

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1. Загальна інформація про навчальну дисципліну

Повна назва навчальної дисципліни	Програмування для мобільних пристроїв
Повна офіційна назва закладу вищої освіти	Сумський державний університет
Повна назва структурного підрозділу	Факультет електроніки та інформаційних технологій. Кафедра інформаційних технологій
Розробник(и)	Нагорний Володимир В`ячеславович
Рівень вищої освіти	Другий рівень вищої освіти, НРК – 7 рівень, QF-LLL – 7 рівень, FQ-EHEA – другий цикл
Семестр вивчення навчальної дисципліни	16 тижнів протягом 2-го семестру
Обсяг навчальної дисципліни	Обсяг становить 5 кред. ЄКТС, 150 год. Для денної форми навчання 64 год. становить контактна робота з викладачем (32 год. лекцій, 32 год. лабораторних занять), 86 год. становить самостійна робота.
Мова викладання	Українська

2. Місце навчальної дисципліни в освітній програмі

Статус дисципліни	Обов`язкова навчальна дисципліна для освітньої програми "Інформаційні технології проектування"
Передумови для вивчення дисципліни	Необхідними для вивчення дисципліни є базові знання з дисципліни "Сховища даних".
Додаткові умови	Додаткові умови відсутні
Обмеження	Обмеження відсутні

3. Мета навчальної дисципліни

Метою вивчення курсу є оволодіння студентом основних теоретичних і практичних прийомів створення додатків для мобільних операційних систем

4. Зміст навчальної дисципліни

Тема 1 ОС Android

Дослідження історії виникнення та еволюції ОС Android, з фокусом на ключових моментах та впливових факторах. Стандарти розробки мобільних додатків. Стандарт до збору вимог до мобільного додатку. Вибір найкращої технології розробки мобільних додатків. Види розробки мобільних додатків. Стандарти UI/UX для дизайну мобільних додатків. Типи вказівок щодо інтерфейсу користувача. Аналіз основних складових та особливостей розробки під ОС Android, включаючи архітектуру, інструменти та практичні поради. Створення та запуск першого Android-проекту, вивчення основних етапів розробки та тестування додатків для платформи Android.

Тема 2 Введення в Kotlin

Основи мови Kotlin: синтаксис, типи даних і основні концепції. Об'єктно-орієнтоване програмування в Kotlin: поглиблене вивчення ООП в Kotlin, включаючи класи, об'єкти, наслідування, інтерфейси та поліморфізм для створення ефективних програм.

Тема 3 Дослідження Activity

Аналіз Activity у програмуванні та їхніх впливів на додатки, оптимізація процесів та вдосконалення користувацького досвіду. Розгляд основних аспектів концепції "Activity" у розробці Android-додатків та їхню роль у структурі додатку. Аналіз всіх етапів життєвого циклу Activity в Android-додатках, включаючи створення, запуск, призупинення та завершення. Вивчення різних станів, таких як "активний", "призупинений" та "зупинений", і їхню важливість для управління ресурсами та взаємодії з користувачем у Android-додатках.

Тема 4 Графічний інтерфейс

Створення графічного інтерфейсу в Android з використанням Kotlin: дослідження базових віджетів і контейнерів для побудови інтуїтивно зрозумілих та привабливих додатків. Робота зі списками в мобільних додатках на платформі Android. Використання Kotlin для створення списків з можливістю додавання, видалення та оновлення даних для покращення користувацького досвіду.

Тема 5 Фрагменти

Використання фрагментів у програмуванні мобільних додатків під Android (Kotlin). Створення і керування фрагментами для покращення користувацького інтерфейсу та роботи з різними екранами. Взаємодія між фрагментами та Activity: Методи комунікації між фрагментами та активностями для побудови динамічних та зручних додатків. Життєвий цикл фрагментів. Етапи життєвого циклу фрагментів для правильного управління станом та подіями в додатку.

Тема 6 Трансляції (Broadcasts)

Механізм розсилки повідомлень між компонентами додатка, що дозволяє реагувати на події системи. Огляд способів роботи зі звичайними та власними трансляціями в мові Kotlin. Створення та реєстрація власних трансляцій для внутрішнього спілкування додатка. Дослідження можливостей використання черг для обробки трансляцій в правильному порядку. Використання функціонального підходу у Kotlin для оптимізації роботи зі звітами і обробки подій трансляцій.

Тема 7 Збереження даних

Можливість зберегти дані в локальному файлі на Android-пристрої для подальшого доступу і використання. Використання механізму SharedPreferences для зберігання простих налаштувань та даних на Android для зручного та структурованого збереження. Створення та взаємодія з SQLite базою даних для ефективного збереження та організації інформації в Android-додатках. Використання мови SQL для створення, оновлення, видалення і вибору даних в SQLite базі даних на мові Kotlin. Робота з транзакціями в SQLite для забезпечення цілісності та консистентності даних у додатку, здатному виконувати кілька операцій одночасно та атомарно.

Тема 8 Обмін даними між програмами за допомогою ContentProvider

ContentProvider, як компонент Android для спільного доступу до даних між програмами через URI. Дозволи програми та їх налаштування для забезпечення безпеки та функціональності додатка. Використання URI для отримання доступу до даних інших додатків через ContentProvider. Створення власного ContentProvider для забезпечення доступу до даних вашого додатка. Обмін даними між додатками за допомогою ContentProvider або інших методів. Використання універсальних типів та делегатів для покращення коду в Kotlin. Використання лінивих функцій для відкладеного обчислення значень та оптимізації продуктивності.

Тема 9 Мультимедіа

Розробка мультимедійних Android додатків на Kotlin для відтворення відео та аудіо, роботи з графічними ресурсами. Реалізація сповіщень у мобільних додатках на Kotlin, включаючи push-сповіщення та взаємодію з користувачем. Розробка функціоналу для роботи з камерою та альбомом пристрою у додатках на Kotlin, з можливістю зйомки та обробки фото та відео. Розробка модулів для відтворення аудіо та відео файлів у додатках на Kotlin, включаючи підтримку різних форматів та інтерфейси користувача.

Тема 10 Робота з фоновією службою

Створення та керування потоками для покращення продуктивності Android додатку, використовуючи мову Kotlin. Забезпечення плавного оновлення UI компонентів в основному потоці та мінімізувати блокування. Використання Kotlin Coroutines та Handler для ефективної обробки асинхронних подій та повідомлень. Створення та використання сервісів для виконання фонових завдань. Способи обміну даними та комунікації між активностями та фоновими сервісами в Kotlin. Аналіз життєвого циклу сервісів. Керування їхнім життєвим циклом в Android додатку. Використання конкретизації узагальнених типів для покращення безпеки та читабельності коду на Kotlin. Використання коваріантності та контраваріантності в Kotlin для оптимізації типізації та поліморфізму.

Тема 11 Робота з мережею

Використання WebView для вбудованого браузеру та відображення веб-сторінок у додатку на Kotlin для Android. Використання HTTP-запитів для отримання та передачі даних з мережі в додатку. Робота з HttpURLConnection для створення зв'язку з сервером та обміну даними. Використання бібліотеки OkHttp для ефективного створення та обробки HTTP-запитів у додатку. Розбір та обробка отриманих даних, таких як JSON, XML. Створення мережевих зворотних викликів (Network Callback) для асинхронного отримання даних з сервера. Використання бібліотеки Retrofit для спрощення роботи з HTTP-запитами та об'єднання їх зі структурою даних Kotlin. Застосування корутин для асинхронних операцій у мережевій взаємодії та обробці даних.

Тема 12 Material Design

ToolBar. Navigation Drawer. DrawerLayout. NavigationView. FloatingActionButton та Snackbar. CoordinatorLayout. CardView Layout. MaterialCardView. AppBarLayout. Оновлення сторінки свайпом. Collapsible ToolBar. CollapsingToolBarLayout. Оптимізація використання системного рядка стану. Створення утиліт в Kotlin. Спрощене використання Toast. Спрощене використання Snackbar.

Тема 13 Високоякісні компоненти розробки. Дослідження Jetpack

ViewModel. LiveData. Map та SwitchMap. Room. WorkManager. Обробка комплексних задач за допомогою WorkManager. Основи роботи з Jetpack Compose. Стан в Jetpack Compose. Бібліотека завантаження зображень Coil. Модульне тестування в Android. Поради щодо модульного тестування в Android. Розробка, керована тестуванням (TDD). Написання модульних тестів до існуючого коду. Використання бібліотеки MockK. Огляд функцій mockk() і spyk(). Розширення бібліотеки MockK для класів Android. Мокінг об'єкту SharedPreferences. Тестування додатків, які використовують базу даних Room. Делегування поведінки mock-об'єкта лямбда-функції.

Тема 14 Використання поглиблених технік створення мобільних додатків

Глобальний контекст у додатках Android, як спільний ресурс, доступний з будь-якого місця програми, де можна отримати доступ до контексту додатка. Передача складних об'єктів між активностями за допомогою Intent, забезпечуючи збереження його стану. Використання інтерфейсу Serializable для збереження об'єкта в стані байтів для передачі через Intent. Оптимізований спосіб передачі об'єктів між компонентами за допомогою Parcelable і Parcelable.Creator. Розробка власного механізму для ефективного логування подій та помилок у додатках Android для налагодження та аналізу. Ін'єкція залежностей в Android. Основи ін'єкцій залежностей. Автоматизована ін'єкція залежностей. Альтернативи ін'єкції залежностей. Ручна ін'єкція залежностей. Управління залежностями за допомогою контейнера. Управління залежностями в потоках програми. Ін'єкції залежностей з використанням Hilt. Впровадження залежностей в класи Android. Визначення прив'язок Hilt. Модулі Hilt. Ін'єкція екземпляра інтерфейсу за допомогою @Binds. Прив'язки компонентів за замовчуванням. Hilt в багатомодульних додатках. Hilt у функціональних модулях. Використання Hilt з іншими бібліотеками Jetpack.

5. Очікувані результати навчання навчальної дисципліни

Після успішного вивчення навчальної дисципліни здобувач вищої освіти зможе:

РН1	Демонструвати вміння проектувати архітектурні рішення для мобільних додатків на платформі Android, включаючи вибір паттернів проектування та оптимальну структуру програми
-----	--

PH2	Оцінювати якість свого програмного продукту, використовуючи методи тестування, аналізу коду та інші підходи для забезпечення якості
PH3	Проводити тестування мобільних додатків на платформі Android, включаючи функціональне, інтеграційне та тестування на різних пристроях і версіях ОС Android
PH4	Виявляти проблеми, які виникають під час роботи мобільних додатків на Android, та розробляти відповідні завдання для їх виправлення або модифікації
PH5	Формувати систематизоване уявлення про концепції, принципи, методи, технології створення програмних додатків для мобільних операційних систем.

6. Роль навчальної дисципліни у досягненні програмних результатів

Програмні результати навчання, досягнення яких забезпечує навчальна дисципліна.
Для спеціальності 122 Комп'ютерні науки:

PP10	Проектувати архітектурні рішення інформаційних та комп'ютерних систем різного призначення
PP13	Оцінювати та забезпечувати якість інформаційних та комп'ютерних систем різного призначення
PP14	Тестувати програмне забезпечення
PP17	Виявляти та усувати проблемні ситуації в процесі експлуатації програмного забезпечення, формулювати завдання для його модифікації або реінжинірингу
PP23	Формувати систематизоване уявлення про концепції, принципи, методи, технології створення програмних додатків для мобільних операційних систем

7. Роль освітнього компонента у формуванні соціальних навичок

Загальні компетентності та соціальні навички, формування яких забезпечує навчальна дисципліна:

CH1	Здатність до абстрактного мислення, аналізу та синтезу.
CH2	Здатність застосовувати знання у практичних ситуаціях.
CH3	Здатність вчитися й оволодівати сучасними знаннями.
CH4	Здатність генерувати нові ідеї (креативність).

8. Види навчальних занять

Тема 1. ОС Android

<p>Лк1 "Становлення ОС Android. Огляд основних засобів для розробки мобільних додатків." (денна)</p> <p>Коротка історія виникнення та становлення. Огляд ОС Android. Системна архітектура Android. Версії ОС Android. Стандарти на розробку мобільних додатків із врахуванням UI/UX. Основні особливості і складові при розробці під ОС Android. Встановлення середовища розробки. Встановлення емулятора. Створення нового проекту. Запуск проекту. Аналіз проекту. Інструменти ведення журналу логів: Android Log Tool.</p>
<p>Лб1 "Робота з журналами логів та дослідження структури проекту в Android Studio" (денна)</p> <p>Знайомство з середовищем розробки Android Studio. Робота з інструментарієм ведення журналів логування.</p>
<p>Тема 2. Введення в Kotlin</p>
<p>Лк2 "Основи мови Kotlin" (денна)</p> <p>Запуск коду Kotlin. Змінні та функції. Конструкції розгалуження та цикли. Об'єктно-орієнтоване програмування. Класи і об'єкти. Наслідування та функція конструктора.</p>
<p>Лк3 "Об'єктно-орієнтоване програмування в Kotlin" (денна)</p> <p>Інтерфейси. Data класи та Singleton. Лямбда вирази. Створення та ітерація колекцій. Функціональні API колекції. Null Safety. Тип Nullable. Інтерполяція рядків. Аргументи функції за замовчуванням.</p>
<p>Лб2 "Робота з примітивати мови Kotlin. Конструкції вибору." (денна)</p> <p>Реалізація завдань із застосуванням знань про типи даних. Особливості реалізації функцій в Kotlin. Використання конструкцій розгалуження: if та when. Застосування циклів.</p>
<p>Лб3 "Класи та об'єкти в Kotlin" (денна)</p> <p>Організація структури коду проекту в Kotlin з використанням об'єктно-орієнтованого підходу.</p>
<p>Тема 3. Дослідження Activity</p>
<p>Лк4 "Знайомство з Activity" (денна)</p> <p>Поняття Activity. Створення розмітки. Реєстрація Activity в файлі AndroidManifest. Використання спливаючих повідомлень. Використання меню в Activity. Знищення Activity. Використання Intent для взаємодії між Activity. Явний Intent. Неявний Intent. Передача даних в Activity. Життєвий цикл Activity. Back Stack. Стани Activity. Перестворення Activity.</p>
<p>Лб4 "Передача даних між Activity за допомогою об'єкта Intent" (денна)</p> <p>Створення та реєстрація Activity. Налаштування взаємозв'язку між Activity. Робота з Intent та життєвим циклом Activity. Створення простих макетів розмітки.</p>
<p>Тема 4. Графічний інтерфейс</p>

Лк5 "Дослідження основних елементів для створення графічного інтерфейсу мобільних додатків" (денна)

Створення графічного інтерфейсу. TextView. Button. EditText. ImageView. ProgressBar. AlertDialog. Базові Layouts. LinearLayout. RelativeLayout. FrameLayout. ListView. Кастомізація ListView. Оптимізація роботи з ListView. Обробка вибору елемента в ListView. RecyclerView. Горизонтальне прокручування в RecyclerView. Обробка вибору елемента в RecyclerView.

Лб5 "Робота із списками. RecyclerView" (денна)

Створення мобільного додатку із наявністю списку. Реалізованого за допомогою компонента RecyclerView та можливістю обробки вибору елемента списку з представленням результату у іншому вікні додатку.

Тема 5. Фрагменти

Лк6 "Фрагменти і їх життєвий цикл" (денна)

Використання фрагментів. Додавання фрагментів динамічно. Back Stack фрагментів. Взаємодія між фрагментами та Activity. Життєвий цикл фрагментів. Приклад використання фрагментів. Функції розширення і перевантаження операторів Kotlin.

Лб6 "Робота фрагментами" (денна)

Створення мобільного додатку. Який має забезпечити використання фрагментів для мобільних пристроїв з портретною орієнтацією екранів (смартфонів) та альбомною орієнтацією екранів (планшетів).

Тема 6. Трансляції (Broadcasts)

Лк7 "Дослідження механізму трансляцій" (денна)

Вступ до механізму трансляції. Отримання системної трансляції. Робота з власними трансляціями. Надсилання впорядкованої по черзі трансляції. Функції вищого порядку в Kotlin. Noinline і Crossinline.

Лб7 "Прослуховування зміни стану компонентів мобільного пристрою" (денна)

Створення мобільного додатку. Який дозволяє прослуховувати за допомогою механізму трансляцій зміни стану компонентів ОС Android мобільного пристрою.

Тема 7. Збереження даних

Лк8 "Механізми збереження даних в ОС Android" (денна)

Збереження в файлі. Запис у файл. Зчитування з файлу. SharedPreferences. Збереження даних за допомогою SharedPreferences. Зчитування даних з SharedPreferences. SQLite Database. Створення бази даних. Додавання даних. Оновлення даних. Видалення даних. Запити з умовою. Використання SQL. Транзакції.

Лб8 "Збереження даних засобами SQLite" (денна)

Створення мобільного додатку з базою даних sqlite. Який дозволяє реалізувати стандартні CRUD операції: створення. Читання. Оновлення. Видалення та запити з умовою.

<p>Тема 8. Обмін даними між програмами за допомогою ContentProvider</p>
<p>Лк9 "ContentProvider як інструмент обміну даними між програмами" (денна) ContentProvider. Запит дозволів під час виконання програми. Доступ до даних інших програм. Читання системних контактів. Створення власного ContentProvider. Обмін даними між програмами. Універсальні типи та делегати в Kotlin. Реалізація lazy функцій.</p>
<p>Лб9 "Використання ContentProvider для обміну даними між двома мобільними додатками" (денна) Створення пари мобільних додатків: перший – зберігає дані в базі даних. Другий – реалізує механізм безпечного доступу до бази даних першого додатку з використанням ContentProvider.</p>
<p>Тема 9. Мультимедіа</p>
<p>Лк10 "Огляд основних інструментів мультимедіа в ОС Android" (денна) Сповіднення. Робота з простими сповіщеннями. Розширені можливості роботи із сповіщеннями. Камера та альбом. Створення фото за допомогою камери. Відображення фото з альбому. Відтворення мультимедійних файлів. Відтворення аудіо файлів. Відтворення відео файлів.</p>
<p>Лб10 "Робота з галереєю" (денна) Створення додатку «Медіа Галерея». Який дозволяє користувачу зробити фото. Зберегти його в галереї пристрою і відобразити вміст галереї у вигляді списку.</p>
<p>Тема 10. Робота з фоновією службою</p>
<p>Лк11 "Фонові служби як інструмент виконання задач в асинхронному режимі" (денна) Багатопоточність в Android. Оновлення графічного інтерфейсу в робочому потоці. Механізм обробки асинхронних повідомлень. Використання AsyncTask. Сервіси. Взаємодія між Activity та сервісом. Життєвий цикл сервісів. Використання сервісів переднього плану. Використання IntentService. Конкретизація (Reify) узагальнених типів. Використання конкретизованих типів в Андроїд. Коваріантність та контраваріантність.</p>
<p>Лб11 "Робота з сервісами в фоновому режимі" (денна) Створення додатку «Таймер». Який використовує сторонній сервіс для обліку часу таймера і управління запуском сервісу відбувається як асинхронна операція з використанням AsyncTask або IntentService.</p>
<p>Тема 11. Робота з мережею</p>
<p>Лк12 "Огляд основних інструментів виконання запитів до віддалених даних" (денна) WebView. Використання HTTP для доступу до мережі. Використання HttpURLConnection. Використання OkHttp. Парсинг даних в XML. SAX парсинг. Парсинг JSON. JSONObject. GSON. Реалізація мережевого зворотного виклику (Network Callback). Мережева бібліотека Retrofit. Базове використання Retrofit. Процес створення складного інтерфейсу адреси. Використання Retrofit Builder. Використання Корутинів для ефективного багатопотокового додатку. Coroutine Scope Builder. Спрощення зворотних викликів з корутинами.</p>

Лб12 "Доступ до даних API засобами бібліотеки Retrofit" (денна)

Створення додатку. Який використовує доступ до даних через відкриту API за допомогою бібліотеки Retrofit та може здійснювати основні CRUD операції та запити з параметрами. Для оптимізації здійснення запитів до API використовуються корутини.

Тема 12. Material Design

Лк13 "Огляд концепції Material Design для створення сучасних графічних інтерфейсів мобільних додатків" (денна)

ToolBar. Navigation Drawer. DrawerLayout. NavigationView. FloatingActionButton та Snackbar. CoordinatorLayout. CardView Layout. MaterialCardView. AppBarLayout. Оновлення сторінки свайпом. Collapsible ToolBar. CollapsingToolBarLayout. Оптимізація використання системного рядка стану. Створення утиліт в Kotlin. Спрощене використання Toast. Спрощене використання Snackbar.

Лб13 "Створення додатку з використанням віджетів Material Design" (денна)

Створення додатку «To-Do», який використовує для побудови графічного інтерфейсу користувача віджети Material Design. Що реалізують базовий функціонал взаємодії з списком справ: додавання, видалення, оновлення.

Тема 13. Високоякісні компоненти розробки. Дослідження Jetpack

Лк14 "Введення в Jetpack" (денна)

ViewModel. Базове використання ViewModel. Передача параметрів у ViewModel. Життєвий цикл. LiveData. Базове використання LiveData. Map та SwitchMap. Room. Використання Room для CRUD операцій. Оновлення Room бази даних. Основи модульного тестування в Android. Ручне модульне тестування. Модульне тестування із використанням JUnit. Використання бібліотеки MockK. Тестування додатків, які використовують базу даних Room.

Лк15 "WorkManager" (денна)

WorkManager. Базове використання WorkManager. Обробка комплексних задач за допомогою WorkManager. Основи роботи з Jetpack Compose. Рядки. Стовпці та основні розміри. Compose-модифікатори. Стилзація тексту. Стан в Jetpack Compose. ConstraintLayout в Jetpack Compose. Бібліотека завантаження зображень Coil.

Лб14 "Jetpack. Room база даних" (денна)

Створення додатку для керування замітками. Який використовує базу даних Room. Проект будується з дотриманням архітектури поділу на зони відповідальності: шар доступу до даних. Шар бізнес-логіки та шар презентації. Для відображення та збереження даних заміток використовується ViewModel разом з LiveData для відстеження змін даних. Для трансформації та фільтрації даних з LiveData використовується Map та SwitchMap.

Лб15 "Модульне тестування мобільних додатків засобами JUnit та MockK" (денна)

Написання модульних тестів до мобільного додатку з використанням фреймворку JUnit та бібліотеки MockK.

Тема 14. Використання поглиблених технік створення мобільних додатків

Лк16 "Поглиблені техніки створення мобільних додатків" (денна)

Отримання контексту глобально. Передача об'єкта в Intent. Serializable. Parcelable. Створення власного інструменту логування. Ін'єкція залежностей в Android. Ручна ін'єкція залежностей. Управління залежностями за допомогою контейнера. Ін'єкції залежностей з використанням Hilt. Використання Hilt з бібліотеками Jetpack.

Лб16 "Передача об'єкта в Intent. Serializable. Parcelable" (денна)

Створення додатку додавання та перегляду заміток з можливістю користувача передавати інформацію про конкретну замітку для перегляду у другому вікні через Intent у вигляді Serializable чи Parcelable об'єктів. При цьому використовуючи глобальний контекст.

9. Стратегія викладання та навчання

9.1 Методи викладання та навчання

Дисципліна передбачає навчання через:

МН1	Лекційне навчання
МН2	Практикоорієнтоване навчання
МН3	Проблемне навчання
МН4	Самостійне навчання

Лекції надають студентам матеріали з основними концепціями, принципами та методами створення мобільних додатків для операційної системи Android (PH1, PH2). Лекції доповнюються лабораторними заняттями, що надають студентам можливість застосовувати теоретичні знання на практичних прикладах (PH1-PH4). Практикоорієнтоване навчання передбачає набуття студентами навичок із створення мобільних додатків для операційної системи Android за допомогою середовища розробки Android Studio та фреймворку Kotlin for Android з використанням сучасних підходів, що використовуються в реальних проектах (PH1-PH5). Проблемне навчання використовується при роботі над контрольною роботою та полягає в аналізі студентами можливостей фреймворку Kotlin та середовища розробки Android Studio до реалізації сучасних мобільних додатків із дотриманням останніх тенденцій у мобільній розробці. Самостійна робота студентів полягає в опрацюванні рекомендованих викладачем літературних джерел на задану тематику.

Опанування дисципліни сприяє розвитку у студентів аналітичного, логічного мислення, тайм-менеджменту, публічної презентації одержаних результатів.

9.2 Види навчальної діяльності

НД1	Інтерактивні лекції
НД2	Виконання лабораторних робіт за темами 1-14
НД3	Контрольна робота (розроблення мобільного додатку)
НД4	Робота з підручниками та релевантними інформаційними джерелами
НД5	Електронне навчання у системі mix.sumdu.edu.ua

10. Методи та критерії оцінювання

10.1. Критерії оцінювання

Визначення	Чотирибальна національна шкала оцінювання	Рейтингова бальна шкала оцінювання
Відмінне виконання лише з незначною кількістю помилок	5 (відмінно)	$90 \leq RD \leq 100$
Вище середнього рівня з кількома помилками	4 (добре)	$82 \leq RD < 89$
Загалом правильна робота з певною кількістю помилок	4 (добре)	$74 \leq RD < 81$
Непогано, але зі значною кількістю недоліків	3 (задовільно)	$64 \leq RD < 73$
Виконання задовольняє мінімальним критеріям	3 (задовільно)	$60 \leq RD < 63$
Можливе повторне складання	2 (незадовільно)	$35 \leq RD < 59$
Необхідний повторний курс з навчальної дисципліни	2 (незадовільно)	$0 \leq RD < 34$

10.2 Методи поточного формативного оцінювання

	Характеристика	Дедлайн, тижні	Зворотний зв'язок
МФО1 Настанови викладача в процесі виконання лабораторних завдань	Перевірка виконаних лабораторних завдань на аудиторному занятті.	1 тиждень	Коментарі під час заняття від викладача
МФО2 Обговорення та самокорекція виконаної роботи студентами	Обговорення із студентами опрацьованих самостійно розділів рекомендованої додаткової літератури для поглибленого вивчення матеріалу дисципліни	2 тижні	Групові та індивідуальні консультації
МФО3 Надання зворотного зв'язку про результати перевірки виконання контрольної роботи здобувачем	Надання викладачем рекомендацій щодо удосконалення проекту в рамках контрольної роботи на основі перевірки проміжних результатів її виконання.	7 тиждень модуля	Коментарі під час заняття від викладача та коментарі на платформі електронного навчання

МФО4 Перевірка та оцінювання письмових завдань	Вказівка на помилки, допущені студентом при виконанні лабораторних робіт, виявлені при перевірці звіту, завантаженого на платформу електронного навчання.	1 тиждень після здачі роботи	Платформа електронного навчання
МФО5 Тести (автоматизовані тести) для самоконтролю	Тести для самоконтролю до кожної теми	Протягом семестру	Платформа електронного навчання
МФО6 Обговорення прикладів протягом лекційних занять	Обговорення із студентами прикладів та практичних кейсів, які розглядаються протягом вивчення теми на лекційному занятті	Протягом лекційного заняття згідно графіку навчального процесу	Освітня платформа для онлайн занять

10.3 Методи підсумкового сумативного оцінювання

	Характеристика	Дедлайн, тижні	Зворотний зв'язок
МСО1 Підсумковий контроль: екзамен	Письмовий екзамен у формі тестування	Екзаменаційний тиждень	Результати екзамена на платформі онлайн-навчання
МСО2 Звіт за результатами виконання лабораторних робіт	Виконання завдань лабораторної роботи та оформлення звіту	1 тиждень	платформа онлайн-навчання, коментарі викладача під час проведення занять
МСО3 Проміжний модульний контроль у формі тестування	Тестування на платформі онлайн-навчання за теоретичним матеріалом, який вивчався протягом модуля	атестаційний тиждень 1,2 модуля	платформа онлайн-навчання
МСО4 Виконання контрольної роботи (підготовка, презентація, захист)	Розробка мобільного додатку для операційної системи Android на задану тематику в рамках виконання контрольної роботи.	7 тиждень 2 модуля	коментарі під час заняття від викладача

МСО5 Оцінювання відповідей на завдання для дискусій та обговорень	Оцінювання правильності відповідей на поставлені проблемні питання з урахуванням активності студента в обговоренні відповідей інших студентів, що приймають участь у дискусії на сторінці форуму платформи онлайн-навчання.	1 тиждень	П л а т ф о р м а онлайн-навчання
--	---	-----------	--------------------------------------

Контрольні заходи:

		Максимальна кількість балів	Мінімальна кількість балів	Можливість перескладання з метою підвищення оцінки
2 семестр		100 балів		
МСО1. Підсумковий контроль: екзамен		40		
		40	24	Ні
МСО2. Звіт за результатами виконання лабораторних робіт		16		
	16x1	16	10	Ні
МСО3. Проміжний модульний контроль у формі тестування		20		
	2x10	20	12	Ні
МСО4. Виконання контрольної роботи (підготовка, презентація, захист)		20		
	Виконання проєкту	10	9	Ні
	Представлення і захист виконаного проєкту	5	3	Ні
	Проведення модульного тестування мобільного додатку	5	3	Ні
МСО5. Оцінювання відповідей на завдання для дискусій та обговорень		4		
	4x1	4	2	Ні

Оцінювання протягом семестру проводиться у формі виконання лабораторних робіт, а також виконання і захисту контрольної роботи. Всі роботи повинні бути виконані самостійно та здані на перевірку вчасно у встановлений викладачем термін. Проміжний контроль знань проводиться наприкінці кожного модуля у формі тестування, при підготовці до якого використовуються матеріали лекцій та рекомендована література. Студент може бути допущений до іспиту, якщо він набрав мінімум 20 балів протягом семестру, і при цьому виконав 60% лабораторних завдань з дисципліни, а також написав модульні контролі з дисципліни. Частина кредитів дисципліни може бути перезарахована за умови успішного проходження масового відкритого онлайн курсу з підтвердженням персоніфікованим сертифікатом освітньої платформи. Перелік масових відкритих онлайн курсів та обсяг кредитів для перезарахування визначається лектором курсу після детального аналізу

структури та наповнення масового відкритого онлайн курсу. Наприклад, для курсу "Developing Android Apps with Kotlin" на платформі Udacity при виконанні розділів курсу можуть бути перезараховані лабораторні роботи 4 (Тема 3), 5 (Тема 4), 6 (Тема 5), 12 (Тема 11), 14 (Тема 13) загальним обсягом 5 балів.

11. Ресурсне забезпечення навчальної дисципліни

11.1 Засоби навчання

ЗН1	Мультимедіа, відео- і звуковідтворювальна, проєкційна апаратура (відеокамери, проєктори, екрани, смартдошки тощо)
ЗН2	Прикладне програмне забезпечення (Android Studio)
ЗН3	Комп'ютери, комп'ютерні системи та мережи

11.2 Інформаційне та навчально-методичне забезпечення

Основна література	
1	Laurence P.-O. Programming Android with Kotlin: Achieving Structured Concurrency with Coroutines / P.-O. Laurence, A. Hinchman-Dominguez, G. B. Meike, M. Dunn. - S.: O'Reilly, 2022. - 355 p.
2	Програмування для мобільних пристроїв : конспект лекцій для студ. денної форми навчання спеціальності 126 "Інформаційні системи та технології" / укл.: В. А. Готович, Т. В. Михайлович. — Тернопіль : Тернопільський нац. техн. ун-т ім. І. Пулюя, 2020. — 216 с.
Допоміжна література	
1	Jackson, Wallace. Android Apps for Absolute Beginners : Covering Android 7 / W. Jackson ; by Wallace Jackson. — 4th ed. 2017. — Berkeley, CA : Apress, 2017. — XXVII, 484 p.
2	Colubri, Andris. Processing for Android : Create Mobile, Sensor-Aware, and VR Applications Using Processing / A. Colubri ; by Andris Colubri. — 1st ed. 2017. — Berkeley, CA : Apress, 2017. — XXIV, 381 p.
Інформаційні ресурси в Інтернеті	
1	Масовий онлайн-курс "Програмування з нуля на Kotlin українською" [Електронний ресурс]. - Режим доступу: https://www.youtube.com/playlist?list=PL3o5sNxukLFBjYifmCcHpwXZSql5-OFB0
2	Джерело офіційної документації для розробників мобільних додатків для операційної системи Android [Електронний ресурс]. - Режим доступу: https://developer.android.com
3	W3Schools українською. Kotlin Підручник [Електронний ресурс]. – Режим доступу: https://w3schoolsua.github.io/kotlin/index
4	Масовий онлайн-курс "Developing Android Apps with Kotlin" [Електронний ресурс]. - Режим доступу: https://www.udacity.com/course/developing-android-apps-with-kotlin--ud9012

5	Дворецький М. Л. Розробка мобільних застосунків для OS Android [Електронний ресурс] : навч. посіб. / М. Л. Дворецький, Ю. О. Нездолій, С. В. Дворецька, І. О. Кандиба. – Миколаїв : Вид-во ЧНУ ім. Петра Могили, 2021. – 140 с. - Режим доступу: https://dspace.chmnu.edu.ua/jspui/handle/123456789/423
6	Кузьма К. Т. Програмування мобільних пристроїв [Електронний ресурс] : навчальний посібник для дистанційного навчання / К.Т. Кузьма. – Миколаїв: СПД Румянцева Г. В., 2021. – 128 с. - Режим доступу: http://dspace.mdu.edu.ua/jspui/handle/123456789/907
7	Jetpack Compose Tutorial Part 1 // Huawei Developers : блог про розробку мобільних додатків для операційної системи Android. URL: https://medium.com/huawei-developers/jetpack-compose-tutorial-part-1-e421fb82a834 (дата звернення: 28.07.2023).
8	Jetpack Compose Tutorial Part 2 // Huawei Developers : блог про розробку мобільних додатків для операційної системи Android. URL: https://medium.com/huawei-developers/jetpack-compose-tutorial-part-2-9f9d5cc853c6 (дата звернення: 28.07.2023).
9	Нагорний В. В. Програмування для мобільних пристроїв [Електронний ресурс] : онлайн курс / В. В. Нагорний. - Суми: СумДУ, 2023. - Режим доступу: https://mix.sumdu.edu.ua/info/nmk/b2846928-ac3e-497a-81b1-10561d2cb1c4