

The background image shows a person's hands holding a smartphone, with a laptop visible in the background. The scene is overlaid with a complex digital network of lines and nodes, along with various floating numbers and percentages, suggesting a high-tech or data-driven environment.

eficode

Progressive Web Apps.

Introduction to progressive web applications.

EFICODE.COM

Niki Ahlskog

Insert cool title here!

- 5 years of ICT working experience
- Bachelor's degree in Information Technology - 2015
- Currently working on Master's degree at Helsinki University



@shnigi_



nikiahlskog



@shnigi



shnigi



eficode

EFICODE.COM

Eficode in a Nutshell




Design



Create



Analyse



Automate

Eficode is a design and technology house with a vision to build the future of software development. Our mission is to guide our customers to the next level with always evolving experts, create world-class solutions to achieve it and make product creation fun again!

200+
PROFESSIONALS

PIONEERING IN
DEVOPS SINCE
2007

YEARLY
PROJECTS
500+

+16M€
TURNOVER
2015-2016

CUSTOMER
SATISFACTION
97%

HELSINKI
TAMPERE
COPENHAGEN
STOCKHOLM
BEIJING

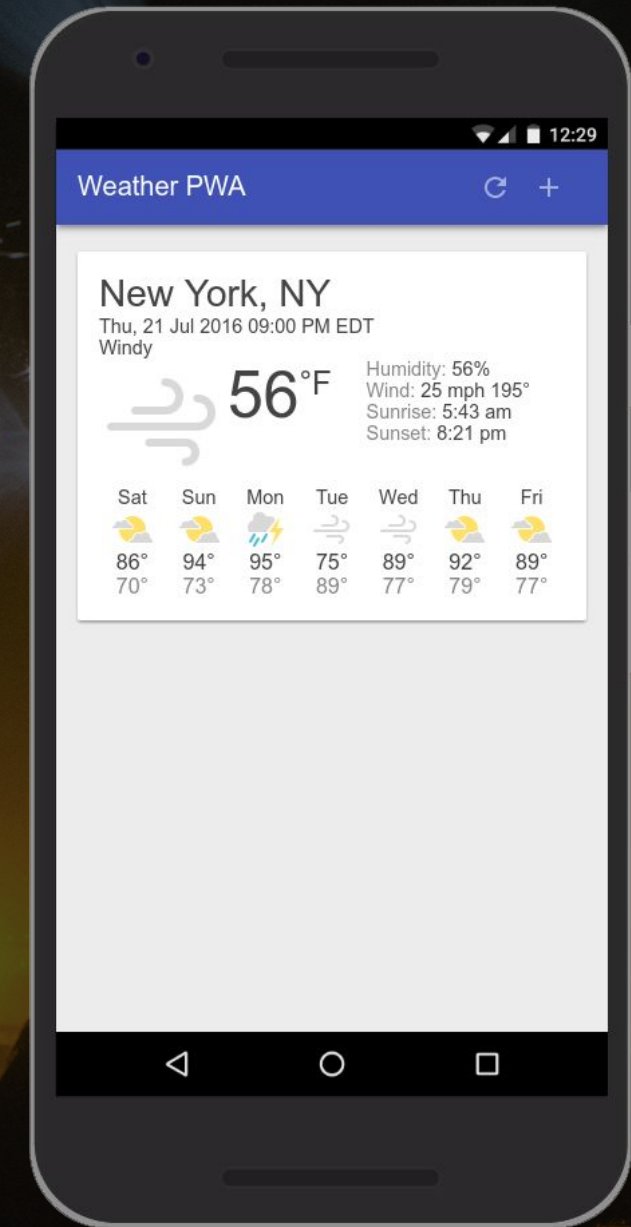
Introduction

A Progressive Web App uses 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.

Ask yourself: How many applications have you installed during past month?

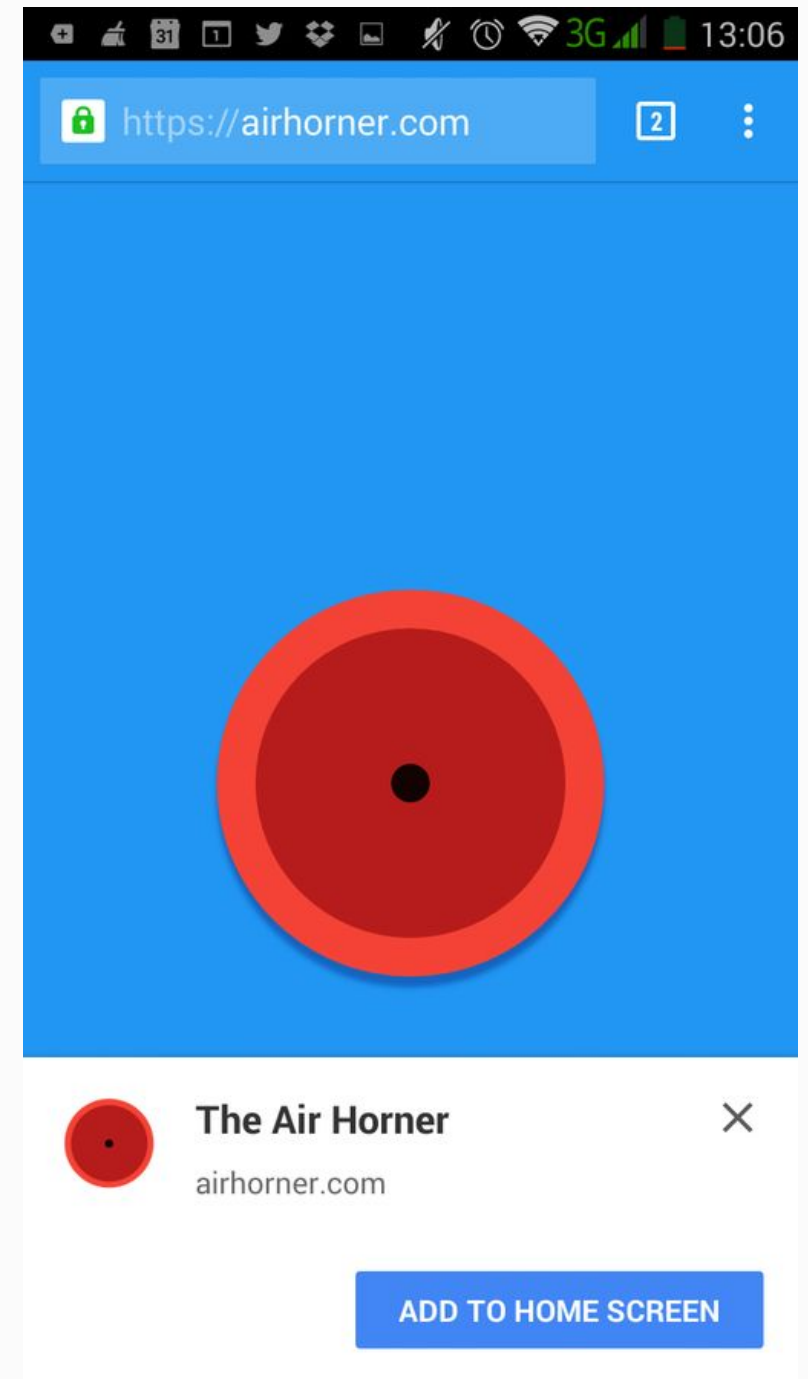
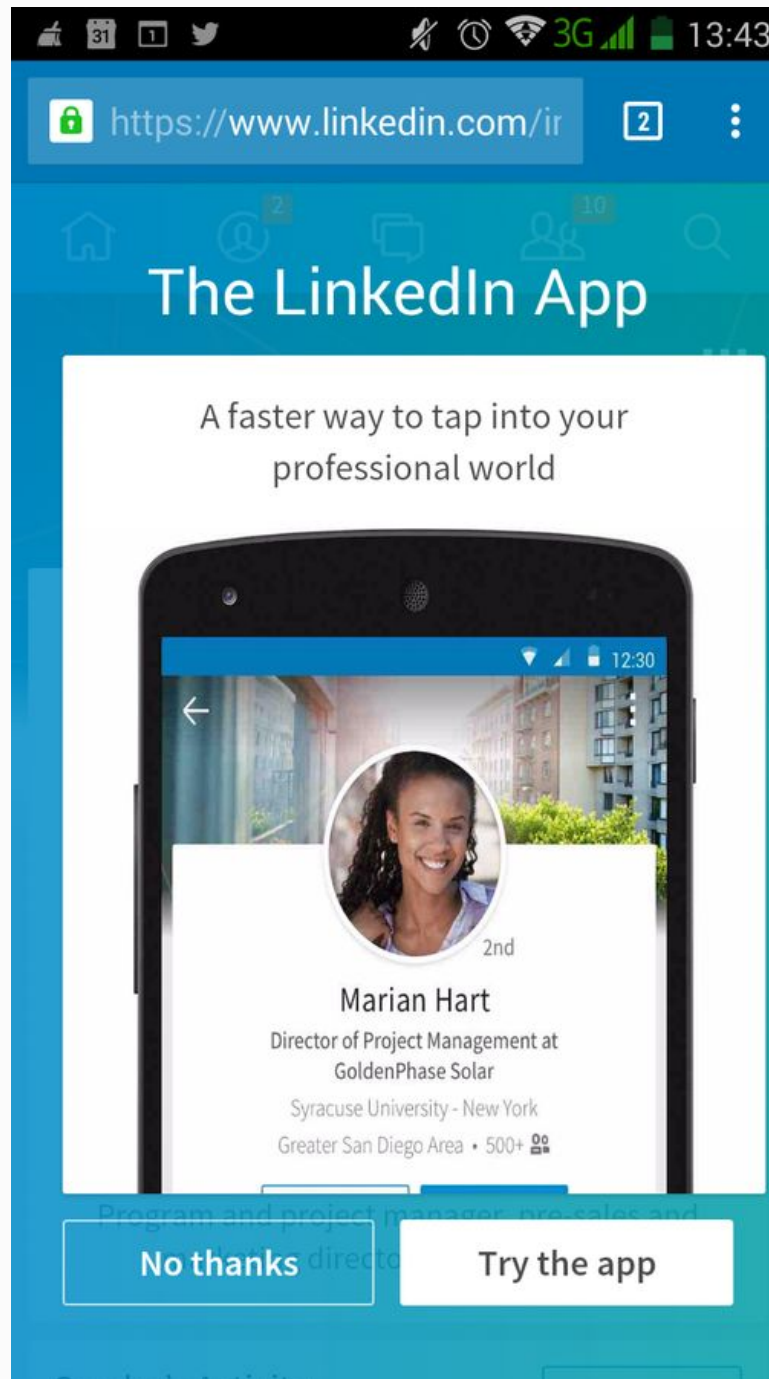
It's important to remember that Progressive Web Apps work everywhere but are supercharged in modern browsers. Progressive enhancement is a backbone of the model.



What is the point?

- Installs are not an issue
- Have no APK size restrictions
- Are always up-to-date (do not require re-installs for an update to happen)
- They work everywhere (a browser is)
- Tend to be trusted more than native apps

EFICODE.COM



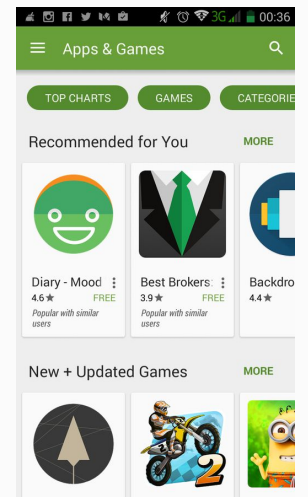
"In a consumer app, every step you make the users take to access your services costs you 20% of your users."

<http://goo.gl/aeFHJ0>

Add to home screen banner

Chrome on Android has support adding in your site to the homescreen for a while now, but recent versions also support proactively suggesting sites to be added using native Web App install banners.

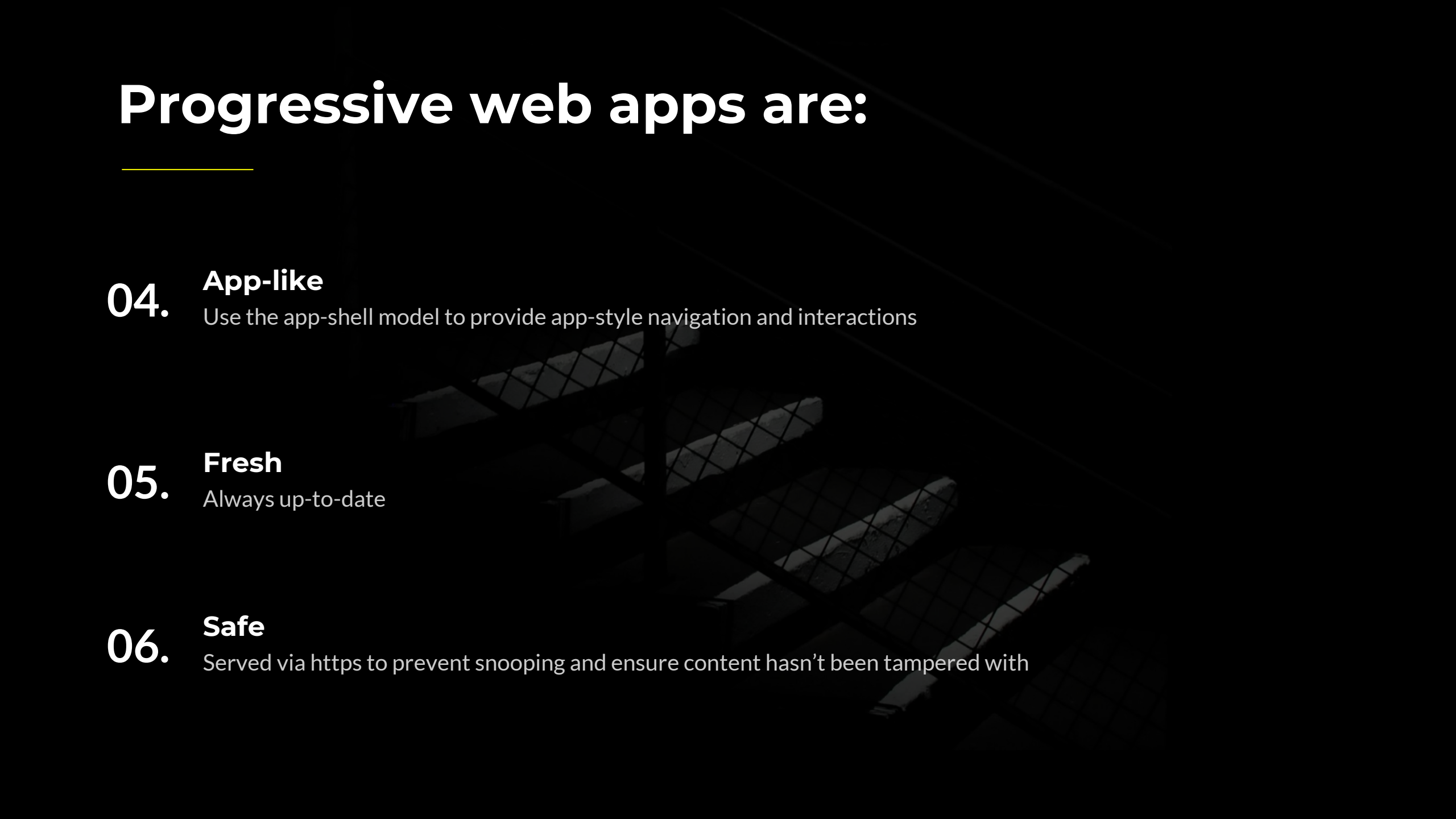
Native App	Progressive
Find the app	Find the app
Download	Use it
Install	
Accept permissions	
Use the app	



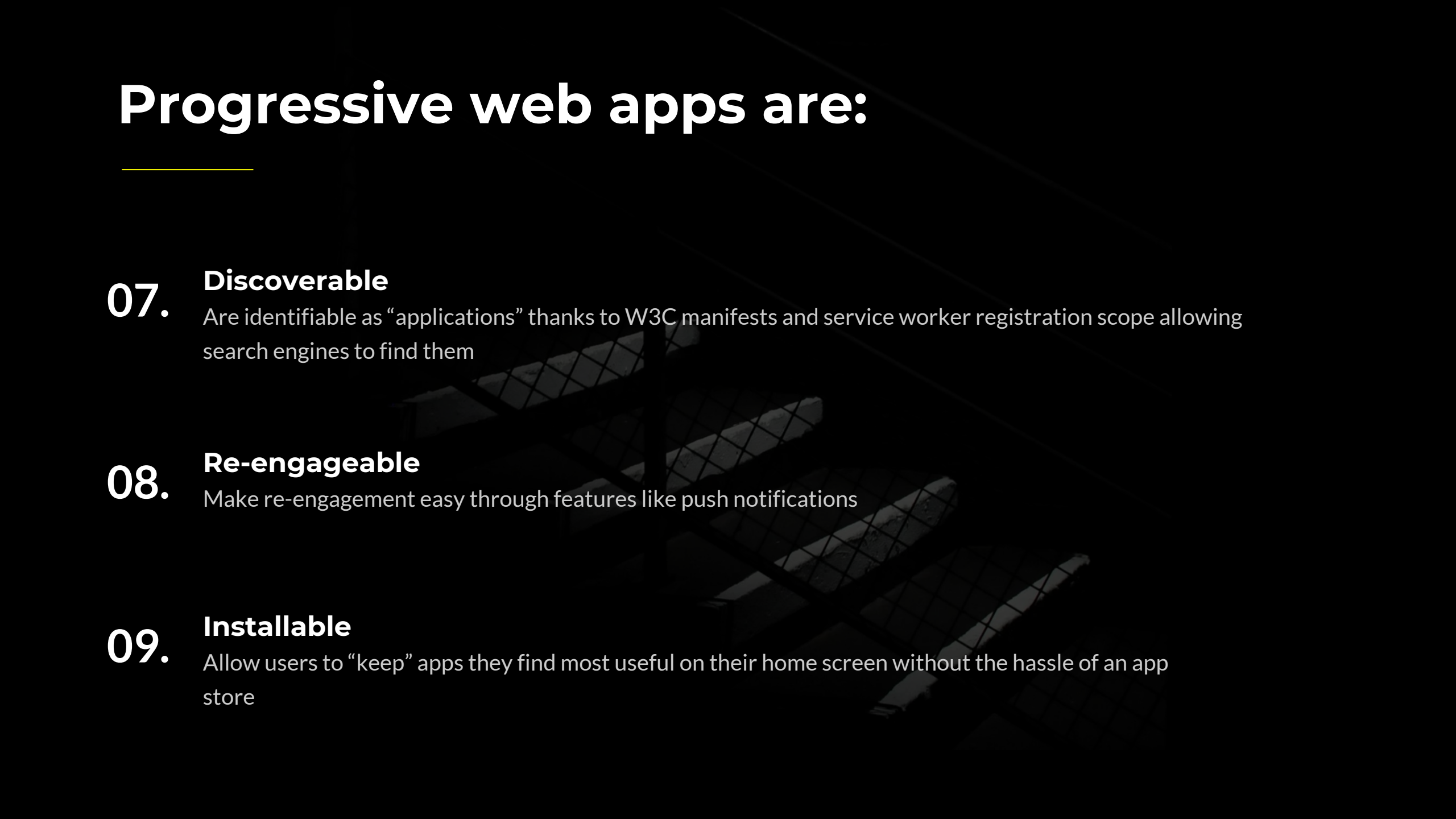
Progressive web apps are:

- 
- 01. Progressive**
Work for every user, regardless of browser choice
 - 02. Responsive**
Fit any form factor, desktop, mobile, tablet or whatever is next
 - 03. Connectivity independent**
Enhanced with service workers to work offline or on low quality networks

Progressive web apps are:

- 
- 04. App-like**
Use the app-shell model to provide app-style navigation and interactions
 - 05. Fresh**
Always up-to-date
 - 06. Safe**
Served via https to prevent snooping and ensure content hasn't been tampered with

Progressive web apps are:

- 
- 07. Discoverable**
Are identifiable as “applications” thanks to W3C manifests and service worker registration scope allowing search engines to find them
 - 08. Re-engageable**
Make re-engagement easy through features like push notifications
 - 09. Installable**
Allow users to “keep” apps they find most useful on their home screen without the hassle of an app store

Principles

Progressive apps aren't packaged and deployed through stores, they're just websites that took all the right vitamins. They progressively become "apps"

- Alex Russel (Google Software Engineer)

01. Web App Manifest

Define app properties: icon, name, orientation, mode etc.

03. Service Workers

Client side network proxy. But that's not the only trick it has up it's sleeves.

02. Add to Home Screen Banner

Re-engage the user with push notifications and by installing the software as an "app"

04. Secure: SSL / HTTPS

Service worker has a lot of power. We wouldn't want it to fall into the wrong hands.



EFICODE.COM

Web App Manifest

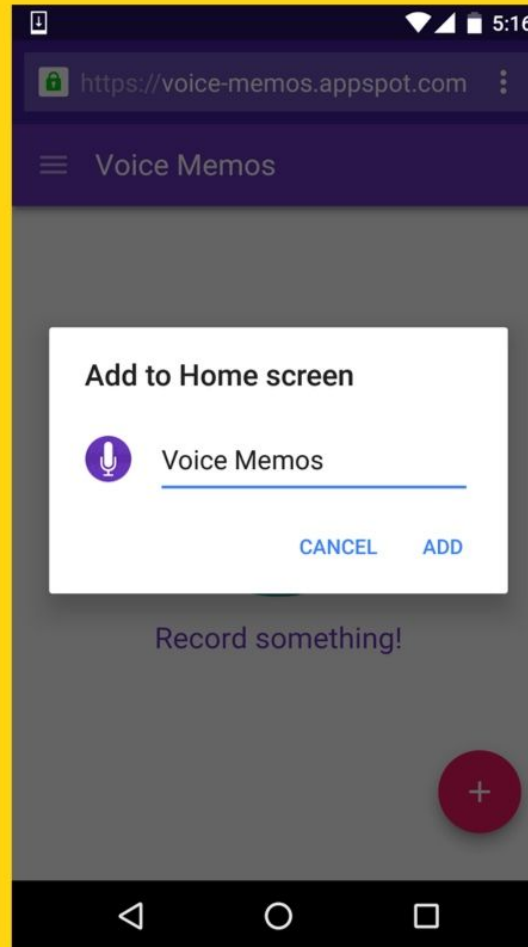
The Manifest for Web applications is a simple JSON file that gives you, the developer, the ability to control how your app appears to the user in the areas that they would expect to see apps (for example the device home screen), direct what the user can launch and more importantly how they can launch it.

```
{
  name: "My Awesome Web App",
  short_name: "Awesome App",
  icons: [{
    src: "images/touch/icon-96x96.png",
    sizes: "96x96",
    type: "image/png"
  }, {
    src: "images/touch/icon-128x128.png",
    sizes: "128x128",
    type: "image/png"
  }, {
    src: "images/touch/apple-touch-icon.png",
    sizes: "152x152",
    type: "image/png"
  }, {
    src: "images/touch/chrome-touch-icon-192x192.png",
    sizes: "192x192",
    type: "image/png"
  }],
  start_url: "/index.html?homescreen=1",
  display: "standalone",
  background_color: "#3E4EB8",
  theme_color: "#2F3BA2"
}
```


Manifest

The manifest enables your web app to have a more native-like presence on the user's home screen. It allows the app to be launched in full-screen mode, provides control over the screen orientation and in recent versions of Chrome on Android supports defining a Splash Screen and theme color for the address bar. It is also used to define a set of icons by size and density used for the aforementioned Splash screen and homescreen icon.

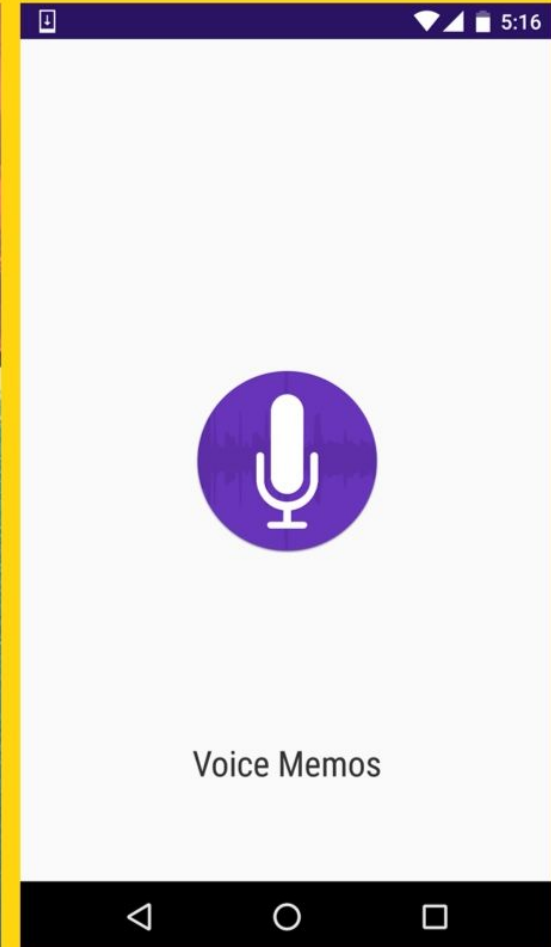
EFICODE.COM



Add to Home screen



Launch from Home screen

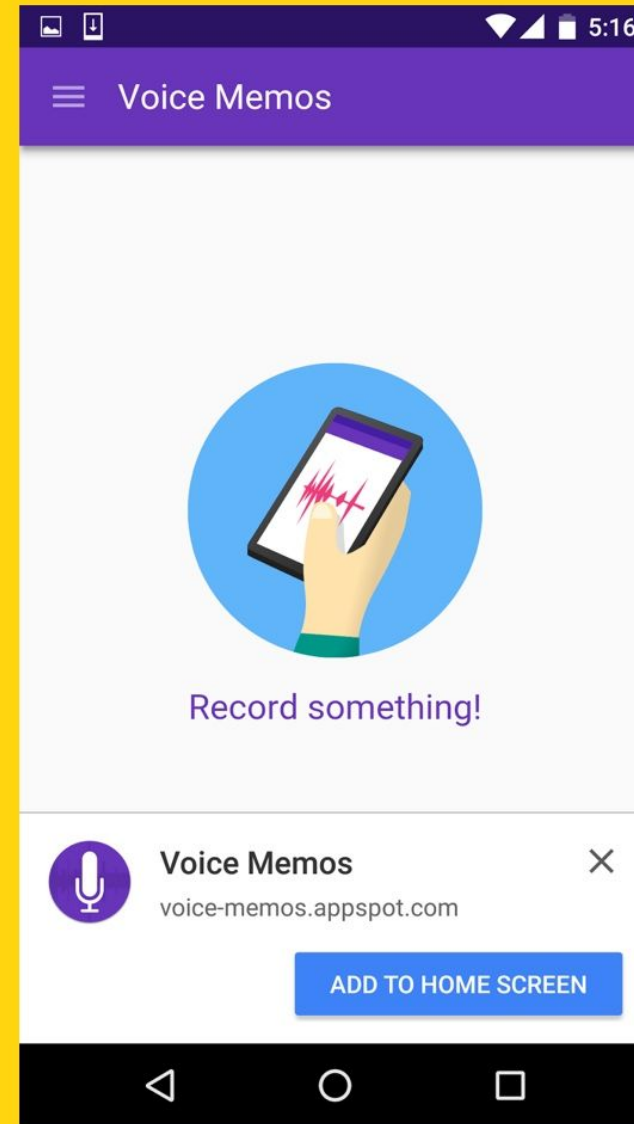


Full-screen "app-like" experience.
Splash screen customisation

In order for the app install prompts to display your app must:

- Have valid Web App manifest
- Be served over HTTPS
- Have valid service worker registered
- Be visited twice, with at least 5 minutes between visits
- A number of App Install banner samples are available, covering basic banners through to more complex use-cases like displaying related applications

EFICODE.COM





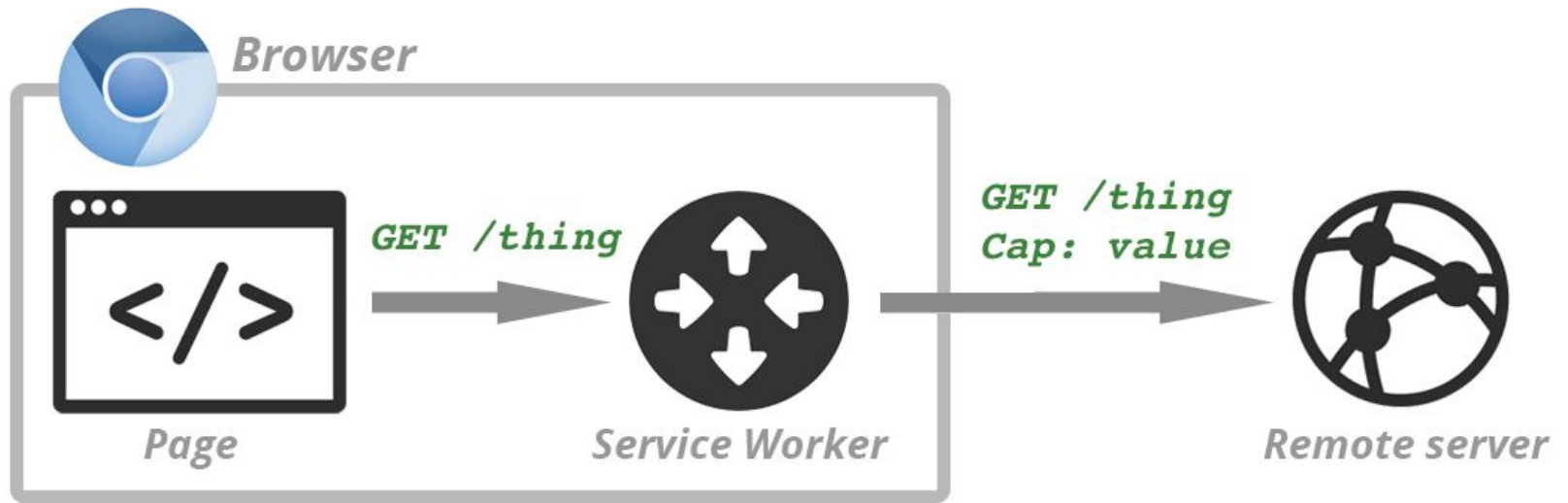
Service worker for offline caching

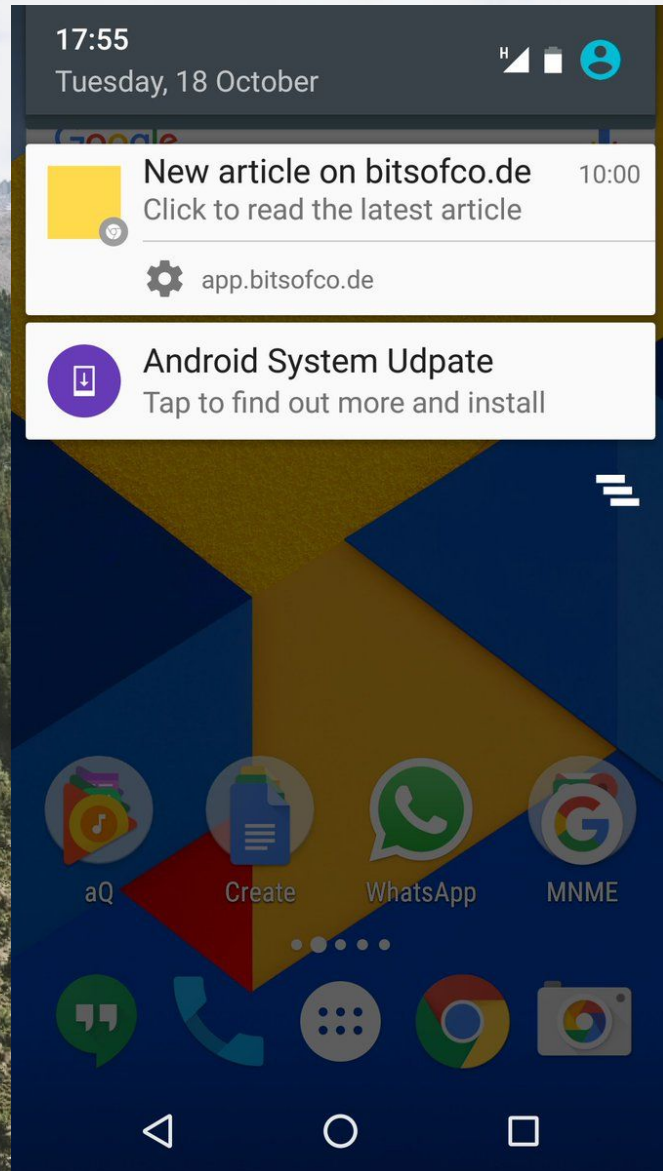
A service worker is a script that runs in the background, separate from your web page. It responds to events, including network requests made from pages it serves. A service worker has an intentionally short lifetime.

Service worker

Service workers are powerful for offline caching but they also offer significant performance wins in the form of instant loading for repeat visits to your site or web app. You can cache your application shell so it works offline and populate its content using JavaScript

EFICODE.COM





Push notifications for re-engagement

Push notifications allow your users to opt-in to timely updates from sites they love and allow you to effectively re-engage them with customized, engaging content.

Thank you!

Questions?



@shnigi_



nikiahlskog



@shnigi



shnigi

niki.ahlskog@eficode.com



<https://www.eficode.com/careers>

EFICODE.COM

