

Науменко О., Переяславська С.  
Розробка мобільного додатку QR CODE  
SCANNER за допомогою фреймворку  
FLUTTER та мови програмування  
DART. *Актуальні виклики розвитку  
інформаційної діяльності в глобальному  
вимірі* : матеріали всеукр. наук.-практ.  
конф. Полтава : Вид-во ДЗ «ЛНУ імені  
Тараса Шевченка», 2024. С.87-94

**Науменко О.В.,  
Переяславська С. О. к.пед.н., доцент,  
м. Полтава**

## **РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ QR CODE SCANNER ЗА ДОПОМОГОЮ ФРЕЙМВОРКУ FLUTTER ТА МОВИ ПРОГРАМУВАННЯ DART**

**Актуальність.** Під час розробки мобільних додатків вибір технологічної платформи має ключове значення. Один із варіантів – стек Flutter + Dart. Flutter - популярний фреймворк, розроблений Google, що надає унікальні можливості для створення кросплатформних додатків. Він застосовується, зазвичай, разом з мовою програмування Dart, яку теж розробила компанія Google [1].

Мобільні додатки на базі цих технологій оптимізовані для швидкого виконання та можуть ефективно вирішувати складні завдання, мають високу продуктивність. Також стек Dart + Flutter дозволяє створювати програми, які можуть працювати на різних платформах, включаючи iOS та Android. Тому обрання цих технологічних платформ як інструментів для розробки мобільного додатку QR CODE SCANNER є доцільним та обґрунтованим [2].

**Мета** – дослідити можливості застосування стеку технологій Dart + Flutter для розробки мобільного додатку (на прикладі QR CODE SCANNER)

QR CODE SCANNER - це мобільний додаток, який призначений для сканування QR коду.

*QR-код, к'юар-код* (скорочено від англ. *quick response code*, «код швидкої відповіді») — матричний код (двовимірний штрих-код), розроблений і представлений японською компанією «Denso-Wave» в 1994 році. Основна перевага QR-коду — це легке розпізнавання обладнанням для сканування (в тому числі й фотокамерою мобільного телефону), що дає можливість використання в торгівлі, на виробництві, в логістиці.

Хоча QR-код з'явився недавно, він охоплює широкий спектр сфер і може бути застосоване для різних цілей. Ось деякі з найбільш поширених способів використання QR-кодів, що використовуються в повсякденному житті [3]:

1. *Посилання і URL-адреси:* QR-коди можуть містити веб-посилання, що дозволяє користувачам швидко отримувати доступ до певних веб-сайтів або веб-сторінок.

2. *Візитні картки та контактні дані:* QR-коди можуть містити інформацію про контакти, таку як ім'я, номер телефону, адреса електронної пошти та веб-сайт, що робить їх зручним засобом обміну контактною інформацією.

3. *Wi-Fi мережі:* QR-коди можуть містити інформацію про підключення до Wi-Fi мережі, що дозволяє користувачам швидко підключатися до мережі без необхідності введення пароля вручну.

4. *Платіжні дані:* QR-коди можуть використовуватися для простого та швидкого здійснення платежів. Наприклад, мобільні додатки для платежів можуть створювати QR-коди для здійснення транзакцій.

5. *Товари та продукти:* QR-коди можуть використовуватися на упаковці товарів для надання додаткової інформації про продукт, його походження, способи використання та інше.

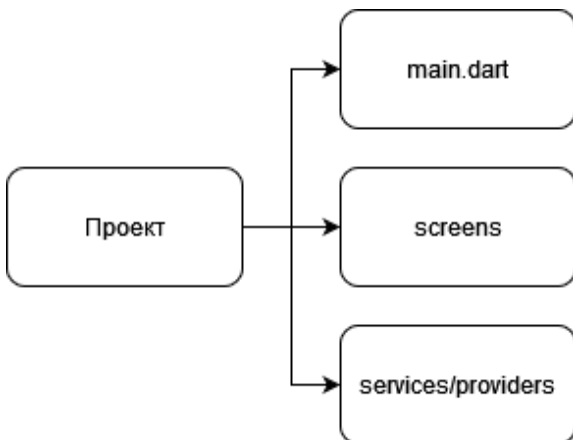
6. *Маркетинг і реклама:* QR-коди можуть використовуватися в маркетингових кампаніях для надання додаткової інформації про продукт або послугу, направлення на веб-сайт компанії або для отримання знижок та спеціальних пропозицій.

7. *Квитки та перепустки:* QR-коди можуть використовуватися для зберігання інформації про квитки на події, авіаквитки, поїздки громадським транспортом та т.п.

8. *Автентифікація і безпека:* QR-коди можуть використовуватися для двофакторної автентифікації або для доступу до захищених ресурсів, таких як будівлі, комп'ютери або особисті облікові записи.

### ***Структура проєкту***

До складу проєкту входить 3 компонента, які наведені на рис.1.



## Рис.1. Компоненти проєкту

Головний файл *main.dart* , з нього і починається виконання коду :

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

Цей код означає що, ми запускаємо проєкт за допомогою MyApp, він має клас MaterialApp, який конфігурує проєкт і відповідає за теми, мови, маршрути і т.п.

Screens - це папка, у якій розташовані екрани додатку та віджети, які описують і взаємодіють з користувачем.

Services/Providers – це компоненти, які обчислюють, реалізують логіку програми і взаємодіють з екранами.

*Екрани додатку* зображені на рис 2 – 4. З домашньої сторінки можна сканувати код за допомогою камери, або скористатися картинкою QR-коду, яку можна буде відсканувати. Додаток дозволяє скористатися електронною поштою, на яку можна буде відправити дані в розшифрованому виді. Бокова панель або drawer містить інформацію про розробника та дозволяє ввімкнути темну тему .

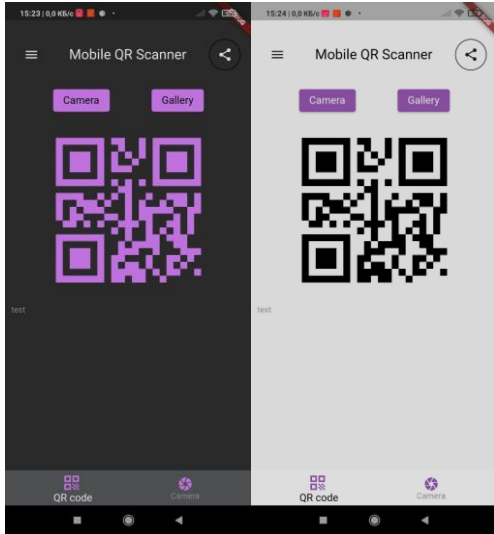


Рис. 2.Екрани додатку - сканований QR код.

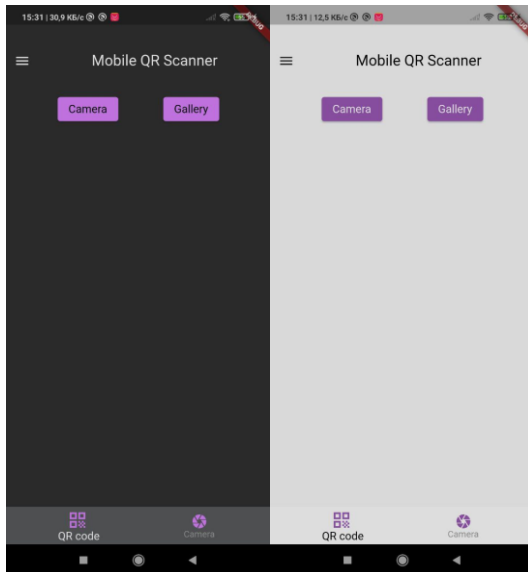


Рис.3. Екрани додатку - домашня сторінка.

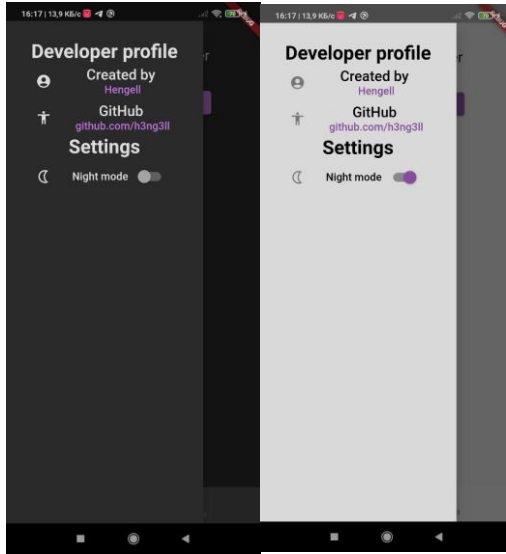


Рис. 3. Екрани додатку - Drawer – бокова панель .

### *Опис основних функцій*

Як середовище розробки обрано Android Studio з завантаженням dart SDK і flutter SDK .

Реалізація мобільного сканеру здійснюється за допомогою віджету MobileScanner:

```

MobileScanner(
  controller: MobileScannerController(returnImage: true , ),
  onDetect: (BarcodeCapture capture ) async {

    final provider = Provider.of<QRCodeProvider>(context ,
listen: false);
    final List<Barcode> barCodes = capture.barcodes;

    String value = "";
    for (final barcode in barCodes) {

```

```

        debugPrint("Successfully got barcode
        ${barcode.rawValue}");
    }
    final Uint8List? image = capture.image;
    if(image != null && barCodes.first.rawValue != null ){
        provider.updateQRCode(image,
        barCodes.first.rawValue!);
        migrateToPage(0);
    }
}
)

```

Лістинг 1. Код функції сканування QR коду

Розглянемо детальніше цей код.

- controller: `MobileScannerController(returnImage: true,):` на цьому етапі створюється екземпляр контролера `MobileScannerController` з параметром `return Image` встановленим на `true`. Цей контролер використовується для управління функціями сканування, включаючи отримання зображень та інформації про розпізнані коди;

- `onDetect: (BarcodeCapture capture) async { ... }`: це асинхронна функція, яка викликається при виявленні баркоду. Параметр `capture` містить інформацію про знайдені баркоди та зображення;

- `final provider = Provider.of<QRCodeProvider>(context, listen: false);`: отримує постачальника QR-кодів з контексту за допомогою пакету `Flutter Provider`. Це дозволяє отримати доступ до даних або стану додатка з будь-якої точки;

- `final List<Barcode> barCodes = capture.barcodes;`: отримує список розпізнаних баркодів з об'єкта `capture`;

- `for (final barcode in barCodes) { ... }`: цикл, який ітерується за кожним розпізнаним баркодом та виводить його значення;

- `final Uint8List? image = capture.image;`: отримує

зображення з об'єкта capture;

- if (image != null && barCodes.first.rawValue != null) {  
... }: перевіряє, чи є зображення, чи було розпізнано якийсь баркод;

- provider.updateQRCode(image, barCodes.first.rawValue!); : викликає метод updateQRCode постачальника QR-кодів для оновлення QR-коду з використанням зображення та значення першого розпізнаного баркоду.

Повний код можна переглянути на GitHub репозиторії [4], який є у вільному доступі, а також завантажити сам додаток [5].

**Висновок.** Під час дослідження було розглянуто процес розробки мобільного додатку QR Code Scanner за допомогою фреймворку Flutter та мови програмування Dart. Застосування обраного стеку технологій дозволило розробити крос-платформний додаток, в якому реалізована імплементація функціональності сканування QR-кодів. Було використано бібліотеку qr\_code\_scanner для доступу до камери пристрою та сканування QR-кодів. Додаток дозволяє обробляти результати сканування та відображати відповідну інформацію. Розробка додатку QR Code Scanner за допомогою Flutter та Dart продемонструвала потужність та гнучкість цих технологій. Flutter дозволяє створювати привабливі та високопродуктивні додатки для різних платформ, використовуючи єдину кодову базу. А Dart забезпечує зручність та швидкість розробки.

## Література

1. **Tashildar A.** Application Development Using Flutter *International Research Journal of Modernization in Engineering Technology and Science*. 2020. Vol. 02. URL: [https://www.irjmets.com/uploadedfiles/paper/volume2/issue\\_8\\_august\\_2020/3180/1628083124.pdf](https://www.irjmets.com/uploadedfiles/paper/volume2/issue_8_august_2020/3180/1628083124.pdf) (дата звернення: 01.04.2024).
2. **Jadaun S.** Analysis of Cross Platform Application Development Over



Multiple Devices using Flutter & Dart. *International Journal of Recent Technology and Engineering (IJRTE)*. 2023. Vol. 12. Issue-1 DOI: 10.35940/ijrte.A7580.0512123. **3. Kurale R.** A Comparative Study of Flutter with other CrossPlatform Mobile Application Development. *International Journal of Creative Research Thoughts (IJCRT)*.2021. Vol. 9, Issue 12. URL: <https://ijcrt.org/papers/IJCRT2112036.pdf> (дата звернення: 04.04.2024). **4. Репозиторій коду Qr code scanner.** URL: <https://github.com/h3ng3ll/mobile-qr-code>. **5. Додаток QR Code Scanner** URL: <https://github.com/h3ng3ll/mobile-qr-code/releases/tag/version1>