

**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 цифрової економіки та системного аналізу

ЗАТВЕРДЖЕНО
вченою радою
(пост. п. 7 від 20.06.2021 р.)
Ректор



А. А. Мазаракі

**ТЕХНОЛОГІЯ РОЗРОБКИ МОБІЛЬНИХ ДОДАТКІВ
MOBILE APPS DEVELOPMENT TECHNOLOGY**

**РОБОЧА ПРОГРАМА /
COURSE OUTLINE**

освітній ступінь	Магістр / Master
галузь знань	12 Інформаційні технології / Information Technologies
спеціальність	124 Системний аналіз / System Analysis
спеціалізація	Інформаційні технології та бізнес-аналітика (Data Science) / Information Technologies and Business Analytics (Data Science)

Київ 2021

Розповсюдження і тиражування без офіційного дозволу КНТЕУ заборонено

Автор: В. В. Кулаженко, кандидат економічних наук, доцент
А.М.Тарасюк, асистент кафедри цифрової економіки та системного аналізу

Розглянуто і схвалено на засіданні кафедри цифрової економіки та системного аналізу «25» травня 2021 р., протокол №10.

Рецензент: С.А.Міценко, кандидат технічних наук, доцент
М. Г. Шарафутдінов, директор з розвитку компанії «Center Research & Development», бізнес-аналітик

**ТЕХНОЛОГІЯ РОЗРОБКИ МОБІЛЬНИХ ДОДАТКІВ/
MOBILE APPS DEVELOPMENT TECHNOLOGY**

**РОБОЧА ПРОГРАМА /
COURSE OUTLINE**

освітній ступінь	магістр / master
галузь знань	12 Інформаційні технології / Information Technologies
спеціальність	124 Системний аналіз / System Analysis
освітня програма	Інформаційні технології та бізнес-аналітика (Data Science) / Information Technologies and Business Analytics (Data Science)

1. СТРУКТУРА ДИСЦИПЛІНИ ТА РОЗПОДІЛ ГОДИН ЗА ТЕМАМИ (ТЕМАТИЧНИЙ ПЛАН)

Для вивчення дисципліни відводиться 180 год / 6 кредитів ЄКТС (лекцій – 20 год., лабораторних занять – 20 год., самостійної роботи – 118 год.), підсумковий контроль – письмовий екзамен.

Назва теми	Кількість годин				Форми контролю*
	Усього годин/ кредитів	із них:			
		лекції	лабора- торні заняття	самостій- на робота студентів	
Тема 1. Мобільні додатки та їх місце серед усіх програмних розробок	12	2	0	10	Т, ПР
Тема 2. Основи розробки мобільних додатків	16	2	4	10	Т, ВПЗ, ВІЗ, ПР
Тема 3. Структури даних	20	2	2	12	Т, ВПЗ, ВІЗ
Тема 4. Проектування класів у додатках. Виключення та узагальнення	20	2	2	12	Т, ВПЗ, ВІЗ
Тема 5. Інтерфейси та їх реалізація у додатку	24	2	2	14	Т, ВПЗ, ВІЗ
Тема 6. Візуальне представлення у мобільних додатках	24	2	2	14	Т, ВПЗ, ВІЗ
Тема 7. Проектування взаємодії на основі життєвого циклу додатку	22	2	2	14	Т, ВПЗ, ВІЗ
Тема 8. Місце патернів у додатках	18	2	2	12	Т, ВПЗ, ВІЗ
Тема 9. Логіювання та документування коду	12	2	2	10	Т, ПР
Тема 10. Тестування коду	12	2	2	10	Т, ПР
<i>Підсумковий контроль – письмовий екзамен</i>					
РАЗОМ	180/6	20	20	118	x

Примітка*:

Т – тестування; ВПЗ – виконання практичних завдань; ВІЗ – виконання індивідуальних завдань; ПР – підготовка презентації.

2. ТЕМАТИКА ТА ЗМІСТ ЛЕКЦІЙНИХ, ЛАБОРАТОРНИХ ЗАНЯТЬ І САМОСТІЙНОЇ РОБОТИ СТУДЕНТІВ

Результати навчання	Навчальна діяльність	Робочий час студента, год.
<p>Знати: основні терміни та поняття аналітики великих даних</p> <p>Вміти: виявляти характеристики та проблеми в обробці великих даних</p>	<p>Тема 1. Мобільні додатки та їх місце серед усіх програмних розробок План лекції 1</p> <ol style="list-style-type: none"> 1. Мобільні додатки та їх місце серед програмних розробок. 2. Технології розробки мобільних додатків. 3. Поняття нативної та кросплатформенної розробки. Переваги та недоліки обох підходів. 4. Нативна розробка на IOS. 5. Нативна розробка Android. <p>Список рекомендованих джерел: Основний: 1[с.75-82],2[с.85-97],4[с.72-90]. Додатковий: 6[с.58-90],8[с.48-60],10[с.75-89]. Інтернет-ресурси:4,5</p> <p>Самостійна робота студентів. Вивчення основних та додаткових джерел, Інтернет-ресурсів, підготовка до тестування, підготовка презентації або наукового повідомлення 1[с.12-32],2[с.8-20],4[с.21-32],6[с.20-44],8[с.28-40],9[с.14-32].</p> <p>Підготувати наукове повідомлення (на 5-7 хв.) або презентацію про концепції розробки мобільних додатків:</p>	<p>2</p> <p>10</p>
<p>Знати: Основні програмування</p> <p>Вміти: писати код на с подібних</p>	<p>Тема 2. Основи розробки мобільних додатків План лекції 2</p> <ol style="list-style-type: none"> 1. Паридигми розробки програмного забезпечення. Об'єктно-орієнтована парадигма розробки як найзручніший підхід до розробки додатків. Основні принципи об'єктно-орієнтованої парадигми. 2. Поняття інкапсуляції. Підходи до визначення інкапсуляції. Інкапсуляція та приховування. Модифікатори доступу як елемент 	<p>2</p>

Результати навчання	Навчальна діяльність	Робочий час студента, год.
<p>мовах</p>	<p>приховування даних.</p> <p>3. Поняття спадкування. Критика спадкування. Переваги застосування спадкування у системах. Поняття Upcastingта Downcasting.</p> <p>4. Поняття поліморфізму. Поліморфізм та статична типізація. Види поліморфізму.</p> <p>Список рекомендованих джерел: Основний: 1[с.75-82],2[с.85-97],4[с.72-90]. Додатковий:6[с.58-90],8[с.48-60],10[с.75-89]. Інтернет-ресурси:4,5</p> <p>Самостійна робота студентів. Підготовка до лабораторного заняття за матеріалами теми, вивчення основних та додаткових джерел, Інтернет-ресурсів, підготовка до тестування, виконання індивідуальних завдань, підготовка презентації або наукового повідомлення 1[с.12-32],2[с.8-20],4[с.21-32],6[с.20-44],8[с.28-40],9[с.14-32].</p> <p style="text-align: center;">Лабораторні заняття 1-2 Онови розробки мобільних додатків на мові C#</p> <p>Мета: ознайомлення із можливостями мови програмування C#, основними типами даних математичними та логічними операціями</p> <p>1. Практичне завдання:</p> <ul style="list-style-type: none"> - За наданими викладачем матеріалом для практичного заняття ознайомитись: - Із вводом та виводом інформації - Основними математичними операціями - Логічним операціям - Циклам та роботі з ними умовними конструкціями 	<p style="text-align: center;">10</p> <p style="text-align: center;">4</p>

Результати навчання	Навчальна діяльність	Робочий час студента, год.
<p>Знати: Структури даних та алгоритми сортування</p> <p>Вміти: Проектувати структури, аналізувати роботу алгоритмів</p>	<p align="center">Тема 3. Структури даних План лекції 3</p> <ol style="list-style-type: none"> 1. Поняття структури даних. Основні структури даних. 2. Масиви. Їх види, застосування. Масив як базова структура даних. 3. Список як удосконалений вид масиву. 4. Зв'язаний список та його напрями застосування. 5. Граф як спосіб класифікації даних. 6. Дерево як елемент для створення алгоритмів та ієрархій. 7. Стек та Черга як спосіб контролю входу та виходу даних. 8. Хеш-таблиця та її застосування. <p align="center">Список рекомендованих джерел:</p> <p>Основний: 1[с.75-82], 2[с.85-97], 4[с.72-90]. Додатковий: 6[с.58-90], 8[с.48-60], 10[с.75-89]. Інтернет-ресурси: 4, 5</p> <p>Самостійна робота студентів. Підготовка до лабораторного заняття за матеріалами теми, вивчення основних та додаткових джерел, Інтернет-ресурсів, підготовка до тестування, виконання індивідуальних завдань 1[с.12-32], 2[с.8-20], 4[с.21-32], 6[с.20-44], 8[с.28-40], 9[с.14-32].</p> <p align="center">Лабораторні заняття 3 Реалізація структур даних на мові С#</p> <p>Мета: ознайомлення із можливостями мови програмування С#, основними типами даних математичними та логічними операціями</p> <p>1. Практичне завдання:</p> <ul style="list-style-type: none"> - На основі вивченої інформації розробити та реалізувати у середовищі Visual Studio: - Масив 2 та 3 вимірний 	<p align="center">2</p> <p align="center">12</p> <p align="center">2</p>

Результати навчання	Навчальна діяльність	Робочий час студента, год.
	<ul style="list-style-type: none"> - <i>Стек</i> - <i>Бінарне дерево</i> - <i>Хеш таблиця</i> - <i>Список та зв'язаний список</i> <p><i>Проаналізувати усі наведені структури та можливості їх застосування</i></p>	
<p>Знати:</p> <p>Вміти:</p>	<p>Тема 4. Проектування класів у додатках. Виключення та узагальнення</p> <p style="text-align: center;">План лекції 4</p> <ol style="list-style-type: none"> 1. Поняття класу у програмування. Види класів. 2. Абстрактні класи та їх можливості. 3. Конструктори класів. Приватний та статичний конструктор. 4. Заборона спадкування класу. Види спадкувань 5. Виключення як спосіб мінімізації впливу користувача. Обробка виключень. 6. Фільтр виключень. Користувацькі виключення. <p>Список рекомендованих джерел: Основний: 1[с.75-82],2[с.85-97],4[с.72-90]. Додатковий: 6[с.58-90],8[с.48-60],10[с.75-89].</p> <p>Інтернет-ресурси:4,5</p> <p>Самостійна робота студентів. Вивчення основних та додаткових джерел, Інтернет-ресурсів, підготовка до тестування, підготовка презентації або наукового повідомлення 1[с.12-32],2[с.8-20],4[с.21-32],6[с.20-44],8[с.28-40],9[с.14-32].</p> <p style="text-align: center;">Лабораторні заняття 4 Проектування на мові С#</p> <p>Мета: розробити проект користувацького додатку, виділити основні класи у додатку, розробити систему виключень</p> <p>1. Практичне завдання:</p>	<p style="text-align: center;">2</p> <p style="text-align: center;">12</p>

Результати навчання	Навчальна діяльність	Робочий час студента, год.
системі R	<p align="center">Лабораторні заняття 7</p> <p align="center">Проектування життєвого циклу додатку</p> <p>Мета: навчитись проектувати візуальну та об'єктну частину додатку відповідно до життєвого циклу</p> <p>2. Практичне завдання:</p> <ul style="list-style-type: none"> - Ознайомитись із життєвим циклом складових додатку - Розробити проект зміни активностей, екранів та функціональної взаємодії для основних етапів роботи - Реалізувати поведінку додатку до завантаження основних даних - Реалізувати модель поведінки про отриманні помилки введення/передачі даних - Зміни параметрів сутностей - Паузі роботи основних екранів 	<p align="center">2</p> <p align="center">2</p>
<p align="center">Знати:</p> <p>принципи обробки даних з URL адрес в системі R</p> <p align="center">Вміти:</p> <p>здійснювати основні операції веб-скрапінгу та парсінгу з використанням пакету <i>rvest</i></p>	<p align="center">Тема 8. Місце патернів у додатках</p> <p align="center">План лекції 8</p> <ol style="list-style-type: none"> 1. Найпоширеніші патерни розробки додатків. Підходи до поділу додатку на рівні. Принципи SOLID,KISS,DRY. 2. Патерн MVC та його особливості. Роль моделі у додатку. Робота з контролерами. Проектування візуального дизайну 3. Патерн MVVM та його особливості. Розробка моделей для системи. Проектування візуального представлення. Розробка рівня View-Model. 4. Поняття DTO(Data Transfer Object) та робота із рівнем даних. 5. Вимоги до підключення сторонніх сервісів та баз даних. Поняття ORM <p align="center">6. Список рекомендованих джерел:</p>	<p align="center">2</p> <p align="center">12</p>

Результати навчання	Навчальна діяльність	Робочий час студента, год.
	<p>Основний: 1[с.75-82],2[с.85-97],4[с.72-90]. Додатковий: 6[с.58-90],8[с.48-60],10[с.75-89]. Інтернет-ресурси: 4,5</p> <p>Самостійна робота студентів. Підготовка до лабораторного заняття за матеріалами теми, вивчення основних та додаткових джерел, Інтернет-ресурсів, підготовка до тестування, виконання індивідуальних завдань 1[с.12-32],2[с.8-20],4[с.21-32],6[с.20-44],8[с.28-40],9[с.14-32].</p> <p style="text-align: center;">Лабораторні заняття 8 Патерни у додатках</p> <p>Мета: навчитись застосовувати поняття патернів при розробці додатку</p> <p>Практичне завдання:</p> <ul style="list-style-type: none"> - Ознайомитись із поняттям патернів та їх основними видами - Визначити доцільність застосування патернів у власному додатку - Запропонувати та реалізувати у своєму додатку 2 породжуючих патерни - Запропонувати та реалізувати у своєму додатку 3 структурних патерни - Запропонувати та реалізувати у своєму додатку 3 поведінкові патерни 	2
<p>Знати:</p> <p>Вміти:</p>	<p style="text-align: center;">Тема 9. Логіювання та документування коду План лекції 9</p> <ol style="list-style-type: none"> 1. Поняття логіювання коду. Створення логів у системі. Правила створення логів. Поняття Message Center та його місця у додатку. Застосування логів для мінімізації шкоди від зовнішніх загроз. 2. Поняття Code convention. Підходи до іменування у додатках. Вимоги до іменування класів та інтерфейсів. різниці у іменуванні приватних та публічних полів. Іменування структур. 	2

Результати навчання	Навчальна діяльність	Робочий час студента, год.
	<p>3. Види іменувань. Угорське іменування та його недоліки.</p> <p>4. Поняття документування коду. Підходи до документування коду. Хороші та погані практики. Документування бізнес-логіки. Документування класів. Правила документування методів та виключень.</p> <p style="text-align: center;">Список рекомендованих джерел:</p> <p>Основний: 1[с.75-82], 2[с.85-97], 4[с.72-90]. Додатковий: 6[с.58-90], 8[с.48-60], 10[с.75-89]. Інтернет-ресурси: 4, 5</p> <p>Самостійна робота студентів. Підготовка до лабораторного заняття за матеріалами теми, вивчення основних та додаткових джерел, Інтернет-ресурсів, підготовка до тестування, виконання індивідуальних завдань 1[с.12-32], 2[с.8-20], 4[с.21-32], 6[с.20-44], 8[с.28-40], 9[с.14-32].</p> <p style="text-align: center;">Лабораторні заняття 9 Якість коду</p> <p>Мета: навчитись проводити аналіз коду</p> <p>Практичне завдання:</p> <ul style="list-style-type: none"> - <i>Аналізуючи свій додаток оцінити з викладачем такі показники:</i> - <i>Якість найменування змінних</i> - <i>Написання та обробку функцій та вхідних параметрів</i> - <i>Обробки виключень</i> - <i>Запропонувати шляхи для рефакторингу наявного коду з метою покращення його підтримки</i> - 	<p style="text-align: center;">10</p> <p style="text-align: center;">2</p>
	<p style="text-align: center;">Тема 10. Тестування коду План лекції 10</p> <p>1. Поняття тестування коду. Поняття тесту. Правила тестування. Методи</p>	<p style="text-align: center;">2</p>

Результати навчання	Навчальна діяльність	Робочий час студента, год.
<p>Знати: основні засоби аналізу надвеликих даних з використанням нейронних мереж, алгоритмів машинного навчання та систем штучного інтелекту</p> <p>Вміти: ідентифікувати структури надвеликих даних, що потребують використання методів та моделей машинного навчання</p>	<p>тестування. Види тестів. Оптимізації тестів та їх навантаження. Поняття покриття системи тестами.</p> <ol style="list-style-type: none"> 2. Тестування за знанням системи. Тестування чорної скриньки (blackbox). Тестування білої скриньки (whitebox).Тестування сірої скриньки (greybox) 3. Тестування за ступенем ізольованості компонентів. Компонентне (модульне) тестування (component/unittesting). Інтеграційне тестування (integration testing).Системне тестування (system/end-to-endtesting). 4. Тестування за часом проведення. Альфа-тестування (alphatesting). Тестування при прийманні або Димове тестування (smoketesting).Тестування нової функціональності (newfeature testing). Регресивне тестування (regression testing). Тестування при здачі (acceptance testing). Бета-тестування (beta testing) 5. Тестування за об'єктом. Функціональне тестування (functional testing) Тестування зручності використання або Юзабіліті-тестування (usability testing).Тестування інтерфейсу користувача (UI testing).Тестування безпеки (security testing).Тестування локалізації (localization testing).Тестування сумісності (compatibility testing). 6. Тестування продуктивності (performance testing).Навантажувальне тестування (load testing). Стрес-тестування (stress testing). Тестування стабільності (stability/endurance/soaktesting). <p>Список рекомендованих джерел: Основний: 1[с.75-82],2[с.85-97],4[с.72-90]. Додатковий:6[с.58-90],8[с.48-60],10[с.75-89]. Інтернет-ресурси:4,5</p> <p>Самостійна робота студентів. Підготовка до лабораторного заняття за</p>	<p>10</p>

Результати навчання	Навчальна діяльність	Робочий час студента, год.
	<p>матеріалами теми, вивчення основних та додаткових джерел, Інтернет-ресурсів, підготовка до тестування, виконання індивідуальних завдань 1[с.12-32],2[с.8-20],4[с.21-32],6[с.20-44],8[с.28-40],9[с.14-32].</p> <p style="text-align: center;">Лабораторні заняття 10</p> <p style="text-align: center;">Тестування</p> <p>Мета: навчитись проводити тестування додатків</p> <p>Практичне завдання:</p> <ul style="list-style-type: none"> - <i>Аналізуючи свій додаток розробити модульні тести для аналізу основних компонентів</i> - <i>Довести покриття тестами коду до 90%+</i> - <i>Провести інтеграційний тест з застосування зовнішнього фреймворку чи бібліотеки запропонованої викладачем</i> - <i>Провести тестування безпеки</i> - <i>Провести стрес тест додатку</i> - <i>Провести тест інтерфейсу користувача</i> 	2
<i>Підсумковий контроль – письмовий екзамен</i>		
Разом		180

Примітка:

курсивом позначено інтерактивні методи навчання

3. СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

Основний

1. ZAMMETTI, Frank. Practical Flutter. Berkeley, CA: Apress, 2019.
2. ALBAHARI, Joseph. C# 9.0 in a Nutshell. " O'ReillyMedia, Inc.", 2021.
3. WINDMILL, Eric. Flutter in action. Simon and Schuster, 2020.
4. MATTHES, Eric. Pythoncrashcourse: A hands-on, project-based introduction to programming. no starch press, 2019.
5. DOOLEY, John F.; DOOLEY. Software Development, Design and Coding. Apress, 2017.

Додатковий

6. BILGIN, Can. Mastering Cross-Platform Development with Xamarin. Packt Publishing Ltd, 2016.
7. BADER, Dan. Python Tricks: A Buffet of Awesome Python Features. BookBaby, 2017.
8. PHILLIPS, Dusty. Creating Apps in Kivy: Mobile with Python. " O'ReillyMedia, Inc.", 2014.
9. SKEET, Jonathan. C# in Depth. Simon and Schuster, 2019.
10. HARTSON, Rex; PYLA, Pardha S. The UX book: Agile UX design for a quality user experience. Morgan Kaufmann, 2018.
11. FERREIRA, Alberto. Universal UX design: Building multicultural user experience. Morgan Kaufmann, 2016.
12. RUMPE, Bernhard. Agile modeling with UML: code generation, Testing, Refactoring. Cham: Springer, 2017.
13. GARCIA, Boni. Mastering Software Testing with JUnit 5: Comprehensive guide to develop high quality Java applications. Packt Publishing Ltd, 2017.
14. DE SMEDT, Tom; DAELEMANS, Walter. Pattern for python. The Journal of Machine Learning Research, 2012, 13.1: 2063-2067. *Якщо період видання - перенести в дод. список.*
15. MCKINNEY, Wes, et al. Data structures for statistical computing in python. In: Proceedings of the 9th Python in Science Conference. 2010. p. 51-56.

Інтернет-ресурси

1. Типи даних. URL: https://uk.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85
2. Структура даних. URL: https://uk.wikipedia.org/wiki/%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85
3. Патерни програмування. URL: <https://refactoring.guru/uk/design-patterns>
4. Документація на C#. URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/>
5. Документація на Kivy. URL: <https://kivy.org/doc/stable/>
6. Документація на Flutter. URL: <https://docs.flutter.dev>