

**CARRERA PROFESIONAL DE ANÁLISIS DE
SISTEMAS
MÓDULO SOPORTE TÉCNICO Y OFIMÁTICO**



VALLE GRANDE
INSTITUTO DE EDUCACIÓN SUPERIOR TECNOLÓGICO PRIVADO

Azure Cognitive Services - Content Moderator - Trabajo 7

Estudiante

Juan Condori Jara

Profesor

Giancarlo Valencia Sotomayor

Cañete - Perú

Julio de 2022

TABLA DE CONTENIDO

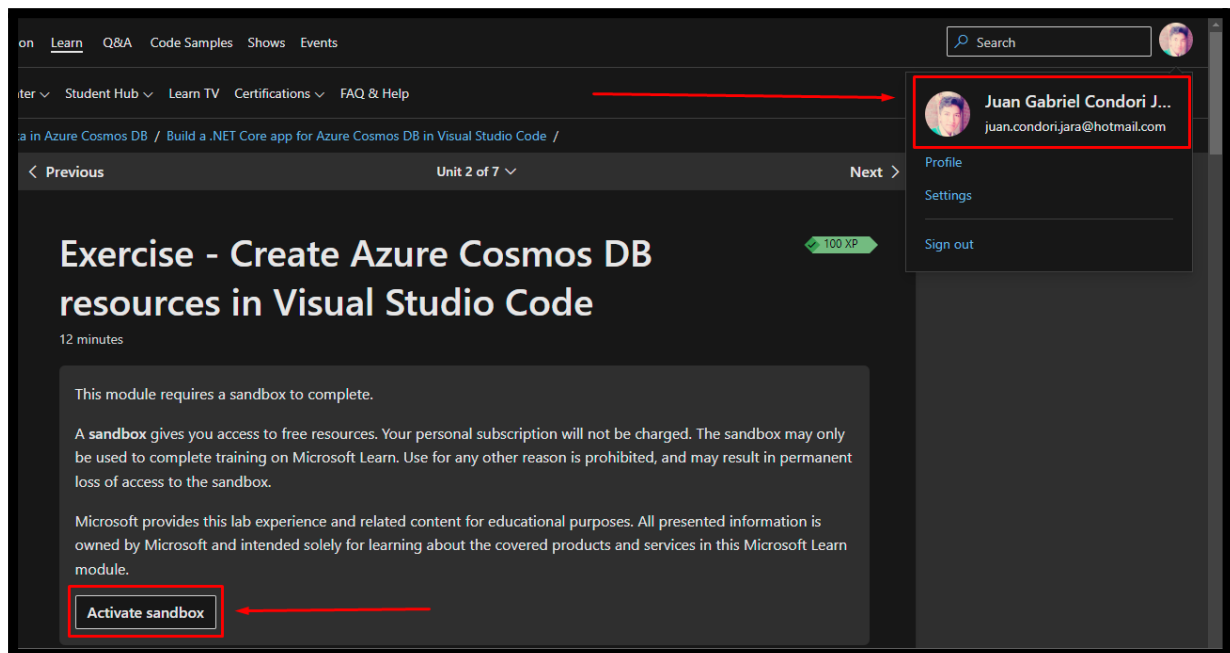
Activar Sandbox Azure.....	3
Cambiar el directorio de perfil.....	4
Crear un Recurso en Azure.....	5
Probando el servicio Rest.....	9
Probando el servicio Rest en Postman.....	13
Consumiendo el servicio Rest en JAVA con Apache Netbeans.....	15
Conclusiones.....	22
Consideraciones.....	22
Anexo.....	22
Referencias.....	22

Azure Cognitive Services - Content Moderator

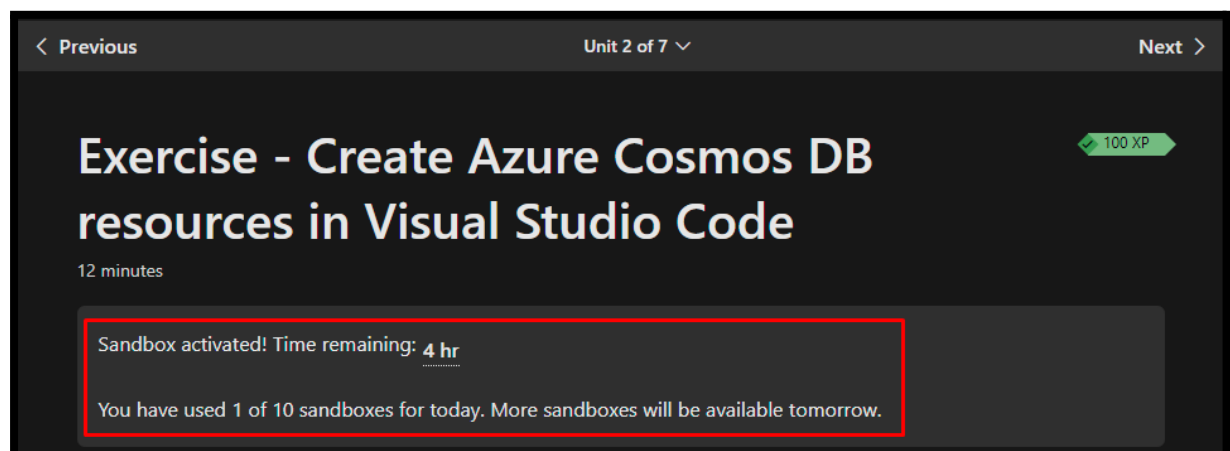
Activar Sandbox Azure

Necesitamos una cuenta Outlook o Hotmail activa, nos dirigimos al link y hacemos clic en Activate Sandbox

<https://docs.microsoft.com/en-us/learn/modules/build-cosmos-db-app-with-vscode/2-setup-cosmosdb>



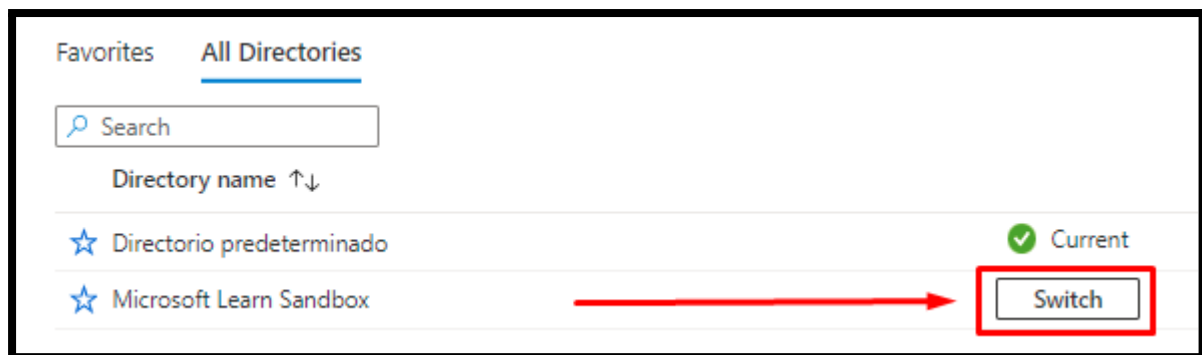
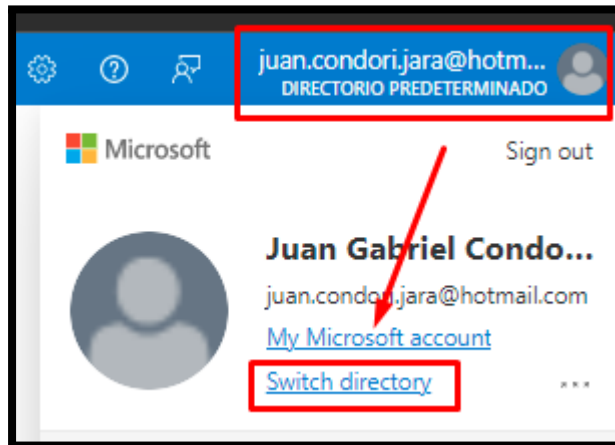
Sólo tenemos 4 horas de uso y 10 espacios aislados por día en la cuenta.



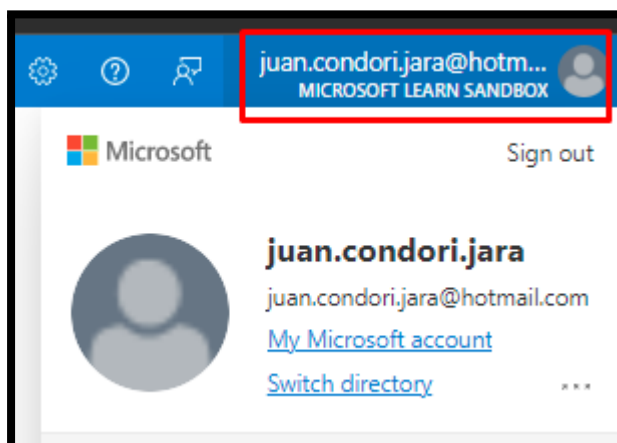
Cambiar el directorio de perfil

Pasamos a cambiar la cuenta en el directorio del perfil.

<https://portal.azure.com/>

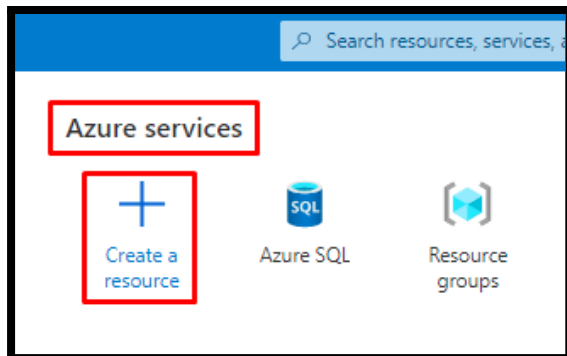


Debemos tener así el directorio de la cuenta, como MICROSOFT LEARN SANDBOX.

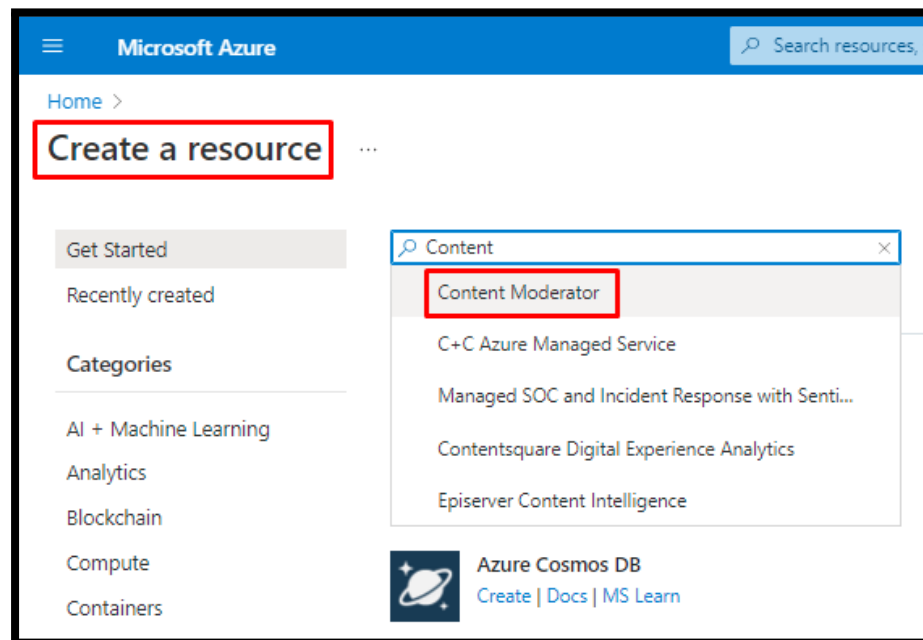


Crear un Recurso en Azure

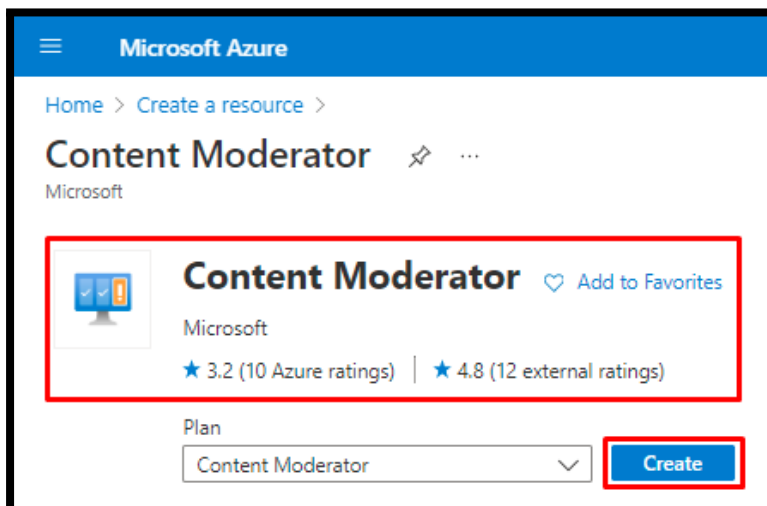
Elegimos la opción de Create a resource



Buscamos el servicio cognitivo de Content Moderator y lo elegimos.



Le damos click a Create.



Rellenamos los siguientes datos sólo tendremos que colocar un nombre válido y que no esté en uso.

Instance Details

Region ⓘ East US

Name * ⓘ Demo

- ✖ The sub-domain name is already used. Please pick a different name.
- ✖ The provided sub-domain name is either invalid or already in use. Please pick a different name.

Rellenado todo los datos le damos en Review + Create.

Microsoft Azure Search resources, services, and docs (G+)

Home > Create a resource > Content Moderator >

Create Content Moderator

videos. Use the built-in review tool for best results.

[Learn more](#)

Project Details

Subscription * ⓘ Concierge Subscription

Resource group * ⓘ learn-c2cfd320-441a-424d-9dad-cd8a1da5c908 [Create new](#)

i Cognitive services resource creation requires subscription registration, we detected that your selected subscription did not register cognitive services resource type before, we will help you to register cognitive services resource type when you select a subscription in subscription dropdown. Click to learn more how to check registration state for your selected subscription.

Instance Details

Region ⓘ East US

Name * ⓘ DemoContentModerator123 ✓

Pricing tier * ⓘ Standard S0 (10 Calls per second)

[View full pricing details](#)

[Review + create](#) < Previous Next : Network >

Una vez validado, le damos clic en Create.

Microsoft Azure

Home > Create a resource > Content Moderator >

Create Content Moderator

Validation Passed

Basics Network Identity Tags Review + create

TERMS

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for additional details.

Basics

Subscription	Concierge Subscription
Region	East US
Name	DemoContentModerator123
Pricing tier	Standard S0 (10 Calls per second)

Network

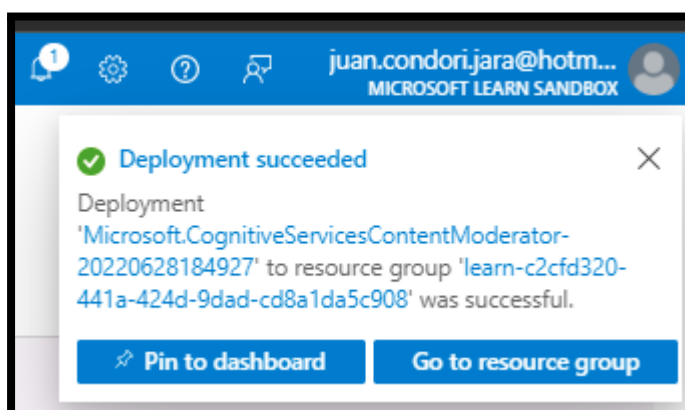
Type	All networks, including the internet, can access this resource.
------	---

Identity

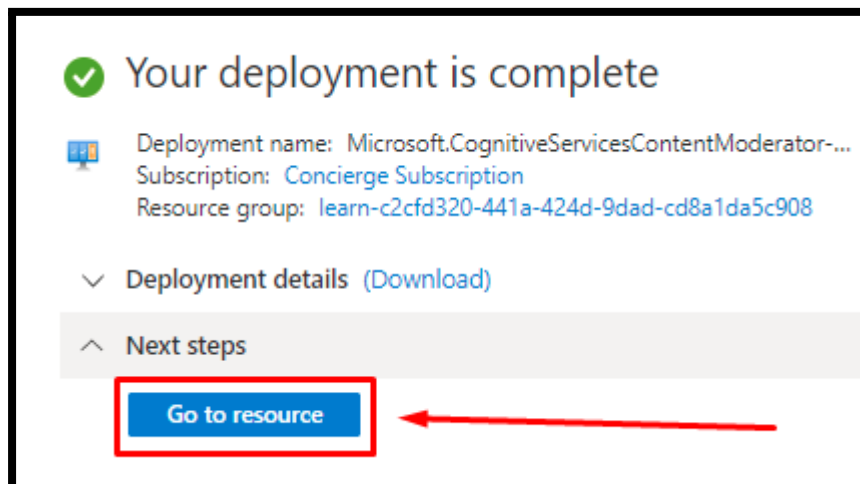
Identity type	None
---------------	------

Create < Previous Next Download a template for automation

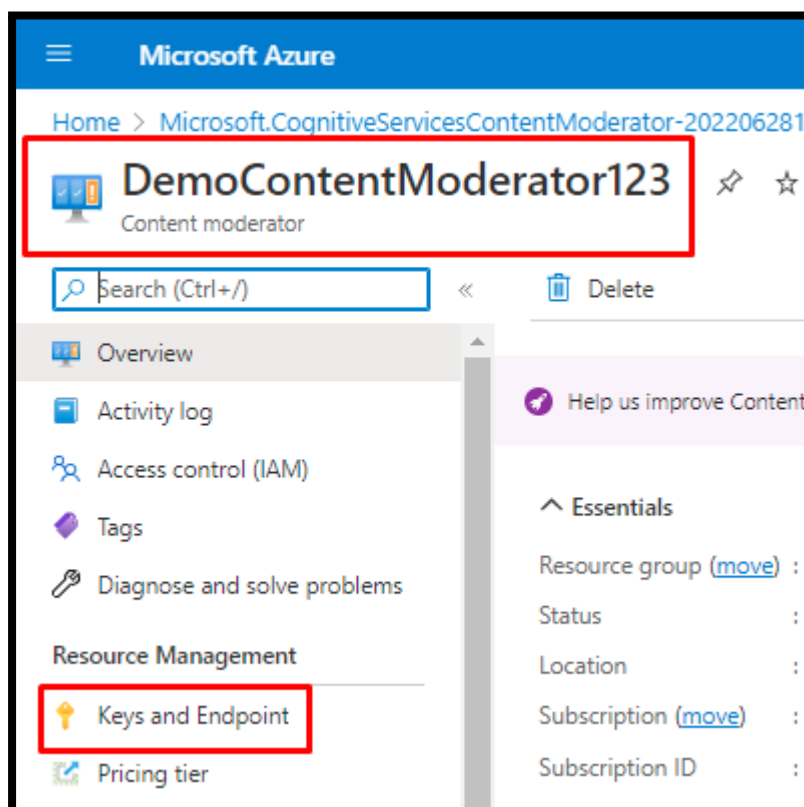
Esperamos a que se termine de crear.



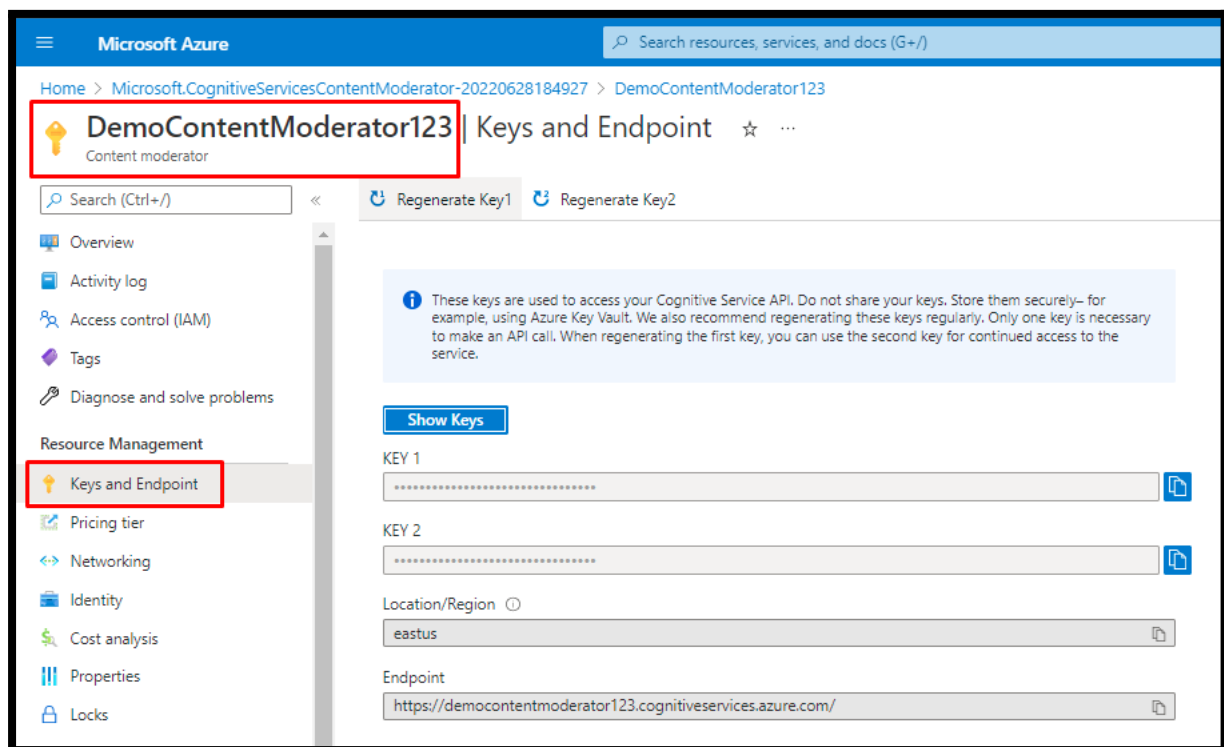
Nos dirigimos al recurso que acabamos de crear.



Entramos a Keys and Endpoint.



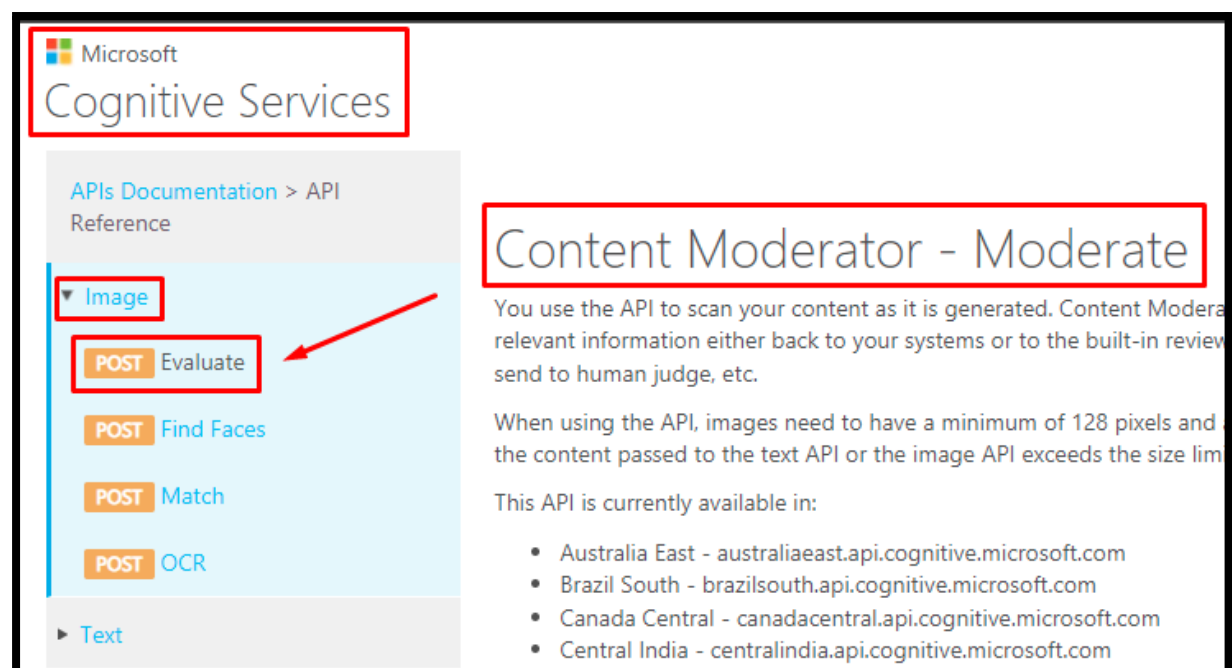
Nos debe mostrar los Keys and Endpoint.



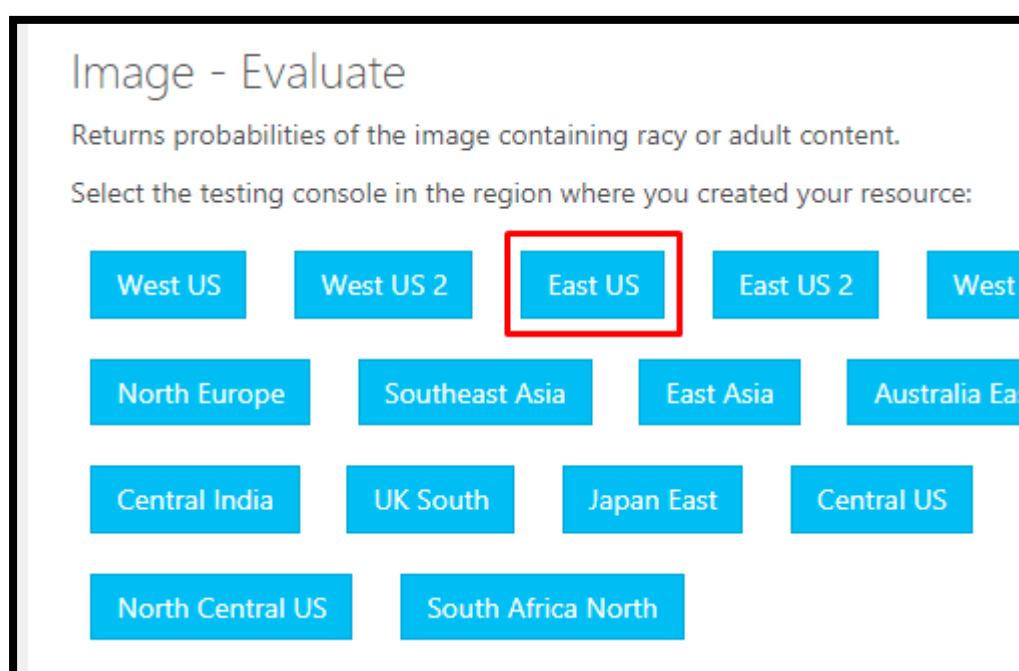
Probando el servicio Rest

Entramos al siguiente enlace.

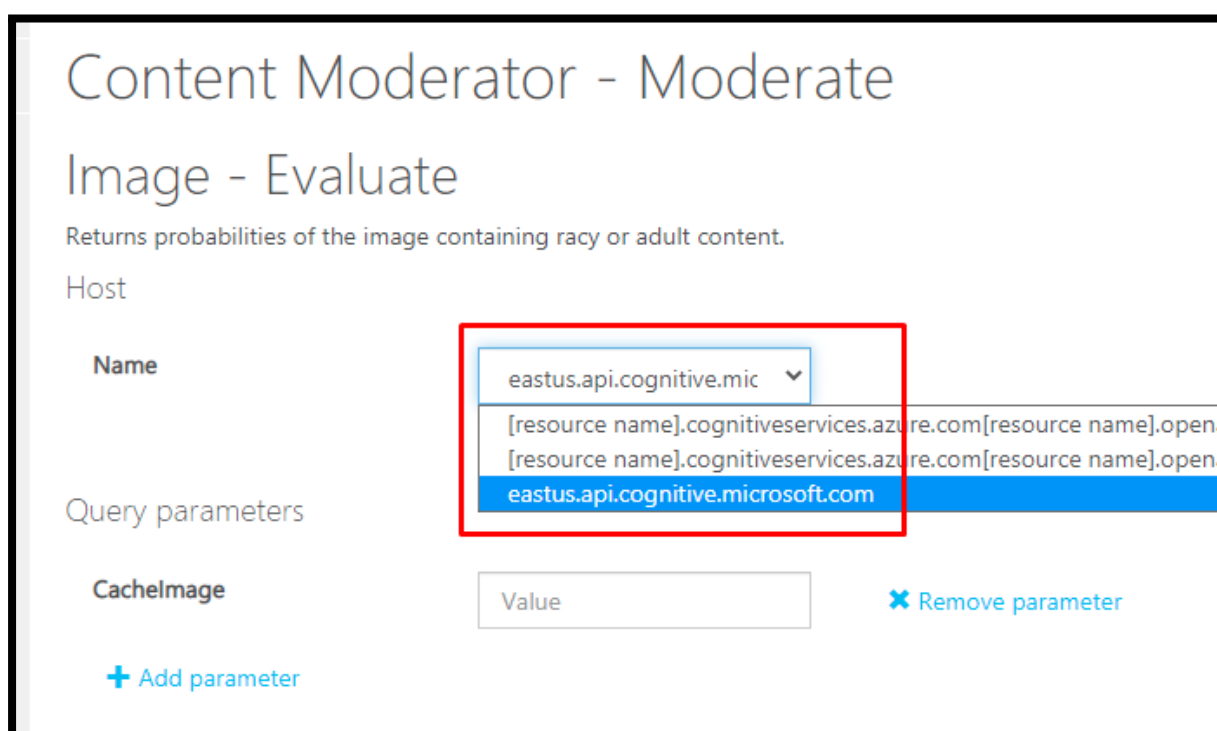
<https://westus.dev.cognitive.microsoft.com/docs/services/57cf753a3f9b070c105bd2c1/operations/57cf753a3f9b070868a1f66c>




Seleccionamos la consola de prueba en la región donde creamos nuestro recurso.



En Host seleccionamos el Name.



Copiamos el KEY 1.



Show Keys

KEY 1

.....

KEY 2

.....

Location/Region ⓘ

eastus

Endpoint

https://democontentmoderator123.cognitiveservices.azure.com/

Copy to clipboard

Y lo pegamos en Ocp-Apim-Subscription-Key.



Headers

Content-Type

application/json

✕ Remove header

Ocp-Apim-Subscription-Key

.....

+ Add header

Probamos y damos en Send.



Request URL

https://eastus.api.cognitive.microsoft.com/contentmoderator/moderate/v1.0/ProcessImage/Evaluate

HTTP request

POST https://eastus.api.cognitive.microsoft.com/contentmoderator/moderate/v1.0/ProcessImage/Evaluate HTTP/1.1

Host: eastus.api.cognitive.microsoft.com

Content-Type: application/json

Ocp-Apim-Subscription-Key:

{

"DataRepresentation": "URL",

"Value": "https://moderatorsampleimages.blob.core.windows.net/samples/sample.jpg"

}

Send

Nos da la siguiente respuesta.

Response content

```
Transfer-Encoding: chunked
csp-billing-usage: CognitiveServices.ContentModerator.Transaction=1
x-envoy-upstream-service-time: 915
apim-request-id: 20b3eed1-7f4a-4b11-8b9e-731d8c15afcf
Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
x-content-type-options: nosniff
Date: Wed, 29 Jun 2022 00:38:45 GMT
Content-Type: application/json; charset=utf-8
```

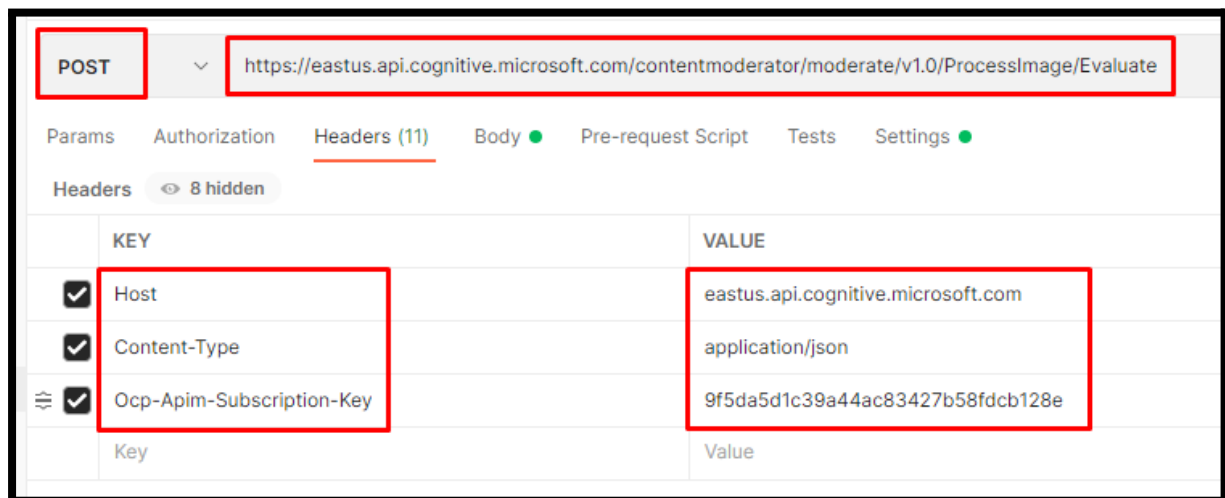
```
{
  "AdultClassificationScore": 0.025337669998407364,
  "IsImageAdultClassified": false,
  "RacyClassificationScore": 0.05301359295845032,
  "IsImageRacyClassified": false,
  "Result": false,
  "AdvancedInfo": [{
    "Key": "ImageDownloadTimeInMs",
    "Value": "802"
  }, {
    "Key": "ImageSizeInBytes",
    "Value": "273405"
  }],
  "Status": {
    "Code": 3000,
    "Description": "OK",
    "Exception": null
  },
  "TrackingId": "20b3eed1-7f4a-4b11-8b9e-731d8c15afcf"
}
```

Probando el servicio Rest en Postman

Colocaremos los siguientes datos en Postman.



Completamos los datos en el Headers en Postman.



Colocamos lo siguiente en el Body en Postman y le damos clic en Send.



Nos da la siguiente respuesta.

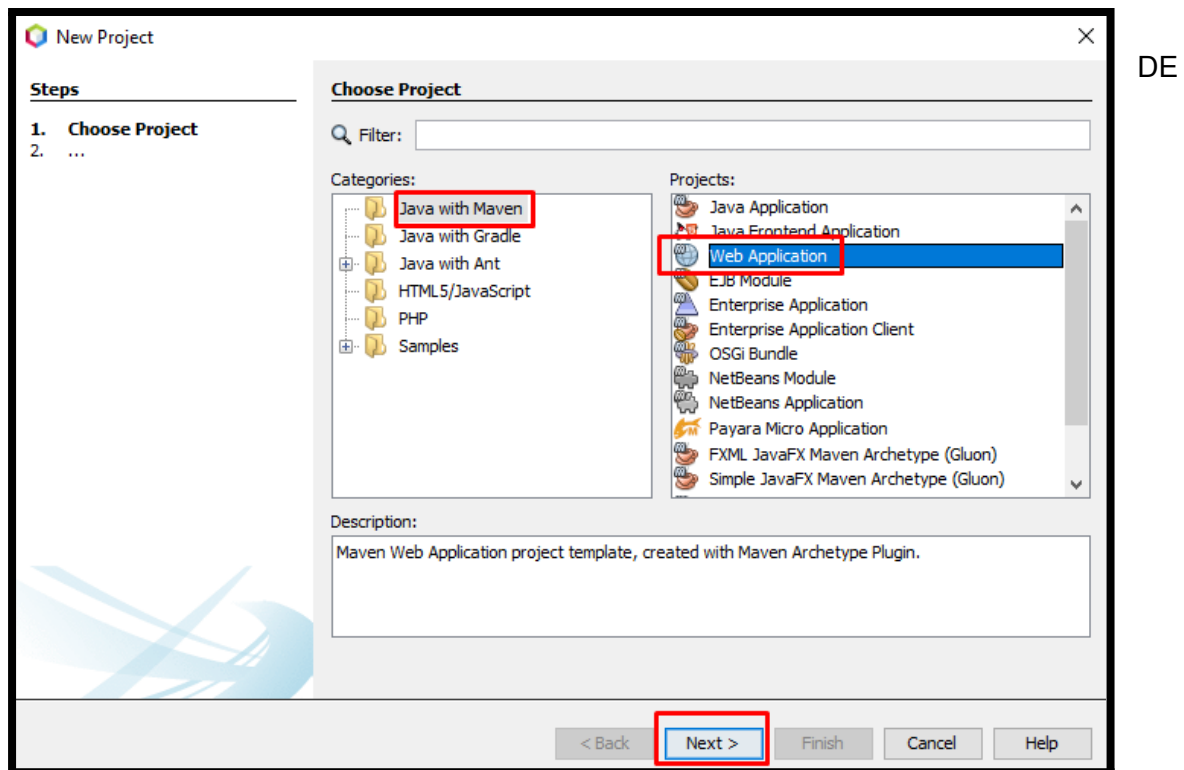


The screenshot shows a web browser's developer tools interface. The 'Body' tab is selected, displaying a JSON response. The JSON is formatted in a 'Pretty' view. The response contains classification scores, image status, advanced information (download time and size), status details, and a tracking ID.

```
1  {
2    "AdultClassificationScore": 0.025337669998407364,
3    "IsImageAdultClassified": false,
4    "RacyClassificationScore": 0.05301359295845032,
5    "IsImageRacyClassified": false,
6    "Result": false,
7    "AdvancedInfo": [
8      {
9        "Key": "ImageDownloadTimeInMs",
10       "Value": "808"
11      },
12      {
13        "Key": "ImageSizeInBytes",
14        "Value": "273405"
15      }
16    ],
17    "Status": {
18      "Code": 3000,
19      "Description": "OK",
20      "Exception": null
21    },
22    "TrackingId": "be1e0a58-a61c-4474-bb8d-45493deda90f"
23  }
```

Consumiendo el servicio Rest en JAVA con Apache Netbeans

Creamos un proyecto MAVEN en Apache Netbeans.

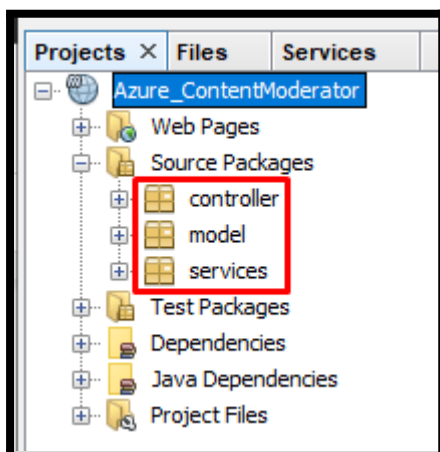


Utilizaremos las siguientes dependencias en nuestro archivo pom.xml.

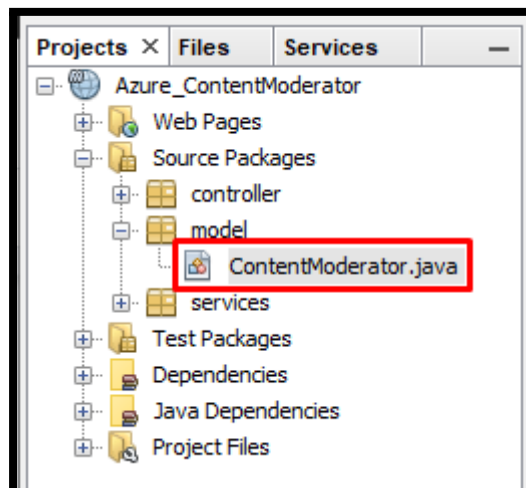
```
<dependencies>
  <dependency>
    <groupId>jakarta.platform</groupId>
    <artifactId>jakarta.jakartaee-api</artifactId>
    <version>${jakarta.jakartaee-api.version}</version>
    <scope>provided</scope>
  </dependency>
  <dependency>
    <groupId>org.primefaces</groupId>
    <artifactId>primefaces</artifactId>
    <version>11.0.0</version>
  </dependency>
  <dependency>
    <groupId>org.projectlombok</groupId>
    <artifactId>lombok</artifactId>
    <version>1.18.20</version>
    <scope>provided</scope>
  </dependency>
  <dependency>
    <groupId>com.google.code.gson</groupId>
    <artifactId>gson</artifactId>
    <version>2.9.0</version>
    <type>jar</type>
  </dependency>
</dependencies>
```

```
<dependency>
  <groupId>org.apache.httpcomponents</groupId>
  <artifactId>httpcore</artifactId>
  <version>4.4.13</version>
  <type>jar</type>
</dependency>
<dependency>
  <groupId>com.squareup.okhttp3</groupId>
  <artifactId>okhttp</artifactId>
  <version>4.9.3</version>
</dependency>
<dependency>
  <groupId>org.apache.httpcomponents</groupId>
  <artifactId>httpclient</artifactId>
  <version>4.5.13</version>
  <type>jar</type>
</dependency>
<dependency>
  <groupId>org.json</groupId>
  <artifactId>json</artifactId>
  <version>20160810</version>
</dependency>
</dependencies>
```

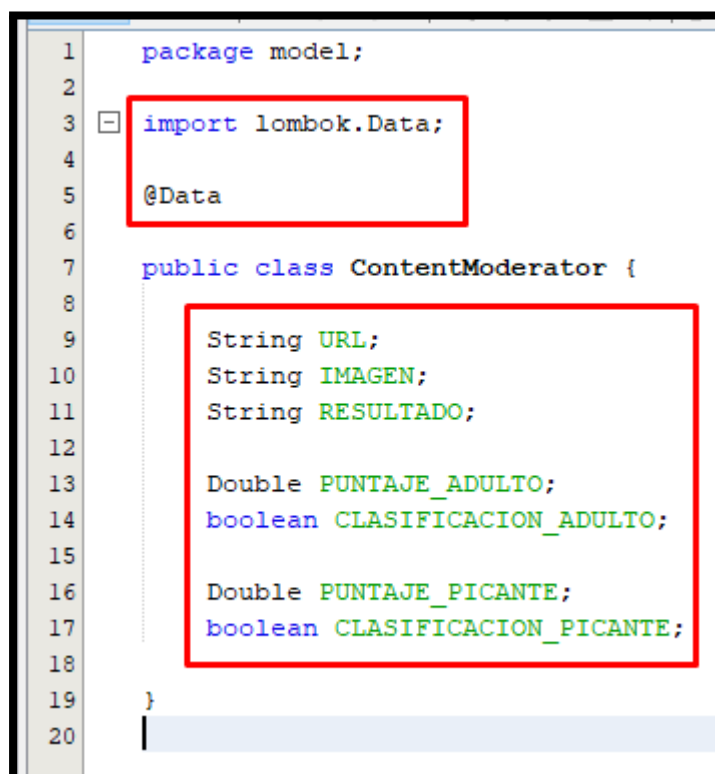
Creamos 3 paquetes, model, services y controller.



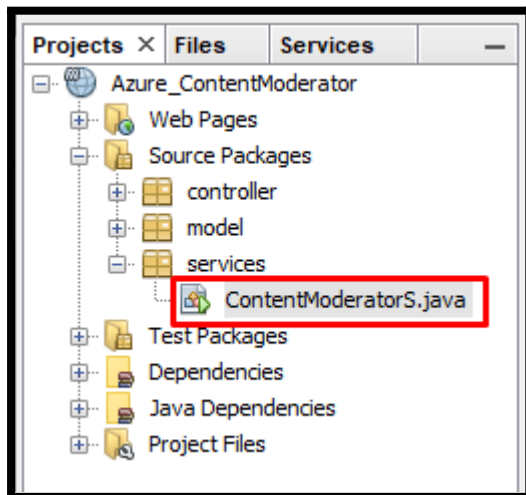
En el paquete model creamos un archivo .java (Java Class) llamado ContentModertor.



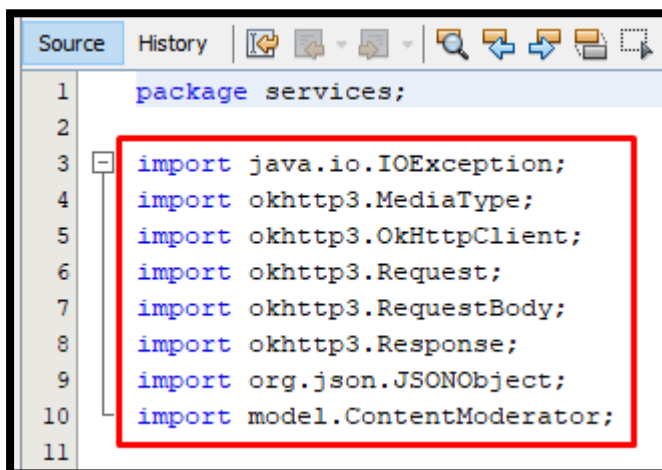
Con las siguientes variables e importando lombok para los Getters & Setters.



En el paquete services creamos un archivo .java (Java Main Class) llamado ContentModertorS.



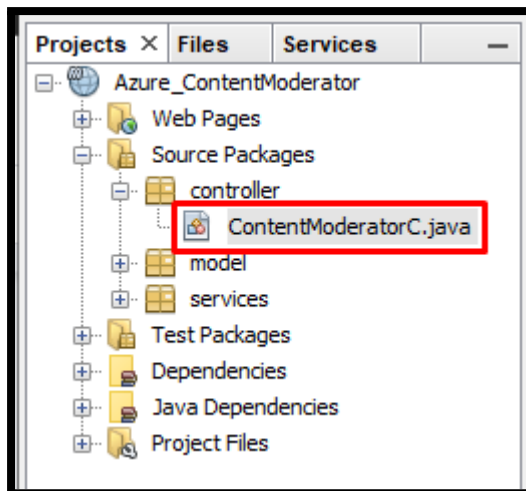
Con las siguientes importaciones que utilizaremos.



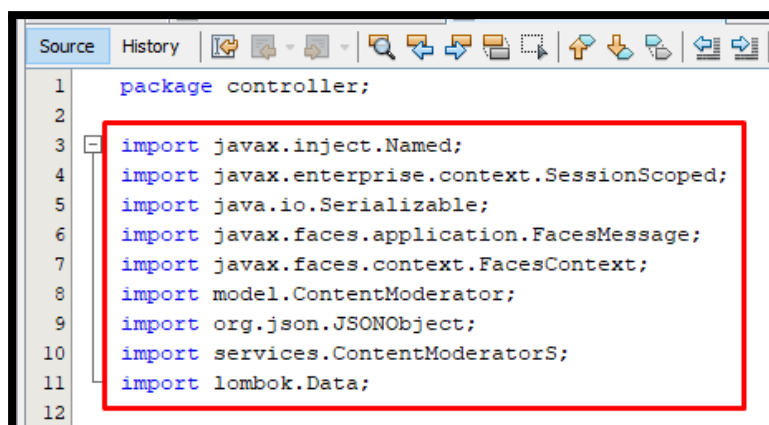
El siguiente método va a consumir el servicio de Azure Content Moderator.



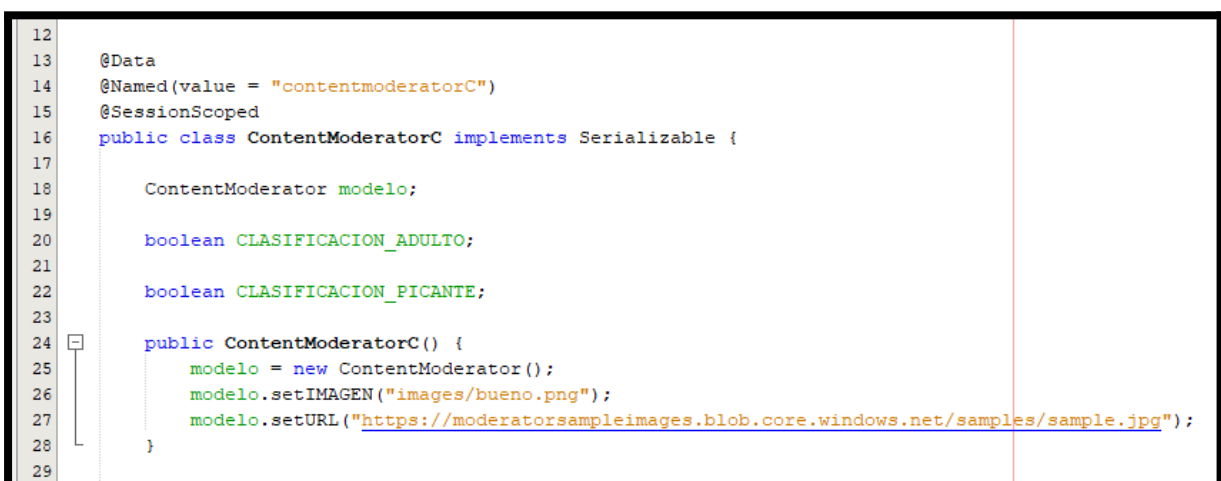
En el paquete controller creamos un archivo .java (JSF Managed Bean) llamado ContentModertorC.



Con las siguientes importaciones que utilizaremos.



Con las siguientes variables.



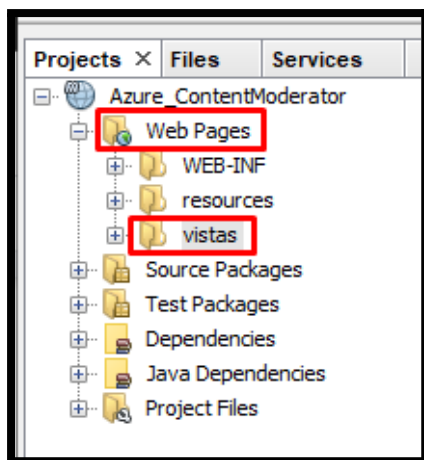
El siguiente método va a consumir el método del archivo de nuestro paquete services, y va a asignar los valores del servicio a nuestras variables de model y del mismo controller.

```
public void obtenerDatosImagen() throws Exception {
    try {
        JSONObject cadenaJson = ContentModeratorS.obtenerJsonImagen(modelo);
        modelo.setPUNTAJE_ADULTO(cadenaJson.getDouble("AdultClassificationScore"));
        modelo.setCLASIFICACION_ADULTO(cadenaJson.getBoolean("IsImageAdultClassified"));
        modelo.setPUNTAJE_PICANTE(cadenaJson.getDouble("RacyClassificationScore"));
        modelo.setCLASIFICACION_PICANTE(cadenaJson.getBoolean("IsImageRacyClassified"));

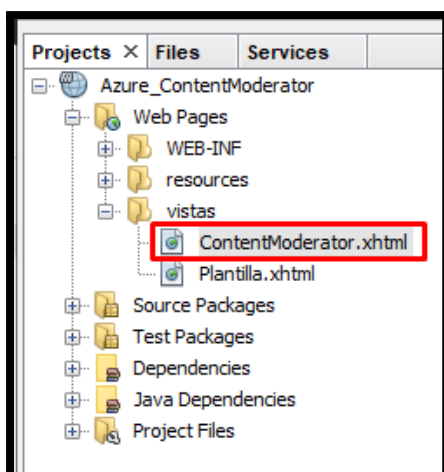
        CLASIFICACION_ADULTO = cadenaJson.getBoolean("IsImageAdultClassified");
        CLASIFICACION_PICANTE = cadenaJson.getBoolean("IsImageRacyClassified");

        if (CLASIFICACION_ADULTO == false && CLASIFICACION_PICANTE == false) {
            modelo.setRESULTADO("NADA OFENSIVO, la IMAGEN no es subido de tono ni para adultos");
            FacesContext.getCurrentInstance().addMessage(null, new FacesMessage(FacesMessage.SEVERITY_INFO, "NADA OFENSIVO", "La IMAGEN no es subido de tono ni para adultos"));
            modelo.setIMAGEN("images/bueno.png");
        }
        if (CLASIFICACION_ADULTO == true || CLASIFICACION_PICANTE == true) {
            modelo.setRESULTADO("OFENSIVO, la IMAGEN es subido de tono O para adultos");
            FacesContext.getCurrentInstance().addMessage(null, new FacesMessage(FacesMessage.SEVERITY_WARN, "OFENSIVO", "La IMAGEN es subido de tono O para adultos"));
            modelo.setIMAGEN("images/malo.png");
        }
        if (CLASIFICACION_ADULTO == true && CLASIFICACION_PICANTE == true) {
            modelo.setRESULTADO("MUY OFENSIVO, la IMAGEN es subido de tono y para adultos");
            FacesContext.getCurrentInstance().addMessage(null, new FacesMessage(FacesMessage.SEVERITY_FATAL, "MUY OFENSIVO", "La IMAGEN es subido de tono y para adultos"));
            modelo.setIMAGEN("images/malo.png");
        }
    } catch (Exception e) {
        System.out.println("Error en obtenerDatosIamgen: " + e.getMessage());
        e.printStackTrace();
    }
}
```

Desde Web Pages creamos un folder llamado vistas.



Creamos un archivo .xhtml llamado ContentModerator que será nuestro Front-End.



Nuestra vista final del archivo ContentModerator.xhtml

Al consultar una imagen nos dirá si la imagen tiene contenido subido de tono o para adultos.

AZURE - CONTENT MODERATOR - EVALUACIÓN DE IMÁGENES

NADA OFENSIVO
La IMAGEN no es subido de tono ni para adultos

Ingrese Link de la Imagen

<https://moderatorsampleimages.blob.core.windows.net/samples/sample.jpg>

CONSULTAR

PROCENTAJE ADULTO	CLASIFICACION ADULTO	PROCENTAJE PICANTE	CLASIFICACION PICANTE
0.025337669998407364	false	0.05301359295845032	false



NADA OFENSIVO, la IMAGEN no es subido de tono ni para adultos

AZURE - CONTENT MODERATOR - EVALUACIÓN DE IMÁGENES

OFENSIVO
La IMAGEN es subido de tono O para adultos

MUY OFENSIVO
La IMAGEN es subido de tono y para adultos

Ingrese Link de la Imagen

<https://www.qhubomedellin.com/wp-content/uploads/2021/08/only-678x381.jpg>

CONSULTAR

PROCENTAJE ADULTO	CLASIFICACION ADULTO	PROCENTAJE PICANTE	CLASIFICACION PICANTE
0.5035867094993591	true	0.7034647166728973	true



MUY OFENSIVO, la IMAGEN es subido de tono y para adultos

Conclusiones

- ❖ Content Moderator procesa el contenido ya sea texto, imagen o video, y devuelve los resultados con información relevante a la herramienta de revisión integrada, como Postman o la misma referencia de API de servicios cognitivos.
- ❖ Las imágenes deben tener un mínimo de 128 píxeles y un tamaño de archivo máximo de 4 MB, de lo contrario, devolverá un código de error que informa sobre el problema.
- ❖ Gracias al servicio cognitivo de Content Moderator podemos obtener información evaluada de contenidos en diversos archivos y tomar decisiones en base a los resultados.

Consideraciones

- ❖ En Primefaces sólo se implementó la imagen por URL.
- ❖ Se consumió el servicio cognitivo Content Moderator de Azure.
- ❖ El servicio devuelve las probabilidades en porcentaje, también en verdadero o falso.
- ❖ El servicio cognitivo Content Moderator es fácil y rápido de crear, para luego ser consumido.

Anexo

- ❖ Repositorio GitHub
https://github.com/juancondorijara/CognitiveServices_ContentModerator.git

Referencias

- ❖ Portal Azure
<https://portal.azure.com/>
- ❖ Aprende los conceptos de moderación de imágenes
<https://docs.microsoft.com/en-us/azure/cognitive-services/content-moderator/image-moderation-api>
- ❖ Referencia de Content Moderator API
<https://westus.dev.cognitive.microsoft.com/docs/services/57cf753a3f9b070c105bd2c1/operations/57cf753a3f9b070868a1f66c>
- ❖ Tutorial Azure Content Moderator
<https://youtu.be/-YF2iJQJwzI>