Google Developer Student Clubs

PWA: Budowanie aplikacji działającej w trybie offline

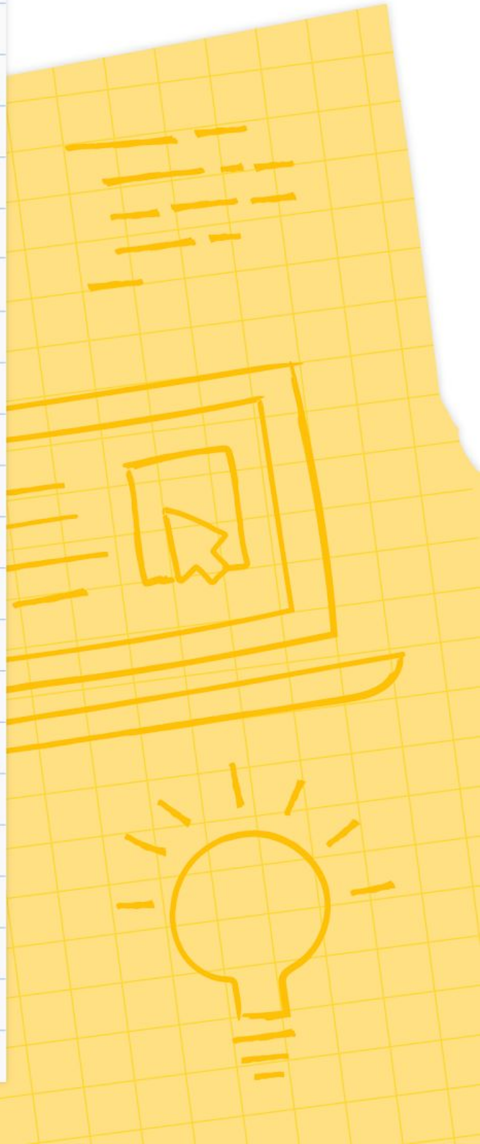
Warsztaty

Krzysztof Kaczyński

Google Developer Student Clubs

Agenda

- Co to znaczy, że aplikacja jest PWA
- Co to jest webmanifest
- Co to jest service worker?
- Cykle życia Service Workera
- Aktualizacja service worker’a
- Cachowanie i strategie serwowanie plików
  - Cache First
  - Network First
  - Stale While Revalidate
  - Network-Only
  - Cache-Only



Co to znaczy, że aplikacja jest PWA

PWA - Progressive Web Application

- Szybkie działanie
- Działa na wielu platformach
- Responsywny design
- Działa offline
- Niestandardowa strona offline pasująca do całej aplikacji
- Można ją zainstalować

Lista specyfikacji, którymi powinna charakteryzować się aplikacja PWA:
What makes a good Progressive Web App?

Co to jest webmanifest

Jest to plik zawierający metadane na temat aplikacji niezbędne do tego aby można było zainstalować aplikacje.

Wszystkie dostępne pola w webmanifest wraz z opisem co robią możesz znaleźć tutaj: [MDN - Web app manifests]

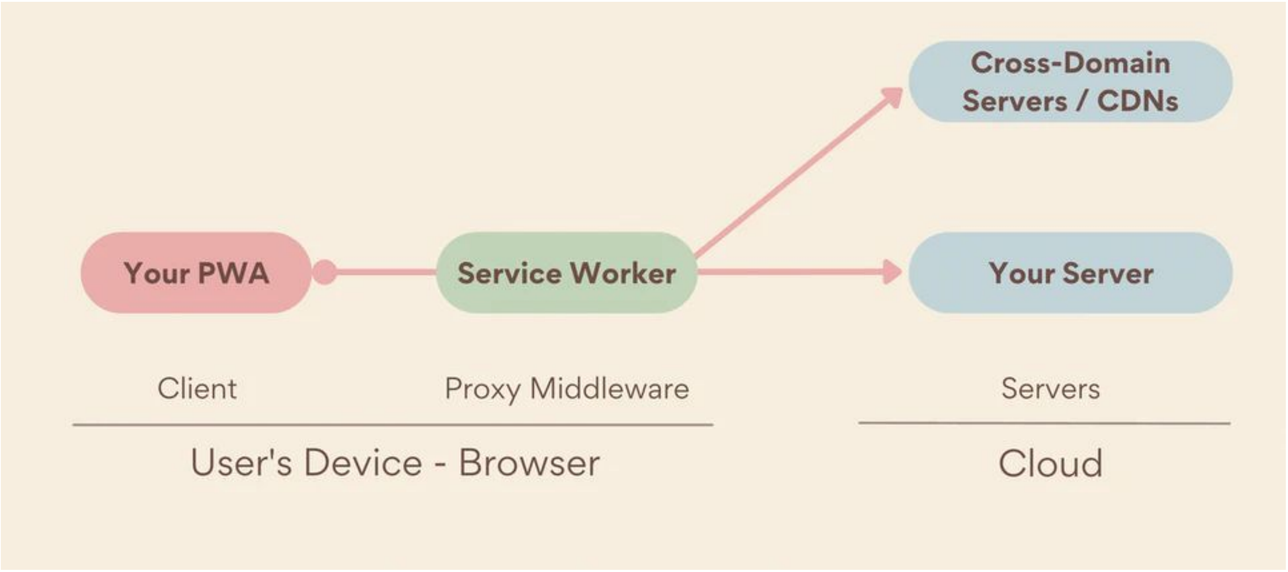
Google Developer Student Clubs

```
{
 "$schema": "https://json.schemastore.org/web-manifest-combined.json",
 "start url": ".",
 "lang": "en",
 "dir": "auto",
 "name": "Pictures - List",
 "short name": "PicLi",
 "description": "Simple app created for gdsc pwa workshops",
 "display": "standalone",
 "orientation": "natural",
 "theme color": "#2d2d2d",
 "background color": "#2d2d2d",
 "categories": ["education", "pictures"],
 "icons": [{
   "src": "assets/homescreen/homescreen48.png",
   "sizes": "48x48",
   "type": "image/png",
   "purpose": "maskable any"
 },
   {
   "src": "assets/homescreen/homescreen72.png",
   "sizes": "72x72",
   "type": "image/png",
    "purpose": "maskable any"
 }, {
   "src": "assets/homescreen/homescreen96.png",
   "sizes": "96x96",
   "type": "image/png",
   "purpose": "maskable any"
 }, {
   "src": "assets/homescreen/homescreen144.png",
   "sizes": "144x144",
   "type": "image/png",
   "purpose": "maskable any"
 }, {
   "src": "assets/homescreen/homescreen168.png",
   "sizes": "168x168",
   "type": "image/png",
   "purpose": "maskable any"
 }, {
   "src": "assets/homescreen/homescreen192.png",
   "sizes": "192x192",
   "type": "image/png",
   "purpose": "maskable any"
 }, {
   "src": "assets/homescreen/homescreen512.png",
   "sizes": "512x512",
   "type": "image/png",
   "purpose": "maskable any"
 }]
}
```

Co to jest service-worker

Dostarczy wszystkie potrzebne dane offline

Jest to proxy pozwalające kontrolować przepływ zapytań sieciowych. Bardzo upraszczając można by powiedzieć, że service worker jest czymś w rodzaju middleware, który może odsyłać nam informacje dla wcześniej ustawionych zapytań



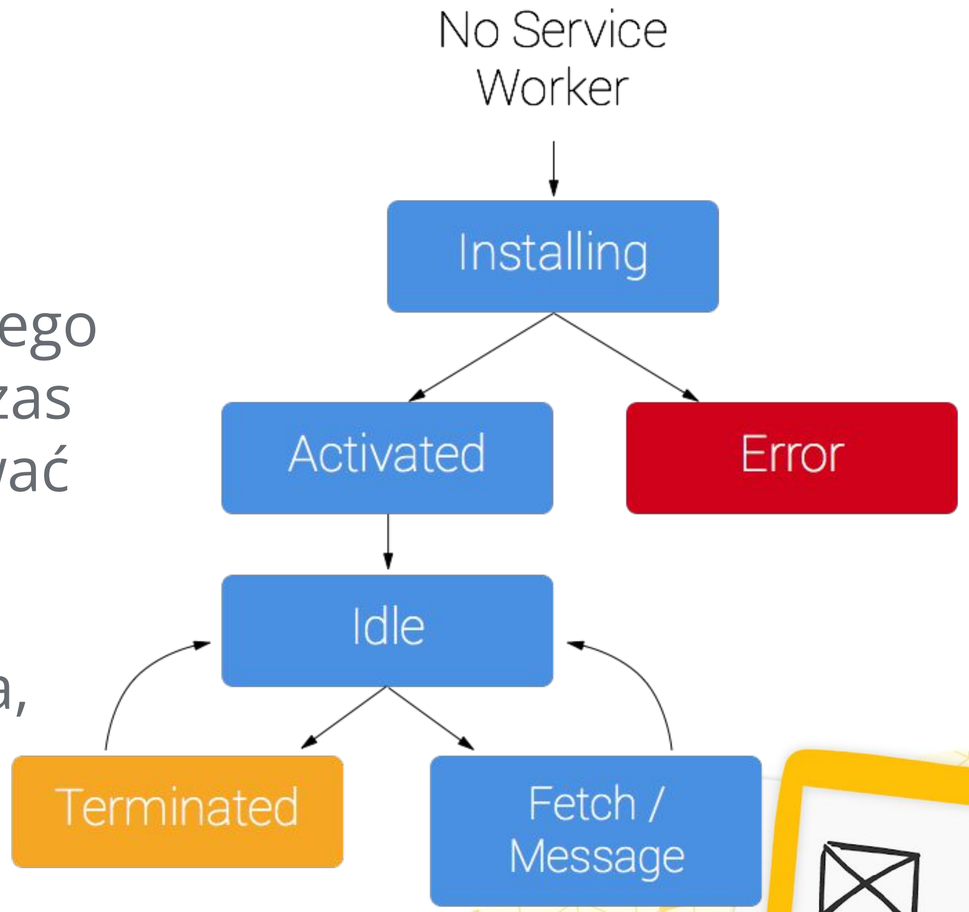
KORZYSTANIE Z SERVICE WORKERA JEST MOŻLIWE TYLKO NA LOCALHOST I PO HTTPS

Cykle życia Service Worker’a

Rejestracja -> wykonuje ją główny wątek za pierwszym razem, kiedy użytkownik odwiedzi aplikację

Instalacja -> wywołuje się tylko raz dla konkretnego sw, aż dopóki nie zostanie zaktualizowany. Podczas instalacji oceniamy też jakie pliki chcemy serwować przez sw

Aktywacja -> sw jest aktywny i kontroluje klienta, może serwować zapamiętane zasoby, wysłać notyfikacje ...



Google Developer Student Clubs

Aktualizacja Service Workera

- Event **push**
- Event **sync**
- Wywołanie navigator.serviceWorker.register()
- Użytkownik zmienia nawigacje w zakresie sw
- Servic Worker zmienił swój kod
- Aktualizacja została wymuszona przez programistę aplikacji

```
navigator.serviceWorker.ready.then((registration) => void registration.update());
```


Obsługiwanie aktualizacji

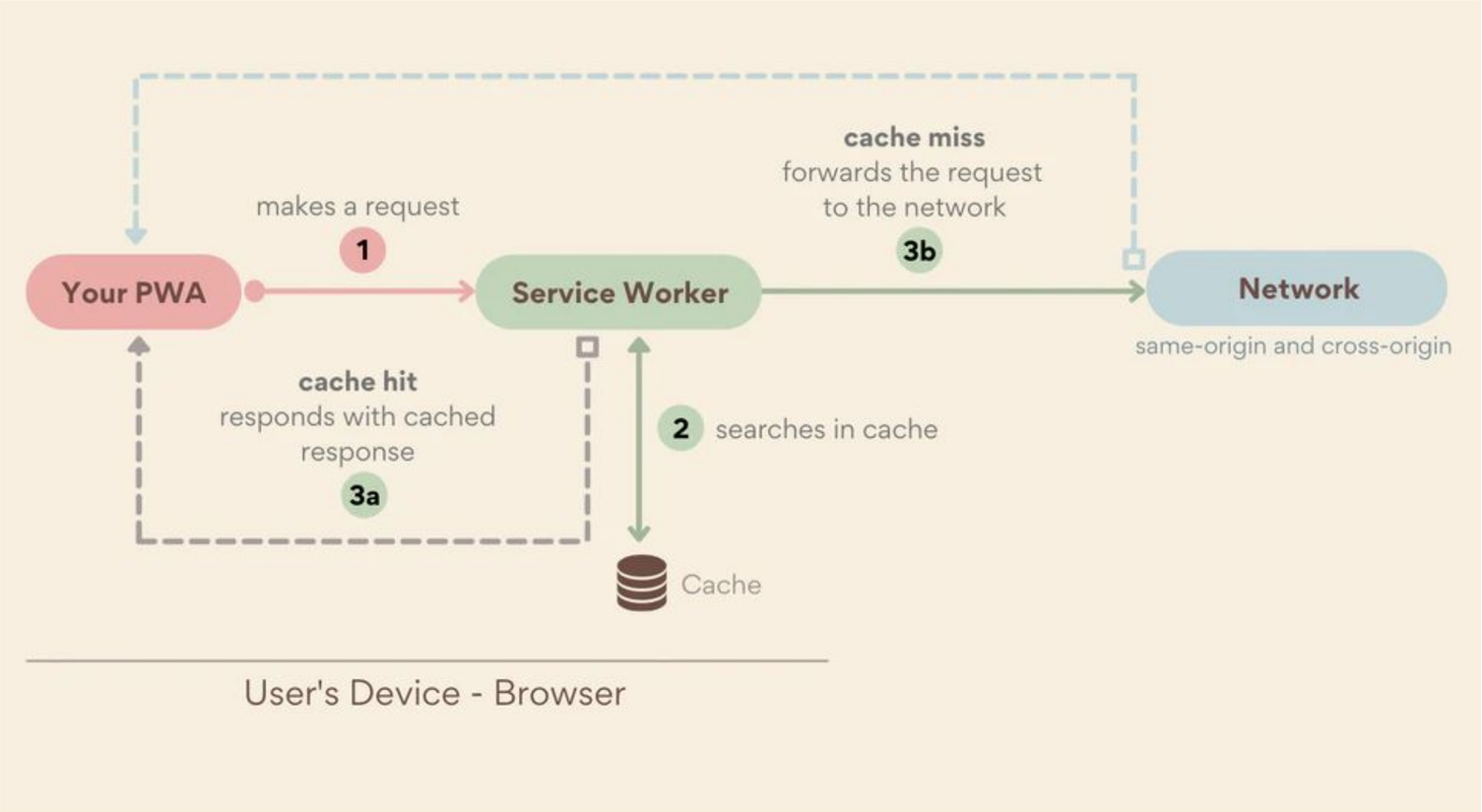
Poza cyklami życia możemy także reagować na różnego typu zmiany, które dostępne są w observablu ***reg.installing.state***

```
navigator.serviceWorker.register('/sw.js').then(reg => {
 reg.installing; // the installing worker, or undefined
 reg.waiting; // the waiting worker, or undefined
 reg.active; // the active worker, or undefined

 reg.addEventListener('updatefound', () => {
   // A wild service worker has appeared in reg.installing!
   const newWorker = reg.installing;

   newWorker.state;
   // "installing" - the install event has fired, but not yet
complete
   // "installed"  - install complete
   // "activating" - the activate event has fired, but not yet
complete
   // "activated"  - fully active
   // "redundant"  - discarded. Either failed install, or it's been
   //                replaced by a newer version

   newWorker.addEventListener('statechange', () => {
     // newWorker.state has changed
   });
 });
});

navigator.serviceWorker.addEventListener('controllerchange', () => {
 // This fires when the service worker controlling this page
 // changes, eg a new worker has skipped waiting and become
 // the new active worker.
});
```

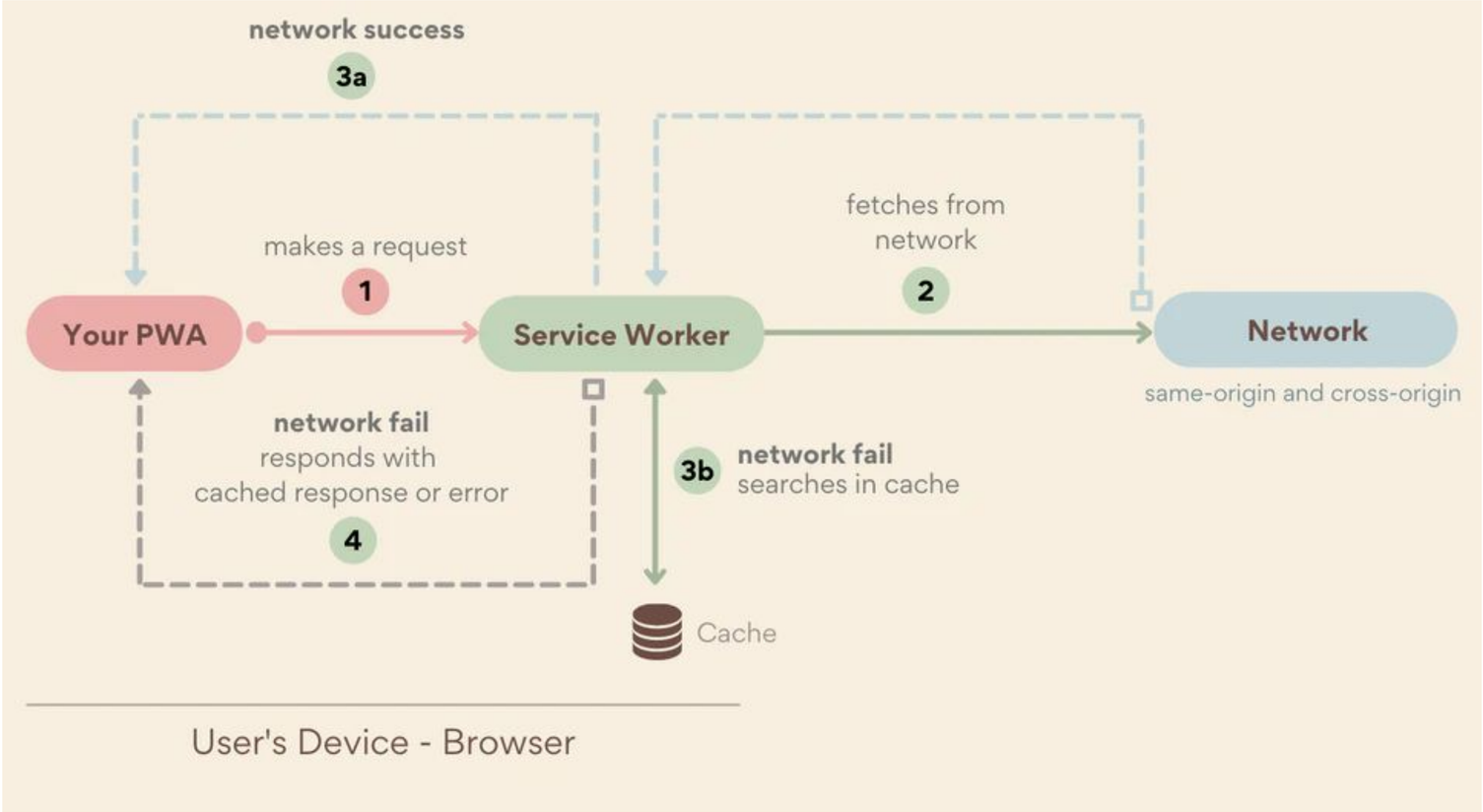
Cachowanie i serwowanie plików

- **Cache First** -> odpowiadaj zapisaną odpowiedzią, jeżeli jest w pamięci sw
- **Network First** -> jeżeli nie odpowiedz na zapytanie nie zostałą zapytana zaserwuj odpowiedź z pamięci sw
- **Stale While Revalidate** -> serwuj odpowiedź w pamięci, ale nie przerywaj zapytania, kiedy przyjdzie odpowiedź zaktualizuj pamięć sw nową wartością
- **Network-Only** -> tylko odpowiedzi serwera (nie korzystaj z pamięci)
- **Cache-Only** -> zawsze odpowiadaj zasobem z pamięci lub rzuć błędem

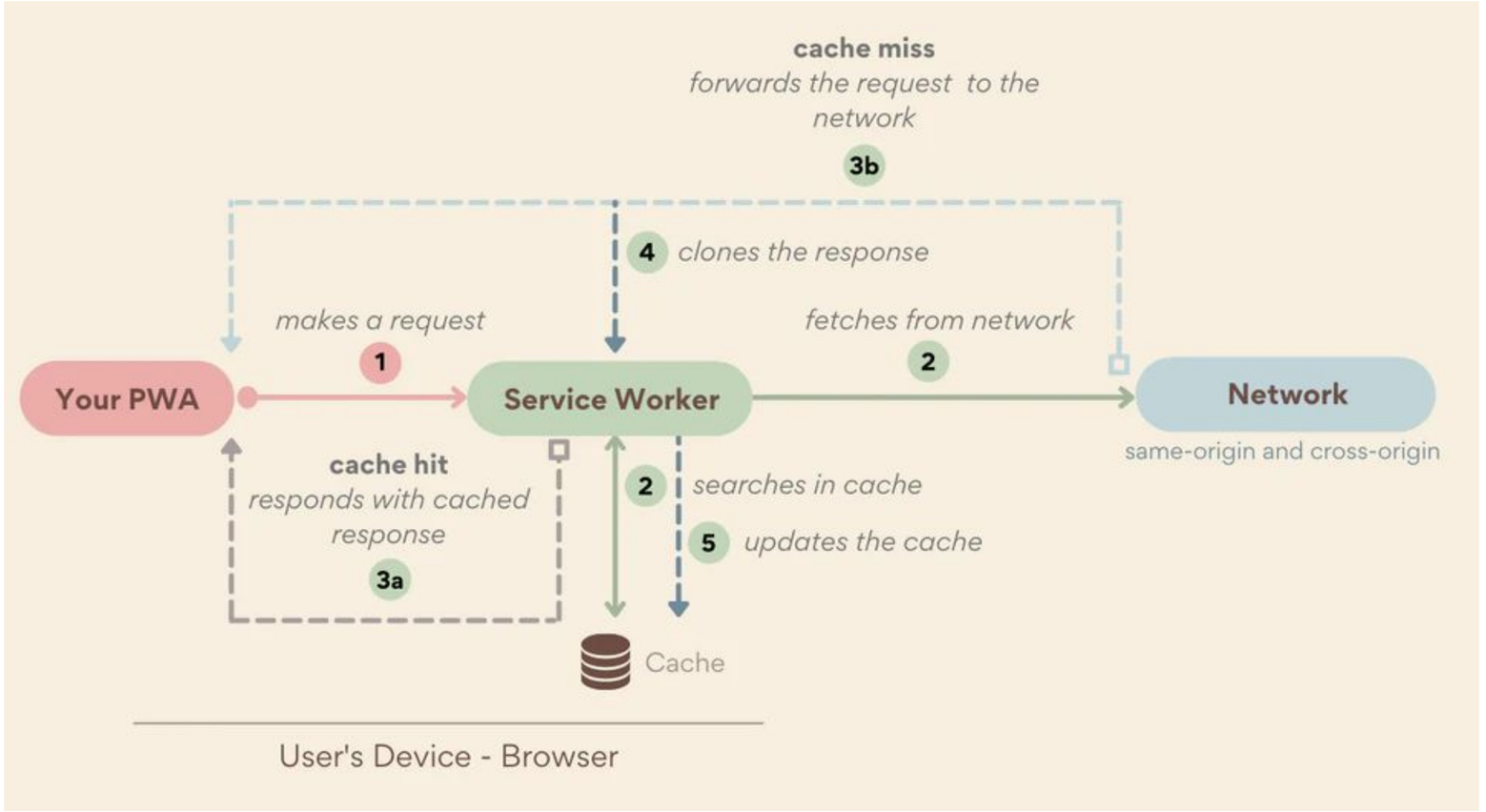
Cache First


cache miss
forwards the request
to the network
3b
makes a request
1
Your PWA
Service Worker
Network
same-origin and cross-origin
cache hit
responds with cached
response
3a
2
searches in cache
Cache
User's Device - Browser

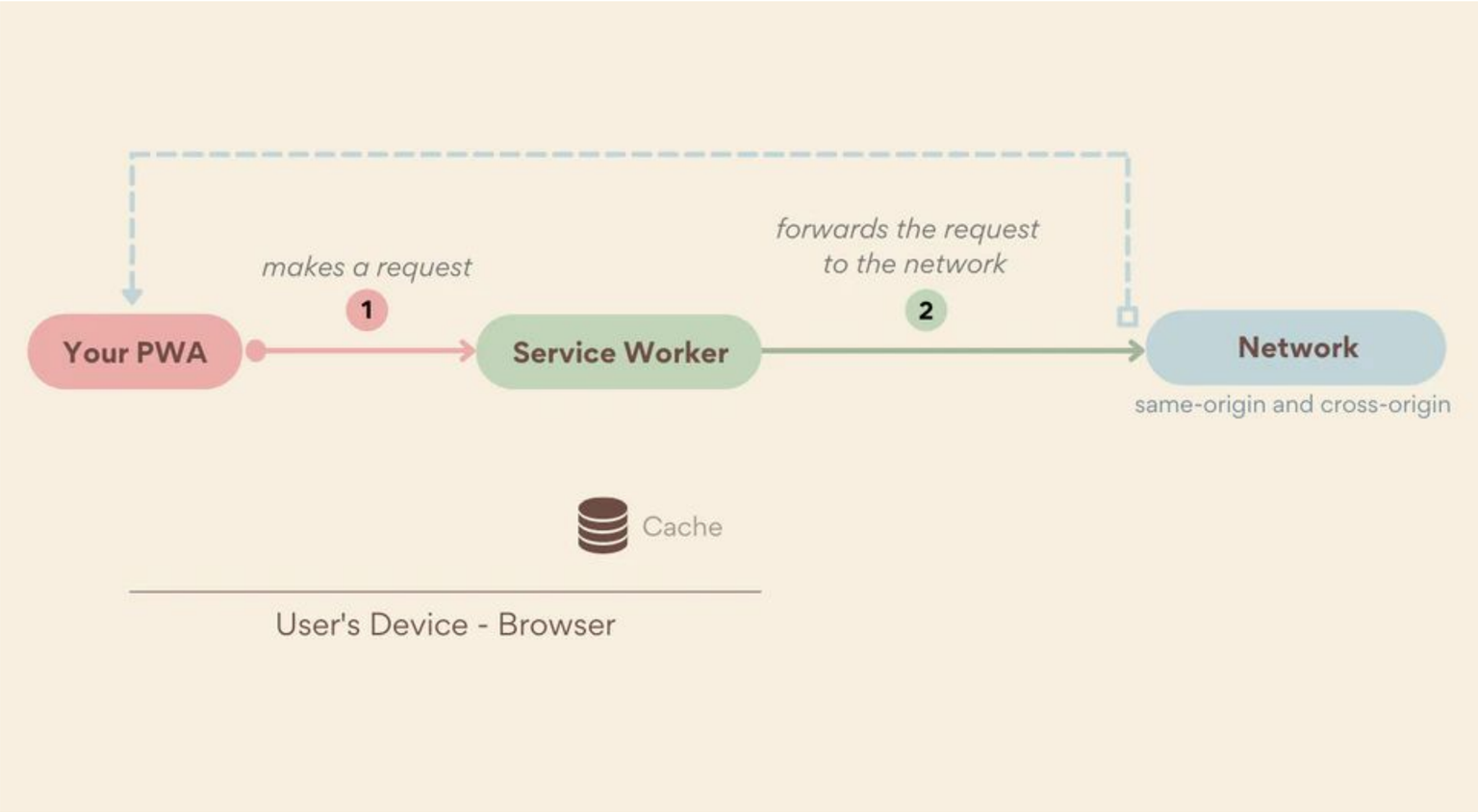
Network First



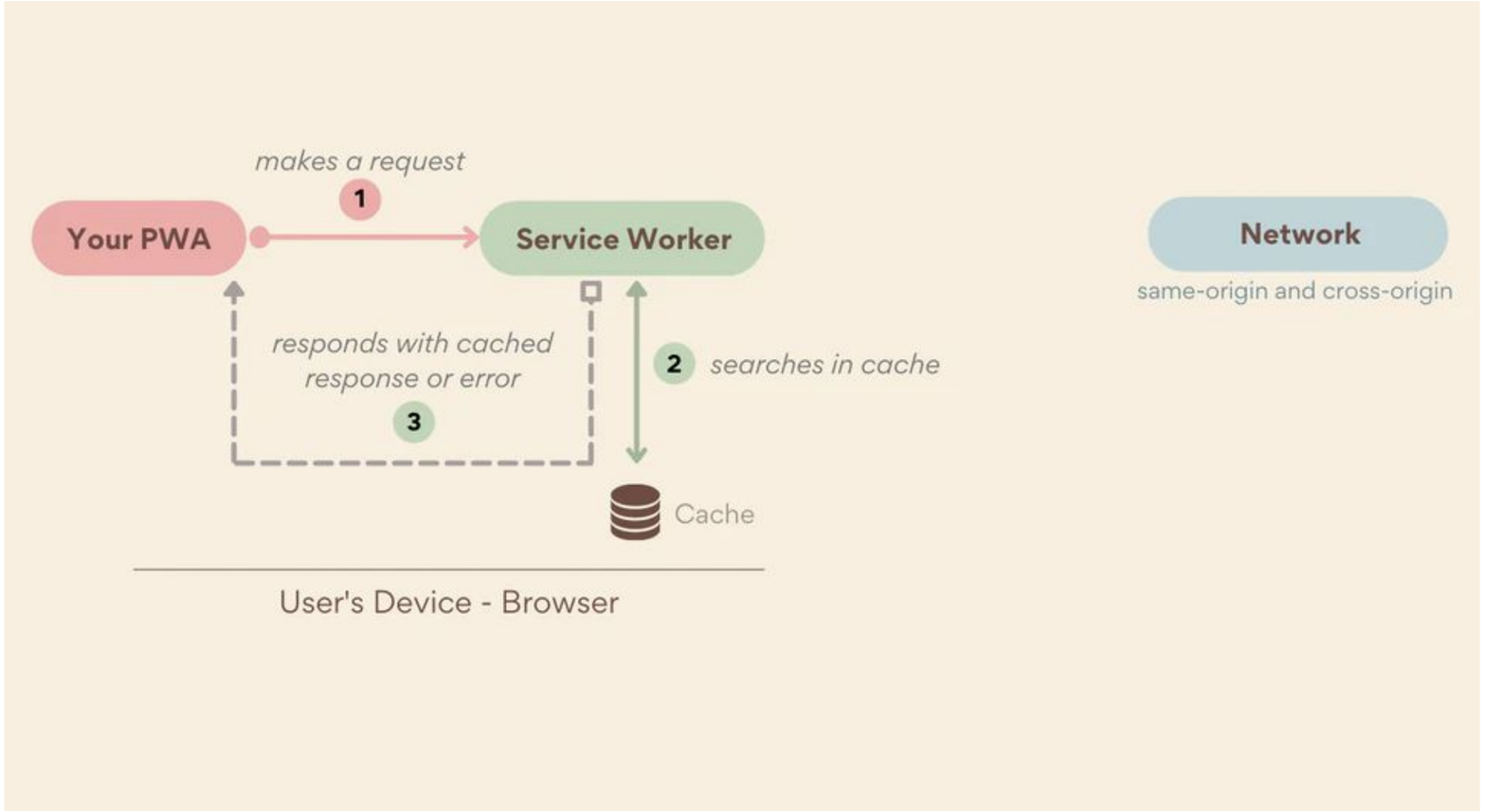
Stale While Revalidate



Network-Only



Cache-Only


makes a request
1
Your PWA
Service Worker
Network
same-origin and cross-origin
responds with cached
response or error
3
2
searches in cache
Cache
User's Device - Browser

DZIĘKUJĘ ZA UWAGĘ