
Create an SPA from 0 to 100 in nn steps

Release 0.0.1

Jean-Philippe Maquestiaux

Dec 27, 2020

CONTENTS:

1	Purpose	1
2	Remarks	3
3	Technologies	5
4	Building the application	7
5	Installing the necessary tools	9
5.1	NodeJS	9
5.2	NPM	9
5.3	YARN	10
5.4	eUI-CLI	10
5.5	Symfony CLI	10
5.6	Composer	10
5.7	GIT	10
5.8	PHP 7.X	10
6	Initialise the application with eUI CLI	11
6.1	Create the folder structure	11
6.2	Launch the eUI CLI	12
6.3	Under Windows known issue	14
6.4	Extra features to install	16
6.5	Retrieve the packages	16
7	Starting with the API	19
7.1	Configure the database access	19
7.2	Configure the API Platform	19
8	Indices and tables	21

PURPOSE

Soon...

REMARKS

This “tutorial” is based on a Windows 10 environment

TECHNOLOGIES

- eUI 10.x (Set of component over Angular 10.x)
- Symfony (5.x) with API-Platform
- EUI Login for securing the app (@ECPHP/eulogin-bundle)
- MySQL for the database

BUILDING THE APPLICATION

DIGIT has create a CLI for generating the skeleton of an SPA frontend using eUI. The CLI offers the option of adding into the skeleton the backend part

- Installing the necessary tools
- Initialise the applications with eUI CLI

INSTALLING THE NECESSARY TOOLS

Before going further, we need to insure that the developer environment is having the necessary software.

You will have to check the installation approach based on your environment (OS, network)

5.1 NodeJS

The version needed is the 10.x.

For managing the NodeJS version, we recommend is usage. It will allow you to install multiple NodeJS version and switch between the different version.

You can find NVM on github: <https://github.com/nvm-sh/nvm>

Several internet site explain how to install it over the different existing OS (Linux, OSX and Windows).

Below some site explaining how to install:

<https://heynode.com/tutorial/install-nodejs-locally-nvm>

<https://docs.microsoft.com/en-us/windows/nodejs/setup-on-windows>

```
1 nvm install 10.23.0
```

5.1.1 A sub of NodeJus

Just for test

5.2 NPM

You will need also to have NPM for installing packages. Which is installed along with NodeJs : <https://www.npmjs.com/get-npm>

5.3 YARN

Yarn is another package manager.

<https://yarnpkg.com/getting-started/install>

```
npm install -g yarn
```

5.4 eUI-CLI

You install the eUI Cli using NPM (or Yarn). The package is located here:

<https://www.npmjs.com/package/@eui/cli>

```
npm install -g @eui/cli@latest
```

5.5 Symfony CLI

You can install also the Symfony CLI. The installed is located: <https://symfony.com/download>

The CLI is not mandatory but provide useful tools / shortcut to the command.

5.6 Composer

You have to install Composer: <https://getcomposer.org/download/>

5.7 GIT

You have to install GIT (which comes with Git-Bash): <https://git-scm.com/downloads>

Eventually, you can install a GUI: <https://git-scm.com/downloads/guis>

5.8 PHP 7.X

You will also need PHP installed: <https://www.php.net/manual/en/install.windows.php>

INITIALISE THE APPLICATION WITH EUI CLI

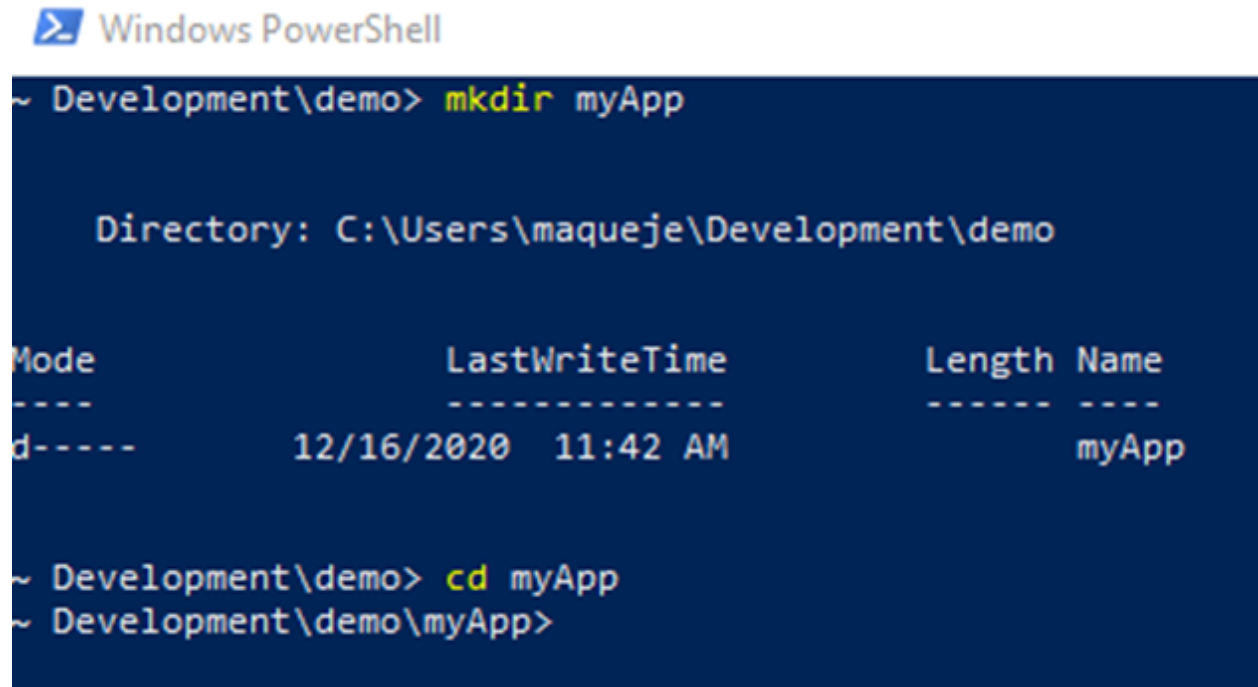
You need to open a command line. As being under windows, launch a PowerShell terminal.



We are creating a folder where we will create the SPA. We will execute the eUI CLI from that folder.

6.1 Create the folder structure

```
1 cd your-root-dev-folder
2 mkdir myApp
3 cd myApp
4 eui-cli
```



A screenshot of a Windows PowerShell terminal window. The title bar reads 'Windows PowerShell'. The prompt is '~ Development\demo>'. The command 'mkdir myApp' has been executed. The output shows the directory path 'Directory: C:\Users\maqueje\Development\demo' followed by a table listing the newly created directory.

Mode	LastWriteTime	Length	Name
d-----	12/16/2020 11:42 AM		myApp

The prompt then changes to '~ Development\demo>' and the command 'cd myApp' is entered. The final prompt is '~ Development\demo\myApp>'.

6.2 Launch the eUI CLI

```
1 cd your-root-dev-folder
2 cd myApp
3 eui-cli
```

The eUI CLI is an interactive systematic tool.

It will let you choose your installation flavour.


```

~ Development\demo\t> eui-cli

eui cli

v10.2.1

Configuring your project : please answer the following :

? Select an application type :
  Angular
  Ionic Angular - mobile
  Spring Boot web module - JDK + Maven required
  Maven web module (JEE Spring maven web module) - JDK + Maven required !!!
  full-stack (JEE Spring multi-maven) - JDK + Maven required !!!
> PHP/Symfony starter - PHP7 and composer required !!!

```

In our case, we are selecting: **PHP/Symfony Starter - PHP6 and composer required !!!**

The next step is for selecting options, we are selecting: **OpenId Connect JWT implementation**

```

~ Development\demo\t> eui-cli

eui cli

v10.2.1

Configuring your project : please answer the following :

? Select an application type : PHP/Symfony starter - PHP7 and composer required !!!
? Select one or more additional configuration options :
>(*) OpenId Connect JWT implementation
  ( ) ECL app-shell template and configuration - for ec.europa.eu apps

```

And just accept the dependencies installation

```
~ Development\demo\t> eui-cli

eui cli

v10.2.1

Configuring your project : please answer the following :

? Select an application type : PHP/Symfony starter - PHP7 and composer required !!!
? Select one or more additional configuration options : OpenId Connect JWT implementation
? Install dependencies when the app is generated : (Y/n)
```

From there, the eUI CLIT will create the folders: one for the front (client) and one for the backoffice (server).

Under windows, the eUI CLI script will fail when trying to execute a post clone script of the backend (symfony) which is written in bash script (init.sh)

6.3 Under Windows known issue

the eUI CLI script will fail when trying to execute a post clone script of the backend (symfony) which is written in bash script (init.sh). Under windows, CMD or PowerShell script can be run natively.

```

Configuring your project : please answer the following :
? Select an application type : PHP/Symfony starter - PHP7 and composer required !!!
? Select one or more additional configuration options : OpenId Connect JWT implementation
? Install dependencies when the app is generated : Yes
--provided config :
{
  appName: 'my-app',
  appScope: '@app',
  appType: 'web-symfony',
  appOptions: [ 'openid-jwt' ],
  appFeatures: [],
  groupId: 'ec.europa.digit',
  artifactId: 'myapp',
  useYarn: true,
  npmInstall: true
}
{
  appName: 'my-app',
  appScope: '@app',
  appType: 'web-symfony',
  appOptions: [ 'openid-jwt' ],
  appFeatures: [],
  groupId: 'ec.europa.digit',
  artifactId: 'myapp',
  useYarn: true,
  npmInstall: true
}
Building app : web-symfony
--cleaning target : C:\Users\maqueje\Development\demo\t
--cloning eui php starter repository
--building web-symfony
>> ----->copydir from : C:\Users\maqueje\AppData\Local\npm\node_modules\@eui\cli\lib\skeletons\_angular\base ====> to
C:\Users\maqueje\Development\demo\t\client
Applying option : openid-jwt
>> ----->copydir from : C:\Users\maqueje\AppData\Local\npm\node_modules\@eui\cli\lib\skeletons\_angular\options\openid
jwt ====> to : C:\Users\maqueje\Development\demo\t\client

> server@1.0.0 init C:\Users\maqueje\Development\demo\t\server
> ./init.sh

'.' is not recognized as an internal or external command,
operable program or batch file.
npm ERR! ELIFECYCLE
npm ERR! 1
npm ERR! server@1.0.0 init: `./init.sh`
npm ERR! Exit status 1
npm ERR!
npm ERR! Failed at the server@1.0.0 init script.
npm ERR! This is probably not a problem with npm. There is likely additional logging output above.
npm WARN Local package.json exists, but node_modules missing, did you mean to install?

npm ERR! A complete log of this run can be found in:
npm ERR! C:\Users\maqueje\AppData\Local\npm-cache\_logs\2020-12-17T10_57_14_024Z-debug.log
Error: Command failed: C:\WINDOWS\system32\cmd.exe /q /s /c "npm run init"
    at makeError (C:\Users\maqueje\AppData\Local\npm\node_modules\@eui\cli\node_modules\execa\index.js:174:9)
    at module.exports.sync (C:\Users\maqueje\AppData\Local\npm\node_modules\@eui\cli\node_modules\execa\index.js:338:15)
    at handleShell (C:\Users\maqueje\AppData\Local\npm\node_modules\@eui\cli\node_modules\execa\index.js:117:9)
    at Function.module.exports.shellSync (C:\Users\maqueje\AppData\Local\npm\node_modules\@eui\cli\node_modules\execa\index.js:361:43)
    at C:\Users\maqueje\AppData\Local\npm\node_modules\@eui\cli\lib\build.js:124:19
    at processTicksAndRejections (internal/process/task_queues.js:93:5) {
  code: 1,
  stdout: null,
  stderr: null,
  failed: true,
  signal: null,
  cmd: 'C:\\WINDOWS\\system32\\cmd.exe /q /s /c "npm run init"',
  timedOut: false
}

```

The init.sh (<https://github.com/MaquestiauxTraining/spa-0-100-eui/blob/main/server/init.sh>) is very simple.

```

1  #!/bin/bash
2  echo 'Checking PHP prerequisites ...'
3  check_cmd() {
4      $1 --version > /dev/null 2>&1
5      OUT=$?

```

(continues on next page)

(continued from previous page)

```
6  if [[ $OUT -ne 0 ]]; then
7      echo "$1 command not found."
8  else
9      echo "$1 command found."
10  fi
11  return $OUT
12 }
13 if check_cmd "php"; then
14     if check_cmd "composer"; then
15         echo "Building Symfony application ..."
16     else
17         echo "ERROR: Symfony application cannot be built."
18         exit 1
19     fi
20 else
21     echo "ERROR: Symfony application cannot be built."
22     exit 1
23 fi
24 cd "$(dirname "$0")"
25 composer create-project symfony/skeleton app
26 cd app
27 composer req api annotations
28 composer req maker --dev
```

You can run manually the composer commands as in the last line of the init.sh (<https://github.com/MaquestiauxTraining/spa-0-100-eui/blob/main/server/init.sh>)

6.4 Extra features to install

You need also to add the following package for securing the back office using EU Login

```
1  composer req ecphp/eu-login-bundle
```

You can add other package such:

```
1  composer req --dev doctrine/doctrine-fixtures-bundle
2  composer req --dev symfony/web-profiler-bundle
```

6.5 Retrieve the packages

For both, Front and Back, you have to retrieve the packages.

For the Front, you have to go in the folder called client:

```
1  cd your-root-dev-folder
2  cd myApp
3  cd client
4  yarn install
```

For the Back, you have to go in the folder called app (sub folder of server) server:

```
1 cd your-root-dev-folder
2 cd myApp
3 cd server
4 cd app
5 composer install
```


STARTING WITH THE API

7.1 Configure the database access

Configure the `/server/app/.env` OR the `/server/app/config/packages/doctrine.yaml`

7.2 Configure the API Platform

Update the configuration files

- `/server/app/config/packages/api_platform.yaml`
- `/server/app/config/routes/api_platform.yaml`

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`