Elasticsearch: Plataforma Virtual

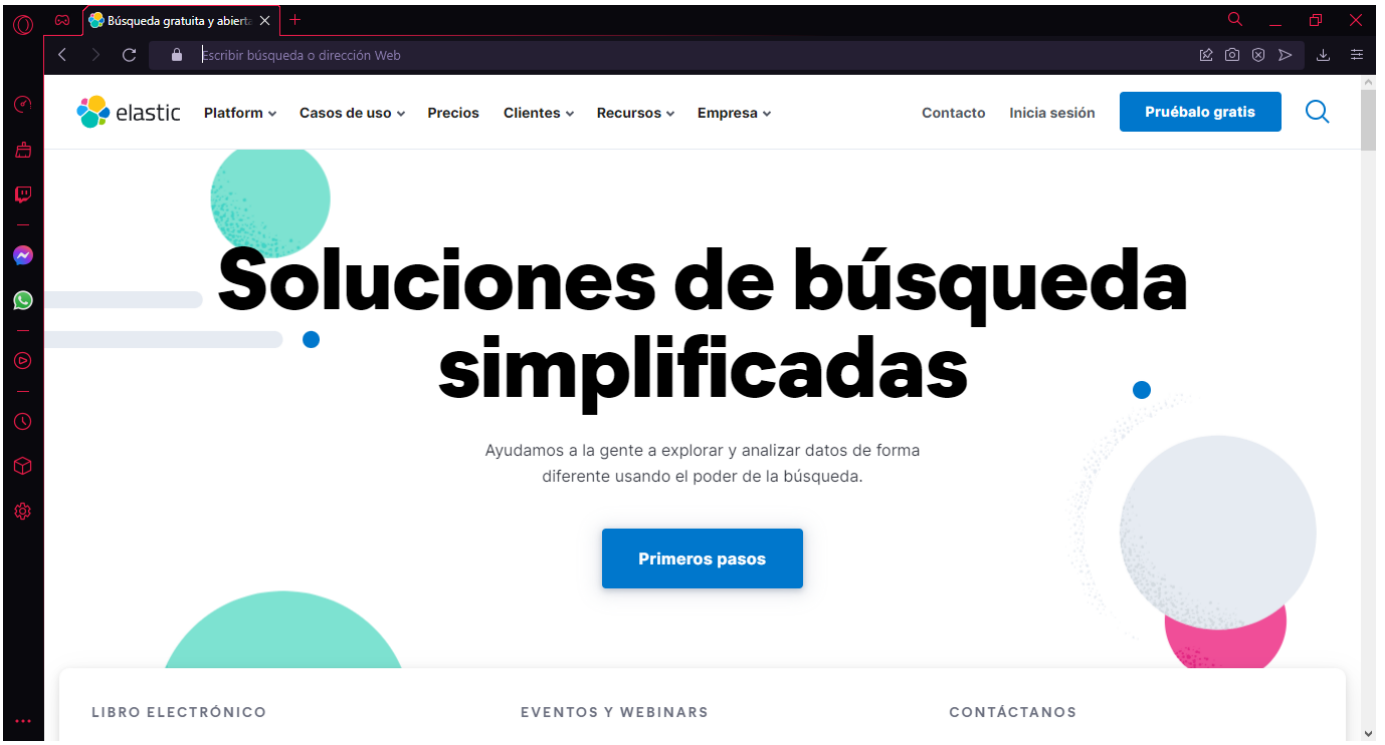
Ing.Luis Felipe Narvaez Gomez. E-mail: luis.narvaez@usantoto.edu.co. Cod: 2312660. Facultad de Ingeniería de Sistemas.

INGRESAR A ELASTIC

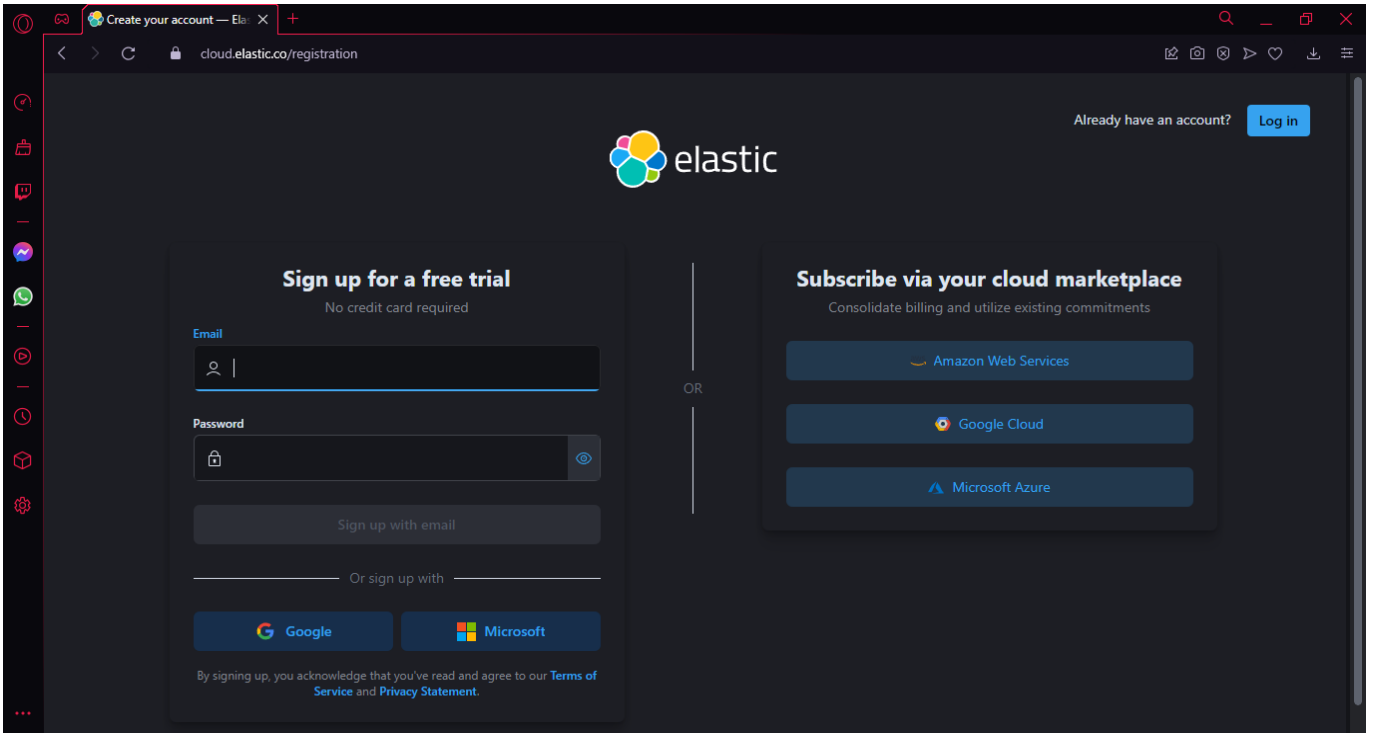
Primero hemos de ingresar al link de la pagina oficial de ElesticSearch, mediante el siguiente enlace:

```
https://www.elastic.co/es/
```

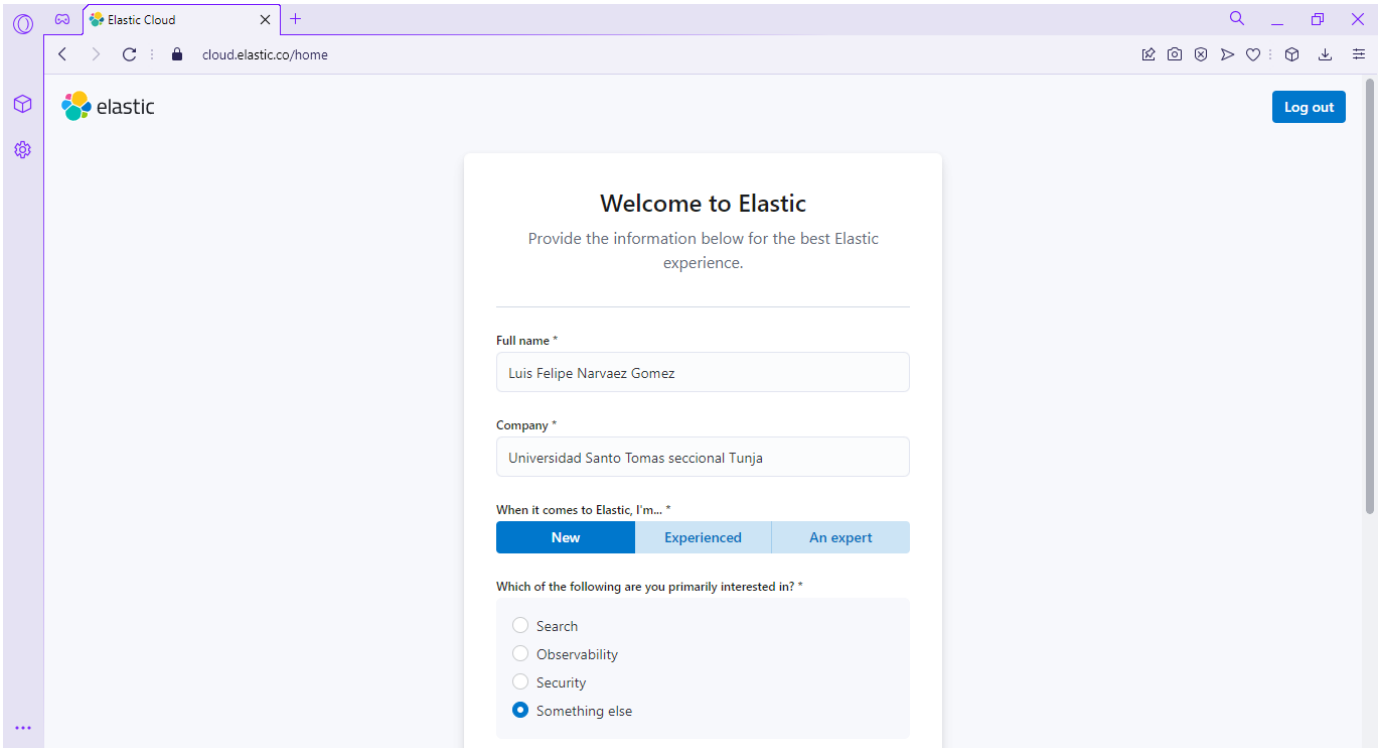
La pagina tiene el siguiente aspecto:



En esta pagina nos dirigiremos al botón de la esquina superior derecha "Pruébalo Gratis" dando clic aquí. Ahí nos encontraremos en una pagina donde se nos pedirá registrarnos para acceder al servicio. Podemos utilizar cuentas de google, microsoft o cualquier otro servicio de correo electrónico como yahoo.

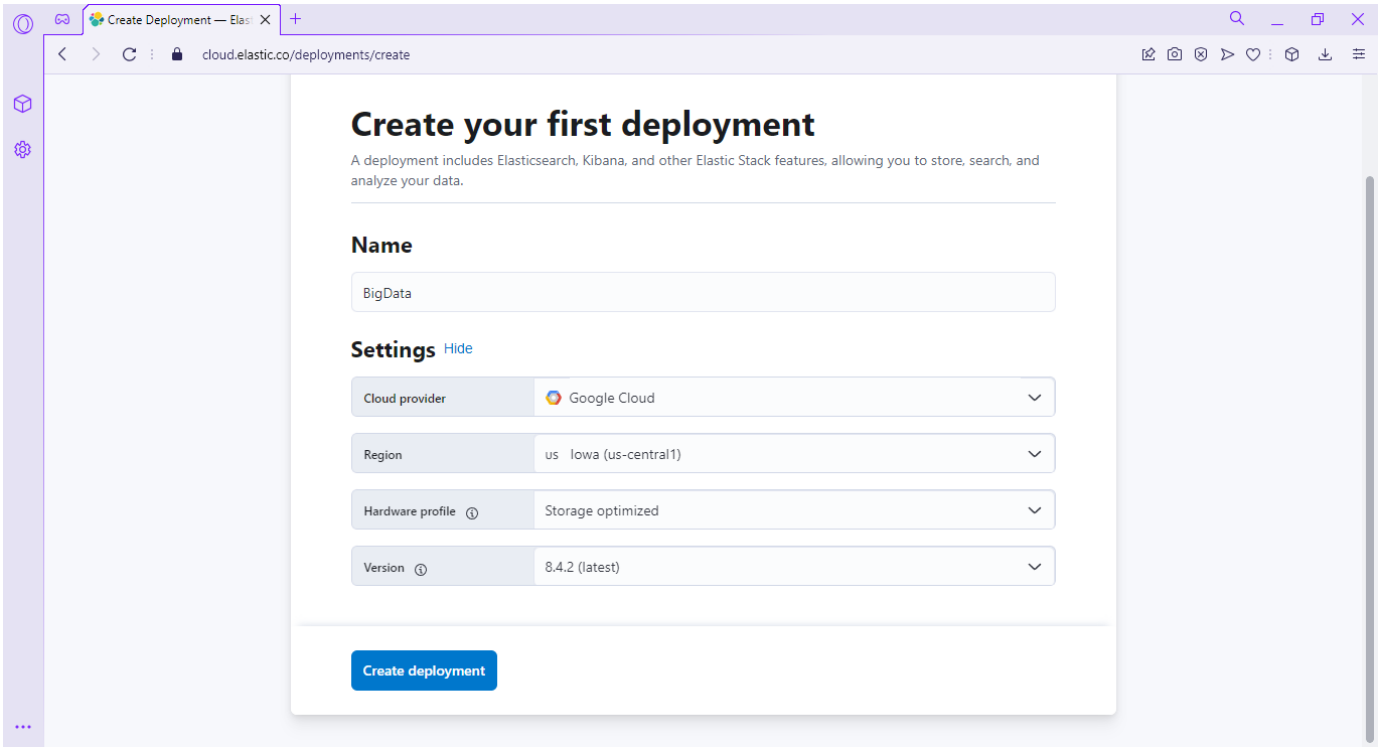


Si somos nuevos en el sistema de elastic se nos pedirá anexar el porque de nuestro registro, así como la compañía o sector al que pertenecemos.

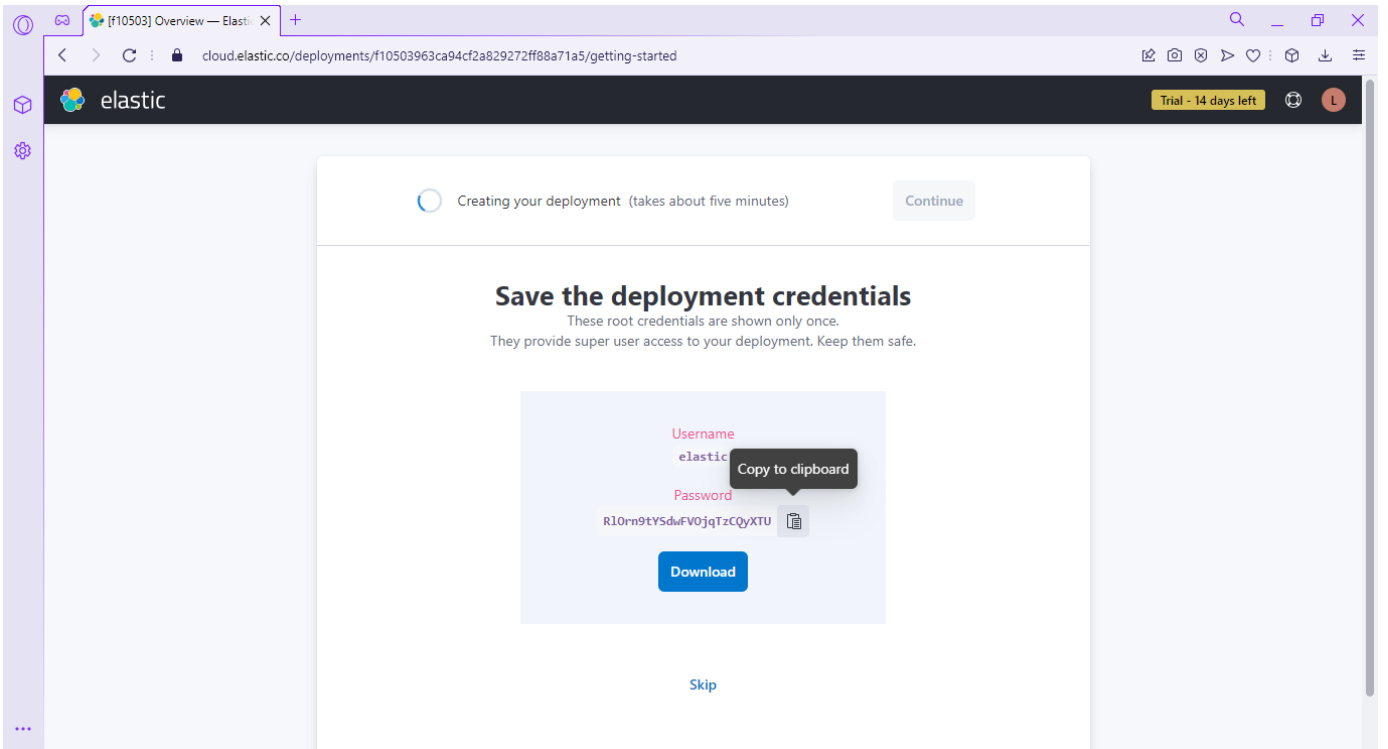


CREAR UN TRABAJO DE DESARROLLO

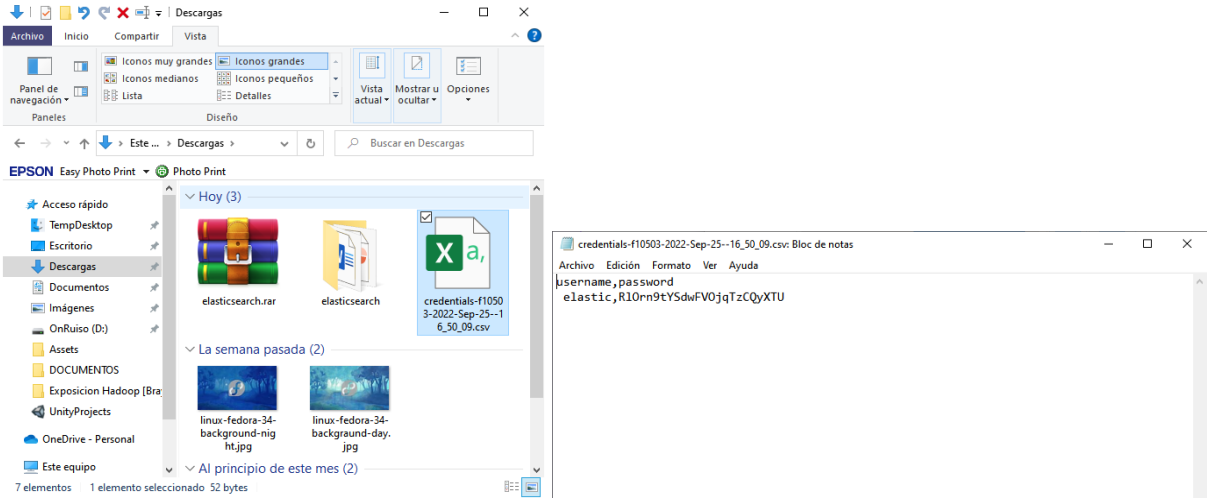
Automáticamente se nos dirigirá a que creemos nuestro primer trabajo desarrollo.



Luego se generaran nuestras credenciales para este Proyecto, las mismas podemos descargarlas en nuestro sistema como un Archivo CSV o bien copiarlas en algún sitio como un archivo de texto plano. Deberemos esperar 5 minutos en lo que se crea el proyecto y nos da tiempo de hacernos con las importantes credenciales de tipo par.



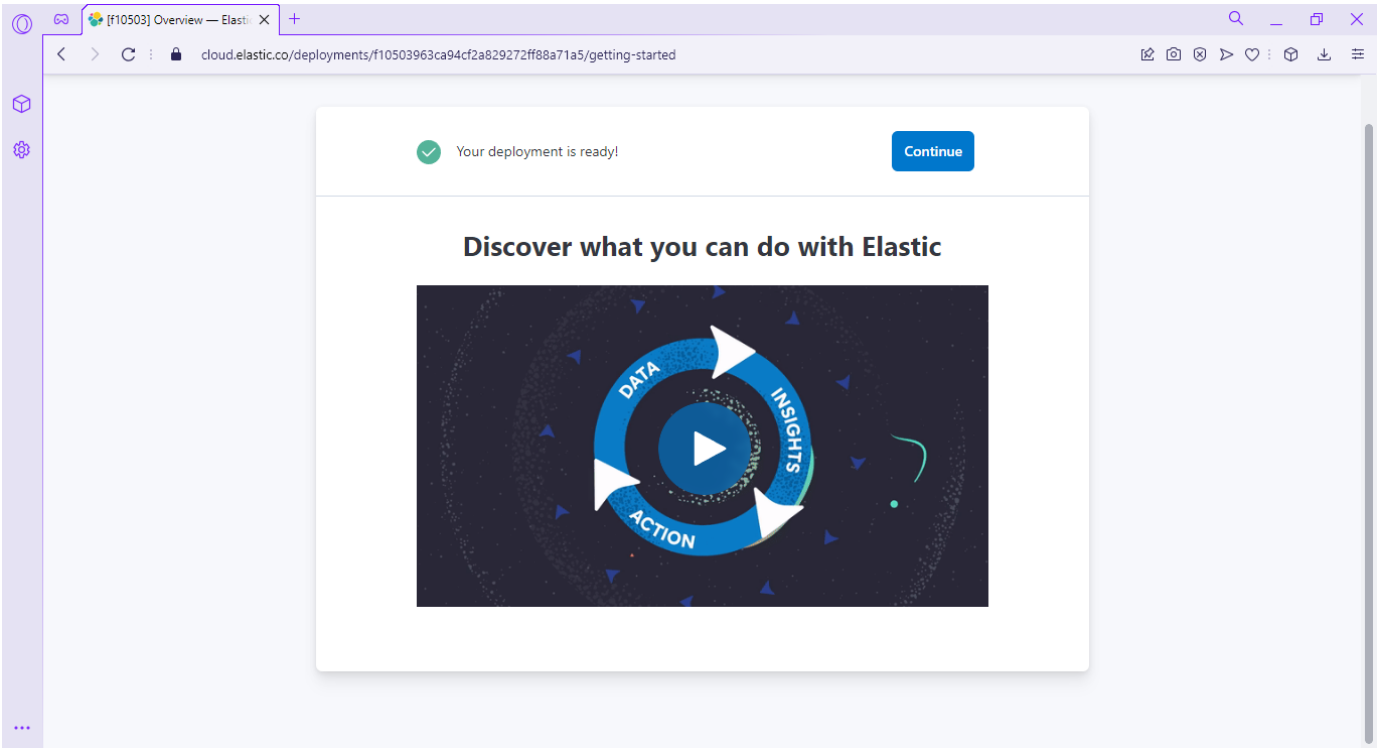
Con el archivo descargado tendríamos lo siguiente:



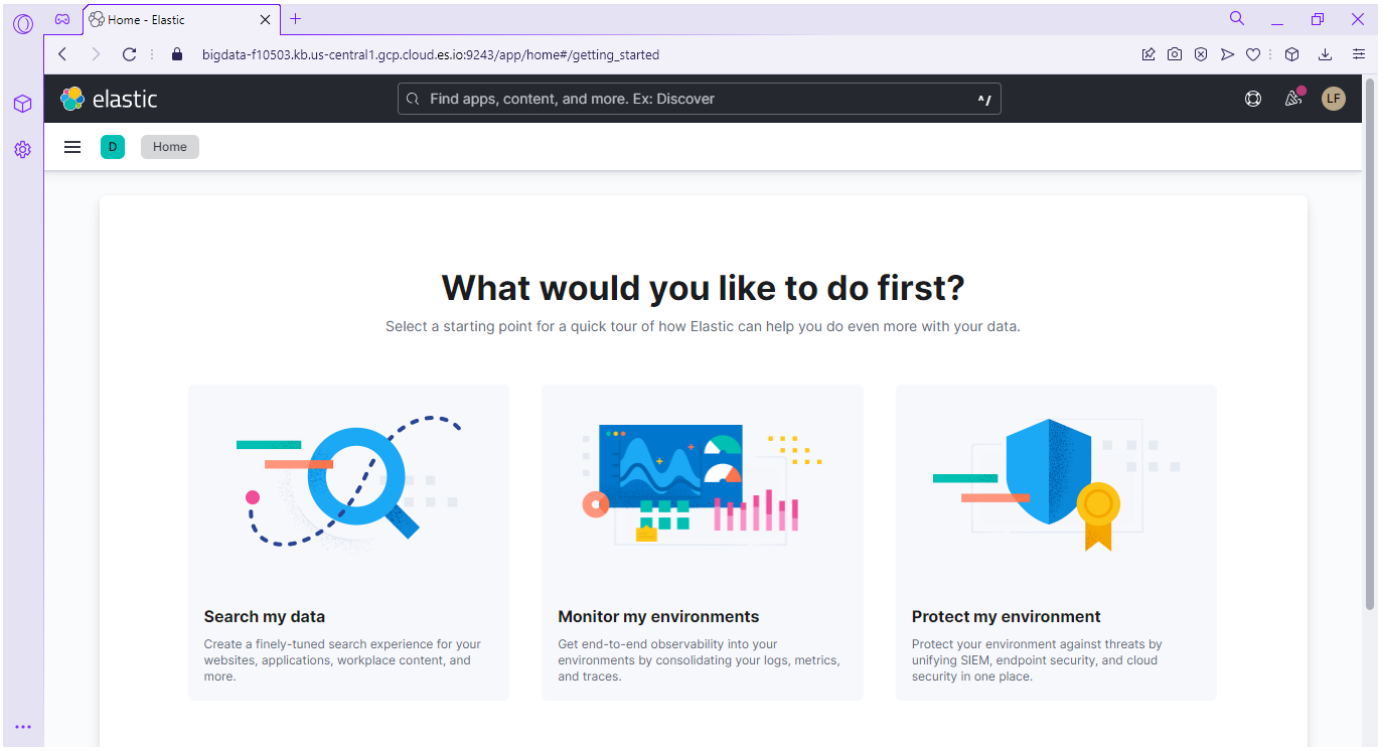
Mientras que como archivo de texto plano en nuestro portapapeles tendríamos algo similar a esto:

```
Username: elastic
Password: RlOrn9tYSdwFVOjqTzCQyXTU
```

Luego de que termine con la creación del proyecto tendríamos lo siguiente:

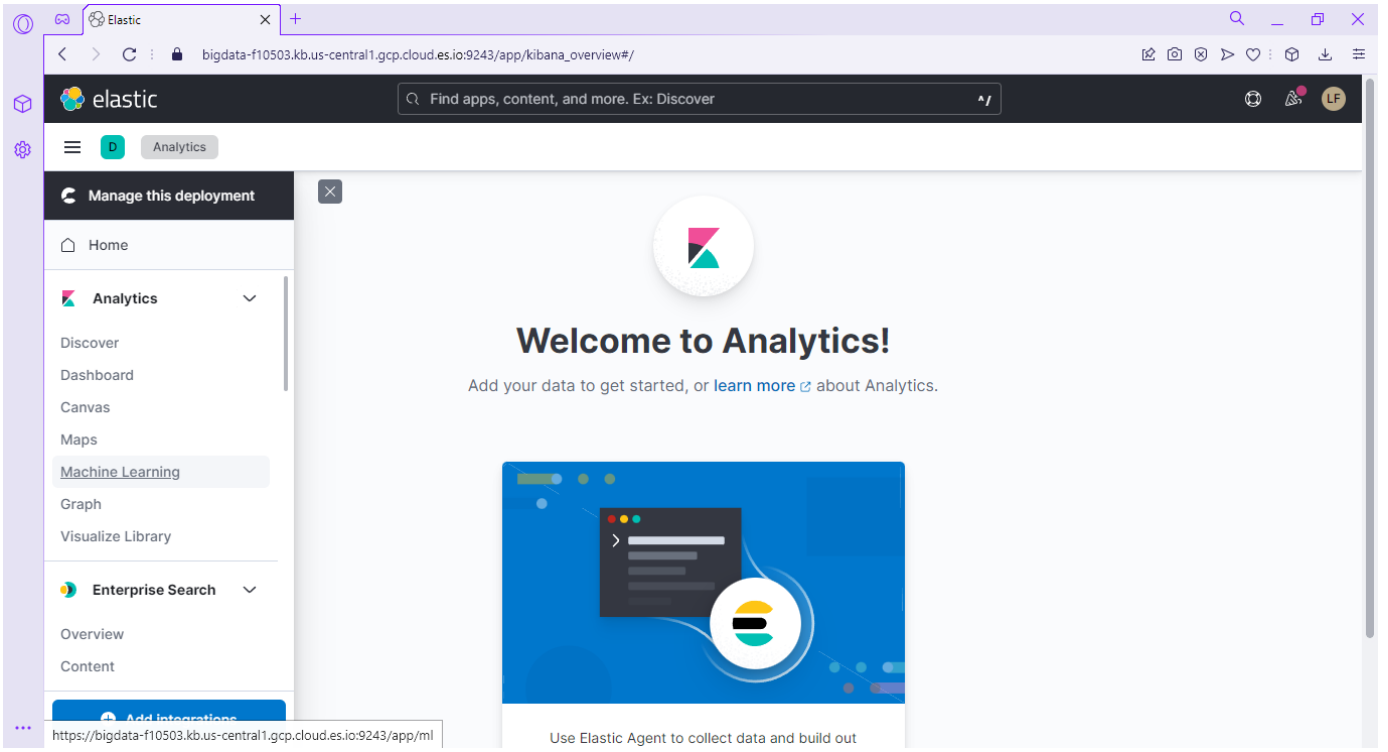


Solo debemos dar clic en el botón de "continue" y tendremos una dashboard de trabajo similar a la siguiente:

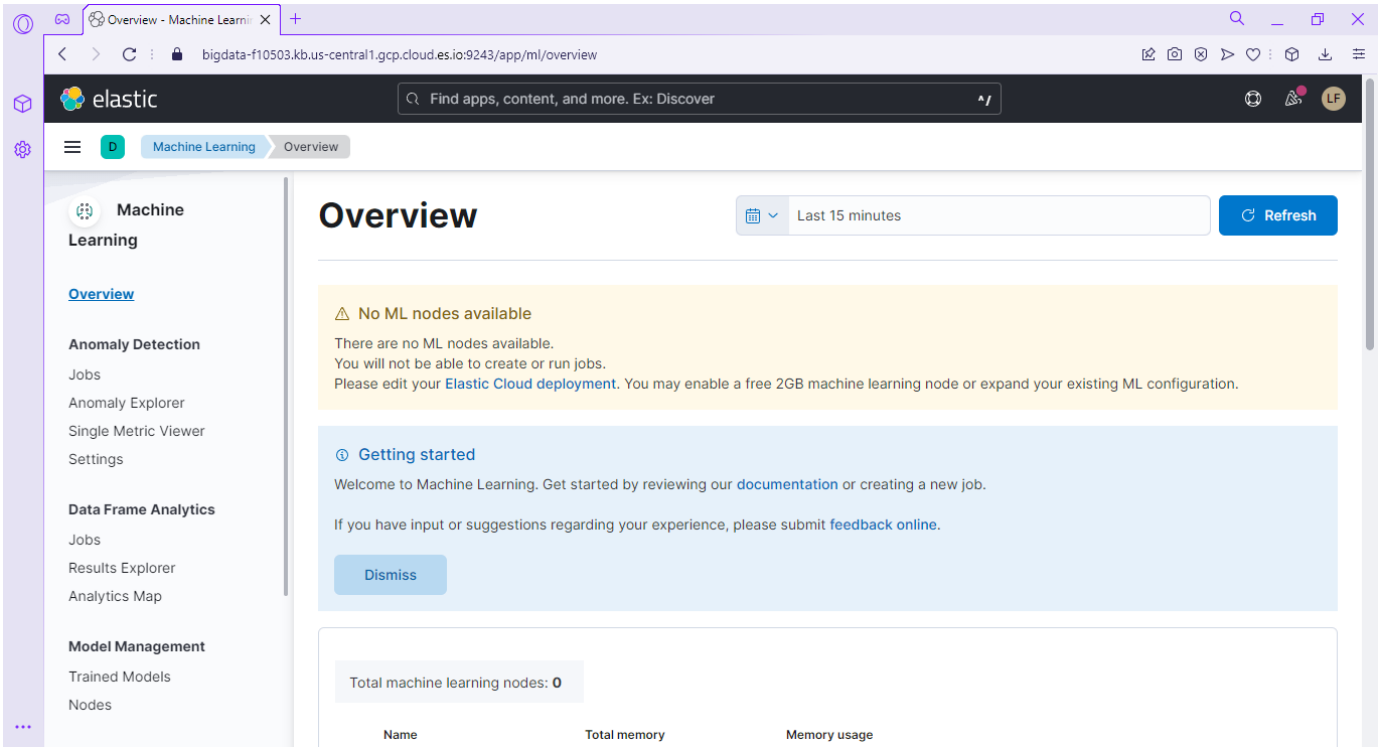


AGREGAR UN DATAFRAME

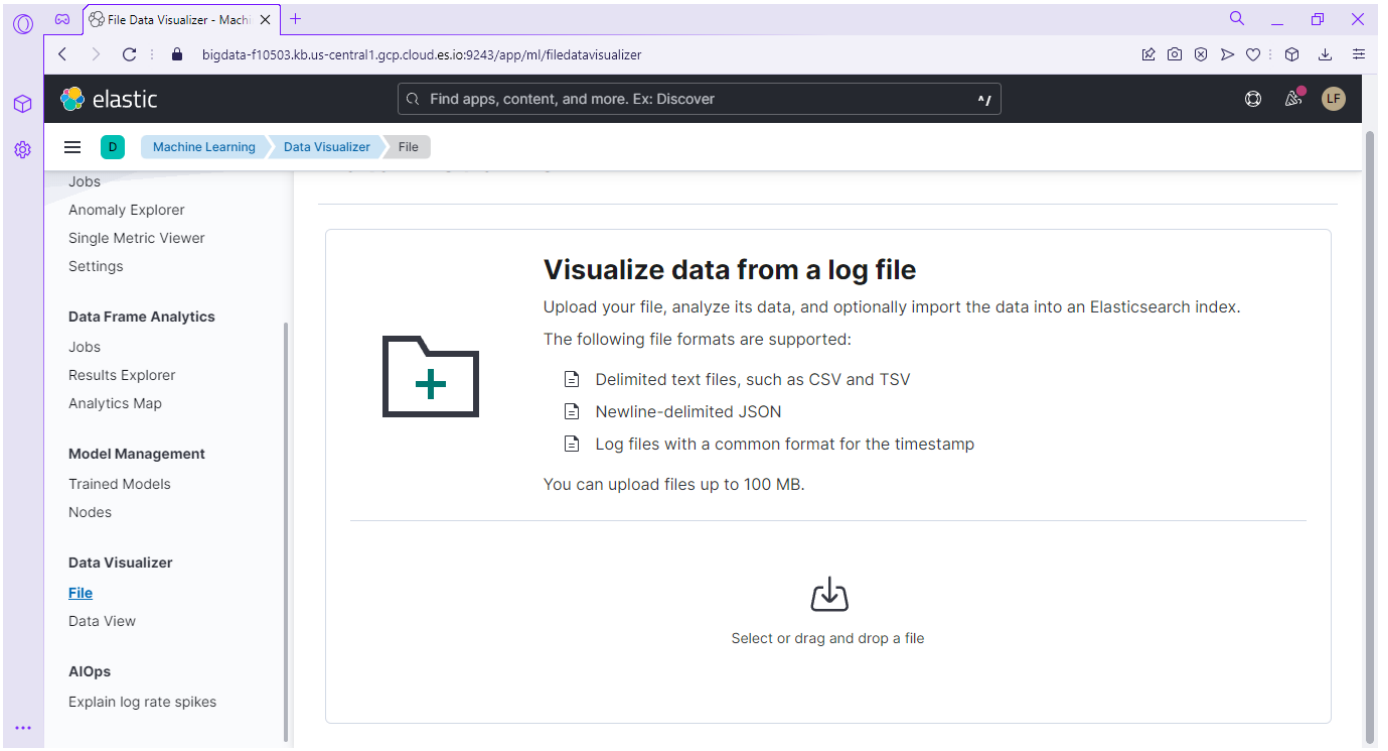
Ahora para poder agregar un data frame, navegaremos por la dashboard a la sección del Menú de Hamburguesa en la parte superior izquierda seleccionándolo, allí encontraremos la función de Analíticas y dentro de ella la opción de Machine Learning.



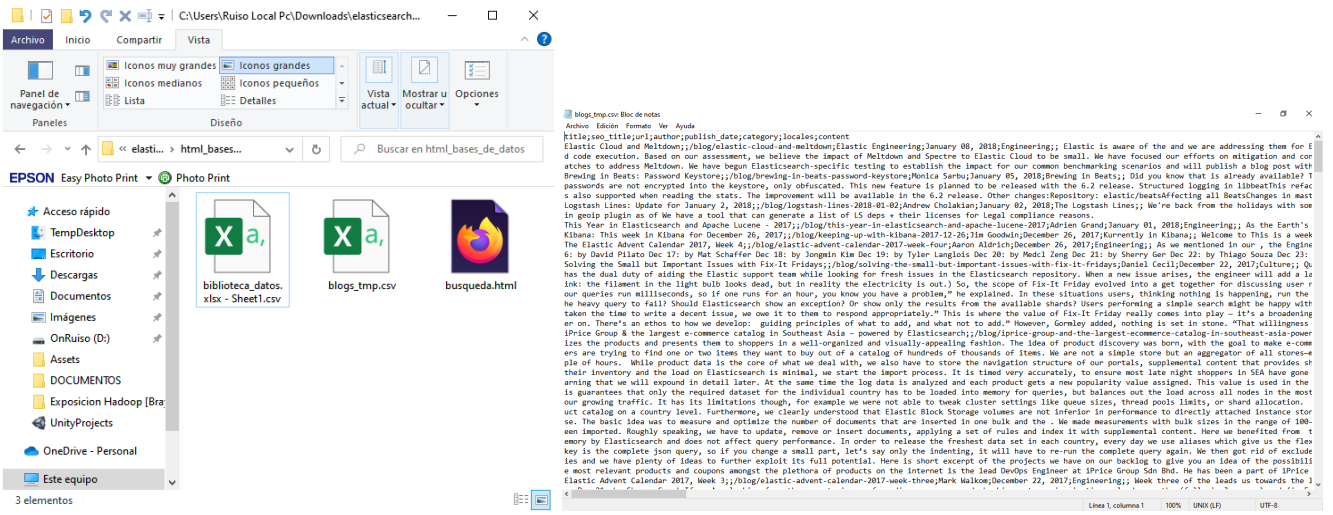
Al seleccionarla encontraremos lo siguiente:



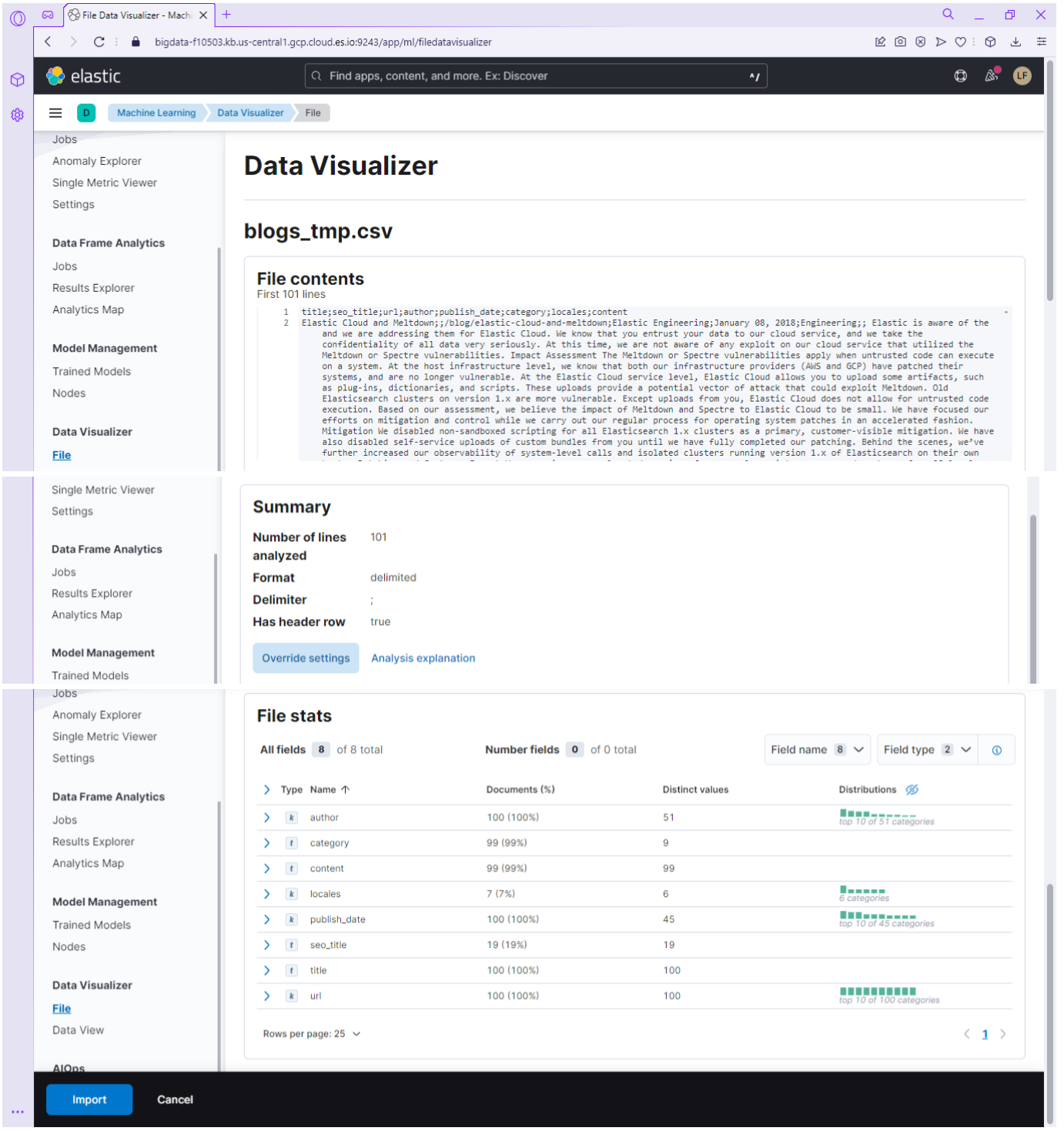
En la anterior pantalla buscaremos en el menú de la izquierda la opción de "Data Visualizer" y en ella seleccionamos "File".



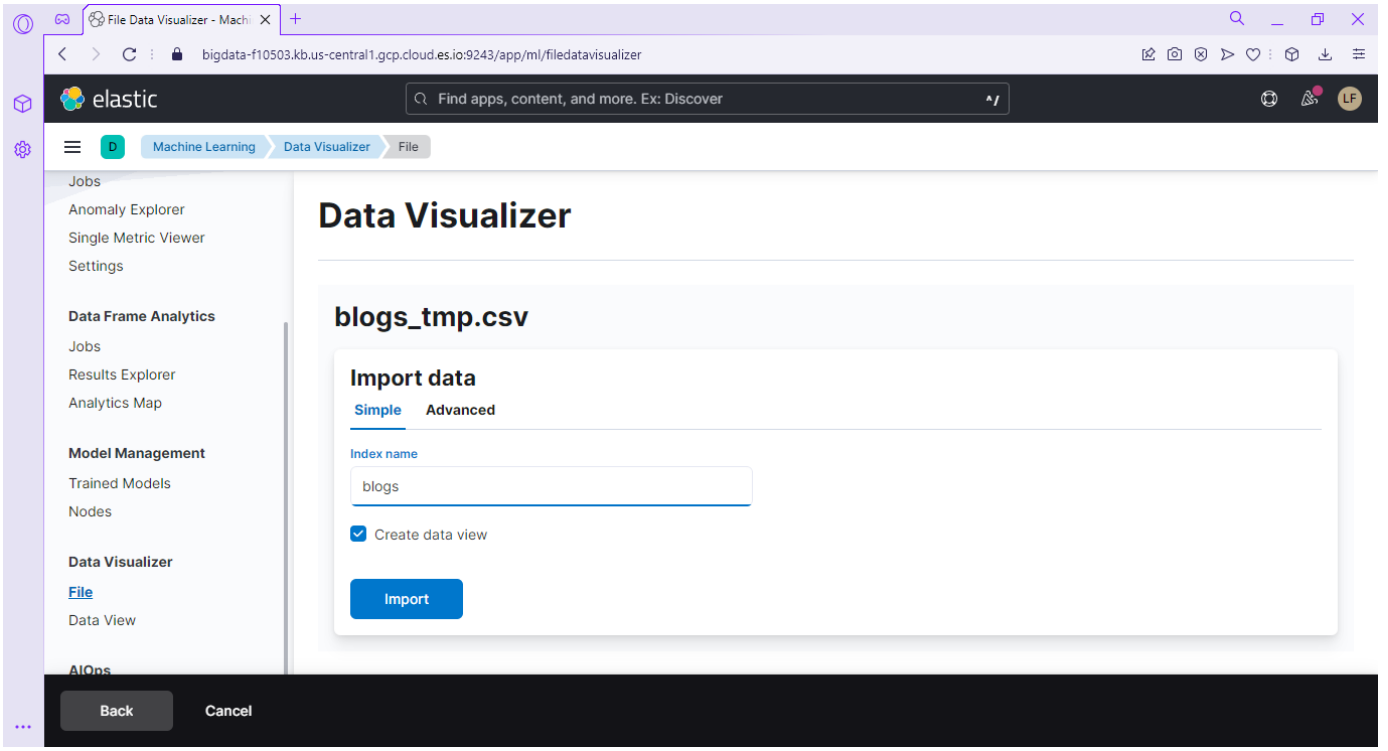
Como podemos ver en la anterior pantalla podremos arrastrar o buscar un archivo qu cumpla con los requerimientos que aquí se especifican. Subiremos un archivo CSV llamado "blogs_tmo.csv".



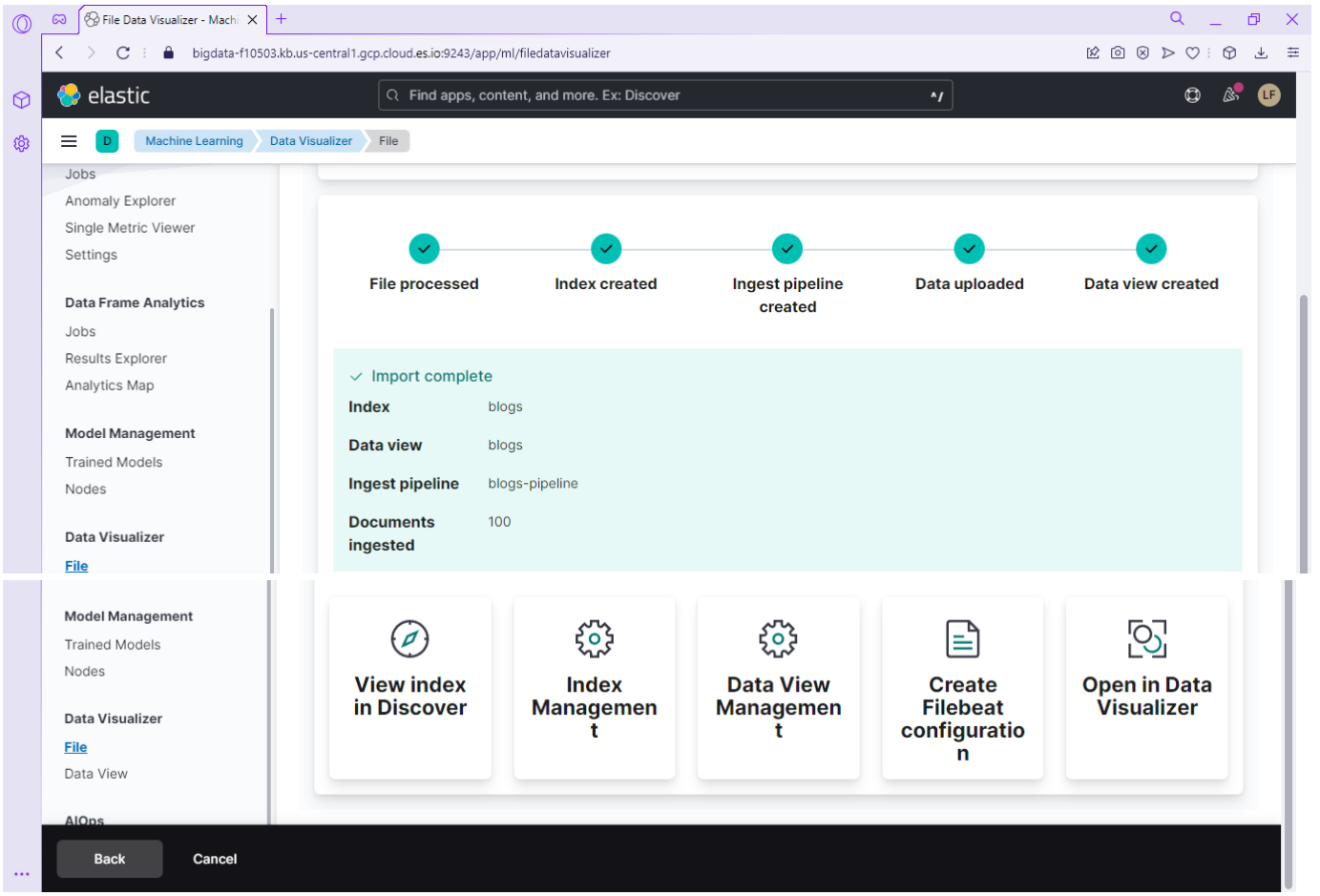
Al subir nuestro dataframe, podemos corroborar su estado en la pantalla:



Continuamente podemos dar clic en Import habiendo verificado que todo este bien. Luego asignamos un nombre para el índice y damos nuevamente en Import.

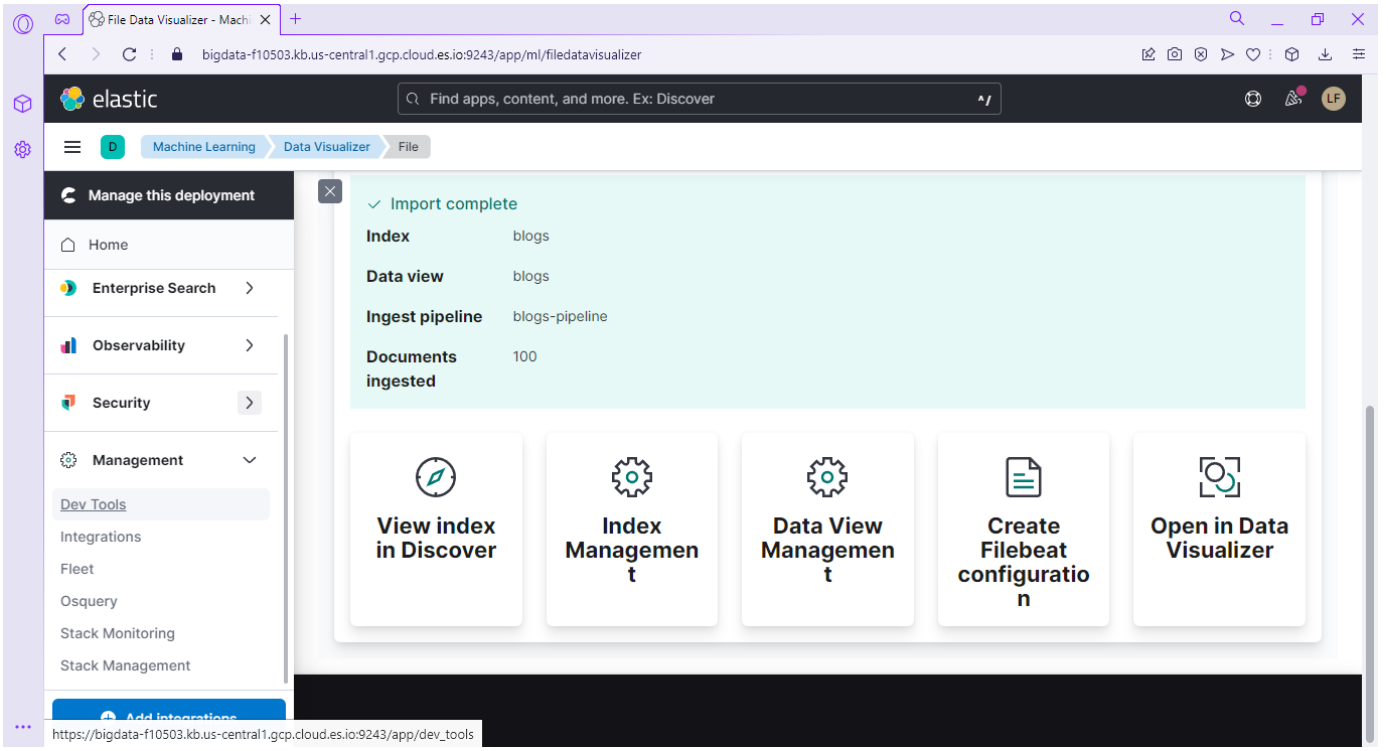


Luego esperamos a que se completen todos los pasos clave en el proceso de la subida de nuestro dataframe.

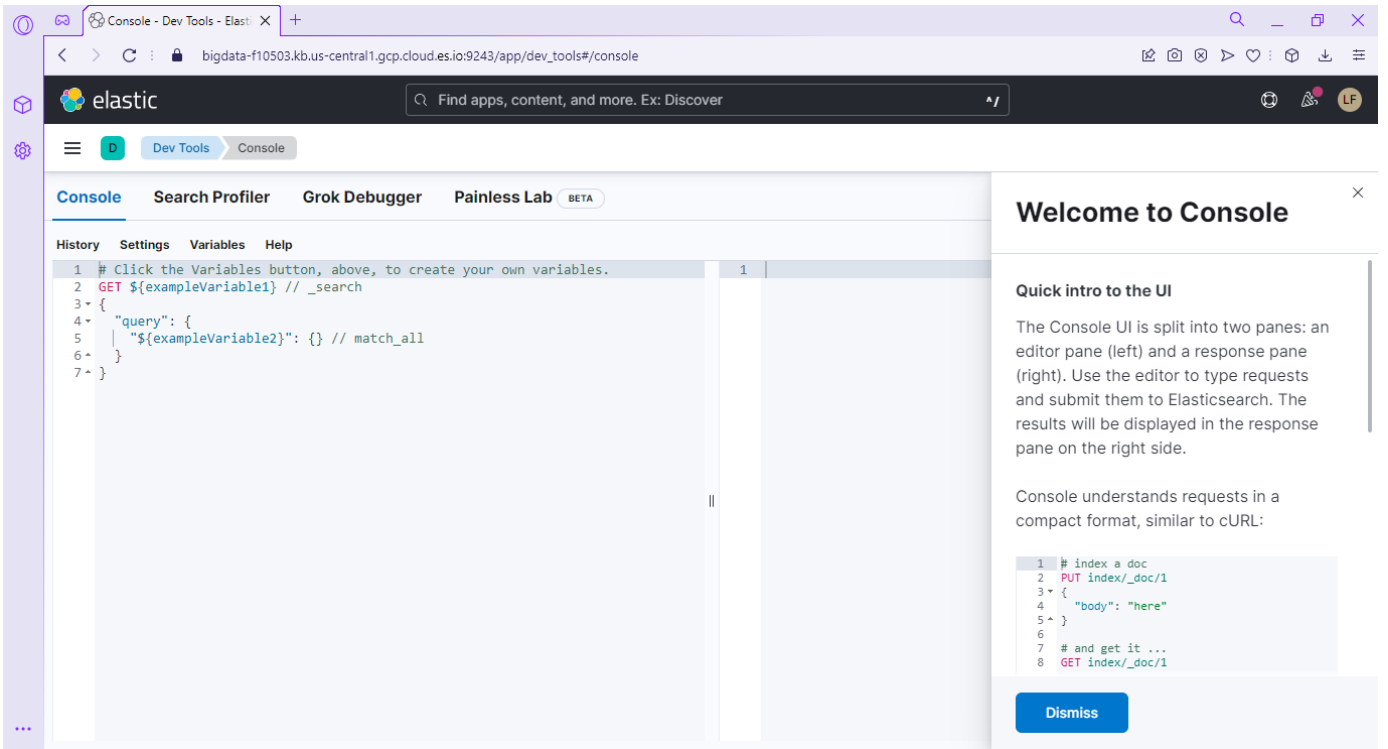


CONSULTAS SOBRE EL DATAFRAME

Y listo, tenemos nuestro DataFrame en nuestro Elastic, ahora podemos empezar a realizar consultas en el. Para esto volvemos al menú lateral izquierdo en la sección de "Management" allí seleccionamos "DevTools".



Al dar clic se nos abrirá la siguiente ventana tipo consola:



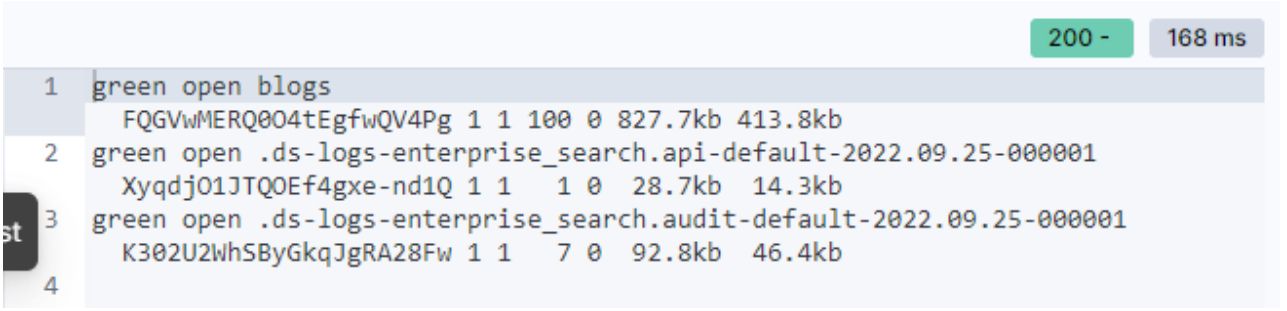
Primero verificamos que los datos estén correctamente almacenados en Elastic, esto por medio de la siguiente función:

```
GET _cat/indices
```

Para ejecutarla basta con dar clic en el símbolo de "Run" ▶️ que se encuentra en la misma linea.

```
# Click the Variables button, above, to create your own variables.
GET ${exampleVariable1} // _search
{
  "query": {
    "${exampleVariable2}": {} // match_all
  }
}
GET _cat/indi
    _cat/indices                    endpoint
```

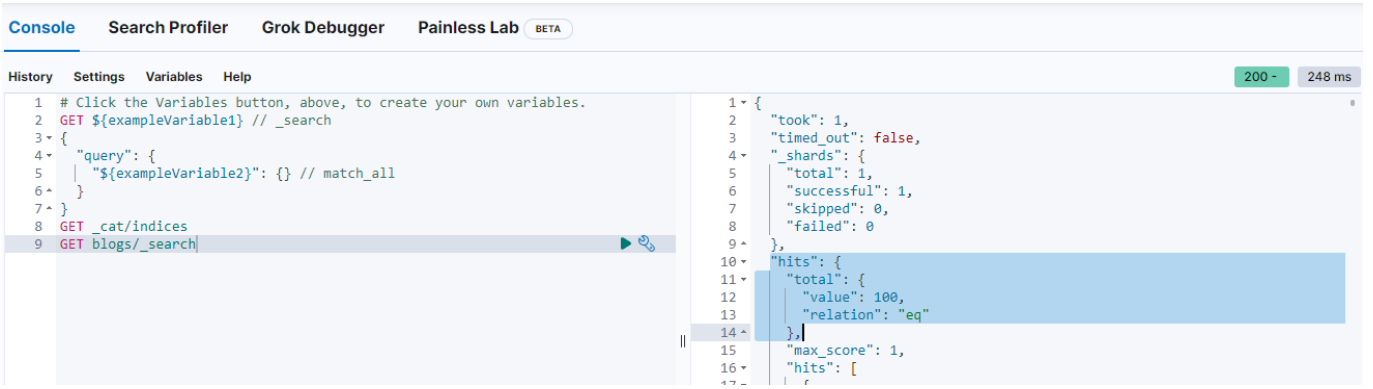
La respuesta es:



Ahora verificamos que los datos que subimos sean los correctos con la siguiente función:

```
GET blogs/_search
```

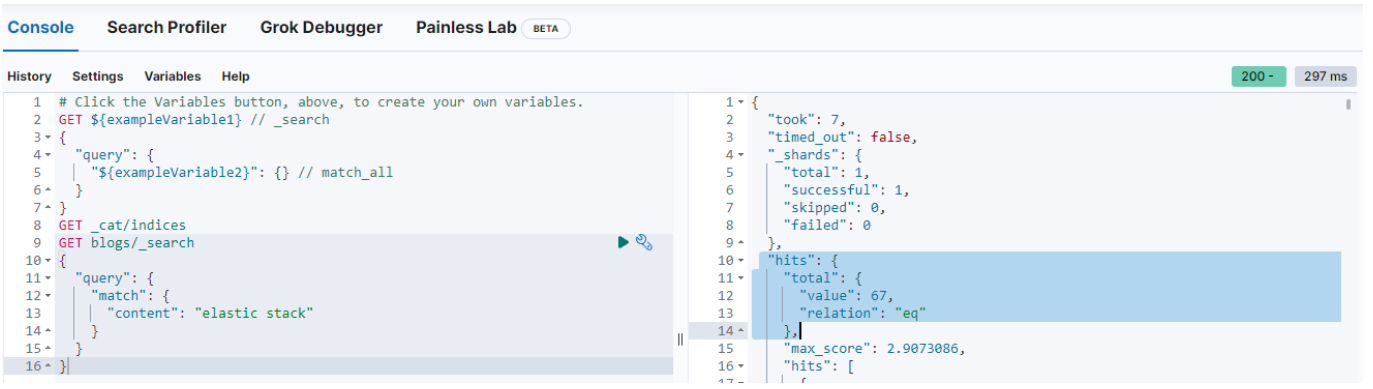
Como respuesta obtenemos lo siguiente:



Podemos consultar una palabra clave dentro de los blogs con la siguiente consulta:

```
GET blogs/_search
{
  "query": {
    "match": {
      "content": "elastic stack"
    }
  }
}
```

Dando como resultado que encontramos 67 registros con la directriz dada.



CONSTRUIR UNA PLANTILLA PARA LAS CONSULTAS

Las plantillas predefinidas de consultas nos benefician al momento de agilizar nuestras consultas, esto es debido a que en vez de que cada tipo de consulta escribamos todo el query respectivo, con la plantilla creada, solo lo utilizamos una vez.
Creemos nuestra consulta de la siguiente manera:

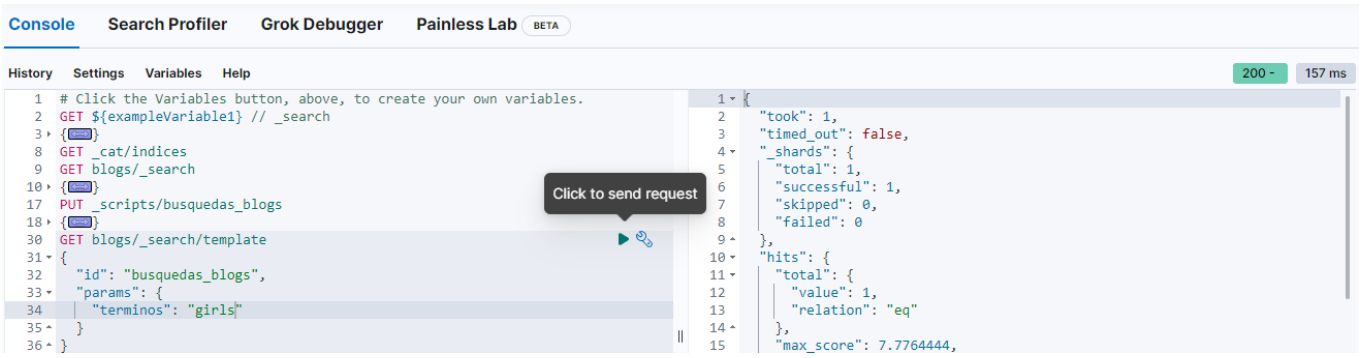
```
PUT _scripts/busquedas_blogs
{
  "script": {
    "lang": "mustache",
    "source": {
      "query": {
        "match": {
          "content": "{{terminos}}"
        }
      }
    }
  }
}
```

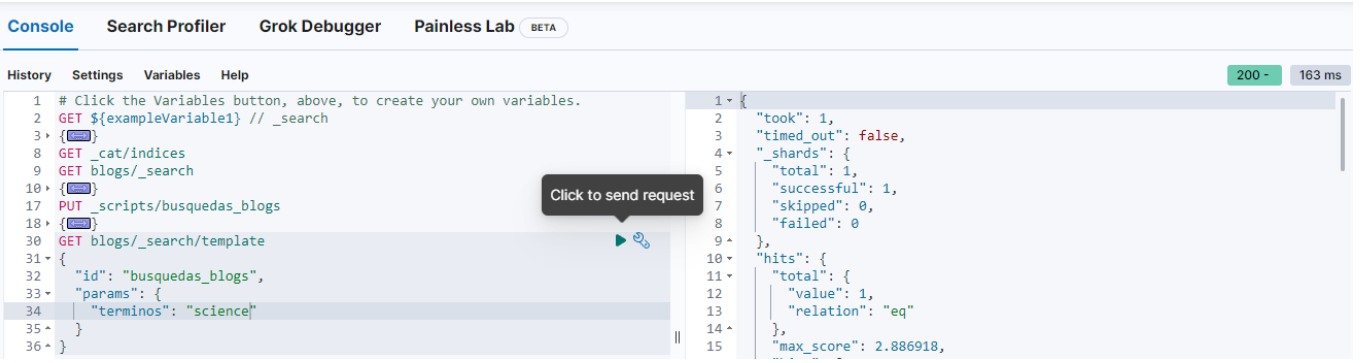
Lo cual se ve así con su confirmación de creación.

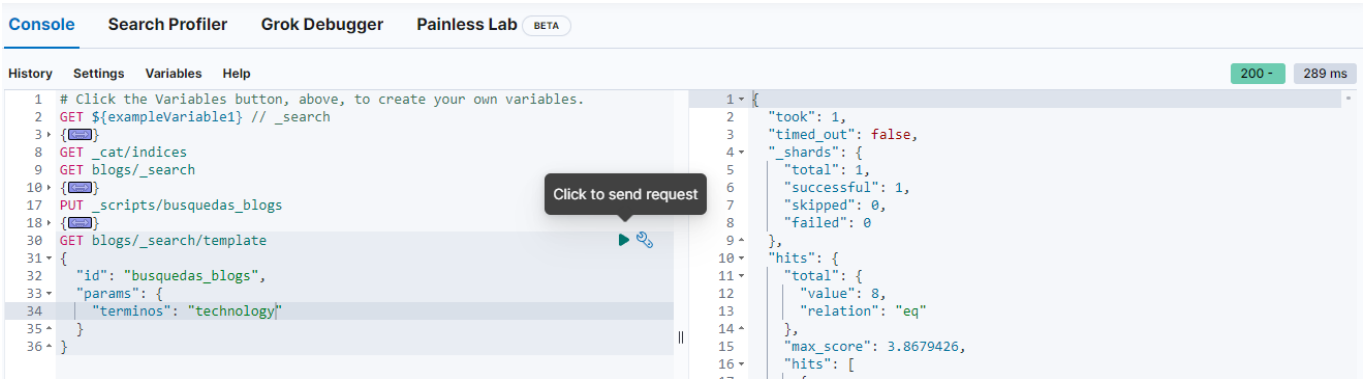
Una vez creada la plantilla podemos utilizarla en diversas consultas, en este caso de términos entre todos los registros.

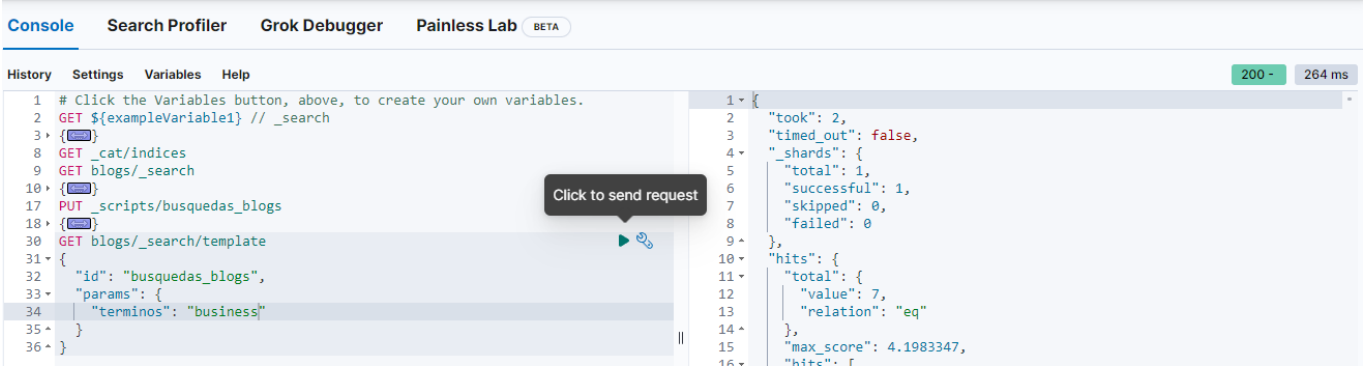
```
GET blogs/_search/template
{
  "id": "busquedas_blogs",
  "params": {
    "terminos": "AQUI PONER EL TERMINO A BUSCAR"
  }
}
```

Obteniendo resultados de la siguiente forma:



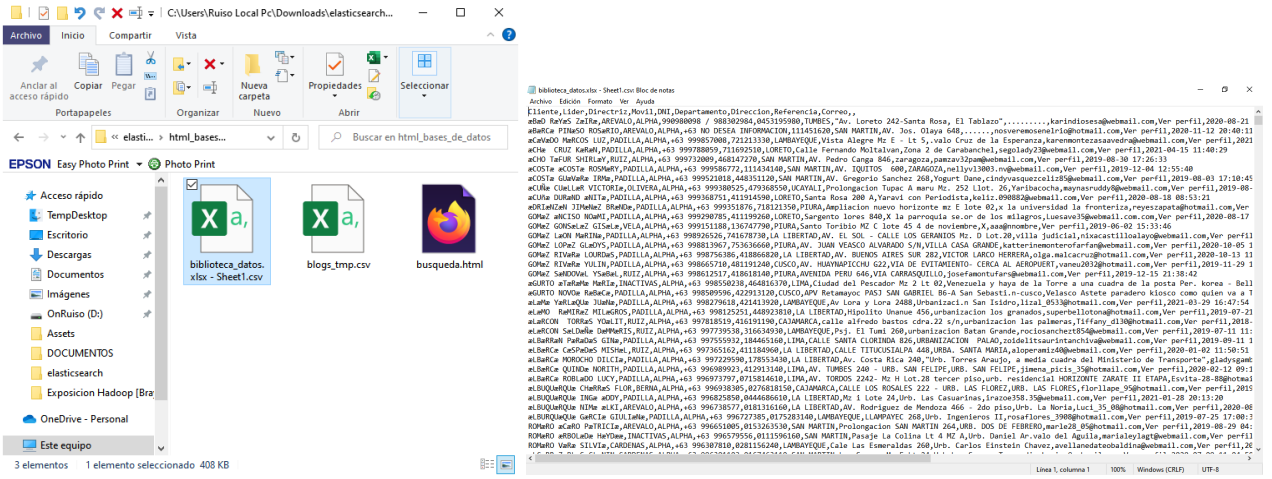




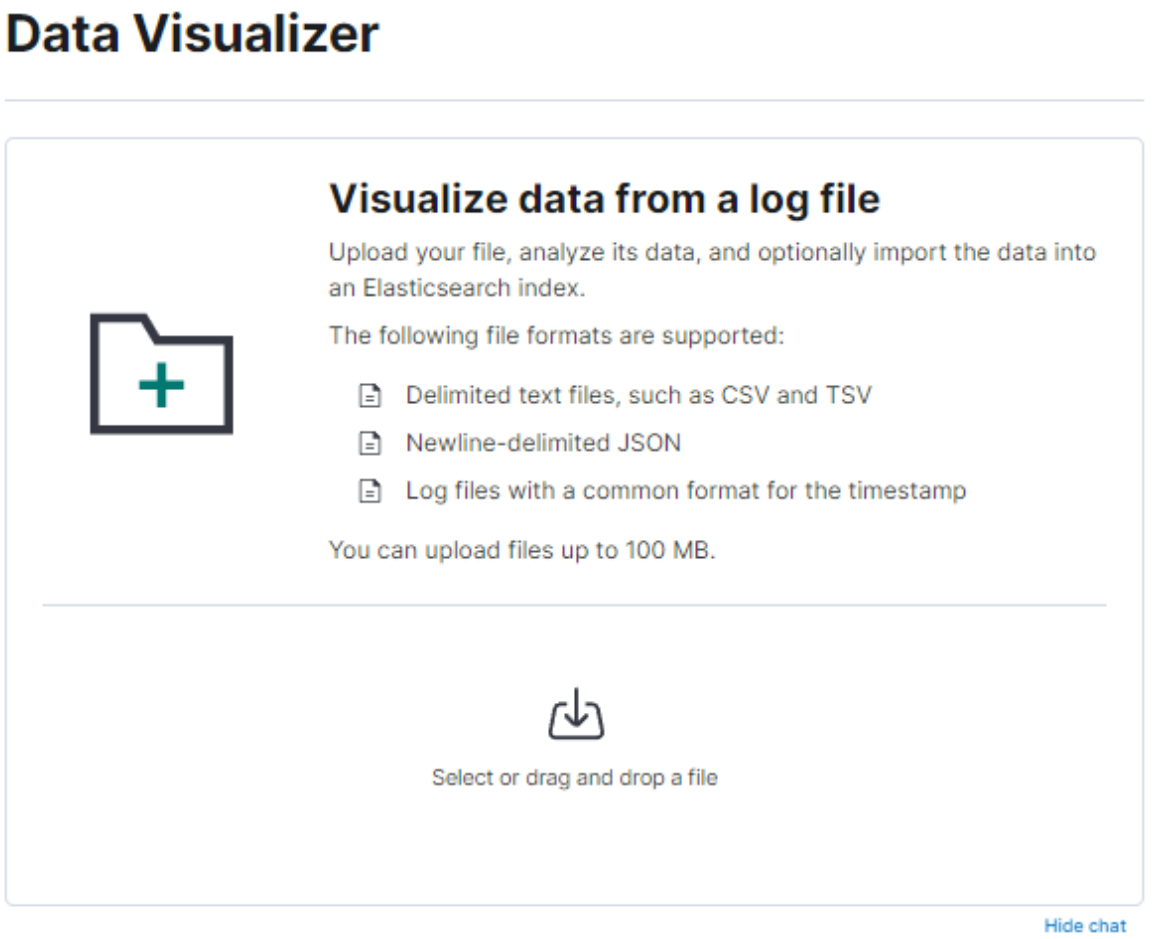


UN NUEVO TRABAJO

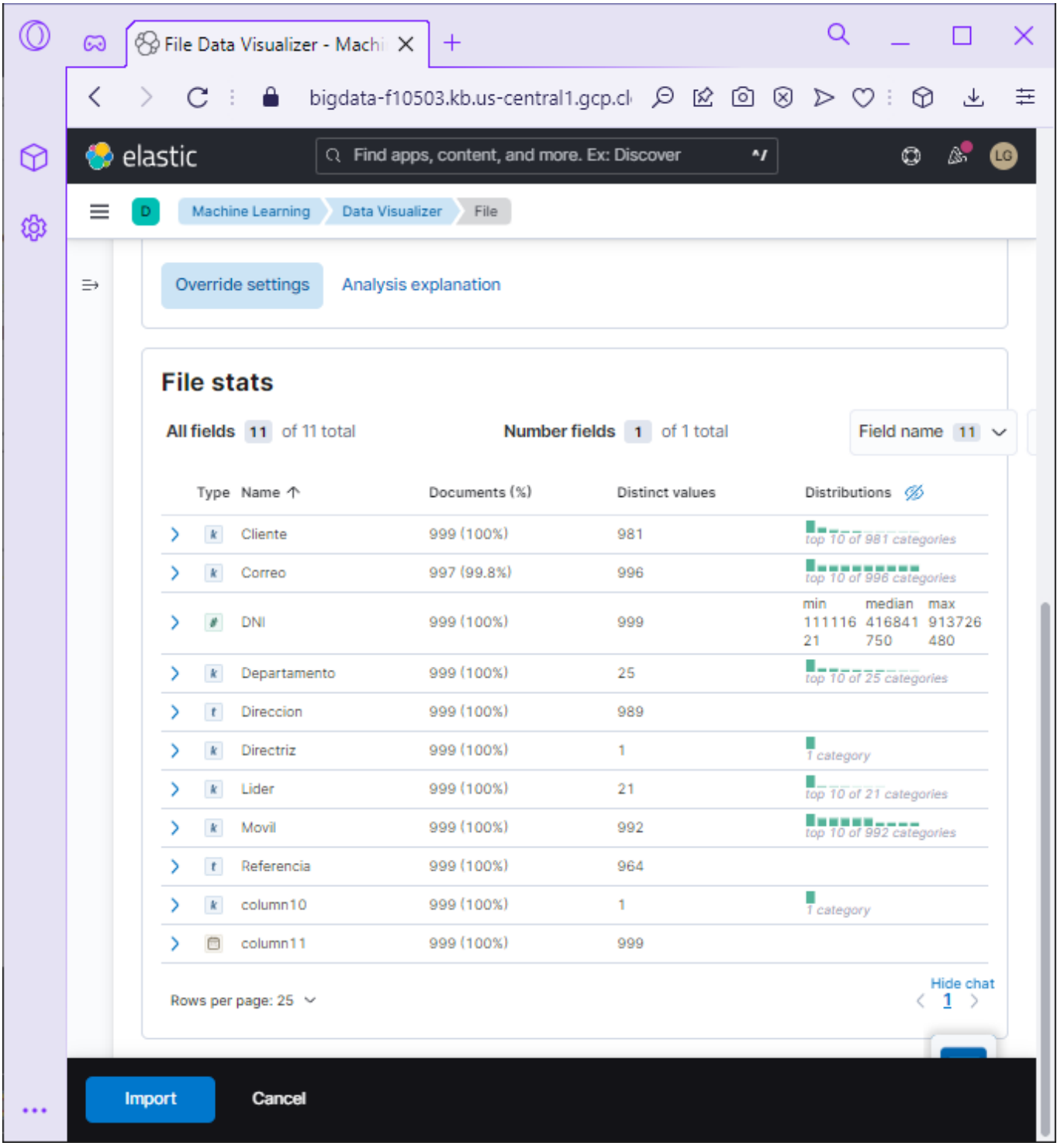
Tratemos de hacer lo mismo ahora con otro dataframe:



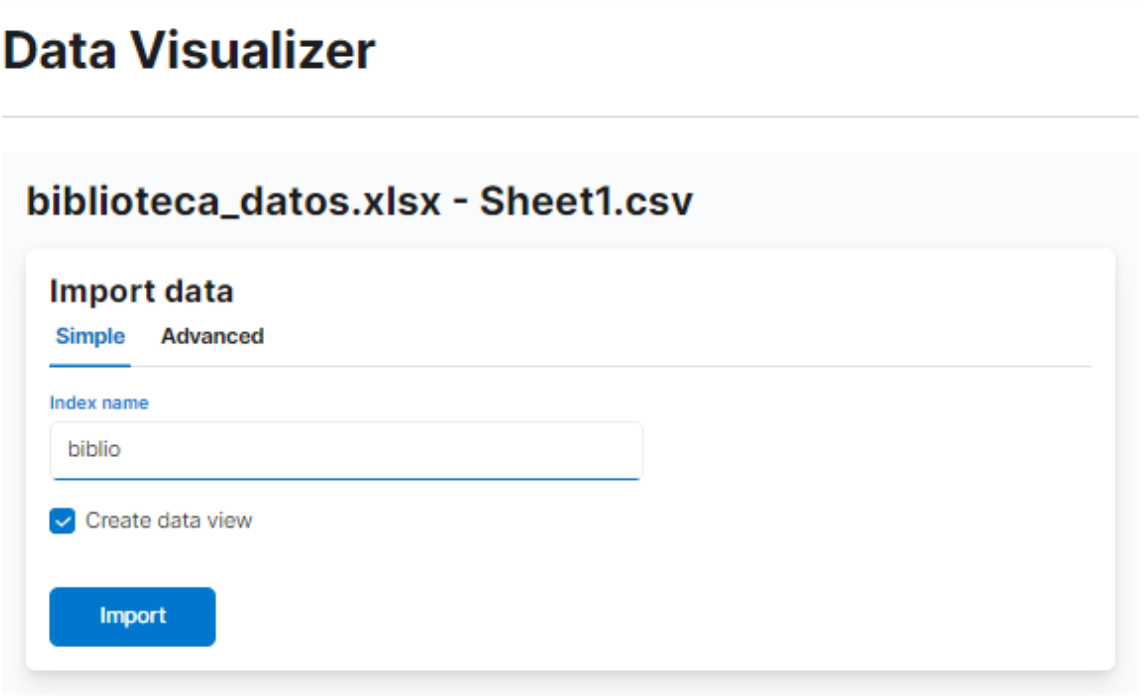
Subir el DataFrame en Data Visualizer.



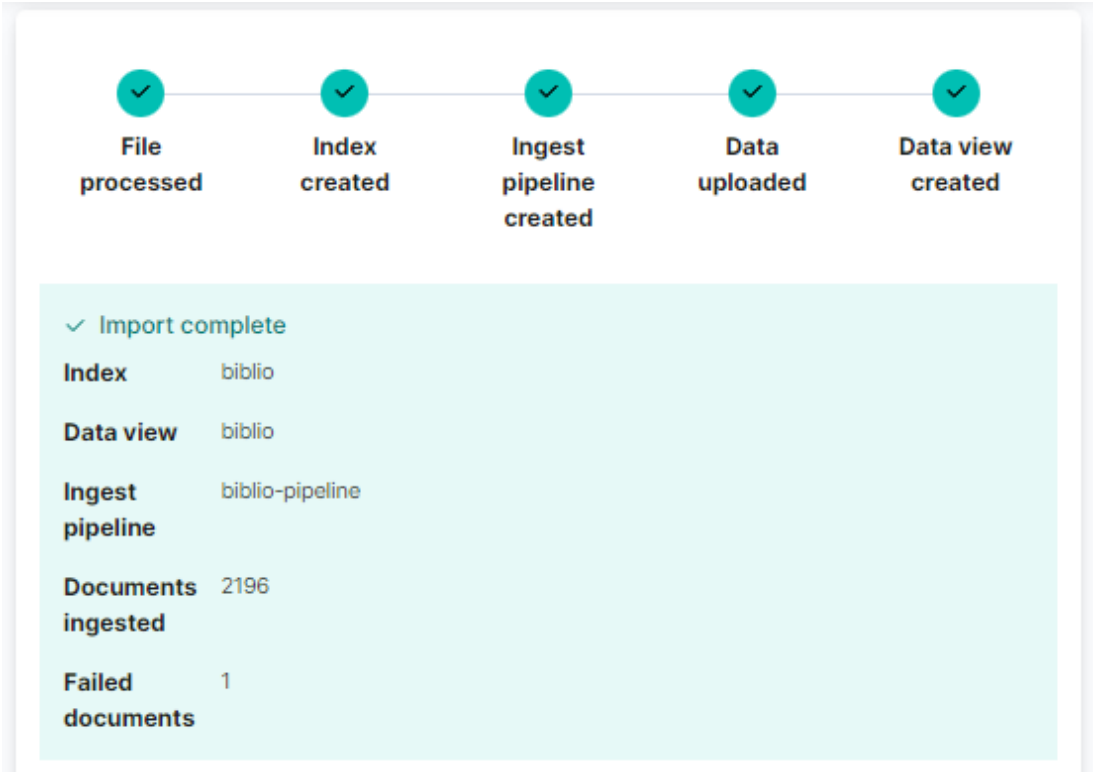
Importar el dataframe tras verificar el estado de la subida.



Indicar el indice de la data



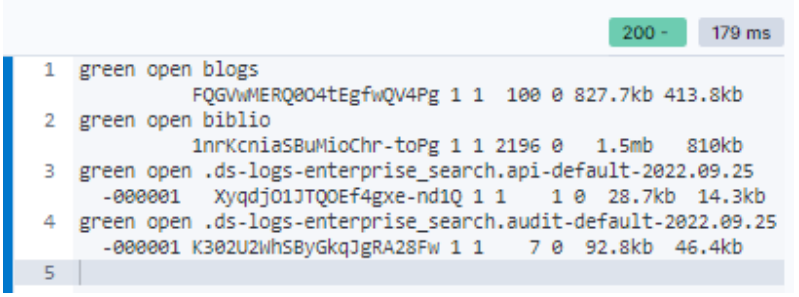
Confirmar la data subida.



Acceder a la consola de consultas. Confirmación de los datos almacenados.

```
GET _cat/indices
```

Este es el resultado:



Verificación de los datos sean los correctos:

```
GET biblio/_search
```

Tenemos alrededor de 2196 resultados en nuestro dataframe, que comparten una estructura similar a la siguiente

Ahora hagamos una consulta mas especifica, por ejemplo sobre la columna de departamento.

```
GET biblio/_search
{
  "query": {
    "match": {
      "Departamento": "TUMBES"
    }
  }
}
```

Tenemos una respuesta de 22 consultas que tienen como departamento TUMBES.

Ahora hagamos una consulta sobre cuantas personas hay en este dataframe de la ciudad de CUSCO.

```
GET biblio/_search
{
  "query": {
    "match": {
      "Departamento": "CUSCO"
    }
  }
}
```

Tenemos como respuesta a 36 personas con el valor de departamento = CUSCO.

Sabiendo de antemano que los nombres de los lideres son AREVALO, PADILLA, RUIZ, OLIVERA, VELA, INACTIVAS, BERNA, CARDENAS, GONZAGA JUAREZ, HUILCA, VERA, LOPEZ, REAT, GUI, ROJAS, FRETEL, JULCA, MESTANZA, REYES, JIMENEZ, DAVILA, CAYAMPE, VILLENA, PINEDA, ANGELES, SAAVEDRA. Podemos hacer una consulta burda por cada uno y contar los hits obtenidos.

```
GET biblio/_search
```

Sumándole uno por uno

```
{"query": {"match": {"Lider": "AREVALO"}}}
{"query": {"match": {"Lider": "PADILLA"}}}
{"query": {"match": {"Lider": "RUIZ"}}}
{"query": {"match": {"Lider": "OLIVERA"}}}
{"query": {"match": {"Lider": "VELA"}}}
{"query": {"match": {"Lider": "INACTIVAS"}}}
{"query": {"match": {"Lider": "BERNA"}}}
{"query": {"match": {"Lider": "CARDENAS"}}}
```

```
{"query": {"match": {"Lider": "GONZAGA"}}}
{"query": {"match": {"Lider": "JUAREZ"}}}
{"query": {"match": {"Lider": "HUILCA"}}}
{"query": {"match": {"Lider": "VERA"}}}
{"query": {"match": {"Lider": "LOPEZ"}}}
{"query": {"match": {"Lider": "REATEGUI"}}}
{"query": {"match": {"Lider": "ROJAS"}}}
{"query": {"match": {"Lider": "FRETEL"}}}
{"query": {"match": {"Lider": "JULCA"}}}
{"query": {"match": {"Lider": "MESTANZA"}}}
{"query": {"match": {"Lider": "REYES"}}}
{"query": {"match": {"Lider": "JIMENEZ"}}}
{"query": {"match": {"Lider": "DAVILA"}}}
{"query": {"match": {"Lider": "CAYAMPE"}}}
{"query": {"match": {"Lider": "VILLENA"}}}
{"query": {"match": {"Lider": "PINEDA"}}}
{"query": {"match": {"Lider": "ANGELES"}}}
{"query": {"match": {"Lider": "SAAVEDRA"}}}
```

Esto nos da como resultado:

	A	B	C	D	E	F	G	H
1	AREVALO	183	REATEGUI	79	CARDENAS	91	CAYAMPE	1
2	PADILLA	1194	ROJAS	13	GONZAGA	24	VILLENA	1
3	RUIZ	144	FRETEL	18	JUAREZ	61	PINEDA	1
4	OLIVERA	11	JULCA	4	HUILCA	98	ANGELES	1
5	VELA	54	MESTANZA	7	VERA	16	SAAVEDRA	1
6	INACTIVAS	111	REYES	4	LOPEZ	12		
7	BERNA	50	JIMENEZ	2	DAVILA	15		