MONITORAMENTO DE ATIVOS DE REDE COM USO DE GSM E SMS SOBRE PLATAFORMA ARDUINO

KOMIDO, Daniel Tsuneo Moreira[1]; REIS, Douglas Henrique[1]; GALLO, Wesley Natanael[2].
(1) Acadêmicos do Curso de Ciência da Computação, UNIFENAS, Alfenas.
(2) Professor do Curso de Ciência da Computação, UNIFENAS, Alfenas.

Resumo - ***Com o crescimento dos mais diversos dispositivos que agora se conectam à internet surgiu o conceito de "Internet das Coisas", que exigiu o desenvolvimento de uma nova forma de gerenciamento desses dispositivos ("coisas"). Neste projeto vamos discorrer sobre os métodos e recursos computacionais utilizados na realização de testes e simulações com o objetivo de implementar um sistema capaz de monitorar as coisas conectadas à rede. Para tanto, foi utilizada a plataforma Arduino, juntamente com as interfaces de comunicação, Shields Ethernet e GSM que, em ambiente simulado, gerenciam máquinas virtuais conectadas à rede criadas pelo RouterOS através do MetaRouter. Essas máquinas virtuais foram submetidas a interrupções propositais a fim de simular as falhas que serão notificadas ao administrador por meio de SMS (Short Message Service). A partir desta etapa mais informações podem ser acessadas via web, sobre quaisquer dispositivos conectados à rede, liberando o administrador de sua presença física junto aos dispositivos monitorados.***

Palavras-Chaves: Arduino; Internet das Coisas; Microcontrolador; Monitoramento de redes.

Palavras-Chaves: Arduino; Internet das Coisas; Microcontrolador.

Abstract - With the growth of the most diverse devices that now connect to the internet came the concept of "Internet of Things", which required the development of a new way of managing these devices. In this project, we will discuss the methods and resources used to perform tests and simulations, in order to implement a system capable of monitoring things connected to the network. To do so, the Arduino platform is used along with the communication interfaces, Shields Ethernet and GSM which, in a simulated environment, manage virtual machines connected to the network created by RouterOS through MetaRouter. These virtual machines were subjected to purposive interruptions in order to simulate the failures that will be notified to the administrator through SMS (Short Message Service). From this stage, information can be accessed, via the web, more information about any devices connected to the network, freeing the administrator of their physical presence next to the monitored devices.

Keywords: Arduino; Internet of things; Microcomputer; Network Monitoring.

D. T. M. Komido, UniversidadeJosé do RosárioVellano (UNIFENAS), Alfenas-MG, Brasil, tsuneokomido@gmail.com
D. H. Reis, UniversidadeJosé do RosárioVellano (UNIFENAS), Alfenas-MG, Brasil, douglas0013@uol.com.br
W. N. Gallo, Universidade José do RosárioVellano (UNIFENAS), Alfenas-MG, Brasil, wesley.gallo@unifenas.br

I. INTRODUÇÃO

Atualmente até pequenas coisas estão conectados(as) à internet, a ponto de ter sido necessário criar um termo para definir esse fenômeno IoT (*Internet of Things*). O crescimento real das "coisas" conectadas a pequenos computadores agrega mais valores e possibilidades de utilização pelas pessoas, com o consequente aumento de oportunidades de negócios cada vez mais rentáveis. Como exemplo, no IoT, até mesmo rebanhos estão conectados. O relatório intitulado "*Augmented Business*" da *The Economist* [3], descreve como as vacas serão monitoradas conforme ilustrado na FIG. 1. A *Sparked*, uma empresa holandesa, implantou sensores que permitem que os fazendeiros monitorem a saúde das vacas e acompanhem seus movimentos, garantindo um suprimento maior e mais saudável de carne para o consumo. Em média, cada animal gera cerca de 200 megabytes de informações por ano.

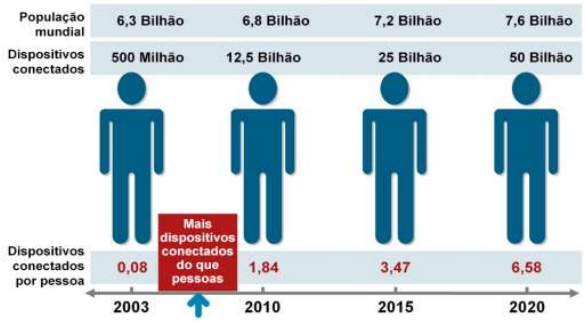
Figura 1– Etiqueta de comunicação RFID

Fonte: https://www.embarcados.com.br/introducao-a-tecnologia-de-identificacao-rfid/

Do ponto de vista acadêmico, torna-se interessante desenvolver um projeto para auxiliar de forma eficaz as atividades de gestão e monitoramento da rede, utilizando testes e notificações de status e falhas de todos os ativos encontrados.

Como ilustrado na FIG. 2, com o atual crescimento da IoT, percebe-se que o número de dispositivos conectados à internet poderá ultrapassar o número de pessoas, conforme uma projeção feita para o ano de 2020.

Figura 2 - Crescimento da internet das coisas



Fonte: http://www.cisco.com/c/dam/global/pt_br/assets/executives/pdf/internet_of_things_iot_ibsg_0411final.pdf, 2016

Com a realidade do aumento dos dispositivos conectados à rede vem a necessidade de se monitorar estes dispositivos. O mercado já oferece sistemas que fazem o monitoramento de dispositivos conectados na rede por meio da internet, porém este projeto possui o diferencial de estar diretamente conectado à rede que se pretende monitorar, sendo assim não necessita de conexão à internet para a realização do monitoramento. Além disso, pode realizar o monitoramento.

Desenvolver um sistema que seja capaz de monitorar qualquer dispositivo que possua IP conectado à rede, através de softwares e hardwares distintos. Utilizando do Arduino e dos Shields Ethernet e GSM (*Global System Mobile*), identificar falhas e notificar o administrador sobre o status do dispositivo, através de SMS (*Short Message Service*).

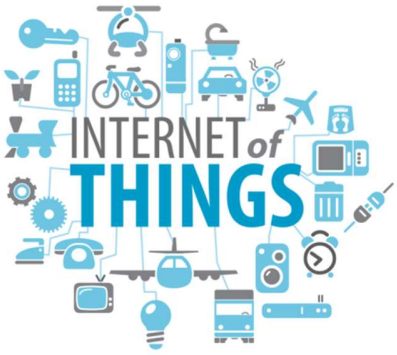
II. REFERENCIAL TEÓRICO

A - Internet das coisas - IoT (*Internet of Things*)

O termo IoT, apareceu pela primeira vez no linguajar tecnológico quando o Auto-ID Center lançou a sua versão inicial da rede EPC (*Eletronic Product Code*) para identificar e traçar o fluxo de mercadorias em cadeias de abastecimento. (Isso aconteceu em Chicago (Estados Unidos), em setembro de 2003, durante o EPC *Symposium* 2003). Já a primeira menção a "Internet das coisas" surge, em papel *Auto-ID Center*, na obra *Electronic Product Code* de autoria de David Brock em 2001.

A Internet das Coisas conforme ilustrado pela FIG. 3 é um conceito em que o mundo virtual da tecnologia da informação integra-se perfeitamente ao mundo real das coisas por meio de computadores e dispositivos de rede no mundo dos negócios e nos cenários cotidianos. A Internet das coisas é sim, uma ferramenta para gerenciar negócios de forma eficiente, mas é também e principalmente um recurso poderoso para facilitar a vida das pessoas. *An architectural approach to wards the future internet of things*, [3].

Figura 3 - Internet of Things.



Fonte: http:likedin.com/topic/internet-of-things, 2016.

B - Sistema GSM

O sistema GSM (*Global System Mobile*) é uma tecnologia móvel, sendo que este é o padrão mais comumente utilizado para celulares em todo o mundo.Telefones GSM são usados em mais de 200 países. O GSM se destaca por sua diferença com seus antecessores sendo que seus canais de comunicação são digitais, visto como um sistema de celulares de segunda geração. O sistema GSM foi utilizado considerando sua cobertura, viabilidade, baixo custo do serviço de troca de SMS e por ser um padrão que partilha elementos comuns com outras tecnologias utilizadas.

C - SMS (*Short Message Service*)

O SMS é um serviço de envio de mensagens simples muito utilizado para comunicações curtas, que se demonstra eficiente por apresentar funcionalidades de fácil incorporação e vantagens como o baixo custo das tecnologias utilizadas, bem como para o usuário e para implantação do serviço, [4].

D - Plataforma Arduino

Conforme descrito em [1], afirma que o Arduino é uma placa fabricada para resolver pequenos problemas do dia a dia. Ela teve, no início, um cunho educacional, mas devido ao seu potencial, que permitia várias implementações e utilidades, vem sendo utilizada atualmente para desempenhar diversas funções.

E - *Shields*

Conforme [2], o propósito dos *Shields* são placas que podem ser conectadas sobre as placas Arduino estendendo suas funcionalidades como: facilidade na montagem e baixo custo de produção.

Shield Ethernet

O *Shield Ethernet* permite que uma placa Arduino conecte-se à *internet*. Ele baseia-se na Wiznet W5100 chip de *Ethernet* (dataSheet). O Wiznet W5100 fornece uma rede (IP) capaz de utilizar dos protocolos TCP (*Transmission Control Protocol*) e UDP (*User Datagram Protocol*), [2].

Arduino GSM/GPRS *Shield*

O *Shield* GSM permite que uma placa Arduino possa se conectar a internet, fazer e receber chamadas de voz e enviar / receber mensagens SMS. O *Shield* utiliza um modem de rádio M10 por Quectel. A biblioteca de GSM tem um grande número de métodos para a comunicação com o *Shield*.

F - O Protocolo TCP/IP

Conforme [6], o TCP/IP é um modelo que se baseia em 4 camadas. Todos os protocolos que pertencem ao conjunto de protocolos TCP/IP estão localizados nas camadas superiores.

É o protocolo mais completo e aceito atualmente, todos os sistemas operacionais modernos oferecem suporte, pode ser utilizá-lo em sensores e atuadores da IoT, proporcionando muitas aplicações, e possibilitando a conexão D2D (Device to Device).

G - Protocolo ICMP

O protocolo será de suma importância para o projeto, pois através dele que será verificado se um dispositivo se encontra ativo na rede.

Segundo [6], o protocolo de mensagens de controle de internet, ICMP (*Internet Control Message Protocol*), é um padrão TCP/IP necessário definido na RFC 792. Com o ICMP, os hosts e roteadores que usam a comunicação IP podem relatar erros e trocar informações de status e controle limitado.

Geralmente, as mensagens ICMP são enviadas automaticamente nas seguintes.

1. Quando um Datagrama IP não consegue chegar ao seu destino.
2. Quando um roteador IP (gateway) não consegue encaminhar datagramas na atual taxa de transmissão.
3. Quando um roteador IP redireciona o host remetente para usar uma rota melhor para o destino.

As mensagens ICMP são mais comumente descritas como, solicitação de eco (determina-se um nó IP (Um host ou roteador) está disponível na rede), resposta de eco (responde a uma solicitação de eco ICMP), destino inacessível (informar ao host que um datagrama não pode ser entregue), retardamento de origem (informar ao host para diminuir a taxa de envio de datagramas devido a congestionamento), redirecionamento (informar ao Host uma rota preferencial) e tempo excedido (Indica que o tempo de vida (TTL) de um datagrama IP já expirou).

Conforme [7], “Ping é um dos métodos mais comuns para testar a acessibilidade aos dispositivos de rede. Ele utiliza uma série de mensagens de eco de ICMP para determinar: se o host remoto está ativo ou inativo”.

H - RouterBoard

O RouterBoard é um roteador integrado no Sistema Operacional RouterOS, podendo atender vários tipos de ambientes, desde um ponto de acesso em um hotel até como um roteador de borda em um datacenter.

O MetaRouter é um novo recurso para virtualização de máquinas do RouterOS que possibilita criar e executar máquinas virtuais com menos de 16MB por máquina, atualmente pode ser criado até 8 máquinas virtuais, [5].

III. METODOLOGIA

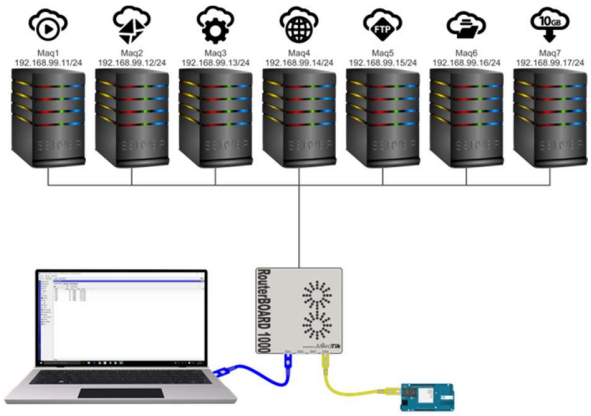
Neste tópico é apresentado, os requisitos de hardware e software necessários para a concepção do projeto.

A - Implementação do Sistema

O projeto utiliza de várias tecnologias como o Arduino Atmega2560, *Shield Ethernet* e *Shield* GSM juntamente com o RouterBOARD. Em conjunto com uma máquina que possui sistema operacional Windows 10, conforme ilustrado na FIG. 4.

Na RouterBoard através do RouterOS foram criadas 7 máquinas virtuais, cada uma com suas respectivas placas de rede (*Ethernet*), sendo que em cada uma foi atribuído um endereço IP entre (192.168.99.11/24 ~ 192.168.99.17/24), todas elas conectadas através de uma *bridge* que faz comunicação direta com o Arduino, e através de uma máquina física que faz o acesso as máquinas virtuais afim de realizar os testes do projeto.

Figura 4 - Visão geral do cenário de testes.



O Arduino Mega realiza as requisições e obtém respostas dos *Shields* (*Ethernet* e GSM SIM900) através de suas portas digitais. Ele é alimentado por uma fonte de 12V4A que por sua vez fornece energia aos respectivos *Shields* e através do cabo USB (*Universal Serial Bus*) é realizada a comunicação com a máquina responsável por sua programação e *upload* do código.

O *Shield Ethernet* é conectado diretamente sobre o Arduino Mega onde assume as funções de todos os pinos, inclusive RX e TX. Sua conexão com o RouterBoard é

estabelecida utilizando a porta RJ45, também foi utilizada a biblioteca *Ethernet* e configurado um endereço físico (MAC) e um IP (192.168.99.2/24). Através da biblioteca ICMP_PING é possível o uso do comando "*ping*", presente no protocolo ICMP, nos dispositivos virtualizados no RouterBOARD.

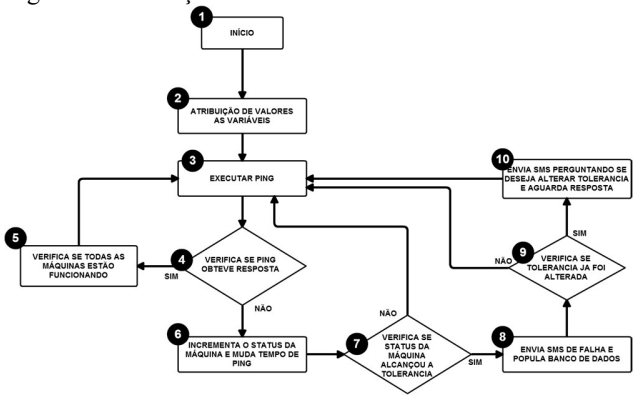
O *Shield* GSM SIM900 é conectado sobre o *Shield Ethernet*, sendo necessária a alteração dos pinos RX e TX, por serem utilizados pelo *Shield Ethernet*. Na biblioteca GSM.cpp foi alterado o pino RX 2 para o pino 19 e TX 3 para o pino 18, para evitar conflitos de leitura e escrita entre os *Shields*.

B - Descrição do Monitoramento

Como ilustrado no fluxograma da FIG. 5, o Arduino Mega utilizando o *Shield Ethernet*, realiza o monitoramento executando o comando "*ping*", que verifica o status dos dispositivos virtualizados no RouterBOARD, endereçados na faixa de IP entre (192.168.99.11/24 ~ 192.168.99.17). O intervalo entre os *pings* é de 1 minuto inicialmente. Em caso de falha o intervalo será alterado para 20 milissegundos.

Para evitar um falso positivo é estipulada uma tolerância de 3000 falhas, que a 20 milissegundos será alcançada em aproximadamente 1 minuto. Ao ser atingida significa que o dispositivo parou de responder, sendo assim, o shield GSM SIM900 notificará o administrador da rede por meio de um SMS (Short Message Service). O administrador pode responder ao Arduino Mega por meio de um SMS, estipulando um novo número de tolerância. Se após (x) ciclos determinados pelo administrador a máquina continuar parada ele estará ciente da falha e tomará as medidas necessárias.

Figura 5 - Descrição do monitoramento

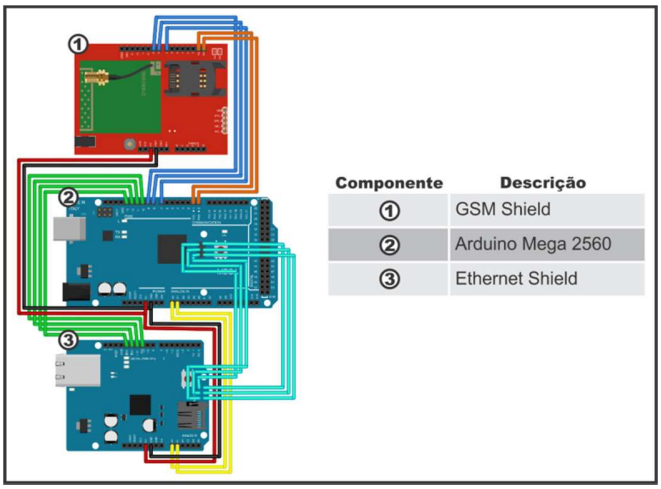


As etapas numeradas no fluxograma estão descritas na sequência.

- Etapa (1) – Momento em que o sistema faz o boot. Nesta etapa ele inicia as interfaces de comunicação Ethernet e GSM, segue para a etapa (2).
- Etapa (2) – São iniciadas as variáveis do sistema como: T = 60000 milissegundos que corresponde a 1 minuto (tempo entre cada ciclo de ping), vetor STATUS = [0,0,0,0,0,0,0,0] que estando em 0 significa que os dispositivos estão respondendo ao ping, QTM = 7, informando que serão monitoradas 7 máquinas, TOLERANCIA = 3000 estabelecida para que somente após 3000 falhas de ping será enviado SMS ao administrador, segue para a etapa (3).
- Etapa (3) – Será executado o ping em cada dispositivo e ao terminar o ciclo das 7 máquinas terá uma espera de 1 minuto para recomeçar, DELAY (T), segue para a etapa (4).
- Etapa (4) – É realizado o teste se o ping obteve resposta. Se verdadeiro é atribuído 0 ao STATUS da máquina e ele segue para a etapa (5), se falso segue para a etapa (6).
- Etapa (5) – Verifica se a soma do vetor STATUS das posições de 1 à 7 é igual a 0 e se o ping que acabou de ser executado foi no último dispositivo monitorado, se verdadeiro significa que todos estão funcionando, assim as variáveis TOLERANCIA e T recebem seus valores iniciais, seguindo para a etapa (3).
- Etapa (6) – Nesse ponto o STATUS da máquina é somadode1 e T = 20 milissegundos que será o novo tempo de espera entre ping entre os ciclos, segue para a etapa (7).
- Etapa (7) – É verificado se o STATUS da máquina é igual à TOLERANCIA se verdadeiro segue para etapa (8), se falso segue para a etapa (3).
- Etapa (8) – O sistema envia o SMS para o administrador com o IP da máquina informando que ela está parada, e envia as informações para o banco de dados. O STATUS da máquina recebe o valor 1, segue para a etapa (9).
- Etapa (9) – Verifica se a TOLERANCIA ainda está com seu valor inicial, se verdadeiro segue para etapa (10), se falso segue para a etapa (3).
- Etapa (10) – É enviado ao administrador um SMS perguntando se ele deseja alterar a TOLERANCIA de falhas para que ele receba a notificação de dispositivo parado, o sistema aguarda 30 segundos, DELAY (30000), lê se recebeu um SMS com novo valor para ser atribuído a TOLERANCIA, segue para a etapa (3).

C - Esquema de ligações do Arduino

Figura 6– Esquemático



A FIG. 6 demonstra todas as ligações realizadas entre o Arduino Mega e os *Shields Ethernet* e GSM. No projeto as placas estão conectadas diretamente uma sobre a outra, na sequência Arduino Mega, *Shield Ethernet* e *Shield* GSM respectivamente de forma simples e com espaço reduzido.

IV. RESULTADOS

Os resultados obtidos com a implementação do sistema desenvolvido podem ser observados através de uma aplicação *web* onde o administrador pode obter as informações dos estados de cada máquina. O sistema apresentada a taxa de falhas do envio de SMS (*Short Message Service*), além de abordar alguns problemas encontrados ao decorrer da concepção do sistema.

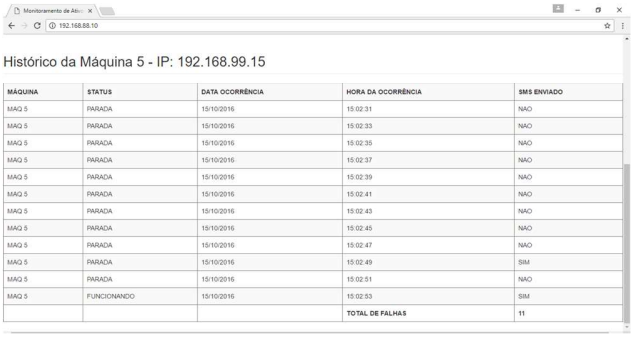
A - Simulação de Falhas

Utilizando o Winbox, (programa com interface gráfica usado para configurar e conectar os dispositivos virtualizados no RouterOS), em uma máquina conectada ao RouterBoard é possível realizar os testes de eficácia do sistema com envio de SMS. Desta forma é possível visualizar e interagir com as máquinas virtualizadas. Aleatoriamente foi escolhida uma entre as máquinas virtuais denominadas de maq1 à maq7, que foi desligada para simular uma falha na rede. Através de uma aplicação *web* é possível acompanhar o *status* de todas as máquinas e no caso de falha em alguma delas visualizar o seu histórico a cada iteração sem sucesso, sendo possível colher as seguintes informações conforme FIGs. 7 e 8 respectivamente:

- Estado da máquina: Funcionando ou Parada;
- Envio do SMS: Sim ou Não;
- Número de falhas;
- Horário da ocorrência: hora de parada ou retorno do funcionamento da máquina.

Figura 7 - Status das Máquinas

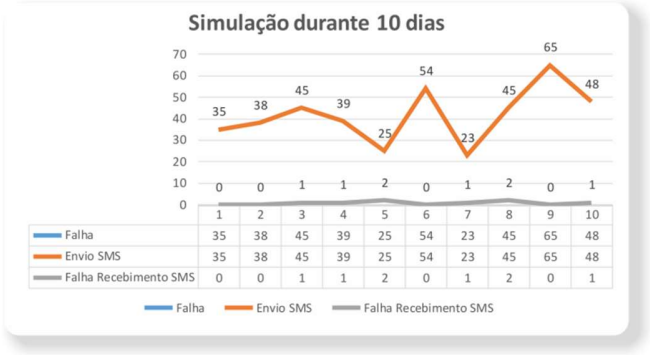
Figura 8 - Histórico da máquina



B - Simulações realizadas no Sistema

Em 10 dias de testes do sistema foram provocadas 417 falhas (em um ambiente simulado) onde em 100% das vezes o SMS foi enviado, em apenas 2% dos casos o administrador não recebeu o SMS por estar fora de área de cobertura da operadora.

Figura 9 - Simulação de erros



O sistema foi configurado para que as mensagens fossem encaminhadas para os desenvolvedores do projeto. Como se observa na FIG. 9, a linha de falha acompanha igualmente a linha de envio de SMS, onde no eixo (Y) é o número de falhas e SMS enviado, e no eixo (X) é o tempo (dias), o número de falhas é igual ao número de SMS enviados comprovando a eficácia do sistema, porém como se observa a falha do recebimento de SMS é baixo em relação ao envio do SMS.

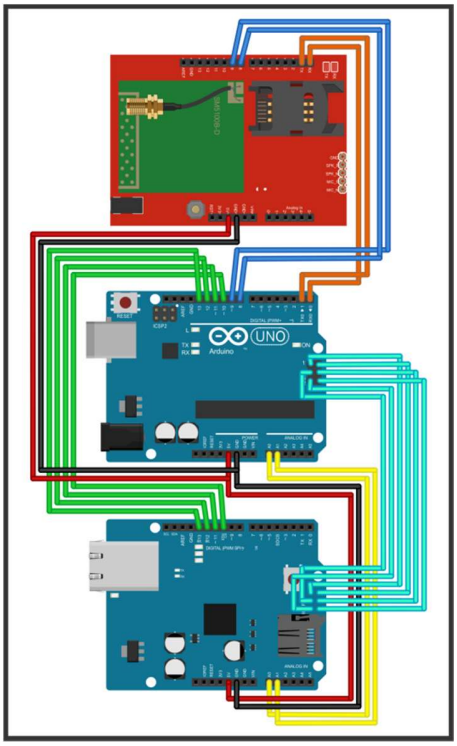
Ao decorrer do tempo foram provocadas falhas pelos desenvolvedores nos celulares como desligamento ou acionamento do modo avião (as conexões do celular são desativadas), falhas possíveis de acontecer ocasionalmente em um cenário real.

C - Dificuldades encontradas no desenvolvimento do projeto

Os desenvolvedores com pouca experiência ao iniciar o desenvolvimento do projeto, adquiriram o Arduino uno, este modelo apresentou uma maior dificuldade no desenvolvimento do projeto, pois não foi possível empilhar os *Shields Ethernet* e GSM sobre ele, devido a problemas de comunicação dos pinos RX e TX, recepção e transmissão de dados respectivamente. Para que se comunicassem teria que ser utilizado um número maior de *jumpers*, que dificultaria o manuseio das placas tanto para transporte quanto para uma nova montagem como ilustrado pela FIG. 10.

A escolha recaiu então sobre o Arduino Mega, que possui diferenças técnicas do Arduino uno, como uma maior quantidade de portas RX e TX, e que fornece a possibilidade de empilhar todas as placas simplificando a comunicação entre elas além de diminuir o espaço físico a ser ocupado.

Figura 10 - Esquemático com Arduino Uno



V. CONCLUSÃO

Com o rápido crescimento da IoT em todo o mundo, torna-se vigente a necessidade de uma ferramenta que permita ao administrador de redes realizar um monitoramento rápido e eficiente para garantir melhor controle do sistema e rápida detecção para consequente correção de possíveis falhas.

Para tentar suprir essa lacuna, o presente trabalho foi desenvolvido em um ambiente simulado com foco na criação de um sistema capaz de realizar monitoramento e notificação (de falhas) ocorridos nos dispositivos da rede, com o objetivo de evitar problemas mais graves.

O uso da plataforma Arduino com as interfaces de comunicação *Ethernet* e GSM, mostrou-se eficaz na notificação de falhas, garantindo ao administrador informações precisas e com baixa latência dos estados das máquinas como "Funcionando e Parado". Esses alertas serão enviados ao administrador via SMS (*Short Message Service*), dispensando assim a tarefa de ficar acessando seus servidores periodicamente com o fim de monitoramento. Além disso, o sistema possui uma aplicação *web* que fornece uma série de outras informações, como por exemplo: detecção do dispositivo afetado, data e hora da falha, status atual e o registro da notificação.

O grande diferencial desse monitoramento, como demonstrado neste estudo, é que esta solução foi implementada localmente utilizando a plataforma Arduino e não depende de conexão com a Internet, liberando o administrador de sua presença junto à rede, uma vez que o sistema provou ser altamente eficaz na notificação das falhas ao administrador.

Em 100% dos testes realizados durante este trabalho, ficou demonstrado que o sistema foi capaz de enviar as notificações mesmo quando o administrador não se encontrava em área de cobertura do GSM, podendo recebê-las posteriormente ao se localizar em área onde haja sinal.

Este trabalho poderá, no futuro, instigar projetos, novos ou complementares, que necessitem da solução apresentada, podendo atender futuras necessidades cada vez mais complexas desse nicho de mercado da internet das coisas.

REFERÊNCIAS

[1] ARDUINO. Introduction, 2015. Disponível em: <https://www.arduino.cc/en/Guide/Introduction/>.

[2] ARDUINO. Arduino Ethernet Shield, 2016. Disponível em: <https://www.arduino.cc/en/Main/ArduinoEthernetShield> .

[3] EVANS, DAVE. 2016. A Internet das Coisas Como a próxima evolução da internet está mudando tudo. Disponível em: <http://www.cisco.com/c/dam/global/pt_br/assets/executives/pdf/internet_of_things_iot_ibsg_0411final.pdf>.

[4] JAMES F., KUROSE; W. ROSS, KEITH. Redes de Computadores e a Internet Uma abordagem Top-Down. Editora Pearson, 3 ed. 2005.

[5] MIKROTIK. 2016. Manual:MetaRouter. Disponível em: <http://wiki.mikrotik.com/wiki/Manual:Metarouter>.

[6] TECHNET. 2015. Protocolo de mensagens de controle da internet (ICMP). Disponível em: <https://technet.microsoft.com/pt-br/library/cc758065(v=ws.10).aspx>.

[7] TELECO. 2015 Rede GSM I: Telefonia Celular. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialalambcel1/pagina_3.asp>.