Data Manipulation Software Conversion

Project summary

Our project is to convert an old LabView software platform used by the Cousins Photosynthesis Lab 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 Tkinter 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

Matplotlib: Used for creating animated and interactive plots in Python.
Tkinter: Used to build a GUI interface in Python.
Pandas: Used for data manipulation.

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. The second executable will be used to run the application.

Functionality

First, run the executable that runs the application. Once running, the application will begin a plot animation of 8 streams of Time vs. Voltage data. The data streams are all different colors and label accordingly in a key.

Currently, there is nothing that can be performed on the data or the plot itself, except for a zoom functionality that immediately returns to a normal zoom state.

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

Known Problems

- When plotting a graph using real-time generated data, the application appears to lag while it is running.
- When we attempt to close the graph, the program crashes.
- 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 report: https://github.com/WSUCptSCapstone-Fall2022Spring2023/biology-labviewtopython/blob/main/Documents/Sprint%201/Sprint_1_Report.pdf

License

If you haven't already, add a file called `LICENSE.txt` with the text of the appropriate license. We recommend using the MIT license: <https://choosealicense.com/licenses/mit/>