

# Bizzy

## User Guide

Thanks for purchasing **Bizzy** | Android Multi-purpose Business App Template + Website, we really appreciate your support and we're sure you'll have so much fun with this template!

If you need to learn the process of publishing apps on **Google Play Store**, read the official documentation: **Official Documentation**

### - App icons and Launch Screen -

You can easily change App icons, logo, buttons and other images by accessing **drawable** folder, its path is **app/src/main/res/drawable**.

The **launch\_screen.png** image is into the **drawable-xxhdpi** folder

For the **App icons**, you have to replace the png images included into all the **mipmaps** folders

### - Configuration -

Check out the variable called **WEBSITE\_PATH** into **Configs.java** and replace its string with the path of your website (where you'll upload all the files of the **website** folder of this package) . For example, if your will upload the files in the room of your website (which usually is the **public\_html** directory), and your website is [mywebsite.com](http://mywebsite.com), the right string to place in the websitePath variable would be:

<http://mywebsite.com/>

\* Remember to put the slash symbol (/) at the end of your url, as shown below

### - The website -

The power of this app is its website, so check out the website folder included in the package and edit the **index.html** file with Sublime text, Dreamweaver, Atom or any text editor software.

**NOTE: the app works only after you have uploaded the files of the website folder into your own server**

Please also check the **contact\_me.php** file that's into the **mail** folder and replace the fake email address with your own email address (the one where users will be able to contact you by filling the contact form on the bottom of the webpage):

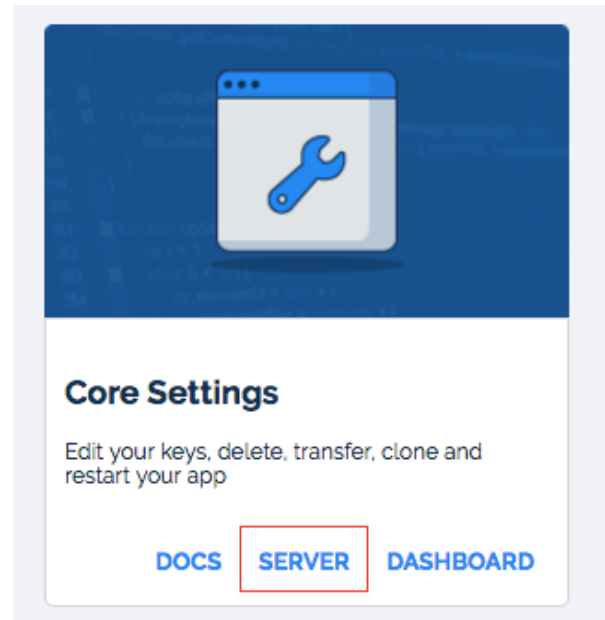
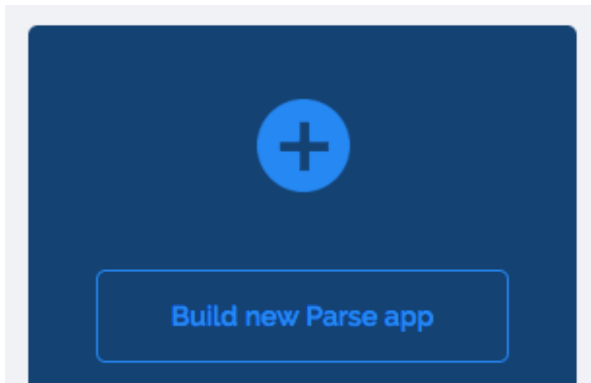
```
$to = 'yourname@yourdomain.com';
```

**IMPORTANT: your website must be hosted by a hosting provider that has PHP enabled (GoDaddy, HostGator and many other have it as default)**

### - Parse setup on back4app (for Push Notifications) -

In order to being able to send Push Notifications, you must create your own App on <https://dashboard.back4app.com/apps> and a Bundle Identifier as mentioned in the beginning of this guide. The link we've provided above points already to the back4app Dashboard, but in case you don't have an account on [back4app.com](https://back4app.com) yet you'll be redirected to the sign up page.

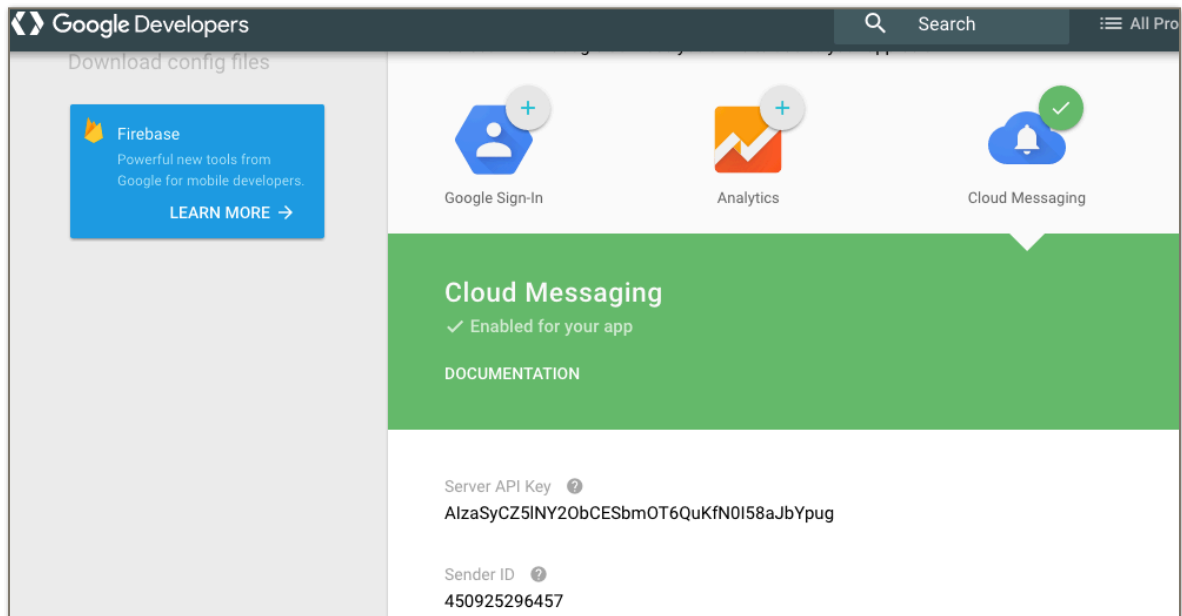
Once you'll be signed in in back4app, enter your dashboard and simply click on the **Build new Parse app** button, give your app a name and a brief description (optional), click **Next** and you'll be redirected to the settings page. Click on **SERVER** in the **Core Settings** box, copy the **App ID** and **Client Key** strings and replace the existing ones into **Configs.java**



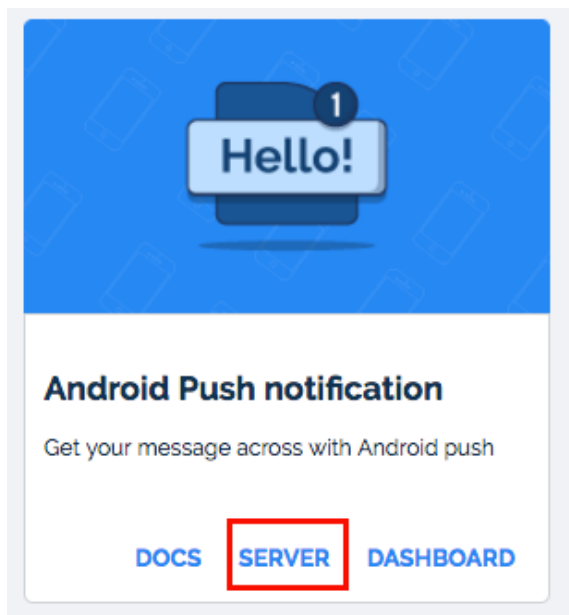
```
public static String PARSE_APP_ID = "fj56THxx0DSbc1edh0Nnc855zyrbXBhHBGkxiR6q";  
public static String PARSE_CLIENT_KEY = "CPZWimlrB33E6o1eRFR7dPHUcjNMuaLp6e8CJlhS";
```

### - Push Notifications setup -

1. Create your app in your Google Developers console at: <https://developers.google.com/mobile/add?platform=android&cntapi=gcm> by giving it a name and pasting your own packager name. Then copy the new generated **Server API Key** and **Sender ID**.



2. Enter your App Dashboard on back{4}app, scroll down to **Android Push Notification**, click on it and paste your **Server API key** and **Sender ID**, than click **Next**:



Android Push Credentials

**GCM Sender ID \***

ex: 000000000000

[Didn't find it? See how to get it.](#)

**API Key \***

ex: abcd1234abcd1234abcd1234abcd1234abcd123

[Didn't find it? See how to get it.](#)

BACK

NEXT

3. Now enter **AndroidManifest.xml** in Android Studio, check all the comments that start with **<!-- IMPORTANT: Replace...** and do what they say.

4. Enter **Home.java** and replace "**253891093066**" with your own GCM Sender ID in this line of code, into **onStart()**:

```
// IMPORTANT: REPLACE "253891093066" WITH YOUR OWN GCM Sender ID
installation.put("GCMSenderId", "253891093066");
```

5. Wait for about half an hour, give time to Google to process Cloud Messaging records, then you will be able to receive Push Notifications and send them to all registered devices via the Parse Push console on back4app:

The screenshot shows the 'Bizzy' web interface for sending push notifications. The left sidebar contains navigation links: 'Core', 'Push' (highlighted), 'Send New Push' (highlighted with a red box), 'Past Pushes', 'Audiences', 'Analytics', and 'App Settings'. The main area is titled 'Send a new campaign' and 'Write your message'. It includes a toggle for 'Text' (selected) and 'JSON'. A text input field contains the message 'Check out our app, we've got news for you!' (highlighted with a red box). Below the input is a section for 'Increment the app badge?' with 'No' and 'Yes' radio buttons. At the bottom, there is a 'Preview' link, a status bar indicating 'It will be sent immediately.', and two buttons: 'Cancel' and 'Send push' (highlighted with a red box).

### - Ad banners setup -

■ **AdMob Unit ID:** You must create a Banner Unit ID on [apps.admob.com](https://apps.admob.com) in order to get its Unit ID and replace the string in **strings.xml** with your own ID, in order to make your AdMob banner work in the app.

```
<string name="ADMOB_BANNER_UNIT_ID">ca-app-pub-9733347540588953/7805958028</string>
```

Have fun with Bizzy, and don't forget to rate it on your Downloads page!



[fvimagination.com](http://fvimagination.com)

shared on [codelist.cc](http://codelist.cc)