



Ministerul Educației, Culturii și Cercetării a Republicii Moldova

Universitatea de Stat a Moldovei

Facultatea Matematică și Informatică

Raport

Lucrare de laborator Nr.3

la disciplina Framework-uri pentru dezvoltarea de aplicații web

Tema: "Servicii"

A efectuat : Pistol Adrian

A verificat : Bodrug Svetlana

Chișinău 2020

Cuprins

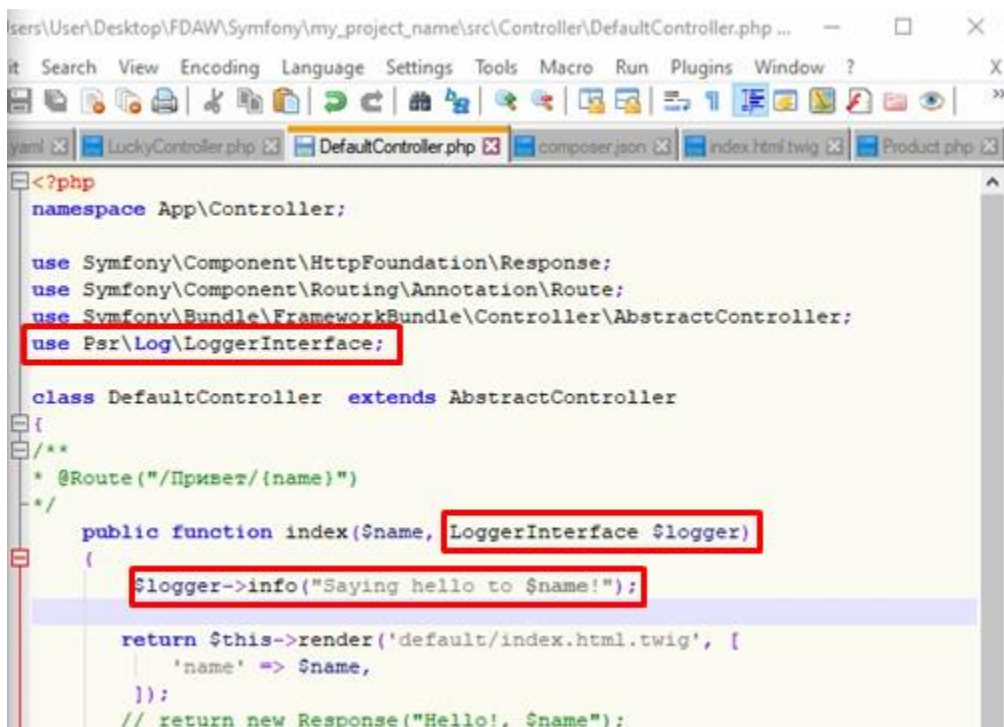
Adăugarea jurnalului	3
Servicii și automatizare	4
Crearea serviciului.....	4
Extinderea și autoconfigurarea Twig.....	6
Rezultat:.....	6

Adăugarea jurnalului

Noua aplicație Symfony este microscopică: constă în esență doar dintr-un sistem de rutare și un controler. Dar, datorită Flex, instalarea de noi funcții este ușoară.

```
C:\Users\User\Desktop\FDAW\Symfony\my_project_name> composer require logger
Using version ^3.6 for symfony/monolog-bundle
./composer.json has been updated
Running composer update symfony/monolog-bundle
Loading composer repositories with package information
Updating dependencies
Nothing to modify in lock file
Writing lock file
Installing dependencies from lock file (including require-dev)
Nothing to install, update or remove
Generating optimized autoload files
```

Aceasta instalează și configurează (prin rețetă) puternica bibliotecă Monolog. Pentru a utiliza loggerul într-un controler, adaugă un argument nou, tastat de LoggerInterface:



```
<?php
namespace App\Controller;

use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\Routing\Annotation\Route;
use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
use Psr\Log\LoggerInterface;

class DefaultController extends AbstractController
{
    /**
     * @Route("/Привет/{name}")
     */
    public function index($name, LoggerInterface $logger)
    {
        $logger->info("Saying hello to $name!");

        return $this->render('default/index.html.twig', [
            'name' => $name,
        ]);
        // return new Response("Hello!, $name");
    }
}
```

```
160 [2020-11-05T20:59:39.573405+00:00] php.DEBUG: Warning: include(C:\Users\User\
161 [2020-11-05T20:59:39.575727+00:00] php.DEBUG: Warning: include(): Failed oper
162 [2020-11-05T21:02:49.905521+00:00] request.INFO: Matched route "app_default_
163 [2020-11-05T21:02:49.920077+00:00] app.INFO: Saying hello to Symfony! [] []
164 [2020-11-05T21:02:49.931884+00:00] php.DEBUG: User Warning: Configure the "cu
165 [2020-11-05T21:02:49.932048+00:00] php.DEBUG: User Notice: Upgrade the curl e
```

Servicii și automatizare

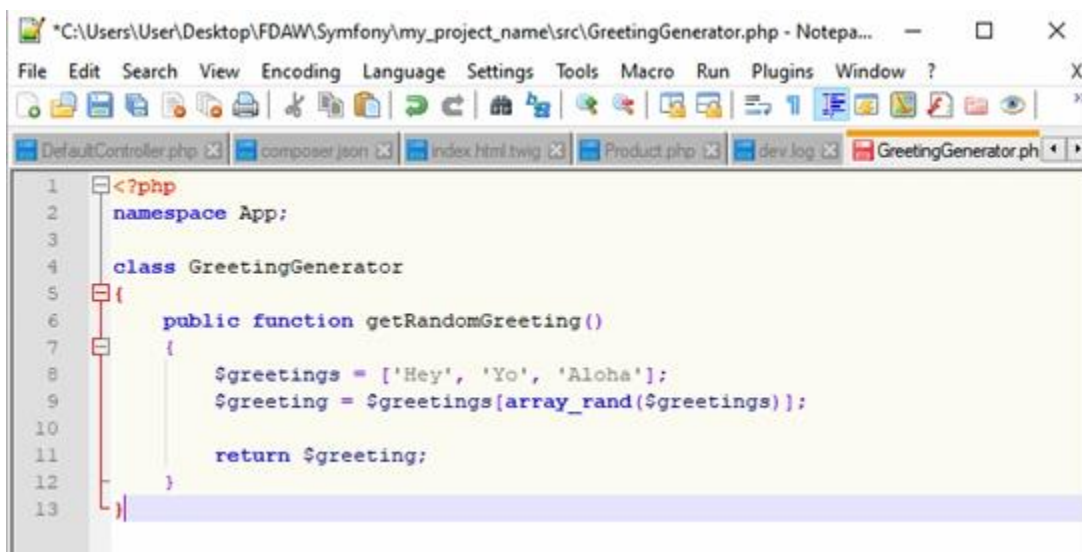
Pot cere de la Symfony să-mi trimită un serviciu folosind tastarea. Cu ajutorul comenzii de mai jos voi afla cumpot fi utilizate alte clase sau interfețe:

```
C:\Users\User\Desktop\FDAW\Symfony\my_project_name> php bin/console debug:autowiring
```

```
TranslationReader reads translation messages from translation files.  
Symfony\Component\Translation\Reader\TranslationReaderInterface (translation.reader)  
  
TranslationWriter writes translation messages.  
Symfony\Component\Translation\Writer\TranslationWriterInterface (translation.writer)  
  
Validates PHP values against constraints.  
Symfony\Component\Validator\Validator\ValidatorInterface (debug.validator)  
  
Covers most simple to advanced caching needs.  
Symfony\Contracts\Cache\CacheInterface (cache.app)  
  
Allows invalidating cached items using tags.  
Symfony\Contracts\Cache\TagAwareCacheInterface (cache.app.taggable)
```

Crearea serviciului

Pentru a menține codul organizat, pot să-mi creez propriile servicii! Dacă doresc să generez o felicitare aleatorie (de exemplu, „Bună ziua”, „Yo” etc.). În loc să introduc acest cod direct în controler, creez o nouă clasă:



```
*C:\Users\User\Desktop\FDAW\Symfony\my_project_name\src\GreetingGenerator.php - Notepa...  
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?  
DefaultController.php composer.json index.html.twig Product.php dev.log GreetingGenerator.ph  
1 <?php  
2 namespace App;  
3  
4 class GreetingGenerator  
5 {  
6     public function getRandomGreeting()  
7     {  
8         $greetings = ['Hey', 'Yo', 'Aloha'];  
9         $greeting = $greetings[array_rand($greetings)];  
10  
11         return $greeting;  
12     }  
13 }
```

Pot utiliza acest lucru în controler imediat:

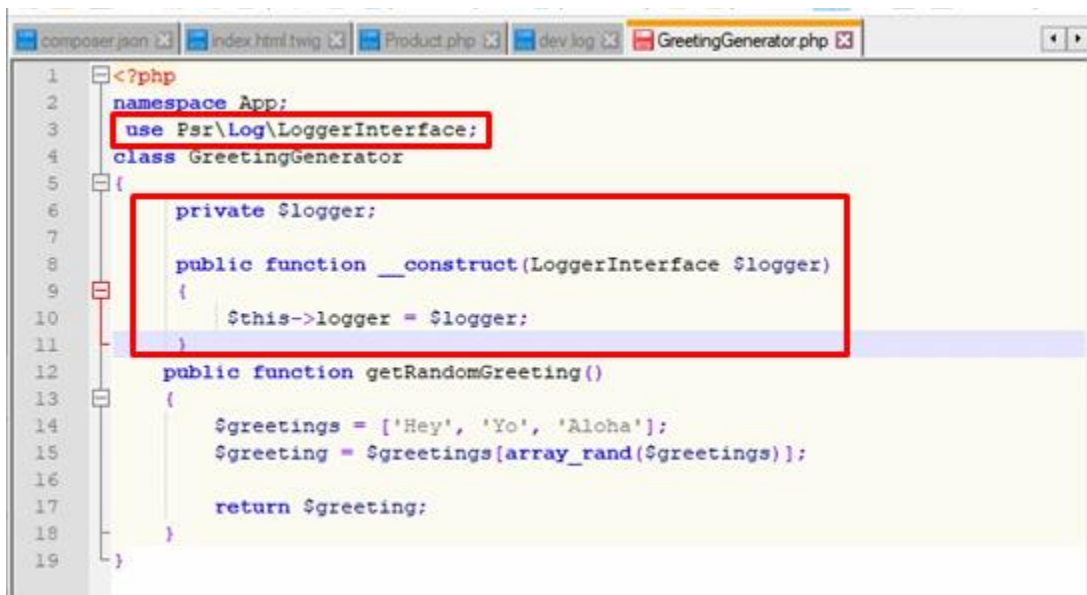


```
use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\Routing\Annotation\Route;
use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
use Psr\Log\LoggerInterface;
use App\GreetingGenerator;

class DefaultController extends AbstractController
{
    /**
     * @Route("/Привет/{name}")
     */
    public function index($name, LoggerInterface $logger, GreetingGenerator $generator)
    {
        $greeting = $generator->getRandomGreeting();
        $logger->info("Saying hello to $name!");

        return $this->render('default/index.html.twig', [
            'name' => $name,
        ]);
        // return new Response("Hello!, $name");
    }
}
```

Symfony va instanția GreetingGenerator automat și îl va transmite ca argument. Pot utiliza cablarea automată în cadrul unui serviciu pentru a accesa alte servicii. Singura diferență este că acest lucru se face în constructor:



```
<?php
namespace App;
use Psr\Log\LoggerInterface;
class GreetingGenerator
{
    private $logger;

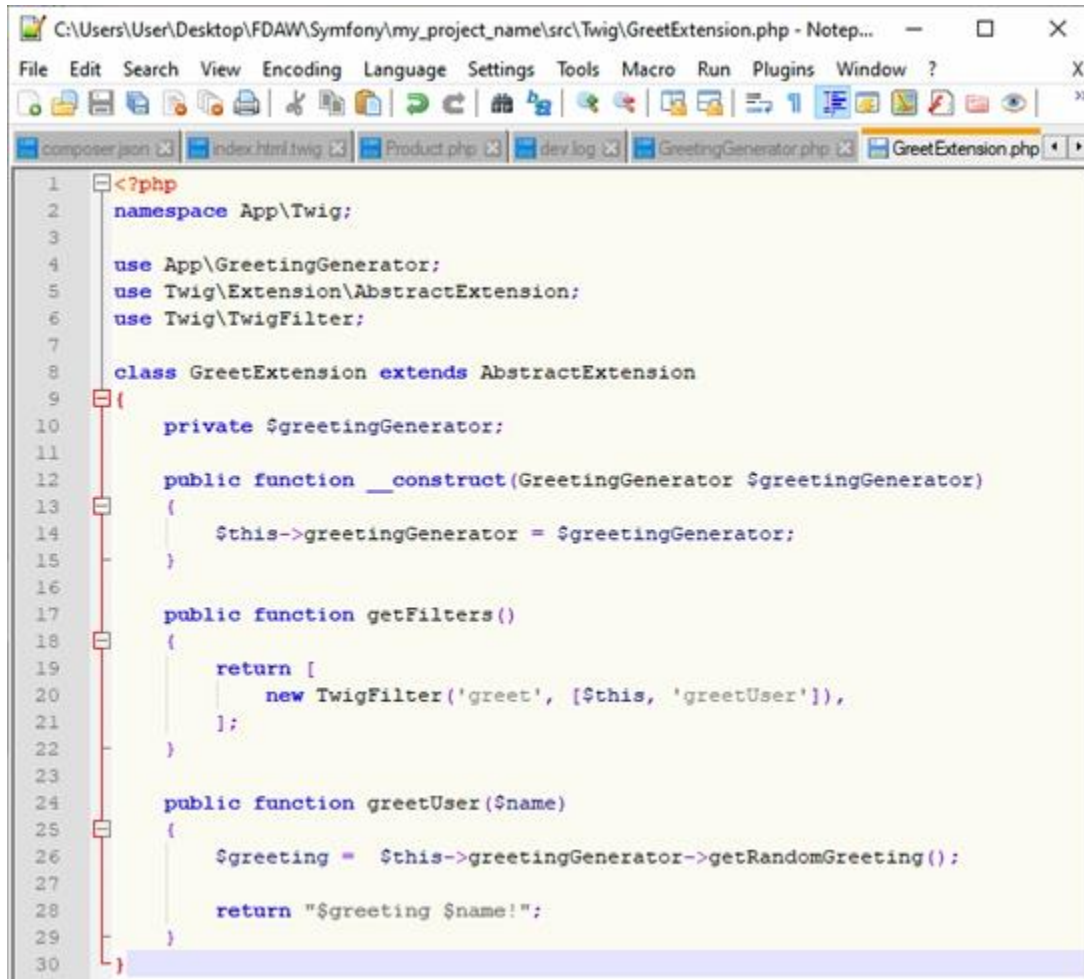
    public function __construct(LoggerInterface $logger)
    {
        $this->logger = $logger;
    }

    public function getRandomGreeting()
    {
        $greetings = ['Hey', 'Yo', 'Aloha'];
        $greeting = $greetings[array_rand($greetings)];

        return $greeting;
    }
}
```

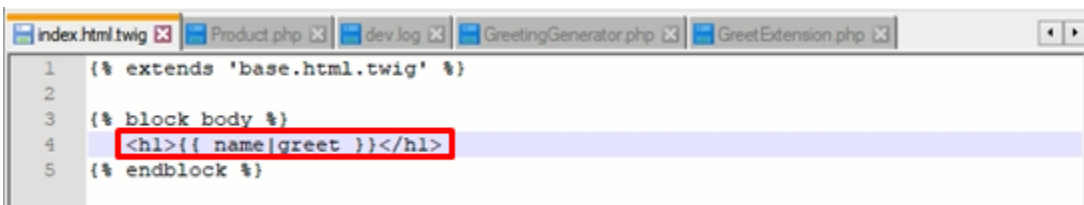
Extinderea și autoconfigurarea Twig

Prin gestionarea serviciilor Symfony, pot extinde Symfony în multe moduri, cum ar fi crearea unui abonat la eveniment sau a unui alegător de securitate pentru reguli complexe de autorizare. Să adaug un nou filtru la Twig numit greet, creând o clasă care extinde AbstractExtension:



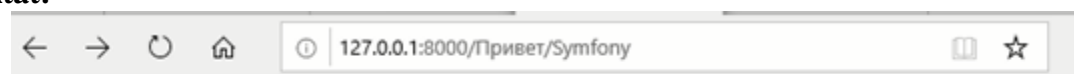
```
1 <?php
2 namespace App\Twig;
3
4 use App\GreetingGenerator;
5 use Twig\Extension\AbstractExtension;
6 use Twig\TwigFilter;
7
8 class GreetExtension extends AbstractExtension
9 {
10     private $greetingGenerator;
11
12     public function __construct(GreetingGenerator $greetingGenerator)
13     {
14         $this->greetingGenerator = $greetingGenerator;
15     }
16
17     public function getFilters()
18     {
19         return [
20             new TwigFilter('greet', [$this, 'greetUser']),
21         ];
22     }
23
24     public function greetUser($name)
25     {
26         $greeting = $this->greetingGenerator->getRandomGreeting();
27
28         return "$greeting $name!";
29     }
30 }
```

După crearea unui singur fișier, îl pot utiliza imediat:



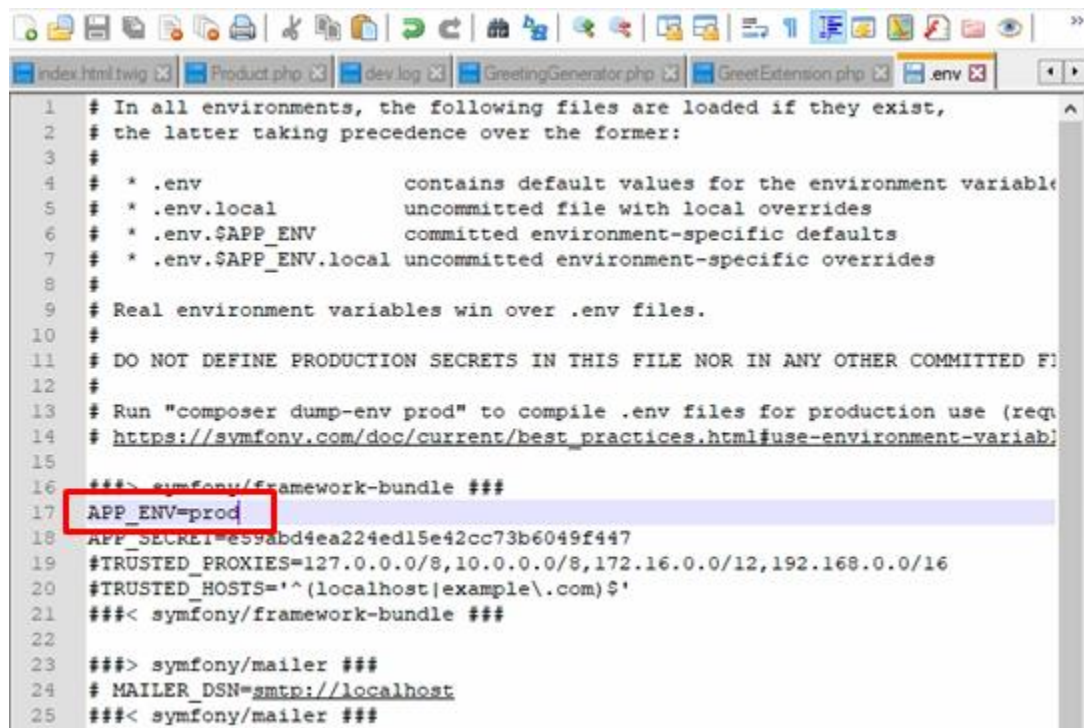
```
1 {% extends 'base.html.twig' %}
2
3 {% block body %}
4     <h1>{{ name|greet }}</h1>
5 {% endblock %}
```

Rezultat:



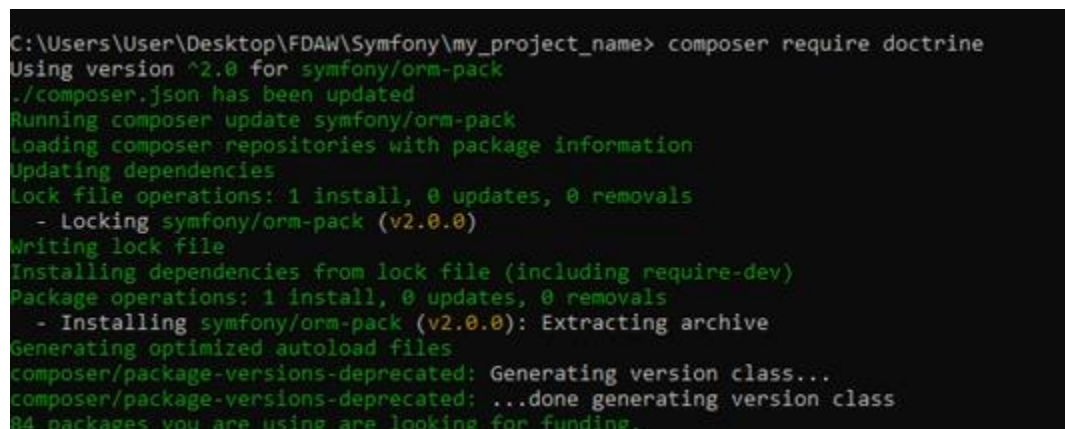
Yo Symfony!

Cum pot schimba configurația? Schimba variabila de mediu APP_ENV din dev în prod:



```
1 # In all environments, the following files are loaded if they exist,
2 # the latter taking precedence over the former:
3 #
4 # * .env             contains default values for the environment variables
5 # * .env.local       uncommitted file with local overrides
6 # * .env.$APP_ENV    committed environment-specific defaults
7 # * .env.$APP_ENV.local uncommitted environment-specific overrides
8 #
9 # Real environment variables win over .env files.
10 #
11 # DO NOT DEFINE PRODUCTION SECRETS IN THIS FILE NOR IN ANY OTHER COMMITTED FILE
12 #
13 # Run "composer dump-env prod" to compile .env files for production use (required for
14 # https://symfony.com/doc/current/best_practices.html#use-environment-variables)
15 #
16 ###> symfony/framework-bundle ###
17 APP_ENV=prod
18 APP_SECRET=essabd4ea224ed15e42cc73b6049f447
19 #TRUSTED_PROXIES=127.0.0.0/8,10.0.0.0/8,172.16.0.0/12,192.168.0.0/16
20 #TRUSTED_HOSTS='^(localhost|example\.com)$'
21 ###< symfony/framework-bundle ###
22
23 ###> symfony/mailer ###
24 # MAILER_DSN=smtp://localhost
25 ###< symfony/mailer ###
```

Pe măsură ce aplicația crește, se adaugă mai multe configurații, după cum este necesar. Instalez Doctrine ORM:



```
C:\Users\User\Desktop\FDAW\Symfony\my_project_name> composer require doctrine
Using version ^2.0 for symfony/orm-pack
./composer.json has been updated
Running composer update symfony/orm-pack
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
- Locking symfony/orm-pack (v2.0.0)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
- Installing symfony/orm-pack (v2.0.0): Extracting archive
Generating optimized autoload files
composer/package-versions-deprecated: Generating version class...
composer/package-versions-deprecated: ...done generating version class
04 packages you are using are looking for funding.
```