

Informe de la E2 del MF0493_3

Índice

Actividad a Realizar.....	1
Script de Instalación.....	3
Base de datos para el registro de email.....	5
Script de la base de datos:.....	5
Código de la aplicación (index.php):.....	6
Anexo A.....	10
Anexo B.....	11
Pruebas realizadas al Script de instalación de componentes en la Raspberry.....	11
Pruebas realizadas a la aplicación.....	13

Actividad a Realizar.



Acciones formativas cofinanciadas por el Gobierno de Canarias y el Servicio Público de Empleo Estatal

E ₂			
HOJA DE INSTRUCCIONES ALUMNADO			
Especialidad:	(IFCD0210) DESARROLLO DE APLICACIONES CON TECNOLOGÍAS WEB	Nº Curso:	19-38/002274
Módulo formativo:	MF0493_3: Implantación de aplicaciones web en entornos internet, intranet y extranet		
Práctica:	Verificación de funcionalidades y rendimiento		
Nombre y Apellidos:			
DNI:		Fecha:	
		Calificación:	

Esta prueba tendrá un peso de **30%** junto con la suma del resto de evaluaciones del aprendizaje.

1. DESCRIPCIÓN DE LA PRÁCTICA

Crea un script con todos los comandos necesarios para instalar en un Raspbian cada una de las siguientes aplicaciones software:

1. Apache
2. Nginx
3. PHP
4. MySQL
5. Servidor Ftp

Crea una aplicación web que tenga solo la funcionalidad de registrar emails. Los emails se guardarán en una tabla. Cada email debe ser único y se debe verificar que tenga formato de email.

Sube el código de la aplicación a tu repositorio de GitHub junto con el script de instalación de las aplicaciones y un documento con la batería de pruebas realizada para verificar el correcto funcionamiento de la aplicación.

Acciones formativas cofinanciadas por el Gobierno de Canarias y el Servicio Público de Empleo Estatal

2. INSTRUCCIONES ESPECÍFICAS

La prueba se realizará de manera telemática. Para la realización de la prueba puede utilizar internet y apuntes. No puede comunicarse de ninguna manera con los compañeros ya que es una prueba individual. Dispondrá de 5 minutos para leer la prueba y consultar posibles dudas.

La prueba empezará cuando el profesor lo indique, teniendo una duración máxima 360 minutos.

Una vez finalizado el tiempo para su realización, tendrán que entregar la hoja de instrucciones, debidamente cumplimentada con su nombre y DNI al docente.

Los criterios a evaluar serán los siguientes:

- La implementación correcta de los diagramas de clase.
- El diseño de los diagramas de clase.
- La calidad del código

Una vez finalizada la prueba entregar vía email la dirección del repositorio GitHub con al correo viernesrayco@gmail.com con el asunto E2 MF0943_3. Indique su nombre y apellidos en el cuerpo del email.

3. EQUIPO Y MATERIAL

Ordenador por alumno/a con internet
Apuntes
Editores de texto plano y navegadores
Herramientas de ofimática
Raspberry

Script de Instalación.

Se procedió a la creación de un Script para la instalación de los distintos componentes que se solicitan en la hoja de actividades insertada anteriormente, el script hace la instalación completamente automatizada y al finalizar, a modo de batería de pruebas, ejecuta varios comandos que nos muestran por pantalla la confirmación de la instalación y la versión instalada de cada uno de los componentes solicitados.

Para mayor información diríjase al archivo **README.MD** del repositorio indicado en el anexo "A" del presente documento y a los comentarios presentes en el mismo Script.

```
#!/bin/bash

if [ "$(whoami)" != "root" ]; then
    echo "Run script as ROOT please. (sudo !!)"
    exit
fi

## Instalamos apache
apt-get install apache2 -y

## Instalamos PHP
apt-get install php php-mysql libapache2-mod-php -y

## Instalar MariaDB
apt-get install mariadb-server mariadb-client -y

## Securizamos la instalación de MariDB
mysql_secure_installation

## Instalamos phpmyadmin
apt-get install phpmyadmin -y

## Cambiamos el propietario del directorio /html que es en donde irá
nuestras páginas, se agrega el usuario pi al grupo
## y se cambian los permisos de lectura y escritura
chown -R www-data:www-data /var/www/html
usermod -g www-data pi
chmod -R 777 /var/www

echo
```

```
echo
echo

## El siguiente comando reinicia apache2 para que corra phpmyadmin
service apache2 reload

## Vamos ahora a instalar Nginx
apt-get install nginx -y
## Atención: El servicio se intentará iniciar en el puerto 80 y no podrá hacerlo por que
## ahí estará corriendo el servicio de Apache.

## Finalmente vamos a instalar Vsftp
apt-get install vsftpd -y
## Ya podremos usar el servidor ftp con las credenciales de los usuarios del sistema

#####
## Ahora vamos a realizar las comprobaciones de la instalación realizada
#####

##Limpia la pantalla
clear

## Vemos la versión de apache
echo "La versión de apache es: "
apache2 -v

## Versión de Nginx
echo
echo "La versión de Nginx es: "
nginx -v

## Versión de PHP
echo
echo "Versión de PHP es: "
php -v

## Version de Vsftp
echo
echo "Versión de Vsftp es: "
vsftpd -v

## Para la comprobación de mariadb será necesario loguearse en el cli del SGBD con las
## credenciales creadas

echo "Instalación finalizada!!!!"
```

Base de datos para el registro de email.

Se procedió a la creación de una base de datos llamada **email** para el registro de email, con la condición de que cada email debería ser único (esta comprobación la llevará a cabo la base de datos) y se debe comprobar la introducción de un email con el formato correcto (esta comprobación la llevará a cabo la aplicación).

Script de la base de datos:

```
-- phpMyAdmin SQL Dump
-- version 5.0.2
-- https://www.phpmyadmin.net/
--
-- Servidor: 127.0.0.1
-- Tiempo de generación: 14-01-2021 a las 13:43:10
-- Versión del servidor: 10.4.14-MariaDB
-- Versión de PHP: 7.4.10

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Base de datos: `email`
--

--
-- Estructura de tabla para la tabla `email`
--

DROP TABLE IF EXISTS `email`;
CREATE TABLE `email` (
  `idEmail` int(11) NOT NULL,
  `email` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--
-- Índices para tablas volcadas
--

--
-- Indices de la tabla `email`
--
ALTER TABLE `email`
  ADD PRIMARY KEY (`idEmail`),
  ADD UNIQUE KEY `email` (`email`);

--
-- AUTO_INCREMENT de las tablas volcadas
--

--
-- AUTO_INCREMENT de la tabla `email`
--
ALTER TABLE `email`
  MODIFY `idEmail` int(11) NOT NULL AUTO_INCREMENT;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

Código de la aplicación (index.php):

```
<!DOCTYPE HTML>
<html>

<head>
  <style>
    .error {
      color: #FF0000;
    }
  </style>
</head>

<body>

  <?php
    //Datos de conexión a la base de datos.
    $db = [
      'host' => 'localhost',
```



```

        'username' => 'root',
        'password' => '',
        'db' => 'email' //Cambiar al nombre de tu base de datos
    ];

    //Abrir conexion a la base de datos
    function connect($db)
    {
        try {
            $conn = new PDO("mysql:host={$db['host']};dbname={$db['db']};charset=utf8", $db['username'], $db['password']);

            // set the PDO error mode to exception
            $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

            return $conn;
        } catch (PDOException $exception) {
            exit($exception->getMessage());
        }
    }

    // define e inicializa variables
    $emailErr = "";
    $email = "";
    $msg = "";

    // Hace una depuración de los datos eliminando espacios en blanco, barras,
    //caracteres especiales.

    function test_input($data)
    {
        $data = trim($data);
        $data = stripslashes($data);
        $data = htmlspecialchars($data);
        return $data;
    }

    // Comprobamos que el email esté informado y correctamente formado

    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        if (empty($_POST["email"])) {
            $emailErr = "Email is required";
        } else {
            $email = test_input($_POST["email"]);

            if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
                $emailErr = "Invalid email format";
            } else {

                // Si todas las condiciones son válidas abrimos
                // la conexión y hacemos la consulta.
            }
        }
    }

```

```

$dbConn = connect($db);
$sql = "INSERT INTO email
      (idEmail, email)
      VALUES
      (null,:email)";
$stmt = $dbConn->prepare($sql);
$stmt->bindValue(':email', $email);

// Capturamos cualquier error
try {
    $stmt->execute();
} catch (\Throwable $th) {
    $error = $th;
}

// Recoge el valor del último registro
$postId = $dbConn->lastInsertId();
$dbConn = "";

// Definimos el mensaje de resultado

if ($postId) {
    $msg = "Email registrado exitosamente con id: {$postId}";
} else {
    $msg = "Lo siento, pero su email no ha podido ser registrado debido a: {$error}";
}

}

}
?>
<!-- Este es el Html de la aplicación -->
<h2>Registro y validación de emails</h2>
<p><span class="error">* required field</span></p>
<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
    E-mail: <input type="text" name="email" value="<?php echo $email; ?>">
    <span class="error">* <?php echo $emailErr; ?></span>
    <br><br>
    <input type="submit" name="submit" value="Submit">
</form>

<?php
// Emitimos mensaje de confirmación

echo "<h2>Your Input:</h2>";

echo $msg;
echo "<br>";

```

```
    ?>  
</body>  
</html>
```

Anexo A

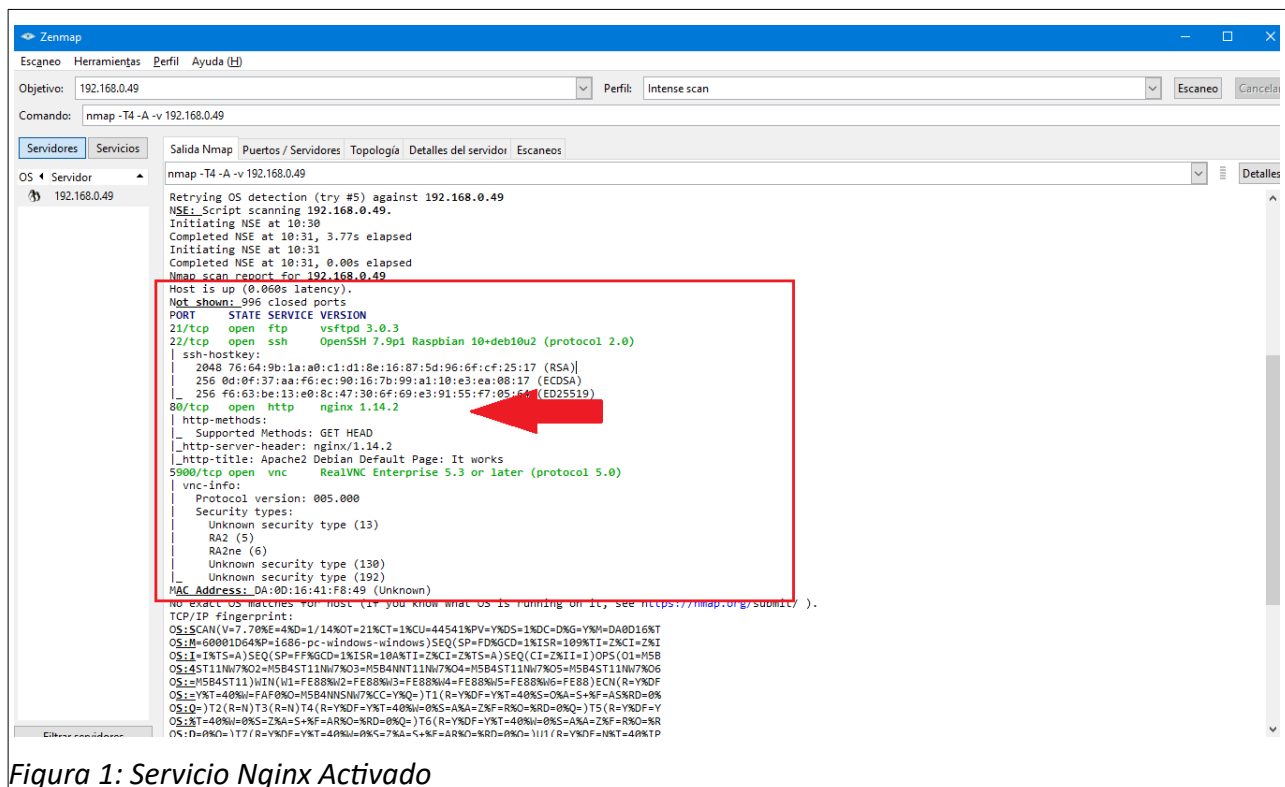
Todo el Código Fuente se puede descargar del repositorio en:

https://github.com/jcfr4n/E2_MF0493_3.git

Anexo B

Pruebas realizadas al Script de instalación de componentes en la Raspberry.

En la Raspberry como ya pudimos observar el mismo Script contiene la comprobación de las versiones de los componentes instalados, además se analizó la ip con Zenmap (Interfaz gráfica de windows para la herramienta Nmap) en donde se puede observar los servicios que están corriendo en ella.



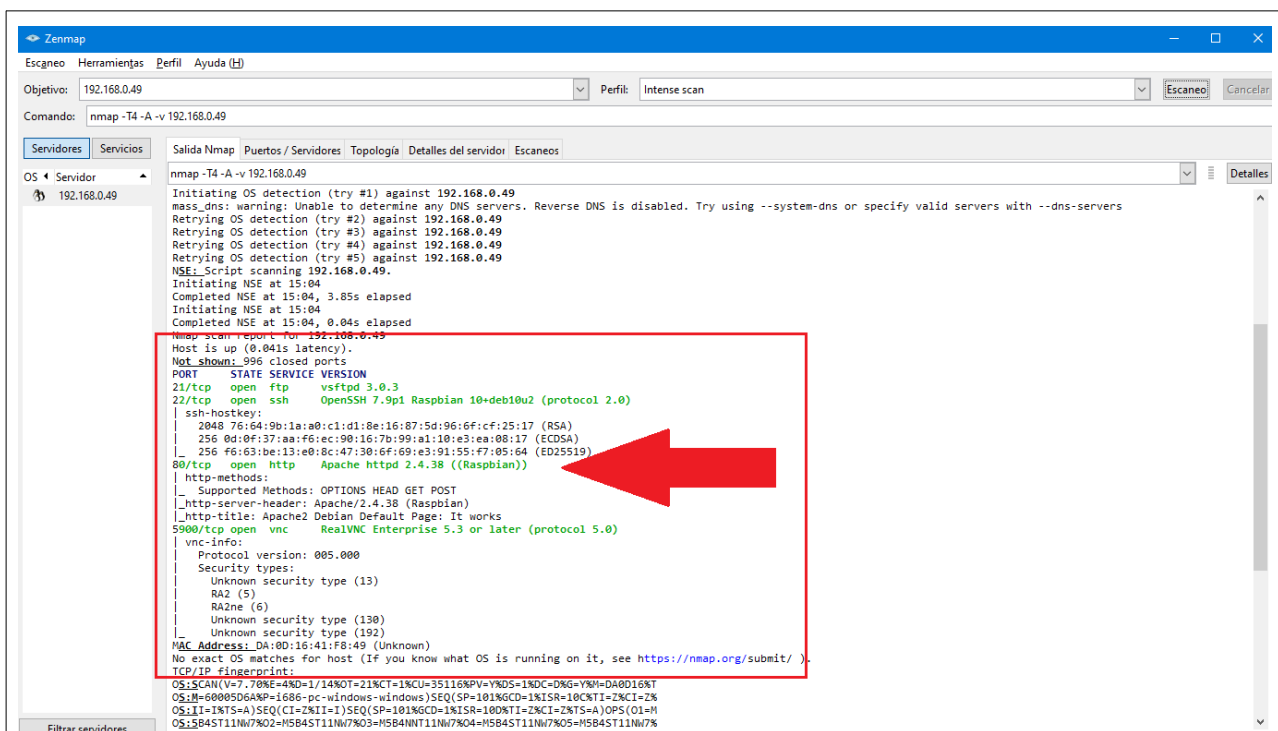


Figura 2: Corriendo el servicio Apache

Pruebas realizadas a la aplicación



Figura 3: Pantalla inicial de aplicación



Figura 4: Submit sin rellenar el campo

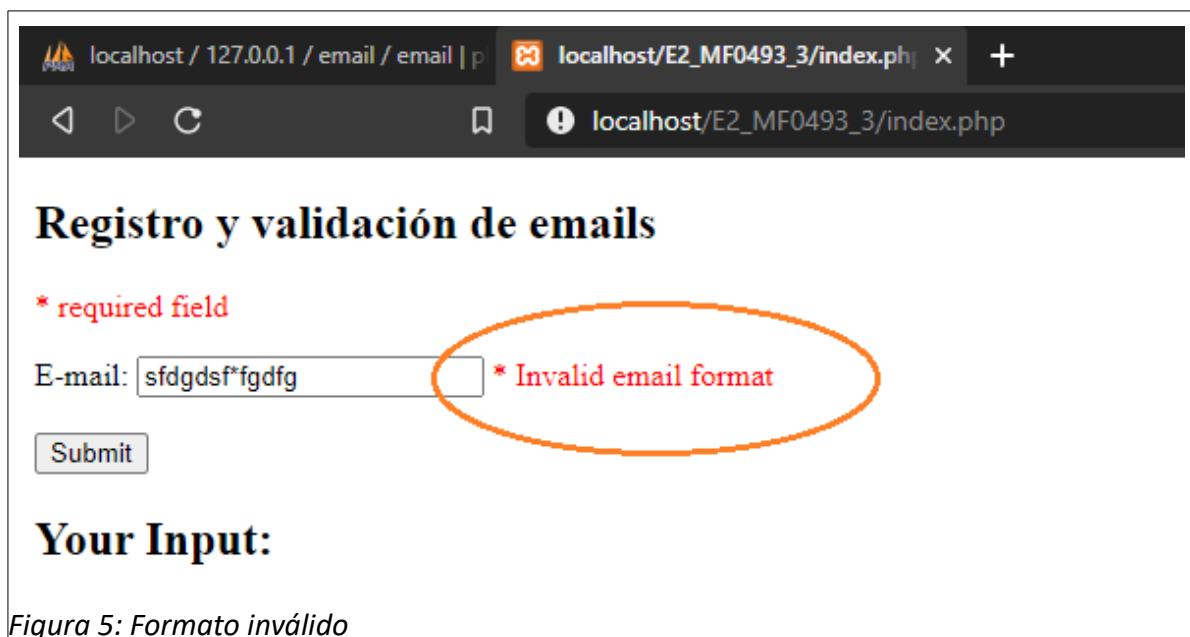


Figura 5: Formato inválido



Figura 6: Email duplicado

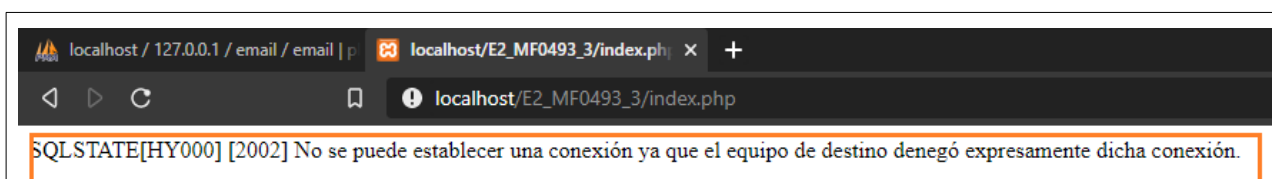


Figura 7: Base de datos inaccesible



Figura 8: Email registrado correctamente