

# Formation Mobile

## Hybride

### Day 4

Cordova, Ionic, Firebase

# Agenda

---

1. Correction travaux pratique

---

2. Plugins Cordova

---

3. ngCordova : Camera

---

4. Sass

---

5. Variables Ionic Sass

---

6. Material Design

---

---

---

---

---

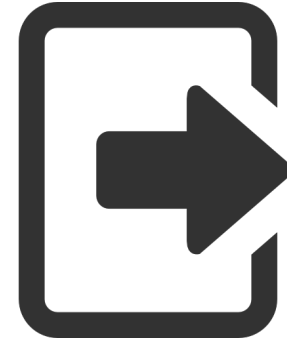
---

---

---

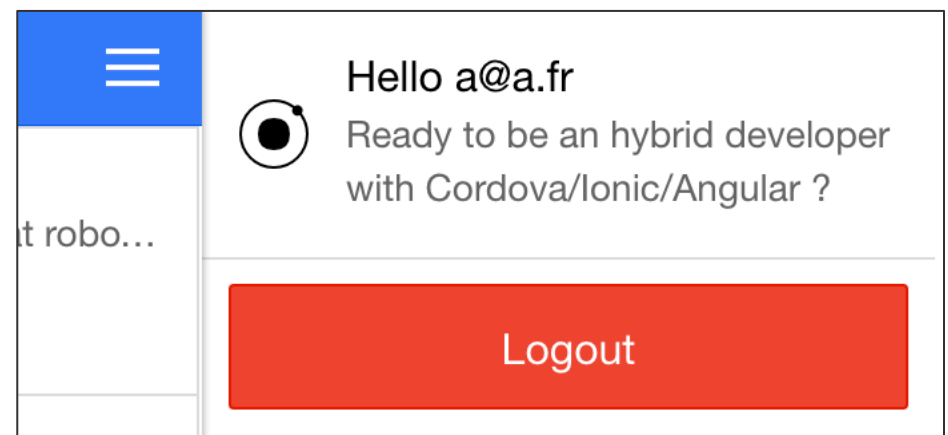
---

Objectif: logout



```
var ref = new Firebase('https://ionic-places.firebaseio.com/');  
var auth = $firebaseAuth(ref);  
  
auth.$unauth();
```

Afficher l'email utilisateur dans le menu



Correction: \$ git checkout valid-10

# Plugins cordova API natives

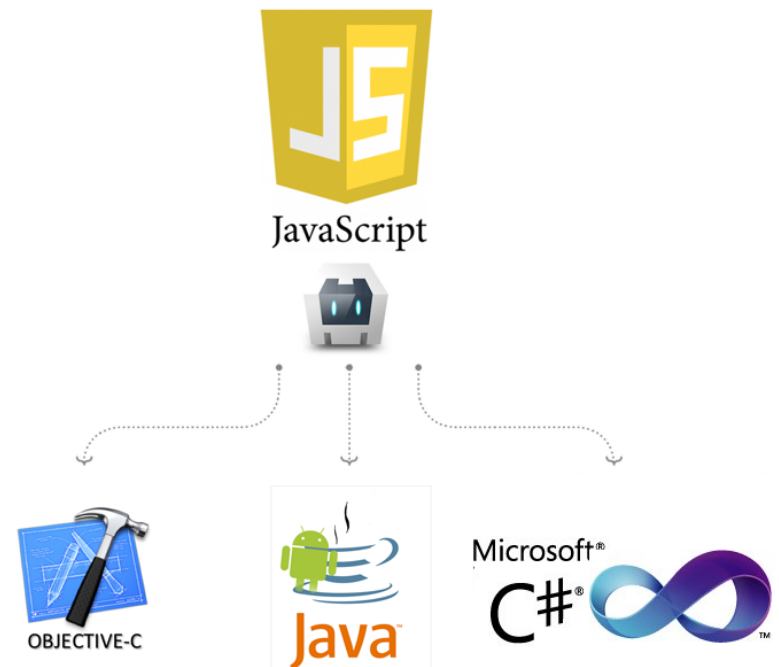
# Possibilité de créer son plugin cordova

## Plugman



Développement spécifique natif pour chaque plateforme  
au sein d'un plugin appelé depuis JavaScript

---



# Java Android



```
demo-plugin.java  * demo-plugin.m  * demo-plugin.js  *
1 package org.super.cool;
2
3 import org.apache.cordova.CordovaPlugin;
4 import org.apache.cordova.CallbackContext;
5
6 import org.json.JSONArray;
7 import org.json.JSONException;
8 import org.json.JSONObject;
9
10 /**
11  * This class echoes a string called from JavaScript.
12  */
13 public class demo-plugin extends CordovaPlugin {
14
15     @Override
16     public boolean execute(String action, JSONArray args, CallbackContext callbackContext) throws JSONException {
17         if (action.equals("coolMethod")) {
18             String message = args.getString(0);
19             this.coolMethod(message, callbackContext);
20             return true;
21         }
22         return false;
23     }
24
25     private void coolMethod(String message, CallbackContext callbackContext) {
26         if (message != null && message.length() > 0) {
27             callbackContext.success(message);
28         } else {
29             callbackContext.error("Expected one non-empty string argument.");
30         }
31     }
32 }
33
```

# Objective-C



```
demo-plugin.java * demo-plugin.m * demo-plugin.js *
1  /******* demo-plugin.m Cordova Plugin Implementation *****/
2
3  #import <Cordova/CDV.h>
4
5  @interface demo-plugin : CDVPlugin {
6      // Member variables go here.
7  }
8
9  - (void)'':(CDVInvokedUrlCommand*)command;
10 @end
11
12 @implementation demo-plugin
13
14 - (void)coolMethod:(CDVInvokedUrlCommand*)command
15 {
16     CDVPluginResult* pluginResult = nil;
17     NSString* echo = [command.arguments objectAtIndex:0];
18
19     if (echo != nil && [echo length] > 0) {
20         pluginResult = [CDVPluginResult resultWithStatus:CDVCommandStatus_OK messageAsString:echo];
21     } else {
22         pluginResult = [CDVPluginResult resultWithStatus:CDVCommandStatus_ERROR];
23     }
24
25     [self.commandDelegate sendPluginResult:pluginResult callbackId:command.callbackId];
26 }
27
28 @end
29
```



```
demo-plugin.java  × demo-plugin.m  × demo-plugin.js  × demo-plugin.cs  ×
1  using System;
2
3  namespace WPCordovaClassLib.Cordova.Commands
4  {
5      public class demo-plugin : BaseCommand
6      {
7          public void coolMethod(string message)
8          {
9              PluginResult result;
10             if (message != "")
11             {
12                 result = new PluginResult(PluginResult.Status.OK, message);
13             } else
14             {
15                 result = new PluginResult(PluginResult.Status.ERROR, message);
16             }
17             DispatchCommandResult(result);
18         }
19     }
20 }
21 }
```




# JavaScript




```
demo-plugin.java  * demo-plugin.m  * demo-plugin.js  *
1  var exec = require('cordova/exec');
2
3  exports.coolMethod = function(arg0, success, error) {
4      exec(success, error, "demo-plugin", "coolMethod", [arg0]);
5  };
6
```


# D'autres plugins ?

**Beaucoup de plugins disponibles ...** avec un niveau de qualité très variable


 APACHE CORDOVA  
PLUGINS REGISTRY


Discover plugins for your  
Apache Cordova project.



Find a plugin... 

[Browse all 939 plugins](#)

  
Contribute  
Want to author & publish plugins?  
[Please read our docs.](#)

  
Utilize  
Need to install or remove a plugin?  
[Check the documentation.](#)

### Most Downloaded

1786164	<a href="#">org.apache.cordova.device</a>
1310918	<a href="#">org.apache.cordova.console</a>
926346	<a href="#">org.apache.cordova.file</a>

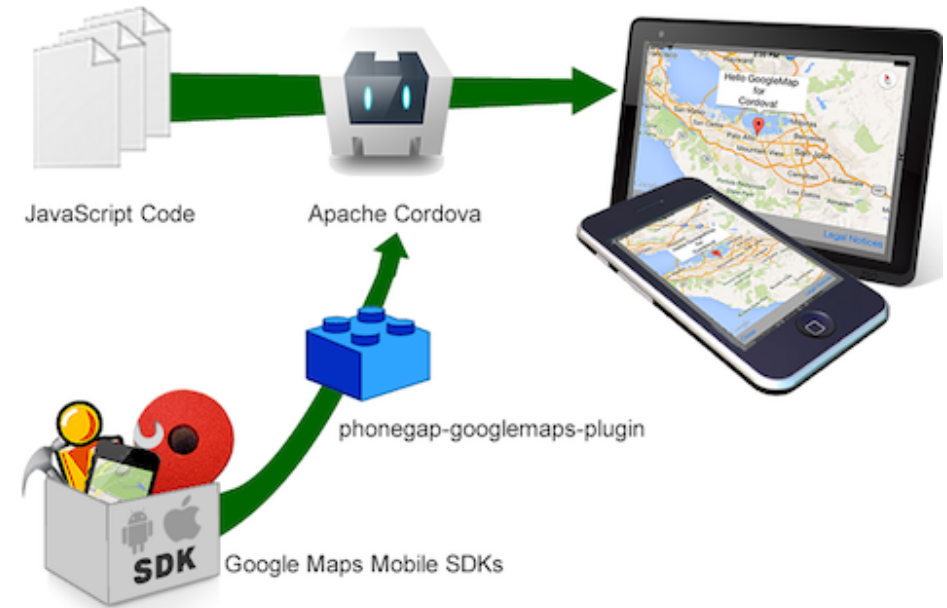
### Last Updated

4 hours ago	<a href="#">com.heyztz.jipwrapper</a>
4 hours ago	<a href="#">com.heyztz.mscwrapper</a>
4 hours ago	<a href="#">com.cranberrygame.cordova.plugin.ad.admob</a>

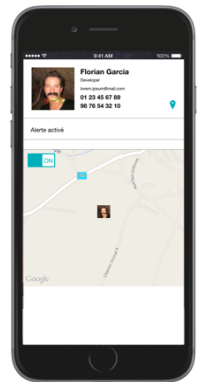
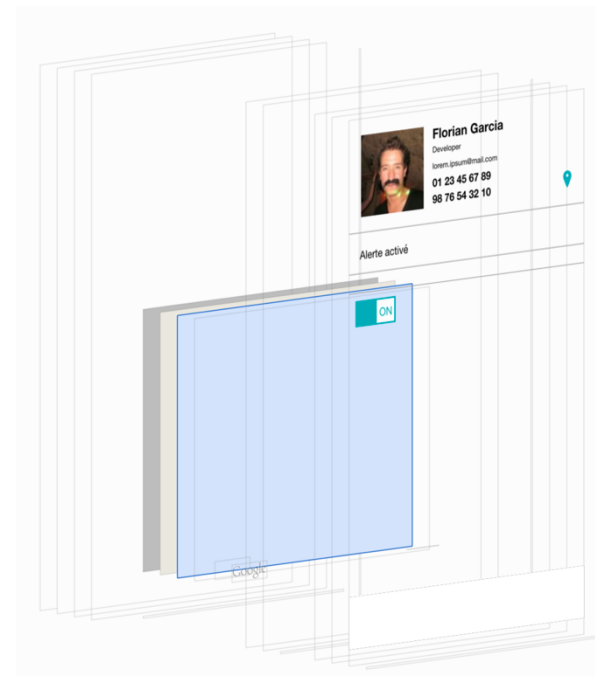
<http://plugins.cordova.io>

# Exemple de plugin: Google Maps

- **plugin.google.maps**



- **Le layer natif se trouve en dessous de la WebView**



plugin.google.maps



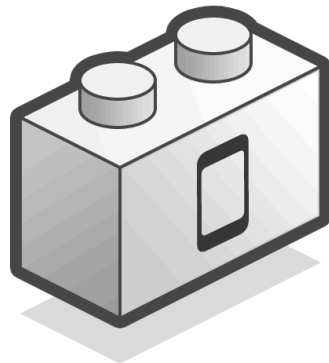
## Points forts

- **Performance Native (IOS/Android)**
- Une seule instance garder en cache
- Ajout de marqueurs
- Animation et déplacement caméra à l'intérieur de la map
- Customisation des marqueurs via une image



## Points faibles

- Animation : léger décalage entre la vue HTML et la map
- Superposition HTML + interactivité à configurer
- Customisation des marqueurs avancée
- Pas de possibilité de grouper les marqueurs.  
(Seulement en beta sur android)



## Du temps de test nécessaire sur chaque plugin.

Des limitations et des divergences entre les plateformes peuvent exister. Il faut donc bien connaître le périmètre des différents plugins pour ne pas se trouver piégé.

# ngCordova:

63+ extensions intégrées à AngularJs



# ngCordova

Built with ❤️ by [Ionic](#)

## Avantages

- Facilement intégré avec Angular
- Une documentation unifié
- Des plugins testés et validés

## CORDOVA WITH THE POWER OF **ANGULARJS**

ngCordova is a collection of **63+** AngularJS extensions on top of the Cordova API that make it easy to build, test, and deploy Cordova mobile apps with AngularJS.



+



=



# ngCordova:

## Installation

```
$ bower install ngCordova
```

Include `ng-cordova.js` or `ng-cordova.min.js` in your `index.html` file **before** `cordova.js` and **after** your AngularJS / Ionic file (since ngCordova depends on AngularJS).

```
<script src="lib/ngCordova/dist/ng-cordova.js"></script>
<script src="cordova.js"></script>
```

## Inject as an Angular dependency

Then, include `ngCordova` as a dependency in your angular module:

```
angular.module('myApp', ['ngCordova'])
```

---

## Device ready event

### Wrap each plugin call with the `deviceready` event - *important !*

Before each plugin you must check if your device has fully loaded, and if the plugins are available using a native cordova event called `deviceready`. Implement it like so:

```
document.addEventListener("deviceready", function () {
  $cordovaPlugin.someFunction().then(success, error);
}, false);

// OR with IONIC

$ionicPlatform.ready(function() {
  $cordovaPlugin.someFunction().then(success, error);
});
```

# ngCordova:



correction:

Camera : \$ git checkout valid-12

Take a picture





# Customisation avec Sass

## SASS c'est quoi ?

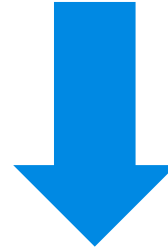
**SASS** est un système qui étend les possibilités de **CSS**. Il permet d'être plus productif et ainsi d'avoir un **CSS** mieux organisé et plus facile à maintenir.

---

- Né dans la communauté **Ruby On Rail**
- Il existe une **Gem Ruby**

SASS c'est  
quoi ?

Écrire son script **Sass**



Le préprocesseur interprète le  
script **Sass**



Les fichiers **css** sont produits  
par le préprocesseur

## 2 syntaxes

### SCSS

---

Syntaxe classique qui utilise **brackets** et **semicolon**, tout ce qui est valide **css** est valide **scss**.  
C'est la syntaxe la plus utilisée.

### SASS

---

À la place d'utiliser les **brackets** et **semicolon**, ce dernier utilise l'indentation pour séparer les blocs de codes.  
Il utilise l'extension **.sass**

# Les Features



## 6 features principales

1. Variables
2. Nesting
3. Mixins
4. Partial
5. @extend
6. Function

# Les variables

---

Les variables vous permettent d'assigner une valeur afin de la réutiliser et de vous en souvenir plus facilement.

Fonctionne avec tout type : *Boolean, color, text, string ....*

## Fichier Sass:

```
$font-stack: Helvetica, sans-serif;
$primary-color: #333;

body {
  font: 100% $font-stack;
  color: $primary-color;
}
```

## Fichier CSS compilé:

```
body {
  font: 100% Helvetica, sans-serif;
  color: #333;
}
```

# Nesting

---

Si vous avez passé du temps à écrire du **CSS** dans un projet, vous avez probablement écrit beaucoup de lignes comme celle-ci:

```
nav.tabbed-navigation { ... }  
nav.tabbed-navigation a { ... }  
nav.tabbed-navigation a span { ... }
```

Cela crée beaucoup de lignes répétées dans votre feuille de style qui sont difficiles à lire, avec une forte dépendance à l'égard des sélecteurs CSS descendants

Si le nom de la classe de l'élément < nav > change , vous devrez réécrire les trois sélecteurs CSS dans l'exemple. Mais avec Sass ...

# Nesting

```
nav {  
  ul {  
    margin: 0;  
    padding: 0;  
    list-style: none;  
  }  
  
  li { display: inline-block; }  
  
  a {  
    display: block;  
    padding: 6px 12px;  
    text-decoration: none;  
  }  
}
```

---

```
nav ul {  
  margin: 0;  
  padding: 0;  
  list-style: none;  
}  
  
nav li {  
  display: inline-block;  
}  
  
nav a {  
  display: block;  
  padding: 6px 12px;  
  text-decoration: none;  
}
```



# Mixins

Les **mixins** vous permettent de définir des styles qui pourront être réutilisés partout dans votre **CSS**.

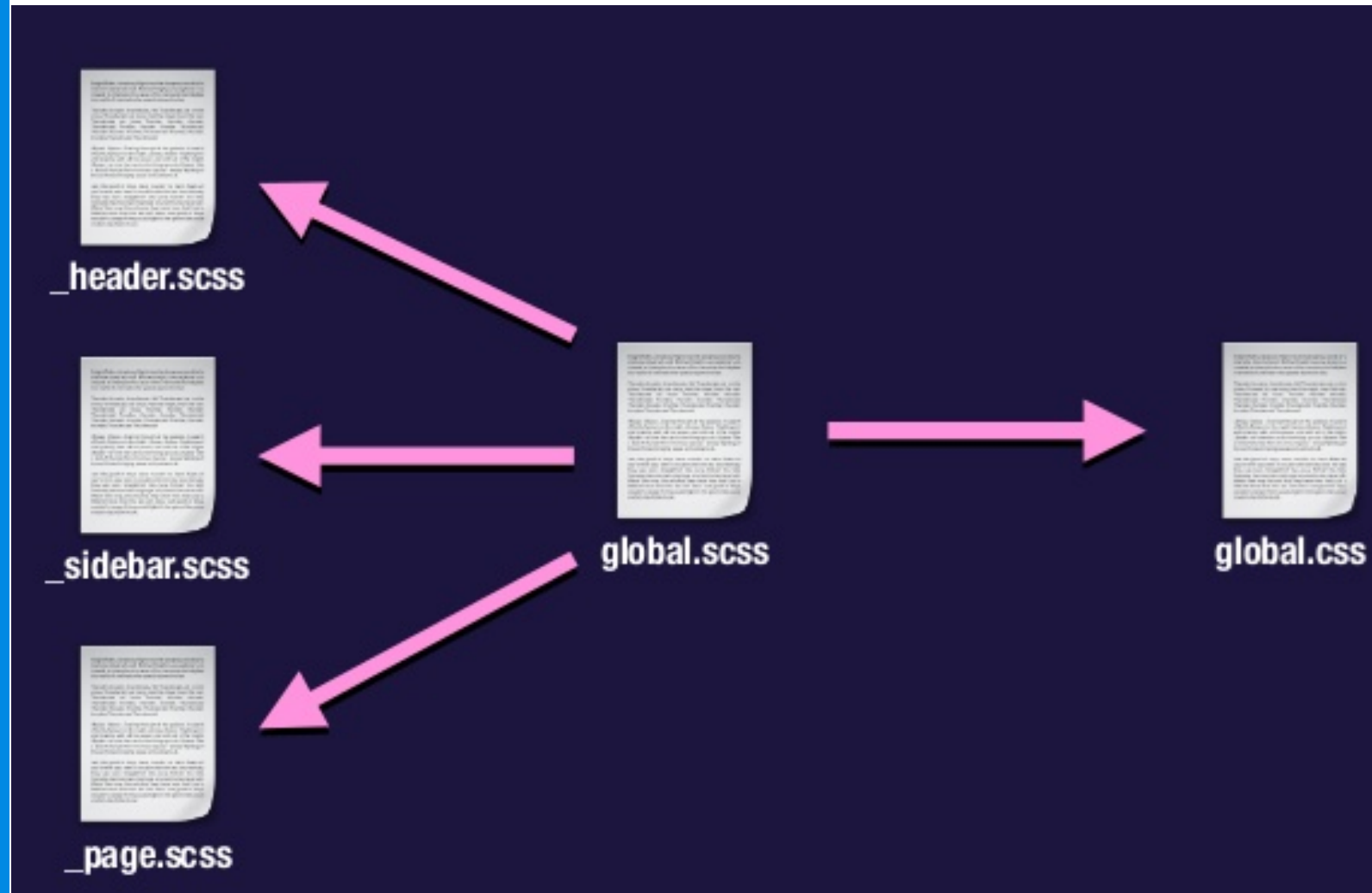
Vous pouvez même passer des valeurs pour rendre votre mixin plus souple .

```
@mixin border-radius($radius) {  
    -webkit-border-radius: $radius;  
    -moz-border-radius: $radius;  
    -ms-border-radius: $radius;  
    border-radius: $radius;  
}  
  
.box { @include border-radius(10px); }
```

```
.box {  
    -webkit-border-radius: 10px;  
    -moz-border-radius: 10px;  
    -ms-border-radius: 10px;  
    border-radius: 10px;  
}
```

# Partials

**Sass** permet de séparer les feuilles de style afin de rendre le code plus modulable et ainsi mieux l'organiser.



# @extend

La directive @extend vous permet de partager un ensemble de propriétés CSS d'un sélecteur à l'autre.

```
.message {  
  border: 1px solid #ccc;  
  padding: 10px;  
  color: #333;  
}  
  
.success {  
  @extend .message;  
  border-color: green;  
}  
  
.error {  
  @extend .message;  
  border-color: red;  
}  
  
.warning {  
  @extend .message;  
  border-color: yellow;  
}
```

```
.message, .success, .error, .warning {  
  border: 1px solid #cccccc;  
  padding: 10px;  
  color: #333;  
}  
  
.success {  
  border-color: green;  
}  
  
.error {  
  border-color: red;  
}  
  
.warning {  
  border-color: yellow;  
}
```

Faire des maths peut s'avérer utile

```
.container { width: 100%; }

article[role="main"] {
  float: left;
  width: 600px / 960px * 100%;
}

aside[role="complimentary"] {
  float: right;
  width: 300px / 960px * 100%;
}
```

---

```
.container {
  width: 100%;
}

article[role="main"] {
  float: left;
  width: 62.5%;
}

aside[role="complimentary"] {
  float: right;
  width: 31.25%;
}
```

## Natives

```
$red: rgb(255, 0, 0);
```

---

## Personnalisés

```
@function addition($a, $b) {  
  @return $a + $b;  
}  
  
@function shade($color, $percent) {  
  @return mix(#000, $color, $percent);  
}  
  
body {  
  background: shade(grey, 50%);  
}
```



```
/*  
To customize the look and feel of Ionic, you can override the variables  
in ionic's _variables.scss file.
```

*For example, you might change some of the default colors:*

```
$light:                #fff !default;  
$stable:               #f8f8f8 !default;  
$positive:             #4a87ee !default;  
$calm:                 #43cee6 !default;  
$balanced:            #66cc33 !default;  
$energized:           #f0b840 !default;  
$assertive:           #ef4e3a !default;  
$royal:               #8a6de9 !default;  
$dark:                #444 !default;  
*/
```

Gestion par template.

# Customisable avec Sass

- Customisation du composant **Toggle**



```
@include transition-timing-function(ease-in-out);
@include transition-duration($toggle-transition-duration);
@include transition-property((background-color, border));

display: inline-block;
box-sizing: border-box;
width: $toggle-width;
height: $toggle-height;
border: solid $toggle-border-width $toggle-off-border-color;
border-radius: $toggle-border-radius;
background-color: $toggle-off-bg-color;
content: ' ';
cursor: pointer;
pointer-events: none;
```

## Deux options :

- **Surcharge** CSS (attention aux performances)
- **Fork** du fichier Sass