

¿Qué es MVC y cómo funciona?

El Modelo Vista Controlador es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos, manteniendo así, distintas capas que se encargan de hacer una tarea muy concreta, lo que ofrece beneficios diversos.

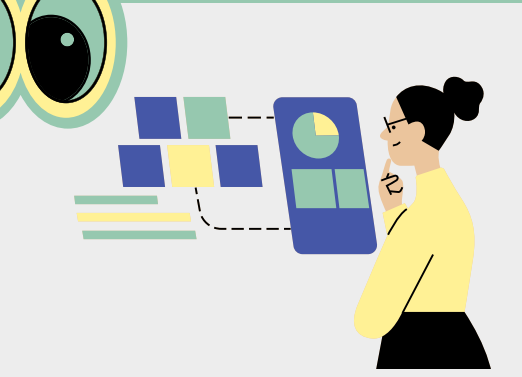
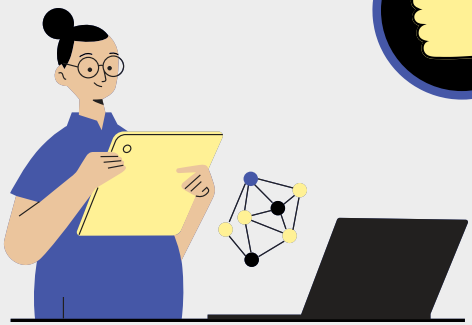


Estructura MVC

1. **Modelo:** es la responsable de gestionar los datos, almacena y recupera entidades utilizadas por una aplicación, generalmente de una base de datos y contiene la lógica implementada por la aplicación.
2. **Vista:** es la encargada de mostrar los datos proporcionados por el modelo, es la parte que ve e interactúa el cliente.
3. **Controlador:** es el que recibe una solicitud del cliente, invoca el modelo para realizar las operaciones solicitadas y envía los datos a la Vista.

Envío de datos a los controladores

El controlador es el encargado de unir la vista y el modelo, por lo que cada vez que el usuario solicite datos desde la vista, los parámetros serán enviados a una función controladora ya sea a través de variables de sesión o en la url de la función

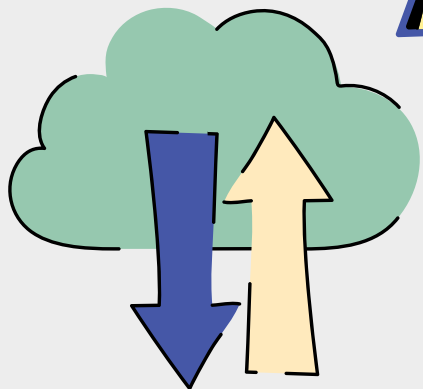


Cómo hacer envío de datos al modelo

En este caso el modelo cuenta con variedad de funciones para que los datos que recupera pueden ser accesibles en otras partes del código. El controlador es el que se encarga de llamar a las funciones del modelo y pasarle los datos necesarios a través de los parámetros de la función.

Cómo hacer envío de controladores y vistas

1. El usuario realiza una petición.
2. El controlador captura la petición.
3. Hace la llamada al modelo correspondiente.
4. El modelo será el encargado de interactuar con la base de datos.
5. El controlador recibe la información y la envía a la vista.
6. La vista muestra la información.



¿Cómo hacer envío de datos a la vista?

El envío de datos a las vistas se hace a través del controlador, luego de que este haya recibido y filtrado información de los modelos. La información puede ser consumida por la vista de varias maneras, la más común es que la vista consuma un API-REST con formato JSON con la información solicitada

REFERENCIAS

- <https://www.ecodeup.com/como-crear-una-plantilla-modelo-vista-controlador-para-tus-proyectos-en-php/>
- <https://victorroblesweb.es/2013/11/18/tutorial-mvc-en-php-nativo/>

Configuración de rutas amigables a través de .htaccess

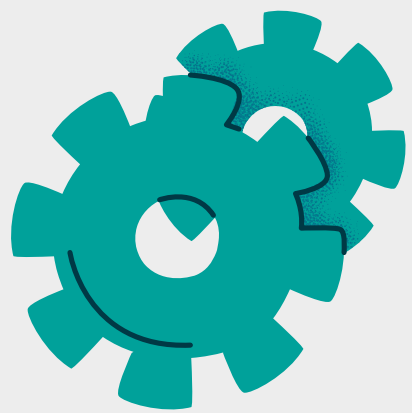


Como realizar la configuración de rutas amigables en MVC con htaccess

El archivo .htaccess es un archivo de configuración de Apache. De modo general, ese archivo es capaz de alterar la configuración del servidor web, aplicando esos cambios al directorio donde el .htaccess esté situado y todos los directorios que dependen de él. En la práctica, este fichero se usa para muchas cosas como, por ejemplo, proteger una carpeta por clave, activar la compresión de los ficheros que se transfieren, etc.

Como configurar URL amigables

- Para comenzar debemos indicar que se ponga en marcha el motor de reescritura de URL con esta línea: RewriteEngine on
- Luego tenemos que generar las redirecciones con la instrucción RewriteRule, indicando primero el patrón de la URL amigable y la redirección que se debe producir. Dicho patrón se debe colocar como una expresión regular y a continuación colocamos la URL a la que se debe redirigir la solicitud.



```
# Activar RewriteEngine
RewriteEngine on

RewriteCond %{SERVER_PORT} 80

RewriteBase /

RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME}.php -f

# Reescribir la URL solicitada por el usuario

# Home
RewriteRule ^inicio$ /index.php [L]

# Producto
# Entrada: producto/NOMBRE_PRODUCTO/
# Salida: productos.php?id=NOMBRE_PRODUCTO
RewriteRule ^producto/(.*)$ /productos.php?id=$1 [L]
```

Ejemplo

Este sería un ejemplo completo de modificación del archivo .htaccess que puede ayudarnos a crear URL amigables para distintos productos

REFERENCIAS

- <https://www.canva.com/design/DAE3jcShDvg/COK93qBc6UQJLfmodcQRiw/edit>