**Overall judging criteria**

**Difficulty**

How technically complex is the problem that your project is tackling, and the solution that you have presented?

**Implementation:**

How complete and functional is your project? Is your project deployed? While your project does not need to be perfectly polished, the solution should at least be functional.

**Applicability:**

Is the project that your team is working on one that impacts people’s lives? Is this something that someone would have a use for in the real world?

**Creativity:**

Does the project address a new or unsolved problem? Does the project create a unique solution to a problem with an existing solution? Does your project solve a current problem we face in web2?

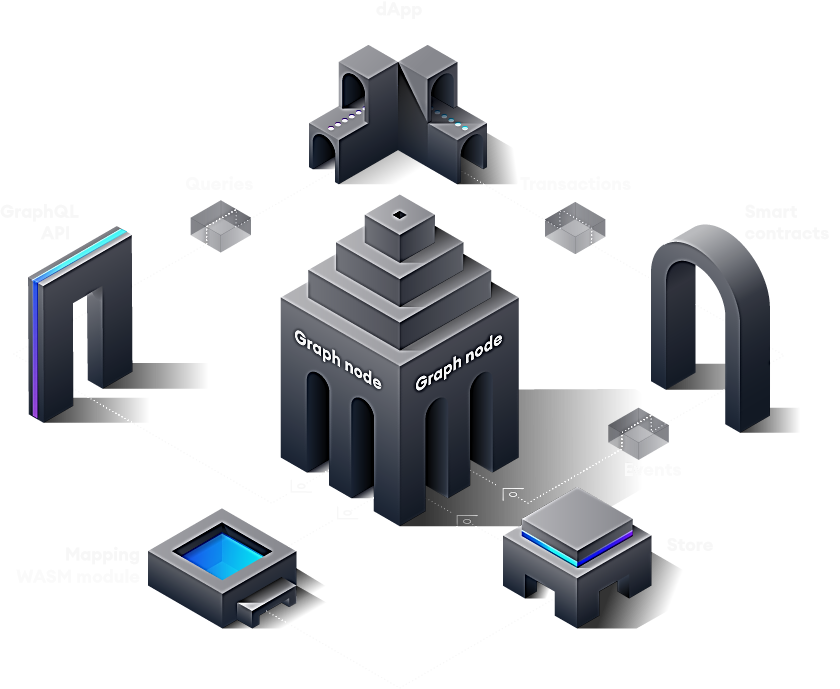
**The Graph**

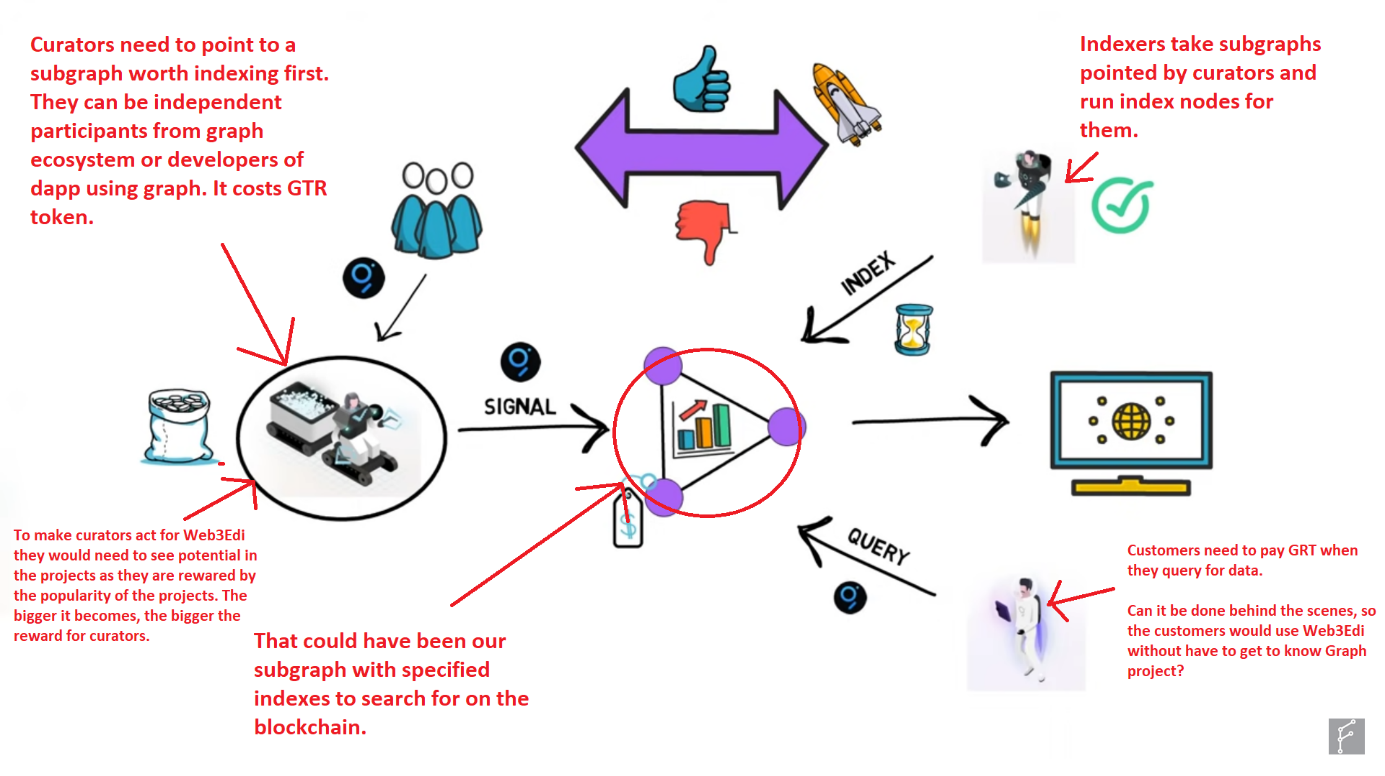
Challenge: Build a dapp leveraging an existing subgraph or built using your own custom subgraph.

The Graph solves this with a decentralized protocol that indexes and enables the performant and efficient querying of blockchain data. These APIs (indexed "subgraphs") can then be queried with a standard GraphQL API. Today, there is a hosted service as well as a decentralized protocol with the same capabilities. Both are backed by the open source implementation of [Graph Node](https://github.com/graphprotocol/graph-node).

The Graph learns what and how to index Ethereum data based on subgraph descriptions, known as the subgraph manifest. The subgraph description defines the smart contracts of interest for a subgraph, the events in those contracts to pay attention to, and how to map event data to data that The Graph will store in its database.

Once you have written a subgraph manifest, you use the Graph CLI to store the definition in IPFS and tell the indexer to start indexing data for that subgraph.





Graph allows for fully decentralized dapp without the need to really on data querring from centralized companies. – using Graph might help to show Web3Edi as fully decentralized service.

How to create a subgraph:

* **Subgraph Studio** - used only for subgraphs that index **Ethereum mainnet**
* **Hosted Service** - used for subgraphs that index **other networks** outside of Ethereum mainnnet (e.g. Binance, Matic, etc)

Technicalities of running the subgraph:

<https://thegraph.com/docs/en/developer/quick-start/>

My opinion: Depending on the work needed for implementing might be useful. Work and investment are required for curators to be interested in protocol and therefore having an indexer provide his node for indexing. Without community support Web3Edi would have to invest in GTR tokens and incentivize indexers by itself. The Graph is not necessary for Web3Edi to work, so it’s not essential. Users don’t even have to know how querying is performed and I guess centralized querying is easier to do. But in the long run The Graph makes the dapp fully decentralized which can make it more interesting for decentralization enthusiasts. We might also find subgraph already existing in The Graph and use it to be eligible for rewards – that might be easier, but I am not sure for what to look for. I guess data such as creator of the course, maybe some sort of type of course is also available for indexing (can such data be stored on the blockchain in the nfts? – considering how OpenSea market operates with properties of nfts I guess we could add such properties to course nfts like “Marketing”; “Programming” etc.). Additionally maybe a completion certificate type itself could be indexed as well.

Here’s a guide on how to build nft with graph: https://www.youtube.com/watch?v=QRljLAbpfPI

**Decentology (Hyperverse solution)**

Challenge: Involving the Hyperverse, participants will create a dapp accompanied by a video walk-through.

The Hyperverse is an open marketplace of community-built, audited, easy to discover smart modules. Our goal - to make it easy for JavaScript developers to build Web3 apps without writing a single line of smart contract code. Think “npm for smart contracts.”

Available with [Flow](https://flow.com/), [Algorand](https://www.algorand.com) and [Ethereum](https://ethereum.org) blockchains. It’s an early access.

You can find the most up to date list of smart modules and apps in the [Hyperverse Monorepo](https://github.com/decentology/hyperverse-mono).

My opinion: As I can see it’s an early build of a service designed to ease the creation of web3 dapps for java developers. The link above shows all available modules for now. It doesn’t seem to be large. I don’t know what of it might be useful, but if we can build some parts of Web3Edi with the use of this Hyperverse, we might be eligible for rewards. We might also do it the other way around and contribute to the “tribes” on the Hyperverse website and also most likely be eligible for rewards. If it’s an easy task for You Tech guys, then I see no reason, why we wouldn’t try to connect our work with Hyperverse.

There’s a video about how to build smart modules with Hyperverse: https://www.youtube.com/watch?v=E9WrvKwUnpg

**Polygon**

Challenge: Just to build on Polygon

My opinion: I saw the documentation and it all seems to be technical about how to build on Polygon, if you are already building on Ethereum. From what I heard from You yesterday, it seems, that you know exactly how to build dapps on Polygon, so I do not see any reason not to build on it. Nowadays ETH mainnet is super expensive and many crypto projects are migrating.

But I found a section about how to use The Graph for indexing, so maybe that might be useful:

https://docs.polygon.technology/docs/develop/graph

**Filecoin**

Challenge: The best use of IPFS and/or Filecoin for content addressing and decentralized storage (projects that use them indirectly via NFT.storage, web3.storage, Fleek, Textile, or other tooling also qualify)

* How the project uses to IPFS and/or Filecoin and/or decentralized storage services that build on top of either with example links to where in code those technologies are used.
* Provide comprehensive README, incl. a clear description of the work and mention of how our tech is used in the Technologies Used section of README.

**Categories for juding criteria:**

* **Design**: Is it easy to use? Does the visual design delight you? If targeting developers, is the experience smooth and intuitive?
* **Novelty/Originality**: Does it have unique and novel features or combine features of other products or services in a unique and novel way?
* **Technical Accomplishment**: Is the code elegant, clean, and technically interesting? Is the project complete? Does the project interact with IPFS/Filecoin under-the-hood (i.e., more than simply retrieve a CID from a gateway)?
* **Potential Impact:** Does it fill a gap in the universe, and have the potential to be used frequently by a large, growing and diverse audience or market?
* **Creativity/Fun/Wow-Factor:** Is it creative and put a smile on people’s faces?

**IPFS** is a decentralized web protocol for sharing data quickly and easily. **Filecoin** is built on top of IPFS and supports storing data long-term via on-chain deals. Together, they help us break free from centralized services while conveniently allowing us to enjoy the same luxuries of speed and guaranteed storage that centralized services would bring.

My opinion: Filecoin is a go to solution for decentralized storage – another brick in the house of fully decentralized apps. Considering that it was obvious to You yesterday, that we are going to build with this I assume that we will. Especially because of the rewards.

The usage of Filecoin is much simpler than The Graph and it’s a much more user oriented service. You just upload your data to the blockchain (to the miners) that create sectors of data. You pay what you’re due in FIL and your data is stored for a limited period of time. It would be essential for Web3Edi to use data storage solution and while Filecoin is a little bit harder to implement than centralized solutions right now, I assume it is not that much harder and gives the “decentralized” proof for the projects. I also heard that some miners would even consider providing storage for free or very cheap to get Fil+ deals that use “Data Caps” status given by the community to the miners and have 10x the mining power added to their storage. It is an info from a year back, so might be outdated.

In the long run Filecoin would be essential for long-term data storage, if we want Web3Edi to be as decentralized as possible or just for the hackathon to win more prizes. Additionally the judges take creativity and fun as part of the criteria, so I think that if possible, our design should be lite, with maybe a small mascot called “Edi” or “Eddi” that represents the idea of Web3Edi on the site. I will take look into, what I can do with that, if we agree on that. Plus the text should most likely also be very user friendly and provide smile for the user with the usage of Web3Edi. Maybe one of the courses in the mockup or demo should be something invigorating like “How to make your child smile” or something silly like that, to show that educational services are not only about tech, but also about everyday problems and concerns.

Addressing “**Novelty/Originality” seems to be the hardest. I think that the idea will score great with “Potential Impact” and that’s where most points for the idea would go to. With “Novelty” the judges might think more about the implementation of the idea in the dapp. Most likely it is essential to underline the future functions like renting and granting courses to charities or schools in the video, to give us more points. Even though we will most likely not be able to implement them in any way in 6 days to the demo, maybe it’s worth to create a small mockup on the side showing potential future expansion with those functions?**

**Also remember that the judges would like to see Filecoin used “under the hood” which I assume means that users don’t need to notice, that they are using decentralized storage at all. The smooth process in the usage of the dapp seems to have high priority as well.**

There’s a link to storage helpers in the documentation. They are mentioned in the challenge description as eligible to use. There is a part focused on NFTS:

Site: <https://nft.storage/> Video: <https://www.youtube.com/watch?v=Ckb4RRJo-W0>

The rest are these:

* [Web3.storage](https://web3.storage): Data storage service that stores and retrieves data on IPFS and Filecoin ([video](https://youtu.be/lPEqg6oL3Nk))
* [Textile Powergate](https://docs.textile.io/powergate/): Highly configurable wrapper for IPFS+Filecoin inside a Docker container, with bridges to NEAR, Polygon, and other smart contract networks ([video](https://www.notion.so/Getting-started-with-IPFS-Filecoin-173c73d4d8d64765a42058594bc46bb7))
* [Fleek Space Daemon and Space SDK](https://fleek.co/space-sdk/): For decentralized browser, mobile, or desktop development

And this below seems to help with integrating it on Polygon:

[Textile Powergate](https://docs.textile.io/powergate/): Highly configurable wrapper for IPFS+Filecoin inside a Docker container, with bridges to NEAR, Polygon, and other smart contract networks ([video](/173c73d4d8d64765a42058594bc46bb7#19581ff0b47b45d9ba051e81cfde23e2))

Example of usage:

[Flowwow](https://github.com/jochasinga/flowwow), an NFT pet shop built on [NFT.storage](http://nft.storage/) x Flow

**Livepeer**

Challenge: Seems like more than the tech side of things they want to see an amazing video first and foremost.

My opinion: For this prize grab, I assume the most important would be video. I will then try to use music, fast transitions, highlighted text and lively commentary to make it more “wow”. I am not experienced enough to make animations from 0, but will be able to put life into the video with ready to use assets from the web – preferably copyrights free.

I will make a scenario for the video and how I see it, after we agree on the path we want to take during today’s call. For the video I will need footage of the demo usage and as such I quess the video will have to be completed on the 5th, when we set our deadline, because it won’t be possible to complete it without proper voice-image synchronization. I can take the footage myself, but for the demo I am counting on You guys.

I assume a good idea would be to make a comparison between web2 courses and web3edi courses where web2 are seen as limiting and Web3edi as liberating.

Because Livepeer allows you to stream videos on the blockchain, maybe we could also think about mentioning in the video presentation a future function for the course instructors to host webinars and live lessons on the Web3Edi platform and distribute them as a NFT to attendees as a proof of being a part of live workshop.

**Course video: 1,5min or 30sec**

**60sec-120sec for demo part & 3 minutes for engaging part**

Maybe it would be even possible to make live celebrations on the site and also give proof of attendance as an NFT. This idea seems to stray off a little from Web3Edi main functionality, mainly because celebrations are not really a part of education service, but it’s worth to talk over on today’s call. (though I just thought how cool would it be if for example someone was on an important event and instead of torn ticket or photos, he could actually show his NFT that proves his presence; this could have some viral potential in case of concerts and such). BUT from what I see making a streaming service on Web3Edi is nowhere near a high priority for Web3Edi, because they seems to care only for the video presentation in the challenge descriptions and not for the Livepeer implementation.

Would I have to mint our video presentation? I guess not, but I do wonder.

Here is guide for video minting:

<https://www.notion.so/Tutorial-How-to-Mint-a-Video-NFT-368a04d00f0f4ca9bb0e59b98ec04c40>

And here for building a live-streaming app:

<https://www.notion.so/Tutorial-Building-a-live-streaming-app-251d71af6b5f4b22b0db4f3b483405a8>

**Radicle (Drips)**

Challenge: Just to be “the best” of all the projects in using drips.

“Drips is a new Ethereum protocol that allows any creator to generate continuous income from their communities without relying on bank accounts or extracting platforms.

Using Drips, creators & developers can issue monthly memberships and give their fans unique benefits in exchange for their recurring financial support. Using the permissionless financial infrastructure of Ethereum, Drips memberships are represented by Non-Fungible Tokens (NFTs). These memberships are available to anyone and can enable benefits across any application on the web that supports Ethereum (Discord, Telegram, Reddit, Snapshot, Radicle Upstream & many more).

Fans can support and engage with their favorite creators and communities by purchasing memberships and by “dripping” funds to any address in the network in recurring configurable intervals.

Finally, Drips allow creators to automatically spread the love every time they get paid by specifying a percentage of their incoming funds to be dripped to other users. The recipient could be a charity, another collective, or another software dependency or creator that makes their work possible. Sharing funds through Drips has the potential to become a new culture norm and building block for a sustainable crypto economy.”

My opinion: Drips webapp seems not to be responding: <https://drips.radicle.xyz/> . In theory it sounds great, because Web3Edi is supposed to be subscription service. From the definition though, it is great that it allows for Web3Edi to give some money back from subscription to charities. It would be cool, if based on which courses NFTs a user has, a part of his subscription fee could go to chosen by the instructor charity as well. I guess it is to technical to implement for hackathon, but I wonder if this is possible at all?

**Hedera**

Challenge: The best usage of Hedera smart contracts.

My opinion: I just took a brief look into it. It is a separate blockchain, so we are most likely not gonna use it, because other projects have more compatibility with Ethereum and Polygon.