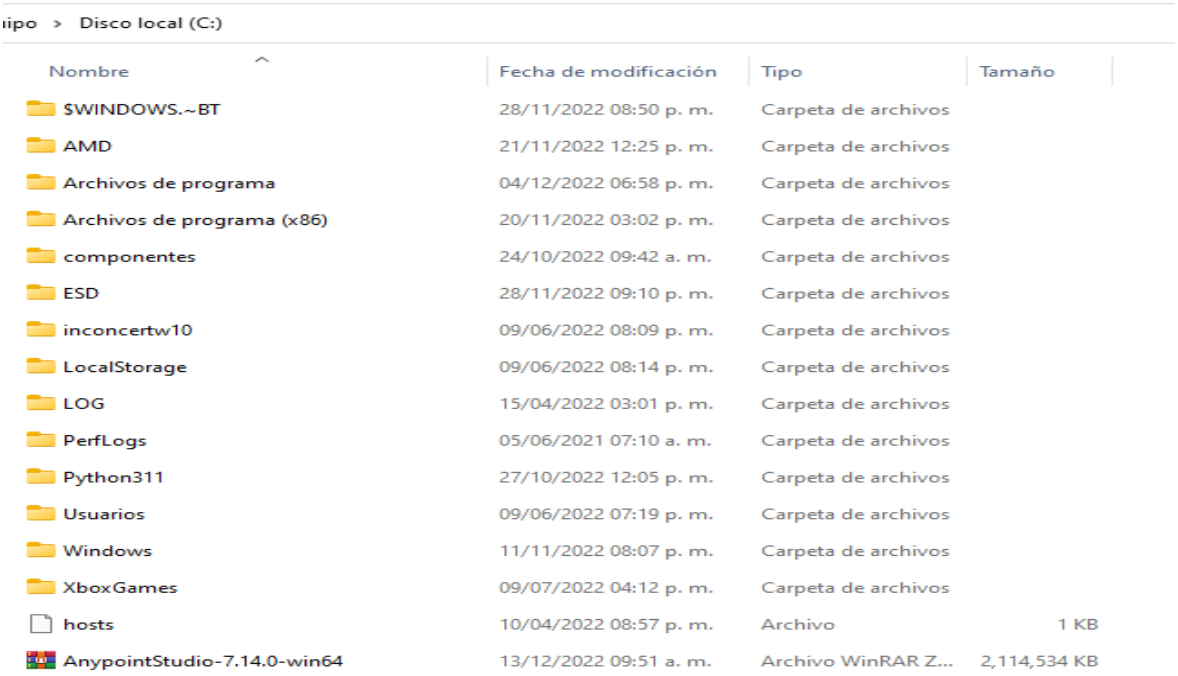
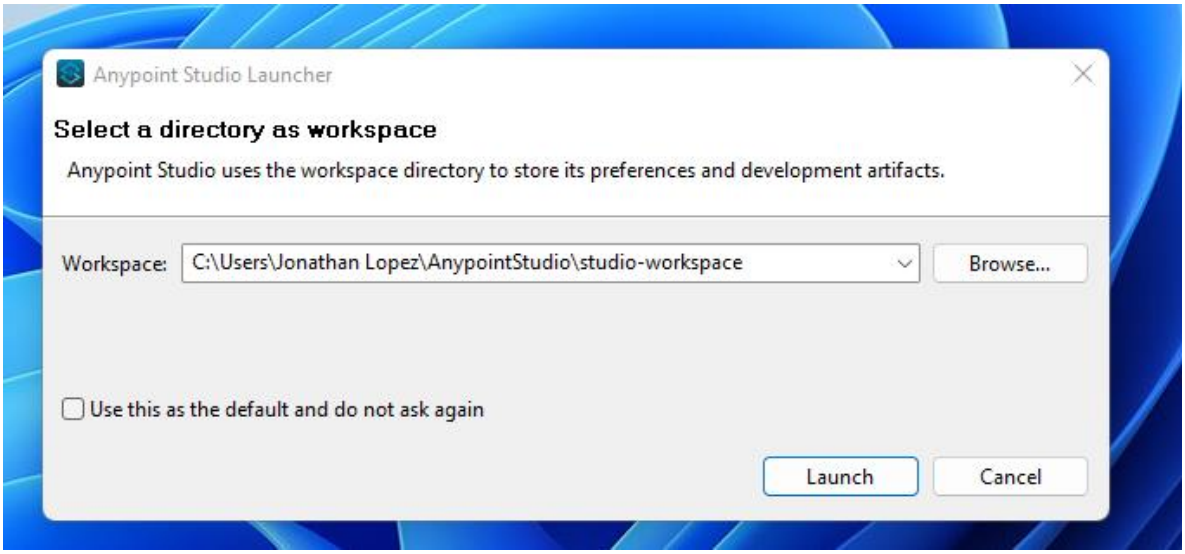
Prueba SPS MuleSoft

Instalación de IDE.

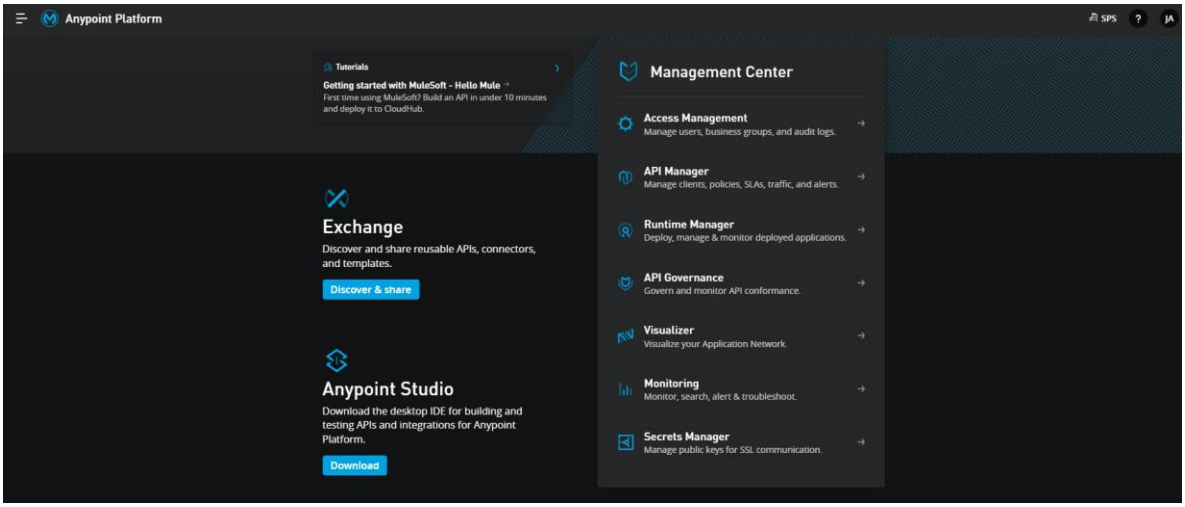
Paso 1: Descarga y descomprimir en carpeta raíz C:



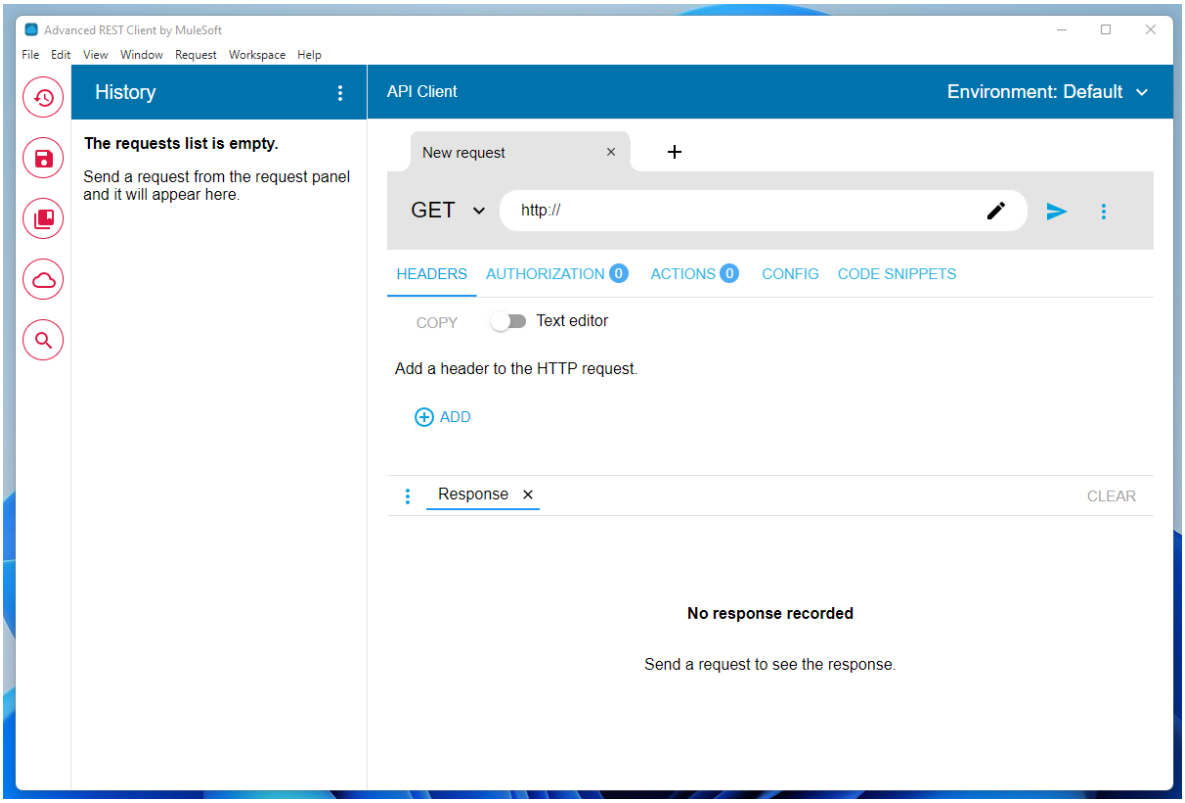
Paso 2: Seleccionar carpeta de trabajo



Registro en plataforma: Anypoint Platform.

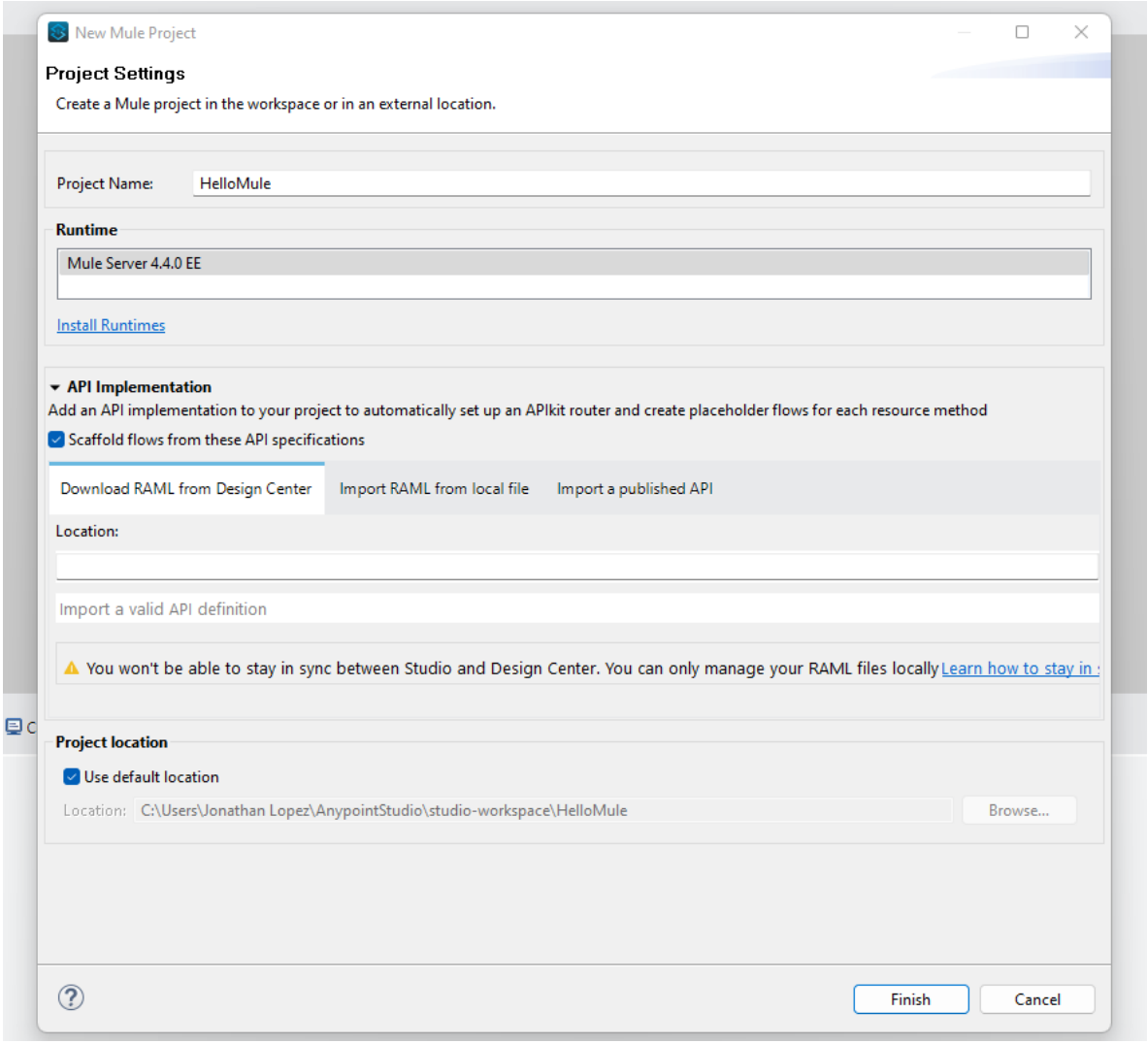


Instalación de REST CLIENT.

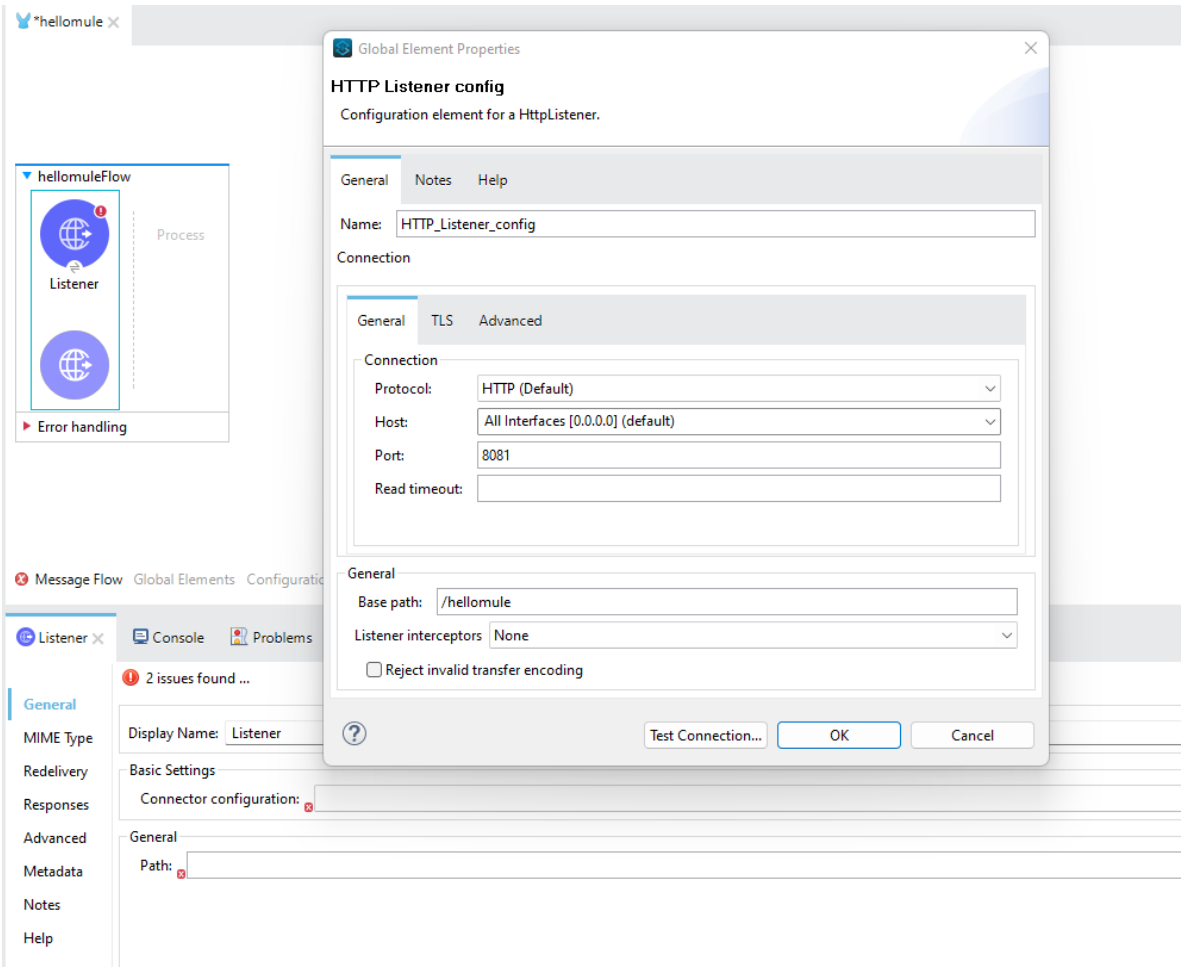


1-. Creación de APP en MULE (HELLO MULE)

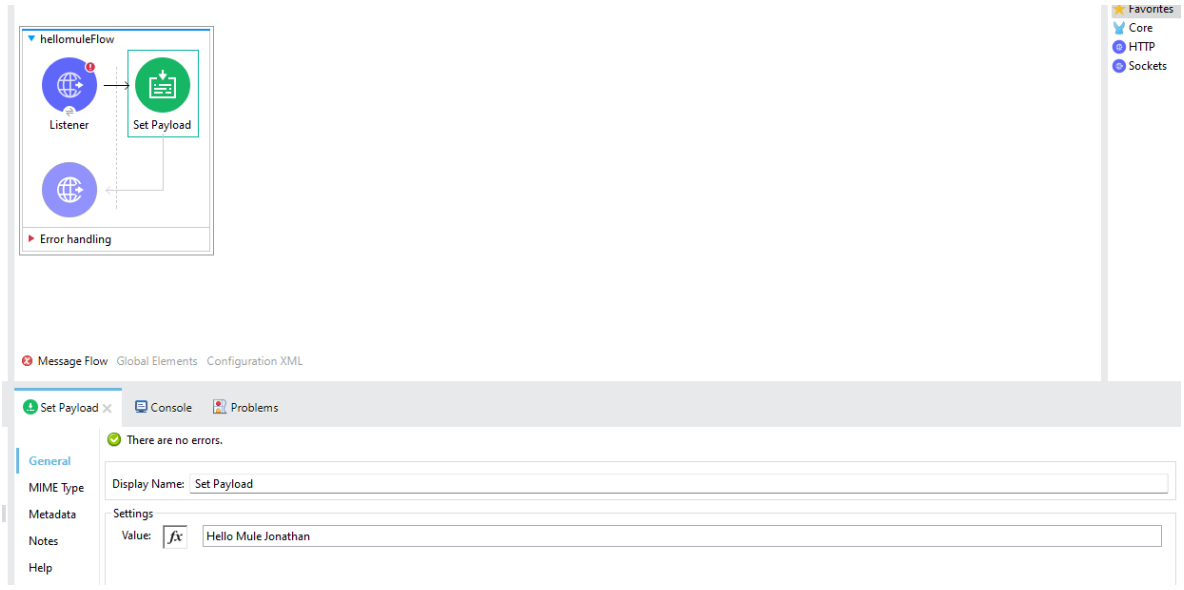
Paso 1: Clic en pestaña File, new Project, New Mule Project.



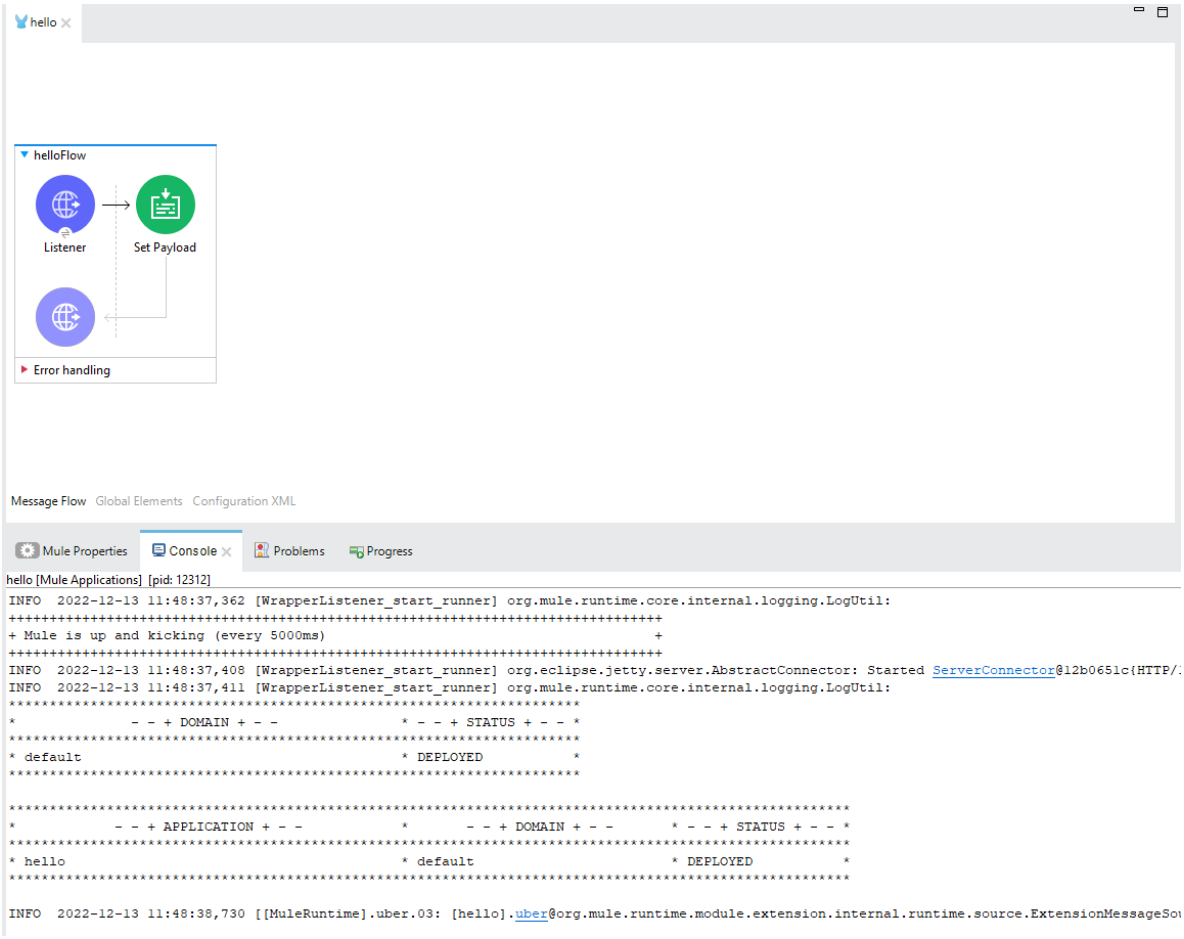
Paso 2: Se agrega http listener y se configura con el path “/hellomule”, en el puerto por default.

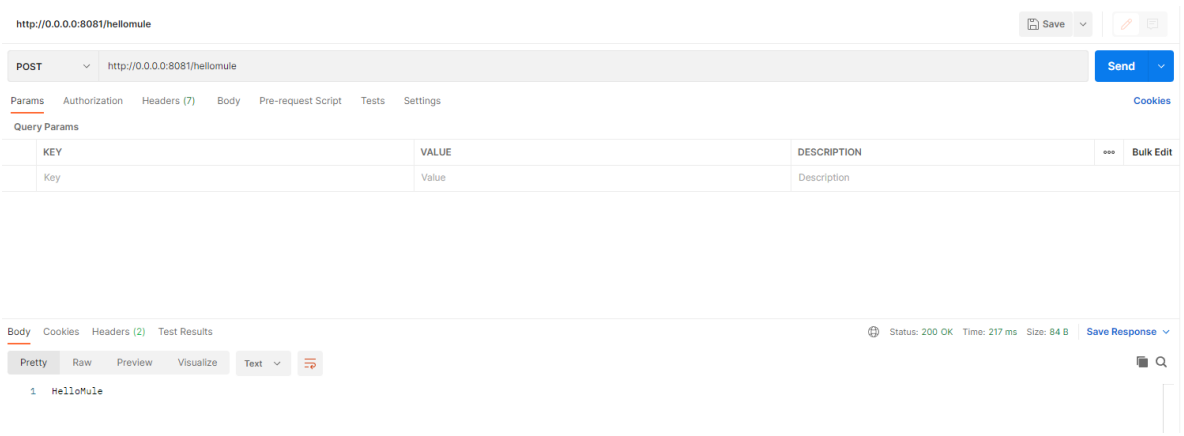


Paso 3: Se agrega método Set Payload y se establece la cadena “Hello Mule” y se guarda el archivo (ctrl +S, archivo – save).

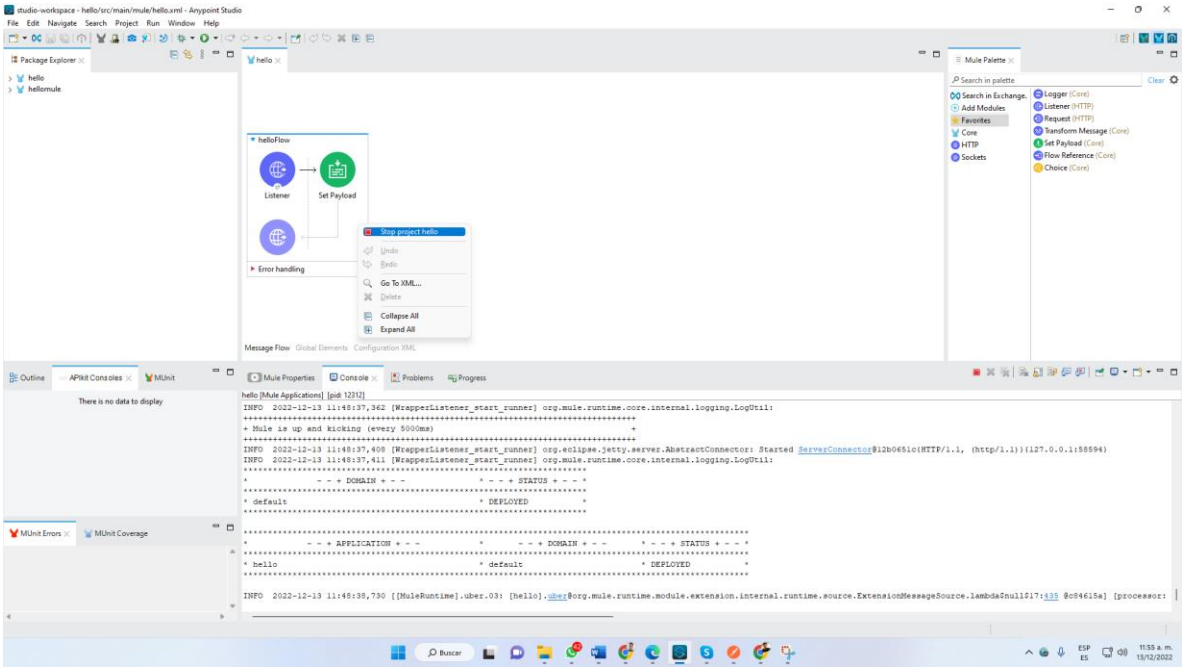


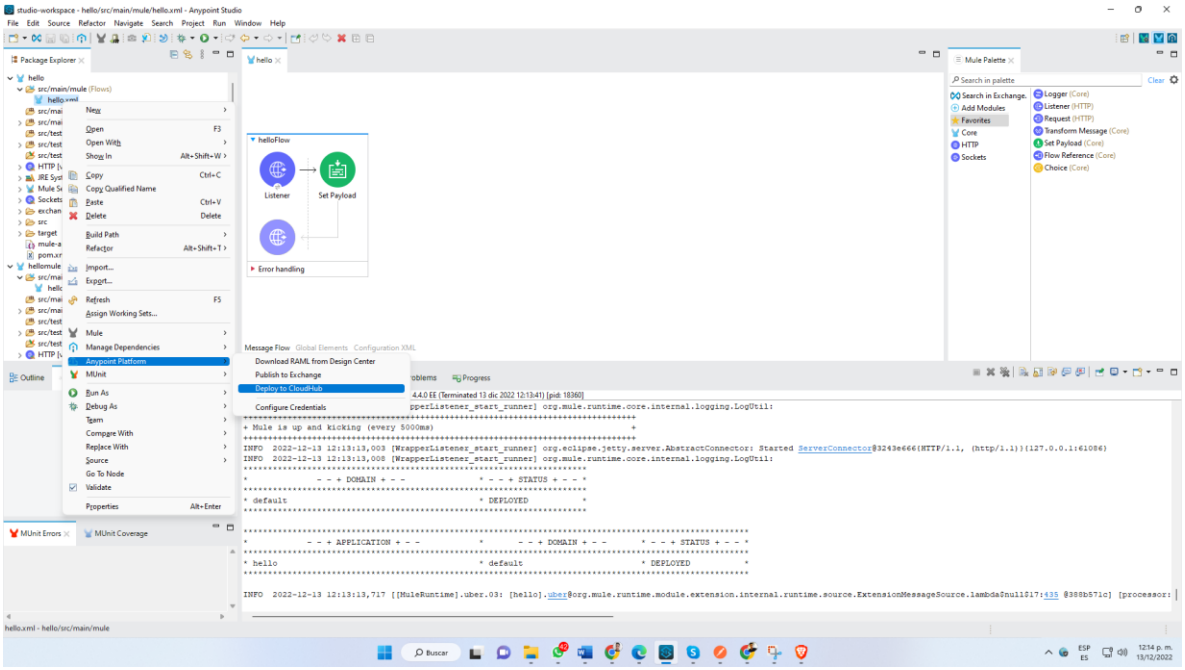
Paso 4: Se da clic izquierdo y se da clic en Run Project “nombreProjecto” y se hace consulta en postman para comprobar el servicio.

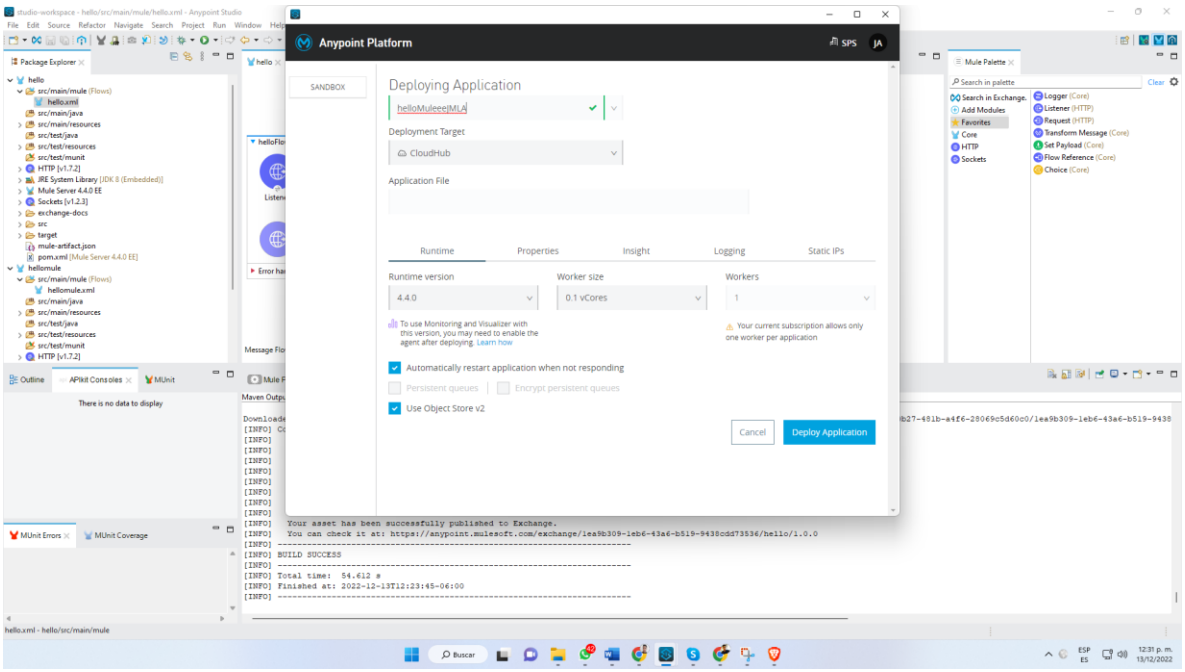




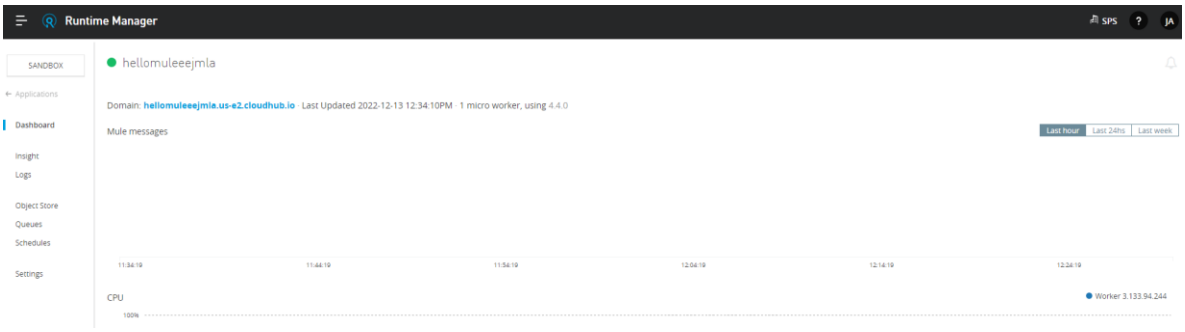
Paso 5: Se detiene el projecto para realizar el deploy, mediante CloudHub

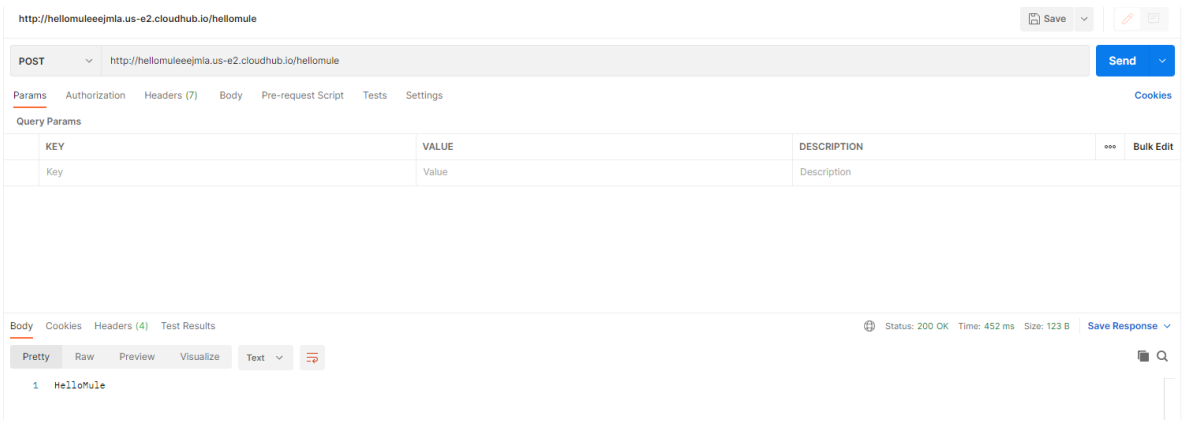

studio-workspace - hello/src/main/mule/hello.xml - Anypoint Studio
File Edit Navigate Search Project Run Window Help
Package Explorer
hello
hellomule
helloFlow
Listener
Set Payload
Error handling
Stop project hello
Undo
Redo
Go To XML...
Delete
Collapse All
Expand All
Message Flow Global Elements Configuration XML
Mule Palette
Search in palette
Clear
Search in Exchange...
Add Modules
Favorites
Core
HTTP
Sockets
Logger (Core)
Listener (HTTP)
Request (HTTP)
Transform Message (Core)
Set Payload (Core)
Flow Reference (Core)
Choice (Core)
Outline
APIkit Consoles
MUnit
Mule Properties
Console
Problems
Progress
There is no data to display
hello [Mule Applications] [pid: 12312]
INFO 2022-12-13 11:48:37,362 [WrapperListener_start_runner] org.mule.runtime.core.internal.logging.LogUtil:
+ Mule is up and kicking (every 5000ms) +
INFO 2022-12-13 11:48:37,408 [WrapperListener_start_runner] org.eclipse.jetty.server.AbstractConnector: Started ServerConnector@12b0651o{HTTP/1.1, (http/1.1)}{127.0.0.1:58594}
INFO 2022-12-13 11:48:37,411 [WrapperListener_start_runner] org.mule.runtime.core.internal.logging.LogUtil:
* - - + DOMAIN + - - * - - + STATUS + - - *
* default * DEPLOYED *
MUnit Errors
MUnit Coverage
* - - + APPLICATION + - - * - - + DOMAIN + - - * - - + STATUS + - - *
* hello * default * DEPLOYED *
INFO 2022-12-13 11:48:38,730 [[MuleRuntime].uber.03: [hello].uber@org.mule.runtime.module.extension.internal.runtime.source.ExtensionMessageSource.lambda$null$17:435 @c94615a] [processor:
Buscar
ESP ES
11:55 a. m. 13/12/2022


studio-workspace - hello/src/main/mule/hello.xml - Anypoint Studio
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
hello
src/main/mule (Flows)
hello.xml
New
Open F3
Open With
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Build Path
Refactor Alt+Shift+T
Import...
Export...
Refresh F5
Assign Working Sets...
Mule
Manage Dependencies
Anypoint Platform
MUnit
Run As
Debug As
Team
Compare With
Replace With
Source
Go To Node
Validate
Properties Alt+Enter
Download RAML from Design Center
Publish to Exchange
Deploy to CloudHub
Configure Credentials
helloFlow
Listener
Set Payload
Error handling
Message Flow Global Elements Configuration XML
Mule Palette
Search in palette
Clear
Search in Exchange...
Add Modules
Favorites
Core
HTTP
Sockets
Logger (Core)
Listener (HTTP)
Request (HTTP)
Transform Message (Core)
Set Payload (Core)
Flow Reference (Core)
Choice (Core)
Outline
Problems
Progress
4.4.0 EE (Terminated 13 dic 2022 12:13:41) [pid: 18360]
INFO 2022-12-13 12:13:13,003 [WrapperListener_start_runner] org.eclipse.jetty.server.AbstractConnector: Started ServerConnector@3243e666{HTTP/1.1, (http/1.1)}{127.0.0.1:61086}
INFO 2022-12-13 12:13:13,008 [WrapperListener_start_runner] org.mule.runtime.core.internal.logging.LogUtil:
* - - + DOMAIN + - - * - - + STATUS + - - *
* default * DEPLOYED *
MUnit Errors
MUnit Coverage
* - - + APPLICATION + - - * - - + DOMAIN + - - * - - + STATUS + - - *
* hello * default * DEPLOYED *
INFO 2022-12-13 12:13:13,717 [[MuleRuntime].uber.03: [hello].uber@org.mule.runtime.module.extension.internal.runtime.source.ExtensionMessageSource.lambda$null$17:435 @388b571c] [processor:
hello.xml - hello/src/main/mule
Buscar
ESP ES
12:14 p. m. 13/12/2022



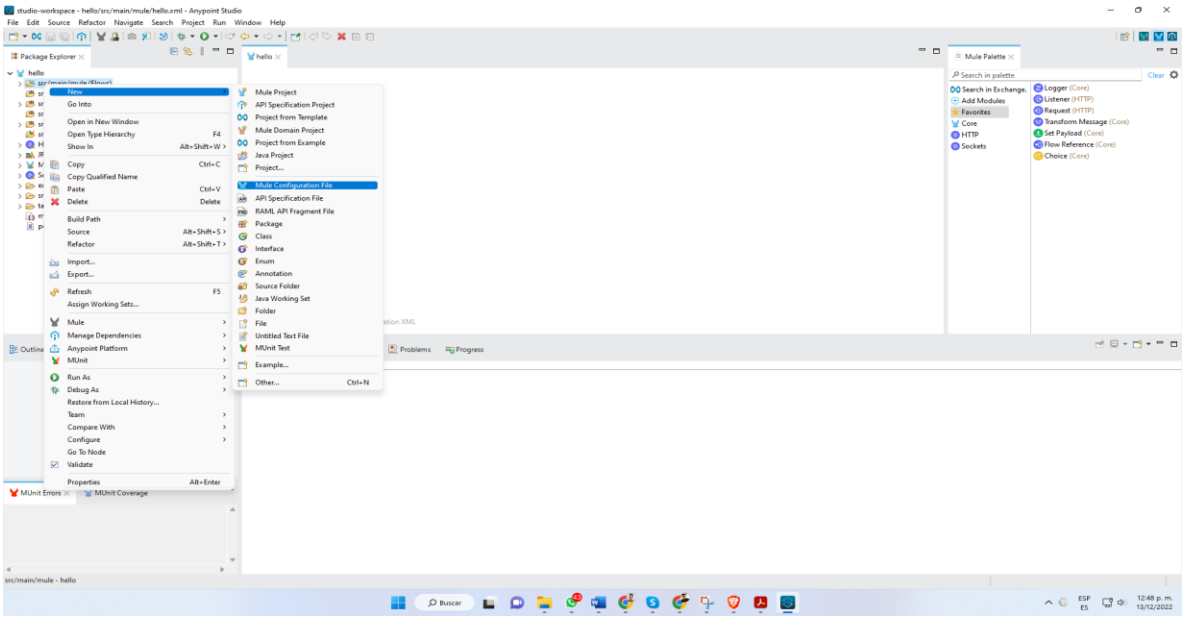
Una vez termino el deploy se abre Anypoint y se toma el URL para probar con una petición post que debe retornar un estado 200 (ok).



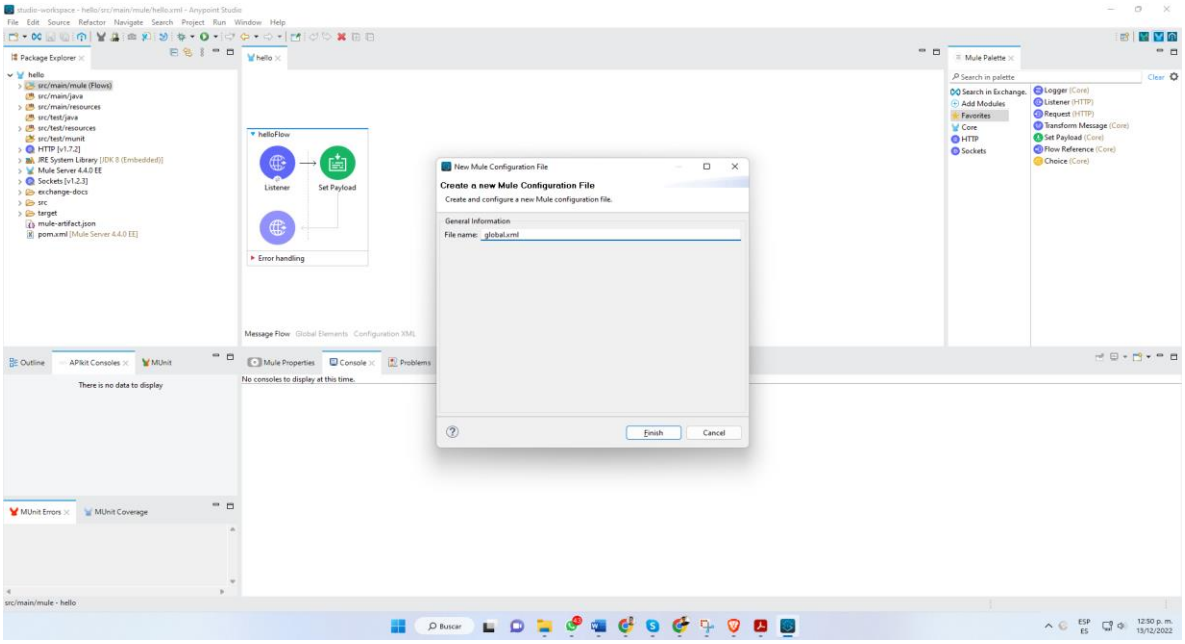


2-. Creación de archivo de propiedades, para mantener y referenciar datos sensibles por separado del código generado.

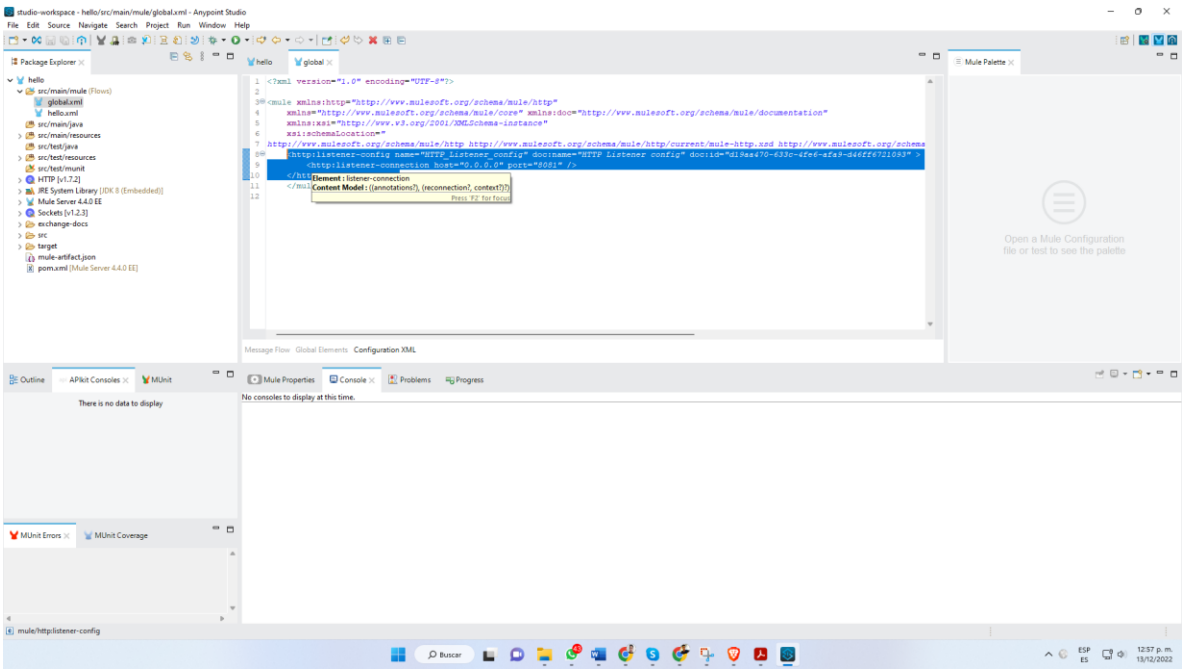
Paso 1: clic en carpeta src/main/mule y new file (Mule Configuration File)



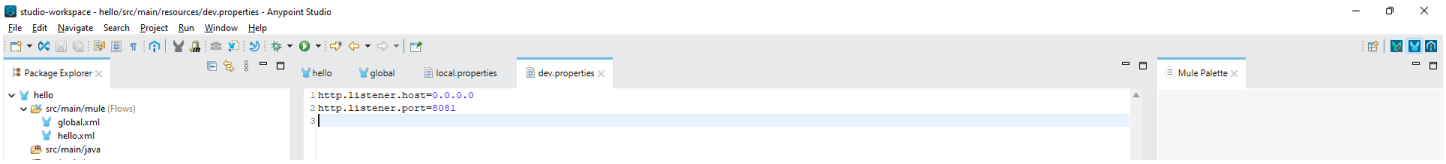
Se le da el nombre global.xml



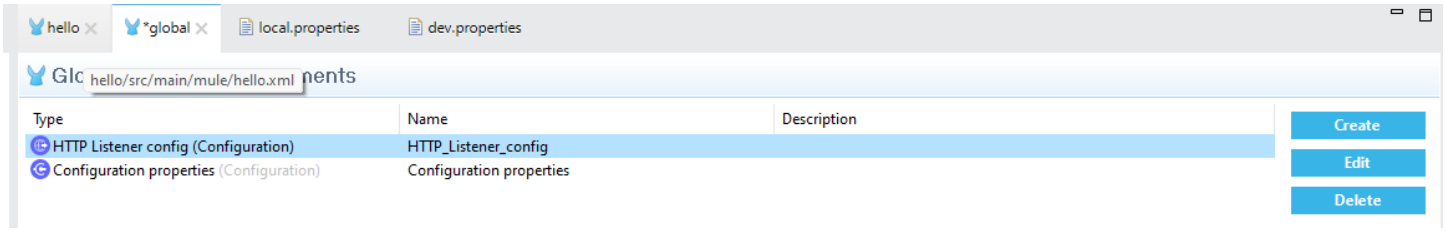
Paso 2: Cortar código http:Listener-config de hello.xml a global XML y guardar los archivos para evitar errores.



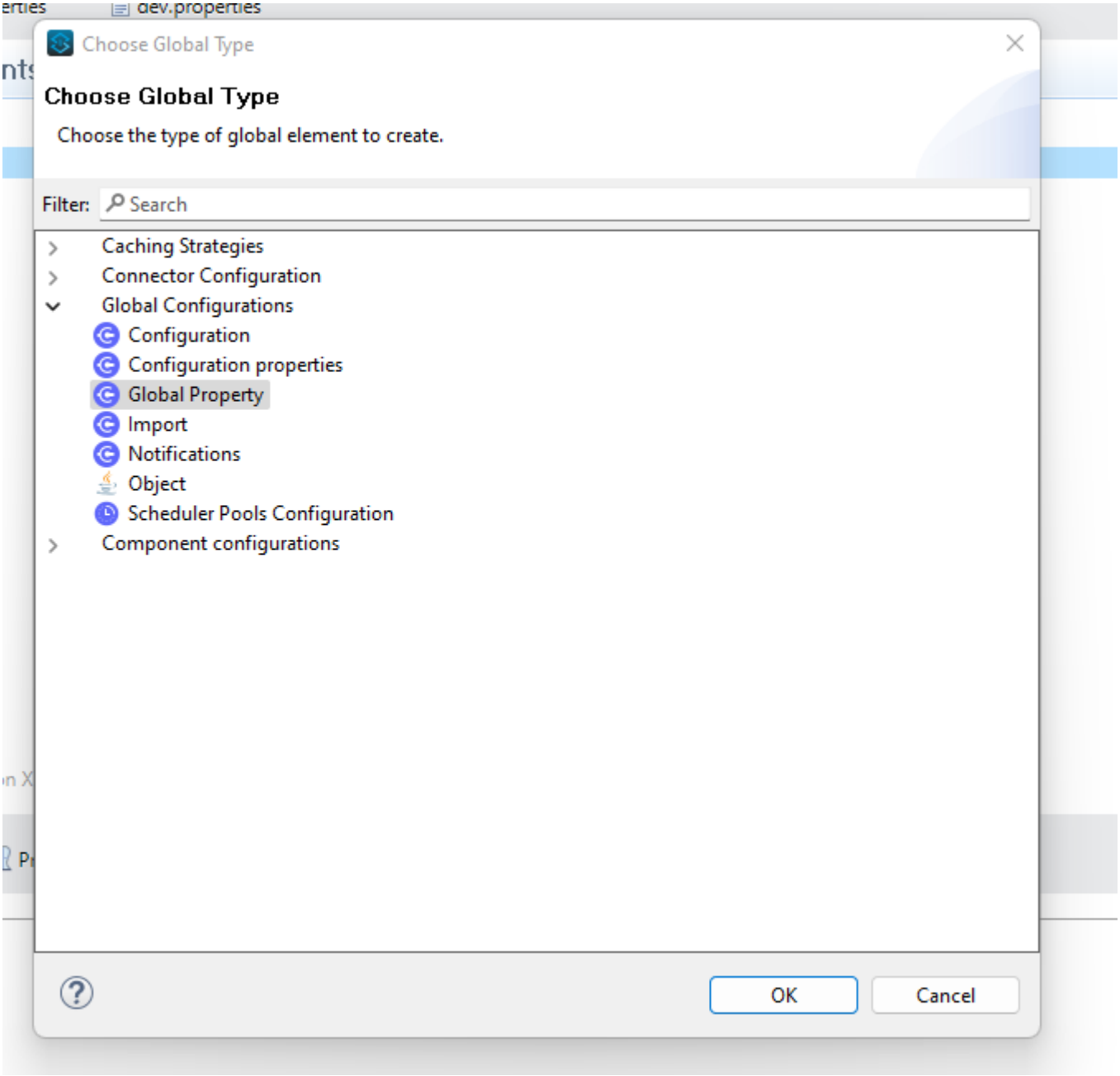
Creación de archivos File, para la configuración local y dev.

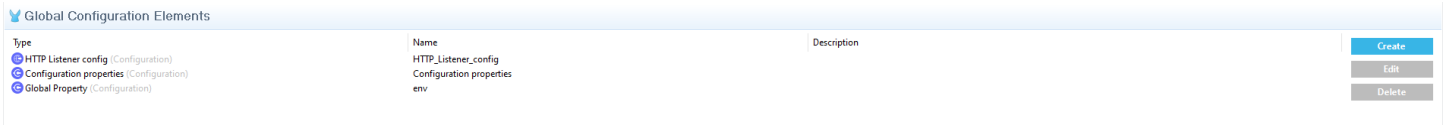


Creacion de elemento global Configuration properties, para poder referencia el archivo donde va tomar nuestras propiedades

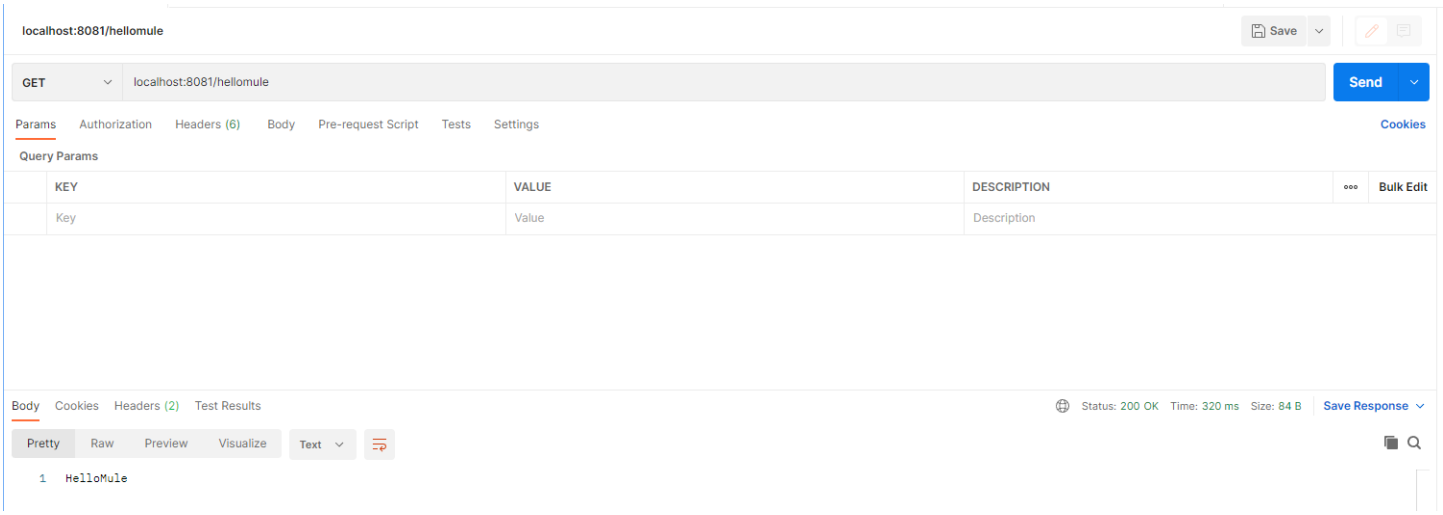


Crear global properties para usar archivo estático



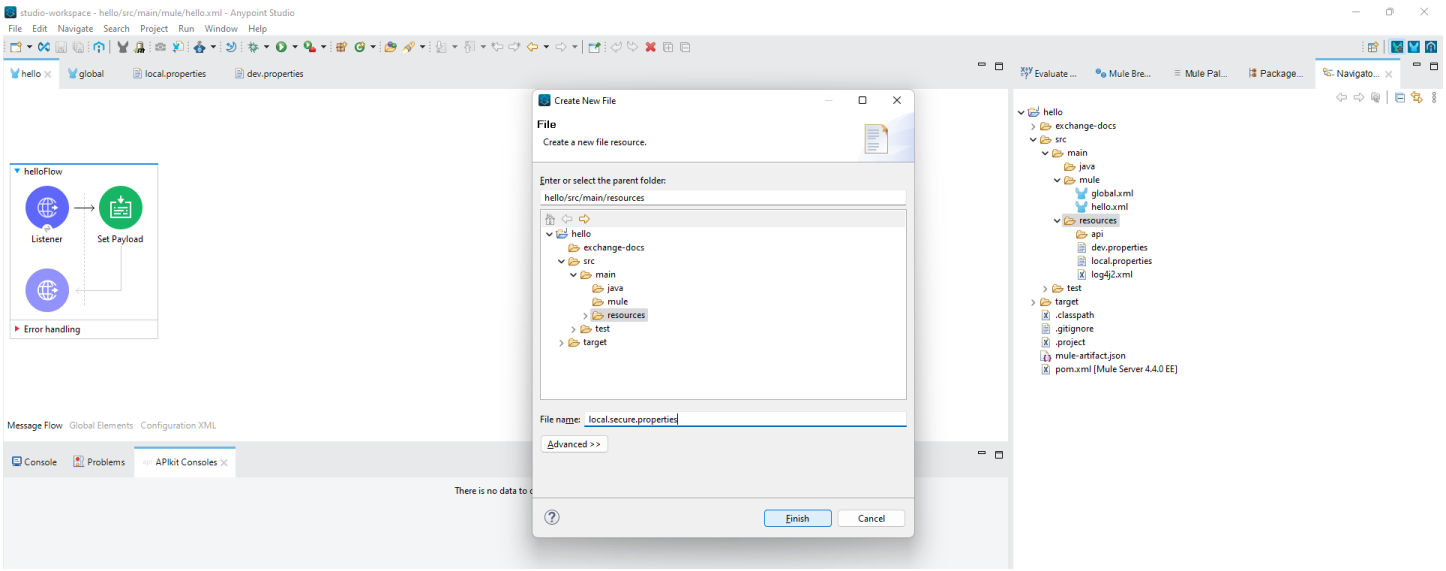


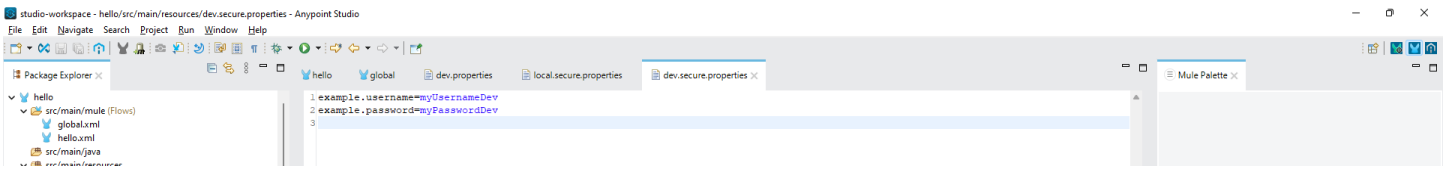
Prueba de configuración con postman método GET



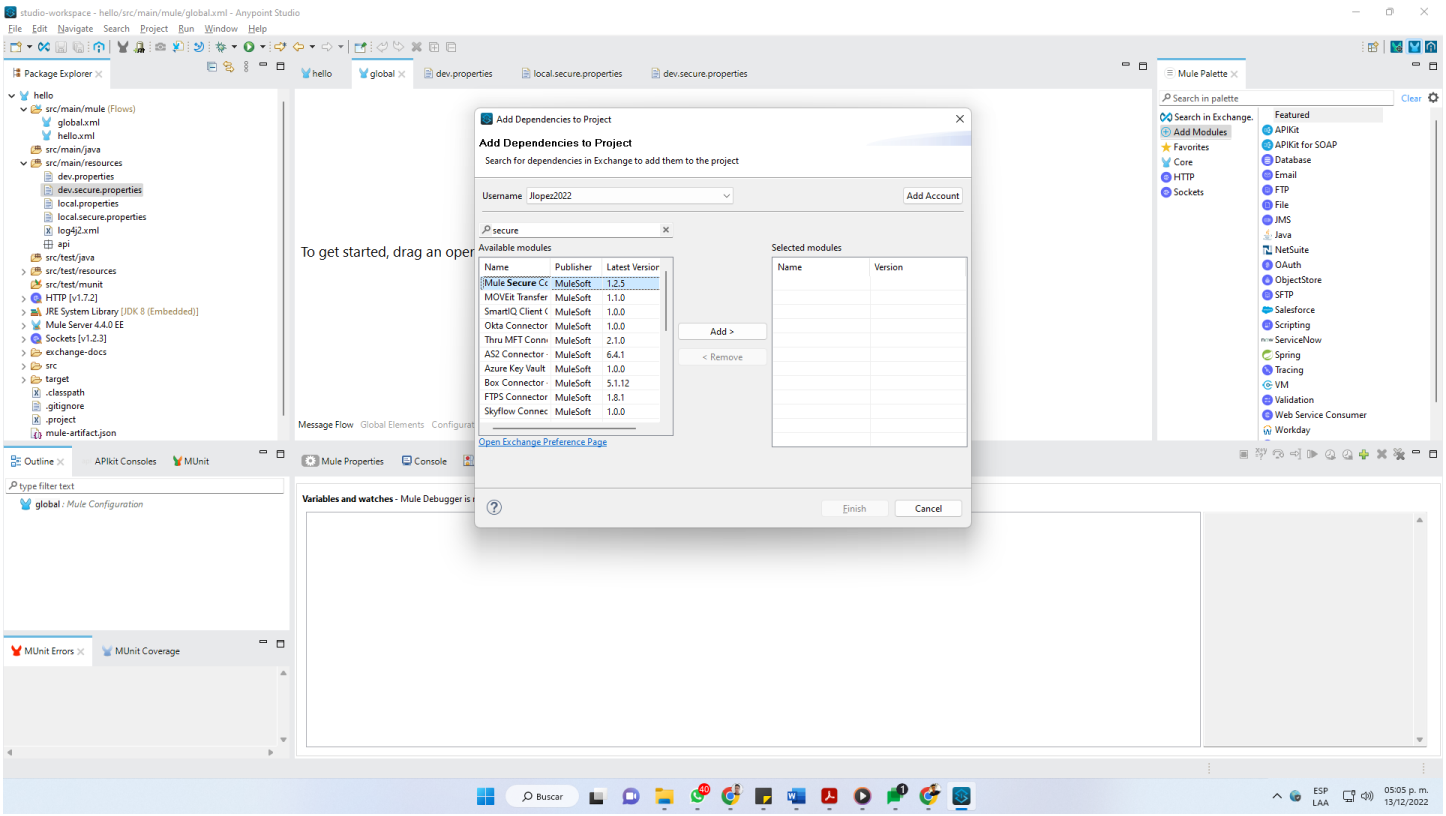
3.- Creación de archivo propiedades seguras, para proteger datos sensibles.

Crear archivo local y dev, dando clic en sources, new file y dar el nombre.

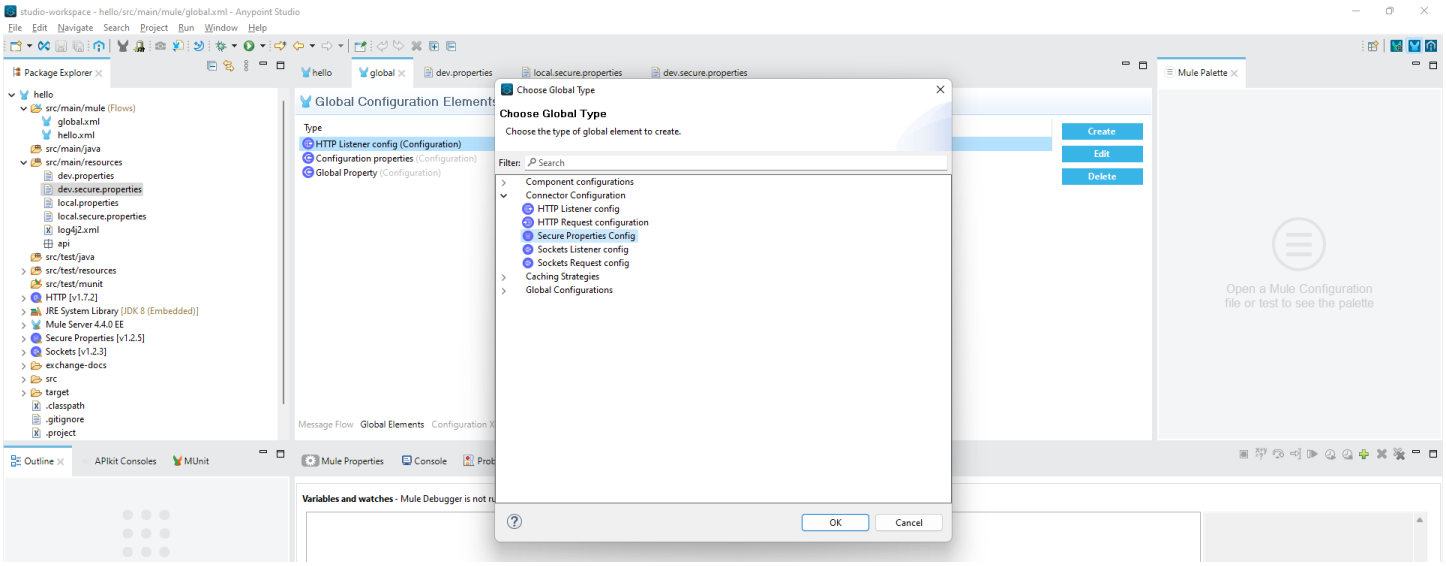




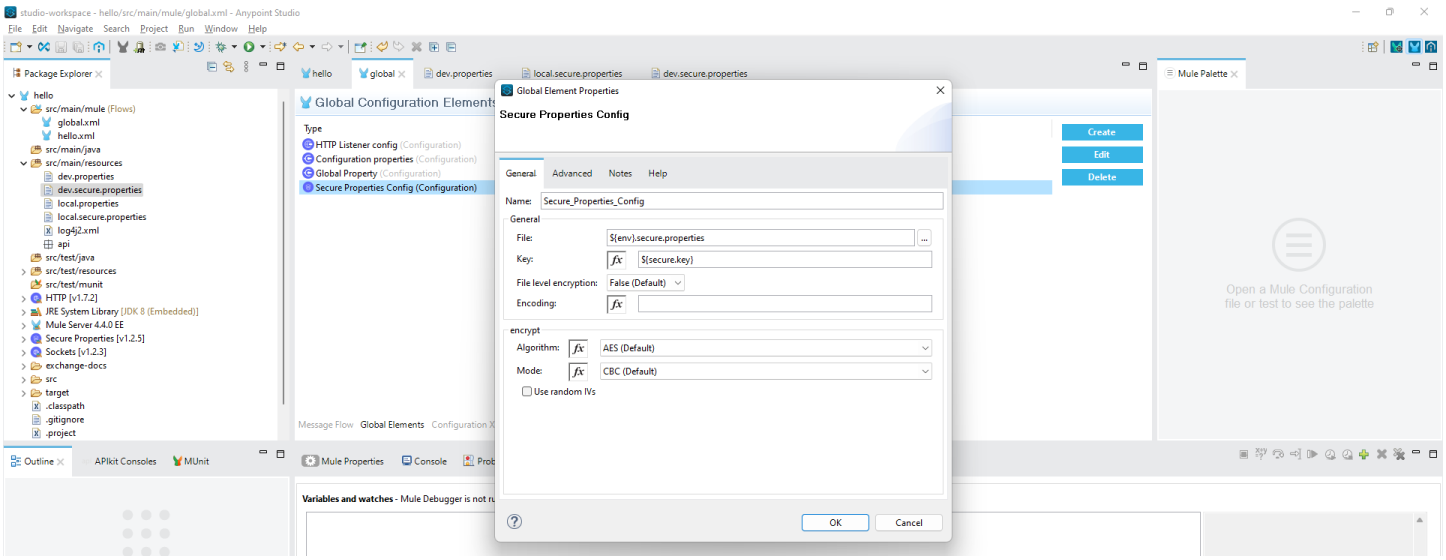
Configurar modulo de seguridad

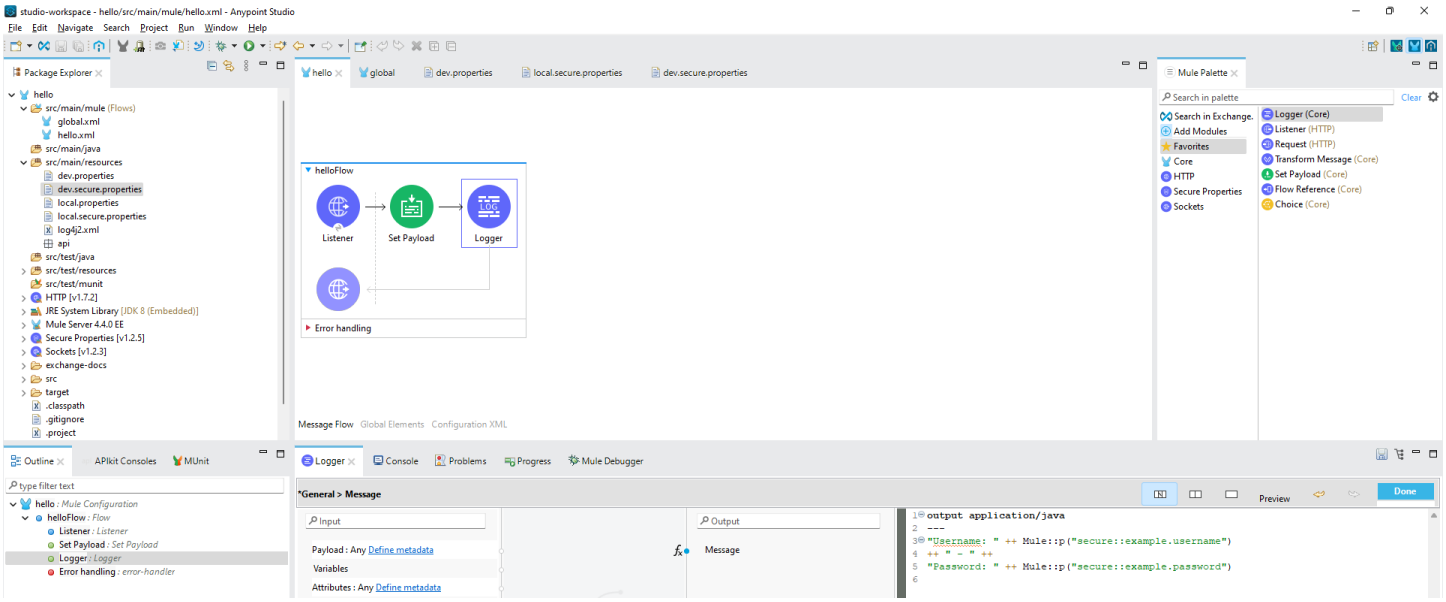


Se agrega archivo config secure properties en el global.xml

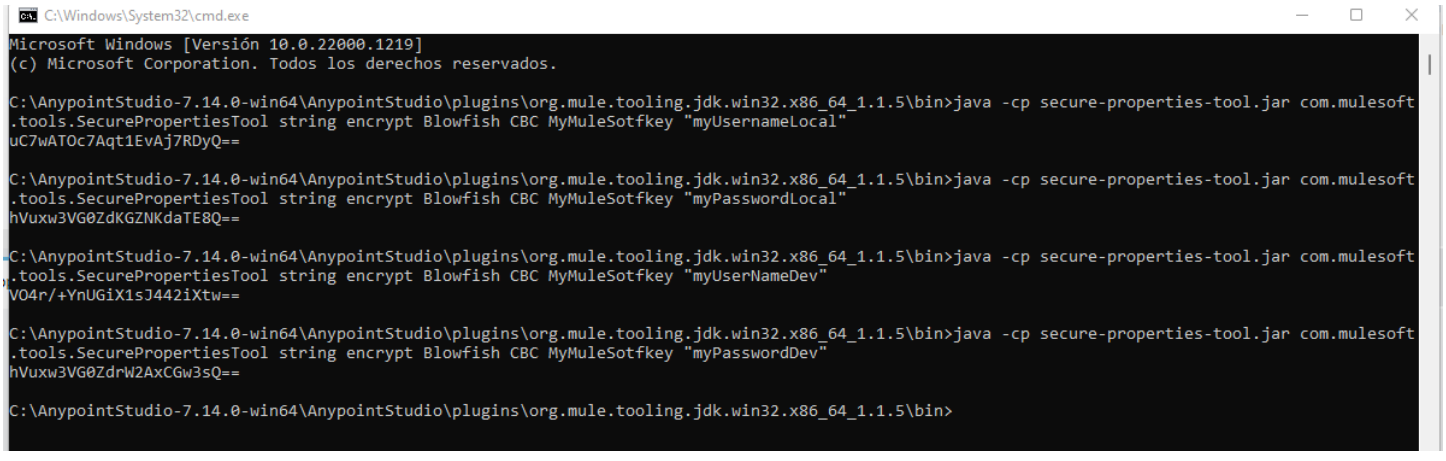


Configuración de propiedades dinámicas



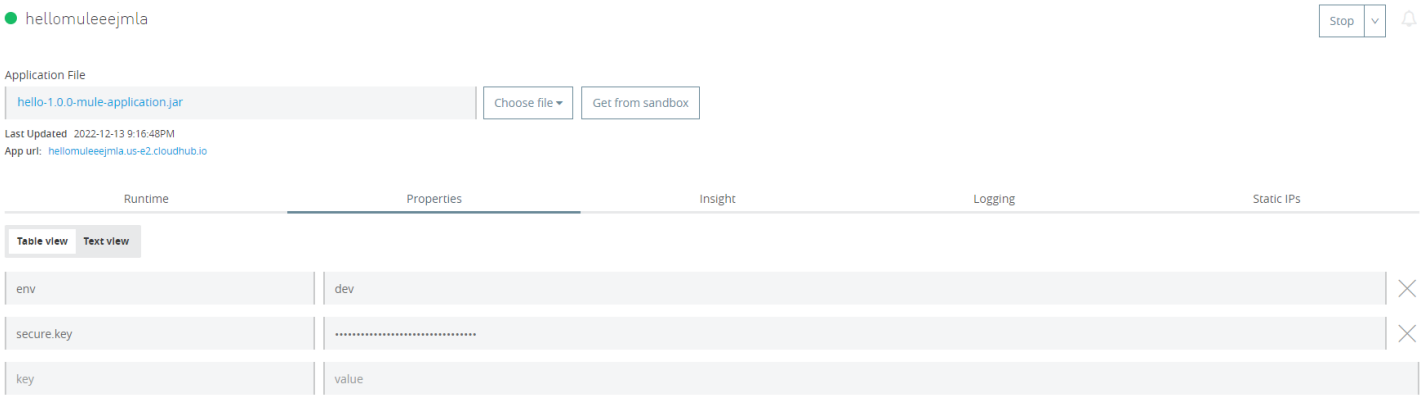


Encriptación de keys



Retorna al ejecutar el código

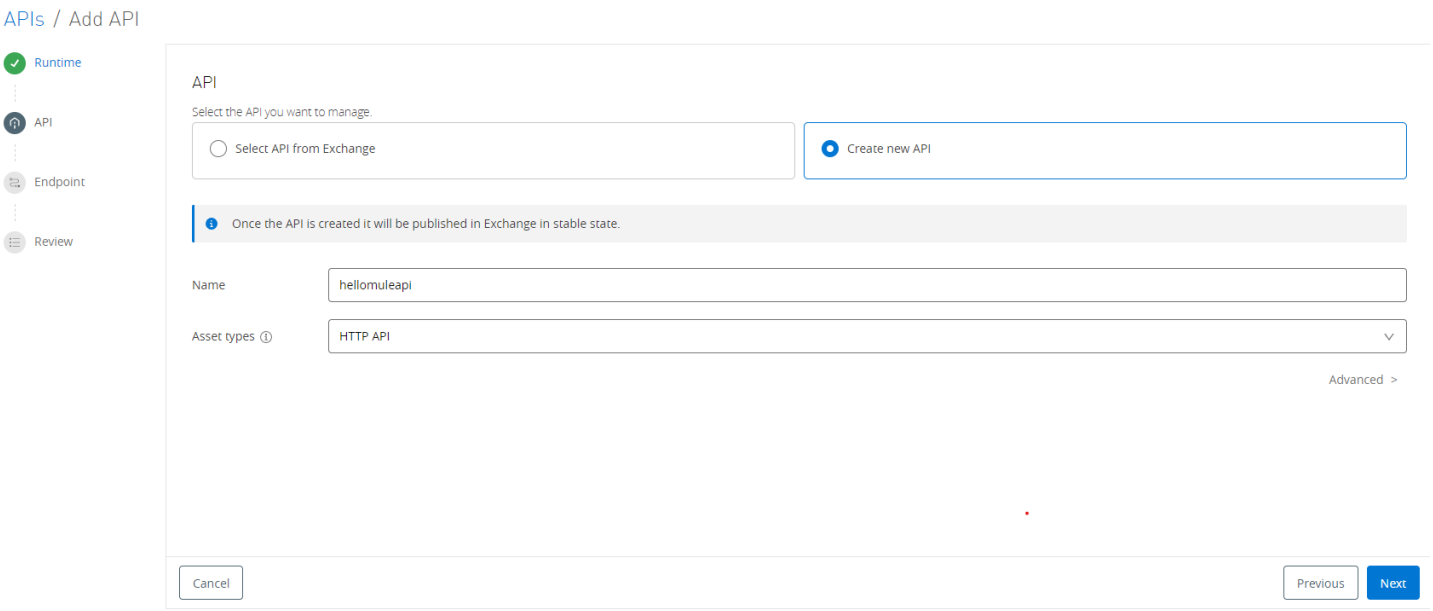
```
:6ed] [processor: ; event: ] org.mule.runtime.module.extension.internal.runtime.source.ExtensionMessageSource: Message sour
org.mule.runtime.core.internal.processor.LoggerMessageProcessor: Username: myUsernameLocal - Password: myPasswordLocal
```



Logs en aniport

```
21:16:48.796    12/13/2022    Deployment    system    SYSTEM
Your application is started.

21:19:27.792    12/13/2022    Worker-0    [MuleRuntime].uber.05: [hellomuleeejmla].helloFlow.CPU_LITE @1851bbd4    INFO
event:1dbd7050-7b5e-11ed-b51e-06552dc0eb08 Username: myUsernameDev - Password: myPasswordDev
```

4.-Creacion de Api en API MANAGER



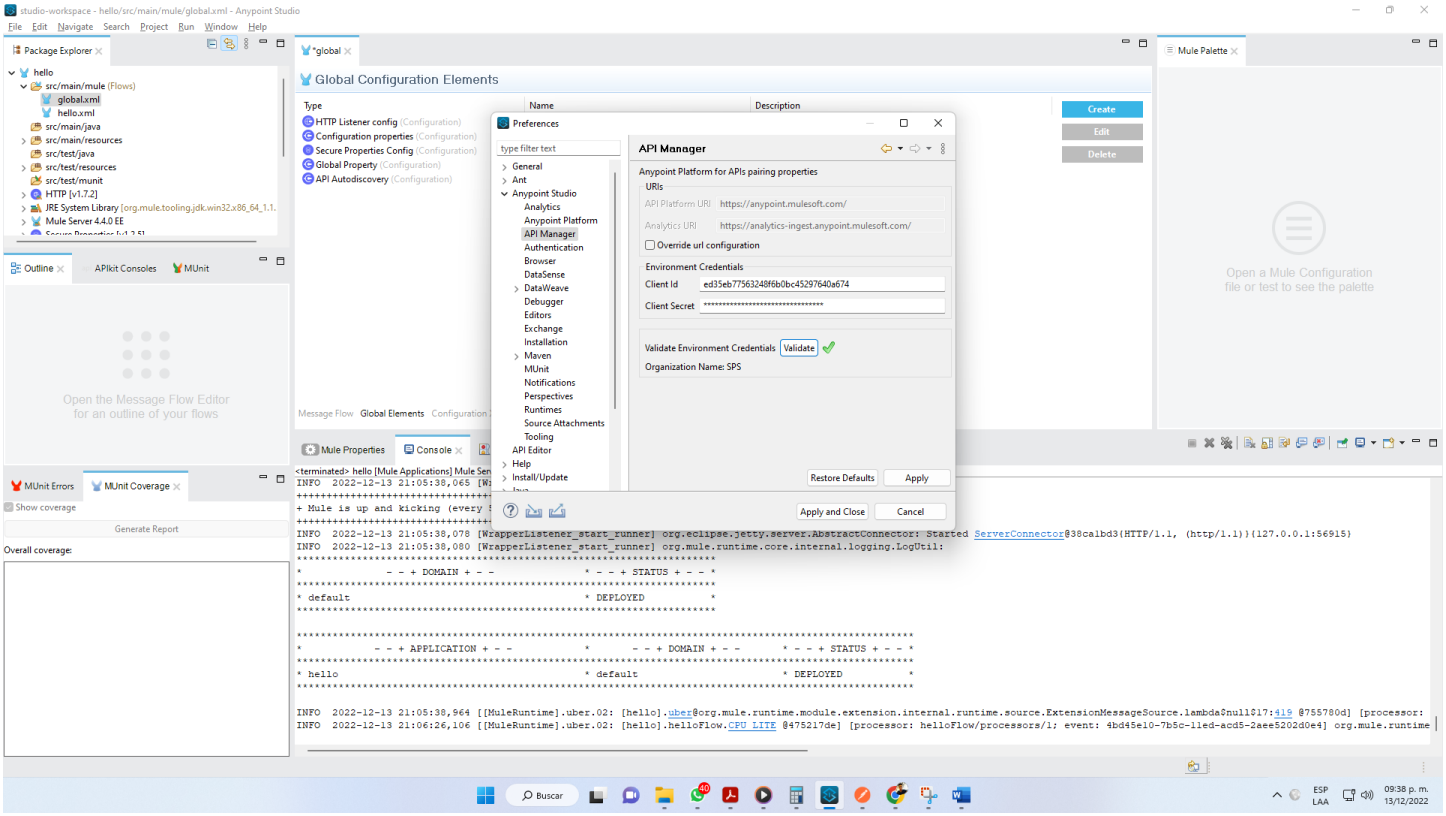
Configuración de api id en properties

dev.properties

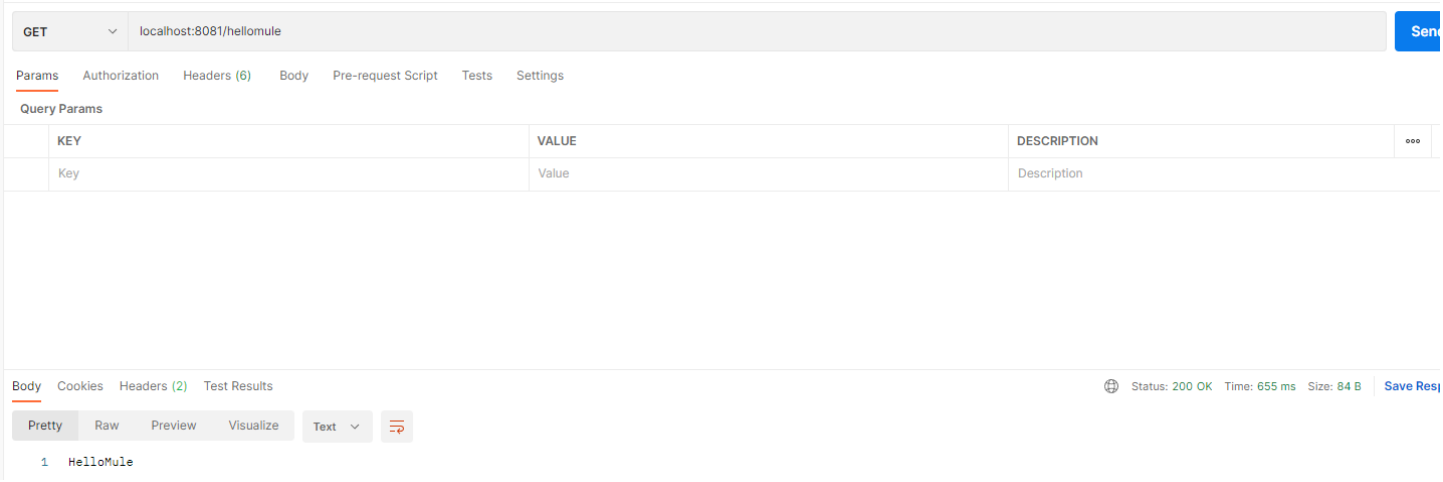
```
http.listener.host=0.0.0.0
http.listener.port=8082

api.id=18367341

```

local.secure.properties | dev.secure.properties | local.properties

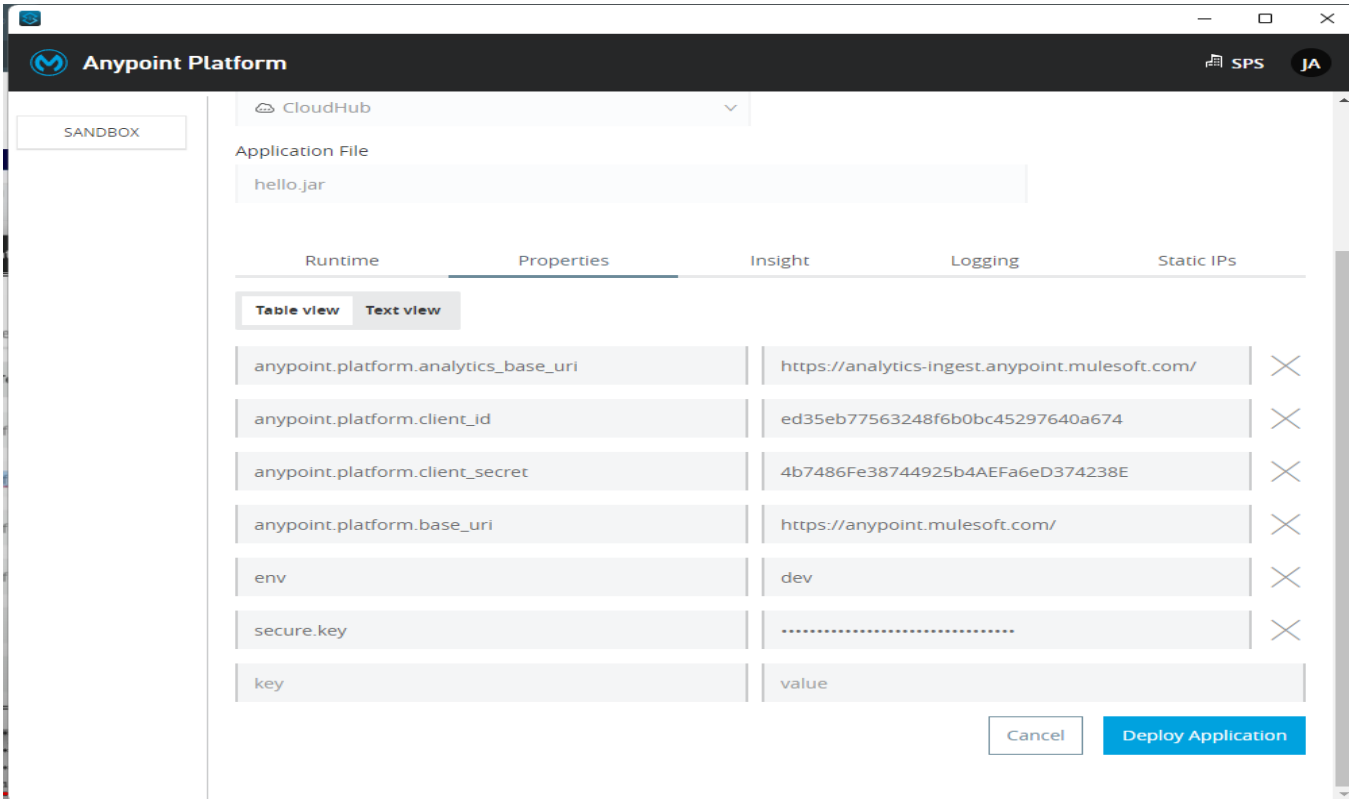
```
http.listener.host=0.0.0.0
http.listener.port=8082

api.id=18367341
```



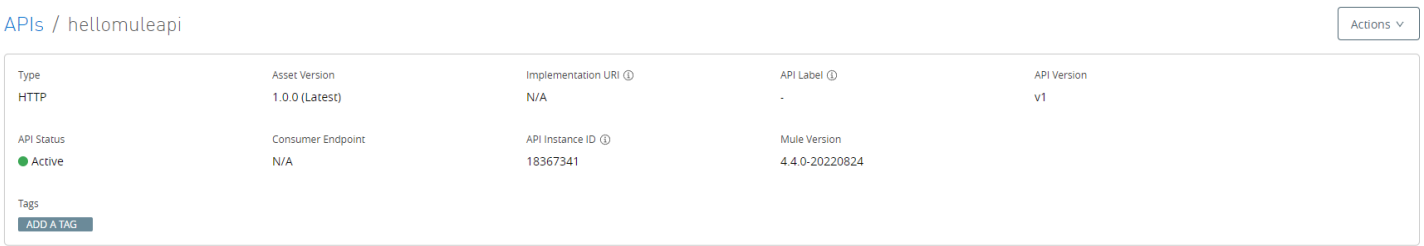
Se realiza deploy y se valida en postman



Deploy en CloudHub



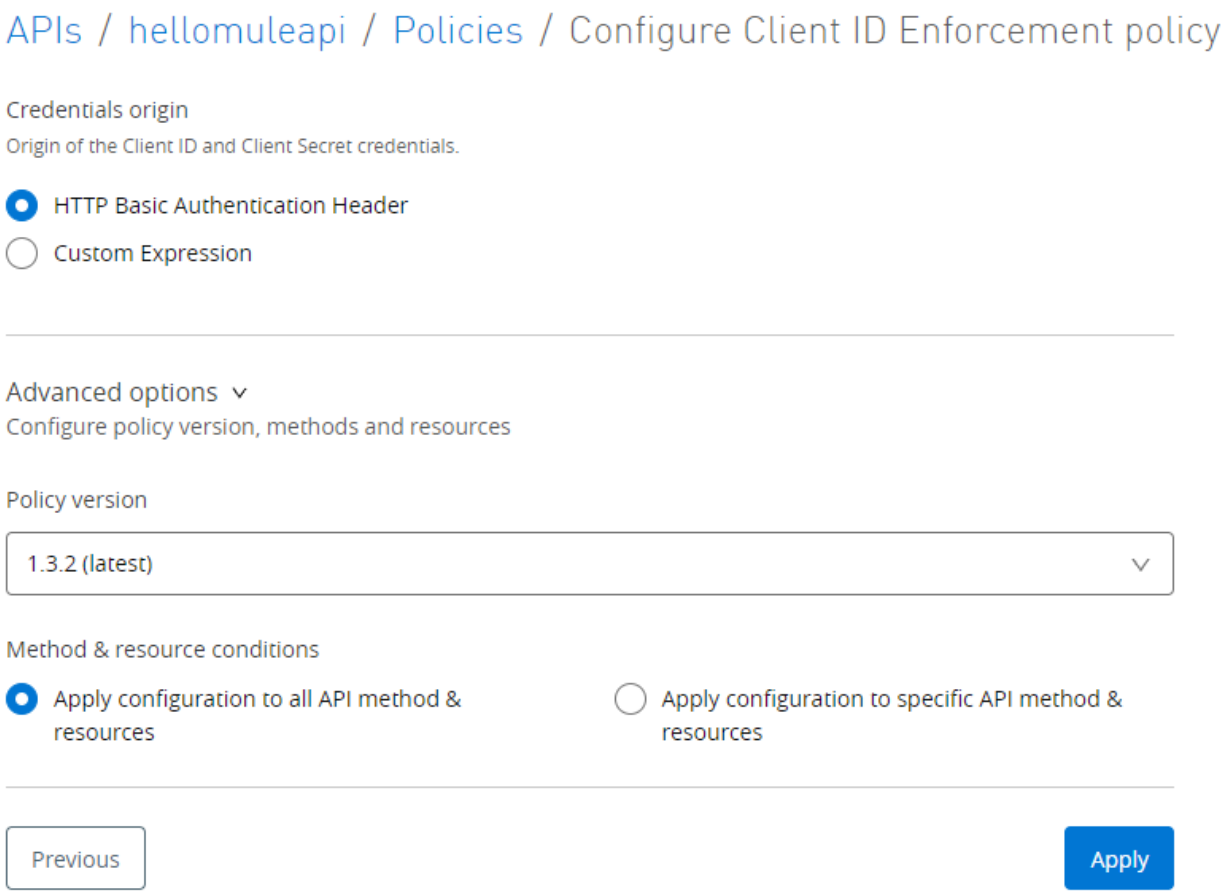
Validación de api activa en API MANAGER



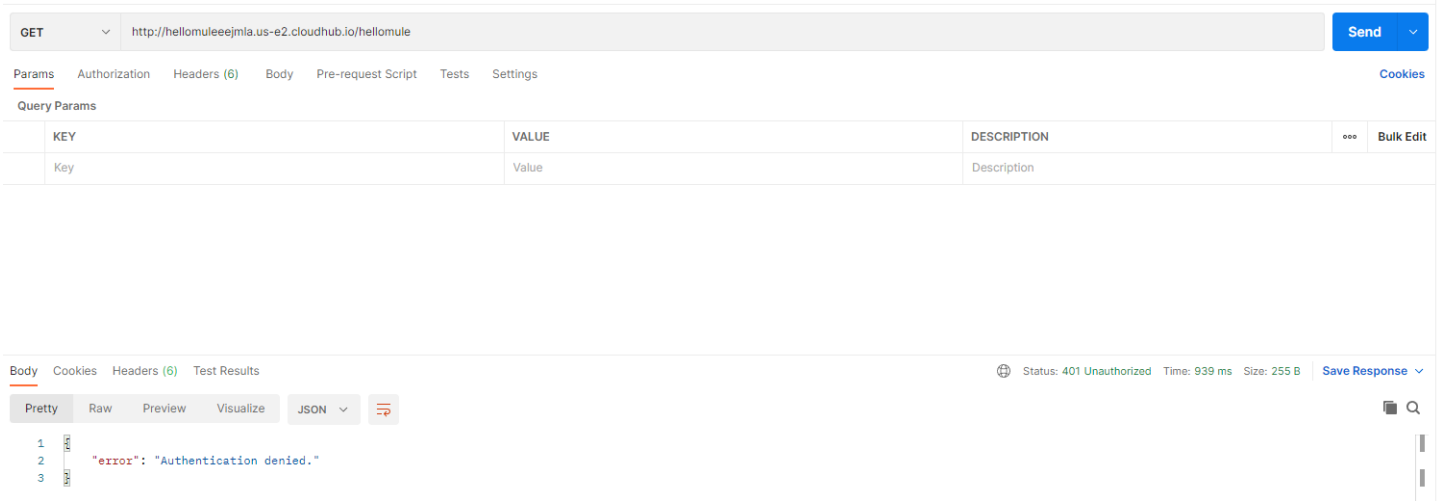
5.- Aplicar client ID a API en API MANAGER

Se debe entrar a API MANAGER en Anypoint, entrar a la API y pólices (políticas).

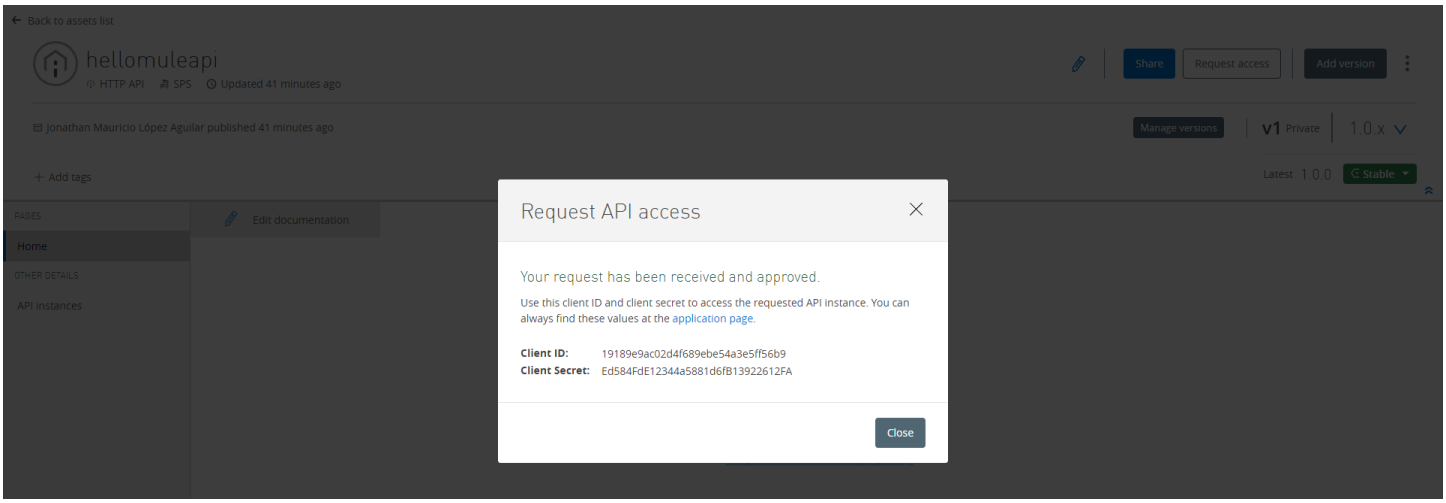
Dar clic en aplicar nueva política y configurar la misma.

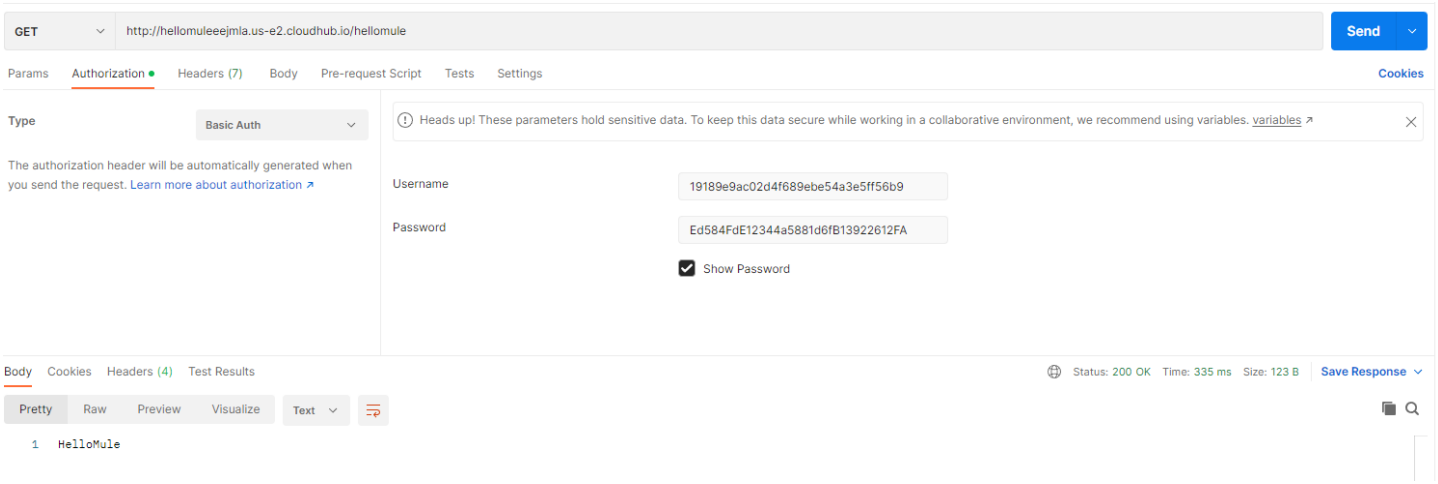


Se comprueba que política funciona, ya que no da acceso a la información, comprobado mediante postman.



Se crea request Access y se comprueba con postman





6.- Conocer sobre el enfoque de First Api Specification

Principalmente una API es una tecnología que permitirá comunicar aplicaciones entre sí, el desarrollador backend proporciona la parte lógica para ser consumida por peticiones de información que solicita el lado del cliente o front end.

Estas deben ser creadas en un estándar haciendo que sean fáciles de manipular y usar esta información.

Como desarrollador de Apis, se debe ser claro en la forma en como los futuros desarrolladores o quien tenga necesidad de consumir el api se podrán conectar o hacer uso de la misma.

Se debe especificar un url base en el cual se alojará la API, también se puede configurar el protocolo de comunicación para el api, como puede ser HTTP.

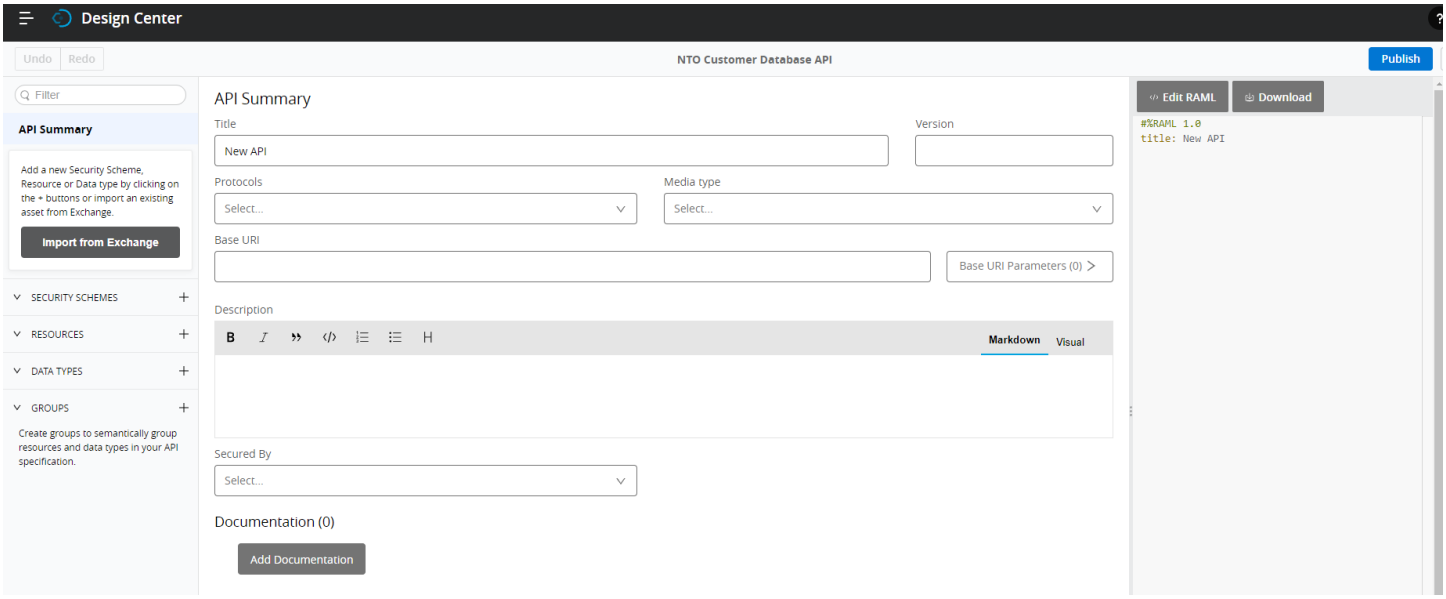
Un api restfull tiene métodos get (recibir información), post (Crear información), patch (Actualizar información de una instancia parcialmente), put (Actualizar una instancia completa) y delete (borrar datos).

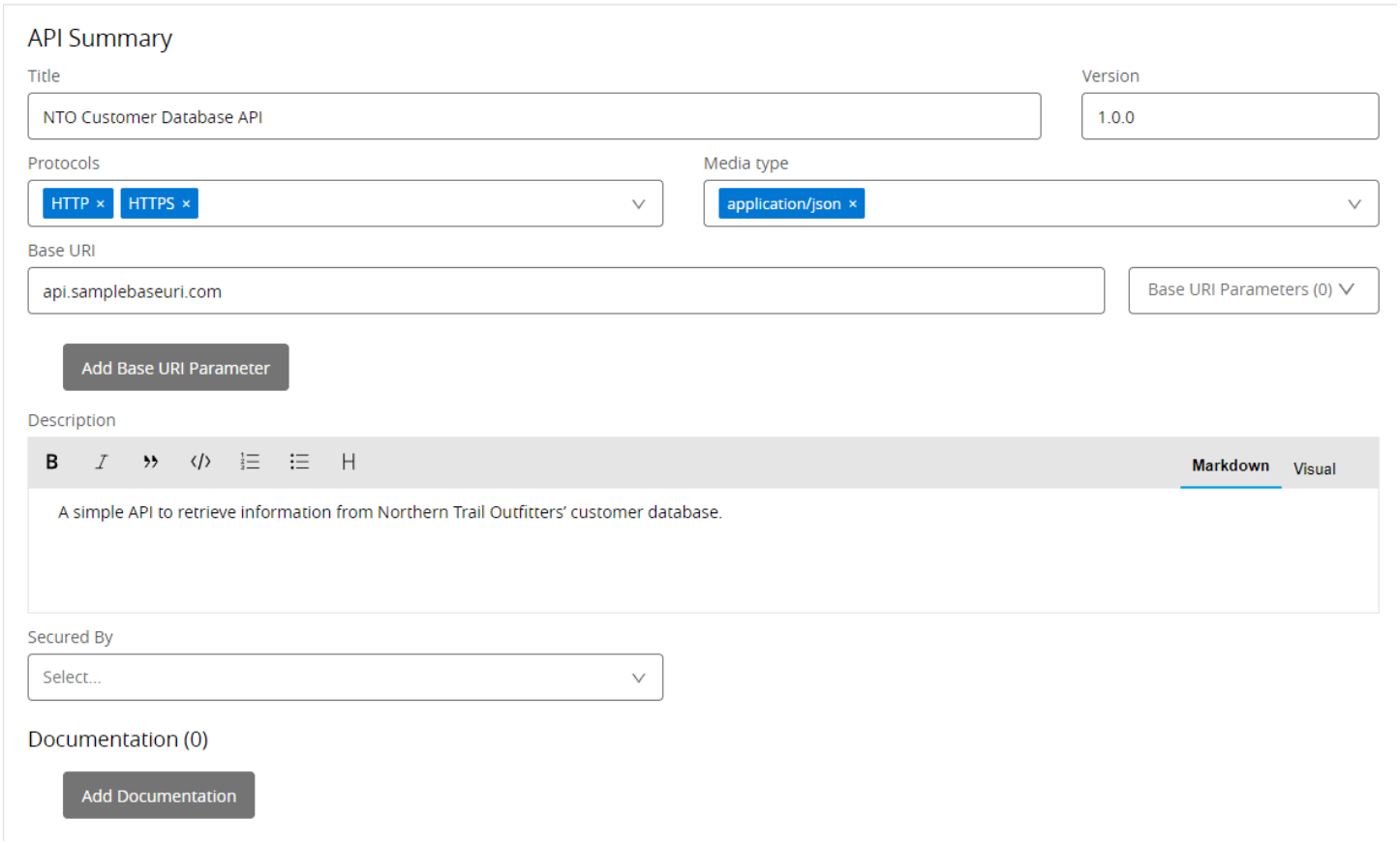
Los parámetros de consulta deben ser creados usando snake_case por convención.

Toda consulta de una API, devuelve un estado los 200 son respuestas exitosas, 400 generalmente errores de ruta o acceso denegado y los 500 errores del servidor.

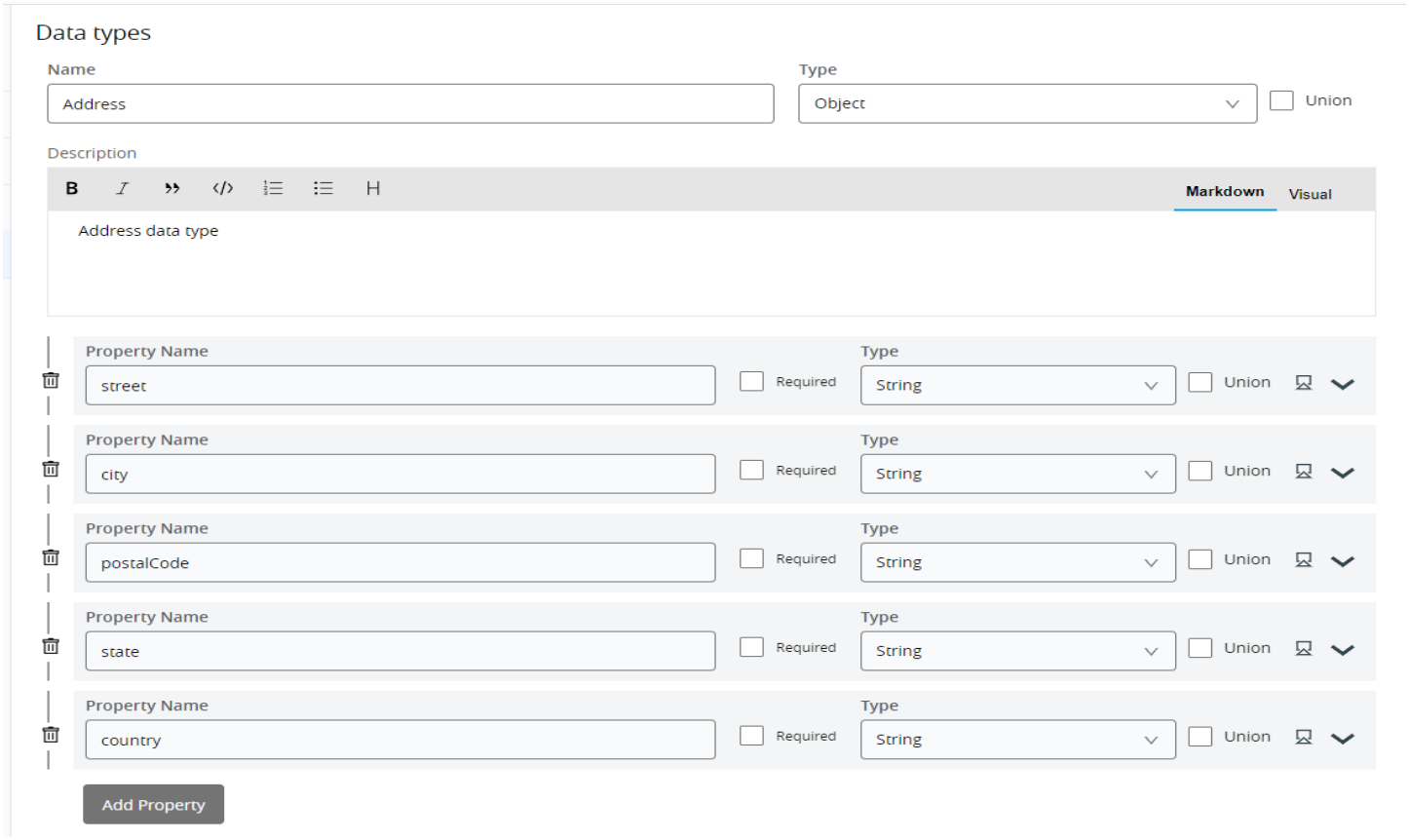
7.- Construcción de API en API DESIGNER.

Se debe entrar a Anypoint y desing Center, dar clic en crear nueva API, NEW API SPECFICATION (SPEC).

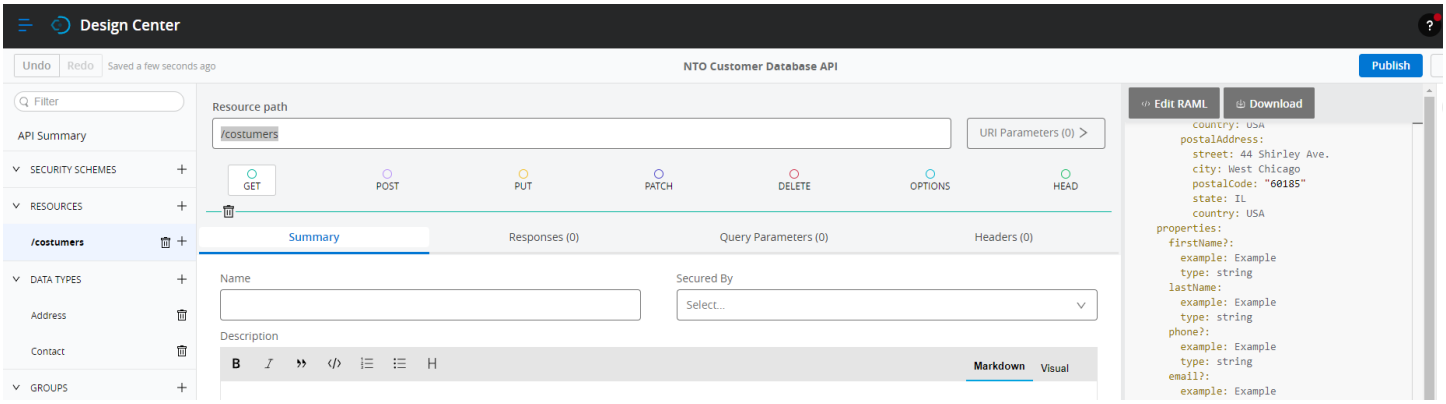




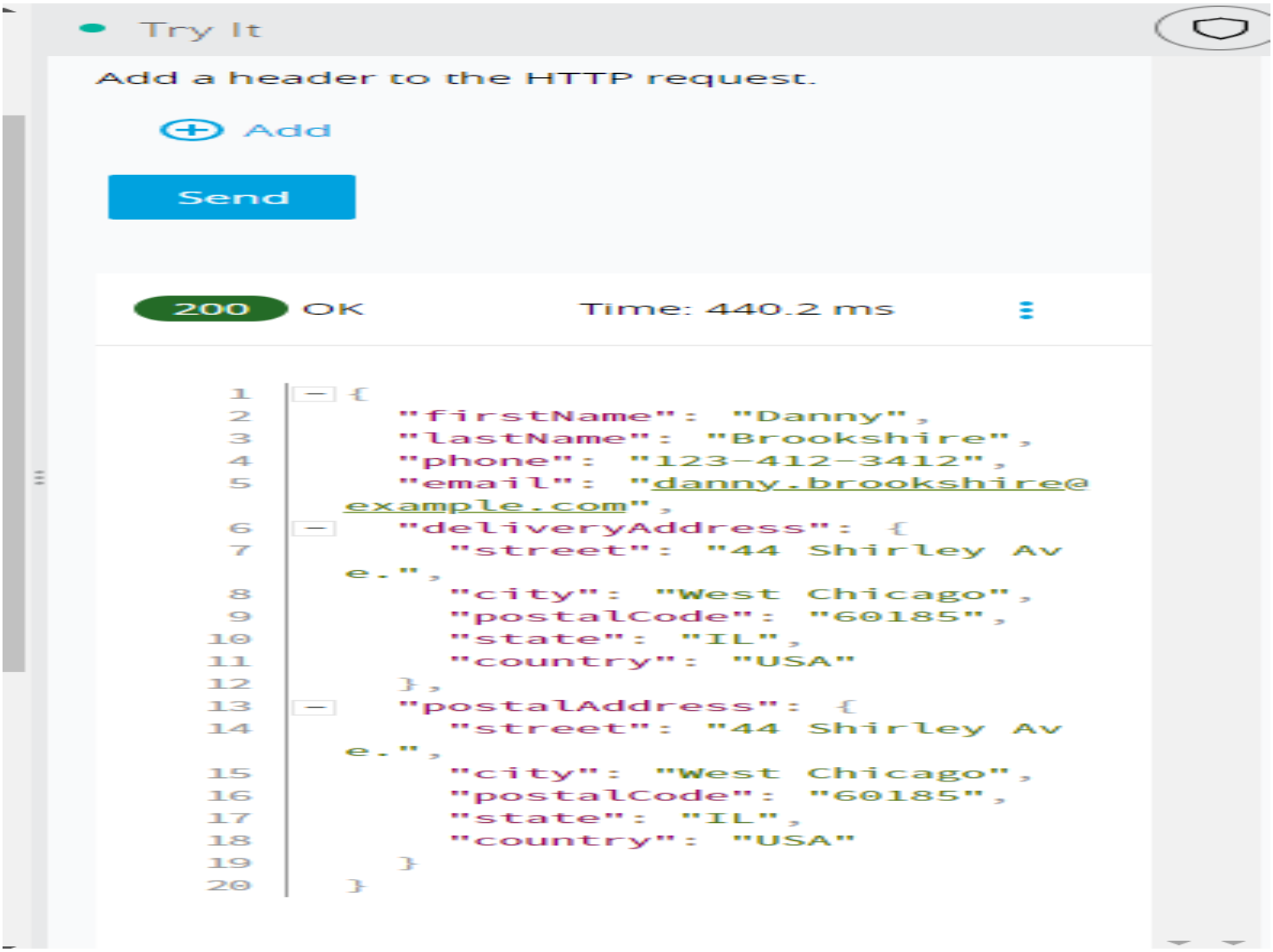
Se agrega Data Type (Address y Contact) y recursos que necesitara nuestra API



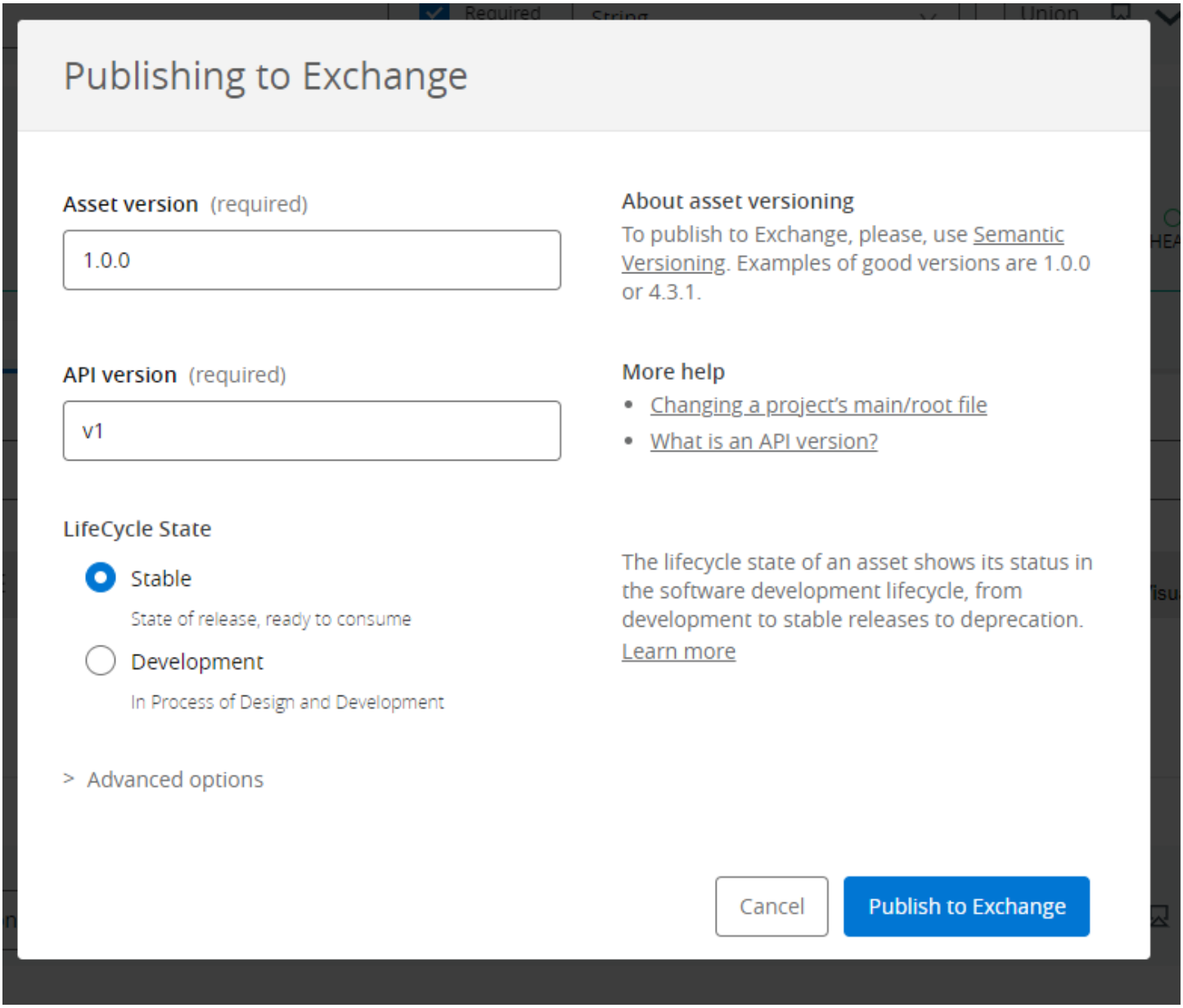
Dar clic en resources para configurar los métodos de nuestra API



Se puede realizar simulaciones de las peticiones y obtener una respuesta

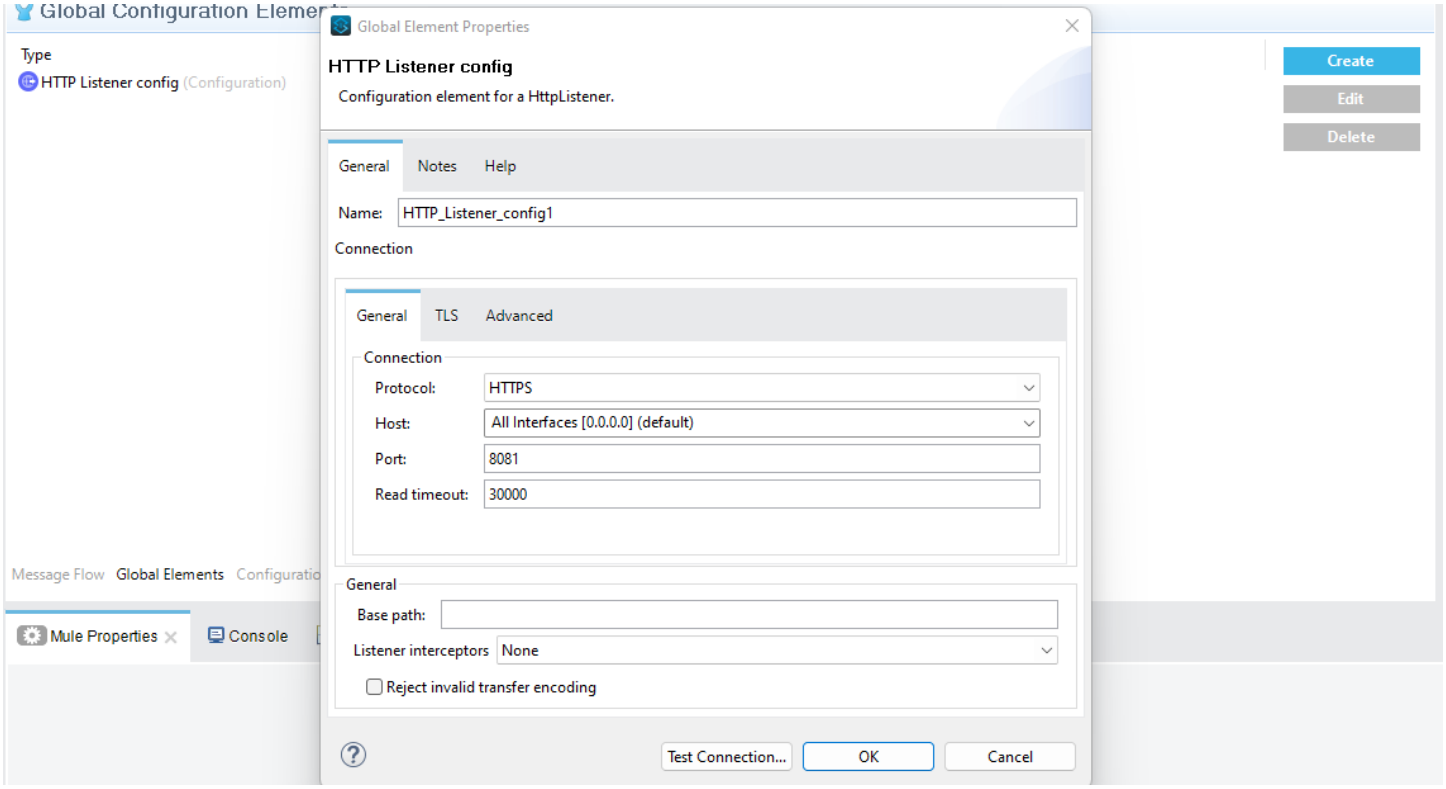


Publicación de la API

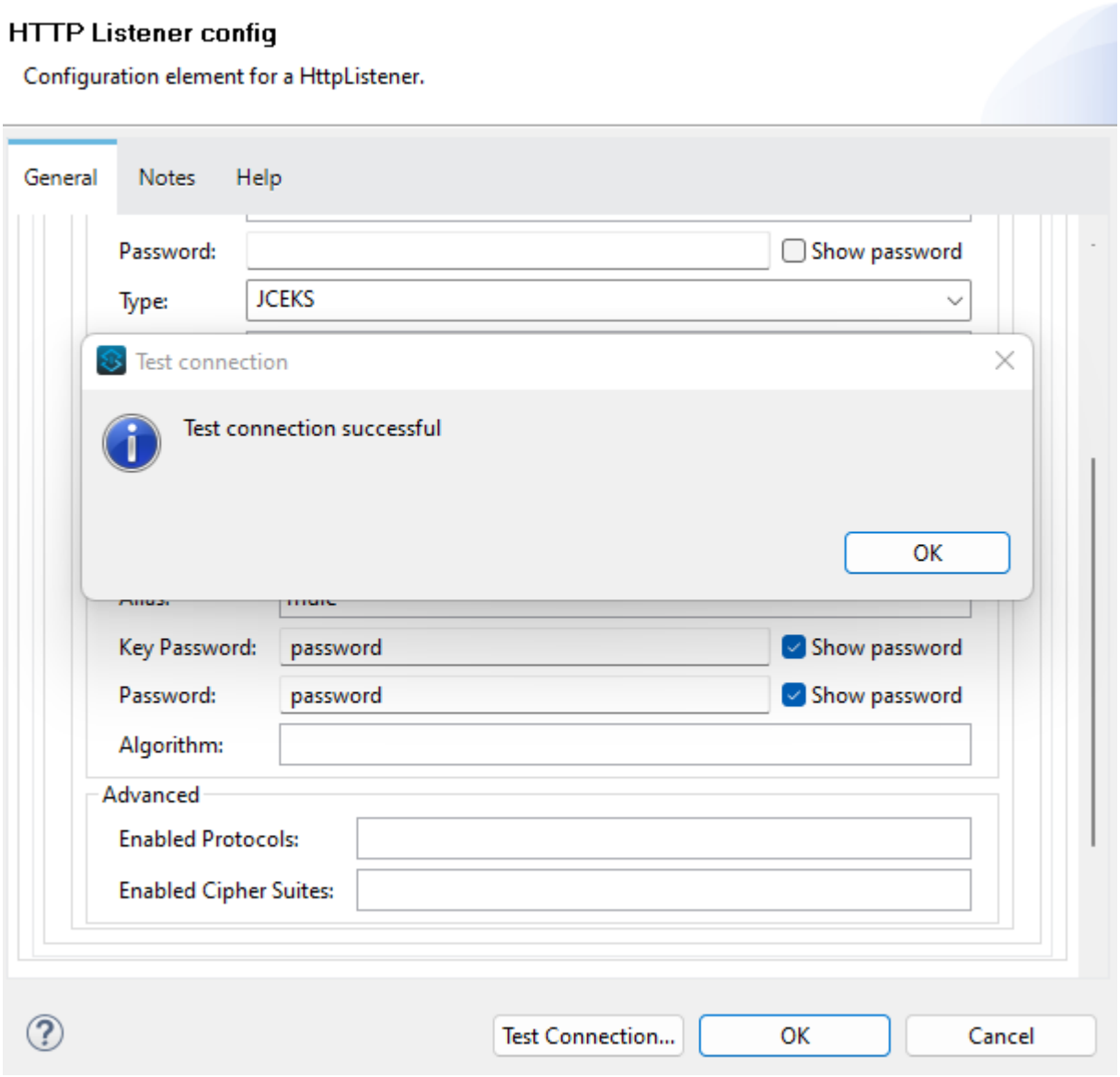

Publishing to Exchange
Asset version (required)
1.0.0
About asset versioning
To publish to Exchange, please, use Semantic Versioning. Examples of good versions are 1.0.0 or 4.3.1.
API version (required)
v1
More help
Changing a project's main/root file
What is an API version?
LifeCycle State
Stable
State of release, ready to consume
Development
In Process of Design and Development
The lifecycle state of an asset shows its status in the software development lifecycle, from development to stable releases to deprecation.
Learn more
> Advanced options
Cancel
Publish to Exchange


Back to assets list
NTO Customer Database API
REST API
SPS
Updated 25 seconds ago
Share
Download
View code
Add version
Jonathan Mauricio López Aguilar published 24 seconds ago
Manage versions
v1 Private
1.0.x
+ Add tags
Latest 1.0.0
Stable
Not Validated

8.- Habilitar HTTPS consume en mule Aplicación




```
  [mexico]:
¿Cuál es el código de país de dos letras de la unidad?
  [mx]:
¿Es correcto CN=jonathan lopez, OU=sps, O=sps, L=mexico, ST=mexico, C=mx?
  [no]:  SI


Warning:
El certificado generado uses the SHA1withRSA signature algorithm which is considered a security risk. This algorithm wil
l be disabled in a future update.

C:\Users\Jonathan Lopez\AnypointStudio>
```



Prueba en POSTMAN