



**CICLO: D.A.W.
MÓDULO DE DESARROLLO WEB EN
ENTORNO SERVIDOR**

Tarea N° 01

**Alumno:
Armando Herrero Silva
76442135W**

Los documentos, elementos gráficos, vídeos, transparencias y otros recursos didácticos incluidos en este contenido pueden contener imprecisiones técnicas o errores tipográficos. Periódicamente se realizan cambios en el contenido. Fomento Ocupacional FOC SL puede realizar en cualquier momento, sin previo aviso, mejoras y/o cambios en el contenido.

Es responsabilidad del usuario el cumplimiento de todas las leyes de derechos de autor aplicables. Ningún elemento de este contenido (documentos, elementos gráficos, vídeos, transparencias y otros recursos didácticos asociados), ni parte de este contenido puede ser reproducida, almacenada o introducida en un sistema de recuperación, ni transmitida de ninguna forma ni por ningún medio (ya sea electrónico, mecánico, por fotocopia, grabación o de otra manera), ni con ningún propósito, sin la previa autorización por escrito de Fomento Ocupacional FOC SL.

Este contenido está protegido por la ley de propiedad intelectual e industrial. Pertenecen a Fomento Ocupacional FOC SL los derechos de autor y los demás derechos de propiedad intelectual e industrial sobre este contenido.

Sin perjuicio de los casos en que la ley aplicable prohíbe la exclusión de la responsabilidad por daños, Fomento Ocupacional FOC SL no se responsabiliza en ningún caso de daños indirectos, sean cuales fueren su naturaleza u origen, que se deriven o de otro modo estén relacionados con el uso de este contenido.

© 2022 Fomento Ocupacional FOC SL todos los derechos reservados.

Contenido

1.	RA01_a.	2
2.	RA01_b.	2
3.	RA01_c.	3
4.	RA01_d.	4
5.	RA01_e.	5
6.	RA01_f.	6
7.	RA04_g.	7
8.	RA04_h.	¡Error! Marcador no definido.
9.	RA04_i.	¡Error! Marcador no definido.

1. RA01_a.

Se han caracterizado y diferenciado los modelos de ejecución de código en el servidor y en el cliente web.

¿Qué diferencias clave existen entre la ejecución de código en el cliente (navegador) y en el servidor? Proporciona un ejemplo de cuándo usarías cada uno.

Las diferencias claves entre la ejecución de código en el cliente y en el servidor son en donde se ejecuta el código, como se mantienen los recursos y que tipo de interacciones tiene con el usuario y los datos.

Usaría una ejecución en entorno cliente si quisiera hacer una validación de un formulario en tiempo real sin enviar datos al servidor.

La ejecución en el servidor la usaría cuando necesito validar datos sensibles como por ejemplo una contraseña o datos personales.

2. RA01_b.

Se han reconocido las ventajas que proporciona la generación dinámica de páginas.

Explica cómo una página web dinámica mejora la experiencia del usuario frente a una página estática. Luego, da un ejemplo práctico de una aplicación donde se beneficie de la generación dinámica de páginas.

La generación de páginas dinámicas permite al usuario modificar ciertos aspectos interactivos de la web sin tener que recargar la misma.

Las redes sociales como Twitter enseñan elementos personalizados según los gustos del usuario y a qué otros usuarios sigan. La sección de “Para ti” de Twitter será diferente según cada usuario, los cuales podrán usarlo a la vez sin que ninguno afecte al otro y sin tener que recargar la página.

3. RA01_c.

Se han identificado los mecanismos de ejecución de código en los servidores Web.

Descargar de Apache Friends el servidor XAMPP y llevar a cabo su instalación.

Probar que funciona correctamente, lanzando la url localhost en el navegador.

Desarrollar una página web que incluya los siguientes elementos:

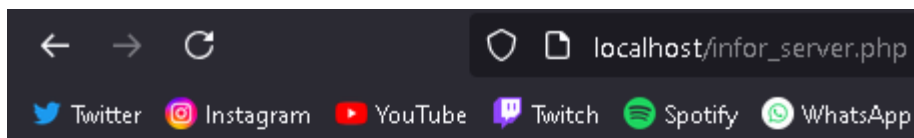
- Cabecera y pie de página.
- Nombre y NIF del alumno.

Un script de servidor php que muestra información sobre la configuración del servidor:

```
<?php
echo "Servidor: " . $_SERVER["SERVER_SOFTWARE"] . "<br>";
echo "Raíz: " . $_SERVER["DOCUMENT_ROOT"] . "<br>";
?>
```

Lanzar la página en el servidor y comprobar su funcionamiento.

```
C: > xampp > htdocs > infor_server.php
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Información del servidor para la tarea</title>
7  </head>
8  <body>
9      <header>
10         <h1>Información del Servidor</h1>
11         <p>Alumno: Armando Herrero Silva - NIF: 76442135W</p>
12     </header>
13
14     <div class="content">
15         <h2>Configuración del servidor</h2>
16         <?php
17             echo "Servidor: " . $_SERVER["SERVER_SOFTWARE"] . "<br>";
18             echo "Raíz: " . $_SERVER["DOCUMENT_ROOT"] . "<br>";
19         ?>
20     </div>
21
22     <footer>
23         <p>© 2024 Esto es el footer de la página.</p>
24     </footer>
25 </body>
26 </html>
```



Información del Servidor

Alumno: Armando Herrero Silva - NIF: 76442135W

Configuración del servidor

Servidor: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12
Raíz: C:/xampp/htdocs

© 2024 Esto es el footer de la página.

4. RA01_d.

Se han reconocido las funcionalidades que aportan los servidores de aplicaciones y su integración con los servidores web.

Ventajas y desventajas de usar un servidor web y un servidor de aplicaciones separados:

- Explica por qué sería ventajoso utilizar un servidor web separado de un servidor de aplicaciones.***
- ¿Cuáles podrían ser algunos desafíos técnicos al mantener ambos servidores separados?***

Las ventajas de usar un servidor web y un servidor de aplicaciones separados son:

- a) Optimización del rendimiento
- b) Escalabilidad de los componentes según la demanda
- c) Seguridad mejorada al limitar el acceso directo al servidor de aplicaciones
- d) Se consigue un mantenimiento sencillo

Las desventajas para mantener los servidores separados serían las siguientes:

- a) Al deber configurar ambos servidores, lleva más esfuerzo que uno solo
- b) Se puede crear una mayor latencia de las solicitudes dinámicas que deben procesar ambos servidores
- c) Mayor coste de mantenimiento
- d) Mayor esfuerzo para mantener la seguridad

5. RA01_e.

Se han identificado y caracterizado los principales lenguajes y tecnologías relacionados con la programación Web en entorno servidor.

Describir brevemente los componentes del paquete de aplicaciones XAMPP y sus funciones. Indicar cinco ventajas del lenguaje de programación de servidor PHP que consideres más importantes.

Los componentes de XAMPP son:

1. Apache
Es el servidor web de XAMPP
2. MySQL
Gestiona bases de datos
3. FileZilla
Es un servidor FTP para transferir archivos entre el servidor y el cliente
4. Mercury
Es un servidor de correo electrónico
5. Tomcat
Tomcat es un servidor de aplicaciones que ejecuta aplicaciones basadas en Java

Las ventajas de PHP son:

1. Facilidad de aprendizaje
2. Gran soporte comunitario
3. Amplia compatibilidad
4. Generación dinámica de contenido
5. Se integra con HTML

6. RA01_f.

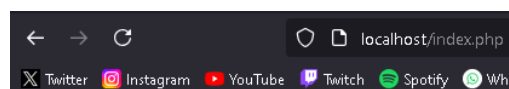
Se han verificado los mecanismos de integración de los lenguajes de marcas con los lenguajes de programación en entorno servidor.

Crea un pequeño proyecto donde utilices un lenguaje de marcas (como HTML) integrado con un lenguaje de programación en servidor (como PHP) para generar contenido dinámico.: Crear una página web donde se pueda agregar y visualizar una lista de tareas almacenadas en un archivo de texto. El contenido del archivo se cargará dinámicamente en la página utilizando PHP.

```

index.php x
C: > xampp > htdocs > index.php
3  <head>
7  </head>
8  <body>
9      <header><h1>Lista de tareas actualizable</h1></header>
10     <h2>Tareas a realizar</h2>
11
12     <form method="post" action="index.php">
13         <input type="text" name="tarea" placeholder="Nueva tarea">
14         <button type="submit">Agregar</button>
15     </form>
16
17     <ul>
18         <?php
19             //Leer el archivo
20             $tareas = file_get_contents("tareas.txt");
21             $tareas = explode("\n", $tareas);
22
23             //Mostrar tareas en lista
24             foreach ($tareas as $tarea) {
25                 if(trim($tarea) != ""){
26                     echo "<li>$tarea</li>";
27                 }
28             }
29         <?>
30     </ul>
31
32     <?php
33         //Añadir nueva tarea
34         if(isset($_POST["tarea"])){
35             $nuevaTarea = $_POST["tarea"];
36             $tareas[] = $nuevaTarea;
37             file_put_contents("tareas.txt", implode("\n", $tareas));
38         }
39     <?>
40 </body>
41 </html>

```



Lista de tareas actualizable

Tareas a realizar

- Sacar al perro
- Comprar gouda
- Hacer salsa de queso para nachos y pasta
- Hacer salsa de queso para nachos y pasta

7. RA01_g.

Se han reconocido y evaluado las herramientas de programación en entorno servidor.

Hacer una breve comparativa de al menos tres entornos de desarrollo integrado (IDE) para PHP.

Elige un framework de programación en entorno servidor (como Django, Express o Laravel) y describe sus principales características y beneficios en el desarrollo de aplicaciones web.

PhpStorm es un IDE muy complejo y potente, el cual está desarrollado específicamente para el uso de PHP. Por otro lado, VisualStudio Code, el cual es muy popular, puede usarse para trabajar con PHP mediante sus plugings, al igual que NetBeans.

Laravel es un framework que goza de una gran comunidad, la cual contribuye activamente en su desarrollo e incluso brindan soporte a otros usuarios. Además cuenta con una muy buena documentación y medidas de seguridad integradas para proteger las aplicaciones de ataques externos.