

Smart irrigation – uma proposta para o uso eficiente da água em hortas escolares

Leandro da Silva Camargo¹, Marcel da Silva Camargo¹, Sandro da Silva Camargo¹

¹ Programa de Pós-graduação em Computação Aplicada, Universidade Federal do Pampa e Embrapa Pecuária Sul, Bagé, Rio Grande do Sul, Brasil.
leandrocamargo@ifsul.edu.br, marcel@sigasistemas.com.br,
sandro.camargo@unipampa.edu.br

Resumo. O presente projeto propõe uma rede de sensores eletrônicos de baixo custo, *open source*, baseados na tecnologia Arduino, desenvolvendo um sistema de coleta, armazenamento e análise de indicadores em tempo real para monitorar os parâmetros de temperatura, umidade, luminosidade e umidade do solo, além de controlar a aplicação de insumos em hortas cultivadas nas escolas municipais de Bagé-RS, tendo como intuito a melhoria no rendimento das culturas, bem como, o uso consciente dos recursos naturais, o principal deles, a água, a qual está em falta na região. Os dados coletados serão mantidos em um servidor Raspberry com banco de dados Mongo DB e, os gráficos gerados dinamicamente utilizando Chart.JS. A criação deste sistema inteligente pode aproximar ainda mais os estudantes do contato com a natureza, além de servir como ferramenta didática, na qual o docente pode apresentar o resultado de cada interação observada em gráficos interativos.

1 Introdução

Em maio de 2015, algumas escolas municipais de Bagé foram contempladas com o Projeto Horta Escolar, tendo como finalidade a de intervir na cultura alimentar e nutricional dos educandos na faixa etária de 7 a 14 anos, promovendo desta forma, a educação integral de crianças e jovens de escolas e comunidades do seu entorno. Provendo por meio das hortas escolares uma alimentação nutritiva, saudável e ambientalmente sustentável, além de apresentar essa temática de forma transversal na prática pedagógica destes educandários. Este projeto foi executado com recursos do Fundo Nacional de Desenvolvimento da Educação (FNDE/MEC), em parceria com o Fundo das Nações Unidas para a Agricultura e Nutrição (FAO). Contudo, o município de Bagé sofre com a irregularidade na distribuição das chuvas, a baixa capacidade dos reservatórios naturais e na disponibilidade de água subterrânea. Neste viés, o uso de sensores para monitorar e garantir o fornecimento de água imediata para a planta, sempre que necessário, dependendo dos valores de umidade a aplicação adequada de insumos pode garantir a manutenção deste importante projeto.

Um sistema de monitoramento dos parâmetros ambientais, como a temperatura, pressão e a umidade do ar e do solo, pode apoiar a produtividade e a qualidade de alimentos gerados nas hortas escolares. Além disso, o sistema de monitoramento em tempo real pode ser aplicado à agricultura em uma escala mais ampla. Este estudo pretende construir uma rede de sensores que coletam os dados do ambiente, com suporte de uma camada de persistência destes valores coletados em um banco de dados e, ainda, o desenvolvimento de um aplicativo para dispositivos móveis que permita o tratamento, visualização e a análise destas coletas.

O sistema disponibilizará os valores obtidos pelos sensores, em tempo real, das condições ambientais apoiando a prática de educação ambiental dos alunos das séries iniciais, onde os personagens poderão acompanhar as reações de cada interação com o cenário monitorado.

A escolha da aplicação deste trabalho em escolas municipais é reforçada pela iniciativa similar adotada pelos Países Baixos, conforme relatado por [1] onde todos os anos 6000 a 7000 crianças com idades entre os 9 e os 10 anos frequentam um programa numa horta escolar perto da sua escola. O programa foi desenvolvido pelo Centro de Natureza e Educação Ambiental de Amsterdã (ANMEC) como parte de uma iniciativa para tornar as aulas de Natureza e Meio Ambiente mais práticas. Conforme [2] no verão, no início dos cursos da escola, as crianças recebem aulas teóricas onde aprendem os fundamentos da jardinagem. No final do verão, eles começam a frequentar os jardins da escola por uma hora e meia a cada semana. Cada criança recebe um lote de terra com plantas pré-cultivadas para cuidar. No final do ano letivo, as crianças colhem suas plantas e, em algumas hortas escolares, conseguem cozinhar pratos com suas frutas e verduras colhidas. As crianças têm um objetivo ao jardinar: garantir que suas plantas cresçam bem e deem frutos. Para realizar essa tarefa, eles precisam saber como cuidar adequadamente de suas plantas. Ao fornecer feedback à criança durante o processo de crescimento da planta, a criança tem mais informações sobre seu comportamento e sabe o que ele tem a mudar sobre suas ações para cuidar melhor dela [1].

Com relação a definição do banco de dados utilizado no trabalho, pesquisas anteriores mostraram que o banco de dados *Not Only SQL* (NoSQL) é capaz de processar uma entrada e saída maciça de dados não estruturados de forma eficiente, além da facilidade em adicionar servidores à medida que o número de sensores ou clientes aumenta [2]. Além disso, devido à natureza imperfeita do dispositivo de detecção e à fragilidade da rede, alguns dos dados coletados podem estar incompletos, criando dados de detecção de valores aberrantes [3]. Ou seja, um banco de dados convencional, estruturado, pode não suportar essa variabilidade dos parâmetros coletados pelos sensores.

Também é finalidade deste sistema, o de prover a irrigação em volumes e intervalos ajustados às condições do ambiente. Tornando o sistema mais eficiente e evitando o desperdício de água. O projeto realizado por [4] apresenta resultados com relação à umidade do solo após aplicar uma abordagem de Estratégia Direcionada por Prioridade, a qual reduziu o desperdício desnecessário da água e garantiu a umidade adequada do solo conforme a cultura.

O trabalho realizado por [5] propôs a instalação de uma rede de sensores para a detecção da umidade do solo e a captura dos valores destes sensores, dentre esses valores, o sistema detecta onde a umidade é menor que o limite definido. Visando a utilização otimizada da água pelo sistema, frente a abordagem manual, pois se um agricultor estiver irrigando toda uma área cultivada na fazenda duas vezes por dia, então definitivamente haverá um grande desperdício de água. O que foi sugerido neste trabalho é que através do sistema de irrigação automatizada é possível controlar o uso desnecessário de recursos naturais.

Em [6], o sistema possui uma rede sem fio distribuída de sensores de umidade e temperatura do solo, colocados na zona radicular das plantas. A unidade de gateway lida com as informações do sensor, aciona os atuadores e transmite dados para um aplicativo da web. Um algoritmo foi desenvolvido com valores limiares de sensores que foram programados em um gateway baseado em microcontrolador para controlar a quantidade de água.

Uma importante solução de software é apresentada por [7], ao entregar uma experiência de gamificação aos estudantes, pois uma planta física, assim como um brinquedo virtual (Tamagotchi), contém um conjunto de fatores básicos que determinam sua condição de crescimento. Esses fatores são: a quantidade de luz solar que recebe, a umidade do solo e a temperatura do ambiente. Desta forma, a condição da planta pode ser percebida medindo esses três fatores com o uso de sensores adequados. Os sensores estão conectados ao solo ao redor da planta e podem enviar os dados para um aplicativo móvel; o dispositivo enviará os dados para a nuvem e compara com os valores preferidos para o crescimento ideal da planta. Dependendo da intensidade da luz, da quantidade de água disponível, da umidade e temperatura do ambiente, o brinquedo digital exibirá a emoção adequada, reproduzindo assim, a condição da planta.

O trabalho está organizado da seguinte forma, na seção 2 é apresentado um estado da arte referente aos conceitos e ferramentas abordados neste artigo. A seção 3 detalha a arquitetura e modelagem do sistema proposto, custos de implantação, bem como, a análise dos dados capturados e trabalhos correlatos. Por último, na seção 4, as considerações parciais do estudo e das ferramentas desenvolvidas até o momento.

2 Materiais e Métodos

2.1 Internet das Coisas

Hoje, a Internet tornou-se onipresente, atingiu quase todos os cantos do mundo e está afetando a vida humana de maneiras inimagináveis e a conectividade está ainda mais abrangente, pois existe uma grande variedade de aparelhos conectados à web, este fenômeno é chamado de “Internet das Coisas” (abreviada como IoT). Este termo foi definido por diferentes autores de várias maneiras diferentes, contudo, duas das definições mais populares serão apresentadas a seguir. Uma definição referente a Internet das Coisas é apresentada como simplesmente uma interação entre os mundos físico e digital [8]. O mundo digital interage com o mundo físico usando uma infini-

dade de sensores e atuadores. Outra definição trata a Internet de "Coisas" como um paradigma no qual as capacidades de computação e de rede estão embutidas em qualquer tipo de objeto concebível [9]. Esses recursos são utilizados para consultar o estado do objeto e alterar seu estado, se possível. No senso comum, a Internet das Coisas refere-se a um novo tipo de abordagem computacional, onde quase todos os dispositivos e aparelhos estão conectados a uma rede.

O uso de sensores na Internet de Coisas (IoT) aumentou significativamente devido ao baixo custo, tamanho reduzido, consumo eficiente de energia e facilidade de uso. Os sensores IoT normalmente coletam uma quantidade enorme de dados e os envia para um servidor remoto. Os dados coletados pelos sensores devem ser analisados e apresentados para diferentes fins. Várias arquiteturas IoT foram aplicadas em muitas áreas diferentes e demonstraram o potencial do IoT para melhorar a qualidade de vida de nossa sociedade [10]. Estas aplicações IoT requerem um ou mais sensores para coletar dados do meio ambiente.

2.2 Sensores baseados no Arduino

Entre as principais tecnologias voltadas para este contexto de sensores para o monitoramento ambiental, destaca-se a utilização do Arduino. O Arduino é uma plataforma de prototipagem eletrônica *open source*, que permite, por exemplo, sensoriar com baixo custo financeiro características do mundo físico, tais como temperatura, umidade, som e movimento [11]. Deste modo, características como autonomia e proatividade, fazem com que a utilização de agentes inteligentes, se torne uma necessidade ou solução apropriada para o gerenciamento deste tipo de sistemas [12]. Um agente inteligente é uma entidade autônoma que percebe seu ambiente através de sensores e age sobre o mesmo utilizando atuadores [13]. Diante desta perspectiva, o monitoramento dos parâmetros ambientais pode ocorrer com a adoção de um sistema multiagente, conforme modelo adotado neste projeto e apresentado na figura 1.

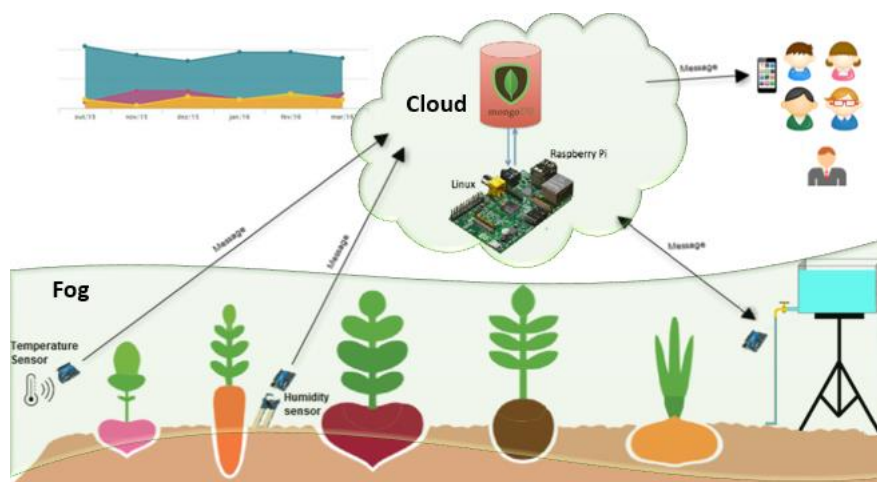


Fig. 1 – Esquema do sistema de monitoramento e controle de hortas escolares.

2.3 Arquitetura Híbrida do Sistema

O sistema proposto possui dois níveis de interação, uma camada de névoa (*fog*), local, tendo o Arduino como um concentrador que recebe os dados de vários sensores. A *Fog Computing* pode ser definido como uma arquitetura de rede, que usa um ou uma multidão de colaboração de usuários finais, tais como celulares ou computadores, ou dispositivos de borda perto do usuário, para entregar uma quantidade substancial de recursos como armazenamento, processamento, comunicação e medidas de controle e gestão de rede [14].

Essa definição difere do conceito de Computação em Nuvem, onde o armazenamento de dados é feito principalmente em data centers na nuvem, onde as conexões e tráfego são encaminhados através de grandes áreas, como as redes de *backbone* para a comunicação, e onde o controle é feito principalmente por gateways de rede [15].

A segunda camada do sistema proposto está na nuvem (*cloud*), demandando um volume menor de interações e recebendo mensagens num fluxo menor. Neste nível o Raspberry assume o papel de servidor de banco de dados. O Raspberry Pi é um computador de placa única do tamanho de um cartão de crédito desenvolvido no Reino Unido pela *Raspberry Pi Foundation* com a intenção de promover o ensino da informática básica nas escolas. O Raspberry Pi em sua terceira versão, selecionada para ser utilizada nesta proposta, possui um processador Broadcom BCM2837 ARM Cortex – A53 de 1.2GHz 64 bits Quad Core, que inclui uma GPU Videocore IV de 400 MHz, e 1 Gigabyte de memória RAM, Wi-Fi 802.11 b/g/n de 2.4GHz e Bluetooth 4.1 (BCM 434380). Ele não inclui um disco rígido interno ou uma unidade de estado sólido, mas usa um cartão SD para inicializar e gravar dados.

A Computação em Nuvem permite várias facilidades, funciona como uma forma centralizada, contudo além do custo de comunicação dos dados também apresenta um custo de latência. Caso o data center não esteja próximo dos utilizadores, aplicações sensíveis a atrasos perdem desempenho no modelo de computação em nuvem. Neste sentido, a Fog estende e complementa a nuvem, funcionando na borda da rede e nos usuários finais.

2.4 Banco de dados NoSQL - MongoDB

O banco de dados SQL tradicional tem várias limitações no processamento de grandes quantidades de dados não estruturados, e para superar este problema, as tecnologias de banco de dados NoSQL foram desenvolvidas. Neste sentido, o MongoDB pode ser uma boa alternativa, pois é um banco de dados NoSQL de código aberto.

O MongoDB armazena seus dados em documentos, estes por sua vez são estruturas de dados complexas emparelhadas com chaves. Eles podem conter pares de valores-chave, pares de chaves-matrizes ou documentos aninhados, ao contrário de bancos de dados relacionais que exigem dados estruturados e possuem tabelas fixas. Os documentos binários JSON (BSON) MongoDB permitem uma estrutura mais complexa ao

suportar matrizes de valores em que cada disposição em si pode ser composta por múltiplos subdocumentos [16, 17].

Vários estudos avaliaram o desempenho do MongoDB. Um destes estudos comparou o desempenho de leitura e escrita dos bancos de dados NoSQL com o PostgreSQL para armazenar e consultar dados de sensores [18]. Os resultados mostraram que o MongoDB é a melhor solução para armazenar dados de sensores de tamanho pequeno ou médio, Casandra é a melhor escolha para dados de sensores grandes, e o PostgreSQL possui melhor capacidade de consulta. Outro estudo comparou a performance de inserção e pesquisa do MongoDB e MySQL em uma única máquina. O resultado mostrou que o MongoDB superou o MySQL em termos de inserção e pesquisa de dados [19]. Recentemente, um repositório baseado em MongoDB para lidar com grande volume de dados RFID e sensor foi desenvolvido [20]. Com base em literaturas anteriores, o MongoDB mostrou menos tempo de resposta nos testes em termos de leitura e gravação em comparação com outros bancos de dados NoSQL. Em síntese, o MongoDB é projetado para suportar as aplicações modernas que possuem grandes volumes de dados que mudam rapidamente, alguns dos quais não são estruturados. Também foi construído com suporte ao armazenamento distribuído.

2.5 Biblioteca gráfica Chart JS

Uma vez que os dados tenham sido coletados e armazenados na plataforma de banco de dados, visualizá-los efetivamente é o próximo passo importante. Os dados podem ser exibidos em formatos diferentes, como XML (*Extensible Markup Language*) e JSON (*JavaScript Object Notation*), e podem ter que ser visualizados usando diferentes tipos de gráficos: pizza, linhas, colunas, etc. Para o sistema proposto neste trabalho foi construído um dashboard (painel de visualização), baseado na linguagem HTML5, e utilizando a biblioteca gráfica para representação dos dados chamada de Chart.JS, por ser de código aberto e suportar mais de 20 tipos diferentes de gráficos [21]. Usando a tag HTML5 Canvas, o Chart.js cria os gráficos no navegador da Web, o que significa páginas mais rápidas e menos carga do servidor Web.

3 Resultados e Discussões

A presente trabalho visa a sustentabilidade das hortas escolares implantadas, conforme [22], o conceito de sustentabilidade baseia-se em três dimensões importantes: são dimensão econômica (relaciona-se com o uso eficiente dos recursos), a dimensão ambiental (garantindo que os recursos naturais estejam disponíveis no futuro) e a dimensão social (saúde humana e bem-estar).

O projeto apresenta uma arquitetura em dois níveis, um que corresponde a nuvem onde o Raspberry PI oferece maior poder computacional e dá suporte à persistência dos dados, realizando a tarefa de concentrador dos parâmetros coletados e servindo como um servidor do banco de dados. Este nível apresenta um custo de montagem de

aproximadamente U\$80 (oitenta dólares americanos) e suporta uma carga de vários sensores comunicando simultaneamente sem degradação da performance, latência ou perda de pacotes. Noutro nível, chamado de *fog*, os sensores Arduino podem trocar informações entre si, bem como, pré-processar alguns dados. Neste nível o custo de montagem é de U\$20 (vinte dólares) dependendo do sensor instalado junto ao Arduino conforme demonstrado na tabela 1.

Tabela 1 – Preço dos componentes de hardware utilizados.

Hardware	Nível	Preço em US\$
Raspberry PI	Cloud	100.00
Arduino Uno R3	Fog	25.00
Sensor de Chuva YL83	Fog	2.00
Gerador Fluxo Água	Fog	20.00
ESP8266	Fog	10.00
Higrometro Soil	Fog	1.00
Sensor Temp. DTH 11	Fog	10.00

O preço total do sistema de monitoramento do ambiente pode variar conforme o número de sensores e as características físicas do espaço onde serão instalados. A figura 2 apresenta um projeto com um conjunto de sensores instalados na placa Arduino para coleta de alguns parâmetros ambientais. Tais dados são transmitidos para o servidor de dados, na *cloud*, através de mensagens de inserção enviadas diretamente no banco de dados MongoDB.

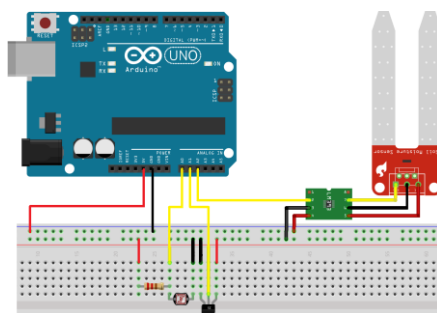


Fig. 2 – Projeto Arduino Uno com sensores de temperatura, intensidade de luz e umidade do solo.

Como é esperado, os sensores serão expostos a uma grande quantidade de condições ambientais (temperatura, umidade, etc.), ainda, um grande volume de inserções na base de dados, neste sentido, o banco de dados NoSQL pode lidar com os dados produzidos pelos sensores IoT. A figura 3 mostra o diagrama de sequência onde fica evidente as interações e trocas de mensagens entre todos os objetos do sistema, tanto no nível do *fog computing* (elementos de cor verde), quanto na *cloud computing* (elementos em azul).

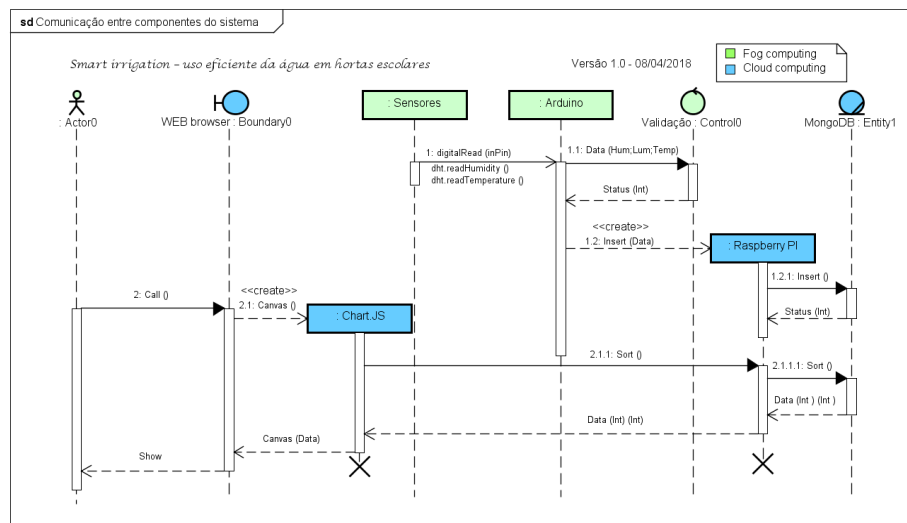


Fig. 3 – Diagrama de sequência com a troca de mensagens entre os objetos e componentes do sistema.

No nível do *fog computing* é realizada uma validação dos parâmetros coletados nos sensores, conforme a definição de intervalo, os dados coletados de umidade, temperatura e luminosidade são transmitidos para o servidor de banco de dados (Raspberry) e o documento armazenado no banco MongoDB juntamente com a data e hora da inserção.

O usuário poderia acessar o endereço do servidor em qualquer dispositivo computacional que possua um navegador. Na figura 4 é exibido o dashboard (painel de visualização) que está disponível para acompanhamento em tempo real dos parâmetros coletados. Para tal, a biblioteca JavaScript necessária para este trabalho deveria suportar representações em gráficos 2D, incluindo gráficos de linhas, gráficos de barras e gráficos de pizza, ser extensível, fácil de usar e também leve, pois o intuito é de oferecer o acesso aos dados pela maior gama de dispositivos computacionais, exigindo apenas um navegador para realizar a visualização dos gráficos. Desta forma, foi definido o Chart.JS, pois esta biblioteca de código aberto usa o elemento de tela HTML5 para plotar gráficos. Oferece seis módulos (bar, pizza, linha, rosca, radar, área polar) que podem ser carregados individualmente para minimizar o tamanho da biblioteca dentro do projeto. Essencialmente, o código carregado é apenas o demandando para cada visualização. Além disso, oferece suporte em todos os navegadores modernos e tem suporte ao mais antigos, como o Microsoft Internet Explorer versão 7 e 8.

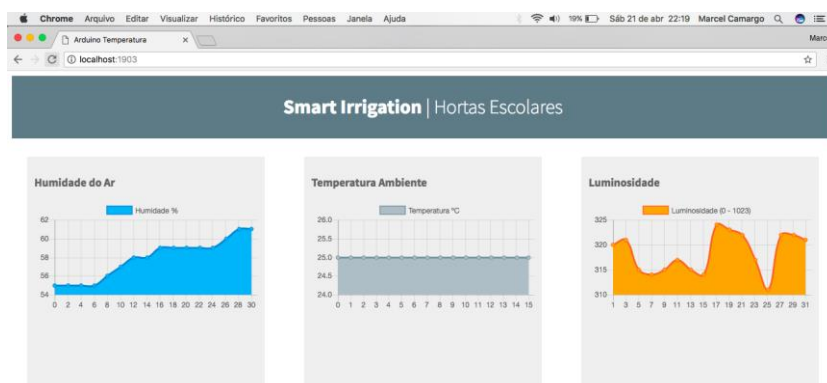


Fig. 4 – Dashboard (painel de visualização) contendo os parâmetros coletados pelos sensores e exibidos em tempo real no navegador.

Para este trabalho, as virtudes em adotar a biblioteca Chart.JS está no tamanho demandado, pois consome apenas 11kb. No dashboard apresentado na figura 4 os professores poderão demonstrar aos estudantes quais reações ocorrem no ambiente conforme sejam realizadas interações, tais como um sombreamento, a irrigação ou as alterações da temperatura no decorrer do dia. Outra característica muito importante que é viabilizada com o uso da biblioteca Chart.JS é demonstrada na figura 5, onde é representada a sua capacidade de ser responsiva, ou seja, os gráficos são redimensionados e visualizados adaptando-se ao tamanho da janela do navegador.



Fig. 5 – Apresentação dos gráficos na tela do smartphone para acompanhamento em tempo real das condições do ambiente.

Contudo, as interações com o ambiente traduzidas em gráficos, mesmo que acompanhadas em tempo real, talvez não sejam tão motivacionais às crianças, mas confor-

me apresentado na figura 6, a proposta realizada por [7] dá um caráter de gamificação e cria um personagem avatar para a planta, algo parecido com o que as crianças estão acostumadas a interagir.



Fig. 6 – Exemplo de gamificação onde o avatar exibe a atitude / humor da planta através de expressões faciais e balões de texto. Adaptado de [7]

Ao utilizar o efeito Tamagotchi como elemento de gamificação, a aplicação digital pode apresentar os dados que são reunidos de uma forma que a criança está acostumada (através de um personagem do jogo), e acha divertido e interessante. Esse processo incentiva a criança a se envolver mais com ela. Como uma planta física também tem necessidades que determinam quão bem ela vai crescer e como essas necessidades podem ser transferidas para o mundo digital, ela pode ser representada em um aplicativo digital como um avatar. Esta alternativa de jogo interativo pode potencializar a adoção deste trabalho pelos estudantes das séries iniciais e efetivamente contribuir com a formação integral das crianças, cumprindo também com seu papel pedagógico.

4 Considerações Parciais

O sistema proposto apresenta algumas virtudes que potencializam sua aplicação como um instrumento de auxílio no uso otimizado da água e no monitoramento da disponibilidade de outros recursos naturais. A solução apresenta um hardware de baixo custo e totalmente *open source*, bem como do lado do software todos os recursos utilizados são de código livre.

Para a análise dos dados o *layout* do *dashboard* apresenta os dados mais significativos para o monitoramento ambiental de forma clara e responsiva, possibilitando uma leitura facilitada dos dados, mesmo em dispositivos móveis com telas reduzidas. A biblioteca que constrói o *dashboard* é componível, carregando apenas os módulos

demandados e, conseqüentemente, exigindo baixo poder computacional no lado do cliente, além da portabilidade, pois é compatível com as versões mais antigas do navegador.

Com relação a aplicação pedagógico do sistema, num primeiro momento a visualização gráfica, em tempo de execução referentes às mudanças ambientais capturadas pelos sensores parecia suficientemente atrativa à atenção das crianças, principalmente quando considerada a formação coletiva de uma turma de estudantes, contudo, após observar as características de outras propostas de software, onde um deles propunha a associação destas leituras com um aspecto de brinquedo virtual, isto potencializa a criação de um vínculo afetivo e motivacional estabelecido entre cada indivíduo criança e o meio ambiente, desta forma, havendo interesse das instituições de ensino em um trabalho com indivíduos, o brinquedo pode tornar-se mais interessante que a visualização das mudanças ambientais meramente através da geração de gráficos.

Referências Bibliográficas

1. HERMSEN, S.; RENES, R. **Ontwerpen Voor Gedragsverandering**. Publication Centre of Exptise Creative Industry Utrecht: Utrecht, Dutch. 2014.
2. VEEN, J.S.; WAAIJ, B.; MEIJER, R.J. **Sensor data storage performance: SQL or NoSQL, Physical or Virtual**. In Proceedings of the IEEE 5th International Conference on Cloud Computing (CLOUD), Honolulu, HI, USA, 24–29. pp. 431–438.2012.
3. ZHUANG, Y.; CHEN, L. **In-network Outlier Cleaning for Data Collection in Sensor Networks**. In Proceedings of the CleanDB, Workshop VLDB, Seoul, Korea, 11 September 2006.
4. MAHATMA, P. K. V. **Soil testing Analysis**. Rahuri: MPKU Rahuri. 2017.
5. SANTOSH D.; RAMESH C. P. **Optimum utilization of natural resources for home garden using wireless sensor networks**. Journal of Information and Optimization Sciences, 38:6, 1077-1085. 2017.
6. BENNIS I.; FOUCHA, H.; ZYTOUNE, O.; ABOUTAJDINE, D. **Drip Irrigation System using Wireless Sensor Networks**. Proceedings of the Federated Conference on Computer Science and Information Systems. ACSIS, Vol. 5. 2015.
7. VALPREDA, F.; ZONDA, I. **Grüt: A Gardening Sensor Kit for Children**. Sensors 2016. 16, 231. 2016.
8. VERMESAN O., FRIESS P., GUILLEMIN P., et al. **Internet of things strategic research roadmap**. in Internet of Things: Global Technological and Societal Trends, vol. 1, pp. 9–52, 2011.
9. PENA-LOPEZ I. **Itu Internet Report 2005: The Internet of Things**. 2005.
10. SETHI, P.; SARANGI, S.R. **Internet of Things: Architectures, Protocols, and Applications**. J. Electr. Comput. Eng. 2017.

11. SUNG, W. T., CHEN, J. H., HSIAO, C. L., and LIN, J. S. **Multi-sensors Data Fusion Based on Arduino Board and XBee Module Technology**. IEE International Symposium Computer. Consumer and Control (IS3C). 2014.
12. JENNINGS, N. R. An Agent-Based approach for building Complex Software Systems. *Communications of the ACM*, 44, pp. 35-39. 2001.
13. JENNINGS, N. R. **An Agent-Based approach for building Complex Software Systems**. *Communications of the ACM*, 44, pp. 35-39. 2001.
14. BONOMI, F. et al. **Fog computing: A platform for internet of things and analytics**. In: BESSIS, N.; DOBRE, C. (Ed.). *Big Data and Internet of Things: A Roadmap for Smart Environments*. Springer International Publishing. (Studies in Computational Intelligence, v. 546). p. 169–186. ISBN 978-3-319-05028-7. Disponível em: <http://dx.doi.org/10.1007/978-3-319-05029-4_7>. 2014.
15. ZHU, J. et al. **Improving web sites performance using edge servers in fog computing architecture**. In: *Service Oriented System Engineering (SOSE)*, 2013 IEEE 7th International Symposium on. [S.l.: s.n.]. p. 320–323. 2013.
16. COPELAND, R. **MongoDB Applied Design Patterns**. O'Reilly Media, Inc.: Sebastopol, CA, USA, 2013.
17. CHODOROW, K. **MongoDB: The Definitive Guide**. O'Reilly Media, Inc.: Sebastopol, CA, USA, 2013.
18. VEEN, J. S.; WAAIJ, B.; MEIJER, R. J. **Sensor data storage performance: SQL or NoSQL, Physical or Virtual**. In *Proceedings of the IEEE 5th International Conference on Cloud Computing (CLOUD)*, Honolulu, HI, USA, 24–29. pp. 431–438. 2012.
19. NYATI, S.S.; PAWAR, S.; INGLE, R. **Performance evaluation of unstructured NoSQL data over distributed framework**. In *Proceedings of the 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Mysore, India, 22–25. pp. 1623–1627. 2013.
20. KANG, Y.S.; PARK, I.H.; RHEE, J.; LEE, Y.H. **MongoDB-based repository design for IoT-generated RFID/sensor big data**. *IEEE Sens.* 16, 485–497. 2915.
21. DOWNIE, N. **Chart.js**. Disponível em: <http://www.chartjs.org/>.
22. EUROPEAN COMMISSION. **A Framework for Indicators for the Economic and Social Dimensions of Sustainable Agriculture and Rural Development**. Agriculture Directorate-General. Disponível em: http://ec.europa.eu/agriculture/publi/reports/sustain/index_en.pdf. Acessado em: 10 de abril de 2018.