів
43. Атрибути маршрутизації
44. Маршрутизація і вкладені ресурси
45. Анотації даних для відображення властивостей
46. Основи валідації
47. Атрибути валідації
48. Валідація моделі в контролері
49. Відображення помилок валідації
50. Створення власної логіки валідації
51. Фільтри в ASP.NET CORE 5
52. Фільтри аутентифікації
53. Фільтри авторизації
54. Фільтри винятків
55. Фільтри дій і результатів
56. Приклади фільтрів дій
57. Глобальні фільтри і перевизначення фільтрів
58. Додаткові вбудовані фільтри
59. Логування винятків
60. Введення в прив'язку моделей
61. DefaultModelBinder
62. Явна прив'язка моделі
63. Постачальники значень

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Технічна документація ASP.NET Core Microsoft. <https://docs.microsoft.com/uk-ua/aspnet/core/?view=aspnetcore-6.0>.
2. Дженніфер Грін, Ендрю Стеллман. Head First C#: Посібник для навчання з програмування в реальному світі за допомогою C# та .NET Core 4th Edition. O'Reilly Media. 2021 рік с.800 (англ).
3. Джозеф Альбахарі, Бен Альбахарі. C# 8.0 Pocket Reference: миттєва допомога для програмістів C# 8.0, 1-е видання. O'Reilly Media. 2019 . с.248 (англ).
4. Джозеф Альбахарі, Ерік Йоганнсен. C# 8.0 in Nutshell: The Definitive Reference 1st Edition. O'Reilly Media. 2020. с. 1088(англ).
5. Ian Griffiths. Programming C# 8.0: Build Windows, Web, and Desktop Applications 1st Edition. O'Reilly Media. 2020. С.802 (англ).
6. Коноваленко І.В. Програмування мовою C# 6.0. Тернопіль, ТНТУ. 2016. С.227.
7. Д. В. Настенко, А. Б. Нестерко. ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ Частина 1. Основи об'єктно-орієнтованого програмування на мові C#. Київ НТУУ «КПІ» 2016 с.76.
8. Павлишко А.В. Савельєва О.В. АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ. Одеса .2017 с.123.
9. І. О. ЗАВАДСЬКИЙ, Р. І. ЗАБОЛОТНИЙ , Основи візуального програмування. Київ Видавнича група ВНУ. С.271.
10. Коноваленко І.В., Марущак П.О., Савків В.Б. Програмування мовою C# 7.0. Тернопіль 2017. с.302.
11. Joseph Albahari. C# 7.0 Pocket Reference: Instant Help for C# 7.0 Programmers 1st Edition. O'Reilly Media. 2017. С.240 (англ).
12. Joseph Albahari. C# 10 in a Nutshell. The Definitive Reference. O'Reilly Media.2022. с.1058 (англ)
13. Джозеф Альбахарі, Бен Альбахарі. Кишеньковий довідник C# 10. Миттєва допомога для програмістів C# 10. 1-е видання. O'Reilly Media. 2022. С.240 (англ)
14. Електронна бібліотека університету. Доступ до ресурсу: http://lib.khnu.km.ua/asp/php_f/page_lib.php.

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. Модульні курси з дисципліни для дистанційної форми навчання (повний комплект матеріалів)
3. Електронна бібліотека університету