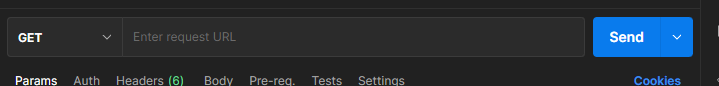
**Seccion 2: Testing Api´s using postman tool**

**3. Install Postman and Start working with Postman:**

* Herramienta usada para probar APis, es opensource, pero tienen partes licenciadas
* Esta disponible en 2 formatos, standalone, web.
* Para instalar el plug in se debe ir a la tienda de Chrome, buscar postman e intalarlos
* O se puede descargar e instalar
* Luego de instalarlo se puede abrir normalmente, para usarla se debe crear una cuenta.
* Una colección se puede tomar como un test suit, en ella se pueden tener varias sentencias.
* En la barra superior es doned se harán las request, en ella se puede seleccionar los diferentes métodos que se tienen



Texto

Descripción generada automáticamente

* En la parte inferior se puede ver la respuesta o body. En este caso retorna un HTML
* Por ejemplo si se quiere usar una request de POST, se debe enviar alguna data, eso se especifica en el body del request

Captura de pantalla de un celular

Descripción generada automáticamente

* Se pueden enviar diferentes tipos como raw, binario, etc

Captura de pantalla de un celular

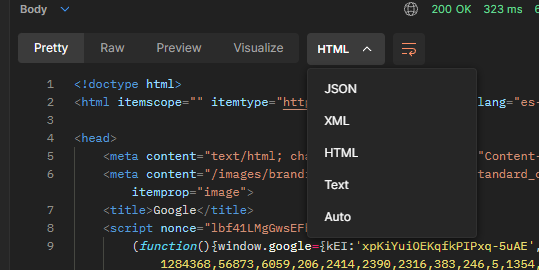
Descripción generada automáticamente

* Si por ejemplo se selecciona raw, luego se puede seleccionar el tipo de texto que se quiere enviar, se pueden enviar diferentes tipos como JS, Text,Json, XML, HTML. Esto depende del formato que soporte el API

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

* A parte del url el desarrollador debe decir que recibe, como, etc
* En la respuesta también se tienen diferentes formatos que se puede seleccionar

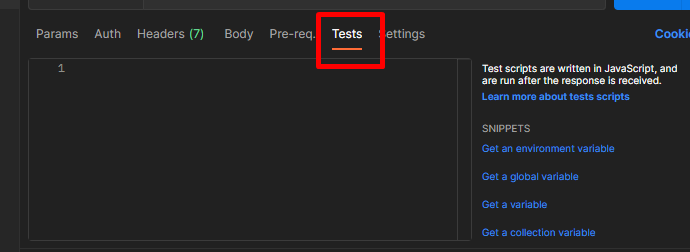


* Una de las cosas mas importantes que se debe validar en la respuesta es el estatus

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* En la parte superior, opciones del request se tiene una ventana de Tests, en esta ventana se puede usar para validar cosas, se pueden tener validations points. Revisar si en la respuesta hay algunos campos o no

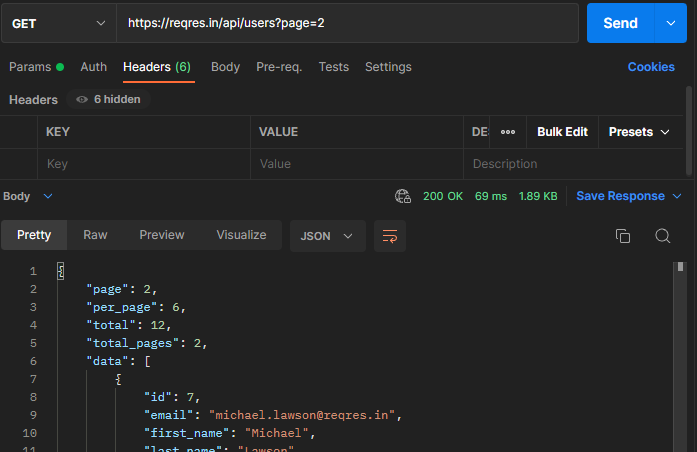


* Alli se puede escribir algún tipo de código, por ejemplo, Java Script
* Hay otra ventana de pre requisitos, antes de enviar el request, se realiza lo que allí se diga

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Los headers se pueden enviar algunas veces ya que las APIs responden a ellas
* EL siguiente tab es te autorizaciones, las APIS no siempre responde a todos los llamados, muchas veces se deben tener unas keys para poder ingresar a ellas.
* Hay 2 tipos de APIS, globales que son las que cualquiera puede acceder a ellas, y otras las cuales son diseñadas por empresas y se necesita alguna key de autenticación para poder acceder a ellas.
* Endpoint: https://reqres.in
* URI : <https://reqres.in/api/users?page=2>
* Parámetros : /api/users
* After ? query string page=2
* Para poder probar una Api se necesita el endpoint y la URI.
* En caso de un Post por ejemplo si se necesita enviar un BODY
* Toda esta información se debe tener en las especificaciones de las APIS
* En el documento de Excel se pueden revisar diferentes Endpoits para probar. En este momemento vamos a probar varias de ellas creando una nueva colección.
* 1. All users Data



* Si la respuesta es buena, se recibirá el status 200
* En el browser se pueden testear las APIs, pero no se verán bien, esto mas que todo paras las GET
* 2. Retorna solo los datos de un usuario

Texto

Descripción generada automáticamente

* 3. User creation – En este caso como es un post debemos agregar un body. Se debe especificar como enviar el body, en este caso seleccionaremos Raw y JSon format

Captura de pantalla de un celular

Descripción generada automáticamente

En este caso se tiene un status 201 lo que significa que fue creado exitosamente

* 4. Login - Se hace un post enviando en el body un email y un password. Se valida que exista, no se crea nada.

Captura de pantalla de un celular

Descripción generada automáticamente

* 5. Wrong login – SE usa el mismo request del punto anterior pero en este se le envían menos datos a ver la respuesta. El status responde con un 400 , lo que indica que está mal

Captura de pantalla de un celular

Descripción generada automáticamente

Lo que se hace con este tipo de request es pegarla la Api, esta va a la base de datos y valida si ese correo y contraseña están allí y retorna la respuesta a la API y la Api al usuario.

* Los headers traen información adicional, información sobre la API.
* 6. All employee Data

Captura de pantalla de un celular

Descripción generada automáticamente

* 7. Single employee data

Texto

Descripción generada automáticamente

* 8. Create new employee

Captura de pantalla de un celular

Descripción generada automáticamente

* 9. Update employee record

Captura de pantalla de un celular

Descripción generada automáticamente

* 10. Delete employee data

Captura de pantalla de un celular

Descripción generada automáticamente

**4. Testing API’s, Validations, Workflows and order of execution in Postman:**

* Como probar las APis localmente , añadir puntos de verificación , worflows , pe request scripts
* Se utilizará localmente una Api en formato .jar . En este caso será “studentAPI-jar” que se encuentra en el archivo de la sección. Se tiene un archivo de configuración
* Para ejecutar el archivo se va a la consola, en esa se va a la carpeta que contiene el archivo y se uusa el comando : java -jar “NombreArchivo”.jar. Se presiona enter y se empieza a ejecutar. No se puede cerrar la ventana del CMD porque se cierra el api

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

* - En este caso no es posible abrir bien el archivo.
* Si la Api corre, se pueden empezar a hacer request desde postman. En el archivo de Excel se pueden ver los llamados que se harán
* Luego se van a agregar los validation points, pueden ser encontrados en el archivo de Excel.
* En Postman se va a la ventana TEST. Se debe poner en JavaScrips, pero en este caso solo usaremos unas sentencias sencillas

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Para el POSt se usan las siguientes validaciones

Captura de pantalla con la imagen de una pantalla

Descripción generada automáticamente

* Para el GET se usan las siguientes validaciones. El contnt type viene en el header

Texto

Descripción generada automáticamente

* Para el PUT se usan las siguientes validaciones:

Captura de pantalla de un celular

Descripción generada automáticamente

* Para el Delete la siguiente validaciones:



* Las validaciones se pueden agregar como Scripts, lo que hicimos anteriormente, o también se pueden agregar como funciones. Estas funcions son un poco mas complicadas para alguien que no sabe de JavaScript. Ambas se pueden usar y tienen el mismo resultado. En el archivo de Excel se pueden ver los ejemplos
* Luego de enviar el request se puede ver el resultado de los Teste en la respuesta pestaña Test Results. En este caso se probaron en otra request ya que la que se esta usando no

funcionó.

Captura de pantalla de un celular

Descripción generada automáticamente

* También se puede agregar un orden de ejecución, es decir, que se ejecuten las request que se quieran una luego de la otra. Postman por defecto corre la colección completa e orden.
* Para correr tola la coleccion se le da en los 3 puntos que esta tiene al lado derecho y se clieckea en Run collection

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* Aparece una nueva ventana con todos los Requess que se tienen y se pueden seleccionar cuales se quieren usar.

Texto

Descripción generada automáticamente

* Se úede agregar cuentas interacciones. En el caso del delay se usa cuando Apis tiene un tiempo largo de respuesta y se necesita esperar un poco. Para iniciar se le da en RUN.
* Luego de ejecutar se pueden ver los resultados, mas que todo de los test que se tiene en cada Request

Texto

Descripción generada automáticamente

* Los request se pueden ordenar manualente, pero también se les puede agregar una línea de cogido al test para que se ejecute a continuación el request se se quiera. En el Excel se pueden ver las líneas de código. Para esto se usa el nombre exacto de la erquest que viene
* Si no se quiere que se ejecute nada después, simplemente se le pone “null”.

postman.setNextRequest("GET AFTER PUT");

* **Pre-request Script:** Antes que se ejecute el request, se pueden tener ciertas cosas que se puedan validar, un script que se correrá antes.
* Los Pre request scripts, los test y las autorizaciones pueden ser creados en 3 niveles: a nivel del request, a nivel del folder o a nivel de colección.

**5. Postman variables, Data Driven Testing, Working with collections & Environments:**

* Se puede crear una variable para no estar cambiando cada request cada ez que se debe hacer algo, simplemente se puede tener una variable, cambiarla en un solo lugar y así se actualiza en todos.
* Hay 2 tipos de variables: Variables de colección y variables globales, Estas últimas también llamadas Variables de entorno.
* En vez de usar por ejemplo una misma url en todas las request, se puede llevar a una variable.
* Para acceder a las variables se va a la colección y se le da en los 3 puntos, luego editar y se puede ver la pestaña de variables.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* En la tabla que aparecen se ponen los valores que se quieren para las variables

Interfaz de usuario gráfica, Texto, Aplicación, Sitio web

Descripción generada automáticamente

* Para usar las variables se va a la request, y se remplaza, en este caso el url.
* Para poder usar la variable se debe poner entre {{}}

Interfaz de usuario gráfica

Descripción generada automáticamente

Pantalla de un celular con letras

Descripción generada automáticamente

* Esta variable solo se puede usar en esta colección.
* Las variables también pueden ser creadas para usar en el body.
* Otra forma de crear las variables es desde el Pre-request scripts de la colección

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Se crean unas nuevas variables para usarlas en el body

Captura de pantalla de un celular

Descripción generada automáticamente

Pantalla negra con letras blancas

Descripción generada automáticamente con confianza media

La respuesta llega con los nombres deseados

* Las variables de colección solo se puden usar dentro de esa misma colección. Las variables globales pueden ser usadas por todos las colecciones de Postman que se tengan
* Para crear variables globales se va a la esquina derecha y se prsiona el ojo. Allí se pueden agregar las variables globales o crear un nuevo ambiente.

Texto

Descripción generada automáticamente

* Las variables globales se llaman de la misma manera {{}}
* **Data driven Testing:** Se usa al misma request múltiples veces. Se debe especificar la data en alguna locación o archivo
* Esto puede usarse desde un JSON o un SCV file
* Se va a usar para el primer ejemplo, los archivos que están en la carpeta DDT Test 1. Se tienen ambos formatos, JSON and CSV. La idea es crear ese caso con diferentes emails y diferentes usuarios traídos desde el archivo
* Se deben cambiar los datos para que se pueden recibir desde el archivo, en este caso el json

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

* Luego se debe ir al runner. # puntos y luego run collections. En el numero de ieraciones se ebe poner 3 en este caso que son los datos que están en el archivo. También en esta pantalla se debe agregar el archivo que se quiere. Se selecciona el formato.
* Presionando en Previes se puede ver los datos que se tienen para la prueba

Texto

Descripción generada automáticamente con confianza media

* Por ultimo se le da en correr

Captura de pantalla de un celular

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Lo mismo se puede hacer con un archivo CSV
* El ejemplo 2 no se podrá hacer porque es en la Api que no funcionó.
* En Postman se pueden exportar o importar colecciones. En la carpeta de la clase se puede encontrar una colección para poder importar. La colección importada se llama Postman API.
* Para poder usar esta Api se debe tener una key especial. Seguir los pasos para generar una key. Se puede agregar como global la llave para que se use siempre que se quiera.

**6-. Testing Real Time Rest API's in Postman ( Facebook Graph API's & CAT API's):**

* Con la APi de facebook se pueden realizar directamente muchas cosas como: Postear un mensaje, leer los mensajes del post, comentar un post, borrar comentarios, borrar mensajes en el post y muchas mas.
* Para poder hacerlo, primero debemos tener una cuenta de Facebook.
* En el documento se pueden ver una pagina de Facebook para developers : developer.facebook.com
* Alli se debe logear y crear una app

Interfaz de usuario gráfica, Aplicación, Sitio web

Descripción generada automáticamente

* Yendo a Tool se puede encontrar el explorador de Graph Api

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Ingresamos al explorador
* En el lado derecho se puede generar en token de acceso y se habilitan los permisos para lo que se quiere mostrar
* En este pantalla se pueden testear las API de Facebook.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* Se pueden seleccioar muchos otros campos para mostrar

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente con confianza baja

* Estas no son las API que vamos a usar.
* Vamos a la documentación de Garp api y bajamos hasta publish, y allí veremos las API que se tienen y como usarlas.
* Los casos de prueba están en el archivo de Excel, todos se pueden realizar desde el Graph Api de Facebook. Tener muy en cuenta los IDs
* Todo esto también se puede hacer desde postman
* En la pantalla del Graph Api en la parte interior se pueden presionar en el botón “Code” y se puede tomar el curl. Este se lleva a postman +, se selecciona importar y luego importar raw text.
* También en esta calse se tiene CAR API. Es una api de psotman, libre y abierta

<https://documenter.getpostman.com/view/4016432/RWToRJCq#intro>

* El curl de cada request se puede copiar o en la misma pagina al lado sueprior derecho se pueden encontrar un botón llamado “Run in postman” el cual importa directamente todas las request en tu maquia o en tu postman web
* Para poder correr las request del CAT AI se debe poner el endpoint como variable de colección¨y también la key.
* El enpoint sería https://api.thecatapi.com
* Para obtener la key se debe ingresar a <https://thecatapi.com/> luego de logearse llega un correo con la key.
* Se pone la key en las variables de la colección
* Cuando se tiene un POST por ejemplo, se tiene un campo en el body que es un archivo, allí se puede poner la imagen que se quiere subir

Captura de pantalla de un celular

Descripción generada automáticamente

* También se pueden subir imágenes como texto, se debe poner el url del lugar donde se tengan.

Captura de pantalla de un celular

Descripción generada automáticamente

**7. How to run Postman Collections from CLI & Testing SOAP Services in Postman:**

* Postman soporta servicios REST y Soap.
* En esta clase se verá como correr colleciones/request desde la línea de comandos
* Para eso se necesitan instalar: node.js, Newman
* Se debe instalar npm pero este se intala cuando se intala node.
* Para instalar node se va a <https://nodejs.org/en/download/> se descarga y se instala normalmente.
* Luego se debe ir a las variables de entorno y agregar node. Se debe agregar en el PATH la rruta hacia el bin y también la rusa solo hasta el nodejs

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto

Descripción generada automáticamente

* Para instalar Newman se usa un simple comando en el cmd usando el componente npm



* Se deben entonces exportar las colecciones
* Para correrla se pueden hacer de diferentes formas:

1. Desde la línea de comandos: Se debe iniciar la linea de comandos en la carpeta donde se encuentra la colección exportada y se usa el comando Newman run “nombre coleccion”

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

1. Con un reporte generado en HTML: para esto se usa el miso comando pero se le agregan otros parámetros: Newman run “nombre coleccion” -r html. Al lanzarlo puede que no este instalado entonces se debe instalar.

Al correrlo se crea una carpeta nueva llama Newman, y allí se encuentra el archivo HTML

<file:///C:/Users/ASUS/Documents/Cursos/Learn%20API%20&%20Microservices%20Testing/Seccion%202/7/newman/newman-run-report-2022-06-28-03-10-44-305-0.html>

Alli se puede ver detalladamente el reporte.

1. Ejecutar una colección remotamente: en este caso no se necesita la colección, solo un link. Para esto se va a la colección de postman, se da en los 3 puntos y se le da en share

Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamente

Esto se puede pasar a las personas que se quieran y pueden correr la colección via el link que se envía.

<https://www.getpostman.com/collections/31ebf34cd832e42dc439>

Esto se puede correr desde la linea de comandos. Se usa entonces: Newman run linkcoleccion



1. Se puede también crear el reporte html de la misma forma, agrando al final -r html

Todas estas formas pueden ser usadas cuando no se tiene el ambiente de postman, simplemente usando la linea de comandos se pueden hacer las pruebas necesarias.

* Cuando hacemos peticiones con REST necesitamos un URAL, payload, body, y se puede solo Post , Get.. etc
* Para SOAP necesitamos el WSDL (Web services descriptions language). Los servicios SOAP también se pueden probar usando Postman
* Para esto el WSDL es lo mas importante. El WSDL está en formato XML
* En este caso se usará uno que esta en los documentos de la clase

<https://www.dataaccess.com/webservicesserver/NumberConversion.wso?WSDL>

* Este servicio tiene 2 opciones, toma un numero y lo regresa en letras y el otro es que toma un número y lo devuelve en dólares.
* Vamos a Postman y creemos una nueva colección.
* Al copiar el URL del WSDL se pone inmediatamente como un SOAP

Captura de pantalla de un celular

Descripción generada automáticamente

* Para verlo mejor vamos a instalar una extensión en Chrome, la extensión se llama Wizdler. Este se usa para obtener el body del WSDL
* Se abre entonces el WSDL, y luego la extensión, ella msotrará las APIs que tiene, seleccionaremos la primera

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

-Al clickearla retornará el body de ese request el cual copiamos y lo ponemos en el body del request en postman

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Texto, Chat o mensaje de texto

Descripción generada automáticamente

* La forma más fácil de usarlo, es importando el link del WSDL, Desde allí se carga todo y no hay que hacer nada más.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente