

Workshop: FOSCommentBundle

Objectif :

Le but de ce Workshop est l'installation de FOSCommentBundle destiné à la gestion des commentaires d'une application web.

Ce Workshop est basé sur ce lien :

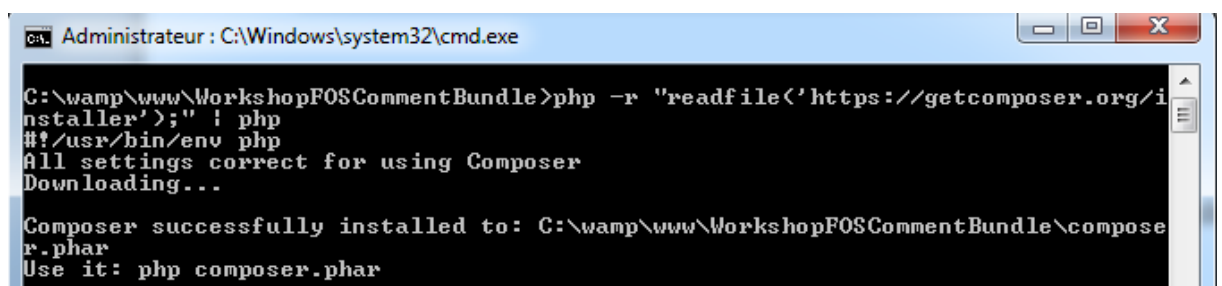
<https://github.com/FriendsOfSymfony/FOSCommentBundle>

Etapes :

1. Créer un projet Symfony WorkshopFOSCommentBundle.
2. Créer un Bundle CommentBundle sous le namespace Esprit.
3. Ajouter le fichier composer.phar :

Il faut tout d'abord ouvrir une fenêtre de commande et se placer à la racine du projet et copier la commande suivante :

```
php -r "readfile('https://getcomposer.org/installer');" | php
```



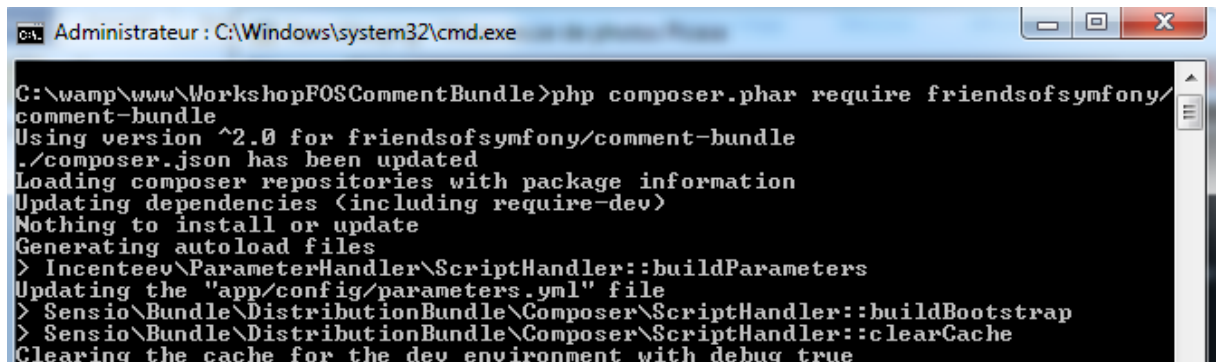
```
Administrateur : C:\Windows\system32\cmd.exe

C:\wamp\www\WorkshopFOSCommentBundle>php -r "readfile('https://getcomposer.org/installer');" | php
All settings correct for using Composer
Downloading...

Composer successfully installed to: C:\wamp\www\WorkshopFOSCommentBundle\composer.phar
Use it: php composer.phar
```

4. Pour installer FOSCommentBundle, il faut exécuter cette commande à la racine du projet:

\$ php composer.phar require friendsofsymfony/comment-bundle



```
Administrateur : C:\Windows\system32\cmd.exe
C:\wamp\www\WorkshopFOSCommentBundle>php composer.phar require friendsofsymfony/
comment-bundle
Using version ^2.0 for friendsofsymfony/comment-bundle
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Nothing to install or update
Generating autoload files
> Incenteev\ParameterHandler\ScriptHandler::buildParameters
Updating the "app/config/parameters.yml" file
> Sensio\Bundle\DistributionBundle\Composer\ScriptHandler::buildBootstrap
> Sensio\Bundle\DistributionBundle\Composer\ScriptHandler::clearCache
Clearing the cache for the dev environment with debug true
```

5. Activer les bundles nécessaires dans le fichier noyau AppKernel.php

```
<?php
// app/AppKernel.php

public function registerBundles()
{
    $bundles = array(
        // ...
        new FOS\RestBundle\FOSRestBundle(),
        new FOS\CommentBundle\FOSCommentBundle(),
        new JMS\SerializerBundle\JMSSerializerBundle($this),
    );
}
```

6. Décommenter le translator :

```
# app/config/config.yml

framework:
    translator: ~
```

7. Créer un nouveau dossier Entity sous le Bundle CommentBundle.
8. Ajouter l'entité Comment comme suit :

```

<?php
// src/Esprit/CommentBundle /Entity/Comment.php

namespace Esprit\CommentBundle \Entity;

use Doctrine\ORM\Mapping as ORM;
use FOS\CommentBundle\Entity\Comment as BaseComment;

/**
 * @ORM\Entity
 * @ORM\ChangeTrackingPolicy("DEFERRED_EXPLICIT")
 */
class Comment extends BaseComment
{
    /**
     * @ORM\Id
     * @ORM\Column(type="integer")
     * @ORM\GeneratedValue(strategy="AUTO")
     */
    protected $id;

    /**
     * Thread of this comment
     *
     * @var Thread
     * @ORM\ManyToOne(targetEntity=" Esprit\CommentBundle \Entity\Thread")
     */
    protected $thread;
}

```

9. Créer l'entité Thread comme suit :

```

<?php
// src/Esprit/CommentBundle /Entity/Thread.php

namespace Esprit\CommentBundle \Entity;

use Doctrine\ORM\Mapping as ORM;
use FOS\CommentBundle\Entity\Thread as BaseThread;

```

```

/**
 * @ORM\Entity
 * @ORM\ChangeTrackingPolicy("DEFERRED_EXPLICIT")
 */
class Thread extends BaseThread
{
    /**
     * @var string $id
     *
     * @ORM\Id
     * @ORM\Column(type="string")
     */
    protected $id;
}

```

10. Configurer l'application dans le fichier config.yml :

```

# app/config/config.yml

fos_comment:
    db_driver: orm
    class:
        model:
            comment: Esprit\CommentBundle\MyBundle\Entity\Comment
            thread: Esprit\CommentBundle \Entity\Thread

assetic:
    bundles: [ "FOSCommentBundle" ]

```

11. Importer le routage du bundle :

```

fos_comment_api:
    type: rest
    resource: "@FOSCommentBundle/Resources/config/routing.yml"
    prefix: /api
    defaults: { _format: html }

```

12. Pour inclure des commentaires sur une page, on utilise la référence javascript .

Le JavaScript asynchrone charge les commentaires après chargement de la page.

On a besoin d'ajouter cette ligne dans le fichier twig où on va avoir le commentaire :

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7/jquery.min.js"></script>
```

Et le code suivant à un endroit désiré dans le fichier twig pour charger les commentaires:

```
{% include 'FOSCommentBundle:Thread:async.html.twig' with {'id': 'foo'} %}
```

13. Pour mieux comprendre, on va créer une entité Article comme suit

```
k?php

namespace Esprit\CommentBundle\Entity;

use Doctrine\ORM\Mapping as ORM;
use FOS\CommentBundle\Entity\Comment as BaseComment;

/**
 * @ORM\Entity
 */
class Article extends BaseComment {

    /**
     * @ORM\Id
     * @ORM\Column(type="integer")
     * @ORM\GeneratedValue(strategy="AUTO")
     */
    protected $id;

    /**
     *
     * @ORM\Column(type="string",length=255)
     */
    protected $nom;

    public function getId() {
        return $this->id;
    }

    public function getNom() {
        return $this->nom;
    }

    public function setId($id) {
        $this->id = $id;
    }

    public function setNom($nom) {
        $this->nom = $nom;
    }

}
```

14. Définir la route :

```
esprit_comment_article:
    path:      /affiche
    defaults: { _controller: EspritCommentBundle:Article:affiche }
```

15. Configurer la base des données :

```
parameters:
    database_driver: pdo_mysql
    database_host: 127.0.0.1
    database_port: null
    database_name: foscomment
    database_user: root
    database_password: null
    mailer_transport: smtp
    mailer_host: 127.0.0.1
    mailer_user: null
    mailer_password: null
    locale: en
    secret: ThisTokenIsNotSoSecretChangeIt
    debug_toolbar: true
    debug_redirects: false
    use_assetic_controller: true
```

NB : N'oublier pas d'exécuter les commandes « Create database » ensuite « create schema » afin de créer notre base des données.

16. Ajouter l'action affiche dans le fichier ArticleController sous le répertoire Esprit/CommentBundle/Controller :

```
<?php

namespace Esprit\CommentBundle\Controller;

use Symfony\Bundle\FrameworkBundle\Controller\Controller;

class ArticleController extends Controller {

    public function afficheAction() {
        $em = $this->getDoctrine()->getManager();
        $articles = $em->getRepository("EspritCommentBundle:Article")->findAll();
        return $this->render("EspritCommentBundle:Article:affiche.html.twig", array("articles" => $articles));
    }

}
```

17. Créer une page affiche.html.twig dans un nouveau dossier Article sous le répertoire Esprit/CommentBundle/Resources/views :

```

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7/jquery.min.js"></script>

<h1>Liste des articles</h1>

<table border="1">
  <tr>
    <th>Id</th>
    <th>Nom</th>
  </tr>
  {% for i in articles %}
    <tr>
      <td>{{i.id}}</td>
      <td>{{i.nom}}</td>
    </tr>
  {% endfor %}
</table>

{% include 'FOSCommentBundle:Thread:async.html.twig' with {'id': 'foo'} %}

```

18. Visualiser le résultat:

Liste des articles

Id	Nom
1	Robe

Add New Comment

By Anonymous - November 19, 2015 10:49

Je veux passer une commande