

Recap:

- Web App/HSML API Integration
 - **Status: 75%** (*missing optional fields, some incorrect output, some bugs*)
- Latency Testing
 - **Status: 30%** (*two approaches selected, now moving to trial-and-error*)
- PhysX-to-Chaos Physics Conversion
 - **Status: 10%** (*last update: Jared running Diego's code*)

Bonus: Hosting your first LLM :)

- **Lingo:**

- **Temperature:**

- From 0.0-1.0
 - ❄️ : Very deterministic, same answer every time
 - 🔥 : Wildly imaginative, high hallucination rate, sometimes nonsensical output

- **System Prompt:**

- A series of instructions to the LLM on how to behave
 - An actor's role in a play: "you are the King of Spain! Act regal!"



- **User Prompt:**

- The user's submission to the LLM (ex. a question to the very regal King of Spain)

Lingo, Continued

- **More Lingo:**

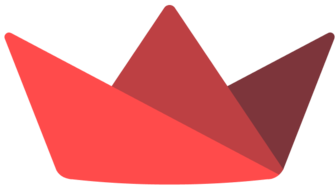
- **Parameters:**

- A rough measure of how complex the internal model is, measured in billions
 -  : More intelligence, nuance, and memory; slower response rate and higher compute
 -  : Faster, cheaper, more "civilian"; less powerful, performs worse at complex tasks

- **"Agentic" AI/Tool Use:**

- AI trained to handle complex tasks on their own through using tools meant for humans
 - ex. Salesforce's new AI customer support agents

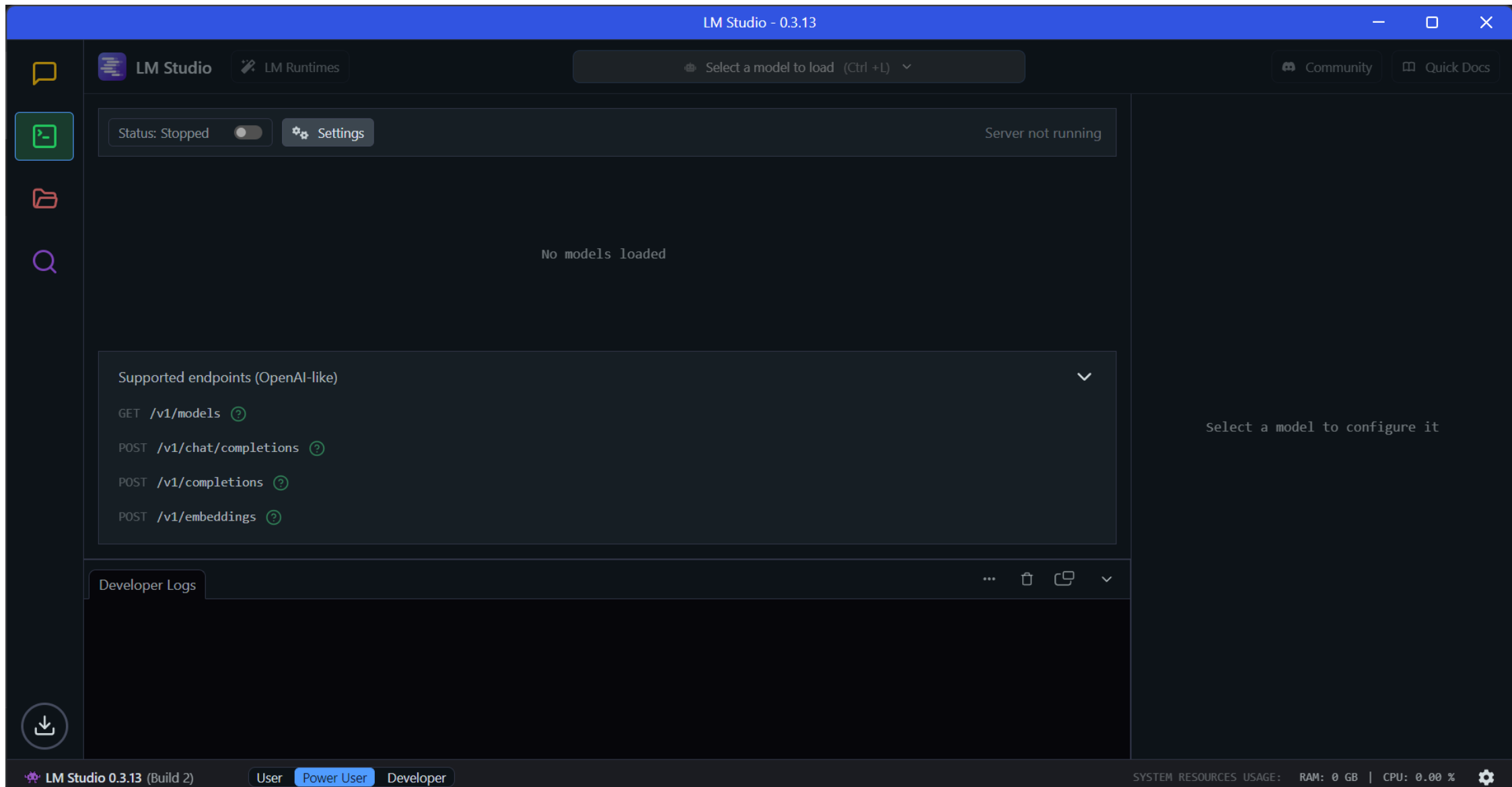
Today's Tech Stack:



Streamlit



LM Studio



Model Search

Ctrl ↑ M

Runtime

Ctrl ↑ R

Hardware

Ctrl ↑ H

App Settings

Ctrl ,

Search for models on Hugging Face...

☒ GGUF ?

Showing staff pic...

Best Match



Gemma 3 27B

State-of-the-art image + text input models f...

Gemma 3 12B

State-of-the-art image + text input models f...

Gemma 3 4B

State-of-the-art image + text input models f...

Gemma 3 1B

Tiny text-only variant of Gemma 3

QwQ 32B

Reasoning model from the Qwen family

granite 3.2 8b

A small and capable LLM from IBM

Qwen2.5 7B Instruct 1M

Powerful general purpose instruct model wit...

Gemma 3 4B



GGUF



Model Card ↗

LM Studio Staff Pick

State-of-the-art image + text input models from Google, built from the same research and tech used to create the Gemini models

Architecture:

gemma

Params:

4B

Stats:

♥ 17

↓ 90111

Last updated:

16 days ago

4 download options available ?

Q4_K_M



Gemma 3 4B Instruct



3.34 GB



Model Readme

Pulled from the model's repository



Community Model > gemma 3 4b it by Google



LM Studio Community models highlights program. Highlighting new & noteworthy models by the community. Join the conversation on [Discord](#).

Cancel

Download 3.34 GB

Chat Completions

This is an OpenAI-like call to the `/v1/chat/completion` endpoint using the `curl` utility. To run it on Mac or Linux, use any terminal. On Windows, use [Git Bash](#).

```
curl http://127.0.0.1:1234/v1/chat/completions \  
  -H "Content-Type: application/json" \  
  -d '{  
    "model": "gemma-3-4b-it",  
    "messages": [  
      { "role": "system", "content": "Always answer in rhymes." },  
      { "role": "user", "content": "Introduce yourself." }  
    ],  
    "temperature": 0.7,  
    "max_tokens": -1,  
    "stream": true  
  }'
```



create a very simple streamlit script that takes in user questions and answers back using our locally hosted gemma. here's our curl:

```
curl http://127.0.0.1:1234/v1/chat/completions \  
-H "Content-Type: application/json" \  
-d '{  
  "model": "gemma-3-27b-it",  
  "messages": [  
    { "role": "system", "content": "Always answer in rhymes." },  
    { "role": "user", "content": "Introduce yourself." }  
  ],  
  "temperature": 0.7,  
  "max_tokens": -1,  
  "stream": true  
'
```



```

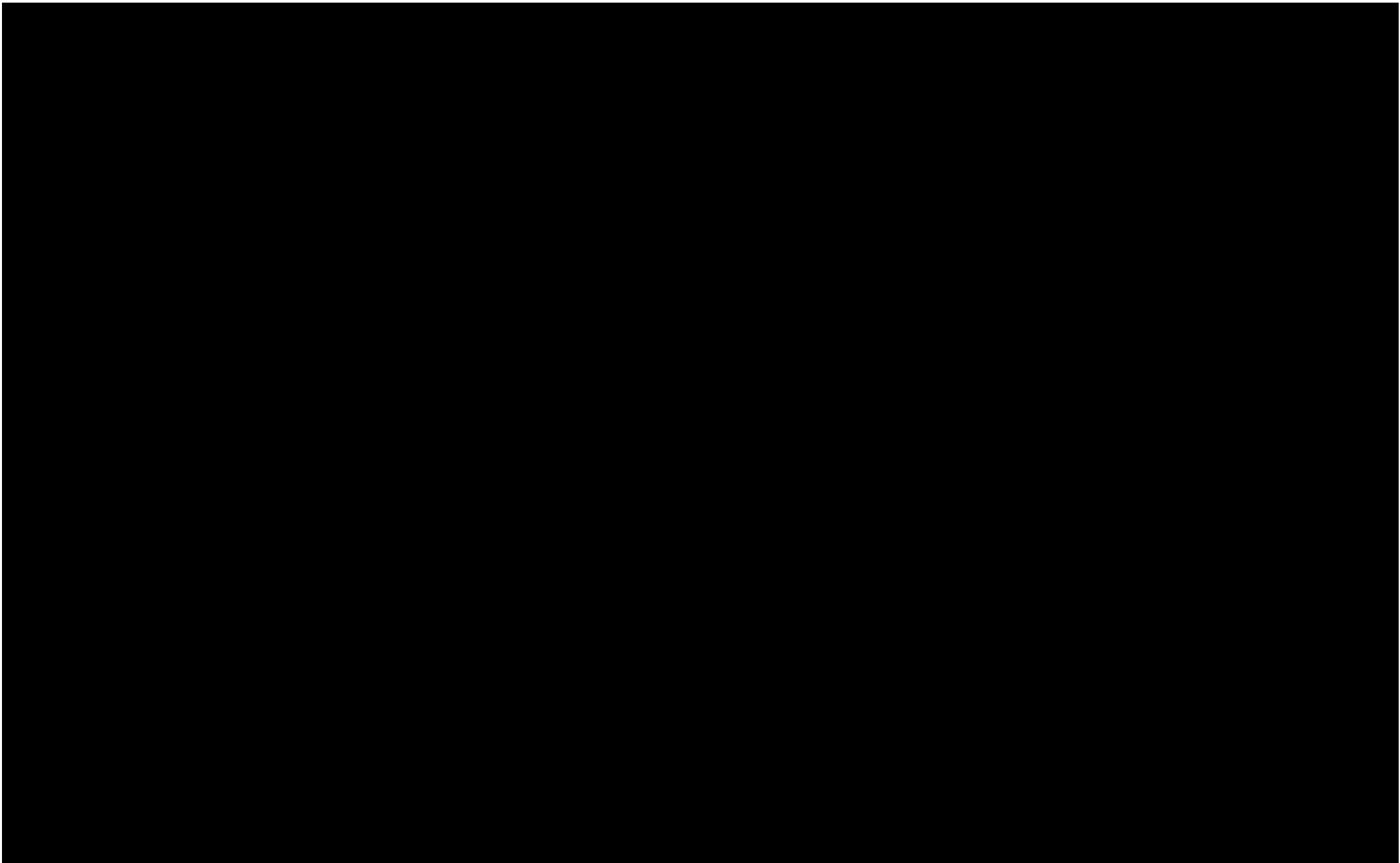
1  # gemma_chat.py
2  import streamlit as st
3  import requests
4  import json
5
6  st.title("🧙 Chat with Gemma (in Rhymes!)")
7
8  # Initialize chat history
9  if "messages" not in st.session_state:
10     st.session_state.messages = [
11         {"role": "system", "content": "Your name is Gemma, a whimsical elf from the forest. Answer in rhyme."}
12     ]
13
14 # Display chat messages from history (excluding system)
15 for msg in st.session_state.messages:
16     if msg["role"] != "system":
17         with st.chat_message(msg["role"]):
18             st.markdown(msg["content"])
19
20 # Accept user input
21 if prompt := st.chat_input("Ask Gemma something..."):
22     # Add user message to session state and display
23     st.session_state.messages.append({"role": "user", "content": prompt})
24     with st.chat_message("user"):
25         st.markdown(prompt)
26
27     # Display assistant message container
28     with st.chat_message("assistant"):
29         # Prepare payload with full chat history
30         payload = {
31             "model": "gemma-3-27b-it",
32             "messages": st.session_state.messages,
33             "temperature": 0.7,
34             "max_tokens": -1,
35             "stream": False
36         }
37
38         try:
39             response = requests.post(
40                 "http://127.0.0.1:1234/v1/chat/completions",
41                 headers={"Content-Type": "application/json"},
42                 data=json.dumps(payload)
43             )
44             response.raise_for_status()
45             content = response.json()["choices"][0]["message"]["content"]
46             st.markdown(content)
47             st.session_state.messages.append({"role": "assistant", "content": content})
48         except Exception as e:
49             st.error(f"Error getting response from Gemma: {e}")

```

```
8  # Initialize chat history
9  if "messages" not in st.session_state:
10     st.session_state.messages = [
11         {"role": "system", "content": "Your name is Gemma, a whimsical elf from the forest. Answer in rhyme."}
12     ]
```

```
27     # Display assistant message container
28     with st.chat_message("assistant"):
29         # Prepare payload with full chat history
30         payload = {
31             "model": "gemma-3-27b-it",
32             "messages": st.session_state.messages,
33             "temperature": 0.7,
34             "max_tokens": -1,
35             "stream": False
36         }
```

```
38         try:
39             response = requests.post(
40                 "http://127.0.0.1:1234/v1/chat/completions",
41                 headers={"Content-Type": "application/json"},
42                 data=json.dumps(payload)
43             )
```



```
2025-03-27 12:26:09 [INFO] [LM STUDIO SERVER] [gemma-3-4b-it] Generated prediction: {
  "id": "chatcmpl-ug8u66gl4vfwy5p6f9mkrj",
  "object": "chat.completion",
  "created": 1743103568,
  "model": "gemma-3-4b-it",
  "choices": [
    {
      "index": 0,
      "logprobs": null,
      "finish_reason": "stop",
      "message": {
        "role": "assistant",
        "content": "Why, my dear friend, it's Gemma true!\nA little elf with a joyful hue. \nI da
in the ferns and sing with the breeze,\nSo lovely to meet you, if you please! ✨"
      }
    }
  ],
  "usage": {
    "prompt_tokens": 88,
    "completion_tokens": 46,
    "total_tokens": 134
  },
  "stats": {},
  "system_fingerprint": "gemma-3-4b-it"
}
```