

localhost

📄 ⬇ ⬆ ⬇ ⬆

Ilmmain · Streamlit

Italy 4-0 Malta: Hosts win to keep pressure on Group C leaders England - BBC Sport

×

News Article URLs

URL 1

https://www.bbc.com/sport/football/67037

URL 2

URL 3

Process URLs


🚴 RUNNING...

Stop

Deploy

⋮

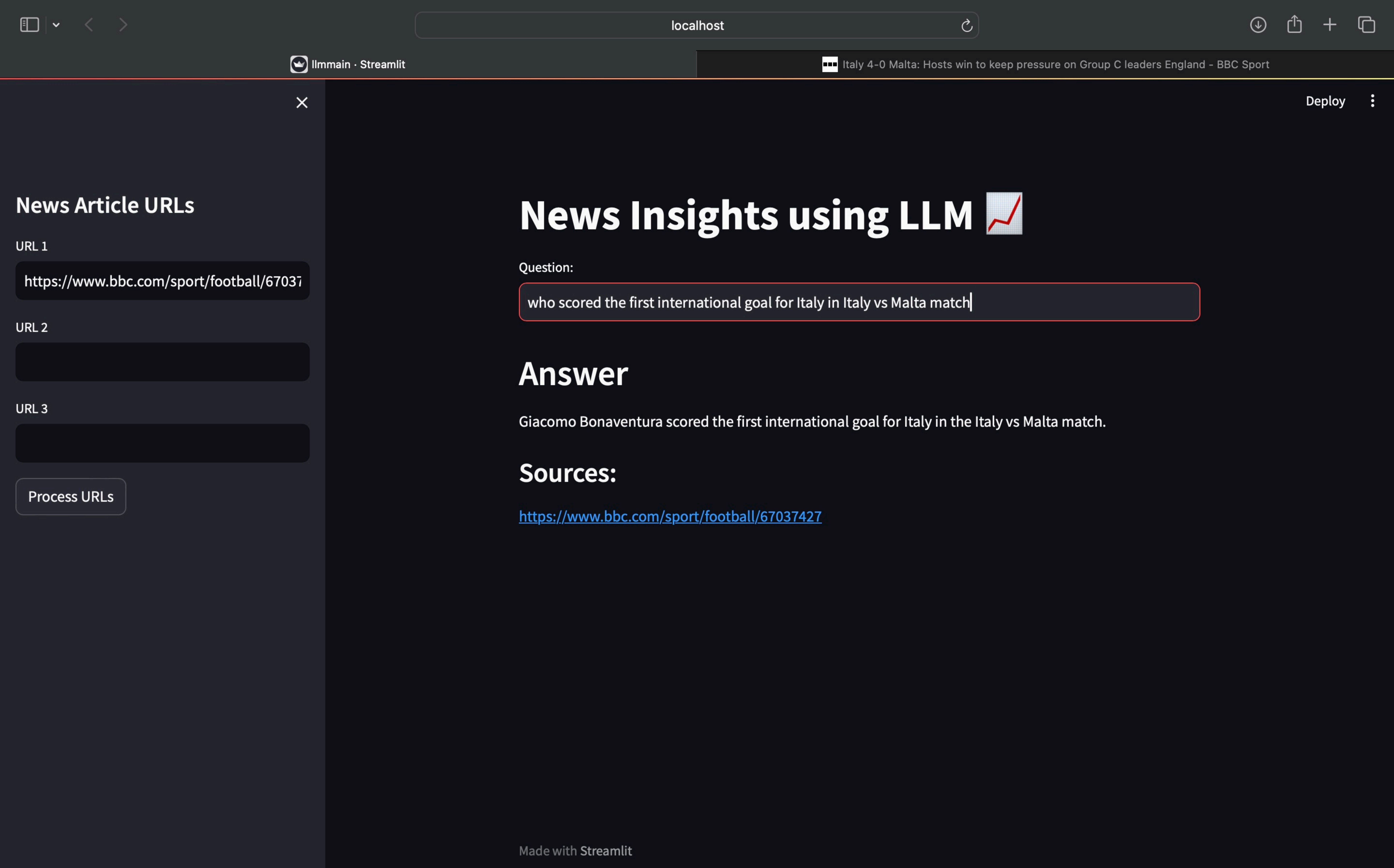
News Insights using LLM



Question:

who scored the first international goal for Italy in Italy vs Malta match

Made with Streamlit



News Article URLs

URL 1

https://www.bbc.com/sport/football/67037

URL 2

URL 3

Process URLs

News Insights using LLM

Question:

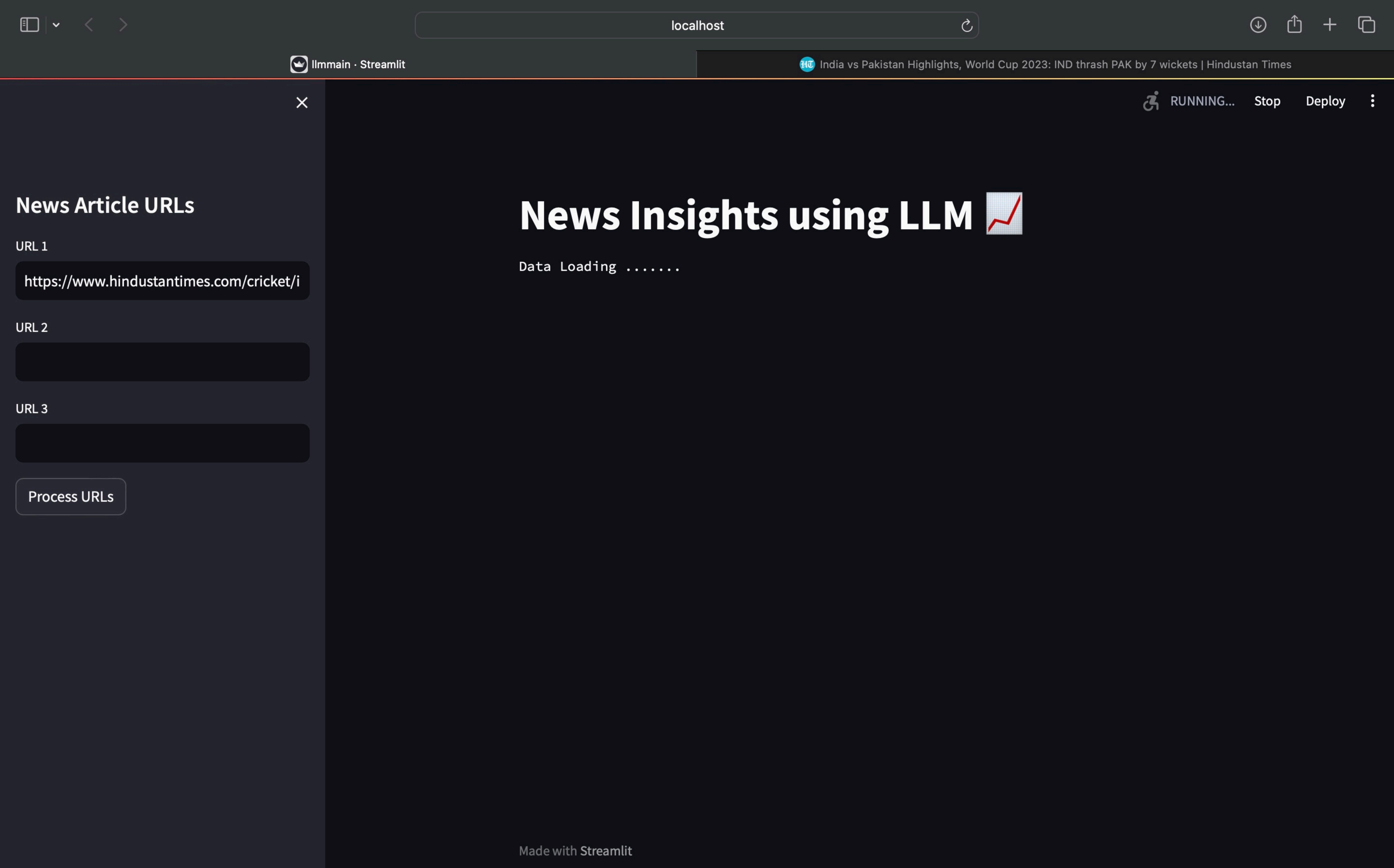
who scored the first international goal for Italy in Italy vs Malta match

Answer

Giacomo Bonaventura scored the first international goal for Italy in the Italy vs Malta match.

Sources:

<https://www.bbc.com/sport/football/67037427>



News Article URLs

URL 1

https://www.hindustantimes.com/cricket/i

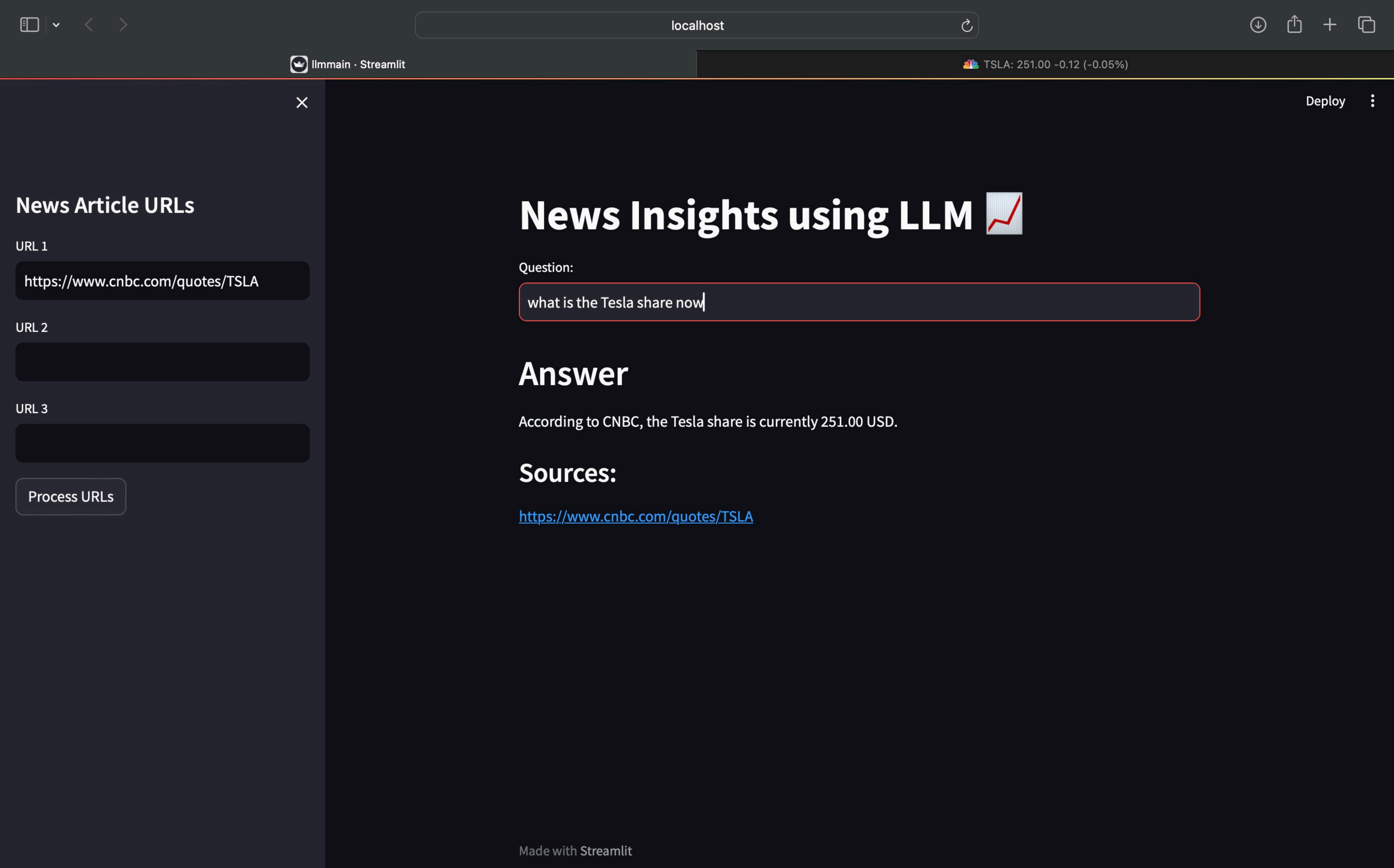
URL 2

URL 3

Process URLs

News Insights using LLM

Data Loading



News Article URLs

URL 1

https://www.cnbc.com/quotes/TSLA

URL 2

URL 3

Process URLs

News Insights using LLM

Question:

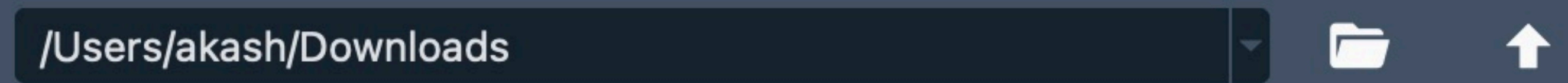
what is the Tesla share now

Answer

According to CNBC, the Tesla share is currently 251.00 USD.

Sources:

<https://www.cnbc.com/quotes/TSLA>



✕ llmmain.py*

```

1 import os
2 import streamlit as st
3 import pickle
4 import time
5 from langchain import OpenAI
6 from langchain.chains import RetrievalQAWithSourcesChain
7 from langchain.text_splitter import RecursiveCharacterTextSplitter
8 from langchain.document_loaders import UnstructuredURLLoader
9 from langchain.embeddings import OpenAIEmbeddings
10 from langchain.vectorstores import FAISS
11
12 os.environ['OPENAI_API_KEY'] = 'sk-...'
13 st.title("News Insights using LLM 📰")
14 st.sidebar.title("News Article URLs")
15 urls = []
16 for i in range(3):
17     url = st.sidebar.text_input(f"URL {i+1}")
18     urls.append(url)
19
20 process_urlclick = st.sidebar.button("Process URLs")
21 file_path = "faiss_store_openai.pkl"
22
23 main_placeholder = st.empty()
24 llm = OpenAI(temperature=0.9, max_tokens=500)
25
26 if process_urlclick:
27     loader = UnstructuredURLLoader(urls=urls)
28     main_placeholder.text("Data Loading .....")
29     data = loader.load()
30     text_splitter = RecursiveCharacterTextSplitter(
31         separators=['\n\n', '\n', '.', ','],
32         chunk_size=1000
33     )
34     main_placeholder.text("Text Splitter started .....")
35     docs = text_splitter.split_documents(data)
36     embeddings = OpenAIEmbeddings()
37     vectorstore_openai = FAISS.from_documents(docs, embeddings)
38     main_placeholder.text("Embedding Vector Started Building.....")
39     time.sleep(2)
40     with open(file_path, "wb") as f:
41         pickle.dump(vectorstore_openai, f)
42
43 query = main_placeholder.text_input("Question: ")
44 if query:
45     if os.path.exists(file_path):
46         with open(file_path, "rb") as f:
47             vectorstore = pickle.load(f)
48             chain = RetrievalQAWithSourcesChain.from_llm(llm=llm, retriever=vectorstore.as_retriever())
49             result = chain({"question": query}, return_only_outputs=True)
50             st.header("Answer")
51             st.write(result["answer"])
52             sources = result.get("sources", "")
53             if sources:
54                 st.subheader("Sources:")
55                 sources_list = sources.split("\n")
56                 for source in sources_list:
57                     st.write(source)

```

Name	Type	Size	Value
file_path	str	22	faiss
i	int	1	2
main_placeholder	delta_generator.DeltaGenerator	1	Delta
process_urlclick	bool	1	False
query	str	0	
url	str	0	
urls	list	3	['',

Help Variable Explorer Plots Files



×

Console 1/A





```
Python 3.11.5 (main, Sep 11 2023, 08:31:25) [Clang 14.0.6 ]
Type "copyright", "credits" or "license()" for more information.
```

```
IPython 8.15.0 -- An enhanced Interactive Python.
```

```
In [1]: runfile('/Users/akash/Downloads/llmmain.py', wdir='/Users/akash/Downloads')
```

2023-10-15 10:22:46.888

```
Warning: to view this Streamlit app on a browser, run it with the
following
command:
```

```
streamlit run /Users/akash/Downloads/llmmain.py [ARGUMENTS]
```

```
In [2]:
```

IPython Console History