

Detecção de Proximidade de Dispositivos *WiFi* com ESP8266 utilizando Intensidade de Sinal Recebido

Francisco I. de S. Alves¹, Matheus S. Lima¹, Sandro C. S. Jucá¹

¹Eixo de Computação - Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE)

61936-000– Maracanaú – CE – Brasil

{ivanalves94, matheusdslhc}@gmail.com, sandrojuca@ifce.edu.br

Abstract. *Today, among the vast areas of computing research there is growing interest in the area of device localization indoors, where growth is tied to advances in wireless technologies. The aim of this paper is to demonstrate the importance and development of an application that uses ESP8266 with the detection of other WiFi devices. The present work aims to estimate the distance between an ESP8266 (access point) and a smartphone. Android with an application developed using the Ionic framework, and furthermore activate trigger LEDs withMQTT..*

Resumo. *Atualmente dentre as vastas áreas de pesquisa da computação existe um crescente interesse na área de localização de dispositivos em ambientes internos, onde crescimento está atrelado aos avanços de tecnologias sem fio. O propósito deste trabalho é demonstrar a importância, e o desenvolvimento de uma aplicação que usa ESP8266 com intuito de detectar proximidade de outros dispositivos com WiFi. O presente trabalho tem como principal objetivo estimar a distância entre um ESP8266(em modo Access Point) e um smartphone Android com um aplicativo desenvolvido com uso do framework Ionic, e a partir disso acionar LEDs através de gatilhos com MQTT.*

1. Introdução

Atualmente, a necessidade de estipular localização, seja de pessoas ou objetos, é algo importante em variadas áreas, na computação ou fora dela. Na computação determinar localização é fundamental na robótica, e na computação móvel seu objetivo é fornecer aos usuários o acesso à rede independente da sua localização, entre outros. Fora da computação, o uso termo se fez necessário desde os séculos passados, o que mudou foi o cenário no qual é empregado.

No século XX, houve a grande revolução nessa área causada pela criação do Sistema de Posicionamento Global, ou simplesmente GPS (*Global Positioning System*), a origem desse conceito deve-se ao advento da era espacial, mais precisamente com o lançamento do satélite artificial SPUTINIK I, pelos russos, quando se começou a estudar os sinais de rádio para fins de navegação[Sena 2016].

O GPS tornou-se ferramenta mundialmente utilizada em ambientes externos, porém mostrou-se ineficaz em ambientes internos, devido ao fato que esses espaços possuem concreto ou revestimento metálico, o que influencia negativamente a forma que os sinais são transmitidos pelos satélites, que por sua vez são peças fundamentais nesta tecnologia. Dado essa impossibilidade, desenvolveu-se outras ferramentas e tecnologias que pudessem empenhar trabalho mais satisfatório em ambientes internos.

O objetivo deste trabalho é analisar diversas tecnologias utilizadas na localização interna em ambientes, os problemas enfrentados, os métodos mais usados, por fim demonstrar aplicativo para detecção de proximidade de um *smartphone* a um NodeMCU ESP8266.

2. Fundamentação Teórica

O papel dos sistemas de posicionamento tornou-se cada vez mais nítido na vida das pessoas, com a integração das tecnologias utilizadas no cotidiano e com a percepção das suas vantagens.

A gama de possibilidades para a utilização da tecnologia em questão mostrou-se bastante ampla. Em uma loja, por exemplo, pode-se utilizá-lo para oferecer sugestões de produtos e promoções de acordo com o departamento que o cliente está visitando. Em hospital o sistema pode ser utilizado para o monitoramento de pacientes e médicos, podendo ser crucial em emergências [Centro de Computação Gráfica 2018]. Existem sistemas que auxiliam a acessibilidade de portadores de deficiência, como o RFLocus, onde seus usuários localizados em ambientes mapeados são informados via áudio sobre pontos de interesse ao seu redor, como qual cômodo se encontra ao seu redor, ou mesmo a presença de obstáculos como escadas e rampas(fixas) [Cassimiro e Tsuchiaya 2017]

Sistemas para localização interna enfrentam diversos fatores que dificultam sua implementação, e que obviamente precisam extintos ou minimizados, esses fatores causam interferências, consequentemente prejudicam a transmissão e recepção dos sinais.

Uma vez implantado, o desempenho do sistema pode ser medido verificando a sua precisão. Para isso, sempre que for realizada uma navegação, ao chegar ao destino pode-se perguntar se foi alcançado o local correto ou qual a distância aproximada em caso de falha, possibilitando que ajustes sejam feitos a fim de melhorar seu desempenho [Rezende e Ynoguti 2016].

Uma das opções mais utilizadas se chama *Beacon*. Um *beacon* é um minúsculo dispositivo que emite sinais por meio de tecnologia *bluetooth low energy*. Os sinais podem ser captados por aplicativos de *smartphones* e *tablets*. Desta forma, podem ser interpretados como gatilho para uma determinada ação no aplicativo [Endeavor 2015].

Outras soluções foram desenvolvidas utilizando conceitos parecidos, porém com algumas divergências. Dentre as soluções, uma delas é a proposta da Cisco em conjunto com a *Qualcomm*, que utiliza a tecnologia *WiFi*. Seu funcionamento é muito similar ao do GPS, que verifica o posicionamento através da correlação de sinais recebidos ao mesmo tempo de quatro ou mais satélites [Rezende e Ynoguti 2016].

O principal ponto positivo dessa abordagem é que não precisamos estar conectados a uma rede para utilizá-la como provedor de localização. As redes *wireless* estão fazendo *broadcast* de pacotes o tempo inteiro ao nosso redor, e os dispositivos móveis estão captando o tempo inteiro os sinais de todas as redes em seu alcance, mesmo que não esteja conectado em nenhuma [Galeno 2016].

3. Materiais e Métodos

Esta seção descreve os componentes de *hardware*, *software* e metodologia do projeto.

3.1. NodeMCU ESP8266

O ESP8266 é fundamental no projeto, pois possui a responsabilidade de prover uma rede *Wifi*, e ao mesmo tempo comportar-se como Cliente quando necessário. As tarefas do sistema embarcado são: abrir conexão com protocolo MQTT, receber informações da aplicação Android, e ativar "gatilhos" a partir dessas informações. O ESP8266 é um *microchip WiFi* de baixo custo com pilha TCP/IP (*Transmission Control Protocol – Internet Protocol*) completa e capacidade microcontroladora produzida pela empresa chinesa *Espressif Systems* [ESP8266 Overview 2018].

3.2. MQTT

O protocolo de rede subjacente da Internet é o TCP/IP. Desenvolvido com base na pilha TCP/IP, o MQTT (*Message Queue Telemetry Transport*) tornou-se o padrão para comunicações de IoT (*Internet of Things*) [Yuan 2017].

O padrão de troca de mensagens no MQTT é o Publicador/Subscritor. Neste padrão, quando um elemento da rede deseja receber uma determinada informação, ele a subscreve, fazendo uma requisição para um outro elemento da rede capaz de gerir as publicações e subscrições. Na rede MQTT este elemento é conhecido como *broker*, o intermediário no processo de comunicação. Elementos que desejam publicar informações o fazem também através do *broker*, enviando-lhe as informações que possuem [Barros 2015].

No projeto foi-se utilizado os serviços do *CloudMQTT*, essa plataforma online fornece ambiente para desenvolvimento de aplicações MQTT.

3.3. Aplicação Mobile

O projeto possui como um dos pontos primordiais o desenvolvimento um aplicativo *mobile* que tem como objetivos identificar a rede do ESP8266, calcular a intensidade do sinal recebido, abrir conexão MQTT, e enviar informações para o sistema embarcado. Para o desenvolvimento e implementação da aplicação *mobile* foi utilizado o *framework* Ionic. Ionic é um *framework Open Source* para desenvolvimento de aplicativos móveis multiplataforma. Para isso, possibilita a implementação do app utilizando tecnologias empregadas na construção do *Front-end* de soluções *web*: HTML5 (*HyperText Markup Language*), CSS3 (*Cascading Style Sheets*) e *Javascript* [DevMedia 2018].

3.4. Técnicas de Localização

Para que se obtenha a localização de estações em um ambiente sem fio, é necessária a existência de pontos de referência fixos e a utilização de uma técnica para inferir a posição física da estação. As diversas técnicas existentes baseiam-se na obtenção da distância entre os pontos conhecidos e a estação móvel através da medida do tempo necessário para a transmissão de dados entre eles, da potência do sinal, ou ângulo (direção) existente entre esses pontos [Bisatto e Perez 2015].

As medições RSSI (*Received Signal Strength Indication*) representam a qualidade relativa de um sinal recebido num dispositivo. O RSSI indica o nível de potência recebido após qualquer perda possível a nível de antena e cabo [NetSpot 2018]. A intensidade de um sinal *wireless* sofre interferência de vários fatores, incluindo a potência do transmissor, a sensibilidade do receptor, a atenuação do meio, a perda de

caminho (*path loss*) e a influência de outros sinais presentes no meio [Jabour e Jabour 2015].

O método de proximidade é baseado na técnica que não calcula a exata coordenada localização, sendo, portanto, uma estimativa de detecção de presença [Moreira e Farias 2017]. Relação distância é dado pela fórmula:

$$RSSI = 10n(\log(d)) - A$$

RSSI representa o valor da intensidade do sinal que chega, o **A**, o valor do RSSI para distância de 1 metro, o **n** corresponde ao *path loss*, que no caso de ambientes aberto é atribuído valor 2 e, por fim **d** representa valor da distância entre os dispositivos. Em resumo:

$$d = 10^{(RSSI - A / (10n))}$$

4. Resultados e Discussões

O projeto consiste em mensurar a distância entre um NodeMCU ESP8266 em modo *Access Point*, e qualquer celular Android com o aplicativo instalado e em uso, mensurar distância entre eles, e por fim, ativar a ação de ligar determinado LED (*Light Emitting Diode*) via MQTT. O projeto teve arquitetura semelhante com a apresentada na Figura 1.

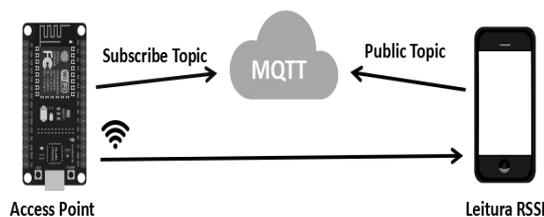


Figura 1. Arquitetura do Projeto.

Inicialmente foi necessário criar uma conta, e depois uma instância de projeto no *CloudMQTT*. Cada instância no *CloudMQTT* possui dados que são de grande importância para correta conexão com o *Broker* MQTT.

Posteriormente foi desenvolvido o código responsável por colocar o NodeMCU em modo *Access Point* e Cliente, simultaneamente, e quando a conexão *WiFi* é iniciada com sucesso é realizada conexão com o *Broker* MQTT, e a subscrição no tópico: “**nodemcu1/led**”, ao receber “**nodemcu1/led/0**” o LED permanece desligado, caso receba “**nodemcu1/led/1**” o LED acende. Na Figura 2 apresenta o esquema prático do projeto, onde verifica-se que o LED ligado no pinoD5 do microcontrolador.

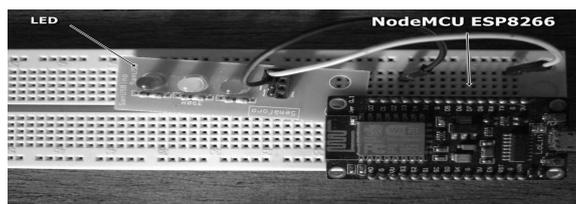


Figura 2. NodeMCU ESP8266 com LEDs.

Com o NodeMCU funcionando corretamente, foi necessário desenvolvimento do aplicativo. O processo de desenvolvimento se deu com o uso do *framework* Ionic

para *front-end*, juntamente com o *Apache Cordova* para realização das funções Nativas da aplicação. Com *Cordova* foi utilizado a plugin *Hotspot*, essa biblioteca fornece diversas funções para gerenciamento das redes no *Android*. O uso do *Hotspot* foi necessário para realizar varredura na rede, e com isso detectar o ESP8266 e obter o endereço MAC (*Media Access Control*) e o RSSI da comunicação. A varredura na rede realiza-se no intervalo de 2 segundos.

Seguidamente, se fez necessário o uso de outra biblioteca, com intuito de realizar comunicação via MQTT com o *Broker* da *CloudMQTT*, essa tarefa foi desenvolvida com ajuda da biblioteca *Ng2-MQTT*.

Os dados de RSSI e MAC, são de grande importância na lógica do projeto, pois com o MAC é garantido que de fato é ESP8266 está que sendo verificado. O RSSI é utilizado para mensurar a distância entre os dispositivos, com base no cálculo visto na seção 3. A distância de 1 metro teve em média um RSSI de -50.

Ao iniciar o aplicativo, na primeira tela (Figura 3) é realizado a varredura na rede e a conexão com MQTT, e a lógica para detecção de proximidade.



Figura 3. Tela principal do aplicativo.

5. Considerações Finais e Trabalhos Futuros

Este trabalho mostrou o desenvolvimento de aplicativo *Android* e a utilização do dispositivo *WiFi* de baixo custo ESP8266, para mensurar distância entre *smartphone* e microcontrolador, além de expor métodos, tecnologias, e técnicas para localização em ambientes internos. A aplicação mostrou-se eficaz, embora apresentando dados por vezes imprecisos mesmo com os dispositivos parados. Portanto verifica-se que carece de melhorias em decorrência direta da atenuação do sinal, causado por fatores externos.

A sequência do trabalho se dará com o uso dos métodos de localização, como o da Trilateração ou Triangulação, com uso de outros NodeMCUs, com intuito de mensurar a localização de *smartphones* de forma mais precisa.

Referências

Cassimiro, André G.A. Tsuchiya, Lucas B. (2017). "RFLocus - Sistema de Localização por meio das redes Wi-fi, direcionado a deficientes visuais". http://paginapessoal.utfpr.edu.br/gustavobborba/if66j-s71-projetos/files/EEX23-17b_RT_RFLocus.pdf. Acesso em: 16/08/2018.

- Barros, M. (2015). “MQTT Protocolos para IoT”, <https://www.embarcados.com.br/mqtt-protocolos-para-iot/>, Acesso em: 16/09/2018.
- Bellecieri, Y. e Jabour, F. C. e Jabour, E. G. (2015). “Localização Indoor Baseada na Leitura Bidirecional do RSSI”, <http://periodicos.jf.ifsudestemg.edu.br/revistabsi/article/view/25>. Acesso em: 20/09/2018.
- Bisatto, A. P. e Peres, André. (2015). “Localização de estação sem fio utilizando trilateração”, <https://www.researchgate.net/publication/228799192/>. Acesso em: 20/09/2018.
- Centro de Computação Gráfica. (2018). “Indoor location: a verdadeira beleza está no interior?”, <http://www.ccg.pt/indoor-location/>. Acesso em: 15/09/2018.
- DevMedia. (2018). “Introdução ao Ionic”, <https://www.devmedia.com.br/guia/ionic/38372>, Acesso em: 20/09/2018.
- Endeavor. (2015). “Beacon: o GPS que ajuda sua marca a localizar as melhores oportunidades”, <https://endeavor.org.br/estrategia-e-gestao/beacon/>, Acesso em: 19/09/2018.
- ESP8266 Overview. (2018). “ESP8266 Overview”, <https://www.espressif.com/en/products/hardware/esp8266ex/overview>. Acesso em: 15/09/2018.
- Ferraz, André. (2017). “Geolocalização indoor ou?”, <http://cio.com.br/opiniao/2017/03/09/geolocalizacao-indoor-ou-beacons/>. Acesso em: 20/09/2018.
- Galeno, A. (2016). “Estratégias de Localização em Ambientes Fechados”. <https://medium.com/@arturgalenom/estrat%C3%A9gias-de-localiza%C3%A7%C3%A3o-em-ambientes-fechados-192792385961>. Acesso em: 20/09/2018.
- Sena, Francisco. (2016). “GPS, origem, funcionamento e aplicações”, <https://pt.linkedin.com/pulse/gps-origem-funcionamento-e-aplicações-francisco-sena>. Acesso em: 15/09/2018.
- Moreira, M. Fabricio, e Farias, M. S. e Carvalho, P. V. (2017). “POSICIONAMENTO EM AMBIENTES INTERNOS COM DISPOSITIVOS WI-FI DE BAIXO CUSTO”, International Nuclear Atlantic Conference, <http://carpedien.ien.gov.br:8080/bitstream/ien/2333/1/ARTIGO%20INAC%2054.pdf>. Acesso em: 20/09/2018.
- NetSpot. (2018). “O que é o RSSI e qual a sua relação com uma rede Wi-Fi”. <https://www.netspotapp.com/pt/what-is-rssi-level.html>. Acesso em: 20/09/2018.
- Rezende, E. M. e Ynoguti, C. A. (2016). “Tecnologias para localização interna (indoor location)”. Acesso em: 15/09/2018.
- Yuan, M. (2017). “Conhecendo o MQTT”, <https://www.ibm.com/developerworks/br/library/iot-mqtt-why-good-for-iot/index.html>. Acesso em: 16/09/2018.