

Bachelorarbeit

Oleksii Baida
Matrikelnummer 7210384

Sicherheits- & Steuerungssystem für das Haus

Bericht

13. Januar 2025

Inhaltsverzeichnis

| | | |
|----------|---|----------|
| 1 | Einleitung | 2 |
| 2 | Grundlagen & Theorie | 3 |
| 2.1 | Hardware | 3 |
| 2.1.1 | Arduino | 3 |
| 2.1.2 | ESP8266 | 3 |
| 2.1.3 | Raspberry Pi | 3 |
| 2.1.4 | M5Stick | 3 |
| 2.1.5 | ESP32 | 3 |
| 2.2 | Kommunikationsprotokolle | 3 |
| 2.2.1 | HTTP | 3 |
| 2.2.2 | MQTT | 3 |
| 2.3 | Software | 3 |
| 2.3.1 | Asyncio | 3 |
| 2.3.2 | FastAPI | 3 |
| 2.3.3 | SQLAlchemy | 3 |
| 2.3.4 | Uvicorn | 3 |
| 2.3.5 | HTML & TailwindCSS | 3 |
| 2.3.6 | Javascript | 3 |
| 2.3.7 | WebSocket | 3 |
| 2.3.8 | Linux-Pakete für Raspberry Pi | 3 |
| 3 | Konzeption des Systems | 4 |
| 4 | Implementierung und praktische Umsetzung | 5 |
| 4.1 | Einrichtung der Hardwarekomponenten | 5 |
| 4.1.1 | Arduino | 5 |
| 4.1.2 | ESP8266 | 5 |
| 4.1.3 | Raspberry Pi | 5 |
| 4.1.4 | M5Stick | 5 |
| 4.1.5 | ESP32 | 5 |
| 4.2 | Softwareentwicklung | 5 |
| 4.2.1 | Webserver | 5 |
| 4.3 | Datenbank | 5 |
| 4.4 | Frontend | 5 |
| 4.5 | Integration der Komponenten | 5 |
| 5 | Ergebnisse und Diskussion | 6 |
| 6 | Quellen | 7 |
| | Abbildungsverzeichnis | 8 |
| | Programmcode | 8 |

1 Einleitung

In einer Welt, die zunehmend von vernetzten Geräten und dem Internet der Dinge geprägt ist, wird die Entwicklung effizienter und benutzerfreundlicher Systeme zur Steuerung und Überwachung von Gebäuden immer relevanter. Im Jahr 2024 wurde in Deutschland die Anzahl von über 19 Millionen Haushalten, die ein oder mehrere smarte Geräte besaßen, verzeichnet. Es wird prognostiziert, dass sich diese Zahl innerhalb der nächsten drei Jahre verdoppeln wird [4].

Moderne Steuerungs- und Sicherheitssysteme tragen zur Effizienzsteigerung und Ressourcenschonung bei. Laut Günther Ohland, Vorstandsmitglied des Branchenverbands SSmarthome Initiative Deutschland“, ermöglichen diese Systeme eine Reduktion des Heizenergieverbrauchs um 20 bis 30 Prozent [5]. Die Kosten für die smarte Technik rechnen sich in der Regel nach zwei Jahren. Die Systeme übernehmen ein Teil der täglichen Aufgaben, wie das Ein- und Ausschalten des Lichts, die Regelung der Raumtemperatur oder das Aufräumen des Hauses etc. Der Aufgabenbereich der Systeme ist dabei nur nach den Bedürfnissen der Benutzerinnen und Benutzer abgegrenzt.

Im Rahmen meiner Bachelorarbeit wird ein System zur Steuerung und Überwachung des Hauses entwickelt. Das Ziel dieser Arbeit ist die Erstellung einer Schnittstelle, die die Interaktion des Benutzers mit den Geräten in seinem Haushalt ermöglicht und den Benutzer über gefährliche Vorgänge in seinem Haus informiert.

Im Rahmen der Entwicklung dieses Systems wurden die aktuellen Technologien zur Erstellung eines Webinterfaces und zur Kommunikation zwischen den Geräten eingesetzt. **TODO Kurz erklären was in Kapitel 2,3,4,5 ... erklärt wird**

2 Grundlagen & Theorie

2.1 Hardware

2.1.1 Arduino

2.1.2 ESP8266

2.1.3 Raspberry Pi

2.1.4 M5Stick

2.1.5 ESP32

2.2 Kommunikationsprotokolle

2.2.1 HTTP

2.2.2 MQTT

2.3 Software

2.3.1 Asyncio

2.3.2 FastAPI

2.3.3 SQLAlchemy

2.3.4 Uvicorn

2.3.5 HTML & TailwindCSS

2.3.6 Javascript

2.3.7 WebSocket

2.3.8 Linux-Pakete für Raspberry Pi

3 Konzeption des Systems

Übersicht über das Gesamtsystem, Systemarchitektur und Datenfluss, Anforderungen und Ziele

4 Implementierung und praktische Umsetzung

4.1 Einrichtung der Hardwarekomponenten

4.1.1 Arduino

4.1.2 ESP8266

4.1.3 Raspberry Pi

4.1.4 M5Stick

4.1.5 ESP32

4.2 Softwareentwicklung

4.2.1 Webserver

4.3 Datenbank

4.4 Frontend

4.5 Integration der Komponenten

MQtt client, erhalten Daten von Broker etc

5 Ergebnisse und Diskussion

6 Quellen

Literatur

- [1] O. Baida, *Anbindung der Sensoren und Aktoren an den Arduino zur Realisierung eines Sicherheitssystems*, Projektarbeit 1, 2024.
- [2] O. Baida, Projektordner für PA2: <https://github.com/oleksiibaida/PA2.git>
- [3] O. Baida, Demo-Video: https://youtu.be/-U1_ye1KLy0
- [4] Statista, „*Smart Home - Anzahl der Haushalte in Deutschland 2028*“, Zugriffen: 13. Januar 2025. [Online]. Verfügbar unter <https://de.statista.com/prognosen/885611/anzahl-der-smart-home-haushalte-in-deutschland>
- [5] J. Breithut, „*Strom und Heizung: Wann ein Smart Home wirklich beim Energiesparen hilft*“, Der Spiegel, 17. Juli 2022. Zugriffen: 13. Januar 2025. [Online]. Verfügbar unter: <https://www.spiegel.de/netzwelt/gadgets/strom-und-heizung-wann-ein-smart-home-wirklich-beim-energiesparen-hilft-a-ffb4b710->

Links zur verwendeten Hardware:

- [6] Arduino.cc, *Arduino UNO*, <https://docs.arduino.cc/hardware/uno-rev3/>
- [7] Raspberry Pi Foundation, *Raspberry Pi 1 B+*, <https://www.raspberrypi.com/products/raspberry-pi-1-model-b-plus/>
- [8] Espressif, *ESP8266*, <https://www.espressif.com/>, <https://www.electronicwings.com/sensors-modules/esp8266-wifi-module>

Links zur verwendeten Software:

- [9] Dr Andy Stanford-Clark, Arlen Nipper, *Message Queuing Telemetry Transport*, <https://mqtt.org/>
- [10] Guido van Rossum, Python Software Foundation, *Python*, <https://www.python.org/>
- [11] Telegram FZ-LLC, *Telegram Messenger*, <https://github.com//telegramdesktop/tdesktop>

Linux-Packete:

- [12] Jouni Malinen, *hostapd*, <https://w1.fi/hostapd/>, Zugriff am: 19. September 2024.
- [13] Simon Kelley, *dnsmasq*, <https://dnsmasq.org/doc.html>, Zugriff am: 20. September 2024.
- [14] Eclipse Foundation, *Eclipse Mosquitto*, <https://mosquitto.org/>

ESP- und Arduino-Bibliotheken

- [15] Knolleary, *PubSubClient*, <https://pubsubclient.knolleary.net/>, Zugriff am: 21. Oktober 2024.
- [16] ESPWIFI.h, <https://arduino-esp8266.readthedocs.io/en/latest/esp8266wifi/readme.html>
- [17] EEPROM.h, <https://docs.arduino.cc/learn/built-in-libraries/eeprom/>

- [18] Keypad.h <https://docs.arduino.cc/libraries/keypad/>
- [19] R. Scholz, *Syncloop*, Persönliche Mitteilungen
Python-Bibliotheken
- [20] Pierre Fersing, Roger Light *paho-mqtt*, <https://pypi.org/project/paho-mqtt/>, Zugriff am: 21. Oktober 2024.
- [21] Open Source, *python-telegram-bot*, <https://docs.python-telegram-bot.org/en/v21.6/>
- [22] Python Software Foundation, *json*, <https://docs.python.org/3/library/json.html>
- [23] Python Software Foundation, *threading*, <https://docs.python.org/3/library/threading.html>
- [24] Python Software Foundation, *queue*, <https://docs.python.org/3/library/queue.html>
- [25] Gerhard Häring, *sqlite3*, <https://docs.python.org/3/library/sqlite3.html>
- [26] Lawrence Hudson, *pyzbar*, <https://github.com/NaturalHistoryMuseum/pyzbar/>
- [27] Intel, *OpenCV*, <https://github.com/opencv/opencv-python>
- [28] Aio-Libs, *aiohttp*, <https://github.com/aio-lib/aiohttp>

Abbildungsverzeichnis

Tabellenverzeichnis

Programmcode