## YouTube Scraper Automation

**Process Description**

* This automation prompts a user to enter a set of keywords and filters to be used for scraping YouTube. The YouTube results are filtered, scraped, then extracting into a local flat file which is then uploaded to an Orchestrator Queue as a Transaction. The Transaction is then downloaded from the Queue and stored as a log file, and finally emailed to the user with summary statistics about items in the Queue.

The following 6 Sequences are used in this automation:

* Main.xaml: The Entry Point workflow that executes each subworkflow
* Enter\_Form\_Process.xaml: Prompts user to fill out a form to be used as the Search Input criteria for YouTube, gets the email address to send the automation results to from UiPath Orchestrator, and creates a logfile
* Load\_YouTube\_Process.xaml: Takes the users inputs from Enter\_Form\_Process.xaml and searches YouTube using the arguments as filters and validates the filters.
* Post\_Data\_To\_Orchestrator.xaml: Scrapes the result-set from YouTube after the filters have been applied and adds them to a DataTables
* Scrape\_Results\_Process.xaml: Posts DataTable of YouTube results to UiPath Orchestrator Queue, then retrieves results from Queue to confirm that they were successfully posted.
* Send\_Status\_Report.xaml: Creates log file of YouTube Search Results, Emails Search Results and Input Log File along with UiPath Orchestrator Summary Statistics

**Bot Type**

* This automation uses an attended bot executed by a user. It is not currently setup to be run on a schedule or executed by a trigger.

**Specific Requirements**

* five data types
  + String: String Data Types are used in the following Sequences:
    - Main.xaml
    - Enter\_Form\_Process.xaml
    - Load\_YouTube\_Process.xaml
    - Post\_Data\_To\_Orchestrator.xaml
    - Scrape\_Results\_Process.xaml
    - Send\_Status\_Report.xaml
  + Int32: Int32 Data Types are used in the following Sequences:
    - Main.xaml
    - Enter\_Form\_Process.xaml
    - Scrape\_Results\_Process.xaml
  + DataTable: DataTable Data Types are used in the following Sequences:
    - Main.xaml
    - Enter\_Form\_Process.xaml
    - Load\_YouTube\_Process.xaml
    - Post\_Data\_To\_Orchestrator.xaml
    - Scrape\_Results\_Process.xaml
    - Send\_Status\_Report.xaml
  + Boolean: Boolean Data Types are used in the following Sequences:
    - Scrape\_Results\_Process.xaml
  + IEnumberable: IEnumerable Data Types are used in the following Sequences:
    - Post\_Data\_To\_Orchestrator.xaml
* two application interactions
  + Application interactions occur in the following Sequences:
    - Enter\_Form\_Process.xaml
      * Writes user input to .csv file
    - Load\_YouTube\_Process.xaml
      * Loads a Browser and navigates to YouTube
    - Post\_Data\_To\_Orchestrator.xaml
      * Uploads Transactions to Orchestrator Queue
    - Scrape\_Results\_Process.xaml
      * Extracts results of YouTube search to DataTable
    - Send\_Status\_Report.xaml
      * Write YouTube search results to .csv file and send an email through Gmail
* asset from Orchestrator
  + The following workflows use an asset from Orchestrator
    - Enter\_Form\_Process.xaml
* three invoked workflows
  + The following workflows are invoked by the automation:
    - Enter\_Form\_Process.xaml
    - Load\_YouTube\_Process.xaml
    - Post\_Data\_To\_Orchestrator.xaml
    - Scrape\_Results\_Process.xaml
    - Send\_Status\_Report.xaml
* queue they used (or why they didn't use a queue)
  + The following workflows use queues:
    - Post\_Data\_To\_Orchestrator.xaml
      * Bulk Adds YouTube search results to Orchestrator Queue then downloads them after they were successfully uploaded.
* data manipulation function or activity
  + The following workflows perform at least one data manipulation activity:
    - Load\_YouTube\_Process.xaml
    - Scrape\_Results\_Process.xaml
    - Send\_Status\_Report.xaml

**Dependencies**

* Using this automation requires the following dependencies:
  + A Gmail account to send emails from (users will be prompted to login to their own gmail
  + An email address that will be used to receive emails stored as an asset in UiPath Orchestrator.
  + account)
  + A Google Chrome Browser to access YouTube
  + A text editor for viewing .csv files
  + A UiPath Orchestrator Queue to upload transactions to.

**Additions or Modifications**

* Possible additions or modifications that could be made to this automation include:
  + Storing search terms and filters as UiPath assets
  + Migrating the automation to an unattended bot that is executed on a schedule
  + Using a ‘headless’ browser or API to scrape YouTube data
  + Storing all results in a UiPath Orchestrator Queue instead of local flat files
  + Moving items from the Queue to a Storage Bucket after a specified period of time