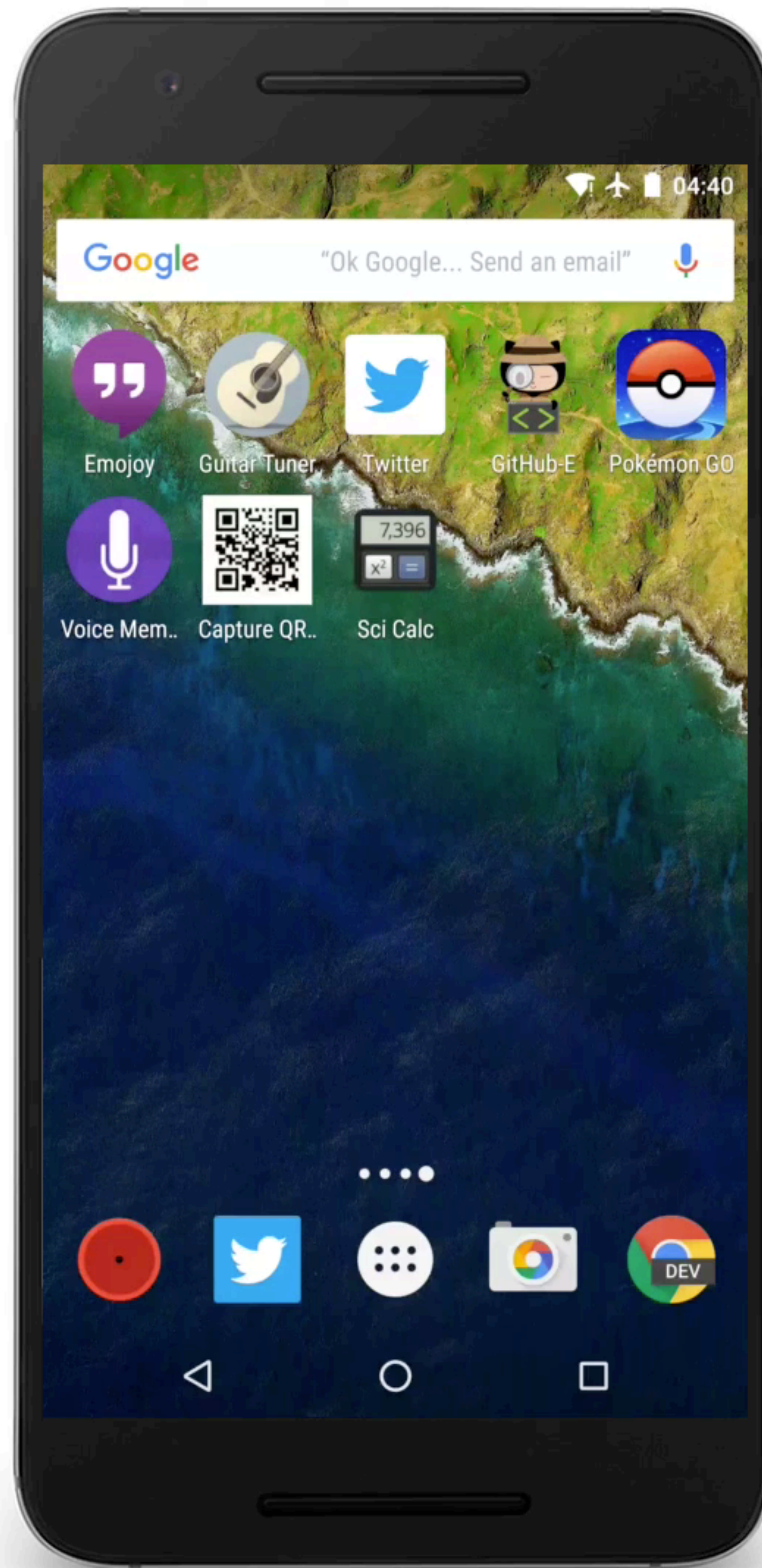


# Instant Loading and Offline

**PROGRESSIVE WEB APP**

ROADSHOW 2016 





# Unable to connect to the Internet

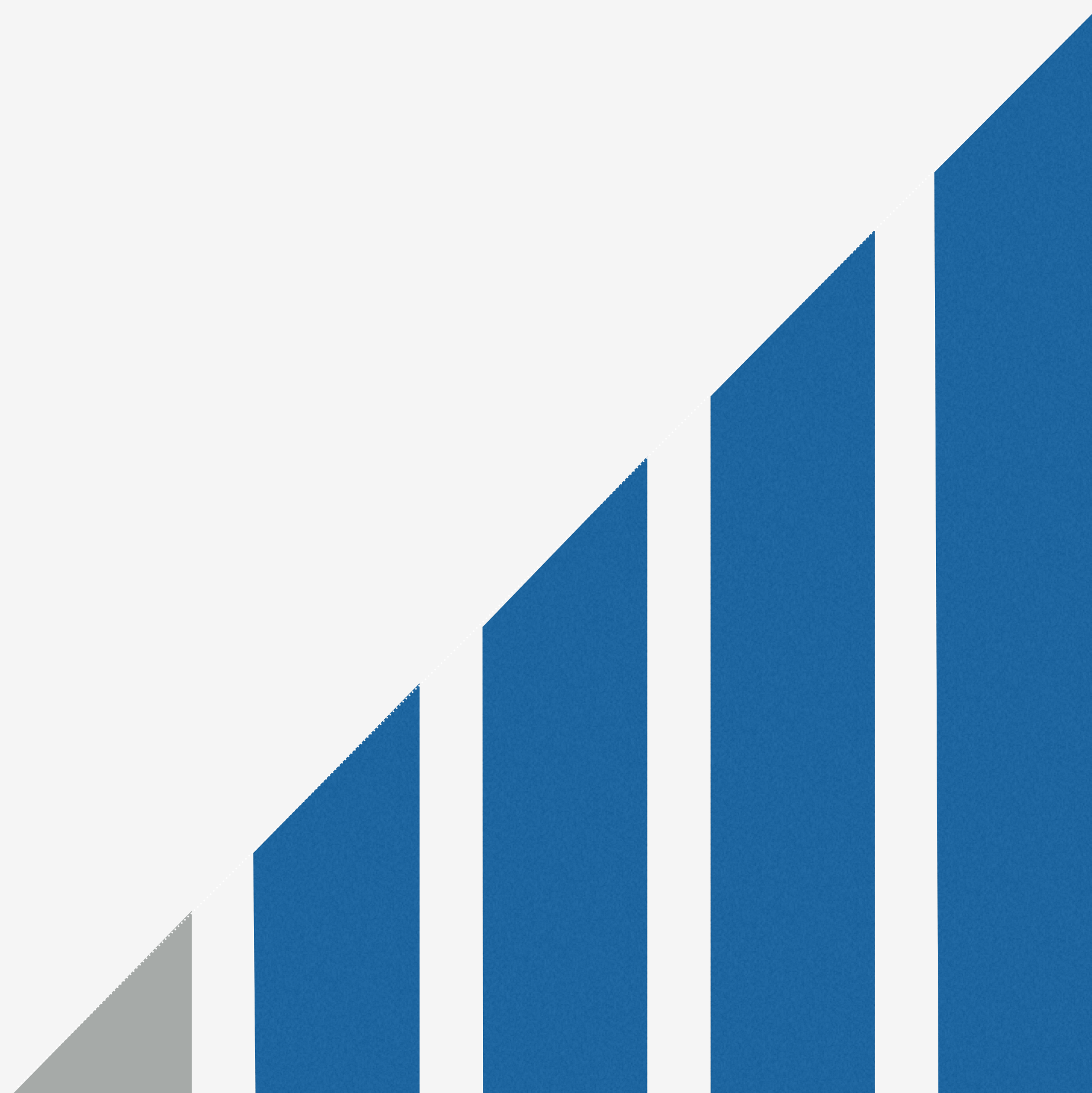
Google Chrome can't display the web page because your computer isn't connected to the Internet.

ERR\_INTERNET\_DISCONNECTED

[Details](#)



# Lie-Fi



# Lie-Fi





A close-up photograph of a person's face and hand holding a smartphone. The person is looking down at the screen. A large circular graphic is overlaid on the left side of the image. It consists of a blue inner circle with the text '53%' in white, and a red outer ring that is partially filled, representing 53% of the circle. The background is blurred, showing other people in a crowd.

53%

of users abandon sites that take  
longer than **3 seconds** to load



# 60%

of world-wide mobile  
connections are **2G**

# Service Worker

Service Workers Nightly

Paul (personal)

← → ↻ [https://slightlyoff.github.io/ServiceWorker/spec/service\\_worker/](https://slightlyoff.github.io/ServiceWorker/spec/service_worker/) ☆ ⋮

## TABLE OF CONTENTS

1

Motivations

2

Model

2.1

Service Worker

2.1.1

Lifetime

2.2

Service Worker Registration

2.2.1

Lifetime

2.3

Service Worker Client

2.4

Selection and Use

2.5

Task Sources

2.6

User Agent Shutdown

3

Client Context

3.1

ServiceWorker

3.1.1

scriptURL

3.1.2

state

3.1.3

postMessage(message, transfer)

3.1.4

Event handler

3.2

ServiceWorkerRegistration

3.2.1

installing

3.2.2

waiting

3.2.3


active

3.2.4

scope

3.2.5

update()



## Service Workers Nightly

Editor's Draft, 29 July 2016

**This version:**  
[https://slightlyoff.github.io/ServiceWorker/spec/service\\_worker/](https://slightlyoff.github.io/ServiceWorker/spec/service_worker/)

**Latest published version:**  
<https://www.w3.org/TR/service-workers/>

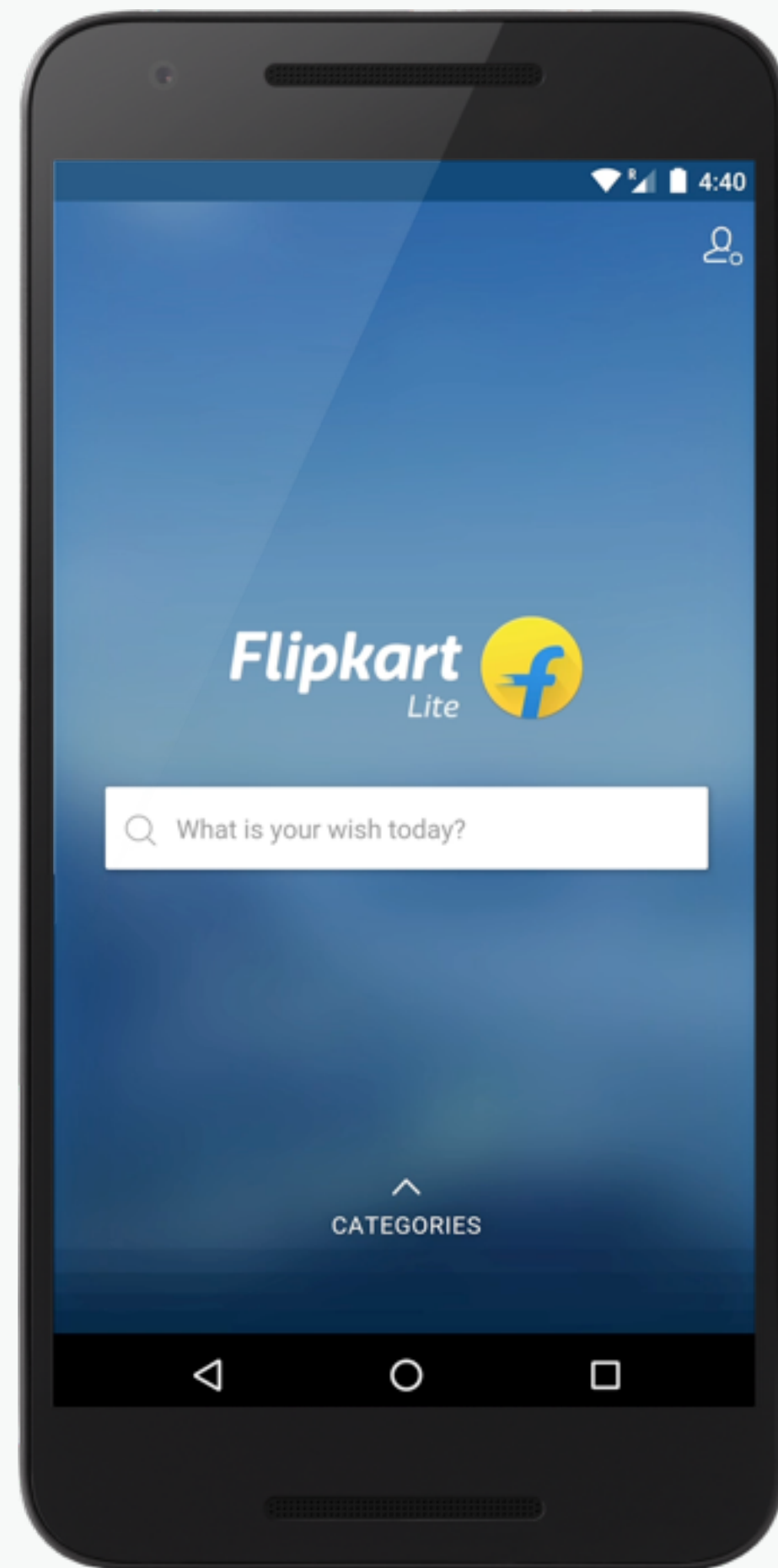
**Feedback:**  
[public-webapps@w3.org](mailto:public-webapps@w3.org) with subject line "[service-workers] ... message topic ..." ([archives](#))

**Issue Tracking:**  
[GitHub](#)  
[Inline In Spec](#)

**Editors:**  
[Alex Russell](#) (Google)  
[Jungkee Song](#) (Samsung Electronics)  
[Jake Archibald](#) (Google)  
[Marijn Kruisselbrink](#) (Google)

[Copyright](#) © 2016 [W3C](#)® ([MIT](#), [ERCIM](#), [Keio](#), [Beihang](#)). W3C [liability](#), [trademark](#) and [permissive document license](#) rules apply.



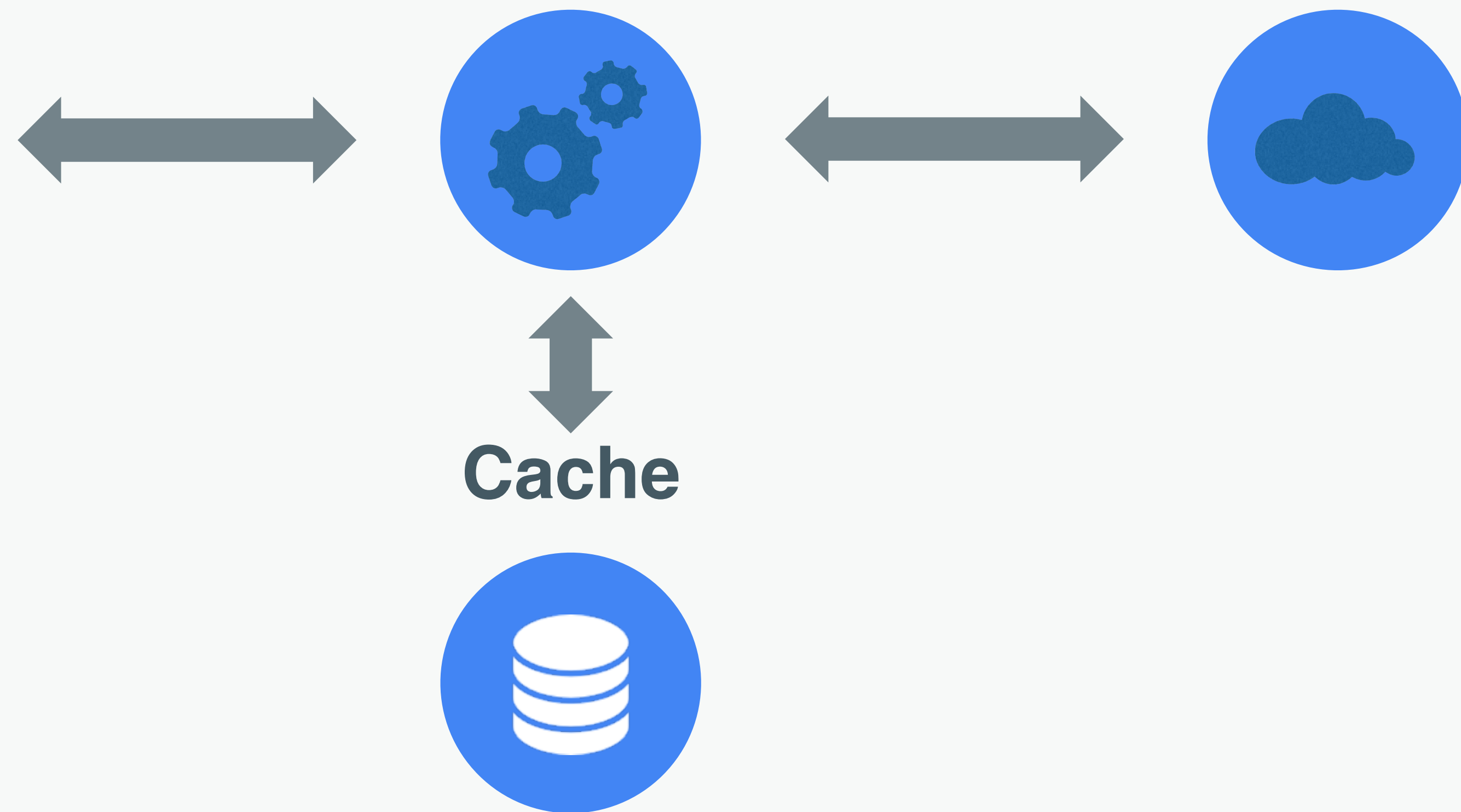


# Client side proxy

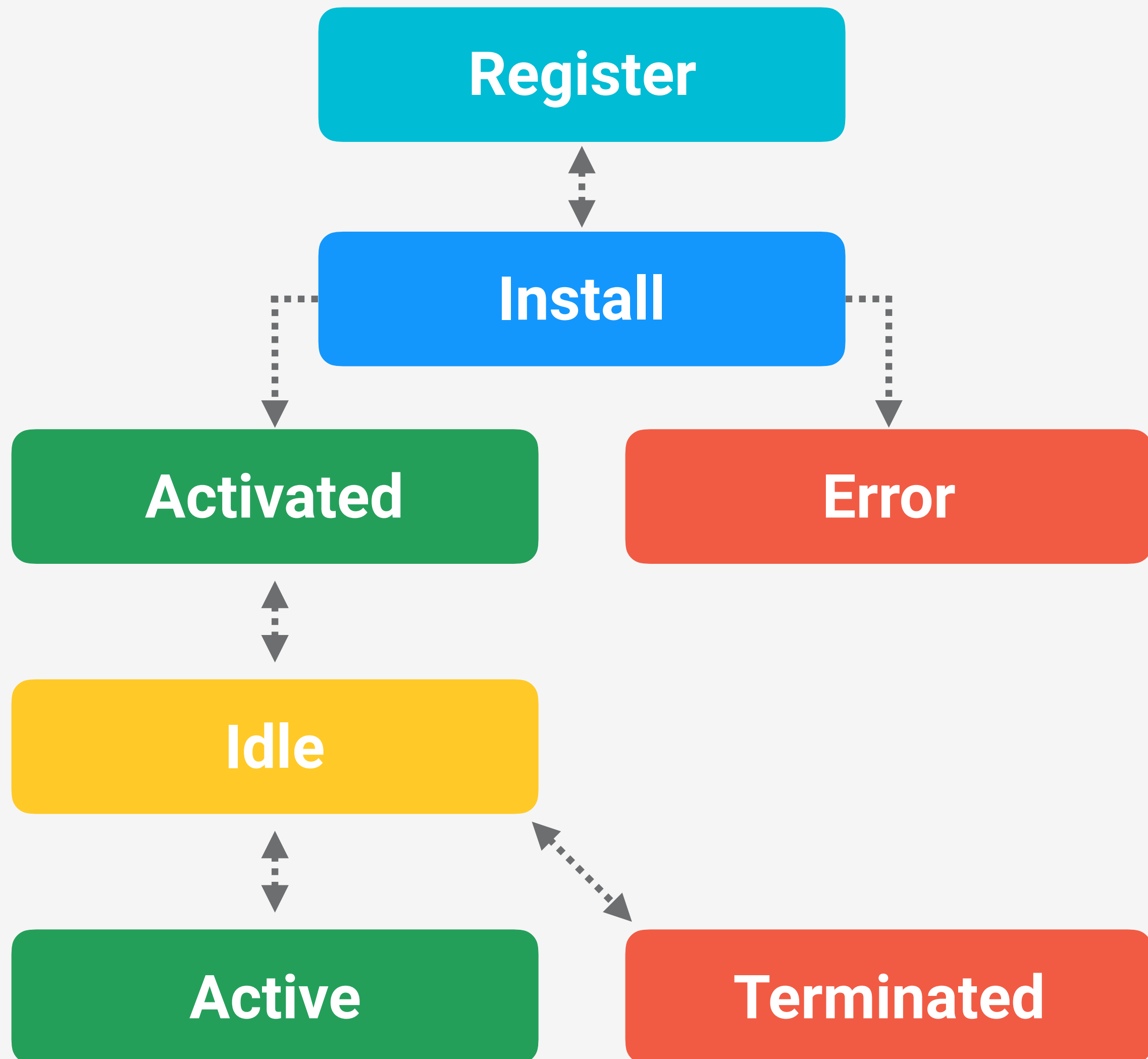
(written in JavaScript)

## Service worker

## Web server



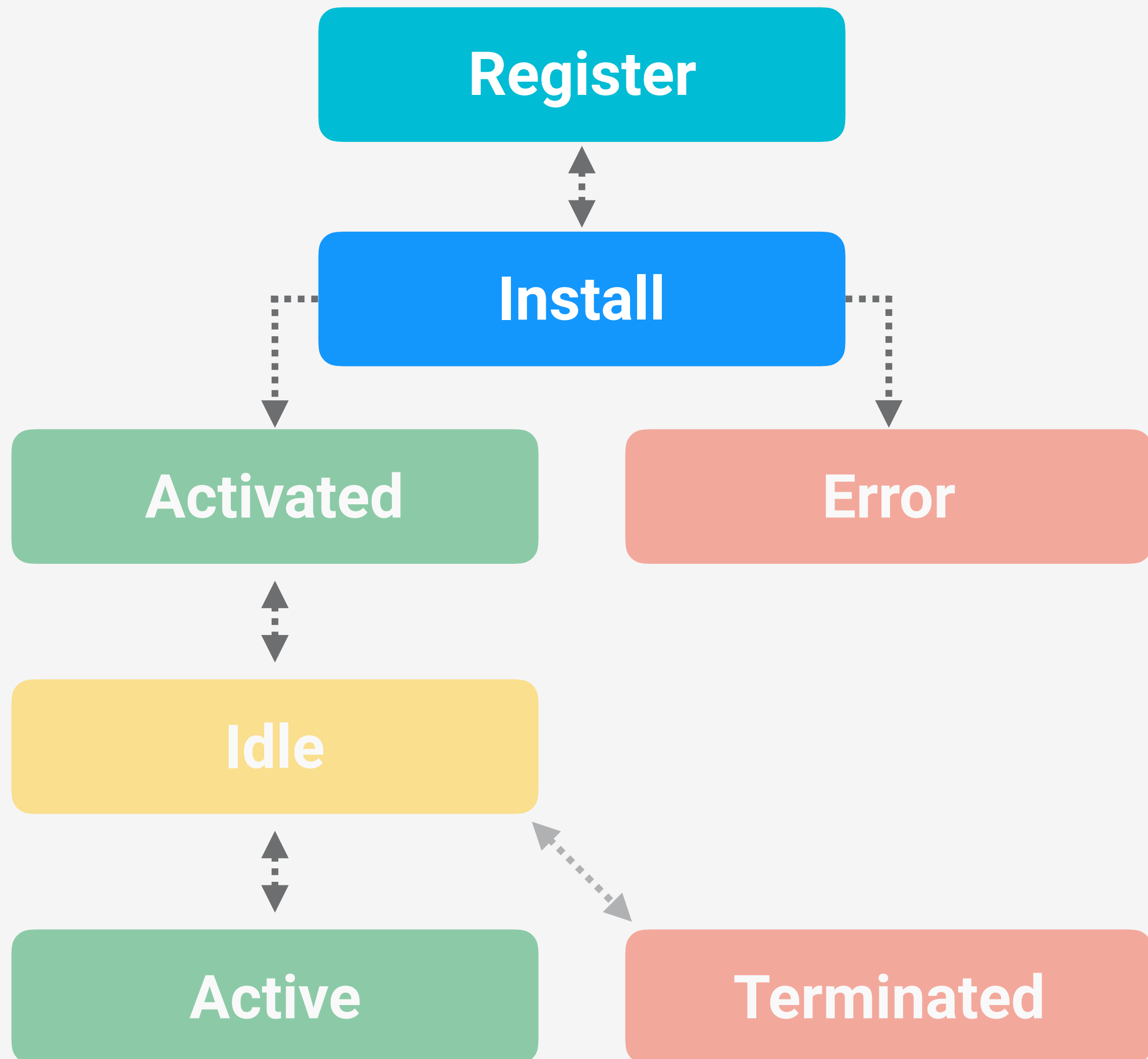
# Service Worker Life Cycle



- Adds app-like lifecycle to a page
- Wakes up only when the OS says
- Only responds to system events

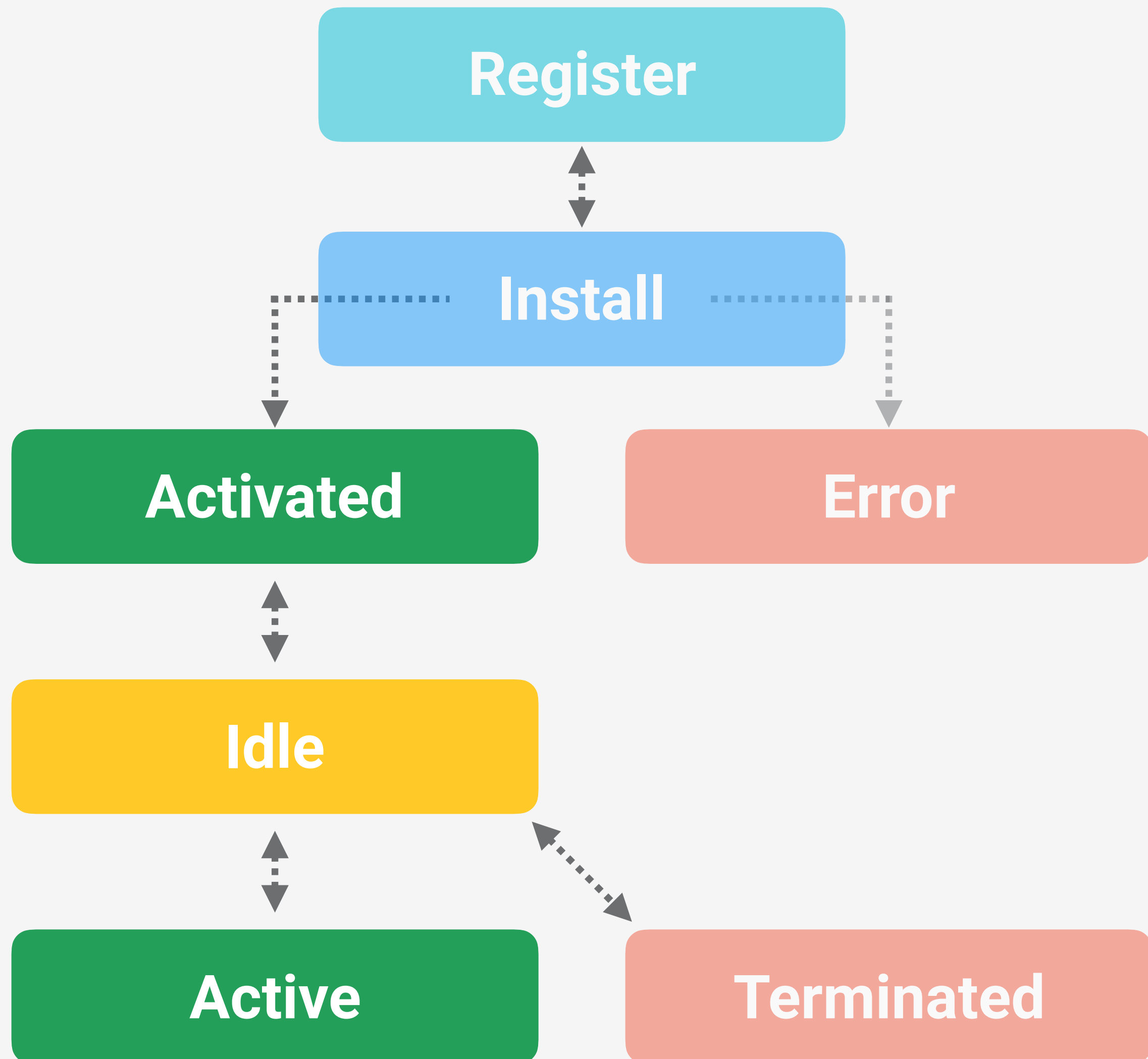


# Service Worker Life Cycle



- Adds app-like lifecycle to a page
- Wakes up only when the OS says
- Only responds to system events

# Service Worker Life Cycle



- Adds app-like lifecycle to a page
- Wakes up only when the OS says
- Only responds to system events

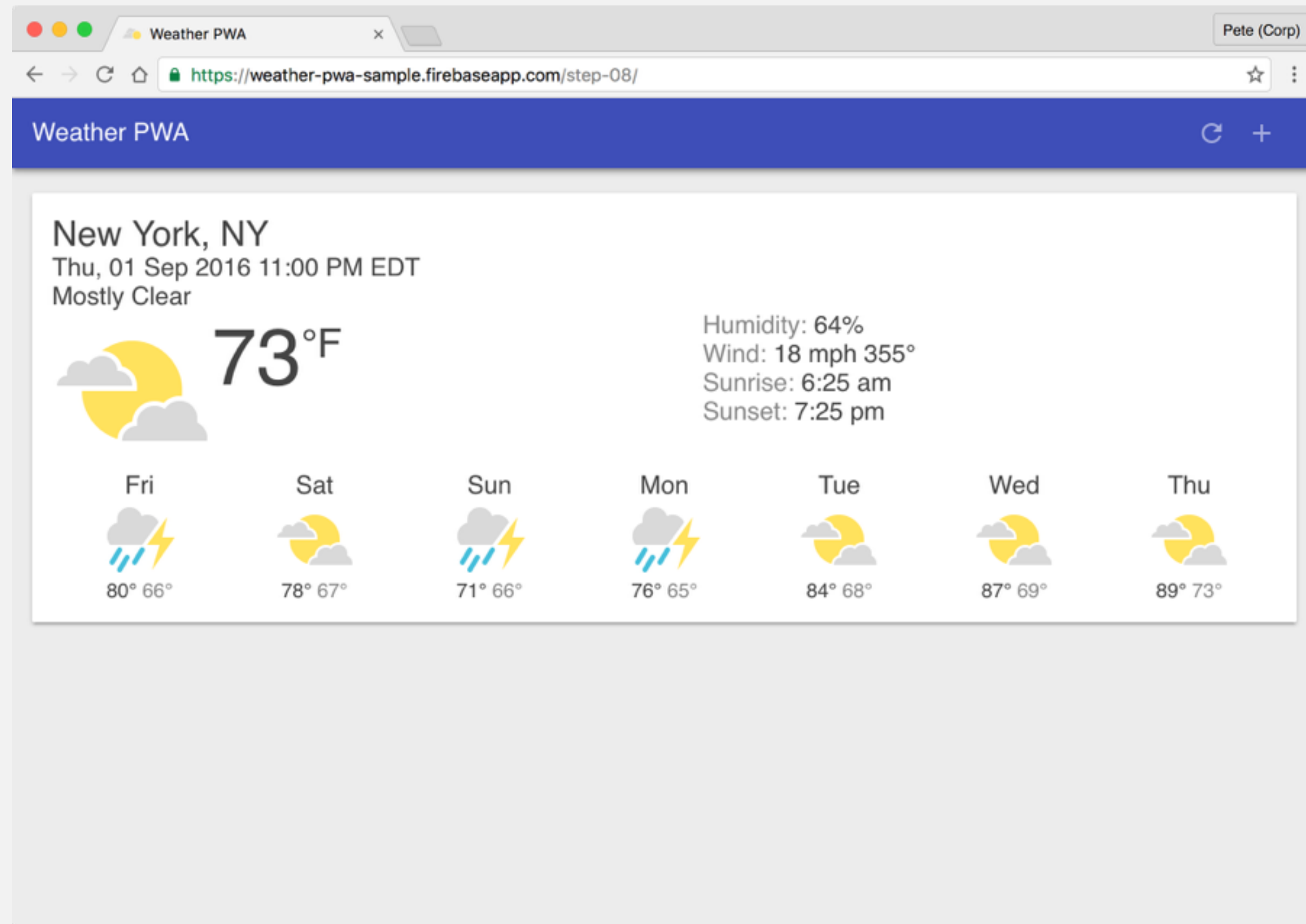


# Service Worker

is for the SECOND load.

# Life Cycle of a Service Worker

## Installing the Service Worker



install

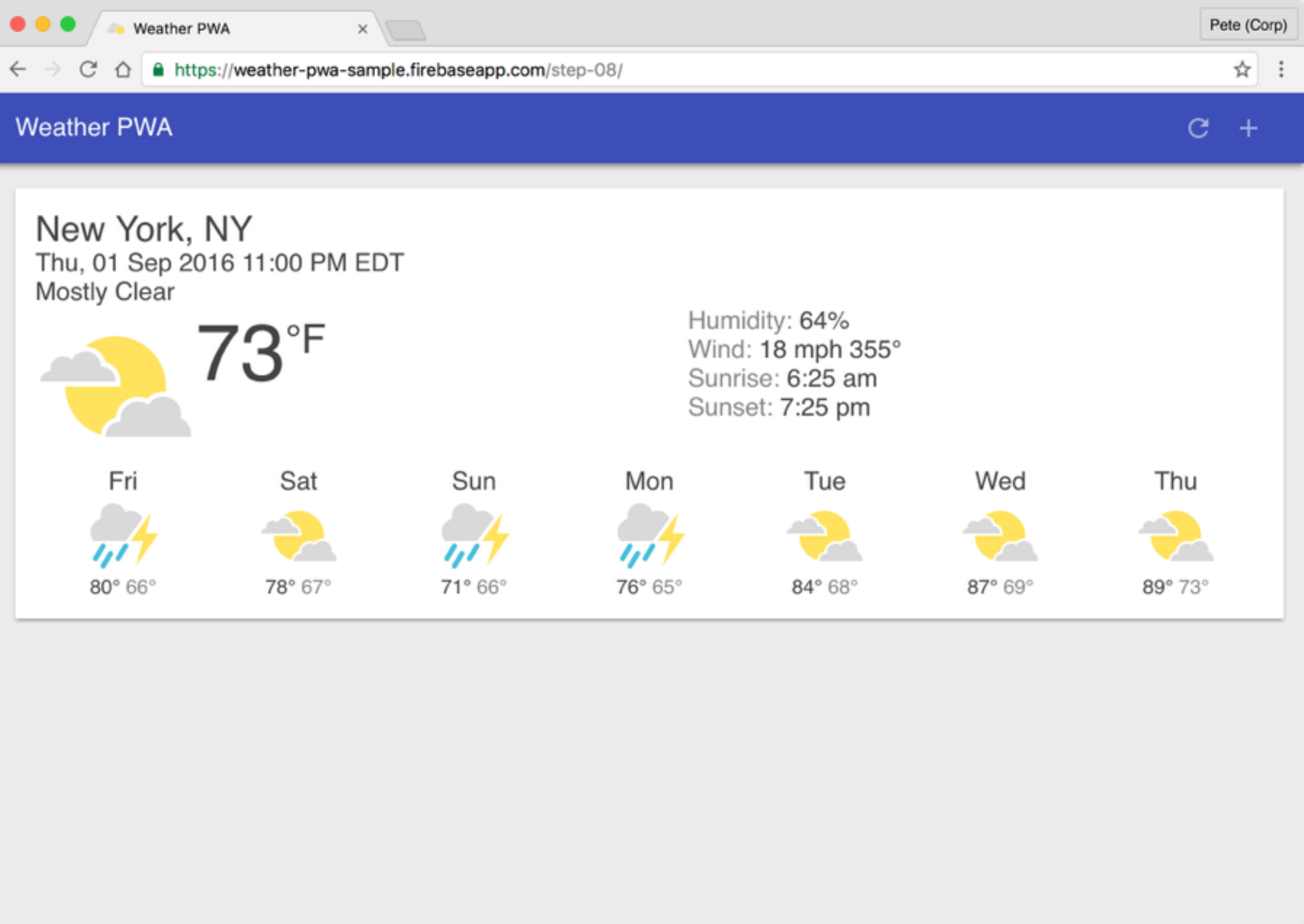
idle

1st Page Load



# Life Cycle of a Service Worker

## Installing the Service Worker



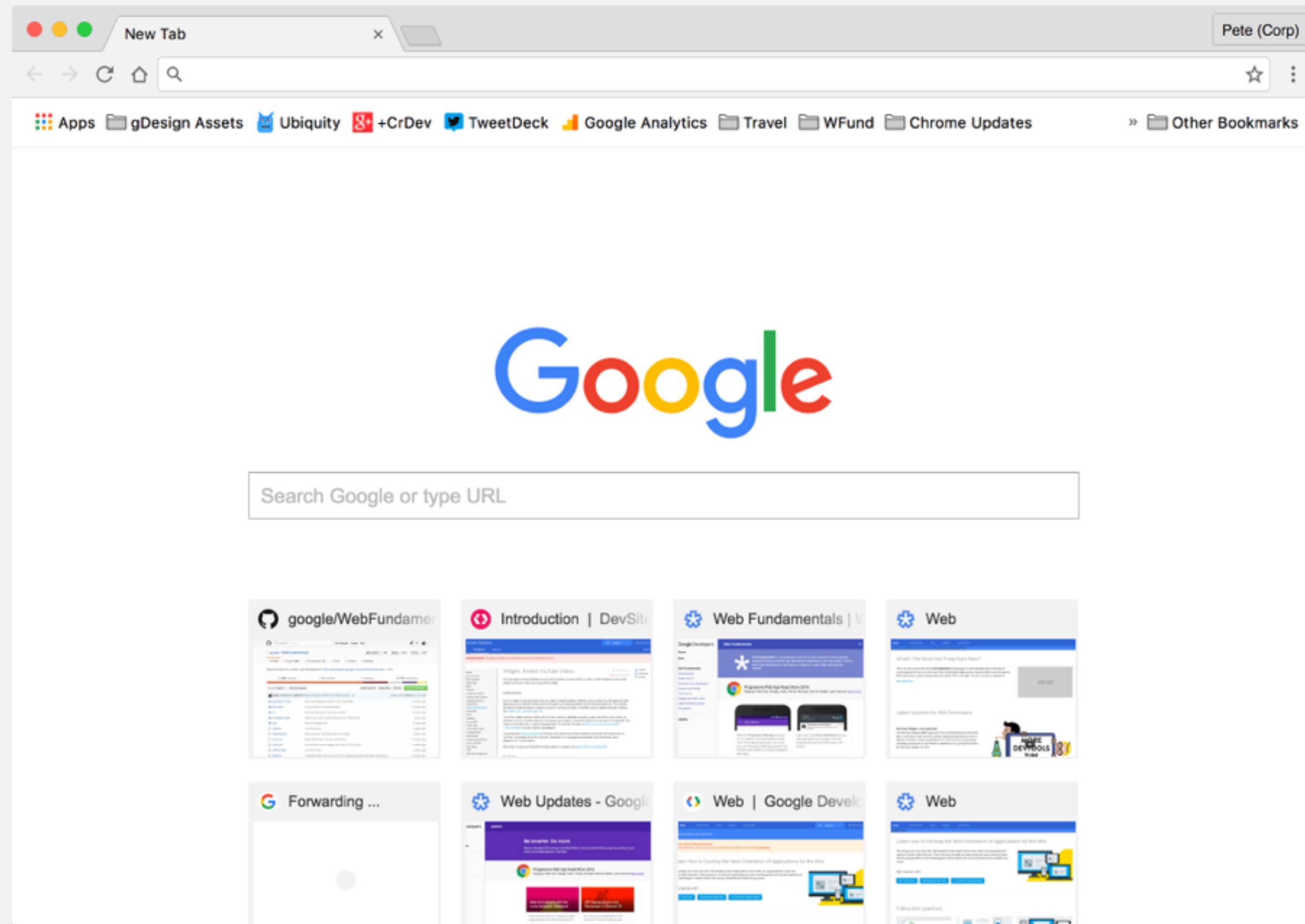
activated

check for  
update

2nd Page Load

# Life Cycle of a Service Worker

## Installing the Service Worker



activated

idle

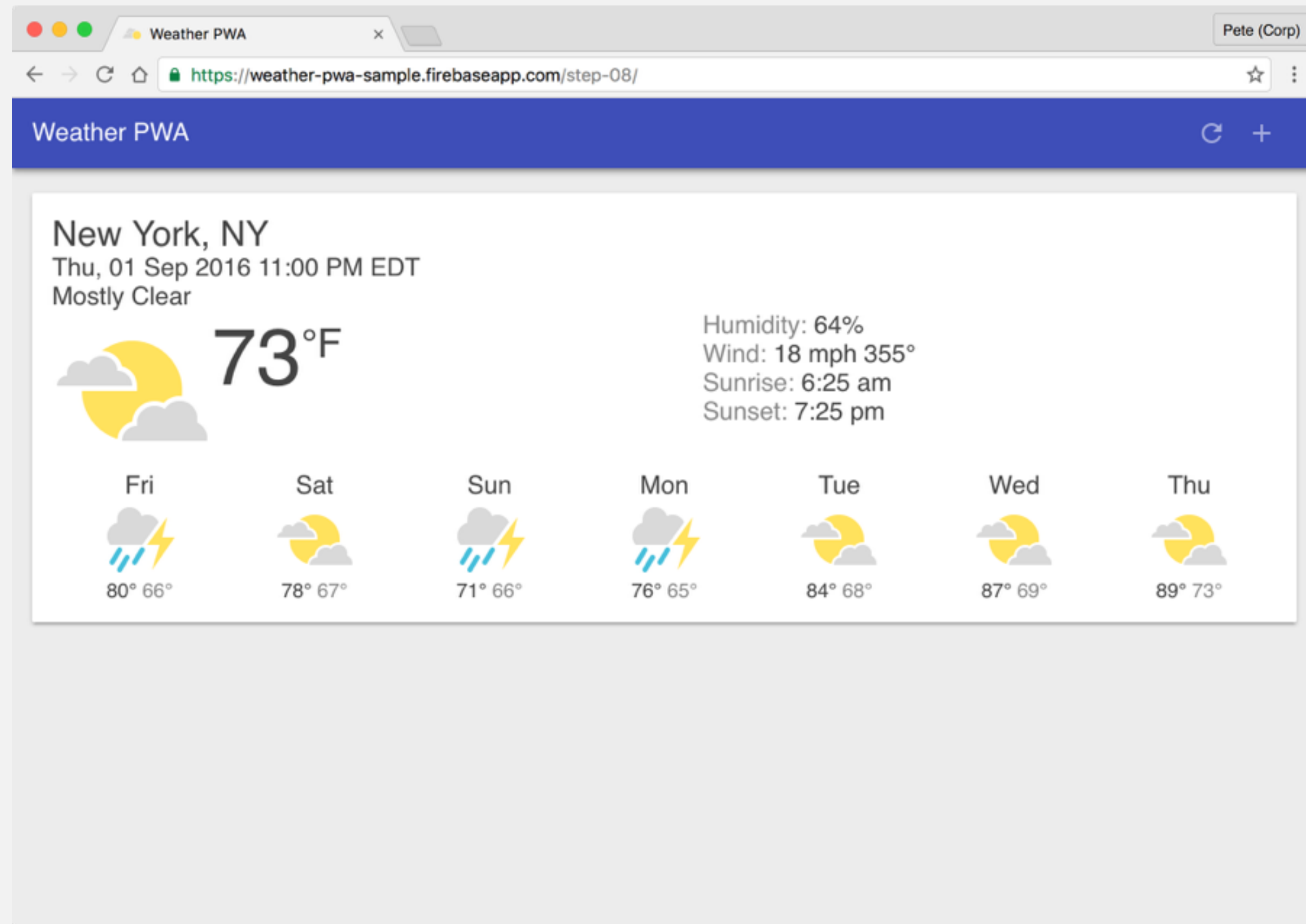
terminated

Close Page



# Life Cycle of a Service Worker

## Updating a Service Worker



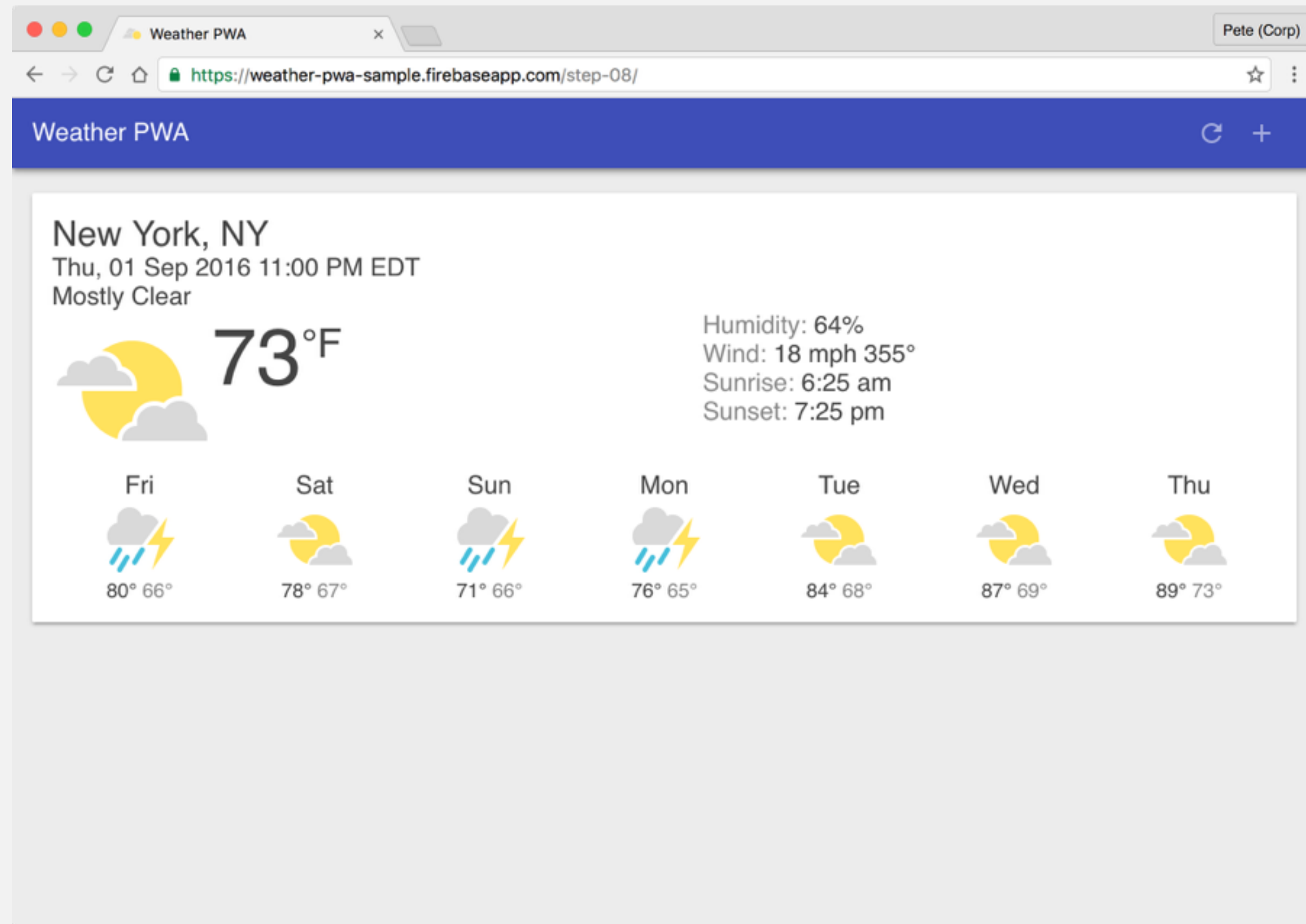
activated

check for  
update

Load Page with Service Worker

# Life Cycle of a Service Worker

## Updating a Service Worker



activated

check for  
update

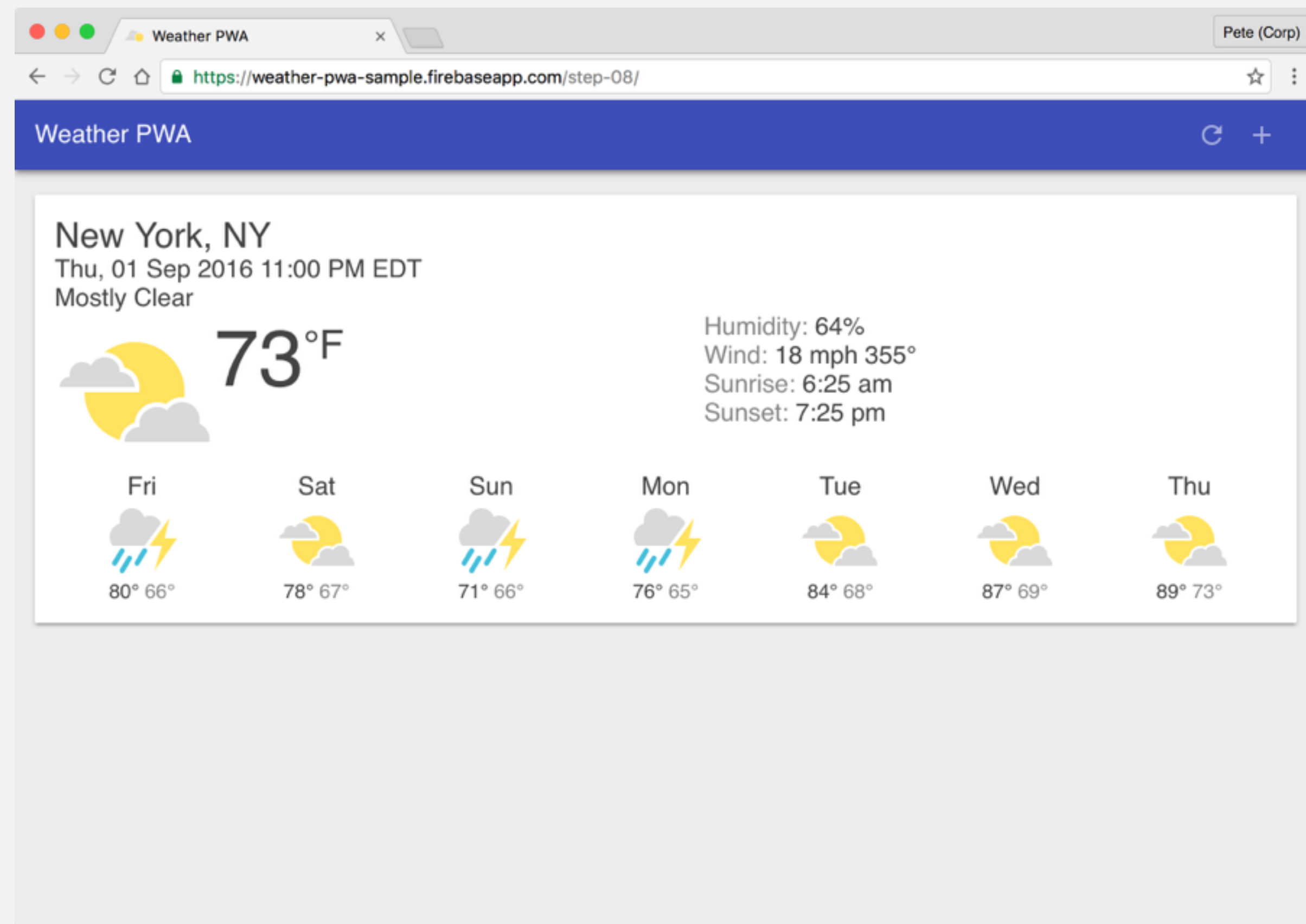
install

idle

Load Page with Service Worker

# Life Cycle of a Service Worker

## Updating a Service Worker



activated

check for  
update

install

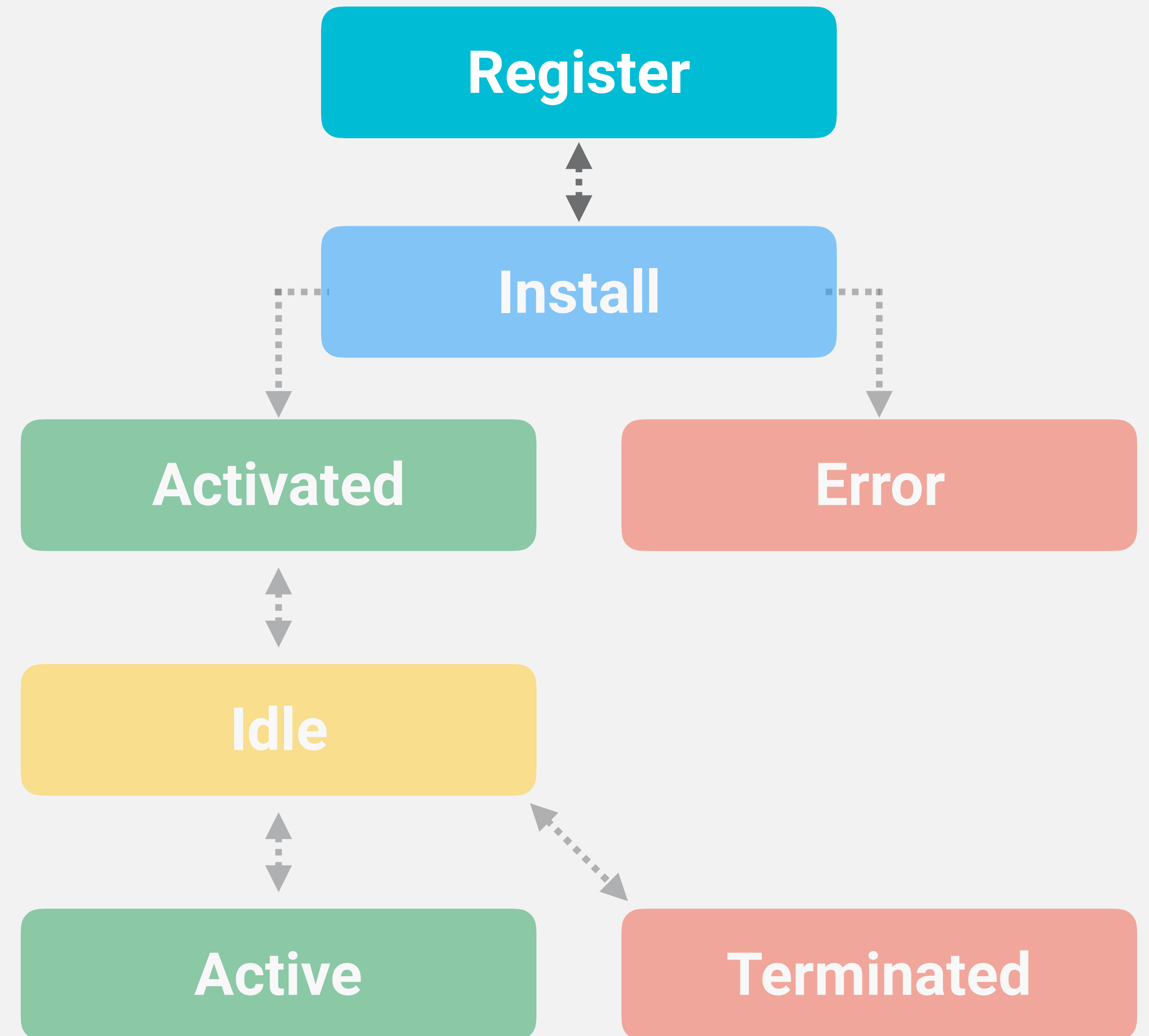
idle

Load Page with Service Worker



# Implementing a simple Service Worker

# Register the Service Worker



# Register the Service Worker

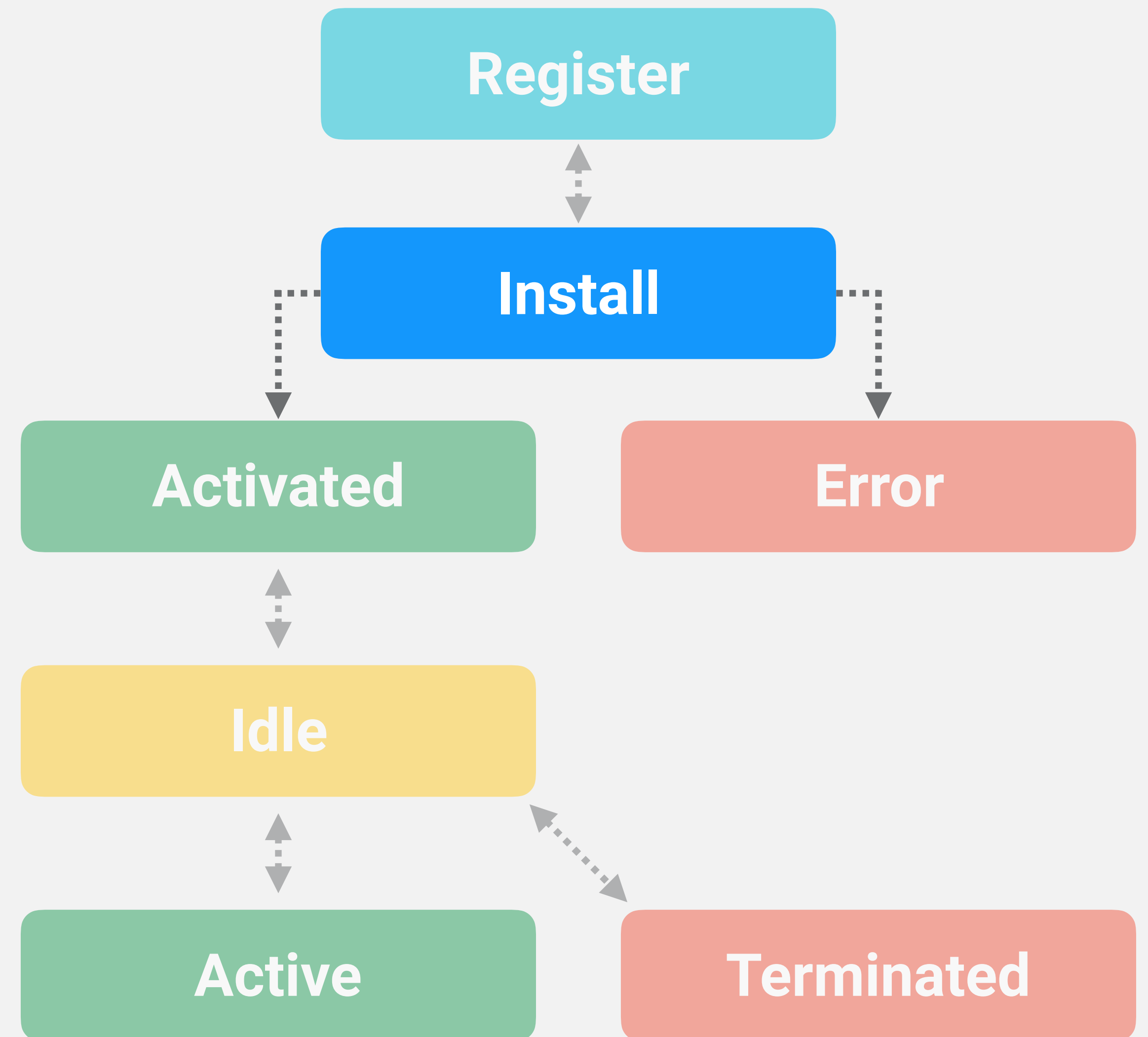
```
if ('serviceWorker' in navigator) {  
  navigator.serviceWorker.register('/service-worker.js')  
    .then(function(reg) {  
      console.log('Service Worker Registered', reg);  
    })  
    .catch(function(err) {  
      console.log('Error registering Service Worker', err);  
    });  
}
```



# Register the Service Worker

```
if ('serviceWorker' in navigator) {  
  navigator.serviceWorker.register('/service-worker.js')  
    .then(function(reg) {  
      console.log('Service Worker Registered', reg);  
    })  
    .catch(function(err) {  
      console.log('Error registering Service Worker', err);  
    });  
}
```

# Add an install Event Handler

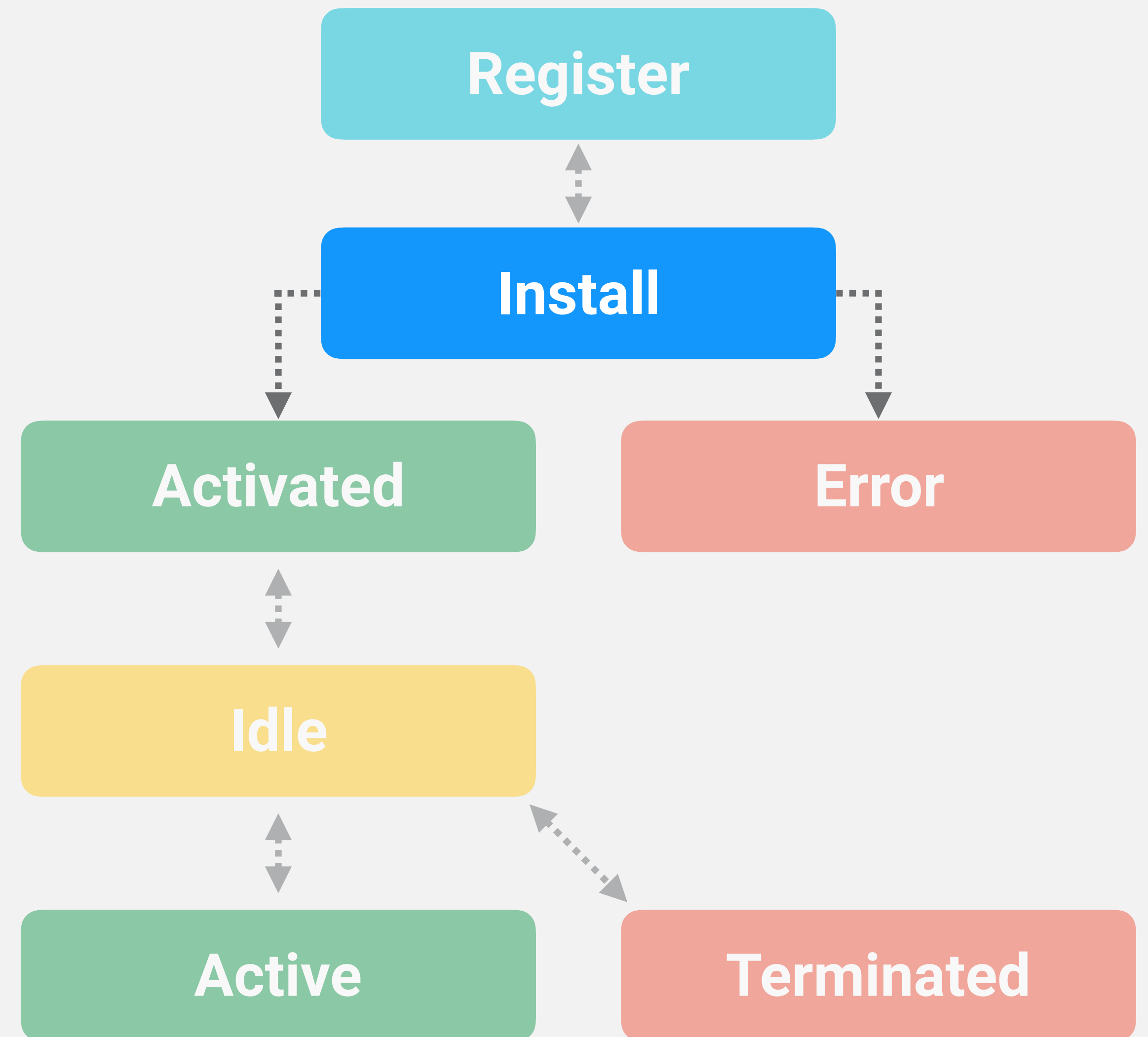


# Add an install Event Handler

```
self.addEventListener('install', function(event) {  
    return self.skipWaiting();  
});
```



# Pre-fetch the App Resources



# Pre-fetch the App Resources

```
var cacheName = 'app-shell-cache-v1';  
var filesToCache = ['/', '/index.html', ...];  
  
self.addEventListener('install', function(event) {  
  event.waitUntil(  
    caches.open(cacheName).then(function(cache) {  
      return cache.addAll(filesToCache);  
    }).then(function() {  
      return self.skipWaiting();  
    })  
  );  
});
```

# Pre-fetch the App Resources

```
var cacheName = 'app-shell-cache-v1';
var filesToCache = ['/', '/index.html', ...];

self.addEventListener('install', function(event) {
  event.waitUntil(
    caches.open(cacheName).then(function(cache) {
      return cache.addAll(filesToCache);
    }).then(function() {
      return self.skipWaiting();
    })
  );
});
```



# Pre-fetch the App Resources

```
var cacheName = 'app-shell-cache-v1';
var filesToCache = ['/', '/index.html', ...];

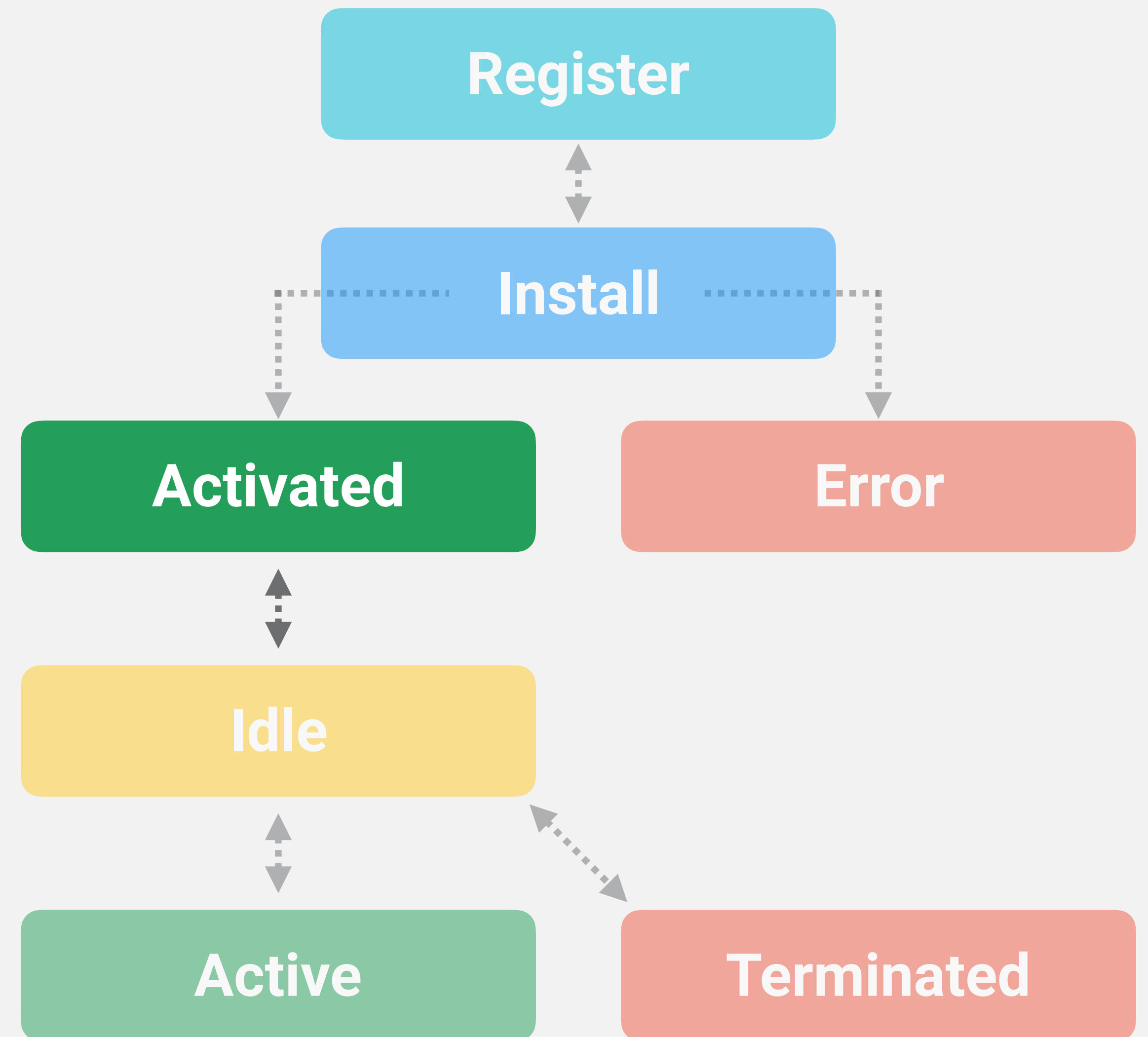
self.addEventListener('install', function(event) {
  event.waitUntil(
    caches.open(cacheName).then(function(cache) {
      return cache.addAll(filesToCache);
    }).then(function() {
      return self.skipWaiting();
    })
  );
});
```

# Pre-fetch the App Resources

```
var cacheName = 'app-shell-cache-v1';
var filesToCache = ['/', '/index.html', ...];

self.addEventListener('install', function(event) {
  event.waitUntil(
    caches.open(cacheName).then(function(cache) {
      return cache.addAll(filesToCache);
    }).then(function() {
      return self.skipWaiting();
    })
  );
});
```

# Add an activate Event Handler



# Add an activate Event Handler

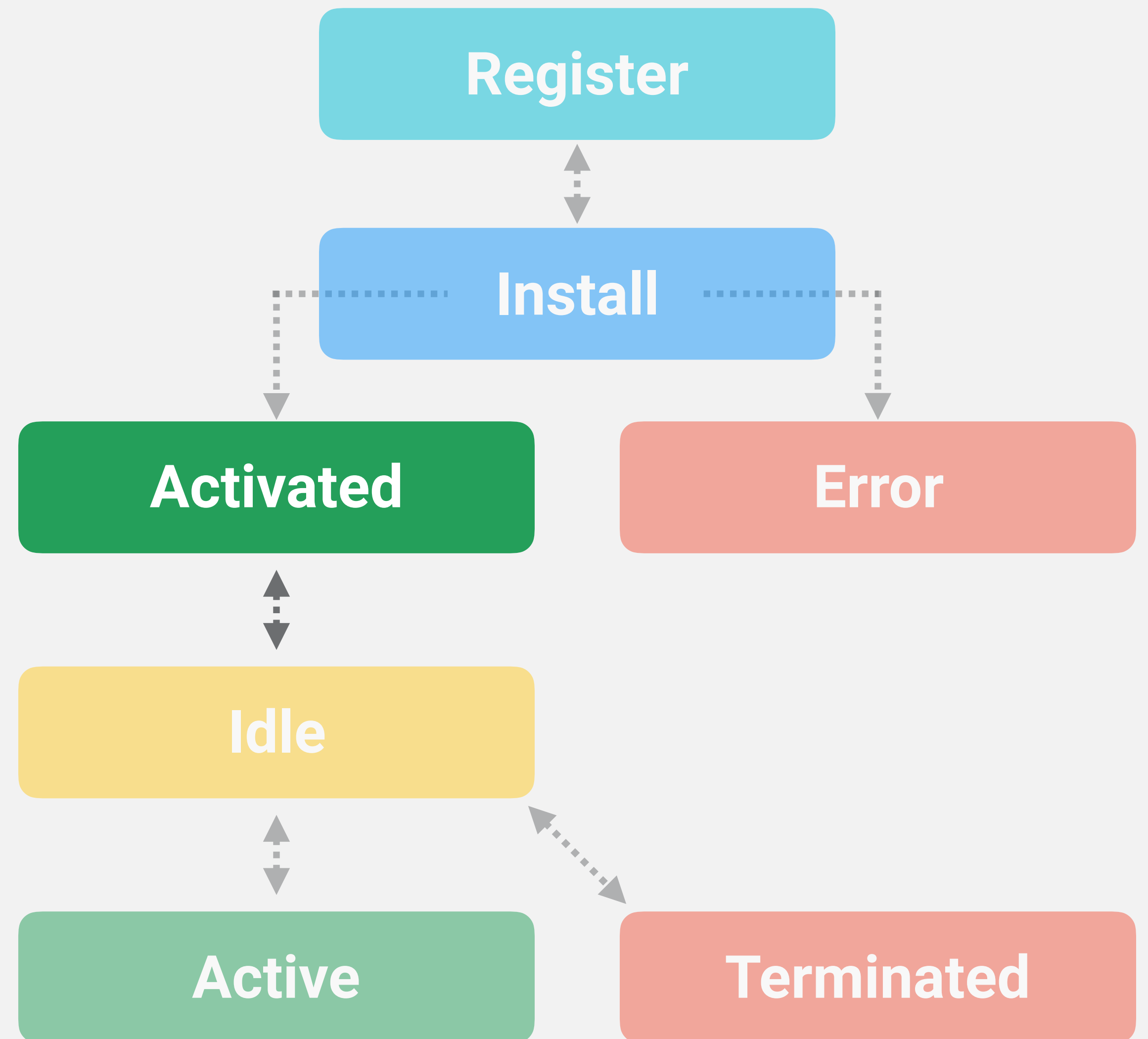
```
self.addEventListener('activate', function(e) {  
  e.waitUntil(  
    caches.keys().then(function(keyList) {  
      return Promise.all(keyList.map(function(key) {  
        if (key !== cacheName) {  
          return caches.delete(key);  
        }  
      }));  
    }));  
  });  
  return self.clients.claim();  
});
```



# Add an activate Event Handler

```
self.addEventListener('activate', function(e) {  
  e.waitUntil(  
    caches.keys().then(function(keyList) {  
      return Promise.all(keyList.map(function(key) {  
        if (key !== cacheName) {  
          return caches.delete(key);  
        }  
      }));  
    })  
  );  
  return self.clients.claim();  
});
```

# Not Done Yet...





# Unable to connect to the Internet

Google Chrome can't display the web page because your computer isn't connected to the Internet.

ERR\_INTERNET\_DISCONNECTED

[Details](#)

# Add a fetch Event Handler

```
self.addEventListener('fetch', function(e) {  
  e.respondWith(  
    caches.match(e.request).then(function(response) {  
      return response || fetch(e.request);  
    })  
  );  
});
```



# Add a fetch Event Handler

```
self.addEventListener('fetch', function(e) {  
  e.respondWith(  
    caches.match(e.request).then(function(response) {  
      return response || fetch(e.request);  
    })  
  );  
});
```

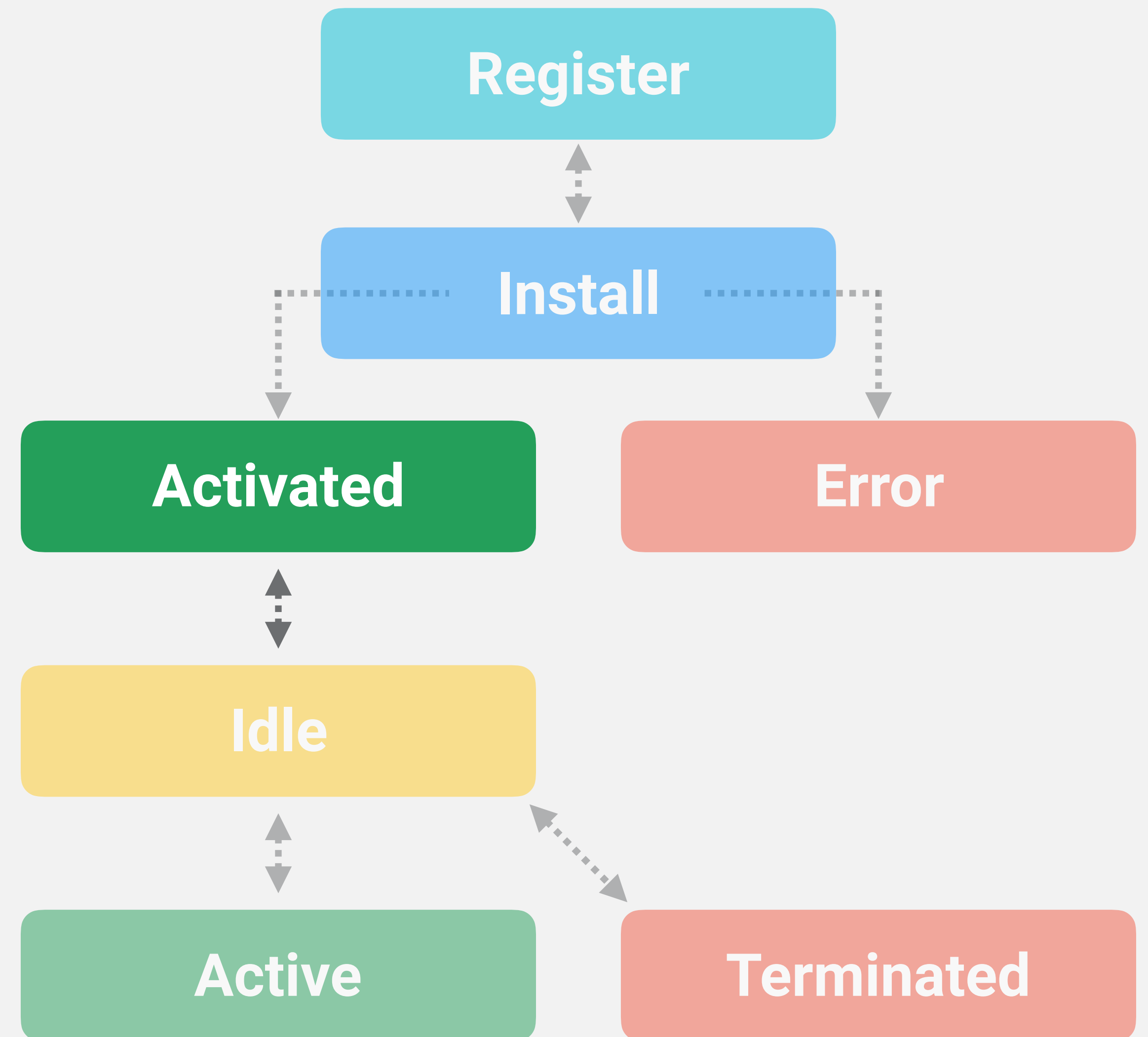
# Add a fetch Event Handler

```
self.addEventListener('fetch', function(e) {  
  e.respondWith(  
    caches.match(e.request).then(function(response) {  
      return response || fetch(e.request);  
    })  
  );  
});
```

# Add a fetch Event Handler

```
self.addEventListener('fetch', function(e) {  
  e.respondWith(  
    caches.match(e.request).then(function(response) {  
      return response || fetch(e.request);  
    })  
  );  
});
```

# Ready to Go!





# Scope

```
if ('serviceWorker' in navigator) {  
  navigator.serviceWorker.register('/service-worker.js')  
    .then(function(reg) {  
      console.log('Service Worker Registered', reg);  
    })  
    .catch(function(err) {  
      console.log('Error registering Service Worker', err);  
    });  
}
```

# Scope

```
if ('serviceWorker' in navigator) {  
  navigator.serviceWorker.register('/service-worker.js')  
    .then(function(reg) {  
      console.log('Service Worker Registered', reg);  
    })  
    .catch(function(err) {  
      console.log('Error registering Service Worker', err);  
    });  
}
```

# Scope

```
if ('serviceWorker' in navigator) {  
  navigator.serviceWorker.register('/scripts/s-worker.js)  
    .then(function(reg) {  
      console.log('Service Worker Registered', reg);  
    })  
    .catch(function(err) {  
      console.log('Error registering Service Worker', err);  
    });  
}
```

# Scope

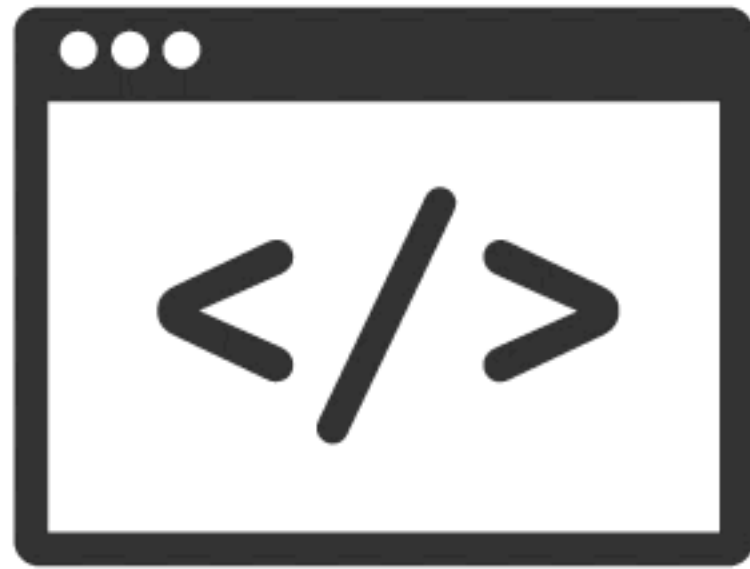
```
if ('serviceWorker' in navigator) {  
  navigator.serviceWorker.register('/service-worker.js')  
    .then(function(reg) {  
      console.log('Service Worker Registered', reg);  
    })  
    .catch(function(err) {  
      console.log('Error registering Service Worker', err);  
    });  
}
```

# Caching strategies

You are in control of the network



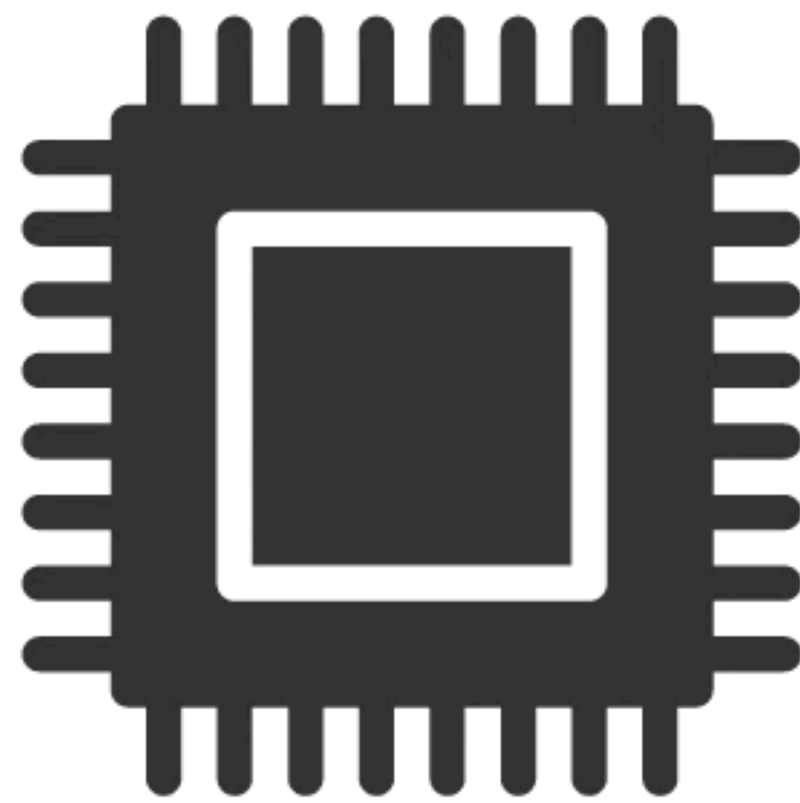
# Cache, Falling Back to Network



Page



Network

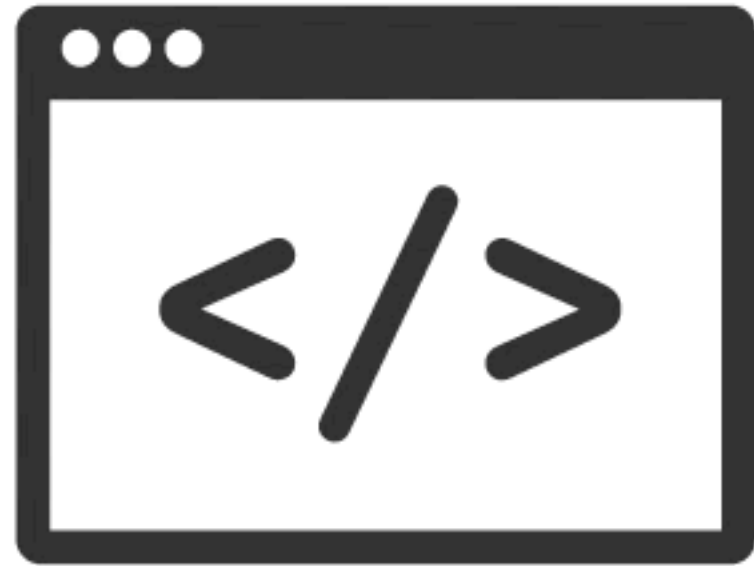


ServiceWorker



Cache

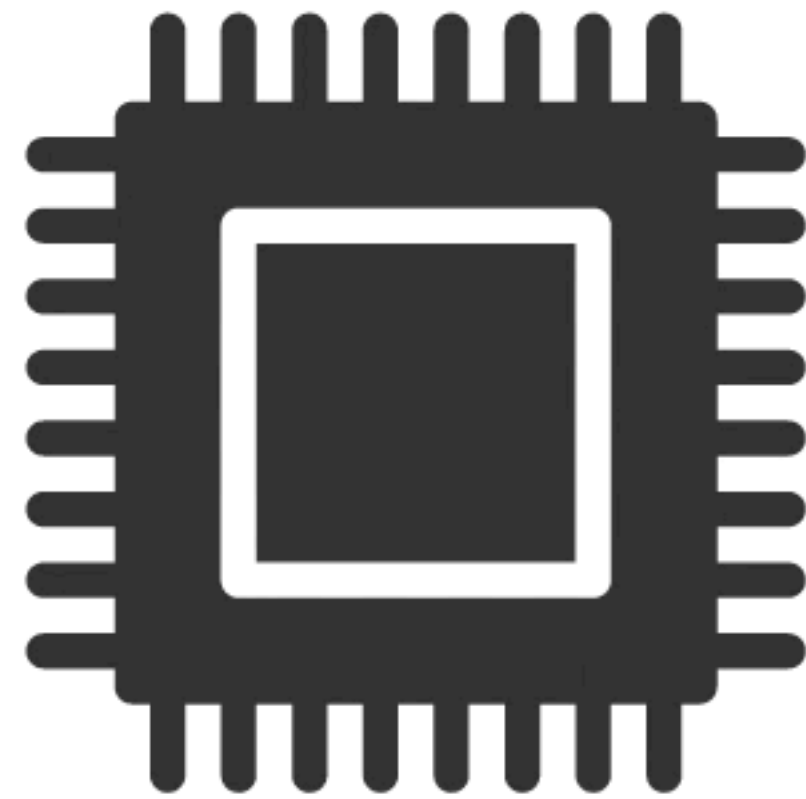
# Network, Falling Back to Cache



Page



Cache

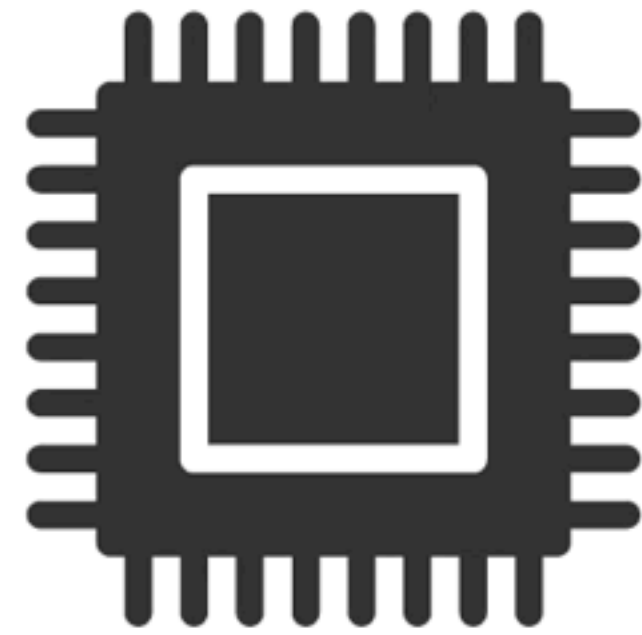


ServiceWorker

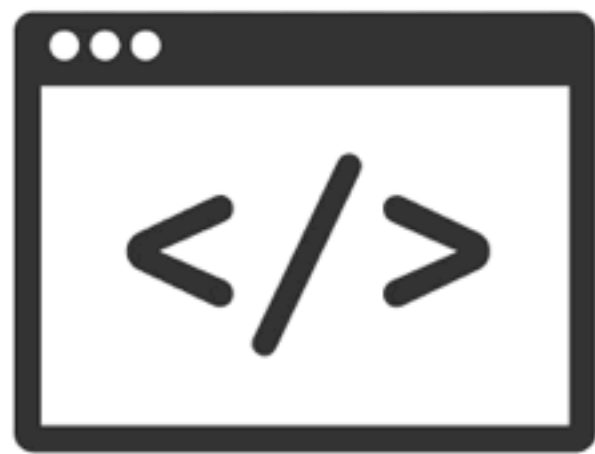


Network

# Cache then Network



ServiceWorker



Page

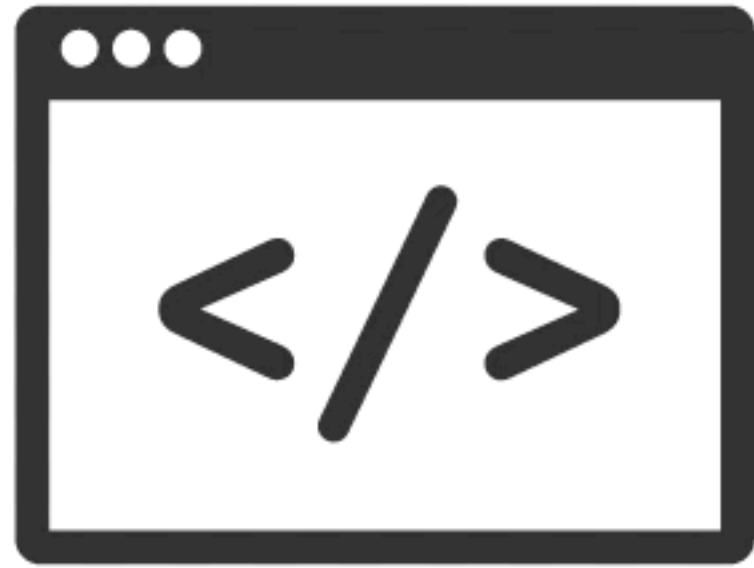


Cache



Network

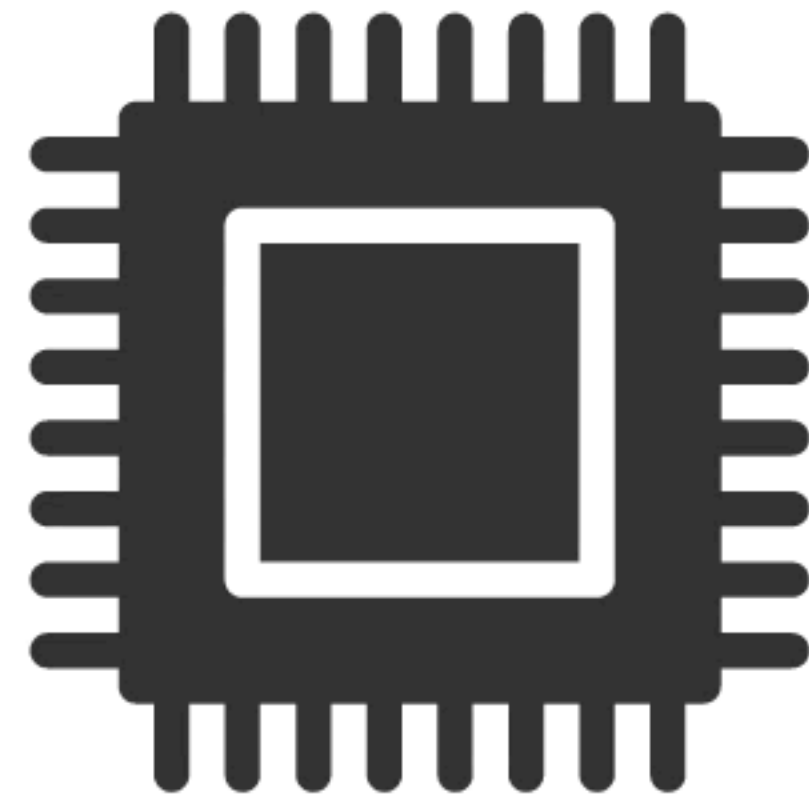
# Generic Fallback



Page



Cache

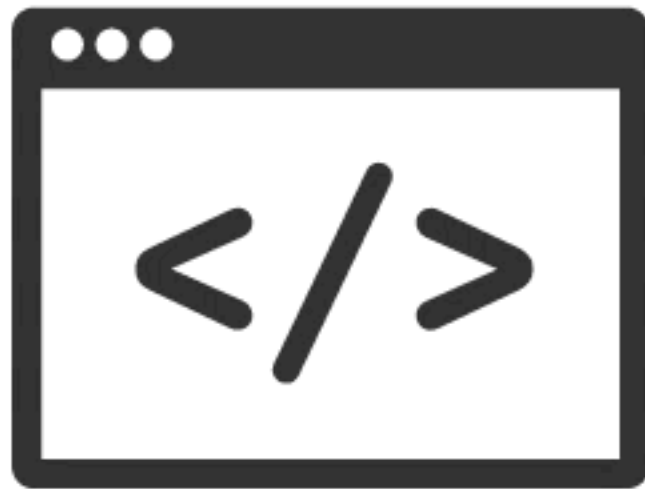


ServiceWorker

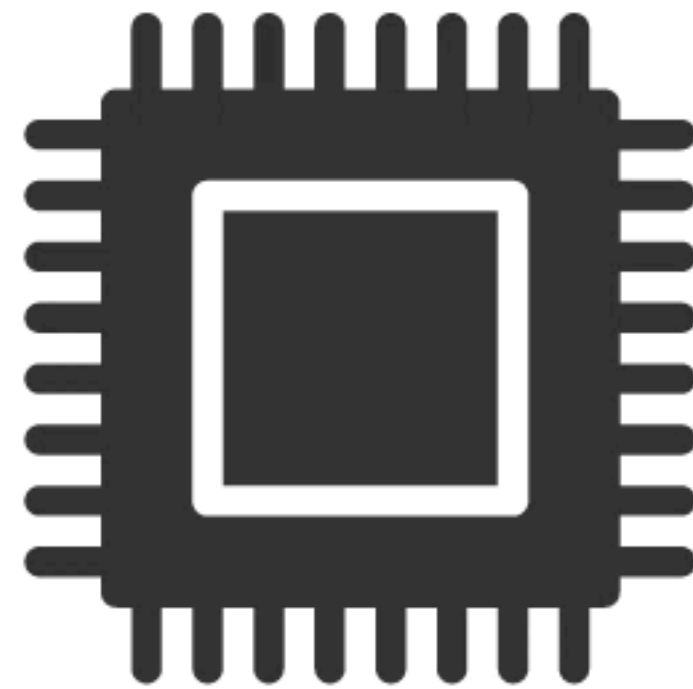


Network

# Cache and Network Race



Page



ServiceWorker



Cache



Network





YOU ARE IN  
control





Tools

# Chrome Dev Tools

The screenshot displays the Chrome DevTools interface with the 'Application' tab selected. The browser window shows a 'Weather PWA' at the URL `https://weather-pwa-sample.firebaseio.com/step-08/`. The page content shows weather information for New York, NY, including the current temperature (56°F), humidity (56%), wind (25 mph), and a 7-day forecast.

**Application Tab Overview:**

- Service Workers:** Shows a service worker for the URL `https://weather-pwa-sample.firebaseio.com/step-08/`. It is active (green dot) and running. The source is `service-worker.js`, received on 9/1/2016 at 10:01:52 PM. The status is '#15031 activated and is running'. The client is the current page.
- Storage:** Includes Local Storage, Session Storage, IndexedDB, Web SQL, and Cookies.
- Cache:** Includes Cache Storage and Application Cache.
- Frames:** Shows the 'top' frame.

**Weather PWA Details:**

**New York, NY**  
Thu, 21 Jul 2016 09:00 PM EDT  
Windy

**56°F**  
Humidity: 56%  
Wind: 25 mph 195°  
Sunrise: 5:43 am  
Sunset: 8:21 pm

Fri	Sat	Sun	Mon	Tue	Wed	Thu
86° 70°	94° 73°	95° 78°	75° 89°	89° 77°	92° 79°	89° 77°

# Chrome Dev Tools: the cache

Weather PWA

12:29

Weather PWA

New York, NY  
Thu, 01 Sep 2016 10:00 PM EDT  
Partly Cloudy  
73°F  
Humidity: 80%  
Wind: 14 mph 0°  
Sunrise: 6:24 am  
Sunset: 7:27 pm  
Fri 80° 73°  
Sat 80° 66°  
Sun 78° 67°  
Mon 71° 66°  
Tue 76° 65°  
Wed 84° 68°  
Thu 87° 69°

Application

Manifest  
Service Workers  
Clear storage

Storage

Local Storage  
Session Storage  
IndexedDB  
Web SQL  
Cookies

Cache

Cache Storage  
weatherPWA-step-8-1 -  
weatherData-v1 - https://  
Application Cache

Frames

top

Elements  
Console  
Sources  
Network  
Timeline  
Profiles  
Application  
Security  
Audits

#	Request	Response
0	https://weather-pwa-sample.firebaseio.com/images/...	OK
1	https://weather-pwa-sample.firebaseio.com/images/...	OK
2	https://weather-pwa-sample.firebaseio.com/images/...	OK
3	https://weather-pwa-sample.firebaseio.com/images/...	OK
4	https://weather-pwa-sample.firebaseio.com/images/...	OK
5	https://weather-pwa-sample.firebaseio.com/images/...	OK
6	https://weather-pwa-sample.firebaseio.com/images/...	OK
7	https://weather-pwa-sample.firebaseio.com/images/...	OK
8	https://weather-pwa-sample.firebaseio.com/images/...	OK
9	https://weather-pwa-sample.firebaseio.com/images/...	OK
10	https://weather-pwa-sample.firebaseio.com/images/...	OK
11	https://weather-pwa-sample.firebaseio.com/images/...	OK
12	https://weather-pwa-sample.firebaseio.com/images/...	OK
13	https://weather-pwa-sample.firebaseio.com/step-08/	OK
14	https://weather-pwa-sample.firebaseio.com/step-08...	OK
15	https://weather-pwa-sample.firebaseio.com/step-08...	OK
16	https://weather-pwa-sample.firebaseio.com/styles/i...	OK



# Chrome Dev Tools: the cache

The screenshot shows the Chrome DevTools interface with the 'Application' tab selected. The left pane displays the 'Weather PWA' application running on a Nexus 5X. The right pane shows the 'Clear storage' dialog for the URL `https://weather-pwa-sample.firebaseio.com/step-08/`. The 'Application' section on the right has the following checked options:

- ☒ Unregister service workers
- ☒ Local and session storage
- ☒ Indexed DB
- ☒ Web SQL
- ☒ Cookies
- ☒ Cache storage
- ☒ Application cache

The 'Clear selected' button is visible at the bottom of the right pane.

**Weather PWA**

New York, NY  
Thu, 01 Sep 2016 10:00 PM EDT  
Partly Cloudy

73°F

Humidity: 80%  
Wind: 14 mph 0°  
Sunrise: 6:24 am  
Sunset: 7:27 pm

Fri	Sat	Sun	Mon	Tue	Wed	Thu
80° 73°	80° 66°	78° 67°	71° 66°	76° 65°	84° 68°	87° 69°

GoogleChrome/sw-toolbox: A c x

GitHub, Inc. [US] | <https://github.com/GoogleChrome/sw-toolbox>

This repository Search Pull requests Issues Gist

GoogleChrome / sw-toolbox

Watch 87 Star 1,411 Fork 112

Code Issues 39 Pull requests 3 Wiki Pulse Graphs

A collection of service worker tools for offlining runtime requests <https://googlechrome.github.io/sw-toolbox/>

229 commits 8 branches 13 releases 11 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

jeffposnick committed with addyosmani Adds new notifyOnCacheUpdate cache option (#174) Latest commit 3590616 a day ago

doc-pages	Adds new notifyOnCacheUpdate cache option (#174)	a day ago
lib	Adds new notifyOnCacheUpdate cache option (#174)	a day ago
project	Fixes #157	2 months ago
recipes	Adds new notifyOnCacheUpdate cache option (#174)	a day ago
test	Adds new notifyOnCacheUpdate cache option (#174)	a day ago

# Service Worker toolbox

# Cache first in sw-toolbox

```
importScripts('/sw-toolbox.js');  
  
toolbox.router.get('/(.*)', toolbox.cacheFirst, {  
  origin: /\.googleapis\.com$/  
});
```



GoogleChrome/sw-precache: x

GitHub, Inc. [US] | https://github.com/GoogleChrome/sw-precache

This repository Search Pull requests Issues Gist

GoogleChrome / sw-precache

Watch 59 Star 1,407 Fork 111

Code Issues 30 Pull requests 0 Wiki Pulse Graphs

A node module to generate service worker code that will precache specific resources so they work offline.

389 commits 7 branches 5 releases 20 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

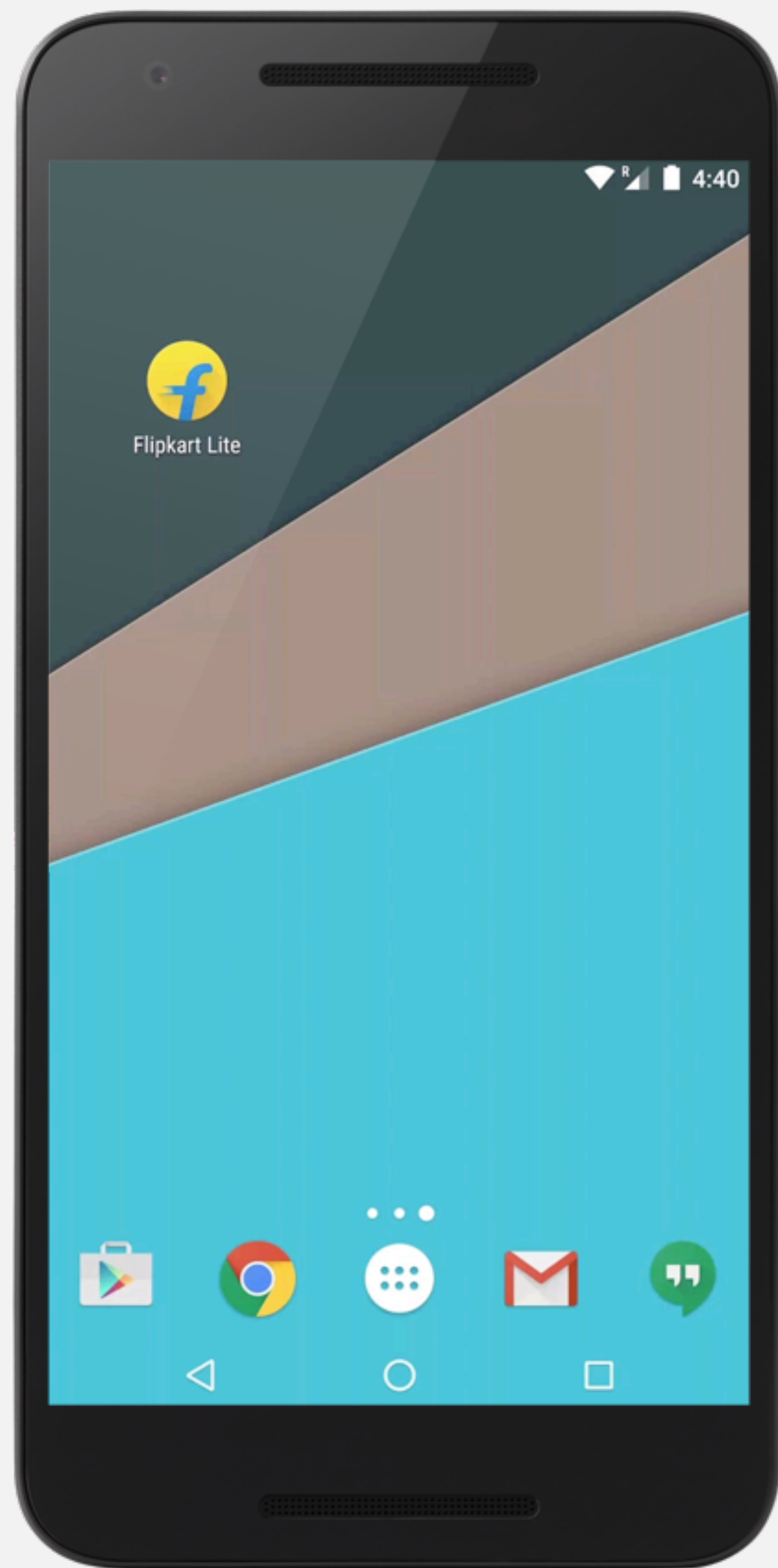
jeffposnick Merge remote-tracking branch 'origin/master' Latest commit 39cc18e 2 days ago

app-shell-demo	Use a thumbnail image	2 months ago
demo	Quick demo update.	2 days ago
lib	Linting cleanup.	a month ago
test	Linting cleanup.	a month ago
eslintignore	Switching over to google linting rules	9 months ago

# sw-precache Node module

# sw-precache

```
precache.write('service-worker.js', {  
  staticFileGlobs: ['**/*.html,css'],  
  runtimeCaching: [  
    {urlPattern: '/', handler: 'fastest'},  
    {urlPattern: '/api/*', handler: 'networkFirst',  
      options: {networkTimeoutSeconds: 5}  
    },  
    {urlPattern: '/posts/*', handler: 'cacheFirst'},  
    {urlPattern: '/styles/*', handler: 'cacheOnly'},  
    {urlPattern: '/inbox.json', handler: 'networkOnly'}  
  ],  
});
```



Service Workers:  
the core underpinning  
Progressive Web Apps

Thank You!

**PROGRESSIVE WEB APP**

ROADSHOW 2016 