

FLUTTER – НОВИЙ ПІДХІД У РОЗРОБЦІ МОБІЛЬНИХ ДОДАТКІВ

Flutter – це мобільний додаток GoogleSDK для створення високоякісних власних інтерфейсів на iOS і Android за дуже короткий час, а також це єдиний спосіб розробки додатків під Google Fuchsia. Flutter працює з існуючим кодом, використовується розробниками та організаціями по всьому світу, і є вільним і відкритим вихідним кодом [1].

Flutter – SDK ((від англ. software Development Kit) — набір із засобів розробки, утиліт і документації, який дозволяє програмістам створювати прикладні програми за визначеною технологією або для певної платформи (програмної або програмно-апаратної)), дозволяє створювати гібридні мобільні додатки. Зокрема, це означає можливість використання однієї і тієї ж кодової бази на платформах Android і iOS. Код пишуть на Dart, мовою розробленою Google [2].

Як React Native, Flutter також має реактивний стиль. Flutter використовує інший підхід, щоб уникнути проблем з продуктивністю, викликаних необхідністю моста JavaScript. Dart компілюється «заздалегідь» (AOT) у нативний код для декількох платформ. Це дозволяє Flutter спілкуватися з платформою, не переходячи через мост JavaScript. Компіляція до нативного коду також покращує час запуску програми [3].

Flutter революційно реалізує віджети. Віджети - це елементи, які впливають і керують переглядом і інтерфейсом до програми. Віджети є однією з найважливіших частин мобільної програми на Flutter. Фактично, вони самі можуть зробити програму [4].

Flutter має нову архітектуру, яка включає віджети, які добре працюють і виглядають, є швидкими, налаштовуються і розширюються. Flutter не використовує віджети платформи (або DOM WebViews), він надає свої власні віджети.

Flutter піднімає віджети і візуалізатор з платформи в додаток, що дозволяє їм бути налаштованим і розширюваним. Все, що вимагає Flutter від платформи, – це полотно, в якому відображаються віджети, щоб вони могли з'являтися на екрані пристрою, і мати доступ до дій (дотиків, таймерів тощо) і сервісів (розташування, камера і т.д.) [5].

Існує ще інтерфейс між програмою Dart і власним нативним кодом платформи (iOS або Android), який кодує та декодує дані, але це може бути на порядок швидше, ніж мост JavaScript [6].

У Dart є сховище програмних пакетів, що дають змогу розширити можливості програм. Наприклад, існує цілий ряд пакетів, які полегшують доступ до Firebase для створення додатка без сервера. Існують також пакети під назвою "плагіни", які полегшують доступ до платформних послуг та апаратних засобів, таких як акселерометр або камера, в ОС-незалежний спосіб [7].

Завдяки такому підходу створення користувацького інтерфейсу полягає у збірці макета з безлічі різних невеликих елементів [8].

Розглянутий принцип поєднується з ще однією базовою ідеєю, на якій заснований Flutter – віддається перевага композиції перед успадкуванням. Це означає, що якщо вам потрібно створити якийсь новий віджет, то, замість того, щоб розширювати клас Widget (в Android це робиться шляхом розширення класу View), новий віджет збирають з декількох існуючих [9].

Дуже особлива можливість Flutter – гаряче перезавантаження. Гаряче перезавантаження – це механізм з високою швидкістю роботи. До того ж він дає можливість змінити код, що описує інтерфейс, в процесі виконання програми зі збереженням раніше введених даних [9].

Ще одна властивість Flutter полягає в тому, що в цій бібліотеці є вражаючий набір вбудованих компонентів для інтерфейсу користувача. Насправді, тут присутні два набори віджетів – віджети в стилі Material Design (для Android) і в стилі Cupertino (для iOS). З них можна вибирати те, що вам потрібною Найприємніше – це крос-платформність. Скажімо, якщо в проєкт доданий який-небудь віджет в стилі Material Design або Cupertino, він буде виглядати однаково на будь-якому пристрої на платформі Android або iOS. В результаті розробник може не турбуватися про зовнішній вигляд створеного ним інтерфейсу на різних пристроях [10].

Зазвичай потрібно, щоб додатки для Android виглядали не так, як додатки для iOS. Відмінності полягають не тільки в кольорах елементів інтерфейсу, але і в розмірах, і в стилі віджетів. Забезпечити правильне відображення інтерфейсів додатків на різних платформах можна за допомогою тем, якими володіє Flutter.

Отже, основні переваги Flutter: [11]

- переваги реактивних переглядів, без мосту JavaScript;
- швидкий, плавний і передбачуваний; код компілює АОТ на нативний код;
- розробник має повний контроль над віджетами та макетом;
- містить віджети, що налаштовуються;
- багато інструментів для розробників, з гарячим перезавантаженням;
- більш продуктивний, більш сумісний, більш комфортний у використанні;

– створення програми для декількох платформ з однієї кодової бази, не відмовляючись від продуктивності або потужності.

Хоча Flutter все ще перебуває в альфа-версії, ця бібліотека вже зібрала навколо себе дуже велику і активну аудиторію. Завдяки цьому Flutter підтримує безліч пакетів (бібліотек, наприклад залежність Gradle в Android). Серед доступних пакетів є такі, які допомагають працювати з зображеннями, виконувати HTTP-запити, ділитися даними, зберігати налаштування, працювати з сенсорами, користуватися можливостями Firebase. І все це можна застосовувати і при розробці для Android, і при розробці для iOS.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. What's Revolutionary about Flutter [Електронний ресурс]. – Режим доступу: <https://hackernoon.com/whats-revolutionary-about-flutter-946915b09514> (дата звернення: 11.03.2019).

2. Пять причин проникнуться симпатией к Flutter [Електронний ресурс]. – Режим доступу: <https://habr.com/ru/company/ruvds/blog/349622/> (дата звернення: 11.03.2019).

3. Flutter (SDK) [Електронний ресурс]. – Режим доступу: [https://ru.wikipedia.org/wiki/Flutter_\(SDK\)](https://ru.wikipedia.org/wiki/Flutter_(SDK)) (дата звернення: 11.03.2019).

4. Build beautiful native apps in record time [Електронний ресурс]. – Режим доступу: <https://github.com/flutter/flutter> (дата звернення: 11.03.2019).

5. Introduction to widgets [Електронний ресурс]. – Режим доступу: <https://flutter.dev/docs/development/ui/widgets-intro> (дата звернення: 11.03.2019).

6. Flutter [Електронний ресурс]. – Режим доступу: <http://developer.alexanderklimov.ru/android/theory/flutter.php> (дата звернення: 11.03.2019).