



Bluemix en Pratique

Exercice 3 - Création d'une application mobile

Version:

1

Modification:

28-juin-15

Propriétaire:

Philippe THOMAS for Devoxx

Table des matières

Exercice 3.a – Déploiement d'une application Mobile Cloud services	4
Exercice 3.b – Télécharger et modifier le code	7
Exercice 3.c – Test sur votre ordinateur	8
Exercice 3.d – Déploiement vers Bluemix.....	9
Exercice 3.d – Explorer et modifier votre App.....	12

IBM Bluemix propose des services variés pour implémenter une application mobile. Il faut tenir compte de tous les aspects d'une application mobile : l'accès, la sécurité, le stockage, la qualité de service mais aussi le type de tablette ou de smartphone et son système d'exploitation (IOS, Android ...). Dans cet exercice, nous allons nous intéresser à une application de type RWD (**Responsive Web Design**) mais il existe d'autres exemples de création d'applications natives ou hybrides avec Bluemix.

Exercice 3.a – Déploiement d'une application Mobile Cloud services

Cet exercice vous montrera comment déployer une application mobile avec un backend constitué de services mobile Bluemix.

Dans cet exercice, nous aurons besoin de 3 clés fournies par Bluemix (Route key, App key et App key). Elles devront être insérées dans l'application pour permettre son fonctionnement.

Pour exécuter cet exercice, vous aurez besoin du CLI de Cloud Foundry et de Bower (<http://bower.io/>)

Dans Bluemix, dans le catalogue, choisir le **Mobile Cloud Boilerplate**.



Figure 1 Boilerplate : IBM Mobile Cloud

Dans ce boilerplate, nous aurons une application et tous les principaux services pour mettre en œuvre une application mobile. Choisir un nom d'application unique comme **MobileXX** avec XX remplacés par vos initiales.

Exercice 3 - Création d'une application mobile

Figure 2 Création de l'application

Cliquer sur **CREATE**, cela va créer et lancer votre nouvelle application.

Figure 3 Les clés de sécurité

Sauvegarder / copier les différentes clés pour plus tard.
Cliquer **CONTINUE**.

Exercice 3 - Création d'une application mobile

The screenshot shows the IBM MobileCloud Platform interface. At the top, it displays the application name "MobileXX" and its route "Routes: MobileXX.stage1.mybluemix.net". On the right, there's a "MOBILE OPTIONS" section with a gear icon. Below this, three main steps are outlined:

1. Enable your Node.js app for mobile. A button labeled "Enable Node.js App" is present.
2. Install and configure the SDK for your client. Icons for iOS, Android, Hybrid, and Java Script are shown.
3. Enter registration credentials. It lists the Route, App Key, and App Secret with their respective values.

In the center, there's a "SDK FOR NODE.JS™" section with fields for Instances (set to 1), Memory Quota (128 MB), and Available Memory (1.375 GB). Buttons for "SAVE" and "RESET" are available. Below this are two large buttons: "ADD A SERVICE OR API" and "BIND A SERVICE OR API".

At the bottom, there are links for "Mobile Application Security" and "Mobile Quality Assurance". On the right side, there's an "APP HEALTH" section showing "Your app is running." with "RESTART" and "STOP" buttons, and an "ACTIVITY LOG" section listing recent events:

- 4/3/15 2:42 PM thomas1@fr.ibm.com started MobileXX app
- 4/3/15 2:40 PM thomas1@fr.ibm.com updated MobileXX app
 - changed routes
- 4/3/15 2:40 PM thomas1@fr.ibm.com created MobileXX app

Figure 4 Accès au lien

Ce boilerplate vient avec une application simple et rudimentaire que l'on peut accéder en cliquant sur le lien en haut de l'écran ou en revenant sur Overview.

hello from the cloud code application - please read the following pages for more information: [Extending Node.js to use Mobile Cloud applications](#)

Figure 5 Notre application et son message « hello for the cloud ».

Nous avons donc une application simple qui tourne dans Bluemix. Nous allons donc étendre cette application pour en faire quelque chose de plus réaliste...

Exercice 3.b – Télécharger et modifier le code

Nos services mobile sont en place. Nous allons utiliser une véritable application (en remplacement de celle proposée par le boilerplate). Cette application utilisera tous les services mobile que nous avons défini.

Ouvrir une fenêtre de commande en ligne ou un terminal et saisir :

```
cd unRépertoire  
git clone https://hub.jazz.net/git/mobilecloud/bluelist-mobiledata  
cd bluelist-mobiledata-mobileweb
```

Vous pouvez aussi bien charger ce code depuis ce lien:

<https://www.ng.bluemix.net/docs/#services/mobiledata/index.html#gettingstarted>

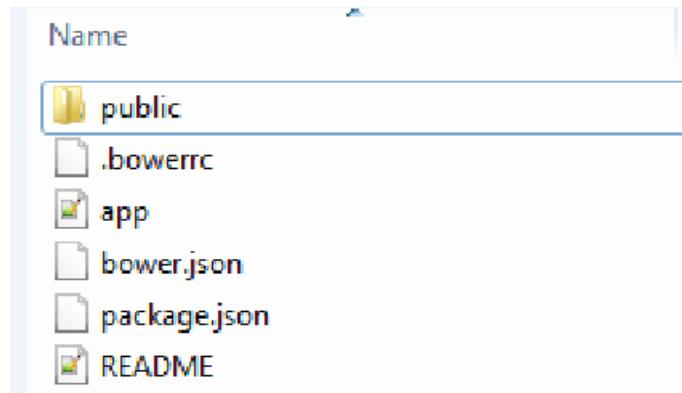


Figure 6 Les composants de l'App

Pour mettre cette application en œuvre, les instructions se trouvent dans le fichier README file. Et nous allons les couvrir ici également.

Dans le fichier **public/Bluelist.json**, nous plaçons nos clés dans les zones correspondantes.

```
Public\Bluelist.json  
{  
  "applicationId": "<INSERT_APPLICATION_ID_HERE>",  
  "applicationSecret": "<INSERT_APPLICATION_SECRET_HERE>",  
  "applicationRoute": "<INSERT_APPLICATION_ROUTE_HERE>"  
}
```

Ces trois clés nous permettent de communiquer avec le backend de Bluemix. Cela correspond à un handshake entre votre mobile et le backend de Bluemix.

Les clés sont saisies avec des double quotes : "applicationId": "5d615294-1c38-4ef6-b019-c1936bdf38f0",

Exercice 3.c – Test sur votre ordinateur

Avant de déployer sur le cloud, nous allons faire tourner notre nouvelle application en local sur votre poste de travail.

Vous devez avoir installé Node.js sur votre machine via :

<http://nodejs.org/>

Nous devons installer les modules associés à notre application :

npm install --production

Dans le même répertoire, installer le gestionnaire bower

npm install -g bower

Quand Bower est installé, installer les dépendances pour le coté client :

bower install

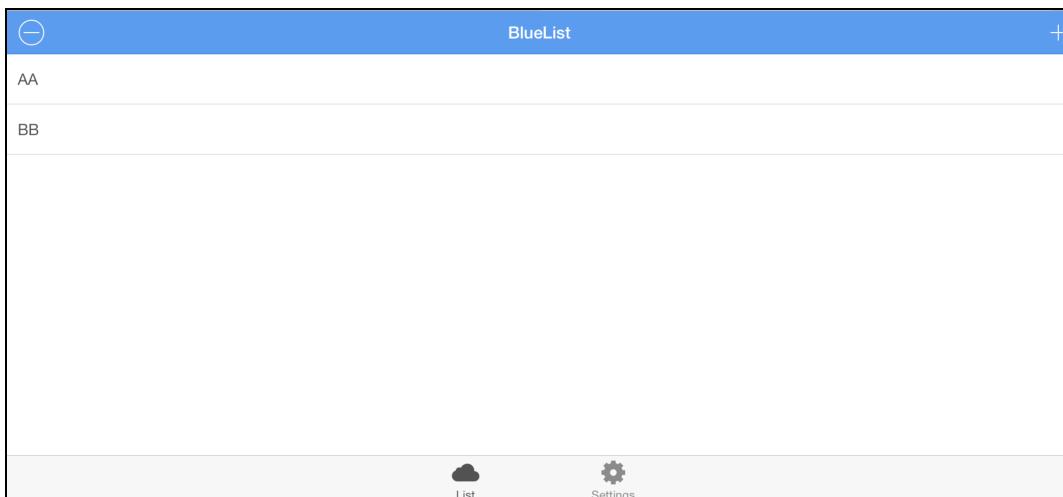
Finalement, vous pouvez démarrer votre application :

node app.js

Pour tester l'application, utiliser l'URL suivante depuis un navigateur

<http://localhost:3000/>

Vous devriez voir l'interface utilisateur de votre application.



Exercice 3.d – Déploiement vers Bluemix

Nous allons maintenant pousser notre application depuis notre poste de travail vers Bluemix. Positionnez vous dans le répertoire de votre application.

`cf push MobileXX` avec XX remplacés par vos initiales.

Cela peut prendre 1 à 2 minutes.

```
App started .....
```

state	since	cpu	memory	disk
#0	running	2014-12-20 09:16:20 AM	0.0%	29.9M of 128M 36.1M of 1G

```
C:\IBM\2015 IBM\2015 code\bluelist-mobiledata\bluelist-mobiledata-mobileweb>
```

Cette application est maintenant accessible par la route de Bluemix que nous avions utilisé précédemment (en début d'exercice).

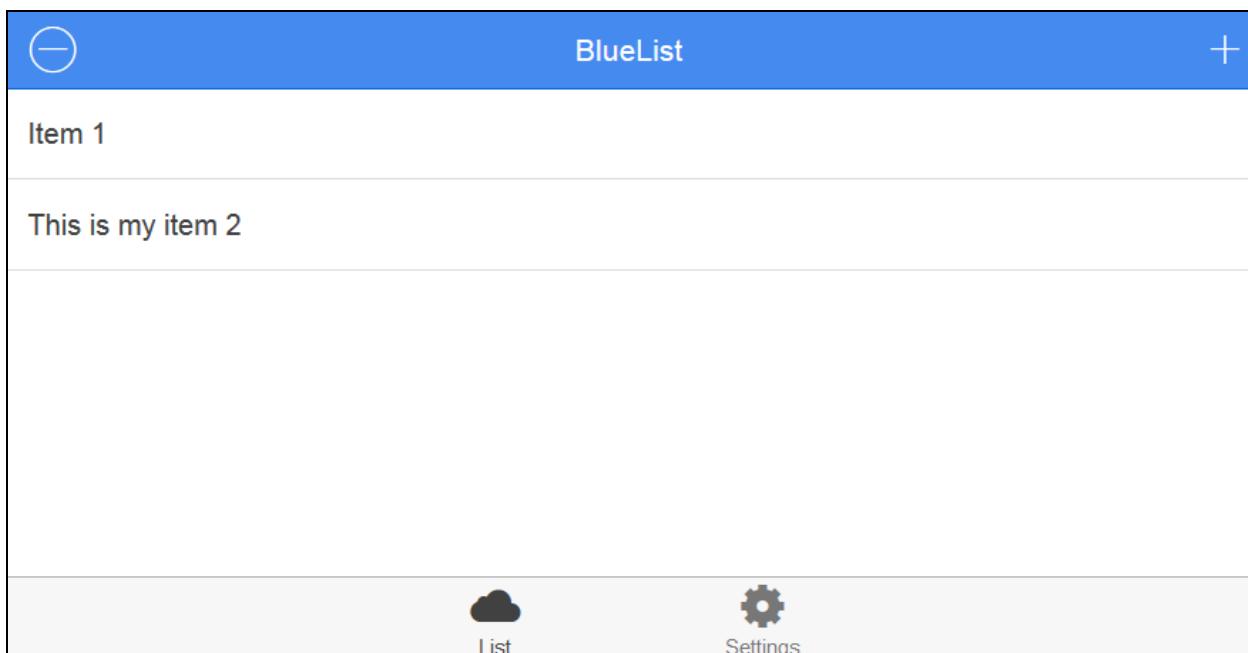


Figure 7 Test de notre App

Nous pouvons ajouter et supprimer dans la Bluelist.

Cette application peut également fonctionner sur votre Smartphone (et il suffit d'utiliser l'URL sur votre Smartphone).

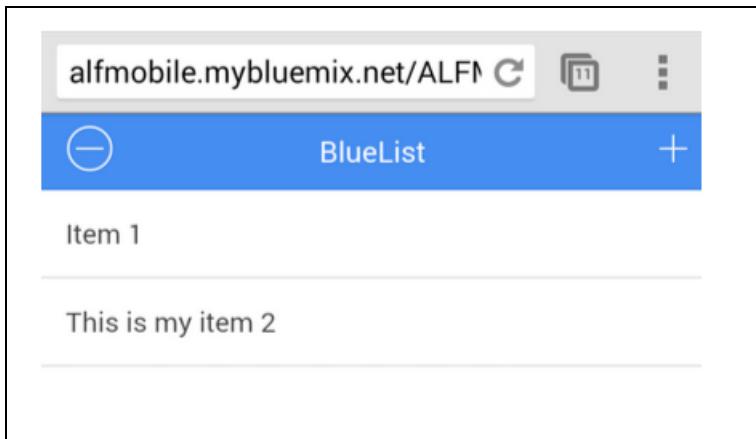


Figure 8 L'App sur notre Smartphone

Regardons de plus près les données saisies dans notre base données mobile.
Dans Bluemix, cliquer sur le Mobile Data Service

The screenshot shows the Bluemix mobile data service interface. At the top, it displays the app name 'LennartMBaaS1' and its URL 'Routes: LennartMBaaS1.mybluemix.net'. Below this, there are sections for 'INSTANCES', 'MEMORY QUOTA', and 'AVAILABLE MEMORY'. On the right, there's an 'APP HEALTH' section showing the app is running, and an 'ACTIVITY LOG' section with recent log entries. The main area contains four service cards: 'SDK FOR NODE.JS™', 'ADD A SERVICE OR API', 'BIND A SERVICE OR API', and 'Mobile Data'. The 'Mobile Data' card is circled in red and shows details for 'LennartMBaaS1-MobileData'.

Figure 9 Our app console

Puis dans la console Mobile Data Service, choisir Manage Data.

The screenshot shows the 'Import Data' section of the Mobile Data Service console. At the top right, it displays 'Current Storage 5.61KB'. Below this is a large blue circular icon with a dashed border containing a white document icon. To the left of the storage info, there's a note: 'Drag a file on to the dial or click to import. You can import files such as image, video or JSON database files and then use the files in the services in your application.' Below this note is a link: 'For more information about the file formats you can import, see [importing data into the mobile data service](#)'.

Below the storage info, there are two sections: 'Data Classes' (containing 2 items) and 'Files' (containing 0 items). Each section has a refresh icon and a delete icon.

Figure 10 La console du Mobile Data Service

Cloudant est une base de données NoSQL qui stocke des données au format JSON. Regardons sa structure.

The screenshot shows the Cloudant interface with two data tables:

- IBMDevice** (2 Rows):

platform	platformVersion	model	name	hardwareId	latestLogin	uid	vers
WEB	OS: Win32	Mozilla/5.0 (Windows; U; Windows NT 5.1; fr; rv:1.8.1.12) Gecko/20080702 Firefox/3.0.12 (.NET CLR 2.0.50727.44)		edb90d3a-b2ad-e3a401e46b26	type iso Date 2014-12-17T14:58:48.627	8d640baf355dfc5c61-a9	
WEB	OS: Win32	Mozilla/5.0 (Windows; U; Windows NT 5.1; fr; rv:1.8.1.12) Gecko/20080702 Firefox/3.0.12 (.NET CLR 2.0.50727.44)		edb90d3a-b2ad-e3a401e46b26	type iso Date 2014-12-17T15:10:13.376	f4c5477214a3adbea81-c2	
- Item** (2 Rows):

name	version	createdAt	modifiedAt	objectId
This is my item 2	1-18f456118178bce39b9ee3d166daea80	12/17/14, 7:10 AM	12/17/14, 7:10 AM	00efb288a96624422b8e8b0870871cbcb019883e
Item 1	1-b6eca994ed9fa98553988263c17b8cd	12/17/14, 7:10 AM	12/17/14, 7:10 AM	b7e0b23e09cbd8910bf37a74408f233abea771b7

Figure 11 Les données dans Cloudant

Exercice 3.d – Vous pouvez explorer et modifier votre App

- Dans le répertoire public et dans index.html, pour modifier le code HTML.
- Les interactions avec l'utilisateur sont dans le code public\js\app.js
- Voir également dans public/js
- Vous pouvez utiliser ce code comme un exemple pour d'autres développements.

Nous nous sommes intéressés à l'application Mobile en responsive design (RWD) dans cet exercice. Dans le répertoire de notre application Mobile, nous avons d'autres exemples de code natif avec IBM MobileFirst pour développer la même application BlueList sur IOS (ObjectiveC ou Swift), Android, ou avec l'aide Cordova.

Name	Date Modified	Size	Kind
▼ bluelist-mobiledata	3 Apr 2015 16:00	--	Folder
► bluelist-mobiledata-android	3 Apr 2015 13:46	--	Folder
► bluelist-mobiledata-cordova	3 Apr 2015 13:46	--	Folder
► bluelist-mobiledata-iOS	3 Apr 2015 16:12	--	Folder
► bluelist-mobiledata-mobileweb	3 Apr 2015 16:32	--	Folder
► bluelist-mobiledata-swift	3 Apr 2015 13:46	--	Folder
License.txt	3 Apr 2015 13:46	9 KB	text
notices.txt	3 Apr 2015 13:46	55 KB	text
project.json	3 Apr 2015 13:46	54 bytes	JSON
README.md	3 Apr 2015 13:46	2 KB	Markdown

Fin du lab.