

# DEV-WEB.IO

Un peu de code, beaucoup de Symfony

CONTACT

## [SYMFONY 4] UTILISER VUE.JS

Dans cet article nous allons nous attaquer à la mise en place de vue.js avec Symfony 4.

[Vue.js](#) est un framework javascript que j'apprécie particulièrement pour sa simplicité.

Avant de commencer assurez-vous d'avoir bien [nodeJS](#) sur votre poste.

On va tout d'abord créer un nouveau projet symfony :

```
1 composer create-project symfony/skeleton symfony-vue
```

Puis ajouter quelques librairies :

```
1 composer require annotations twig encore asset
2 composer require server --dev
```

Et enfin installer vue.js :

```
1 npm install --dev vue vue-loader vue-template-compiler
```

Afin d'initialiser le projet, il suffit de faire :

```
1 npm install
```

Ces commandes vont créer les répertoires assets, node\_module ainsi que le fichier webpack.config.js

C'est ce dernier que nous allons tout de suite modifier :

```
1 var Encore = require('@symfony/webpack-encore');
2
3 Encore
4   .setOutputPath('public/build/')
5   .setPublicPath('/build')
6   .cleanupOutputBeforeBuild()
7   .enableSourceMaps(!Encore.isProduction())
8   .enableVersioning(Encore.isProduction())
9   .addEntry('app', './assets/js/main.js')
10  .enableVueLoader()
11 ;
```

```
12
13 module.exports = Encore.getWebpackConfig();
```

Et il faut enfin donner le chemin à symfony du manifest.json afin qu'il trouve les différents assets (le manifest.json sera créé au moment du build)

```
1 # config/packages/framework.yaml
2 assets:
3     json_manifest_path: '%kernel.project_dir%/public/build/manifest.json'
```

Voilà pour l'installation, on va maintenant passer aux choses sérieuses en créant nos pages.

On va d'abord créer notre controller, rien de compliqué ici :

```
1 <?php
2 // src/Controller/DefaultController.php
3
4 namespace App\Controller;
5
6 use Symfony\Component\HttpFoundation\Response;
7 use Symfony\Component\Routing\Annotation\Route;
8 use Symfony\Bundle\FrameworkBundle\Controller\Controller;
9
10 class DefaultController extends Controller
11 {
12     /**
13      * @Route("/")
14      */
15     public function default()
16     {
17         return $this->render('base.html.twig');
18     }
19 }
```

Puis le fichier twig templates/base.html.twig :

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <meta charset="UTF-8">
5         <title>{% block title %}Welcome!{% endblock %}</title>
6         {% block stylesheets %}{% endblock %}
7     </head>
8     <body>
9         {% block body %}
10         <div id="app">Oops... Problème</div>
11         {% endblock %}
12
13         {% block javascripts %}
14         <script src="{{ asset('build/app.js') }}"></script>
15         {% endblock %}
16     </body>
17 </html>
```

Le composant vue.js dans assets/js/App.vue :

```
1 <template>
2   <div id="app">
3     <hello></hello>
4   </div>
5 </template>
6
7 <script>
8   import Hello from './components/Hello'
9
10   class TestClassSyntax {
11
```

```
12 }
13
14 export default {
15   name: 'app',
16   components: {
17     Hello
18   }
19 }
20 </script>
```

Le fichier main de vue dans assets/js/main.js :

```
1 import Vue from 'vue'
2 import App from './App'
3
4 new Vue({
5   el: '#app',
6   template: '<App/>',
7   components: { App }
8 })
```

Et enfin le fichier assets/js/components/Hello.vue :

```
1 <template>
2   <div class="hello">
3     <h1>{{ msg }}</h1>
4   </div>
5 </template>
6
7 <script>
8 export default {
9   name: 'hello',
10  data () {
11    return {
12      msg: 'Welcome to Your Vue.js App'
13    }
14  }
15 }
16 </script>
```

Si vous lancer votre serveur à ce moment :

```
1 php bin/console server:run
```

vous allez avoir une erreur 404 vous disant qu'il ne trouve pas le fichier build/app.js, c'est tout à faire normal puisqu'il faut que l'on lance la génération de ce fichier en faisant :

```
1 ./node_modules/.bin/encore dev
```

Si tout se passe bien, en allant sur <http://127.0.0.1:8000> vous devez avoir un « Welcome to Your Vue.js App » s'afficher, et un « Oups... Problème » si il y a un... problème.

Petite astuce si voulez voir vos modifs en direct sans à chaque fois refaire une génération, à la place de la commande ci-dessus il faut faire :

```
1 ./node_modules/.bin/encore dev-server --hot
```

Vous pouvez retrouver les sources ici : <https://github.com/gponty/symfony-vue>

Publié dans [Symfony](#) le 11 janvier 2018. [17 commentaires](#)

## 17 COMMENTAIRES



**selim**

28 février 2018 à 1 h 02 min

bonjour j'arrive a cette etape mais il y a des choses que je comprend pas dans ce chapitre :  
la premier est pourquoi on crée un nouveau projet et es que il est en relation avec notre ancien projet ou non ?

la deuxième : j'ai un problème lors de la création du projet ( je l'avait créer au sein de l'ancien projet

merci d'avance 😊

RÉPONDRE



**Guillaume**

28 février 2018 à 13 h 57 min

Bonjour,

Ce chapitre n'a pas de rapport avec les anciens posts, d'où la création d'un nouveau projet.

Il me faudrait ton erreur et ton code pour que je puisse t'aider.

RÉPONDRE



**Maria Baker**

31 juillet 2018 à 13 h 44 min

Merci mais, ça ne marche pas avec le tout dernier version du VueLoader

RÉPONDRE



**rolodub**

11 mars 2018 à 17 h 08 min

Bonjour,

Quand je suis la procédure, il n'y a pas de fichier build/manifest.json créé...

As-tu une idée?

Merci!

RÉPONDRE

---



**Guillaume**

12 mars 2018 à 13 h 58 min

Oui tu as raison, le fichier manifest n'est créé seulement au moment de la phase de build :

```
./node_modules/.bin/encore dev
```

Je mets à jour l'article.

RÉPONDRE

---



**AMANI**

26 mars 2018 à 4 h 00 min

Bonjour Guillaume,

Merci pour les posts particulièrement, je voudrais savoir si dans la suite de Symfony 4, il pourrait avoir une validation email par token.

Merci et bonne continuation

Cdt

RÉPONDRE

---



**Guillaume**

26 mars 2018 à 10 h 03 min

Bonjour,

Oui sans problème, par contre il faut que tu le développes toi-même.

RÉPONDRE

---



**Clément**

18 juin 2018 à 19 h 56 min

A la fin du tuto, j'ai cette erreur :

You may need an appropriate loader to handle this file type.

Comment ça se fait-il ?

Merci de votre réponse

RÉPONDRE



**BobK**

11 octobre 2018 à 15 h 46 min

Il faut utiliser une version précédente de vueLoader.

Dans package.json : « vue-loader »: « ^14.2.2 »

et relancer npm install

RÉPONDRE



**Mathieu**

26 juillet 2018 à 18 h 00 min

Attention avec vue-loader : avec la version 15 webpack ne fonctionne pas.

<https://github.com/symfony/webpack-encore/issues/311>

Pour ma part j'ai du installer une version spécifique de vue-loader.

RÉPONDRE



**youstra**

14 octobre 2018 à 21 h 48 min

Quelle version de vue-loader avez-vous installé ?

RÉPONDRE



**youstra**

14 octobre 2018 à 19 h 44 min

Salut!

Je suis ce tuto depuis un bon moment et j'avoue qu'il est très bien fait. Ça m'a beaucoup aidé.

Cependant je rencontre un tout petit problème; je n'arrive pas à exécuter la commande :

./node\_modules/.bin/encore dev. A chaque fois, : '.' n'est pas reconnu en tant que commande...

RÉPONDRE



**youstra**

14 octobre 2018 à 19 h 57 min

Super tuto!

Mais j'ai un soucis. La commande ./node\_modules/.bin/encore dev ne fonctionne pas.

J'ai le message suivant:

'.' n'est pas reconnu en tant que commande interne  
ou externe, un programme exécutable ou un fichier de commandes.

Et surtout un grand merci

RÉPONDRE



**bernis**

14 novembre 2018 à 19 h 37 min

bonsoir,  
la compilation ne sort pas d'erreur, mais à l'appel de la page

vue.esm.js:591 [Vue warn]: Failed to mount component: template or render function not defined.

found in

—>

j'essaye de comprendre Runtime + Compiler vs. Runtime-only  
mais de l'aide serait la bien venue

RÉPONDRE



**Guillaume**

15 novembre 2018 à 10 h 42 min

Arf, comme ça je ne vois pas, il faudrait que tu publies ton code sur Github

RÉPONDRE



**modogo2000**

11 janvier 2019 à 20 h 37 min

Bonjour

Pourquoi ne pas utiliser symfony comme une api ? Et garder la logique de vue.js

RÉPONDRE

---

Ping : [Just another PoC of Symfony on steroids : Api Platform, Vue.js, Mercure and Panther – Stefano Alletti](#)

---

## LAISSER UN COMMENTAIRE

Votre adresse de messagerie ne sera pas publiée. Les champs obligatoires sont indiqués avec \*

Commentaire

Nom \*

Adresse de messagerie \*

Site web

**LAISSER UN COMMENTAIRE**

Ce site utilise Akismet pour réduire les indésirables. En savoir plus sur comment les données de vos commentaires sont utilisées.

---

## ARTICLES RÉCENTS

[Switch user](#)

[Test de la distro Linux Elementary](#)

[Utiliser Symfony dans docker](#)

[Utiliser docker pour le développement web](#)

[Symfony 4 : Gestion utilisateurs sans FOSUserBundle v2018 : Chapitre 3](#)

---

## COMMENTAIRES RÉCENTS



[Arnaud Stellato dans Symfony 4 :  
Gestion utilisateurs sans FOSUser-  
Bundle v2018 : Chapitre 1](#)

[Origami dans Symfony 4 : Gestion uti-  
lisateurs sans FOSUserBundle v2018  
: Chapitre 1](#)

[Guillaume dans Symfony 4 : Gestion  
utilisateurs sans FOSUserBundle  
v2018 : Chapitre 2](#)

[Malik dans Symfony 4 : Gestion utili-  
sateurs sans FOSUserBundle v2018 :  
Chapitre 2](#)

[Flo dans Symfony 4 : Gestion utilisa-  
teurs sans FOSUserBundle v2018 :  
Chapitre 3](#)

---

PLEASE FOLLOW &  
LIKE US :)



---

## ARCHIVES

[mars 2019](#)

[février 2019](#)

[novembre 2018](#)

[octobre 2018](#)

[juillet 2018](#)

[janvier 2018](#)

[décembre 2017](#)

[octobre 2017](#)

[mai 2017](#)

[juin 2016](#)

---

## CATÉGORIES

[Docker](#)

[Non classé](#)

[Symfony](#)

---

## MÉTA

[Connexion](#)

[Flux RSS des articles](#)

[RSS des commentaires](#)

[Site de WordPress-FR](#)

FIÈREMENT PROPULSÉ PAR WORDPRESS · THÈME : SUITS PAR THEME WEAVER | TRADUCTION :  
WOLFORG.