

HORTOMAÇÃO: SISTEMA DE AUTOMAÇÃO DE HORTA UTILIZANDO IOT EM PARCERIA COM ESCOLA ESTADUAL

*HORTOMAÇÃO: VEGETABLE GARDEN AUTOMATION SYSTEM USING IOT IN
PARTNERSHIP WITH STATE SCHOOL*

Marcos M. A. Ribeiro¹, Osvalir Valle Junior², Cristiano P. Martins³

¹Faculdade de Tecnologia Prof. José Camargo – Fatec Jales, marcos.ribeiro26@fatec.sp.gov.br

²Faculdade de Tecnologia Prof. José Camargo – Fatec Jales, osvalir.valle@fatec.sp.gov.br

³Faculdade de Tecnologia Prof. José Camargo – Fatec Jales, cristiano.martins01@fatec.sp.gov.br

**Informação e comunicação
Subárea: Tecnologia da Informação**

RESUMO

O presente trabalho tem o intuito de orientar os alunos da escola EE Dom Artur Horsthuis a automatizar a horta da instituição, pelo motivo de a escola não ser aberta em finais de semana, impossibilitando assim, o canteiro de ser regado pelos alunos. Pensando nisso, foi firmada uma parceria entre a Fatec Jales e a escola estadual para desenvolver uma solução que proporcionasse aos alunos noções de automação, programação e IoT (Internet of Things). Para a solução foram utilizados o microcontrolador ESP32, sensores e um atuador para que a irrigação funcionasse automaticamente por intermédio de um aplicativo mobile. O usuário pode acompanhar em tempo real a umidade do solo da horta através do acesso a uma listagem sobre o histórico da irrigação e uma tela com as opções de ligar e desligar a irrigação. Os estudantes da Fatec Jales foram instruídos a estudar sobre as tecnologias envolvidas e depois compartilhar com os alunos da escola através de encontros periódicos. O projeto atingiu os objetivos, pois além dos conceitos compartilhados, foi desenvolvido o aplicativo e o sistema de automação com o ESP32. Os professores de ambas instituições acompanharam cada fase do processo e todos compartilharam e receberam novos conceitos, o que aproximou também o ensino superior da comunidade.

Palavras-chave: automatização; ESP32; aplicativo; parceria.

ABSTRACT

The present paper aims to guide the students of EE Dom Artur Horsthuis school to automate the institution's vegetable garden, because the school is not open on weekends, thereby making it impossible for the flowerbed to be watered by the students. Bearing that in mind, a partnership was signed between Fatec Jales and the state school to develop a solution that would provide students with notions of automation, programming and IoT (Internet of Things). As a solution, the ESP32 microcontroller, sensors and an actuator were used for the irrigation to work automatically through a mobile application. The user can monitor the soil moisture in the garden in real time through access to a listing on the irrigation history and a screen with the options to turn irrigation on and off. Fatec Jales students were instructed to study about the involved technologies and then share with the school's students through periodic meetings. The project achieved its objectives, as in addition to the shared concepts, the application and the automation system were developed with ESP32. Professors from both institutions accompanied each phase of the process and all shared and received new concepts, which brings higher education closer to the community.

Keywords: automation; ESP32; application; partnership.

1 INTRODUÇÃO

Atualmente é muito difícil encontrar uma área que ainda não tenha alguma integração com a tecnologia, pois sem ela, o setor encontra dificuldades para evoluir e se mantém estagnado em relação ao mercado, podendo gerar consequências ruins às empresas. A tecnologia traz benefícios como automação e otimização de tarefas, pois, com sua evolução constante, é uma alternativa muito chamativa para empresas e para as pessoas possuírem a tecnologia em suas casas, levando-nos a pensar sobre IoT (*internet of things*, internet das coisas).

A internet das coisas, hoje, representa o crescimento de dispositivos com conectividade e internet. Isso significa que “coisas”, podendo ser um pequeno objeto ou até mesmo ambientes, adquirem capacidade de se comunicar entre eles e conseqüentemente com pessoas (CARRION; QUARESMA, 2019). Em geral, como exemplo tem-se a Alexa, que é uma assistente virtual da Amazon. Através dessa tecnologia, é possível fazer a integração, caso o usuário possua uma casa inteligente, conectando os equipamentos com a assistente usando apenas comandos de voz para controlar dispositivos eletrônicos da residência, bastando ter acesso à internet.

Com isso em mente, o projeto teve início a partir de uma parceria entre a Fatec Jales e a escola estadual Dom Artur Horsthuis. A escola parceira apresentou um problema em relação à irrigação de hortaliças no canteiro da escola pelos alunos, pois nos finais de semana a escola permanece fechada. Como essas plantas são sensíveis à temperatura e umidade e uma vez que os alunos não podem regá-las nesses dias da semana e feriados, veio a ideia da automação da horta. Os professores responsáveis pela horta entraram em contato com a FATEC para que fosse desenvolvido um projeto de automação da horta da escola. Para que a automação fosse efetivada, pensou-se no microcontrolador Arduino com um *shield* Wi-Fi para uma comunicação sem fio e o uso de sensores e atuadores que fizesse a leitura do solo e, se necessário, acionasse uma válvula que irrigaria o canteiro.

Também foi pensado num software com o intuito do usuário poder acompanhar com precisão as informações a respeito do canteiro e desligar o modo automático caso queira. Sabendo disso foram escolhidas algumas tecnologias para o desenvolvimento do aplicativo, fazendo uso da biblioteca React Native (Javascript) para o desenvolvimento do *front-end* (interface gráfica para o usuário) do sistema, PHP para o *back-end* (sistema em servidor em que são realizadas as ações) e MySQL para o banco de dados, onde serão armazenadas as informações produzidas pelos sensores e controle de atuação.

O desenvolvimento é um dos objetivos do projeto, mas a integração entre os estudantes da FATEC e alunos da escola foi um dos maiores motivadores para impulsionar o projeto. A ideia foi ensinar aos alunos da escola Dom Artur sobre automação, assim, houve encontros entre os alunos da escola estadual e os estudantes da Fatec Professor José Camargo – Fatec Jales com auxílio dos professores das instituições envolvidas. De início, os alunos tiveram uma introdução com o Arduino com auxílio de simulador. Nos encontros seguintes, iniciou-se a prática real com o microcontrolador Arduino UNO. Dessa forma, foi possível mostrar a lógica com a parte da programação e dos componentes possibilitando a montagem do protótipo com o microcontrolador, sensores e atuadores. Por fim, foi integrado o software com as funções necessárias. O objetivo foi alcançado, uma vez que o projeto foi desenvolvido em parceria e proporcionou a aprendizagem dos alunos da escola, o crescimento dos estudantes da faculdade e a integração da Fatec com a comunidade, uma das grandes funções da instituição na região.

2 REFERENCIAL TEÓRICO

O Projeto de automação de horta não é inovador, mas foi utilizado para integração entre as instituições de ensino, buscando aplicar, na prática, vários conceitos apresentados e

desenvolvidos dentro da faculdade. Essa metodologia despertou a busca por novas tecnologias e aplicações. Inicialmente foi utilizado o Arduino, uma tecnologia disponível na FATEC e na Escola.

De todas as placas de prototipação Arduino que existem no mercado, a mais conhecida é a Arduino UNO. As características dessa placa permitem que ela possa ser utilizada na implementação de vários tipos de projetos de automação.

Porém, foi pesquisado sobre os módulos ESP8266 NodeMCU, que foram implementados para serem utilizados em vários tipos de projetos de automação em comparação ao Arduino UNO.

Em geral, o que sobressai entre o ESP e o Arduino é que, os modelos NodeMCU possuem mais memória e velocidade de processamento (80 MHz), sendo cerca de cinco vezes maior em comparação ao Arduino UNO (16 MHz). Mas o Arduino também tem seus pontos que o faz ser melhor em certos projetos, pois o ESP8266 NodeMCU só tem a presença de uma única porta de entrada analógica e menor número de pinos GPIO (*General-Purpose Input-Output*, basicamente uma entrada e saída de uso geral; é um tipo de pino que está presente em circuitos integrados sem função específica, ficando vago para o objetivo do usuário) úteis (AUTOMAÇÃO EM 5 MINUTOS, 2019). Porém existe o ESP32, uma evolução do ESP8266.

2.1 ENSINO DA PROGRAMAÇÃO NO BRASIL

No artigo de Silva (2016) é citado o trabalho de Pereira Júnior e Rapkiewicz (2005) que mostra uma visão crítica sobre o ensino da programação no Brasil de 1999 até 2004. O artigo apresenta todos os trabalhos publicados no SBIE e WIE nesse período. O trabalho também destacou os artigos que focavam no ensino da programação e nas estratégias de artigos.

[...] ao analisar que pesquisas têm sido feitas no Brasil através de dois dos maiores eventos sobre Informática e Educação (WEI e SBIE) verificou-se que a comunidade científica brasileira tem buscado constantemente propor soluções para os problemas dentro do tema adotado, porém, ainda é pouco se comparado a pesquisas em outras áreas e domínios (PEREIRA JÚNIOR; RAPKIEWICZ, 2005, p. 5 apud SILVA, 2016).

No entanto, há muitas dificuldades em abordar o conteúdo com alunos da educação básica (HIRANO, 2015 apud SILVA, L., 2018). Kelleher e Pausch (2005 apud SILVA, L., 2018) afirmam que as ferramentas de construção de códigos não contribuem positivamente para a aprendizagem, pois suas sintaxes são muito complexas para os iniciantes, principalmente quando o ensino é aplicado a crianças. Dessa forma, é necessária a realização de pesquisas para desenvolver metodologias capazes de abordar o ensino de lógica de programação de forma lúdica e criativa.

Pensando nisso, foram preparadas aulas com conteúdos relacionados à programação de forma que ficasse simples para os alunos entenderem, ser dinâmico e despertar o interesse para essa área. Fez-se uso constante de projetos práticos nos encontros, juntamente tirando eventuais dúvidas relacionadas às aulas e atividades que os alunos tinham que desenvolver, para exercitar o conteúdo apresentado e gradualmente foram apresentados novos conteúdos, que se tornaram um pouco mais difíceis, mas os alunos não apresentaram dificuldades de entendimento.

3 METODOLOGIA

Esse projeto propõe um sistema para automação de irrigação de um canteiro de horta em um ambiente escolar. É fruto de uma parceria entre a escola Dom Artur Horsthuis e a Fatec Jales, com foco principal no ensino de programação para os alunos da escola.

O processo de desenvolvimento do projeto iniciou-se com a apresentação dos envolvidos e a realização de uma análise do local. Decidiu-se que o formato de encontros entre os alunos das instituições seria o mais eficaz e funcional, pois os universitários teriam um período para preparar o encontro juntamente com o professor, e, assim resultando em orientações elaboradas, sucintas e de fácil entendimento para os alunos da escola. O conteúdo dos encontros teve o foco em tecnologias como a IoT e programação básica, para que também os alunos não tivessem tantas dificuldades de compreensão e aplicação das tecnologias envolvidas.

Durante o levantamento bibliográfico das tecnologias disponíveis, foi decidido utilizar o microcontrolador ESP32 por ser uma tecnologia promissora em projeto IoT. Desse modo, deu-se início às pesquisas sobre esse módulo e como utilizá-lo e de projetos com o mesmo intuito. Tendo feito esse levantamento, percebeu-se que poucos projetos possuíam placas com wifi integrado e nenhuma em específico com o módulo citado, assim, seguindo com a ideia foram levantados pelos alunos juntamente com o professor, os sensores e atuadores a serem usados no protótipo.

Para o desenvolvimento do aplicativo optou-se por servir-se das tecnologias utilizadas durante a graduação como C, Javascript, PHP, React Native e MySQL.

3.1 APRESENTAÇÃO DOS MICROCONTROLADORES PESQUISADOS

No artigo é comparado também o uso da placa ESP32, não só a ESP8266 e Arduino Uno. Nota-se que a placa ESP32 tem o maior espaço de memória RAM, permite uma corrente elétrica maior no circuito, e possui *Bluetooth* e WiFi no seu sistema.

Existe uma grande diferença entre memórias RAM e FLASH da ESP para o Arduino. Em quesito arquitetura, as duas placas modelo ESP trabalham com um processador 32 bits, enquanto o Arduino trabalha com um de 8 bits.

Contudo, o uso de cada placa depende, em maior parte, do objetivo no projeto. Caso o sistema não necessite de conexão WiFi, *Bluetooth* e não precise de um espaço maior, considera-se que o Arduino seja a opção mais viável. Para projetos mais complexos envolvendo IoT, o ESP se mostra uma opção mais vantajosa. Ainda assim, se o uso das placas for para fins estudantis e para aprendizado, mostra-se que o Arduino é melhor pela facilidade de manuseio para iniciantes, para depois, utilizar placas com mais ferramentas como a ESP (TORELLA, 2022).

Em outra comparação, começando pelo Arduino, que em relação a programação nos microcontroladores da família PIC (*peripheral interface controller*), que segundo o trabalho de Cartolano Júnior e Avegliano (2004) são:

4 (quatro) famílias de microcontroladores de 8 bits no barramento de dados que se adaptam ao objetivo do projeto. Sendo elas:
PIC12CXXX: Linha Compacta.
PIC16C5X/PIC16C55X: Linha Base.
PIC16CXX: Linha Intermediária.
PIC17CXX: Topo de Linha.

Precisa-se trabalhar diretamente com registradores; no Arduino o usuário especifica o nível lógico na porta que ele deseja, deixando-o livre para pensar na lógica de cada comando e não no hardware em si. Esse processo tem dois lados: o primeiro lado é que o usuário fica livre para pensar no hardware, registradores, frequência do cristal e selecionar as configurações; o segundo lado é que a eficiência do código cai. Em códigos grandes, a eficiência tem um grande impacto, pois o código não está otimizado com o hardware, o que faz com que o tempo gasto para processar o comando aumente, mas para códigos pequenos e objetivos a diferença apresentada não chega a ser tão relevante (OLIVEIRA, 2019).

Inicialmente escolheu-se utilizar o Arduino como microcontrolador, juntamente com sensores e atuadores, para a automatização do canteiro com hortaliças, uma vez que seriam definidos horário e prazo programados para início e finalização da irrigação. Porém, não havia sido citado a necessidade do monitoramento nem possibilidade de controle remoto pelo usuário. A placa Arduino é bastante limitada nesse ponto por não possuir conexão Wi-fi de forma nativa, assim, foram necessários outros componentes para acessar a internet ou redes disponíveis, dificultando a troca de informações com servidores e bancos de dados e a comunicação com aplicativos.

Nesse levantamento inicial, observou-se que no artigo de Souza et al. (2021) também foi escolhido como microcontrolador o Arduino, e foi usado o monitor para serem mostrados os registros coletados com os sensores. Essa não é a forma mais eficiente de amostragem de dados para o usuário, pois ele teria que entender pelo menos de algumas operações com o ambiente de programação do Arduino para poder compilar o projeto e acompanhar os dados e, se houve irrigação ou não no presente momento. Além disso, apenas com o Arduino, a complexidade para se trocar informações com bancos de dados e com aplicativos é maior se comparada com o uso de outros microcontroladores, tais como o ESP8266 e o ESP32 (SOUZA et al., 2021).

3.2 ARDUINO UNO, ESP8266 E ESP32

Agora serão apresentados os microcontroladores pesquisados e suas características. Através da Figura 1, é possível observar a diferença física entre os equipamentos.

Figura 1 – Microcontroladores Arduino UNO, ESP8266 e ESP32



Fonte: Elaborado pelos autores.

O Arduino teve seu início em 2005 na Itália, pelo professor Massimo Banzi, para ensinar eletrônica e programação para seus alunos de design. Com ajuda de seu aluno David Mellis que criou a linguagem de programação do Arduino e de David Cuartielles que o ajudou a criar a placa (MARINHO, 2012).

O ESP inicialmente, foi desenvolvido por uma empresa chinesa chamada Espressif que em 2014 fabricou seu primeiro chip, o ESP8266, utilizando-o no módulo ESP-01. O ESP8266 teve um crescimento de popularidade e, com isso, vários fabricantes criaram módulos diferentes e um dos mais conhecidos é o NodeMCU ESP8266 (OLIVEIRA, 2019).

O maior diferencial entres os microcontroladores é a velocidade de processamento, memória e a presença de conexão Wi-fi integrada. O Arduino fica em desvantagem comparado aos ESP em velocidade de processamento com sua frequência máxima alcançando apenas 16MHz e o ESP32 podendo chegar a 240MHz em cada um dos 2 núcleos, sendo esse o mais poderoso entre os três. Em memória ROM e RAM o Arduino também é inferior tendo 2KB de RAM e 4KB de ROM, o ESP8266 com 36KB de RAM e 64Kb de ROM e o ESP32 520KB de

RAM e 448KB de ROM, novamente apresentando características superiores entre os modelos citados.

A presença de conexão sem fio (Wi-fi) se dá apenas nos modelos ESP. Apenas o ESP32 possui Bluetooth. O Arduino não possui conexão sem fio. Uma vantagem do Arduino em relação aos outros microcontroladores é sua alimentação que é de 5V, com os outros modelos sendo de 2,2V até 3,3V apenas. A temperatura média de operação entre os microcontroladores pode-se considerar equivalente, não possuindo nenhum com temperatura menor, não sendo um critério que se considera relevante, apenas em casos isolados.

O ESP32 é a parte principal do sistema, responsável por realizar a ligação entre os sensores e acionadores (HENRIQUES, 2019). Esse foi o microcontrolador escolhido para o projeto em razão das características, pelo custo acessível e pela necessidade de acessar a internet uma das características de aplicações IoT.

3.3 LINGUAGEM DE PROGRAMAÇÃO UTILIZADA NO ESP32

Devido a placa ser desenvolvida para projetos de IoT, é uma das mais populares do mercado. Sua linguagem de programação oficial é a Lua. Ela não é uma linguagem difícil de ser compreendida, porém requer estudo para melhor compreensão (COMO..., 2022). Contudo, a linguagem escolhida para o projeto foi a linguagem C, por ser utilizada na faculdade.

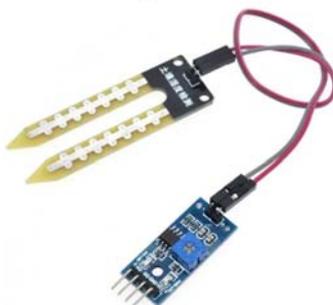
3.4 SELEÇÃO DE SENSORES E ATUADORES

As placas microcontroladores conseguem ler sensores e acionar atuadores com base nos pinos de entrada e saída, indiferente se são digitais ou analógicas (CANTU, 2021). Sendo assim, os sensores geram leituras em “troca de informações” com o microcontrolador e os atuadores são acionados pela placa com base na lógica usada. A ideia no projeto é de o sensor de umidade ler a informação em um intervalo de tempo e se estiver abaixo do necessário para a planta crescer a lógica do microcontrolador irá abrir a válvula e regar as plantas.

3.4.1 Sensor de Umidade do Solo

O sensor de umidade de solo é apresentado através da Figura 2. A principal função é fazer a leitura da umidade do solo e informar a um microcontrolador (BAÚ DA ELETRÔNICA, 2022b). A composição do solo pode afetar os valores de medição do sensor por conta de sais minerais presentes no solo. O sensor também possui duas saídas, uma analógica que retorna o valor lido da umidade e a digital, que retorna um sinal alto (3,3V) ou baixo (0V) (ADOLPHS, 2020).

Figura 2 – Sensor escolhido para medição da umidade do solo



Fonte: BAÚ DA ELETRÔNICA, 2022b.

3.4.2 Válvula Solenoide

A válvula escolhida no projeto é uma Válvula Solenoide para Água 127V 180° (1/2 x 1/2). Seu princípio básico é que uma corrente elétrica passe pela bobina deixando-a energizada e criando um campo magnético, assim o êmbolo de dentro da bobina é movido e a válvula se abre (PANIN, 2022). O modelo da válvula Solenoide é apresentado na Figura 2.

Figura 3 – Válvula Solenoide para Água 127V escolhida para o projeto



Fonte: CANAL AGRÍCOLA, 2022.

3.4.3 Módulo Relé

O relé pode ser usado em muitos projetos a fim de acionar cargas de até 250V, 7A ou 125V, 10A. Ele pode ter várias funções, como: ligar equipamentos de uma rede, controlar luzes, motores etc. (BAÚ DA ELETRÔNICA, 2022a). O módulo é apresentado através da Figura 4.

Figura 4 – Módulo relé 3,3V



Fonte: BAÚ DA ELETRÔNICA, 2022a.

3.5 DEFINIÇÃO DE TECNOLOGIAS

3.5.1 Linguagens de Programação e Banco de Dados

Dentre as linguagens utilizadas no projeto é possível apresentar a linguagem C, que tem aplicabilidade para qualquer tarefa de programação, sendo utilizada em vários projetos e oferece liberdade e controle ao programador (RICARTE, 1995). A linguagem foi utilizada no projeto para programar a parte do protótipo com o ESP32 e integração com os sensores e atuadores.

A linguagem Javascript é uma linguagem de programação Web, sendo usada em sua maioria nos sites e navegadores modernos, em computadores, consoles de jogos, *tablets* e *smartphones*; também roda em vários navegadores sendo eles Google Chrome, Firefox, Safari e Edge. Algumas páginas web que apresentam informações estáticas tem o apelido de documentos. Outras se assemelham a documentos; estas podem carregar informações, ser gráficas e podem operar offline e salvar dados para que assim que o usuário a visite novamente elas possam restaurar os dados. Tudo isso graças ao Javascript. Ao usá-lo em uma aplicação, os dois maiores benefícios que se tem são: a escalabilidade, podendo no futuro uma aplicação aguentar a demanda de crescimento de usuários; o outro benefício ao optar por esta linguagem é a performance, que é o desempenho do site, levando em conta os recursos que serão consumidos, pois o Javascript é uma linguagem orientada a eventos. Isso faz com que as operações contidas em um software só sejam executadas caso forem requisitadas.

O React Native é um framework JavaScript, para programação e renderização de aplicações móveis para iOS e Android. Baseado em React, que é a biblioteca Javascript do Facebook, mas ao contrário do React ela não visa construir interfaces Web e sim plataformas móveis, possibilitando assim ao programador escrever aplicações móveis com visual realmente “nativo”, com o uso de uma biblioteca JavaScript, que permite ser executada e interagir com iOS e Android, de forma a se igualar ao código nativo (SILVA, W., 2018). O React Native renderiza os próprios componentes para visualização nativa real para Android ou iOS. Sendo possível por conta de sua camada de abstração, mais conhecida como “ponte”, que permite que ele invoque APIs de renderização Java no caso de aparelhos Android e Objective-C em dispositivos da maçã. E como ele é baseado em JavaScript, isso faz com que o aplicativo consiga acesso a informações específicas da plataforma que está implantado, como bateria ou localização (CRUZ; PETRUCCELLI; SOTTO, 2018). Sendo o *framework* (base do sistema) com mais familiaridade por parte dos autores, sendo assim optado para o projeto.

Inicialmente, chamada de PHP/FI, a linguagem PHP foi criada em 1994 por Rasmus Lerdof. A ferramenta era um conjunto de binários escritos em C. Em 1998 com muitos *devs* (desenvolvedores) do mundo se esforçando, para isso, foi lançado o PHP 3.0, anunciado pelo time de desenvolvedores do PHP, sendo essa a primeira versão que mais pode ser assemelhada com o PHP atual (TAVARES; CREMASCO, 2010). Atualmente é uma linguagem de script *open source* (código aberto), com sua principal atuação no *back-end*, no processamento de dados. O PHP é orientado a objetos, então podendo contar com conceitos como polimorfismo, encapsulamento, abstração e herança (PIETRO; PETRUCCELLI, 2019). Com foco no projeto o PHP foi escolhido devido ser uma linguagem *back-end* orientada a objetos, não ser de difícil entendimento e ter compatibilidade com o React Native, anteriormente citado.

O banco de dados utilizado foi o MySQL. “MySQL é uma implementação cliente-servidor que consiste em um servidor e diferentes programas clientes e bibliotecas.”, como é ressaltado por Gonzaga e Birckan (2000). O SQL é uma linguagem adotada por padrão que facilita no armazenamento e acesso das informações. Mesmo não tendo necessidade de um volume tão grande a armazenamento de informações, o MySQL consegue manipular os dados de uma forma fácil e com um volume alto de informações. Ele possui muitas vantagens e entre elas podem ser citadas, a fácil manipulação de um número ilimitado de usuários simultâneos, a alta velocidade de execução, também possuir APIs em PHP, que é a linguagem back-end usada no projeto, alta disponibilidade de tipos de dados, possibilidade de consultar tabelas de diferentes bancos em uma busca só e o servidor emite mensagens de erro em várias linguagens. Por esses e outros benefícios, optou-se pelo MySQL, e também pela familiaridade dos envolvidos no projeto com essa tecnologia (GONZAGA; BIRCKAN, 2000).

4 ANÁLISE E DISCUSSÃO DOS RESULTADOS

No início do projeto houve encontros para que os envolvidos no projeto se conhecessem, para analisar o local que o projeto seria implantado e verificar os componentes que seriam utilizados para a confecção do sistema. Foi também criado um grupo com os envolvidos para combinar-se os encontros e tirar dúvidas dos alunos.

Com foco no ensino de automação aos alunos da escola Dom Artur, houve encontros entre eles e os estudantes da Fatec, logicamente com a presença de professores de ambas as instituições. Para que alunos tivessem uma introdução, inicialmente foi apresentado o TinkerCAD, que é um simulador de circuitos, para assim explicar aos alunos a estrutura da placa Arduino UNO, como fazer ligações entre componentes, ligar um led, tudo sendo visto por eles em código e a lógica responsável pela operação, para posteriormente eles mesmos fazerem as ligações e códigos sozinhos e tirarem suas dúvidas com os acadêmicos. Os alunos aprenderam a fazer um semáforo no simulador com os conhecimentos adquiridos, como é possível observar na Figura 5. Após este encontro eles tiveram um contato com o sensor de luminosidade e aprenderam como funciona a ligação dele e como ver os dados gerados pelo sensor na tela, ainda pelo simulador.

Figura 5 – Universitário explicando aos alunos como usar o simulador para aprender a fazer um semáforo em Arduino, com supervisão do professor



Fonte: Elaborado pelos autores.

A partir desses encontros, os alunos tiveram um primeiro contato com os componentes físicos. Inicialmente, todos eles fizeram um led piscar; após isso aprenderam como fazê-lo apagar conforme a luminosidade que o sensor absorvia, e a lógica para programar essa função, passando assim os conhecimentos com o simulador para a prática real, mostrado através da Figura 6.

Figura 6 – Alunos aprendendo Arduino na prática



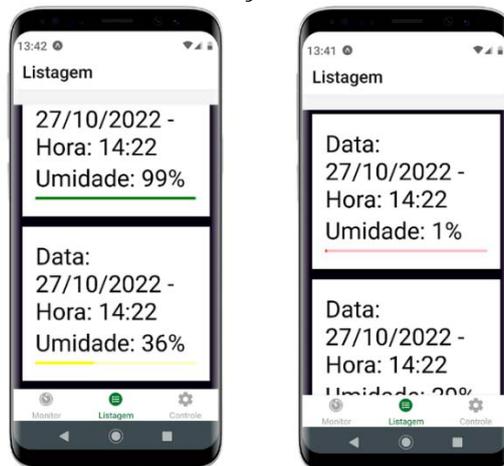
Fonte: Elaborado pelos autores.

Posteriormente, os alunos tiveram a oportunidade de conhecer sobre banco de dados, ensinando para eles como armazenar as informações do Arduino em um banco de dados. Para isso usou-se o Arduino com sensor de luminosidade para fazer leituras e gerar valores, a linguagem de programação Java com a biblioteca RxTx para pegar os dados do sensor, e por fim, o MySQL para armazenagem dos dados.

Finalmente, os alunos tiveram a oportunidade de montar o protótipo com o uso do microcontrolador ESP32 e com o sensor de umidade, aprendendo também como programar ambos, com o uso da linguagem C, para o funcionamento do sensor e configuração do microcontrolador, por meio da plataforma de desenvolvimento Arduino IDE.

O aplicativo desenvolvido para monitoramento foi desenvolvido utilizando-se das tecnologias citadas no item 2.5. Contém tela para monitoramento da umidade do solo do canteiro, como apresentado na Figura 7.

Figura 7 – Telas de listagem de monitoramento do canteiro, contendo níveis de umidade e datas em que foram lidas as informações



Fonte: Elaborado pelos autores.

Juntamente com uma listagem que informa precisamente o período que o canteiro foi ligado ou desligado, como observa-se na Figura 8.

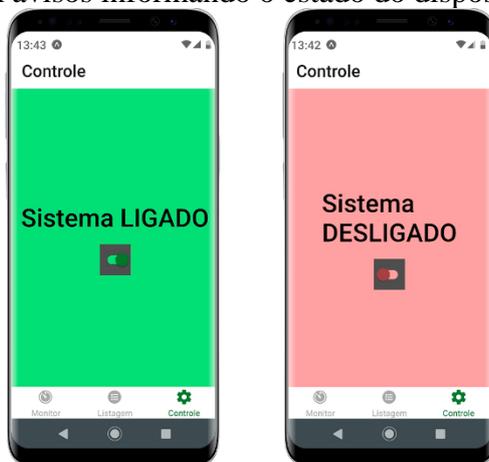
Figura 8 – Tela de listagem para informar ao usuário quando a irrigação foi ligada ou desligada com precisão



Fonte: Elaborado pelos autores.

Na próxima tela, apresentada na Figura 9, é oferecido um botão para ligar e desligar a irrigação com a confirmação representada pela cor e pelo plano de fundo da tela.

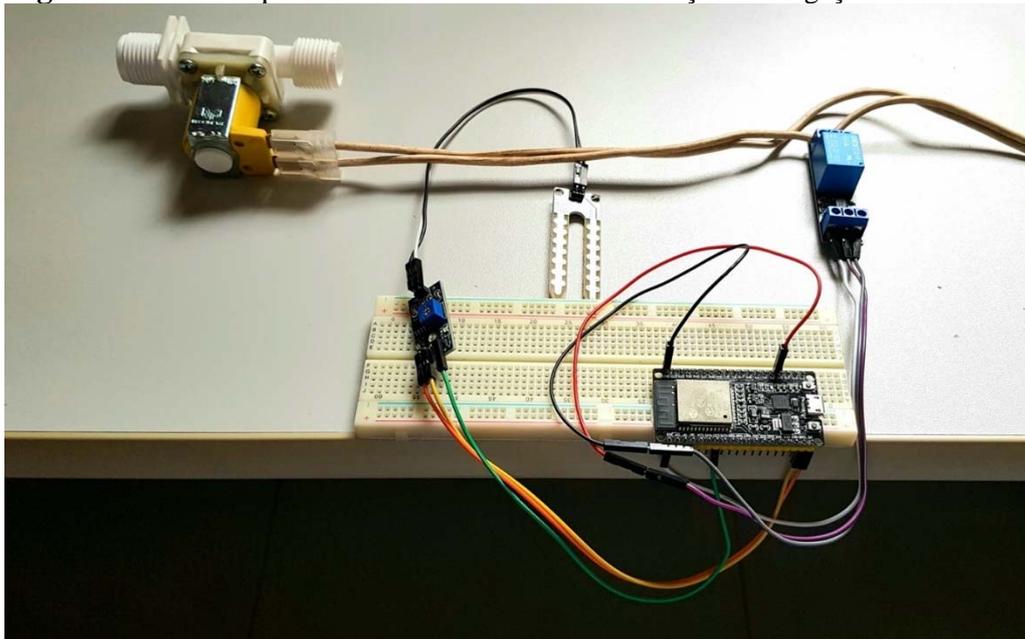
Figura 9 – Telas de controle do sistema, podendo ser ligado e desligado, com avisos informando o estado do dispositivo acima do botão



Fonte: Elaborado pelos autores.

A Figura 10 é utilizada para mostrar todo o sistema eletrônico de automação. Foram envolvidos a placa de prototipação ESP32 com sistema de transmissão wifi, um relé para controlar a válvula solenoide, além do sensor de umidade para fazer a leitura do solo.

Figura 10 – Protótipo do sistema eletrônico de automação de irrigação utilizando IoT



Fonte: Elaborado pelos autores.

5 CONSIDERAÇÕES FINAIS

O objetivo do ensino de *IoT* e programação para os alunos da escola Dom Artur, que é o principal foco da parceria, foi realizado com êxito, ao qual os alunos absorveram todo conteúdo passado pelos universitários. Alguns alunos até mesmo usaram os conhecimentos adquiridos para projetos a serem enviados para feiras de tecnologia entre escolas estaduais.

O projeto foi desenvolvido, mas a implantação do protótipo no ambiente ainda não foi feita, pois dentro do cronograma acontecerá na última fase. Com os testes feitos e ações realizadas concluiu-se que o projeto pode ser implementado na escola de forma que funcione corretamente. Foram realizados testes com o acionamento da válvula em um canteiro e no aplicativo comprovando as funcionalidades solicitadas.

Uma dificuldade encontrada é quando a internet falhar. Caso a válvula seja aberta pelo aplicativo e a internet não estiver funcionando, o próprio sistema no Arduino lê a umidade do canteiro e fecha a válvula de forma automática.

O aplicativo *mobile* teve seu desenvolvimento finalizado e testado em ambiente simulado. Conclui-se que o programa cumpre seu papel, dando ao usuário o controle proposto para monitorar a horta, tendo informações reais sobre o canteiro em qualquer instante e local que tenha acesso à internet, sabendo quando foi irrigado pela última vez juntamente com os dados da horta no momento da irrigação e podendo também desligar a irrigação automática, caso o usuário necessite.

REFERÊNCIAS

ADOLPHS, R. S. **Projeto**: sistema de irrigação automático supervisionado para hortas. 2020. Trabalho de Conclusão de Curso (Bacharel em Engenharia de Controle e Automação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2020. Disponível em: <https://www.lume.ufrgs.br/bitstream/handle/10183/217376/001120677.pdf?sequence=1>. Acesso em: 15 out. 2022.

AUTOMAÇÃO EM 5 MINUTOS. **ESP8266 NodeMCU**: o substituto do Arduino em soluções IoT. 2019. Disponível em: Acesso em: 21 set. 2022.

BAÚ DA ELETRÔNICA. **Módulo Relé 5V**. Disponível em:
https://www.baudaeletronica.com.br/modulo-rele-5v.html?gclid=Cj0KCQjwteOaBhDuARIsADBqRejQ8ENToIomX_GHTXdkV-sFSDG-5GpVAAD8dskAfpFcuSdEeGspdfUaAvIIEALw_wcB. Acesso em: 25 out. 2022a.

BAÚ DA ELETRÔNICA. **Sensor de umidade do solo**. Disponível em:
https://www.baudaeletronica.com.br/sensor-de-umidade-do-solo.html?gclid=CjwKCAjwKmaBhBMEiwAyINuwNqFf_HV0178f9dVjIFh1NMdLu2E2mHtlXW_q2QCcytAKlxFuynoBoCfMUQAvD_BwE. Acesso em: 13 out. 2022b.

CANAL AGRÍCOLA. **Válvula Solenóide de entrada de água 180° Rosca 3/4" VA04 127V**. Disponível em: https://www.canalagricola.com.br/valvula-solenoid-entrada-agua-va04-127v?gclid=CjwKCAiAv9ucBhBXEiwA6N8nYA1FJctUwi4jauOQfi9jKpjUBpoWWWhN8_b5zStLytxf0N-fGLMYfmxC4oUQAvD_BwE. Acesso em: 13 out. 2022.

CANTU, E. **Arduino**: sensores e atuadores. 2021. Disponível em:
http://wiki.foz.ifpr.edu.br/wiki/index.php/Arduino:_Sensores_e_Atuadores. Acesso em: 25 out. 2022.

CARRION, P.; QUARESMA, M. Internet da Coisas IoT: definições e aplicabilidade aos usuários finais. **Human Factors in Design**, Florianópolis, v. 8, n. 15, p. 49-66, 2019. Disponível em:
<https://www.revistas.udesc.br/index.php/hfd/article/view/2316796308152019049>. Acesso em: 21 out. 2022.

CARTOLANO JÚNIOR, E. A.; AVEGLIANO, P. B. **Experiência microcontroladores PIC**. 2004. Disponível em: <https://www.trabalhosfeitos.com/ensaios/Introdu%C3%A7%C3%A3o-Ao-Pic/64118974.html>. Acesso em: 12 out. 2022.

COMO programar o NodeMCU com Arduino IDE. Disponível em:
<https://www.robocore.net/tutoriais/como-programar-nodemcu-arduino-ide#:~:text=Desenvolvida%20para%20ser%20uma%20plataforma,programa%C3%A7%C3%A3o%20oficial%20%C3%A9%20a%20Lua>. Acesso em: 11 out. 2022.

CRUZ, V. S.; PETRUCCELLI, E. E.; SOTTO, E. C. S. A linguagem Javascript como alternativa para o desenvolvimento de aplicações multiplataforma. **Revista Interface Tecnológica**, Taquaritinga, v. 5, n. 2, 2018. Disponível em:
<https://revista.fatectq.edu.br/interfacetecnologica/article/view/476/297>. Acesso em: 25 out. 2022.

GONZAGA, F. S.; BIRCKAN, G. **Curso de PHP e MySQL**. 2000. Disponível em:
<https://docplayer.com.br/476523-Curso-de-php-e-mysql-flavio-s-gonzaga-bim-inf-ufsc-br-guilherme-birckan-birckan-inf-ufsc-br.html>. Acesso em: 22 out. 2022.

HENRIQUES, R. V. B. **Laboratório Maker**: introdução à internet das coisas IoT. Porto Alegre, 2019.

MARINHO, M. **Introdução ao Arduino**. 2012. Disponível em:
https://www.academia.edu/5229813/Introdu%C3%A7%C3%A3o_ao_Arduino. Acesso em:
23 out. 2022.

OLIVEIRA, J. **Arduino, ESP32 e ESP8266: comparação**. 2019. Disponível em:
<https://xprojetos.net/arduino-esp32-e-esp8266-comparacao/>. Acesso em: 18 out. 2022.

PANIN, L. K. **Válvula solenoide: o que é? como funciona?** Disponível em:
<http://www.tecniar.com.br/noticias/valvula-solenoide-o-que-e-como-funciona/>. Acesso em:
13 out. 2022.

PIETRO, F. E.; PETRUCCELLI, E. E. Website fotográfico desenvolvido em PHP. **Revista Interface Tecnológica**, Taquaritinga, v. 16, n. 1, 2019. Disponível em:
<https://revista.fatectq.edu.br/interfacetecnologica/article/view/621/343>. Acesso em: 25 out.
2022.

RICARTE, I. L. M. **Programação orientada a objetos com C++**. 1995. Disponível em:
https://www.researchgate.net/profile/Ivan-Ricarte/publication/242509581_Programacao_orientada_a_objetos_Desenvolvimento_avanca_do_em_C/links/609d361092851cfd32d0cd6/Programacao-orientada-a-objetos-Desenvolvimento-avancado-em-C.pdf. Acesso em: 21 out. 2022.

SILVA, J. C. **Ensino de programação para alunos do ensino básico: um levantamento das pesquisas realizadas no Brasil**. 2016. Artigo de Graduação (Licenciatura em Ciência da Computação) – Universidade Federal da Paraíba, Mamanguape, 2016. Disponível em:
<https://repositorio.ufpb.br/jspui/bitstream/123456789/3328/1/JCS14062017.pdf>. Acesso em:
13 out. 2022.

SILVA, L. O. S. **Ensino de lógica de programação para alunos do ensino fundamental utilizando a ferramenta scratch**. 2018. Trabalho de Conclusão de Curso (Bacharel em Ciência da Computação) – Universidade Federal do Pará, Belém, 2018. Disponível em:
https://bdm.ufpa.br:8443/jspui/bitstream/prefix/1517/1/TCC_EnsinoLogicaProgramacao.pdf. Acesso em: 21 out. 2022.

SILVA, W. C. S. **Aplicações móveis nativas com React Native e Firebase: um estudo de caso**. 2018. Monografia (Bacharel em Ciência da Computação) – Universidade Federal do Maranhão, São Luis, 2018. Disponível em:
<https://monografias.ufma.br/jspui/bitstream/123456789/3498/1/WERLITON-Silva.pdf>. Acesso em: 21 out. 2022.

SOUZA, J. S. *et al.* **Utilização da plataforma open-source Arduino para o controle de um sistema de irrigação de horta caseira baseado em lógica Et**. 2021. Disponível em:
http://www.netlogconference.com/proceedings/papers/NETLOG_2020_paper_80.pdf. Acesso em: 18 out. 2022.

TAVARES, D. F.; CREMASCO, E. F. **PHP: hypertext preprocessor**. 2010. Disponível em:
<http://www.inf.ufes.br/~vitorsouza/archive/2020/wp-content/uploads/teaching-lp-20162-seminario-php.pdf>. Acesso em: 22 out. 2022.

TORELLA, M. **Arduino ou ESP: descubra a melhor opção!** Disponível em:
<https://lobodarobotica.com/blog/arduino-ou-esp-descubra-a-melhor-opcao/>. Acesso em: 21 set. 2022.