



Реактивне програмування

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни	
Рівень вищої освіти	Перший (бакалаврський)
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення 123 Комп'ютерна інженерія 126 Інформаційні системи та технології
Освітня програма	Інженерія програмного забезпечення комп'ютерних систем Інженерія програмного забезпечення інформаційних систем Комп'ютерні системи та мережі Інтегровані інформаційні системи Інформаційні управляючі системи та технології Інформаційне забезпечення робототехнічних систем
Статус дисципліни	вибіркова
Форма навчання	заочна
Рік підготовки, семестр	4 курс, осінній семестр
Обсяг дисципліни	4.0 кредитів ECTS (120 годин); 6 год. лекцій; 6 год. лаб. роб.; 108 год. СРС.
Семестровий контроль/контрольні заходи	Залік/Домашня контрольна робота
Розклад занять	Лекції; Лабораторні заняття Електронний розклад: https://schedule.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу/викладачів	Лектор: к.т.н., доц. кафедри ІІІ Полупан Ю.В. Juliya_polupan@i.ua Telegram: @Juliya_Polupan Лабораторні заняття: к.т.н., доц. кафедри ІІІ Полупан Ю.В.
Розміщення курсу	Посилання на дистанційний ресурс (Moodle, Google-drive, тощо): Moodle https://do.ipk.kpi.ua/course/view.php?id=6967 <u>Лекції:</u> 1) https://do.ipk.kpi.ua/course/view.php?id=6967 <u>Лабораторні роботи:</u> 1) https://do.ipk.kpi.ua/course/view.php?id=6967 <u>Базова література:</u> 1) https://do.ipk.kpi.ua/course/view.php?id=6967 Відеоконференції: 1) https://us05web.zoom.us/j/2936312006?pwd=TZ3NBf3taJj2AWd1oCpwsTnPRzCBh2.1 Посилання на курс RxJava для отримання сертифікату https://www.educative.io/courses/reactive-programming-rxjava

Програма навчальної дисципліни

1. Опис навчальної дисципліни її мета, предмет навчання та результати навчання

Опис навчальної дисципліни:

Вивчення дисципліни спрямовано на оволодіння парадигмою реактивного програмування на мові JavaScript та TypeScript з використанням найвідомішого фреймворку Angular, а також на вивчення особливостей створення асинхронних та багато потокових додатків. Студенти познайомляться з реактивним підходом до задач та навчатися створювати програми, які вібрали в себе найкращі риси цієї нової та дуже перспективної технології. Дисципліна містить докладний виклад концепцій та принципів використання реактивного програмування взагалі та RxJS зокрема.

Метою дисципліни є отримання студентами фундаментальних знань про принципи побудови і функціонування односторінкових веб-додатків (SPA), дослідження особливостей традиційних і перспективних методик веб-розробки з використанням найпоширеніших фреймворків, набуття ключових фахових компетентностей, теоретичних знань і практичних навичок з технологій веб-розробки реактивних додатків.

Предмет навчальної дисципліни – ключові структури даних, характерні для реактивного програмування, методи і засоби створення асинхронних та дієво-орієнтованих додатків, теоретичні засади та використання реактивного підходу для вирішення прикладних задач.

Завданнями дисципліни є ознайомлення студентів із реактивним підходом до розробки прикладних програм, з існуючими технологіями реактивного програмування, що використовують мови JavaScript і TypeScript та з популярним фреймворком Angular і бібліотекою RxJS.

Компетентності

Інтегральна компетентність: здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов із застосуванням теорій та методів інформаційних технологій; здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

За результатами вивчення дисципліни студент повинен набути досвід (**результати навчання**): зі створення та опрацювання асинхронних потоків даних в JavaScript; з використання бібліотеки RxJS для розробки асинхронних дієво-орієнтованих програм; з підготовки середовища розробки для створення реактивних веб-додатків на базі фреймворку Angular; з проектування та побудови модульних додатків з використанням Angular; з використання найновіших інструментів JavaScript – зокрема npm.

Студенти також повинні вміти аналізувати роботу реактивних додатків та виявляти помилки при роботі з асинхронними потоками даних та функціями зворотного виклику; вміти транслювати події користувача в асинхронний потік даних та коректно оброблювати цей потік при допомозі бібліотеки RxJS та можливостей JavaScript; вміти проектувати реактивні веб-додатки з урахуванням потреб потенційної аудиторії, аналізувати, аргументувати, приймати рішення при проектуванні, розробці, впровадженні та супроводженні таких додатків в умовах встановлених технологічних та законодавчих технічних завдань; вміти застосовувати методи компонентної розробки програмного забезпечення; знати спеціалізовані мови та технології програмування.

2. Перереквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)

Вивчення дисципліни спирається на базові знання студентів з об'єктно-орієнтованого програмування, паралельного програмування, багатопоточності, та функціонального програмування, а також на знання, уміння і навички, необхідні для раціонального використання

засобів інформаційних технологій при розв’язуванні задач опрацювання інформації, розуміння ролі інформаційних технологій, основ інформаційної культури та гігієни.

Знання та навички, набуті студентом при вивченні дисциплін, використовуються в розробці веб-додатків на їхній основі.

3. Зміст навчальної дисципліни

Тема 1	Реактивність: основні поняття. Паттерни OBSERVER та ITERATOR. Маніфест реактивності.
Тема 2	Angular: основні поняття. ChangeDetection. Стратегія Default.
Тема 3.	Основні структурні блоки Angular-додатку.
Тема 4	Концепція сигналів в Angular як сучасна модель реактивного управління станом.
Тема 5	Ключові поняття RxJS. Гарячі та холодні Observableв RxJS.

4. Навчальні матеріали та ресурси.

Основна література

1. Mateusz Dobrowolski, Mateusz Stefańczyk “The Ultimate Guide to Angular Evolution. How Each Angular Version Impacts Efficiency, DX, UX and App Performance.” Created in cooperation with angular.love.

URL: https://houseofangular.io/wp-content/uploads/2024/07/The_Ultimate_Guide_To_Angular_Evolution_Angular18.pdf

2. Реактивне програмування. Лабораторний практикум. [Електронний ресурс]: навч. посіб. для здобувачів ступеня бакалавра за освіт. програмою «Інженерія програмного забезпечення інформаційних систем» спец. 121 «Інженерія програмного забезпечення» // КПІ ім. Ігоря Сікорського; уклад.: Ю.В. Полупан – Електронні текстові дані (1 файл: 5,06 МБ). – Київ : КПІ ім. Ігоря Сікорського, 2024. – 253 с. URL: <https://ela.kpi.ua/items/ee796996-c6a1-408d-87f8-fcdb4b9cce8f>

Додаткова література

3. Freeman A. Pro Angular 16. Sixth Edition. – Shelter Island (NY): Manning Publications, March 2024. – 875 p. – ISBN 978-1-6334-3669-5.

4. Armen Vardanyan “Modern Angular. Also covers signals, standalone , ssr, zoneless , and more.” Manning Publications Co., 2025. 286 page. ISBN-13: 978-1-63343-692-3.

5. Alvaro Camillo Neto “Angular Design Patterns and Best Practices”. 1st edition. 2024 Packt Publishing. 249 page. ISBN 978-1-83763-197-1.

6. Lamis Chebbi "Reactive Patterns with RxJS and Angular Signals". 2ND edition. Published by Packt Publishing Ltd, 2024. 235 page. ISBN 978-1-83508-770-1.

7. Lamis Chebbi "Reactive Patterns with RxJS for Angular". 1ND edition. Packt Publishing, 2022. 204 page. ISBN 978-1-80181-151-4.

8. Bampakos A., Deeleman P. “Learning Angular. A practical guide to building web applications with modern Angular”. Fifth Edition. Packt Publishing, 2024. 457 page.

9. Freeman A. “Pro Angular: Build Powerful and Dynamic Web Apps”. Fifth Edition. – Shelter Island (NY): Manning Publications, March 2024. – 880 p. – ISBN-13 (electronic): 978-1-4842-8176-5.

10. Doguhan Uluca Angular для Enterprise-Ready Web Applications. -PacktPublishing, 2020

11. Valerio De Sanctis ASP.NET Core 5 and Angular: Full-stack web development with .NET 5 and Angular 11.

12. Oswald Campesato Angular and Machine Learning. - Mercury Learning and Information, 2020

13. Adam Freeman, «Pro Angular. Build Powerful and Dynamic Web Apps». Fifth Edition. Apress, 2022. DOI: <https://doi.org/10.1007/978-1-4842-8176-5>.

14. Adam Freeman, "Pro Angular". Second Edition. Apress.

Інформаційні посилання

15. Офіційний сайт Angular -<https://angular.io>
16. Сучасний підручник JavaScript. – [Електронний ресурс]. Режим доступу: <https://uk.javascript.info>
17. The State of JavaScript. – [Електронний ресурс]. – Режим доступу: <https://2019.stateofjs.com/>
18. 2020 Developer Survey-<https://insights.stackoverflow.com/survey/2020>
19. Guide of RxJS <https://rxjs.dev/guide/overview>
20. TypeScript Documentation <https://www.typescriptlang.org/docs/>
21. RxJS guide: <https://rxjs-dev.firebaseapp.com/guide/overview>

Навчальний контент

5. Методика опанування навчальної дисципліни

5.1. Структура навчальної дисципліни

Найменування розділів і тем	Розподіл по видах занять			
	Всього	Лекції	Лаб.	СРС
Реактивне програмування				
Тема 1. Реактивність: основні поняття. Паттерни OBSERVER та ITERATOR. Маніфест реактивності.	22	2		20
Тема 2. Angular: основні поняття. ChangeDetection. Стратегія Default.	26	2	2	22
Тема 3. Основні структурні блоки Angular-додатку.	24		2	22
Тема 4. Концепція сигналів в Angular як сучасна модель реактивного управління станом.	26	2	2	22
Тема 5. Ключові поняття RxJS. Гарячі та холодні Observable RxJS.	22			22
Всього	120	6	6	108

5.2. Лекційні заняття

№	Назва теми лекції та перелік основних питань	Кількість аудиторних годин
Тема 1, 5 (Лекція 1)	Реактивність: основні поняття. Паттерни OBSERVER та ITERATOR. Маніфест реактивності. Типи реактивності: push-реактивність, pull реактивність. Паттерн Rx та об'єкт Observable. Засоби обробки потоків. Ключові поняття RxJS: основні способи використання і терміни. Використання observables для передачі значень. Визначення спостерігачів. Підписка. Створення observables. Гарячі та холодні Observable в RxJS. Властивості потоку Hot та Cold observable.Subject. BehaviorSubject. ReplaySubject. AsyncSubject. Особливості Subject. Створення Subject. Початкове значення BehaviorSubject.	2
Тема 2, 3. (Лекція 2)	Angular: основні поняття. Односторінкові та багатосторінкові додатки. Історія створення фреймворку. Інструментарій Angular-розробника. Пакетний менеджер npm. Angular CLI. Створення Angular-додатку. Angular.json, package.json. ChangeDetection. Стратегія Default. Цикл виявлення змін при стратегії Default. Zone.js. OnPush стратегія виявлення змін. Зміна Input-посилання. Подія всередині компонента чи його нащадків. Ручний запуск виявлення змін. Async Pipe.	2

Тема 4. (Лекція 3)	Концепція сигналів в Angular як сучасна модель реактивного управління станом. Переваги Signal. Створення Signal. Записувані сигнали(writable). Обчислювальні сигнали computed. Сигнали Effect(). Методи Signal. Порівняння Signal та RxJS. RxJS Interop – набір утиліт для поєднання сигналів і Observables.	2
Разом годин:		6

5.3. Лабораторні роботи

№	Назва лабораторної роботи	Кількість аудиторних годин
Тема 1, 2.	Створення Angular-додатків “HelloApp” і «Shopping list». Прив'язка даних в Angular. Розгортання Angular-додатку «Shopping list» на платформі FireBase. Встановлення Node.js. Установка Angular CLI. Встановлення пакетів. Package.json. Створення компонента та модуля Angular. Визначення конфігурації. Інтерполяція. Прив'язка властивостей елементів HTML. Прив'язка до атрибуту. Прив'язка до події. Двостороння прив'язка.	2
Тема 2, 4.	Прив'язка даних в Angular. Реактивність Angular і signals. Робота з реактивними розширеннями RxJS. Створення додатку зі standalone компонентами. Створення джерела даних з декількома продуктами. Робота з продуктами при допомозі сигналів computed. Виведення повідомлень в консоль при допомозі сигналів effect(). Використання Observables з бібліотеки RxJS.	2
Тема 3, 5.	Pipe. Створення Pipes. Pure та Impure Pipes. AsyncPipe. Створення проекту «Blog». Створення необхідних компонентів. Вбудовані pipes. Передача параметрів в pipes. Фільтрація даних при допомозі pipes. Виведення поточного часу та дати при допомозі AsyncPipe. Створення компонентів post-form для створення нового поста та компонента post – для відображення існуючих постів. Передача параметрів у компонент. Створення постів. Використання директиви ng-template.	2
Разом годин:		6

6. Самостійна робота студента

До самостійної роботи студента відноситься, в основному, виконання завдання лабораторних робіт, робота з документацією, а також опрацювання лекційного та додаткового теоретичного матеріалу за наданими лекціями, літературою і інформаційними матеріалами.

№	Назви тем і питань додаткового теоретичного матеріалу, що виносяться на самостійне опрацювання та посилання на навчальну літературу	Кількість годин для самостійної роботи
ДКР	Створення проекту «SportShop». Вибір товарів та оформлення замовлення. Встановлення та налаштування засобів розробника, створення кореневих структурних блоків для проекту (модель даних, фіктивне джерело даних, репозиторій моделі, сховище, компоненти магазину та шаблони). Розширення функціональності проекту «SportShop». Реалізація підтримки кошика для вибору товарів користувачем та процесу оформлення замовлення. Заміна фіктивного	20

	джерела даних у проєкті джерелом, що надсилає запити до HTTP REST-сумісної веб-служби.	
Тема 1, 2, 3	Основні структурні блоки Angular-додатку. Створення дочірнього компонента. Передача даних між компонентами. Життєвий цикл компоненту. Вступ до директив.	5
Тема 3	Модулі в Angular. Робота з компонентами у фреймворку Angular. Standalone компоненти. Стили та, шаблони компонента. ng-content. Взаємодія між компонентами. Життєвий цикл компонента.	5
Тема 3	Директиви. Створення атрибутивних та структурних директив. Директиви ngClass та ngStyle. Взаємодія з користувачем, HostListener та HostBinding. Отримання параметрів у директивах.	5
Тема 3	Сервіси в Angular. Опціональні сервіси. Ієрархія сервісів. Впровадження сервісу в інший сервіс. Сервіси глобального та локального рівня.	5
Тема 3	Pipe. Створення Pipes. Pure та Impure Pipes. AsyncPipe. Вбудовані pipes. Параметри в pipes. Форматування дат, форматування чисел. Форматування валюти. Ланцюжки pipes.	5
Тема 3	Маршрутизація в Angular. Покрокове налаштування маршрутизації. RouterOutlet. Стилзація активних посилань. Маршрути з параметрами. Параметри рядка запиту. Вкладені маршрути. Guards. canActivate, CanDeactivate.	5
Тема 5	Мультикастинг: основні поняття, призначення, приклади використання. Multicast(), publish(),share(). Відмінності між share() та publish() + refCount(). Обробка помилок.	5
Тема 5	Schedulers в RxJS. ObserveOn(). SubscribeOn(). Event Loop. Типи RxJS Schedulers: queue, asap, async, animationFrame. Перевизначення синхронної передачі даних на асинхронну. Модульна контрольна робота.	5
Тема 1, 2, 3.	Робота з формами в Angular. Модуль FormsModule та директива NgModel. Отримання і зміна моделі. Стан моделі та валідація. Директива NgForm. Reactive Forms.	5
Тема 1,5.	HTTP та взаємодія з сервером. HttpClient та відправка запитів. Об'єкт Observableта бібліотека RxJS. Обробка помилок. Відправлення даних в запиті. POST-запити.	5
Тема 5.	Observable в бібліотека RxJS в Angular. Використання Observables для передачі значень. Спостерігаємі значення в Angular. Практичне використання спостережених значень.	5
Тема 2,3.	Впровадження залежностей в Angular. Ін'єкція залежностей в Angular. Створення служби, що інжектується. Налаштування постачальників залежностей. Ієрархічні інжектори.	5
Тема 2.	Роутінг та навігація. Angular Routing. Загальні задачі маршрутизації. Використання маршрутів Angular в одно сторінковому додатку. Створення користувальницьких відповідностей маршрута.	5
Тема 2,3.	Тестування в Angular. Тестування сервісів. Основи тестування компонентів. Сценарії тестування компонентів. Тестування директив атрибутів. Тестування пайпів. API утіліти для тестування.	5
Тема 2,3.	Бібліотеки Angular. Використання бібліотек Angular, що опубліковані на npm. Створення бібліотек. Формат пакетів Angular.	5
Тема 2,3.	Інтернаціоналізація в Angular. Загальні задачі інтернаціоналізації. Приклад інтернаціоналізації додатку Angular. Необов'язкові методи інтернаціоналізації. Імпорт глобальних варіантів даних локалі.	5

Тема 2,3.	Анімація вAngular. Переходи та тригери анімації. Складні анімаційні послідовності. Багаторазові анімації. Анімація переходу між маршрутами.	4
Тема 2,3.	Інструменти в Angular. АОТ компілятор. Компіляція з випередженням часу. Параметри компілятора Angular. Помилки метаданих АОТ. Перевірка типів шаблонів.	4
Разом годин:		108

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

- **Форма організації освітнього процесу, види навчальних занять і оцінювання результатів навчання** регламентуються Положенням про організацію освітнього процесу в Національному технічному університеті України «Київському політехнічному інституті імені Ігоря Сікорського». <https://kpi.ua/regulations>
- **Політика виставлення оцінок:** кожна оцінка виставляється відповідно до розроблених викладачем та заздалегідь оголошених студентам критеріїв, а також мотивується в індивідуальному порядку на вимогу студента; у випадку, коли студентом набирає менше 60 балів за усі передбачені навчальним планом види контролю по освітньому компоненту (лабораторні роботи, сертифікат, контрольна робота) залік студент не отримує. РСО дисципліни розроблено згідно [Положення про систему оцінювання результатів навчання](#) в КПІ ім. Ігоря Сікорського.
- **Відвідування не є обов'язковим.** Якщо студент не може бути присутнім на заняттях, він все одно несе відповідальність за опрацювання теоретичного матеріалу та виконання лабораторних завдань згідно графіку. Всі матеріали доступні студентам в середовищі Moodle платформи дистанційного навчання «Сікорський». **Відпрацювання пропущеного лабораторного заняття здійснюється шляхом самостійного виконання завдання і його захисту відповідно до розкладу або до графіку консультацій викладача.**
- **Політика академічної поведінки та доброчесності:** конфліктні ситуації мають відкрито обговорюватись в академічних групах з викладачем, необхідно бути взаємно толерантним, поважати думку іншого. Плагіат та інші форми нечесної роботи неприпустимі. Всі види завдань студент має виконувати самостійно із використанням рекомендованої літератури й отриманих знань та навичок. Цитування в письмових роботах допускається тільки із відповідним посиланням на авторський текст. Недопустимі підказки і списування у ході захисту лабораторних робіт, на контрольних роботах, на іспиті.
- **Норми академічної етики:** дисциплінованість; дотримання субординації; чесність; відповідальність; робота в аудиторії з відключеними мобільними телефонами. Повага один до одного дає можливість ефективніше досягати поставлених командних результатів. При виконанні лабораторних робіт студент може користуватися ноутбуками. Заборонено використовувати свій ноутбук чи телефон для аудіо- чи відеозапису без офіційного дозволу викладача.
- **Дотримання академічної доброчесності студентів й викладачів** регламентується кодексом честі Національного технічного університету України «Київський політехнічний інститут», положення про організацію освітнього процесу в КПІ ім. Ігоря Сікорського <https://kpi.ua/code>

8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

1. Рейтинг студента з кредитного модуля розраховується зі 100 балів, з них 60 балів складає шкала L та 40 балів шкала ДКР. До контрольних заходів з дисципліни належать захист лабораторних робіт/сертифікату та домашня контрольна робота. Шкала L протягом семестру може формуватися двома різними способами.

Спосіб 1: Шкала L протягом семестру складається з балів, що студент отримує за виконання та захист 3-ох лабораторних робіт (першої, другої та третьої ЛР $3 \cdot 20 = 60$). Як оцінюється кожна лабораторна робота дивиться таблицю 1.

Спосіб 2: Шкала L протягом семестру складається з балів, що студент отримує за захист сертифікату по курсу RxJava (<https://www.educative.io/courses/reactive-programming-rxjava>) та одної лабораторної роботи (другої ЛР). Як в цьому випадку формується шкала L дивиться таблицю 2.

Шкала ДКР складається з балів, які студент отримає за одну Домашню контрольну роботу протягом семестру, $ДКР^{\max} = 40$ балів.

2. Критерії нарахування балів L

Спосіб №1: виконання та захист всіх трьох лабораторних робіт.

2.1.1. Виконання та захист лабораторних робіт (максимальна кількість балів для $L_{1-3}^{\max} = 20$).

1) представлення звіту та додатку на хостінгу Firebase для $L_{1-3} = 10$ балів.

2) захист роботи складається із захисту звіту по роботі та із захисту додатку на хостінгу:

а) при захисті звіту виставляються бали, згідно таблиці 1, де:

* Повні відповіді на всі питання по звіту: не менше 90% потрібної інформації (повні, безпомилкові відповіді).

** Часткові відповіді на питання по звіту: достатньо повна відповідь, не менше 75% потрібної інформації або незначні неточності.

*** Задовільні відповіді на питання по звіту: неповна відповідь, не менше 60% потрібної інформації та деякі помилки (завдання виконане з певними недоліками).

**** Незадовільні відповіді на всі питання по звіту (або звіту, додатку, контрольної немає): відповідь не відповідає умовам до «задовільні відповіді».

б) при захисті додатку на хостінгу бали виставляються також, згідно таблиці 1, де:

* Повні відповіді на всі питання по додатку: не менше 90% потрібної інформації (повні, безпомилкові відповіді).

** Часткові відповіді на питання по додатку: достатньо повна відповідь, не менше 75% потрібної інформації або незначні неточності.

*** Задовільні відповіді на питання по додатку: неповна відповідь, не менше 60% потрібної інформації та деякі помилки (завдання виконане з певними недоліками).

**** Незадовільні відповіді на всі питання по додатку (або звіту, додатку, контрольної немає): відповідь не відповідає умовам до «задовільні відповіді».

Спосіб №2: отримання та захист сертифікату по курсу RxJava (<https://www.educative.io/courses/reactive-programming-rxjava>) + виконання та захист одної лабораторної роботи (2-ої лабораторної роботи). Сертифікат з курсу RxJava треба продемонструвати в системі <https://www.educative.io>.

2.2.1. Виконання та захист лабораторної роботи (максимальна кількість балів для $L_2^{\max} = 20$).

1) представлення звіту та додатку на хостінгу Firebase або в режимі відеоконференції для $L_2 = 10$ балів.

2) захист лабораторної роботи складається із захисту звіту по роботі та із захисту додатку на хостінгу або локально в режимі відеоконференції:

а) при захисті звіту виставляються бали, згідно таблиці 2, де:

* Повні відповіді на всі питання по звіту: не менше 90% потрібної інформації (повні, безпомилкові відповіді).

** Часткові відповіді на питання по звіту: достатньо повна відповідь, не менше 75% потрібної інформації або незначні неточності.

*** Задовільні відповіді на питання по звіту: неповна відповідь, не менше 60% потрібної інформації та деякі помилки (завдання виконане з певними недоліками).

**** Незадовільні відповіді на всі питання по звіту (або звіту, додатку, контрольної немає): відповідь не відповідає умовам до «задовільні відповіді».

б) при захисті додатку на хостінгу або локально в режимі відеоконференції бали виставляються також, згідно таблиці 2, де:

* Повні відповіді на всі питання по додатку: не менше 90% потрібної інформації (повні, безпомилкові відповіді).

** Часткові відповіді на питання по додатку: достатньо повна відповідь, не менше 75% потрібної інформації або незначні неточності.

*** Задовільні відповіді на питання по додатку: неповна відповідь, не менше 60% потрібної інформації та деякі помилки (завдання виконане з певними недоліками).

**** Незадовільні відповіді на всі питання по додатку (або звіту, додатку, контрольної немає): відповідь не відповідає умовам до «задовільні відповіді».

2.2.2.Отримання та захист сертифікату по курсу RxJava (<https://www.educative.io/courses/reactive-programming-rxjava>). Мінімальна кількість балів для $L_5^{\min}=20$ (тільки за наявність сертифікату в системі Мудл та в системі <https://www.educative.io>) максимальна кількість балів для $L_5^{\max}=40$).

1) наявність сертифікату в системі Мудл (<https://do.ipk.kpi.ua/> та в системі <https://www.educative.io>) $L_5^{\min}$: 20 балів;

2) захист сертифікату складається із відповідей на питання по курсу RxJava та лекційному матеріалу. При цьому студент повинен знаходитись в системі <https://www.educative.io/courses/reactive-programming-rxjava>, повинен показати в системі сертифікат, та дати відповіді на питання:

а) при захисті сертифікату виставляються бали, згідно таблиці 2, де:

* Повні відповіді на всі питання: не менше 90% потрібної інформації (повні, безпомилкові відповіді).

** Часткові відповіді на питання: достатньо повна відповідь, не менше 75% потрібної інформації або незначні неточності.

*** Задовільні відповіді на питання: неповна відповідь, не менше 60% потрібної інформації та деякі помилки (завдання виконане з певними недоліками).

**** Незадовільні відповіді на всі питання (або сертифікату немає в системі Мудл): відповідь не відповідає умовам до «задовільні відповіді».

2.3. За участь у виконанні завдань із удосконалення дидактичних матеріалів з дисципліни: від 1 до 5 заохочуваних балів.

Кількість заохочуваних балів не може перевищувати 6 (10% від шкали L).

3. Умовою отримання заліку є:

- для способу 1 - набір не менше ніж 60 балів по всім видам контролю $\sum_{i=1}^3 L_i + \text{ДКР} \geq 60$.
- для способу 2 - набір не менше ніж 60 балів по всім видам контролю $L_5 + L_2 + \text{ДКР} \geq 60$.

При цьому, бали по всім видам контролю отримувати не обов'язково, незалежно від того, який спосіб для нарахування балів (спосіб 1 чи спосіб 2) студент обрав. Для заліку достатньо мати 60 балів.

4. Сума всіх балів $L_i + \text{ДКР}$ переводиться до залікової оцінки згідно з таблицею 3:

Таблиця 3.

Бали ($L_i + \text{ДКР}$): лабораторні роботи/сертифікат + контрольна робота	Оцінка
95...100	Відмінно
85...94	Дуже добре
75...84	Добре
65...74	Задовільно
60...64	Достатньо
Менше 60	Незадовільно

Таблиця 1. Розподіл рейтингових балів при умові вчасного виконання відповідного виду контролю (Спосіб 1)

№	Вид контролю	Оцінюван ня звіту на платформі Мудл без участі студента (бали) R ₁	Перевірка роботи додатку на платформі Firebase або локально без участі студента (бали)		Захист додатку на Firebase або локально (відповіді на питання в зумі), бали R ₃				Захист звіту (відповіді на питання в зумі) R ₄				Контрольна робота Додаток+звіт R ₅				Мін. кількість балів для отримання заліку (за наявність)	Макс. кількість балів (із захистом)
					(*)	(**)	(***)	(****)	(*)	(**)	(***)	(****)	(+ (!)	(++ (!!)	(+++ (!!!)	(++++ (!!!!) або контрольно ї немає		
1	Лаб. робота №1 (L ₁)	(!) – 5 (!!) – 3 (!!!) – 2 (!!!!) – 1	Laba1_1: (+) 2,5 (++) 2 (+++) 1,5 (++++) 1	Laba1_2: (+) 2,5 (++) 2 (+++) 1,5 (++++) 1	5	3	2	0	5	3	2	0					L ₁ ^{min} : 10	L ₁ ^{max} : 20
2	Лаб. робота №2 (L ₂)	(!) – 5 (!!) – 3 (!!!) – 2 (!!!!) – 1	Laba2_1: (+) 2,5 (++) 2 (+++) 1,5 (++++) 1	Laba2_2: (+) 2,5 (++) 2 (+++) 1,5 (++++) 1	5	3	2	0	5	3	2	0					L ₂ ^{min} : 10	L ₂ ^{max} : 20
3	Лаб. робота №3 (L ₃)	(!) – 5 (!!) – 3 (!!!) – 2 (!!!!) – 1	Laba3_1: (+) 2,5 (++) 2 (+++) 1,5 (++++) 1	Laba3_4: (+) 2,5 (++) 2 (+++) 1,5 (++++) 1	5	3	2	0	5	3	2	0					L ₃ ^{min} : 10	L ₃ ^{max} : 20
4	Домашня контрольна робота (ДКР)												40	30	20	0	ДКР ^{min} : 30	ДКР ^{max} : 40
5	Загалом:																60	100
6	Мінімальна оцінка по даному критерію	3*5=15	3*5=15											30			60	
7	Максимальна оцінка по даному критерію	3*5=15	3*5=15		15				15				40					100

(*) - Повні відповіді на всі питання при захисті

(**) - Часткові відповіді на питання при захисті

(***) - Задовільні відповіді на питання при захисті

(****) - Незадовільні відповіді на всі питання при захисті

- (+) – Повністю працюючий додаток. Виконані всі завдання лабораторної роботи.
- (++) – Додаток працює на хостингу (або локально). Помилки при роботі додатку немає, але частина функціоналу відсутня або не працює.
- (+++)- Додаток працює на хостингу (або локально), але частина функцій недоступна через помилку, частина сторінок не завантажується.
- (++++)- Під час старту додатка виникає помилка, або додатку немає.
- (!) – Звіт оформлено згідно вимог. В звіті дано повні відповіді на всі питання.
- (!!) - Звіт оформлено згідно вимог. Часткові відповіді на питання.
- (!!!) – Звіт оформлено не за вимогами. Задовільні відповіді на питання.
- (!!!!) - Звіт оформлено не за вимогами. Незадовільні відповіді на всі питання.

Таблиця 2. Розподіл рейтингових балів при умові вчасного виконання відповідного виду контролю (Спосіб 2)

№	Вид контролю	Оцінювання звіту на платформі Мудл без участі студента (бали) R ₁ (за наявність)	Перевірка роботи додатку на платформі Firebase або локально без участі студента (бали) (за наявність)		Захист додатку на Firebase або локально (відповіді на питання в зумі), бали R ₃				Захист звіту або сертифікату (відповіді на питання в зумі) R ₄				Контрольна робота Додаток+звіт R ₅				Мін. кількість балів для отримання заліку (за наявність)	Макс. кількість балів (із захистом)
					(*)	(**)	(***)	(****)	(*)	(**)	(***)	(****)	(*) (!)	(**) (!!)	(***) (!!!)	(****) (!!!!) або контрольної немає		
1	Сертифікат з курсу RxJava (L _s)	наявність сертифікату в Мудл: 20 балів							40	30	20	0					L _s ^{min} : 20	L _s ^{max} : 40
2	Лаб. робота №2 (L ₂)	(!) – 5 (!!) – 3 (!!!) – 2 (!!!!) – 1	Laba2_1: (+) 2,5 (++) 2 (+++) 1,5 (+++++) 1	Laba2_2: (+) 2,5 (++) 2 (+++) 1,5 (+++++) 1	5	3	2	0	5	3	2	0					L ₂ ^{min} : 10	L ₂ ^{max} : 20
3	Домашня контрольна робота (ДКР)												40	30	20	0	ДКР ^{min} : 30	ДКР ^{max} : 40
4	Загалом:																60	100
5	Мінімальна оцінка по даному критерію	20+5=25	5											30			60	
6	Максимальна оцінка по даному критерію	5	5		5				45				40					100

Робочу програму навчальної дисципліни (силабус):

Складено доц. кафедри ІПІ, к.т.н., доц., Полупан Ю.В.

Ухвалено кафедрою ІПІ (протокол № 2/1 від 10.10.2025р.)

Погоджено Методичною комісією факультету (протокол № 3 від 17.10.2025р.)