

MANUAL DE INSTALACIÓN PLATAFORMA VIRTUAL Versión 1.0

DESARROLLO DE UNA PLATAFORMA VIRTUAL DE BOLSA DE
PRÁCTICAS PREPROFESIONALES DE LA ESFOT

TRABAJO DE TITULACIÓN POR:

GABRIELA GARCÍA

WESTER MENDOZA

ÍNDICE DE CONTENIDOS

1.	Despliegue de la base de datos en Alwaysdata	1
1.1.	Creación de una nueva base MySQL en alwaysdata	1
1.2.	Crear usuario de base de datos.....	2
1.3.	Ingresar <i>phpMyAdmin</i>	3
2.	Despliegue del <i>backend</i> en <i>Hostinger</i>	4
3.	Despliegue del <i>frontend</i> en <i>Firebase</i>	10
4.	Credenciales de acceso para la plataforma virtual y la API.....	19
5.	Repositorio del código fuente de la plataforma web.....	19

1. Despliegue de la base de datos en Alwaysdata

A continuación, se detalla el procedimiento para realizar el despliegue a producción de la base de datos en *Alwaysdata*. Para lo cual se debe iniciar sesión en la plataforma de *Alwaysdata* como se ilustra en la **Fig. 1**.

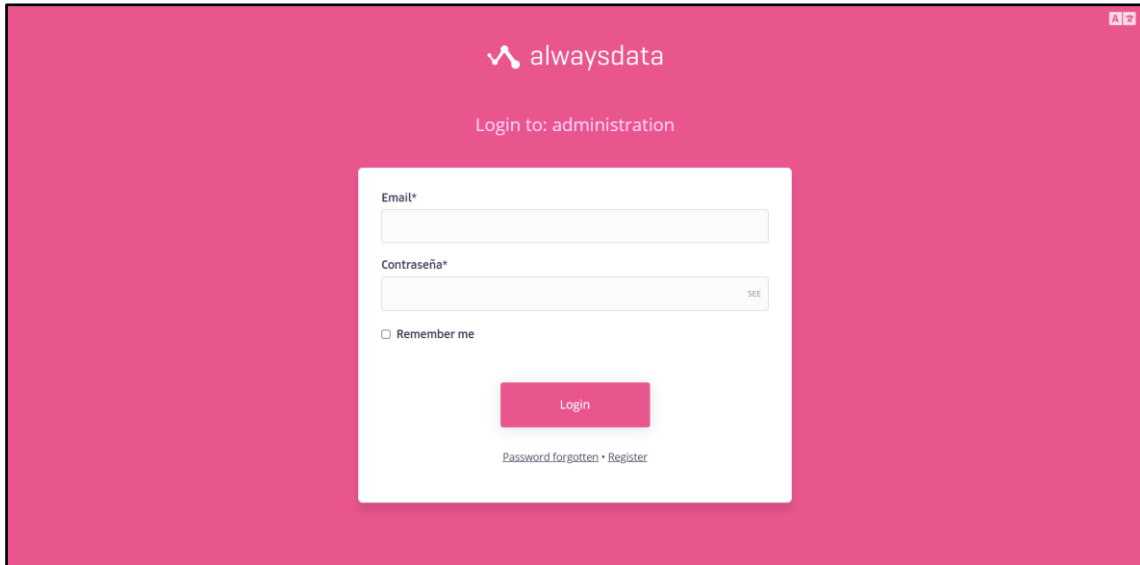


Fig. 1: Inicio de Sesión de Alwaysdata.

1.1. Creación de una nueva base MySQL en alwaysdata

Para crear una nueva base de datos *MySQL*, hacer clic en la opción “Databases” y escoger *MySQL*. Luego hacer clic, “Add database” llenar el formulario colocando el nombre de la base de datos, como ilustra la **Fig. 2**.

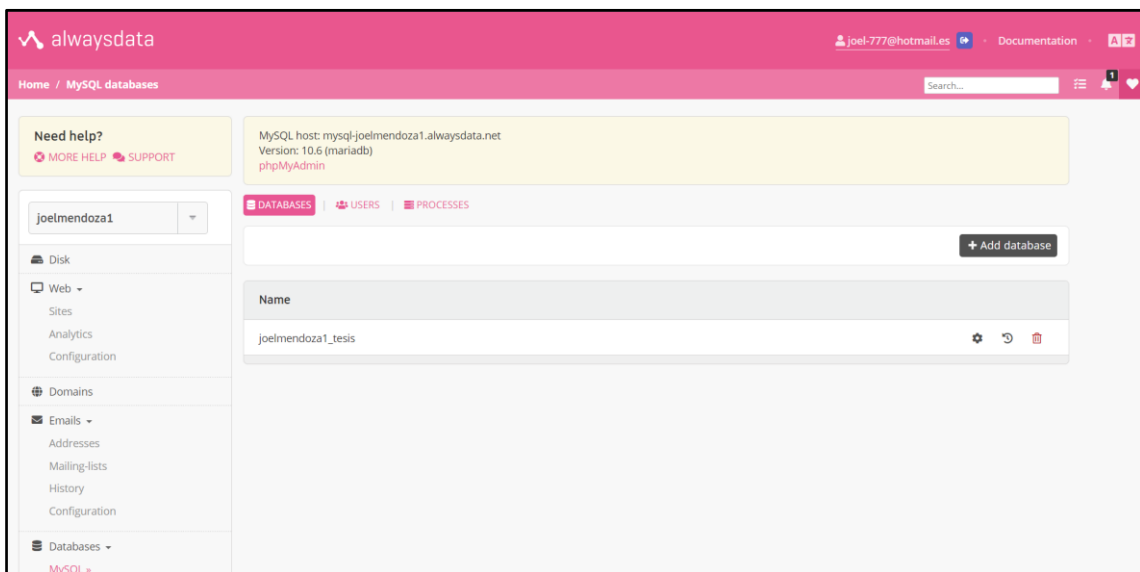


Fig. 2: Creación de base MySQL en Alwaysdata.

1.2. Crear usuario de base de datos

Para crear un usuario de la base de datos, entrar en la opción “Users”, hacer clic en “Add User”, tal como ilustra la **Fig. 3**, luego llenar el formulario colocando el nombre de usuario y contraseña.

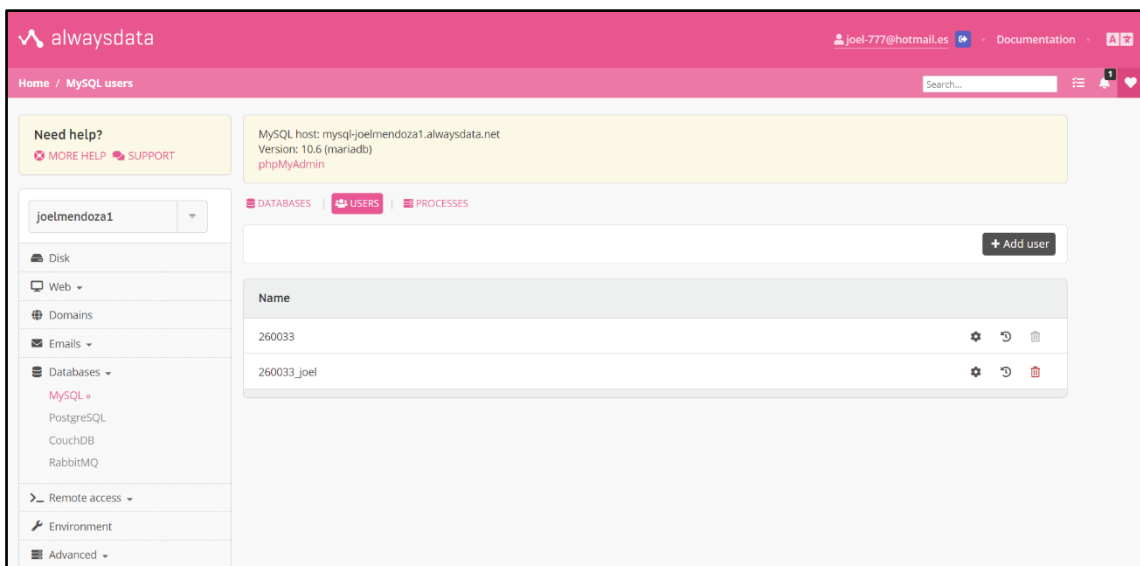


Fig. 3: Creación de usuario de base de datos.

1.3. Ingresar *phpMyAdmin*.

Una creada la base de datos y el usuario ingresar en la opción *phpMyAdmin*, ingresa a inicio de sesión para la base de datos como ilustra la **Fig. 4**, luego colocar las credenciales del usuario posteriormente creado.

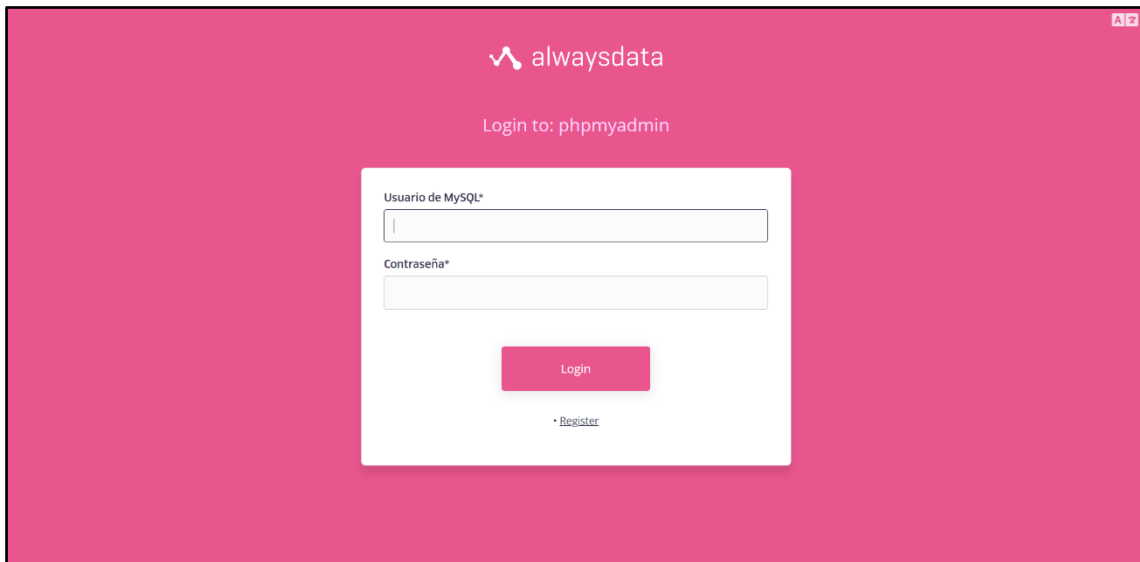


Fig. 4: Inicio de sesión para phpMyadmin.

Una vez iniciada la sesión en *phpMyAdmin*, mostrara una plataforma como se ilustra en la **Fig. 5**.

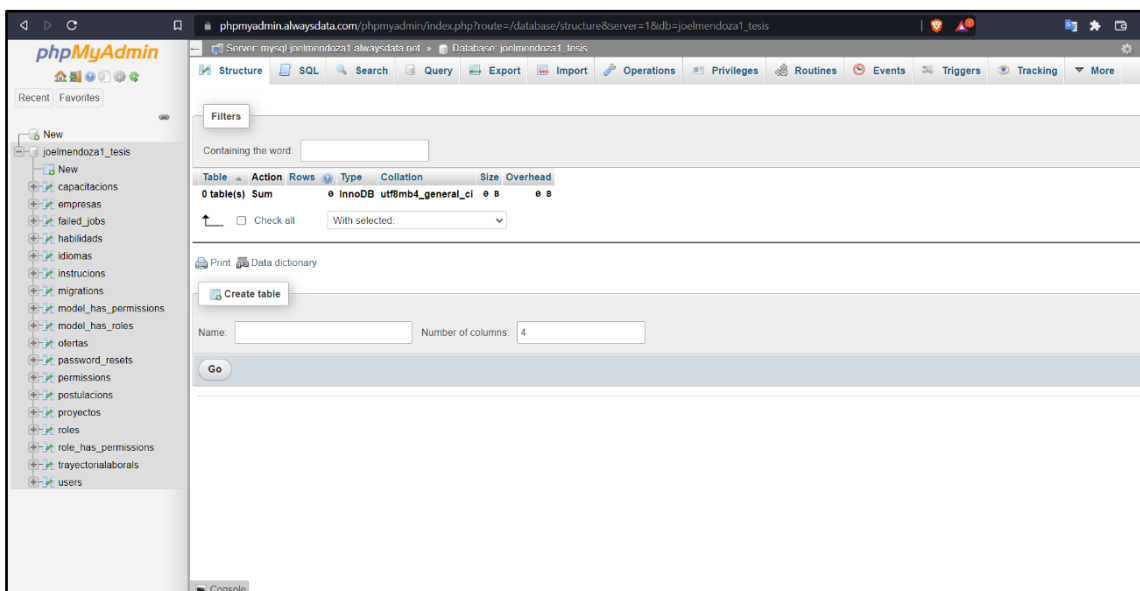


Fig. 5: phpMyAdmin de alwaysdata.

Se procede a configurar en el documento “.env” colocando los parámetros de conexión para la base de datos, en “DB_CONNECTION” se coloca “mysql”, en “DB_HOST” se coloca “mysql-joelmendoza1.alwaysdata.net”, donde se coloca el host de la base, en “DB_PORT” se coloca “3306”, el cual es el puerto de conexión por defecto de conexión a la base, en “DB_DATABASE” se coloca “joelmendoza1_tesis”, el cual es el nombre de la base de datos, en “DB_USERNAME” se coloca “260033_joel”, donde se coloca el nombre de usuario y por último la contraseña de la base en “DB_PASSWORD” se coloca “123123Joel@/”, configurando tal y como se muestra la en la **Fig. 6**. Esto permite la conexión del proyecto de *Laravel* con la base en *alwaysdata*.

```
DB_CONNECTION=mysql
DB_HOST=mysql-joelmendoza1.alwaysdata.net
DB_PORT=3306
DB_DATABASE=joelmendoza1_tesis
DB_USERNAME=260033_joel
DB_PASSWORD=123123Joel@/
|
```

Fig. 6: Configuración del .env

2. Despliegue del *backend* en *Hostinger*.

A continuación, se detalla el procedimiento para realizar el despliegue a producción del *backend* de la plataforma virtual en *Hostinger*.

Primero comprimir el directorio del proyecto como archivo zip, lo ideal es que se haya borrado la carpeta *vendor* para que el zip no sea tan pesado. El archivo zip lo cargamos y descomprimos en el administrador de archivos de nuestro sitio en *Hostinger* como se observa en la **Fig. 7**














<input type="checkbox"/>	Name ▾	Size	Date	Permissions
<input type="checkbox"/>	 .git		2022-03-27 10:53:00	drwxr-xr-x
<input type="checkbox"/>	 .idea		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 .vercel		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 app		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 bootstrap		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 config		2022-03-27 10:19:00	drwxr-xr-x
<input type="checkbox"/>	 database		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 public		2022-03-27 10:42:00	drwxrwxrwx
<input type="checkbox"/>	 public_html		2022-03-27 10:42:00	drwxrwxrwx
<input type="checkbox"/>	 resources		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 routes		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 storage		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 tests		2022-03-27 08:55:00	drwxr-xr-x

Fig. 7: Descompresión del archivo zip

Ir al *dashboard* de *Hostinger* y dar clic en la opción Acceso SSH como se ilustra en la Fig. 8.



Fig. 8: Dashboard de Hostinger

Habilitar la opción "Administrar acceso SSH" y copiar el valor de "Comando SSH CLI", como indica la **¡Error! No se encuentra el origen de la referencia.**

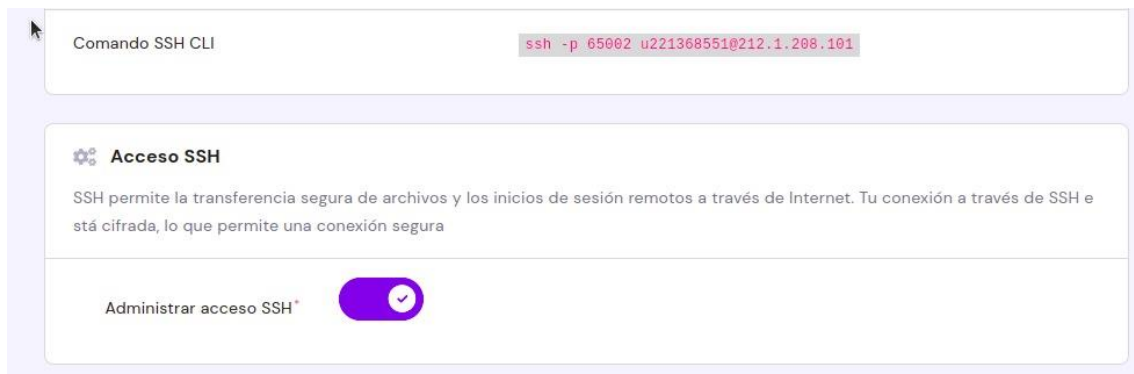


Fig. 9: Administración de acceso SSH

Abrir una terminal y ejecutar el comando copiado anteriormente, esto se lo puede observar en la **Fig. 10**.

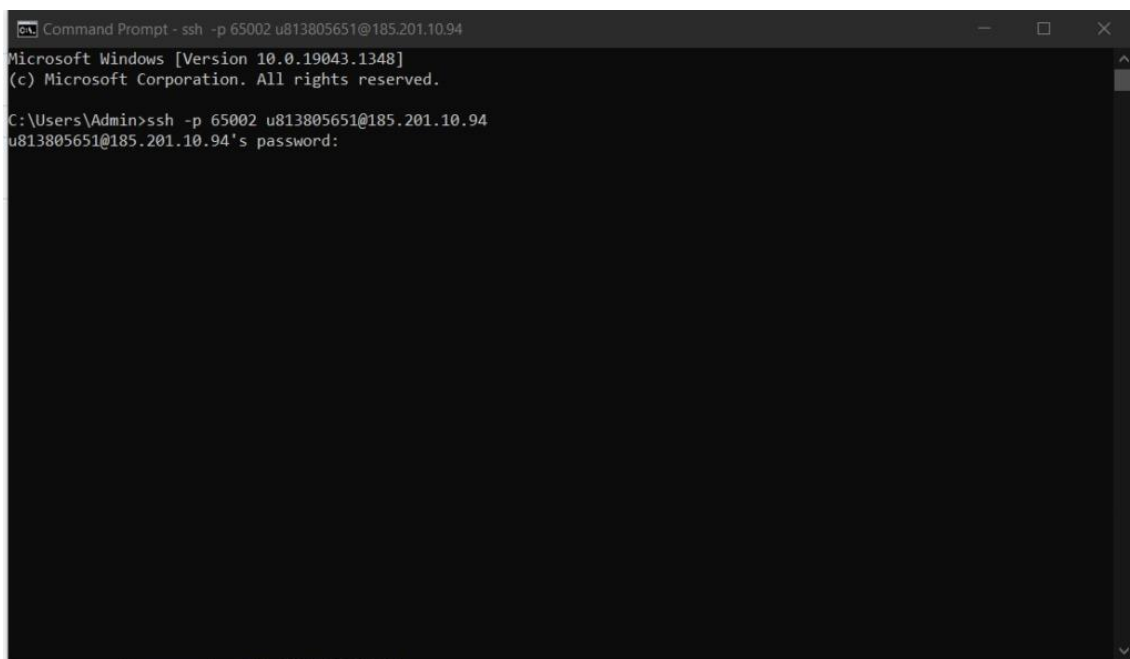
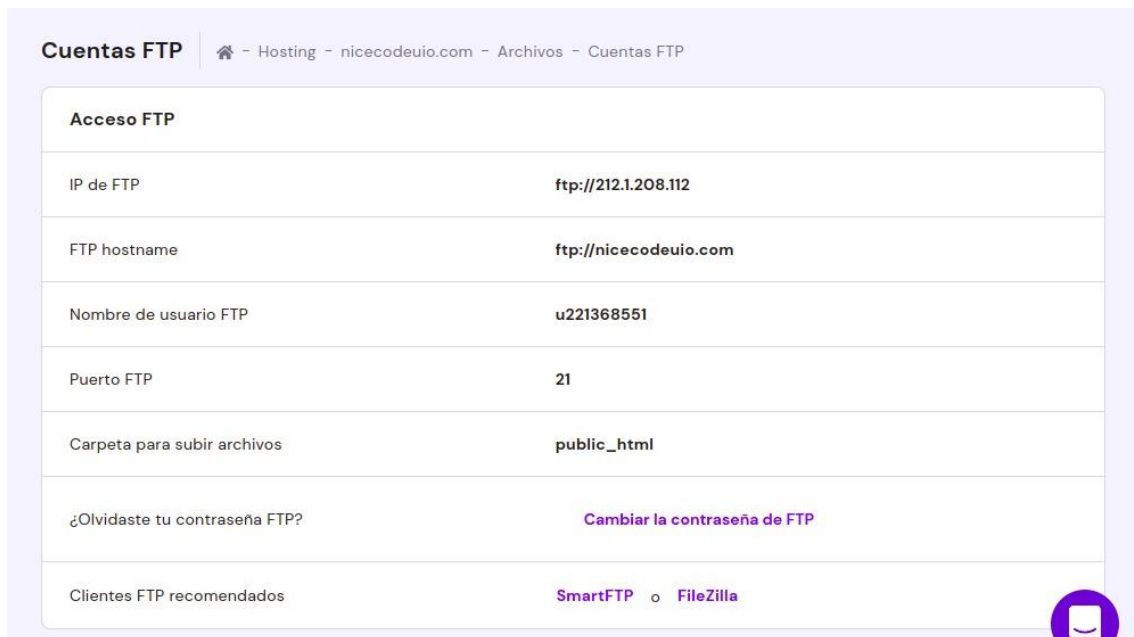


Fig. 10: Ejecución del comando SSH

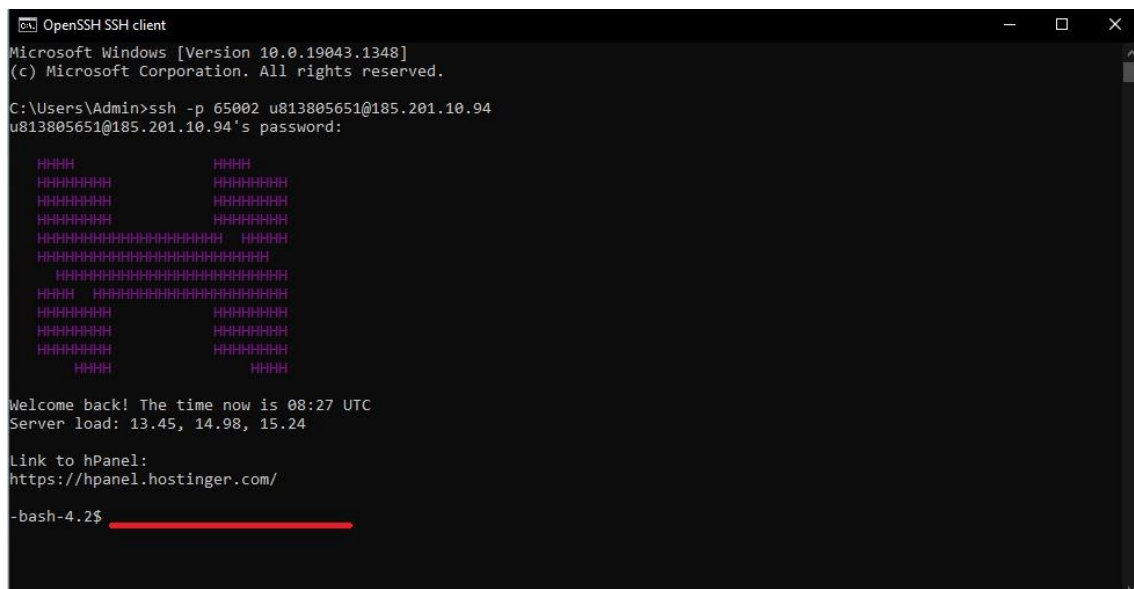
Para obtener la contraseña solicitada, ir a "Cuentas FTP", como se observa en la **Fig. 11**.



Acceso FTP	
IP de FTP	ftp://212.1.208.112
FTP hostname	ftp://nicecodeuio.com
Nombre de usuario FTP	u221368551
Puerto FTP	21
Carpeta para subir archivos	public_html
¿Olvidaste tu contraseña FTP?	Cambiar la contraseña de FTP
Clientes FTP recomendados	SmartFTP o FileZilla

Fig. 11: Contraseña de cuenta FTP

En la terminal colocar la contraseña obtenida en el paso anterior y apretar *enter*, esto se observa en la **Fig. 12**



```
OpenSSH SSH client
Microsoft Windows [Version 10.0.19043.1348]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>ssh -p 65002 u813805651@185.201.10.94
u813805651@185.201.10.94's password:

Welcome back! The time now is 08:27 UTC
Server load: 13.45, 14.98, 15.24

Link to hPanel:
https://hpanel.hostinger.com/

-bash-4.2$
```

Fig. 12: Ingresar la contraseña obtenida.

Copiar el contenido de la carpeta *public* a la carpeta *public_html*, ilustrado en la **Fig. 13**.

<input type="checkbox"/>	 .htaccess	0.6 kB	2022-03-27 09:28:00	-rwx-r--r--
<input type="checkbox"/>	 favicon.ico	0.1 kB	2022-03-27 08:58:00	-rwx-r--r--
<input type="checkbox"/>	 index.php	1.8 kB	2022-03-27 08:58:00	-rwx-r--r--
<input type="checkbox"/>	 robots.txt	0.1 kB	2022-03-27 08:58:00	-rwx-r--r--
<input type="checkbox"/>	 web.config	1.2 kB	2022-03-27 08:58:00	-rwx-r--r--

Fig. 13: Copiar contenido de carpetas

Volver al administrador de archivos y eliminar el archivo llamado *default.php*, esto se evidencia en la **Fig. 14**














<input type="checkbox"/>	Name ▾	Size	Date	Permissions
<input type="checkbox"/>	 .git		2022-03-27 10:53:00	drwxr-xr-x
<input type="checkbox"/>	 .idea		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 .vercel		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 app		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 bootstrap		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 config		2022-03-27 10:19:00	drwxr-xr-x
<input type="checkbox"/>	 database		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 public		2022-03-27 10:42:00	drwxrwxrwx
<input type="checkbox"/>	 public_html		2022-03-27 10:42:00	drwxrwxrwx
<input type="checkbox"/>	 resources		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 routes		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 storage		2022-03-27 08:55:00	drwxr-xr-x
<input type="checkbox"/>	 tests		2022-03-27 08:55:00	drwxr-xr-x

Fig. 14: Eliminar el archivo *default.php*

Volver a la terminal y ejecutar *composer install*, para poder instalar todas las dependencias del proyecto, como se observa en la **Fig. 15**.

```
OpenSSH SSH client
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\gabuc>ssh -p 65002 u221368551@212.1.208.101
The authenticity of host '[212.1.208.101]:65002 ([212.1.208.101]:65002)' can't be established.
ECDSA key fingerprint is SHA256:WWQoM86kollMyf6gqpDSMgoMEs2Qw0PwLPQIs8rOXTA.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '[212.1.208.101]:65002' (ECDSA) to the list of known hosts.
u221368551@212.1.208.101's password:

      HHHH      HHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHHHHHHHHHHHHH  HHHHHH
      HHHHHHHHHHHHHHHHHHHHHHHHHHH
      HHHHHHHHHHHHHHHHHHHHHHHHHHH
      HHHH  HHHHHHHHHHHHHHHHHHHHHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHH      HHHH

Welcome back! The time now is 22:56 UTC
Server load: 11.58, 12.15, 13.16

Link to hPanel:
https://hpanel.hostinger.com/

-bash-4.2$ composer install
```

Fig. 15: Ejecutar composer install

Ejecutar los comandos: `php artisan storage:link` como se muestra en la Fig. 16 y posteriormente ejecutar `mv storage` `/home/u221368551/domains/nicecodeuio.com/public_html/pasantias-api/public_html`, para crear y enlazar la carpeta *storage*, mostrado en la Fig. 17 donde se almacenarán las imágenes y archivos.

```
OpenSSH SSH client
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\gabuc>ssh -p 65002 u221368551@212.1.208.101
The authenticity of host '[212.1.208.101]:65002 ([212.1.208.101]:65002)' can't be established.
ECDSA key fingerprint is SHA256:WWQoM86kollMyf6gqpDSMgoMEs2Qw0PwLPQIs8rOXTA.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '[212.1.208.101]:65002' (ECDSA) to the list of known hosts.
u221368551@212.1.208.101's password:

      HHHH      HHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHHHHHHHHHHHHH  HHHHHH
      HHHHHHHHHHHHHHHHHHHHHHHHHHH
      HHHHHHHHHHHHHHHHHHHHHHHHHHH
      HHHH  HHHHHHHHHHHHHHHHHHHHHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHHHHHH  HHHHHHHH
      HHHH      HHHH

Welcome back! The time now is 22:56 UTC
Server load: 11.58, 12.15, 13.16

Link to hPanel:
https://hpanel.hostinger.com/

-bash-4.2$ php artisan storage:link
```

Fig. 16: Ejecutar comando `php artisan storage:link`

```
-bash-4.2$ mv storage /home/u221368551/domains/nicecodeuio.com/public_html/pasantias-api/public_html mv storage /home/u221368551/domains/nicecodeuio.com/public_html/pasantias-api/public_html mv storage /home/u221368551/domains/nicecodeuio.com/public_html/pasantias-api/public_html
```

Fig. 17: Crear y enlazar la carpeta storage

Ejecutar php artisan migrate --seed, para migrar la base de datos y correr los seeders, como se puede ver en la **Fig. 18**

```
-bash-4.2$ php artisan migrate --seed
```

Fig. 18: Ejecutar el comando php artisan migrate --seed

3. Despliegue del *frontend* en *Firebase*.

A continuación, se detalla el procedimiento para realizar el despliegue a producción del *frontend* de la plataforma virtual en *Firebase*. Primero, ir a *firebase*, autenticarse y dar clic al botón "Ir a la consola", como se ilustra en la **Fig. 19**.

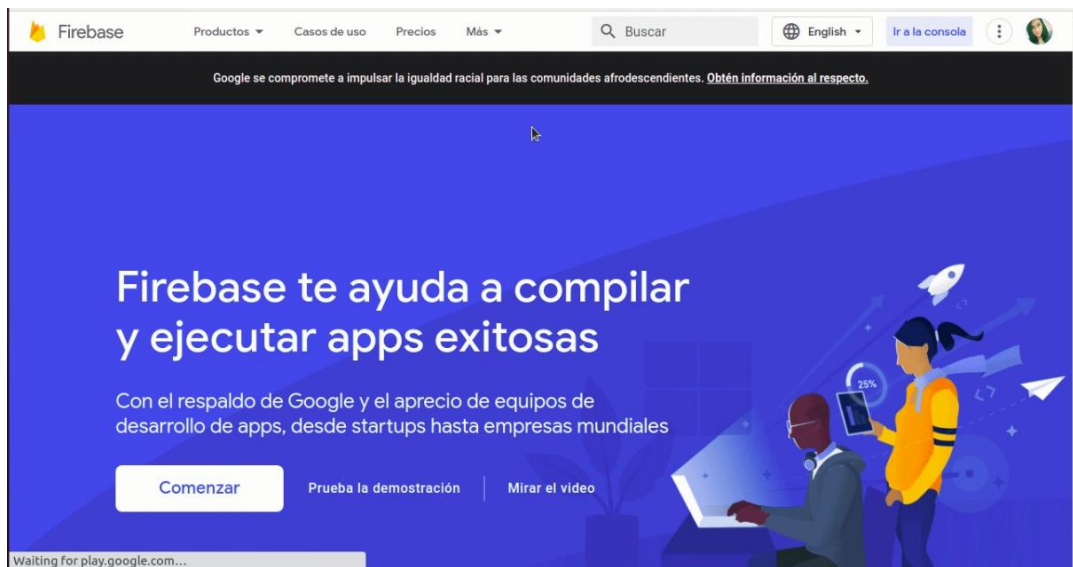


Fig. 19: Autenticación en firebase

Una vez dentro de la consola, dar clic en "Agregar proyecto", ilustrado en la **Fig. 20**.

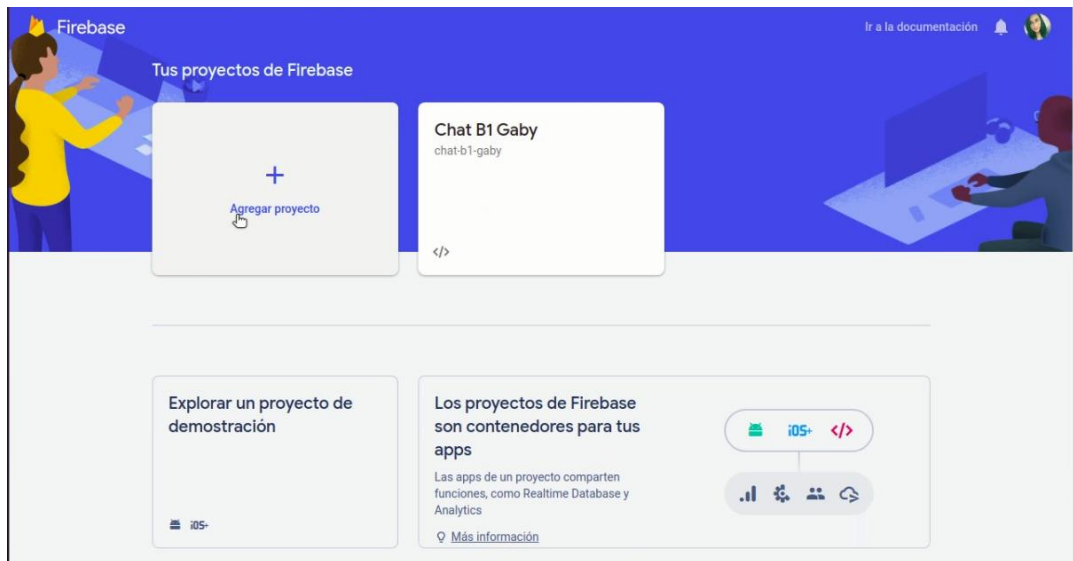


Fig. 20: Agregar proyecto

En la **Fig. 21** se observa la asignación de un nombre al proyecto y continuar con el proceso de creación.

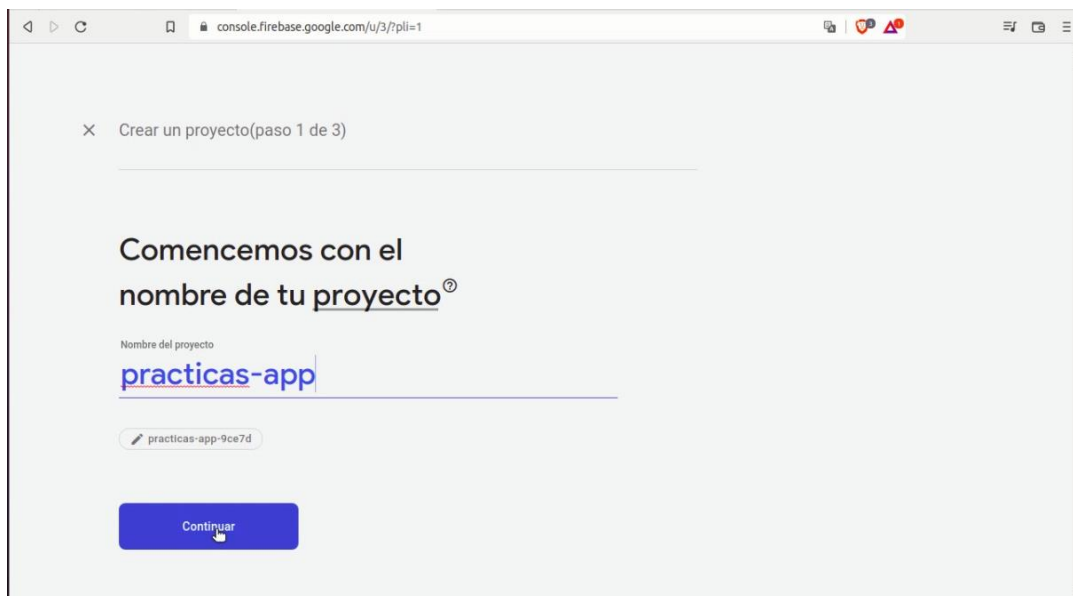


Fig. 21: Asignar nombre al proyecto.

Una vez que el proyecto haya sido creado, ingresar al mismo, ir al módulo "Hosting" y dar clic en "Comenzar". Esto se ilustra en la **Fig. 22**.

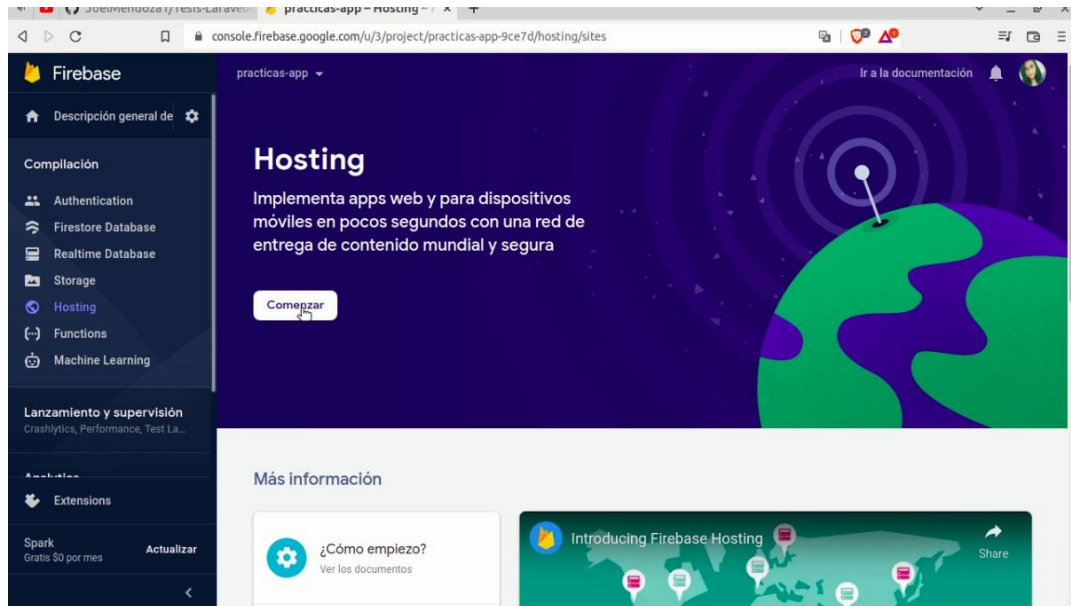


Fig. 22: Módulo Hosting

En la **Fig. 23** se copia el comando para instalar Firebase CLI

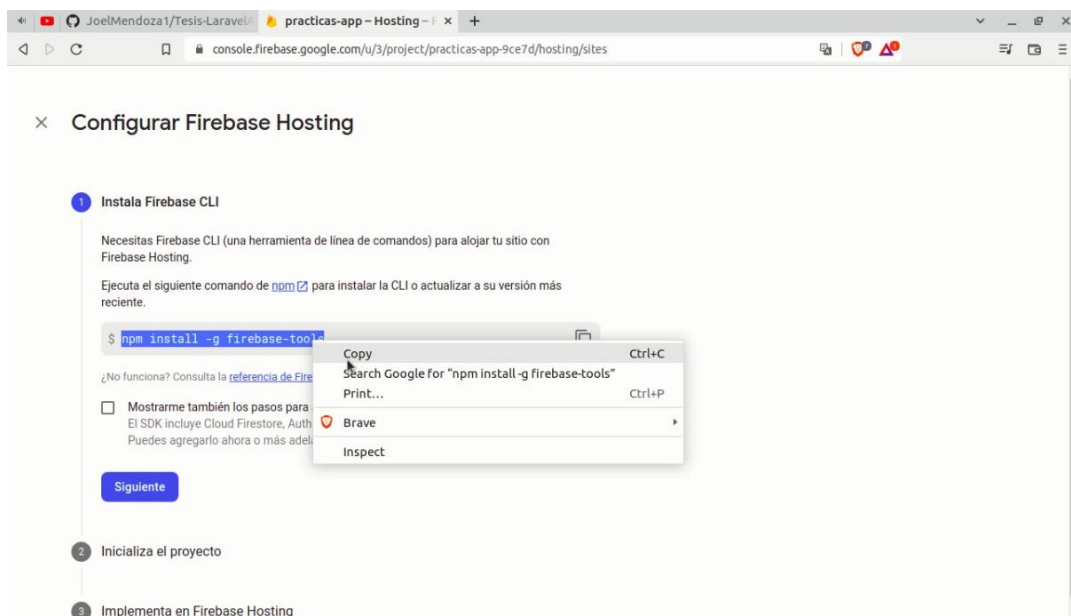


Fig. 23: Comando para instalar *firebase* CLI

Abrir una terminal en la raíz del proyecto de frontend y ejecutar el comando copiado anteriormente, esto se ilustra en la **Fig. 24**.

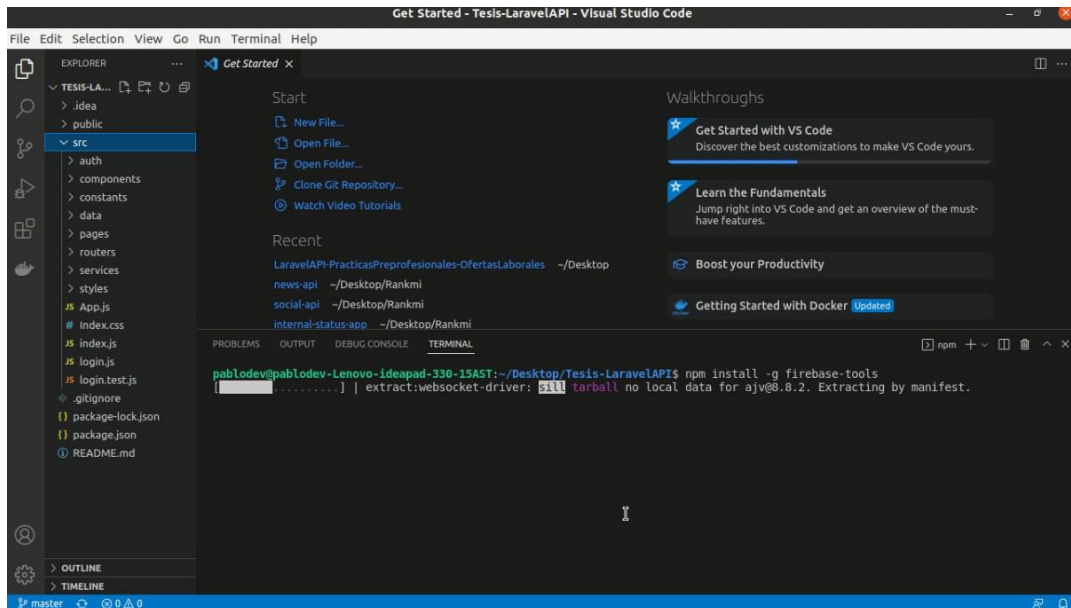


Fig. 24: Ejecutar el comando de *firebase*

Ejecutar *firebase login* y continuar con el proceso de autenticación. Esto se representa en la **Fig. 25**.

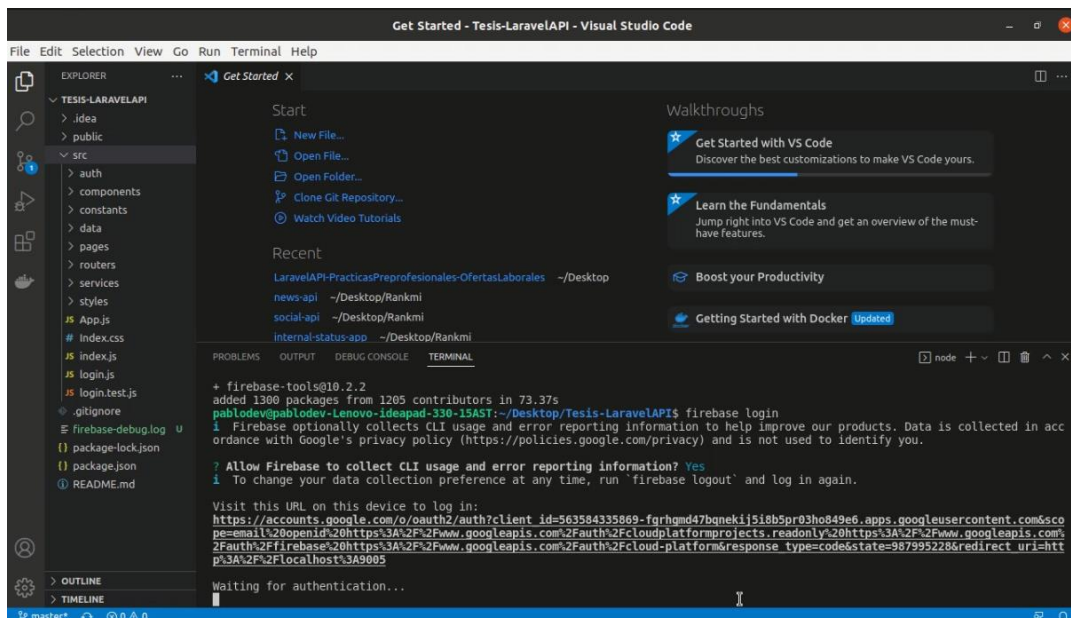


Fig. 25: *Firebase Login*

Ejecutar *firebase init* para enlazar el proyecto de *frontend* con el proyecto de *firebase*. Esto se muestra en la **Fig. 26**.

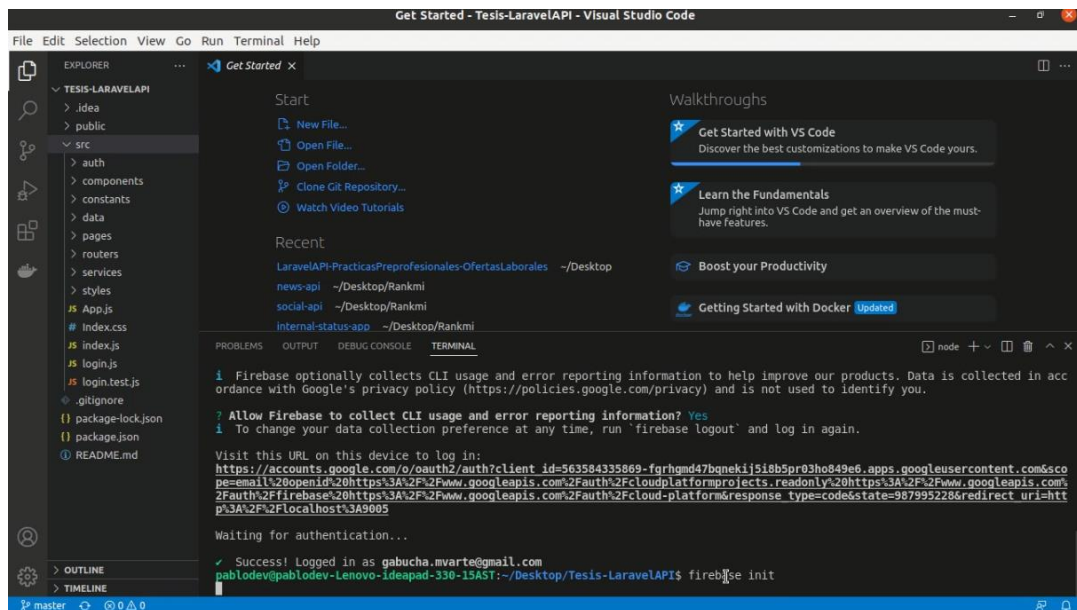


Fig. 26: Ejecutar *Firebase init*.

En la **Fig. 27** se observa que se selecciona la opción "*Hosting: Configure files for Firebase Hosting...*"

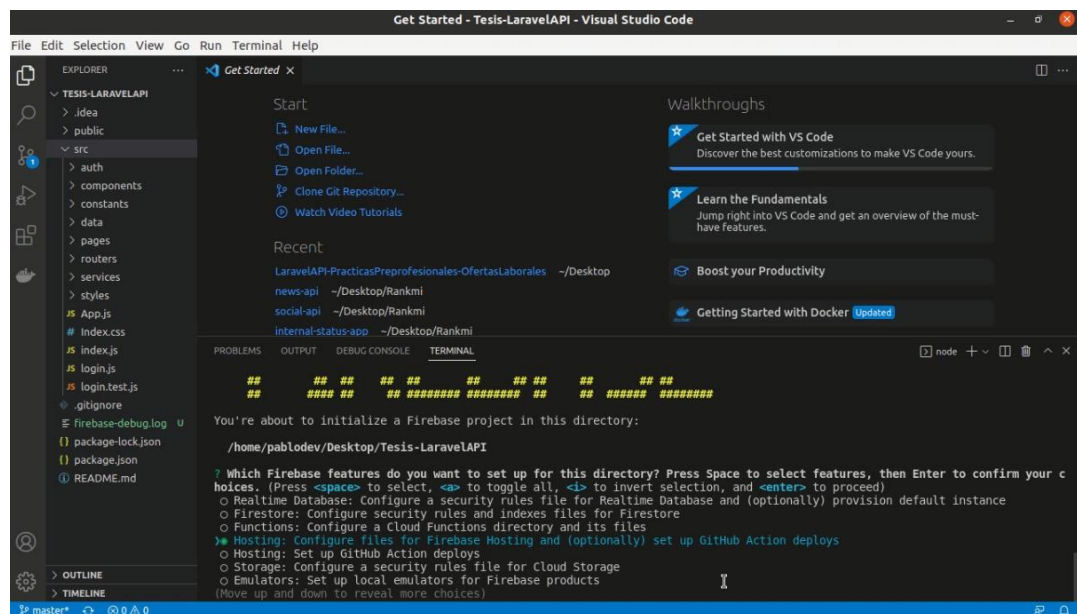


Fig. 27: Seleccionar la opción Hosting: Configure ...

Seleccionar el proyecto con el que queremos conectar, en este caso "practicas-app...". En la **Fig. 28** se observa dicho paso.

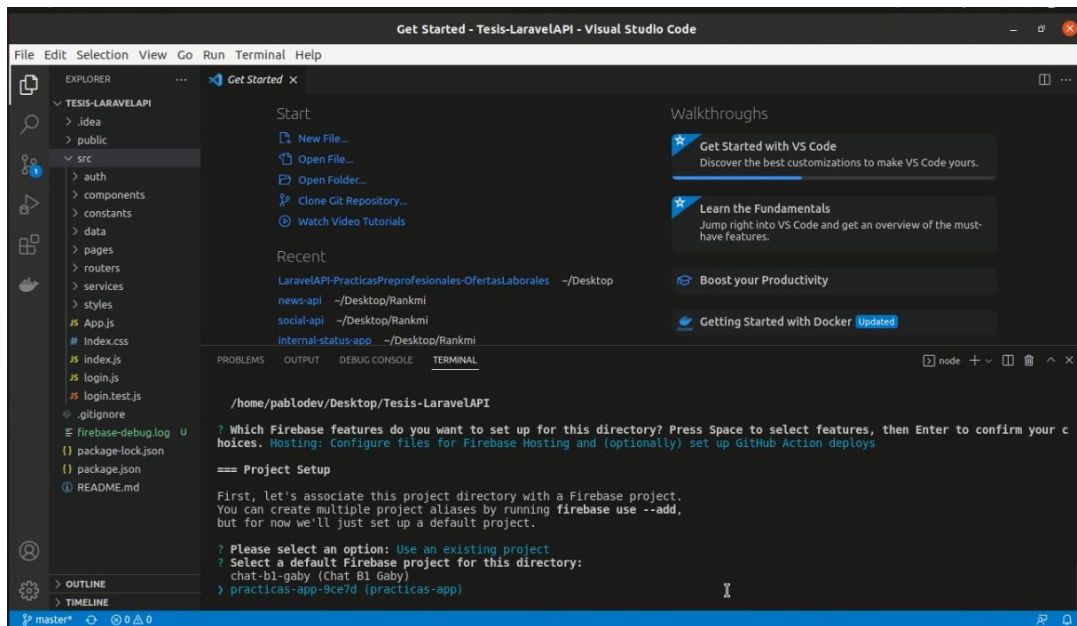


Fig. 28: Seleccionar proyecto a conectar

El resto de las configuraciones deben quedar como muestra la **Fig. 29**.

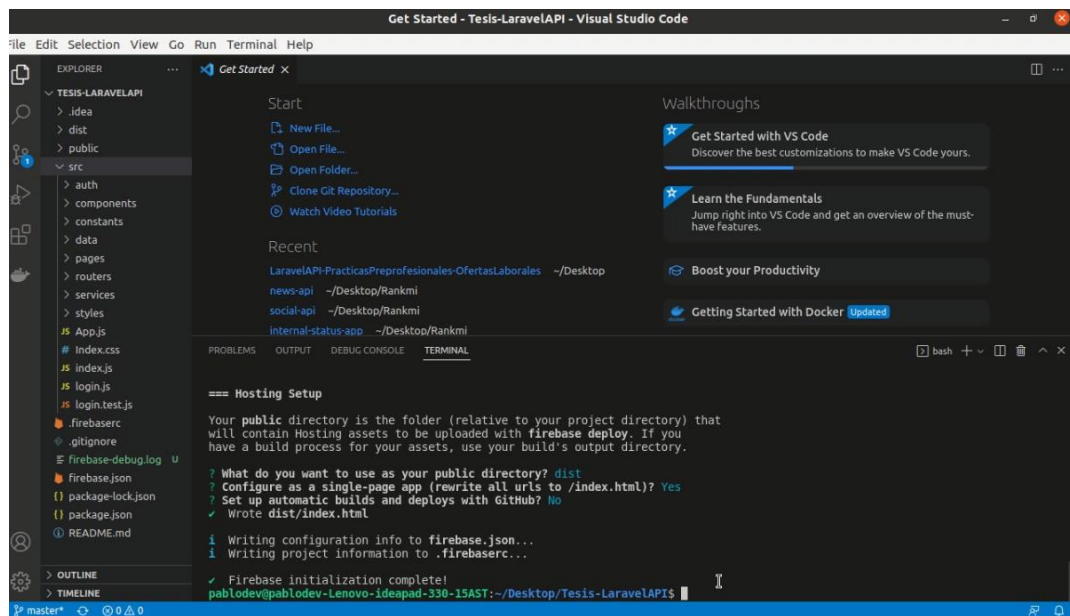


Fig. 29: Configuraciones

Después de este proceso, en el proyecto se crea una carpeta llamada *dist*, eliminar dicha carpeta como se observa en la **Fig. 30**.

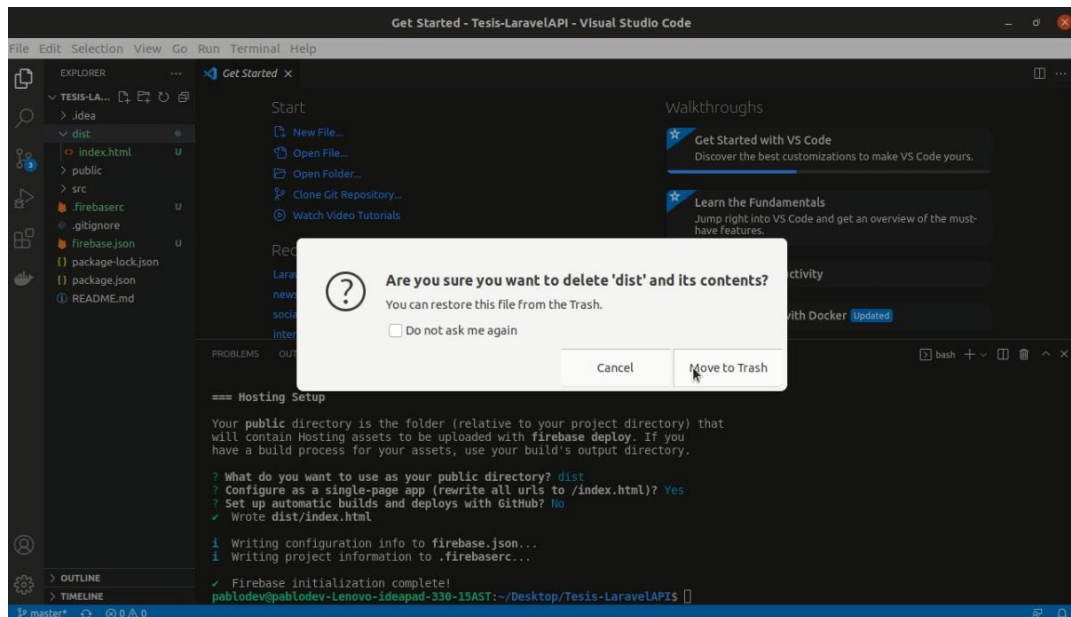


Fig. 30: Eliminar carpeta *dist*

Ejecutar el comando *npm run build* como se observa en la **Fig. 31**.

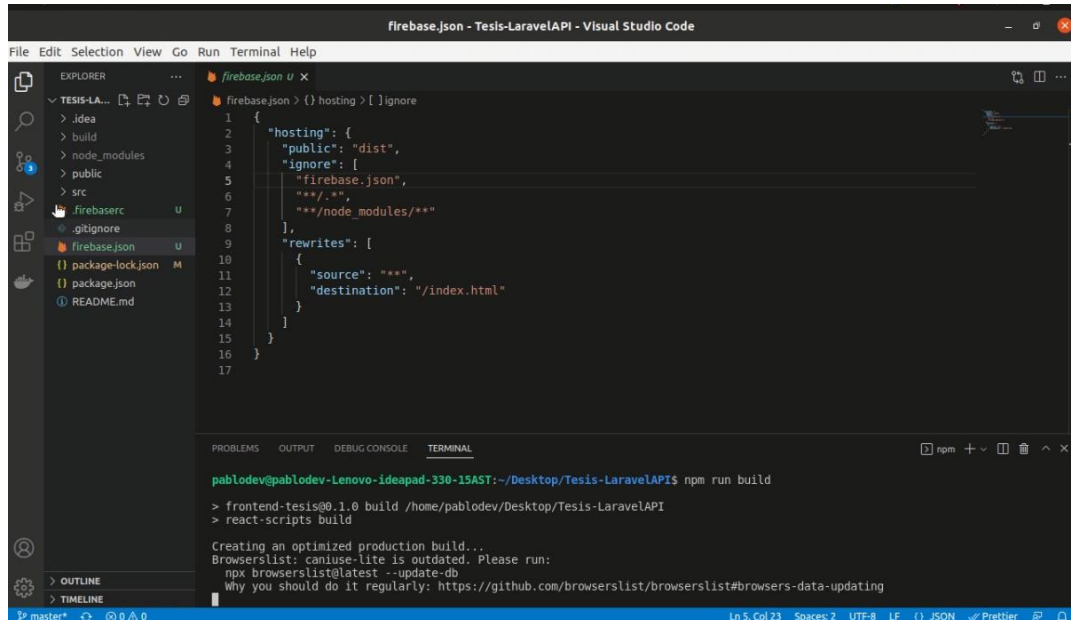
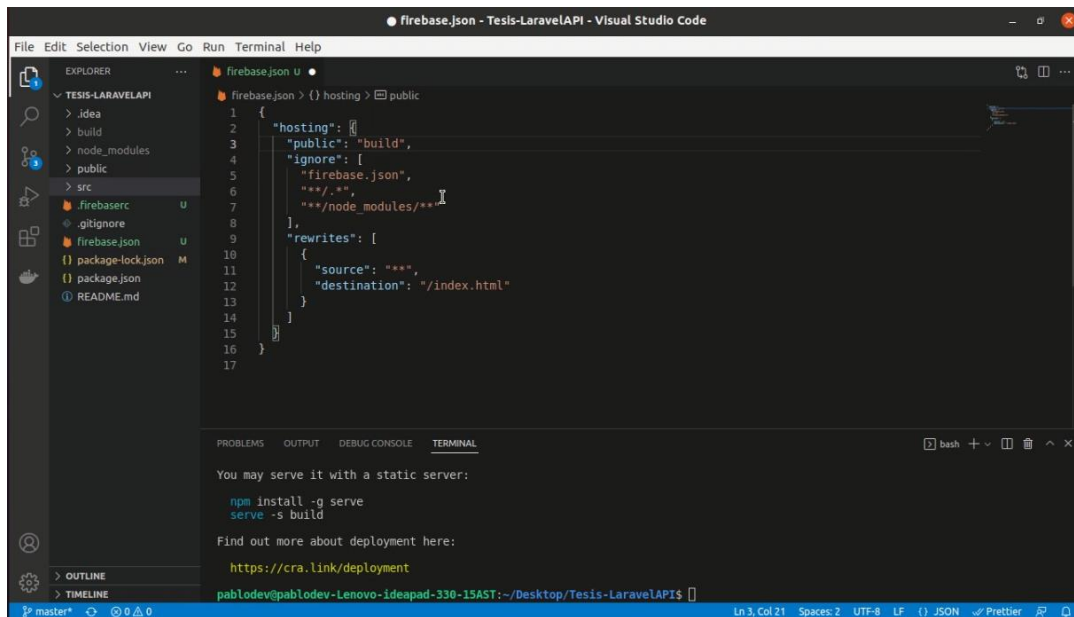


Fig. 31: Ejecutar comando *npm run build*

Modificar la key *"public"* del archivo *firebase.json*, cambiar el valor *"dist"* por *"build"*, como se muestra en la **Fig. 32**.



The screenshot shows the Visual Studio Code editor with the *firebase.json* file open. The file content is as follows:

```
1 {
2   "hosting": {
3     "public": "build",
4     "ignore": [
5       "firebase.json",
6       "**/.*",
7       "**/node_modules/**"
8     ],
9     "rewrites": [
10      {
11        "source": "**",
12        "destination": "/index.html"
13      }
14    ]
15  }
16 }
```

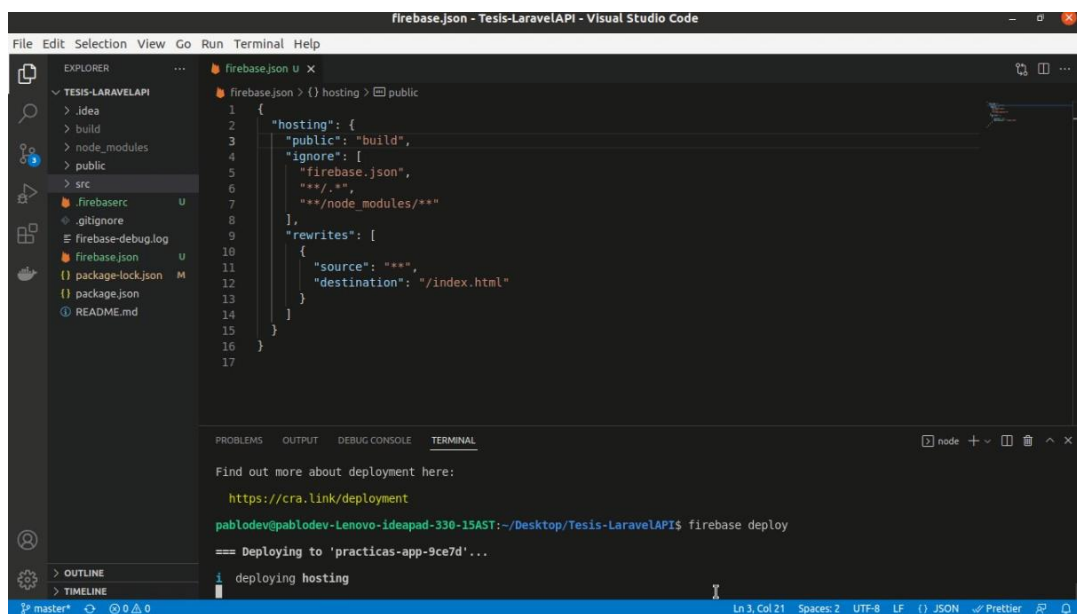
The Explorer sidebar on the left shows the project structure with *firebase.json* selected. The Terminal panel at the bottom displays instructions for serving the application with a static server:

```
You may serve it with a static server:
npm install -g serve
serve -s build

Find out more about deployment here:
https://cra.link/deployment
```

Fig. 32: Modificar la key *"public"*

Ejecutar el comando *firebase deploy* como se muestra en la **Fig. 33** y en la **Fig. 34** se observa la url retornada.



The screenshot shows the Visual Studio Code editor with the *firebase.json* file open. The file content is the same as in Fig. 32. The Terminal panel at the bottom shows the execution of the *firebase deploy* command:

```
Find out more about deployment here:
https://cra.link/deployment

pablodev@pablodev-Lenovo-ideapad-330-15AST:~/Desktop/Tesis-LaravelAPI$ firebase deploy
=== Deploying to 'practicas-app-9ce7d'...

i deploying hosting
```

Fig. 33: Ejecutar comando *firebase deploy*

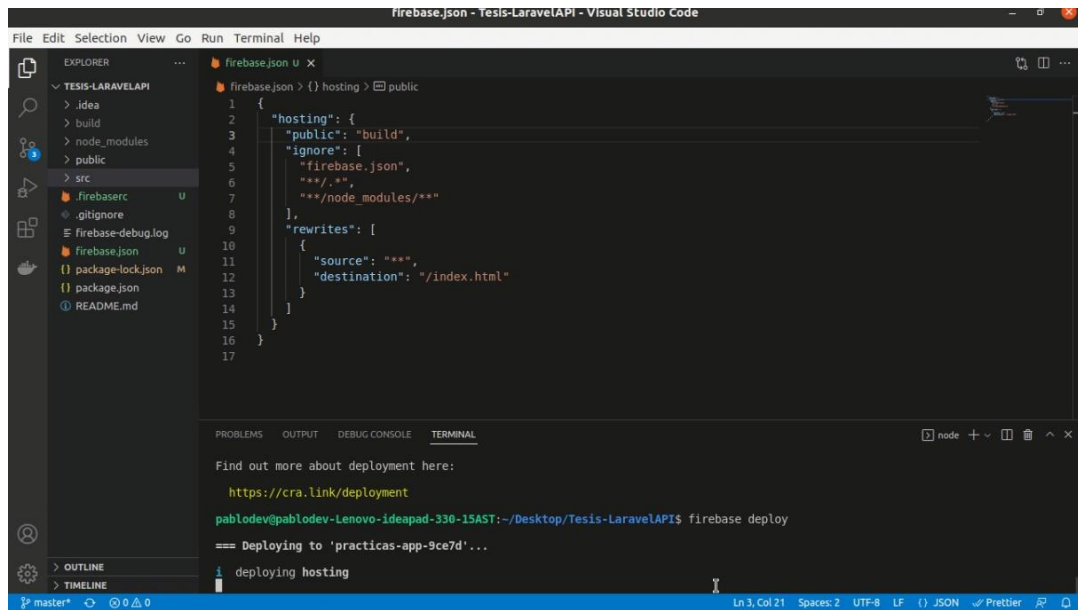


Fig. 34: URL de retorno

Con la URL de retorno visitar la plataforma virtual, como se muestra en la **Fig. 35**



Fig. 35: Plataforma Virtual

4. Credenciales de acceso para la plataforma virtual y la API

Para acceder a la plataforma virtual en producción, ingresar a la siguiente URL:

<https://practicass-app-9ce7d.web.app/>

https://nicecodeuio.com/pasantias-api/public_html/api

Para acceder al API en producción, ingresar a la siguiente URL:

https://nicecodeuio.com/pasantias-api/public_html/api

Credenciales para el perfil administrador:

Correo del usuario: admin@prueba.com

Contraseña: 123123

Credenciales para el perfil empresa:

Correo del usuario: erin.bauch@legros.com

Contraseña: 123123

Credenciales para el perfil pasante:

Correo del usuario: fmertz@hotmail.com

Contraseña: 123123

5. Repositorio del código fuente de la plataforma web

El código fuente de todo el proyecto, se encuentra alojado en el repositorio GitHub, el cual se puede acceder a través de la siguiente URL:

<https://github.com/JoelMendoza1/Tesis>