

BuildTool

The powerful Cross-Platform app builder

This documentation covers all the things you need to know about BuildTool to get started. Read this documentation carefully, and when you really can't get out, don't hesitate to contact us.

Title:	BuildTool
Created:	13/1/2014
Latest update:	23/3/2014
By:	AppResponsive
Email:	buildtool@robinbonnes.nl

TABLE OF CONTENTS

INTRODUCTION	2
INSTALLATION	3
QUICK START	5
CREATING / INSTALLING TEMPLATES.....	8
CREATING / INSTALLING APPS.....	10
TESTING / PUBLISHING APPS.....	20
UPDATING BUILDTOOL.....	23
CHANGELOG	24

INTRODUCTION

First of all welcome to BuildTool and thanks for purchasing!

With BuildTool you can create your own cross-platform HTML(5)/CSS(3) apps within minutes. It can support these platforms:

- Android
- Apple iOS
- Windows Phone
- BlackBerry (in Phonegap 2.x)
- Web OS (in Phonegap 2.x)
- Symbian (in Phonegap 2.x)

After creating your app, you will be able to download it using the PhoneGap Build API, and publish it to the app stores.

Before your start installing BuildTool, make sure you meet the BuildTool requirements:

- [PhoneGap Build Account](#)
- Apache Webserver
- PHP5 or higher
- MySQL Database
- PHP GD Extension
- PHP CURL Extension
- Write Permission

You can check if you meet these requirements by running the installer, or by uploading and running [this](#) requirements checker script, it will give you an overview of the BuildTool requirements.

Buildtool just checked your requirements, and your server **does** meet the requirements!
Move on to the next step.

Requirement		Result
Apache	✓	2.4.4
PHP Version	✓	5.4.16
MySQL	✓	AVAILABLE
GD Library	✓	AVAILABLE
CURL Library	✓	AVAILABLE
Writable: "/uploads"	✓	YES

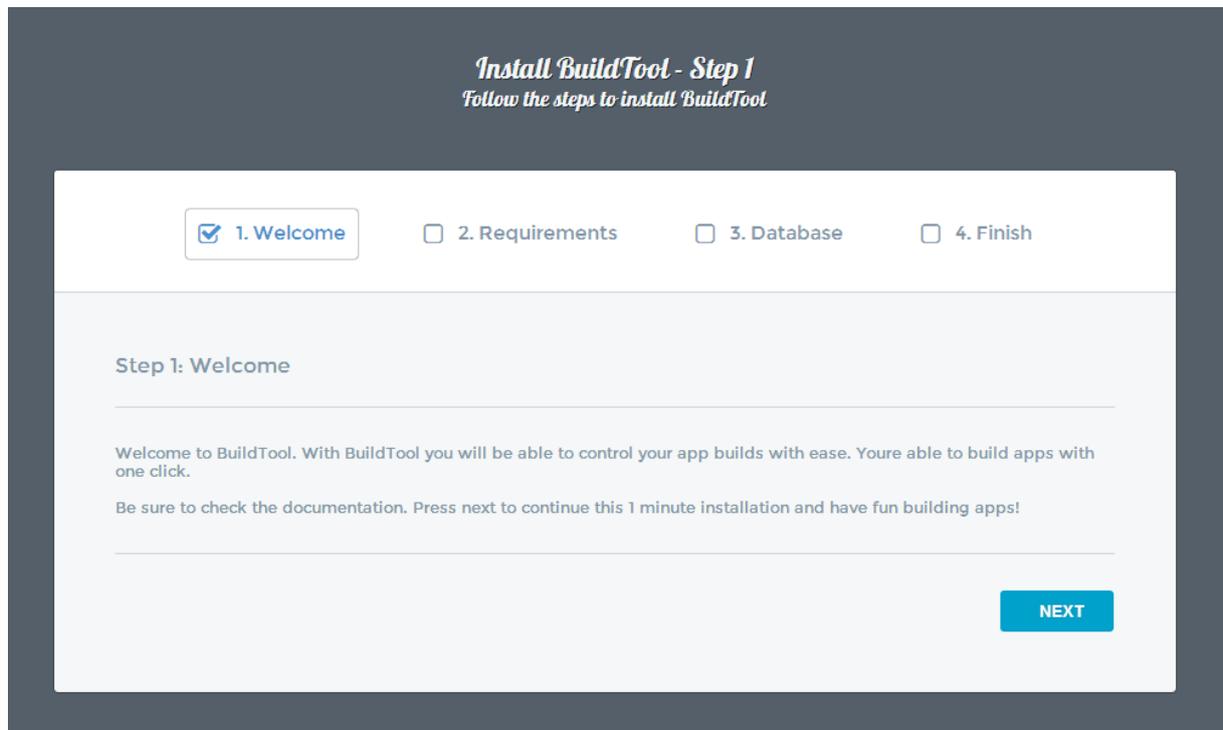
So you're ready? Lets head over to the installation!

INSTALLATION

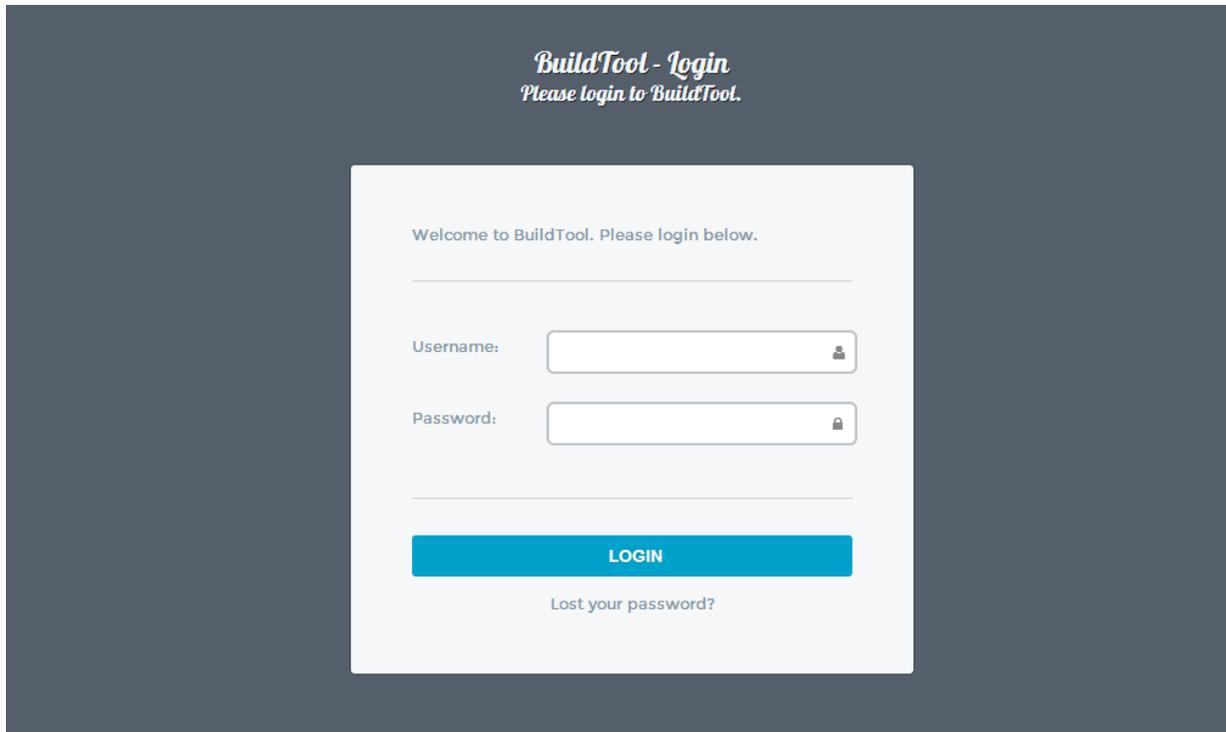
Installing BuildTool is a piece of cake. Unzip the provided zip you've downloaded from CodeCanyon and upload all files in the 'upload' folder to your webserver by your chosen FTP client.



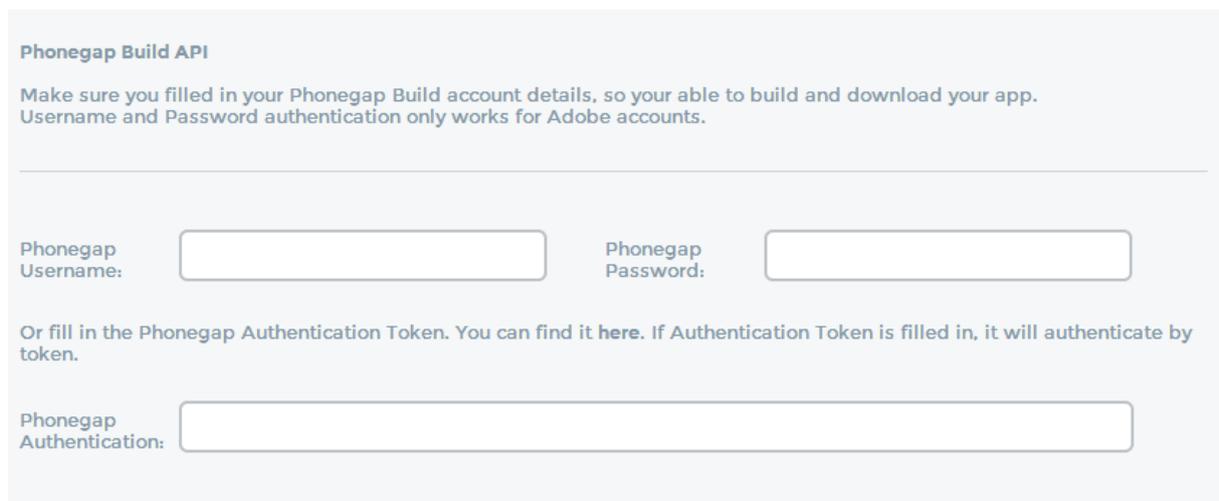
After successfully uploading all files, you should be able to run the installer by simply browsing to the address you've uploaded the files at.



Simply follow the instructions given on screen. Check your requirements, fill in your MySQL database details and create an user account. After installation, head over to the login screen.



Fill in the login details you've just created and login. After logging in, go to settings and fill in your PhoneGap Build API key / login details. You can register a free PhoneGap Build account [here](#), and obtain your API key [here](#).



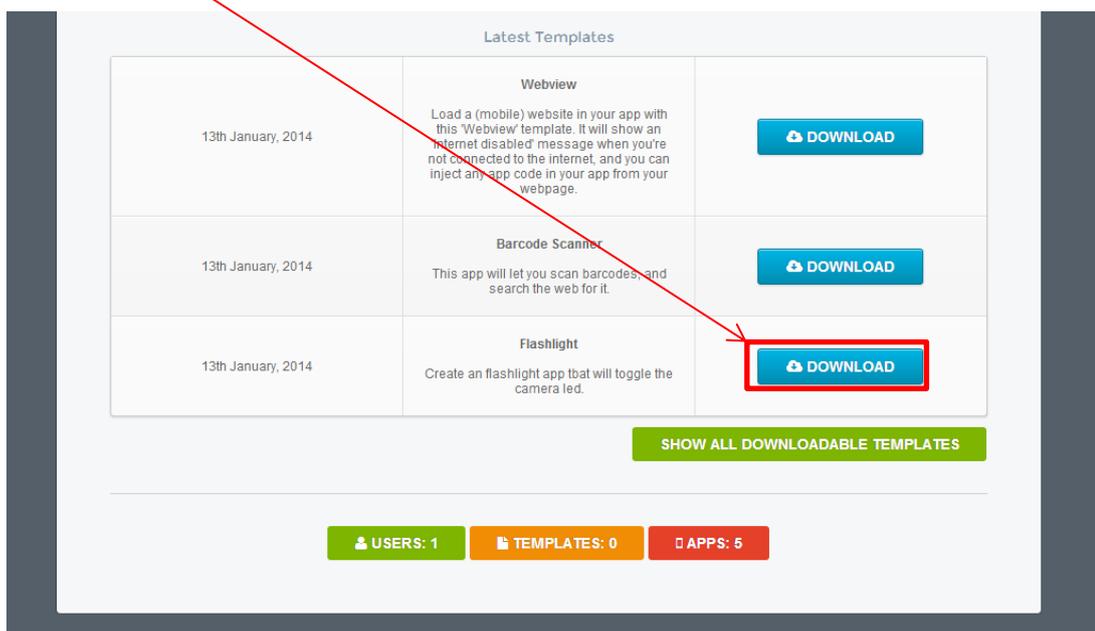
After saving your PhoneGap Build authentication settings, your able to create templates and apps.

QUICK START

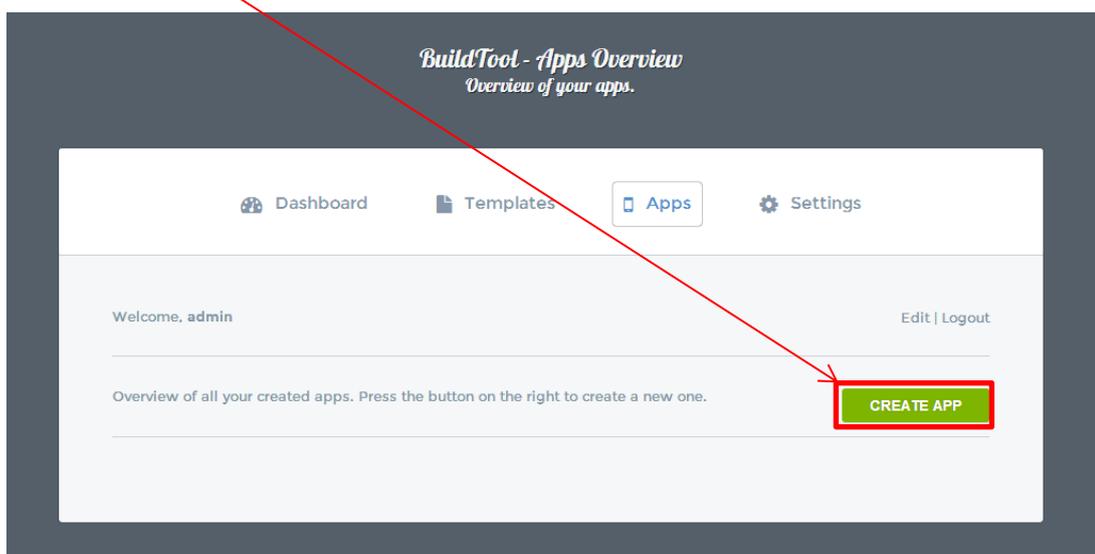
Make sure you've followed the installation steps above to proceed this quick start.

After installing BuildTool your able to create your own apps. For example you want to create an flashlight app with your own about message. This is very easy to do:

1. Download the flashlight template from the BuildTool Dashboard:



2. Click the 'Create App' button on the apps page:



3. Fill in the global settings and template settings:

BuildTool - New App
Create a new app

Dashboard Templates **Apps** Settings

Welcome, admin Edit | Logout

Global Settings
Add your new app here. First provide some global information.

App Name*: My Flashlight App Package Identifier*: com.yourcompany.flashlightapp
Version*: 1.0 Version Code*: 1

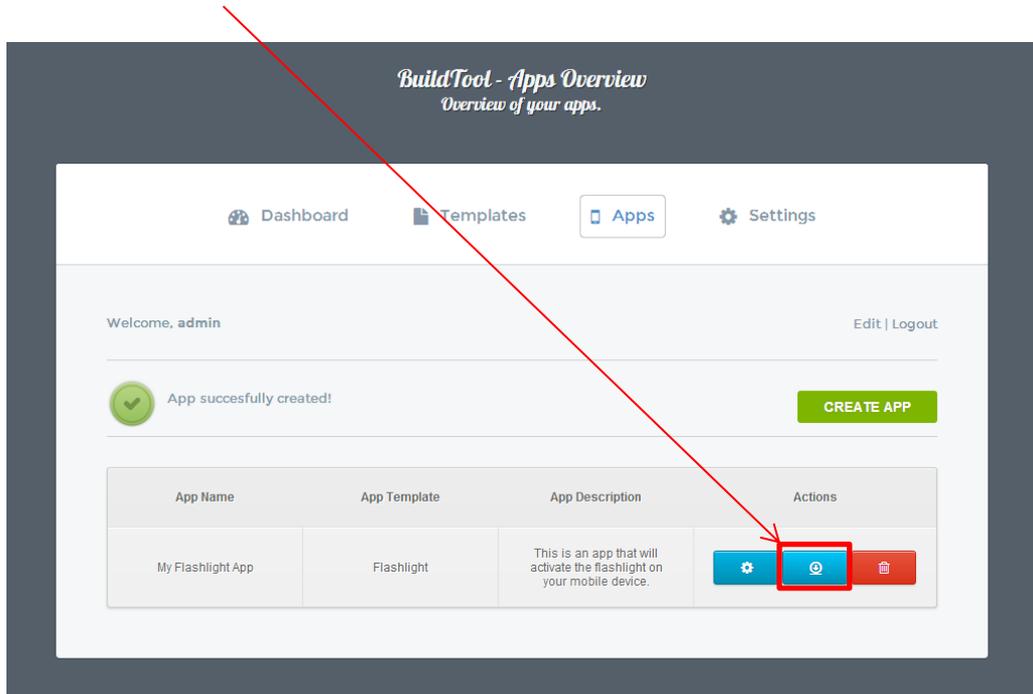
App Description:
This is an app that will activate the flashlight on your mobile device.

Template: Flashlight Debug:

Template Settings
Please fill in the template options.

appinfo: My about message and copyright her

4. Click at the bottom of the page on 'Save'.
5. Select this button on the apps page to build and download your app:



CREATING / INSTALLING TEMPLATES

Creating templates

When you want to create your own template, you should create a default PhoneGap Build project. This is a zip file containing at least a config.xml, index.html and your custom JavaScript code.

In the index.html you should define phonegap.js as following:

```
<script src="phonegap.js"></script>
```

In the config.xml you can create your template options by creating an option element containing the following attributes:

- 'name' -> The name of the option, This should be a string containing letters and underscores and no spaces.
- 'required' -> Is this option required, or can it be left blank? This should be a boolean, true means required, false means not required.
- 'type' -> What type of option is it? You can choose out of: 'textfield', 'select' or 'checkbox'.
- 'options' -> This is only required for options having 'select' as 'type'. Define the options for the select box separated by a '|'.
- 'value' -> This is the (default) value for the option. Should be a string for 'textfield' and 'select', and should be a 1 (checked) or 0 (unchecked) for 'checkbox'.
- 'tooltip' -> If you want to provide additional information, you can add a tooltip about the option, html should be encoded.

Example:

```
<option name="username" required="true" type="textfield" value="admin" tooltip="Please enter your username"/>
<option name="password" required="true" type="textfield" value="admin"/>
<option name="autobackup" required="false" type="select" options="yes|no" value="yes"/>
<option name="show_debuginfo" required="false" type="checkbox" value="0"/>
```

You can read these options in your javascript code like this:

```
var debug = false; // Debug to true when testing in browser, bt_config will be auto-created.
var option_value_you_want_to_read = '';

if(debug)
{
    $.get("config.xml", function(data)
    {
        option_value_you_want_to_read = $(data).find('option[name=option_abc]').attr('value');
    });
}
else
{
    $.get("bt_config.xml", function(data)
    {
        option_value_you_want_to_read = $(data).find('option[name=option_abc]').attr('value');
    });
}
```

Installing templates

Installing templates is very easy. Either pick an existing template from the overview on the dashboard:

Latest Templates		
5th September, 2013	Webview Load every mobile webpage in your app with this Webview template. It will show an internet disabled message when you're not connected to the internet.	<input checked="" type="checkbox"/> INSTALLED
5th September, 2013	Barcode Scanner This app will let you scan barcodes, and search the web for it.	<input checked="" type="checkbox"/> INSTALLED

Or go to Templates -> Add Template and you're able to upload your own template:

Welcome, **admin** Edit | Logout

Add your new template here. Upload your zip package containing the /www folder from your Phonegap project. Make sure your package contains at least an index.html file.

Template Name:

Template Package: **BROWSE**

Template Description:

CANCEL **ADD**

CREATING / INSTALLING APPS

To create apps with BuildTool, go to 'Apps' and fill in the form. The form is split up in five parts:

- **Global Settings:** The global settings for your app
- **Template Settings:** The settings specific for your chosen template
- **Certificates:** The certificates for Android, iOS and BlackBerry
- **Artwork:** The artwork (icons and splash screens) for your app
- **Extras:** Advanced settings like Push Notifications, Stats, Ads etc.

Global Settings

Global Settings
Add your new app here. First provide some global information.

App Name*: Package Identifier*:

Version*: Version Code*:

App Description:

Template: Debug:

In the global settings you have to define the following fields:

- **App Name:** This should be the name that will be shown in the phone's menu.
- **Package/Bundle Identifier:** This is the identifier to recognize your apps on phones and in the stores. Make sure the identifier is the same as your Apple App ID, when you want to publish your app to the App Store. This identifier may only contain uppercase letters, lowercase letters and underscores. Once you published your application, this identifier can't be changed.

- Version: The version number of your app. It's the number shown to the users that download your app. Make sure you raise this number on each app update. (For example from '1.0' to '2.0').
- Versioncode: The version code is an internal version number, used to determine whether one version is more recent than another. Make sure each time you update your app, you raise this number. (For example from '1' to '2').
- App Description: App description is the description you give to your app. It won't be shown anywhere except for in the config file and in BuildTool.
- Template: Select your template, the one you want to build your app for.
- Debug: Whether or not to put your app in debug mode. When your app is in debug mode, it will be signed with an debug key (in Android) and you can debug your app. More information about debugging your app can you find [here](#).

Template Settings

Template Settings

Please fill in the template options.

webview_url*:

The template settings will change per template. Read more about adding templates in the section above. The template options are the options that will be specific for your app.

Certificates

Certificates

If you want to publish your app to the stores, you need to provide your certificates and keys for certain platforms. Read the documentation to get more information about signing your apps.

[ANDROID](#) [IOS](#) [BLACKBERRY](#)

Android, iOS and BlackBerry need certificates if you want to publish them to the stores. For each platform you have to create different certificates:

- **Android:** For Android you have to create a key and a keystore. With this key and keystore you can sign your app and make it ready for publishing in the Play Store. Here is how to obtain your keystore including your private key:
 1. Download and install Java [here](#).
 2. Set Java_Home directory [here](#).
 3. Open the command prompt (cmd.exe) as an Administrator, then Run the following command: **keytool -genkey -v -keystore [keystore_name].keystore -alias [alias_name] -keyalg RSA -keysize 2048 -validity 10000**
 4. keytool will ask for your keystore password. Enter your password and confirm.
 5. Next, keytool will ask for additional information. Supply appropriately or press enter.
 6. Next, keytool will ask a password for the alias. Return if it's the same as the keystore password. Otherwise enter your password and confirm.
 7. Your key is ready for use, fill in the fields in BuildTool as you did when using keytool.
- **iOS:** For iOS you have to create and download your developer certificate from your Apple Developer account [here](#). Follow these video instructions to get your developer certificate on [Windows](#) or [Mac](#). Once you've downloaded your developer certificate, follow the instructions below to convert it to a p12 file.

Mac ([video](#)):

1. Open the Keychain Access application (in the Applications/Utilities folder).
2. If you have not already added the certificate to Keychain, select File > Import. Then navigate to the certificate file (the .cer file) you obtained from Apple.
3. Select the Keys category in Keychain Access.
4. Select the private key associated with your iPhone Development Certificate. The private key is identified by the iPhone Developer: public certificate that is paired with it.
5. Command-click the iPhone Developer certificate and select, Export "iPhone Developer: Name...".
6. Save your keystore in the Personal Information Exchange (.p12) file format.
7. You will be prompted to create a password that is used when you use the keystore to sign applications or transfer the key and certificate in this keystore to another keystore.

Windows ([video](#)):

1. Download and install OpenSSL
2. Convert the developer certificate file you receive from Apple into a PEM certificate file. To do this, run the following command-line statement from the OpenSSL bin directory: **openssl x509 -in developer_identity.cer -inform DER -out developer_identity.pem -outform PEM**

3. If you are using the private key from the keychain on a Mac computer, convert it into a PEM key: **openssl pkcs12 -nocerts -in mykey.p12 -out mykey.pem**
4. You can now generate a valid P12 file, based on the key and the PEM version of the iPhone developer certificate: **openssl pkcs12 -export -inkey mykey.key -in developer_identity.pem -out iphone_dev.p12**

After you generated the p12 file, you need to create your provisioning profile before this, make sure you've created an App ID ([video](#)) and device ID ([video](#)) for testing your app first. Then create and download your provisioning profile ([video](#)). Upload the provisioning profile and p12 file in BuildTool and fill in your password.

- **BlackBerry:** For BlackBerry you have to generate a csk and db file. Follow [these](#) instructions to create them.

Artwork

Artwork

Here you can change your app icon and splash screen. Provide for each platform/device the artwork, or only provide the default icon/splashscreen. Your icons and splashscreen will be sized/scaled when their not provided in the correct format.

[ICON MANAGEMENT](#) [SPLASHSCREEN MANAGEMENT](#)

The artwork for your app exists out of icons and splash screens. You can simply upload just the default icon and splash screen, and all devices will load them. If you want the best visibility, you have to upload your icon's and splash screens in their different sizes for each different platform and/or device as shown in the modal window.

Extras

For each app are many extras available to fine-tune your app. For example push notifications for iOS and Android, or advanced statistics in Google Analytics, or iAds to make money.

Extras – Push Notifications

With Push Notifications you will be able to send unlimited pushes to all devices from within BuildTool. No third-party service is needed that will charge you for send push notifications.

Extra's

Here you can setup your app extras, like Push Notifications, Stats and Ads. Read the documentation to get more info on how to setup Push Notifications, Google Analytics, and Admob.

Push Notifications | Stats | Ads | Preferences | Plugins | iOS Extra's | Android Extra's

Setup **unlimited** Push Notifications. Please fill in the required fields and, you will be able to send notifications in BuildTool after building your app.

Enable Android Push Notifications: ?

Android GCM Sender ID: ?

Android GCM API Key: ?

Enable iOS Push Notifications: ?

iOS Push Certificate (.cer): ?

iOS Push Private Key (.pem): ?

iOS Push Private Key Password: ?

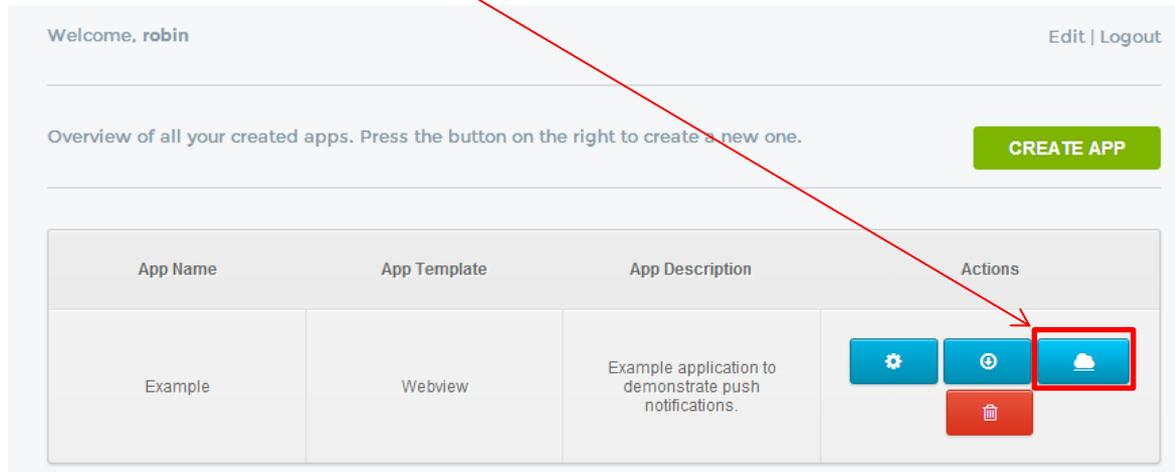
iOS Development Mode: ?

In order to be able to send push notifications you have to fill in some settings.

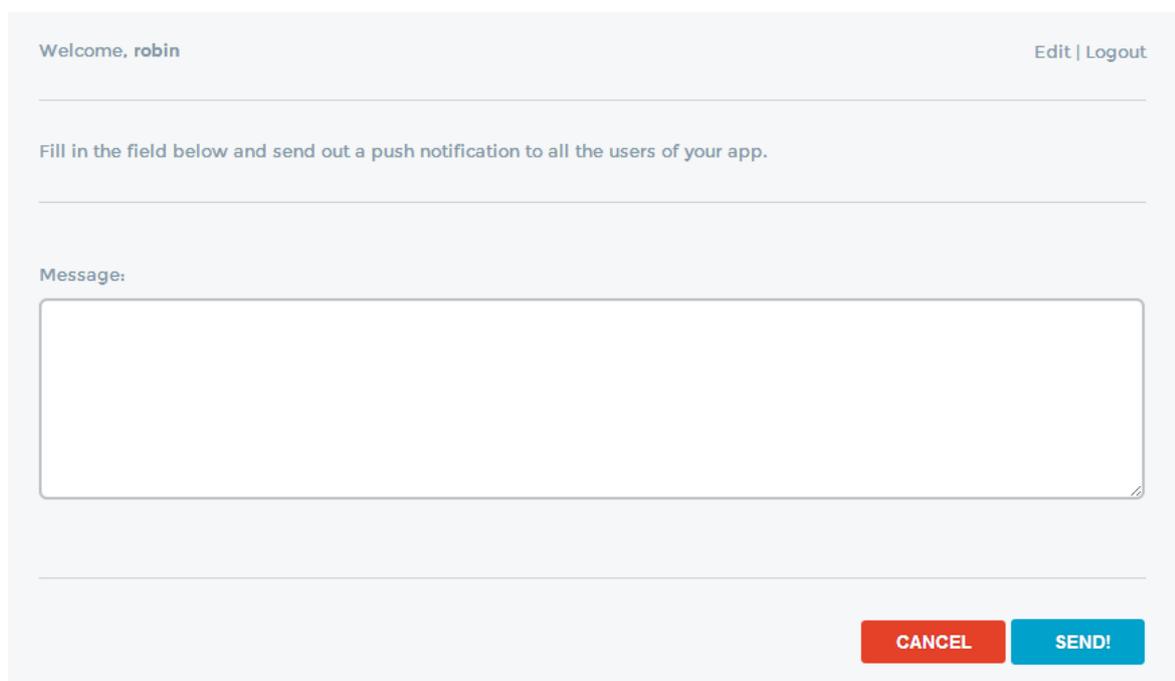
For Android you have to create a GCM Sender ID/ Google Developer Project Number and a GCM API Key. Read [this](#) article for full information about how to obtain these. It takes about two minutes.

For iOS it's a little bit more complicated. You have to create a push certificate and private key and combine to one .PEM file. Read [this](#) tutorial to get this .PEM file.

After filling in your push notification settings, create your app and you will see this icon appearing on the app overview page:



Click that icon, fill in you push notification and hit the send button:



The message will be shown as notification in your app. And if your app isn't open, it will pop up in the notification/statusbar.

Extras – Stats

To get very advanced statistics about your app, you can use Google Analytics Mobile. With this you can find out who uses your apps now, on what devices, where your users come from and many more.

Extra's

Here you can setup your app extras, like Push Notifications, Stats and Ads. Read the documentation to get more info on how to setup Push Notifications, Google Analytics, and Admob.

Push Notifications Stats Ads Preferences Plugins iOS Extra's Android Extra's

Setup Google Analytics to retrieve usage information from your app on Android and iOS. Please fill in the required fields and, you will be able to get all your app statistics in BuildTool after building your app.

Enable Google Analytics:

Google Analytics Account ID:

All you have to do is enable this option and fill in your Google Analytics Account ID. [Here](#) you can register for Google Analytics, and follow [this](#) short tutorial to generate your Account ID.

Extras - Ads

Here you can add iAds to your mobile application and make money.

Extra's

Here you can setup your app extras, like Push Notifications, Stats and Ads. Read the documentation to get more info on how to setup Push Notifications, Google Analytics, and Admob.

Push Notifications Stats Ads Preferences Plugins iOS Extra's Android Extra's

Add Ads to your app. Currently only iAds are available for iOS. Due too Phonegap Build not (yet) allowing Admob as plugin, it isn't available in Buildtool.

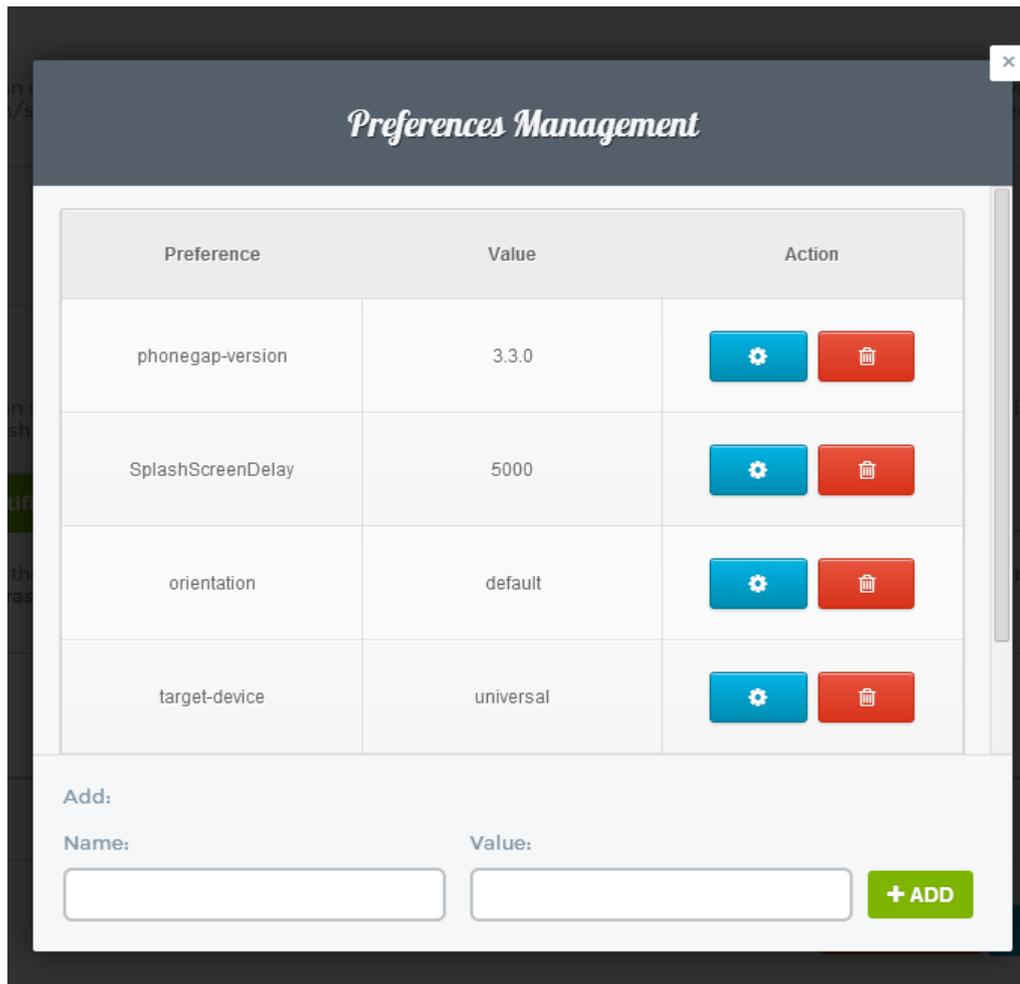
Enable iAds:

Position of ads:

It's as easy as hitting the enable button and specify the position of your ads. Make sure your app is enabled for the iAd network before you submit it to the App Store. You can achieve within the iTunes Connect portal.

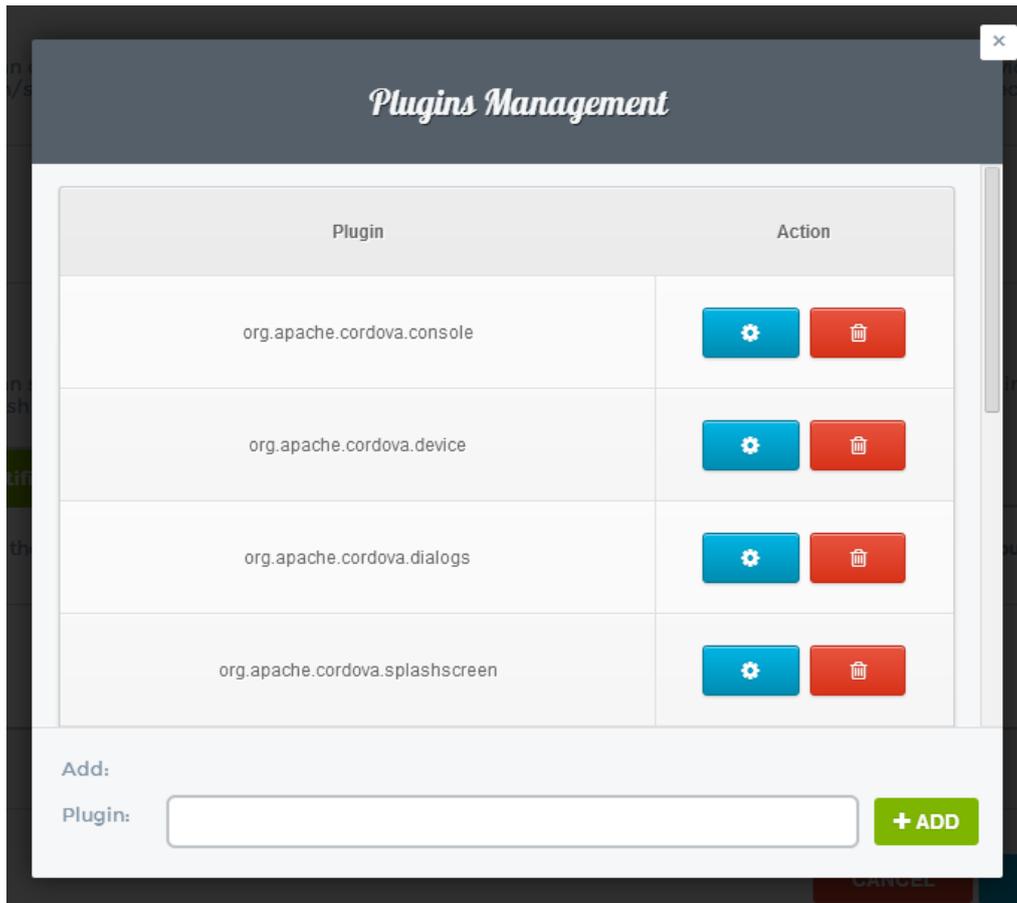
Extras – Preferences

When you click the green button on the preferences tab, you can setup some advanced preferences. Make sure you only change these when necessary, because these preferences may change per template. You can find a full list [here](#).



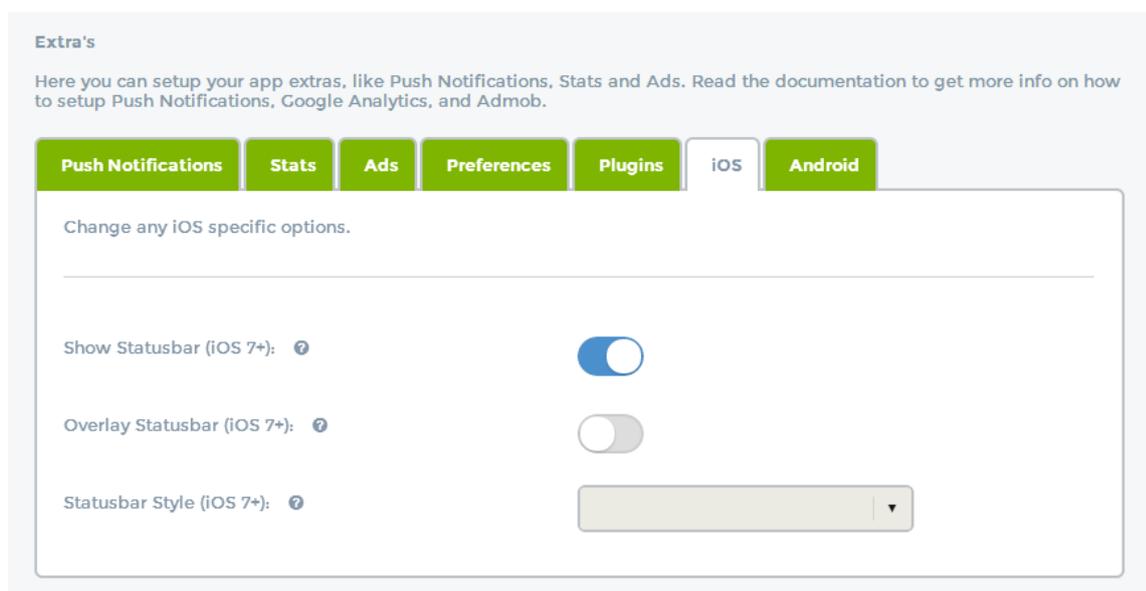
Extras - Plugins

Each app relies on plugins. Most likely the plugins on this list are needed to run your app. You can add plugins to the list when you expand your template or remove some of the features. Don't edit these if you don't need to. You can find a full list of plugins [here](#).



Extras – iOS specific

Setup some iOS specific options. For example the statusbar behaviour in iOS7 and above.



Extras – Android specific

Setup some Android specific options. None available yet.

Extra's

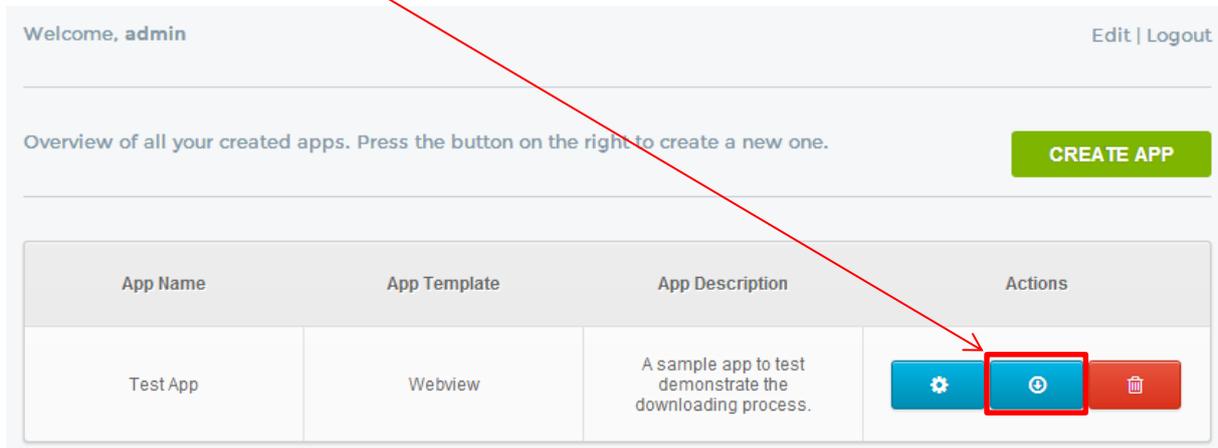
Here you can setup your app extras, like Push Notifications, Stats and Ads. Read the documentation to get more info on how to setup Push Notifications, Google Analytics, and Admob.

Push Notifications Stats Ads Preferences Plugins iOS Android

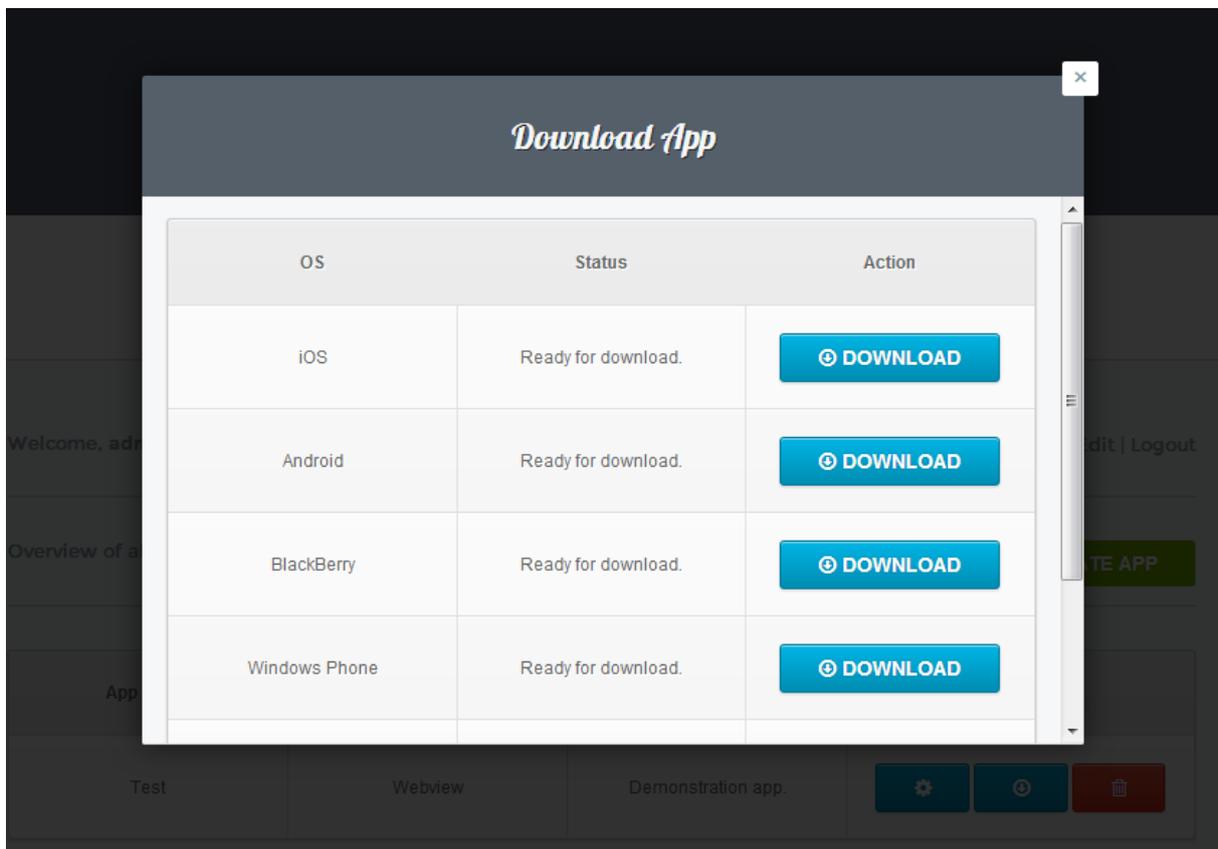
Change any Android specific options. None available yet.

TESTING / PUBLISHING APPS

Before you can test or publish your app to the stores, you have to build and download it. Click on this icon next to your app to get its download link:



It will give you the download links for each platform. Click on the download button next to the platform you want to download your app for:



After downloading your app, you could either test your app, or directly publish your app to the app store.

Testing apps

To test the app download the app for the platform you want to test your app on, and then follow these steps to deploy the app to your device:

- Android:
 1. Copy your app to your mobile phones' SD card.
 2. Go to: Settings -> Applications or Security -> Allow 'App install from unknown sources'.
 3. Download a File Manager from the Play Store, and browse to your app on your SD card.

- iOS:
 1. Make sure your device ID is allowed in the provisioning profile of your app. (See: **Certificates**)
 2. Open iTunes, drag your app into 'Apps', and sync it with your device.

- BlackBerry:
 1. Make sure your app is not signed in BuildTool.
 2. Copy your app to your mobile phones' SD card.
 3. Run the 'Files' application and browse to your app on your SD card.

- Symbian:
 1. Copy your app to your mobile phones' SD card.
 2. Open your File Manager and install the app.

- Windows Phone:
 1. Copy your app to your mobile phones' SD card.
 2. Go to Start -> Store -> SD card and install your app.

- Web OS:
 1. Copy your app to your mobile phones' SD card.
 2. Open your File Manager and install the app.

If you want to debug your app, check out this documentation [here](#).

Publishing apps

To publish your app to the stores, download the app first, and then follow these steps to publish your app:

- Android:
 1. Create an Android Developer account [here](#).
 2. Login in your Developer Console [here](#).
 3. Click the 'Add new application' button and upload your APK file.
 4. Upload your app package and fill in the required information.

- iOS:
 1. Create an Apple Developer account [here](#).
 2. Go to [iTunes Connect](#) -> 'Manage your apps' -> 'Add New app'.
 3. Upload your app package and fill in the required information.

- BlackBerry:
 1. Create a BlackBerry ID [here](#).
 2. Login [here](#) and create a BlackBerry Vendor Account.
 3. Click 'Add Product' and follow the steps.

- Symbian:
 1. Create a Nokia publisher account [here](#).
 2. Go to the [Nokia Publish self-serve tool](#) to add your app.
 3. Upload your app package and fill in the required information.

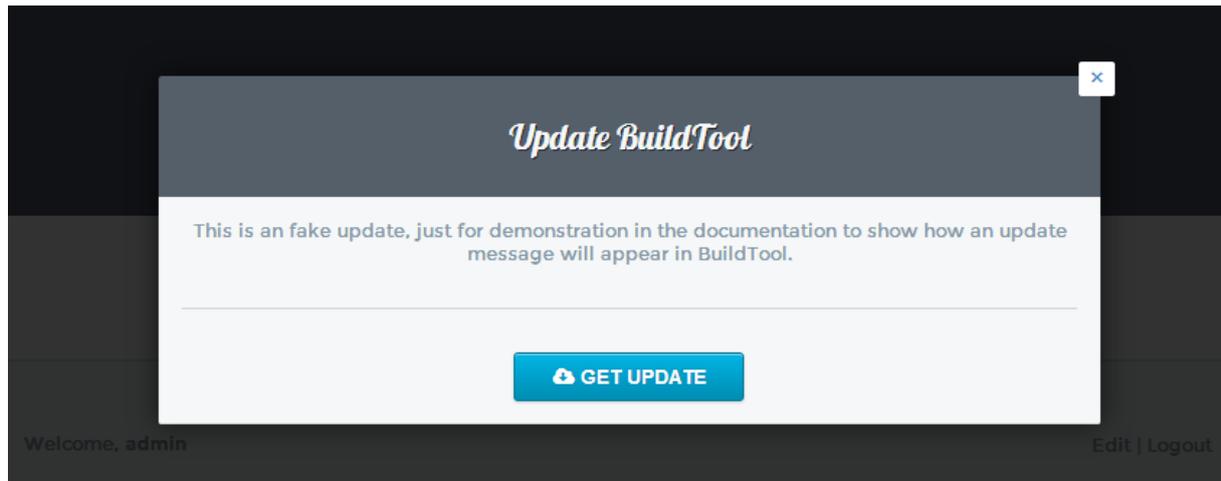
- Windows Phone:
 1. Create a Microsoft Developer Services account [here](#).
 2. Login to your [Windows Store Dashboard](#) and click 'Submit an app'.
 3. Upload your app package and fill in the required information.

- Web OS:
 1. Sign up for an Developer Account [here](#).
 2. After confirming your email fill in the additional information.
 3. Go to [My Apps](#) and click 'Upload new App'.
 4. Upload your app package and fill in the required information.

UPDATING BUILDTOOL

Auto update

When a new BuildTool version gets released, you will be prompted with a message about updating BuildTool:



When you press 'Get Update', BuildTool will auto update its files and database. No settings will be lost.

Manual update

When auto update is not working, download the new zip from CodeCanyon and unzip the archive and overwrite the files by FTP. Now browse to your domain and run the updater.

CHANGELOG

23-03-2014:

v1.5 – Added Push Notification support, Google Analytics, iAds, platform specific options, and more. Fixed bugs in client/admin role system, and some minor bugfixes.

11-03-2014:

v1.4 – Added admin/client user roles. You can add your own billing system for your clients. Please Note: You have to buy an extended license if you charge your clients for a BuildTool account!

30-01-2014:

v1.3 – Fixed uploading iOS splashscreens, added remove option for certificates and artwork, minor bugfixes.

22-01-2014:

v1.2 – Fixed installer, fixed uploading artwork, added tooltip support for template options.

21-01-2014:

v1.1 – Renamed htaccess file, minor bugfixes.

21-01-2014:

v1.0 – Initial release