EgyBest Series Bot

Description*:*

This bot takes a series' links from the user, along with the required seasons and quality. Then creates a folder for each series that contains text files with the downloadable links of the episodes for each season (All of these folders are created in the following path: "C:\EgyBestBot").

Process type:

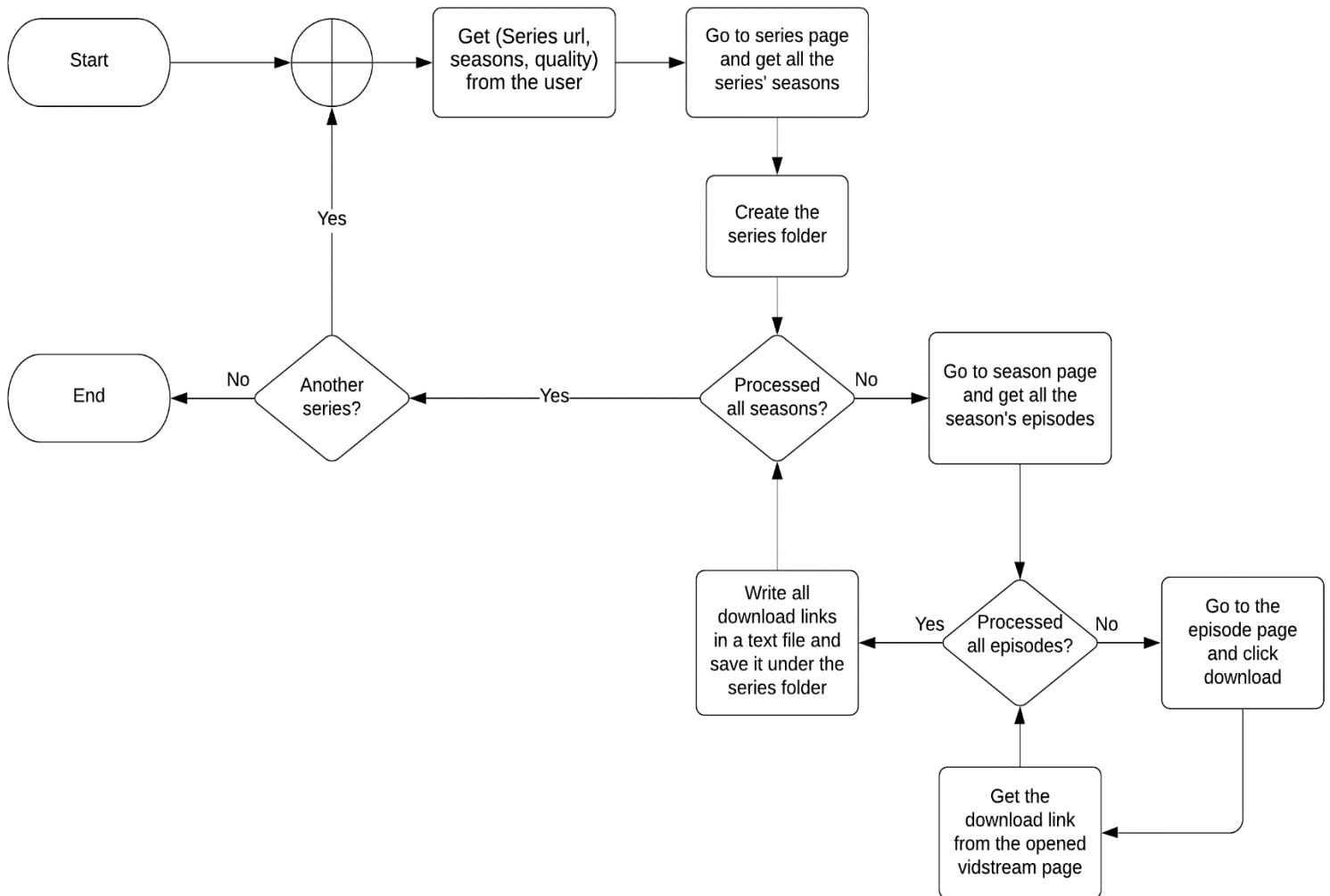Attended (To be triggered by the user)

Prerequisites to run the process:

1 – Microsoft Edge is installed and UiPath extension is installed on it.

2 – The ad blocker MUST be disabled for egybest.

NOTE: there are many domains for egybest, e.g.(https://vega.egybest.pw/, https://fool.egybest.golf/, etc..) so you must disable the ad blocker on the required one.

Handling unknow errors:

Take a screenshot of the whole screen and save it in the "Exceptions Screenshots" folder.

Detailed process map:

Process workflows:

#	Workflow File Name	Description	Pre/Post conditions	Arguments
1	InitAllSettings	Initialize, populate and output a configuration Dictionary to be used throughout the project. Settings and constants are read from the local configuration file, and assets are fetched from Orchestrator. Asset values overwrite settings and constant values if they are defined with the same name.	Pre-condition: N/A Post-condition: The output config dictionary contains the required configuration for the process.	in_ConfigFilePath: String in_ConfigSheets: String[] out_Config: Dictionary<String, Object>
2	KillAllProcesses	Force the termination of the Windows processes representing applications used in the process being automated.	Pre-condition: N/A Post-condition: All the processes name in the passed array -if any- are terminated successfully.	in_Config: Dictionary<String, Object>
3	TakeScreenshot	Capture a screenshot, log its name and location and save it with the PNG extension. If no specific file path is passed as argument, it saves the image in the folder specified by in Folder.	Pre-condition: N/A Post-condition: Screenshot of the whole screen is taken and saved successfully.	in_Config: Dictionary<String, Object> in_Folder: String Io_FilePath: String

4	EgyBest_GetSeriesInfo	Gets the following info from the user: (Series link, required seasons and required quality). Then scraps the required seasons and pass the information as output arguments.	Pre-condition: MSEdge is open. Post-condition: Output arguments are in a correct format.	in_Config: Dictionary<String, Object> out_SeriesName: Sting out_ReqQuality: String out_SeasonsDT: DataTable
5	EgyBest_ProcessSeries	Process each required season in the current series. Then save the downloadable links in the current series folder.	Pre-condition: The passed seasons data table and the name of the series are not empty. Post-condition: All the required seasons in the current series are processed successfully.	in_SeriesName: Sting in_ReqQuality: String in_SeasonsDT: DataTable in_Config: Dictionary<String, Object>
6	EgyBest_ProcessSeason	Process each episode in the current season. Then pass the downloadable links as an output.	Pre-condition: The passed season URL is correct. Post-condition: All the episodes in the current season are processed successfully.	in_SeasonURL: String in_ReqQuality: String in_Config: Dictionary<String, Object> out_SeasonDownloadLinks: String
7	EgyBest_ProcessEpisode	Navigates to the episode page, get the episode download link of the	Pre-condition: The passed episode URL is	in_EpisodeURL: String

		required quality (if the required quality doesn't exist, the highest available quality will be selected)	correct. Post-condition: The episode download link is successfully extracted.	in_ReqQuality: String in_Config: Dictionary<String, Object> out_EpisodeDownloadLink: String