

PRACTICA

Tarea para DWES02.

REALIZADO POR:

AHMED HASSAN KHAMIS



Sumario

Descripción del enunciado.....	3
Requisitos extra.....	3
Descripción de la solución.....	4
Descripción lógica/funcional de cada componente.....	4
Descripción técnica de cada componente.....	6
Manual de despliegue.....	8



Descripción del enunciado

Debes programar una aplicación para mantener una pequeña agenda en una única página web programada en PHP sin acceso a base de datos o persistencia de cualquier tipo.

La agenda almacenará únicamente dos datos de cada persona: su nombre y un número de teléfono: no podrá haber nombres repetidos en la agenda.

En la parte superior de la página web se mostrará el contenido de la agenda que tenga en cada momento. En la parte inferior debe figurar un sencillo formulario con dos cuadros de texto, uno para el nombre y otro para el número de teléfono.

Cada vez que se envíe el formulario:

- Si el nombre está vacío, se mostrará una advertencia.
- Si el nombre que se introdujo no existe en la agenda, y el número de teléfono no está vacío, se añadirá a la agenda.
- Si el nombre que se introdujo ya existe en la agenda y se indica un número de teléfono, se sustituirá el número de teléfono anterior.
- Si el nombre que se introdujo ya existe en la agenda y no se indica número de teléfono, se eliminará de la agenda la entrada correspondiente a ese nombre.

Requisitos extra

- Generar la estructura de la página PHP.
- Mostrar los contactos existentes en la agenda.
- Generar el formulario de introducción de nuevo contacto.
- Introducir los datos de la agenda como campos ocultos en el formulario.
- Comprobar los datos enviados por el formulario, mostrando una advertencia cuando no se cubre el nombre.
- Introducir en la agenda los datos de un nuevo contacto.
- Modificar el teléfono de un contacto ya existente.
- Eliminar de la agenda un contacto.
- Utilizar un array asociativo.
- Introducir comentarios en el código.

Descripción de la solución

La solución aplicada para este aplicativo es el uso de un campo oculto para almacenar datos de la agenda, para luego con cada envío recibir los datos desde variables de servidor y imprimir mediante php toda la pagina.

Descripción técnica de cada componente.

Para entender las capturas los comentarios están insertado **TRAS CADA CAPTURA/S.**

```
// Variables
error_reporting(E_ERROR);
ini_set('display_errors', '1');

// echo $_POST['agenda']; para comprobar los datos del array

//seccion donde se comprueba que no esta definida la variable y se inicializa.
if (!isset($_POST['agenda'])) {
    $_POST['agenda'] = "";
}

// Array asociativo con los Datos
$agenda = [];

//separar datos de agenda
$contactos = explode(";", $_POST['agenda']);

//Introducir los datos y separar por nombre y telefono en array asociativo
for ($i = 0; $i < count($contactos); $i++) {
    $campos = [];
    $campos = explode("-", $contactos[$i]);
    $agenda[$campos[0]] = $campos[1];
}
```

las primeras lineas son para mostrar los errores ocurridos en el aplicativo en la propia pagina. Tras eso se encuentra un check de la variable post['agenda'] para comprobar si la primera vez llega vacía que se inicialice. Luego se inicializa el array agenda. Se dividen los parámetros de post agenda (esto seria efectivo tras el primer envío) con explode para poder tratarlos en la siguiente sección que trata de rellenar el array agenda tras volver a separar el array de contactos en mini arrays de 2 el cual contendrá para cada contacto un nombre y un teléfono.

```
//comprobaciones sobre el envio de datos. Agregar y borrar contactos
if (isset($_POST['enviar']) && $_POST['nombre'] != "" && $_POST['telefono'] != "") {
    $agenda[$_POST['nombre']] = $_POST['telefono'];
} elseif (isset($_POST['enviar']) && $_POST['nombre'] != "" && $_POST['telefono'] == "") {
    unset($agenda[$_POST['nombre']]);
} else {
    $_POST['agenda'] .= "";
}
```

para esta parte se comprueba los datos introducidos acorde a los requisitos de la practica y se manejan según lo solicitado mediante la función isset y algunas condicionales, tras esto se procede a agregar el nuevo contacto (si pasa el filtro) dentro de agenda, en caso de que no pase el primer filtro procedemos a comprobar si cumple la condición de borrado, si la cumple se procede a borrar el registro mediante la función de unset. Y por ultimo en caso de no ser de las anteriores se procede a no agregar ni modificar nada en la agenda.

En caso de preguntarse donde esta la sección de edición.. no existe ya que se trabaja con arrays asociativos lo cual nos permite que no haya redundancia de datos y en caso de insertarse un usuario ya guardado lo único que se hace es sobrescribir mediante las ventajas de los arrays asociativos.

```
//imprimir formulario
print_r("<!DOCTYPE html>
<html lang='en'>
<head>
  <meta charset='UTF-8'>
  <meta name='viewport' content='width=device-width, initial-scale=1.0'>
  <title>Document</title>
  <style>
    body {
      margin: 30px;
    }
    form{
      width: fit-content;
      padding: 10px;
      display: flex;
      flex-direction: column;
    }
  </style>
</head>
<body>
  <h1>Agenda</h1>
  <table>
    <tr><th>NOMBRE</th><th>TELEFONO</th></tr>
  </table>
  <hr>
  <h2>Agregar Contacto</h2>
  <form action='' method='post' style='border: 1px solid black;'>
    <label for='nombre'>Nombre:</label>
    <input id='nombre' name='nombre' type='text' placeholder='Introduce nombre aqui...'>
    <br>
    <label for='telefono'>Telefono:</label>
    <input id='telefono' name='telefono' type='number' min=1 placeholder='Introduce telefono aqui...'>
    <br>
    <input type='text' id='agenda' name='agenda' value='.' $_POST['agenda'] . " hidden>
    <input id='enviar' name='enviar' type='submit' value='Enviar'>
  </form>
</body>
</html>");
```

```
//continuacion del html
print_r("</table>
<hr>
<h2>Agregar Contacto</h2>
<form action='' method='post' style='border: 1px solid black;'>
  <label for='nombre'>Nombre:</label>
  <input id='nombre' name='nombre' type='text' placeholder='Introduce nombre aqui...'>
  <br>
  <label for='telefono'>Telefono:</label>
  <input id='telefono' name='telefono' type='number' min=1 placeholder='Introduce telefono aqui...'>
  <br>
  <input type='text' id='agenda' name='agenda' value='.' $_POST['agenda'] . " hidden>
  <input id='enviar' name='enviar' type='submit' value='Enviar'>
</form>
</body>
</html>");
```

```
//continuacion del html
print_r("</table>
<hr>
<h2>Agregar Contacto</h2>
<form action='' method='post' style='border: 1px solid black;'>
  <label for='nombre'>Nombre:</label>
  <input id='nombre' name='nombre' type='text' placeholder='Introduce nombre aqui...'>
  <br>
  <label for='telefono'>Telefono:</label>
  <input id='telefono' name='telefono' type='number' min=1 placeholder='Introduce telefono aqui...'>
  <br>
  <input type='text' id='agenda' name='agenda' value='.' $_POST['agenda'] . " hidden>
  <input id='enviar' name='enviar' type='submit' value='Enviar'>
</form>
</body>
</html>");
```

secciones donde se imprime partes del formulario con la función `print_r` y también se inserta en un campo oculto llamado `agenda` en el formulario los datos de agenda mediante la variable `post` de `agenda`.

```
//check de nombre introducido para dar color al borde del campo
if (isset($_POST['enviar']) && $_POST['nombre'] == "") {
  print_r("<input type='text' {
    border-color:red;
  }");
}
```

```
//check de nombre para imprimir en caso de ausencia al enviar el formulario
if (isset($_POST['enviar']) && $_POST['nombre'] == "") {
  print_r("<span>RELLENA ESTE CAMPO</span>");
}
```

secciones donde se muestra al usuario que le falta en el formulario algunos campos mediante condicionales y como se describió en otras secciones la función `isset`, `print_r`.

```
//introducir datos en post de agenda tras actualizar con el nuevo registro
//y imprimir los datos en tabla de la agenda
foreach ($agenda as $key => $value) {
  $_POST['agenda'] .= $key . "-" . $value . ";";
  if ($key) {
    print_r("<tr><td>" . $key . "</td><td>" . $value . "</td></tr>");
  }
}
```

en esta sección se imprimen los contactos mediante un bucle foreach que recorre el array asociativo y agrega al post de agenda que anteriormente se procedió con su vaciamiento para poder sobrescribirlo con los datos en el array asociativo. Tras eso se imprimen en filas y celdas dentro de una tabla para su posterior lectura por parte del usuario final.

Descripción lógica/funcional de cada componente.

Para esta sección el código se divide en varias partes, por no hacer repetitivo el documento ni los comentarios aplicados en la solución. Para entender las capturas los comentarios están insertado **TRAS CADA CAPTURA/S**.

```
// Variables
error_reporting(E_ERROR);
ini_set('display_errors', '1');

// echo $_POST['agenda']; para comprobar los datos del array

//seccion donde se comprueba que no esta definida la variable y se inicializa.
if (!isset($_POST['agenda'])) {
    $_POST['agenda'] = "";
}

// Array asociativo con los Datos
$agenda = [];

//separar datos de agenda
$contactos = explode(";", $_POST['agenda']);

//introducir los datos y separar por nombre y telefono en array asociativo
for ($i = 0; $i < count($contactos); $i++) {
    $campos = [];
    $campos = explode("-", $contactos[$i]);
    $agenda[$campos[0]] = $campos[1];
}
}
```

En esta sección se inicializa variables y se tratan los datos a procesar mas adelante ademas de activar la notificación de errores.

```
//comprobaciones sobre el envio de datos. Agregar y borrar contactos
if (isset($_POST['enviar']) && $_POST['nombre'] != "" && $_POST['telefono'] != "") {
    $agenda[$_POST['nombre']] = $_POST['telefono'];
} elseif (isset($_POST['enviar']) && $_POST['nombre'] != "" && $_POST['telefono'] == "") {
    unset($agenda[$_POST['nombre']]);
} else {
    $_POST['agenda'] .= "";
}
}
```

para esta parte se comprueba los datos introducidos acorde a los requisitos de la practica y se manejan según lo solicitado.

NOTA: soy consciente de que el programa no esta haciendo chequeos de espacios o caracteres especiales y demás, pero esto se debe a que en los requisitos establecidos no se pide dicho código aparte del gran hincapié que se hizo en las clases de no hacer algo que no se solicitó.

```
//imprimir formulario
print_r("<!DOCTYPE html>
<html lang='en'>
<head>
  <meta charset='UTF-8'>
  <meta name='viewport' content='width=device-width, initial-scale=1.0'>
  <title>Document</title>
  <style>
    body {
      margin: 30px;
    }
    form{
      width: fit-content;
      padding: 10px;
      display: flex;
      flex-direction: column;
    }
  </style>
</head>
<body>
  <h1>Agenda</h1>
  <table>
    <tr><th>NOMBRE</th><th>TELEFONO</th></tr>
  </table>
  <hr>
  <h2>Agregar Contacto</h2>
  <form action='' method='post' style='border: 1px solid black;'>
    <label for='nombre'>Nombre:</label>
    <input id='nombre' name='nombre' type='text' placeholder='Introduce nombre aqui...'>
    <br>
    <label for='telefono'>Telefono:</label>
    <input id='telefono' name='telefono' type='number' min=1 placeholder='Introduce telefono aqui...'>
    <br>
    <input type='text' id='agenda' name='agenda' value='.' . $_POST['agenda'] . " hidden"
    <input id='enviar' name='enviar' type='submit' value='Enviar'>
  </form>
</body>
</html>");
```

```
//continuacion del html
print_r("</table>
<hr>
<h2>Agregar Contacto</h2>
<form action='' method='post' style='border: 1px solid black;'>
  <label for='nombre'>Nombre:</label>");
```

```
//continuacion del html
print_r("</label>
<input id='nombre' name='nombre' type='text' placeholder='Introduce nombre aqui...'>
<br>
<label for='telefono'>Telefono:</label>
<input id='telefono' name='telefono' type='number' min=1 placeholder='Introduce telefono aqui...'>
<br>
<input type='text' id='agenda' name='agenda' value='.' . $_POST['agenda'] . " hidden"
<input id='enviar' name='enviar' type='submit' value='Enviar'>
</form>
</body>
</html>");
```

secciones donde se imprime partes del formulario.

```
//check de nombre introducido para dar color al borde del campo
if (isset($_POST['enviar']) && $_POST['nombre'] == "") {
  print_r("input[type='text'] {
    border-color:red;
  }");
}
```

```
//check de nombre para imprimir en caso de ausencia al enviar el formulario
if (isset($_POST['enviar']) && $_POST['nombre'] == "") {
  print_r('<span>RELLENA ESTE CAMPO</span>');
}
```

secciones donde se muestra al usuario que le falta en el formulario.

```
//introducir datos en post de agenda tras actualizar con el nuevo registro
//y imprimir los datos en tabla de la agenda
foreach ($agenda as $key => $value) {
  $_POST['agenda'] .= $key . "-" . $value . ";";
  if ($key) {
    print_r("<tr><td>" . $key . "</td><td>" . $value . "</td></tr>");
  }
}
```

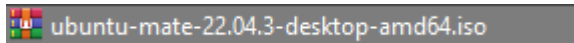
en esta sección imprimo los contactos.

Manual de despliegue.

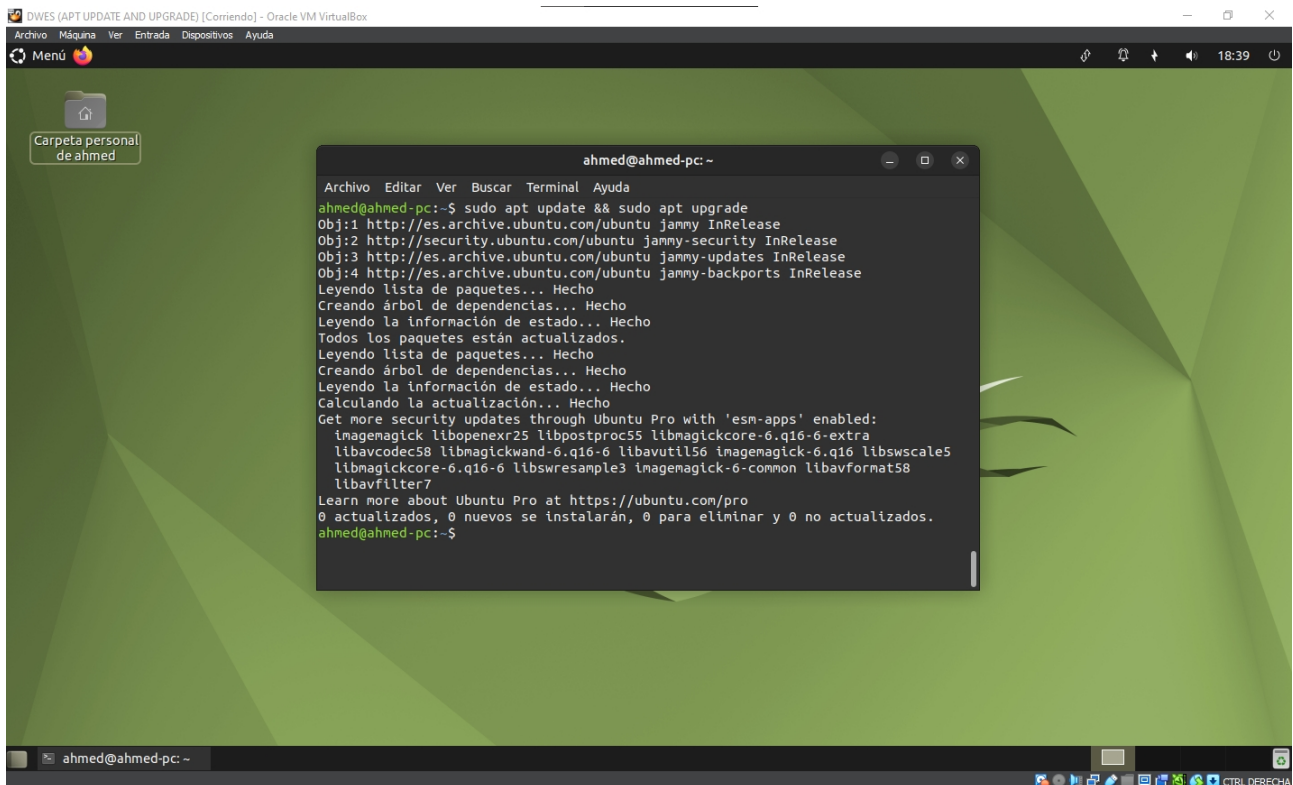
Para esta sección tengo que puntuar en que los siguientes requisitos voy a omitir su instalación en profundidad y poner solo el comando necesario de inicio porque se da por hecho que este montado el servidor de dicha manera, en caso contrario ponerse en contacto con el desarrollador.

Requisitos mínimos para ejecutar el aplicativo:

- Linux (server o desktop).

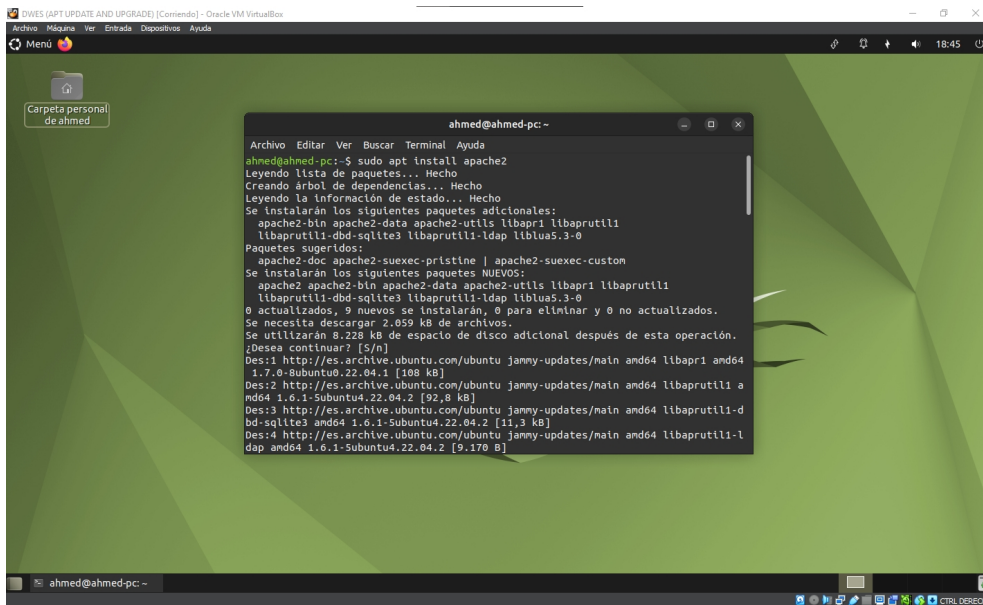


- Ejecutar actualizaciones → `sudo apt update && apt upgrade`



- Apache2.

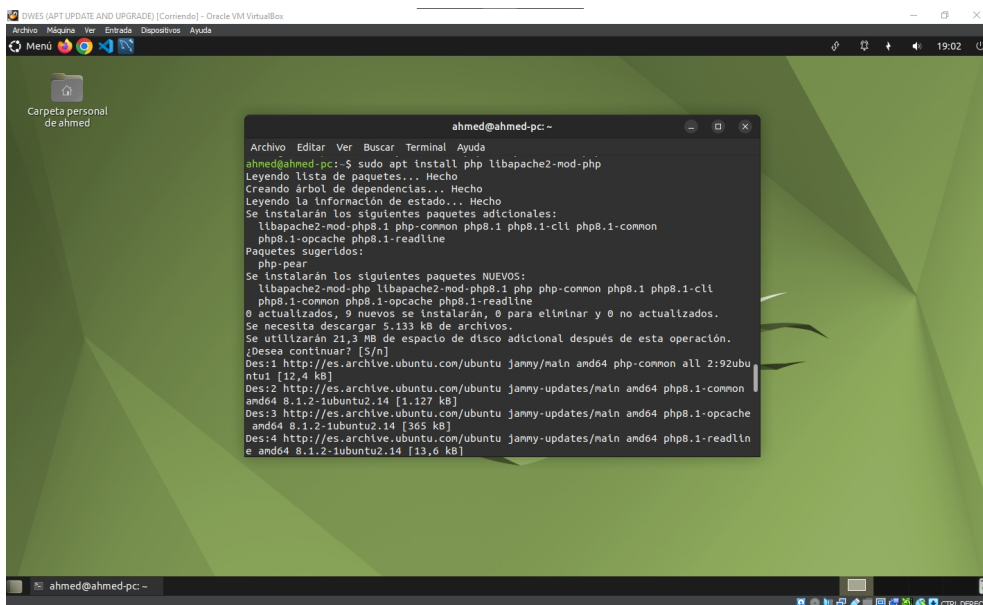
→ sudo apt install apache2



```
ahmed@ahmed-pc:~$ sudo apt install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
apache2-bin apache2-data apache2-utils libapr1 libaprutil1
libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.3-0
Paquetes sugeridos:
apache2-doc apache2-suexec-pristine | apache2-suexec-custom
Se instalarán los siguientes paquetes NUEVOS:
apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.3-0
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 2.059 kB de archivos.
Se utilizarán 8.228 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [Y/n]
Des:1 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libapr1 amd64
1.7.0-8ubuntu0.22.04.1 [108 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1 a
md64 1.6.1-5ubuntu4.22.04.2 [92,8 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1-d
bd-sqlite3 amd64 1.6.1-5ubuntu4.22.04.2 [11,3 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1-l
dap amd64 1.6.1-5ubuntu4.22.04.2 [9,170 B]
```

- Interprete de PHP.

→ sudo apt install php libapache2-mod-php

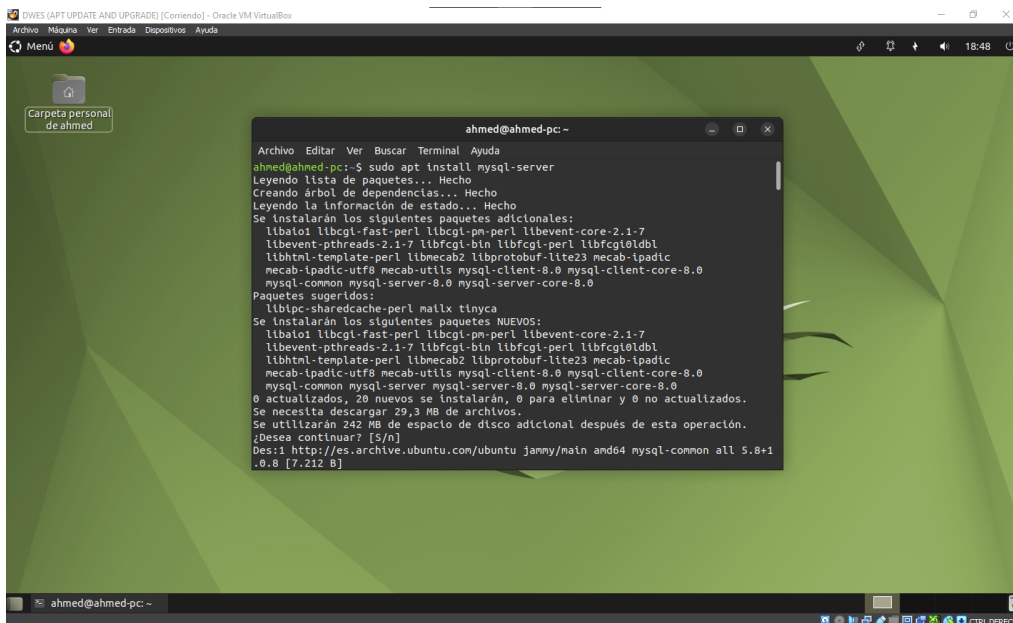


```
ahmed@ahmed-pc:~$ sudo apt install php libapache2-mod-php
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
libapache2-mod-php8.1 php-common php8.1 php8.1-cli php8.1-common
php8.1-opcache php8.1-readline
Paquetes sugeridos:
php-pear
Se instalarán los siguientes paquetes NUEVOS:
libapache2-mod-php libapache2-mod-php8.1 php php-common php8.1 php8.1-cli
php8.1-common php8.1-opcache php8.1-readline
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 5.133 kB de archivos.
Se utilizarán 21,3 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [Y/n]
Des:1 http://es.archive.ubuntu.com/ubuntu jammy/main amd64 php-common all 2:92ubu
ntu1 [12,4 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-commo
amd64 8.1.2-1ubuntu2.14 [1,127 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-opcace
amd64 8.1.2-1ubuntu2.14 [365 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-readlin
e amd64 8.1.2-1ubuntu2.14 [13,6 kB]
```

Opcionales:

- MySQL.

→ sudo apt install mysql-server

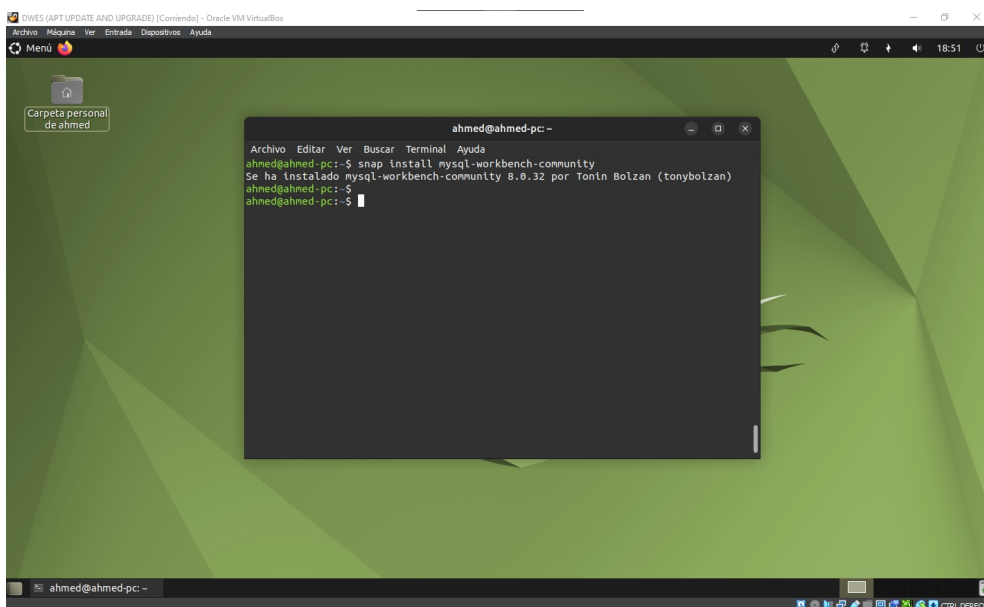


```
ahmed@ahmed-pc:~$ sudo apt install mysql-server
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
libbrotli1 libbrotli-fast-perl libbrotli-bin libbrotli-perl libbrotli0 libbrotli-dev
libevent-pthreads-2.1-7 libevent-core-2.1-7 libevent-dev libevent-perl libevent-ruby
libhtml-template-perl libmecab2 libmecab-dev libmecab-perl libmecab-ruby
mecab-ipadic-utf8 mecab-utf8 mecab-utf8-dev mecab-utf8-perl mecab-utf8-ruby
mysql-common mysql-server mysql-server-core-8.0 mysql-client-core-8.0
Paquetes sugeridos:
libbrotli-dev libbrotli-fast-perl libbrotli-bin libbrotli-perl libbrotli0 libbrotli-dev
Se instalarán los siguientes paquetes NUEVOS:
libbrotli1 libbrotli-fast-perl libbrotli-bin libbrotli-perl libbrotli0 libbrotli-dev
libevent-pthreads-2.1-7 libevent-core-2.1-7 libevent-dev libevent-perl libevent-ruby
libhtml-template-perl libmecab2 libmecab-dev libmecab-perl libmecab-ruby
mecab-ipadic-utf8 mecab-utf8 mecab-utf8-dev mecab-utf8-perl mecab-utf8-ruby
mysql-common mysql-server mysql-server-core-8.0 mysql-client-core-8.0
0 actualizados, 20 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 29,3 MB de archivos.
Se utilizarán 242 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
Des:1 http://es.archive.ubuntu.com/ubuntu jammy/main amd64 mysql-common all 5.8+1
.0.0 [7.212 B]
```

- PhpMyAdmin.

→ sudo apt install phpmyadmin(si se tiene el repositorio configurado)

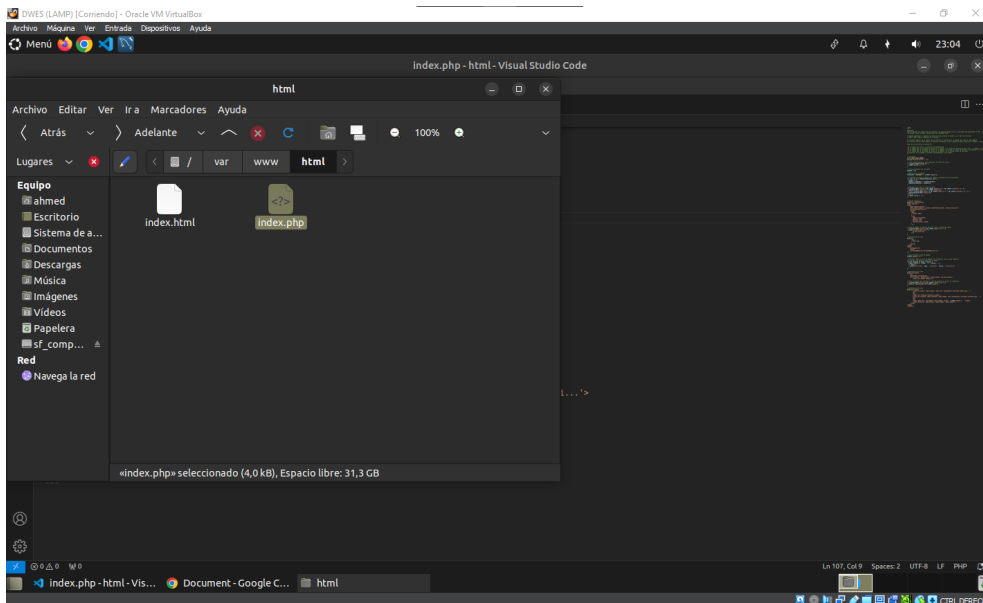
en este caso instalare el workbench



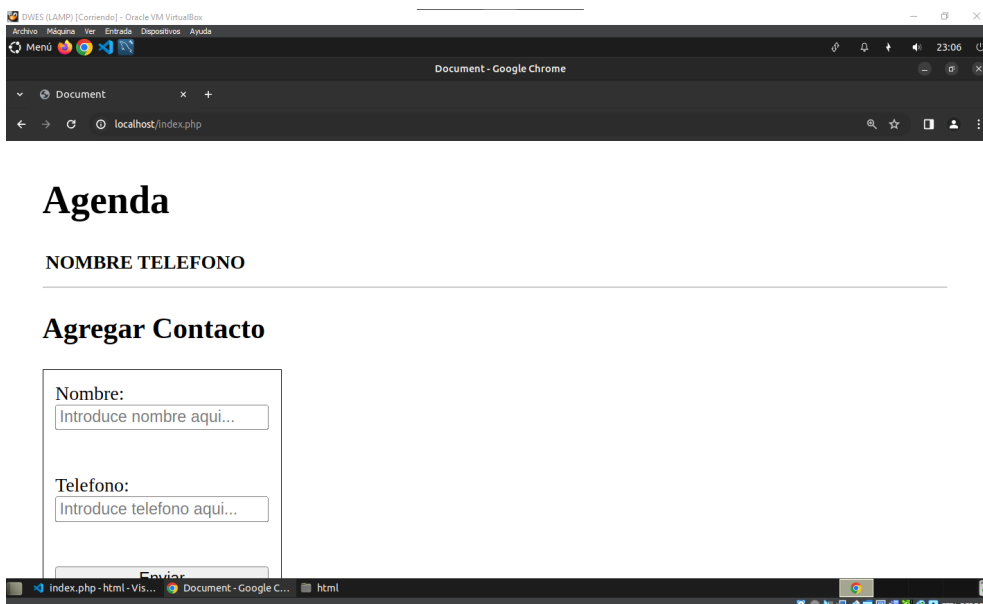
```
ahmed@ahmed-pc:~$ snap install mysql-workbench-community
Se ha instalado mysql-workbench-community 8.0.32 por Tonin Bolzan (tonbolzan)
ahmed@ahmed-pc:~$
```

Para empezar se tiene que habilitar el modulo de php para que apache2 pueda servir las paginas que en mi caso lo hice nada mas instalar php.

Una vez hecho lo anterior o ya tenerlo montado, necesitamos mover los archivos a un fichero en el servidor Linux, donde el apache2 pueda servirlo, que en este caso usare la configuración por defecto de apache.



tras todo lo anterior solo tenemos que comprobar que nuestra pagina este en correcto funcionamiento, esto se hace entrando en un navegador web y visitando nuestra pagina mediante la dirección ip de nuestro servidor o nombre de dominio. Para este caso como el entorno es grafico lo podre usar la dirección local “localhost” para comprobar la pagina y luego poner el nombre o la ruta de archivo necesaria.



y ya tendríamos nuestra aplicación de agendas lista y desplegada.