

DESPLIEGUE DE APLICACIONES WEB



UT2

Tarea 2 - Manual
Técnico



MIGERIATREX

Autor: Alberto M. Sánchez Macías
DAW 2 - B

Índice

Índice.....	2
1. Descripción del proyecto.....	3
2. Objetivos del proyecto.....	3
3. Descripción de las fases desarrolladas.....	3
3.1 Iniciación del proyecto.....	3
3.2 Planificación del proyecto.....	4
3.3 Arquitectura de la aplicación.....	4
3.3.1 FRONTEND.....	4
3.3.2 BACKEND.....	4
3.4. Proceso de despliegue.....	4
Descripción del proceso de despliegue:.....	5
3.4.1 - Update de la máquina virtual.....	5
3.4.2 - Upgrade de la máquina virtual(Parte 1).....	5
3.4.3 - Upgrade de la máquina virtual(Parte 2).....	6
3.4.4 - Instalación Apache2.....	6
3.4.5 - Accedemos al servidor de Apache.....	7
3.4.6 - Creación de la carpeta del proyecto.....	7
3.4.7 - Creación del archivo index del proyecto.....	8
3.4.8 - Generamos un certificado SSL.....	9
3.4.9 - Creación del archivo “conf” de nuestro proyecto.....	10
3.4.10 - Modificación del archivo hosts.....	12
3.4.11 - Activación de nuestro sitio web.....	13
3.4.12 - Accedemos a nuestro sitio web.....	13
3.4.13 - Creación de los directorios en los que se va a dividir nuestro proyecto.....	14
3.4.14 - Creación de los usuarios de nuestra web.....	15
3.4.15 - Asignación de permisos de grupo a cada fichero.....	21
3.4.16 - Comprobación de autenticación de usuarios.....	22
3.4.17 - Habilitar nuestro dominio web para que se pueda ver desde Windows.....	26
4. Cronograma.....	31

1. Descripción del proyecto

La empresa miGeriatrex es una residencia geriátrica que ha confiado en nosotros para el desarrollo de su página web con el objetivo de tener una mejor comunicación para las familias con los residentes y un mayor alcance tanto a nivel regional como nacional para sus nuevos clientes. También nos han comunicado que necesitan un espacio para el personal de la residencia que se divida en tres zonas, una para los doctores, otra para los administrativos y una común para todo el personal de la residencia.

2. Objetivos del proyecto

El objetivo que hemos marcado para nuestro proyecto es desarrollar una página web moderna y fácilmente accesible, que proporcione información sobre los servicios que se ofrecen, mejoren la visibilidad de la residencia en internet, que las familias puedan comunicarse fácilmente con los residentes y tanto los doctores como los administrativos dispongan de un espacio de trabajo en línea donde poder consultar y registrar tanto informes de los pacientes en caso de los doctores, como las cuentas y documentación administrativa en el caso de los administrativos. Este espacio en línea tiene como objetivo facilitar la comunicación entre el personal de la residencia y para ello nos han comunicado que necesitan disponer de un espacio común.

3. Descripción de las fases desarrolladas

3.1 Iniciación del proyecto

El objetivo principal de éste proyecto es desarrollar una página web moderna y accesible, que proporcione información sobre los servicios que se ofrecen y mejorar la visibilidad de la residencia en línea.

Hemos tenido una reunión con el dueño de la residencia José Manuel Pérez Martínez y nos ha transmitido los detalles que necesita la página web:

- Que los familiares puedan comunicarse fácilmente con los residentes.
- Que se muestre todos los servicios que ofrecen en la residencia.
- Un espacio en línea para el desarrollo del trabajo del personal para llevar un mejor control de la información, que los datos relevantes estén centralizados y se puedan acceder a ellos de manera segura.

3.2 Planificación del proyecto

El tiempo que se ha determinado para el desarrollo de la aplicación web es de 5 meses.

3.3 Arquitectura de la aplicación

3.3.1 FRONTEND

Tecnologías usadas:

- HTML.
- CSS.
- JavaScript.

Entorno de desarrollo:

- **Editor de código:** Visual Studio Code Versión 1.94.2
- **Control de versiones:** GitHub/GitHub Desktop.

3.3.2 BACKEND

Tecnologías usadas:

- **Lenguajes de programación:** Java.
- **Bases de datos:** MySQL.

Entorno de desarrollo:

- **IDE:** NetBeans Versión 19 con JDK 11.
- **Editor BBDD:** MySQL WorkBench Versión 8.0.
- **Servidor:** Apache HTTP Server Versión 2.4.58.
- **Máquina Virtual:** Ubuntu con ISO → ubuntu-24.04.1-desktop-amd64.
- **Control de versiones:** GitHub/GitHub Desktop.

3.4. Proceso de despliegue

El proceso de despliegue se ha realizado en una máquina virtual Ubuntu Versión 24.04.1.

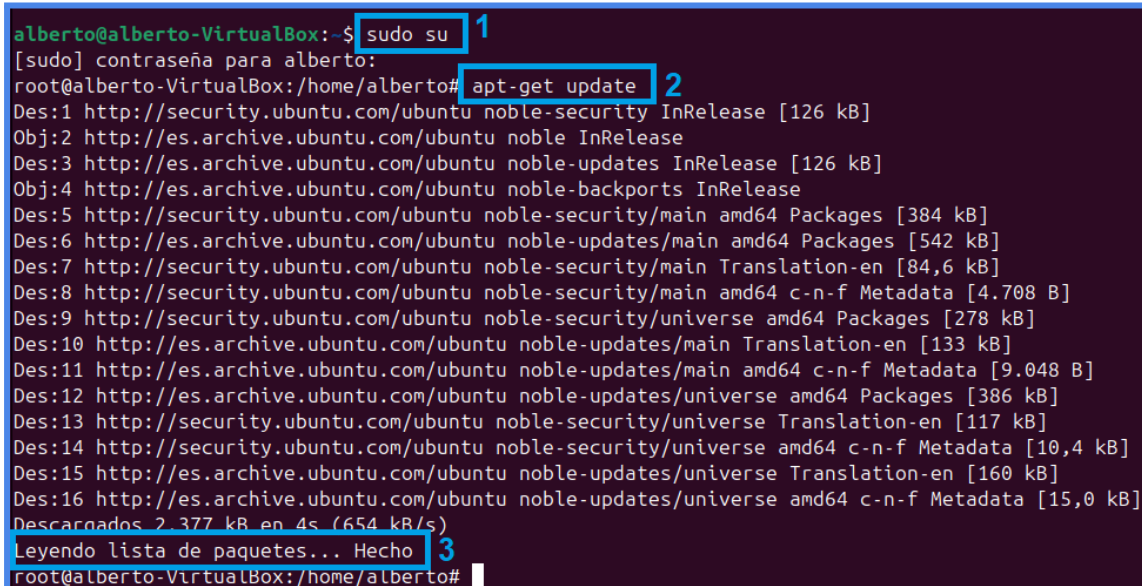
Una vez levantada nuestra máquina Ubuntu procedemos a instalar el servidor de apache, para ello primero tenemos que hacer un **update** para actualizar el software de la máquina y un **upgrade** para asegurarnos de tener la última versión del software.

***Nota:** Para **TODOS** los comandos que se van a realizar se necesita tener los permisos de root. Para ello utilizaremos el comando **sudo su** e ingresamos la contraseña del administrador.

***Nota2:** Todos los **comandos** estarán señalados de **color verde** en el documento.

Descripción del proceso de despliegue:

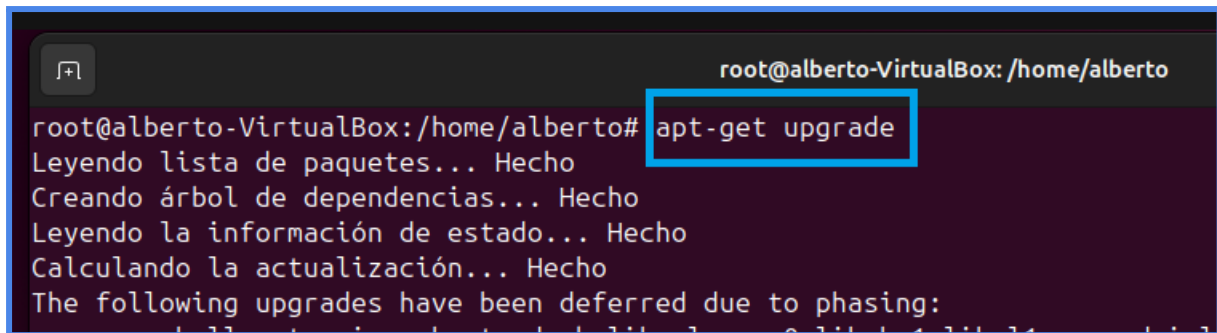
3.4.1 - Update de la máquina virtual.



```
alberto@alberto-VirtualBox:~$ sudo su 1
[sudo] contraseña para alberto:
root@alberto-VirtualBox:/home/alberto# apt-get update 2
Des:1 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Obj:2 http://es.archive.ubuntu.com/ubuntu noble InRelease
Des:3 http://es.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Obj:4 http://es.archive.ubuntu.com/ubuntu noble-backports InRelease
Des:5 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [384 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [542 kB]
Des:7 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [84,6 kB]
Des:8 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [4.708 B]
Des:9 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [278 kB]
Des:10 http://es.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [133 kB]
Des:11 http://es.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [9.048 B]
Des:12 http://es.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [386 kB]
Des:13 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [117 kB]
Des:14 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [10,4 kB]
Des:15 http://es.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [160 kB]
Des:16 http://es.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [15,0 kB]
Descargados 2.377 kB en 4s (654 kB/s)
Leyendo lista de paquetes... Hecho 3
root@alberto-VirtualBox:/home/alberto#
```

Explicación de la imagen: Primero nos otorgamos los privilegios del super usuario con **sudo su**⁽¹⁾, luego ingresamos el comando **apt-get update**⁽²⁾ y la terminal nos mostrará un mensaje de que se ha realizado con éxito el update⁽³⁾.

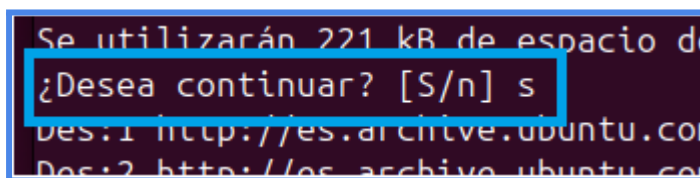
3.4.2 - Upgrade de la máquina virtual(Parte 1).



```
root@alberto-VirtualBox: /home/alberto
root@alberto-VirtualBox:/home/alberto# apt-get upgrade
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Calculando la actualización... Hecho
The following upgrades have been deferred due to phasing:
```

Explicación foto: Realizamos el upgrade con el comando **apt-get upgrade**.

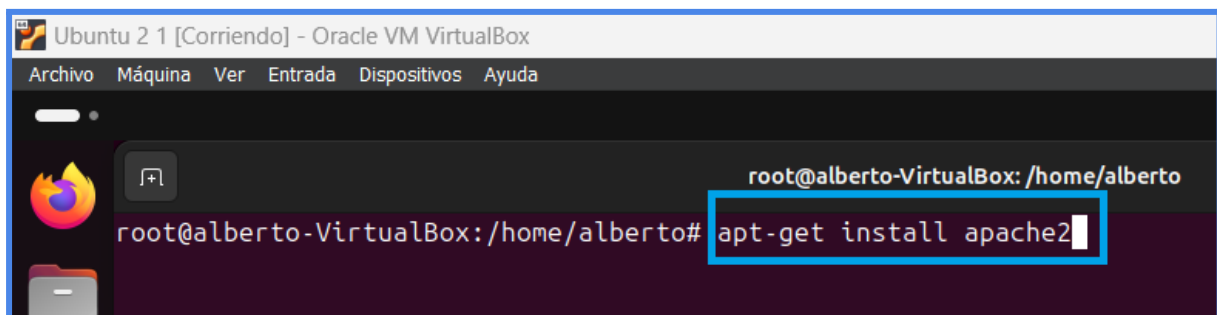
3.4.3 - Upgrade de la máquina virtual(Parte 2).



```
Se utilizarán 221 kB de espacio d
¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.co
Des:2 http://es.archive.ubuntu.co
```

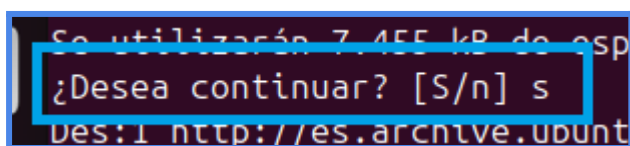
Explicación foto: A la hora de realizar el upgrade nos preguntarán si deseamos continuar y le decimos que sí.

3.4.4 - Instalación Apache2.



```
Ubuntu 2 1 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
root@alberto-VirtualBox: /home/alberto
root@alberto-VirtualBox:/home/alberto# apt-get install apache2
```

Explicación foto: Para instalar el servidor de apache2 en nuestra máquina virtual lo haremos a través del comando **apt-get install apache2**.



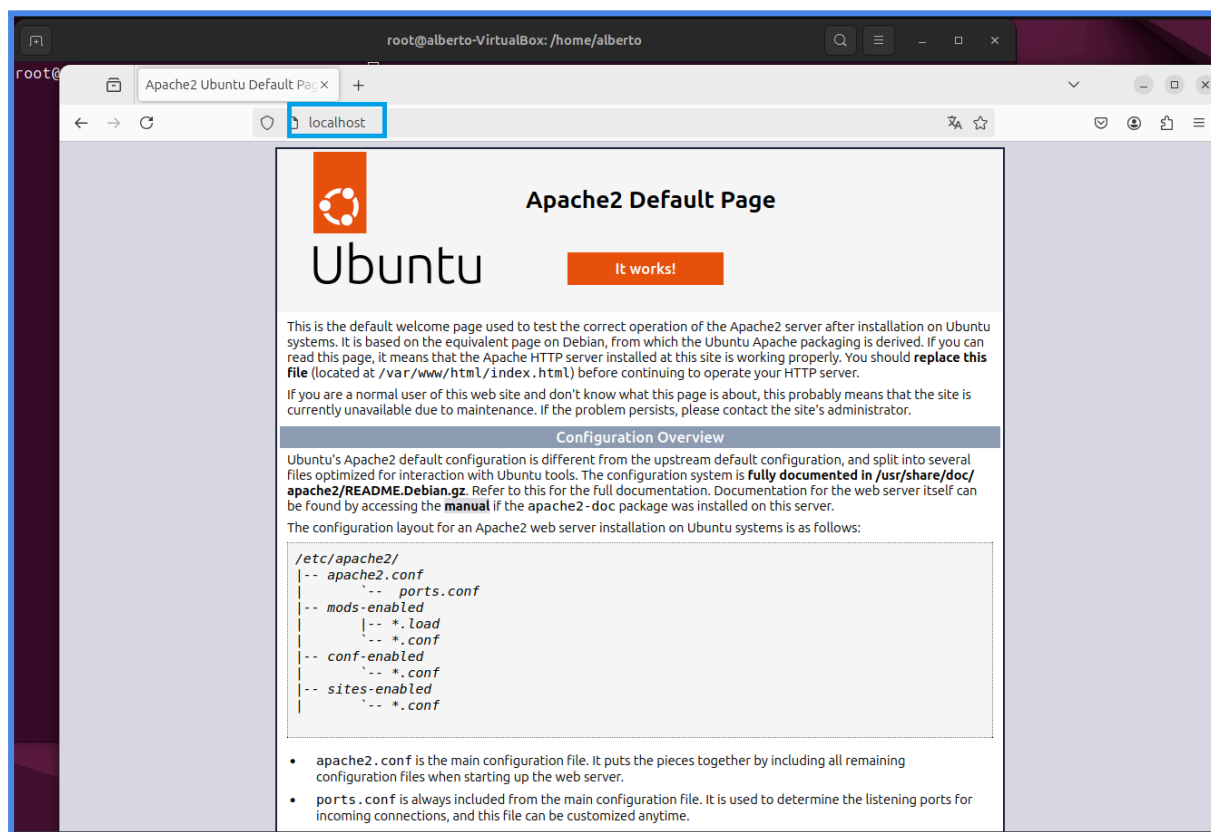
```
Se utilizarán 7.455 kB de esp
¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubunt
```

Explicación foto: Confirmamos la instalación.

```
root@alberto-VirtualBox:/home/alberto# apache2 -v
Server version: Apache/2.4.58 (Ubuntu)
Server built: 2024-07-17T18:55:23
```

Explicación foto: Con el comando **apache2 -v** podemos saber la versión de apache que se ha instalado. En este caso se ha instalado la **versión 2.4.58**.

3.4.5 - Accedemos al servidor de Apache.



Explicación foto: Una vez instalado el servidor de apache nos iremos al navegador y comprobaremos si está levantado. Para ello podemos buscarlo de dos formas: escribiendo **localhost** o **127.0.0.1** en el buscador. Si el servidor está levantado nos debe aparecer lo que se muestra en la foto.

3.4.6 - Creación de la carpeta del proyecto.

```
root@alberto-VirtualBox: /var/www
root@alberto-VirtualBox:/home/alberto# cd /var/www 1
root@alberto-VirtualBox:/var/www# ls 2
html
root@alberto-VirtualBox:/var/www# mkdir migeriatrex 3
root@alberto-VirtualBox:/var/www# ls 4
html migeriatrex 5
root@alberto-VirtualBox:/var/www#
```

Explicación foto: Ahora debemos crear la carpeta de nuestro proyecto. Para ello nos iremos al directorio www a usando el comando `cd /var/www`⁽¹⁾, hacemos un `ls`⁽²⁾ para ver su contenido y observamos que sólo está la carpeta html. Procedemos a crear el directorio del proyecto en éste caso se llamará **migeriatrex**, para ello usamos el comando `mkdir migeriatrex`⁽³⁾. Hacemos otro `ls`⁽⁴⁾ para ver si se ha creado correctamente y vemos que está creado⁽⁵⁾.

3.4.7 - Creación del archivo index del proyecto.

```
root@alberto-VirtualBox: /var/www/migeriatrex
root@alberto-VirtualBox:/var/www# cd migeriatrex/ 1
root@alberto-VirtualBox:/var/www/migeriatrex# nano index.html 2
```

Explicación foto: Una vez creado el directorio del proyecto accedemos a él mediante comando `cd migeriatrex`⁽¹⁾. Una vez dentro, hacemos un `nano index.html`⁽²⁾ que a la vez que lo estamos creando también lo podemos modificar.

```
GNU nano 7.2 index.html *
Esta es la pagina web de migeriatrex
```

Explicación foto: Una vez dentro de **index.html** escribimos un mensaje en texto plano que luego visualizaremos cuando activemos nuestro sitio web.

3.4.8 - Generamos un certificado SSL.

```
root@alberto-VirtualBox: /etc/apache2/sites-enabled# a2enmod ssl
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
    systemctl restart apache2
root@alberto-VirtualBox: /etc/apache2/sites-enabled# service apache2 reload
root@alberto-VirtualBox: /etc/apache2/sites-enabled#
```

Explicación foto: Antes de generar un certificado ssl primero debemos habilitar el módulo SSL. Lo haremos a través del comando `a2enmod ssl`⁽¹⁾. Una vez habilitado debemos hacer un reload del servidor. Para ello usaremos el comando `service apache2 reload`⁽²⁾.

```
000-default.conf default-ssl.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# openssl req -x509 -nodes -days 365 -newkey rsa:2048 -k
eyout /etc/ssl/private/migeriatrex.key -out /etc/ssl/certs/migeriatrex.crt
```

Explicación foto: Una vez activado el módulo generamos el certificado con el comando `openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/migeriatrex.key -out /etc/certs/migeriatrex.crt(1)`.

[illegible]

Explicación foto: Le damos a la tecla **intro** en todos los apartados y se nos generará el certificado SSL.

```
root@alberto-VirtualBox:/etc/apache2/sites-available# apache2ctl -t
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set
the 'ServerName' directive globally to suppress this message
Syntax OK
root@alberto-VirtualBox:/etc/apache2/sites-available# service apache2 reload
Warning: The unit file, source configuration file or drop-ins of apache2.service changed on disk. Run 'system
ctl daemon-reload' to reload units.
root@alberto-VirtualBox:/etc/apache2/sites-available# systemctl daemon-reload
root@alberto-VirtualBox:/etc/apache2/sites-available#
```

Explicación foto: Una vez generado el certificado SSL haremos uso del comando **apachectl -t**⁽¹⁾ que sirve para comprobar si hay errores de sintaxis en caso de que nos diga **“Syntax OK”**⁽²⁾ procedemos a hacer un reload del servidor con el comando **service apache2 reload**⁽³⁾. En caso de que nos salga un Warning de daemon-reload usamos el comando **systemctl daemon-reload**⁽⁴⁾.

3.4.9 - Creación del archivo “conf” de nuestro proyecto.

```
Máquina Ver Entrada Dispositivos Ayuda
14 de oct 20:42
root@alberto-VirtualBox: /etc/apache2/sites-available
root@alberto-VirtualBox:/etc/apache2/sites-available# ls
000-default.conf default-ssl.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# cp default-ssl.conf migeriatrex.conf
```

Explicación foto: Para crear el archivo conf de nuestro proyecto haremos una copia del archivo **default-ssl**. Lo haremos a través del comando **cp default-ssl.conf migeriatrex.conf**⁽¹⁾.

```
root@alberto-VirtualBox: /etc/apache2/sites-available
root@alberto-VirtualBox:/etc/apache2/sites-available# ls
000-default.conf default-ssl.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# cp default-ssl.conf migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# ls
000-default.conf default-ssl.conf migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available#
```

Explicación foto: Hacemos un **ls**⁽¹⁾ para comprobar que se haya creado correctamente⁽²⁾.

```
root@alberto-VirtualBox: /etc/apache2/sites-available
root@alberto-VirtualBox:/etc/apache2/sites-available# ls
000-default.conf  default-ssl.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# cp default-ssl.conf migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# ls
000-default.conf  default-ssl.conf  migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# nano migeriatrex.conf
```

Explicación foto: Procedemos a modificar el archivo conf → **nano** **migeriatrex.conf**.

```
GNU nano 7.2 migeriatrex.conf *
<VirtualHost *:443>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www/migeriatrex

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn
```

Explicación foto: Dentro del archivo **migeriatrex.conf**⁽¹⁾ modificaremos el apartado de **DocumentRoot** → **/var/www/migeriatrex**⁽²⁾.

```
# A self-signed (snakeoil) certificate can be created by installing
# the ssl-cert package. See
# /usr/share/doc/apache2/README.Debian.gz for more info.
# If both key and certificate are stored in the same file, only the
# SSLCertificateFile directive is needed.
SSLCertificateFile /etc/ssl/certs/migeriatrex.crt
SSLCertificateKeyFile /etc/ssl/private/migeriatrex.key
```

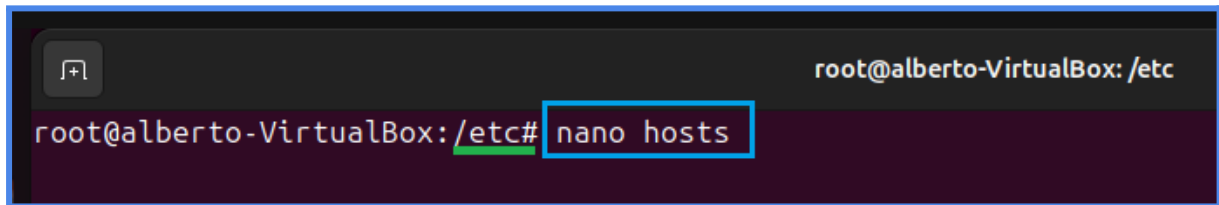
Explicación foto: Un poco más abajo del archivo modificaremos las dos líneas siguientes⁽²⁾.

Línea 1 → **/etc/ssl/certs/migeriatrex.crt**

Línea 2 → **/etc/ssl/private/migeriatrex.key**

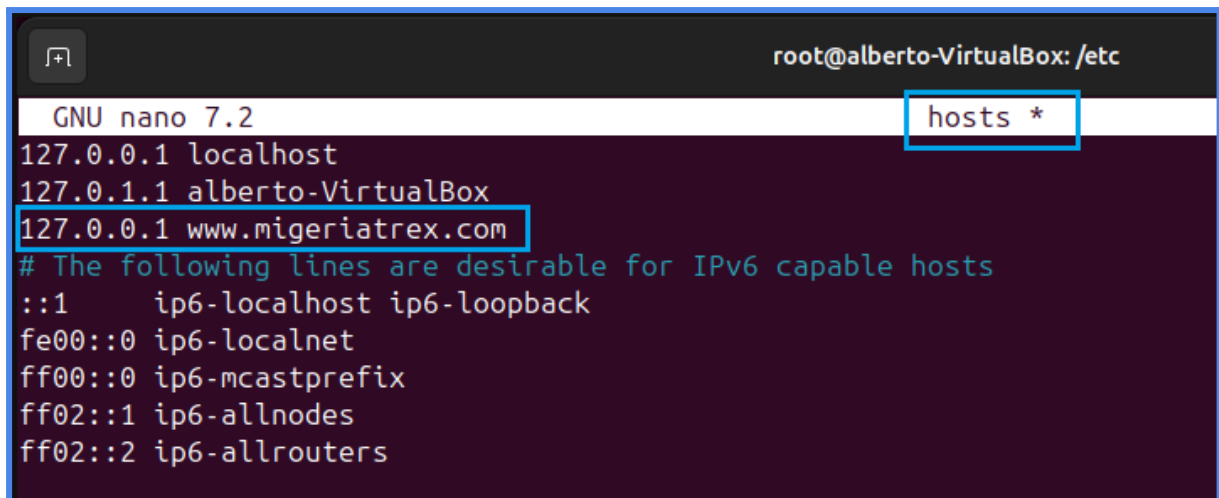
***Nota:** Una vez modificado, guardamos el proyecto y realizaremos un **apache2ctl -t** para asegurarnos de que no haya errores de sintaxis.

3.4.10 - Modificación del archivo hosts.



```
root@alberto-VirtualBox: /etc
root@alberto-VirtualBox: /etc# nano hosts
```

Explicación foto: Modificaremos el archivo con nombre **hosts** que se encuentra en el fichero **etc** → **cd /etc**. Para modificarlo haremos un **nano hosts**⁽¹⁾.



```
GNU nano 7.2
127.0.0.1 localhost
127.0.1.1 alberto-VirtualBox
127.0.0.1 www.migeriatrex.com
# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0  ip6-localnet
ff00::0  ip6-mcastprefix
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
```

Explicación foto: Dentro del archivo **hosts**⁽¹⁾ asociaremos el dominio de nuestro proyecto a la ip de localhost → “**127.0.0.1 www.migeriatrex.com**”⁽²⁾.

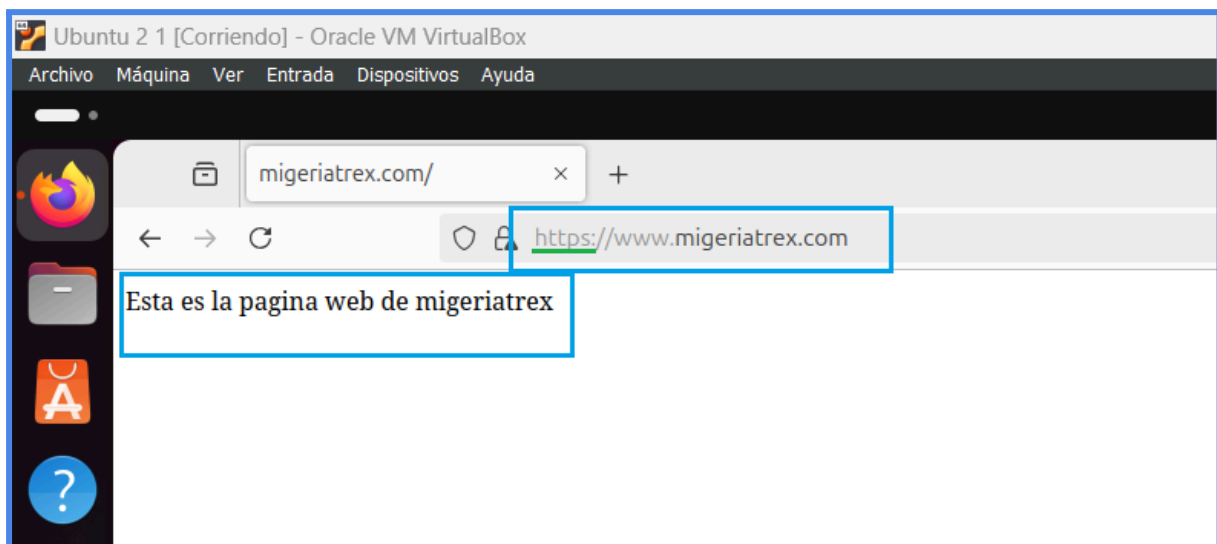
***Nota:** Una vez modificado, guardamos el proyecto y realizaremos un **apache2ctl -t** para asegurarnos de que no haya errores de sintaxis.

3.4.11 - Activación de nuestro sitio web.

```
root@alberto-VirtualBox: /etc/apache2/sites-available
root@alberto-VirtualBox:/etc# cd apache2 1
root@alberto-VirtualBox:/etc/apache2# cd sites-available/ 2
root@alberto-VirtualBox:/etc/apache2/sites-available# ls
000-default.conf default-ssl.conf migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# a2ensite migeriatrex.conf 3
Enabling site migeriatrex.
To activate the new configuration, you need to run:
    systemctl reload apache2
root@alberto-VirtualBox:/etc/apache2/sites-available# service apache2 reload 4
```

Explicación foto: Para la activación de nuestro sitio web debemos habilitar nuestro archivo conf. Para ello nos iremos al directorio sites-available → **cd /etc/apache2/sites-available**^(1y2). Para habilitarlo haremos uso del comando **a2ensite migeriatrex.conf**⁽³⁾. Una vez habilitado haremos un reload → **service apache2 reload**⁽⁴⁾.

3.4.12 - Accedemos a nuestro sitio web.



Explicación foto: Para acceder a nuestro sitio web iremos a nuestro navegador y escribiremos “**https://www.migeriatrex.com**” y si todo se ha configurado de manera correcta se nos mostrará el mensaje que escribimos antes en el index de nuestra web.

3.4.13 - Creación de los directorios en los que se va a dividir nuestro proyecto.

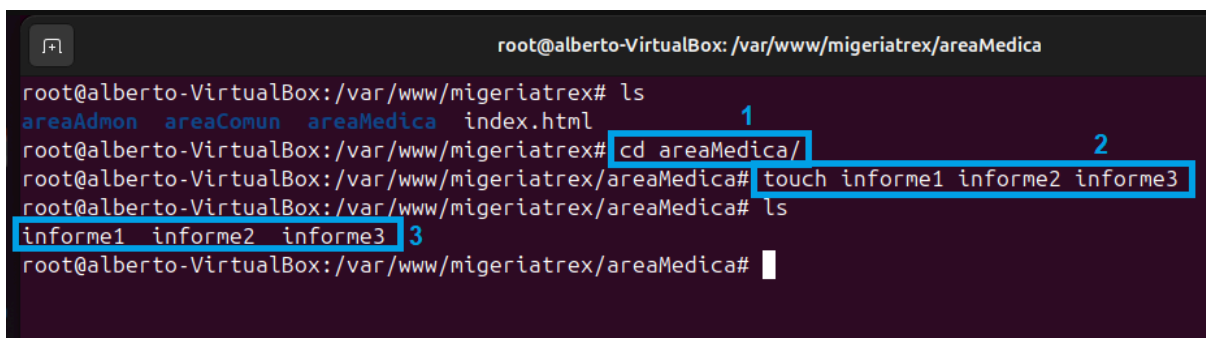
Paso 1 - Creamos los directorios.



```
root@alberto-VirtualBox: /var/www/migeriatrex
root@alberto-VirtualBox:/home/alberto# cd /var/www 1
root@alberto-VirtualBox:/var/www# ls
html migeriatrex
root@alberto-VirtualBox:/var/www# cd migeriatrex/ 2
root@alberto-VirtualBox:/var/www/migeriatrex# ls
index.html
root@alberto-VirtualBox:/var/www/migeriatrex# mkdir areaMedica 3
root@alberto-VirtualBox:/var/www/migeriatrex# ls
areaMedica index.html
root@alberto-VirtualBox:/var/www/migeriatrex# mkdir areaAdmon 4
root@alberto-VirtualBox:/var/www/migeriatrex# ls
areaAdmon areaMedica index.html
root@alberto-VirtualBox:/var/www/migeriatrex# mkdir areaComun 5
root@alberto-VirtualBox:/var/www/migeriatrex# ls
areaAdmon areaComun areaMedica index.html
root@alberto-VirtualBox:/var/www/migeriatrex#
```

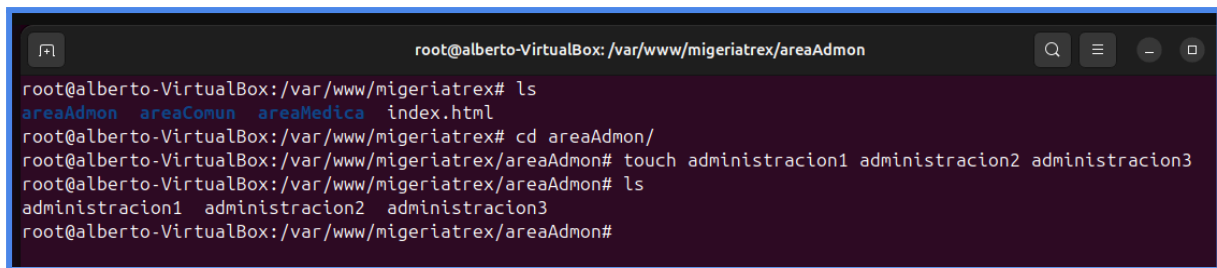
Explicación foto: Una vez que tengamos nuestro sitio web operativo procederemos a crear los directorios dónde vamos a tener la información de cada usuario. En éste caso vamos a hacer 3 directorios → **areaMedica**⁽³⁾, **areaAdmon**⁽⁴⁾ y **areaComun**⁽⁵⁾.

Paso 2 - Creamos los ficheros correspondientes en cada directorio.



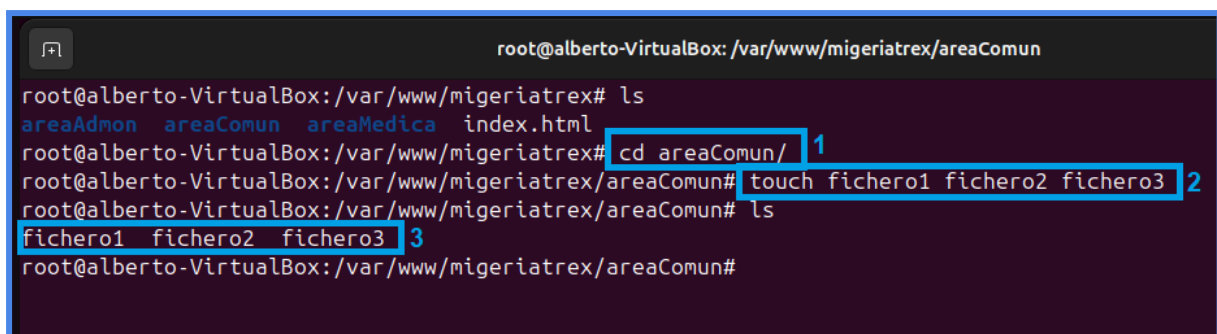
```
root@alberto-VirtualBox: /var/www/migeriatrex/areaMedica
root@alberto-VirtualBox:/var/www/migeriatrex# ls
areaAdmon areaComun areaMedica index.html 1
root@alberto-VirtualBox:/var/www/migeriatrex# cd areaMedica/ 2
root@alberto-VirtualBox:/var/www/migeriatrex/areaMedica# touch informe1 informe2 informe3
root@alberto-VirtualBox:/var/www/migeriatrex/areaMedica# ls
informe1 informe2 informe3 3
root@alberto-VirtualBox:/var/www/migeriatrex/areaMedica#
```

Explicación foto: Una vez dentro del directorio **areaMedica** creamos los ficheros a través del comando **touch informe1 informe2 informe3**. Luego hacemos un **ls** para comprobar que se han creado correctamente.



```
root@alberto-VirtualBox: /var/www/migeriatrex/areaAdmon
root@alberto-VirtualBox:/var/www/migeriatrex# ls
areaAdmon areaComun areaMedica index.html
root@alberto-VirtualBox:/var/www/migeriatrex# cd areaAdmon/
root@alberto-VirtualBox:/var/www/migeriatrex/areaAdmon# touch administracion1 administracion2 administracion3
root@alberto-VirtualBox:/var/www/migeriatrex/areaAdmon# ls
administracion1 administracion2 administracion3
root@alberto-VirtualBox:/var/www/migeriatrex/areaAdmon#
```

Explicación foto: Una vez dentro del directorio **areaAdmon** creamos los ficheros a través del comando **touch administracion1 administracion2 administracion3**. Luego hacemos un **ls** para comprobar que se han creado correctamente.

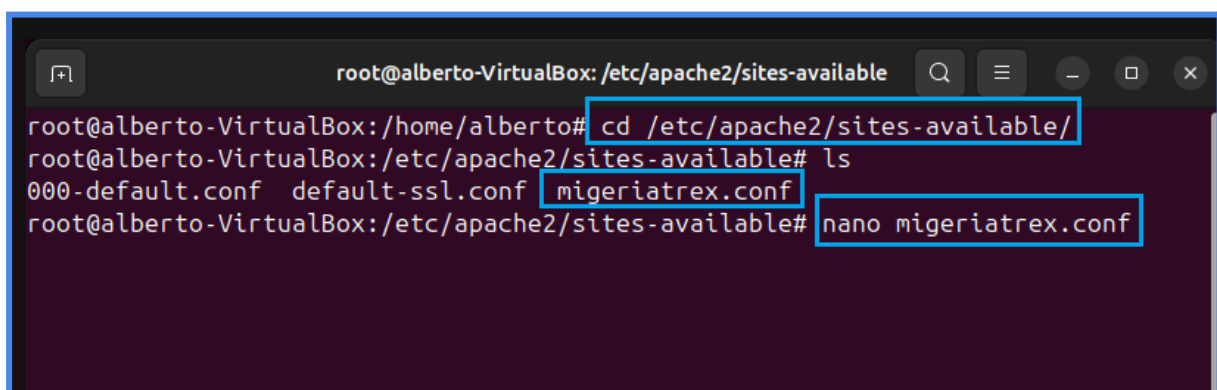


```
root@alberto-VirtualBox: /var/www/migeriatrex/areaComun
root@alberto-VirtualBox:/var/www/migeriatrex# ls
areaAdmon areaComun areaMedica index.html
root@alberto-VirtualBox:/var/www/migeriatrex# cd areaComun/ 1
root@alberto-VirtualBox:/var/www/migeriatrex/areaComun# touch fichero1 fichero2 fichero3 2
root@alberto-VirtualBox:/var/www/migeriatrex/areaComun# ls
fichero1 fichero2 fichero3 3
root@alberto-VirtualBox:/var/www/migeriatrex/areaComun#
```

Explicación foto: Una vez dentro del directorio **areaComun** creamos los ficheros a través del comando **touch fichero1 fichero2 fichero3**. Luego hacemos un **ls** para comprobar que se han creado correctamente.

3.4.14 - Creación de los usuarios de nuestra web.

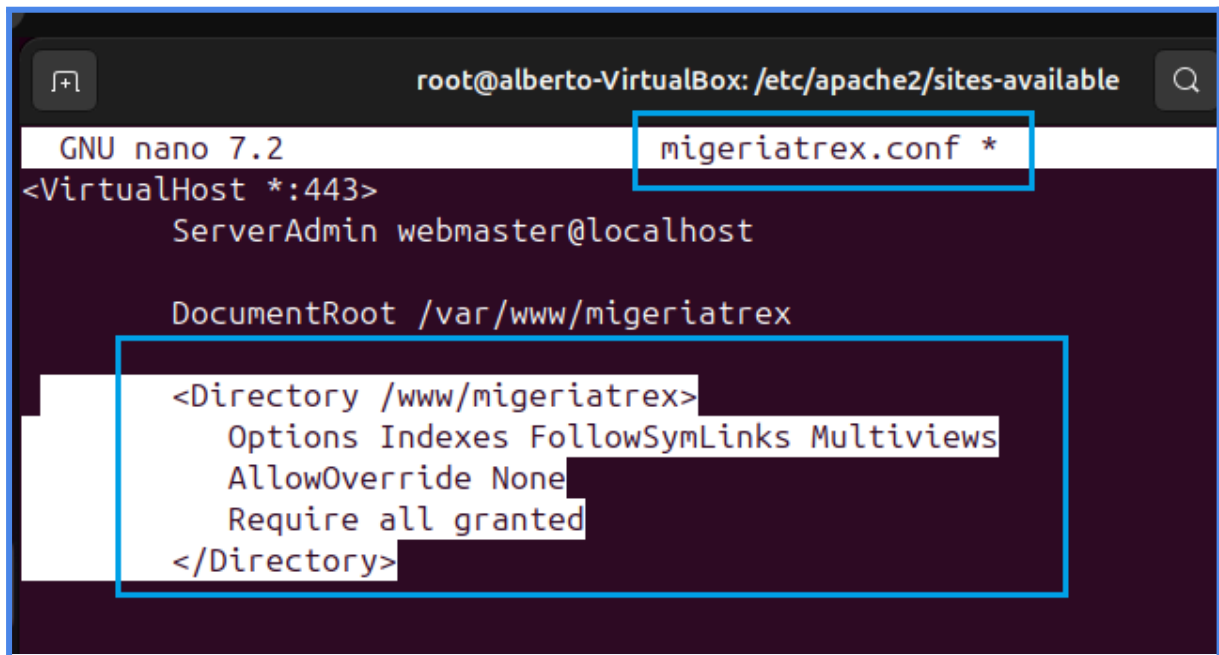
Ahora procederemos a crear los usuarios que tendrán permisos para ver los directorios **areaMedica**, **areaAdmon** y **areaComun**.



```
root@alberto-VirtualBox: /etc/apache2/sites-available
root@alberto-VirtualBox:/home/alberto# cd /etc/apache2/sites-available/
root@alberto-VirtualBox:/etc/apache2/sites-available# ls
000-default.conf default-ssl.conf migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# nano migeriatrex.conf
```

Explicación foto: Para crear usuarios primero nos dirigimos a nuestro archivo **migeriatrex.conf**.

Paso 1 - Permisos del sitio web www.migeriatrex.com



The screenshot shows a terminal window with the title bar 'root@alberto-VirtualBox: /etc/apache2/sites-available'. The nano editor is open, editing 'migeriatrex.conf *'. The configuration content is as follows:

```
GNU nano 7.2 migeriatrex.conf *
<VirtualHost *:443>
    ServerAdmin webmaster@localhost

    DocumentRoot /var/www/migeriatrex

    <Directory /www/migeriatrex>
        Options Indexes FollowSymLinks Multiviews
        AllowOverride None
        Require all granted
    </Directory>
```

Explicación foto: Una vez dentro, primero añadimos los permisos de la página principal.

***Corrección:** en la primera línea **<Directory /www/migeriatrex>** lo cambiamos por → **<Directory /var/www/migeriatrex>** para que coja la ruta completa.

Paso 2 - Permisos del resto de ficheros de nuestro sitio web.

```
root@alberto-VirtualBox: /etc/apache2/sites-available
GNU nano 7.2 migeriatrex.conf *
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/migeriatrex/areaMedica>
    Order deny,allow
    AuthUserFile "/etc/apache2/claves.txt"
    AuthName "Identificacion como Usuario-Doctor"
    AuthType Basic
    Require valid-user
</Directory>

<Directory /var/www/migeriatrex/areaAdmon>
    Order deny,allow
    AuthUserFile "/etc/apache2/claves.txt"
    AuthName "Identificacion como Usuario-Admon"
    AuthType Basic
    Require valid-user
</Directory>

<Directory /var/www/migeriatrex/areaComun>
    Order Deny,allow
    AuthUserFile "/etc/apache2/claves.txt"
    AuthName "Identificacion como Usuario"
    AuthType Basic
    Require valid-user
</Directory>

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
```

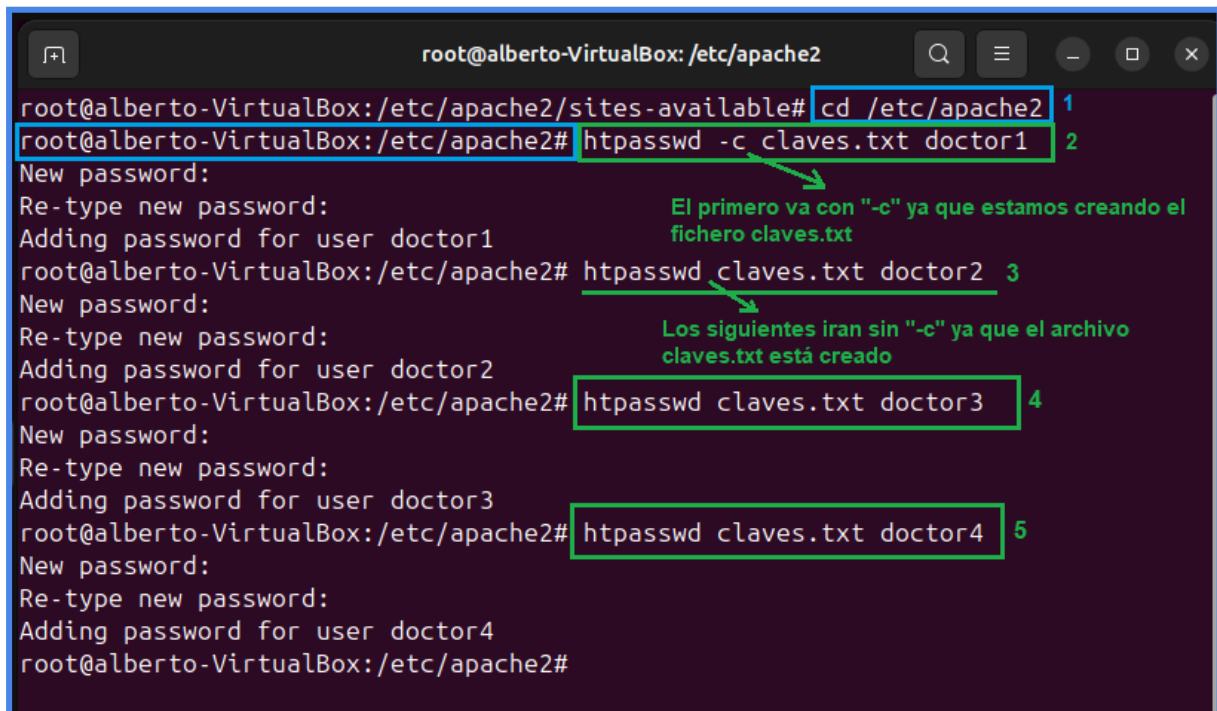
Explicación foto: Después añadimos los permisos a los 3 directorios.

Como se puede observar en la línea de **AuthUserFile** estamos haciendo referencia a un archivo llamado **claves.txt**. En ese archivo se guardarán todos los usuarios con sus contraseñas codificadas pero ese archivo se creará más adelante.

```
root@alberto-VirtualBox: /etc/apache2/sites-available# nano migeriatrex.conf
root@alberto-VirtualBox: /etc/apache2/sites-available# apachectl -t
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
root@alberto-VirtualBox: /etc/apache2/sites-available#
```

Explicación foto: Realizamos un **apachectl -t** para comprobar que la sintaxis es correcta.

Paso 3 - Creación de los usuarios.

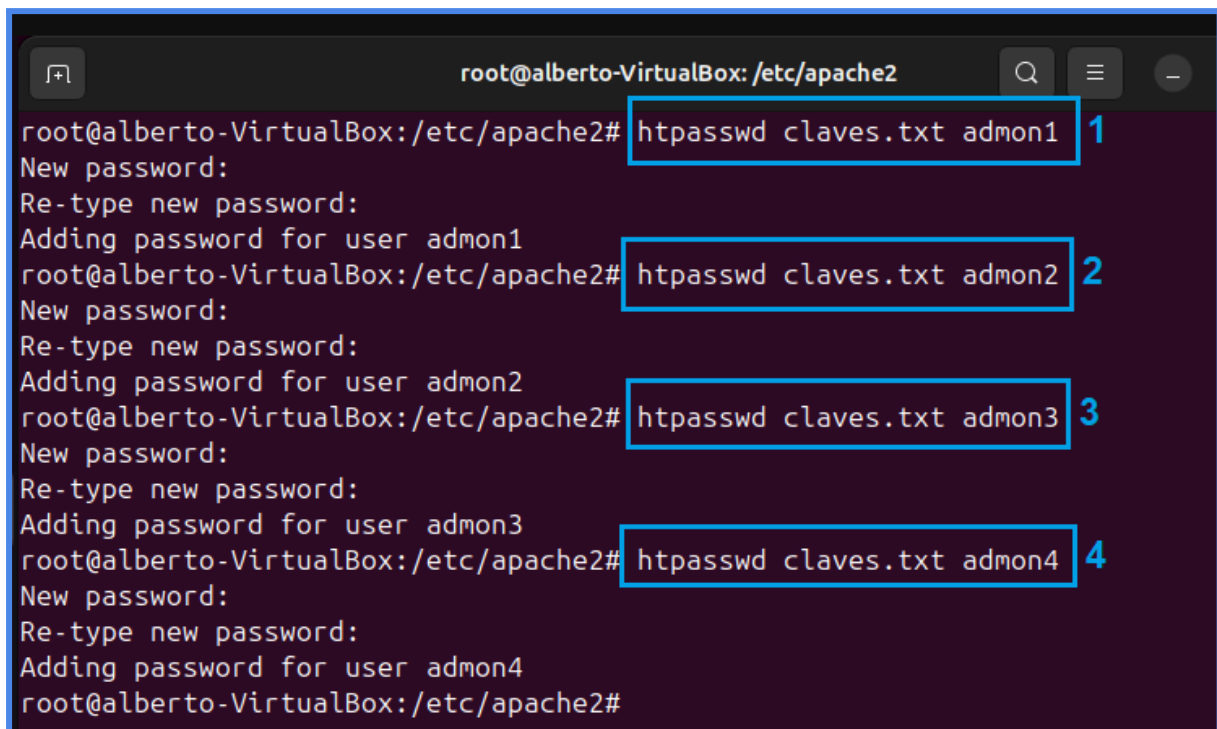


```
root@alberto-VirtualBox: /etc/apache2
root@alberto-VirtualBox:/etc/apache2/sites-available# cd /etc/apache2 1
root@alberto-VirtualBox:/etc/apache2# htpasswd -c claves.txt doctor1 2
New password:
Re-type new password:
Adding password for user doctor1
root@alberto-VirtualBox:/etc/apache2# htpasswd claves.txt doctor2 3
New password:
Re-type new password:
Adding password for user doctor2
root@alberto-VirtualBox:/etc/apache2# htpasswd claves.txt doctor3 4
New password:
Re-type new password:
Adding password for user doctor3
root@alberto-VirtualBox:/etc/apache2# htpasswd claves.txt doctor4 5
New password:
Re-type new password:
Adding password for user doctor4
root@alberto-VirtualBox:/etc/apache2#
```

El primero va con "-c" ya que estamos creando el fichero claves.txt

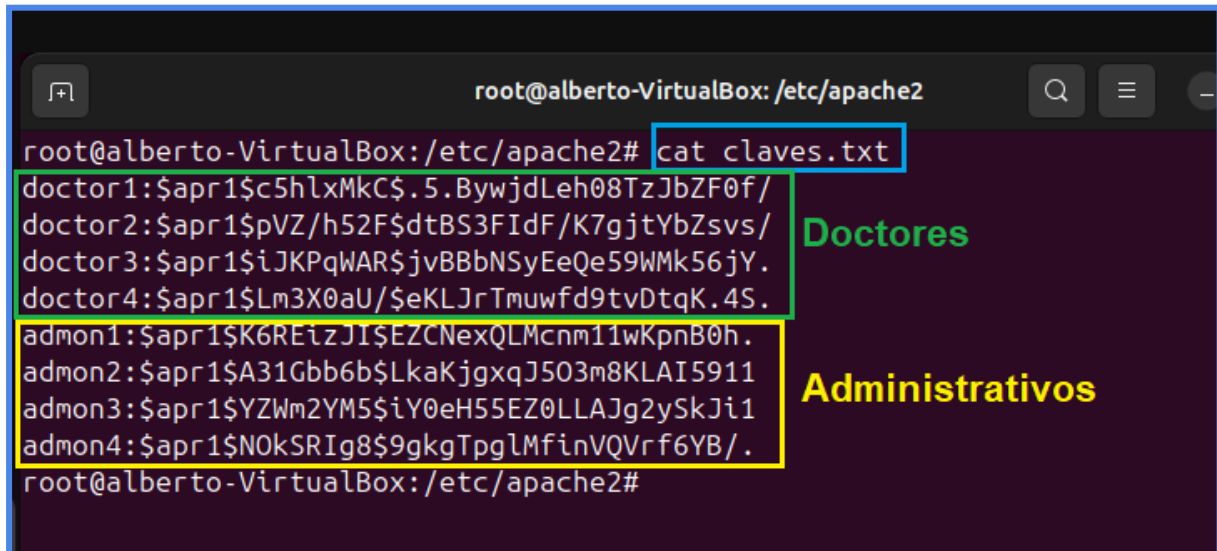
Los siguientes iran sin "-c" ya que el archivo claves.txt está creado

Explicación foto: Una vez dentro del fichero `/etc/apache2`⁽¹⁾ procedemos a crear los usuarios. Para ello usaremos el comando `htpasswd -c claves.txt doctor1`⁽²⁾ para el primer usuario. (***OJO!** - Solo en el primero ya que el `-c` es para crear `claves.txt`). Para el resto de usuarios usaremos el comando `htpasswd claves.txt doctor2`⁽³⁻⁴⁻⁵⁾.



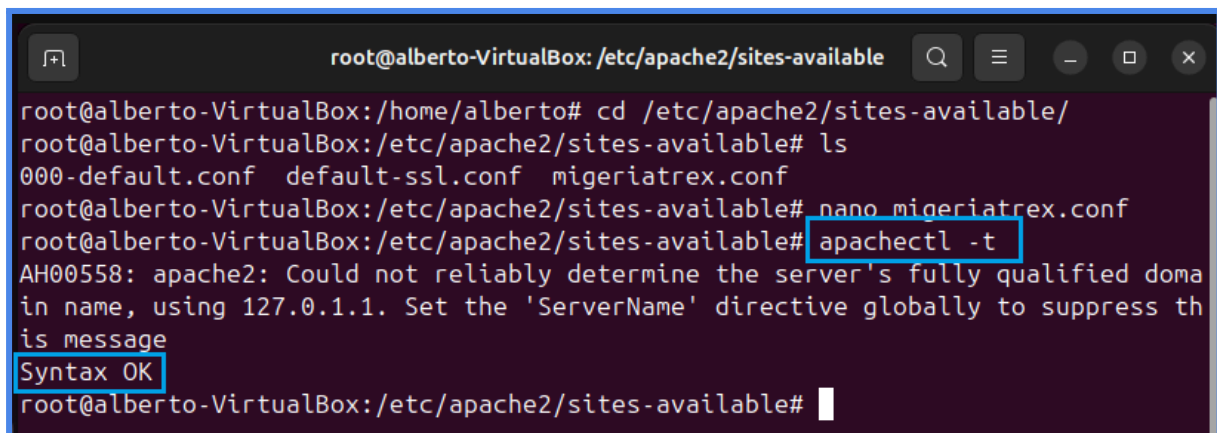
```
root@alberto-VirtualBox: /etc/apache2
root@alberto-VirtualBox:/etc/apache2# htpasswd claves.txt admon1 1
New password:
Re-type new password:
Adding password for user admon1
root@alberto-VirtualBox:/etc/apache2# htpasswd claves.txt admon2 2
New password:
Re-type new password:
Adding password for user admon2
root@alberto-VirtualBox:/etc/apache2# htpasswd claves.txt admon3 3
New password:
Re-type new password:
Adding password for user admon3
root@alberto-VirtualBox:/etc/apache2# htpasswd claves.txt admon4 4
New password:
Re-type new password:
Adding password for user admon4
root@alberto-VirtualBox:/etc/apache2#
```

Explicación foto: Una vez creados los usuarios doctores procedemos a crear los usuarios admon. Para ello usaremos el comando `htpasswd claves.txt admon1`⁽¹⁻²⁻³⁻⁴⁾. Como se puede observar en la imagen no hace falta añadir `-c` ya que claves.txt ya está creado.



```
root@alberto-VirtualBox: /etc/apache2
root@alberto-VirtualBox:/etc/apache2# cat claves.txt
doctor1:$apr1$c5hlxMkC$.5.BywjdLeh08TzJbZF0f/
doctor2:$apr1$pVZ/h52F$dtBS3FIdF/K7gjtYbZsvs/
doctor3:$apr1$iJKPqWAR$jvBBnSyEeQe59WMk56jY.
doctor4:$apr1$Lm3X0aU/$eKLJrTmuwfd9tvDtqK.4S.
admon1:$apr1$K6REizJ1$EZCNexQLMcnm11wKpnB0h.
admon2:$apr1$A31Gbb6b$LkaKjgxqJ503m8KLAI5911
admon3:$apr1$YZWm2YM5$iY0eH55EZ0LLAJg2ySkJi1
admon4:$apr1$N0kSRIg8$9gkgTpgLMfinVQVrf6YB/.
root@alberto-VirtualBox:/etc/apache2#
```

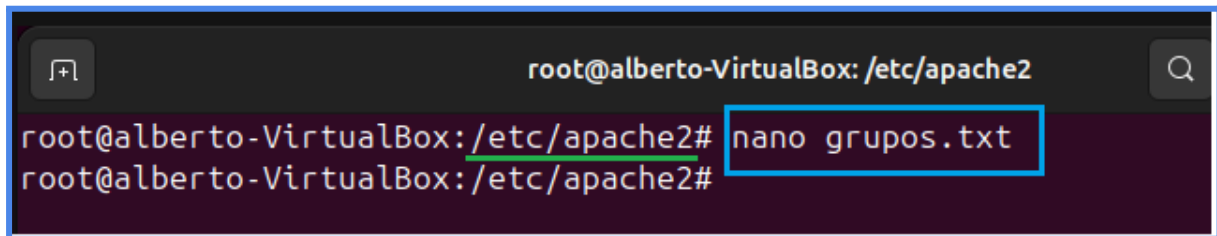
Explicación foto: Una vez creado todos los usuarios comprobamos el fichero claves.txt para ello usaremos el comando `cat claves.txt`.



```
root@alberto-VirtualBox: /etc/apache2/sites-available
root@alberto-VirtualBox:/home/alberto# cd /etc/apache2/sites-available/
root@alberto-VirtualBox:/etc/apache2/sites-available# ls
000-default.conf  default-ssl.conf  migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# nano migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# apachectl -t
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
root@alberto-VirtualBox:/etc/apache2/sites-available#
```

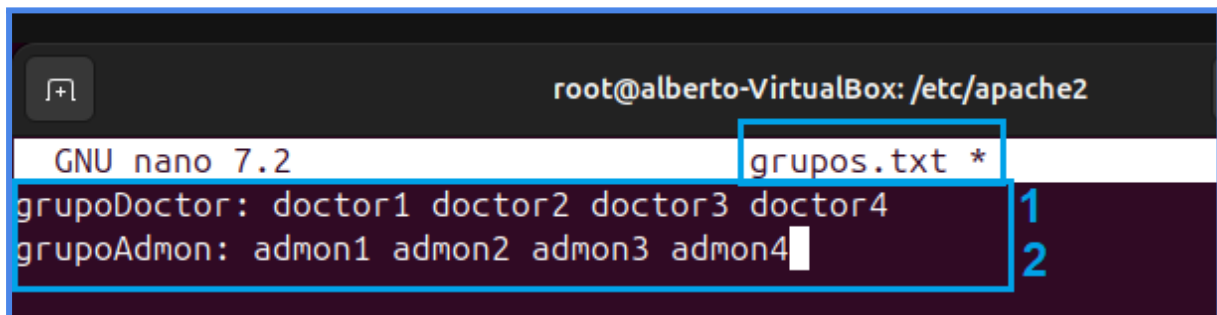
Explicación foto: Realizamos un `apachectl -t` para comprobar que la sintaxis es correcta.

Paso 4 - Creación de grupos.



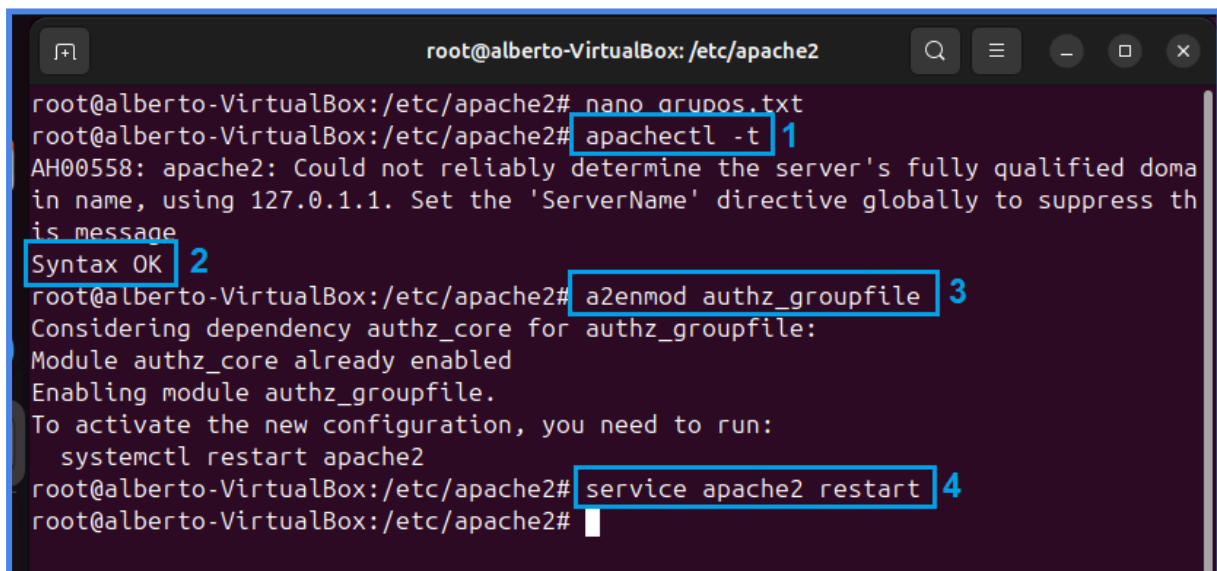
```
root@alberto-VirtualBox: /etc/apache2
root@alberto-VirtualBox:/etc/apache2# nano grupos.txt
root@alberto-VirtualBox:/etc/apache2#
```

Explicación foto: Para la creación de grupos dentro de **/etc/apache2** haremos un **nano grupos.txt** para crear el fichero y modificar su contenido.



```
GNU nano 7.2 grupos.txt *
grupoDoctor: doctor1 doctor2 doctor3 doctor4 1
grupoAdmon: admon1 admon2 admon3 admon4 2
```

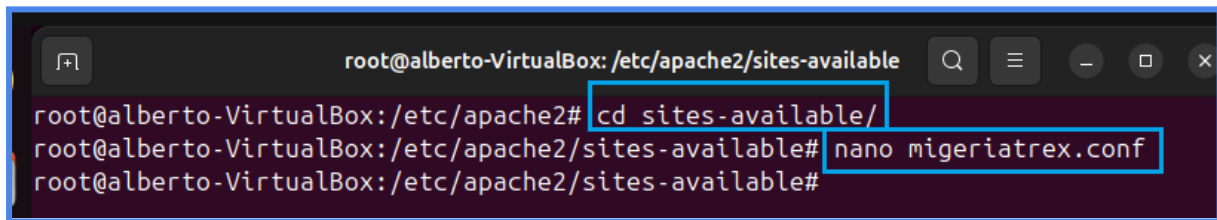
Explicación foto: Una vez dentro de **grupos.txt** añadimos los nombres de los grupos que vamos a tener y le asignamos los usuarios que van a pertenecer a cada grupo.



```
root@alberto-VirtualBox:/etc/apache2# nano grupos.txt
root@alberto-VirtualBox:/etc/apache2# apachectl -t 1
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK 2
root@alberto-VirtualBox:/etc/apache2# a2enmod authz_groupfile 3
Considering dependency authz_core for authz_groupfile:
Module authz_core already enabled
Enabling module authz_groupfile.
To activate the new configuration, you need to run:
    systemctl restart apache2
root@alberto-VirtualBox:/etc/apache2# service apache2 restart 4
root@alberto-VirtualBox:/etc/apache2#
```

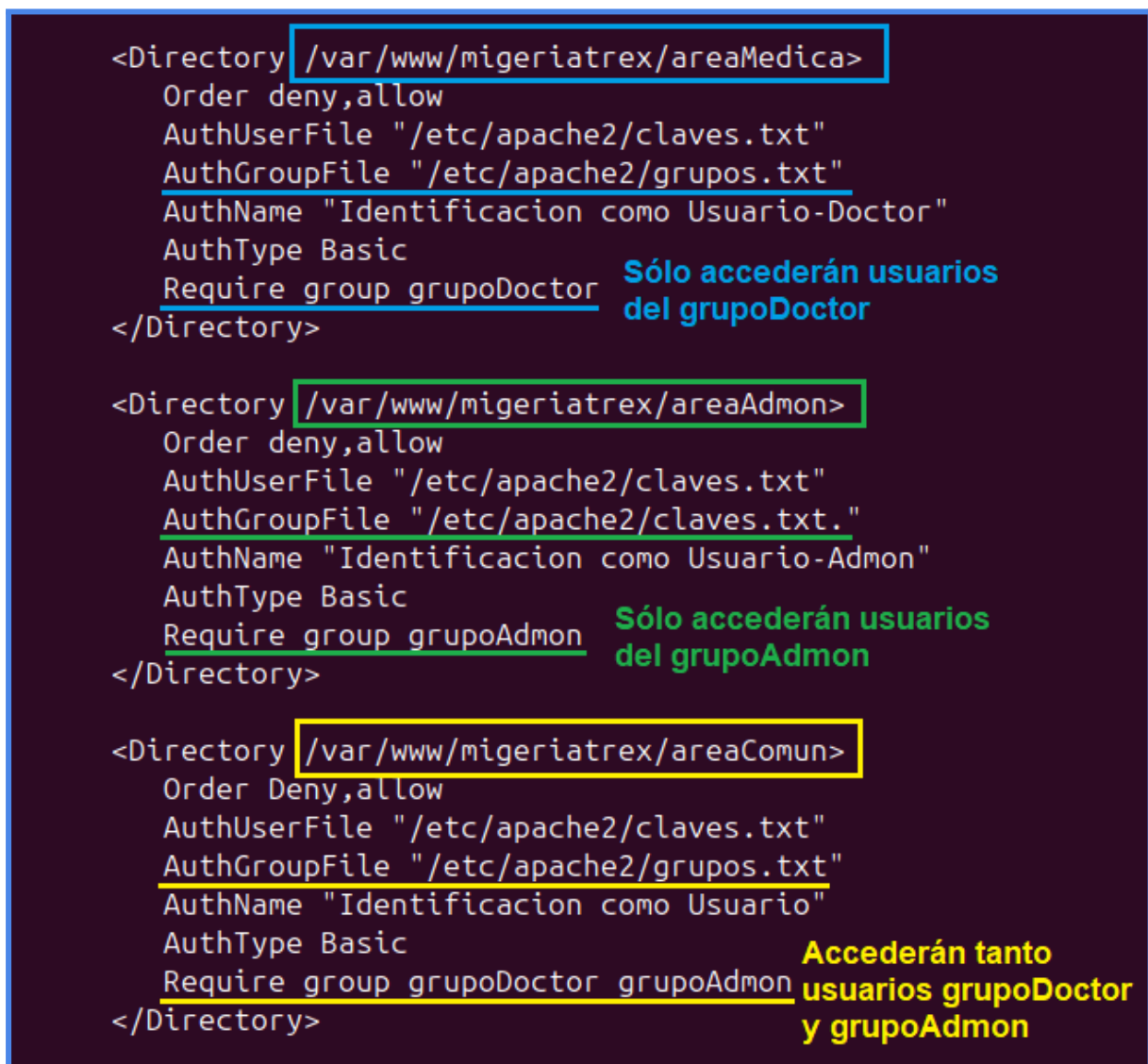
Explicación foto: Una vez modificado el archivo grupos.txt haremos un **apachectl -t**⁽¹⁾ para comprobar la sintaxis⁽²⁾ y procederemos a activar el módulo de los grupos. Para ello usaremos el comando **a2enmod authz_groupfile**⁽³⁾ y una vez activado el módulo haremos un restart → **service apache2 restart**⁽⁴⁾.

3.4.15 - Asignación de permisos de grupo a cada fichero.



```
root@alberto-VirtualBox: /etc/apache2/sites-available
root@alberto-VirtualBox:/etc/apache2# cd sites-available/
root@alberto-VirtualBox:/etc/apache2/sites-available# nano migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available#
```

Explicación foto: Dentro de la ruta `/etc/apache2/sites-available` haremos un `nano migeriatrex.conf` para añadir los grupos que van a tener permisos para visualizar el contenido de cada fichero de nuestro sitio web.



```
<Directory /var/www/migeriatrex/areaMedica>
    Order deny,allow
    AuthUserFile "/etc/apache2/claves.txt"
    AuthGroupFile "/etc/apache2/grupos.txt"
    AuthName "Identificacion como Usuario-Doctor"
    AuthType Basic
    Require group grupoDoctor
</Directory>

<Directory /var/www/migeriatrex/areaAdmon>
    Order deny,allow
    AuthUserFile "/etc/apache2/claves.txt"
    AuthGroupFile "/etc/apache2/claves.txt."
    AuthName "Identificacion como Usuario-Admon"
    AuthType Basic
    Require group grupoAdmon
</Directory>

<Directory /var/www/migeriatrex/areaComun>
    Order Deny,allow
    AuthUserFile "/etc/apache2/claves.txt"
    AuthGroupFile "/etc/apache2/grupos.txt"
    AuthName "Identificacion como Usuario"
    AuthType Basic
    Require group grupoDoctor grupoAdmon
</Directory>
```

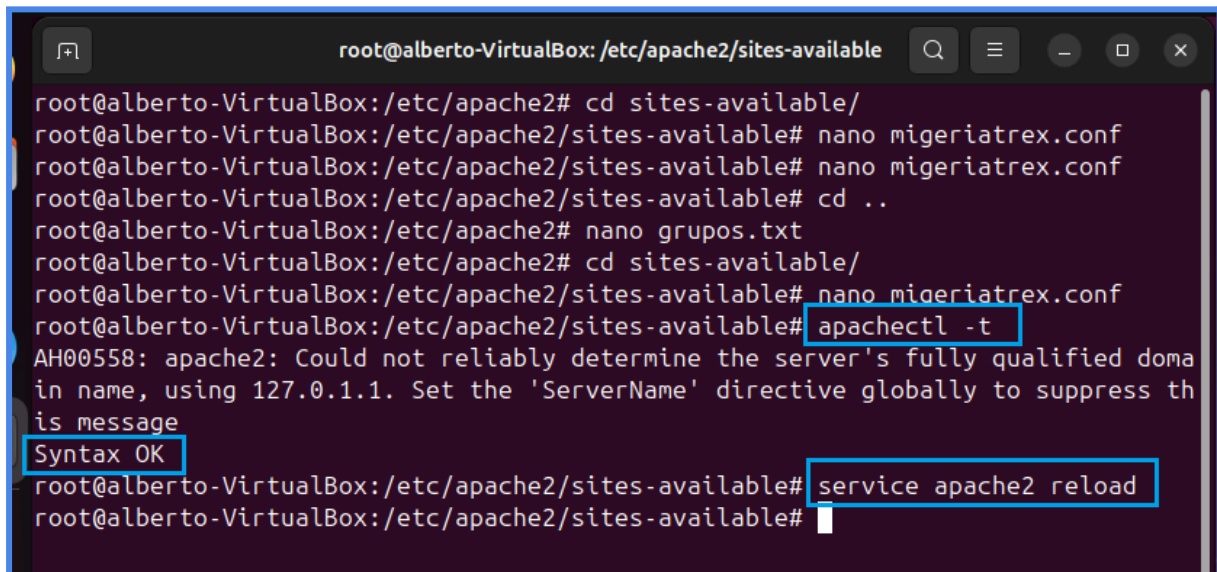
Sólo accederán usuarios del grupoDoctor

Sólo accederán usuarios del grupoAdmon

Accederán tanto usuarios grupoDoctor y grupoAdmon

Explicación foto: Ahora modificaremos los permisos que hicimos previamente en nuestro archivo `migeriatrex.conf` y añadiremos las líneas **AuthGroupFile** a cada directorio.

***Corrección:** En el apartado del directorio **areaComun** la línea **AuthGroupFile** “/etc/apache2/claves.txt. la cambiamos por → **AuthGroupFile** “/etc/apache2/grupos.txt.



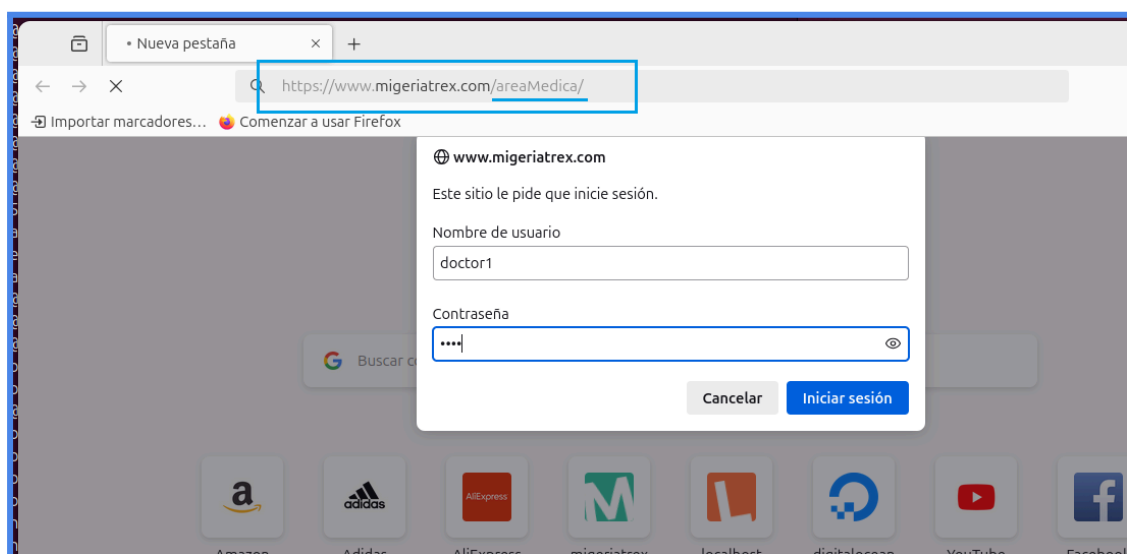
```
root@alberto-VirtualBox: /etc/apache2/sites-available
root@alberto-VirtualBox:/etc/apache2# cd sites-available/
root@alberto-VirtualBox:/etc/apache2/sites-available# nano migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# nano migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# cd ..
root@alberto-VirtualBox:/etc/apache2# nano grupos.txt
root@alberto-VirtualBox:/etc/apache2# cd sites-available/
root@alberto-VirtualBox:/etc/apache2/sites-available# nano migeriatrex.conf
root@alberto-VirtualBox:/etc/apache2/sites-available# apachectl -t
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
root@alberto-VirtualBox:/etc/apache2/sites-available# service apache2 reload
root@alberto-VirtualBox:/etc/apache2/sites-available#
```

Explicación foto: Una vez modificado el archivo migeriatrex.conf procedemos nuevamente a hacer un **apachectl -t** para comprobar la sintaxis y realizaremos un **service apache2** reload para recargar el servidor.

3.4.16 - Comprobación de autenticación de usuarios.

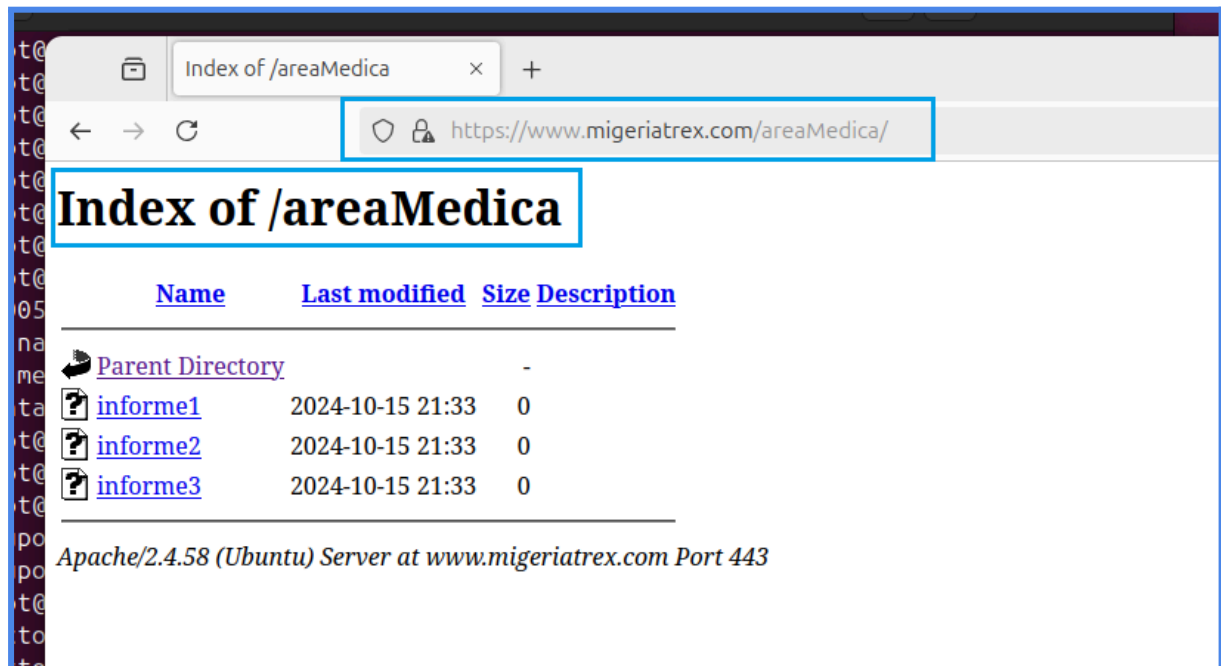
Ahora procederemos a comprobar si los usuarios tienen los permisos en cada área de nuestro proyecto.

Paso 1 - Comprobar área médica.



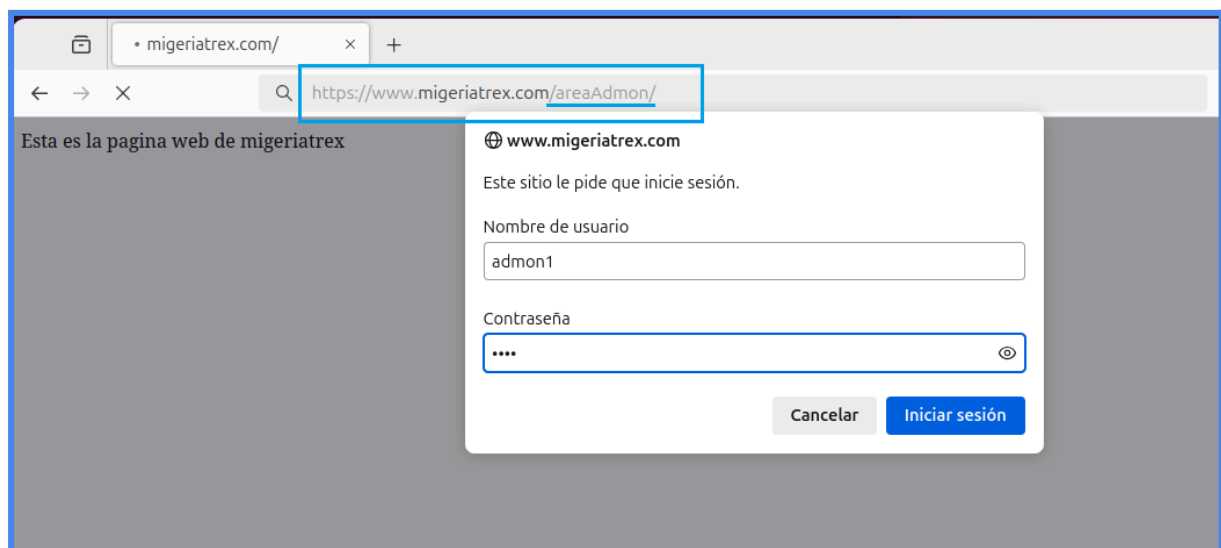
Explicación foto: Procedemos a logearnos con un usuario doctor(**doctor1**) dentro del **areaMedica** de nuestro sitio web para ver si tenemos acceso para ver el contenido de la página en caso de que no pueda la página volverá a pedir el usuario y su contraseña.

Paso 1.1 - Resultado de vista área médica.



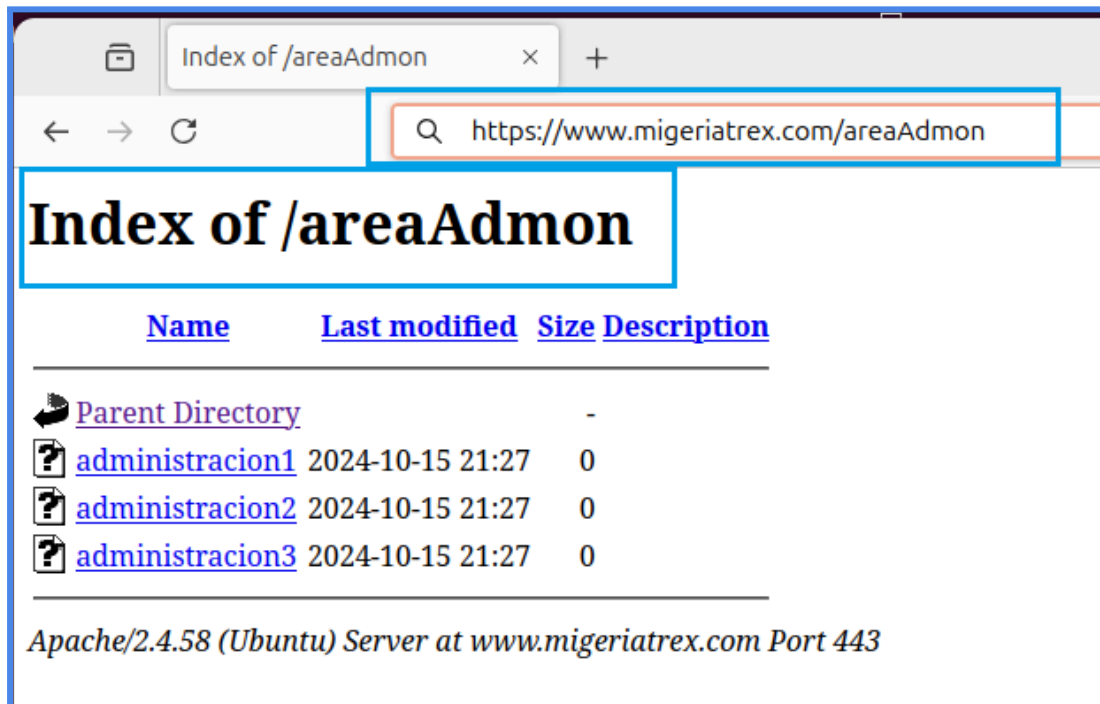
Explicación foto: Nos hemos logueado satisfactoriamente dentro de la página como **doctor1** y podemos ver el contenido de la página.

Paso 2 - Comprobar área médica.



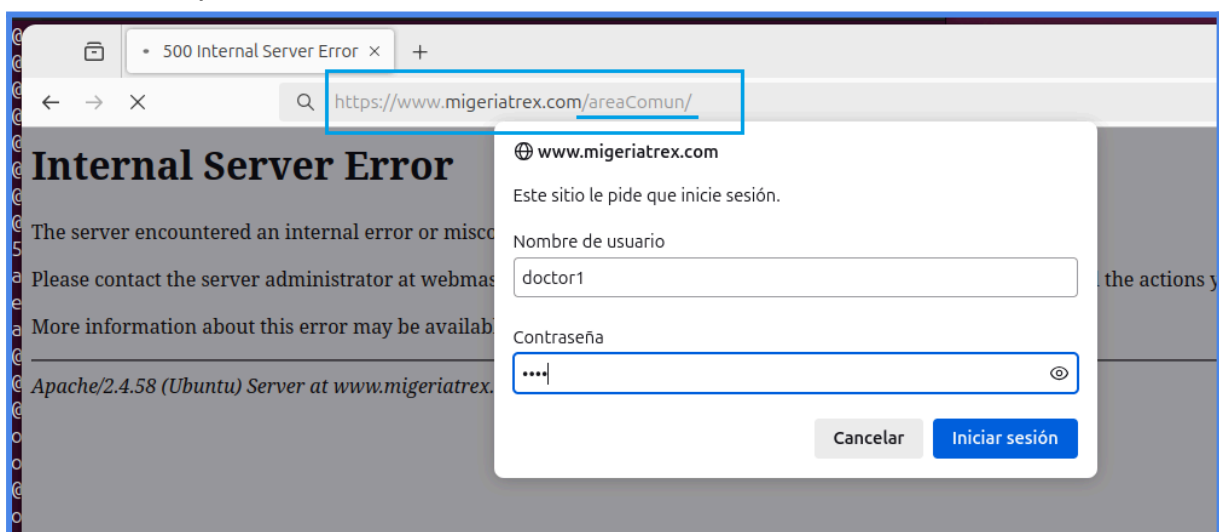
Explicación foto: Procedemos a logearnos con un usuario administrativo(**admon1**) dentro del **areaAdmon** de nuestro sitio web para ver si tenemos acceso para ver el contenido de la página en caso de que no pueda la página volverá a pedir el usuario y su contraseña.

Paso 2.2 - Comprobar área administrativa.



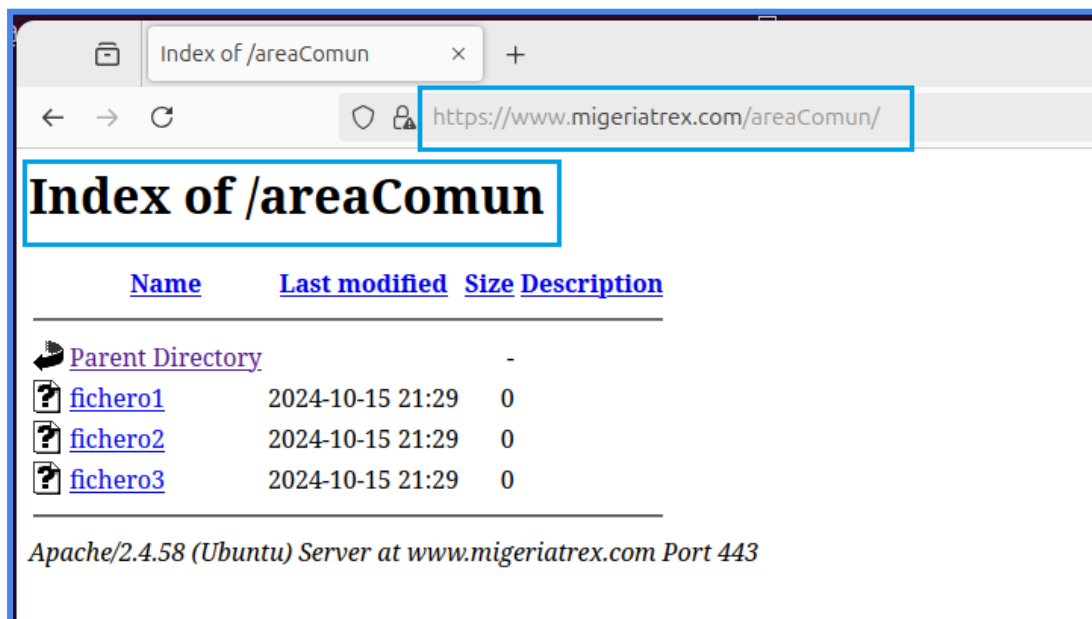
Explicación foto: Nos hemos logueado satisfactoriamente dentro de la página como **admon1** y podemos ver el contenido de la página.

Paso 3 - Comprobar área común.



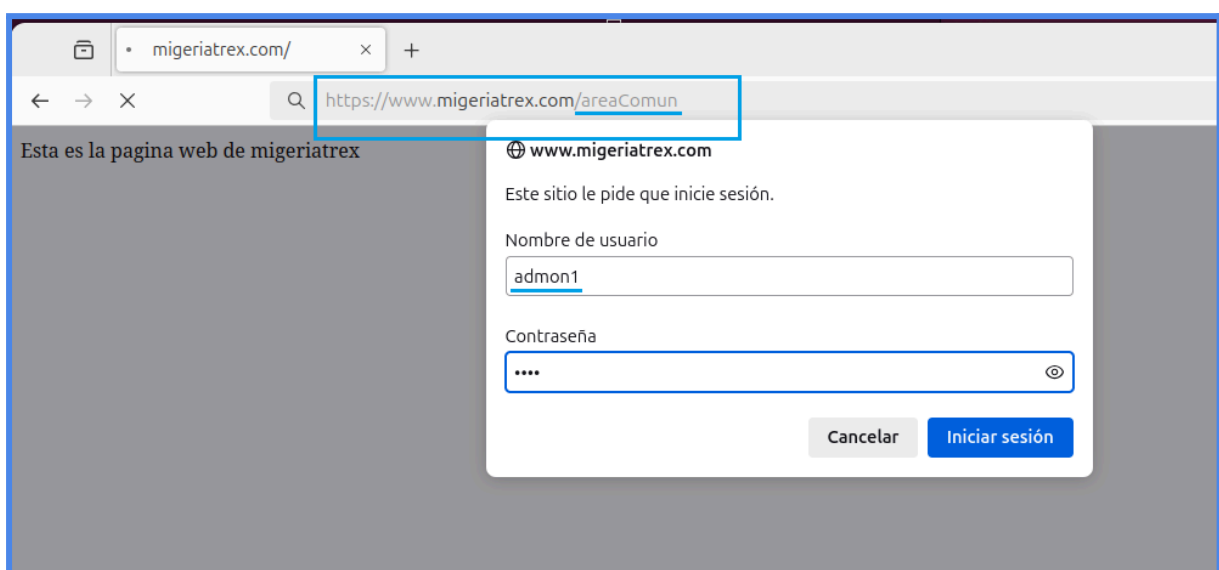
Explicación foto: Procedemos a logearnos con un usuario doctor(**doctor1**) dentro del **areaComun** de nuestro sitio web para ver si tenemos acceso para ver el contenido de la página en caso de que no pueda la página volverá a pedir el usuario y su contraseña.

Paso 3.2 - Comprobar área común después de logearnos como doctor.



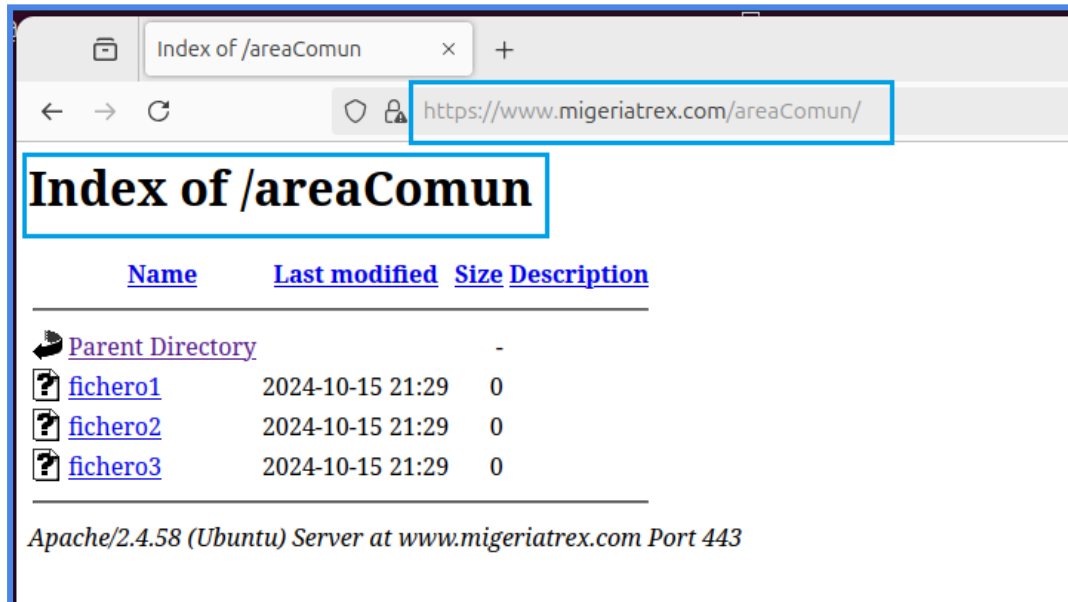
Explicación foto: Nos hemos logueado satisfactoriamente dentro de la página como **doctor1** y podemos ver el contenido de la página.

Paso 3.3 - Comprobar área común como administrativo.



Explicación foto: Procedemos a logearnos con un usuario doctor(**admon1**) dentro del **areaComun** de nuestro sitio web para ver si tenemos acceso para ver el contenido de la página en caso de que no pueda la página volverá a pedir el usuario y su contraseña.

Paso 3.4 - Comprobar área común después de logearnos como administrativo.



3.4.17 - Habilitar nuestro dominio web para que se pueda ver desde Windows.

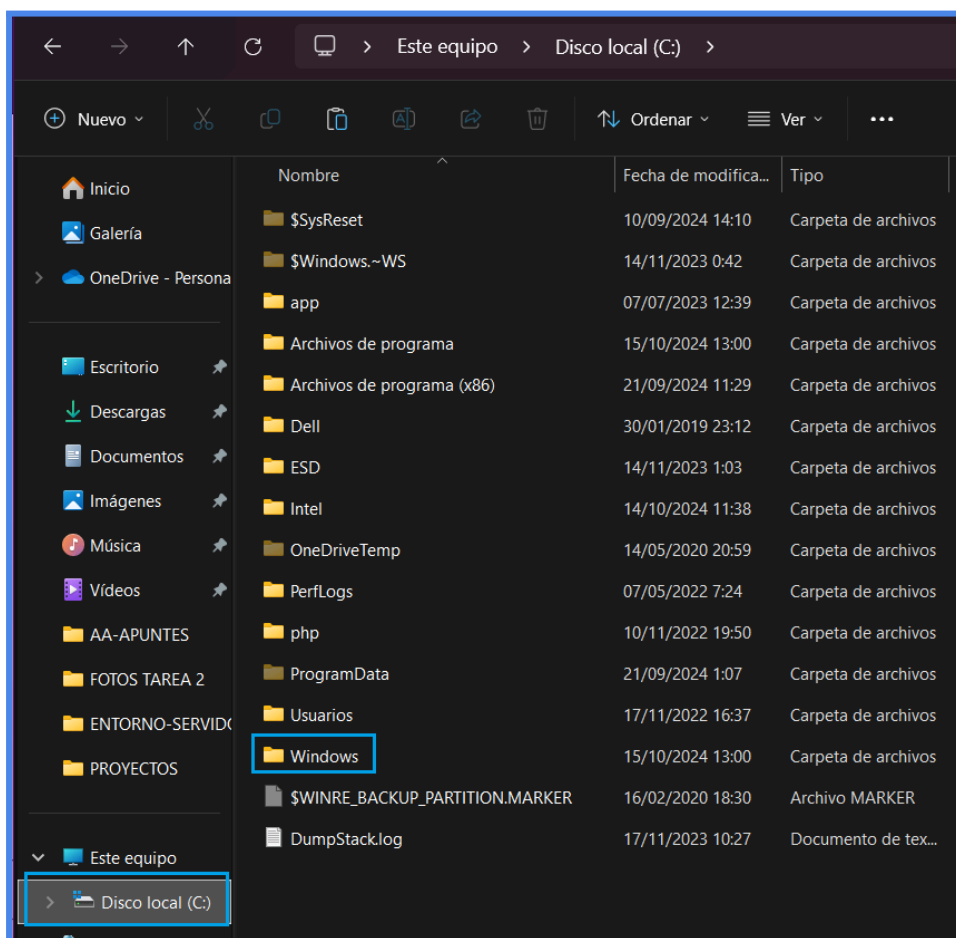
Ahora procederemos a habilitar nuestro dominio para poder verlo fuera de la máquina Ubuntu.

```
root@alberto-VirtualBox: /var/www/migeriatrex
Preparando para desempaquetar .../net-tools_2.10-0.1ubuntu4_amd64.deb ...
Desempaquetando net-tools (2.10-0.1ubuntu4) ...
Configurando net-tools (2.10-0.1ubuntu4) ...
Procesando disparadores para man-db (2.12.0-4build2) ...
root@alberto-VirtualBox:/var/www/migeriatrex# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.151 netmask          broadcast
    inet6          prefixlen 64 scopeid 0x20<link>
    ether          txqueuelen 1000 (Ethernet)
    RX packets 49674 bytes 17135262 (17.1 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8928 bytes 1321454 (1.3 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

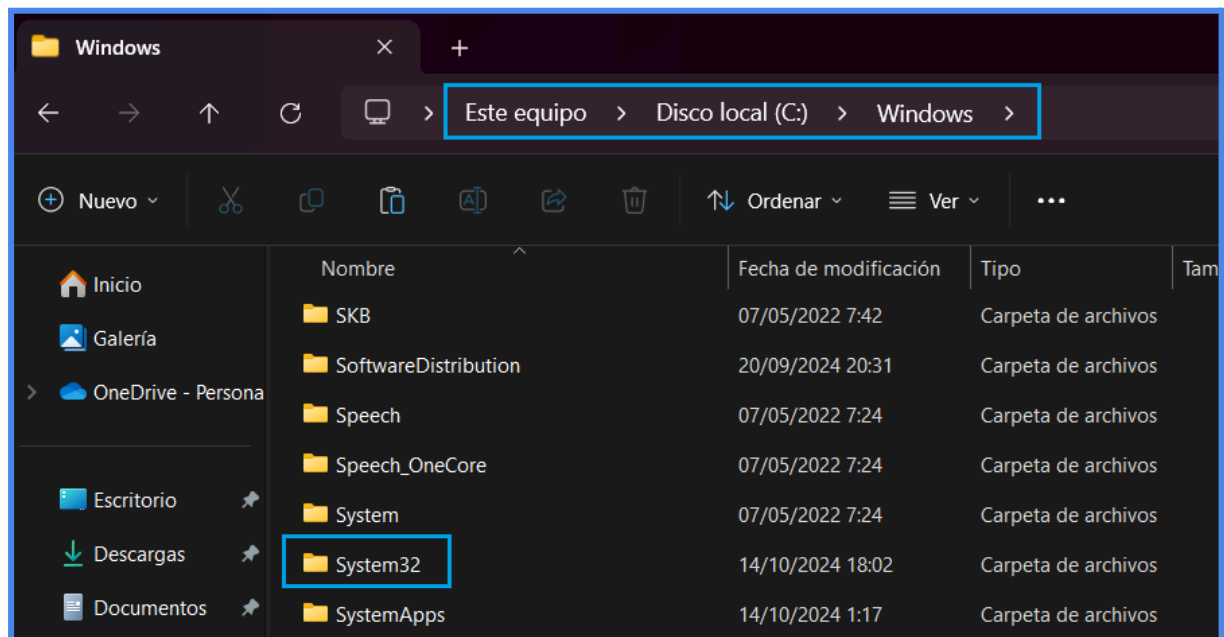
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet          netmask
    inet6          prefixlen scopeid 0x10<host>
    loop txqueuelen 1000 (Bucle local)
    RX packets 1099 bytes 165301 (165.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1099 bytes 165301 (165.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@alberto-VirtualBox:/var/www/migeriatrex# S
```

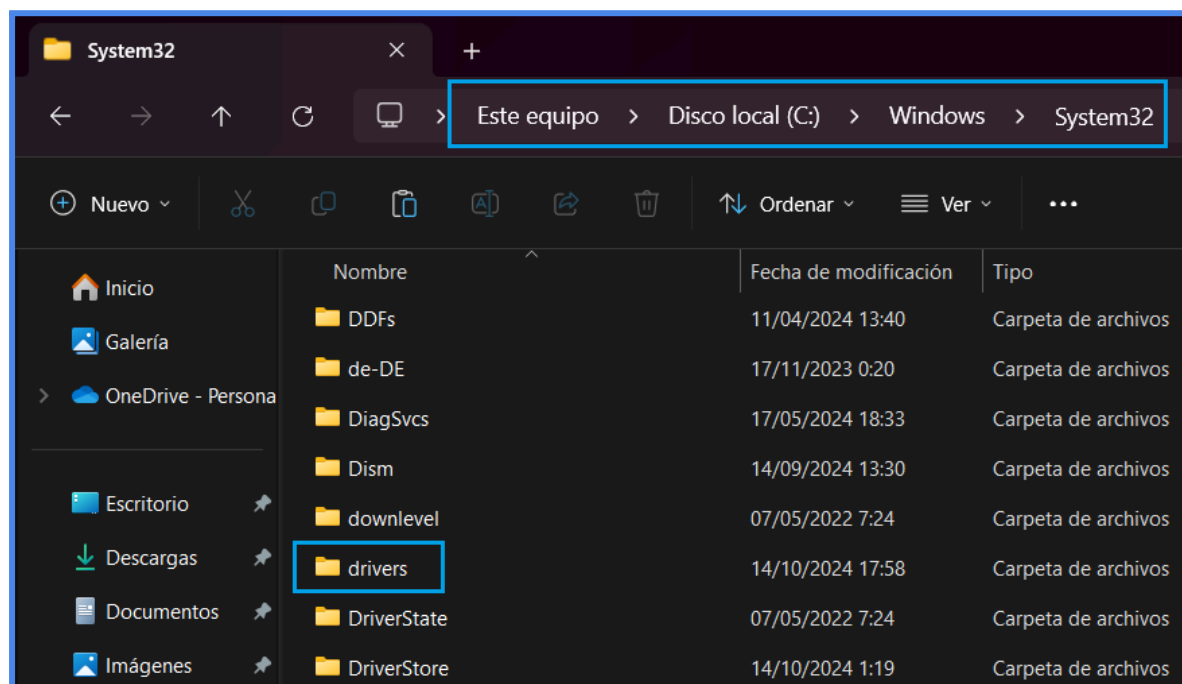
Explicación foto: Obtenemos la ip de nuestra máquina ubuntu.



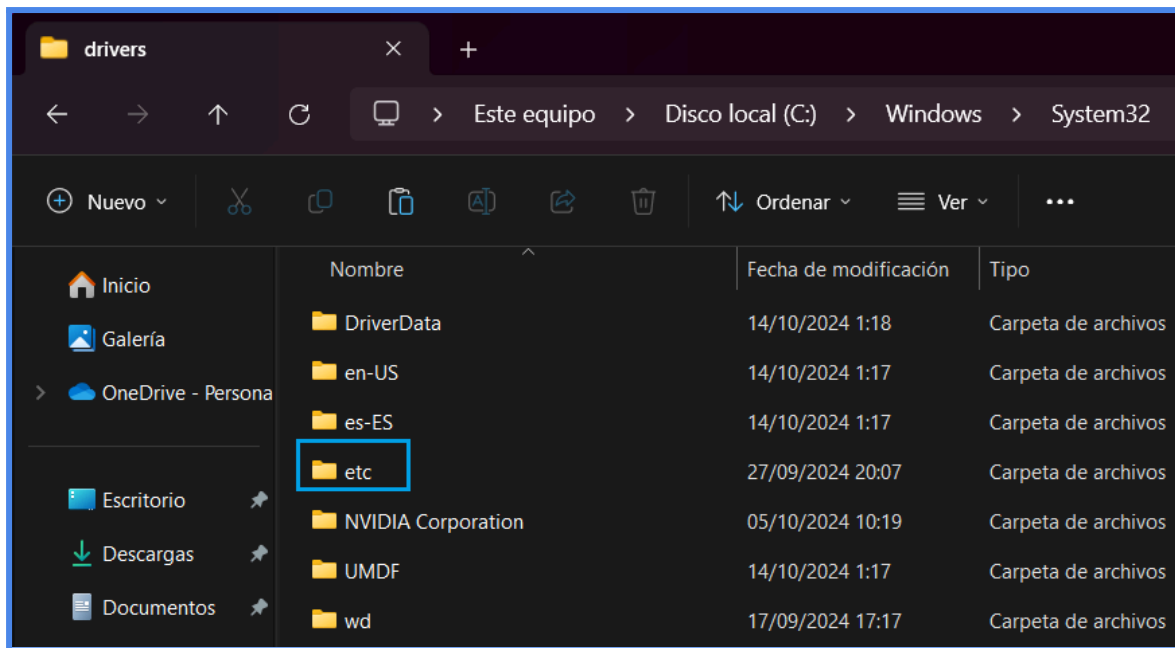
Explicación foto: En **Windows** accedemos al fichero **C/Windows**.



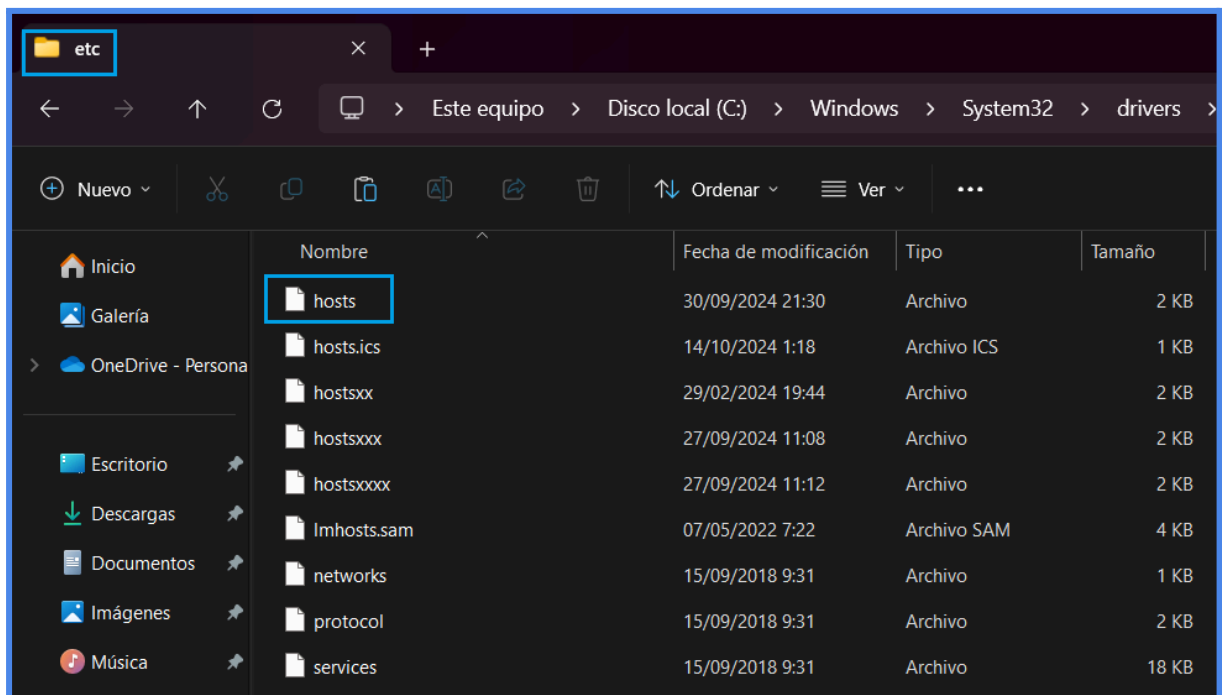
Explicación foto: En **Windows** accedemos al fichero **C/Windows/System32**.



Explicación foto: En **Windows** accedemos al fichero **C/Windows/System32/drivers**.



Explicación foto: En **Windows** accedemos al fichero `C/Windows/System32/drivers/etc`.



Explicación foto: Una vez dentro del fichero **etc** procederemos a modificar el archivo con nombre **hosts**.

```
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
# 192.168.100.151 rhino.acme.com      # source server
#                 x.acme.com         # x client host
#
# www.albertosanchezdaw2b.com
# www.gestordeficheros.com
# www.migeriatrex.com
#
# localhost name resolution is handled within DNS itself.
# 127.0.0.1 localhost
```

Explicación foto: Añadiremos la ip de nuestra máquina ubuntu y el dominio de nuestro sitio web y guardaremos el archivo dentro del fichero etc.



Explicación foto: Procederemos a buscar nuestro sitio web desde Windows y nos aparecerá nuestra página web.

4. Cronograma

Periodo estipulado para el desarrollo de la aplicación web: **5 meses**.

Planificación	Diseño	Desarrollo Frontend	Desarrollo Backend	Pruebas	Despliegue	Mantenimiento
2 Semanas	3 Semanas	5 Semanas	5 Semanas	3 Semanas	2 Semanas	Continuo