PWAs Today

Ire Aderinokun

Cofounder and VP Engineering @ Buycoins

PWA SUMMIT 2021

Websites were **less capable**

Websites had **no offline support**

Websites were **harder to access**

Image from https://web.dev/what-are-pwas
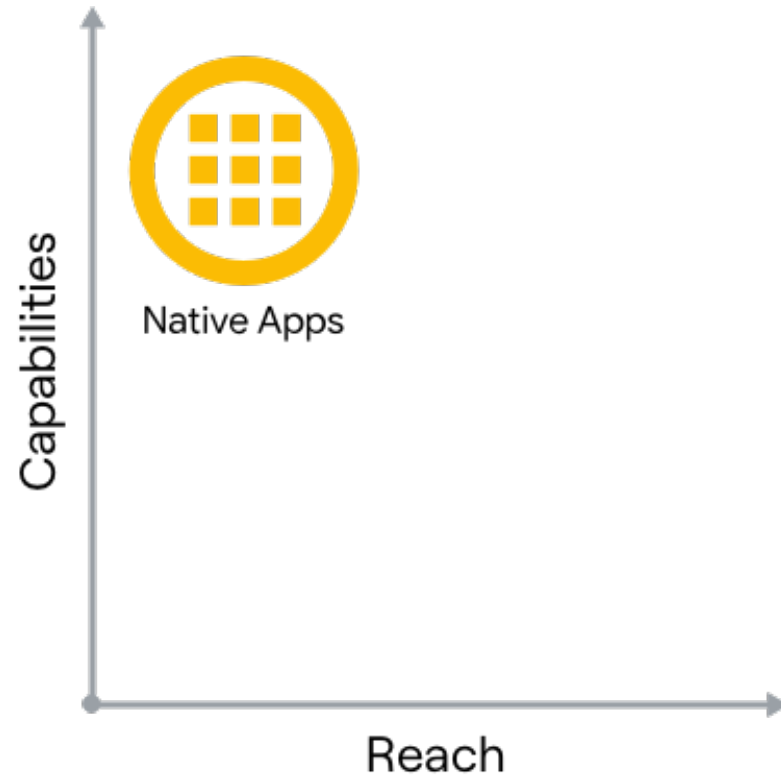
Image from https://web.dev/what-are-pwas

Image from https://web.dev/what-are-pwas

Progressive Web Apps use **modern web capabilities to deliver an app-like user experience**. They evolve from pages in browser tabs to immersive, top-level apps, maintaining the web's low friction at every moment.

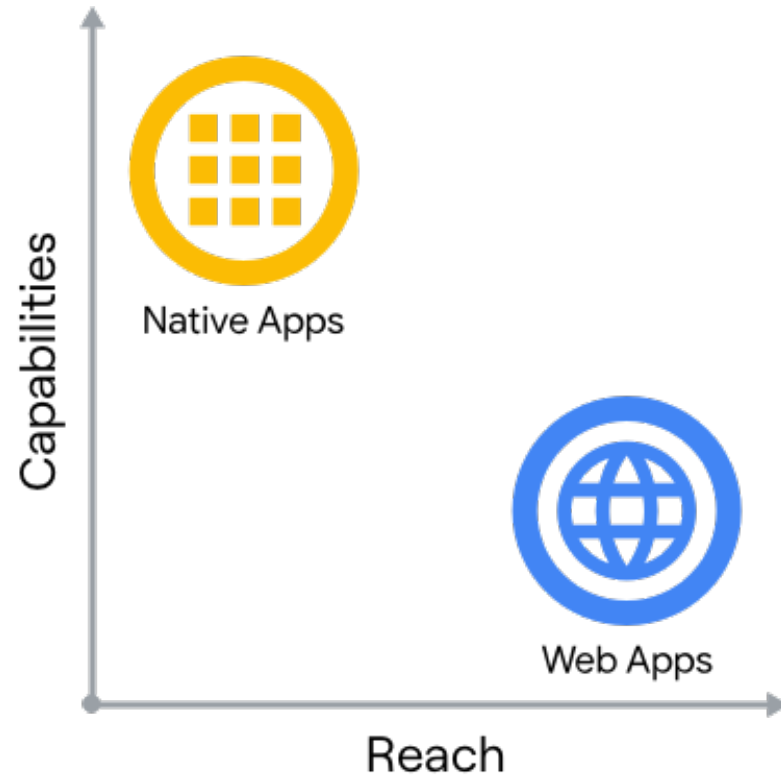https://developers.google.com/web/updates/2015/12/getting-started-pwa#what_is_a_progressive_web_app

PWA SUMMIT 2021
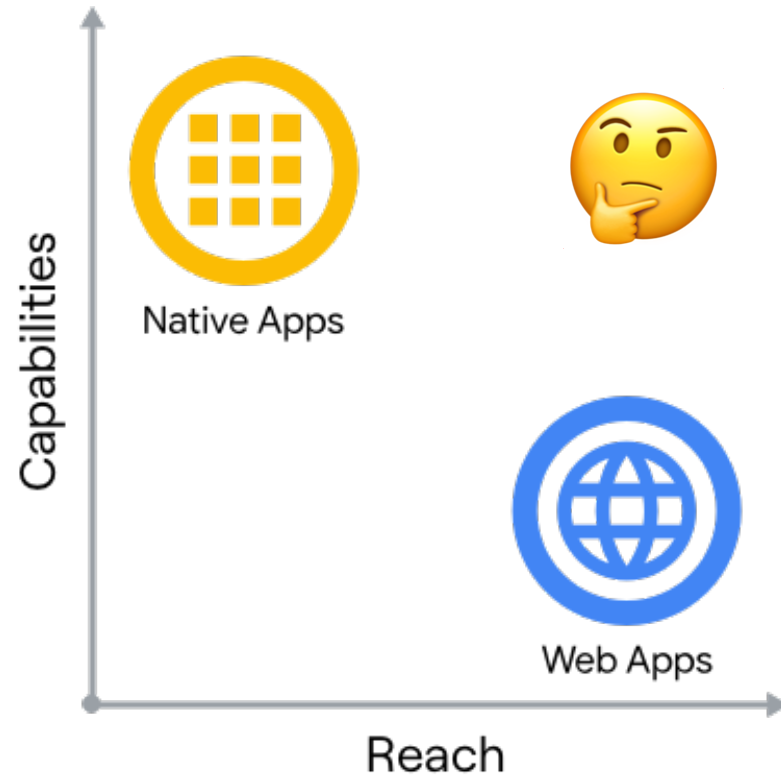
Progressive Web Apps use modern web capabilities to deliver an app-like user experience. They evolve from pages in browser tabs to immersive, top-level apps, **maintaining the web's low friction** at every moment.

https://developers.google.com/web/updates/2015/12/getting-started-pwa#what_is_a_progressive_web_app

Image from https://web.dev/what-are-pwas

💪 Capable   ✅ Reliable   ⬇️ Installable

Capable — uses modern APIs to deliver an app-like experience

PWA SUMMIT 2021

Looking for new shoes?
www.shoes.com
Check our brand new summer collection!

Shoes Summer Collection
www.shoes.com
Our new shoes collection is waiting for you!

MacBook

Image from https://www.opal.ai/web-push-notifications

WEBASSEMBLY

Reliable — works well regardless of network

OFF

ON LINE

Installable — can be installed to the device and used outside the browser
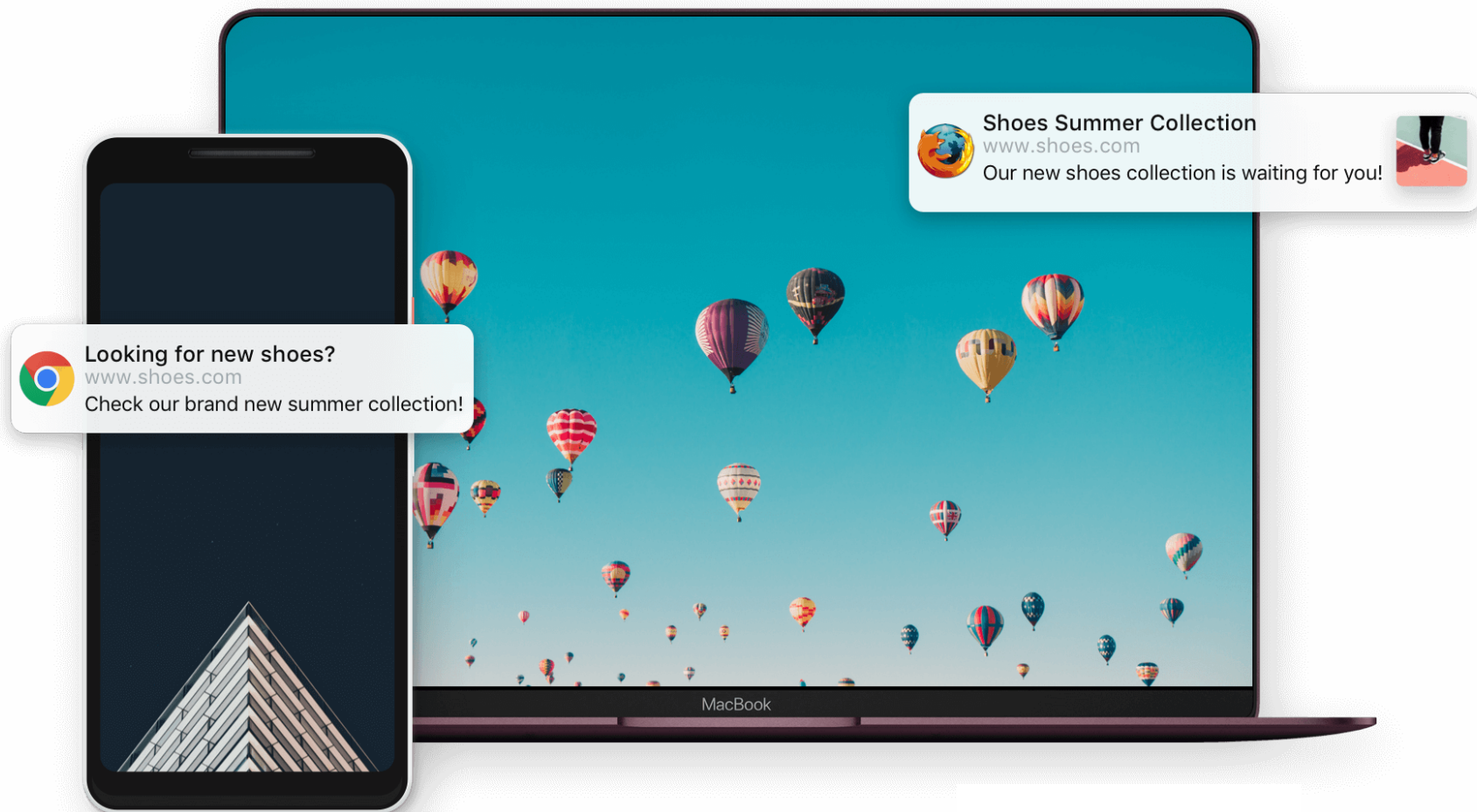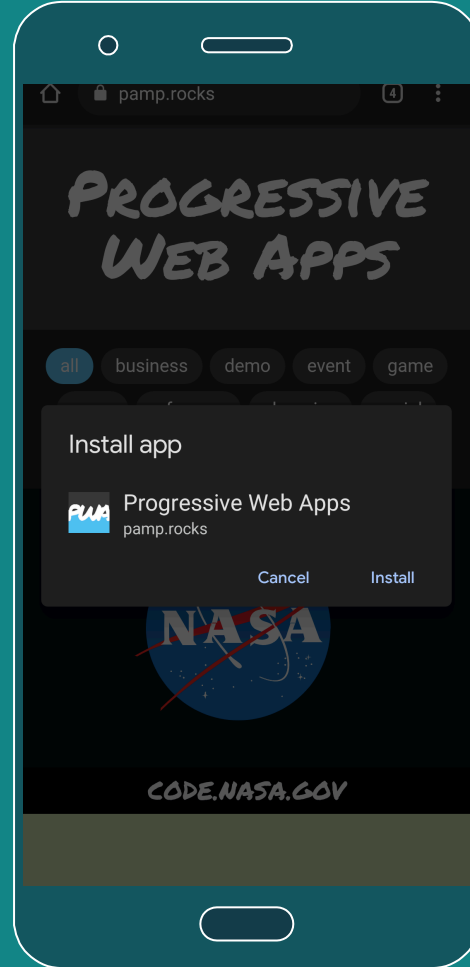
Install app

Progressive Web Apps
pamp.rocks

Cancel   Install
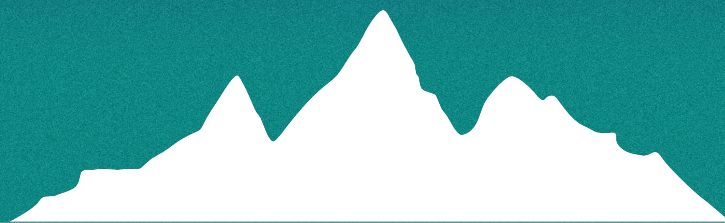
https://pamp.rocks/
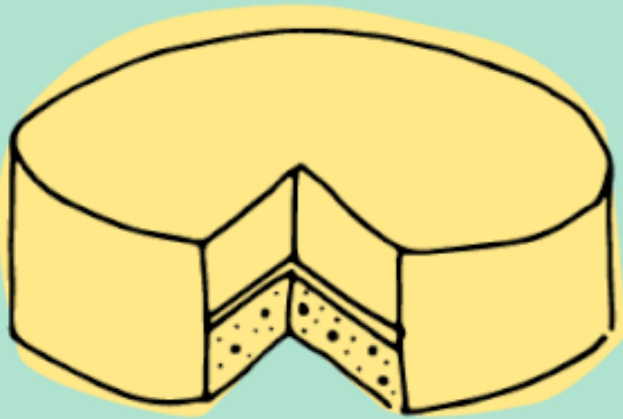
Standard website

Fully capable, reliable, and installable PWA

In 2016,
PWAs weren't really "PWAs"

Push API 📄 - WD

API to allow messages to be pushed from a server to a browser,
even when the site isn't focused or even open in the browser.

Current aligned | Usage relative | **Date relative** | Filtered | **All** | ⚙

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android
		36	41		27		all				
		37	42		28						
		38	43		29						
	12	39	1 2 44		30						
		40	1 2 45		31	9.2					
	13	41	1 2 46	9	32						
		42	1 2 47		33						
2016		43	1 2		34						

BackgroundFetchEvent API

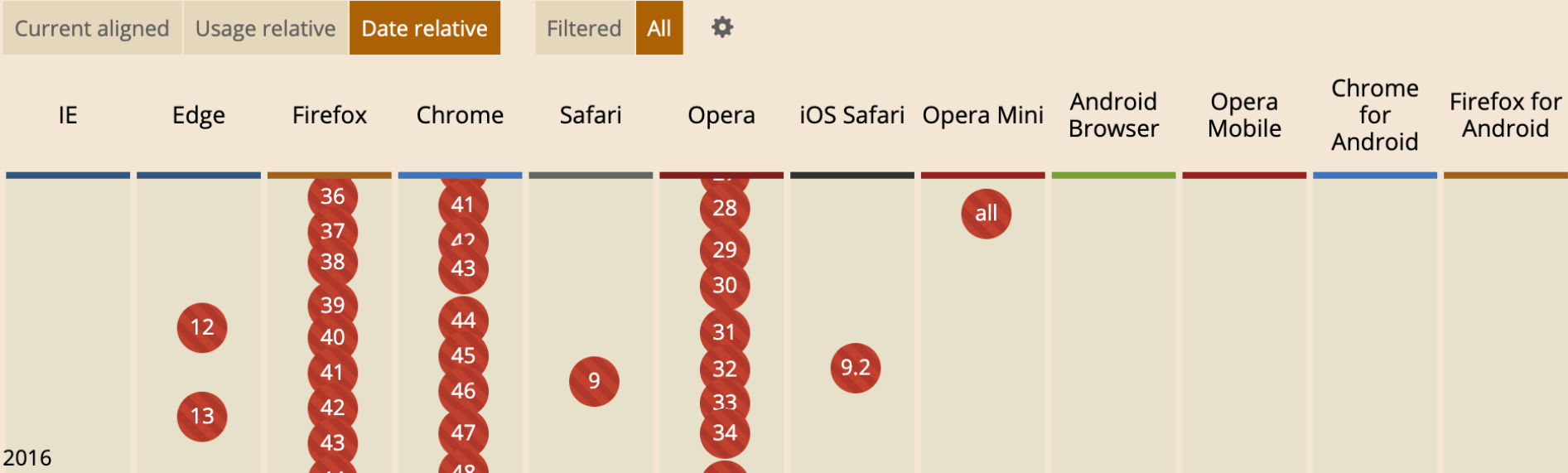| IE	Edge	Firefox	Chrome	Safari	Opera	Safari on iOS	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android
		36	40		27		all				
		37	41		28						
		38	42		29						
		39	43		30						
	12	40	44		31	9.2					
		41	45	9	32						
	13	42	46		33						
2016		43	47		34						
			48								
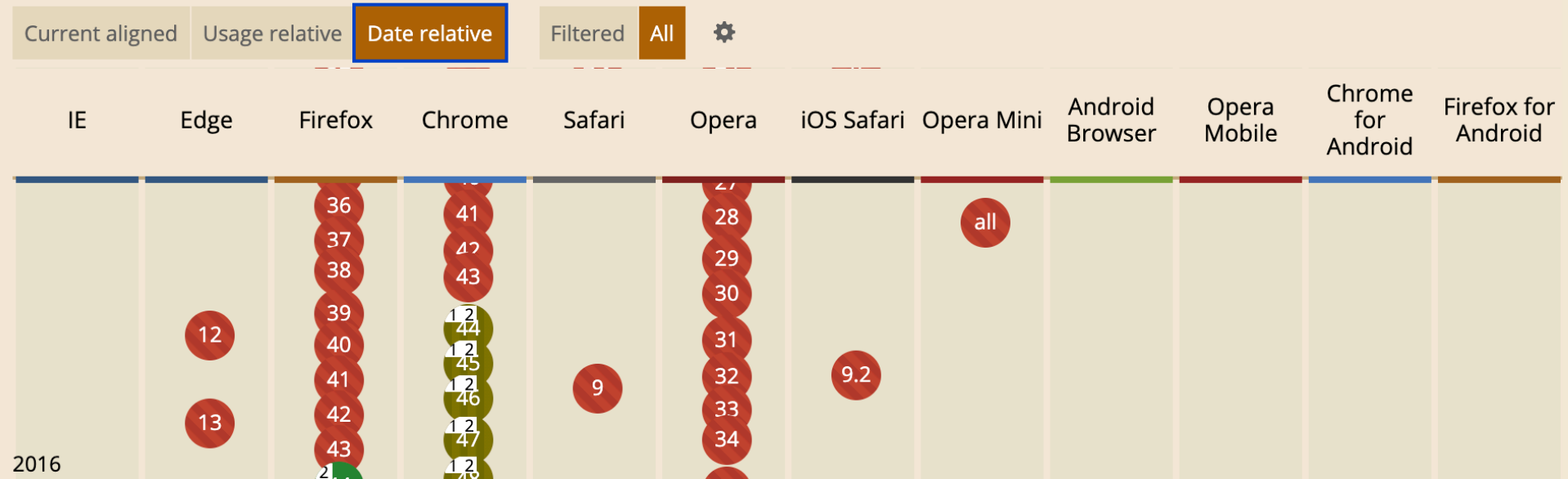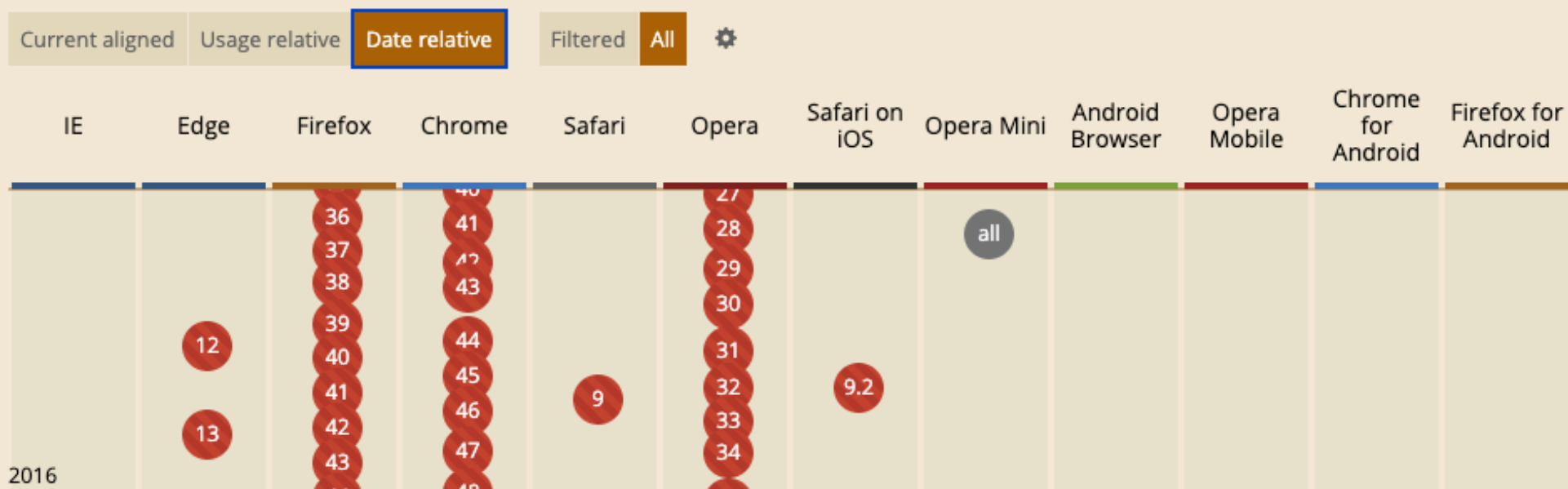
File System Access API 📄 - UNOFF

API for manipulating files in the device's local file system (not in a sandbox).

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android
	12	36	41		28		all				
	13	37	42		29						
		38	43		30	9.2					
		39	44	9	31						
		40	45		32						
		41	46		33						
		42	47		34						
		43	48								

2016

```javascript
if (awesomeFeature in window) {

    // do awesome thing

} else {

    // same old thing

}
```

is SERVICEWORKER ready?

ServiceWorker enthusiasm

The first thing any implementation needs.
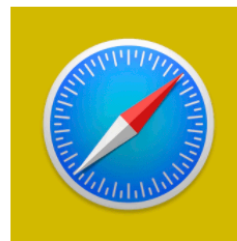
Firefox: Follow Bug 903441 for the latest development.

Safari: Under consideration, Brief positive signals in five year plan.

Edge: Under consideration, but positive signals.

Support does not include iOS versions of third-party browsers on that platform (see Safari support).

Promises

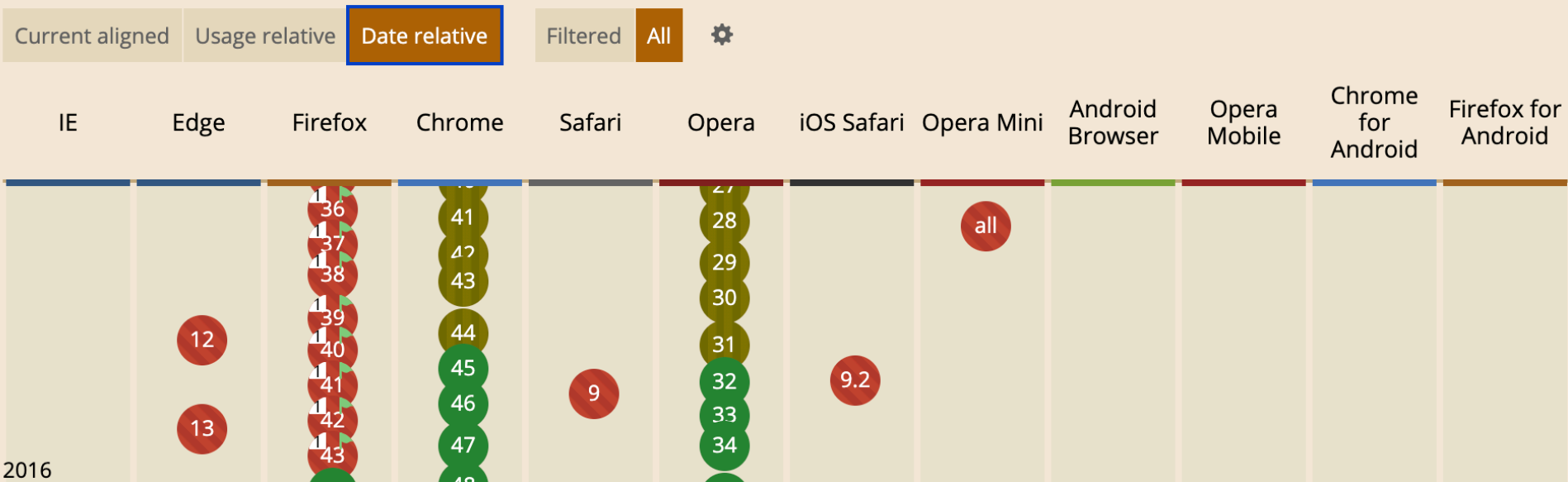Not required by ServiceWorker. Spec.

36+   29+   23+   7.1+

Service Workers 📄 - CR

Method that enables applications to take advantage of persistent background processing, including hooks to enable bootstrapping of web applications while offline.

| Current aligned | Usage relative | Date relative | | Filtered | All | ⚙ |

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android
		36	41		27		all				
		37	42		28						
		38	43		29						
	12	39	44		30						
		40	45		31						
		41	46	9	32	9.2					
	13	42	47		33						
		43	48		34						

2016

Why my Service Worker is always waiting to activate?
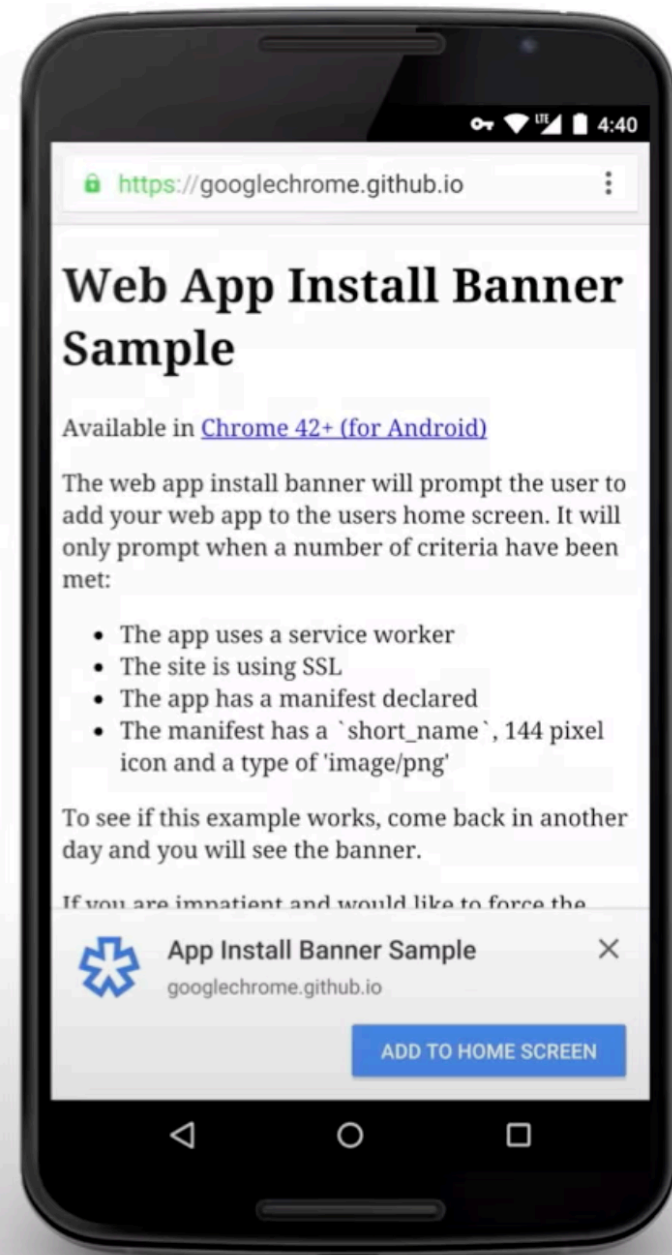
Asked 3 years, 6 months ago   Active 3 years, 6 months ago   Viewed 13k times
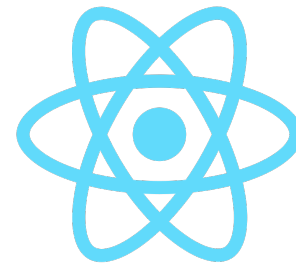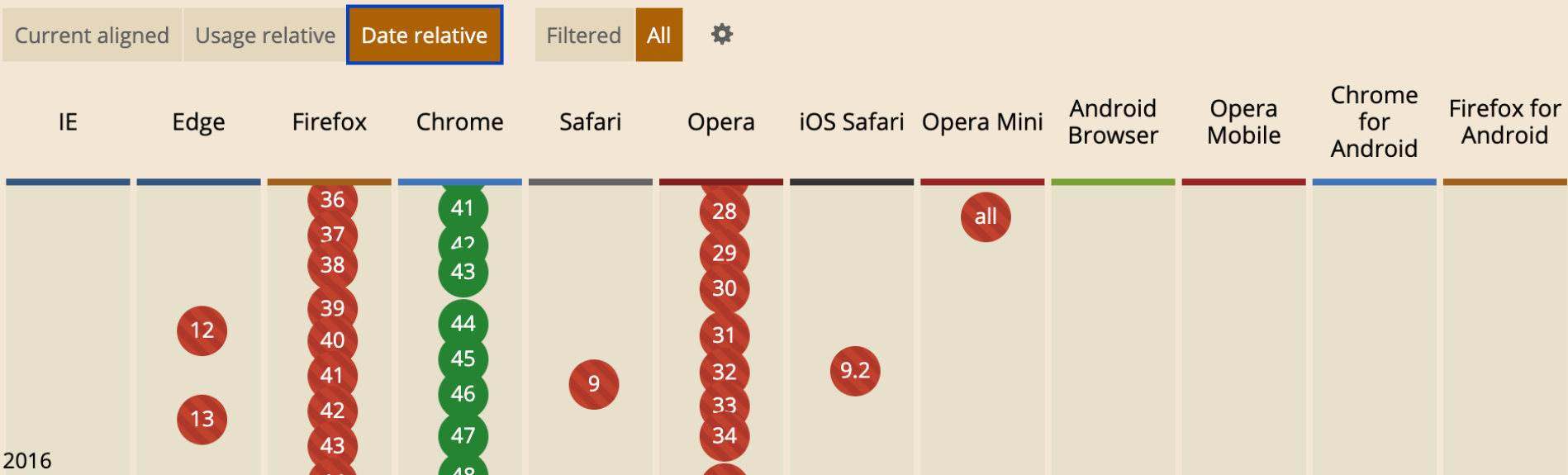
Add to Homescreen !== Install

Requirements for "Add to Home Screen"

1. A manifest.json file

2. A service worker

3. Visit frequency heuristics 🤔

Add to home screen (A2HS) 📄 - WD

The ability for a user to "install" a website and use it as if it was a natively installed app. To enable this behaviour, a website must serve a valid Web App Manifest and load it's assets through a [Service Worker](#).

Current aligned	Usage relative	**Date relative**		Filtered	**All**	⚙

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android
		36	41		28		all				
		37	42		29						
		38	43		30						
	12	39	44		31	9.2					
		40	45	9	32						
	13	41	46		33						
2016		42	47		34						
		43	48								

🙁 Capable

🙁 Reliable

🙁 Installable

PWAs Today
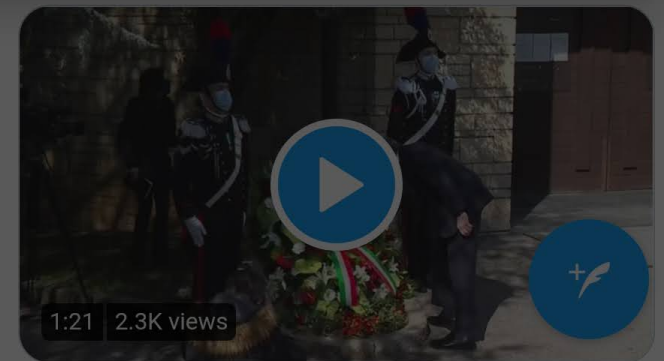
~1,500,000 websites may be installable on mobile home screens, offering an app experience

https://firt.dev/pwa-2021

PWA SUMMIT 2021
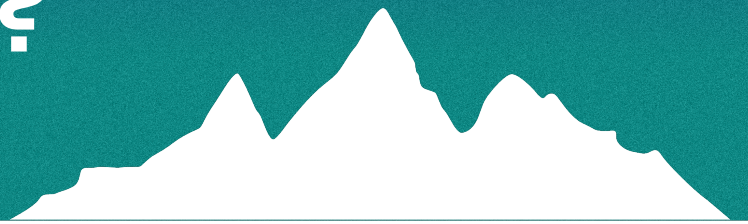
Almost **20% of page loads** are controlled by Service Workers

Sep 1, 2021
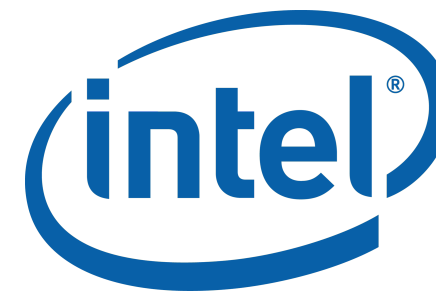Monthly Average:18.717387
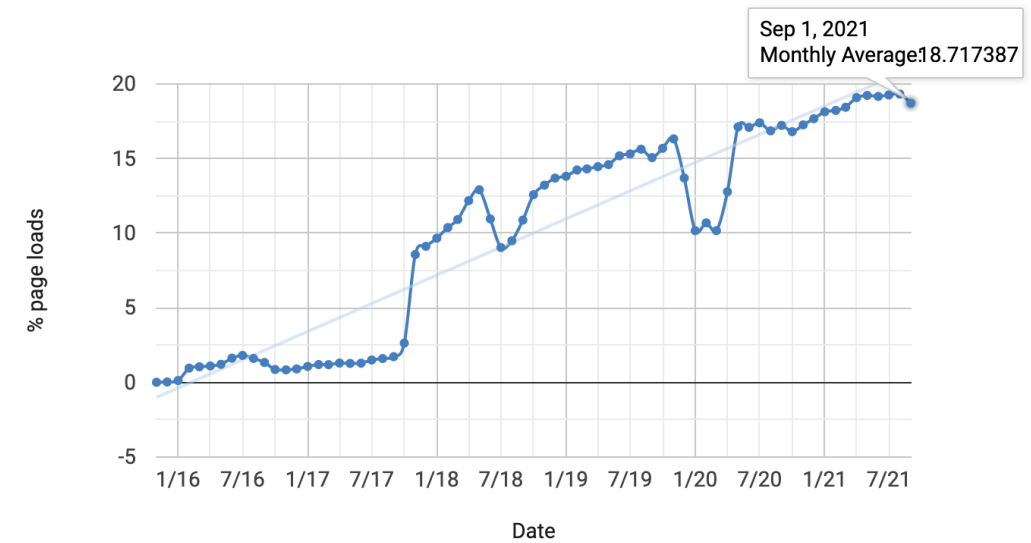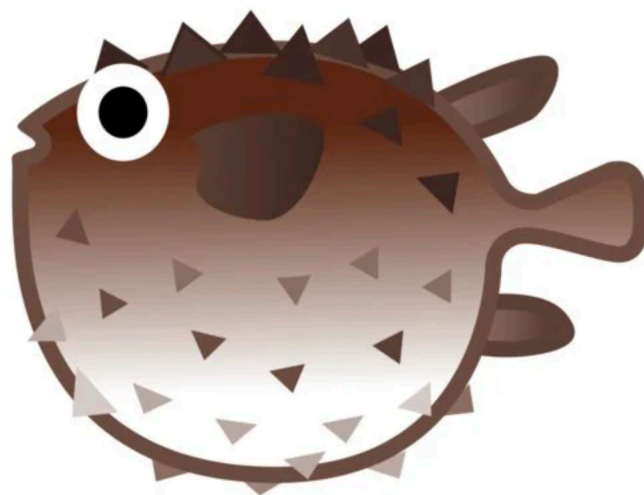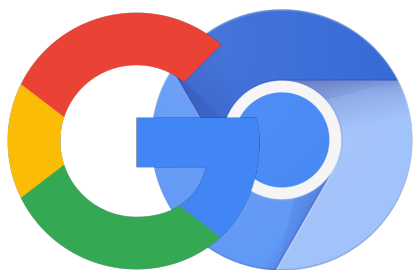
🧐 Capable

🧐 Reliable

🧐 Installable

How capable are PWAs today?

Microsoft

SAMSUNG

intel

Project Fugu is an effort to **close gaps in the web's capabilities** by enabling new classes of applications to run on the web

http://www.chromium.org/teams/web-capabilities-fugu

PWA SUMMIT 2021

New Web Capabilities and You

5:10 PM - 5:30 PM

Project Fugu, also known as the Web Capabilities project, is a cross-company effort between Google, Microsoft, Intel, and others, to identify, spec, and ship new powerful capabilities to the web without compromising the core safety of the platform. Learn about the process for shipping these new features, how to try them out and give feedback before they're finalized, what some of these new powerful APIs are like to use, and why you should start considering them for your project today.

Maye Edwin

What has been **shipped**? 🚀

PWAs today are **engaging**

Notifications

Web notifications

☑ **Web notifications**
Enabled

☑ **Offline notifications**
Enabled

Sound   5

Private Chats

☑ **Notifications for private chats**
Enabled

☐ **Message Preview**
Disabled

Groups

☐ **Notifications for groups**
Disabled

☐ **Message Preview**
Enabled

← **Push notifications**

New matches
You just got a new match.   ⬤

Messages
Someone sent you a new message.   ⬤

Message likes
Someone liked your message.   ⬤

Super Like
You've been Super Liked.   ⬤

← **Push notifications**
@ireaderinokun

Push notifications   ◯

Get push notifications to find out what's going on when you're not on Twitter. You can turn them off anytime.

Turn on push notifications

To receive notifications as they happen, turn on push notifications. You'll also receive them when you're not on Twitter. Turn them off anytime.

Turn on

← **Push notifications**

Likes

◯ Off
◯ From people I follow
⦿ From everyone

johnappleseed liked your photo.

Comments

⦿ Off
◯ From people I follow
◯ From everyone

johnappleseed commented: "Nice shot!"

Comment likes and pins

⦿ Off
◯ From people I follow

johnappleseed liked your comment: "Nice shot!" and other similar notifications.

Likes and comments on photos of you

ff
From people I follow

Web Notifications API

```
Notification.requestPermission();

new Notification("Hello PWA Summit!");
```

Push API

```
self.addEventListener('push', function(event) {

    event.waitUntil(

        self.registration.showNotification('Hello PWA Summit!', { body: 'Hi!' })

    );

});
```

Web Notifications

Method of alerting the user outside of a web page by displaying notifications (that do not require interaction by the user).

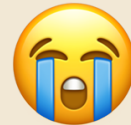| IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	90	89	90	13.1	13.7			4.4	12.0
10	91	90	91	14	14.4			4.4.4	13.0
11	92	91	92	14.1	14.7	all	92	92	14.0
		92	93	15					

✓ ✗ Partial Support Prefixed

Global: 77.95% + 0.71% = 78.66%

Web Notifications

Method of alerting the user outside of a web page by displaying notifications (that do not require interaction by the user).

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	90	89	90	13.1	13.7			4.4	12.0
10	91	90	91	14	14.4			4.4.4	13.0
11	92	91	92	14.1	14.7	all	92	92	14.0
		92	93	15					

😭

✓	✗	Partial Support	Prefixed

Global: 77.95% + 0.71% = 78.66%

Push API

API to allow messages to be pushed from a server to a browser, even when the site isn't focused or even open in the browser.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	90	89	90	13.1	13.7			4.4	12.0
10	91	90	91	14	14.4			4.4.4	13.0
11	92	91	92	14.1	14.7	all	92	92	14.0
		92	93	15					

✓ ✗ Partial Support

✱

Twitter

App Badging API

```
navigator.setAppBadge(2);

navigator.clearAppBadge();
```

Notification API: badge

Current aligned | Usage relative | Date relative   Filtered | **All** ⚙

IE	Edge*	Firefox	Chrome	Safari	Opera	Safari on iOS*	Opera Mini*	Android Browser*	Opera Mobile*	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet
	12-17		4-52		10-38								4-5.4
6-10	18-92	2-91	53-92	3.1-14	39-78	3.2-14.7		2.1-4.4.4	12-12.1				6.2-13.0
11	93	92	93	14.1	79	15	all	93	64	93	92	12.12	14.0
		93-94	94-96	15									
				TP									

moz://a

| ▼ | 👤 | Badging API | | worth prototyping | 💬 |

Description This specification defines an API allowing web applications to set an application-wide badge, shown in an operating-system-specific place associated with the application (such as the shelf or home screen), for the purpose of notifying the user when the state of the application has changed (e.g., when new messages have arrived), without showing a more heavyweight notification.
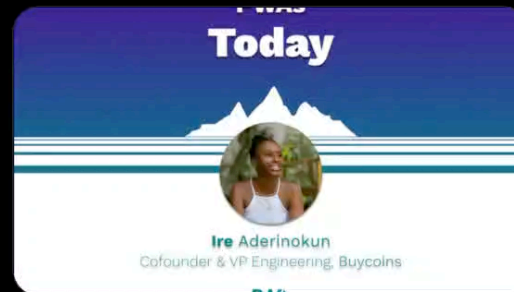
https://mozilla.github.io/standards-positions/#badging

PWAs today can **interact with other apps**

Ire Aderinokun ✔
11.3K Tweets

PWASummit @PwaSummit · Sep 7 ···

"PWAs Today" with @ireaderinokun. 7th of October 2021, only at pwasummit.org. Get your free ticket today! #pwasummit

💬   ⟲ 7   ♥ 15   ⤴

Topics to follow

Tweets about the Topics you follow show up in your Home timeline

Web development ➕	✖	Cloud computir
Open source ➕	✖	Databases
Computer programming ➕	✖	FinTech
Cloud platforms ➕	✖	Startups
Machine learning ➕	✖	Data science

More Topics

Web Share API

```
navigator.share({

    title: "",

    text: "",

    url: "",

});
```

PWA SUMMIT
6 & 7 of October

PWA SUMMIT

A free, online, single-track conference focused on helping everyone succeed with Progressive Web Apps.

Add to calendar 📅

Register for free 🎟

Schedule

Web Share Target API

```
"share_target": {

    "action": "compose/tweet",

    "enctype": "multipart/form-data",

    "method": "POST",

    "params"

},
```

Web Share API

A way to allow websites to invoke the native sharing capabilities of the host platform

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	91	90	91	13.1	14.4			4.4	12.0
10	92	91	92	14	14.7			4.4.4	13.0
11	93	92	93	14.1	15	all	93	93	14.0
		93	94	15					

✓	✗	Partial Support

Global: 60.02% + 26.88% = 86.90%

Web Share Target API

PWAs today have **real access to the device**

Pull requests   Issues   Marketplace   Explore

Ire Aderinokun
Your personal account   ⇄ Switch to another account ▾

Go to your personal profile

Account settings ✓

Profile

Account

Appearance

Account security

Billing & plans

Security log

Security & analysis

Sponsorship log

Emails

Notifications

Scheduled reminders

Theme preferences

Choose how GitHub looks to you. Select a single theme, or sync with your system and automatically switch between day and night themes.

Theme mode

Sync with system ⇅   GitHub theme will match your system active settings

☀ Day theme

This theme will be active when your system is set to "light mode"
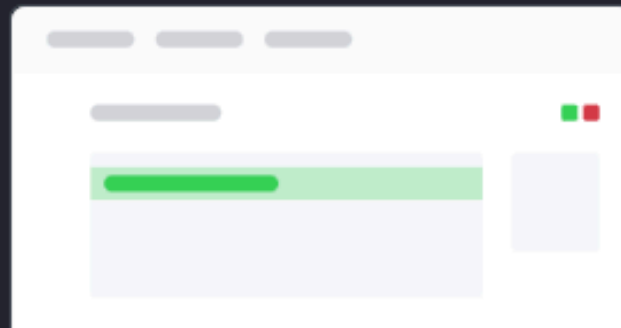
🌙 Night theme   Active

This theme will be active when your system is set to "dark mode"
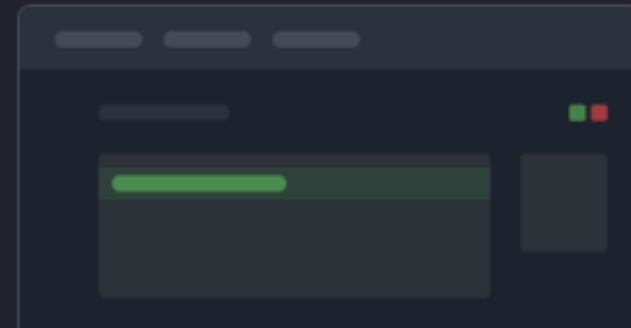
Contact Picker API Demo

Access to the user's contacts has been a feature of native apps since (almost) the dawn of time. The **Contact Picker API** is a new, on-demand picker that allows users to select an entry or entries from their contact list and share limited details of the selected contact(s) with a website. It allows users to share only what they want, when they want, and makes it easier for users to reach and connect with their friends and family.

Note: This demo requires Chrome 80 or later running on Android M or later to work properly.

- [ ] Multiple results?
- [x] Include name?
- [ ] Include email addresses?
- [ ] Include telephone numbers?
- [ ] Include addresses? (Chrome 84 and later)
- [x] Include icons? (Chrome 84 and later)

This demo **does not** share or upload your contacts. It simply shows them on screen, and no contact data is transferred off this device.

Open contact picker

Results

- Results will be populated here.
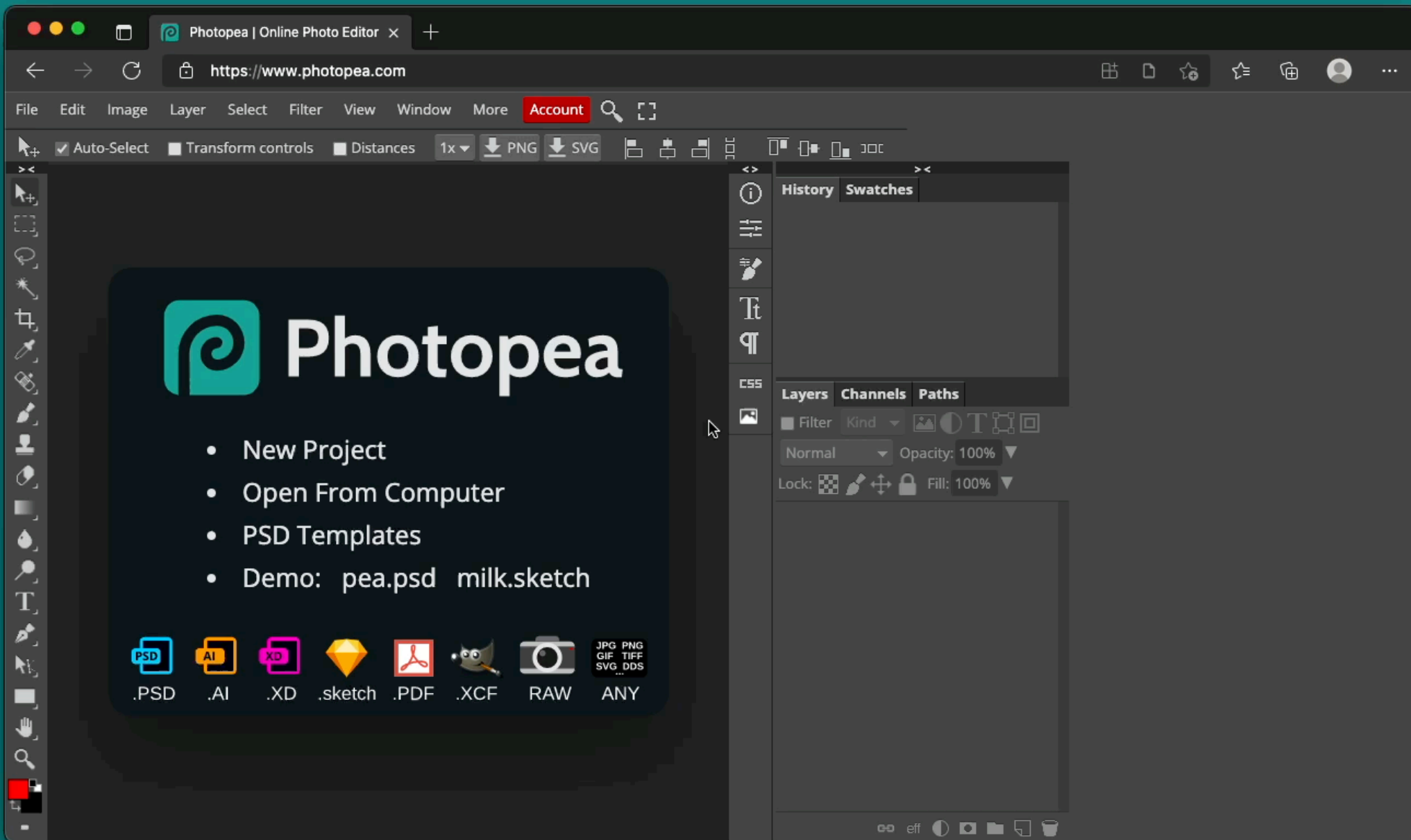
prefers-color-scheme media query

Media query to detect if the user has set their system to use a light or dark color theme.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	91	90	91	13.1	14.4			4.4	12.0
10	92	91	92	14	14.7			4.4.4	13.0
11	93	92	93	14.1	15	all	93	93	14.0
		93	94	15					

✓ ✗ Partial Support

Global: 91.96% + 0% = **91.96%**

Navigator API: contacts

Global   42.1%

| Current aligned | Usage relative | Date relative | | Filtered | All | ⚙ |

Safari on iOS *	Opera Mini *	Android Browser *	Opera Mobile *	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet	QQ Browser
							4-12.0	
3.2-14.7		2.1-4.4.4	12-12.1				13.0	
15	all	93	64	93	92	12.12	14.0	10.4

File System Access API

API for manipulating files in the device's local file system (not in a sandbox).

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	90	89	90	13.1	13.7			4.4	12.0
10	91	90	91	14	14.4			4.4.4	13.0
11	92	91	92	14.1	14.7	all	92	92	14.0
		92	93	15					

✓ ✗ Partial Support

PWAs today can **work in the background**

10:09

Wed, 15 Sep ☁ 17°C

🔊 Chrome   📱 Phone speaker

Champagne Poetry
Drake   ⏸ ⏭

Media Session API

```
navigator.mediaSession.metadata = new MediaMetadata({

    title: 'Champagne Poetry',

    artist: 'Drake',

    album: 'Certified Lover Boy',

    artwork: [{ src: '…/512', sizes: '512x512', type: 'image/png' },]

});
```

MediaSession API: metadata

Current aligned	Usage relative	Date relative		Filtered	All	⚙

IE	Edge	Firefox *	Chrome	Safari	Opera	Safari on iOS *	Opera Mini *	Android Browser *	Opera Mobile *	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet
		2-70											
	12-18	[1] 71-81 🚩	4-72										4-6.4
6-10	79-92	82-91	73-92	3.1-14	10-78	3.2-14.7		2.1-4.4.4	12-12.1				7.2-13.0
11	93	92	93	14.1	79	15	all	93	64	93	[2] 92	12.12	14.0
		93-94	94-96	15-TP									

Is Safari the new IE?
And lots of tangents

When the hype train turns out to be a b...
Also image previews, performance, and WA...

Blog build tools, JS blocks, and opener-...
Also, dog teeth, Jedi safety, and limescale

How does back/forward actually work, ...
Also, active clothing protocols, hotel ratings...

The big build-tool bonanza
The present and future of webpack, Rollup, ...

How to avoid getting UTF'd by text enc...
And how to be DOM-ready for anything.

Background Fetch API

```javascript
navigator.serviceWorker.ready.then(async (registration) ⇒ {

    await registration.backgroundFetch.fetch(

        'video-fetch', ['/video.mp4'], { /* options */ }

    );

});
```

BackgroundFetchEvent API

Usage
% of all users   ?

Global   71.84%

Current aligned Usage relative Date relative   Filtered **All** ⚙

IE	Edge*	Firefox	Chrome	Safari	Opera	Safari on iOS*	Opera Mini*	Android Browser*	Opera Mobile*	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet
	12-18		4-73		10-60								4-10.1
6-10	79-92	2-91	74-92	3.1-14	62-78	3.2-14.7		2.1-4.4.4	12-12.1				11.1-13.0
11	93	92	93	14.1	79	15	all	93	64	93	92	12.12	14.0
		93-94	94-96	15									
				TP									

https://www.chromestatus.com/metrics/feature/timeline/popularity/2549

Image from https://robert-askam.co.uk/posts/how-do-make-a-post-request-to-a-server-using-background-sync-and-service-workershttps://

PeriodicSyncEvent API

IE	Edge *	Firefox	Chrome	Safari	Opera	Safari on iOS *	Opera Mini *	Android Browser *	Opera Mobile *	Chrome for Android	Firefox for Android
	12-79		4-79		10-66						
6-10	80-91	2-90	80-92	3.1-14	67-77	3.2-14.4		2.1-4.4.4	12-12.1		
11	92	91	93	14.1	78	14.7	all	92	64	92	90
		92-93	94-96	15							
				TP							

https://www.chromestatus.com/metrics/feature/timeline/popularity/2549
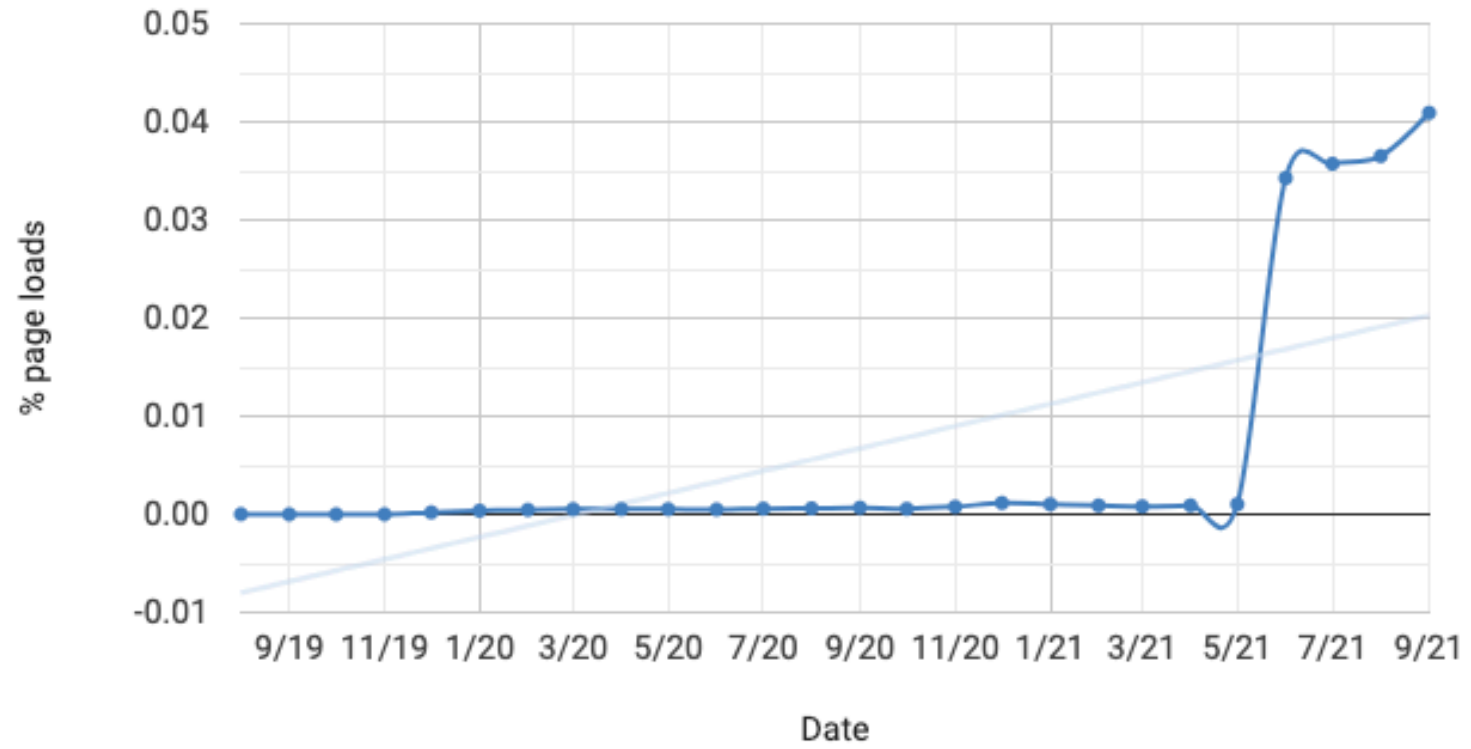
Browsers:
"We've launched 30+ features to make the web more capable"

Developers:
"New features you say?"

Developers:
Doesn't implement new features

Fugu API Tracker

	STABLE	BETA	DEV
	Chromium 93	Chromium 94	Chromium 95
	Stable 21 days ago (Aug 31, 2021)	Stable in 0 days (Sep 21, 2021)	Stable in 28 days (Oct 19, 2021)

Shipped

API	Milestone	Platforms	
Web Bluetooth API	M56		+
WebUSB API	M61	🐧 🪟 💿 🍎 🤖	+
Web Share Target	M71	🤖	+
Web Share API Level 2	M75	🤖	+
Web Share Target Level 2	M76	🤖	+
Async Clipboard: Read and Write Images	M76	🐧 🤖 🪟 💿 🍎	+

https://fugu-tracker.web.app/

Planning a Progressive Web App

8:40 PM - 9:00 PM

A Progressive Web App can be as simple as adding a service worker and a manifest file to an existing website or as complex as spotify.com. How do you decide which features of a PWA make sense for your users and your business? What does it mean to be an "app" and how much does your PWA have to feel like one? This session will provide a framework for thinking about Progressive Web App features and how to put together a roadmap that delivers incremental improvements that benefit users.

Jason Grigsby

Gerardo Rodriguez

How capable are PWAs today?

✅ Close to feature parity with native applications

❌ Platform parity

❌ Usage of features

How capable are PWAs today?

✅ Close to feature parity with native ~~applicati~~ons 🙂

❌ Platform parity

❌ Usage of features

How reliable are PWAs today?

is
SERVICEWORKER
ready?

https://jakearchibald.github.io/isserviceworkerready/

Status * Spec * Intro * Resources * GitHub

is
SERVICEWORKER?
ready?

Yes.

Status * Spec * Intro * Resources * GitHub

Service Workers

Method that enables applications to take advantage of persistent background processing, including hooks to enable bootstrapping of web applications while offline.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	90	89	90	13.1	13.7			4.4	12.0
10	91	90	91	14	14.4			4.4.4	13.0
11	92	91	92	14.1	14.7	all	92	92	14.0
		92	93	15					

✓ ✗ Partial Support

Global: 95.78% + 0.09% = 95.87%

Why my Service Worker is always waiting to activate?

Asked 3 years, 6 months ago   Active 3 years, 6 months ago   Viewed 13k times

SKIPWAITING 🤩

Cache JS & CSS

```
import {registerRoute} from 'workbox-routing';

import {StaleWhileRevalidate} from 'workbox-strategies';


registerRoute(

   ({request}) ⇒ request.destination ≡ 'script' || request.destination ≡ 'style',

   new StaleWhileRevalidate()

);
```

Offline Google Analytics

```
import * as googleAnalytics from 'workbox-google-analytics';

googleAnalytics.initialize();
```

Service Offline Page

```
const networkOnly = new NetworkOnly();

registerRoute( new NavigationRoute(async (params) ⇒ {

    try {

        return await networkOnly.handle(params);

    } catch (error) {

        return caches.match(FALLBACK_HTML_URL, { cacheName: CACHE_NAME });

    }

}) );
```

App shell

App shell is a way to render a portion of your application via a route at build time. It can improve the user experience by quickly launching a static rendered page (a skeleton common to all pages) while the browser downloads the full client version and switches to it automatically after the code loads.

This gives users a meaningful first paint of your application that appears quickly because the browser can simply render the HTML and CSS without the need to initialize any JavaScript.

Learn more in The App Shell Model ↗.

Step 1: Prepare the application

You can do this with the following CLI command:

```
ng new my-app --routing
```

Making a Progressive Web App

The production build has all the tools necessary to generate a first-class Progressive Web App, but **the offline/cache-first behavior is opt-in only**.

Starting with Create React App 4, you can add a `src/service-worker.js` file to your project to use the built-in support for Workbox's `InjectManifest` plugin, which will compile your service worker and inject into it a list of URLs to precache.

If you start a new project using one of the PWA custom templates, you'll get a `src/service-worker.js` file that serves as a good starting point for an offline-first service worker:

```
npx create-react-app my-app --template cra-template-pwa
```

The TypeScript equivalent is:

```
npx create-react-app my-app --template cra-template-pwa-typescript
```

If you know that you won't be using service workers, or if you'd prefer to use a different approach to creating your service worker, don't create a `src/service-worker.js` file. The `InjectManifest` plugin won't be run in that case.

In addition to creating your local `src/service-worker.js` file, it needs to be registered before it will be used. In order to opt-in t

Core Vue CLI Plugins

@vue/cli-plugin-pwa

> pwa plugin for vue-cli

The service worker added with this plugin is only enabled in the production environment (e.g. only if you run `npm run build` or `yarn build`). Enabling service worker in a development mode is not a recommended practice, because it can lead to the situation when previously cached assets are used and the latest local changes are not included.

Instead, in the development mode the `noopServiceWorker.js` is included. This service worker file is effectively a 'no-op' that will reset any previous service worker registered for the same host:port combination.

If you need to test a service worker locally, build the application and run a simple HTTP-server from your build directory. It's recommended to use a browser incognito window to avoid complications with your browser cache.

Configuration

PWAs today are **offline-first**

Home

What's happening?

Tweet

Looks like you lost your connection. Please check it and try again.

Retry

Search Twitter

Looks like you lost your connection. Please check it and try again.

Retry

OneDrive

Search everything

Ire Aderinokun

New

Sort

My files

Recent

Photos

Shared

Recycle bin

Something went wrong

Something went wrong

Please try again or refresh the page.

Go to my OneDrive

Get 1TB (1000 GB) of storage and the latest Office apps for PC or Mac. Learn more.

Go premium

3.7 MB used of 5 GB

Get the OneDrive apps
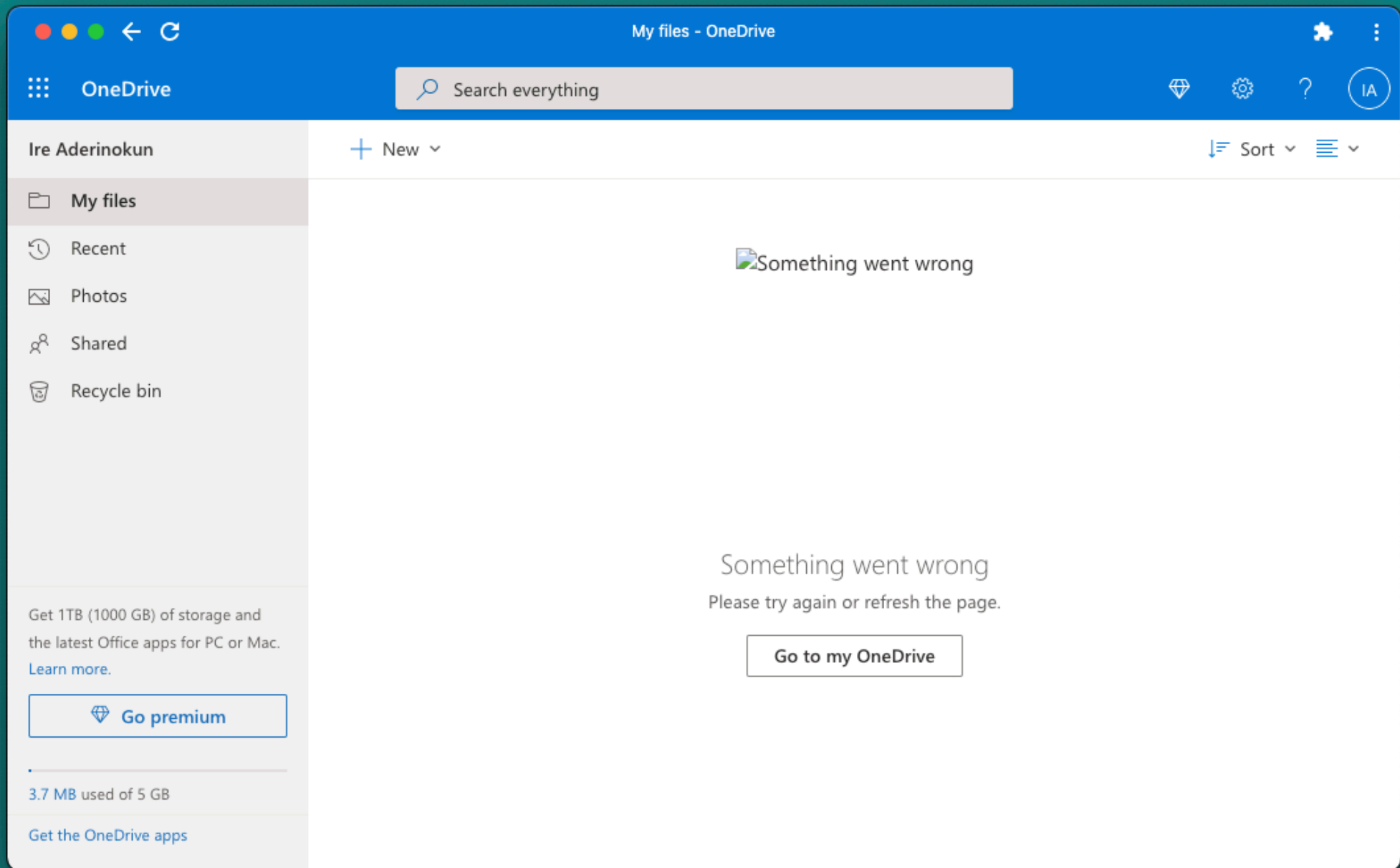
bitsofcode

😱 You're Offline!

Looks like you're not connected to the internet! No worries though, if you've saved any articles before, you'll see them listed below.

1 offline articles

- Your first performance budget with Lighthouse
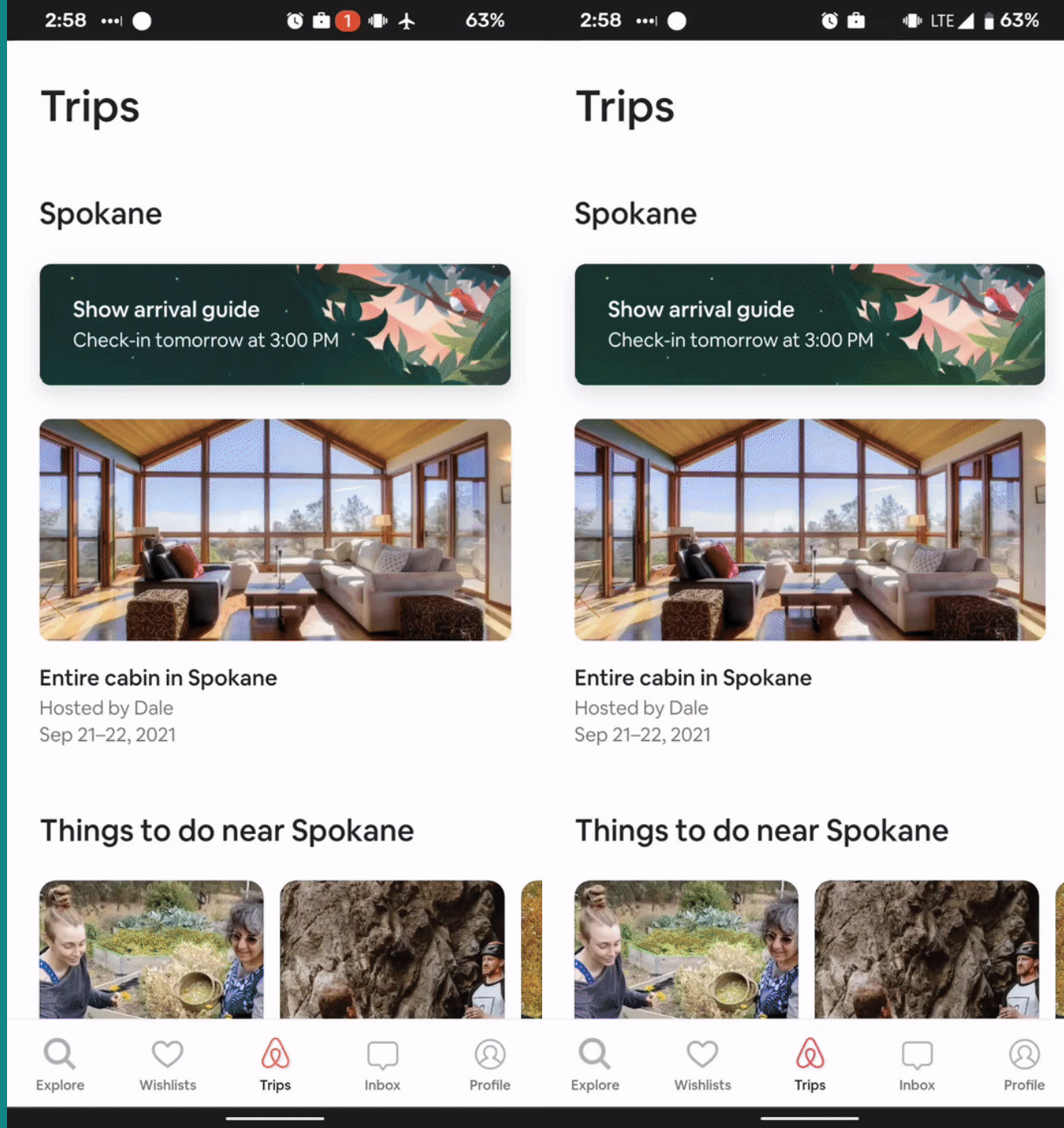
Offline
Online

Creating Delightful User Experiences with PWAs

7:10 PM - 7:30 PM

Progressive Web Apps open up the ability to create moments of delight for our users that we otherwise couldn't do with regular websites. Learn how Airbnb's done just that to provide a high quality installation experience, seamless transitions, and an offline arrival guide across a multi-year, distributed effort.

Callie Riggins

Storage APIs

IndexedDB

Local Storage & Session Storage

Cache Storage

Securing PWA Authentication Offline

5:30 PM - 5:50 PM

Did you know you can secure your PWA and still support offline functionality? Learn how an app to help African fish farms authenticates users while still allowing them to work offline, in remote areas, and sync changes in the background, and how you can apply these lessons to your PWAs.

Jeroen Kocyigit

PWAs today are **fast**

APP SHELL INSTANT LOADING

Image from https://www.youtube.com/watch?v=QhUzmR8eZAo
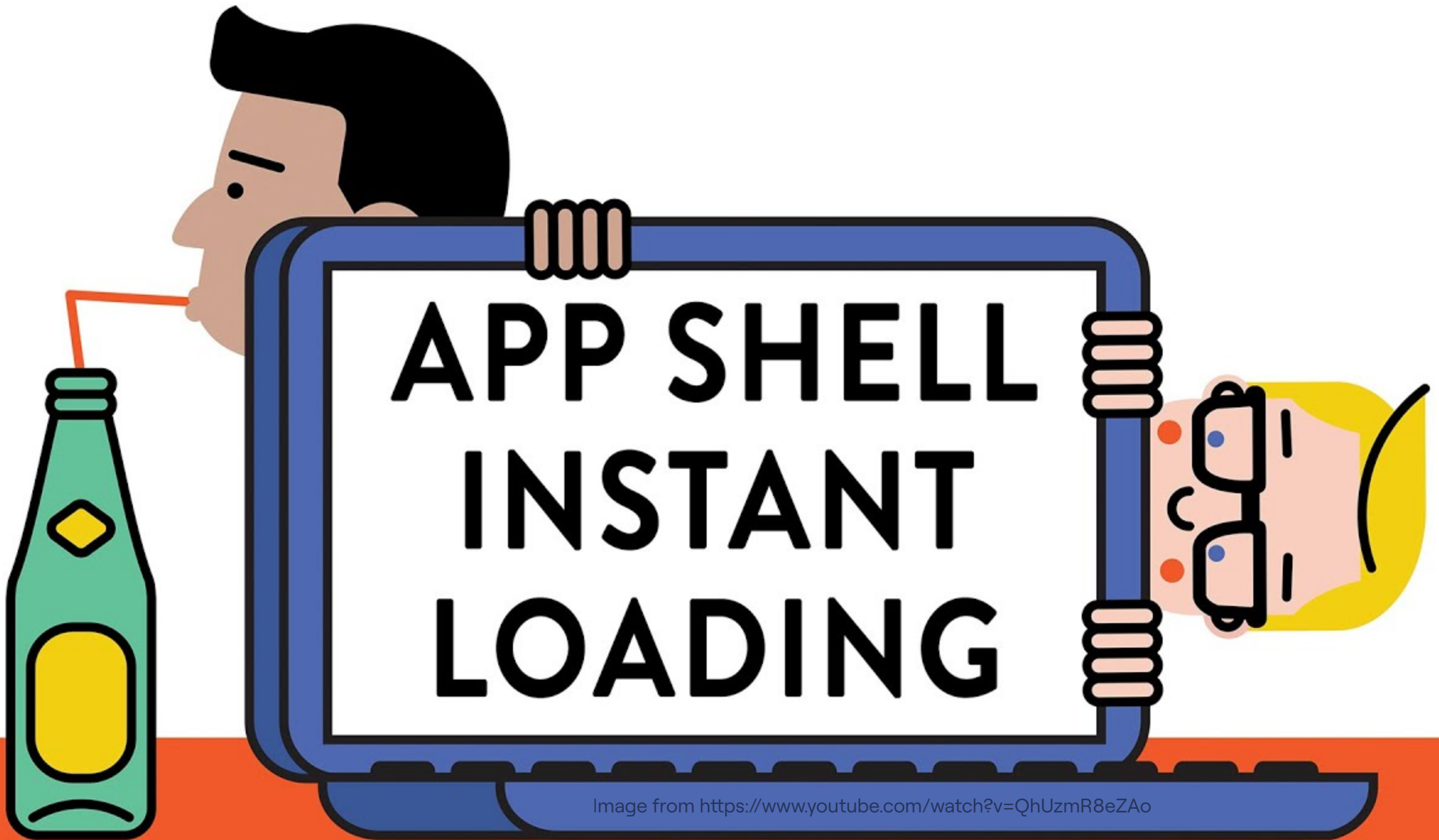
The app "shell" is the **minimal HTML, CSS and JavaScript required to power the user interface** and when cached offline can ensure instant, reliably good performance to users on repeat visits. This means the application shell is not loaded from the network every time the user visits. Only the necessary content is needed from the network.

https://developers.google.com/web/fundamentals/architecture/app-shell

Search Twitter

🔍 twi - Google Search

🕐 twitter app

🕐 twitter app vs website

🐦 Ire Aderinokun (@ireaderinokun) / Twitter - twitter.com/ireaderinokun

🐦 Notifications / Twitter - twitter.com/notifications

🐦 Home / Twitter - twitter.com/home

　　📑 Switch to this tab

🌐 Akinleye Oladipupo (@IREOLUWA) / Twitter - twitter.com/ireoluwa

My assignme…

https

Notion – The …

Amazon.co.uk

Google Calen…

PWA in 2021

Sign in to Ban…

buycoinsafric…

To

Add shortcut

After launching Twitter Lite in 2017, the company saw a —

65% increase in pages per session

75% increase in Tweets sent

20% decrease in bounce rate

https://developers.google.com/web/showcase/2017/twitter

Twitter Lite is now the fastest, least expensive, and most reliable way to use Twitter.

The web app rivals the performance of our native apps but requires less than 3% of the device storage *space compared to Twitter for Android.*

— Nicolas Gallagher, Engineering Lead for Twitter Lite

https://developers.google.com/web/showcase/2017/twitter

PWA SUMMIT 2021
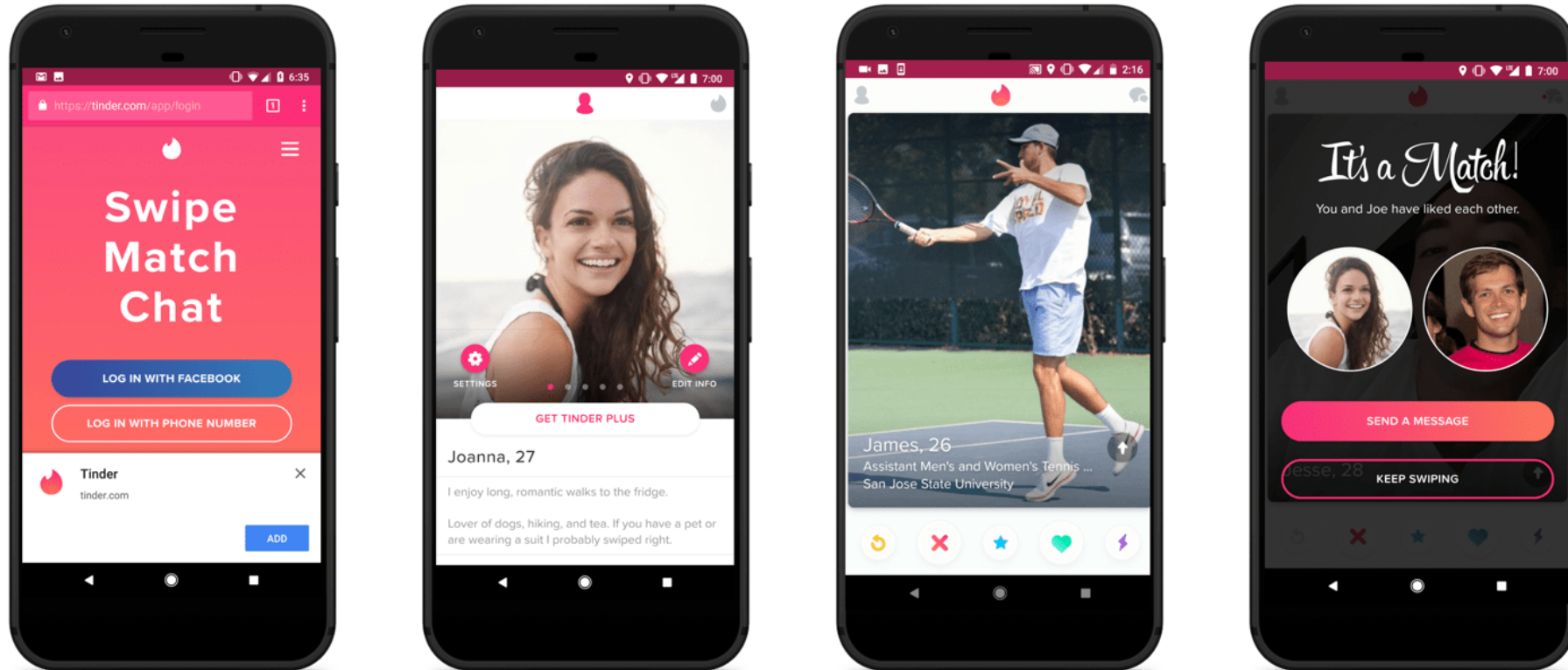
Size: Comparing the PWA to the native apps

Image from https://medium.com/@addyosmani/a-tinder-progressive-web-app-performance-case-study-78919d98ece0
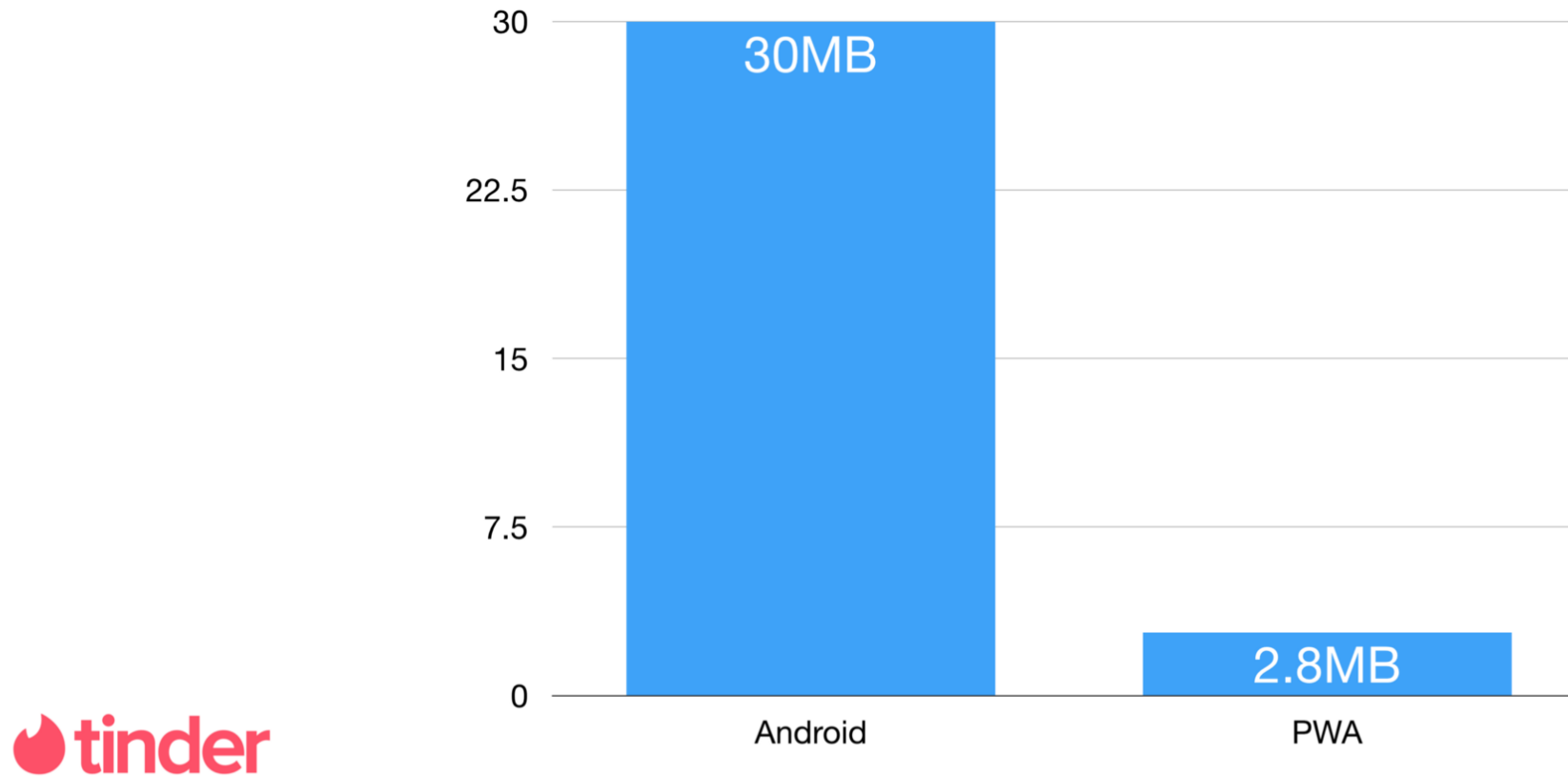
After launching its PWA, **Thomas Kent** lowered its bounce rate by **57%** and increased the revenue from organic traffic by **79%**.

#E-commerce   #Revenue   #Performance

Butcher of Blue's PWA increased mobile users by **154%**, monthly active users by **154%**, and conversion by **169%**. Pages load **85%** faster.

#Conversions   #E-commerce   #Fashion
#Performance

Kubota launched their e-commerce PWA which resulted in a **192%** growth of daily visitors and **26%** growth of average monthly visits.

#Conversions   #E-commerce   #Fashion
#Performance

Fashion brand **Rooted Objects** saw an increase in conversions of **162%** after launching their PWA. Page load times decreased by **25%** and average redirection time decreased by **80%**.

#Conversions   #E-commerce   #Fashion
#Performance   #Revenue

Tajawal and Almosafer saw an increase of **3×** in conversion rate from **0.3%** to almost **1.2%** within two weeks. Page load time decreased by **3×** from **13s** to **3.6s** for time to interactive.

#Conversions   #Engagement   #Offline   #Performance
#Service Worker   #Travel

Debenhams found that the user journey from browsing to purchase was **2×** to **4×** on its PWA. The PWA delivered a **40%** increase in mobile revenue and a **20%** increase in conversion.

#Conversions   #E-commerce   #Performance
#Revenue   #Service Worker

Commonfloor groups saw **2×** increase in daily active users and a **10%** increase in overall traffic by moving to a PWA with app shell.

#App Shell   #E-commerce   #Performance   #Social

Streamling platform **ZEE5** launched a PWA to expand its reach. The PWA is **3×** faster and reduced buffering time **50%**.

#E-commerce   #Media   #Performance
#Service Worker

The **Starbucks** PWA has increased daily active users **2×**. Orders on desktop are nearly the same rate as mobile.

#Conversions   #E-commerce   #Performance
#Service Worker

UK clothing brand **George** saw a **31%** increase in conversion, **2×** lower bounce rate, and **20%** increase in page views after upgrading to a PWA. The PWA was **3.8×** faster than the previous site.

#Conversions   #E-commerce   #Offline   #Performance
#Service Worker

Treebo launched a PWA and saw a **4×** increase in conversion rate year-over-year. Conversion rates for repeat users saw a **3×** increase and their median interactive time on mobile dropped to **1.5s**.

#Conversions   #Offline   #Performance
#Service Worker   #Travel

OpenSooq's new PWA is only **28.3KB** and has improved engagement with a **25%** increase in average time on page resulting in **260%** more leads.

#App Shell   #Classified   #Engagement
#Home Screen   #Notifications   #Service Worker

https://www.pwastats.com/

Performant Progressive Enhancement for PWAs

4:50 PM - 5:10 PM

When every byte matters and you need to support every user and every device, you need to embrace progressive enhancement from the service worker up. Supporting users who may not support Progressive Web Apps, on low-end devices, on expensive pre-paid internet plans, or while offline, requires rethinking of the core architecture of most PWAs. Performant progressive enhancement provides an approach that a research team at Illinois Institute of Technology in Chicago, in partnership with Howard Brown Health, has developed for these needs, allowing them, and you, to improve PWA user engagement for everyone

Karl Stolley

Kinza Haque

Steve N. Du Bois

PWAs today are **accessible**

The web **wins** at accessibility 🥇

"

*Looking back over one full year since we started rebuilding our mobile web, we're so proud of the experience we've created for our users. Not only is it significantly faster, **it's also our first platform to support right-to-left languages** and "night mode". Investing in a full-featured PWA has exceeded our expectations. And we're just getting started*

— Zack Argyle, Engineering Manager for Pinterest Core Experience

https://medium.com/pinterest-engineering/a-one-year-pwa-retrospective-f4a2f4129e05

Possible With Accessibility?

6:50 PM - 7:10 PM

Are PWAs possible with accessibility? The answer is a resounding *yes*, but it doesn't happen by accident. Accessibility is something you need to factor into everything, from your icons to your manifest to your PWA itself. Learn what you need to do to make your whole PWA experience accessible, from load to install and beyond.

Léonie Watson

How reliable are PWAs today?

✅ Service worker, and it's tooling, are well supported

✅ PWAs are faster, more resilient, and more accessible

How reliable are PWAs today?

😄

✅ Service worker, and it's tooling, are ~~supported~~

✅ PWAs are faster, more resilient, and more accessible

How installable are PWAs today?

Add to home screen (A2HS)

The ability for a user to "install" a website and use it as if it was a natively installed app. To enable this behaviour, a website must serve a valid Web App Manifest and load it's assets through a [Service Worker] (https://caniuse.com/#feat=serviceworkers).

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	90	89	90	13.1	13.7			4.4	12.0
10	91	90	91	14	14.4			4.4.4	13.0
11	92	91	92	14.1	14.7	all	92	92	14.0
		92	93	15					

✓   ✗   Partial Support

Global: 73.94% + 13.74% = 87.68%
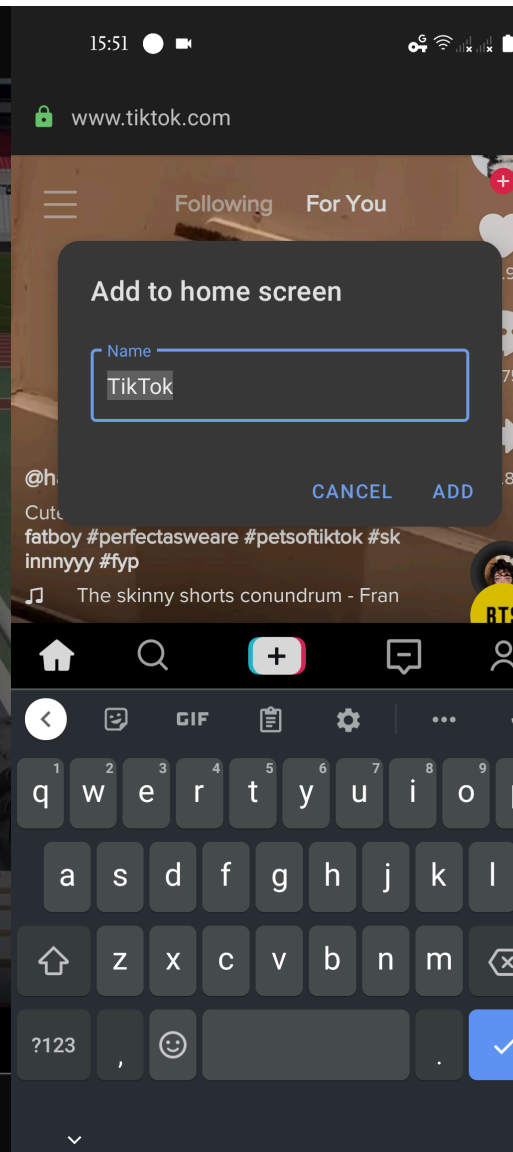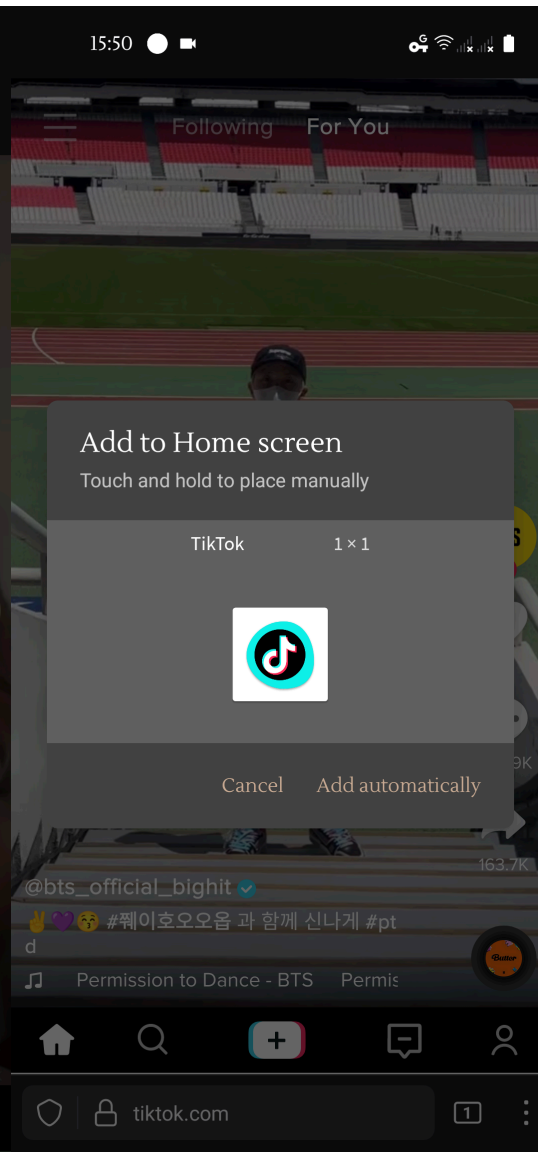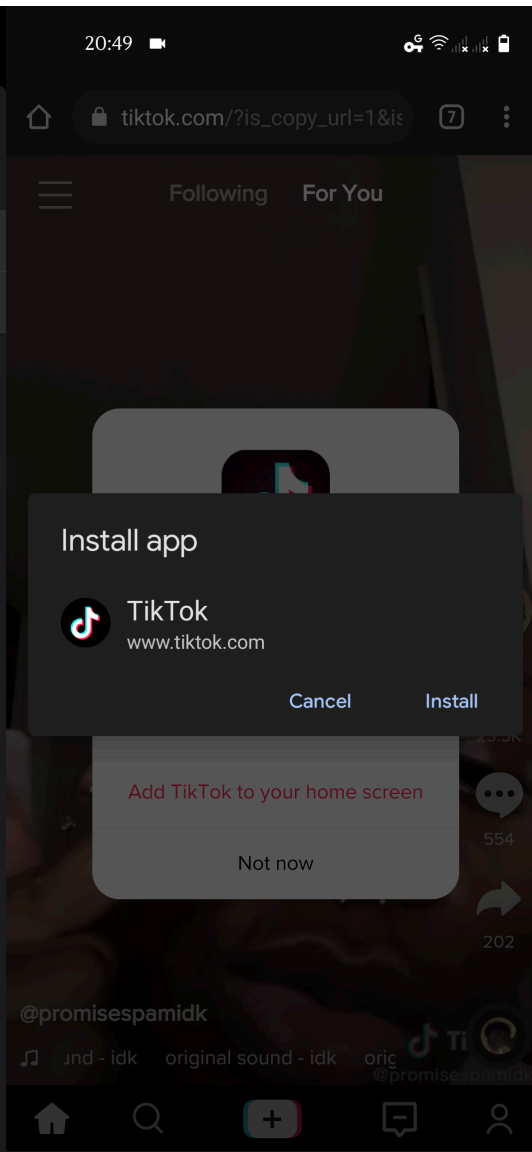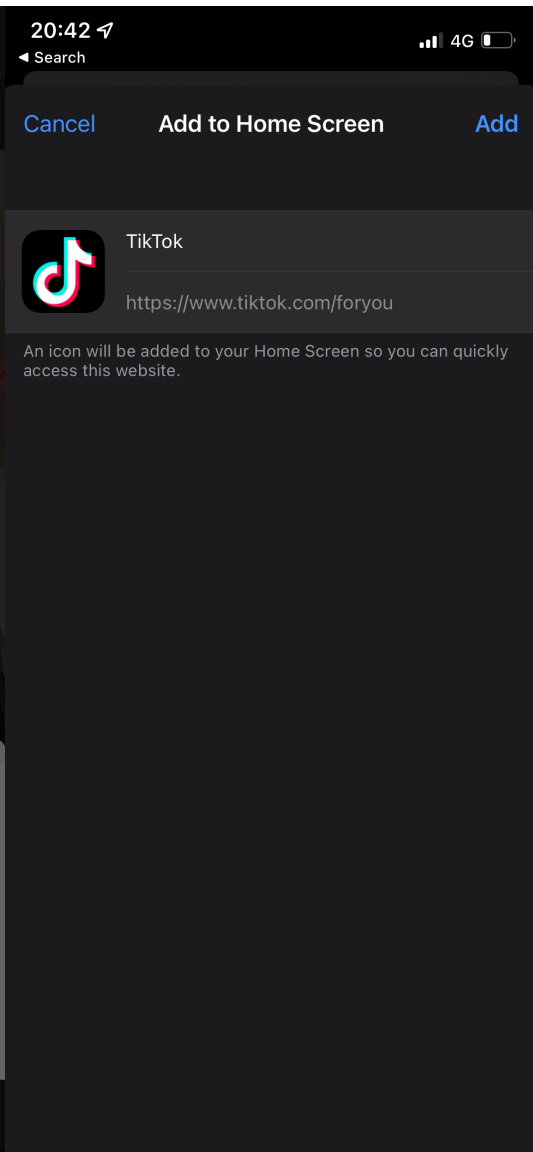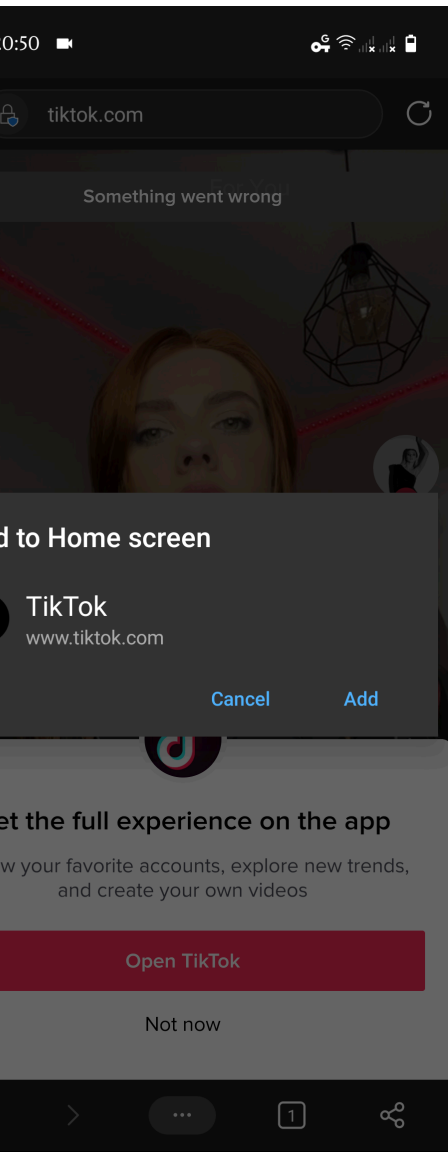
Requirements for "Add to Home Screen"

Requirements for "Add to Home Screen"

1. Web app manifest

App Manifest

manifest.json

Identity

Name	Twitter
Short name	Twitter
Description	Get breaking news, politics, trending music, world events, sports scores and the latest global news stories as they unfold – all with less data.

Presentation

Start URL	https://twitter.com/?utm_source=homescreen&utm_medium=shortcut
Theme color	☐ #ffffff
Background color	☐ #ffffff
Orientation	
Display	standalone

Icons

☐ Show only the minimum safe area for maskable icons

Need help? Read documentation on maskable icons.

Primary icon as used by

Chrome
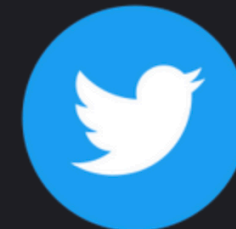
192×192px

image/png

Requirements for "Add to Home Screen"
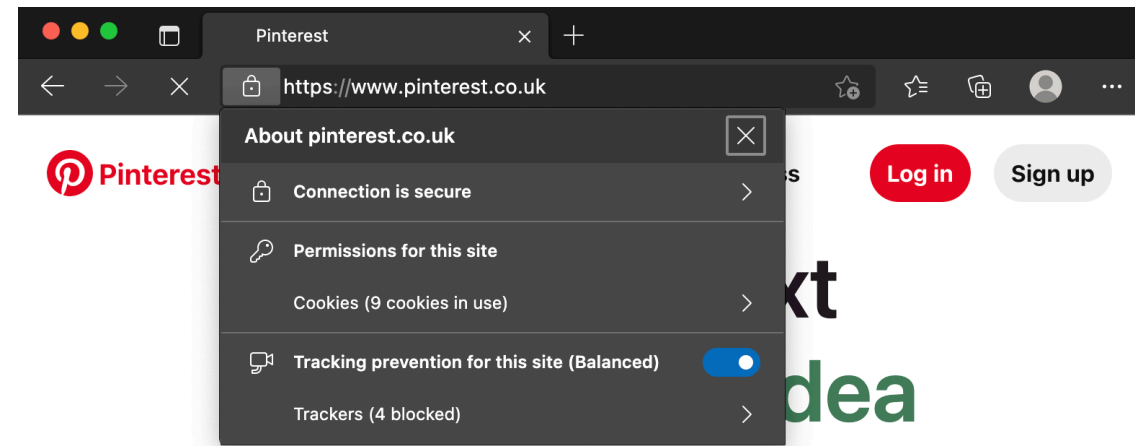
1. Web app manifest

2. HTTPS

Requirements for "Add to Home Screen"

1. Web app manifest
2. HTTPS
3. Service worker (which returns a 200 response when offline)

Page

ServiceWorker

Network

Cache
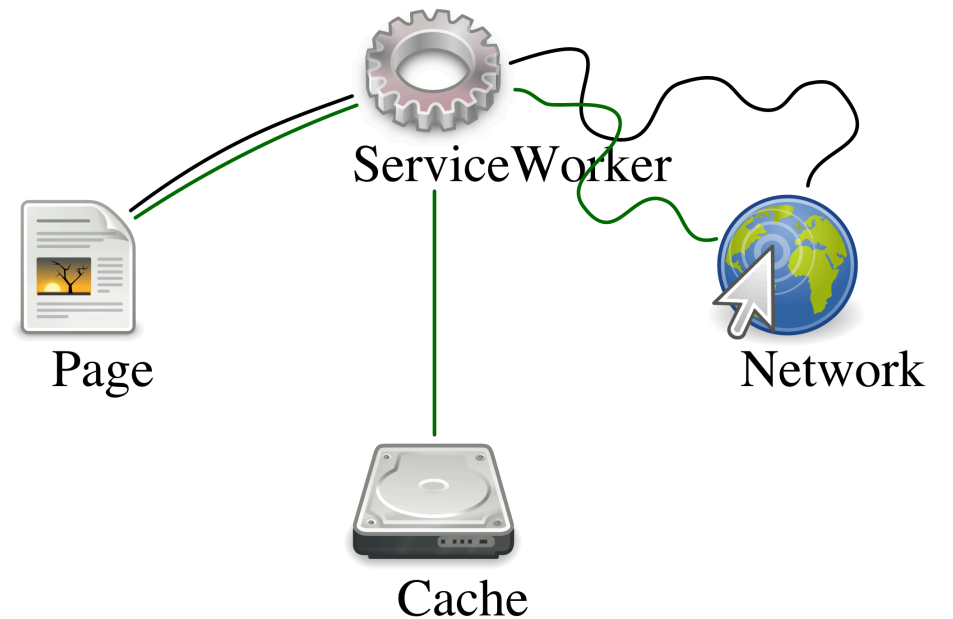
Requirements for "Add to Home Screen"

1. Web app manifest

2. HTTPS

3. Service worker (which returns a 200 response when offline)

4. ~~User engagement heuristic~~

More control over the **install flow**

Detect if a PWA can be installed

```
window.addEventListener('beforeinstallprompt', (e) ⇒ {


});
```

Save install prompt for later

```
let installPrompt;

window.addEventListener('beforeinstallprompt', (e) ⇒ {
    e.preventDefault();
    installPrompt = e;
});
```

Detect if a PWA has been installed

```
window.addEventListener('appinstalled', (e) ⇒ {


});
```
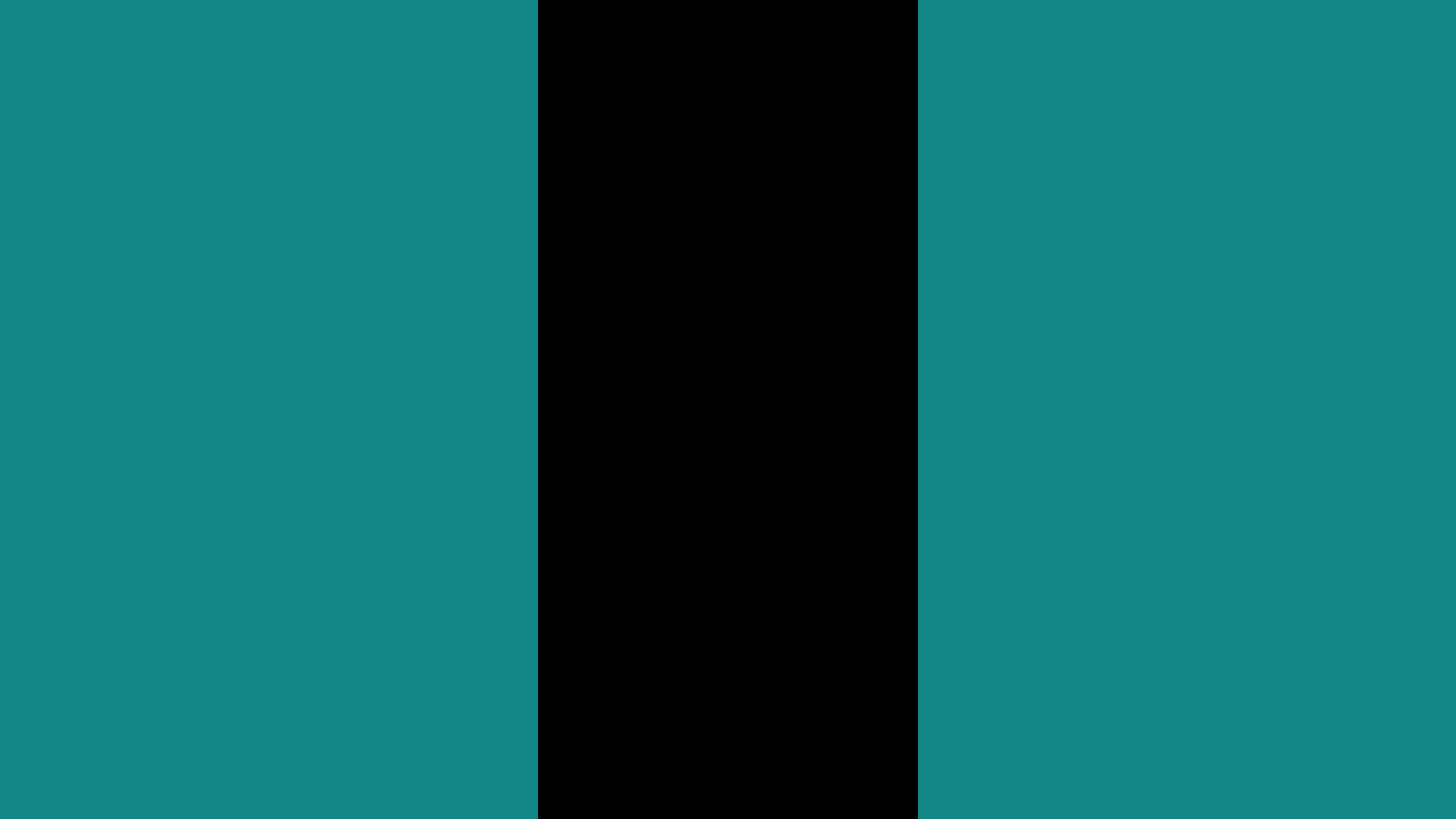
More control over the **launch experience**

Twitter

Telegram WebZ

Instagram

Tinder

Splash Screen

```
{

    "name": "...",

    "background_color": "...",

    "icons": ["sizes": "512x512", "type": "image/png", "src": "..."],

}
```
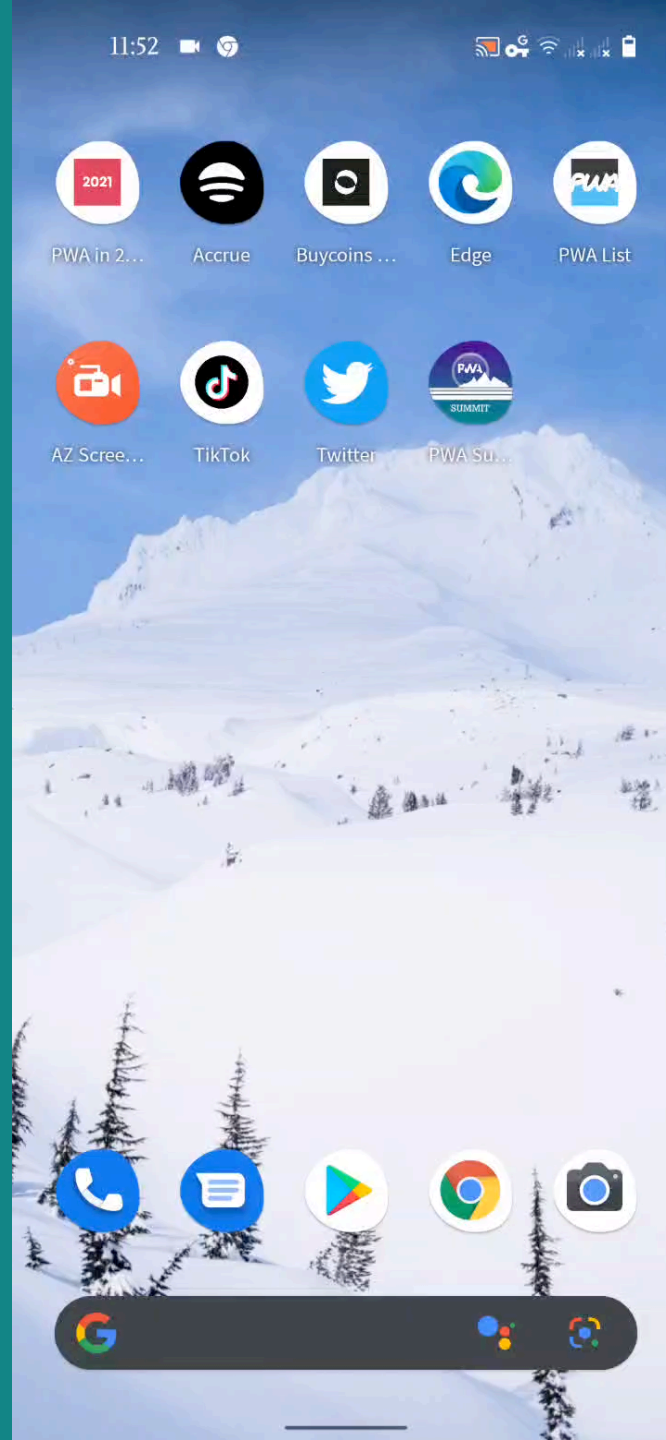
App Shortcuts

```
{

    "shortcuts": [{

        "name": "New Tweet",

        "url": "/compose/tweet",

        "icons": [{ /* ... */}]

    }]

}
```

Entanglement Web A...

Web Store

Slides

Docs

Google Drive

Gmail

YouTube

Desktop, formerly Dr...

bitsofcode

Squoosh

SSA

Twitter

TikTok
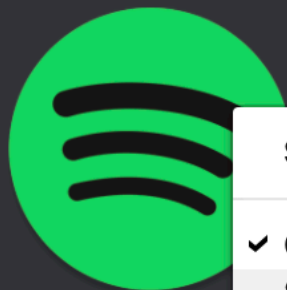
Spotify

Spotify

✓ Open as a Window

Start app when you sign in

Options

Remove from Chrome...

View in Web Store

Create Shortcut

Run on OS Login

```
navigator.runOnOsLogin.set({ mode: "windowed" }).then(() => {

    // Permission approved

});
```

Make your PWAs Look and Launch Beautifully

6:30 PM - 6:50 PM

Building beautiful-looking icons and splash screen images for your PWA is an important aspect of the launch experience for your users. Discover the current, and upcoming, web standards to make for beautiful launch experiences, how to make them great on different platforms, and automation approaches to help you on your way toward creating stunning launch experiences for your PWAs.

Önder Ceylan

Add to Homescreen == Install

WebAPK

A WebAPK is an Android Application Package (APK) **automatically generated from a PWA** and installed to the device.

Benefits of WebAPK

Benefits of WebAPK
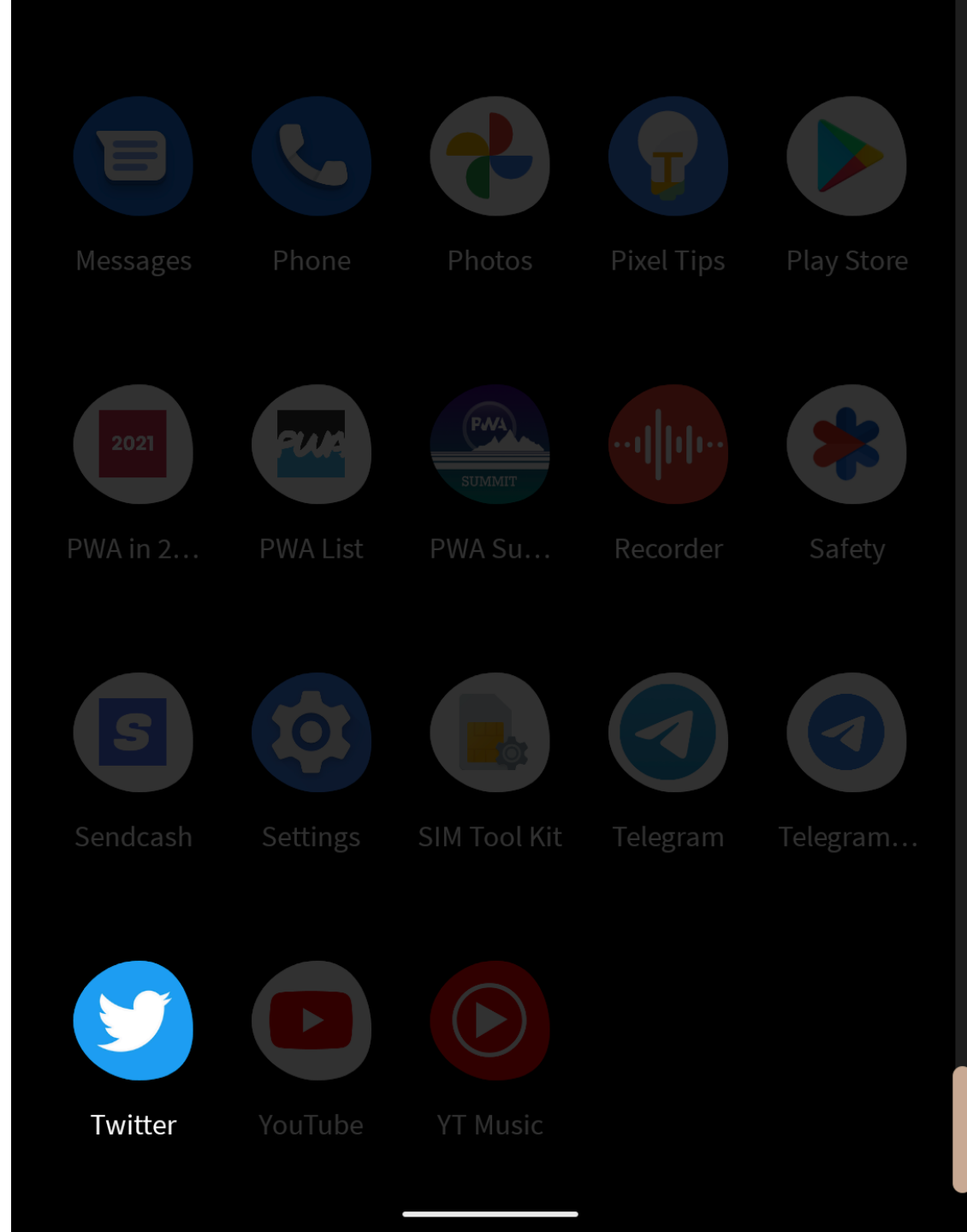
✅ In the app drawer

Benefits of WebAPK

✅ In the app drawer

✅ In app settings

App info

Twitter

Open   Uninstall   Force stop

Notifications
~0 notifications per week

Permissions
No permissions requested

Storage and cache
297 kB used in internal storage
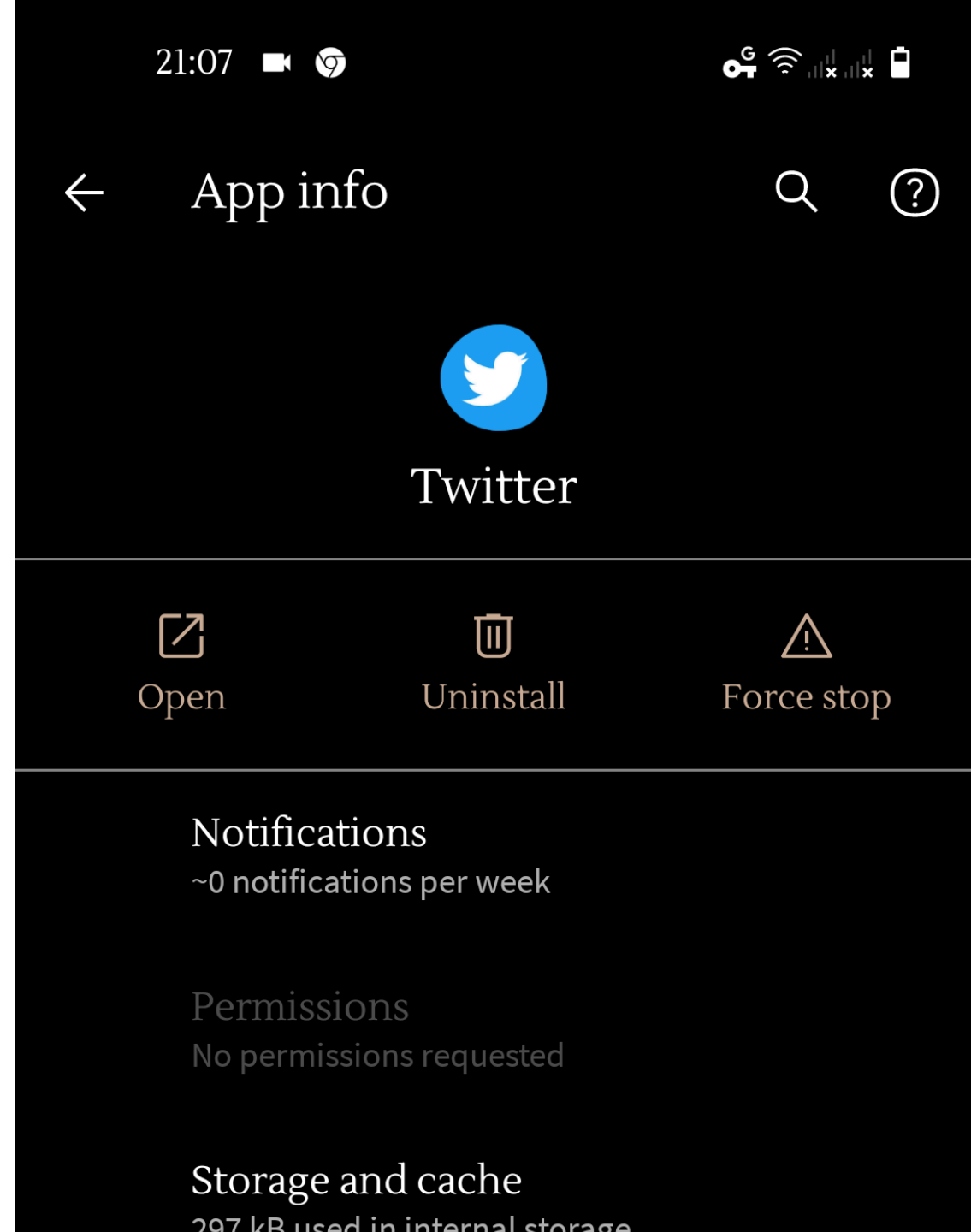
Benefits of WebAPK

✅ In the app drawer

✅ In app settings

✅ Includes intent filters

```
"start_url": "/",
"display": "standalone",
```

```
<intent-filter>
  <action android:name="android.intent.action.VIEW" />
  <category android:name="android.intent.category.DEFAULT" />
  <category android:name="android.intent.category.BROWSABLE" />
  <data
    android:scheme="https"
    android:host="example.com"
    android:pathPrefix="/" />
</intent-filter>
```

Ship your PWA to the app stores at lightning speed.

Test

PWABuilder will make sure your web app is a PWA and ready for the stores!

Manage

Our Report Card will let you know if your PWA is store-ready. If not, PWABuilder will help you get there!

Package

Once you are ready, PWABuilder can package your PWA for the app stores in minutes!

Explore

PWAs are moving forward fast, learn about new web APIs, store readiness, and more!

Enter the URL to your PWA   Start

Resour

https://www.pwabuilder.com/
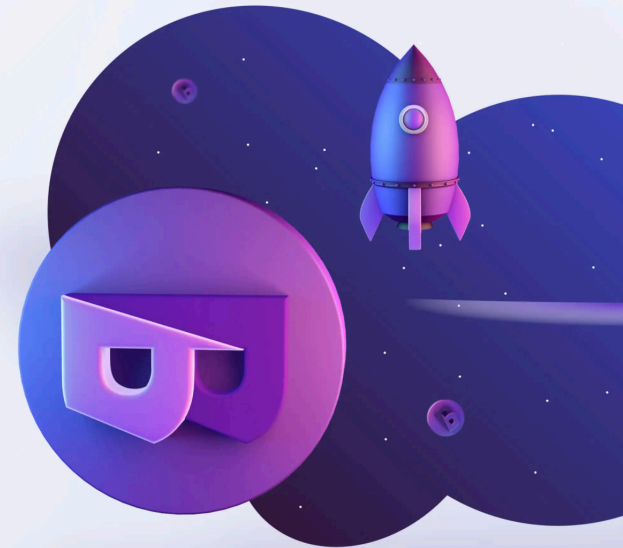
Windows

Publish your PWA to the Microsoft Store to make it available to the 1 billion Windows users worldwide.

Store Package

Test Package

Android

Publish your PWA to the Google Play Store to make your app more discoverable for Android users.

Store Package

SAMSUNG

Samsung

Provide the URL to your PWA to Samsung for inclusion in the Samsung Finder app. You will need to follow up with Samsung after submission for updates on the deployment.

Submit

https://www.pwabuilder.com/

Bringing your PWA to App Stores

8:20 PM - 8:40 PM

App stores are an important distribution channel for Progressive Web Apps; they provide exposure alongside the other apps users are searching for, and have built-in ways to manage purchases and subscriptions. In this talk, BeFunky will share their experience launching their PWA into the Play store, from bundling it for release to integrating with Play payments with the Digital Goods and Payment Request APIs, all while sharing what they've learned and what getting into an app store has done for their product.

Micah Engle-
Eshleman

PWA → TWA

Trusted Web Activity is a way to **open your PWA from your Android app** using a protocol based on Custom Tabs

https://developers.google.com/web/android/trusted-web-activity

main ▾   **bubblewrap** / README.md   Go to file   ...

andreban Remove mention to slack channel (#432) ... ✓   Latest commit 95eecbc on 5 Jan   🕐 History

👥 **6 contributors**

68 lines (49 sloc) | 2.92 KB   Raw   Blame   🖥   ✏   🗑

Bubblewrap

🐙 Node CI passing

Bubblewrap is a set of tools and libraries designed to help developers to create, build and update projects for Android Applications that launch Progressive Web App (PWA) using Trusted Web Activity (TWA).

Requirements

https://github.com/GoogleChromeLabs/bubblewrap

PWAs Today

Native Apps

"Given the overlap in terms of capabilities, we often get asked about the recommended approach: PWA or UWP (Universal Windows Platform). **We see this as a false dichotomy!** … the choice between PWA and native should be evaluated on a case-by-case basis."

https://blogs.windows.com/msedgedev/2018/02/06/welcoming-progressive-web-apps-edge-windows-10/#Vsxe9rFMkrMQbXiJ.97

"Given the overlap in terms of capabilities, we often get asked about the recommended approach: PWA or UWP (Universal Windows Platform). We see this as a false dichotomy! ... **the choice between PWA and native should be evaluated on a case-by-case basis**."

https://blogs.windows.com/msedgedev/2018/02/06/welcoming-progressive-web-apps-edge-windows-10/#Vsxe9rFMkrMQbXiJ.97

How installable are PWAs today?

✅ "Add to Homescreen" is well supported

✅ More control over the install experience

✅ Deeper integration with the device

❌ Platform parity

How installable are PWAs today?

✅ "Add to Homescreen" is well suppo 😁

✅ More control over the install experi

✅ Deeper integration with the device

❌ Platform parity

PWAs Tomorrow
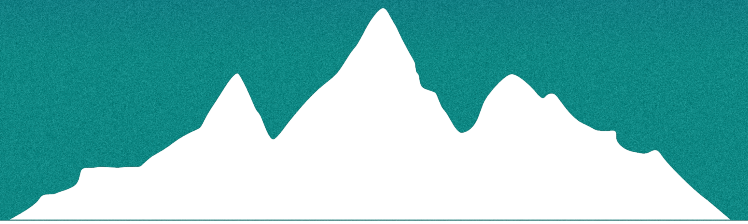
🤩 Capable

🤩 Reliable

🤩 Installable
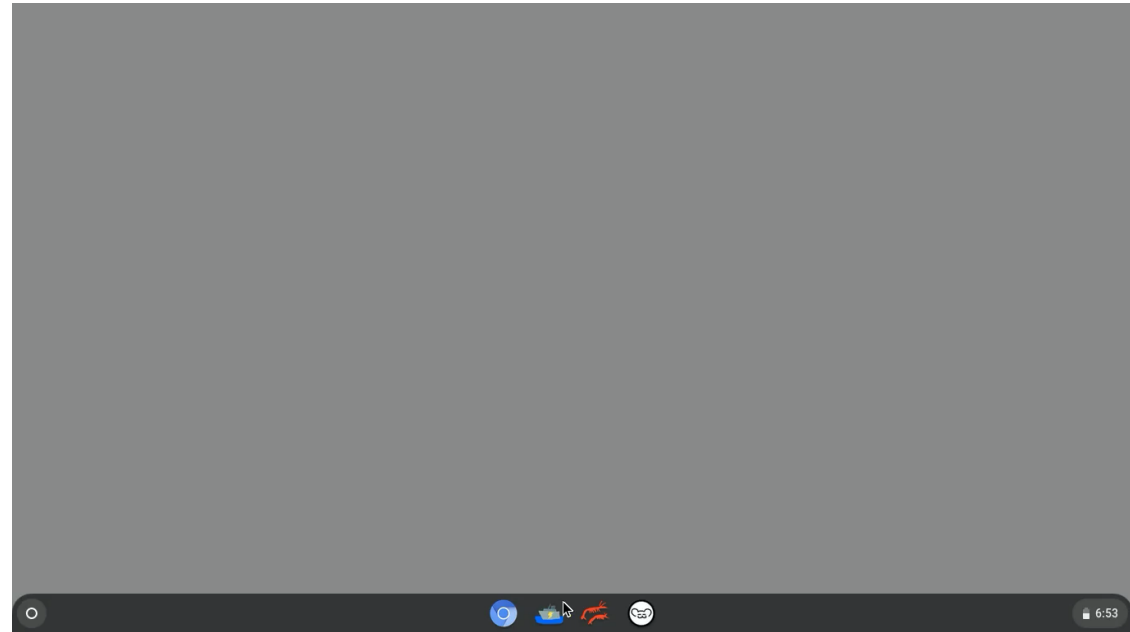
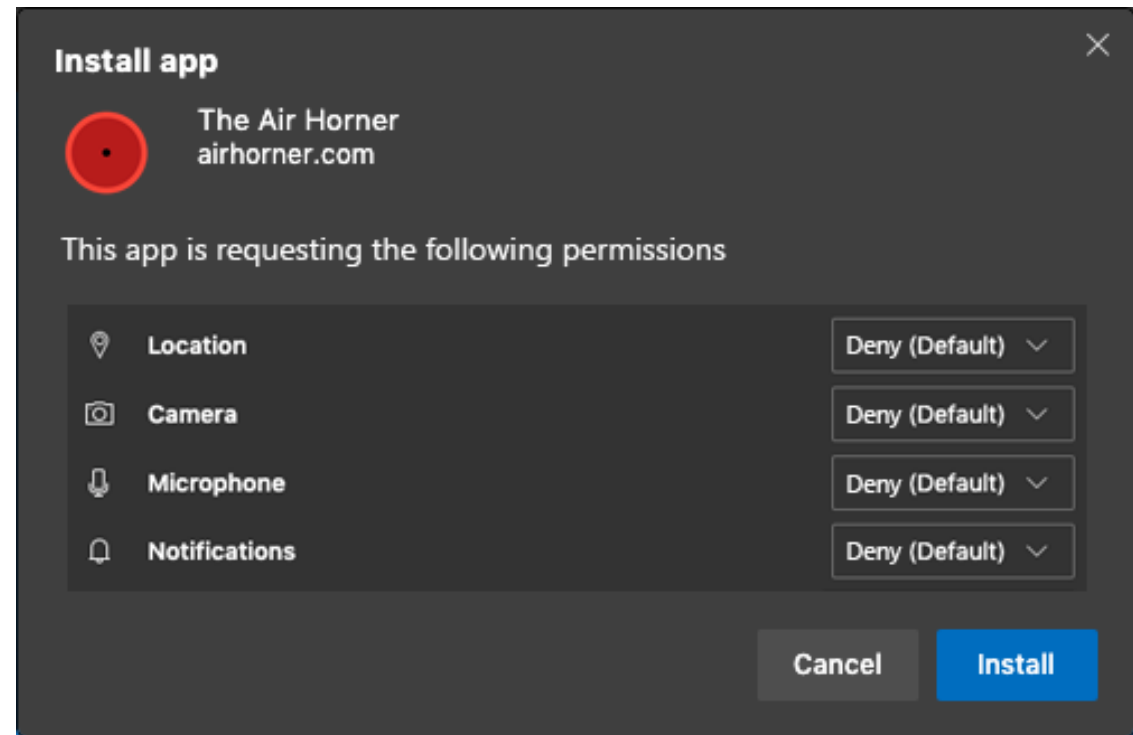# All the APIs!

File Handling API
(Origin Trial)

https://github.com/WICG/file-handling/blob/main/explainer.md
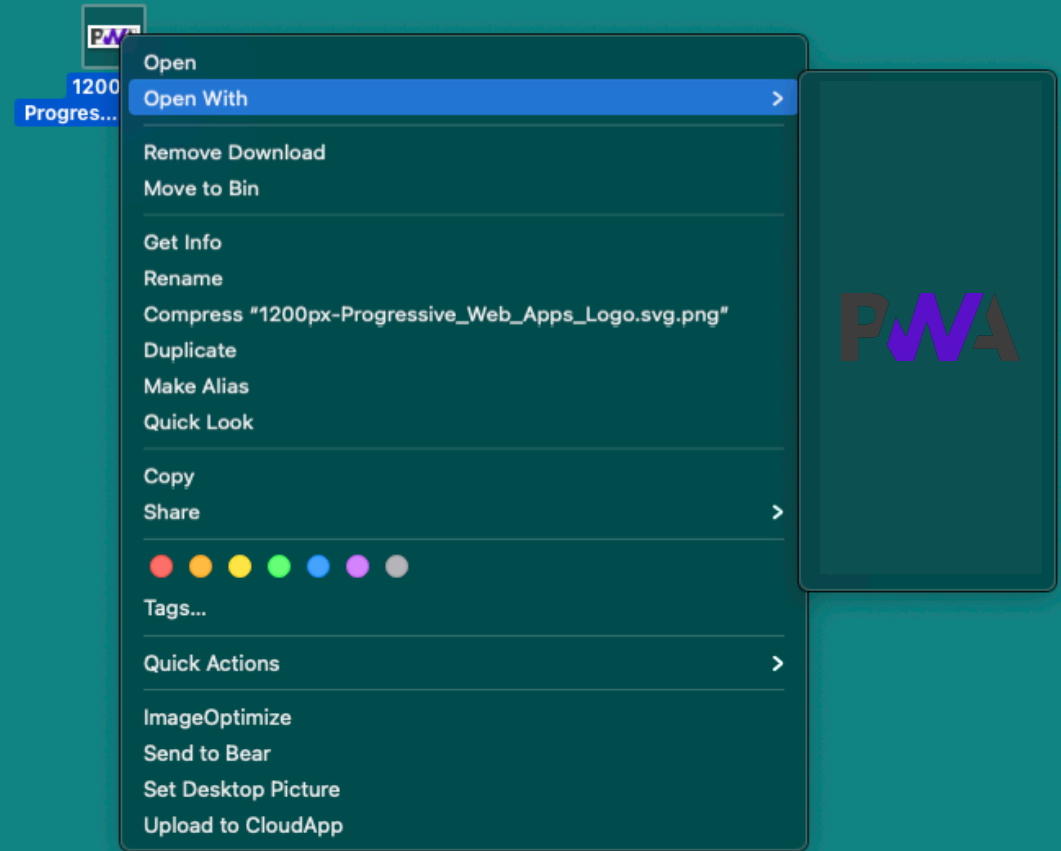
Tabbed Application Mode
(Developer Trial)

https://web.dev/tabbed-application-mode/

Install-Time Permissions
(Under Consideration)

https://github.com/MicrosoftEdge/MSEdgeExplainers/blob/main/
InstallTimePermissionsPrompt/Explainer.md

Honourable mentions

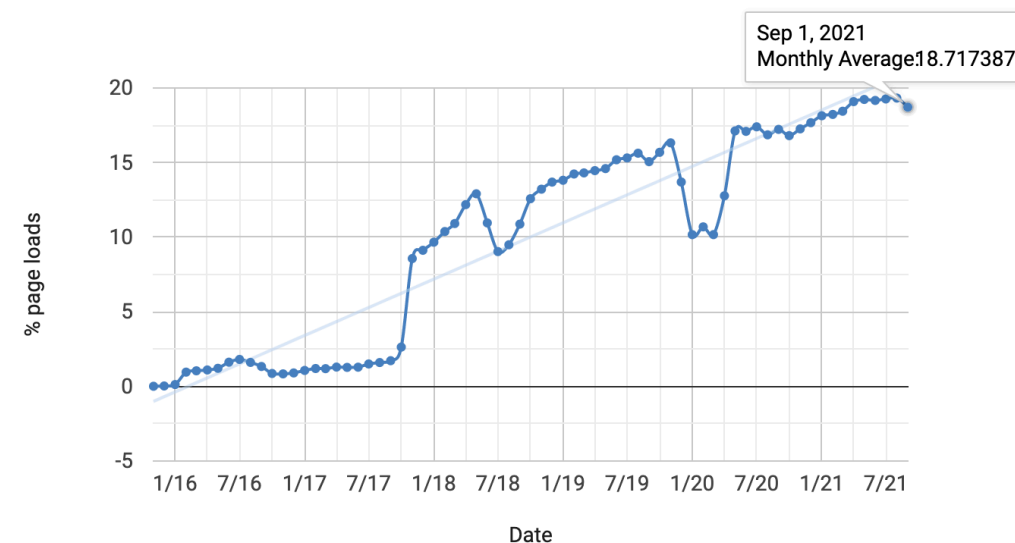Offline-only PWAs

Detecting orientation change events
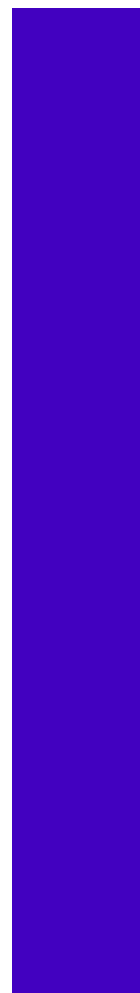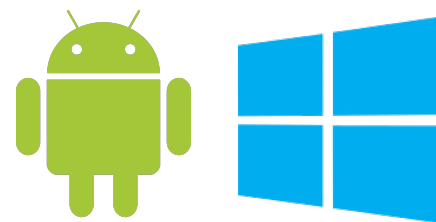
Widgets

Detect/block screenshots

Splash screen API

Local font access

Only

~~Alm~~ost **20% of page loads** are controlled by Service Workers

Sep 1, 2021
Monthly Average: 18.717387

% page loads

20

15

10

5

0

-5

1/16   7/16   1/17   7/17   1/18   7/18   1/19   7/19   1/20   7/20   1/21   7/21
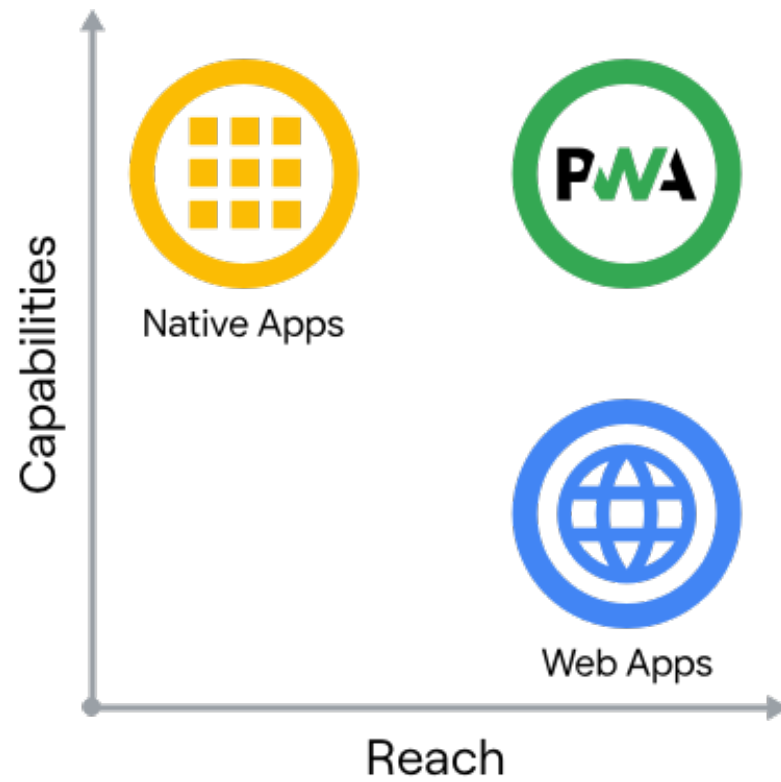
Date

Image from https://web.dev/what-are-pwas

Web on Everything!

Future of Web Apps Panel

10:00 PM - 11:00 PM

A live, moderated panel where industry experts from Google, Microsoft, Intel, Samsung, and the W3C will discuss the future of web apps and take questions from attendees.

Thank you!

Ire Aderinokun 🇳🇬

COO & VP Engineering of Buycoins

Google Web Expert

ireaderinokun.com

bitsofco.de

@ireaderinokun