1. ­­Before running the script, download the excel file template titled AutoTestingFields.xlsx and place it in your default downloads folder on Windows.
2. A picture containing graphical user interface

   Description automatically generated
   1. The excel file with three sections, in the grey area is where you put your info into and you can add as many lines as you want. The yellow and blue section is where the script will output its data that it will find, where yellow is preliminary info and blue is the post info after receiving.
3. To run the script, navigate to the EWM\_WD3\_HU\_Receiving\_and\_Putaway\_Script.exe file or the EWM\_QWD\_HU\_Receiving\_and\_Putaway\_Script.exe file, depending on what environment you want to run the script on, that can be stored anywhere you want, and simply double click it to run the program. You do not need to do anything else as the program is running.
4. The script will go through and check the preliminary info, do receiving and putaway process, and gather post info. It will output this information into the same AutoTestingFields.xlsx in the yellow and blue sections so make sure that file isn’t opened when running the script.
5. VERY IMPORTANT – when the script is about to go into the RF menu for the first time for receiving the first line in the excel file, make sure the chrome window is active and that you are not doing anything else on the computer. The script needs to press enter in order to select the certificate for entering the RF ui. The script only needs to do this once, and so after this you can let it run in the background.
6. When running the script, it will open a black console window where it will print out statements describing what it is currently doing. It will create a new chrome window where it will state that its being run by automated test software as seen below. The script runs in the background independently, simulating the clicks and key presses, but it doesn’t take control of your computer, allowing you to continue other work.
   1. Graphical user interface, text, application

      Description automatically generated

Text

Description automatically generated

When running the program, the console will something like this. If you see some strange errors involving not being able to connect to Bluetooth or devices, then that is normal.

1. Important to note is that if the script is able to finish all of the pre info gathering steps but fails at receiving, it will still output the pre info into the yellow section of the AutoTestingFields.xlsx file. The next time you run the script, it should be able to go straight into receiving for that line that wasn’t able to finish and continue on with the script.
2. Also to note is that the script will download other excel files within the EWM ui, but it deletes those files in your directory so you don’t have to worry about a bunch of downloaded excel files piling up on your computer.
3. After it finishes running, it will export an excel file to a Script Results folder in your C:\ within another folder with the date as month-day-year and the excel file itself will have the time at the end of its filename as hour-minute-second (in 24 hour format).
   1. 
   2. 
4. In the results file there will be 3 sheets, Script Results, Data, and Screenshot
   1. Script Results will list out each step that the script performed and if it passed with detail as to which element on screen it was interacting with. If the script times out, it will still export an excel file and the last step listed in the Script Results sheet was the last step it did.
   2. The Data tab isn’t necessary to look at since the results will be outputted it to the AutoTestingFields.xlsx file
   3. The Screen shot tab displays a screenshot of the last thing the script saw in the browser before ending.
5. To close the script, simply close the browser window or the console window and it will exit out of everything, but you should only do this once the console window says “SCRIPT HAS FINISHED RUNNING” which will happen either once the script has finished completing the process or if the script failed at detecting something in which it the script will timeout by itself after about 30 seconds giving an error message in the console window. Then you can close the windows. Check the excel file to see what the last step was and what it saw in the screenshot.
6. It should not ever be possible for it to fail at launching the chrome window at least, unless you don’t have chrome installed.
7. The script is very consistent in running, so you shouldn’t encounter too many errors, unless the web elements of EWM have changed in some way or the script was too fast or too slow to detect a web element which are the most possible errors that could happen. If you encounter the script consistently failing at detecting a certain element, then please let me know by sharing the excel file that is outputted.
8. If you want to run the program again edit the AutoTestingFields.xlsx to add as many new lines as you want of parameters in the grey section and run the exe file again.