

Home automation and control using the ZigBee protocol

VOICU, BABICIU^a

^aTechnical University of Cluj-Napoca,
North University Center of Baia Mare,
Faculty of Science,
Department of Mathematics and Computer Science,
Baia Mare, Romania

Email addresses: voicubabiciu@gmail.com (VOICU,
BABICIU)

Abstract

The ZigBee protocol is a popular choice for the management of intelligent devices, due to its low power consumption and interoperability. In this project, a system for managing smart devices using this protocol was developed. The main objective of the project is to integrate as many smart devices from different manufacturers as possible. The application allows the user to control various functionalities of devices installed in their homes, such as lighting and temperature. Additionally, the application provides information about the status of each device, allowing the user to troubleshoot any issues that may arise. Overall, this project demonstrated the feasibility of using ZigBee for the management of intelligent devices and provided a valuable tool for home control.

Keywords: ZigBee, BLE, MQTT, Z-Stack, CC2652P2, ZigBee2MQTT



1. Introducere

Industria caselor inteligente s-a dezvoltat rapid în ultimii ani, datorită progreselor în tehnologia wireless. Sistemele de automatizare a locuinței permit proprietarilor de case să controleze o gamă variată de dispozitive și aparate din locuințele lor de la distanță, folosind un smartphone.

Din ce în ce mai multe grupuri de cercetare lucrează în acest domeniu, cum ar fi MIT, Siemens, Cisco, IBM, Xerox, Microsoft etc. În cadrul acestor grupuri, au fost înființate aproape 20 de laboratoare pentru locuințe. În aceste laboratoare la domiciliu, sunt utilizate peste 30 de aparate, sunt produse peste 5 protocoale de rețea și au fost utilizate peste 3 tehnici de inteligență artificială. În această lucrare, se va acorda atenție în principal rețelelor casnice inteligente, gateway-urilor și aparatelor casnice inteligente. Atenția acordată acestui concept a provocat atât o creștere semnificativă în rândul producătorilor de aparatură casnică inteligentă cât și extinderea unor companii existente pe acest sector de piață, câteva exemple fiind: Google, Apple, Bosch, Amazon, Samsung, Ikea, Philips, etc.

În septembrie 2003, Housing Learning & Improvement Network (Rețeaua de învățare și îmbunătățire a locuințelor) a publicat o definiție a casei inteligente oferită de Intertek, care a realizat proiectul numit DTI Smart Homes Project (Proiectul DTI pentru case inteligente). În viziunea Intertek, o casă inteligentă este "o locuință care încorporează o rețea de comunicații care conectează principalele aparate și servicii electrice și permite controlul, monitorizarea sau accesarea acestora de la distanță", prin "la distanță" înțelegându-se aici atât din interiorul locuinței, cât și din afara acesteia. Prin urmare, o locuință inteligentă trebuie să conțină trei elemente: rețeaua internă, controlul inteligent și automatizarea locuinței. Rețeaua internă este baza unei case inteligente și poate fi cu fir, prin cablu și fără fir. Controlul inteligent înseamnă gateway-uri pentru gestionarea sistemelor. Automatizarea locuinței înseamnă produsele din interiorul locuințelor și legăturile cu serviciile și sistemele din afara locuinței [1].

2. Scopul și Obiectivele Proiectului

2.1. Scop

Proiectul se adresează, în general, utilizatorilor casnici, dar poate fi implementat cu ușurință și în clădiri corporative sau locuințe sociale, care necesită control și monitorizare de la distanță. Sistemul poate fi utilizat pentru a oferi o avertizare timpurie cu privire la pericolele potențiale, cum ar fi scurgerile de gaz, incendiile și alertele de intruziune. În plus, sistemul poate fi utilizat pentru a monitoriza consumul de energie și pentru a optimiza sistemele de încălzire, ventilație și aer condiționat (HVAC). Proiectul utilizează tehnologie de ultimă generație pentru a oferi o soluție fiabilă și rentabilă pentru automatizarea locuinței.

Așadar scopul proiectului este de a oferi posibilitatea integrării cu dispozitive inteligente de la diverși producători care utilizează același protocol, reducând astfel numărul de aplicații necesare pentru a le controla și costul achiziționării de gateway-uri suplimentare.

2.2. Obiective

Pe măsură ce piața caselor inteligente continuă să crească, este important ca producătorii să se concentreze pe crearea unor soluții care nu sunt doar avansate, ci și ușor de utilizat. O experiență de utilizare excelentă este esențială pentru orice tip de tehnologie, iar casa inteligentă nu face excepție.

Principalele obiective:

- Interfață ușor de utilizat și multilingv pentru o integrare cât mai bună la nivel global.
- Posibilitatea creării unui cont de utilizator.
- Posibilitatea de creare și gestionare a camerelor imobilului.
- Posibilitatea adăugării și gestionării a cât mai multe dispozitive inteligente care operează pe același protocol.
- Controlul dispozitivelor de la distanță.

3. Analiză și Fundamentare Teoretică

3.1. Protocolul ZigBee

Utilizarea ZigBee pentru interconectarea dispozitivelor accesibile face ca propagarea și protocolarea datelor să fie transparente, sigure și asigurate. Schimbul de date ZigBee utilizează frecvența radio de 2,4 GHz, banda ISM, are un consum redus de energie, un cost redus, permite personalizări în vederea îmbunătățirii și este utilizat în mare parte în chipset-urile 802.15.4[2].

ZigBee este un protocol de rețea fără fir dezvoltat de Comitetul IEEE și ZigBee Alliance care nu necesită viteze de transfer de date atât de mari ca cele permise de Bluetooth. În plus, protocolul poate fi implementat în rețele mesh[3].

Avantajele utilizării ZigBee sunt:

- Fiabil
- Suportă un număr mare de noduri
- Ușor de implementat
- Durată de viață foarte lungă a bateriei
- Siguranță
- Cost redus
- Poate fi utilizat la nivel global
- Sprijin vibrant din partea industriei, cu treizeci sau mai multe furnizori care furnizează produse și servicii
- Controlul dispozitivelor de la distanță.
- Protocol cu standarde deschise, fără taxe de licență sau cu taxe de licență neglijabile
- Chipset-uri disponibile din mai multe surse
- Firmware actualizabil de la distanță

3.2. Tipuri de dispozitive logice ZigBee

3.2.1. Coordonator

Formează rădăcina arborelui de rețea și poate face legătura cu alte rețele. Există exact un coordonator în fiecare rețea. Acesta este responsabil de inițierea rețelei și de selectarea parametrilor de rețea, cum ar fi canalul de radiofrecvență, identificatorul unic al rețelei și stabilirea altor parametri operaționali. De asemenea, acesta poate stoca informații despre rețea, chei de securitate [4].

3.2.2. Router

Acționează ca nod intermediar, retransmițând date de la alte dispozitive. Routerul se poate conecta la o rețea deja existentă, putând, de asemenea, să accepte conexiuni de la alte dispozitive și să fie un fel de re-transmițător către rețea. Rețeaua poate fi extinsă prin utilizarea de routere ZigBee[4].

3.2.3. Dispozitive finale

Dispozitivele finale pot fi dispozitive cu consum redus de energie/alimentate cu baterii. Acestea pot colecta diverse informații de la senzori și întrerupătoare. Acestea au suficientă funcționalitate pentru a vorbi cu părinții lor (fie coordonatorul, fie un router) și nu pot retransmite date de la alte dispozitive. Această funcționalitate redusă permite potențialul de reducere a costului lor. Acestea acceptă modele mai bune de consum redus de energie. Aceste dispozitive nu trebuie să rămână treze tot timpul, în timp ce dispozitivele aparținând celorlalte două categorii trebuie să o facă. Fiecare dispozitiv final poate avea până la 240 de noduri finale, care sunt separate aplicații separate care împart același radio[4].

3.3. Arhitectura ZigBee

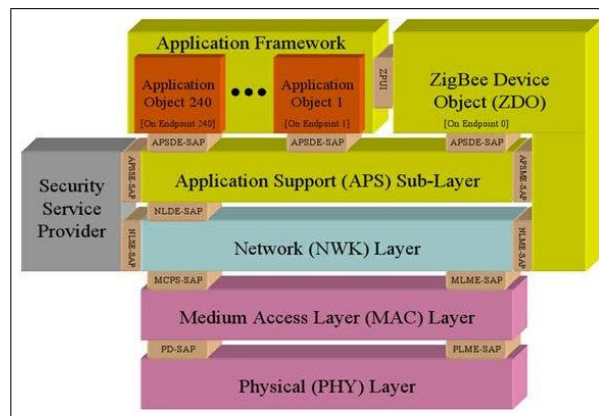


Figura 1: Straturile arhitecturii ZigBee

3.3.1. Stratul Fizic (PHY)

Serviciul de date PHY permite transmiterea și recepția unităților de date de protocol PHY (PPDU) pe canalul radio fizic. Caracteristicile PHY sunt activarea și dezactivarea emițătorului-receptor radio, detectarea energiei (ED), indicarea calității legăturii (LQI), selectarea canalului, evaluarea canalului clar (CCA) și transmiterea și recepționarea pachetelor pe mediul fizic[5].

3.3.2. Medium Access (MAC) Layer

Substratul MAC furnizează două servicii: serviciul de date MAC și serviciul de gestionare MAC care face legătura cu punctul de acces la servicii (SAP) al entității de gestionare a substratului MAC (MLME) (MLMESAP). Serviciul de date MAC permite transmiterea și recepționarea unităților de date de protocol MAC (MPDU) prin serviciul de date PHY. Caracteristicile substratului MAC sunt gestionarea balizelor, accesul la canale, gestionarea GTS, validarea cadrelor, livrarea de cadre confirmate, asocierea și disocierea[5].

3.3.3. Network (NWK) Layer

Se ocupă de toate operațiunile legate de rețea, cum ar fi configurarea rețelei, conectarea și deconectarea dispozitivelor finale la rețea, rutarea, configurarea dispozitivelor etc[5].

3.3.4. Application Support (APS) Sub-Layer

Activează serviciile necesare pentru ca obiectele dispozitiv ZigBee și obiectele de aplicație să aibă o interfață cu straturile de rețea pentru serviciile de gestionare a datelor. Responsabil de potrivirea a două dispozitive în funcție de serviciile și nevoile acestora[5].

3.3.5. Application Framework (AF)

Oferă două tipuri de servicii de date, respectiv servicii de perechi cheie-valoare și servicii de mesaje generice. Mesajul generic este o structură definită de dezvoltator, în timp ce perechea cheie-valoare este utilizată pentru a obține atribute în cadrul obiectelor aplicației[5].

3.3.6. ZigBee Device Object (ZDO)

Este responsabil de gestionarea generală a dispozitivului și este responsabil în special de definirea modului de funcționare a dispozitivului, spunând dacă un dispozitiv coordonează comunicațiile de rețea sau acționează ca un dispozitiv final[5].

3.4. ZigBee CC2652P2

ZigBee CC2652P¹ este un coordonator ZigBee proiectat pentru GPIO-ul Raspberry Pi care rulează Z-Stack Firmware 3.x. CC2652P este actuala generație de microcontrolere pentru coordonatoarele ZigBee 3.x și are un amplificator de putere integrat (+20dBm) pentru o mai bună acoperire. ZigBee CC2652 a fost proiectat special pentru aplicațiile coordonatorului ZigBee și oferă performanțe excelente, consum redus de energie și un factor de formă mic. Acesta este pe deplin compatibil cu standardul ZigBee 3.0 și poate fi utilizat ca un coordonator ZigBee independent sau ca parte a unei rețele ZigBee.

3.5. Firmware-ul Z-Stack

Z-Stack² este firmware-ul pentru SimpleLink™ CC26x2. Acesta conține tot ceea ce este necesar pentru o soluție conformă cu ZigBee, inclusiv instrumente, exemple de aplicații, documentație și cod sursă. Z-Stack include, de asemenea, suport pentru configurarea procesorului de rețea, care poate fi utilizat în arhitecturi cu două cipuri în care gazda implementează biblioteca Cluster și aplicațiile ZigBee. Z-Stack este compatibil cu specificația ZigBee 3.0 și oferă niveluri fără precedent de performanță, rază de acțiune, securitate și eficiență energetică îmbunătățite, stimulând astfel adoptarea ZigBee în aplicații care variază de la automatizarea clădirilor la agricultură și la dispozitive portabile[6]. Z-Stack este liber de drepturi de autor și oferă producătorilor flexibilitatea de a crea soluții proprietare pe baza cadrului de standarde deschise al ZigBee Alliance. Prin urmare, Z-Stack a fost ales ca bază pentru multe soluții ZigBee de generație următoare.

3.6. Bluetooth Low Energy

Bluetooth Low Energy (BLE) este o tehnologie complet nouă, care a fost concepută atât ca o tehnologie complementară la tehnologia Bluetooth clasică, cât și ca o tehnologie wireless cu cea mai mică putere posibilă care poate fi proiectată și construită. Deși folosește marca Bluetooth și împrumută o mulțime de tehnologii de la compania mamă, Bluetooth low energy ar trebui considerată o tehnologie diferită, care se adresează unor obiective de proiectare diferite și unor segmente de piață diferite [7].

Bluetooth Low Energy ia o direcție complet diferită. În loc să mărească ratele de date disponibile, a fost optimizat pentru un consum foarte redus. Acest lucru înseamnă că, nu se vor obține viteze de date ridicate, sau chiar menținerea unei conexiuni timp de mai multe ore sau zile[8].

¹<https://www.ti.com/product/CC2652P>

²Documentație Z-Stack <https://usermanual.wiki/Pdf/ZStack203020Developers20guide.1808814057/view>

3.7. ZigBee2MQTT

ZigBee2MQTT³ este un instrument open-source care permite utilizarea dispozitivelor ZigBee fără a avea nevoie de o punte sau de un gateway proprietar din partea producătorului. ZigBee2MQTT creează o punte între dispozitivele ZigBee și MQTT (a se vedea figura 2, ceea ce vă permite integrarea dispozitivelor ZigBee cu orice infrastructură de automatizare a locuinței pe care o utilizați. Acest lucru face posibilă utilizarea dispozitivelor ZigBee cu orice număr de sisteme de automatizare a locuinței existente, precum și cu configurații personalizate. ZigBee2MQTT este o soluție flexibilă și puternică pentru cei care doresc să utilizeze dispozitive ZigBee în configurația lor de automatizare a locuinței.

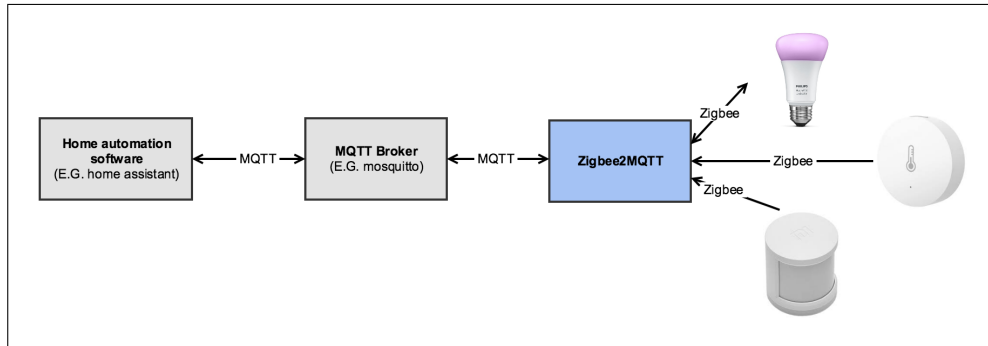


Figura 2: Arhitectura ZigBee2MQTT

ZigBee2MQTT se integrează bine cu (aproape) orice soluție de automatizare a locuinței, care utilizează MQTT, două dintre acestea fiind HomeAssistant⁴ și Domoticz⁵.

4. Proiectare de Detaliu și Implementare

4.1. Specificațiile sistemului

S-a dorit realizarea unui sistem care să fie capabil să ofere o soluție robustă și versatilă pentru controlul dispozitivelor ZigBee într-un mediu casnic și/sau business.

Proiectul este conceput pentru a elimina nevoia de coordonatori ZigBee suplimentari dintr-un imobil casnic pentru a permite utilizarea dispozitivelor inteligente de la diferiți producători. Sistemul utilizează un Raspberry Pi 4 Model B+⁶ echipat cu un shield ZigBee având ca și controler central chipset-ul CC2652P. A fost dezvoltat un script Python pentru a stabili și menține comunicarea cu toate nodurile ZigBee din rețea. Scriptul expune, de asemenea, o serie de servicii bluetooth care permit configurarea coordonatorului din cadrul aplicației mobile.

De asemenea, sistemul este capabil să se refacă automat în cazul întreruperilor de rețea și poate gestiona în mod dinamic aderarea și părăsirea nodurilor la rețea.

4.2. Arhitectura sistemului

Baza de date MongoDB este utilizată pentru a stoca toate datele colectate de sistem. Acestea pot fi accesate prin REST API și MQTT Broker, ceea ce permite aplicației mobile să utilizeze datele. Brokerul MQTT este utilizat pentru a oferi un canal de comunicare între aplicația mobilă și coordonator. Aplicația mobilă este utilizată pentru a controla sistemul și pentru a vizualiza datele colectate de sistem. Coordonatorul este responsabil de gestionarea dispozitivelor ZigBee și de colectarea datelor de la acestea. Dispozitivele ZigBee sunt utilizate pentru a colecta date de mediu din casa utilizatorului. Diagrama arhitecturii poate fi observată în figura 3. După cum reiese și din arhitectura sistemului, back-end-ul înglobează serviciul REST și broker-ul MQTT, Acestea au fost dezvoltate folosind **Node.js**, respectiv librăriile **Koa.js** pentru REST API și **Aedes** pentru broker-ul MQTT.

³ZigBee2MQTT <https://www.ZigBee2mqtt.io/>

⁴HomeAssistant <https://www.home-assistant.io/>

⁵Domoticz <https://www.domoticz.com/>

⁶<https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>

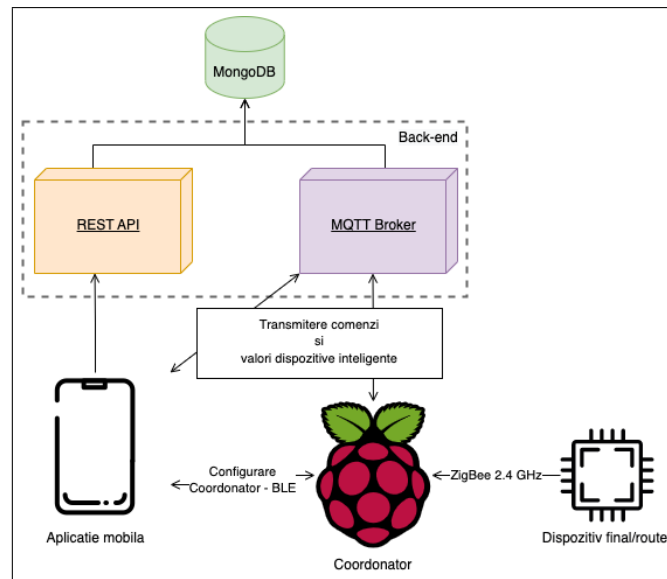


Figura 3: Arhitectura sistemului

4.3. Arhitectura conceptuala REST

REST este o arhitectură care facilitează comunicarea între sisteme. Prin utilizarea unui model client-server, REST permite modificarea diferitelor părți ale sistemului fără a le afecta pe celelalte. Acest lucru se datorează faptului că fiecare modul știe cum să comunice cu celălalt și ce format de informații să folosească. Prin urmare, modificările sau noile cerințe pot fi implementate fără a afecta în vreun fel funcționarea celeilalte părți. Atâta timp cât fiecare modul știe cum să comunice cu celălalt și ce format de informații să utilizeze, acestea pot fi ținute separate.[9] Acesta este un mare avantaj, deoarece orice parte a sistemului poate fi modificată cu schimbări sau cerințe noi fără a afecta în vreun fel funcționarea celeilalte părți. În consecință, REST este o alegere excelentă pentru sistemele mari și complexe.

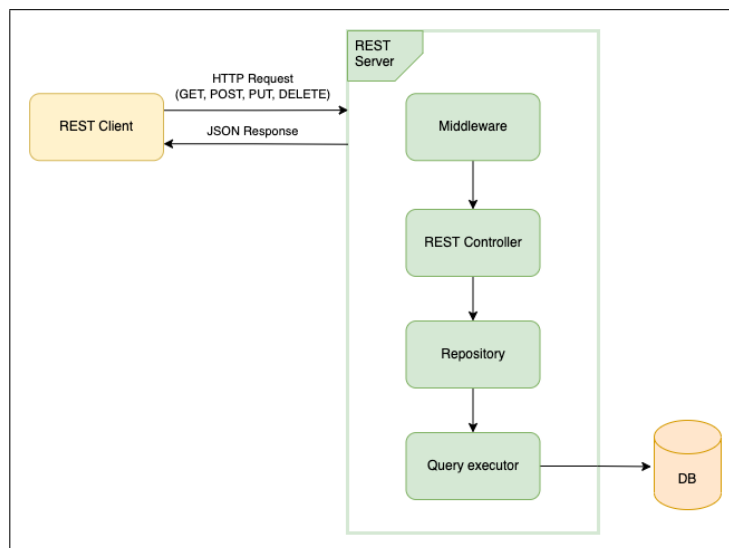


Figura 4: Arhitectura REST a sistemului

4.4. Arhitectura conceptuala MQTT

Pentru transmiterea de date către și de la dispozitive inteligente, a fost ales protocolul MQTT. Arhitectura broker-ului poate fi observată în figura 5.

Principalul avantaj al MQTT este consumul extrem de redus de lățime de bandă în rețea. Acest lucru se datorează faptului că utilizează un model de mesagerie foarte simplu și eficient de publicare/subscriere. În acest model, fiecare client se abonează la un set de subiecte și primește mesaje. Acest lucru reduce consumul de lățime de bandă al rețelei[10].

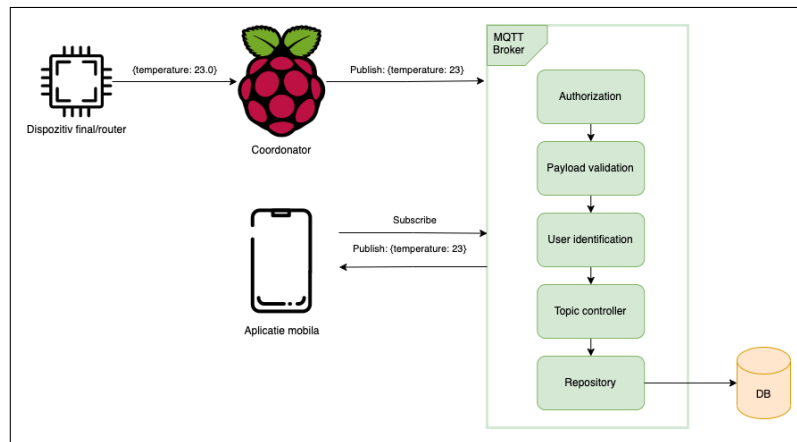


Figura 5: Arhitectura MQTT a sistemului

4.5. Arhitectura coordonatorului

Asa cum a fost menționat și anterior coordonatorul ZigBee este alcătuit dintr-o placă Raspberry Pi 4 având conectat la pinii GPIO un shield cu chipsetul CC2652P⁷. Pentru a fi posibilă comunicarea prin protocolul UART dintre cele două componente chipsetul a fost flash-uit cu firmware-ul Z-Stack 3. Schema modului poate fi vizualizată în figura 6.

Adițional serviciul ZigBee2MQTT a mai fost implementat un script Python responsabil de configurarea serviciului folosind informațiile utilizatorului precum și de resetarea acestuia, în cazul în care se dorește reconfigurarea rețelei ZigBee.

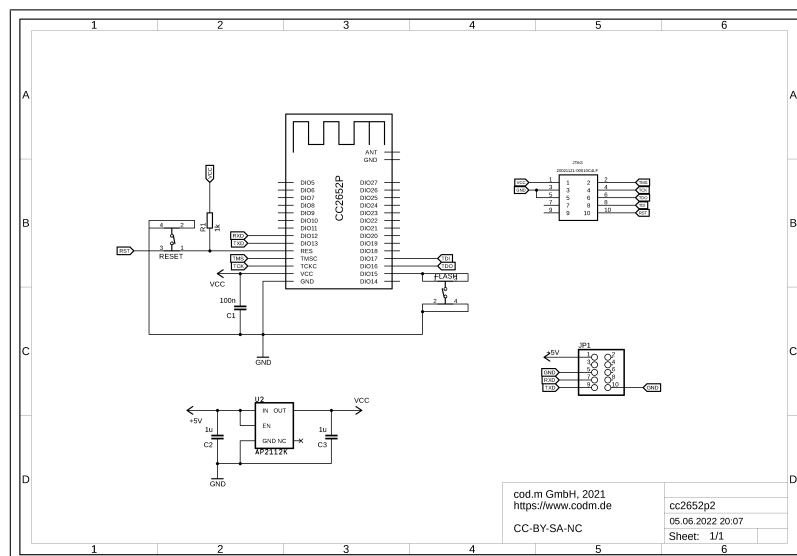


Figura 6: Schema coordonator ZigBee

⁷<https://github.com/codm/cc2652-raspberry-pi-module>

4.6. Autentificare

Pentru autentificarea utilizatorului în sistem acesta trebuie să furnizeze adresa de email și parola folosită la crearea contului. Server-ul HTTP va procesa cererea clientului iar dacă credențialele furnizate sunt corecte acesta va răspunde cu un token de sesiune care va fi folosit pentru viitoarele cereri. Diagrama de secvență pentru procesul de autentificare al unui client poate fi observată în figura 7.

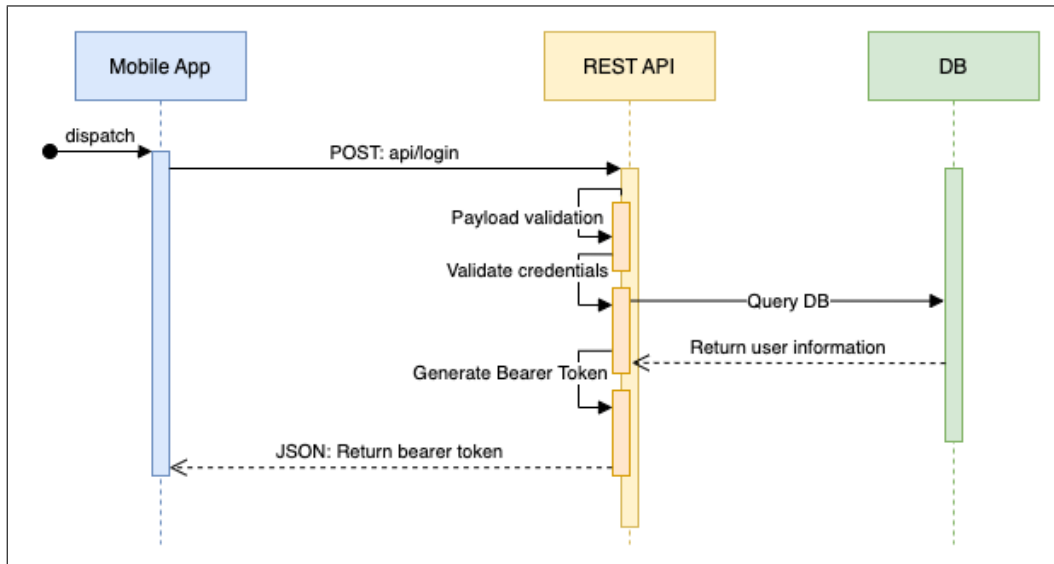


Figura 7: Diagrama de secvență login

4.7. Configurare coordonator

Asa cum a fost precizat și în secțiunea 4.1 coordonatorul ZigBee este configurat prin BLE. Pentru ca acest lucru să fie posibil a fost nevoie să se definească o serie de servicii și caracteristici Bluetooth Low Energy. Identificarea acestora se face după un cod unic de identificare pe 16 biți numit **UUID (Universally Unique Identifier)** [11]. O serie de astfel de coduri UUID care au fost deja atribuite unor producători este disponibilă pe internet.⁸

Așadar serviciile și caracteristicile care au fost definite pot fi văzute în tabelul 1.

Tabela 1: Servicii și caracteristici BLE

UUID	UUID Serviciu	Descriere
Servicii		
0x1829		Serviciu folosit pentru configurarea coordonatorului
Caracteristici		
d64c143e-6cbd-11eb-9439-0242ac130002	0x1829	Caracteristică utilizată pentru scrierea configurărilor pe coordonator
5b0d3cfc-6cbe-11eb-9439-0242ac130002	0x1829	Caracteristică utilizată pentru citirea configurărilor de pe coordonator

Pentru a se putea stabili o conexiune stabilă BLE cu coordonatorul ZigBee trebuie să se urmeze o serie de pași. Așadar aceștia sunt descriși în lista următoare

⁸<https://btprodspecificationrefs.blob.core.windows.net/assigned-values/16-bit%20UUID%20Numbers%20Document.pdf>

- **Scanare dispozitive** - Scanarea se va efectua filtrând dispozitivele BLE existente după informațiile producătorului (id, nume) implementate pe Raspberry PI, în cazul de față fiind id: **0x1B58** și nume: **housey.ro**.
- **Conectare la dispozitiv**
- **Descoperire servicii** - acest lucru v-a interoga perifericul bluetooth pentru toate serviciile pe care acesta le oferă. În cazul de față v-a returna serviciul prezentat în tabelul 1.
- **Descoperire caracteristici** - fiecare serviciu poate avea o serie de caracteristici, în cazul nostru ambele caracteristici sunt atribuite aceluiași serviciu.
- **Scrierea unei caracteristici** - În funcție de flag-urile pe care le are setate o caracteristica, acesta poate suporta una sau mai multe funcții:
 - Citire (Read)
 - Scriere (Write)
 - Notificare (Notify)

O caracteristica poate emite valori la un interval de timp specificat. Tot o data notificarea poate fi oprită atât de către perifericul bluetooth, cât și de clientul conectat la acesta. În cazul sistemului implementat o data ce configurarea coordonatorului a început, caracteristica pentru citirea datelor v-a începe sa notifice statusul configurării la un interval de 3 secunde. O data ce configurarea a fost realizată cu succes aceasta se va opri.

5. Înrolare dispozitiv inteligent în rețeaua ZigBee

Pentru înrolarea unui nou dispozitiv în rețea este necesar ca acesta sa fie suportat de către serviciul ZigBee2MQTT ⁹. În cazul în care se dorește înrolarea unui dispozitiv care nu este suportat se pot urma pașii descriși în documentație ¹⁰.

6. Concluzii

Sistemul este conceput pentru a funcționa cu o varietate de dispozitive ZigBee, inclusiv becuri, prize și întrerupătoare. De asemenea, poate fi utilizat pentru a controla alte dispozitive care utilizează profilul ZigBee, cum ar fi termostatele și încuietorile ușilor. Sistemul include un controler central care se conectează la rețeaua casnică și la o serie de dispozitive compatibile. Controlerul este responsabil de gestionarea dispozitivelor ZigBee și de furnizarea unei interfețe pentru utilizator. Acesta include, de asemenea, o aplicație mobilă care permite utilizatorului să controleze de la distanță dispozitivele ZigBee. Sistemul este ușor de instalat și de configurat. Aplicația mobilă oferă o interfață simplă pentru controlul dispozitivelor ZigBee. În general, proiectul oferă o soluție robustă și versatilă pentru controlul dispozitivelor ZigBee într-un mediu casnic.

6.1. Contribuții

- Posibilitatea de integrare a dispozitivelor smart de la diferiți producători într-un singur ecosistem.
- Posibilitatea de găzduire a sistemului local
- Posibilitatea de integrare a dispozitivelor proprii

⁹Dispozitive suportate: <https://www.ZigBee2mqtt.io/supported-devices/>

¹⁰ZigBee2MQTT înrolare dispozitive noi: https://www.ZigBee2mqtt.io/advanced/support-new-devices/01_support_new_devices.html

6.2. Dezvoltări ulterioare

Se dorește continuarea proiectului prin implementarea următoarelor funcționalități:

- Posibilitatea creării de scene inteligente
- Sistem de notificare push
- Integrare cu Google Assistant și Amazon Alexa
- Monitorizare consum locuință
- Analizarea datelor de la senzori și creare statistici
- Dashboard web

Bibliografie

- [1] R. Harper, *Inside the smart home*. Springer Science & Business Media, 2006.
- [2] H. G. C. Ferreira, E. D. Canedo, and R. T. De Sousa, "Iot architecture to enable intercommunication through rest api and upnp using ip, zigbee and arduino," in *2013 IEEE 9th international conference on wireless and mobile computing, networking and communications (WiMob)*. IEEE, 2013, pp. 53–60.
- [3] S. Safaric and K. Malaric, "Zigbee wireless standard," in *Proceedings ELMAR 2006*. IEEE, 2006, pp. 259–262.
- [4] K. Gill, S.-H. Yang, F. Yao, and X. Lu, "A zigbee-based home automation system," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 2, pp. 422–430, 2009.
- [5] C. M. Ramya, M. Shanmugaraj, and R. Prabakaran, "Study on zigbee technology," in *2011 3rd international conference on electronics computer technology*, vol. 6. IEEE, 2011, pp. 297–301.
- [6] T. Rust, "Fuzzing zigbee using z-stack," 2022.
- [7] N. K. Gupta, *Inside Bluetooth low energy*. Artech House, 2016.
- [8] A. Kurniawan, "Bluetooth low energy (ble)," in *Beginning Arduino Nano 33 IoT*. Springer, 2021, pp. 157–181.
- [9] S. Kumari *et al.*, "Rest based api," *International Journal of Trend in Scientific Research and Development*, vol. 1, no. 4, pp. 571–575, 2017.
- [10] U. Hunkeler, H. L. Truong, and A. Stanford-Clark, "Mqtt-s—a publish/subscribe protocol for wireless sensor networks," in *2008 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE'08)*. IEEE, 2008, pp. 791–798.
- [11] C. Zuo, H. Wen, Z. Lin, and Y. Zhang, "Automatic fingerprinting of vulnerable ble iot devices with static uuids from mobile apps," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 1469–1483.