

Springbot Integration for Magento2

Springbot, an eCommerce marketing platform designed for small to mid-sized online businesses, helps eCommerce stores grow revenue by taking smarter, data-driven marketing actions. If you would like more information on how Springbot can help grow your business we encourage you to [schedule a free demo](#).

This extension connects your Magento2 store to Springbot's marketing platform.

Installation

To install via composer, run the following commands from your Magento2 root directory

```
composer require springbot/magento2-plugin
php bin/magento module:enable Springbot_Queue
php bin/magento module:enable Springbot_Main --clear-static-content;
php bin/magento setup:upgrade
php bin/magento cache:clean
```

If you have compilation enabled, re-run the compilation process:

```
php bin/magento setup:di:compile
```

Once you have installed the plugin you should see a Springbot menu item on your Magento admin menu. Enter your Springbot credentials here.

About the Integration

We've designed our integration to be as lightweight as possible. The extension exposes several Springbot specific endpoints leveraging Magento2's included API to keep your store synced with our service. Once synced, all marketing actions are performed within the Springbot dashboard located at <https://app.springbot.com>.

Springbot utilizes a job queuing system to defer sync jobs so that they may be run asynchronously. By doing this we are able to avoid executing sync related tasks on page loads. Instead, a special queue endpoint is exposed via the Magento2 API that our sync service requests on a periodic basis. When first installed, the sync service performs a full retroactive sync of all existing store data. From then on, when an action is performed on your store that results in the creation/modification/deletion of data, a lightweight job is added to the queue to be processed asynchronously at a later time.

The extension also automatically places a small asynchronous javascript snippet in the footer of each page to track visitors and load 3rd party integrations such as Adroll. Because it is loaded in the

background after the page has fully rendered, the end result is zero impact on the frontend rendering of your pages.

Committing and Contributing

If you are contributing to this project please create a branch with the issue number as the branch name. Ensure any new functionality includes proper testing. When creating new master releases please tag your commit and update the composer.json version accordingly. Github webhooks are currently configured with packagist.org such that any newly tagged versions are pushed to packagist automatically:

```
git commit -m "My Changes"  
git tag 1.0.0  
git push origin --tags
```