

IoT Nummerplade Scanner

Projektbeskrivelse

Dette projekt er et IoT-system designet til at registrere biler, scanne deres nummerplader og præsentere data på en webside. Systemet består af en ESP32 med en bevægelsessensor, en Raspberry Pi med kamera til nummerpladescanning, og en anden Raspberry Pi som server til datahåndtering og præsentation.

ESP32'en detekterer biler via en bevægelsessensor, aktiverer en Raspberry Pi til at scanne nummerpladen, og sender data til en MQTT-broker. Dataene gemmes lokalt på ESP32'en, hvis der ikke er forbindelse, og sendes senere. Systemet er energieffektivt via deep sleep og kan let konfigureres med en brugervenlig WiFi-opsætningsportal.

Komponenter

Hardware:

- ESP32 (med MH-Sensor-series flying-fish):**
 - Bevægelsessensor til at registrere bilers bevægelse.
 - Debouncing for at undgå falske positive.
 - RTC (Real Time Clock) for at bevare tidsdata under deep sleep.
 - LittleFS til lokal datalagring.
 - Deep sleep for at spare strøm ved inaktivitet.
 - Raspberry Pi med Raspberry Pi Camera V2:**
 - Kamera til nummerpladescanning.
 - Returnerer nummerpladeoplysninger til ESP32.
 - Raspberry Pi som server:**
 - MQTT-subscriber til at modtage data fra ESP32.
 - Flask-webserver til præsentation af data.
 - SQLite-database til at gemme nummerpladedata.
-

Funktionalitet

ESP32

- Bevægelsesdetektion:**
 - Når en bil detekteres, sender ESP32 en HTTP-forespørgsel til nummerpladescanneren.

- Nummerpladen sendes til en MQTT-broker eller gemmes i LittleFS, hvis der ikke er forbindelse.
- 2. **Deep Sleep:**
 - ESP32 går i deep sleep efter 30 sekunders inaktivitet.
 - Bevægelsessensoren vækker ESP32 ved ny aktivitet.
- 3. **WiFi-opsætning:**
 - Hvis ESP32 ikke kan forbinde til WiFi, opretter den et Access Point med en webside til opsætning.
- 4. **RTC:**
 - RTC bruges til at holde styr på tid under deep sleep.
- 5. **Debouncing:**
 - Implementeret for at undgå falske detektioner fra sensoren.

Raspberry Pi med kamera

1. **Nummerpladescanning:**
 - Modtager HTTP-forespørgsler fra ESP32.
 - Bruger kamera til at scanne nummerplader.
 - Returnerer valid nummerplade eller en fejlbesked.

Raspberry Pi-server

1. **MQTT-subscriber:**
 - Modtager data fra ESP32 via MQTT og gemmer det i en SQLite-database.
 2. **Flask-webserver:**
 - Viser nummerpladedata i realtid på en responsiv hjemmeside.
-

Systemarkitektur

ESP32:

- Bevægelsessensor → HTTP-forespørgsel → Raspberry Pi med kamera.
- Nummerpladedata → MQTT → Raspberry Pi-server eller gemmes i LittleFS.

Raspberry Pi med kamera:

- HTTP-request fra ESP32 → Nummerpladescanning → HTTP-respons til ESP32.

Raspberry Pi-server:

- MQTT-data fra ESP32 → Database → Flask-webside.
-

Quickguide

1. Hardwareopsætning

1. Tilslut bevægelsessensoren (MH-Sensor-series flying-fish) til ESP32 GPIO33 (data) og 3V3 eller 5V.
 2. Tilslut Raspberry Pi Camera V2 korrekt til CSI-porten på Raspberry Pi.
 3. Sørg for, at begge Raspberry Pi'er er korrekt opsat og tilsluttet netværket.
-

2. Softwareopsætning

ESP32:

1. Upload ESP32-koden med PlatformIO eller Arduino IDE.
2. Juster `mqttBroker`, `mqttPort`, og `mqttTopic` i koden, hvis nødvendigt.

Raspberry Pi med kamera:

1. Installer Flask og nødvendige Python-biblioteker (`opencv`, `requests`).
2. Upload Python-scriptet til nummerpladescanning.

Raspberry Pi-server:

1. Installer Flask, SQLite og `paho-mqtt`.
 2. Opsæt Python-scriptet til at subscribe på MQTT og gemme data i databasen.
-

Diagram

