# Symfony 4

# Phpstorm con Symfony (instalación de plugin)

# Instalación

Primero asegúrese de usar PHP 7.1 o superior y tener Composer instalado.

Para crear un nuevo proyecto:

composer create-project symfony/skeleton my-project

Una vez creado el proyecto, ingresamos al mismo para instalar un servidor propio del proyecto:

composer require server

Para levantar el proyecto: php bin/console server:run

# Rutas y controladores

<https://symfony.com/doc/current/page_creation.html>

Instalamos el makerBundle:

composer require maker

Comando para crear un controlador:

php bin/console make:controller Product

# Templates (twig)

Instalar Assets:

composer require asset

# Creacion de bundles

# Bases de datos y doctrine ORM

<https://symfony.com/doc/current/doctrine.html>

<http://symfony.com/doc/current/validation.html>

# Formularios

<https://symfony.com/doc/current/forms.html>

Para crear un archive:

php bin/console make:form Usuario

# Pasos para un nuevo Proyecto:

**Crear un nuevo proyecto:** composer create-project symfony/skeleton my-project

**Instalamos un servidor para el proyecto:** composer require server

**Instalamos las rutas por Anotaciones:** composer require annotations

**Instalamos el componente Twig:** composer require twig

**Instalamos el componente Forms:** composer require symfony/form

**Instalamos el componente Maker para generar controladores, entidades, forms, entre otros mediante consola:** composer require maker

**Instalamos el componente doctrine para trabajar con base de datos:** composer require symfony/orm-pack

**Configuramos la conexión de la bdd en el archivo .env para crear la base de datos:**

php bin/console doctrine:database:create

**Para crear una clase entidad:** php bin/console make:entity

**Migraciones de los cambios hechos en la base de datos:**

php bin/console make:migration

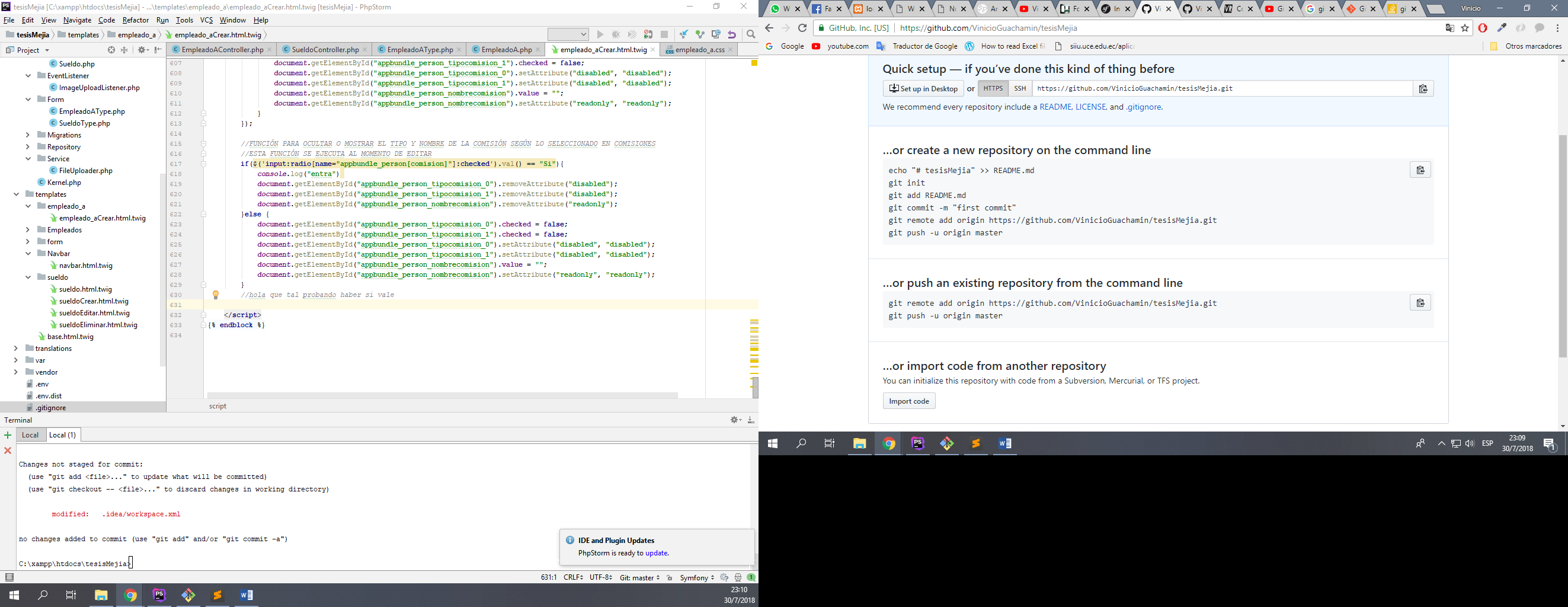
php bin/console doctrine:migrations:migrate

**Si prefiere agregar nuevas propiedades manualmente, puede generar los métodos getter & setter:**

php bin/console make:entity –regenerate

# Usando git:

Cuando creamos un repositorio nuevo en github, debemos seguir estas instrucciones.



|  |  |
| --- | --- |
| **COMANDO** | **DESCRIPCIÓN** |
| echo "# tesisMejia" >> README.md | Para crear un archivo Readme |
| git init | Para inicializar un proyecto con git |
| git add . | Para agregar todos los archivos modificados |
| git commit -m “Mi primer commit” | Para hacer commit de los archivos modificados |
| git remote add origin https://github.com/VinicioGuachamin/tesisMejia.git | Para conectar el proyecto creado localmente con el repositorio creado en github |
| git push -u origin master | Para subir todos los cambios realizados localmente al repositorio |
| git pull origin master | Para sincronizar lo que está en el repositorio de github con el proyecto que se encuentra localmente |
| git checkout origin/master -f  git checkout master -f | Puede verificar diferentes ramas con fuerza, luego volver a master |
|  |  |