

Sistema Para Monitoramento e Detecção de Falhas Relacionadas à Disponibilidade de Redes Wi-Fi

Valsirene Sonia Pereira¹, Claudio Correa¹

¹Departamento de Ciência da Computação
Pontifícia Universidade Católica de Minas Gerais (PUC Minas)
37.714-620 – Poços de Caldas – MG – Brasil
valsirene.pereira@sga.pucminas.br, correa@pucpcaldas.br

Abstract. *The general objective of this work is to propose an architectural model to help monitor and detect faults related to the availability of Wi-Fi networks. The specific objective of this prototype is to allow the development of an application independent of the existing physical infrastructure, low cost, with the possibility of remote monitoring, to support the initiatives of control and diagnosis of unavailability and unscheduled interruptions related to Wi-Fi networks. Due to the characteristics proposed by the architecture, its implementation becomes feasible, as it proves to be a tool capable of supporting the identification of inconsistencies regarding the availability of these types of networks.*

Resumo. *O objetivo geral desse trabalho é propor um modelo de arquitetura para auxiliar o monitoramento e a detecção de falhas relacionadas à disponibilidade de redes Wi-Fi. O objetivo específico desse protótipo é permitir o desenvolvimento de uma aplicação independente da infraestrutura física existente, de baixo custo, com possibilidade de monitoramento remoto para auxiliar o suporte nas iniciativas de controle e diagnóstico de indisponibilidades e interrupções não programadas referentes às redes Wi-Fi. Devido às características propostas pela arquitetura, torna-se viável a sua implementação, pois se mostra como uma ferramenta capaz de apoiar na identificação de inconsistências referentes à disponibilidade desses tipos de redes.*

1. Introdução

O acesso à Internet é essencial na rotina de empresas e organizações, uma vez que a maioria de suas atividades depende desse recurso para serem realizadas. Por essa razão, essas instituições têm buscado soluções para acompanhar esses serviços a fim de fornecer melhor qualidade e evitar falhas operacionais.

Modelos propostos encontrados na literatura, utilizam dispositivos embarcados em sistemas auxiliares para monitorar esses recursos, como o dispositivo microcontrolador *NodeMCU Esp8266*, tendo como objetivo torná-los mais seguros, visto que sua indisponibilidade pode gerar grandes prejuízos às empresas e organizações [1] [2].

Essa necessidade de fornecer maior segurança nesse tipo de cenário justifica o desenvolvimento de sistemas para auxiliar o controle e monitoramento, buscando

otimizar custo e benefício. Já a análise dos trabalhos relacionados, incentiva a utilização de dispositivos microcontroladores como uma alternativa viável devido ao baixo custo, capacidade de processamento e baixo consumo de energia.

Neste contexto, este trabalho tem como objetivo desenvolver uma ferramenta independente da infraestrutura física existente, que auxilie o monitoramento da disponibilidade de redes sem fio (*Wi-Fi*) utilizando componentes microcontroladores. A proposta utiliza um dispositivo (placa) microcontrolador *NodeMCU Esp8266*, que dispõe do módulo *ESP8266* integrado; este módulo possui a tecnologia *Wi-Fi* integrada, a qual é utilizada para troca de informações conforme a arquitetura proposta, que será abordada na seção 2.

Espera-se que esse trabalho possa contribuir para o desenvolvimento de aplicações que possam apoiar as ações do suporte, identificando problemas referentes à disponibilidade de *ESSID*¹ (*Extended Service Set Identifier*) específicos.

As próximas seções estão organizadas da seguinte forma: na Seção 2 é apresentada a descrição do modelo proposto, a seção 3 apresenta os componentes utilizados e desenvolvimento da aplicação, na seção 4 são apresentados alguns testes realizados e, por fim, na Seção 5 é feita uma conclusão sobre este trabalho e propostas futuras.

2. Descrição do Modelo Proposto

A princípio, a aplicação desenvolvida fundamenta-se nos conceitos de cliente e servidor com a utilização da *API (Application Programming Interface) socket (TCP/IP - Transmission Control Protocol/Internet Protocol)* [3]. A comunicação entre o dispositivo (servidor) microcontrolador (*NodeMCU Esp8266*) e o cliente (navegador) é feita diretamente por meio da tecnologia *Wi-Fi* incorporada ao dispositivo, permitindo dessa forma que o monitoramento possa ser feito de forma remota (por exemplo o dispositivo pode ser posicionado em locais distantes do usuário que pode se conectar ao dispositivo por meio de um celular), considerando o alcance de comunicação da tecnologia *Wi-Fi*. A coleta das informações é feita pelo dispositivo que realiza a busca na área monitorada identificando as redes específicas. Uma vez identificadas, será analisado o seu comportamento. Ao se constatar alguma inconsistência, serão enviadas mensagens alertando o suporte. A **figura 1** representa, de forma simplificada, como é o funcionamento da aplicação e a comunicação entre o dispositivo e a interface (navegador/usuário). Uma descrição mais detalhada desse modelo é feita na seção 2.1.

¹ Conjunto de Serviços Estendidos (*ESSID*). Identifica pontos de acesso (dispositivos) à internet. “Nome” que identifica uma rede *Wi-Fi*.

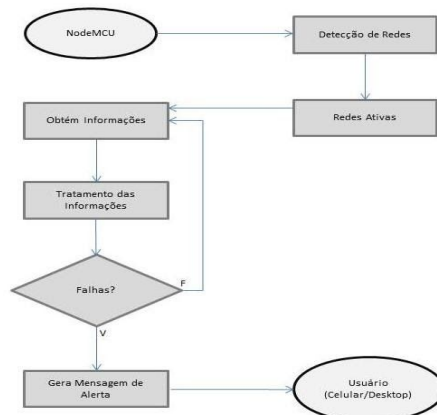


Figura 1. Fluxo de comunicação

2.1. Arquitetura

A arquitetura proposta é independente da infraestrutura física das redes observadas, o que permite que o *NodeMCU Esp8266*, seja colocado em diferentes locais e realize a varredura em tempo real. Esse dispositivo, definido como um *AP (Access Point)* e configurado como um servidor web faz a procura na área monitorada a fim de identificar os *ESSID* específicos. Se esses *ESSID* (redes *Wi-Fi*) foram encontrados durante essa varredura, são retornadas as informações correspondentes à intensidade do sinal. Considera-se que uma determinada rede (*ESSID*) está indisponível se, ao realizar a busca em um determinado local, o dispositivo não conseguiu localizá-la. Ao constatar essa indisponibilidade, é gerado um alerta que é armazenado em uma lista circular (para não sobrecarregar o usuário com inúmeros alertas desnecessários) para posterior armazenamento em um banco de dados. Quando o número de alertas dessa lista atinge um valor específico, é enviada uma mensagem de alerta ao cliente (navegador/suporte) conectado ao dispositivo. Na **figura 2** apresenta-se o modelo da arquitetura proposta.

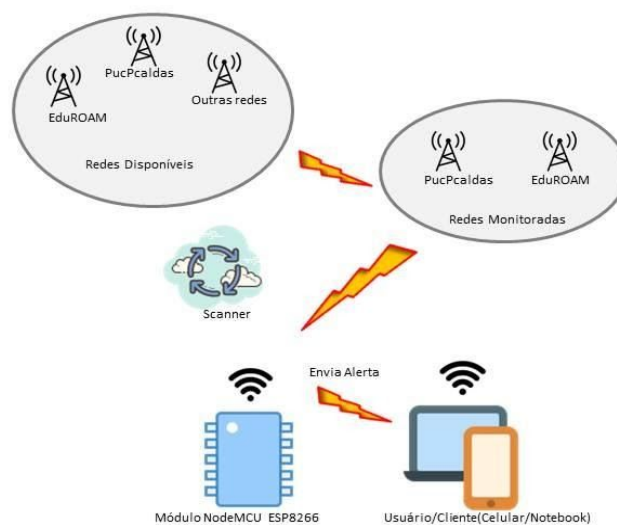


Figura 2. Protótipo da arquitetura mostrando (como Exemplo) os ESSID da universidade “PUC Minas” no campus de Poços de Caldas

Devido às características propostas pela arquitetura, pressupõe-se que o sistema de monitoramento proposto, possa auxiliar na identificação de falhas operacionais relacionadas aos *ESSID*, bem como identificar locais específicos onde o acesso a essas redes é mais difícil, considerando fatores como a intensidade do sinal. Outra característica a ser considerada é que a implementação se torna facilitada, visto que podem ser utilizadas ferramentas de plataforma aberta, como a Interface de programação “Arduino”, especificadas na seção 3.

3. Componentes Utilizados e Desenvolvimento da Aplicação

As ferramentas de desenvolvimento, dispositivo utilizado e o desenvolvimento desse trabalho estão descritos nas subseções 3.1, 3.2 e 3.3.

3.1. NodeMCU Esp8266

O módulo *Esp8266* é um sistema desenvolvido pela empresa *Espressif Systems*; é um sistema embarcado, de plataforma aberta, que possui tecnologia *Wi-Fi* (*Wi-Fi* - padrão *IEEE 802.11*[4]) integrada e baixo consumo de energia. Possui um processador Tensilica (*RISC-Tensilica L106* de 32 bits), com velocidade máxima de clock de 160MHZ e sistema operacional *RTOS* [5]. O dispositivo *NodeMCU Esp8266*, na **figura 3**, é uma placa projetada para aplicações *IoT* (*Internet of Things*) que possui o módulo *Esp8266* integrado, além de características como circuito de regulação de tensão, interface serial *USB* (*Universal Serial Bus*) para programação (que pode ser feita nas linguagens *Lua* ou *C/C++* utilizando a *IDE - Integrated Development Environment - Arduino*), tensão de operação de 4,5v à 9v, *SPI* (*Serial Peripheral Interface*), suporta até 5 conexões *TCP/IP* e custo baixo [6].

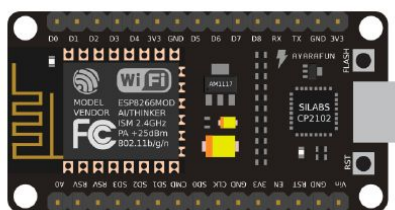


Figura 3. Módulo NodeMCU Esp8266. Esta figura foi extraída e modificada de [6]

3.2. Interface de Desenvolvimento

Como interface de desenvolvimento e programação do dispositivo, utilizou-se a *IDE Arduino* devido às vantagens oferecidas, como facilidade de instalação e configuração, e por ser uma plataforma *open source*. A interface permite a inclusão das bibliotecas necessárias à programação, que é realizada por meio de *sketchs* (arquivos com códigos) e especificamente nessa aplicação utilizando as linguagens *C/C++* [7].

3.3. Desenvolvimento da Aplicação

O dispositivo *NodeMCU Esp8266* foi configurado como um ponto de acesso, sendo o “servidor” da aplicação. É o responsável por realizar o monitoramento buscando as informações das redes *Wi-Fi*, conforme a arquitetura apresentada na **figura 2**.

No código gravado no dispositivo (*NodeMCU ESP8266*) e representado no trecho do pseudocódigo da **figura 4**, um cliente (navegador) se conecta ao dispositivo por meio do seu endereço lógico IP para ter acesso às informações. A função “*buscaRedes()*” (linha 3), dentro de um laço, encontra o número de *ESSID* disponíveis na área monitorada. Para isso, utiliza a função “*WiFi.scanNetworks()*” da biblioteca “*WiFi.h*”, que faz a varredura dessa área e atribui o número de redes encontradas à variável “*N*” (linha 4), se o valor retornado por essa função for zero, indica que nenhuma rede *Wi-Fi* foi encontrada, caso contrário, faz-se uma busca pelo identificador da rede utilizando-se a função “*WiFi.SSID()*”. Esta função obtém o *SSID* (uma string) ou identificador (nome) de cada rede e atribui a variável “*S*” (linha 11). Se “*S*” for igual à rede específica (monitorada) (linha 12), obtém-se informações como intensidade do sinal, por exemplo, e envia ao cliente (linhas 15 e 16). Se não for encontrada uma rede específica disponível (linha 17), é enviada uma mensagem (linha 19) ao cliente (navegador/suporte) informando que a rede “*S*” está indisponível (linha 20).

```
1 Enquanto um cliente está conectado ao dispositivo
2 //declarar variáveis
3 Enquanto o dispositivo busca por redes
4 N <- numero de redes encontradas
5 Se N = zero
6     Imprime "Nenhuma rede Wi-Fi disponível"
7 Se não
8     Contador <- zero
9     Encontrou <- False
10    Para contador = zero ate contador < "N" faça
11        S <- nome do SSID (contador)
12        Se S = rede monitorada
13            Rede Wi-Fi foi localizada
14            Encontrou <- True
15            Envia informações sobre S ao cliente
16            "Rede Encontrada, Intensidade do Sinal"
17    Se Encontrou = False
18        Rede Wi-Fi não foi localizada
19        Envia alerta ao cliente (navegador/suporte)
20        "Rede não encontrada.."
```

Figura 4. Pseudocódigo do algoritmo mostrando como é realizada a busca dos ESSID

4. Testes Realizados

Alguns parâmetros podem ser utilizados para medir o alcance de um sinal de *RF* (*Rádio Frequência*), como a potência do transmissor e ganho da antena. A potência do transmissor é medida em *decibel miliwatt (dBm)*, onde o valor de “*dB*” (*decibel*) é comparado a 1 *miliwatt* (por exemplo, 0 *dBm* é igual a 1 *miliwatt*). Segundo algumas regras de uso geral, um aumento de 3*dB* corresponde ao dobro de potência de transmissão, se houver uma diminuição de 3*dB*, essa potência reduz-se pela metade. As antenas funcionam como uma interface e têm a função de tratar uma quantidade específica de potência emitida pelo transmissor. O ganho da antena é representado em *dB_i*, que corresponde ao ganho da antena em relação a uma fonte isotrópica (modelo idealizado de antenas isotrópicas). Esse modelo pressupõe que o sinal seja transmitido com a mesma intensidade em todas as direções a sua volta [8] [9] [10].

Além dos fatores descritos, outro fator que pode influenciar na distância em que um sinal pode alcançar, é a sensibilidade do receptor, ou seja, o nível de potência mínimo do sinal para que seja possível realizar uma conexão. Em geral, esse fator é informado pelo fabricante. Entende-se como sensibilidade do receptor, o nível de potência mínimo (em *dBm* ou *mW*) do sinal para que esse receptor identifique um determinado sinal, por exemplo -3dBm (*decibel miliwatt*) corresponde a 0.5 mW (*miliwatt*); -70dBm representa 0.0000001 mW . O Nível mínimo de potência do receptor está relacionado com a taxa de transmissão (*mbps*). Para maior velocidade na taxa de transmissão, é preciso intensidade mais forte do sinal. Por exemplo, para uma taxa de transmissão de 54mbps , é necessário um sinal de -69dBm (Conforme dados da **tabela 1**). Ao se reduzir em 3dBm um sinal, a sua potência reduz-se pela metade [8] [9]. A **tabela 1** mostra algumas informações como referência.

Tabela 1. Níveis mínimos da potência de um sinal e taxa de transmissão do dispositivo AP 541N-2,4GHz da empresa CISCO [11]

Taxa de Transmissão	Nível Mínimo da Potência
1 mbps	-91 dBm
6 mbps	-86 dBm
11 mbps	-85 dBm
54 mbps	-69 dBm

Com base nas informações descritas acima, alguns testes utilizando-se a aplicação desenvolvida foram realizados no campus da *PUC Minas* em Poços de Caldas, observando-se uma das redes *Wi-Fi* da universidade. Porém, deve-se considerar que os testes preliminares realizados, tiveram como objetivos apenas identificar se o *ESSID* observado estava disponível em determinado local e medir a sensibilidade de recepção do sinal em todos os *AP* (*Access Point*).

Nos testes realizados no prédio 1, do campus, considera-se o seguinte cenário: a existência de vários *AP* localizados em pontos diferentes dentro do prédio, sendo que todos os *AP* representa o mesmo *ESSID*. O dispositivo (*NodeMCU Esp8266*), foi posicionado no primeiro andar (sala 228) do prédio, realizando o monitoramento por um período de trinta minutos, atualizando as informações em um intervalo de tempo de cinco minutos. As informações representadas na **tabela 2** mostram a intensidade do sinal (dos *AP*) e o maior número de *AP* localizados pelo dispositivo nesse local (sala 228).

Tabela 2. Informações sobre o ESSID monitorado, no Prédio 1

Prédio 1 - Primeiro Andar							
AP	A	B	C	D	E	F	G
Sinal (dbm)	-88	-89	-84	-63	-77	-69	-81

Analisando-se os dados da **tabela 2**, em comparação às informações descritas nos primeiros parágrafos desta seção e aos dados da **tabela 1**, nota-se que os AP **D**, **F** e **E**, possuem um sinal mais forte, possibilitando maior taxa de transmissão e consequentemente maior conectividade, porém, é importante ressaltar que deve ser considerada, a localização do dispositivo (*NodeMCU Esp8266*) dentro do prédio. Em outros locais, dentro do mesmo prédio, essas informações tiveram valores diferentes, e alguns AP não foram localizados pelo dispositivo.

A análise dos testes iniciais foi focada nas informações sobre a intensidade do sinal e localização do *ESSID*, a fim de identificar quais AP têm melhor conectividade. Nesse sentido, a aplicação desenvolvida mostrou-se útil para obtenção desses dados referentes ao *ESSID* monitorado. Propostas futuras para otimizar essa aplicação, estão descritas na seção 5.

5. Considerações Finais e Trabalhos Futuros

O trabalho realizado apresentou a proposta de um modelo de arquitetura para um sistema de monitoramento de disponibilidade de *ESSID*, demonstrando a viabilidade de sua implementação devido às características, como baixo custo dos componentes utilizados, possibilidade de monitoramento remoto e arquitetura independente da infraestrutura física existente. Com base no modelo e às características apresentadas, conclui-se que o sistema desenvolvido pode ser uma ferramenta potencialmente útil para apoiar o suporte na detecção de falhas operacionais relacionadas à disponibilidade, bem como a obtenção de informações de *ESSID*.

Como objetivo futuro, pretende-se otimizar esta aplicação para buscar informações sobre o canal de comunicação em que uma determinada rede está operando, para identificar os canais menos utilizados ou congestionados, e possibilitar que AP's possam ser reconfigurados ou reposicionados, melhorando a conectividade dessa rede. Outro objetivo é melhorar a interface (lado cliente) permitindo maior interatividade com o usuário.

Referências

- [1] Saha, Saraswati; Majumdar, Anupam, "Data Center Temperature monitoring with ESP8266 based Wireless sensor Network and cloud based dashboard with real time alert system", (2017), Devices for Integrated Circuit, ieeexplore.ieee.org.
- [2] Santos R.H.S.; Modelo de Gestão de Falhas no gerenciamento de infraestrutura de data center (2014), repositorio.unesp.br.

- [3] Kurose, James F.; Redes de Computadores e a internet - uma abordagem topo-down-6-edição (2014).
- [4] <http://www.ieee802.org/11/>, (acessado em 15/04/19).
- [5] <http://espressif.com/>, (acessado em 15/04/2019).
- [6] Veloso, A.F., Sousa, B.A., Braz, A.R., Rabelo, R.A.L., Brit, E.M.; Prototipação com NodeMCU para Internet das Coisas em Smart Cities: Uma pesquisa - III Escola Regional de Informática do Piauí. Livro Anais - Artigos e Minicursos, v. 1, n. 1, p. 105-110, (2017).
- [7] <http://www.arduino.cc/en/>, (acessado em 15/04/2019).
- [8] https://www.cisco.com/c/pt_br/support/docs/wireless-mobility/wireless-lan-wlan/23231-powervalues-23231.html, (acessado em 03-11-2019).
- [9] Redes wireless atualizado-Calculando a potência de transmissão e de recepção <https://www.hardware.com.br/tutoriais/calculando-potencia-wireless/>, (acessado em 02-11-2019).
- [10] Alencar, Marcelo Sampaio de. Ondas Eletromagnéticas e teoria de antenas. São Paulo, Erica, (2010).
- [11] Cisco AP 541N Wireless Access Point Data Sheet-https://www.cisco.com/c/en/us/products/collateral/wireless/ap541n-wireless-access-point/data_sheet_c78-566239.html, (acessado em 10-11-2019).