

# Flutter VS Xamarin: что выбрать для кросс-платформенной разработки

23.03.2023

Если вы решили разрабатывать мобильное приложение и прочитали сравнение Flutter VS Xamarin, то, скорее всего, слышали о кроссплатформенных фреймворках, которые позволяют разработчикам писать код один раз для приложений как для iOS, так и для Android. В статье мы покажем различия между этими двумя технологиями, чтобы вы могли понять, какая из них больше подходит для вашего приложения.

## Обзор Flutter

Flutter был создан компанией Google в 2017 году. Фреймворк работает на языке Dart, также разработанном Google. Flutter известен тем, как он обрабатывает тяжелую графику и анимацию, обеспечивая почти нативную производительность приложений. Среди популярных приложений, созданных с использованием Flutter, – несколько приложений Google, Alibaba, My BMW и многие другие.

# Apps built with Flutter

Google Ads



Alibaba



My BMW



## Обзор Xamarin

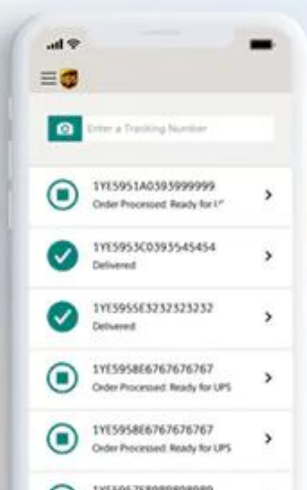
Кроссплатформенный фреймворк Xamarin был разработан одноименной американской компанией, основанной в 2011 году. Технология основана на проекте с открытым исходным кодом Mono и включает инструменты для создания приложений на Android, iOS и других платформах. В 2016 году Xamarin был приобретен компанией Microsoft и стал частью фреймворка .NET. Xamarin работает на языке C#. Среди примеров приложений Xamarin – BBC Good Food, UPS Mobile, Alaska Airlines и другие.

# Apps built with Xamarin

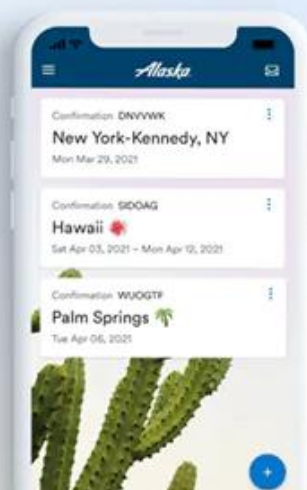
BBC Good Food



UPS Mobile



Alaska Airlines



## Flutter VS Xamarin: цена

Flutter – это набор для разработки программного обеспечения с открытым исходным кодом, который доступен совершенно бесплатно для всех разработчиков как для некоммерческого, так и для коммерческого использования.

Xamarin также является проектом с открытым исходным кодом и может свободно использоваться частными лицами и малыми предприятиями. Однако, поскольку Xamarin тесно интегрирован с Visual Studio, крупным компаниям необходимо приобрести годовую подписку Visual Studio Enterprise, стоимость которой начинается от \$799 и может достигать \$5 999 на пользователя.

**Кто победил: Flutter.**

## Архитектура и повторное использование кода

Xamarin работает на языке C#, а поскольку C# является одним из

языков .NET framework, это открывает широкие возможности для использования функций .NET, таких как синтаксис запросов LINQ, асинхронное программирование и другие. Благодаря API Xamarin.Forms и стандартным элементам интерфейса коэффициент повторного использования кода на разных платформах может достигать 96%. Кроме того, Xamarin предоставляет инструменты Xamarin.Android и Xamarin.iOS для создания специфических для каждой платформы приложений с коэффициентом повторного использования кода до 75%. Код, написанный в Xamarin, может быть повторно использован только в рамках технологического стека .NET.

Flutter использует язык Dart и большинство его компонентов являются встроенными, предоставляя все инструменты, необходимые для кросс-платформенной разработки. Приложения на Flutter создаются с помощью виджетов, которые очень хорошо настраиваются и соответствуют по внешнему виду нативному пользовательскому интерфейсу. Для рендеринга фреймворк использует мощный движок Skia C++. Возможность повторного использования кода Flutter достигает 80 процентов. Сравнивая Flutter VS Xamarin, отметим, что оба фреймворка имеют надежную архитектуру и получают большую технологическую поддержку от Microsoft (Xamarin) и Google (Flutter).

**Кто победил: однозначного победителя нет.**

## **Разработка и тестирование**

Flutter поставляется с большим количеством виджетов и API для упрощения процесса разработки, а также функцией Hot Reload, которая позволяет разработчику видеть изменения кода мгновенно, без необходимости перезагрузки всего приложения. Более того, фреймворк поддерживает тестирование на уровне интеграции (всего приложения или его части), виджета (отдельного компонента) или юнита (отдельной функции или класса), а Google предоставляет исчерпывающую документацию по тестированию Flutter.

В Xamarin есть функция, очень похожая на Hot Reload во Flutter – Live Reload, которая позволяет разработчикам видеть изменения кода вживую, без компиляции и развертывания. Фреймворк имеет свою среду Test Cloud и поддерживает модульное и UI-тестирование в Visual Studio, а также сторонние инструменты тестирования, такие как Appium, XCUITest или Espresso. Xamarin и Flutter оба имеют большие возможности тестирования и поддерживают автоматизацию тестирования, но многоуровневое тестирование выводит Flutter немного вперед.

**Кто победил: Flutter.**

## **Flutter VS Xamarin: производительность и пользовательский интерфейс**

Приложения, созданные с помощью Xamarin, демонстрируют в целом хорошую производительность, но это сильно зависит от типа используемого фреймворка Xamarin. В то время как инструменты Xamarin.Android и Xamarin.iOS обеспечивают производительность, сравнимую с нативными приложениями, благодаря использованию нативных компонентов (и более специфичного для платформы кода), производительность Xamarin.Forms (разделяет большую часть кода между платформами) ниже, особенно когда речь идет об обработке тяжелой графики. Несмотря на то, что Xamarin.Forms разделяет некоторые компоненты пользовательского интерфейса между платформами, многие компоненты приходится разрабатывать отдельно для iOS и Android, что означает, что фреймворк может быть не лучшим выбором для приложений с тяжелым пользовательским интерфейсом.

Поскольку Flutter не требует JavaScript-моста для связи с родными компонентами и использует мощный движок рендеринга Skia, фреймворк обеспечивает высочайшую производительность среди кроссплатформенных технологий, с меньшим количеством заиканий и выпадающих кадров. Фреймворк не использует нативные

элементы пользовательского интерфейса, обновляя внешний вид (иконки, шрифты) и характеристики (прокрутка, навигация) виджетов для соответствия визуальному стилю платформы с помощью Material Design и систем стиля Купертино.

**Кто победил: Flutter.**

## Кривая обучения

Для создания приложений с помощью Flutter разработчику необходимо знать Dart, который является относительно новым языком кодирования и имеет ограниченное применение за пределами фреймворка Flutter. Таким образом, большинству разработчиков, за исключением тех, кто знаком с Dart по другим проектам Google, придется изучить новый язык, прежде чем начать разработку на Flutter. Хорошей новостью является то, что, по словам многих программистов, изучение Dart довольно просто для любого, кто знаком с JavaScript, C++ и другими объектно-ориентированными языками программирования, но, тем не менее, это требует времени и определенной степени самоотдачи.

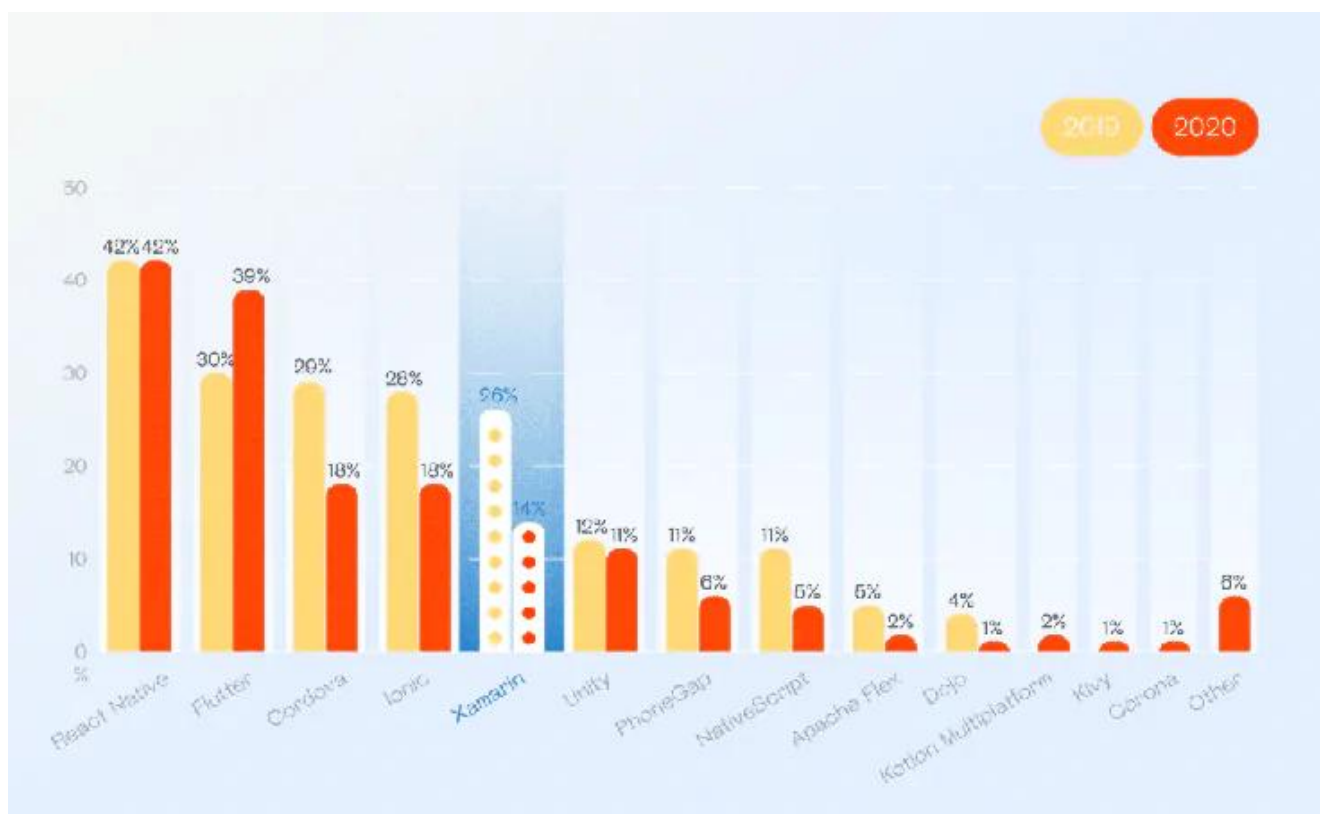
Xamarin работает на C#, который является одним из пяти самых популярных языков программирования. Он относится к языкам .NET, поэтому любому, кто работал в экосистеме Microsoft, не составит труда начать разработку на Xamarin. Опыт работы с Visual Studio Code будет еще одним большим плюсом, поскольку Xamarin интегрирован с Visual Studio IDE. В целом, широко используемый язык сокращает кривую обучения для многих программистов.

**Кто победил: Xamarin.**

## Популярность среди разработчиков и поддержка сообщества

Согласно исследованию, Flutter стал вторым по популярности кроссплатформенным фреймворком в 2020 году, в то время как

Xamarin занял пятое место. Доля разработчиков, выбирающих Flutter, выросла с 30% до 39% в период 2019-2020 годов, в то время как число сторонников Xamarin сократилось с 26% до 14%. Снижение популярности может замедлить разработку, поскольку стало меньше людей, которые могут помочь с возникающими проблемами, и меньше сторонних библиотек. В любом случае, поскольку обе технологии поддерживаются такими технологическими гигантами, как Google и Microsoft, существует множество разработчиков, знакомых с обоими фреймворками и использующих их в настоящее время. Таким образом, в нашем сравнении Flutter VS Xamarin побеждает первый, но второй также популярен.



**Кто победил: Flutter.**

	<i>Flutter</i>	<i>Xamarin</i>
<i>Price</i>	Completely free.	Free for individuals and small enterprises; large companies have to buy Visual Studio license.
<i>Code reuse across platforms</i>	Around 80%.	75%–95%, depending on tools used.
<i>Testing and development</i>	Supports Hot Reload, testing automation and multi-level tests.	Supports Live Reload and testing automation.
<i>Performance</i>	Higher performance, fewer dropped frames.	Lower performance, especially with high-demanding graphics.
<i>UI</i>	Adapts to the platform.	Adapts to the platform, but some elements have to be written separately.
<i>Easy to learn</i>	Requires more time to learn Dart language.	Based on widely known C# — easier to get started with.
<i>Community support</i>	More popular	Less popular

Xamarin VS Flutter сравнение

## Заключение

Оба фреймворка предлагают сопоставимые инструменты и возможности для разработки как простых, так и сложных кроссплатформенных приложений. Однако Xamarin не является бесплатным для крупномасштабного коммерческого использования, а поддержка сообщества в последнее время снизилась. Этот фреймворк может быть лучшим выбором для тех, кто уже работает в экосистеме Microsoft, или когда остальная часть технологического стека написана на языках .NET. Кроме того, если простые приложения, созданные с помощью Xamarin, демонстрируют почти родную производительность, то приложения с тяжелым пользовательским интерфейсом могут страдать от



снижения FPS.

С другой стороны, Flutter лучше других кроссплатформенных технологий справляется с тяжелой анимацией и графикой, а широкий набор инструментов тестирования и настраиваемых виджетов делает его подходящим для создания любых приложений, от MVP до сложных и UI-тяжелых приложений. Необходимое условие изучения языка Dart может быть недостатком для некоторых, но с учетом постоянно растущей популярности фреймворка, разработчик может быть уверен, что вновь приобретенные навыки будут востребованы.