Tutorial

Back-End (Flask)

Para la ejecución del proyecto en Python:

1. Clona el repositorio en una carpeta especifica:

En mi caso lo clonaré en mi carpeta “Tutorial”, y ejecutaré el comando

“git clone <https://github.com/Digitization-Model/mobile_structure.git>”

Text

Description automatically generated

1. Instalar los requerimientos necesarios:

Necesitamos instalar los requerimientos pre escritos en requirements.txt para así ejecutar de manera correcta nuestro script app.py sin ningún error. Lo haremos de la siguiente manera.



Una vez hecho esto ya esta completo la configuración para ejecutar nuestro backend en Python.

1. Crear base de datos para Back-End:

En mi caso yo utilizaré MySQL Server para alojar los datos que manipularé (inserta, muestra, actualiza y elimina) en el backend.

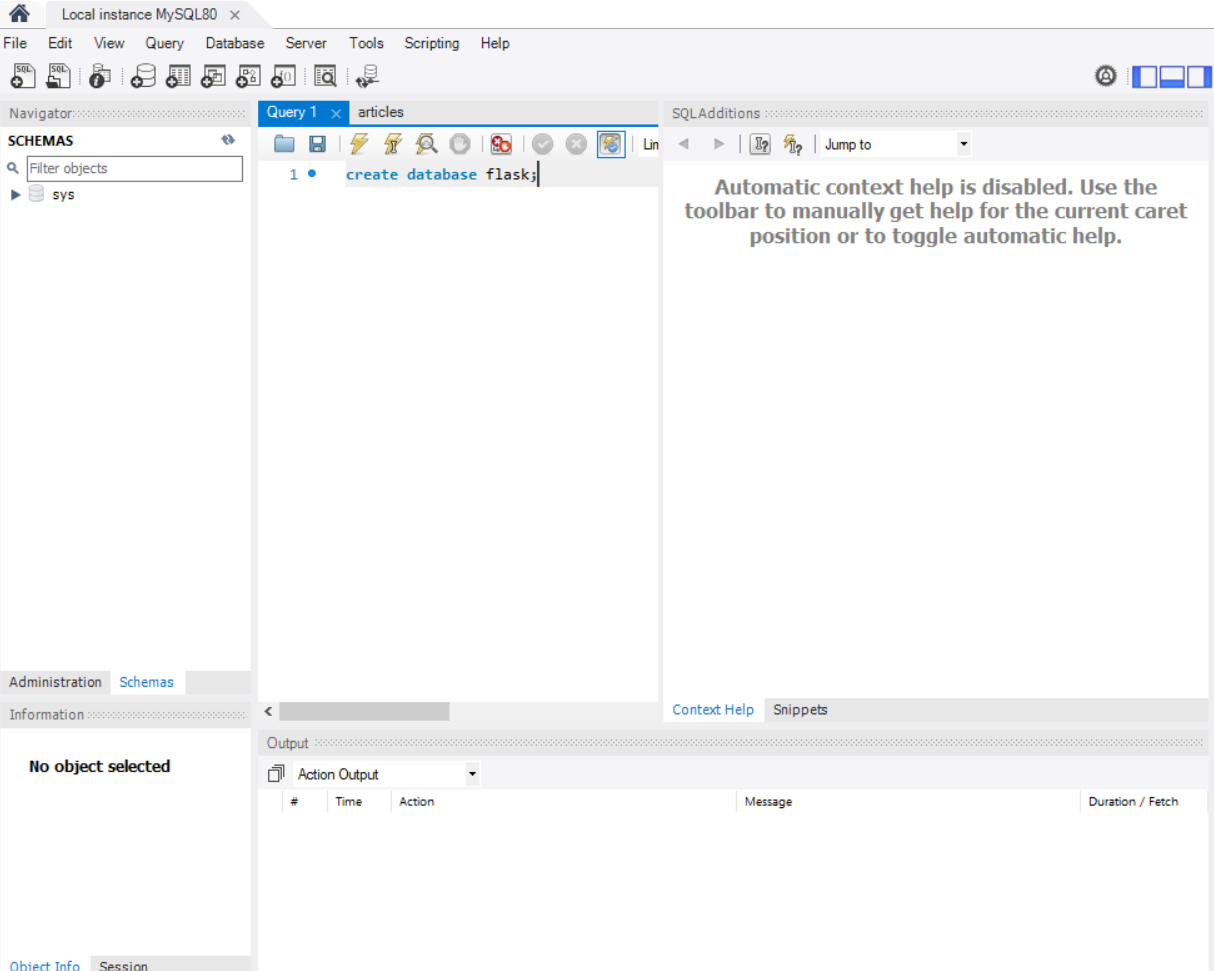
Utilizaré MySQL Workench para utilizar el entorno grafico de MySQL

Graphical user interface, text, application

Description automatically generated

Hago click en “Local Instance MySQL80” y coloco mi contraseña preconfigurada.

Se abrirá el siguiente interfaz y crearemos nuestra base de datos llamada “flask” con el siguiente comando query: “créate database flask;”



A partir de ese momento ya tendrá creada su base de datos en MySQL y su interfaz tendría que verse de la siguiente manera. Graphical user interface, text, application

Description automatically generated

Front-End (React Native)

Para la ejecución del proyecto en Java Script:

1. Clona el repositorio en una carpeta especifica:

En caso si realizó la clonación del proyecto Back-End ya cuenta con el proyecto “AppProject”, el cual contiene nuestro Front- End.

En el caso de que la carpeta “AppProject” se encuentre vacía, realice el siguiente paso.

Elija primero en donde quiere clonar el proyecto, en mi caso lo haré en la misma carpeta “Tutorial”, y ejecutaré el comando

“git clone <https://github.com/Digitization-Model/mobile_structure_AppProject.git>”

Text

Description automatically generated

1. Instalar Expo-CLI:

Para la ejecución del proyecto en React Native es necesario instalar Expo-CLI, esto lo haremos con la finalidad de tener las funciones necesarias para realizar nuestro proyecto sin poder tocar código basado en Android y iOS, en cambio React Native-CLI si nos permite hacer eso. Acabe recalcar que con proyecto Expo-CLI y React Native-CLI se pueden crear aplicaciones cross-platform (ejecución en Android e iOS).

Antes de instalar Expo-CLI es necesario que tenga Node.js para hacer uso del comando npm, el cual es necesario para la instalación de Expo-CLI, en el siguiente enlace puede descargar Node.js , se recomienda descargar la versión LTS.

[Descarga | Node.js (nodejs.org)](https://nodejs.org/es/download/)

Una vez instalado Node.js, se recomienda primero ingresar a la carpeta del proyecto desde el terminal.

A continuación, instalamos npm con el siguiente comando

“npm install”



Finalmente, en nuestro terminal ejecutamos el siguiente comando:

“npm install -g expo-cli”



Para ejecutar el proyecto ingrese el siguiente comando:

“npm start”



Verá en su terminal lo siguiente

Qr code

Description automatically generated

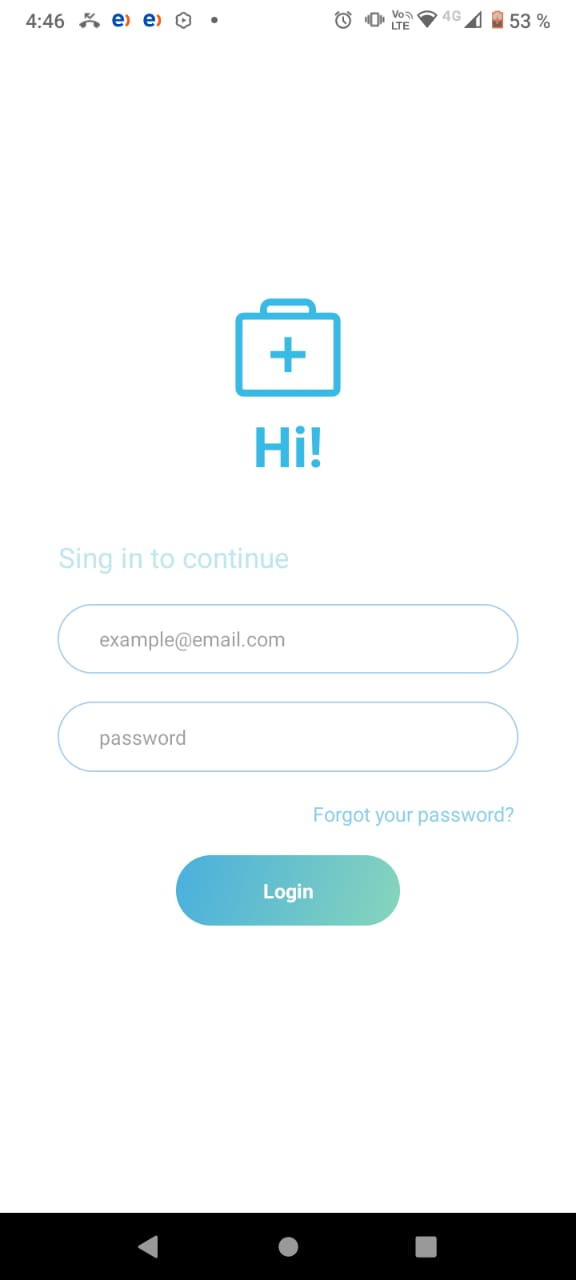
1. Visualización de nuestro proyecto en nuestro celular:

Si cuenta con un dispositivo Android o iOS puede ejecutarlo descargando desde su Marketplace la siguiente aplicación.

Graphical user interface, application

Description automatically generated

Ingresé a la aplicación y escaneé el código QR que su terminal le mostró. De esa manera ya podrá tener el proyecto ejecutando en su celular.



NOTA: Si desea ingresar al código del proyecto, es necesario decir que los únicos archivos donde se realizaron cambios fueron App.js y components/Login.js