5/14/19

Work on switching Smartsheet code from Python to Java.

Issue with string input for path, backslashes aren’t allowed. Cannot solve using same thing as Python.

Manually fixed string and moved on to iterating through image folder to attach to rows.

Issue with for loop, for loop must be set up differently from python for loop

Can get images to upload to rows in the right order.

Still need to:

* Only upload the right amount of images
* Be able to select start row
* Be able to select whatever path you want
* Work on front end and user interface

5/15/19

Continue work on Smartsheet API in Java

* Able to upload batch images
  + Must manually write information into code
  + Cannot prompt user through console or GUI
* Issues
  + May need to “Package and Distribute” even for java code
    - Easier to do with Java than with Python
  + Cannot get python script to work as a standalone .exe file
  + Java code is not completed, and is still very unfamiliar to use for me.
  + Works for very specific cases (Unless code is changed manually)
    - Cannot choose start row
    - Cannot change path
    - Cannot change API Token
    - Cannot change Sheet ID
* Pros of keeping it a python script
  + Once everyone has python and knows how to install additional modules, any script given to them can be ran easily. More scripts can be created, more comfortable with Python.
* Decided to continue work on Python script and stop working on Java program

5/16/19

Work on install documentation for python script

Work on possibly implementing Ian’s work in dynamo into its own script, or into the smartsheet row attachment script.

* Beginning of Navisworks script come from the smartsheet attachments script.
  + Copy and paste sections needed to access sheets, copy and paste GUI and change prompts
  + Store information in new variables
  + Import XML file and parse through viewpoint names and comments
    - Ian has been working on this script. GUI will be implemented once he is done.
* Goal is to create new rows based on the information given by the XML file.
  + Information to include:
    - Viewpoint description
    - Comments
    - Status
* Could possibly take exported images and rename them based on cells in row (such as description or comment)
  + Would need to rename with the row number in front of any information to keep order
* Ian is working on his portion of automation outside of dynamo, will work on implementing both scripts together when Ian has completed what he wants to do.

5/31/19

Before R&D Meeting work on creating new rows and uploading information to those rows. This is a good starting spot for when Ian gets his script working and can pull information to upload from the Navisworks XML file.

6/5/19

* Work on making GUI more user friendly
  + Addition of drop-down list for all sheet name
  + Attempting to create tree list to let users see organization of workplaces, folders, and sheets
  + Added progress meter that updates with every attachment, so it updates every 5 seconds
  + Progress meter takes the place of the input window, so it no longer stays open and goes into “not responding” mode while the files are being attached.
* Having trouble getting all the information needed
  + Need to get all workspaces. Inside of the workspaces I need to get all its folders and sheets. If it has folders, I need to get what’s inside of those. The potential complexity of the workspace and folder structure makes it very difficult to make a catch-all script.
  + The problem with just letting you pick a sheet from a list is that there could be too many sheets to go through, and there could be duplicate sheets.
  + Being able to choose a sheet while seeing where it is located would be very helpful.

6/25/19

XML File Parsing

* Currently have it working so it can pull information from a specific XML file and put that info into a new excel file
* User can use file browse to select their XML file, folder browse to choose destination of file, and enter the name of the file (They do not have to put .xml at the end of the file name)
* Tested with even more folders and it did not work. It got the names for some of the folders, but not all. It also did not populate the excel file properly.
* Trying to use recursion to go indefinitely through an element tree
* Recursion seems to be partially working. Problems with it printing ‘None’ repeatedly for every element that does not have a name attribute. Need to create ‘if’ to not ‘None’
* Able to get it to only print elements with names, now everything is printing.
* Everything seems to be in the right order, I just need to structure it to properly export all the information to excel
* The way that the recursion works makes it go through every possible element that has a name attribute but because it’s basically an infinite loop, there is no way to differentiate each element name to create a string for each path down the tree.
* Recursion is the best way for the script to work with any folder structure, the other methods I have tried always throw exceptions

**July:**

* Got recursion to work with XML file
  + Utilized functions, objects, empty arrays
    - Functions used to control information and processes
    - Objects used to store information for each “cell” going into excel
    - Empty arrays used as arguments to save information during recursion
* Put all of it together in one program
  + Needed to have 3 main operations
    - XML to Excel
    - File Rename
    - File Upload
  + Created new functions for each operation and called those functions depending on input from user
    - Still want to make GUI work better
  + Still want to distribute program better
    - Web app?
    - AWS or Google Web App?