

# Laboratorio: Azure Cognitive Search

## Objetivo

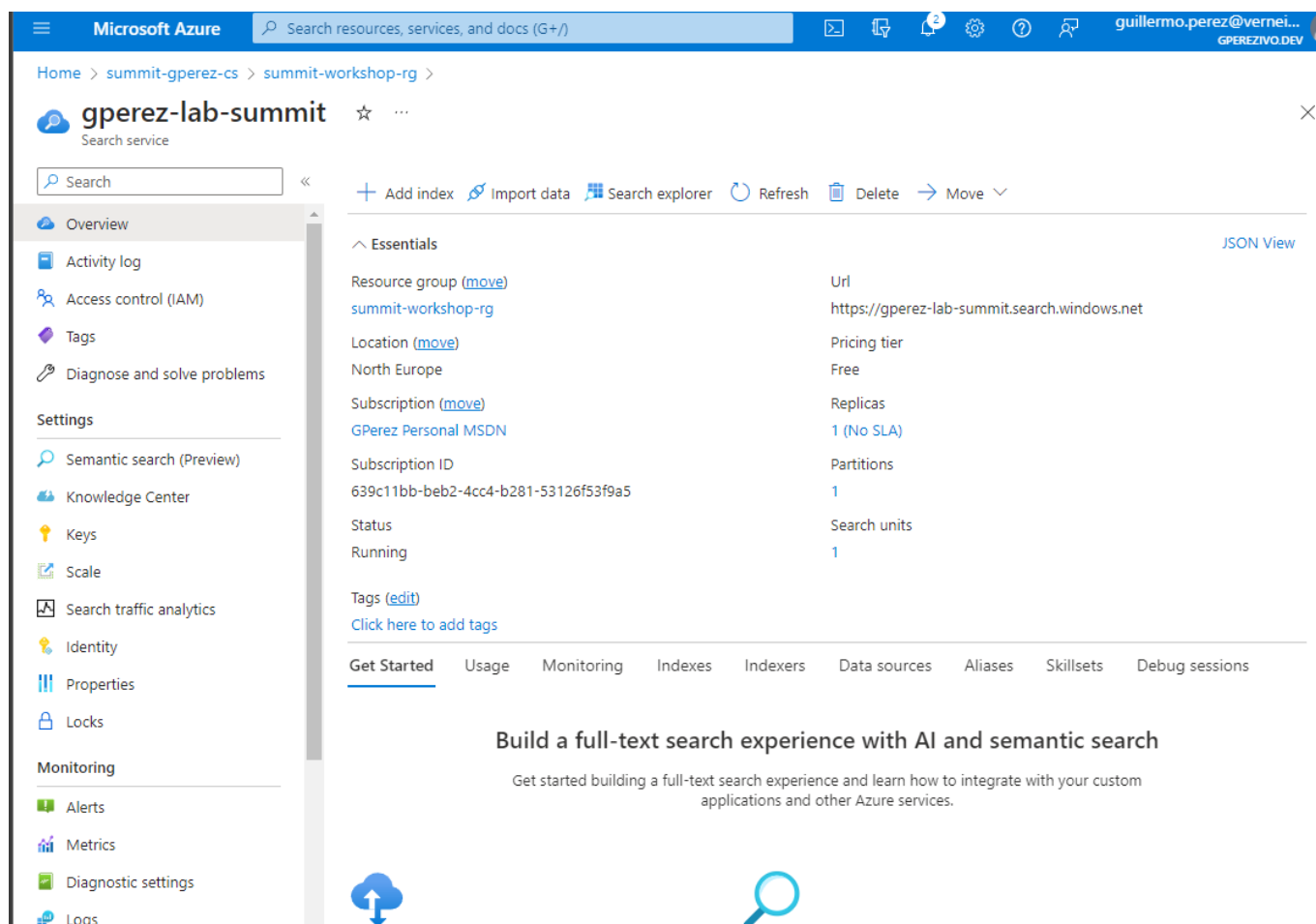
Vamos a crear un Azure Cognitive Search para hacer un buscador de imágenes que tenemos en un blob storage.

## Crear recursos

- Desplegar un grupo de recursos (o tu grupo de recursos de la práctica anterior)

### Crear un Azure Cognitive Search

- En el buscador de recursos, buscar **Azure Cognitive Search**
- Seleccionar **Create** o **Crear**
- Seleccionar el grupo de recursos creado
- Seleccionar el plan de **tarifa free**



The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes the 'Microsoft Azure' logo, a search bar, and user information for 'guillermo.perez@vernei...'. The main content area displays the 'gperez-lab-summit' search service overview. On the left, a sidebar lists various service settings and monitoring options. The main panel shows essential details about the search service, including its resource group, location, subscription, and status. A 'JSON View' link is available for the service details. Below the overview, there are tabs for 'Get Started', 'Usage', 'Monitoring', 'Indexes', 'Indexers', 'Data sources', 'Aliases', 'Skillsets', and 'Debug sessions'. A prominent message encourages building a full-text search experience with AI and semantic search.

Essentials	
Resource group (move)	Url
summit-workshop-rg	https://gperez-lab-summit.search.windows.net
Location (move)	Pricing tier
North Europe	Free
Subscription (move)	Replicas
GPerez Personal MSDN	1 (No SLA)
Subscription ID	Partitions
639c11bb-beb2-4cc4-b281-53126f53f9a5	1
Status	Search units
Running	1
Tags (edit)	
Click here to add tags	

Así se verá el recurso desplegado

### Crear un Cognitive Service

- En el buscador de recursos, buscar **Cognitive Services**

- Seleccionar la opción **Cognitive Services multi-service account** o **Cuenta multi-servicio de Cognitive Services**
  - Si no lo encontramos: [Crear un Cognitive Service](#)
- Seleccionar el grupo de recursos creado
- Seleccionar la ubicación **North Europe** o **Europa del Norte**
- Seleccionar el plan de **tarifa free** (Solo se puede tener uno por suscripción, si ya tenemos uno no hace falta crear otro)
- Seleccionar **Review + create** o **Revisar y crear**


Crear una cuenta de almacenamiento.

- Si has hecho la práctica de Azure ML puedes usar la cuenta de almacenamiento que creaste en esa práctica creando un contenedor nuevo
- Si no, crea una cuenta de almacenamiento nueva y crea un contenedor nuevo
- Establecer el contenedor como público (este paso es opcional)
- Subir las imágenes de la carpeta **images** a tu contenedor

## Crear un datasource

- Accediendo al recurso de Azure Cognitive Search, en el menú lateral, seleccionar **Overview** o **Visión general**.
- Seleccionar la pestaña **Data sources** o **Orígenes de datos**.
- Seleccionar **Add data source** o **Agregar origen de datos**.

Home > [summit-gperez-cs](#) > [summit-workshop-rg](#) >

 **gperez-lab-summit** ☆ ...  
Search service

Search

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Semantic search (Preview)

Knowledge Center

Keys

Scale

Search traffic analytics

+ Add index Import data Search explorer Refresh Delete

Essentials

Resource group ([move](#)) : summit-workshop-rg

Location ([move](#)) : North Europe

Subscription ([move](#)) : GPerez Personal MSDN

Subscription ID : 639c11bb-beb2-4cc4-b281-53126f53f9a5

Status : Running

Tags ([edit](#)) : [Click here to add tags](#)

Get Started Usage Monitoring Indexes Indexers **Data sources**

+ Add data source Refresh Delete

- Rellenar el formulario de creación con el nombre y hacer clic en seleccionar la cadena de conexión.

- Buscar la cuenta de almacenamiento y seleccionar el contenedor creado anteriormente.
- Seleccionar **OK** y **Save** o **Guardar**.

## Crear un skillset

El skillset es la parte que nos permite llamar a componentes externos en el proceso de indexación. En este caso, vamos a pasar el contenido de las imágenes por el servicio de Computer Vision de Azure para extraer las etiquetas de las imágenes, una descripción, marcas y personajes.

- Accediendo al recurso de Azure Cognitive Search, en el menú lateral, seleccionar **Overview** o **Visión general**.
- Seleccionar la pestaña **Skillsets** o **Conjuntos de habilidades**.
- Seleccionar **Add skillset** o **Agregar conjunto de habilidades**.
- Utilizando la plantilla **Image Analysis Skill** que nos proporciona el portal añadirlo dentro de los corchetes de **skills** del JSON.
- En el apartado **visualFeatures** añadir **tags**, **description**, **categories** y **brands**.
- En el apartado **inputs** añadir el nombre del campo que queremos pasar al servicio de Computer Vision, en este caso **image** y el origen de los datos, en este caso **/document/normalized\_images/**.
- En el apartado **outputs** añadir los campos que queremos que se indexen en el buscador, en este caso **tags**, **description**, **categories** y **brands**.
- Introducir el nombre tanto de la skill como del skillset.
- En la pestaña de **Cognitive Services** seleccionar el servicio que hemos creado anteriormente.
- Hacer clic en **Save** o **Guardar**.

### Ejemplo de json:

```
{
  "name": "image-data-extraction",
  "skills": [
    {
      "@odata.type": "#Microsoft.Skills.Vision.ImageAnalysisSkill",
      "name": "image-data",
      "description": "",
      "context": "/document/normalized_images/*",
      "defaultLanguageCode": "en",
      "visualFeatures": [
        "brands",
        "categories",
        "description",
        "tags"
      ],
      "details": [],
      "inputs": [
        {
          "name": "image",
          "source": "/document/normalized_images/*"
        }
      ],
      "outputs": [
        {
```

```
        "name": "brands"
      },
      {
        "name": "categories"
      },
      {
        "name": "description"
      },
      {
        "name": "tags"
      }
    ]
  },
],
}
```

## Crear un índice

El índice es la parte que nos permite definir los campos que se van a indexar y que se van a poder buscar. Es decir, el contenido que se va a poder buscar.

- Accediendo al recurso de Azure Cognitive Search, en el menú lateral, seleccionar **Overview** o **Visión general**.
- Seleccionar la pestaña **Indexes** o **Índices**.
- Seleccionar **Add index** o **Agregar índice**.
- Introducir el nombre del índice.
- En el apartado **fields** añadir la siguiente columna:
  - **metadata\_storage\_path**:
    - type: **Edm.String**
    - key: **false**
    - retrievable: **true**
- Guardamos el índice.
- Una vez creado y guardado, lo seleccionamos de la lista de índices y entramos en el menú de **Edit JSON** o **Editar JSON**.
- En la lista de fiels añadimos el siguiente fragmento

```
{
  "name": "brands",
  "type": "Collection(Edm.ComplexType)",
  "fields": [
    {
      "name": "name",
      "type": "Edm.String",
      "searchable": true,
      "filterable": true,
      "retrievable": true,
      "sortable": false,
      "facettable": true,

```

```
    "key": false,
    "indexAnalyzer": null,
    "searchAnalyzer": null,
    "analyzer": null,
    "normalizer": null,
    "synonymMaps": []
  },
  {
    "name": "confidence",
    "type": "Edm.Double",
    "searchable": false,
    "filterable": false,
    "retrievable": true,
    "sortable": false,
    "facettable": false,
    "key": false,
    "indexAnalyzer": null,
    "searchAnalyzer": null,
    "analyzer": null,
    "normalizer": null,
    "synonymMaps": []
  },
  {
    "name": "rectangle",
    "type": "Edm.ComplexType",
    "fields": [
      {
        "name": "x",
        "type": "Edm.Int32",
        "searchable": false,
        "filterable": false,
        "retrievable": true,
        "sortable": false,
        "facettable": false,
        "key": false,
        "indexAnalyzer": null,
        "searchAnalyzer": null,
        "analyzer": null,
        "normalizer": null,
        "synonymMaps": []
      },
      {
        "name": "y",
        "type": "Edm.Int32",
        "searchable": false,
        "filterable": false,
        "retrievable": true,
        "sortable": false,
        "facettable": false,
        "key": false,
        "indexAnalyzer": null,
        "searchAnalyzer": null,
        "analyzer": null,
        "normalizer": null,
```

```

        "synonymMaps": []
    },
    {
        "name": "w",
        "type": "Edm.Int32",
        "searchable": false,
        "filterable": false,
        "retrievable": true,
        "sortable": false,
        "facettable": false,
        "key": false,
        "indexAnalyzer": null,
        "searchAnalyzer": null,
        "analyzer": null,
        "normalizer": null,
        "synonymMaps": []
    },
    {
        "name": "h",
        "type": "Edm.Int32",
        "searchable": false,
        "filterable": false,
        "retrievable": true,
        "sortable": false,
        "facettable": false,
        "key": false,
        "indexAnalyzer": null,
        "searchAnalyzer": null,
        "analyzer": null,
        "normalizer": null,
        "synonymMaps": []
    }
]
}
],
{
    "name": "categories",
    "type": "Collection(Edm.ComplexType)",
    "fields": [
        {
            "name": "name",
            "type": "Edm.String",
            "searchable": true,
            "filterable": true,
            "retrievable": true,
            "sortable": false,
            "facettable": true,
            "key": false,
            "indexAnalyzer": null,
            "searchAnalyzer": null,
            "analyzer": null,
            "normalizer": null,
            "synonymMaps": []
        }
    ]
}

```

```
    },
    {
      "name": "score",
      "type": "Edm.Double",
      "searchable": false,
      "filterable": false,
      "retrievable": true,
      "sortable": false,
      "facettable": false,
      "key": false,
      "indexAnalyzer": null,
      "searchAnalyzer": null,
      "analyzer": null,
      "normalizer": null,
      "synonymMaps": []
    },
    {
      "name": "detail",
      "type": "Edm.ComplexType",
      "fields": [
        {
          "name": "celebrities",
          "type": "Collection(Edm.ComplexType)",
          "fields": [
            {
              "name": "name",
              "type": "Edm.String",
              "searchable": true,
              "filterable": true,
              "retrievable": true,
              "sortable": false,
              "facettable": true,
              "key": false,
              "indexAnalyzer": null,
              "searchAnalyzer": null,
              "analyzer": null,
              "normalizer": null,
              "synonymMaps": []
            }
          ],
        },
        {
          "name": "faceBoundingBox",
          "type": "Collection(Edm.ComplexType)",
          "fields": [
            {
              "name": "x",
              "type": "Edm.Int32",
              "searchable": false,
              "filterable": false,
              "retrievable": true,
              "sortable": false,
              "facettable": false,
              "key": false,
              "indexAnalyzer": null,
              "searchAnalyzer": null,
            }
          ],
        }
      ]
    }
  ]
}
```

```
        "analyzer": null,
        "normalizer": null,
        "synonymMaps": []
    },
    {
        "name": "y",
        "type": "Edm.Int32",
        "searchable": false,
        "filterable": false,
        "retrievable": true,
        "sortable": false,
        "facettable": false,
        "key": false,
        "indexAnalyzer": null,
        "searchAnalyzer": null,
        "analyzer": null,
        "normalizer": null,
        "synonymMaps": []
    }
]
},
{
    "name": "confidence",
    "type": "Edm.Double",
    "searchable": false,
    "filterable": false,
    "retrievable": true,
    "sortable": false,
    "facettable": false,
    "key": false,
    "indexAnalyzer": null,
    "searchAnalyzer": null,
    "analyzer": null,
    "normalizer": null,
    "synonymMaps": []
}
]
},
{
    "name": "landmarks",
    "type": "Collection(Edm.ComplexType)",
    "fields": [
        {
            "name": "name",
            "type": "Edm.String",
            "searchable": true,
            "filterable": true,
            "retrievable": true,
            "sortable": false,
            "facettable": true,
            "key": false,
            "indexAnalyzer": null,
            "searchAnalyzer": null,
            "analyzer": null,
```



```

        "normalizer": null,
        "synonymMaps": []
    },
    {
        "name": "confidence",
        "type": "Edm.Double",
        "searchable": false,
        "filterable": false,
        "retrievable": true,
        "sortable": false,
        "facettable": false,
        "key": false,
        "indexAnalyzer": null,
        "searchAnalyzer": null,
        "analyzer": null,
        "normalizer": null,
        "synonymMaps": []
    }
]
}
]
},
{
    "name": "description",
    "type": "Collection(Edm.ComplexType)",
    "fields": [
        {
            "name": "tags",
            "type": "Collection(Edm.String)",
            "searchable": true,
            "filterable": false,
            "retrievable": true,
            "sortable": false,
            "facettable": false,
            "key": false,
            "indexAnalyzer": null,
            "searchAnalyzer": null,
            "analyzer": null,
            "normalizer": null,
            "synonymMaps": []
        },
        {
            "name": "captions",
            "type": "Collection(Edm.ComplexType)",
            "fields": [
                {
                    "name": "text",
                    "type": "Edm.String",
                    "searchable": true,
                    "filterable": false,
                    "retrievable": true,
                    "sortable": false,

```

```






        "facettable": false,
        "key": false,
        "indexAnalyzer": null,
        "searchAnalyzer": null,
        "analyzer": null,
        "normalizer": null,
        "synonymMaps": []
    },
    {
        "name": "confidence",
        "type": "Edm.Double",
        "searchable": false,
        "filterable": false,
        "retrievable": true,
        "sortable": false,
        "facettable": false,
        "key": false,
        "indexAnalyzer": null,
        "searchAnalyzer": null,
        "analyzer": null,
        "normalizer": null,
        "synonymMaps": []
    }
]
}
],
},
{
    "name": "tags",
    "type": "Collection(Edm.ComplexType)",
    "fields": [
        {
            "name": "name",
            "type": "Edm.String",
            "searchable": true,
            "filterable": false,
            "retrievable": true,
            "sortable": false,
            "facettable": true,
            "key": false,
            "indexAnalyzer": null,
            "searchAnalyzer": null,
            "analyzer": null,
            "normalizer": null,
            "synonymMaps": []
        },
        {
            "name": "hint",
            "type": "Edm.String",
            "searchable": true,
            "filterable": false,
            "retrievable": true,
            "sortable": false,
            "facettable": false,

```

```
        "key": false,  
        "indexAnalyzer": null,  
        "searchAnalyzer": null,  
        "analyzer": null,  
        "normalizer": null,  
        "synonymMaps": []  
    },  
    {  
        "name": "confidence",  
        "type": "Edm.Double",  
        "searchable": false,  
        "filterable": false,  
        "retrievable": true,  
        "sortable": false,  
        "facettable": false,  
        "key": false,  
        "indexAnalyzer": null,  
        "searchAnalyzer": null,  
        "analyzer": null,  
        "normalizer": null,  
        "synonymMaps": []  
    }  
]  
}
```

Con esto incluiremos los campos que necesitamos para almacenar la información de la imagen que obtenemos del skillset.

Esto podemos hacerlo también mediante el formulario inicial de fields, pero por la complejidad de los tipos, es más sencillo hacerlo mediante el editor de código.

	id	String	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	metadata_storage_path	String	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	brands	ComplexTypeCollection					
	name	String	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Standar... 
	confidence	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	rectangle	ComplexType					
	x	Int32	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	y	Int32	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	w	Int32	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	h	Int32	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	categories	ComplexTypeCollection					
	name	String	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Standar... 
	score	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	detail	ComplexType					
	celebrities	ComplexTypeCollection					
	name	String	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Standar... 
	faceBoundingBox	ComplexTypeCollection					
	x	Int32	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	y	Int32	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	confidence	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	landmarks	ComplexTypeCollection					
	name	String	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Standar... 
	confidence	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	description	ComplexTypeCollection					
	tags	StringCollection	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Standar... 
	captions	ComplexTypeCollection					
	text	String	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Standar... 
	confidence	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
	tags	ComplexTypeCollection					
	name	String	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Standar... 
	hint	String	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Standar... 
	confidence	Double	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		

**Nota:** Si ya tenemos un índice creado, podemos usarlo, pero tendremos que añadir los campos que hemos añadido en el skillset.

## Crear un indexador

El indexador es el proceso que se encarga de leer los datos de la fuente de datos y los indexa en el índice tras aplicar los skills.

- Vamos a la sección de Indexers y pulsamos en **Add Indexer**.
- Seleccionamos el índice que hemos creado en el paso anterior.
- Seleccionamos la fuente de datos que hemos creado con las imágenes.
- Seleccionamos el skillset que hemos creado de análisis de imágenes.

- En la opción de **Image action** seleccionamos **GenerateNormalizedImages** para que se procesen las imágenes y se pasen por el skillset.
- En la pestaña de **Indexer Definition (JSON)** vamos a introducir el siguiente contenido dentro de los corchetes de **outputFieldMappings**:

```
{
  "sourceFieldName": "/document/normalized_images/*/description",
  "targetFieldName": "description"
},
{
  "sourceFieldName": "/document/normalized_images/*/tags/*",
  "targetFieldName": "tags"
},
{
  "sourceFieldName": "/document/normalized_images/*/categories/*",
  "targetFieldName": "categories"
},
{
  "sourceFieldName": "/document/normalized_images/*/brands/*",
  "targetFieldName": "brands"
}
```

Con esto indicamos que los campos que se generan en el skillset se mapeen a los campos del índice.

- Guardamos el indexer y lo ejecutamos.

## Buscar en el índice

- Vamos a la sección de **Search Explorer**.
- Seleccionamos el índice que hemos creado.
- En la caja de texto de consulta introducimos **\*** para que nos devuelva todos los documentos.
- Pulsamos en **Search**.
- En la parte inferior de la pantalla veremos los resultados de la búsqueda.

## Crear una aplicación de búsqueda

- Vamos a la sección de **Indexes** y seleccionamos el índice que hemos creado.
- En la parte superior pulsamos en **Create Demo App**.

**Nota:** Nos pedirá que modifiquemos el CORS para poder acceder de manera externa al recurso, aceptamos y guardamos.

- En **Individual result** asignamos las siguientes columnas del índice:
  - **metadata\_storage\_path** en **Thumbnail**
  - **metadata\_storage\_path** en **Title**
  - **description.captions.text** en **Description**
- En la sección **Sidebar** podemos seleccionar los campos que sean *facetables* de nuestro índice para que aparezcan en la barra lateral de la aplicación con un recuento de los valores que hay en cada uno.

- brands/name
  - categories/name
  - categories/detail/celebrities/name
  - categories/detail/landmarks/name
  - tags/name
- Creamos la aplicación y abrimos el html que se descarga.

