

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ
УНІВЕРСИТЕТ
СИСТЕМА УПРАВЛІННЯ ЯКІСТЮ
Система забезпечення якості освітньої діяльності та якості вищої освіти
сертифікована на відповідність ДСТУ ISO 9001:2015 / ISO 9001:2015
Кафедра програмної інженерії та кібербезпеки

СИЛАБУС

ТЕХНОЛОГІЯ WPF-ЗАСТОСУВАНЬ /
TECHNOLOGY WPF APPLICATION

SYLLABUS

освітній ступінь	магістр / master
галузь знань	12 Інформаційні технології / Information Technology
спеціальність	121 Інженерія програмного забезпечення / Software Engineering
спеціалізація	Інженерія програмного забезпечення / Software Engineering

Гарант освітньо-
професійної програми
"Інженерія програмного
забезпечення" другого
(магістерського) рівня
вищої освіти

С, Др
Завідувач
кафедри про-
грамної ін-
женерії та
кібербезпеки,
д.т.н., проф.
Криворучко О.В.

Київ 2019

Автори: Котенко Н.О., к. пед. н, ст. викладач
Жирова Т.О., к. пед. н, ст. викладач
Гнатченко Д.Д., асистент

Силабус розглянуто і затверджено на засіданні кафедри програмної інженерії та кібербезпеки 10 вересня 2019 р., протокол № 3..

СИЛАБУС

ТЕХНОЛОГІЯ WPF-ЗАСТОСУВАНЬ / TECHNOLOGY WPF APPLICATION

SYLLABUS

освітній ступінь	магістр / master
галузь знань	12 Інформаційні технології / Information Technology
спеціальність	121 Інженерія програмного забезпечення / Software Engineering
спеціалізація	Інженерія програмного забезпечення / Software Engineering

АНОТАЦІЯ КУРСУ

1. Викладачі:

1.1. **Лектор:** Котенко Наталія Олексіївна

- к.пед.н., старший викладач кафедри програмної інженерії та кібербезпеки;
- педагогічний стаж – 13 років;
- контактний телефон: +38(067)067-45-27-040;
- e-mail: kotenkono@knute.edu.ua
- наукові інтереси: програмування, тестування програмного забезпечення;
- стажування та підвищення кваліфікації:
- додаткова інформація

1.2. **Ассистент лектора:** Жирова Тетяна Олександрівна,

- к.пед.н., старший викладач кафедри програмної інженерії та кібербезпеки;
- педагогічний стаж – 13 років;
- контактний телефон: +38(098)244-09-47;
- e-mail: zhygova@knute.edu.ua
- наукові інтереси: програмування, тестування програмного забезпечення;
- стажування та підвищення кваліфікації:
- додаткова інформація:

2. Дисципліна: «Технологія WPF-застосувань»

- рік навчання: 2
- семестр навчання: 1
- кількість кредитів: 6
- кількість годин за семестр: 180
 - лекційних: 14
 - лабораторних: 28
 - на самостійне опрацювання: 138

3. Час та місце проведення:

- аудиторні заняття - відповідно до розкладу КНТЕУ з врахуванням специфіки дисципліни передбачено аудиторіях: 504, 510а, 514;
- поза аудиторна робота - самостійна робота студента, результат виконання якої висвітлено засобами Office 365;

4. Пререквізити та постреквізити навчальної дисципліни:

- **пререквізити:** «Об'єктно-орієнтоване програмування», «Архітектура та проектування програмного забезпечення», «Архітектура комп'ютера».
- **постреквізити:** компетентності, які будуть корисні при вивченні спеціальних дисциплін, виконанні дипломних робіт та у подальшій професійній діяльності, а саме:

<i>Групи компетентностей</i>	<i>Зміст компетентностей</i>
1. Соціально-особистісні та загально-культурні компетентності	<ul style="list-style-type: none"> ▪ Розуміти різноманітність процесів формування й розвитку проекту по створенню програмного продукту у сучасному світі; ▪ Приймати етичні рішення, посилаючись на загальні етичні принципи і на етичний кодекс інженера з програмного забезпечення;
2. Загальнонаукові (інтелектуальні) компетентності	<ul style="list-style-type: none"> ▪ Знати і розуміти специфіку використання технології WPF;
3. Універсальні професійні компетентності	<ul style="list-style-type: none"> ▪ Здатність до абстрактного мислення, аналізу та синтезу; ▪ Здатність проведення теоретичних та прикладних досліджень на відповідному рівні; ▪ Креативність, здатність до системного мислення; ▪ Адаптивність і комунікабельність. Вміння надати допомогу та консультації працівникам та населенню з практичних питань безпеки життєдіяльності та захисту у надзвичайних ситуаціях;
4. Спеціалізовані професійні компетентності (професійно-функціональні знання та вміння)	<ul style="list-style-type: none"> ▪ Знати: <ul style="list-style-type: none"> - загальні принципи побудови інтерфейсу комп'ютерних програм для користувача; - мову розмітки XAML для створення динамічних інтерфейсів; - принципи побудови користувацького інтерфейсу програм засобами WPF; - принципи взаємодії прикладних програм з реляційними системами управління базами даних; - структуру WPF-документів і засобів їхнього захисту від несанкціонованої зміни; ▪ Вміти: <ul style="list-style-type: none"> - створювати WPF-програми; - розробляти інтерфейс користувача для WPF-програм економічного спрямування; - використовувати мову XAML для проектування інтерфейсу комп'ютерних програм; - налаштовувати і модифікувати зовнішній вигляд WPF-програми; - розробляти інтерфейс користувача комп'ютерних програм з використанням різноманітних WPF-моделей макетів сторінок;

<i>Групи компетентностей</i>	<i>Зміст компетентностей</i>
	<ul style="list-style-type: none"> - відображати та змінювати дані, що зберігаються в базах даних; - створювати нові елементи керування для WPF-програми; - керувати документами в WPF-додатках; - забезпечувати захист документів від несанкціонованої зміни; - додавати графічну і мультимедіа підтримку в WPF-додатках.

5. Характеристика дисципліни:

- 5.1. Призначення навчальної дисципліни:** Багато років .NET розробники створювали настільні прикладні програми, використовуючи технологію Windows Forms. Windows Forms – зріла і повнофункціональна технологія, здатна виконати основні завдання сучасних бізнес програм. Але робота технології жорстко пов'язана на архітектурі операційної системи і деякі завдання, які повинні виконуватися швидко і без зайвого програмування, вимагають великих витрат часу і зусиль. Windows Presentation Foundation (WPF) змінила світ програмування настільних прикладних програм. Поклавши в основу технологію DirectX, Microsoft надає можливість розробникам швидко створювати складні елементи управління і повністю керувати процесом візуалізації. WPF комбінує кращі аспекти традиційної розробки для Windows, з безліччю нововведень, дозволяючи будувати насичені графікою інтерфейси для користувача. WPF дає можливість природно поєднувати в одній програмі традиційні інтерфейси, тривимірну графіку, аудіо і відео, анімацію, динамічну зміну обкладинок, мультисенсорне введення, форматування документів і розпізнавання мови.
- 5.2. Мета вивчення дисципліни:** формування у майбутніх фахівців необхідного рівня знань щодо застосування сучасних мов та засобів програмування для виготовлення настільних прикладних програм, що забезпечить конкурентоспроможність випускників університету на ринку праці і сприятиме успішній роботі в різних областях сучасного бізнесу.
- 5.3. Задачі вивчення дисципліни:** є теоретична та практична підготовка майбутніх фахівців з таких питань:
- Створення застосунків з використанням технології WPF.
 - Розробка користувальницького інтерфейсу.
 - Налаштування та модифікація зовнішнього вигляду програми.
 - Прив'язка даних (Data Binding).
 - Колекції як джерело даних.

- Створення нових елементів керування.
- Керування документами.
- Графіка і мультимедіа.

5.4. Зміст навчальної дисципліни: відповідає освітній програмі, навчальному плану, а також запитам стейкхолдерів.

5.5. План вивчення дисципліни

Результати навчання	Навчальна діяльність	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
Тема 1. Вступ до технології WPF		12	5
Знати і розуміти специфіку технології WPF. Знати основні етапи становлення технології WPF. Мати цілісне уявлення про сучасний стан використання технології WPF, її особливості.	Лекція 1. Вступ до технології WPF. <i>План лекції:</i> 1. Предмет, завдання і структура дисципліни. Зв'язок дисципліни з іншими дисциплінами. 2. Історія розвитку технології WPF. Погляд у минуле. 3. Чому не Silverlight? Список рекомендованих джерел: Основний: 1, 2, 4. Додатковий: 5, 6, 8 Інтернет-ресурси: 9, 10, 17, 19	2	
Знати і розуміти історичні передумови виникнення технології WPF та її еволюцію. Знати відмінності між технологіями WPF та Silverlight.	Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем. Питання винесені на самостійне опрацювання: 1. Виникнення WPF. 2. Еволюція WPF. 3. Переваги та особливості WPF. 4. Вибір вікон або сторінок. 5. Обробка подій елементів керування WPF. 6. Модель навігації в WPF. 7. Навігації за гіперпосиланнями. 8. Служба переходів.	10	5
SoftSkills: творче мислення та сприймання конструктивно і критики.			

1	2	3	4
Тема 2. Введення в мову розмітки XAML		30	15
Знати визначення XAML. Мати цілісне уявлення про сучасний стан та сфери використання мови розмітки XAML. Знати нюанси використання елементів та атрибутів;	Лекція 2. Введення в мову розмітки XAML <i>План лекції:</i> 1. Означення XAML. 2. Елементи і атрибути. 3. Базові типи і простір імен. 4. Елементи властивостей. 5. Властивості і розширення розмітки XAML. 6. Дочірні об'єктні елементи. 7. Диспетчери компонування. 8. Компіляція XAML. Список рекомендованих джерел: Основний: 1, 3, 5. Додатковий: 6, 7. Інтернет-ресурси: 9, 14, 17, 19.	2	
базові типи і простір імен; ключові слова XAML. Вміти одні і ті ж задачі розв'язувати з допомогою XAML та C# Розуміти особливості використання класів зчитування та запису XAML. <i>SoftSkills:</i> управління часом, знання іноземної мови, робота в команді.	Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем. Питання винесені на самостійне опрацювання: 1. Поєднання XAML і процедурного коду. 2. Клас.XamlReader 3. Перетворення типів. 4. Створення об'єктів з допомогою конструктора з аргументами. 5. Створення екземплярів з допомогою фабричних методів. 6. Трюки з класами зчитування і запису XAML. 7. Ключові слова XAML.	24	5
	Лабораторні заняття №1 «Переведення прикладів XAML на C# та навпаки». <i>Мета:</i> Засвоїти синтаксис XAML та навчитися одні і ті ж задачі вирішувати з допомогою XAML та C #. <i>Завдання:</i> 1. Запропонований код на C# подати у вигляді XAML. 2. Запропонований XAML код подати у вигляді C# коду. 3. Виконати завдання відповідно до свого варіанту з допомогою XAML та C #. 4. Створити динамічну заставку засобами мови XAML, тематику обрати на власний розсуд. <i>План заняття:</i> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Захист виконаної роботи. -	4	10

1	2	3	4
Тема 3. Розробка інтерфейсу користувача		30	15
Знати і розуміти специфіку розробки інтерфейсу користувача на основі технології WPF. Вміти: розташовувати елементи керування на сторінці; розробляти інтерфейс з використанням елементів керування Content Controls; розробляти інтерфейс з використанням елементів управління Items Controls; використовувати елементи керування Windows Forms в застосуванні WPF. <i>SoftSkills:</i> комунікативні навички, тайм менеджмент, критичне мислення.	Лекція №3 Розробка інтерфейсу користувача <i>План лекції:</i> 1. Розташування елементів керування на сторінці. 2. Розробка інтерфейсу з використанням елементів керування Content Controls. 3. Розробка інтерфейсу з використанням елементів управління Items Controls. 4. Використання елементів керування Windows Forms в застосуванні WPF. Список рекомендованих джерел: Основний: 1, 2, 3. Додатковий: 6, 8. Інтернет-ресурси: 13 - 19.	2	
	Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем. Питання винесені на самостійне опрацювання: 1. Класи макета в WPF (Canvas, DockPanel, Grid, StackPanel, VirtualizingStackPanel, WrapPanel). 2. Моделі вмісту. 3. Класи Headered Content Controls. 4. Загальна характеристика Items Controls. 5. Класи Items Controls. 6. Обробка подій. 7. Обробка подій Item Selection. 8. Причини використання елементів Windows Forms в WPF. 9. Посилання на елементи Windows Forms у застосуванні WPF. 10. Використання елементів Windows Forms у XAML. 11. Взаємодія з елементами Windows Forms.	24	5
	Лабораторні заняття №2 «Розробка інтерфейсу користувача» <i>Мета:</i> Навчитися створювати інтерфейс користувача засобами WPF. <i>Завдання:</i> Оволодіти технологією створення інтерфейсу користувача засобами WPF відповідно до вказаної тематик. <i>План заняття:</i> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Захист виконаної роботи. -	4	10

1	2	3	4
	Тема 4. Налаштування та модифікація зовнішнього вигляду програми	26	15
Знати основні нюанси налаштування та модифікації зовнішнього вигляду програми. Вміти створювати цілісний інтерфейс користувача з використанням стилів;	<p>Лекція 4 Налаштування та модифікація зовнішнього вигляду програми</p> <p><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Створення цілісного інтерфейсу користувача з використанням стилів. 2. Зміна зовнішнього вигляду елементів керування шляхом модифікації його шаблону. 3. Поліпшення призначеного для користувача інтерфейсу за рахунок анімації. <p>Список рекомендованих джерел: Основний: 1, 2, 4. Додатковий: 6-8. Інтернет-ресурси: 15 - 20.</p>	2	
змінювати зовнішній вигляд елементів керування шляхом модифікації його шаблону; використовувати засоби анімації. SoftSkills: комунікативні навички, робота в команді, творчі навички, сприйняття конструктивно і критики. SoftSkills: комунікативні навички, робота в команді, творчі навички, сприйняття конструктивно і критики,	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання:</p> <ol style="list-style-type: none"> 1. Призначення ресурсів. 2. Опис ресурсів. 3. Посилання на ресурси в XAML. 4. Посилання на ресурси в програмі. 5. Повторне використання ресурсів у застосуваннях. 6. Призначення стилів. 7. Опис стилів. 8. Розширення стилів. 9. Програмна установка стилів. 10. Призначення шаблонів елементів керування. 11. Опис шаблону для Content Control. 12. Опис шаблону для Items Control. 13. Забезпечення налаштувань користувача при використанні прив'язки шаблонів. 14. Призначення тригерів. 15. Опис тригера властивостей. 16. Призначення анімації. 17. Опис анімації. 	20	5
колективний тайм менеджмент, знання іноземної	<p>Лабораторні заняття №3 «Налаштування та модифікація зовнішнього вигляду WPF програми»</p> <p><i>Мета:</i> Засвоїти методи налаштування та модифікації зовнішнього вигляду WPF програми.</p> <p><i>Завдання:</i> Модифікувати інтерфейс користувача, створений при виконанні попередньої</p>	4	10

1	2	3	4
мови.	лабораторної роботи, засобами WPF відповідно до зауважень та рекомендацій викладача. Обов'язково скористатися засобами анімації. <i>План заняття:</i> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Захист виконаної роботи.		
Тема 5. Створення нових елементів керування.		26	15
Знати основні принципи створення нових елементів керування. Уміти розробляти та створювати елементи керування для поліпшення інтерфейсу користувача. Уміти редагувати створені компоненти.	Лекція 5. Створення нових елементів керування <i>План лекції:</i> 1. Огляд основних етапів розробки елементів керування. 2. Створення елементів керування користувача: а. Створення інтерфейсу елемента керування користувача; б. Наділення елемента керування користувача поведінкою; в. Додавання до елементів керування користувача властивостей залежності; г. Додавання до елементів керування користувача подій, що маршрутизуються. Список рекомендованих джерел: Основний: 1, 2, 4 Додатковий: 7, 8. Інтернет-ресурси: 9 – 12, 18.	2	
<i>SoftSkills:</i> комунікативні навички, робота в команді, творчі навички, сприйняття конструктивно і критики, колективний тайм менеджмент, знання іноземної мови	Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем. Питання винесені на самостійне опрацювання: 1. Призначення нових елементів керування. 2. Засоби створення нових елементів керування. 3. Елементи керування користувача. 4. FrameworkElement-похідні елементів керування. 5. Реалізація властивостей і подій. 6. Створення користувацьких елементів управління. 7. Реалізація команд. 8. Удосконалення елементів керування за допомогою тем.	20	5

1	2	3	4
	<p>Лабораторні заняття №4 «Удосконалення інтерфейсів користувача за допомогою створення нових елементів керування»</p> <p><i>Мета:</i> Навчитися створювати нові елементи керування та з їх допомогою вдосконалити користувацькі інтерфейси.</p> <p><i>Завдання:</i> Удосконалити інтерфейс користувача, розроблений під час виконання попередньої лабораторної роботи, за допомогою створення нових елементів керування.</p> <p><i>План заняття:</i></p> <ul style="list-style-type: none"> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Захист виконаної роботи. 	4	5
	Контрольна робота		5
Тема 6. Прив'язка даних (Data Binding)		36	15
<p>Знати механізми прив'язки даних та модель прив'язки даних у WPF. Вміти створювати прив'язку даних та використовувати механізм відстежування змін властивостей. Знати принципи прив'язки елементів керування до колекції об'єктів; відображення даних з використанням шаблонів даних та подань. Уміти здійснювати навігацію за записами.</p> <p><i>SoftSkills:</i> комунікативні</p>	<p>Лекція №6. Прив'язка даних (Data Binding)</p> <p><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Огляд механізмів прив'язки даних. Модель прив'язки даних у WPF. 2. Створення прив'язки даних. 3. Використання механізму відстежування зміни властивостей. 4. Перетворення даних 5. Прив'язка елементів керування до колекції об'єктів. 6. Відображення даних з використанням шаблонів даних. 7. Відображення даних з використанням подань. 8. Навігація за записами. <p>Список рекомендованих джерел: Основний: 1, 2, 5. Додатковий: 6, 8. Інтернет-ресурси: 9 – 20</p>	2	
	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем.</p> <p>Питання винесені на самостійне опрацювання:</p> <ol style="list-style-type: none"> 1. Модель прив'язки даних у WPF. 2. Джерела і цілі. 3. Режими прив'язки даних. 4. Прив'язка до властивості класу. 5. Прив'язка множини елементів до класу. 6. Прив'язка до всього об'єкта. 7. Прив'язка до інших елементів користувацького інтерфейсу. 8. Прив'язка до даних XML. 	28	5

1	2	3	4
<p>навички, робота в команді, творчі навички, сприйняття конструктивно і критики, колективний тайм менеджмент, знання іноземної мови</p>	<ol style="list-style-type: none"> 9. Призначення властивості оповіщення про зміни. 10. Передача оповіщення про зміни до мети прив'язки. 11. Передача змінених значень до джерела прив'язки. 12. Перетворення даних за замовчуванням. 13. Використання перетворювача значень (Custom Value Converter). 14. Огляд прив'язок до колекцій. 15. Поняття колекції перегляду. 16. Визначення класу колекції перегляду. 17. Прив'язка до об'єктів ADO.NET. 18. Поняття шаблону даних. 19. Опис і використання шаблону даних. 20. Опис шаблону даних як ресурсу. 21. Використання тригерів, даних у шаблоні даних. 22. Поняття подання. 23. Створення і використання подань. 24. Сортування даних. 25. Фільтрація даних. 26. Групування даних. 27. Використання списку. 28. Використання колекції елементів. 29. Створення інтерфейсу Master-Detail. 30. Відношення Master-Detail в ADO.NET. 31. Організація обчислень. 		
	<p>Лабораторні заняття №5 «Прив'язка елементів інтерфейсу користувача до колекцій» <i>Мета:</i> Навчитися здійснювати прив'язку елементів інтерфейсу користувача. <i>Завдання:</i> Здійснити прив'язку елементів інтерфейсу користувача до колекції. Використати розроблений під час виконання попередньої лабораторної роботи інтерфейс користувача. <i>План заняття:</i></p> <ul style="list-style-type: none"> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Захист виконаної роботи. 	6	10
Тема 7. Керування документами		28	20
<p>Знати структуру WPF-документів і засобів їхнього захисту від несанкціонованої зміни. Вміти керувати</p>	<p>Лекція №7. Керування документами <i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Перевірка даних за замовчуванням. 2. Візуалізація результату перевірки. 3. Визначення користувальницького правила перевірки. 4. Опис правила перевірки з використанням XAML. Створення і перегляд змінюваних документів. 	2	

1	2	3	4
<p>документами в WPF-застосуваннях; забезпечувати захист документів від несанкціонованої зміни шляхом додавання електронного підпису; додавати графічну і мультимедіальну підтримку в WPF-застосуваннях; керувати процесом друку документів на основі програмних засобів.</p>	<p>5. Створення і перегляд фіксованих документів. 6. Стиснення (упаковка) документів. 7. Цифровий підпис змісту. 8. Друкування документів. Список рекомендованих джерел: Основний: 1, 2, 3 Додатковий: 6. Інтернет-ресурси: 11 – 20.</p>		
<p><i>SoftSkills:</i> комунікативні навички, робота в команді, творчі навички, сприйняття конструктивно і критики, колективний тайм менеджмент, знання іноземної мови</p>	<p>Завдання для самостійної роботи: Вивчення та доповнення матеріалів лекції на основі самостійного опрацювання основних літературних джерел, зазначених у списку та електронних матеріалів, які надаються викладачем. Питання винесені на самостійне опрацювання: 1. Перевірка даних за замовчуванням. 2. Візуалізація результату перевірки. 3. Визначення користувачького правила перевірки. 4. Опис правила перевірки з використанням XAML. 5. Поняття змінюваного документа (Flow Document). 6. Опис змінюваного документа. 7. Типи контейнерів для змінюваного документа. 8. Елементи у змінюваному документі. 9. Налаштування тексту. 10. Встановлення об'єкта FlowDocumentReader. 11. Перевірка функції масштабу і режимів. 12. Заміна FlowDocumentReader на RichTextBox. 13. Зміна форматування тексту в RichTextBox. 14. Зміна властивості TextDecorations в елементі Paragraph. 15. Зміна установки Typography елемента Paragraph. 16. Поняття фіксованого документа. 17. Опис фіксованого документа FixedDocument. 18. Опис переглядача фіксованих документів FixedDocument Viewer. 19. Підтримка стиснення документів. 20. Упаковка частин документів в ZIP-файл. 21. Цифровий підпис змісту. 22. Пов'язана інформація з пакетами або частинами. 23. Створення підписаного ZIP-пакета. 24. Поняття XML Paper Specification. 25. Керування завданнями друку. 26. Керування чергою друку.</p>	20	5

1	2	3	4
	<p>Лабораторні заняття №6 «Перевірка даних на інтерфейсному рівні»</p> <p><i>Мета:</i> навчитися забезпечувати захист документів від несанкціонованої зміни.</p> <p><i>Завдання:</i> Модифікувати створений на попередній лабораторній роботі інтерфейс користувача з метою забезпечити захист документів від несанкціонованої зміни шляхом додавання електронного підпису та інших методів захисту. Додавати графічну і мультимедіапідтримку. Додати можливість керувати процесом друку документів.</p> <p><i>План заняття:</i></p> <ul style="list-style-type: none"> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Захист виконаної роботи. 	6	10
	Контрольна робота		5
Разом		180 годин/ 6 кредитів	100 балів
Підсумковий контроль		Письмовий екзамен	

6. Список рекомендованих джерел

Основний

1. Yuen S. Mastering Windows Presentation Foundation: Master the art of building modern desktop applications on Windows Paperback – February 17, 2017. – 568 p.
2. Stephens R. WPF 3d: Three-Dimensional Graphics with WPF and C# Paperback – February 8, 2018. – 426 p.
3. Chowdhury K. Windows Presentation Foundation Development Cookbook: 100 recipes to build rich desktop client applications on Windows Paperback – February 23, 2018. - 524 p.
4. Nathan A. WPF 4 Unleashed / A. Nathan. – Sams, 2010. – 848 p.
5. MacDonald M. Pro WPF 4.5 in C#: Windows Presentation Foundation in .NET 4.5 4th ed. – Apress; 4th ed. edition (November 27, 2012). – 1078 p.

Додатковий

6. Troelsen A. Pro C# 7: With .NET and .NET Core. Apress; 8th ed. edition (November 21, 2017). – 1372 p.
7. Stephens R. WPF Programmer's Reference: Windows Presentation Foundation with C# 2010 and .NET 4 – Wrox, / R. Stephens 2010. – 624 p.
8. Solis D. Illustrated WPF (Expert's Voice in .Net). / D. Solis – Apress, 2009. – 507 p.

Інтернет-ресурси

9. Руководство по WPF. Режим доступу: <https://metanit.com/sharp/wpf/>
10. Walkthrough: My first WPF desktop application. Режим доступу: <https://docs.microsoft.com/en-us/dotnet/framework/wpf/getting-started/walkthrough-my-first-wpf-desktop-application>
11. GitHub. Режим доступу: <https://github.com/dotnet/wpf>

- 12.Руководство по WPF. Режим доступу: <https://metanit.com/sharp/wpf/>
- 13.Windows Presentation Foundation. Режим доступу: http://uk.wikipedia.org/wiki/Windows_Presentation_Foundation.
- 14.Windows Presentation Foundation (WPF) in Visual Studio. Режим доступу: <http://windowsclient.net/wpf/default.aspx>
- 15.Tutorial: Create your first WPF application in Visual Studio 2019. Режим доступу: <https://docs.microsoft.com/en-us/dotnet/framework/wpf/getting-started/walkthrough-my-first-wpf-desktop-application>
- 16.Wpftutorial. Режим доступу: <http://www.wpftutorial.net>.
- 17>Welcome to Windows Presentation Foundation! Режим доступу: <http://wpf.codeplex.com>.
- 18.Windows Presentation Foundation. Режим доступу: <https://github.com/dotnet/wpf>.
- 19.WPF Tutorial. Режим доступу: <https://www.tutorialspoint.com/wpf/index.htm>.
- 20.WPF Tutorial for Beginners: Learn in 1 Day. Режим доступу: <https://www.guru99.com/wpf-tutorial.html>

7. Контроль та оцінювання результатів навчання: положення про оцінювання результатів навчання студентів і аспірантів наказ КНТЕУ №2891 від 16.09.2019р.

Під час вивчення дисципліни «Проектування мультимедійних систем» викладачем здійснюється поточний та підсумковий контроль. Поточний контроль та оцінювання передбачає:

- перевірку рівня засвоєння теоретичного матеріалу (тестування за матеріалами лекції, який здійснюється на початку кожної наступної лекції з використанням 365 Office);
- захист лабораторних робіт (проходить під час наступної лабораторної роботи);
- перевірка засвоєння матеріалу, що винесений на самостійне опрацювання під час фронтального опитування на лекції.

8. Політика навчальної дисципліни:

8.1. Відвідування лекційних та лабораторних занять: відвідування лекційних та лабораторних занять є обов'язковим. Допускаються пропуски занять з таких поважних причин, як хвороба (викладачу надається копія довідки від медичного закладу), участь в олімпіаді, творчому конкурсі тощо за попередньою домовленістю та згодою викладача за умови дозволу деканату (надаються документи чи інші матеріали, які підтверджують заявлену участь у діяльності студента).

8.2. Відпрацювання пропущених занять: відпрацювання пропущених занять є обов'язковим незалежно від причини пропущеного заняття. Лекційне заняття має бути відпрацьоване до наступної лекції на консультації викладача з використанням ПЗ 365 Office Teams. Відпрацювання лекційного матеріалу передбачає вивчення пропущеного теоретичного матеріалу та складання тесту за цим

матеріалом. Лабораторне заняття відпрацьовується під час консультації викладача (розклад консультацій на сайті).

- 8.3. Правила поведінки під час занять:** обов'язковим є дотримання техніки безпеки в комп'ютерних лабораторіях. Студенти повинні приймати активну участь в обговоренні навчально матеріалу ознайомившись з ним напередодні (навчальний матеріал надається викладачем).
- 8.4. За порушення академічної доброчесності** студенти будуть притягнені до такої академічної відповідальності:
- повторне проходження оцінювання (контрольна робота, іспит, залік тощо);
 - повторне проходження навчального курсу.