# Buscadores elasticos


## ElasticSearch

**Luis Fernando Castellanos Guarin**

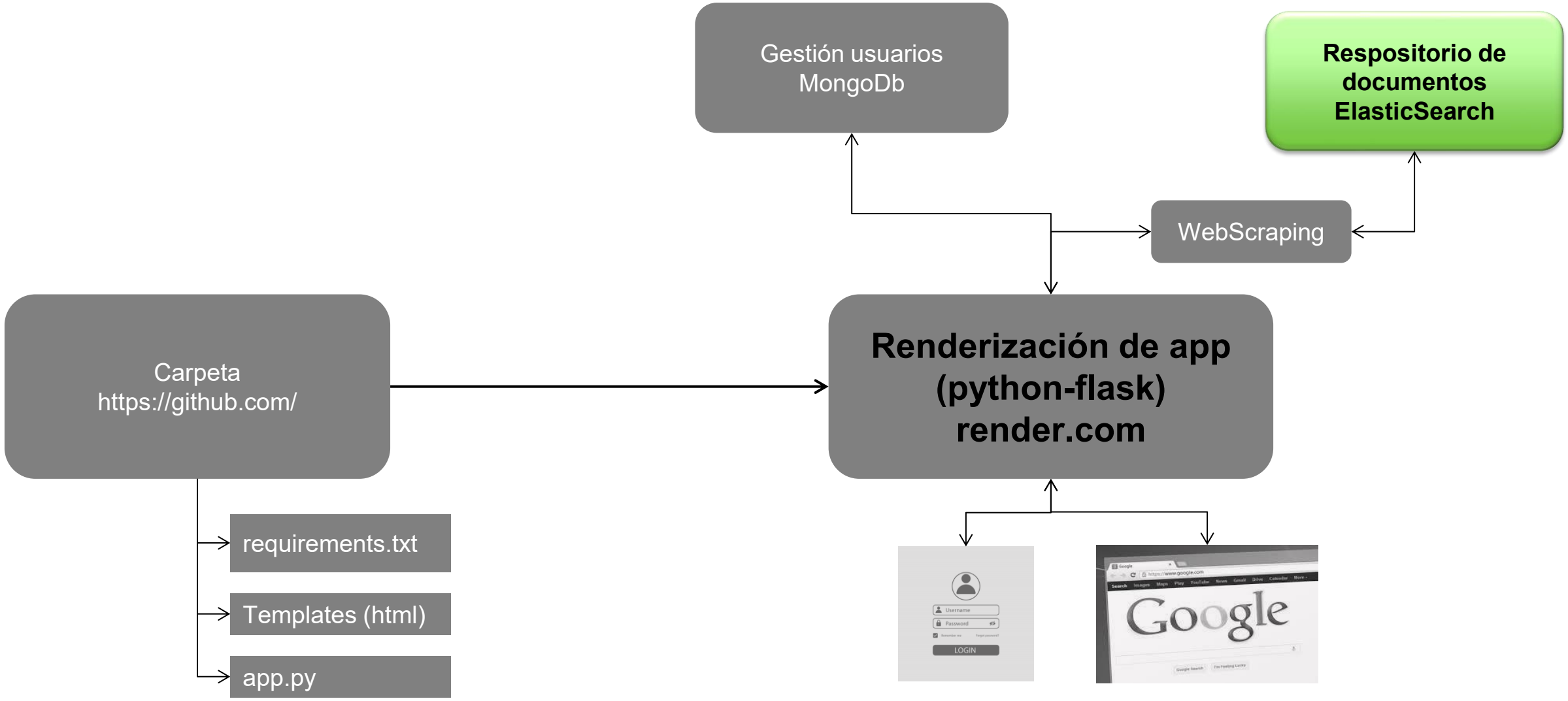
**2025**

UNIVERSIDAD CENTRAL

YO ♥ UC

# 1. Proyecto final (renderización web)

UNIVERSIDAD CENTRAL



# 2. ¿Qué es una búsqueda elástica?

UNIVERSIDAD CENTRAL

- Motor de búsqueda **distribuido** y **open source** basado en **Apache Lucene**.
- Permite **almacenar, indexar y buscar grandes volúmenes de datos** casi en tiempo real.
- Opera con **documentos JSON**, no tablas relacionales.
- Se usa en sistemas de **análisis de texto, monitoreo, inteligencia documental y ciberseguridad**.

| Concepto | Descripción |
|---|---|
| **Documento** | Unidad mínima de información (similar a un registro JSON). |
| **Índice (Index)** | Conjunto de documentos de una misma estructura o tema. |
| **Campo** | Cada atributo o propiedad dentro de un documento. |
| **Mapping** | Esquema o definición de los tipos de campos (texto, fecha, entero, etc). |
| **Análisis** | Proceso de convertir texto en tokens para indexar y buscar (tokenización, stemming, stopwords, etc). |
| **Query DSL** | Lenguaje JSON para realizar consultas complejas (match, bool, range, aggregation, etc). |

# 3. ElasticSearch / OpenSearch

UNIVERSIDAD CENTRAL

- Una búsqueda elástica es un proceso rápido, escalable y flexible para localizar información relevante dentro de grandes volúmenes de datos en texto libre.
- Se basa en el motor Lucene, que indexa documentos para hacer consultas por relevancia (no exactas)

No busca coincidencias exactas como una base de datos SQL → usa análisis lingüístico, relevancia y puntuación (score).

**Ejemplo:**

Buscas “niños jugando”

y el motor también encontrará documentos con resultados como:

- “**infantes** que juegan”
- “**niñas** cansadas que **jugaron**”

Gracias al análisis del lenguaje.

```
{
  "autor": "Andersen",
  "tipo": "infantil",
  "texto": "Había una vez una princesa que vivía en un castillo de cristal."
}
```

*Ejemplo de documento JSON*

# 4. Empresas que usan ElasticSearch/OpenSearch

UNIVERSIDAD CENTRAL

Principales empresas:

- **Netflix** → búsqueda de títulos y recomendaciones.
- **Uber** → monitoreo de logs y análisis en tiempo real.
- **Wikipedia** → búsquedas internas.
- **GitHub** → búsqueda de código.
- **NASA** → exploración de datos científicos.
- **Corte Constitucional de Colombia** → buscador de jurisprudencia/expedientes/documentos judiciales
- **JEP**

**Ventajas competitivas:**

- Búsqueda en tiempo real sobre grandes volúmenes.
- Escalabilidad horizontal (clusters).
- Potente análisis textual multilingüe.
- Integración con Kibana / Dashboards.
- Extensible a Machine Learning y búsquedas vectoriales.

# 5. ¿Cómo se indexan documentos? (Análisis lingüístico)

UNIVERSIDAD CENTRAL

**Etapas del análisis en español:**

1. **Tokenización**: divide el texto en palabras.
2. **Normalización**: convierte a minúsculas, elimina acentos.
3. **Eliminación de StopWords**: quita palabras comunes (“el”, “la”, “y”, “pero”).
4. **Lematización / stemming**: reduce palabras a su forma base (“niños” → “niño”, “jugando” → “jugar”).
5. **Indexación**: los tokens resultantes se almacenan en un índice invertido.

# 6. Ejemplo de indexación de un cuento infantil

UNIVERSIDAD CENTRAL

**Texto original:**

“Los niños jugaban felices en el parque y soñaban con volar alto.”

**Resultado :**

["niño", "jugar", "feliz", "parque", "soñar", "volar", "alto"]

# 7. Como indexar un documento

UNIVERSIDAD CENTRAL

```
client.index(
  index="cuentos",
  id=1,
  body={
    "autor": "Bécquer",
    "tipo": "infantil",
    "texto": "Los niños jugaban bajo la luna y hablaban con el viento."
  }
)
```

# 7. Como indexar masivamente

UNIVERSIDAD CENTRAL

```
acciones = [
  {"index": {"_index": "cuentos", "_id": 1}},
  {"autor": "Poe", "tipo": "terror", "texto": "La criatura se arrastró en la oscuridad."},
  {"index": {"_index": "cuentos", "_id": 2}},
  {"autor": "García Márquez", "tipo": "político", "texto": "El presidente soñó con un país sin hambre."}
]
client.bulk(body=acciones)
```

# 8. Tipos de búsquedas

UNIVERSIDAD CENTRAL

| Tipo | Ejemplo | Descripción |
|---|---|---|
| **Match** | "match": {"texto": "dragón"}} | Busca relevancia textual. |
| **Term** | "term": {"tipo": "terror"}} | Coincidencia exacta (palabra clave). |
| **Bool (filtros)** | Must, Should, Filter | Combinaciones lógicas de condiciones. |
| **Range** | "range": {"fecha_creacion": {"gte": "1900"}} | Filtra por rangos numéricos o de fecha. |

# 3. Ejemplo de una búsqueda

UNIVERSIDAD CENTRAL

```
// Búsqueda básica
{
  "query": {
    "match": {
      "texto": "magia y dragones"
    }
  }
}

// Búsqueda con filtro por campo
{
  "query": {
    "bool": {
      "must": {"match": {"tipo": "fantástico"}},
      "filter": {"range": {"fecha_creacion": {"gte": "2000-01-01"}}}
    }
  }
}
```

# 4. Servidor de elasticSearch

UNIVERSIDAD CENTRAL

## 1. https://elastic.co/

### 1.1 Loguearse gratis
### 1.2 Crear un index  (free)

## 2. Ejercicios en Google colab

https://www.inesdi.com/blog/big-data-analytics-que-es-y-para-que-sirve/

UNIVERSIDAD CENTRAL

YO ♥ UC

# Gracias