МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ АВТОМОБІЛЬНО-ДОРОЖНІЙ УНІВЕРСИТЕТ

Кафедра комп'ютерних технологій і мехатроніки

«ЗАТВЕРДЖУЮ»
Гарант освітньо-професійної програми «Програмне забезпечення систем» першого (бакалаврського) рівня вищої освіти, завідувач кафедри КТМ, д.т.н.,

професор _____ Ніконов О.Я.
«03» 09 2019р.

СИЛАБУС
КРОС-ПЛАТФОРМНЕ ПРОГРАМУВАННЯ / CROSS-PLATFORM PROGRAMMING SYLLABUS

освітній ступінь	**бакалавр** / bachelor
галузь знань	**12 Інформаційні технології** / Information Technology
спеціальність	**121 Інженерія програмного забезпечення** / Software Engineering
освітня програма	**Програмне забезпечення систем** / Systems Software

Харків 2019

Автор: Тімонін Володимир Олексійович, кандидат технічних наук, старший науковий співробітник, доцент

Силабус розглянуто та затверджено на засіданні кафедри комп'ютерних технологій і мехатроніки, протокол № 18 від «27» червня 2019 р.

СИЛАБУС

КРОС-ПЛАТФОРМНЕ ПРОГРАМУВАННЯ /

CROSS-PLATFORM PROGRAMMING

SYLLABUS

освітній ступінь	**бакалавр** / bachelor
галузь знань	**12 Інформаційні технології /** Information Technology
спеціальність	**121 Інженерія програмного забезпечення /** Software Engineering
освітня програма	**Програмне забезпечення систем /** Systems Software

ВСТУП

Одну з лідируючих позицій у розвитку сьогоднішнього світу займають IT-технології. З розвитком і впровадженням в повсякденне життя інформаційно-комунікаційних технологій різко зріс інтерес до програмування. Сучасним фахівцям доводиться мати справу з величезною кількістю різноманітних мов програмування різних рівнів і призначень. Потреба в ефективних засобах розробки програмного забезпечення призвела до появи об'єктно-орієнтованих систем програмування, призначених для швидкої розробки програмних додатків.

Спочатку крос-платформна розробка мала на увазі створення контенту для ПК і ігрових приставок. В даний час крос-платформність означає доступність програми не тільки на ПК, але і на мобільних пристроях з різними операційними системами. Крос-платформність розглядається як засіб підвищення мобільності, стійкості, зниження вартості розробки. Крос-платформний підхід передбачає уніфікацію здебільшого програмного коду програми, його незалежність від подальшого вибору платформи.

Крос-платформне програмне забезпечення - програмне забезпечення, яке працює більш ніж на одній апаратній платформі і / або операційній системі.

Програма вивчення навчальної дисципліни "**Крос-платформне програмування**" складена відповідно до освітньо-кваліфікаційної характеристики та навчального плану підготовки **бакалавра** спеціальності **121 «Інженерія програмного забезпечення»**.

Викладання дисципліни розраховано на виконання студентами практичних робіт та самостійних завдань у програмному середовищі MS Visual Studio.

1. МЕТА, ПРЕДМЕТ ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1.1. **Метою** вивчення навчальної дисципліни є вивчення сучасних технологій програмування для різних архітектур і платформ, теоретична і практична підготовка фахівців у галузі сучасних технологій програмування на рівні професійних вимог за напрямом.

1.2. **Предметом** вивчення навчальної дисципліни є концепція крос-платформного програмування, принципи крос-платформного програмування, методи і засоби реалізації в середовищі візуального програмування, а також методи їх використання при розробленні додатків.

1.3.**Основними завданнями** вивчення навчальної дисципліни є формування у студентів необхідних знань у галузі сучасних технологій програмування; формування системного базового уявлення, первинних знань, умінь та навичок з основ крос-платформного програмування; вивчення етапів створення додатків в інтегрованих середовищах розробки; вироблення практичних навичок аналітичного та експериментального дослідження основних методів і засобів, що використовуються в області крос-платформного програмування при рішенні прикладних задач.

1.4. **По завершені вивчення дисципліни студенти повинні**:

знати:

особливості програмування для різних операційних систем; основні аспекти концепції крос-платформного програмування; призначення основних компонентів середовища розробки; особливості розробки графічних інтерфейсів користувача; особливості програмування, керованого подіями; особливості оброблення виняткових ситуацій; особливості обробки подій від команд меню, елементів управління тощо; правила використання бібліотек класів середовищ розробника програм.

вміти:

розробляти програми для роботи в різних операційних системах; створювати крос-платформні програми на рівні виконання;створювати програми на крос-платформних інтерпретованих мовах; проектувати людино-машинний інтерфейс інформаційних систем; реалізувати прототипи архітектури програмного забезпечення; установлювати, налаштовувати та обслуговувати системне, інструментальне і прикладне програмне забезпечення та інформаційні системи.

1.5. **Міждисциплінарні зв'язки**:

Вивченню дисципліни:

передують: "Алгоритмізація та програмування", "Об'єктно-орієнтоване програмування", "Вища математика", "Дискретна математика", "Теорія алгоритмів";

потребують її вивчення: "Організація баз даних і знань", "Програмування в ОС Android", "Веб-технології та веб-дизайн", "Технологія створення програмних продуктів", "Технологія розподілених систем та паралельних обчислень", "Методи та системи штучного інтелекту", "Проектно-технологічна практика ", "Дипломне проектування".

2.ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	**Характеристика навчальної дисципліни**
Кількість кредитів - 5 Кількість годин - 150	нормативна (нормативна, за вибором ВНЗ, за вибором студента)
Семестр викладання дисципліни	6
Вид контролю:	екзамен (залік, екзамен)
Розподіл часу:	
- лекції (годин)	16
- практичні, семінарські (годин)	-
- лабораторні роботи (годин)	32
- самостійна робота студентів (годин)	72
- курсовий проект (годин)	-

- курсова робота (годин)	-
розрахунково-графічна робота (контрольна робота)	-
підготовка та складання екзамену	30

3. РОЗПОДІЛ ДИСЦИПЛІНИ У ГОДИНАХ ЗА ФОРМАМИ ОРГАНІЗАЦІЇ ОСВІТНЬОГО ПРОЦЕСУ ТА ВИДАМИ НАВЧАЛЬНИХ ЗАНЯТЬ

Результати навчання	**Навчальна діяльність**	**Робочий час студента (год.)**	**Оцінювання (бал)**
1	2	3	4
Тема 1. Крос-платформна розробка додатків.			
Знати *особливості програмування для різних операційних систем; основні аспекти концепції крос-платформного програмування.* ***Вміти*** *встановлювати і налаштовувати засоби розробки; створювати крос-платформні програми на рівні виконання; створювати програми на крос-платформних інтерпретованих мовах; проектувати*	Лекція №1. Крос-платформна розробка додатків. *Основні питання:* 1. Крос-платформне програмне забезпечення. 2. Крос-платформні мови програмування. 3. Крос-платформні середовища виконання. 4. Крос-платформні призначені для користувача інтерфейси. *Список рекомендованих джерел:* Основний – 4 Інтернет-ресурси –13, 14, 15	2	
	Самостійна робота студента (СРС[1]). Введення в мову XAML. *Основні питання:* 1. Коротка характеристика мови XAML. 2. Елементи мови XAML та їх властивості. 3. Взаємодія XAML і C #.	3	
	Практична робота №1 (ПР1). Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення простих	2	0…100

[1] Завданнями для самостійної роботи є вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем, а також в практичному відпрацюванню незавершених завдань в практичних роботах.

людино-машинний інтерфейс;	крос-платформних додатків. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички зі створення простих крос-платформних додатків.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Методика створення крос-платформних програм. *Основні питання:* 1. Встановлення та налаштування Xamarin. 2. Створення та налагодження проекту в Visual Studio. 3. Створення програми в Visual Studio.	3	
	ПР2. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з текстовими полями. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички по створенню крос-платформних додатків з текстовими полями.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.	2	0…100

	СРС. Структура крос-платформного додатка. *Основні питання:* 1. Рішення та проекти. 2. Структура крос-платформного додатку. 3. Структура основного проекту. 4. Структура проекту для Windows. 5. Структура проекту для Android. 6. Структура проекту для iOS.	3	
Тема 2. Елементи в Xamarin і їх властивості.			
Знати *призначення основних компонентів середовища розробки; особливості розробки графічних інтерфейсів користувача; особливості обробки подій від елементів управління; засоби розробки крос-платформних додатків; методику створення крос-платформних додатків.* ***Вміти*** *створювати крос-платформні програми на рівні виконання; створювати програми на крос-платформних інтерпретованих*	Лекція №2. Елементи в Xamarin і їх властивості. *Основні питання:* 1. Структура класів XamarinForms. 2. Елементи в Xamarin і їх властивості. 3. Спливаючі вікна. *Список рекомендованих джерел:* Основний – 1 ÷ 8 Додатковий – 9 ÷ 12 Інтернет-ресурси –13, 14, 15	2	
	СРС. Xamarin і крос-платформна розробка. *Основні питання:* 1. Короткі відомості про Xamarin. 2. Переносимість коду. 3. Сторонні компоненти.	3	
	ПР3. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з елементами управління. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички по створенню крос-платформних додатків з елементами управління.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків).	2	0…100

мовах; проектувати людино-машинний інтерфейс; використовуючи елементи Xamarin, розробляти крос-платформні додатки.	3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Основні елементи фреймворку Xamarin *Основні питання:* 1. Елементи введення і виведення і їх властивості. 2. Елементи управління і їх властивості.	3	
	ПР4. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з додатковими елементами управління. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички по створенню крос-платформних додатків з елементами управління.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.	2	0…100
	СРС. Розробка обробників подій. *Основні питання:* 1. Поняття про обробниках подій. 2. Способи створення обробників подій.	3	
Тема 3. Робота з зображеннями.			
Знати *призначення основних компонентів середовища розробки; особливості розробки графічних*	Лекція №3. Робота з зображеннями. *Основні питання:* 1. Елемент Image і його властивості. 2. Робота з локальними зображеннями. 3. Робота з вбудованими зображеннями. 4. Робота з зображеннями з мережі.	2	

інтерфейсів користувача; особливості програмування, керованого подіями; особливості роботи з зображеннями. ***Вміти*** *створювати крос-платформні програми на рівні виконання; створювати програми на крос-платформних інтерпретованих мовах; проектувати людино-машинний інтерфейс; розробляти крос-платформні додатки з використанням зображень.*	*Список рекомендованих джерел:* Основний – 1 ÷ 8 Додатковий – 9 ÷ 12 Інтернет-ресурси –13, 14, 15		
	СРС. Класифікація зображень в мобільних пристроях. *Основні питання:* 1. Поняття про зображень в мобільних пристроях. 2. Розміщення зображень у проекті. 3. Особливості роботи з зображеннями.	3	
	ПР5. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків, що працюють з зображеннями. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички по створенню крос-платформних додатків, що працюють з зображеннями.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.	2	0…100
	СРС. Елементи для роботи з часом і датою. *Основні питання:* 1. Елементи DatePicker і його властивості. 2. Елементи TimePicker і його властивості.	3	
	ПР6. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з таймерами. *Мета роботи - досліджувати можливості інтегрованого середовища*	2	0…100

	розробки Visual Studio і отримати практичні навички по створенню крос-платформних додатків з таймерами. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Створення додатків з графічними примітивами. *Основні питання:* 1. Об'єкт Drawable. 2. Створення об'єктів Drawable в коді програми. 3. Клас TransitionDrawable. 4. Клас ShapeDrawable. 5. Малювання на канві.	3	
	Тема 4. Робота з таблицями.		
Знати *призначення основних компонентів середовища розробки; особливості розробки графічних інтерфейсів користувача; особливості програмування, керованого подіями; особливості створення додатків з таблицями і вкладками.*	Лекція №4. Робота з таблицями. *Основні питання:* 1. Елемент TableView і його властивості. 2. Клас Cell і його властивості. 3. Робота з об'єктами EntryCell, SwitchCell, TextCell, ImageCell. *Список рекомендованих джерел:* Основний – 1 ÷ 8 Додатковий – 9 ÷ 12 Інтернет-ресурси –13, 14, 15	2	
	СРС. Типи осередків в таблицях. *Основні питання:* 1. Клас Cell і його властивості. 2. Об'єкт EntryCell і його властивості. 3. Об'єкт SwitchCell і його властивості. 4. Об'єкт TextCell і його властивості. 5. Об'єкт ImageCell і його властивості.	3	

Вміти *створювати крос-платформні програми на рівні виконання; створювати програми на крос-платформних інтерпретованих мовах; проектувати людино-машинний інтерфейс; створювати додатки з таблицями і вкладками.*			
	ПР7. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з таблицями. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички по створенню крос-платформних додатків з таблицями.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.	2	0…100
	СРС. Способі створення і роботі з вкладками. *Основні питання:* 1. Клас TabbedPage. 2. Методика створення програми з вкладками.	3	
	ПР8. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з вкладками. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички зі створення додатків з вкладками.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.	2	0…100

	СРС. Розробка коду, що залежить від платформи. *Основні питання:* 1. Поняття про платформо-залежним коді. 2. Особливості розробки коду, що залежить від платформи. 3. Платформо залежність в XAML.	3	
Тема 5. Контейнери.			
Знати *правила використання бібліотек класів середовищ розробника програм; сучасні стилі оформлення графічного інтерфейсу; особливості створення додатків для різних видів інтерфейсів користувача.* ***Вміти*** *створювати крос-платформні програми на рівні виконання; проектувати людино-машинний інтерфейс; створювати додатки для різних видів інтерфейсів користувача.*	Лекція №5. Контейнери. *Основні питання:* 1. Поняття про контейнерах. 2. Елемент StackLayout і його властивості. 3. Елемент AbsoluteLayout і його властивості. 4. Елемент RelativeLayout і його властивості. 5. Елемент Grid і його властивості. *Список рекомендованих джерел:* Основний – 1 ÷ 8 Додатковий – 9 ÷ 12 Інтернет-ресурси –13, 14, 15	2	
	СРС. Основні елементи компоновки. *Основні питання:* 1. Поняття про елементи компоновки. 2. Особливості компоновки єлементів.	3	
	ПР9. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з елементами компонування. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички зі створення додатків з елементами компонування.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків).	2	0…100

	3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Застосування елемента RelativeLayout. *Основні питання:* 1. Елемент RelativeLayout і його властивості. 2. Позиціонування щодо сторін контейнера. 3. Позиціонування щодо інших елементів.	3	
	ПР10. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з елементами компонування (контейнер Grid). *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички зі створення додатків з елементами компонування (контейнер Grid).* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.	2	0…100
	СРС. Застосування компонентів в елементі Grid. *Основні питання:* 1. Елемент Grid і його властивості. 2. Особливості використання елемента Grid. 3. Способи використання елемента Grid.	3	
Тема 6. Прив'язка даних.			
Знати *правила використання бібліотек класів*	Лекція №6. Прив'язка даних. *Основні питання:* 1. Особливості прив'язки даних в	2	

середовищ розробника програм; особливості прив'язки даних; засоби прив'язки даних; типи ресурсів. ***Вміти*** *створювати крос-платформні програми на рівні виконання; проектувати людино-машинний інтерфейс; створювати додатки з прив'язкою даних; налаштовувати зовнішній вигляд додатку; використовувати різні ресурси.*	Xamarin Forms. 2. Засоби прив'язки даних. 3. Інтерфейс INotifyPropertyChanged. *Список рекомендованих джерел:* Основний – 1 ÷ 8 Додатковий – 9 ÷ 12 Інтернет-ресурси –13, 14, 15		
	СРС. Особливості прив'язки даних в мобільних пристроях. *Основні питання:* 1. Патерн Model-View-ViewModel. 2. Команди взаємодія з користувачем в MVVM.	3	
	ПР11. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків з прив'язкою даних. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички зі створення додатків з прив'язкою даних.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.	2	0…100
	СРС. Ресурси в Xamarin. *Основні питання:* 1. Концепція ресурсів в Xamarin. 2. Ресурси додатків. 3. Управління ресурсами в коді C #. 4. Динамічні ресурси.	3	
	ПР12. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків, що працюють зі списками. *Мета роботи - досліджувати можливості інтегрованого середовища*	2	0…100

	розробки Visual Studio і отримати практичні навички зі створення додатків, що працюють зі списками. План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Налаштування зовнішнього вигляду об'єкту ListView. *Основні питання:* 1. Елементи класу ListView і їх властивості. 2. Створення роздільник рядків. 3. Визначення висоти рядків. 4. Створення заголовка і футера об'єкта ListView.	3	
	Тема 7. Навігація.		
Знати *правила використання бібліотек класів середовищ розробника програм; механізм навігації; принципи і способи передачі даних при навігації.* ***Вміти*** *створювати крос-платформні програми на рівні виконання; проектувати людино-машинний інтерфейс;*	Лекція №7. Навігація. *Основні питання:* 1. Загальні відомості про навігації. 2. Методика створення програми з навігацією. 3. Робота зі стеками навігації. 4. Управління навігацією. 5. Передача даних при навігації. *Список рекомендованих джерел:* Основний – 1 ÷ 8 Додатковий – 9 ÷ 12 Інтернет-ресурси –13, 14, 15	2	
	СРС. Робота з об'єктами TextCell, ImageView, ViewCell. *Основні питання:* 1. Об'єкт TextCell і його властивості. 2. Об'єкт ImageCell і його властивості. 3. Об'єкт ViewCell і його властивості.	3	
	ПР13. Дослідження можливостей	2	0…100

створювати додатки, що використовують навігацію.	інтегрованого середовища розробки Visual Studio для створення додатків, що працюють зі списками складних об'єктів. *Мета роботи - досліджувати можливості інтегрованої середовища розробки Visual Studio і отримати практичні навички зі створення додатків, що працюють зі списками складних об'єктів.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.		
	СРС. Робота зі стеками навігації. *Основні питання:* 1. Можливості навігації по сторінках. 2. Методи, що використовуються при навігації, і їх особливості.	3	
	ПР14. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків, що використовують навігацію. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички зі створення додатків, що використовують навігацію.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи.	2	0…100

	4. Тестування знань.		
	СРС. Передача даних в крос-платформних програмах. *Основні питання:* 1. Передача даних за допомогою конструктора. 2. Передача даних за допомогою BindingContext.	3	
Тема 8. Робота з файлами.			
Знати *правила використання бібліотек класів середовищ розробника програм; особливості файлової структури в мобільних пристроях; принципи роботі баз даних на мобільних пристроях.* ***Вміти*** *створювати крос-платформні програми на рівні виконання; створювати додатки, що працюють з файлами; конструювати архітектуру мобільного додатку з урахуванням принципів роботи баз даних.*	Лекція №8. Робота з файлами. *Основні питання:* 1. Загальні відомості про файли. 2. Класи для роботи з файлами. 3. Методика створення додатків, що працюють з файлами. *Список рекомендованих джерел:* Основний – 1 ÷ 8 Додатковий – 9 ÷ 12 Інтернет-ресурси –13, 14, 15	2	
	СРС. Класи для роботи з файлами. *Основні питання:* Клас ApplicationData (властивості, методи, події). Клас StorageFolder (властивості, методи, події).	3	
	ПР15. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення навігаційних додатків. Передача даних при навігації. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички зі створення додатків, що передають дані при навігації.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи.	2	0…100

	4. Тестування знань.		
	СРС. Особливості створення додатків з файлами під ОС Windows, Android та iOS. *Основні питання:* 1. Класи, інтерфейси і методи, що використовуються при створенні додатків під ОС Windows. 2. Класи, інтерфейси і методи, що використовуються при створенні додатків під ОС Android.	3	
	ПР16. Дослідження можливостей інтегрованого середовища розробки Visual Studio для створення додатків, що працюють з файлами. *Мета роботи - досліджувати можливості інтегрованого середовища розробки Visual Studio і отримати практичні навички зі створення додатків, що працюють з файлами.* План заняття: 1. Актуалізація теоретичного матеріалу. 2. Виконання завдань практичної роботи (розробка алгоритму, написання додатків, обробка помилок, тестування додатків). 3. Презентація і захист виконаної роботи. 4. Тестування знань.	2	0…100
	СРС. Використання баз даних в мобільних пристроях. *Основні питання:* 1. Налаштування проекту для роботи з SQLite.NET. 2. Основні операції з SQLite.NET. 3. Підключення до існуючої бази даних. 4. Асинхронне підключення до SQLite.	3	
Разом		120	0…100

5. КОНТРОЛЬ ТА ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Під час вивчення дисципліни «Крос-платформне програмування» викладачем здійснюється поточний та підсумковий контроль з використанням тестових завдань. Поточний контроль та оцінювання передбачає:

- перевірку рівня засвоєння теоретичного матеріалу (тестування за матеріалами лекції);
- захист практичних робіт (перевірка розроблених проектів та тестування за матеріалами практичних робіт);
- перевірка засвоєння матеріалу, що винесений на самостійне опрацювання під час фронтального опитування на лекції.

Тестування знань студентів здійснюється під час проведення практичних занять за допомогою автоматизованої системи знань. Підсумкова оцінка (ПО) знань з навчальної дисципліни визначається як середньозважена результатів засвоєння окремих блоків в інтервалі (1…100) балів:

$$\text{ПО} = \frac{\sum_{i=1}^{I} K_i}{N},$$

де К – кількість балів за кожен блок,

I – кількість зданих блоків,

N – загальна кількість блоків за дисципліну.

У разі незгоди студента з підсумковою оцінкою, підсумковий контроль знань та компетентностей студента з навчальної дисципліни здійснюється на підставі екзамену.

Відповідність підсумкових семестрових рейтингових оцінок у балах оцінкам за національною шкалою та шкалою ECTS

Оцінка в балах	Оцінка за національною шкалою	Оцінка за шкалою ЄКТС	
		Оцінка	Критерії
90-100	**Відмінно**	**A**	**"Відмінно"** - теоретичний зміст курсу освоєний цілком, без прогалин, необхідні практичні навички роботи з освоєним матеріалом сформовані, усі передбачені програмою навчання навчальні завдання виконані, якість їхнього виконання оцінено числом балів, близьким до максимального.
82 – 89	**Добре**	**B**	**"Дуже добре"** - теоретичний зміст курсу освоєний цілком, без прогалин, необхідні практичні навички роботи з освоєним матеріалом в основному сформовані, усі передбачені програмою навчання навчальні завдання виконані, якість виконання більшості з них оцінено числом балів, близьким до максимального.
75 – 81		**C**	**"Добре"** - теоретичний зміст курсу освоєний цілком, без прогалин, деякі практичні навички роботи з освоєним матеріалом сформовані недостатньо, усі передбачені програмою навчання навчальні завдання виконані, якість виконання жодного з них не оцінено мінімальним числом балів, деякі види завдань виконані з помилкам
67 – 74	**Задовільно**	**D**	**"Задовільно"** - теоретичний зміст курсу освоєний частково, але прогалини не носять істотного характеру, необхідні практичні навички роботи з освоєним матеріалом в основному сформовані, більшість передбачених програмою навчання навчальних завдань виконано, деякі з виконаних завдань, можливо, містять помилки.
60 – 66		**E**	**"Достатньо"** - теоретичний зміст курсу освоєний частково, деякі практичні навички роботи не сформовані, багато передбачені програмою навчання навчальні завдання не виконані, або якість виконання деяких з них оцінено числом балів, близьким до мінімального.
35 – 59	**Незадовільно**	**FX**	**"Незадовільно"** - теоретичний зміст курсу освоєний частково, необхідні практичні навички роботи не сформовані, більшість передбачених програм навчання навчальних завдань не виконано, або якість їхнього виконання оцінено числом балів, близьким до мінімального; при додатковій самостійній роботі над матеріалом курсу можливе підвищення якості виконання навчальних завдань(з можливістю повторного складання)
1 – 34		**F**	**" Неприйнятно "** - теоретичний зміст курсу не освоєно, необхідні практичні навички роботи не сформовані, усі виконані навчальні завдання містять грубі помилки, додаткова самостійна робота над матеріалом курсу не приведе до якого-небудь значимого підвищення якості виконання навчальних завдань.(з обов'язковим повторним курсом)

6. ЧАС ТА МІСЦЕ ПРОВЕДЕННЯ:

- аудиторні заняття – відповідно до розкладу ХНАДУ, ауд. 214, 216;
- позааудиторна робота – самостійна робота студента, результат виконання якої висвітлено засобами Visual Studio.

7. СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Основний

1. Petzold C. Creating Mobile Apps with Xamarin.Forms. Cross-platform C# programming for iOS, Android, and Windows. – Вашингтон: Microsoft Press, 2016. – 1187 с.: ил.
2. Hermes D. Xamarin Mobile Application Development. Cross-Platform C# and Xamarin.Forms Fundamentals. – Изд-во: Apress, 2015. – 425 с.: ил.
3. Ликнесс Дж. Приложения для Windows 8 на C# и XAML. – СПб.: Питер, 2013. – 368 с.: ил.
4. Прайс Марк Дж. C# 7 и .NET Core. Кросс-платформенная разработка для профессионалов. 3-е изд. — СПб.: Питер, 2018. — 640 с.: ил.
5. Тімонін В.О. Методичні вказівки до проведення практичних робіт з дисципліни «Крос-платформне програмування» для студентів за спеціальністю 121 «Інженерія програмного забезпечення». – Харків: ХНАДУ, 2019. – 84 с.
6. Харди Б., Филлипс Б., Стюарт К., Марсикано К. Android. Программирование для профессионалов. 2-е изд. – СПб.: Питер, 2016. – 640 с.: ил.
7. Шилдт, Г. Java: руководство для начинающих, 5-е изд.: Пер. с англ. – М.: ООО “И.Д.Вильямс”, 2012. – 624 с.: ил.
8. Фелкер Д. Android: разработка приложений для чайников. Пер. с англ. – М.: ООО “И.Д. Вильямс”, 2012. – 336 с.: ил.

Додатковий

9. Дейтел П., Дейтел Х., ДейтелЭ., Моргано М. Android для программистов: создаём приложения. – СПб.: Питер, 2013. – 560 с.: ил.
10. Петцольд Ч. Программирование для Microsoft Windows 8. Разработка для Windows Store на C# - М.: «Русская Редакция», 2014.- 1008 с.: ил.
11. Харди Б., Филлипс Б., Стюарт К., Марсикано К. Android. Программирование для профессионалов. 2-е изд. – СПб.: Питер, 2016. – 640 с.: ил.
12. Хорстманн К., Корнелл Г. Java. Библиотека профессионала, том1. Основы. Пер. с англ. – М.: ООО “И.Д. Вильямс”, 2014. – 864 с.: ил.

Інтернет-ресурси

13. https://developer.xamarin.com.
14. https://msdn.microsoft.com.
15. http://www.w3ii.com.