

### Descripción del problema:

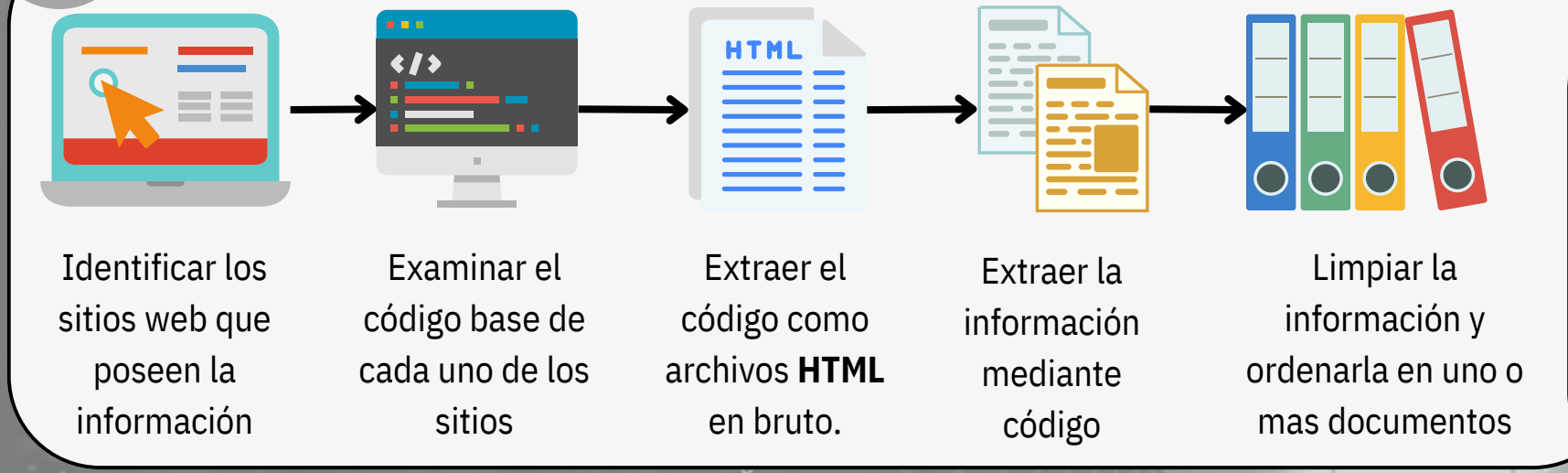
Muchos documentos tales como **reglamentos y normativas** pueden ser complejos de entender para el usuario común. Por lo que procesos que impliquen el uso de esta información pueden ser complejos y generar demasiadas dudas.

### Solución Propuesta:

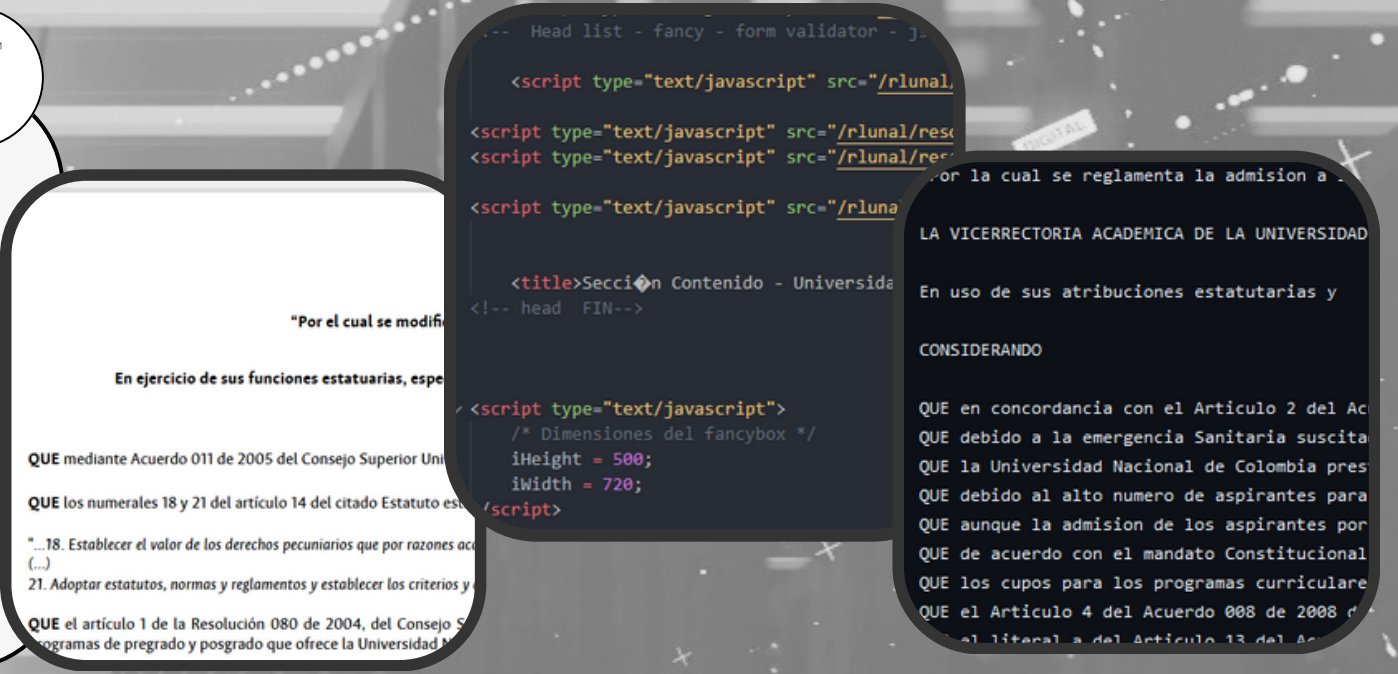
Implementar un **modelo de inteligencia artificial** el cual pueda.

- Aprender sobre documentos.
- Responder preguntas respecto a los contextos documentales.
- Tener una interfaz de manejo sencillo o intuitivo.

### 1 Recolección de información



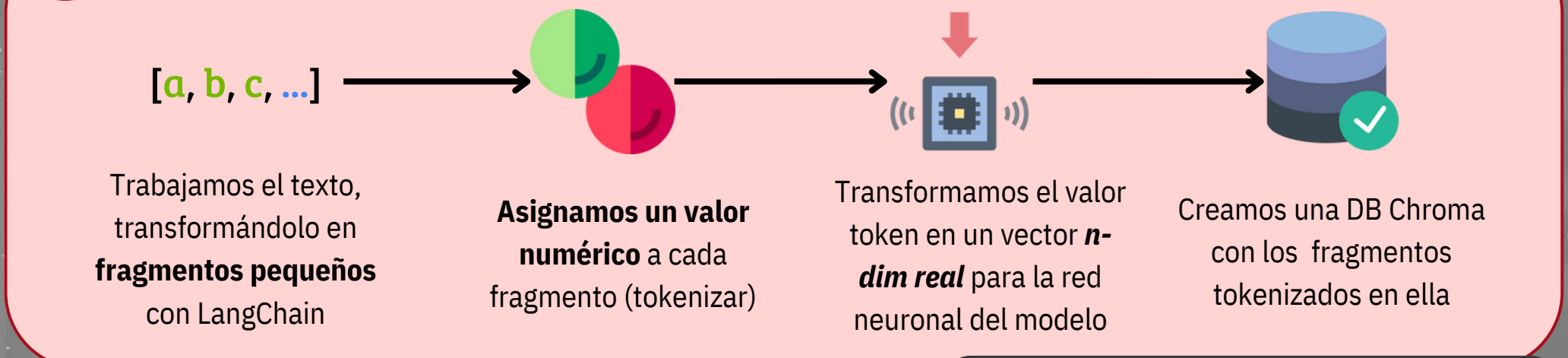
Beautifulsoup python



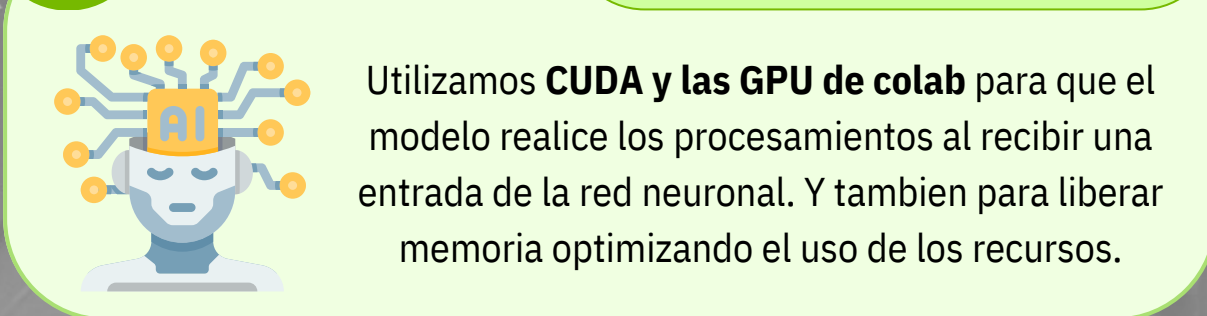
### 2 Obtención del Modelo



### 3 Transformando Datos



### 4 Procesamiento

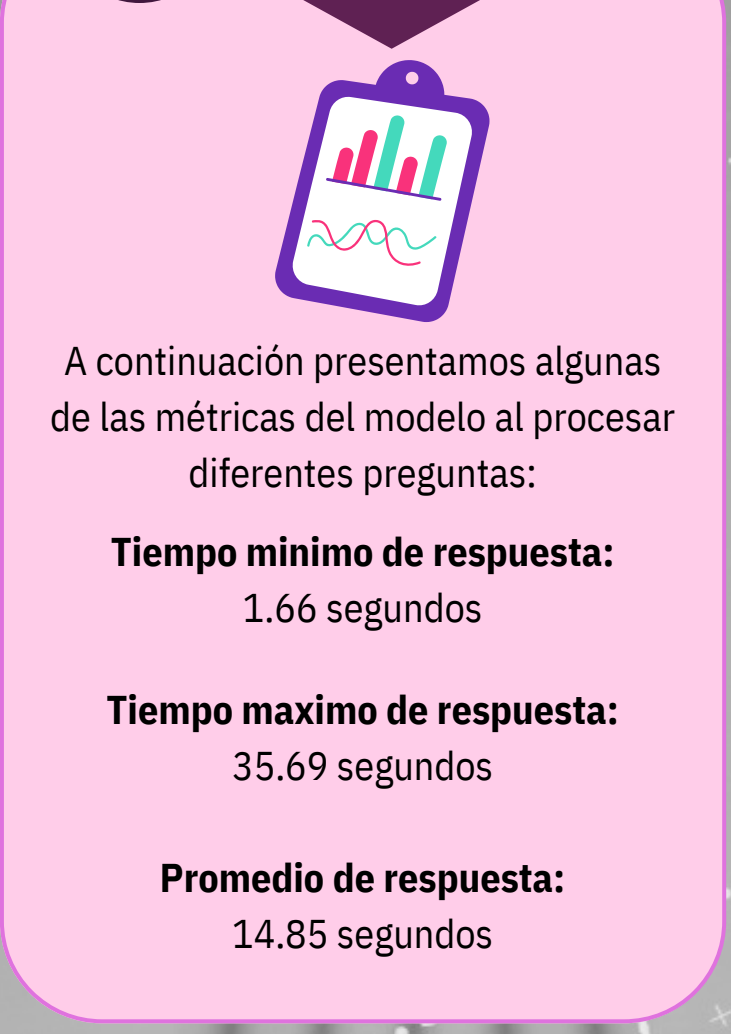


```
1 from langchain.text_splitter import RecursiveCharacterTextSplitter
2
3 text_splitter = RecursiveCharacterTextSplitter(
4     separators=["\n", ".", ",", ":", ";"],
5     chunk_size=100,
6     chunk_overlap=10,
7     length_function = len,
8     add_start_index = True
9 )
10
11 documents = text_splitter.split_documents(documents)
12
13 print(f'Cantidad de fragmentos generados: {len(documents)}\n')
14 print(documents)
```

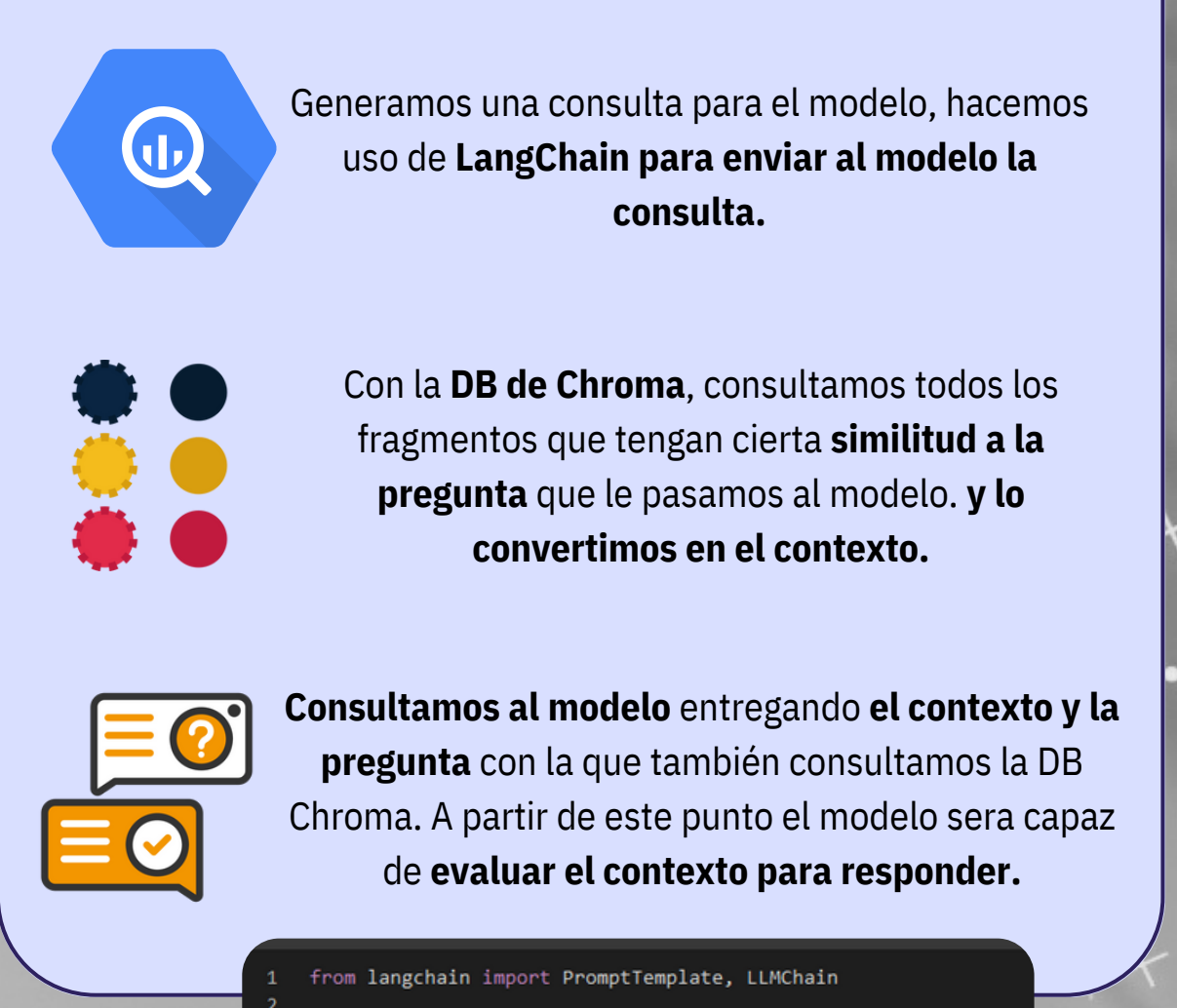
### 7 Despliegue



### 6 Metricas



### 5 Uso y Consultas



```
1 from langchain import PromptTemplate, LLMChain
2
3 user_input = "Que dice la normativa de admision a la Universidad Nacional de Colombia?"
4
5 prompt = PromptTemplate(
6     input_variables=["question"],
7     template="{question}"
8 )
9
10 gc.collect()
11 torch.cuda.empty_cache()
12
13 llm_chain = LLMChain(prompt=prompt, llm=local_llm)
14
15 print(llm_chain.run(user_input))
```