General Installation Guide

Symfony 2.2

*Authors: Atabak Sahraei, Hans Becker, Manuel Schermuly*

Contents

[Symfony Configuration on MAC 2](#_Toc357254311)

[Symfony Configuration on Windows (xampp 1.8.1) 2](#_Toc357254312)

[Symfony Configuration (system independent) 2](#_Toc357254313)

[Installing Composer 3](#_Toc357254314)

[Installing the Friends of Symfony (FOS) User Bundle 3](#_Toc357254315)

[Configuration FOS 3](#_Toc357254316)

[Database Handling 6](#_Toc357254317)

[Installing the SonataDoctrineORMAdminBundle (optional) 6](#_Toc357254318)

# Symfony Configuration on MAC

1. set Environment Path to PHP Server
2. OSX: the path is set but to the standard Apple Web Server. Change it.
3. Example for mamp php5
   * Backup current php version by renaming it (optional)
   * $ sudo mv /usr/bin/php /usr/bin/php4
4. Create a symbolic link pointing to the PHP version supplied by MAMP
5. $ sudo ln -s /Applications/MAMP/bin/php5/bin/php /usr/bin/php
6. If you want use Apple Web Server, you must set the timezone constant
7. For the commandline, you have to change your directory to Symfony directory in htdocs

# Symfony Configuration on Windows (xampp 1.8.1)

Extract the Zip-Package to htdocs (Result: htdocs/Symfony)

Open localhost/Symfony/web/config.php -> 2 errors

1. Error: activate php\_intl.dll (<http://www.tiv.net/2010/12/php-intl-extension-windows-apache.html>)

All Files are included in xampp, you just have to activate them:

1. Uncomment the following line „extension=php\_intl.dll” in php.ini
2. The PATH environment variable has to point to xampp\php
3. Restart Apache-Server

2. Error: install PHP APC (<http://docs.moodle.org/20/en/Installing_APC_in_Windows#Download_the_correct_version>)

1. Download php\_apc.dll: <http://downloads.php.net/pierre/php_apc-3.1.10-5.4-vc9-x86.zip>
2. Extract php\_apc.dll to xampp\php\ext\ts (Thread Safe)
3. Add the following line to php.ini: extension=php\_apc.dll
4. Restart Apache-Server

# Symfony Configuration (system independent)

A default bundle, `AcmeDemoBundle`, shows you Symfony2 in action. After

playing with it, you can remove it by following these steps:

1. delete the `src/Acme` directory;
2. remove the routing entries referencing AcmeBundle in

`app/config/routing\_dev.yml`;

1. remove the AcmeBundle from the registered bundles in `app/AppKernel.php`;
2. remove the `web/bundles/acmedemo` directory;
3. remove the `security.providers`, `security.firewalls.login` and

`security.firewalls.secured\_area` entries in the `security.yml` file or

tweak the security configuration to fit your needs.

Configure the yml-file as followed:

# app/config/parameters.yml

parameters:

database\_driver: pdo\_mysql

database\_host: localhost

database\_port: '3306'

database\_name: symfony

database\_user: root

database\_password: root

database\_path: null

mailer\_transport: smtp

mailer\_host: smtp.domain.de

mailer\_user: user@domain.de

mailer\_password: passwort

mailer\_encryption: tls

locale: en

secret: something\_secret

# Installing Composer

For installing plugins you have to download the php Composer:

<http://getcomposer.org/>

Run this in your terminal to get the latest Composer version:

curl -sS https://getcomposer.org/installer | php

Or if you don't have curl (works on Windows):

php -r "eval('?>'.file\_get\_contents('https://getcomposer.org/installer'));"

# Installing the Friends of Symfony (FOS) User Bundle

Composer path

<https://packagist.org/packages/friendsofsymfony/user-bundle>

Instructions

<https://github.com/FriendsOfSymfony/FOSUserBundle/blob/master/Resources/doc/index.md>

Add FOSUserBundle in your composer.json:

{

"require": {

"friendsofsymfony/user-bundle": “\*"

}

}

Now tell composer to download the bundle by running the command:

$ php composer.phar update friendsofsymfony/user-bundle

# Configuration FOS

**Step 1**

Add the Following Command to app/AppKernel.php

$bundles = array(

...

new FOS\UserBundle\FOSUserBundle(),

);

**Step 2**

Add the Following Command to app/config/config.yml

#FOS

fos\_user:

db\_driver: orm

firewall\_name: main

user\_class: WAN\MedienFTPBundle\Entity\User

**Step 3**

Add the Following Command to app/config/security.yml

security:

encoders:

#FOS

FOS\UserBundle\Model\UserInterface: sha512

role\_hierarchy:

#FOS

ROLE\_ADMIN: ROLE\_USER

ROLE\_SUPER\_ADMIN: ROLE\_ADMIN

providers:

#FOS

fos\_userbundle:

id: fos\_user.user\_provider.username

firewalls:

#FOS

main:

pattern: ^/

form\_login:

provider: fos\_userbundle

csrf\_provider: form.csrf\_provider

logout: true

anonymous: true

#FOS

access\_control:

- { path: ^/login$, role: IS\_AUTHENTICATED\_ANONYMOUSLY }

- { path: ^/register, role: IS\_AUTHENTICATED\_ANONYMOUSLY }

- { path: ^/resetting, role: IS\_AUTHENTICATED\_ANONYMOUSLY }

- { path: ^/admin/, role: ROLE\_ADMIN }

**Step 4**

Add the Following Command to app/config/routing.yml

fos\_user\_security:

resource: "@FOSUserBundle/Resources/config/routing/security.xml"

fos\_user\_profile:

resource: "@FOSUserBundle/Resources/config/routing/profile.xml"

prefix: /profile

fos\_user\_register:

resource: "@FOSUserBundle/Resources/config/routing/registration.xml"

prefix: /register

fos\_user\_resetting:

resource: "@FOSUserBundle/Resources/config/routing/resetting.xml"

prefix: /resetting

fos\_user\_change\_password:

resource: "@FOSUserBundle/Resources/config/routing/change\_password.xml"

prefix: /profile

**Step 5**

Create your Bundle

php app/console generate:bundle --namespace=WAN/MedienFTPBundle --format=yml

**Step 6**

Create the User Class which you references in Step 2

In this case you must create the following file src/namespace/Entity/User.php

<?php

// src/Acme/UserBundle/Entity/User.php

namespace WAN\MedienFTPBundle\Entity;

use FOS\UserBundle\Entity\User as BaseUser;

use Doctrine\ORM\Mapping as ORM;

use Symfony\Component\Validator\Constraints as Assert;

/\*\*

\* @ORM\Entity

\* @ORM\Table(name="medienftp\_user")

\*/

class User extends BaseUser {

//Start ID

/\*\*

\* @ORM\Id

\* @ORM\Column(type="integer")

\* @ORM\GeneratedValue(strategy="AUTO")

\*/

protected $id;

//END ID

//Start group as String

/\*\*

\* @ORM\Column(type="string", length=255)

\* @Assert\NotBlank(message="Please enter your name.", groups={"Registration", "Profile"})

\* @Assert\MinLength(limit="3", message="Name too short”, groups={"Registration", "Profile"})

\* @Assert\MaxLength(limit="255", message="name too long", groups={"Registration", "Profile"})

\*/

protected $group;

//END group

public function \_\_construct() {

parent::\_\_construct();

// your own logic

}

}

**Step 7**

Execute the following command to refresh your Database schema

php app/console doctrine:schema:update –force

**Step 8**

You now can login at <http://localhost/Symfony/web/app_dev.php/login>

You now Regist at <http://localhost/Symfony/web/app_dev.php/register>

You now can logout at <http://localhost/Symfony/web/app_dev.php/logout>

# Database Handling

To Create, Update, Delete Database Items

$ php app/console doctrine:database:drop --force

$ php app/console doctrine:database:create

$ php app/console doctrine:schema:update --force

# Installing the SonataDoctrineORMAdminBundle (optional)

Composer path

<https://packagist.org/packages/sonata-project/doctrine-orm-admin-bundle>

Instructions

<http://sonata-project.org/bundles/admin/2-1/doc/index.html>

Add the Bundle in your composer.json:

{

"require": {

"sonata-project/doctrine-orm-admin-bundle": "2.1.\*@dev"

}

}