

# Ejecución Malla de Validación en Databricks

---

El siguiente documento explica el procedimiento que se debe realizar con el objetivo de poder obtener los resultados de la malla de validación en desde un notebook de [Databricks](#).

## Generación de la Malla de Validación de Tipo JSON

Para poder realizar la malla de validación desde un notebook de databricks, es necesario de antemano, haber cargado en el repositorio establecido, el archivo de tipo JSON, referente a la malla de validación del formulario al cuál se le desea realizar la validación.

Esta función es [create\\_json\\_malla](#), y su ejecución es de la siguiente manera:

### 1. Obtener el archivo excel con la malla de validación

Tener el archivo excel con el esquema definido en [Esquema Archivos](#) en la carpeta `data/validation_excel` que se encuentra definida y explicada en el documento:

Es importante tener claro el nombre del archivo en Excel ya que este será utilizado más adelante en el código.

### 2. General el archivo de tipo JSON desde un notebook

Una vez esta guardado con la estructura correcta, desde un notebook de [Jupyter](#) en el entorno local se deben ejecutar los siguientes comandos

```
# Se importan los paquetes que permite conectar el notebook a la ruta con los módulos
import sys
import os
```

```
# Se define la ruta a la carpeta del proyecto
ruta = r"ruta\a_la_carpeta\del_proyecto"
```

```
# Se añade al sistema de Python la ruta donde se encuentran los módulos
ruta_modulo = os.path.join(ruta, "scripts")
sys.append(ruta_modulo)
```

```
# Se importa la función create_json_malla del módulo malla_functions
from malla_functions import create_json_malla
```

```
# Se genera el archivo de tipo JSON y se almacena en la ruta definida
create_json_malla(name_malla = 'nombre_del_archivo_excel_sin_extension', ruta =
ruta)
```

Este código tomará el archivo de tipo excel y procederá a crear el documento tipo JSON con la validación en la carpeta `data\validation_json`.

## 2.1 Ejemplo del código

Suponiendo que se generó el archivo excel con la malla de validación para un formulario de excel cuyo nombre es `Malla_Formulario_RIT.xlsx`, el código de Python que se utilizara para realizazr este proceso es:

```
# Carga de Modulos
import sys
import os

# Definición de la ruta
ruta = r"C\Usuario\Documentos\MallaValidacion"

# Adición de la ruta al sistema
ruta_modulo = os.path.join(ruta, "scripts")
sys.append(ruta_modulo)

# Importación de la función desde el módulo
from malla_functions import create_json_malla

# Generar el archivo JSON
create_json_malla(name_malla = 'Malla_Formulario_RIT', ruta = ruta)
```

En el ejemplo, dentro de la carpeta `data\validation_json`, se generará un archivo JSON `Malla_Formulario_RIT.json` que contendrá la malla de validación para el formulario definido.

## 3. Realizar un Commit y Push de la información generada en el repositorio del Proyecto en GitHub

Una vez se ha generado el archivo JSON con la malla de validación, es necesario cargar el archivo al repositorio del proyecto en github con el objetivo de poder tener la información disponible en Databricks.

Para realizar esto, desde una consola de comandos del sistema operativo o desde un git bash, aplique los siguientes comandos:

Desde la consola de comando o el git bash dirijase a la carpeta donde se encuentra el proyecto, para realizar esto puede colocar toda la ruta luego del comando `cd`, o través de este mismo comando puede ingresar en cada paso a cada una de las carpetas `cd ruta`, `cd a_la_carpeta`, `cd del_proyecto`.

```
cd "ruta\a_la_carpeta\del_proyecto"
```

Se realiza un pull del repositorio alojado en GitHub con el objetivo de tener la última información disponible del repositorio y evitar conflictos a la hora de cargar cambios.

```
git pull origin master
```

Se añaden a los cambios del repositorio el archivo que generó:

```
git add data/validation_data/nombre_de_la_malla.json
```

Se realiza un `commit -m`, este comando la indica al repositorio que guarde cómo un nuevo cambio el archivo que se acabo de generar, al momento de general el commit se debe añadir el mensaje "*Cargue malla de validacion JSON nombre\_de\_la\_malla*", siguiendo el ejemplo del paso 2.1, el commit sería: `git commit -m "Cargue malla de validación JSON Malla_Formulario_RIT"`.

```
git commit -m "Cargue malla de validacion JSON nombre_de_la_malla"
```

Se suben los cambios realizados al repositorio que se encuentra en GitHub:

```
git push origin master
```