



# DOCUMENTACIÓN PROYECTO 1

---

## Etapas 1-4

**Ismael Aliaga Molina**

080 Formación

1º Desarrollo de Aplicaciones Web

## Tabla de contenido

Documentación Proyecto 1.....	3
Petición del cliente .....	3
Etapa 1.....	4
Etapa 2.....	5
Etapa 3.....	6
Etapa 4.....	7
¿Cómo utilizar la Aplicación Web? .....	8

# Documentación Proyecto 1

## Petición del cliente

### INTRODUCCIÓN

Nuestro cliente tiene la costumbre de anotar en un fichero las tareas pendientes nuevas que le han surgido durante el día, mover a otro fichero aquellas tareas en la que está trabajando y moverlas a otro fichero cuando están finalizadas.

Para su comodidad, nos ha pedido que le realicemos una aplicación web que le muestre las tres listas.

### REQUISITOS DEL PROYECTO

- Los datos estarán en tres ficheros planos:
  - Pendientes.txt
  - Enprogreso.txt
  - Finalizadas.txt
- La aplicación web mostrará 3 listas diferenciadas.
- La aplicación web mostrará los datos de los ficheros correspondientes.

# Etapa 1

## INTRODUCCIÓN

Hay que crear una Página Web que muestre 3 tableros para cada uno de los archivos, por lo tanto, los pasos que vamos a seguir en esta primera etapa son:

Pensar estructura HTML para la creación de 3 tableros (Tablas).

Realizar Mockup con la estructura, texto y colores (Preview del acabado final).

Crear documento bien estructurado en .php donde vamos a utilizar los lenguajes de marcas HTML y CSS para darle forma y color a los 3 tableros.

## ESTRUCTURA HTML Y MOCKUP

Lo que nos pide el cliente es una Aplicación Web de un “ToDo”, por lo que nos interesa que el cliente nada mas cargar la aplicación tenga a un golpe de vista el contenido de cada uno de los 3 ficheros diferenciados entre si.

La estructura que vamos a definir es:

Tareas Pendientes	Tareas En Progreso	Tareas Finalizadas
Contenido completo del archivo pendientes.txt	Contenido completo del archivo enprogreso.txt	Contenido completo del archivo finalizadas.txt

Una vez con tenemos en mente la anterior estructura pasamos a realizar el mockup para ver el resultado final que debería quedar.

Tareas Pendientes	Tareas En Progreso	Tareas Finalizadas
Tarea Pendiente 1	Tarea En Progreso 1	Tarea Finalizadas 1
Tarea Pendiente 2	Tarea En Progreso 2	Tarea Finalizadas 2
Tarea Pendiente 3	Tarea En Progreso 3	Tarea Finalizadas 3
Tarea Pendiente 4	Tarea En Progreso 4	Tarea Finalizadas 4
Tarea Pendiente 5	Tarea En Progreso 5	Tarea Finalizadas 5
Tarea Pendiente ...	Tarea En Progreso ...	Tarea Finalizadas ...

## Etapa 2

### INTRODUCCIÓN

Una vez establecida la estructura de las 3 tablas, contenidos y colores en el documento etapa1.php y mostrando correctamente la idea que anteriormente hemos plasmado en el mockup, tenemos que mostrar 1 línea por cada documento txt en nuestra Aplicación Web.

### PROGRAMANDO LECTURA E IMPRESIÓN DE 1 LÍNEA

Para programar esta segunda etapa vamos a utilizar en lenguaje de programación PHP que introduciremos dentro de los tableros que hemos creado anteriormente con HTML.

En mi caso quiero organizar por una parte el documento .php y una carpeta donde se encuentren los 3 archivos .txt. De esta manera conseguimos tener toda la aplicación más ordenada y ser más intuitiva.

Por lo que vamos a tener los archivos de este proyecto estructurados de la siguiente manera:

- Proyecto1.php
- Tareas/nombreakivo.txt

Una vez que tenemos la estructura creada y los archivos txt del cliente, creamos un bloque de php en cada tablero donde se tiene que imprimir cada archivo.

Para abrir el fichero txt creamos una variable y en ella utilizamos la instrucción fopen con la ruta donde se encuentra físicamente el documento txt para que pueda abrirlo y le indicamos que sea solo de lectura.

Una vez tenemos el txt localizado y abierto tenemos que leer la primera línea, para ello creamos una nueva variable donde se almacenara el contenido de esa primera línea y utilizamos la instrucción fgets y le indicamos de donde queremos que lea esa línea, en este caso sería de la variable que hemos utilizado para abrir el fichero con el fopen.

En este punto ya tenemos la primera línea en la 2º variable que hemos creado, solo queda realizar un echo a la 2º variable que contiene la línea para que la imprima en pantalla y poder ver el contenido de la primera línea en la aplicación. Para acabar utilizaremos la función fclose para cerrar el archivo.

Este proceso tenemos que replicarlo para los siguientes 2 tableros nombrando cada variable semánticamente para poder tener la aplicación mas ordenada y saber de un vistazo que hace y a que hace referencia cada una de las variables.

## **Etapas 3**

### **INTRODUCCIÓN**

Una vez hemos programado la etapa 2, tenemos que programar la etapa 3. Para ello vamos a pasar de imprimir 1 línea en pantalla por cada tablero a imprimir 3 líneas en pantalla por cada tablero.

### **PROGRAMANDO LECTURA E IMPRESIÓN DE 3 LÍNEA**

Para programar esta tercera etapa vamos a partir del código anterior.

Después del echo que hemos utilizado para imprimir la 1º línea del archivo utilizaremos otro echo con un `br` (salto de línea en html) para que el programa no muestre todas las líneas en la misma línea si no que deje entre línea y línea un salto de página.

Seguidamente volveremos a utilizar la variable que habíamos creado para que almacenara la línea de texto y le indicaremos que vuelva a leer la siguiente línea del archivo con otro fgets, es decir volveremos a reutilizar el código que contenía la 2º variable y el fgets sin ninguna variación. Esto lo que va a hacer es que en esa 2º variable va a volver a guardar el valor de la nueva línea en este caso la 2º línea.

Y para mostrar esta 2º línea volveremos a utilizar un echo a la 2º variable para que muestre el nuevo valor que en este caso es imprimir la 2º línea del documento.

Para mostrar la 3ª línea tendremos que repetir los mismos pasos que hemos utilizado para mostrar la 2ª línea, es decir, utilizar un `echo br` para que nos realice un salto de línea, volver a utilizar la 2ª variable con el mismo `fgets` para que continúe leyendo la siguiente línea en este caso la 3ª línea y volver a imprimir en pantalla el contenido de la 2ª variable que en este caso es la 3ª línea del documento. Para acabar utilizaremos la función `fclose` para cerrar el archivo.

Este proceso habría que repetirlo para los otros 2 tablonos.

## Etapa 4

### INTRODUCCIÓN

Llegamos a la etapa final del proyecto. En este punto tenemos que mostrar todas las líneas que contenga el documento por lo tanto tenemos que cambiar la manera en la que hemos programado la Aplicación Web.

Hasta ahora nuestra aplicación nos lee las líneas que nosotros queramos ya que manualmente le decimos cuantas líneas tiene que leer, esto es eficaz si sabemos el nº de líneas que tiene un documento o queremos mostrar X líneas de un documento.

En este caso como el cliente va a hacer y deshacer los documentos con las diferentes tareas pendientes, por realizar y realizadas, no podemos saber el nº totales de líneas que va a tener cada documento. Por ello vamos a realizar la última parte del 1º proyecto con un bucle `while` para que no pare de leer el documento hasta que este finalice.

### PROGRAMANDO LECTURA E IMPRESIÓN COMPLETA

Para programar esta cuarta etapa vamos a comenzar abriendo el fichero `txt` creando una variable y en ella utilizamos la instrucción `fopen` con la ruta donde se encuentra físicamente el documento `txt` para que pueda abrirlo y le indicamos que sea solo de lectura.

Después creamos un bucle while y le asignamos la condición fgets del archivo que hemos abierto anteriormente para que guarde el contenido de cada línea en una nueva 2º variable. Lo que conseguimos con este planteamiento es que el bucle no pare de reproducirse mientras en el documento tengamos líneas por leer.

Seguidamente como queremos que nos imprima todo el documento dentro del bucle vamos a indicar que queremos que se imprima en pantalla la 2º variable que es la que contiene la línea del documento con un echo y le volvemos a añadir otro echo con un br para que nos realice un salto de página.

De esta manera vamos a conseguir que lea la línea y la almacene en la 2º variable, que imprima la línea de la 2º variable, realice un salto de página, vuelva a guardar el valor de la 2º línea en la 2º variable, vuelva a imprimirse la línea de la 2º variable que en este caso sería la 2º línea del documento, imprimiría otro salto de página y no pararía de realizar este proceso hasta que en el documento no existan más líneas por mostrar. En este caso al no tener más líneas por mostrar el bucle nos devolverá el valor FALSE y se parará. Para acabar utilizaremos la función fclose fuera del bucle para que una vez acabe el bucle while se cierre el archivo.

Este paso tendremos que repetirlo en los dos tablonos que quedan.

## ¿Cómo utilizar la Aplicación Web?

### VER APLICACIÓN WEB

Para poder ver el contenido de los 3 archivos en la Aplicación Web hay que ejecutar el fichero etapa4.php

### MODIFICAR CONTENIDO FICHEROS DE TEXTO

Para agregar o quitar tareas de los ficheros de texto hay que abrir la carpeta tareas, dentro de esta carpeta encontraremos tres ficheros de texto:

Pendientes.txt



- Al editar este fichero se pueden establecer nuevas tareas o eliminarlas de la columna “Tareas Pendientes”.

#### Enprogreso.txt

- Al editar este fichero se pueden establecer nuevas tareas o eliminarlas de la columna “Tareas En Progreso”.

#### Finalizadas.txt

- Al editar este fichero se pueden establecer nuevas tareas o eliminarlas de la columna “Tareas Finalizadas”.