

Doc installation API note de frais

Pré-requis

Assurez d'avoir l'outil composer installé en global sur votre système. Cet outil ne fait pas partie de Symfony mais est obligatoire pour celui-ci. C'est un outil de gestion de dépendances qui nous servira à créer des bundles et autres bibliothèques facilement.

Avec la commande `php -v`, assurez-vous d'avoir une version php supérieur ou égale à la *version 7*. Si ça n'est pas le cas télécharger celle-ci et installez la sur votre serveur.

Créer un dossier, à l'endroit voulu pour accueillir le projet (serveur ou localhost) à l'aide de la commande `mkdir`.

Vérifier également que *bin/console* soit exécutable.

Installation sur un serveur

Nettoyer le cache

```
php bin/console cache:clear  
ou  
php bin/console cache:clear --env=prod
```

Si vous effectuez des changements sur des routes de votre projet, utilisez de préférence la deuxième commande.

Téléchargement sur le GIT

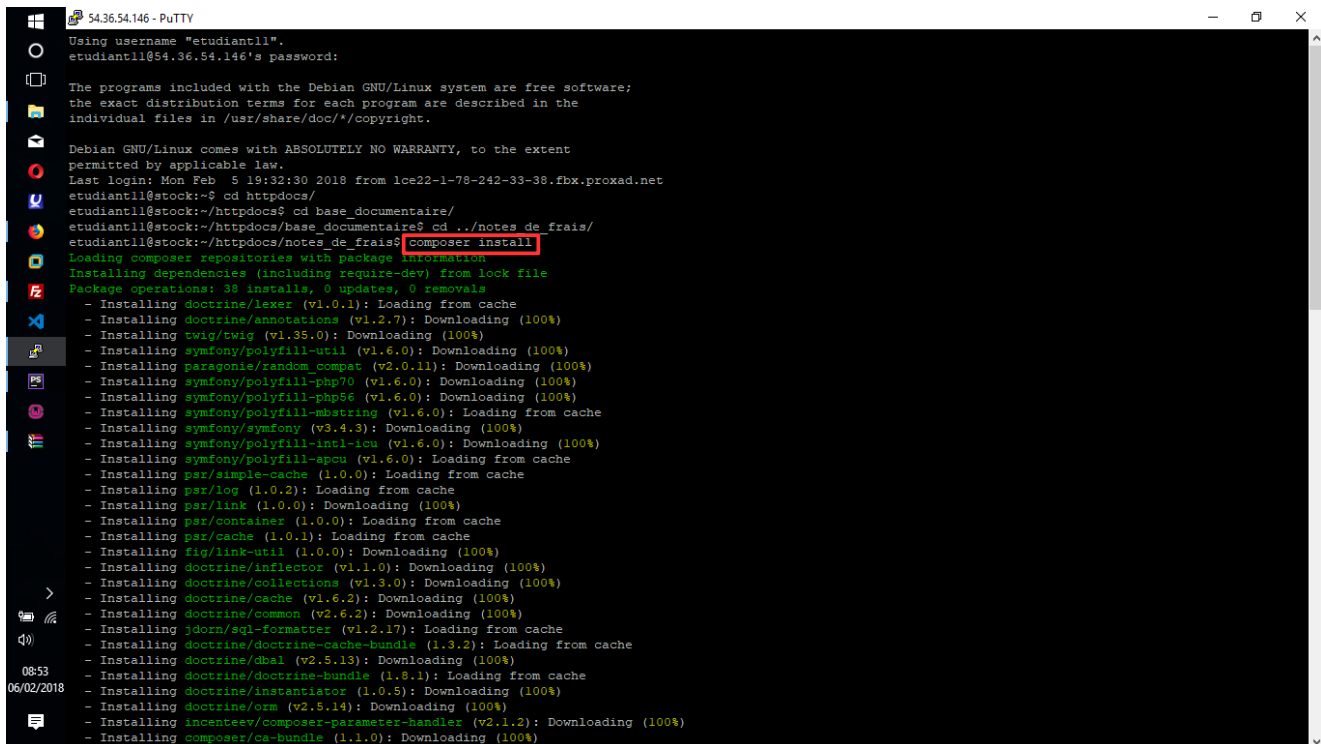
Accéder à la page du projet sur GIT : <https://gitlab.com/clem2a9/RESTApp/>

Si vous souhaitez installer ce projet sur votre serveur, initialiser le en copiant le contenu du dossier *v1* à l'endroit de votre choix sur votre serveur.

Ensuite, vous devez utiliser le `composer install` afin de configurer les dépendances (Cf fig.1).

Pendant cette installation vous pourrez également donner les paramètres désirés pour la base de données nécessaire (Cf fig.2).

Ne pas hésiter à zoomer 😊

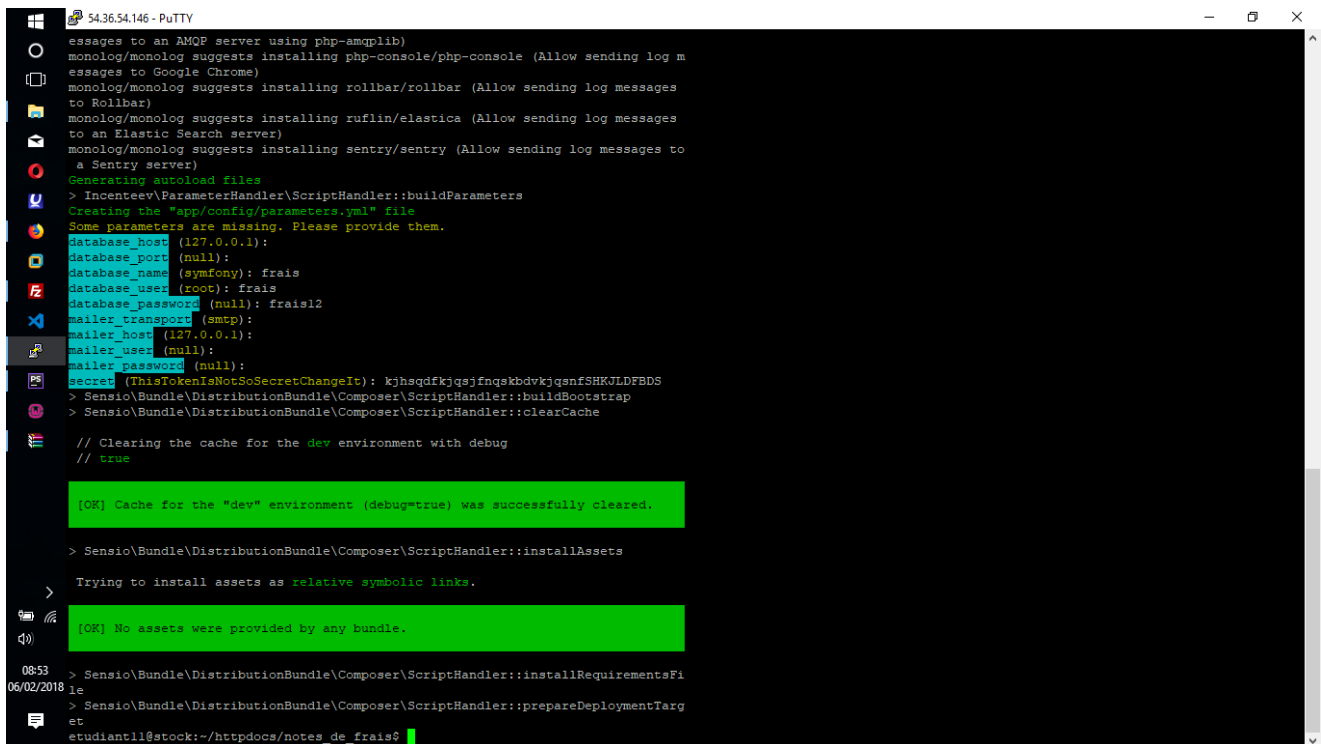


```
54.36.54.146 - PuTTY
Using username "etudiant11".
etudiant11@54.36.54.146's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Feb  5 19:32:30 2018 from 1ce22-1-78-242-33-38.fbx.proxad.net
etudiant11@stock:~$ cd httpdocs/
etudiant11@stock:~/httpdocs$ cd base_documentaire/
etudiant11@stock:~/httpdocs/base_documentaire$ cd ../notes_de_frais/
etudiant11@stock:~/httpdocs/notes_de_frais$ composer install
Loading composer repositories with package information
Installing dependencies (including require-dev) from lock file
Package operations: 38 installs, 0 updates, 0 removals
- Installing doctrine/lexer (v1.0.1): Loading from cache
- Installing doctrine/annotations (v1.2.7): Downloading (100%)
- Installing twig/twig (v1.35.0): Downloading (100%)
- Installing symfony/polyfill-util (v1.6.0): Downloading (100%)
- Installing paragonie/random-compat (v2.0.11): Downloading (100%)
- Installing symfony/polyfill-php70 (v1.6.0): Downloading (100%)
- Installing symfony/polyfill-php56 (v1.6.0): Downloading (100%)
- Installing symfony/polyfill-mbstring (v1.6.0): Loading from cache
- Installing symfony/symfony (v3.4.3): Downloading (100%)
- Installing symfony/polyfill-intl-icu (v1.6.0): Downloading (100%)
- Installing symfony/polyfill-apcu (v1.6.0): Loading from cache
- Installing psr/simple-cache (1.0.0): Loading from cache
- Installing psr/log (1.0.2): Loading from cache
- Installing psr/link (1.0.0): Downloading (100%)
- Installing psr/container (1.0.0): Loading from cache
- Installing psr/cache (1.0.1): Loading from cache
- Installing fig/link-util (1.0.0): Downloading (100%)
- Installing doctrine/infectoer (v1.1.0): Downloading (100%)
- Installing doctrine/collections (v1.3.0): Downloading (100%)
- Installing doctrine/cache (v1.6.2): Downloading (100%)
- Installing doctrine/common (v2.6.2): Downloading (100%)
- Installing jdoorn/sql-formatter (v1.2.17): Loading from cache
- Installing doctrine/doctrine-cache-bundle (1.3.2): Loading from cache
- Installing doctrine/dbal (v2.5.13): Downloading (100%)
- Installing doctrine/doctrine-bundle (1.8.1): Loading from cache
- Installing doctrine/instantiator (1.0.5): Downloading (100%)
- Installing doctrine/orm (v2.5.14): Downloading (100%)
- Installing incenter/composer-parameter-handler (v2.1.2): Downloading (100%)
- Installing composer/ca-bundle (1.1.0): Downloading (100%)
```

fig.1 - Screenshot composer install



```
54.36.54.146 - PuTTY
messages to an AMQP server using php-amqp (lib)
monolog/monolog suggests installing php-console/php-console (Allow sending log m
essages to Google Chrome)
monolog/monolog suggests installing rollbar/rollbar (Allow sending log messages
to Rollbar)
monolog/monolog suggests installing ruflin/elastica (Allow sending log messages
to an Elastic Search server)
monolog/monolog suggests installing sentry/sentry (Allow sending log messages to
a Sentry server)
Generating autoload files
> Incenteev\ParameterHandler\ScriptHandler::buildParameters
Creating the "app/Config/parameters.yml" file
Some parameters are missing. Please provide them.
database_host (127.0.0.1):
database_port (null):
database_name (symfony): frais
database_user (root): frais
database_password (null): fraisl2
mailer_transport (smtp):
mailer_host (127.0.0.1):
mailer_user (null):
mailer_password (null):
secret (ThisTokenIsNotSoSecretChangeIt): kjhsqdfkjqsjfnsgkdvkjqsfnfSHKJLDFBDS
> Sensio\Bundle\DistributionBundle\Composer\ScriptHandler::buildBootstrap
> Sensio\Bundle\DistributionBundle\Composer\ScriptHandler::clearCache
// Clearing the cache for the dev environment with debug
// true

[OK] Cache for the "dev" environment (debug=true) was successfully cleared.

> Sensio\Bundle\DistributionBundle\Composer\ScriptHandler::installAssets
> Trying to install assets as relative symbolic links.

[OK] No assets were provided by any bundle.

08:53
06/02/2018
> Sensio\Bundle\DistributionBundle\Composer\ScriptHandler::installRequirementsFi
le
> Sensio\Bundle\DistributionBundle\Composer\ScriptHandler::prepareDeploymentTarg
et
etudiant11@stock:~/httpdocs/notes_de_frais$
```

fig.2 - Paramétrage de la Base de Données

Cela permettra également d'ajouter les dossiers *vendor* et *config*, indispensables au bon fonctionnement de notre application.

Synchronisation sur le serveur

Une fois ceci fait vous pouvez synchroniser votre dossier contenant votre projet sur votre serveur.

Pour cela vous pouvez utiliser Filezilla ou un autre logiciel.

Votre projet est maintenant opérationnel, vous pouvez le développer en local et synchroniser directement sur le serveur.











Installation de la BDD

Un fichier nommé *frais.sql* est disponible dans le projet GIT.
Pour cela créer votre base et importez-y nos tables.

Puis copier le fichier « parameters.yml.sample » dans v1/app/config et le renommé comme « parameters.yml » avec les données de connexion à votre base de données si cela n'a pas déjà été fait lors de l'installation.

Chemin : v1/app/config

..

 config.yml	Back-End FraisApp	15 days ago
 config_dev.yml	Back-End FraisApp	15 days ago
 config_prod.yml	Back-End FraisApp	15 days ago
 config_test.yml	Back-End FraisApp	15 days ago
 parameters.yml.dist	Back-End FraisApp	15 days ago
 parameters.yml.sample	gitignore	15 days ago
 routing.yml	Back-End FraisApp	15 days ago
 routing_dev.yml	Back-End FraisApp	15 days ago
 security.yml	Back-End FraisApp	15 days ago
 services.yml	Back-End FraisApp	15 days ago

```
# This file is auto-generated during the composer install
parameters:
database_host: 127.0.0.1
database_port: null
database_name: database.name
database_user: database.user.name
database_password: database.user.pwd
mailer_transport: smtp
mailer_host: 127.0.0.1
mailer_user: null
mailer_password: null
secret: SecretID
```

6. Informations complémentaires:

7. Activer le débbugger :

Ouvrez le fichier web/app.php, on va activer le mode *debugger* pour cet environnement. Il correspond au deuxième argument du constructeur du Kernel :

```
<?php
// web/app.php
// ...
$kernel = new AppKernel('prod', true); // Définissez ce 2e argument à true
```

Pensez à bien remettre ce paramètre à false lorsque vous avez fini vos tests !

4. Qualité du code :

Vous pouvez vérifier la qualité de votre code sur [insight sensiolabs](https://insight.sensiolabs.com/) avant de le publier sur votre serveur.

5. Sécurité du code :

Il faut ensuite vérifier la sécurité des dépendances. Pour cela, il existe un outil également créé par SensioLabs : Security Checker (security.sensiolabs.org). Vous avez deux solutions pour vérifier vos dépendances.

La première est d'envoyer manuellement votre fichier composer.lock sur l'interface en ligne de l'outil. C'est lui qui contient les versions exactes des dépendances que vous utilisez, cela permet donc à l'outil de vérifier avec sa base de données internes des failles de sécurité.