# Apuntes PHP – Master UDEMY Víctor Robles

Programas interesantes: cygwin64 Terminal, Camtasia Studio, Camtasia Recorder, Vegas Pro

20211109

Todo código de PHP tiene que ir dentro de esta carpeta:

F:\wamp64\www

F:\wamp64\www\master-php\aprendiendo-php\01-hola-mundo

Al poner index.php hace que sea el archivo que primero se cargue en el navegador.

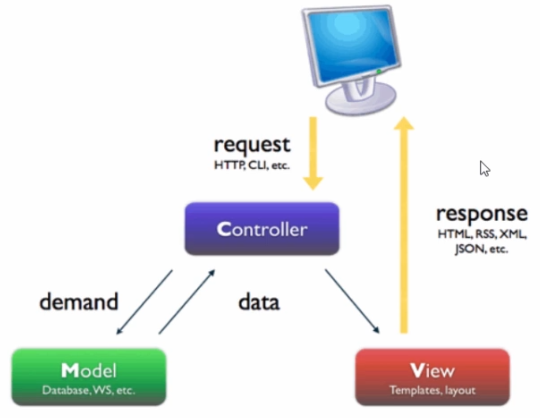
20220117

VÍDEO 310: ¿QUE ES UN FRAMEWORK?:

**Un framework lo que te viene a decir es: "construye tu aplicación, no tu herramienta"**

VÍDEO 312. ¿POR QUE USAR UN FRAMEWORK PARA PHP? - MOTIVO 2:

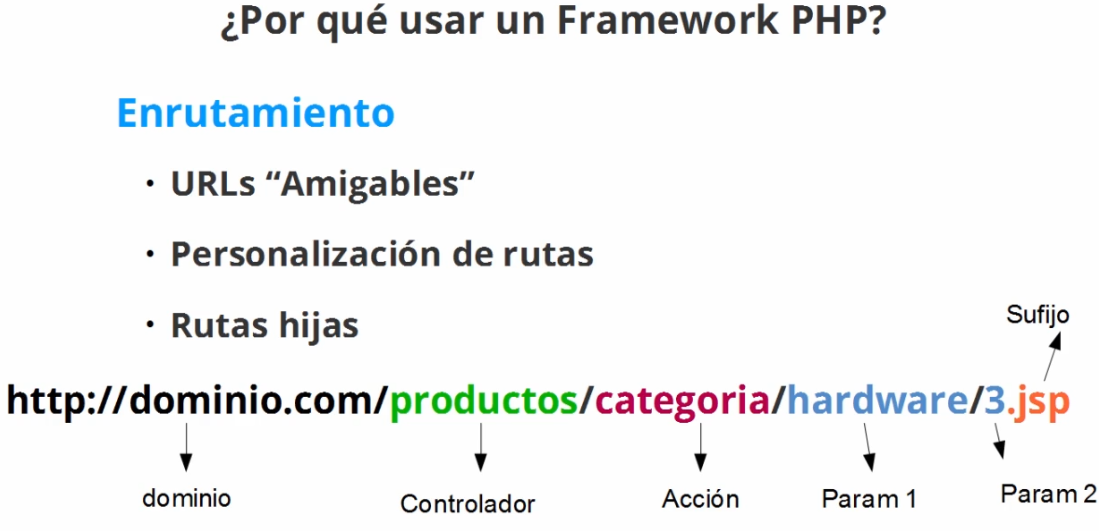
MVC es un patrón de diseño de 3 capas: modelo, vista y controlador



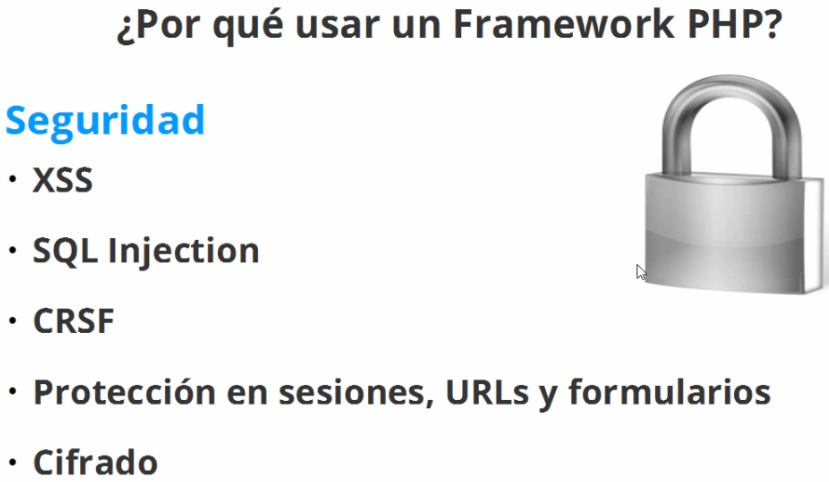
313. ¿POR QUE USAR UN FRAMEWORK PARA PHP? - MOTIVO 3:

1. Independencia del SGBD (PostgreSQL, SQLite, MySQL, SQLServer) que usemos  
2. Query Builders y ORMs (Objet Relational Mapping) (Active Record, Doctrine, etc)  
3. Optimización de consultas  
4. Persistencia

315. ¿POR QUE USAR UN FRAMEWORK PARA PHP? - MOTIVO 5:



316. ¿POR QUE USAR UN FRAMEWORK PARA PHP? - MOTIVO 6:



318. DESVENTAJA DE LOS FRAMEWORKS:

1. Curva de aprendizaje  
2. Desconocimiento del núcleo  
3. A más abstracción, menos rendimiento

320: INSTALAR LARAVEL:

Laravel instalación:  
<https://laravel.com/docs/5.7/installation>

CMD:

***>*** dir***>*** F:

***F:\wamp64-2\www\master-php>***   
composer create-project laravel/laravel aprendiendo-laravel "6.\*" --prefer-dist

***F:\wamp64-2\www\master-php>***   
composer create-project laravel/laravel aprendiendo-laravel-v8 "8.\*" --prefer-dist

***F:\wamp64-2\www\master-php>***   
composer create-project laravel/laravel aprendiendo-laravel-v5 "5.6.\*" --prefer-dist

***F:\wamp64-2\www\master-php\aprendiendo-laravel>***   
composer install --ignore-platform-reqs

322. CREAR UN HOST VIRTUAL:

Añadir mi host virtual en:  
C:\Windows\System32\drivers\etc\hosts

Añadir el proyecto en:  
F:\wamp64-2\bin\apache\apache2.4.51\conf\extra\httpd-vhosts.conf

URL del proyecto: <http://aprendiendo-laravel.com.devel/>

327. CONSOLA ARTISAN:

F:\wamp64-2\www\master-php\aprendiendo-laravel-v5> php artisan help  
F:\wamp64-2\www\master-php\aprendiendo-laravel-v5> php artisan list  
F:\wamp64-2\www\master-php\aprendiendo-laravel-v5> php artisan route:list

336. CONTROLADORES BÁSICOS

F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan make:controller PeliculaController

337. CONTROLADORES RESOURCE

F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan make:controller UsuarioController –resource

340. MIDDLEWARES

F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan make:middleware TestYear

343. MIGRACIONES

F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan make:migration create\_usuarios\_table --table=usuarios

F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan migrate  
F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan migrate:rollback  
F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan migrate:refresh

345. SEEDERS

F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan make:seed frutas\_seed  
F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan db:seed --class=frutas\_seed

346. LISTAR DATOS

F:\wamp64-2\www\master-php\aprendiendo-laravel-v5>php artisan make:controller FrutaController

354. DISEÑAR LA BASE DE DATOS DEL PROYECTO DE LARAVEL

En Laravel el campo se debe llamar id, no identificador ni leches, que si no el ORM no lo reconoce. También utiliza internamente los campos: id, role y remember\_token

355. GENERAR UN PROYECTO NUEVO DE LARAVEL

***F:\wamp64-2\www\master-php>***   
composer create-project laravel/laravel proyecto-laravel-v5 "5.6.\*" --prefer-dist

VÍDEO 356: CREAR LA BBDD:

Es obligatorio que cuando un campo sea auto-incremental, esa sea la clave primaria.

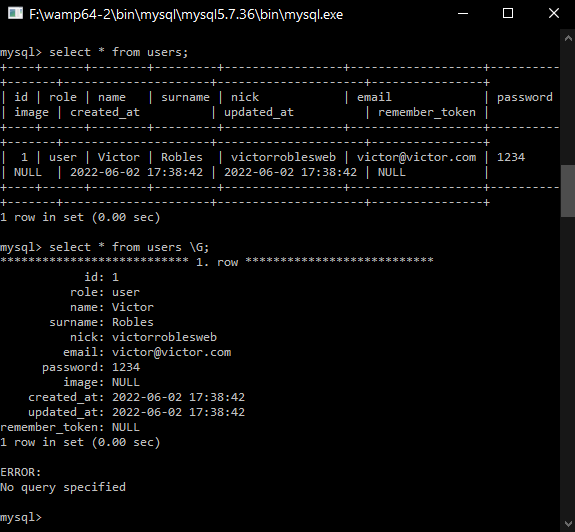
VÍDEO 358: CREAR LOS MODELOS:

F:\wamp64-2\www\master-php>cd aprendiendo-laravel-v5  
F:\wamp64-2\www\master-php\aprendiendo-laravel-v5> php artisan make:model Image

(se crea en la carpeta app)

Habla un poco de ORMs

VÍDEO 360: RELLENAR LA BBDD:



362. LOGIN Y REGISTRO DE USUARIOS (AUTH LARAVEL 5)

F:\wamp64-2\www\master-php\proyecto-laravel>php artisan make:auth

365. FORMULARIO DE CONFIGURACIÓN

F:\wamp64-2\www\master-php\proyecto-laravel>

373. FOMULARIO PARA CREAR IMÁGENES

F:\wamp64-2\www\master-php\proyecto-laravel>php artisan make:controller ImageController

381. FORMATEAR FECHAS

<https://victorroblesweb.es/2018/01/18/crear-helpers-en-laravel-5/>

Crear un provider:  
F:\wamp64-2\www\master-php\proyecto-laravel> php artisan make:provider FormatTimeServiceProvider

383. VALIDAR FORMULARIO DE COMENTARIOS

F:\wamp64-2\www\master-php\proyecto-laravel>php artisan make:controller CommentController

389. MÉTODO LIKE

F:\wamp64-2\www\master-php\proyecto-laravel>php artisan make:controller LikeController

393. CAMBIAR DE COLOR EL BOTON DE LIKE

JQuery ya viene incluido en Laravel

410. NOVEDADES E INSTALACIÓN DE NUEVAS VERSIONES DE LARAVEL

Laravel 5 y Laravel 6 son prácticamente iguales a nivel de codigo  
F:\wamp64-2\www> composer create-project --prefer-dist laravel/laravel pruebas-laravel6 “6.0.\*”  
<https://laravel.com/docs/6.x>

**F:\wamp64-2\www> cls**F:\wamp64-2\www\master-php\pruebas-laravel6> composer require laravel/ui:\*   
F:\wamp64-2\www\master-php\pruebas-laravel6> php artisan ui vue --auth  
F:\wamp64-2\www\master-php\pruebas-laravel6> npm install && npm run dev

VÍDEO 413: INSTALAR SYMPHONY 4.4:

***F:\wamp64-2\www\master-php>***composer create-project symfony/website-skeleton aprendiendo-symfony “4.2.\*”

***F:\wamp64-2\www\master-php>***composer create-project symfony/website-skeleton aprendiendo-symfony-v4.4 “4.\*”

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***composer require symfony/apache-pack

VIDEO 416: CREAR CONTROLADORES:

Mostrar todos los comandos

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console list

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console make:controller HomeController

VÍDEO 427: FUNCIONES PREDEFINIDAS TWIG:

<https://twig.symfony.com/doc/2.x/>

VIDEO 430: CREAR EXTENSIONES:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console make:twig-extension MiFiltro

VIDEO 431: LISTAR RUTAS:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***   
php bin/console debug:router

VÍDEO 432: CONEXIÓN A LA BASE DE DATOS EN SYMFONY:

DROP DATABASE `aprendiendo-sf4`;

(Se necesita el caracter ` para escapar el guion -)

En el archivo .env modificamos los parámetros para que se conecte a una DB:

*DATABASE\_URL="mysql://root@127.0.0.1:3308/aprendiendo\_sf4?serverVersion=5.7"*

Y después desde el cmd creamos la base de datos aprendiendo\_sf4   
(no utilizar el – porque luego para borrarlo hay que escapar la secuencia con `):

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console doctrine:database:create

VÍDEO 433: GENERAR ENTIDADES DESDE LA BASE DE DATOS:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console doctrine:mapping:convert --from-database yml ./src/Entity

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console doctrine:mapping:import App\Entity yml --path=src/Entity

Si ponemos yml no me va a crear la clase, entonces tenemos que poner annotation:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console doctrine:mapping:import App\Entity annotation --path=src/Entity

Generamos los getters y los setters:  
***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console make:entity --regenerate App

VÍDEO 434: GENERAR ENTIDADES CON SYMFONY:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console make:entity Usuario

Rellenar con getters y setters:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console make:entity --regenerate

VÍDEO 436: GENERAR TABLAS DESDE ENTIDADES:

***F:\>***cd wamp64-2/www/master-php/aprendiendo-symfony-v4.4

Crear un archivo en la carpeta migrations del proyecto con la tabla de la DB:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***  
php bin/console doctrine:migrations:diff

Ejecutar la migración:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***  
php bin/console doctrine:migrations:migrate

437: HACER CAMBIOS EN ENTIDADES

Tras crear la tabla en la DB de SQL vamos a crear la entidad automáticamente en el Netbeans:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console doctrine:mapping:import App\Entity annotation --path=src/Entity

Crear los getters y los setters de la App completa:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console make:entity --regenerate App

Se pueden crear solo de una entidad:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console make:entity --regenerate App\Entity\Animales

438: GUARDAR EN LA BASE DE DATOS:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console make:controller AnimalController

**ALTER TABLE Animal CHANGE [columna1] [Nuevo nombre columna] varchar(255);**

438: COMANDO SQL:

***F:\wamp64-2\www\master-php\aprendiendo-symfony-v4.4>***php bin/console doctrine:query:sql “SELECT \* FROM ANIMALES”

446: QUERY BUILDER:

<https://www.doctrine-project.org/projects/doctrine-orm/en/2.8/reference/query-builder.html>

<https://www.doctrine-project.org/projects/doctrine-orm/en/2.6/reference/query-builder.html>

461. INSTALAR SYMFONY PARA EL PROYECTO COMPLETO:

***F:\wamp64-2\www\master-php>***composer create-project symfony/website-skeleton proyecto-symfony “4.\*”

***F:\wamp64-2\www\master-php\proyecto-symfony>***composer require symfony/apache-pack

465. GENERAR ENTIDADES DESDE LAS TABLAS DE LA DB:

***F:\wamp64-2\www\master-php\proyecto-symfony>***  
php bin/console doctrine:mapping:import App\Entity annotation --path=src/Entity

Aquí es donde crearías el formulario editable teniendo ya una entidad: ESTO NO VIENE EN EL VÍDEO  
***F:\wamp64-2\www\master-php\ruta-del-proyecto>***  
php bin/console make:crud

Crear los getters y los setters:

***F:\wamp64-2\www\master-php\proyecto-symfony>***php bin/console make:entity --regenerate App

470. GUARDAR EL USUARIO REGISTRADO:

Cifrar contraseñas en symfony 4.4  
***\proyecto-symfony\config\packages\security.yaml***

475. LOGIN DE USUARIOS:

Es un archivo de: proyecto-symfony/config/packages/security.yaml

486. CONTROL DE ACCESO:

Control de acceso está en la ruta: Proyecto-symfony/config/packages/security.yaml

CREAR UN FORMULARIO EDITABLE TENIENDO YA UNA ENTIDAD:

***F:\wamp64-2\www\master-php\ruta-del-proyecto>***php bin/console make:crud

489. SYMFONY 5:

Borrar la pantalla del cmd:

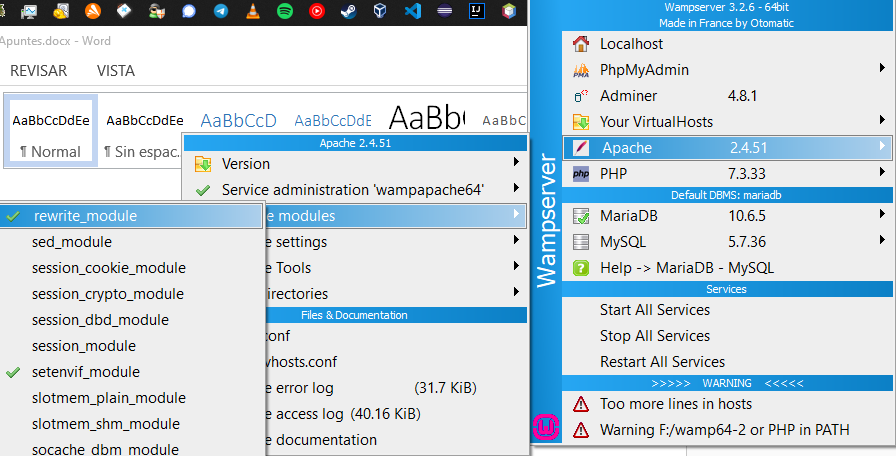
***C:\Users\W10Alex>***cls

WORDPRESS

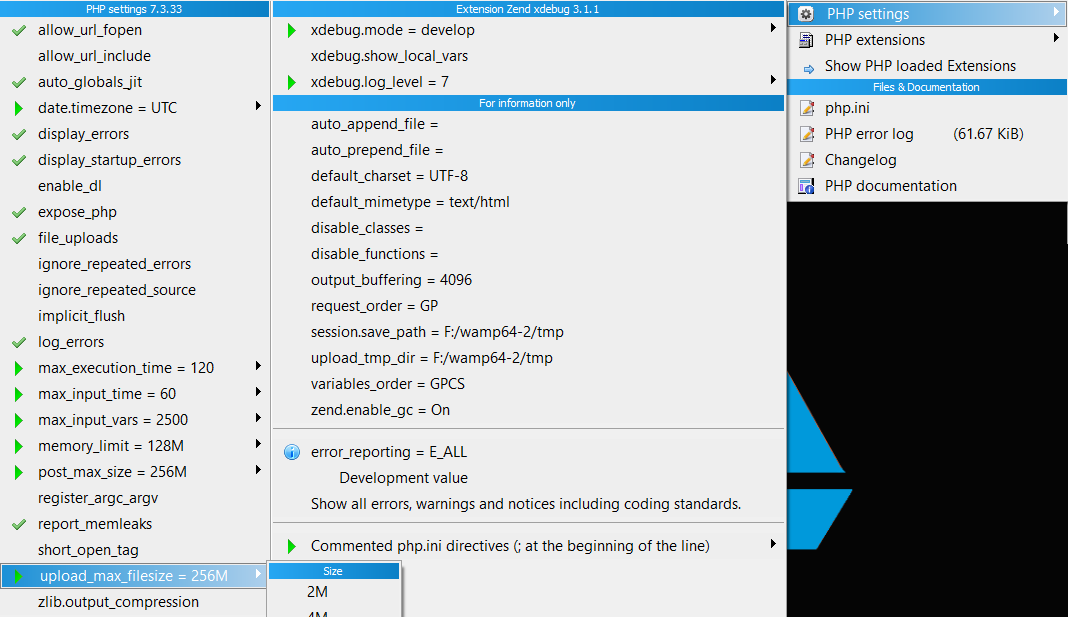
Wordpress.org -> gratuito, es necesario algún conocimiento informático  
Wordpress.com -> de pago, 0 conocimientos informáticos

493. PREPARAR EL SERVIDOR WEB LOCAL

En Apache (wampserver) hay que tener activado el siguiente modulo: rewrite\_module

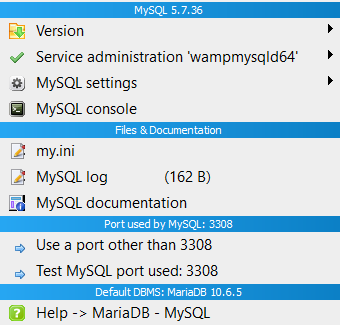


PHP settings: post\_max\_size = 256M  
PHP settings: upload\_max\_filesize = 256M



494. INSTALAR WORDPRESS:

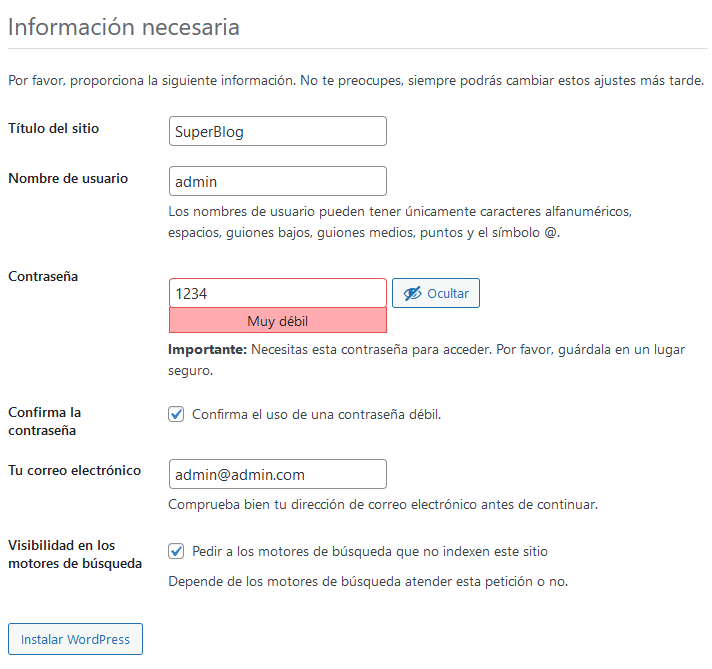
<https://es.wordpress.org/>





Se crea el: wp-config.php   
y se puede ver el nuevo puerto:





Después de este paso se crean todas las tablas en la DB.

<http://localhost/master-php/wordpress/wp-login>

user: admin  
pw: 1234