Raspberry Pi real-time sensors using self-developed backend and an Android application

Jaakko Ikäheimo, TVT19KMO
Information and Communications Technology, Software Development

Introduction

This poster works as a documentation of a personal mini project made for the IoT section of the mobile programming study period. It describes some central parts of the project and its product.

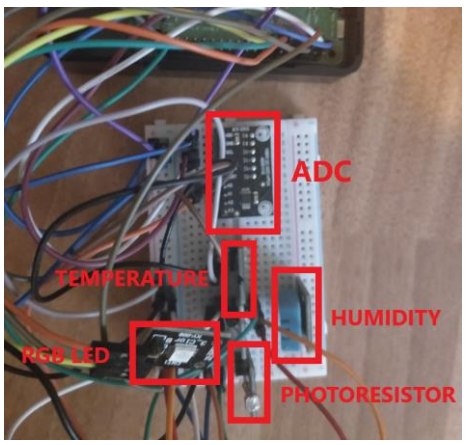

FIGURE 1. Different gadgets attached to Raspberry Pi.

Objectives

The main objective of the project was to develop a system to observe temperature, humidity, and luminosity sensor modules in real-time with an Android application. The system would use a self-developed backend and its services to store and transfer the measured sensor data from the Raspi. In addition to reading sensor data, the user could be able to control an RGB led by using the application. From these objectives, the ultimate goal was to get at least the “real-time” sensor reading to work at some level.

Mini project

ECTS credits: 3

Date of publication: 2020, Autumn

Instructor: Teemu Korpela

Methods

Different sensor modules from past projects were used to measure temperature, humidity, and luminosity of the environment. Python program was developed to use threading for reading sensor data, to handle MQTT messages, and to communicate with the chat service of the server.

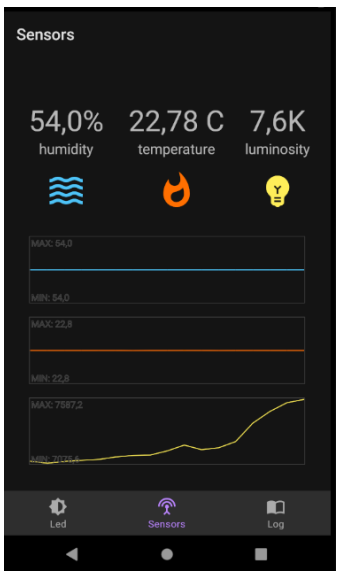

FIGURE 2. Sensors view of the Android application.

The Node.js app on the server was developed by using Express.js and Socket.io libraries to handle the API requests and logging with chat service. Besides, the Node app would listen for MQTT messages and store the data associated with them in the database. The android application was developed to connect with the server's logging service to receive messages from Raspberry Pi, and API was used to fetch recent data from the database. Real-time data observation was implemented by using the MQTT protocol.

.

Results

By using the combination of the MQTT and API + database, it was possible to update the developed Android application with both past and the current sensor data. The project was finished without any major complications, and every primary objective was met. Although some of the features planned along the project were too time-consuming for a mini project, the project was a great learning success.

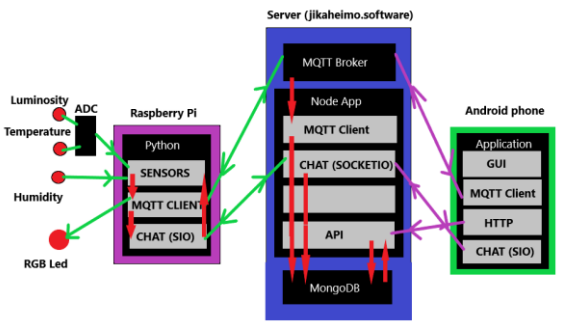

FIGURE 3. System architecture.

Conclusions

The project offered a lot of insight into how to develop a "fully working" IoT system. Also, it provided an opportunity to set up a working server environment.