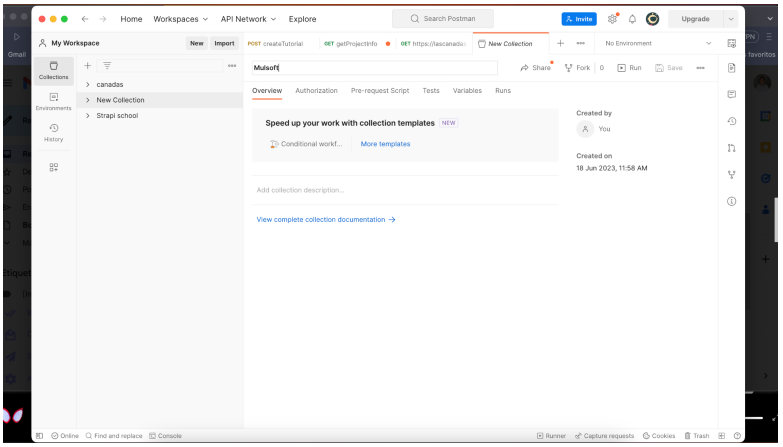
Practicas con Mulsoft para SP Solutions

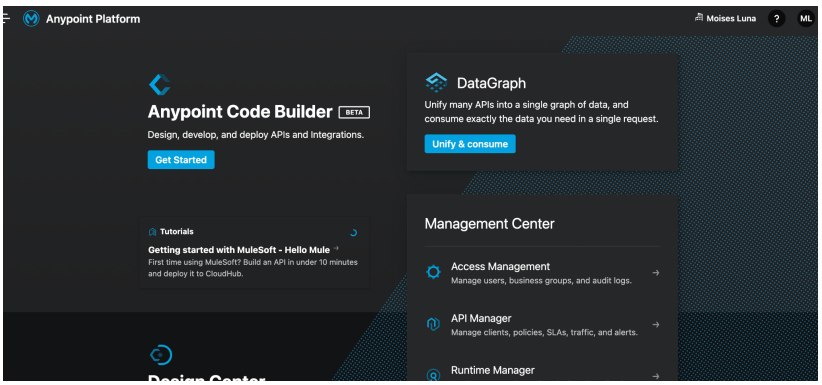
A continuación listo la serie de pasos que realicé durante la práctica trabajando con la suite de herramientas de integración de MuleSoft.

Build your first Hello Mule Application

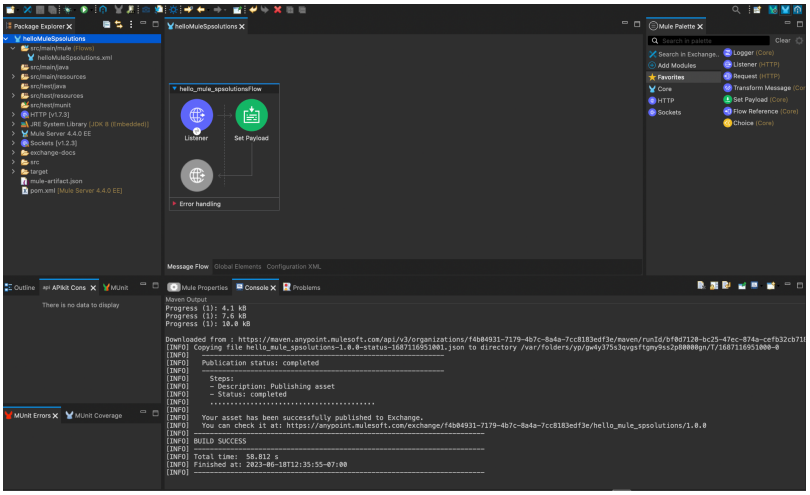
1. **Crear una nueva colección de API's en postman para el ejercicio.**



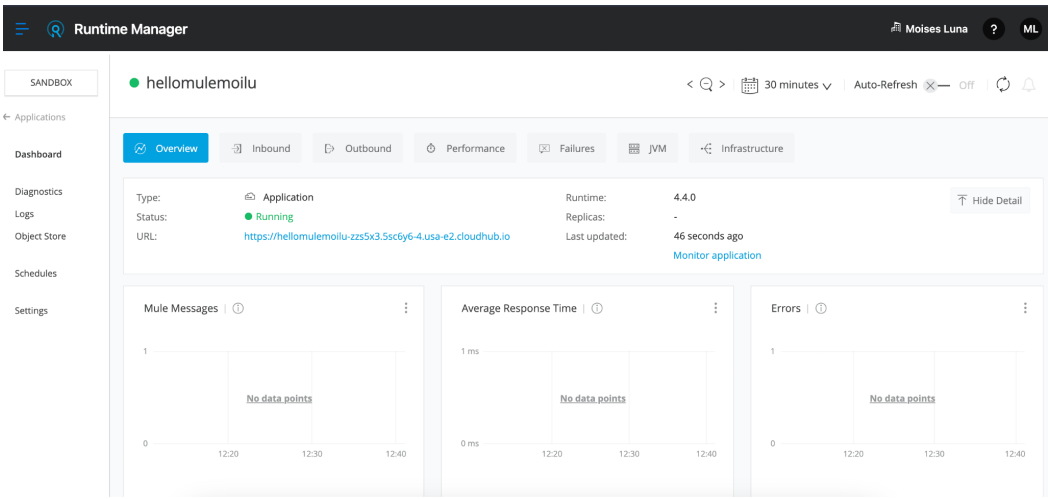
2. **Cree cuenta en anypoint para realizar el ejercicio**



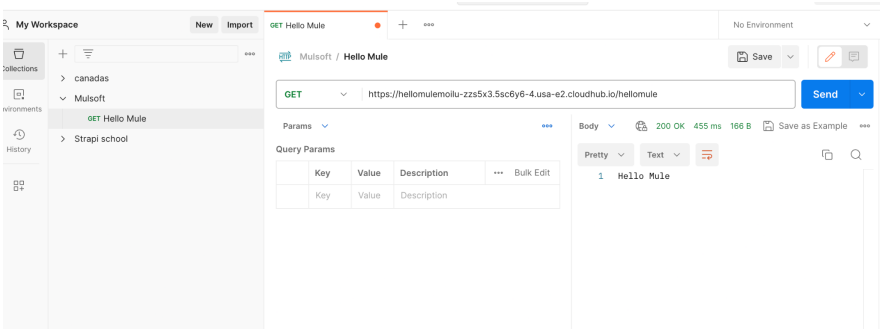
3. **Creación de aplicación Hello World en MuleSoft**



4. Despliegue de la aplicación en CloudHub

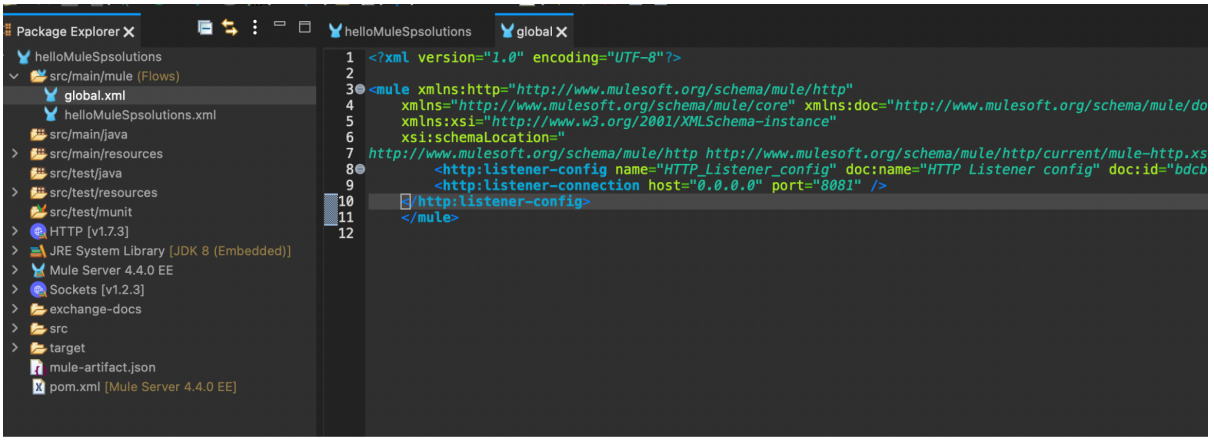


5. Uso de la aplicación en PostMan llamando al endpoint que se desplegó

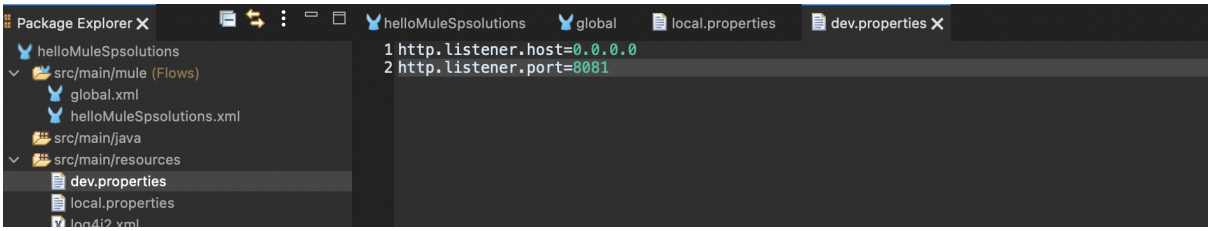


How to set up your global elements and properties

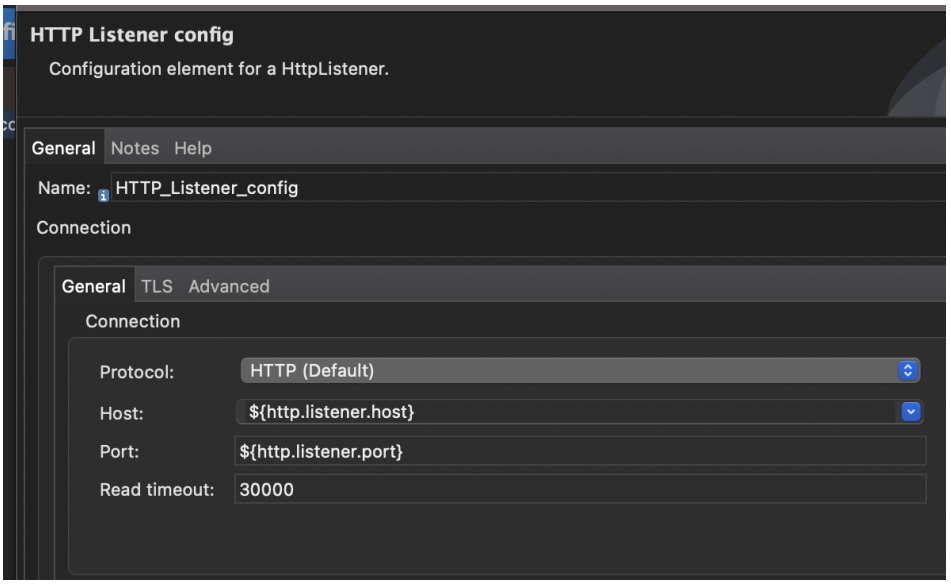
1. **Cree nuevo archivo global**



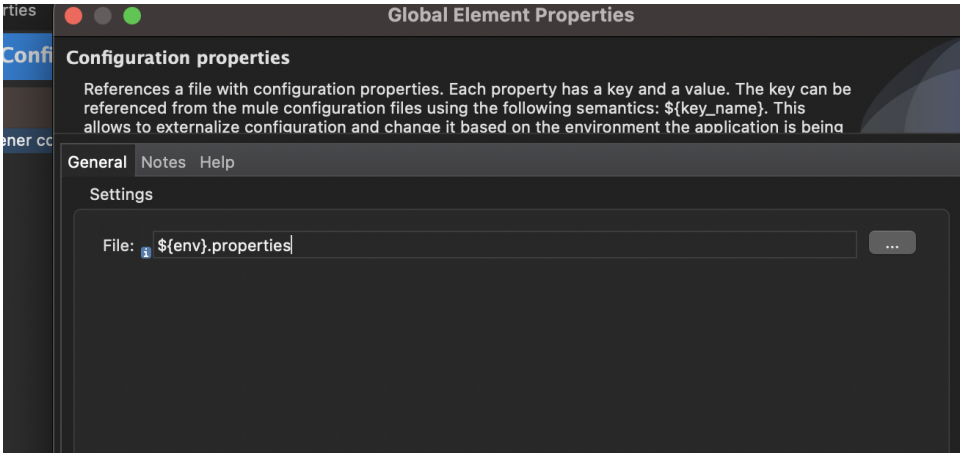
2. **Cree archivos de configuración local y dev para evitar que esté hardcoded.**



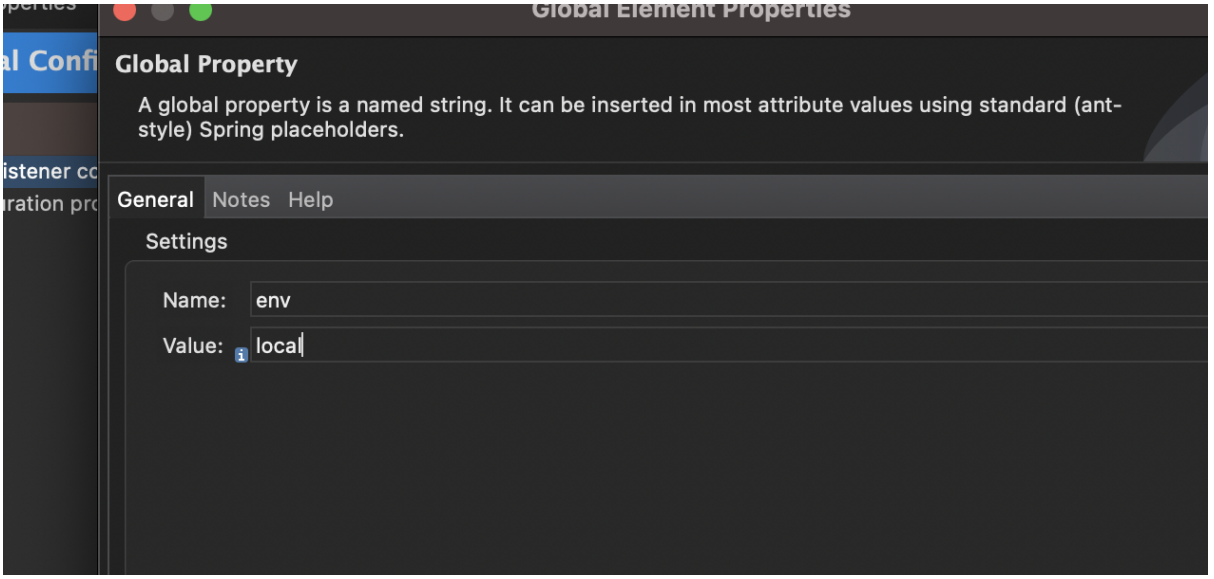
3. **Agregué en la configuración el nombre de las variables creadas**



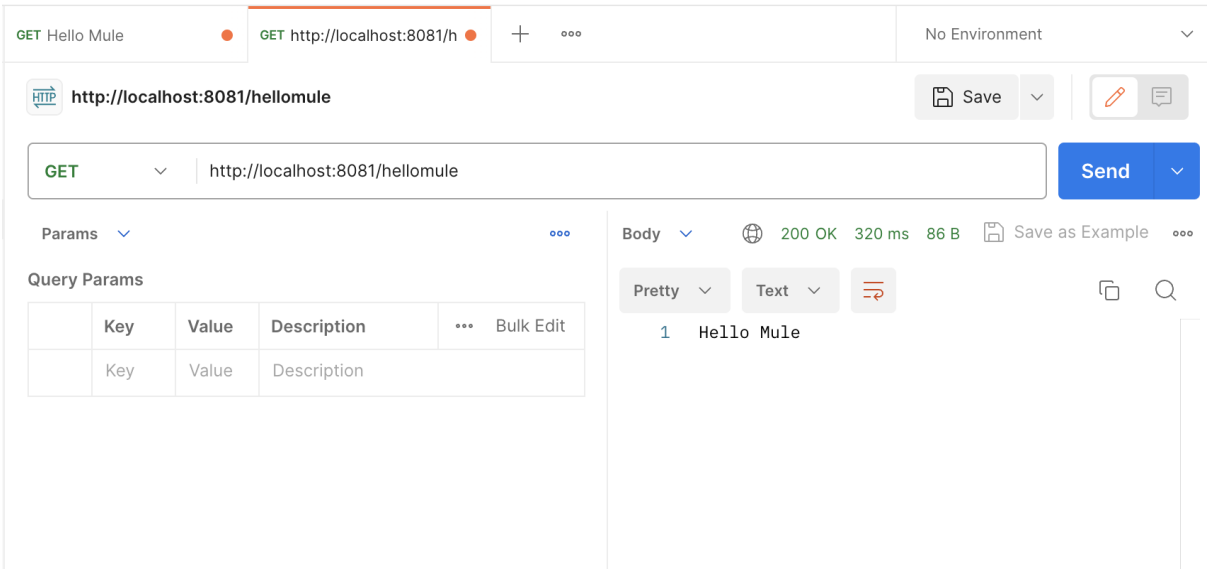
4. **Cree la configuración de las propiedades(esto sirve para seleccionar una según el ambiente de desarrollo)**



5. **Seleccioné las propiedades globales para el ambiente en base a lo creado anteriormente.**

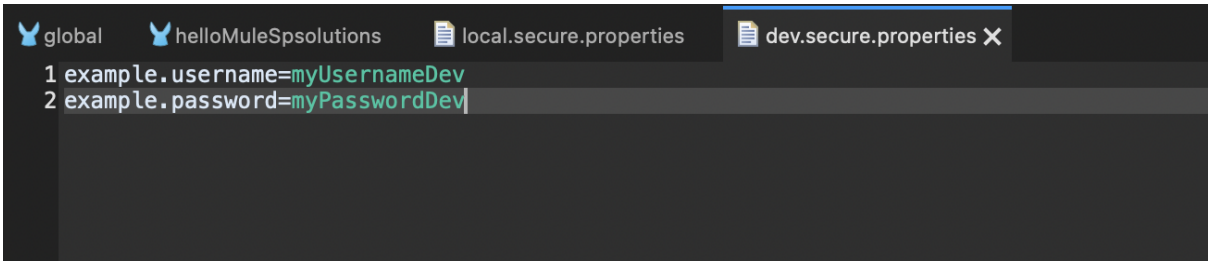


6. **Verificando que todo este correcto tras estas confgs.**

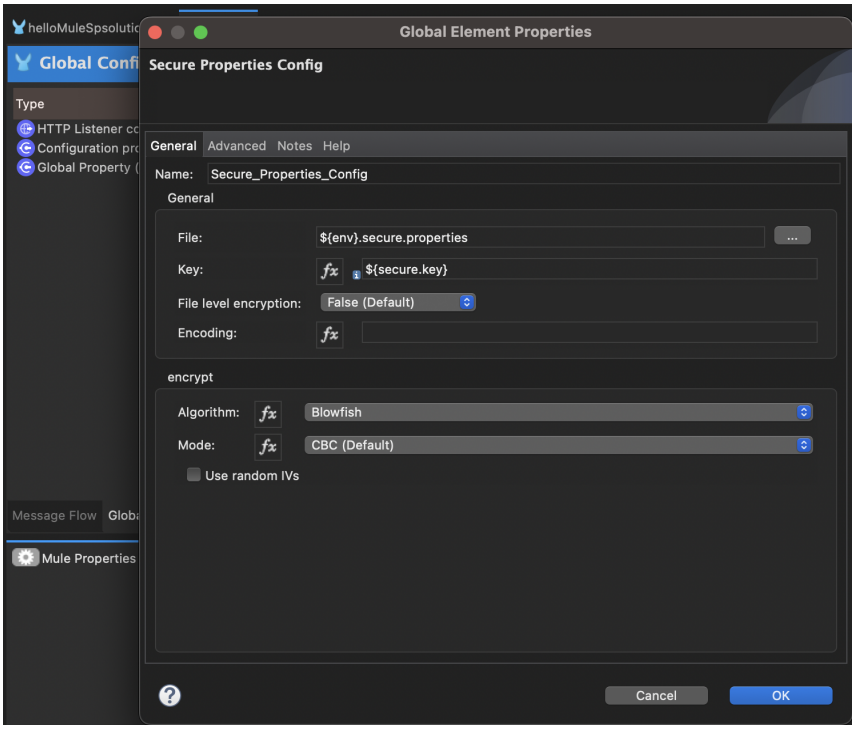


How to secure properties before deployment

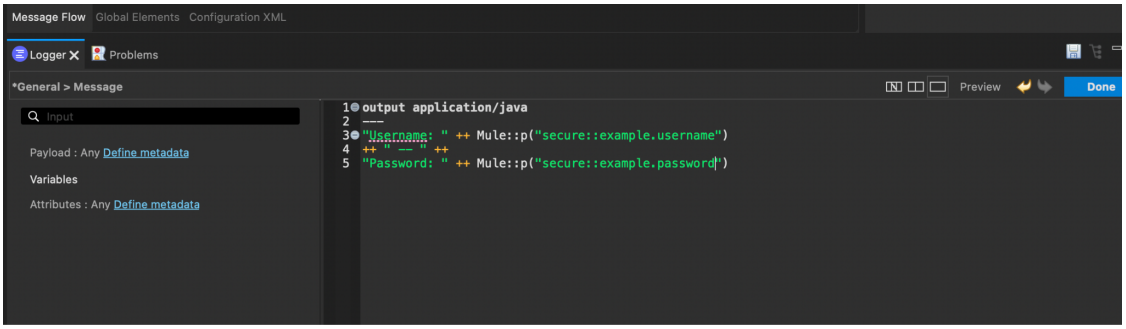
1. **Cree archivos de propiedades seguras**



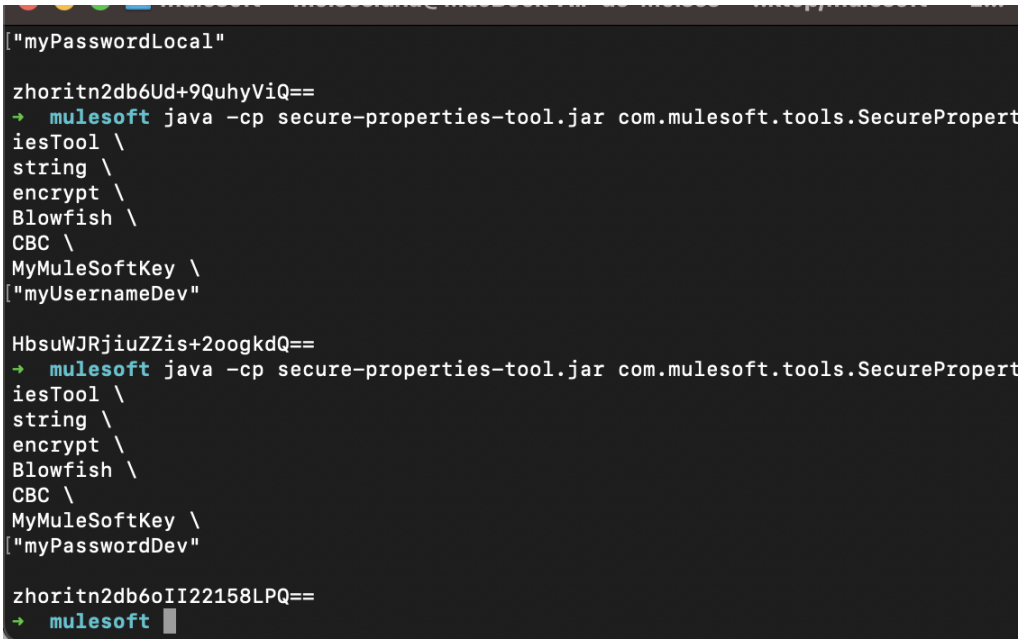
2. **Cree la configuración de propiedades seguras**

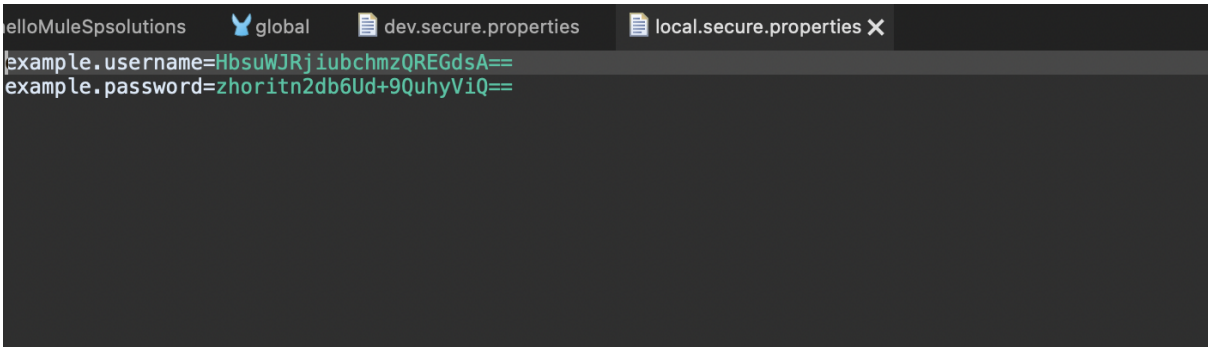


3. **Cree un script que va a leer las credenciales usando el lenguaje de programación de mulesoft**

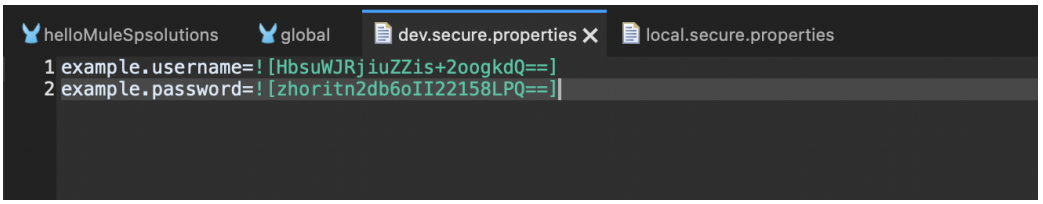


4. **Se encriptan y pegan las credenciales encriptadas**

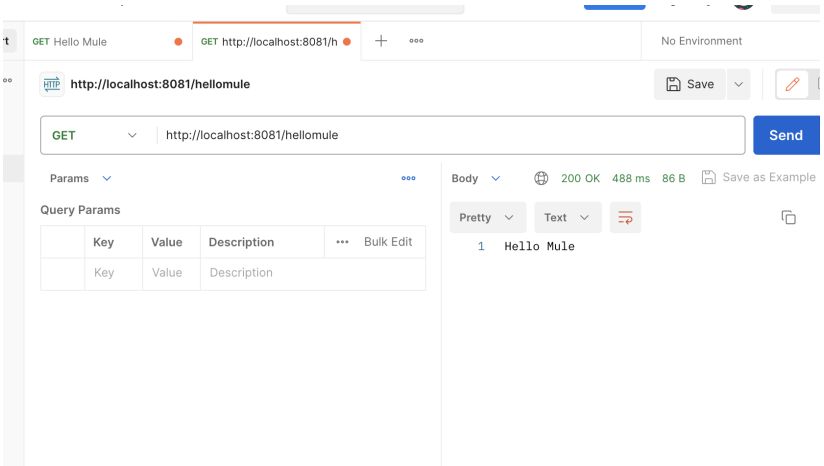




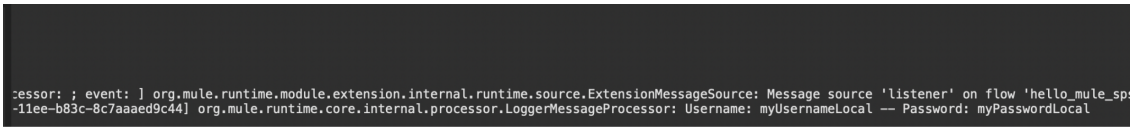
5. **Agregar la secure.key en environment**



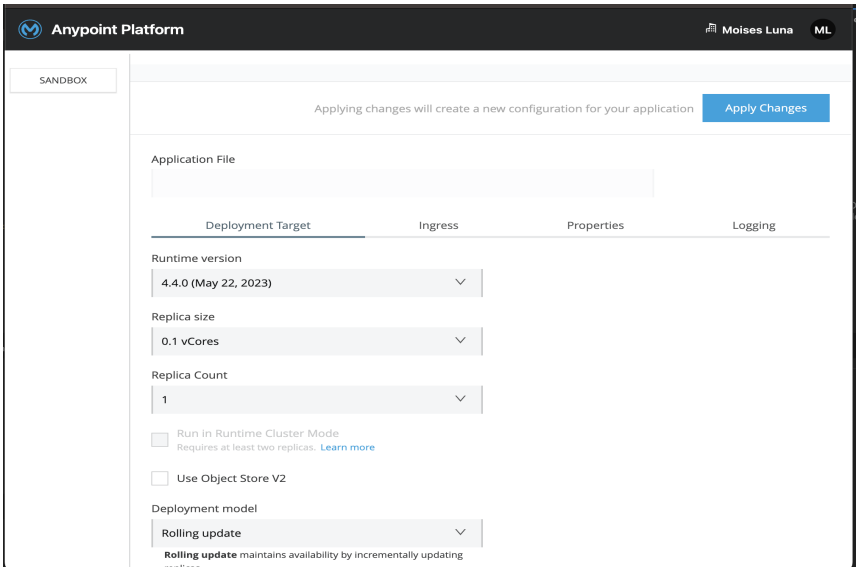
6. **Correr y probar que la aplicación siga funcionando**
   Llamada al API



Logs después de la llamada

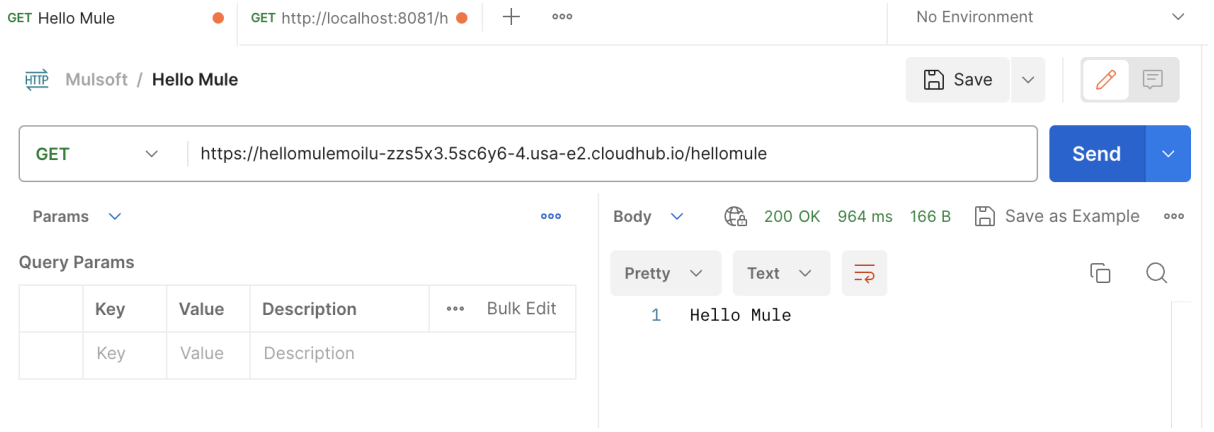


7. Despliegué de aplicación

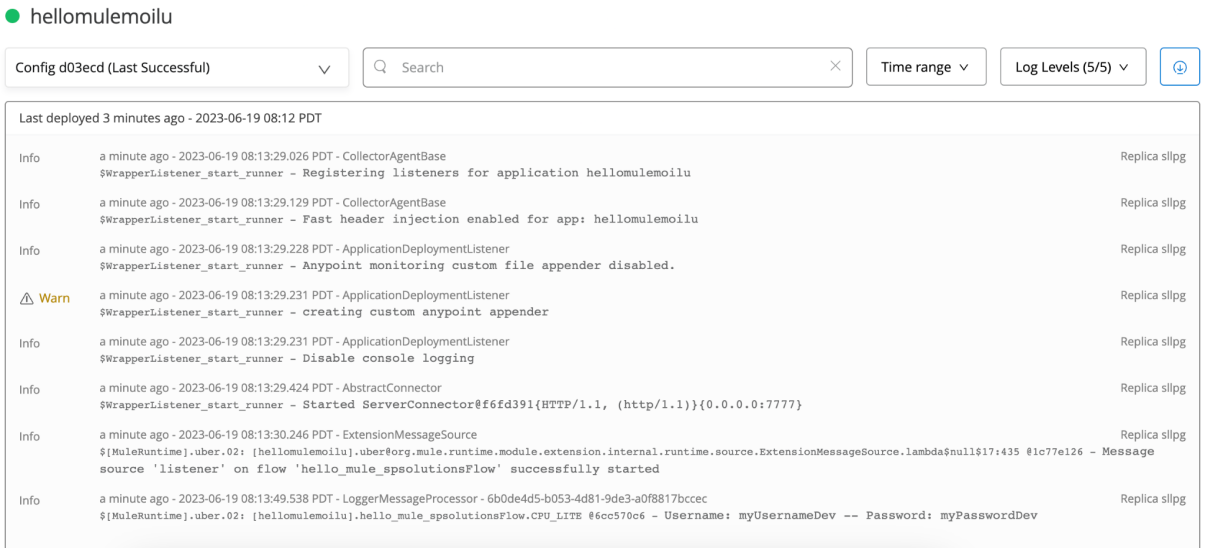


8. Escondiendo propiedades secretas agregando al mule-artifact.json(después de eso se despliega otra vez la aplicación)

9. Verificando que la aplicación siga funcionando

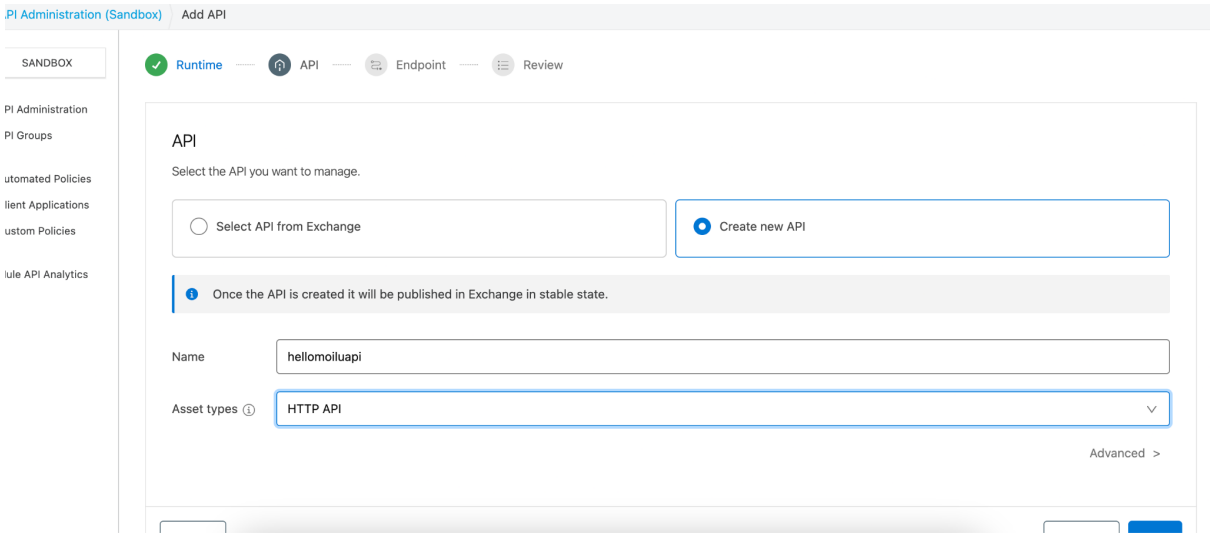


10. Todo correcto en los logs de la app

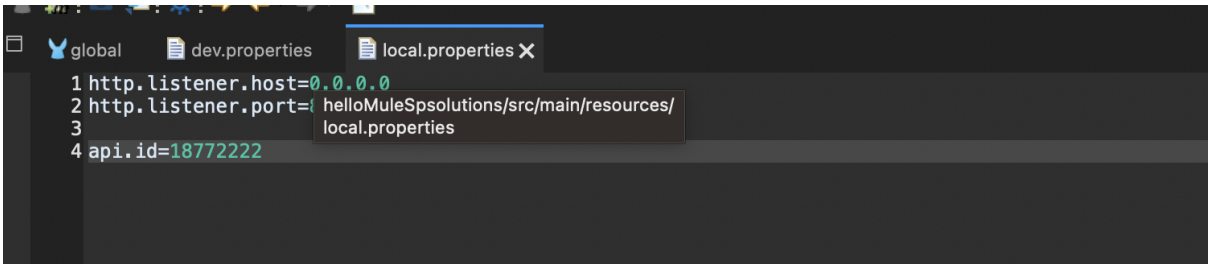


How to set up API autodiscovery

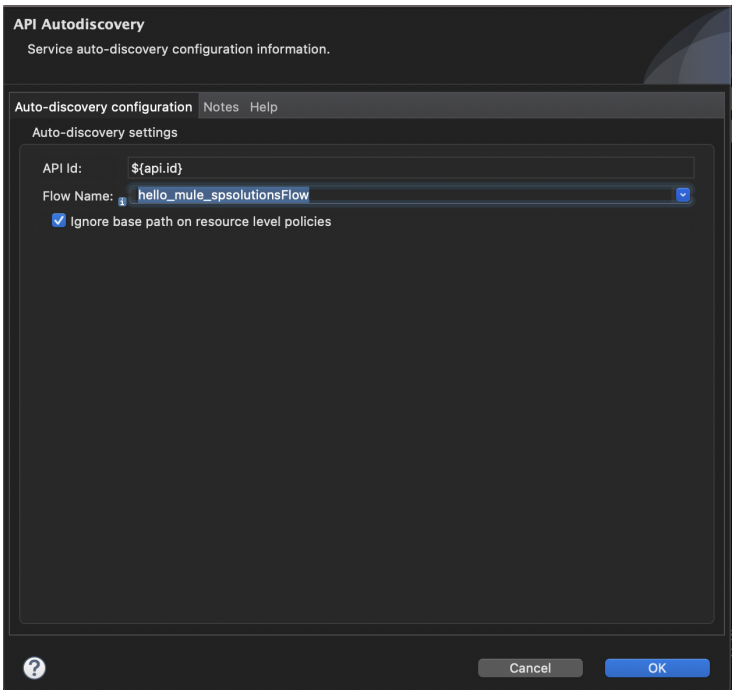
1. Creación de nueva API en AnyPoint



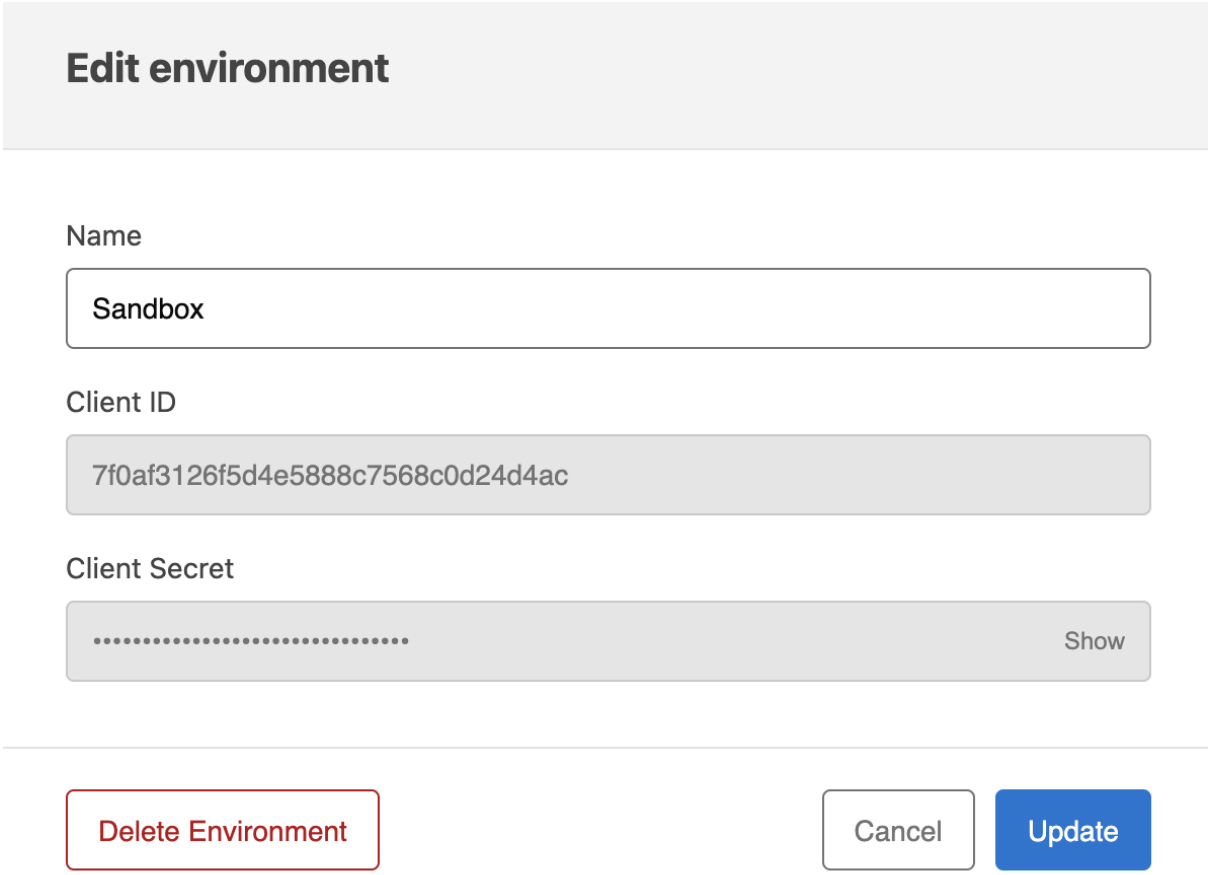
2. Agregando el app ID a la aplicación para configurar las policies(?)



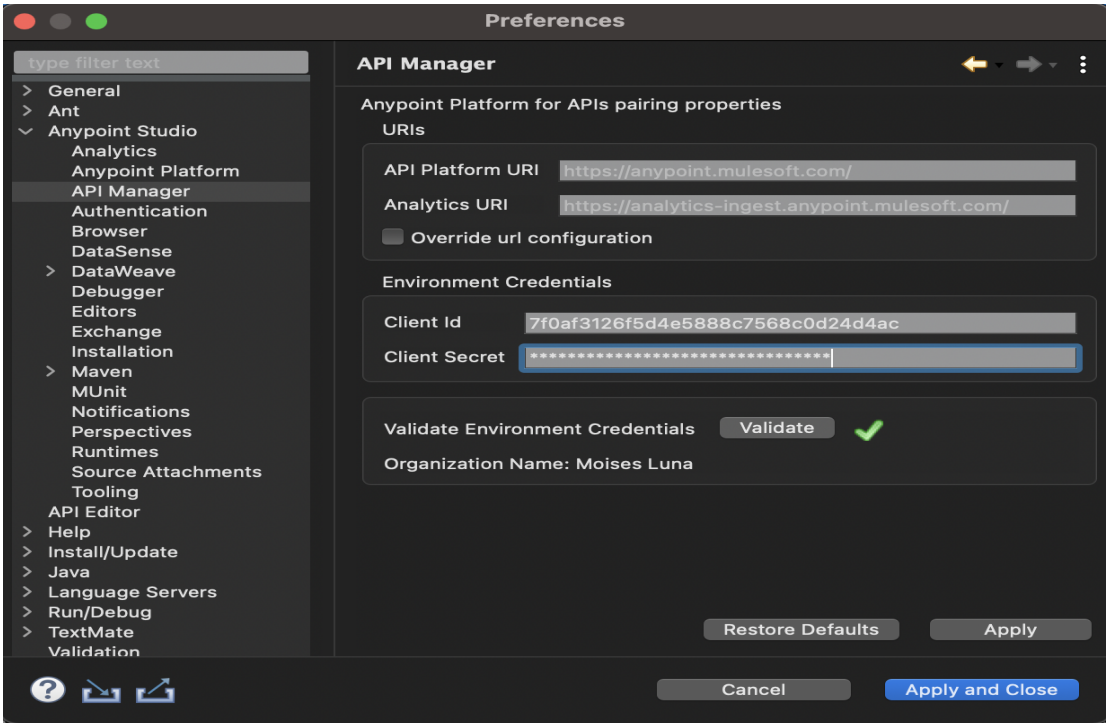
3. Cambiando configuración global para el API autodiscovery



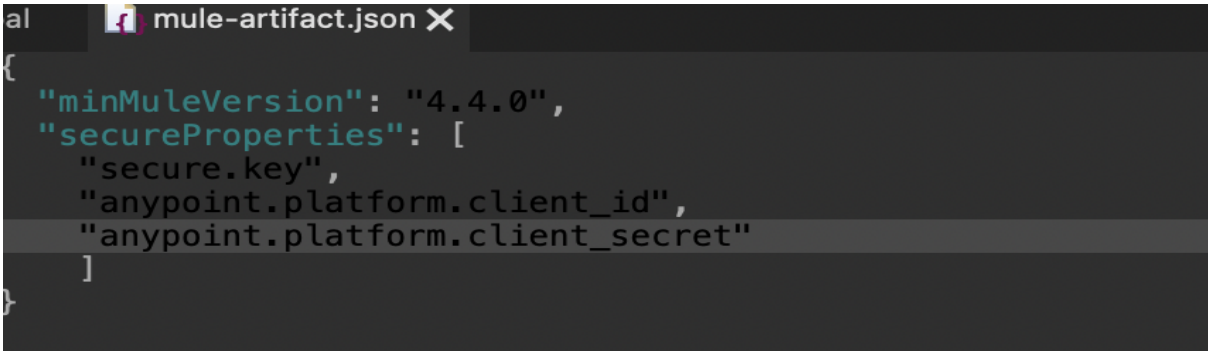
4. **Copiando ID de cliente y secret de AnyPoint Platform**



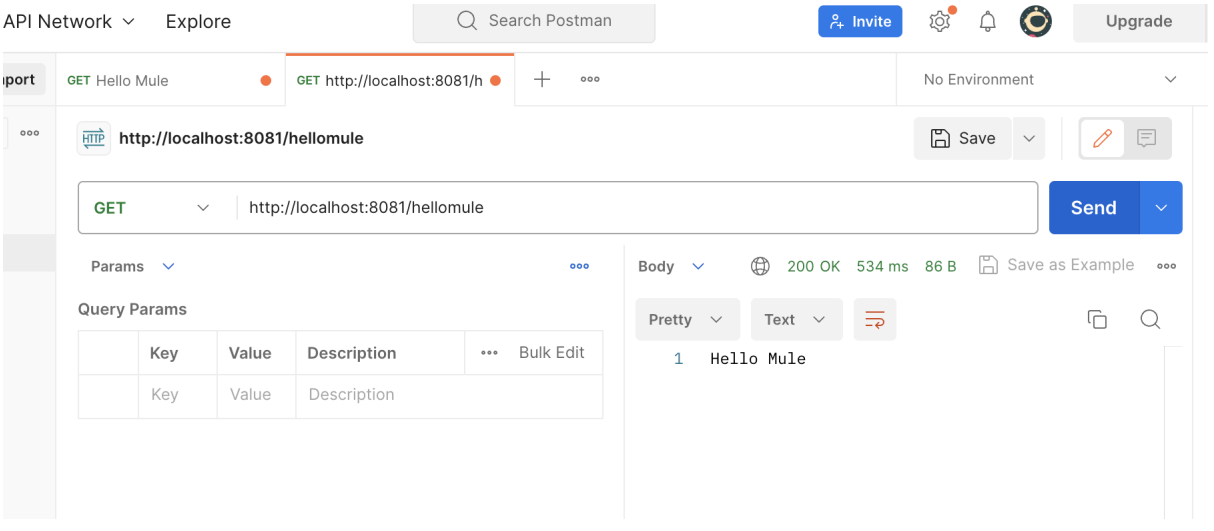
5. **Agregando configuración de la organización de platform**

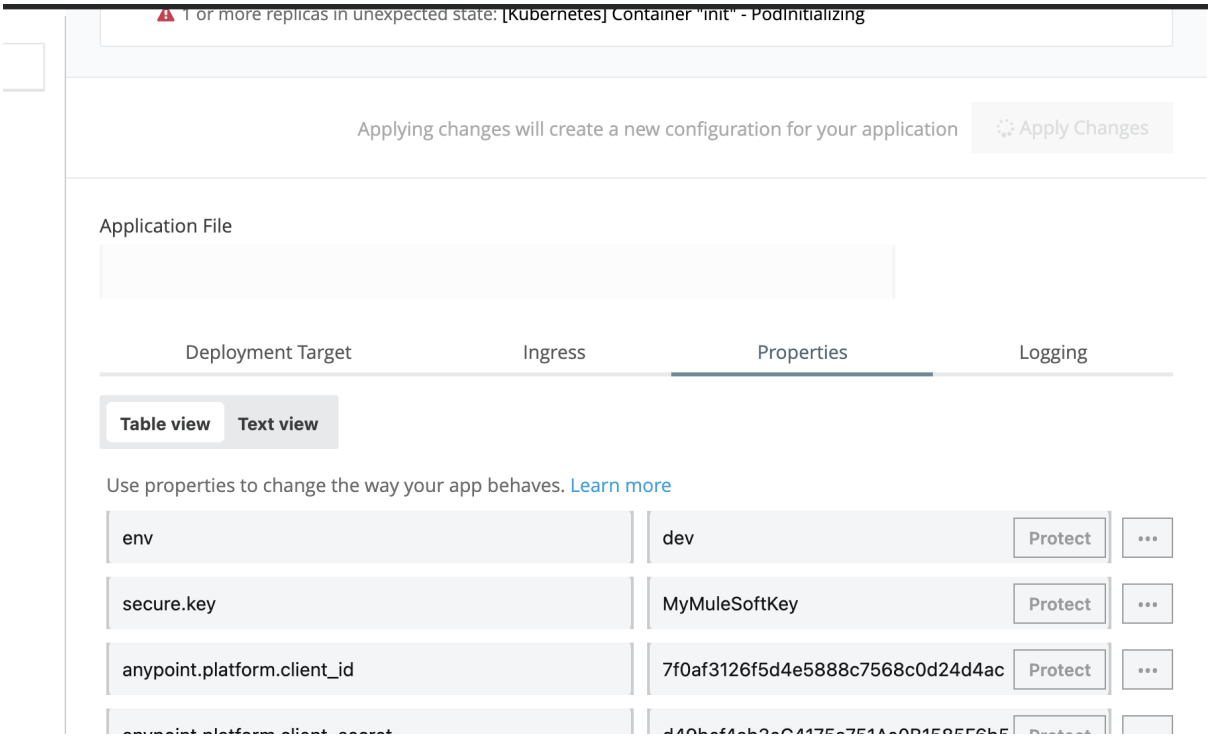


6. **Escondiendo ID y secret para que no sean visibles en la plataform**

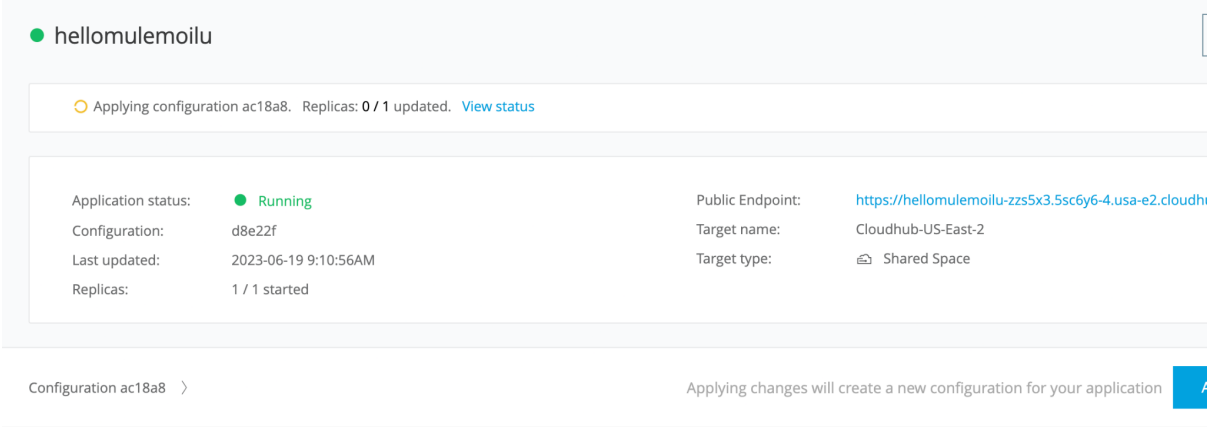


7. **Después de correr nuevamente la aplicación, reviso en postman que todo siga corriendo bien y se despliega otra vez a CloudHub**

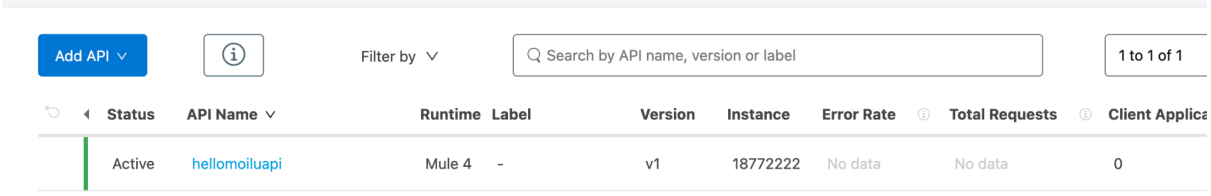




8. **Una vez que se despliega la aplicación se revisa en PostMan que todo esté bien.**

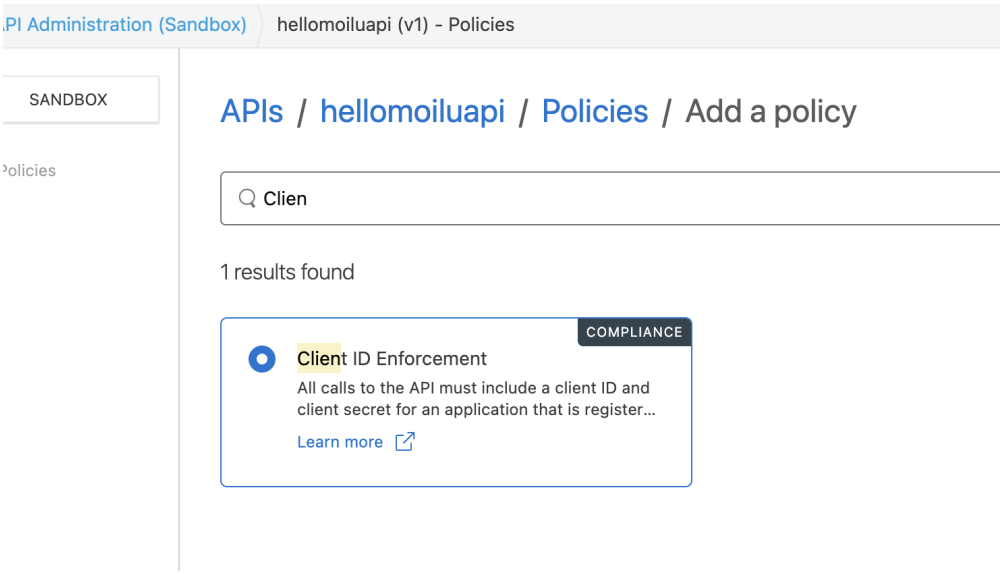


9. **Finalmente se ve en API manager que el API se encuentra activa**

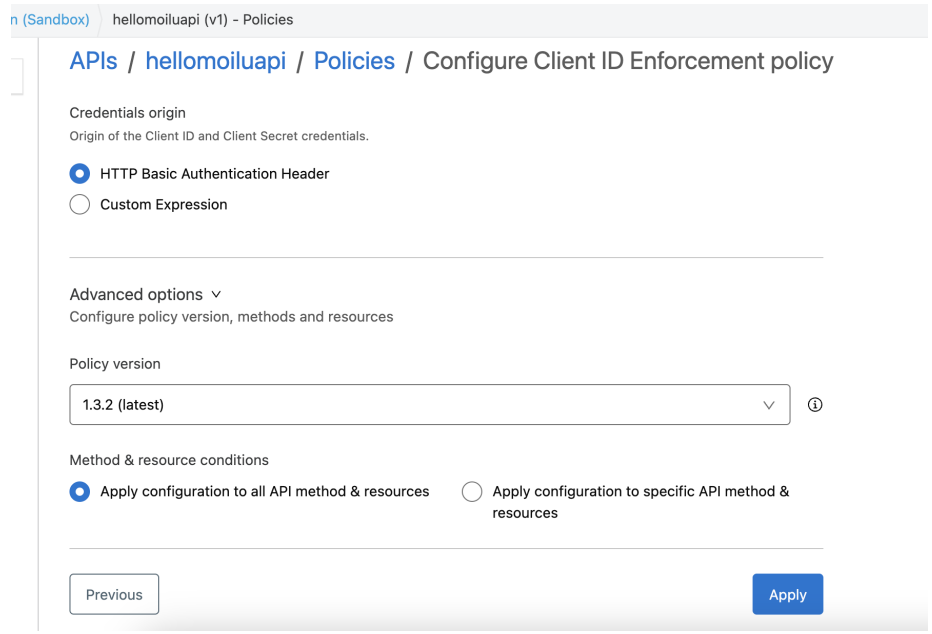


How to set up Client ID enforcement policy

1. Agregando nueva policy

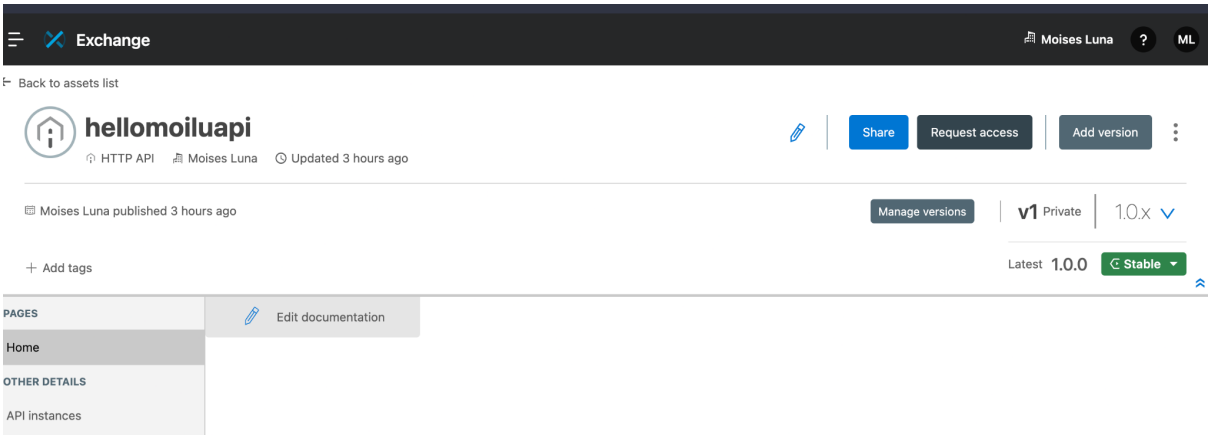


2. Configurando el Client enforcement policy

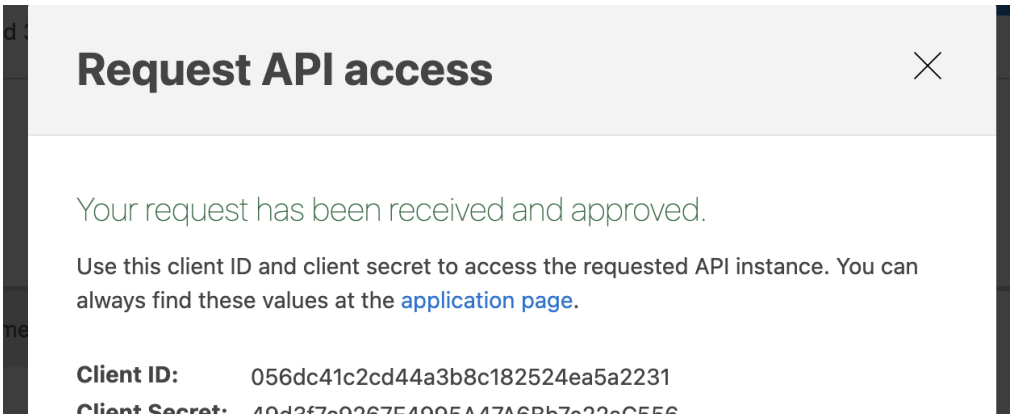


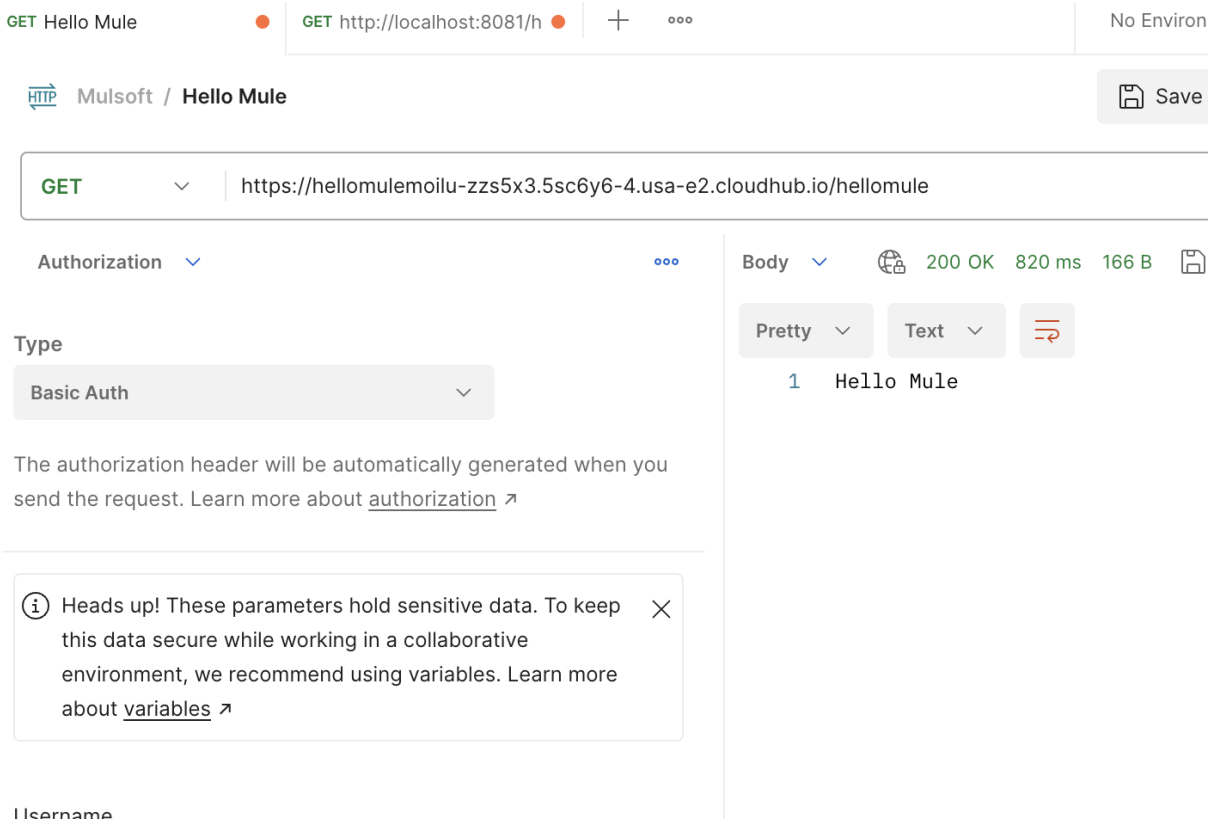
3. Verifico que la configuración este correcta con postman. El error de autorizacion indica que sí

4. **Ir a la sección Exchange para obtener los valores necesarios de autenticación del API para que sea accesible desde PostMan**



5. **Extraer llaves de acceso y agregarlas a PostMan**





Best practices to design your first API specification

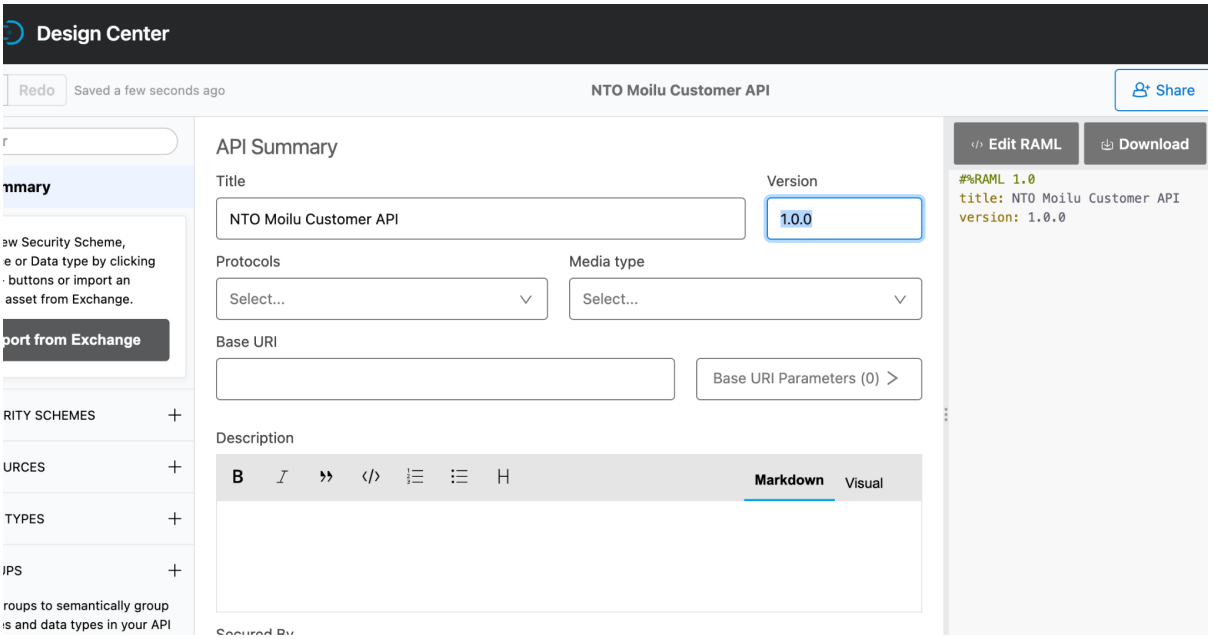
Este tutorial explica que lo que son las especificaciones. En resúmen es la planeación de los estándares que debe seguir un API que vas a crear.

Algunos de estos estándares son:

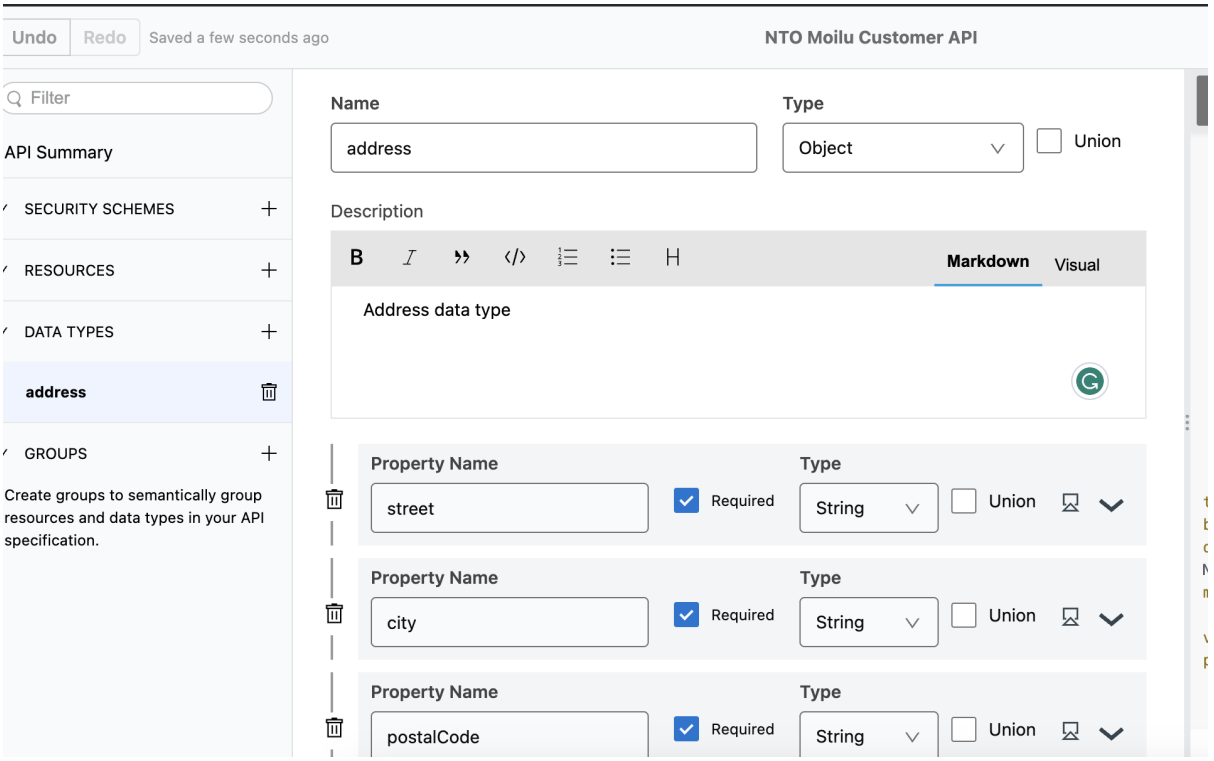
- Nombrar correctamente los recursos que va a devolver el API. Usar adjetivos en vez de verbos(/contacts)
- Definir el nombre y url base correctamente para que defina la industria correcta.
- Tener claro que métodos HTTP va a soportar.
- Mantener los query parameters en snake_case.
- Definir si el API tendrá funcionalidades de ordenamiento, filtración y paginación.

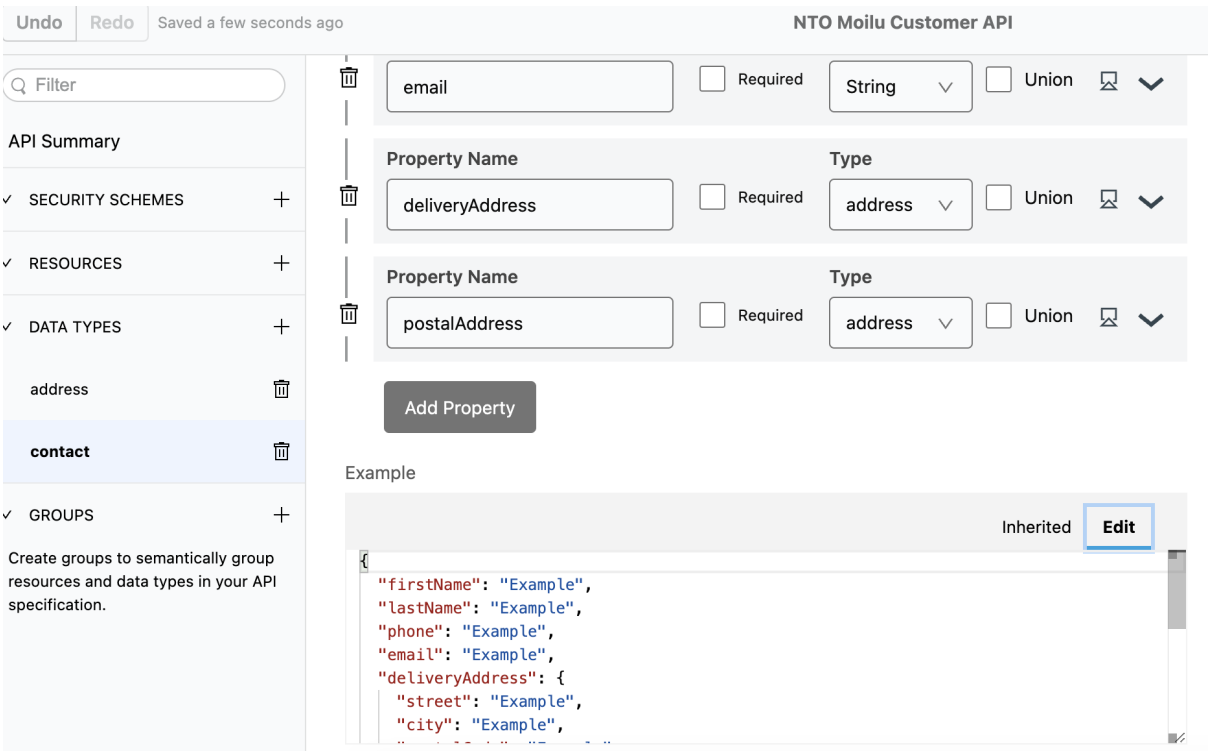
Build your first API specification

1. **Creación de nueva especificación en el Design Center siguiendo la opción Guide Me**

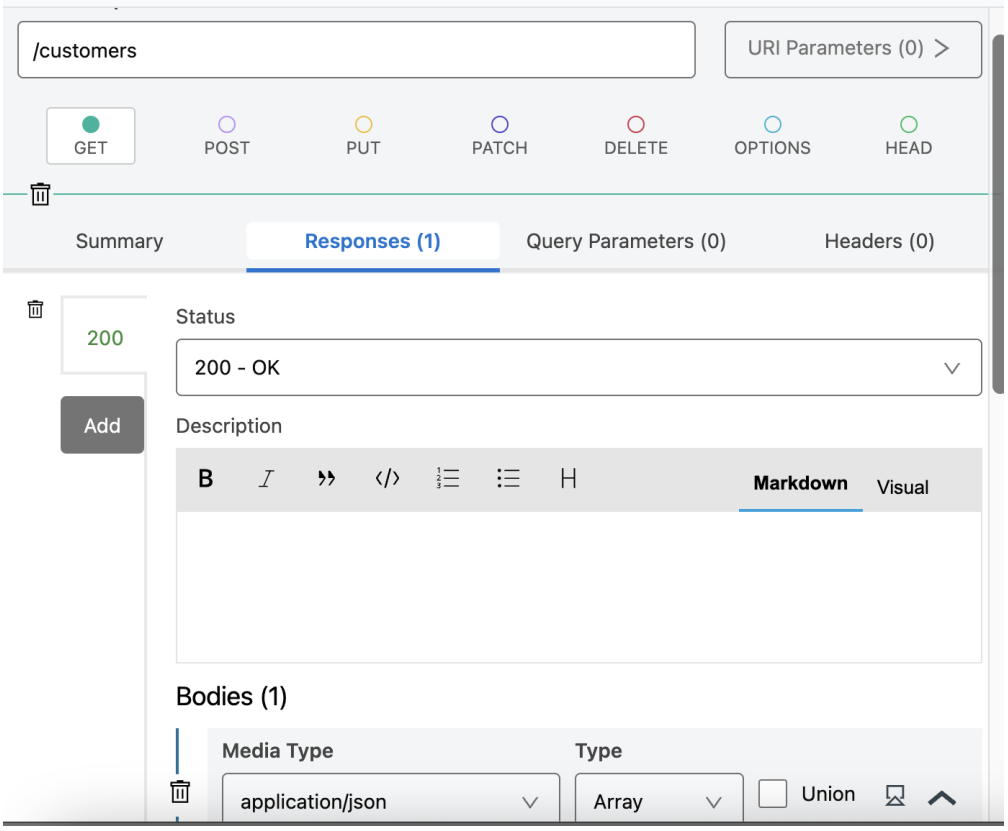


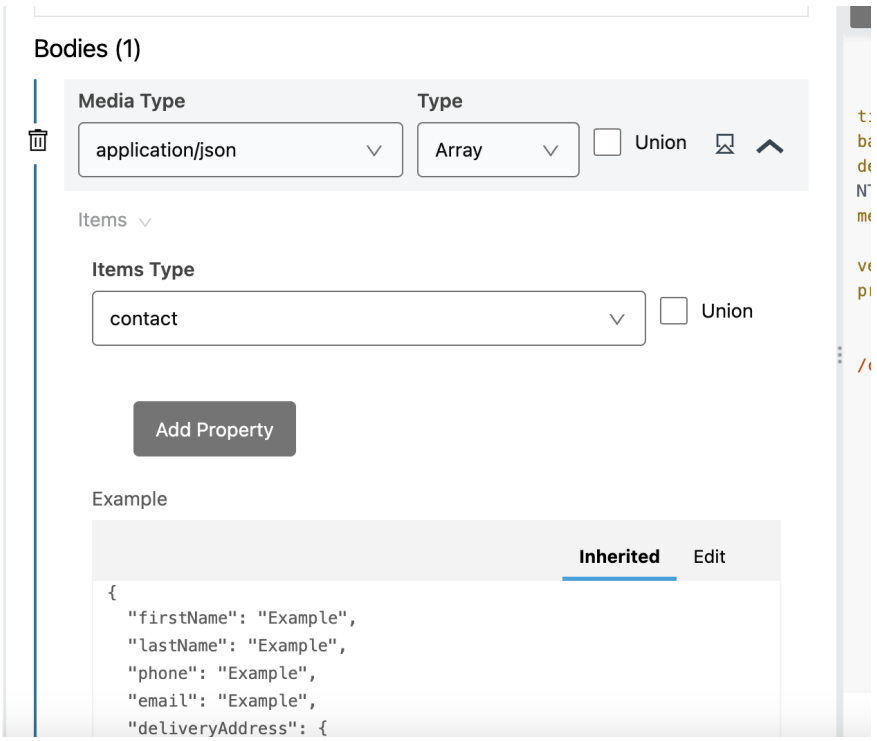
2. **Agregado de data types al especificación del API**

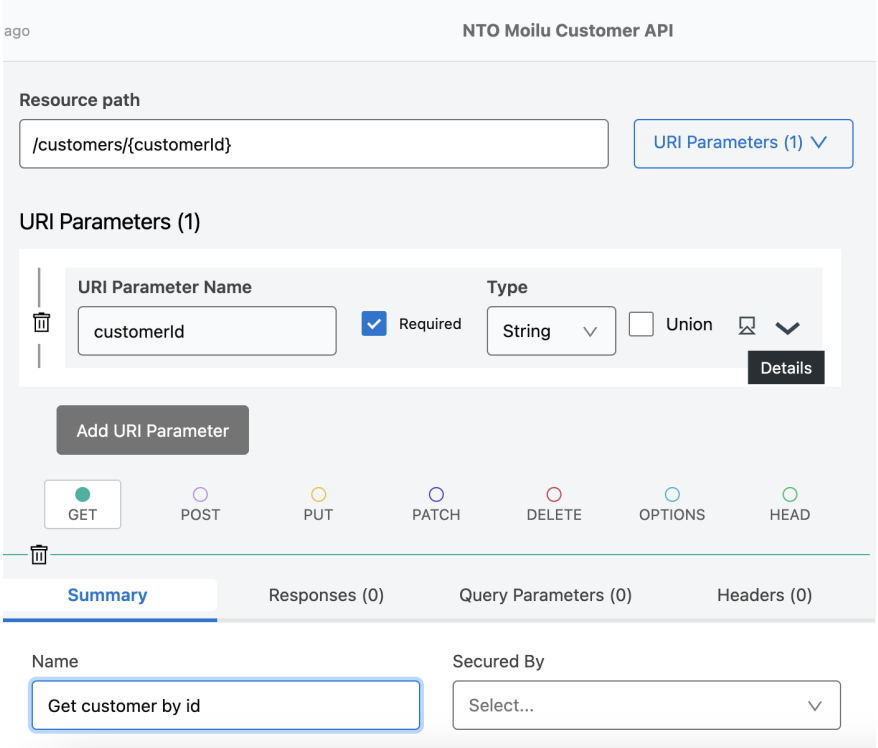


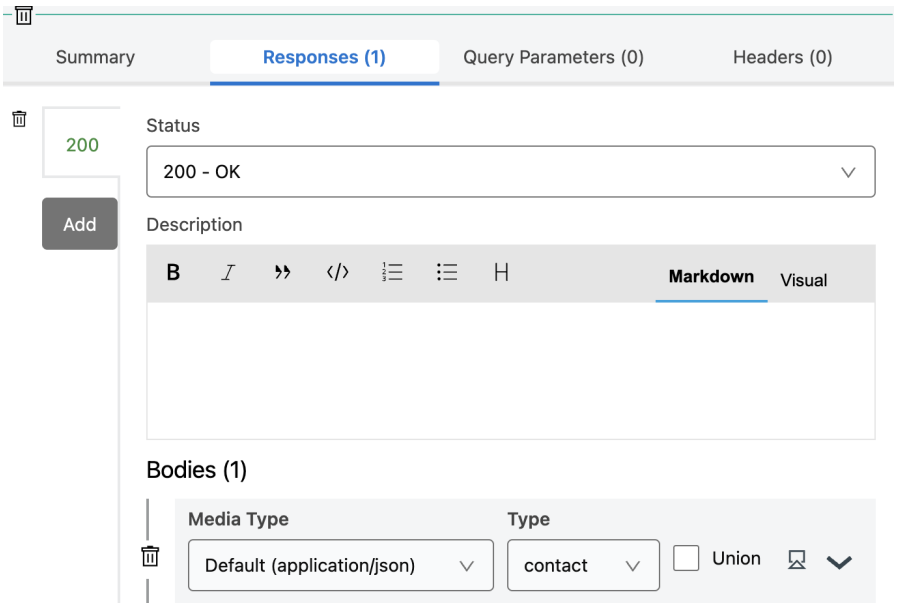


3. Agregando un recurso nuevo a la especificación

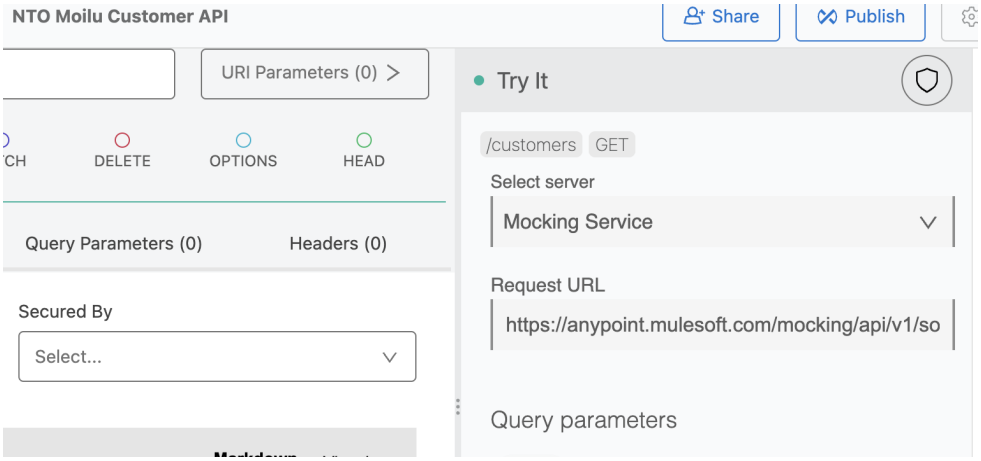



Bodies (1)
Media Type
application/json
Type
Array
Union
Items
Items Type
contact
Union
Add Property
Example
Inherited
Edit
{
"firstName": "Example",
"lastName": "Example",
"phone": "Example",
"email": "Example",
"deliveryAddress": {


ago
NTO Moilu Customer API
Resource path
/customers/{customerId}
URI Parameters (1)
URI Parameters (1)
URI Parameter Name
customerId
Required
Type
String
Union
Details
Add URI Parameter
GET
POST
PUT
PATCH
DELETE
OPTIONS
HEAD
Summary
Responses (0)
Query Parameters (0)
Headers (0)
Name
Get customer by id
Secured By
Select...



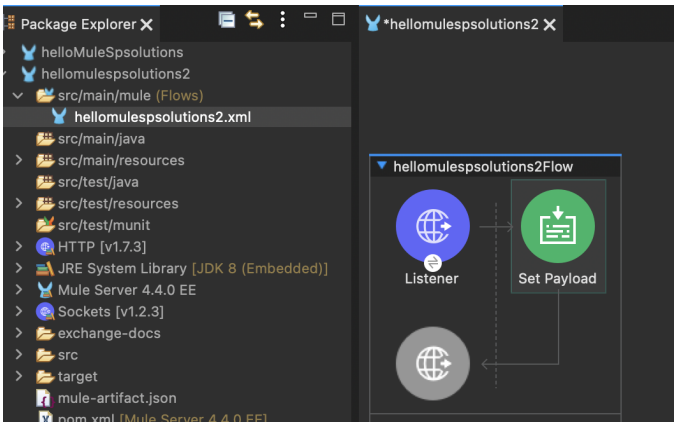
4. **Poniendo a prueba el API con el mocking service**



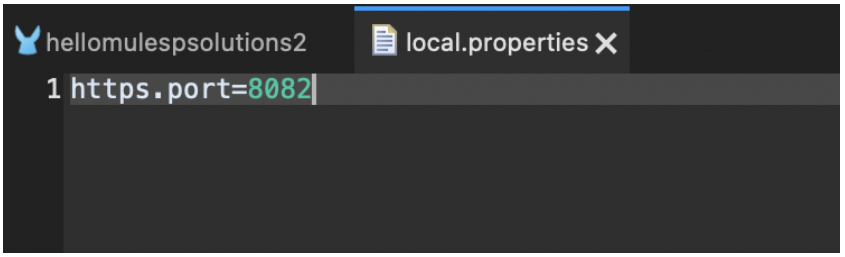
5. **Especificación en Exchange**
   https://anypoint.mulesoft.com/exchange/f4b04931-7179-4b7c-8a4a-7cc8183edf3e/nto-moilu-customer-api/minor/1.0/

How to configure an HTTPS endpoint in Anypoint Studio

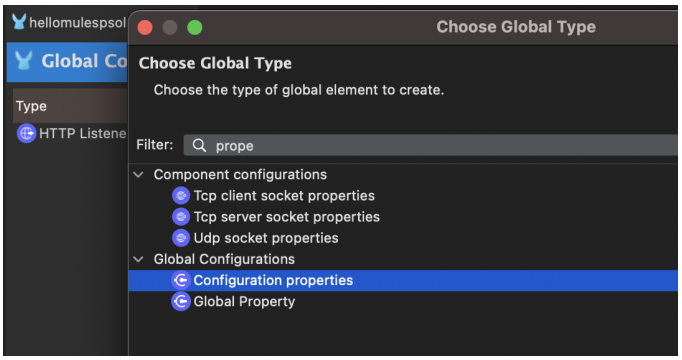
1. **Creación de nuevo proyecto**



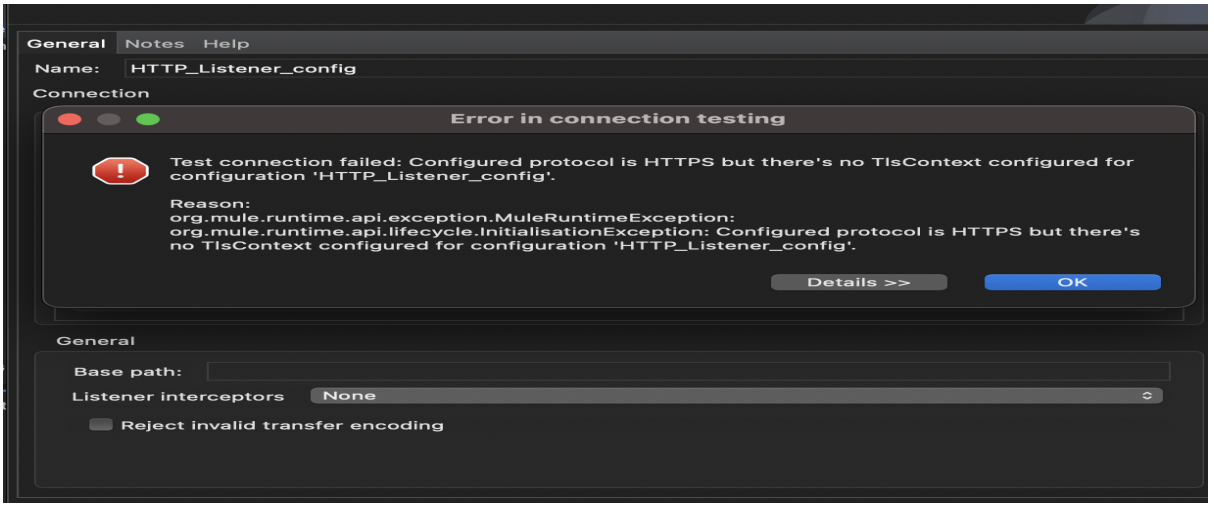
2. **Agregado archivo de propiedades en recursos con configuración de puerto https**



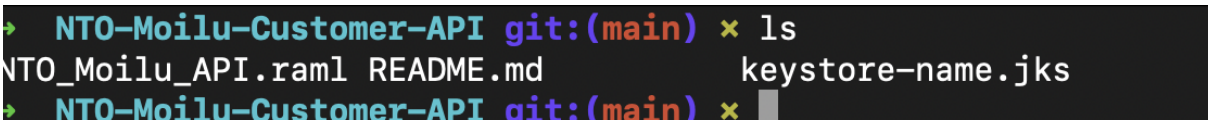
3. **Agregando configuración de propiedades**



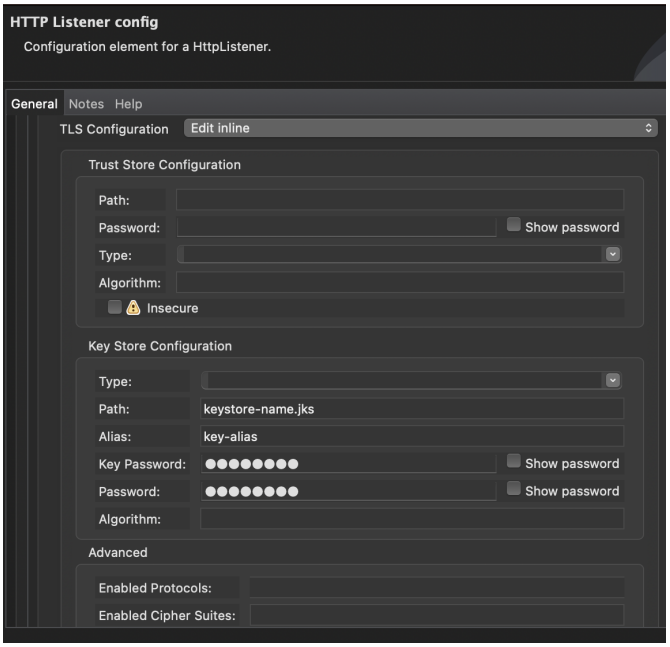
4. **Se selecciona el puerto y HTTPS, la primer prueba de conexión debe fallar**



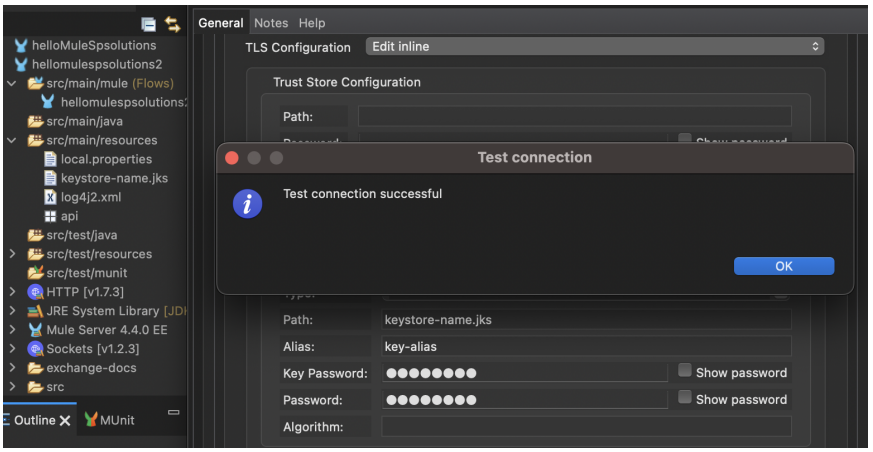
5. **Se debe arreglar la configuración TLS para que no falle la conexión. Para esto primero se agrega un keystore**



6. **Agregado de configuración del keystore**



7. **Prueba de conexión**



How to configure an HTTPS endpoint in Anypoint Studio

Pendiente