

Formation Mobile Hybride

Cordova, Ionic

Présentation du cours



POUR QUI ?

Des développeurs, avec une connaissance Typescript et Angular 2 intéressés par le développement hybride. La connaissance de Sass ou Gulp est un plus.

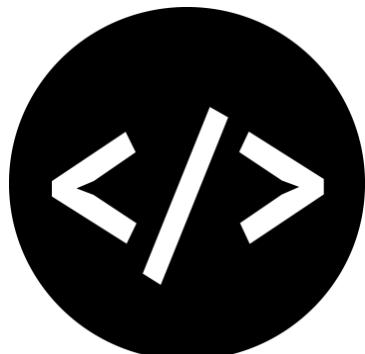


APPLICATION HYBRIDE

Nous allons parcourir ensemble le fonctionnement et l'utilisation des technologies : Ionic/Cordova



ionic



TRAVAUX PRATIQUES

Les parties théoriques seront accompagnées de travaux pratiques tout au long de la formation pour mener à bien un prototype que vous pourrez finaliser par vous même.

Qui suis je ?

Mickael Dumand



Développeur Front End
Freelance

#JavaScript #Angular #React #Redux #Webpack
#Babel #TypeScript #Cordova #Gulp/Grunt
#Sass #Ionic #Git

@MeKimak

Agenda

-
1. Contexte mobile hybride

 2. Installation des outils

 3. Les composants Ionic

 4. Ionic navigation

 5. Mise en place d'un prototype

Contexte Mobile Hybride

Créer un service mobile



SITE MOBILE

Élaboration d'un site offrant une expérience de lecture optimale sur tous les devices.

Ce site peut être « Responsive », Dédié au canal mobile ou Adaptatif



APPLICATION HYBRIDE

Un moteur natif qui encapsule du HTML5/JS fortement mutualisé entre les plateformes.

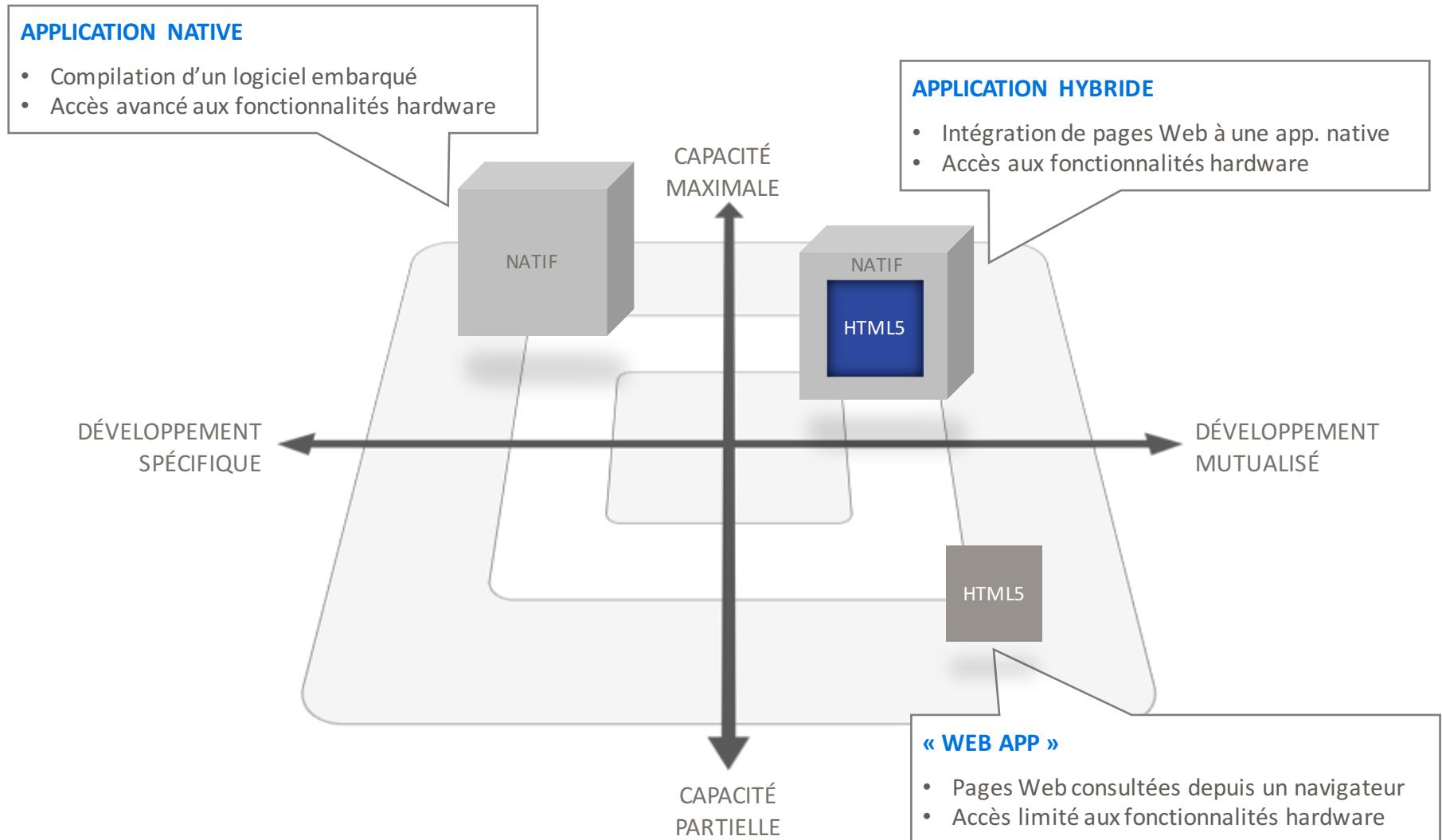
Permet d'utiliser les fonctionnalités natives du mobile et une distribution par les stores.



NATIF

Application développée dans un langage spécifique à un OS et distribuée par les stores.

Comment choisir ?



Application native



« je préfère le natif ! »



Application hybride



« C'est lent mais c'est moins cher »

**Pourquoi les gens
pensent ça Doc ?**

**Je trouve ça cool
l'hybride moi...**



→
Temps

**Monte Marty, je
vais t'expliquer.**



→
Temps

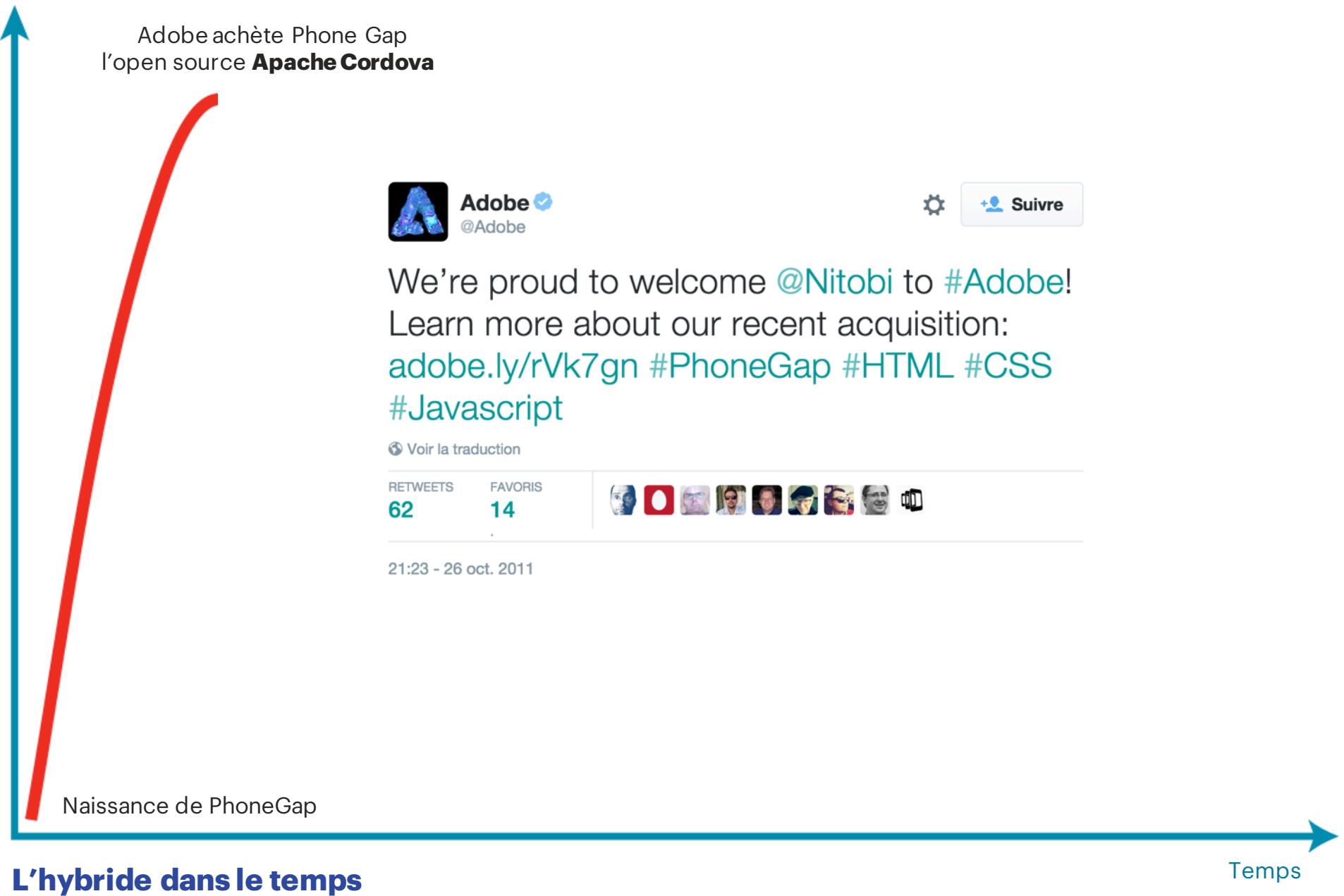
En 2009 la société Nitobi crée PhoneGap

Popularité



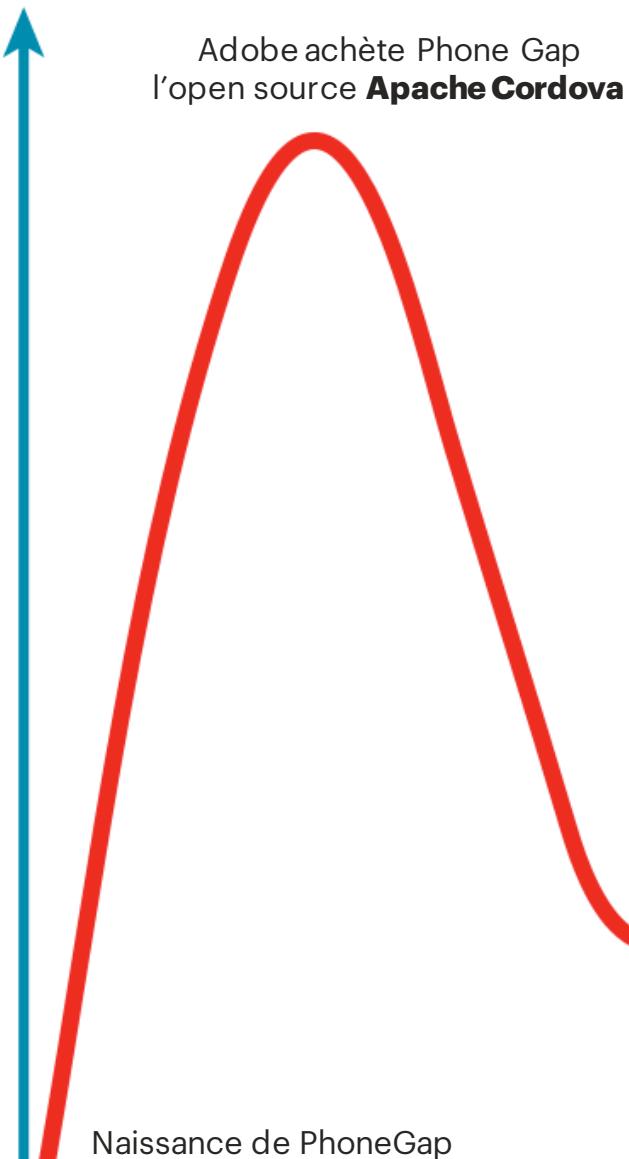
En 2011 Adobe rachète Nitobi et open source le projet

Popularité



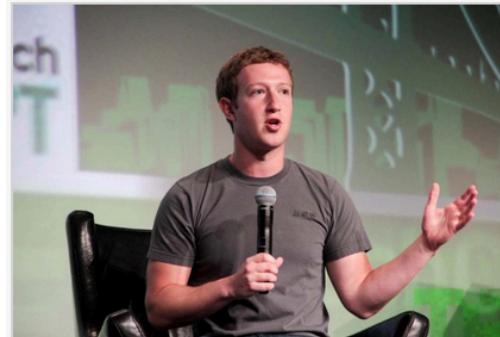
En 2012 Facebook abandonne HTML5 pour ses applications mobile

Popularité



Mark Zuckerberg: Our Biggest Mistake Was Betting Too Much On HTML5

Posted Sep 11, 2012 by *Drew Olanoff (@drew)*



Today, Mark Zuckerberg revealed that Facebook's mobile strategy relied too much on HTML5, rather than native applications.

L'hybride dans le temps

Temps

En 2012 Facebook abandonne HTML5 pour ses applications mobile

Popularité

Adobe achète Phone Gap
l'open source **Apache Cordova**



GOD HAS SPOKEN

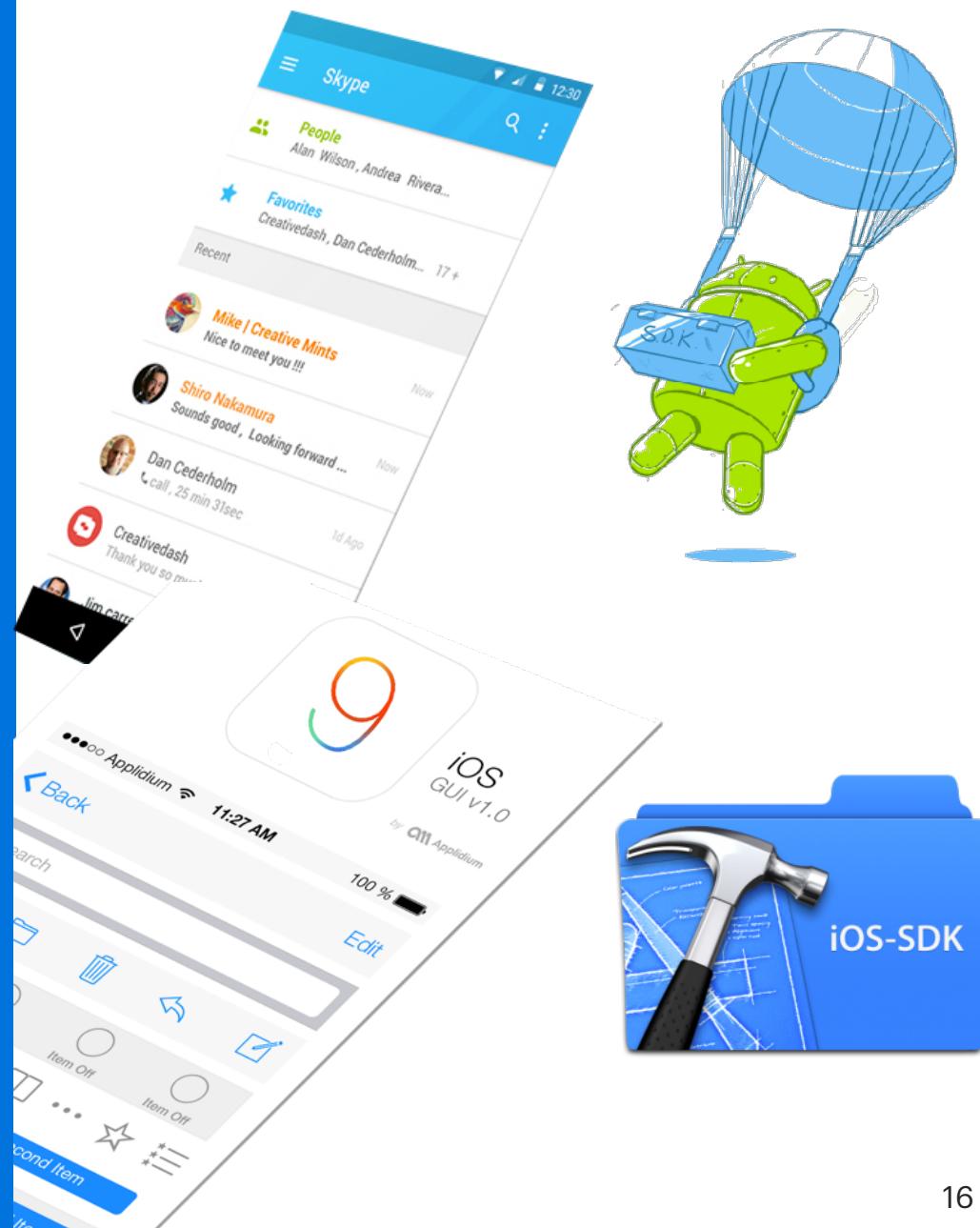
Il devient possible de développer des applications mobiles pour du natif

dans le temps

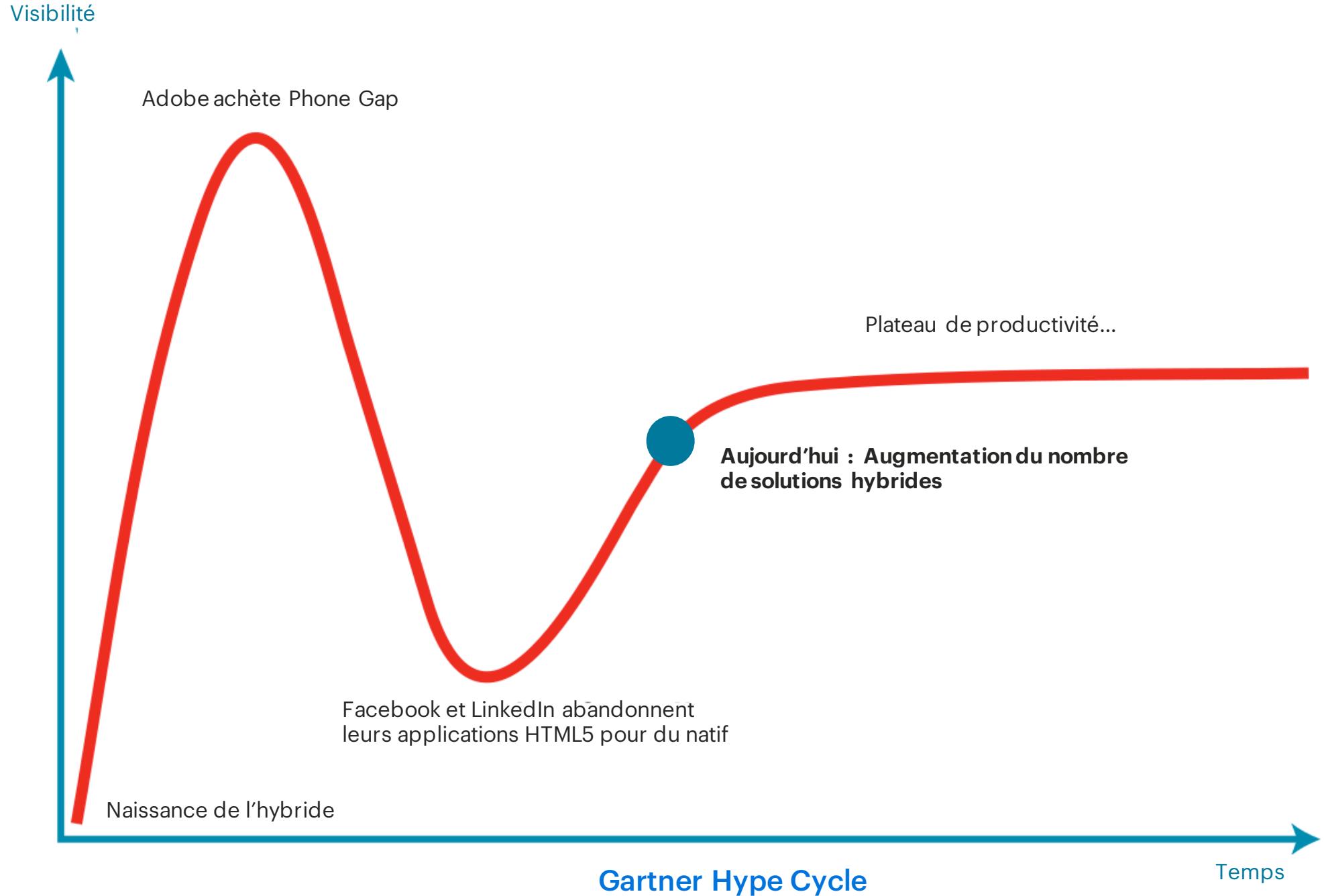
Temps

Le vrai problème ?

Aucun SDK HTML5 pour mobile !



Mais on est en 2016 : les choses changent !



Aperçu des solutions hybrides et alternatives



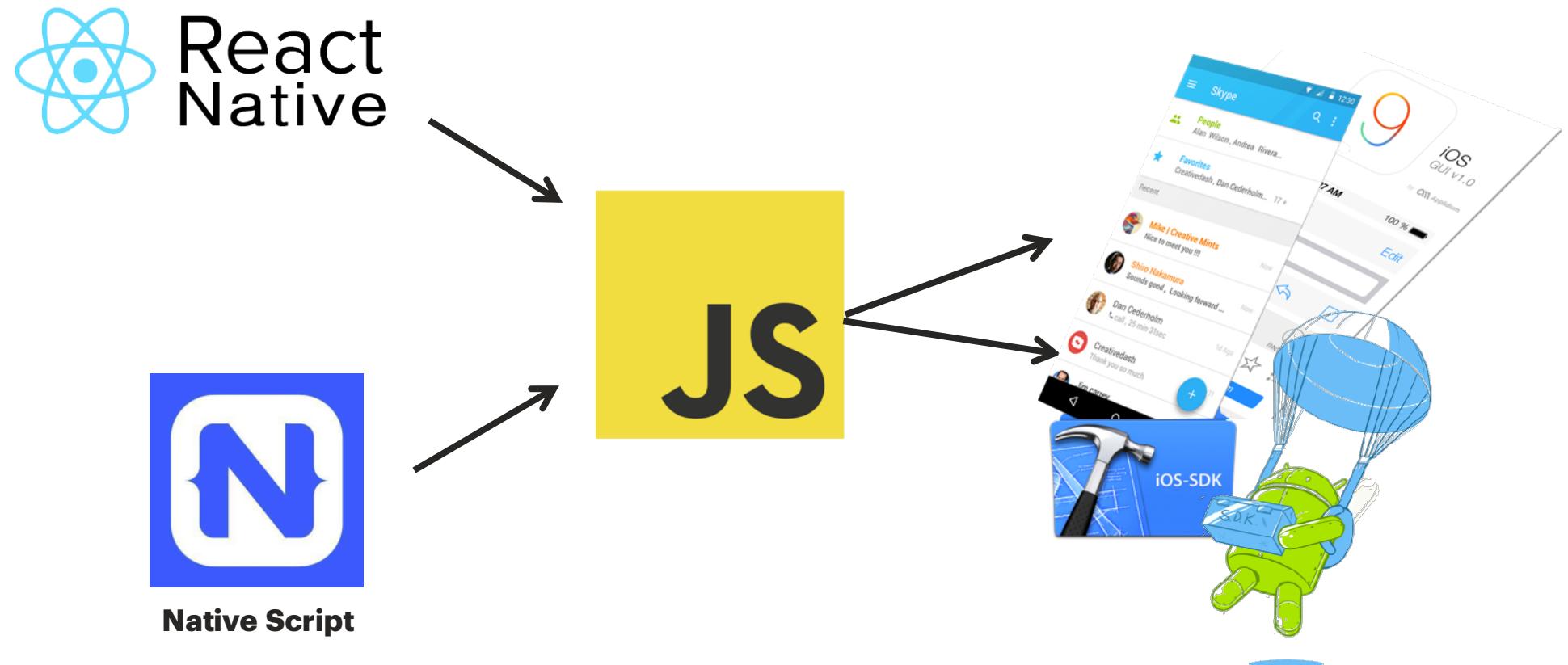


**LE MONDE SÉDIVISE EN
DEUX CATÉGORIES...**

Panorama des solutions hybrides



Piloter des composants natifs en JavaScript





Comparaison

Mobile Hybrid				 APACHE CORDOVA™			 doc		
 ionic	✓	✓	~	✓	✓	~	+++	free	open-source
 NativeScript	✓	✓	~	✓	✓	✗	++	free	open-source
 React Native	✓	✗	✗	✗	✗	✓	+	free	open-source
 appcelerator® titanium	✓	✓	✗	✗	✗	✗	++	\$259/mo Team	owner
 Onsen UI	✓	✓	✗	✓	✓	~	++	free	open-source
 famous.us	✓	✓	~	~	✓	~	-	free	open-source
 TouchstoneJS	✓	~	✗	✓	✗	✓	-	free	open-source
 Supersonic	~	~	✗	✗	✓	✓	---	free	open-source

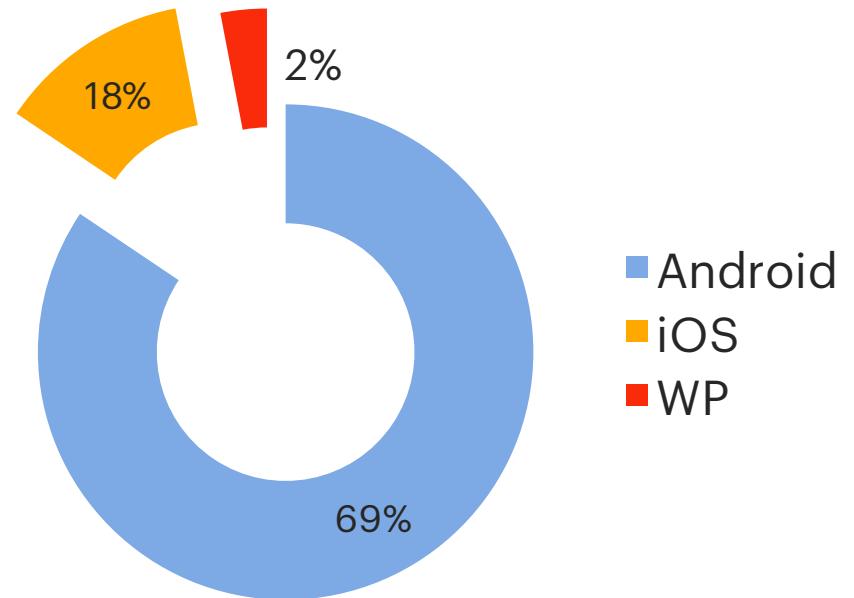
Operating System

Tooling

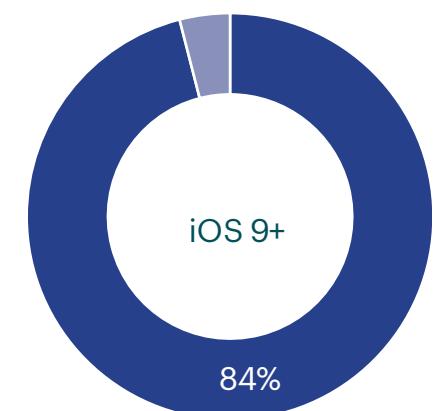
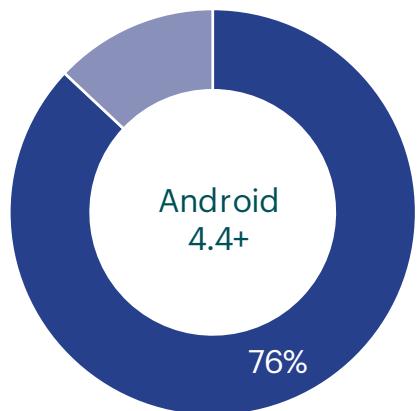
Community

Vision actuel du marché mobile mondial

Android et iOS dominent le monde



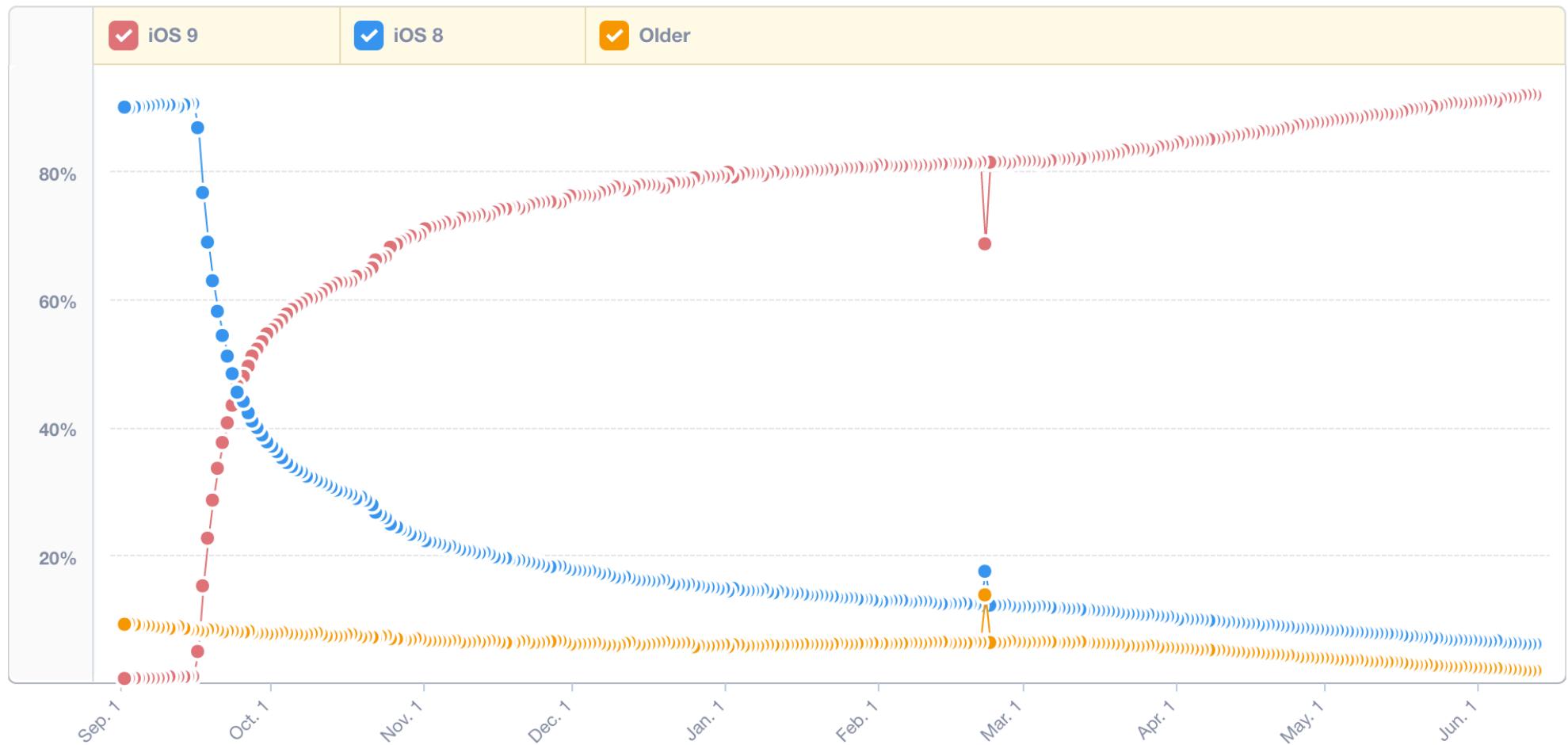
Fragmentation des OS majoritaires



iOS 9 adoption ▾

Sep 1, 2015 - Jun 13, 2016 ▾

Hour Day



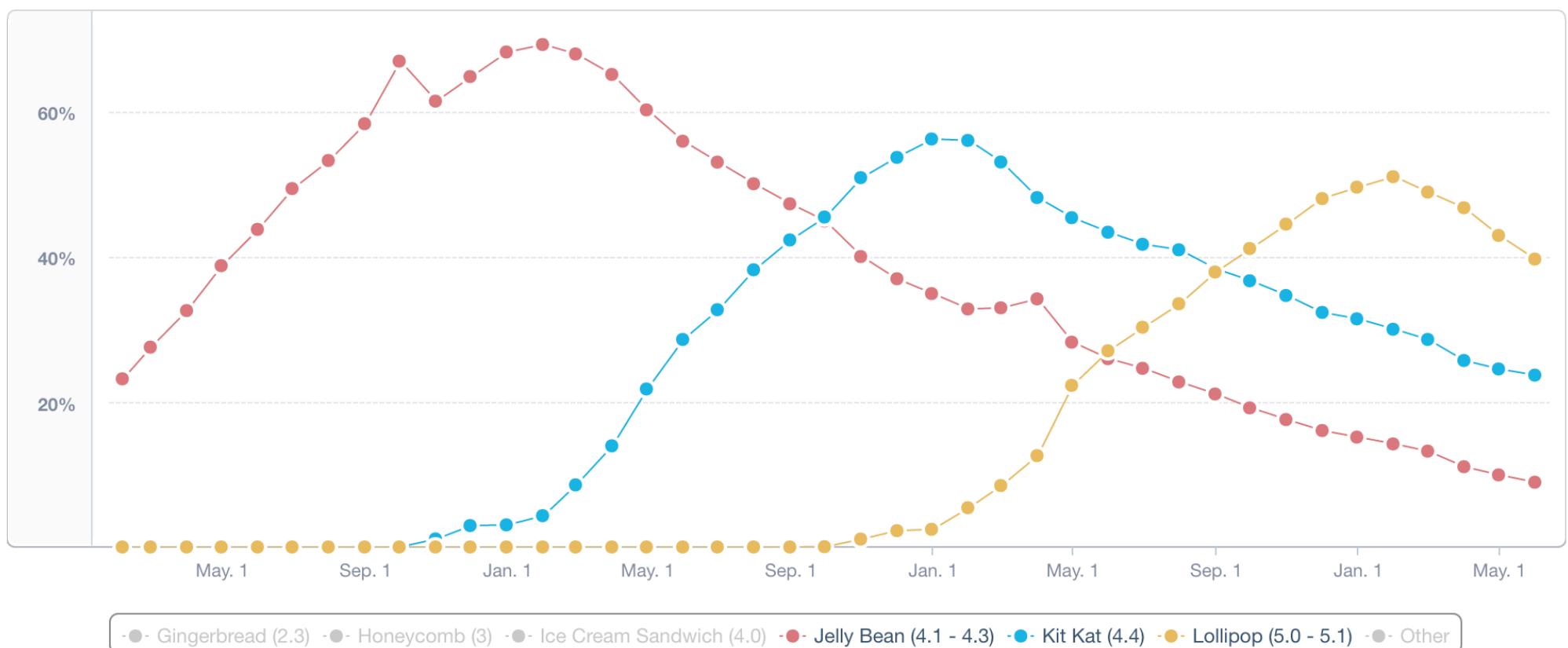
Android

Android OS Adoption

Feb 5th, 2013

Jun 13th, 2016

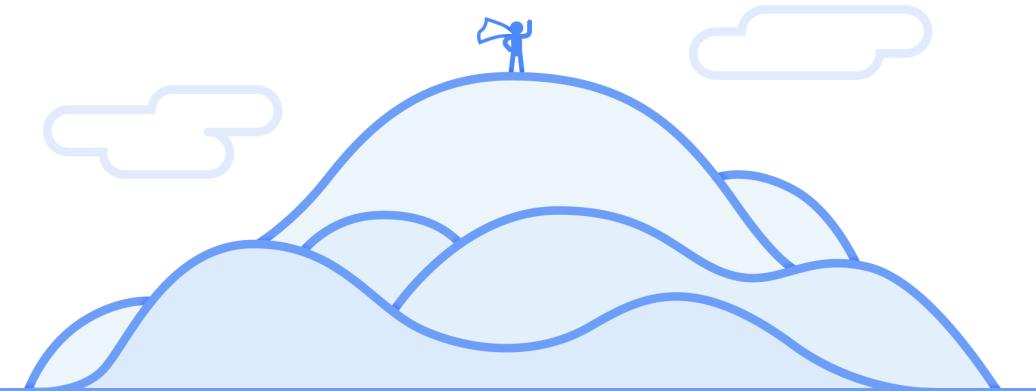
DONE



Pourquoi choisir Ionic ?



ionic



Ionic n'est pas seulement un framework, c'est un SDK.

- Documentation complète
- Une large communauté, des tutoriaux et beaucoup de ressources disponibles.

15,872 stars

2,367 forks

21.8K followers

-
- Après avoir levé \$1 million à ses débuts la société lève \$2.6 million en 2015 et \$8.5 million en 2016.
 - Certaines offres d'emplois parlent d'ionic.

-
- Basé sur AngularJs et Typescript

- Un écosystème complet, build, application de test, extensions etc.

Les outils Ionic

Ionic CLI



Ionic View



Ionic lab



Ionic Creator



Ionic Box



Ionicons



Ionic push



ngCordova



L'écosystème



Apache Cordova

Plateforme pour construire des applications natives en utilisant les technologies HTML,CSS et JavaScript.



Gulp

Outil d'automatisation « task runner » il permet d'écrire des tâches récurrentes de développement en JavaScript.



AngularJs 2 + Typescript

Framework JavaScript pour programmer des applications Web SPA (Single Page Application)



Sass

Outil permettant d'étendre les fonctionnalités CSS par la compilation.

Rappel

ECMAScript 6 (ES2015)

Classes

```
class Shape {  
    constructor (id, x, y) {  
        this.id = id  
        this.move(x, y)  
    }  
    move (x, y) {  
        this.x = x  
        this.y = y  
    }  
}
```

Modules

```
// lib/math.js  
export function sum (x, y) { return x + y }  
export var pi = 3.141593  
  
// someApp.js  
import * as math from "lib/math"  
console.log("2π = " + math.sum(math.pi, math.pi))  
  
// otherApp.js  
import { sum, pi } from "lib/math"  
console.log("2π = " + sum(pi, pi))
```

Promises

```
doSomething().then((response) => {  
    console.log(response);  
});
```

Arrow functions

```
someFunction((response) => {  
    console.log(response);  
});
```

rather than:

```
someFunction(function(response){  
    console.log(response);  
});
```

```
function add(x: number, y :number):number {  
    return x + y;  
}  
add('a', 'b'); // compiler error
```

Example

```
export class MyApp {  
  
    rootPage: any = HomePage;  
  
    constructor(platform: Platform) {  
  
        platform.ready().then(() => {  
  
            });  
    }  
}
```

Binding a Property to a Value

```
<input [value]="firstName">
```

Calling a Function on an Event

```
<button (click)="someFunction($event)">
```

Rendering Expressions

```
<p>Hi, {{name}}</p>
```

Two Way Data Binding

```
<input [(ngModel)]="name">
```

```
<input [value]="name" (input)="name = $event.target.value">
```

Angular 2

Creating a Local Variable

```
<p #myParagraph></p>
```

```
<button (click)="myParagraph.innerHTML = 'Once upon a time...'">
```

Embedded Templates

```
<p *something="someExpression"></p>
  =
<template [something]="someExpression">
  <p></p>
</template>
```

Directives

```
<section *ngIf="showSection"></section>
<li *ngFor="let item of items"></li>
```

Decorators

```
@Component({
  selector: 'my-component',
  services: [MyService]
})
```

« Information »

**Ionic 2 est
encore
en Beta.**

Mais assez stable
pour une utilisation
avancée.

Installation

Pré-requis

NODEJS + NPM



RUBY + SASS

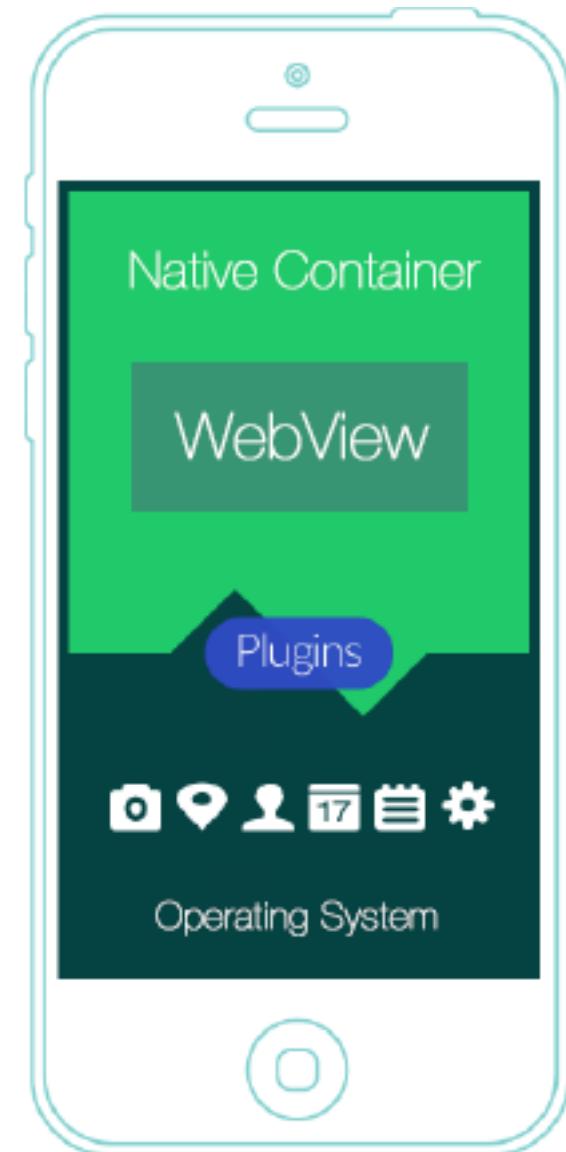


CORDOVA + IONIC



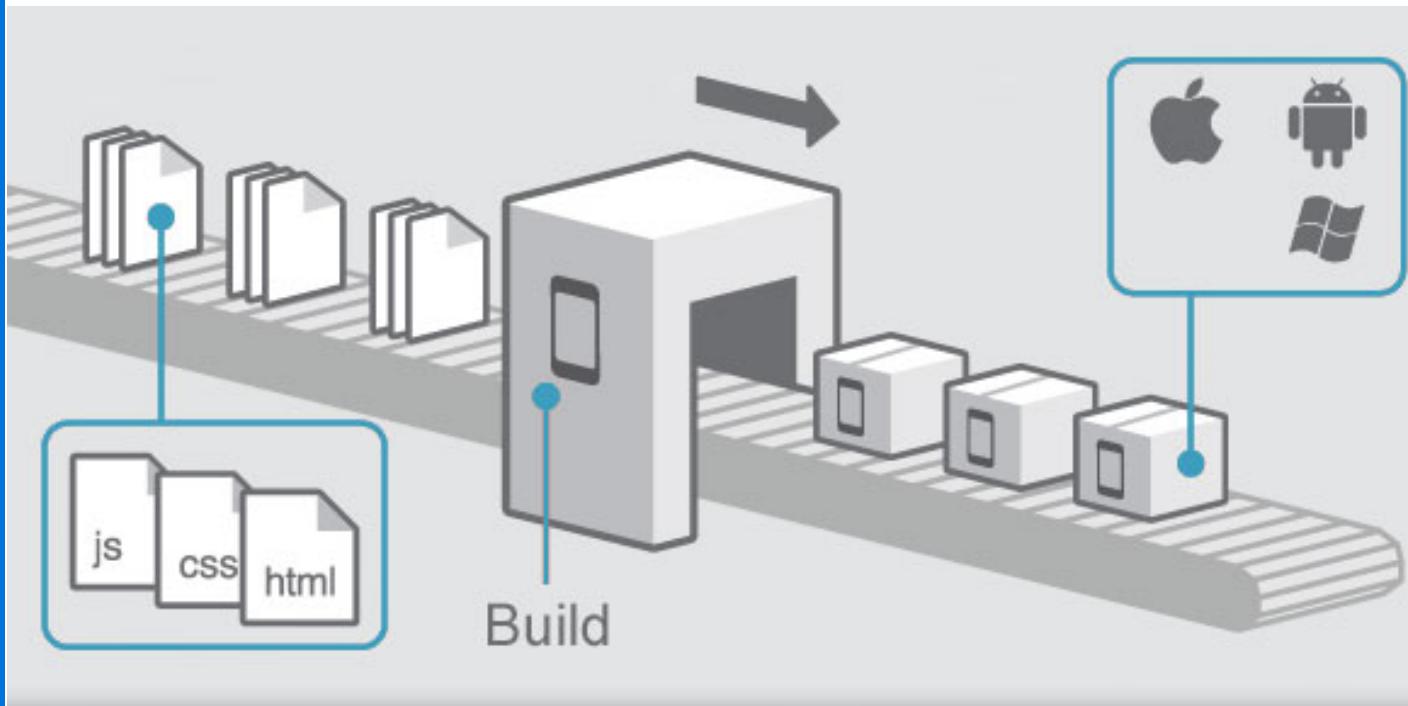
Comment fonctionne l'hybride ?

Une WebView* dans un conteneur natif



*WebView: composant natif pour afficher des pages web.

Outil de build multi-plateforme



APACHE
CORDOVA™

Cordova CLI

- **Installation**

```
$ sudo npm install -g cordova
```

- **Ajouter une nouvelle plateforme**

```
$ cordova platform add ios
```

```
$ cordova platform add android
```

```
$ cordova platform add wp8
```

- **Compiler**

```
$ cordova build android
```

```
$ cordova emulate android
```

```
$ cordova run android
```

Ionic CLI

- **Installation**

```
$ sudo npm install -g ionic@beta
```

- **Ajouter une nouvelle plateforme**

```
$ ionic platform add ios
```

```
$ ionic platform add android
```

```
$ ionic platform add wp8
```

- **Compiler**

```
$ ionic build android
```

```
$ ionic emulate android
```

```
$ ionic run android
```

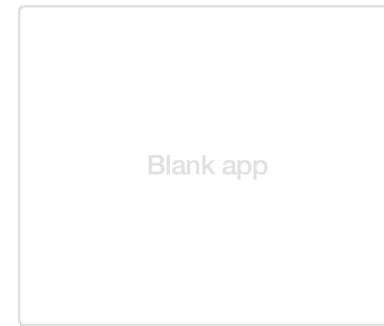
```
$ ionic help
```

Available tasks: (use --help or -h for more info)

<code>start</code>	Starts a new Ionic project in the specified PATH
<code>serve</code>	Start a local development server for app dev/testing
<code>platform</code>	Add platform target for building an Ionic app
<code>run</code>	Run an Ionic project on a connected device
<code>emulate</code>	Emulate an Ionic project on a simulator or emulator
<code>build</code>	Locally build an Ionic project for a given platform
<code>plugin</code>	Add a Cordova plugin
<code>resources</code>	Automatically create icon and splash screen resources (beta) Put your images in the ./resources directory, named splash or icon. Accepted file types are .png, .ai, and .psd. Icons should be 192x192 px without rounded corners. Splashscreens should be 2208x2208 px, with the image centered in the middle.
<code>package</code>	Package an app using the Ionic Build service (beta)
<code>upload</code>	Upload an app to your Ionic account
<code>share</code>	Share an app with a client, co-worker, friend, or customer
<code>lib</code>	Gets Ionic library version or updates the Ionic library
<code>setup</code>	Configure the project with a build tool (beta)
<code>browser</code>	Add another browser for a platform (beta)
<code>service</code>	Add an Ionic service package and install any required plugins
<code>add</code>	Add an Ion, bower component, or addon to the project
<code>remove</code>	Remove an Ion, bower component, or addon from the project
<code>list</code>	List Ions, bower components, or addons in the project
<code>ions</code>	List available ions to add to your project
<code>templates</code>	List available Ionic starter templates
<code>info</code>	List information about the users runtime environment
<code>help</code>	Provides help for a certain command
<code>link</code>	Sets your Ionic App ID for your project
<code>hooks</code>	Manage your Ionic Cordova hooks
<code>state</code>	Saves or restores state of your Ionic Application using the package.json file
<code>docs</code>	Opens up the documentation for Ionic

Let's go ionic !

Créer un projet



Blank app

\$ ionic start hello blank --v2

Your system information:

```
Cordova CLI: 6.2.0
Ionic Framework Version: 2.0.0-beta.10
Ionic CLI Version: 2.0.0-beta.25
Ionic App Lib Version: 2.0.0-beta.15
ios-deploy version: Not installed
ios-sim version: Not installed
OS: Mac OS X El Capitan
Node Version: v4.4.7
Xcode version: Xcode 7.3.1 Build version 7D1014
```

\$ ionic info

Lancer le projet dans le navigateur



\$ ionic serve

Structure d'un projet

```
hello
  app
  hooks
  node_modules
  platforms
  plugins
  resources
  www
    build
      index.html
  .gitignore
  config.xml
  gulpfile.js
  ionic.config.json
  package.json
```

```
www/index.html
1  <!DOCTYPE html>
2  <html dir="ltr" lang="en">
3  |
4  <head>
5    <title>Ionic</title>
6    <meta charset="UTF-8">
7    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no">
8    <meta name="format-detection" content="telephone=no">
9    <meta name="msapplication-tap-highlight" content="no">
10   <link ios-href="build/css/app.ios.css" rel="stylesheet">
11   <link md-href="build/css/app.md.css" rel="stylesheet">
12   <link wp-href="build/css/app.wp.css" rel="stylesheet">
13 </head>
14
15
16 <body>
17   <ion-app></ion-app>
18
19   
20   <script src="cordova.js"></script>
21   
22   <script src="build/js/es6-shim.min.js"></script>
23   
24   <script src="build/js/Reflect.js"></script>
25   <script src="build/js/zone.js"></script>
26   
27   <script src="build/js/app.bundle.js"></script>
28 </body>
29
```

Structure d'un projet

The image shows a file explorer interface with two panes. The left pane displays the project structure:

```
hello
  └── app
      ├── pages
      │   └── home
      │       ├── home.html
      │       ├── home.js
      │       └── home.scss
      └── theme
          ├── app.core.scss
          ├── app.ios.scss
          ├── app.md.scss
          ├── app.variables.scss
          ├── app.wp.scss
          └── app.js
```

The right pane shows the content of the `index.html` file:

```
<!DOCTYPE html>
<html dir="ltr" lang="en">
|
<head>
    <title>Ionic</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no">
    <meta name="format-detection" content="telephone=no">
    <meta name="msapplication-tap-highlight" content="no">
    <link ios-href="build/css/app.ios.css" rel="stylesheet">
    <link md-href="build/css/app.md.css" rel="stylesheet">
    <link wp-href="build/css/app.wp.css" rel="stylesheet">
</head>

<body>
    <ion-app></ion-app>

    
    <script src="cordova.js"></script>
    
    <script src="build/js/es6-shim.min.js"></script>
    
    <script src="build/js/Reflect.js"></script>
    <script src="build/js/zone.js"></script>
    
    <script src="build/js/app.bundle.js"></script>
</body>
```

Structure d'un projet

The image shows a file explorer on the left and a code editor on the right. The file explorer displays the project structure:

- hello (selected)
- app
 - pages
 - home
 - home.html
 - home.scss
 - home.ts
 - theme
 - app.core.scss
 - app.ios.scss
 - app.md.scss
 - app.variables.scss
 - app.wp.scss
 - hooks
 - node_modules
 - platforms
 - plugins
 - resources
 - typings
 - www
 - build
 - index.html
 - .gitignore
 - config.xml
 - gulpfile.js
 - ionic.config.json
 - package.json

app/app.ts

```
1 import {Component} from '@angular/core';
2 import {Platform, ionicBootstrap} from 'ionic-an
3 import {StatusBar} from 'ionic-native';
4 import {HomePage} from './pages/home/home';
5
6
7 @Component({
8   template: '<ion-nav [root]="rootPage"></ion-na
9 })
10 export class MyApp {
11   rootPage: any = HomePage;
12
13   constructor(platform: Platform) {
14     platform.ready().then(() => {
15       // Okay, so the platform is ready and our
16       // Here you can do any higher level native
17       StatusBar.styleDefault();
18     });
19   }
20 }
21
22 ionicBootstrap(MyApp);
23
```

Ionic generators

\$ ionic generate --list

Angular:

- * component
- * directive
- * provider
- * pipe

Ionic:

- * page
- * tabs

\$ ionic g component MyComponent --ts

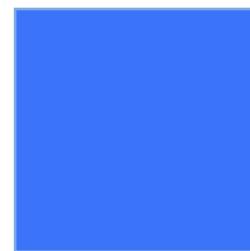
The screenshot shows a terminal window with two panes. The left pane displays a file tree for an Ionic project named 'hello'. The right pane shows the command being run: '\$ ionic g component MyComponent --ts'. Below this, the generated code for 'my-component.ts' is displayed.

```
...home/home.html .../my-component/my-component.ts .../home/home.ts app/app.ts
1 import { Component } from '@angular/core';
2
3 /*
4   Generated class for the MyComponent component.
5
6   See https://angular.io/docs/ts/latest/api/core/ComponentMetadata-class.html
7   for more info on Angular 2 Components.
8 */
9 @Component({
10   selector: 'my-component',
11   templateUrl: 'build/components/my-component/my-component.html'
12 })
13 export class MyComponent {
14
15   text: string;
16
17   constructor() {
18     this.text = 'Hello World';
19   }
20 }
```

Les composants Ionic

Ionic UI scaffolding

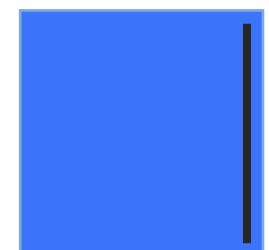
<ion-app>



<ion-nav>



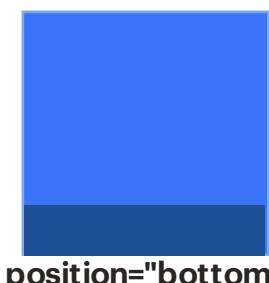
<ion-content>



<ion-toolbar>



<ion-toolbar>



<ion-nav-bar>



<ion-buttons>





UI Components

A comprehensive preview of our mobile UI components - everything you'll need.

Icon Buttons

[Demo Source](#)

To add icons to a button, add an icon component inside of it:

```
<!-- Float the icon left -->
<button>
  <ion-icon name="home"></ion-icon>
  Left Icon
</button>

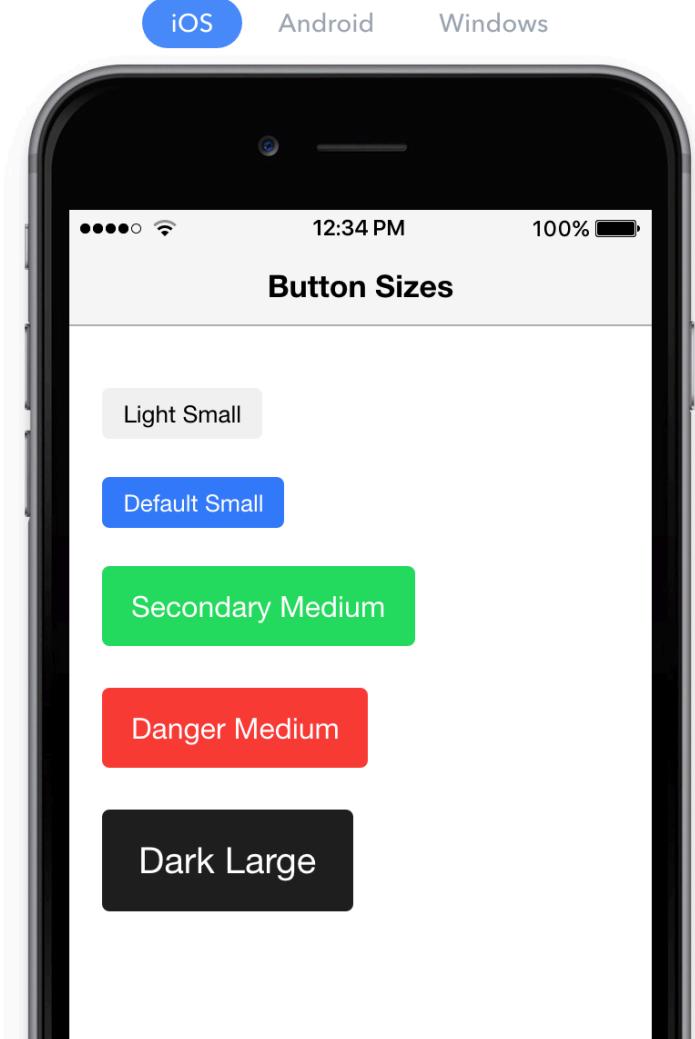
<!-- Float the icon right -->
<button>
  Right Icon
  <ion-icon name="home"></ion-icon>
</button>

<!-- Only icon (no text) -->
<button>
  <ion-icon name="home"></ion-icon>
</button>
```

iOS

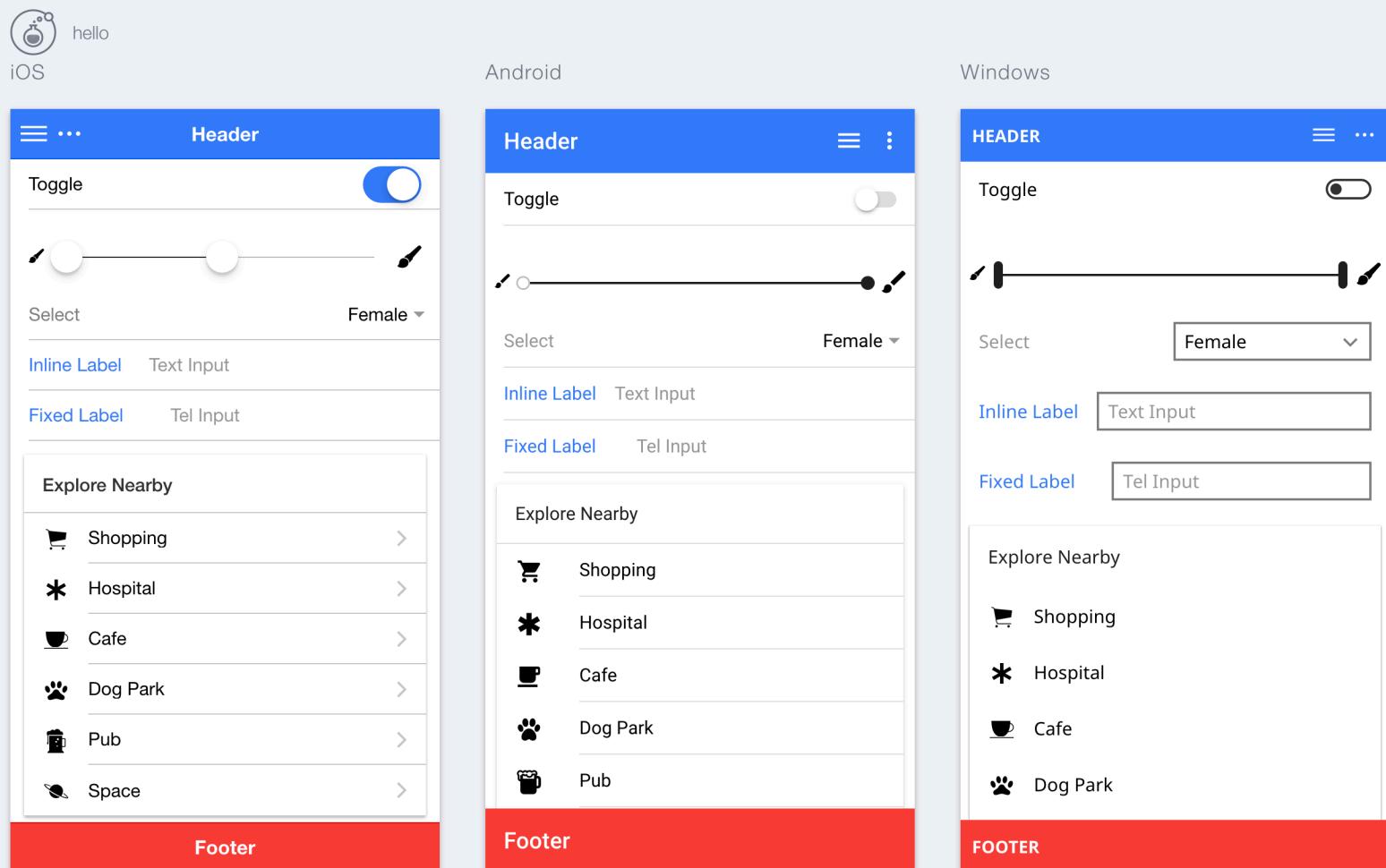
Android

Windows



Platform continuity

\$ ionic serve --lab



A vous de jouer !

Platforms config

The image shows a code editor interface with two main panes. On the left is a file tree for an Ionic project named 'hello'. The tree includes 'app', 'pages' (with 'home' and 'theme' subfolders), 'hooks', 'node_modules', 'platforms', 'plugins', 'resources', 'typings', and 'www' (containing 'build', 'index.html', '.gitignore', 'config.xml', 'gulpfile.js', 'ionic.config.json', and 'package.json'). On the right is the content of the file 'app/app.ts'. The code defines a class 'MyApp' with a constructor that sets the root page to 'HomePage'. It then calls 'ionicBootstrap' with 'MyApp' as the entry component, 'mode: 'md'', and an object for the 'platforms' configuration, which includes 'ios' settings for 'tabbarPlacement' and 'iconMode'.

```
app/app.ts
4 import { HomePage } from './pages/home/home';
5
6
7 @Component({
8   template: '<ion-nav [root]="rootPage"></ion-nav>'
9 })
10 export class MyApp {
11   rootPage: any = HomePage;
12
13   constructor(platform: Platform) {
14     platform.ready().then(() => {
15       // Okay, so the platform is ready and our
16       // // Here you can do any higher level native
17       StatusBar.styleDefault();
18     });
19   }
20 }
21
22 ionicBootstrap(MyApp, null, {
23   mode: 'md',
24   platforms: {
25     ios: {
26       tabbarPlacement: 'bottom',
27       iconMode: 'md',
28     }
29   }
30 });
```

Ionic Navigation

Ionic 2 navigation != Angular 2 Router

driftyco / ionic

Watch 1,516 Star 24,544 Fork 4,974

Code Issues 428 Pull requests 17 Pulse Graphs

Update to new Angular 2 Router #6563

Open dheran opened this issue on 18 May · 5 comments

dheran commented on 18 May

Ionic 2

Update ionic to new Angular 2 Router

5

Ionitron added the **v2** label on 18 May

adambadley commented on 18 May

Ionic member +

This will still be a work in progress as the new new Angular 2 router still has some holes that does not allow us to update yet. The next release will not be integrated with the ng2 router yet, and we're working with the Angular team to solve this.

adambadley self-assigned this on 18 May

Labels **v2**

Milestone No milestone

Assignees adambadley

6 participants

Notifications

Unsubscribe

You're receiving notifications because you commented.

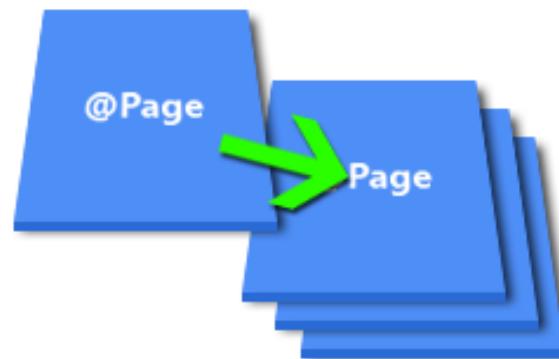
Pour le moment...

Ionic Navigation Stack

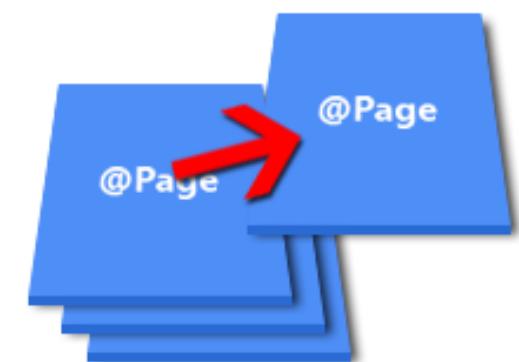
Navigation Stack



Push



Pop



Aucune URL de page. Pas obligatoirement nécessaire sur mobile.

rootPage = CatsPage

```
@Component({
  template: '<ion-nav [root]="rootPage"></ion-nav>'
})
export class MyApp {

  private rootPage:any = CatsPage;

  constructor(private platform:Platform) {
```



this.nav.push(DogsPage);

```
import {DogsPage} from '.../pages/dogs/dogs';

@Component({
  templateUrl: 'build/pages/cats/cats.html',
})
export class CatsPage {
  constructor(public nav: NavController) {

  }
  next(): void {
    this.nav.push(DogsPage);
  }
}
```



this.nav.push(CowsPage);

```
import {CowsPage} from '.../.../pages/cows/cows';

@Component({
  templateUrl: 'build/pages/dogs/dogs.html',
})
export class DogsPage {
  constructor(public nav: NavController) {

  }
  next(): void {
    this.nav.push(CowsPage);
  }
}
```



this.nav.pop();

```
@Component({
  templateUrl: 'build/pages/cows/cows.html',
})
export class CowsPage {
  constructor(public nav: NavController) {

  }
  next(): void {
    this.nav.pop();
  }
}
```



This.nav.setRoot(CatsPage)

```
import {CatsPage} from '.../.../pages/cats/cats';

@Component({
  templateUrl: 'build/pages/dogs/dogs.html',
})
export class DogsPage {
  constructor(public nav: NavController) {

  }
  next(): void {
    this.nav.setRoot(CatsPage);
  }
}
```



Tabs

\$ ionic start test-tabs tabs --lab

iOS

Home

Welcome to Ionic!

This starter project comes with simple tabs-based layout for apps that are going to primarily use a Tabbed UI.

Take a look at the app/ directory to add or change tabs, update any existing page or create new pages.

Android

Home



HOME



ABOUT



CONTACT

Welcome to Ionic!

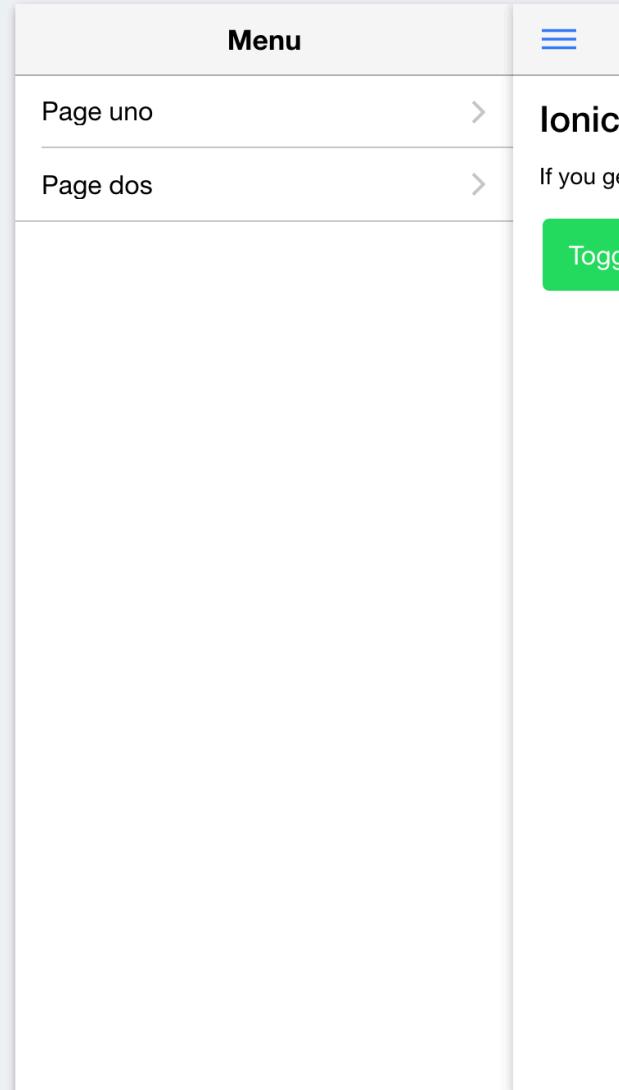
This starter project comes with simple tabs-based layout for apps that are going to primarily use a Tabbed UI.

Take a look at the app/ directory to add or change tabs, update any existing page or create new pages.

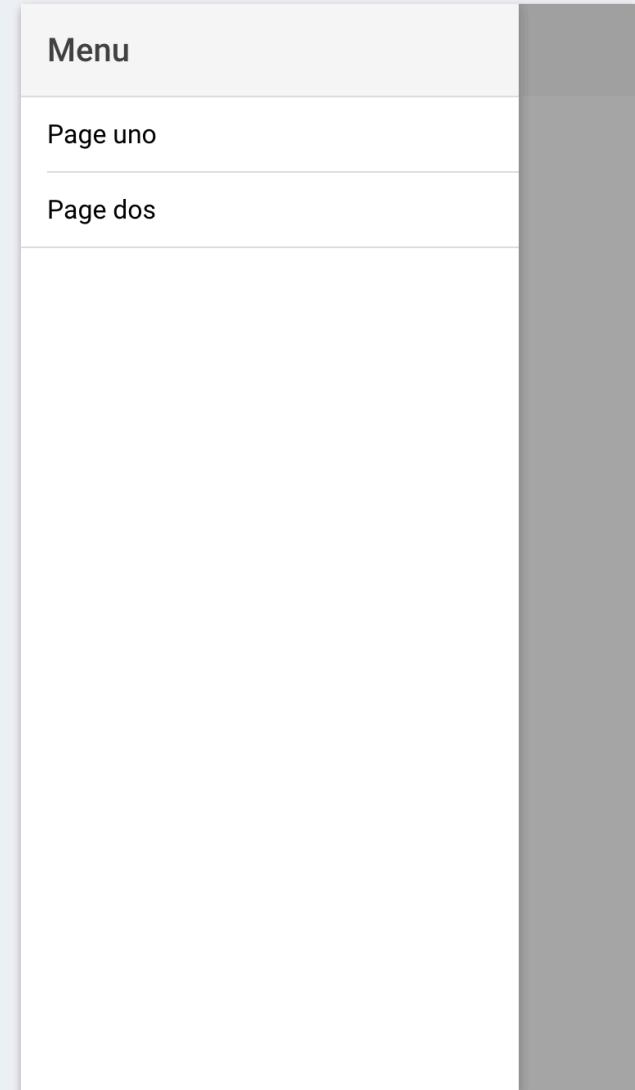
Sidemenu

\$ ionic start test-sidemenu sidemenu --lab

iOS

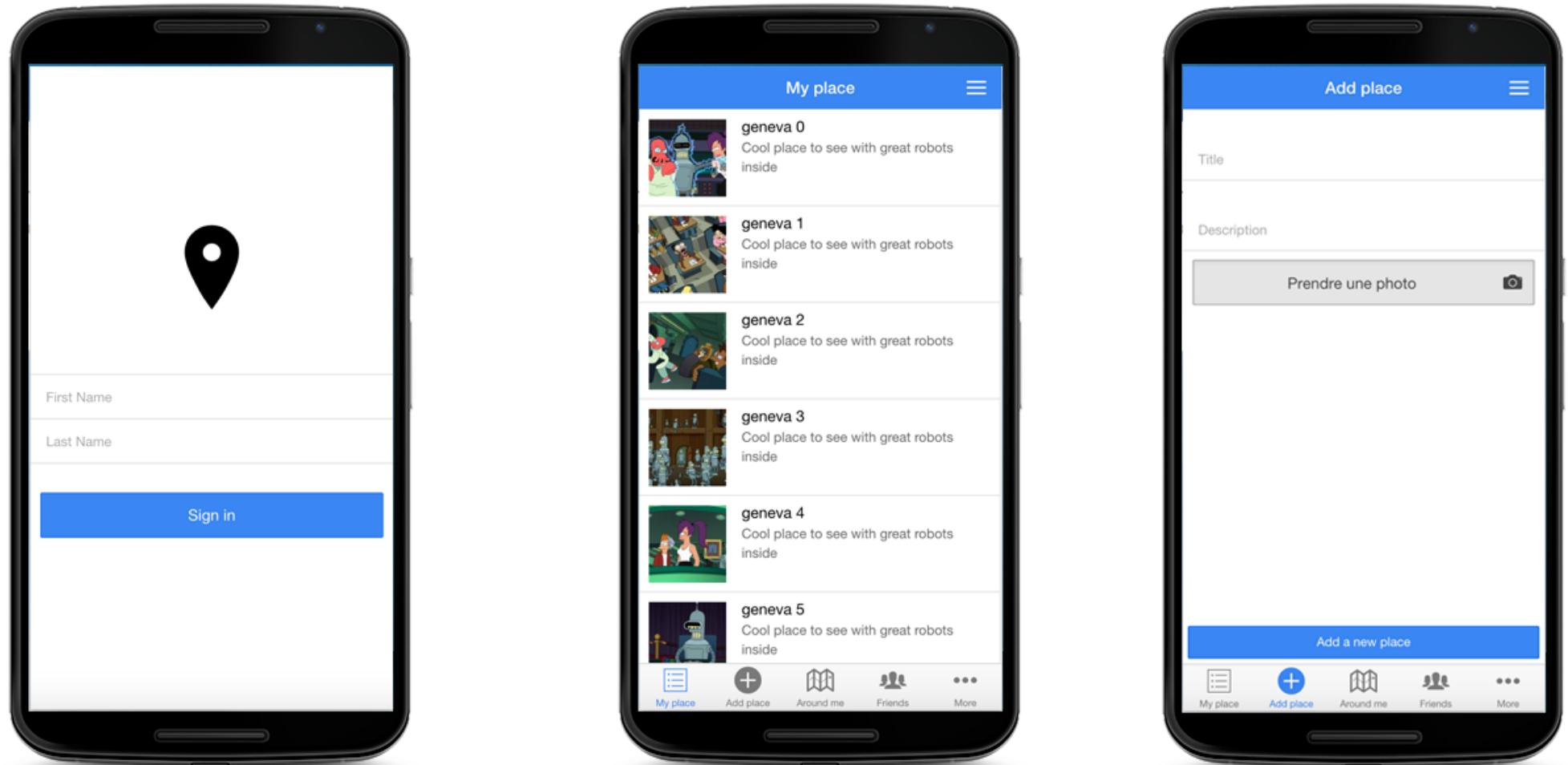


Android



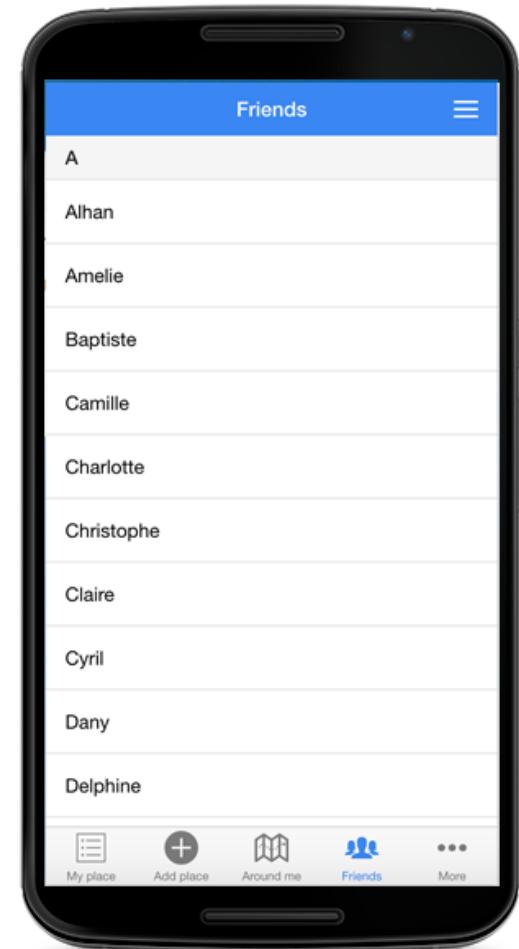
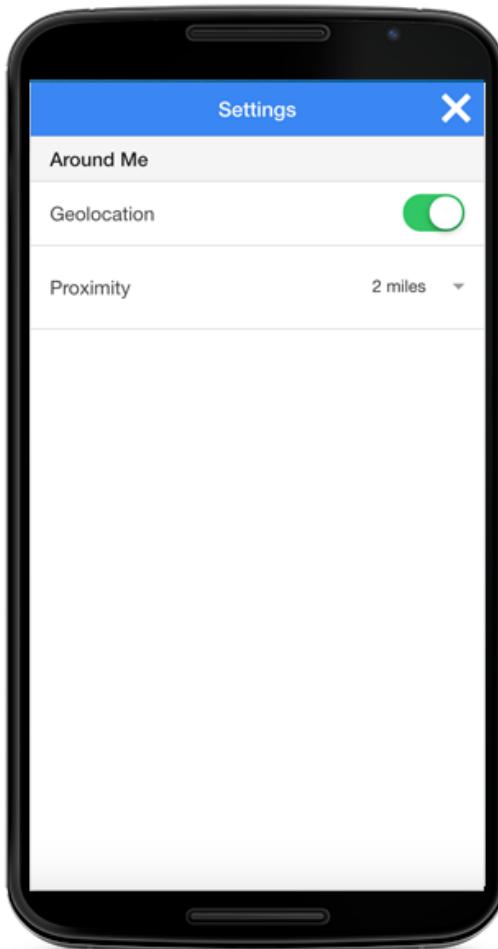
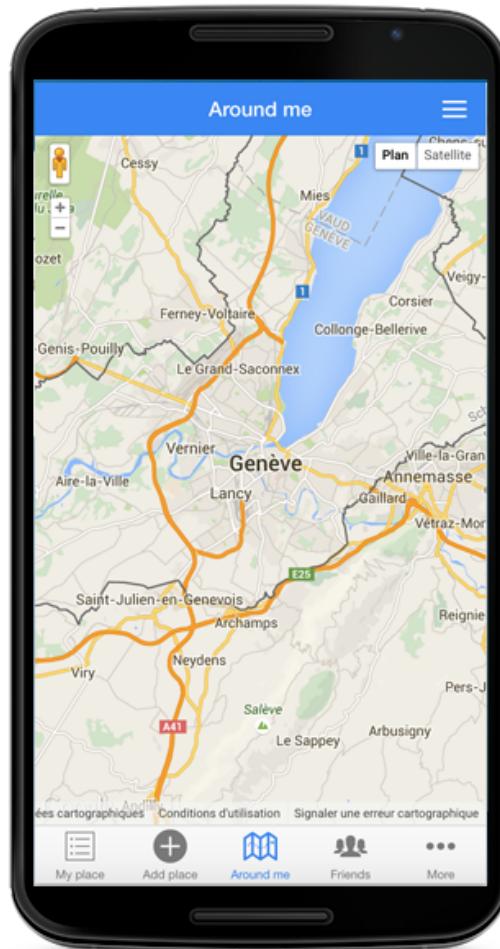
Démarrons un nouveau projet

Objectif:



Prototype « Ionic Places »

Objectif:



Prototype « Ionic Places »

Installation

Récupérer le projet

```
$ git clone https://github.com/kimak/ionic2-places
```

Un compte github ? « Fork me. »



```
$ ionic serve
```

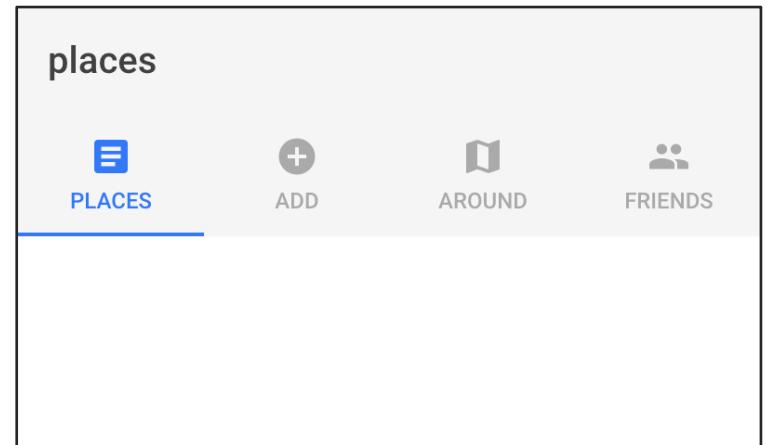
Ionic Blank

The world is your oyster.

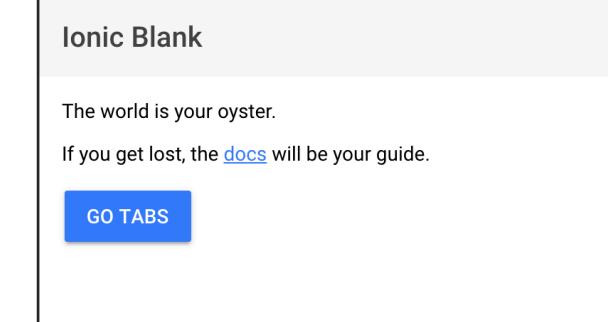
If you get lost, the [docs](#) will be your guide.

Pratique

1) Mettre en place un « tabs »



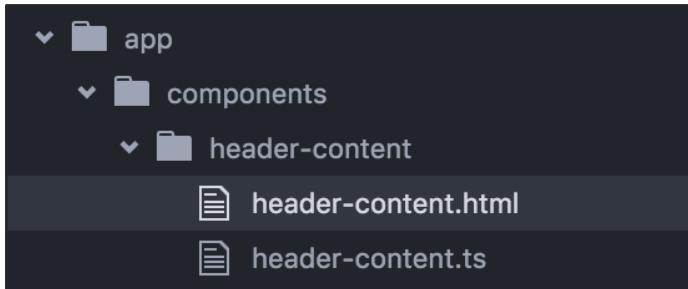
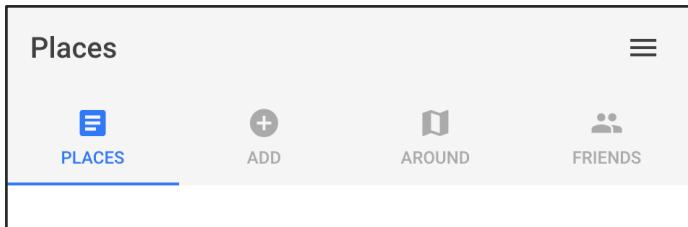
+ Mettre en place un bouton vers les tabs



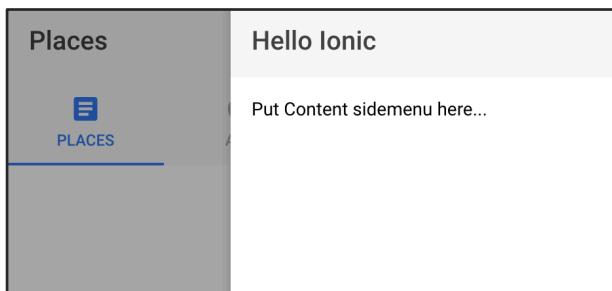
```
goTabs(){
  this.nav.push(TabsPage)
}
```

Pratique

2) Mettre en place un composant header



+ Mettre en place un sidemenu



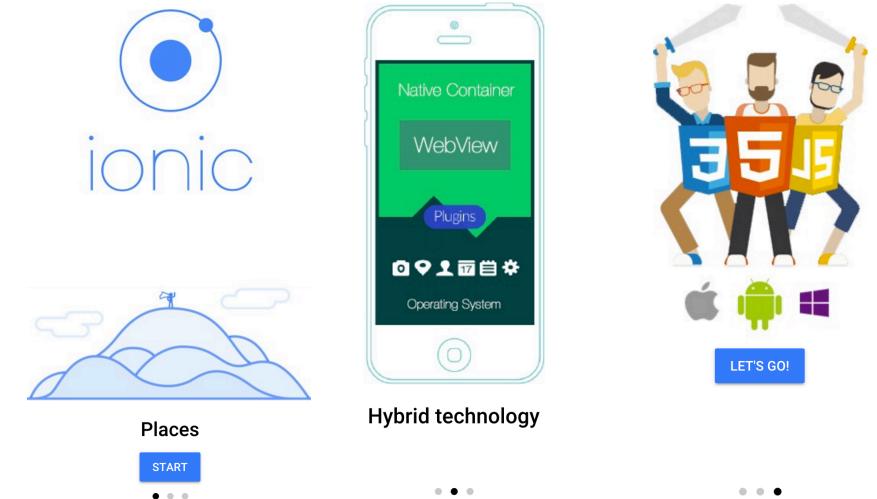
```
<ion-menu [content]="content" side="right">

  <ion-toolbar>
    <ion-title>Hello Ionic</ion-title>
  </ion-toolbar>

  <ion-content padding>
    Put Content sidemenu here...
  </ion-content>
</ion-menu>
```

Pratique

5) Mettre en place un slider sur la home »



4) Mettre en place une page login

A screenshot of a mobile application's login screen. At the top left is a back arrow icon and the word "login". Below this is a "Username" input field. Underneath is a "Password" input field. At the bottom is a large blue "SIGN IN" button.

Testons sur nos téléphones



Ionic View App

The screenshot shows the Ionic View app interface on a mobile device. At the top, it displays "Carrier" and "3:14 PM". Below this is a header section with a camera icon, the text "MY APPS", and a gear icon. The main content area lists four applications:

- Facebook**
ID: A5D64ED3
SIZE: 2.9 MB
MODIFIED: FEB 18, 2014 3:45:25 PM
- Snapcat**
ID: AFF7CB5B
SIZE: 112.3 KB
MODIFIED: FEB 17, 2014 2:12:32 PM
- Grumblr**
ID: D37DD3A5
SIZE: 8.4 MB
MODIFIED: FEB 11, 2014 11:07:06 AM
- Slappy Bird**
ID: F3DF48C8
SIZE: 2.7 MB
MODIFIED: FEB 18, 2014 10:38:01 AM