

**Name of the Experiement :** Store with IPFS – Decentralized File Upload

# Coding Phase : Pseudo Code/Flow Chart/Algorithm

 Start

 Create a .env file with API credentials.

 Import required modules.

 Define an asynchronous uploadToIPFS() function.

 Inside the function:

* Read the file using fs.createReadStream.
* Append to FormData.
* Post to Pinata endpoint.

 Handle response:

* Display IPFS hash and gateway link.

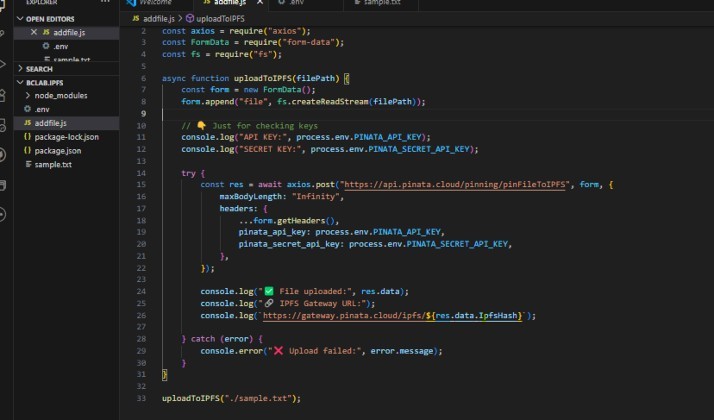
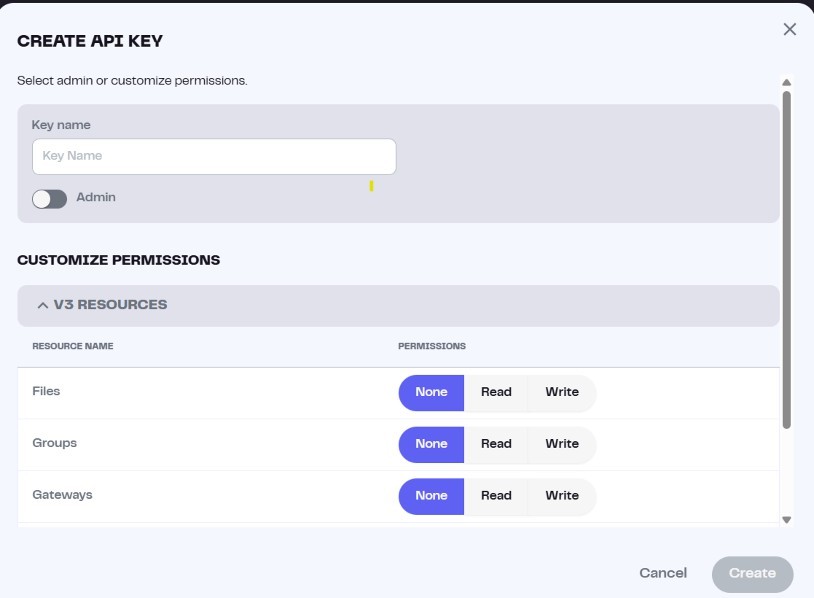
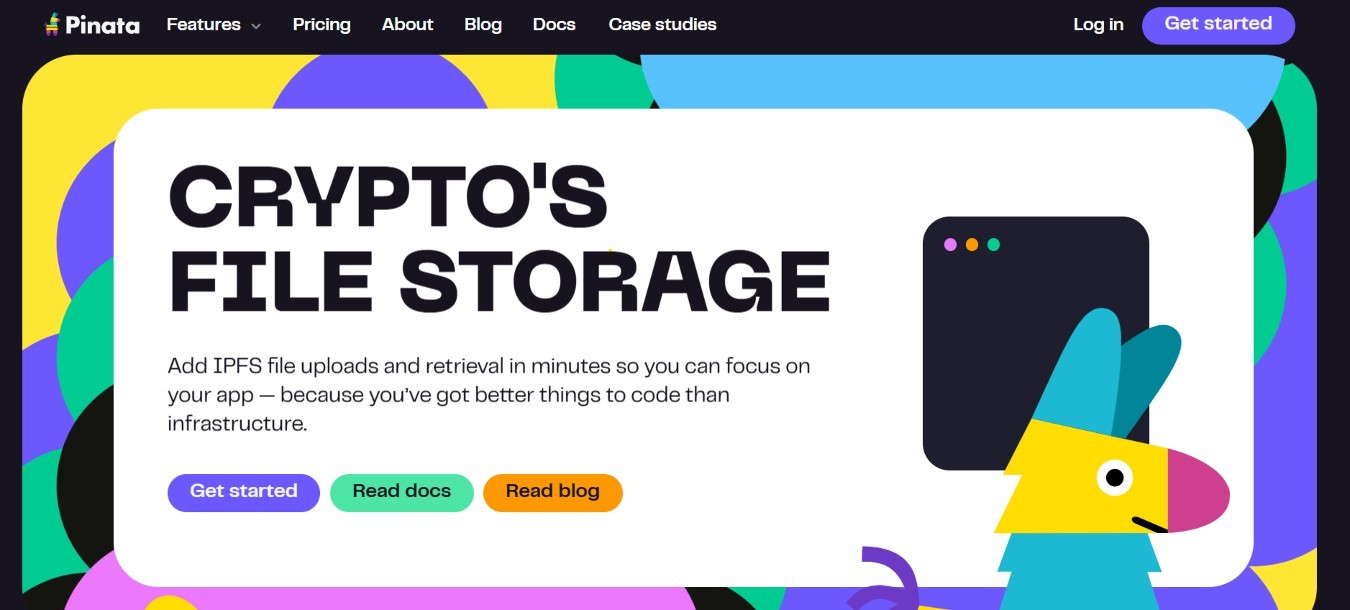
 Handle errors.

 End

**Apparatus/Software Used**:

* Node.js
* Axios
* Dotenv
* Form-data
* Pinata

# Testing Phase:



Step 1 : Open Pinata Website

* Visit piñata.cloud
* Login or sign up with your g mail

Step2: Create api key

* Click api keys options
* Click new key
* Enter key name and click on admin option and click create

Step3 : Open vs code

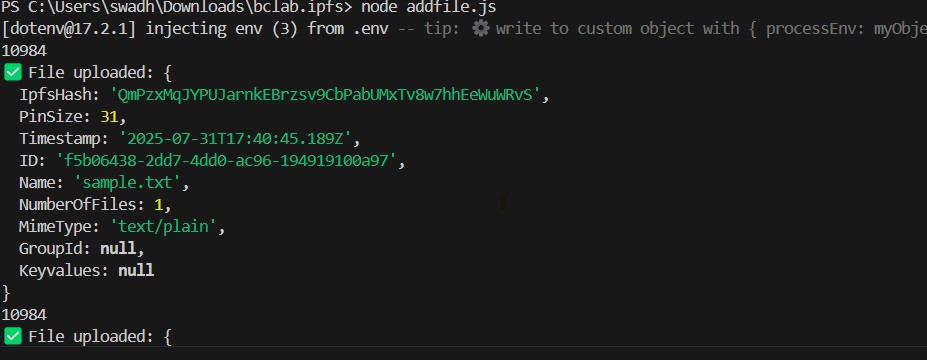
* Open vs code with a folder .
* Create a file name is addfile.js
* Write the code
* Create a other file .env and write the your

-PINATA\_API\_KEY

-PINATA\_SECRT\_API\_KEY

-PINATA\_JWT\_TOKEN

# Implementation Phase: Final Output (no error)



1. Open terminal

•Write code

* + Npm init –y
  + npm install axios dotenv form-data
  + node addfile.js

1. Out put

{

IpfsHash: 'QmPzxMqJYPUJarnkEBrzsv9CbPabUMxTv8w7hhEeWuWRvS', PinSize: 31,

Timestamp: '2025-07-31T17:40:45.189Z',

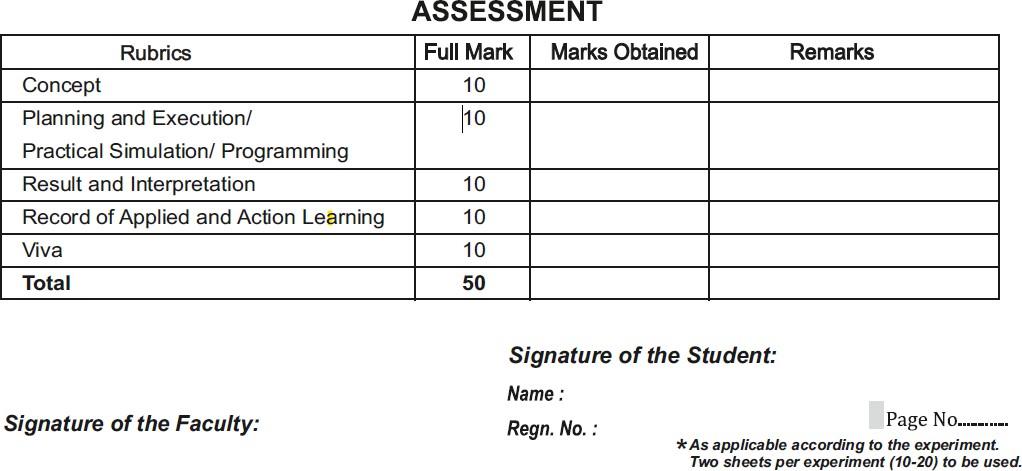
ID: 'f5b06438-2dd7-4dd0-ac96-194919100a97',

Name: 'sample.txt', NumberOfFiles: 1, MimeType: 'text/plain', GroupId: null, Keyvalues: null

}

**Observations**

* File was uploaded successfully to IPFS using **Pinata’s Web UI**
* A unique **CID** was assigned to the file
* Anyone with the **gateway link** can access the file
* No coding or local node was required — just browser + internet
* Very useful for storing and sharing decentralized files

****