

Fragnova Network Testnet

This document will show you how to access and play around with the Fragnova Network testnet via Polkadot's App Explorer.

We will cover how to create a Polkadot account, access the Fragnova testnet, upload a Proto, and set the uploaded Proto's metadata on the testnet. We will also verify our testnet transactions by browsing the Fragnova marketplace.

Useful links: [Fragnova Testnet](#) - [Fragnova Marketplace](#) - [Fragnova Discord](#)

Substrate, Polkadot, and Fragnova

[Substrate](#) is the SDK with which you can build parachains (diverse individual blockchains that can run in parallel) and [Polkadot](#) is the means of securing these chains and allowing them to communicate with each other.

[Fragnova](#) is our custom blockchain built on Substrate and our test network (aka testnet) is now deployed and connected to Polkadot.

Accessing the Fragnova testnet

Any custom blockchain network built on Substrate can be connected to Polkadot and accessed via Polkadot's App Explorer menu.

However, since the Fragnova testnet is still being integrated with Polkadot's App Explorer menu we'll instead use [this direct link](#) to access our testnet via the App Explorer.

Note: Some browsers, like Firefox, can have trouble connecting. Please use Chrome.

The screenshot shows the Polkadot/Substrate Portal interface for the Fragnova Testnet. At the top, there's a header with the portal logo, a search bar containing the URL 'polkadot.js.org/apps/?rpc=wss%3A%2F%2Frpc.frgnova.network#explorer', and navigation icons for back, forward, and refresh. Below the header is a navigation bar with tabs for 'Accounts', 'Network' (selected), 'Developer', 'Settings' (with a notification badge), 'GitHub', and 'Wiki'. The main content area has a sub-navigation bar with tabs for 'Explorer' (selected), 'Chain info' (highlighted with an orange border), 'Block details', 'Latency', 'Forks', 'Node info', and 'API stats'. Under 'Chain info', it displays 'last block' (4.7 s), 'target' (6 s), and 'total issuance' (1.9999 BNOVA). The 'recent blocks' section lists the last 7 blocks, each with a hash and timestamp. The 'recent events' section indicates 'No events available'.

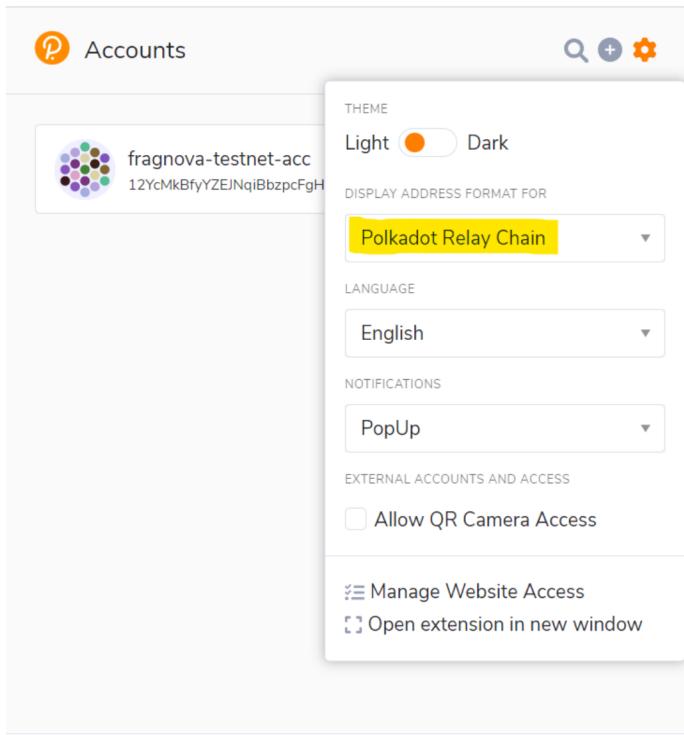
Block Number	Hash
76,731	0xae9a498480b6b8b327b5c9411254527ab514763c23a8fd5732e70942...
76,730	0xb4bdc4af5cddd77e69590647b2048fd54a1f1d8a5a18b19c7d88a588...
76,729	0xb5fe28c8d5c2b710a4659d5afb660a872717db26b860d0f76dd1d98c...
76,728	0x74dc2b3fd47bc9893f5e8aed42b721cbea779a864234ae99a0f2128f...
76,727	0x06fa2daffaa2aafaa5755d0a3745f6e4e15243a38baaa56c9c045502...
76,726	0xad6923974bca246726d24d034748c8bef624f5678cf7ffaaa6ff52ca...
76,725	0x3dfc374f80c69a8a45a7ba148b22f7b12a9375449f793c1cbf290ef8...

Get a Polkadot account

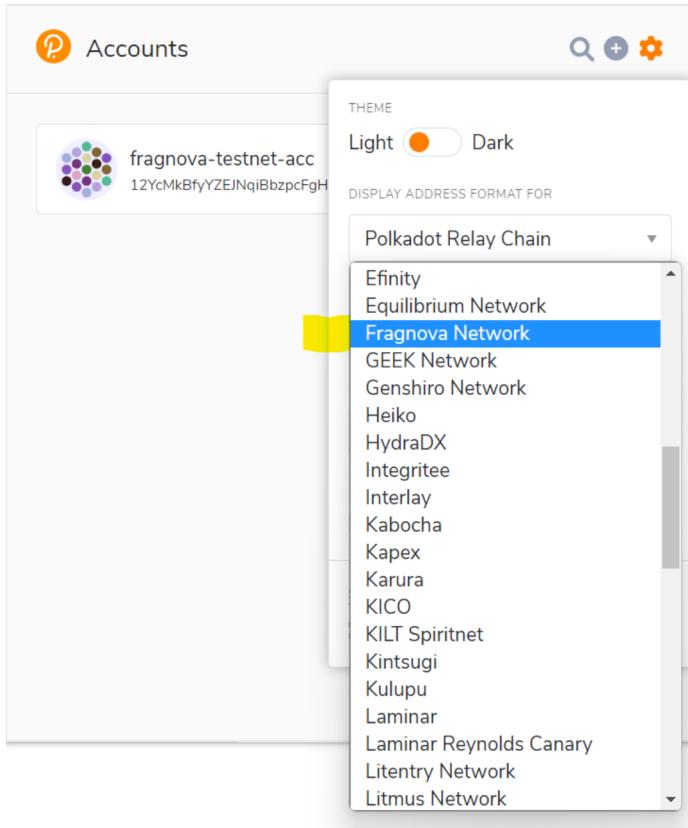
You'll need a Polkadot account to access the Fragnova testnet.

If you already have a Polkadot account skip to the next section, else follow the instructions [here](#) to create one using Polkadot's secure browser extension.

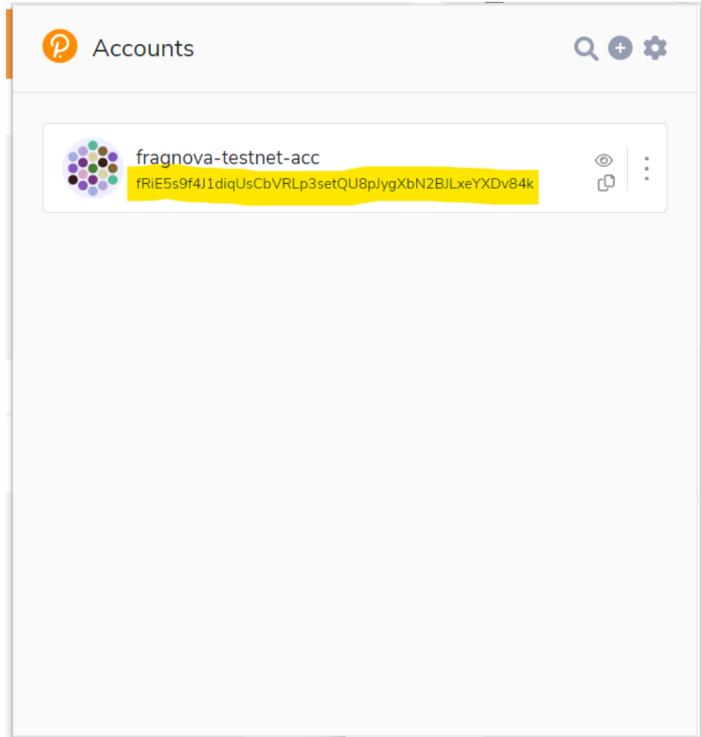
Towards the [end of the tutorial](#) you'll notice that you've set up your account address to display as per the Polkadot mainnet format.



Let's go back to the browser extension settings and select the `Fragnova Network` option instead.



This will allow the browser extension to display your account address in the Fragnova Network format like it's shown on the App Explorer's [Accounts](#) page (App Explorer > Header Menu > Accounts > Accounts).



A screenshot of the Fragnova Testnet dashboard. At the top, it shows the network name 'Fragnova Testnet' and a dropdown for 'Accounts'. Below the header, there are tabs for 'Accounts', 'My accounts', and 'Vanity generator'. The 'My accounts' tab is selected. It displays a balance of '999,999 MNOVA'. There are filters for 'sort by date' and 'filter by name or tags'. On the right side, there's a detailed view of an account: 'FRAGNOVA-TESTNET-ACC (EXTENSION)', address (fRxE5s9f4J1diqUsCbVRLp3setQU8pJygXbN2BJLxeYXDv84k), type 'injected', and flags 'Injected'. It also shows a 'Balance' section with 'TOTAL' and 'TRANSFERABLE' amounts both at '999,999 MNOVA'. A 'Send' button and a 'Copy' button are visible next to the account details.

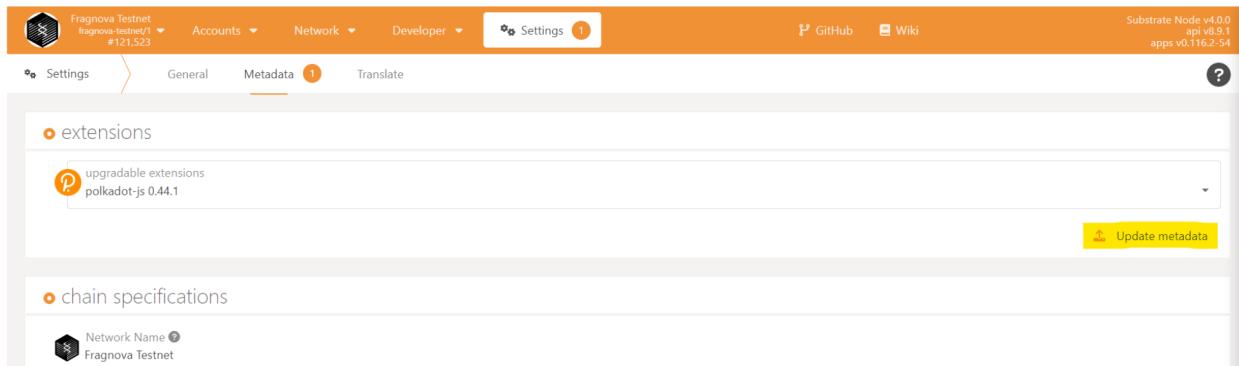
Note - Both your Polkadot account and Fragnova account addresses are representations of your actual account. These are transformations based on the [ss58](#) prefix allotted to that chain (for example, all Polkadot accounts start with the `1` while all Fragnova accounts start with `fR`).

Update chain metadata

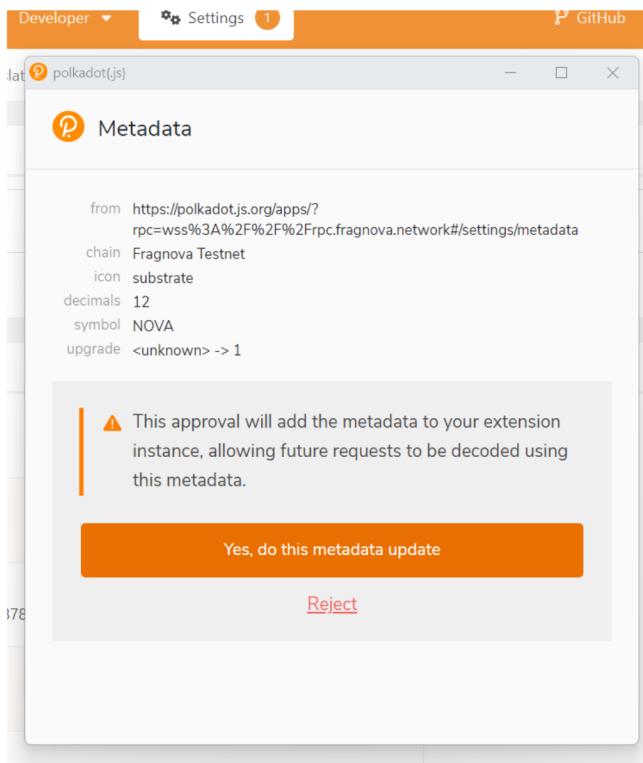
You should also update your metadata to enable your Polkadot browser extension to parse and decode the raw call data instead of doing it manually via the Polkadot App > Extrinsics > Decode (more details [here](#)).

To update your extension metadata for Fragnova click the **Settings** tab in the header menu and then click the **Metadata** tab.

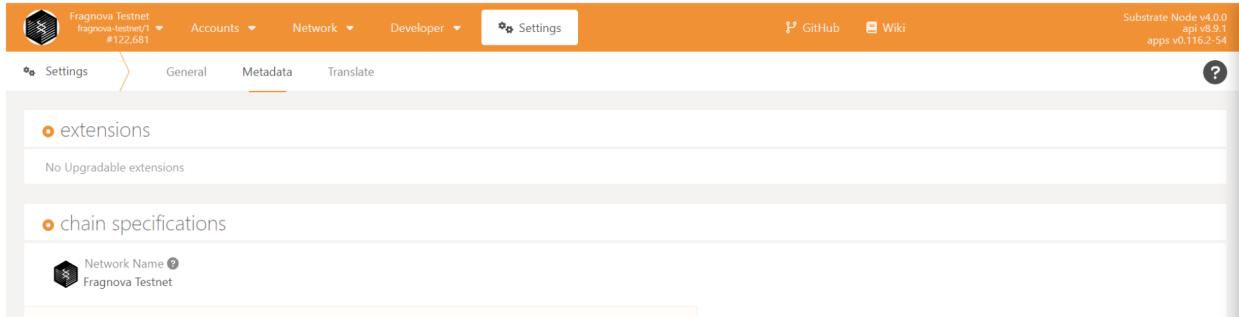
On this page, click the **Update Metadata** button on the right side in the extensions pane.



A pop-up will open for confirmation.



After confirmation, you'll see that the **Update metadata** button is no longer visible.



Get some free NOVA

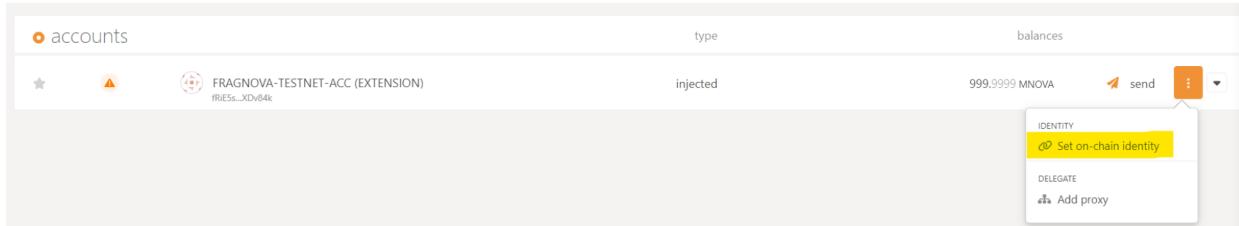
As this is a testnet you'll need some amount of test currency to be able to submit any kind of transactions on the network. Don't worry, you can get the test currency for free!

Fragnova's digital currency is called NOVA and you can get some test NOVA on our Discord channel [X-Y-Z](#) by sharing your Fragnova testnet account address.

Set on-chain identity

If you want to display your name (or an identifier) along with the assets you upload, then you should set your on-chain identity.

Go to the [Accounts](#) page, click the three vertical dots on the right, and then click the Set on-chain identity option.



A modal will open up. Set your display name (as well as include any other fields) and click Set Identity.

register identity

display name ?	satendra
legal name ?	<none>
email ?	<none>
web ?	<none>
twitter ?	<none>
riot name ?	<none>
total deposit ?	10.0000
include field <input type="checkbox"/>	
pico	

Clear Identity Set Identity

Since this is itself a transaction on the chain, you'll be required to Sign and Submit it.

authorize transaction

Sending transaction identity.setIdentity(info)
Set an account's identity information and reserve the appropriate deposit.

Fees of 86.2981 micro NOVA will be applied to the submission

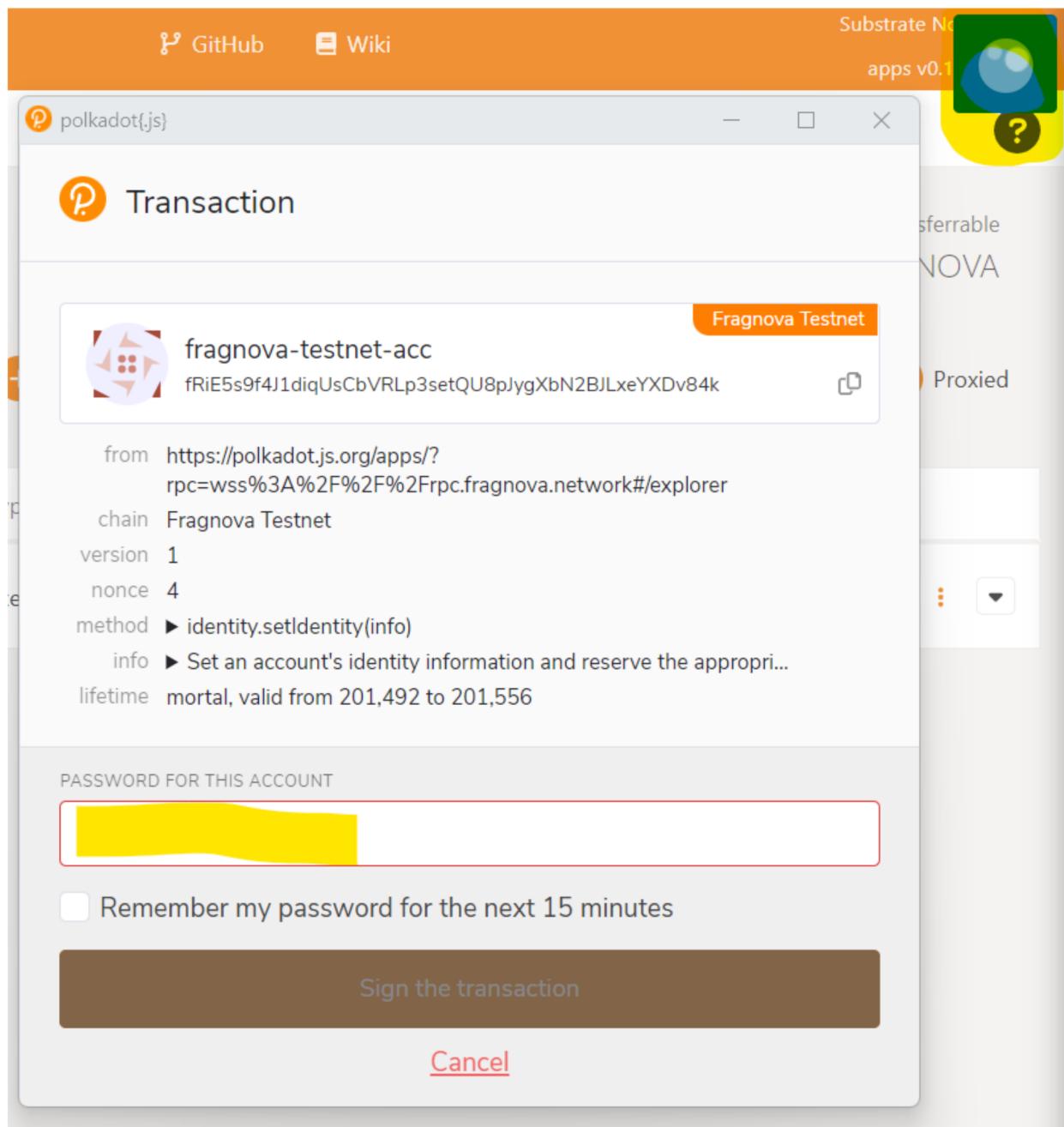
sending from my account FRAGNOVA-TESTNET-ACC (EXTENSION)	The details of the transaction including the type, the description (as available from the chain metadata) as well as any parameters and fee estimations (as available) for the specific type of call.
<input type="checkbox"/> Do not include a tip for the block author	The sending account that will be used to send this transaction. Any applicable fees will be paid by this account.
call hash 0x1b1659135828010862103b7ca10a118cbcd69f603dfddafe766e2713187d04e7	<input type="checkbox"/> Adding an optional tip to the transaction could allow for higher priority, especially when the chain is busy.
<input type="checkbox"/> Sign and Submit	Sign and Submit
total deposit ? 10.0000	pico

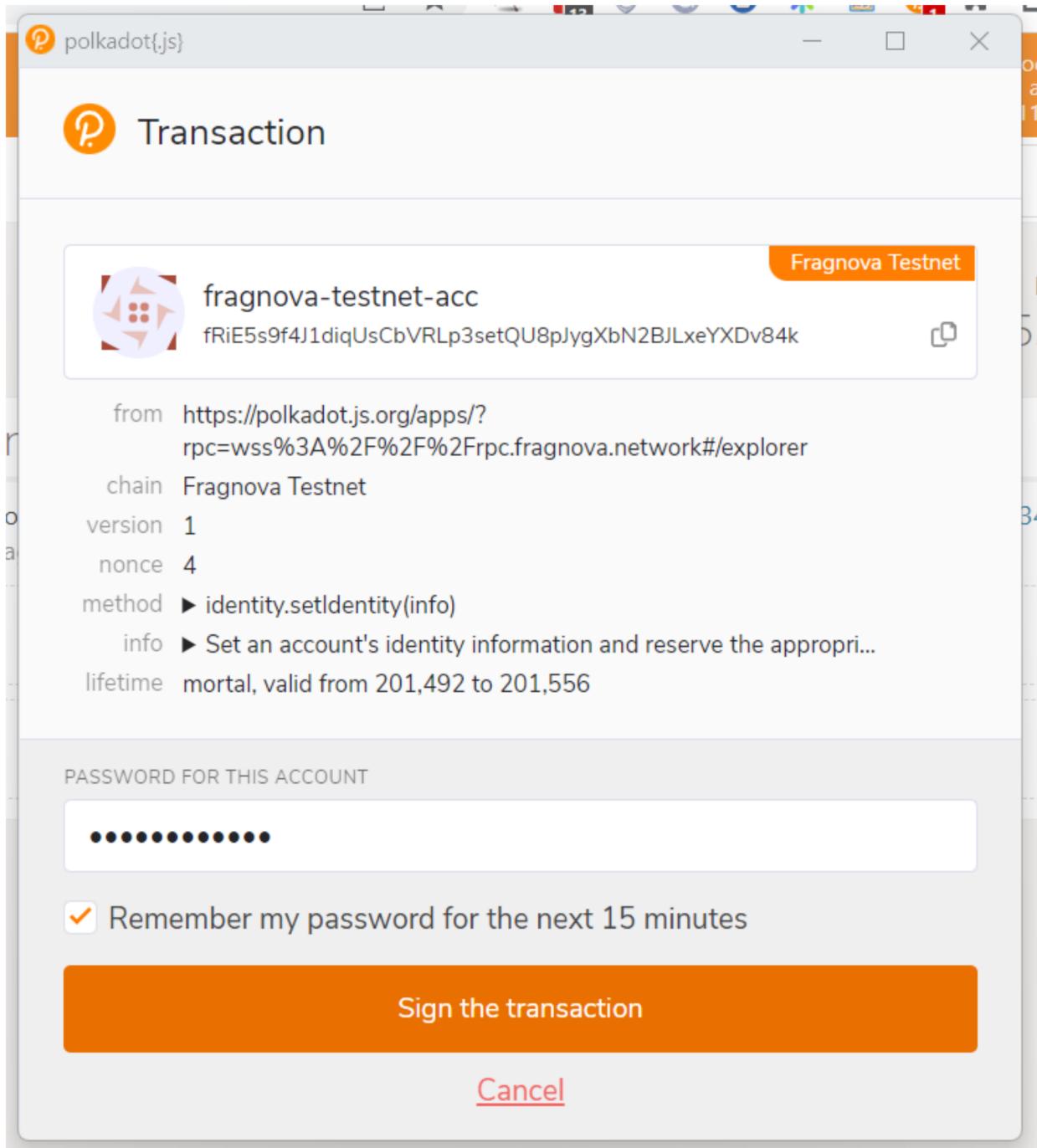
Clear Identity Set Identity

Even though you're signed in, a password confirmation is required for every transaction.

The Polkadot browser extension will prompt you for the same (you can set it to remember the password for the next 15 minutes).

Till you provide the password the transaction will show as trying to connect (see the rotating balls in the top right corner).





Once submitted, the transaction will complete within a few seconds.

Head over to the [Network](#) page and you should be able to see it.

The screenshot shows the Fragnova Testnet Explorer interface. At the top, there are tabs for Accounts, Network, Developer, and Settings. Below the tabs, there's a search bar with placeholder text "block hash or number to query". The main content area has two sections: "recent blocks" on the left and "recent events" on the right. The "recent blocks" section lists several blocks with their hash and timestamp. The "recent events" section lists several events with their type and timestamp.

On the [Accounts](#) page you'll see that your on-chain identity has been updated.

The screenshot shows the Fragnova Testnet Accounts page. At the top, there are tabs for Accounts, Network, Developer, and Settings. The "Accounts" tab is selected. The main content area displays a summary of total balance (999.9999 MNOVA) and total transferrable amount (999.9999 MNOVA). Below this, there are buttons for sorting accounts by date or name/tags, and links for adding accounts, restoring JSON, adding via QR, multisig, and proxied accounts. A table below shows the account details, including the account name (SAIENGRA), type (injected), and balance (999.9999 MNOVA).

Transacting on the testnet

In this tutorial, we will focus on how to upload Proofs and set their metadata (but feel free to explore other Proto functions too).

Navigate to the [Extrinsics](#) page (Header menu > Developer > Extrinsics).

Fragnova Testnet
Fragnova-testnet/1
#202,237

Substrate Node v4.0.0
api v8.9.1
apps v0.116.2-65-x

Accounts My accounts Vanity generator

total balance 999,9999 MNOVA

sort by date filter by name or tag

accounts type balances

Contracts Chain state Extrinsic RPC calls Sign and verify Files (IPFS) JavaScript Utilities

Add account Restore JSON Add via QR Multisig Proxied

Click the system dropdown and select protos.

Fragnova Testnet
Fragnova-testnet/1
#202,272

Substrate Node v4.0.0
api v8.9.1
apps v0.116.2-65-x

Extrinsics Submission Decode

using the selected account
- SATENDRA

submit the following extrinsic
system -> setCode(code)

Set the new runtime code.

code: Bytes
0x prefixed hex, e.g. 0x1234 or ascii data

file upload

Submit Unsigned Submit Transaction

Fragnova Testnet
Fragnova-testnet/1
#202,266

Substrate Node v4.0.0
api v8.9.1
apps v0.116.2-65-x

Extrinsics Submission Decode

using the selected account
- SATENDRA

free balance 999,9999 MNOVA
fr1E5s9f4J1diqUsCbVRlp3setQU8p3ygXbN2B3LxeYXDV84k

system
grandpa
identity
indices
multisig
protos
mnova

setCode(code)

Set the new runtime code.

file upload

Submit Unsigned Submit Transaction

Click on the field displayed to the right of protos and you'll see all the available transactions you can do with protos.

The screenshot shows the Substrate Node v4.0.0 extrinsics submission interface. The top navigation bar includes tabs for 'Extrinsics', 'Submission', and 'Decode'. The 'Extrinsics' tab is active. The main area displays the selected account as 'SATENDRA'. A dropdown menu under 'submit the following extrinsic' shows the option 'detach(protoHash, targetChain, targetAccount)' highlighted with a yellow box. Below this, input fields for 'protoHash' (with placeholder '0x prefixed hex, e.g. 0x1234 or ascii data'), 'targetChain' (set to 'PalletDetachSupportedChains EthereumMainnet'), and 'targetAccount' (with placeholder '0x prefixed hex, e.g. 0x1234 or ascii data') are visible. A file upload button is also present. At the bottom right are 'Submit Unsigned' and 'Submit Transaction' buttons.

This screenshot is similar to the first one but shows the 'detach' extrinsic's documentation expanded. The 'detach(protoHash, targetChain, targetAccount)' entry is highlighted with a yellow box and has a tooltip explaining it as 'Detach a proto from this chain by emitting an event that includes a signature.' Below this, other extrinsics like 'patch', 'setMetadata', 'stake', 'transfer', and 'unstake' are listed with their descriptions. The bottom right buttons are 'Submit Unsigned' and 'Submit Transaction'.

Upload a Proto

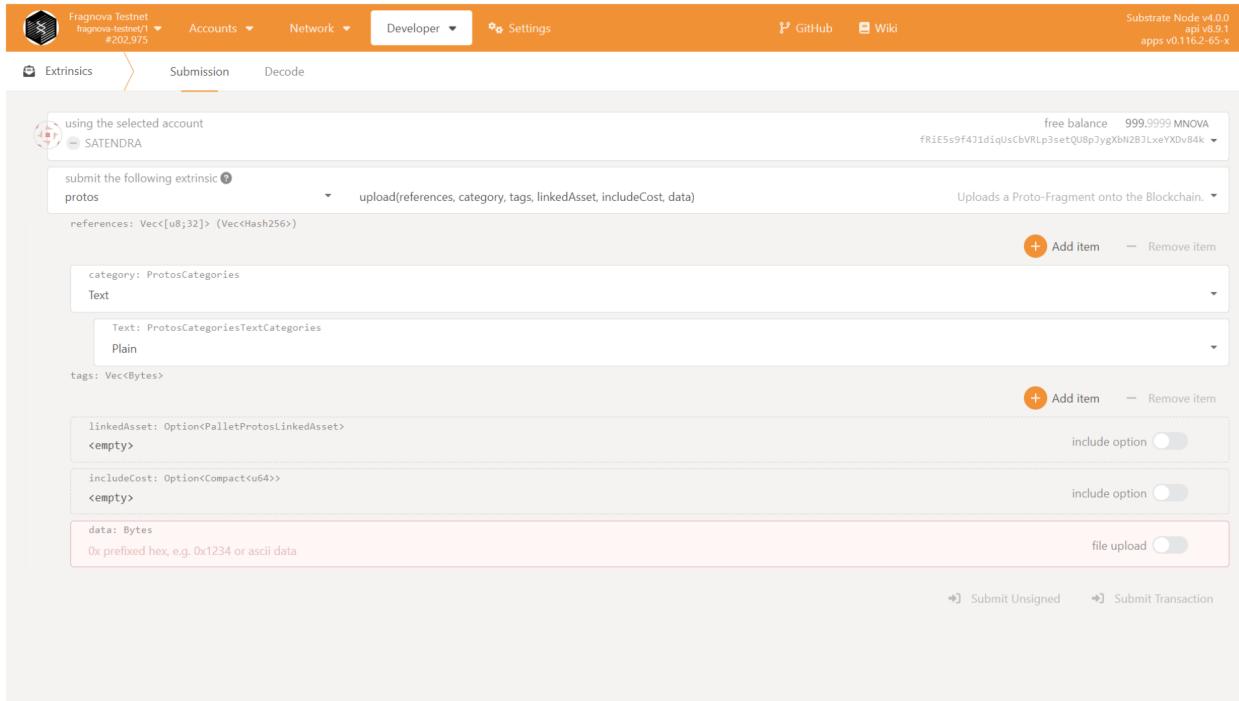
To upload a Proto, choose the **upload** dropdown option.

The screenshot shows the Substrate Node v4.0.0 extrinsics submission interface. The 'Extrinsics' tab is active. The main area displays the selected account as 'SATENDRA'. A dropdown menu under 'submit the following extrinsic' shows the option 'upload(references, category, tags, linkedAsset, includeCost, data)' highlighted with a yellow box. Below this, input fields for 'protoHash' (placeholder '0x prefixed hex, e.g. 0x1234 or ascii data'), 'targetChain' (set to 'PalletDetachSupportedChains EthereumMainnet'), and 'targetAccount' (placeholder '0x prefixed hex, e.g. 0x1234 or ascii data') are visible. A file upload button is also present. At the bottom right are 'Submit Unsigned' and 'Submit Transaction' buttons. A tooltip for the 'upload' entry states 'Uploads a Proto-Fragment onto the Blockchain.'

You'll see a form that has a number of fields - Category, Subcategory, and Data being the important ones for now.

NOTE - For this tutorial, we will ignore references: `Vec<[u8;32]>` (`Vec<Hash256>`) and tags: `Vec<Bytes>` fields (although you can upload tags as text strings like "Animation", "3D model", etc). You can also remove both these fields from the form by clicking their Remove item buttons).

The screenshot shows the Fragnova Testnet extrinsics submission interface. At the top, there are navigation links for Accounts, Network, Developer, and Settings, along with GitHub and Wiki links. The top right corner displays "Substrate Node v4.0.0", "api v8.9.1", and "apps v0.116.2-65-x". The main area is titled "Extrinsics" with tabs for Submission and Decode. A message indicates "using the selected account SATENDRA". Below this, a form is displayed for "upload(references, category, tags, linkedAsset, includeCost, data)". The "protos" field contains "references: Vec<[u8;32]> (Vec<Hash256>)". The "category" field is set to "ProtoCategories" and has a dropdown menu showing "Text" and "Plain". The "tags" field is set to "Vec<Bytes>". The "linkedAsset" field is set to "Option<PalletProtosLinkedAsset> <empty>". The "includeCost" field is set to "Option<Compact<u64>> <empty>". The "data" field is set to "Bytes: Bytes" and has a dropdown menu showing "0x prefixed hex, e.g. 0x1234 or ascii data". There are "Add item" and "Remove item" buttons for each field, and a "file upload" toggle switch for the data field.



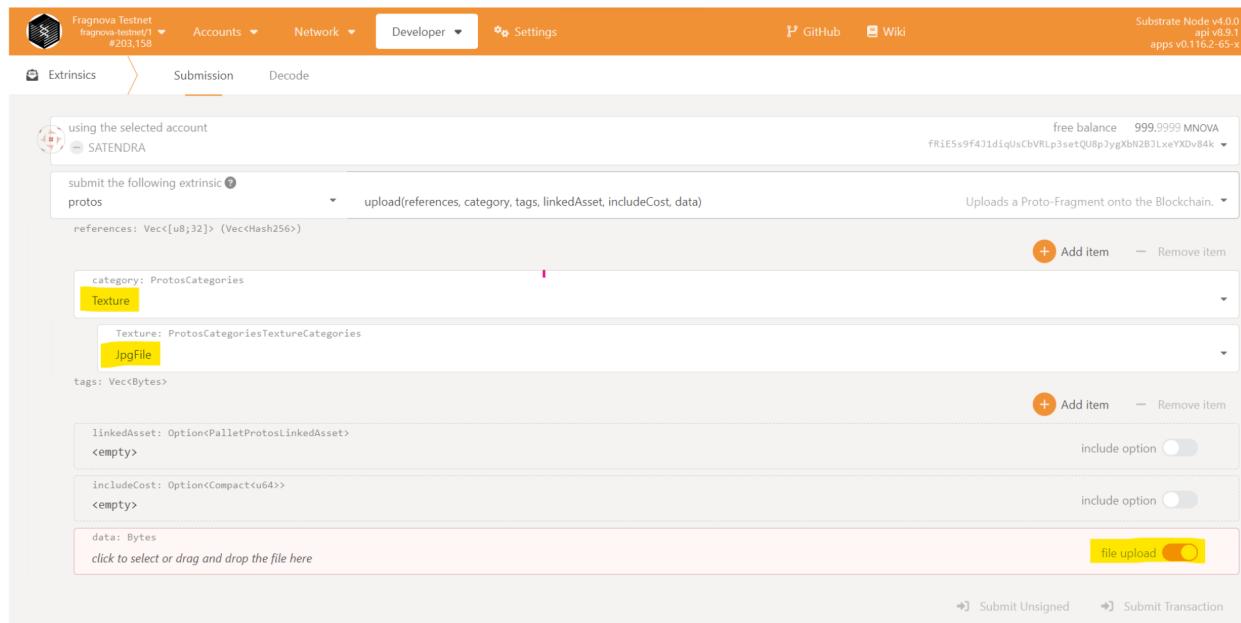
The following three fields define the kind of Proto we want to upload and its data.

- `category: ProtosCategories` - Category of the Proto
 - Text
 - Trait
 - Shards
 - Audio
 - Texture
 - Vector
 - Video
 - Model
 - Binary
- `Text: ProtosCategoriesTextCategories` - Subcategory of the Proto (based on the Category)
 - Text (Plain, Json)
 - Trait
 - Shards
 - Audio (OggFile, Mp3File)
 - Texture (PngFile, JpgFile)
 - Vector (SvgFile, TtfFile)
 - Video (MkvFile, Mp4File)
 - Model (GltfFile, Sdf, PhysicsCollider)
 - Binary (WasmProgram, WasmReactor, BlendFile)
- `data: Bytes` - Data of the Proto (text string or a file upload)

Uploading a texture Proto

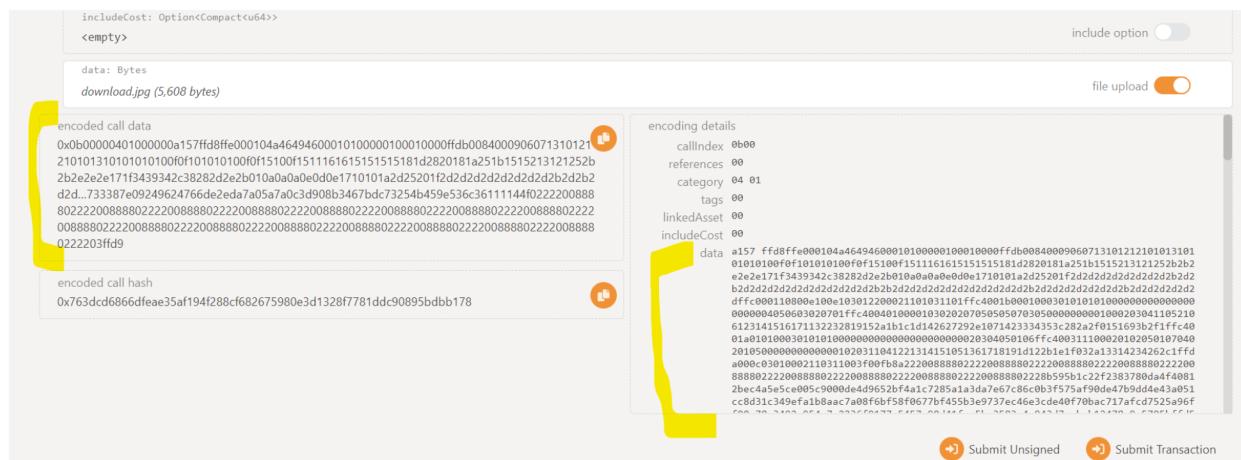
Let's upload a jpg image file Proto.

Select the required category (**Texture**) and subcategory (**jpg**), and upload a jpg file by enabling the **file upload** toggle in the **data: Bytes** field.



The screenshot shows the Substrate Node v4.0 UI with the 'Extrinsic' tab selected. The 'Submit' button is highlighted. The 'data: Bytes' field has a yellow border and the 'file upload' toggle is turned on.

The moment you upload your file (or any data) for the transaction, you'll see the encoded call, data, and hash that will be transmitted in this transaction.



The screenshot shows the Substrate Node v4.0 UI displaying the transaction details after file upload. The 'data' field contains the encoded data and the 'file upload' toggle is turned on.

Now click the **Submit Transaction** button to sign and submit the transaction via a modal.

Verify your account number (under Bytes) and note the Proto hash identifier (under [u8; 32])

The screenshot shows the Fragnova Testnet dashboard. At the top, it displays account information (fragnova-testnet/1 #203,244), network status (Network v4.0.0), developer tools (Developer v8.9.1), and node details (Substrate Node v4.0.0, api v8.9.1, apps v0.116.2-65-x). Below this, there are tabs for Explorer, Chain info (selected), Block details, Latency, Forks, Node info, and API stats. The Chain info section shows metrics like last block (6.3 s), target (6 s), total issuance (4.9999 BNOVA), and finalized block (203,242). The API stats section shows best block (203,244) and last events (1). On the left, a sidebar lists recent blocks (e.g., 203,244, 203,243, 203,242, 203,241, 203,240, 203,239, 203,238) with their hex values. On the right, a sidebar lists recent events (e.g., protos.Uploaded, protos.MetadataChanged, protos.MetadataChanged, protos.MetadataChanged) with their hex values.

Verify on the marketplace

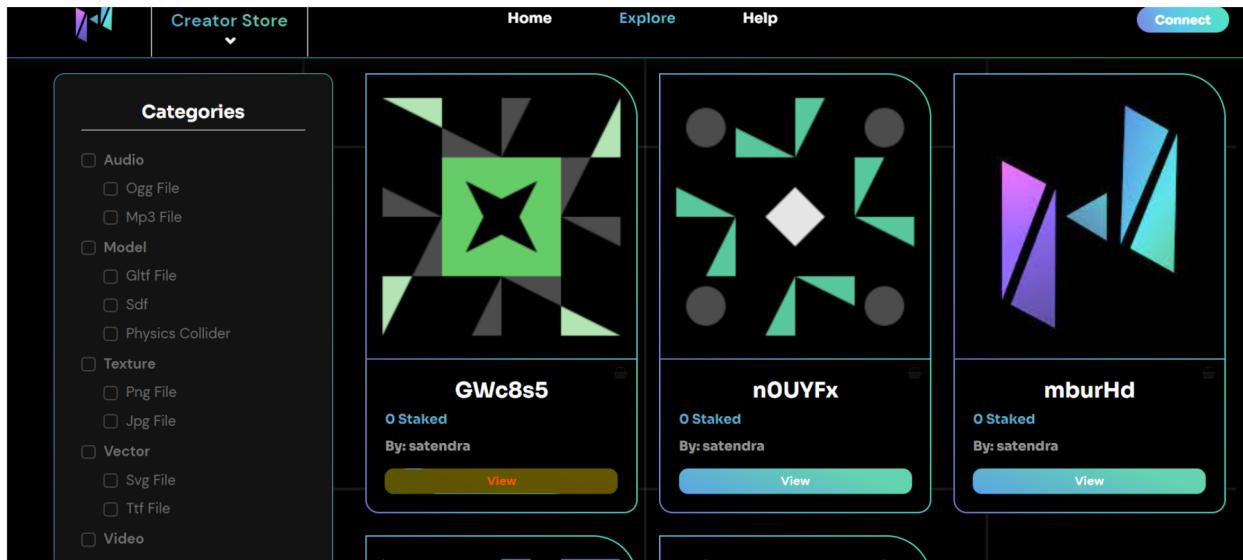
Now let's verify that our Proto has indeed been uploaded to the Fragnova testnet.

Head over to the [Creator Store](#) on the Fragnova Marketplace (currently a staging environment) and search by your account number to see all the Protos you have uploaded on Fragnova's network.

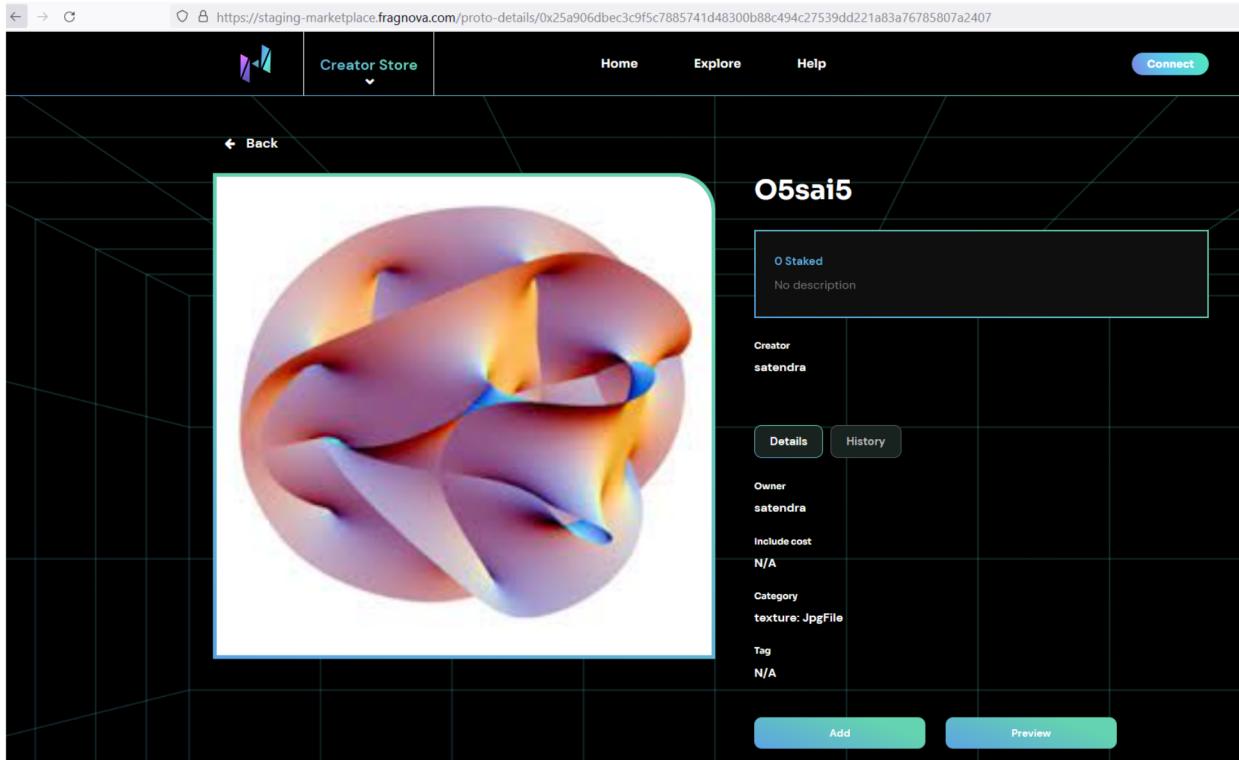
The screenshot shows the Fragnova Creator Store Explore page. The top navigation bar includes a logo, a dropdown menu for 'Creator Store', and links for Home, Explore, Help, and Connect. The main heading is 'Explore' with the sub-instruction 'Browse through the best community creations and up your game with the Protos available on the Fragnova Network.' Below this, there is a search bar with the placeholder 'Discover' and a hex value 'fRiE5s9f4JldiqUsCbVRlp3setQU8pJygXbN2BJLxeYXDv84k'. To the right of the search bar is a 'Sort By' dropdown. On the left, a 'Categories' sidebar lists 'Audio' (Ogg File, Mp3 File), 'Model' (Gltf File, Sdf, Physics Collider), and 'Texture'. Three Proto preview cards are shown: one with a green starburst icon, one with geometric shapes, and one with purple and blue abstract shapes.

NOTE - The thumbnail images shown here are not the actual Proto data/file but the Proto metadata image (which is initially generated automatically like the first two images, or can be set later via `setMetadata` function, like the 3rd image).

So click `View` on the first Proto in this list (or you can alternatively append your Proto's hash to the `view-proto` link <https://staging-marketplace.fragnova.com/proto-details/>)



Either way, you'll see the Proto we just uploaded, including the actual Proto image data and the Proto owner's display name.



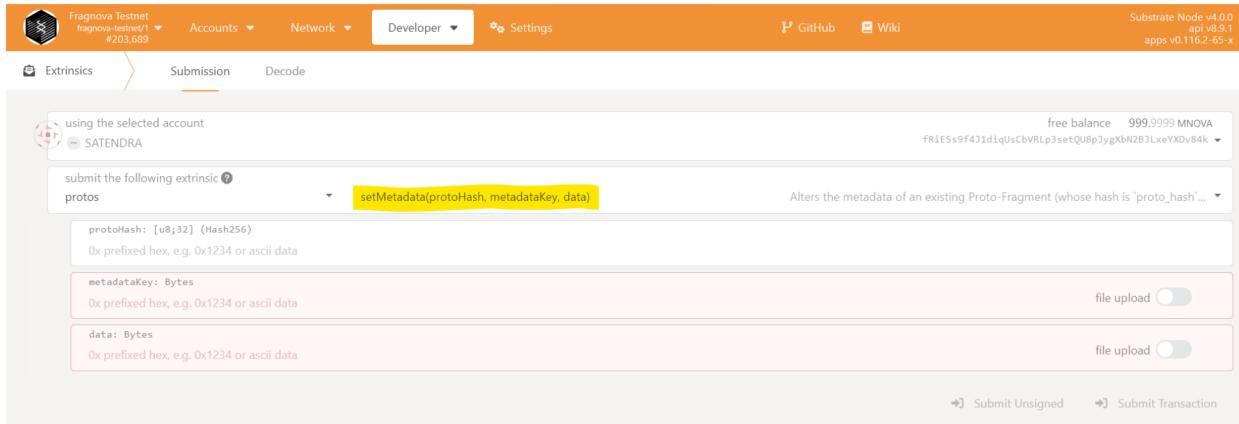
Similarly, you can upload and view other categories of Prots too, after choosing the correct category/subcategory combination.

Set Proto metadata

As you must have noticed, our recently uploaded Proto doesn't have much information - other than the Proto file data and the Proto owner (or account owner's) name.

To remedy this, let's add some metadata to our Proto.

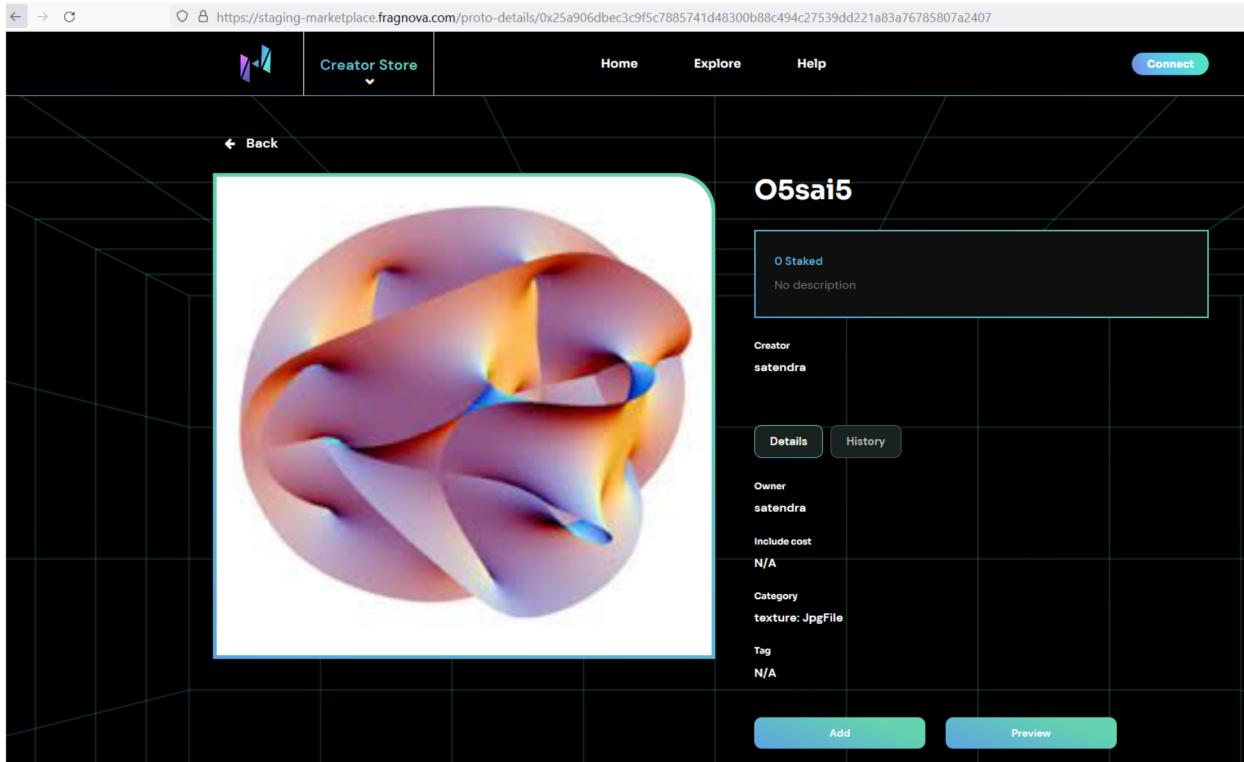
Head back to the [Extrinsics](#) page, but this time choose the `setMetadata` dropdown option under `protos`.



This form has three fields:

- `protoHash: [u8; 32] (Hash256)` - The hash of the Proto we want to update metadata for (in our case, the recently updated Proto).
- `metadataKey: Bytes` - The key of the metadata that we're updating. Accepts the following as valid values (but only one can be updated at a time):
 - `image` - a thumbnail image for the Proto
 - `json_description` - description of the Proto in plain text or json format
 - `json_attributes` - description of the Proto's attributes in json format (for e.g., `{"color": "blue", "size": "200x200"}`)
 - `title` - a title for the Proto in plain text
- `data: Bytes` - Text data for the description/title or the image data for thumbnail (enable file upload toggle).

We see that our newly uploaded Proto doesn't have a valid title (it's a randomly generated string at the moment) - `05sai5`



Updating the title

So let's add a meaningful title to this image Proto via the `setMetadata` function.

Grab this Proto's hash from the proto-details link page (`.../proto-details/<proto-hash>`) - in this case it's `0x25a906dbec3c9f5c7885741d48300b88c494c27539dd221a83a76785807a2407` - and paste it in the `setMetadata` function's `protoHash` field.

Set the `metadataKey` field to `title` and `data` field to `Calabi-Yau-manifold`.

Submit the transaction

using the selected account
SATENDRA

submit the following extrinsic ?
protos

protoHash: [u8;32] (Hash256)
0x25a906dbe... (highlighted)

metadataKey: Bytes
title

data: Bytes
Calabi-Yau-manifold (highlighted)

encoded call data
0xb0325a906dbe... (highlighted)

encoded call hash
0xd46511a7ed... (highlighted)

encoding details

callIndex 0b03
protoHash 25a906dbe...
metadataKey 14 7469746c5
data 4c 43616c6162692d5961752d6d616e69666f6c64
link #/extrinsics/decode/0xb0325a906dbe... (highlighted)

Submit Unsigned Submit Transaction

After signing and submitting the transaction, head over to the [Network](#) page and verify our `setMetadata` transaction went through.

last block 1.1 s target 6 s total issuance 4.9999 BNOVA

last events 1 finalized 203,920 best 203,923

recent blocks

- 203,923 0xc977cd7d83a4814e6ff7d6e5cd20bfff3729843a3d6141b8f2affc...
- 203,922 0x3c3eac304380de2155a198d34ca0b4e40ba0a3ea7a9c785c4ee8...
- 203,921 0x6b704ca8cdb23752b34f222c17e37b96a04b38257373a4cf2abc9...
- 203,920 0xc07dd8bdc1ab64450a9ec29f758762a5ef00fe021c44ba77531a...
- 203,919 ...

recent events

protos.MetadataChanged A Proto-Fragment metadata has changed

[u8;32]
0x25a906dbe... (highlighted)

Bytes
title (highlighted)

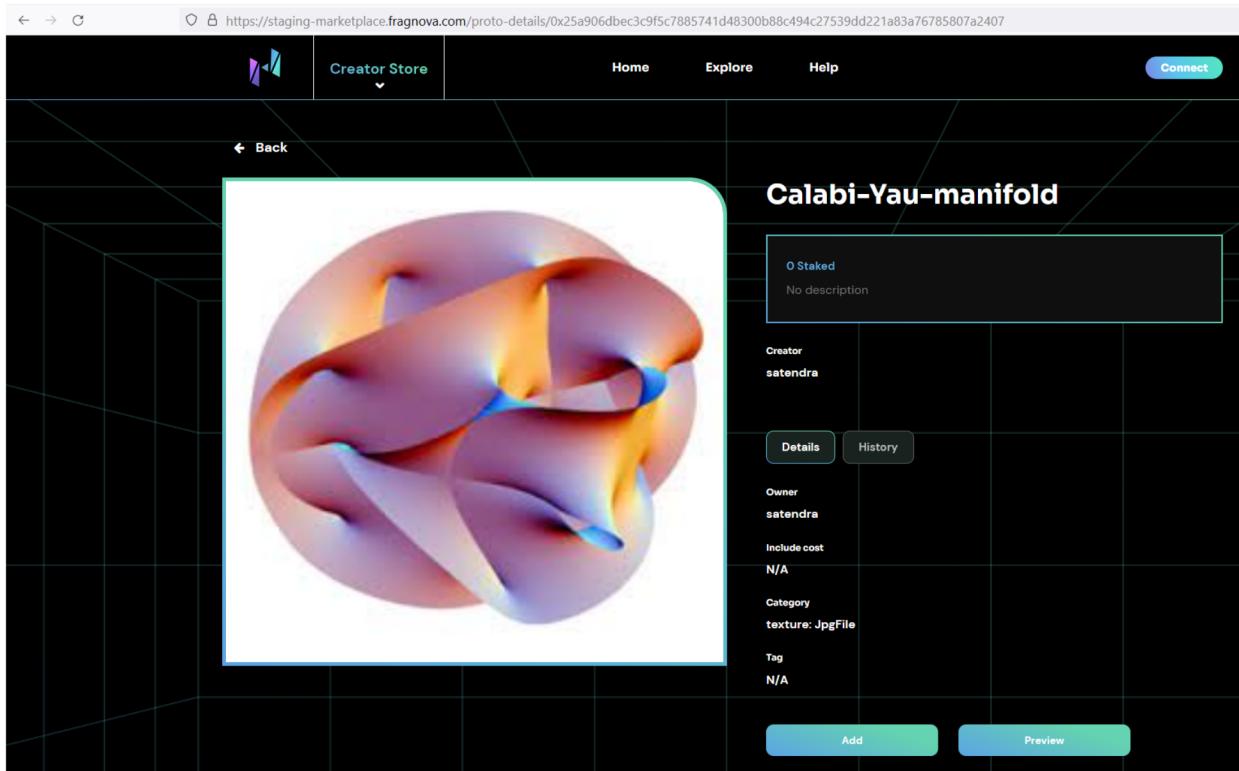
protos.Uploaded

Verify on the marketplace

Go to the [Creator Store](#) and again, search by your account number to list out your Prots and click to view the first Proto in that list, or go directly to the Proto by appending the Proto's hash to the proto-details link.

In our case, this link is

<https://staging-marketplace.fraqnova.com/proto-details/0x25a906dbe3c9f5c7885741d48300b88c494c27539dd221a83a76785807a2407>



You'll see that the title of the Proto has been updated.

Similarly, you can update the thumbnail image and the Proto description via the `setMetadata` function.

Other transactions

In this guide, we successfully uploaded a Proto to our testnet, set its metadata, and verified both these transactions on the Fragnova Marketplace.

However, these are not the only Proto transactions supported by Fragnova testnet.

If you head back to the [Extrinsics](#) page (Header menu > Developer > Extrinsics), you'll see a list of Proto transactions that allow you to attach/detach, stake/unstake, or transfer Proofs.

The screenshot shows the Substrate Node UI for the Fragnova Testnet. The top navigation bar includes links for Accounts, Network, Developer (selected), Settings, GitHub, and Wiki. The status bar indicates Substrate Node v4.0, api v8.10.1, and apps v0.116.2-70-x. The main area is titled 'Extrinsics' with tabs for Submission and Decode. A sub-section titled 'using the selected account' shows the account 'SATENDRA'. A dropdown menu for 'submit the following extrinsic' is open, showing several options under the 'protos' category. The 'detach' extrinsic is highlighted. Other visible extrinsics include 'patch', 'setMetadata', 'stake', 'transfer', and 'unstake'. At the bottom right are buttons for 'Submit Unsigned' and 'Submit Transaction'.

Besides Protos, the Fragnova testnet also supports Fragment transactions. You can create, buy, mint, or publish/unpublish Fragments on the testnet.

The screenshot shows the Substrate Node UI for the Fragnova Testnet. The top navigation bar includes links for Accounts, Network, Developer (selected), Settings, GitHub, and Wiki. The status bar indicates Substrate Node v4.0, api v8.10.1, and apps v0.116.2-70-x. The main area is titled 'Extrinsics' with tabs for Submission and Decode. A sub-section titled 'using the selected account' shows the account 'SATENDRA'. A dropdown menu for 'submit the following extrinsic' is open, showing several options under the 'fragments' category. The 'buy' extrinsic is highlighted. Other visible extrinsics include 'create', 'give', 'mint', and 'publish'. At the bottom right are buttons for 'Submit Unsigned' and 'Submit Transaction'.

Feel free to play around with these Protos and Fragments queries.

However, note that the Fragnova testnet and marketplace are under heavy development and evolving continuously, so expect this document to be a little outdated in some places (nothing major).

Also, as the Fragnova Marketplace for Fragments is still WIP, you cannot yet validate the Fragment transactions on the Fragnova Marketplace.