



# MANUAL Integración de Mapbox Tiling Services con Power BI

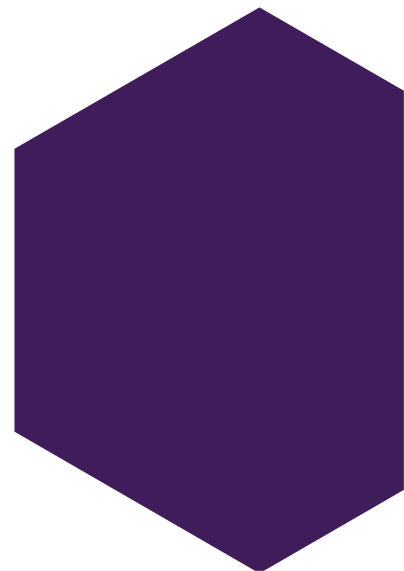
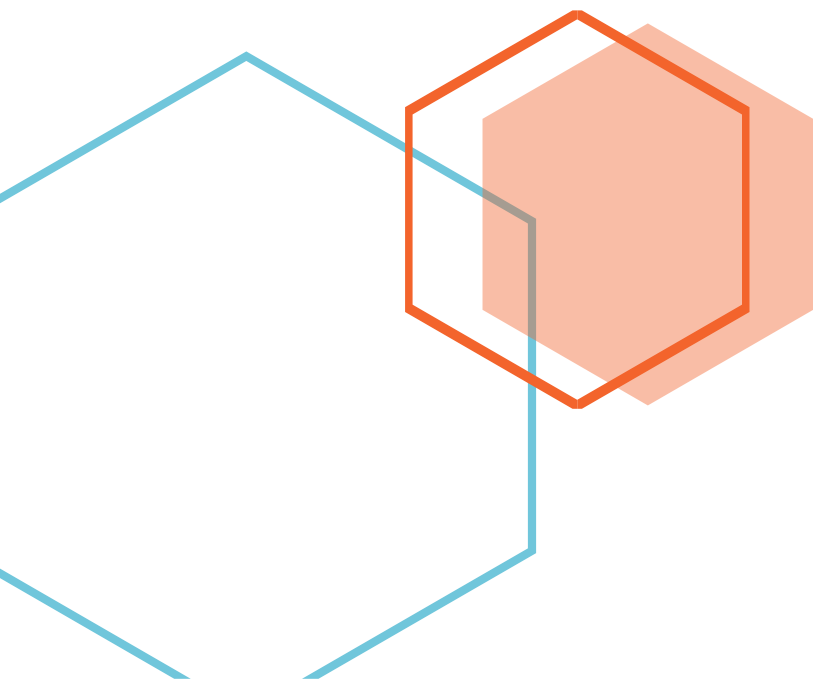
---

## UBUNTU

### HERRAMIENTAS:

Mapbox, Power BI, Máquina Virtual (VMware station)

SO: Ubuntu, Python, Mapsheper



## Contenido

1. Descarga e instalación de MVware Station. ....	3
2. Descarga e instalación del sistema operativo Ubuntu. ....	3
3. Instalación de Python en Ubuntu. ....	3
4. Creación de JSON utilizando Shapes. ....	3
5. Conversión de archivo JSON a LD. ....	5
6. Creación de Token en Mapbox. ....	6
7. Creación de Mapbox Tiling Services. ....	7
8. Modificar zoom del tileset. ....	8
Eliminar formas específicas del JSON. ....	9

## 1. Descarga e instalación de VMware Station.

- 1.1 Descargar máquina virtual presionando el siguiente enlace:  
<https://www.vmware.com/go/getplayer-win>
- 1.2 Realizar una instalación básica (dejar todos los valores por defecto). Para ejecutar correctamente, habilitar la virtualización en la computadora en caso de que no esté habilitada.

## 2. Descarga e instalación del sistema operativo Ubuntu.

- 2.1 Descargar Ubuntu presionando el siguiente enlace:  
<https://ubuntu.com/download/desktop/thank-you?version=20.04.1&architecture=amd64>
- 2.2 Instalar Ubuntu en máquina virtual. Aplicar una instalación estándar con sus valores por defecto.

## 3. Instalación de Python en Ubuntu.

- 3.1 Actualizar lista de paquetes antes de la instalación de Python pip, ejecutar la siguiente línea de código: **apt update**
- 3.2 Abrir el Terminal de Ubuntu e ingresar a siguiente línea para instalar el Python:  
**apt install python3-pip.**

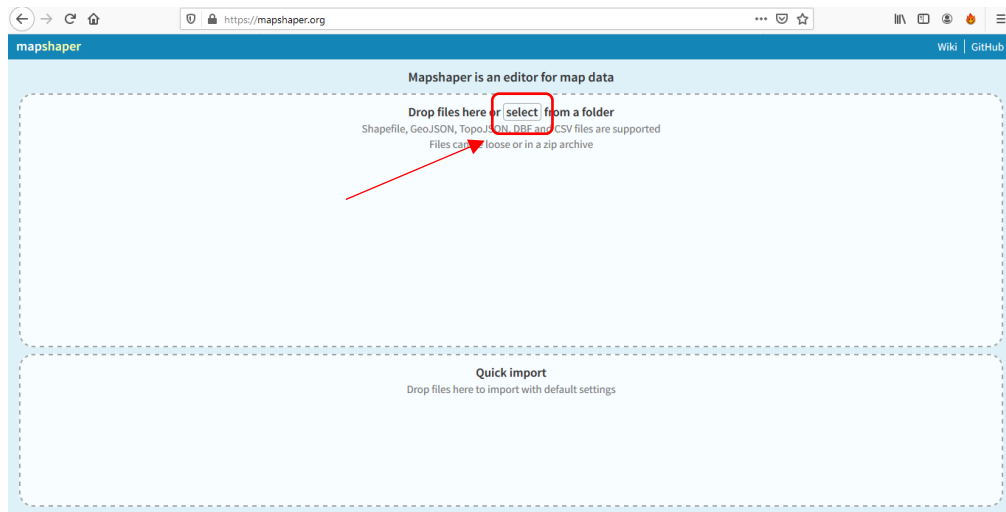
**Observación:** Si las líneas de código no se ejecutan correctamente se pueden ejecutar los comandos como administrador o como usuario normal con privilegios sudo.

## 4. Creación de JSON utilizando Shapes.

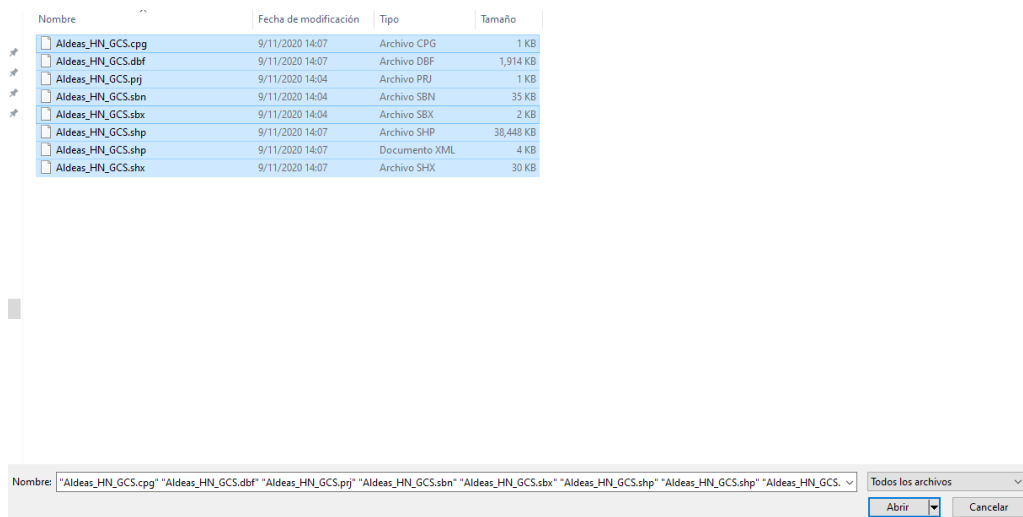
Se necesita crear un JSON para poder utilizarlo como base de Tileset y adaptar las formas (shapes) al mapa.

- 4.1 Al tener los **shapes** de la forma deseada se tienen que convertir utilizando el siguiente enlace: <https://mapshaper.org/>

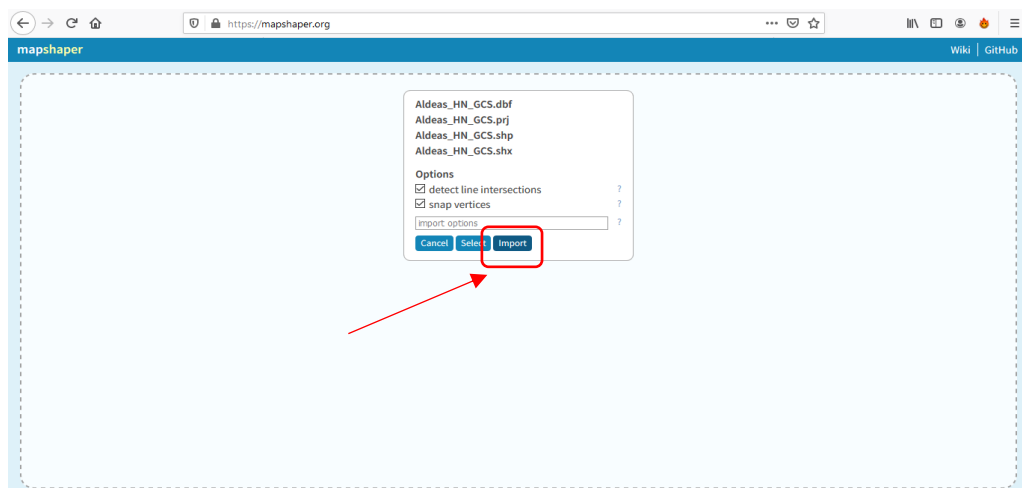
## 4.2 Hacer clic en el botón **select**.



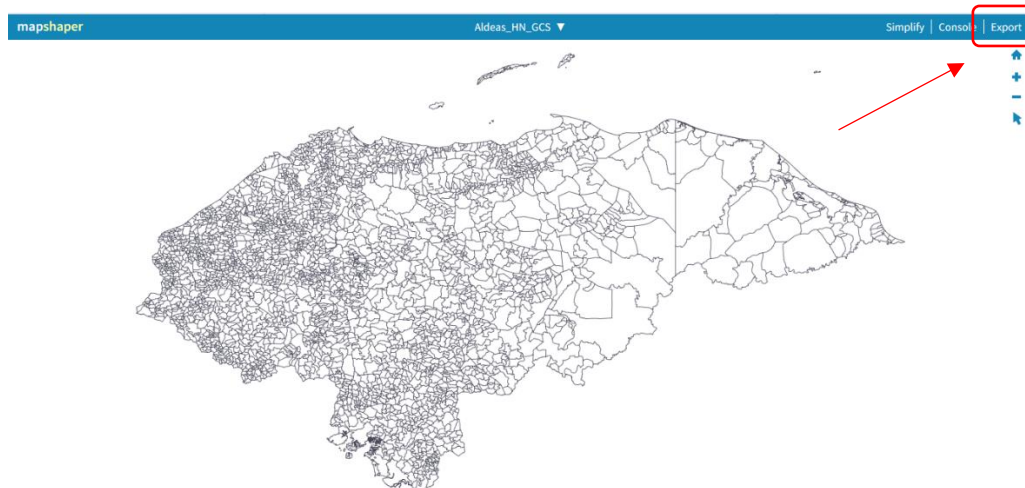
## 4.3 Seleccionar todos los archivos del **shape** y hacer clic en abrir.



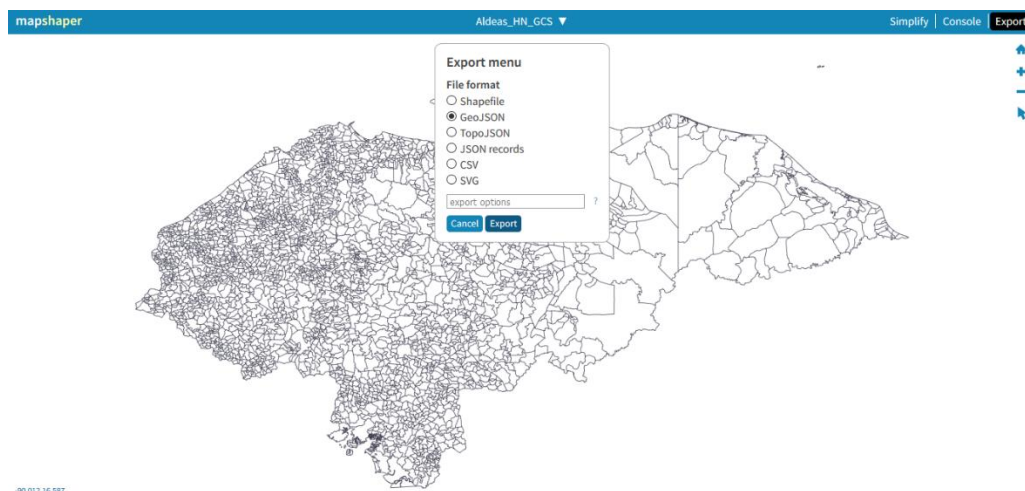
## 4.4 Marcar las opciones: **detect line intersections** y **snap vértices** y hacer clic en **importar**.



- 4.5 Se cargará el mapa o forma que hayamos seleccionado, luego, hacer clic en **exportar**.



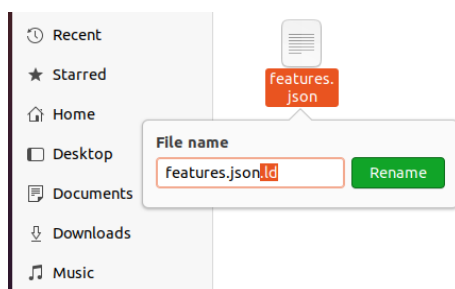
- 4.6 Seleccionar la opción **GeoJSON** y hacer clic en **exportar**.



Automáticamente se descargará el archivo JSON.

## 5. Conversión de archivo JSON a LD.

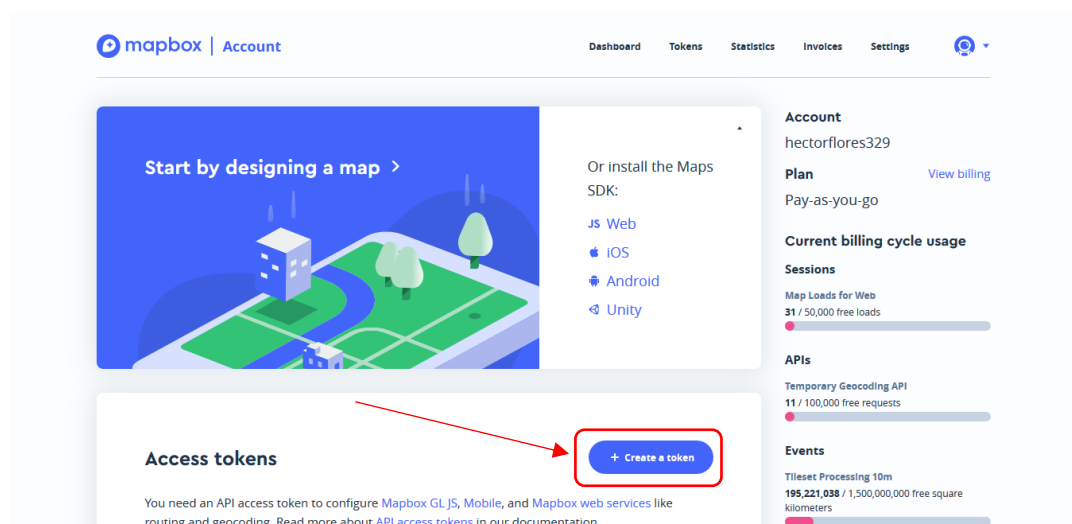
- 5.1 Una vez creado el JSON se **copia el archivo** a la máquina virtual.
- 5.2 Para convertirlo a un archivo **.ld** se renombra el archivo agregando la extensión **.ld**.



## 6. Creación de Token en Mapbox.

Se necesita una **cuenta** para poder crear un Tieset, en caso de no tener una cuenta, crearla en <https://www.mapbox.com/>

- 6.1 Una vez ingresado a la cuenta, se mostrará la siguiente pantalla. Hacer clic en **create a token**.



- 6.2 Se mostrará la siguiente pantalla en donde se le asignará en **nombre** de nuestro Token y se le darán los alcances o permisos. Luego presionar el botón **create token** ubicado en la parte final de la página.

### Create an access token

#### Token name

Choose a name to help associate it with a project.

Name

nombre\_del\_token

16 / 128

#### Token scopes

All tokens, regardless of the scopes included, are able to view styles, tilesets, and geocode locations for the token's owner. [Learn more.](#)

##### Public scopes

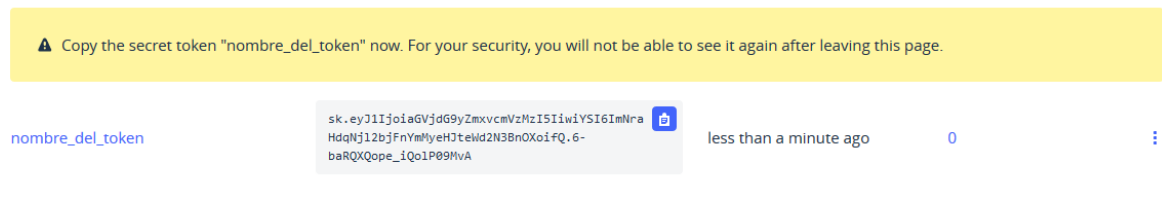
☒ STYLES:TILES ☒ STYLES:READ ☒ FONTS:READ ☒ DATASETS:READ  
☒ VISION:READ

##### Secret scopes

☒ SCOPES:LIST ☒ MAP:READ ☒ MAP:WRITE ☒ USER:READ  
☒ USER:WRITE ☒ UPLOADS:READ ☒ UPLOADS:LIST ☒ UPLOADS:WRITE  
☒ FONTS:LIST ☒ FONTS:WRITE ☒ STYLES:WRITE ☒ STYLES:LIST  
☒ TOKENS:READ ☒ TOKENS:WRITE ☒ DATASETS:LIST ☒ DATASETS:WRITE  
☒ TILESETS:LIST ☒ TILESETS:READ ☒ TILESETS:WRITE ☒ DOWNLOADS:READ  
☒ VISION:DOWNLOAD ☒ NAVIGATION:DOWNLOAD ☒ OFFLINE:READ ☒ OFFLINE:WRITE

Este Token se utilizará al momento de **modificar el zoom** de del Tileset.

6.3 Si el **Token** se crea correctamente, se deberá mostrar la siguiente pantalla:



**Observación:** Copiar y pegar token en un archivo de texto ya que no se mostrará nuevamente por seguridad.

## 7. Creación de Mapbox Tiling Services.

Abrir la terminal de Ubuntu y ejecutar los siguientes comandos y realizar las siguientes instrucciones:

7.1 `export MAPBOX_ACCESS_TOKEN = Token con acceso a los permisos de lectura, escritura.`

7.2 `tilesets upload-source username nombre_del_tileset  
/home/user/archive_json.ld`

Si el **Tileset** se crea correctamente, devolverá una línea como la siguiente: que será utilizado para crear la capa y aplicar el zoom al Tileset.

7.3 Crear un archivo **JSON** que permita crear el nombre de la capa. Copiar y pegar el siguiente código en el **editor de texto** de Python y guardar el archivo con la extensión **nombre\_del\_archivo.json**.

```
{
  "version": 1,
  "layers": {
    "hello_world": {
      "source": "mapbox://tileset-source/username/populated-places-source",
      "minzoom": 0,
      "maxzoom": 5
    }
  }
}
```

- 7.4 Crear un nuevo **tilesets** con la siguiente:  
**tilesets create username.hello-world-tiles --recipe hello-world-recipe.json --name "hello world"**

Si se crea correctamente el tileset deberá aparecer el siguiente mensaje en la terminal:

**message: 'Successfully created empty tileset username.hello-world-tiles. Publish your tileset to begin processing your data into vector tiles.'**

El **tileset** deberá aparecer en la lista de Mapbox en la web, pero sin información. La información aparecerá hasta publicarlo.

- 7.5 Publicar un tileset:  
**tilesets publish username.hello-world-tiles**

- 7.6 Para revisar el estado del tileset hay que ejecutar la siguiente línea:  
**tilesets status username.hello-world-tiles**

## 8. Modificar zoom del tileset.

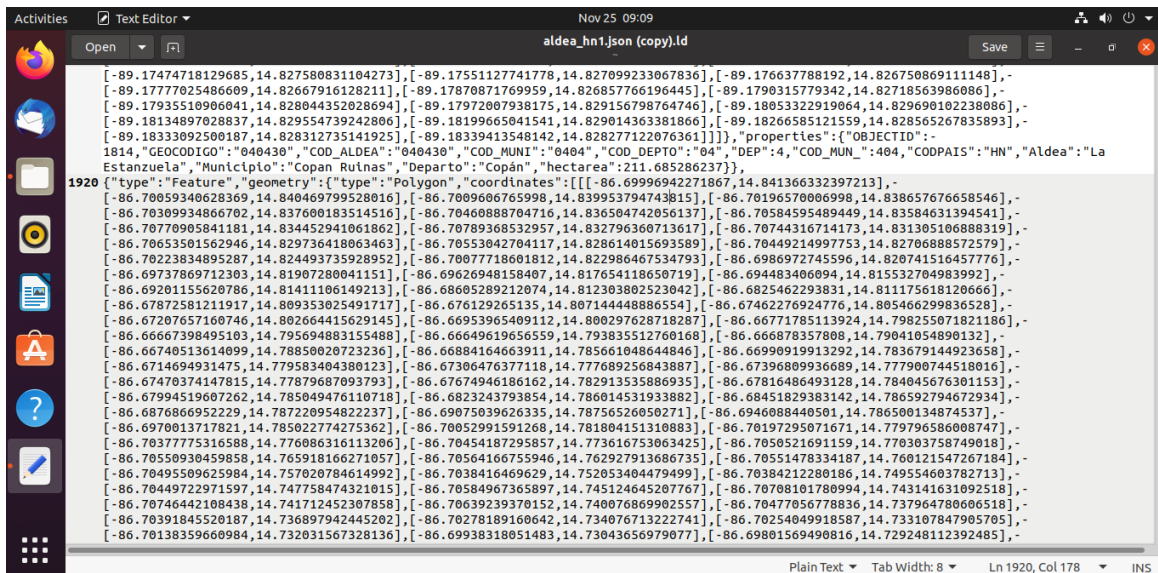
- 8.1 `export MAPBOX_ACCESS_TOKEN = Token con acceso a los permisos de lectura, escritura.`
- 8.2 Actualizar JSON desde Python utilizando el siguiente comando para actualizar el JSON con los cambios realizados:  
**tilesets update-recipe <tileset\_id> /path/to/recipe.json**
- 8.3 Actualizar el tileset para visualizar los cambios.  
**tilesets publish username.hello-world-tiles**
-



## Eliminar formas específicas del JSON.

Recomendación: eliminar las formas específicas utilizando el editor de texto en Ubuntu ya que permite una mayor identificación de shapes por bloques en el código.

- 9.1 Abrir el archivo a editar con un **editor de texto**.
- 9.2 Identificar la forma (shape) que se desea eliminar, se puede identificar por cada uno de sus **campos**.
- 9.3 Al identificarlo, se marcará el **bloque (con color gris)** y se diferenciará de los demás como se muestra a continuación:



- 9.4 Eliminar el bloque seleccionado y **guardar cambios**.