

# Frontend Framework – Choice Motivation

The choice of a mobile app development framework plays an important role in ensuring the success and efficiency of app development projects. While there are a lot of mobile app development frameworks available, two of them take the crown. Flutter and React Native stand out as leaders in the industry. It is crucial to investigate and understand the strengths and weaknesses of these frameworks to make informed decisions when embarking on a project to create a mobile app that enables users to see their position on a custom floor plan and find others using the same app.

## 1. Frontend Frameworks - Summary

### 1.2 Flutter

Flutter is an open-source UI software development kit created by **Google**. It allows developers to build **natively compiled applications for mobile, web, and desktop from a single codebase**. It uses the **Dart programming language**, which is known for its efficiency and performance. However, the learning curve can be rather steep and perhaps not the best choice for a project with a strict deadline, when the people developing it have no previous experience with the Dart language.

Flutter offers a **rich set of pre-designed widgets and a hot-reload feature**, enabling real-time updates and faster development cycles. Its architecture encourages a highly customized user interface design. It has gained popularity for its **seamless cross-platform compatibility** and the ability to create visually appealing apps with smooth animations.

### 1.3 React Native

React Native is a popular open-source framework developed by Facebook that **uses JavaScript and React for building mobile applications**. It allows developers to write code in JavaScript and reuse it across different platforms, **including iOS and Android**.

React Native boasts a large and active community, providing access to a wide range of libraries and third-party plugins. The framework is known for its "Learn Once, Write Anywhere" philosophy, **making it easier for web developers to transition to mobile development**. It's valued for its strong performance and the **ability to integrate with native modules for platform-specific features**.

## 2. Concerning the actual project

When choosing between Flutter and React Native for creating an app that displays the user's position on a custom floor plan and allows them to find others using the same app, several factors are to be taken into consideration:

### 2.1 Performance & user experience

Both Flutter and React Native offer excellent performance. However, Flutter, with its Dart-based architecture, has a slight edge in creating a highly customized and visually engaging user interface. It's particularly **advantageous when designing custom floor plans and intricate user interactions.**

### 2.2 Development speed

Both frameworks' "hot-reload" feature facilitates rapid development cycles, which is beneficial for quick iterations during app development. This can be especially useful for maintaining a dynamic user experience when tracking and finding others on a floor plan.

### 2.3 Developer skillset

The choice between Flutter and React Native may also depend on the expertise of your development team. If your team is already experienced in JavaScript, React Native may be a more familiar choice. Conversely, if you're already experienced in the Dart language, Flutter might be the better choice

In conclusion, both Flutter and React Native are strong contenders for creating the app. While React Native offers faster development cycles and a larger community, Flutter provides superior control over custom UI and animations.

## 3. Final decision

Although both frameworks seem to be an excellent choice for this project, I advise we use Flutter for this project. Since we're going to work with custom floor plans used for navigation, customization will have crucial factor. Most likely, we're going to use some sort of Google Maps API for making the navigation work. Since this framework is designed by Google, there is a good chance that integrating this with Flutter will be less troublesome than working with React Native. We both have developed applications in Flutter before, so we're already familiar with the whole architecture.