# **Functionality Test**

# **Bot 1 Test Cases**

## File Summarization & Checking

### **1. Ensure Bot1 accurately identifies and summarizes all files in the specified DCS workspace**

**Initial Display of the tab** “ToBeTransferredToLocalDrive” in the excel known as FileAddToRegistry.

This excel can be found in C:\NYCFolderList

A screenshot of a phone

Description automatically generated

The “ToBeTransferredToLocalDrive” Tab should be empty at the start.

The NYC-UAT folder in your DCS workspace will have test files loaded (you may do a quick check).

Run bot 1 and there should be a prompt saying that bot 1 has completed.

**Expected Result:**

**Did you see the “Expected Result” as shown above?**

Yes

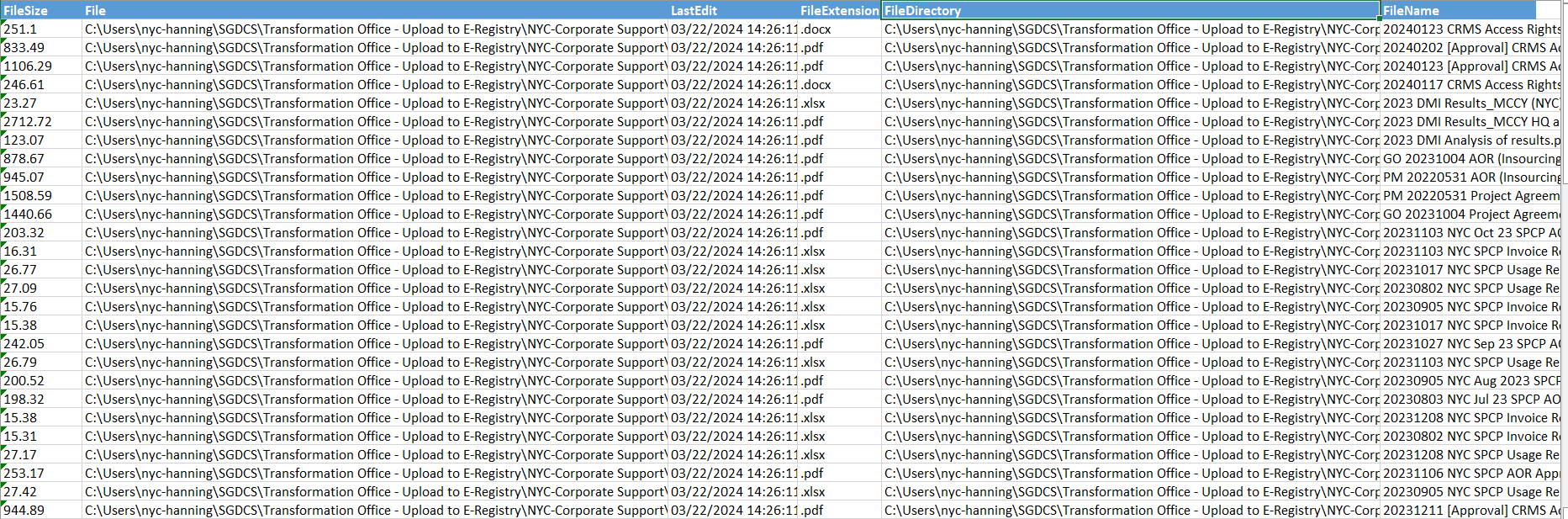
No

### **2. Ensure Bot1 accurately identifies and summarizes all files in the specified DCS workspace**

A screenshot of a phone

Description automatically generated**Initial Display before we run bot1 was blank like the picture below:**

However, we have run bot one already. Hence, we should see the result like the one below



**Expected Result:**

The summary Excel sheet ("ToBeTransferredToLocalDrive" tab) should list all relevant file details accurately. Data includes file size, File Path, Last Edit, File Directory & File Name

**Did you see the “Expected Result” as shown above?**

Yes

No

### **3. Validate that Bot1 accurately identifies and summarizes all files in the Log tab**

A screenshot of a phone

Description automatically generated**Initial Display:**

The “Log” Tab should be empty or filled according to the files scanned the previous month

**Expected Result:**

A screenshot of a computer

Description automatically generated

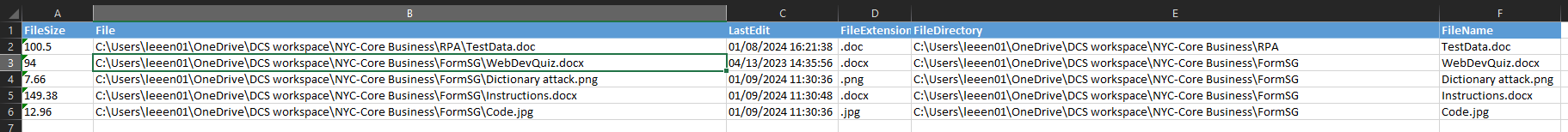
The summary Excel sheet ("Log" tab) should inform us that the bot1 run is complete with xx number of files transferred as shown above.

**Did you see the “Expected Result” as shown above?**

Yes

No

### **4. Validate that Bot 1 distinguishes between file types that should be transferred to Local Drive and those that should not be transferred (e.g a video file and wrong naming convention)**

**Expected Result:**

By giving Bot1 a variety of files and conditions, the "ToBeTransferredToLocalDrive" and "NotToBeTransferred" tabs should reflect accurate categorization based on conditions in the reason Column. Do a quick count to see if the number of files in the to be transferred tally with the information in the (“log” tab) of “filesAddToRegistry” excel.

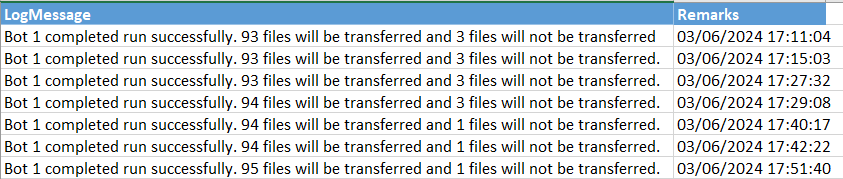
**Did you see the “Expected Result” as shown above?**

Yes

No

### **5. Validate that when Bot1 completes its run, it will update the “Log” tab in the “filesAddToRegistry” excel spreadsheet with the date of execution**

**Expected result:**



When bot 1 runs, it will check against the number of files in all the folders and tabulate the total that will are to be transferred and those that are not to be transferred and this will be logged under the log tab along with today’s date as well as time that the log was generated.

**Did you see the “Expected Result” as shown above?**

Yes

No

## File Upload Check

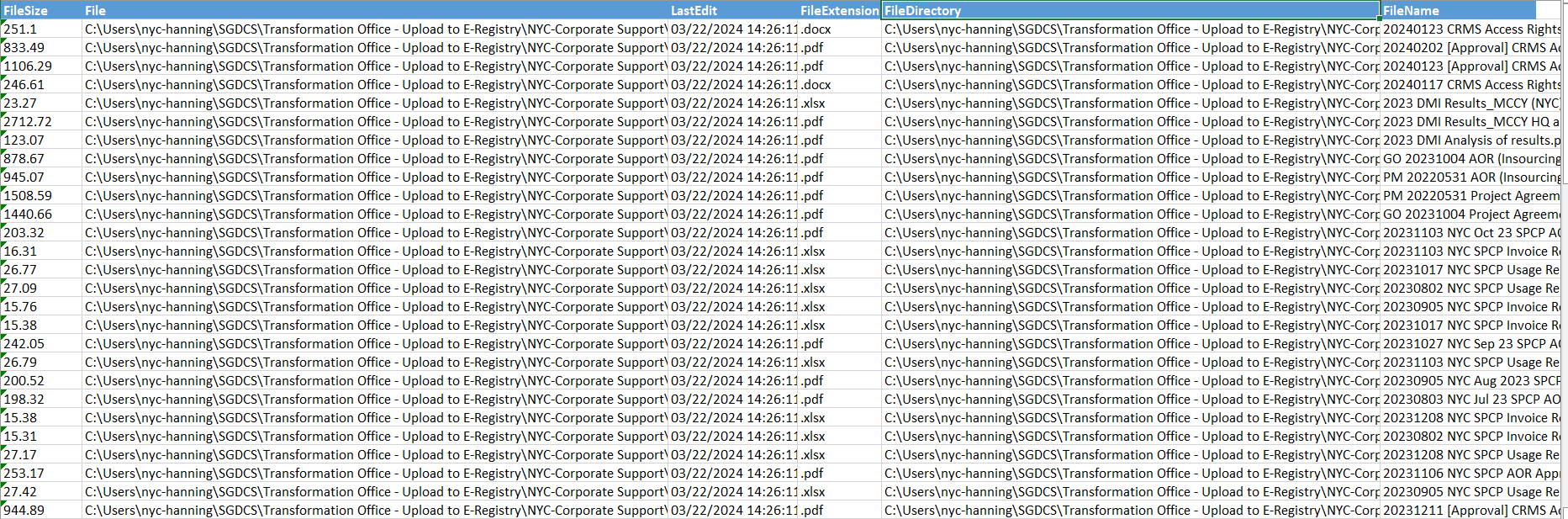
### **6. Confirm that Bot1 do checks the file sizes against NYC Requirements of approved files**

**Initial Requirements:**

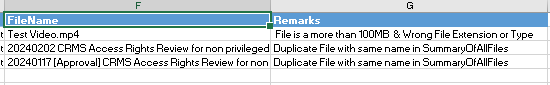
The following is an example of some files in NYC-Corporate Supports' Sub-directories.

A white text with black text

Description automatically generated

Include files with various formats and sizes, both acceptable and unacceptable. In this case, we have files in NYC-Corporate Supports’ Sub-Directories.

**Expected Result:**



The info “NotToBeTransferred” Excel Sheet should list all the rejected files with some remarks

**Did you see the “Expected Result” as shown above?**

Yes

No

### **7. Rename the file extension of any document. (eg. changing from .docx to .mp4, or simply adding a file with the file extention that is not allowed in UiPath. It will still block that file based on the list of extension that are disallowed in eRegistry.**

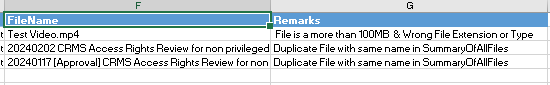
Initial Result:



Add a file above that has MP4 as its extension into the Folder within DCS workspace as shown above or u can choose to change any particular file present from docx to MP4. Below is a zip file that contains the Test Video shown above as well as a file that is a duplicate within the summary which u can use to test for duplicate file checks later on in bot2.



Expected Result:



“Test Video.MP4” would be filtered out in the "NotToBeTransferred" tab. However if u changed the extension of any file present instead u should see that it is no longer is present in the “ToBeTransferredToLocalDrive” Tab.

**Did you see the “Expected Result” as shown above?**

Yes

No

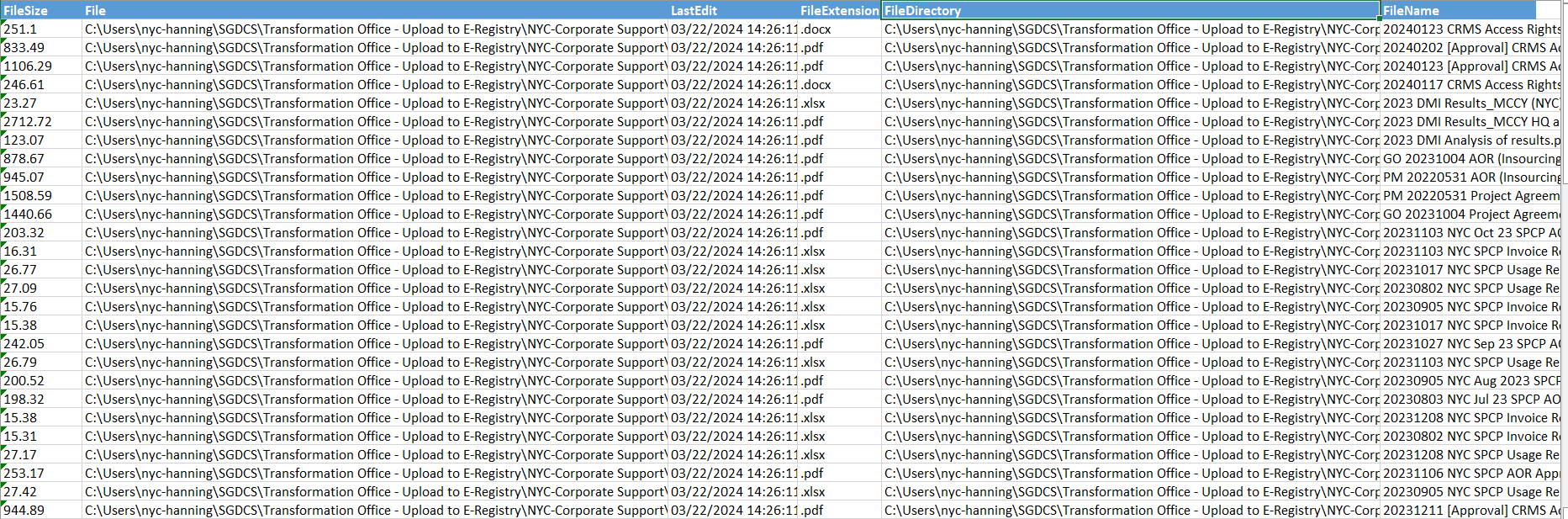
# Bot 2 Test Cases

## Running the bot more than once

### **8. If user press on the bot2 twice accidentally to check which moved files from Cloud to local**

Initial:

Run Bot 1 and you will see all the records of the files in current DCS workspace as shown in the illustration below. Then, run Bot 2. Try running bot 2 and you should see the expected results below.



Expected Result when running bot 2 again:

A screenshot of a chat

Description automatically generated

The User would receive a prompt that Bot 2 has been run twice and will not try to move any files from the DCS workspace to the localDrive again. It will inform user to complete the workflow with Bot3 first.

**Did you see the “Expected Result” as shown above?**

Yes

No

## File Processing and Update

### **9. Validate that Bot2 accurately updates the summary table with the processedFlag.**

Expected Results:



Files under the “TobeTransferredToERegistry” would be moved to the local drive and have the “processedFlag” and “Remarks” columns updated according to which folder the files have been moved to. (1 for processed, 0 for not processed).

User should check to see if the list for files from “ToBeTransferredToERegistry” tab will now appear in **“FilesMovedToLocal” tab.**

**Did you see the “Expected Result” as shown above?**

Yes

No

Files removed by user after we run Bot1, and before we run Bot2

**10. If a user accidentally deletes their file while we are processing with Bot2, Bot 2 will update the “ProcessedFlag” and “Remarks” column to show that file does not exist as it has been moved**

Initial preparation for this testing:

Run Bot1 with any file in DCS workspace. Then, remove any file in the DCS Workspace. Then run Bot2.

A screenshot of a computer

Description automatically generated

“Week 2” is deleted from the FormSG Folder

Expected Result:

You will see the remarks and processedFlag in the “FilesMovedToLocal” tab as shown below

A screenshot of a computer

Description automatically generated

The deleted file would be shown to the user as “File Does Not exist” and a Processed Flag of “0” as the file has not been moved by Bot2. This will help RM to inform any user to include the document it is required to be filed. Otherwise, Bot3 will only process the remaining files.

**Did you see the “End Result” as shown above?**

Yes

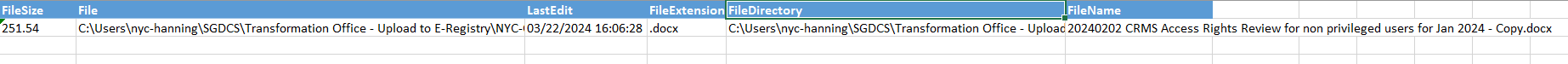
No

## Interaction between Bot2 and Bot 1 Data

### **11. Ensure that Bot 2 interacts with the data provided by Bot 1**

**Initial Preparation:**

**Put in any 1-5 testing documents (you may name them anything or follow the example below) into the DCS Workspace. Then run Bot 1. You will see all the records of files in DCS Workspace + the 5 additional random documents.**

****

Just for this UAT purpose, remove all the records other records captured by Bot1 and keep only the 1-5 documents you created for testing in “ToBeTransferredToLocalDrive” tab.

In this example, 1 Files are to be moved from the “ToBeTransferredToLocalDrive” tab which is the data created by Bot 1

Expected Result:

Run Bot 2. Bot 2 will then cross-reference the data from Bot1 and move the same number of files which in this case/example, is 1 files into the NYCLocalDrive.

A screenshot of a computer

Description automatically generated

Open up the C:\NYCLocalDrive and check if the 1 files has physically been moved.

**Did you see the “Expected Result” as shown above?**

Yes

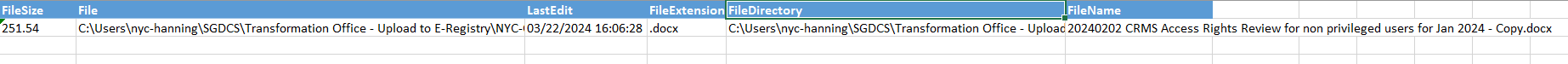
No

## Handling File Details

### **12. Validate that Bot2 documents all the file details (e.g FileName, FileSize, LastEdit, FileExtension, FileDirectory) accurately in the "FilesMovedToLocal" tab.**

Initial preparation:

After we have run the earlier test case and confirms that the files have been moved by Bot2. You may now check the details of each file, e.g file name extention, file path/filedirectory and the extension. See if the details are correctly put into the "FilesMovedToLocal" tab

****Expected Results:

A screenshot of a computer

Description automatically generated

The "FilesMovedToLocal" tabs should display consistent and accurate file details processed and moved files. E.g All the files in “ToBeTransferredTo” tab created by Bot1 should now all be moved to the “FilesMovedToLocal” tab.

**Did you see the “Expected Result” as shown above?**

Yes

No

## Adding extra files into the DCS Workspace

### **13. Run Bot1 and let it record the list of files to be transferred. Once you see the list like the example below, try to add additional new file into the DCS workspace. Then, run Bot2 and see if the new files that we last minute added will be moved to the NYCLocalDrive.**

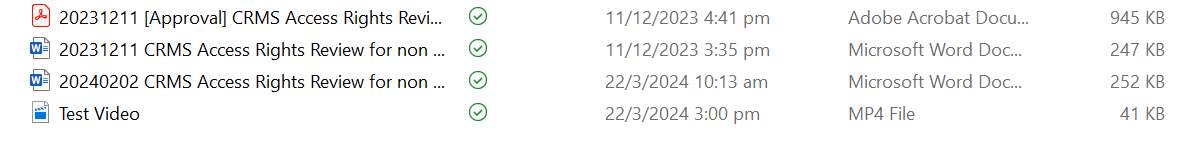
Initial Display and preparation:

A close-up of a number

Description automatically generated

For instance, we have added an Extra File “20240202 CRMS Access Rights Review for non privileged users for Jan 2024” to the C:\Users\nyc-hanning\SGDCS\Transformation Office - Upload to E-Registry\NYC-Corporate Support\IT and Digitalisation\Systems\Customer Relationship Management System (CRMS) after we have run Bot1. We know that this is added last minute because the details of this file is not shown in “FilesToBeTransferred” tab. run Bot 2

Expected Result:



As u can see from the above picture the previous folders have been moved and u can verify this by the date of the other 3 files still present other than “20240202 CRMS Access Rights Review for non privileged users for Jan 2024”

The system should behave in a way that the newly added File should be left inside the DCS workspace folder and will not be moved to the NYC local drive during this batch of processing. It will be moved in the next batch. This is to ensure a clean and clear cut of based on the monthly batch processing requirement.

**Did you see the “Expected Result” as shown above?**

Yes

No

### File Logging

**14. Number of successful files should be documented correctly**

Expected Result:



When bot2 is ran, it should accurately tabulate the total files that are to be moved from DCS workspace over to NYCLocalDrive. This means that only the files within the excel tab of “FilesMovedToLocal” with a ProcessedFlag of “1” will be recorded under the “Log” tab as transferred any of those with a “0” will not be

**Did you see the “Expected Result” as shown above?**

Yes

No

### Seamless file transfer

**15. Dealing with “Hidden” as well as “Read-Only” Files**

Initial result:

A number of numbers on a white background

Description automatically generated

As seen in the image, one file is hidden in DCS workspace before the transfer

Expected Result:

A screenshot of a computer

Description automatically generated

When Bot2 is ran it should document the transfer of the hidden file under FileAddToRegistry Excel Spreadsheet under the “FilesMovedToLocal” Tab and when u open DCS workspace it should be moved over to NYCLocalDrive still as a Hidden file/Read-Only file

**Did you see the “Expected Result” as shown above?**

Yes

## No

# Bot 3 Test Cases

## UI Automation for Bulk Transfer (Root folders only)

### **16. Monitor the bot as it scans the root folders. The bot should look through every root folder, if there are files inside, it will copy ALL files and paste in the eRegistry afterwards.**

### Expected Result:

A screenshot of a computer

Description automatically generated

### **17. At the end of the bulk transfer, the bot should wait for all the bulk files to be moved to eRegistry before carrying out the next step.**

### Expected Result:

### A screenshot of a computer Description automatically generated

### The bot waits for all the files to be transferred over to the eRegistry before moving on.

## UI Automation for Single Transfer (After bulk transfer is completed)

### **18. Verify that the bot only uses single transfer for stray files outside of root folders. Verify that the bot identifies and selects only the files intended for eRegistry**

### Expected Result:

### 

### The bot will search for files that are to be transferred to eRegistry 1 by 1. All stray files will be transferred over with this approach.

## UI Automation when typing the eRegistry file path

### **19. Ensure that Bot 3 successfully launches and interacts with the file explorer interface and ERegistry OneDrive Folder**

Expected Result:

A close-up of a computer screen

Description automatically generated

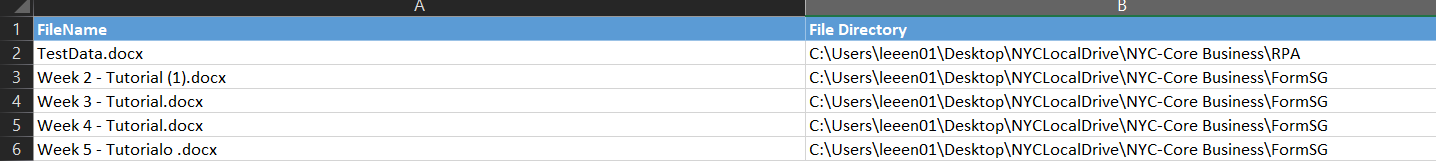
A screenshot of a computer

Description automatically generated

## Handling Timeouts and Unique Scenarios

### **20. Validate the handling of timeouts and other unique scenarios during UI automation.**

Expected Result:



“fileNotMovedToReg” Sheet will contain the files that are not moved when there is a runtime error or any scenario that might happen during the automation

## Documentation and checking of Unmoved Files

### **21. Confirm that Bot 3 checks if there are any files present in the folder and provides a report of how many files have been moved at the end and how many files are not transferred.**

Expected Result:

@[Bot3\_Demo1.mp4](https://drive.google.com/file/d/1H491pRvr8cAH_AKtZOUn5GNIs3UhBWGS/view?usp=drive_link)

Checking if folder is empty

@[Bot3\_Demo2.mp4](https://drive.google.com/file/d/1H7fF2wX9XOwlAyDujHRnucY2c5WMrtmw/view?usp=drive_link)

Provide Documentation in the Logs and Gives User a report at the end

## Unforeseen circumstances – eRegistry Glitch

### **22. Validate the handling of glitch where a popup is opened when typing the file path of the eRegistry**

### Even if the glitch occurs, for the bulk folders, the bot will continue with the grab method to copy each of these files one by one to the eRegistry. Confirm that at the end of the run, all files are transferred.

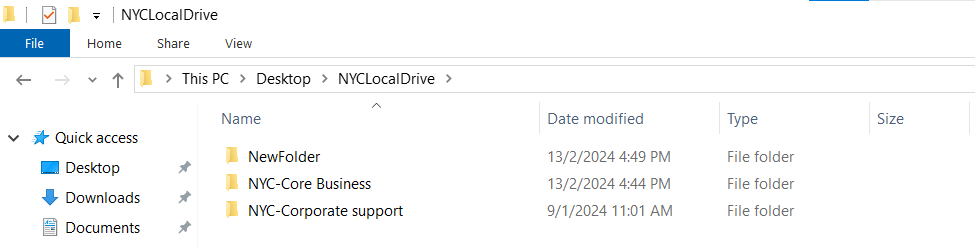
### Expected Result:

### A screenshot of a computer error message Description automatically generated

## Dynamic file pathing

### **23. Ensure that if there are new folders in the file path of the E-Registry, the Bot 3 will change the directory according to the new folders and move the files accordingly**

Initial Result:



“NewFolder” & Files are added inside the Local Drive to test if the Dynamic File Path would work

Expected Result:

@[Bot3\_Demo3.mp4](https://drive.google.com/file/d/1HHK95pnkvupYEo_fMQ3gjDjQZUDqIs3M/view?usp=drive_link)

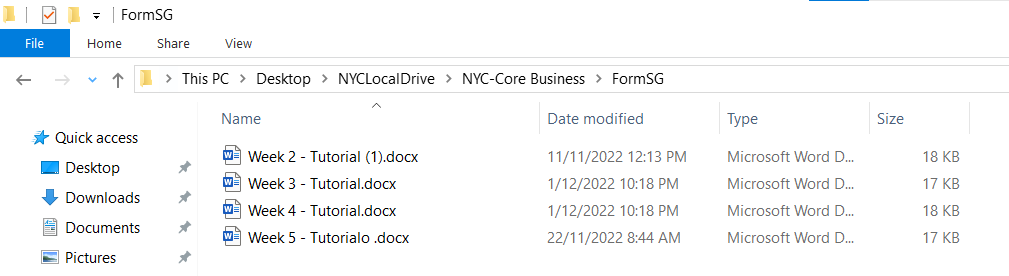
The Bot 3 will automatically move folders to the NewFolder and files accordingly

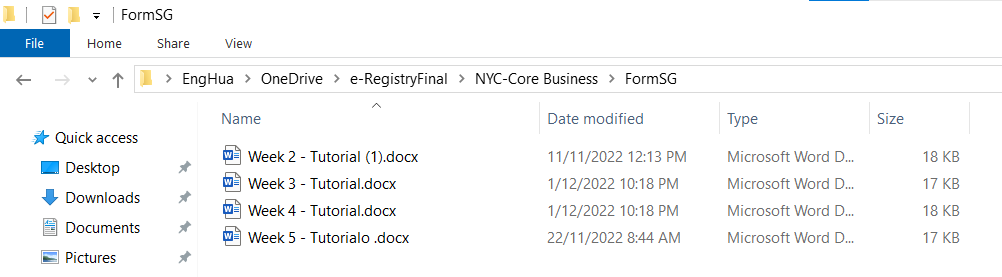
## 

## Duplicate Files in final E-Registry (This is unlikely to happen as NYC already have data governance to ensure no duplicate file name)

### **24. When Bot 3 uses CV to move files from location x to location y, if there is duplicate file present (file A exists in both location x and location y), the user will be prompted to either replace or skip the file**

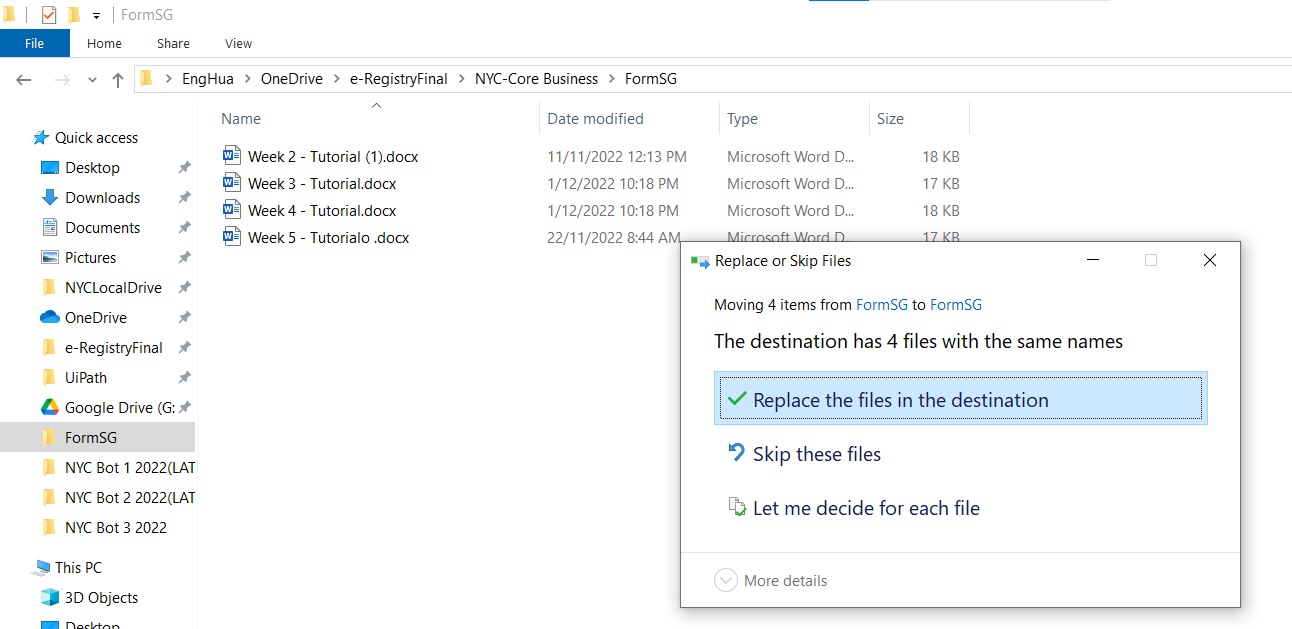
Initial Result:





Files are duplicated in both the Local Drive and the OneDrive E-Registry

Expected Result:



@[Bot3\_Demo4.mp4](https://drive.google.com/file/d/1HT1n07YaXK1lmBcnsxvncAeRmfXUDMNI/view?usp=drive_link)

File explorer would prompt the user if they want to replace the duplicate files or let it remain as it is.

### 

### 

## 

## 

## 

# **Load Test**