# Module Documentation

J:\Quality Data\Data Technician\eCase Migration

Project purpose and summary of project. Installation instructions.

Download Python - <https://www.python.org/>

Install Python. Select the checkbox ‘Add Python to PATH’

Run the file requirements.bat in J:\Quality Data\Data Technician\eCase Migration. This will install any packages needed to the local machine.

Make a shortcut of the file ‘RunECaseMigration.bat’ to open the program when you need to create printouts from eCase.

All files are created and saved in ‘G:\eCase\Downloads’, unless otherwise specified.

\_\_init\_\_.py

-Creates the GUI from tkinter of the window and buttons. Assigns functions created in Buttonfunctions.py to buttons.

Buttonfunctions.py

-Defines the below functions to be used in \_\_init\_\_.py as buttons

doctors\_Allocations()

* Opens a selenium browser with eCase\_Login, and downloads eCase reports, then creates two reports, one with a list of residents and who their doctor is. The second file is a summary of each doctor and how many resident’s they are looking after

bowel\_Files()

* Opens a new GUI window with two buttons for the below two functions

eCase\_Bowel\_Report()

1. Opens a selenium browser with eCase\_Login, and downloads the bowel eCase reports. Creates an excel file with a sheet for each area, for this month’s Resident’s bowel records

prev\_Case\_Bowel\_Report()

1. Opens a selenium browser with eCase\_Login, and downloads the bowel eCase reports. Creates an excel file with a sheet for each area, for last month’s Resident’s bowel records

eCase\_Care\_Plans()

* Opens a selenium browser with eCase\_Login, and downloads eCase reports, creates an excel file with a sheet for each area, showing careplans for each resident.

area\_Checkboxes()

* Opens a new GUI window with checkboxes for each area. When you select whichever area you want, and select ‘Edit’ or ‘Print’, the function podiatry\_List will be called

podiatry\_List(view: str)

1. Opens a selenium browser, with eCase\_Login, and prints, or opens, and excel file with a list of Resident’s and their care levels per area

eCase\_Data\_Download()

* Opens a selenium browser with eCase\_Login, and downloads eCase data reports, then writes this data into eCaseData.xlsx in J:\Quality Data\Clinical Data. eCaseGraphs.xlsx has a collection of pivot tables to analyse this data.

printing\_Files()

* Opens a new window with a text entry for NHI number, and three buttons, one for each of the following functions

front\_Sheet(window,entry)

1. Opens a selenium browser with eCase\_Login, and downloads eCase reports, resident\_Image, and preferred\_Name. Uses this to create a formatted excel file for Admissions manager and Accountants, with Resident info and Resident Contacts.

door\_Label(window,entry)

1. Opens a selenium browser with eCase\_Login, and downloads eCase reports with resident\_Contacts. Downloads the resident\_Image, and preferred\_Name, then uses create\_Door\_Label to create a formatted excel file with the resident’s name and photo to place on their door

label\_List(window,entry)

1. Opens a selenium browser with eCase\_Login, and downloads eCase reports with resident\_Contacts. Gets the preferred\_Name, and then create\_Label\_List is called to generate a formatted excel file to print sticky labels

pi\_risks()

* Opens a selenium browser with eCase\_Login, and navigates to each resident’s file, grabbing their PI Risk level. Creates a file of PI Risk levels for each resident at G:\eCase\Downloads\PIRiskLevels.xlsx

resident\_Birthdays()

* Opens a new window with two buttons, one for the following two functions

resident\_Birthday\_List()

1. Opens a selenium browser with eCase\_Login, and downloads eCase reports with eCase\_Birthdays. Then creates an excel file with all current residents and their birthdates

village\_Birthday\_List()

1. Opens a selenium browser with eCase\_Login, and downloads eCase reports with eCase\_Birthdays. Then creates a formatted excel file with the next month's birthdays in date, Villa order for all Village residents.

temp\_Movements()

* Opens a selenium browser with eCase\_Login, and downloads eCase reports with eCase\_Movements, then appends new temporary movements to G:\eCase\Downloads\eCaseTempMoves.xlsxmand\_Training()

mand\_Training()

* Takes csv file from MYOB, and updates the ‘Training to be book.xlsx’ file with all mandatory training. In 'J:\Quality Data\Training'. For Jane

print\_ClinFiles()

* Prints all files from J: \Forms and Standard letters\Clinical Files for Admissions\For Clinical File Folder

staff\_Birthdays()

* Takes a csv file from MYOB and creates a formatted file for this and the next two months.

walls\_Roche()

* Takes a csv file, copy pasted from the Walls&Roche medication PDF (sourced from Clinical managers), and formats it for pivot table use

file\_Available(file: str)

* Checks whether the file given in the argument is open or not by any user. Does this by attempting to rename it quickly, and renaming back to what it was originally.

popup\_Error(msg: str)

* Creates a pop up with a message (msg) on it. Used for general error messages.

quit(window)

* Destroys the GUI window

eCaseDataImport.py

-Processing of data files downloaded from eCaseDownloader into a format to have a pivot table access. All files are created and saved in ‘G:\eCase\Downloads’, unless otherwise specified.

care\_Level\_List(print\_Area,viewType)

* Takes reports downloaded with care\_Level\_CSV(driver). Generates an excel file for each area with a list of residents and their care levels.

Import()

* Takes reports downloaded with eCase\_Data(driver). Transfers all data into an excel file ‘J:\Quality Data\Clinical Data\eCaseData.xlsx’. Also formats this for use in pivot tables.

bowel\_Import(wing)

* Takes reports downloaded with bowel\_Report(driver,wing), or prev\_Bowel\_Report (driver,wing). Transfers this data into an excel file in ‘J:\Quality Data\Clinical Data\BowelsRecord’. Formatted into a sheet for each area, as well as a sheet that has all areas in it. Pivot Table ready. Needs to b

bowel\_Setup()

* Sets up an excel file for use in bowel\_Import. Establishes a sheet for each area.

bowel\_Report\_Cleanup()

* Run after bowel\_Import in order to save the working excel document with the current date.

care\_Plans\_Setup()

* Sets up an excel file with a sheet per area.

careplan\_Import(wing)

* Takes reports generated from eCase\_Care\_Plans\_Audit(driver). Will use the file generated from care\_Plans\_Setup() and list all careplans for each area and resident. Places in each area’s sheet, as well as in the sheet for All areas

excel\_Sort(file: str,sheet: str,cols: list,all\_cols: list)

* Takes an excel file and sheet as string. This will sort the excel file by the order of the columns in the argument ‘cols’, and write the columns specified in the argument ‘all\_cols’ to the excel file

careplans\_Missing\_Audits()

* Opens the excel file made in care\_Plans\_Setup(), and careplan\_Import(wing). Then saves a separate excel file for each area, with contents of only mandatory care plans and their status. For Lou, saves in ‘J:\Quality Data\Audits\CQR audits’.

doctor\_Numbers()

* Takes the file downloaded with doctor\_Numbers\_Download(driver). Establishes two new excel files, one with a summary of doctors allocation per area, and the other with a list of current residents with who their doctor is.

eCaseDownloader.py

-Functions to login, download reports from the eCase report generator, and close the selenium chrome browser. All files are downloaded to G:\eCase\Downloads

eCase\_Login()

* Establishes an instance of chrome in selenium. Navigates to eCase and logs in with ‘dpugh’ credentials.

eCase\_Data(driver)

* Navigates to the report screen, and downloads all reports with the keyword ‘data’.

eCase\_Bowel(driver)

* Navigates to the report screen, and downloads all reports with the keyword ‘bowel’

eCase\_Care\_Plans\_Audit(driver)

* Navigates to the report screen and filters reports to only reports with keyword ‘cp\_’

care\_Plan\_Audits\_Download(driver,wing)

* Clicks the ‘Generate’ button, and enters the wing into the filter box, and downloads the subsequent report. Should be used in a loop of wings, in conjunction with eCase\_Care\_Plans\_Audit. This is used in a loop within ButtonFunctions.py, to pass in each wing and download the wing's file. This file is then used to create a file with a sheet for each area and their care plan’s status.

bowel\_Report(driver,wing)

* Gets the current month’s bowel report, from start of month, to current date. Handles the selecting of dates within the eCase Month selector GUI. Needs to be run for each wing needed

prev\_Bowel\_Report(driver,wing)

* Gets the previous month’s bowel report, from start to end of month. Handles the selecting of dates within the eCase Month selector GUI. Needs to be run for each wing needed

resident\_Image(driver,NHI)

* Gets the resident’s image and saves it in the eCase\Downloads folder with the NHI as the name

preferred\_Name(driver,NHI)

* Gets the resident’s preferred name, and saves it in a text file in the eCase\Downloads folder, named p\_name.txt

resident\_Contacts(driver, NHI)

* Downloads all reports starting with the name ‘fs’. Will be Resident info, and Resident Contact’s info

doctor\_Numbers\_Download(driver)

* Downloads the report with ‘doctor’ in the name, made by ‘dpugh’. Report has a list of residents and who their doctor is.

eCase\_EPOA(driver)

* Downloads the report with ‘epoa’ in the name, made by ‘dpugh’. Report has the list of resident EPOAs

eCase\_Birthdays(driver)

* Downloads the report with ‘birthdayList’ in the name. Report has the list of resident birth dates.

care\_Level\_CSV(driver)

* Downloads reports with ‘pod\_’ in the name. Downloads the report ‘pod\_MCF’, and ‘pod\_Residents’, both with the level of care for each resident

eCase\_Movements(driver)

* Downloads CSV of ‘temp\_movements’. Handles the selecting of dates within eCase date selector, selects from 1 July 2018, till the end of the current month.

eCase\_Close(driver)

* Closes the current selenium chrome driver instance

printingDocuments.py

-Processing and printing of admission/administrative files, from eCase or from established files on the network

clinical\_Files()

* Print all files in the folder \\SAV-FP01\data$\Shared Data\Care Manager\Forms and Standard letters\Clinical Files for Admissions\For Clinical File Folder. This is for Admission files.

temp\_Movements\_Print()

* Takes the csv report of Temporary movements from eCase, and appends any new movements that are related to Social Leave, Hospital, or Home visits. This is for Elizabeth Pane.

create\_Frontsheet()

* Takes the fs\_Res and fs\_Con reports from eCase, and produces a formatted front sheet for use in admission files. Prints out 2 copies with banking account information, and 1 without for the admission filing. This is for the Admission officer and accountants

create\_Door\_Label()

* Takes the fs\_Res and fs\_Con reports from eCase, and prints a formatted Door Label to place on the front of the resident’s room.

create\_Label\_List()

* Takes the fs\_Res and fs\_Con reports from eCase, and produces a formatted excel document for printing on a sheet of sticky labels. This won’t automatically print out the list, but it will open the formatted document to be printed from the bypass tray with the sticky labels. For Admissions officer and receptionist.

birthdays\_Printout()

* Creates an excel document on all resident birthdays.

village\_Birthdays()

* Creates a formatted excel document of all Village residents with a birthday this month.

staffDocs.py

-Processing of csv files downloaded from MYOB.

training\_Lists()

* Creates the mandatory training to be booked list by using the training.csv file in J:\Quality Data\Data Technician\StaffDbases. Places it in the directory specified in the function. Creates an individual sheet for each mandatory training, and puts the relevant employee information in each sheet.

birthday\_List()

* Creates a birthday list, sorted by date, for this and the next two months. Will place in StaffDbase in Datatec folder

styles.py

-Custom code snippets to make it easier to format files

printSettings(file\_Directory: str, widths = [], sheet: str, one\_Page=True, landscape=True)

* With specifying the file and sheet, this can assign the sheets print settings to be one page, landscape/portrait, as well as assign column sizes with the width argument, where each item in the list should be a number, and defines x amount of columns. For excel, the width for each column in pixels is x\*7

full\_Border(ws: str, cell\_range: str, border = 'thin')

* Openpyxl can create a border on a single cell, but I’ve created this to create a border encompassing a range of cells. Accepts worksheet, and cell range in the excel format, i.e ‘B2:G7’. Border can be one of {‘double’, ‘mediumDashDotDot’, ‘slantDashDot’, ‘dashDotDot’, ‘dotted’, ‘hair’, ‘mediumDashed’, ‘dashed’, ‘dashDot’, ‘thin’, ‘mediumDashDot’, ‘medium’, ‘thick’}

wrmeds.py

-Data processing of the CSV files derived from the PDF of Walls & Roche files from Clinical managers

wrmeds\_Counts()

* Copy all the text from PDF and paste into a CSV file. Save and exit, and run this function on this new CSV