# Usability Test: Google Flutter API Documentation

**Team Wired Frames**
Avinash Ratnam, Anandita Trivedi, Dev Sharma, Mengyun Liu, Rongqi Bei

Date: April 19, 2023

Word count: 1899

URL to Google Doc version of this document: Usability Test Report: Flutter

URL to supporting Google Slides: Usability Test Presentation Slides - Flutter

## Executive Summary

This report presents the findings of a usability test conducted on the Flutter API Documentation website. The research objective was to identify usability problems that prevent novice Flutter users from searching and consuming the site's information. The report includes the methodology used for participant selection and recruitment, the test protocol, and the results of the usability test.

The report presents two findings and recommendations around site navigation and information search, specifically the sidebar and search terms and search bar with search results, and one finding and recommendation around information consumption, specifically content and examples. The report suggests consolidating related information on the same page instead of jumping to another page and improving the spacing and indentation of the right menu bar to make the information hierarchy better.

Additionally, the report recommends larger font sizes to illustrate titles and key content. The report also found that most users were comfortable searching within a page using the ctrl+F or cmd+F feature to search for properties and methods within a class page.

The report concludes by suggesting that the Flutter API Documentation website could benefit from further improvements to its user experience and overall usability for novice users.

# Introduction

This UX report presents the findings of a usability test conducted on the Flutter API Documentation website, with the research objective of identifying usability problems that prevent novice Flutter users from searching and consuming the site's information. The study aimed to evaluate how the site facilitates users to navigate themselves to the desired information and whether the site presents information in a user-friendly way for comprehension and supporting users' own programming practices.

**Research objective:** Identify usability problems that prevent novice Flutter users from **searching** and **consuming** the site's information. For searching, we measure how the site facilitates users to navigate themselves to the desired information. For consumption, we evaluate whether the site presents information in a user-friendly way for comprehension and for supporting users' own programming practices.

The Flutter API Documentation website is an essential resource for Flutter developers, providing information about APIs, classes, methods, and properties that developers can use to build their applications. However, the site's complexity and a vast amount of information can make it challenging for novice users to search for and consume the information they need.

To address this issue, we conducted a usability test with a group of participants, which involved completing tasks related to searching and consuming information on the Flutter API Documentation website. The usability test aimed to identify usability problems and gather feedback from users to improve the site's user experience for novice Flutter developers.

**The rationale for choosing novice users:** A professional user we interviewed shared that he seldom visits the API documentation site. He either directly searches within his code editor or checks the source code. Oppositely, some novice and intermediate users check the reference site sometimes to figure out how to implement certain effects.

In this report, we present our research objectives, methodology, and findings related to the usability problems that prevent novice Flutter users from searching and consuming the site's information. We also provide recommendations for improving the site's user experience and overall usability for novice users.

# Methods

### Participants Selection and Recruitment

In order to better study how novice and experienced Flutter users operate and use Flutter to navigate and consume information, we recruited five participants with a variety of programming and Flutter usage backgrounds. All our participants were selected and required from the potential interviewee's database collected previously.

## Lessons Learned from An Pilot Test

Our team members initially designed the test tasks based on different main functions and features. A pilot test was performed with a group member's friend who has Flutter usage experience. During the online Zoom section, the participant was asked to complete the tasks and think aloud by following the task sheet. We discovered that tasks lacked coherence and were isolated during the pilot test, so we decided to set up a scenario and link each task together based on the workflow logic.

| | P1 | P2 | P3 | P4 | P5 |
|---|---|---|---|---|---|
| **Programming background general** | Experienced with Python, Java, C, and C++; Mainly code for machine learning stuff. | 6 years of programming experience, experienced with Python, Java, C, and C++; Data Processing. Have used CSS, familiar with JavaScript, HTML, and Node.js. | Experienced with front-end programming, including HTML, css. javascript, react.js, and angular. 4 years of experience with Python programming for machine learning; Experienced with | Mechanical engineering bachelor<br>ECE master<br>Python for robotics<br>2 years of programming | Experienced in Python programming and C programming; has a mechanical engineering background and is more familiar with Matlab and other such software. |
| **Background with Flutter** | Have learned Flutter on his own for building a self-driven project. | New user | Had learned Flutter on her own. | New user | Had a basic understanding of Flutter; Tried learning once before but dropped out due to inadequate time. |
| **Gender** | Male | Male | Female | Male | Male |
| **Notes** | He is also the P1 for our interview assignment) | N/A | N/A | N/A | N/A |

***Table 1.*** *The de-identified demographic information of participants*

## Usability Testing: Set-up, Task, and Procedure

We conducted one-hour usability tests with these five participants over Zoom using laptops, which includes

- **a warm-up interview** inquiring about participants' programming background (Appendix A)
- **six usability test tasks**: five usability test tasks and a backup task
- **debriefing sessions** asking follow-up questions regarding participants' performance on each task
- **a post-test oral survey** exploring their attitudes toward this site (Appendix B).

The six usability tasks were cohesively designed around a scenario where the participants were asked to polish a mood card by adding a shadow below it and making the corner rounded (figure 1).

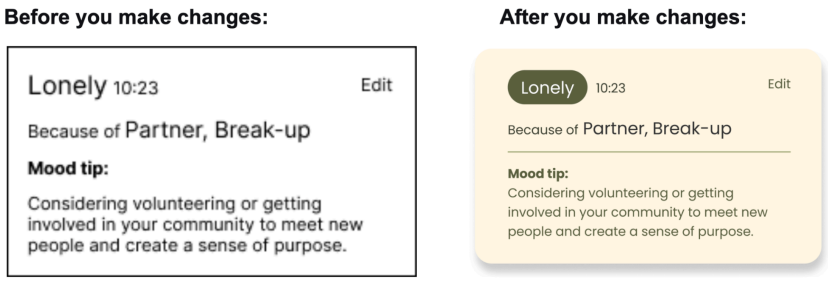


***Figure 1.*** *The scenario of the usability test*

Following the research goal, the tasks aim to evaluate the usability and effectiveness of the API documentation site in supporting information search, information consumption, and navigation, as well as some small functions (Table 2).

| Task No. | Task Focus | Summary of Task |
|---|---|---|
| 1 | Small functions | Change the bright/dark mode of the API documentation site. |
| 2 | Information Search & Navigation | Search to find the documentation for the “Card” Class. |
| 3 | Information Search & Navigation | Locate the method for adding shadow. |
| 4 | Information Consumption | Acquire the method and experiment thoughts within the website (i.e., the interactive codel boxes). |
| 5.1 | Information Search | Locate the method for border-radius |
| 5.2 | Information consumption | Acquire the method and experiment thoughts within the website (i.e., the interactive code boxes). |
| 6 (backup) | Small functions | Copy the code with an icon next to the code snippets. |

***Table 2.*** *A summary of the task*

We had two researchers (1 moderator and 1 note-taker) for each usability test. Before the test, participants gave informed consent to let researchers cloud record the test sessions; and they were told that their participations were voluntary and they can pause or exit the test at any time. All participants will be compensated with a Flutter T-shirt. During the usability test, participants were asked to share their screens and follow the think-aloud practices.

## Findings and Recommendations

We obtained two findings and recommendations around site navigation and information search,

1. Sidebar
2. Search Term and Search Bar + Search Results

and one finding and recommendation around information consumption

3. Content and examples

## 1. Sidebar

**Finding 1.1**
During the usability testing, three of five participants were disturbed and confused about redirecting to other pages when they clicked on the items in the right sidebar menu. Jumping to other pages went against what they expected for the operation of clicking the item in the right sidebar menu, and required them to read and comprehend the new page again in order to understand its material, which had a significant impact on consistency and continuity in navigating and searching the information.

**Finding 1.2**
In the usability testing, we found that only one of the five participants used the sidebar menu to locate the information they needed when completing the task. In a later follow-up conversation, both experienced Flutter participants stated that the sidebar was a huge list that contained too many items without a clear hierarchy, which made them feel very overwhelmed and did not want to spend time figuring out how to use it. The other two novice Flutter participants said they could not understand the classification logic of the contents in the sidebar menu after they glanced over the content of the sidebar.

**Recommendations:**
Flutter can consider consolidating and displaying related information on the same page by having the right sidebar menu to link and locate to a certain place at the current page instead of jumping to another page for more information. Also, the spacing and indentation of the right menu bar could be improved to make the information hierarchy better, and changing to larger font to illustrate the titles and key content.

## 2. Search Term and Search Bar + Search Results

**Findings**
During our tasks, most of the users we interviewed were comfortable searching within a page using ctrl+F or cmd+F. They used this feature to search for properties and methods within a class page. Most users treated the search bar as they would with any other search interface, i.e., they would search for generic terms and not specific properties. Unfortunately,  these searches would end up with no results being returned. However, most users used the search box rather than the search results page. They felt the immediate results being shown were enough to make a decision. User P5 happened to press enter while searching for a term and was surprised at the number of search results for a single word. They were confused by the results being only classes. They questioned if there was any hierarchy in the search results and if different classes were going to give them different results for the same method (for example).

**Recommendations**
A more comprehensive search box that allows displaying the search term primarily, with the parent class (if any) taking a secondary role. This should be especially useful on the search results page, where none of the results point toward the actual search result

(unless it is a class). Allow searches to be more generic and not rely on keywords. So far, the search is built on exact text matching the search term to terms on a page. This may be useful for veteran flutter users, but beginners looking for a certain property might be stuck. Finally, an in-built 'search this page' feature could be implemented, although ctrl+f or cmd+f seems to raise no complaints so far.

## 3. Content and examples

### Finding 3.1

Four out of five participants checked the examples provided on the Card class page first before browsing the textual content, trying to find examples that implement the same effect (e.g., adding a shadow, making corners rounded) as what they were asked to realize. However, none of them successfully learned how to add shadow to the card by scanning through the examples since none of the existing examples presents a card with a shadow behind it.

### Finding 3.2

All participants appreciated the interactive code boxes embedded in the Card Class page. *"I can quickly experiment my thoughts and play around with the variable to see what would happen, which is really great"* (P3). P2 also realized that the code box can debug while he tapped the code in, he was surprised by this feature and considered it to be helpful.

### Recommendations

Firstly, we suggest the examples include more variations. For example, besides making the three cards in Figure 1 varied in background colors, we can also add shadow to the first card, give the second card a sharp corner, and let the third card contain images.

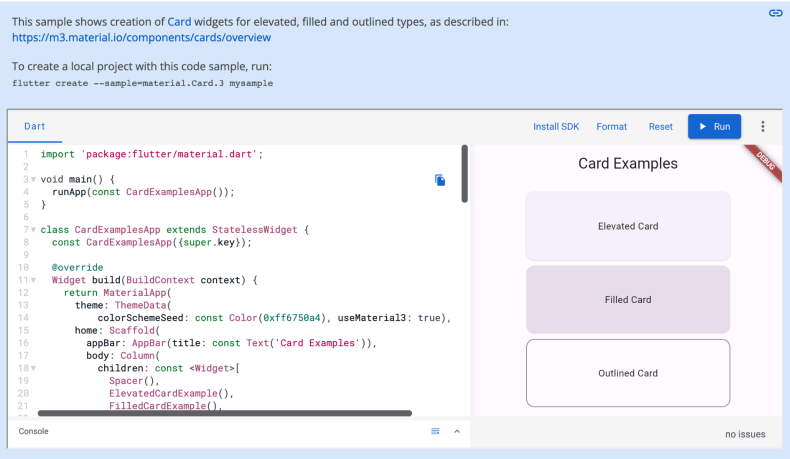


***Figure 2.*** *An example from the Card Class*

Secondly, to effectively communicate with users about what they can expect from the examples, it would be better to highlight the information that the examples aim to convey.

# Discussion

In the final phase of our research, we conducted a usability test on the Flutter API Documentation with a group of five Flutter novices to evaluate the site's usability in supporting information search and consumption.

### Limitations

Small sample size and limited variation in participants' experience with the Flutter API are the two limitations in our usability testing. However, we focus on the novices population because we found they tend to use the API documentation site more compared to experienced users in our interview studies. Also, we believe that if we test both novices and experienced users with the same list of tasks, the results would be less reliable.

### Next steps for the project

In the next phase of the project, we suggest that Google deploys a survey to gather more information about users' experience with Flutter API and their expectations of API documentation. This survey will help us understand users' needs and preferences better, enabling us to design a more effective usability test.
In addition, we suggest that

It would be helpful to also conduct a usability test with more users with various experiences using Flutter to evaluate the system's usability. It would be good to design different tasks to test different aspects of the site with a different population. This will result in collecting more comprehensive data, allowing our client to identify other potential usability issues and develop a documentation system that meets the needs of a broad range of users.

# Conclusion

In conclusion, this UX report presents the findings of a usability test conducted on the Flutter API Documentation website with the aim of identifying usability problems that prevent novice Flutter users from searching and consuming the site's information. The study evaluated how the site facilitates users to navigate themselves to the desired information and whether the site presents information in a user-friendly way for comprehension and supporting users' own programming practices.

The report highlights two major findings and recommendations related to site navigation and information search and one finding and recommendation around information consumption. The recommendations suggest that Flutter can consider consolidating and displaying related information on the same page by having the right sidebar menu to link and locate to a certain place on the current page instead of jumping to another page for more information. Also, the spacing and indentation of the right menu bar could be improved to make the information hierarchy better. Furthermore, the search bar and

search results page can be improved by providing more accurate results and suggesting specific properties to search for, enhancing the search experience for users.

Overall, this report provides valuable insights for improving the Flutter API Documentation website's user experience and overall usability for novice users. It is hoped that these recommendations will help Flutter to improve its website and provide a more user-friendly experience for its users, especially for those who are new to the Flutter framework.

## Appendices

**A. Warm-up interview questions**

- Can you tell me about your experience with programming in general? When did you first start, and what languages or tools have you worked with?
- Can you tell me more about your experience with Flutter?
- How comfortable do you feel with coding with Flutter? Do you consider yourself a beginner, intermediate, or advanced programmer?

**B. Post-test oral survey**

1. [Open-ended question] What frustrated you most about this site?
2. [Open-ended question] What did you like about the site?
3. [Scale-rating] From 1 (extremely unsatisfied) to 5 (extremely satisfied), for each task, please rate your satisfaction with the site in supporting you to complete the task.

**C. Usability test sheet for participants**

https://docs.google.com/document/d/1XU26SKNi2GV1i0YfjQmbZAtABaysqgXCEKqQithnp88/edit?usp=sharing

**D. Usability Test Analysis - the rainbow spreadsheet**

https://docs.google.com/spreadsheets/d/1aNQA6hMca9ILXas5X4HmKHemLj1VVWvr2VbypLj5KsE/edit?usp=sharing