Experimental features

- Project Fugu
  - Track capabilities
  - Request a capability
  - The process for each capability
- Origin trials
- Safari experimental features
- Firefox experimental features
- Resources

The web is a powerful platform, but there are still gaps in what it can solve. Those who want to develop for the web but need more different capabilities are forced to bundle their web apps in wrappers and publish them in app stores.

Developers may choose to ship their own custom browser as platform-specific apps, which disproportionately increases the size of their web apps. This will also force them to take on the additional security and maintenance burden of keeping both the browser fork and wrapper up to date.

This approach also loses the reach of the web, often being forced to choose what devices and operating systems to support, which often require different wrappers, and which further expands the security and maintenance burden.

Filling these capabilities gaps is the key to fixing this and thereby enabling the web to be the software platform of the future, covering as many use cases as possible, in comparison with platform-specific SDKs.

Project Fugu

This is where the capabilities project, known as Project Fugu, comes in. The capabilities project, or Project Fugu, is a cross-company effort to make it possible for web apps to have the same capabilities as platform-specific apps by exposing the capabilities of these platforms to the web, while maintaining user security, privacy, trust, and other core tenets of the web.

Track capabilities

There is a public tracker to keep up with all the work to ship new capabilities. On the tracker, you can see:

- The status of each API being worked on or considered.
- Platforms the API is targeted for.
- Links and additional information for each API.
- Search and filter capabilities.
- A timeline view.

Request a capability

What if you want to request a new capability? You can file an issue in the Chromium bug tracker, or you can go to webwewant.fyi and fill out a form to reach the corresponding browser vendors.

The process for each capability

Before launch, there are two states an in-progress API could be in, and both allow you to test them.

- A developer trial: the feature is behind a flag, the API isn't necessarily stable, and you shouldn't implement it for real users. You can enable or disable flags on Chromium-based browsers by going to `about:flags`, allowing you to test on your own browser instance.

{% Aside 'important' %} Under no circumstances should users ever be told to turn flags on or off. Because such features are still under development, changing flag defaults can have unanticipated consequences for the stability of the browser. {% endAside %}

- An origin trial: a state where features can be enabled by origin, developers can run tests for a wider audience than their single browser instance, but the feature is still being tested and the implementation can change, more details below.

Origin trials

Origin trials allow you to try out new features and give feedback to the web standards community on usability, practicality, and effectiveness. APIs available under origin trial are:

- Experimental, they may change and may become unavailable, including not extending beyond the experiment, and may be unavailable for some time, even if they are eventually enabled for all users. So where possible, you should implement feature

detection or graceful degradation to handle the case when the feature is unavailable.
- Stable enough to use with real users, but they may change throughout the course of the trial.
- Limited across all users globally to ensure they don't become a de facto standard, so it's recommended you activate the feature in your codebase following each browser's trial guidelines for a subset of your users.
- Limited to the browser vendor which starts the origin trial, so a Chrome origin trial will not work in Safari, Firefox, or Edge, for instance.

If these requirements are OK for you, you can register an origin to participate in a trial. You can find instructions to sign up for an origin trial in Chrome here and for Microsoft Edge follow this link

{% Img src="image/RK2djpBgopg9kzCyJbUSjhEGmnw1/d5ueJjfbepXwq72v2UjU.png", alt="A list of available origin trials for Google Chrome.", width="800", height="510" %}

{% Img src="image/RK2djpBgopg9kzCyJbUSjhEGmnw1/hSzBgQsAvgkCUN6h1RQN.png", alt="A list of available origin trials for Microsoft Edge.", width="800", height="541" %}

Safari experimental features

Safari also allows developers to try and test new and unfinished APIs and capabilities, but it doesn't provide origin trials. You can't enable Safari's experimental features for users.

Safari's solution is similar to flags, known as experimental features. Every new version of Safari comes with many experimental features, some of them enabled and some disabled by default.

As a developer, you can change those default settings using:

- The develop, experimental features menu in Safari for macOS.
- The iOS and iPadOS Settings App, by going to Safari, Advanced, Experimental Features.

{% Img src="image/RK2djpBgopg9kzCyJbUSjhEGmnw1/Y6agrnMYlpbQ970kYyYp.png", alt="Experimental features available on Safari on iPadOS.", width="800", height="525" %}

Firefox experimental features

Firefox offers experimental features through settings that you can enable or disable by accessing the Configuration Editor using `about:config`.

Resources

- The capabilities project
- New capabilities status
- Fugu Public Tracker
- Getting started with Chrome origin trials
- Microsoft Edge Origin Trials Guide
- The Web We Want, feedback form
- Firefox experimental features
- WebKit blog

□□□ #2
□ □□□ □□□ 2026-10-31 22:55:23 UTC
□ □□□ □□□ 2026-10-31 22:55:24 UTC