CIS 314 Final Project

Summary:

The final project is an open-ended project which constitutes 25% of a learner's final grade. Open-ended means that you are permitted to select the type of system you design and build, making it as complex or simple as you feel is appropriate to achieve your goals.

Each project is evaluated individually. This means that just because a project is more complex than another, that does not mean that the more-complex project will score higher. However, each project should demonstrate the learner's knowledge and mastery of the topics discussed during the course.

The project does **not** need to utilize all of the topics discussed this semester but should incorporate as many as are necessary. The final project does **not** need to build off a previous midterm project, but if elected to do so, should expand on the functionality, correct deficiencies, and improve code from the midterm.

Unlike the midterm, the final can make use of as many external packages and libraries as necessary. Many of the topics discussed in the second half of the semester rely heavily on external packages. As such, and to keep with the principles of being "Pythonic," final projects can make full use of the Python community.

Project code is expected to successfully run without meaningful modification. If the project requires certain conditions, such as a folder of files, such conditions should be documented in the README. *De minimus* alterations, such as specifying the directory of files or the exact name of an image file do not fall within the bounds of modification.

Requirements and Parameters:

- Project Organization:
    - Group or Solo:
        - Groups: A group leader must be selected and group members identified before project work begins
    - Project Title
    - Project Summary
    - Project Goals
- Technical Requirements:
    - Python as primary language
    - Upload source code to GitHub
    - Package source code into single file (.tar.gz) uploaded to Brightspace
        - All dependencies listed in pyproject.toml
        - README.md
        - LICENSE file
- Documentation Requirements:

    - Proper summary of project in README.md file
    - Appropriate license selection
    - A list of packages required
    - Commenting throughout code
- Post-Project Report
    - Did the project achieve its goals?
    - Did the goals shift during development?

Grading Assessment:

Projects will be evaluated on their ability to fulfill their stated goals and adhere to the requirements set forth above. The following grading rubric will be used:

Topic	Points
Project Organization Solo/Group Notification Project Title Project Summary Project Goals/Objectives	20
Technical Requirements Upload to GitHub Source Code Packaged into .tar.gz	30
Project Contents/Code Pythonic Code Proper Commenting Successful Execution	100
Documentation Clear summary of project in README Valid license selection Dependencies in Pyproject.toml	30
Post-Project Report	20
Total	200