

```
import os
from dotenv import load_dotenv

from langchain_community.llms import Ollama
import streamlit as st
from langchain_core.prompts import ChatPromptTemplate
from langchain_core.output_parsers import StrOutputParser

load_dotenv()

## Langsmith Tracking
os.environ["LANGCHAIN_API_KEY"] = os.getenv("LANGCHAIN_API_KEY")
os.environ["LANGCHAIN_TRACING_V2"] = "true"
os.environ["LANGCHAIN_PROJECT"] = os.getenv("LANGCHAIN_PROJECT")

## Prompt Template
prompt = ChatPromptTemplate.from_messages(
    [
        ("system", "You are a helpful assistant. Please respond to the question asked"),
        ("user", "Question:{question}")
    ]
)

## streamlit framework
st.title("Langchain Demo With Gemma Model")
input_text = st.text_input("What question you have in mind?")

## Ollama Llama2 model
llm = Ollama(model="gemma:2b")
output_parser = StrOutputParser()
chain = prompt | llm | output_parser

if input_text:
    st.write(chain.invoke({"question": input_text}))
```