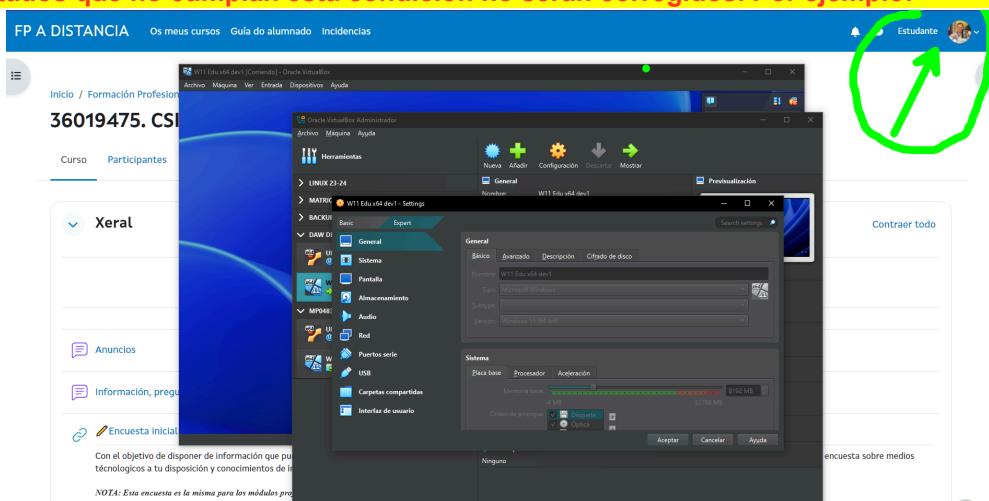


**AVISO:** Las tareas son trabajo individual y no deben ser compartidas con otros compañer@s. Si se detecta o se sospecha que una tarea es copia total o parcial de otra, serán anuladas las 2 tareas, independientemente de quién realizara la versión original. La nota de ambas tareas será de 0 puntos

En los apartados en los que es necesario entregar las capturas de pantalla, éstas deben tener como fondo de pantalla la plataforma de fpdistancia con tu usuario mostrando claramente la foto de tu perfil. Aquellos apartados/subapartados que no cumplan esta condición no serán corregidos. Por ejemplo:



**CONSEJO PARA NOVAT@s: RECUERDA IR GUARDANDO TU TRABAJO (HACIENDO CLIC EN EL DISKETE DE LA BARRA DE HERRAMIENTAS) CADA POCO TIEMPO PARA NO PERDERLO.**

### TAREA 1: Trabajo con Visual Studio Code (3 puntos)

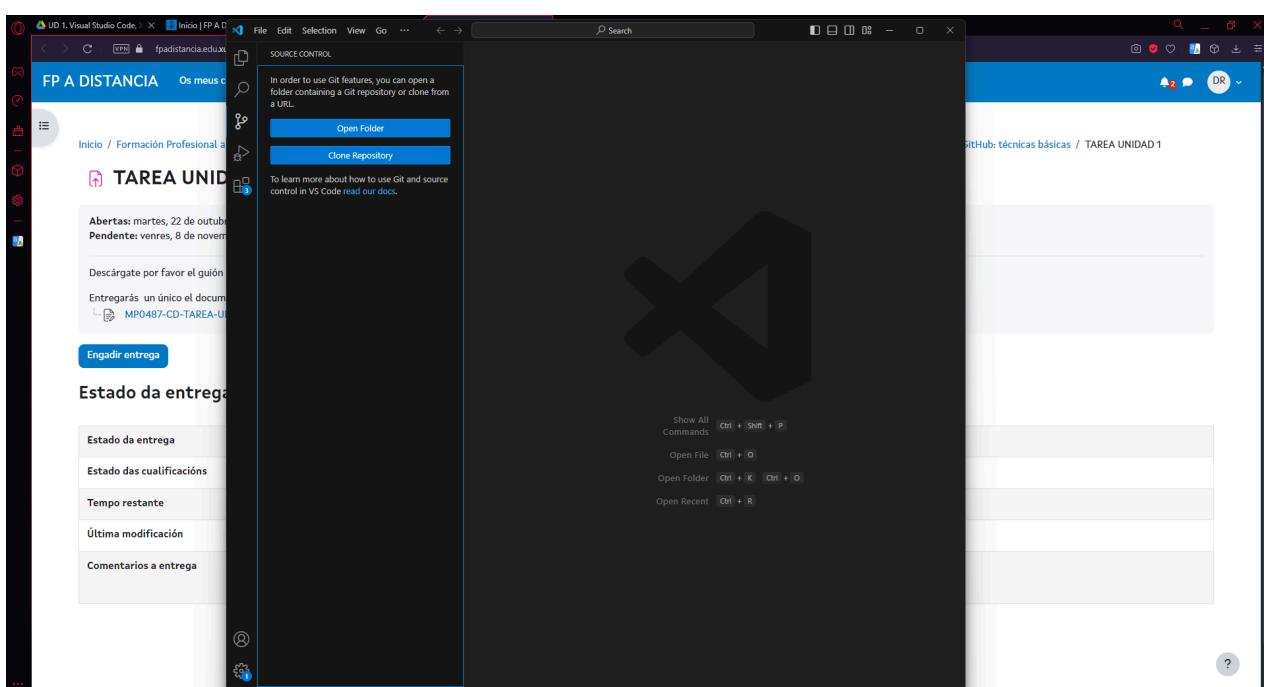
**NOTA:** cuando no se indica cuánto puntuá cada apartado de una TAREA, significa que todos puntuán lo mismo (puntos de la tarea / número de apartados). Por ejemplo, en este caso: cada apartado puntuá 3 / 8 puntos.

Realiza las tareas siguientes:

- 1) Instala Visual Studio Code y Git en tu máquina.

**CAPTURA/S DE PANTALLA** que muestra Visual Studio Code (VSC) abierto y mostrando que aparece la opción "Clonar el repositorio Git" en la pantalla de Bienvenida:

Ya los tengo instalados y puedo utilizar Git desde VSC.

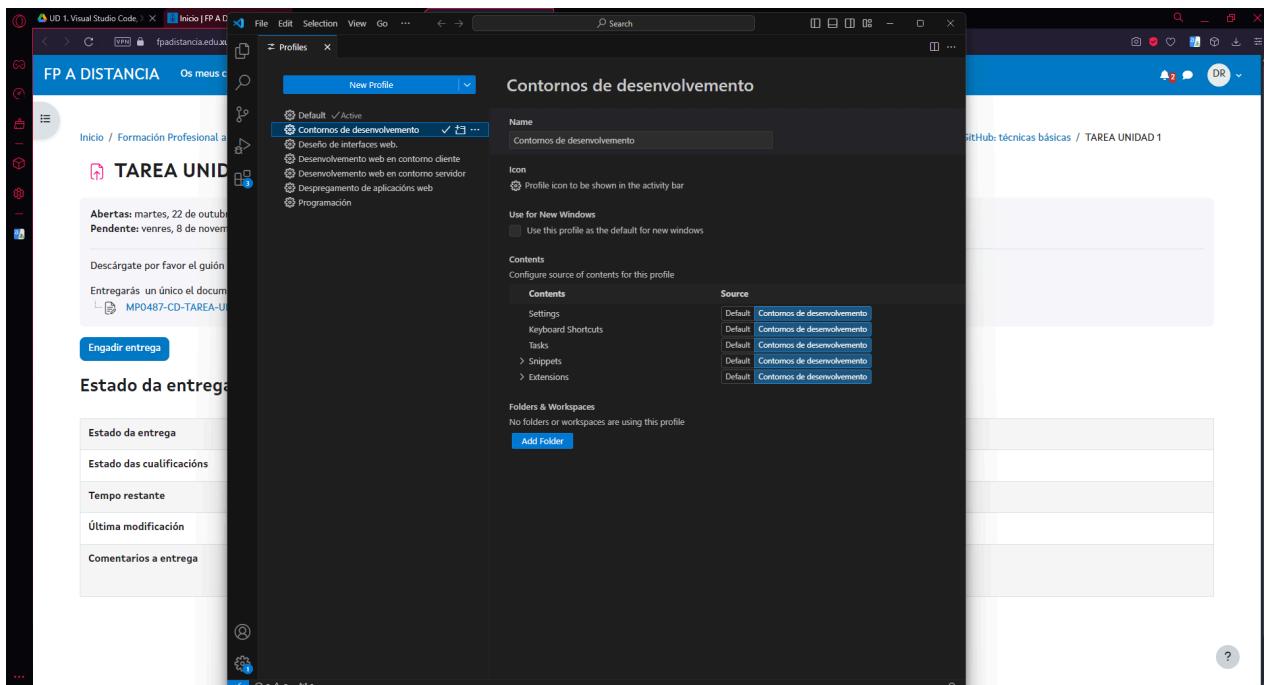


## 2) Perfiles de VSC

En VSC crea un perfil para cada módulo formativo en el que estás matriculado, para tener separadas las opciones de configuración, extensiones, etc. de forma que lo que se configura en VSC en un módulo formativo no afecta a otros.

### CAPTURA/S DE PANTALLA de VSC en los que aparecen los perfiles así creados

He creado un perfil para cada módulo.

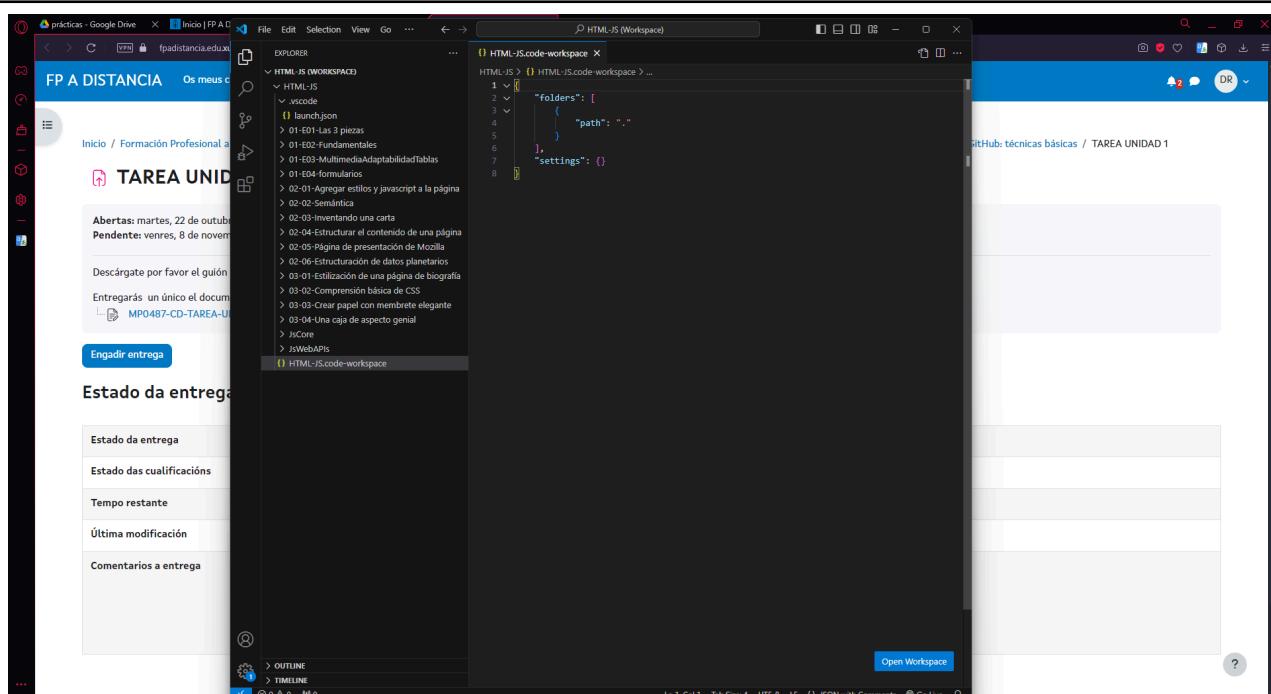


## 3) Workspaces/Áreas de trabajo

- Crea en tu máquina una carpeta CODIGO y dentro 3 carpetas: Java, HTML-JS y PHP.
- Descárgate del repo <https://github.com/davidgacaeduxuntgal/Curso24-25-MP0487-CD>, en la carpeta UNIDAD-01/CODIGO-PRUEBAS, código de ejemplo para meter en las carpetas correspondientes creadas en a) (Ejemplos de Java en CODIGO/Java, ejemplos de Javascript en CODIGO/HTML-JS y ejemplos de PHP en CODIGO/PHP).
- Configura, para cada carpeta la depuración de VSC para el tipo específico de código de cada carpeta: cada carpeta se configurará como un Workspace distinto.

### CAPTURA/S DE PANTALLA de VSC en los que aparece el contenido de la configuración del Workspace para CODIGO/HTML-JS y el contenido del launch.json de configuración de depuración.

`CODIGO\HTML-JS\HTML-JS.code-workspace`

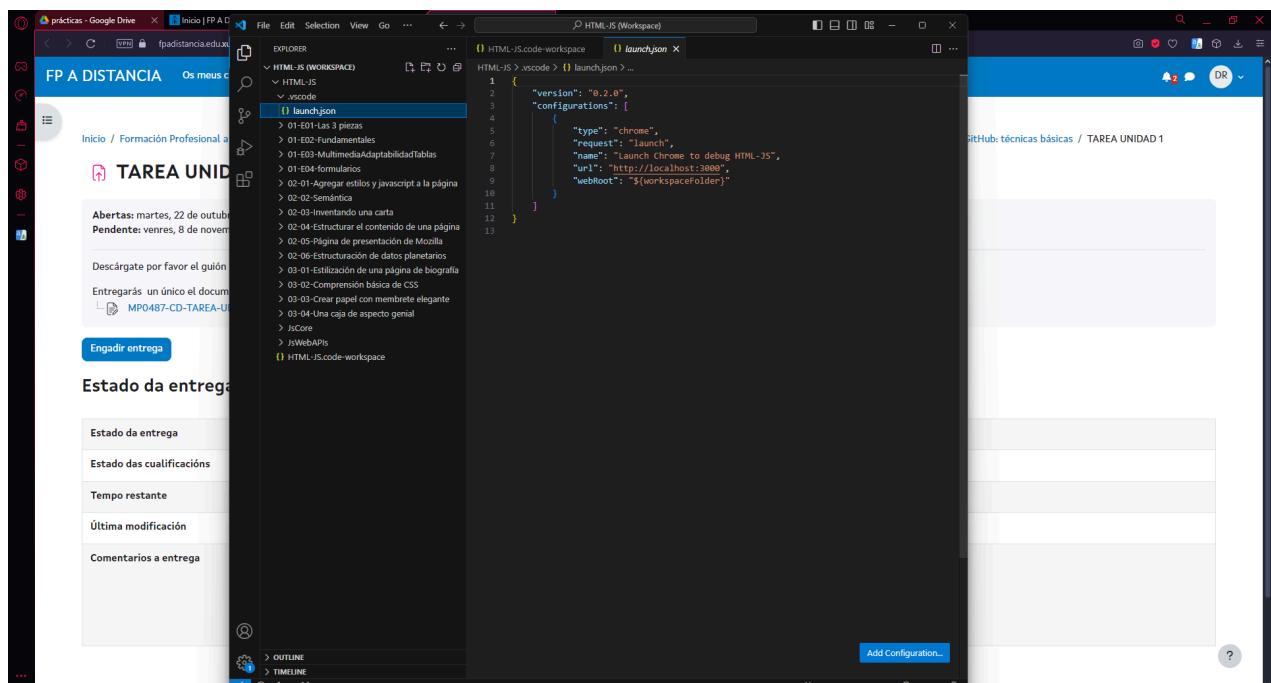


```

{
    "version": "0.2.0",
    "configurations": [
        {
            "type": "chrome",
            "request": "launch",
            "name": "Launch Chrome to debug HTML-JS",
            "url": "http://localhost:3000",
            "webRoot": "${workspaceFolder}"
        }
    ]
}

```

### CODIGO\HTML-JS\vscode\launch.json

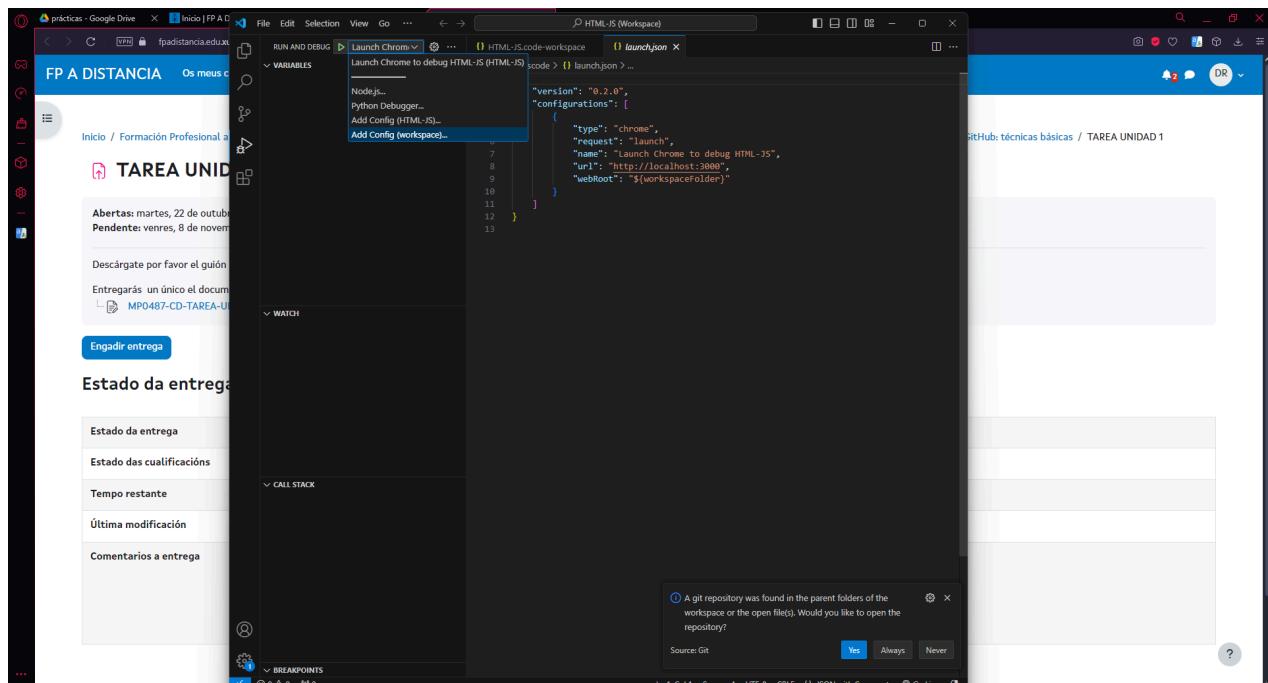


```

{
    "version": "0.2.0",
    "configurations": [
        {
            "type": "chrome",
            "request": "launch",
            "name": "Launch Chrome to debug HTML-JS",
            "url": "http://localhost:3000",
            "webRoot": "${workspaceFolder}"
        }
    ]
}

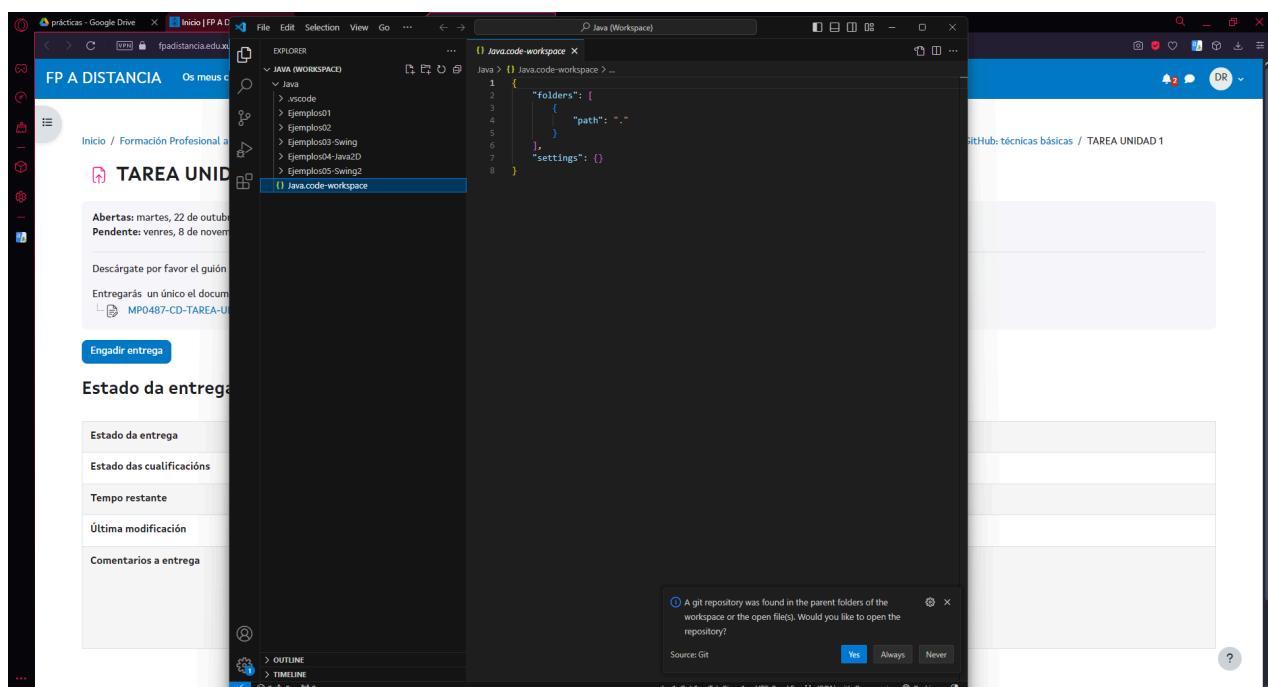
```

### Vista de depuración para HTML-JS

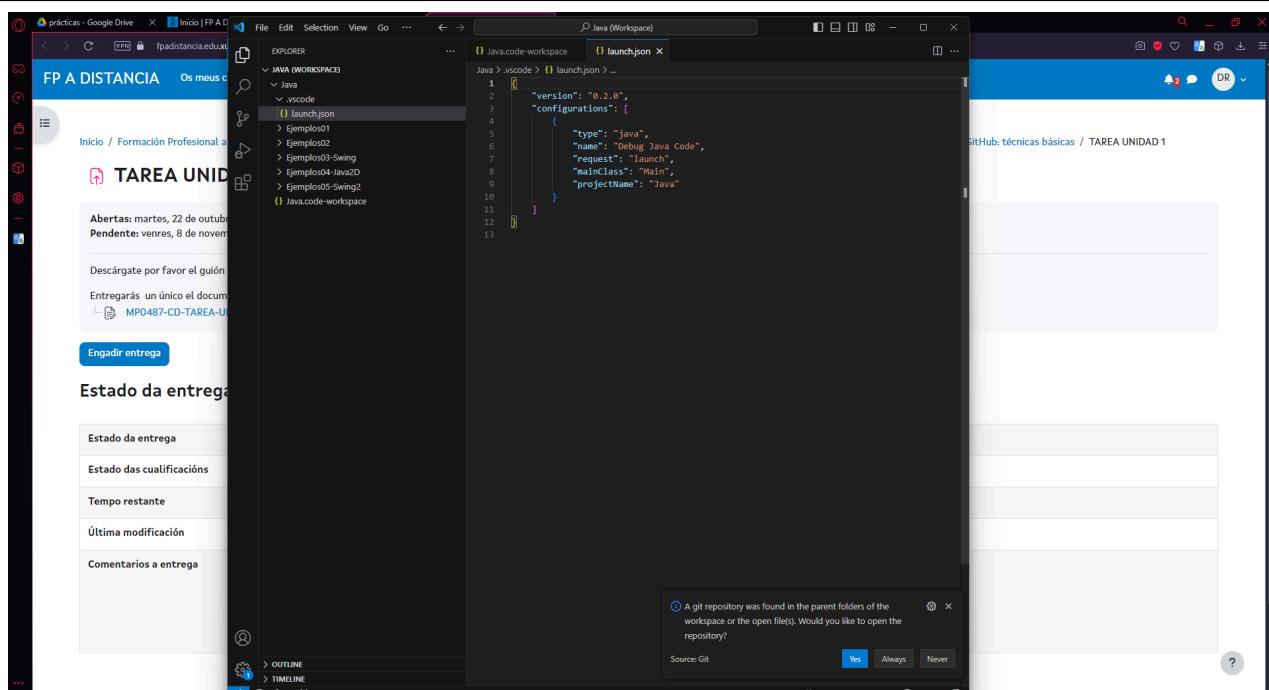


**CAPTURA/S DE PANTALLA de VSC en los que aparece el contenido de la configuración del Workspace para CODIGO/Java y el contenido del launch.json de configuración de depuración.**

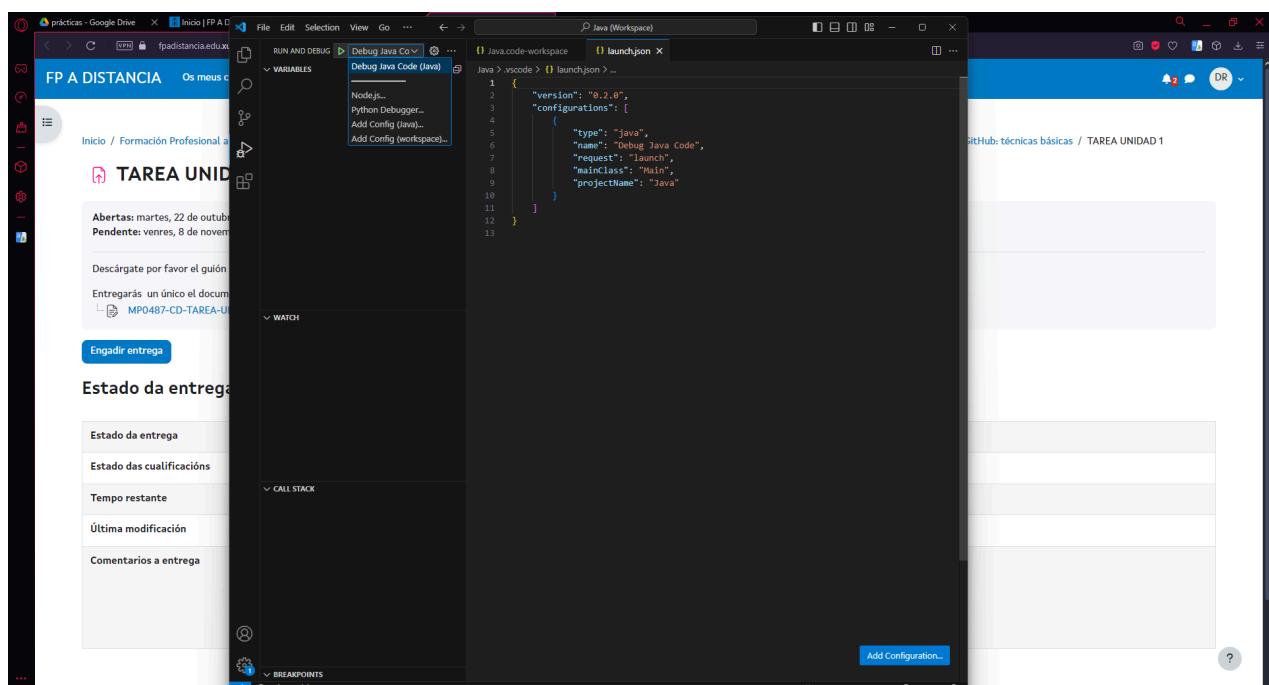
**CODIGO\JAVA\JAVA.code-workspace**



**CODIGO\Java\vscode\launch.json**



Vista de depuración para Java

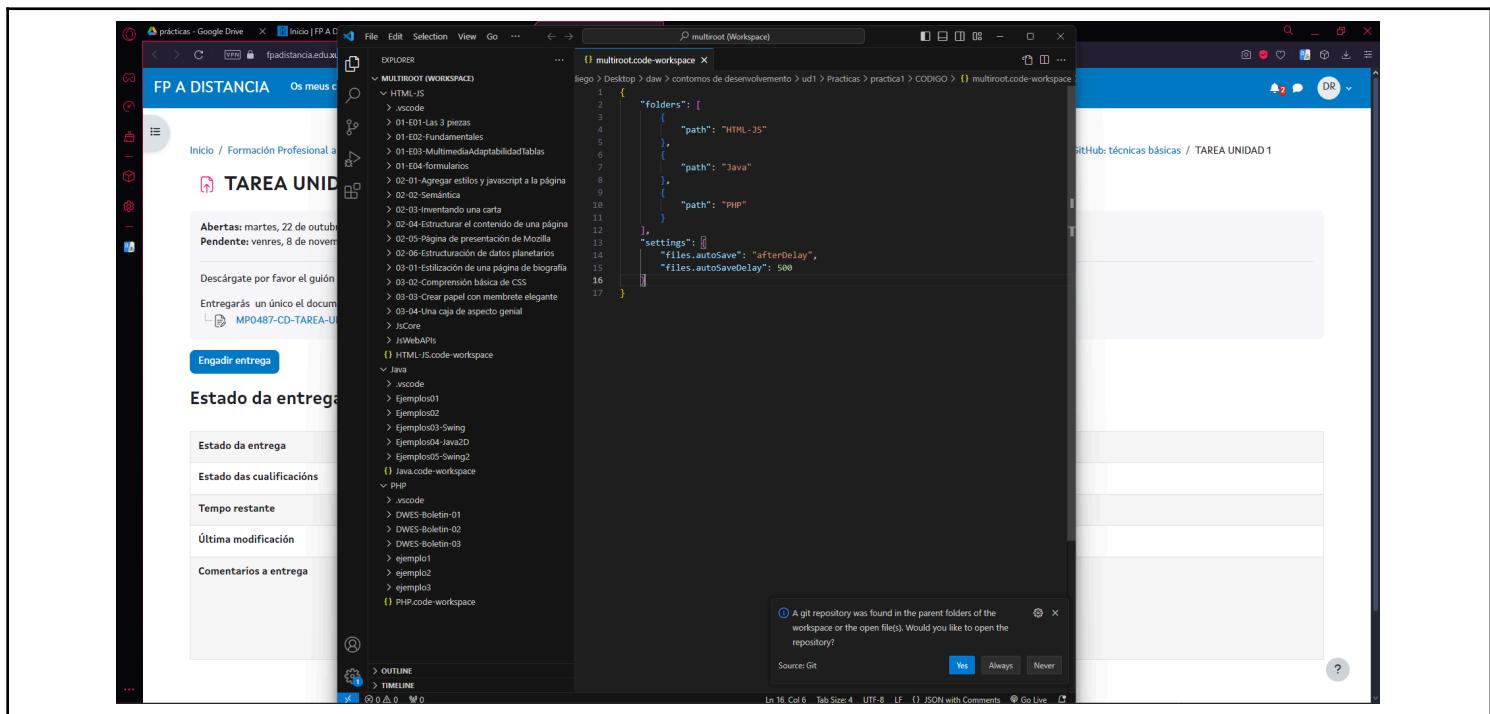


#### 4) Multiroot-workspace

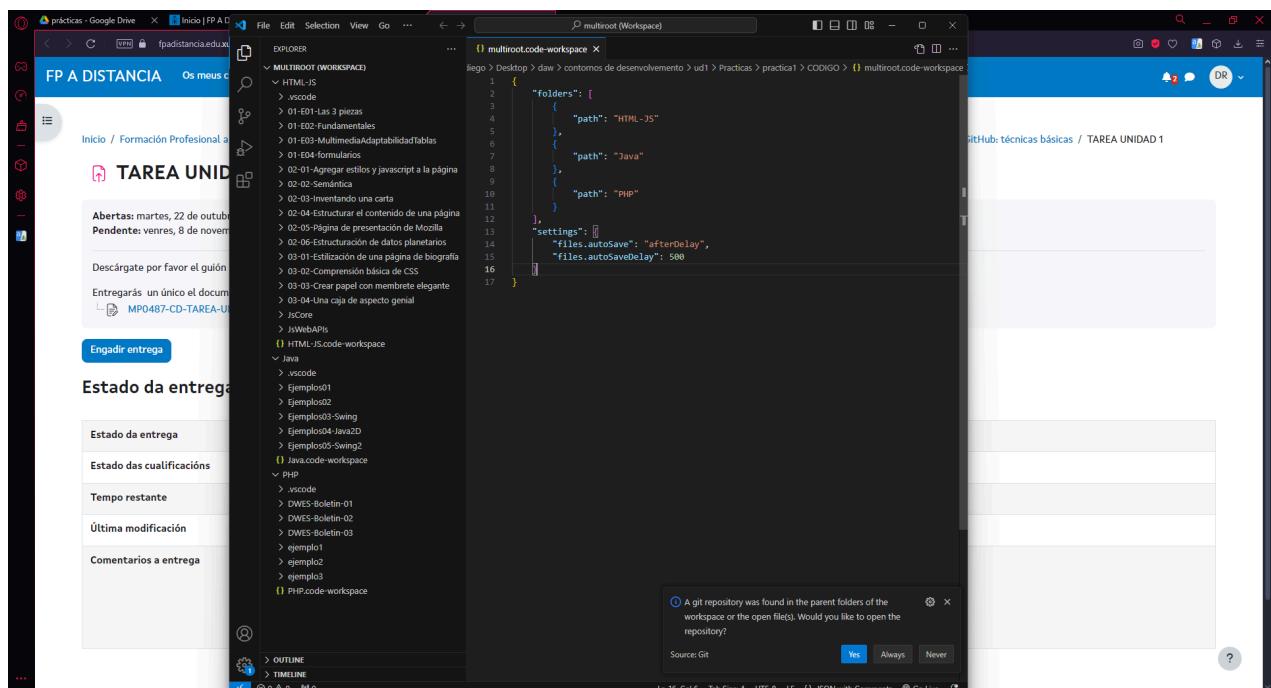
Ahora las 3 carpetas del apartado anterior queremos que comparten ciertas opciones de configuración y que se puedan abrir a la vez en una misma instancia de VSC.

- a) Crea un Multiroot-workspace para ello.
- b) Modifica en tiempo de autosave (autoguardado) a 500ms (milisegundos).
- c) Guarda el multiroot-workspace.

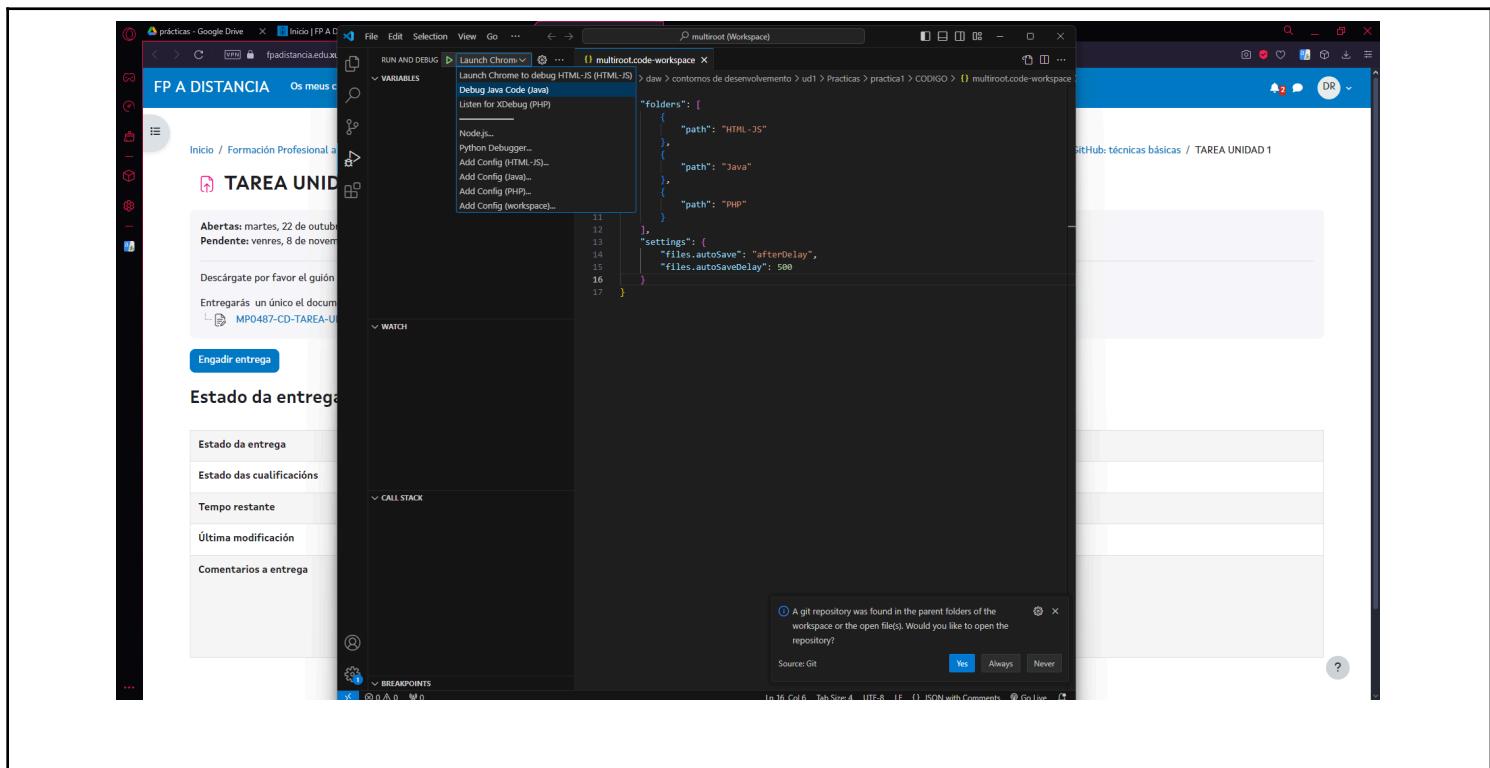
**CAPTURA/S DE PANTALLA de VSC en la que aparece el multiroot-workspace creado:**



## CAPTURA/S DE PANTALLA del archivo de configuración del multiroot-workspace creado:



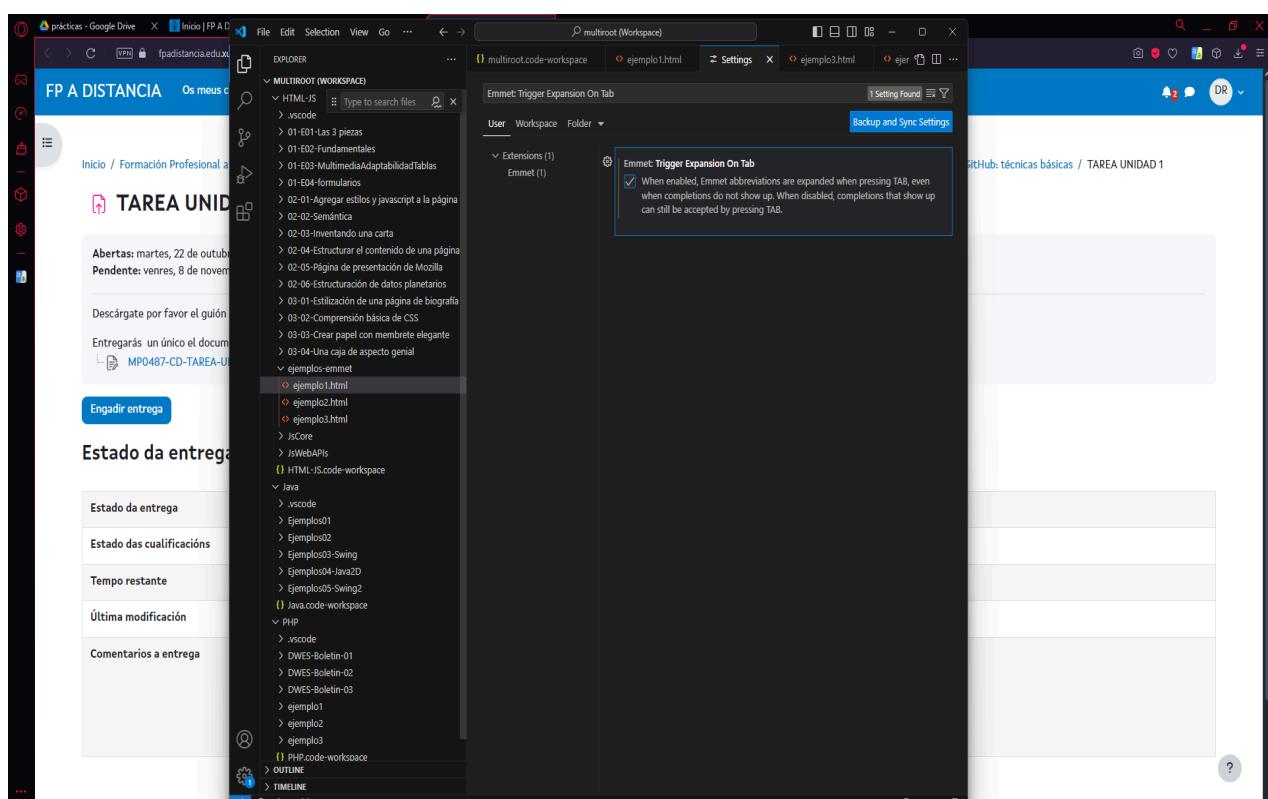
Depuración global

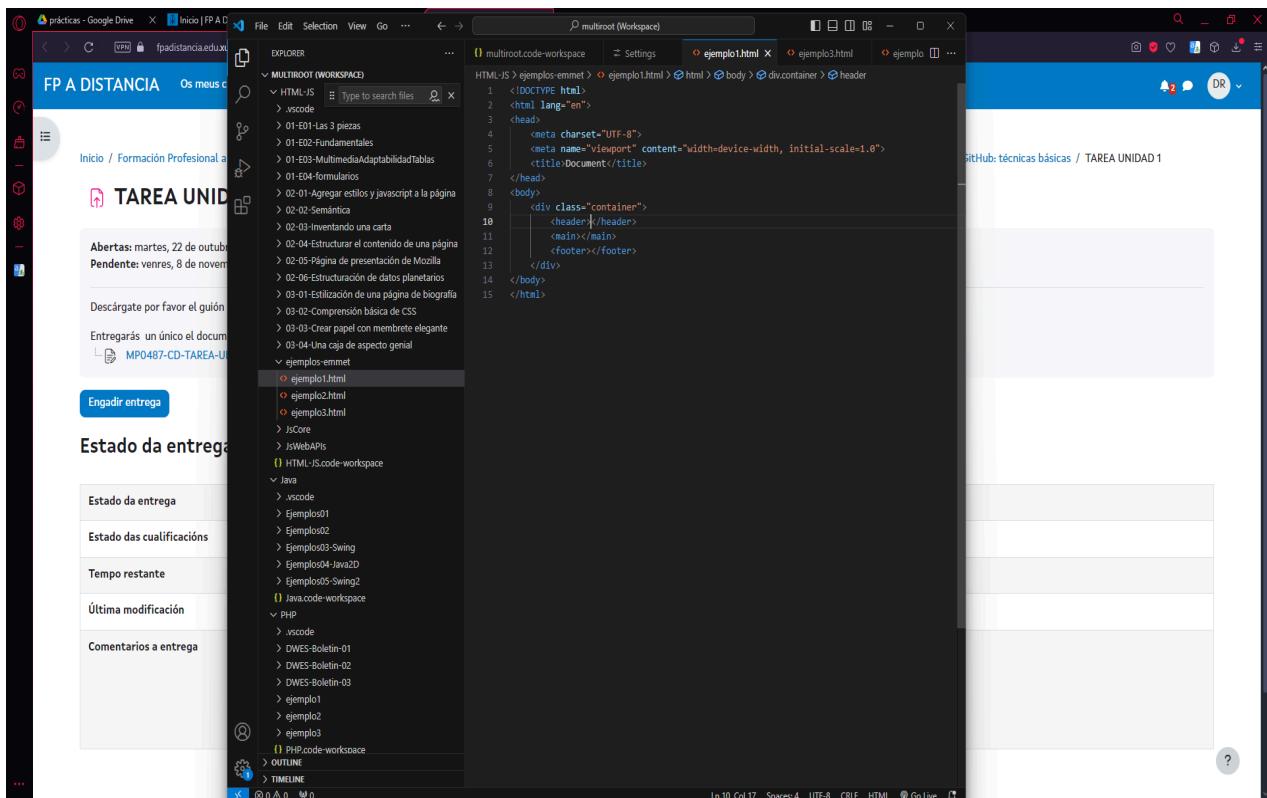


## 5) Emmet

Muestra 3 ejemplos de código HTML generado usando Emmet, junto con la expresión emmet usada:

### CAPTURA/S DE PANTALLA de VSC en la que aparece ejemplo Emmet 1:



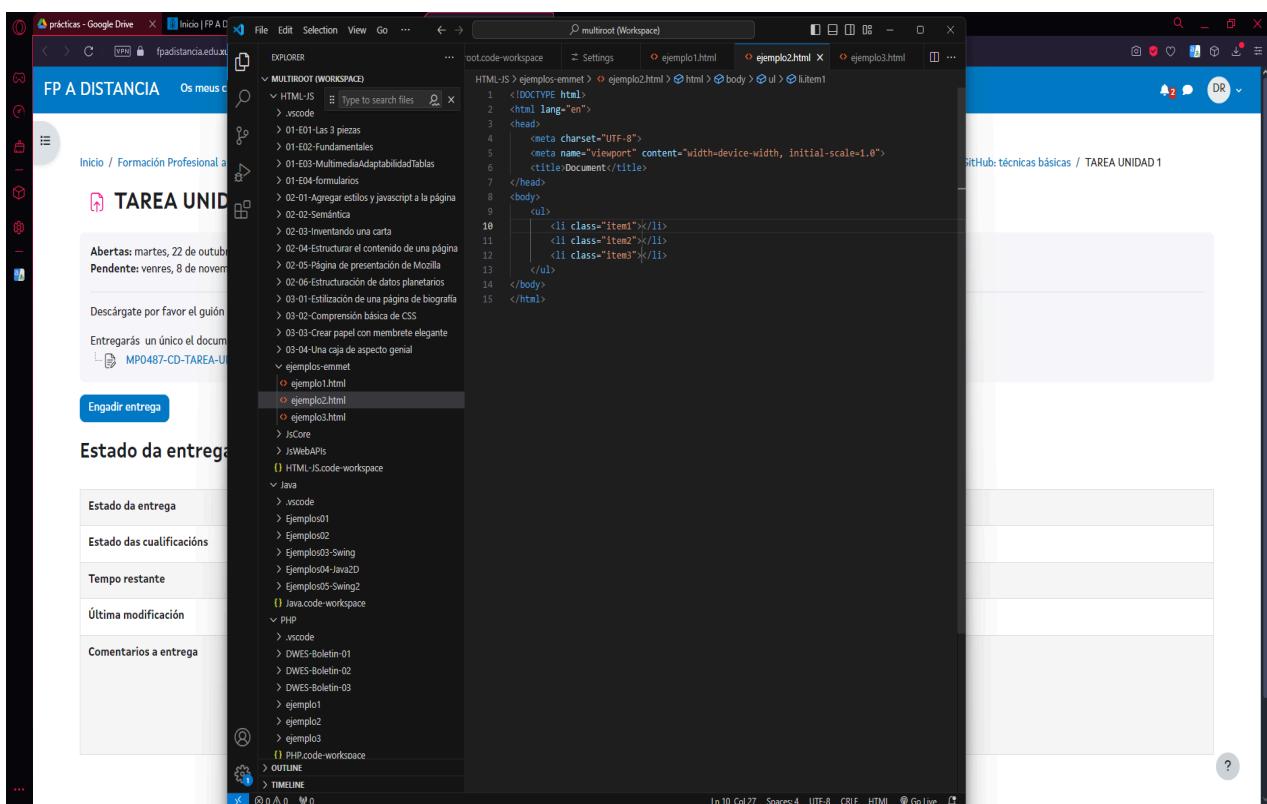
*div.container>header+main+footer*


```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
</head>
<body>
<div class="container">
<header></header>
<main></main>
<footer></footer>
</div>
</body>
</html>

```

## CAPTURA/S DE PANTALLA de VSC en la que aparece ejemplo Emmet 2:

*ul>li.item\$\*3*


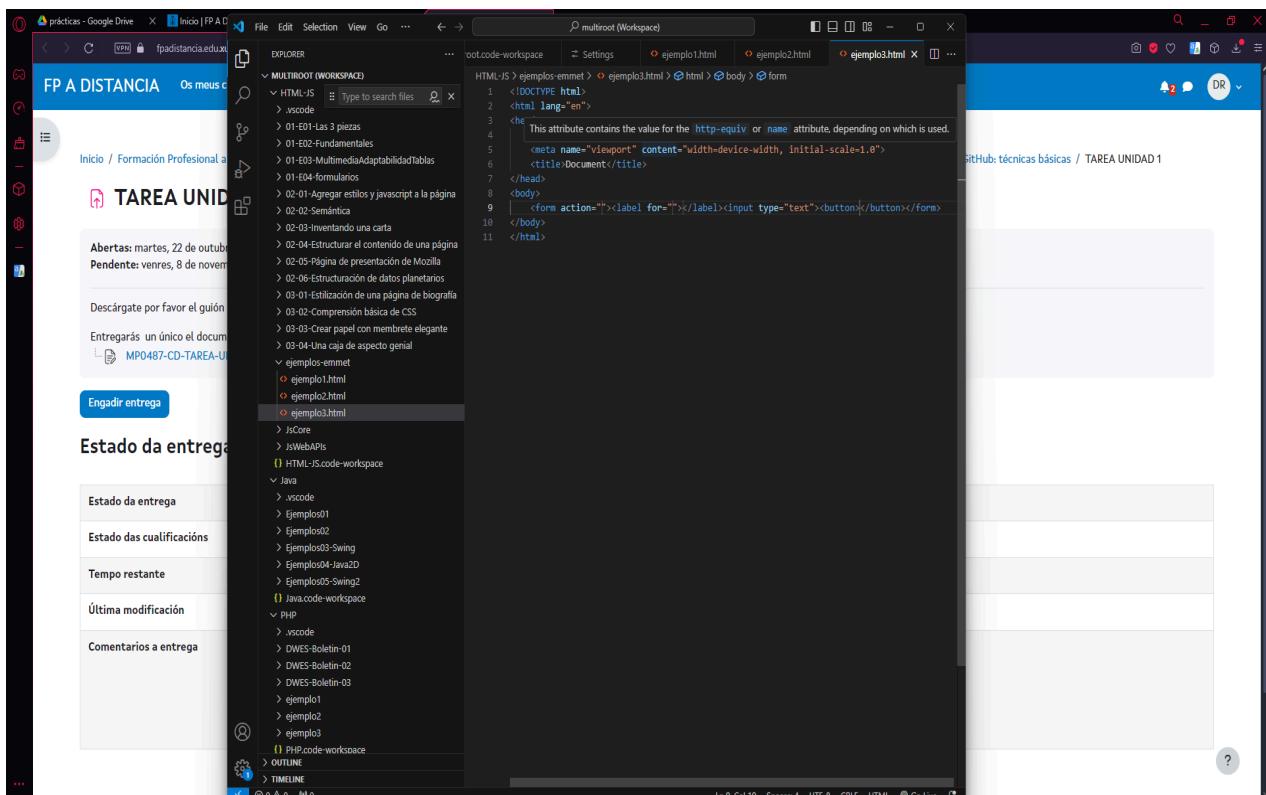
```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
</head>
<body>
<ul>
<li class="item1"></li>
<li class="item2"></li>
<li class="item3"></li>
</ul>
</body>
</html>

```

**CAPTURA/S DE PANTALLA de VSC en la que aparece ejemplo Emmet 2:**

*form>label+input[type="text"]+button*



The screenshot shows a VS Code interface with a dark theme. On the left is the Explorer sidebar, which lists several workspace folders and files. One file, 'ejemplo3.html', is currently selected and expanded, showing its code structure. The main editor area displays the expanded code:

```

<!DOCTYPE html>
<html lang="en">
<head>
    This attribute contains the value for the http-equiv or name attribute, depending on which is used.
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <form action=""><label for=""/></label><input type="text"><button>/button</button></form>
</body>
</html>

```

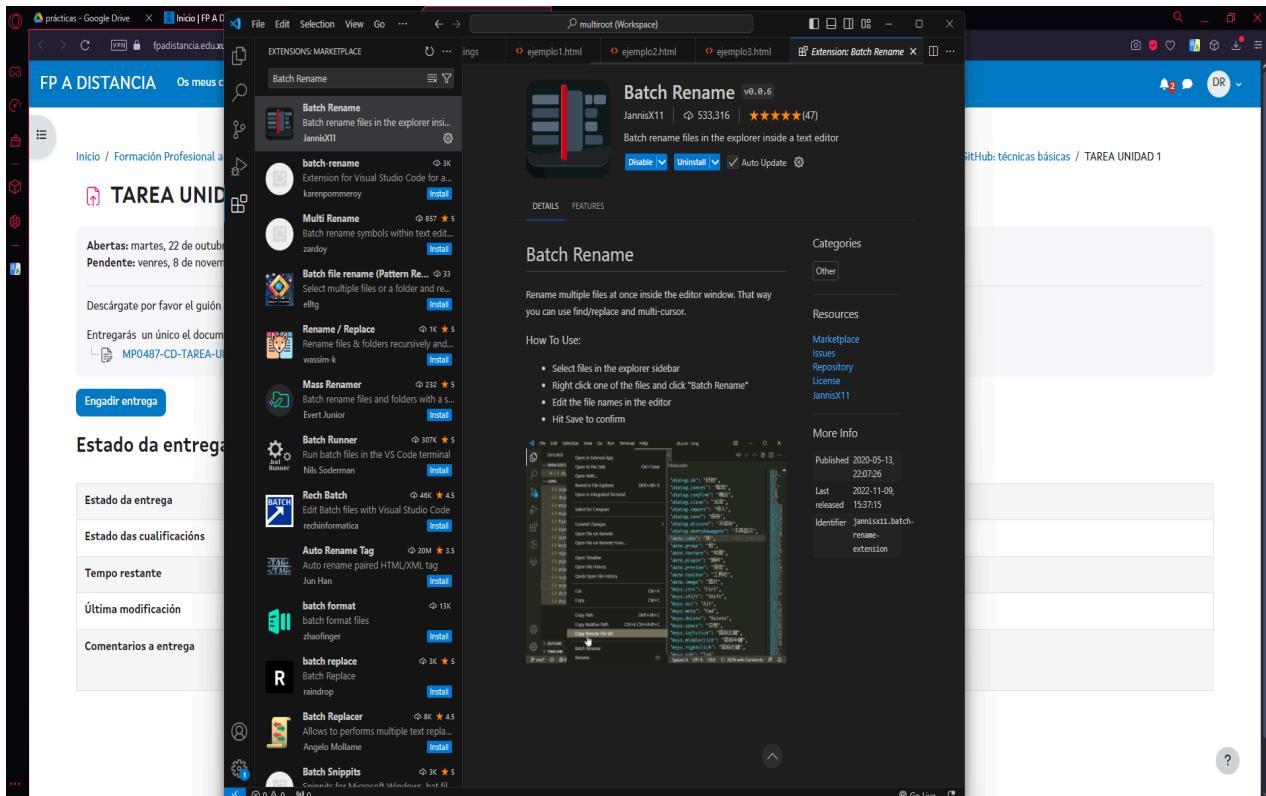
To the right of the editor is a browser window showing a simple HTML page with a form containing a label and an input field.

**6) Instala:**

- a) una extensión que te permita modificar el nombre de varios archivos de golpe.
- b) una extensión que te permita, al mismo tiempo que modificas el código HTML/CSS de una página, ver los resultados inmediatamente reflejados en un navegador abierto.

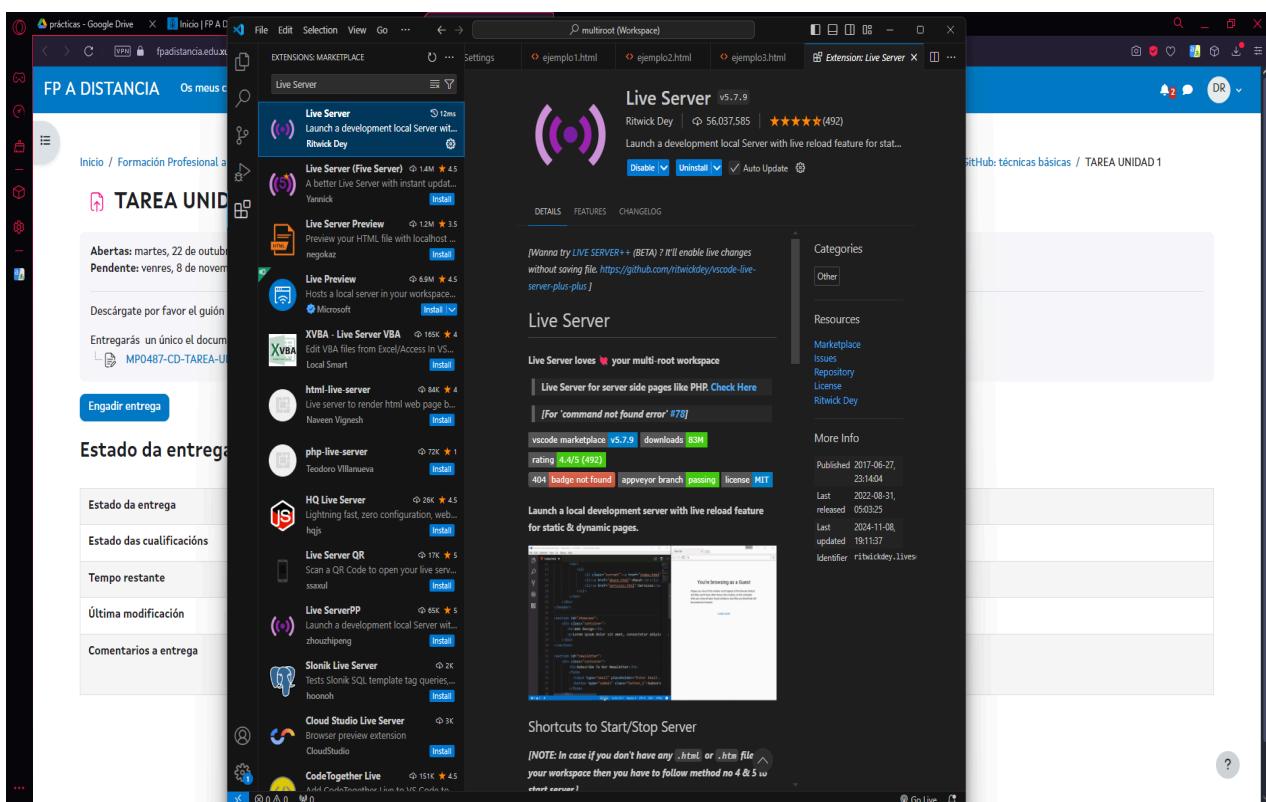
**CAPTURA/S DE PANTALLA de VSC en la que aparece la extensión a) instalada**

## Batch Rename



## CAPTURA/S DE PANTALLA de VSC en la que aparece la extensión b) instalada

### Live Server



7) Configura VSC para poder depurar código HTML/CSS/Javascript corriendo en Firefox

## ENUMERACIÓN DE LOS PASOS SEGUIDOS

Paso 1: Instalar la extensión “Debugger for Firefox”

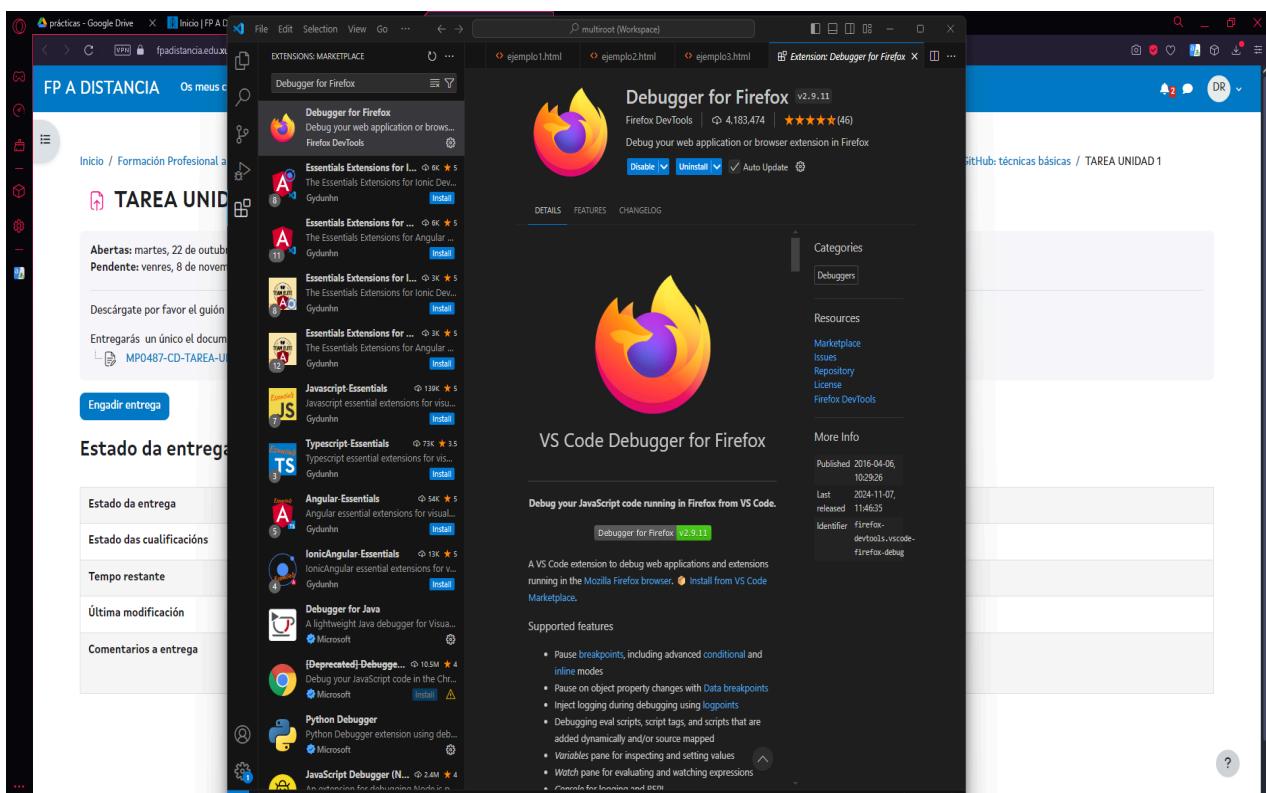
Paso 2: Crear una Configuración de Depuración en Visual Studio Code

Paso 3: Configuración en Firefox

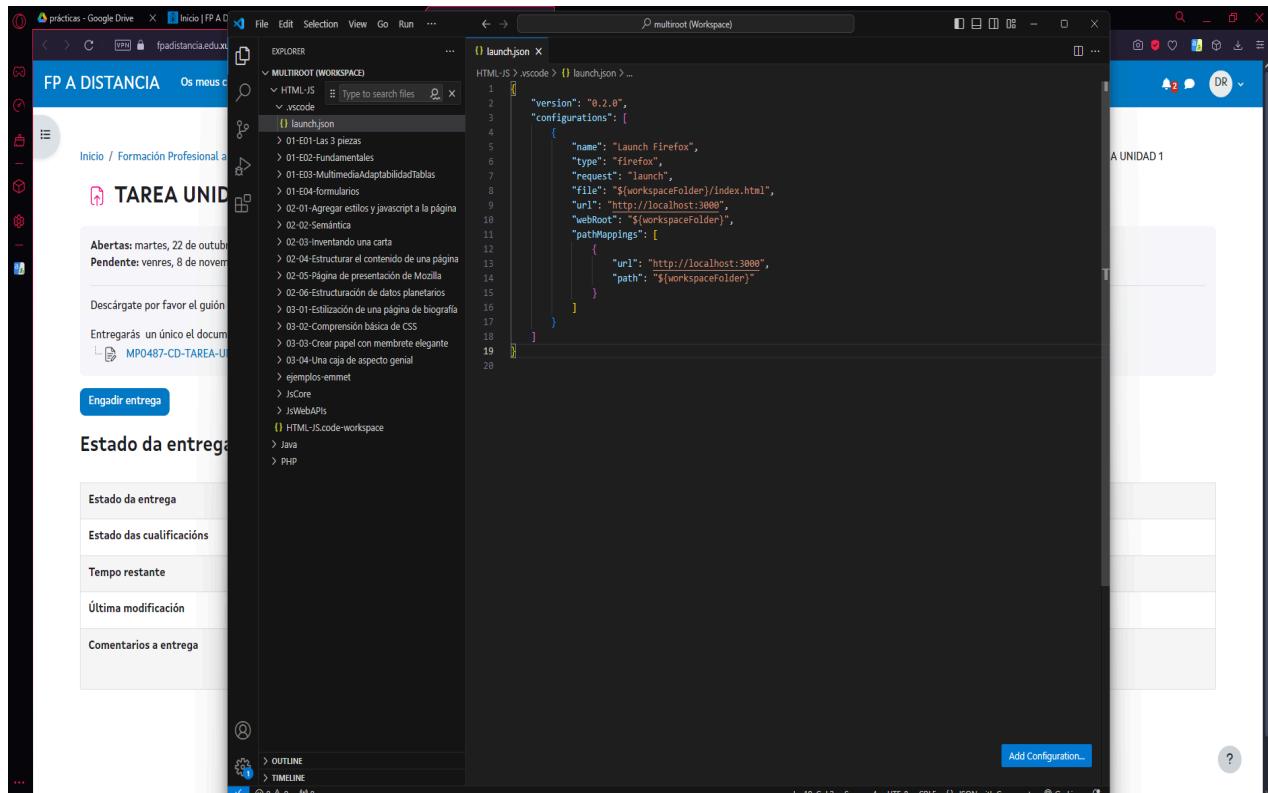
Paso 4: Iniciar la Depuración

## CAPTURA/S DE PANTALLA de VSC en las que se muestren los cambios que hubo que hacer:

Paso 1: Instalar la extensión “Debugger for Firefox”

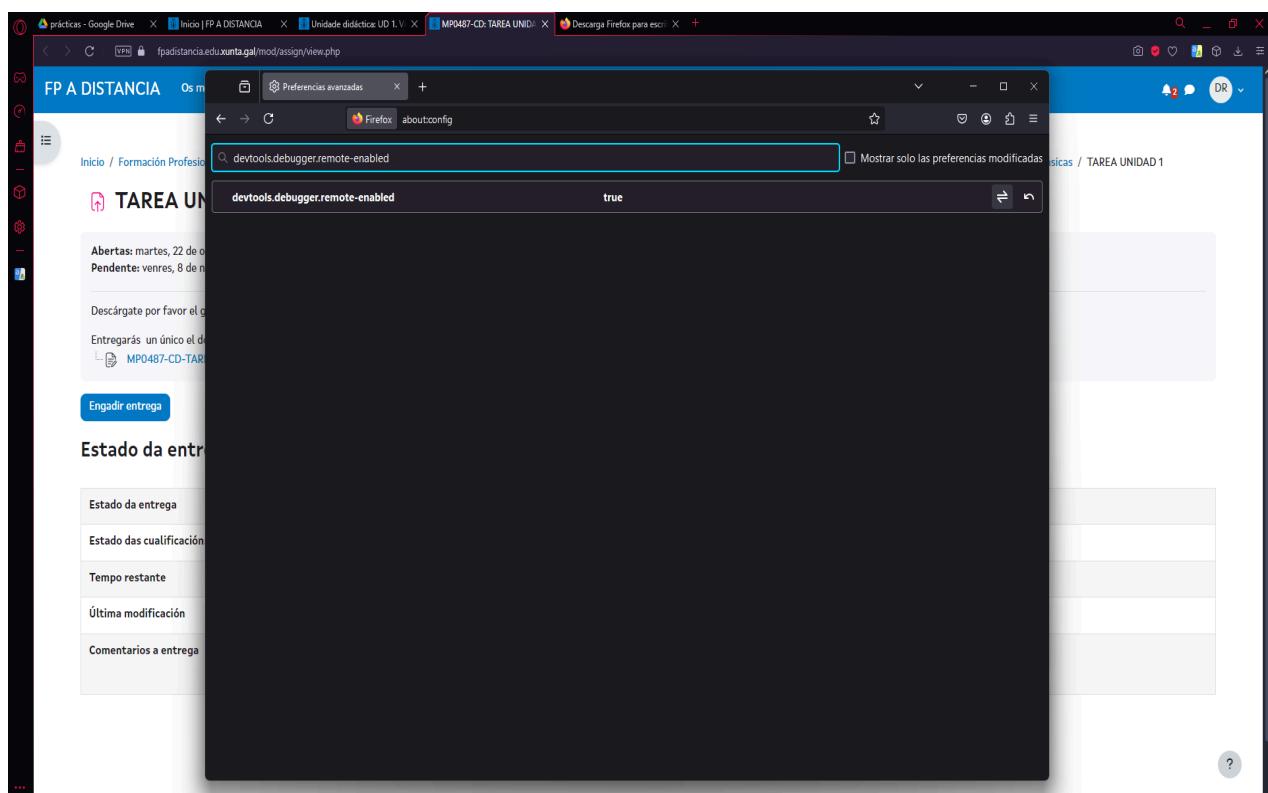


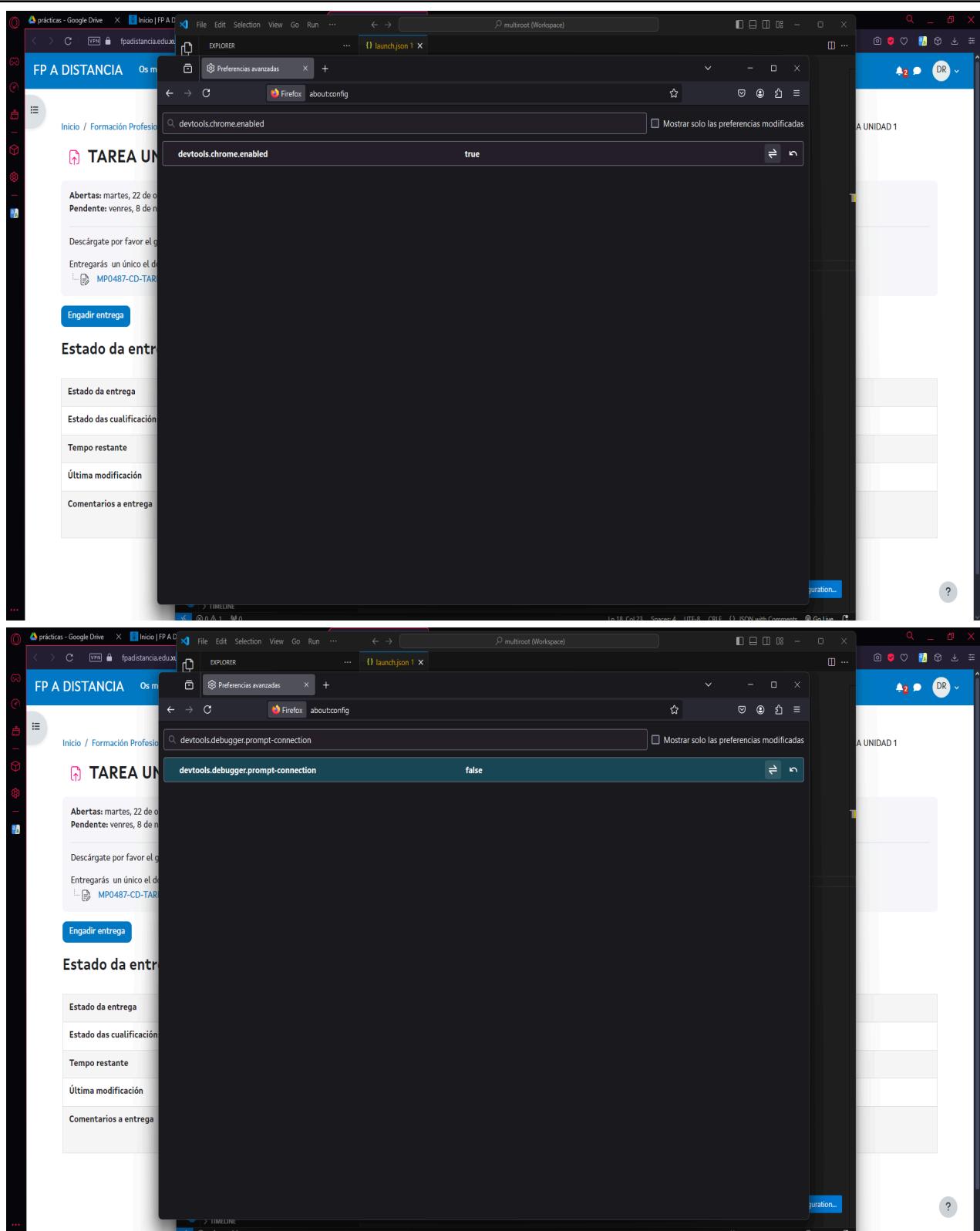
## Paso 2: Crear una Configuración de Depuración en Visual Studio Code



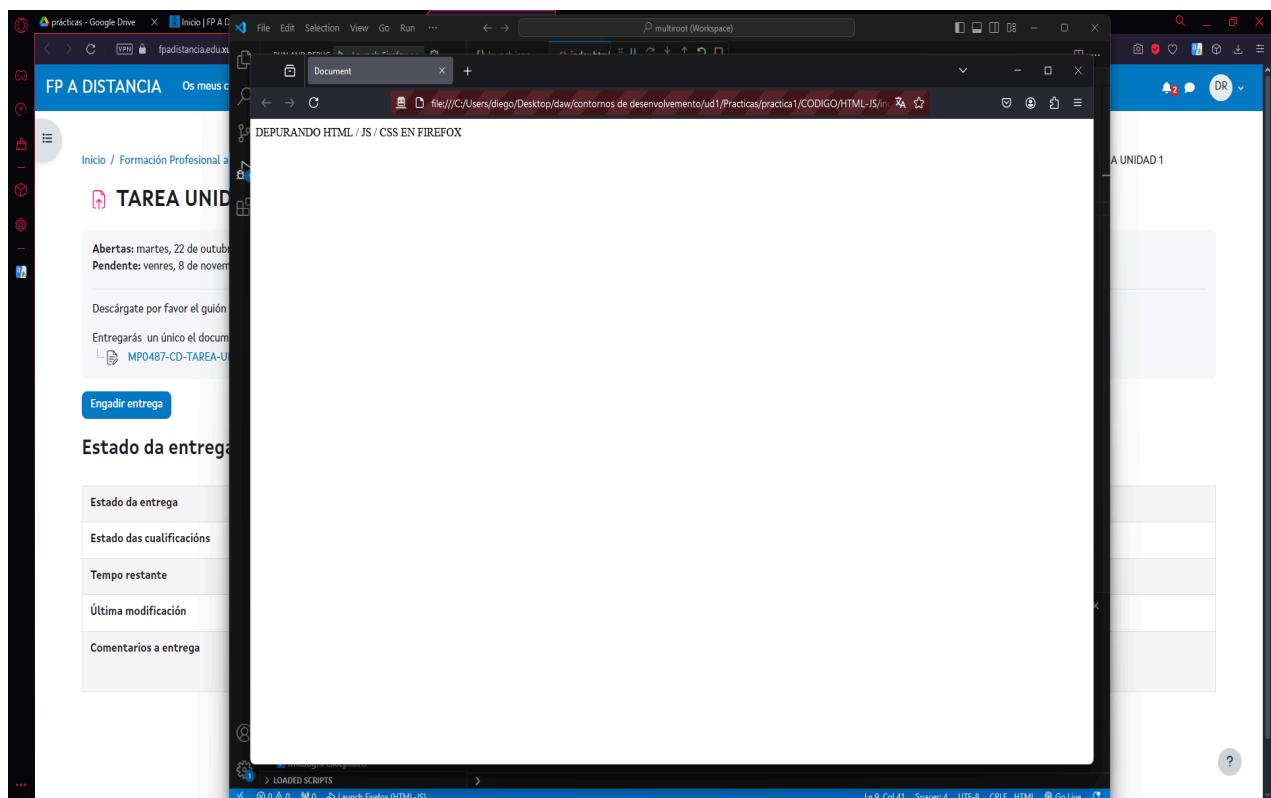
## CAPTURA/S DE PANTALLA de Firefox en las que se muestre los cambios de configuración que hubo que hacer:

### Paso 3: Configuración en Firefox





Paso 4: Iniciar la Depuración



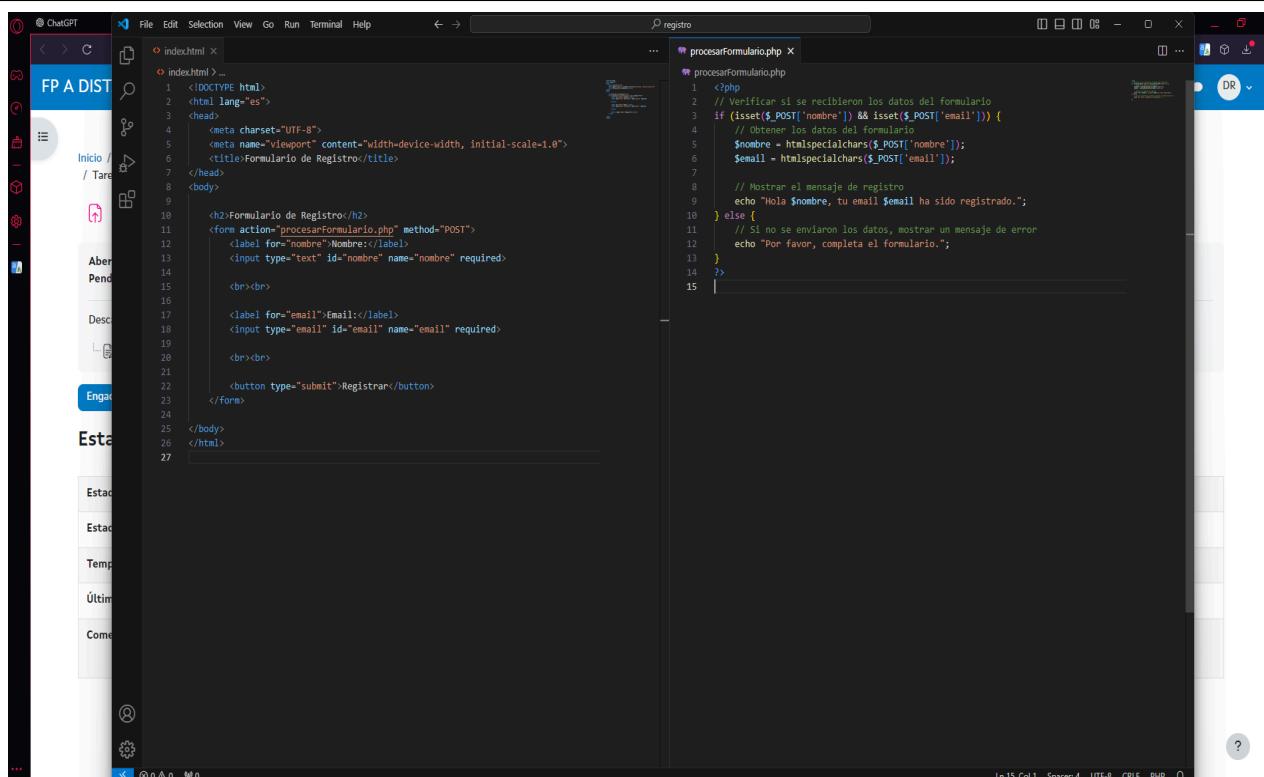
8) Configura VSC para poder depurar código PHP:

#### ENUMERACIÓN DE LOS PASOS SEGUIDOS

1. Crear una aplicación PHP.
2. Instalar la extensión XDebug.
3. Crear el archivo launch.json.

#### CAPTURA/S DE PANTALLA de VSC en las que se muestren los cambios que hubo que hacer:

Adjunto el paso a paso de la tarea realizada para desarrollo web en Contorno Servidor.



```

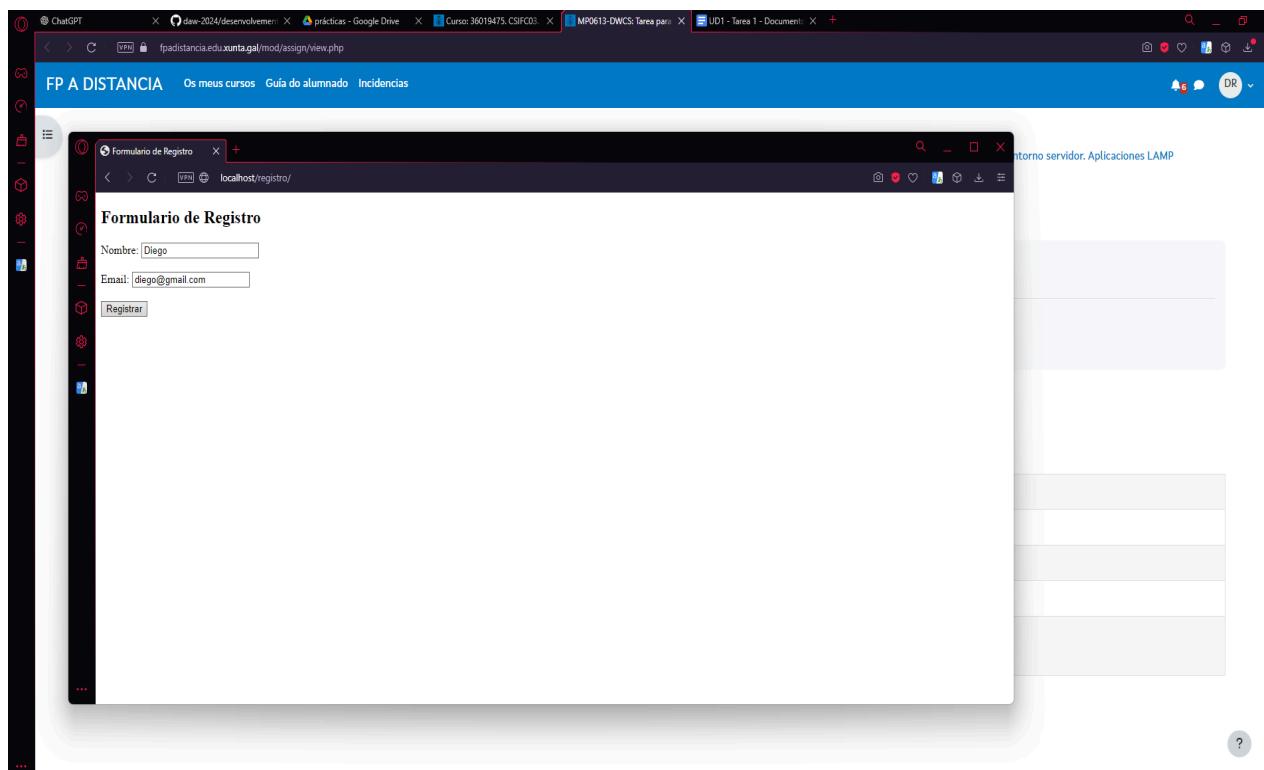
index.html
1 <!DOCTYPE html>
2 <html lang="es">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Formulario de Registro</title>
7   </head>
8   <body>
9
10    <h2>Formulario de Registro</h2>
11    <form action="procesarFormulario.php" method="POST">
12      <label for="nombre">Nombre:</label>
13      <input type="text" id="nombre" name="nombre" required>
14
15      <br><br>
16
17      <label for="email">Email:</label>
18      <input type="email" id="email" name="email" required>
19
20      <br><br>
21
22      <button type="submit">Registrar</button>
23    </form>
24
25  </body>
26 </html>
27

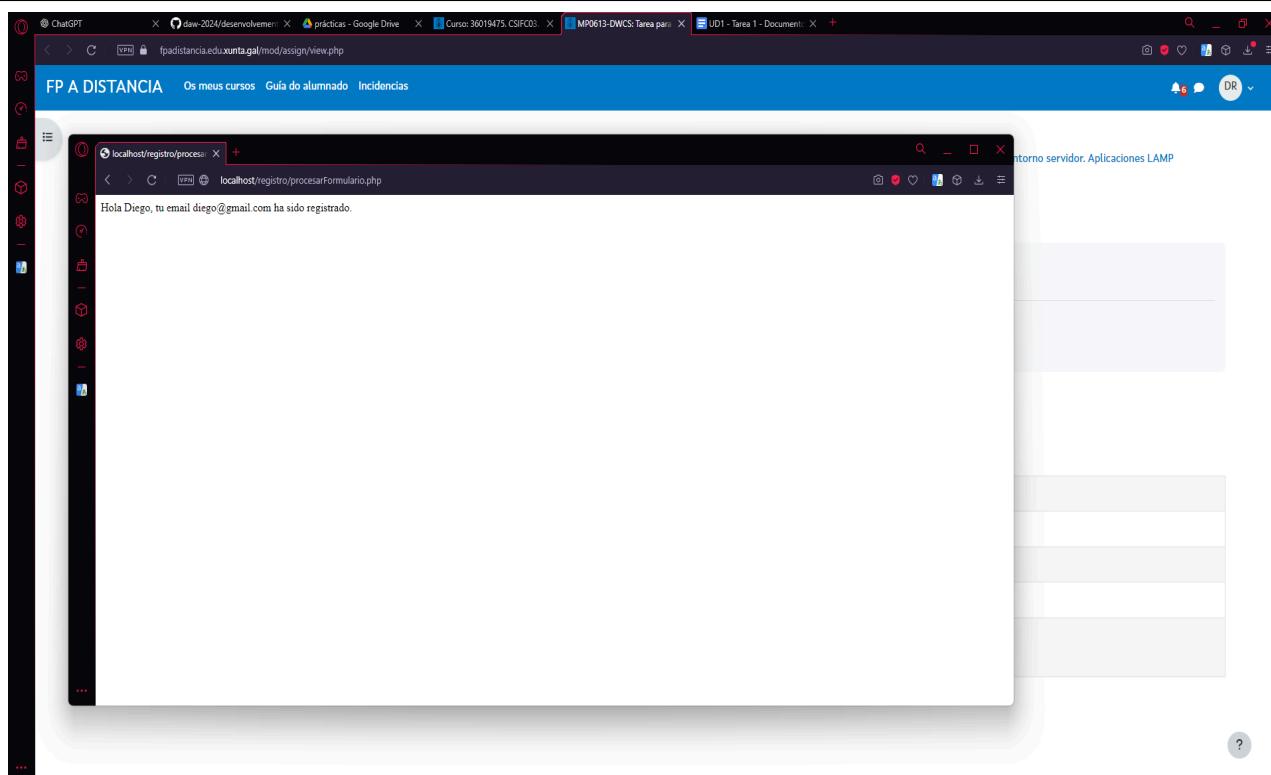
```

```

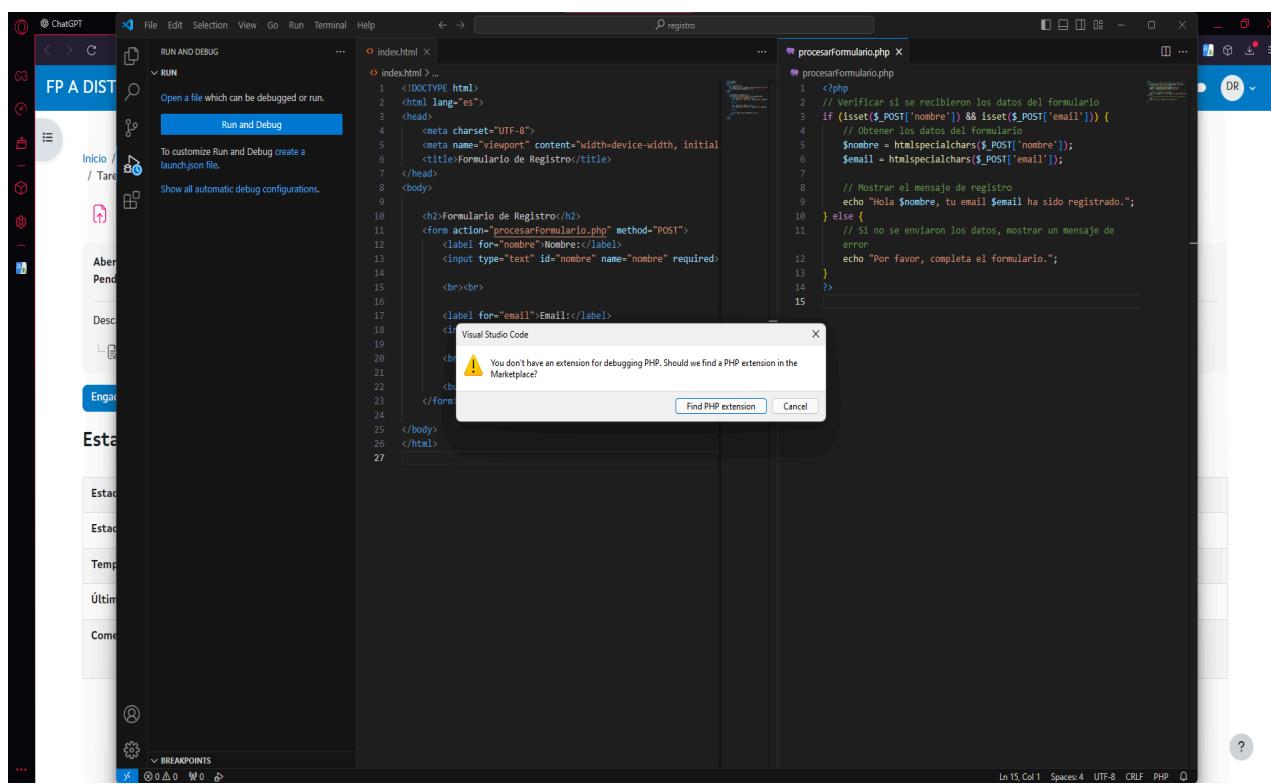
procesarFormulario.php
1 <?php
2 // Verificar si se recibieron los datos del formulario
3 if (isset($_POST['nombre']) && isset($_POST['email'])) {
4   // Obtener los datos del formulario
5   $nombre = htmlspecialchars($_POST['nombre']);
6   $email = htmlspecialchars($_POST['email']);
7
8   // Mostrar el mensaje de registro
9   echo "Hola $nombre, tu email $email ha sido registrado.";
10 } else {
11   // Si no se enviaron los datos, mostrar un mensaje de error
12   echo "Por favor, completa el formulario.";
13 }
14 ?>
15

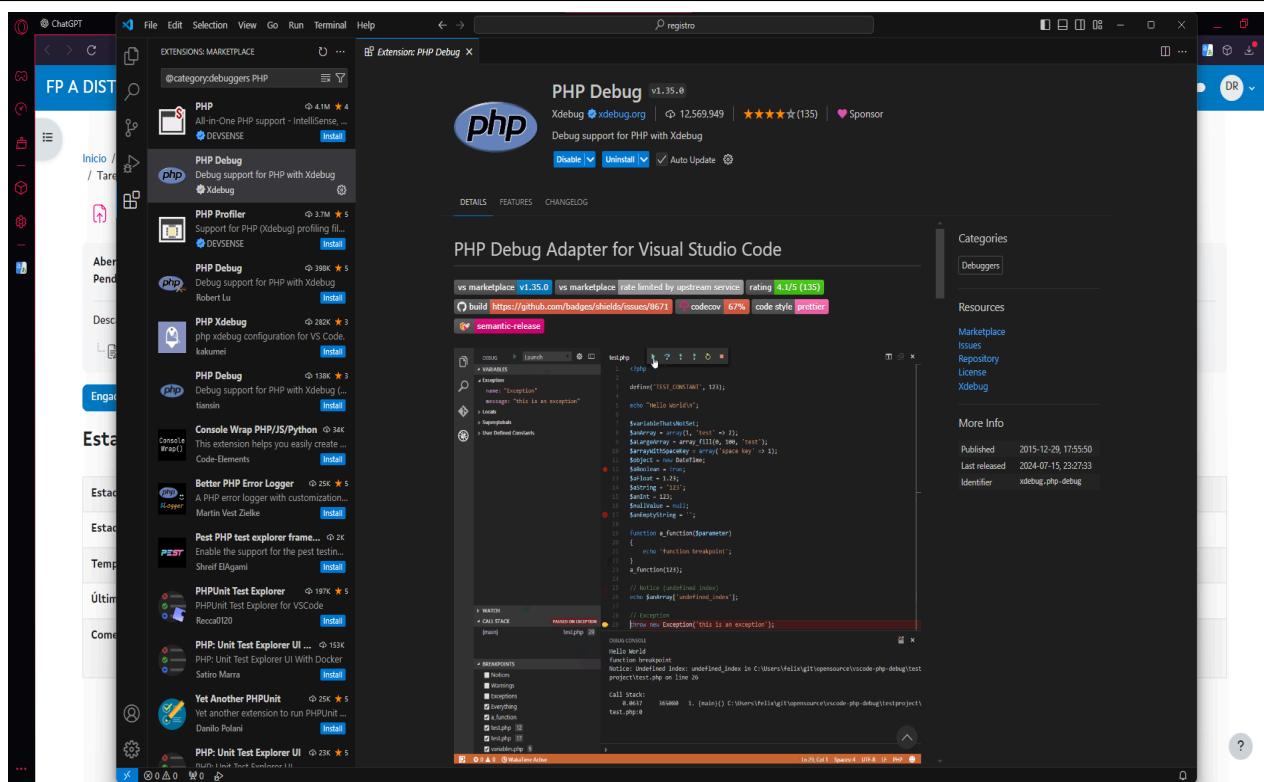
```



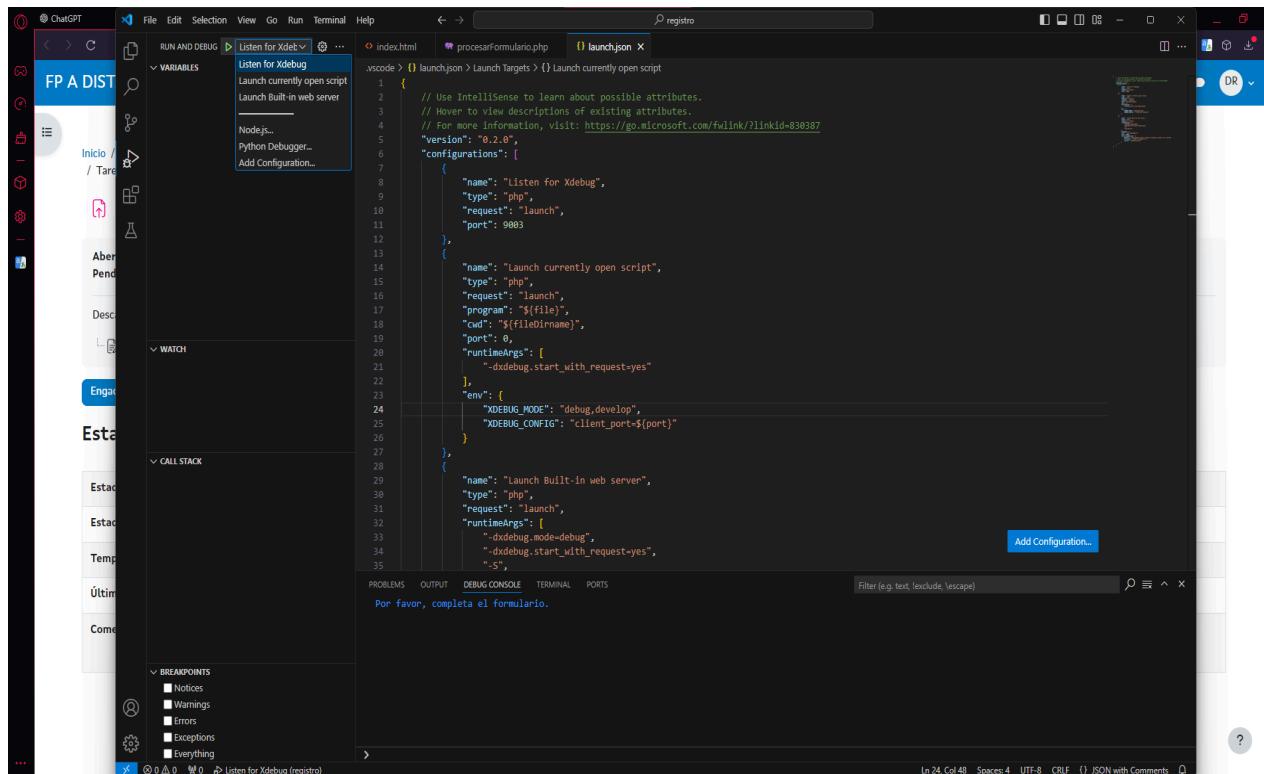


Instalamos la extensión XDebug para PHP.





Crear el archivo *launch.json* para añadir los modos de debug por defecto. Se genera de forma automática pulsando en el enlace que aparece tras instalar la extensión. Ahora ya podemos debuggear la aplicación PHP en los siguientes modos:



**TAREA 2: Problema de Git, GitHub y Visual Studio Code (4 puntos)**

Realiza las tareas siguientes:

- 1) Configura git desde línea de comandos para usar tus datos de desarrolladora/desarrollador en las siguientes tareas. Además queremos que por defecto la rama “master” se llame “main”. Deben aparecer los comandos que ejecutas y que funcionan correctamente.

**CAPTURA/S DE PANTALLA**

- 2) Configura tu cuenta 1 de GitHub en Visual Studio Code. La cuenta debe ser la que está asociada al usuario 1 @iesrodeira.gal que se te facilitó.

**CAPTURA/S DE PANTALLA**

- 3) Desde Visual Studio Code haz un clon a del siguiente repositorio de GitHub:  
<https://github.com/davidgacaeduxuntagal/Curso24-25-MP0487-CD.git>

**CAPTURA/S DE PANTALLA**

- ¿Qué comando/s git usarías desde la consola para conseguir lo mismo?

**CAPTURA/S DE PANTALLA**

- 4) Usando VSC, Crea una rama (branch) nueva y modifica el código Java en varios puntos.

**CAPTURA/S DE PANTALLA**

- ¿Qué comando git usarías desde la consola para crear la rama y pasar a esa rama directamente?

**CAPTURA/S DE PANTALLA**

- 5) En Visual Studio Code añade los cambios a la staging area y haz un commit

**CAPTURA/S DE PANTALLA**

- ¿Qué comando git usarías desde la consola para realizar lo mismo?

**CAPTURA/S DE PANTALLA**

- 6) Sincroniza los cambios con GitHub:
- ¿Te has posible? ¿Cómo lo podrías solucionar?

**CAPTURA/S DE PANTALLA**

- 7) En GitHub, crea un fork del repositorio anterior y luego crea un clon en Visual Studio Code

**CAPTURA/S DE PANTALLA**

- 8) Haz cambios en el código (rama main) y luego haz una Pull Request desde Visual Studio Code, con el mensaje “Este es mi paso 8 de la tarea 1 de Entornos de Desarrollo”

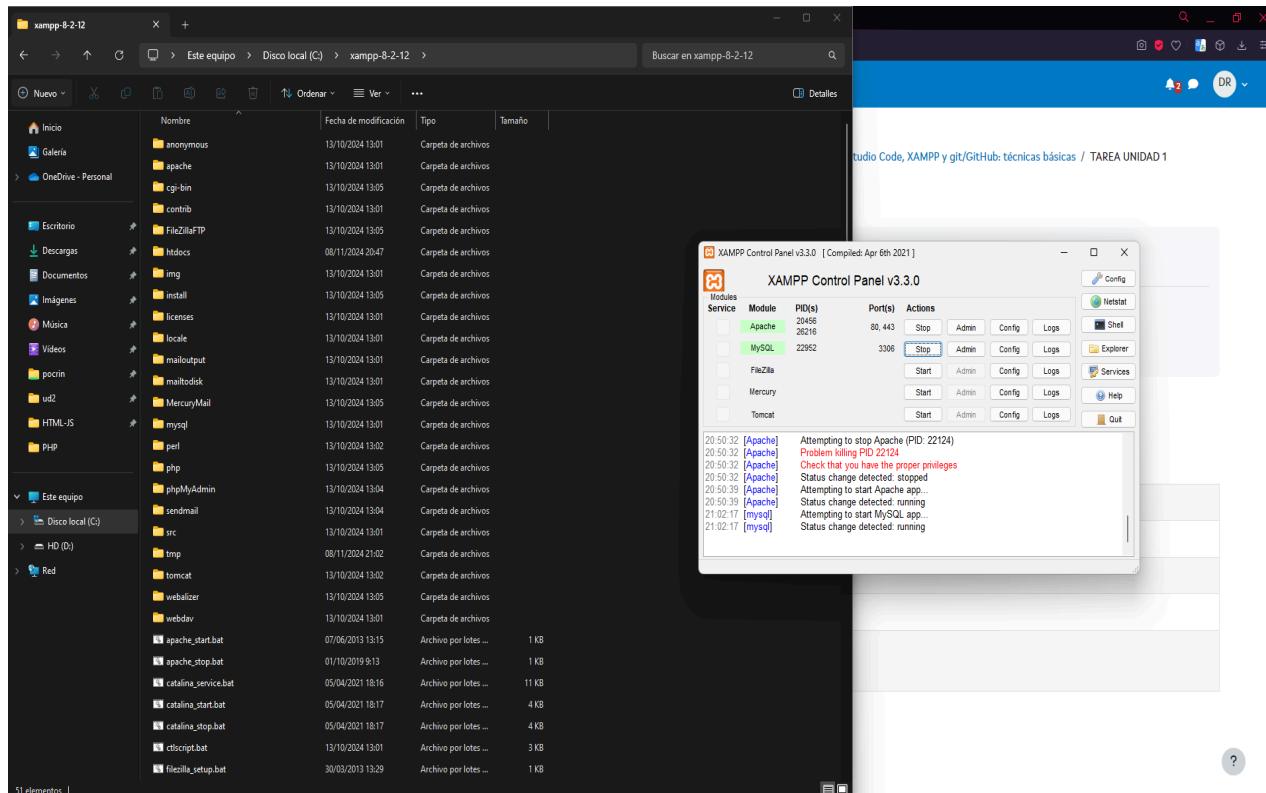
**CAPTURA/S DE PANTALLA**

### TAREA 3: Instalación y configuración de XAMPP (3 puntos)

Realiza las tareas siguientes:

- 1) Descarga e instala la versión de XAMPP .

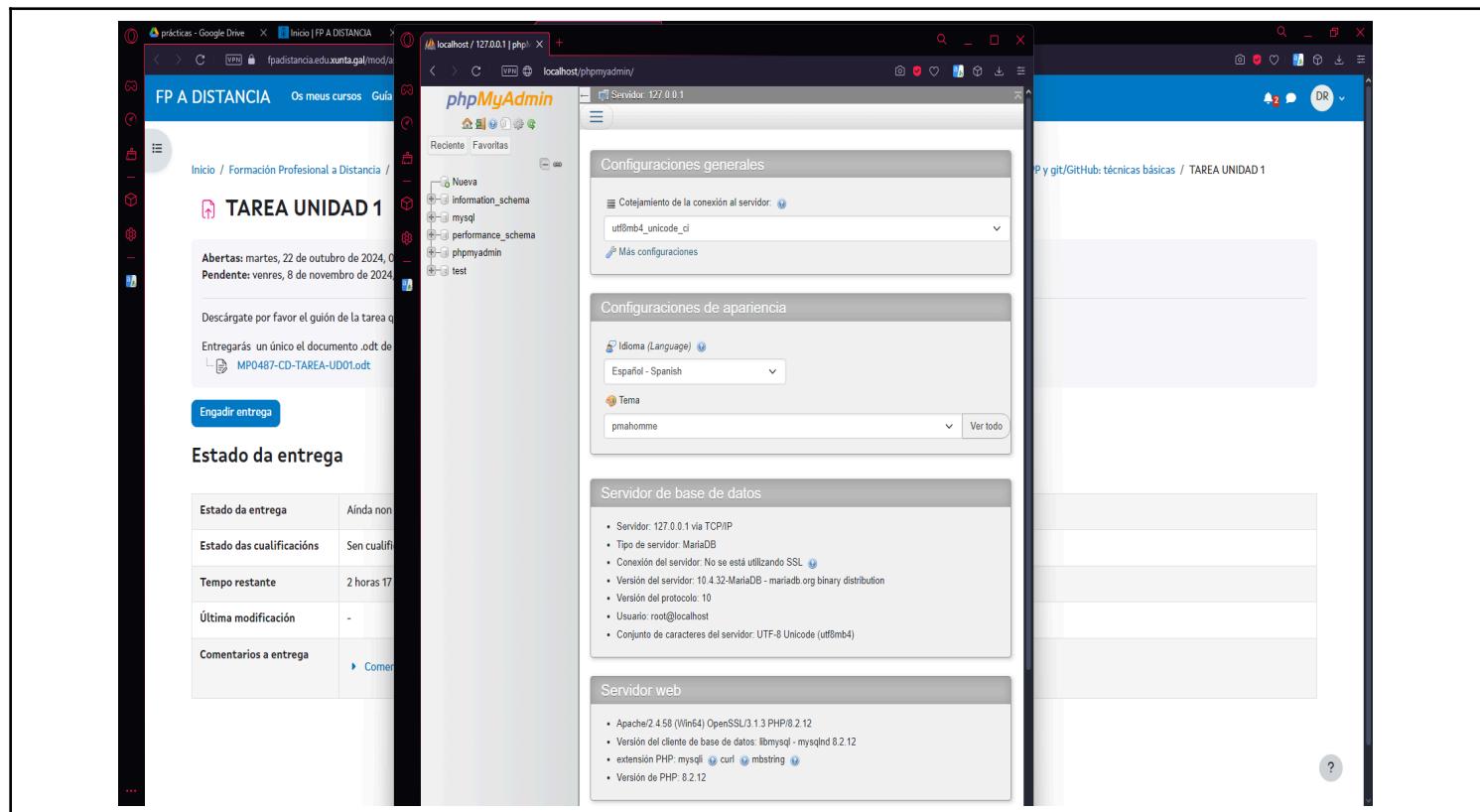
**CAPTURA/S DE PANTALLA que muestra la consola de gestión de XAMPP en donde aparecen levantados los servicios de servidor web y servidor de base de datos. En el fondo, debe aparecer una ventana del Explorador de archivos de Windows, mostrando la carpeta donde se instaló XAMPP con todas sus subcarpetas.**



- 2) **Phpmmyadmin**

Entra en phpmymyadmin.

**CAPTURA/S DE PANTALLA de phpmymyadmin con la sesión iniciada:**



### 3) XDebug

Configura XAMPP para que se pueda depurar código PHP usando XDebug.

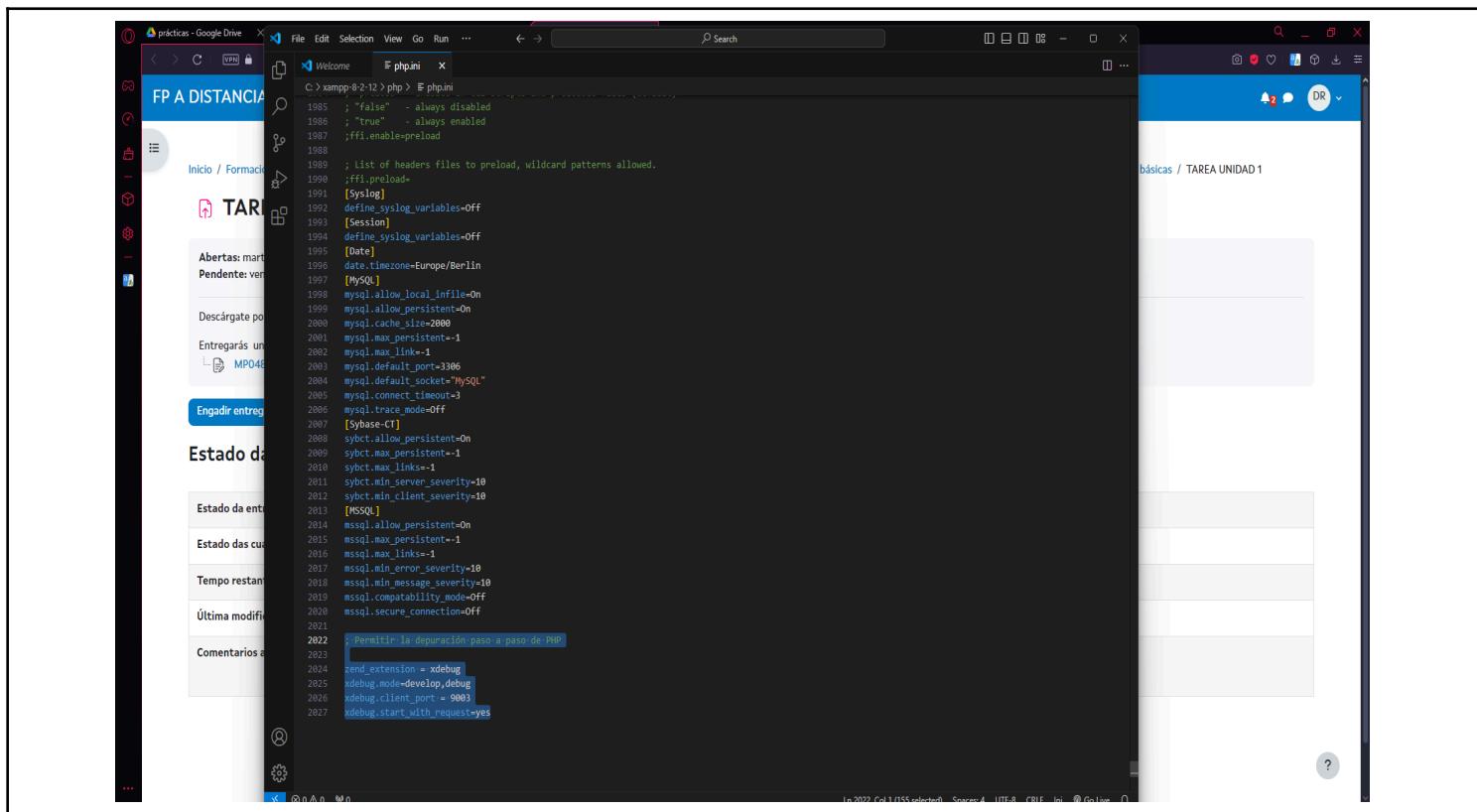
#### ENUMERACIÓN DE LOS PASOS PRINCIPALES:

Descargar la .dll adecuada: <https://xdebug.org/docs/install>

Configurar el fichero C:\xampp-8-2-12\php\php.ini para la nueva extensión.

Comprobar que funciona: función xdebug\_info().

#### CAPTURA/S DE PANTALLA de líneas añadidas al archivo de configuración de PHP:



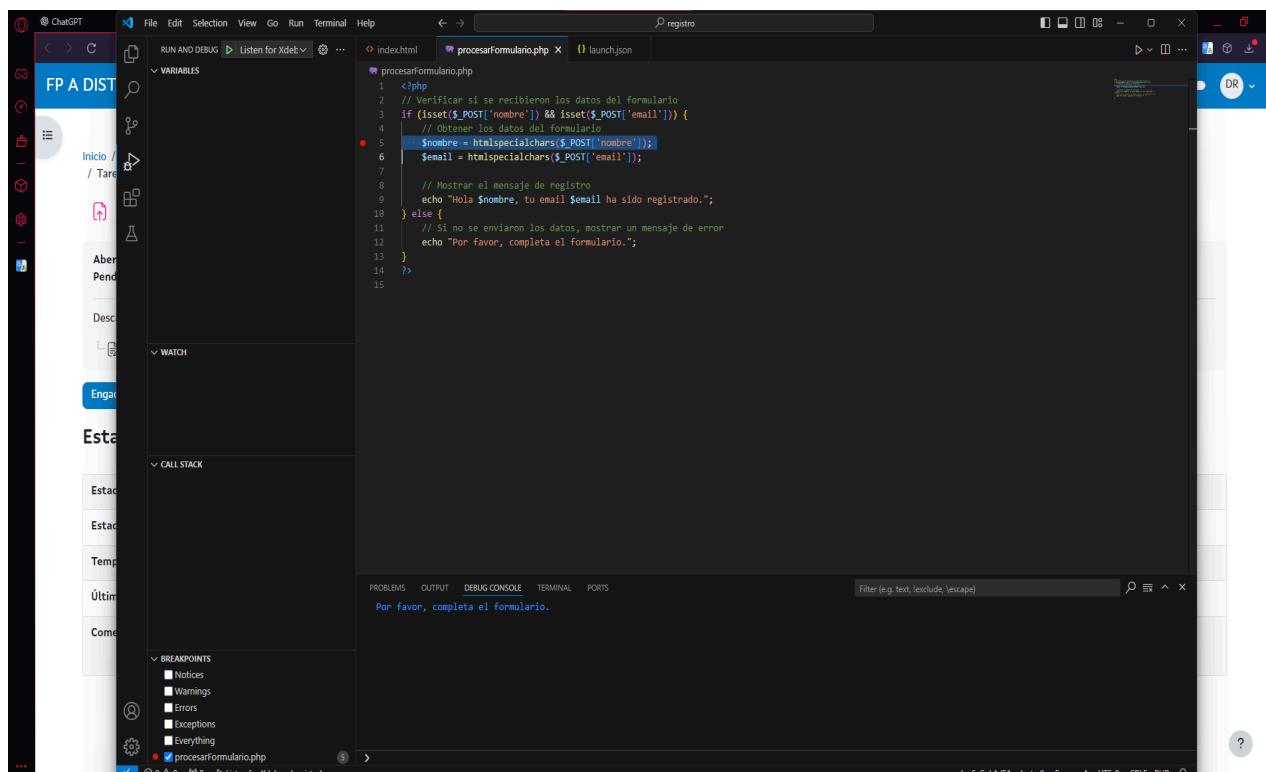
```

1985 ; "false" - always disabled
1986 ; "true" - always enabled
1987 ;ffl.enable=preload
1988
1989 ; List of headers files to preload, wildcard patterns allowed.
1990 ;ffl.preload=
1991 [Syslog]
1992 define_syslog_variables=Off
1993 [Session]
1994 define_syslog_variables=Off
1995 [Date]
1996 date.timezone=Europe/Berlin
1997 [MySQL]
1998 mysql.allow_local_infile=On
1999 mysql.allow_persistent=On
2000 mysql.cache_size=2000
2001 mysql.max_persistent=-1
2002 mysql.max_links=3
2003 mysql.default_port=3306
2004 mysql.default_socket="MySQL"
2005 mysql.connect_timeout=3
2006 mysql.trace_mode=Off
2007 [Sybase-CT]
2008 sybct.allow_persistent=On
2009 sybct.max_persistent=1
2010 sybct.max_links=1
2011 sybct.min_server_severity=10
2012 sybct.min_client_severity=10
2013 [MSSQL]
2014 mssql.allow_persistent=On
2015 mssql.max_persistent=1
2016 mssql.max_links=1
2017 mssql.min_error_severity=10
2018 mssql.min_message_severity=10
2019 mssql.compatibility_mode=Off
2020 mssql.secure_connection=Off
2021
2022 ; Permitir la depuración paso a paso de PHP
2023 zend_extension = xdebug
2024 xdebug.mode=develop,debug
2025 xdebug.client_port = 9003
2026 xdebug.start_with_request=yes
2027

```

## CAPTURA/S DE PANTALLA de VSC con un breakpoint/punto de ruptura capturado al abrir un ejemplo de código PHP.

De nuevo, adjunto el paso a paso que seguí para la tarea de Desarrollo web en Contorno Servidor.

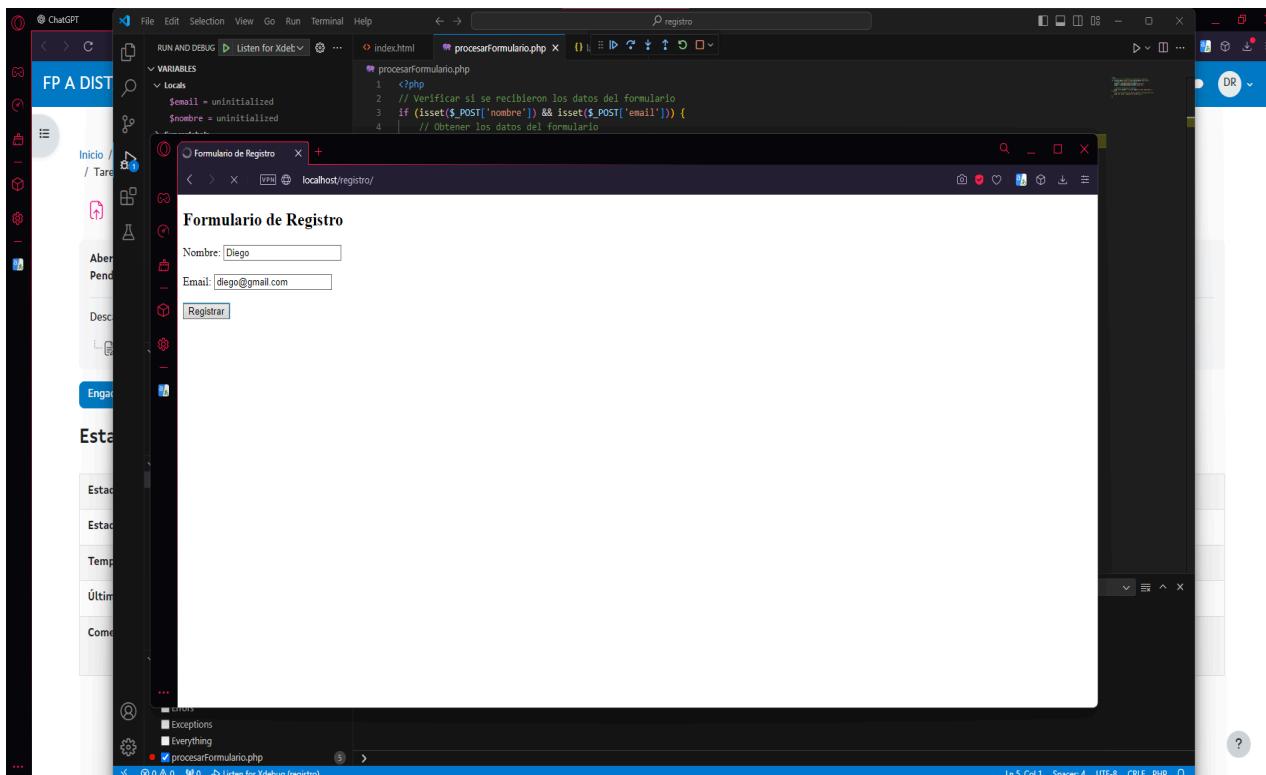


```

1 <?php
2 // Verificar si se recibieron los datos del formulario
3 if (isset($_POST['nombre']) && isset($_POST['email'])) {
4     // Obtener los datos del formulario
5     $nombre = htmlspecialchars($_POST['nombre']);
6     $email = htmlspecialchars($_POST['email']);
7
8     // Mostrar el mensaje de registro
9     echo "Hola $nombre, tu email $email ha sido registrado.";
10 } else {
11     // Si no se enviaron los datos, mostrar un mensaje de error
12     echo "Por favor, completa el formulario.";
13 }
14 ?>
15

```

Al enviar los datos se detiene en el breakpoint, por eso las variables aparecen sin valor en el debug de VSC.



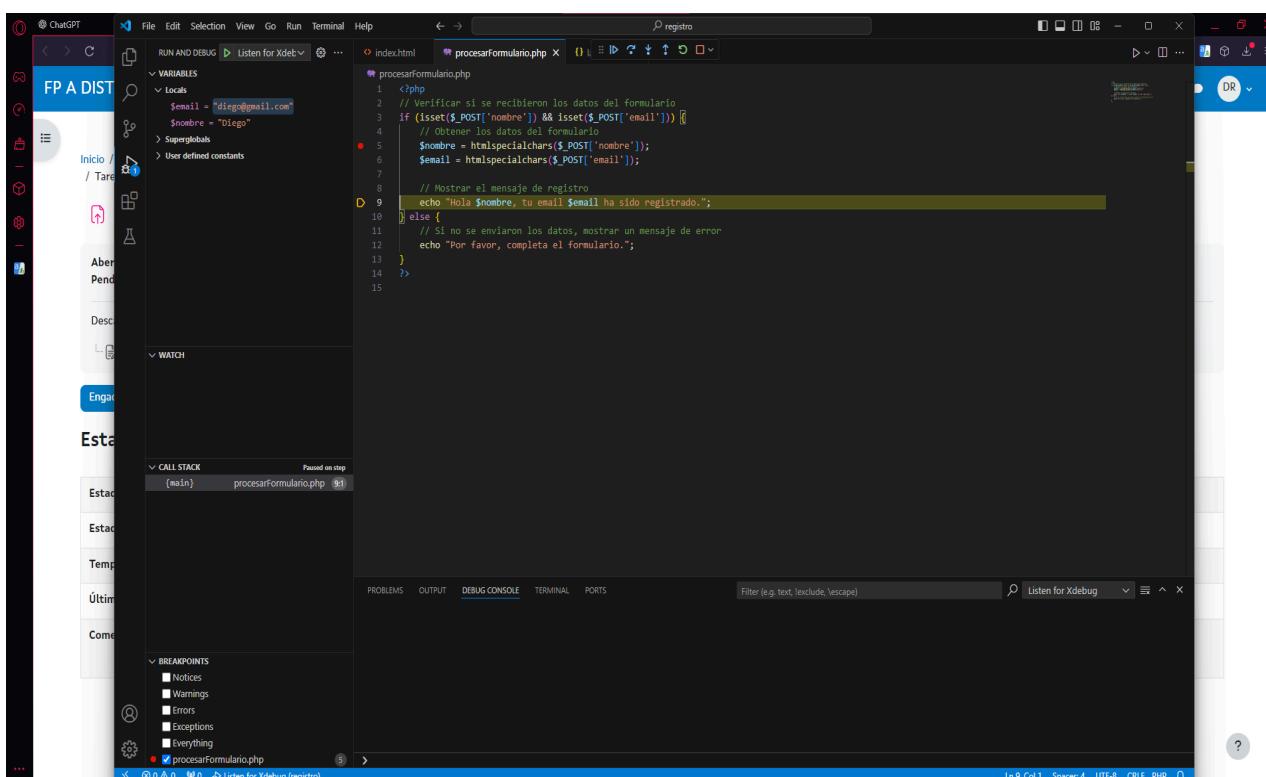
```

<?php
// Verifican si se recibieron los datos del formulario
if (isset($_POST['nombre']) && isset($_POST['email'])) {
    // Obtener los datos del Formulario
    $nombre = htmlspecialchars($_POST['nombre']);
    $email = htmlspecialchars($_POST['email']);

    // Mostrar el mensaje de registro
    echo "Hola $nombre, tu email $email ha sido registrado.";
} else {
    // Si no se enviaron los datos, mostrar un mensaje de error
    echo "Por favor, completa el formulario.";
}

```

Le doy a avanzar, la ejecución continua y las variables toman su valor.



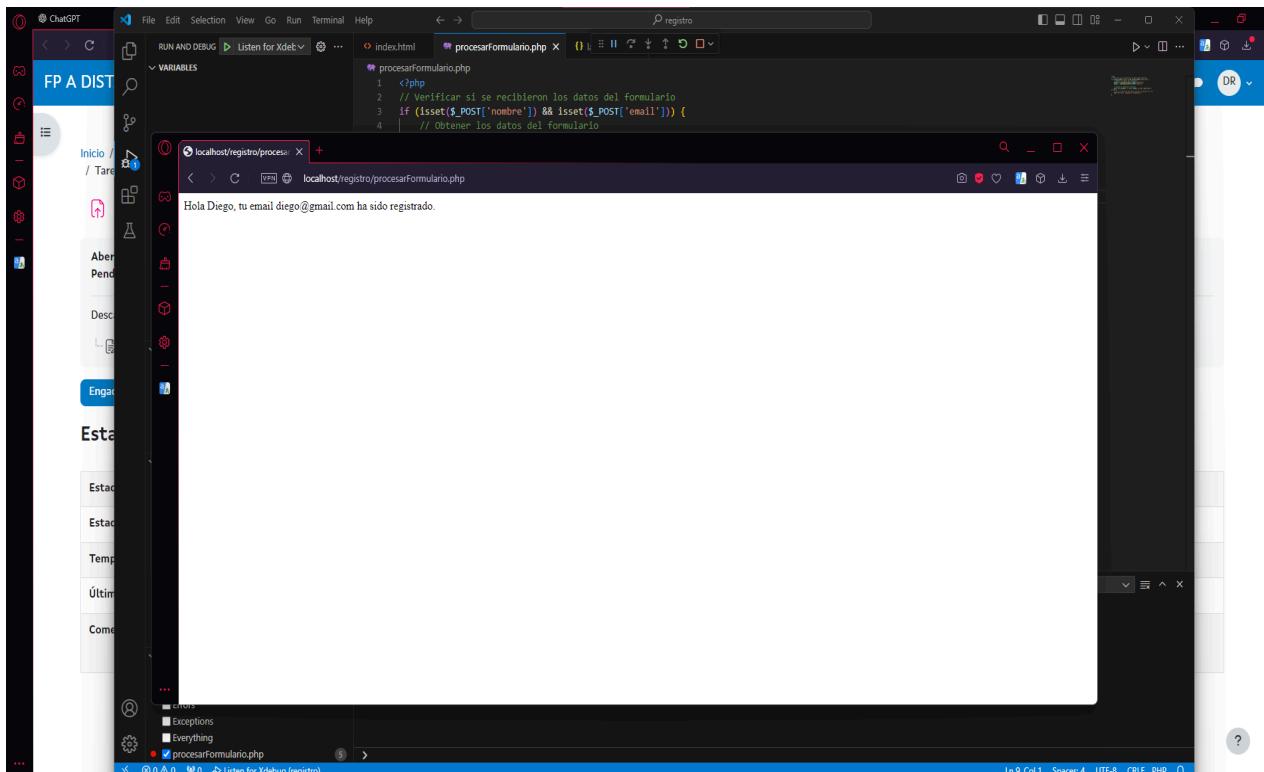
```

<?php
// Verifican si se recibieron los datos del formulario
if (isset($_POST['nombre']) && isset($_POST['email'])) {
    // Obtener los datos del Formulario
    $nombre = htmlspecialchars($_POST['nombre']);
    $email = htmlspecialchars($_POST['email']);

    // Mostrar el mensaje de registro
    echo "Hola $nombre, tu email $email ha sido registrado.";
} else {
    // Si no se enviaron los datos, mostrar un mensaje de error
    echo "Por favor, completa el formulario.";
}

```

Los datos se procesan y se muestra el resultado de la ejecución.



The screenshot shows the Xdebug interface in a development environment. On the left, there's a sidebar with various tools and links. The main area has tabs for 'index.html' and 'procesarFormulario.php'. The code editor shows the following PHP script:

```
<?php  
// Verificar si se recibieron los datos del formulario  
if (isset($_POST['nombre']) && isset($_POST['email'])) {  
    // Obtener los datos del formulario
```

To the right of the code editor is a browser window displaying the output of the script: "Hola Diego, tu email diego@gmail.com ha sido registrado."

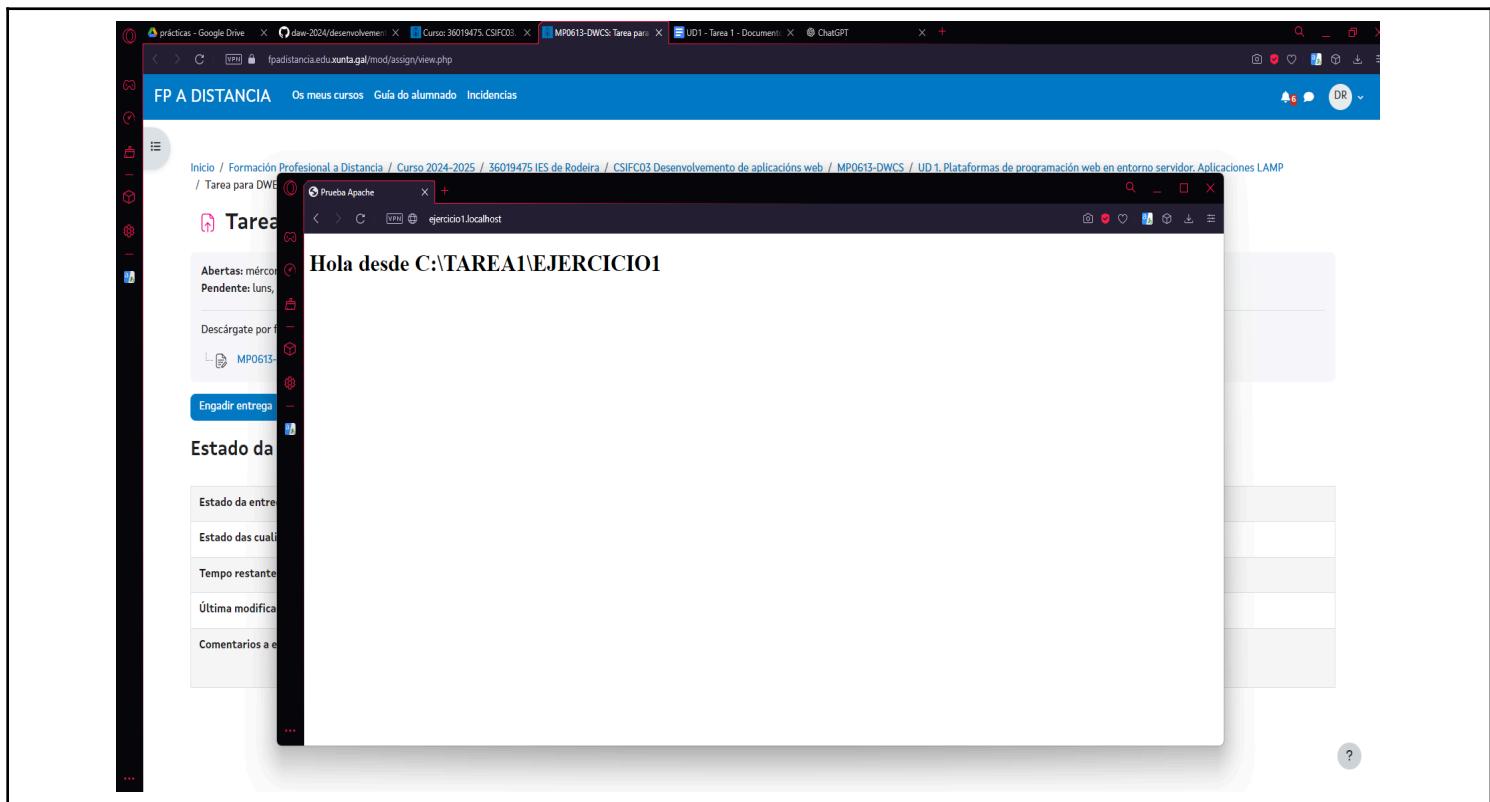
#### 4) Hosts virtuales de Apache

Configura un host virtual de Apache para que sirva la carpeta C:\TAREA-CD cuando se accede a la dirección <http://tarea1-cd.localhost> desde el navegador de tu elección. Mete dentro de C:\TAREA-CD una página web de muestra (puedes copiar y pegar de los ejemplos mencionados previamente).

#### ENUMERACIÓN DE LOS PASOS PRINCIPALES:

#### CAPTURA/S DE PANTALLA del acceso correcto a la página web <http://tarea1-cd.localhost>

<http://ejercicio1.localhost>



**CAPTURA/S DE PANTALLA del contenido relevante del fichero de configuración de Apache (httpd) que te permite hacer lo anterior:**

Agregar lo siguiente al archivo `C:\xampp-8-2-12\apache\conf\extra\httpd-vhosts.conf`.

