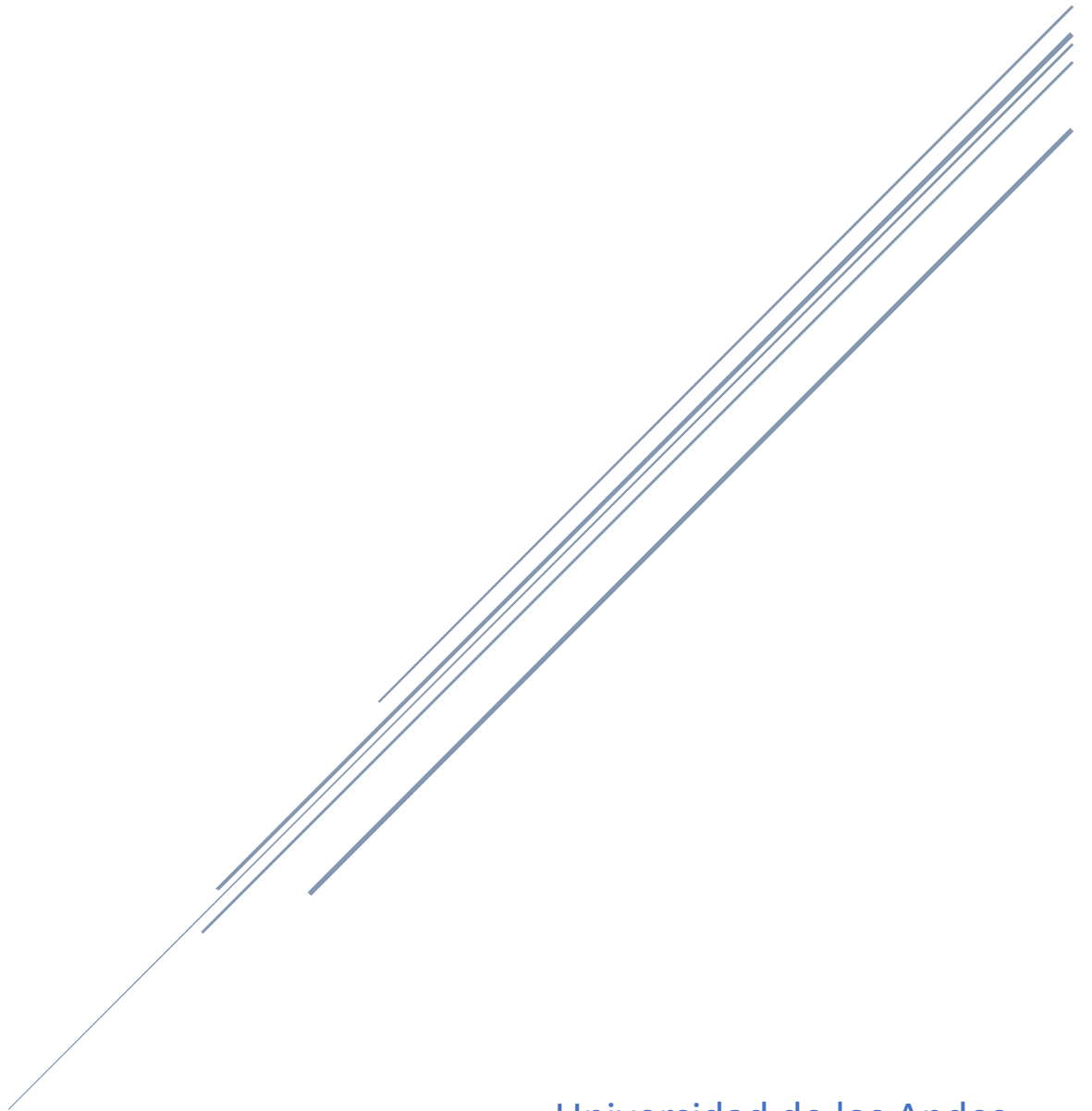


MANUAL TÉCNICO

App Isolation Forest - ElectroDunas



Universidad de los Andes
Proyecto aplicado en analítica de datos

Contenido

0.	Recursos.....	2
1.	Dockerización.....	2
2.	Creación de recursos en Azure	4
3.	Push de la imagen a Azure.....	8
4.	Creación del Pod de Kubernetes.....	10
5.	Creación del Site en Sharepoint.....	11
6.	Importar flujos de Power Automate.....	13
7.	Importar aplicación de PowerApps	15
8.	Publicación de PowerBI	18
9.	Consideraciones finales	20
	Anexos	21
	Diagrama Esquemático.....	21

0. Recursos

Los recursos necesarios para la implementación del modelo son:

1. **App.rar**: Contiene empaquetado el modelo junto con el archivo Python para la configuración de FastAPI en el contenedor Docker.
2. **App-Isolation_20240526151154.zip**: Contiene la aplicación en PowerApps la cual puede ser importada en otros entornos y/o dominios acorde a las necesidades.
3. **Dashboard anomalías.pbix**: Es el Dashboard el cual con el login al dominio correspondiente se puede publicar. Para la publicación se requiere una licencia de Power BI Pro.
4. **Flujo-isolation_20240526151327**: Es el paquete en Power Automate para la comunicación con el Kubernetes y Power Apps.
5. **Send_Alerta_20240526151235.zip**: Es el paquete en Power Automate para el envío de alertas a través de correo electrónico.

En cuanto al software o plataformas necesarias, se debe contar con Office 365 mínimo E1, pequeñas y medianas empresas, educación o cualquier que soporte PowerApps y PowerAutomate. Adicional se debe contar con Azure para crear un grupo de recursos que soportarán los contenedores y Kubernetes. Finalmente aunque opcional, una licencia de Power BI Pro, pero en su defecto se puede usar Power BI Desktop para la visualización y actualización del informe.

Para la plataforma de Sharepoint, usualmente viene integrada en cualquier paquete de Office 365, pero es importante la creación de un Site para garantizar la creación del repositorio y las listas correspondientes.

1. Dockerización

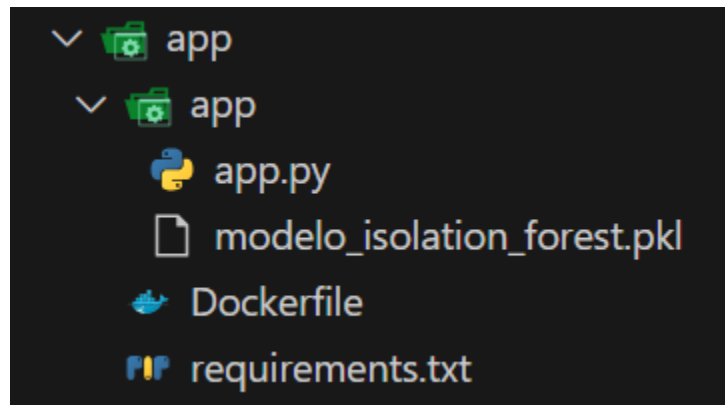
Para este apartado es importante contar con el paquete App.rar, donde contiene el modelo. En caso que no se tenga el modelo en formato Pickle, recordar que en el notebook “Proyecto de Detección de Anomalías en ElectroDunas” se cuenta con una instrucción para la exportación del modelo:



```
1 import pickle
2 with open('modelo_isolation_forest.pkl', 'wb') as file:
3     pickle.dump(model_if, file)
```

Los pasos para el proceso de Dockerización son:

1. Descomprimir el archivo app.rar.
2. Identificar la estructura de app, la cual contiene un archivo Python con la configuración de FastAPI, un modelo Pickle, en este caso es el modelo de Isolation Forest, un archivo DockerFile donde se encuentra la configuración de la dockerización y un archivo requirements.txt con los requerimientos necesarios para realizar las predicciones (librerías de Python necesarias para que todo funcione correctamente).



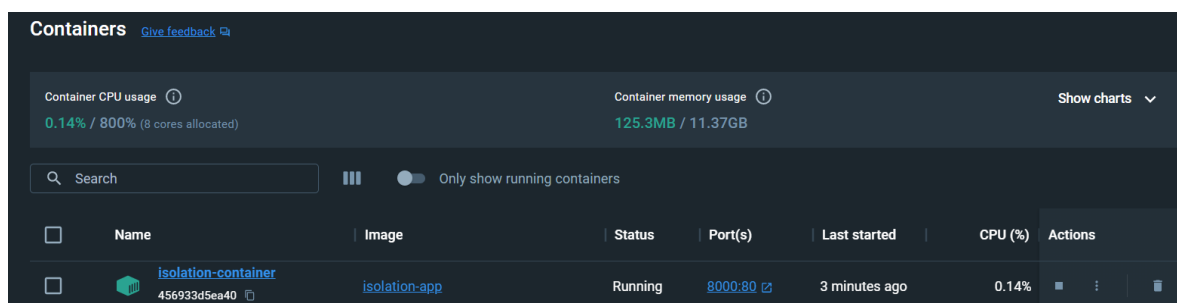
3. Abrir una terminal y cambiar la ruta donde está el DockerFile. Antes de dockerizar se puede comprobar que la aplicación funciona correctamente si se ejecuta el siguiente comando:

```
C:\Users\JD_01\OneDrive\Andes\Ciclo VIII\Proyecto aplicado en analítica de datos\Datos ElectroDunas\app>python app/app.py
INFO: Started server process [7892]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://127.0.0.1:80 (Press CTRL+C to quit)
INFO: 127.0.0.1:62554 - "GET /docs HTTP/1.1" 200 OK
INFO: 127.0.0.1:62554 - "GET /openapi.json HTTP/1.1" 200 OK
```

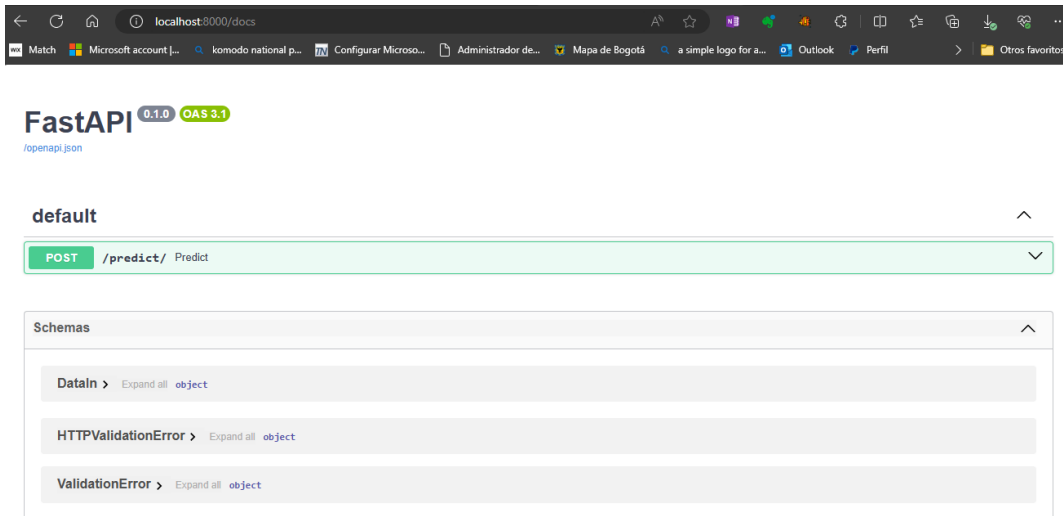
4. Ejecutar la siguiente línea:

```
C:\Users\JD_01\OneDrive\Andes\Ciclo VIII\Proyecto aplicado en analítica de datos\Datos ElectroDunas\app>docker build -t isolation-app .
```

5. Lo anterior permite que se cree una imagen en Docker en local, como se evidencia a continuación:



- Desde Docker puedes correr localmente la imagen a través del puerto 8000, donde se puede confirmar que el swagger funciona correctamente:

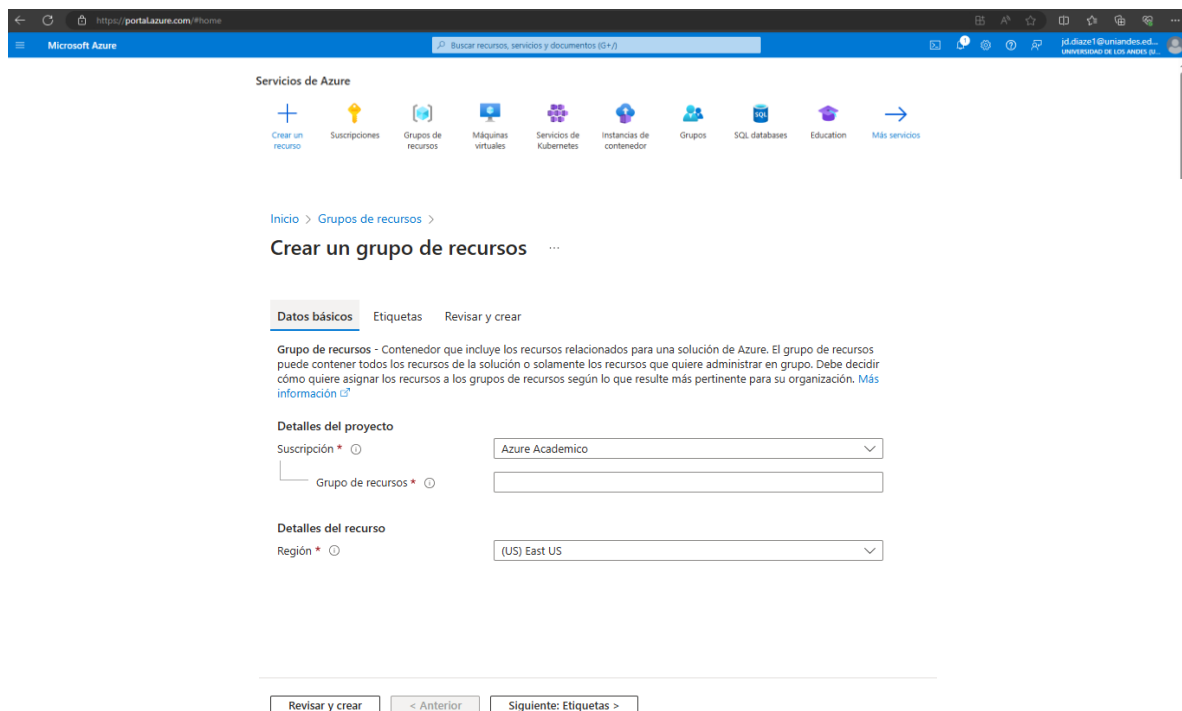


2. Creación de recursos en Azure

Con la imagen creada, se debe crear un grupo de recursos donde estarán las plataformas de Container Registry y Azure Kubernetes que permitirán contar con el Backend de la solución.

Para la creación de los recursos se pueden seguir los siguientes pasos:

- Desde la web de portal.azure.com se debe crear un grupo de recursos dando clic inicialmente en “Grupo de Recursos”, luego crear:



2. En este apartado se debe seleccionar la suscripción y el nombre del Grupo de Recursos. Una vez creado se puede evidenciar que al final quedó creado el grupo como “RG-JD”:

Inicio >

Grupos de recursos

Universidad de los andes (uniandes.onmicrosoft.com)

+ Crear ⚙ Administrar vista Actualizar Exportar a CSV 🔗 Abrir consulta Asignar etiquetas

Filtrar por cualquier campo Suscripción es igual a todo Ubicación es igual a todo X Agregar filtro

Mostrando de 1 a 5 de 5 registros. Sin agrupar Vista de lista

<input type="checkbox"/> Nombre ↑↓	Suscripción ↑↓	Ubicación ↑↓	
<input type="checkbox"/> (9) DefaultResourceGroup-EUS	AzureJD	East US	...
<input type="checkbox"/> (9) MA_defaultazuremonitorworkspace-eus_eastus_managed	AzureJD	East US	...
<input type="checkbox"/> (9) MC_RG-JD_KS-JD_eastus	AzureJD	East US	...
<input type="checkbox"/> (9) NetworkWatcherRG	AzureJD	East US	...
<input type="checkbox"/> (9) RG-JD	AzureJD	East US	...

3. Abrir el grupo de recursos creado, para luego crear las plataformas necesarias, en este caso Azure Container Registry. Dar clic el el botón “Crear” y luego buscar en el MarketPlace la plataforma de Container Registry:

RG-JD Grupo de recursos

Buscar + Crear ⚙ Administrar vista Eliminar grupo de recursos Actualizar Exportar a CSV ...

Información general

- Registro de actividad
- Control de acceso (IAM)
- Etiquetas
- Visualizador de recursos
- Eventos

Información esencial Vista JSON

Recursos Recomendaciones (2)

Filtrar por cualquier ca... Tipo es igual a todo X Agregar filtro Más (1)

Mostrando de 1 a 12 de 12 registros. ☐ Mostrar tipos ocultos Sin agrupar

Vista de lista

Marketplace

Get Started

Service Providers

Management

- Private Marketplace
- Private Offer Management

My Marketplace

- Favorites
- Recently created
- Private plans


Categories

- Contenedores (23)
- Proceso (22)

container registry

☐ Azure services only

Showing 1 to 20 of 45 results for 'con



Container Registry

Microsoft

Solution Template

Manage a Docker private registry as a first-class Azure resource.

Price varies

Create

- Una vez seleccionado, se debe seleccionar la suscripción y el grupo de recursos, usar un nombre adecuado y seguir los pasos correspondientes:

[Inicio](#) > [Grupos de recursos](#) > [RG-JD](#) > [Marketplace](#) > [Container Registry](#) >

Crear Registro de contenedor ...

Datos básicos Redes Cifrado Etiquetas Revisar y crear

Azure Container Registry permite compilar, almacenar y administrar artefactos e imágenes de contenedor en un registro privado para todos los tipos de implementación de contenedor. Use registros de contenedor de Azure con sus canalizaciones de desarrollo e implementación de contenedores actuales. Use Azure Container Registry Tasks para compilar imágenes de contenedor en Azure a petición, o bien automatizar compilaciones desencadenadas por actualizaciones del código fuente, actualizaciones de la imagen base de un contenedor o temporizadores. [Más información](#)

Detalles del proyecto

Suscripción *

Grupo de recursos * [Crear nuevo](#)

Detalles de instancia

Nombre del Registro * .azurecr.io

Ubicación *

Uso de zonas de disponibilidad ⓘ ☐

Revisar y crear

< Anterior

Siguiente: Redes >

- De forma similar se creará el Kubernte, los pasos son “Crear” desde el grupo de recursos, y en el Marketplace seleccionar Azure Kubernetes:

Marketplace ...

Get Started

Service Providers

Management

Private Marketplace

Private Offer Management

My Marketplace

Favorites

My solutions

Recently created

Private plans

Categories

Contenedores (24)



New! Get AI-c

Ask AI to suggest pro

☐ Azure services only

Showing 1 to 20 of 58 results for 'azure'



Azure Kubernetes Service (AKS)

Microsoft

Azure Service

A managed cluster with a Kubernetes orchestrator for container deployments.

- Una vez seleccionado el servicio, seguir los pasos considerando que el clúster puede ser “Economía de producción” si un grupo selecto de personas consumirán el servicio, esto permite que el consumo promedio mensual no supere los 50 dólares:

Crear un clúster de Kubernetes

Datos básicos Grupos de nodos Redes Integraciones Supervisión Avanzado Etiquetas Revisar y crear

Azure Kubernetes Service (AKS) administra el entorno de Kubernetes hospedado, a la vez que facilita y agiliza la implementación y la administración de aplicaciones en contenedores sin necesidad de experiencia relativa. También elimina la carga de las operaciones en curso y el mantenimiento mediante el aprovisionamiento, la actualización y el escalado de los recursos a petición, sin tener que desconectar las aplicaciones. [Más información](#)

Detalles del proyecto

Seleccione una suscripción para administrar los recursos implementados y los costos. Use los grupos de recursos como carpetas para organizar y administrar todos los recursos.

Suscripción *

Grupo de recursos *
[Crear nuevo](#)

Detalles del clúster

Configuración preestablecida del clúster *

Para personalizar rápidamente el clúster de Kubernetes, elija una de las

Anterior **Siguiente** Revisar y crear


- En la sección de grupo de nodos, por términos de economía en los recursos se puede seleccionar uno (1), de igual forma recordar que es fácilmente escalable:

Crear un clúster de Kubernetes

Datos básicos Grupos de nodos Redes Integraciones Supervisión Avanzado Etiquetas Revisar y crear

Grupos de nodos

Además del grupo de nodos principal necesario configurado en la pestaña Conceptos básicos, también puede agregar grupos de nodos opcionales para controlar una gran variedad de cargas de trabajo. [Más información](#)

+ Agregar grupo de nodos  Eliminar

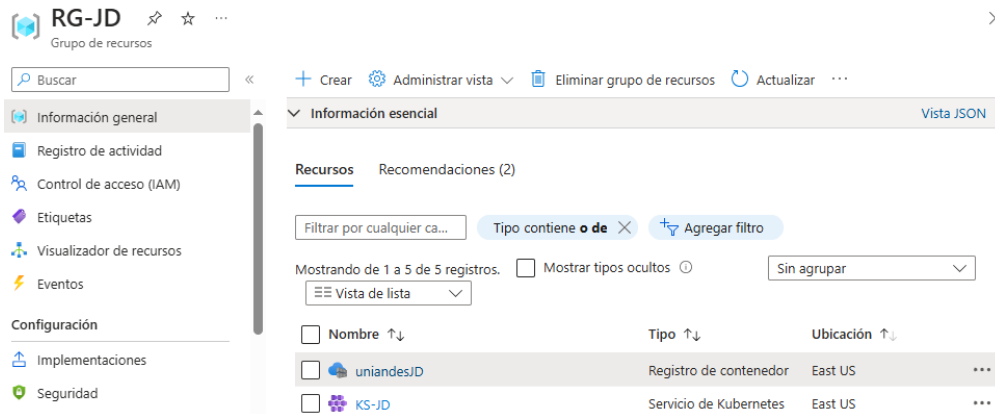
<input type="checkbox"/>	Nombre	Modo	Tamaño de nodo	SKU DEL SISTEMA...	Número de no...	Zonas
<input type="checkbox"/>	agentpool	System	Standard_DS2_v2 (...)	Ubuntu	1 - 1	Ningun

Habilitar nodos virtuales

Los nodos virtuales permiten el escalado flexible respaldado por Azure Container Instances sin servidor. [Más información](#)

Anterior **Siguiente** Revisar y crear

- Una vez creados los recursos se puede evidenciar que tenemos las plataformas necesarias para cubrir la parte Backend de la solución:

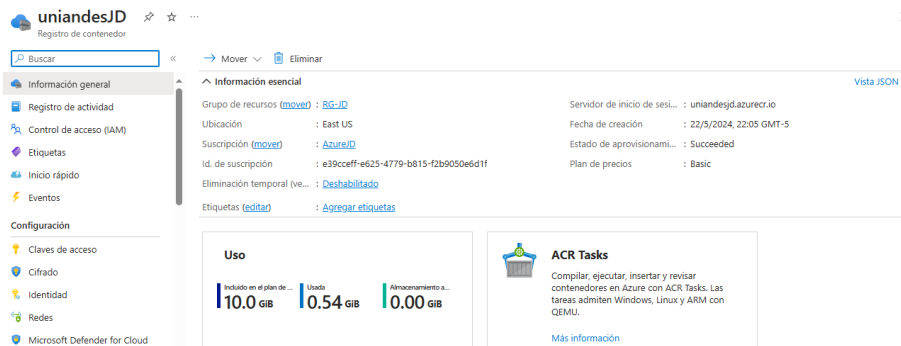


Nombre	Tipo	Ubicación
uniandesJD	Registro de contenedor	East US
KS-JD	Servicio de Kubernetes	East US

3. Push de la imagen a Azure

Para realizar el push de la imagen a Azure, se requiere ejecutar unas instrucciones desde la terminal para luego enviar a Azure. Los pasos a seguir son:

- Entrar al recurso de Azure Container Registry creado en la sección anterior:



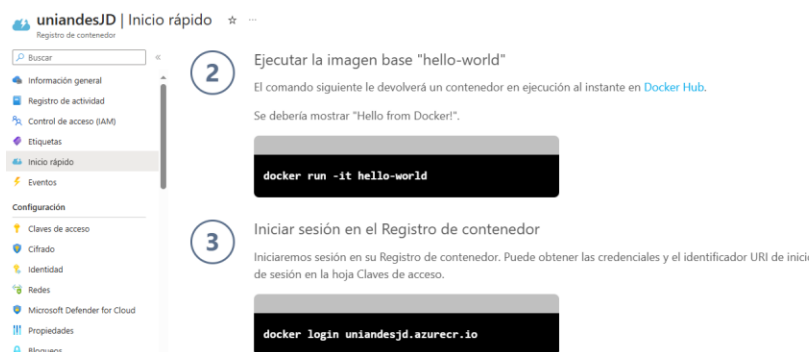
Uso

Uso	Almacenamiento a...
10.0 GIB	0.54 GIB

ACR Tasks

Compilar, ejecutar, insertar y revisar contenedores en Azure con ACR Tasks. Las tareas admiten Windows, Linux y ARM con QEMU.

- En la sección de “Inicio Rápido” se pueden explorar los pasos para realizar este proceso:



2 Ejecutar la imagen base "hello-world"

El comando siguiente le devolverá un contenedor en ejecución al instante en [Docker Hub](#).

Se debería mostrar "Hello from Docker!".

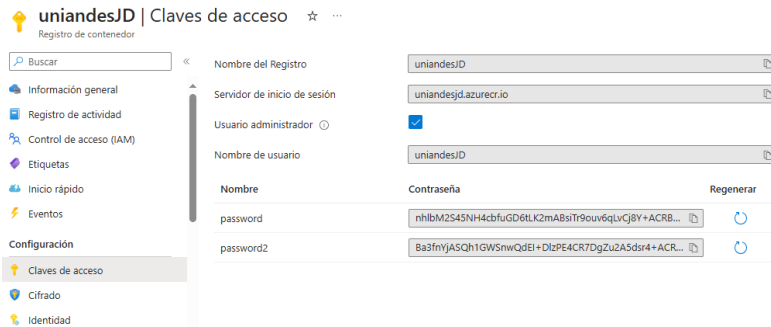
```
docker run -it hello-world
```

3 Iniciar sesión en el Registro de contenedor

Iniciaremos sesión en su Registro de contenedor. Puede obtener las credenciales y el identificador URI de inicio de sesión en la hoja Claves de acceso.

```
docker login uniandesjd.azurecr.io
```

3. De igual forma en la terminal se debe realizar el login, en la sección de “Claves de acceso” se puede contar con usuario y clave para realizarlo desde la terminal:



```
C:\Users\JD_01\OneDrive\Andes\Ciclo VIII\Proyecto aplicado en analítica de datos\Datos ElectroDunas\app>docker login uniandesjd.azurecr.io
Username: uniandesJD
Password:
Login Succeeded
```

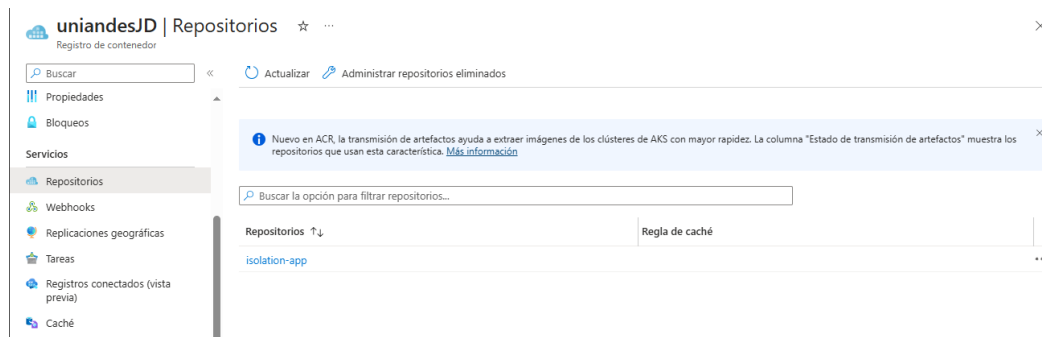
4. Una vez realizado el login, realizar el push:

```
C:\Users\JD_01\OneDrive\Andes\Ciclo VIII\Proyecto aplicado en analítica de datos\Datos ElectroDunas\app>docker push uniandesjd.azurecr.io/isolation-app
Using default tag: latest
The push refers to repository [uniandesjd.azurecr.io/isolation-app]
ee06e8cb06ca: Preparing
ec287426ed11: Preparing
e717ba135ef1: Preparing
380ae90835b9: Preparing
f360a95d5ce5: Preparing
9301fe799a03: Preparing
```

5. Se puede realizar el pull para comprobar que todo se haya realizado correctamente:

```
C:\Users\JD_01\OneDrive\Andes\Ciclo VIII\Proyecto aplicado en analítica de datos\Datos ElectroDunas\app>docker pull uniandesjd.azurecr.io/isolation-app
Using default tag: latest
latest: Pulling from isolation-app
Digest: sha256:523c563f1b4e69509f21bda3197902e183626fe14f148942b6149e8c795ae13d
Status: Image is up to date for uniandesjd.azurecr.io/isolation-app:latest
uniandesjd.azurecr.io/isolation-app:latest
```

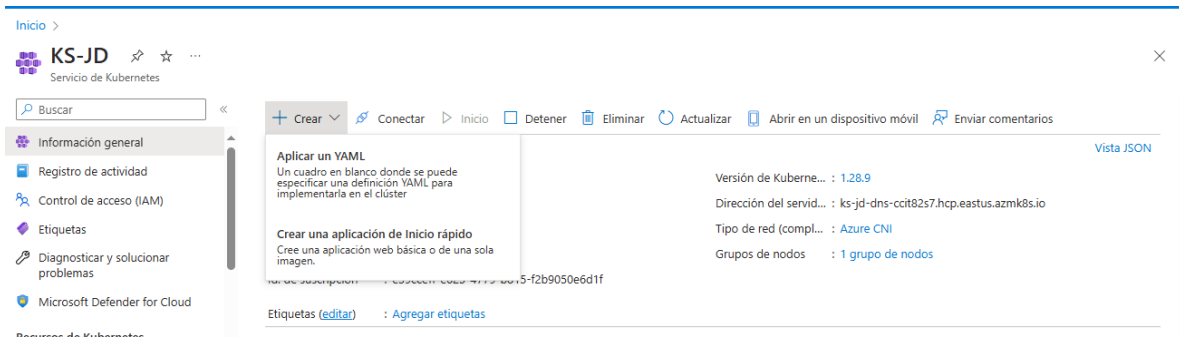
6. Finalmente, desde Azure en la sección servicios, “Repositorios” se puede evidenciar la imagen:



4. Creación del Pod de Kubernetes

El servicio de Kubernetes permite que la aplicación sea escalable y confiable. Dado que ya se cuenta con la información de la imagen en Azure, se puede crear el Pod de la siguiente forma:

1. En la sección del recurso del Kubernete creado en la sección 2, clic en el botón de crear y luego seleccionar “Crear una aplicación de inicio rápido”:

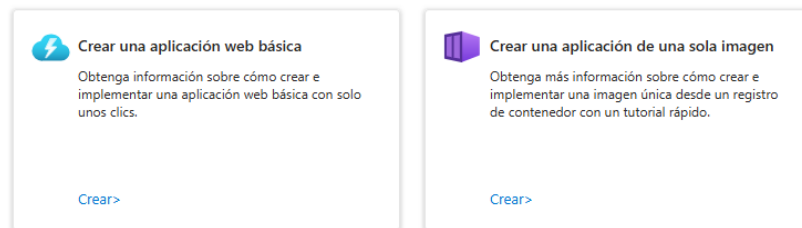


2. Seleccionar “Crear una aplicación de una sola imagen”:

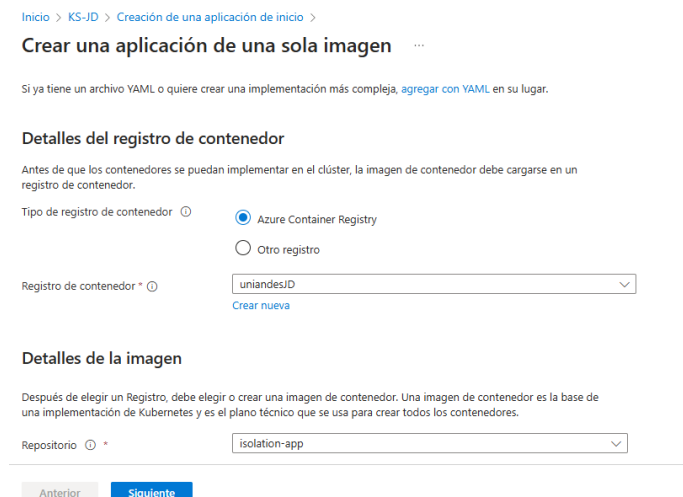
Creación de una aplicación de inicio

Puede implementar una aplicación web básica o de una sola imagen para empezar a usar Kubernetes. No se requiere ninguna experiencia anterior de Kubernetes.

Si ya tiene un archivo YAML o quiere crear una implementación más compleja, [agregar con YAML](#) en su lugar.



3. Seleccionar como registro de contenedor “Azure Container Registry”, luego seleccionar el Registro de contenedor correspondiente y el Repositorio “isolation-app”:



- Seguir las instrucciones donde solo se necesita una réplica si el consumo de servicio no va a ser tan alto y con el fin de garantizar economía del recurso.
- Una vez creado el Pod de Kubernetes, se puede comprobar que la aplicación “isolation-app” fue creada con éxito:

KS-JD | Cargas de trabajo

Recursos de Kubernetes

- Espacios de nombres
- Cargas de trabajo
- Servicios y entradas
- Almacenamiento
- Configuración
- Recursos personalizados
- Eventos
- Comando de ejecución

Configuración

- Grupos de nodos
- Configuración de clústeres
- Escalado de aplicaciones

Implementaciones

Filtrar por nombre de implementación:

Filtrar por espacio de nombres:

Agregar filtro de etiqueta

<input type="checkbox"/>	Nombre	Espacio de nombres	Listo	Actualizado	Disponible	Edad ↓
<input type="checkbox"/>	coredns	kube-system	✓ 2/2	2	2	4 días
<input type="checkbox"/>	coredns-autoscaler	kube-system	✓ 1/1	1	1	4 días
<input type="checkbox"/>	konnnectivity-agent	kube-system	✓ 2/2	2	2	4 días
<input type="checkbox"/>	metrics-server	kube-system	✓ 2/2	2	2	4 días
<input type="checkbox"/>	ama-metrics	kube-system	✓ 1/1	1	1	4 días
<input type="checkbox"/>	ama-metrics-ksm	kube-system	✓ 1/1	1	1	4 días
<input type="checkbox"/>	ama-metrics-operator-targets	kube-system	✓ 1/1	1	1	4 días
<input type="checkbox"/>	isolation-app	default-1716439366677	✓ 1/1	1	1	4 días

- En la sección de “Servicios y entradas” se puede confirmar la ip del servicio, el cuál será clave para la configuración en el flujo de Power Automate:

KS-JD | Servicios y entradas

Recursos de Kubernetes

- Espacios de nombres
- Cargas de trabajo
- Servicios y entradas
- Almacenamiento
- Configuración
- Recursos personalizados
- Eventos
- Comando de ejecución

Configuración

- Grupos de nodos

Servicios

Filtrar por nombre de servicio:

Filtrar por espacio de nombres:

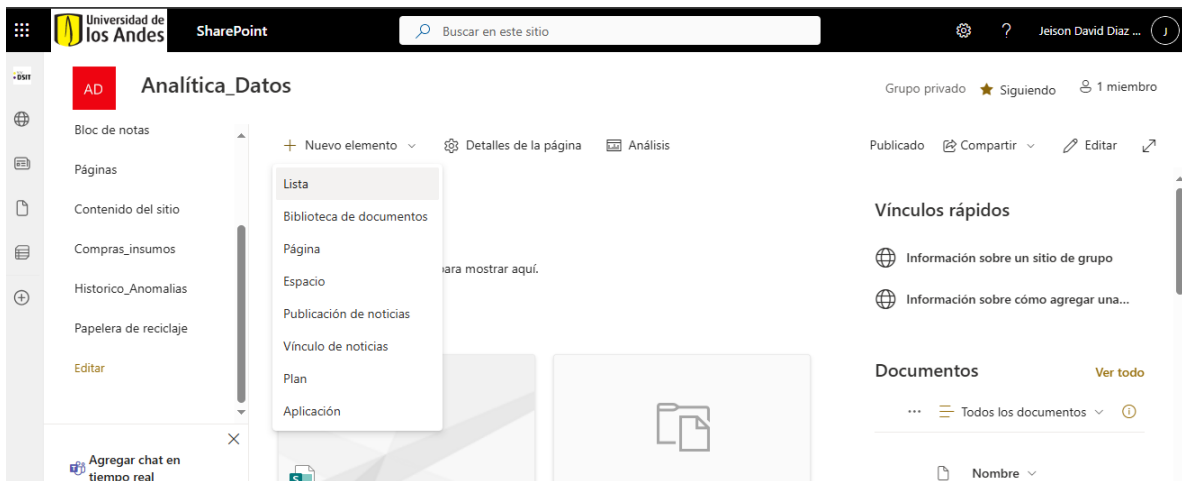
Agregar filtro de etiqueta

<input type="checkbox"/>	Nombre	Espacio de nombres	Estado	Tipo	Dirección IP d...	Dirección IP ext...
<input type="checkbox"/>	kubernetes	default	✓ Aceptar	ClusterIP	10.0.0.1	
<input type="checkbox"/>	kube-dns	kube-system	✓ Aceptar	ClusterIP	10.0.0.10	
<input type="checkbox"/>	metrics-server	kube-system	✓ Aceptar	ClusterIP	10.0.104.156	
<input type="checkbox"/>	ama-metrics-ksm	kube-system	✓ Aceptar	ClusterIP	10.0.67.66	
<input type="checkbox"/>	ama-metrics-operator-t...	kube-system	✓ Aceptar	ClusterIP	10.0.61.154	
<input type="checkbox"/>	isolation-app-service	default-1716439366677	✓ Aceptar	LoadBalancer	10.0.205.51	20.75.217.53

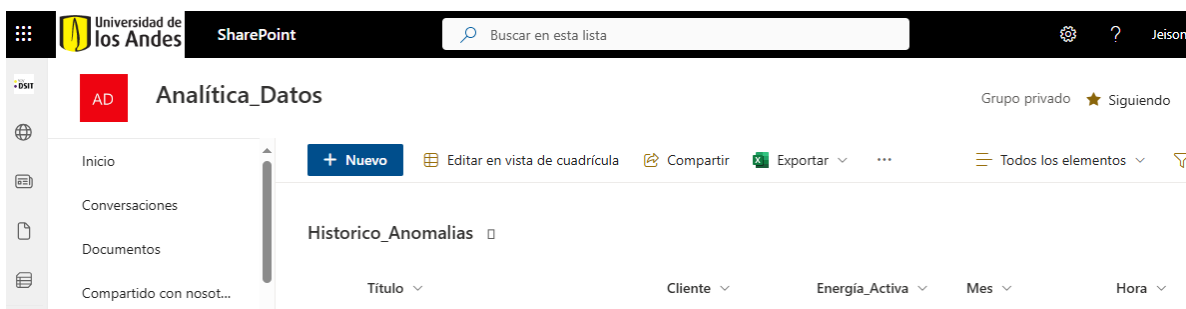
5. Creación del Site en Sharepoint

Es importante crear un Site en Sharepoint donde se tendrá la documentación y la lista de histórico de anomalías donde PowerApps registrará información y es consumida por Power BI. Los pasos a seguir son:

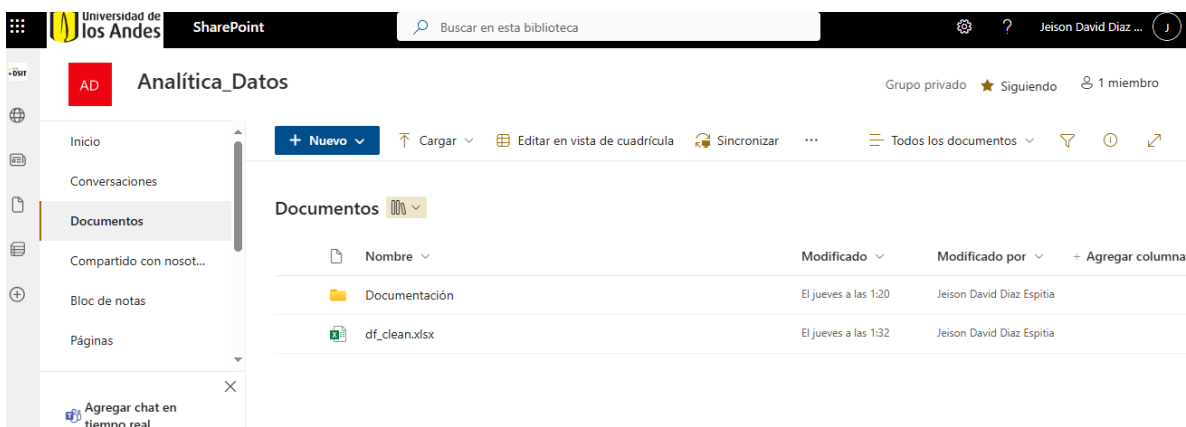
1. Crear un Site en Sharepoint con el nombre deseado, luego el Nuevo elemento crear una lista con el nombre “Histórico_Anomalias”:

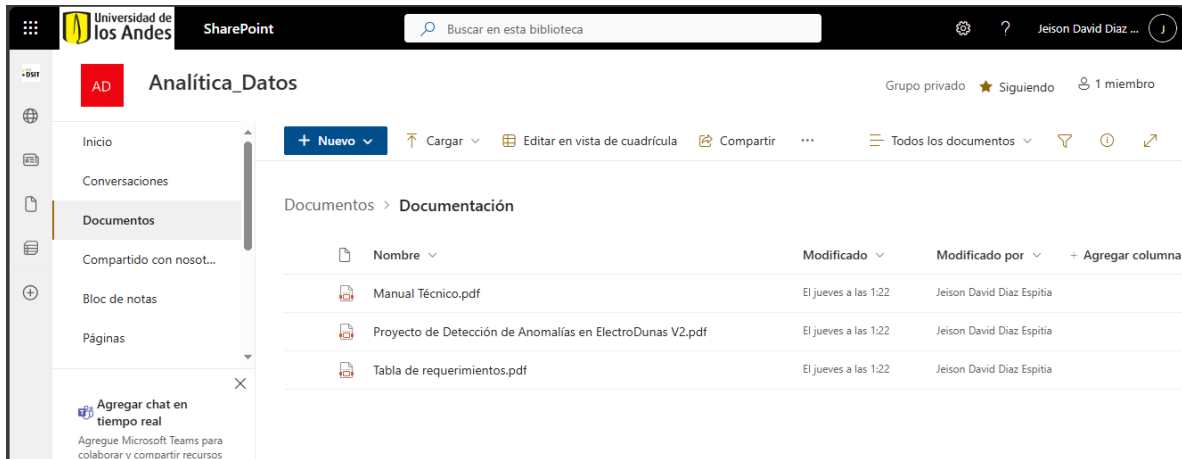


2. Una vez dentro de la lista, crear cuatro campos con “Cliente”, “Energía_Activa”, “Mes” y “Hora” donde todas serán de tipo “Línea de Texto”:



3. En la ficha de documentos, cargar la información de “df_clean.xlsx” y la carpeta de documentación con los archivos necesarios:

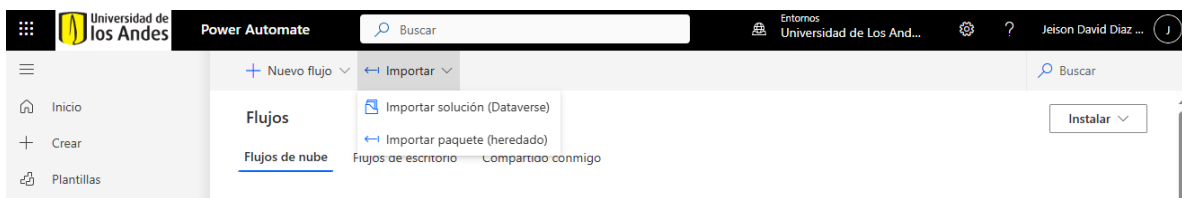




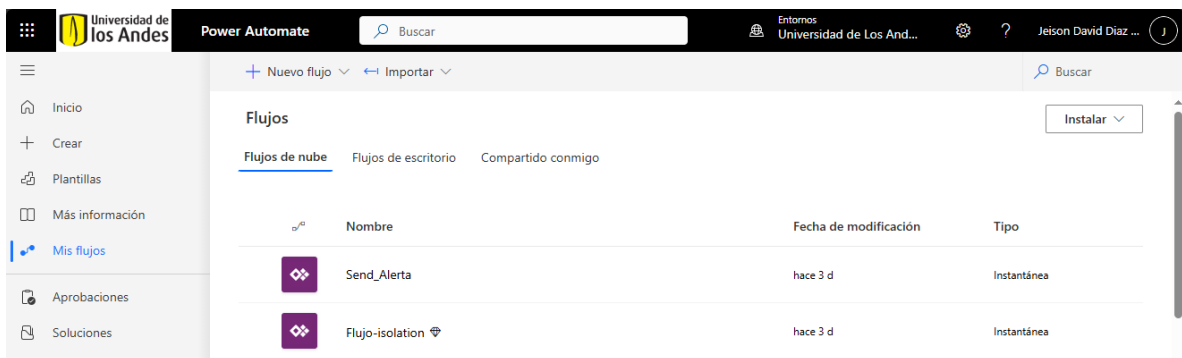
6. Importar flujos de Power Automate

Los recursos necesarios para la creación de estos flujos son Flujo-isolation_20240526151327.zip y Send_Alerta_20240526151235.zip. Los pasos a seguir son:

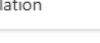
1. En Power Automate importar los dos archivos Zip en la opción importar y luego Importar paquete:



2. Una vez importados los paquetes, se podrá visualizar que se cargaron correctamente:



3. Para el caso de Flujo-isolation se debe editar para configurar la ip en el paso de HTTP:



Universidad de los Andes

Power Automate

Buscar

Entornos Universidad de los Andes

Flujo: isolation

Enviar comentarios

Copilot

Guardar

Compartir

HTTP

Parámetros

Configuración

Vista de código

Acerca de

URI *

http://20.75.217.53/predict/

Method *

POST

Headers

Content-Type	application/json
Accept	*/*

Queries

Escribir clave	Escribir valor
----------------	----------------

Body

```
{
  "Active_energy": Active_energy,
  "Mes": Mes,
  "Hora": Hora
}
```

PowerApps (V2)

+

HTTP

+

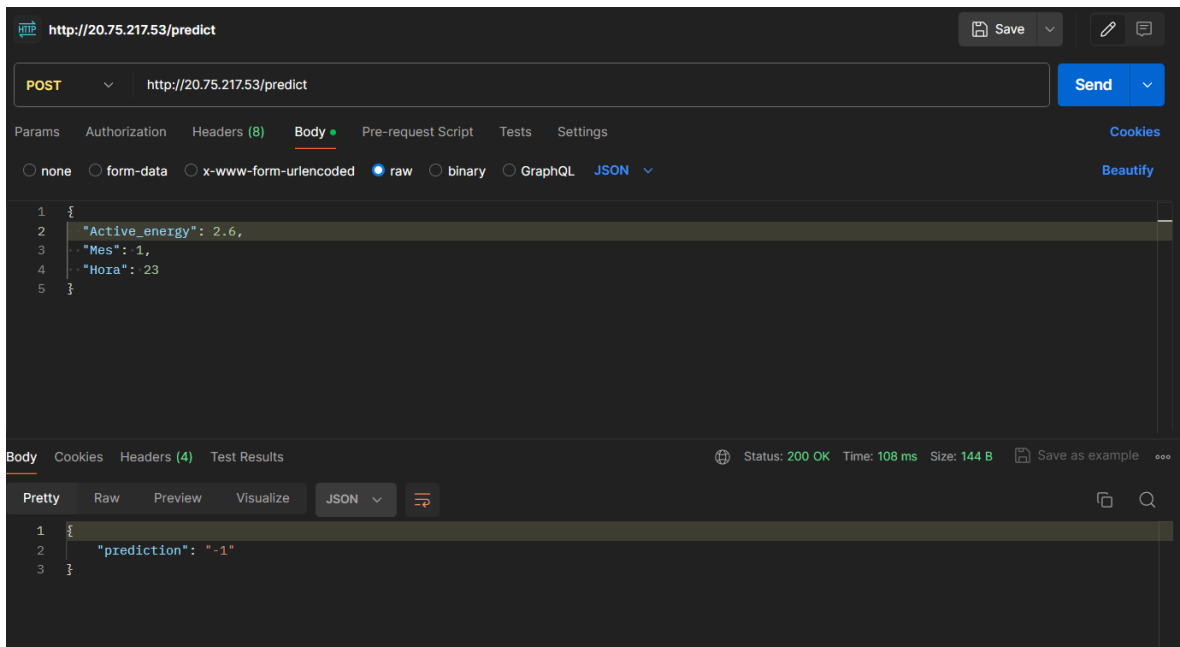
Inicializar variable

+

Responder a un flujo de aplicación Power App

4. En el flujo Send_Alerta se debe configurar los destinatarios dando clic en editar y luego modificando el paso de “Enviar correo electrónico (V2):

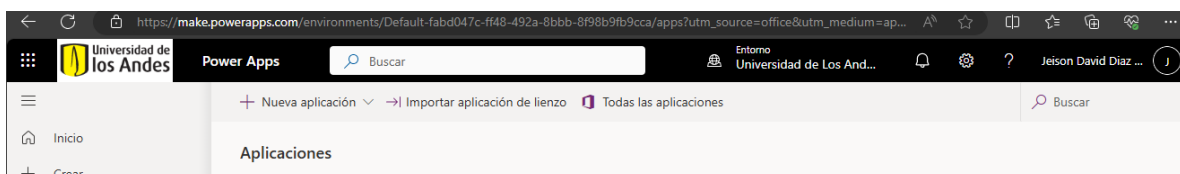
- Finalmente recordar que para la configuración de la IP, se pueden realizar pruebas a través de Postman apuntando el consumo de servicio a la URL [http://\[IP\]/predict](http://[IP]/predict) usando POST:



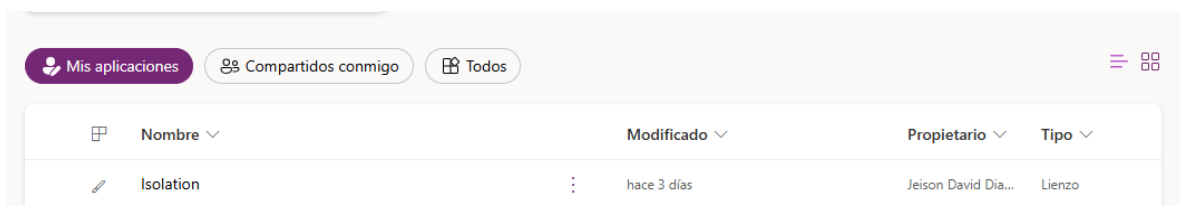
7. Importar aplicación de PowerApps

Para este caso, se debe usar el paquete `App-Isolation_20240526151154.zip`, donde a través de la plataforma de PowerApps se importará el paquete siguiendo los siguientes pasos:

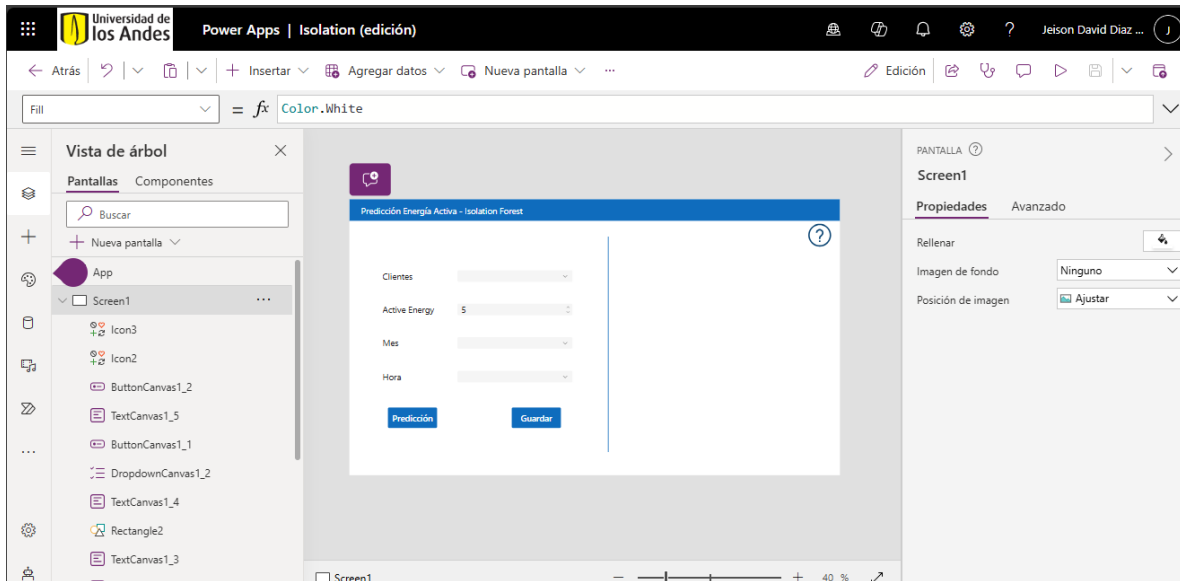
- Importar el paquete de PowerApps dando clic en “Importar aplicación de lienzo”:



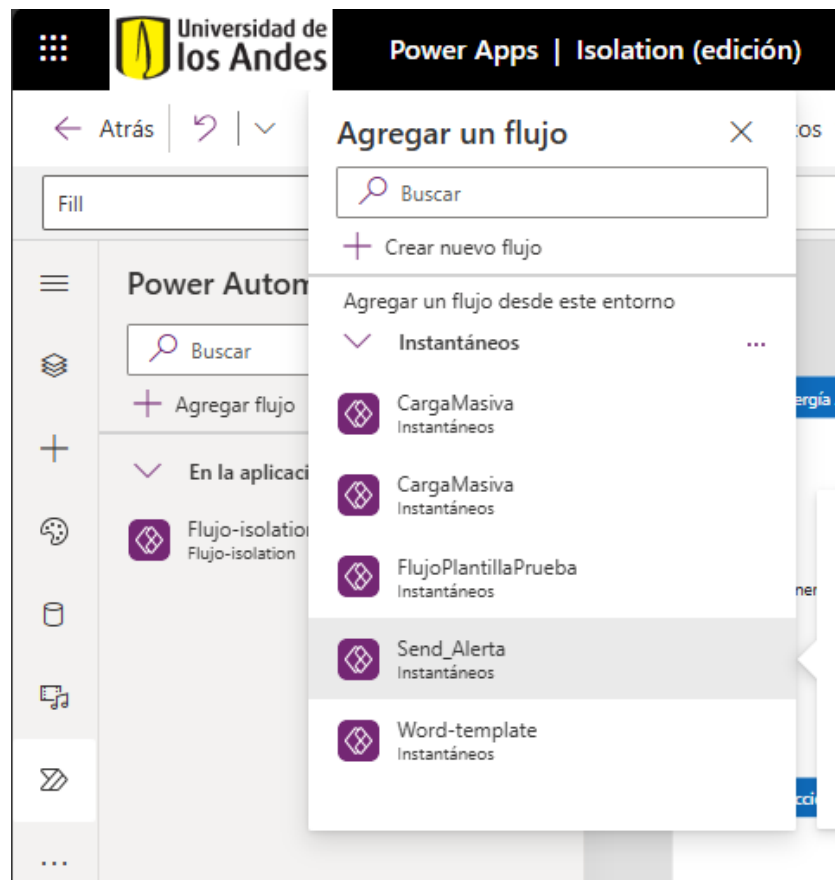
- Una vez importado, se podrá observar la aplicación en el listado:



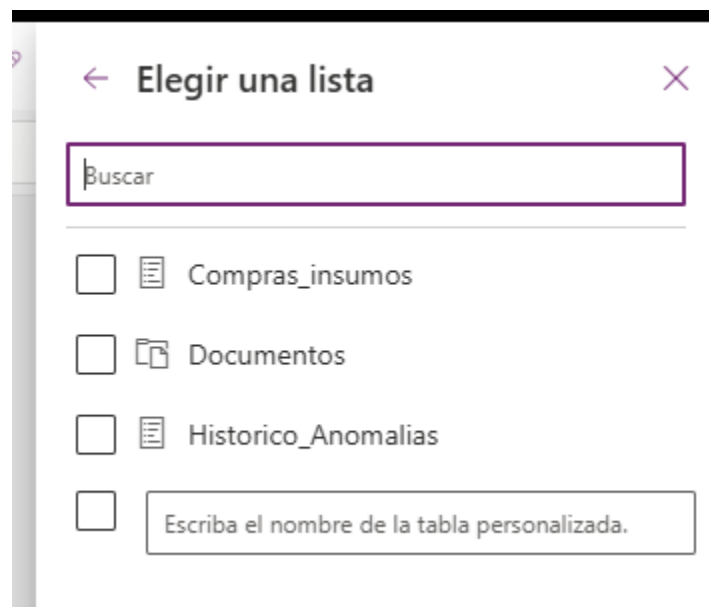
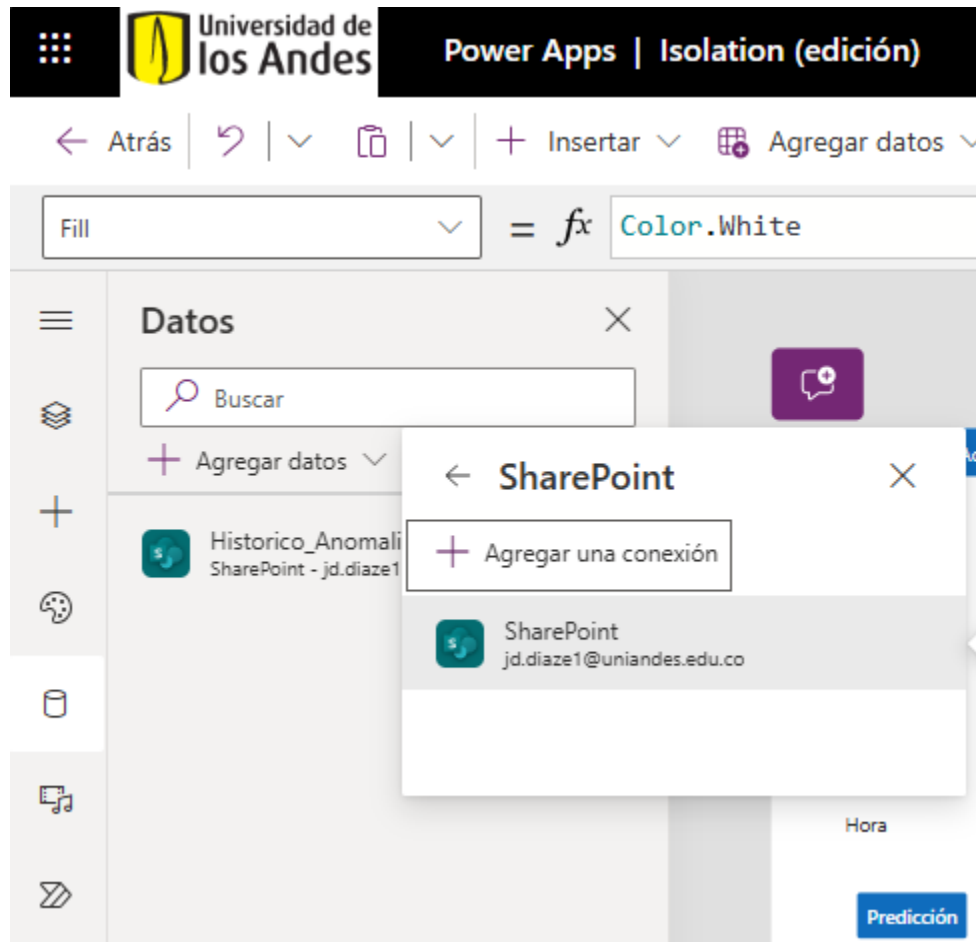
3. Se debe dar clic en editar la aplicación para entrar al modo desarrollador en línea y modificar unas conexiones:



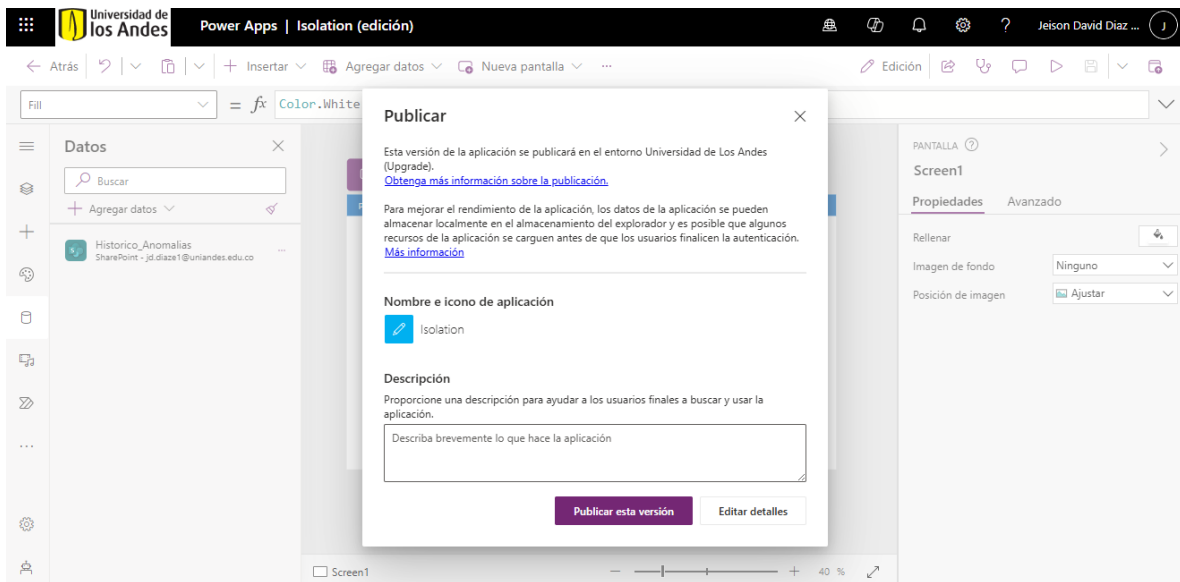
4. En el menú de PowerAutomate, dentro de PowerApps se pueden importar los flujos especificados en la sección 6:



5. En el menú de datos de PowerApps se puede importar la lista de Sharepoint creada en la sección 5:



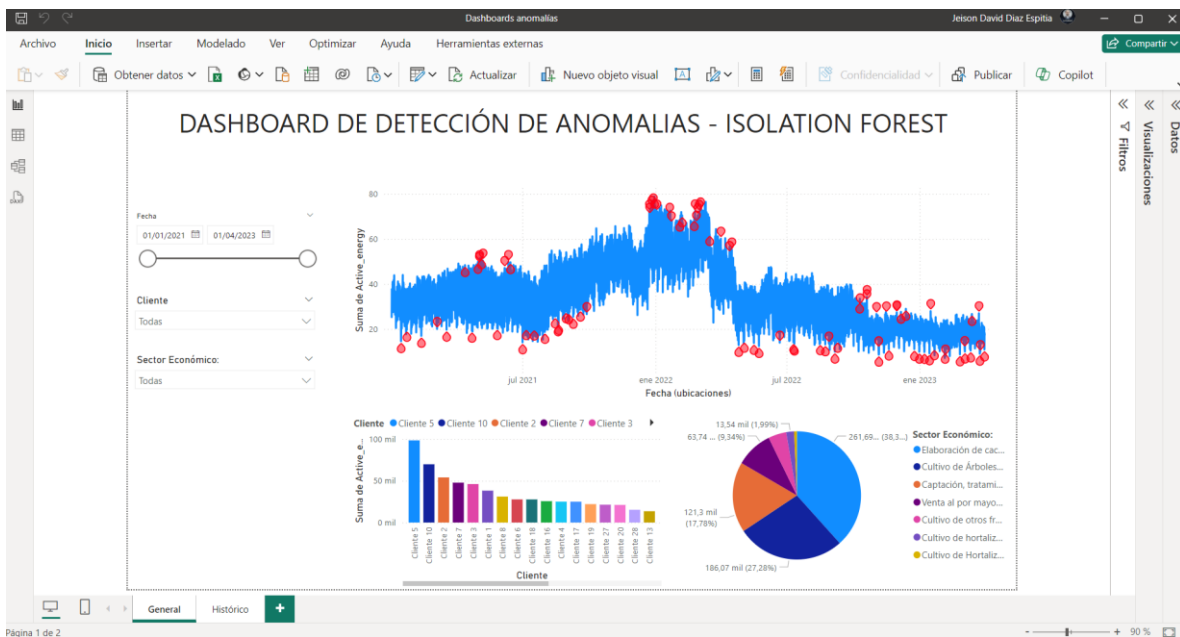
- Finalmente, se puede dar clic en el ícono de publicar en la parte superior derecha y publicar la versión. Antes de publicar es importante realizar pruebas para validar que todo funcione correctamente:



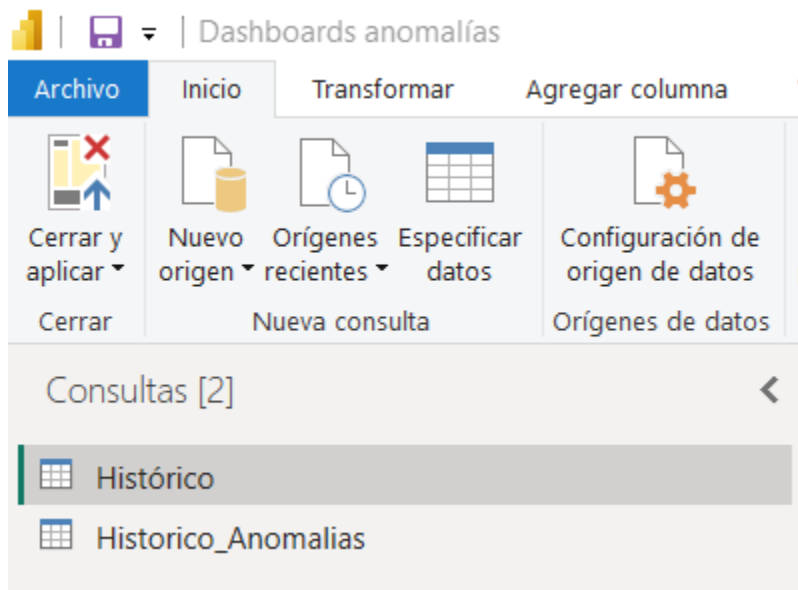
8. Publicación de PowerBI

Finalmente, con todas las configuraciones, se puede publicar el tablero siguiendo los siguientes pasos:

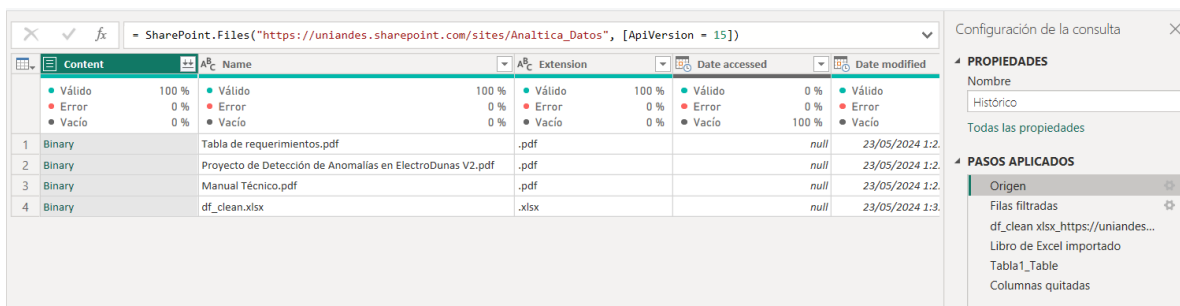
1. Abrir el archivo “Dashboards anomalías.pbix” en Power BI Desktop:



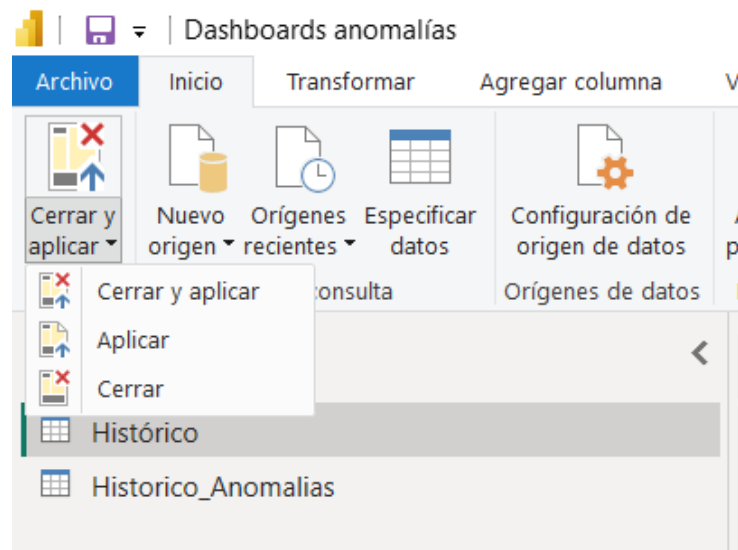
2. En el editor de Power Query de la aplicación, identificar las dos consultas con las que cuenta:



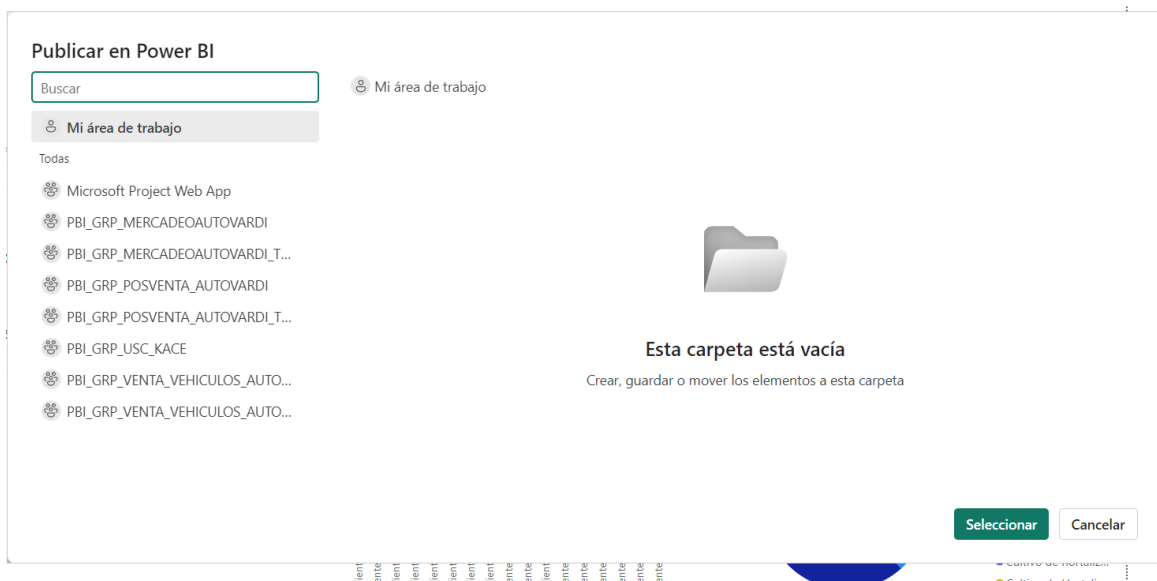
3. Para cada consulta, modificar el primer paso de origen apuntando a las URL de las listas de Sharepoint y archivo correspondiente:



4. Cerrar y Aplicar los cambios realizados desde el editor de Power Query:



5. Clic en “Publicar” en el menú Inicio del programa, y seleccionar el espacio de trabajo con la licencia correspondiente:



9. Consideraciones finales

Es importante validar cada uno de los recursos, desde la dockerización, implementación en Azure, y configuración de las plataformas de Office 365 como son Sharepoint, Power Apps y Power Automate.

En caso de alguna pregunta o soporte por favor contactarse con Jeison David Díaz Espitia al celular +573208357098 por llamada o WhatsApp, o al correo jd.diaze1@uniandes.edu.co.

Finalmente considerar que si se necesita una prueba con el demo se debe contar con un dominio uniandes y solicitar permisos a la plataforma enviando un correo a jd.diaze1@uniandes.edu.co.

Anexos

Diagrama Esquemático

