

РІВНЕНСЬКИЙ ДЕРЖАВНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ

Факультет математики та інформатики

Кафедра інформаційно-комунікаційних технологій та
методики викладання інформатики

«До захисту допущено»

Завідувач кафедри

_____ Войтович І.С.

«__» _____ 20__ р.

протокол № _____

СОЛОВІЙОВ МИКОЛА ДМИТРОВИЧ

КВАЛІФІКАЦІЙНА РОБОТА

**РОЗРОБЛЕННЯ ЕЛЕКТРОННОГО ОСВІТНЬОГО РЕСУРСУ З
ПРОГРАМУВАННЯ МОБІЛЬНИХ ЗАСТОСУНКІВ ДЛЯ ПІДГОТОВКИ
МАЙБУТНІХ ФАХІВЦІВ З ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

015 Професійна освіта (за спеціалізаціями)
015.39 Професійна освіта (Цифрові технології)

Подається на здобуття другого (магістерського) рівня вищої освіти
Кваліфікаційна робота містить результати власних досліджень і немає
запозичень з праць інших авторів без відповідних посилань

(підпис)

(прізвище, ініціали)

Науковий керівник: Шроль Тетяна Степанівна, доцент кафедри
інформаційно-комунікаційних технологій та методики викладання
інформатики., кандидат педагогічних наук

АНОТАЦІЯ

Соловійов М.Д. Розроблення електронного освітнього ресурсу з програмування мобільних застосунків для підготовки майбутніх фахівців з інформаційних технологій. Кваліфікаційна робота на здобуття освітнього ступеня магістра за спеціальністю 015,39 Професійна освіта (Цифрові технології). Рівненський державний гуманітарний університет, Рівне, 2023. 98 с.

Кваліфікаційна робота прив'язана особливостям розробки освітнього ресурсу з програмування мобільних застосунків для підготовки майбутніх фахівців з інформаційних технологій у закладах професійно-технічної та фахової передвищої освіти та вищої освіти.

У ході виконання кваліфікаційної роботи було здійснено аналіз сучасних освітніх ресурсів із розробки мобільних застосунків, їх зміст та структуру, відповідність освітнім програмам підготовки майбутніх фахівців з ІТ-технологій.

Описано переваги та недоліки використання освітніх ресурсів. Визначено вимоги до освітніх ресурсів, які слід враховувати під час їх розробки, для навчання програмуванню мобільних застосунків майбутніх ІТ-фахівців. Зазначено, що при розробці освітнього контенту (лекційних, лабораторних, тестів тощо), зокрема під час формулювання змісту практичних завдань та інструкцій до їх виконання слід орієнтуватися на вимоги, що ставляться перед розробниками на всіх етапах розробки мобільних застосунків.

В результаті у відповідності до змісту програми дисципліни «Програмування мобільних пристроїв» для здобувачів спеціальності 015.39 Професійна освіта (Цифрові технології) було розроблено та впроваджено електронний освітній ресурс засобами платформи Moodle (відео, текстові та графічні матеріали, тести тощо) для розробки кросплатформних мобільних застосунків засобами Flutter.

Ключові слова: електронний освітній ресурс, мобільний застосунок, майбутній ІТ-фахівець

ABSTRACT

Solovyov M. D. Development of an electronic educational resource on mobile application programming for training future information technology specialists. Qualifying work of the master on a specialty 015.39 Vocational Education (Digital Technologies). Rivne State University for the Humanities, Rivne, 2023. 98 p.

The qualification work is devoted to the peculiarities of developing an educational resource on mobile application programming for the training of future IT specialists in vocational and professional higher education and higher education institutions.

In the course of the qualification work, the author analysed modern educational resources for the development of mobile applications, their content and structure, and their compliance with educational programmes for the training of future IT specialists.

The advantages and disadvantages of using educational resources are described. The requirements for educational resources that should be taken into account when developing them for teaching mobile application programming to future IT specialists are determined. It is noted that when developing educational content (lectures, labs, tests, etc.), in particular when formulating the content of practical tasks and instructions for their implementation, one should be guided by the requirements set for developers at all stages of mobile application development.

As a result, in accordance with the content of the discipline "Mobile Device Programming" for applicants of speciality 015.39 Vocational Education (Digital Technologies), an electronic educational resource was developed and implemented using the Moodle platform (videos, text and graphic materials, tests, etc.) for the development of cross-platform mobile applications using Flutter.

Keywords: electronic educational resource, mobile application, future IT specialist

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	5
ВСТУП.....	6
РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ІСНУЮЧИХ ЕЛЕКТРОННИХ ОСВІТНІХ РЕСУРСІВ ЩОДО РОЗРОБКИ МОБІЛЬНИХ ЗАСТОСУНКІВ ДЛЯ РІЗНИХ ПЛАТФОРМ.....	9
1.1. Теоретичні аспекти використання електронних освітніх ресурсів у підготовці майбутніх фахівців з інформаційних технологій.....	9
1.2. Освітні платформи та ресурси з розробки мобільних застосунків.....	15
1.3. Аналіз та обґрунтування вибору засобів з розробки освітнього ресурсу.....	18
РОЗДІЛ 2. ТЕОРЕТИКО-МЕТОДИЧНІ АСПЕКТИ ПРОЕКТУВАННЯ ТА РОЗРОБКИ ЕОР З ПРОГРАМУВАННЯ МОБІЛЬНИХ ЗАСТОСУНКІВ	26
2.1. Основні етапи розробки електронних освітніх ресурсів	26
2.2. Особливості проектування та створення ЕОР із розробки мобільних застосунків	29
2.3. Технології розробки мобільних застосунків	33
2.4. Методичні аспекти підготовки майбутній фахівців з інформаційних технологій до розробки мобільних застосунків.....	36
РОЗДІЛ 3. ВПРОВАДЖЕННЯ ЕОР В ОСВІТНІЙ ПРОЦЕС ПІДГОТОВКИ БАКАЛАВРІВ СПЕЦІАЛЬНОСТІ ПРОФЕСІЙНА ОСВІТА (ЦИФРОВІ ТЕХНОЛОГІЇ).....	43
3.1. Структура ЕОР з розробки мобільних застосунків	43
3.2. Методичні рекомендації щодо використання ЕОР з розробки мобільних застосунків	46
ВИСНОВКИ.....	52
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	54
ДОДАТКИ.....	61

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ЕОР	–	електронний освітній ресурс
ТЗ	–	технічне завдання
CMS	–	система управління контентом
LMS	–	система управління навчанням
HTML	–	мова гіпертекстової розмітки
CSS	–	каскадні таблиці стилів

ВСТУП

Швидкий темп розвитку інформаційних технологій (ІТ) визначає важливість для закладів вищої освіти залишатися в тренді, бути гнучкими, швидко орієнтуватися до змін в ІТ-сфері. Зокрема, адаптувати й змінювати зміст освітніх та робочих програм, розробляти електронні освітні ресурси (ЕОР), контент яких можна швидко змінити в залежності від розвитку тих чи інших технологій.

Освітня онлайн-платформа – це засіб навчання, який дозволяє зберігати і переглядати відеозаписи уроків, дає студенту можливість повернутися до пройденого матеріалу і повторити вивчене. Надає змогу переглядати навчальний матеріал у зручний час і слідкувати за своїм прогресом. Найчастіше в закладах освіти використовують пакет хмарних сервісів G Suite for Education, портал Microsoft 365 for Education, EdEra, edX, Coursera, Щоденник.ua тощо, які надають багатий функціонал для організації освітньої діяльності та взаємодії всіх учасників освітнього процесу.

Більшість таких платформ надають хороші ЕОР у форматі курсів з різних напрямків ІТ, зокрема із розробки мобільних застосунків, але найчастіше на платній основі. Крім того, не завжди зміст, контент та структура таких курсів відповідають освітній програмі, відповідно навчальному плану та робочим програмам освітньої компоненти (дисципліни) закладу вищої освіти (ЗВО). Тому одним із важливим завдань, виконання якого сприятиме якісній підготовці майбутніх ІТ-фахівців, є створення якісного освітнього ресурсу, контент якого відповідає сучасним тенденціям розвитку мобільних технологій, та містить професійно-орієнтовані завдання, що передбачають виконання здобувачем всіх етапів розробки мобільних застосунків: від постановки технічного завдання, проектування, розробки до розміщення в магазинах мобільних застосунків.

Актуальність виявленої проблеми зумовили вибір теми кваліфікаційної роботи «Розроблення електронного освітнього ресурсу з програмування

мобільних застосунків для підготовки майбутніх фахівців з інформаційних технологій».

Об'єкт дослідження – підготовка майбутніх ІТ фахівців.

Предмет дослідження – електронні освітні ресурсу з програмування мобільних застосунків для підготовки майбутніх фахівців з інформаційних технологій.

Мета дослідження – розглянути сучасні технології проектування і розробки освітніх електронних ресурсів та розробити ЕОР з програмування мобільних застосунків для підготовки майбутніх фахівців з інформаційних технологій.

Відповідно до мети роботи визначені такі основні *завдання*:

- дослідження, аналіз та систематизація інформації відповідно до тематики кваліфікаційної роботи в науковій, науково-методичній літературі та практиці;
- дослідження та аналіз існуючих електронних освітніх ресурсів щодо розробки мобільних застосунків для різних платформ;
- аналіз та обґрунтування вибору засобів з розробки освітнього ресурсу
- підготовка освітнього контенту у відповідності до змісту програми дисципліни «Програмування мобільних застосунків» для здобувачів спеціальності 015.39 Професійна освіта (Цифрові технології);
- структурування та розміщення матеріалу в одній із систем LMS/CMS\$
- впровадження електронного освітнього ресурсу в освітній процес з підготовки фахівців з інформаційних технологій на прикладі підготовки бакалаврів спеціальності 015.39 Професійна освіта (Цифрові технології).

Результати дослідження доповідалися на звітній науково-практичній конференції співробітників, студентів та аспірантів РДГУ, а також на XVI Всеукраїнській науково-практичній конференції «Інформаційні технології в професійній діяльності» (м. Рівне, 1 листопада 2023 року) (додаток А). За результатами опубліковано тези конференцій (Соловйов&Шроль, 2023, С.195-197).

Практична значущість результатів роботи полягає в тому, щоб використовувати результати роботи в підготовці фахівців із інформаційних технологій, зокрема здобувачів вищої освіти спеціальності 015 «Професійна освіта (Цифрові технології)» під час розробки ними мобільних застосунків.

Магістерська робота складається з вступу, трьох розділів, висновків, списку використаних джерел (52 найменувань), 8 додатків.

РОЗДІЛ 1

ДОСЛІДЖЕННЯ ТА АНАЛІЗ ІСНУЮЧИХ ЕЛЕКТРОННИХ ОСВІТНІХ РЕСУРСІВ ЩОДО РОЗРОБКИ МОБІЛЬНИХ ЗАСТОСУНКІВ ДЛЯ РІЗНИХ ПЛАТФОРМ

1.1. Теоретичні аспекти використання електронних освітніх ресурсів у підготовці майбутніх фахівців з інформаційних технологій

Для реузальтативної підготовки майбутніх фахівців із ІТ технологій важливим є використання в освітньому процесі якісних освітніх ресурсів, зміст та відповідний контент яких буде відповідати професійним завданням та відображатиме специфіку їх майбутньої професійної діяльності.

Проблеми щодо розробки та впровадження ЕОР в освітній процес закладів вищої освіти досить ґрунтовно розкрито у працях, як вітчизняних так і зарубіжних вчених. Можна викоремити дослідження, в яких розкрито:

- загальні аспекти використання електронних освітніх ресурсів, спрямовані на різні предметні області (Т.Я. Грановська) та навчання онлайн (В.Е.Зайцев, К.С. Бабко);
- вплив технологій та відповідних електронних ресурсів на результати навчання та якість навчання (С.О. Карплюк, Д. С. Вербівський);
- педагогічні та методичні аспекти використання електронних ресурсів в освітньому процесі (О. Кучерук, Т. Магдич, С. Крупко, О.Е. Колмакова), зокрема на базі SMART-технологій (О.В. Клочко, О.Д. Гуменний, В.І. Клочко) та для організації STEAM-навчання (Т. Шарова, С. Шаров, Б. Кременський);
- використання штучного інтелекту для підтримки навчання та персоналізації електронних освітніх ресурсів (І. Візнюк, Н. Буглай, Л. Куцак, А. Поліщук, В. Киливник тощо);

- технологічні особливості розробки електронних освітніх ресурсів та систем електронного навчання (М.А. Бойко, А. Ломанченко, О. Хлопонін, Т. Деркач, Т. Дмитренко та ін);
- особливості проектування, розробки та використання в освітньому процесі імерсивних освітніх ресурсів (В.О. Колмакова (2020), Н. Сороко, О. Гаєвська та ін.);
- методи та критерії для оцінки ефективності та якості створених різнотипних електронних освітніх ресурсів та курсів (Н.В. Морзе, О.Г. Глазунова, М.В. Мокрієв, В. Нерсесов, С.Г. Литвинова тощо);
- використання мобільних застосунків в освітньому процесі (С.В. Лазаренко, Д. В. Борисенко) та платформ й мобільних додатків для створення і використання контенту із технологією доповненої реальності в освітньому процесі (Д. В. Мацокін, І.М. Пахомова тощо).

Відповідно до наказу «Про затвердження Положення про електронні освітні ресурси» електронні освітні ресурси визначаються, як «засоби навчання на цифрових носіях будь-якого типу або розміщені в інформаційно-телекомунікаційних системах, які відтворюються за допомогою електронних технічних засобів і застосовуються в освітньому процесі» (Про затвердження Положення про електронні освітні ресурси: наказ МОН від 01.10.2012 № 1060)

У визначеному положенні відображено вимоги до електронного підручника визначені Положенням про електронний підручник (затвердженим наказом Міністерства освіти і науки України від 02 травня 2018 року № 440) та до дистанційних курсів – Положенням про дистанційне навчання (затвердженим наказом Міністерства освіти і науки України від 25 квітня 2013 року № 466). «Метою створення ЕОР є забезпечення модернізації освітнього процесу, змістове наповнення освітнього простору, надання рівного доступу учасникам освітнього процесу незалежно від місця їх проживання та форми навчання відповідно до якісних навчальних і методичних матеріалів, створених на основі інформаційно-комунікаційних технологій».

До основних видів ЕОР відповідно до Положення про ЕОР (2012) віднесено: електронний документ, електронне видання, електронний аналог друкованого видання, електронні дидактичні демонстраційні матеріали, депозитарій електронних ресурсів, курс дистанційного навчання, електронний лабораторний практикум, комп'ютерний тест, електронний освітній ігровий ресурс, електронний словник, електронний довідник, електронний навчальний посібник, електронний підручник, електронні методичні матеріали.

На рис.1.1 вказано класифікацію ЕОР, що згруповані за функціональною ознакою їх використання відповідно до етапів організації освітнього процесу відповідно до визначеного положення та досліджень О.Е. Коневщинської (О.Е. Коневщинська 2014, с.169).

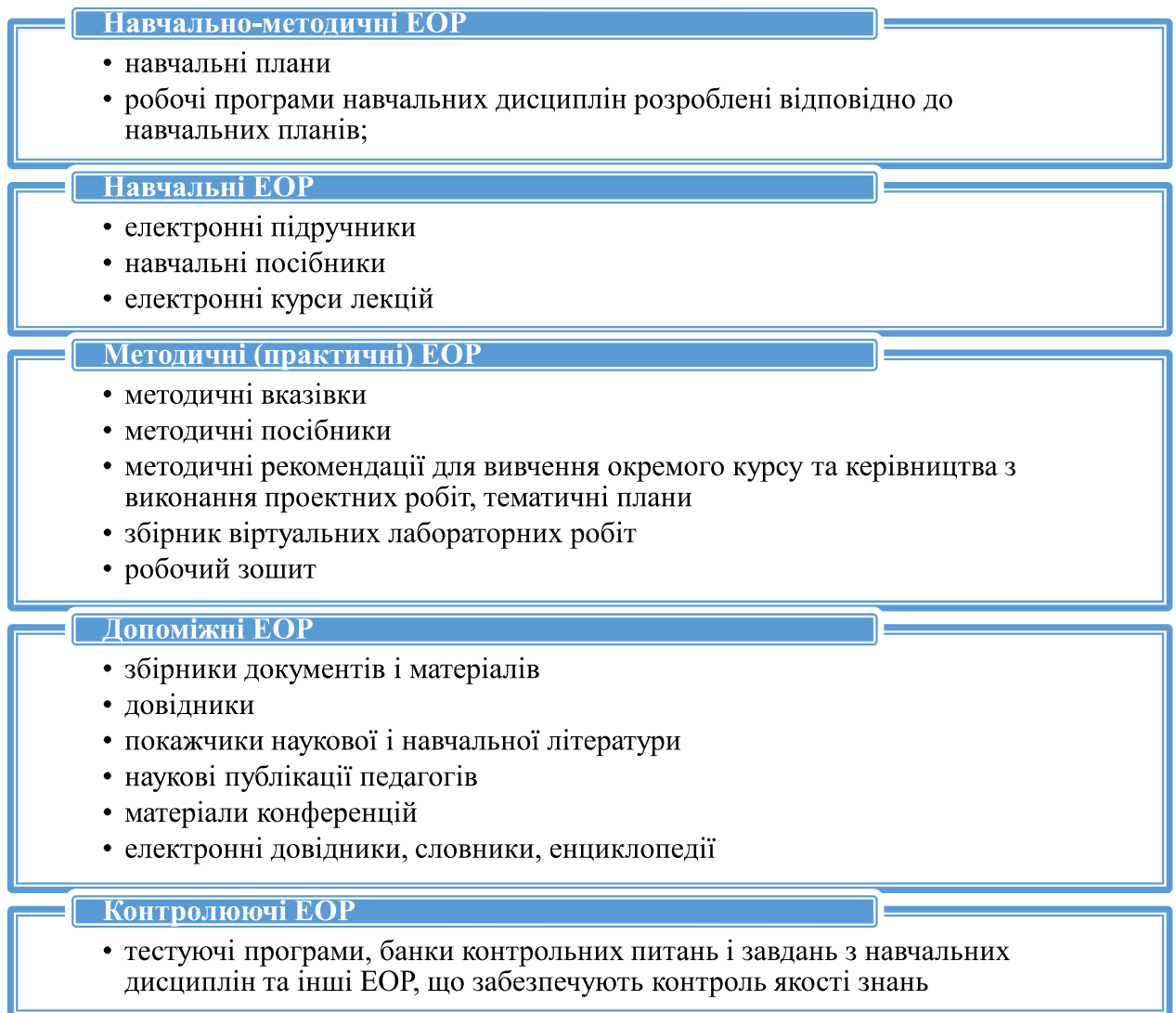


Рис. 1.1. Класифікація ЕОР за функціональною ознакою їх використання в освітньому процесі

Існують і інші класифікації. Так група авторів М.І. Женченко, О.М. Мельник, І.В. Женченко, Я.В. Прихода (2020) запропонували фасетну класифікацію ЕОР (табл.Б.1, додаток Б), що визначаються такими ознаками:

- 1) *загальні*: мова, структура
- 2) *освітні*: напрям використання, функції в освітньому процесі, кінцеві користувачі (аудиторія), реалізація ігрового складника, форма (рівень) освіти;
- 3) *видавничо-технологічні*: наявність друкованого еквівалента, наявність редакційно-видавничого опрацювання, розробник, технологія розроблення, технологія використання;
- 4) *технічні*: форма існування, подання даних, спосіб використання даних, тип подання і природа основної інформації, характер взаємодії з користувачем, формат, залежність від операційних систем (мобільних платформ), стабільність;
- 5) *правові*: обмеженість доступу, тип ліцензії.

Використання електронних освітніх ресурсів в підготовці майбутніх фахівців з інформаційних технологій має величезний потенціал для покращення якості освіти і підвищення ефективності навчання.

До *переваг* використання ЕОР в освітньому процесі можна віднести наступні:

- *Доступ до актуальної інформації*. Електронні ресурси дозволяють студентам отримувати доступ до найновішої інформації з галузі інформаційних технологій. Інтернет, онлайн-курси, електронні книги та статті надають можливість ознайомлюватися з останніми технологічними тенденціями та розробками.
- *Інтерактивність*. Електронні ресурси можуть бути спроектовані з використанням інтерактивних елементів, таких як відеоуроки, інтерактивні завдання та віртуальні лабораторії. Це забезпечує залучення студентів та покращує їх розуміння складних концепцій.

- *Онлайн-спільноти та співпраця.* Використання електронних платформ дозволяє створювати онлайн-спільноти, де студенти можуть обговорювати питання, співпрацювати над проектами та ділитися знаннями. Це стимулює колективне навчання та обмін досвідом.
- *Адаптивне навчання.* Електронні платформи можуть використовувати технології штучного інтелекту для адаптації навчального процесу до індивідуальних потреб студентів. Це може покращити ефективність навчання та допомогти студентам з різним рівнем підготовки.
- *Оцінювання та звітність.* Електронні системи можуть автоматизувати процес оцінювання та надавати студентам миттєвий звіт про їхні успіхи. Це сприяє зрозумінню та виправленню помилок, дозволяючи студентам ефективніше навчатися.
- *Гнучкість навчання.* Електронні ресурси дозволяють студентам навчатися власним темпом та в будь-який зручний для них час. Це особливо важливо для тих, хто має робочий графік чи інші обмеження.
- *Віддалене навчання.* Застосування електронних ресурсів дозволяє здійснювати віддалене навчання, що особливо актуально у сучасних умовах, коли зміни в робочому середовищі можуть обмежувати фізичну присутність студентів на заняттях.

Хоча використання ЕОР у підготовці ІТ-фахівців має безліч переваг, існують також деякі *недоліки* та *виклики*, які можуть виникати:

- *Відсутність особистої взаємодії.* Електронні ресурси можуть не забезпечити той рівень особистої взаємодії, який може бути важливим у підготовці ІТ-фахівців, особливо в аспекті колективної роботи, обговорень та комунікації.
- *Нестабільний доступ до Інтернету.* Залежність від електронних ресурсів передбачає наявність стійкого та швидкого Інтернет-з'єднання, що може бути проблематичним в деяких регіонах або для студентів із обмеженим доступом.

- *Недостатня мотивація та саморегуляція.* Деякі студенти можуть мати труднощі у саморегуляції та підтриманні мотивації без прямого нагляду вчителя чи наставника.
- *Відсутність практичних навичок.* Онлайн-курси можуть не завжди надавати достатньо можливостей для отримання практичних навичок та роботи з реальними завданнями, що може бути важливим для підготовки ІТ-спеціалістів.
- *Брак інтерактивності та практичних вправ.* Деякі теми, особливо ті, що вимагають інтерактивності та практичних вправ, можуть бути важко передати через електронні ресурси.
- *Проблеми з безпекою та конфіденційністю.* У разі поганої безпеки електронних платформ може виникнути ризик витоку конфіденційної інформації чи атак на систему.
- *Специфічність предметів.* Деякі аспекти ІТ-освіти, такі як робота з апаратним забезпеченням, можуть бути складніше вивчати в онлайн-форматі, оскільки вимагають фізичного доступу до обладнання.

Загалом, використання ЕОР у підготовці фахівців з інформаційних технологій може сприяти покращенню якості навчання та готувати їх до викликів майбутньої професії та швидко змінюючогося технологічного світу. А для подолання вище перерахованих недоліків, важливо розробляти ефективні стратегії викладання в умовах змішаного та дистанційного навчання, використовувати різноманітні підходи та методи до побудови освітнього середовища та відповідних ресурсів у формі текстових, аудіо, відео матеріалів та інтегрувати практичні завдання в програми навчання. Також важливо враховувати індивідуальні потреби студентів та надавати можливості для взаємодії та обговорення.

Зазначимо, ЕОР включають в себе будь-які матеріали, які можна використовувати для навчання, у формі відеолекцій, інтерактивних вправ, електронних підручників, віртуальних лабораторій тощо. Вони часто інтегруються в системи управління контентом (CMS) або системи управління

навчанням (CLM). Останні ще називають платформами електронного навчання (e-learning platforms).

Основні характеристики електронних освітніх ресурсів:

- мультимедійний контент: відео, аудіо, графіка для більш ефективного навчання;
- інтерактивність: можливість взаємодії та залучення здобувачів до навчально-пінавальної діяльності;
- адаптивність: можливість адаптувати освітній контент до індивідуальних потреб та темпу навчання здобувачів.

1.2. Освітні платформи та ресурси з розробки мобільних застосунків

Існує безліч електронних освітніх ресурсів, які можна використовувати для вивчення різних предметів та навичок. Нижче наведено перелік кількох популярних електронних освітніх ресурсів, які допомагають в освоєнні нових знань та вдосконаленні вмінь:

- *Coursera* (<https://www.coursera.org/>): надає онлайн-курси від університетів та організацій з усього світу. Має широкий спектр предметів, включаючи науку, техніку, гуманітарні науки та інші.
- *edX* (<https://www.edx.org/>): платформа для навчання онлайн, яка пропонує курси від університетів та коледжів. Здійснюється спільно з MIT, Harvard та іншими великими установами.
- *Udacity* (<https://www.udacity.com/>): фокусується на курсах з технічних та комп'ютерних наук, включаючи програмування, розробку веб-сайтів та штучний інтелект.
- *Khan Academy* (<https://www.khanacademy.org/>): надає безкоштовні уроки з різних предметів для учнів усіх вікових груп. Є особливий акцент на математиці та науці.

- *Codecademy* (<https://www.codecademy.com/>): спеціалізується на навчанні програмування та веб-розробці шляхом інтерактивних вправ та проєктів.
- *LinkedIn Learning* (<https://www.linkedin.com/learning/>): має широкий вибір курсів з бізнесу, технічних навичок та творчості. Представляється як платформа для професійного розвитку.
- *Udemy* (<https://www.udemy.com/>): онлайн-платформа, де викладачі можуть створювати та пропонувати свої власні курси. Має різноманітні тематики.
- *MIT OpenCourseWare* (<https://ocw.mit.edu/index.htm>): MIT надає безкоштовний доступ до матеріалів більшості своїх курсів, включаючи лекції, завдання та інші ресурси.
- *Google Classroom* (<https://classroom.google.com/>): платформа для організації віддаленого навчання та обміну матеріалами між вчителями та учнями.
- *Duolingo* (<https://www.duolingo.com/>): онлайн-платформа для вивчення мов за допомогою ігор та вправ.

Ці ресурси можуть бути використані для навчання різних предметів, включаючи програмування, мови, науку, математику та багато іншого. Залежно від ваших інтересів та навичок, ви можете знайти відповідний ресурс для свого самонавчання чи професійного розвитку.

Існує багато електронних освітніх ресурсів, призначених для програмістів, які хочуть вивчати розробку мобільних застосунків. Зокрема, ресурси для вивчення програмування мобільних застосунків поділяються на кілька категорій:

1. *Онлайн-курси:*

- *Coursera* (<https://www.coursera.org/>): Пропонує курси від університетів і компаній з розробки мобільних застосунків, такі як "Android App Development" від Google та "iOS App Development" від Apple.

- *edX* (<https://www.edx.org/>): Подібно до Coursera, edX пропонує курси від університетів та організацій, які охоплюють різні аспекти розробки мобільних застосунків.

2. Офіційні документації та ресурси від виробників платформ:

- *Android Developers* (<https://developer.android.com/>): Офіційний сайт розробників Android, який містить документацію, навчальні матеріали та приклади коду;
- *Apple Developer Documentation* (<https://developer.apple.com/documentation/>): Офіційна документація для розробників iOS та macOS від Apple;
- *Guid React Native* (<https://reactnative.dev/docs/getting-started>)
- *Flutter/Dart*

3. Інтерактивні платформи для вивчення програмування:

- *Udacity* (<https://www.udacity.com/>): Пропонує курси та проекти з розробки мобільних застосунків, розроблені спільно з компаніями-партнерами, такими як Google та Facebook;
- *Codecademy* (<https://www.codecademy.com/>): Надає інтерактивні курси з різних мов програмування, включаючи Java для розробки Android-застосунків та Swift для розробки iOS-застосунків.
- *Udemy* (<https://udemy.com/>): надає доступ до понад 210 000 онлайн-відеокурсів, а нові курси додаються на сайт щомісяця.

4. YouTube-канали та відеоуроки:

- *The Net Ninja* (<https://netninja.dev/>): Має велику кількість безкоштовних відеоуроків з розробки мобільних застосунків для платформ Android та iOS;
- *Traversy Media* (<https://www.traversymedia.com/>): Пропонує відеоуроки з різних аспектів веб-розробки, включаючи розробку мобільних застосунків з використанням фреймворків, таких як React Native;

5. Форуми та спільноти:

- *Stack Overflow* (<https://stackoverflow.com/>): Форум, де розробники можуть задавати та відповідати на питання, пов'язані з розробкою мобільних застосунків.
- *Reddit* (<https://www.reddit.com/r/androiddev/> та <https://www.reddit.com/r/iOSProgramming/>): Спільноти для розробників Android та iOS, де можна обговорювати питання та ділитися досвідом.

Перш ніж обирати ресурс для вивчення програмування мобільних застосунків, важливо врахувати свій рівень підготовки, мову програмування, яку хочете вивчати, а також конкретні платформи (Android або iOS).

1.3. Аналіз та обґрунтування вибору засобів з розробки освітнього ресурсу

Розробка електронних освітніх ресурсів може використовувати різноманітні технології в залежності від цілей, платформи та характеру матеріалів. До ключових технологій, які часто використовуються при створенні електронних освітніх ресурсів, відносяться:

1. Веб-розробка:

HTML/CSS/JavaScript: Веб-технології використовуються для створення інтерактивних веб-сайтів та електронних курсів. JavaScript дозволяє реалізовувати інтерактивні елементи, а HTML та CSS - структуру та стиль.

2. Електронні книги (E-books):

EPUB/PDF/MOBI: Формати електронних книг використовуються для розповсюдження навчальних матеріалів. EPUB є популярним форматом для електронних книг із підтримкою взаємодії та мультимедійних елементів.

3. Віртуальна реальність (VR) та доповнена реальність (AR):

VR/AR технології: Використовуються для створення іммерсивних навчальних досвідів. VR може занурити користувача у віртуальне середовище, тоді як AR може додавати інтерактивні елементи в реальний світ.

4. *Інтерактивні платформи:*

Adobe Captivate, Articulate Storyline: Ці інструменти дозволяють розробникам створювати інтерактивні модулі та курси без програмування.

5. *Системи управління навчанням (Learning Management Systems, LMS):*

Moodle, Blackboard, Canvas тощо: Використовуються для управління, відстеження та організації електронних курсів.

6. *Мультимедійні технології:*

Аудіо/відео: Використовуються для створення відеоуроків, віртуальних лекцій та інших мультимедійних матеріалів.

7. *Системи керування контентом (CMS):*

WordPress, Joomla, Drupal: Допмагають у створенні та управлінні контентом, особливо корисні для освітніх блогів та онлайн-курсів.

8. *Інтерактивні ігри для навчання:*

Unity, Unreal Engine: Використовуються для розробки серйозних ігор, які включають в себе навчальні аспекти.

9. *Бази даних:*

MySQL, MongoDB: Забезпечують зберігання та управління великою кількістю даних, таких як результати тестів, інформація про студентів тощо.

10. *Мобільні додатки:*

Android Studio (для Android), Xcode (для iOS), React Native, Flutter: Використовуються для розробки мобільних додатків для навчання та доступу до електронних ресурсів.

Розробка електронних освітніх ресурсів може включати в себе комбінацію вищеперерахованих технологій в залежності від потреб та специфікацій конкретного проекту.

Важливо враховувати, де і як розмістити створені ресурси. Зокрема, вважаємо, що доцільним є використання LMS. Перед вибором платформи важливо здійснити детальний аналіз потреб та вивчити конкретні плани та ціноутворення, щоб обрати оптимальне рішення для конкретної організації.

В схемі на рис. 1.2 описано особливості найпопулярніших платформ для розміщення освітніх ресурсів та керуванням цими ресурсами.

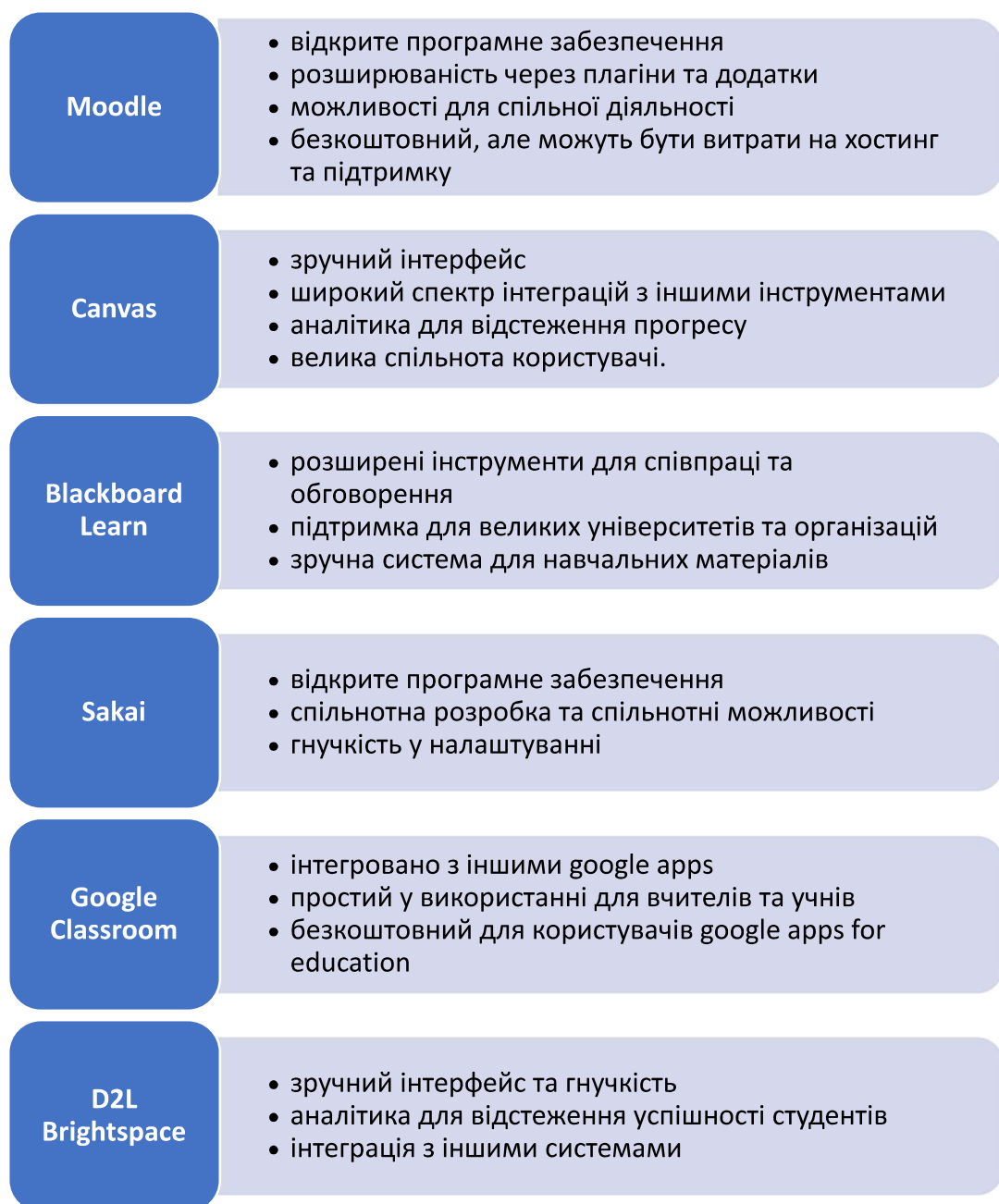


Рис.1.2 LMS платформи для розробки та розміщення освітніх ресурсів

Кожна з цих перерахованих платформ має свої особливості та переваги, і вибір LMS залежить від конкретних потреб та вимог організації чи закладу освіти. Порівняльний аналіз платформ для управління навчанням може включати різні аспекти, такі як функціонал, користувацький інтерфейс, спільнотні можливості, вартість та інше.

Щодо аналізу вартості платформ для управління навчанням (LMS) може бути складним завданням, оскільки ціни можуть значно відрізнятись в залежності від ряду факторів, таких як розмір організації, обсяг використання, потреби у вищих можливостях та інші. Також, багато LMS пропонують різні плани та варіанти ціноутворення. Нижче наведено загальний огляд *вартості кількох* популярних платформ LMS:

- *Moodle*: безкоштовний (відкрите програмне забезпечення). Однак, вартість може зростати при наявності специфічних потреб, таких як хостинг, підтримка, налаштування.
- *Canvas*: пропонує різні плани ціноутворення. Вартість зазвичай визначається в залежності від розміру організації та обраного плану. Вони надають індивідуальні котировки для кожного клієнта.
- *Blackboard Learn*: пропонує індивідуальні котировки, і ціни можуть бути визначені на основі різних факторів, таких як розмір установи, функціональність та інші.
- *Sakai*: відкрите програмне забезпечення, яке можна використовувати безкоштовно, але витрати можуть виникнути на налаштування, підтримку та інші додаткові послуги.
- *Google Classroom*: безкоштовний для користувачів Google Apps for Education, але вартість використання може змінюватись в залежності від інших пакетів Google Apps.
- *D2L Brightspace*: пропонує індивідуальні котировки, ціни можуть залежати від конкретних потреб та розміру користувача.

Нижче подано порівняльний *аналіз функціональних можливостей* декількох популярних платформ для управління навчанням (Learning Management Systems, LMS):

1) *Moodle*:

- Управління курсами та ресурсами.
- Система відстеження прогресу студентів.
- Форуми та інструменти спілкування.
- Завдання та тести.
- Підтримка різноманітних типів контенту.
- Розширюваність через плагіни.

2) *Canvas*:

- Гнучкі інструменти для створення та керування курсами.
- Спільнотні можливості для співпраці.
- Аналітика та звітність для відстеження прогресу студентів.
- Інтеграція з багатьма іншими засобами.

3) *Blackboard Learn*:

- Великий вибір інструментів для викладачів та студентів.
- Зручна система для створення матеріалів та завдань.
- Форуми та інструменти для обговорень.
- Засоби для оцінювання та звітність.

4) *Sakai*:

- Гнучкість у конфігурації та налаштуванні курсів.
- Форуми для обговорень та спілкування.
- Інтеграція з іншими інструментами через розширені плагіни.

5) *Google Classroom*:

- Інтеграція з іншими Google Apps (Docs, Sheets, Slides).
- Спільні папки та обмін матеріалами.
- Зручний інтерфейс для вчителів та учнів.

6) *D2L Brightspace*:

- Зручний інтерфейс для користувачів.
- Інструменти для співпраці та обговорень.
- Аналітика та інструменти для відстеження успішності.

Це лише загальний огляд та функціонал може різнитися в залежності від версій платформ та рівня їхнього налаштування. При виборі LMS важливо враховувати конкретні потреби та завдання організації чи навчального закладу.

Користувацький інтерфейс (UI) платформ для управління навчанням (Learning Management Systems, LMS) грає важливу роль у використанні та прийнятті платформи викладачами, студентами та іншими користувачами. Ось порівняльний аналіз користувацького інтерфейсу кількох популярних LMS:

1) *Moodle*:

- Гнучкий та розширюваний, але може виглядати менш інтуїтивно для новачків.
- Навігація здебільшого зосереджена на бічному меню.
- Часто використовується стандартний для багатьох систем дизайн.

2) *Canvas*:

- Сучасний та інтуїтивний інтерфейс.
- Зручна система для створення та редагування курсів.
- Добре організована навігація та доступність інструментів.

3) *Blackboard Learn*:

- Може виглядати трохи застарілим в порівнянні з сучасними LMS.
- Достатньо багато опцій та інструментів, що може бути важким для новачків.
- Меню та організація інтерфейсу може варіюватися залежно від версій та налаштувань.

4) *Sakai*:

- Відкрите програмне забезпечення зі спільотною розробкою, але інтерфейс може виглядати менш пристосованим.
- Навігація може бути менш стандартизованою, але гнучкою.

5) *Google Classroom*:

- Дуже простий та легкий у використанні інтерфейс.
- Інтегрований з іншими Google Apps.

- Зручний для вчителів та учнів, але може бути менше функціоналу порівняно з іншими LMS.

6) *D2L Brightspace:*

- Сучасний та інтуїтивний дизайн.
- Гнучкі інструменти та налаштування.
- Зручна навігація та організація.

Вибір LMS на основі користувацького інтерфейсу може бути залежним від специфічних потреб користувачів та рівня їхньої технічної підготовки. Багато з цих LMS пропонують демонстраційні версії або безкоштовні пробні періоди, що полегшує вибір відповідної системи.

Для розміщення освітнього ресурсу для навчання програмуванню мобільних застосунків вибрано платформу Moodle.

Moodle (Modular Object-Oriented Dynamic Learning Environment) – це відкрита платформа для управління навчанням (LMS), яка дозволяє створювати та управляти онлайн-курсами. Moodle розроблено з використанням технологій веб-розробки та засновано на концепції вільного та відкритого програмного забезпечення. Основна ідея Moodle – надання викладачам та студентам інструментів для взаємодії та обміну навчальним матеріалом в онлайн-середовищі.

Основні функціональні можливості Moodle:

- *Онлайн-курси.* Створення та управління курсами, включаючи завдання, тести, домашні завдання та інші матеріали.
- *Форуми.* Вбудовані інструменти для створення обговорень та форумів, що дозволяє вчителям та студентам обговорювати матеріали та питання курсу.
- *Оцінювання та тести.* Можливість створювати та проводити тести та інші форми оцінювання з різними видами питань.
- *Завдання.* Можливість виставляти завдання студентам, приймати їхні роботи та виставляти оцінки.
- *Завдання та курси з самонавчання.* Студенти можуть працювати над завданнями та матеріалами курсу у власному темпі.

- *Інтерактивні ресурси.* Підтримка вбудованих інтерактивних ресурсів, таких як відео, зображення, веб-сайти тощо.
- *Календар та нагадування.* Можливість створення календарів подій та нагадувань для студентів та вчителів.
- *Спільноти та соціальні засоби.* Можливість створення спільнот та використання соціальних функцій для взаємодії між учасниками.
- *Звітність.* Вбудовані інструменти для створення звітів та відстеження прогресу студентів.

При порівнянні LMS важливо враховувати специфічні потреби та контекст використання. Наприклад, для великих університетів може бути важливою гнучкість та розширюваність, тоді як для малих шкіл — простота та низькі витрати. Також слід враховувати інтеграційні можливості, безпеку, підтримку користувачів та інші аспекти вибору платформи для управління навчанням.

Дослідники А.С. Ломанченко, О.С. Хлопонін, Т.М. Деркач, Т.А. Дмитренко проаналізували такі платформи, як Moodle (<https://moodle.org>), ATutor (<https://github.com/atutor/ATutor/releases>), ILIAS (<https://github.com/ILIAS-eLearning/ILIAS>). В результаті дійшли висновку, що «система Moodle є найбільш функціональною, проте складною у використанні для користувачів, які не знайомі з веб-технологіями» (Ломанченко, Хлопонін, Деркач, & Дмитренко, 2021, с.88).

Важливим аспектом використання електронного освітнього ресурсу у форматі курсу є наявність контролю за результатами навчально-пізнавальної діяльності студентів. Погоджуємося із А.О. Томіліною, що саме «платформа Moodle має великі можливості щодо впровадження цілісної системи контролю й оцінювання знань та вмінь студентів» та має ряд переваг, таких як: «легкість організації, різноманітність варіацій, швидкість, доступність, об'єктивність, прозорість, своєчасне програмне забезпечення, відповідність сучасним темпам інформаційного потоку тощо» (Томіліна, 2015, с.228).

РОЗДІЛ 2

ТЕОРЕТИКО-МЕТОДИЧНІ АСПЕКТИ ПРОЕКТУВАННЯ ТА РОЗРОБКИ ЕОР З ПРОГРАМУВАННЯ МОБІЛЬНИХ ЗАСТОСУНКІВ

2.1. Основні етапи розробки електронних освітніх ресурсів

Розробка електронного освітнього ресурсу — це складний та багатоетапний процес, загальні етапи розробки якого представлено на рис.2.1.

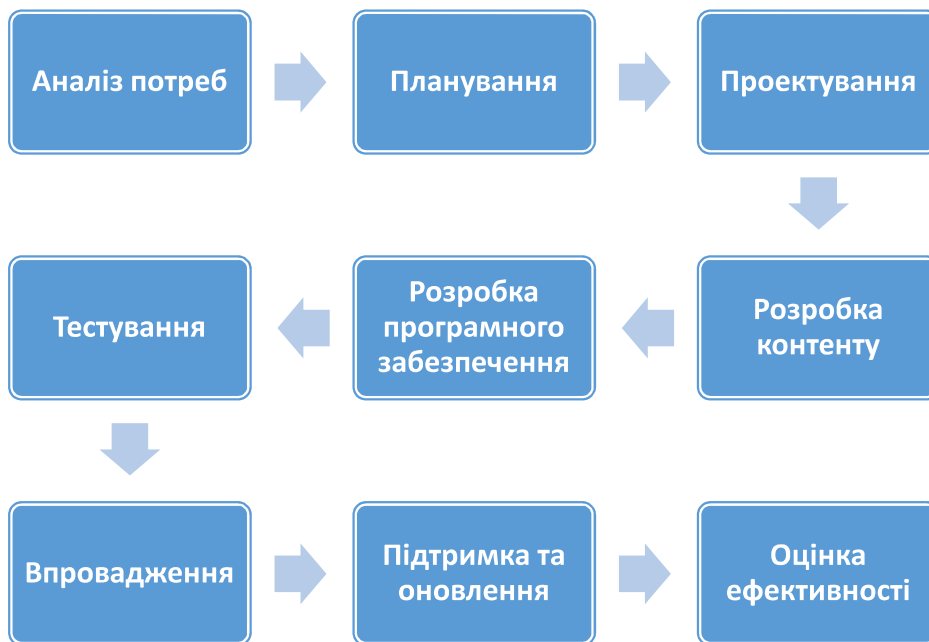


Рис. 2.1 Етапи розробки електронного освітнього ресурсу

Опишемо особливості кожного з етапів:

1. Аналіз потреб:

- визначення цільової аудиторії та їхніх потреб у навчанні;
- вивчення особливостей предметної області;
- визначення основних мет та завдань електронного ресурсу.

2. Планування:

- розробка структури та змісту ресурсу;
- визначення форматів контенту (текст, відео, зображення, інтерактивні вправи);
- встановлення методів оцінювання знань користувачів.

3.Проектування:

- створення дизайну інтерфейсу;
- розробка архітектури бази даних (якщо необхідно);
- вибір технологій для розробки та вибір засобів авторингу (якщо це інтерактивний контент).

4.Розробка контенту:

- створення навчальних матеріалів та ресурсів;
- розробка відеолекцій, текстових матеріалів, інтерактивних вправ;
- підготовка графічного та мультимедійного контенту.

5.Розробка програмного забезпечення:

- розробка програмного коду для функціональності ресурсу;
- інтеграція засобів для спілкування та взаємодії.

6.Тестування:

- проведення тестів на різних пристроях та браузерях;
- виявлення та виправлення помилок;
- тестування взаємодії користувача з ресурсом.

7.Впровадження:

- розгортання ресурсу на серверах або в хмарному середовищі;
- підготовка та проведення навчання для користувачів, якщо це необхідно;
- запуск ресурсу для широкого використання.

8.Підтримка та оновлення:

- моніторинг роботи ресурсу та реагування на можливі проблеми;
- проведення регулярних оновлень для вдосконалення та розширення функціоналу;
- забезпечення технічної підтримки користувачам.

9.Оцінка ефективності:

- збір та аналіз статистики використання ресурсу;
- оцінка задоволеності та успішності користувачів;

- впровадження змін на основі отриманих даних для покращення ресурсу.

Кожен з цих етапів важливий для створення високоякісного та ефективного електронного освітнього ресурсу. Під час розробки слід також враховувати засоби для створення доступних та користувацьких інтерфейсів, адаптивного дизайну та високої безпеки.

Крім того, слід враховувати на якому етапі чи в якому із напрямків освітнього процесу будуть використовуватися ЕОР. Зокрема, ЕОР можуть використовуватися в освітньому процесі в таких напрямках, як: «при підготовці і проведенні занять; для створення авторських мультимедійних посібників; в рамках індивідуальної та групової наукової діяльності студентів; в управлінні освітнім процесом» (Подлесний, Єрфорт, Жук,&Криворучек, 2018, с.54).).

Як зазначає С.Г.Литвинова важливо, щоб було правильно побудована структура ЕОР, що забезпечуватиме «можливість ефективного досягнення навчально-виховної мети, і, в залежності від функціонального призначення, включає: змістову частину, програмну частину, методичні рекомендації для вчителя, методичні рекомендації для учня, настанову користувача для адміністратора локальної мережі комп'ютерного класу або системного адміністратора навчального закладу» (Литвинова, 2015, с.72).

В роботі В.В. Лапінського (2013) визначено такі вимоги до змісту та організації ЕОР:

- під час проектування та створення засобів навчання, як складників ЕОР, потрібно враховувати ієрархії розумових дій та операцій суб'єкта навчання;
- структурування освітнього контенту та його подання у ЕОР повинне відбуватися із врахуванням принципів послідовності та системності;
- різнотипний контент (цифрові технології, відеоматеріали, онлайн-ресурси, довідеові системи тощо), як складники ЕОР, мають органічно інтегруватися в освітній процес, забезпечуючи

можливість групової та самостійної навчально-пізнавальної діяльності учасників цього процесу;

- програмні засоби необхідно супроводжувати відповідним методичним забезпеченням.

В додатку Б подано загальні вимоги у Положенні про електронні освітні ресурси, які теж слід врахувати під час розробки ЕОР потрібно дотримуватися ряд вимог (додаток В).

2.2. Особливості проектування та створення ЕОР із розробки мобільних застосунків

Відповідно до статистичних даних Statcounter GlobalStats (Statcounter GlobalStats. URL: <https://gs.statcounter.com/os-market-share#monthly-202209-202310-bar> (Дата звернення: 20.10.2023) станом на жовтень 2023 року частка використання пристроїв під керуванням операційних систем Android та iOS у всьому світі становила відповідно 40,46% та 17,08% по відношенню до інших операційних систем. Тому актуальним завданням ІТ-освіти є якісна підготовка ІТ-спеціалістів із програмування мобільних пристроїв, в тому числі розробка ЕОР.

Електронні освітні (дидактичні) у порівнянні з друкованими їх версіями мають ряд беззаперечних переваг, що визначаються наявністю: теоретичного (лекційного) матеріалу у форматі (відео, презентації, текстові файли тощо); переліку практичних завдань до лабораторних робіт і методичних рекомендацій для їх реалізації; тренувальних вправ і невеликих завдань для вивчення основних понять з вивчення мобільних технологій та набуття практичних умінь для їх проектування та створення; електронного журналу успішності; інтерактивних і тестових завдань тощо (Соловійов&Шроль, 2023, С.196).

Група дослідників (С.В. Подлесний, Ю.О. Єрфорт, Я.А. Жук, В.В. Криворучек) зазначає, «процес формування ЕОР вимагає програмного забезпечення, яке можна розділити на дві групи: для створення і для

використання електронних ресурсів. Стосовно до ЕОР – це програмне забезпечення ділиться на ПО для створення електронних послуг і ПО для надання електронних послуг. Таке програмне забезпечення створюється і експлуатується або окремо, або разом. Окремо програмне забезпечення створюється в тому випадку, коли група творців інформаційних послуг незалежна від групи споживачів» (Подлесний, Єрфорт, Жук,&Криворучек, 2018, с.53-54).

Як зазначають далі автори: «Якщо ЕОР функціонують в єдиному середовищі споживачів, які їх формують і модифікують, то програмне забезпечення інтегрує обидві групи» (Подлесний, Єрфорт, Жук,&Криворучек, 2018, с.54). В даному випадку передбачається, що одна частина програмного забезпечення відповідного середовища конфігурує або структурує систему, тоді, як друга забезпечує виконання обробки інформації, реалізацію виробничих процесів, отримання електронних документів і надання електронних послуг.

Вибираючи технології (засоби розробки, платформу для розміщення тощо) для проектування, створення та впровадження ЕОР з програмування мобільних застосунків, слід перевірити чи можливе забезпечення таких вимог, як: повнота інформації, систематизація та підготовка освітнього контенту відповідно до програми навчальної дисципліни «Програмування мобільних пристроїв» для студентів спеціальності 015.39 Професійна освіта (Цифрові технології)» (додаток Г); якість методичного інструментарію та технічного виконання, що визначається логічністю і послідовністю викладу освітнього контенту, наявністю наочності, гіпермедіа і мультимедіа технологій, інтерактивності тощо.

Ефективність використання платформи Moodle, з відповідно створеним курсом, підтверджено учасниками курсу (група ЦТ-41, 2022/2023 н.р.), які взяли участь в опитуванні та оцінювали функціонал платформи за такими критеріями (можливостями), як: дидактичні (рис.2.2а), педагогічні (рис.2.2б),

розвиваючі (рис.2.2в), програмні і адміністрація (рис.2.2г), функціональні (рис.2.2д).

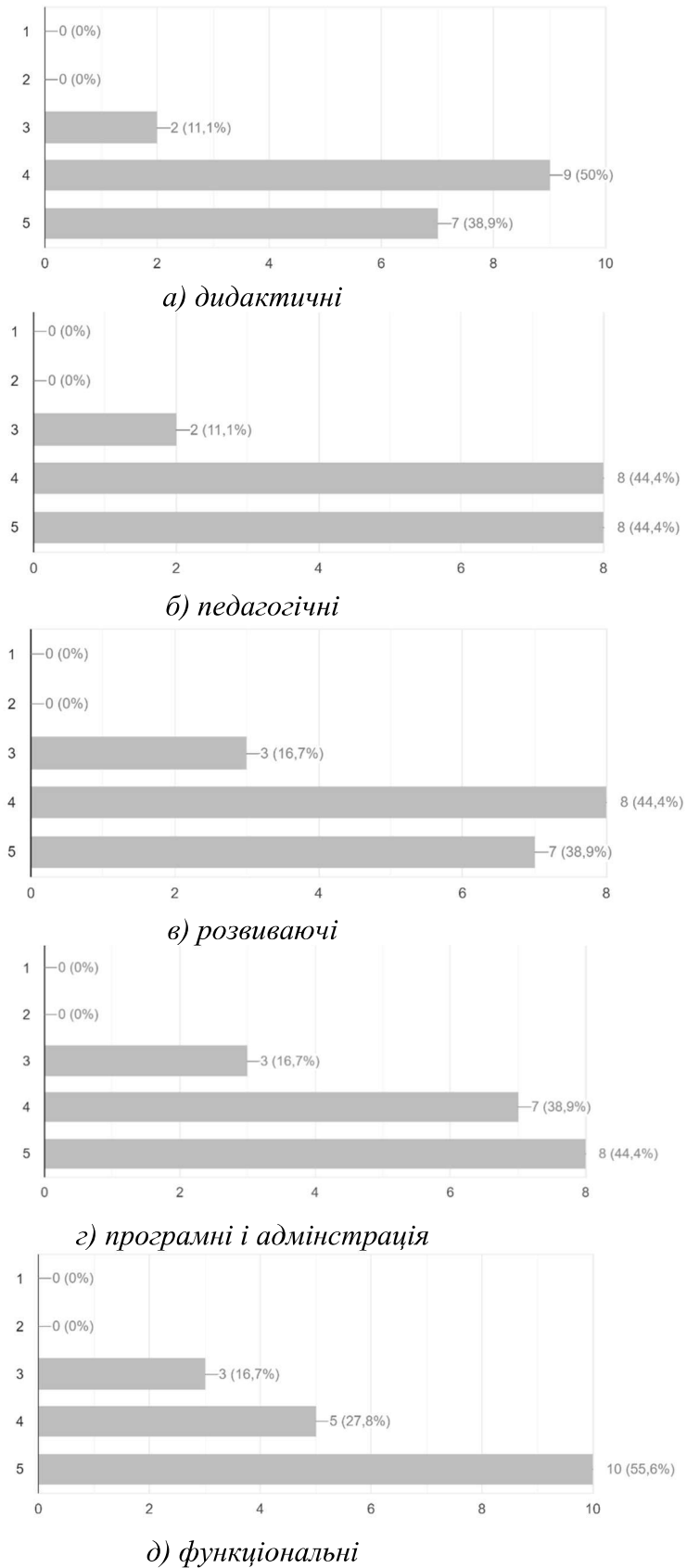


Рис. 2.2 Результати оцінювання можливостей платформи Moodle учасниками курсу

Крім того, під час проектування структури і контенту ЕОР слід орієнтуватися на вимоги, що визначаються основними завданнями розробника мобільних застосунків. Зокрема, в залежності від складності мобільного застосунку і його розробки в межах однієї лабораторної роботи (практичного завдання в межах однієї теми) чи в межах кількох лабораторних робіт слід в ході їх виконання врахувати проходження таких етапів, як: створення технічного завдання на розробку мобільного застосунку; обговорення із замовником (викладачем, тьютором курсу) етапів і процесу роботи над проектом; побудова архітектури застосунку; програмування (написання коду); робота із дизайнерами та із тестувальниками над налагодженням і тестуванням застосунків (якщо це груповий проект); допомога в створенні інструкцій по роботі із готовим застосунком; оформлення документації; підтримка мобільних застосунків; розміщення застосунків у AppStore і Google Play Market (Соловійов&Шроль, 2023, С.195-197).

На рис.2.3 представлено схему, що відображає структуру та етапи виконання здобувачем лабораторної роботи.

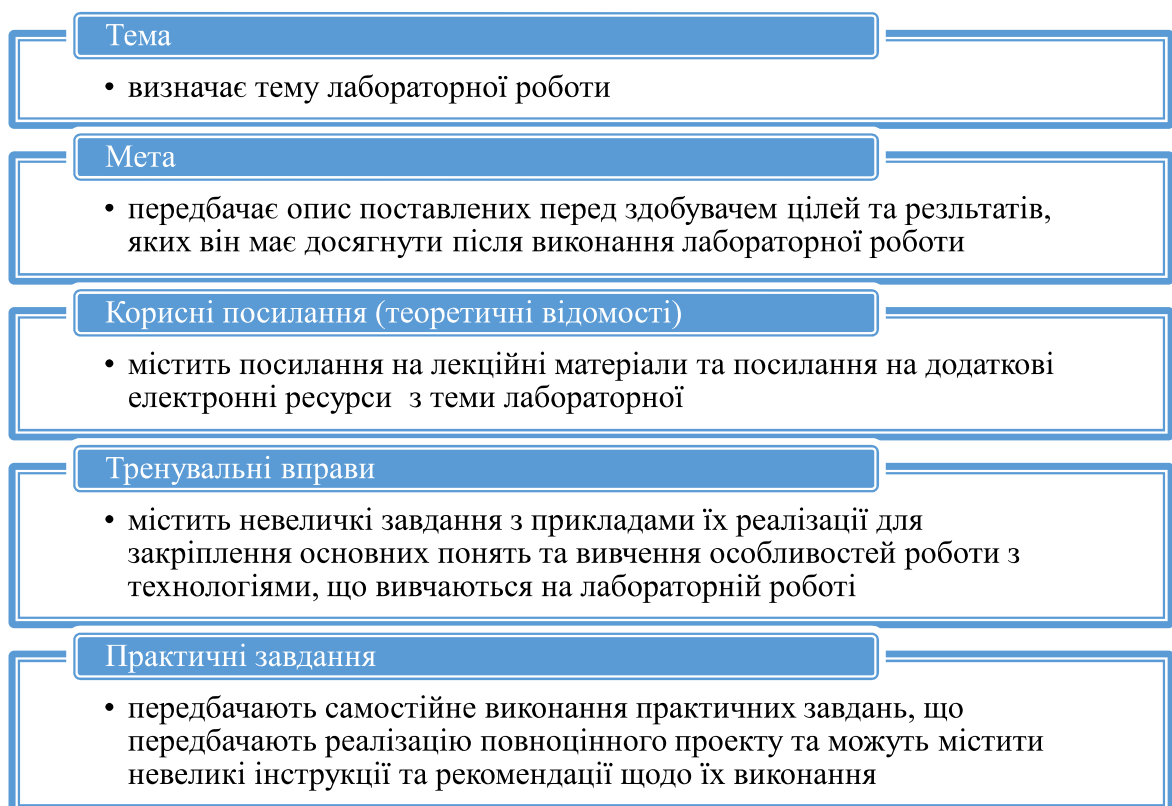


Рис. 2.3 Структура методичних рекомендацій до виконання лабораторної роботи

Акцентуємо увагу на тому, що результатом виконання однієї лабораторної роботи або послідовного виконання кількох лабораторних робіт є хоча б і невеличкий, але повноцінний мобільний застосунок.

2.3. Технології розробки мобільних застосунків

У своїй роботі Н.В. Ічанська, С.І. Улько надали рекомендації щодо вибору сучасних технологій та існуючих і широкоживаних засобів розробки, що є найбільш популярними для створення мобільних додатків, серед яких вони виокремили технології для розробки (Ічанська, Улько, 2020, С.76):

- *нативних застосунків*, що розроблені для конкретної платформи (iOS чи Android) з урахуванням її специфіки та доступом до всіх ресурсів (Apache Cordova, Java для Android і Objective-C / Swift для iOS);
- *багатоплатформних застосунків*, що розроблені для кількох платформ одразу і мають аналогічну функціональність незалежно від самої платформи і можуть бути нативні *багатоплатформні* (Xamarin, React Native, NativeScript) та *гібридні* (HTML5, CSS і JavaScript).

Автори також надали рекомендації щодо вибору типу застосунку відповідно до його призначення (Ічанська, Улько, 2020 , с.77), зокрема:

«1. Розробка гібридного додатку є доцільною при організації невеликих за масштабом проєктів, які не потребують складної логіки та дизайну. Наприклад, інформаційний портал.

2. Розробка нативного додатку є доцільною при організації швидкодійних, безпечних та багатоструктурних за своїм дизайном проєктів. Наприклад, соціальна мережа чи додаток для банку.

3. Розробка нативного багатоплатформного додатку є найбільш універсальною. Реалізує основні задачі масштабних проєктів, гарантує швидкодію, безпеку та дизайн відповідно базової платформи. Наприклад, додаток навчального закладу з власним форумом та можливістю подавати новини, проводити онлайн тестування чи опитування»

В даній роботі зосереджено увагу на розробці освітнього ресурсу у форматі курс на платформі Moodle із використанням фреймворку Flutter.

Flutter – це відкритий фреймворк для розробки крос-платформових мобільних застосунків. *Dart* – мова програмування, створена Google, яка використовується для написання коду в Flutter. Кросплатформність передбачає написання одного коду для мобільних платформ iOS та Android, емулюючи в собі інтерфейс і елементи обох операційних систем.

В останні роки популярність даного фреймворку серед інших технологій для розробки кросплатформних застосунків за даними Statista (2023) зростає від 30% у 2019 році до 46% в 2022 році і продовжує зростати (рис.2.4).

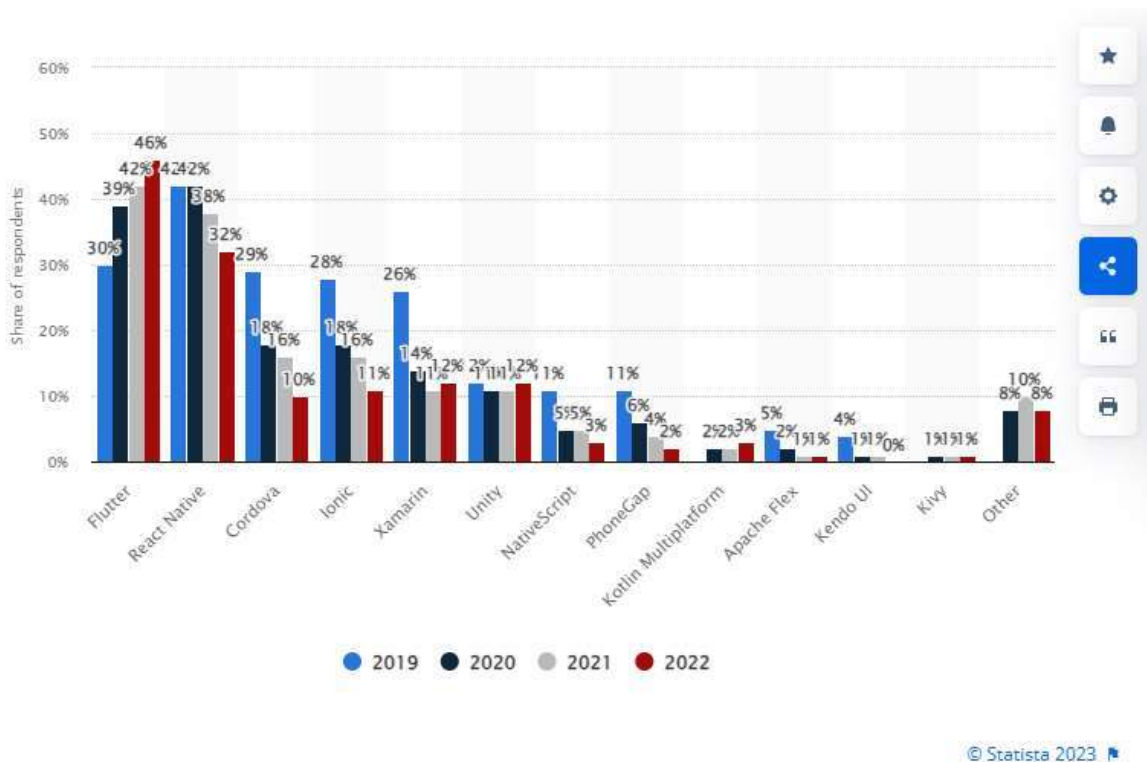


Рис. 2.4 Кросплатформні мобільні фреймворки, що використовувалися розробниками усього світу, за 2019-2022 рр.

Характеристики та переваги Flutter:

- *Крос-платформовість*: за допомогою Flutter можна розробляти один код, який працює як на iOS, так і на Android, що дозволяє зекономити час і зусилля.

- *Швидкість та продуктивність*: Flutter використовує власний двигун відображення, що забезпечує високу продуктивність та швидке оновлення інтерфейсу користувача.
- *Гарний інтерфейс*: завдяки власній бібліотеці віджетів, Flutter дозволяє легко створювати красиві та адаптивні інтерфейси для різних пристроїв.
- *Hot Reload*: миттєве оновлення змін у коді без перезавантаження додатку, що значно полегшує ітеративний процес розробки.
- *Розширюваність та адаптивність*: Flutter дозволяє легко розширювати функціональність додатків та адаптувати їх для різних платформ та розмірів екрану.
- *Широкий спектр пакетів та плагінів*: Flutter має активну громаду розробників, що призводить до широкого вибору пакетів та плагінів для різних задач.
- *Доступ до низькорівневих функцій*: Flutter надає доступ до платформеного API, що дозволяє використовувати функції, недоступні на вищих рівнях абстракції.

Можливості Dart:

- *Забезпечення високої продуктивності*: Dart визначається як "just-in-time" (JIT) та "ahead-of-time" (AOT) мова, що дозволяє досягти високої швидкодії в процесі розробки та виробництва.
- *Багатозадачність та асинхронність*: Dart підтримує асинхронні операції та використовує Future та Stream для ефективної роботи з асинхронним кодом.
- *Гнучка система типів*: Dart має гнучку систему типів, яка дозволяє розробникам вибирати між строгим та динамічним типізацією в залежності від конкретних потреб.
- *Широкий вибір стандартних бібліотек*: Dart постачається з багатьма корисними стандартними бібліотеками для роботи з рядками, колекціями, мережею і багато іншого.

- *Орієнтована на об'єкти мова*: Dart побудований на принципах об'єктно-орієнтованого програмування, що полегшує створення модульного та підтримуваного коду.

За цими характеристиками Flutter та Dart стають потужним інструментом для розробки мобільних додатків, які мають красивий інтерфейс, високу продуктивність та крос-платформову сумісність.

2.4. Методичні аспекти підготовки майбутній фахівців з інформаційних технологій до розробки мобільних застосунків

Під час розробки лабораторних завдань важливим є визначення технічного завдання, що ставиться перед розробниками. Наприклад, технічне завдання (ТЗ) для розробки застосунку магазину комп'ютерної техніки на Flutter/Dart може виглядати наступним чином:

1. Опис проекту:

1.1. Назва проекту:

- магазин комп'ютерної техніки.

1.2. Мета проекту:

- розробка крос-платформового мобільного додатку для магазину комп'ютерної техніки, який надає можливість перегляду та покупки товарів.

2. Функціональні вимоги:

2.1. Авторизація та реєстрація:

- можливість реєстрації та авторизації користувачів;
- аутентифікація через електронну пошту або соціальні мережі;

2.2. Каталог товарів:

- відображення категорій товарів;
- фільтри для сортування та пошуку товарів;
- сторінки з описом та характеристиками кожного товару;

2.3. Кошик та оформлення замовлення:

- можливість додавання товарів до кошика;
- перегляд та редагування змісту кошика;
- оформлення замовлення з введенням адреси та інших необхідних даних;

2.4. Особистий кабінет:

- відображення історії замовлень;
- зміна особистих даних користувача;

2.5. Опції платежу:

- інтеграція з системами онлайн-платежів (наприклад, Stripe, PayPal);

2.6. Управління адміністратором:

- можливість додавання, редагування та видалення товарів;
- відстеження та обробка замовлень;

3. Дизайн та інтерфейс:

3.1. Дизайн:

- мінімалістичний та зручний дизайн інтерфейсу;
- забезпечення адаптивності для різних розмірів екрану;

3.2. Графіка:

- використання візуальних елементів для покращення користувацького досвіду;

4. Технічні вимоги:

4.1. Технології:

- використання Flutter/Dart для крос-платформової розробки.

4.2. Сервер та база даних:

- розгортання сервера для збереження даних про товари, користувачів та замовлення;
- використання бази даних для збереження та управління інформацією;

4.3. Автентифікація:

- використання механізмів безпеки для захисту даних користувачів;

4.4. Забезпечення продуктивності:

- використання оптимізацій для забезпечення швидкої роботи додатку;

4.5. Тестування:

- розробка тестових сценаріїв та використання Flutter-тестування для перевірки функціональності;

5. Завдання та терміни:

5.1. Завдання:

- розробка інтерфейсу та його затвердження;
- розробка функціоналу додатку;
- тестування та виправлення помилок;
- розгортання на серверах та в магазинах додатків;

5.2. Терміни:

- додати терміни для кожного етапу розробки;

Це, звичайно, загальний шаблон, який здобувачі повинні адаптувати до конкретних вимог та специфікацій магазину комп'ютерної техніки, який вони розробляють.

Відповідно до поставленого технічного завдання розробка мобільного застосунку комп'ютерної техніки передбачала виконання здобувачами наступних етапів розробки мобільних застосунків:

1. Аналіз та планування:

- *Визначення функціональності:* визначення основних функцій, які повинен мати застосунок, такі як перегляд продуктів, додавання їх у кошик, оформлення замовлення тощо.
- *Дизайн інтерфейсу:* розробка макету інтерфейсу, враховуючи кращі практики дизайну.

2. Налаштування робочого середовища:

- *Встановлення Flutter та Dart SDK:* встановлення необхідного програмного забезпечення для розробки на Flutter.
- *Створення нового проекту:* запуск команд для створення нового проекту Flutter.

bashCopy code

flutter create my_computer_store_app

3. Створення інтерфейсу користувача:

- *Використання віджетів Flutter:* створення екранів для перегляду списку продуктів, вікно деталей продукту, кошика та ін.
- *Робота з анімаціями:* додавання анімації для поліпшення взаємодії та вигляду додатку.

4. *Інтеграція з базою даних:* використання локальної бази даних (наприклад, SQLite) або хмарної послуги (наприклад, Firebase) для зберігання інформації про продукти та замовлення.

5. *Робота з API:* якщо в завданні передбачено використання даних з сервера, то виконується інтеграцію з API для отримання інформації про продукти.

6. Робота з кошиком та оформленням замовлення:

- *Створення функціональності кошика:* надання користувачам можливості додавати продукти у кошик та видаляти їх.
- *Оформлення замовлення:* реалізація функціоналу для оформлення замовлення, введення доставкової інформації та підтвердження покупки.

7. *Тестування:* проведення тестування коду, включаючи одиничне та інтеграційне тестування для забезпечення стабільності та працездатності.

8. *Розгортання на пристрої:* після успішного тестування передбачає розгортання застосунку на пристрої або емуляторі.

9. *Підтримка та Оновлення:* передбачає підтримку та редагування на зворотній зв'язок для користувачів та реагування на їх зауваження для оновлення та вдосконалення додатку.

10. Маркетинг та Реклама: передбачає створення матеріалів та рекламних компаній для просування та реклами застосунку.

Це лише загальний огляд, і конкретні деталі та функціонал можуть варіюватися в залежності від потреб замовника, яким може виступати викладач, чи технічного завдання. Flutter та Dart можуть значно полегшити розробку крос-платформових мобільних застосунків, забезпечуючи швидкий та ефективний процес розробки.

Наведемо ще один приклад технічного завдання на прикладі розробки мобільного застосунку «Рецепти піци», що є одним із завдань лабораторної роботи (додаток Д) на Flutter/Dart може мати наступну структуру:

1. Опис проекту:

- Назва проекту: Рецепти піци
- Платформа: Мобільний додаток
- Фреймворк: Flutter/Dart

2. Основні функціональні вимоги:

- Головний екран:
 - Заголовок: "Рецепти піци"
 - Меню: Категорії піци (наприклад, класичні, вегетаріанські, екзотичні)
 - Пошук: Можливість пошуку рецептів за назвою чи інгредієнтами
- Сторінка категорії:
 - Список піц вибраної категорії
 - Кожен елемент містить зображення, назву та короткий опис піци
- Сторінка рецепту:
 - Зображення піци
 - Назва піци
 - Інгредієнти та їхні кількості
 - Інструкції з приготування
- Додавання рецепту:
 - Форма для введення назви, зображення, інгредієнтів та інструкцій

- Можливість вибору категорії
- Редагування та видалення рецепту:
 - Можливість редагування всіх полів рецепту
 - Підтвердження видалення рецепту
- Обробка фото:
 - Можливість додавання фотографії для рецепту

3. UI/UX:

- Модерний та привабливий дизайн
- Інтуїтивний і легкий для користувача інтерфейс

4. Безпека:

- Захист даних користувачів
- Використання безпечних методів збереження паролів, якщо є аутентифікація

5. Локалізація:

- Підтримка англійської та, за можливістю, інших мов

6. Зберігання даних:

- Локальне зберігання рецептів на пристрої
- Запити на сервер для синхронізації даних між пристроями користувачів (якщо планується серверна частина)

7. Підтримка платформ:

- Додаток повинен працювати як на Android, так і на iOS

8. Тестування:

- Можливість тестування всіх функцій та взаємодій між екранами
- Написання unit-тестів для важливих функцій

9. Документація:

- Коротка документація з інструкцією по встановленню та користуванню додатком

10. Розгортання та публікація:

- Інструкція з розгортання додатку на пристроях користувачів

- Підготовка до публікації в магазинах додатків (Google Play, App Store)

11. Підтримка:

- Можливість звертання до служби підтримки або надсилання відгуків через додаток

12. Завдання на майбутнє:

- Додати можливість ділитися рецептами через соціальні мережі
- Реалізувати аутентифікацію користувачів для збереження особистих рецептів

13. Технічні вимоги:

- Використання Flutter/Dart
- Використання стандартів коду та архітектури Flutter
- Використання пакетів для роботи з маршрутизацією, станом додатку та іншими основними завданнями

14. Команда розробників:

- Визначення складу команди, ролі та відповідальності

15. Терміни виконання:

- Визначення термінів для кожного етапу розробки

Це технічне завдання може слугувати основою для студентського проекту "Рецепти піци" на Flutter/Dart. Зверніть увагу, що деякі пункти можуть бути адаптовані залежно від конкретних потреб та вимог проекту.

РОЗДІЛ 3

ВПРОВАДЖЕННЯ ЕОР В ОСВІТНІЙ ПРОЦЕС ПІДГОТОВКИ БАКАЛАВРІВ СПЕЦІАЛЬНОСТІ ПРОФЕСІЙНА ОСВІТА (ЦИФРОВІ ТЕХНОЛОГІЇ)

3.1. Структура ЕОР із розробки мобільних застосунків

Під час роботи над даною кваліфікаційною роботою було розроблено електронний освітній ресурс у форматі онлайн курсу, який розміщений на платформі Moodle (<https://iktmedia.moodlecloud.com/>, додаток М) (рис.3.1).

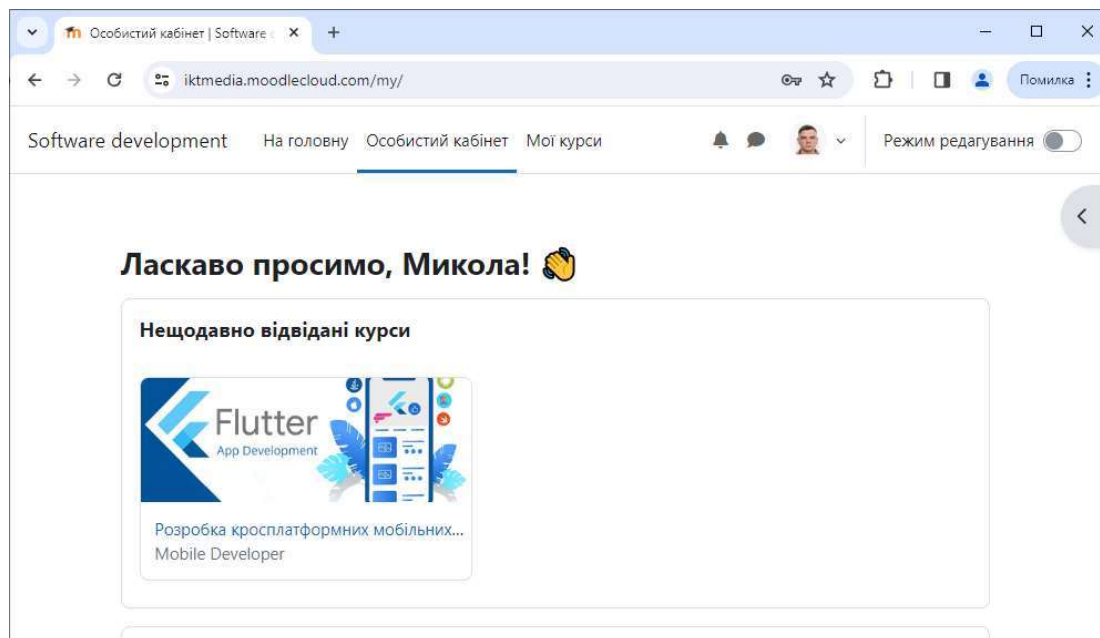


Рис. 3.1 Сторінка курсу «Розробка кросплатформних мобільних застосунків із використанням Flutter&Dart» (<https://iktmedia.moodlecloud.com/>)

Курс практично-орієнтований і структурований за темами відповідно до тем лабораторних робіт змістовного модуля 2 «Розробка кросплатформних мобільних застосунків» (див. додаток Г) навчальної дисципліни «Програмування мобільних пристроїв» (рис.3.2). Кожна тема містить теоретичні відомості (лекційний матеріал), вправи для виконання за вказаними інструкціями, завдання для самостійного виконання.

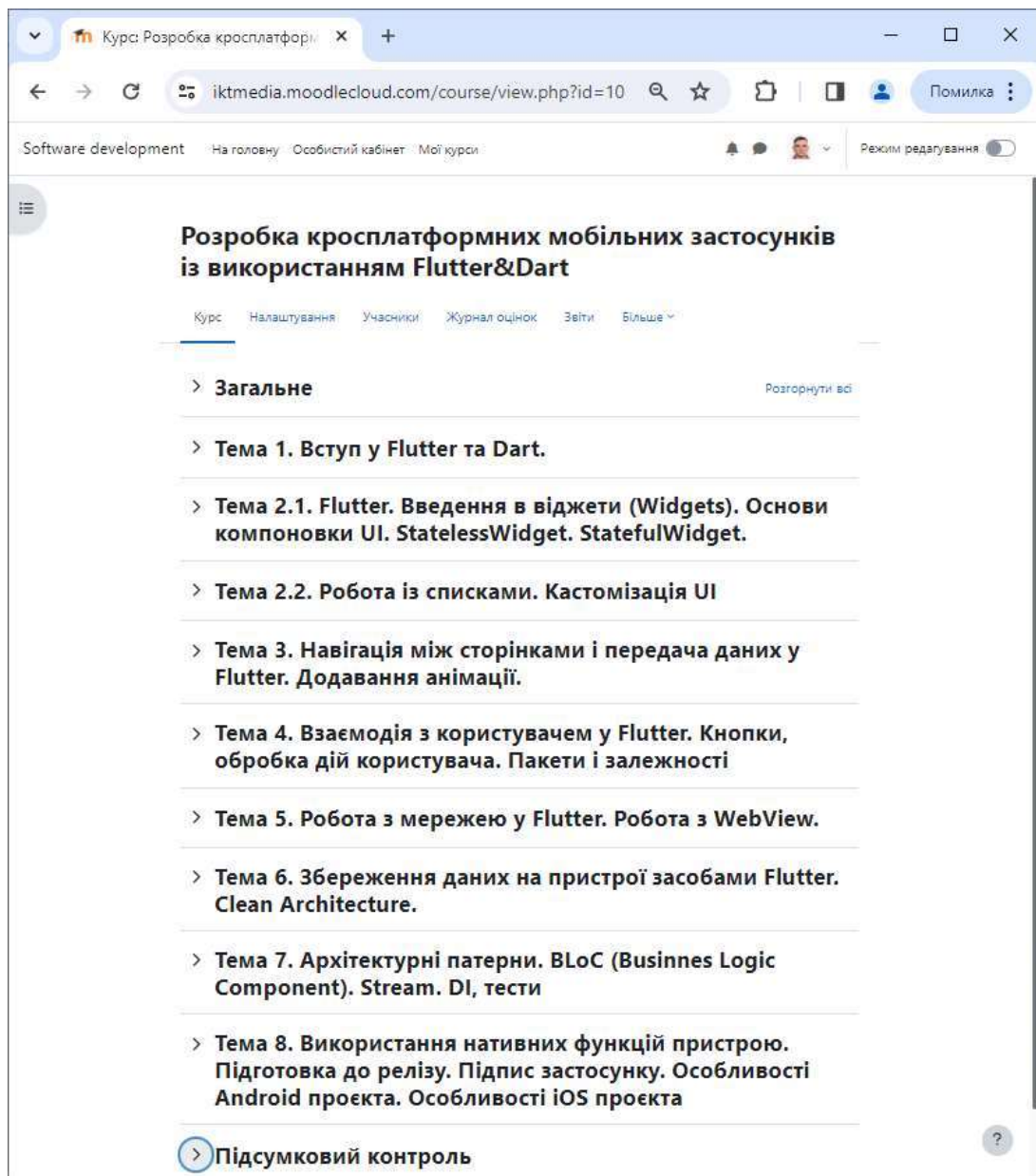


Рис.3.2 Структура ЕОР – онлайн-курсу на платформі Moodle

На початку курсу в розділі «Загальний» розміщено:

- «Оголошення та запитання щодо курсу», що створені на основі модуля «Форум» платформи Moodle;
- «Глосарій термінів курсу», що створений на основі модуля «Глосарій» платформи Moodle;
- «Інструкції для роботи з курсом», що створені на основі модуля «Сторінка» платформи Moodle;

- «Анкета учасника курсів», що створена на основі ресурсу «Гіперпосилання» платформи Moodle та мыстить посилання на форму, що створена за допомогою Google Forms.

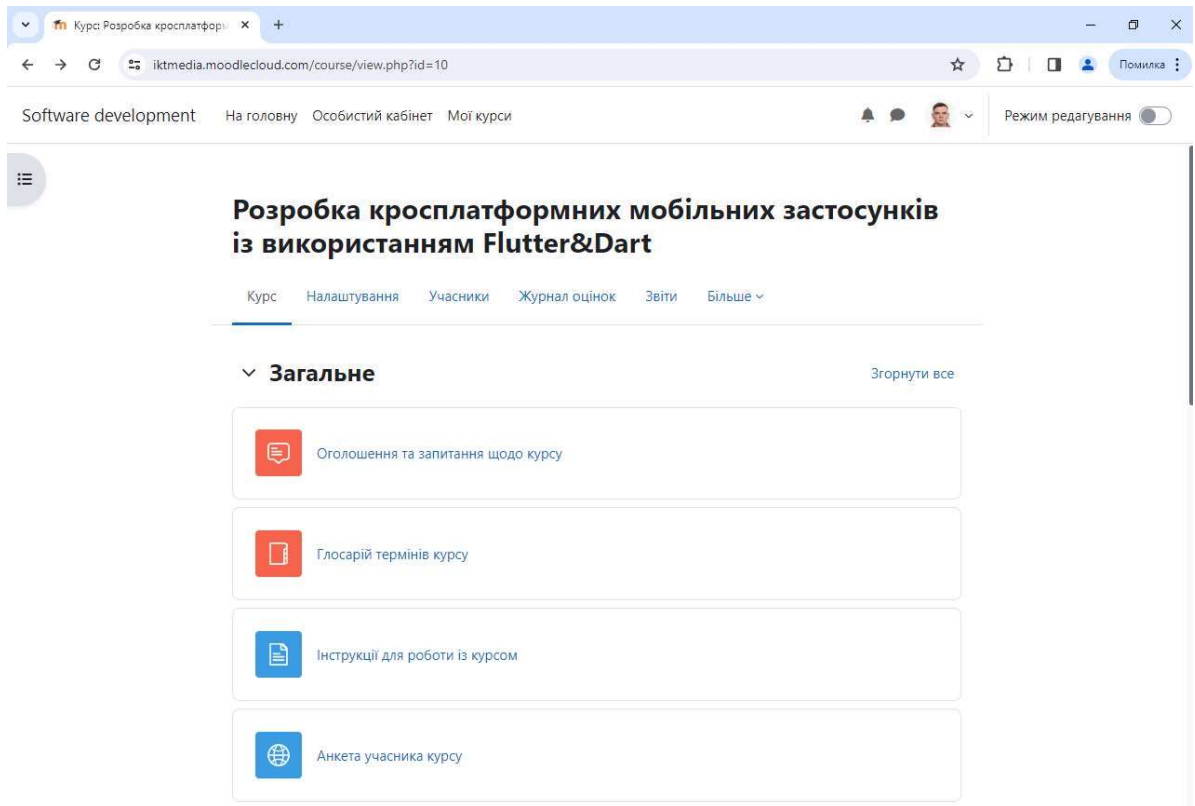


Рис.3.3. Розділ курсу «Загальний»

В кінці курсу слухач може пройти тестовий модульний контроль (рис.3.4), тестові завдання якого дозволяють не тільки перевірити, але сприяють систематизації та більш глибокому засвоєнню отриманих знань та навиків. Адже, тестові завдання поділені на три рівні за складністю їх виконання (до 70 балів, 70-90 балів та більше 90). Сам підсумковий тест включає 25 питань (рис. 3.3): 12 завдань (до 70 балів), 7 завдань (на 70-90 балів), 4 завдання (більше 90 балів). Приклад завдань згенерованих з «банку питань» підсумкового тесту представлений в додатку К.

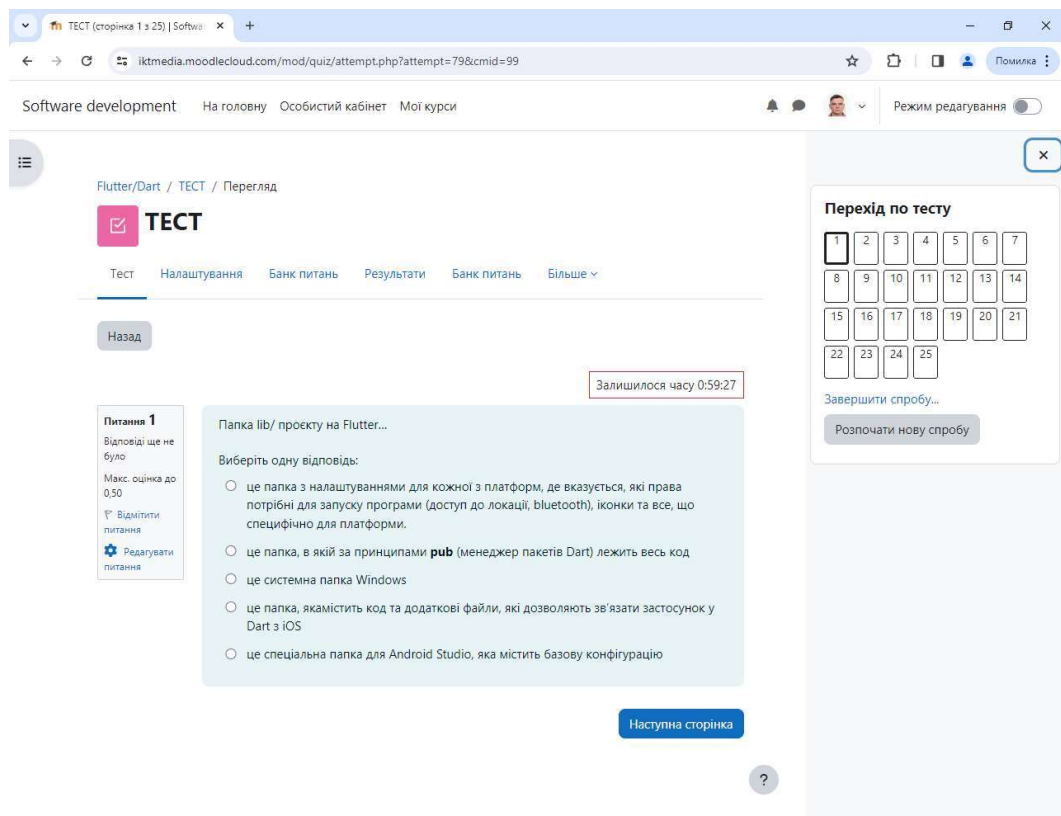


Рис.3.4 Підсумковий контроль

3.2. Методичні рекомендації щодо використання ЕОР з розробки мобільних застосунків

Розроблені електронні матеріали інтегровані в електронний освітній ресурс у форматі курсу було розміщено на платформі Moodle у хмарній її версії для тестування за адресою <https://iktmedia.moodlecloud.com/> і на платформі <https://do.rshu.edu.ua/> – для апробації.

Оскільки курс є практично-орієнтований, то передбачає виконання лабораторних робіт, що містить ряд практично-орієнтованих завдань. Наприклад, під час вивчення особливостей компоновки дизайну користувача засобами Widgets студенти проектують та будуть застосунок «Погода» (рис.3.5), в процесі вивчення навігації між екранами розробляють застосунок «Рецепт піци» (рис.3.6), а під час побудови інтерфейсу (форм) для взаємодії з користувачем вони додають у застосунок (чи рецептів піци чи магазину) авторизацію (рис.3.7).



Рис.3.5

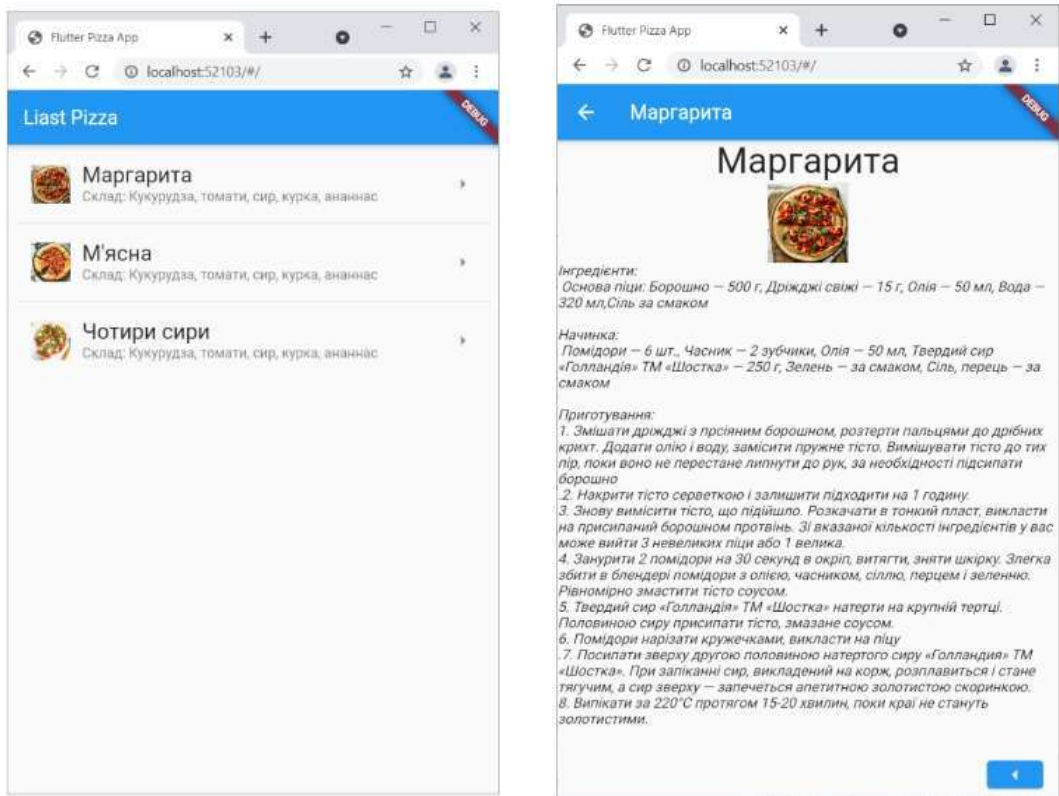


Рис. 3.6

Рис.3.7

Виконання лабораторної роботи передбачає проходження наступних етапів (див. додаток Д):

- опрацювання теоретичного матеріалу у формі;
- виконання тренувальних вправ за наданими інструкціями для вивчення основних можливостей Flutter у відповідності до теми, яку проходить учасник курсу;
- виконання практичних завдань самостійно.

Для підготовки звіту і для захисту виконаної практичної роботи (лабораторної) в діяльності на платформі Moodle потрібно:

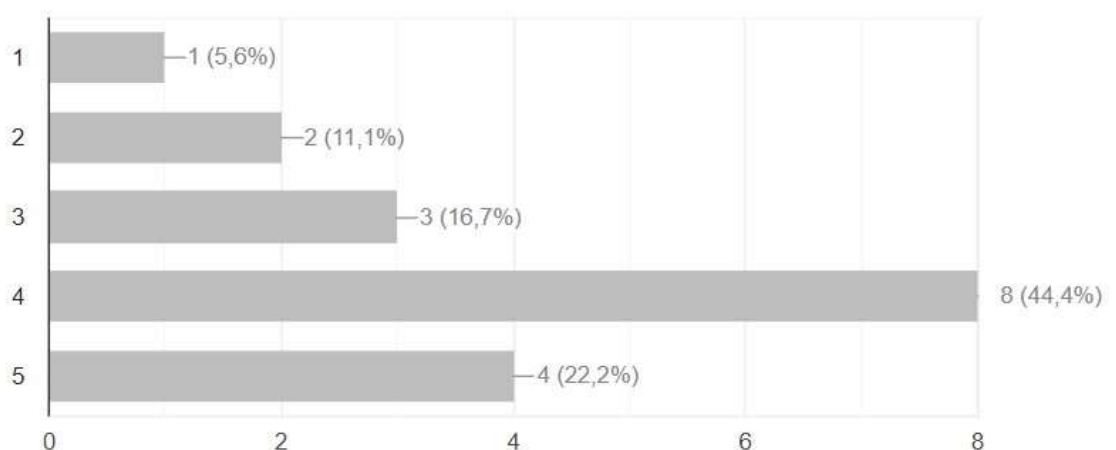
- 1) зберегти файли проекту, оформити звіт із скрінами запуску застосунків та завантажити в папку «ПМД» на Google Диску;
- 2) після натиснення кнопки «Здати роботу» в полі "текст-онлайн" потрібно вставити посилання для доступу викладачу (тьютору) курсу до папки "ПМД" та натиснути кнопку "Зберегти";
- 3) відправити викладачу (тьютору) для попереднього оцінення натиснувши кнопку "Відправити на оцінення";

- 4) додаткові запитання від викладача (тьютора) та захист в режимі онлайн (через Goole Meet, коментарем або листом) та офлайн.

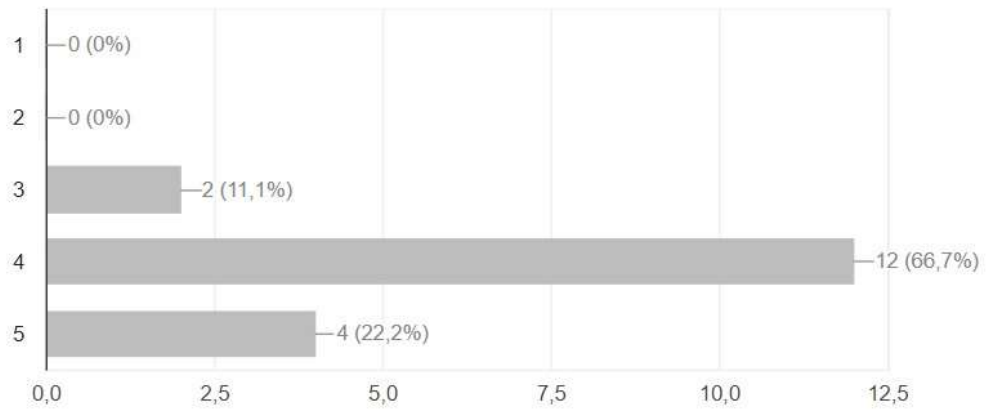
Після апробації матеріалів курсу під час опрацювання матеріалів змістовго модуля 2 здобувачами було проведено анкетування (додаток К), розробленого на основі анекти Т.С. Шроль (Шроль, 2017, с.303-304) щодо якості даного освітнього ресурсу (онлайн-курсу) та його використання у форматі дистанційного та змішаного навчання.

На рис. 3.8, рис. 3.9 та рис. 3.10 наведено результати самооцінювання студентами власних компетентностей з інформаційних технологій у відповідності до питань анкети «Розробка кросплатформних мобільних застосунків із використанням *Flutter&Dart*» (додаток Л):

- загальні компетентності в галузі ІТ (користувач ІТ)
- компетентності в галузі педагогічних технологій (ведення звітності, створення журналів, підготовка звітних документів, створення та редагування дидактичних та методичних матеріалів та іншого з використанням різноманітного ПЗ)
- предметні компетентності (використання різноманітних ІТ для вивчення та навчання інформатики та програмування, розробки програмних продуктів).

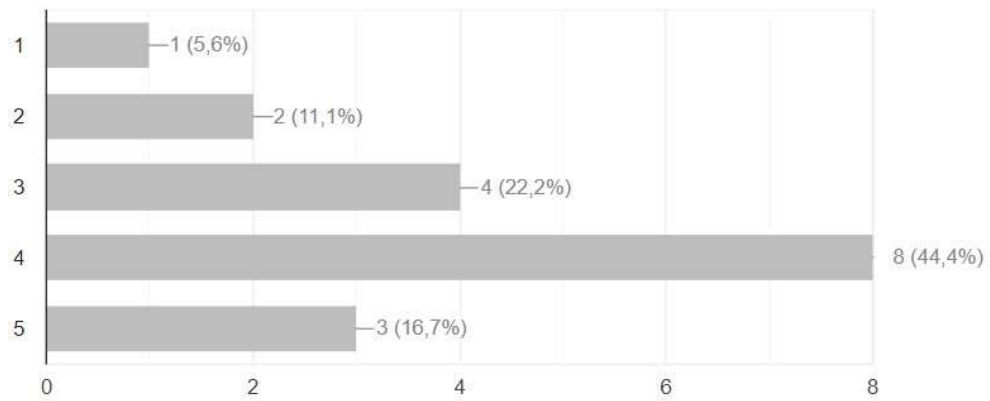


а) Перед початком курсу

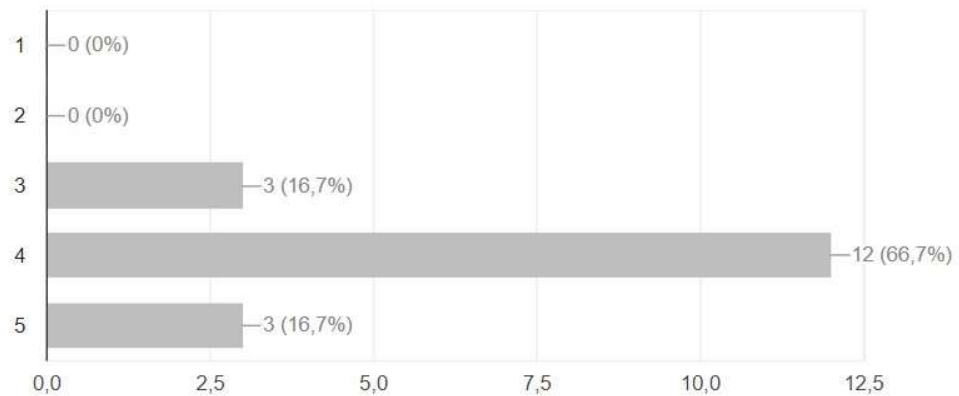


б) Після завершення курсу

Рис. 3.8. Рівень сформованості загальних компетентностей в галузі ІТ до та після завершення курсу

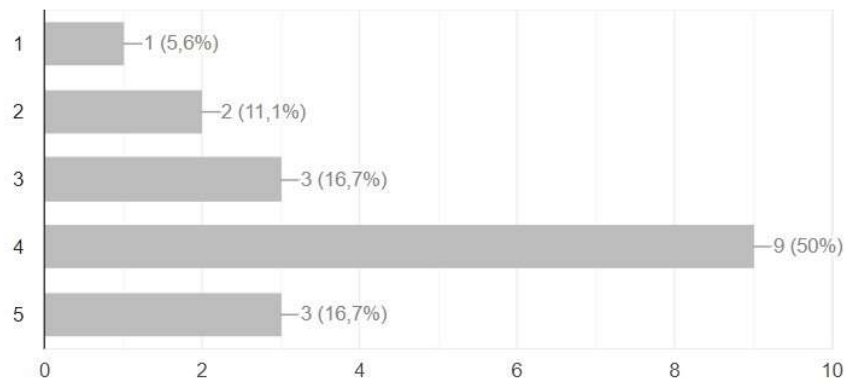


а) Перед початком курсу

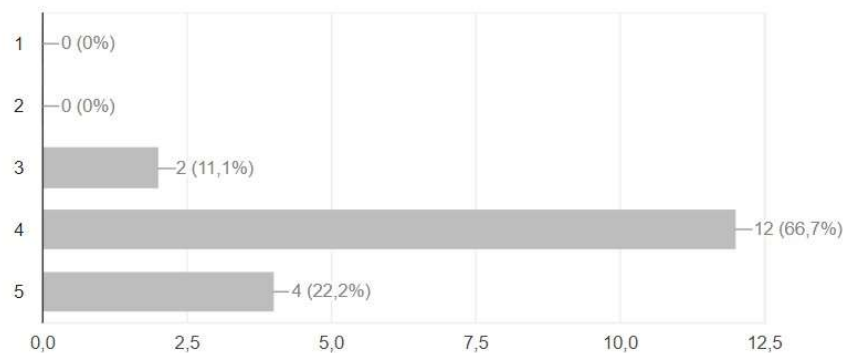


б) Після завершення курсу

Рис. 3.9. Рівень сформованості компетентностей в галузі педагогічних технологій до та після завершення курсу



а) Перед початком курсу



б) Після завершення курсу

Рис. 3.10. Рівень сформованості предметних компетентностей до та після завершення курсу

Аналіз результатів представлених діаграм вказує на зростання здатностей використовувати ІТ, як для особистісного розвитку і навчання, так і для майбутньої професійної діяльності, зокрема в процесі розробки мобільних застосунків.

Даний курс, а також результати магістерської роботи можуть бути використані під час навчання програмуванню мобільних застосунків, зокрема кросплатформенних, в курсі «Програмування мобільних застосунків» здобувачами освіти спеціальності 015 «Професійна освіта (Цифрові технології)» в закладах вищої освіти та підготовки здобувачів у закладах професійно-технічної та фахової передвищої освіти спеціальностей: інженерія програмного забезпечення; комп'ютерна інженерія; розробка програмного забезпечення тощо.

ВИСНОВКИ

У ході написання магістерської роботи було розглянуто поняття електронного освітнього ресурсу, особливості його проектування та створення у формі дистанційного курсу для навчання програмуванню мобільних кросплатформних застосунків для закладів вищої, професійно-технічної та фахової передвищої освіти.

Дослідження, аналіз та систематизація інформації щодо використання ЕОР з розробки мобільних застосунків для підготовки ІТ-фахівців в науковій, науково-методичній літературі та практиці не достатньо розкрито. Хоча існує досить значна кількість ЕОР з розробки мобільних застосунків у формі відео, електронних підручників, курсів на платформах Udemu, Coursera, Edx тощо, проте важливим є подальше удосконалення, розроблення таких ресурсів та методики їх використання, які мотивували б до навчально-пізнавальної діяльності та сприяли б успішному опануванню технологій розробки мобільних застосунків засобами Flutter.

В межах виконання роботи було розроблено ЕОР у форматі курсу в системі Moodle, що передбачає виконання різних видів діяльності на цій платформі: перегляд текстових та відео-матеріалів, виконання тренувальних вправ за відповідними інструкціями, практичних завдань для самостійного виконання та підсумкового модульного контролю. Матеріали курсу відповідають змісту програми навчальної дисципліни «Програмування мобільних пристроїв» для здобувачів спеціальності 015.39 Професійна освіта (Цифрові технології) РДГУ, зокрема змістового модуля 2 «Розробка кросплатформних мобільних застосунків із використанням фреймворку Flutter».

Матеріали курсу були апробовані під час проведення лабораторних занять підготовки бакалаврів спеціальності 015.39 Професійна освіта (Цифрові технології) в період 2022/2023 та 2023/2024.

Результати магістерської роботи можуть бути використані під час навчання програмуванню, зокрема з розробки мобільних застосунків для здобувачів спеціальності 015 «Професійна освіта (Цифрові технології та

підготовки здобувачів у закладах професійно-технічної та фахової передвищої освіти спеціальностей інженерія програмного забезпечення; комп'ютерна інженерія; розробка програмного забезпечення тощо.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Артеменко, В.Б., Артищук, І.В., & Гудзовата, О.О. (2018). Сучасні веб-інструменти в онлайн-курсі “Теорія і методика дистанційного навчання”. Вісник Львівського торговельно-економічного університету. Технічні науки. Вип. 21, 126-133
2. Бойко М.А. (2019). Розробка та впровадження електронних освітніх ресурсів у процесі навчання інформатики учнів початкової школи :автореферат. [Автореф. дис. канд. пед. наук, Державний заклад «Луганський національний університет імені Тараса Шевченка»].
3. Борисенко Д.В. (2018). Використання мобільних додатків при розробленні дизайн-продукту у навчанні майбутніх фахівців з дизайну. *Інформаційні технології і засоби навчання*, 68(6), 47–63. <https://doi.org/10.33407/itlt.v68i6.2224>
4. Візнюк, І., Буглай, Н., Куцак, Л., Поліщук, А., & Киливник, В. (2021). Використання штучного інтелекту в освіті. *Modern Information Technologies and Innovation Methodologies of Education in Professional Training Methodology Theory Experience Problems*, 14–22. <https://doi.org/10.31652/2412-1142-2021-59-14-22>
5. Галицький, О. В. (2021). Управління електронними освітніми ресурсами з використанням веб-орієнтованих комп’ютерних систем. [Дис. канд. пед. наук, Нац. пед. ун-т ім. М.П. Драгоманова].
6. Грановська, Т.Я. (2018) Застосування засобів мобільних технологій для навчання учнів предметам циклу точних і природничих наук. Педагогіка формування творчої особистості у вищій і загальноосвітніх школах. №61. Т.1. с.49-52. http://pedagogy-journal.kpu.zp.ua/archive/2018/61/part_1/12.pdf
7. Дущенко, О.С. (2020). Підходи до класифікації та створення додатків. Наукові записки. Серія: Педагогічні науки, (191), 70-73. <https://doi.org/10.36550/2415-7988-2020-1-191-70-73>

8. Женченко, М. І., Мельник, О. М., Женченко, І. В., & Прихода Я. В. (2020). Видавничо-педагогічний підхід до проблеми типологізації електронних освітніх ресурсів. Соціальні комунікації. Поліграфія і видавнича справа. 1 (79). 121-141. DOI:10.32403/0554-4866-2020-1-79-121-141.
9. Журавель, Д.П., Журавель, В.П., & Бабаніна, Т.В. (2021). Методологія використання електронних освітніх ресурсів під час викладання предметів природничоматематичного циклу. Збірник науково-методичних праць Таврійського державного агротехнологічного університету імені Дмитра Моторного, (24), 129-137.
<http://elar.tsatu.edu.ua/bitstream/123456789/13806/1/%D0%96%D1%83%D1%80%D0%B0%D0%B2%D0%B5%D0%BB%D1%8C%20%D0%94.%D0%9F.-1.pdf>
10. Зайцев, В.Е., & Бабко, К.С. (2018, 8-9 листопада). Електронний освітній ресурс як інструмент інформаційнокомунікаційних технологій. Сучасні інформаційні технології та інноваційні методики навчання: досвід, тенденції, перспективи», № 2. С.84-87.
http://dspace.tnpu.edu.ua/bitstream/123456789/15256/1/Zaitsev_Babko.pdf
11. Ічанська, Н.В., & Улько, С.І. (2020). Основні аспекти створення мобільних додатків та вибір інструментів їх розробки. *Системи управління, навігації та зв'язку*. Збірник наукових праць, 1(59), 74-78.
<https://doi.org/10.26906/SUNZ.2020.1.074>
12. Карплюк, С.О., & Вербівський, Д.С. (2018). Розробка електронних освітніх ресурсів як засіб підвищення ефективності вивчення курсу загальної фізики в педагогічному закладі вищої освіти.
<http://enpuir.npu.edu.ua/handle/123456789/23677>
13. Клочко, О.В., Гуменний, О.Д., & Клочко, В.І. (2018). Smart-технології електронних освітніх ресурсів. Науковий часопис НПУ імені М. П. Драгоманова. Серія №2. Комп'ютерно-орієнтовані системи навчання:

- Зб. наук. праць /Редрада. К.: НПУ імені М.П. Драгоманова. 20 (27), 44-50. <http://enpuir.npu.edu.ua/handle/123456789/21863>
14. Колмакова, В.О. (2022). Імерсивні технології як сучасна освітня стратегія підготовки майбутніх фахівців. Українські студії в європейському контексті. (5), 177-182. http://obrii.org.ua/usec/storage/article/Kolmakova_2022_177.pdf
15. Коневщинська, О. Е. (2014) Електронні освітні ресурси у межах інформаційного забезпечення діяльності ресурсних центрів дистанційної освіти. Інформаційні технології і засоби навчання, 43(5), 164-173. <https://lib.iitta.gov.ua/7607/1/1140-4156-1-PB.pdf>
16. Крупко, С. (2022а) Аналіз моделі використання електронних освітніх ресурсів у закладах загальної середньої освіти. Нові технології навчання: збірник наукових праць. ДНУ «Інститут модернізації змісту освіти», Київ. (96), 112-119. DOI: <https://doi.org/10.52256/2710-3560.2022.96.15>
17. Крупко, С. (2022б). Електронні освітні ресурси: реалії сучасного освітнього середовища. Проблеми освіти, 2(97), 226-238.
18. Кучерук, О. А., & Магдич, Т. П. (2020). Використання електронних освітніх ресурсів для формування громадянської компетентності учнів ліцею на уроках української мови. Інформаційні технології і засоби навчання, 75(1), 56–75. <https://doi.org/10.33407/itlt.v75i1.3515>
19. Лазаренко, С.В. (2017) Мобільні додатки та технології для організації самостійного оволодіння курсантами англomовною лексико-граматичною компетентністю у читанні. Innovative solutions in modern science, 1(10), oai:ojs.naukajournal.org:article/1092
20. Лапінський, В.В. (2013). Електронні освітні ресурси – дидактичні вимоги і класифікація Педагогіка вищої школи: методологія, теорія, технології, 1 (3) (2 (50)), 214-218. <https://lib.iitta.gov.ua/id/eprint/5369>
21. Литвинова, С.Г. (2015) Особливості розробки критеріїв оцінювання електронних освітніх ресурсів. Наукові записки. Випуск 7. Серія:

- Проблеми методики фізико-математичної і технологічної освіти. Частина 3. Кіровоград: РВВ КДПУ ім. В.Винниченка, 70-75. <https://phm.cuspu.edu.ua/ojs/index.php/NZ-PMFMTO/article/view/133>
22. Литвинова, С.Г. (2013). До питання експертизи якості електронних освітніх ресурсів. Інформаційні технології і засоби навчання, 34(2), 21–27. <https://doi.org/10.33407/itlt.v34i2.812>
23. Мацюкін, Д.В., & Пахомова, І.М. (2020). Платформи й мобільні додатки для створення та використання контенту із технологією доповненої реальності в освітньому процесі. Проблеми сучасної освіти, (11), 153-160. вилучено із <https://periodicals.karazin.ua/issuededu/article/view/17672>
24. Мінтій, І.С., Шокалюк, С.В., Литвинова, С.Г., & Пінчук, О.П. (2020). Проектування електронних навчальних курсів на основі типового Moodle-курсу університету. Вісник післядипломної освіти. «Серія «Педагогічні науки», 14 (43), 66–84. <http://elibrary.kdpu.edu.ua/xmlui/handle/123456789/4108>
25. Морзе, Н.В., Глазунова, О.Г., & Мокрієв, М.В. (2016) Методика створення електронного навчального курсу (на базі платформи дистанційного навчання Moodle 3): Навчальний посібник. К. 240 с. https://elearn.nubip.edu.ua/pluginfile.php/656406/mod_resource/content/0/%D0%9F%D0%BE%D1%81%D1%96%D0%B1%D0%BD%D0%B8%D0%BA_%D0%95%D0%9D%D0%9A.pdf
26. Мороз, Т.О., & Ендрес, В.С. (2019) Переваги використання гібридних мобільних додатків та прогресивних веб-додатків у бізнесі. Вісник аграрної науки Причорномор'я, Вип. 1. С.96-102. DOI: 10.31521/2313-092X/2019-1(101)-14
27. Нерсесов, Вадим. Moodle Made Easy: персоналізація, розробка та ефективне використання LMS Moodle 4. 99 с. URL: https://www.undp.org/sites/g/files/zskgke326/files/2023-07/moodle_made_easy_manual.pdf

28. Освітньо-професійна програма «професійна освіта (цифрові технології)» першого (бакалаврського) рівня вищої освіти за спеціальністю 015 «Професійна освіта (Цифрові технології)» галузі знань 01 «Освіта/Педагогіка». РДГУ. Рівне. 2020. URL: https://www.rshu.edu.ua/images/osvitni_programi/osv_prog_bak_015_39_prof_osv_cifr_teh_2020.pdf
29. Подлесний, С.В., Єрфорт, Ю.О., Жук, Я.А., & Криворучек, В.В. (2018) Електронні освітні ресурси як основа якісної інженерної підготовки в системі вищої освіти. Вісник Донбаської державної машинобудівної академії. № 1 (43), 53-58. [http://www.dgma.donetsk.ua/science_public/ddma/Herald_1\(43\)_2018/article/12.pdf](http://www.dgma.donetsk.ua/science_public/ddma/Herald_1(43)_2018/article/12.pdf)
30. Поляков, А.О., Федорченко, В.М., & Шматко, О.В. (2017). Аналіз методів і технологій розроблення мобільних додатків для платформи Android : навч. посіб.; Харківський національний економічний університет ім. С. Кузнеця. Х.: ХНЕУ ім. С. Кузнеця, 286. <http://repository.hneu.edu.ua/handle/123456789/20105>
31. Про затвердження Положення про дистанційне навчання: наказ Міністерства освіти і науки від 25.04.2013 № 466. URL: <https://zakon.rada.gov.ua/laws/show/z0703-13#n18>
32. Про затвердження Положення про електронний підручник: наказ Міністерства освіти і науки від 02.05.2018 № 440. URL: <https://zakon.rada.gov.ua/laws/show/z0621-18#n14>
33. Про затвердження Положення про електронні освітні ресурси: наказ Міністерства освіти і науки, молоді та спорту України від 01.10.2012 № 1060. <http://zakon0.rada.gov.ua/laws/show/z1695-12>
34. Про внесення змін до Положення про електронні освітні ресурси: Наказ Міністерства освіти і науки від 29.05.2019 № 749. <https://ips.ligazakon.net/document/view/Re33637?an=1>

35. Соловійов М., Шроль Т.С. Електронний освітній ресурс із розробки мобільних застосунків як ефективний засіб підготовки майбутніх ІТ-фахівців. Інформаційні технології в професійній діяльності : матеріали XVI Всеукраїнської науково-практичної конференції / Рівне : РВВ РДГУ. 2023. С.195-197.
36. Сороко Н.В. (2022, 2 листопада 2021р) Підходи до використання імерсивних технологій в закладах загальної освіти. Мультимедійні технології в освіті та інших сферах діяльності: науково-практична конференція з міжнародною участю. К.: НАУ, 113-115. <https://er.nau.edu.ua/handle/NAU/55788>
37. Сороко, Н., & Гаєвська, О. (2022). Використання імерсивних технологій для підготовки майбутніх перекладачів східних мов. *Modern Information Technologies and Innovation Methodologies of Education in Professional Training Methodology Theory Experience Problems*, (62), 287–298. <https://doi.org/10.31652/2412-1142-2021-62-287-298>
38. Томіліна А. О. Можливості розробок електронних освітніх ресурсів на базі електронної платформи Moodle під час навчання студентів іноземної мови. *Педагогіка вищої та середньої школи*. 2015. Вип. 46. С. 225-230. URL: http://nbuv.gov.ua/UJRN/PVSSh_2015_46_43
39. Шматко О. В., Поляков А. О., Федорченко В. М. Аналіз методів і технологій розробки мобільних додатків для платформи Android : навч. посіб. Харків : НТУ «ХПІ», 2018. 284 с. URL: <https://repository.kpi.kharkov.ua/server/api/core/bitstreams/a4786ba9-e3ae-431c-bd93-a1be1a0eb64b/content>
40. Шроль, Т.С. (2017) Формування ІКТ-компетентності майбутніх учителів математики. [Дис. канд. пед. наук, Київ, Інститут вищої освіти НАПН України]. https://ihed.org.ua/wp-content/uploads/2019/01/Shrol_20-06-2018_disertac.pdf

41. Шарова, Т., Шаров, С., & Кремінський, Б. (2023). Освітні ресурси для організації STEAM-навчання. Молодь і ринок №3(211), 52-56. DOI: <https://doi.org/10.24919/2308-4634.2023.276173>
42. Biessek, A. Flutter for Beginners. 1st edn. Packt Publishing. 2019. 512 p.
43. Bonnie Eisenman. Learning React Native. Building Native Mobile Apps with JavaScript. Second Edition. O'Reilly Media. 2018. 226 p.
44. Eric Windmill. Flutter in Action. Manning. 2020. 368 p.
45. Flutter documentation (2023). <https://docs.flutter.dev/>
46. Frank Zammetti. Practical React Native: Build Two Full Projects and One Full Game using React Native. Pottstown, 2018. 334 p. <https://doi.org/10.1007/978-1-4842-3939-1>
47. Lomanchenko, A., Khloponin, O., Derkach, T., & Dmytrenko, T. (2021). Technological characteristics of the e-learning systems development. Control, Navigation and Communication Systems. Academic Journal, 2(64), 84-88. <https://doi.org/https://doi.org/10.26906/SUNZ.2021.2.084>
48. Mainkar, P., & Giordano, S. (2019) Google Flutter Mobile Development Quick Start Guide. 1st edn. Packt Publishing, 152 p.
49. Marco L. Napoli. (2019) Beginning Flutter: A Hands On Guide to App Development, p.528
50. Nader, Dabit. (2019). React Native in Action. Developing iOS and Android apps with JavaScript. Manning. 320 p.
51. Simone Alessandria, Brian Kayfitz. (2021, 18, June) Flutter Cookbook: Over 100 proven techniques and solutions for app development with Flutter 2.2 and Dart. Packt Publishing. 646 p.
52. Statista (2023). <https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/>

ДОДАТКИ

Додаток А

Сертифікат учасника XVI Всеукраїнської науково-практичної конференції



Міністерство освіти і науки України
Департамент освіти і науки Рівненської ОДА
Рівненський державний гуманітарний університет
Громадська спілка «Рівне ІТ-освіта»

СЕРТИФІКАТ №2023-166
учасника
XVI Всеукраїнської науково-практичної конференції
**“ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ
В ПРОФЕСІЙНІЙ ДІЯЛЬНОСТІ”**
1 листопада 2023 року, м. Рівне
Соловйов Микола

Завідувач кафедри цифрових технологій
та методики навчання інформатики РДГУ,
голова програмного комітету конференції



проф. Войтович І.С.

Таблиця Б.1

Фасетна класифікація ЕОР

(Женченко, Мельник, Женченко&Прихода, 2020, с.132-134)

КЛАСИФІКАЦІЙНІ ОЗНАКИ		ВИДИ ЕОР
1		2
1. ЗАГАЛЬНІ	1.1. Мова	1.1.1. ЕОР державною мовою 1.1.2. ЕОР іноземною мовою 1.1.3. Перекладні ЕОР
	1.2. Структура	1.2.1. Прості (неподільні) ЕОР 1.2.2. Складені ЕОР 1.2.2.1. ЕОР із лінійними зв'язками між навчальними об'єктами 1.2.2.2. ЕОР із ієрархічними зв'язками між навчальними об'єктами
2. ОСВІТНІ	2.1. Напрямок використання	2.1.1. ЕОР навчального призначення 2.1.2. ЕОР управлінського призначення 2.1.3. ЕОР науково-дослідного призначення
	2.2. Функції в освітньому процесі	2.2.1. <i>Навчальні ЕОР</i> (електронна копія друкованого підручника, електронний підручник, електронний курс лекцій, електронний текст лекцій, електронний навчальний посібник тощо) 2.2.2. <i>Навчально-методичні ЕОР</i> (електронний навчальний план, електронна робоча програма, електронний методичний посібник, електронні методичні рекомендації тощо) 2.2.3. <i>Навчально-практичні ЕОР</i> (електронний практикум, електронний тренажер, збірник віртуальних лабораторних робіт, електронний робочий зошит) 2.2.4. <i>Навчально-ігрові ЕОР</i> (електронні навчальні ігри, електронні освітні ігрові ресурси) 2.2.5. <i>Довідкові ЕОР</i> (електронний довідник, електронний словник, електронна енциклопедія, електронна база даних) 2.2.6. <i>Демонстраційні ЕОР</i> (електронні наочні посібники, електронні презентації, аудіовізуальні твори, статичні та динамічні моделі, доповнена і віртуальна реальність)

Продовження табл. Б.1

	1	2
2. ОСВІТНІ		<p>2.2.7. <i>Контрольні ЕОР</i> (електронні тести, електронні банки контрольних питань із навчальних дисциплін)</p> <p>2.2.8. <i>Допоміжні ЕОР</i> (електронна хрестоматія, електронний збірник нормативних документів, електронні бібліографічні покажчики наукової та навчальної літератури, електронні наукові публікації, електронний щоденник, електронний журнал тощо)</p>
	2.3. Кінцеві користувачі (аудиторія)	<p>2.3.1. ЕОР для вчителів (викладачів)</p> <p>2.3.2. ЕОР для авторів-розробників</p> <p>2.3.3. ЕОР для учнів (студентів)</p> <p>2.3.4. ЕОР для органів управління у сфері освіти</p>
	2.4. Реалізація ігрового складника	<p>2.4.1. Ігрові ЕОР (такі, що мають ігровий складник)</p> <p>2.4.2. ЕОР без ігрового складника</p>
	2.5. Форма (рівень) освіти	<p>2.5.1. ЕОР для формальної освіти</p> <p>2.5.1.1. ЕОР для дошкільної освіти</p> <p>2.5.1.2. ЕОР для загальної середньої освіти (школи)</p> <p>2.5.1.3. ЕОР для вищої освіти (ВНЗ)</p> <p>2.5.1.4. ЕОР для системи підвищення кваліфікації</p> <p>2.5.2. ЕОР для неформальної освіти</p>
3. ВИДАВНИЧО-ТЕХНОЛОГІЧНІ	3.1. Наявність друкованого еквівалента	<p>3.1.1. Електронні копії друкованого видання.</p> <p>3.1.2. Електронні аналоги друкованого видання, збагачені інтерактивними і/або мультимедійними елементами.</p> <p>3.1.3. Самостійні електронні видання.</p>
	3.2. Наявність редакційно-видавничого опрацювання	<p>3.2.1. Електронні навчальні матеріали</p> <p>3.2.2. Електронні видання</p>
	3.3. Розробник	<p>3.3.1. ЕОР, розроблені видавцями.</p> <p>3.3.2. ЕОР, розроблені педагогічними працівниками</p> <p>3.3.3. ЕОР, розроблені ІТ-компаніями</p> <p>3.3.4. ЕОР, розроблені видавцями у співпраці з ІТ-компаніями</p> <p>3.3.5. ЕОР, розроблені видавцями у співпраці з педагогічними працівниками</p> <p>3.3.6. ЕОР, розроблені ІТ-компаніями у співпраці з педагогічними працівниками</p> <p>3.3.7. ЕОР, розроблені видавцями у співпраці з ІТ-компаніями та педагогічними працівниками</p>

Продовження табл. Б.1

	1	2
	3.4. Технологія розроблення	3.4.1. EOP-документи 3.4.2. EOP-програмні засоби (прикладні програми і системні програмні засоби)
	3.5. Технологія використання	3.5.1. Локальні (офлайнові) EOP 3.5.2. Мережне (онлайнові) EOP 3.5.3. EOP комбінованого використання
4. ТЕХНІЧНІ	4.1. Форма існування	4.1.1. Дані (та їх сукупності) 4.1.2. Комп'ютерні програми
	4.2. Подання даних	4.2.1. EOP із послідовними даними 4.2.2. EOP із реляційними даними 4.2.3. EOP із індексованими даними 4.2.4. EOP із мережними даними 4.2.5. EOP із об'єктно орієнтованими даними 4.2.6. EOP із даними-сутностями 4.2.7. EOP-форматований файл
	4.3. Спосіб використання даних	4.3.1. EOP із даними для індивідуального використання. 4.3.2. EOP із даними для використання багатьма послідовними користувачами 4.3.3. EOP із даними для використання конкурентними взаємовиключними користувачами
	4.4. Тип подання і природа основної інформації	4.4.1. Мономедійні EOP 4.4.1.1. Текстові 4.4.1.2. Образотворчі 4.4.1.3. Нотні 4.4.1.4. Картографічні 4.4.1.5. Звукові 4.4.2. Мультимедійні EOP
	4.5. Характер взаємодії з користувачем	4.5.1. Детерміновані (статичні) EOP 4.5.2. Недетерміновані (інтерактивні) EOP
	4.6. Формат	4.6.1. Моноформатні EOP 4.6.2. Мультиформатні EOP
	4.7. Залежність від операційних систем (мобільних платформ)	4.7.1. Платформозалежні EOP 4.7.2. Платформонезалежні EOP
	4.8. Стабільність	4.8.1. EOP із постійними змінами 4.8.2. EOP із інкрементними змінами (нарощування) 4.8.3. EOP із маловірогідними змінами
5. ПРАВОВІ	5.1. Обмеженість доступу	5.1.1. Загальнодоступні EOP 5.1.2. EOP з обмеженим доступом (персональні й корпоративні)
	5.2. Тип ліцензії	5.2.1. EOP з вільною (відкритою) ліцензією 5.2.2. EOP з пропріетарною (закритою) ліцензією

Загальні вимоги до ЕОР

(відповідно до Про внесення змін до Положення про електронні освітні ресурси: Наказ Міністерства освіти і науки від 29.05.2019 № 749.

<https://ips.ligazakon.net/document/view/Re33637?an=1>

III. Загальні вимоги до ЕОР:

1. Загальні вимоги

- функціональність;
- безпечність;
- надійність функціонування;
- зручність використання для користувача;
- крос-платформність;
- відповідність засадам реалізації принципів державної політики цифрового розвитку;
- відповідність законодавству України щодо захисту авторських прав;
- відповідність міжнародним стандартам (Experience API тощо).

2. Електронний навчальний посібник може містити:

- теоретичний матеріал з його систематизованим викладенням, що відповідає навчальній програмі;
- контрольні запитання до кожної теми;
- практико-орієнтовані компетентнісні завдання різних рівнів складності;
- тестові завдання;
- словник;
- список основної та додаткової літератури тощо.

3. Електронний освітній ігровий ресурс (ЕОІР) може містити:

- теоретичну та практичну частини, що відповідають навчальній програмі;
- практико-орієнтовані компетентнісні завдання різних рівнів складності, подані в ігровій формі;
- інтерактивні елементи;
- тестові завдання тощо.

Додаток Г

Тематика лекційних та лабораторних занять з навчальної дисципліни
«Програмування мобільних застосунків»

Змістовий модуль «Розробка кросплатформних мобільних застосунків із використанням фреймворку Flutter»

Тема 1. Вступ у Flutter та Dart.

Встановлення Flutter та Android Studio (Windows). Встановлення Android Emulator (Windows). Встановлення VS Code (Windows). Огляд основних можливостей і правила мови Dart (змінні, базові типи даних, умовні оператори і цикли, функції, класи і інтерфейси, Mixins, Generics, основи асинхронних операцій в Dart). Огляд папок у Flutter додатку. Hot reload і Hot restart.

Тема 2. Flutter. Введення в віджети (Widgets). Основи компоновки UI. Робота із списками. Кастомізація UI.

Введення у віджети. MaterialApp. Scaffold. Типи віджетів. Stateless Widget і його життєвий цикл. StatefulWidget і його життєвий цикл. Контейнери та управління компонованням. Padding. Align і Center. Container. Row і Column. Expanded. Stack.

Статичний ListView. Динамічний ListView та відображення заголовку.

Теми, детальна кастомізація UI. Створення власного віджета. Робота із ресурсами: Image, Font

Тема 3. Навігація між сторінками і передача даних у Flutter. Додавання анімації.

Route і Navigator. Передача даних. Проста навігація. Навігація по імені. Проста передача даних між екранами. Передача даних при навігації по імені. Передача даних з onGenerateRoute. Передача даних на попередній екран.

AnimationController і Animation. Одночасна анімація. Поетапна анімація. Контроль за ходом анімації. Анімація на основі кривих (Curves). Анімація Route.

Тема 4. Взаємодія з користувачем у Flutter. Кнопки, обробка дій користувача. Пакети і залежності

Form. TextField і TextFormField. Стилiзація полів введення. Обмеження введення в поля форми. Отримання тексту з полів форми. Валiдація полів форми. Випадаючий список. «Фокусування» полів введення.SnackBar. AlertDialog. Використання жестів GestureDetector. Відправлення форми. Кнопки, обробка дій користувача. Використання об'єктно-орієнтованого програмування на Flutter.

Управління версіями пакетів. Неопубліковані пакети

Тема 5. Робота з мережею у Flutter. Робота з WebView.

Робота з HTTP. Введення в JSON. Ручна серіалізація JSON. Створення моделі та відображення даних. Автоматично генерована серіалізація JSON. Типи станів. Ephemeral State (локальний стан). Vanilla (setState ()).

Application State (глобальний стан). InheritedWidget. ScopedModel. Provider.

Управління веб-сторінками в WebView. Відстеження змін в WebView. Впровадження Javascript в WebView. Робота з локальними Веб-сторінками, CSS і JS

Тема 6. Збереження даних на пристрої засобами Flutter. Clean Architecture.

Зберігання даних на Flutter. Робота з файлами (Читання / Запис). Робота з Preferences і Firebase. Плагін shared_preferences. Робота з SQLite. Написання чистого коду та реалізація патерну Repository.

Тема 7. Архітектурні патерни. BLoC (Business Logic Component). Stream. DI, тести

BLoC на потоках (Stream). BLoC (flutter_bloc 4.0). Flutter BLoC 4.0. Робота з мережею. Flutter. Flutter Cubit 6.0. Робота з мережею.

Структура програми на базі BLoC. Рівень домену: Entity і Equatable, Repository, UseCases. Рівень даних: Models, Remote data source, Local data

source, Repository. Рівень представлення: Рівень представлення: Cubit. Впровадження залежностей (Dependency Injection). Відображення списку персонажів. Детальна інформація про персонажа. Пошук персонажа

Використання патерна Dependency Injection. Написання Unit-тестів, widget (UI-тести), інтеграційних тестів

Тема 8. Використання нативних функцій пристрою. Підготовка до релізу. Підпис застосунку. Особливості Android проєкта. Особливості iOS проєкта

Використання нативних функцій пристрою (камери, карти, місцезнаходження тощо). Підготовка до релізу. Підпис застосунку. Запуск коду Native Swift, ObjectiveC, Java або Kotlin. Публікація в App Store.

Приклад інструкцій до виконання лабораторної роботи

Тема: Route і Navigator. Передача даних. Проста навігація. Навігація по імені. Проста передача даних між екранами. Передача даних при навігації по імені. Передача даних з onGenerateRoute. Передача даних на попередній екран. AnimationController і Animation. Одночасна анімація. Поетапна анімація. Контроль за ходом анімації. Анімація на основі кривих (Curves). Анімація Route..

Теоретичні відомості

1. <https://flutter.dev/docs/cookbook/navigation>

Хід роботи

Тренувальне завдання 1. Анімація віджет на різних екранах (<https://flutter.dev/docs/cookbook/navigation/hero-animations>).

Реалізація завдання

Використовуйте віджет `Hero` для анімації віджета з одного екрана на інший відповідно до наступних кроків:

- 1) Створіть два екрани з одним зображенням.

```
import 'package:flutter/material.dart';
```

```
class mainScreen extends StatelessWidget {
  const mainScreen({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Main Screen'),
      ),
      body: GestureDetector(
        onTap: () {
          Navigator.push(context, MaterialPageRoute(builder: (context) {
            return const detailScreen();
          }));
        },
      ),
      child: Image.network(
        'https://picsum.photos/250?image=9',
      ),
    );
  }
}
```

```
class detailScreen extends StatelessWidget {
  const detailScreen({Key? key}) : super(key: key);
```

```
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: GestureDetector(
        onTap: () {
          Navigator.pop(context);
        },
      ),
      child: Center(
        child: Image.network(
          'https://picsum.photos/250?image=9',
        ),
      ),
    );
  }
}
```

```

    ),
  );
}
}

```

2) Додайте віджет **Hero** на перший екран.

```

Hero(
  tag: 'imageHero',
  child: Image.network(
    'https://picsum.photos/250?image=9',
  ),
)

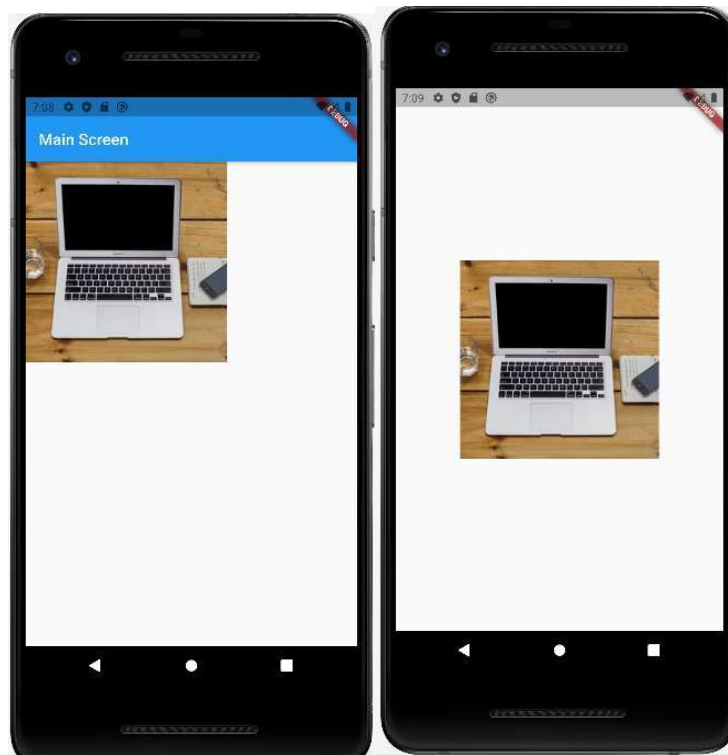
```

3) Додайте віджет **Hero** на другий екран.

```

Hero(
  tag: 'imageHero',
  child: Image.network(
    'https://picsum.photos/250?image=9',
  ),
)

```



Екран 1

Екран 2

Тренувальне завдання 2. Перехід на новий екран і повернення назад (<https://flutter.dev/docs/cookbook/navigation/navigation-basics>).

Більшість програм містять кілька екранів для відображення різних типів інформації. Наприклад, програма може мати екран, на якому відображаються товари. Коли користувач натискає на зображення товару, то на новому екрані відображаються відомості про продукт.

ЗАУВАЖЕННЯ! У Flutter екрани (*screens*) та сторінки (*pages*) називаються маршрутами (*routes*). В Android маршрут еквівалентний *Activity*. В iOS маршрут еквівалентний *ViewController*. У Flutter маршрут – це лише віджет.

Наступні пункти описують кроки реалізації переміщення між двома маршрутами (*routes*):

- 1) Потрібно створити два маршрути.
- 2) Перейти до другого маршруту за допомогою `Navigator.push()`.
- 3) Повернутися до першого маршруту за допомогою `Navigator.pop()`.

Реалізація завдання

1) Спочатку створіть два маршрути для роботи. Оскільки це базовий приклад, кожен маршрут містить лише одну кнопку. Натискання кнопки на першому маршруті переходить до другого маршруту. Натискання кнопки на другому маршруті повертається до першого маршруту.

```
class FirstRoute extends StatelessWidget {
  const FirstRoute({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('First Route'),
      ),
      body: Center(
        child: ElevatedButton(
          child: const Text('Open route'),
          onPressed: () {
            // Navigate to second route when tapped.
          },
        ),
      ),
    );
  }
}
```

```
class SecondRoute extends StatelessWidget {
  const SecondRoute({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("Second Route"),
      ),
      body: Center(
        child: ElevatedButton(
          onPressed: () {
            // Navigate back to first route when tapped.
          },
          child: const Text('Go back!'),
        ),
      ),
    );
  }
}
```

2) Перейдіть до другого маршруту за допомогою *Navigator.push()*. Для переходу на новий маршрут скористайтеся *Navigator.push()* методом *push()*. Метод додає до *Route* пакету маршрутів , якими керують

Navigator. Можна створити свій власний Route або скористатися а *MaterialPageRoute*, що корисно, оскільки він переходить до нового маршруту за допомогою анімації, специфічної для платформи.

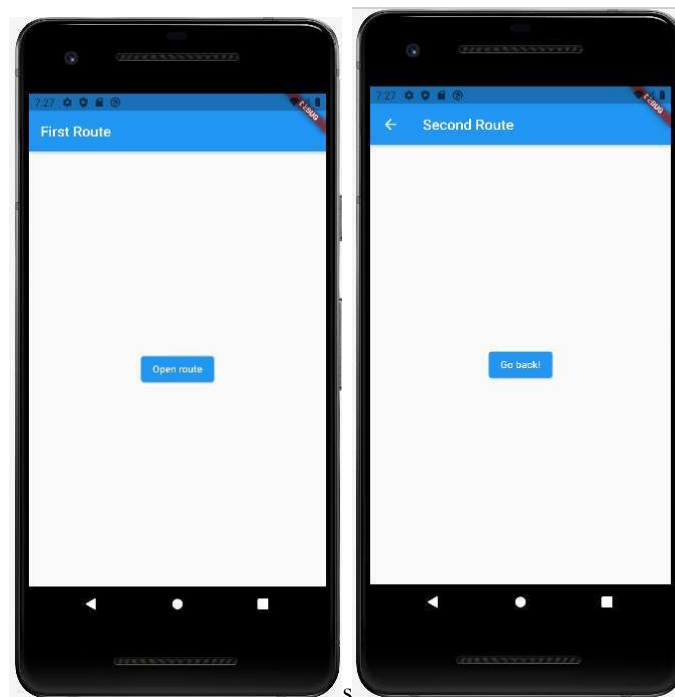
У методі *build()* віджета *FirstRoute* оновить *onPressed()* зворотний виклик:

```
onPressed: () {
  Navigator.push(
    context,
    MaterialPageRoute(builder: (context) => const SecondRoute()),
  );
},
```

3) Поверніться до першого маршруту за допомогою *Navigator.pop()*. Як закрити другий маршрут і повернутися до першого? За допомогою методу *Navigator.pop()*. Метод *pop()* видаляє поточний *Route* з стека маршрутів, якими керують *Navigator*.

Щоб реалізувати повернення до вихідного маршруту, оновить *onPressed()* зворотний виклик у віджеті *SecondRoute*:

```
onPressed: () {
  Navigator.pop(context);
},
```



Перехід між екранами (Routes)

Тренувальне завдання 3. Переміщення за допомогою імен маршрутів routes (<https://flutter.dev/docs/cookbook/navigation/named-routes>)

Якщо потрібно перейти на той самий екран у багатьох частинах програми, такий підхід може призвести до дублювання коду. Рішення полягає у визначенні *іменованого маршруту* та використанні назви маршруту для навігації.

Для роботи з іменованими маршрутами скористаємося функцією *Navigator.pushNamed()*. Реалізація переходу між двома іменованими маршрутами передбачає виконання таких дій:

1. Створення двох екранів.
2. Визначення маршрутів.
3. Перехід до другого екрана за допомогою *Navigator.pushNamed()*.
4. Поверніться до першого екрана за допомогою *Navigator.pop()*.

Реалізація завдання

1) *Створіть два екрани для роботи.* Перший екран містить кнопку, яка переходить до другого екрана. Другий екран містить кнопку, яка повертається до першого.

```

import 'package:flutter/material.dart';

class FirstScreen extends StatelessWidget {
  const FirstScreen({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('First Screen'),
      ),
      body: Center(
        child: ElevatedButton(
          onPressed: () {
          },
          child: const Text('Launch screen'),
        ),
      ),
    );
  }
}

class SecondScreen extends StatelessWidget {
  const SecondScreen({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Second Screen'),
      ),
      body: Center(
        child: ElevatedButton(
          onPressed: () {
          },
          child: const Text('Go back!'),
        ),
      ),
    );
  }
}

```

2) Далі визначте маршрути, надавши конструктору [MaterialApp](#) додаткові властивості: *initialRoute* і *routes*. У властивості *initialRoute* визначається, який маршрут повинен відобразитися після запуску додатку. Властивість *routes* визначає доступні іменовані маршрути і віджети для збірки при навігації по цих маршрутах.

```

void main() {
  runApp(
    MaterialApp(

```

```

title: 'Named Routes Demo',
initialRoute: '/',
routes: {
  '/': (context) => const FirstScreen(),
  '/second': (context) => const SecondScreen(),
},
),
);
}

```

ЗАУВАЖЕННЯ! Коли використовується `initialRoute`, то властивість `home` в `MaterialApp` не використовується.

3) *Перейдіть на другий екран.* Коли віджети та маршрути встановлені, запусіть навігацію за допомогою методу `Navigator.pushNamed()`. Даний метод вказує Flutter, що потрібно створити віджет, визначений у таблиці `routes`, і запустити екран.

У методі `build()` віджету `FirstScreen` оновіть `onPressed()` зворотний виклик:

```

onPressed: () {
  // Navigate to the second screen using a named route.
  Navigator.pushNamed(context, '/second');
}

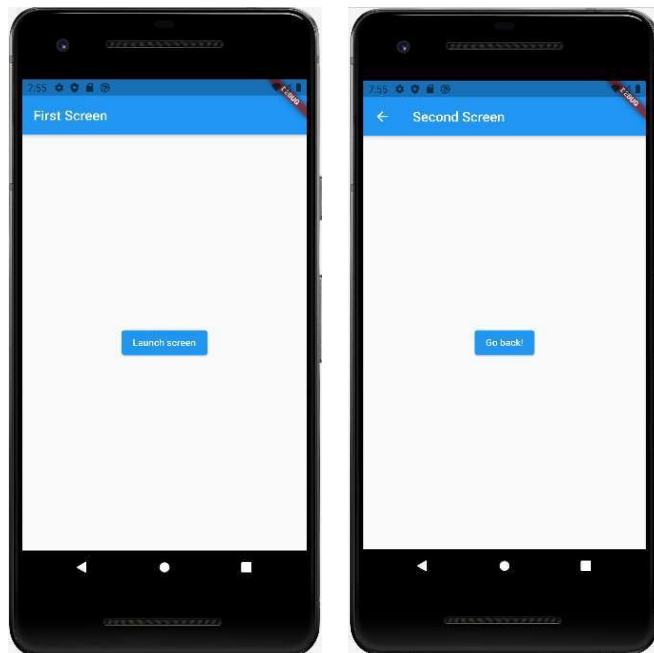
```

3) *Поверніть на перший екран.* Для переходу до першого екрану використовується функція `Navigator.pop()`.

```

onPressed: () {
  // Navigate back to the first screen by popping the current route
  // off the stack.
  Navigator.pop(context);
}

```



Тренувальне завдання 4. Передача аргументів (даних) іменованому маршруту (<https://flutter.dev/docs/cookbook/navigation/navigate-with-arguments>).

Navigator забезпечує здатність орієнтуватися в іменованому маршруті з будь-якої частини програми з використанням загального ідентифікатора. У деяких випадках також може знадобитися передати аргументи

іменованому маршруту. Наприклад, можете перейти до маршруту `/user` та передати йому інформацію про користувача.

Це завдання можна виконати за допомогою параметра `arguments` методу `Navigator.pushNamed()`, витягнувши аргументи за допомогою методу у `ModalRoute.of()` або всередині функції `onGenerateRoute()`, наданої конструктору `MaterialApp` або `CupertinoApp`.

Для того, щоб передати аргументи через іменованій маршрут і прочитати аргументи із використанням `ModalRoute.of()` і `onGenerateRoute()` потрібно виконати наступні кроки:

1. Визначити аргументи, які потрібно передати.
2. Створити віджет, який витягує аргументи.
3. Зареєструвати віджет у таблиці `routes`.
4. Перейти до віджета

Реалізація завдання

1. Визначаємо аргументи, які потрібно передати:

```
class ScreenArguments {
  final String title;
  final String message;

  ScreenArguments(this.title, this.message);
}
```

2. Створюємо віджет, який витягне та відобразить `title` і `message` з `ScreenArguments`. Для доступу до `ScreenArguments` скористаємося методом `ModalRoute.of()`. Цей метод повертає поточний маршрут з аргументами.

```
class ExtractArgumentsScreen extends StatelessWidget {
  const ExtractArgumentsScreen({Key? key}) : super(key: key);

  static const routeName = '/extractArguments';

  @override
  Widget build(BuildContext context) {

    final args = ModalRoute.of(context)!.settings.arguments as ScreenArguments;

    return Scaffold(
      appBar: AppBar(
        title: Text(args.title),
      ),
      body: Center(
        child: Text(args.message),
      ),
    );
  }
}
```

3. Реєструємо віджет у таблиці `routes`, додавши запис до наданого `routes` віджета `MaterialApp`. Визначаємо `routes`, який віджет повинен бути створений на основі імені маршруту.

```
MaterialApp(
  routes: {
    ExtractArgumentsScreen.routeName: (context) =>
```

- ```

 const ExtractArgumentsScreen(),
 },

```
4. Виконуємо перехід до віджета `ExtractArgumentsScreen` після натиснення кнопки за допомогою `Navigator.pushNamed()`. Передача аргументи маршруту виконується через властивість `arguments`. У `ExtractArgumentsScreen` витягуються з цих аргументів `title` і `message`.

```

ElevatedButton(
 onPressed: () {
 Navigator.pushNamed(
 context,
 ExtractArgumentsScreen.routeName,
 arguments: ScreenArguments(
 'Extract Arguments Screen',
 'This message is extracted in the build method.',
),
);
 },
 child: const Text('Navigate to screen that extracts arguments'),
),

```

Крім того, потрібно витягнути аргументи за допомогою `onGenerateRoute`. Замість отримання аргументів безпосередньо всередині віджета, також можна витягти аргументи всередині функції `onGenerateRoute()` функції та передати їх віджету. Функція `onGenerateRoute()` створює правильний маршрут, виходячи з заданих `RouteSettings`.

```

MaterialApp(
 routes: {
 ExtractArgumentsScreen.routeName: (context) =>
 const ExtractArgumentsScreen(),
 },
 onGenerateRoute: (settings) {
 if (settings.name == PassArgumentsScreen.routeName) {
 final args = settings.arguments as ScreenArguments;
 return MaterialPageRoute(
 builder: (context) {
 return PassArgumentsScreen(
 title: args.title,
 message: args.message,
);
 },
);
 }
 assert(false, 'Need to implement ${settings.name}');
 return null;
 },
 title: 'Navigation with Arguments',
 home: const HomeScreen(),
);

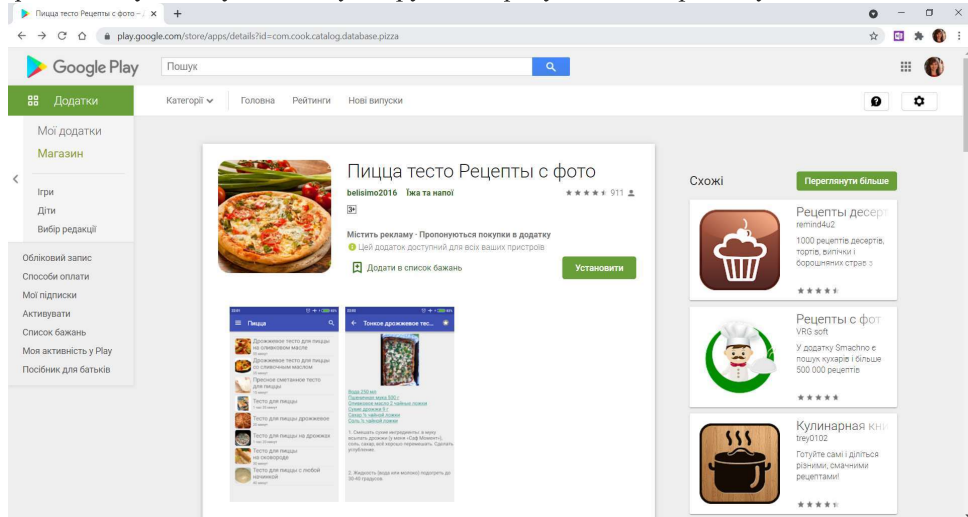
```



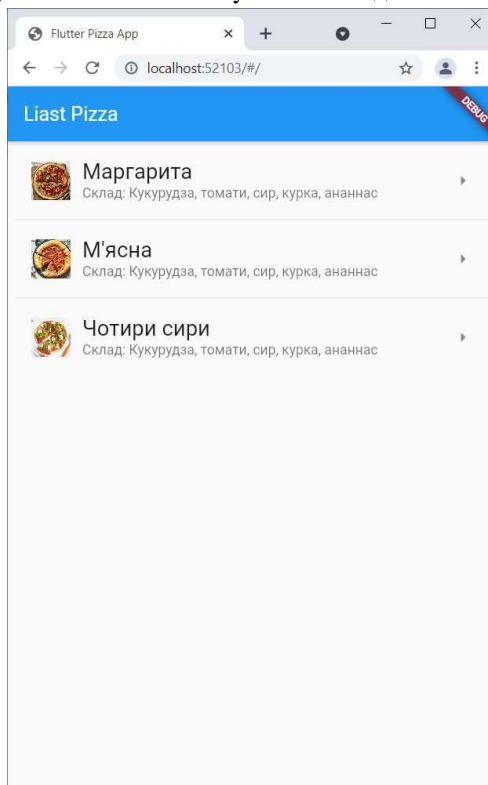
Тренувальне завдання 5. Повернення даних з нового (другого екрану) (<https://flutter.dev/docs/cookbook/navigation/returning-data>)

### ВПРАВА ДЛЯ САМОСТІЙНОГО ВИКОНАННЯ

Реалізувати застосунок рецептів піци (або вибрати самостійно тему). Наприклад, створити список піц із коротким описом (<https://play.google.com/store/apps/details?id=com.cook.catalog.database.pizza>) (див. завд 4. Лаб.12). При виборі елемента виконується запуск другого екрану із описом рецепту.



Застосунок може мати наступний вигляд:



Скрін з освітнього відео з демонстрацією розробки мобільного застосунку

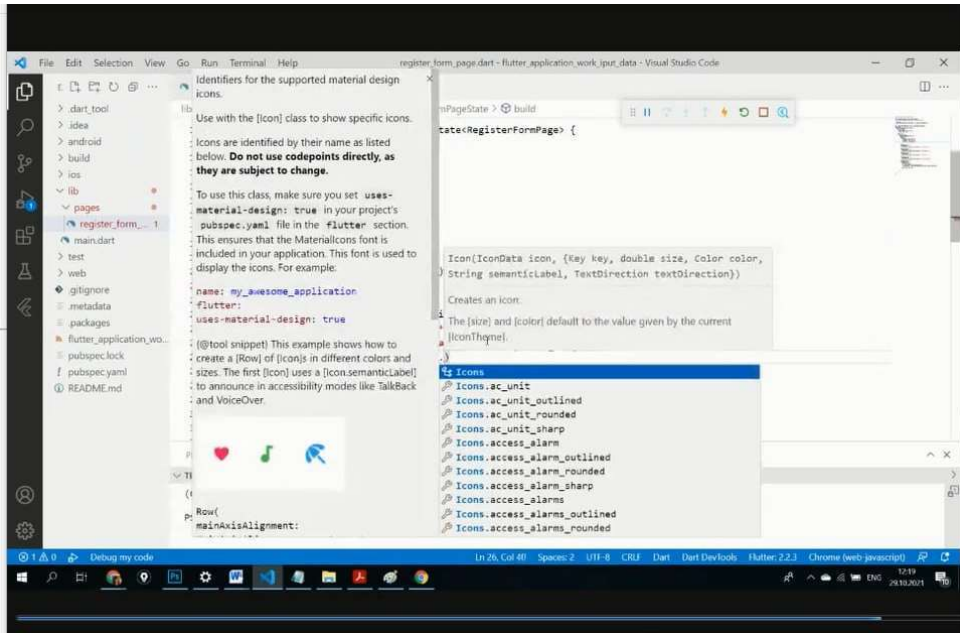


Рис. Ж.1

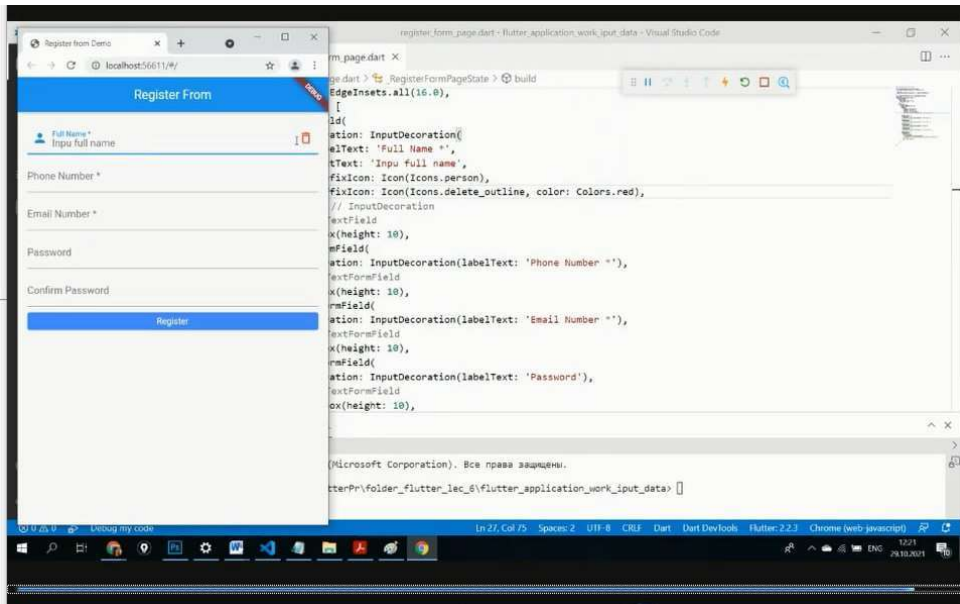


Рис. Ж.2

## Запитання підсумкового тесту

**Питання 1**

Відповіді ще не було

Макс. оцінка до 0,50

 Відмітити питання

 Редагувати питання

Якого елемента НЕМА в структурі проєкту на Flutter?

Виберіть одну відповідь:

- ios
- blackberry
- android
- .idea
- web

**Питання 2**

Відповіді ще не було

Макс. оцінка до 0,50

 Відмітити питання

 Редагувати питання

Як установити шрифт тексту Roboto.ttf?


Виберіть одну відповідь:

- fontFamily.Roboto
- FontFamily("Roboto")
- fontFamily("Roboto")
- fontFamily(Roboto)
- FontFamily("Roboto")

**Питання 3**

Відповіді ще не було

Макс. оцінка до 0,50

 Відмітити питання

 Редагувати питання

Який параметр в `ListView.separated()` приймає об'єкт `IndexedWidgetBuilder`, який створює елементи списку?

Виберіть одну відповідь:

- itemCount
- reverse
- itemBuilder
- scrollDirection
- padding

**Питання 4**

Відповіді ще не було

Макс. оцінка до 0,50

🚩 Відмітити питання

⚙️ Редагувати питання

Який із параметрів віджета Row задає простір, що займає віджет по горизонталі?

Виберіть одну відповідь:

- verticalDirection
- children
- crossAxisAlignment
- textDirection
- textBaseline
- mainAxisAlignment
- mainAxisAlignment

**Питання 5**

Відповіді ще не було

Макс. оцінка до 0,50

🚩 Відмітити питання

⚙️ Редагувати питання

Параметр trailing конструктора ListTile у Flutter ...

Виберіть одну відповідь:

- встановлює віджет, який відображається після основного вмісту
- встановлює віджет, який міститься перед елементом списку
- встановлює віджет, який представляє основний вміст ListTile
- встановлює додатковий віджет, розташований під основним вмістом
- представляє тип bool і вказує, чи елемент списку міститиме трирядковий текст

**Питання 6**

Відповіді ще не було

Макс. оцінка до 0,50

🚩 Відмітити питання

⚙️ Редагувати питання

Параметр onTap конструктора ListTile у Flutter ...

Виберіть одну відповідь:

- представляє тип bool і вказує, чи буде виділено цей елемент списку
- визначає функцію типу GestureDetectorCallback, яка викликається при натисканні на цей елемент списку
- визначає функцію типу GestureDetectorLongPressCallback, яка викликається при довгому натисканні на цей елемент списку
- визначає форма курсору, яку отримує покажчик миші при наведенні на цей елемент списку. Представляє тип MouseCursor
- нема вірної відповіді

**Питання 7**

Відповіді ще не було

Макс. оцінка до 0,50

🚩 Відмітити питання

⚙️ Редагувати питання

Для успадкування одного класу від іншого застосовується якийсь оператор?

Виберіть одну відповідь:

- implements
- extending
- extends
- inherits
- including

**Питання 8**

Відповіді ще не було

Макс. оцінка до 0,50

🚩 Відмітити питання

⚙️ Редагувати питання

Вкажіть параметр віджету **Text**, який є обов'язковим для визначення віджета.

Виберіть одну відповідь:

- textAlign
- textStyle
- textDirection
- text
- textWidthBasis

**Питання 9**

Відповіді ще не було

Макс. оцінка до 0,50

🚩 Відмітити питання

⚙️ Редагувати питання

Який параметр конструктора **ListTile** у Flutter встановлює колір, який отримає **ListTile** при виділенні?

Виберіть одну відповідь:

- hoverColor
- selectedTileColor
- focusColor
- ніякий
- tileColor



**Питання 10**

Відповіді ще не було

Макс. оцінка до 0,50

 Відмітити питання

 Редагувати питання

Тип даних змінної в Dart можна змінити, якщо оголосити її змінну з ключовим словом **dynamic**

Виберіть одну відповідь:


- Правильно
- Неправильно

**Питання 11**

Відповіді ще не було

Макс. оцінка до 0,50

 Відмітити питання

 Редагувати питання

Вкажіть параметр віджету **Text**, що визначає, чи буде текст переноситися на наступний рядок при досягненні відведеної для нього довжини.

Виберіть одну відповідь:


- overflow
- textScaleFactor
- textDirection
- textAlign
- softWrap

**Питання 12**

Відповіді ще не було

Макс. оцінка до 0,50

 Відмітити питання

 Редагувати питання

Який параметр віджета **ElevatedButton** у Flutter визначає, чи буде кнопка отримувати фокус за замовчуванням?

Виберіть одну відповідь:


- autofocus
- onPressed
- clipBehavior
- focusNode
- onLongPress

**Питання 13**

Відповіді ще не було

Макс. оцінка до 0,50

 Відмітити питання

 Редагувати питання

Який із параметрів конструктора **TextField** у Flutter задає стиль тексту?

Виберіть одну відповідь:

- style
- textDirection
- textAlignVertical
- textAlign
- decoration

**Питання 14**

Відповіді ще не було

Макс. оцінка до 0.50

🚩 Відмітити питання

⚙️ Редагувати питання

Вкажіть параметр віджету **Text**, що встановлює мінімальну висоту рядка відносно базової лінії.

Виберіть одну відповідь:

- textDirection
- style
- strutStyle
- textAlign
- locale

**Питання 15**

Відповіді ще не було

Макс. оцінка до 1.00

🚩 Відмітити питання

⚙️ Редагувати питання

Яким буде виведений рядок «Hello!» в результаті виконання наступного фрагменту коду:

```
import 'package:flutter/material.dart';

void main() {
 runApp(Align(
 alignment: Alignment.centerLeft,
 child:Text(
 'Hello!',
 textDirection: TextDirection.ltr,
 style: TextStyle(fontSize: 28)
)
)
);
}
```

Виберіть одну відповідь:

- Текст буде виведений і розміщений зліва по горизонталі і в центрі по вертикалі екрану
- Текст буде виведений і розміщений справа по горизонталі і в центрі по вертикалі екрану
- Текст буде виведений і розміщений в центрі по горизонталі і по вертикалі екрану
- Текст буде виведений і розміщений в центрі по горизонталі і зліва по вертикалі екрану
- Текст буде виведений і розміщений в центрі по горизонталі і справа по вертикалі екрану

**Питання 16**

Відповіді ще не було

Макс. оцінка до 1,00

Відмітити питання

Редагувати питання

Для встановлення вирівнювання у віджеті *Align* застосовується параметр *alignment*. Встановіть відповідність між значеннями аргумента *alignment* визначеного *AlignmentGeometry* та парою *Alignment(x,y)*

|                                   |            |
|-----------------------------------|------------|
| <code>Alignment.topRight</code>   | Вибрати... |
| <code>Alignment.center</code>     | Вибрати... |
| <code>Alignment.topCenter</code>  | Вибрати... |
| <code>Alignment.centerLeft</code> | Вибрати... |

**Питання 17**

Відповіді ще не було

Макс. оцінка до 1,00

Відмітити питання

Редагувати питання

Що буде виведено в результаті виконання наступної програми (мова Dart):

```
void main() {
 int? num1 = 8;
 int num2 = 8;
 if (num1 > num2) {
 print("Перше число більше другого");
 } else if (num1 < num2) {
 print("Перше число менше другого");
 } else if (num1 == num2) {
 print("Перше число дорівнює другому");
 } else {
 print("Невизначений результат");
 }
}
```

Виберіть одну відповідь:

- Перше число більше другого
- Перше число дорівнює другому
- Програма завершиться помилкою
- Перше число менше другого
- Невизначений результат



**Питання 18**

Відповіді ще не було

Макс. оцінка до 1,00

Відмітити питання

Редагувати питання

Яка із наведених функцій оголошена НЕкоректно?

Виберіть одну відповідь:

- Всі коректно визначені
- `String getName async()`
- `Future getName() async()`
- `Void getName() async()`
- `Future<String> getName async()`

**Питання 19**

Відповіді ще не було

Макс. оцінка до 1,00

Відмітити питання

Редагувати питання

Встановіть відповідність між конструкторами обов'язкового параметра *padding* (класу *EdgeInsetsGeometry*) віджета *Padding*, що задає відступи вкладеного в нього віджета *child*, та їх призначенням

Визначає простір, який відповідає параметру *padding*

Вибрати...

Встановлює одне значення *double* для всіх

чотирьох відступів (зліва, зверху, праворуч і знизу)

Вибрати...

Встановлює симетрично верхній і нижній відступ по вертикалі та відповідно лівий і правий по горизонталі.

Вибрати...

**Питання 20**

Відповіді ще не було

Макс. оцінка до 1,00

🚩 Відмітити питання

⚙️ Редагувати питання

Що виведе на консоль наступна програма (мова Dart)?

```
void main() {
 int? num;
 switch (num) {
 case 0:
 print("число дорівнює 0");
 break;
 case 1:
 print("число дорівнює 1");
 break;
 case 2:
 print("число дорівнює 2");
 break;
 }
}
```

Виберіть одну відповідь:

- число дорівнює 0
- програма завершиться помилкою
- число дорівнює 1
- Нічого
- число дорівнює 2

**Питання 21**

Відповіді ще не було

Макс. оцінка до 1,00

 Відмітити питання

 Редагувати питання

Вкажіть, як правильно прописати у файлі `pubspec.yaml` завантажений в проєкт на Flutter шрифт `IndieFlower-Regular.ttf`, розміщення якого в структурі проєкту представлено на рисунку нижче, щоб його можна було використовувати для зміни шрифту тексту на екрані застосунку.

```

> .dart_tool
> .idea
> android
▼ assets
 ▼ fonts
 ▲ IndieFlower-Regular.ttf
 > images
> build
> ios
▼ lib
 ● main.dart
> test
> web

```

Виберіть одну відповідь:

- fonts:
  - family: IndieFlower
  - asset: fonts/IndieFlower-Regular.ttf
- fonts:
  - family: IndieFlower
  - asset: assets/fonts/IndieFlower-Regular.ttf
- fonts:
  - family: IndieFlower
  - fonts:
    - asset: assets/fonts/IndieFlower-Regular.ttf
- fonts:
  - family: IndieFlower
  - fonts:
    - asset: fonts/IndieFlower-Regular.ttf
- family: IndieFlower
  - fonts:
    - asset: assets/fonts/IndieFlower-Regular.ttf

**Питання 22**

Відповіді ще не було

Макс. оцінка до 1,50

 Відмітити питання

 Редагувати питання

Вкажіть, який код визначає структуру віджетів, що зображений на рисунку:



Виберіть одну відповідь:

- Column(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: <Widget>[  
 Text(  
 'Info with ua.sinoptik.ua',  
 style: TextStyle(fontSize: 15.0),  
 ),  
 Column(  
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
 children: <Widget>[  
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),  
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),  
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),  
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),  
 Icon(Icons.star, size: 15.0, color: Colors.black),  
 ],  
 ),  
 ],  
 );

```
○ Row(
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,
 children: <Widget>[
 Text(
 'Info with ua.sinoptik.ua',
 style: TextStyle(fontSize: 15.0),
),
 Column(
 mainAxisAlignment: MainAxisAlignment.min,
 children: <Widget>[
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
],
),
],
);

○ Row(
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,
 children: <Widget>[
 Text(
 'Info with ua.sinoptik.ua',
 style: TextStyle(fontSize: 15.0),
),
 Row(
 mainAxisAlignment: MainAxisAlignment.min,
 children: <Widget>[
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
],
),
],
);
```

```

○ Row(
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,
 children: <Widget>[
 Text(
 'Info with ua.sinoptik.ua',
 style: TextStyle(fontSize: 15.0),
),
 Row(
 mainAxisAlignment: MainAxisAlignment.min,
 children: <Widget>[
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.black),
],
),
],
);

○ Row(
 mainAxisAlignment: MainAxisAlignment.spaceEvenly,
 children: <Widget>[
 Text(
 'Info with ua.sinoptik.ua',
 style: TextStyle(fontSize: 15.0),
),
 Column(
 mainAxisAlignment: MainAxisAlignment.min,
 children: <Widget>[
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.yellow[600]),
 Icon(Icons.star, size: 15.0, color: Colors.black),
],
),
],
);

```

## Питання 23

Відповіді ще не було

Макс. оцінка до 1,50

Відмітити питання

Редагувати питання

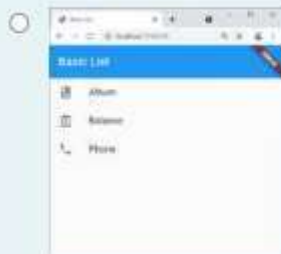
Вкажіть, яким буде результат виконання програми, що будує список віджетів у Flutter?

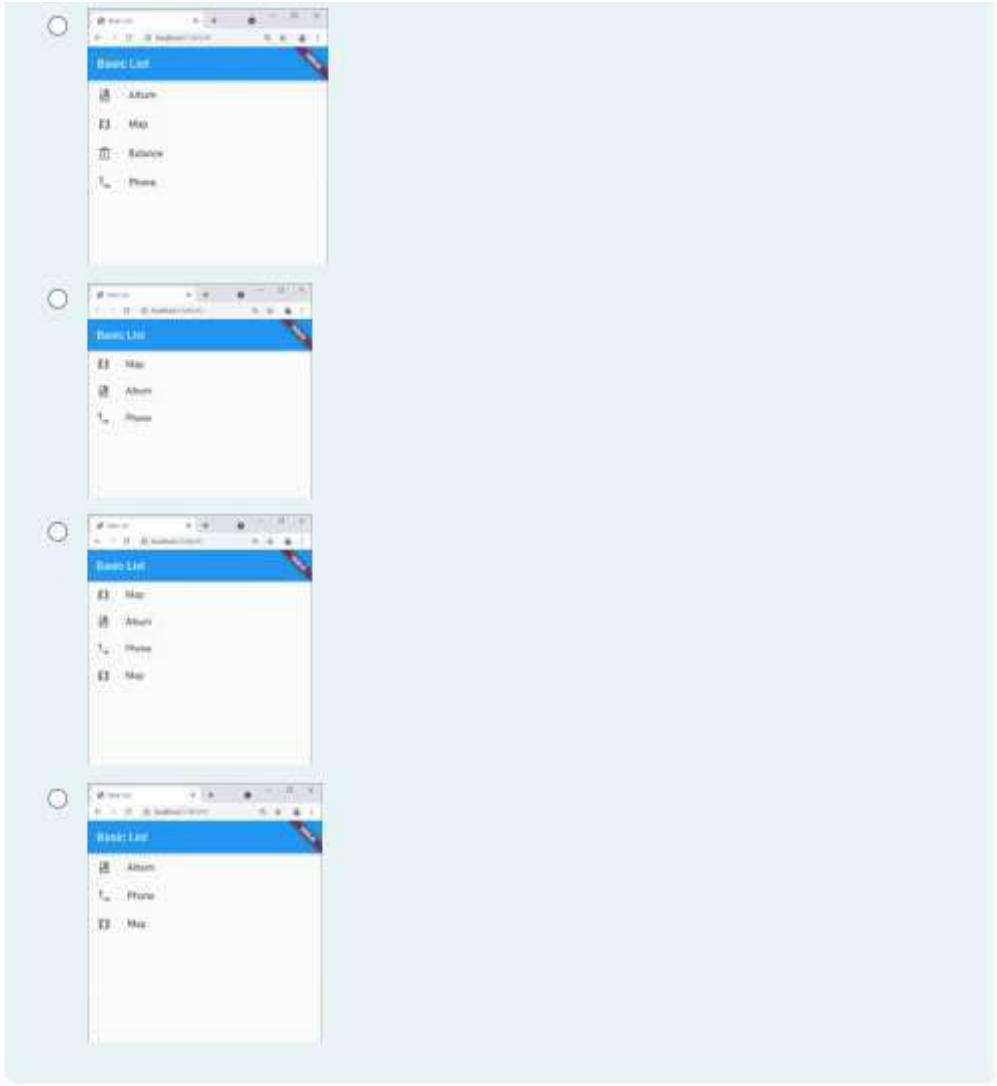
```
import 'package:flutter/material.dart';

void main() => runApp(const MyApp());

class MyApp extends StatelessWidget {
 const MyApp({Key? key}) : super(key: key);
 @override
 Widget build(BuildContext context) {
 const title = 'Basic List';
 return MaterialApp(
 title: title,
 home: Scaffold(
 appBar: AppBar(
 title: const Text(title),
), // AppBar
 body: ListView(
 children: const <Widget>[
 ListTile(
 leading: Icon(Icons.photo_album),
 title: Text('Album'),
), // ListTile
 ListTile(
 leading: Icon(Icons.phone),
 title: Text('Phone'),
), // ListTile
 ListTile(
 leading: Icon(Icons.map),
 title: Text('Map'),
), // ListTile
], // <Widget>[]
), // ListView
), // Scaffold
); // MaterialApp
 }
}
```

Виберіть одну відповідь:







**Питання 24**

Відповіді ще не було

Макс. оцінка до 1,50

🚩 Відмітити питання

⚙️ Редагувати питання

Скільки разів буде виконуватись наступний цикл та що буде виведено на екран?

```
void main() {
 int counter = 0;
 while (true) {
 if (counter % 2 != 0) {
 print(counter);
 }
 counter++;
 if (counter == 10) break;
 }
}
```

Виберіть одну відповідь:

- Цикл буде виконуватись 5 разів, на консоль будуть виведені всі непарні числа від 1 до 9 (1, 3, 5, 7, 9)
- Цикл буде виконуватись 10 разів, на консоль будуть виведені всі непарні числа від 0 до 9 (0, 3, 6, 9)
- Цикл буде виконуватись 10 разів, на консоль будуть виведені всі непарні числа від 0 до 9 (0, 3, 6, 9)
- Цикл буде виконуватись 5 разів, на консоль будуть виведені всі парні числа від 0 до 8 (0, 2, 4, 6, 8)
- Цикл буде виконуватись 10 разів, на консоль будуть виведені всі непарні числа від 1 до 9 (1, 3, 5, 7, 9)

## Питання 25

Відповіді ще не було

Макс. оцінка до 1,50

Відмітити питання

Редагувати питання

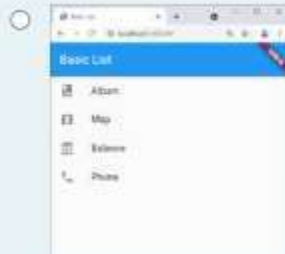
Вкажіть, яким буде результат виконання програми, що будує список віджетів у Flutter?

```
import 'package:flutter/material.dart';

void main() => runApp(const MyApp());

class MyApp extends StatelessWidget {
 const MyApp({Key? key}) : super(key: key);
 @override
 Widget build(BuildContext context) {
 const title = 'Basic List';
 return MaterialApp(
 title: title,
 home: Scaffold(
 appBar: AppBar(
 title: const Text(title),
), // AppBar
 body: ListView(
 children: const <Widget>[
 ListTile(
 leading: Icon(Icons.photo_album),
 title: Text('Album'),
), // ListTile
 ListTile(
 leading: Icon(Icons.account_balance),
 title: Text('Balance'),
), // ListTile
 ListTile(
 leading: Icon(Icons.phone),
 title: Text('Phone'),
), // ListTile
], // <Widget>[]
), // ListView
), // Scaffold
); // MaterialApp
 }
}
```

Виберіть одну відповідь:



The image displays four sequential screenshots of a mobile application's 'Notes List' screen, arranged vertically. Each screenshot shows a list of notes with a blue header and a pencil icon in the top right corner. The items in the list are: 'Map', 'Album', 'Phone', and 'Map'. The order of items varies between screenshots: the first has 'Map' at the top and bottom; the second has 'Album' at the top and 'Phone' at the bottom; the third has 'Album' at the top and 'Map' at the bottom; and the fourth has 'Map' at the top and 'Phone' at the bottom.

Завершити спробу...

## Додаток Л

*Анкета учасника курсу «Розробка кросплатформних мобільних застосунків із використанням Flutter&Dart»*

Опишіть свої враження від курсу, давши відповіді на запитання запропоновані нижче:

1. Вкажіть Ваше прізвище та ініціали: \_\_\_\_\_

2. Опишіть, будь-ласка, Ваші враження після закінчення курсу у форматі змішаної навчальної діяльності (онлайн та офлайн): **ЩО СПОДОБАЛОСЬ.**

Ваша відповідь \_\_\_\_\_

3. Опишіть, будь-ласка, Ваші враження після закінчення курсу у форматі змішаної навчальної діяльності (онлайн та офлайн): **ЩО НЕ СПОДОБАЛОСЬ.**

Ваша відповідь \_\_\_\_\_

4. Як Ви оцінюєте курс? \*

| <b>Структурні компоненти курсу</b>                          | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|-------------------------------------------------------------|----------|----------|----------|----------|----------|
| <i>Тема і програма курсу</i>                                |          |          |          |          |          |
| <i>Теоретичні відомості</i>                                 |          |          |          |          |          |
| <i>Практичні завдання</i>                                   |          |          |          |          |          |
| <i>Компетентність викладача</i>                             |          |          |          |          |          |
| <i>Різноманітність засобів курсу</i>                        |          |          |          |          |          |
| <i>Дидактичне і методичне забезпечення</i>                  |          |          |          |          |          |
| <i>Програмне забезпечення</i>                               |          |          |          |          |          |
| <i>Комп'ютерне забезпечення</i>                             |          |          |          |          |          |
| <i>Актуальність, новітність і інноваційність курсів</i>     |          |          |          |          |          |
| <i>Використання отриманих компетентностей у майбутньому</i> |          |          |          |          |          |

5. Як Ви оцінюєте свою компетентність в галузі ІКТ для свого професійного розвитку та самоосвіти, як викладача комп'ютерних дисциплін та інженера-програміста ПЕРЕД КУРСАМИ?

| <b>Компоненти ІКТ-компетентності</b>                                                                                                                                                                                                                                    | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|----------|----------|----------|----------|
| <i>Загальна – здатність щодо використання ІКТ загального призначення у навчальній діяльності (користувач ПК).</i>                                                                                                                                                       |          |          |          |          |          |
| <i>Педагогічна – здатність щодо використання ІКТ в галузі педагогічних технологій (ведення звітності, створення журналів, підготовка звітних документів, створення та редагування дидактичних та методичних матеріалів та іншого з використанням різноманітного ПЗ)</i> |          |          |          |          |          |
| <i>Предметні компетентності (використання різноманітних ІКТ для вивчення та навчання інформатики та програмування, розробки програмних продуктів).</i>                                                                                                                  |          |          |          |          |          |

6. Як Ви оцінюєте свою компетентність в галузі ІКТ для свого професійного

розвитку та самоосвіти, як викладача комп'ютерних дисциплін та інженера-програміста ПІСЛЯ ЗАВЕРШЕННЯ КУРСУ?

| <b>Компоненти ІКТ-компетентності</b>                                                                                                                                                                                                                                    | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|----------|----------|----------|----------|
| <i>Загальна – здатність щодо використання ІКТ загального призначення у навчальній діяльності (користувач ПК).</i>                                                                                                                                                       |          |          |          |          |          |
| <i>Педагогічна – здатність щодо використання ІКТ в галузі педагогічних технологій (ведення звітності, створення журналів, підготовка звітних документів, створення та редагування дидактичних та методичних матеріалів та іншого з використанням різноманітного ПЗ)</i> |          |          |          |          |          |
| <i>Предметні компетентності (використання різноманітних ІКТ для вивчення та навчання інформатики та програмування, розробки програмних продуктів)</i>                                                                                                                   |          |          |          |          |          |

7. Як Ви оцінюєте можливості платформи MoodleCloud? \*

| <b>Функції</b>                     | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> | <b>5</b> |
|------------------------------------|----------|----------|----------|----------|----------|
| <i>Дидактичні</i>                  |          |          |          |          |          |
| <i>Педагогічні</i>                 |          |          |          |          |          |
| <i>Розвиваючі</i>                  |          |          |          |          |          |
| <i>Програмні і адміністрування</i> |          |          |          |          |          |
| <i>Функціональні</i>               |          |          |          |          |          |

8. Які елементи курсу вважаєте найбільш корисними, цікавими і важливими?  
(можна вибрати кілька відповідей)

- ресурси (текстовий файл, веб-сторінка, посилання, каталог, електронна книжка, відео тощо)
- глосарій
- журнал
- тести
- форум (новини та оголошення)
- семінар
- аудіоконференція або вебінар (BigBlueButtonBN)
- Інше: \_\_\_\_\_

9. Чи мотивував Вас курс до подальшого поглиблення своїх знань і практичних умінь щодо створення, редагування елементів (відео, аудіо, анімація тощо) мультимедіа додатків і подальшого їх використання:

|                                            | <b>так</b> | <b>не можу визначитися</b> | <b>ні</b> |
|--------------------------------------------|------------|----------------------------|-----------|
| <i>у власній навчальній діяльності?</i>    |            |                            |           |
| <i>у майбутній професійній діяльності?</i> |            |                            |           |

10. Чи хотіли б Ви удосконалити свої знання щодо організації освітньої діяльності з використанням платформ дистанційної освіти, зокрема LMS MOODLE?

- Так
- не можу визначитися
- ні

*Посилання на онлайн-курс: «Розробка кросплатформних мобільних застосунків із використанням Flutter&Dart»*



<https://iktmedia.moodlecloud.com/course/view.php?id=10>