

NGUYỄN MINH TIỀN

EMBEDDED FIRMWARE ENGINEER

 | 22/11/1998

 | 5751062057@st.utc2.edu.vn

 | 38B Gò Cát, phường Phú Hữu, Quận 9, TP. Hồ Chí Minh

 | 0797557533

 | <https://github.com/spiderrock98>

EDUCATION



09/2016 - 02/2021

UNIVERSITY OF
TRANSPORT AND
COMMUNICATIONS CAMPUS
IN HO CHI MINH

Electronics engineering and
industrial informatics

GPA: 3.15/4

PRIZES AND AWARDS



08/2019

STUDENTS IN SCIENTIFIC
RESEARCH

2nd Place

UNIVERSITY OF TRANSPORT AND
COMMUNICATIONS CAMPUS IN
HO CHI MINH

TECHNICAL SKILLS



UART / I2C / SPI / MQTT /
WebSocket /

■ ■ ■ ■ ■ □ □ □ □ □

C/C++ Basic

■ ■ ■ ■ ■ □ □ □ □ □

8051, PIC, ESP8266, ESP32

■ ■ ■ ■ ■ □ □ □ □ □

PROJECTS



05/2019

07/2019

DIGITS RECOGNIZED ON SCADA SCREEN

- Follows the order of TTI Viet Nam VSIP2
- KNNs algorithm to train 10-digits model
- Implement computer vision with Python on Raspberry Pi 3 (Debian)
- Threading in Python to handle two cameras in parallel for real-time responding

Source: <https://github.com/spiderrock98/KNN-DigitRecognize>

01/2020

04/2020

LINE FOLLOWER ROBOT

- Follows the order of my university - in series of College entrance activities
- Implement on Arduino Uno
- Hardware: TCRT5000 - Module photodiode line follower, L298N - DC motor driver
- Script to calculate average ADC value then applied smith trigger window to determine line

Source: <https://github.com/spiderrock98/XeDoLineUTC2>

02/2020

05/2020

AUTOMATE BELL RINGING AT CAO MOC CHURCH

- PIC16F887 interact with sensors drive motor
- Dimmer AC to control pull speeds
- Rotate Encoder read feedback rotates to change direction, protect over-rotates, prevent starting system if bell still shaking
- Arduino Pro Micro to monitor on LCD 1602 and schedule bell ringing using DS3231 and Keypad

Source: <https://spiderrock98.github.io/RingTheBell/>

Algorithm



HTML / CSS/ JavaScript



NodeJS Framework / Git / Github



English



Computer Vision (Python)



Microsoft Office



03/2020

12/2020

PLATFORM IOTS IN AGRICULTURE

- NodeJS in the back-end with 3 child server: webserver, camera WebSocket server, data-command WebSocket server
- Pair new gateways (ESP32) with user account by send network info from dashboard application to gateway via UART
- Pair new garden devices (Arduino Nano ATmega328P) with gateway by scanning QR code through camera stream of ESP32
- Get camera stream from ESP32-CAM back to camera WebSocket server for display in dashboard as surveillance camera
- Garden devices and server exchange information (sensor values, threshold config, control commands) via gateway back to data-command WebSocket server
- Firebase Realtime Database to store user and device information
- Firebase Authentication to authenticate users
- ElectronJS front-end framework to build desktop application dashboard, monitor sensor, upload configs (ssid, password, userID, usb port, upload baudrate) to setting blank new gateway. Execute python script in background process to detect USB port, get network info
- Visualize sensor data using Google Chart API
- Locate place using Google Map API
- Deploy project on Raspberry Pi 3B, router port forwarding to access server from different networks

Source: <https://github.com/spiderrock98/PairDevices>

WORK EXPERIENCE



06/2020

09/2020

ELEX LAB - ELEXLAB.VN

Internship - Back-end Developer

- Outsource 2 IoTs Smart Home projects for company
- > Source: <https://github.com/spiderrock98/ClientElex-DATN>
- > Source: <https://github.com/spiderrock98/ToHaCoffee>