

SIT225 Data Capture Technologies

Distinction Task: Store data to non-Firebase cloud database

Overview

Firebase is not the only NoSQL database when you need to decide a cloud database. There are options to consider such as whether the database can be operated locally or in the cloud, where one operable in both local computer and cloud, certainly has advantages. Popular alternatives to Firebase include MongoDB, Parse, Heroku and AWS Amplify.

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. MQTT protocol (HiveMQ).

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

Week 5 activities in the unit site.

Task Objective

In 5.1P, you have read Gyroscope data from Arduino through serial communication channel to your computer using Python script, which uploads data to Firebase real-time database. In this task, you will need to repeat the same with few changes including change in Arduino-to-computer data transfer protocol and the cloud database.

Steps:

1. Create a MQTT account in any popular MQTT server such as HiveMQ and note down URL and port to be used in Arduino sketch.
2. Configure Arduino sketch to use MQTT protocol so that data readings from sensor can be published to a MQTT queue.
3. Python script should be able to subscribe to the queue where Arduino is publishing data to.
4. Upon receiving data in Python script, try to upload to an alternative NoSQL database (let's call it alternative DB-1) other than Firebase. Do necessary configuration to setup the

database. Cloud version of the database is preferred but not mandatory, you can install it on your computer. Justify your decision.

5. Keep transferring data between Arduino and Python script for at least 30 minutes to have enough data.
6. Now query alternative DB-1 to receive all data you have sent so far, process the JSON data array to format each sample as CSV and store in a file.
7. Try to clean data variables in the CSV file which includes removing any non-number or empty fields, if exists. This step can be a manual process, or you can use Python tools such as Numpy or Pandas, if you want to. Justify your action.
8. Read the CSV file, plot x, y, z variables separately as 3 line-graphs and a graph with all 3 variables in a single graph.
9. Observe the data changing pattern of the single variable plot and combined plot and try to come up with your comments. Following items are subject to observation, but not limited to, -
 - a. Any repeating pattern,
 - b. Up and down trends in data variables, or
 - c. Relative changes of Gyroscope variables x, y or z.
 - d. You should refer to Gyroscope variable semantics in the Arduino document (<https://docs.arduino.cc/tutorials/nano-33-iot/imu-gyroscope>).
10. Repeat steps 4-9 using another NoSQL database (let's call it alternative DB-2). Data analysis may result in slightly varying patterns since you are collecting data for 2 databases separately.
11. Compare Firebase and the 2 alternatives in configuration, data storing APIs, ease of use and any other aspects you want to share your knowledge about.
12. Compare data transfer protocols such as serial communication and MQTT in terms of data sending and receiving/parsing efforts. Highlight interesting aspects you want to share your knowledge about.

Submission details

Q1. Perform week 5 activities mentioned in the unit site and produce outputs.

Q2. Fill in a well formatted report with items mentioned in the steps section above including discussion on databases and data transfer protocols. Better use visuals to make your discussion engaging such as figures and tables.

Q3. Paste Python and Arduino sketch and explain program steps.

Q4. Create a video in Panopto/CloudDeakin showing your program execution, data collection, data upload/retrieve to/from alternative databases, share the video link here.

Q5. Create a subdirectory 'week-5.2D' under directory 'SIT225_2024T2' in your drive where you copy the Python script file, Arduino sketch file, data file and the generated graphs if any. 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***.