

jahia
Digital Industrialization

JAHIA DOCUMENTATION

Private App store installation and usages

Rooted in Open Source CMS, Jahia's Digital Industrialization paradigm is about streamlining Enterprise digital projects across channels to truly control time-to-market and TCO, project after project.

Jahia Solutions Group SA

9 route des Jeunes,
CH-1227 Les acacias
Geneva, Switzerland

<http://www.jahia.com>

Summary

JAHIA DOCUMENTATION.....	1
Summary.....	2
Introduction	3
1 Requirements	3
2 Installation	4
2.1 Download	4
2.2 Configuration.....	4
2.2.1 Repository settings.....	5
2.2.2 Roles.....	6
2.2.3 Categories.....	7
3 Using and managing a Private App Store.....	8
3.1.1 As a developer	8
3.1.2 As a site administrator.....	11
3.2 Digital Factory Studio	12
3.2.1 Select your repository type.....	12
3.2.2 Releasing the module	14
3.2.3 Finalize the release	14
4 Digital Factory administration	17
4.1.1 Define stores	17
4.1.2 Browse available modules	18
4.1.3 Get informed	19

Introduction

The Private App Store Package is a jar file containing modules and templates than can be deployed on a Digital Factory instance to create a browsable repository of modules.

A Private App Store establish a bridge between developers and running instances of Digital Factory, enabling seamless and secure distribution of new or updated modules, reducing risks, time spent on maintenance an TCO.

Developers can publish their modules directly from the Digital Factory Studio. Users with enough privileges (site administrators, webmasters...) can use the App Store to

- Browse the catalog of modules
- Download and install a module manually or
- Install new modules in one click from the Digital Factory GUI
- Stay informed of new versions and
- Install updated modules in one click from the Digital Factory GUI

Each module stored can receive a description and can be categorized, commented, rated, illustrated with screenshots or videos. For each version of a module there can be specific release notes.

1 Requirements

- a Digital Factory 7.001 or higher
- a dedicated Maven repository

You can find more information on maven repositories here

<https://maven.apache.org/guides/introduction/introduction-to-repositories.html>

If you don't have any Maven repositories manager yet, we suggest you to have a look at Nexus
<http://www.sonatype.org/nexus/>.

The repository stores the artifacts uploaded on the Private App Store.

The Private App Store will proxy all the access to the maven repository. It means that Digital Factory users will only be aware of the Private App Store, but not of the Maven repository itself.

2 Installation

2.1 Download

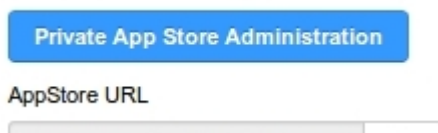
Get the store package from Jahia Public App Store: <http://store.jahia.com/home.html>

From your server modules administration page, upload the package.

This will install the Private App Store module and its template-set, then create a new Private App Store web site ready to be configured.

2.2 Configuration

Once your store is installed, log as root to configure it



The screenshot shows a web interface for 'Private App Store Administration'. It features a blue header bar with the title. Below the header, there is a label 'AppStore URL' followed by a text input field. The input field is currently empty, and a small cursor is visible at the end of the field.

2.2.1 Repository settings

[Private App Store Settings](#) [Private App Store roles](#) [Categories](#)

Private App Store Settings

Url of nexus repository

Repository Identifier

User

Password

Url of nexus repository: Complete url to your repository

Repository identifier: Identifier for the repository

User and password: user on the Nexus repository with write access

2.2.2 Roles

[Private App Store Settings](#)
[Private App Store roles](#)
[Categories](#)

Private App Store roles

Roles



Private App Store administrator

#		Name	Properties
+ Members			

Private App Store developer

#		Name	Properties
+ Members			

reader

#		Name	Properties
1		Guest user	Guest user
2		users	users
+ Members			

Administrator: the user is allowed to moderate modules and perform any operation on any modules

Developer : the user can add new modules, edit them and add new version to his own modules

Reader: if you want to forbid the access to your App Store, you can specify which user can read the store. Note that you will need to specify it in Digital Factory administration to access the store.

2.2.3 Categories

The store uses a flat list of categories.

You need to choose the root of the store categories in your category tree.



Private App Store Settings Private App Store roles **Categories**

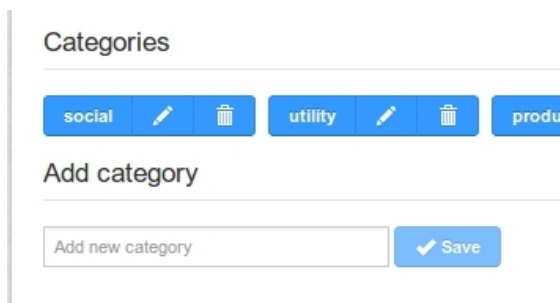
Categories

Root category for Private App Store modules

Selected Root category :

Categories Save

You can also add or edit categories



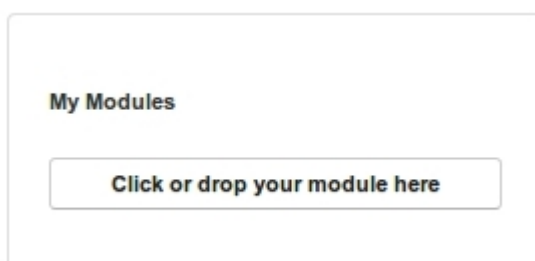
Categories

social utility produ

Add category

Add new category Save

Your store is now ready to be used.



My Modules

Click or drop your module here

3 Using and managing a Private App Store

3.1.1 As a developer

3.1.1.1 Upload a new released module

Developers can upload any Digital Factory released module using the web form

- by clicking on the upload button then select the jar file on your disk
- by simply drag and dropping the jar file on the button

My Modules

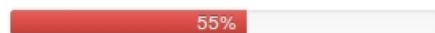


This operation creates a new store entry.

3.1.1.2 Developer panel

Developer Panel

Module page completion



Todo list (mandatory)

- Category
- How to install
- Icon
- Screenshots
- Video
- FAQ
- Tags

View as Publish Delete

Private App Store Admin Panel

Reviewed by Jahia Supported by Jahia

With the developer panel, you can see the unfilled fields of the module, and perform actions:

- **View as:** allow to view your module page as a guest user
- **Publish** : when all mandatory fields are set, publish the module make it available for the store readers
- **Delete:** delete your module.

By default, after an upload, only the category is mandatory.

3.1.1.3 Edit module details



You can add information on your module, clicking on the specific tab. We encourage you to populate it as much as possible.

3.1.1.4 Publishing the module



When all mandatory fields are set, you can publish your module by clicking on the publish button on the developer panel. By publishing the module the first time, you allow to download the latest version of the module if none is already published.



Note that until the module itself is published, it will not be available to download even if a version is published.

Once published, you can click on *unpublish* to remove the module to visitors, completely or a specific version.

3.1.1.5 Upload a new version

From the version panel of a module you can upload a new version of the module, the module artifact ID and group ID must match, if not, the operation will create a new module.

Once a new version is uploaded, you can make it available by clicking on the publish button

[Download version 1.5](#)[Publish](#)[Download version 1.4](#)[Unpublish](#)[Download version 1.3](#)[Publish](#)

If the version 1.4 is the one currently published, this version will be available for download

[📄 Download current version \(1.4\)](#)





Only one version can be published at a time.

3.1.1.6 Manage module permissions

From this panel, it is possible to control who can see (read) the module, by choosing individual users or groups.

[Overview](#) [Screenshots](#) [Video](#) [Reviews](#) [Installation](#) [FAQ](#) [Versions](#) [Permissions](#)**Module permissions****Roles**

reader

#		Name	Properties	Actions
1		Guest user	Guest user	
2		users	users	

[+ Members](#)

3.1.2 As a site administrator

You can define the settings of the store.

You can do all the same operations as an owner of a module on all modules of the store.

For each module, you have the possibility to flag them as reviewed or supported.

- Reviewed usually means that the module has been tested successfully
- Supported means that some kind of guarantee / maintenance is available

If your organization does not use those concepts, just don't use those fields.

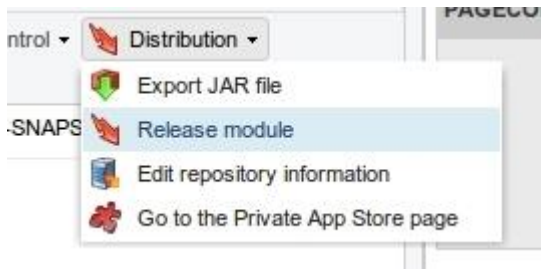
Private App Store Admin Panel[Reviewed by Jahia](#) [Supported by Jahia](#)

Enabled, this will display icons on modules list



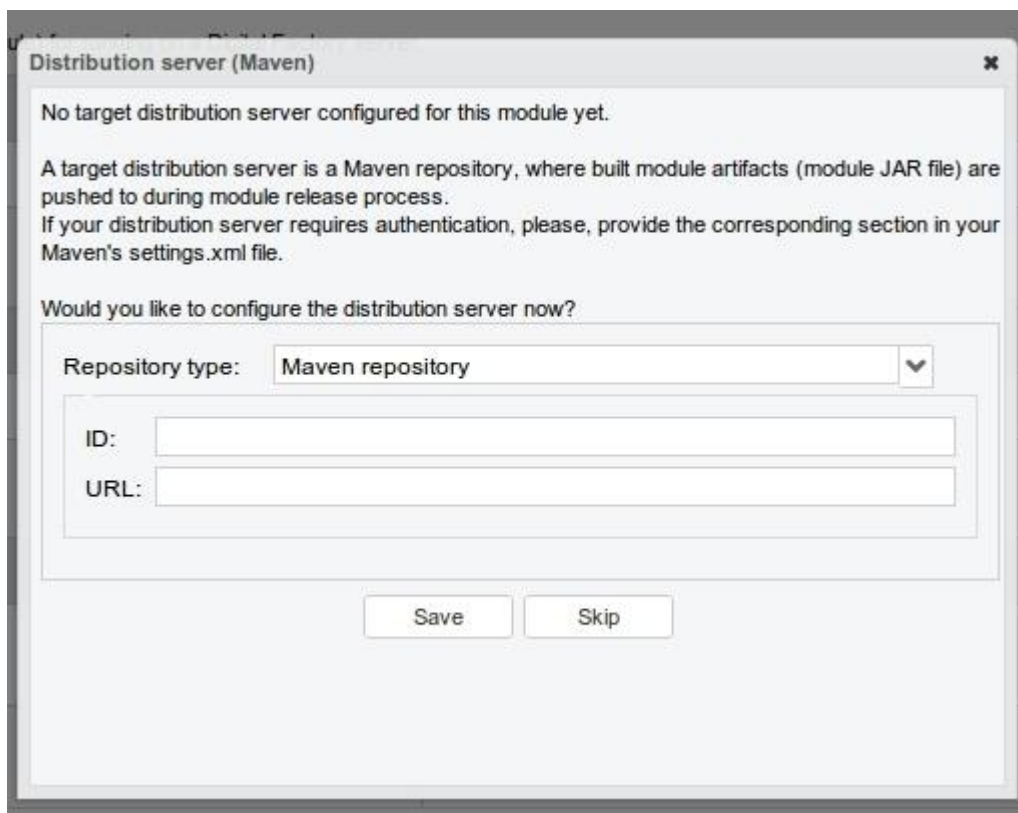
3.2 Digital Factory Studio

The Digital factory studio integrates a release process based on maven to ease the releasing of modules. This allows releasing and deploying on a maven repository and/or on app store. On the first release you have to choose your release process, this can be changed later by clicking on **Edit repository information** in the same menu.



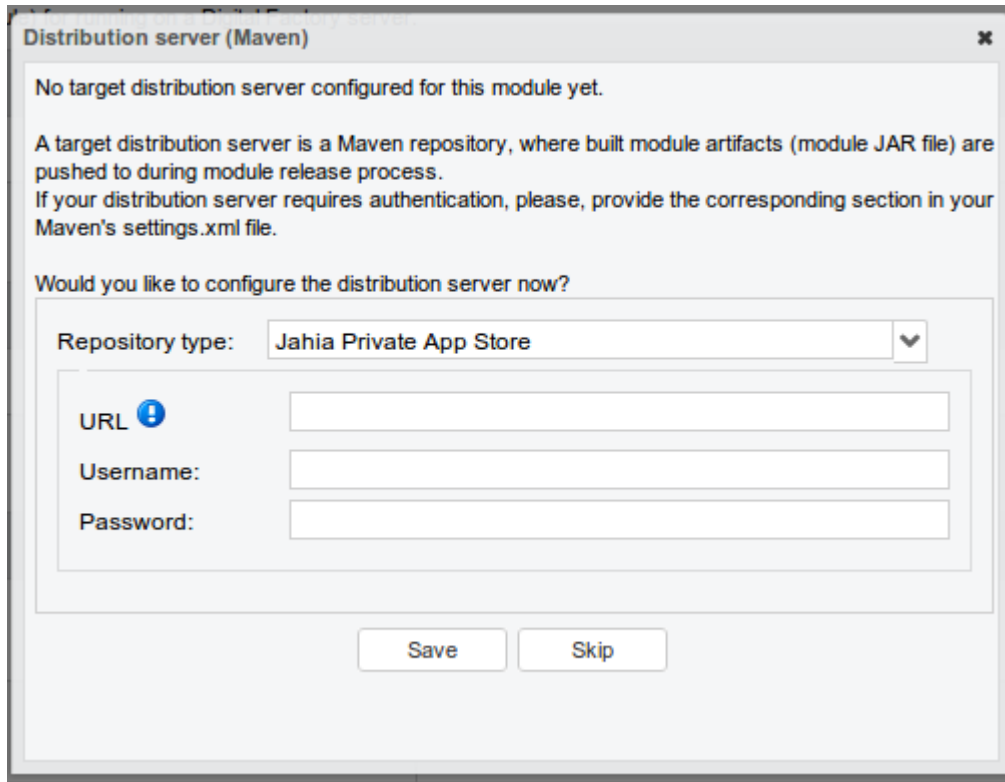
3.2.1 Select your repository type

3.2.1.1 Releasing on a maven repository




According to your maven settings, you have to define the maven repository ID and its URL before clicking on save.

3.2.1.2 Releasing on a Jahia Private App Store



Selecting Jahia Private App Store let you define the store url that you can find on the homepage of the store you've previously installed.

AppStore URL

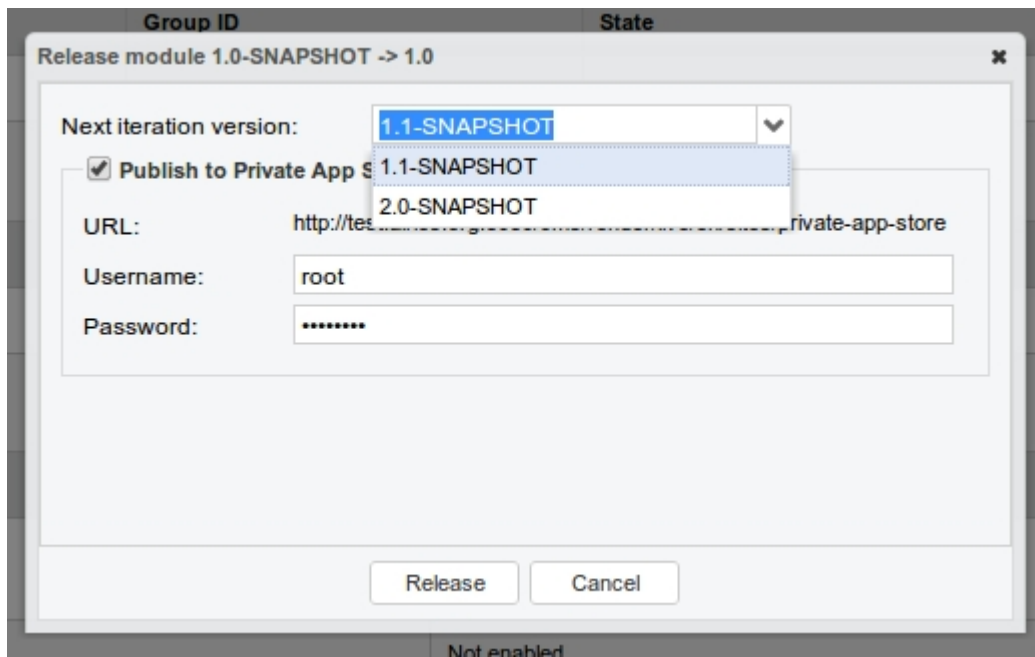
 

You have to use credentials with developer permissions, then click on save.

If you skip this step, the jar will be released but store locally.

3.2.2 Releasing the module

Once the repository type is chosen, you can release the module, depending of your configuration you will be able to send it to a store or deploy it on a maven repository.

A screenshot of a 'Release module' dialog box. The title bar says 'Release module 1.0-SNAPSHOT -> 1.0'. Inside, there's a 'Next iteration version:' dropdown menu with '1.1-SNAPSHOT' selected. Below it is a checkbox labeled 'Publish to Private App Store' which is checked. Under the checkbox, there are three input fields: 'URL:' with the value 'http://test.jahia.com/private-app-store', 'Username:' with the value 'root', and 'Password:' with a masked value '*****'. At the bottom, there are two buttons: 'Release' and 'Cancel'. The dialog box is set against a background that shows a table with columns 'Group ID' and 'State', and a row with the value 'Not enabled'.

You can select a major or minor release type, then click on **release** to perform the release

Note: if you have a scm set on your module, the maven plugin used to do the release will tag your module.

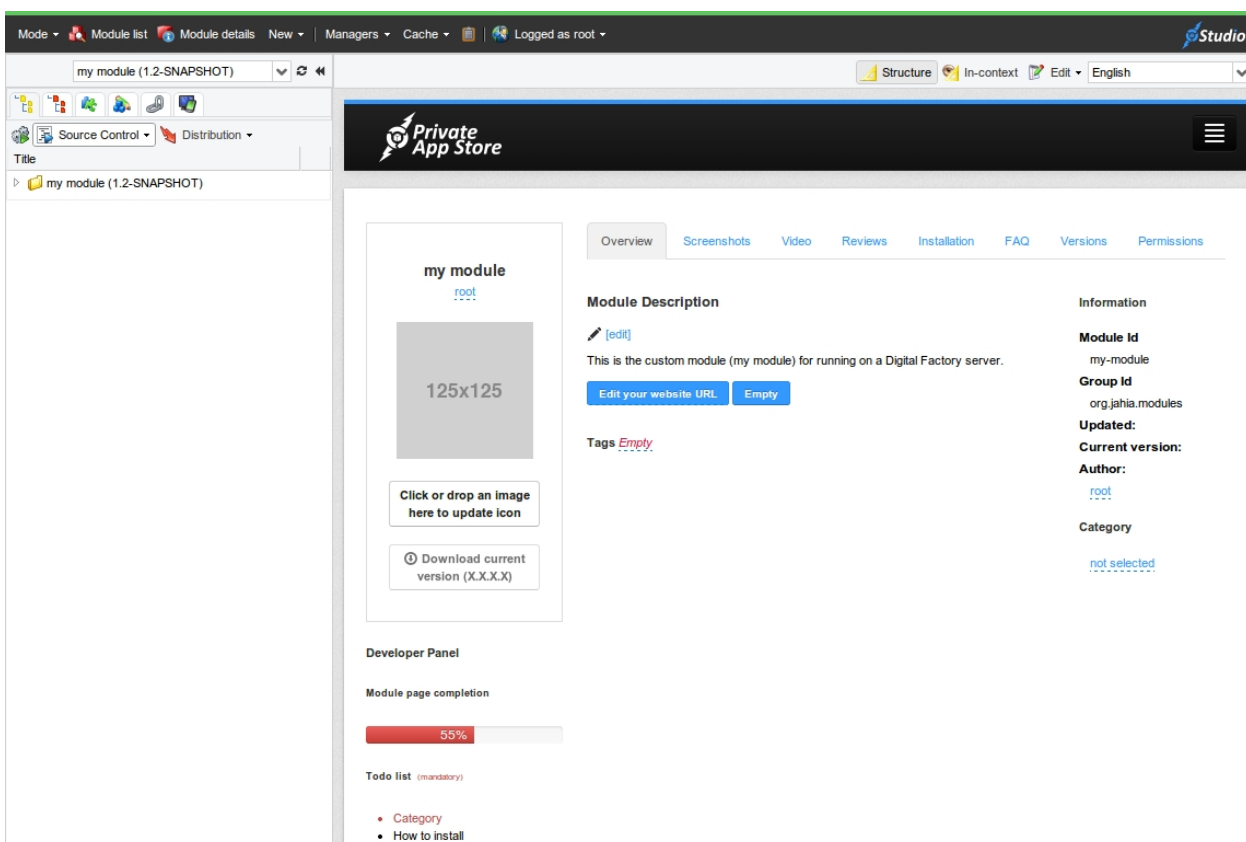
3.2.3 Finalize the release

3.2.3.1 Without a Private App Store



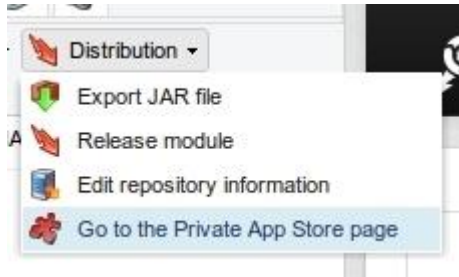
After releasing, if you didn't set store information, you will be able to download a copy of the released jar of the module.

3.2.3.2 With a Private App Store



You can edit within the digital factory studio, the store information of the module as if you were on the App Store web site.

Once a module is pushed to a store, you can access to its store page by clicking on **Go to the private App Store page**



4 Digital Factory administration

4.1.1 Define stores

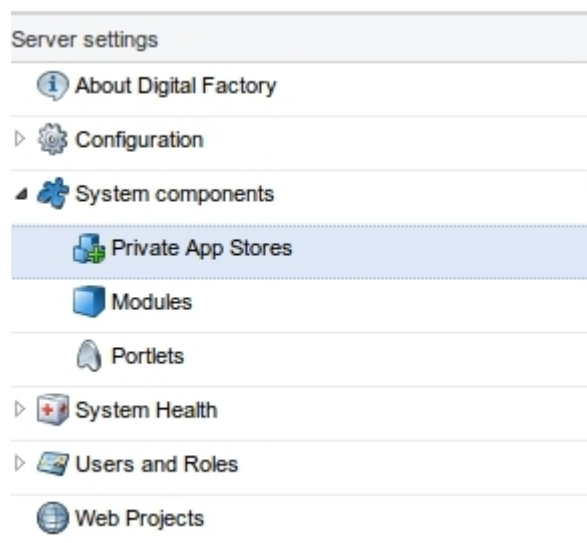
A Digital Factory instance can register an unlimited number of stores. The URL of a store, used to establish a direct connection between a Digital Factory instance and the Private App Store, is displayed on the home page of the web version of the Store. Click on the icon to copy this URL.

AppStore URL

<http://test.airies.org:8080/cms/ren>



Go to the Digital Factory Administration > Server Settings > System components > Private App Stores



The main panel displays the stores already declared and a form to register new ones.

Private App Stores

Private App Store URL	User	
http://store.jahia.com/cms/render/live/en/sites/private-app-store		edit ✕ Delete

Add Private App Store


Private App Store URL ⓘ

User

Password

✓ Save

✓ Reset

© Copyright 2002-2014  Jahia Solutions Group SA

You can set multiple store locations, for instance an internal store and several ones from your contractors.

Use the Private App Store reader credentials if they needed to access the store (the administrator of the store should give you that information). Leave those fields empty if the App Store has a public access.

4.1.2 Browse available modules

Once you've added one or several Private App Stores, the modules registered in those Stores will be accessible directly in the "Modules" page of the settings panel. You can then download and install those modules in one click, directly from Digital Factory GUI.

The list of modules coming from app stores is refreshed **once a day**.

Refresh can be forced by clicking on



4.1.3 Stay informed

When a new version of an already installed module is available, the download button becomes enabled again so you can install it.



Jahia Solutions Group SA

9 route des Jeunes,
CH-1227 Les acacias
Geneva, Switzerland

<http://www.jahia.com>

