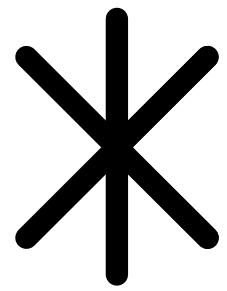


KnowX

CROODS



# Equipo - CROODS



Carolina González



Ozner Leyva



Sofia Cantú



Roberto García



Diego Gutiérrez

**“Una empresa puede invertir miles de dólares en la investigación, sin embargo, si no tiene claro el problema a investigar esos dólares serán un desperdicio.”**

Gustavo Riveros, Jefe de Análisis de Mercado de la empresa Bavaria SA

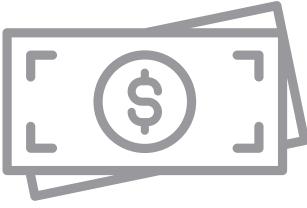
**La falta de un objetivo claro dificulta la determinación de la información deseada, lo cual conduce a una técnica inefectiva de recolección de datos. Estas técnicas deben adaptarse a la información que se busca.**



# Nuestra Visión y Misión



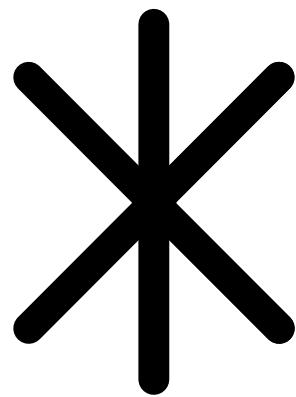
*Ahorro de Tiempo*



*Gastos Conscientes*

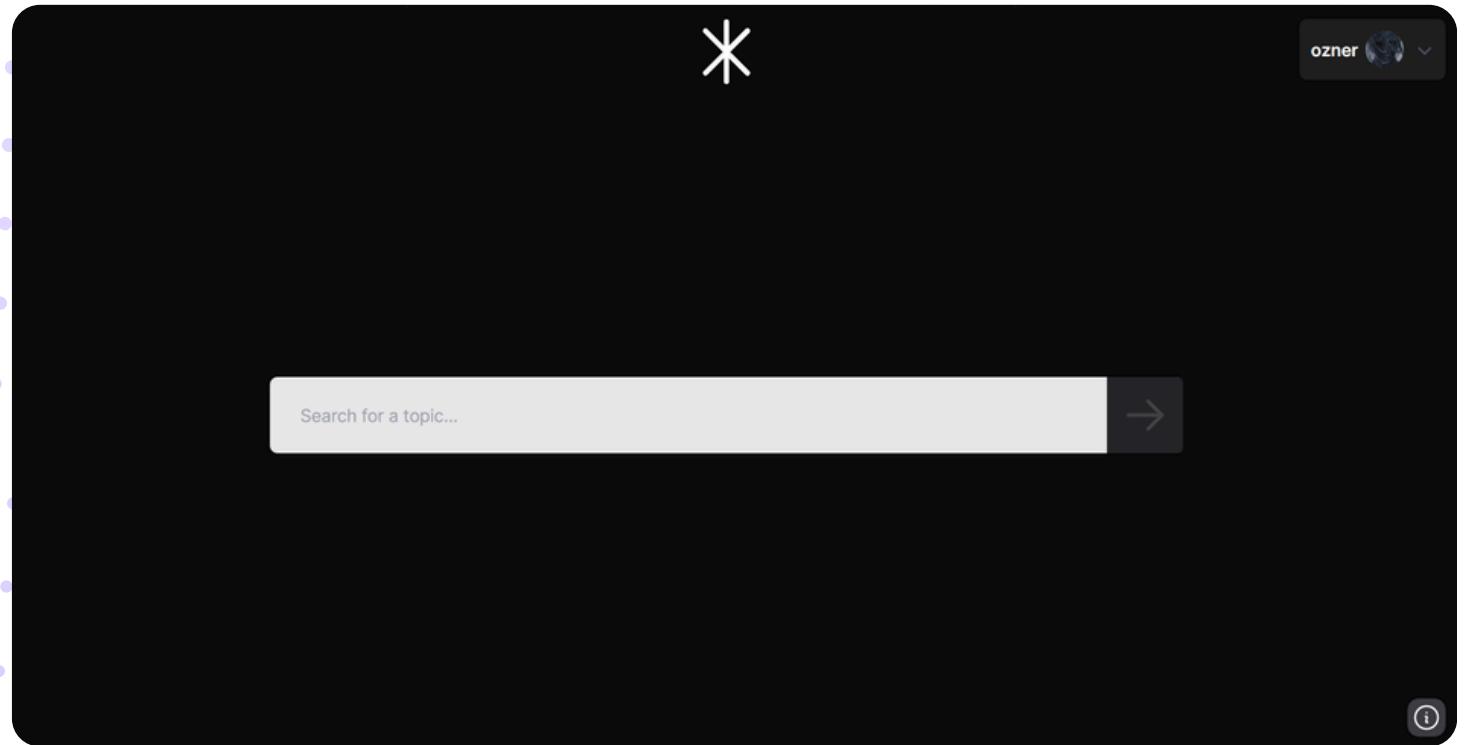


*Decisiones Actualizadas*

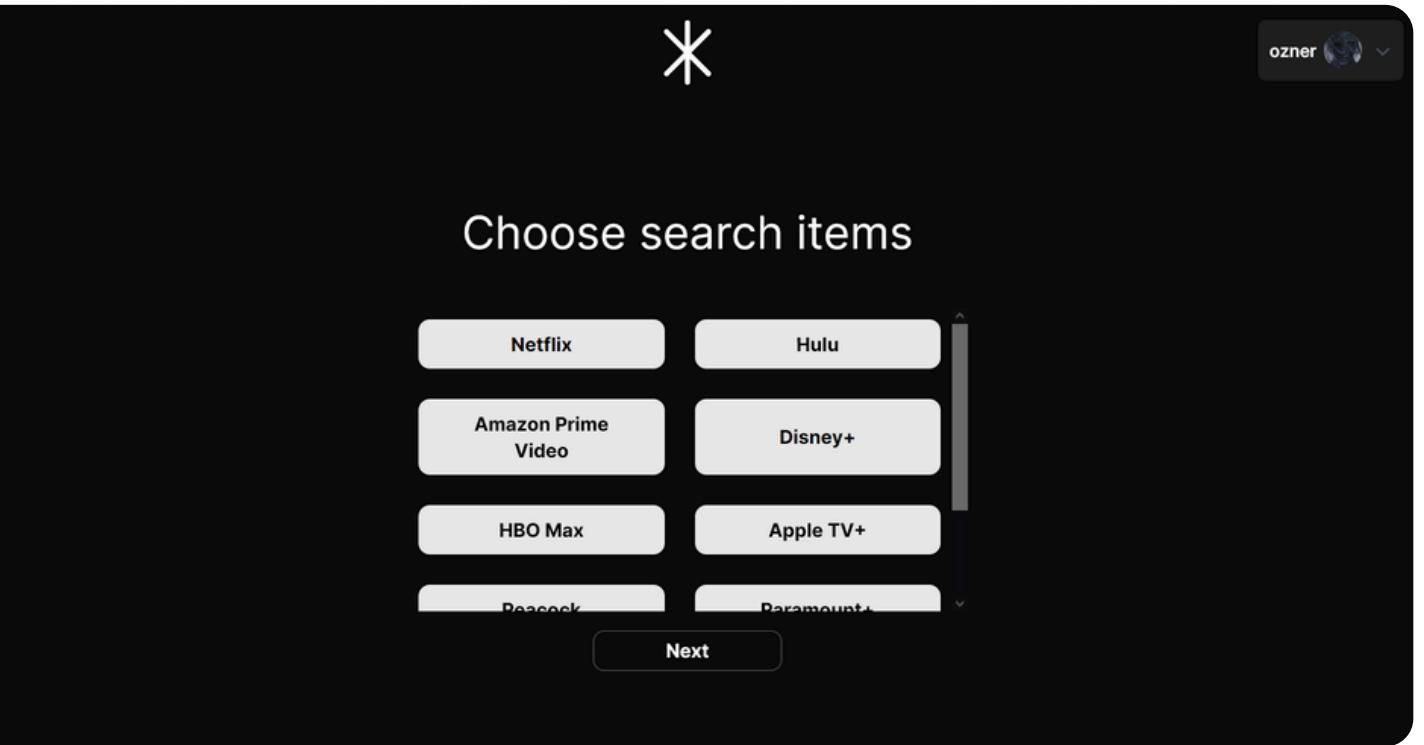


KnowX

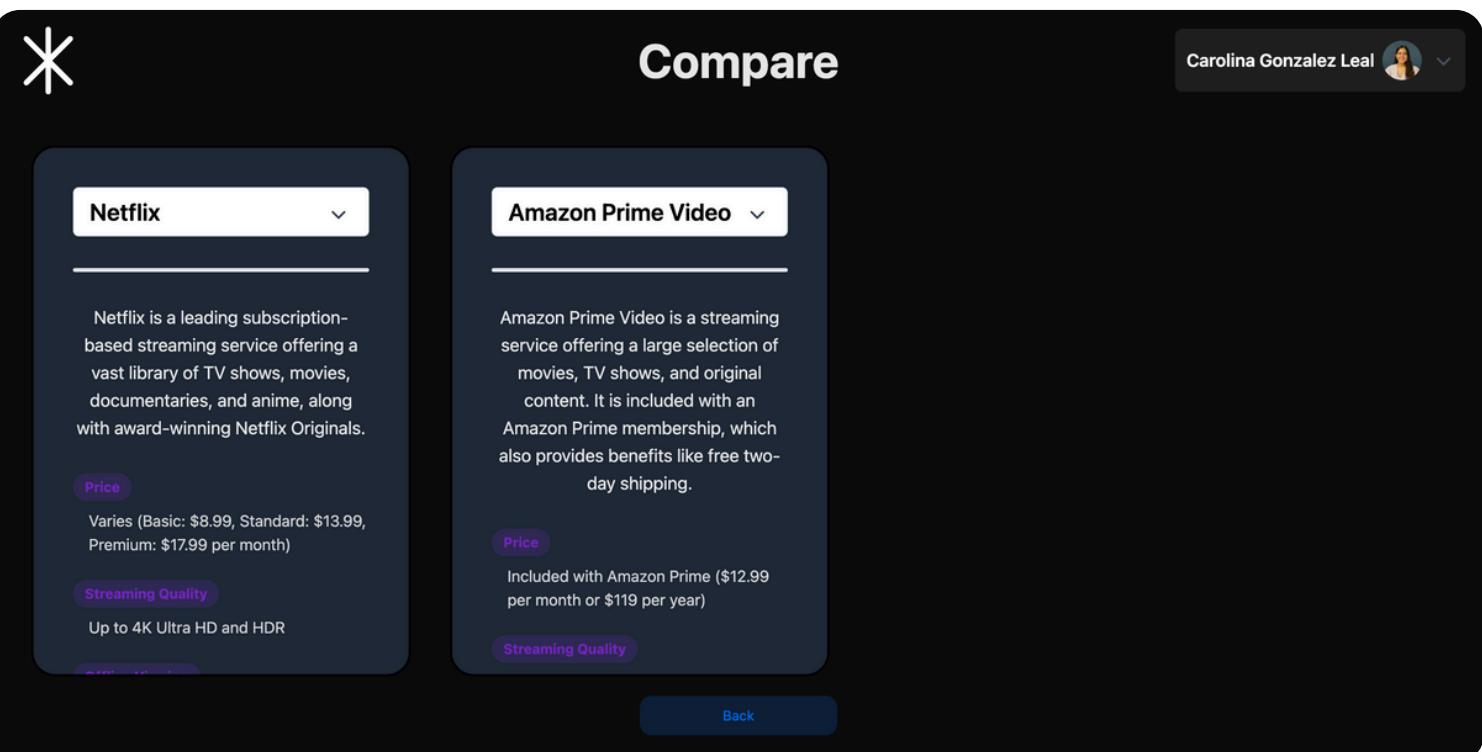
KnowX es una herramienta que *facilita* y  
*optimiza* el proceso de investigación de  
mercado mediante el uso de AI.



*Busca*



*Especifica*



*Compara*

# Modelo de negocio B2B



Investigadores de mercado pueden tomar ventaja de las funciones a través de suscripciones.



## SAAS

Se crearán planes de uso que las empresas podrán seleccionar dependiendo de sus necesidades.



Empresas pueden establecer un convenio de filial con la plataforma

# KnowX

## External Components

Auth Providers

WebScraper

Link consulting

Page parser

Ai Interpreter

## Presentation

Authentication

Dashboard

History

Subscription Plan

Search detailing

Results

Compare

Account

## Integration

Auth Providers account  
permission API

Topic Detailing API

Dynamic History API

Web Scraping API

User History API

Subscription API

## Storage



Serverless Relational  
DB



Browser Cookies

## Legend

Future Integrations

Current Integrations

# KnowX

## External Components



## Presentation

Authentication, Dashboard, History, Search Detailing, Results, Compare



## Integration

Auth Providers account permission, User History



Web scraping, topic detailing



## Storage

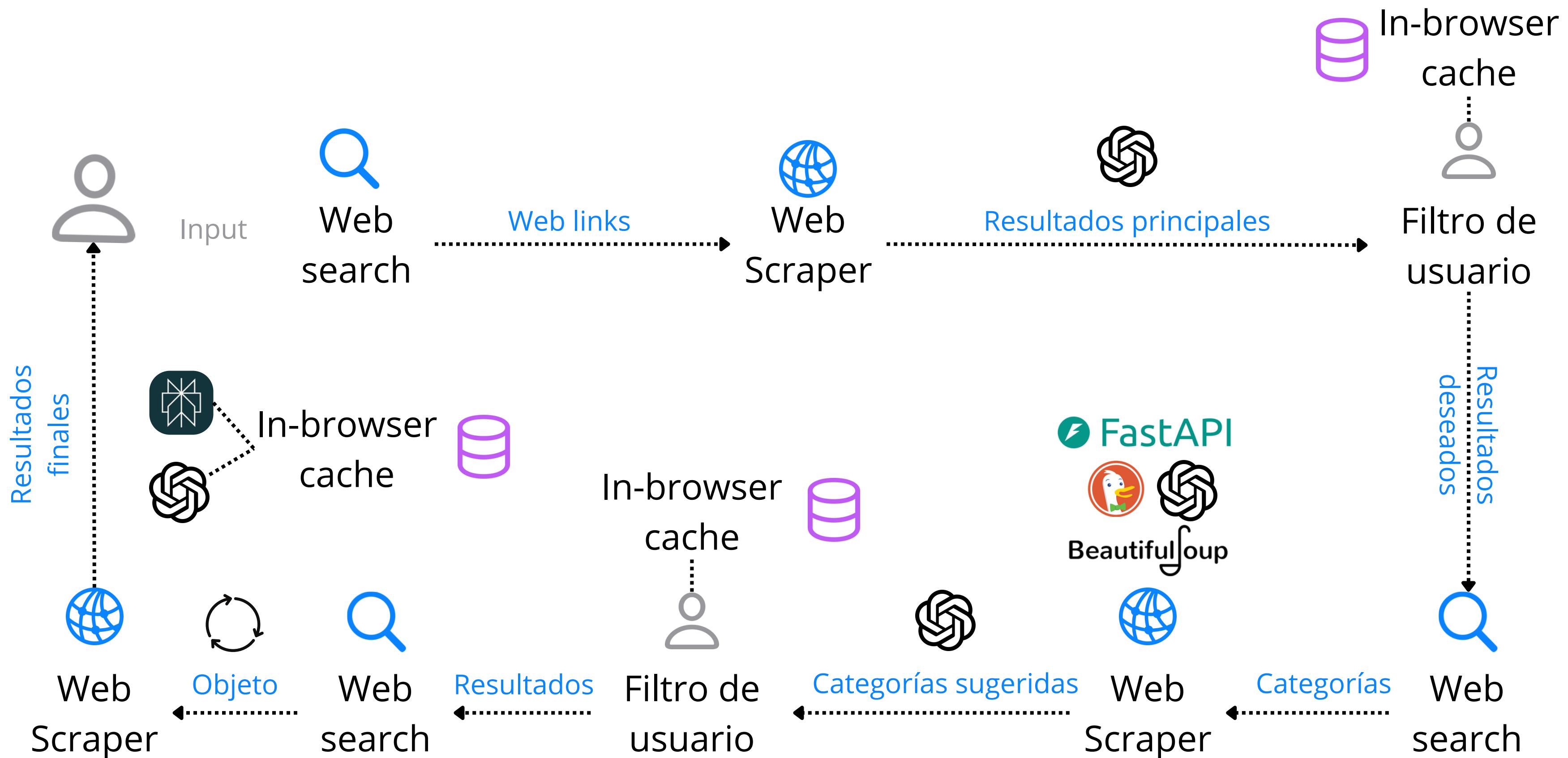
PostgreSQL Neon



Serverless Relational DB



# Tecnología



# La magia de KnowX



```
prompts = [
    ("Top objects extractions", f"""We are talking about {givenObject}. Based on what it is, what it offers and the given information,
    please return the top results or objects that ONLY FITS this category. Strictly return ONLY a comma separated list WITH NO
    numeration with the different sub-topic/object names. Try to extract as many objects as possible with a LIMIT of 10 results.
    If you can't find any, return an empty list. DO NOT INCLUDE OBJECTS OUT OF THE GIVEN CATEGORY. ONLY INCLUDE WHAT IS ASKED FOR
    SUCH AS COMPANIES, BRANDS, TOOLS, ETC. DO NOT COMBINE OBJECTS FROM DIFFERENT CATEGORIES."""),
]
```

```
prompts = [
    ("Features", f"""According to the topic {givenTopic}, please return the features that each object of the category should have.
    You will need to think like a market researcher and what features are important for the user, remember that depending on the
    topic one of the most important features is the price. The features should be separated by a comma. For example: "Price, Content
    Type, Devices, License Type". There should be a minimum of 3 features and a maximum of 8. The first feature MUST be "Description"."""),
]
```

```
"content": f"""We are talking about the following objects: {objects}. Based on what each of them are in the context of '{initialPrompt}',
return a dictionary, a description of the tool and the answers to following categories for EACH of these categories: {feature}
If no information is found on a category, return "N/A" on that key.
THE AMOUNT OF CATEGORIES RETURNED SHOULD ALWAYS EQUAL TO THE AMOUNT OF OBJECTS IN Categories inside results.
Strictly return ONLY a dictionary with the following structure NOTHING ELSE, your response must be able to be parsed from str

{
    categories: [..., ..., ...],
    results: [
        {
            Name: "",
            Description: "",
            Categories: [
                { Name: "...", Value: "..." },
                { Name: "...", Value: "..." },
                { Name: "...", Value: "..." },
                ...
            ]
        }
    ]
}
```

```
# Get raw information from the links provided
raw_documents = await scrape_links_to_documents(list(links))

# Get the documents in chunks from the information provided
documents = await resize_documents(raw_documents, CHUNK_SIZE, CHUNK_OVERLAP)

embeddings = OpenAIEMBEDDINGS(disallowed_special=())
vectorstore = await create_vectorstore(faiss.FAISS, embeddings, documents)

response = await run_chain_on(givenTopic, vectorstore, 1)
```

# El futuro de KnowX

