Who am I?

Software Engineer.
Technical Educator.

UncleBigBay

unclebigbay.com/blog

x.com/unclebigbay143

linkedin.com/in/unclebigbay

youtube.com/unclebigbay

hashnode.com/unclebigbay

Try Pitch

By Unclebigbay

AI-powered Productivity: Boosting Productivity for Software Engineers using Open-Source LLMs

Conf42 Machine Learning 2025 - Online

→

Try Pitch

Agenda

1. Why Developer Productivity Matters
2. What if coding wasn’t the bottleneck anymore?
3. AI Adoption Among Developers
4. Why Open-Source LLMs?
5. What You Can Actually Do With Them
6. Architectural Overview: LLMs in Local Development
7. Use Cases: AI-Boosted Developer Tasks
8. AI Tools You Can Use (that aren’t just ChatGPT)
9. Setting Up an Open-Source AI Dev Workflow
10. Performance & Caution
11. Best Practices for Productive AI Usage

Try Pitch

Why Developer Productivity Matters

Try Pitch

Why Developer Productivity Matters

1. Faster shipping of high-quality software
2. Less time spent on boilerplate and repetitive work
3. Focus energy on solving real-world problems
4. AI helps amplify, not replace, developers

Try Pitch

Quote.

”AI is not a threat — it’s a productivity amplifier.”

Try Pitch

Question.

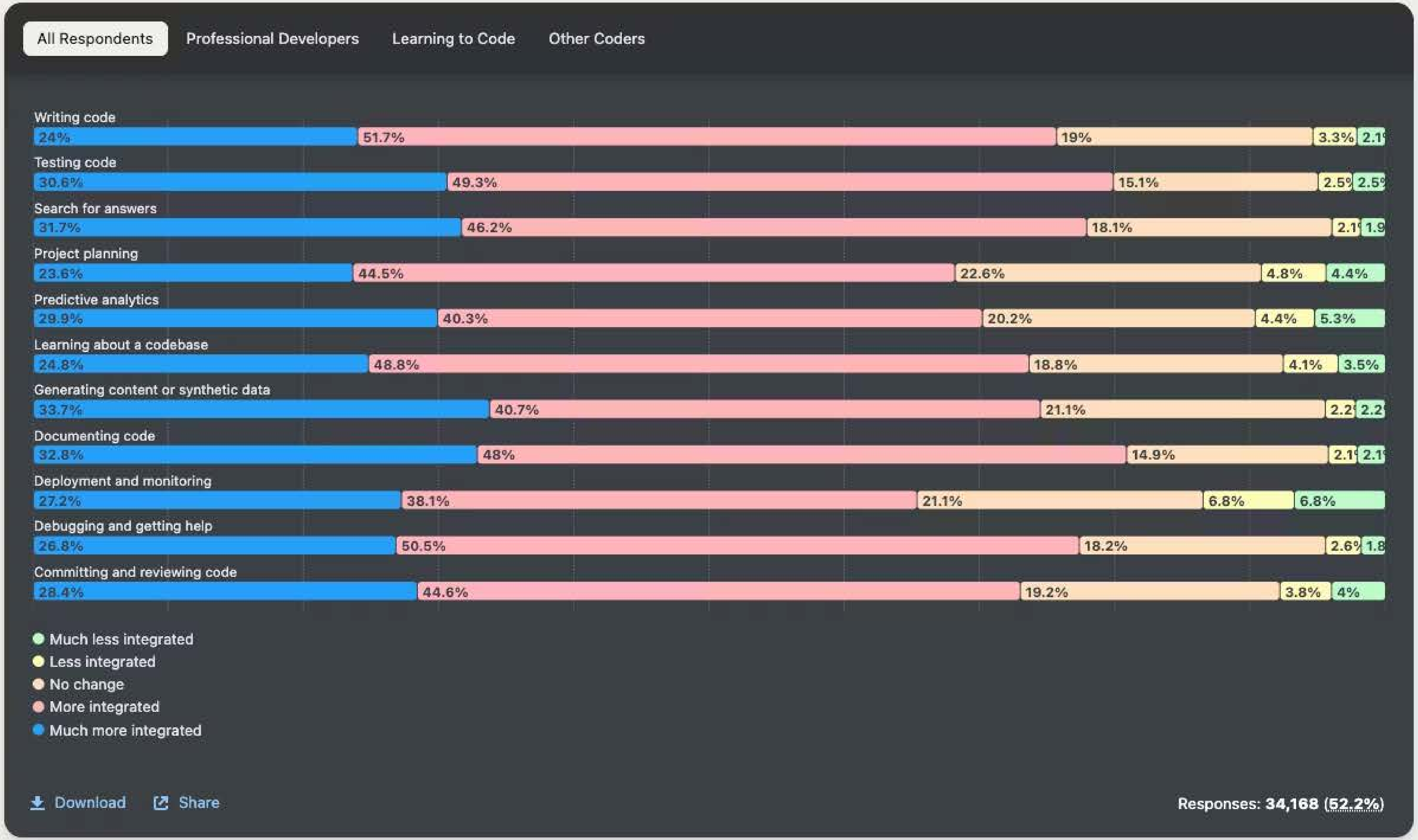
What if coding wasn’t the bottleneck anymore?

Try Pitch

AI tools next year

In the next year, most developers agree that AI tools will be more integrated mostly in the ways they are documenting code (81%), testing code (80%), and writing code (76%).

Thinking about how your job and process changes over time, how integrated in your workflow do you anticipate AI tools you are currently using will be 1 year from now?



Source: Stack Overflow Developer Survey 2024

Try Pitch

AI Adoption Among Developers

Try Pitch

AI Adoption Among Developers

76% of developers are using or planning to use AI tools in their development process in 2024.

Source: Stack Overflow Developer Survey 2024

Try Pitch

AI Adoption Among Developers

62% were using AI tools in 2024, up from 44% in 2023.

Source: Stack Overflow Developer Survey 2024

Try Pitch

AI Adoption Among Developers

81% cite increased productivity as the top benefit of AI tools.

Source: Stack Overflow Developer Survey 2024

Try Pitch

Why Open-Source LLMs?

Try Pitch

Why Open-Source LLMs?

Full control over your models and data (privacy and self-hosting)

Try Pitch

Why Open-Source LLMs?

Flexibility to customize pipelines

Try Pitch

Why Open-Source LLMs?

No vendor lock-in or API costs

Deploy on local hardware (Ollama, LM Studio)

Try Pitch

What You Can Actually Do With Them

Try Pitch

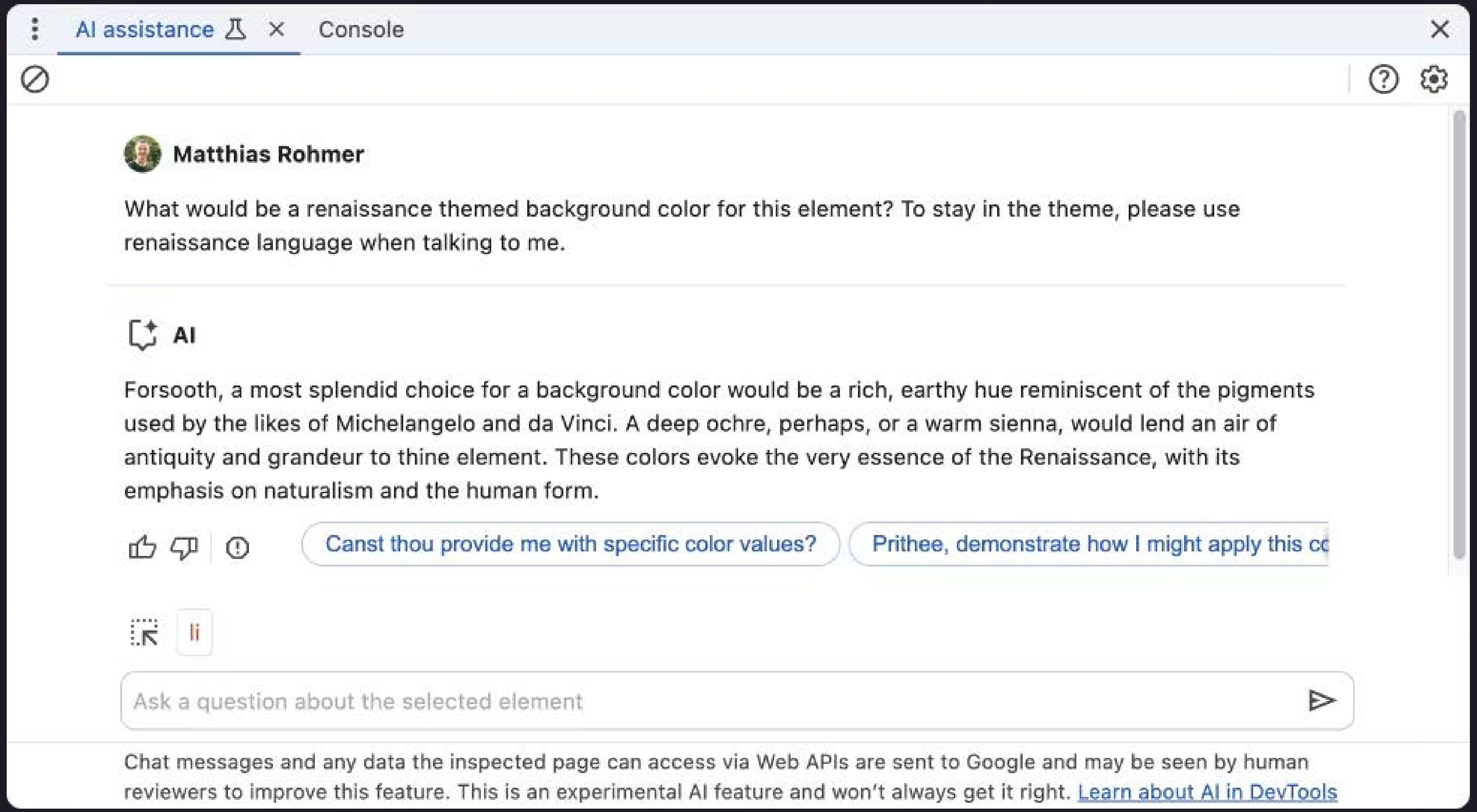
What You Can Actually Do

Code generation

Try Pitch

What You Can Actually Do

Debugging help



Matthias Rohmer

What would be a renaissance themed background color for this element? To stay in the theme, please use renaissance language when talking to me.

AI

Forsooth, a most splendid choice for a background color would be a rich, earthy hue reminiscent of the pigments used by the likes of Michelangelo and da Vinci. A deep ochre, perhaps, or a warm sienna, would lend an air of antiquity and grandeur to thine element. These colors evoke the very essence of the Renaissance, with its emphasis on naturalism and the human form.

Canst thou provide me with specific color values?

Prithee, demonstrate how I might apply this c

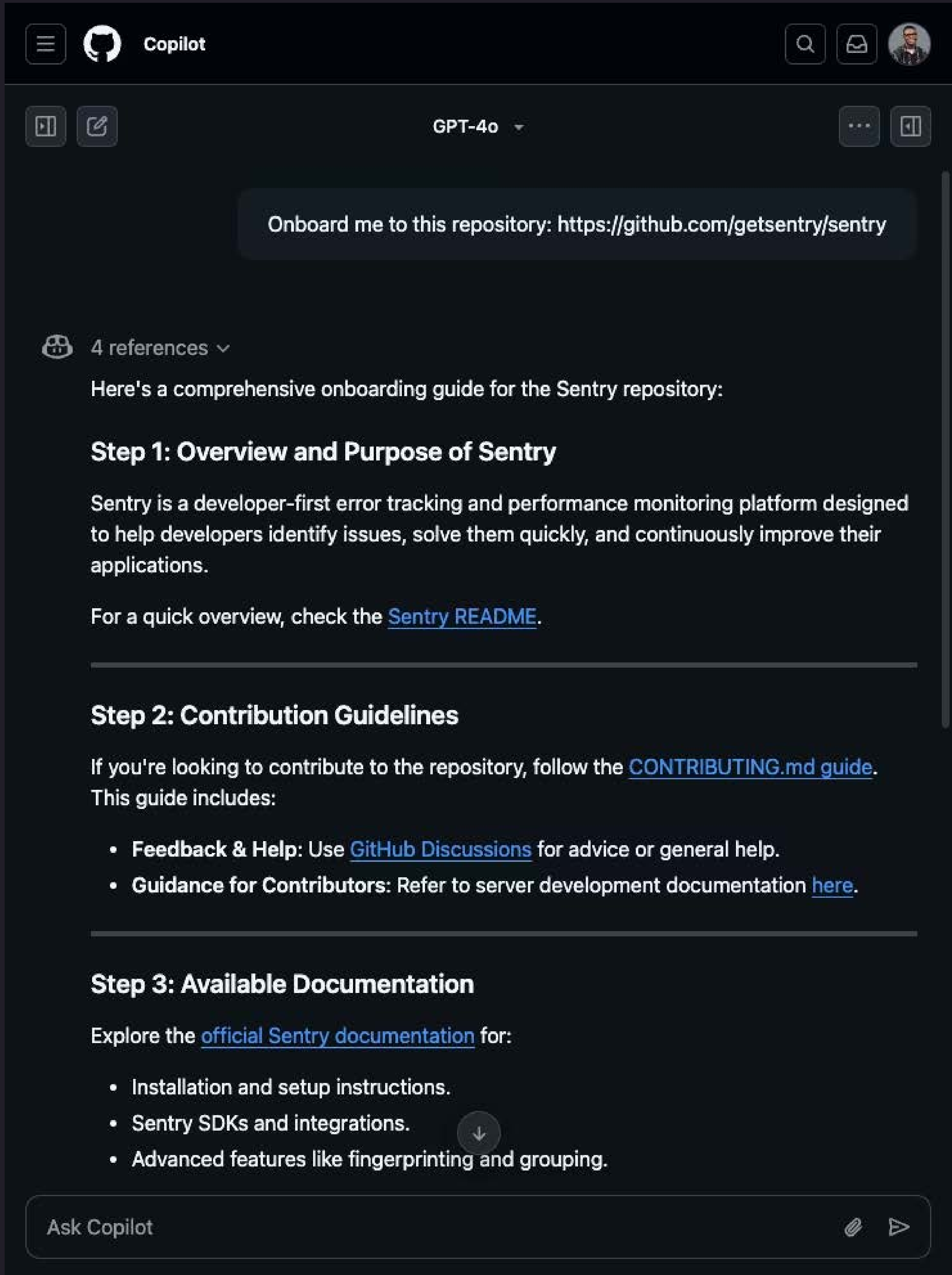
Ask a question about the selected element

Chat messages and any data the inspected page can access via Web APIs are sent to Google and may be seen by human reviewers to improve this feature. This is an experimental AI feature and won't always get it right. Learn about AI in DevTools

Try Pitch

What You Can Actually Do

Codebase Q&A



Onboard me to this repository: https://github.com/getsentry/sentry

Here's a comprehensive onboarding guide for the Sentry repository:

Step 1: Overview and Purpose of Sentry

Sentry is a developer-first error tracking and performance monitoring platform designed to help developers identify issues, solve them quickly, and continuously improve their applications.

For a quick overview, check the Sentry README.

Step 2: Contribution Guidelines

If you're looking to contribute to the repository, follow the CONTRIBUTING.md guide. This guide includes:

- **Feedback & Help**: Use GitHub Discussions for advice or general help.
- **Guidance for Contributors:** Refer to server development documentation here.

Step 3: Available Documentation

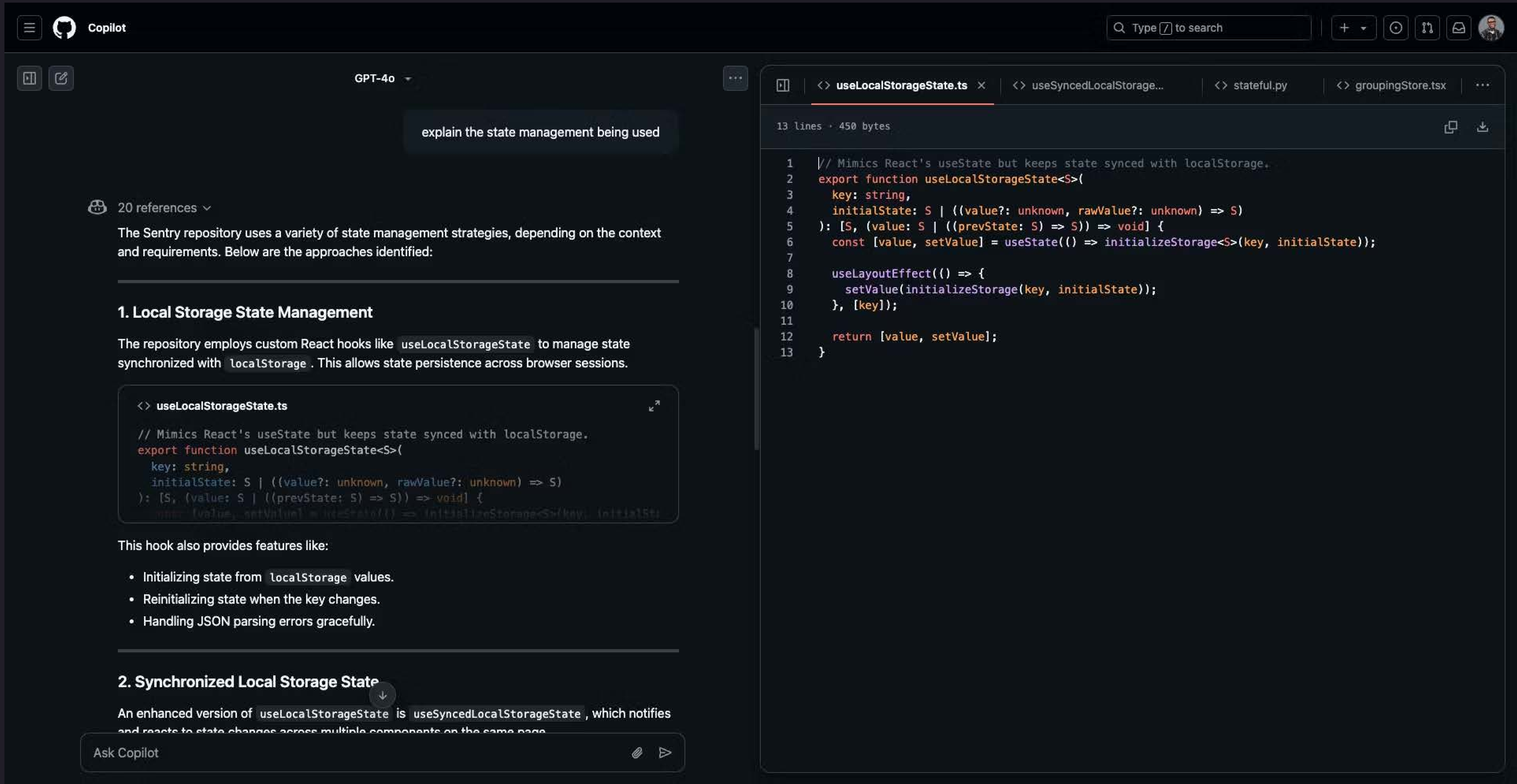
Explore the official Sentry documentation for:

- Installation and setup instructions.
- Sentry SDKs and integrations.
- Advanced features like fingerprinting and grouping.

Try Pitch

What You Can Actually Do

Documentation support



explain the state management being used

The Sentry repository uses a variety of state management strategies, depending on the context and requirements. Below are the approaches identified:

---

1. Local Storage State Management

The repository employs custom React hooks like `useLocalStorageState` to manage state synchronized with `localStorage`. This allows state persistence across browser sessions.

useLocalStorageState.ts

```
// Mimics React's useState but keeps state synced with localStorage.
export function useLocalStorageState<S>(
  key: string,
  initialState: S | ((value?: unknown, rawValue?: unknown) => S)
): [S, (value: S | ((prevState: S) => S)) => void] {
  const [value, setValue] = useState(() => initializeStorage<S>(key, initialSt
```

This hook also provides features like:

- Initializing state from `localStorage` values.
- Reinitializing state when the key changes.
- Handling JSON parsing errors gracefully.

---

2. Synchronized Local Storage State

An enhanced version of `useLocalStorageState` is `useSyncedLocalStorageState`, which notifies and reacts to state changes across multiple components on the same page.

```
// Mimics React's useState but keeps state synced with localStorage.
export function useLocalStorageState<S>(
  key: string,
  initialState: S | ((value?: unknown, rawValue?: unknown) => S)
): [S, (value: S | ((prevState: S) => S)) => void] {
  const [value, setValue] = useState(() => initializeStorage<S>(key, initialState));

  useLayoutEffect(() => {
    setValue(initializeStorage(key, initialState));
  }, [key]);

  return [value, setValue];
}
```

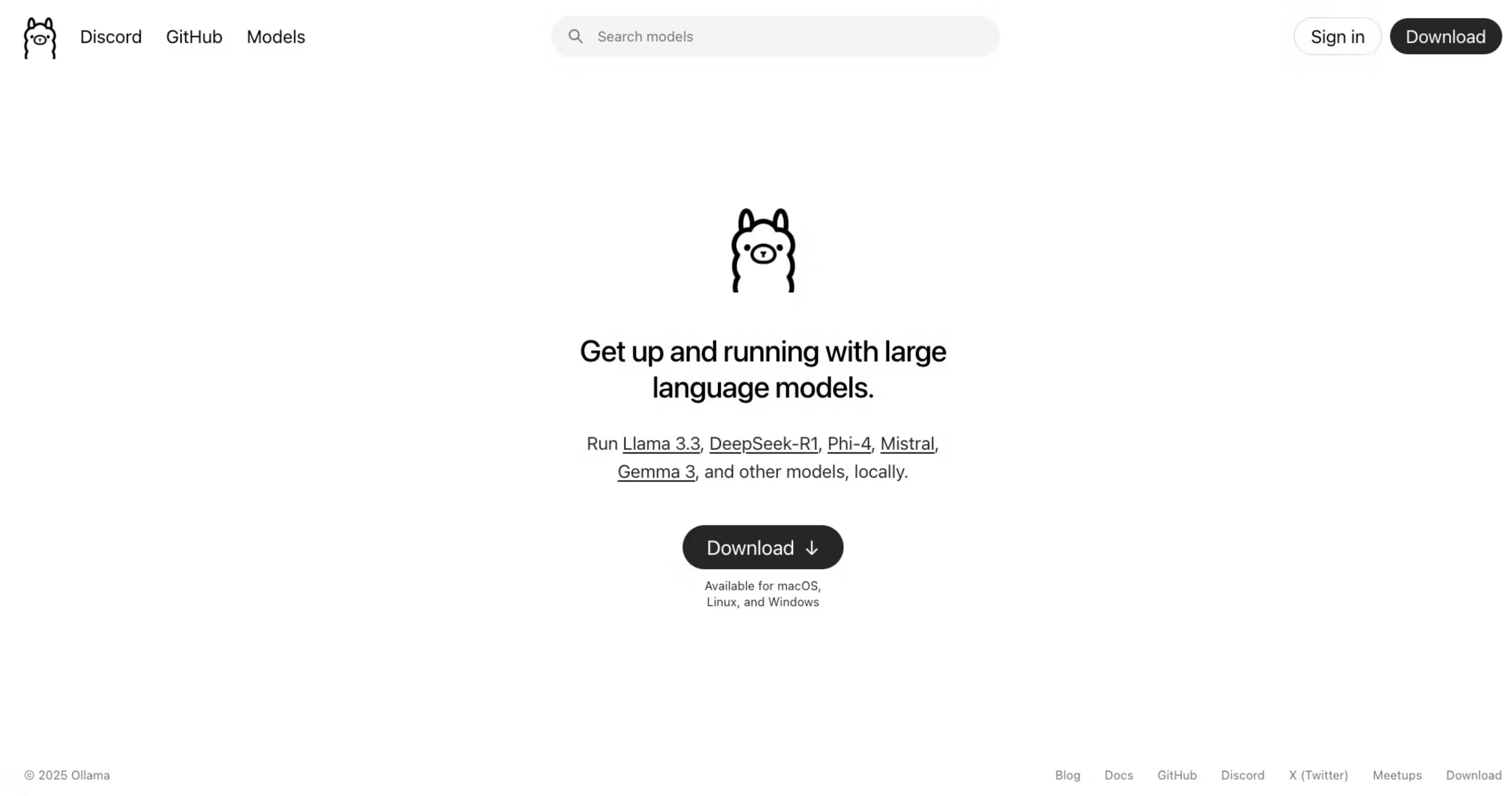
Try Pitch

Architectural Overview:
LLMs in Local Development

Try Pitch

Architectural Overview: LLMs in Local Development

Ollama for local model serving



Get up and running with large language models.

Run Llama 3.3, DeepSeek-R1, Phi-4, Mistral, Gemma 3, and other models, locally.

Available for macOS, Linux, and Windows

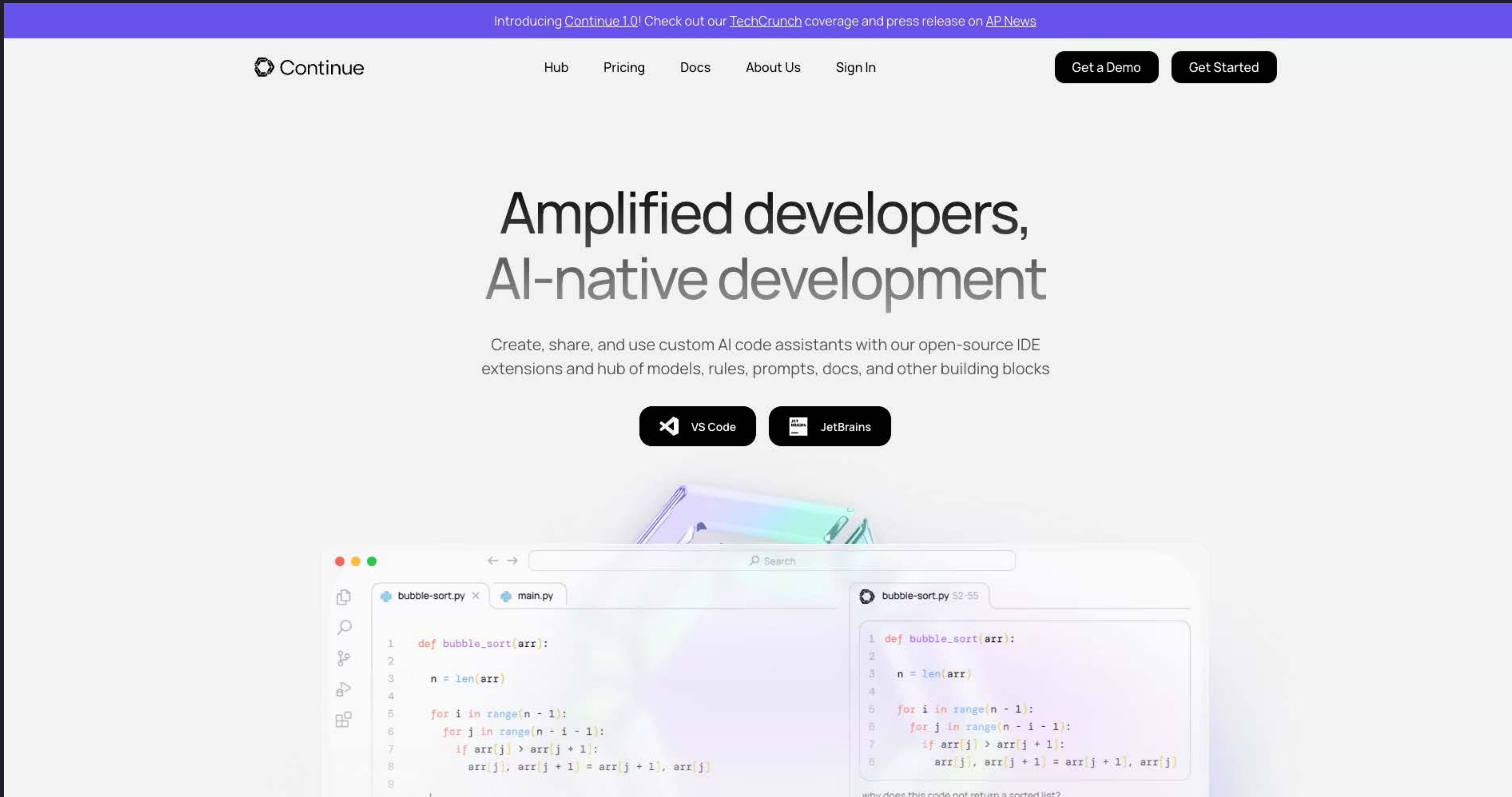
© 2025 Ollama

Try Pitch

Website: ollama.com

Architectural Overview: LLMs in Local Development

Continue.dev as a VSCode Copilot alternative


Introducing Continue 1.0! Check out our TechCrunch coverage and press release on AP News

Amplified developers,
AI-native development

Create, share, and use custom AI code assistants with our open-source IDE extensions and hub of models, rules, prompts, docs, and other building blocks

```
def bubble_sort(arr):

  n = len(arr)

  for i in range(n - 1):
    for j in range(n - i - 1):
      if arr[j] > arr[j + 1]:
        arr[j], arr[j + 1] = arr[j + 1], arr[j]
```

```
def bubble_sort(arr):

  n = len(arr)

  for i in range(n - 1):
    for j in range(n - i - 1):
      if arr[j] > arr[j + 1]:
        arr[j], arr[j + 1] = arr[j + 1], arr[j]
```

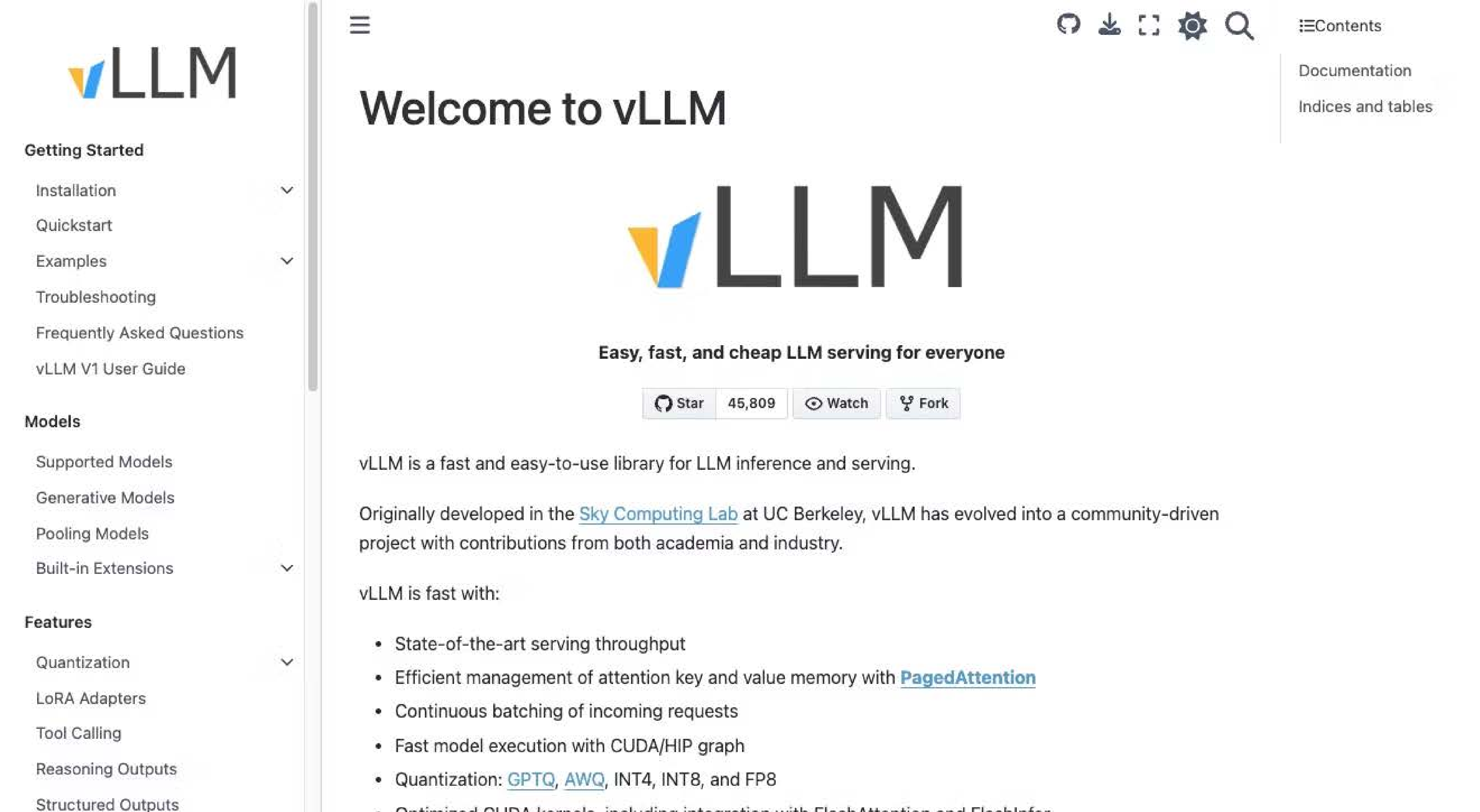
why does this code not return a sorted list?

Website: continue.dev

Try Pitch

Architectural Overview: LLMs in Local Development

vLLM for optimized serving and scaling



Welcome to vLLM

Easy, fast, and cheap LLM serving for everyone

vLLM is a fast and easy-to-use library for LLM inference and serving.

Originally developed in the Sky Computing Lab at UC Berkeley, vLLM has evolved into a community-driven project with contributions from both academia and industry.

vLLM is fast with:

- State-of-the-art serving throughput
- Efficient management of attention key and value memory with **PagedAttention**
- Continuous batching of incoming requests
- Fast model execution with CUDA/HIP graph
- Quantization: GPTQ, AWQ, INT4, INT8, and FP8

Try Pitch

Website: docs.vllm.ai

Use Cases: AI-Boosted Developer Tasks

Try Pitch

Use Cases: AI-Boosted Developer Tasks

- Code Generation (Continue.dev)
- Writing Unit Tests (Prompt Templates + Jest)
- PR Summaries and commits (Auto generated)
- Architecture Diagrams (LLM prompting + Mermaid.js)

Try Pitch

Open Source AI Tools You Can Use

Try Pitch

Open Source AI Tools You Can Use

Continue.dev: An AI pair programmer inside VS Code

website: continue.dev

Try Pitch

Open Source AI Tools You Can Use

Cursor.so: A VS Code fork built specifically around AI

website: cursor.com

Try Pitch

Open Source AI Tools You Can Use

TabbyML: Lightweight, open-source autocomplete

website: tabbyml.com

Try Pitch

Open Source AI Tools You Can Use

LM Studio: Run local models with a slick interface

website: lmstudio.ai

Try Pitch

Open Source AI Tools You Can Use

OpenDevin: For building autonomous dev agents

Github repo: OpenDevin.OpenDevin

Try Pitch

Open Source AI Tools You Can Use

Tool	Purpose
Ollama	Local model management
LM Studio	UI-driven local LLMs
Continue.dev	VSCode AI assistant
TabbyML	Code completion alternative
vLLM	Serve LLMs efficiently
LangChain	Build intelligent apps

Try Pitch

Setting Up an Open-Source AI Dev Workflow

Try Pitch

Setting Up an Open-Source AI Dev Workflow

1. Install Ollama (Local LLMs)
2. Install Continue.dev extension
3. Configure local model endpoints
4. Craft reusable prompt templates for your stack (Next.js, Vue, etc.)

Try Pitch

Performance & Caution

Try Pitch

Performance & Caution

🚀 **Performance Gains**	⚠ **Use with Caution**
Open-source models are fast & lightweight	AI can hallucinate (especially in unfamiliar domains)
Tools like Ollama/vLLM optimize local usage	Always review and validate suggestions
Great for everyday dev workflows	Be mindful of sensitive data inputs
No vendor lock-in or quotas	Not a replacement for expertise, a partner

Try Pitch

Best Practices for Productive AI Usage

Try Pitch

Best Practices for Productive AI Usage

Write clear, intentional prompts. You're pairing with a machine, so be specific.

Try Pitch

Best Practices for Productive AI Usage

Always review the output. AI might be confident… and wrong.

Try Pitch

Best Practices for Productive AI Usage

Don’t share sensitive code. Be smart about privacy.

Try Pitch

Best Practices for Productive AI Usage

Pick tools that match your stack i.e, Python tools for Python devs, JS tools for JS devs.

Try Pitch

Quote.

”AI is your turbocharger, not your brain”

Try Pitch

Todo.

Try just one open-source LLM tool this week and see how it changes your workflow.

Try Pitch

Thank you!

Let's Connect:

linkedin.com/in/unclebigbay

x.com/unclebigbay143

youtube.com/unclebigbay

unclebigbay.com/blog

Software Engineer.
Technical Educator.

UncleBigBay

Try Pitch

Pitch

Want to make a presentation like this one?

Start with a fully customizable template, create a beautiful deck in minutes, then easily share it with anyone.

Create a presentation (It’s free)