POC - Media Shuttle

Overview

The **Ames smart flow process** automation, Planned directory structuring, API integration, and download-upload process automation. we were initially planned with *`S3`* storage for upload and download and the process should be faster than we have in the system. As the discussion with the IT team, they have informed the media shuttle storage system. It supports LocalFileSystem and cloud services as well as is cost-effective.

Areas to test

1. able to upload files and folders by manual
2. able to download files and folders by manual
3. able to upload files and folders by automation
4. able to download files and folders by automation
5. able to do versioning on each upload either manual or automation
6. able to check upload process completion state
7. able to copy/move files or folders one to another
8. able to transfer file within system through REST API
9. able to register a user/email through REST API
10. able to check history of package id through REST API
11. able to check current active transfers through REST API
12. able to create a new portal through REST
13. able to set user rols or persmission

Exercises

Point 1 - manual upload

> Yes, can upload files and folders manually by creating an upload request (link)

> Steps

1. create an empty package id
    1. required fields are `portal_id`
2. create an upload URL
    1. required fields are
        1. `portal_id`
        2. `package_id`
        3. grants `upload`
        4. `email`
    2. optional fields are
        1. `destinationPath`
        2. `webhook`
3. conditions
    1. `email` should be registered/activated on the media shuttle portal
    2. cannot create multiple `upload` requests on a single package id

3. the user has to login into the portal to upload files
4. Once the link has been used it expires automatically
5. Once files are uploaded the files will be placed under `destinationPath` if given in the payload otherwise it is placed as `root path`
6. If the same files are present inside the directory then they will be overwritten

---

Point 2 - manual download

> Yes, we can download files and folders manually by creating a download files request(link)

> Steps

1. create a download URL
    1. required fields are
        1. `portal_id`
        2. `package_id` - existing
        3. grants `download`
        4. `email`
    2. optional fields are
        1. `webhook`
2. conditions
    1. `email` should be registered/activated on the media shuttle portal
    2. can create multiple `download` requests on a single package id
    3. the user has to login into the portal to upload files
    4. Once the link has been used it expires automatically
    5. the package id's associated files are only displayed on the download window

Point 3 upload files by automation

> No, there is no option to upload a file directly through the rest API.
>
> But, still there are possibilities.

> Steps

1. move or copy the files directly through the SCDX server path.
2. have to execute a script inside the fileServer

Import:

2. has to associate files into a `package-id`. Then can share these files at in download link
3. If the same files are present inside the directory then they will be overwritten

Point 4 download files by automation

> No, there is no option to upload the file directly through the rest API.
>
> But, still there are possibilities.

> Steps

1. move or copy the files directly through the SDCX server path.
2. have to execute/run a script inside the fileServer

Point 5 file versioning

> No, there is no option to file versioning each upload on the same file it will overwrite the existing one

Point 6 Process state

> Yes, can receive process state on success as well as failure cases through `webhook`

> Steps

1. have to set a `webhook` on `upload` or download request link

Point 7 copy folder to folder

> No, there is no rest API found for folder copy but, can do it by manually.

Import:

1. have to associate files into a `package-id`.Then can share these files at in download link

Overall Result

S.No	Task	Status
1.	able to upload files and folders by manual	**Yes**
2.	able to download files and folders by manual	**Yes**
3.	able to upload files and folders by automation	**Yes**
4.	able to download files and folders by automation	**Yes**
5.	able to do versioning on each upload either manual or automation	**No**
6.	able to check upload process completion state	**Yes**
7.	able to copy/move files or folders one to another	**Yes**
8.	able to transfer file within system through REST API	**No**
9.	able to register a user/email through REST API	**Yes**
10.	able to check history of package id through REST API	**Yes**
11.	able to check current active transfers through REST API	**Yes**
12.	able to create a new portal through REST	**Yes**
13.	able to set user rols or persmission	**Yes**

Download and Upload Status

S.No	Type	Size	Mode	Duration (Minutes)

S.No	Type	Size	Mode	Duration (Minutes)
1.	Upload	250 Mb	Without - Agent	0:02:17
2.	Upload	250 Mb	With - Agent	0:01:44
3.	Upload	3.5 Gb	Without - Agent	0:22:01
4.	Upload	3.5 Gb	With - Agent	0:20:01
5.	Download	250 Mb	Without - Agent	0:03:17
6.	Download	250 Mb	With - Agent	0:02:17
7.	Download	3.5 Gb	Without - Agent	0:14:50
8.	Download	3.5 Gb	With - Agent	0:14:01

Conclusion

The download and upload rates not very huge differences.

Note:

- email not received even shared to upload request.
- cannot create a folder as we discussed in the plan
- cannot create a link more than once for a packageId
- cannot upload a file on the Linux platform. Due to there being no signiant app for Linux