

MANUAL DE INSTALACIÓN

Aplicación solicitada por Factoria F5 para la realización de los diferentes grupos del Hackaton que organizan de manera anual integrando a todos los alumnos de los diferentes bootcamp.

Este manual de instalación fue creado para ser utilizado en distribuciones Linux como Ubuntu, Linux mint.

Requerimientos básicos.

Visual Studio Code.

Python.org.

Node.js.

MySQL.

Xampp



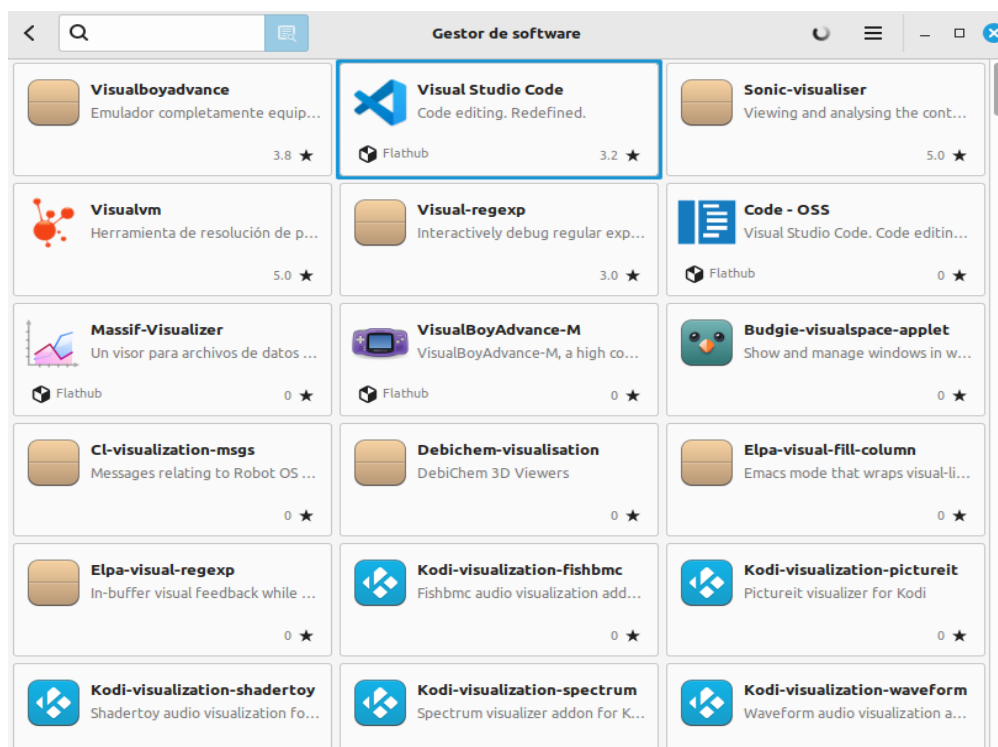
Elías Riquelme
Oussama Brahmi
Sebastián Rodríguez
Tina Calleja

Instalar Visual Studio Code.

Buscamos en el Gestor de Software en el menú INICIO.




Buscamos en el Gestor de Software VISUAL STUDIO CODE.




- \$ sudo apt-get update

```
- $ sudo apt-get upgrade
```



visual


Gestor de software



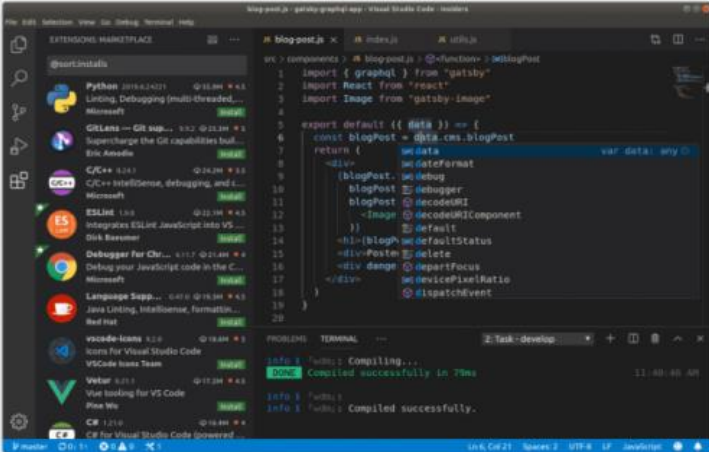
Visual Studio Code

Code editing. Redefined.

Instalando...



Flatpak (Flathub)



Detalles

Nombre:

com.visualstudio.code

Versión:

1.89.1

Tamaño:

1,4 GB para descargar, 4,1 GB de espacio de disco requerido

Rama:

stable

Remoto:

Flathub

Reseñas

★★★★☆

3.2

35 Reseñas

★★★★

12

★★★★

4

★★★

6

★★

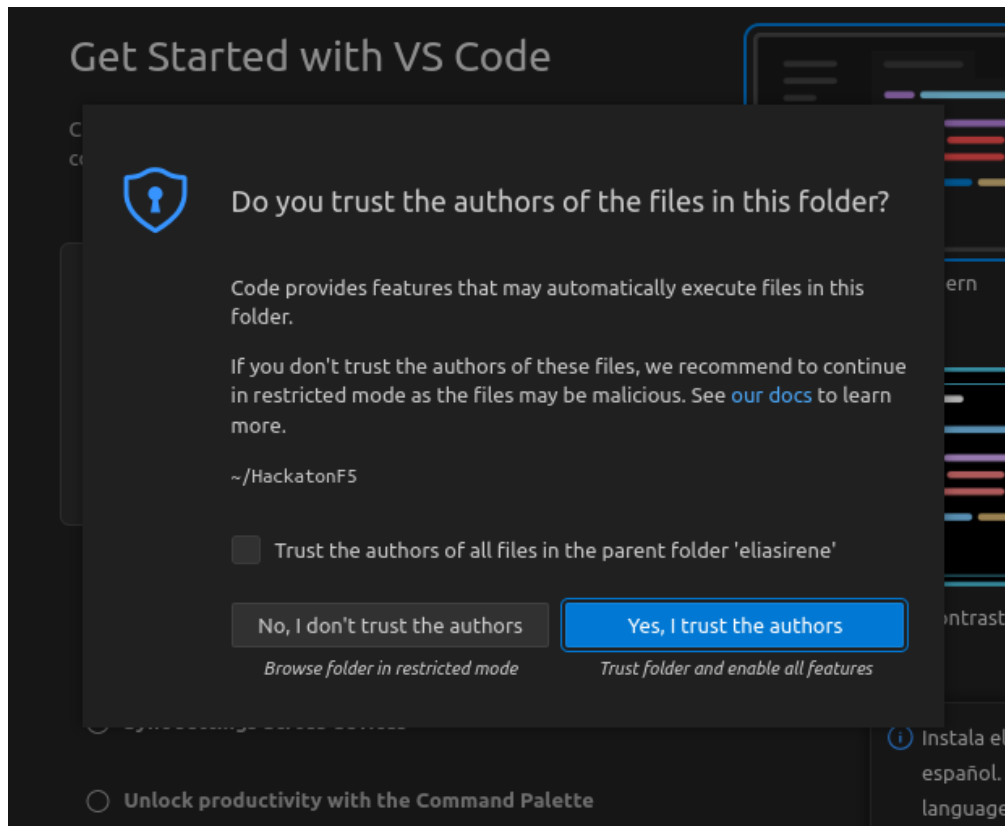
6

★

7

Abre el VSC y verás como primera opción o primera pantalla, lo siguiente:

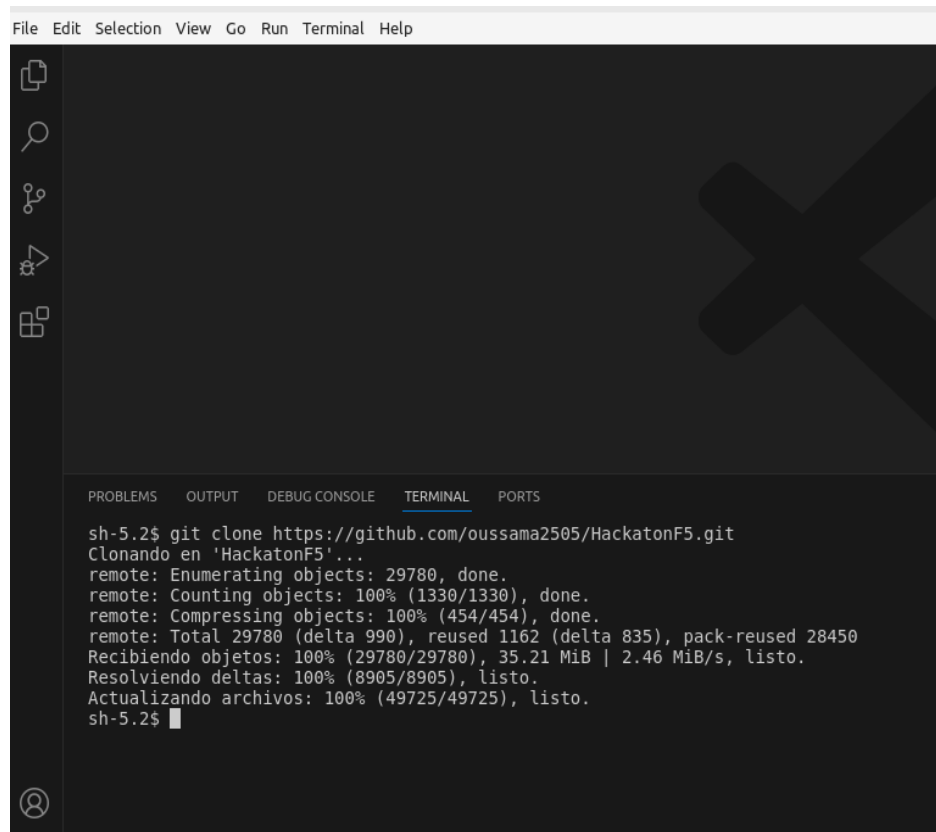
Seleccionas Si, que confías en los autores.



Inicialización del proyecto en VSC.

Se deberá clonar el repositorio con el siguiente comando en la terminal del propio VSC o desde una terminal de linux

```
git clone https://github.com/oussama2505/HackatonF5.git
```



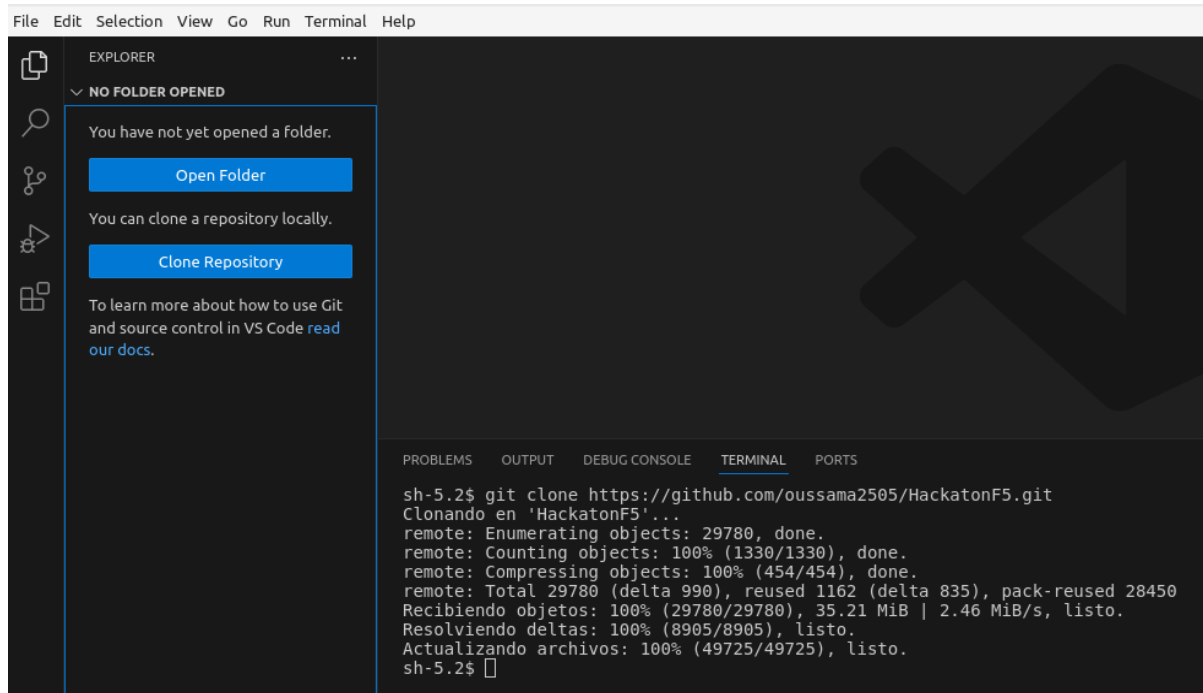
```
File Edit Selection View Go Run Terminal Help

sh-5.2$ git clone https://github.com/oussama2505/HackatonF5.git
Clonando en 'HackatonF5'...
remote: Enumerating objects: 29780, done.
remote: Counting objects: 100% (1330/1330), done.
remote: Compressing objects: 100% (454/454), done.
remote: Total 29780 (delta 990), reused 1162 (delta 835), pack-reused 28450
Recibiendo objetos: 100% (29780/29780), 35.21 MiB | 2.46 MiB/s, listo.
Resolviendo deltas: 100% (8905/8905), listo.
Actualizando archivos: 100% (49725/49725), listo.
sh-5.2$
```

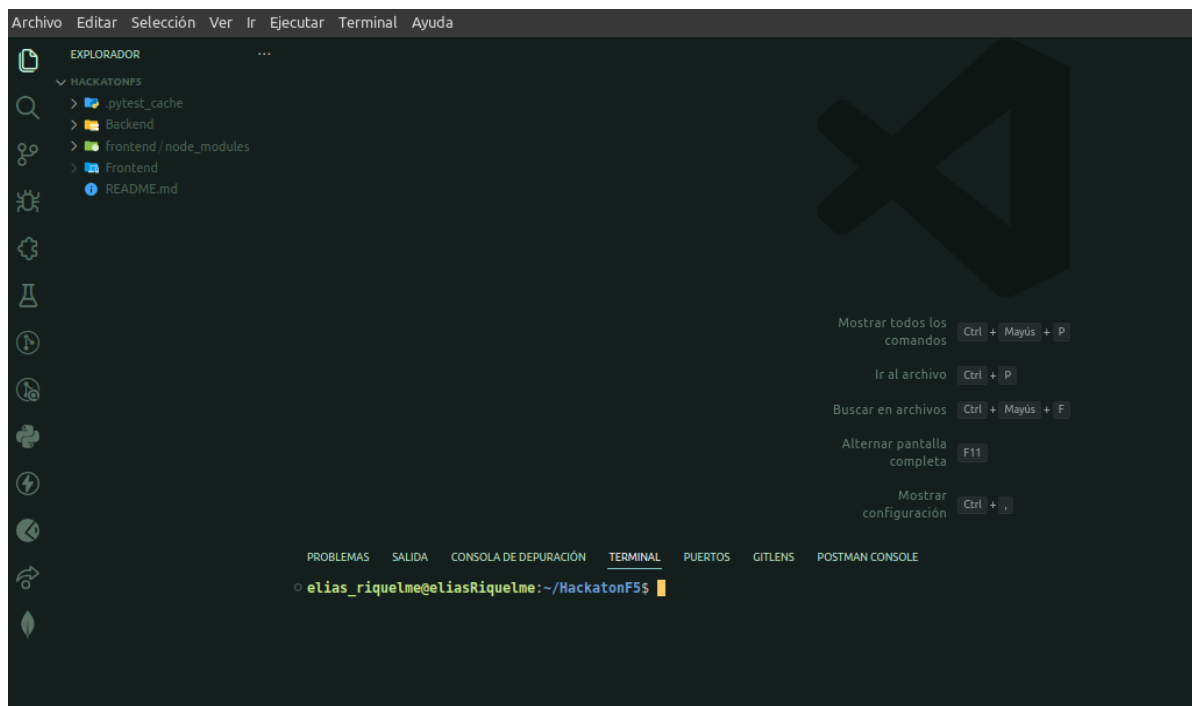
Luego de clonarlo deberás abrir el proyecto que se genera de manera automática en la raíz de tu ordenador.

Vamos a abrir la carpeta como indicamos anteriormente.
Seleccionamos open Folder y seleccionamos la carpeta que se llama /HackatonF5

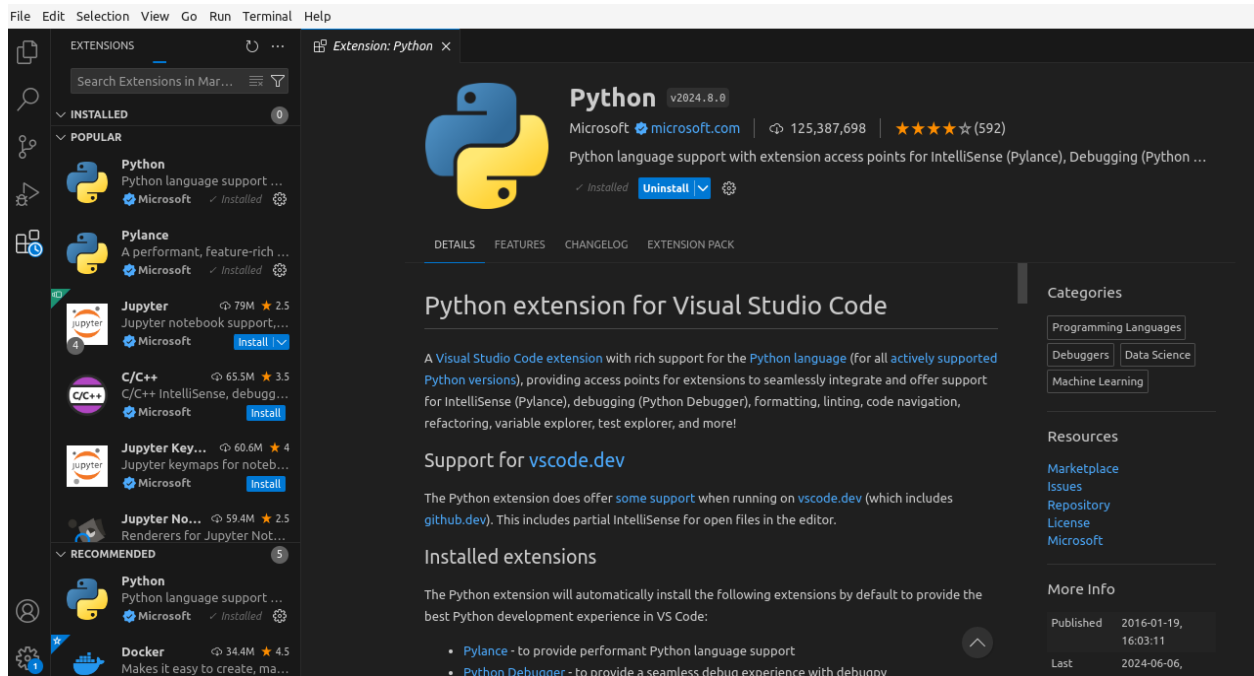
Deberías estar viendo algo así



A continuación, después de abrir la carpeta contenedora, deberíamos estar viendo la siguiente pantalla.



Debemos instalar extensiones que pueden ayudarnos en el correcto funcionamiento como la de Python a través del gestor de extensiones de VSC.



Debemos instalar un entorno virtual para instalar las librerías de Python los pasos a seguir son los siguientes

(probablemente te lo sugiera el mismo VSC)

En una terminal de linux nos ubicamos en la ruta relativa donde hemos almacenado el proyecto, por ejemplo
/home/<usuario>/HackatonF5/Backend

Debería verse así

```
Notebook-PC:~/HackatonF5$ cd Backend/  
Notebook-PC:~/HackatonF5/Backend$
```

Ejecutamos:

- \$ python3 -m venv venv

* Verificar si está instalado Virtualenv y la versión

- \$ virtualenv --version
virtualenv 20.26.1 from /usr/local/lib/python3.10/dist-packages/virtualenv/__init__.py

* Luego instalamos nuestro entorno virtual en donde este nuestro proyecto

- \$ virtualenv venv

* Ahora activamos este entorno virtual

- \$ source venv/bin/activate

Una vez tenemos instalado y activado el entorno virtual debería verse así.

```
● elias_riquelme@eliasRiquelme:~/HackatonF5$ cd Frontend/hackathon-admin/  
○ elias_riquelme@eliasRiquelme:~/HackatonF5/Frontend/hackathon-admin$
```

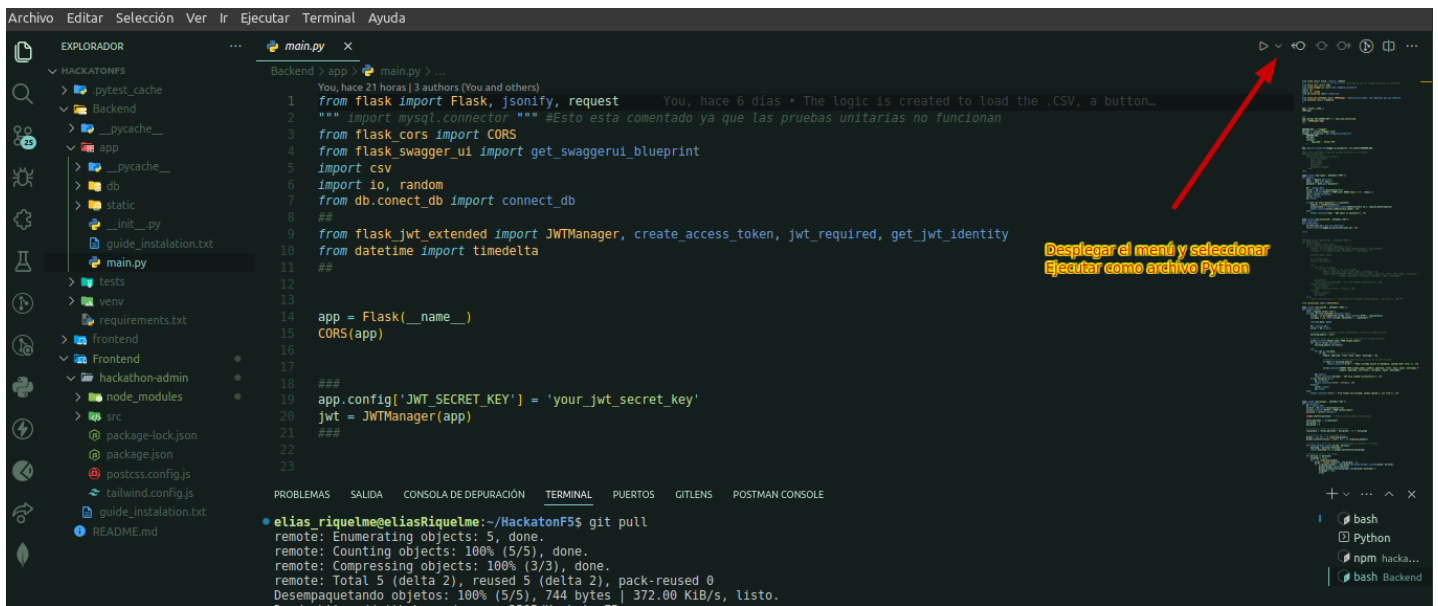
Ahora podemos instalar cada una de las librerías necesarias para que puedas ejecutar el Backend de manera correcta.

- pip install Flask
- pip install Flask-Cors
- pip install flask-swagger-ui
- pip install mysql-connector-python
- pip install Flask-JWT-Extended

Debemos ejecutar el backend para que este se inicialice.

Para esto debemos abrir el archivo main.py que se encuentra en Backend/app/main.py

y ejecutar como archivo python en el icono > situado arriba a la derecha como lo indica en la siguiente imagen.



Se debería ver como en la siguiente imagen.

Luego debemos ejecutar el servicio de Frontend con el siguiente comando en la terminal que tenemos abierta en la ruta Frontend/hackathon-admin

- \$ npm run serve

```
o elias_riquelme@eliasRiquelme:~/HackatonF5/Frontend/hackathon-admin$ npm run serve
> hackathon-admin@0.1.0 serve
> vue-cli-service serve

INFO Starting development server...
[51%] building (49/122 modules)

warn - The `purge`/`content` options have changed in Tailwind CSS v3.0.
warn - Update your configuration file to eliminate this warning.

DONE Compiled successfully in 5582ms

App running at:
- Local: http://localhost:8080/
- Network: http://192.168.0.38:8080/

Note that the development build is not optimized.
To create a production build, run npm run build.

DONE Compiled successfully in 502ms

App running at:
- Local: http://localhost:8080/
- Network: http://192.168.0.38:8080/
```

Copiar el enlace que nos ofrece en - App running at:

-Local: <http://localhost:8080/>

Ahora debemos instalar el servidor para crear la base de datos y poder conectarnos a ella.

En esta vamos a colocar el listado de alumnos que nos proporciona la organización en formato CSV y así generar la lista de equipos (este paso se realiza cuando te encuentres en la aplicación luego de finalizar este manual de instalación).

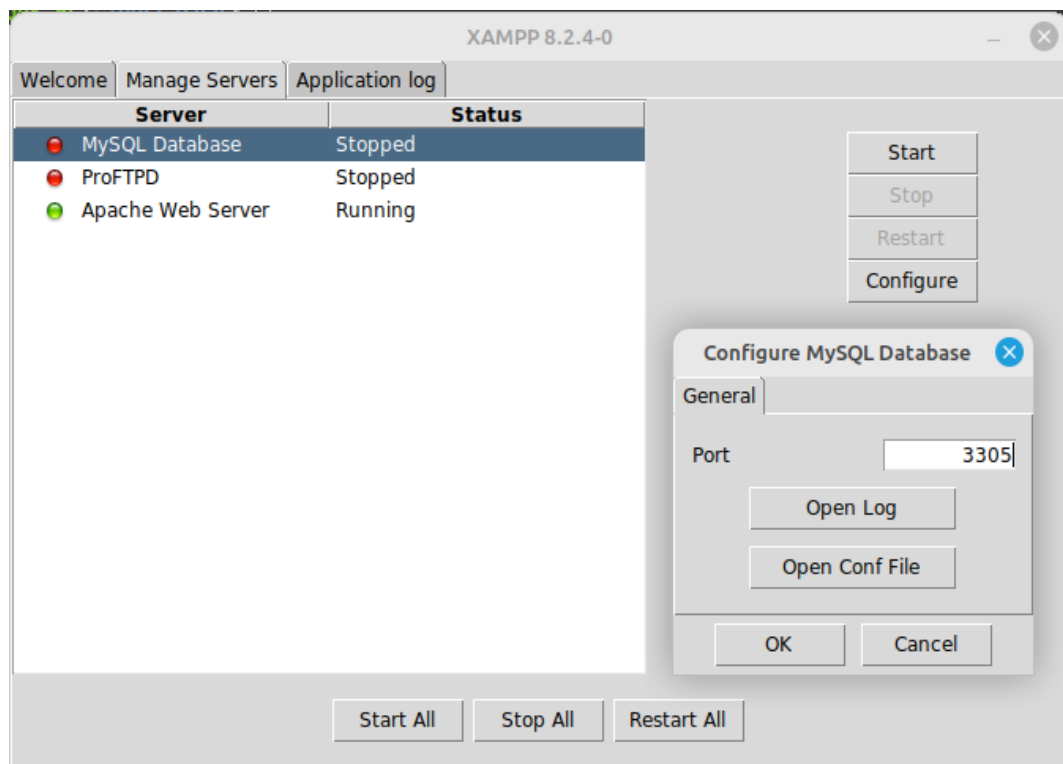
También vamos a gestionar los usuarios y contraseñas que vamos a otorgar a los usuarios que van a gestionar dicho programa.

Primero tenemos que instalar XAMPP y seguiremos estos pasos para poder instalarlo y poder ejecutarlo.

En el siguiente enlace podemos encontrar el paso a paso XAMPP para linux.

A la hora de configurar los puertos, deberá modificar como se muestra a continuación.

En MySQL Database el port debe modificarse a 3305

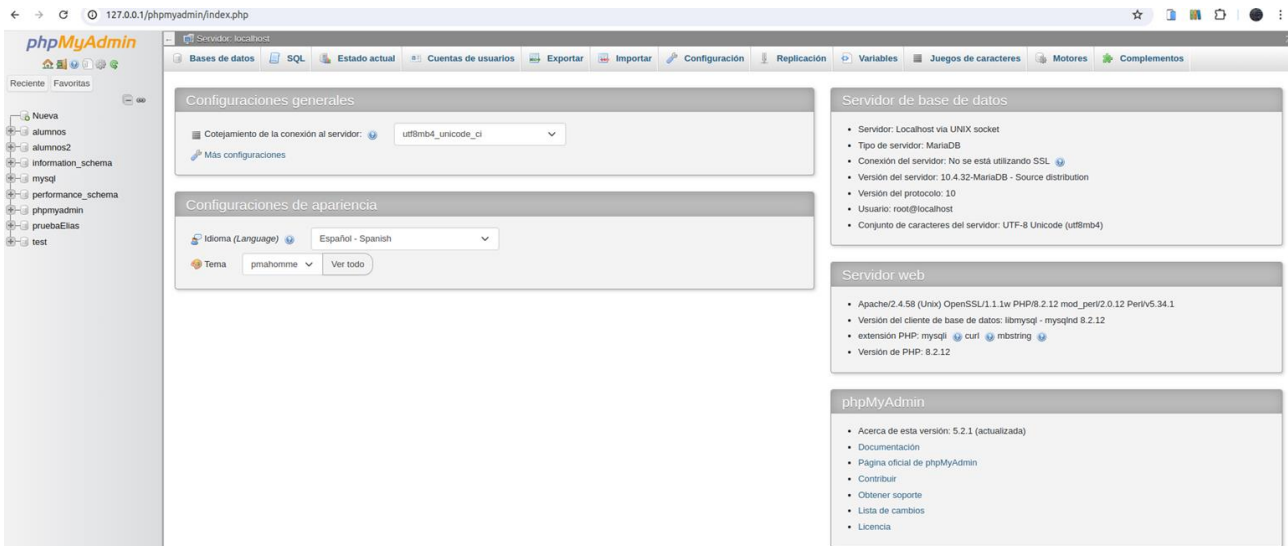


Ahora debe entrar en `http://127.0.0.1`



The screenshot shows the XAMPP dashboard for Linux 8.2.12. At the top, there's a navigation bar with links: Apache Friends, FAQs, HOW-TO Guides, PHPInfo, and phpMyAdmin. Below this is a large banner with the XAMPP logo and the text "XAMPP Apache + MariaDB + PHP + Perl". The main content area has a heading "Welcome to XAMPP for Linux 8.2.12" followed by a paragraph stating that XAMPP has been successfully installed and listing the components (Apache, MariaDB, PHP, and Perl). It also mentions that XAMPP is for development purposes and is not secure for production. A link to the XAMPP Control Panel is provided. Below this is a "Community" section with information about the XAMPP community and links to forums, Facebook, and Twitter. At the bottom, there's a footer with social media icons, a copyright notice for 2022, and a mention of the CDN provided by fastly.

Y hacer click en la pestaña phpMyAdmin del menú superior veremos la siguiente pantalla



The screenshot shows the phpMyAdmin interface. The top navigation bar includes tabs for Bases de datos, SQL, Estado actual, Cuentas de usuarios, Exportar, Importar, Configuración, Replicación, Variables, Juegos de caracteres, Motores, and Complementos. The left sidebar shows a tree view of databases, including Nueva, alumnos, alumnos2, information_schema, mysql, performance_schema, phpmyadmin, pruebaElias, and test. The main content area is divided into several sections: "Configuraciones generales" (General configurations) with a dropdown for "Cotejamiento de la conexión al servidor" (utf8mb4_unicode_ci), "Configuraciones de apariencia" (Appearance configurations) with a dropdown for "Idioma (Language)" (Español - Spanish) and a "Tema" (pmahomme) button, "Servidor de base de datos" (Database server) with details about the server (localhost via UNIX socket, MariaDB, no SSL, version 10.4.32-MariaDB, protocol 10, user root@localhost, UTF-8 Unicode (utf8mb4)), "Servidor web" (Web server) with details about the web server (Apache/2.4.58 (Unix) OpenSSL/1.1.1w PHP/8.2.12 mod_perl/2.0.12 Perl/v5.34.1, client library libmysql - mysqlnd 8.2.12, PHP extension mysql, curl, mbstring), and "phpMyAdmin" with details about the application (version 5.2.1 (actualizada), documentation, official page, contribute, support, changes, license).

Ahora debemos crear una base de datos en la que vamos a incluir dos tablas.

La base de datos se llamara alumnos, dentro de esta colocaremos la tabla 'alumno_tabla' y 'users'.

Crear una nueva tabla llamada alumno_tabla con seis columnas

```
CREATE TABLE `alumno_tabla` (  
  `nombre` text NOT NULL,  
  `apellido` text NOT NULL,  
  `front` int(11) NOT NULL,  
  `back` int(11) NOT NULL,  
  `email` text NOT NULL,  
  `bootcamp` text NOT NULL  
) ENGINE=CSV DEFAULT CHARSET=utf8 COLLATE=utf8_spanish2_ci;
```

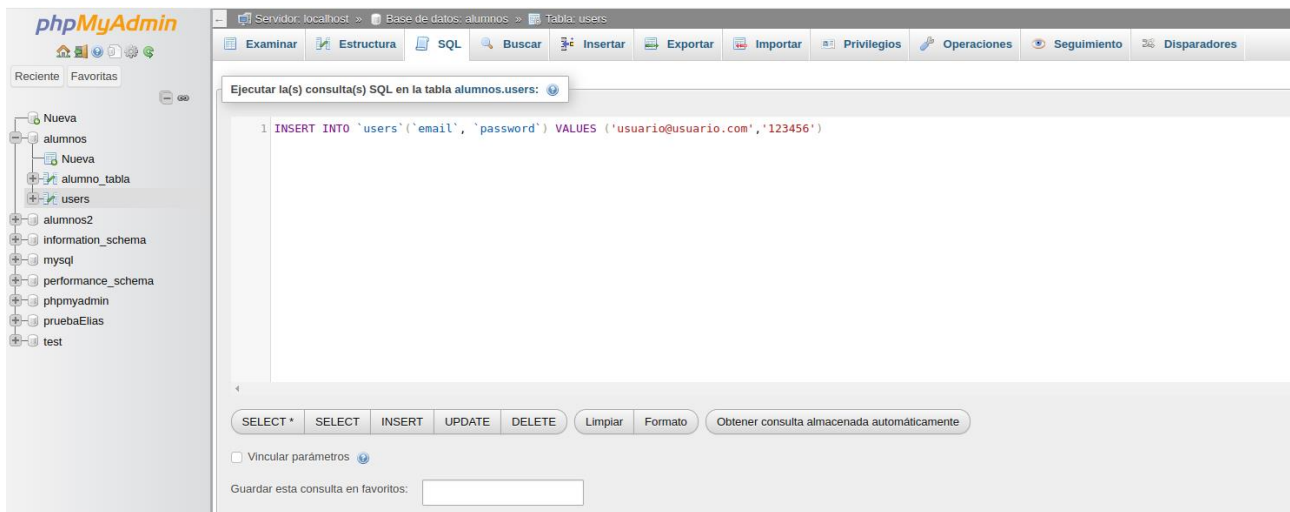
Una vez creada la tabla de alumnos podemos a través del front subir nuestro primer archivo CSV.

Vamos a crear una tabla nueva llamada users en la que vamos a almacenar los usuarios autorizados para la utilización de la app.

Haz clic en la pestaña "SQL" en la parte superior de la página, En el área de texto, pega el script SQL proporcionado más abajo.

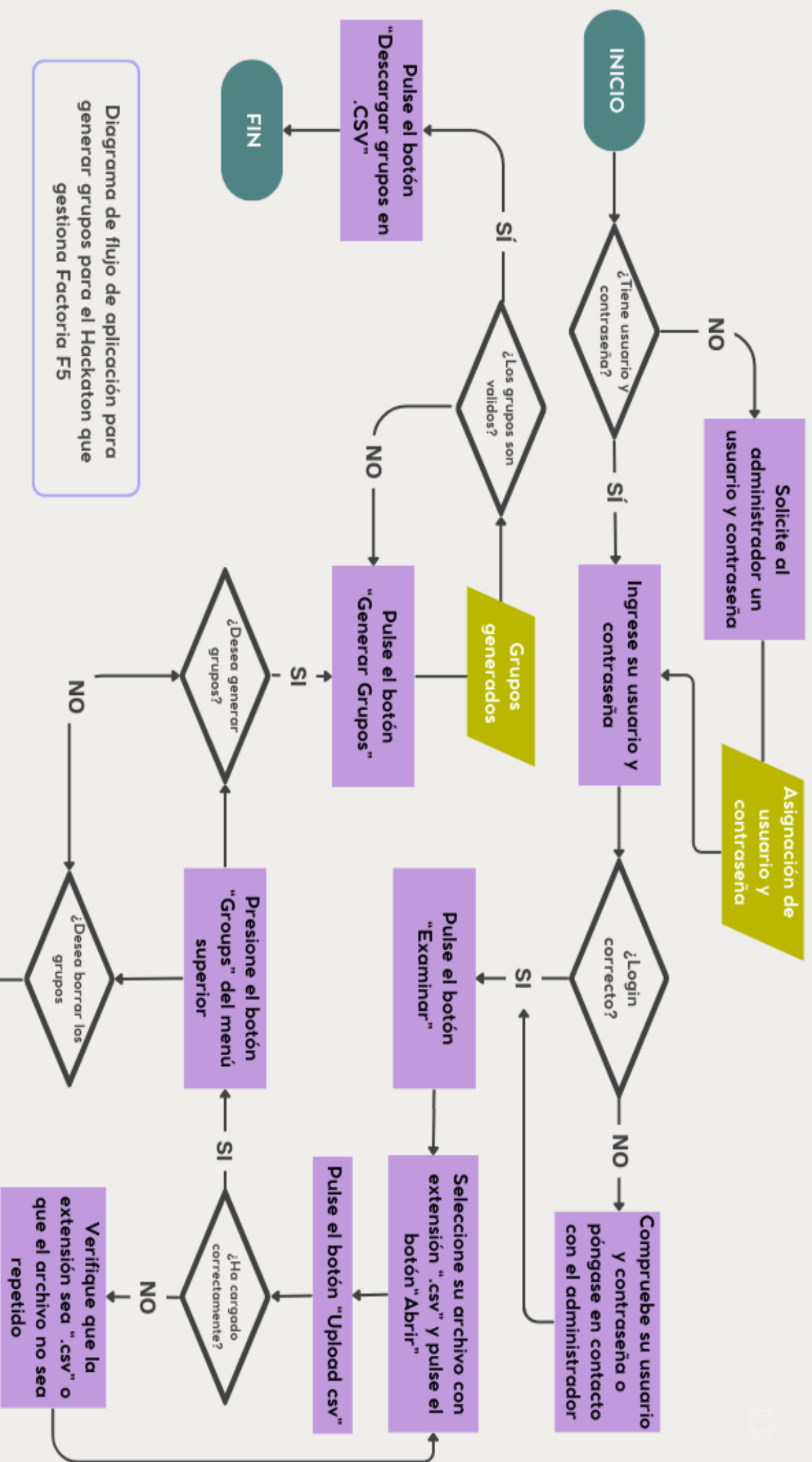
```
CREATE TABLE IF NOT EXISTS users (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  email VARCHAR(255) NOT NULL,  
  password VARCHAR(255) NOT NULL,  
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
  updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON  
UPDATE CURRENT_TIMESTAMP );
```

Para crear los usuarios que van a utilizar la aplicación, no hay más que crear los usuarios integrando el email y la contraseña como se muestra a continuación.



De esta manera deberíamos tener la estructura completa de la aplicación, puede haber varias funciones que se vean afectadas por los cambios de versiones en el sistema operativo, etc.

Para la utilización de la APP puedes seguir el diagrama de flujo adjunto en este documento.



Aplicación creada por
Eliás Riquelme
Oussama Brahmi
Sebastián Rodríguez
Tina Calleja
Alumnos del bootcamp de Peñascal F5 Bilbao