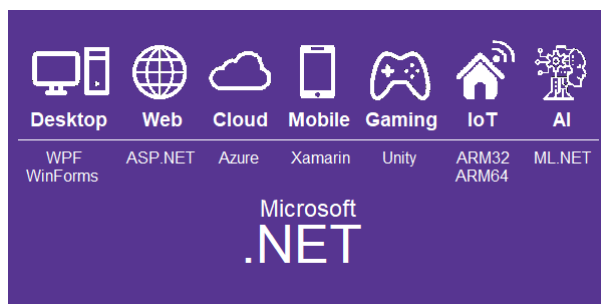


[RE-88] ТЕХНОЛОГІЇ .NET ДЛЯ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	G - Інженерія, виробництво та будівництво
Спеціальність	G5 - Електроніка, електронні комунікації, приладобудування та радіотехніка
Освітня програма	Всі ОП
Статус дисципліни	Вибіркова (Ф-каталог)
Форма здобуття вищої освіти	Очна
Рік підготовки, семестр	Доступно для вибору починаючи з 2-го курсу, осінній семестр
Обсяг дисципліни	4 кред. (Лекц. 16 год, Практик. 30 год, Лаб. 0 год, СРС. 74 год)
Семестровий контроль/контрольні заходи	Залік
Розклад занять	https://schedule.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лекц.: Нікітчук А. В. , Практ.: Нікітчук А. В. ,
Розміщення курсу	https://do.ipk.kpi.ua/course/view.php?id=5927

Програма навчальної дисципліни

1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Мета: ознайомитись з основними концепціями і особливостями платформи *Microsoft .NET* (.NET), отримати знання та навички, необхідні для промислової розробки програмних продуктів із відповідним функціоналом для радіотехнічних інформаційних систем.

.NET — це платформа, яка відображає новітні тенденції в розробці і пропонує спеціалістам-початківцям безліч можливостей. Знання основ .NET дозволять бути гнучкими у виборі

спеціалізації та сфери програмування.

Основні цілі дисципліни:

1. Вивчення основних понять та компонентів платформи *.NET*.
2. Знайомство з інструментами та середовищем розробки *Visual Studio* для підвищення продуктивності при розробці додатків.
3. Освоєння мови програмування *C#* для створення десктопних, веб- та мобільних додатків на платформі *.NET*.
4. Використання різних бібліотек та фреймворків, які входять до *.NET*-екосистеми, для реалізації різних функціональностей.
5. Ознайомлення з розробкою веб-додатків за допомогою технології *ASP.NET Core*, включаючи роботу з *MVC (Model-View-Controller)*.
6. Створення баз даних на сервері *MSSQL*, підключення до проекту і використання.
7. Ознайомлення з розробкою мобільних та корсплатформових додатків на платформі *.NET*, за допомогою *MAUI* (в минулому *Xamarin*).
8. Вивчення принципів та шаблонів програмування, основ рефакторингу, для постійного покращення якості програмного продукту.

В результаті вивчення дисципліни формуються наступні компетенції:

ПК 19	Здатність застосовувати технологію об'єктно-орієнтованого програмування та базові патерни проектування при створенні програмного забезпечення із відповідним функціоналом для радіотехнічних інформаційних систем та реалізовувати програми в різних середовищах програмування.
ПК 20	Здатність обирати методи та засоби обробки інформації із застосуванням інтелектуальних технологій.
ПРН 4	Застосовувати бази даних, математичне і програмне забезпечення для обробки даних та комп'ютерного моделювання телекомунікаційних та радіотехнічних систем, та інтелектуальних технологій радіоелектроніки.

Студент буде знати:

- мову програмування *C#*;
- основні концепції технології *.NET* та базові принципи створення програмного забезпечення за її допомогою;
- основні бібліотеки, що утворюють *.NET*.

Студент буде вміти:

- працювати в середовищі програмування *Microsoft Visual Studio*;
- створювати програми на мові *C#* з використанням об'єктно-орієнтованої технології;
- створювати ПЗ з графічним інтерфейсом користувача (для десктопного, веб- та мобільного використання);
- розробляти ПЗ для взаємодії з базами даних;
- розробляти ПЗ для роботи в телекомунікаційній мережі Інтернет за допомогою технології *ASP.NET Core*.

Крім цього, за допомогою *.NET* на *C#* можливо створювати програми IoT (інтернету речей) для Raspberry Pi, HummingBoard, BeagleBoard, Pine A64 тощо. Використовуючи наявні бібліотеки та фреймворки з відкритим кодом для взаємодії зі спеціалізованим обладнанням, таким як датчики, аналого-цифрові перетворювачі та LCD-пристрої. Також, за допомогою *.NET* можна створювати комп'ютерні ігри під платформу Unity (саме так створювались такі відомі ігри, як: *Hearthstone: Heroes of Warcraft*; *Cities Skylines*; *Cuphead*; *Pillars of Eternity*; *Pokemon Go*; *Subnautica* та інші).

2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі

навчання за відповідною освітньою програмою)

До початку вивчення дисципліни студент повинен володіти:

- знаннями з інформатики та базовими навичками програмування (дисципліни: Інформатика, Вступ до спеціальності);
- англійською мовою (або навичками використання онлайн перекладачів).

Пов'язані дисципліни:

- [Контроль якості програмного забезпечення](#).
- [Програмування вбудованих систем Інтернету речей](#).
- [Комп'ютерні мережі та безпека за технологіями CISCO](#).

3. Зміст навчальної дисципліни

Тема 1. Комп'ютери, програмування та Microsoft.NET

Тема 2. Основи виконання програм та мови програмування C#

Тема 3. Основні оператори C#

Тема 4. Особливості реалізації ООП та взаємодії між класами

Тема 5. Обробка винятків. Основи LINQ

Тема 6. Інтерфейс програмування додатків

Тема 7. Створення веб-застосунків за архітектурним шаблоном MVC

Тема 8. Принципи та шаблони проектування програмного забезпечення

4. Навчальні матеріали та ресурси

Електронні ресурси

1. Бібліотека документації та навчальних ресурсів Microsoft для розробників та інших професіоналів, які працюють із технологіями — <https://docs.microsoft.com/uk-ua/>
2. W3Schools — найбільший безкоштовний освітній веб-сайт для розробників та навчання програмуванню онлайн. Щороку відображається 3 мільярди сторінок. 60 мільйонів відвідувачів щомісяця — <https://www.w3schools.com/>
3. C# Corner — Глобальна мережева спільнота розробників програмного забезпечення. У 2021 році C# Corner обслужив 29,4 мільйона відвідувачів — <https://www.c-sharpcorner.com/>
4. Programiz — навчальний веб-сайт для вивчення програмування. Мільйони користувачів переглядають підручники та приклади Qjт з усього світу — <https://www.programiz.com/>
5. Stackify - допомагає розробникам писати кращий код. Їх продукти дозволяють тестувати свій код під час написання. Це допомагає розробникам завчасно виправляти проблеми з продуктивністю та краще перевіряти код — <https://stackify.com/>

Книги

1. Head First патерни проектування, Ерік Фрімен, Елізабет Робсон, Кеті Сьєрра, Берт Бейтс (укр.)
2. Чистий Код, Роберт Мартін (укр.)
3. C#: Learn C# in One Day and Learn It Well. C# for Beginners with Hands-on Project, LCF Publishing, 161 pages (англ.)
4. C# 8.0 and .NET Core 3.0, Mark J. Price (англ.)
5. C# in Depth: тонкощі програмування, Джон Скит (англ.)

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Заняття	Опис
Тема 1. Комп'ютери, програмування та <i>Microsoft.NET</i>	
Лекція 1	Організаційні моменти. Комп'ютерна система. Програмне забезпечення, комп'ютерні програми. Операційна система. Мови програмування. Як комп'ютер обробляє програми. Екосистема <i>Microsoft.NET (.NET)</i> .
Тема 2. Основи виконання програм та мови програмування <i>C#</i>	
Лекція 2	Система загальних типів (CTS). Специфікація загальної мови (CLS). Загальномовне середовище виконання (CLR). Змінні та константи. Типи даних в <i>.NET</i> . Категорії даних: тип-значення та тип-посилання.
ПР 1	Знайомство з середовищем <i>Visual Studio</i> , редактором коду, рішеннями, проектами та їх тестуванням
Тема 3. Основні оператори <i>C#</i>	
Лекція 3	Оператори: арифметичні, присвоювання, логічні, порівняння, рівності, булеві. Структура програми. Простори імен. Оператори вибору. Оператори ітерацій. Масиви. Парадигма ООП.
ПР 2	Математичні операції. Оператори умови та циклу.
ПР 3	Одновимірні масиви. Багатовимірні масиви.
ПР 4	Методи
ПР 5	Класи
Тема 4. Особливості реалізації ООП та взаємодії між класами	
Лекція 4	Особливості реалізації ООП (конструктор; <i>this</i> ; створення об'єкту; статичний клас; деструктор). Взаємодія між класами (асоціація; композиція; агрегація; успадковування). Поліморфізм (під час компіляції; під час виконання; приховування методів).
ПР 6	Наслідування (створення базового класу та класу нащадку, оголошення їх вмісту, робота з методами таких класів)
ПР 7	Асоціація, композиція та агрегація (абстрактний клас, класи-нащадки, абстрактні методи, перевизначення методів)
Тема 5. Обробка винятків. Основи <i>LINQ</i>	
Лекція 5	Винятки та обробка виняткових ситуацій. <i>LINQ</i> (англ. <i>Language Integrated Query</i> - запити, інтегровані в мову). Запити до різних джерел даних або форматів даних.
ПР 8	Винятки та обробка виняткових ситуацій (перевірка аргументів, використання конструкції <i>try...catch</i> , оператор <i>throw</i>).
ПР 9	Виконання запитів даних за допомогою <i>LINQ</i> .
Тема 6. Інтерфейс програмування додатків	
Лекція 6	Створення першої програми з графічним інтерфейсом (форма). Додавання елемента керування до форми. Створення обробників подій. Налаштування розміру та масштабу. Приклади створення програм.
ПР 10	Створення програми з графічним інтерфейсом на основі проекту <i>Windows Forms (.NET)</i> .
ПР 11	Створення бази даних та підключення її до проекту. Налаштування програми для роботи з базою даних. Публікація та перевірка працездатності застосунку.
Тема 7. Створення вебзастосунків за архітектурним шаблоном <i>MVC</i>	
Лекція 7	<i>ASP.NET</i> — технологія створення вебзастосунків і вебсервісів від компанії Майкрософт. Архітектурний шаблон <i>Model-View-Controller (MVC)</i> , що розділяє програму на три основні групи компонентів. Створення додатку <i>ASP.NET MVC</i> , структура папок, додавання компонентів. Передача даних між контролером та представленням.
ПР 12	Створення веб-застосунку за допомогою технології <i>ASP.NET Core</i> та архітектурного шаблону <i>MVC</i> частина 1: користувацький інтерфейс (<i>Front-end</i>) його зв'язок з контролером (<i>Back-end</i>).
ПР 13	Підключення <i>Entity Framework Core</i> . Робота з базою даних у додатку <i>ASP.NET Core MVC</i> .
Тема 8. Принципи та шаблони проектування програмного забезпечення	
Лекція 8	Принципи ООП програмування та дизайну <i>SOLID</i> . Шаблони (патерни) програмування. Рефакторинг.
ПР 14	Створення кросплатформового застосунку <i>MAUI (Multi-platform App UI)</i> для мобільних і настільних пристроїв. Налаштування емулятору та пристрою <i>Android</i> .

6. Самостійна робота студента

1. На протязі семестру:

- Вивчення лекційного матеріалу.
- Опрацювання літературних джерел.
- Відповіді на запитання для самоперевірки.

2. На протязі тижня перед запланованою датою:

- Підготовка до виконання практичних робіт.
- Підготовка до написання контрольної роботи.
- Підготовка до виконання домашньої контрольної роботи.
- Підготовка до заліку.

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Правила відвідування занять:

- для лекцій та практичних - відвідування занять (відеоконференцій Zoom) за розкладом;
- допускається самостійне вивчення матеріалу за допомогою записів лекцій та інших матеріалів розміщених у відповідному дистанційному курсі;
- допускається виконання практичних завдань в асинхронному режимі.

Правила поведінки на заняттях:

- на заняттях необхідно використовувати мережу Інтернет для: виконання завдань в дистанційному курсі; ознайомлення з наведеними посиланнями; доступу до сучасних впорядкованих джерел інформації;
- допускається використання мобільних телефонів, ноутбуків та іншої техніки.

Правила виконання практичних робіт:

- у разі виникнення у викладача запитань до отриманих результатів - необхідно усно пройти процедуру захисту (відповісти на запитання);
- вчасним вважається проходження процедури захисту на занятті присвяченому роботі або на наступному за розкладом.

Правила призначення заохочувальних балів:

- заохочувальні бали призначаються за виконання зазначених в роботах додаткових завдань.

Правила призначення штрафних балів:

- штрафні бали можуть призначатися за невчасну здачу/захист практичних робіт.

Політика дедлайнів та перескладань:

- проходження тестів, контрольних та здача практичних робіт виконується до останнього в семестрі заняття з дисципліни.

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

- Поточний контроль: опитування (тести) за темами лекцій (8 балів),

виконання практичних робіт (75 балів), МКР (10 балів), ДКР (7 балів).

- Календарний контроль: проводиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу.
- Семестровий контроль: залік.
- Умови допуску до семестрового контролю: семестровий рейтинг більше 60 балів.

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою

Кількість балів	Оцінка
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

9. Додаткова інформація з дисципліни (освітнього компонента)

...

Опис матеріально-технічного та інформаційного забезпечення дисципліни

Заняття проводяться на комп'ютерах, розміщених в навчальному класі кафедри ПРЕ. Також студенти можуть використовувати власні комп'ютери. Основним програмним забезпеченням є *Microsoft Visual Studio Community* - це безкоштовне повнофункціональне розширюване інтегроване середовище розробки (*IDE*) для створення сучасних додатків *Android*, *iOS* та *Windows*, а також веб-додатків та хмарних служб.

Робочу програму навчальної дисципліни (силабус):

Складено [Нікітчук А. В.](#);

Ухвалено кафедрою ПРЕ (протокол № 06/2025 від 25.06.2025)

Погоджено методичною комісією факультету/ННІ (протокол № 06/2025 від 26.06.2025)