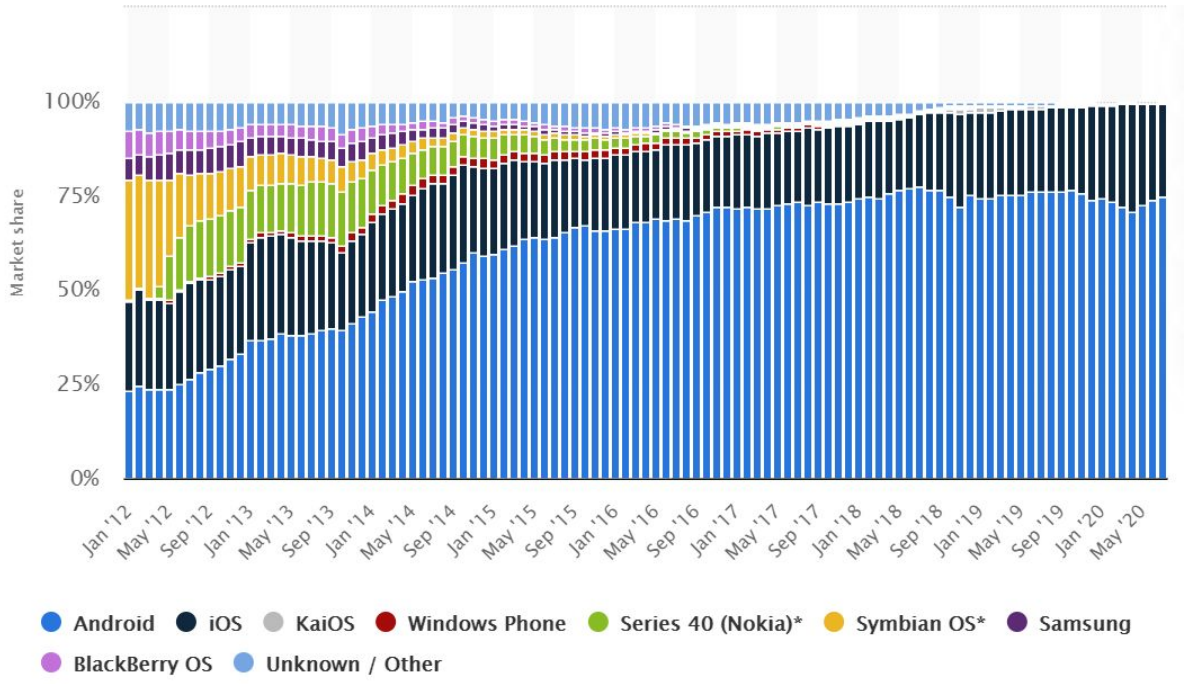
Hybrid Applications

Nowadays you need to build for many platforms to reach all your users while maintaining quality to keep them happy.



But we have now frameworks that allow us to code hybrid apps that work for many platforms at once. So what are our options?

ReactJS

What is React?

ReactJS is an open-source, front end, JavaScript library for building user interfaces or UI components on websites.

Why React, and not native JavaScript?

Ease of Use, easier composition of UI elements, ability to only update certain elements inside the page without the need of updating the whole page, which improves the speed and makes it more efficient. It achieves this using the MVC model[1], where the views or elements on the page are divided in components.

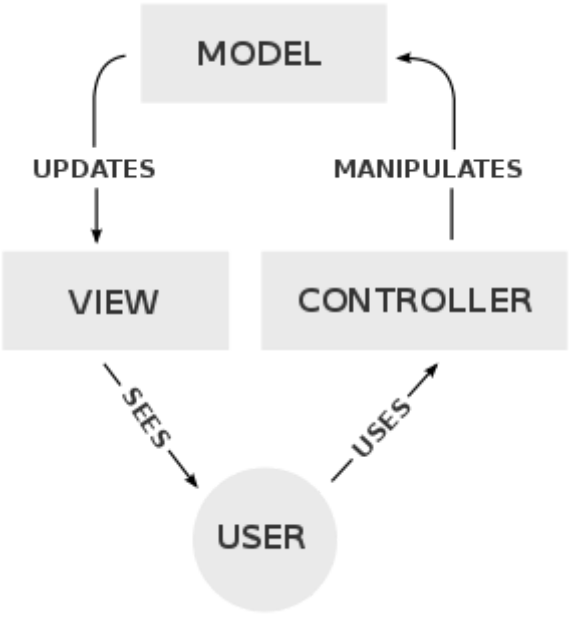


Figure 1. MVC

*Components are "**isolated**" pieces of code which are able to execute asynchronously on the page. Which is the core idea of SinglePageApplication, which is the basis of the library ReactJS.*
The components get their view updated from the model or the other way around. But never bidirectional.

How to work with Components?

React provides a very simple and efficient way to build component trees. It boasts a functional programming style where component definitions are declarative. Composing your app from React components is like composing a JavaScript program from functions.

How is the MVC model achieved?

ReactJS introduces an intermediary to achieve the Sìngle Page Application with asynchronous update of UI elements. This intermediary is known as Virtual DOM.

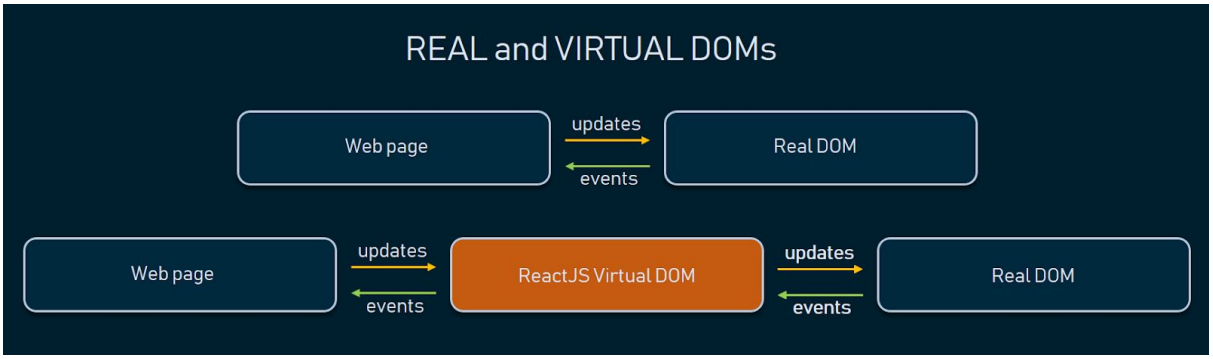


Figure 2. Virtual DOM

Why ReactJS over Angular?

Angular is a framework while ReactJS is a library, what it means is that it gives us lots of flexibility over the design of our Webpage, whereas Angular is a framework which is much more rigid and hard to modify to the needs.

ReactJS on the other hand, as it being a library we can incorporate anything we want and develop to our specific needs, as it is a View first than model and Controller Library.

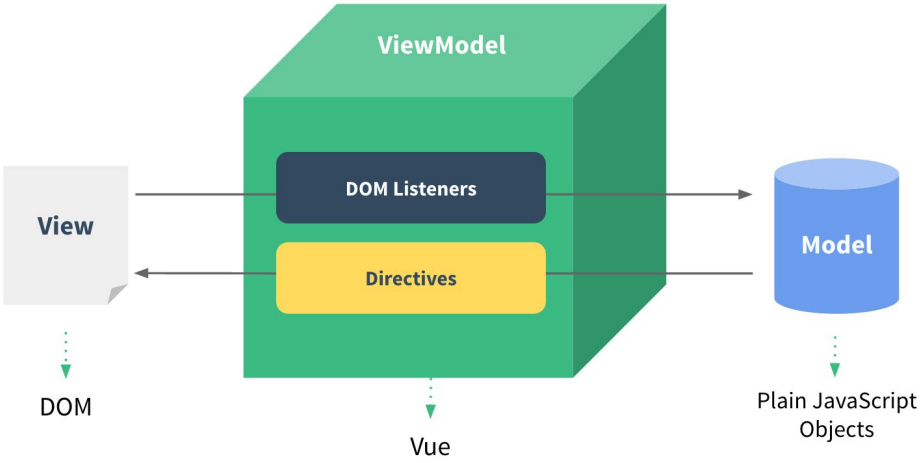
Pros & Cons of ReactJS

Pros	**Cons**
Reusability	Short Term Developement Cycle
Virtual DOM	Poor Documentation (Lots of Changes)
UniDirectionality(Data Direction from parents to child)	JSX(HTML in JavaScript)
OpenSource	No Controller (FLUX or Redux as a solution)

VUE.JS

What is VUE.JS?

Evan You, Vue.js is an open-source progressive JavaScript framework for building user interfaces (UIs) and single-page applications; it is commonly referred to as Vue. This framework uses “high decoupling”, allowing developers to progressively create user interfaces (UIs).



Why VUE.JS, and not native JavaScript?

Library modularization using a framework is common in frontend development. Both React and Angular have modularization. But what differentiates Vue.js from other alternatives is its "high decoupling", how easy it is to extend functionalities, and how well all parts work once more modules are included. For example, if we want to organize and render small visual components, all we need is Vue.js's 'core' library; it is unnecessary to include additional libraries. As the application grows, we have libraries to manage routes such as 'vue-router', libraries to manage the global state such as 'vuex' or libraries to build responsive web applications such as 'bootstrap-vue'. Additionally, if our application needs to be optimized or needs good SEO, we can include the 'vue-server-rendering' library. In the following figure, we can see how the libraries we just mentioned are progressively included, from a small SPA to multi-page applications (MPA).

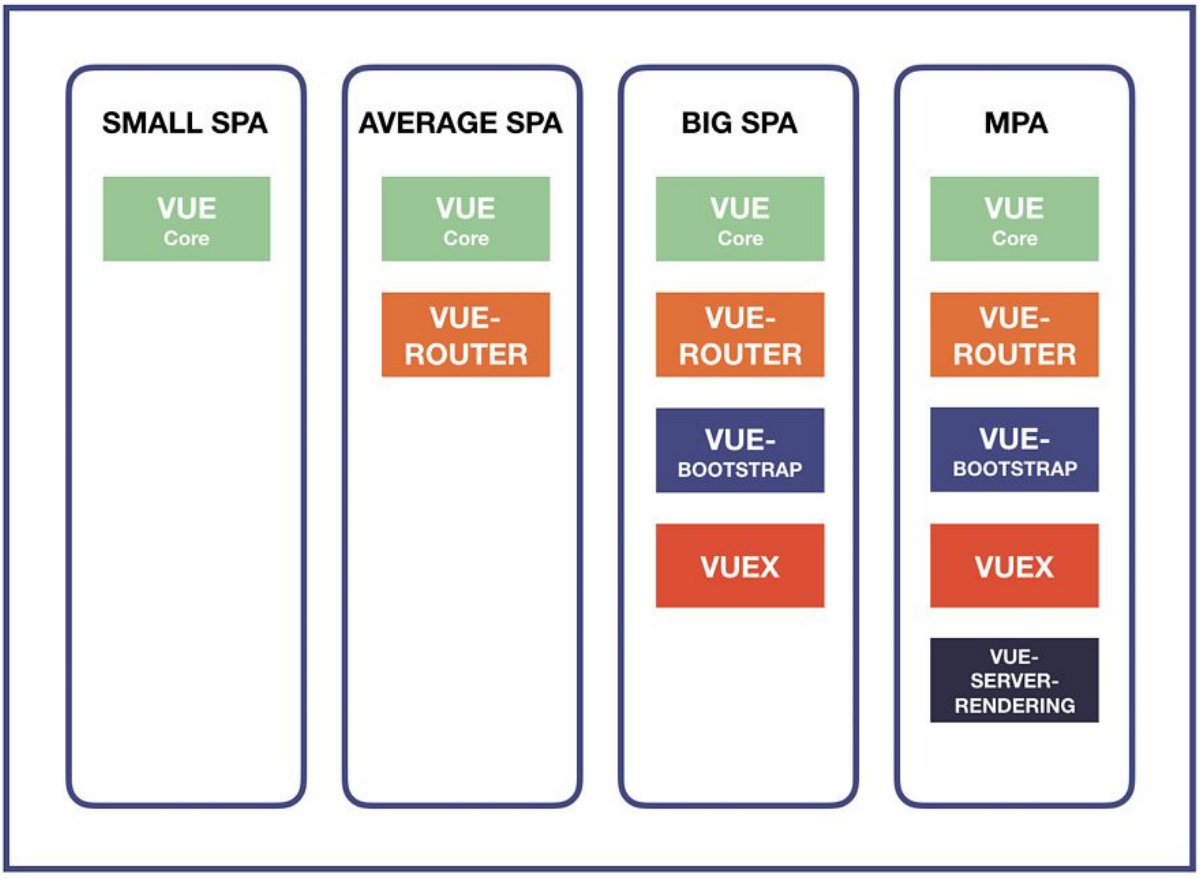


Figure 3. Vue.JS Modularization

Vue.js's component system is reactive, which means that Vue.js knows how to communicate through asynchronous events; for example, a child component can communicate with its parent component through events. With Vue.js there is no friction with other libraries or resources, in other words, we can use the tool with which we are most comfortable with. For example, we can write only HTML and JavaScript or if we want we can add CSS, JSX or TypeScript.

Why Vue.JS over Angular & ReactJS?

Angular is a framework while VueJS is a library, what it means is that it gives us lots of flexibility over the design of our Webpage, whereas Angular is a framework which is much more rigid and hard to modify to the needs.

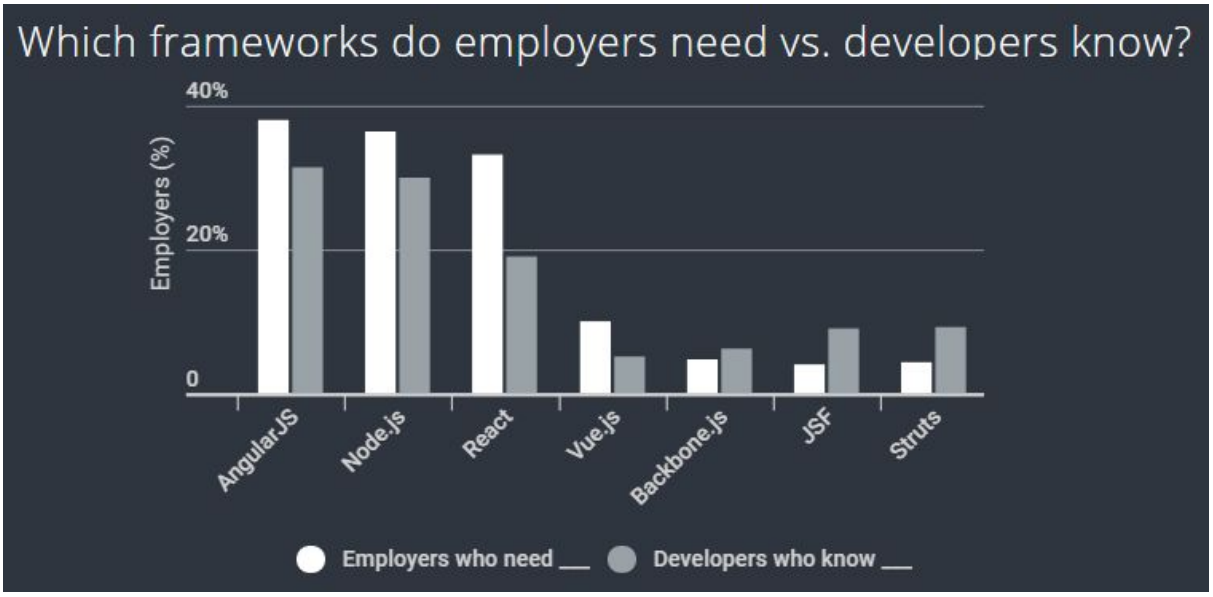
Vue.JS on the other hand is like react.js but built with simplicity and readability in mind. It achieves this simplicity using declarative models.

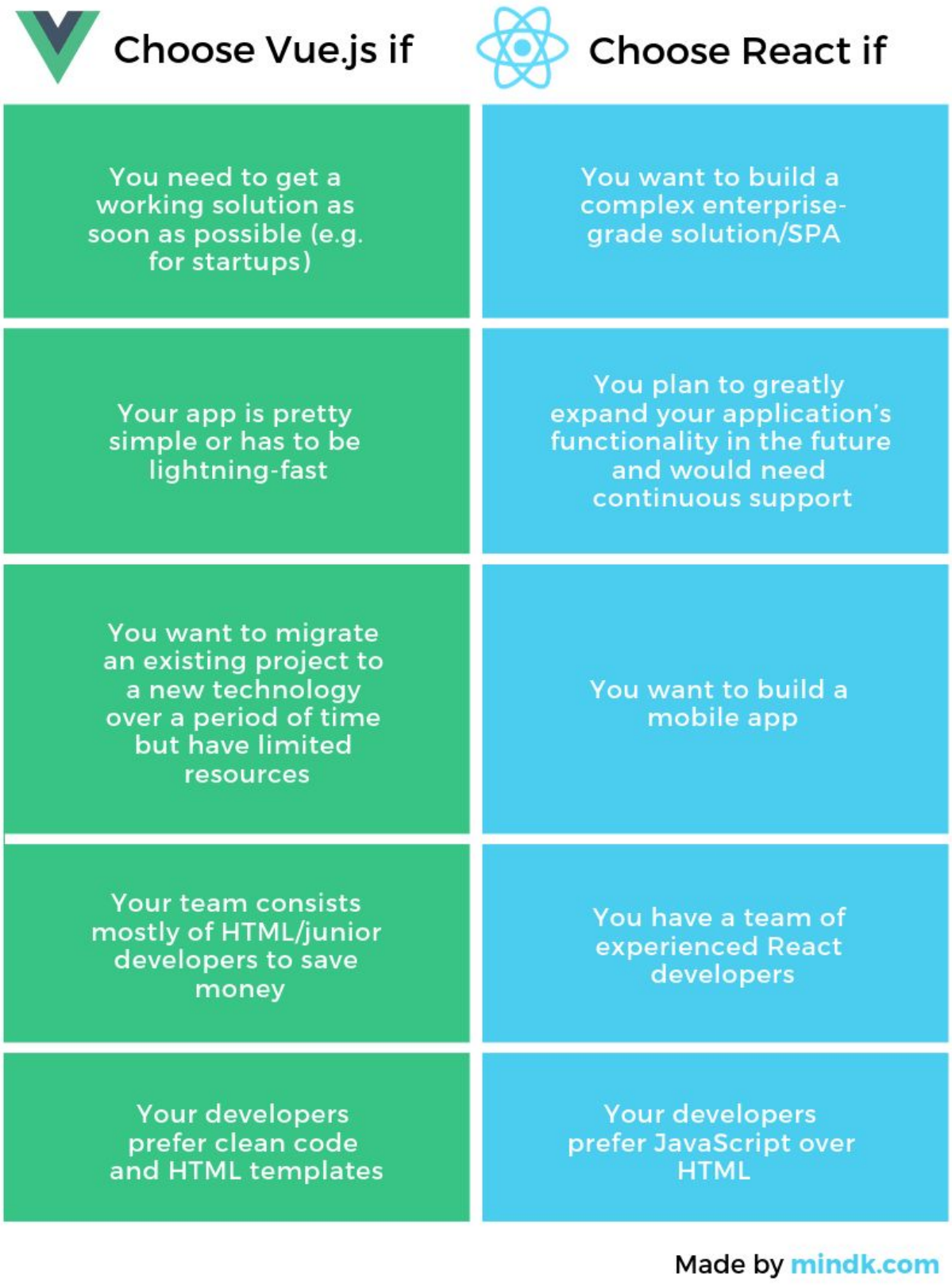
Unlike React, where we need to import external libraries for routing and data management, VueJS already incorporates them into its own library while still being comparatively lighter.

VueJS just works out of the box without any boilerplate or extra setup, just one line of code which is reserved for import and you can start programming without the need to learn JSX(React) or ES2015(Angular).

Pros & Cons of Vue.JS

Pros	Cons
Stable Progessive Library	Two-way data flow between components
Easy to Learn and Understand	Becomes Complex to scale
Simple and Flexible	Still new in market
MVVM Architecture	
Focused toward project organization	
State management is simple and intuitive	
It is a lightweight framework in comparison to React or Angular (around 20-30kb in size)	





-React Installation:
https://code.visualstudio.com/docs/nodejs/reactjs-tutorial
-Vue.JS Installation:

https://code.visualstudio.com/docs/nodejs/vuejs-tutorial

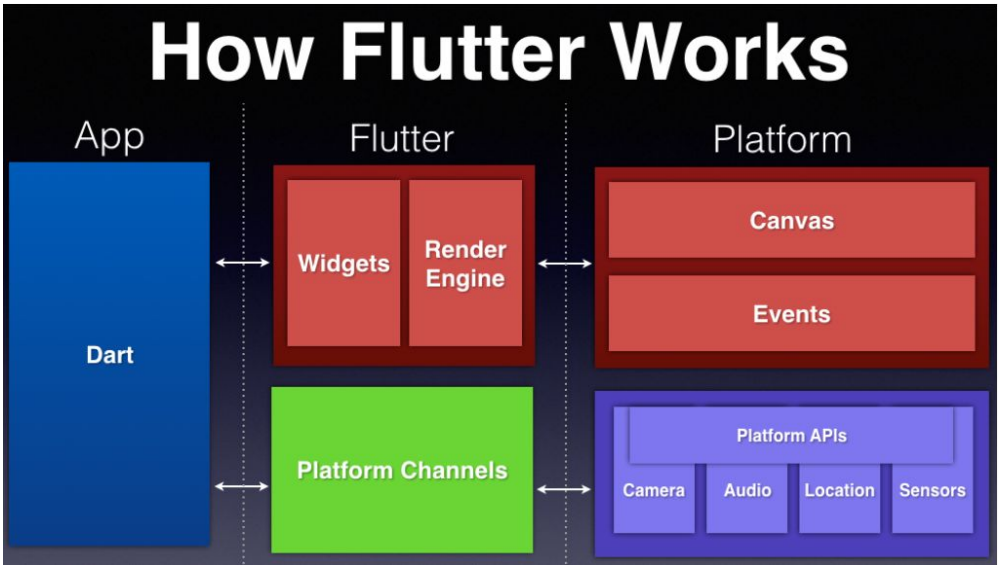
-**Script Installation Error PowerShell ESLint**
Open: PowerShell as Administrator
Execute: Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass
Finished!

Flutter

What is Flutter?
Flutter is Google's mobile UI free and open source framework that provides a fast and expressive way for developers to build native apps on both IOS and Android.

Why using Flutter for mobile apps?
Mobile users expect their apps to have beautiful designs, smooth animations and great performance, to deliver on this developers have to create new features faster than ever without compromising quality or performance. And apps and interfaces made with Flutter are built from a single code base compiled directly to native ARM code, use the GPU and can access platform APIs and services.

How does flutter work?
Flutter provides a set of widgets, managed and rendered directly by Flutter's framework and engine. Widgets are rendered onto a Skia canvas and sent to the platform. The platform displays the canvas and sends events back to the app.



Widgets describe how the view should look, given its current configuration and state. When the state changes, the widget rebuilds its description and the framework compares it with the previous description to determine the minimal changes needed to update the UI.

Types of Widgets

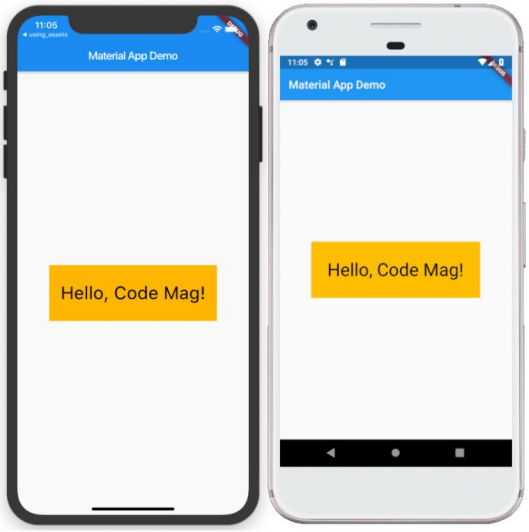
- *Stateless widgets: Changing the properties of stateless widgets has no effect on the rendering of the widget.*
- *Stateful widgets: Changing the properties of stately widgets triggers the life cycle hooks and updates its UI using the new state.*

Hot reload & Hot restart

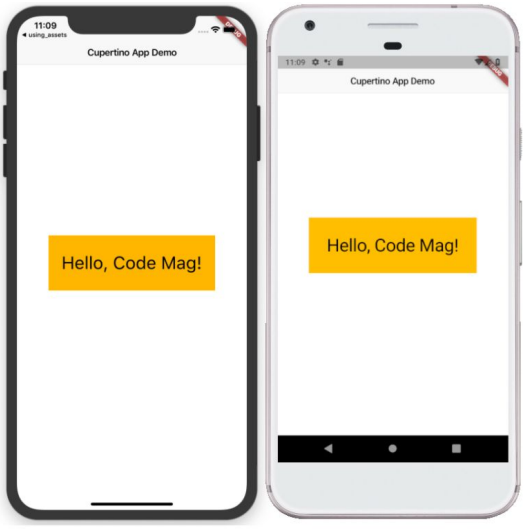
- *Hot reload (press "r" in Terminal): This option allows you to update the UI without restarting the application.*
- *Hot restart (press "R" in Terminal): This option allows you to restart the application.*

Design style classes

- *MaterialApp: The MaterialApp class represents an application that uses material design. It implements the Material design language for iOS, Android, and Web.*



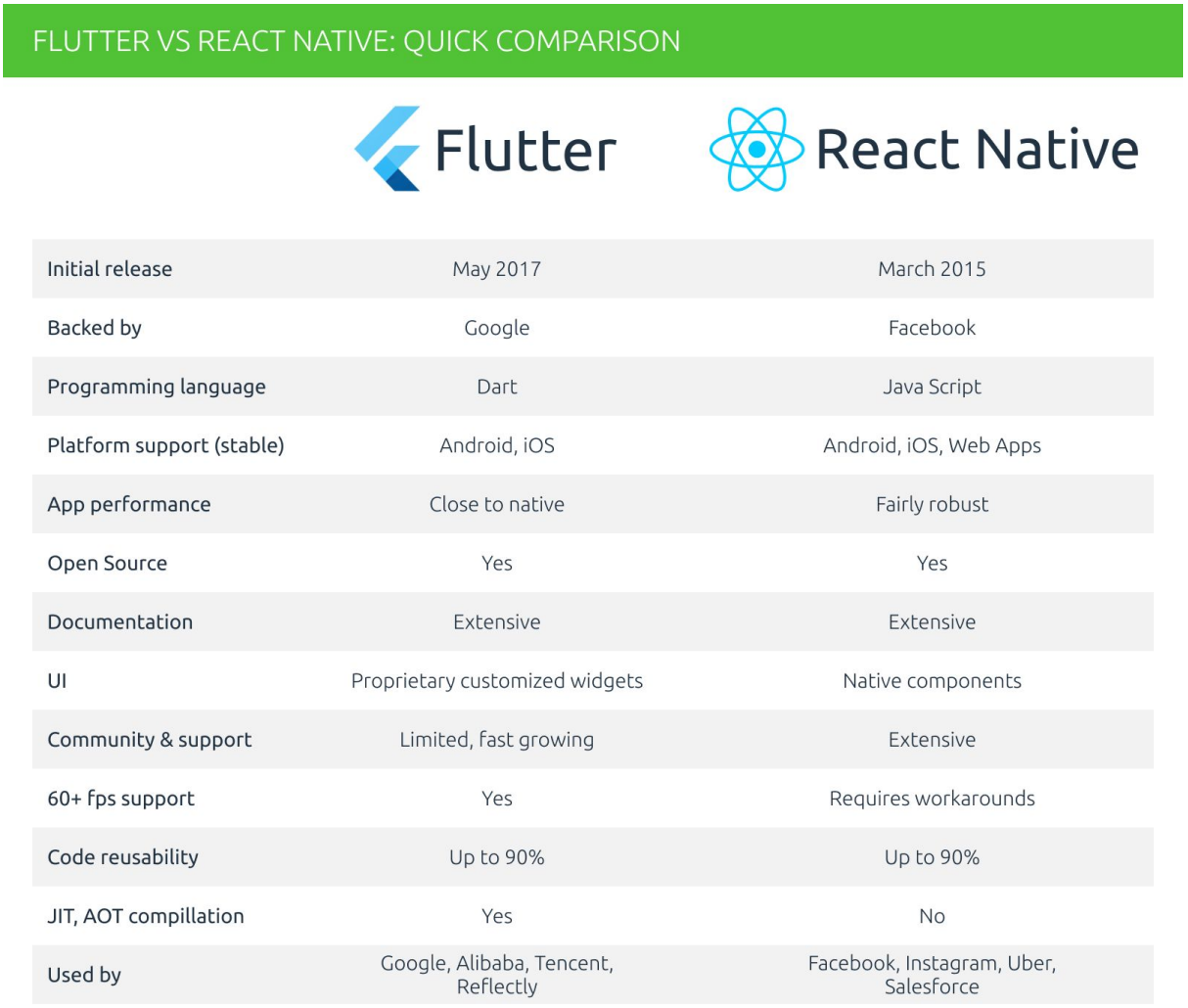
- *CupertinoApp: The CupertinoApp class represents an application that uses Cupertino design. It implements the current iOS design language based on Apple's Human Interface Guidelines.*



Pros & Cons of Flutter

Pros	Cons
Faster User interface Coding.	Limited Library support.
Hot Reload & Hot Restart.	No TV support.
Best-suited for MVP.	Large size app file.
Improved accessibility.	Bluetooth issues (FlutterBlue).
Quick to learn and easy to set up.	
Compatible with older versions.	

Flutter vs React Native



https://flutter.dev/
https://medium.com/@hrcovertx/understanding-how-flutter-works-fdd54e3ca0d1

https://www.cumulations.com/blogs/163/flutter-pros-cons
https://www.iflexion.com/blog/flutter-vs-react-native

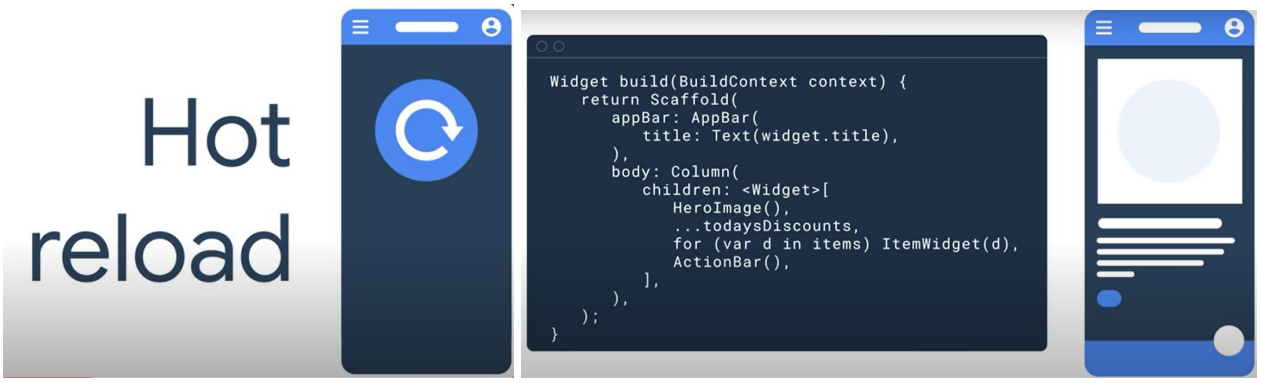
Dart

What is Dart?

Dart is a client-optimized programming language for apps on multiple platforms. It is developed by Google and is used to build mobile, desktop, server, and web applications. Flutter uses Dart to ship an application for Android iOS and the web from a single code base with a fast development experience.

Why using Dart for mobile apps?

If you're developing for mobile you might be used to slow compile and debug cycles
Dark changes this by enabling what a flutter's most loved features Hot Reload, which injects updated Dart source code into your running app and rebuilds your UI in less than a second, so you can see your changes instantly.

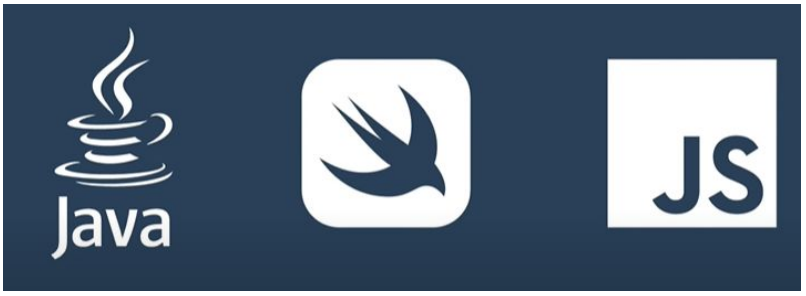


Dart is quick and optimized:

Darts production quality compilers compile to ARM and x64 machine code for mobile or optimized JavaScript for the web enabling quick app startup times and smooth animations.

Is Dart easy to learn?

Dart is easy to learn and you'll pick up Dart quickly if you're familiar with languages such as Java, Swift and JavaScript.



Pros & Cons of Dart

Pros	**Cons**
Familiar language & good documentation.	Not a great developing community.
Robust core libraries, baked into the SDK.	Not as mature as Java.
Built-in package manager.(pub)	Not as many packages as Node.js.
Compiles to JavaScript.	Fairly new language.
dart2js performs tree shaking to help reduce code size.	Dart is currently in development.
Dart runs on the server, with bundles for http server, web sockets, files, directories, and more.	

https://dart.dev/
https://codecarbon.com/pros-cons-dart-language/