Progressive Web App

Agenda

Progressive Web Application

Type of Application

Delivered Through the Web

Using Web Technologies

Working on Any Platform

Supported by Google

Supported by Microsoft

Supported by Mobile Browsers

Available in Google Play Store

Available in Microsoft Store

PWA Characteristics

Progressive

Discoverable

App-Like

Conectivity-Independent

Installable

Fresh

PWA Introduction

https://youtu.be/fuhAmUpEEHA

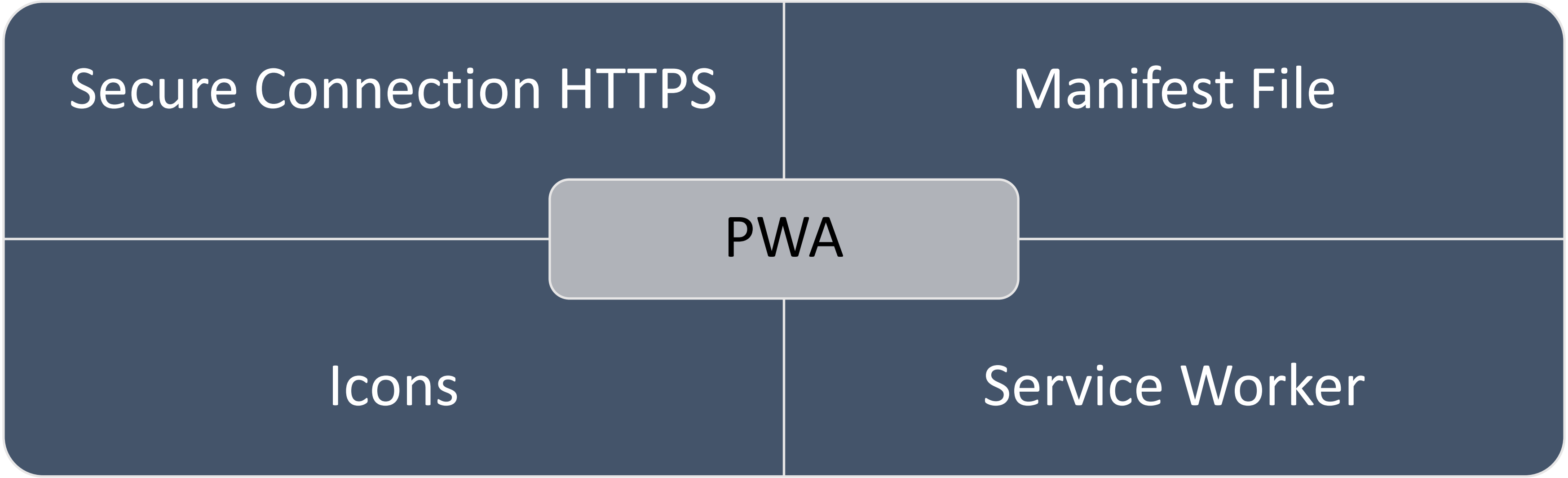
PWA Training

https://youtu.be/DfFlBWCQjzA

PWA Example

https://youtu.be/UVcPyiRLcgU

PWA Requirements



Secure Connection HTTPS

Manifest File

PWA

Icons

Service Worker

Main App File

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Hello World</title>
  <meta name="viewport" content="width=device-width,
                        initial-scale=1.0">
</head>
<body>
  <div>
    <h1>Hello World!</h1>
  </div>
</body>
</html>
```

Web App Manifest

Nazwa aplikacji

Ikony aplikacji

URL do uruchomienia aplikacji

Domyślna orientacja okna aplikacji

Tryb wyświetlania

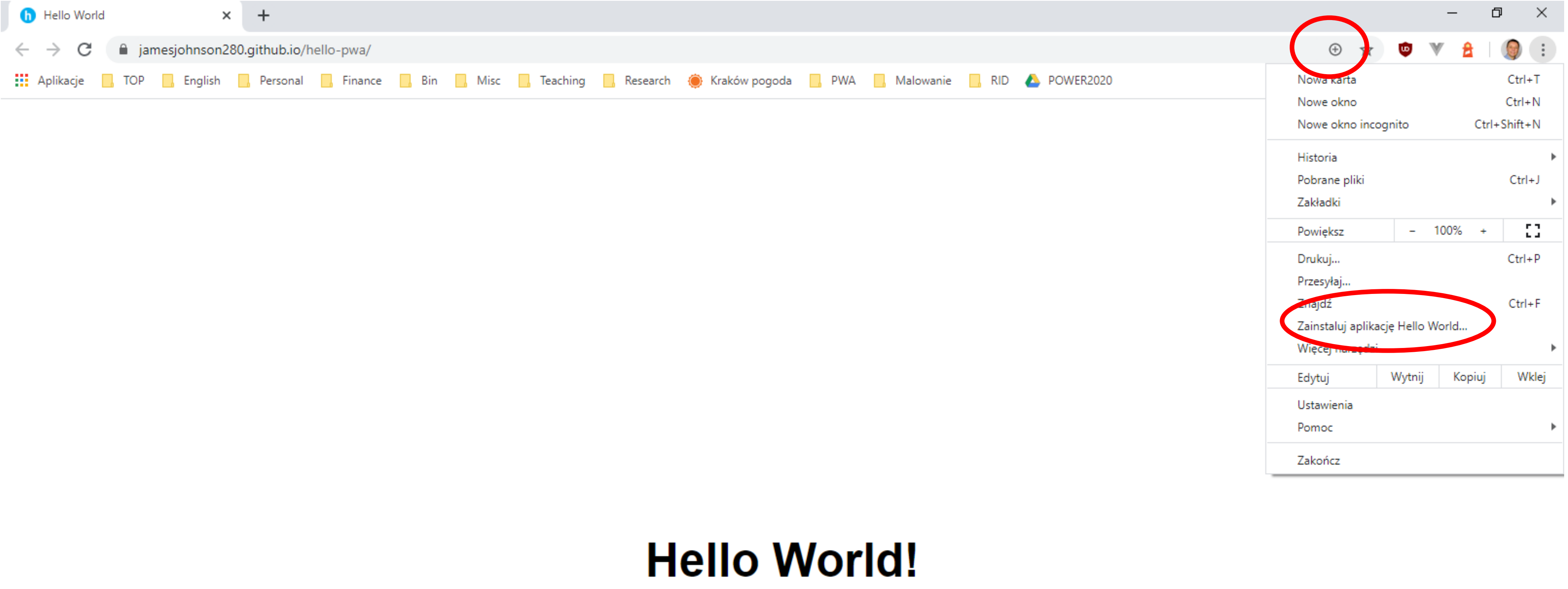
Manifest File Example

```
{
  "short_name": "Hello",
  "name": "Hello World",
  "description": "My First PWA",
  "icons": [
    {
      "src":   "icons-192.png",
      "type":  "image/png",
      "sizes": "192x192"
    },
    {
      "src":   "icons-512.png",
      "type":  "image/png",
      "sizes": "512x512"
    }
  ],
  "start_url": "/index.html",
  "background_color": "white",
  "display": "standalone",
  "scope": "/",
  "theme_color": "white"
}
```

Adding Manifest to App HTML Files

```
<link rel="manifest" href="manifest.json">
```

PWA Installation



PWA Examples

https://airly.eu/map/pl/

https://jamesjohnson280.github.io/hello-pwa/

https://github.com/jamesjohnson280/hello-pwa

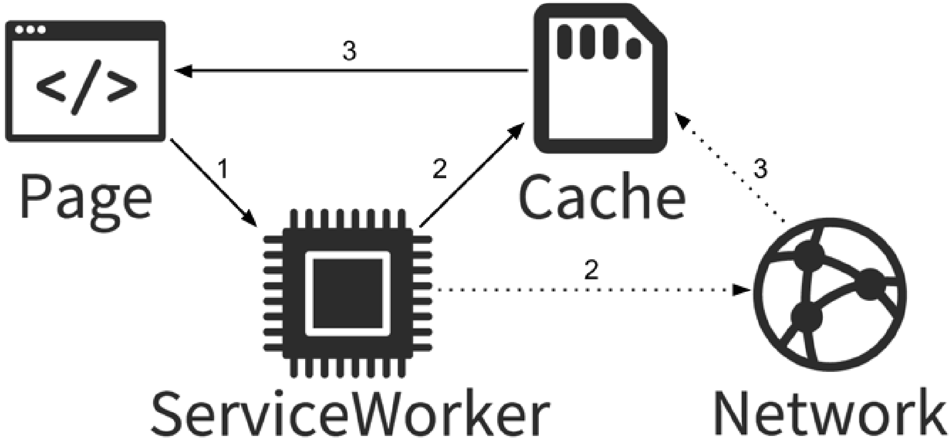
Service Worker

Offline first / Cache first

Virtual proxy between browser and network

Run on a separated thread

No access to DOM



Service Worker Registration

```
if ('serviceWorker' in navigator) {
  // Register a service worker hosted at the root of the
  // site using the default scope.

navigator.serviceWorker.register('/sw.js').then(function
(registration) {
    console.log('Service worker registration
succeeded:', registration);
  }, /*catch*/ function(error) {
    console.log('Service worker registration failed:',
error);
  });
} else {
  console.log('Service workers are not supported.');
}
```

Service Worker Registration

```
window.addEventListener('load', () => {
  if (!('serviceWorker' in navigator)) {
    // service workers not supported
    return
  }

  navigator.serviceWorker.register('/sw.js').then(
    () => {
      // registered!
    },
    err => {
      console.error('SW registration failed!, err)
    }
  )
})
```

https://developer.mozilla.org/en-US/docs/Web/API/ServiceWorkerContainer/register

SW Events

INSTALL - is sent when the service worker is being installed.

ACTIVATE - is sent when the service worker has been registered and installed. This place is where you can clean up anything related to the older version of the service worker if it’s been updated.

FETCH - is sent whenever a page of your site requires a network resource. It can be a new page, a JSON API, an image, a CSS file, whatever.

SYNC - is sent if the browser previously detected that the connection was unavailable, and now signals the service worker that the internet connection is working.

PUSH - is invoked by the Push API when a new push event is received.

Serving Cached Resources

```
self.addEventListener('install', event => {
  event.waitUntil(
    caches
      .open('my-site-name')
      .then(cache =>
        cache.addAll([
          'favicon.ico',
          'style.css',
          'script.js'
        ])
      )
  )
})
```

Fetching a cached resource

```
self.addEventListener('fetch', event => {
  event.respondWith(
    caches.match(event.request).then(response => {
      if (response) {
        //we found an entry in the cache!
        return response
      }
      return fetch(event.request)
    })
  )
})
```

Introduction to Service Workers

https://youtu.be/jVfXiv03y5c

PWA From Scratch

https://www.freecodecamp.org/news/build-a-pwa-from-scratch-with-html-css-and-javascript/

PWA Builder

https://www.pwabuilder.com/