Make it installable

- Listen for the beforeinstallprompt event
- Handle the install button click
- Track the install event
- Further reading

This glitch contains the web manifest with the required fields to make a Web App installable.

It also has an install button that is hidden by default.

Listen for the beforeinstallprompt event

When the browser fires the `beforeinstallprompt` event, that's the indication that the Web App can be installed and an install button can be shown to the user. The `beforeinstallprompt` event is fired when the app meets the installability criteria.

Capturing the event enables developers to customize the installation and prompt the user to install when they consider it is the best time.

{% Instruction 'remix', 'ol' %}

1. Add a `beforeinstallprompt` event handler to the `window` object.
2. Save the `event` as a global variable; we'll need it later to show the prompt.
3. Unhide the install button.

Code:

```
window.addEventListener('beforeinstallprompt', (event) => {
  // Prevent the mini-infobar from appearing on mobile.
  event.preventDefault();
```

```
  console.log('👍', 'beforeinstallprompt', event);
  // Stash the event so it can be triggered later.
  window.deferredPrompt = event;
  // Remove the 'hidden' class from the install button container.
  divInstall.classList.toggle('hidden', false);
});
```

Handle the install button click

To show the install prompt, call `prompt()` on the saved `beforeinstallprompt` event. Calling `prompt()` is done in the install button click handler because `prompt()` must be called from a user gesture.

{% Aside 'gotcha' %} You need to open your app in its own browser tab. The side-by-side preview pane in Glitch doesn't work for this codelab, since it will be run in an iframe. Choose Glitch's "Preview in a new window" mode instead. {% endAside %}

1. Add a click event handler for the install button.
2. Call `prompt()` on the saved `beforeinstallprompt` event.
3. Log the results of the prompt.
4. Set the saved `beforeinstallprompt` event to null.
5. Hide the install button.

Code:

```
butInstall.addEventListener('click', async () => {
  console.log('👍', 'butInstall-clicked');
  const promptEvent = window.deferredPrompt;
  if (!promptEvent) {
    // The deferred prompt isn't available.
    return;
  }
  // Show the install prompt.
  promptEvent.prompt();
  // Log the result
  const result = await promptEvent.userChoice;
  console.log('👍', 'userChoice', result);
  // Reset the deferred prompt variable, since
  // prompt() can only be called once.
  window.deferredPrompt = null;
```

```
  // Hide the install button.
  divInstall.classList.toggle('hidden', true);
});
```

Track the install event

Installing a Web App through an install button is only one way users can install it. They can also use Chrome's menu, the mini-infobar, and through an icon in the omnibox. You can track all of these ways of installation by listening for the `appinstalled` event.

1. Add an `appinstalled` event handler to the `window` object.
2. Log the install event to analytics or other mechanism.

Code:

```
window.addEventListener('appinstalled', (event) => {
  console.log('□□, 'appinstalled', event);
  // Clear the deferredPrompt so it can be garbage collected
  window.deferredPrompt = null;
});
```

Further reading

Congratulations, you app is now installable!

Here are some additional things that you can do:

- Detect if your app is launched from the home screen
- Show the operating system's app install prompt instead

□□□ #2
□ □□□ □□□ 2026-10-31 23:08:43 UTC
□ □□□ □□□ 2026-10-31 23:08:44 UTC