

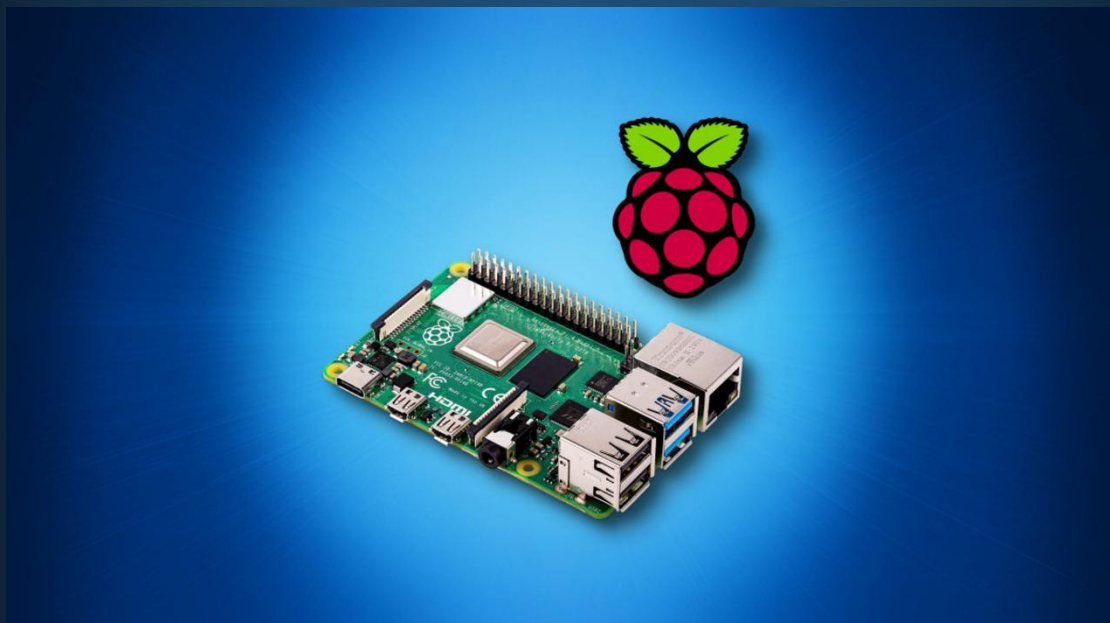
Proyecto Raspberry Pi 4

Andrés Córdoba Gutiérrez

2º Desarrollo de Aplicaciones Web

Proyecto Inspirado en el siguiente:

<https://soloelectronicos.com/tag/prevision-del-tiempo-con-raspberry/>



INDICE

1- Introducción	3
2- Requisitos	4
3- Análisis y diseño:	5
a) Guía de estilos: Colores y tipografía	5
b) Logotipos	5
c) MockUps	5
d) Actores	5
e) Base de Datos	6
4- Implementación	7
a) Implementación de ssh	7
b) Instalación de Programas	10
c) Configuración phpMyAdmin	15
d) Arbol de archivos	17
e) Xplanet	17
f) Programación	19
5- Pasos del Proyecto	18
6- Conclusiones	21
7- Enlaces y despliegue	22
8- Dificultades del Proyecto	23

1- Introducción

La idea de este proyecto fue propuesta por el departamento, consistiendo en la creación de una estación gráfica, para ello se usara una Raspberry Pi4 cedida por el centro, en la que se realizaran la creación de ficheros que darán lugar a una aplicación que podrá ilustrar diferentes opciones:

- Temperatura en tiempo real
- Fases Lunares
- Uso de la red Wifi
- Etc...

Siendo el resultado esperado algo similar a lo mostrado en la siguiente imagen, obtenida de la web de la que se nutre todo este proyecto:



2- Requisitos

- Repositorio de GitHub con commits coherentes y medianamente constantes.
- Utilización de un Framework como mínimo.
- Diseño amigable y responsive.
- Utilización de etiquetas semánticas, alt en todas las imágenes y otras herramientas para mejorar el SEO...
- Diseño de una base de datos con varias relaciones entre sus entidades (tablas).
- Habilitación de mínimo dos actores en la plataforma con diferentes roles y permisos.
- Utilización de conexiones asíncronas con AJAX u otras tecnologías para mejorar la experiencia de usuario.
- Actualizaciones cada 2 semanas.
- Defensa y presentación del proyecto en Clase.

3- Análisis y diseño:

a) Guía de estilos: Colores y tipografía

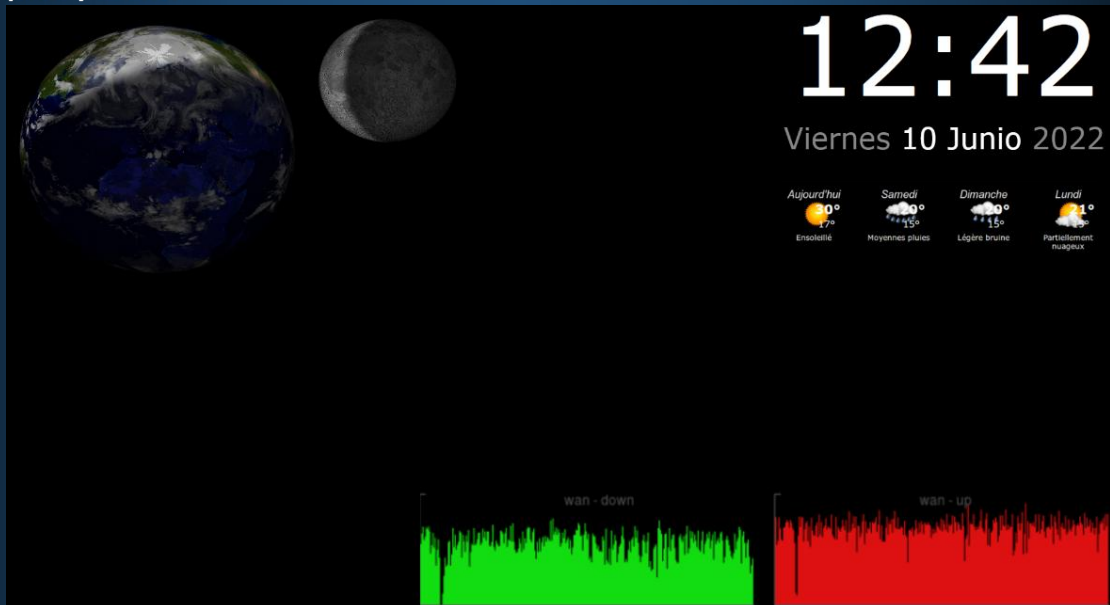
En este caso utilizaremos la guía que nos ofrece el blog del que se obtiene toda la información, con la posibilidad de modificaciones futuras con el cambio del archivo de estilos.

b) Logotipos

El logotipo a usar será el de la propia raspberry ya que es el principal protagonista de todo el proyecto

c) MockUps

Con este diseño se puede ver donde irán implementados cada uno de los módulos del proyecto

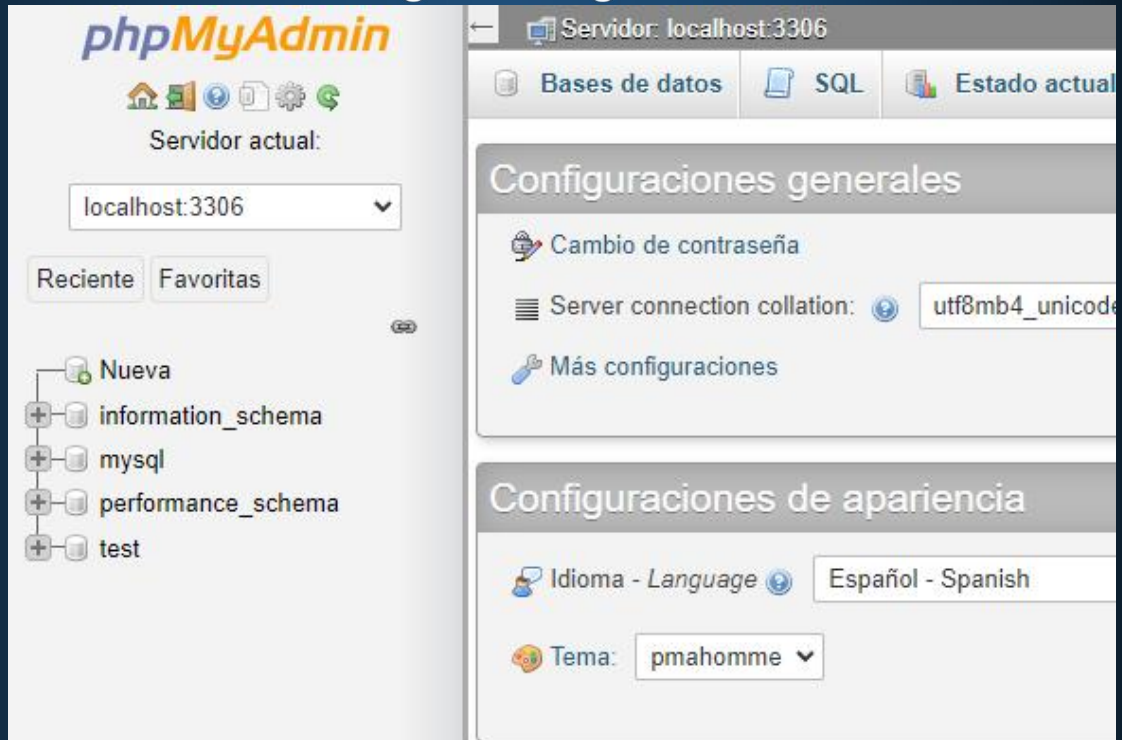


d) Actores

No se identifica ningún actor de por si, ya que no tiene ningún tipo de interacción con la aplicación, y su función es mostrar los datos en tiempo real.

e) Base de Datos

En este proyecto la gran mayoría de los datos son obtenidos en tiempo real ya sea por la hora y fecha del sistema o descarga de imágenes.



4- Implementación

a) Implementación de ssh

Lo primero a la hora de trabajar con RaspBerry son una serie de configuraciones:

-Usar ssh para trabajar desde nuestra maquina principal, usaremos PuTTY para conectarnos, para ello tendremos que hacer unas configuraciones previas en la RaspBerry

1º Configuraciones de la RapBerry

Con el comando - sudo raspi-config

```
Raspberry Pi Software Configuration Tool (raspi-config)

1 System Options      Configure system settings
2 Display Options     Configure display settings
3 Interface Options   Configure connections to peripherals
4 Performance Options Configure performance settings
5 Localisation Options Configure language and regional settings
6 Advanced Options    Configure advanced settings
8 Update              Update this tool to the latest version
9 About raspi-config  Information about this configuration tool

<Select>                                <Finish>
```

Y elegimos “Expand Filesystem”

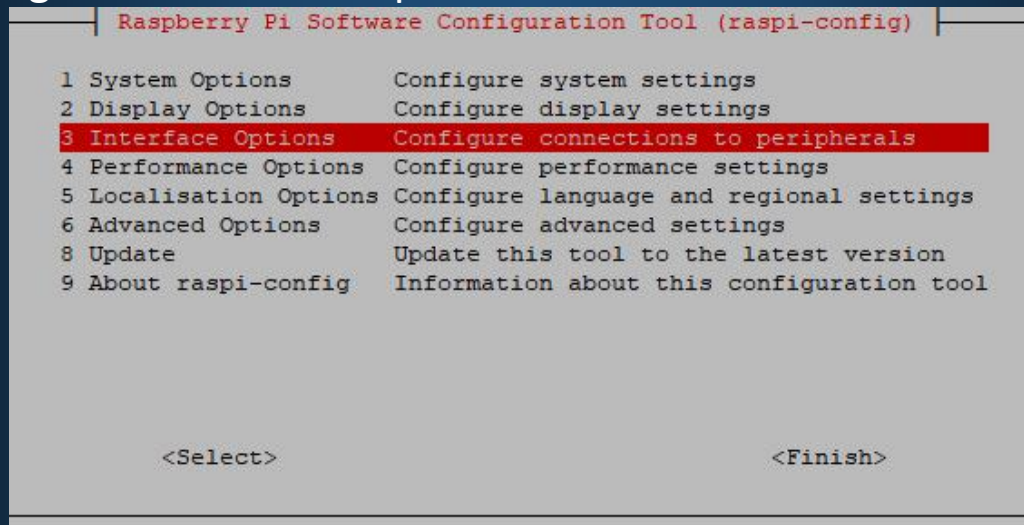
```
Raspberry Pi Software Configuration Tool (raspi-config)

A1 Expand Filesystem  Ensures that all of the SD card is available
A3 Compositor         Enable/disable xcompmgr composition manager
A4 Network Interface Names Enable/disable predictable network i/f names
A5 Network Proxy Settings Configure network proxy settings
A6 Boot Order         Choose network or USB device boot
A7 Bootloader Version Select latest or default boot ROM software
A9 Wayland            Enable experimental Wayland backend

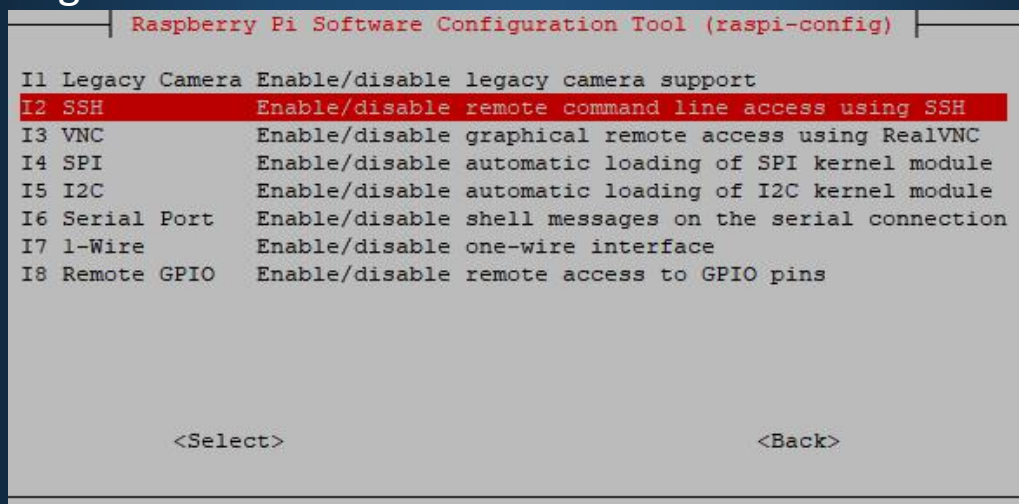
<Select>                                <Back>
```

Una vez realizado estos pasos nos pedirá reiniciar

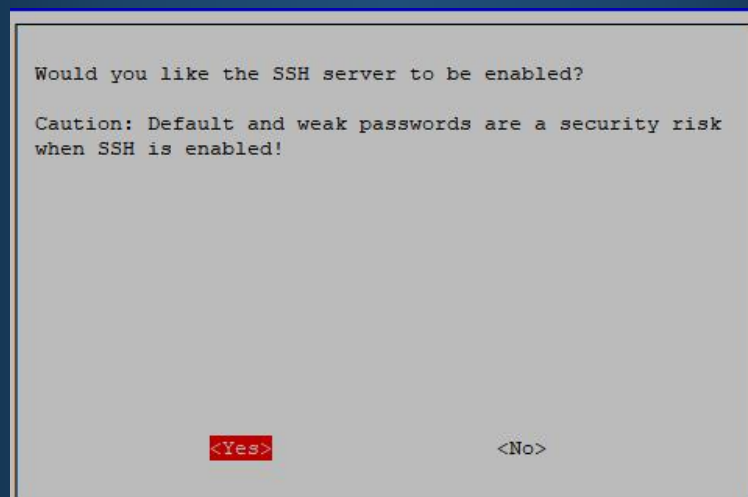
Una vez más usaremos el comando - sudo raspi-config
Elegimos “Interface Options”



Y elegiremos “SSH”



Nos preguntara si estamos seguros de activarlo elegimos
“Yes”



Ahora cambiaremos a un puerto más específico para trabajar con el, con el comando:

- sudo nano /etc/ssh/sshd_config

Cambiaremos el puerto por defecto quitando el “#” y añadiendo el puerto deseado

```
Port 3022
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::
```

Una vez guardado los cambios usaremos el comando

- sudo service ssh restart

Además cambiaremos la ip a una fija para mayor comodidad, con el comando:

- sudo nano /etc/dhcpd.conf

Cambiaremos un párrafo para que surtan los cambios
En este caso los dejaremos como muestra la imagen

```
# Example static IP configuration:
interface eth0
static ip_address=192.168.1.11/24
#static ip6_address=fd51:42f8:caae:d92e::ff/64
static routers=192.168.1.1
static domain_name_servers=192.168.1.1 8.8.8.8 fd51:42f8:caae:d92e::1
```

b) Instalación de Programas

Vamos a instalar LAMP que son las siglas de:

Linux ← es la propia Raspberry Pi con Raspbian

Apache ← es el servidor de páginas web

MySQL ← es la base de datos

PHP ← es el lenguaje de programación web desde el lado del servidor

Apache:

Usaremos los siguientes comandos:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

```
sudo apt-get install apache2 -y
```

El modificador -y sirve para evitar estar pulsando Yes en todas las preguntas que nos hagan.

Ir a <http://192.168.1.11> para ver que se ha instalado bien.

Obtendremos un resultado como el siguiente:



debian

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.Load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default

Para cambiar la página principal inicial:

- Vemos que se llama index.html
- Y está en el directorio /var/www/html

Para ir a este directorio usamos el comando:

- cd /var/www/html

Y con el comando:

- ls -l

Nos aparecerá lo siguiente:

```
pl@raspberrypi:/var/www/html $ ls -l
total 12
-rw-r--r-- 1 root root 10701 May 16 14:33 index.html
pl@raspberrypi:/var/www/html $
```

Si no te deja editar el archivo usa:

- sudo chown -R p1: index.html

Si queremos tener permisos para tocar más directorios usaremos:

- sudo chown -R p1:www-data /var/www
- sudo chmod u+rxw,g+rx-w,o-rwx /var/www

PHP:

Para instalarlo usaremos los siguientes comandos:

- `sudo apt-get install php libapache2-mod-php -y`

Para comprobar la versión usaremos:

- `php -v`

```
pi@raspberrypi:/var/www/html $ php -v
PHP 7.4.28 (cli) (built: Feb 17 2022 16:17:19) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.28, Copyright (c), by Zend Technologies
```

Cambiamos la extensión del archivo de .html a .php

- `mv index.html index.php`

MySQL:

Usaremos los siguientes comandos:

- `sudo apt-get install mariadb-server`
- `sudo mysql_secure_installation`
- `sudo service apache2 restart`

En la batería de preguntas que nos hacen hemos seleccionado lo siguiente:

 p1@raspberrypi: ~

p1@raspberrypi:~ \$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] n
... skipping.

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] n
... skipping.

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!

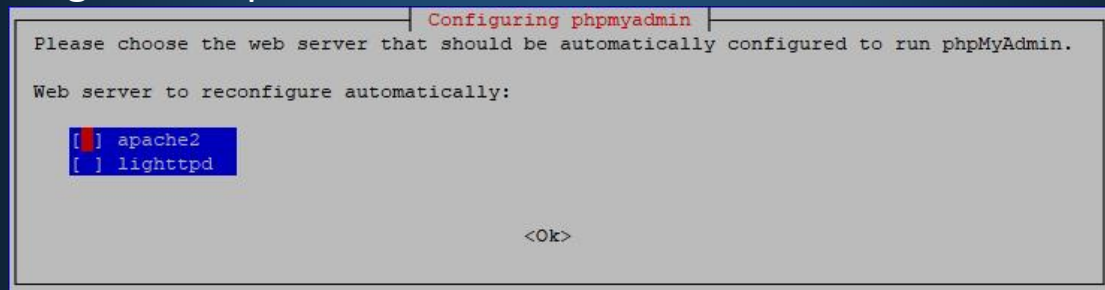
p1@raspberrypi:~ \$ █

phpMyAdmin:

Usaremos los siguientes comandos:

- `sudo apt install phpmyadmin`

Elegimos Apache2 como servidor:



Después introduciremos la siguiente línea:

- `sudo systemctl restart apache2.service`

Y a la siguiente ventana con “Espacio” aceptaremos:

c) Configuración phpMyAdmin

Editamos el fichero de configuración:

- `sudo nano /etc/apache2/apache2.conf`

Dentro del archivo añadimos al final la línea:


- `Include /etc/phpmyadmin/apache.conf`

Reiniciamos:

- `sudo /etc/init.d/apache2 restart`

Tendremos este resultado:

⚠ No es seguro | 192.168.1.11/phpmyadmin/



Bienvenido a phpMyAdmin

Idioma - Language

Español - Spanish ▼

Iniciar sesión ?

Usuario:

Contraseña:

Continuar

Credenciales:

Usuario: root

Password: 1234

phpMyAdmin

Servidor actual: localhost:3306

Reciente Favoritas

- Nueva
- information_schema
- mysql
- performance_schema
- test

Bases de datos SQL Estado actual Cuentas de usuarios Exportar Importar Configuración Replicación Variables Más

Configuraciones generales

- Cambio de contraseña
- Server connection collation: utf8mb4_unicode_ci
- Más configuraciones

Configuraciones de apariencia

- Idioma - Language: Español - Spanish
- Tema: pmahomme

Servidor de base de datos

- Servidor: Localhost via UNIX socket
- Tipo de servidor: MariaDB
- Conexión del servidor: No se está utilizando SSL
- Versión del servidor: 10.5.15-MariaDB-0+deb11u1 - Debian 11
- Versión del protocolo: 10
- Usuario: root@localhost
- Conjunto de caracteres del servidor: UTF-8 Unicode (utf8mb4)

Servidor web

- Apache/2.4.53 (Debian)
- Versión del cliente de base de datos: libmysql - mysqlnd 7.4.28

d) Arbol de archivos

index.php: Página principal que es llamada por el navegador

ajax.php: Contiene el código que se ejecuta cuando se hacen peticiones AJAX

.inc.php: Contiene las funciones de PHP

javascript.js: Contiene las funciones de JavaScript
hoja de estilo style.css

e) Xplanet

Instalaremos xplanet con el siguiente comando:

- apt-get install xplanet

f) Programación

Conectandonos por ssh con Putty en un principio, después se uso WinSCP para poder abrir los archivos con Visual Code, un entorno conocido y utilizado en el ciclo, facilitando la modificación de los archivos.

5- Pasos del Proyecto

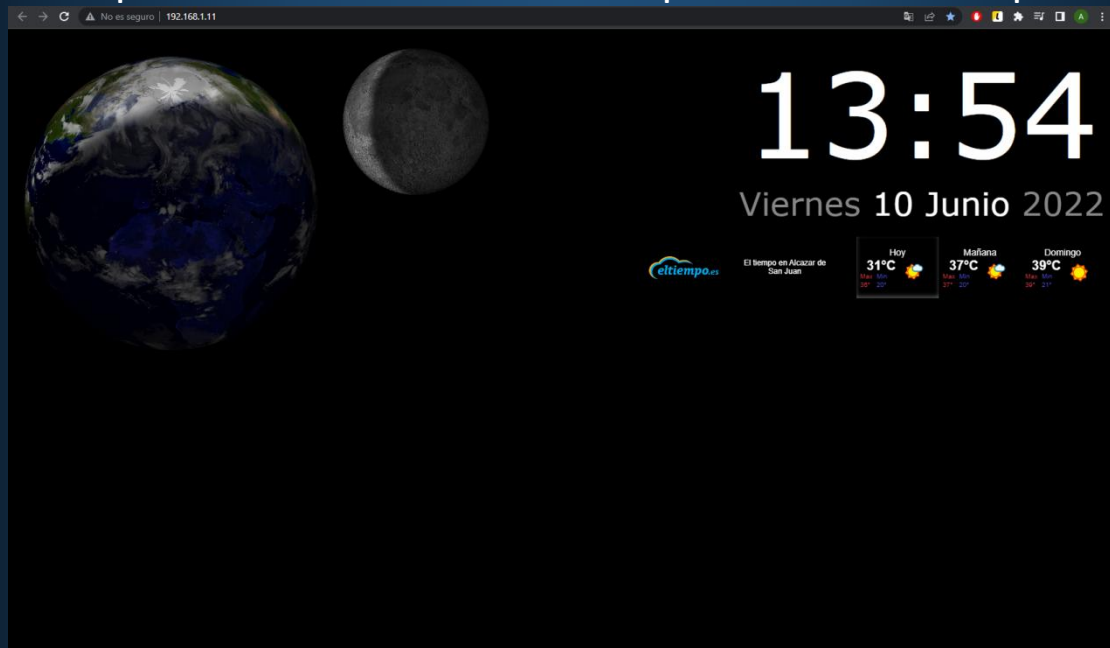
Implementamos el modulo de fase lunar y terrestre



Implementamos el modulo de la hora



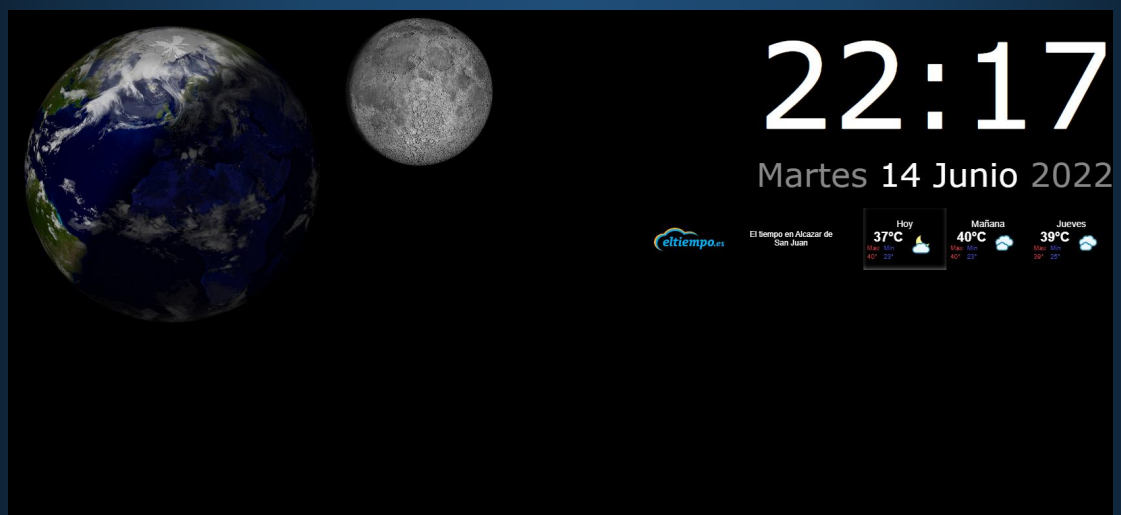
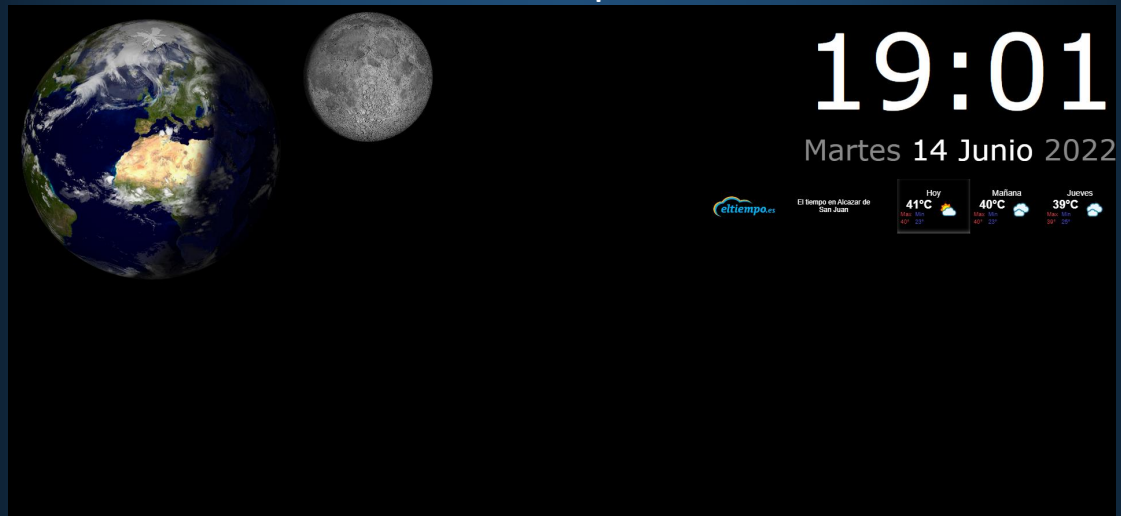
Implementamos el modulo de previsión del tiempo



Implantación de las coordenadas correctas de España, para centrar la vista sobre el país y su estado con respecto al sol



Se ve como cambia la zona iluminada por el sol y la fase de la luna con la ejecución del comando sh Xplanet.sh
Se muestran diferentes horas para ver avance del sol



6- Conclusiones

Proyecto con módulos muy variados en dificultad como en implantación, da como resultado una interfaz bastante limpia mostrando los datos deseados.

Se han solucionado bastantes problemas, pero con los últimos módulos no ha sido posible acabarlos a tiempo.

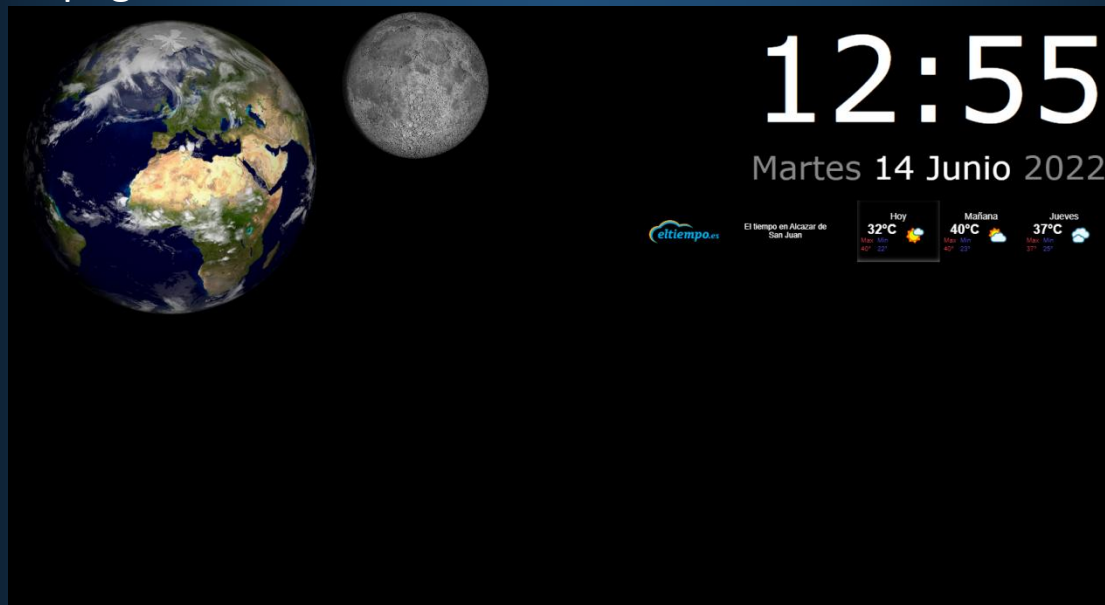
El proyecto es totalmente modificable pudiendo implementar nuevos módulos o los que han quedado pendientes.

7- Enlaces y despliegue

La aplicación se despliega en cuanto se enciende la Raspberry pi 4, lanza el servidor y podemos verlo en la ip asignada 192.168.1.11 si queremos conectarnos remotamente para modificar los archivos los haremos desde el puerto 3022.

En esta página veremos actualizarse en tiempo real toda la información.

La página final seria esta:



8- Dificultades del Proyecto

En las primeras semanas del proyecto a resultado complicado la instalación del sistema operativo en la Raspberry Pi 4, dando más errores de los esperados debido a que la versión del sistema que explicaba la guía no coincidía (debido a las actualizaciones).

En la reunión del 25 de abril se pedirá ayuda al tutor para solucionarlo y poder seguir con el proyecto.

Después de la tutoria llegamos a la conclusión de que el fallo lo estaba ocasionando mi monitor, se pudo solucionar usando otro monitor con un adaptador.

Problemas con base de datos corrupta, solucionado gracias a la ayuda del tutor durante la tutoria.

Problemas al implementar los módulos de:

- Estado de servidores NAS
- TeamSpeak clientes conectados
- Clientes VPN conectados
- Ancho de Banda

Motivos:

- El router del que se dispone dificulta o hace imposible la utilización de puertos
- Dificultad para entender ciertos pasos del blog