

Sistema Ubíquo de Estores Inteligentes

1st João Lopes

ISUS group

FCT/UFP

Porto, Portugal

37133@ufp.edu.pt

2nd Rúben Pinto

ISUS group

FCT/UFP

Porto, Portugal

40115@ufp.edu.pt

3rd Rui S. Moreira

ISUS group

FCT/UFP, INESC TEC and LIACC

Porto, Portugal

rmoreira@ufp.edu.pt (0000-0002-4123-0983)

Resumo—Este paper apresenta uma solução IoT para regular estores de acordo com a luminosidade recebida do exterior de forma automática ou manual no contexto de uma casa inteligente. Para tal, diversas tecnologias de redes sem fios e de comunicação foram exploradas de forma a atingir os objetivos pretendidos. Estas tecnologias incluem o uso de um Arduino com um módulo XBee usado para controlar o motor do estore, receber a informação do sensor de luminosidade e comunicar estes dados para um Raspberry Pi via Zigbee. O Raspberry Pi, com recurso a uma Pycom com um módulo LoPy para comunicação via LoRa, e o protocolo de mensagens MQTT, os dados são enviados para o serviço ThingSpeak.

Index Terms—Mobile and Ubiquitous Computing, Pervasive Computing, Smart Home, Smart Blinds, Context Aware System, Internet of Things (IoT)

I. INTRODUÇÃO

Com o avanço das tecnologias da Internet das Coisas (IoT) e das redes móveis, cada vez mais surgem soluções para ajudar as pessoas a gerir as suas vidas de formas diversas. Uma das maiores tendências são os utensílios inteligentes. Temos como exemplos destes utensílios os aspiradores robô, esfregonas robô, purificadores de ar inteligentes, luzes inteligentes, termostatos, entre outros dispositivos que possam ser usados numa casa. Uma solução de estores inteligentes foi aqui estudada de forma a que, dependendo da iluminação exterior, os estores sejam automaticamente abertos ou fechados.

Esta solução utiliza um sensor de luminosidade e o motor do estore ligados a um Arduino com um módulo Zigbee, um Raspberry Pi, um Pycom com um módulo LoPy e o protocolo MQTT para a troca de mensagens tanto dentro da casa como para o serviço externo ThingSpeak.

Este paper está estruturado da seguinte forma: a Secção II apresenta trabalhos relacionados que exploram soluções de sistemas inteligentes com funcionalidades semelhantes. A Secção III a arquitetura do sistema que se pretende implementar seguindo os objetivos e requisitos estabelecidos. Na secção IV é aprofundada a implementação do sistema seguindo as indicações da Secção 3. Na Secção V é apresentada uma proposta sobre como pode o sistema ser avaliado. A Secção 5 apresenta as conclusões tiradas deste paper, referindo as vantagens de existir um sistema como este e que trabalho futuro pode ser realizado para estender esta solução.

This project was supported by FCT/UFP under the scope of CM course

II. ESTADO DA ARTE

Numa breve pesquisa, não encontramos estudos específicos de estores inteligentes. Foi então necessário investigar outros sistemas semelhantes, como luzes inteligentes.

Em [1] é apresentada uma proposta de controlo e monitorização de luzes por uma aplicação móvel, sendo possível controlar as luzes mesmo fora de casa. Em suma, as luzes estão ligadas a um Relay Module que recebe corrente de um Arduino caso o estado da luz seja ligado. O Arduino, para além de enviar corrente, comunica o id da luz e o seu estado a um Raspberry Pi 3. O Raspberry Pi 3 verifica o estado com um servidor. O utilizador pode, a partir de uma aplicação móvel, controlar o estado das luzes da casa comunicando com o servidor. Infelizmente, em nenhum momento são mencionados quais os protocolos de comunicação são usados, mas temos um exemplo de como o sistema é montado a nível de hardware e software.

A estrutura do sistema de [2] é semelhante à de [1], mas não é específica a nenhum tipo de dispositivo IoT. Trata-se de desenhar e implementar um sistema de controlo de uma casa inteligente. Aqui há vários tipos de dispositivos IoT diferentes, ligados a um terminal Zigbee. A informação desta intranet passa pelo coordenador Zigbee para uma gateway de envia os dados para um servidor na cloud por TCP/IP. O utilizador pode a partir do seu telemóvel ou PC controlar todos os seus dispositivos. Este estudo, tal como o próprio autor referiu, pode ser usado como referência para desenvolver um sistema de controlo de casa inteligente a baixo custo.

III. DESCRIÇÃO DO SISTEMA

A. Objetivos e Requisitos

A proposta deve respeitar os seguintes requisitos.

- Deve abrir ou fechar o estore de acordo com a luminosidade detetada.
- O sistema deve ser automático.
- Consumo energético deve ser baixo.
- A comunicação do estado do do sensor de luminosidade deve ser feita periodicamente.

B. Arquitetura do Sistema

O sistema tem como base estores que são controlados por um Arduino. Os Arduinos devem comunicar via Zigbee para um coordenador, ligado a um microcomputador Raspberry Pi.

O Raspberry Pi manda por Wi-Fi este valor para um Pycom. O Pycom no final envia estes dados para o ThingSpeak para que sejam processados. Podemos ver isto representado no esquema da Figura 1.



Figura 1. Visão geral da arquitetura do sistema ao nível do hardware. Da esquerda para a direita: Arduino (estore), Raspberry Pi, serviço ThingSpeak.

IV. IMPLEMENTAÇÃO DO SISTEMA

Ao Arduino estão ligados o sensor de luminosidade e o motor. E, para não estar dependente da eletricidade de casa, tem uma bateria que é carregada por um painel solar direcionado para a janela. Para o desenvolvimento do código do Arduino, foi usado o Arduino IDE. O sensor envia valores entre 0 e 4096, que são depois convertidos para valores inteiros entre 0 e 100 para que sejam interpretados como percentagens. Quando o estore está aberto e luminosidade atinge um valor de 30% ou menos, o estore fecha. Quando está fechado, se detetar um valor de 50% ou mais, o estore abre. Foi também instalado um módulo XBee no Arduino para comunicar via Zigbee com uma Raspberry Pi.

A Raspberry Pi é o centro do sistema. Esta pode estar ligada diretamente à eletricidade de casa e existe para poder receber os dados dos estores com o módulo XBee coordenador. Estes dados devem de seguida ser passados para uma Pycom, com um módulo LoPy, para que possam ser redirecionados para a plataforma ThingSpeak via LoRa. Toda esta troca de mensagens requer o protocolo de mensagens IoT Message Queuing Telemetry Transport (MQTT). Usando o Mosquitto, é iniciado um MQTT broker na Raspberry Pi. O Zigbee2MQTT é usado para fazer a publicação das mensagens recebidas por Zigbee para um tópico do MQTT.

O módulo LoPy permite a comunicação por 3 protocolos distintos: WiFi, BLE e LoRa. Para comunicar com a Raspberry Pi e ter contacto com o MQTT broker, foi estabelecida uma ligação via WiFi. O Pycom fica subscrito ao tópico onde são enviadas as mensagens dos estores. Quando recebe dados da fila, o Pycom redireciona-os para uma LoRa gateway para chegar ao ThingSpeak. E para tal, deve estar também conectado ao MQTT broker do serviço e fazer a publicação da mensagem para o tópico indicado na plataforma.

V. AVALIAÇÃO DO SISTEMA

Dado que o sistema foi proposto mas não desenvolvido, é possível apenas indicar como a avaliação pode ser realizada.

Sendo que estes estores recorrem a baterias, é importante avaliar todos os aspetos relacionados com o consumo energético. É relevante saber quanto o Arduino consome para se poder criar previsões de quanto tempo a bateria irá durar sem e com um painel solar. É importante salientar que

painéis solares em diferentes regiões de uma casa pode ou não fornecer energia suficiente à bateria. Dados sobre a energia que estes conseguem adquirir em diferentes condições podem vir a ser úteis para complementar estas previsões.

Outro aspeto a testar é o tempo de comunicação dos Arduinos. Uma das maiores fontes de consumo energético são as comunicações sem fios. Portanto, quanto menos tempo o Arduino passar a comunicar, melhor. É importante saber quanto tempo é usado neste processo.

VI. CONCLUSÃO

Em casas inteligentes existem vários tipos de dispositivos inteligentes, mas durante as nossas pesquisas não encontramos nenhum estudo que procure solucionar a automatização de estores. Isto é algo que pode complementar qualquer casa inteligente, para cada vez mais a tornar num local que serve o seu dono. Esta solução pode ser facilmente estendida de várias formas em trabalho futuro. Seria interessante desenvolver para que os dados no ThingSpeak sejam enviados para um servidor para que o utilizador possa aceder por uma aplicação móvel. O utilizador poderia modificar os valores de abertura e fecho automáticos dos estores, modificar o estado do estore de forma mais analógica e não apenas abrir/fechar de forma a manter um certo nível de luminosidade dentro da divisão, e permitir ao utilizador alterar manualmente o estado do motor com a aplicação.

REFERÊNCIAS

- [1] A. Efendi, A. Siswanto and A. Sudarman, "Application Control and Monitoring of Light Usage in Smart Home Environment," 2018 Third International Conference on Informatics and Computing (ICIC), Palembang, Indonesia, 2018, pp. 1-5, doi: 10.1109/IAC.2018.8780546.
- [2] X. Mao, K. Li, Z. Zhang and J. Liang, "Design and implementation of a new smart home control system based on internet of things," 2017 International Smart Cities Conference (ISC2), Wuxi, China, 2017, pp. 1-5, doi: 10.1109/ISC2.2017.8090790.