

Plataforma de hardware e software para aquisição e disponibilização de dados utilizando computação em nuvem

Mauricio Alex Zientarski Karrei¹, Willingthon Pavan¹, José Maurício Cunha Fernandes¹, Renato Weiller Dallagasperina¹

¹ Programa de Pós-Graduação em Computação Aplicada, Instituto de Ciências Exatas e Geociências, Universidade de Passo Fundo, Passo Fundo, Rio Grande do Sul, Brasil,
122962@upf.br, pavan@upf.br, jmauricio@upf.br, 71429@upf.br

RESUMO

Este artigo apresenta um estudo e implementação de um sistema de coleta e armazenamento de dados providos de sensores. Uma solução integrada de hardware e software é abordada, discutindo metodologias e técnicas de aquisição e armazenamento centralizado dos dados capturados, além de oferecer uma interface para acesso do usuário. Diante disso, foi desenvolvido um sistema de captura e transmissão de dados que permite fácil adaptação com diferentes tipos de sensores e possibilita a cobertura de grandes áreas físicas. Ainda, são abordadas e aplicadas técnicas de armazenamento de dados em nuvem, a qual oferece maior facilidade na gestão dos dados, além de permitir a integração com outros sistemas, tais como modelos de simulação de culturas.

PALAVRAS-CHAVE: Sensoriamento Remoto, Baixo Custo, Coleta de Dados.

ABSTRACT

This paper presents a study and implementation of a system to collection and storage from sensors. An integrated solution of hardware and software is shown, discussing methodologies and techniques of acquisition and centralized storage of the captured data, besides to provide an interface for user access. Therefore, it developed a capture system and data transmission which allows easier adaptation to different types of sensors and enables the coverage of large physical areas. Still, is dissussed and implemented techniques for cloud computing, which offers greater ease in data management and enables integration with other systems as crop simulation models.

KEYWORDS: Remote sensing, Low cost, Data collect.

INTRODUÇÃO

Com o crescimento da conectividade e da necessidade de soluções que facilitem a interconexão de dispositivos, a demanda por soluções que fazem uso de meios de comunicação aumenta significativamente a cada dia, impulsionando o desenvolvimento de novos recursos tecnológicos capazes de atender essas novas demandas do mercado. Estas necessidades podem estar aliadas a várias áreas do conhecimento, incluindo por exemplo, automação industrial, domótica, segurança e coleta de dados.

A técnica de coleta automática de dados climáticos em áreas de cultivo tornou-se cada vez mais importante, servindo como matéria prima necessária para alimentar sistemas e ferramentas de auxílio na identificação de doenças e do desenvolvimento de plantas. Este processo é de fundamental importância para muitas atividades agrícolas, auxiliando na análise e avaliação de riscos e servindo como base para ferramentas de apoio à tomadas de decisões, o que resulta em um aumento de produtividade e redução de custos na produção (LOUREIRO *et al.*, 2003).

Outro desafio comumente encontrado está no processo de acesso aos dados coletados, onde a descentralização das informações pode dificultar o processo de análise e avaliação da área monitorada. Estes são fatores importantes a se considerar pois são capazes de determinar a qualidade da produção, influenciando diretamente sobre o valor agregado.

Desta forma, este trabalho visa apresentar uma solução de baixo custo para coleta, armazenamento e tratamento de dados climáticos. É apresentado o desenvolvimento de uma solução de hardware e de software capaz de realizar a coleta e transmissão de dados obtidos por sensores para um servidor em nuvem. Além disso, é abordada a implementação de um sistema para gerenciamento dos dispositivos, a qual também provê acesso ao usuário por meio de uma interface *Web*, possibilitando a visualização dos dados coletados em tempo real.

Este trabalho está dividido em quatro partes: a primeira seção apresenta uma introdução sobre o problema a ser tratado e sobre o projeto proposto, discutindo seus objetivos e motivações. Na segunda seção são apresentados fundamentos teóricos e sua aplicabilidade neste projeto, com vistas a facilitar a compreensão do trabalho. Na terceira seção, o projeto proposto é descrito juntamente com um exemplo de aplicação do sistema, e por fim, a quarta seção apresenta as conclusões do trabalho.

MATERIAL E MÉTODOS

Nesta seção são discutidos conceitos fundamentais para a plena compreensão do trabalho. Este capítulo apresenta uma breve revisão dos principais conceitos e protocolos de

comunicação *wireless*, bem como a sua utilização. Além disso, são discutidas tecnologias para a construção do software do sistema que contextualizam este trabalho.

Redes de Sensores Sem Fio (RSSF)

Uma Rede de Sensores Sem Fio (RSSF) é um tipo específico de rede *ad-hoc*. Nas RSSF, cada nó é formado por dispositivos que apresentam uma determinada capacidade computacional responsável pelo funcionamento de aplicações e a (re)transmissão de mensagens em toda a rede. Estes nós se comunicam por uma interface sem fio (*ZigBee*, *WiFi*, *BlueTooth*, etc) e possuem autonomia de energia limitada (RSSF, 2014).

Zigbee

Definido como um protocolo com características de uma rede *WPAN* (*Wireless Personal Area Network*), o protocolo *Zigbee* foi desenvolvido com foco principal de utilização em equipamentos eletrônicos, automação, sensoriamento, periféricos, etc. Este protocolo oferece baixa taxa de transmissão de dados aliada a um consumo de energia minimizado, tornando este apropriado para sistemas que requeiram maior autonomia, visto que este é capaz de se manter funcional durante vários meses ou até anos com o uso de baterias (SILVA, 2007).

Este padrão descreve dois tipos de nós (dispositivos) na rede: FFD (*Full Function Device*), o qual possui um conjunto completo dos serviços do protocolo, e outro RFD (*Reduced Function Device*), capaz de se comunicar apenas com dispositivos FFD. Este último oferece maior simplicidade e conseqüentemente menores custos energéticos (GUEDES, 2014).

Digimesh

Digimesh é um protocolo proprietário desenvolvido pela empresa *Digi International*, e faz uso da topologia *mesh*, sendo muito utilizado por sua capacidade de autorrecuperação em ambientes com muitos obstáculos (ROVERI, 2012).

Neste protocolo não há subdivisões de componentes que integram a rede, ou seja, todos os nós da rede são capazes de exercer as mesmas funções. Portanto cada elemento da rede pode receber e/ou transmitir e rotear pacotes da rede. Por estas características, este protocolo foi adotado para a implementação do projeto, visto que grandes áreas físicas podem ser monitoradas com a utilização de apenas uma rede formada com a topologia *mesh*.

Xbee

Dentre os vários modelos de hardware utilizados pelo mercado está o Xbee, atualmente fabricado pela empresa *Digi International*. Estes módulos basicamente são compostos por um microcontrolador (que contém o *firmware* com o protocolo de rede) e um rádio transceptor para a comunicação. Tais módulos diferem-se conforme suas especificações técnicas, como potência de transmissão e capacidade dos microcontroladores (SILVA 2011).

Como meio de transmissão de para o projeto proposto, dados utilizou-se o Xbee S1-PRO com o protocolo *digimesh*, por incluir em sua especificação um alcance de 1600 metros (em área aberta sem barreiras), e possuir capacidade de roteamento e formação de rede de topologia *mesh*, o que permite a cobertura de grandes áreas físicas (DIGI, 2015).

Arduino Uno

Arduino Uno é uma placa de desenvolvimento microcontrolada baseada no microcontrolador ATmega328 da empresa Atmel Corporation e possui 14 entradas e saídas digitais, 6 entradas analógicas, conexão USB e conexão serial. Esta placa é uma alternativa para o interfaceamento de diversos tipos de sensores com um módulo de transmissão sem fio, visto que suas entradas analógicas e digitais permitem a captura das informações dos sensores, onde estes dados podem ser processados e tratados, além de possuir comunicação serial, permitindo a comunicação com um dispositivo de transmissão sem fio, ou computador (ARDUINO, 2013).

Por possuir tais características, esta placa foi escolhida para a leitura de dados de sensores e o interfaceamento com o dispositivo de transmissão Xbee, por meio do canal de comunicação serial, disponível no Arduino.

Sensores

Sensores são dispositivos que respondem a estímulos ambientais de maneira específica. Estes dispositivos podem transformar grandezas físicas e/ou químicas em grandezas elétricas que podem ser representadas de forma analógica ou digital. Grandezas analógicas possuem valores para todo instante de tempo de maneira contínua, e para o tratamento e armazenamento destas grandezas é necessária sua conversão para o formato digital. Esta conversão pode ser realizada, utilizando-se de conversores Analógico/Digitais (A/D). As grandezas digitais são grandezas discretas, ou seja, não possuem valores para todo instante de tempo e podem ser tratadas e armazenadas sem a necessidade de nenhuma conversão (STEFFENS, 2014).

Neste trabalho utilizou-se o dispositivo DHT22, que é um sensor digital capacitivo de umidade relativa do ar e também possui um termistor que atua como sensor de temperatura.

Além disso, este dispositivo possui características de compensação de temperatura, estabilidade, baixo consumo de energia, sem necessitar de nenhum componente externo (SPARKFUN, 2014).

Node JS

Node JS é uma tecnologia ideal para a criação de aplicações em rede escaláveis (QUIXABEIRA; ALMEIDA; FILHO, 2013). Esta plataforma, desenvolvida na linguagem C++, permite executar um código fonte JavaScript em aplicações executadas pelo servidor, de forma a gerenciar a alocação de memória para objetos, limpando-os quando não são mais necessários (GOOGLE, 2014).

O desenvolvimento da aplicação baseou-se na utilização da plataforma Node JS com vistas a manter uma boa performance da aplicação. A exploração de recursos de I/O não bloqueante (sem bloquear o fluxo de execução normal do programa), permite facilitar a gerência dos sinais oriundos da comunicação dos módulos sem fios, por meio de chamadas de funções *callback* (SCHROEDER; SANTOS, 2014). Outro aspecto importante, para a escolha desta tecnologia, se dá na grande quantidade de pacotes disponíveis para integração de outros serviços, como por exemplo: sistema gerenciador de banco de dados Mysql e o *framework* Express (BONFIM; LIANG, 2014), onde este último foi utilizado como base para o desenvolvimento da aplicação, por oferecer um conjunto de recursos para o desenvolvimento *Web*, com a adoção do padrão MVC (*Model-View-Controller*).

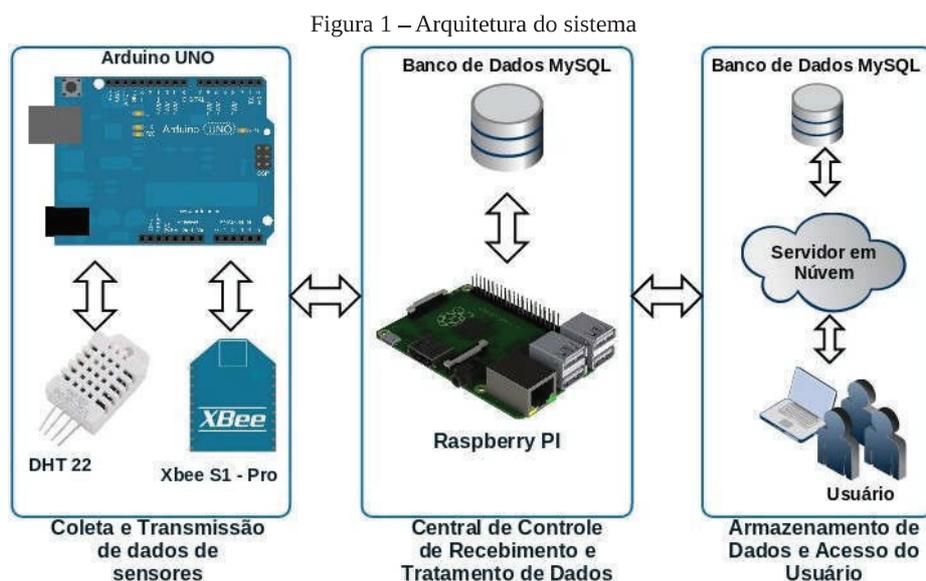
Tecnologias de Criação de Interface

Para o desenvolvimento da interface responsável por apresentar os dados ao usuário, utilizou-se recursos do *framework* Bootstrap, o qual oferece recursos para criação de *layouts* responsivos por meio de códigos JavaScript, CSS e HTML (W3SCHOOLS, 2014). Além disso, foram utilizados recursos da API Google Maps V3 para a integração e apresentação dos dados ao usuário. Esta API consiste de basicamente de recursos em linguagem JavaScript, dos quais podem ser utilizados recursos para definir o comportamento do mapa e de suas extensões (SVENNERBERG et al., 2010).

RESULTADOS E DISCUSSÃO

Nesta seção são descritas as estratégias utilizadas para atingir os objetivos, além de fornecer detalhes sobre o funcionamento e a operação do sistema. O sistema foi dividido em três partes: 1) rede de comunicação e sistema de captura de dados de sensores, 2) sistema de recebimento e tratamento dos dados e 3) servidor de armazenamento dos dados e acesso do usuário.

A Figura 1 apresenta a arquitetura do sistema, exibindo a estrutura de captura e transmissão dos dados de sensores para a central de recebimento e tratamento dos dados (Central de Controle), que tem por incumbência verificar a consistência das informações recebidas e as transmitir para o servidor em nuvem, que por sua vez é responsável por guardar as informações em uma base de dados, e prover o acesso do usuário ao sistema.



Fonte: (autor, 2015)

Central de Controle

Para o recebimento dos sinais e interfaceamento com o nodo receptor dos dados, utilizou-se uma Raspberry PI, a qual é um mini computador de propósito geral que utiliza sistema operacional Linux e oferece recursos de I/O para interfaceamento com outros dispositivos, além de permitir o vínculo com sensores (UPTON; HALFACREE, 2012). Este hardware é portanto o responsável por armazenar e transmitir os dados oriundos dos módulos coletores de dados para um servidor em nuvem por meio de uma conexão com a internet. No caso de ausência desta conexão, os dados permanecem armazenados em uma base de dados MySQL instalada na Raspberry PI até que uma conexão seja estabelecida.

Utilizou-se também, da plataforma Node JS para a construção de um software capaz de efetuar a leitura de dados oriundos de uma comunicação serial, a qual o Xbee receptor e a Raspberry PI são conectados. Além disso, este software é responsável por realizar a validação dos dados (testar se as informações recebidas são válidas) e enviá-las ao servidor em nuvem ou armazená-las na base de dados local (em caso de ausência de conexão com a internet).

Portanto, este serviço permite gerenciar e sincronizar as informações recebidas por meio da comunicação oriunda dos nodos sensores com o servidor em nuvem.

Servidor em nuvem

No que se refere ao controle dos sistemas de coleta instalados e para o recebimento dos dados pelo servidor em nuvem, foi desenvolvida uma plataforma *Web* que permite a realização do cadastro de novas centrais de controle (Raspberry PI), bem como a visualização dos dados coletados e transmitidos pelos sensores vinculados às centrais de controle. Dessa forma, é possível realizar a extração de informações sobre a área monitorada.

Para o cadastro de novas centrais de controle, são inseridas informações como email, senha (utilizados para sincronização com o servidor em nuvem) e as coordenadas geográficas do local em que o sistema deve operar, onde este último valor é utilizado como referência para o cadastro da localização geográfica de novos nodos sensores no sistema (facilitar a localização da área na qual os sensores serão instalados).

Para permitir que o sistema de coleta de dados sincronize seus dados com o servidor, utilizou-se a técnica da sincronização por meio da autenticação de usuário e senha. Para tal, o sistema executado pela central de recebimento dos dados dos sensores deve ser previamente sincronizada com o servidor utilizando-se dos dados de cadastro (email e senha), os quais foram cadastrados no servidor.

O acesso a interface é disponibilizada por meio de um endereço da *Web*, que quando acessado realiza uma requisição ao servidor solicitando como resposta a coordenada geográfica da área em que a central de controle requisitada foi cadastrada, apresentando-a em um mapa (construído com a API Google Maps). Neste instante, também, são retornados (caso houver) as informações dos nodos coletores instalados na área, bem como seus dados já coletados, possibilitando dessa forma, o acesso do usuário a um histórico dos dados coletados.

Aplicação da solução

Para os testes e implantação da solução, aplicou-se o sistema em um ambiente de coleta de dados climáticos, o qual foi responsável por capturar as variáveis de temperatura e umidade relativa com o uso do dispositivo DHT22.

Para tanto, o sistema desenvolvido foi responsável por realizar uma leitura periódica dos dados sensores e transmiti-los até a central de controle. Para a realização da leitura dos dados dos sensores, estes foram conectados a uma porta digital do Arduino, o qual foi responsável por realizar uma escrita em um canal de comunicação serial, onde o Xbee, por sua vez,

transmite o valor coletado. Para a energização dos módulos utilizou-se fontes eletrônicas com tensão de 9 Volts e corrente elétrica de 1 Ampere, conectadas diretamente a energia elétrica.

O sistema foi projetado para capturar e transmitir o valor de sensores dos mais variados tipos. Para isso, a utilização da placa Arduino permitiu maior controle sobre a emissão dos dados, além de permitir a flexibilidade da captura de dados de sensores, visto que é possível adaptar facilmente o sistema para leitura de diferentes tipos de sensores, ou até mesmo acionar dispositivos. Um exemplo deste última aplicabilidade, pode ser considerado o controle de operação de um sistema de irrigação (ligar/desligar).

A Figura 2 apresenta os pontos de instalação dos módulos, sendo que a cor amarela, simboliza o estado operante do módulo (identificado por meio de envios temporários), e a cor vermelha indica uma possível falha no sistema (nodo não está enviando nenhum sinal à Central de Controle). Além disso, são exibidas informações dos dados coletados pelo coletor 01, juntamente com a data e horário da última coleta.

Figura 2 – Ambiente de aplicação e dados coletados



Fonte: (autor, 2015)

Cada módulo é composto por um identificador único do equipamento, identificador da área instalada e um tipo (sensor de umidade, sensor de temperatura, etc), permitindo estender

as funcionalidades, adequando-se facilmente em diferentes aplicações. Além disso, foi implementado o controle de envios periódicos de sinalização, onde este sinal indica se o mesmo está funcionando corretamente. Para isso, são transmitidas informações como o tipo de dado que está sendo transmitido (sinalização de operação do módulo ou valor da leitura de um sensor). A Figura 3 apresenta o código construído na linguagem C/C++ o qual constitui a função executada pelo Arduino para envio dos dados à central de controle.

Figura 3 – Pacote de Transmissão dos dados

```
void sendDataSensor(int event) {  
    uint8_t payload[] = { ID_DEVICE, ID_AREA, TP_DISPOSITIVO, event };  
  
    DMtxRequest tx = DMtxRequest(addrController, payload, sizeof(payload));  
  
    xbee.send(tx);  
}
```

Fonte: (autor, 2015)

CONCLUSÕES

Este trabalho abordou o desenvolvimento de uma solução de hardware e software para atender as necessidades de coleta de dados, além de oferecer uma solução de controle e gerenciamento da área monitorada, assim como dos dispositivos que compõe o sistema. É um esforço no sentido de fornecer uma solução para gerenciamento e transmissão dos mais variados tipos de dados, possibilitando a fácil adaptação para diversas aplicações. Além disso, esta solução pode ser utilizada como alternativa de baixo custo, dentre diferentes ferramentas de coleta de dados existente no mercado atual, como por exemplo, estações meteorológicas.

Dessa forma, o sistema aqui apresentado, mostrou-se apto a ser utilizado em grandes áreas, visto que este oferece características de roteamento utilizando a topologia *mesh*, por meio do protocolo *digimesh*, o qual também pode ser uma alternativa adequada ao o protocolo *Zigbee*, comumente utilizado em aplicações de RSSF. Além disso, os resultados obtidos com os testes e a aplicação do sistema atende os objetivos estabelecidos, permitindo realizar a coleta e transmissão de dados de sensores e o acompanhamento dos dados em tempo real.

REFERÊNCIAS

ARDUINO. Disponível em: <<http://arduino.cc/en/Main/arduinoBoardUno>>. Acesso em novembro de 2014, v. 15, n. 03, 2013.

BONFIM, F. L.; LIANG, M. Aplicações Escaláveis com Mean Stack. Monografia (Graduação em Ciência da Computação) – Ciências Exatas, Universidade Federal do Paraná, Paraná, 2014.

DIGI, International. Xbee S1-PRO. Disponível em <<http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/point-multipoint-rfmodules/xbee-series1-module>>. Acesso em novembro de 2014.

GOOGLE. V8 engine. Disponível em <<https://developers.google.com/v8/>> Acesso em novembro de 2014.

GUEDES, A. L. Uma plataforma de hardware e software para desenvolvimento de rede de sensores sem fio aplicada a agrometeorologia. 2013. 91 f. Dissertação (Programa de Pós-Graduação em Engenharia Elétrica) – Universidade Estadual de Londrina, Londrina, 2013.

LOUREIRO, A.F.; et al. Redes de sensores sem fio. In: XXI Simpósio Brasileiro de Redes de Computadores (SBRC). 2003. p. 179-226.

QUIXABEIRA, R. F.; ALMEIDA, WILL, R. M.; FILHO, P. M. A. Utilização da tecnologia Node JS para consumo eficiente sobre recursos de servidores web. Conselho Editorial, p. 100, 2013.

ROVERI, M. R. Automação Residencial. 2012. 87 f. Monografia (Graduação em Tecnólogo em Redes de Computadores) – Faculdade Politec, Santa Bárbara d'Oeste, 2012.

RSSF. Rede de Sensores sem Fio. Disponível em <http://www.gta.ufrj.br/grad/12_1/rssf/def.html>. Acesso em novembro de 2014.

SCHROEDER, R.; SANTOS, F. Arquitetura e Testes de Serviços Web de alto desempenho com NODE.JS e MongoDB. 2014.

SILVA, A. T. Módulos de comunicação wireless para sensores. 2007. 84f. Monografia (Curso da Licenciatura em Engenharia Electrotécnica e de Computadores) – Faculdade de Engenharia, Universidade do Porto, Porto, 2007.

SILVA, L. H. S. Desenvolvimento de uma rede de sensores sem fio utilizando zigbee para aplicações diversas. 2011. 96f. Monografia (Graduação em Engenharia da Computação) - Escola Politécnica de Pernambuco, Universidade de Pernambuco, Pernambuco, 2011.

SPARKFUN, DHT22. Disponível em <<https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>>. Acesso em novembro de 2014.

STEFFENS, C. A. O Funcionamento de alguns Sensores. Disponível em <<http://www.if.ufrgs.br/mpef/mef004/20061/Cesar/SENSORES-Definicao.html>>. Acesso em novembro de 2014.

SVENNERBERG, G. et al. Beginning Google Maps API 3. 2010, 328p.

UPTON, E.; HALFACREE, G.. Meet the Raspberry Pi., 2012, 420p.

W3SCHOOLS. Bootstrap. Disponível em <<http://www.w3schools.com/bootstrap/>>. Acesso em novembro de 2014.