Criteria A: Planning

A/ Defining the problem

My client, Mr. Greenwood, is an IGCSE ITGS teacher at my school, often having to provide past paper questions as class- or home-work. Currently, he has been manually going through each set of past papers and screenshotting questions, then manually sorting them based on year *(See Appendix, Transcript 1)*. This has been time consuming and repetitive and is a common chore that most teachers have to do.

Consequently, I approached him with the intention of solving his problem with a technology-based solution. Previously, the client has stated his frustration in how “it’s complicated by page breaks.” In our interview, he further stated “[a program] that you could just put all the PDFs into a folder and run it on and get a bunch of sorted out questions would be perfect.” Individual images of separate questions will then be produced, indicated by an appended ‘_q(x)’ at the end of each file name, with ‘(x)’ being the question number. For example, an image of Q4 from Summer 2020, paper 1, time zone 1, will be ‘0417_s20_qp_11-q4.jpg’.

B/ Rationale for the proposed solution

Initially, I suggested building a program with a GUI for my client in order to make it more presentable and easier to use. However, my client said “I don’t see any benefit to me in having a GUI”, therefore we decided on a command-line program as it would not require him to learn new skills, with him being familiar with computers and the command-line as an IT teacher. Advantages of not using a GUI includes simplicity along with functionality, better potential for future development and no real limitation on the software and hardware front. Possible limitations include the exclusivity of the script to those who know how to run shell scripts.

My client then provided me with a folder full of test files (past paper PDFs), which can also be obtained from online IGCSE past paper sources. There are no security implications regarding this project.

As for the development of the product, though the optimal solution is to write it using shell scripting, I feel as though I am more adequately familiar with Python 3.9 in Sublime Text 3 and

would therefore be able to create a better and more efficient product with it. There is a wider range of existing libraries which makes Python a much more flexible choice, such as pdf2image, pytesseract, cv2, and PIL. Alternatively, I could write a shell script which utilises different resource operations, which then runs an external application written in Python.

C/ Success criteria

1. Requires minimum user input
2. Parse through a PDF file and produce separate cropped images of questions and its respective answer.
3. Must output files with correctly formatted file names which includes:
    a. Course code
    b. Year and session
    c. Paper
    d. Question number
4. Each image file must contain the full question with all of its accompanying subquestions.
5. Each image file should have an accompanying mark scheme image.
6. Must be able to run on Cambridge IGCSE ICT exam paper 1 post 2017 syllabus and format change.
7. Final output must be two folders, one containing questions and the other containing mark schemes. Each file must be named accordingly and each folder must contain the full paper.

Word Count: 415