

Universidade de São Paulo

# Roteiro IPFS



Inserindo e recuperando arquivos no IPFS  
- sistema de arquivos P2P

<b>Trabalho de PCS3544 Segurança da Informação</b>
Otávio Henrique Monteiro

24 de Julho de 2022

# Conteúdo

<b>Universidade de São Paulo</b>	<b>1</b>
<b>Conteúdo</b>	<b>2</b>
<b>O IPFS</b>	<b>3</b>
<b>Instalação</b>	<b>3</b>
<b>IPFS Desktop</b>	<b>4</b>
Adicionando arquivos	6
Compartilhando arquivos	7
Importando arquivos	8
Exibindo arquivos por meio de um gateway	9
<b>CLI</b>	<b>10</b>
<b>Integração com outros projetos</b>	<b>12</b>
Exemplo em Javascript (Node)	12
Exemplo com recuperação	13

# O IPFS

O IPFS pode ser descrito como um protocolo descentralizado para armazenar conteúdo - como dados, websites, arquivos e aplicações - bem como acessar esses dados de forma descentralizada. Podemos usar o IPFS para várias finalidades diferentes, como para eliminar os problemas de censura, de um único ponto de falha ou de armazenamento distribuído.

As informações geralmente são organizadas na Internet através de locais, i.e. endereços, que disponibilizam dados de um servidor, por exemplo por meio de uma URL. O IPFS, por outro lado, utiliza "endereçamento baseado em conteúdo", onde os dados estão localizados com base no conteúdo em si por meio de um identificador exclusivo ou CID (Content ID), chamado hash (que é calculado sobre esse conteúdo).

Ao armazenar arquivos no IPFS, o sistema cria um objeto IPFS, que são limitados na quantidade de espaço que podem ocupar, definido em 256 KBs. Para arquivos maiores que 256 KBs é criado um objeto que referencia os demais que armazenam o conteúdo. Isso poderá ser verificado em um dos exemplos tratados adiante.

Para encontrar dados dentro da rede, o IPFS utiliza Tabelas de Hash Distribuídas ([DHTs](#)). Uma tabela hash é uma estrutura que mapeia chaves para diferentes valores e esta tabela é distribuída pelos nós da rede, que após localizar o arquivo determinam quais pares o hospedam.

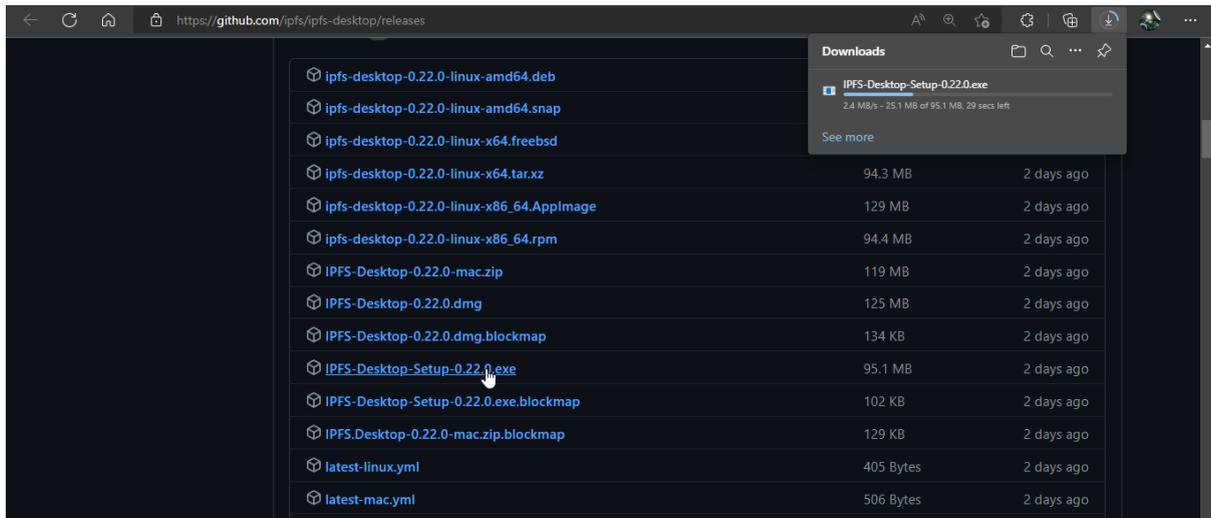
## Instalação

Existem dois principais modos de usar o [IPFS](#). Por meio de um GUI integrado do IPFS Desktop ou por meio da linha de comando, podendo ambos serem usados simultaneamente. Alternativamente, é possível a integração com projetos por meio de bibliotecas e pacotes específicos (e.g. para aplicações web e blockchain).

The screenshot shows a web browser window with the address bar displaying 'https://ipfs.io/#install'. The page header includes the IPFS logo and navigation links: 'Install IPFS', 'About', 'Install', 'Docs', 'Team', 'Blog', and 'Help'. Below the header, a sub-header reads 'Join the future of the web right now — just choose the option that's right for you.' The main content area is titled 'Store and provide files' and features two installation options:

- IPFS Desktop**: Labeled 'IPFS for everyone', this option offers a GUI with a menu bar and tray shortcuts for adding, pinning, and sharing files. It includes a full IPFS node ready for heavy-duty hosting and development. A button labeled 'Install IPFS Desktop' is provided.
- Command-line install**: Labeled 'All IPFS, no frills', this option is for users who want to use IPFS from their terminal. It provides step-by-step instructions for getting up and running on the command line using the Go implementation of IPFS, with directions for Windows, macOS, and Linux. A button labeled 'Install the IPFS CLI' is provided.

Detalhes sobre versões e passos de instalação podem ser encontrados no [site da ipfs.io](https://ipfs.io) para a versão Desktop e CLI, para diferentes sistemas operacionais.

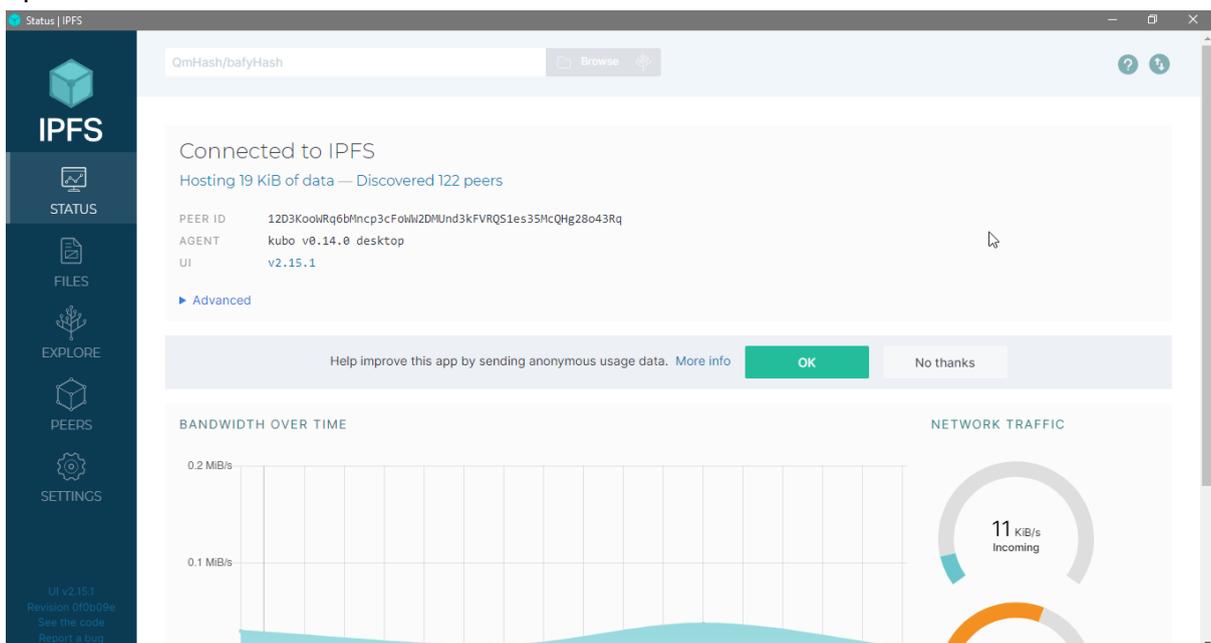


Neste tutorial instalaremos a versão do desktop GUI para Windows, baixado diretamente do repositório GitHub. Posteriormente a versão CLI (kubo) também será explorada.

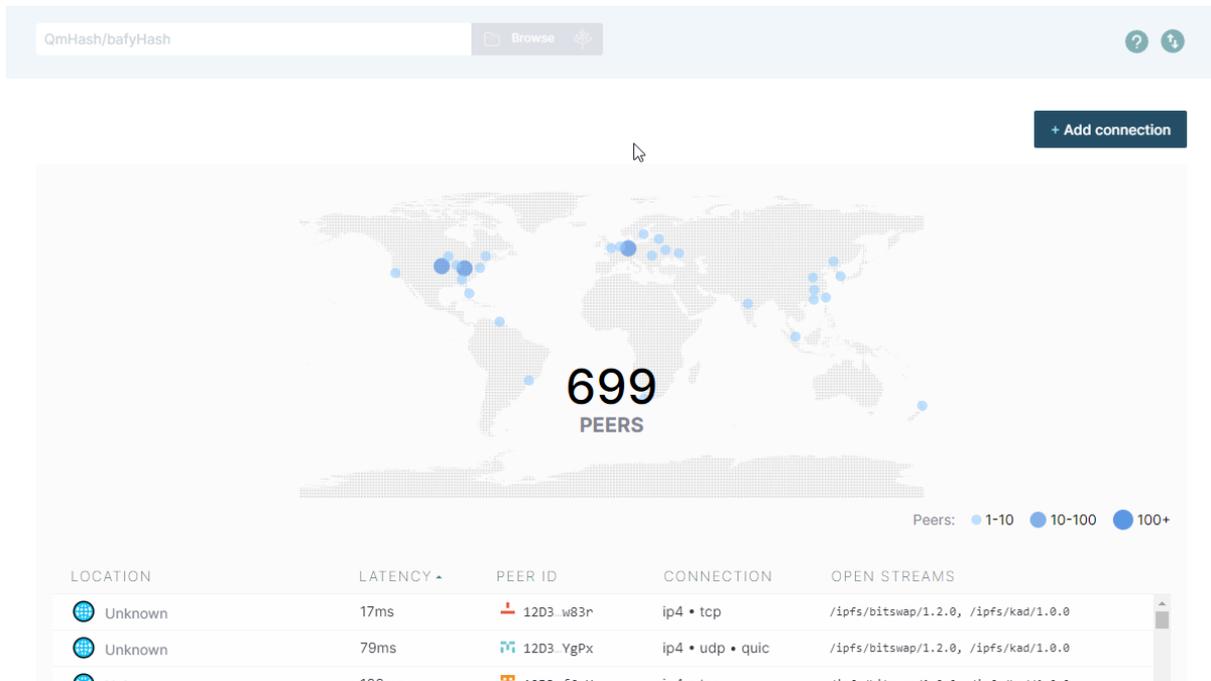
## IPFS Desktop

Com a instalação padrão realizada, um daemon de conexão é executado e fica disponível na barra de tarefas do SO, contando com múltiplos atalhos

O GUI do IPFS Desktop é aberto após a instalação, contando com uma barra lateral, à esquerda, de recursos disponíveis. A página inicial mostra o estado da conexão com a rede IPFS (realizada após permissão concedida pelo firewall) e o status de Download e Upload.



Outros recursos interessantes também estão disponíveis, tal como um mapa dinâmico de conexões com pares, onde podemos ver que o maior número de conexões ocorre nos EUA e Europa.



É recomendado ativar a opção nas configurações de dicas sobre o funcionamento do IPFS por CLI, útil para seções posteriores deste roteiro.

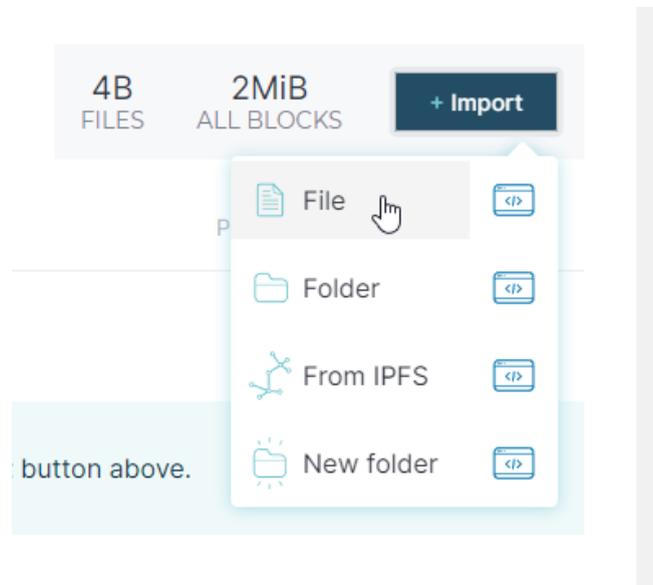
**CLI TUTOR MODE**

Enable command-line interface (CLI) tutor mode

Enable this option to display a "view code"  icon next to common IPFS commands. Clicking it opens a modal with that command's CLI code, so you can paste it into the IPFS command-line interface in your terminal.

## Adicionando arquivos

A adição de arquivos a rede IPFS é muito simples pela versão Desktop. Basta abrir a aba de Arquivos (files) e selecionar o botão de importação, que pode ser um arquivo ou diretório. Ao escolher um arquivo de seu computador, por exemplo, será gerado um hash correspondente e o arquivo estará disponível na rede para download por pares.



Neste exemplo, selecionamos uma foto da USP, que aparece na lista de arquivos em seu nó da rede.

Name ↑	Pin Status	Size
 usp.jpg QmbUx5jY1kNnF35fXDJ4ATkatYgUja8A9kk3Yg1VUV1xb		208 KiB ...

Esse arquivo pode ser visualizado localmente ou compartilhado.

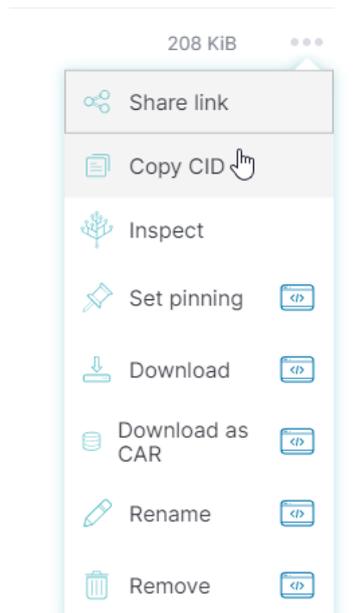
Files / usp.jpg

209KiB FILES    3MiB ALL BLOCKS    ... More

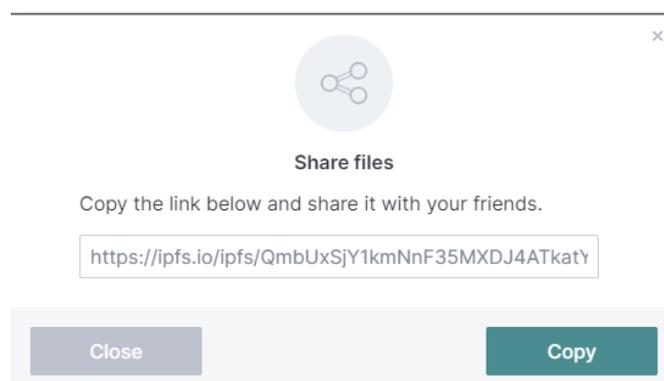


## Compartilhando arquivos

O compartilhamento pode acontecer por meio de hash (CID) ou link gerado por meio de um gateway, essas opções encontram-se no lado direito na lista de arquivos.



Para visualização por link, uma informação importante é o gateway utilizado. No caso, o gateway público padrão da instalação é do ipfs.io, mas este pode apresentar lentidão que leva a timeout quando usado no navegador. Para solucionar tal questão, podemos mudar o gateway padrão nas configurações ou alterar diretamente os links gerados.



Em subseção seguinte tratamos mais sobre o compartilhamento de links para exibição por gateway.

## Importando arquivos

Arquivos da rede também podem ser adicionados ao seu nó para download, exibição e hospedagem. Na aba de arquivos, ao selecionar a importação pela IPFS, abre-se a opção de importar um arquivo ou diretório por meio de seu hash correspondente, bem como selecionar um nome de armazenamento e exibição local.

  
**Import from IPFS**

Insert an IPFS path (CID) to import. Examples:

`/ipfs/QmZTR5bcpQD7cFgTorqxZDYaew1WqgFbd2ud9QqGPAkK2V`

`QmPZ9gcCEPqKTo6aq61g2nXGUhM4iCL3ewB6LDXZCtioEB`

`kDDP1mXWo6uco/wiki/InterPlanetary_File_System.html`

Name (optional)

Cancel

Import

Neste exemplo utilizamos o hash correspondente aos arquivos do espelho da Wikipédia existente no IPFS (o diretório contém mais de 600GB de dados!) escolhendo somente a página referente ao tópico de InterPlanetary File System.

Name ↑	Pin Status	Size
 InterPlanetary_File_System.html <small>baFkreibhnnjgk717kfr42vukcocc3wnkhtqzawte6uikyxgis6uzq</small>		12 KiB ...
 usp.jpg <small>QmblXsJY1kmInF35MxD34ATkatYgUja8A9kl3Yg1VUV1xb</small>		208 KiB ...

O arquivo baixado pode ser visualizado diretamente no GUI, mas existem outras formas de exibição.

Files / Inte...html

221KiB  
FILES 75MiB  
ALL BLOCKS More

```
<html><head>
  <meta charset="UTF-8">
  <title>InterPlanetary File System</title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="..-/s/style.css">
  <script src="..-/j/head.js"></script>
</head>
<body class="mw-body mw-body-content mediawiki" style="background-color: white; margin: 0; border-width: 0px; padding: 0px;">
  <div id="content" class="mw-body" style="padding: 1em; border-width: 0px; max-width: 55.8em; margin: 0 auto 0 auto">
    <a id="top"></a>
    <h1 id="titleHeading" style="background-color: white; margin: 0;">InterPlanetary File System</h1>
    <div id="mw-content-text">
  <div class="thumb tright"><div class="thumbinner" style="width:222px">[1]</sup> It was initially designed by Juan Benet.<sup>[2]</sup> The goal of IPFS is to facilitate a permanent and decentralized method of storing and sharing files.<sup>[3]</sup>

### Description

IPFS is a [peer-to-peer](#) distributed file system that seeks to connect all computing devices with the same system of files. In some ways, IPFS is similar to the [World Wide Web](#), but IPFS could be seen as a single [BitTorrent](#) swarm, exchanging objects within one [Git](#) repository. In other words, IPFS provides a high-throughput, content-addressed [block storage](#) model, with content-addressed [hyperlinks](#). This forms a generalized [Merkle directed acyclic graph](#) (DAG). IPFS combines a [distributed hash table](#), an incentivized block exchange, and a self-certifying namespace. IPFS has no single point of failure, and nodes do not need to trust each other.<sup>[4]</sup> Distributed Content Delivery saves bandwidth and prevents DDoS attacks which HTTP struggles with.

The filesystem can be accessed in a variety of ways, including via [FUSE](#) and over [HTTP](#). A local file can be added to the IPFS



[https://ipfs.io/ipfs/QmXoyvizjW3WknFiJnKLwHCnL72vedxjQkDDP1mXWo6uco/wiki/InterPlanetary\\_File\\_System.html](https://ipfs.io/ipfs/QmXoyvizjW3WknFiJnKLwHCnL72vedxjQkDDP1mXWo6uco/wiki/InterPlanetary_File_System.html)

Uma boa alternativa de gateway é o da infura, [ipfs.infura.io](https://ipfs.infura.io), mas existem outros [gateways públicos](#). A foto da USP publicada na rede anteriormente, por exemplo, pode ser vista em:

<https://ipfs.infura.io/ipfs/QmbUxSjY1kmNnF35MXDJ4ATkatYgUja8A9kW3YgiVUV1xb>

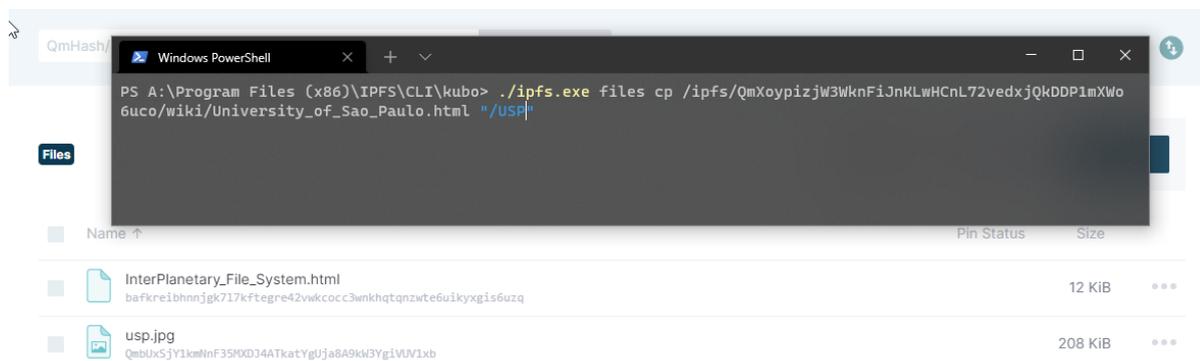
# CLI

A versão por linha de comando foi baixada e exportada, sendo utilizado o daemon da versão desktop. A instalação e uso pode ser encontrada na página de tutoriais da [ipfs.io](https://ipfs.io).

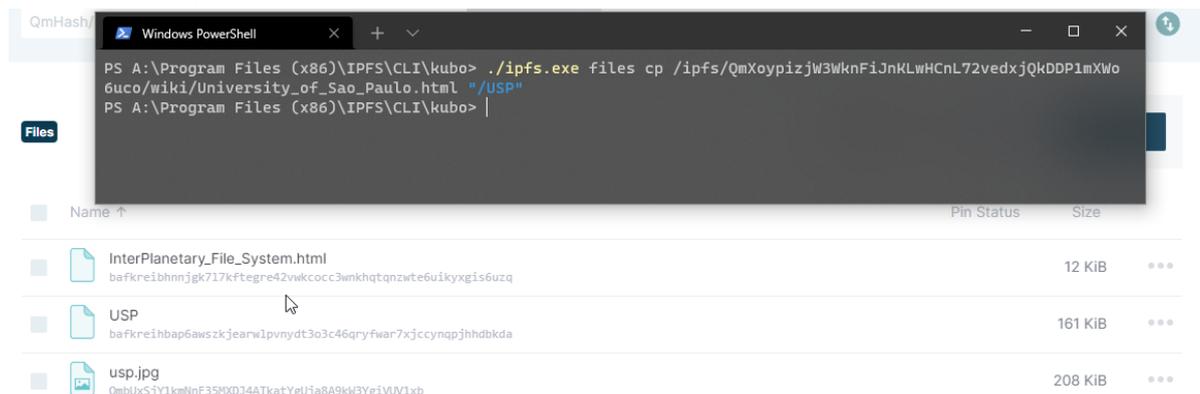
```
Directory: A:\Program Files (x86)\IPFS\CLI\kubo

Mode                LastWriteTime         Length Name
----                -
-a-----          21/07/2022    02:38           948 install.sh
-a-----          21/07/2022    02:38      60462080 ipfs.exe
-a-----          21/07/2022    02:38           508 LICENSE
-a-----          21/07/2022    02:38           520 LICENSE-APACHE
-a-----          21/07/2022    02:38       1046 LICENSE-MIT
-a-----          21/07/2022    02:38           467 README.md
```

O comando para o executável pode ser adicionado ao PATH das variáveis de ambiente, porém nesses exemplos foi utilizada referência direta ao arquivo do *kubo*.



Vemos aqui a adição do arquivo referente à página da USP na Wikipedia, com sua adição também aos arquivos do nó (ao invés de somente o download). O nome do arquivo local também foi alterado com o comando (que pode ser visualizado também pelas dicas da versão Desktop).

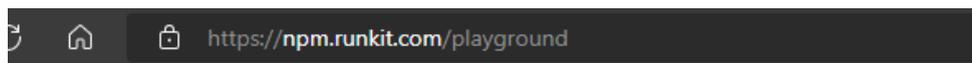




# Integração com outros projetos

Por meio de bibliotecas e pacotes específicos da aplicação podemos realizar a integração para upload e recuperação de arquivos. Isso pode ser feito em diferentes linguagens, principalmente Go e Javascript.

## Exemplo em Javascript (Node)



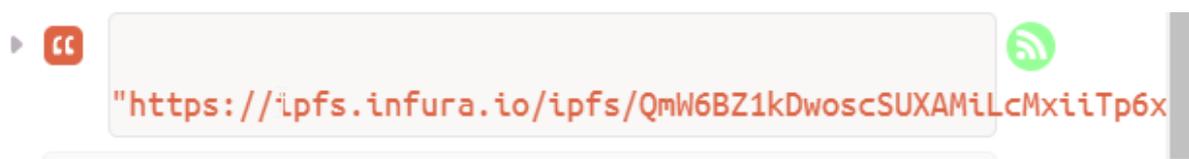
This is a **playground** to test code. It runs a full **Node.js** environment and already has all of **npm**'s 1,000,000+ packages pre-installed, including **playground**. Try it out:

```
1  const IPFS = require('ipfs-mini' 1.1.5 );
2  const ipfs = new IPFS({host: 'ipfs.infura.io',
3  port: 5001, protocol:'https' });
4  const data = "Texto de exemplo a armazenar na
5  Infura IPFS";
6  ipfs.add(data, (err, hash) => {
7    if(err) {
8      return console.log(err);
9    }
10   console.log('https://ipfs.infura.io/ipfs/' +
11   hash);
12 })
```

Save on RunKit   Node 14   help   **run**

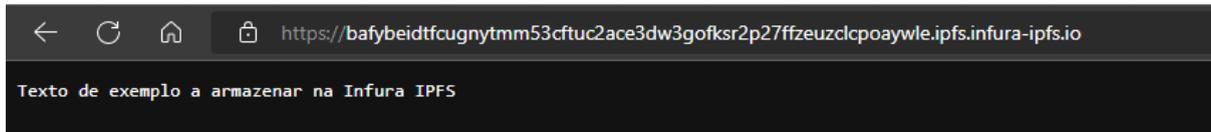
[RunKit+npm+IPFS playground | Código](#)

Ao executar o código, temos como log o seguinte link para o arquivo no gateway da Infura:



Exibindo o resultado final:

<https://ipfs.infura.io/ipfs/QmW6BZ1kDwoscSUXAMiLcMxiiTp6xaSrQZbu8yPydCrJcp>  
<https://bafybeidtfcugnytmm53cftuc2ace3dw3gofksr2p27ffzeuzclcpoaywle.ipfs.infura-ipfs.io>



## Exemplo com recuperação

Este exemplo utiliza a Moralis, plataforma pensada na Web3, com o upload e busca para recuperação do arquivo:

```
1 // Save file input to IPFS
2 const data = fileInput.files[0]
3 const file = new Moralis.File(data.name, data)
4 await file.saveIPFS();
5
6 //console.log(file.ipfs(), file.hash())
7
8 // Save file reference to Moralis
9 const jobApplication = new Moralis.Object('Applications')
10 jobApplication.set('name', 'Satoshi')
11 jobApplication.set('resume', file)
12 await jobApplication.save()
13
14 // Retrieve file
15 const query = new Moralis.Query('Applications')
16 query.equalTo('name', 'Satoshi')
17 query.find().then(function ([application]) {
18   const ipfs = application.get('resume').ipfs()
19   const hash = application.get('resume').hash()
20   console.log('IPFS url', ipfs)
21   console.log('IPFS hash', hash)
22 })
```