PROJECT DRAFT/ LAYOUT / BRAINSTORM

***PYTHON – 5 days project finished by Saturday 11:59:59pm***

**PROMPT: Create a portable application that can validate an image is a jpeg and find out where that image was taken and return the zip code it was taken in.**

**Determining the tools/approach based on specifications**

+Main functionality which are Validating an image is a jpeg and find out where that image is taken/return zip code.

* Validating an image is JPEG: ~~based on .extension~~ or header magic number
  + **FF D8 at the first 16 bits, FF D9 at end 16 bits**. Ensure EXIF JPG JPEG JFIF
  + https://www.onlinehexeditor.com/#
* Find out where that image is: **EXIF ITSELF then GeoPy Then Google VISION**/ ~~Flickr API/ Unsplash/ Tineye/ Geocoding~~
  + JPEG may not be EXIF
  + EXIF may not store gpsinfo
  + May use AWS and Google cloud services
  + If EXIF then -> pipe lat\_long to Geopy or Google place Api
  + If not EXIF then -> send image to Google Vision ML

+ Your tool must work on Linux, OSX, Windows + GUI also/ easy to install and use (GUI is extra credit)

* Use **PyInstaller** to turn script into .exe -> Cross platform and simple to install but may raise warning against EXE files. ( if that happen, send full code with instruction to .exe maybe set this up inside a docker container with port forwarded to used via http requests?)

NOTE: Other options include Py2exe, Nuitka, etc but these are not as reliable as Pyinstaller as they are newer

* Python GUI Choices : **TKinter** / ~~PyQT/ Kivy/ WxPython.~~

NOTE: BACKWARD COMPATIBILITY!!! Comes with Python !! no extra installation, works well with Pyinstaller

+You must convert any image metadata yourself and have code to show it. HEX to binary/string .

* Create your own function to translate hex to binary/string instead of using lib.

+You may use libraries to read metadata, but make sure they are portable.

* Using the standard library or import library via URL… maybe?

**POP FACTOR Possible**

0/ portable application single executable – prod – CHECK 66.67%

1/ GUI for desktop use – prod - CHECK

2/ Multiple image functionality – prod - CHECK

3/ Graphical design , add your logo in there – prod – CHECK / SCRAPPED, photo not portable and unnecessary

4/ Clean in-code Normal documentation -prod- CHECK

4.5/ Sphinx documentation – prod- No Time

5/ CSV export – prod – CHECK

5.5/ Try to use one of EVT Partner solutions <https://evtcorp.com/technology-partners/> - prod – research would take too long

5.75/ Use AWS s3 and Google Cloud API instead! – prod – CHECK? – GCP took file path no need for s3

5.8/ Use Selenium Web Scraping to deal with images that we found long/lat but both GeoPy and Google Service fail to determine zip code – prod – No Time

6/ REST API !! -> web server version instead of app . Return json – prod – No Time

7/ Ansible/Docker to setup Docker container /ip to use rest api. – prod -No Time

7.5/ Put Docker in Kubernetes pods for resource management? – prod – No time

8/ DEPLOY IT TO THE CLOUD! Use MS Azure instead of AWS/GCP (already used it before) – test – No Time

Deliverables:

* + Code & executable … When you sell a product you sell both?
  + If code deliverance then setup scripts to install needed python packages
  + README
  + Tests