

Центральноукраїнський національний технічний університет
Кафедра кібербезпеки та програмного забезпечення



“ЗАТВЕРДЖУЮ”

Проректор з науково-педагогічної роботи

Андрій КИРИЧЕНКО

“25” 08 2025 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Кросплатформені мови програмування

спеціальність _____ (назва навчальної дисципліни)
F7 Комп'ютерна інженерія
освітня програма _____ (шифр і назва спеціальності)
«Комп'ютерна інженерія»
факультет _____ (назва освітньої програми)
механіко-технологічний
(назва факультету)

2025-2026 навчальний рік

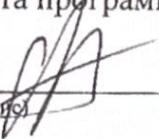
Розробники: Коваленко Олександр Володимирович, доктор технічних наук,
доцент кафедри кібербезпеки та програмного забезпечення
Центральноукраїнського національного технічного університету,
Мошуренко Д. А. Senior Android Engineer, Україна м. Київ, Білефельд
(Німеччина) «Triple A Internetshops GmbH»

(вказати авторів, їхні посади, наукові ступені та вчені звання)

Робочу програму схвалено на засіданні кафедри кібербезпеки та програмного
забезпечення

Протокол №1 від 26 червня 2025 року

Завідувач кафедри кібербезпеки та програмного забезпечення _____



(підпис)

(Олексій СМІРНОВ)
(прізвище та ініціали)

Декан факультету _____ механіко-технологічний _____



(підпис)

(Віталій МАЖАРА)
(прізвище та ініціали)

© Центральноукраїнський
національний технічний університет,
2025 рік

© Коваленко О.В., Мошуренко Д. А.
2025 рік

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 7	Галузь знань <u>F Інформаційні технології</u> (шифр і назва)	нормативна	
Загальна кількість годин: денна форма навчання – 210 заочна форма навчання (повна, бакалавр) – 210	Спеціальність: <u>F7 «Комп'ютерна інженерія»</u>	Рік підготовки	
		2-й	2-й
		Семестр	
		4-й	4-й
	Освітньо-кваліфікаційний рівень: Бакалавр	Лекції	
		32 год.	4 год.
		Лабораторні	
		48 год.	6 год.
		Самостійна робота	
		130 год.	200 год.
Вид контролю:			
Зал, КР.	Зал, КР.		

2. Мета та завдання навчальної дисципліни

Метою викладання дисципліни «Кросплатформені мови програмування» є забезпечення здобувачів вищої освіти комплексом знань, умінь та навичок, необхідних для застосування в професійній діяльності у сфері розробки мобільного ПЗ під керуванням ОС Android. Розглядаються різноманітні аспекти реалізації ПЗ.

Навчальний курс «Кросплатформені мови програмування» призначений для набуття теоретичних знань та практичних навичок розробки програм під керуванням ОС Android. Під час вивчення матеріалу навчального курсу можна розглядаються питання від самої простої програми до публікації своєї роботи на Google Play. Також у навчальному курсі розглядаються наступні питання: огляд середовищ мобільної розробки; види мобільних програм та їх структура; створення користувальницького інтерфейсу незалежно від різних розмірів екранів пристроїв; використання переваг платформи Android; робота з компонентами та ресурсами ОС Android; робота з даними з датчиків смартфона; використання бібліотек.

Основними **завданнями** вивчення навчальної дисципліни є формування наступних **компетенцій бакалавра з комп'ютерної інженерії**:

Завданням вивчення дисципліни є формування компетентностей (Z – загальних, P – фахових):

– **P2.** Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

– **P3.** Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.

У результаті вивчення дисципліни студент повинен забезпечити наступні **програмні результати навчання**:

Знати:

– **N3.** Знати новітні технології в галузі комп'ютерної інженерії.

– **N4.** Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті.

Вміти:

– **N3.** Знати новітні технології в галузі комп'ютерної інженерії.

– **N4.** Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті.

– **N6.** Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.

– **N9.** Вміти застосовувати знання технічних характеристик, конструктивних особливостей, призначення і правил експлуатації програмно-технічних засобів комп'ютерних систем та мереж для вирішення технічних задач спеціальності.

– **N10.** Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.

Набути навичок автономії і відповідальності:

– **N19.** Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.

– **N20.** Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.

– **N21.** Якісно виконувати роботу та досягати поставленої мети з дотриманням вимог професійної етики.

Пререквізити

Враховуючи послідовність накопичення знань і набуття вмінь, для опанування навчальної дисципліни необхідні знання й вміння, здобуті під час вивчення навчальних дисциплін «Об'єктно орієнтоване програмування», «Бази даних».

3. Програма навчальної дисципліни

Лекція 1 Огляд курсу, термінологія, історія розвитку

Статистичні дані використання мобільних операційних систем, інженерні підходи, платформи Google Play, App Store. Частка Android на ринку мобільних платформ. Історія розвитку та оновлень мобільної операційної системи Android. Рівні Android API, зворотна та пряма сумісність.

Лекція 2 Дорожня карта розробника мобільних додатків

Послідовність вивчення матеріалу (Android/iOS developer roadmap). Налаштування, інсталяція використання мобільних додатків (Android package). Маніфест програми. Архітектура мобільного додатка та складові компоненти, інженерні складові.

Лекція 3 Кросплатформені мови програмування

Історія розвитку, переваги та недоліки мов програмування Kotlin, Java. Середовища розробки мовою програмування Kotlin, Java. Використання емулятора та реального телефону. Головні характеристики, архітектура, відносини із іншими мовами та їх вплив на мову Kotlin. Характеристики Java virtual machine. Синтаксис мов Kotlin, Java. Конструкції, операції, типи даних (числові, рядкові та інші), цикли. Елементи об'єктно-орієнтованого програмування. Лямбда-функції. Ключові слова та реалізація об'єктно-орієнтованого програмування. Приклади вихідних кодів реалізації програм на мовах Kotlin, Java.

Лекція 4 Життєвий цикл розробки мобільних додатків

Життєвий цикл розробки мобільного додатку SDLC/SSDLC Android, Apple iPhone. Планування розробки мобільних додатків. Дизайн проекту. Інженерна складова розробки. Створення прототипу. Розробка додатків. Управління якістю. Випуск продукту. Технічне обслуговування, Підтримка.

Лекція 5 Етапи планування мобільних додатків

Визначення цільової аудиторії з взаємозв'язком параметрів розробки. Мобільні ASO та SEO оптимізації. Приклади реалізованих мобільних додатків з цільовою аудиторією. Приклад пошуку цільової аудиторії. Сегментація аудиторії за допомогою методики Марка Шеррінгтона, Метод спектру з протилежними ознаками. Метод середнього розміру чека.

Лекція 6 Принципи проектування та розробки мобільних інтерфейсів UI/UX (частина 1)

Літературна основа, термінологія. Інтерфейс користувача (UI), Досвід користувача (UX) стандарт ISO 9241-210, загальний досвід взаємодії компанії зі своїми клієнтами (CE). Враження від мобільного продукту (BX). Приклади найкращих методик UX для мобільних пристроїв.

Лекція 7 Принципи проектування та розробки мобільних інтерфейсів UI/UX (частина 2)

Наукові підходи реалізації досвіду мобільного користувача на практиці. Закон Фіттса. Закон Хіка. Закон Якоба. Закон Міллера. Бритва Оккама. Закон Паркінсона. Закон Теслера. Ефект Фон Ресторфа. Поріг Доерті.

Лекція 8 Архітектура ОС Android

Розгляд ядра ОС Android на базі Linux, ядра iOS XNU, пояснення рівнів апаратної абстракції та системних служб, аналіз середовищ виконання ART і Objective C Runtime, опис моделей безпеки з підписами додатків і sandbox, демонстрація механізмів енергоменеджменту і продуктивності, узагальнення ключових відмінностей і сучасних тенденцій розвитку платформ.

Лекція 9 Стек датчиків мобільної ОС Android

Архітектура платформи Android. Офіційні типи датчиків Android визначені у Sensor.h. Рівні та стек датчиків. Набір для розробки програмного забезпечення (Software Development Kit). Рівень апаратної абстракції датчиків (HAL) та взаємодія Sensor.h з Sensor.cpp. HAL 1, 2, 2.1, AIDL HAL. Драйвер ядра, концентратор датчиків, фізичні мікросхеми датчиків. Розгляд вихідних кодів (Java, Kotlin) взаємодії з основними типами датчиків: Акселерометр; Магнітометр; Барометр; Пульсометр; Педометр; Сканер відбитків пальців; Датчик світла; Дані GNSS(GPS); Датчик температури; Датчик гіроскопа; Датчик відносної вологості.

Лекція 10 Існуючі різновиди та взаємодія базових та композитивних датчиків

Осі мобільного пристрою. Автомобільні мобільні осі. Фізичні шкали, розмірності та приклади використання датчиків різних фірм. Базові та композитивні датчики (активність, положення, некалібровані та взаємодія). Використання фільтрів Google Play для націлювання на певні конфігурації датчиків. Приклади застосування різних типів датчиків на практиці. Тестові набори комбінацій для перевірки роботи датчиків.

Лекція 11 Мобільна супутникова система навігації (GNSS)

Всесвітня геодезична система (WGS). Метод трилатерації як система виявлення місцезнаходження мобільного пристрою. Супутникова система навігації (GNSS, Global Navigation Satellite System). Складові GNSS: Global Positioning System NAVSTAR (США); Глобальна Навігаційна Супутникова Система; Галілео; BeiDou/COMPASS (Китай); Indian Regional Navigation Satellite System (IRNSS, Індія); QZSS (Японія). СУПУТНИКОВА СИСТЕМА ДИФЕРЕНЦІАЛЬНОЇ КОРЕКЦІЇ (SBAS): WAAS; EGNOS; GAGAN; MSAS; SDCM; SNAS; SACCSA; SouthPAN. Зменшення точності GNSS даних: GDOP; Рефракція атмосфери; Багатопроменеві ефекти; Вибіркова доступність. Час старту GNSS підсистеми та дані альманаху та ефемерид. Приклад вихідного коду open-source GNSS testing app та код отримання даних з GNSS.

Лекція 12 Розробка додатків з врахуванням потреб людей з обмеженими можливостями здоров'я

Розгляд набору практик, які покращують функціональність програми для людей з обмеженими можливостями здоров'я (з різними типами інвалідності). Як вибір неправильного UI програми може вплинути на аудиторію. Рекомендації щодо доступності веб-контенту та мобільних програм (Web Content Accessibility Guidelines - WCAG). Рівні WCAG: A; AA; AAA. Принципи WCAG: Відчутність; Працездатність; Зрозумілість; Стійкість. Повний контрольний список доступності для мобільних пристроїв.

Лекція 13 Особливості забезпечення якості мобільних додатків QA/ QC

Співвідношення між quality assurance (QA), quality control (QC) та тестуванням. Стандарт ISO 9000:2015. Автоматизоване тестування мобільних додатків. Ручне тестування мобільних додатків. Емулятор, симулятор, тестування на реальній хмарі мобільного пристрою. Існуючі типи тестування мобільних додатків: Тестування документації; Тестування функціональності; Тестування зручності використання; Тестування інтерфейсу мобільного додатку; Тестування на сумісність; Тестування продуктивності; Тестування переривань; Тестування безпеки; Тестування локалізації; Бета-тестування.

Лекція 14 Захист інформації в мобільних додатках на рівні розробника (частина 1)

Інформаційна безпека, Мобільна ідентифікація, автентифікація, авторизація. Безпека даних мобільних пристроїв. Статистика мобільних загроз. Огляд існуючих типів вразливостей мобільних додатків. Життєвий цикл розробки програмного забезпечення безпеки SSDLC. Найвпливовіші нормативи конфіденційності даних у світі: правило конфіденційності HIPAA (закон про перенесення та підзвітність медичного страхування); PCI DSS (стандарт безпеки даних індустрії платіжних карток); GDPR (європейський загальний регламент захисту даних); CCPA (каліфорнійський закон про захист прав споживачів).

Лекція 15 Захист інформації в мобільних додатках на рівні розробника (частина 2)

Open Web Application Security Project (OWASP). Open Web Application Security Project Mobile Application Security Project (OWASP MAS). Open Web Application Security Project Mobile Application Security Verification Standard (OWASP MASVS). Open Web Application Security Project Mobile Application Security Testing Guide (OWASP MASTG). OWASP Top Mobile Risks. Static Application Security Testing (SAST). Dynamic Application Security Testing (DAST). Interactive Application Security Testing (IAST). Run-time Application Security Protection (RASP). Найкращі практики безпеки мобільних додатків для розробників. Безпека мобільних додатків на мережному рівні.

Лекція 16 Ключові показники ефективності мобільних додатків (KPI)

Класифікація існуючих KPI мобільних додатків, методи збору та нормалізації даних, метрики залучення та утримання користувачів (DAU, MAU, churn), показники продуктивності й стабільності (час запуску, FPS, crash rate), показники монетизації та LTV, якісні метрики задоволеності користувачів (CSAT, NPS, оцінки в сторах), аналітичні інструменти і дашборди для відстеження KPI.

3.1 Курсова робота з дисципліни «Кросплатформені мови програмування»

Навчальна дисципліна «Кросплатформені мови програмування» передбачає виконання курсової роботи, яка є складовою самостійної роботи здобувачів вищої освіти та спрямована на закріплення теоретичних знань і формування практичних навичок проектування, реалізації та використання мобільних додатків.

Курсова робота виконується на основі індивідуального завдання (тему пропонує здобувач освіти або на вибір з запропонованих тем в методичних вказівках) та полягає у розв'язанні інженерно прикладної задачі з проектування мобільного ПЗ.

Мета курсової роботи: закріплення та поглиблення теоретичних знань з дисципліни «Кросплатформені мови програмування», формування практичних навичок аналізу, проектування, реалізації, тестування та супроводу мобільного

програмного забезпечення, а також набуття досвіду самостійного розв'язання інженерно прикладної задачі з розробки мобільного додатка (зокрема під ОС Android) на основі сучасних засобів і технологій програмування.

Основні завдання курсової роботи

У процесі виконання курсової роботи здобувач повинен:

- Обрати та обґрунтувати тему курсової роботи відповідно до змісту дисципліни і поставленої інженерно прикладної задачі.
- Провести аналіз предметної області, визначити проблему, мету розробки та очікувані результати.
- Сформулювати функціональні та, за потреби, нефункціональні вимоги до мобільного додатка.
- Розробити структуру та логіку роботи програмного засобу, спроектувати основні модулі, сценарії взаємодії та інтерфейс користувача.
- Обґрунтувати вибір інструментальних засобів, мов програмування, бібліотек, фреймворків та середовища розробки.
- Реалізувати мобільний додаток відповідно до поставленого завдання з використанням засобів кросплатформеного програмування.
- Забезпечити коректну роботу додатка з даними, локальним збереженням, мережею або іншими компонентами відповідно до теми роботи.
- Провести тестування та налаштування ПЗ, усунути виявлені помилки.
- Оцінити працездатність і якість розробленого рішення на прикладах використання.
- Оформити пояснювальну записку відповідно до встановлених вимог та підготувати матеріали до захисту.
- Продемонструвати вміння аргументовано представляти результати роботи та відповідати на запитання під час захисту.
- Дотримуватися принципів академічної доброчесності під час виконання, оформлення та представлення курсової роботи.

Форма контролю

Підсумковим етапом виконання курсової роботи є її захист. Захист курсової роботи є формою контролю з дисципліни «Кросплатформені мови програмування».

Захист курсової роботи проводиться в терміни, визначені кафедрою, до початку екзаменаційної сесії та є умовою допуску до екзамену з дисципліни.

4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	денна форма						заочна форма навчання (повна, бакалавр)					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб.	інд.	с.р.		л	п	лаб.	інд.	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Тема 1	13	2		3		8	13,05	0,25		0,3		12,5
Тема 2	13	2		3		8	13,05	0,25		0,3		12,5
Тема 3	13	2		3		8	13,05	0,25		0,3		12,5
Тема 4	13	2		3		8	13,05	0,25		0,3		12,5
Тема 5	13	2		3		8	13,05	0,25		0,3		12,5
Тема 6	13	2		3		8	13,05	0,25		0,3		12,5
Тема 7	13	2		3		8	13,05	0,25		0,3		12,5
Тема 8	13	2		3		8	13,05	0,25		0,3		12,5
Тема 9	13	2		3		8	13,05	0,25		0,3		12,5
Тема 10	13	2		3		8	13,05	0,25		0,3		12,5
Тема 11	13	2		3		8	13,25	0,25		0,5		12,5
Тема 12	13	2		3		8	13,25	0,25		0,5		12,5
Тема 13	13	2		3		8	13,25	0,25		0,5		12,5
Тема 14	13	2		3		8	13,25	0,25		0,5		12,5
Тема 15	14	2		3		9	13,25	0,25		0,5		12,5
Тема 16	14	2		3		9	13,25	0,25		0,5		12,5
Усього годин	210	32		48		130	210	4		6		200

5. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин	
		денна форма навчання	заочна форма навчання (повна, бакалавр)
1	ЛР 1. Основи розробки мобільного ПЗ. Створення першого мобільного ПЗ під керуванням ОС Android.	4	0,75
2	ЛР 2. Робота з контекстним меню. Створення мобільного ПЗ під керуванням ОС Android, робота з компонентами та контекстним меню.	6	0,75
3	ЛР 3. Основи розробки багатівіконного мобільного ПЗ. Створення багатоекранного мобільного ПЗ під керуванням ОС Android.	6	0,75
4	ЛР 4. Основи роботи з текстовими даними. Створення мобільного ПЗ, обробка масивів текстових даних.	6	0,75
5	ЛР 5. Основи роботи зі списками даних. Створення мобільного ПЗ, обробка даних різних типів, робота зі списками.	6	0,75
6	ЛР 6. Основи роботи з різноманітними масивами даних. Створення мобільного ПЗ, обробка різноманітних масивів даних, робота з багаторядковими полями введення/виведення даних.	6	0,75
7	ЛР 7. Основи маніпулювання вікнами ПЗ. Створення мобільного ПЗ, основи роботи з слайдерами та подання графічної інформації.	6	0,75
8	ЛР 8. Робота з Json та Xml даними. зчитування, парсинг та відображення даних у форматах JSON, XML з сортуванням і фільтрацією даних	8	0,75
Усього годин		48	6

6. Самостійна робота

№ з/п	Назва теми	Кількість годин	
		денна форма навчання	заочна форма навчання (повна, бакалавр)
1	Робота з сенсорами.	5	8
2	Створення і зміна баз даних SQLite через Android програми.	5	8
3	Публікація Android програм на Google Play.	5	8
4	Архітектура операційної системи Android.	5	8
5	Архітектура операційної системи iPhone OS.	5	8
6	Мови розмітки XML та XAML.	5	8
7	Робота в інтегрованому середовищі програмування Eclipse.	5	8
8	Робота в інтегрованому середовищі програмування Android Studio.	5	8
9	Робота в інтегрованому середовищі програмування Qt Creator.	5	8
10	Характеристики бездротової технологій передачі даних Bluetooth.	5	8
11	Характеристики бездротової технологій передачі даних Wi-Fi.	5	8
12	Характеристики бездротової технологій передачі даних GSM.	5	8
13	Призначення та характеристики системи позиціонування GPS.	5	8
14	Мережі зв'язку 3-го покоління 3G (HSDPA, HSUPA).	5	8
15	Елементи управління для створення інтерфейсу користувача ПЗ на платформі ОС Android.	5	8
16	Бібліотеки Android архітектури Surface Manager.	5	8
17	Графіка та анімація в ПЗ для ОС Android.	5	8
18	Розповсюдження і публікація прикладень на Google Play.	5	8
19	Життєвий цикл Activity та Fragment і збереження стану інтерфейсу при зміні конфігурації.	5	8
20	Навігація в Android застосунках: Intent, передавання даних між екранами, Back Stack.	5	8
21	Дозволи в Android (runtime permissions) та безпечний доступ до камери, пам'яті, геолокації і мікрофона.	5	8
22	Фонове виконання задач у Android: Service, Foreground Service, WorkManager.	5	8

23	Мережевий обмін даними в Android. HTTP запити, REST API, обробка помилок з'єднання.	7	8
24	Сповіщення в Android Notification, канали сповіщень, дії користувача у сповіщеннях.	6	8
25	Оптимізація продуктивності Android застосунків: використання пам'яті, робота з ресурсами, енергоефективність.	7	8
Усього годин		130	200

Для опанування матеріалу дисципліни «Кросплатформені мови програмування» окрім лекційних та лабораторних занять, тобто аудиторного навантаження, значна увага приділяється самостійній роботі.

До основних видів самостійної роботи студента відносимо:

1. Вивчення лекційного матеріалу.
2. Робота з літературними джерелами.
3. Розв'язання практичних задач за індивідуальними варіантами.
4. Підготовка до модульних, підсумкового контролю, заліку (денна та заочна).
5. Виконання контрольної роботи для заочної форми навчання.

Студенти заочної форми навчання (ЗФН) здебільшого вивчають матеріал самостійно впродовж семестру, тобто самостійно відпрацьовують теми лекцій, а також лабораторних робіт. Для них на початку семестру проводиться установча сесія, під час якої начитують лекції та проводять лабораторні роботи.

Для підвищення рейтингу впродовж семестру студент може виконати згідно запропонованої викладачем теми самостійну роботу, обсяг якої складає не менше 10 сторінок.

7. Індивідуальні завдання

Для студентів заочної форми навчання передбачено виконання контрольних робіт за індивідуальним варіантом (Методичні вказівки до виконання контрольних робіт з дисципліни «Кросплатформені мови програмування» для заочної форми навчання).

Метою виконання контрольних робіт студентами заочної форми навчання є оволодіння практичними навиками розв'язання завдань. Приблизний обсяг контрольної роботи – 10 сторінок (зразок виконання контрольних робіт студентам надаються), плановий обсяг виконання роботи – 9 годин на одну роботу.

8. Методи навчання

Провідна форма навчання – лекція. Лекція дозволяє дуже економно, з мінімальними затратами часу і викладача, і студентів, надати великий обсяг інформації по темі, що розглядається. За характером логіки пізнання впроваджуються аналітичний, індуктивний та дедуктивний методи.

Супровідні методи – лабораторні роботи.

Основна дидактична мета практичного заняття – закріплення й деталізація знань, а головне – формування навичок і вмінь. Для проведення практичного заняття викладач готує відповідні методичні матеріали: тести для виявлення рівня оволодіння необхідними теоретичними положеннями ; набір практичних завдань різної складності для розв'язування їх на занятті та дидактичні засоби.

9. Контроль знань

Види контролю: поточний, підсумковий.

Методи контролю: спостереження за навчальною діяльністю, усне опитування, письмовий контроль, тестовий контроль.

Форма підсумкового контролю: залік, захист курсової роботи.

Контроль знань і умінь здобувачів (поточний і підсумковий) здійснюється згідно з кредитною трансферно-накопичувальною системою організації освітнього процесу в ЦНТУ. Рейтинг студента із засвоєння дисципліни визначається за 100-бальною шкалою. Він складається із рейтингу з поточної навчальної роботи впродовж семестру, для оцінювання якої призначається 100 балів (по 50 балів за кожен змістовний модуль).

Критерії оцінки заліку:

- «зараховано» – студент має стійкі знання про основні поняття дисципліни, може сформулювати взаємозв'язки між поняттями.
- виконано і захищено усі лабораторні роботи.
- захищено курсову роботу.
- «незараховано» – студент має значні пропуски в знаннях, не може сформулювати взаємозв'язку між поняттями, що вивчаються в курсі, не має уявлення про більшість основних понять дисципліни, що вивчається.
- не виконано і не захищено усі лабораторні роботи та курсову роботу.

Розподіл балів, які отримують студенти при вивченні дисципліни «Кросплатформені мови програмування»

Поточний контроль та самостійну роботу																							РАЗОМ	
Змістовий модуль 1											Змістовий модуль 2													
Л1	Л2	Л3	Л4	Л5	Л6	Л7	Л8	ЛР1	ЛР2	ЛР3	ЛР4	Л9	Л10	Л11	Л12	Л13	Л14	Л15	Л16	ЛР5	ЛР6	ЛР7	ЛР8	
4	4	4	4	4	4	4	4	4	4	5	5	4	4	4	4	4	4	4	4	4	4	5	5	100
50											50													

Примітка:Л – теоретичні (лекційні) заняття, ЛР – лабораторні заняття

Критерії оцінювання курсової роботи

Оцінювання курсової роботи здійснюється за такими критеріями:

- Актуальність теми та коректність постановки задачі. Відповідність теми змісту дисципліни, чіткість формулювання мети, завдань і очікуваних результатів.

– Аналіз предметної області та обґрунтування рішення. Повнота аналізу, обґрунтованість вибору підходів, технологій, інструментів і архітектурних рішень.

– Якість проектування програмного засобу. Логічність структури додатка, продуманість модульної побудови, коректність опису алгоритмів і сценаріїв роботи, зручність інтерфейсу користувача.

– Якість програмної реалізації. Працездатність додатка, відповідність реалізації поставленим вимогам, коректність програмного коду, доцільність використання засобів розробки.

– Функціональність та практична цінність результату. Повнота реалізації основних функцій, стабільність роботи, наявність демонстраційних сценаріїв використання, придатність до практичного застосування.

– Тестування та налаштування. Наявність перевірки основних функцій, опис результатів тестування, виявлення та усунення помилок.

– Якість оформлення пояснювальної записки. Структурованість матеріалу, логічність викладу, коректність термінології, відповідність вимогам (ДСТУ) до оформлення, наявність ілюстрацій, схем, фрагментів коду (за потреби).

– Рівень самостійності виконання та дотримання академічної доброчесності. Самостійність розробки, коректне використання джерел, відсутність неправомірних запозичень.

– Чіткість і змістовність доповіді, демонстрація роботи додатка, аргументованість відповідей на запитання.

При виставленні оцінки враховуються результати навчальної роботи студента протягом семестру

Шкала оцінювання: національна та ЄКТС

Сума балів за всі види навчальної діяльності	Оцінка ЄКТС	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90-100	A	відмінно	зараховано
82-89	B	добре	
74-81	C		
64-73	D	задовільно	
60-63	E		
35-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
1-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

10. Рекомендована література

Базова

1. Murphy M. L. Elements of Android Jetpack. [Електронний ресурс]. 2021. URL: <https://commonsware.com/Jetpack/Jetpack-FINAL.pdf>
2. Murphy M. L. Exploring Android. FINAL Version [Електронний ресурс]. 2021. URL: <https://commonsware.com/AndExplore/AndExplore-FINAL.pdf>
3. Murphy, M. L. Elements of Android Room. FINAL Version [Електронний ресурс]. 2021. URL: <https://commonsware.com/Room/Room-FINAL.pdf>
4. Google. Android Developers, офіційна документація та навчальні матеріали з розробки під Android [Електронний ресурс]. URL: <https://developer.android.com/?hl=en>
5. Google. Курси з Android розробки (включно з навчальними шляхами та codelabs) [Електронний ресурс]. URL: <https://developer.android.com/courses?hl=en>
6. Google. Курс Android Basics with Compose, базова розробка застосунків на Kotlin та Jetpack Compose [Електронний ресурс]. URL: <https://developer.android.com/courses/android-basics-compose/course?hl=en>
7. Google. Jetpack Compose, документація UI тулкіта та приклади [Електронний ресурс]. URL: <https://developer.android.com/develop/ui/compose/documentation?hl=en>
8. Google. Архітектура Android застосунку, рекомендації щодо шарів, ViewModel, repository та залежностей [Електронний ресурс]. URL: <https://developer.android.com/topic/architecture?hl=en>
9. Google. Android API Reference, довідник класів та API платформи [Електронний ресурс]. URL: <https://developer.android.com/reference?hl=en>
10. Kovalenko O., Poperehnyak S., Grinenko S., Grinenko O., Radivilova T. «Methods for Assessing the Maturity Levels of Software Ecosystems». *CEUR Workshop Proceedings Volume 2654*, 2019, Pages 251-261. Режим доступу: <https://www.scopus.com/record/display.uri?eid=2-s2.0-85091278920&origin=resultslist> (Scopus).
11. Коваленко О.В. Методи та засоби управління безпекою додатків. Інформаційно-керуючі системи на залізничному транспорті. №4, 2018. – С. 41-44. Режим доступу: http://nbuv.gov.ua/UJRN/Ikszt_2018_4_7 (Фахове видання. Категорія «Б»)
12. Коваленко О.В. Удосконалений метод управління ризиками розробки програмного забезпечення на основі напівмарковської моделі прийняття рішень. Сучасні інформаційні системи. – Випуск 2 (3). – Харків. – 2018. – С. 41-48. Режим доступу: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/40480> (Фахове видання. Категорія «Б»)
13. Коваленко О.В. Методи якісного аналізу та кількісної оцінки ризиків розробки програмного забезпечення. Системи управління, навігації та зв'язку. – Випуск 3 (49). – Полтава: ПолтНТУ. – 2018. – С. 116-125. Режим доступу: <http://journals.nupp.edu.ua/sunz/article/view/1146> (Фахове видання. Категорія «Б»)

14. Коваленко О.В. Моделі та методи розроблення програмного забезпечення комп'ютерних систем для підвищення безпеки даних: **монографія** / О.В. Коваленко // К.: Вид. «КОД» – 2019. – 295 с.

Допоміжна

15. Google. Тестування Android застосунків, підходи та інструменти (unit, instrumented, UI tests) [Електронний ресурс]. URL: <https://developer.android.com/training/testing?hl=en>
16. Google. Room, офіційна документація з роботи з локальною БД через ORM над SQLite [Електронний ресурс]. URL: <https://developer.android.com/training/data-storage/room?hl=en>
17. Google. DataStore, сучасне зберігання налаштувань та даних (Preferences, Proto) [Електронний ресурс]. URL: <https://developer.android.com/topic/libraries/architecture/datastore?hl=en>
18. Google. Практики безпеки Android застосунків, рекомендації щодо даних, дозволів і взаємодій [Електронний ресурс]. URL: <https://developer.android.com/privacy-and-security/security-best-practices?hl=en>
19. Google. Baseline Profiles, оптимізація продуктивності та швидкості запуску [Електронний ресурс]. URL: <https://developer.android.com/topic/performance/baselineprofiles/overview?hl=en>
20. Google. Профілювання в Android Studio, інструменти аналізу CPU, пам'яті, енергії та мережі [Електронний ресурс]. URL: <https://developer.android.com/studio/profile?hl=en>
21. Kotlin. Coroutines overview, основи корутин та асинхронності в Kotlin для Android [Електронний ресурс]. URL: <https://kotlinlang.org/docs/coroutines-overview.html>
22. Gradle. Gradle User Manual, документація системи збірки для Android проєктів [Електронний ресурс]. URL: <https://docs.gradle.org/>
23. Google. Налаштування Firebase для Android проєкту, підключення SDK та конфігурація [Електронний ресурс]. URL: <https://firebase.google.com/docs/android/setup?hl=en>
24. Google. Довідка Play Console, створення та первинне налаштування застосунку для публікації [Електронний ресурс]. URL: <https://support.google.com/googleplay/android-developer/answer/9859152?hl=en>
25. Dawn Griffiths, David Griffiths. Head First Android Development. O'Reilly Media, Inc. 2021. 1414 с.
26. Poul Klausen. JAVA 17: more about Java and Android software development. Bookboon. 2018. 257 с.
27. Peter Sommerhoff. Kotlin for Android App Development. Addison-Welsey Professional. 2019. 435 с.
28. Alexey Soshin. Kotlin Design Patterns and Best Practices. Packt Publishing. 2022. 513 с.
29. Pierre-Olivier Laurence, Amanda Hinchman-Dominguez. Programming Android with Kotlin. O'Reilly Media. 2022. 355 с.

30. Kevin D. Moore, Carlos Mota, Saeed Taheri. Kotlin Multiplatform by Tutorials. Razeware LLC. 2022. 400 с.
31. Nathan Metzler. Kotlin Programming for Beginners. Independently published. 2021. 158 с.
32. Filip Babić, Luka Kordić, Nishant Srivastava. Kotlin Coroutines by Tutorials. Razeware LLC. 2022. 287 с.
33. Irina Galata, Victoria Gonda, Joe Howard, Ellen Shapiro. Kotlin Apprentice. Razeware LLC. 2021. 491 с.
34. René Cacheaux & Josh Berlin. Advanced iOS App Architecture. Razeware LLC. 2022. 334 с.
35. Ehab Amer, Alexis Gallagher, Matt Galloway, Eli Ganim, Ben Morrow, Cosmin Pupăză. Swift Apprentice. Razeware LLC. 2021. 555 с.
36. Neil Smyth. SwiftUI Essentials. Payload Media. 2022. 526 с.
37. Will Grant. 101 UX Principles. Packt Publishing. 2022. 432 с.

Методичне забезпечення

38. Коваленко О.В., Коваленко А.С. «Кросплатформові мови програмування». Методичні вказівки до виконання лабораторних робіт для студентів денної та заочної форми навчання галузі Інформаційні технології. – Кропивницький: ЦНТУ – 2025. – 37 с.
39. Методичні рекомендації до виконання, оформлення та захисту курсової роботи з дисципліни «Кросплатформові мови програмування»: для студентів денної та заочної форм навчання галузі Інформаційні технології / [уклад.: О.В. Коваленко, А.С. Коваленко]; М-во освіти і науки України, Центральноукраїн. нац. техн. ун-т, каф. кібербезпеки та програмного забезпечення. - Кропивницький : ЦНТУ, 2025.

Інформаційні ресурси

40. Курс «Кросплатформені мови програмування» на сервері дистанційної освіта ЦНТУ. – URL: <https://moodle.kntu.kr.ua/course/view.php?id=1040>
41. Онлайн-курси UDEMY. – URL: <https://www.udemy.com/> –платформа онлайн-курсів різних ІТ тематик.
42. Онлайн-курси Prometheus. – URL: <https://prometheus.org.ua/> – українська платформа безкоштовних онлайн-курсів
43. Онлайн-курси Coursera. – URL: <https://www.coursera.org> –платформа онлайн-курсів різних ІТ тематик.
44. <https://biblprog.org.ua/ua/programming/> – каталог безкоштовних середовищ розроблення ПЗ.