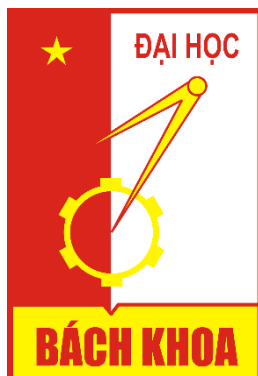


TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



Báo cáo Project I

Đề tài

Xây dựng hệ thống Backup và lưu trữ lâu dài dữ liệu lớn

GVHD: TS. Trần Nguyên Ngọc

Sinh viên thực hiện : Phương Trung Kiên

MSSV : 20183776

Hà Nội , Tháng 10 Năm 2022

I: Mô tả bài toán	Error! Bookmark not defined.
II: Hệ thống tự động hóa kiểm soát luồng dữ liệu	Error! Bookmark not defined.
1. Khái niệm Apache Nifi	5
2. Luồng xử lý hệ thống.....	6
4. Các thành phần trong hệ thống.....	6
III. Hệ thống lưu trữ dữ liệu.....	Error! Bookmark not defined.
1. Khái niệm:.....	Error! Bookmark not defined.
2. Lợi ích đem lại.....	Error! Bookmark not defined.
3. Cài đặt Minio S3.....	Error! Bookmark not defined.
4. Phân quyền người dùng trong S3 minio ..	Error! Bookmark not defined.
5. Vận hành lưu trữ dữ liệu tại các Bucket	Error! Bookmark not defined.
IV. Quản lý hệ thống Backup và lưu trữ.....	Error! Bookmark not defined.
V. Khó khăn và hướng giải quyết	Error! Bookmark not defined.
VI. Kết luận	Error! Bookmark not defined.

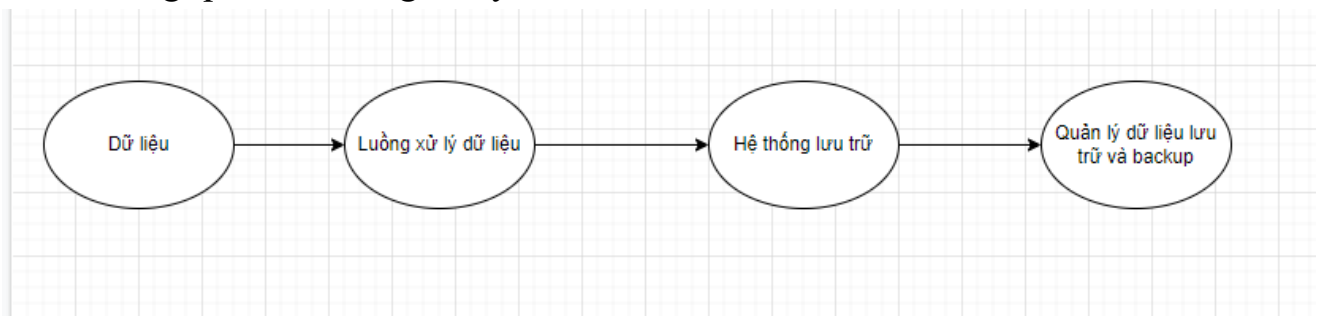
Lời mở đầu

Hiện nay, với sự phát triển của công nghệ thông tin thì khối lượng dữ liệu siêu khổng lồ việc làm sao để quản lý, sử dụng được hết nguồn dữ liệu đó là điều mong muốn của hầu hết các doanh nghiệp. Tuy nhiên ngoài việc sử dụng hợp lý các nguồn dữ liệu lớn thì doanh nghiệp cũng cần phải đảm bảo dữ liệu đó được lưu trữ một cách an toàn và dài hạn để khi cần có thể đưa ra để sử dụng. Vì vậy trong học phần Project I kỳ hè này em đã tiến hành nghiên cứu hệ thống mới để phân nào có thể giải quyết vấn đề lưu trữ kể trên.

I. Mô tả bài toán

Yêu cầu bài toán: Bài toán được đặt ra là đưa dữ liệu vào lưu trữ lâu dài trong hệ thống Object Storage. Loại dữ liệu nào thì được lưu trữ với thời gian như thế nào. Trong quá trình đẩy dữ liệu vào lưu trữ trong hệ thống thì phải biết được dữ liệu có được đẩy thành công hay là thất bại trong quá trình lưu trữ và backup lại dữ liệu trở về. Dữ liệu nào không đúng định dạng phải cảnh báo về hệ thống để đưa ra hướng giải quyết.

- Tổng quan hệ thống xử lý:



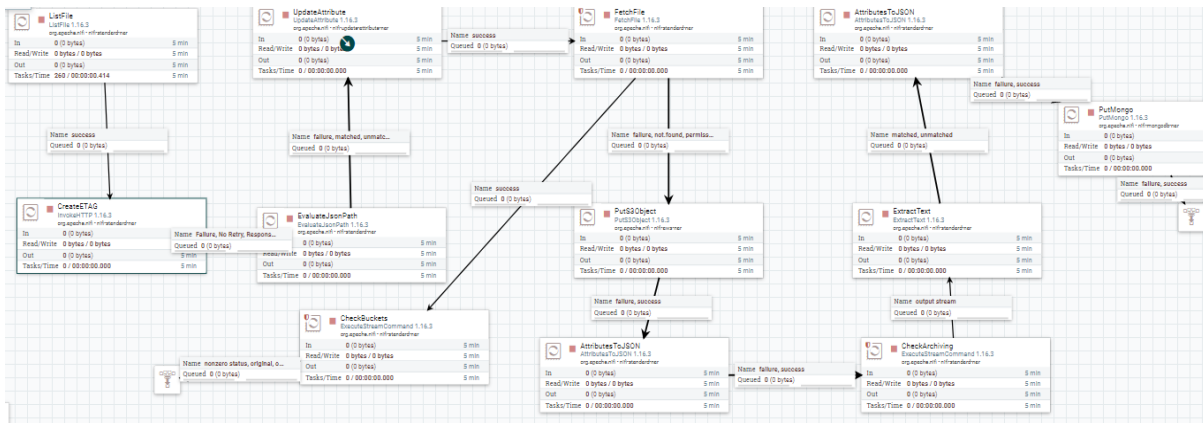
II. Hệ thống tự động hóa, kiểm soát luồng dữ liệu

2.1 Khái niệm Apache Nifi:

- Apache NiFi là một hệ thống phân luồng dữ liệu. Nó hỗ trợ mạnh mẽ cho việc theo dõi, giám sát hoạt động của các luồng dữ liệu, nhờ việc hiển thị một cách chi tiết các thông số trên các biểu đồ theo thời gian thực. Đặc biệt, nó có thể mở rộng về logic, chuyển đổi dữ liệu và định tuyến dữ liệu theo quy mô của hệ thống.
- FlowFile trong Nifi là dữ liệu gốc đi kèm là metadata tương ứng. Nó cho phép người dùng xử lý không chỉ CSV hoặc dữ liệu từ các bản ghi, mà còn cả hình ảnh, video, âm thanh hoặc bất kỳ dữ liệu nhị phân nào khác.
- Mỗi phần "User data" (tức là dữ liệu mà người dùng đưa vào NiFi để xử lý) được gọi là FlowFile. Một FlowFile được tạo thành từ hai phần: Thuộc tính và Nội dung (Attributes and Content). Nội dung chính là dữ liệu người dùng. Thuộc tính là các cặp key-value được liên kết nhằm định danh giữa các dữ liệu người dùng.
- UUID: Universally Unique Identifier giá trị nhận dạng duy nhất giúp phân biệt các FlowFiles khác trong hệ thống.

- filename: Tên tệp mà người dùng có thể đọc được, có thể được sử dụng khi lưu trữ dữ liệu vào đĩa hoặc một dịch vụ ngoài
- Path: Giá trị có cấu trúc phân cấp có thể được sử dụng khi lưu trữ dữ liệu vào đĩa hoặc dịch vụ bên ngoài
- Trong Nifi ta sẽ sử dụng các “Processor” để tạo thành một luồng xử lý công việc. Mỗi một Processor sau khi thực thi sẽ đưa ra kết quả là các Flowfile bao gồm thuộc tính và nội dung của nó được lưu trong hàng đợi của Nifi.

2.2 Luồng xử lý của hệ thống:




- Tổng quan quy trình luồng xử lý:
 - Lấy thông tin về các file trong thư mục lựa chọn theo dõi.
 - Gọi API để tạo các mã HASH tương ứng với từng file đó.
 - Lưu mã Hash đó thành thuộc tính tương ứng của từng file.
 - Sau khi có thông tin file sẽ tiến hành nạp nội dung của file để thu được file hoàn chỉnh.
 - Kiểm tra xem Bucket sẽ lưu trữ các file dữ liệu trong Minio đã tồn tại chưa.
 - Đẩy dữ liệu vào kho lưu trữ s3 Minio với các thông tin phù hợp.
 - Kiểm tra xem dữ liệu đã đẩy vào thành công chưa.
 - Lưu thông tin kiểm tra để dễ dàng theo dõi

2.3 Các thành phần trong hệ thống

a) ListFile

ListFile là một bộ xử lý sẽ theo dõi truy xuất danh sách các tệp trong thư mục nội bộ. Bộ xử lý này chỉ theo dõi và lấy thông tin từ các file trong thư mục mà không xóa bỏ file trong thư mục đó. ListFile sẽ theo dõi thư mục đó, khi có file nào được sửa hoặc có một file mới thì ListFile sẽ cập nhật lên hệ thống. ListFile sẽ đi cùng

bộ xử lý FetchFile để có thể lấy nội dung của cả file đó. Người dùng có thể thêm một số thuộc tính để tùy chỉnh file thích hợp.



ListFile

ListFile 1.16.3

org.apache.nifi - nifi-standard-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

Configure Processor | ListFile 1.16.3

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field


Property	Value
Input Directory	C:\Users\Admin\Downloads\Kien
Listing Strategy	Tracking Timestamps
Recurse Subdirectories	true
Record Writer	No value set
Input Directory Location	Local
File Filter	[\.\.]*
Path Filter	No value set
Include File Attributes	true
Minimum File Age	0 sec
Maximum File Age	No value set
Minimum File Size	0 B
Maximum File Size	No value set

CANCEL

APPLY

b) CreateETAG(InvokeHTTP)

Bộ xử lý invokeHTTP có chức năng dùng để tương tác với các giao thức HTTP.Flow gọi đến các URL sau đó lấy thông về. Trong hệ thống này khi ListFile lấy thông tin về các file rồi gửi lên hệ thống thì invokeHTTP sẽ gọi đến API lấy các thông tin file để tạo các mã hash tương ứng với file trong thư mục đó.



CreateETAG

InvokeHTTP 1.16.3

org.apache.nifi - nifi-standard-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

Configure Processor
InvokeHTTP 1.16.3

Stopped

SETTINGS
SCHEDULING
PROPERTIES
RELATIONSHIPS
COMMENTS


Required field

Property	Value
HTTP Method	GET
Remote URL	http://192.168.56.1:2308/TestHash/?path=\${absolu...
SSL Context Service	No value set
Connection Timeout	5 secs
Read Timeout	15 secs
Idle Timeout	5 mins
Max Idle Connections	5
Include Date Header	True
Follow Redirects	True
Cookie Strategy	DISABLED
Disable HTTP/2	False
FlowFile Naming Strategy	RANDOM

CANCEL
APPLY

c) EvaluateJsonPath

Bộ xử lý EvaluateJsonPath này dùng để ghi nội dung vào thuộc tính của nó hay chuyển tính nội dung thành định dạng Json để dễ dàng xử lý. Trong hệ thống này em dùng với mục đích để chuyển thông tin mã Hash được tạo ra trong flowfile trước đó rồi cập nhật vào nội dung từ từng file.

	<div><div>EvaluateJsonPath</div><div>EvaluateJsonPath 1.16.3</div><div>org.apache.nifi - nifi-standard-nar</div></div>
In	0 (0 bytes) 5 min
Read/Write	0 bytes / 0 bytes 5 min
Out	0 (0 bytes) 5 min
Tasks/Time	0 / 00:00:00.000 5 min

Configure Processor | EvaluateJsonPath 1.16.3

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field


Property	Value
Destination	flowfile-attribute
Return Type	auto-detect
Path Not Found Behavior	ignore
Null Value Representation	empty string
HashValue	\$\$.HashCode

CANCEL

APPLY

d) UpdateAttribute

Bộ xử lý này dùng để cập nhật thuộc tính của flowfile bằng cách xử dụng các thuộc tính cũ có sẵn hoặc do người dùng tự thêm vào. Người dùng cũng có thể thêm một số điều kiện cho phù hợp với từng tính huống cụ thể. Trong hệ thống này em dùng để cập nhật lại mã hash trong từng flowfile.



UpdateAttribute

UpdateAttribute 1.16.3

org.apache.nifi - nifi-update-attribute-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

Configure Processor | UpdateAttribute 1.16.3

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property	Value
Delete Attributes Expression	No value set
Store State	Do not store state
Stateful Variables Initial Value	No value set
Cache Value Lookup Cache Size	100

⚙️

ADVANCED

CANCEL

APPLY

e) FetchFile

FetchFile là bộ xử lý luôn đi cùng với ListFile. FetchFile sẽ lấy thông tin mà Listfile lấy được từ các file trong thư mục theo dõi. Fetchfile sẽ đọc các thông tin rồi nạp nội dung của từng file để thu được nội dung đúng của nó.

Property	Value
File to Fetch	\$(absolute.path)/\$(filename)
Completion Strategy	None
Move Destination Directory	No value set
Move Conflict Strategy	Rename
Log level when file not found	ERROR
Log level when permission denied	ERROR

f) PutS3Object

PutS3Object sẽ đẩy các flowfile vào bucket đã cấu hình tương ứng trong s3. Người dùng cần nhập các thông tin cần thiết để hệ thống có thể kết nối đến kho lưu trữ s3 tương ứng.

Property	Value
File to Fetch	\$(absolute.path)/\$(filename)
Completion Strategy	None
Move Destination Directory	No value set
Move Conflict Strategy	Rename
Log level when file not found	ERROR
Log level when permission denied	ERROR

Configure Processor | PutS3Object 1.16.3

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field


Property	Value
Object Key	\$(filename)
Bucket	bucket08
Content Type	No value set
Content Disposition	No value set
Cache Control	No value set
Access Key ID	Sensitive value set
Secret Access Key	Sensitive value set
Credentials File	No value set
AWS Credentials Provider Service	No value set
Object Tags Prefix	No value set
Remove Tag Prefix	False
Storage Class	Standard

CANCEL

APPLY

g) AttributesToJson

Bộ xử lý AttributeToJSON có nhiệm vụ là sẽ tổng hợp tất cả các thuộc tính của một flowfile thành một chuỗi dưới dạng JSON và lưu trữ trong nội dung hoặc tạo thành một thuộc tính mới ngay trong flowfile đó. Trong trường hợp này thì em sẽ lưu trữ vào nội dung xong sẽ gửi đến bộ xử lý khác và tiếp tục xử lý.

	<div>AttributesToJson</div> <div>AttributesToJson 1.16.3</div> <div>org.apache.nifi - nifi-standard-nar</div>	
In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

Configure Processor | AttributesToJson 1.16.3

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property	Value
Attributes List	No value set
Attributes Regular Expression	No value set
Destination	flowfile-content
Include Core Attributes	true
Null Value	false

CANCEL

APPLY

h) CheckBucket(ExecuteStreamCommand)

Trong bộ xử lý này sẽ chạy môi trường python tại máy chủ cục bộ, kiểm tra xem bucket mà mình muốn lưu trữ đã tồn tại trong hệ thống hay chưa, nếu chưa thì sẽ tạo bucket có tên tương ứng.

CheckBuckets
ExecuteStreamCommand 1.16.3
org.apache.nifi - nifi-standard-nar

In

0 (0 bytes)

5 min

Read/Write

0 bytes / 0 bytes

5 min

Out

0 (0 bytes)

5 min

Tasks/Time

0 / 00:00:00.000

5 min

Configure Processor | ExecuteStreamCommand 1.16.3

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property	Value
Command Arguments Strategy	Command Arguments Property
Command Arguments	CheckBuckets.py
Command Path	python
Ignore STDIN	false
Working Directory	D:\Language\Company\minio-py\TheEnd
Argument Delimiter	;
Output Destination Attribute	No value set
Max Attribute Length	256

CANCEL

APPLY


```

TheEnd > CheckBuckets.py > ...
1  from minio import Minio
2
3  client = Minio(
4      "192.168.56.105:9000",
5      access_key = "admin",
6      secret_key = "password@31",
7      secure = False,
8  )
9  found = client.bucket_exists("bucket08")
10 if not found:
11     client.make_bucket("bucket08")

```

i) CheckArchiving(ExecuteStreamCommand)

Khi lấy được thông tin từ bộ xử lý trước thì bộ xử lý này sẽ đọc các thông tin và kiểm tra xem mã hash mà ta đã tạo lúc đầu tiên có giống với mã ETAG được tạo khi được đẩy lên lưu trữ hay không, lúc đó thì ta có thể biết được hệ thống có lưu trữ thành công chính xxacs dữ liệu hay không.



CheckArchiving
 ExecuteStreamCommand 1.16.3
 org.apache.nifi - nifi-standard-nar

In	0 (0 bytes)	5 min
Read/Write	0 bytes / 0 bytes	5 min
Out	0 (0 bytes)	5 min
Tasks/Time	0 / 00:00:00.000	5 min

Configure Processor | ExecuteStreamCommand 1.16.3

Stopped

SETTINGS

SCHEDULING

PROPERTIES

RELATIONSHIPS

COMMENTS

Required field

Property	Value
Command Arguments Strategy	Command Arguments Property
Command Arguments	CheckFileArchiving.py
Command Path	python
Ignore STDIN	false
Working Directory	D:\Language\Company\minio-py\TheEnd
Argument Delimiter	;
Output Destination Attribute	No value set
Max Attribute Length	256

CANCEL

APPLY

```

#read content from Nifi
a = sys.stdin.readline()
HT=json.loads(a)
client = Minio(
    "192.168.56.105:9000",
    access_key = "admin",
    secret_key = "password@31",
    secure = False,
)
try:
    response = client.get_object(HT["s3.bucket"], HT["s3.key"])
    sEtag=response.__dict__['headers']['ETag']
    if(sEtag[1:(len(sEtag)-1)]==response.__dict__['headers']['x-amz-meta-s3.hashvalue']):
        sys.stdout.write(" successfully "+ HT["s3.bucket"])
    else:
        sys.stdout.write(" Error Archiving "+ HT["s3.bucket"])
    # Read data from response.
finally:
    response.close()
    response.release_conn()

```

III. Hệ thống lưu trữ dữ liệu

3.1 Khái niệm:

- Quản lý dữ liệu dưới dạng object, Mỗi thành phần lưu trữ sẽ bao gồm dữ liệu và một phần metadata.
- Hệ thống lưu trữ Object cho phép người dùng có thể lưu trữ một lượng lớn dữ liệu phi cấu trúc. Dữ liệu thường được lưu trữ như video, ảnh, bài hát,...

3.2 Lợi ích:

- Phân tích dữ liệu lớn hơn
- Khả năng mở rộng vô hạn.
- Truy xuất dữ liệu nhanh hơn.
- Giảm chi phí.

3.3 Cài đặt minio s3

- Khái niệm: Minio là một máy chủ lưu trữ đối tượng triển khai API công khai giống như Amazon S3. Minio có thể được sử dụng để lưu trữ dữ liệu phi cấu trúc như ảnh, video, tệp nhật ký, bản sao lưu và hình ảnh. Kích thước của một đối tượng có thể từ vài KB đến 5TB. Các tệp được tổ chức theo cái gọi là "bucket".
- Cài đặt Minio về máy:

```
[root@linuxhelp opt]# wget https://dl.minio.io/server/minio/release/linux-amd64/
minio
--2017-12-12 06:24:05-- https://dl.minio.io/server/minio/release/linux-amd64/mi
nio
Resolving dl.minio.io (dl.minio.io)... 162.243.132.171
Connecting to dl.minio.io (dl.minio.io)|162.243.132.171|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 21919008 (21M) [application/octet-stream]
Saving to: 'minio'

100%[=====>] 2,19,19,008 915KB/s in 29s

2017-12-12 06:24:35 (744 KB/s) - 'minio' saved [21919008/21919008]
```

- Cấu hình minio trong /etc/default/minio

```
MINIO_OPTS="--address 192.168.56.105:9000 --console-address 192.168.56.105:9001"
MINIO_ROOT_USER=admin

MINIO_VOLUMES="/tmp/minio"
```

- Tạo service file trong system

```
WorkingDirectory=/usr/local/

User=minio
Group=minio

EnvironmentFile=/etc/default/minio
ExecStartPre=/bin/bash -c "if [ -z \"${MINIO_VOLUMES}\" ]; then echo \"Variable MINIO_VOLUMES not set in /etc/default/minio\"; exit 1; fi"

ExecStart=/usr/local/bin/minio server $MINIO_OPTS $MINIO_VOLUMES

Restart=always

# Specifies the maximum file descriptor number that can be opened by this process
LimitNOFILE=65536

# Disable timeout logic and wait until process is stopped
TimeoutStopSec=infinity

# SIGTERM signal is used to stop Minio

SendSIGKILL=no

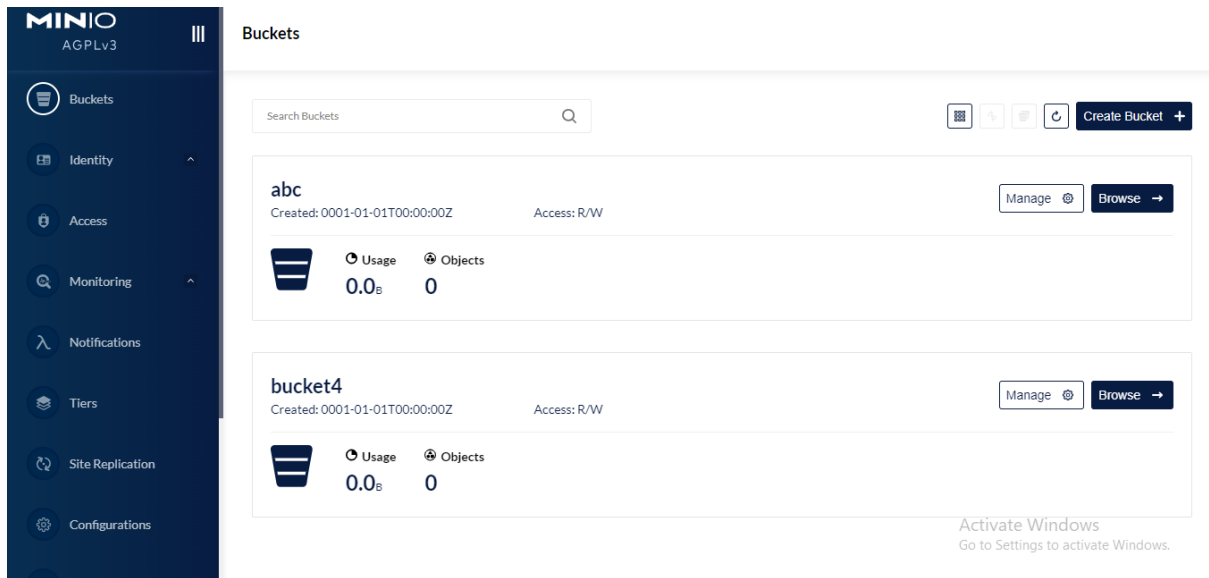
SuccessExitStatus=0

[[Install]]
WantedBy=multi-user.target
```

- Khởi chạy hệ thống Minio:

```
[kien@localhost ~]$ sudo systemctl status minio.service
[sudo] password for kien:
● minio.service - Minio
   Loaded: loaded (/etc/systemd/system/minio.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2022-09-05 11:47:04 +07; 28min ago
     Docs: https://docs.minio.io
   Process: 1176 ExecStartPre=/bin/bash -c if [ -z "${MINIO_VOLUMES}" ]; then echo "Variable MINIO_VOLUMES not set in /etc/default/minio"; exit 1; fi
  code=exited, status=0/SUCCESS
   Main PID: 1192 (minio)
    Tasks: 9
   CGroup: /system.slice/minio.service
           └─1192 /usr/local/bin/minio server --address 192.168.56.105:9000 --console-address 192.168.56.105:9001 /tmp/minio

Sep 05 11:47:20 localhost.localdomain minio[1192]: MinIO Object Storage Server
Sep 05 11:47:20 localhost.localdomain minio[1192]: Copyright: 2015-2022 MinIO, Inc.
Sep 05 11:47:20 localhost.localdomain minio[1192]: License: GNU AGPLV3 <https://www.gnu.org/licenses/agpl-3.0.html>
Sep 05 11:47:20 localhost.localdomain minio[1192]: Version: RELEASE.2022-08-05T23-27-09Z (go1.18.5 linux/amd64)
Sep 05 11:47:20 localhost.localdomain minio[1192]: Status: 1 Online, 0 Offline.
Sep 05 11:47:20 localhost.localdomain minio[1192]: API: http://192.168.56.105:9000
Sep 05 11:47:20 localhost.localdomain minio[1192]: Console: http://192.168.56.105:9001
Sep 05 11:47:20 localhost.localdomain minio[1192]: Documentation: https://docs.min.io
Sep 05 11:47:23 localhost.localdomain minio[1192]: You are running an older version of MinIO released 3 weeks ago
Sep 05 11:47:23 localhost.localdomain minio[1192]: Update: Run 'mc admin update'
```



3.4 Phân quyền người dùng có thể vào minio s3 server:

- Admin có thể tạo các tài khoản cho người dùng khác đăng nhập vào kho lưu trữ các Object với các quyền hạn nhất định do admin quyết định xem.
- Các tài khoản không trong group thì có quyền riêng của nó, nhưng nếu nó thuộc một group nào đó thì nó sẽ tuân theo quyền hạn của group đó

Create User

User Name

asdddd

Password

••••••••



Assign Policies

Start typing to search for a Policy



Select Policy



diagnostics



readonly



readwrite



writeonly

Assign Groups

Start typing to search for Groups




Select Group



Kien123

3.5 Vận hành lưu trữ dữ liệu tại các Bucket trong S3:

- Thời gian lưu giữ:
 - Các dữ liệu được đưa vào các Object sẽ có thời gian lưu giữ nhất định cho đến khi hết thời gian lưu giữ sẽ chuyển sang một giai đoạn khác.
- Sao khi hết thời gian lưu giữ:
 - Dữ liệu sau khi hết thời gian lưu giữ trong Bucket sẽ có 2 lựa chọn:
Dữ liệu sẽ bị xóa đi, Dữ liệu sẽ được chuyển sang một nơi lưu trữ khác

 Add Lifecycle Rule ✕

Type of lifecycle

☒ Expiry ☐ Transition

Object Version

Current Version

After

days

Filters

Cancel

Save

IV. Hệ thống quản lý Backup và dữ liệu đã lưu trữ:

4.1 Hệ thống Backup dữ liệu

Enter your Data

Object

DSC_4680.JPG

Bucket

test123

Folder

C:\Users\Admin\Downloads\DSC_4680.JPG

Submit

Success Backup

- Người dùng nhập thông tin cần thiết bao gồm: Tên Object cần Backup về, Bucket nào chứa Object đó, Thư mục mà người dùng muốn lưu trữ Object đó.

4.2 Quản lý thông tin dữ liệu đã lưu trữ

Data Collection		Backup
Name	Time	Status
21M10.fig	Thu, 06 Oct 2022 09:43:58 GMT	successfully bucket08
DSC_4679.JPG	Thu, 06 Oct 2022 09:43:58 GMT	successfully bucket08
DSC_4702.JPG	Thu, 06 Oct 2022 09:43:59 GMT	successfully bucket08
NifiPython.txt	Thu, 06 Oct 2022 09:44:00 GMT	successfully bucket08
weather3.csv	Thu, 06 Oct 2022 09:44:00 GMT	successfully bucket08
DSC_4680.JPG	Thu, 06 Oct 2022 09:43:59 GMT	successfully bucket08

- Những Object đã được đẩy vào lưu trữ thành công thì sẽ hiển thị thông tin và thời gian đưa vào lưu trữ thành công.

V. Khó khăn và hướng giải quyết

5.1 Khó khăn

- Hệ thống bị lỗi: Mạng bị lỗi, đĩa bị lỗi, phần mềm bị treo.
- Truy cập dữ liệu vượt quá giới hạn cho phép
- Nhận được dữ liệu quá lớn, quá nhỏ, quá nhanh, quá chậm hoặc ở định dạng sai.
- Các hệ thống phát triển với tốc độ khác nhau: Các giao thức và định dạng được sử dụng bởi một hệ thống nhất định có thể thay đổi bất cứ lúc nào và thường không phân biệt các hệ thống xung quanh chúng. Dataflow tồn tại để kết nối một hệ thống phân tán lớn, các thành phần được thiết kế lỏng lẻo hoặc không hoàn toàn để hoạt động cùng nhau.
- Chính sách bảo mật giữa hệ thống với hệ thống, hệ thống với các tương tác của người dùng phải an toàn, đáng tin cậy.

5.2 Hướng giải quyết

- Kiểm soát dữ liệu trước khi đưa vào lưu trữ.
- Trong quá trình xử lý dữ liệu tiến hành phân loại xử lý qua các thông tin, định dạng dữ liệu cho phù hợp để lưu trữ.
- Những hệ thống có bảo mật thì cần chú ý để giúp cho hệ thống được an toàn trong quá trình lưu trữ và thu thập xử lý dữ liệu.
- Lựa chọn phương thức, hệ thống một cách hiệu quả nhất để có thể thu được kết quả tốt nhất trong hệ thống.

• VI. Kết luận

Qua việc thực hiện các buổi báo cáo project I đã giúp em hiểu và học thêm được rất nhiều kiến thức để có thể phân tích và hiểu hơn được hệ thống mà em muốn tìm hiểu và làm. Tìm hiểu được thêm các kiến thức mới để giải quyết các nghiệp vụ mà hệ thống cần. Cũng cố, ôn lại cũng như hiểu được sự phối hợp giữa những kiến thức của các môn học đã và đang học ở trên trường, ... Và từ đó thực hiện triển khai hệ thống của bản thân. Cuối cùng, hiểu được để xây dựng được một hệ thống chạy được ổn định trong thực tế thì phải giải quyết nhiều vấn đề và trong đó việc phân tích và hiểu được hệ thống ban đầu có vai trò rất quan trọng.

Em xin cảm ơn thầy Trần Nguyên Ngọc đã tận tình hướng dẫn, giúp đỡ, đóng góp ý kiến để em có thể hoàn thiện và hiểu hơn về hệ thống “Backup và lưu trữ lâu dài dữ liệu lớn”.