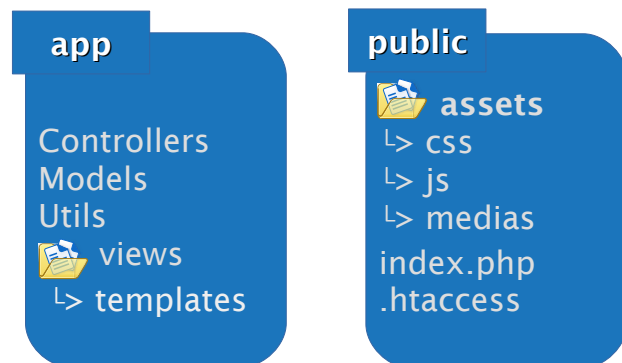


## S06-E01 / Architecture MVC - oFramework – schéma °2

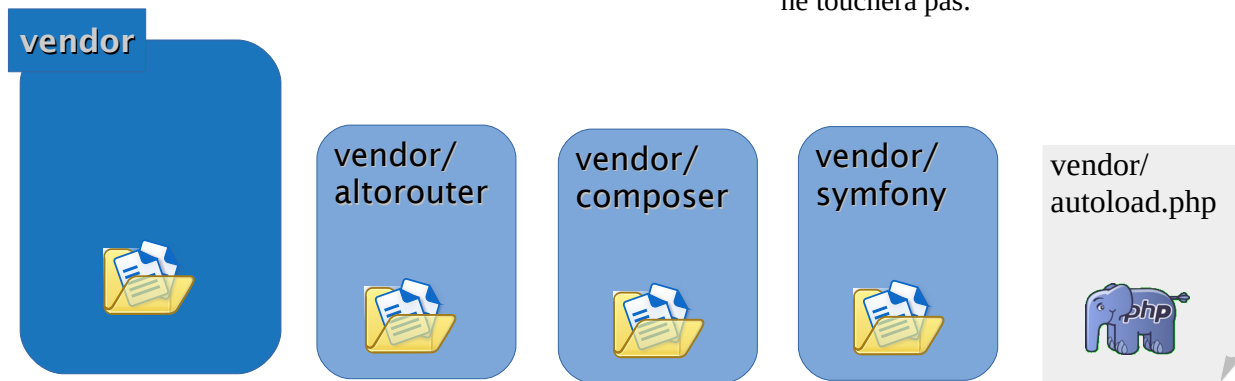
Installation de package ou dépendance : via **composer** dans le terminal OU le site [packagist.org](https://packagist.org)



Après avoir initialisé les dépendances sélectionnées avec *composer init* et répondu aux questions, faire un *composer install*.

Les dépendances s'installeront à la racine du dossier courant. Un dossier **vendor** sera créé et contiendra les dépendances téléchargées : *alotorouter*, *composer*, *var-dumper*, ainsi qu'un fichier *autoload.php*

A la racine du dossier il y aura également un fichier *.gitignore* contenant un dossier *vendor*. Un fichier *composer.json*, contenant *alotorouter* et *var-dumper*. Ainsi qu'un fichier *composer.lock* que l'on ne touchera pas.



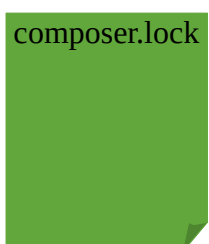
Dans ce fichier on met le dossier *vendor*, car cela évite de le versionner dans 'git'. On peut même le supprimer, car on peut en tapant dans le terminal *composer install*, réinstaller le dossier *vendor*, puisqu'il ne contient que les fichiers téléchargés. N'oubliez pas si vous supprimez le dossier *vendor* de le ré-installer pour pouvoir utiliser les dépendances.



En revanche le fichier *composer.json* doit-être versionné dans 'git'. C'est à partir de lui que l'on pourra réinstaller les dépendances dans le dossier. C'est la notice d'utilisation des dépendances.

AltoRouter est une classe de routage pour PHP 5.3+, fortement inspirée de *klein.php*.

- <https://packagist.org/>
- <http://alotorouter.com/>



Ne pas s'en occuper pour le moment.

Mécanisme *Symfony* pour explorer les variables PHP. Méthode native PHP *var\_dump()*, améliorée.

- <https://packagist.org/>

## # Installation d'un routeur via Composer

28:52

**Composer**, outils qui permet d'installer des dépendances ou paquets on appel cela un *gestionnaire de paquets*. Il permet entre autre d'installer **altorouter** ou **var\_dumper**.

Autre *gestionnaire de paquets* →

Atom	: apm (Atom Package Manager)
Linux Debian	: apt (Advanced Packaging Tool)
Linux Red-Hat	: rpm (Red Hat Package Manager)
Mac	: brew ( <a href="https://brew.sh/index_fr">https://brew.sh/index_fr</a> )
Node	: npm ( )
Node	: yarn ( <a href="https://yarnpkg.com/lang/en/">https://yarnpkg.com/lang/en/</a> )
Front-end	: bower ( <a href="https://bower.io/">https://bower.io/</a> )

Dans notre cas on a un gestionnaire spécifique pour PHP c'est **Composer**. → Doc composer <https://getcomposer.org/>

Ou **Composer** va t'il chercher les informations ?

↳ sur le site <https://packagist.org/>. Il référence tous les paquets que **Composer** est capable d'installer.



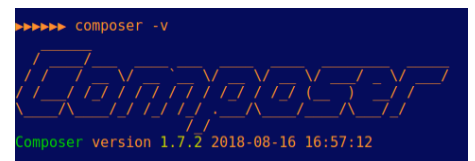
Il est possible d'installer les paquets à partir de **packagist**. De plus outre le fait de ne pas passer par la ligne de commande, il fourni une doc pour l'utilisation des paquets.

## # Initialisation des packages via le Terminal en ligne de commande

33:41

Comment faire pour disposer de **Composer** sachant qu'il est normalement présent.

↳ Dans le Terminal Linux taper **composer -v** pour connaître votre version actuellement installée.



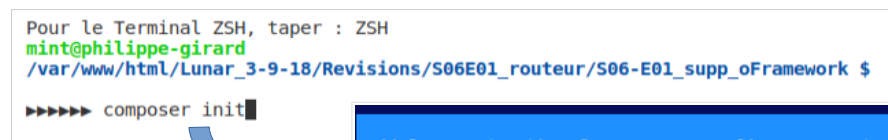
Se placer dans votre dossier ou se trouve votre site en construction avec les dossiers « app » et « public ».

↳ Ouvrir un **terminal** via **ctrl + alt + t** et aller dans votre dossier en ligne de commande.

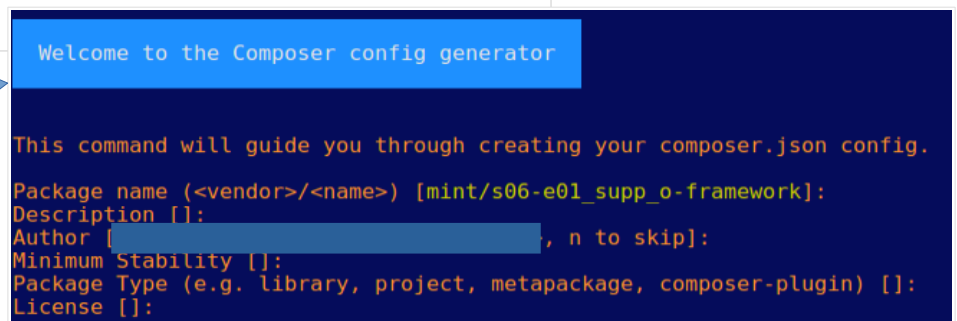
Ou utiliser **caja**, l'explorer de Linux est ouvrir le **terminal** via le clique droit sur le dossier concerné et choisir dans le menu déroulant la commande « Ouvrir dans le terminal ».

Une fois dans celui-ci, entrer la commande suivante :

↳ **composer init** // On va initialiser Composer dans le dossier courant.



Répondre aux questions qui concernent le nom, la description et l'auteur du ou des packages à installer, en appuyant sur la touche **Enter** pour passer à la question suivante. Sauf si vous voulez répondre aux questions.



### Définition des dépendances :

C'est maintenant que l'on va sélectionner nos packages.

- On a besoin d'un routeur pour définir les routes vers les pages de notre site, nous utiliseront **Altorouter**.
- Nous allons installer aussi un package pour le débogage, **var\_dumper**.

*Attention on prochains choix .*

```
Define your dependencies.
```

```
Would you like to define your dependencies (require) interactively [yes]? yes
```

Ici on me demande si je veux définir la dépendance à installer. Je tape **yes**.

```
Search for a package: altorouter
```

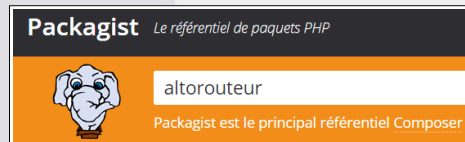
On me demande ensuite quel package faut-il chercher. Je tape **altorouter**

```
Found 15 packages matching altorouter
```

```
[0] altorouter/altorouter
[1] niahoo/altorouter
[2] benoclock/alto-dispatcher
[3] tekla/core-router
[4] law909/altorouter
[5] chiron/chiron-router
[6] law909/nineparams
[7] egx/rfw
[8] mikbox74/yii2-autorouter
[9] cupcakephp/cupcakephp
[10] eightyfive/laravel-autoroute
[11] smrtr/haltorouter
[12] netgusto/autoroute-bundle
[13] lesichkovm/php-autoroute
[14] smrtr/spawn-point
```

**Composer** a trouvé 15 dépendances portant ou ayant un lien avec **altorouter** via **packagist**.

Vous pouvez voir le détail des packages que vous voulez installer, via le site : <https://packagist.org>



```
Enter package # to add, or the complete package name if it is not listed: 0
```

Nous choisirons la première version d'**Altorouteur** c'est-à-dire la '0'. Taper **0**.

```
Enter the version constraint to require (or leave blank to use the latest version):
```

Je tape sur la touche **entrer** pour installer la dernière version d'**Altorouteur**.

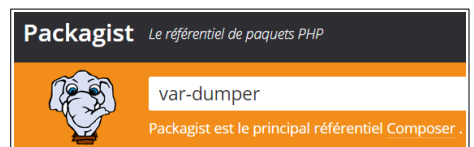
```
Using version ^1.2 for altorouter/altorouter
```

Dernière version la 1.2

```
Search for a package:
```

**Installons une seconde dépendance**, nous avons besoin de **var\_dumper** aussi pour aider à déboguer le code de façon plus graphique. Allons voir sur <https://packagist.org> ce qu'il y a.

```
Search for a package: var-dumper
```



```
Found 15 packages matching var-dumper
```

```
[0] symfony/var-dumper
[1] yiisoft/var-dumper
[2] rdehnhardt/var-dumper
[3] morrislaptop/var-dumper-with-context
[4] phalcon/dd
[5] hellonico/timber-dump-extension
[6] shyim/var-dumper
[7] omouren/external-var-dumper-bundle
[8] awesome/var-dumper
[9] keversc/var-trumper
[10] phalcongelist/dd Abandoned. Use phalcon/dd instead.
[11] ivoba/dump-die
[12] htmlburger/carbon-debug
[13] hellonico/twig-dump-extension
[14] setbased/abc-var-dumper-writer
```

```
Enter package # to add, or the complete package name if it is not listed: 0
```

```
Enter the version constraint to require (or leave blank to use the latest version):
```

Je tape sur la touche **entrer** pour installer la dernière version de **var-dumper**.

```
Using version ^4.3 for symfony/var-dumper
```

Dernière version la 4.3

**Dernière étape :**

```
Search for a package:
```

Pour passer la nouvelle recherche de dépendance appuyer sur la touche `entrer`.

Would you like to define your dev dependencies (require-dev) interactively [yes]? `no`

Ici on ne veut pas installer des dépendances de développement, donc on tape : `no`

```
{
  "name": "mint/s06-e01_supp_o-framework",
  "require": {
    "altorouter/altorouter": "^1.2",
    "symfony/var-dumper": "^4.3"
  },
  "authors": [
    {
      "name": "MartyRizey",
      "email": "martyrizey@gmail.com"
    }
  ]
}
```

Dépendances installées, **Altorouter** et **var-dumper**

Auteur de la création des dépendances.

Do you confirm generation [yes]? `yes`

Nous confirmons la génération en tapant : `yes`.

Would you like the vendor directory added to your .gitignore [yes]? `yes`

Oui nous voulons ajouter le dossier **vendor** au **.gitignore**: `yes`

Voilà l'installation des packages est terminée. Et dans notre arborescence voilà ce que cela donne.

La création d'un fichier **.gitignore** et **composer.json**

S06-E01_SUPP_OFFRAMEWORK	
▶ app	
▶ public	
◆ .gitignore	U
() composer.json	U
📄 README.md	M

Pour le Terminal ZSH, taper : ZSH

```
mint@philippe-girard
/var/www/html/Lunar_3-9-18/Revisions/S06E01_routeur/S06-E01_supp_oFramework $
>>>>> ls -la
total 32
drwxr-xr-x 5 mint mint 4096 juil.  6 13:08 .
drwxr-xr-x 3 mint mint 4096 juil.  5 18:16 ..
drwxr-xr-x 6 mint mint 4096 juil.  2 14:08 app
-rw-r--r-- 1 mint mint  272 juil.  6 13:03 composer.json
drwxr-xr-x 8 mint mint 4096 juil.  6 18:01 .git
-rw-r--r-- 1 mint mint   9 juil.  6 13:08 .gitignore
drwxr-xr-x 3 mint mint 4096 juil.  5 21:04 public
-rwxr-xr-x 1 mint mint 1337 juil.  6 09:13 README.md
```

Récap de la procédure 38:32

44:26

En confirmant la génération cela nous créer un dossier **vendor** dans le fichier **.gitignore** et créer le fichier **composer.json**.

```
1 {
2   "name": "mint/s06-e01_supp_o-framework",
3   "require": {
4     "altorouter/altorouter": "^1.2",
5     "symfony/var-dumper": "^4.3"
6   },
7   "authors": [
8     {
9       "name": "MartyRizey",
10      "email": "martyrizey@gmail.com"
11    }
12  ]
13 }
```

En faisant un **Composer init**, on a initialisé ou requis des packages en vu de leur installation dans le fichier **composer.json**.

Ici **Altorouter** version 1.2 et **var-dumper** version 4.3 seront requis en vu de leur installation.

l'auteur de cette initialisation est **MartyRizey** et son email est **martyrizey@gmail.com**.

La ligne 7 à 12 peuvent être supprimées. Ainsi que la ligne 2.

## # Installation des packages initialisés via le Terminal en ligne de commande

Dans le dossier de votre site, à l'endroit où se trouve le fichier `composer.json` entrer la commande suivante :

L> **composer install** // cela installera les dépendances initialisées .

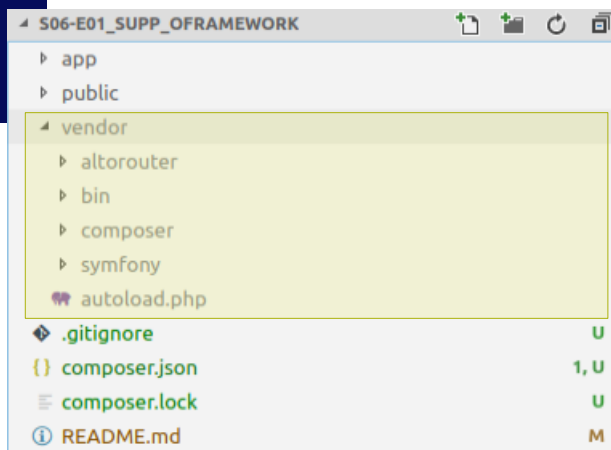
```
Pour le Terminal ZSH, taper : ZSH
mint@philippe-girard
/var/www/html/Lunar_3-9-18/Revisions/S06E01_routeur/S06-E01_supp_oFramework $
>>>>> composer install

Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 4 installs, 0 updates, 0 removals
- Installing altorouter/altorouter (v1.2.0): Loading from cache
- Installing symfony/polyfill-php72 (v1.11.0): Downloading (100%)
- Installing symfony/polyfill-mbstring (v1.11.0): Downloading (100%)
- Installing symfony/var-dumper (v4.3.2): Downloading (100%)
symfony/var-dumper suggests installing ext-intl (To show region name in time zone dump)
symfony/var-dumper suggests installing symfony/console (To use the ServerDumpCommand and/or the bin/
var-dump-server script)
Writing lock file
Generating autoload files
```

Qu'est-ce que **Composer** à fait ?

```
drwxr-xr-x 6 mint mint 4096 juil. 6 19:02 .
drwxr-xr-x 3 mint mint 4096 juil. 5 18:16 ..
drwxr-xr-x 6 mint mint 4096 juil. 2 14:08 app
-rw-r--r-- 1 mint mint 272 juil. 6 13:03 composer.json
-rw-r--r-- 1 mint mint 9123 juil. 6 19:02 composer.lock
drwxr-xr-x 8 mint mint 4096 juil. 6 18:01 .git
-rw-r--r-- 1 mint mint 9 juil. 6 13:08 .gitignore
drwxr-xr-x 3 mint mint 4096 juil. 5 21:04 public
-rwxr-xr-x 1 mint mint 1337 juil. 6 09:13 README.md
drwxr-xr-x 6 mint mint 4096 juil. 6 19:02 vendor
```

Il a créé un fichier **composer.lock** qui comporte des informations sur ce qu'il a fait. Pour le moment on ne s'en occupe pas.



Il a également créé un dossier **vendor** avec dedans des sous-dossiers, *altorouter*, *bin*, *composer*, *symfony*, *autoload.php* . Pareil pour le moment on ne s'en occupe pas.

Composer a travaillé et nous permet maintenant d'accéder facilement à *altorouter* et *var\_dumper*.

52:35

Pourquoi ajouter le dossier **vendor** dans le fichier caché **.gitignore** ?

Et bien le dossier **vendor** n'a pas vocation à être versionné, car on peut réinstaller les dépendances avec un **composer install** qui sont initialisées dans le fichier `composer.json` qui lui ne devra jamais être supprimé car c'est un peu la notice d'installation des dépendances initialisées. Ce dossier normalement grossira au fur et à mesure que vous aurez besoin de telle ou telle dépendance, donc il deviendra lourd à versionner via *Git*.

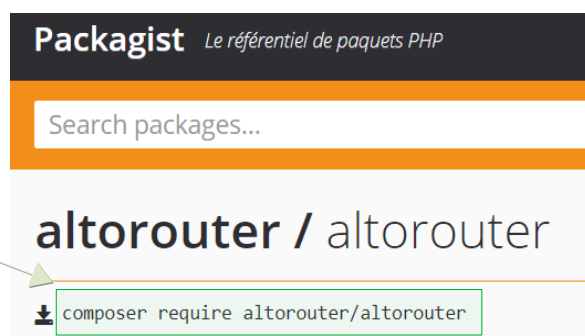
Donc il suffira de supprimer ce dossier avant de le versionner et de refaire un **composer install** afin de réinstaller les packages.

## # Installation des dépendances via packagist.org

Doc → <https://packagist.org/>

Après votre recherche pour choisir votre package. Dans le terminal, dans le dossier racine de votre projet, copier-coller la ligne « **composer require + nom de la dépendance** ».

Il le téléchargera dans le dossier *vendor* et ajoutera la dépendance à installer via la commande **composer install** au fichier `composer.json` à la bonne ligne.



[https://github.com/MartyRizey/S06-E01\\_supp\\_oFramework/tree/2f5e8362ae4102ae81d52003c02c35a831ca2c08](https://github.com/MartyRizey/S06-E01_supp_oFramework/tree/2f5e8362ae4102ae81d52003c02c35a831ca2c08)

D:\win\_Linux Partage\A\_utiliser\_dans\_Linux\Lunar\Video\_Cours\Saison-6\_API\S06E01-oFramework\_autoload-namespace-psr4

L> 57:38 suite → point d'entrée : index.php