

Разработка web-приложений для мобильных систем

Кроссплатформенн

ОСТЬ Кроссплатформенное ПО представляет собой тип программного обеспечения, которое потенциально может работать или запускаться на нескольких вычислительных платформах, например iOS, Windows, Android и т. д.

Среды разработки кросс-платформенных приложений не требуют написания кода для каждой платформы. Одноразовое кодирование создает базу или основу приложения для работы на каждой платформе.

Используя кроссплатформенные среды разработки приложений, компании-разработчики мобильных приложений могут создавать приложения, которые можно развернуть в любой операционной системе.

Фреймворки кроссплатформенного приложения в первую очередь охватывают совместимость с Android и iOS, но не только.

Кроссплатформенность позволяет разработчикам один раз написать код и запустить его на любой из платформ.

Преимущества кроссплатформенной разработки

Повторное использование

Основная причина, по которой предприятия выбирают кроссплатформенные платформы приложений, заключается в возможности повторного использования кода.

Концепция WORA (Write Once Run Anywhere) позволяет разработчикам писать код только один раз. Кодовую базу можно повторно использовать для запуска приложений на нескольких платформах. Концепция помогает устранить повторение и, следовательно, экономит эксплуатационные расходы.

Преимущества кроссплатформенной разработки

Кроссплатформенность помогает снизить затраты. Путем выбора между качеством и затратами, позволяя компаниям быть гибкими.

Меньше времени + ресурсов + усилий на разработку приложения прямо пропорционально снижению затрат.

Преимущества кроссплатформенной разработки

Простота обслуживания

Поскольку кроссплатформенность предоставляет единую кодовую базу для нескольких платформ, разработчикам приложений нужно писать и поддерживать единый код.

Это приводит к быстрому и простому развертыванию, обновлениям, обслуживанию и исправлению ошибок. Когда разработчики создают новые обновления, они легко мгновенно синхронизируются с кодовой базой, что экономит время и усилия.

Преимущества кроссплатформенной разработки

Поскольку кроссплатформенные приложения доступны на нескольких платформах, это дает максимальный охват.

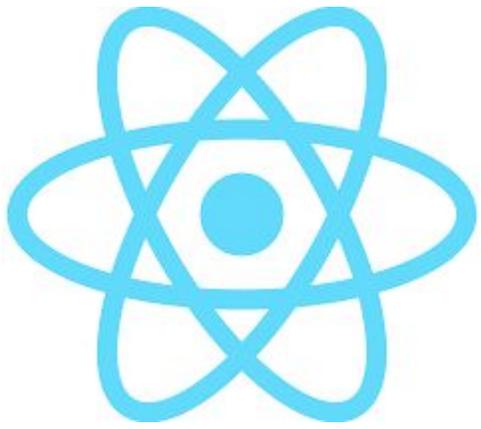
Это помогает охватить больше аудитории, позволяя разработчикам и компаниям развертывать приложение на нескольких платформах, таких как Android, Windows, macOS, iOS и т. д.

Это приводит к более широкому охвату рынка потребителей.

Преимущества кроссплатформенной разработки

Структура кроссплатформенности упрощает совместное использование унифицированного UI/UX между различными платформами.

Последовательность взаимодействия с пользователем помогает распознавать приложение на любой платформе.



Phone**Gap**



Ionic



Это UI фреймворк с открытым исходным кодом, выпущенный по лицензии MIT и предназначенный для разработки высокопроизводительных и высококачественных мобильных и настольных приложений.

При разработке ionic-приложений используются веб-технологии, такие как HTML, CSS и JavaScript с интеграцией для распространенных фреймворков, таких как Angular и React и Vue.

Ionic

Ionic Features

- 
- 1 Cross-Platform
 - 2 Web Standards-based
 - 3 Beautiful Design
 - 4 Simplicity
 - 5 Cordova Plugin
 - 6 Licence
 - 7 Ionic CLI
 - 8 Framework Compatibility
 - 9 JavaScript components
 - 10 Angular

Ionic

Кроссплатформенный

Приложение, основанное на платформе Ionic, может быть развернуто на нескольких платформах, таких как Android, iOS, десктоп и в Интернете как прогрессивное веб-приложение(PWA), и все с одной кодовой базой. Таким образом, приложения Ionic могут быть написаны один раз и работать везде.

На основе веб-стандартов

Платформа Ionic основана на надежных стандартных веб-технологиях, таких как HTML, CSS и JavaScript, с использованием современных веб-API (например, пользовательских элементов и Shadow DOM). Благодаря этому компоненты Ionic имеют стабильный API, что позволяет запускать приложение Ionic на нескольких платформах.

Красивый дизайн

Фреймворк Ionic позволяет нам создавать чистое, простое и функциональное приложение. Он предназначен для работы и красивого отображения на нескольких платформах. Приложение Ionic содержит предварительно разработанные компоненты, типографику, интерактивные парадигмы и великолепную базовую тему. зр

Ionic

Простота

Фреймворк Ionic отличается простотой, легкостью в освоении и может быть доступен любому, у кого есть навыки веб-разработки.

Плагин Cordova

Предлагает API, необходимый для использования собственных функций устройства с кодом JavaScript.

Лицензия

Ionic framework – это бесплатный проект с открытым исходным кодом, выпущенный под лицензией MIT. Это означает, что мы можем использовать его в личных или коммерческих проектах бесплатно.

Ionic CLI

Ionic CLI или интерфейс командной строки – это инструмент, который предоставляет полезные команды для разработчиков Ionic. Это команды, которые используются для запуска, сборки, запуска и эмуляции приложений Ionic.

Ionic

Совместимость с фреймворками

Ionic совместим с большинством front-end фреймворков, таких как Angular.js, React.js и Vue.js.

Компоненты JavaScript

Расширяет компоненты CSS с помощью функций JavaScript, чтобы охватить все мобильные элементы, которые невозможно реализовать только с помощью HTML и CSS.

Angular

В то время как основные компоненты работают как отдельная библиотека веб-компонентов, пакет `angular` упрощает интеграцию с экосистемой `angular`. Пакет `angular` включает в себя все функции, которые ожидают разработчики `angular`, и интегрируется с основными библиотеками `Angular`.

Ionic

Плюсы

- Ionic помогает сократить расходы, сэкономить время при найме разработчиков для платформы или обучения персонала новым технологиям.
- Его можно легко поддерживать с помощью встроенных инструментов браузера и инструментов отладки.
- Возможность превратить ваше Ionic-приложение в мобильное приложение или PWA.

Минусы

- Возможно, вам придется столкнуться с проблемами безопасности, которые существуют как в веб-приложениях, так и в нативных.
- Не подходит для разработки игровых приложений.
- Каждый раз, когда разработчики хотят применить новые изменения, им приходится перезапускать свое приложение.

React-Native



React Native – это JavaScript фреймворк, разработанный Facebook для разработки мобильных приложений с нативным рендерингом для Android и iOS.

Эти приложения реализуются с использованием комбинации разметки JavaScript и XML-Esque, называемой JSX, близкой к веб-версии React.

React Native «Bridge» вызывает собственные API-интерфейсы рендеринга Objective-C (для iOS) или Java (для Android).

React-Native

Кроссплатформенное использование

Обеспечивает подход «Learn once write everywhere» . Работает как для платформы Android, так и для устройств iOS.

Производительность

Код, написанный в React Native, компилируется в нативный код, что позволяет использовать его для обеих операционных систем, а также он работает одинаково на обеих платформах.

JavaScript

Знание JavaScript используется для создания как мобильных так и веб-приложений.

Сообщество

Большое сообщество React и React Native помогает найти ответы на многие вопросы.

Hot Reloading

Изменения вносимые в код вашего приложения будут сразу видны во время разработки. Если бизнес-логика изменена, ее отображение перезагружается на экране.

React-Native

Плюсы

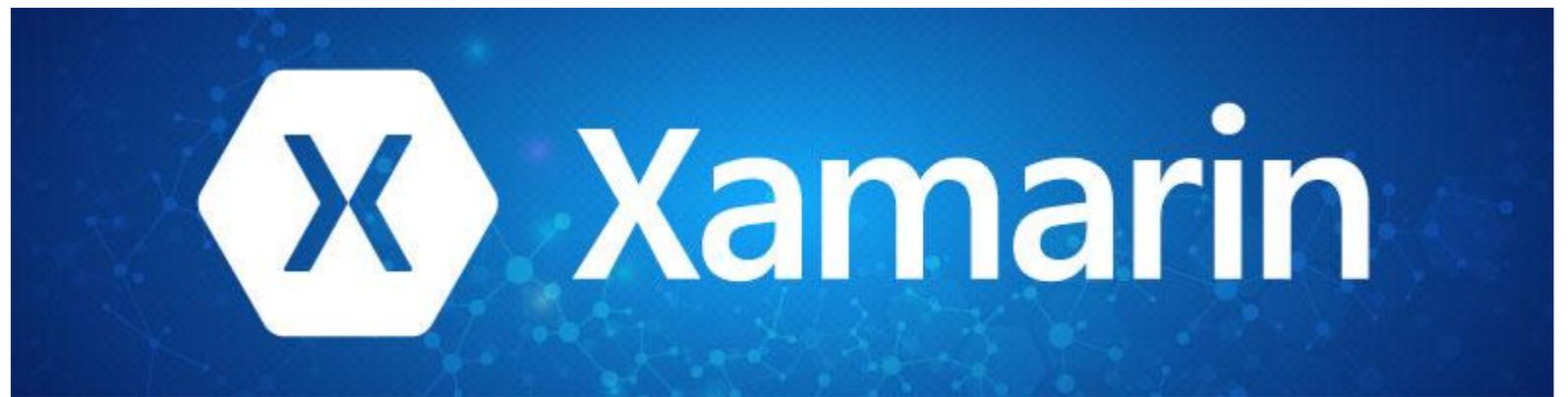
- Имеет «готовые к применению компоненты», поэтому вместо того, чтобы писать код с нуля, вы можете использовать готовые компоненты, что в конечном итоге сэкономит время.
- React Native позволяет разработчикам JavaScript создавать приложения, предназначенные для нескольких платформ, требующие лишь минимального количества нативного кода.
- React Native предоставляет функцию «горячей перезагрузки», которая позволяет немедленно вносить изменения в код в приложении для iOS и Android.

React-Native

Минусы

- JavaScript также известен своей хрупкостью. Поэтому вам нужно внимательно следить за фрагментами вредоносного кода, которые могут очень легко повредить основные функции вашего приложения.
- Приложения React-Native имеют проблемы с эффективностью и скоростью, а вычисления с плавающей запятой выполняются не самым эффективным способом с точки зрения использования и управления памятью.
- Если Apple или Google выпускает новую функцию или временное улучшение, фреймворку React-Native всегда требуется некоторое время, чтобы сторонняя зависимость добавила его поддержку.

Xamarin



Xamarin отличается тем, что предлагает единый язык C# и среду выполнения, которая работает на трех мобильных платформах (Android, iOS и Windows).

В Xamarin используется одна кодовая база C#, которая имеет доступ ко всем функциям, доступным для собственного SDK.

Инструменты Xamarin легко доступны для загрузки вместе с Visual Studio. Чтобы использовать Xamarin в Windows, нам нужно установить Visual Studio, которая может быть с бесплатной версией или платной лицензией.

Xamarin

- 1) Полная привязка для пакетов SDK: Xamarin содержит привязку для всех пакетов SDK как для Android, так и для iOS. Эти привязки строго типизированы, что означает, что они просты в навигации и просты в использовании. Xamarin обеспечивает надежную проверку типов как во время компиляции, так и во время разработки. Это приводит к уменьшению количества ошибок во время выполнения и повышению качества приложений.
- 2) Взаимодействие Objective-C, Java, C и C++: Xamarin предоставляет средства для прямого вызова библиотек Objective-C, Java, C и C++. Это дает нам возможность использовать широкий спектр стороннего кода, который уже был создан. При этом мы можем воспользоваться преимуществами существующих библиотек iOS и Android, написанных на Objective-C, Java или C/C++. Кроме того, Xamarin предлагает привязку проектов, что позволяет нам эффективно подключать собственные библиотеки Objective-C и Java с помощью декларативного синтаксиса.
- 3) Современные языковые конструкции. Приложения Xamarin написаны на C#. C# - это современный язык, который включает в себя значительные улучшения по сравнению с Objective-C и Java. Он состоит из функций динамического языка, функциональных конструкций, таких как Lambdas, LINQ, функции параллельного программирования, сложных универсальных шаблонов и многого другого.

Xamarin

Плюсы

- Xamarin имеет большую группу из более чем 60 000 участников из более чем 3700 компаний.
- Вы можете повторно использовать до 96% исходного кода, что ускоряет цикл разработки.
- Xamarin устраняет все проблемы совместимости с оборудованием, используя подключаемые модули и специальные API для работы с обычными устройствами на разных платформах.

Xamarin

Минусы

- Xamarin – это бесплатная платформа с открытым исходным кодом для отдельных разработчиков, однако она может вызвать и немалые материальные затраты, включая покупку лицензии Visual Studio.
- Согласно исследованию Stack Overflow 2018 года, всего 7,4% разработчиков использовали систему Xamarin по сравнению с 28,3% React и 8,3% Cordova.
- Как правило, приложения Xamarin больше нативных (последние могут быть в два раза меньше).

Flutter



Flutter – разработан Google и выпущен в мае 2017 года, представляет собой свободный мобильный UI фреймворк с открытым исходным кодом.

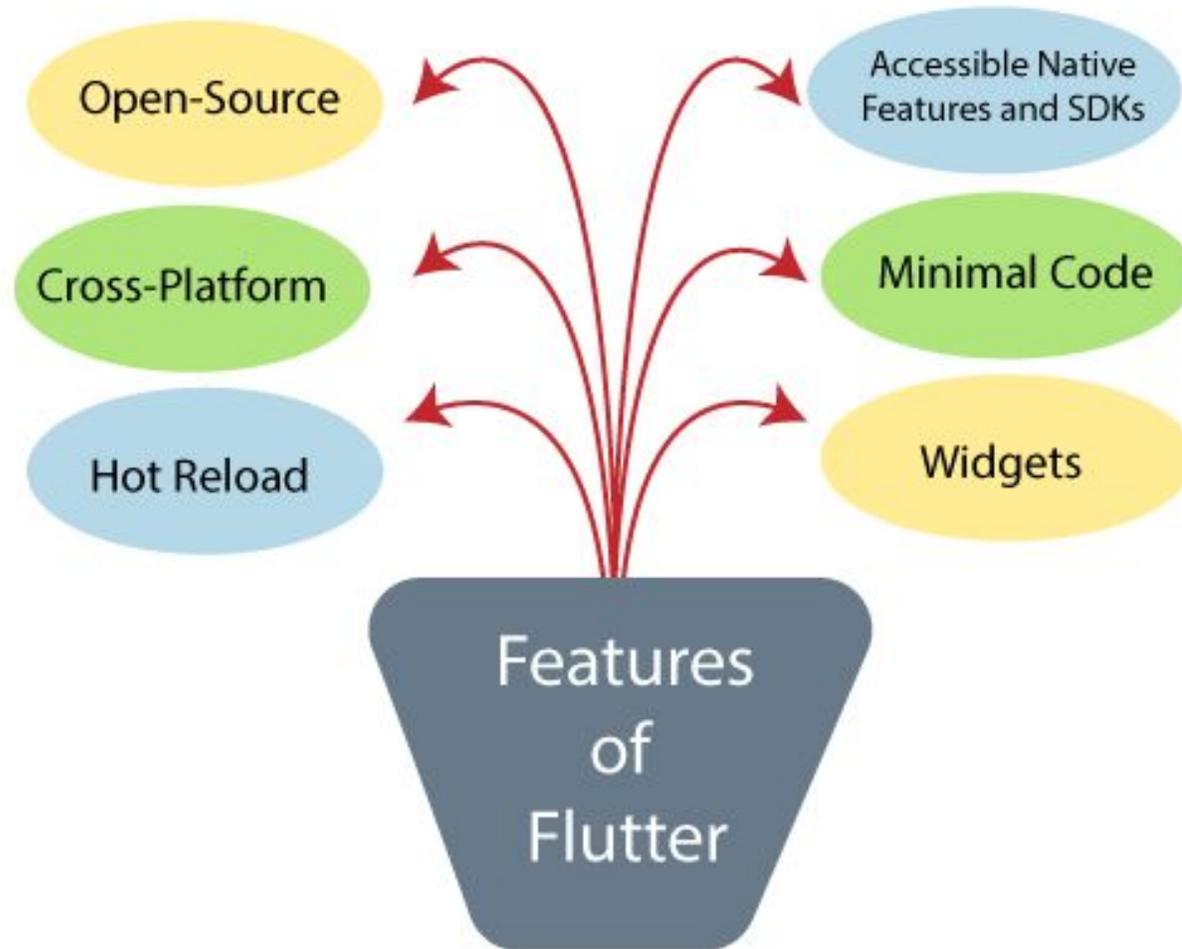
Dart – это язык программирования для разработки приложений Flutter.

Он состоит из двух основных частей:

SDK (Software Development KIT): набор инструментов, которые помогут вам разрабатывать приложения. Сюда входят инструменты для компиляции кода в нативный машинный код – код iOS и Android.

Framework (библиотека пользовательского интерфейса на основе виджетов): набор многократно используемых элементов пользовательского интерфейса (кнопки, ввод текста, ползунки и т. д.), которые можно персонализировать в соответствии с вашими потребностями.

Flutter



Flutter

Open-Source

Flutter – это бесплатный фреймворк с открытым исходным кодом для разработки мобильных приложений.

Кроссплатформенность

Эта функция позволяет Flutter написать код один раз, поддерживать его и запускать на разных платформах. Это экономит время и усилия разработчиков.

Hot Reload

Всякий раз, когда разработчик вносит изменения в код, эти изменения можно увидеть мгновенно с помощью Hot Reload. Это означает, что изменения сразу видны в самом приложении. Это очень удобная функция, которая позволяет разработчику мгновенно исправлять ошибки.

Flutter

Доступные собственные функции и SDK

Эта функция упрощает процесс разработки приложений благодаря собственному коду Flutter, сторонней интеграции и API-интерфейсам платформы. Таким образом, мы можем легко получить доступ к SDK на обеих платформах.

Минимум кода

Приложения Flutter разрабатываются на языке программирования Dart, который использует компиляцию JIT и AOT для сокращения общего времени запуска, функционирования и повышения производительности. JIT улучшает систему разработки и обновляет пользовательский интерфейс, не прилагая дополнительных усилий для создания нового.

Виджеты

Фреймворк Flutter предлагает виджеты, с помощью которых можно разрабатывать настраиваемые специфические UI.

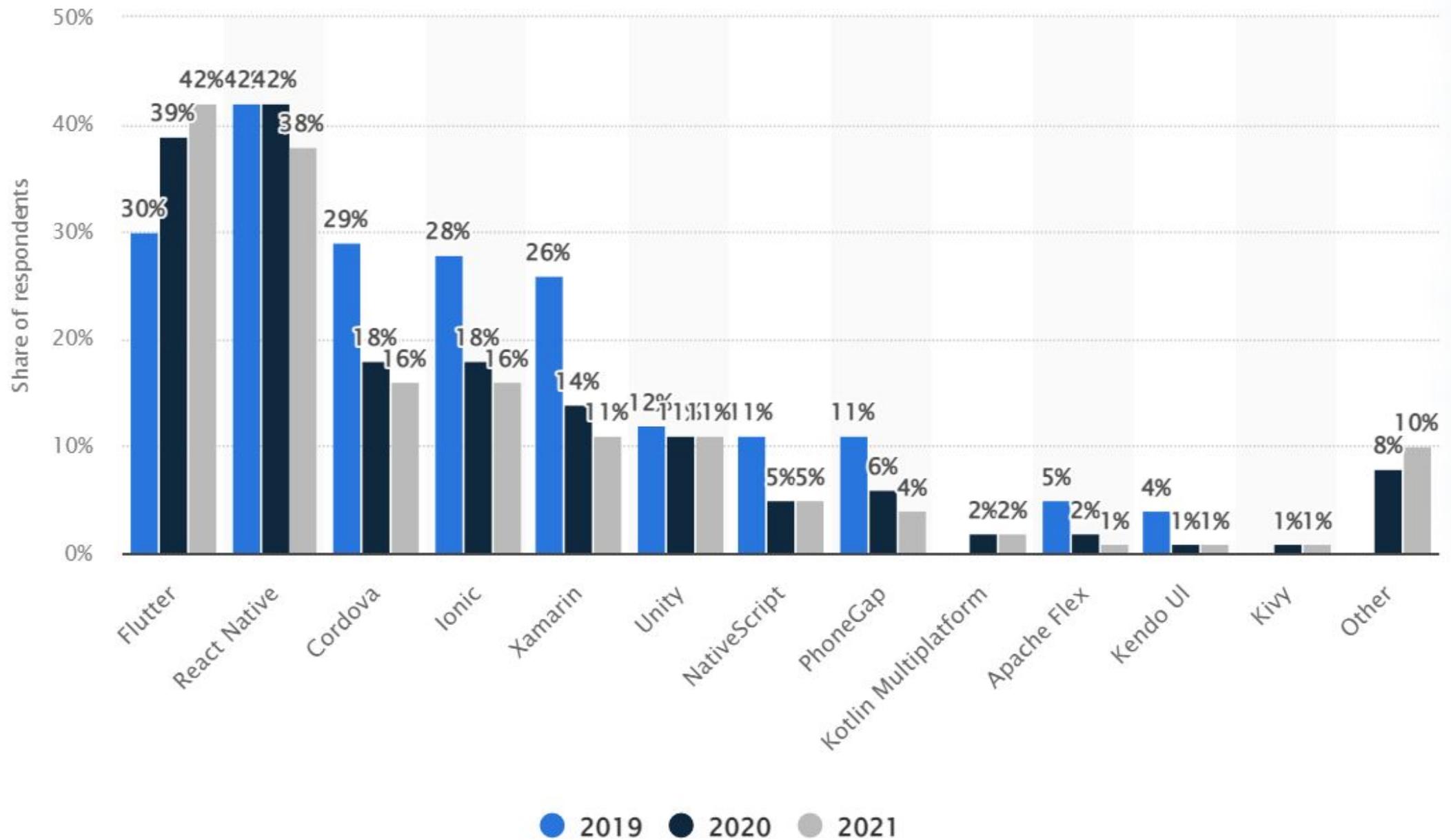
Самое главное, во Flutter есть два набора виджетов: виджеты **Material Design** и **Cupertino**, которые помогают обеспечить бесперебойную работу на всех платформах.

Flutter

Плюсы

- Это идеальная платформа для создания MVP. Вместо того, чтобы тратить лишние деньги и время на два разных приложения, можно создать мобильное приложение на Flutter, которое выглядит одинаково как для Android, так и для iOS.
- Поскольку виджеты являются частью приложения, а не платформы, проблем с совместимостью в различных версиях ОС, скорее всего, будет меньше или совсем не будет.
- Flutter работает на Android Jelly Bean или новее и iOS 8 или новее. Внешний вид вашего приложения будет одинаковым даже для более старых версий устройств iOS и Android.

Attribute	 Ionic	 React	 Xamarin	 Flutter
 Programming Language	CSS, HTML5 and Typescript + JavaScript	Java, Swift, JavaScript+ or Objective C	C# with .net environment	Dart
 Performance	Moderate ★★★★★	Close-to-native ★★★★★	iOS/Android: Close-to-native ★★★★★ Forms: Moderate ★★★★★	Amazing ★★★★★
 User Interface	CSS, HTML	Uses Native UI Controllers	Uses Native UI Controllers	Uses Proprietary Widgets for stunning UI
 Market & Community	Strong	Very Strong	Strong	New to market, so not very popular
 Platforms Supported	Android 4.4+, iOS 8+, Windows 10	Android 4.1+, iOS 8+	Android 4.0.3+, iOS 8+, Window 10	Android Jelly Bean, v16, 4.1.x or newer and iOS 8+
 Code Reusability	98% code reusable	90% code reusable	96% code reusable	50-90% code reusable
 Well-known Application	JustWatch, Pacifica and Nationwide	Instagram, Facebook, Airbnb, UberEats	Storyo, Olo, The World Bank	GoogleAds, The New York Times, eBay
 Costing	Open-source + Paid plans	Open-source	Open-source + Paid plans	Open-source



Flutter

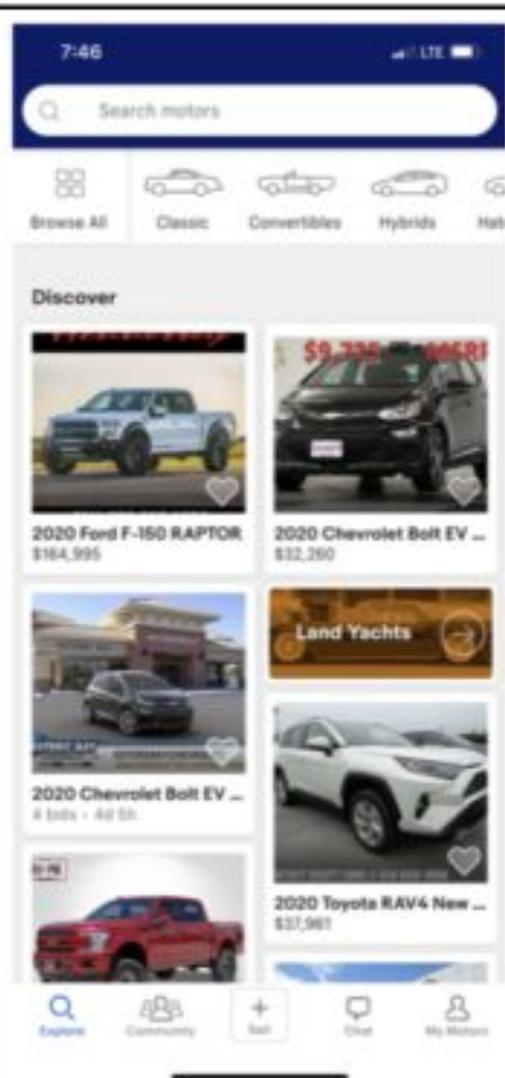
- Flutter – мультиплатформенный open-source фреймворк;
- Приложения пишутся на Dart;
- Под капотом свой графический движок (Skia) и рантайм (Dart VM);
- Поддерживаются платформы: iOS и Android, Linux, macOS, Windows, Web



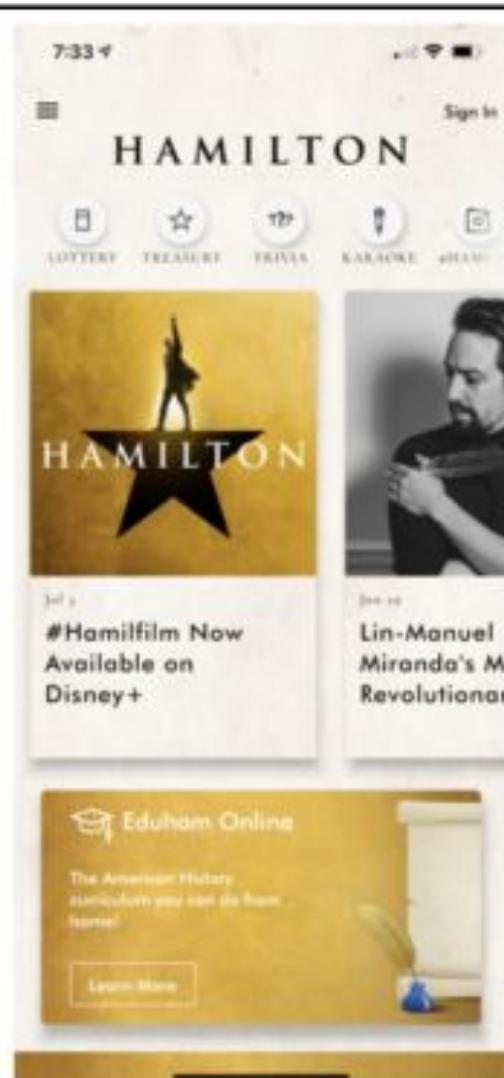
Flutter



*все картинки
кликабельны



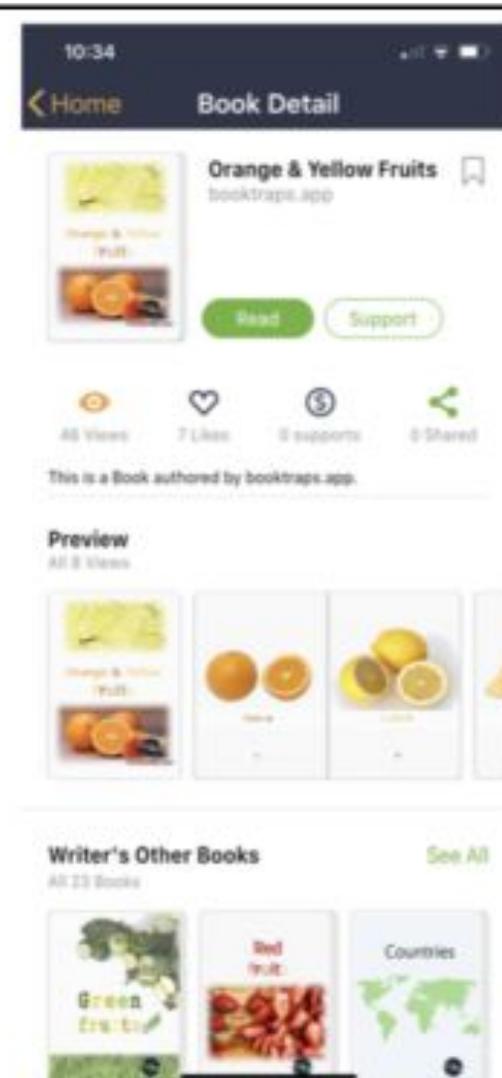
eBay Motors



Hamilton

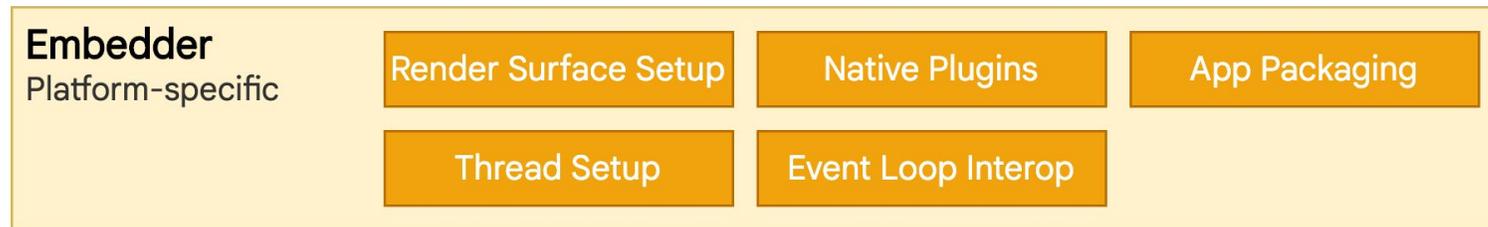
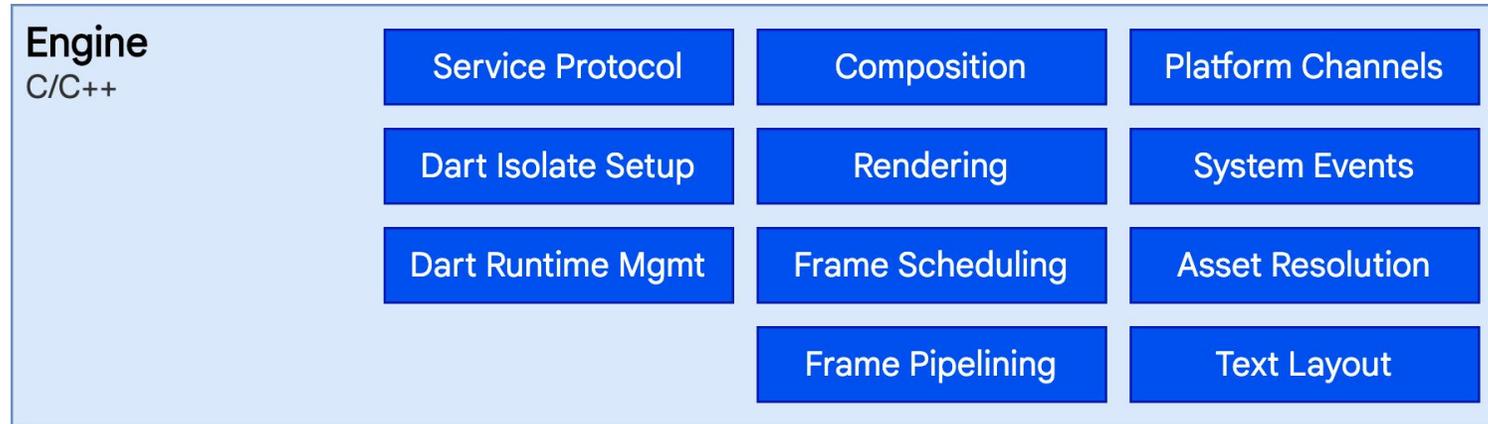
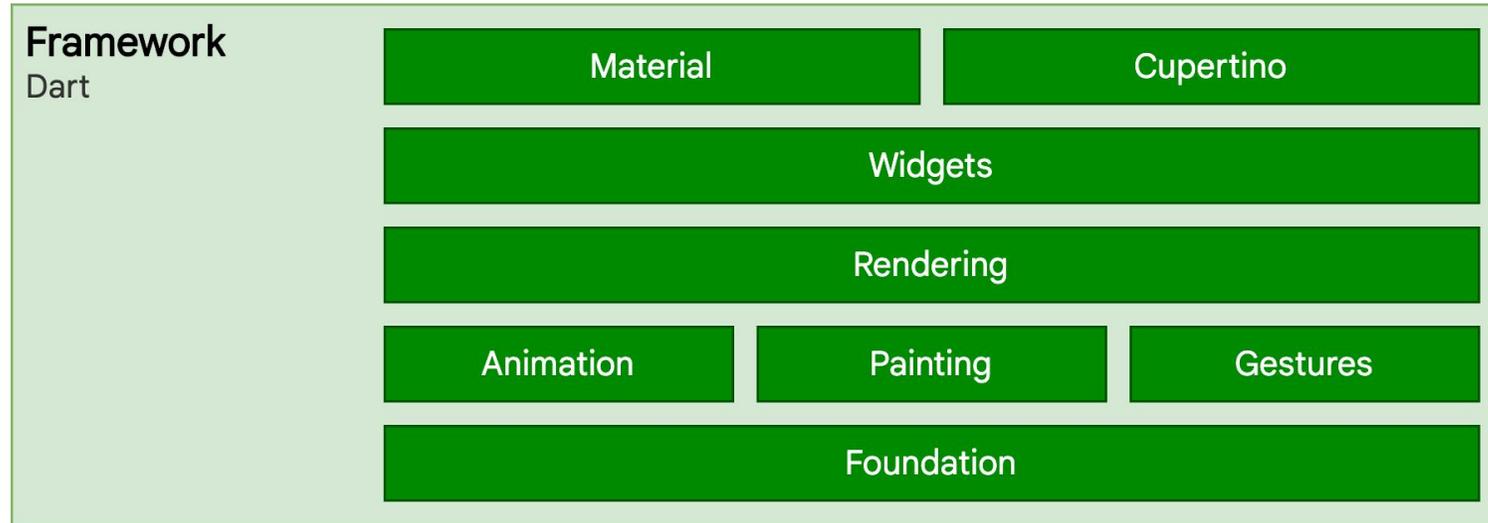


Hue BT



BookTraps

Архитектура фреймворка



Архитектура фреймворка

Framework
Dart

- UI (контролы, анимации, стилизация, user input);
- Виджеты и стейт-менеджмент.

Engine
C/C++

- Skia (графический движок);
- Dart (Runtime, GC);
- Text (LibTxt).

Embedder
Platform Specific

- Отрисовка на канвас;
- Нативные плагины.

Архитектура фреймворка

Framework
Dart

- UI (контролы, анимации, стилизация, user input);
- Виджеты и стейт-менеджмент.

<https://github.com/flutter/flutter>

Типы сборок и КОМПИЛЯЦИЯ

Debug

JIT

- Low FPS;
- «Тяжёлое» приложение (> 40 Мб);
- Работает Hot Reload.

Release

AOT

- High FPS;
- «Лёгкое» приложение (>7 Мб).

Типы сборок и КОМПИЛЯЦИЯ

- Android - Gradle;
- iOS – Xcode Build;



Fast Development

Paint your app to life in milliseconds with Stateful Hot Reload. Use a rich set of fully-customizable widgets to build native interfaces in minutes.



Expressive and Flexible UI

Quickly ship features with a focus on native end-user experiences. Layered architecture allows for full customization, which results in incredibly fast rendering and expressive and flexible designs.



Native Performance

Flutter's widgets incorporate all critical platform differences such as scrolling, navigation, icons and fonts to provide full native performance on both iOS and Android.



Fast Development

Paint your app to life in milliseconds with Stateful Hot Reload. Use a rich set of fully-customizable widgets to build native interfaces in minutes.

- По-сравнению с двумя нативными приложениями - вероятно;
- Когда приложение не заточено на глубокую работу с платформенными API.



Fast Development

Paint your app to life in milliseconds with Stateful Hot Reload. Use a rich set of fully-customizable widgets to build native interfaces in minutes.



Expressive and Flexible UI

Quickly ship features with a focus on native end-user experiences. Layered architecture allows for full customization, which results in incredibly fast rendering and expressive and flexible designs.



Native Performance

Flutter's widgets incorporate all critical platform differences such as scrolling, navigation, icons and fonts to provide full native performance on both iOS and Android.

- Нативный UI/UX – это возможно, но вручную;
- Apple-юзеры всё ещё находят отличия в UI



Expressive and Flexible UI

Quickly ship features with a focus on native end-user experiences. Layered architecture allows for full customization, which results in incredibly fast rendering and expressive and flexible designs.

- Для унифицированного UI/UX – идеально.



Fast Development

Paint your app to life in milliseconds with Stateful Hot Reload. Use a rich set of fully-customizable widgets to build native interfaces in minutes.



Expressive and Flexible UI

Quickly ship features with a focus on native end-user experiences. Layered architecture allows for full customization, which results in incredibly fast rendering and expressive and flexible designs.



Native Performance

Flutter's widgets incorporate all critical platform differences such as scrolling, navigation, icons and fonts to provide full native performance on both iOS and Android.

- Нет бутылочных горлышек на рендеринге (прощай, JavaScript Bridge);
- Простота реализации асинхронных задач – миф (Future исполняется в UI-потоке);
- Из коробки « не летает »
Используем профайлер, лучшие практики и т.д!



Native Performance

Flutter's widgets incorporate all critical platform differences such as scrolling, navigation, icons and fonts to provide full native performance on both iOS and Android.

Назад в будущее (пишем на Dart)

- Не Kotlin;
- Однопоточный (но есть Isolate);
- Нет модификаторов доступа: _приватность;
- Нет перегрузки функций;
- Mixins – интерфейсы с реализацией;
- Enum без состояния;



Dart.equals(Java) == true?

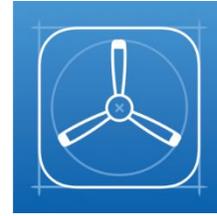
```
static String normalize(String inputString, {bool withPrefix = false}) {  
    StringBuffer buff = StringBuffer();  
    for (var i = 0; i < inputString.length; i++) {  
        String o = inputString[i];  
        if (int.tryParse(o) != null) {  
            buff.write(o);  
        }  
    }  
    String res = EMPTY_STRING;  
    if (withPrefix) {  
        res += PHONE_PREFIX;  
    }  
    res += buff.toString();  
    return res;  
}
```

Async/awa

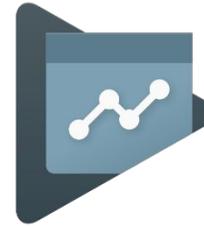
it

```
Future<Shop> _findNearestShop() async {  
  final userLocation = await _locInteractor.getLocationSafe().single;  
  final shops = await getShops(  
    lat: userLocation.latitude,  
    lon: userLocation.longitude,  
  ).single;  
  shops.sort((shop1, shop2) => _compareShops(sh1, sh2, userLocation));  
  return shops.first;  
}
```

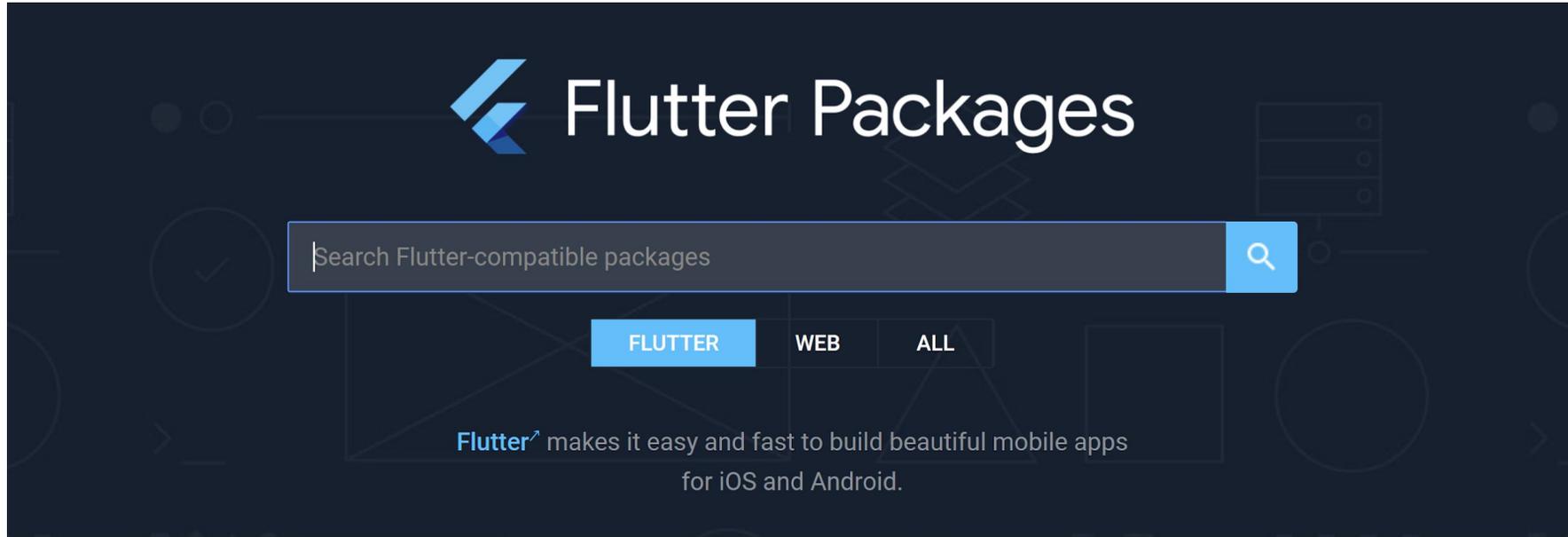
- Flutter + Dart;
- Android + Kotlin;
- iOS + Swift;
- App Store Connect + TestFlight;
- Google Play Console + Firebase App Distribution.



iOS



https://pub.dev/flutter



Top Flutter-compatible packages

[shared_preferences](#)



FLUTTER

Flutter plugin for reading and writing simple key-value pairs. Wraps NSUserDefaults on iOS and SharedPreferences on Android.

[url_launcher](#)



FLUTTER

Flutter plugin for launching a URL on Android and iOS. Supports web, phone, SMS, and email schemes.

[path_provider](#)



FLUTTER

Flutter plugin for getting commonly used locations on the Android & iOS file systems, such as the temp and app data directories.

Проверенные плагины

- https://pub.dev/packages/flutter_svg – поддержка векторной графики (позволяет использовать *.svg);
- <https://pub.dev/packages/camera> – удобное API для работы с камерой (фото + видео);
- https://pub.dev/packages/flutter_masked_text – маскирование полей ввода;
- https://pub.dev/packages/device_info – получение информации о девайсе.

- На Flutter можно написать вообще всё что угодно;

- На Flutter можно написать вообще всё что угодно;
- Для этого нужно знать очень много разных вещей;

- На Flutter можно написать вообще всё что угодно;
- Для этого нужно знать очень много разных вещей;
- Не всегда после этого становится понятно, зачем вообще было всё это писать на Flutter.