



DOCUMENTATION

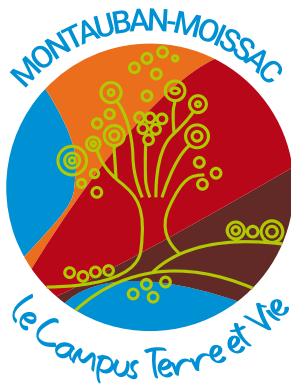
**IRRIGATION
CONNECTÉE**

2021/2022

Table des matières

I. Contexte et analyse du projet.....	5
I. Analyse de l'existant.....	5
II. Analyse du besoin.....	6
II. Installer Symfony sous Windows.....	8
I. Installation de php.....	8
I. Désactivation de l'UAC.....	8
II. Installation de XAMPP.....	10
II. Installation de Composer.....	14
III. Installation de Symfony.....	16
IV. Débuter avec Symfony.....	17
I. Création d'un projet.....	17
II. Démarrer Symfony.....	18
V. Architecture de Symfony.....	19
VI. Interaction avec une base de données.....	20
I. L'ORM Doctrine.....	20
II. Connexion à la BDD.....	21
III. Les Entités.....	21
IV. Les Migrations.....	21
III. Installer Symfony sous Debian.....	22
I. Installation de php.....	22
II. Installation de Composer.....	22
III. Installation de Symfony.....	22
IV. Débuter avec Symfony.....	23
I. Création d'un projet.....	23
II. Démarrer Symfony.....	24
V. Interaction avec une base de données.....	26
IV. Démarrage du projet d'irrigation connectée.....	27
I. Archives du projet.....	27
II. Mise en place de la BDD.....	28
III. Connexion au site.....	29
IV. Modification de l'adresse Mail.....	29
V. Installer Django sous Windows.....	31
I. Installation de Python.....	31
I. Vérification.....	31
II. Téléchargement de Python.....	31
III. Installation de Python.....	32
IV. Droits de sécurité.....	34
V. Installation des bibliothèques.....	37
II. Installation de Django.....	39
III. Débuter avec Django.....	40

I. Création d'un projet.....	40
II. Architecture de Django.....	40
III. Démarrer Django.....	41
IV. Installation Xampp.....	42
VI. Installer Django sous Debian.....	43
I. Installation de Python.....	43
I. Vérification.....	43
II. Installation de Python.....	43
III. Installation des bibliothèques.....	44
II. Installation de Django.....	44
VII. Démarrage du projet d'Irrigation Connectée.....	45
I. Mise en place de la BDD.....	45
II. Changement de l'adresse mail.....	45
III. Démarrage du site.....	45
IV. Premier lancement du site.....	45
VIII. Installation partie Matérielle.....	47
I. ESPTools et pySerial.....	47
II. Gestion des droits.....	47
III. Flasher une nouvelle adresse MAC sur un ESP32.....	48
IV. Téléverser le code sur les cartes.....	48
V. Assemblage du piquet.....	49



Contexte et analyse de l'application logicielle

[23/06/2022]

I. Contexte et analyse du projet

I. Analyse de l'existant

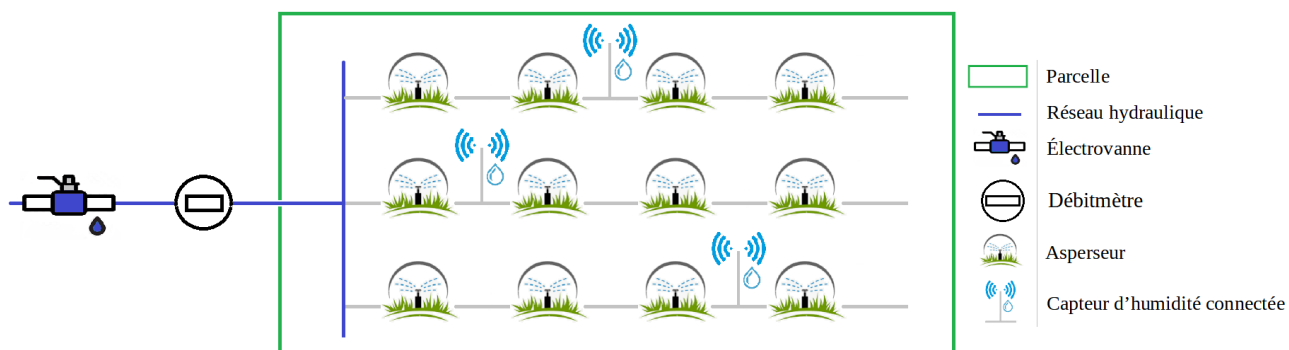
Le domaine du Lycée Agricole de Montauban, étant situé au bord de la rivière, est autorisé à puiser l'eau du Tarn afin d'irriguer l'ensemble de leur exploitation. Pour assurer le puisage de l'eau, le système est composé de deux pompes directement immergées dans le Tarn. La première pompe est dite auto-régulée. Ainsi, elle adapte le débit de l'eau en fonction de la demande. La deuxième pompe est fixe et se met en marche lorsque la première pompe ne suffit plus pour répondre aux besoins.

Le système de pompage est alimenté et commandé par une armoire électrique située au sein même du domaine agricole. En ce qui concerne les vannes, celles-ci sont situées en tête de parcelle sur l'ensemble de l'exploitation et sont contrôlées manuellement. La méthode d'irrigation présente sur l'exploitation est une technique d'irrigation moderne, il s'agit de l'irrigation par aspersion. Enfin, à ce stade, le contrôle de l'irrigation se fait intégralement de manière visuelle.

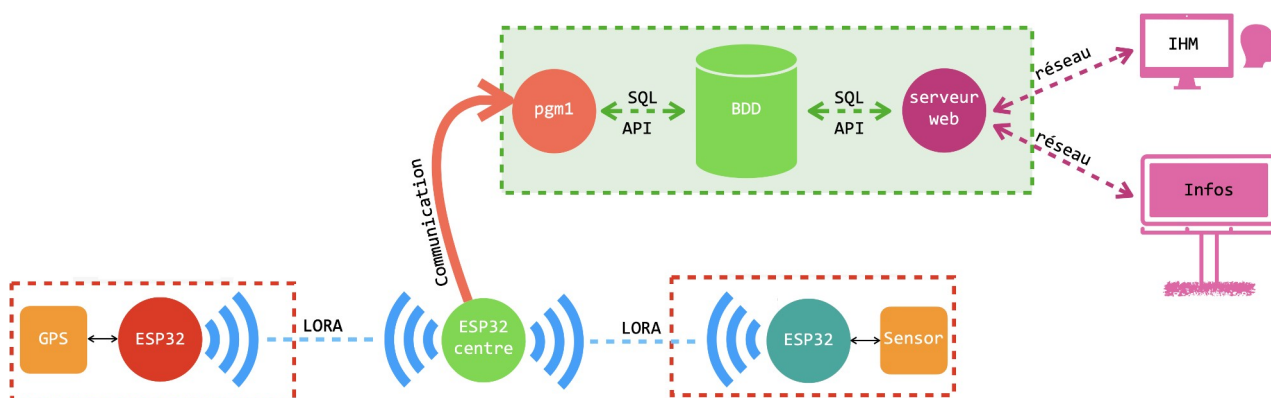


II. Analyse du besoin

Suite à l'analyse du système existant, il a été possible d'établir un cahier des charges. Le contrôle des vannes et la vérification de l'irrigation se faisant intégralement de manière manuelle, c'est sur ces principaux points que les besoins ont été établis. Ainsi, notre projet consiste à mettre en place un système permettant de contrôler les vannes mais également l'irrigation des parcelles au travers d'une application logicielle. Le contrôle de l'exploitation doit être accessible sur divers appareils. De plus, le chef d'exploitation souhaitant avoir accès à des mesures tel que le taux d'humidité des sols sur les différentes parcelles, il est nécessaire d'ajouter divers capteurs et actionneurs à l'installation présente sur le domaine agricole. En conséquence, le système sera complété par des capteurs d'humidité connectés, par un débitmètre ainsi qu'une électrovanne.



Ici, les données envoyées par les divers appareils sont réceptionnées par l'ESP32 puis envoyées via des requêtes HTTP POST à une page web. Les données envoyées sur cette page seront traitées et envoyées dans la base de données. Enfin ces mêmes données seront exploitées afin de les afficher sur l'IHM.





Installation du Framework Symfony sous Windows et Debian

[23/06/2022]

II. Installer Symfony sous Windows

Avant de commencer à utiliser Symfony il sera nécessaire d'installer quelques pré-requis :

- php est un langage de programmation qui permettra d'implémenter les projets.
- Composer est un gestionnaire de dépendance écrit en php. Il permettra de gérer en ligne de commande les bibliothèques nécessaires pour développer les projets.
- CLI Symfony permettra de créer, exécuter et gérer les projets.

I. Installation de php

Il est possible d'installer php sous windows à partir du lien suivant : <https://www.php.net/download>

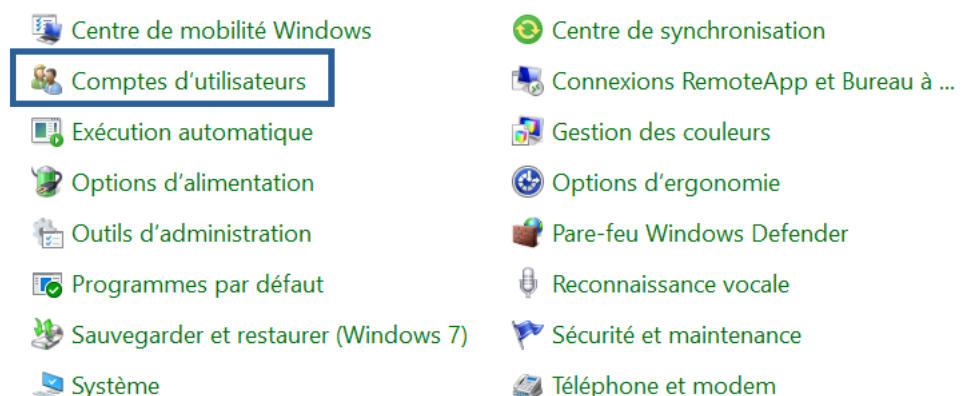
Cependant, il existe XAMPP. Celui-ci est un ensemble de logiciels permettant de créer facilement une interface web interagissant avec une base de données. Entre autres, XAMPP comprend Apache, MySQL avec MariaDB, php et perl. XAMPP est disponible en suivant le lien suivant : <https://www.apachefriends.org/index.html>

I. Désactivation de l'UAC

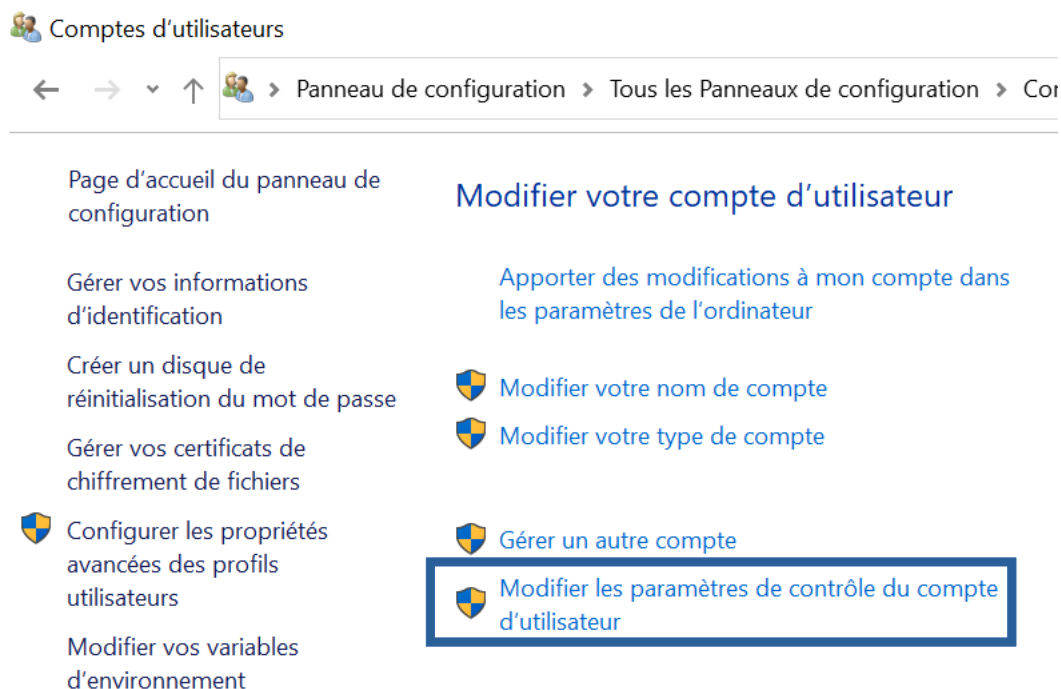
Qu'est-ce que l'UAC (User Account Control) ?

Pour résumer, l'UAC est une fonctionnalité de sécurité de Windows 10 qui empêche toute modification non autorisée ou involontaire du système d'exploitation. Ici, il est nécessaire de désactiver cette fonctionnalité avant de procéder aux installations qui vont suivre.

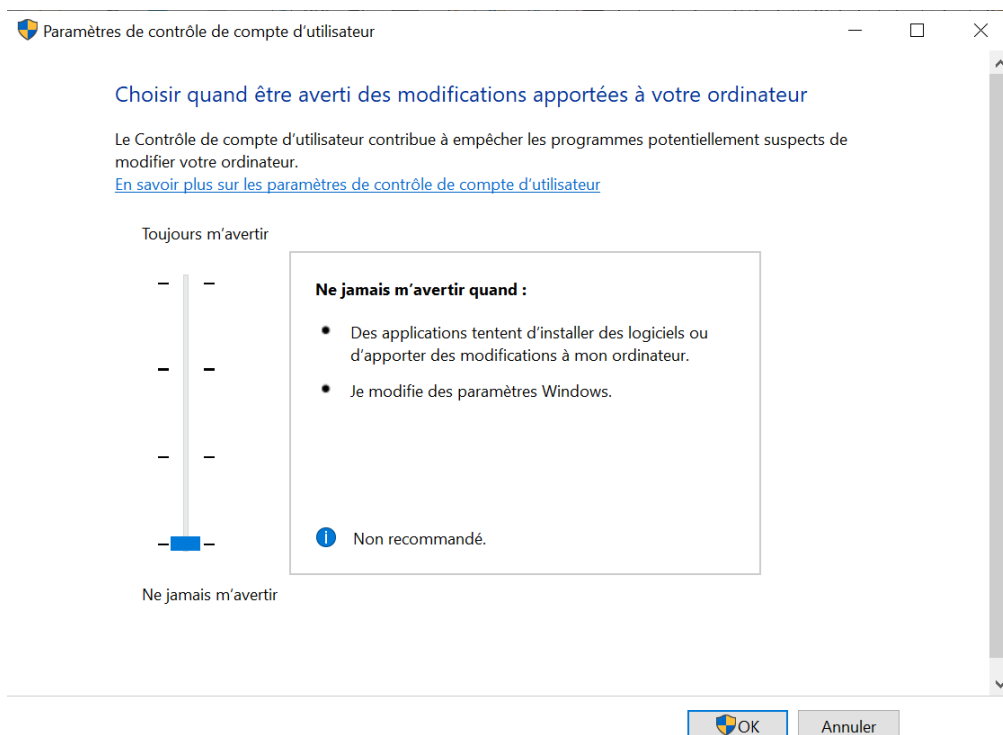
Tout d'abord, rendez-vous dans le panneau de configuration puis rechercher **Comptes d'utilisateurs** :



Ensuite, sélectionner **Modifier les paramètres de contrôle du compte d'utilisateur** :



Puis, abaisser le curseur à **Ne jamais m'avertir**. A ce niveau, l'UAC ne vous avertit pas lorsqu'un programme tente d'effectuer des modifications, d'installer un logiciel ou de modifier les paramètres de Windows. Cependant, cela est nécessaire afin que php puisse fonctionner convenablement.



A présent, nous pouvons procéder aux installations.

II. Installation de XAMPP

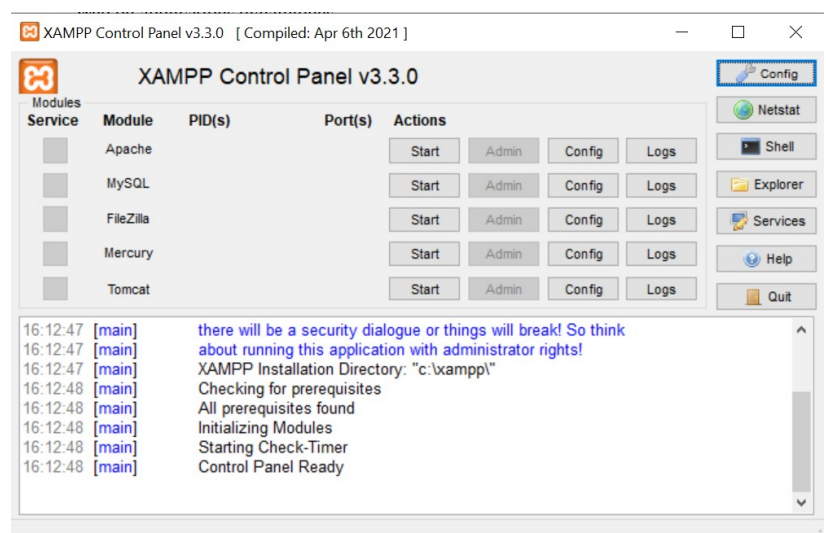
Qu'est-ce que XAMPP ?

XAMPP possède de nombreux avantages pour mettre en place un serveur web local. XAMPP regroupe divers logiciels qui sont relativement simple d'utilisation. Parmi ces logiciels PHP, MySQL et Apache pourront être utilisés.



- Apache : Apache permet de délivrer des contenus Web.
- MySQL : MySQL est un système de gestion de base de données. Autrement dit, il sert à l'enregistrement de données.
- PHP : PHP est un langage script côté serveur permettant de créer des pages Web ou applications dynamiques.

XAMPP Control Panel permet de contrôler les composants du serveur de test. L'interface claire recueille toutes les actions. Ce panneau de configuration permet de démarrer ou d'arrêter certains modules :

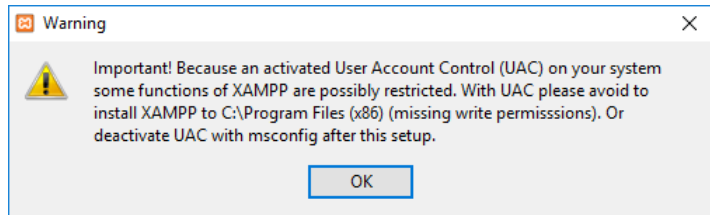


« Panneau de contrôle »

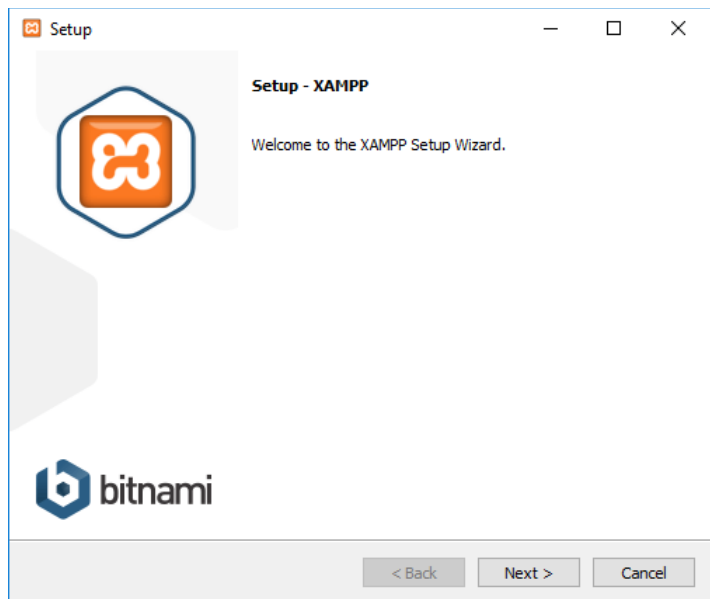
Cette interface propose des actions diverses :

- Config : permet la configuration de XAMPP ainsi que des composants partiels
- Netstat: affiche les processus courants sur l'ordinateur local
- Shell: ouvre une cellule UNIX
- Explorer: ouvre le dossier XAMPP dans Windows Explorer
- Services : affiche les services fonctionnant en arrière-plan
- Aide : propose des liens vers des forums pour utilisateurs
- Terminer : ferme le XAMPP-Control-Panel

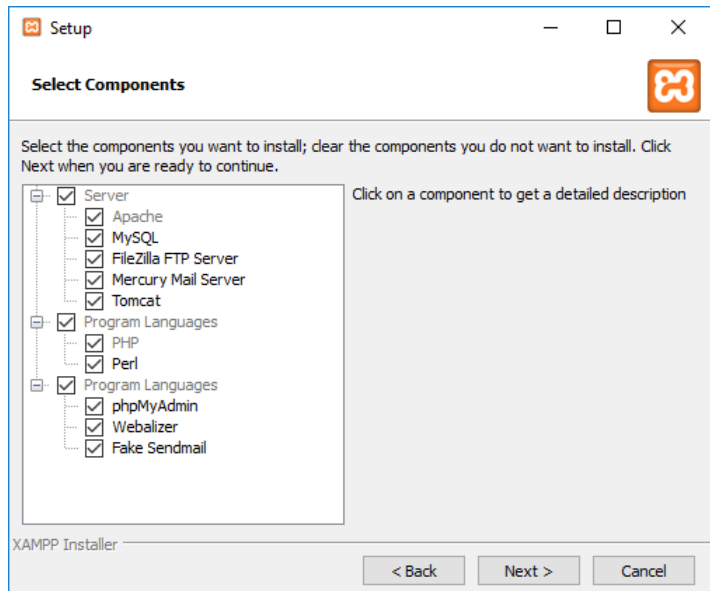
Lors de l'installation de XAMPP, un message nous avertis qu'il est important de désactiver l'UAC :



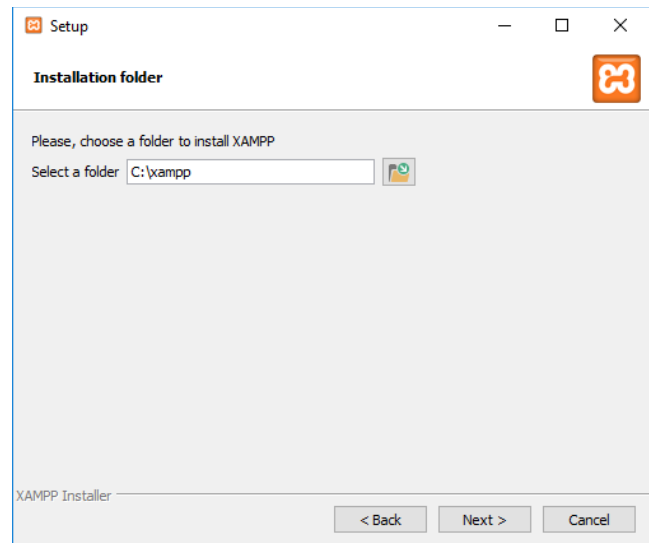
Cliquer sur **Next** :



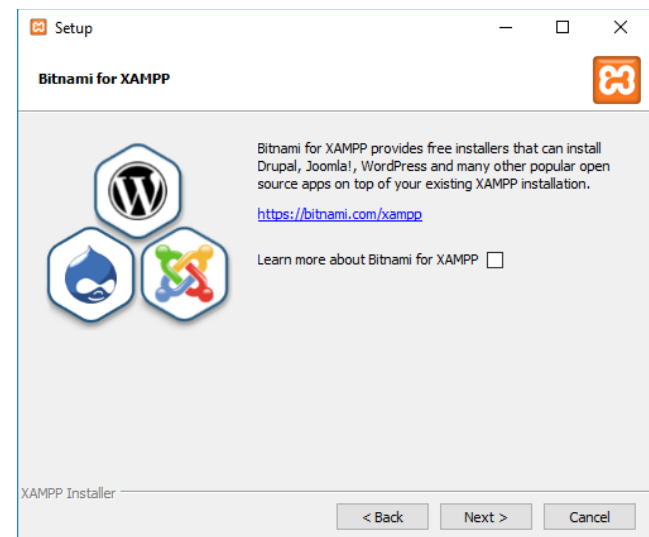
Ici, nous avons la possibilité de sélectionner les composants que nous souhaitons installer, nous allons laisser les choix par défauts :



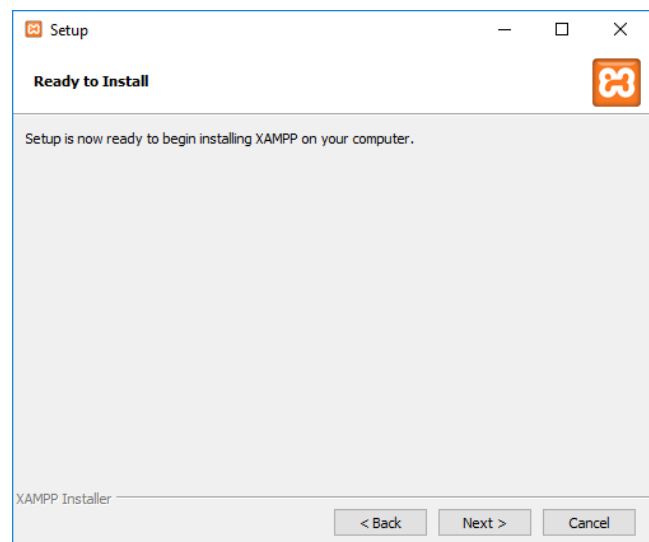
Ici, nous avons la possibilité de sélectionner où sera réalisé l'installation. Il est fortement conseillé de réaliser l'installation à la racine. Certains systèmes d'exploitation tel que Linux ne supportent pas les caractères spéciaux :



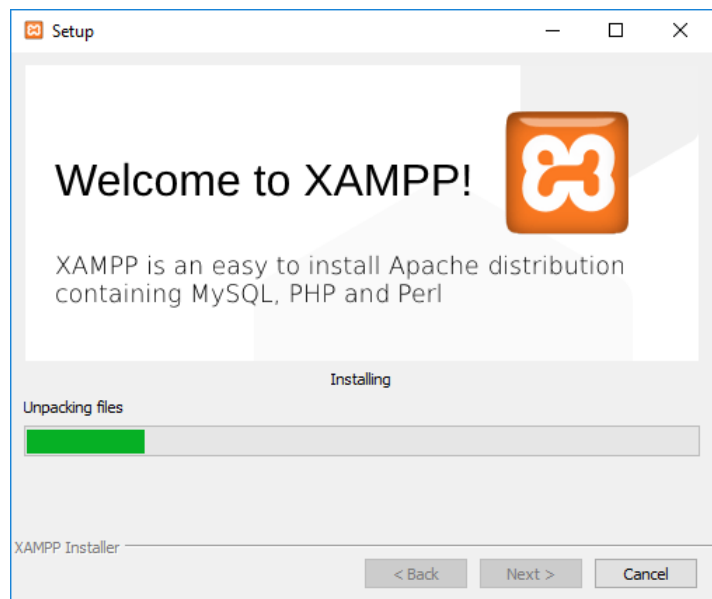
Ici, on laisse par défaut et on poursuit :



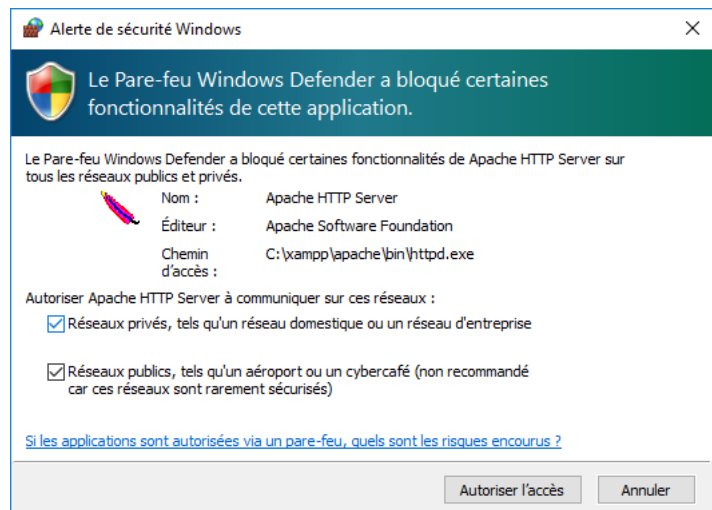
Encore une fois, on continue en cliquant sur **Next** :



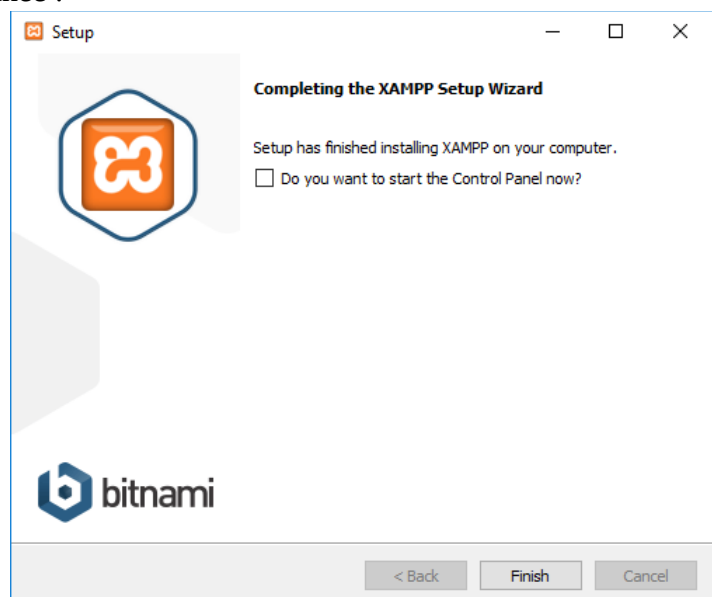
Ensuite, l'installation commence :



Une fois l'installation fini, veuillez autoriser l'accès :



A présent, l'installation de XAMPP est terminée :



II. Installation de Composer

Qu'est-ce que Composer ?

Composer est un outil indispensable, il sert de gestionnaire de dépendance entre l'application et les librairies. De façon plus précise, Composer permet de gérer pour chaque projet, la liste des modules et bibliothèques nécessaires à son fonctionnement ainsi que leurs versions. Il est utilisable via la console en ligne de commande.



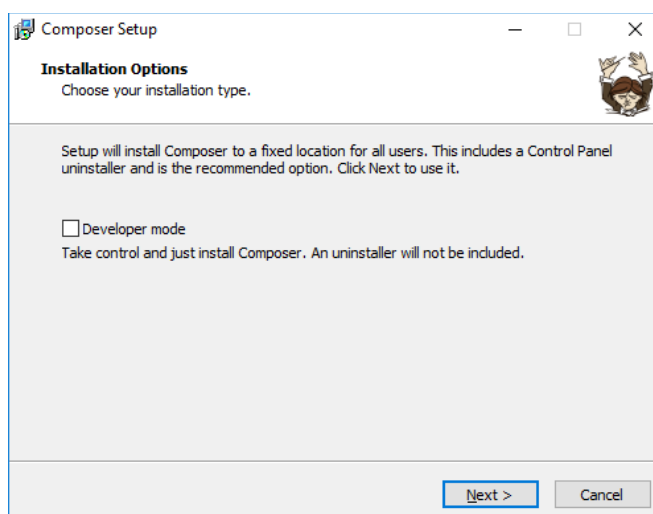
Composer dispose de plusieurs paramètres, très utiles :

1. require: ajoute la bibliothèque en paramètre au fichier composer.json et l'installe.
2. install : installe toutes les bibliothèques du composer.json. Il s'agit de la commande à lancer pour installer toutes les dépendances d'un dépôt PHP.
3. update : met à jour les bibliothèques du composer.json, selon les versions permises qui y sont mentionnées.
4. remove : désinstalle une bibliothèque et la retire du composer.json.

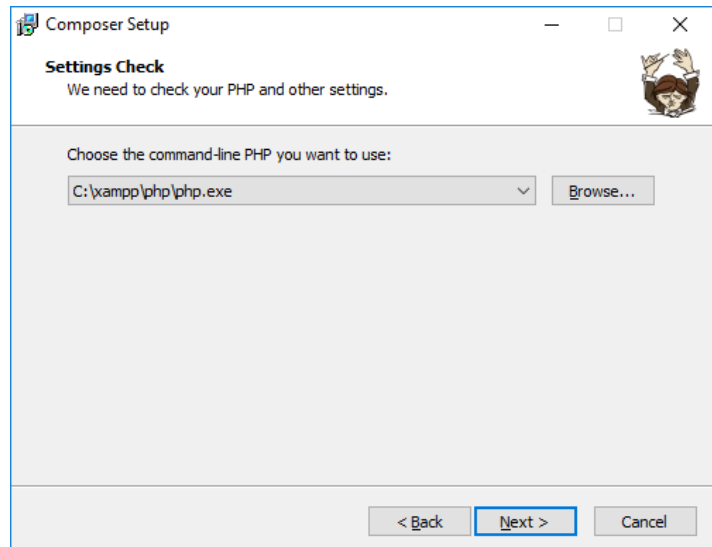
Ces commandes auront pour effet d'ajouter des lignes dans le fichier composer.json. Le fichier composer.json est un fichier qui stocke tous les paquets (dépendances) d'un projet, ainsi que leurs versions.

Il est possible de télécharger Composer à partir du lien suivant : <https://getcomposer.org/download/>

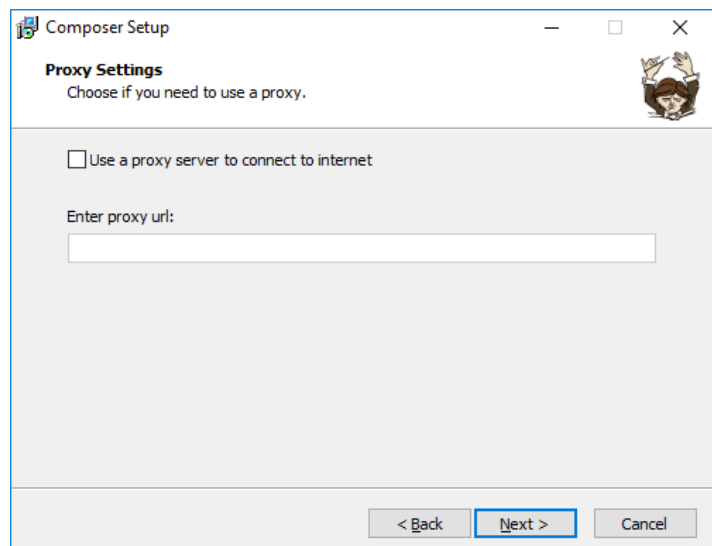
Il est conseillé de télécharger le fichier Composer-setup.exe. Celui-ci configurera votre variable d'environnement PATH afin que vous puissiez simplement appeler composer depuis n'importe quel répertoire. Ici, on ne sélectionne pas le mode développeur et on poursuit en cliquant sur **Next** :



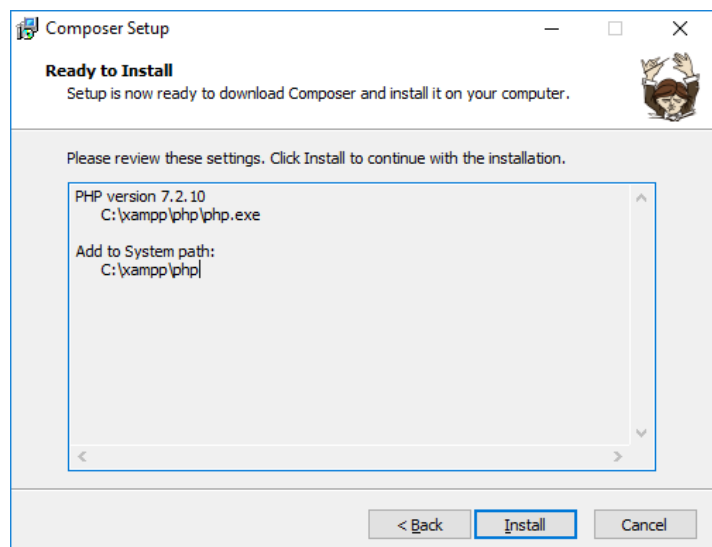
Ensuite, il faut sélectionner le chemin du fichier php :



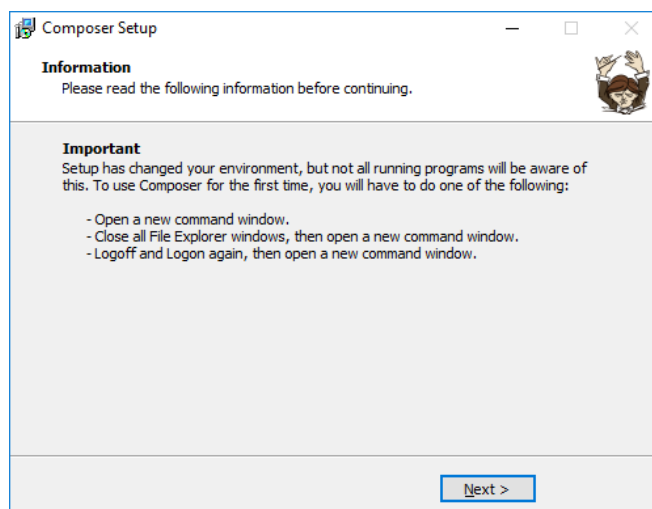
Si l'installation se réalise au sein d'un établissement scolaire, il sera nécessaire de renseigner le proxy sinon ne rien renseigner :



Composer, nous indique les modifications qui vont être réalisées :



On continue en cliquant sur **Next** :



Enfin, on termine l'installation :



III. Installation de Symfony

Qu'est-ce que Symfony ?

Symfony est un puissant framework PHP. Celui-ci est utilisé pour développer des applications web ou des sites web. Relativement simple d'accès, Symfony est un ensemble de composants qui facilitent le développement web en réduisant de façon considérable le temps de développement.



L'installation de Symfony 6 ne se fait plus de manière graphique mais à partir d'une CLI. Symfony peut s'installer avec Scoop ou GoFish. Pour la suite de l'installation cela se réalisera avec Scoop. L'installation avec Scoop se fait avec les commandes suivantes :

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\Melissa> Set-ExecutionPolicy RemoteSigned -scope CurrentUser
```


La commande **Set-ExecutionPolicy RemoteSigned -scope CurrentUser** permet de modifier la stratégie d'exécution (script).

Sélectionner **O** afin de ne pas modifier la stratégie pour tout :

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\Melissa> Set-ExecutionPolicy RemoteSigned -scope CurrentUser

Modification de la stratégie d'exécution
La stratégie d'exécution permet de vous prémunir contre les scripts que vous jugez non fiables. En modifiant la
stratégie d'exécution, vous vous exposez aux risques de sécurité décrits dans la rubrique d'aide
about_Execution_Policies à l'adresse https://go.microsoft.com/fwlink/?LinkID=135170. Voulez-vous modifier la stratégie
d'exécution ?
[O] Oui [T] Oui pour tout [N] Non [U] Non pour tout [S] Suspendre [?] Aide (la valeur par défaut est « N ») :
```

Ensuite, entrer la commande **iwr -useb get.scoop.sh | iex** afin d'installer Scoop :

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\Melissa> iwr -useb get.scoop.sh | iex
```

Enfin, pour installer Symfony, entrer la commande **scoop install symfony-cli** :

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\Melissa> scoop install symfony-cli
```

IV. Débuter avec Symfony

I. Création d'un projet

Un projet peut-être créé de deux façons. Tout d'abord avec Symfony en entrant la commande suivante :

```
PS C:\Users\Melissa> symfony new ProjetTest --full
```

Il est également possible de créer un projet avec composer en entrant la commande qui suit :

```
PS C:\Users\Melissa> composer create-project symfony/skeleton ProjetTest
```

Dans notre cas, le projet a été installé à la racine du répertoire utilisateur.

II. Démarrer Symfony

Pour lancer le projet il n'est pas nécessaire de démarrer un serveur Web local, il est possible de le faire en ligne de commande avec le serveur intégré de php. La commande est : **symfony server:start**

```
Windows PowerShell
PS C:\Users\Melissa> cd ProjetTest
PS C:\Users\Melissa\ProjetTest> symfony server:start

INFO  A new Symfony CLI version is available (5.4.5, currently running 5.3.4).

If you installed the Symfony CLI via a package manager, updates are going to be automatic.
If not, upgrade by downloading the new version at https://github.com/symfony-cli/symfony-cli/releases
And replace the current binary (symfony.exe) by the new one.

[WARNING] run "symfony.exe server:ca:install" first if you want to run the web server with TLS support, or use "--p12"
or "--no-tls" to avoid this warning

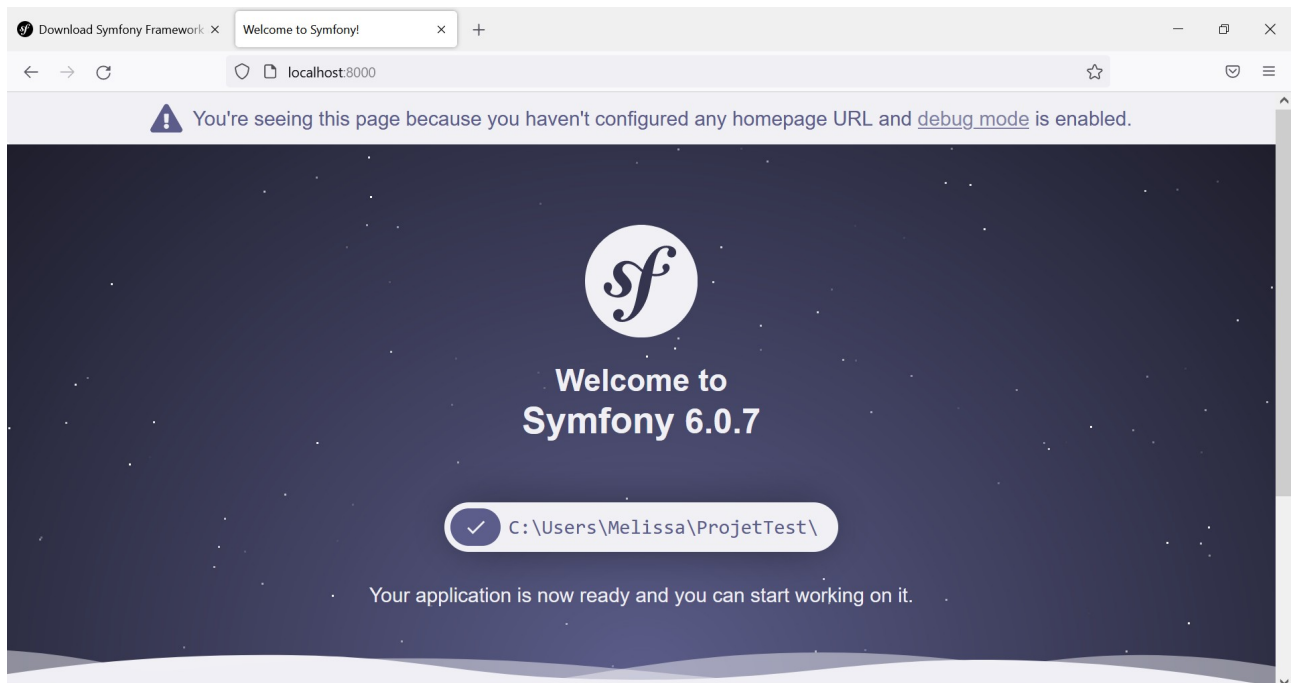
Tailing Web Server log file (C:\Users\Melissa\.symfony5\log\c1553bee7192a92497166c8da0f67088f3fc9807.log)
Tailing PHP CGI log file (C:\Users\Melissa\.symfony5\log\c1553bee7192a92497166c8da0f67088f3fc9807\79ca75f9e90b4126a5955a33ea6a41ec5e854698.log)

[WARNING] The local web server is optimized for local development and MUST never be used in a production setup.

[OK] Web server listening
The Web server is using PHP CGI 8.1.2
http://127.0.0.1:8000

[Web Server ] Apr 10 17:49:42 |DEBUG| PHP | Reloading PHP versions
[Web Server ] Apr 10 17:49:43 |DEBUG| PHP | Using PHP version 8.1.2 (from default version in $PATH)
[Web Server ] Apr 10 17:49:43 |INFO | PHP | listening path="C:\xampp\php\php-cgi.exe" php="8.1.2" port=64175
```

Le résultat du projet est visible à l'adresse : <http://127.0.0.1:8000> ou <http://localhost:8000>



V. Architecture de Symfony

A ce niveau là il est possible d'apercevoir divers répertoires dans le projet précédemment créé avec Symfony. Chacun d'entre eux ont un rôle bien précis :

`/bin`

Ce dossier contient les fichiers de commandes permettant, par exemple, de mettre à jour la base de données ou encore lancer les tests unitaires.

`/config`

Toute la configuration des packages, services et routes se fera dans ce dossier. Les fichiers de configuration sont par défaut en YAML. Cependant, il est tout à fait possible d'utiliser PHP ou XML.

`/public`

C'est le point d'entrée de l'application : chaque requête passe forcément par ce dossier et le fichier `index.php`.

`/src`

C'est le cœur du projet ! C'est principalement ici que la logique de l'application sera implémentée.

- [Controller] : Définition des points d'entrée de votre application.
- [Entity] : Définition de la structure de votre BDD (base de données) au travers de classes.
- [Repository] : Un Repository est toujours rattaché à une Entity, il permet de créer les fonctions qui iront faire une requête auprès de la table d'une Entity. Symfony utilise l'ORM Doctrine qui permet de créer nos requêtes SQL.

`/templates`

Les fichiers de template Twig ont comme format `monfichier.html.twig` et viennent rajouter quelques fonctionnalités au HTML classique :

- `{{ ... }}` : appel à une variable ou une fonction PHP, ou un template Twig parent.
- `{% ... %}` : permet de mettre en place une condition, une boucle ou un bloc HTML.
- `{# ... #}` : commentaires.

`/var`

Ici seront stockés le cache et les fichiers de log. Il est possible dans les fichiers de config de paramétrer la mise en cache et ce que l'on écrit dans les logs.

`/vendor`

Dans ce dossier, Composer y installe tous les paquets et librairies.

VI. Interaction avec une base de données

I. L'ORM Doctrine

Qu'est-ce que Doctrine ?

Doctrine est un mappeur relationnel objet (ORM). Dans la plupart des projets Symfony, l'ORM Doctrine est utilisé pour gérer les objets en base de données. L'ORM est composé de deux objets principaux :



- l'Entity Manager qui fournit des fonctions que l'on manipule pour créer, éditer, rechercher et supprimer nos objets.
- le Data Mapper qui s'occupe d'effectuer les requêtes auprès du système de gestion de base de données sélectionné. Le Data Mapper est une sorte de couche qui synchronise la donnée stockée en base avec les objets PHP.

Pour mapper des Entités, Doctrine est capable de comprendre les formats suivants :

- des annotations dans les commentaires des objets PHP ;
- des fichiers XML ;
- des fichiers YAML ;
- des fichiers de configuration PHP.

Ensuite, pour la manipulation des données, on indique à Doctrine avec des annotations quels sont les objets que nous souhaitons manipuler en base et quelles relations existent entre eux. Elles sont de trois types principalement :

- les relations 1 à 1 ;
- les relations 1 à n ;
- les relations n à n.

Pour bénéficier de l'ORM, il faut installer Doctrine via le pack orm Symfony, ainsi que le MakerBundle, qui aidera à générer du code :

```
PS C:\Users\Melissa\ProjetTest> composer require symfony/orm-pack
```

```
PS C:\Users\Melissa\ProjetTest> composer require --dev symfony/maker-bundle
```

II. Connexion à la BDD

Doctrine est capable de créer et mettre à jour une base de données. Les informations de connexion à la base de données sont stockées dans une variable d'environnement appelée `DATABASE_URL`. Pour le développement, il est possible de personnaliser ceci dans le fichier `.env` stocké à la racine du projet. En supposant que la base de données sera gérée avec `mysql` il faudra alors modifier la ligne suivante :

```
###> doctrine/doctrine-bundle ###
# Format described at https://www.doctrine-project.org/projects/doctrine-dbal/en/latest/reference/configuration.html#connecting-using-a-url
# IMPORTANT: You MUST configure your server version, either here or in config/packages/doctrine.yaml
#
# DATABASE_URL="sqlite:///kernel.project_dir/var/data.db"
# DATABASE_URL="mysql://db_user:db_password@127.0.0.1:3306/db_name?serverVersion=5.7&charset=utf8mb4"
# DATABASE_URL="postgresql://symfony:ChangeMe@127.0.0.1:5432/app?serverVersion=13&charset=utf8"
###< doctrine/doctrine-bundle ###
```

Maintenant les paramètres de connexion sont configurés, Doctrine peut créer la base de données `db_name` à partir de la commande suivante :

```
PS C:\Users\Melissa\ProjetTest> php bin/console doctrine:database:create
```

III. Les Entités

Dans Symfony une entité est une classe PHP, qui peut être connectée à une table de la base de données via l'ORM. Lorsqu'une entité est liée à une table via l'ORM, il y a en général un fichier "repository" associé. La commande ci-dessous permet de créer une entité ou modifier une entité déjà existante en saisissant son nom :

```
PS C:\Users\Melissa\ProjetTest> php bin/console make:entity
```

IV. Les Migrations

Le bundle `DoctrineMigrationsBundle` intègre la bibliothèque Doctrine Migrations dans les applications Symfony. Les migrations de base de données aident à versionner les modifications apportées au schéma de la base de données et à les appliquer de manière prévisible sur chaque serveur exécutant l'application.

Deux commandes essentielles :

- `php bin/console make:migration` → Elle permet de créer un fichier comportant toutes les modifications de la base de données.
- `php bin/console doctrine:migrations:migrate` → Celle-ci permet d'envoyer les fichiers afin d'exécuter les modifications sur la base de données

III. Installer Symfony sous Debian

I. Installation de php

Pour installer php entrer la commande suivante :

```
eLeve@DESKTOP-5FLFUDB:~$ sudo apt install wget php-cli php-zip unzip
```

II. Installation de Composer

Ensuite, la commande qui suit permet de télécharger l'installateur de composer depuis le site <https://getcomposer.org/installer> :

```
eLeve@DESKTOP-5FLFUDB:~$ wget -O composer-setup.php https://getcomposer.org/installer
--2022-04-13 14:02:58-- https://getcomposer.org/installer
Résolution de getcomposer.org (getcomposer.org)... 54.36.53.46, 2001:41d0:302:1100::8:104f
Connexion à getcomposer.org (getcomposer.org)[54.36.53.46]:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 57721 (56K) [application/octet-stream]
Sauvegarde en : « composer-setup.php »

composer-setup.php      100%[=====>] 56,37K  --.-KB/s  ds 0,02s
```

Composer est une application CLI à fichier unique qui peut être installée globalement ou dans le cadre du projet. L'installation globale nécessite les privilèges sudo. Pour installer Composer globalement afin qu'il soit disponible pour tous les utilisateurs, il faut placer le fichier dans un répertoire qui se trouve dans le système PATH . La commande suivante installe Composer dans le répertoire /usr/local/bin :

```
eLeve@DESKTOP-5FLFUDB:~$ sudo php composer-setup.php --install-dir=/usr/local/bin --filename=composer
All settings correct for using Composer
Downloading...

Composer (version 2.3.4) successfully installed to: /usr/local/bin/composer
```

III. Installation de Symfony

Le téléchargement du paquet permettant l'installation de Symfony 6 sous Debian se fait à partir de la commande suivante :

```
eLeve@DESKTOP-5FLFUDB:~$ echo 'deb [trusted=yes] https://repo.symfony.com/apt/ /' | sudo tee /etc/apt/sources.list.d/symfony-cli.list
[sudo] Mot de passe de eLeve :
deb [trusted=yes] https://repo.symfony.com/apt/ /
```

Ensuite on met à jour le système :

```
eLeve@DESKTOP-5FLFUDB:~$ sudo apt update
```

Enfin, l'installation de Symfony se fait à partir de la commande suivante :

```
eLeve@DESKTOP-5FLFUDB:~$ sudo apt install symfony-cli
```

IV. Débuter avec Symfony

I. Création d'un projet

La création d'un projet sous Debian peut aussi se faire de 2 façons. Avec Symfony il faut entrer la commande suivante :

```
eLeve@DESKTOP-5FLFUDB:~$ symfony new ProjetTest
* Creating a new Symfony project with Composer
WARNING: Unable to find Composer, downloading one. It is recommended to install Composer yourself at https://getcomposer.org/download/
(running /home/eLeve/.symfony5/composer/composer.phar create-project symfony/skeleton /home/eLeve/ProjetTest --no-interaction)

* Setting up the project under Git version control
(running git init /home/eLeve/ProjetTest)
```

Avec Composer il faut créer un dossier, se déplacer à l'intérieur puis entrer la commande suivante :

```
eLeve@DESKTOP-5FLFUDB:~$ mkdir ProjetTest
eLeve@DESKTOP-5FLFUDB:~$ cd ProjetTest/
eLeve@DESKTOP-5FLFUDB:~/ProjetTest$ composer require nesbot/carbon
```

Enfin, il faut entrer la commande **composer require nesbot/carbon** :

```
eLeve@DESKTOP-5FLFUDB:~$ mkdir ProjetTest
eLeve@DESKTOP-5FLFUDB:~$ cd ProjetTest/
eLeve@DESKTOP-5FLFUDB:~/ProjetTest$ composer require nesbot/carbon
Info from https://repo.packagist.org: #StandWithUkraine
Using version ^2.57 for nesbot/carbon
./composer.json has been created
Running composer update nesbot/carbon
Loading composer repositories with package information
Updating dependencies
Lock file operations: 6 installs, 0 updates, 0 removals
- Locking nesbot/carbon (2.57.0)
- Locking symfony/deprecation-contracts (v2.5.1)
- Locking symfony/polyfill-mbstring (v1.25.0)
- Locking symfony/polyfill-php80 (v1.25.0)
- Locking symfony/translation (v5.4.7)
- Locking symfony/translation-contracts (v2.5.1)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 6 installs, 0 updates, 0 removals
- Downloading symfony/translation-contracts (v2.5.1)
- Downloading symfony/polyfill-php80 (v1.25.0)
- Downloading symfony/polyfill-mbstring (v1.25.0)
- Downloading symfony/deprecation-contracts (v2.5.1)
- Downloading symfony/translation (v5.4.7)
- Downloading nesbot/carbon (2.57.0)
- Installing symfony/translation-contracts (v2.5.1): Extracting archive
- Installing symfony/polyfill-php80 (v1.25.0): Extracting archive
- Installing symfony/polyfill-mbstring (v1.25.0): Extracting archive
- Installing symfony/deprecation-contracts (v2.5.1): Extracting archive
- Installing symfony/translation (v5.4.7): Extracting archive
- Installing nesbot/carbon (2.57.0): Extracting archive
3 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating autoload files
6 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
```

II. Démarrer Symfony

Avant de démarrer l'application Symfony, il est nécessaire de vérifier si le projet possède tous les pré-requis nécessaires. Cette vérification s'effectue avec la commande **symfony check:requirements**:

```
eLeve@DESKTOP-5FLFUDB:~/ProjetTest$ symfony check:requirements

Symfony Requirements Checker
~~~~~

> PHP is using the following php.ini file:
/etc/php/7.3/cli/php.ini

> Checking Symfony requirements:

.....E.....WW...W.....

[ERROR]
Your system is not ready to run Symfony projects

Fix the following mandatory requirements
~~~~~

* simplexml_import_dom() must be available
  > Install and enable the SimpleXML extension.

Optional recommendations to improve your setup
~~~~~

* PHP-DOM and PHP-XML modules should be installed
  > Install and enable the PHP-DOM and the PHP-XML modules.

* mb_strlen() should be available
  > Install and enable the mbstring extension.

* intl extension should be available
  > Install and enable the intl extension (used for validators).

Note The command console can use a different php.ini file
~~~~~ than the one used by your web server.
Please check that both the console and the web server
```


Avant de lancer le projet il faut installer **dom**. Pour installer **dom**, il faut entrer la commande suivante :

```
eLeve@DESKTOP-5FLFUDB:~$ sudo apt-get install php-xml
[sudo] Mot de passe de eleve : █
```

Une fois que **dom** a été installé, le projet est prêt à être lancé :

```
eLeve@DESKTOP-5FLFUDB:~/ProjetTest$ symfony check:requirements

Symfony Requirements Checker
~~~~~

> PHP is using the following php.ini file:
/etc/php/7.3/cli/php.ini

> Checking Symfony requirements:

.....W...W.....

[OK]
Your system is ready to run Symfony projects

Optional recommendations to improve your setup
~~~~~

* mb_strlen() should be available
  > Install and enable the mbstring extension.

* intl extension should be available
  > Install and enable the intl extension (used for validators).

Note The command console can use a different php.ini file
~~~~~
than the one used by your web server.
Please check that both the console and the web server
are using the same PHP version and configuration.
```

Le projet peut donc être lancé à partir de la commande **symfony server:start** :

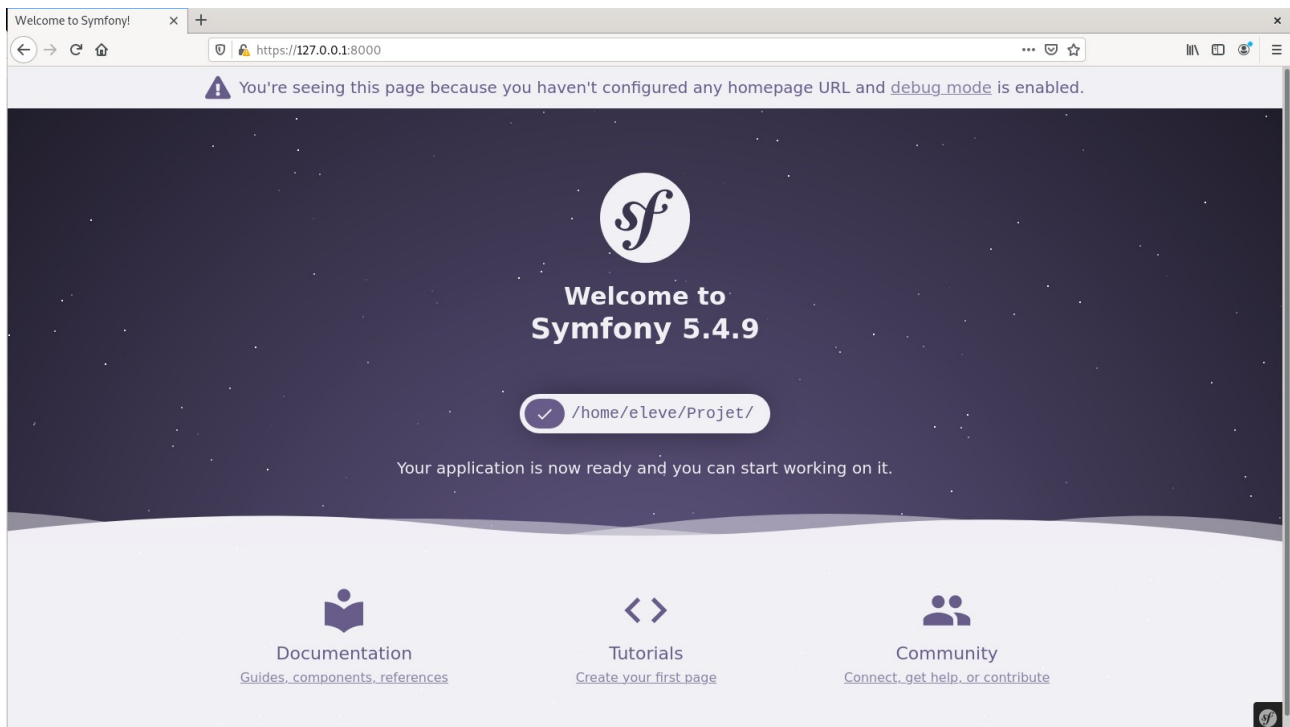
```
eLeve@DESKTOP-5FLFUDB:~/Projet$ symfony server:start
Following Web Server log file (/home/eLeve/.symfony5/log/238acd6bf7e7504ae74cb0b437ffdc7d1f55ce28.log)
Following PHP log file (/home/eLeve/.symfony5/log/238acd6bf7e7504ae74cb0b437ffdc7d1f55ce28/7daf403c7589f4927632ed3b6af762a992f09b78.log)

[WARNING] The local web server is optimized for local development and MUST never be used in a production setup.

[OK] Web server listening
The Web server is using PHP CLI 7.3.31
https://127.0.0.1:8080

[Web Server ] Jun 23 14:19:36 |DEBUG | PHP | Reloading PHP versions
[Web Server ] Jun 23 14:19:36 |DEBUG | PHP | Using PHP version 7.3.31 (from default version in $PATH)
[Web Server ] Jun 23 14:19:36 |INFO | PHP | Listening path="/usr/bin/php7.3" php="7.3.31" port=46595
[Web Server ] Jun 23 14:19:56 |DEBUG | SERVER | GET (404) / ?= "127.0.0.1"
[PHP ] [Thu Jun 23 14:19:56 2022] 127.0.0.1:53188 [404]: /
[Web Server ] Jun 23 14:19:56 |INFO | SERVER | GET (200) / wdt/7645ef ?p="127.0.0.1"
[PHP ] [Thu Jun 23 14:19:56 2022] 127.0.0.1:53190 [200]: / wdt/7645ef
```

Résultat obtenu :



V. Interaction avec une base de données

L'interaction de Symfony avec une base de données sous Debian se fait de la même façon que sous Windows.

IV. Démarrage du projet d'irrigation connectée

Maintenant que l'ensemble des pré-requis nécessaires ont été installé et vérifié, on peut à présent passer au démarrage du site web d'irrigation connectée sous Symfony. Pour cela il faudra dans un premier décompresser l'une des deux archives du projet.

I. Archives du projet

La première archive nommée **IrrigationComplet.zip** et la seconde nommée **IrrigationIncomplet.zip** sont toutes deux des dossiers .zip du projet avec l'un complet et l'autre par conséquent incomplet. Si vous avez du temps et que vous ne souhaitez pas entrer une liste de commande, je vous conseille d'utiliser l'archive **IrrigationComplet.zip** qui possède en son sein les dossiers **var** et **vendor** très lourd donc très lent à charger.

Si vous sélectionner la deuxième option, le procédé est le suivant. Tout d'abord, il faut décompresser l'archive. Ensuite, il faut ouvrir une fenêtre PowerShell au sein de ce projet. Pour cela, maintenez la touche **Maj** et effectuer un **clic droit**. Ensuite il sera nécessaire de rentrer une liste de commande afin de constituer notre dossier **var** et **vendor**. Ces dossier sont primordiaux pour le bon fonctionnement du site. Dans l'ordre qui suit, on entrera donc les commandes suivantes:

- composer install
- composer require logger
- composer require annotations
- composer require twig
- composer require easycorp/easyadmin-bundle
- composer require symfony/orm-pack
- composer require symfonycasts/verify-email-bundle
- composer require symfony/password-hasher
- composer require symfonycasts/reset-password-bundle
- composer require symfony/mailer
- composer require symfony/google-mailer
- symfony check:security

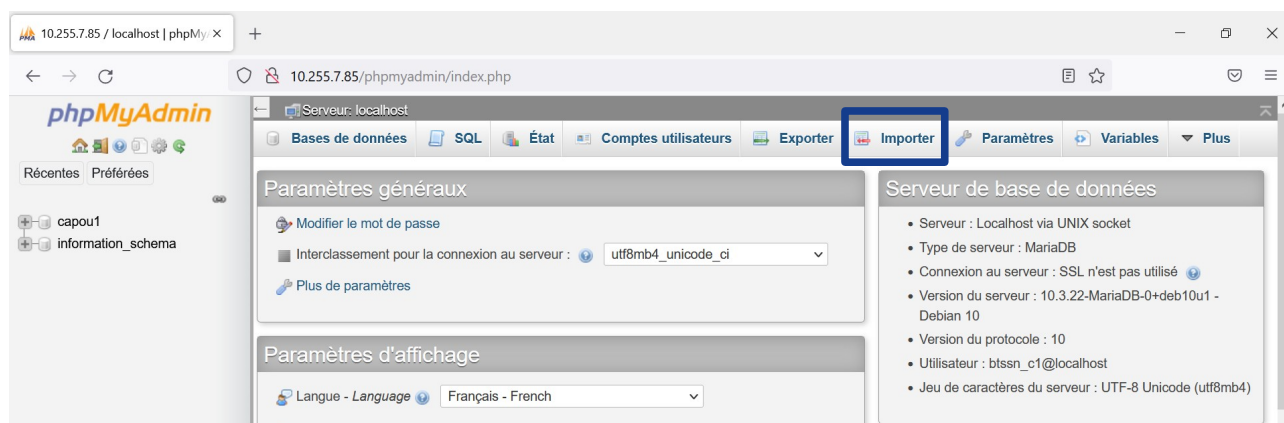
Attention : Une connexion à internet est requise afin que composer puisse récupérer les paquets requis sur le web.

Après ces installations, on pourra ainsi démarrer le serveur web avec la commande vue précédemment **symfony server:start**. Il est possible que le site affiche une erreur SQL, dans ce cas il suffira de modifier dans le fichier **.env.local** l'adresse de la base de données avec une adresse de BDD valide. Par défaut l'adresse est 10.255.7.85. Une fois réglé, on sera alors redirigé vers la page d'accueil du site.

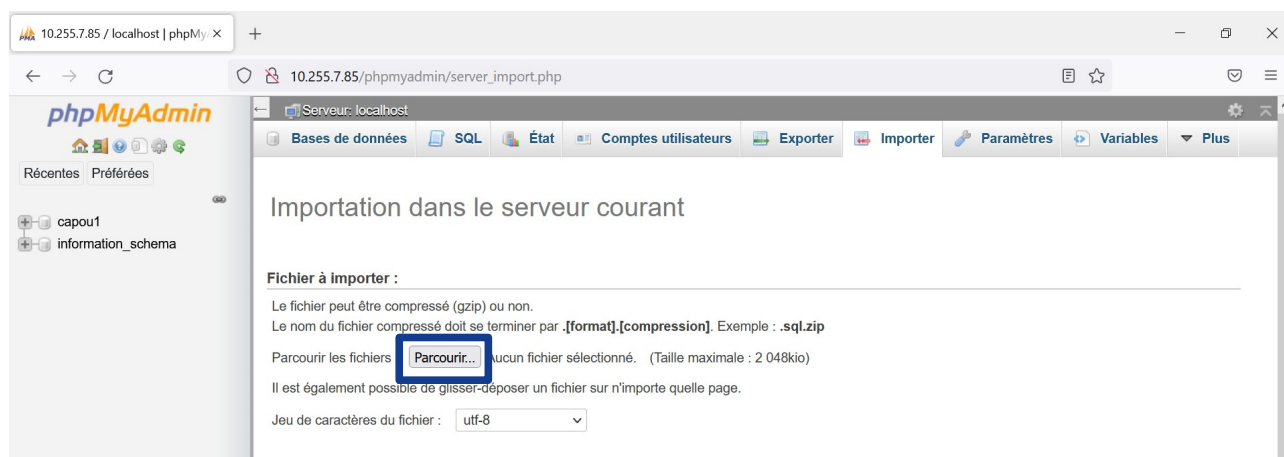
II. Mise en place de la BDD

Pour mettre en place la base de données, deux méthodes s'offrent à nous, la première étant d'accéder à la base de données présente sur le serveur 10.255.7.85 (en local ou 90.63.226.129 en distant) ou de créer une nouvelle base de données sur un serveur.

La seconde méthode est de créer sa base de données en local avec l'aide de Xampp ou de Wamp. Il suffit pour cela de renseigner l'adresse local de la BDD dans le fichier .env.local ainsi que ces identifiants. Afin de créer la BDD et les tables en son sein, il faudra simplement importer le fichier .sql dans phpMyAdmin. Pour cela on cliquera sur « Importer » :



puis « Parcourir » :



Ensuite on sélectionnera le fichier capou1.sql :



puis on cliquera sur « Exécuter » :

Autres options :

☒ Activer la vérification des clés étrangères

Format :

SQL

Options spécifiques au format :

Mode de compatibilité SQL : NONE

☒ Ne pas utiliser AUTO_INCREMENT pour la valeur zéro

Exécuter

Console de requêtes SQL

Dans les deux cas de création de BDD, des données seront d'ores et déjà présente en leur sein.

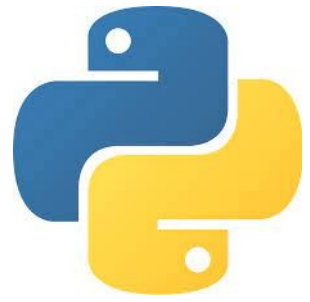
III. Connexion au site

En raison de la présence de données dans la BDD, on peut donc s'authentifier autant en administrateur qu'en technicien. Les identifiants de connexion de l'administrateur sont « admin@test.fr » avec comme mot de passe « admin ». Pour chaque compte technicien le mot de passe est « technicien » et trois adresses mail sont disponibles « raymond.mahe@marie.fr », « oceane.francois@chauveau.com » « mcarre@sfr.fr » (ces adresses sont factices et ne possèdent pas de boîte mail). La connexion redirige alors chaque utilisateur vers son interface personnelle.

En conclusion, pour développer le système de fonctionnement du site web, veuillez-vous référer à la documentation [guideUtilisationIrrigationConnecteeSymfony.pdf](#) .

IV. Modification de l'adresse Mail

L'adresse mail renseignée au sein du projet permet de générer les mails. Lorsque vous reprenez le projet il est nécessaire de modifier celle-ci. Pour ce faire, ouvrez le fichier MailerController.php situé dans le dossier /src/Controller.



Installation du Framework Django sous Windows et Debian

[23/06/2022]

V. Installer Django sous Windows

I. Installation de Python

I. Vérification

Avant de procéder à l'installation vous pouvez vérifier que Python n'est pas installé en entrant la commande **python** au sein du powerShell. Si python est installé vous obtiendrez le résultat suivant :

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

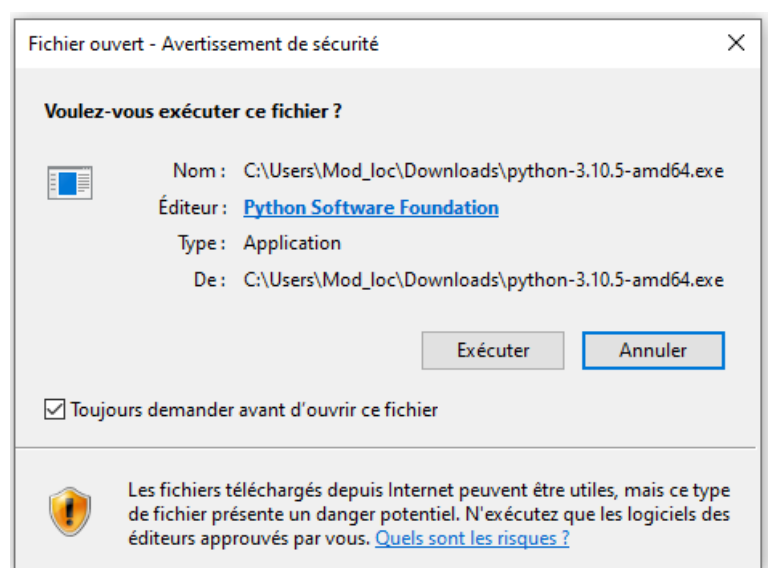
Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\Mod_loc> python
Python 3.6.2 (v3.6.2:5fd33b5, Jul 8 2017, 04:57:36) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

II. Téléchargement de Python

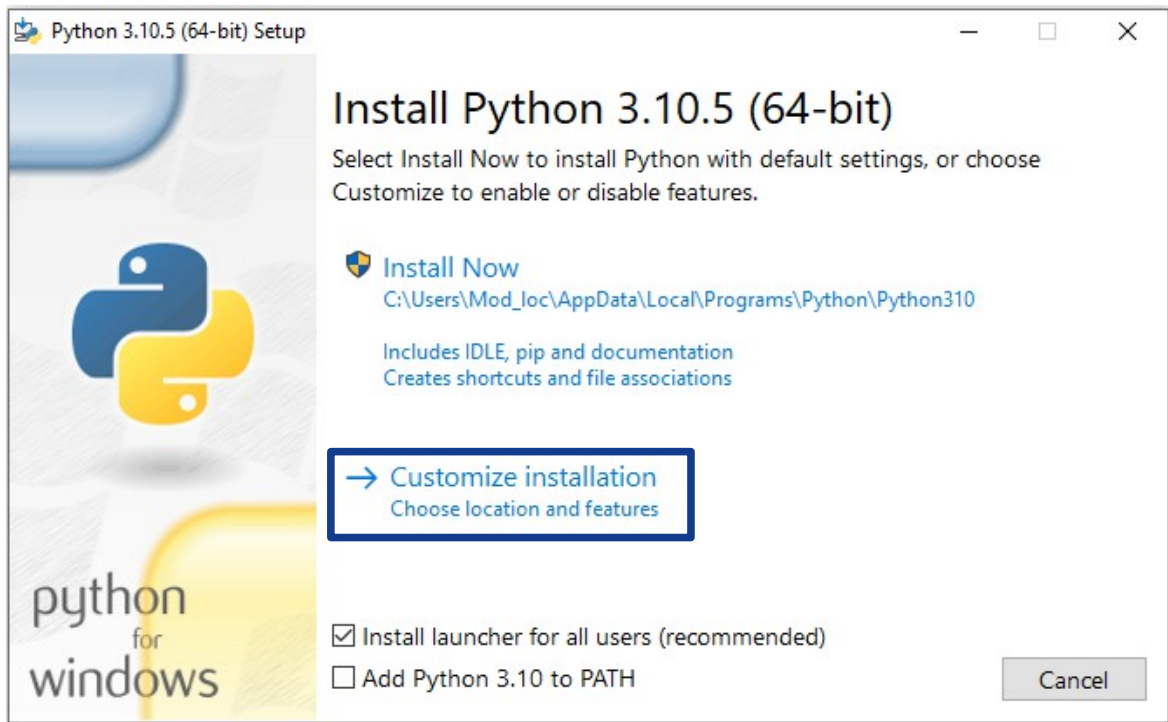
Cependant, si Python n'est pas installé, celui-ci est téléchargeable à l'URL suivante :

<https://www.python.org/downloads/release/python-3105>

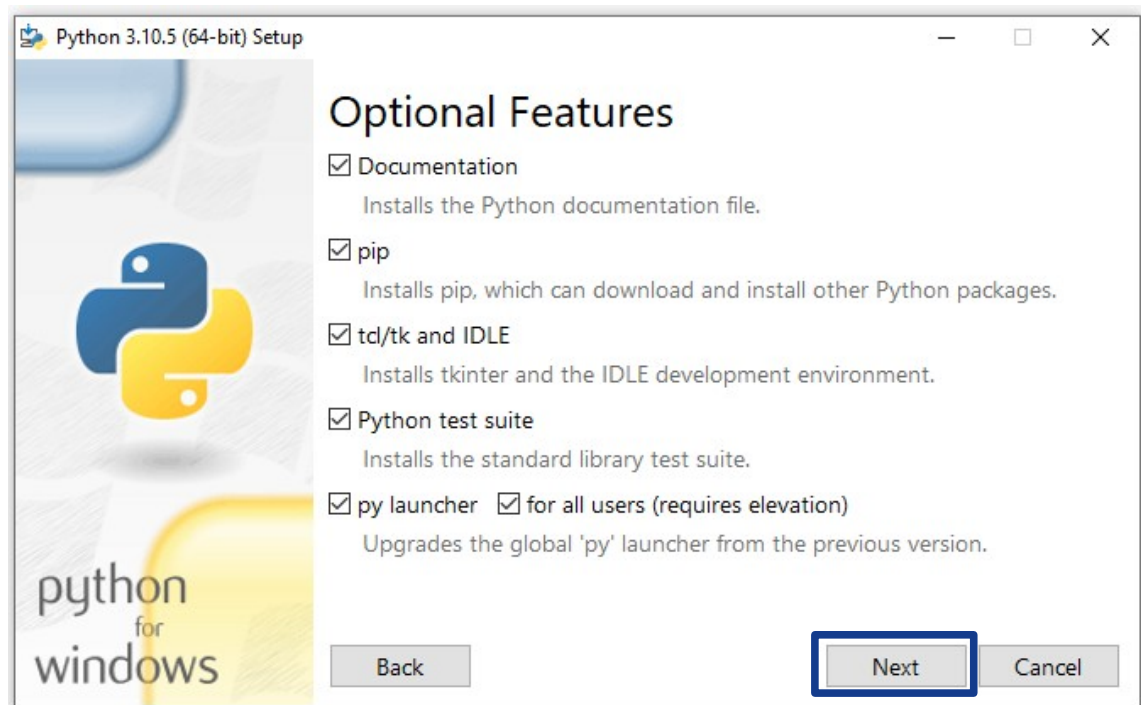


III. Installation de Python

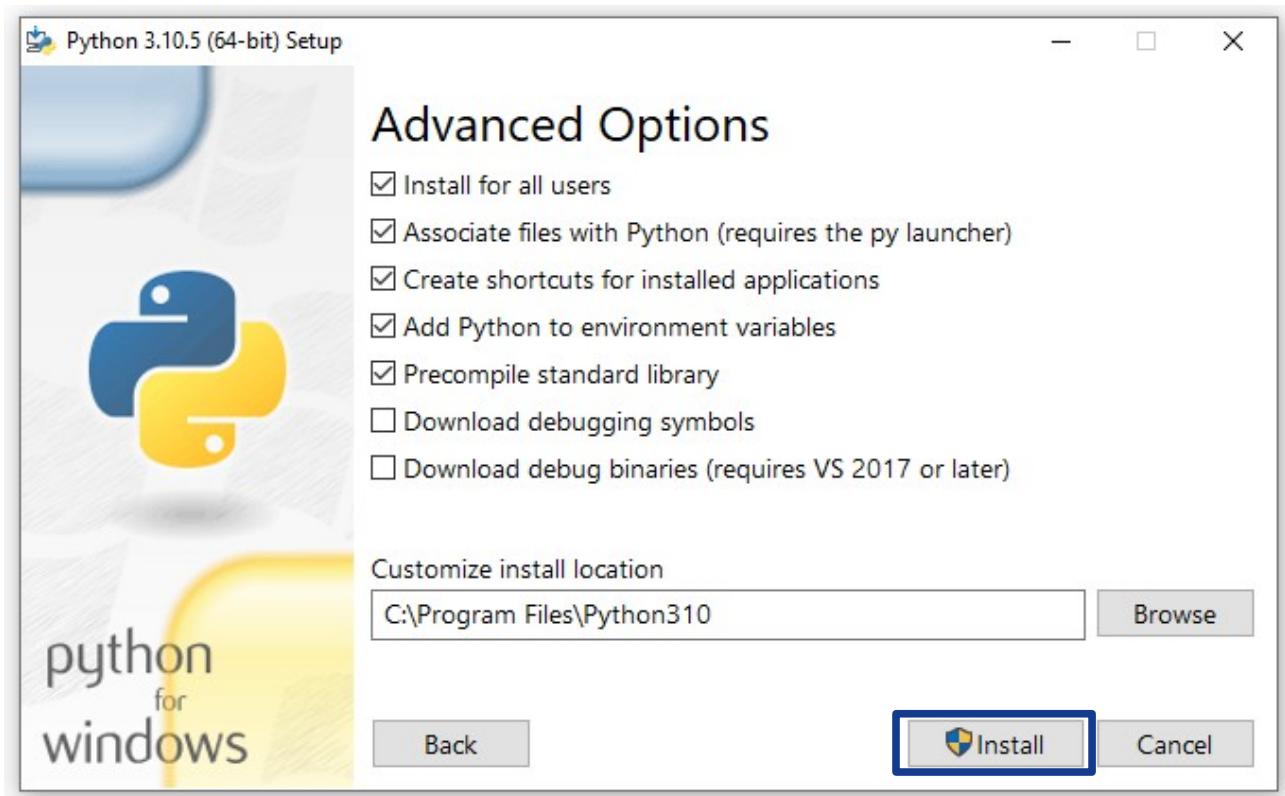
Une fois le fichier exécutable lancé, vous aurez la possibilité de sélectionner le type d'installation. Ici, nous allons sélectionner le deuxième choix :



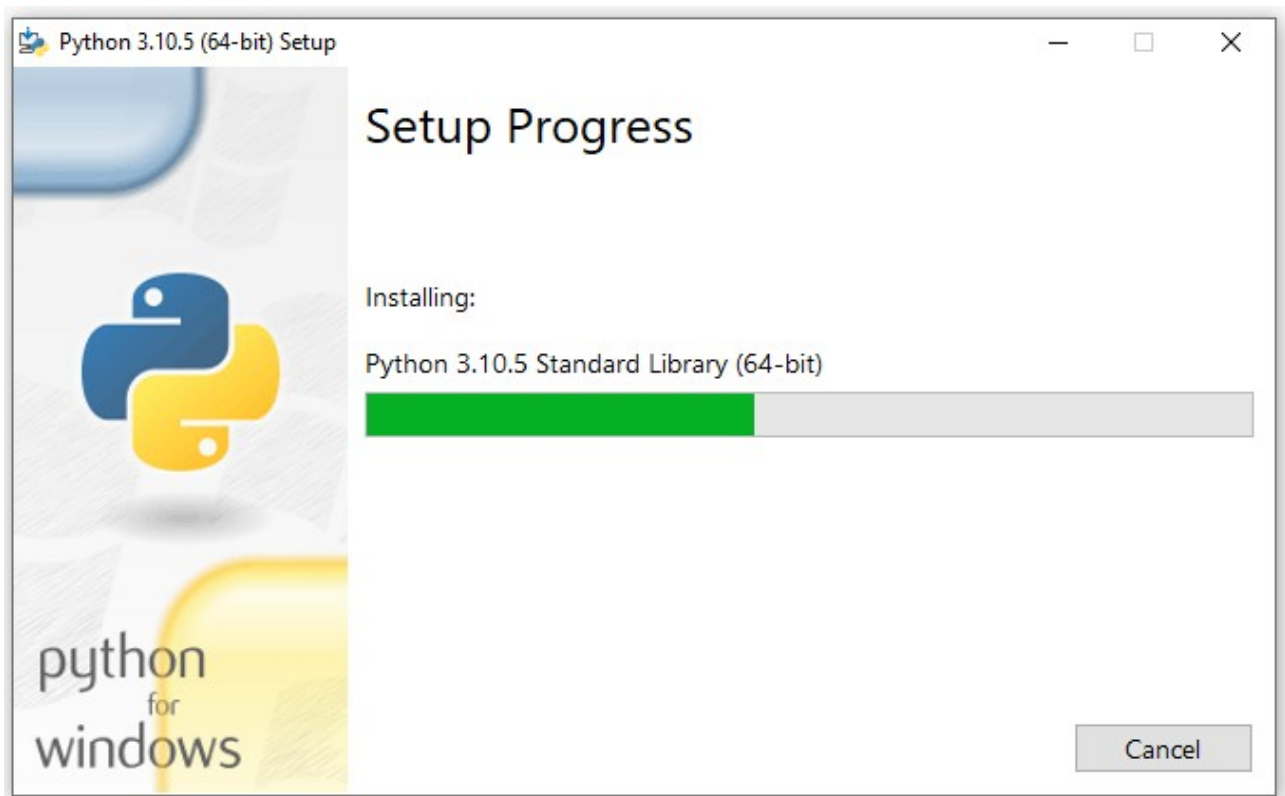
Ici, nous allons conserver les paramètres par défaut et cliquer sur **Next**



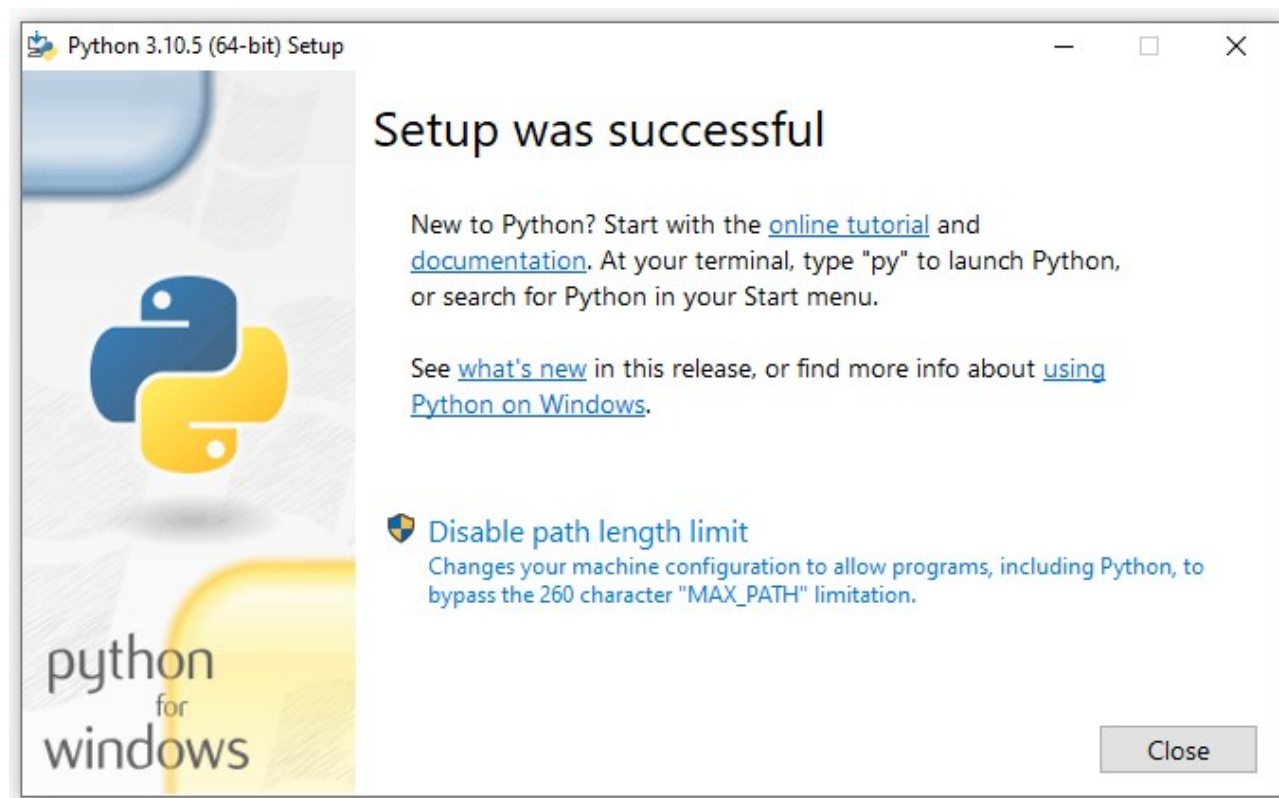
Parmi les options, sélectionner les choix suivant puis cliquer sur « Install » :



L'installation peut prendre un certain temps, patientez :

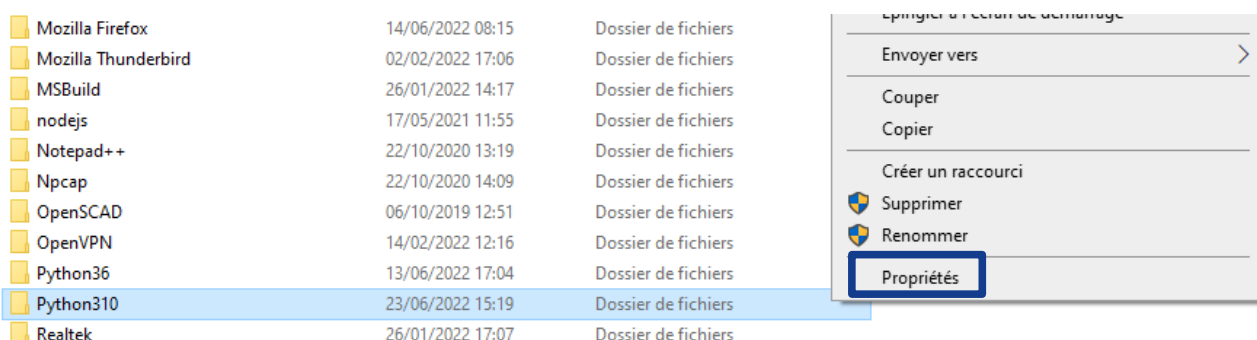


A présent, vous pouvez mettre fin à l'installation en fermant la fenêtre :

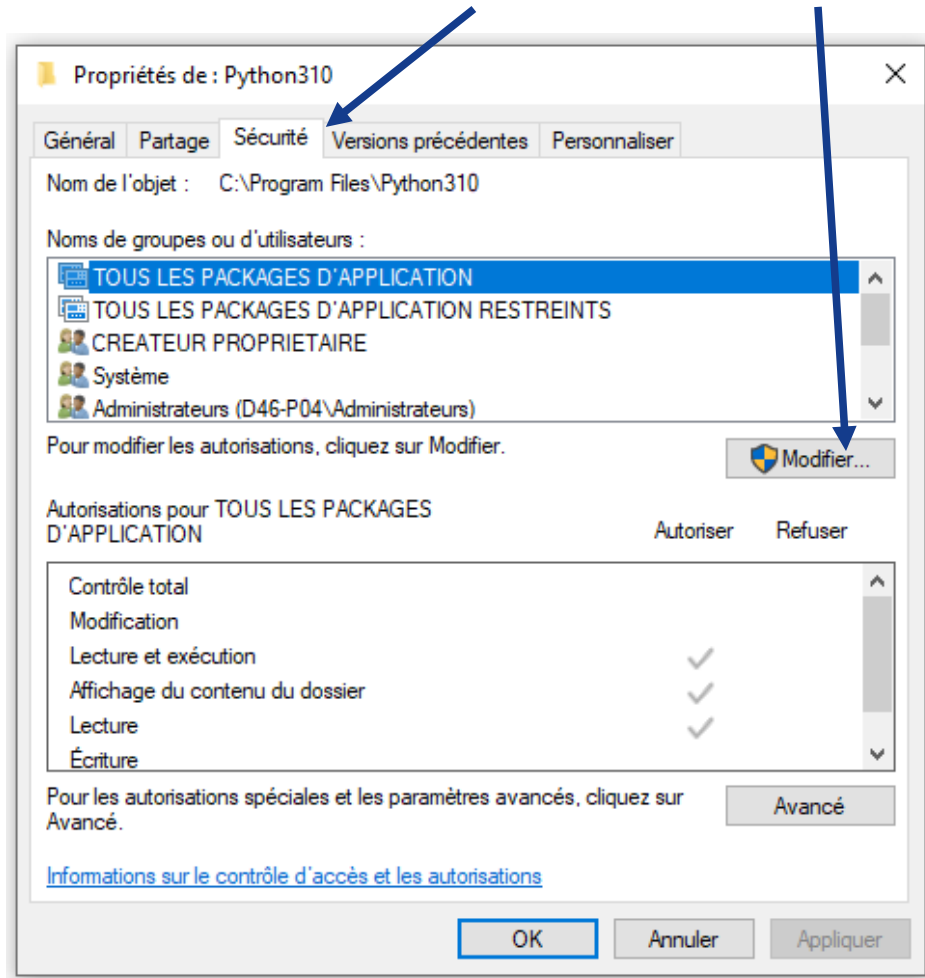


IV. Droits de sécurité

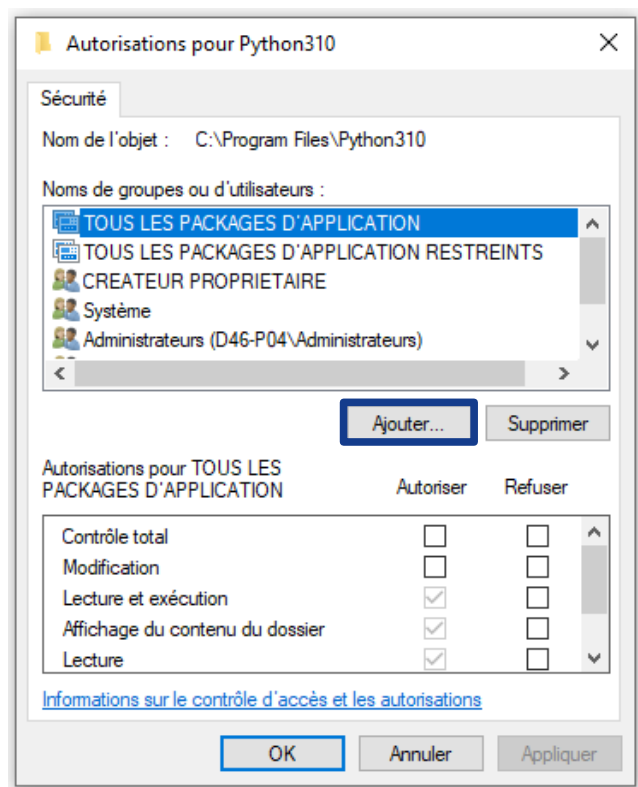
Afin de posséder un fonctionnement optimal, il est nécessaire de modifier les droits de sécurité. Pour ce faire, rendez vous dans le répertoire « C:\Program Files ». Ensuite, effectuer un clic droit sur le répertoire Python310 et cliquer sur propriétés :



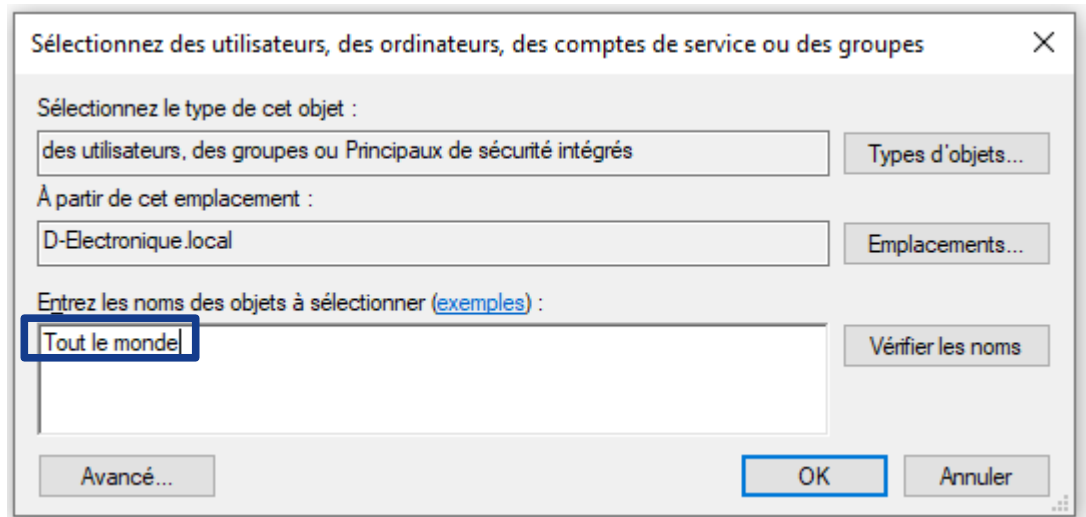
Ensuite, sélectionner l'onglet « Sécurité » et cliquer sur « Modifier » :



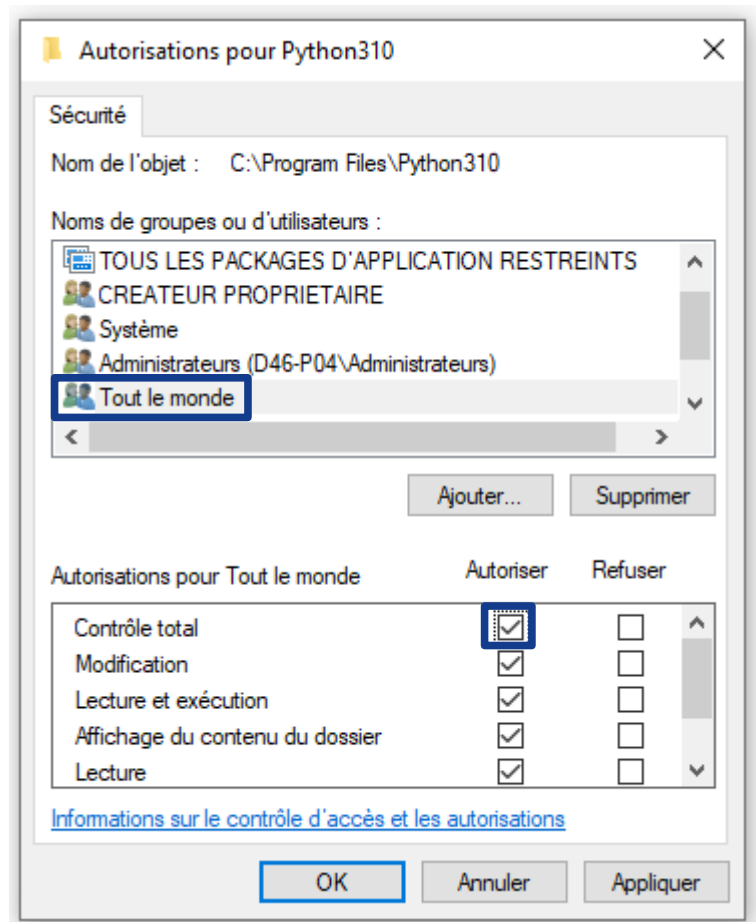
Cliquer sur « Ajouter » :



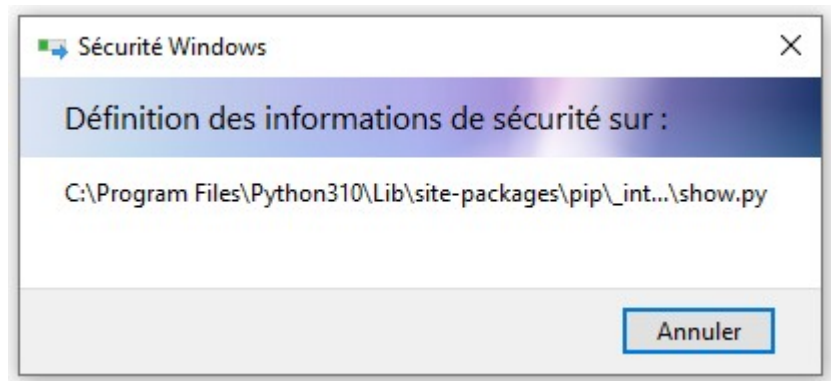
Puis ajouter « Tous le monde » dans le champ texte et valider :



Ensuite il faut accorder tous les droits à « Tout le monde » :



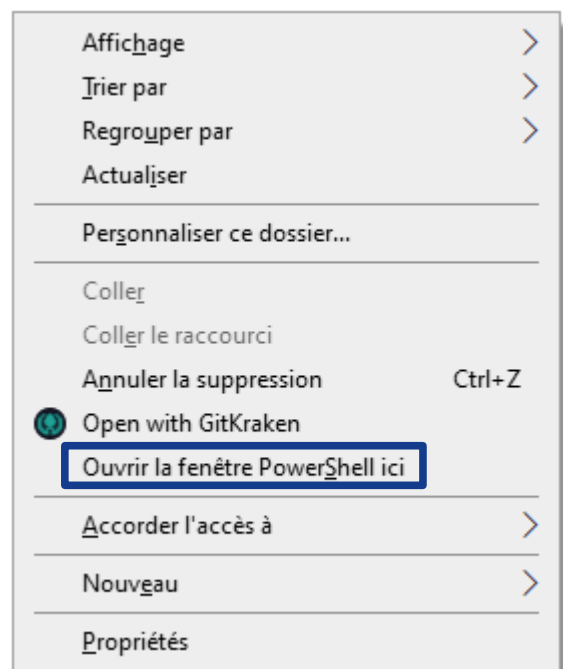
Les droits de sécurité se modifient :



Enfin, vous pouvez valider et fermer la fenêtre.

V. Installation des bibliothèques

L'installation des bibliothèques se déroule dans le répertoire « C:\Program Files\Python310\Scripts ». Une fois situé dans ce répertoire ouvrez une fenêtre PowerShell en maintenant la touche « Maj » et en réalisant un clic droit. Sur la fenêtre qui apparaît, sélectionner « Ouvrir une fenêtre de commande ici ».



Suite à ça une fenêtre PowerShell s'ouvre.

Tout d'abord, il est nécessaire de mettre à jour les outils permettant d'installer les bibliothèques. Cela se fait avec la commande suivante :

```
Windows PowerShell
PS C:\Program Files\Python310\Scripts> python -m pip install -U pip setuptools
```

Si la mise à jour se déroule correctement vous obtiendrez le résultat suivant :

```
Windows PowerShell
PS C:\Program Files\Python310\Scripts> python -m pip install -U pip setuptools
Requirement already satisfied: pip in c:\program files\python310\lib\site-packages (22.0.4)
Collecting pip
  Using cached pip-22.1.2-py3-none-any.whl (2.1 MB)
Requirement already satisfied: setuptools in c:\program files\python310\lib\site-packages (58.1.0)
Collecting setuptools
  Using cached setuptools-62.6.0-py3-none-any.whl (1.2 MB)
Installing collected packages: setuptools, pip
  Attempting uninstall: setuptools
    Found existing installation: setuptools 58.1.0
    Uninstalling setuptools-58.1.0:
      Successfully uninstalled setuptools-58.1.0
```

A présent, nous avons la possibilité d'installer les bibliothèques en exécutant les commandes :

- `python -m pip install matplotlib`
- `python -m pip install pillow`
- `python -m pip install pygame`
- `python -m pip install OpenCVPython`
- `python -m pip install Serial`
- `python -m pip install pySerial`

```
Windows PowerShell
PS C:\Program Files\Python310\Scripts> python -m pip install matplotlib
Collecting matplotlib
  Downloading matplotlib-3.5.2-cp310-cp310-win_amd64.whl (7.2 MB)
----- 7.2/7.2 MB 32.9 MB/s eta 0:00:00
Requirement already satisfied: pillow>=6.2.0 in c:\users\mod_loc\appdata\roaming\python\python310\site-packages (from matplotlib) (9.1.1)
Collecting numpy>=1.17
  Downloading numpy-1.23.0-cp310-cp310-win_amd64.whl (14.6 MB)
----- 14.6/14.6 MB 40.9 MB/s eta 0:00:00
Collecting fonttools>=4.22.0
  Downloading fonttools-4.33.3-py3-none-any.whl (930 kB)
----- 930.9/930.9 KB 57.5 MB/s eta 0:00:00
Collecting kiwisolver>=1.0.1
  Downloading kiwisolver-1.4.3-cp310-cp310-win_amd64.whl (55 kB)
----- 55.3/55.3 KB ? eta 0:00:00
Collecting packaging>=20.0
  Using cached packaging-21.3-py3-none-any.whl (40 kB)
Collecting python-dateutil>=2.7
  Using cached python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
Collecting pyparsing>=2.2.1
  Downloading pyparsing-3.0.9-py3-none-any.whl (98 kB)
----- 98.3/98.3 KB ? eta 0:00:00
Collecting cycler>=0.10
  Using cached cycler-0.11.0-py3-none-any.whl (6.4 kB)
Collecting six>=1.5
  Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: six, pyparsing, numpy, kiwisolver, fonttools, cycler, python-dateutil, packaging, matplotlib
Successfully installed cycler-0.11.0 fonttools-4.33.3 kiwisolver-1.4.3 matplotlib-3.5.2
```

II. Installation de Django

Qu'est-ce que Django ?

Django est un framework écrit en Python qui permet de mettre en place une application Web tout en réduisant considérablement le temps de développement. Ce framework prend en charge de nombreuses parties du développement web. Cela nous permet donc de nous concentrer sur l'écriture de l'application.



Avant d'installer Django vous avez la possibilité de vérifier si celui-ci n'est pas existant avec la commande suivante :

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\Mod_loc> py -m django --version
4.0.5
PS C:\Users\Mod_loc>
```

info : Si Django n'est pas installé vous obtiendrez une erreur indiquant « Aucun module nommé django ».

Dans le cas où Django ne serait pas existant, l'installation de ce cadre de travail peut se faire en entrant la ligne de commande suivante :

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

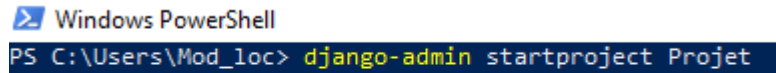
Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\Mod_loc> py -m pip install Django
Collecting Django
  Downloading Django-4.0.5-py3-none-any.whl (8.0 MB)
  ----- 8.0/8.0 MB 12.9 MB/s eta 0:00:00
Collecting tzdata
  Downloading tzdata-2022.1-py2.py3-none-any.whl (339 kB)
  ----- 339.5/339.5 KB 20.6 MB/s eta 0:00:00
Collecting asgiref<4,>=3.4.1
  Downloading asgiref-3.5.2-py3-none-any.whl (22 kB)
Collecting sqlparse>=0.2.2
  Downloading sqlparse-0.4.2-py3-none-any.whl (42 kB)
  ----- 42.3/42.3 KB 2.0 MB/s eta 0:00:00
Installing collected packages: tzdata, sqlparse, asgiref, Django
Successfully installed Django-4.0.5 asgiref-3.5.2 sqlparse-0.4.2 tzdata-2022.1
```


III. Débuter avec Django

I. Création d'un projet

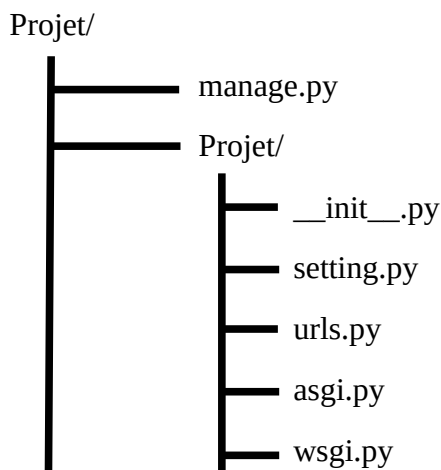
Un projet peut-être créé en entrant la commande suivante au sein d'une fenêtre PowerShell:



```
Windows PowerShell
PS C:\Users\Mod_loc> django-admin startproject Projet
```

info : Il est fortement conseillé d'éviter de nommer les projets avec des noms appartenant à des composants intégrés à Python ou Django. Concrètement, cela signifie que vous devez éviter d'utiliser des noms comme django (qui pourrait entrer en conflit avec Django lui-même) ou test (qui lui pourrait entrer en conflit avec un paquet intégré de Python).

II. Architecture de Django



Projet/ Ce répertoire contient tout les dossiers et fichiers du projet. Il se comporte comme un conteneur.

Manage.py Ce fichier est un utilitaire en ligne de commande qui permet d'interagir avec le projet Django

Projet/.__init__.py Fichier permettant de paramétrer et configurer le projet Django

Projet/urls.py Ce fichier permet de déclarer les URL qui seront utilisées au sein du projet

Projet/asgi.py Ce fichier est un point d'entrée pour les serveurs Web compatible ASGI

Projet/wsgi.py Ce fichier est un point d'entrée pour les serveurs Web compatible WSGI

III. Démarrer Django

Pour lancer le projet, Django possède son propre serveur Web. Il s'agit d'un serveur léger qui est uniquement écrit en Python. La commande permettant de lancer le serveur est la suivante :

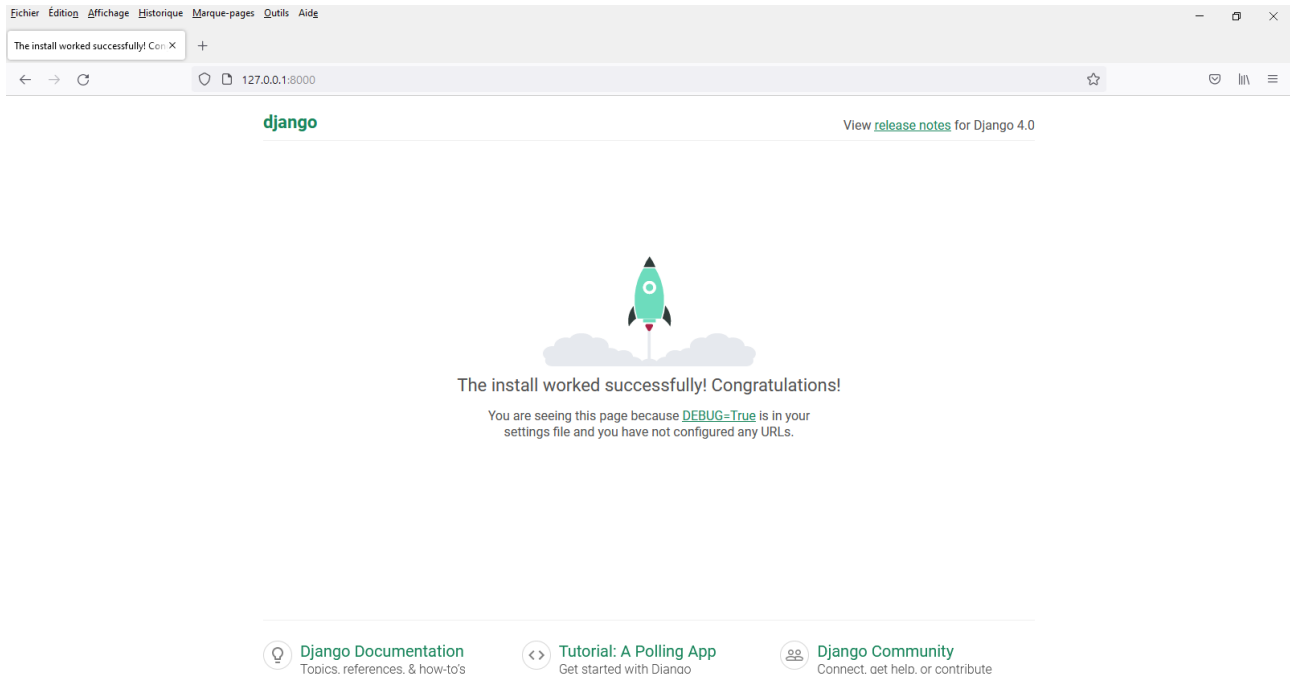
```
Windows PowerShell
PS C:\Users\Mod_loc\Projet> py manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

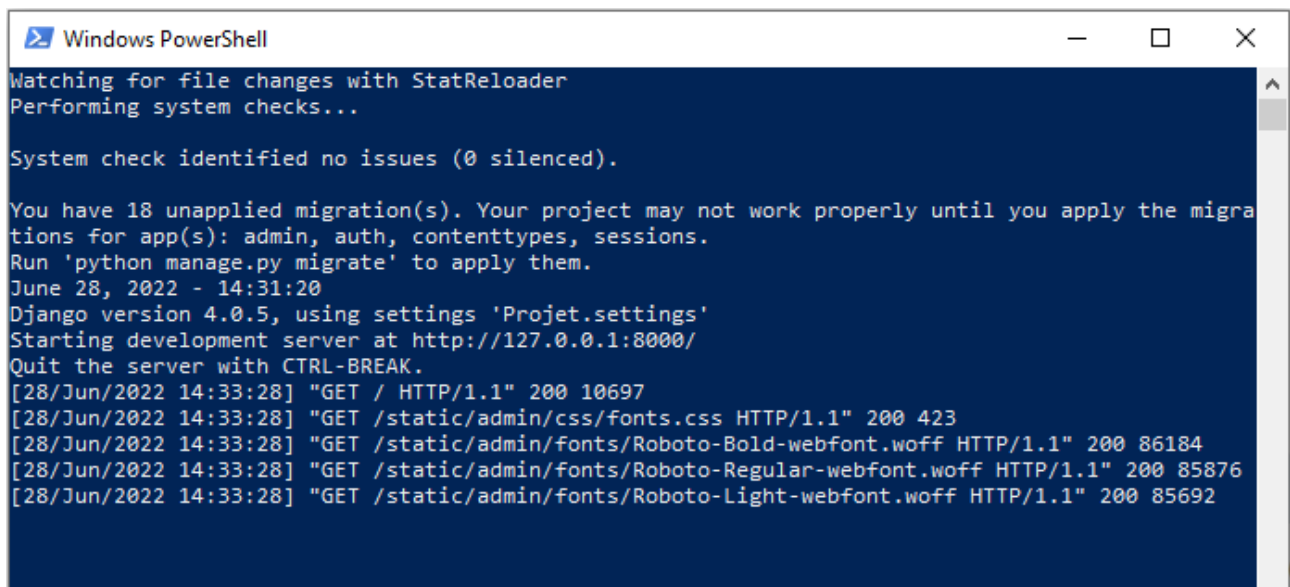
You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.

Run 'python manage.py migrate' to apply them.
June 28, 2022 - 14:31:20
Django version 4.0.5, using settings 'Projet.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Le résultat du projet est visible à l'adresse : <http://127.0.0.1:8000>



Au sein du PowerShell on peut visualiser l'accès au site. La visualisation de cette accès ce fait via les requêtes HTTP GET :



```
Windows PowerShell
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
June 28, 2022 - 14:31:20
Django version 4.0.5, using settings 'Projet.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[28/Jun/2022 14:33:28] "GET / HTTP/1.1" 200 10697
[28/Jun/2022 14:33:28] "GET /static/admin/css/fonts.css HTTP/1.1" 200 423
[28/Jun/2022 14:33:28] "GET /static/admin/fonts/Roboto-Bold-webfont.woff HTTP/1.1" 200 86184
[28/Jun/2022 14:33:28] "GET /static/admin/fonts/Roboto-Regular-webfont.woff HTTP/1.1" 200 85876
[28/Jun/2022 14:33:28] "GET /static/admin/fonts/Roboto-Light-webfont.woff HTTP/1.1" 200 85692
```

IV. Installation Xampp

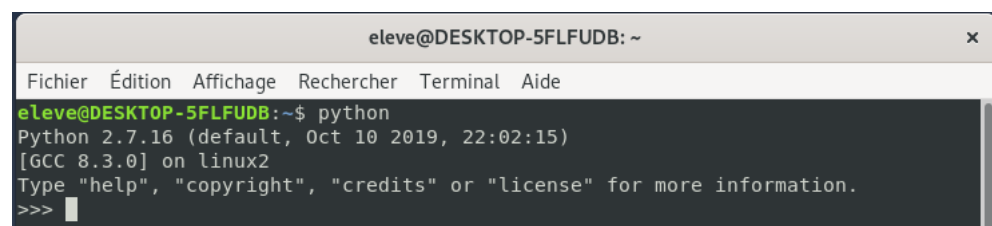
Le procédé d'installation avec le framework Python étant identique à celui effectué précédemment lors de l'installation de Symfony, veuillez vous référer au chapitre adéquate.

VI. Installer Django sous Debian

I. Installation de Python

I. Vérification

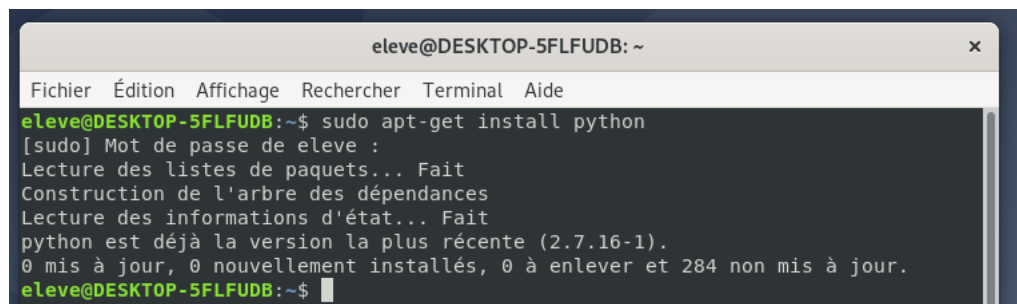
Avant de procéder à l'installation de Python, vous pouvez vérifier si celui-ci n'est pas déjà existant en entrant la commande suivante au sein d'un terminal :

A terminal window titled 'eleve@DESKTOP-5FLFUDB: ~' with a menu bar (Fichier, Édition, Affichage, Rechercher, Terminal, Aide). The command 'python' has been executed, resulting in the output: 'Python 2.7.16 (default, Oct 10 2019, 22:02:15) [GCC 8.3.0] on linux2. Type "help", "copyright", "credits" or "license" for more information.' followed by a prompt '>>>' and a cursor.

```
eleve@DESKTOP-5FLFUDB: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
eleve@DESKTOP-5FLFUDB:~$ python  
Python 2.7.16 (default, Oct 10 2019, 22:02:15)  
[GCC 8.3.0] on linux2  
Type "help", "copyright", "credits" or "license" for more information.  
>>> █
```

II. Installation de Python

Pour installer Python il faut entrer la commande suivante :

A terminal window titled 'eleve@DESKTOP-5FLFUDB: ~' with a menu bar (Fichier, Édition, Affichage, Rechercher, Terminal, Aide). The command 'sudo apt-get install python' has been executed. The output shows the process of checking the package list, building the dependency tree, and confirming that Python 2.7.16-1 is already the latest version installed. The prompt returns to 'eleve@DESKTOP-5FLFUDB:~\$' with a cursor.

```
eleve@DESKTOP-5FLFUDB:~$ sudo apt-get install python  
[sudo] Mot de passe de eleve :  
Lecture des listes de paquets... Fait  
Construction de l'arbre des dépendances  
Lecture des informations d'état... Fait  
python est déjà la version la plus récente (2.7.16-1).  
0 mis à jour, 0 nouvellement installés, 0 à enlever et 284 non mis à jour.  
eleve@DESKTOP-5FLFUDB:~$ █
```

III. Installation des bibliothèques

Pour installer des bibliothèques il faut entrer la commande suivante :

```
Fichier Édition Affichage Rechercher Terminal Aide
eLeve@DESKTOP-SFLFUDB:~$ sudo apt-get install python3-pip
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  dh-python libpython3-dev libpython3.7 libpython3.7-dev libpython3.7-minimal
  libpython3.7-stdlib python3-asn1crypto python3-cffi-backend python3-crypto
  python3-cryptography python3-dev python3-entrypoints python3-keyring
  python3-keyrings.alt python3-secretstorage python3-setuptools python3-wheel
  python3.7 python3.7-dev python3.7-minimal
Paquets suggérés :
  python-crypto-doc python-cryptography-doc python3-cryptography-vectors
  libkf5wallet-bin gir1.2-gnomekeyring-1.0 python-secretstorage-doc
  python-setuptools-doc python3.7-venv python3.7-doc
Les NOUVEAUX paquets suivants seront installés :
  dh-python libpython3-dev libpython3.7-dev python3-asn1crypto
  python3-cffi-backend python3-crypto python3-cryptography python3-dev
  python3-entrypoints python3-keyring python3-keyrings.alt python3-pip
  python3-secretstorage python3-setuptools python3-wheel python3.7-dev
Les paquets suivants seront mis à jour :
  libpython3.7 libpython3.7-minimal libpython3.7-stdlib python3.7
  python3.7-minimal
5 mis à jour, 16 nouvellement installés, 0 à enlever et 279 non mis à jour.
Il est nécessaire de prendre 50,2 Mo/56,1 Mo dans les archives.
Après cette opération, 90,8 Mo d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [0/n] 0
```

II. Installation de Django

Enfin, pour installer Django Il faut utiliser pip3 :

```
eLeve@DESKTOP-SFLFUDB: ~
Fichier Édition Affichage Rechercher Terminal Aide
eLeve@DESKTOP-SFLFUDB:~$ pip3 install Django
Collecting Django
  Downloading https://files.pythonhosted.org/packages/c3/68/b63abc009adee25c1bd266b3740e173eab656608cf21641594a37e02cf57/Django-3.2.13-py3-none-any.whl (7.9MB)
    100% |#####| 7.9MB 266kB/s
Collecting asgiref<4,>=3.3.2 (from Django)
  Downloading https://files.pythonhosted.org/packages/af/6d/ea3a5c3027c3f14b0321cd4f7e594c776ebe64e4b927432ca6917512a4f7/asgiref-3.5.2-py3-none-any.whl
Collecting sqlparse>=0.2.2 (from Django)
  Downloading https://files.pythonhosted.org/packages/05/40/d836d55fb3f467243ee839ab7b814822fda522cd395fa41e282684e71ee5/sqlparse-0.4.2-py3-none-any.whl (42kB)
    100% |#####| 51kB 11.3MB/s
Collecting pytz (from Django)
  Downloading https://files.pythonhosted.org/packages/60/2e/dec1cc18c51b8df33c7c4d0a321b084cf38e1733b98f9d15018880fb4970/pytz-2022.1-py2.py3-none-any.whl (503kB)
    100% |#####| 512kB 3.7MB/s
Collecting typing-extensions; python version < "3.8" (from asgiref<4,>=3.3.2->Django)
  Downloading https://files.pythonhosted.org/packages/75/e1/932e06004039dd670c9d5e1df0cd606bf46e29a28e65d5bb28e894ea29c9/typing_extensions-4.2.0-py3-none-any.whl
Installing collected packages: typing-extensions, asgiref, sqlparse, pytz, Django
Successfully installed Django-3.2.13 asgiref-3.5.2 pytz-2022.1 sqlparse-0.4.2 typing-extensions-4.2.0
eLeve@DESKTOP-SFLFUDB:~$
```

VII. Démarrage du projet d'Irrigation Connectée

I. Mise en place de la BDD

Il y a deux manières de mettre en place la base de données:

La première est d'utiliser la base de données présente au lycée il n'y a donc aucun changement à faire lors de l'installation du site car la base de données à laquelle va se connecter le site est automatiquement celle qui se trouve au lycée Antoine Bourdelle. La seconde manière est de créer une base de données de zéro. Pour cela il faut démarrer Xampp et une fois arrivé au Control Panel (car Xampp n'est disponible qu'en Anglais et Allemand) il faut démarrer les modules Apache et MySql en cliquant sur « start ».

Une fois les modules démarrés, cliquez sur le bouton « admin » du module MySql puis cliquez sur « phpMyAdmin », puis sur « nouvelle base de données ». Mettez le nom de la BDD puis appuyer sur « créer » et choisissez où vous voulez dans votre ordinateur. Enfin il faut lier la base de données au site, pour cela il faut ouvrir le dossier du site (nommer src) aller dans le dossier « projetCapou » puis dans le fichier « settings.py » puis descendre jusqu'à la ligne 75 et remplacer « capou2 » par le nom de la BDD nouvellement créée « btssn_c2 » par le nom d'utilisateur que vous avez créé sur votre BDD. De même pour « b0urdelle », l'adresse IP et le port.

II. Changement de l'adresse mail

Pour changer l'adresse mail d'où les mails vont être envoyés (mails pour les mots de passe par exemple) il faut aussi aller dans le fichier « settings.py » tout à la fin du fichier. L'adresse à changer est celle de « EMAIL_HOST_USER » ainsi que le mot de passe juste en dessous selon l'adresse mail.

III. Démarrage du site

Pour démarrer le site il faut aller sur un terminal, le plus simple est d'utiliser celui de Visual Studio Code, et d'écrire dans celui-ci `python manage.py runserver` en spécifiant l'adresse voulue pour Windows et `python3 manage.py runserver` et l'adresse sous Debian.

IV. Premier lancement du site

Lors du premier lancement du site il vous faudra créer un compte administrateur. Pour cela en premier lieu il faut se rendre à la page d'accueil du site en tapant dans la barre de recherche l'adresse IP que vous avez attribuée ci-dessus et vous serez directement envoyé sur la page d'accueil du site. Depuis cette page cliquez sur le bouton « connexion » en haut à droite de la page puis connecter vous en administrateur par défaut « IDENTIFIANT » et « MOT DE PASSE ». Puis depuis ce compte créer l'administrateur principal.



Installation de la partie Matérielle sous Windows et Debian

[23/06/2022]

VIII. Installation partie Matérielle

Cette partie détail les étapes nécessaires à l'installation des piquets. Ceux-ci seront déployés au sein de l'exploitation agricole du Lycée Capou.

I. ESPTools et pySerial

Afin de posséder un fonctionnement optimal, il est nécessaire d'installer quelques prérequis :

- Python 3 (peu importe la version)
- pip (gestionnaire de paquets)

Si vous avez suivi l'installation de Python effectuée précédemment, le gestionnaire de dépendance PIP est déjà installé. Il vous suffit donc d'entrer la commande suivante afin d'installer la librairie :

```
PS C:\Program Files\Python310\Scripts> python -m pip install pySerial esptool
```

info : cette librairie doit être installée en procédant de la même façon que pour les librairies précédemment installée.

II. Gestion des droits

Afin de pouvoir utiliser les ports séries, il est nécessaire de modifier les droits qui y sont associés. Pour se faire, il est nécessaire de passer en mode super utilisateur et d'entrer les commandes suivantes :

```
# usermod -aG dialout <user>
# usermod -aG uucp <user>
# usermod -aG tty <user>
```

Info : Si l'un de ces groupes n'est pas disponible, il y en aura automatiquement un qui permettra de vous octroyer les droits d'utilisations des ports série.

Une fois cela fait, il sera nécessaire de redémarrer l'ordinateur afin que tout ces changements soient pris en compte.

III. Flasher une nouvelle adresse MAC sur un ESP32

Tout d'abord, il faut récupérer l'un des divers Vendor ID (les 3 premiers octets de l'adresse MAC) correspondant à l'entreprise Espressif Inc. (entreprise qui produit les ESP32). Ce Vendor ID permet de déterminer si il s'agit bien d'un ESP32 présent au sein d'un réseau Wi-Fi par exemple.

Exemple : **0C:B8:15** est un Vendor ID.

Ensuite, le reste peut être généré aléatoirement.

Pour flasher la nouvelle adresse MAC il faut entrer la commande suivante :

```
$ espefuse.py -p <portSérie> -b 115200 burn_custom_mac 0C:B8:15:XX:XX:XX
```

L'utilitaire vous demandera une confirmation car l'opération est irréversible. Vous pouvez confirmez en tapant « BURN » en toutes lettres.

IV. Téléverser le code sur les cartes

Pour éviter d'installer de nombreuses dépendances, au sein du projet vous disposez d'un script en Python nommé flashCode.py. Ce script permet d'automatiser le téléversement du code. Afin de téléverser les fichiers sur l'ESP32 il vous suffit d'exécuter le script et de sélectionner le port série que vous souhaitez. La suite se réalisera de manière automatique. Cette commande permet d'exécuter le script Python :

```
[salim@t450s TestFlash]$ python flashCode.py
Ports série disponibles:
1) /dev/ttyACM0
Choisissez votre port série: 1
```

Résultat obtenu :

```
Running stub...
Stub running...
Changing baud rate to 921600
Changed.
Configuring flash size...
Flash will be erased from 0x00001000 to 0x00007fff...
Flash will be erased from 0x00008000 to 0x00008fff...
Flash will be erased from 0x00010000 to 0x00048fff...
Compressed 25968 bytes to 15134...
Wrote 25968 bytes (15134 compressed) at 0x00001000 in 0.6 seconds (effective 335
.8 kbit/s)...
Hash of data verified.
Compressed 3072 bytes to 103...
Wrote 3072 bytes (103 compressed) at 0x00008000 in 0.1 seconds (effective 344.3
kbit/s)...
Hash of data verified.
Compressed 232432 bytes to 122432...
Wrote 232432 bytes (122432 compressed) at 0x00010000 in 2.1 seconds (effective 8
80.6 kbit/s)...
Hash of data verified.
Leaving...
Hard resetting via RTS pin...
```


V. Assemblage du piquet

L'assemblage du piquet se déroule de la manière suivante. Tout d'abord, il faut visser la carte mère dans son compartiment. Ensuite, il faut rajouter les éléments suivants :

