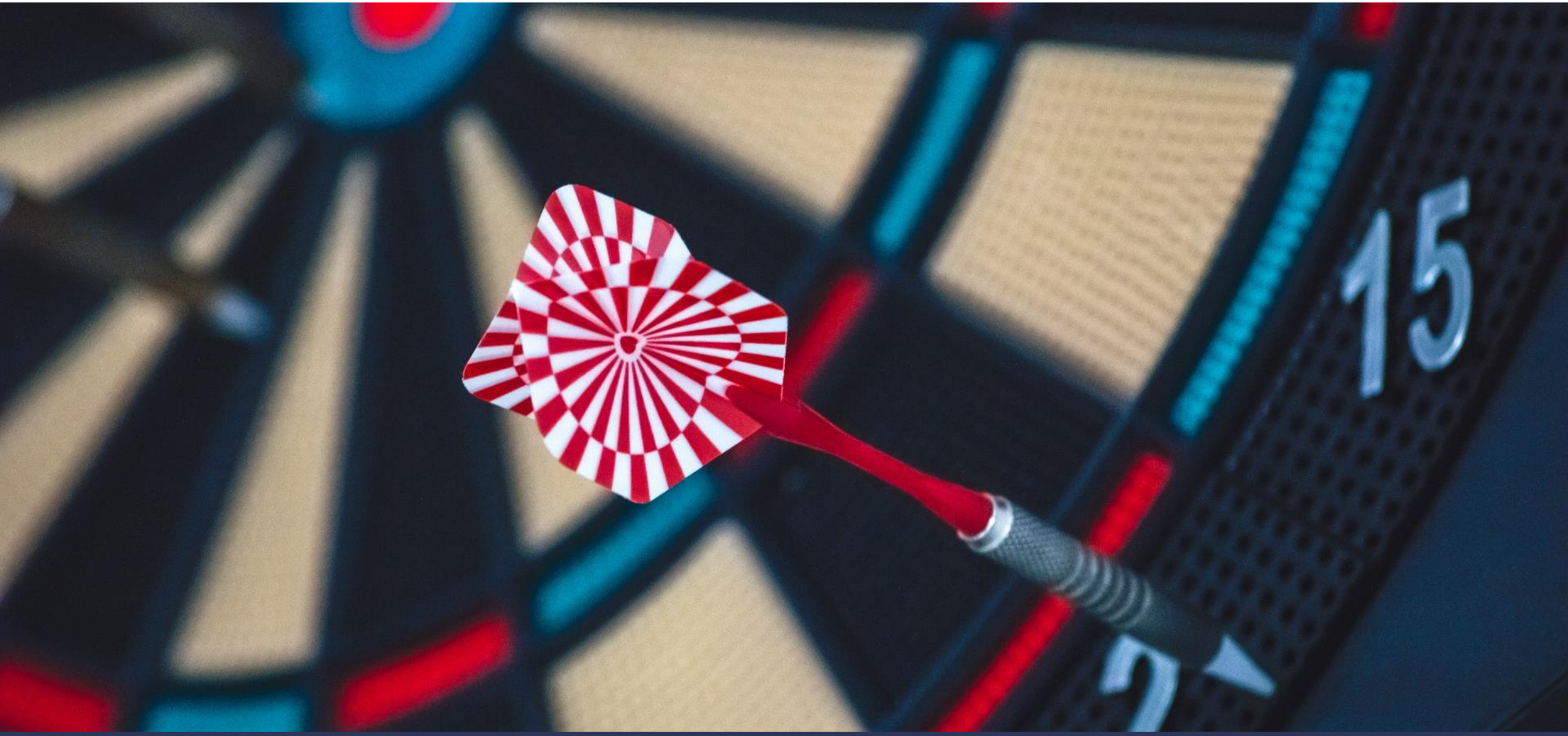
ITDevCon9
European Delphi Conference

Marco Breveglieri
ABLS Team

PROGRESSIVE WEB APPS (PWA) IN DELPHI

OBIETTIVO



Obiettivo

Trasformare applicazioni Web in app mobile con poco sforzo e tanti benefit.

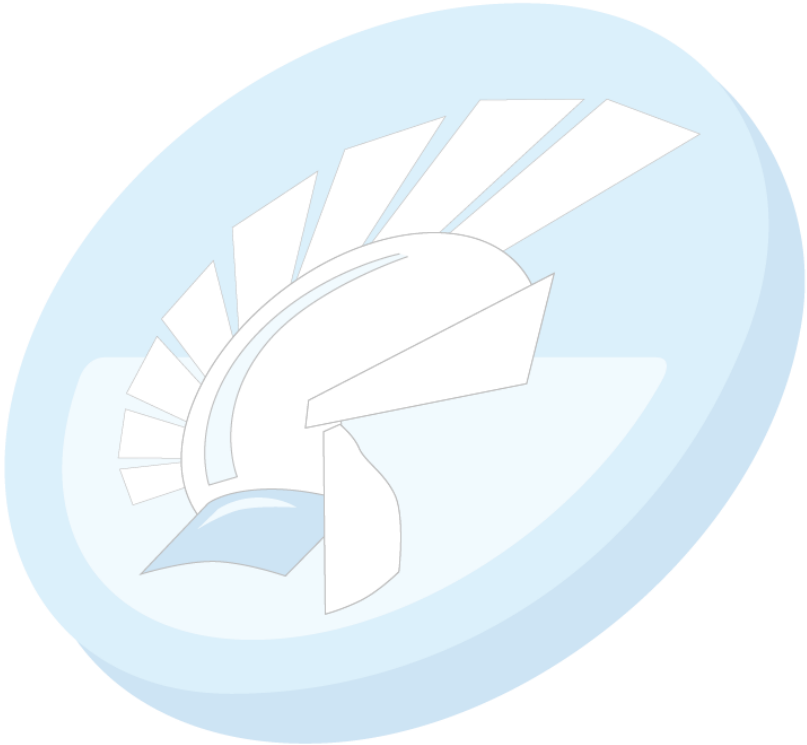
Scenari

- Devo creare una Mobile App e una Web App
  - Ho già creato la Web App, devo creare la versione Mobile
  - Ho già creato la Mobile App, devo creare la versione Web
- La mia Mobile App deve girare su piattaforme particolari (es. Android TV, Raspberry PI, Windows IoT, …)
- Non conosco FireMonkey, ma ho usato altri framework (es. IntraWeb, uniGUI, ExtJS, TMS Web Core, …)
- Ho creato una Mobile App, ma nessuno la installa
- Voglio aumentare il "reach potenziale" della mia App (Web o Mobile)

C'È UNA SOLUZIONE?

ITDevCon9
European Delphi Conference

Progressive Web Apps (PWA)



DEFINIZIONE /1

«Experiences that combine the best of web and the best of apps».

Alex Russel (2015)

DEFINIZIONE /2

«E' una normalissima applicazione Web, ma con qualcosa in più».

Marco Breveglieri (2018) 😄

DEFINIZIONE /3

«A new way to deliver amazing user experiences on the web».

Google

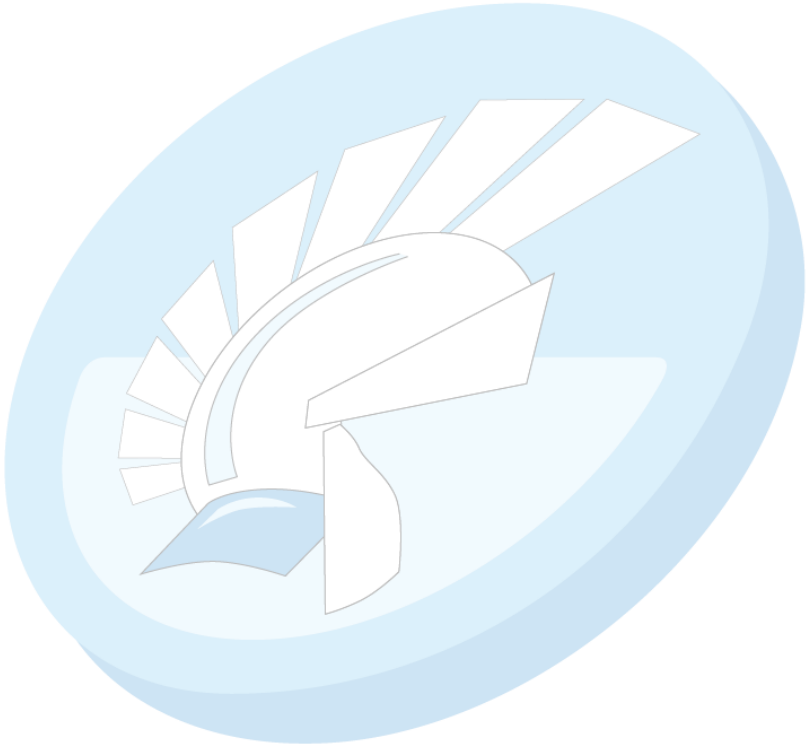
Cosa sono le PWA?

- Termine coniato da Alex Russel nel 2015 (1)
- Appaiono all'utente come app native
- Possono lavorare sia online che offline
- Sono indipendenti dal sistema operativo

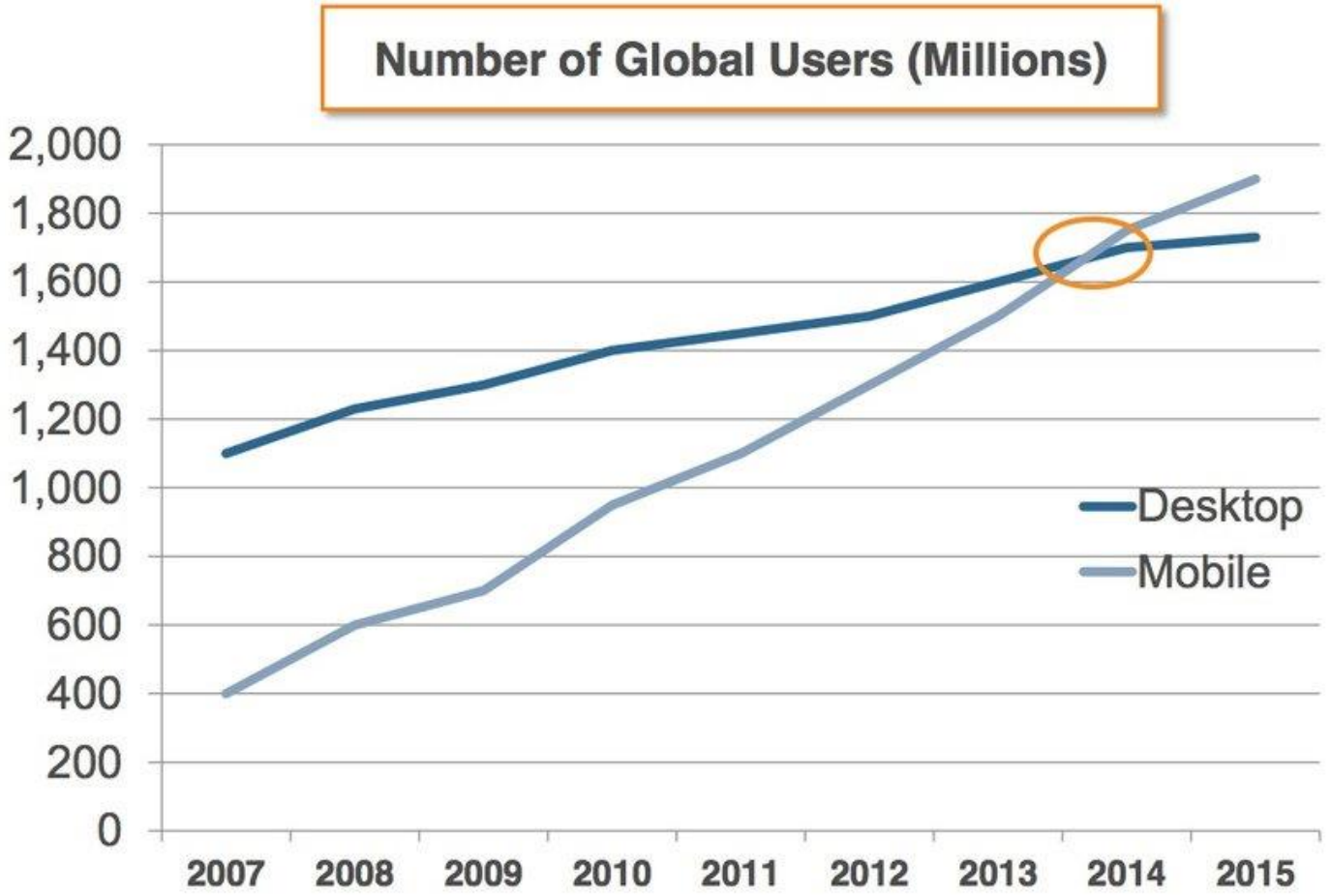
(1) https://medium.com/@slightlylate/progressive-apps-escaping-tabs-without-losing-our-soul-3b93a8561955

ITDevCon9
European Delphi Conference

Perché le PWA?



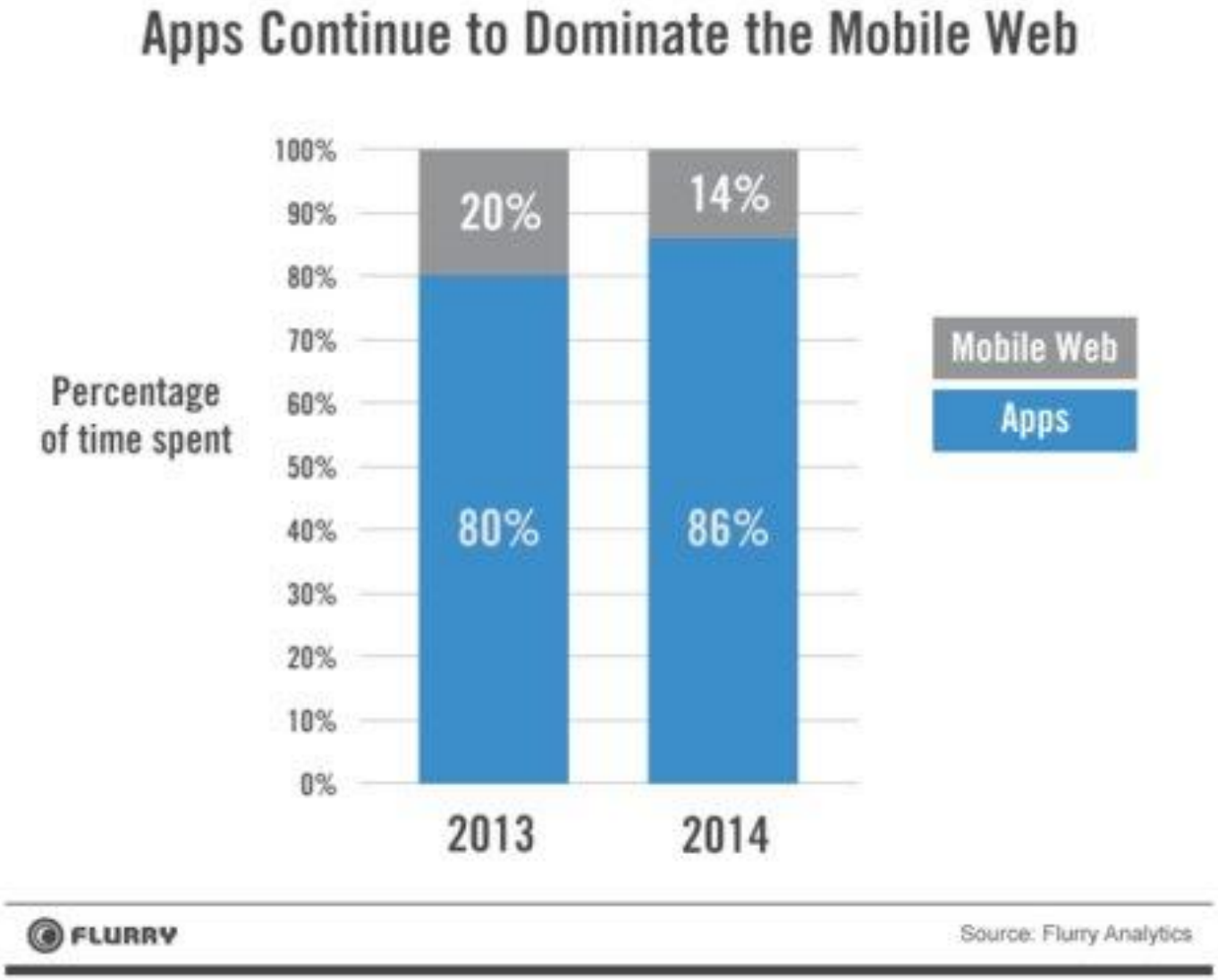
DAL DESKTOP AL MOBILE



comScore.

Source: Morgan Stanley Research

LE APP DOMINANO IL MOBILE



STATISTICHE

80%

del tempo viene trascorso da ciascun utente nell'uso delle 3 applicazioni preferite

ZERO

sono il numero medio di nuove applicazioni installate da ciascun utente per ogni mese

PERCHÉ SONO IMPORTANTI?

oltre

1 MILIARDO

di utenti navigano con Chrome su dispositivi mobile ogni mese

(rilevamento del 2016)

VANTAGGI PRINCIPALI

- Maggior gradimento da parte degli utenti
  - Aggiunta icona all'home screen
    - Solo quando sono stati raggiunti gli standard qualitativi
    - Il browser propone la scelta all'utente in automatico
- Indipendenza dallo stato della rete
  - Caricamento istantaneo della applicazione
    - Il controllo viene affidato al Service Worker
    - E' possibile sfruttare cache e altre risorse scaricate
  - Minor traffico generato dal server
    - Case Study di Konga ha mostrato un risparmio del 63% nel caricamento della pagina iniziale e 84% per il completamento della prima transazione
- Maggior coinvolgimento dell'utente
- Incremento del numero di conversioni (da utenti nuovi ad abituali)

CASE STUDY: ALIEXPRESS

AliExpress

2x more

page views

74% increase

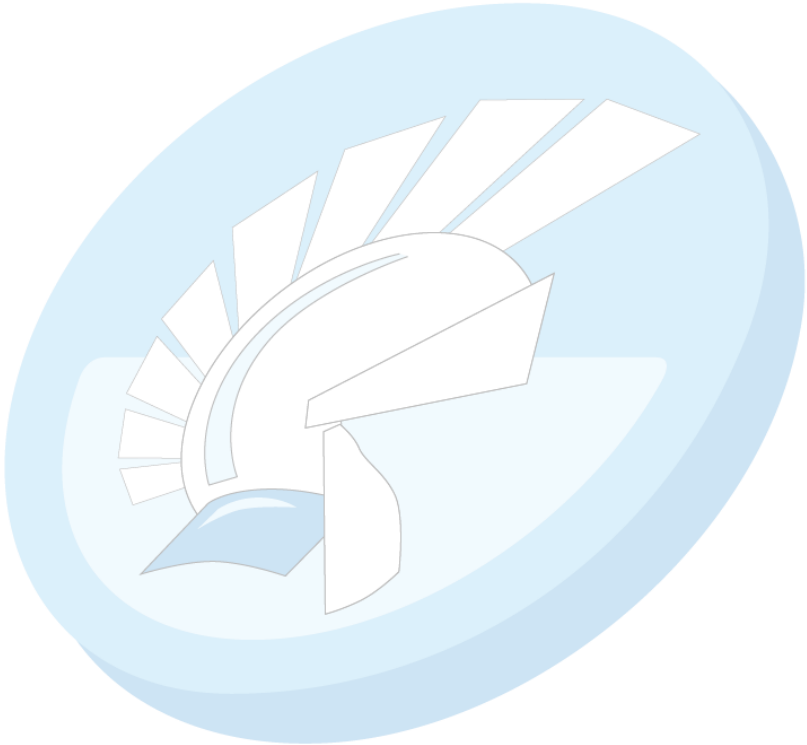
in time spent

82% more

conversions on iOS

Se ne possono trovare altri ancora qui: 👉 **https://www.pwastats.com/**

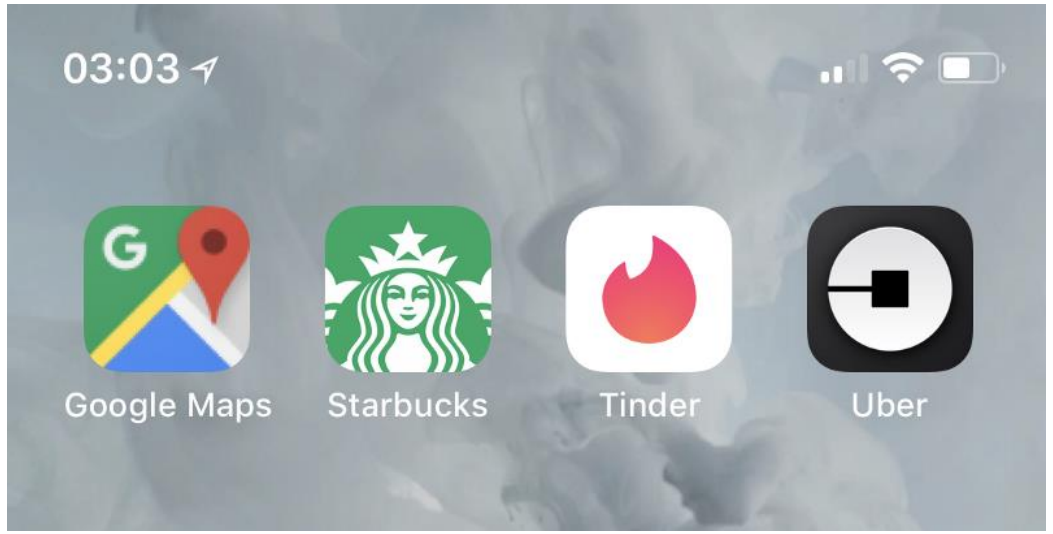
QUALE SUPPORTO?



L'annuncio di Steve Jobs

PWA su iOS

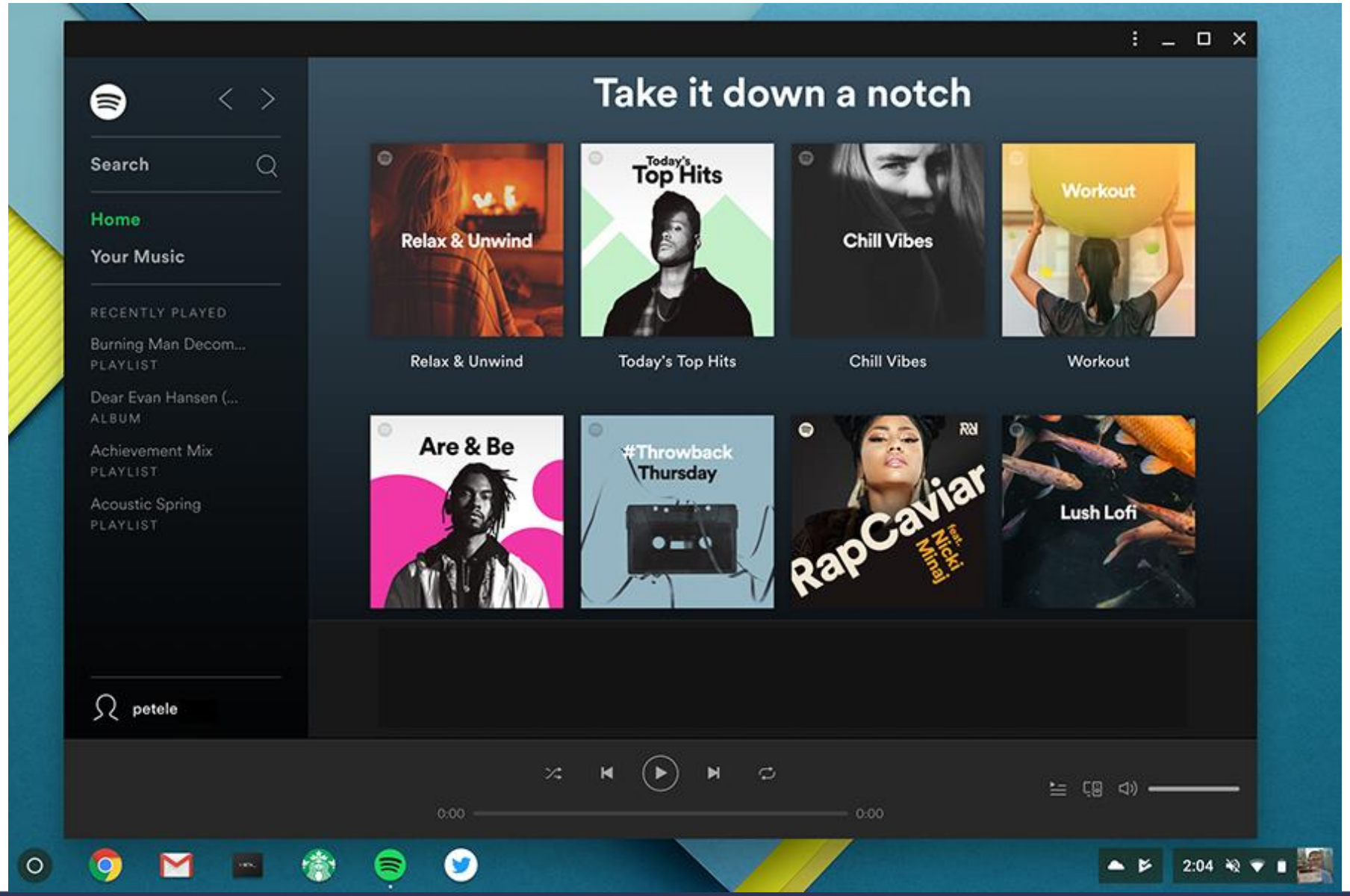
- Le PWA su iOS possono accedere a
  - Geolocalizzazione
  - Sensori (magnetometro, accelerometro, giroscopio)
  - Camera
  - Audio in output
  - Sintetizzatore vocale
  - Apple Pay
  - WebAssembly, WebRTC, WebGL e altre feature sperimentali



PWA su Android

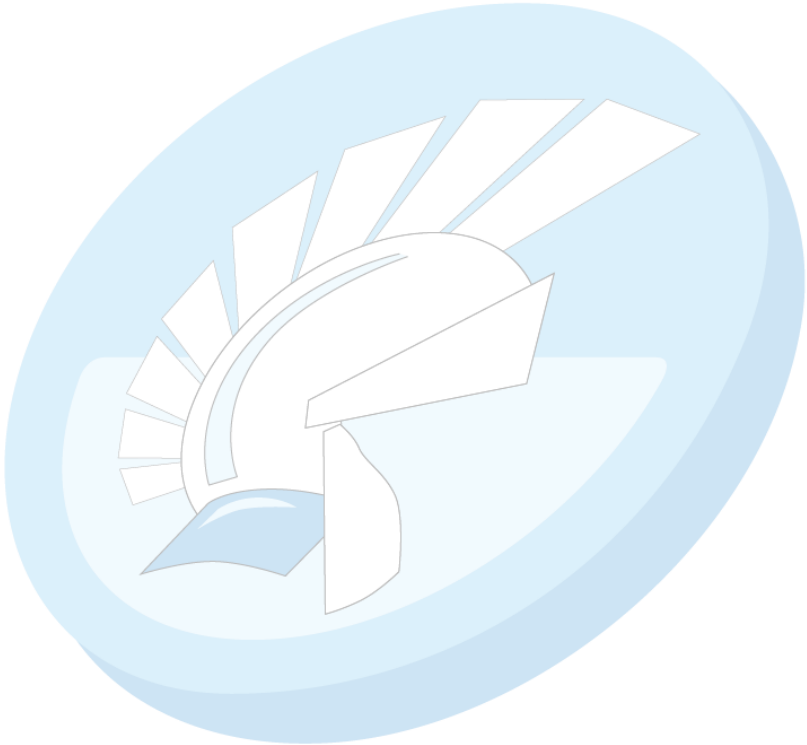
- Le PWA su Android hanno le stesse feature di iOS, in più
  - Possono usare fino a 50MB
  - Non vengono cancellate a meno che non sia necessario
  - Accesso a Bluetooth LE
  - Riconoscimento vocale
  - Sincronizzazione in background
  - Web Push Notification
  - Web App Banner per invitare l'utente a installare l'app
  - Customizzazioni varie su icona, orientamenti, colori, ecc. e finestra stand-alone

ORA ANCHE SUL DESKTOP!



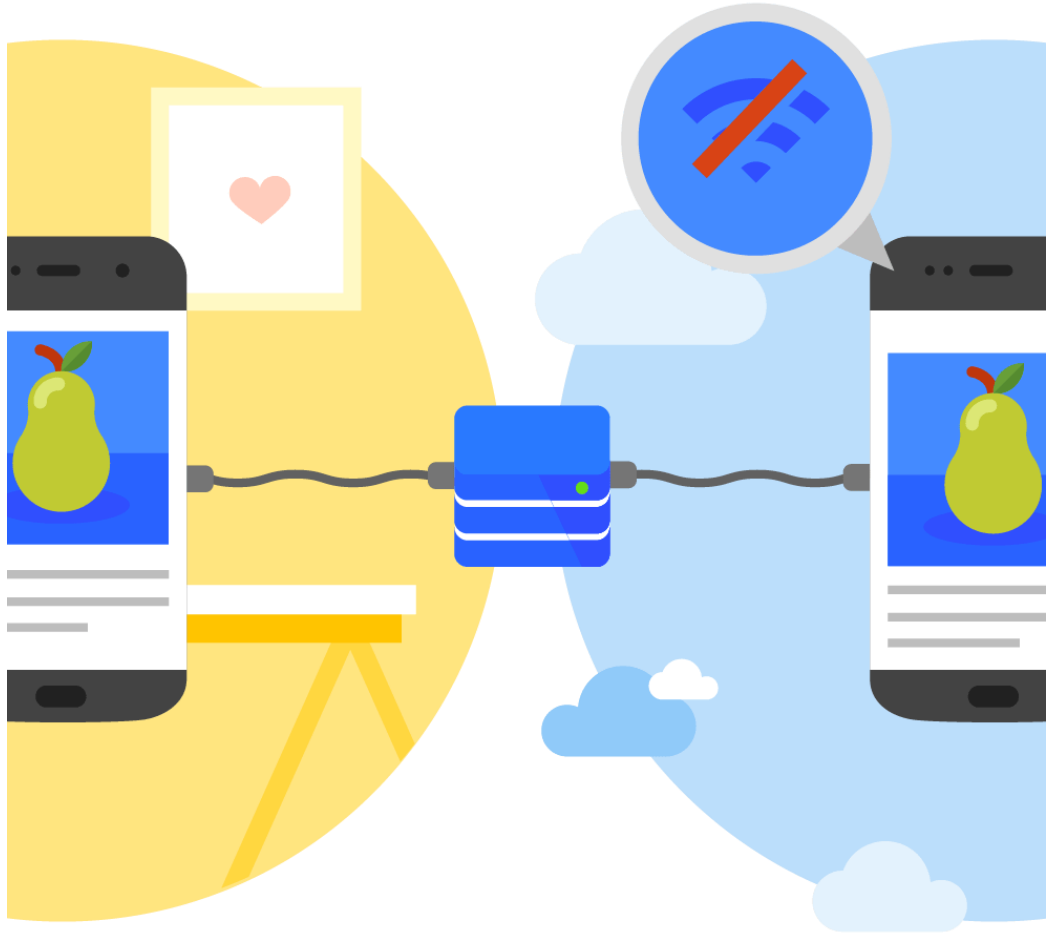
ITDevCon9
European Delphi Conference

CARATTERISTICHE PRINCIPALI



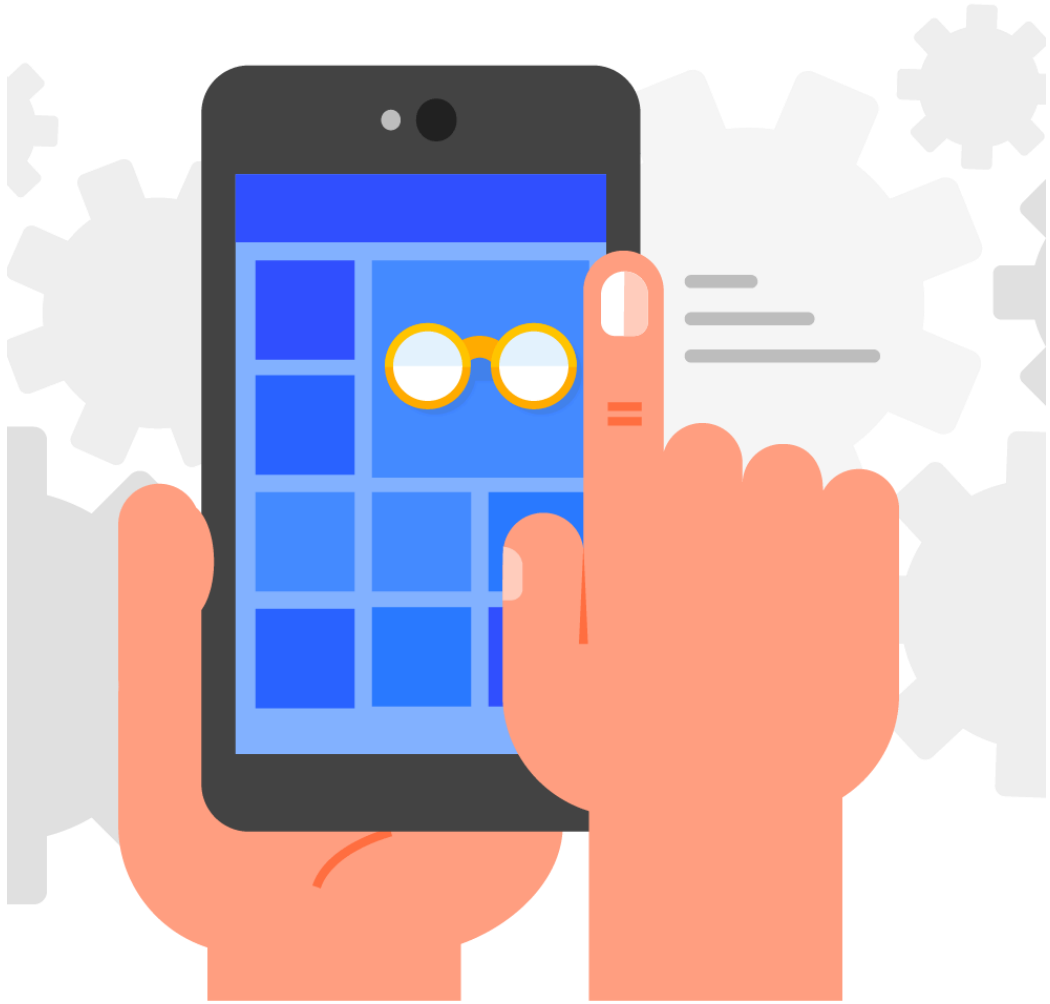
AFFIDABILI

- Si caricano istantaneamente
- Sono indipendenti dallo stato della rete
- Non mostrano mai il famigerato "downasaur"



RAPIDE

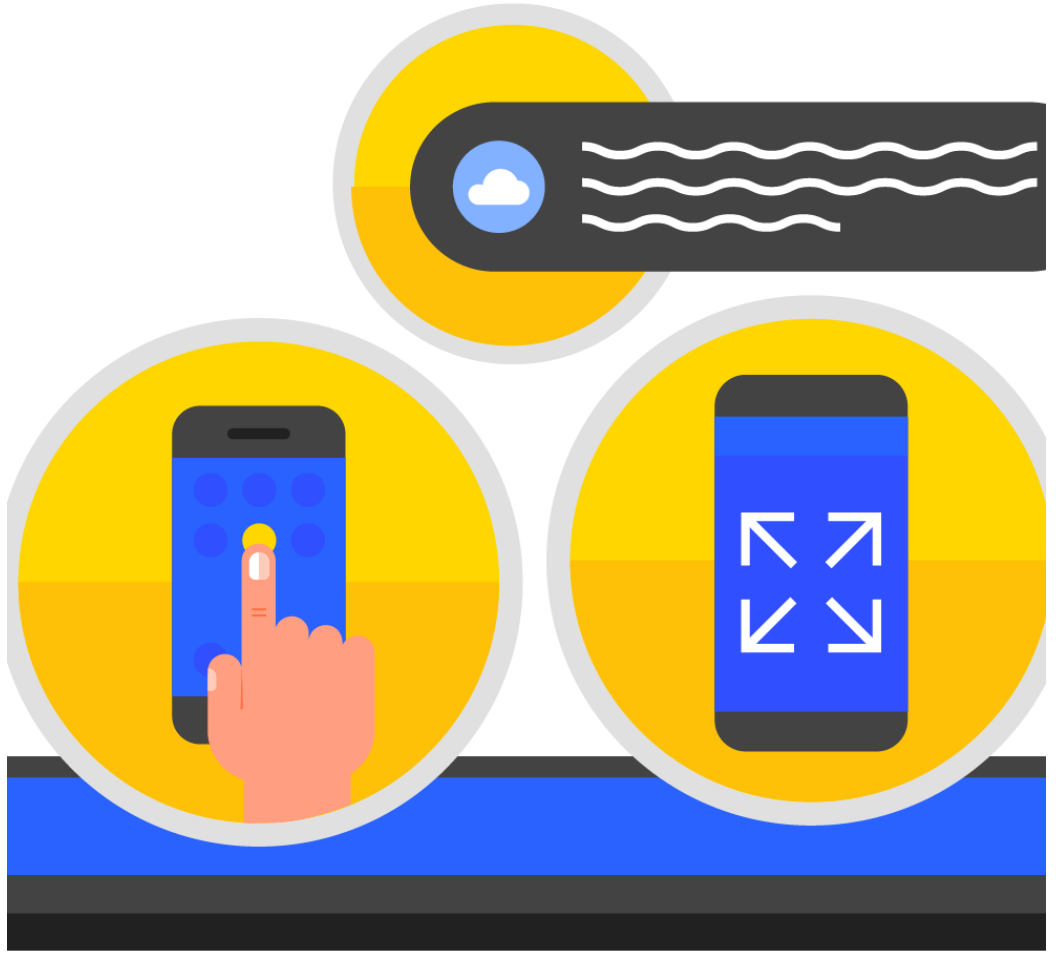
- Rispondono subito alle interazioni dell'utente (*)
- Offrono animazioni fluide e scorrimento di buona qualità
- Riducono percettivamente i tempi di attesa
- Mantengono focalizzata l'attenzione dell'utente



(*) Il 53% degli utenti abbandona siti che impiegano più di 3 secondi a caricarsi.

COINVOLGENTI

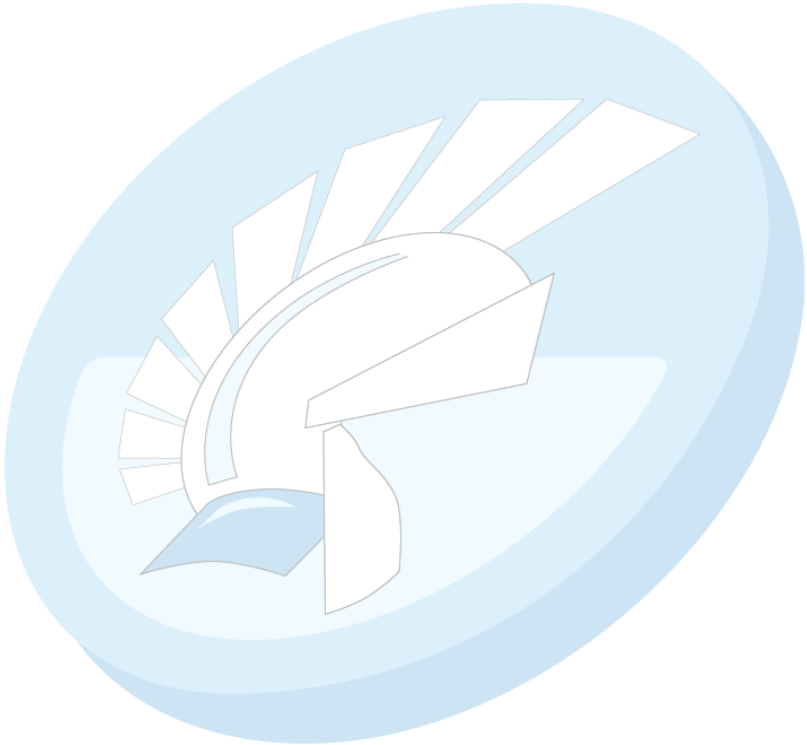
- Funzionano in modo naturale su device mobili
- Si possono installare nella schermata home dell'utente
- Non c'è bisogno di cercarle e scaricarle da uno Store
- Forniscono una esperienza immersiva
  - Visualizzazione in full screen
  - Supporto notifiche push
  - Personalizzazione aspetto e funzionamento



Demo!

ITDevCon9
European Delphi Conference

Architettura



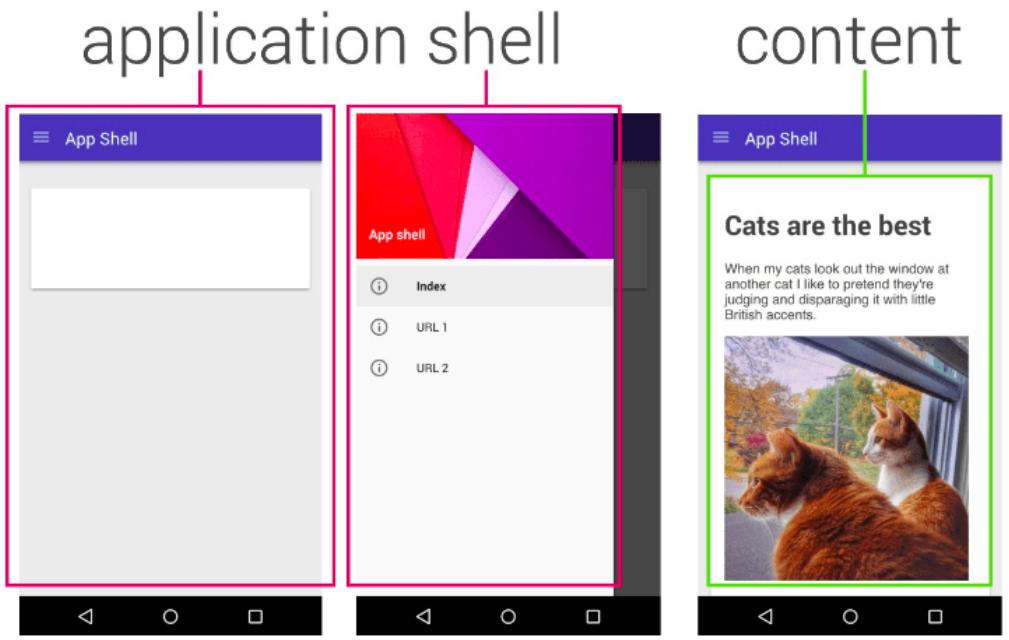
SHELL /1

- Rappresenta il "guscio" della nostra applicazione
- E' la pagina Web (esistente o nuova) che mostra l'interfaccia utente
  - Markup HTML (struttura)
  - Stili CSS (aspetto)
  - Codice JavaScript (comportamento)
- Deve soddisfare i seguenti requisiti:
  - Caricarsi molto rapidamente
  - Essere memorizzata nella cache assieme alle risorse incorporate
  - Visualizzare i contenuti in modo dinamico (*)

(*) ovvero modificare l'interfaccia intervenendo direttamente sulla pagina, evitando la classica navigazione, aiutandosi eventualmente con un framework (facoltativo).

SHELL /2

- Concentrarsi sulla velocità per fornire alla PWA tutto ciò che la rende simile a un'applicazione nativa
- Analizzare il progetto pensando a ciò che deve fare
  - Cosa deve apparire immediatamente sullo schermo? (esempio: un logo, un menu di navigazione, ecc.)
  - Di quali risorse ho bisogno subito per visualizzare l'applicazione? (esempio: immagini, script JavaScript, stili CSS)



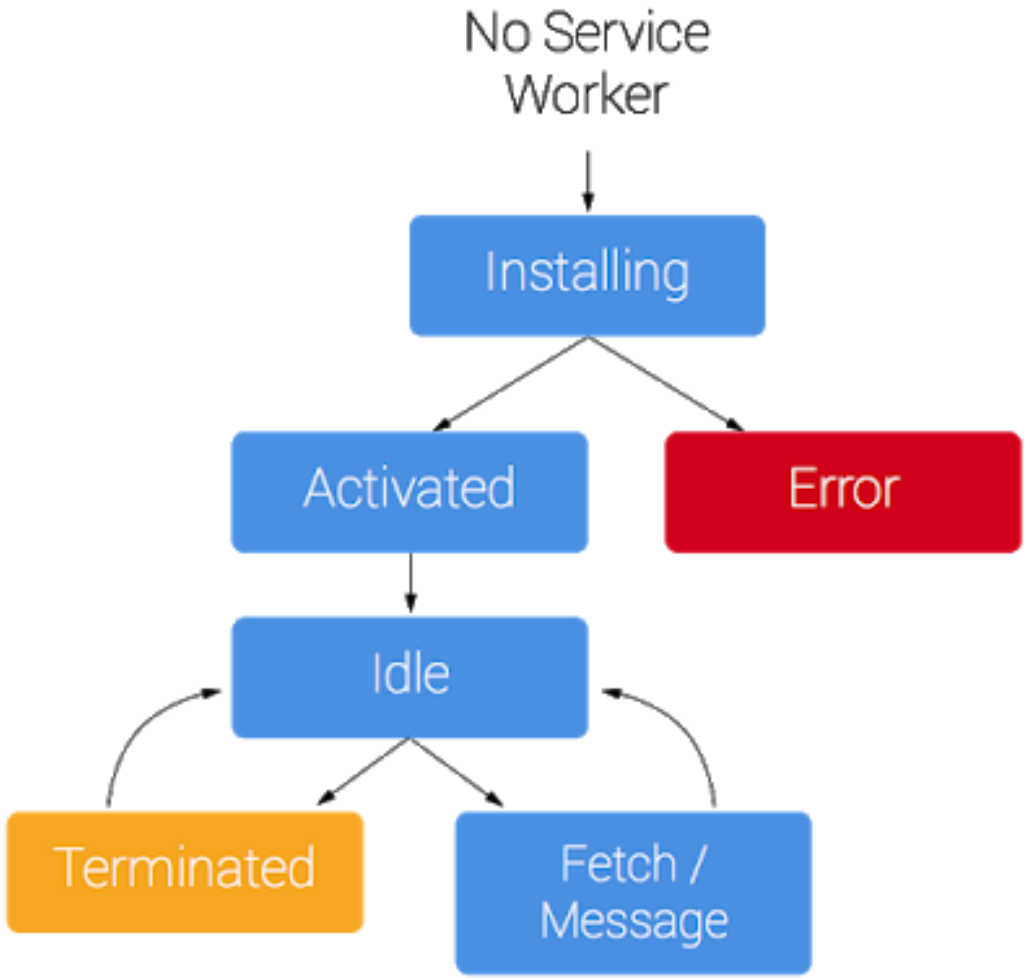
SERVICE WORKER /1

- Si tratta di uno script che consente di eseguire codice in background all'interno del browser
- Viene inizializzato all'interno di una pagina Web, ma può vivere all'esterno di essa
- Può gestire tutte le logiche che non sono espressamente legate all'interfaccia utente (non può accedere al DOM!)
- Nel contesto delle PWA
  - Riceve i comandi che richiedono l'interazione con la rete e ne gestisce il trasferimento dei dati da/verso il server
  - Gestisce la cache per memorizzare risorse e dati delle risposte
  - E' responsabile dell'esperienza utente offline

Service Worker /2

- Il Service Worker ha un proprio ciclo di vita
  - Viene attivato dalla pagina shell
  - Viene installato nel sistema se non è già presente
  - Scarica le risorse statiche all'interno della cache (o gestisce quelle già scaricate)
  - Manterrà il controllo sulle pagine gestite fino a chiusura da parte del sistema (es. per risparmiare memoria/batteria)

La struttura tradizionale dell'app deve essere ripensata!

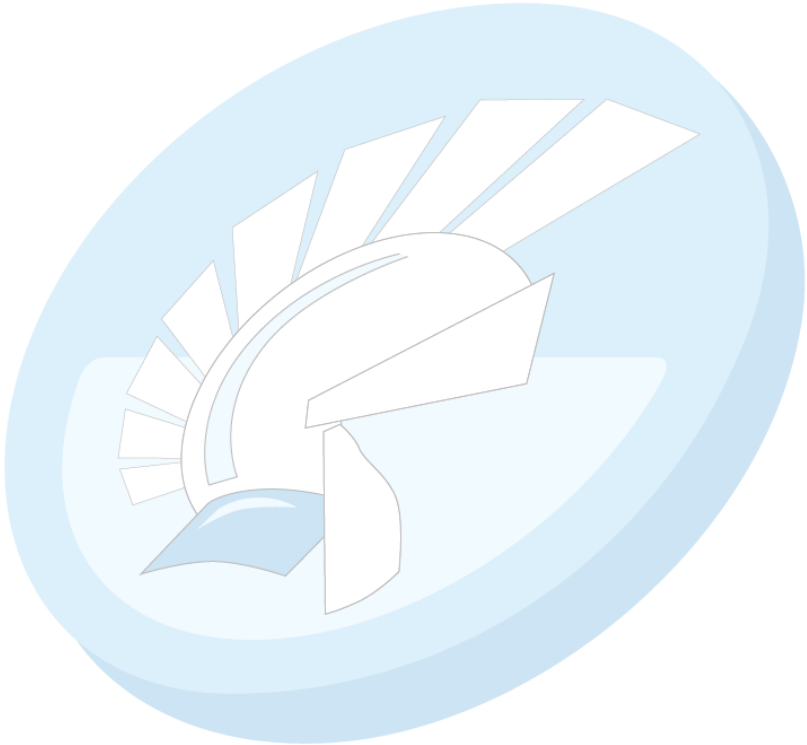


Web App Manifest

- Può essere aggiunto anche a siti Web tradizionali
- E' costituito da un semplice file con nome **manifest.json**
- Configura diverse opzioni dell'applicazione
  - Attribuisce un nome (titolo) da mostrare
  - Definisce l'aspetto visuale dell'applicazione
  - Configurare la modalità di avvio
  - Specifica quali icone visualizzare
    - Icona nella schermata Home
    - Logo nella schermata di caricamento
  - Personalizza il browser che la ospita
    - Visualizzare o meno la barra degli indirizzi

ITDevCon9
European Delphi Conference

PWA in Delphi!



INTRODUZIONE

La nostra PWA è basata sull'esempio *WineCellar*,
incluso nel pacchetto di **DelphiMVCFramework**.

👉 https://github.com/danieleteti/delphimvcframework

Cosa impareremo:

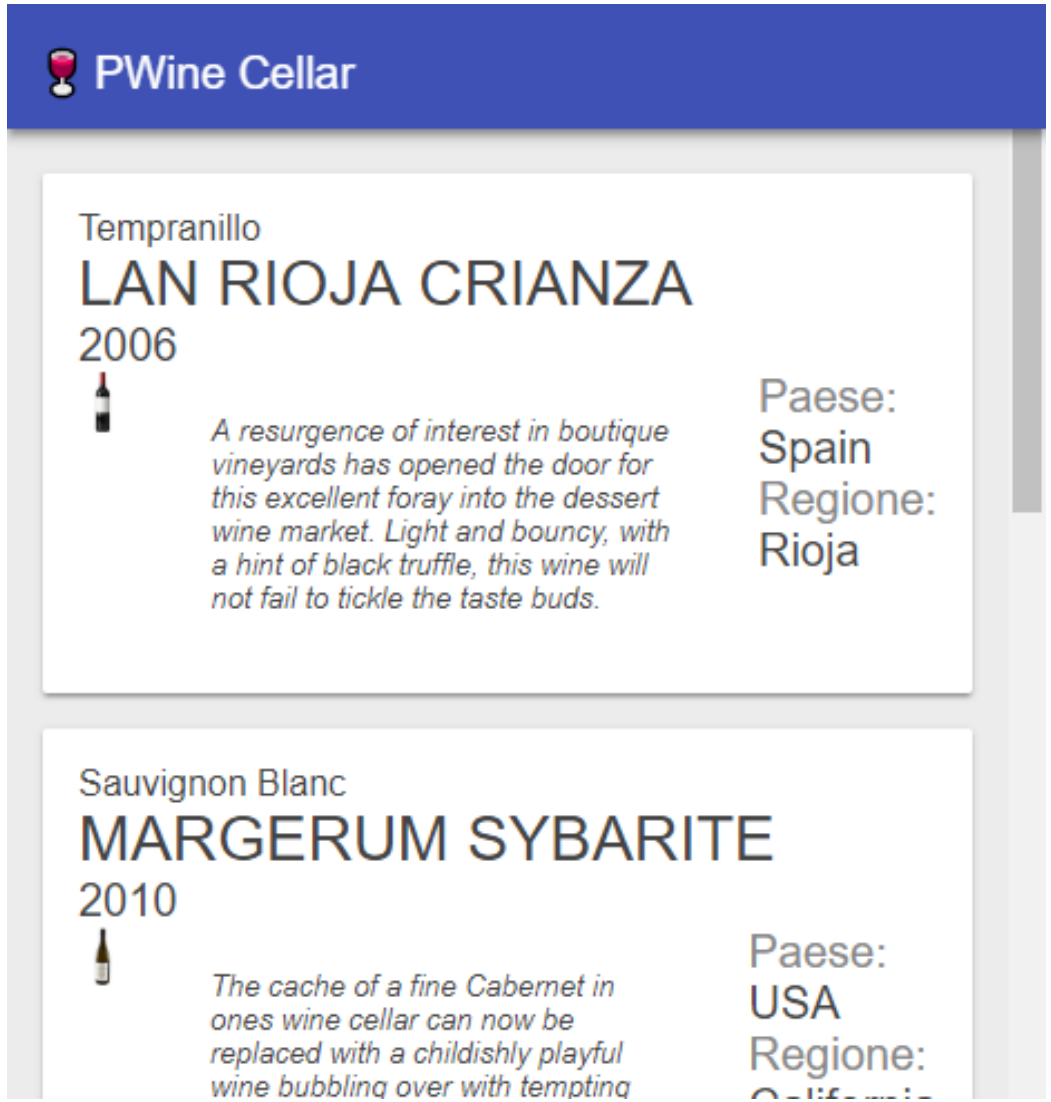
- Come applicare i principi di realizzazione delle PWA
- Come creare la shell della nostra applicazione
- Come creare un backend in Delphi
- Come scaricare e inviare dati al backand
- Gestire le interazioni dell'utente (es. per aggiungere un vino)

ATTREZZATURA

- Una versione recente del **browser Chrome**
  👉 https://www.google.com/chrome/
- Lo **starter kit** fornito da Google
  👉 https://github.com/googlecodelabs/your-first-pwapp/archive/master.zip
- Un **Web editor** ricco e funzionale (purtroppo Delphi non lo è)
  - Visual Studio Code
    👉 https://code.visualstudio.com/
  - JetBrains Web Storm
    👉 https://www.jetbrains.com/webstorm/
  - NetBeans
    👉 https://netbeans.org/
  - Atom, SublimeText, o quello che preferisci.

INGREDIENTI

- Intestazione con titolo dell'app, pulsanti di refresh e inserimento vino
- Container che ospita le card dei vini disponibili
- La card relativa al singolo vino
- Una dialog per aggiungere un nuovo vino
- Un indicatore di caricamento
- Tutto ciò che suggerisce la fantasia (per il vostro progetto)



DEMO!

"Talk is cheap, show Me the code"

- Linus Torvalds

Registrare il Service Worker

```
if ('serviceWorker' in navigator) {
  navigator.serviceWorker
    .register('./service-worker.js')
    .then(function() {
      console.log('Service Worker registered');
    });
}
```

CACHE DELLE RISORSE

```
var cacheName = 'p-wine-cellar-cache-1';
var filesToCache = [];

self.addEventListener('install', function(e) {
  console.log('[ServiceWorker] Install');
  e.waitUntil(
    caches.open(cacheName).then(function(cache) {
      console.log('[ServiceWorker] Caching app shell');
      return cache.addAll(filesToCache);
    })
  );
});
```

AGGIORNAMENTO DELLA CACHE

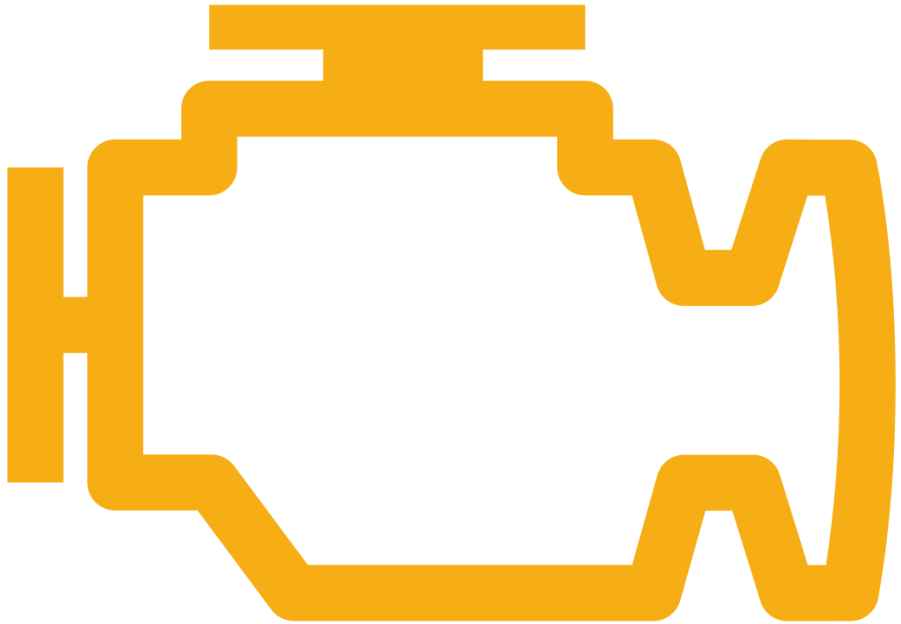
```
self.addEventListener('activate', function (e) {
  console.log('[ServiceWorker] Activate');
  e.waitUntil(
    caches.keys().then(function (keyList) {
      return Promise.all(keyList.map(function (key) {
        if (key !== cacheName) {
          console.log('[ServiceWorker] Removing old cache', key);
          return caches.delete(key);
        }
      }));
    })
  );

  return self.clients.claim();
});
```

RESTITUZIONE DI ELEMENTI DALLA CACHE

```
self.addEventListener('fetch', function(e) {
  console.log('[ServiceWorker] Fetch',
e.request.url);
  e.respondWith(
    caches.match(e.request).then(function(response)
{
      return response || fetch(e.request);
    })
  );
});
```

ATTENZIONE!

- L'aggiornamento della cache dipende dalla chiave
- Ogni volta vengono riscaricati nuovamente tutti i file
- Se il Service Worker va in cache non viene aggiornato
- Il codice non controlla se è disponibile una versione nuova della risorsa online



👉Usare tool come **Workbox**! (https://developers.google.com/web/tools/workbox/)

Aggiungi un "Manifest"

- Aggiungi un file "manifest.json" alla tua app per
  - Creare una icona nella home screen dell'utente
  - Lanciare l'applicazione in "full mode"
  - Controllare l'orientamento dello schermo per una visualizzazione ottimale
  - Sfruttare lo "splash screen" al momento dell'avvio
  - Personalizzare i colori dell'applicazione

E' possibile identificare se l'app viene lanciata dalla home oppure dal browser tramite il relativo URL!

TEST DELLE APPLICAZIONI

Chrome Lighthouse

- E' un tool open source
- Esegue test automatizzati su pagine e applicazioni Web
- Aiuta a incrementare le performance, la qualità e il buon funzionamento delle app
- Genera un report dei risultati da utilizzare per misurare i benefici delle correzioni e delle migliorie apportate

About me

Marco Breveglieri

Software Developer, Trainer and Consultant

@ABLS Team (Reggio Emilia)

- Homepage: www.breveglieri.it
- Blog: www.compilaquindiva.com
- Podcast: www.delphipodcast.com

Proud member of: DELPHI CLUB ITALIA

Q & A


ONE DOES NOT SIMPLY
GIVE A PRESENTATION WITHOUT Q&A
imgflip.com

GRAZIE!