

**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

ЗАТВЕРДЖЕНО

вченою радою
пост. 10 п. 10 від “21” червня 2018 р.)

Ректор



[Signature]
А.А. Мазаракі

**ТЕХНОЛОГІЯ WPF-ЗАСТОСУВАНЬ /
TECHNOLOGY WPF APPLICATION**

**ПРОГРАМА ТА РОБОЧА ПРОГРАМА/
CURRICULUM AND 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

Київ 2018

Розповсюдження і тиражування без офіційного дозволу КНТЕУ заборонено

Автори: Котенко Н. О., канд. пед. наук, ст. викл.,
Жирова Т. О., канд. пед. наук, ст. викл.,
Гнатченко Д.Д., асистент

Програму та робочу програму розглянуто і затверджено на засіданні кафедри програмної інженерії та інформаційних систем 15 травня 2018 р., протокол №26.

Рецензенти: М.О. Цензура, канд. техн. наук, доц.,
О.О. Руденко, Front-end Team Lead at Astound Commerce

ТЕХНОЛОГІЯ WPF-ЗАСТОСУВАНЬ / TECHNOLOGY WPF APPLICATION

ПРОГРАМА ТА РОБОЧА ПРОГРАМА/ CURRICULUM AND 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

Автори: Котенко Н. О., канд. пед. наук, ст. викл.,
Жирова Т. О., канд. пед. наук, ст. викл.,
Гнатченко Д.Д., асистент

Редактор
Комп'ютерна верстка

Підп. до друку _____, Формат 60x84/16. Папір письм.
Ризографія. Ум. друк. арк. . Ум. фарбо-відб. .
Обл.-вид. арк. . Тираж пр. Зам. .

Центр підготовки навчально-методичних видань КНТЕУ
02156, Київ-156, вул. Кіото, 19

ВСТУП

Програма та робоча програма дисципліни «Технологія WPF-застосувань» призначена для підготовки студентів денної форми навчання за освітнім ступенем «магістр» спеціальності 121 «Інженерія програмного забезпечення».

Програму та робочу програму підготовлено відповідно до освітньо-професійної програми підготовки магістрів, структурно-логічної схеми освітнього процесу та вимог освітньо-професійної програми спеціальності 121 «Інженерія програмного забезпечення».

Навчальна дисципліна належить до групи вибіркових дисциплін і забезпечує підготовку студентів до майбутньої професійної діяльності в умовах сучасного розвитку засобів мультимедіа.

Згідно з навчальним планом КНТЕУ дисципліна «Технологія WPF-застосувань» викладається студентам другого курсу магістратури.

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.

Програма та робоча програма складається з наступних частин:

1. Мета, завдання та результати вивчення дисципліни (компетентності), її місце в освітньому процесі.
2. Зміст дисципліни.
3. Структура дисципліни та розподіл годин за темами (тематичний план).
4. Тематика та зміст лекційних та лабораторних занять, самостійної роботи студентів.
5. Список рекомендованих джерел.

1. МЕТА, ЗАВДАННЯ ТА РЕЗУЛЬТАТИ ВИВЧЕННЯ (КОМПЕТЕНТНОСТІ) ДИСЦИПЛІНИ, ЇЇ МІСЦЕ В ОСВІТНЬОМУ ПРОЦЕСІ

Метою вивчення дисципліни «Технологія WPF-застосувань» є 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.

Завданням дисципліни «Технологія WPF-застосувань» є теоретична практична підготовка 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.

Предметом дисципліни «Технологія WPF-застосувань» безпосередньо є технологія Windows Presentation Foundation як основа інтерфейсу користувача комп'ютерних програмних засобів.

Компетентності зазначеної дисципліни – отримати певну систему знань, на основі якої фахівець зможе 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; 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.

Місце в освітньому процесі. Дисципліна базується на знаннях, що студенти отримали під час вивчення дисциплін «Об'єктно-орієнтоване програмування», «Архітектура та проектування програмного забезпечення», «Архітектура комп'ютера». Дисципліна надає студентам необхідні знання та навички, які будуть корисні при вивченні спеціальних дисциплін, виконанні дипломних робіт, у подальшій професійній діяльності.

Дисципліна вивчається у 3 семестрі. Вивчення дисципліни розраховано на 180 години (6 кредитів) і передбачає використання наступних видів занять: лекцій – 14 год., лабораторних робіт – 28 год., самостійної роботи – 138 год.

Всі лабораторні завдання виконуються на основі інтерактивних методів навчання у комп'ютерному середовищі.

2. ЗМІСТ ДИСЦИПЛІНИ

ТЕМА 1. Вступ до технології WPF

Вступ. Мета та задача дисципліни, її місце в освітньому процесі.

Поняття інтерфейсу. Апаратний, програмний, інтерфейс користувача. Види інтерфейсів користувача. Ключові елементи інтерфейсу користувача. Психологічні аспекти інтерфейсу користувача. Основні завдання при розробці інтерфейсів користувача.

Історія розвитку технології WPF. Погляд у минуле. Що таке Silverlight. Спільне та відмінне між WPF і Silverlight. Чому не Silverlight?

Призначення та основні особливості WPF. Визначення інтерфейсу користувача в WPF. Мова XAML. Переваги поділу зовнішнього вигляду та поведінки. Переваги та особливості WPF. Типи WPF-застосувань. Створення найпростішої WPF-програми. Порядок створення WPF-програми в Visual Studio. Визначення в застосуванні. Вибір вікон або сторінок. Додавання елементів керування. Побудова та виконання WPF-програми. Обробка подій. Модель подій в WPF. Обробка подій елементів керування WPF. Навігація між сторінками. Модель навігації в WPF. Навігації за гіперпосиланнями. Служба переходів.

Список рекомендованих джерел:

Основний: 1, 2, 3

Додатковий: 4, 7, 8

Інтернет-ресурси: 9 - 13

ТЕМА 2. Введення в мову розмітки XAML

Введення в мову розмітки XAML. Опис способів завантаження та компіляції мови XAML. Правила формування розмітки інтерфейсу програми. Елементи і атрибути. Базові типи і простір імен. Елементи властивостей. Формування розмітки і іменування елементів. Властивості і розширення розмітки XAML. Дочірні об'єктні елементи. Поєднання XAML і процедурного коду. Диспетчери компонування. Трюки з класами зчитування і запису XAML. Ключові слова XAML.

Список рекомендованих джерел:

Основний: 1, 2, 3

Додатковий: 6 - 8

Інтернет-ресурси: 15 - 20

ТЕМА 3. Розробка інтерфейсу користувача

Розташування елементів керування на сторінці. WPF-моделі макета сторінок. Класи макета в WPF (Canvas, DockPanel, Grid, StackPanel, VirtualizingStackPanel, WrapPanel). Розробка інтерфейсу з використанням елементів керування Content Controls. Моделі вмісту. Класи Headered Content Controls. Розробка інтерфейсу з використанням елементів управління Items Controls. Загальна характеристика Items Controls. Класи Items Controls. Обробка подій. Обробка подій Item Selection. Використання елементів керування Windows Forms у застосуванні WPF. Причини використання елементів Windows Forms в

WPF. Посилання на елементи Windows Forms у застосуванні WPF. Використання елементів Windows Forms в XAML. Взаємодія з елементами Windows Forms.

Список рекомендованих джерел:

Основний: 1, 2, 3, 4

Додатковий: 6

Інтернет-ресурси: 18 - 20

ТЕМА 4. Налаштування та модифікація зовнішнього вигляду програми

Спільне використання логічних ресурсів у застосуванні. Призначення ресурсів. Опис ресурсів. Посилання на ресурси в XAML. Посилання на ресурси в програмі. Повторне використання ресурсів у застосуваннях. Створення цілісного інтерфейсу користувача з використанням стилів. Призначення стилів. Опис стилів. Розширення стилів. Програмна установка стилів. Зміна зовнішнього вигляду елементів керування шляхом модифікації його шаблону. Призначення шаблонів елементів керування. Опис шаблону для Content Control. Опис шаблону для Items Control. Забезпечення налаштувань користувача при використанні прив'язки шаблонів. Поліпшення призначеного для користувача інтерфейсу за рахунок анімації. Призначення тригерів. Опис тригера властивостей. Призначення анімації. Опис анімації.

Список рекомендованих джерел:

Основний: 1, 2, 3, 4, 5

Додатковий: 7

Інтернет-ресурси: 9 - 13

ТЕМА 5. Створення нових елементів керування

Огляд розробки елементів керування та їх призначення. Засоби створення нових елементів керування. Користувальницькі елементи керування. FrameworkElement-похідні елементів керування. Створення елементів керування. Реалізація властивостей і подій. Створення користувацьких елементів управління. Реалізація команд. Удосконалення елементів керування за допомогою тем.

Список рекомендованих джерел:

Основний: 1, 2, 3, 4

Додатковий: 6 - 8

Інтернет-ресурси: 14 - 20

ТЕМА 6. Прив'язка даних (Data Binding)

Огляд механізмів прив'язки даних. Модель прив'язки даних у WPF. Джерела і цілі. Режими прив'язки даних. Створення прив'язки даних. Прив'язка до властивості класу. Прив'язка множини елементів до класу. Прив'язка до всього об'єкта. Прив'язка до інших елементів користувацького інтерфейсу. Прив'язка до даних XML. Використання механізму відстеження зміни властивостей. Призначення властивості оповіщення про зміни. Передача оповіщення про зміни до мети прив'язки. Передача змінених значень до джерела прив'язки. Перетворення даних. Перетворення даних за замовчуванням.

Використання користувальницького перетворювача значень (Custom Value Converter).

Прив'язка елементів керування до колекції об'єктів. Огляд прив'язок до колекцій. Поняття колекції перегляду. Визначення класу колекції перегляду. Прив'язка до об'єктів ADO.NET. Відображення даних з використанням шаблонів даних. Поняття шаблону даних. Опис і використання шаблону даних. Опис шаблону даних як ресурсу. Використання тригерів, даних у шаблоні даних. Поняття подання. Створення і використання подань. Сортування даних. Фільтрація даних. Групування даних. Навігація за записами. Використання списку. Використання колекції елементів. Створення інтерфейсу Master-Detail. Відношення MasterDetail в ADO.NET. Організація обчислень.

Список рекомендованих джерел:

Основний: 1, 2, 5

Додатковий: 7

Інтернет-ресурси: 10 - 13

ТЕМА 7. Керування документами

Перевірка даних за замовчуванням. Візуалізація результату перевірки. Визначення користувачького правила перевірки. Опис правила перевірки з використанням XAML.

Створення і перегляд змінюваних документів. Поняття змінюваного документа (Flow Document). Опис змінюваного документа. Типи контейнерів для змінюваного документа. Елементи у змінюваному документі. Налаштування тексту. Встановлення об'єкта FlowDocumentReader. Перевірка функції масштабу і режимів. Заміна FlowDocumentReader на RichTextBox. Зміна форматування тексту в RichTextBox. Зміна властивості TextDecorations в елементі Paragraph. Зміна установки Typography елемента Paragraph. Створення і перегляд фіксованих документів. Опис фіксованого документа FixedDocument. Опис переглядача фіксованих документів FixedDocumentViewer.

Стиснення (упаковка) документів. Підтримка стиснення документів. Упаковка частин документів у ZIP-файл. Цифровий підпис змісту. Пов'язана інформація з пакетами або частинами. Створення підписаного ZIP-пакета.

Друкування документів. Поняття XML Paper Specification. Керування завданнями друку. Керування чергою друку.

Список рекомендованих джерел:

Основний: 1 - 5

Додатковий: 6 - 8

Інтернет-ресурси: 19 - 20

3. Структура дисципліни та розподіл годин за темами (тематичний план)

Назва теми	Кількість годин				Форми контролю*
	Всього годин / кредитів	з них			
		Лекції	Лабораторні заняття/мод. контроль	Самостійна робота	
1	2	4	5	7	8
Тема 1. Вступ до технології WPF	12	2	0	10	К, Т, ПСР
Тема 2. Введення в мову розмітки XAML	30	2	4	24	К, Т, ПСР, ПЗ
Тема 3. Розробка інтерфейсу користувача.	22	2	4	16	К, Т, ПСР, ПЗ
Тема 4. Налаштування та модифікація зовнішнього вигляду програми	26	2	4	20	К, Т, ПСР, ПЗ
Тема 5. Створення нових елементів керування	26	2	4	20	К, Т, КР, ПСР, ПЗ
Тема 6. Прив'язка даних (Data Binding)	36	2	6	28	К, Т, ПСР, ПЗ,
Тема 7. Керування документами	28	2	6	20	К, Т, КР, ПСР, ПЗ
Разом	180/6	14	28	138	
Підсумковий контроль – Екзамен письмовий					

*К – перевірка конспекту; ПСР – перевірка самостійної роботи; ПЗ – перевірка індивідуальних завдань; Т – тестування на ПЕОМ, КР – контрольна робота.

4. Тематика та зміст лекцій, лабораторних занять, самостійної роботи студентів

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. Знати і розуміти історичні передумови виникнення технології WPF та її еволюцію. Знати відмінності між технологіями WPF та Silverlight.	<p style="text-align: center;">Лекція 1. Вступ до технології WPF.</p> <p><i>План лекції:</i></p> <ol style="list-style-type: none"> Предмет, завдання і структура дисципліни. Зв'язок дисципліни з іншими дисциплінами. Історія розвитку технології WPF. Погляд у минуле. Чому не Silverlight? <p>Список рекомендованих джерел: Основний: 1, 2, 4. Додатковий: 5, 6, 8 Інтернет-ресурси: 9, 10, 17, 19</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"> Виникнення WPF. Еволюція WPF. Переваги та особливості WPF. Вибір вікон або сторінок. Обробка подій елементів керування WPF. Модель навігації в WPF. Навігації за гіперпосиланнями. Служба переходів. 	10	5
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	<p style="text-align: center;">Лекція 2. Введення в мову розмітки XAML</p> <p><i>План лекції:</i></p> <ol style="list-style-type: none"> Означення XAML. Елементи і атрибути. Базові типи і простір імен. Елементи властивостей. Властивості і розширення розмітки XAML. Дочірні об'єктні елементи. Диспетчери компонування. Компіляція XAML. <p>Список рекомендованих джерел: Основний: 1, 3, 5.</p>	2	

Результати навчання	Навчальна діяльність*	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
атрибутив; 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.	Додатковий: 6, 7. Інтернет-ресурси: 9, 14, 17, 19.		
	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
	Лабораторне 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
Тема 3. Розробка інтерфейсу користувача		30	15
Знати і розуміти специфіку розробки інтерфейсу користувача на основі	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.	2	

Результати навчання	Навчальна діяльність*	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
технології WPF. Вміти: розташовувати елементи керування на сторінці; розробляти інтерфейс з використанням елементів керування Content Controls; розробляти інтерфейсу з використанням елементів управління Items Controls; використовувати елементи керування Windows Forms в застосуванні WPF.	4. Використання елементів керування Windows Forms в застосуванні WPF. Список рекомендованих джерел: Основний: 1, 2, 3. Додатковий: 6, 8. Інтернет-ресурси: 13 - 19.		
	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
	Лабораторне 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
Тема 4. Налаштування та модифікація зовнішнього вигляду програми		26	15
Знати основні нюанси	Лекція 4 Налаштування та модифікація зовнішнього вигляду програми	2	

Результати навчання	Навчальна діяльність*	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
налагодження та модифікації зовнішнього вигляду програми. Вміти створювати цілісний інтерфейс користувача з використанням стилів; змінювати зовнішній вигляд елементів керування шляхом модифікації його шаблону; використовувати засоби анімації.	<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>		
	<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
	<p>Лабораторне заняття №3 «Налагодження та модифікація зовнішнього вигляду WPF програми»</p> <p><i>Мета:</i> Засвоїти методи налагодження та модифікації зовнішнього вигляду WPF програми.</p>	4	10

Результати навчання	Навчальна діяльність*	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
	<p><i>Завдання:</i> Модифікувати інтерфейс користувача, створений при виконанні попередньої лабораторної роботи, засобами WPF відповідно до зауважень та рекомендацій викладача. Обов'язково скористатися засобами анімації.</p> <p><i>План зняття:</i></p> <ul style="list-style-type: none"> - 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.	<p align="center">Лекція 5. Створення нових елементів керування</p> <p><i>План лекції:</i></p> <ol style="list-style-type: none"> 1. Огляд основних етапів розробки елементів керування. 2. Створення елементів керування користувача: <ol style="list-style-type: none"> a. Створення інтерфейсу елемента керування користувача; b. Наділення елемента керування користувача поведінкою; c. Додавання до елементів керування користувача властивостей залежності; d. Додавання до елементів керування користувача подій, що маршрутизуються. <p>Список рекомендованих джерел: Основний: 1, 2, 4 Додатковий: 7, 8. Інтернет-ресурси: 9 – 12, 18.</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. 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. 	20	5

Результати навчання	Навчальна діяльність*	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
	8. Удосконалення елементів керування за допомогою тем.		
	<p>Лабораторне 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 вдосконалювати 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
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	<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:</p> <p>Основний: 1, 2, 5. Додатковий: 6, 8. Інтернет-ресурси: 9 – 20</p>	2	
	<p>Завдання для самостійної роботи:</p> <p>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. 	28	5

Результати навчання	Навчальна діяльність*	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
даних та подань. Уміти здійснювати навігацію за записами.	<ol style="list-style-type: none"> 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. 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>Лабораторне 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</p>	6	10

Результати навчання	Навчальна діяльність*	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
	попередньої лабораторної роботи інтерфейс користувача. <i>План заняття:</i> - Актуалізація теоретичного матеріалу. - Виконання завдань лабораторної роботи. - Захист виконаної роботи.		
Тема 7. Керування документами		28	20
Знати структуру WPF-документів і засобів їхнього захисту від несанкціонованої зміни. Вміти керувати документами в WPF-застосуваннях; забезпечувати захист документів від несанкціонованої зміни шляхом додавання електронного підпису; додавати графічну і мультимедіа підтримку в WPF-застосуваннях; керувати процесом друку документів на основі програмних засобів.	Лекція №7. Керування документами <i>План лекції:</i> 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. 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.	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. Перевірка даних за замовчуванням. 2. Візуалізація результату перевірки. 3. Визначення користувацького 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.	20	5

Результати навчання	Навчальна діяльність*	Робочий час студента (год.)	Оцінювання (бал)
1	2	3	4
	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.		
	Лабораторне заняття №6 «Перевірка даних на інтерфейсному рівні» <i>Мета:</i> навчитися забезпечувати захист документів від несанкціонованої зміни. <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. <i>План заняття:</i> <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	

* всі лабораторні роботи виконуються на основі інтерактивних методів навчання у комп'ютерному середовищі

5. СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ¹

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>

¹ Курсивом виділені джерела, наявні в бібліотеці КНТЕУ