Service Workers at Your Service

Workshop

Michal Gregor

Martin Farkaš

10/31/2018

About Us

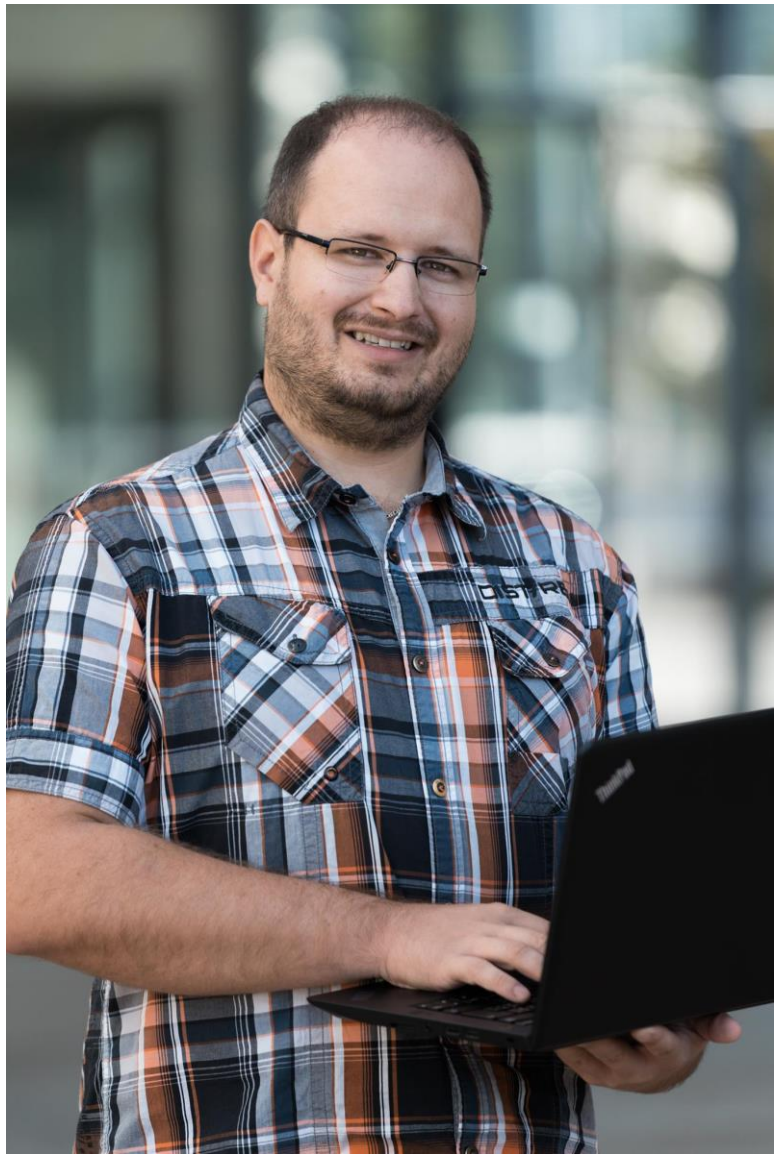
UNICORN

With Unicorn's IT solutions and services our customers gain an edge by exploiting the potential of modern IT technologies and innovations to support their business.

- Custom Solutions
- Vertical Solutions
- Skills & Resources
- Banking & Insurance
- Energy & Utilities
- Manufacturing & Trade

More than **300** EUROPEAN INDUSTRY LEADING COMPANIES trust **Unicorn** as their long-term reliable IT partner thanks to its **flexibility and capability** to deliver.

Michal



Software engineer, full-stack developer and an evangelist at Unicorn.

Always happy to help his co-workers with coding and architectural problems and issues, he dedicates a lot of his time to teaching others through mentoring and coaching activities both at Unicorn and at the University.

When he is not in the IT realm, he enjoys a good film or book, swimming and walking with a camera.

Martin

Software engineer at Unicorn where he has worked on numerous projects in banking, energy and insurance over the last 10 years.

He has recently been working as an evangelist helping developers transition to React, NodeJS, .NET Core and microservice architecture.

He spends his spare time with his two daughters and his wife while attempting to renovate their century-old house.

Concept

UNICORN

Workshop is about …

- What the hell are **Service Workers**?
- How can it help me to improve user experience?
- How to plug it into my app?
- How to debug issues?

You will need …

- 3 hours of your time
- Notebook + Internet
- **Chrome v69+**
  - https://www.google.com/chrome/
- **Node.js v8.12+**
  - https://nodejs.org
- **Yarn** v1.10+
  - https://yarnpkg.com
- **VS Code** / **Webstorm** is recommended
  - https://code.visualstudio.com/
  - https://www.jetbrains.com/webstorm/
- To know **JavaScript Promises & Basics of React**
  - https://developers.google.com/web/fundamentals/primers/promises
  - https://reactjs.org/tutorial/tutorial.html

Timeline

- Some really necessary theory [30m]
- Environment preparation [15m]
- Exercise 1 – Add SW to app [15m]
- Exercise 2 – SW lifecycle [15m]
- Exercise 3 – Static cache [15m]

- **Plus4U Coffee break** [15m]

- Exercise 4 - Dynamic cache [25m]
- Exercise 5 - Offline fallback [10m]
- Exercise 6 – Notification [15m]
- Exercise 7 – Workbox* [10m]
- Discussion [15m]

Theory

UNICORN

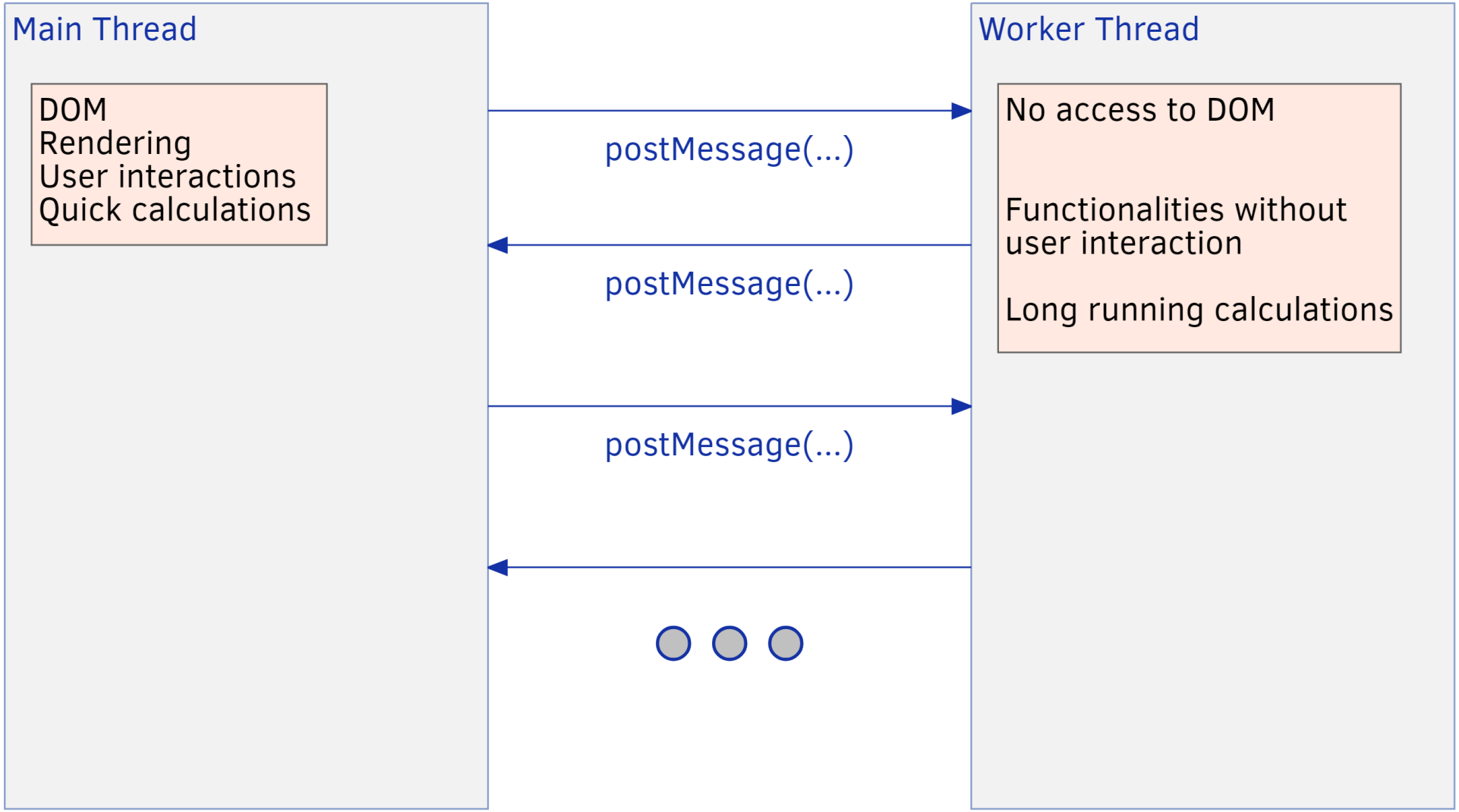
Progressive Web App is …

- Pages are **responsive** on tablets & mobile devices
- All app URLs load while **offline**
- **Metadata** provided for Add to Home screen
- First load **fast** even on 3G
- Site works **cross-browser**
- Page transitions don't feel like they block on the network
- Each page has a URL
- Site uses **cache-first** networking

In short, **PWA is trying to look as native app.**

See more criteria on the official Google PWA Checklist
https://developers.google.com/web/progressive-web-apps/checklist

Web Worker is …



Service Workers is …

- …W3C standard
- …**JS script** running in the background of browser as **JS Worker**
- …programmable **network proxy**
- …for functionalities **without user interactions**
  - Caching and offline-first apps
  - Push notifications
  - Background sync
- …**separated from a web page**
  - NO direct access to DOM
  - Communication via postMessage interface
- …**terminated when not in use**, and restarted when it's next needed.
  - Global state has to pe persisted (e.g. via IndexedDB API)
- …substitution for AppCache API

CacheStorage is …

- New API for cache management
- Defined in Service Workers standard
- Can be used without Service Workers
- Can hold multiple caches with unique name
- Each cache holds key-value pairs
- Available methods are
  - **open()**
  - **has()**
  - **match()**
  - **delete()**
  - **keys()**

Cache is

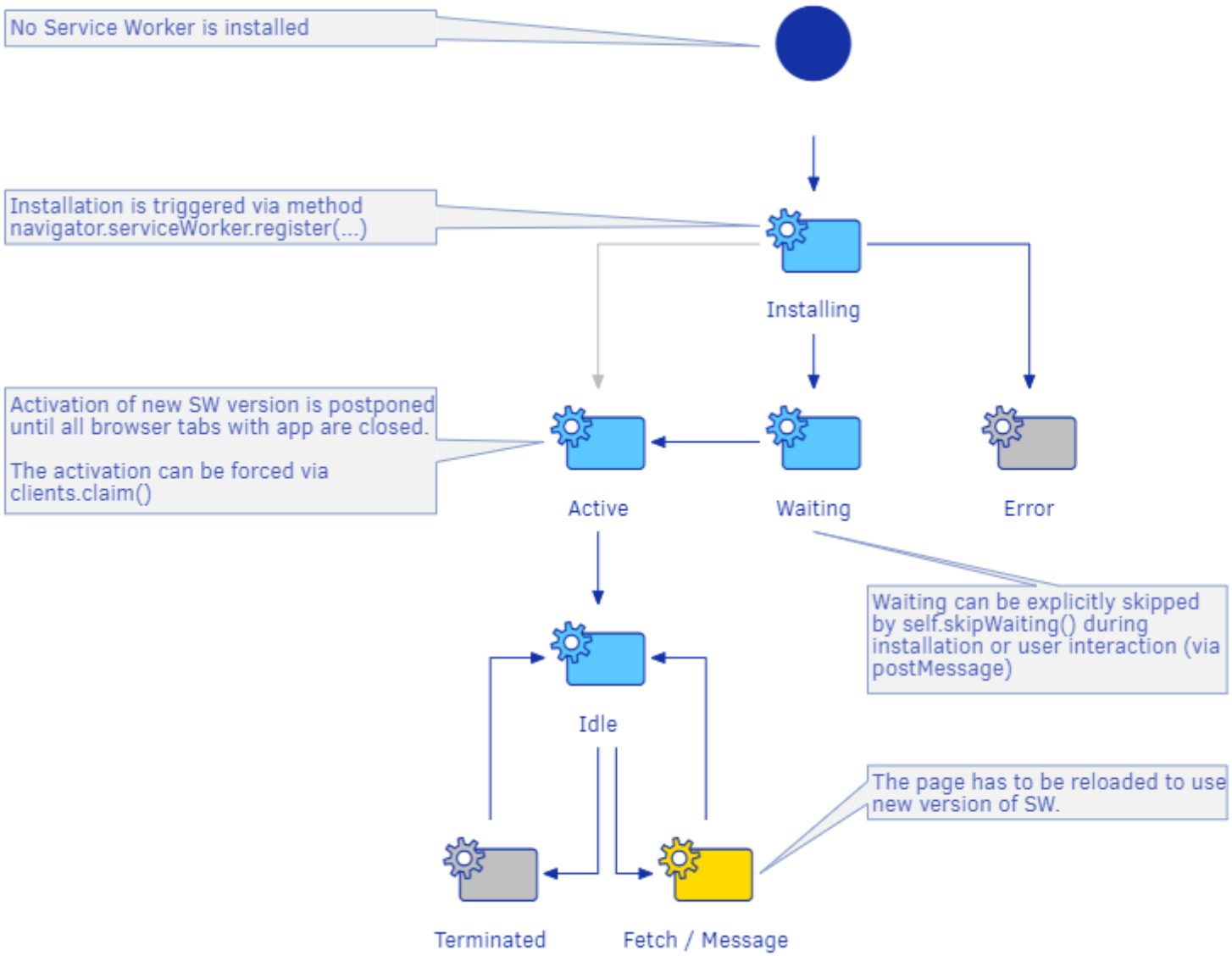
- Part of CacheStorage
- It is specific Cache in CacheStorage
- Defined by unique name
- Everything is Promise
- Available methods are
  - **add()**
  - **addAll()**
  - **delete()**
  - **keys()**
  - **match()**
  - **matchAll()**
  - **put()**

Browsers Support

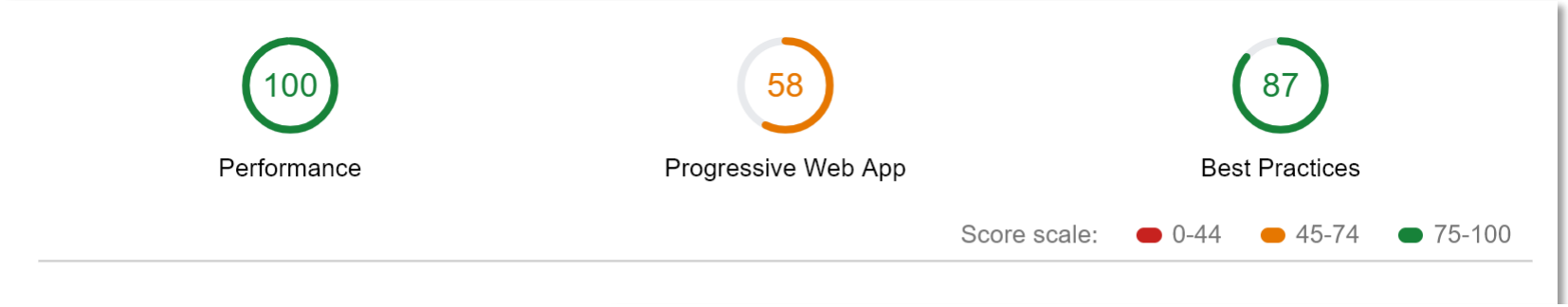
- Supported by modern browsers
  - Chrome 49+
  - Firefox 61+
  - Safari 11.1+*
  - Edge 17+
  - Chrome for Android 69+
  - UC Browser for Android 11.8+
  - Samsung Internet 4+
- **It is usable!**
- More details on
  - https://jakearchibald.github.io/isserviceworkerready/
  - https://caniuse.com/#search=service%20workers

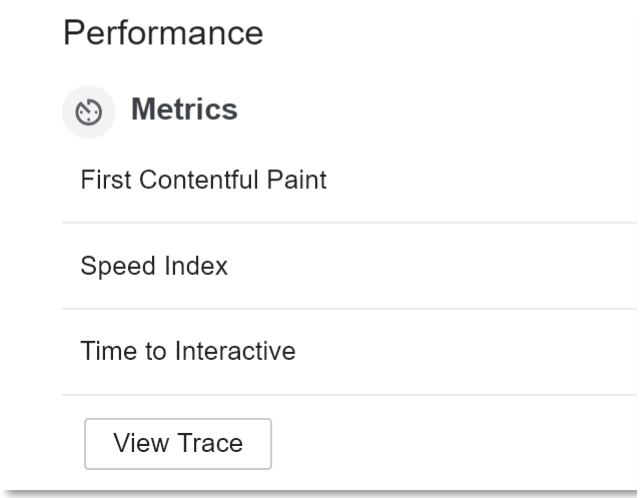
- * e.g. cache restrictions, no push notifications

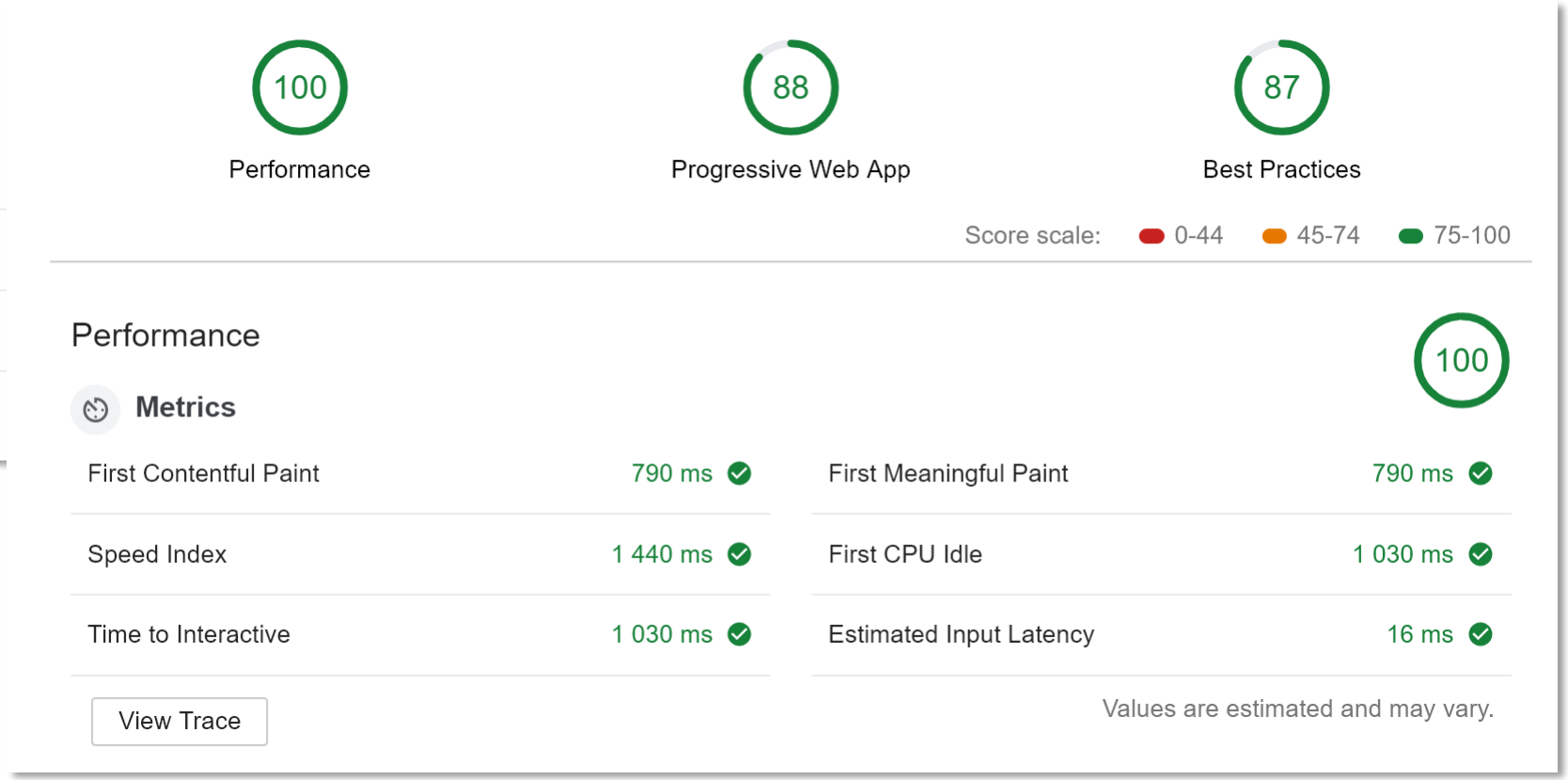
Lifecycle



Google Audits / Lighthouse





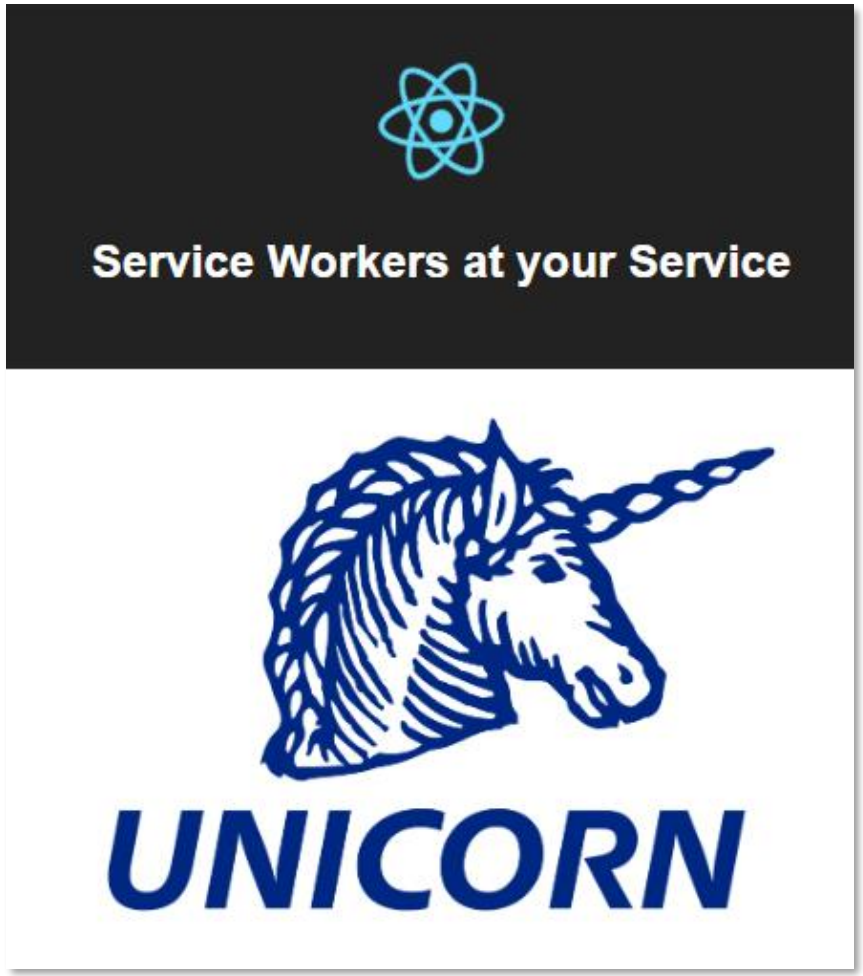


Environment

UNICORN

Setup

- Clone or download repository
  - git clone https://github.com/UnicornUniverse/reactiveconf-service-worker
  - (https://bit.ly/2OkBnAk)
- Open root folder in IDE
- Open terminal in IDE
  - yarn
  - cd client
  - yarn
  - cd ..
  - yarn dev
- Check app is running on localhost:3000

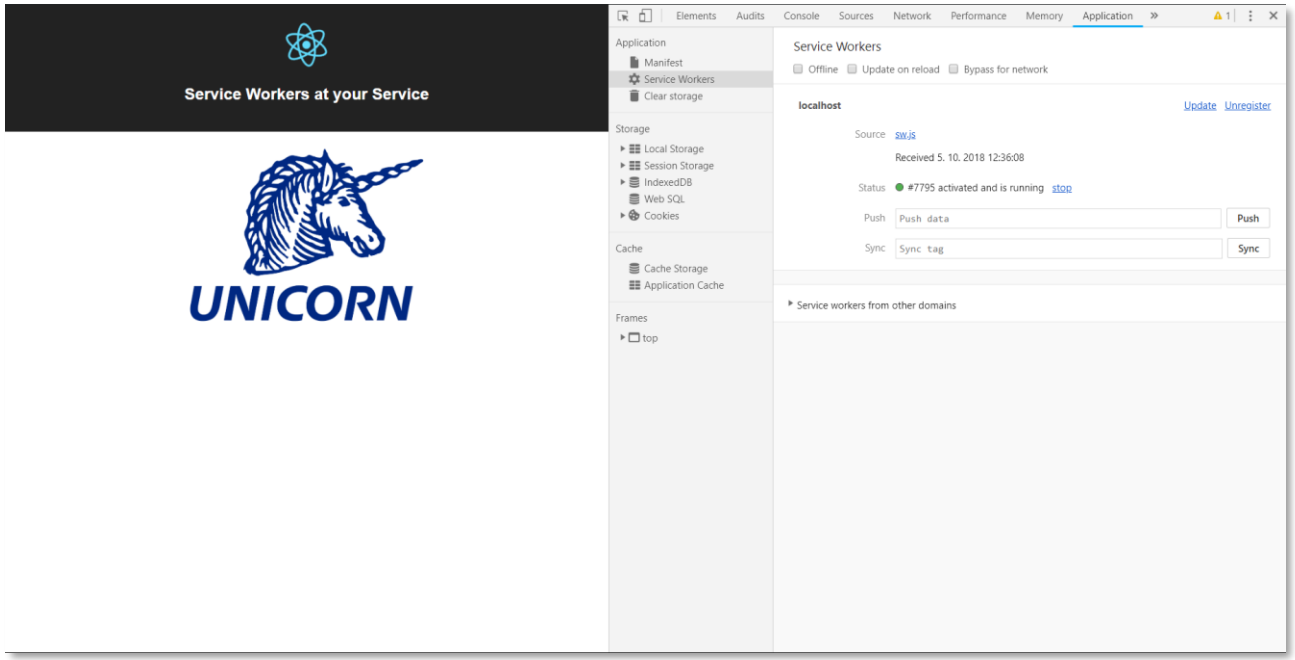


Exercises

UNICORN

Exercise 1 – Add Service Worker (SW) to App

1. Open file */client/src/index.js*
   - Uncomment SW registration
2. Open file /cliens/src/registerServiceWorker.js
   - Check code of SW registration
3. Open file */client/public/sw.js*
   - Check import of sw-ex-1.js
4. Open file /client/public/sw-ex-1.js
   - Check code of SW
5. Run & open app on localhost:3000
6. Open Developer Tools (F12)
7. Show tab Application
   - Check SW is activated and running
8. Show tab Sources
   - Check running threads of app (2)



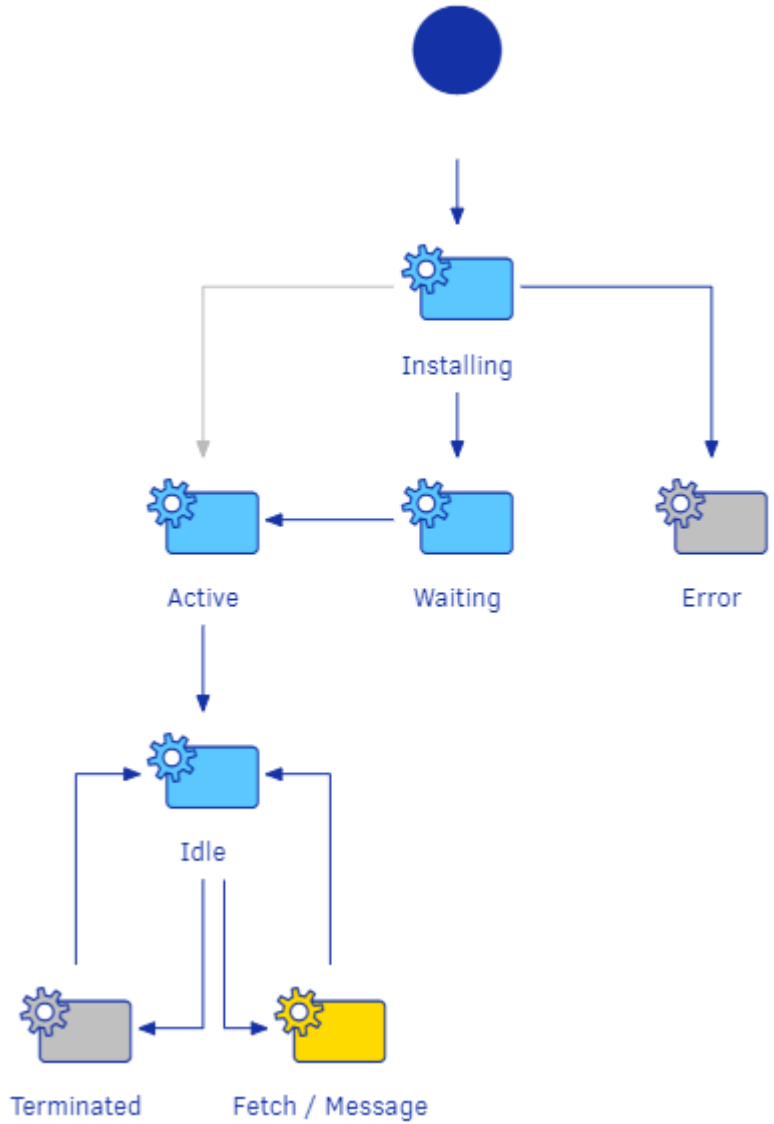
Exercise 1 – Registration of SW vs. onLoad event

Name	Initiator	Size	Timeline – End Time
localhost	Other	5.0KB	
all-3f1def857d.css	(index):18	1.2KB	
service-worker.js	service-wo…	0B	
all-3f1def857d.css	Other	(from disk cache)	
shell	Other	867B	
icon.png	Other	12.8KB	
app-604ca22560.js	(index):28	3.6KB	
app-604ca22560.js	Other	(from disk cache)	
third-party-af348c43d…	Other	86.7KB	
urCpKyUrZPrCeFdx.sta…	(index):22	11.6KB	
qWOHtBmCQpUvHckS.…	(index):22	12.1KB	

GHFgbZVLevXEQr1M.st…	(index):17	28.6KB	
utE62K2XClvs1KRx.stan…	(index):17	31.9KB	
third-party-af348c43d8.js	(index):22	86.7KB	
service-worker.js	service-wo…	0B	
app-604ca22560.js	Other	(from disk cache)	
all-3f1def857d.css	Other	(from disk cache)	
third-party-af348c43d…	Other	(from disk cache)	
icon.png	Other	12.8KB	
icon.png	Other	(from disk cache)	
shell	Other	867B	

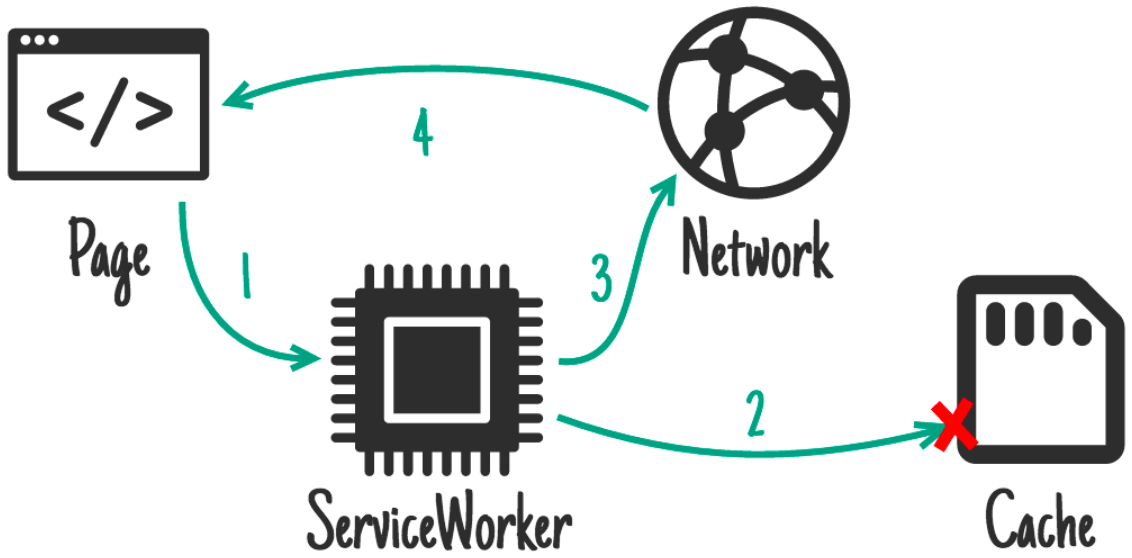
Exercise 2 – SW Lifecycle

1. Open file /client/public/sw.js
   1. Comment out sw-ex-1.js
   2. Uncomment sw-ex-2.js
2. Run & open app on localhost:3000
3. Check picture (UNI)
4. Open second tab and check picture (UNI Systems)
5. Update sw-ex-2.js and change pictures.
6. Open third tab and check picture (UNI Systems)
7. Check threads in Source tab of Dev Tools (3)
8. Close all tabs with app to unregister old SW
9. Open tab with app and check picture (UNI Elite)
10. Press CTRL + F5 and check picture (UNI)



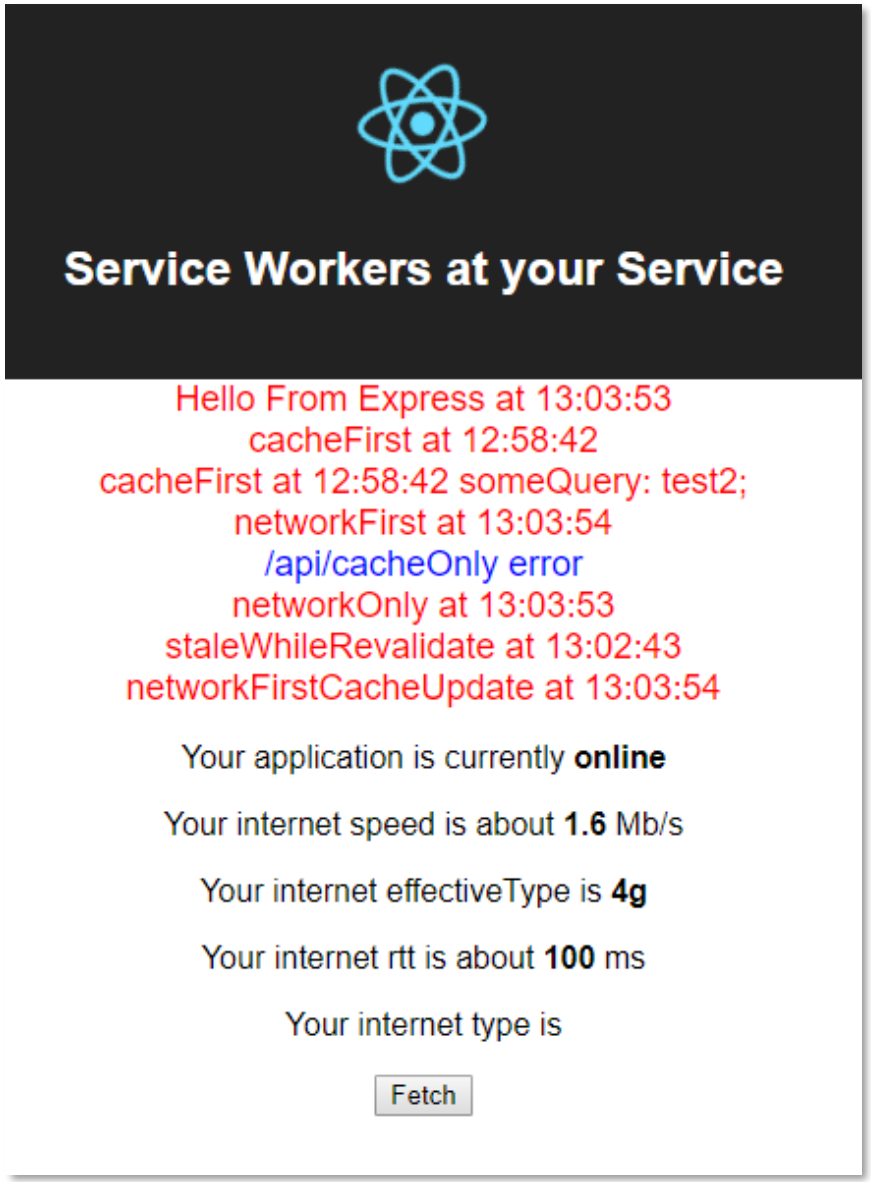
Exercise 3 – Static cache

1. Open file /client/public/sw.js
   1. Comment out sw-ex-2.js
   2. Uncomment sw-ex-3.js
2. Run & open app on localhost:3000
3. Open Developer Tools with tab Application
4. Check Cache
5. Turn connection of and refresh app
6. Check app is running in offline
7. Change app version in sw-ex-3.js
8. Add some dummy comment to sw.js
9. Skip waiting of SW in Dev Tools
10. Refresh page and check cache



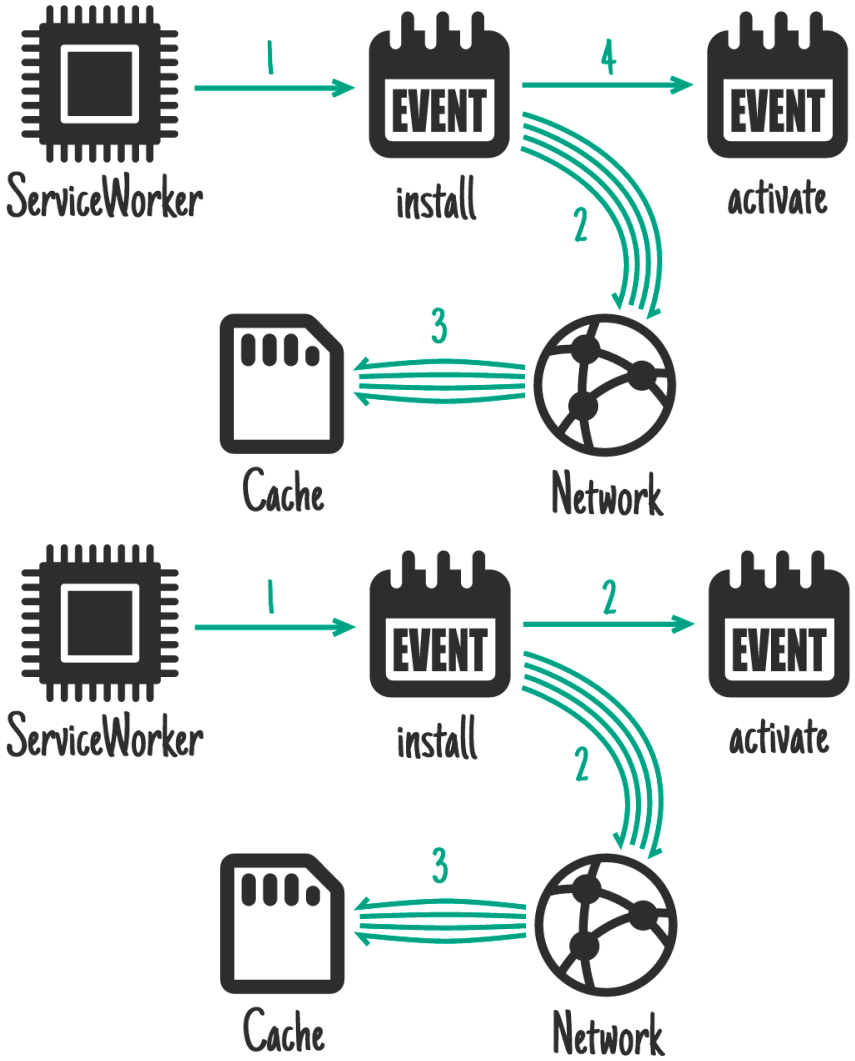
Exercise 4 – Dynamic cache

1. Open file /client/public/sw.js
   1. Comment out sw-ex-3.js
   2. Uncomment sw-ex-4.js
2. Open file /client/src/App.js
   1. Comment out <Ex2/>
   2. Uncomment <Ex4/>
3. Run & open app on localhost:3000
4. Open Network tab in Dev Tools
5. Click Fetch button
6. Check behavior of cached resources based on cache strategy

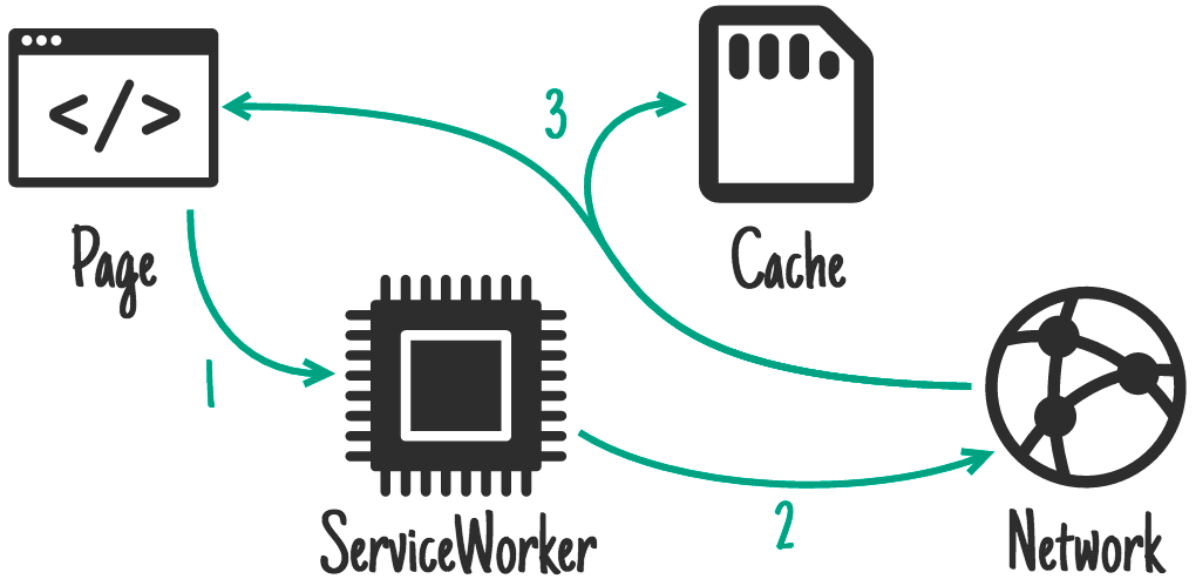


Exercise 4 – Adding to cache

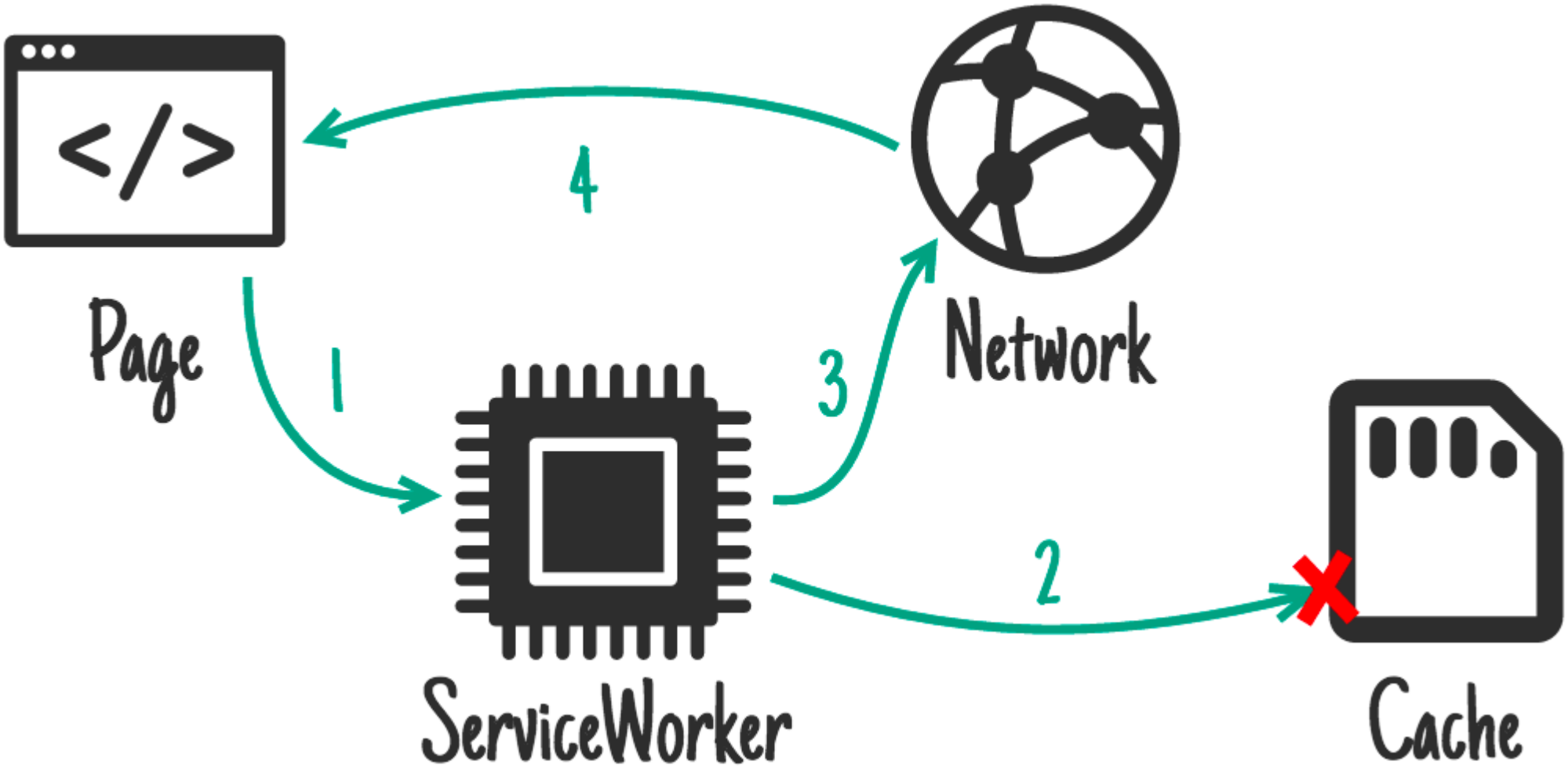
- On install



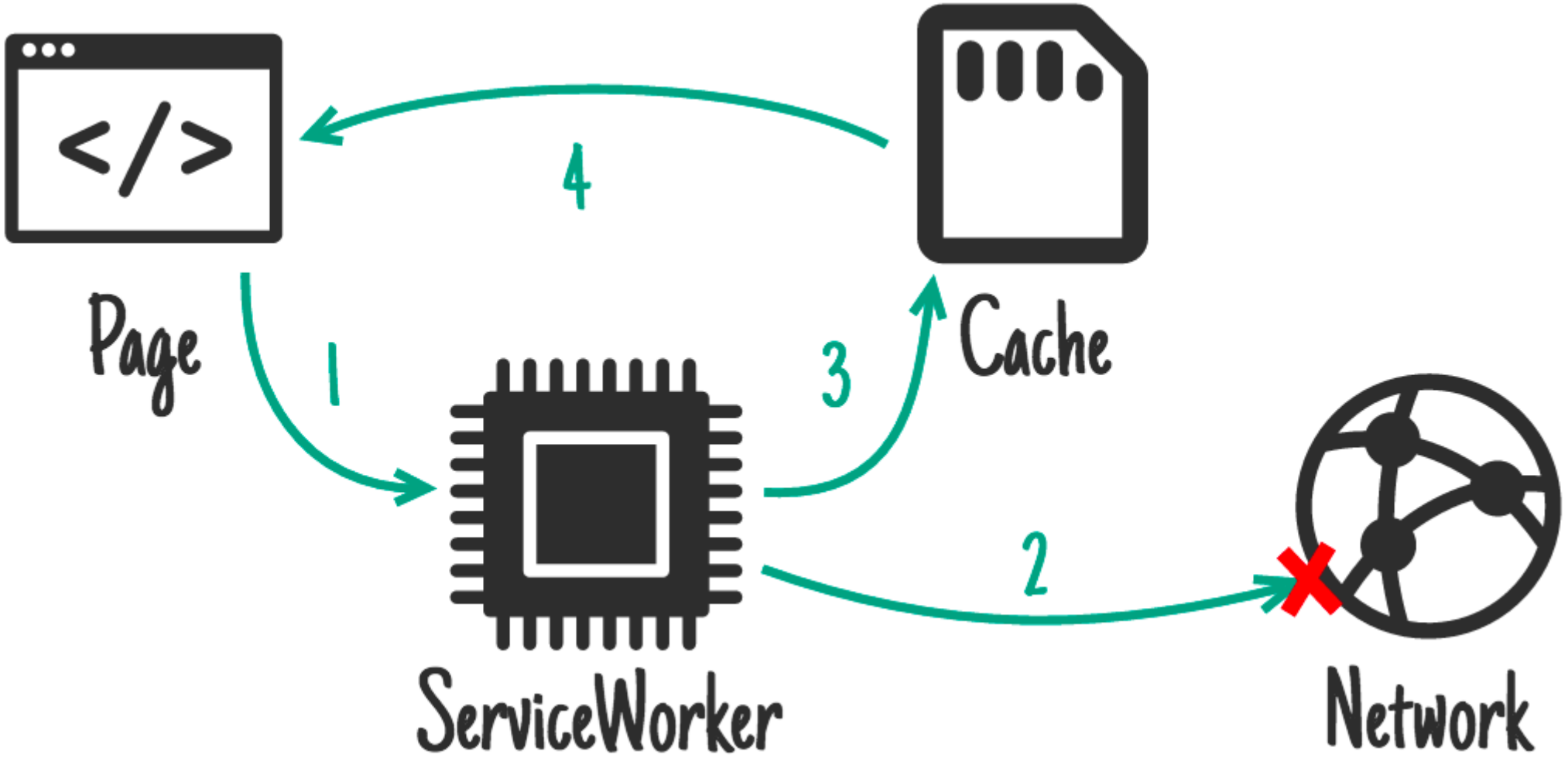
- On network response



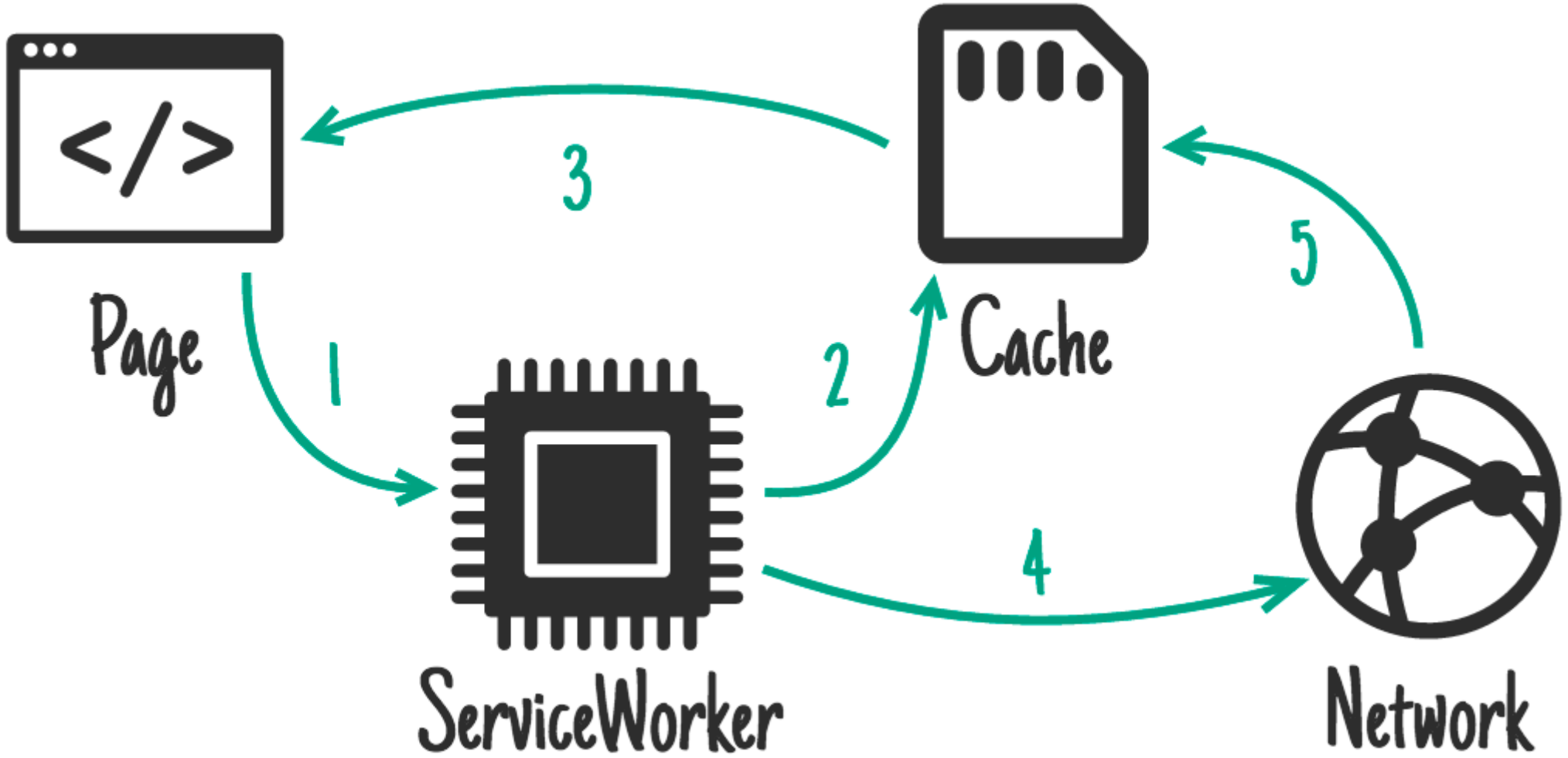
Exercise 4 – Cache-first



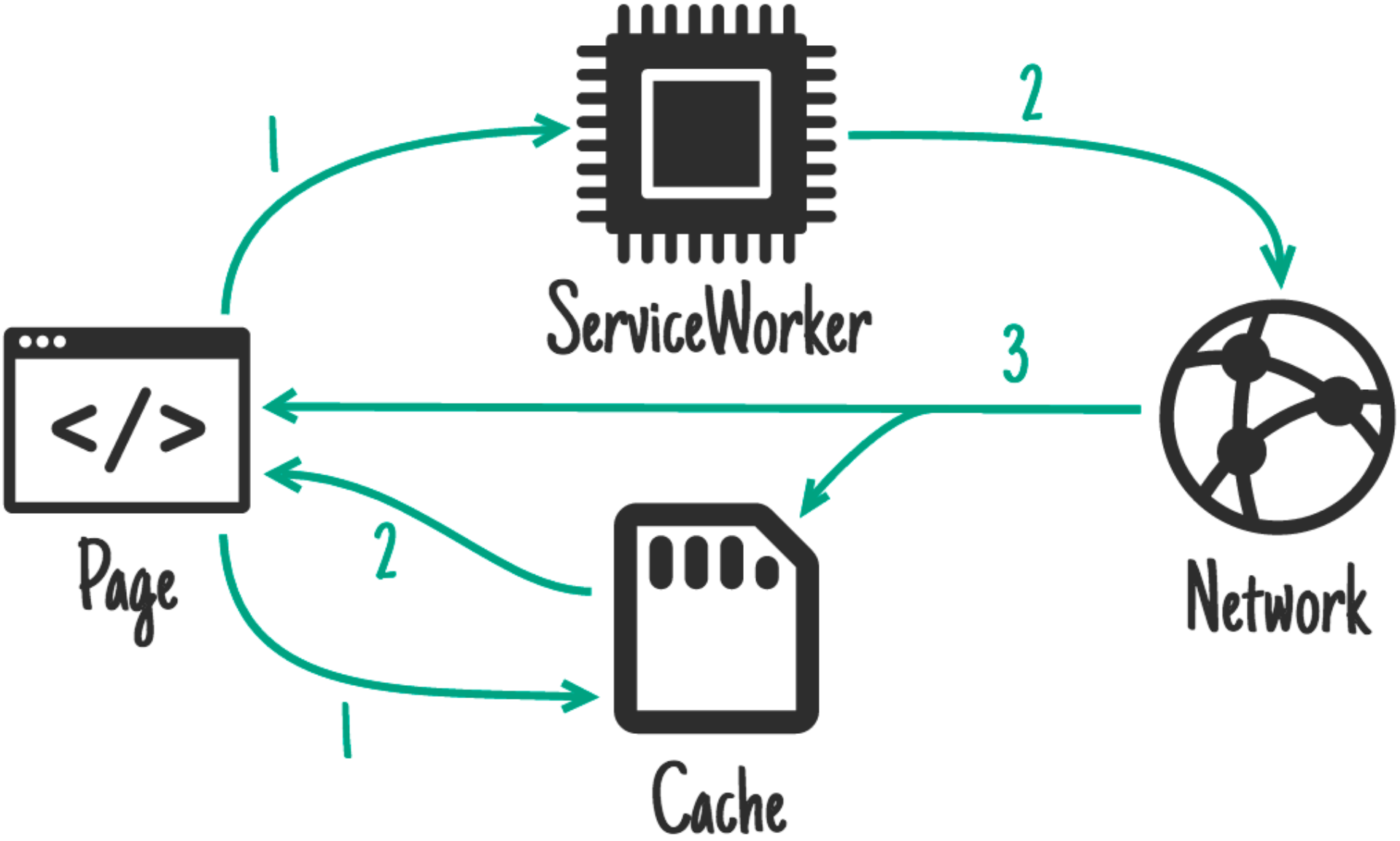
Exercise 4 – Network-first



Exercise 4 – Stale-while-revalidate

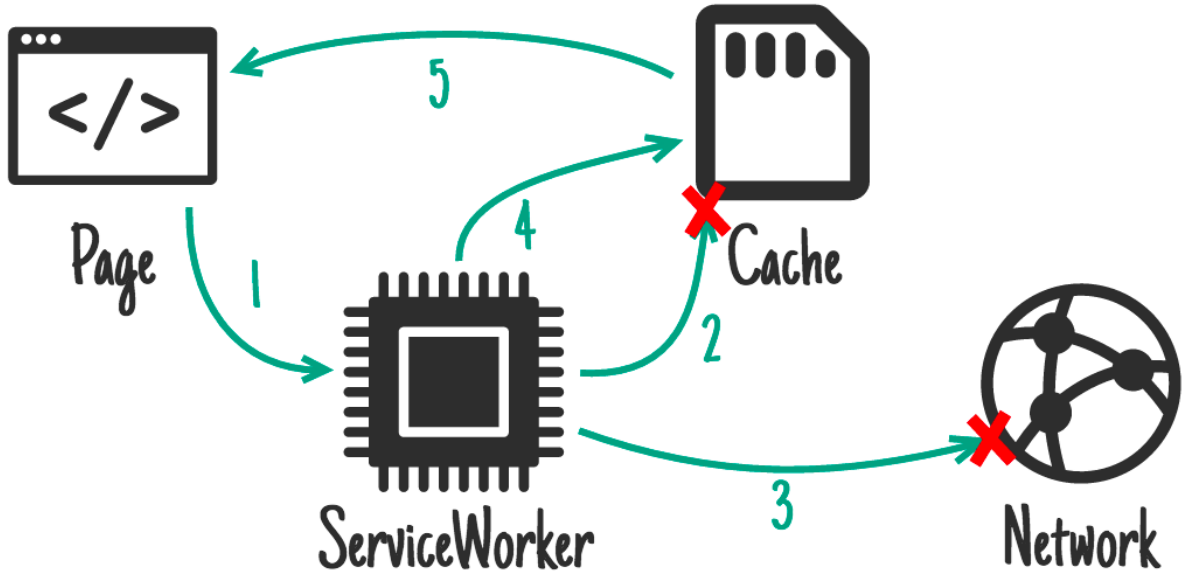


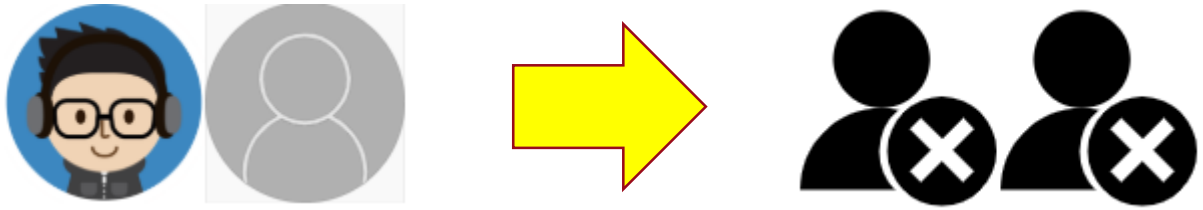
Exercise 4 – Cache & Update



Exercise 5 – Offline fallback

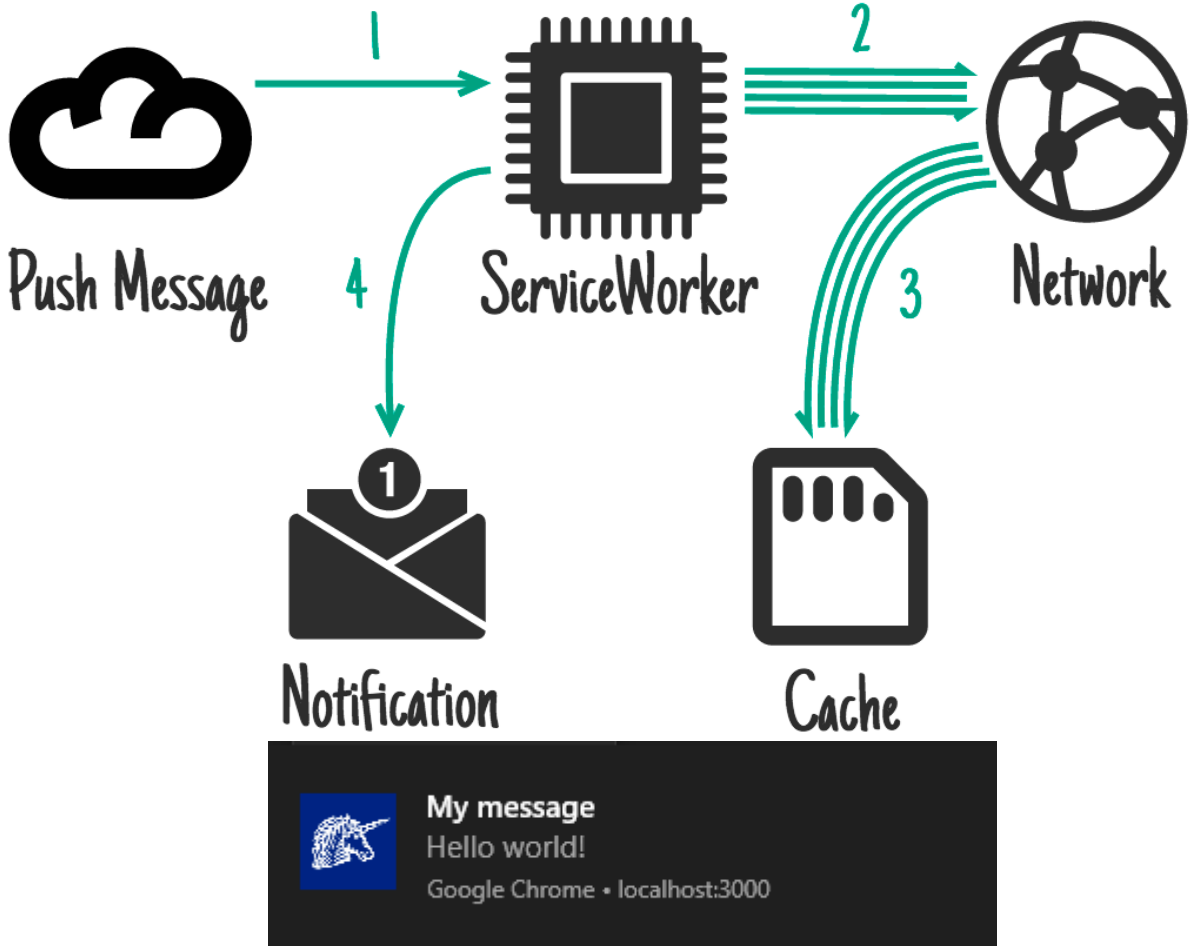
1. Open file /client/public/sw.js
   1. Comment out sw-ex-4.js
   2. Uncomment sw-ex-5.js
2. Open file /client/src/App.js
   1. Comment out <Ex4/>
   2. Uncomment <Ex5/>
3. Run & open app on localhost:3000
4. Check the image for missing source
5. Open Network tab in Dev Tools
6. Turn app to offline mode
7. Check the images for the offline



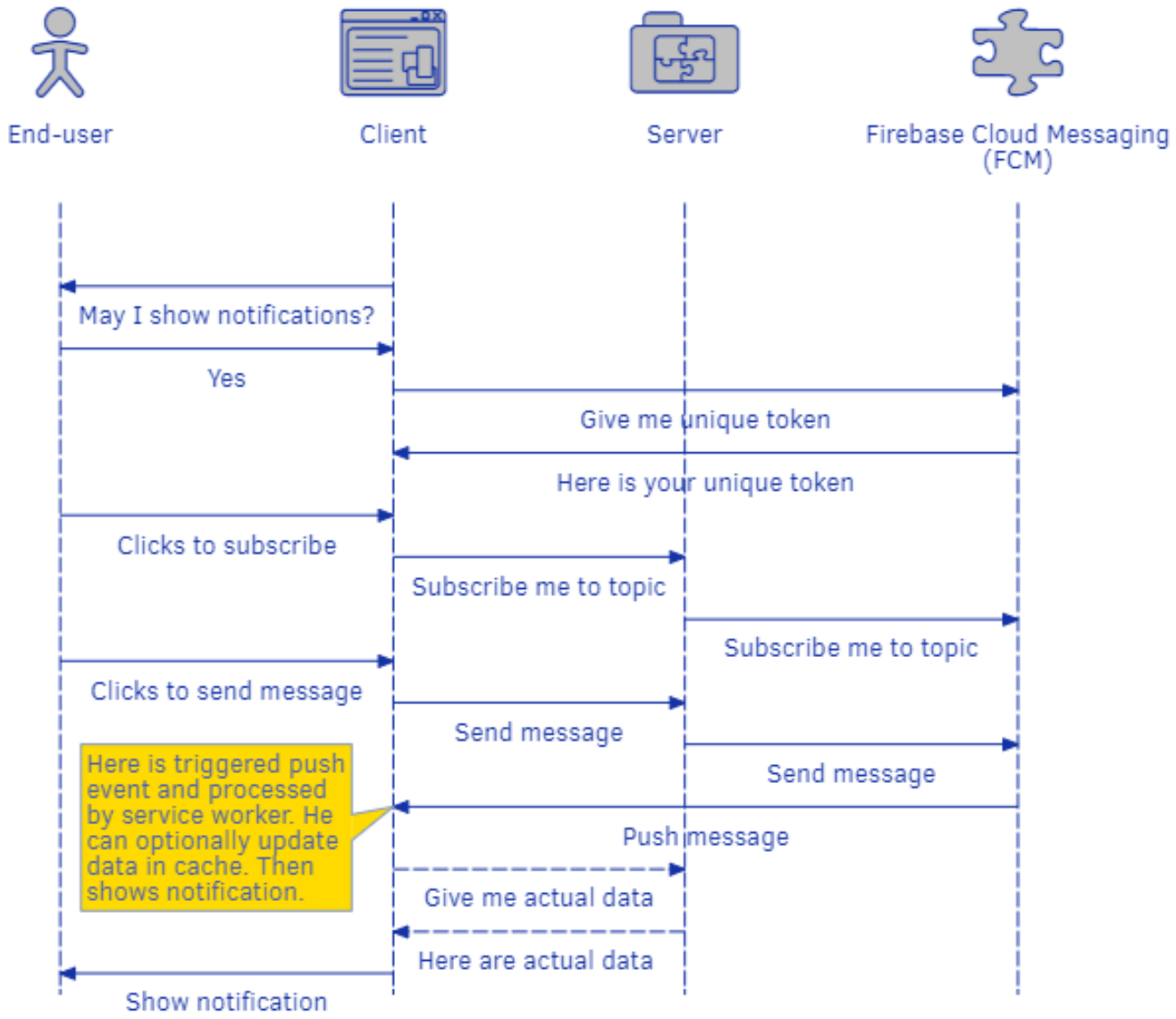


Exercise 6 – Notification

1. Open file /server/serviceAccountKey.js
   - Repair project_id, private_key_id and private_key
- Open file /server/server.js
   - Uncomment line 16 (//credential…)
1. Open file /client/public/sw.js
   - Comment out sw-ex-5.js
   - Uncomment sw-ex-6.js
2. Open file /client/src/App.js
   1. Comment out <Ex5/>
   2. Uncomment <Ex6/>
3. Run & open app on localhost:3000
4. Check token is generated
5. Fill title and message
6. Click PUSH MESSAGE TO ME
7. Click REGISTER TO TOPIC
8. Click PUSH MESSAGE TO TOPIC
9. Check notification bar of your OS / Browser



Exercise 6 – Behind the scene



Exercise 7 – Workbox

Workbox

Caveats

- By default, a page's fetches won't go through a service worker unless the page request itself went through a service worker. So you'll need to **refresh the page** to see the effects of the service worker.
- The default scope of a service worker registration is **./ relative to the script URL**.
- Service worker is considered **updated if it's byte-different** to the one the browser already has.
- You may **not be updating from the previous version**. It may be a service worker many versions old.
- Be careful that you **don't delete caches for your other sites**.
- **Don’t change the URL** of service worker!
- **Delay** service worker's **initial registration** until after the first page has loaded.
- Service worker requires **HTTPS**.

Discussion

UNICORN

Topics…

- Service-Worker-Allowed header
- Cache on Demand
- Message Relay
- Request Defferer
- Background Sync

Resources

- https://developers.google.com/web/fundamentals/primers/service-workers/
- https://developers.google.com/web/fundamentals/instant-and-offline/offline-cookbook/
- https://jakearchibald.github.io/isserviceworkerready/
- https://codelabs.developers.google.com/codelabs/debugging-service-workers/index.html