Data Manipulation Software Conversion

Project summary

Our project is to convert an old LabView software platform used by the Cousins Photosynthesis Lab at Washington State University for data acquisition, manipulation, and calculations to a Python application.

Additional Information About the Project

Our team has been tasked with converting an old LabView software platform to a Python application. The old platform performs data acquisition, manipulation, and presentation on inputs received from instruments in Cousins Photosynthesis Lab in The School of Biological Sciences at Washington State University. The new application will replicate the current solutions ability to collect data and make calculations and manipulations but will also enhance the efficiency and usability. The application will take the form of a PyQt5 Desktop Application supported by other Python libraries. The end goal of this project is to provide an updated Python application that is more user-friendly and efficient than the current software.

Installation

Prerequisites
The user needs to have the latest version of python installed on the system. The user also needs to have all the drivers up to date.

Add-ons

PyQt5: Used to build a GUI interface in Python.
PyQtGraph: Used for creating animated and interactive plots in Python.
Pandas: Used for data manipulation and analysis.
NumPy: Used for high-level mathematical functions and handling large arrays and matrices.

Installation Steps

The user will be provided with two executables. The first executable will install/update all the required and necessary prerequisites on the system [package-installation.py]. The second executable will be used to run the application [main.py].

Functionality

After running the package-installation.py executable to install/update the needed prerequisites, run the main.py executable that runs the application. Once running, the application will show a window with a blank plot, a Start button, a Pause button, a speed slider, and checkboxes.

Clicking the Start button will begin a plot animation of 8 streams of Time vs. Voltage data. The data streams are all different colors. The user can select which plots (graphs 1 - 8) are visible with the checkboxes below. The speed of the graph can be manipulated with the speed slider below as well. Right clicking on the plot will display a menu of options including the option to change the scale of the axes.

Clicking the Pause button will stop the plot animation where it is. Once the plot is paused, two vertical bars will appear on the plot, these are the Mean Bars, and they can be moved to any position on the plot. The Mean bars will highlight the range of the plot between them. You can move either bar or you can move the highlighted area all at once.

The folder with all the data was truncated when pushing to GitHub, so not all the data will be plotted

Known Problems

- Cleanup and proper file organization are required for the code.
- Implementing an object-oriented programming strategy is necessary.
- Application of the SOLID Principles requires revision.

Contributing

1. Fork it!
2. Create your feature branch: `git checkout -b my-new-feature`
3. Commit your changes: `git commit -am 'Add some feature'`
4. Push to the branch: `git push origin my-new-feature`
5. Submit a pull request :D

Additional Documentation

- Sprint 1 report: : https://github.com/WSUCptSCapstone-Fall2022Spring2023/biology-labviewtopython/blob/main/Documents/Sprint%201/Sprint_1_Report.pdf
- Sprint 2 report: https://github.com/WSUCptSCapstone-Fall2022Spring2023/biology-labviewtopython/blob/main/Documents/Sprint%202/Sprint_2_Report.pdf