

## **Artigo sobre o trabalho Assist Automation, AA!**

Como a próxima geração de Internet pode mudar as coisas?

Tales Igor Ebert<sup>1</sup> Artur Comuneo<sup>1</sup> Leandro Vian<sup>1</sup> Fabrício Pretto<sup>1</sup>

<sup>1</sup>Engenharia de Software – Centro Universitário UNIVATES  
Caixa Postal 155, Bairro Universitário, Lajeado- RS CEP 95.900-00

***Abstract:** The Internet of Things is a technological revolution that aims to connect any object to the World Wide Web. Increasingly arise appliances, transportation and even shoes, clothes and handles connected to the Internet and to other devices such as computers and smartphones. The connection between applications and objects is one of the biggest challenges for the Internet of things. Allow objects to communicate with each other by sending and receiving information between applications that transform data into information makes the digital and physical world become one. A major challenge is to combine theory with practice and bring together various areas of knowledge in a single purpose. The article presents a cloud automation experiment that brings together a number of challenges, from the creation of the automation prototype development of a software to automate and control environments remotely.*

**Keywords:** IoT, automation, embedded software.

**Resumo:** A Internet das Coisas é uma revolução tecnológica que tem como objetivo conectar qualquer objeto à rede mundial de computadores. Cada vez mais surgem eletrodomésticos, meios de transporte e até mesmo tênis, roupas e maçanetas conectadas à Internet e a outros dispositivos, como computadores e *smartphones*. A conexão entre aplicativos e objetos é um dos maiores desafios para Internet das Coisas. Permitir que objetos se comuniquem entre si, enviando e recebendo informações entre aplicativos que transformam dados em informações fazem com que o mundo digital e físico se tornem uma coisa só. Um dos grandes desafios é associar a teoria com a prática e reunir várias áreas de conhecimento em um único propósito. O artigo apresenta um experimento de automação em nuvem que reúne diversos desafios, desde a confecção do protótipo de automação ao desenvolvimento de um software para automatizar e controlar ambientes remotamente.

**Palavras chave:** Internet das Coisas, automação, software embarcado.

## **1 Introdução**

A Internet das Coisas (IoT), ou simplesmente dos objetos mudará tudo [CISCO, 2011]. Isso pode ser considerado uma declaração arrojada, mas considerando que a Internet mudou a forma e velocidade de propagação do conhecimento, claramente é uma das invenções mais importantes da humanidade.

Com os objetos conectados, as possibilidades de coletar, analisar e distribuir informação e transformar tudo em conhecimento é um salto na evolução da Internet. As informações obtidas possibilitam um maior controle e compreensão de como os sistemas interagem entre eles transformando dados em informações.

A Internet das Coisas promete fechar a lacuna entre ricos e pobres, possibilitando melhorar a distribuição de recursos no mundo para aqueles que mais precisam, permitir entender a forma como o mundo funciona, ou até mudar a forma como vivemos e monitoramos as nossas vidas, sendo mais proativos [CISCO, 2011]. Mas existem barreiras, uma delas é a comunicação e a geração de energia, o que deverá nas próximas gerações alimentar milhões de sensores.

Mas em paralelo, existem outras pesquisas que viabilizem que isso aconteça, uma delas é a geração de energia solar, e até a transição de IPV4 para IPV6 que vai permitir uma melhor escalabilidade na comunicação entre os objetos.

O objetivo deste artigo é apresentar a construção de uma solução de automação e monitoramento de ambientes. Para este trabalho foi escolhido como estudo de caso, uma fonte de jardim, que reunindo conceitos da Internet das Coisas, vai possibilitar conectar pequenos objetos, como, sensores, câmeras e disjuntores para coleta de dados e acionamento de dispositivos remotos de forma automática ou manual.

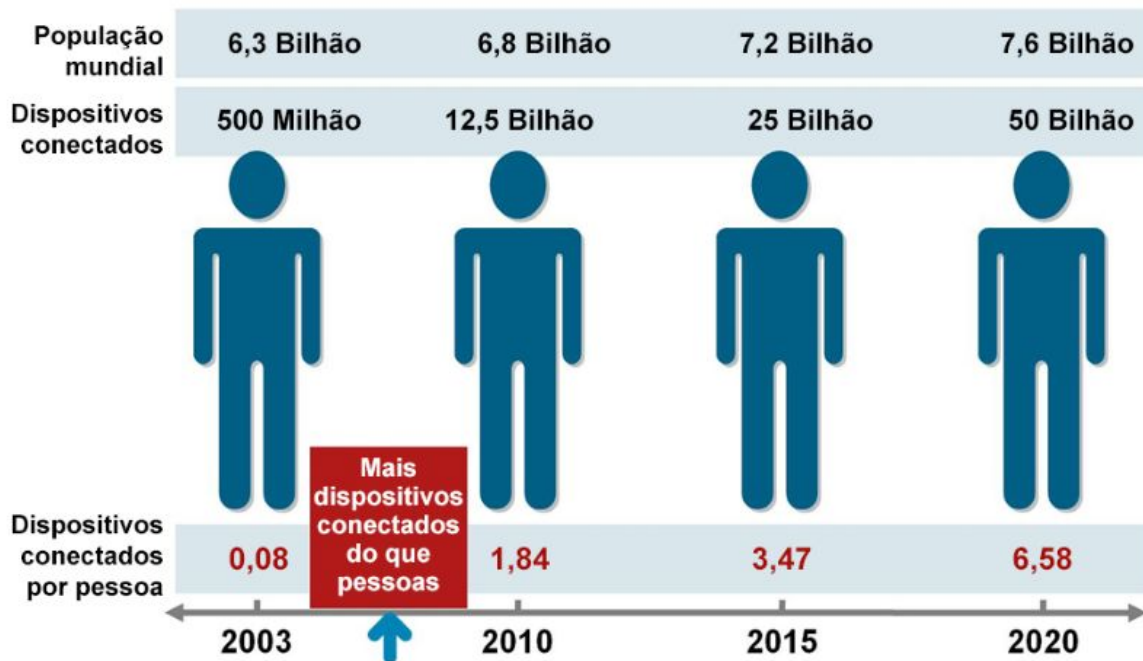
O trabalho é dividido em três partes. O capítulo um apresenta uma breve introdução sobre o histórico da Internet das Coisas. O capítulo dois descreve o embasamento teórico sobre o tema abordado. O capítulo três descreve a montagem do protótipo de hardware e a construção do aplicativo para o usuário interagir com os objetos.

## **2 A Internet das Coisas na atualidade**

De acordo com a Cisco, até o abril de 2001, foram conectados a Internet mais objetos do que o número de pessoas que existe no mundo. Em 2003, existiam 6,3 bilhões de pessoas e 500 milhões de dispositivos conectados na Internet.

Com a criação de *smartphones* e *tablets*, estima-se que o número de dispositivos conectados à Internet dê um salto para 50 bilhões em 2020, e a população subiu para 7,6 bilhões [CISCO, 2011].

A Figura 1, mostra uma projeção sobre o crescimento de dispositivos por pessoa.



Fonte: Cisco IBSG, abril de 2011

Figura 1. A projeção do crescimento da Internet das Coisas até 2020[CISCO, 2011].

A estimativa não leva em conta os dispositivos que ainda estão desconectados na Internet, um grande número de dispositivos com suporte à Internet permanece desconectado, segundo a ITU (União Internacional de Telecomunicações), apenas 40% da população tem acesso à Internet. Logo, são 4 bilhões de pessoas no mundo que ainda não têm Internet. Mas essa realidade pode mudar, com projetos que pretendem levar a Internet de uma forma barata para todos, a exemplo o projeto Loon da Google, o qual corresponde a uma rede de balões que viaja pelos confins do espaço. A finalidade do projeto é conectar pessoas em áreas rurais e remotas, ajudar a preencher falhas de cobertura e ajudar a recuperar a conexão com a Internet em áreas que passaram por desastres [LOON, 2015].

## 2.1 Dispositivos como uma única rede

Existem inúmeros dispositivos que controlam coisas específicas, como sistemas de ventilação, sistemas elétricos e eletrônicos, automações diversificadas e até na linha automobilística existem vários produtos que controlam os veículos. A medida que a Internet das Coisas evoluir, é natural que esses dispositivos que trabalham isolados passem a estar conectados a uma rede, desta forma transformando informações isoladas em informações integradas [CISCO, 2011].

Os usuários esperam que as informações sejam transferidas no momento e tão rapidamente quanto estão sendo solicitadas, isso causa uma intensa competição pela informação, seja nos negócios ou no âmbito pessoal [RODRIGUES, 2013]. A troca de informação entre diferentes hardwares e softwares precisa ocorrer de forma transparente ao usuário e como um único sistema, de forma eficiente com o menor custo de processamento.

Os sistemas embarcados utilizados na Internet das Coisas, apresentam um apoio no desenvolvimento urbano através de reconhecimento inteligente, geo localização, rastreamento, gerenciamento de dispositivos, monitoramento pessoas e ambientes em missão crítica [RODRIGUES, 2013].

### **3. O protótipo - Assist Automation**

Este capítulo apresenta a descrição dos elementos empregados para o desenvolvimento da solução de automação proposta, bem como a arquitetura de hardware e software utilizados. O protótipo desenvolvido e validado, contou com implementações de hardware, envolvendo componentes eletrônicos e peças, como também implementações de software, para gerenciar captura, processamento, armazenamento, transmissão e exibição das informações ao usuário. Por meio de uma solução web, ou seja, acessível via navegador de Internet, é possível monitorar um ambiente e interagir com os elementos nele contidos, tudo isso, de forma remota, a partir de um *smartphone*.

Na camada de hardware, o trabalho baseou-se na plataforma de desenvolvimento Arduino, que combinado a uma interface de comunicação por rede possibilita infinitas combinações de aplicabilidade. Com uma aplicação modular, possibilita a comunicação e coleta de dados de vários dispositivos remotos.

Foram selecionados diversos tipos de sensores, o DTH11 (sensor de temperatura), LM35 (sensor de temperatura de baixa precisão) e DTH22 (sensor de temperatura e umidade de alta precisão), DS18B20 (sensor de temperatura da água). O Arduino R3 foi escolhido como plataforma de desenvolvimento embarcado, que conta com microcontrolador AtMega328. Os demais materiais utilizados foram reciclados de equipamentos queimados.

Uma das etapas mais importantes deste projeto foi a definição deste ferramental a ser utilizado pela solução de integração. Antes da preparação do ambiente e implementação da solução e do protótipo, um tempo considerável foi reservado para a definição de quais destas ferramentas e tecnologias seriam utilizadas no projeto.

Para programação do *front-end* foi escolhido o Framework Adianti versão 3.0 para PHP, por ser livre de licença e reconhecido pela sua robustez. O banco de dados escolhido foi o PostgreSQL.

Para a camada de comunicação entre hardware e software, vários *webservices* permitem a comunicação entre diferentes arquiteturas, estes atuam como uma *interface* de comunicação. O *webservice* é uma solução utilizada na integração de sistemas e na

comunicação entre aplicações diferentes. Com esta tecnologia é possível que novas aplicações possam interagir com aquelas que já existem e que sistemas desenvolvidos em plataformas diferentes sejam compatíveis. Os *webservices* são componentes que permitem às aplicações enviar e receber dados em diferentes formatos [WEBSREST, 2011].

O *webservice* estabelece comunicação com o hardware através de comunicação REST, que é qualquer comunicação sobre HTTP/HTTPS que não tenha nenhuma camada no meio. O REST é favorável a compatibilidade, e pode ser aplicado em dispositivos com baixa capacidade de processamento e recursos, pois trata-se de simples requisições HTTP [WEBSREST, 2011].

Para segurança de transmissão entre equipamentos e software, foi escolhido a utilização de um certificado SSL, que se trata do processo mais rigoroso de validação da Internet. Inclui a verificação de que a empresa foi legalmente registrada, encontra-se ativa e que detém o registro do domínio para o qual o certificado será emitido, além de dados adicionais, como o endereço físico. Isso garante que os dados serão criptografados e somente o cliente e o lado do servidor possuem a chave para descriptografar a informação [CERT.BR, 2015].

A Figura 2 apresenta o diagrama que explica de forma macro como funciona a integração de hardware e software sobre a visão de arquitetura do AA (Assist Automation). O trabalho focou-se em criar um sistema de automação de uma fonte de jardim, que permite o monitoramento de sensores, acionamento de bombas, assistir a câmera em tempo real e acionar a iluminação remotamente.

A arquitetura foi pensada em escalabilidade, e para permitir que a aplicação colete dados de diferentes hardwares em diferentes tecnologias, uma camada de *webservice* foi pensada desde o início, desta forma qualquer novo hardware que seja implementado a camada do *webservice* tem a responsabilidade de implementar essa tecnologia, se comunicar com o dispositivo remoto e tratar os dados no formato definido pelo sistema, desta forma o sistema *front-end* tem um baixo nível de acoplamento com a tecnologia dos dispositivos.

O *webservice* fornece uma funcionalidade de caixa preta para a aplicação remota, ele executa uma unidade de trabalho, como por exemplo, a coleta e envio de dados para um hardware, validação e tratamento dos dados. Depois que realiza essa função o qual foi ordenado, está pronto para atender a outro pedido [CURBERA, 2002].

Nesse contexto de interação entre software e hardware, o *webservice* irá cumprir um dos papéis mais importantes do trabalho, e deve-se ressaltar que, ao mesmo tempo que a solução fica modular, ela depende 100% do funcionamento eficiente da camada de *webservice*.

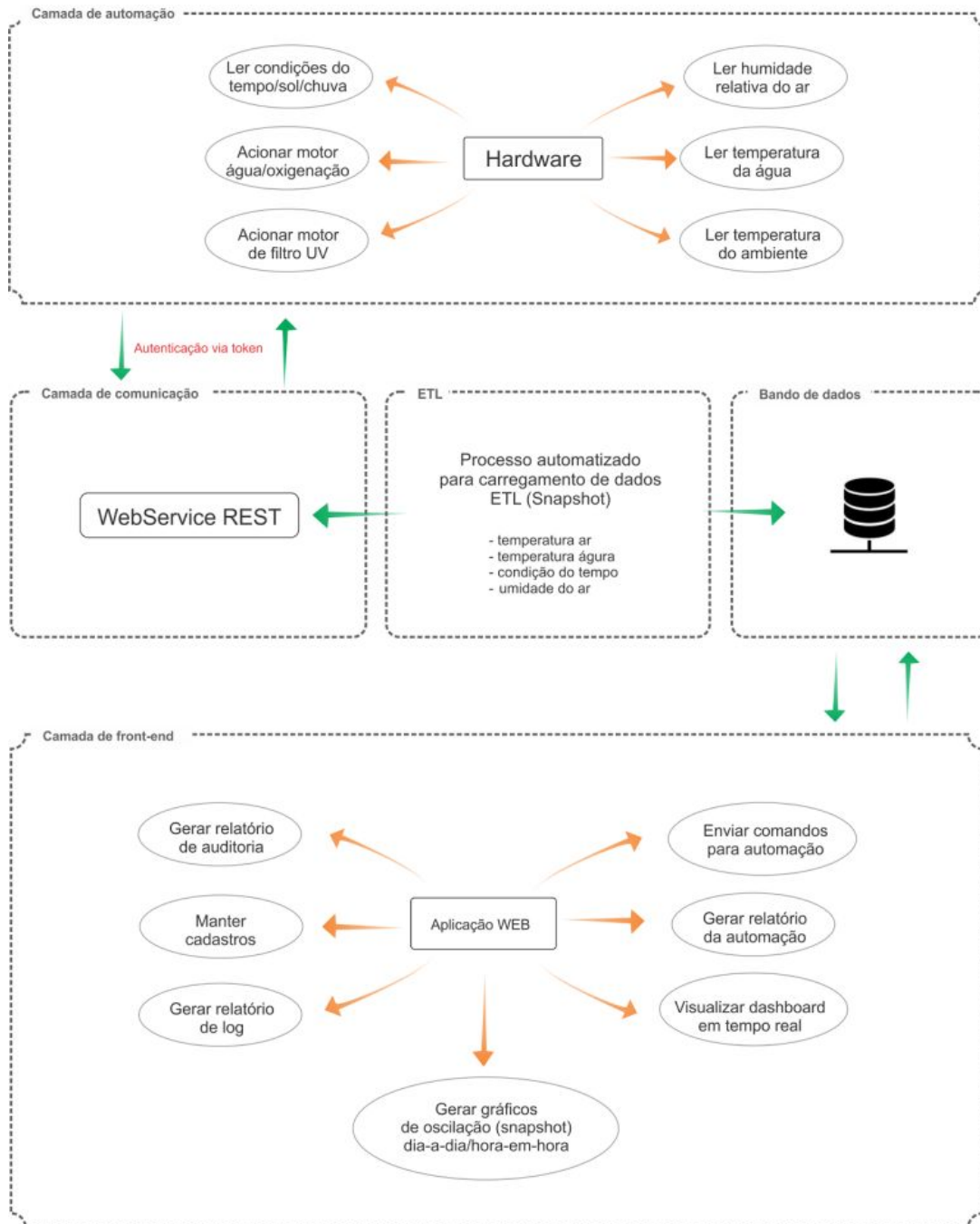


Figura 2. Visão de arquitetura do Assist Automation, AA!

Batizado como Assist Automation, ou simplesmente AA, o aplicativo de *front-end* deve cumprir as funcionalidades de coletar informações de diferentes hardwares.

Conforme a Figura 3, o processo de carregamento/leitura dos dados do hardware trabalha com o conceito do processo de ETL (Extract, Transform and Load), destina-se à extração dos dados do hardware, na transformação da informação para o padrão da aplicação do usuário (*front-end*) e gravação dos dados processados para a base de destino.

O processo será executado de forma assíncrona para evitar problemas de disputa pelo hardware no *front-end*, a extração e carga são obrigatórios nesse processo, sendo a transformação e limpeza opcional [ABREU, 2008].

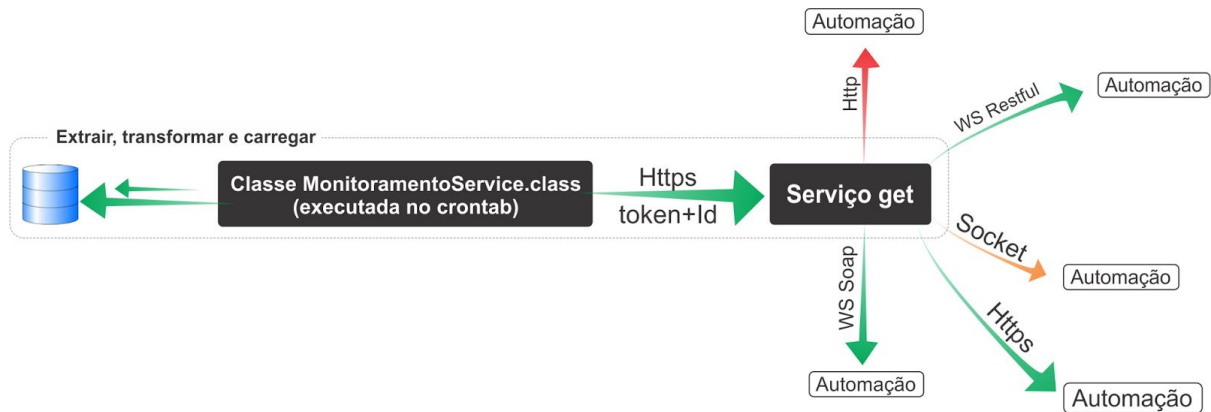


Figura 3: Esquema técnico do processo de carregamento e transformação dos dados com protocolos de comunicação distintos;

### 3.1 Ambiente de testes do protótipo

Para a execução do trabalho, foi criado um ambiente de testes capaz de reproduzir todas características técnicas iguais ao ambiente de produção. O ambiente de testes simula testes de carga e de estresse do sistema para testar sua estabilidade. É importante que o ambiente de teste seja capaz de reproduzir situações inesperadas no ambiente de produção [ALMEIRA, 2010].

Qualquer recurso de um sistema, hardware, software ou largura de banda podem virar um gargalo. Em aplicações de qualquer natureza, gargalos afetam diretamente o sucesso de um produto no que diz respeito à escalabilidade [ORACLE, 2009].

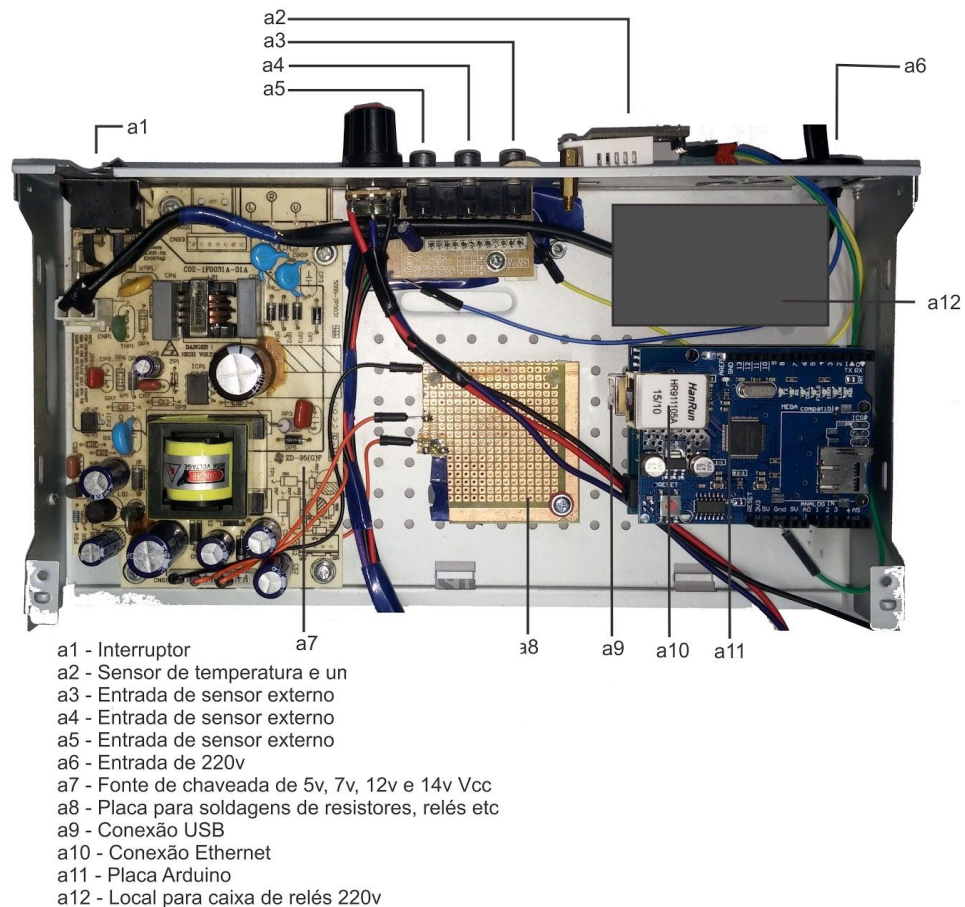
Os testes de desempenho podem ser definidos como “testes realizados para avaliar a conformidade de um sistema ou componente com os requisitos de desempenho” [ORACLE, 2009]. Exatamente como a literatura retrata, os testes de carga ajudaram a identificar rapidamente as limitações e gargalos do protótipo de hardware.

### 3.2 Construção do hardware

O Arduino é um hardware programável e tem um microcontrolador de 16Hertz, com 32Kbytes de memória flash, sendo que 5Kbytes é reservado para o bootloader.

É possível adicionar recursos à placa Arduino, através do acoplamento de placas auxiliares, como por exemplo uma interface de rede, que é indispensável para execução desse trabalho. A interface de rede foi utilizada para programar um pequeno servidor capaz de enviar e receber as informações do ou para o *front-end* da aplicação.

A Figura 4 mostra um visão geral interna do protótipo utilizado no trabalho. A montagem do hardware consistiu-se em conectar todas as partes modulares a placa Arduino, ligar os sensores e relé nos respectivos pinos de entrada e saída, realizar a soldagem de alguns componentes eletrônicos como relés, resistores, leds e etc. Todos os sensores e também o módulo de rede possuem biblioteca que facilita a programação do Arduino [ARDUINO, 2013].



*Figura 4: Visão interna no protótipo utilizado no Assist Automation, AA!;*

### 3.3 Testes de carga do hardware de automação

Tão importante quanto possuir um ambiente de testes é saber como testar o sistema. As técnicas de testes são executadas de forma exaustiva para garantir o funcionamento correto em ambiente de produção [ALMEIRA, 2010]. Os testes foram baseados em dois tipos. Primeiro é o teste de funcionalidade, onde é executado pela técnica de “caixa preta”, nesse teste o objetivo é garantir que o sistema atenda os requisitos. O segundo teste, será através de pequenos programas através de “caixa branca”, que são responsáveis por simular grandes cargas de requisição para determinadas partes do sistema. Para executar os testes de conexões simultâneas, foi utilizado um simples script em programação bash do Linux que executa



requisições em paralelo, capaz de simular inúmeras conexões simultâneas. Com essa simulação foi possível determinar com precisão o número máximo de requisições simultâneas suportadas pelo Arduino.

O número máximo de conexões sem afetar o desempenho foi de quatro clientes conectados. Identificado esse gargalo, reafirmou-se que a leitura em tempo real dos dados do hardware não permite ter escalabilidade, e a estratégia inicial de buscar as informações do banco de dados após seu carregamento continua sendo a mais viável.

### 3.4 Camada de “front-end” com Adianti Framework

O Adianti Framework PHP é utilizado para o desenvolvimento web e desktop, utiliza a arquitetura MVC que tem o objetivo de construir uma aplicação de forma segura, robusta, rápida e padronizada. Para a construção da aplicação de demonstração, foi utilizado para o *front-end* o template *AdminLTE* integrado com *Bootstrap* e fontes *Awesome*. Para comunicação com o banco de dados, o Adianti Framework utiliza uma implementação simples do Active Records que trabalha com a persistência de objetos no banco de dados. Atualmente o Adianti Framework está na versão 3.0 e possui uma ótima documentação e livros publicados [ADIANTI, 2016].

Para uma solução completa de automação foi necessário definir algumas funcionalidades de cada usuário do sistema. A Figura 5 apresenta o diagrama que mostra de forma macro o que cada ator do sistema pode executar. Conforme a notação UML [VARGAS, 2010], conhecido como Diagrama de Casos de Uso.

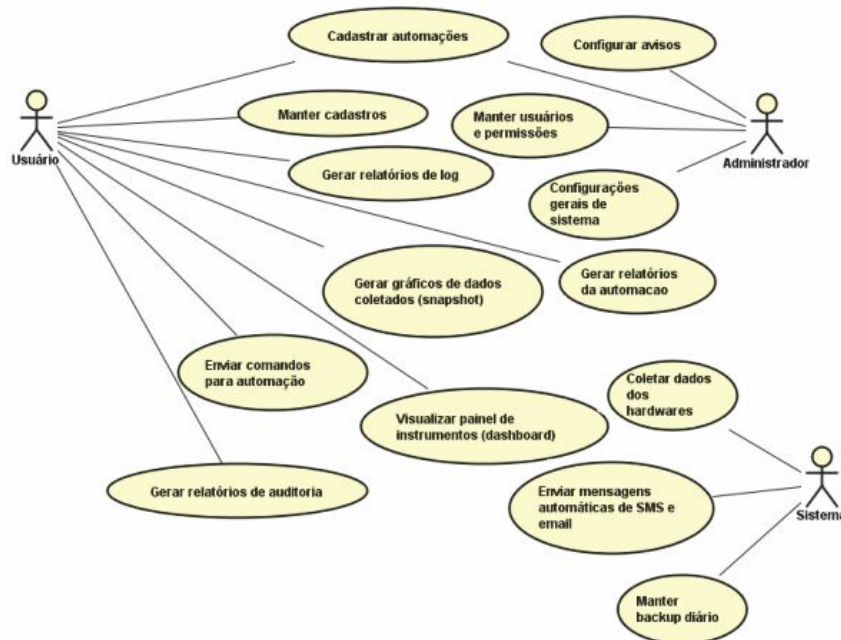


Figura 5. Diagrama de casos de uso e funções por tipo de usuário.

### 3.5 Testes do protótipo em um caso real

O protótipo foi testado em um caso real, com o hardware automatizando uma fonte de jardim. O hardware foi capaz de controlar uma câmera, permitindo a exibição de imagens à distância, ligar luz infravermelha para visão noturna, operação de ligar e desligar o motor que alimenta a cascata da piscina e também a coleta da temperatura da água, ar e umidade relativa do ar.

A Figura 6 apresenta a interface de monitoramento que é acessada pelo navegador, nela é possível visualizar a câmera de monitoramento do jardim, transmitindo em tempo real as informações de temperatura da água, temperatura do ar e umidade do ar. Por meio dos botões de controle, é possível rotacionar a câmera, acionar luz noturna e desligar a mesma, ligar o motor da cascata e filtro UV dentre outras funções disponíveis no painel.

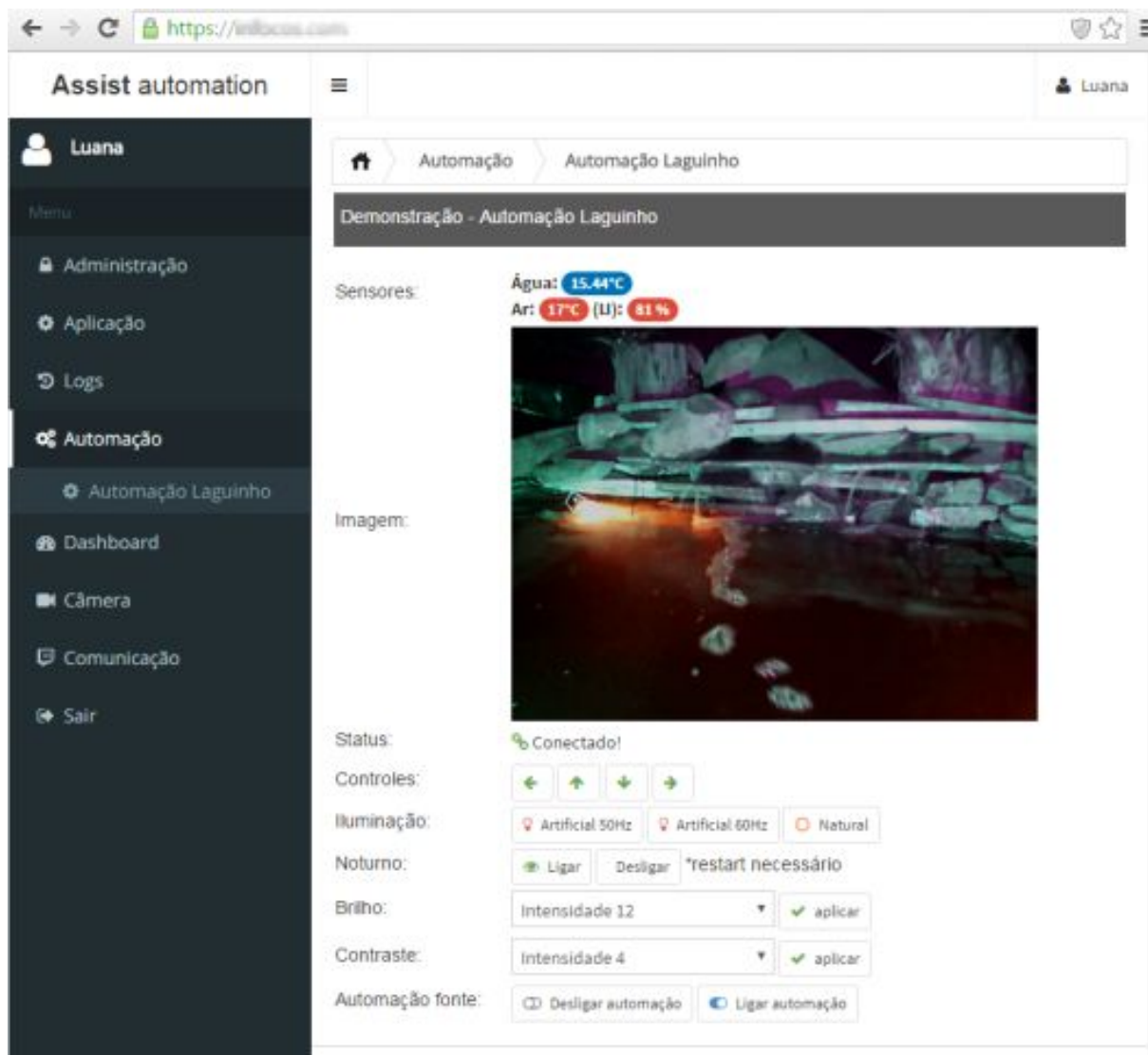


Figura 6: Tela do aplicativo construído para o Assist Automation - AA.

A Figura 6, representa a tela principal do aplicativo Assist Automation, que faz o envio e recebimento de informações para o hardware.

O hardware baseado na plataforma Arduino, operou sobre condições adversas, com temperaturas entre 2 a 35 graus, sob sol e chuva. Durante aproximadamente três meses, realizou a leitura de 213.461 mil registros de umidade do ar e temperaturas do ar e da água.

O dados foram coletados pelo *webservice*, que de forma programada, executava uma leitura a cada 60 segundos. O protótipo ficou instalado em uma fonte de jardim, e para enviar ou receber dados, o *webservice* fazia uma requisição HTTP, passando por um roteador conectado à Internet, que redirecionava as requisições até o Arduino conectado via comunicação sem fio (wi-fi) no centro do gramado.

A partir do momento que o *webservice* alcançava o hardware, ele carregava, transformava e armazenava as informações no banco de dados, para que fossem disponíveis via Internet, acessadas via *smartphone* ou computador pela aplicação do usuário.

As leituras de temperaturas e umidade eram feitas por sensores na água [DS18B20], e no ar [DHT22/DHT11] e requisitadas via HTTP. Da mesma forma, o acionamento de dispositivos remotos eram feitos por HTTP, a requisição aciona ou desliga uma porta lógica do Arduino que por sua vez ligava ou desligava um relé, acionando o motor da fonte, iluminação e filtro de água UV.

Para garantir a eliminação de bactérias pelo filtro UV e oxigenação da água constante, foi implementado também uma função de acionamento da automação de forma agendada a cada 30 minutos, esse agendamento foi programado na camada do hardware e não depende de conectividade da Internet, reduzindo o risco de falha.

#### **4. Conclusão**

O conceito de conectar qualquer objeto a rede começou a ser discutida em 1991 por Bill Joy, co fundador da Sun Microsystem, em 1999 Kelvin Ashton do MIT propôs o termo “Internet das Coisas”, ou simplesmente “IoT”, mesmo assim, se voltarmos a uma década atrás, com a afirmação de que poderíamos controlar qualquer coisa à distância, poderia soar como ficção científica.

O trabalho Assist Automation - AA provou que na atualidade é possível aplicar esse conceito para qualquer objeto, no caso deste trabalho, automatizando uma fonte de jardim com diversos sensores.

A construção do aplicação do usuário (*front-end*) e camada de carregamento de dados, usando a técnica de “*webservices*”, permite que o AA comunique-se com uma infinita combinação de diferentes hardwares remotos, e permite ter a capacidade e escalabilidade para controlar inúmeros objetos de qualquer lugar.

Através dos resultados obtidos com Assist Automation - AA, também conclui-se que

é possível desenvolver um produto final de automação capaz de automatizar desde funções básicas, até recursos avançados da automação, como automações do segmento de indústrias.

### **Agradecimentos**

Este trabalho não seria possível sem a colaboração, ideias, estímulos e empenho de diversas pessoas. Dentre a família, amigos, os colegas de grupo, a Universidade UNIVATES, e em especial ao Professor Fabrício Pretto, que acreditou no trabalho e conduziu as idéias e conceitos que foram amplamente aplicados.

### **Referências**

ABREU, 2008. Desmistificando o conceito de ETL. Disponível em:

<[http://www.fsma.edu.br/si/Artigos/V2\\_Artigo1.pdf](http://www.fsma.edu.br/si/Artigos/V2_Artigo1.pdf)>

ADIANTI, 2016 Adianti Framework para PHP. Disponível em:

<[http://www.adianti.com.br/frame\\_mostra.pdf](http://www.adianti.com.br/frame_mostra.pdf)>

ALMEIRA, 2010 Qualidade dos testes. Disponível em:"

<<http://www.linhadecodigo.com.br/artigo/2775/introducao-ao-teste-de-software.aspx>>

ARDUINO, 2013 Introdução ao kit de desenvolvimento Arduino. Disponível em:

[http://www.telecom.uff.br/pet/petws/downloads/tutoriais/arduino/Tut\\_Arduino.pdf](http://www.telecom.uff.br/pet/petws/downloads/tutoriais/arduino/Tut_Arduino.pdf)

CERT.BR, 2015 Cert.br, Cartilha de segurança para Internet. Disponível em:

<<http://cartilha.cert.br/criptografia/>>

CISCO, 2011 A Internet da Coisas. Disponível em: Como a próxima geração da Internet está mudando tudo. Disponível em:

<[http://www.cisco.com/web/BR/assets/executives/pdf/internet\\_of\\_things\\_iot\\_ibsg\\_0411final.pdf](http://www.cisco.com/web/BR/assets/executives/pdf/internet_of_things_iot_ibsg_0411final.pdf)>

CURBERA, 2013 DUFTLER Matthew, KHALAF Rania. Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI. Disponível em:

<<http://www.immagic.com/eLibrary/ARCHIVES/SUPRSEDED/W3C/W000520N.pdf>>

ETL, 2008 Desmistificando o conceito de ETL. Disponível em:

<[http://www.fsma.edu.br/si/Artigos/V2\\_Artigo1.pdf](http://www.fsma.edu.br/si/Artigos/V2_Artigo1.pdf)>

KIMBALL, Ralp 1998 Data Warehouse Toolkit. Cassiolato, edição 1998

LOON, 2015 Project Loon Google. Disponível em:

<<https://www.google.com/intl/pt-BR/loon/where/>>

MIS, 2014 Measuring the Information Society Report. Disponível em:

<[http://www.itu.int/en/ITU-D/Statistics/Documents/publications/mis2014/MIS2014\\_without\\_Annex\\_4.pdf](http://www.itu.int/en/ITU-D/Statistics/Documents/publications/mis2014/MIS2014_without_Annex_4.pdf)>

ORACLE, 2009 Identificação rápida de gargalos, uma forma eficiente de realizar testes de carga. Disponível em:

<<http://www.oracle.com/technetwork/pt/oem/grid-control/documentation/identificacao-rapida-de-gargalos-432312-ptb.pdf>>

RODRIGUES, 2013 Tecnologia do futuro, Internet das coisas, microcontroladores e webservices. Disponível em:

<<https://www.imed.edu.br/Uploads/90036a9a-cc09-4b96-8c2e-74912cd41d59.pdf>>

WEBSREST, 2011] Webservices WS - versus Web Services REST. Disponível em:

<<http://www.seer.ufrgs.br/reic/article/viewFile/22140/12928>>

VARGAS, 2010 A história de UML e seus diagramas. Disponível em:

<[https://projetos.inf.ufsc.br/arquivos\\_projetos/projeto\\_721/artigo.tcc.pdf](https://projetos.inf.ufsc.br/arquivos_projetos/projeto_721/artigo.tcc.pdf)>