

# INSTRUCTIVO DE IMPLEMENTACIÓN DE PROYECTO

PRESENTADO POR: TEAM TECNOGURU

JUAN ESTEBAN VANEGAS RODRIGUEZ - 20201020036

DAVID SANTIAGO PINZON MAHECHA - 20201020028

PAOLA ANDREA CUELLAR BELTRAN- 20191020119

DAVID GERARDO DIAZ GOMEZ - 20201020087

FUNDAMENTOS DE INGENIERIA DE SOFTWARE

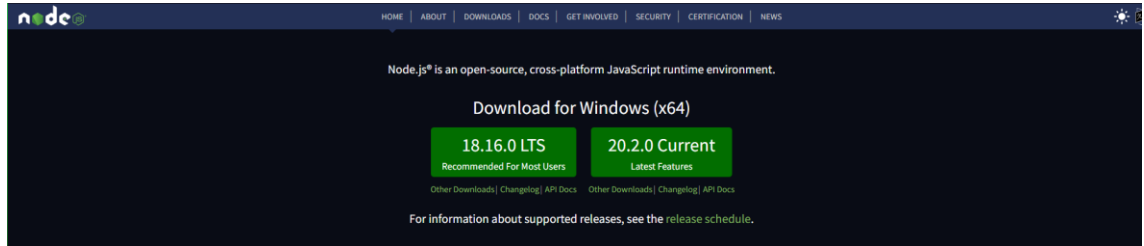
UNIVERSIDAD DISTRITAL FRANCISCO JOSE DE CALDAS

FACULTAD DE INGENIERIA

INGENIERIA DE SISTEMAS

BOGOTÁ D.C. 2023

1. Instalar node.js en la computadora:



Se debe instalar la versión 18.16.0 LTS, la cual puede ser descargada de:

<https://nodejs.org/en>

El archivo .exe descargado debe ser ejecutado, seguir las directrices de instalación.

2. Instalar Python en la computadora:

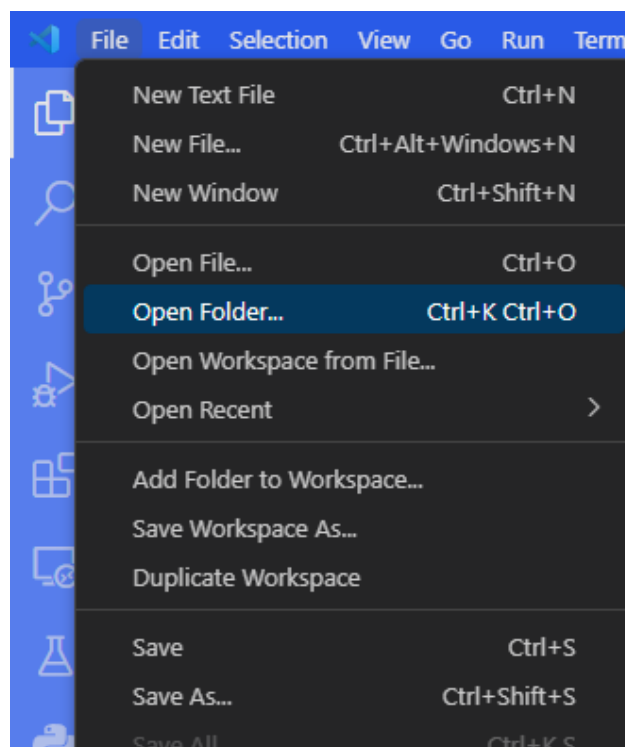


Se debe instalar la versión 3.11.3, la cual puede ser descargada de:

<https://www.python.org/downloads/>

El archivo .exe descargado debe ser ejecutado, seguir las directrices de instalación.

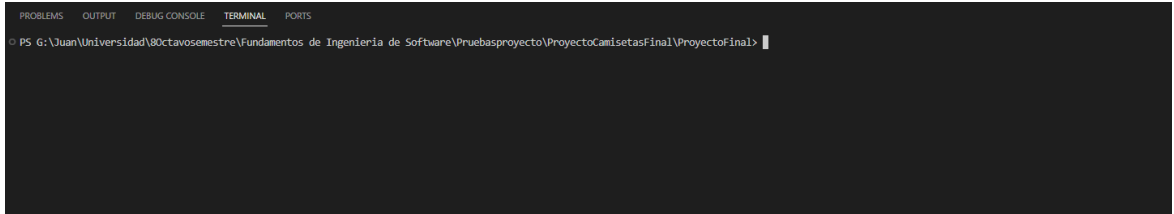
3. Descomprimir el archivo ZIP
4. En el IDE Visual Studio Code, ir a la sección de File que se encuentra en la parte superior izquierda, y seleccionar Open Folder.



5. Seleccionar la carpeta "ProyectoFinal" (dentro de la carpeta "ProyectoCamisetasFinal").



6. Una vez abierto el proyecto, abrir en la sección de Terminal una consola Command Prompt, la cual una vez abierta se vera como en la imagen:



7. En esta consola, instalaremos el sistema de gestión de paquetes de Node.js, con el comando npm install.

```
npm install
```

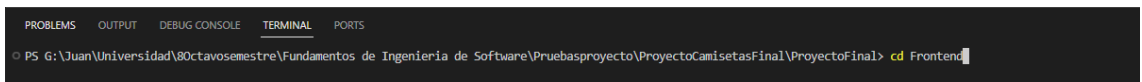
8. Siguiendo a esto, instalaremos lo referente a angular de manera global (npm install -g @angular/cli):

```
npm install -g @angular/cli
```

9. Una vez instalado lo anterior, procederemos a instalar fast api de Python Microsoft (py -m pip install "fastapi[all]"):

```
py -m pip install "fastapi[all]"
```

10. A continuación, para las dependencias se debe hacer un cd hacia la dirección Frontend:



11. Una vez redireccionado el cmd, se instalan dependencias, es decir, las librerías de angular:

```
ng add @ng-bootstrap/ng-bootstrap
```

Se instalan las librerías de estilos Bootstrap.

```
npm i pdfmake
```

Se instala la librería de pdfmaker.

```
npm i notiflix
```

Se instala la librería de notiflix.

## 12. Ejecutar la aplicación:

Abrimos una nueva Terminal (Powershell preferiblemente) y nos posicionamos en la carpeta API:

```
PS G:\Juan\Universidad\8Octavosemestre\Fundamentos de Ingenieria de Software\Pruebasproyecto\ProyectoCamisetasFinal\ProyectoFinal> cd API
```

Una vez dentro de la carpeta API, ejecutamos el comando `py -m uvicorn main:app --reload`

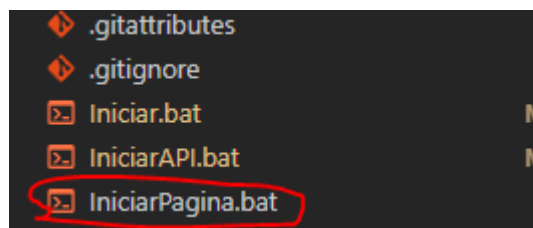
```
PS G:\Juan\Universidad\8Octavosemestre\Fundamentos de Ingenieria de Software\Pruebasproyecto\ProyectoCamisetasFinal\ProyectoFinal\API> py -m uvicorn main:app --reload
```

Si todo ha funcionado correctamente saldrá lo siguiente:

```
INFO: Will watch for changes in these directories: ['G:\Juan\Universidad\8Octavosemestre\Fundamentos de Ingenieria de Software\Pruebasproyecto\ProyectoCamisetasFinal\ProyectoFinal\API']
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO: Started reloader process [19260] using WatchFiles
INFO: Started server process [19396]
INFO: Waiting for application startup.
INFO: Application startup complete.
```

Esto quiere decir que el Backend y la conexión a la base de datos está completamente funcional.

Ahora abrimos otra Terminal nueva, en esta terminal vamos a arrastrar el archivo que dice IniciarPagina.bat



Saldrá de esta manera:

```
PS G:\Juan\Universidad\8Octavosemestre\Fundamentos de Ingenieria de Software\Pruebasproyecto\ProyectoCamisetasFinal\ProyectoFinal> & 'g:\Juan\Universidad\8Octavosemestre\Fundamentos de Ingenieria de Software\Pruebasproyecto\ProyectoCamisetasFinal\ProyectoFinal\IniciarPagina.bat'
```

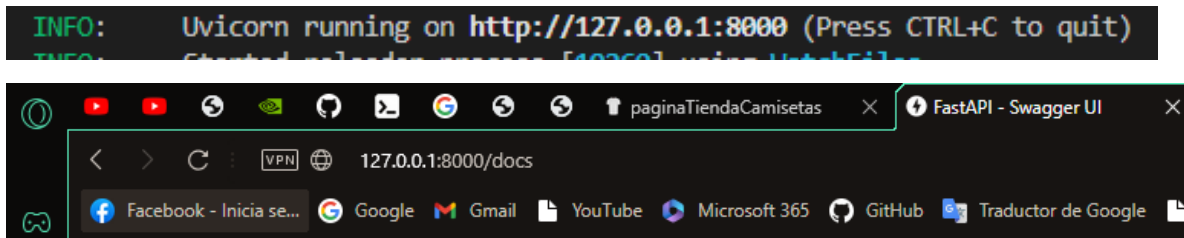
Damos enter y esperamos a que se ejecute el Frontend (se abre automáticamente en el navegador por defecto del sistema)



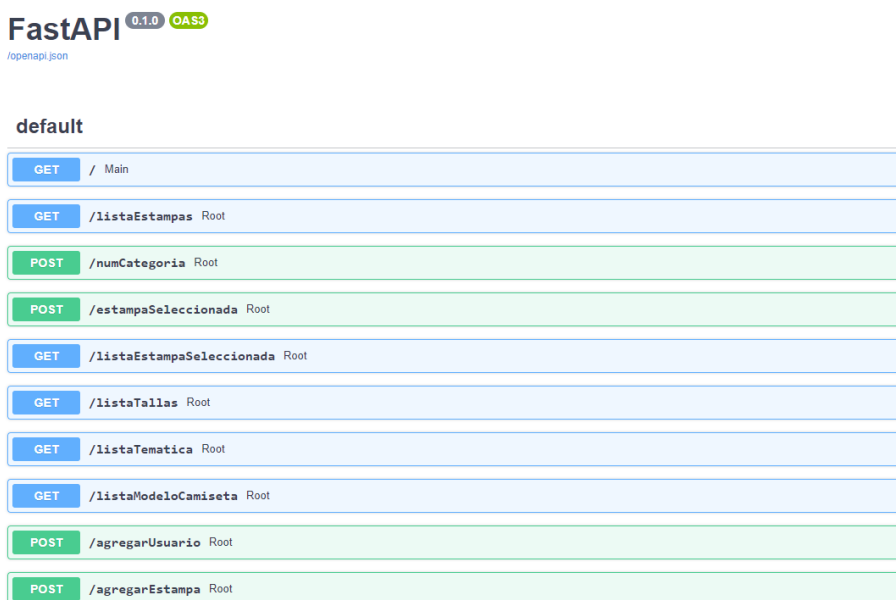
Una vez cargada la página, ya está lista para ser usada.

## INSTRUCTIVO PARA EJECUTAR PRUEBAS UNITARIAS (Por medio de FASTAPI):

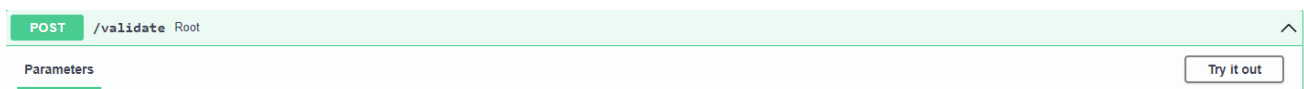
Después de ejecutar el comando `py -m uvicorn main:app --reload`, en la segunda fila INFO de esa misma terminal donde se ejecutó se puede ver un URL, lo copiamos y pegamos en el navegador, añadiendo `/docs` al final.



Se abrirá lo siguiente:



FASTAPI nos permite hacer pruebas unitarias a las funciones que se usan, para ejecutar una prueba unitaria le damos click a la función que queremos probar, y le damos al botón Try it out:



Al ser una función tipo POST, si damos en Try it Out se activa un apartado donde podemos probar con los valores que queramos la función, los valores a probar deben ser agregados en los atributos, donde dice "string", se reemplaza lo de adentro de las comillas por el dato a probar.


### Request body required


```
{  
  "correo": "string",  
  "contrasena": "string"  
}
```

Y finalmente damos al botón de Ejecutar, quien nos dará abajo la impresión (sea correcto o error) según los datos puestos.

#### NOTA:

1. No olvidar importar los dos scripts SQL en la base de datos, la cual por efectos del aplicativo se deben ejecutar en el siguiente orden:

 basetienda.sql 1

 inserciones.sql 2

2. El usuario y contraseña de la base de datos debe ser AdminTienda, además, debe encontrarse alojado el puerto de oracledb en el 1521.
3. Para las pruebas unitarias, si se desea probar una función GET, al no recibir datos si no retornarlos, no se dan datos de prueba, pero si se puede probar la recogida de datos.