UD8 INTERFACES NATURALES. Reconocimiento facial

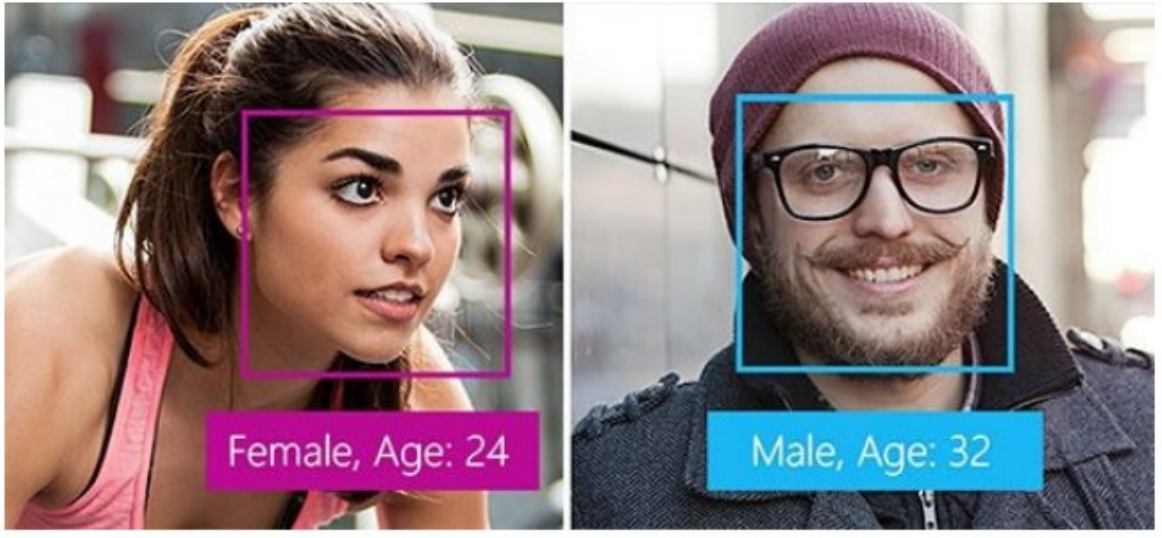
Ciclo: Desarrollo de Aplicaciones Multiplataforma
Módulo: Desarrollo de interfaces.

Contenidos

1. Funcionalidades de Face
2. Estructuras de datos
3. Información de las caras
4. Características del servicio.
5. Detección (Detect)
6. Verificación (Verify)
7. Identificación (Identify)
8. Recursos

Funcionalidades de Face

- Face es el servicio de reconocimiento facial de Azure Cognitive Services. Este servicio ofrece cinco operaciones relacionadas con el reconocimiento de rostros en imágenes:
  - **Detección (Detect):** Es la funcionalidad básica de Face, que permite detectar las caras en una imagen, y obtener información de cada una de las caras detectadas.

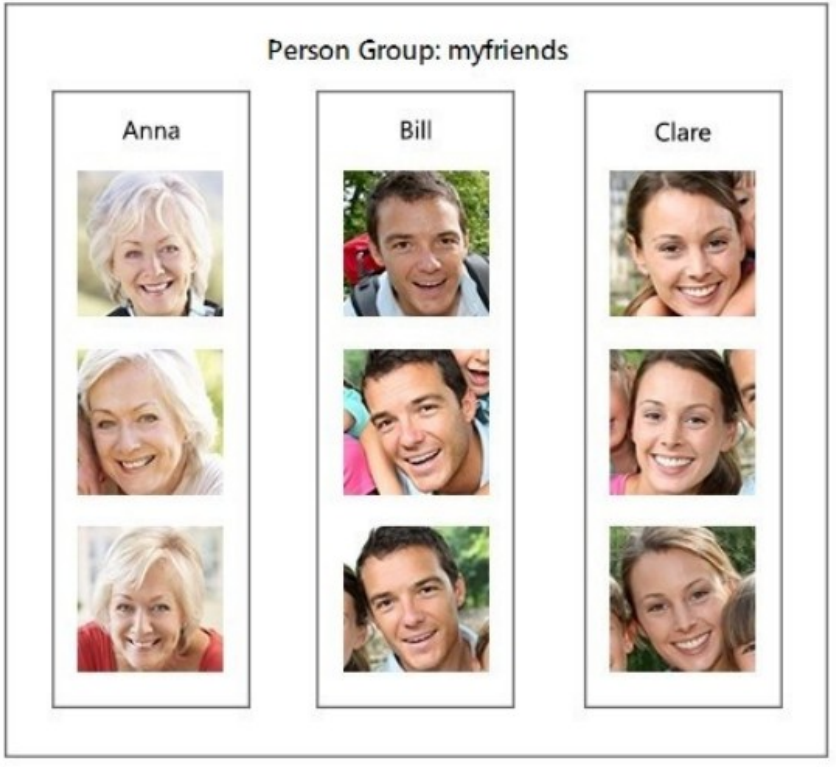


- **Caras similares (Find Similar):** Permite localizar caras similares a una dada dentro de un conjunto de caras. Por defecto se seleccionan caras que pertenezcan a la misma persona, pero si se indica un número mínimo de caras a obtener, se seleccionarán las más parecidas, aunque pertenezcan a otras personas.

- **Verificación (Verify):** Esta opción permite determinar si dos rostros pertenecen a la misma persona.

- **Agrupación (Group):** Permite agrupar un conjunto de caras por similitud (normalmente se crea un grupo para cada persona).

- **Identificación (Identify):** Esta funcionalidad permite identificar una persona a partir de una imagen con su cara. Para utilizarlo es necesario haber asociado a dicha persona un conjunto de imágenes de su cara.



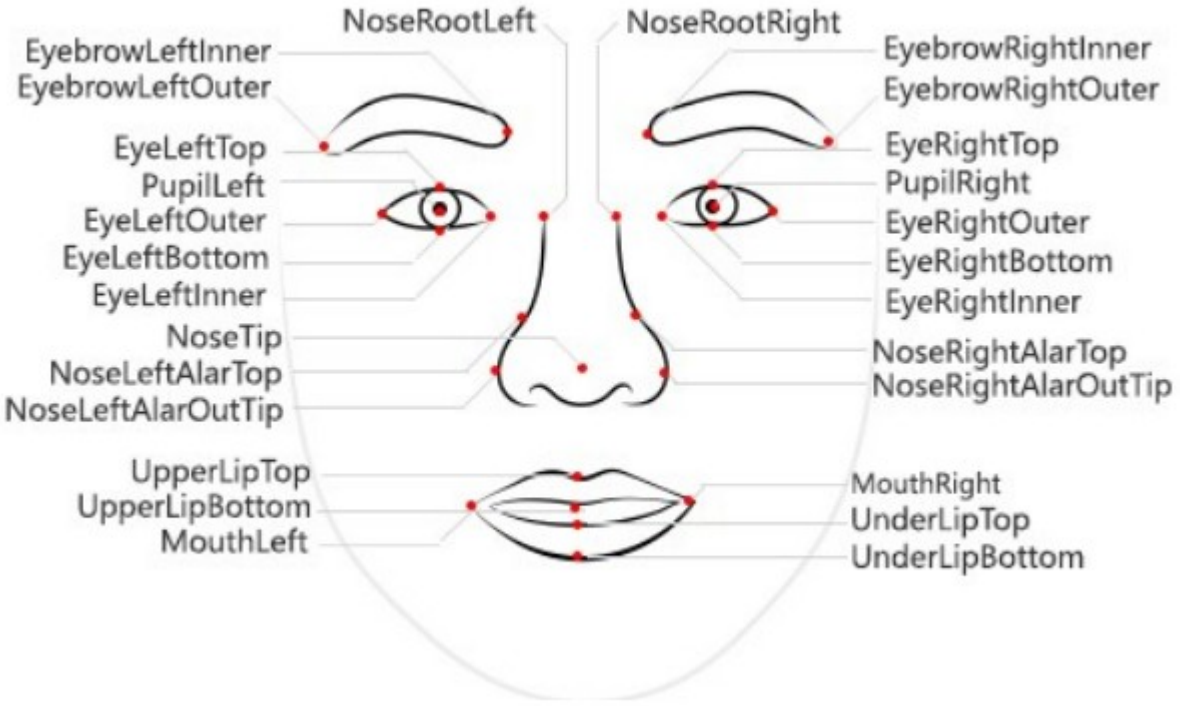
Estructuras de datos

- Cuando utilicemos las funcionalidades de la API de Face tendremos que manipular diferentes estructuras de datos, entre las que se encuentran las siguientes:
  - **DetectedFace**: representa una cara detectada en una imagen. Incluye información de la cara (en la petición se puede indicar qué información se desea obtener) y por defecto tiene una vigencia de 24 horas.
  - **PersistedFace**: es similar al anterior, pero no tiene caducidad. Cuando un objeto DetectedFace se añade a algún tipo de colección de caras (FaceList o Person) se convierte en PersistedFace.
  - **FaceList**:lista variada de objetos PersistedFace, que normalmente pertenecen a personas diferentes.
  - **Person**: representa una persona, y está formado por un conjunto de objetos PersistedFace de esa persona, además de su nombre.
  - **PersonGroup**: lista de objetos Person. Un objeto Person siempre debe pertenecer a un PersonGroup.

Información de las caras

- Cuando utilicemos la funcionalidad de detección de caras, para cada una de las caras detectadas (objeto DetectedFace) podremos obtener información descriptiva de la cara.
- Siempre se obtendrá la localización de la cara dentro de la imagen (lo que se denomina el rectángulo de la cara), pero además se puede solicitar la siguiente información:

- Puntos de referencia de la cara (Landmarks): coordenadas de los distintos puntos que definen las partes de la cara.



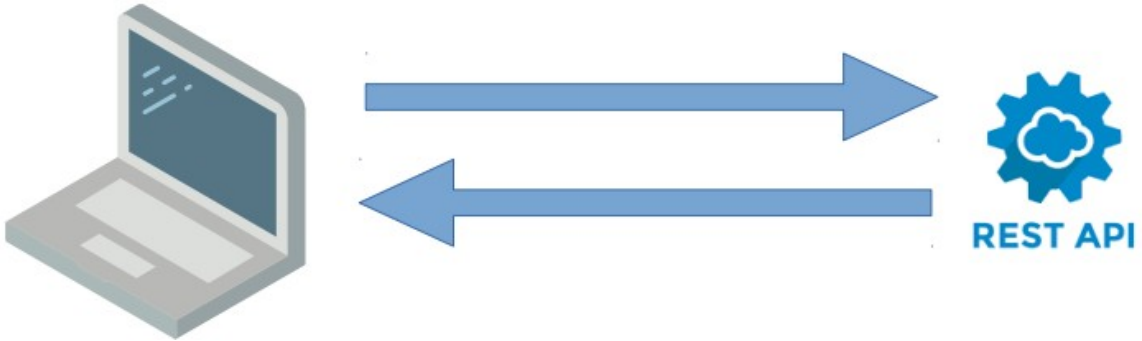
- Atributos de la cara: características descriptivas de la cara, entre las que se encuentran la edad, el sexo, la emoción o la presencia de mascarilla.

Características del servicio

- Estás son algunas de las características y limitaciones más importantes de Face:
  - Las imágenes que se procesan pueden estar en los formatos JPEG, PNG, BMP y GIF (si se trata de una animación se procesará el primer fotograma).
  - El servicio admite procesar imágenes a partir de una URL o recibiendo los datos binarios de la imagen.
  - El tamaño máximo de las imágenes a procesar es de 6 MB.
  - El número de máximo de caras que se pueden detectar en una misma imagen es de 100.
  - Las imágenes procesadas no se almacenan en Azure. Únicamente se guardan los datos descriptivos de las caras detectadas.

Detección de caras (Face - Detect)

- El servicio de detección de caras (Detect) permite obtener las caras presentes en una imagen. La petición al servicio REST será de tipo POST, y se configurará de la siguiente forma:
  - URL: https://{endpoint}/face/v1.0/detect (el endpoint lo obtendremos en el portal de Azure).
  - Parámetros:
    - returnFaceId: indica si queremos obtener el identificador asociado a cada cara detectada. Este identificador permite utilizar una cara en otros servicios (como en el de validación).
    - returnFaceLandMarks: indica si se quieren obtener los puntos de referencia de la cara.
    - returnFaceAttributes: indica los atributos que se quieren obtener de la cara (separados por comas).



- Cabeceras:
  - Content-Type: application/json si vamos a indicar la URL de la imagen, y application/octet-stream si vamos a proporcionar la imagen en binario.
  - Ocp-Apim-Subscription-Key: clave para acceder a la API, que obtendremos del portal de Azure.
- Cuerpo:
  - La URL de la imagen en formato JSON, por ejemplo:

```
{"url":"http://example.com/1.jpg"}
```

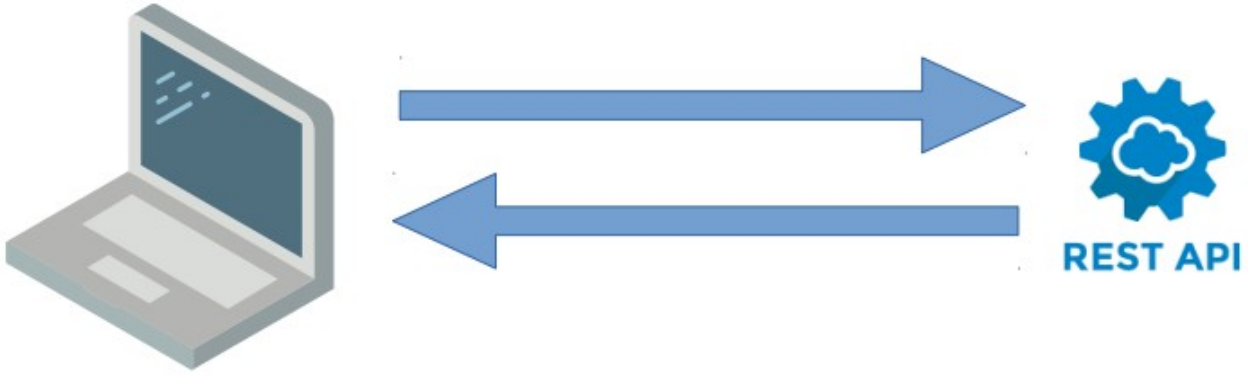
  - O bien los datos binarios de la imagen.

- La respuesta se recibirá en formato JSON, y contendrá una lista de objetos DetectedFace.
- Para cada uno de ellos se incluirá el rectángulo de la cara (faceRectangle) y la información que se haya solicitado en los parámetros de la petición.

- Ejemplo de respuesta para una imagen en la que se detectan dos caras, y se ha solicitado el identificador de la cara y los atributos edad y género.


```
[{
 "faceId": "d0c61551-994f-47b3-a766-1403eb4cc54c",
 "faceRectangle": {
          "top": 50,
          "left": 130,
           "width": 64,
           "height": 64
 },
 "faceAttributes": {
           "gender": "male",
           "age": 26.0
 }
}, {
 "faceId": "d4b39176-8be4-486f-968f-7646528b4cda",
 "faceRectangle": {
           "top": 56,
           "left": 16,
           "width": 57,
           "height": 57
 },
 "faceAttributes": {
           "gender": "female",
           "age": 31.0
 }
}]
```

Parámetros	*returnFaceId, returnFaceLandMarks, returnFaceAttributes*
Cabeceras	*Content-Type, Ocp-Apim-Subscription-Key*
Cuerpo	La imagen a analizar (una URL o los datos binarios)



faceId	Identificador de la cara
faceRectangle	Coordenadas del rectángulo de la cara
faceLandmarks	Puntos de referencia de la cara
faceAttributes	Atributos solicitados de la cara

Verificación de caras (Face - Verify)

- El servicio de verificación de caras (Verify) permite establecer si dos caras detectadas previamente pertenecen a la misma persona. La petición al servicio REST será de tipo POST, y se configurará de la siguiente forma:
  - URL: https://{endpoint}/face/v1.0/verify (el endpoint lo obtendremos en el portal de Azure).
  - Cabeceras:
    - Content-Type: application/json
    - Ocp-Apim-Subscription-Key: clave para acceder a la API, que obtendremos del portal de Azure.

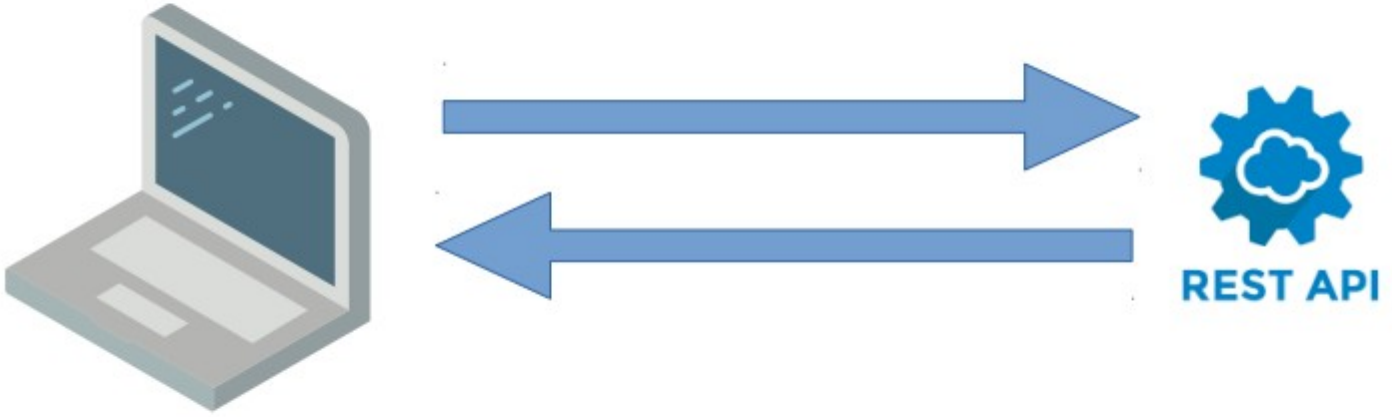
- Cuerpo: los faceId de las dos caras que vamos a comparar.

```
{"faceId1": "c5c24a82-6845-4031-9d5d-978df9175426",
 "faceId2": "815df99c-598f-4926-930a-a734b3fd651c"}
```

- La respuesta se recibirá en formato JSON, y contendrá el campo booleano isIdentical, que indicará si las dos caras pertenecen a la misma persona, y el campo confidence con el nivel de confianza de la verificación (entre 0 y 1). isIdentical será verdadero siempre que la confianza sea mayor o igual a 0,5.

```
{"isIdentical": true, "confidence": 0.9}
```

Parámetros	
Cabeceras	*Content-Type, Ocp-Apim-Subscription-Key*
Cuerpo	*faceId1* y *faceId2*



isIdentical	Booleano que indica si se trata de la misma persona
confidence	Nivel de confianza de la verificación (de 0 a 1)

Gestión de grupos de personas

- Algunas de las funcionalidades del reconocimiento facial (como la identificación, que se trata en el apartado siguiente) se basan en el concepto de personas y grupos de personas.
- En Face, una persona está formada básicamente por un identificador y un conjunto de imágenes de esa persona.
- Un grupo de personas no es más que un conjunto de personas con un identificador.

- Para preparar un grupo de personas deberemos seguir los siguientes pasos:

1. Crear el grupo de personas (objeto de tipo PersonGroup).
2. Crear personas dentro del grupo (objeto Person).
3. Asociar imágenes (que incluirán la cara de la persona) a cada una de las personas.
4. Una vez que se han asociado las imágenes, se debe entrenar el grupo para que pueda ser utilizado (por ejemplo, en una identificación).

- Toda la gestión de grupos y personas se realiza también a través de la API de Face.

Identificación de caras (Face - Identify)

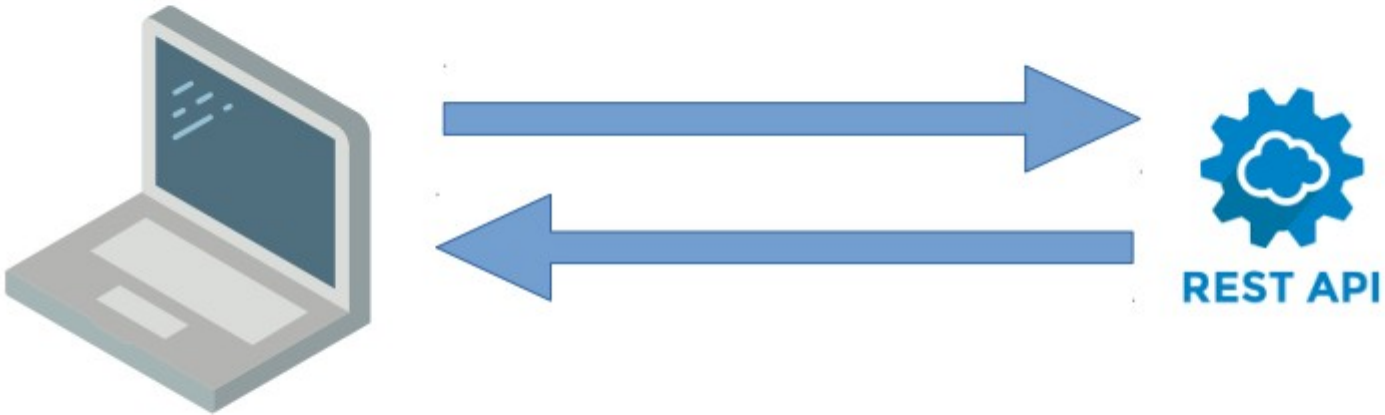
- El servicio de identificación de caras (Identify) permite establecer si una cara detectada previamente pertenece a una persona existente en un grupo de personas. La petición al servicio REST será de tipo POST, y se configurará de la siguiente forma:
  - URL: https://{endpoint}/face/v1.0/identify (el endpoint lo obtendremos en el portal de Azure).
  - Cabeceras:
    - Content-Type: application/json
    - Ocp-Apim-Subscription-Key: clave para acceder a la API, que obtendremos del portal de Azure.
  - Cuerpo:
    - La lista de identificadores de caras que se quieren identificar.
    - El identificador del grupo de personas en el que se va a buscar.

```
{"PersonGroupId": "sample_group",
 "faceIds":["c5c24a82-6845-4031-9d5d-978df9175426",
 "65d083d4-9447-47d1-af30-b626144bf0fb"]}
```

- La respuesta se recibirá en formato JSON, y contendrá, para cada una de las caras incluidas en la petición, el faceId de la cara, y una lista de personas candidatas encontradas en el grupo de personas.
- Para cada una de esas personas, se incluirá su identificador (personId) y el nivel de confianza de la identificación (entre 0 y 1). Las personas candidatas estarán ordenadas por confianza de mayor a menor.


```
[{
 "faceId": "c5c24a82-6845-4031-9d5d-978df9175426",  "candidates":[{
 "personId": "25985303-c537-4467-b41d-bdb45cd95ca1",  "confidence": 0.92
 }]
 },
 {
 "faceId": "65d083d4-9447-47d1-af30-b626144bf0fb",  "candidates":[{
 "personId": "2ae4935b-9659-44c3-977f-61fac20d0538",  "confidence": 0.89
 }]
 }]
```

Parámetros	
Cabeceras	*Content-Type, Ocp-Apim-Subscription-Key*
Cuerpo	*faceIds, personGroupId*



faceId	Identificador de la cara
candidates	Personas candidatas para esa cara
personId	Identificador de la persona
confidence	Confianza de la identificación (entre 0 y 1)

Recursos

- Página principal de Face

https://azure.microsoft.com/es-es/services/cognitive-services/face

- Face en Microsoft Docs

https://docs.microsoft.com/es-es/azure/cognitive-services/face

- Referencia de la API Face

https://docs.microsoft.com/es-es/azure/cognitive-services/face/apireference

- Ejemplos en GitHub

https://github.com/Azure-Samples/cognitive-services-quickstart-code

Creación del recurso Face en el portal de Azure

- Para poder utilizar la API de reconocimiento facial deberemos crear un recurso de tipo Face en el portal de Azure.
- En la ventana de creación del recurso se solicitará la siguiente información:
  - **Suscripción**: deberemos escoger nuestra suscripción de Azure para estudiantes.
  - **Grupo de recursos**: es un contenedor que permite agrupar recursos de Azure relacionados entre sí. Se puede crear un único grupo de recursos para todos los recursos que iremos creando en el curso, o uno para cada uno de los temas.
-

- **Región**: establece en cuál de las regiones de Azure se creará el recurso. Podemos dejar la región por defecto (normalmente Este de EE. UU.) aunque en alguna ocasión es posible que necesitemos seleccionar otra región (algunos servicios no están disponibles en todas las regiones para la suscripción de Azure para estudiantes).
- **Nombre**: el nombre que le queremos dar al recurso. Este nombre también se utilizará para formar la URL del punto de conexión a la API.
- **Plan de tarifa**: determina el nivel de servicio y la facturación asociada. En nuestro caso, seleccionaremos el plan gratuito (F0).

- Una vez creado el recurso, nos dirigiremos a la sección Claves y punto de conexión del recurso para obtener la URL a la que deberemos hacer las peticiones (extremo) y la clave de autenticación de la API (se ofrecen dos claves pero solo es necesario utilizar una de ellas). Estos dos valores (la URL y la clave) son los que necesitaremos para utilizar el servicio desde nuestro programa.