

[

1024x573 719 KB

](https://collective.flashbots.net/uploads/default/original/2X/a/aa22bc82146b4c5e6aea993efeb5e684ddf61091.png)

Summary

This post outlines the launch and capabilities of [mev.fyi](#), a novel open-source research chatbot that focuses on the domain of MEV and blockchain research.

Developed with the support of a Flashbots grant, mev.fyi is a full-stack application that employs a Retrieval Augmented Generation (RAG) model to provide users with accurate and detailed insights into MEV-related topics.

mev.fyi is designed to serve both newcomers and veterans in the blockchain space by offering immediate access to a wealth of information on MEV. By integrating directly with sources such as [Ethereum.org](#), academic papers, articles, and videos, it ensures users receive comprehensive responses to their inquiries.

Accessible through its website and a dedicated Twitter bot ([@mevfyi](#)), the platform simplifies the process of engaging with complex MEV concepts. Users can prompt the system with questions by using specific commands, such as `explain` `tweet|thread`

, e.g. “`@mevfyi`

`explain tweet`, what are the implications of [EIP-7623] as a protocol developer?” as shown [here](#).

(PMF has been achieved internally)

With the successful completion of the Flashbots grant on February 2024, mev.fyi stands as a testament to collaborative innovation in blockchain research. It invites the community to explore its functionalities and contribute, furthering the collective understanding of MEV dynamics.

The purpose behind mev.fyi

The realm of blockchain research spans a vast array of topics, from cryptography and auctions to the intricacies of data structures and algorithms. The rapid pace of development in this field often leaves both [newcomers](#) and seasoned professionals struggling to stay informed.

Recognizing this challenge, and the significant investment of time required to curate and index knowledge, mev.fyi was conceived. Inspired by the comprehensive documentation efforts like those seen on [Ethereum.org](#), the vision for mev.fyi was to create a chatbot that could provide easy access to a continuously updated broad spectrum of resources, including documentation, forums, academic papers, articles, and YouTube videos.

The enthusiastic support from the community has been pivotal in turning this vision into a reality. I extend my deepest gratitude to [Flashbots](#) and all the contributors who have joined us [on this journey](#) aiming at making mev.fyi a cornerstone for blockchain education and research, with a special mention of [Fred](#)'s championing efforts.

How mev.fyi works

Architecture

At its core, mev.fyi utilizes Retrieval Augmented Generation (RAG) technology. This framework empowers mev.fyi's large language model (LLM) to process and respond to queries by accessing a comprehensive database. This process ensures the delivery of accurate and contextually rich answers, enhancing the user experience. Unlike traditional LLM interfaces, mev.fyi enhances the user experience by directly linking to the sources cited in responses, facilitating deeper research and understanding.

On the mev.fyi front-end, users can ask MEV-related questions, and get a reply in two parts:

1 the text response answering the user prompt

2 the list of sources used as the baseline for the response which the user can explore further.

Diagram: user interface

The image below provides an overview of the user interface for mev.fyi, showcasing features such as the chat interface, and

the list of Top Sources.

[

1600×1062 271 KB

](https://collective.flashbots.net/uploads/default/original/2X/7/73a4ea177659a5b5084945a4fc3d201fbafbb1c9.jpeg)

Diagram: data ingestion

Find the systems architecture overview for data ingestion as follows:

[

1600×1197 161 KB

](https://collective.flashbots.net/uploads/default/original/2X/8/883a136f99fa864b331cb18a962a64737a3579fa.jpeg)

Diagram: mev.fyi frontend chatbot & Twitterbot

The images below depict the system architecture diagrams for (1) the mev.fyi frontend chatbot and (2) the Twitter bot:

[

|742.6230928468063x849.8903733686519

1396×1600 145 KB

](https://collective.flashbots.net/uploads/default/original/2X/4/41c7700ccbcab13ed374b9ff6684872702a135e4.jpeg)

[

|706.695652173913x640.4280155642024

1600×1452 154 KB

](https://collective.flashbots.net/uploads/default/original/2X/6/6845d47c71868e3679ff4ca4a72c87d7d2d398f3.jpeg)

Use-cases

There are several use cases from specific domain questions to explaining Twitter threads, to returning sources on a given topic:

1. Answer: novice-questions like [“What is MEV?”](#), latest research questions like [“What are execution tickets?”](#), or latest use-cases like [“What are the practical implications of data blobs and proto-danksharding \(EIP-4844\)?”](#)
2. Explain Twitter tweets or entire threads: ask for a tweet explanation [about intents applications](#), ask for a thread explanation of the importance of having [consumer hardware validators for ZK-proof verification](#) (and ASIC for zk-proof generation)
3. Asking for sources about a given topic: give me sources [about Liquid Staking Tokens \(LSTs\)](#)

But mev.fyi is capable of so much more and these use-cases are only scratching the surface of what’s possible - I can’t wait for you to share all the questions and applications you could come up with!

The close collaboration with the Flashbots team has been a delight. There has been a constant delivery from both sides, no matter how far from the original scope we got. Everything has been async, from our first Twitter conversation to the grant formalisation to the months of working together. The level of professionalism makes it an incredible experience. I strongly encourage anybody to work with Flashbots.

[

1600×400 937 KB

](https://collective.flashbots.net/uploads/default/original/2X/9/99c0947bb7227b0d0d73226234b8d0a3557bcad5.png)

Looking ahead: development plans following the grant’s conclusion

This grant concluded on 2024-02 following the delivery of the initial scope namely a full-stack functioning RAG-bot.

The project's evolution from here can focus on improving overall user interaction through several key areas: enhancing the front-end and mobile user experience, decreasing backend response time, refining the accuracy of information provided, adding new languages and data sources, facilitating conversations with the Twitter bot via direct messages, and introducing the ability to include images from sourced content as well as integration with the Dune API.

Updates and new GitHub issues will be documented in the project's [repositories](#) to guide these improvements. Database updates are reflected in the [MEV.fyi Research Hub sheet](#).

Moreover, the project is exploring various monetization strategies to support its maintenance and development, ensuring that mev.fyi remains a neutral and valuable resource for the community.

Open questions on funding options for mev.fyi maintenance and development

For mev.fyi to thrive and evolve, identifying effective funding avenues is essential. Possible approaches include:

- Proposing value-added services to distinct stakeholders:
- to developers, protocols/foundations: providing API access, bots for communication platforms (Telegram, Twitter, Discord), partnerships incentivised by adding documentation, YouTube channels, podcasts [...]
- to content creators: incorporating their materials (articles, videos, books) into the database, make mev.fyi the content creation enabler e.g. powering threads, taking part in podcasts to tell about mevfyi
- other stakeholders: to be determined
- to developers, protocols/foundations: providing API access, bots for communication platforms (Telegram, Twitter, Discord), partnerships incentivised by adding documentation, YouTube channels, podcasts [...]
- to content creators: incorporating their materials (articles, videos, books) into the database, make mev.fyi the content creation enabler e.g. powering threads, taking part in podcasts to tell about mevfyi
- other stakeholders: to be determined
- To what extent is it acceptable to reference materials behind a paywall, such as books, online courses, or reports? e.g. enable paywall content based on user active subscriptions from partners.
- Would it be advisable to create a dashboard that provides a detailed breakdown of operational and development costs? This could potentially encourage financial contributions by suggesting donations of a specific percentage of the project's operational costs or by offering the funding of specific features.

On the design choices for reranking

In the development of mev.fyi, we've faced the challenge of efficiently managing our database to serve user queries without compromising on speed or accuracy. To this end, we've implemented a content reranking process that emphasizes academically reliable sources, ensuring high-quality responses.

In practice:

- Adding more documents is a direct trade-off with response time and would lead to a worse user experience.
- Prioritizing any source inherently introduces a bias. We favour research papers and articles over YouTube transcripts to ensure that responses are accurate, though this usually results in more specialized terminology being used.
- Given that this is open-source software, users dissatisfied with the content selection have the option to fork and customize the application themselves. The app's design is modular, allowing for easy modification or replacement of the database through metadata filtering.
- Currently, we are utilizing the Pinecone pod database at an astonishingly low 0% of its total capacity, indicating the potential to incorporate millions more vectors. Presently, the database contains approximately 60k vectors.

Please let me know your feedback on the above open questions!

I would also love to have a mindshare in anything related to distributing the application to as many users as possible (say, the Next Billion

).

Thoughts on the experience and how it came together

I owe a tremendous debt of gratitude to Fred and the entire Flashbots team for their unwavering support throughout the development of mev.fyi. Their guidance was indispensable.

Additionally, the pioneers of RAG technology, including [Jerry Liu](#), the minds behind LlamaIndex, and the creative forces at Anyscale ([Goku](#) and [Philipp](#)), Pinecone, Weaviate, and YouTube's educational content creators, provided an invaluable foundation for our LLM backend.

Despite the initial ease of setting up and populating the database, navigating the complexities of the Llama Index library presented a steep learning curve, offering lessons in the nuances of advanced API usage.

Thanks to [Andres](#)' extensive support and oversight, the intricate process of data integration and model optimization was significantly streamlined, making the concept of 'chunking' a manageable aspect of our development journey. Achieving a functional LLM backend required about two months of dedicated effort, punctuated by continuous improvements in data quality, metadata enrichment, and user interaction refinement.

Embarking on front-end development without previous experience was challenging, leading to a prolonged period of trial and error before settling on a desktop-focused user experience. Fred's insights were instrumental in navigating these challenges and shaping the user interface into its current form.