SIT225 Data Capture Technologies

HD Task: Data dashboard alternative to Plotly

Overview

Dashboard is a way of data visualisation and user interaction. Plotly Dash is one of the great ways of designing a dashboard. There are alternatives including Bokeh, Streamlit and more. You can pick some of them to design a descent dashboard based on your needs.

Hardware Required

i. Arduino Nano 33 IoT device,
ii. USB cable,
iii. LSM6DS3 module on the Arduino Nano 33 IoT for Gyroscope data.

Software Required

i. Python 3.
ii. Alternative library for dashboard.

Pre-requisites: You must do the following before this task

Week 6 activities in the unit site.

Task Objective

In 6.1P, you dealt with Plotly Dash for visualisation and user interfaces for interactions. In this task, you will need to explore 2 alternative visualisation libraries suitable to create flexible dashboards.

Steps:

1. Connect the sensor for collecting data samples, if previous week's data is not available. Make sure you perform different activities during the data collection so that interesting patterns are observed during data analysis.
2. Data should be available in CSV format.
3. Identify an alternative Python library which can create a dashboard on Gyroscope CSV data with below requirements easily.
    a. Graph types should be selected based on a drop-down menu including, but not limited to, basic charts (scatter plots and line charts) and statistical charts (distribution plots).
    b. Interactively select data variables x, y, and z with options to select either of them or all of them using a drop-down menu.

c. Interactively select number of data samples to display in the graph. A text box can be suitable for the user to input the number of samples, the graph is then updated accordingly. In case the number of samples is less than the total number of samples, there should be a next/previous button for navigation through the data, clicking on the navigation button previous (or next) N samples are displayed.
d. Create another visual, such as table, to summarise the data currently being displayed in the graph. This summary needs to be updated if the graph changes due to user action on the graph UI.
e. Dashboard elements need to be updated based on continuous sensor data. As a suggestion, consider the existence of another CSV data file after 10 seconds of displaying the previous CSV file and meanwhile new data fills in a buffer to create another CSV file.
f. Propose your idea explaining how the availability of new data can be integrated into the dashboard library to consume fresh data and refresh dashboard graphics.

4. Repeat step 3 for another alternative Python library suitable for dashboard creation.
5. Contrast Plotly with two alternatives from different aspects including ease of use, flexibility and hackability. This exploration should consider continuous data integration and dashboard update support availability in the alternative libraries.

Submission details

Q1. Perform week 6 activities mentioned in the unit site and produce outputs.

Q2. Explain steps mentioned above which require explanation including sub-items in step 3.

Q3. Paste alternative dashboard Python codes and explain the program steps for each alternative library, how plots were generated, and interaction was handled. Provide insight on each plot w.r.t. the data segments.

Q4. Create a video in Panopto/CloudDeakin showing your program execution, graph output, user interaction for two alternative libraries and share the video link here.

Q5. Create a subdirectory ‘week-6.2D’ under directory ‘SIT225_2024T2’ in your drive where you copy the Python script file, Arduino sketch file if any, data file and the generated graphs. Commit and push to changes to GitHub. Include the link to your repository here with a GitHub page screenshot of weekly folder content. A tutor may try to access your GitHub link, if necessary. Give access to your tutor by adding tutor’s email address as a collaborator of your private repository.

Instructions

Consolidate outputs following the submission details above into a single PDF file.

Submit your work

When you are ready, login to OnTrack and submit your pdf which consolidates all the items mentioned in the submission detail section above. Remember to save and backup your work.

Complete your work

After your submission, your OnTrack reviewer (tutor) will review your submission and give you feedback in about 5 business days. Your reviewer may further ask you some questions on the weekly topics and/or about your submissions. You are required to address your OnTrack reviewer's questions as a form of task discussions. Please frequently login to OnTrack for the task ***Discuss/Demonstrate*** or ***Resubmit*** equivalent to fix your work (if needed) based on the feedback to get your task signed as ***Complete***.