

## **Matheus Guadagnino Ribeiro**

### **NODE-RED: ESTUDO EXPLORATÓRIO E NODES CUSTOMIZADOS** PARA INTERNET OF THINGS

**Assis** 2016

Av. Getúlio Vargas, 1200 – Vila Nova Santana – Assis – SP – 19807-634 Fone/Fax: (0XX18) 3302-1055 homepage: www.fema.edu.br



## Matheus Guadagnino Ribeiro

### NODE-RED: ESTUDO EXPLORATÓRIO E NODES CUSTOMIZADOS PARA INTERNET OF THINGS

Projeto de pesquisa apresentado ao curso de Bacharel em Ciências da Computação do Instituto Municipal de Ensino Superior de Assis - IMESA e a Fundação Educacional do Município de Assis -FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando (a): Matheus Guadagnino Ribeiro Orientador (a): Prof. MSc. Guilherme de Cleva **Farto** 

**Assis** 2016



### FICHA CATALOGRÁFICA

RIBEIRO, Matheus Guadagnino.

NODE-RED: ESTUDO EXPLORATÓRIO E NODES CUSTOMIZADOS PARA INTERNET OF THINGS / Matheus Guadagnino Ribeiro. Fundação Educacional do Município de Assis -FEMA - Assis, 2016.

Número de páginas. 53

Orientador: Prof. MSc. Guilherme de Cleva Farto

Trabalho de Conclusão de Curso - Instituto Municipal de Ensino Superior de Assis -**IMESA** 

1. Node-Red. 2. Arduino. 3. Internet das Coisas.

CDD: 001.6

Biblioteca da FEMA



### NODE-RED: ESTUDO EXPLORATÓRIO E NODES CUSTOMIZADOS PARA INTERNET OF THINGS

#### **MATHEUS GUADAGNINO RIBEIRO**

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador: Prof. Msc. Guilherme de Cleva Farto

Examinador: Prof. Dr. Luiz Carlos Begosso

### **Assis** 2016



#### **RESUMO**

Atualmente Internet é umas das coisas mais usadas no mundo como em comunicação entretenimento e trabalho, visando isso porque não usar a internet para facilitar a vida no cotidiano. Aplicar a Internet nos objetos pessoais e torná-los mais dinâmicos. Essa é a ideia da Internet of Things (Internet das Coisas), colocar os objetos online para ter uma maior interação com o usuário.

A pesquisa trata-se de uma base sobre loT e da plataforma utilizada para desenvolver, chamado Node-Red.

A plataforma é simples, intuitiva e nesse projeto irá utilizar o Arduino e os Nodes Customizados para uma atividade.

Palavras-chave: Node-Red; Arduino; Internet das Coisas.



### **ABSTRACT**

Currently the Internet is one of the things most used in the world as in entertainment, communication and work. So why not use the internet to make life easier in daily life. Apply the Internet on personal objects and make them more dynamic. That's the idea of the Internet of Things (IoT), put online objects to have more interaction with the user.

The research it is a base for IoT and platform used to develop called Node-Red. The platform is simple, intuitive, and this project will use the Arduino and Custom Nodes for an activity.

**Keywords:** Node-Red; Arduino; Internet of Things.



# LISTA DE ILUSTRAÇÕES

Figura 1: Aplicações Inteligentes nas Cidades	5
Figura 2: Veículos com IoT	7
Figura 3: M2M Conexões por Região	7
Figura 4: Internet das coisas entre 2008 e 2009	8
Figura 5: Intel Edison e Intel Galileo	12
Figura 6: Modelo de Nodes (Retirado do Site oficial, Nov. 2015)	13
Figura 7: Página de Instalação Node-Red (Retirado do Site oficial, Mai 2016	<b>5).</b> . 14
Figura 8: Página de Download Python (Retirado do Site oficial, Mai 2016)	14
Figura 9: Modelo do Node em execução	15
Figura 10: Hello World – Node-Red	16
Figura 11: Fritzing – Exemplo do Arduino	21
Figura 12: Fritzing – Exemplo com o Arduino e Protoboard	21
Figura 13: Sensor DHT11	22
Figura 14: Internet das coisas Aplicação Nodes no Twitter	24
Figura 15: Modelo de Implementação dos Nodes	25
Figura 16: Modelo de Implementação dos Nodes	25
Figura 17: Modelo de Node Blink	30
Figura 18: Modelo de Node Sete Segmentos	35

# SUMÁRIO

1. II	NTRODUÇÃO	. 1
1.1.	OBJETIVOS	. 2
1.2.	JUSTIFICATIVAS	. 2
1.3.	MOTIVAÇÃO.	. 3
1.4.	PERSPECTIVA DE CONTRIBUIÇÃO	. 3
1.5.	METODOLOGIA DE PESQUISA	. 3
2. II	NTERNET OF THINGS (IOT)	. 4
2.1.	INTRODUÇÃO	. 4
2.2.	CONCEITOS	. 5
2.3.	ESTATÍSTICAS	. 7
2.4.	TECNOLOGIAS E PLATAFORMAS RELACIONADAS	. 9
2.4.1.	ARDUINO	10
2.4.2.	RASPBERRY PI	11
2.4.3.	INTEL EDISON E INTEL GALILEO	11
2.5.	DESAFIOS E OPORTUNIDADES	12
3. F	PLATAFORMA NODE-RED.	13
3.1.	INTRODUÇÃO	13
3.2.	INSTALAÇÃO NODE-RED	13
3.3.	FUNCIONALIDADES E RECURSOS	
3.4.	BENEFÍCIOS.	17
3.5.	PACOTE NODE-RED ARDUINO	18
3.5.1.	INTRODUÇÃO	18
3.5.2.	FUNCIONALIDADE E RECURSOS	19
3.6.	PACOTE JOHNNY-FIVE.	19
3.6.1.	INTRODUÇÃO	19
3.6.2.	FUNCIONALIDADE E RECURSOS	20
3.7.	EXEMPLOS MOTIVACIONAIS	20
3.7.1.	ARDUINO BLINK	20

3.7.2.	ARDUINO DHT11 LOGGER	22
3.7.3.	SENTIMENTAL TWEET 2 RGB	23
3.8.	IMPLEMENTAÇÃO DOS NODES	24
3.8.1.	MODELO ARQUITETURAL	24
4. F	PROPOSTA DE TRABALHO	26
4.1.	DEFINIÇÃO DO CENÁRIO EXPERIMENTAL	26
4.2.	NODES CUSTOMIZADOS	26
5. (	CONCLUSÃO	40
5.1.	TRABALHOS FUTUROS	41
REFE	RÊNCIAS	42

## 1. INTRODUÇÃO

Nos últimos anos, o crescimento da área da tecnologia da informação, tem sido muito significante. Em 1991 começou uma discussão sobre conectar objetos uns aos outros, através *TCP/IP*, visando utilizar a Internet, já que estava cada vez mais acessível. *William Nelson Joy* ou "*Bill Joy*" (Cientista da computação, co-fundador da empresa *Sun Microsystems*), teve a ideia de conectar várias redes a dispositivos. Mas foi em 1999 em que *Kevin Ashton (MIT)*, surgiu com o nome "*Internet of Things*".

Ashton; That 'Internet of Things' Thing, Jornal RFID. "O fato de ser a primeira pessoa a dizer Internet of Thing, não me dá o direito de ter todo o controle de como outros devem utilizá-lo". No mesmo artigo foi dito o seguinte.

"O problema é que as pessoas têm um tempo limitado, a atenção e precisão, isso significa que não somos muito bons em captar dados sobre as coisas do mundo real e isso é uma grande coisa. Estamos envolvidos em nossas economias, sociedade e sobrevivência e eles não são baseados em ideias ou informações, mas sim em coisas".

Kevin Ashton (2009).

A inclusão da loT abrange uma grande área de novos dispositivos e métodos a serem utilizados e nesse trabalho apresenta um desses novos métodos, conhecido como NODE. Com os anos, inúmeros projetos foram apresentados e a expectativa é de crescer ainda mais.

#### 1.1. OBJETIVOS.

Esse projeto tem como objetivo a pesquisa dessa nova tecnologia conhecida como Node-Red, que está voltada para o desenvolvimento de novas aplicações em IoT. Para entender um pouco mais sobre essa plataforma, a pesquisa abrange na criação da IoT, quando foi criada, o principal objetivo e funcionalidades. Seguindo então com a pesquisa sobre os Nodes-Red, visando à criação dos nodes, as aplicações, modo de instalação e o conhecimento da plataforma de desenvolvimento. Como o trabalho contém uma interação com o Arduino é necessário uma breve pesquisa das funcionalidades.

A pesquisa foi baseada no para o conhecimento da nova plataforma e no desenvolvimento dos Nodes Customizados e na interação com o Arduino. Alguns exemplos com o display de Sete segmentos, Leds e sensores.

#### 1.2. JUSTIFICATIVAS.

A tecnologia está crescendo cada vez mais com o dia a dia e essa evolução tem como base deixar as atividades do cotidiano mais fácil e práticas. IoT tem como objetivo colocar os objetos na Internet, podendo assim acessar e controlar de qualquer lugar.

Devido o crescimento da Internet, ter os objetos conectados e interligados vai ser cada vez mais comum e mais pessoas vão adquirir essa pratica e pelo fato de poder controlar através do próprio smartphone, ou até mesmo redes sociais, deixa essa nova tecnologia mais atrativa.

## 1.3. MOTIVAÇÃO.

Com esse projeto voltado mais para a pesquisa dessa nova tecnologia a principal motivação é aprender algo novo e que tem uma grande oportunidade de crescimento dentro da tecnologia, como o conceito de IoT e até mesmo os Nodes são tecnologia novas no mercado e existem poucos materiais disponíveis.

A pesquisa vai proporcionar um conhecimento para poder desenvolver os Nodes e entender de fato como funciona e a sua comunicação com aparelhos periféricos e também a aplicação em IoT.

O fator de existir poucos Nodes customizados disponíveis é interessante, pois abre uma grande oportunidade para a criação desses novos Nodes.

## 1.4. PERSPECTIVA DE CONTRIBUIÇÃO.

A pesquisa vai proporcionar um bom conhecimento sobre o assunto e gerar novos materiais para consulta e pesquisas futuras.

Poder realizar novos trabalhos seguindo esse campo de aplicação e aprender mais sobre IoT e contribuir com o crescimento da tecnologia.

Disponibilizar novos Nodes customizados para que outros desenvolvedores possam utilizar ou até mesmo alterar e melhorar para o uso adequado.

#### 1.5. METODOLOGIA DE PESQUISA.

Esse trabalho realizado através de pesquisas em artigos científicos, livros, teses, monografia e matérias confiáveis, proporciona a realização do projeto.

A pesquisa focou na compreensão da Internet of Things (IoT), no estudo da nova plataforma de desenvolvimento conhecida como Node-Red e outros componentes como o Arduino que faz parte do trabalho.

Como resultado do estudo, a liberação dos Nodes Customizados no site e fórum, para a alteração e ajustes dos Nós.

### 2. INTERNET OF THINGS (IOT)

### 2.1. INTRODUÇÃO.

Internet of Things é um termo utilizado desde 1999, foi um trabalho realizado no MIT (Massachusetts Institute of Technology) onde um grupo estava fazendo estudos com frequências de radio em redes (RFID). Foram inclusos sete laboratórios de diferentes (escolhidos pelo Auto-ID Center) universidades em quatro continentes e foram escolhidas para projetar a arquitetura da IoT que tem como intenção interligar ok objetos do dia-a-dia com a Internet. Onde poder controlar a casa e aparelhos domésticos enquanto está no escritório ou até mesmo na rua? O Objetivo de IoT é isso, aperfeiçoar e simplificar a vida do usuário.

Essa revolução tecnológica promete ajudar o cotidiano a fazer uma simples tarefa como descongelar uma geladeira, ligar um forno ou até mesmo ajudar com cuidados de crianças. Tudo isso através de um smartphone ou tablet.

É possível dizer que a loT é a evolução da internet de como conhecemos atualmente melhorando muito a capacidade de compartilhar, analisar, distribuir e coletar informações. À medida que a população do planeta cresce, assim aumenta a necessidade de cuidar dos recursos, visando uma melhor maneira de vida, pois o desejo de uma vida mais saudável e confortável para as pessoas e familiares é o que garante a evolução da loT, pois a necessidade de monitorar os campos para ter melhores alimentos, e nas cidades para melhor agilidade pessoal, consiste em coleta, transmissão, analise e distribuição dessas informações em grande escala.

#### 2.2. CONCEITOS

A ideia de ter tudo conectado na internet é uma ideia vaga, pois a grandeza de ter tudo interligado, até mesmo os objetos que usamos diariamente e essa tecnolocia de aplicação é chamada de IoT.

Com essas novas aplicações, surgiram novos meios de implementações e formas mais simples de ultilização, não só no meio pessoal, mas também é útil nas empresas e terá um grande inpacto nos setores economicos.

É possível notar a diferença entre lot e a World Wide Web, onde a web é uma camada de aplicação que funciona sobre a Internet e sua função é a interface com o usuario, transmitindo informações por toda a Internet.

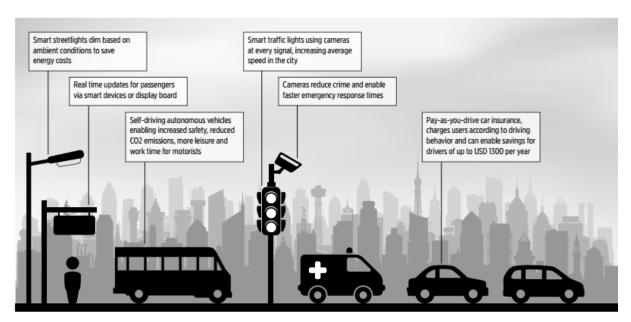


Figura 1: Aplicações Inteligentes nas Cidades Fonte Connected Living Understanding IoT (July 2014).

No caso a loT é a evolução da Internet, dando um grande salto nas aplicações melhorando o potencial de como as pessoas vivem. A loT tem como ideia tornar a Internet algo mais interativo permitindo assim sermos mais proativos. Quando mudamos o foco para conectar mais objetos do que pessoas, é possivel uma grande oportunidade para o crescimento de aplicativos e sensores na area de automação, comunicação e isso se destaca pois está mudando a vida das pessoas para melhor. Um exemplo da loT na pratica, estão sendo implantados sensores em vacas (nas orelhas) com a intenção de monitorar a saúde das vacas e acompanhar o desenvolvimento, tendo assim uma ideia da alimentação, da qualidade da carne e leite e os suprimentos necessarios. Os dados gerados por cada animal chega aproximadamente 200megabytes por ano.

Apesar do crescimento da IoT, existem algumas barreiras, desafios que podem atrasar um pouco o desenvolvimento. Um dos protocolos é o IPv6, que surgiu em 2010, pois os endereços IPv4 esgotaram, mesmo não causando um grande impacto para o publico e foi que surgiu o novo padrão. Onde pode diminuir o potencial da IoT, pois os bilhões de sensores exigirão o endereço IP e também o IPv6.

Outra barreira é a energia dos sensores, imagina ter que trocar a bateria de bilhões de sensores implantados no planeta todo, isso obviamente não é possível e, portanto para a loT atingir o potencial, seria necessários dispositivos autossustentáveis, algo que possa gerar energia a partir de meios naturais como luz, agua ou vento.

Uma descoberta importante, cientistas anunciaram uma espécie de nanogerador, um chip flexivel que usa os movimentos do corpo, como compressão dos dedos para gerar eletrecidade.

Por ultimo, temos a barreira dos padrões. Pois embora haja um grande progresso nas normas e tecnicas, ainda não é o suficiente. A IEEE é umas das responsaveis para tentar resolver esses problemas de transferencias dos pacotes IPv6, para que possa ser por tipos de redes diferentes.

### 2.3. ESTATÍSTICAS

Conforme mostrado na figura 2, verifica-se um aumento na adoção de veículos baseados em IoT.

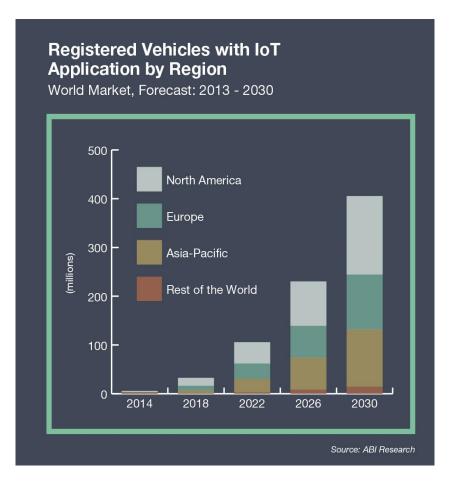


Figura 2: Veículos com IoT
Fonte Gil Press (Diretor Sênior Leadership Marketing at EMC).

Region	M2M % total connections (2013)	M2M CAGR (2010-2013)	Connections CAGR (2010-2013)
Africa	1.0%	41.3%	15.0%
Asia	2.1%	55.0%	10.4%
Europe	5.1%	28.6%	2.4%
Latin America	2.1%	43.7%	7.8%
Northern America	9.3%	22.5%	3.6%
Oceania	5.1%	25.8%	5.5%
Global	2.8%	37.6%	8.8%

Figura 3: M2M Conexões por Região

Fonte Connected Living Understanding IoT (July 2014)

Janus Bryzek (conhecido como pai dos sensores e VP da Fairchild Semicondutor) menciona "Novas previsões e oportunidades para IoT GE estimaram tem o potencial para chegar de \$10 à 15\$ trilhões nos próximos 20 anos e a Cisco estima entre \$19 Trilhões é a previsão do valor econômico até 2020".

De acordo com a Cisco IBSG foi feito o levantamento entre 2008 e 2009 e mostra que o crescimento da IoT nesse período foi muito grande, podendo assim dizer que nasceu a grande grade de comunicação entre os veículos inteligentes com mais pessoas conectadas entre si.

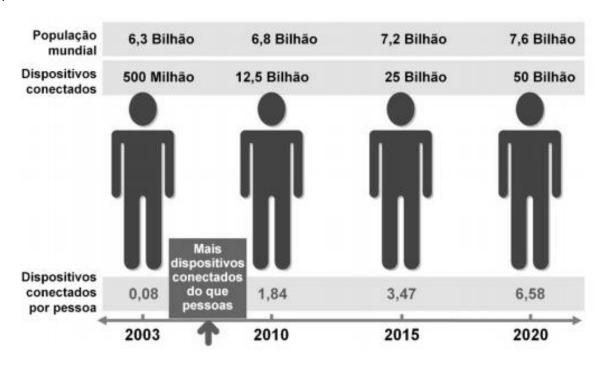


Figura 4: Internet das coisas entre 2008 e 2009 Fonte Cisco – A internet das coisas (Abril 2011)

Olhando os dados a Cisco prevê que até 2020 haverá aproximadamente 50 bilhões de dispositivos conectados para um total de 7.6 bilhões de pessoas no mundo, equivalendo quase sete dispositivos por pessoa.

#### 2.4. TECNOLOGIAS E PLATAFORMAS RELACIONADAS

Com o surgimento de nova tecnologia, vêm também novas plataformas e métodos de desenvolvimentos, facilitando assim a implementação de loT em novos objetos.

Existem plataformas especificas para cada tipo de área de desenvolvimento, caso queira desenvolver algo para a casa e os eletrodomésticos é uma plataforma, se for realizar algo para roupas e acessórios é outro tipo diferente.

Middleware é uma das plataformas utilizada para coletar dados e WSO2 é outra plataforma usada em lot.

Podemos encontrar *IoT* no nosso dia a dia como "*Nike FuelBand SE*", uma pulseira que registra a atividade física, distância etc.

*NeuroOn*: é uma máscara que monitora o sono do usuário, permitindo buscar horários e melhorar a qualidades do sono.

ToyMail. É um brinquedo capaz de enviar mensagens para as crianças.

Existem diversas utilidades e aplicações que ajuda o usuário administrando melhor o dia e atividades.

A importância do Node-Red em Internet of Things é a capacidade de desenvolver ferramentas para dispositivos físicos, através de APIs e outros recursos. E nada mais significante que poder interligar objetos através dos nodes (o princípio da IoT). É uma ferramenta visual que pode ser usado também para montar fluxos para outros serviços.

#### **2.4.1. ARDUINO**

O Arduino é uma placa de prototipagem eletrônica com o objetivo de fazer uma comunicação mais simples entre software e hardware. É possível enviar e receber informações através de sensores, sendo possível fazer codificações em sua própria IDE, mas também pode ser feito a implementação em outras linguagens e fazer a comunicação através de APis.

Existem vários tipos de Arduinos, cada um para um tipo especifico de projeto, para grande, médios e pequenos, cada um com sua capacidade de processamento.

O Arduino que será utilizado neste projeto é o Uno, mas também contém vários modelos alguns são:

- Arduino Extreme, para programação mais pesada.
- Arduino Mini é uma versão miniatura usada para montagens superficiais
- Arduino Nano, menor que o mini e ligado apenas por USB para projetos compactos.
- LilyPad, modelo simplificado para aplicações portáteis.
- Arduino NG é a nova geração.
- Arduino BT tem a interação com Bluetooth.
- Arduino Due, que usa um microcontrolador ARM.
- Arduino Leonardo, que é similar ao o UNO, porém possui 12 portas analógicas e 20 portas digitais e possui também um controlador USB, podendo assim reconhecer dispositivos.
- Arduino Mega, usado para grandes projetos, pois tem uma alta capacidade de processamento.

#### 2.4.2. RASPBERRY PI

O Raspberry, nada mais é que um microcomputador com uma alta capacidade de processamento, superior até mesmo ao Arduino.

A placa tem o tamanho de um cartão de crédito é pode ser conectados diversos dispositivos externos.

Com todo o hardware integrado em uma única placa, incentiva diversas mentes a pesquisar e descobrir como funciona.

Existem vários tipos de linguagem para se programar o Raspberry como C/C++, Java, Python entre outras. Alguns sistemas operacionais que pode ser instalado.

Atualmente encontram-se vários tipos de projetos utilizando o Raspberry, entre diversos tipos, pode-se usa-lo para segurança como, câmeras inteligentes, travas para fechaduras, pode também para criar um pequeno servidor de segurança para uma rede interna, controles gráficos e sonoros de multimídias e em robótica também.

#### 2.4.3. INTEL EDISON E INTEL GALILEO

Quando olhamos para o Intel Edison não damos muito crédito, apesar de parecer apenas um simples cartão SD transparente é algo muito mais que isso. Na verdade se trata de uma espécie de microcomputador que possui recursos de processamento com a possibilidade de suportar vários gadgets e eletrônicos.

Contendo dois núcleos Quark rodando com a capacidade de 400Mhz, garante a facilidade de integra-lo aos diversos dispositivos eletrônicos.

Já o Intel Galileo é uma placa de desenvolvimento de protótipos para o Arduino. Como é bem fácil de utilizar é indicada para projetos rápidos, mas que também tenha um bom nível de complexidade. Processador Quark SoC X1000. Assim como o Arduino que suporta vários dispositivos de E/S, compatibilidade com diferentes Hardwares.



Figura 5: Intel Edison e Intel Galileo

#### 2.5. DESAFIOS E OPORTUNIDADES

Atualmente a necessidade de ter tudo conectado é muito grande vivemos em um mundo de informações e praticidade e os Nodes em IoT é um modo de desenvolver essas praticidades que adquirimos. Existem modos de simplificar coisas do nosso cotidiano em que perdemos muito tempo. Algumas plataformas como Arduino e Raspberry entre outros, nos proporciona esse meio para desenvolver os Nodes. O maior desafio vai ser desenvolver e aplicar essa teoria da forma simples e pratica.

Aprofundar no assunto emergente que vem crescendo abundantemente para poder me capacitar em futuros projetos.

Apesar do grande crescimento ainda hoje existem poucas implementações usando a plataforma dos Nodes. Assim como não existe muita documentação referente aos Nodes customizados.

#### 3. PLATAFORMA NODE-RED.

## 3.1. INTRODUÇÃO.

Os Nodes são uma forma mais fácil e dinâmica de programar para IoT, contendo uma interface mais simples e intuitiva, pode criar inúmeros projetos com customizações. Assim podemos controlar objetos em tempo real, desde ligar uma lâmpada com uma postagem no Twitter ou controlar um drone direto do smartphone, entretanto podemos propor uma série de comandos para ser repedido em algum momento do dia ou quando um evento acontecer. Através do Node pode ser usado através de sensores "movimento, luz, temperatura entre outros".

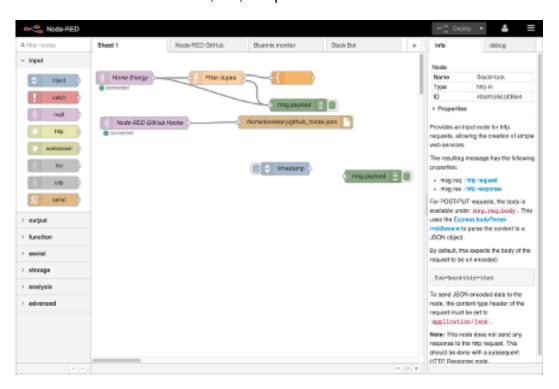


Figura 6: Modelo de Nodes (Retirado do Site oficial, Nov. 2015).

## 3.2. INSTALAÇÃO NODE-RED

Para fazer a instalação da plataforma Node-Red é simples. Acessar o site oficial do Node e ir na opção Documentation e selecionar o Istallation(no canto esquerdo). Ou acessar o seguinte link: http://nodered.org/docs/getting-started/installation, em seguida deve fazer o download do Node.js para o Sistema Operacional que está utilizando.

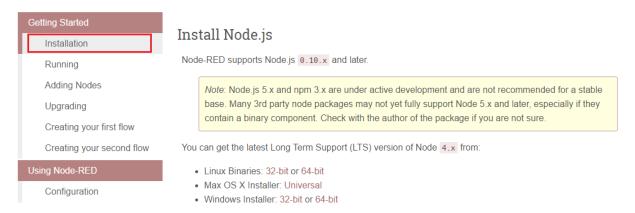


Figura 7: Página de Instalação Node-Red (Retirado do Site oficial, Mai 2016).

Depois de fazer a instalação do Node.js que irá criar um diretório no endereço C:\Program Files\nodejs. O próximo passo é fazer a instalação do Python na versão 2.7.11 https://www.python.org/downloads/.

Release version	Release date		Click for more	
Python 3.4.4	2015-12-21	Download	Release Notes	
Python 3.5.1	2015-12-07	🏖 Download	Release Notes	
Python 2.7.11	2015-12-05	🅹 Download	Release Notes	
Python 3.5.0	2015-09-13	Download	Release Notes	
Python 2.7.10	2015-05-23	Download	Release Notes	
Python 3.4.3	2015-02-25	Download	Release Notes	
Python 2.7.9	2014-12-10	🍮 Download	Release Notes	
Bushama a a	201 - 10 12	- Danieland	Delegas Massa	

Figura 8: Página de Download Python (Retirado do Site oficial, Mai 2016).

Durante a instalação do Python na terceira tela de Customização deverá deixar marcado a opção "Add python.exe to Path" (selecione "Will be installed on local hard drive" - para criar a variável de ambiente).

Feito a instalação do Python o próximo passo é executar o CMD do Windows e ir até o diretório em que o Node.js foi instalado. No diretório deve ser executado os seguintes comandos:

• "npm install -g --unsafe-perm node-red". Para instalar o Node-Red.

- "npm i -g node-red-node-arduino". (A plataforma do Arduino que será utilizada nesse projeto).
- "npm install -g arduino-firmata". Necessário para reconhecer o board na hora de execução.

Feito isso o Node-Red está instalado e reconhecendo o Arduino. Para executar o Node ainda no diretório do Node.js digite no CMD "node-red". A plataforma será carregada e irá disponibilizar um endereço que deve ser acessado através do navegador.

Figura 9: Modelo do Node em execução

### 3.2.1. Comunicação Porta Serial.

É Possível fazer a comunicação dos nodes através da porta serial, em comandos de entrada e saída é possível realizar algumas atividades.

Para isso acontecer é necessário realizar alguns comandos para que o rode-red possa reconhecer.

- "npm i node-red-node-serialport". Para instalar
- "npm install node-red-node-serialport@0.0.5". Para fazer o Update.

#### 3.3. FUNCIONALIDADES E RECURSOS

Node Red é basicamente um engine que processa eventos, fazendo a comunicação entre software e hardware. O'Leary¹ disse em uma entrevista.

"Dave e eu sabemos como fazer o código comunicar com a porta serial, podemos levar um tempo para lembrar como faz e como escreve, mas e formos usar Node Red é apenas o caso de arrastar o node e passar os detalhes da porta serial e está pronto, isto no proporciona uma grande agilidade e rapidez nos protótipos".

Os nodes não retira completamente a necessidade de escrever códigos, mas reduzir o modo de fazer nos proporciona maior tempo nos processos de criação em vez ficar mais tempo codificando.

O processo de codificação é muito mais simples e ágil. Como vemos na imagem a seguir.



Figura 10: Hello World - Node-Red

\_

<sup>&</sup>lt;sup>1</sup> Nick O'Leary e Dave Conway-Jones especialista em tecnologia emergente na IBM.

Quando aprendemos algo de novo como musica, esportes, atividades, uma nova linguagem de programação e etc... Sempre partimos do básico, na programação não é diferente, a codificação de inicialização é o famoso Hello World, onde consiste em apensas retornar a respectiva frase. Na Figura acima, vemos como é o processo de codificação, uma maneira simples e rápida. Na imagem mostra a interface gráfica e outras partes que o constitui, os blocos arredondados são chamado de nós que é uma representação visual do código em JavaScript.

Para construir o Hello World é necessário arrastar para a janela centro os "nós" um que projetado para saída de mensagens para outros tipos de nós. Neste primeiro nó está definido para a saída de uma string "Hello".

Próximo é uma nó função que é programado (JavaScript) incluindo a string "World" a qualquer mensagem que receber.

Agora que estes nós estão interligados notasse que a maioria tem um circulo cinza em sua volta, se o circulo se encontra no lado esquerdo, representa uma porta de entrada e se estiver do lado direito representa uma porta de saída e é possível conecta-los apenas arrastando um para o outro.

A próxima etapa é colocar o né de debug que vai receber a mensagem na tela de debug. Depois disso o programa está pronto, esse método em Node-Red é chamado de Fluxo.

### 3.4. BENEFÍCIOS.

Alguns dos maiores benéficos é praticidade em codificar os Nodes, ternando o trabalho todo mais simples e rápido de ser concluído.

Podemos dizer que o Node não é apenas exibir um conjunto de mensagens, também pode realizar vários conjuntos de comando, Web e Hardware e é nessa hora que as coisas ficam interessantes.

Dom Bramley um fotografo da IBM que tira fotos do céu noturno, criou um conjunto de Nodes para monitorar o tempo onde os equipamentos ficavam. A funcionalidade era simples construir um fluxo conectado em uma API de clima e a cada

determinado tempo fazia a verificação climática para previsões de chuvas locais e outro fluxo para calcular o tempo necessário para conseguir chegar ao equipamento, então quando apontava chuva na previsão, acendia um tipo de luz ambiente em sua casa. E logo depois outra luz em sua cabeceira para indicar o tempo necessário até o equipamento.

Existe uma comunidade de produção ativa para a criação de novos Nodes e já que a plataforma Node-Red é Open-Source hospedado na GitHub, portanto qualquer pessoa pode contribuir.

Os Fluxos em Node podem ser representados em JavaScript Object Notation ou JASON e pode ser facilmente exportados e importados para a plataforma Node-Red e podem ser compartilhados online.

#### 3.5. PACOTE NODE-RED ARDUINO.

## 3.5.1.INTRODUÇÃO.

O principal objetivo da loT é conectar os objetos em uma rede facilitando assim o uso, mas para determinadas aplicações e objetos são necessários sensores, justamente para colher, analisar e repassar dados para frente.

Para configurar esses sensores é necessário microcontroladores (Arduino) que é a parte física do projeto e para fazer a ligação com o software é necessário uma plataforma de desenvolvimento no caso desse projeto, os Nodes.

Para poder codificar os comandos que faz a comunicação com o Hardware, é necessário alguns pacotes.

O Node.js é um interpretador do código JavaScript que ajuda na codificação fazendo códigos manipular diversas conexão simultânea em uma mesma maquina.

#### 3.5.2. FUNCIONALIDADE E RECURSOS.

Sua principal função é aperfeiçoar e complementar as ferramentas existentes na linguagem.

Alguns pacotes são de modelagem, proporcionando aplicações e mais meios de melhorar o código. Existem pacotes específicos para criação de gráficos e diagramas, pacotes de cálculos matemáticos e pacotes para aperfeiçoar os testes com o objetivo de deixar o código mais completo.

#### 3.6. PACOTE JOHNNY-FIVE.

## 3.6.1.INTRODUÇÃO.

Johnny-Five é um pacote open-source, baseado em programação de framework, robóticas e IoT em JavaScript, foi desenvolvido pela Bocoup em 2012.

Mantido por uma comunidade de desenvolvedores de software e engenheiros de Hardware, sua comunidade é atualizada diariamente com diversos tipos de projetos em construção, fazendo isso visando explorar os projetos Open-source para o crescimento de novos projetos em robóticas, assim todos contribuem. Cerca de 80 Desenvolvedores contribuem para a construção robusta.

Capaz de ser escrito por vários programas, como: Arduino (Todos os modelos), Eletric IMP, Intel Galileo & Edson, PcDuino3, Raspberry PI entre outros.

Testado com uma variedade de placas, alguns Arduinos compatíveis. Para o nonarduino ou alguns plugins específicos para a plataforma IOS é possível a comunicação entre o código e o Hardware, indeferindo o tipo de linguagem que a plataforma falar.

#### 3.6.2. FUNCIONALIDADE E RECURSOS.

Baseado em um kit que controla projetos de hardwares, isso é basicamente a ideia do Johnny-Five, pode não resolver tudo que é necessário, mas se o foco for a desenvolver algo mais robusto, então é a escolha certa.

APis altamente testada, consiste em comportar e combinar com qualquer tipo de plataforma de hardware.

Isso permite uma total liberdade para crescer, construir e testar diversas bibliotecas JavaScript que quiser.

O programa descrito anteriormente como "Hello World", o código a seguir mostra como seria essa codificação em Johnny-Five.

```
var five = require("johnny-five");
  var board = new five.Board();

board.on("ready", function() {
    // Create an Led on pin 13
    var led = new five.Led(13);
    // Blink every half second
    led.blink(500);
    });
```

Johnny-Five proporciona uma leitura chamada Read-Eval-Print-Loop (REPL) que permite você controlar, mudar e atualizar comandos para o Hardware de forma interativa, enquanto está em execução, facilitando bastante para realizar os devidos testes.

#### 3.7. EXEMPLOS MOTIVACIONAIS.

#### 3.7.1. ARDUINO BLINK.

Arduino Blink é uma espécie de emulador do Arduino. Funciona como se fosse um Arduino digital, que contem também um protoboard para testar os projetos.

Dentro desse "emulador" alem de conter o microcontrolador e o protoboard. Possui também diversos tipos de sensores, LEDs, painéis, servos, controladores entre outros.

É ideal para quem não tem como comprar todos os equipamentos e quer aprender a linguagem. Assim como é possível testar se o projeto vai funcionar ou não, antes de fazer algo que possa ser definitivo.

O Arduino Blink é muito útil para quem está tentando aprender a linguagem, pois contém métodos para uma simples aprendizagem. Existe também vários modelos do Arduino e outros microcontrolador e computador.

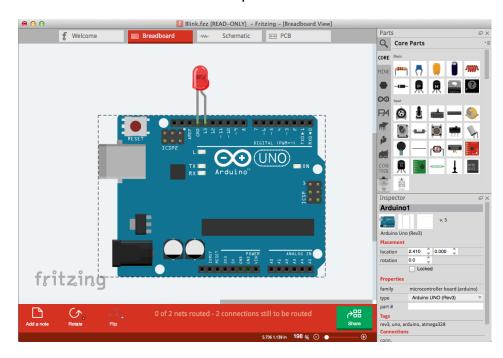


Figura 11: Fritzing – Exemplo do Arduino

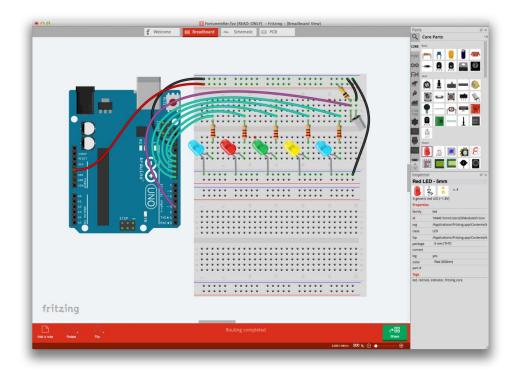


Figura 12: Fritzing – Exemplo com o Arduino e Protoboard

#### 3.7.2. ARDUINO DHT11 LOGGER

O Arduino DHT11, é uma espécie de sensor, que monitora a temperatura e umidade do clima e pode simplesmente incluir em qualquer tipo de Arduino.

Fornece uma leitura fácil e instantânea e sua biblioteca pré-definida, deixa tudo ainda mais simples para analisar as funções necessárias.

Consegue medir a temperatura entre zero e 50 graus Celsius e uma umidade em torno de 20 a 90%

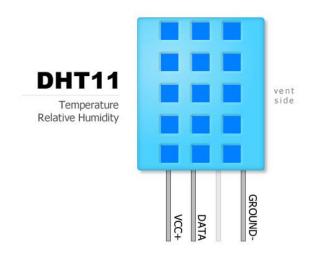


Figura 13: Sensor DHT11

Baseado em dados anteriores do George Hadjikyriacou, foi criada uma nova versão de biblioteca DewPoint, a versão 0.3.2, foi removido da classe os cálculos de conversão, para poder deixa-la mais rápida.

Agora a interface da classe suporta apenas a função de medir temperatura e umidade. Essa função vai fazer o processo de leitura e transmissão dos dados e uma função de tempo limite, que é para o reconhecimento dos dados.

A biblioteca retorna três valores, um mostrando se os sensores estão corretos, outra é a soma das verificações (Isso pode dizer que houve problemas nas leituras e os dados podem não está corretos) e por ultimo o tempo limite em que a comunicação falhou.

A seguir um exemplo de como é a codificação apara a leitura do clima.

```
//Programa : Sensor de umidade e temperatura DHT11
                       //Autor : Arduino e Cia
                           #include <dht.h>
#define dht dpin Al //Pino DATA do Sensor ligado na porta Analogica Al
                    dht DHT; //Inicializa o sensor
                             void setup()
                         Serial.begin(9600);
delay(1000);//Aguarda 1 seg antes de acessar as informações do sensor
                             void loop()
         DHT.read11(dht_dpin); //Lê as informações do sensor
                     Serial.print("Umidade = ");
                     Serial.print(DHT.humidity);
                        Serial.print(" % ");
                   Serial.print("Temperatura = ");
                    Serial.print(DHT.temperature);
                    Serial.println(" Celsius ");
 //Não diminuir o valor abaixo. O ideal é a leitura a cada 2 segundos
                             delay(2000);
```

#### 3.7.3. SENTIMENTAL TWEET 2 RGB.

Digamos que esteja no Twitter e entre algumas postagens, a pessoa poste algo emocional. Depois que algo é postado, não tem como voltar a trás, um Tweet levado pelo lado emocional pode ter diferente repercussões no futuro.

Agora imagina uma API que analisa os Tweets e consegue categorizar se a mensagem é emocional ou não, e ainda melhor, se for negativa ou positiva, se vai causar bons ou maus retornos para o usuário. Podemos assim chama-lo de uma espécie de Tweet sábio.

Alguns tipos de Tweets Positivos.

- Gosto desse lugar.
- O clima está tão bom.
- Acordei tão bem essa manhã

Agora alguns exemplos Negativos.

- Odeio ficar aqui.
- Não gosto desse carro.
- Acordei muito irritado essa manhã.

A API consegue diferenciar e classificar as frases, dividindo por palavras separadas. Assim fica mais fácil analisar o conjunto de palavras e definir se é positiva ou negativa.

Também é possível passar um fluxo(utilizando os Nodes) para acender um determinado tipo de LED se o Tweet for sentimental.

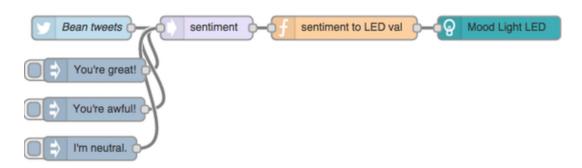


Figura 14: Internet das coisas Aplicação Nodes no Twitter

Fonte hackster.io 07 de Dezembro de 2015.

# 3.8. IMPLEMENTAÇÃO DOS NODES.

#### 3.8.1.MODELO ARQUITETURAL.

Node-Red é uma plataforma de fácil manuseio e bastante intuitiva, pois ser uma ferramenta na interface gráfica torna as coisas um pouco mais fáceis.

Os processos podem ser executados diretamente do navegador, bastando apenas arrastar os Nodes e fazer as ligações.

Existem os nós com o formato retangular e as pontas arredondadas. Cada um é de um tipo, existe nó para string, para condições e para finalizar o comando.

Alguns exemplos de Nodes customizados.

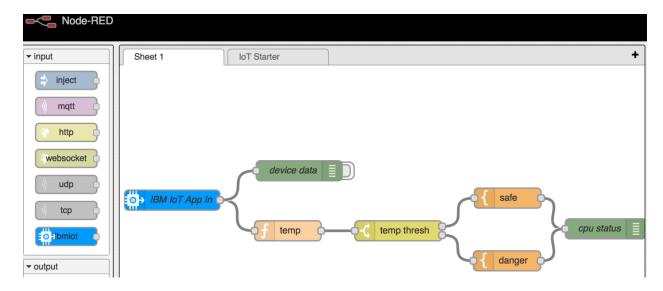


Figura 15: Modelo de Implementação dos Nodes

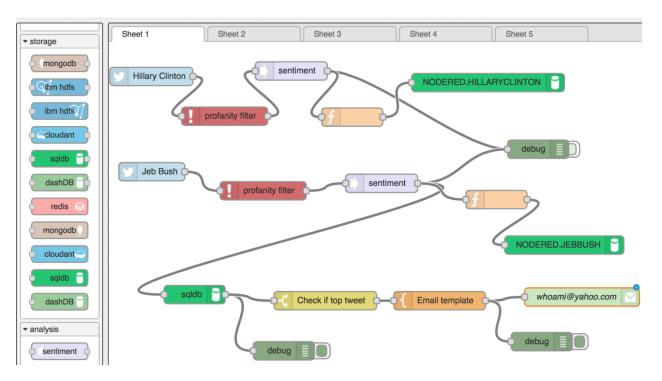


Figura 16: Modelo de Implementação dos Nodes.

#### 4. PROPOSTA DE TRABALHO

## 4.1. DEFINIÇÃO DO CENÁRIO EXPERIMENTAL.

O objetivo desse projeto é pesquisar como explorar os conceitos, as tecnologias e plataformas para *Internet of Things*. Visando explorar também os conceitos e funcionalidades da plataforma *Node-RED*. Nessa pesquisa será implementado também alguns exemplos com *Nodes* para a *Internet of Things* e programar conjuntos customizados de Nós. Fazendo através do Arduino para implementar um dispositivo.

O desenvolvimento deste projeto de pesquisa consiste em aprender mais sobre os *Nodes* para a área acadêmica científica atribuindo na prática o uso da *IoT*. Como existe pouco material de pesquisa, o intuito é aprender os meios e métodos simplificados/lúdicos com o *Node-RED*.

#### 4.2. NODES CUSTOMIZADOS.

Para construir Nodes Customizados é necessário alguns passos, primeiro terá que incluir o modelo. Esse novo Node é feito em JavaScript, onde é criado os parâmetros e as funções que deve ser executada. O Java é o primeiro passo em seguida é feito a codificação em HTML, como a plataforma Node-red é usada online através do navegador é necessário acrescentar a extensão para o novo Node.

Existe mais uma forma para acrescentar os modelos customizados, através do própria criação é possível exportar o novo modelo já montado e com todos os parâmetros já definidos, bastando assim apenas conecta-lo.

Quando estiver com esses processos prontos, basta colocar os arquivos JS e HTML no diretório onde o Node esta instalado, C:\Program Files\nodejs\node modules\node-red\nodes.

Seguindo para a execução é necessário que faça o upload do standart Firmadata que fica localizado nos exemplos prontos.

#### 4.2.1.NODE BLINK

O Node Blink para Arduino é um modelo já existente que foi proporcionado por outros desenvolvedores. Nesse projeto foi feito de uma maneira um pouco diferente, para que a luz do LED acendesse quando for passado o valor "1" e apagar quando o valor for "0".

A seguir segue o Exemplo em do código em JAVASCRIPT.

```
module.exports = function(RED) {
    "use strict";
    // var foo = require("foo-library");
    function SimpleBlink(n) {
        RED.nodes.createNode(this, n);
            this.active = n.active;
        this.topic = n.topic;
            this.value = n.value;
        var node = this;
            node.status({fill:"red", shape:"ring", text:"off"});
        this.on('input', function (msg) {
                  var msg = {};
                  msg.topic = this.topic;
                  msg.payload = this.value + ":" + this.active;
            node.send(msg);
        });
        this.on("close", function() {
    RED.nodes.registerType("simple-blink", SimpleBlink);
      RED.httpAdmin.post("/blink/:id/:value/:state",
RED.auth.needsPermission("blink.write"), function(req, res) {
        var node = RED.nodes.getNode(req.params.id);
        if (node != null) {
            try {
                        node.active = req.params.state === "true";
                        if (node.active) {
      node.status({fill:"green", shape:"dot", text:"on"});
                        } else {
      node.status({fill:"red", shape:"ring", text:"off"});
                node.receive();
                res.sendStatus(200);
            } catch(err) {
                res.sendStatus(500);
                node.error(RED. ("blink.failed", {error:err.toString()}));
            }
        } else {
            res.sendStatus(404);
    });
```

# Esse é o modelo para a criação do novo Node Customizado. Seguinte o modelo em HTML.

```
<script type="text/x-red" data-template-name="simple-blink">
    <div class="form-row">
        <label for="node-input-value"><i class="fa fa-tasks"></i> Value
(Port) </label>
        <input type="text" id="node-input-value" placeholder="Value">
    </div>
    <br/>
      <div class="form-row">
        <label for="node-input-topic"><i class="fa fa-tasks"></i> Top-
ic</label>
        <input type="text" id="node-input-topic" placeholder="Topic">
    </div>
    <br/>
    <div class="form-row">
        <label for="node-input-name"><i class="fa fa-tag"></i> Name</label>
        <input type="text" id="node-input-name" placeholder="Name">
    </div>
</script>
<script type="text/x-red" data-help-name="simple-blink">
   Exemplo de Node customizado para API de Node-RED.
   Exibe uma saÃ-da <code>msg</code> alternada entre <code>0</code> e
<code>1</code>.
</script>
<script type="text/javascript">
    RED.nodes.registerType('simple-blink',{
        category: 'FEMA',
           color: '#a6bbcf',
        defaults: {
                 active: {value:false},
            name: {value:""},
            topic: {value:""},
                 value: {value:"", required:true}
        },
        inputs:0,
        outputs:1,
        icon: "simple-blink.png",
        label: function() {
           return (this.name || this.topic || this.value || "simple-
blink") + " : " + this.active;
        labelStyle: function() {
            return this.name ? "node label italic" : "";
        },
           // align: "right",
           button: {
                 toggle: "active",
            onclick: function() {
                       var value =
(this.value).replace(/&/g,"&").replace(/</g,"&lt;").replace(/>/g,">"
```

```
var node = this;
                $.ajax({
                    url: "blink/" + this.id + "/" + value + "/" +
this.active,
                    type: "POST",
                    success: function(resp) {
RED.notify(node._("blink.success", {label:label}), "success");
                    error: function(jqXHR,textStatus,errorThrown) {
                        if (jqXHR.status == 404) {
RED.notify(node. ("common.notification.error", {message:node. ("common.notif
ication.errors.not-deployed") }), "error");
                         } else if (jqXHR.status == 500) {
RED.notify(node. ("common.notification.error", {message:node. ("inject.error
s.failed") }), "error");
                         } else if (jqXHR.status == 0) {
RED.notify(node. ("common.notification.error", {message:node. ("common.notif
ication.errors.no-response") }) , "error");
                         } else {
RED.notify(node. ("common.notification.error", {message:node. ("common.notif
tion.errors.unexpected", {status:jqXHR.status, message:textStatus})}), "error"
);
                    }
                });
            }
    });
</script>
```

Em Seguida existe a Sketch que deve ser executada no Arduino.

```
void setup() {
    Serial.begin(9600);
    pinMode(13, OUTPUT);
}
void loop() {
    if (Serial.available()) {
        String c = Serial.readString();
        digitalWrite(13, c.equals("1") ? HIGH : LOW);
        Serial.flush();
}
```

Por último tem a exportação feita de dentro do Node.

```
[{"id":"4f13b960.62b158","type":"inject","z":"135faab5.c8ae45","name":"
","topic":"","payload":"0","payloadType":"str","repeat":"","crontab":""
,"once":false,"x":213,"y":99,"wires":[["ecd06c55.86bb2"]]},{"id":"8e3c4
091.1387f","type":"inject","z":"135faab5.c8ae45","name":"","topic":"","
paylo-
ad":"1","payloadType":"str","repeat":"","crontab":"","once":false,"x":2
14,"y":166,"wires":[["ecd06c55.86bb2"]]},{"id":"ecd06c55.86bb2","type":
"serial
out","z":"135faab5.c8ae45","name":"arduino","serial":"d4490d06.93ca5","
x":433,"y":131,"wires":[]},{"id":"d4490d06.93ca5","type":"serial-
port","z":"135faab5.c8ae45","serialport":"COM12","serialbaud":"9600","d
ata-
bits":"8","parity":"none","stopbits":"1","newline":"\\n","bin":"false",
"out":"char","addchar":false}]
```

Para fazer a importação do código acima é necessário acessar o Menu no canto direito superior, Import e Clipboard.

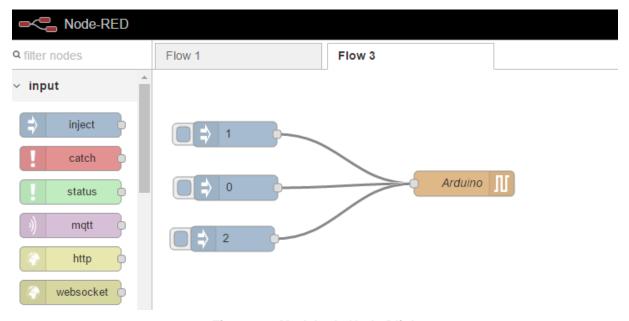


Figura 17: Modelo de Node Blink

## 4.2.2. NODE SETE SEGMENTO

Esse é um novo modelo de Node Customizado que foi desenvolvido para esse projeto. Um Node que controla um Display de Sete Segmentos.

Para adicionar esse Node é necessário o mesmo processo feito com o Blink.

#### Código em JAVASCRIPT.

```
module.exports = function(RED) {
    "use strict";
```

```
// var foo = require("foo-library");
    function SeteSequiment(n) {
        RED.nodes.createNode(this, n);
            this.active = n.active;
        this.topic = n.topic;
            this.value = n.value;
        var node = this;
            node.status({fill:"red", shape:"ring", text:"off"});
        this.on('input', function (msg) {
                  var msg = {};
                  msg.topic = this.topic;
                  msg.payload = this.value + ":" + this.active;
            node.send(msg);
        });
        this.on("close", function() {
        });
    RED.nodes.registerType("7-segmento", 7segmento);
      RED.httpAdmin.post("/SeteSegmento/:id/:value/:state",
RED.auth.needsPermission("SeteSegmento.write"), function(req, res) {
        var node = RED.nodes.getNode(req.params.id);
        if (node != null) {
            try {
                        node.active = reg.params.state === "true";
                        if (node.active) {
      node.status({fill:"green", shape:"dot", text:"on"});
                        } else {
      node.status({fill:"red", shape:"ring", text:"off"});
                node.receive();
                res.sendStatus(200);
            } catch(err) {
                res.sendStatus(500);
                node.error(RED. (" 7segmento.failed", {er-
ror:err.toString()}));
        } else {
            res.sendStatus(404);
    });
```

## Código em HTML.

```
<script type="text/x-red" data-template-name="SeteSegmento">
```

```
<div class="form-row">
        <label for="node-input-value"><i class="fa fa-tasks"></i> Value
(Port)</label>
        <input type="text" id="node-input-value" placeholder="Value">
    </div>
    <br/>
      <div class="form-row">
        <label for="node-input-topic"><i class="fa fa-tasks"></i> Top-
ic</label>
        <input type="text" id="node-input-topic" placeholder="Topic">
    </div>
   <br/>
   <div class="form-row">
        <label for="node-input-name"><i class="fa fa-tag"></i> Name</label>
        <input type="text" id="node-input-name" placeholder="Name">
    </div>
</script>
<script type="text/x-red" data-help-name="SeteSegmento">
   Exemplo de Node customizado para API de Node-RED.
</script>
<script type="text/javascript">
    RED.nodes.registerType('SeteSegmento',{
        category: 'FEMA-TCC',
            color: '#a6bbcf',
        defaults: {
                  active: {value:true},
            name: {value:""},
            topic: {value:""},
                 value: {value:"", required:true}
        } ,
        inputs:0,
        outputs:7,
        icon: "7segmento.png",
        label: function() {
           return (this.name || this.topic || this.value || "SeteSegmen-
to") + " : " + this.active;
        labelStyle: function() {
            return this.name ? "node label italic" : "";
            // align: "right",
           button: {
                  toggle: "active",
            onclick: function() {
                       var value =
(this.value).replace(/&/g, "&").replace(/</g, "&lt;").replace(/>/g, "&gt;"
);
                var node = this;
                $.ajax({
                    url: "SeteSegmento/" + this.id + "/" + value + "/" +
this.active,
                    type: "POST",
                    success: function(resp) {
```

```
RED.notify(node. ("SeteSegmento.success", {label:label}), "success");
                    } ,
                    error: function(jqXHR,textStatus,errorThrown) {
                        if (jqXHR.status == 404) {
RED.notify(node. ("common.notification.error", {message:node. ("common.notif
ication.errors.not-deployed") }), "error");
                         } else if (jqXHR.status == 500) {
RED.notify(node._("common.notification.error", {message:node._("inject.error")
s.failed")}),"error");
                        } else if (jqXHR.status == 0) {
RED.notify(node. ("common.notification.error", {message:node. ("common.notif
ication.errors.no-response") }), "error");
                         } else {
RED.notify(node. ("common.notification.error", {message:node. ("common.notif
tion.errors.unexpected", {status:jqXHR.status, message:textStatus})}), "error"
                    }
                });
            }
    });
</script>
```

#### Sketch no Arduino.

```
void setup() {
 // put your setup code here, to run once:
int i;
 pinMode(2, OUTPUT);
 pinMode(3, OUTPUT);
 pinMode(4, OUTPUT);
 pinMode(5, OUTPUT);
 pinMode(6, OUTPUT);
 pinMode(7, OUTPUT):
 pinMode(8, OUTPUT);
 controlarPonto(0);
 void controlarPonto(byte valor){
  digitalWrite(9, valor);
  }
  void imprimirDigito(byte digito){
  int indice = 0;
for (byte contador =0; contador <7; contador ++){
 digitalWrite(portas[indice],
 digitos_display[digito][contador]
 );
indice++;
 }
}
void loop() {
 if (Serial.available()){
  String c = Serial.readString();
  for (int i = 0; i <=9; i++){
   imprimirDigito(i);
   delay(500);
  Serial.flush();
```

## Import no Node-Red.

```
[{"id":"c7e8f2fc.971fc","type":"serial-port","z":"56452ca6.2c3d84","serialport":"COM12","serialbaud":"57600","data bits":"8","parity":"none","stopbits":"1","newline":"\\n","bin":"false","out ":"char","addchar":false}, {"id":"f5158864.3a6ce8","type":"inject","z":"5645 2ca6.2c3d84","name":"","topic":"","payload":"0","payloadType":"num","repeat ":"","crontab":"","once":false,"x":131,"y":47,"wires":[["89b0ea89.9daef8"]]}, {"id":"dd17064d.fa71c8","type":"inject","z":"56452ca6.2c3d84","name":"","
```

```
to-
pic":"", "payload":"1", "payloadType": "num", "repeat": "", "crontab": "", "once":f
se, "x":131, "y":92, "wires":[["89b0ea89.9daef8"]]}, {"id":"7aabdb6.f912924", "t
ype":"inject","z":"56452ca6.2c3d84","name":"","topic":"","payload":"2","pay
adType":"num", "repeat":"", "crontab":"", "once":false, "x":132, "y":131, "wires"
:[["89b0ea89.9daef8"]]},{"id":"19911e17.dba5f2","type":"inject","z":"56452c
a6.2c3d84", "name": "", "topic": "", "payload": "3", "payloadType": "num", "repeat":
"", "crontab": "", "once": false, "x":130, "y":171, "wires": [["89b0ea89.9daef8"]]}
,{"id":"3e748b28.0e6644","type":"inject","z":"56452ca6.2c3d84","name":"","t
opic":"", "payload":"4", "payloadType": "num", "repeat": "", "crontab": "", "once":
se, "x":129, "y":216, "wires":[["89b0ea89.9daef8"]]}, {"id":"f11a7d5b.59a39", "t
ype":"inject","z":"56452ca6.2c3d84","name":"","topic":"","payload":"6","pay
adType":"num", "repeat":"", "crontab":"", "once":false, "x":130, "y":305, "wires"
:[["89b0ea89.9daef8"]]},{"id":"66a5c28c.e06bcc","type":"inject","z":"56452c
a6.2c3d84", "name": "", "topic": "", "payload": "7", "payloadType": "num", "repeat":
"", "crontab":"", "once":false, "x":129, "y":348, "wires":[["89b0ea89.9daef8"]]}
,{"id":"8457619b.3ec2e","type":"inject","z":"56452ca6.2c3d84","name":"","to
pic":"","payload":"8","payloadType":"num","repeat":"","crontab":"","once":f
se,"x":131,"y":387,"wires":[["89b0ea89.9daef8"]]},{"id":"b44dbf31.38cd6","t
ype":"inject","z":"56452ca6.2c3d84","name":"","topic":"","payload":"5","pay
adType":"num", "repeat":"", "crontab":"", "once":false, "x":130, "y":262, "wires"
:[["89b0ea89.9daef8"]]},{"id":"e17f108c.2745c","type":"inject","z":"56452ca
6.2c3d84", "name": "", "topic": "", "payload": "9", "payloadType": "num", "repeat": "
","crontab":"","once":false,"x":131,"y":427,"wires":[["89b0ea89.9daef8"]]},
{"id":"89b0ea89.9daef8","type":"serial
out","z":"56452ca6.2c3d84","name":"Arduino","serial":"c7e8f2fc.971fc","x":3
93, "y":219, "wires":[]}]
```



Figura 18: Modelo de Node Sete Segmentos

## 4.2.3. NODE POTENCIOMETRO

Um novo modelo de Node Customizado, esse é para controlar um potenciômetro e faze-lo interagir com outros LEDS.

Seguindo o mesmo padrão dos anteriores para a implantação dos novos Nodes.

## Código JAVASCRIPT.

```
module.exports = function(RED) {
    "use strict";
    // var foo = require("foo-library");
    function Potenc(n) {
        RED.nodes.createNode(this, n);
            this.active = n.active;
        this.topic = n.topic;
            this.value = n.value;
        var node = this;
            node.status({fill:"red", shape:"ring", text:"off"});
        this.on('input', function (msg) {
                  var msq = {};
                  msq.topic = this.topic;
                  msq.payload = this.value + ":" + this.active;
            node.send(msg);
        });
        this.on("close", function() {
        });
    RED.nodes.registerType("Potenciometro", Potenciometro);
      RED.httpAdmin.post("/Potenciometro/:id/:value/:state",
RED.auth.needsPermission("Potenciometro.write"), function(req, res) {
        var node = RED.nodes.getNode(req.params.id);
        if (node != null) {
            try {
                        node.active = reg.params.state === "true";
                        if (node.active) {
      node.status({fill:"green", shape:"dot", text:"on"});
                        } else {
```

## Código HTML.

```
<script type="text/x-red" data-template-name="Potenciometro">
    <div class="form-row">
        <label for="node-input-value"><i class="fa fa-tasks"></i> Value
(Port) </label>
        <input type="text" id="node-input-value" placeholder="Value">
    </div>
    <br/>
     <div class="form-row">
        <label for="node-input-topic"><i class="fa fa-tasks"></i> Top-
ic</label>
        <input type="text" id="node-input-topic" placeholder="Topic">
    </div>
   <br/>
    <div class="form-row">
        <label for="node-input-name"><i class="fa fa-tag"></i> Name</label>
        <input type="text" id="node-input-name" placeholder="Name">
   </div>
</script>
<script type="text/x-red" data-help-name="Potenciometro">
  Exemplo de Node customizado para API de Node-RED.
</script>
<script type="text/javascript">
   RED.nodes.registerType('Potenciometro',{
        category: 'FEMA-TCC',
           color: '#a6bbcf',
        defaults: {
                 active: {value:true},
            name: {value:""},
                 value: {value:"",
                 required: true }
        },
        inputs:0,
        outputs:1,
```

```
icon: "Potenciometro.png",
        label: function() {
            return (this.name || this.topic || this.value || "Potenc") +
" : " + this.active;
        labelStyle: function() {
            return this.name ? "node label italic" : "";
        },
            // align: "right",
            button: {
                  toggle: "active",
            onclick: function() {
                        var value =
(this.value).replace(/&/g, "&").replace(/</g, "&lt;").replace(/>/g, "&gt;"
);
                var node = this;
                $.ajax({
                    url: "Potenciometro/" + this.id + "/" + value + "/" +
this.active,
                    type:"POST",
                    success: function(resp) {
RED.notify(node. ("Potenciometro.success", {label:label}), "success");
                    },
                    error: function(jqXHR, textStatus, errorThrown) {
                        if (jqXHR.status == 404) {
RED.notify(node._("common.notification.error", {message:node._("common.notif
ication.errors.not-deployed") }), "error");
                         } else if (jqXHR.status == 500) {
RED.notify(node._("common.notification.error", {message:node. ("inject.error")
s.failed") }), "error");
                         } else if (jqXHR.status == 0) {
RED.notify(node._("common.notification.error", {message:node._("common.notif
ication.errors.no-response") }), "error");
                         } else {
RED.notify(node. ("common.notification.error", {message:node. ("common.notif
ica-
tion.errors.unexpected", {status:jqXHR.status, message:textStatus})}), "error"
);
                    }
                });
    });
</script>
```

## Sketch Arduino.

int portaLed = 13;

```
void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
pinMode(portaLed, OUTPUT);
}

void loop() {

    if (Serial.available()){
        String c = Serial.readString();
        int valorPotenciometro = analogRead(A1);
        int novoValorPotenciometro =
            map(valorPotenciometro,0, 1023,0,255);
            analogWrite(portaLed, novoValorPotenciometro);
        Serial.println(novoValorPotenciometro);
        delay(100);
        Serial.flush;
    }
}
```

# 5. CONCLUSÃO

Com esta pesquisa disponibiliza novos materiais teóricos e práticos sobre *Node-RED* e modos de integração em *Internet of Things*. Adquirindo assim uma dominação do assunto para diversos tipos de projetos, como em sensores e novos modos de codificação.

Pretende-se disponibilizar o novo conjunto de *Nodes* Customizados e os novos trabalhos e projetos, mostrando na prática, para que no futuro possam ser aproveitados ou melhorados por outras pessoas.

A pesquisa abordou o conhecimento e a utilização dessa nova ferramenta existente, focando no aprendizado e aperfeiçoamento, que também contém uma proposta de estudo para a implementação de ferramentas de apoio em plataformas do *Java* e *Arduino*.

Apesar de ser uma tecnologia recente os Nodes prometem vir pra ficar, pois a simplicidade e a praticidade de aplicação é bastante interessante e pode crescer muito.

O node pode ser um pouco complexo de início, pois não existem muitas informações sobre e a maioria disponível está no próprio site e no fórum oficial. A instalação é um pouco trabalhosa, pois para reconhecer os pacotes e realizar as instalações disponíveis para o funcionamento requer um bom conhecimento.

Com esse projeto mostra os passos a passos para a instalação exemplos para começar a desenvolver.

## 5.1. TRABALHOS FUTUROS.

Futuros trabalhos podem ser desenvolvidos com o término da pesquisa, novos tipos de Nodes para serem usados no dia a dia, como um sensor de presença quando chegar à residência, um controlador de temperatura ou mesmo aplicações de nodes no controle de drones e robôs.

Além de novos componentes que podem ser usados para a implementação dos nodes, pode usar os diversos modelos de Arduinos (dependendo do tipo de projeto) e também o Raspberry PI. Com a facilidade desses microprocessadores o desenvolvimento dos node podem ser feito tanto em JavaScript quanto em Jason e HTML. Cada plataforma tem as bibliotecas necessárias para a utilização de motores, sensores, controladores e até display de imagem.

# **REFERÊNCIAS**

**Arduino e Cia**: site voltado a tudo relacionado ao Arduino, Acessado 20 de Março de 2016 http://www.arduinoecia.com.br/2013/05/sensor-de-umidade-e-temperatura-dht11.html.

**ASHTON**, Kevin. *Thet 'Internet of Things' Things*. 2009. Entrevista sobre o artigo do mesmo nome para o Jornal renomado RFID Jornal– http://www.rfidjornal.com/articles/view?4986.

Descrição e analise da placa Intel Galileo, publicada no próprio site da Intel. http://www.intel.com.br/content/www/br/pt/do-it-yourself/galileo-maker-quark-board.html#

VERMESAN,O.; FRIESS, P. Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems: Publicado pela editora River Publishers Aalborg

2013. http://www.internet-of-things-research.eu/pdf/Converging\_Technologies\_for\_Smart\_Environments\_and\_Integrated Ecosystems IERC Book Open Access 2013.pdf

**EVANS**, Dave. Artigo publicado pelo cisco em abril de 2011 Intitulado como *Internet das coisas*. *Como a próxima evolução da Internet está mudando tudo*. http://www.cisco.com/web/BR/assets/executives/pdf/internet\_of\_things\_iot\_ibsg\_041 1final.pdf

**GSM Association:** Understanding the Internet of Things (IoT). Artigo publicado em Júlio de 2014.

**HEATH, N**, How IBM's Node-RED is hacking together the internet of things, postado dia 13 de março de 2014 http://www.techrepublic.com/article/node-red/.

**KARASINSKI**, V. Artigo publicado em 09 de Janeiro de 2014. Analise feita para a Intel sobre a placa Intel Edson.

**LIMA**, T. Analise publicada em 14 de março de 2014, sobre a nova placa. http://www.embarcados.com.br/intel-galileo/

PIRES, P.F.; DELICATO, F.C.; BATISTA, T; BARROS, T; CAVALCANTE, E.; PITANGA, M.; Minicurso SBRC publicado em 2015. Cap. 3 Plataformas para internet das coisas.

**PRESS**, Gil. Internet of Things By The Numbers: Market Estimates And Forecasts . 2014. Matéria publicada em um site de tecnologia. Pesquisa sobre o crescimento da IoT e suas expectativas,

http://www.forbes.com/sites/gilpress/2014/08/22/internet-of-things-by-the-numbers-market-estimates-and-forecasts/

**PESSOA,** C.R.M, JAMIL, G.L.; Colaboradores: GEREMIAS, T.; SANTOS, P.H.S.; ROSA, M.M.F.; SILVA, T.B.; Trabalho de Graduação por estudantes de Engenharia de telecomunicações da FEA-Fumec. A INTERNET DAS COISAS: SERÁ A INTERNET DO FUTURO OU ESTÁ PRESTES A SE TORNAR A REALIDADE DO PRESENTE?

**SGIER,** M. Licenciado e contribuído por MIT 2015- Johnny-Five https://github.com/rwaldron/johnny-five.

**SINGER,** T.; TUDO CONECTADO: CONCEITOS E REPRESENTAÇÕES DA INTERNET DAS COISAS. Artigo apresentado no NT 1 – Sociabilidade, novas tecnologias e práticas interacionais do II Simpósio em Tecnologias Digitais e Sociabilidade realizado nos dias 11 e 12 de outubro de 2012 em Salvador-BA

Site Oficial Johnny-Five, acessado dia 20 de Março de 2016 http://johnny-five.io/.

Site Sobre Nodes: acessado dia 20 de Março de 2016 http://node-ardx.org/.

**Site** oficial do Arduino: A DHT11 For Arduino 16 de novembro de 2014 http://playground.arduino.cc/Main/DHT11Lib.

**Site** sobre tecnologia, weet sentiment to LED using Node-RED, publicado dia 7 de dezembro de 2015 https://www.hackster.io/punch-through/tweet-sentiment-to-led-using-node-red-6a380a.

**Site** da IBM, Tecnologia emergente – Node-Rede, acessado dia 20 de março de 2016 https://www-304.ibm.com/connections/blogs/et/entry/node-red?lang=pt\_br.

**WALDRON**, R.; JavaScript Arduino Programming on Nodes postado em 15 de maio de 2012 e acessado no dia 19 de março de 2016 https://bocoup.com/weblog/javascript-arduino-programming-with-nodejs.

**ZAZESKI,** S. Arduino LCD HDT11 Temperature, publicado dia 16 de novembro de 2013 http://steve.zazeski.com/arduino-dht11-temperature-and-humidity-data-logger/