

Міністерство освіти і науки України

Тернопільський національний технічний університет імені Івана Пулюя

(повне найменування вищого навчального закладу)

Факультет комп'ютерно-інформаційних систем і програмної інженерії

/назва факультету/

Кафедра комп'ютерних наук

/назва кафедри/

ЗАТВЕРДЖУЮ

Декан факультету

Ігор БАРАН

(підпис) (прізвище та ініціали)

« 102 » 09 2024 р.



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

КРОС-ПЛАТФОРМНЕ ПРОГРАМУВАННЯ

/назва дисципліни/

галузь знань

12 Інформаційні технології

/шифр і назва галузі знань/

рівень вищої освіти

перший (бакалаврський)

/назва/

спеціальність

122 Комп'ютерні науки

/шифр і назва/

освітня програма

«Комп'ютерні науки»

/назва/

спеціалізація

/назва/

вид дисципліни

вибіркова дисципліна циклу професійної підготовки

/обов'язкова/вибіркова/

Тернопіль – 2024 рік

Робоча програма з навчальної дисципліни Крос-платформне програмування
/назва дисципліни/

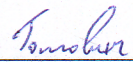
для студентів факультету комп'ютерно-інформаційних систем і програмної інженерії
/назва факультету(ів)/

Розробник:

доцент кафедри комп'ютерних наук,

кандидат технічних наук, доцент

/посада, науковий ступінь та вчене звання/


/підпис/

Володимир ГОТОВИЧ

/ініціали та прізвище/

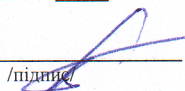
Робоча програма розглянута та схвалена

на засіданні кафедри комп'ютерних наук

/назва/

Протокол від « 26 » серпня 2024 року № 1

Завідувач кафедри


/підпис/

Ігор БОДНАРЧУК

/ініціали та прізвище/

Робоча програма розглянута та схвалена НМК

факультету комп'ютерно-інформаційних систем і програмної інженерії

Протокол від « 02 » вересня 2024 року № 1

Секретар НМК


/підпис/

Богдана МЛИНКО

/ініціали та прізвище/

Робоча програма погоджена:

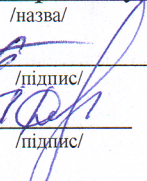
Спеціальність 122 Комп'ютерні науки

/шифр і назва/

освітня програма «Комп'ютерні науки»

/назва/

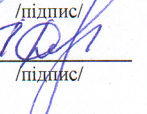
Завідувач випускової кафедри


/підпис/

Ігор БОДНАРЧУК

/ініціали та прізвище/

Гарант освітньої програми


/підпис/

Леся ДМИТРОЦА

/ініціали та прізвище/

1. Структура навчальної дисципліни

Показник	Всього годин	
	Денна форма навчання (ДФН)	Заочна (дистанційна) форма навчання (ЗФН)
Кількість кредитів/год	3/90	3/90
Аудиторні заняття, год.	42	12
Самостійна робота, год.	48	78
Аудиторні заняття:		
– лекції, год.	14	6
– лабораторні заняття, год.	28	6
– практичні заняття, год.	–	–
– семінарські заняття, год.	–	–
Самостійна робота:		
– підготовка до лабораторних (практичних – семінарських) занять, год.	18	22
– опрацювання окремих розділів програми, які не виносяться на лекції, год.	26	47
– виконання контрольних завдання, год.	–	–
– виконання індивідуальних завдань, год.	–	–
– виконання курсових проектів (робіт), год.	–	–
– підготовка та складання <u>заліків</u> , <u>екзаменів</u> , контрольних робіт, рефератів, есе, <u>тестування</u> , год.	4	9
Екзамен	–	–
Залік	3	3

Частка годин самостійної роботи студента:

денна форма навчання - 53 %;

заочна (дистанційна) форма навчання - 87 %.

2. Мета та завдання навчальної дисципліни

2.1. Мета вивчення навчальної дисципліни

Основною метою дисципліни “Крос-платформне програмування” є отримання студентами теоретичних знань та практичних навичок у сфері сучасних підходів і технологій проектування та розробки крос-платформного програмного забезпечення, а також поглиблення практичного досвіду при використанні мови програмування C# в процесі розробки програмних застосунків такого типу.

Предметом вивчення дисципліни “Крос-платформне програмування” є сучасні технології крос-платформного та компонентно-орієнтованого проектування програмного забезпечення із використанням можливостей платформи Microsoft .NET та мови програмування C#.

Важливим аспектом успішного засвоєння навчального матеріалу дисципліни є самостійна робота студентів з використанням інструментальних засобів розробки та тестування програмного забезпечення (інтегрованих середовищ розробки) та відповідної технічної літератури.

2.2. Завдання навчальної дисципліни

Завданням дисципліни є оволодіння студентами теоретичними основами проектування архітектури крос-платформного програмного забезпечення а також набуття теоретичних знань та практичних навичок розробки крос-платформних мобільних додатків на основі платформи .Net MAUI від Microsoft. По завершенню вивчення дисципліни студент повинен набути таких знань та вмінь:

Знати:

1. Основні терміни та поняття щодо крос-платформного програмного забезпечення.
2. Основні терміни та поняття щодо програмних компонентів та компонентно-орієнтованого підходу до розробки програмного забезпечення.
3. Принципи організації (зв'язності) та сполучуваності компонентів.

4. Найпоширеніші архітектурні стилі та шаблони, які використовуються при проектуванні крос-платформного програмного забезпечення.

5. Загальні рекомендації щодо побудови архітектури крос-платформної програмної системи.

6. Засоби, які надає середовище Microsoft Visual Studio для розробки та відлагодження програмного забезпечення на основі платформи .Net MAUI.

7. Основні засоби, які надає платформа .Net MAUI щодо проектування, розробки та розгортання крос-платформних мобільних додатків.

Вміти:

1. Проектувати крос-платформне програмне забезпечення на основі використання принципів об'єктно-орієнтованого та компонентно-орієнтованого підходів;

2. Використовувати можливості інструментального середовища Microsoft Visual Studio при розробці програмного забезпечення на основі платформи Xamarin (.Net MAUI);

3. Застосовувати набуті знання в професійній діяльності при розробці крос-платформних мобільних додатків на основі платформи .Net MAUI.

Вивчення студентами дисципліни сприяє формуванню в них наступних програмних результатів навчання, відповідно до освітньої програми:

- ПР5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.

- ПР14. Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктноорієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничотехнічних систем.

3. Опис навчальної дисципліни

3.1. Лекційні заняття

№	Тема заняття та короткий зміст	Кількість годин	
		ДФН	ЗФН
1	2	3	4
1.	Тема 1. Вступ до крос-платформного програмування. Поняття крос-платформності, її типи. Крос-платформність на різних рівнях. Крос-платформні мови та середовища виконання. Крос-платформне програмне забезпечення. Портування і мобільність програмного забезпечення. Підходи до реалізації кросплатформності. Переваги та недоліки кросплатформної розробки ПЗ.	1	0,5
2.	Тема 2. Означення та властивості компонентів. Визначення та властивості компонентів. Узагальнена модель компонентної системи. Компонентна модель .Net Framework. Динамічна бібліотека DLL як приклад компонента.	1	0,5
3.	Тема 3. Принципи організації (зв'язності) компонентів. Принципи SOLID. Принципи організації (зв'язності) компонентів. Діаграма протиріч при визначенні зв'язності компонентів.	1	0,5
4.	Тема 4. Принципи сполучуваності компонентів Принцип ациклічності залежностей. Принцип стійких залежностей. Принцип стійкості абстракцій.	1	0,5
5.	Тема 5. Архітектурні стилі та шаблони. Поняття архітектурного стилю. Поєднання архітектурних стилів. Архітектура клієнт/сервер. Компонентна архітектура. Проектування на основі предметної області. Багаторівнева архітектура. Архітектура, що базується на шині повідомлень. N-рівнева / 3-рівнева архітектура. Сервісно-орієнтована архітектура.	1,5	0,5
6.	Тема 6. Загальні рекомендації щодо побудови архітектури програмної системи. Поняття архітектури програмної системи та її вплив на різні аспекти функціонування системи. Чиста архітектура. Перетин кордонів між колами архітектури.	1	0,25

1	2	3	4
7.	Тема 7. Основи платформи .Net MAUI. Основні платформи, присутні сьогодні на ринку мобільних пристроїв та перешкоди в розробці крос-платформних мобільних додатків. Платформа .Net MAUI. Життєвий цикл додатку. Інсталяція фреймворку .Net MAUI та найпростіший проєкт додатку. Запуск додатку для відлагодження. Структура програмного рішення типового додатку .Net MAUI. Ресурси проєкту.	1,5	0,5
8.	Тема 8. Основи мови розмітки XAML. Загальна характеристика XAML. Структура типового файлу. Задання вмісту сторінки. Сторінкова навігація. Взаємодія програмного коду і XAML. Особливості синтаксису XAML. Розширення розмітки XAML.	1	0,5
9.	Тема 9. Додатки на основі .NET MAUI Shell. Візуальна ієрархія елементів інтерфейсу в додатку на основі Shell. Загальна характеристика навігації в додатку на основі Shell. Функція пошуку. Опис візуальної ієрархії елементів інтерфейсу в додатку. Елементи інтерфейсу типу Flyout. Елементи інтерфейсу типу Tab. Елементи інтерфейсу типу Page.	1	0,5
10.	Тема 10. Стилi та теми. Стилi: Загальна характеристика. Явні (explicit) стилi. Неявні (implicit) стилi. Застосування стилiв до похідних типiв елементiв iнтерфейсу. Глобальнi стилi. Динамiчнi стилi. Класи стилiв. Теми: Загальна характеристика. Оголошення тема за замовчуванням. Використання тем. Завантаження теми динамiчно.	1	0,25
11.	Тема 11. Елементи компонування iнтерфейсу користувача. Загальна характеристика елементiв компонування iнтерфейсу користувача. Сторiнки (Pages). Макети (Layouts). Представлення (Views). Позицювання та вирiвнювання елементiв iнтерфейсу.	1	0,5
12.	Тема 12. Основи прив'язки даних. Загальна характеристика механiзму прив'язки даних. Прив'язка перегляду до перегляду (View-to-view bindings). Режими прив'язки. Форматування стрiчок. Прив'язка без контексту прив'язки. Прив'язка даних та колекцiї.	1	0,5

1	2	3	4
13.	Тема 13. Прив'язка даних та архітектурний шаблон MVVM. Загальна характеристика архітектурного шаблону MVVM. Реалізація простого MVVM в додатках .Net MAUI. Інтерактивний MVVM.	1	0,5
Усього годин		14	6

3.2. Лабораторні заняття

№	Тема заняття	Кількість годин	
		ДФН	ЗФН
1.	Створення бібліотеки класів .NET	4	1
2.	Створення додатку веб-API на основі ASP.NET Core	4	1
3.	Створення додатку на основі платформи ASP.NET Core MVC	4	1
4.	Створення простого додатку на платформі .Net MAUI	4	1
5.	Створення багатоекранного додатку	4	1
6.	Налаштування макету розмітки інтерфейсу користувача	4	1
7.	Створення додатку з використанням механізму прив'язки	4	-
Усього годин		36	6

3.3. Самостійна робота

№	Найменування робіт	Кількість годин	
		ДФН	ЗФН
1	2	3	4
1.	Тема 1. Вступ до крос-платформного програмування. Портування і мобільність програмного забезпечення.	3	4
2.	Тема 2. Означення та властивості компонентів. Узагальнена модель компонентної системи. Компоненті моделі COM, COM+, DCOM, CORBA.	3	5
3.	Тема 3. Принципи організації (зв'язності) компонентів. Взаємозв'язок між принципами SOLID та принципами організації (зв'язності) компонентів.	3	4
4.	Тема 4. Принципи сполучуваності компонентів. Граф залежностей компонентів. Залежність між стійкістю та абстрактністю компонента.	3	4
5.	Тема 5. Архітектурні стилі та шаблони. Поеднання архітектурних стилів. Порівняння найпоширеніших архітектурних стилів. Їх переваги та недоліки.	3	5
6.	Тема 6. Загальні рекомендації щодо побудови архітектури програмної системи. Загальна характеристика чистої (ідеальної) архітектури. Перетин кордонів між колами чистої архітектури.	3	4
7.	Підготовка до тестування	2	4
8.	Тема 7. Основи платформи .Net MAUI. Відлагодження додатку з використанням емулятора та фізичного пристрою. Запуск додатку на виконання на різних платформах. Міграція додатку із Xamarin.	4	5
9.	Тема 8. Основи мови розмітки XAML. Синтаксис XAML. Розширення розмітки XAML. Універсальні шаблони. Простори імен.	4	5
10.	Тема 9. Додатки на основі .NET MAUI Shell. Навігація в додатку. Маршрути. Стек навігації. Події навігації.	4	5

1	2	3	4
11.	Тема 10. Стилi та теми. Успадкування стилiв. Застосування стилiв на основi CSS.	3	4
12.	Тема 11. Елементи компонування iнтерфейсу користувача. Застосування рiзних елементiв iнтерфейсу (Views) та рiзних макетiв (Layouts) в межах одного екрану. Анiмацiя. Жести.	4	5
13.	Тема 12. Основи прив'язки даних. Вiдносна прив'язка. Множинна прив'язка. Команди.	4	5
14.	Тема 13. Прив'язка даних та архiтектурний шаблон MVVM. Команди та взаємодiя iз користувачем в рамках MVVM. Скомпiльованi прив'язки.	3	4
15.	Пiдготовка до тестування	2	5
Усього годин		48	78

Важливою частиною самостiйної роботи є практичне застосування середовища розробки (наприклад, Visual Studio) для виконання як запропонованих завдань лабораторних робiт так i зразкiв програмного коду, якi мiстяться в лекцiях та лiтературних джерелах.

Крiм того, до самостiйної роботи також належать пiдготовка до захисту лабораторних робiт i пiдготовка до складання змiстових модулiв.

4. Критерії оцінювання результатів навчання студентів

Форма підсумкового семестрового контролю – **залік**

Підсумкова семестрова оцінка (див. табл. нижче) складається з суми балів, отриманих студентом при захисті лабораторних робіт, двох тестувань (за перший і другий модулі) та отриманих балів за залік (підсумкова семестрова оцінка).

За виконання 7-ми лабораторних робіт студент отримує максимально до 42 балів (7 * 6).

МОДУЛЬ 1			МОДУЛЬ 2			ПІДСУМКОВА СЕМЕСТРОВА ОЦІНКА	РАЗОМ З ДИСЦИПЛІНИ
АУДИТОРНА ТА САМОСТІЙНА РОБОТА			АУДИТОРНА ТА САМОСТІЙНА РОБОТА				
ТЕОРЕТИЧНИЙ КУРС (ТЕСТУВАННЯ)	ЛАБОРАТОРНА РОБОТА		ТЕОРЕТИЧНИЙ КУРС (ТЕСТУВАННЯ)	ЛАБОРАТОРНА РОБОТА			
СЕМЕСТРОВА ОЦІНКА						25	100
15	18		18	24		за кожних три бали семестрової оцінки студент отримує 1 бал підсумкової	
№ ЛЕКЦІЙ	ВИД РОБІТ	БАЛ	№ ЛЕКЦІЙ	ВИД РОБІТ	БАЛ		
Лекція 1			Лекція 8	Лаб. роб. №4	6		
Лекція 2	Лаб. роб. №1	6	Лекція 8				
Лекція 3			Лекція 9	Лаб. роб. №5	6		
Лекція 4			Лекція 10				
Лекція 5	Лаб. роб. №2	6	Лекція 11	Лаб. роб. №6	6		
Лекція 6	Лаб. роб. №3	6	Лекція 12	Лаб. роб. №7	6		
			Лекція 13				

Звіт з кожної лабораторної роботи повинен містити тему, мету роботи, номер варіанту завдання, знімки екрану (демонструють отримані результати) та висновки по роботі. Звіт завантажується студентом в скриньку для завдань. Програмний код розроблених під час виконання лабораторної роботи програм завантажується у репозиторій (GitLab, GitHub). Номер варіанту завдання визначається на основі порядкового номера студента в списку групи.

Захист лабораторних робіт відбувається усно, шляхом відповіді на 3-4 запитання викладача, які стосуються теоретичних положень виконаної роботи та програмного коду, розробленого студентом. При цьому враховуються: повнота та

точність відповідей студента, степінь виконання завдань роботи (виконано всі завдання чи не всі), акуратність оформлення звіту і програмного коду.

За результатами проходження модульного контролю №1 студент отримує від 0 до 15 балів, модульного контролю №2 – від 0 до 18 балів.

Модульний контроль відбувається з використанням системи дистанційного навчання, шляхом відповіді студентом на послідовність тестових запитань за обмежений час. При цьому необхідно дати відповіді як на запитання з одним правильним варіантом відповіді (серед чотирьох запропонованих варіантів), так і на запитання із декількома правильними варіантами відповіді (серед п'яти запропонованих варіантів). Максимально допускається до трьох спроб здачі модульного тесту, з яких зараховується найкраща спроба. На модульний тест виносяться запитання, які покривають теоретичний та практичний матеріал модуля.

Контроль здійснюється засобами електронного навчального курсу (ЕНК) на сервері дистанційного навчання <http://dl.tntu.edu.ua>, ідентифікатор дисципліни ID 2551.

До підсумкового семестрового контролю (складання семестрового заліку) допускаються студенти, які протягом семестру виконали всі види навчальної роботи, успішно пройшли проміжні (модульні) контролі і набрали не менше 45 балів семестрової оцінки та за умови отримання не менше 60% балів за результатами кожного проміжного (модульного) контролю рівня знань.

Підсумкова оцінка записується за 100-бальною шкалою із подальшим переведенням її у шкалу Європейської кредитно-трансферної системи (ECTS) відповідно A, B, C, D, E, F, FX. При цьому чотирибальна шкала оцінок (з записом семестрової оцінки «відмінно» - A, «добре» - B,C, «задовільно» - D, E відповідають підсумковому результату «зараховано», «незадовільно» - F, FX відповідає підсумковому результату «не зараховано»).

5. Навчально-методичне забезпечення

1. Конспект лекцій з дисципліни «Крос-платформне програмування» для здобувачів освітнього ступеня «бакалавр» спеціальності 122 «Комп'ютерні науки» всіх форм навчання / укл.: Готович В.А. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2022 р., 277 с.

2. Методичні вказівки до виконання лабораторних робіт з дисципліни «Крос-платформне програмування» для здобувачів освітнього ступеня «бакалавр», спеціальності 122 «Комп'ютерні науки» всіх форм навчання / укл.: Готович В.А., Гром'як Р.С. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2022 р., 94 с.

3. Методичні вказівки для самостійної роботи студентів з дисципліни «Крос-платформне програмування» для здобувачів освітнього ступеня «бакалавр» спеціальності 122 «Комп'ютерні науки» всіх форм навчання / укл.: Готович В.А. – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя, 2023 р., 16 с.

6. Рекомендована література

Основна

1. Clean Architecture: A Craftsman's Guide to Software Structure and Design (Robert C. Martin Series) ISBN-13: 9780134494166.

2. Microsoft Application Architecture Guide, 2nd Edition (Patterns & Practices). ISBN-13: 978-0735627109 [Електронний ресурс]. – Режим доступу: [https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff650706\(v=pandp.10\)](https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff650706(v=pandp.10))

3. NET Multi-platform App UI documentation. [Електронний ресурс]. – Режим доступу: <https://learn.microsoft.com/en-us/dotnet/maui/?view=net-maui-8.0>

4. Build mobile and desktop apps with .NET MAUI. [Електронний ресурс]. – Режим доступу: <https://learn.microsoft.com/en-us/training/paths/build-apps-with-dotnet-maui>

Допоміжна

1. Creating Mobile Apps with Xamarin.Forms. Charles Petzold. [Електронний ресурс]. – Режим доступу: <https://learn.microsoft.com/en-us/xamarin/xamarin-forms/creating-mobile-apps-xamarin-forms/>

2. Functional Design: Principles, Patterns, and Practices (Robert C. Martin Series) ISBN-13: 978-0138176396

7. Інформаційні ресурси

1. Електронний навчальний курс “Крос-платформне програмування” в системі дистанційного навчання ТНТУ імені І. Пулюя. [Електронний ресурс]. – Режим доступу: <https://dl.tntu.edu.ua/bounce.php?course=2551>

2. .NET MAUI Course for Beginners – Create Cross-Platform Apps with C#. [Електронний ресурс]. – Режим доступу: https://www.youtube.com/watch?v=n3tA3Ku65_8

3. Learn .NET MAUI - Full Course for Beginners. [Електронний ресурс]. – Режим доступу: https://www.youtube.com/watch?v=DUNLR_NJv8U

4. Youtube-канал про розробку мобільних додатків з використанням Xamarin.Forms. [Електронний ресурс]. – Режим доступу: <https://www.youtube.com/watch?v=XRVflcWyy-s&list=PLH3y3SWteZd2xUX46npPFspQar5BTHUIh>

8. Зміни та доповнення до робочої програми навчальної дисципліни

№	Зміст внесених змін (доповнень)	Дата і № протоколу засідання кафедри	Примітки
1	2	3	4

1	2	3	4