



School:Campus:

Academic Year: Subject Name: Subject Code:

Semester: Program: Branch: Specialization:

Date:

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment: Store with IPFS – Decentralized File Upload

***Coding Phase: Pseudo Code / Flow Chart / Algorithm**

Step 1: First we need to set our Pinata API key, secret and JWT

Step 2: Read the file to be uploaded

Step 3: Create a POST request to Pinata's upload API

URL: <https://api.pinata.cloud/pinning/pinFileToIPFS>

Headers:

- pinata_jwt

Body:

- Attach the file as form-data

Step 4: Send the request

Step 5: Receive the response from Pinata

If upload is successful:

- Extract and display the CID from the response

Else:

- Show an error message

Step 6: End

*Software Used

- Node.js
- Npm
- VS Code
- Web browser
- Terminal/command prompt
- NPM packages(Dependencies)
- Pinata - IPFS pinning service to store files on IPFS

Page No.....

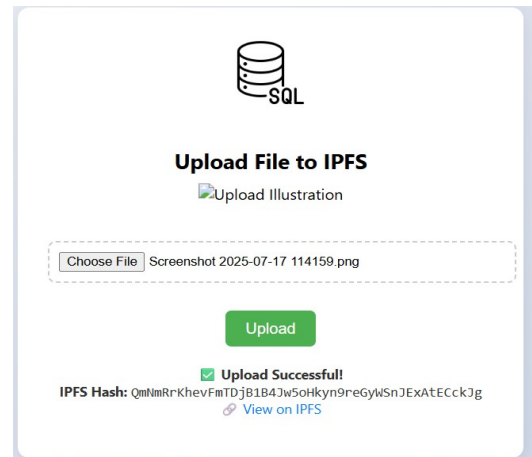
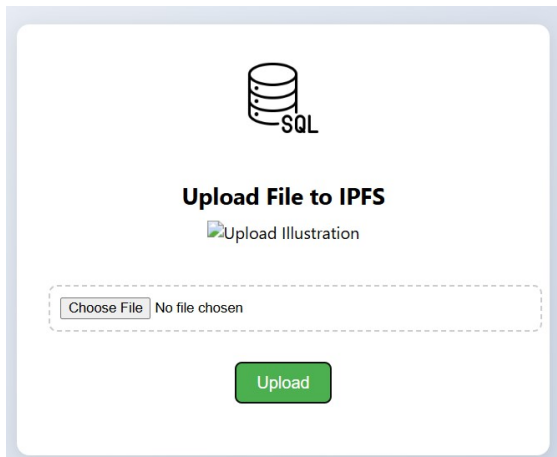
*** As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.**

* Testing Phase: Compilation of Code (error detection)

```
PS C:\Users\asus\Desktop\IPFS\ipfs-frontend> node server.js
[dotenv@17.2.1] injecting env (3) from .env -- tip: ✨ run anywhere with `dotenvx run -- yourcommand`
Server running at http://localhost:3000
AxiosError: Request failed with status code 401
    at settle (C:\Users\asus\Desktop\IPFS\ipfs-frontend\node_modules\axios\dist\node\axios.cjs:2090:12)
    at IncomingMessage.handleStreamEnd (C:\Users\asus\Desktop\IPFS\ipfs-frontend\node_modules\axios\dist\node\axios.cjs:3207:11)
    at IncomingMessage.emit (node:events:530:35)
    at endReadableNT (node:internal/streams/readable:1698:12)
    at process.processTicksAndRejections (node:internal/process/task_queues:90:21)
    at Axios.request (C:\Users\asus\Desktop\IPFS\ipfs-frontend\node_modules\axios\dist\node\axios.cjs:4317:41)
    at process.processTicksAndRejections (node:internal/process/task_queues:105:5)
    at async C:\Users\asus\Desktop\IPFS\ipfs-frontend\server.js:33:22 {
  code: 'ERR_BAD_REQUEST',
  config: {
    url: 'http://localhost:3000/api/v1/keys',
    method: 'get',
    headers: {
      Accept: 'application/json',
      'Content-Type': 'application/json',
    },
    data: undefined,
  },
  request: {
    _readableState: [Object],
    aborted: false,
    timeout: 0,
    headers: {
      accept: 'application/json',
      'content-type': 'application/json',
    },
    url: 'http://localhost:3000/api/v1/keys',
    method: 'GET',
    data: undefined,
  },
  status: 401,
  statusText: 'Unauthorized',
  message: 'Request failed with status code 401'
}
```

```
PS C:\Users\asus\Desktop\IPFS\ipfs-frontend> node server.js
[dotenv@17.2.1] injecting env (3) from .env -- tip: 🐼 observe env with Radar: https://dotenvx.com/radar
Server running at http://localhost:3000
ReferenceError: respond is not defined
    at C:\Users\asus\Desktop\IPFS\ipfs-frontend\server.js:46:55
    at process.processTicksAndRejections (node:internal/process/task_queues:105:5)
```

*** Implementation Phase: Final Output (no error)**



Workspace

Himanshu's Workspace

▼

FILES

PUBLIC

PRIVATE

Public files are accessible via IPFS

tan-tricky-basilisk-330.mypinata.clo...

+ Add

🔍

Search files and CIDs

<input type="checkbox"/>	NAME	CID	SIZE	CREATION DATE	FILE ID
<input type="checkbox"/>	Screenshot 2025-07-17 114159.png	QmHmR...CckJg	311.00 KB	8/1/2025	
<input type="checkbox"/>	sample1.txt	QmXfw...fgrDk	42 B	7/31/2025	
<input type="checkbox"/>	hello.txt	bafkr...dtmve	12 B	7/31/2025	

Rows per page: 10

◀ ▶

IPFS

Files

Groups

Gateways

Analytics

DEVELOPER

API Keys

Webhooks

Access Controls

EXTENSIONS

Marketplace

Integrations

* Observations

- ❖ Upon successful file upload, the server confirms with a clear success notification.
- ❖ The response includes a unique IPFS content identifier (hash), indicating the file is securely stored on IPFS.
- ❖ A direct link to view the file through Pinata's IPFS gateway is provided in the response.
- ❖ The server removes the uploaded file from local storage immediately after pinning it to IPFS.
- ❖ If the upload process encounters issues or no file is selected, the server responds with an error message.
- ❖ The output data is returned as JSON, requiring a frontend interface or API tool for proper visualization.

ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	50		

Signature of the Student :

Name :

Signature of the Faculty :

Regn. No. :

Page No.....

**** As applicable according to the experiment.
Two sheets per experiment (10-20) to be used***