

Business Directory Ionic 1.16 Documentation

shared on codelist.cc

Quick start Guide - Installation and user support

Created: Jan 19, 2015

Updated: Dec 13, 2017

By: AppSeed, <http://appseed.io/>

Thank you for purchasing our product. If you have any question that are beyond of the scope of this help file, please feel free to email via our [support center](#) form. Thank you!

Table of contents:

[Preparing your local environment \(first-time preparations\)](#)

[NodeJS](#)

[Sass](#)

[Git](#)

[Cordova and Ionic](#)

[Tools](#)

[Run for the first time](#)

[Download and extract](#)

[Install libraries](#)

[Run/Build the application \(after the first-time preparations\)](#)

[Run a local development server](#)

[Run in the emulator](#)

[Build the app](#)

[Personalize the app](#)

[Themes - Customize the appearance](#)

[Activate a theme](#)

[Configuration](#)

[Data Sources Creation](#)

[Firebase](#)

[Create a Firebase app](#)

[Set Database Permissions](#)

[Create/Add data](#)

[Firebase Data source Configuration](#)

[Backendless](#)

[Create a new app](#)

[Create data](#)

[Connect mobile app with your Backendless data](#)

[JSON Data sources](#)

[Businesses](#)

[Common](#)

[News, Products, Services and Catalogs](#)

[Wordpress](#)

[JSON API plugin](#)

[Drupal](#)

[Services and JSON View](#)

[JSON Data Sources Configuration](#)

[Push Notifications](#)

[Create Security Profiles](#)

[Plugins](#)

[Set Ionic Platform's credentials](#)

[Set GCM id](#)

[Send a Push Notification](#)

[POSTMAN](#)

[Ionic Platform UI](#)

[Support](#)

[References / Links:](#)

[Thank you](#)

Preparing your local environment (first-time preparations)

NodeJS

Firstly, ensure that [NodeJS](#) is installed in your computer. If NodeJS is not installed, please install version **6.x**.

Sass

In order to install Sass on Linux, Windows or Mac follow the [given instructions](#).

Git

[GIT](#) should be installed in your computer. Follow the instructions for your platform to install it:
<https://www.atlassian.com/git/tutorials/install-git>

Cordova and Ionic

This is a [Cordova](#) and [Ionic](#) based application, so [Cordova](#) and [Ionic](#) should be installed on your computer for this application to run. Since this app targets iPhone and Android mobile devices, your environment should be properly configured and the corresponding SDK should be installed. If not, you will be still able to run the application into a Browser.

Please check the “[Get Started with Cordova](#)” and “[Getting Started with Ionic](#)” sections in the official Cordova and Ionic websites respectively.

Tools

This project is based on the popular “[Ionic Framework Generator](#)” that boosts the overall development process by integrating a couple of very popular automation tools like [Grunt](#) and [Bower](#).

Install these tools by following the instructions in their corresponding web pages:

1. [Install Bower](#)
2. [Getting started with Grunt - Install the CLI](#)

Run for the first time

In the screen captures that follow, we will demonstrate the process of preparing your

environment and running the project for the first time.

Download and extract

Download the provided .zip file and extract it, you will see something similar to what is shown in the next screen:



The highlighted folder is the Ionic 1 project directory.

Install libraries

Open a terminal window and navigate to `project` directory and install the NodeJS dependencies:

```
$ npm install
```

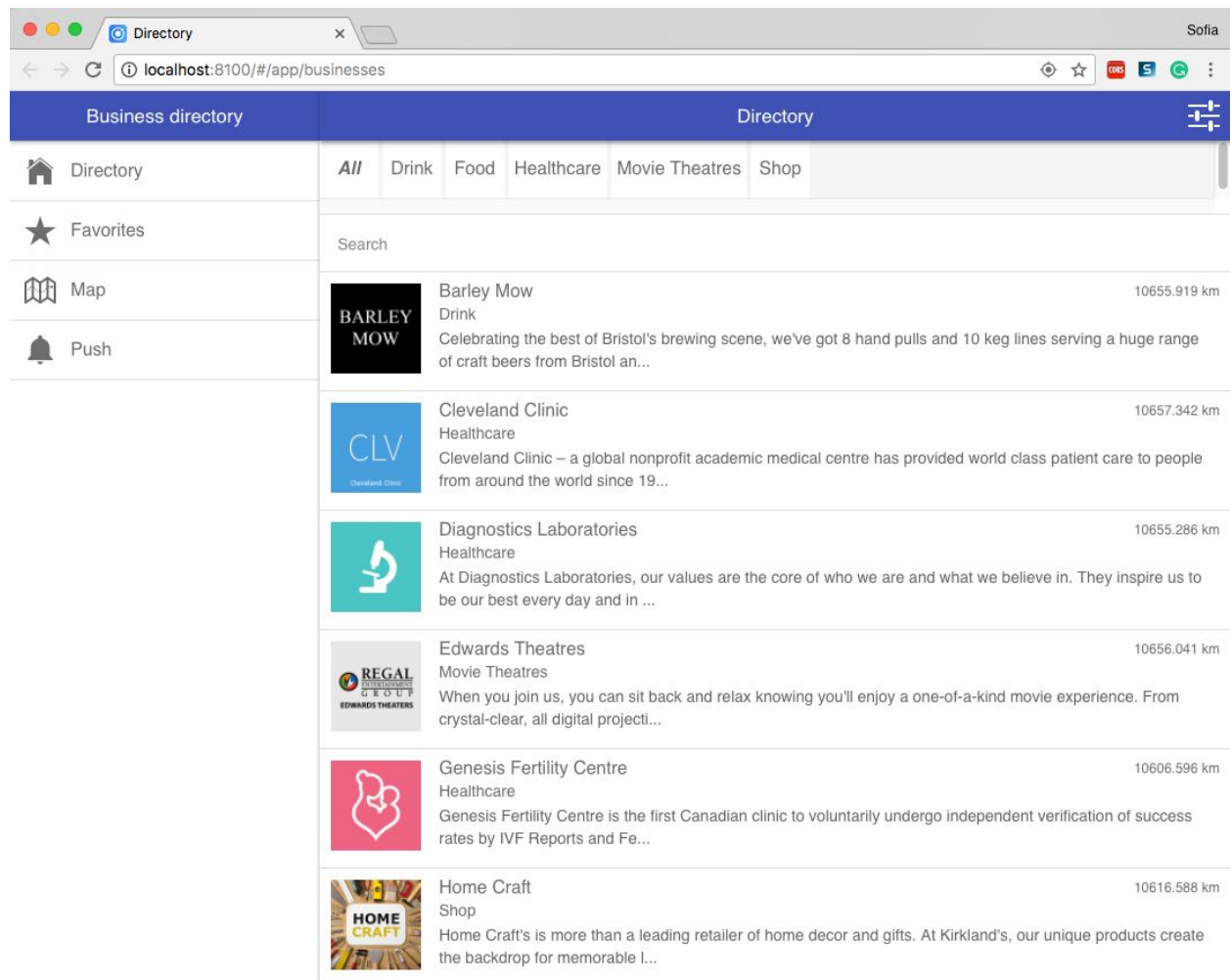
Run/Build the application (after the first-time preparations)

Run a local development server

Navigate to `project` folder and run the application in the browser:

```
$ npm run dev
```

A browser window will open and the app will run.



Run in the emulator

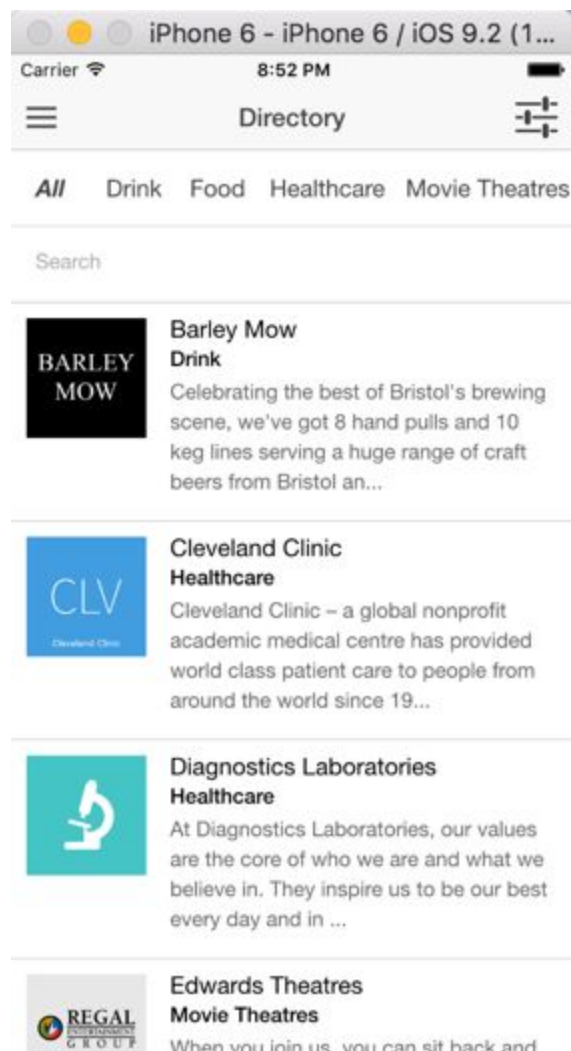
First the preferred platform should be added. To add the iOS platform run:

```
$ npm run add:ios
```

Now the application is ready to start inside a simulator:

```
$ npm run emulate:ios
```

The iPhone simulator will launch and the Business Directory Ionic app will start.



Similarly, to add the Android platform run:

```
$ npm run add:android
```

To run the app inside an Android emulator:

```
$ npm run emulate:android
```

Build the app

In order to build all the added platforms, run the command:

```
$ npm run build
```

After building your project, [www] folder will be created which is the actual cordova directory and where all the plugins and required libraries should be installed.

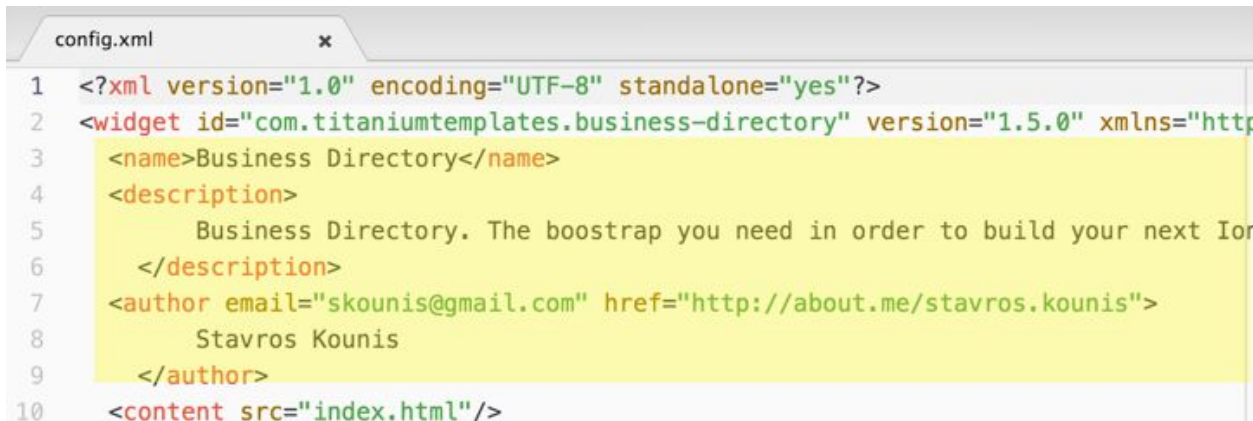
Personalize the app

Once you get familiar with the application, the first step is to personalize it. In order to do this, edit the `ionic.project` and `config.xml` files and replace the highlighted fields:



```
1 {  
2   "name": "business-directory",  
3   "app_id": "e1f02edd"  
4 }
```

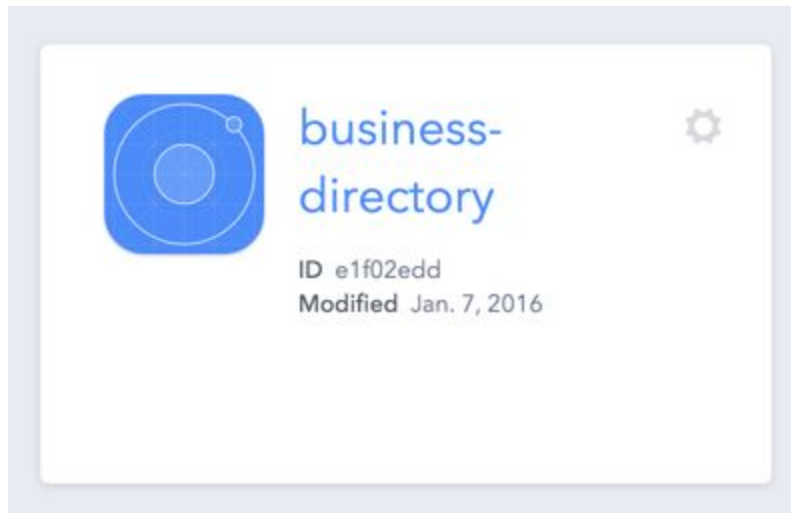
ionic.project



```
1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
2 <widget id="com.titaniumtemplates.business-directory" version="1.5.0" xmlns="http://www.apache.org/licenses/LICENSE-2.0">  
3   <name>Business Directory</name>  
4   <description>  
5     Business Directory. The bootstrap you need in order to build your next Ionic app.  
6   </description>  
7   <author email="skounis@gmail.com" href="http://about.me/stavros.kounis">  
8     Stavros Kounis  
9   </author>  
10  <content src="index.html"/>  
</widget>
```

config.xml

As for the `app_id`, use the ID of the corresponding application in your ionic.io account.



Themes - Customize the appearance

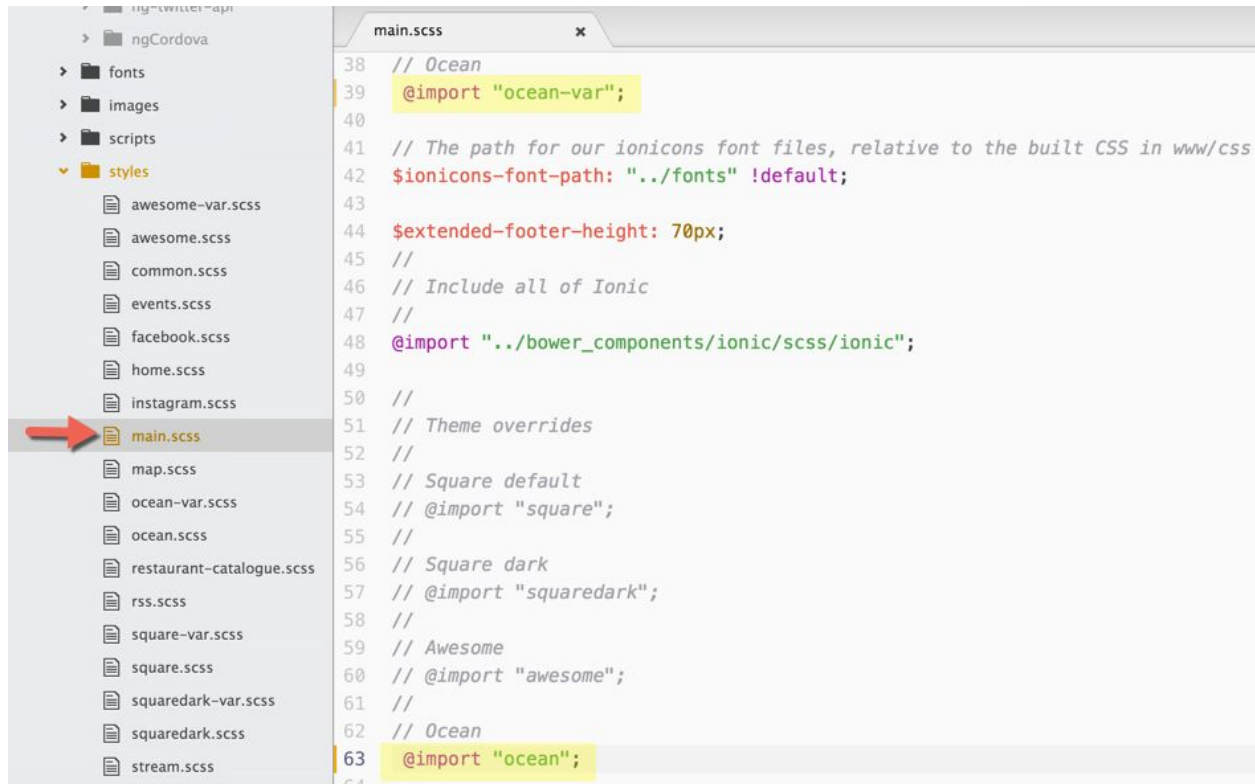
Business Directory Ionic app comes with theme options ready to be used. Each theme consists of two `.scss` files:

1. Variables: `{theme}-var.scss`
2. Overrides: `{theme}.scss`

These files are located under `app/styles/` path and are used in `main.scss` file where a theme can be activated as described in the next section.

Activate a theme

In order to activate a theme, edit `main.scss` file under `app/styles/` path and uncomment the theme you opt for. For example, in case of selecting “Ocean” theme, you should uncomment the lines showing below:



main.scss

Save the changes and the theme is all set up.

Configuration

All the configuration of Business Directory app is done through the creation of the JSON files which include information about businesses, their news, products, catalogs, services and many more.

Data Sources Creation

Firestore

This app uses Firestore as a backend to read the data content it needs. In this section, you will find all the steps required for the app setup in order to be connected with Firestore.

Create a Firestore app

First you should create a free Firestore account and create a new app there.

Create a project

×

Project name

Country/region ?

United States ▼

By default, your Firebase Analytics data will enhance other Firebase features and Google products. You can control how your Firebase Analytics data is shared in your settings at anytime. [Learn more](#)

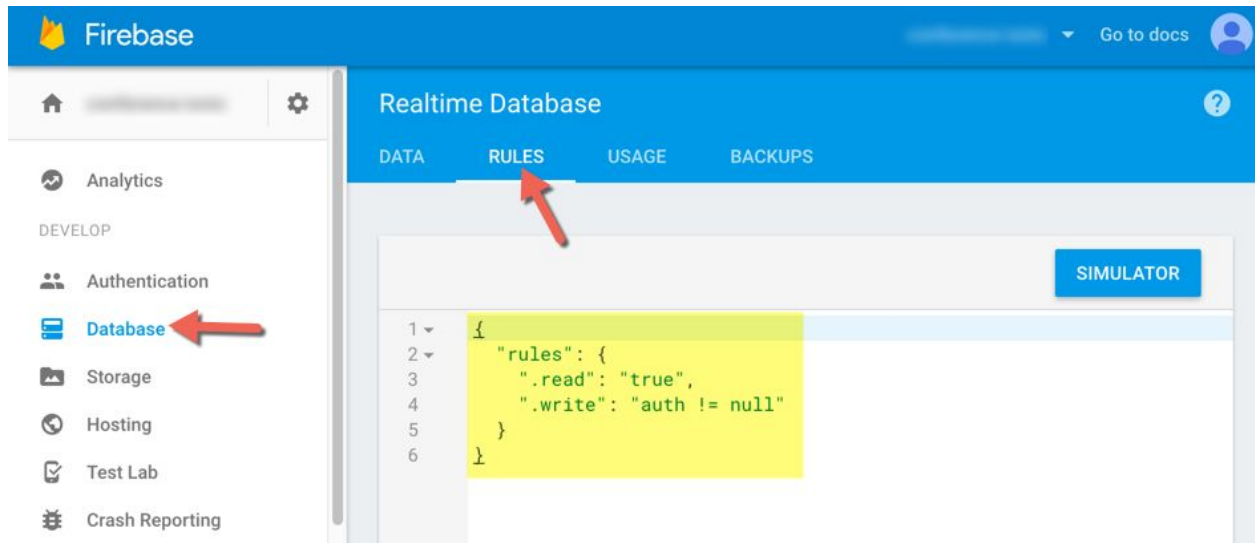
CANCEL

CREATE PROJECT

Set Database Permissions

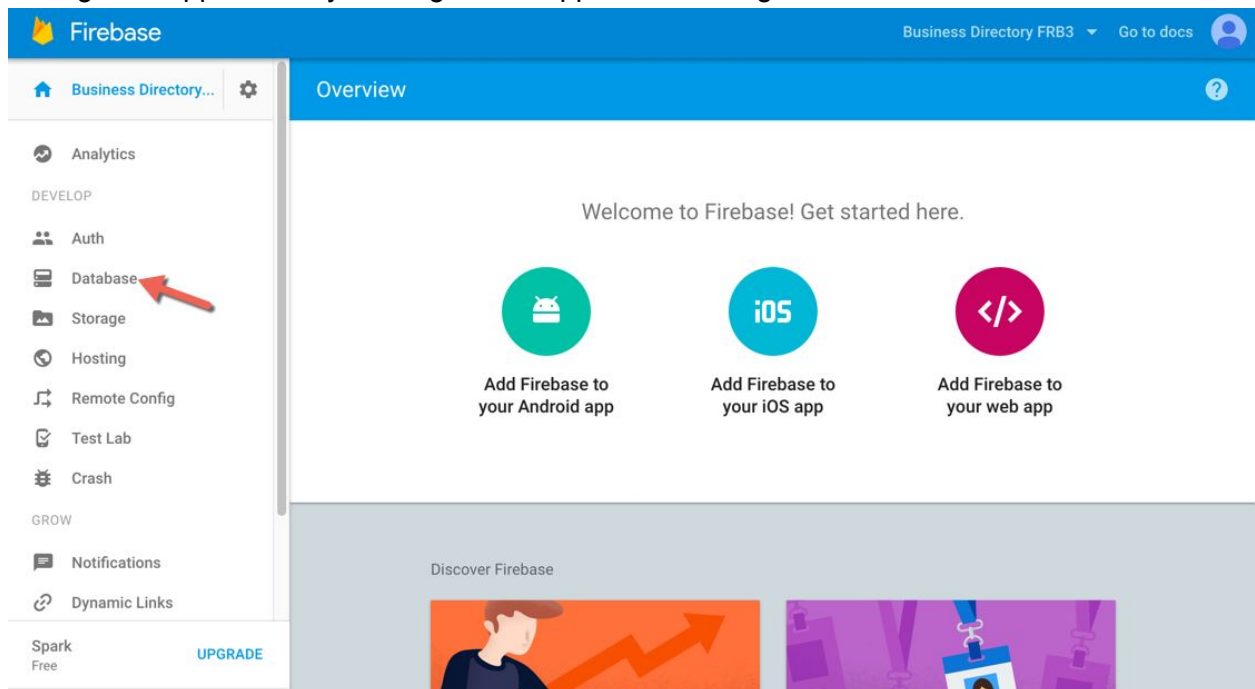
In order to allow the app to read all the data you have created in the database, you should set the permissions by navigating to the “Rules” tab and pasting the following rules:

```
{
  "rules": {
    ".read": "true",
    ".write": "true"
  }
}
```

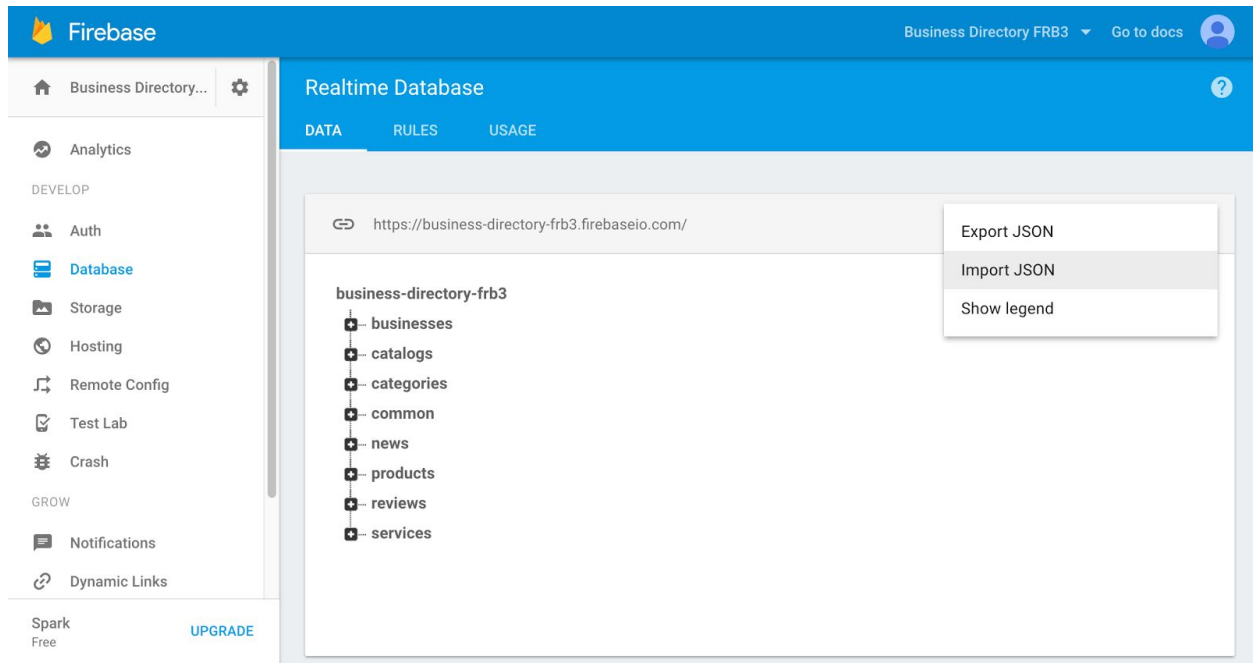


Create/Add data

Manage the app's data by clicking on the app and selecting "Database" from the side menu.



Then, choose the "Import JSON" option from the menu to import your data.



There is an example JSON file under `misc/_firebase/release` path which can be imported to the Firebase project created in the previous step. This file indicates the appropriate data structure for Business Directory.

The final JSON file for the Firebase data import should be similar to the one under the `project/misc/_firebase/release` path.

It is worth noting that this JSON file represents a large JSON object which includes the following data:

- Businesses
- News
- Products
- Services
- Catalogs
- Common

These are explained in more detail in “JSON Data sources” section.

Firebase Data source Configuration

To set the Firebase project the app will work with, you should set the “`dataProvider`” variable as “FIREBASE” and define the “`apiKey`”, “`authDomain`” and “`databaseURL`” in `Gruntfile.js`:

```

Gruntfile.js x
59     },
60     development: {
61         constants: {
62             ENV: {
63                 name: 'development',
64                 // LOCAL or REMOTE or FIREBASE
65                 // dataProvider: 'REMOTE',
66                 // apiUrl: 'https://skounis.s3.amazonaws.com/mobile-apps/business-d',
67                 // dataProvider: 'LOCAL',
68                 // apiUrl: 'misc/',
69                 dataProvider: 'FIREBASE',
70                 firebase: {
71                     apiKey: 'AIzaSyAPinktmcHhcL88Rzm0p98W0cJ30LmEhqA',
72                     authDomain: 'business-directory-frb3.firebaseio.com',
73                     databaseURL: 'https://business-directory-frb3.firebaseio.com',
74                 },
75                 youtubeKey: 'AIzaSyDael5MmCQa1GKQNKQYyp',
76                 ionicPrivateKey: 'a9265eaf15a20cc8516c7',
77                 ionicPublicKey: 'e30d4d540b8c75d1f167bbf242423c3fb23fe10275d1c016',
78                 ionicAppId: '241b6d37', // '2113c758',
79                 gcmId: '228071472080'
80             }

```

Backendless

You can opt for using Backendless as a backend to store the data of your mobile app. In this section, you will find all the steps required for the app setup in order to be configured and connected with Backendless. For more information about Backendless, visit

<https://backendless.com/>.

Create a new app

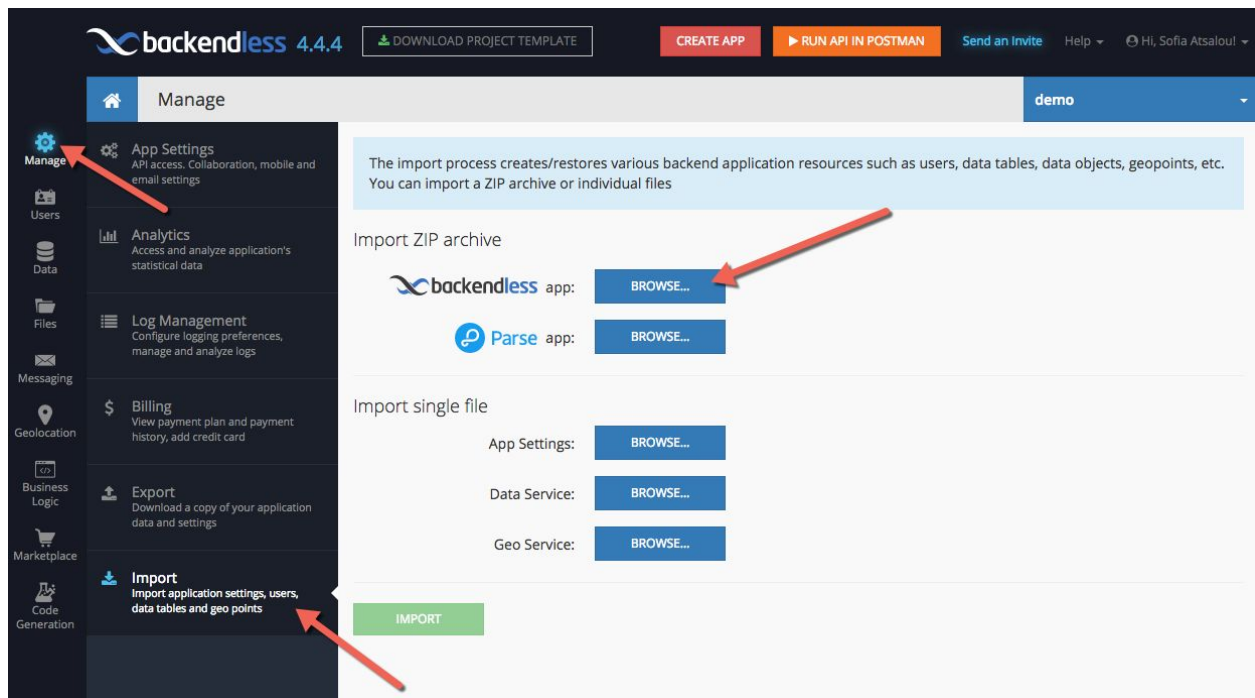
Firstly, visit <https://develop.backendless.com/registration> and create an account and, then, create a new app.

The screenshot shows the Backendless 4.4.4 dashboard. The top navigation bar includes the Backendless logo, version 4.4.4, and buttons for 'DOWNLOAD PROJECT TEMPLATE', 'CREATE APP', 'RUN API IN POSTMAN', 'Send an Invite', 'Help', and a user profile 'Hi, Sofia Atsaloul'. The left sidebar contains icons for 'Manage', 'Users', 'Data', 'Files', 'Messaging', 'Geolocation', 'Business Logic', 'Marketplace', and 'Code Generation'. The main content area is titled 'Dashboard' and displays the configuration for the 'business_directory' app. It includes fields for 'App Name', 'Billing Plan' (Developer), 'App URL in Apple Store', 'App URL in Google Play', 'Application ID', 'iOS API key', 'Android API key', 'JS API key', and 'REST API key'. There are 'RESET APP' and 'DELETE APP' buttons. A dropdown menu is open, showing a list of app names: 'business_directory', 'catalogue', 'hotel', 'real_estate', and 'restaurant'. A red arrow points to the 'Create new app' button at the bottom of the dropdown. Below the configuration section, there are four cards showing usage statistics: 'API/MINUTE' (0/300), 'PUB/SUB MESSAGES' (0/100000), 'PUSH MESSAGES' (0/100000), and 'DISK SPACE' (209 KB/5 GB). Each card has an 'ADD MORE' button. At the bottom, there are tabs for 'THIS MONTH', 'SIX MONTHS', and 'THIS YEAR'.

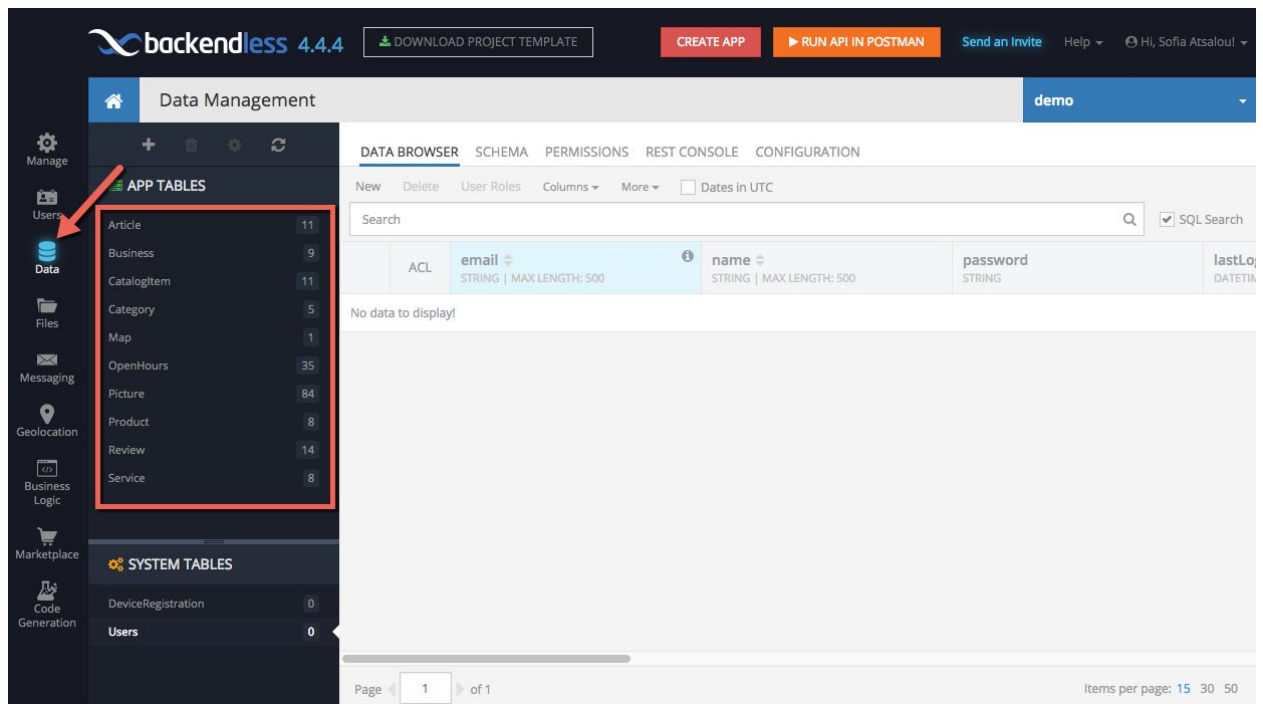
Create data

To create your data, use the sample ZIP file located under the `project/misc/_backendless/` path. In this way, the appropriate data structure of the app will be initialized so you can add/edit your own data using the same structure.

To import the data, go to Manage -> Import and browse the ZIP file as shown in the screen capture below:



Finally, hit the “Import” button and wait a couple of seconds for the import to be finished. At the end, you should be able to see all the data you imported by clicking on “Data” from the side menu.



To enable the app to load the related data/objects, select each one of the tables and make sure the “autoload” option is checked where there is a relation.

backendless 4.4.4 [DOWNLOAD PROJECT TEMPLATE](#) [CREATE APP](#) [▶ RUN API IN POSTMAN](#) [Send an Invite](#) Help Hi, Sofia Atsaloul

Data Management demo

APP TABLES

Table Name	Count	ACL	Columns	Relations	autoload
Article	11		body (TEXT MAX LENGTH: 21000)	business (1:1 Relation)	<input checked="" type="checkbox"/>
Business	9				
CatalogItem	11		Ut vulputate te dolor ipsum nonum...	business (1:1 Relation)	
Category	5		In iusto et ullamcorper dolore dolor,...	business (1:1 Relation)	
Map	1		Te eu dolore ullamcorper nulla aliqu...	business (1:1 Relation)	
OpenHours	35		Te eu dolore ullamcorper nulla aliqu...	business (1:1 Relation)	
Picture	84		Nostrud et vel nulla praesent luptat...	business (1:1 Relation)	
Product	8		Nostrud et vel nulla praesent luptat...	business (1:1 Relation)	
Review	20		Te eu dolore ullamcorper nulla aliqu...	business (1:1 Relation)	
Service	8		Erat consequat blandit enim lobortis...	business (1:1 Relation)	
System Tables					
DeviceRegistration	0		In iusto et ullamcorper dolore dolor,...	business (1:1 Relation)	
Users	0		Nostrud et vel nulla praesent luptat...	business (1:1 Relation)	
			Ut vulputate te dolor ipsum nonum...	business (1:1 Relation)	

Page 1 of 1 Items per page: 15 30 50

For more information about how to create data and relations, read Backendless official [documentation](#).

Connect mobile app with your Backendless data

To connect the mobile app with your Backendless data, open the `Gruntfile.js` file, uncomment the corresponding “dataProvider” and set the application ID and the JS API key as shown:


```

Gruntfile.js
60     development: {
61         constants: {
62             ENV: {
63                 name: 'development',
64                 // LOCAL or REMOTE or FIREBASE
65                 // dataProvider: 'REMOTE',
66                 // apiUrl: 'https://skounis.s3.amazonaws.com/mobile-apps/business-d',
67                 // dataProvider: 'LOCAL',
68                 // apiUrl: 'misc/',
69                 // dataProvider: 'FIREBASE',
70             },
71             firebase: {
72                 apiKey: 'AIzaSyAPinktmcHhcl88Rzm0p98W0cJ30LmEhQA',
73                 authDomain: 'business-directory-frb3.firebaseio.com',
74                 databaseURL: 'https://business-directory-frb3.firebaseio.com'
75             },
76             dataProvider: 'BACKENDLESS',
77             // BACKENDLESS
78             data: {
79                 appId: '23A8662E-FEFD-1784-FF25-94058263E100',
80                 appKey: '77C08DC8-2C95-D7E5-FF8D-0B0DF1D92700'
81             },

```

To find your application ID and your JS API Key, go to Manage -> App Settings and copy the “Application ID” and the “JS API key” as highlighted in the following screen capture:

The screenshot shows the Backendless 4.4.4 web interface. The 'Manage' tab is selected, and the 'App Settings' page is displayed. The 'Application ID' and 'JS API key' are highlighted in yellow. The 'Social Settings' section is also visible at the bottom.

API KEYS	SOCIAL	MOBILE	EMAIL	DOMAIN CONTROL	CUSTOM DOMAIN	GIT	TEAM	DELETE APP
App ID and API Keys								
Application ID	95979667-B2F6-F42C-FFB2-99550FEED600							
iOS API key	03E7F306-CD3F-F3B8-FF8C-0DFB7C740E00							
Android API key	12A66FB0-681E-A0D4-FF55-54F27885F400							
JS API key	97DEE533-4C93-11A7-FFE3-DEEEB3428300							
REST API key	D9585FDD-0231-56D7-FF89-EBDCB3D57A00							
Windows Phone API key	19AEC78-195B-6BE1-FFCD-E9AEBBCE2000							
Code Runner API key	A4344119-2842-3A88-FF6C-6669D97AB000							
Social Settings								
Facebook App ID / API Key:	Backendless default application							

JSON Data sources

In case you chose Firebase as the app's data provider, you can skip this section. Otherwise, this section will guide you in order to make the app read the data it needs from a local or a remote source other than Firebase. In this case the sources should be JSON files located either locally or on a remote server.

Businesses

The example of the JSON file used in order to provide all the business related information is the following:

→ <http://skounis.s3.amazonaws.com/mobile-apps/business-directory/businesses.json>

Notice that this is the file for all the businesses that are going to be displayed in the app. In this file, the addresses to some other JSON files included. These files are:

- `news.json`,
- `catalogs.json`,
- `products.json`
- `services.json`.

The next section describes more the mentioned JSON files.

Additionally, in this file information that is going to be shown on **Home** screen, such as business name, as well as **Contact us** screen, such as business description, open hours etc. is also included.

What's more, `businesses.json` file should contain map annotations that are going to be displayed on the **Map** screen. You are able to set as many annotation points as you wish.

Common

The `origin` property of the map sets the point where it will be centered. This is the **Map** where all the annotations are going to be displayed on.

Origin property together with the zoom level of the map are included in the `common.json` file as it is a piece of information that is in common for all the businesses.

The example of `common.json` file that is used for demonstrational purposes is the following:

→ <http://skounis.s3.amazonaws.com/mobile-apps/business-directory/common.json>

News, Products, Services and Catalogs

Each business that will be displayed in the app should have its own news, products, services and catalogs JSON files.

As a point of reference, the app uses the following:

- News
<https://skounis-dev.s3.amazonaws.com/mobile-apps/business-directory/01/news.json>
- Products
<https://skounis-dev.s3.amazonaws.com/mobile-apps/business-directory/01/products.json>
- Services
<https://skounis-dev.s3.amazonaws.com/mobile-apps/business-directory/01/services.json>
- Catalogs
<https://skounis-dev.s3.amazonaws.com/mobile-apps/business-directory/01/catalogs.json>

These correspond to one business and, thus, they should be created for all the businesses.

Wordpress

This feature loads articles from a remote Wordpress site. The [Wordpress JSON API](#) is used for the creation of JSON feed of the posts. This should be included in the `businesses.json` file.

For demonstration purposes a Wordpress website has been installed. Its URL and JSON feed that is used in this application are provided by the following links:

- Site: <http://demo.titaniumtemplates.com/wordpress/>
- JSON: <http://demo.titaniumtemplates.com/wordpress/?json=1>

JSON API plugin

Please note that, firstly, the [JSON API plugin](#) needs to be installed and activated. The related [documentation](#) is also available. According to that, the generation of the feed is done by finding the location on a website that you want to get a JSON feed and add “`?json=1`” at the end.

Drupal

This feature loads articles from a remote Drupal web site. [Services](#) module is used for the creation of JSON feed of the posts which should be included in the `businesses.json` file.

For demonstration purposes a Drupal website has been installed. Its URL and JSON feed that is used in this application are provided by the following links:

- Site: <http://demo.titaniumtemplates.com/drupal>
- JSON: http://demo.titaniumtemplates.com/drupal/rest/views/rest_api

Services and JSON View

Using Drupal, initially, you should install the [Services](#) module and enable it. A View that exposes a JSON feed of the articles should also be created.

The following tutorial describes this process:

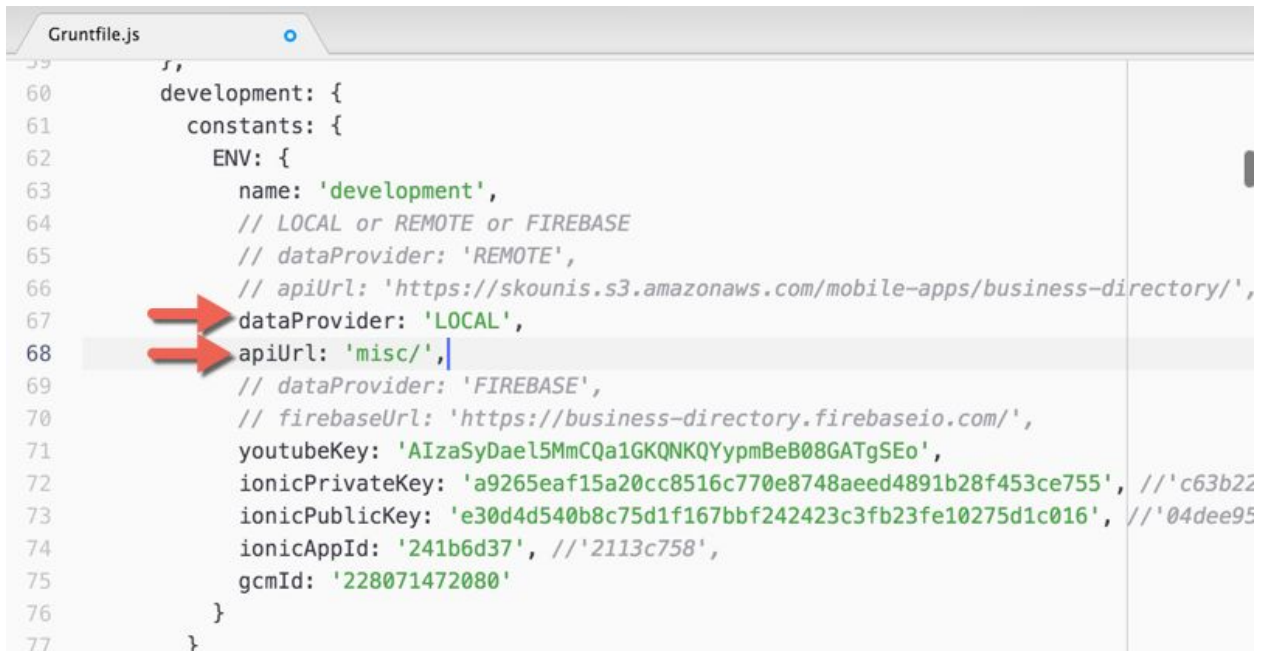
- A Beginners Guide to the Drupal Services Module
<https://www.ostraining.com/blog/drupal/services/>

All the modules that are needed for this are listed below:

1. <https://www.drupal.org/project/services>
2. <https://www.drupal.org/project/ctools>
3. <https://www.drupal.org/project/libraries>
4. <https://www.drupal.org/project/views>
5. https://www.drupal.org/project/services_views

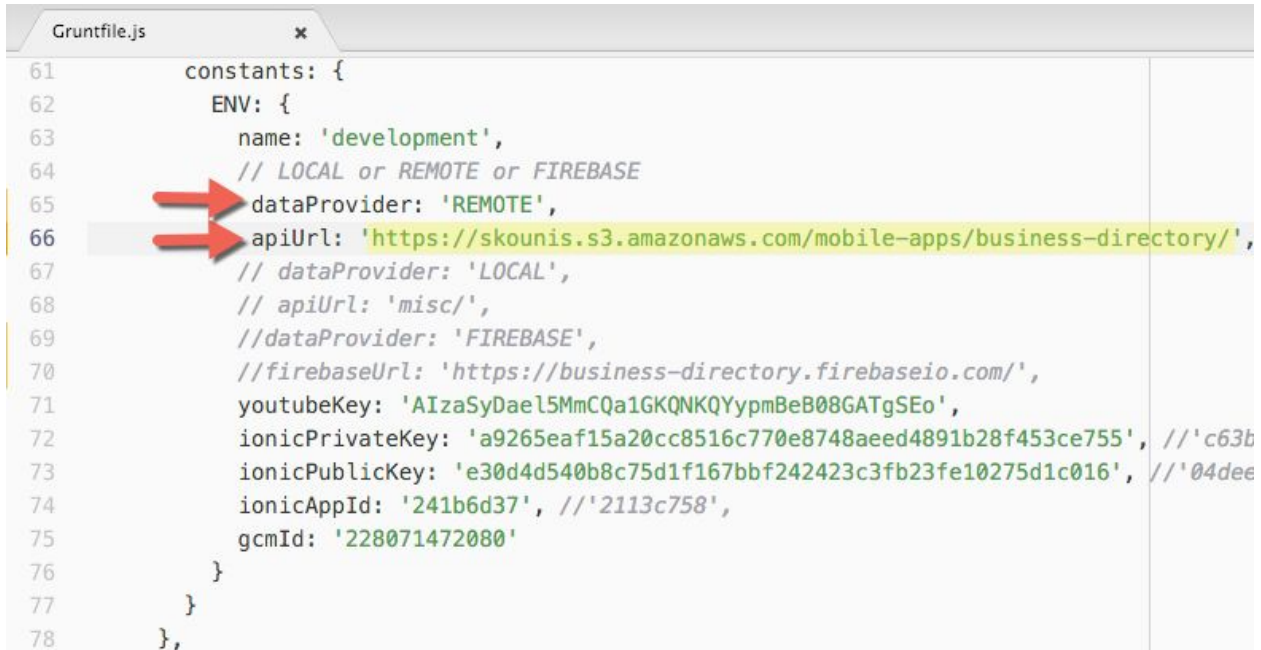
JSON Data Sources Configuration

In case you choose to fetch the data from a local location, you should locate the `businesses.json` and `common.json` files in `misc` folder and set the “dataProvider” variable as “LOCAL” in the `Gruntfile.js`. Make sure you uncomment the related lines as shown below:



```
Gruntfile.js
59  },
60  development: {
61    constants: {
62      ENV: {
63        name: 'development',
64        // LOCAL or REMOTE or FIREBASE
65        // dataProvider: 'REMOTE',
66        // apiUrl: 'https://skounis.s3.amazonaws.com/mobile-apps/business-directory/',
67        dataProvider: 'LOCAL',
68        apiUrl: 'misc/',
69        // dataProvider: 'FIREBASE',
70        // firebaseUrl: 'https://business-directory.firebaseio.com/',
71        youtubeKey: 'AIzaSyDae15MmCQa1GKQNKQYypmBeB08GATgSEo',
72        ionicPrivateKey: 'a9265eaf15a20cc8516c770e8748aeed4891b28f453ce755', // 'c63b22
73        ionicPublicKey: 'e30d4d540b8c75d1f167bbf242423c3fb23fe10275d1c016', // '04dee95
74        ionicAppId: '241b6d37', // '2113c758',
75        gcmId: '228071472080'
76      }
77    }
}
```

In case you choose to fetch the data from a remote location, in `Gruntfile.js` file set the “dataProvider” variable as “REMOTE”. Make sure you uncomment the related lines and set the URL to the folder where the `businesses.json` and `common.json` files are located.



```

61     constants: {
62       ENV: {
63         name: 'development',
64         // LOCAL or REMOTE or FIREBASE
65         dataProvider: 'REMOTE',
66         apiUrl: 'https://skounis.s3.amazonaws.com/mobile-apps/business-directory/',
67         // dataProvider: 'LOCAL',
68         // apiUrl: 'misc/',
69         //dataProvider: 'FIREBASE',
70         //firebaseUrl: 'https://business-directory.firebaseio.com/',
71         youtubeKey: 'AIzaSyDael5MmCQa1GKQNKQYypmBeB08GATgSEo',
72         ionicPrivateKey: 'a9265eaf15a20cc8516c770e8748aeed4891b28f453ce755', //'c63b
73         ionicPublicKey: 'e30d4d540b8c75d1f167bbf242423c3fb23fe10275d1c016', //'04dee
74         ionicAppId: '241b6d37', //'2113c758',
75         gcmId: '228071472080'
76       }
77     }
78   },

```

Push Notifications

Before you proceed further please check the related documentation provided by Ionic:

<https://docs.ionic.io/services/push/>

Create Security Profiles

Firstly, you should create a security profile with push credentials for Android. For more information, please, visit [Ionic official documentation](#). Similarly, you should create a security profile for iOS as the [Ionic documentation](#) describes.

Plugins

Install the required plugins by using the following command where SENDER_ID is your FCM(Firebase Cloud Messaging) project number:

```
$ cordova plugin add phonegap-plugin-push --variable SENDER_ID=12341234
```

Set Ionic Platform's credentials

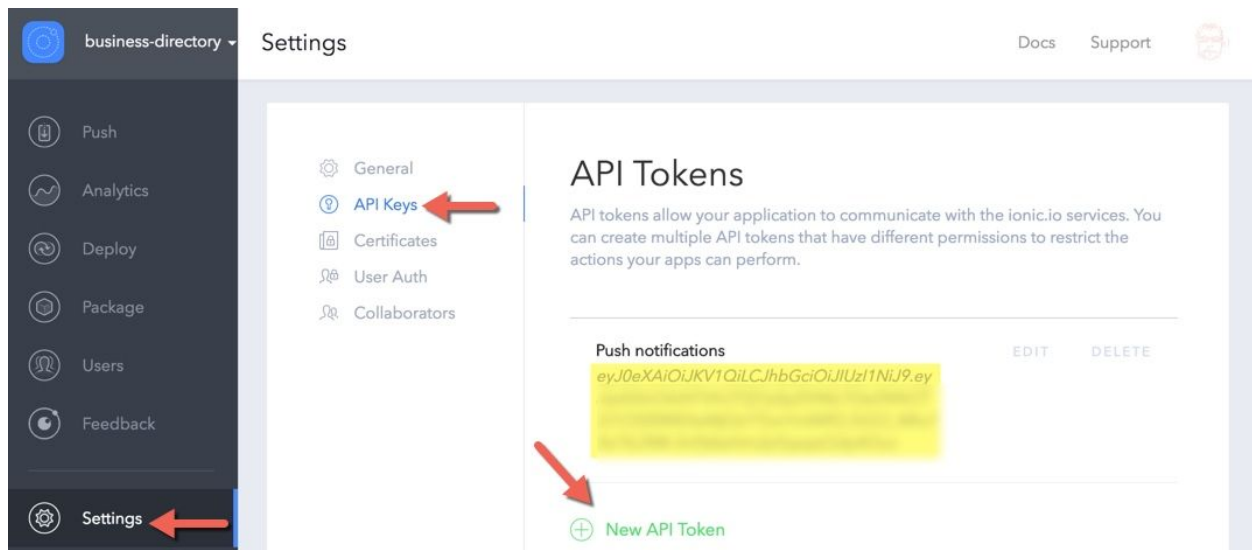
Please, confirm that you have already created the corresponding Application in your ionic.io account and you have properly set the ID of your application in the Gruntfile.js

```

Gruntfile.js
59     },
60     development: {
61         constants: {
62             ENV: {
63                 name: 'development',
64                 // LOCAL or REMOTE or FIREBASE
65                 // dataProvider: 'REMOTE',
66                 // apiUrl: 'https://skounis.s3.amazonaws.com/mobile-apps/business-di
67                 // dataProvider: 'LOCAL',
68                 // apiUrl: 'misc/',
69                 dataProvider: 'FIREBASE',
70                 firebase: {
71                     apiKey: 'AIzaSyAPinktmcHhcL88Rzm0p98W0cJ30LmEhqA',
72                     authDomain: 'business-directory-frb3.firebaseio.com',
73                     databaseURL: 'https://business-directory-frb3.firebaseio.com'
74                 },
75                 youtubeKey: 'AIzaSyDael5MmCQa1GKQNKQYypmBeB08GATgSEo',
76                 ionicCloudApiToken: 'eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqdGkiOi
77                 ionicSecurityProfile: 'clouddev', // https://apps.ionic.io/app/
78                 ionicAppId: '241b6d37', //'2113c758',
79                 gcmId: '228071472080' // Google Developers Console
80             }
90         }

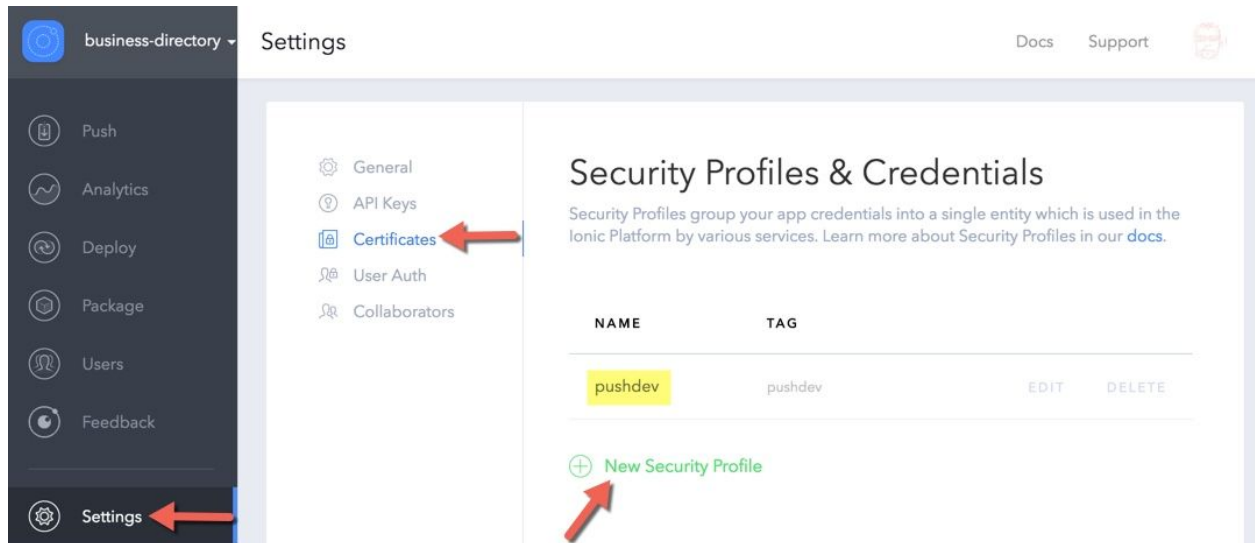
```

Also, you will need an API token so you can use Ionic API to send push notifications. Simply, go to the [Ionic Platform](#) and hit settings to create a token. Please, make a note of this token:



The screenshot shows the Ionic Platform web interface. On the left, a sidebar menu has 'Settings' highlighted with a red arrow. The main content area is titled 'API Tokens' and includes a list of tokens under 'Push notifications'. One token is visible, highlighted in yellow: 'eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqdGkiOi'. A red arrow points to the 'New API Token' button at the bottom of the list.

Additionally, define your iOS and Android security profiles by creating a certificate on the [Ionic Platform](#). Please, make a note of your certificate name:



Finally, set your `ionicCloudApiToken` and `ionicSecurityProfile` with your API token and certificate name in `Gruntfile.js`:

```
Gruntfile.js
59     },
60     development: {
61       constants: {
62         ENV: {
63           name: 'development',
64           // LOCAL or REMOTE or FIREBASE
65           // dataProvider: 'REMOTE',
66           // apiUrl: 'https://skounis.s3.amazonaws.com/mobile-apps/business-di
67           // dataProvider: 'LOCAL',
68           // apiUrl: 'misc/',
69           dataProvider: 'FIREBASE',
70           firebase: {
71             apiKey: 'AIzaSyAPinktmcHhcL88Rzm0p98W0cJ30LmEhQA',
72             authDomain: 'business-directory-frb3.firebaseio.com',
73             databaseURL: 'https://business-directory-frb3.firebaseio.com'
74           },
75           youtubeKey: 'AIzaSyDael5MmCQa1GKQNKQYypmBeB08GATgSEo',
76           ionicCloudApiToken: 'eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqdGkiOi
77           ionicSecurityProfile: 'clouddev', // https://apps.ionic.io/app/
78           ionicAppId: '241b6d37', // '2113c758',
79           gcmId: '228071472080' // Google Developers Console
80         }
78       }
79     }
80   }
```

Set GCM id

Note that in order to push notifications to the registered Android devices, you will need to set the `gcmId` (FCM project number) in `Gruntfile.js` as shown below:

```

Gruntfile.js
59     },
60     development: {
61         constants: {
62             ENV: {
63                 name: 'development',
64                 // LOCAL or REMOTE or FIREBASE
65                 // dataProvider: 'REMOTE',
66                 // apiUrl: 'https://skounis.s3.amazonaws.com/mobile-apps/business-di
67                 // dataProvider: 'LOCAL',
68                 // apiUrl: 'misc/',
69                 dataProvider: 'FIREBASE',
70                 firebase: {
71                     apiKey: 'AIzaSyAPinktmcHhcL88Rzm0p98W0cJ30LmEhqA',
72                     authDomain: 'business-directory-frb3.firebaseio.com',
73                     databaseURL: 'https://business-directory-frb3.firebaseio.com'
74                 },
75                 youtubeKey: 'AIzaSyDael5MmCQa1GKQNKQYypmBeB08GATgSEo',
76                 ionicCloudApiToken: 'eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqdGkiOiI
77                 ionicSecurityProfile: 'clouddev', // https://apps.ionic.io/app/
78                 ionicAppId: '241b6d37', //'2113c758',
79                 gcmId: '228071472080' // Google Developers Console
80             }

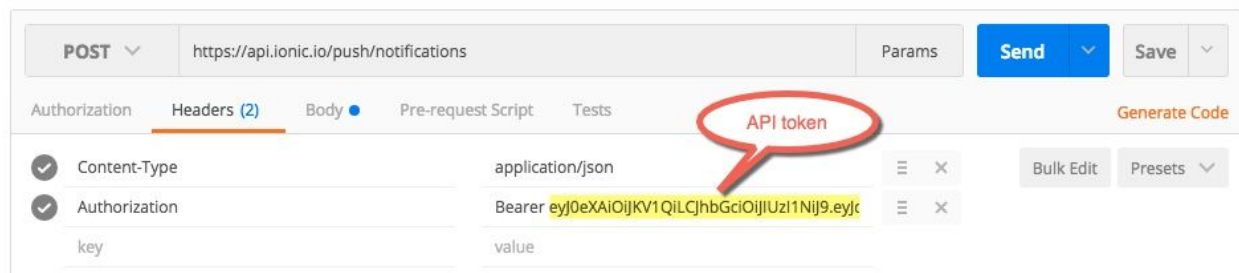
```

Send a Push Notification

Please, bear in mind that you can send push notifications only to real devices. In order to do this, you can use [POSTMAN](#) or the UI of the Ionic Platform.

POSTMAN

To send a notification to all the registered devices, you should use the related [API endpoint](#) and make a POST call. Also, you should have the following headers. For the authorization use the API token you created previously.



Set the body and place your certificate name where highlighted:

POST `https://api.ionic.io/push/notifications` Params Send Save

Authorization Headers (2) Body Pre-request Script Tests Generate Code

form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 {  
2   "send_to_all": "true",  
3   "profile": "clouddev",  
4   "notification": {  
5     "message": "This is a test push"  
6   }  
7 }
```

You may need to troubleshoot in case of a failed push notification. In order to see the state of your push notification and any error messages use the related [API endpoint](#). The GET request should look similar to the following:

GET `https://api.ionic.io/push/notifications/7c7b46a7-cb1f-4d04-88f3-d6742d851331/messages` Params Send Save

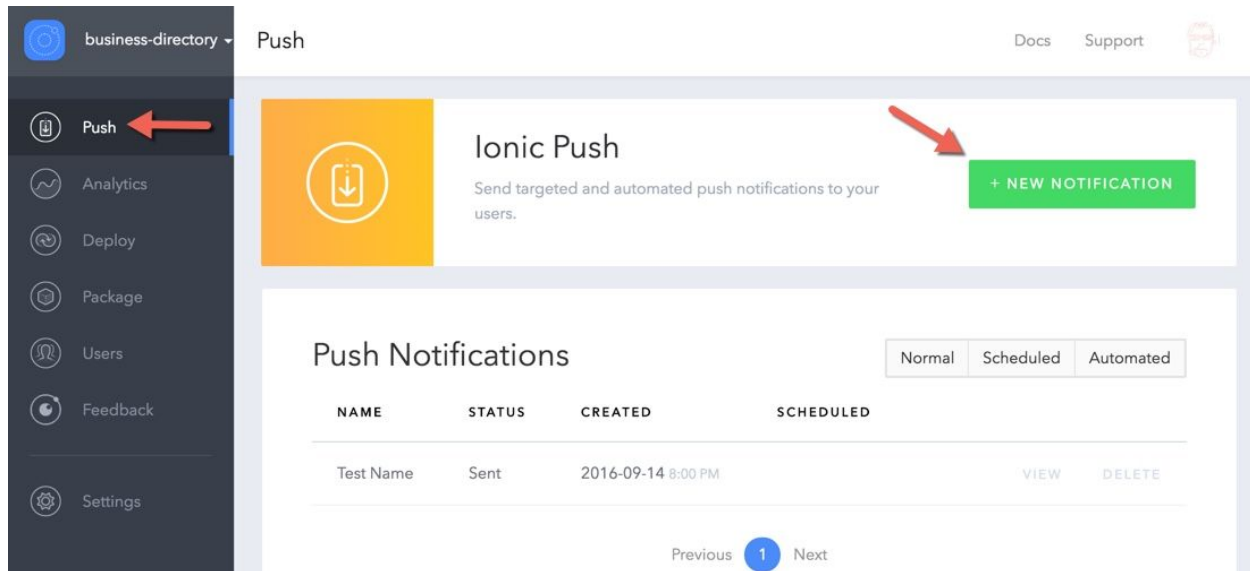
Authorization Headers (2) Body Pre-request Script Tests Generate Code

✓ Authorization	Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJqd...	≡ ×
✓ Content-Type	application/json	≡ ×
key	value	

Bulk Edit Presets

Ionic Platform UI

To send a notification using the UI of Ionic Platform, simply, create a new push notification.



Support

With regard to technical questions, new ideas and suggestions, you may use the dedicated form and choose the product your enquiry refers to:

<http://support.appseed.io/customer/portal/questions/new>

References / Links:

- [YouTube channel](#)
Periodically, video demonstrations and tutorials related to this product will be published in this YouTube channel.
- [Codecanyon User page](#)
You may contact us by using our user page on Codecanyon.
- [Quick Start Guide](#)
The online version of this document.

Thank you

Thank you again for purchasing our product. If you have any questions that are beyond of the scope of this help file, please feel free to email also via our [support center](#) form.

--- The Appseed team.