

[Open in app ↗](#)

Search



# MESOP, a New UI framework

A new kid on the block from Google

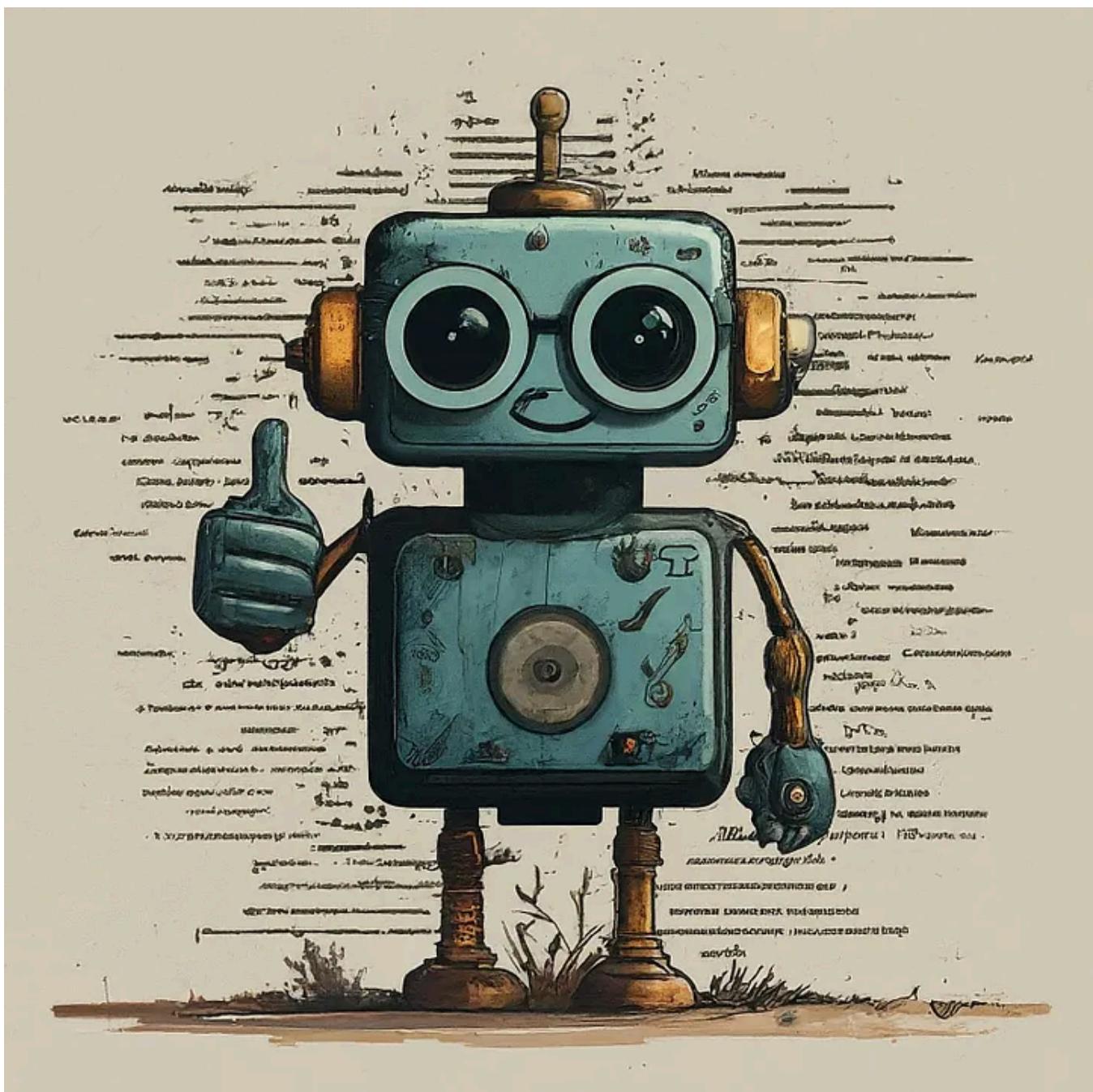


Yogesh Haribhau Kulkarni (PhD)

Published in Analytics Vidhya

4 min read · Just now

 [Listen](#) [Share](#) [More](#)



Robot Mascot ([source](#))

## Why Another UI Framework?

In the world of UI frameworks, tools like Streamlit and Gradio have already made significant impacts. They have streamlined the process of building and deploying web applications, especially for machine learning and data science projects. However, Mesop, developed by Google engineers, brings unique features and advantages to the table. Let's explore why Mesop is a valuable addition to the existing UI frameworks and what differentiates it from Streamlit and Gradio.

Building applications that use large language models (LLMs) presents unique challenges. Testing these applications and getting user feedback quickly is crucial. One solution to these challenges is Mesop, a new framework from Google designed to simplify the process of creating web UIs with Python. In this blog, we'll explore

what Mesop is, how it compares to other tools, and how you can use it to build and test your applications swiftly.

## Introduction to Mesop

Mesop is a framework developed by Google engineers during their 20% project time. It aims to make it easy for engineers, especially those with limited front-end skills, to create web UIs quickly. Mesop offers a variety of components and demos to help you get started:

1. **High-Level Components:** Pre-built elements like chat interfaces, text-to-text, and text-to-image examples.
2. **Low-Level Components:** Elements like text boxes, markdown support, buttons, and more, allowing for custom designs.
3. **Demos:** Example projects such as LLM playgrounds, showcasing how to use Mesop to create various tools.

## Key Features

1. **Rapid UI Building:** Mesop allows you to create UIs swiftly, helping you test your ideas and get feedback faster.
2. **Python-Based:** Built with Python, it integrates seamlessly into the workflow of many developers.
3. **Open Source:** Mesop is open-sourced, making it accessible to everyone and encouraging community contributions.
4. **High-Level Components:** It comes with pre-built components like chat interfaces, text inputs, and buttons.
5. **Flask Integration:** Mesop uses Flask to handle the backend, simplifying server-side operations.

## Comparison with Other Tools

1. **Streamlit:** Previously popular within Google, Streamlit has been acquired by Snowflake and is possibly less favored internally now. Mesop could be seen as Google's in-house alternative.
2. **Gradio:** Acquired by Hugging Face, Gradio is another popular tool for building UIs quickly. However, Mesop offers more control and customization options for Google engineers.

## Building a Simple Chat Bot

Let's walk through building a simple chat bot using Mesop, LangChain, and Groq.

1. **Set Up Mesop:** Start by importing the necessary packages.

```
import Mesop as me
import Mesop.labs as mel
```

## 2. Define the Chat Function: Create a function to handle chat interactions.

```
def transform(user_input, chat_history):
    # Combine history into a single string
    history_string = "\n".join([f"{msg['role']}: {msg['content']}" for msg in c
    response = run_conversation_chain(user_input, history_string)
    return response
```

## 3. Initialize the Chat Interface: Use Mesop's components to create the chat interface.

```
me.app.chat(transform)
```

### Running the Application

You can run your Mesop application locally or in a cloud environment like Google Colab. Mesop provides a streamlined process for deploying your apps, making it easy to share and test with others.

### Example: Chat Bot with Memory

Here's a more detailed example of setting up a chat bot with memory using Mesop and LangChain.

#### 1. Initialize Packages:

```
import Mesop as me
import Mesop.labs as mel
from langchain import ConversationChain
from groq import Groq
```

## 2. Set Up the API and Prompts:

Here we are using Large Language Model using groq

```
groq_api_key = "your_api_key"
groq_model = Groq(api_key=groq_api_key, model="Llama 3 70B")

def transform(user_input, chat_history):
    history_string = "\n".join([f"{msg['role']}: {msg['content']}" for msg in ch
    response = groq_model.generate(user_input, history_string)
    return response
```

But we can also try models from Hugging Face

```
from langchain_huggingface import HuggingFaceEndpoint

# Replace "your_access_token" with your Hugging Face access token
HUGGINGFACEHUB_API_TOKEN = "your_access_token"

# Define the Hugging Face model ID (e.g., meta-llama/Meta-Llama-3-70B-Instruct)
model_id = "meta-llama/Meta-Llama-3-70B-Instruct"

def transform(user_input, chat_history):
    history_string = "\n".join([f"{msg['role']}: {msg['content']}" for msg in cha
    # Instantiate Hugging Face Endpoint
    llm = HuggingFaceEndpoint(
        repo_id=model_id,
        task="text-generation",
        max_new_tokens=512 # Maximum number of new tokens to generate
    )

    # Generate response using user input and chat history
    response = llm.generate(history_string + "\n" + user_input)
    return response
```

**Exercise:** Can you modify the code above to use Google's gemma in place of Meta's Llama?

## 3. Create and Run the Chat Interface:

```
me.app.chat(transform)
```

## Future Applications

Mesop is versatile and can be used for various applications beyond chat bots:

1. **Text-to-Image Tools:** Build tools for image manipulation.
2. **Interactive Demos:** Create interactive demos for your LLM models.
3. **Testing Platforms:** Develop quick prototypes to gather user feedback.

## Conclusion

Mesop is a powerful tool for developers looking to build and test applications quickly. Its Python-based approach, high-level components, and open-source nature make it a valuable addition to your toolkit. Whether you're building a chat bot or a complex UI, Mesop simplifies the process, allowing you to focus on your application's functionality.

Explore Mesop today and see how it can enhance your development workflow. More info below:

### **GitHub - google/mesop: Build delightful web apps quickly in Python**

Build delightful web apps quickly in Python. Contribute to google/mesop development by creating an account on GitHub.

[github.com](https://github.com/google/mesop)

*Click pic below or visit [LinkedIn](#) to know more about the author*



Artificial Intelligence

Chatbots

Google

Mesop

UI



Edit profile

## Written by Yogesh Haribhau Kulkarni (PhD)

1.6K Followers · Writer for Analytics Vidhya

PhD in Geometric Modeling | Google Developer Expert (Machine Learning) | Top Writer 3x (Medium) | More at <https://www.linkedin.com/in/yogeshkulkarni/>

### More from Yogesh Haribhau Kulkarni (PhD) and Analytics Vidhya



Yogesh Haribhau Kulkarni (PhD) in ILLUMINATION

## Unraveling the Secrets of the Mind

A Tour of Mental Models

5 min read · Jun 3, 2024

250

5

...



Yogesh Haribhau Kulkarni (PhD) in Technology Hits

## Specs for 'Chatbot on Resumes'

Product Requirements Document with hints of Implementation

8 min read · May 17, 2024

14

...



Yogesh Haribhau Kulkarni (PhD) in ILLUMINATION Videos and Podcasts

## 5 Fronts of the AI-War

Based on the TEDx IIITA talk by Rajiv Malhotra

3 min read · 3 days ago



...



Yogesh Haribhau Kulkarni (PhD) in ILLUMINATION Videos and Podcasts

## The Equation of True Happiness

Based on talks by Arthur C Brooks

4 min read · Sep 17, 2023

179

1

+

...

See all from Yogesh Haribhau Kulkarni (PhD)

See all from Analytics Vidhya

## Recommended from Medium



Jennifer Reif in Neo4j Developer Blog

### GenAI Starter Kit: Everything You Need to Build an Application with Spring AI in Java

Use Spring AI to extend the richness of the Spring ecosystem and write a GenAI app in Java

7 min read · Jun 11, 2024

6

1

+

...



 Almog Baku in Towards Data Science

## Building LLM Apps: A Clear Step-By-Step Guide

Comprehensive Steps for Building LLM-Native Apps: From Initial Idea to Experimentation, Evaluation, and Productization

12 min read · Jun 10, 2024

 5.5K

 13



...

### Lists



#### AI Regulation

6 stories · 482 saves



#### Generative AI Recommended Reading

52 stories · 1135 saves



#### ChatGPT

21 stories · 678 saves



#### Natural Language Processing

1517 stories · 1050 saves



 Malcolm Frank

## Missing the boat? Why IT Services leaders are struggling in the AI boom.

AI was hyped as the golden ticket for IT services, promising explosive revenue growth and soaring valuations. However, the reality on the...

4 min read · 4 days ago

 137  8



...

[amazon.com](#)

*Software Development Engineer*

Seattle, WA

Mar. 2020 – May 2021

- Developed Amazon checkout and payment services to handle traffic of 10 Million daily global transactions
- Integrated Iframes for credit cards and bank accounts to secure 80% of all consumer traffic and prevent CSRF, cross-site scripting, and cookie-jacking
- Led Your Transactions implementation for JavaScript front-end framework to showcase consumer transactions and reduce call center costs by \$25 Million
- Recovered Saudi Arabia checkout failure impacting 4000+ customers due to incorrect GET form redirection

### Projects

**NinjaPrep.io (React)**

- Platform to offer coding problem practice with built in code editor and written + video solutions in React
- Utilized Nginx to reverse proxy IP address on Digital Ocean hosts
- Developed using Styled-Components for 95% CSS styling to ensure proper CSS scoping
- Implemented Docker with Seccomp to safely run user submitted code with < 2.2s runtime

**HeatMap (JavaScript)**

- Visualized Google Takeout location data of location history using Google Maps API and Google Maps heatmap code with React
- Included local file system storage to reliably handle 5mb of location history data
- Implemented Express to include routing between pages and jQuery to parse Google Map and implement heatmap overlay



Alexander Nguyen in Level Up Coding

## The resume that got a software engineer a \$300,000 job at Google.

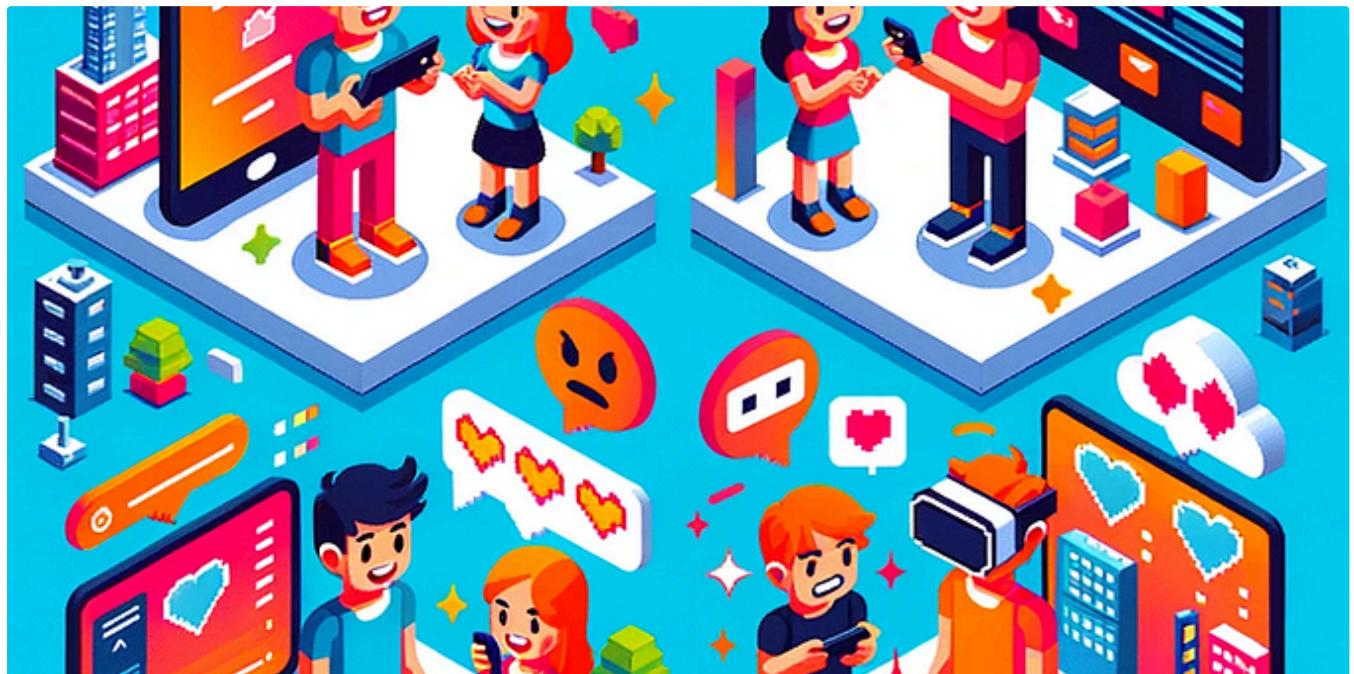
1-page. Well-formatted.

◆ · 4 min read · Jun 1, 2024

👏 6.4K 🗣 56



...



Andrew Tipp in UX Planet

## Emotional design: why it matters in UX

What does designing for emotion mean, and how can you do it?

9 min read · 6 days ago

👏 111 🗣 1



...

## Programming—not prompting—Language Models

[Get Started with DSPy](#)

### The Way of DSPy



Vishal Rajput in AI Guys

## Prompt Engineering Is Dead: DSPy Is New Paradigm For Prompting

DSPy Paradigm: Let's program—not prompt—LLMs

★ · 11 min read · May 29, 2024

3K

30

+

...

[See more recommendations](#)