

PROCEDIMENTO PARA INSTALAÇÃO DE UM SERVIDOR WEB LOCAL BASEADO EM RASPBERRY PI PARA UMA PLATAFORMA IOT. PROPOSTA DE APOIO PARA ATIVIDADES PRÁTICAS DE ENSINO

PROCEDURE FOR INSTALLATION OF A RASPBERRY PI-BASED LOCAL WEB SERVER FOR AN IOT NETWORK. PROPOSED AS A SUPPORT TOOL FOR TEACHING ACTIVITIES

PROCEDIMIENTO PARA LA INSTALACIÓN DE UN SERVIDOR WEB LOCAL BASADO EN UNA RASPBERRY PI PARA UNA RED IOT. PROPUESTA DE APOYO PARA ACTIVIDADES PRÁCTICAS DOCENTES

Derick Mazelli Barbosa¹
Andy Blanco-Rodriguez²

RESUMO: A Internet das Coisas (Internet of Things, IoT) vem, há muito tempo, aumentando a importância dos sistemas digitais integrados e que funcionam em grande escala. Este conceito evoluiu conforme a internet sem fio se tornou mais difundida e rápida. Sensores embarcados cresceram em sofisticação e as pessoas e empresas começaram a entender que a tecnologia pode ser uma ferramenta de uso tanto pessoal quanto profissional. Assim, a quantidade de dispositivos conectados cresceu, a capacidade de processamento e de armazenamento destes sistemas eletrônicos também aumentou, enquanto os custos de produção diminuíram. Atualmente existem várias plataformas de IoT privadas como Node-RED, Ubidots, TagoIO e ThingSpeak. Estas possibilitam a conexão de dispositivos de forma ágil e fácil, mas não permitem acesso a todos os subsistemas digitais, possuem restrições de configuração e oferecem pouca capacidade de aprimoramento para os usuários. Vale frisar que estas limitações das redes IoT comerciais tornam-se desvantagens significativas quando comparadas com as plataformas IoT desenvolvidas na medida da aplicação. Nesse contexto, surge o atual estudo, que tem o intuito de propor um procedimento geral e básico que auxilie na instalação de aplicações digitais em um servidor web local de baixo custo. Esta estrutura pode ser projetada como um exemplo de base para o desenvolvimento de uma plataforma IoT. O processo pode ser executado durante atividades de ensino em uma instituição educacional, devido principalmente à sua simplicidade, baixo custo e generalização. O sistema estará baseado em ferramentas de hardware de baixo custo e softwares de livre acesso. Espera-se contribuir com temas práticos relacionados ao ensino, colocar diversas tecnologias da informação/comunicação ao serviço dos estudantes e auxiliar no aprimoramento de habilidades e competências dos alunos por meio de atividades práticas com ferramentas digitais e de hardware.

Palavras-chave: Servidor web. Raspberry Pi. Internet das Coisas (IoT). Tecnologias aplicadas na educação. Atividades práticas de ensino.

¹ Estudante de graduação de Engenharia Física - Universidade de São Paulo - (USP).

² Professor - Universidade de São Paulo - (USP).

ABSTRACT: The Internet of Things (IoT) has long been increasing the importance of integrated digital systems that operate on a large scale. This concept evolved as wireless internet became more widespread and faster. Embedded sensors grew in sophistication, and people and companies began to understand that IoT technologies can be a tool for both personal and professional use. Thus, the number of connected devices grew, and the processing and storage capacity of these electronic systems also increased, while production costs decreased. Currently, there are several private IoT platforms such as Node-RED, Ubidots, TagoIO, and ThingSpeak. These enable the connection of devices quickly and easily, but they do not allow access to all digital subsystems, have configuration restrictions, and offer little capacity for user enhancement. It is worth noting that these limitations of commercial IoT networks become significant disadvantages when compared to IoT platforms developed specifically for custom applications. Then, this study aims to propose a general and basic procedure to assist in the installation of digital applications on a low-cost local web server. This structure can serve as a basic example for the development of an IoT platform. The process can be implemented during teaching activities in an educational institution, mainly due to its simplicity, low cost, and wide generalisation potential. The system will be based on low-cost hardware tools and open-source software. It is expected to contribute to practical topics related to teaching, put various information/communication technologies at the service of students, and assist in improving their skills and competencies through practical activities with digital and hardware tools.

Keywords: Web server. Raspberry Pi. Internet of Things (IoT). Technologies applied in education. Practical teaching activities.

RESUMEN: Internet de las Cosas (Internet of Things, IoT) hace mucho tiempo que ha acelerado la importancia de los sistemas digitales integrados y que funcionan en gran escala. Este concepto evolucionó a medida que el internet inalámbrico se volvió más difundido y rápido. Las aplicaciones con sensores embebidos crecieron en sofisticación y ganaron popularidad. Por eso, industrias, personas y empresas comprendieron que esta tecnología IoT puede ser una herramienta tanto de uso personal como profesional. De esa forma, aumentó la cantidad de dispositivos conectados a la red y también creció la capacidad de procesamiento y almacenamiento de estos sistemas electrónicos. Al mismo tiempo, sus costos de producción y comercialización se redujeron considerablemente. Actualmente existen diversas plataformas privadas de IoT como Node-RED, Ubidots, TagoIO e ThingSpeak. Estas posibilitan la conexión de dispositivos de manera ágil y fácil, pero no permiten acceso a todos los subsistemas digitales, poseen restricciones de configuración y ofrecen poca capacidad de mejora de los usuarios. Vale destacar que estas limitaciones de las redes IoT comerciales se convierten en desventajas significativas cuando se comparan con otras plataformas IoT desarrolladas a la medida de la aplicación. En ese contexto surge el actual estudio, que tiene como propósito indicar un procedimiento general y básico para auxiliar en la instalación de aplicaciones digitales en un servidor web local de bajo costo. Esta estructura puede ser proyectada como un ejemplo de base para el desarrollo de una plataforma IoT. El proceso puede ser ejecutado durante actividades de enseñanza en instituciones educativas, debido principalmente a su simplicidad, bajo costo y generalización. El sistema estará basado en herramientas de hardware de bajo costo y software de libre acceso. Se espera contribuir con temas prácticos relacionados a la enseñanza, colocar diversas tecnologías de la información/comunicación al servicio de los estudiantes y auxiliar en la mejora de habilidades y competencias de los alumnos por medio de actividades prácticas con herramientas digitales y de hardware.

Palabras clave: Servidor web. Raspberry Pi. Internet de las Cosas (IoT). Tecnologías aplicadas a la educación. Actividades prácticas en la enseñanza.

INTRODUÇÃO

O termo “Internet das Coisas” (ou Internet of Things, IoT) foi cunhado no final dos anos 90 do século passado pelo empresário Kevin Ashton. Ele foi um dos fundadores do Auto-ID Center no MIT e formava parte de uma equipe que descobriu como conectar objetos à internet por meio de uma etiqueta RFID (Radio Frequency Identification, ou Identificação por Radiofrequência). O próprio Ashton relatou ser o primeiro em usar a frase “Internet of Things” durante uma apresentação em 1999, e o termo se popularizou desde então (Rayes et al., 2019).

Internet das Coisas refere-se a uma sofisticada abordagem sobre a interconexão de objetos, sistemas e tecnologias, através da internet. Ao longo dos anos, essa metodologia proporcionou a criação de uma rede global de dispositivos (Koreshoff et al., 2013). Sendo que, uma característica importante dos sistemas IoT é a interoperabilidade. Segundo o Institute of Electrical and Electronics Engineers (IEEE), a interoperabilidade consiste na capacidade de um sistema de interagir com outros sistemas ou produtos de forma simplificada, sem nenhum esforço adicional (Vasseur et al., 2010; Tolk, 2013).

Baseadas na abordagem IoT, há um inúmero de aplicações acadêmicas, empresariais e industriais, que vêm crescendo aceleradamente (Rodrigo et al., 2025; Soares, 2026). Por exemplo, na área da saúde dispositivos conectados podem ajudar no monitoramento de atividades físicas, sono e outros hábitos. Para medição de poluentes atmosféricos, as plataformas de monitoramento com redes IoT são fundamentais (Moursi et al., 2021; Vormittag et al., 2021). Também podem mensurar variáveis fisiológicas de pacientes, emitir registros eletrônicos e outros acessórios, inclusive, podem ajudar a salvar vidas. Soluções inteligentes baseadas em IoT melhoram o transporte e aceleram o tráfego, reduzem o consumo de combustível e priorizam cronogramas de reparos de veículos. Além disso, no ramo industrial, o monitoramento automático de sensores diagnóstica e promove a manutenção preditiva de ativos industriais, o que evita perdas financeiras devido à superestimação ou subestimação de parâmetros críticos. IoT permite construir estruturas para as chamadas casas, fazendas e cidades inteligentes. O que inclui o gerenciamento eficaz de manejo de resíduos, uso eficiente de energia, semáforos inteligentes, etc. O uso de tecnologias IoT também está associado a diversos aplicativos com milhões de usuários no

mundo como ifood, Uber, Airbnb, Mercado Livre, whatsapp, entre outros serviços (Rayes et al., 2019).

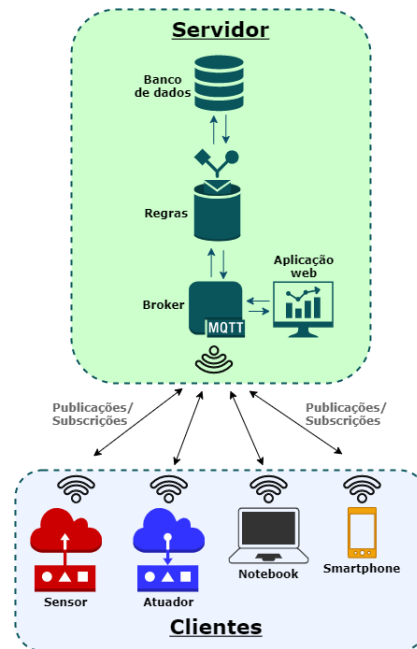
Para a implementação de redes IoT é necessária uma estrutura física em termos de recursos computacionais, estrutura de rede de comunicação e serviços digitais. Em relação estrutura física, um elemento central do sistema é o servidor. Este funciona como um nó de comunicação, no qual estão instalados diversos softwares que atuam de forma coerente, segura e responsiva, permitindo comunicação bidirecional entre muitos dispositivos conectados a essa estrutura. Para esta finalidade, usualmente são utilizados supercomputadores, computadores com alta capacidade e velocidade para processamento da informação ou servidores privados virtuais (Virtual Private Server, VPS). Estas soluções são particularmente viáveis em indústrias, empresas ou empreendimentos com poder financeiro. No entanto, as opções supracitadas para implementar um servidor em ambientes educativos e universidades resultam restritas às capacidades orçamentárias das unidades. Nesse contexto, surge o atual estudo, que visa propor um procedimento básico para a seleção e instalação de um servidor web baseado em uma placa Raspberry Pi. Esta estrutura de hardware de baixo custo, quando comparada com as outras opções supracitadas, facilita a implantação de redes IoT pelos próprios discentes durante atividades práticas de ensino. Este projeto se insere como uma etapa importante para estimular o desenvolvimento de habilidades nos estudantes em temas multidisciplinares como redes de computadores, sistemas digitais e programação. Almeja-se que a proposta seja útil para a aplicação de múltiplos conhecimentos em tecnologias digitais através de atividades práticas, utilizando ferramentas de hardware de baixo custo e programas gratuitos de código aberto.

MÉTODOS

Estrutura básica de uma plataforma IoT

O atual estudo apresenta um procedimento básico para instalar e executar os serviços básicos digitais em um servidor web. Um exemplo genérico de uma plataforma IoT é apresentado na Figura 1. Com esta estrutura é possível implementar o monitoramento de dados em tempo real e transmissão de dados de telemetria.

Figura 1 - Esquema geral da plataforma IoT para monitoramento de dados em tempo real, tratamento, armazenamento e visualização da informação.



Fonte: autoria própria

Esta rede genérica é baseada no funcionamento de vários subsistemas digitais operando em um servidor, dispositivos clientes e a comunicação entre os elementos da plataforma. No caso do servidor, na Figura 1 foram representados o banco de dados, as regras, o broker e a aplicação web. Em relação aos clientes, observam-se diferentes dispositivos IoT's conectados à rede, como um sensor, um elemento atuador, um smartphone e um computador pessoal. Para a conexão entre o servidor e os dispositivos é frequente utilizar algum padrão de comunicação sem fio como o IEEE 802.11 (wi-fi) ou IEEE 802.15.4 (Zigbee, ISA100.11a e 6LowPAN). A escolha de qual padrão usar depende da aplicação e do ambiente físico (Vasseur et al., 2010).

Como estrutura física de um servidor IoT, podem ser usados diversos tipos de computadores, desde os desktops e notebooks comuns até supercomputadores ou os computadores de tipo servidor propriamente dito. Outras alternativas mais baratas e de desempenho eficiente são as placas de Raspberry Pi ou BeagleBone. Inclusive, o servidor pode ser instalado na nuvem como um Servidor Privado Virtual (VPS) utilizando provedores de Amazon ou Google.

Em um servidor podem ser instalados diversas aplicações e programas para gerenciar os serviços do sistema IoT. O banco de dados é um elemento básico, pois armazena os dados gerados pelos clientes. Por exemplo, séries temporais a partir da resposta de sensores, informações de usuários para identificação e controle de acesso, assim como possíveis relações entre os dados dos dispositivos. Geralmente, no banco de dados a informação é organizada em tabelas, e para esta finalidade podem ser utilizados softwares como MySQL, MongoDB, e Microsoft SQL Server, dentre outras opções.

Outro componente comum nos servidores são as regras, úteis para a detecção de limites nas medições, geração de alarmes, envio de avisos, etc. Através deste subsistema é possível estabelecer comparações entre as informações do banco de dados e, conseqüentemente, emitir respostas via o broker. Um software que pode ser utilizado para estabelecer as regras é o Node.js, que é um ambiente de execução multiplataforma.

A aplicação web também constitui um serviço frequente nos servidores, devido a que funciona como interfase entre os usuários e os dispositivos conectados. Estas aplicações permitem tanto a visualização de dados e gráficos, quanto o acionamento de comandos de controle. Ou seja, a interação pode ocorrer em dois sentidos, por exemplo: 1- leitura de um sensor e transmissão da informação para o broker; retransmissão dos dados para a aplicação web; e visualização dos dados na aplicação web; 2- acionamento de um botão na aplicação web e transmissão da informação para o broker; retransmissão do comando para um dispositivo atuador; e variação efetiva no estado deste dispositivo como abertura ou fechamento de uma válvula, ventilador ou bomba de ar. O desenvolvimento de aplicações web implica o trabalho com os chamados front-end e back-end. O primeiro é responsável pela apresentação visual, organização e desenho gráfico, além de cuidar da experiência do usuário. Neste sentido, existem diversas ferramentas de programação como HTML, CSS e JavaScript. O back-end, entretanto, está relacionado com a lógica de interação entre a apresentação da aplicação web e o banco de dados, lidando também com as respostas a requisições dos usuários. Para implementar estas funções, o PHP é uma abordagem frequente.

O broker é um componente essencial e necessário da plataforma IoT, pois gerencia o fluxo de dados entre os subsistemas. O broker funciona como um intermediário, que controla e viabiliza de forma eficiente a transmissão/recepção da informação entre os

elementos do sistema. Neste âmbito, recomenda-se a comunicação através de MQTT (Message Queue Telemetry Transport), por ser um protocolo de transmissão de dados leve, assíncrono, baseado no paradigma de publicação/subscrição, que foi originalmente desenvolvido pela empresa IBM para aplicações de telemetria. MQTT é orientado a mensagens, sendo estas publicadas sob um determinado tópico e em um endereço específico. Desta forma, clientes subscritos a um ou mais tópicos receberão atualizações de outros clientes que tenham publicado sob os mesmos tópicos. Por exemplo, criando um tópico com o nome “temperatura” no broker, um cliente (dispositivo com um sensor de temperatura) pode publicar sob esse tópico os dados adquiridos nas medições, e outro cliente (dispositivo acessando a uma aplicação web) pode visualizar essa informação em tempo real. Dois dos broker disponíveis para desenvolvimento de sistemas IoT são Mosca, Mosquitto e EMQX.

RESULTADOS

Proposta de procedimento para instalação e configuração de aplicações em um servidor web baseado em uma placa Raspberry Pi

Nesta seção é apresentada uma proposta de procedimento para a instalação e configuração dos softwares em uma Raspberry Pi, que será utilizada como servidor web de uma rede IoT. Salienta-se que a aplicação específica desta proposta depende do contexto institucional, em termos de infraestrutura, capacitação docente e estrutura das ementas das disciplinas. Além disso, a implantação do servidor da rede IoT também dependerá de fatores como: recursos computacionais, disponibilidade de internet, tipo de acesso aos recursos da rede (administrador), entre outros. Mesmo que o atual estudo tenha uma abordagem de baixo custo e facilidade de execução, os pontos acima citados influenciam a profundidade do desenvolvimento do projeto.

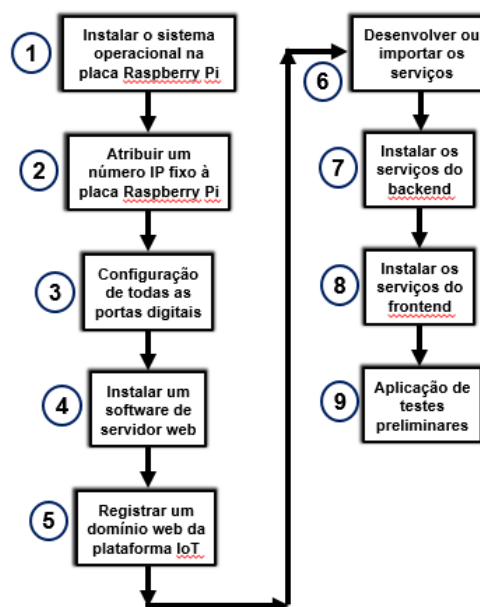
No processo, recomenda-se o uso de computadores pessoais para o desenvolvimento de código e para testes funcionais por parte de alunos e professores. Também é útil contar com uma tela, mouse e teclado por cada placa Raspberry Pi para facilitar a instalação inicial do sistema operacional. Esta placa, principalmente para aplicações de pequeno porte, oferece soluções eficientes, relativamente fáceis de implementar e de baixo custo. Também resulta atrativa em termos de eficiência energética, dimensões reduzidas, privacidade dos dados, e

versatilidade. Todas estas características tornam as placas Raspberry Pi uma opção ótima para aprendizado e desenvolvimento de servidores web locais. Por esses motivos esta ferramenta é recomendada para atividades práticas de ensino.

Para execução do projeto em sala de aula, o procedimento proposto pode ser realizado por grupos de estudantes, em que cada membro pode ir focando no aprendizado de ferramentas específicas da plataforma IoT, de acordo com a preferência individual, por exemplo. A finalização das atividades vai depender das habilidades e competências dos discentes, assim como da quantidade e complexidade dos serviços que sejam implementados efetivamente no servidor web e na plataforma IoT como um todo.

Na Figura 2, apresenta-se a proposta de procedimento geral para a instalação do servidor web em uma placa Raspberry Pi. A etapa 1 é constituída pela seleção do modelo da placa e pela instalação do sistema operacional. Sugerem-se, por exemplo, as seguintes versões de sistemas operacionais: Raspberry Pi OS (anteriormente Raspbian) para os modelos Raspberry Pi 3; Raspberry Pi OS (32/64-bit, baseado no Debian Bookworm), ou Ubuntu para os modelos Raspberry Pi 4; e Raspberry Pi OS (baseado no Debian 12 “Bookworm”) a Raspberry Pi 5.

Figura 2 – Fluxograma da proposta com as principais etapas de instalação e configuração de serviços no servidor web da Raspberry Pi.



Fonte: autoria própria

A etapa 2 deve garantir que o endereço IP da Raspberry Pi seja estático, ou seja, que a placa do servidor possua um número fixo de IP. Às vezes, o router ou outros elementos que gerenciam a rede operam com endereços IP dinâmicos (Dynamic Host Configuration Protocol, DHCP), o que dificulta ou inviabiliza o acesso aos dispositivos automaticamente. Para evitar este tipo de problema, é necessário operar com um endereço IP estático. No contexto das instituições de ensino, é frequente que este tipo de recursos de rede seja administrado por um departamento de Tecnologias de Informação (TI). Nestes casos, basta solicitar assistência à equipe de TI e utilizar o recurso disponível. A etapa 3, entretanto, visa a configuração das portas digitais de todas as aplicações que sejam instaladas no servidor. Um exemplo de portas utilizadas e a sua descrição são mostrados na Tabela 1. Cada uma destas portas digitais será mais detalhada através de um exemplo posterior.

Tabela 1 – Exemplo de algumas portas que podem ser utilizadas nas aplicações do servidor web da plataforma IoT.

Portas	Descrição da funcionalidade da porta
22	Raspberry Pi. Comunicação SSH entre a placa e o computador do desenvolvedor
80	Raspberry Pi. Conexão web http
443	Raspberry Pi. Conexão web https
1883	Broker. Porta padrão para o protocolo MQTT
1884	Broker. Protocolo MQTT com encriptação SSL
8093	Broker. MQTT através de WebSocket (navegador web)
8094	Broker. MQTT através de WebSocket com certificação (navegador web)
18083	Broker. Dashboard do EMQX

Fonte: autoria própria

A fase 4 da Figura 2 indica a instalação de um software de servidor web, por exemplo, Apache ou NGINX. Estes são amplamente utilizados como proxy reverso, atuando como intermediários entre os clientes conectados à plataforma e o servidor web (principalmente as aplicações do back-end). O proxy reverso recebe requisições dos clientes e as encaminha aos subsistemas internos do servidor, o que contribui para a segurança, a escalabilidade e o desempenho da rede. A etapa 5 consiste no registro do domínio web e, mesmo não sendo essencial, este passo é recomendado, pois favorece e simplifica o acesso dos usuários à plataforma. Um domínio web é o endereço único e exclusivo de um site na internet,

utilizado para acessar as páginas facilmente e sem a necessidade de escrever uma sequência numérica (número IP). Para cadastrar um domínio, existem diversas possibilidades, livres de custos ou não. No entanto, em ambientes de ensino, os profissionais da TI usualmente têm privilégios para gerenciar domínios, o que ajudaria nas atividades com os estudantes. Por exemplo, cada grupo de alunos dentro da turma poderia escolher um nome para o seu domínio web.

A etapa 6 corresponde ao desenvolvimento ou à importação das aplicações que serão utilizadas na rede IoT. A quantidade de serviços e o tipo dependem da arquitetura da rede que se pretenda implementar. Para atividades de ensino, onde geralmente a maioria dos alunos não possui conhecimentos especializados em IoT, é mais vantajosa desde o ponto de vista didático a implementação de redes simples (Figura 3). Neste sentido, poderia ser projetada como exercício inicial, porém completa, uma rede IoT com os seguintes elementos: um cliente sensor, um cliente de aplicação web (front-end) e o broker (back-end). As fases 7 e 8 do procedimento estão associadas à implementação desses serviços de front-end e back-end no servidor web. No primeiro caso, para implementar as aplicações web existem muitas possibilidades de softwares, além disso, há opções gratuitas de painéis pré-prontos como flatkit, etc. Estes permitem a montagem mais ágil e dinâmica de dashboards atrativos e multifuncionais. Em relação ao back-end, o EMQX é um software que ganhou popularidade nos últimos anos devido à sua alta eficiência para implementar serviços de broker. Por fim, como última etapa desta proposta, recomenda-se a realização de testes entre cada um dos componentes da rede, principalmente em termos de comunicação, já que a plataforma deve operar com diversos protocolos e em tempo real. Apesar de esta fase 9 estar formalmente colocada para concluir o procedimento, o mais natural seria ir realizando pequenos testes de funcionamento na medida em que vão se incorporando serviços à rede. Assim, somente restaria no final a realização de testes mais globais e direcionados à estabilidade de funcionamento no tempo, tanto do servidor web quanto da rede IoT.

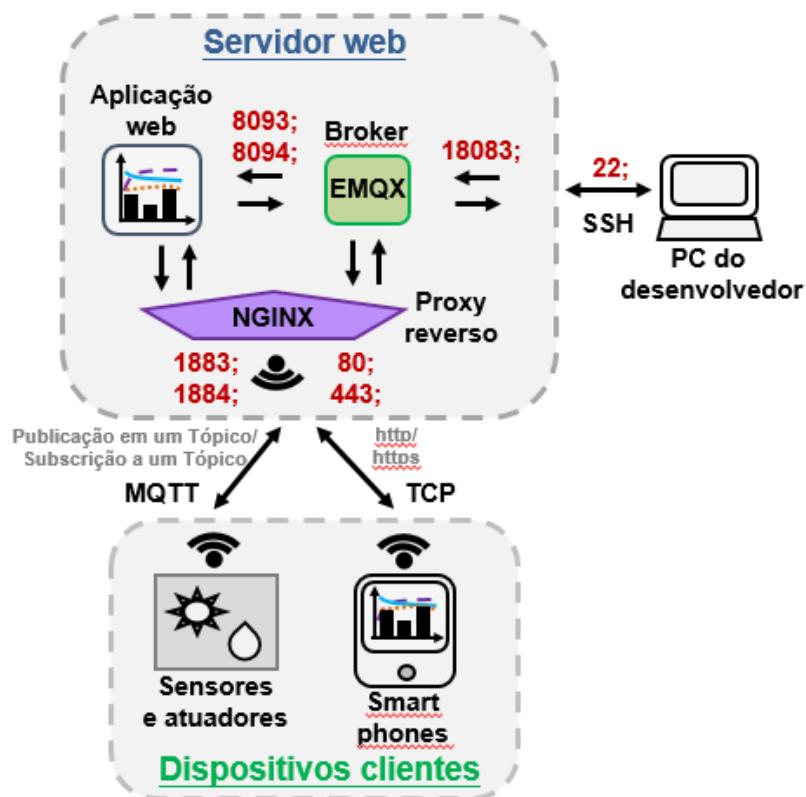
Exemplo de uma arquitetura básica de rede IoT utilizando um servidor web local rodando em uma Raspberry Pi.

Na Figura 3 é mostrado um exemplo elementar de uma rede IoT. Esta organização

dos componentes da plataforma pode ser usada como base para a realização de exercícios de ensino em sala de aula. Observam-se as aplicações utilizadas, a comunicação entre os nós, as portas digitais de cada elo, assim como os dispositivos clientes, e o computador do desenvolvedor (alunos e professores).

Quando um dispositivo sensor adquire um dado e o publica sobre um tópico, essa ação é executada através da comunicação MQTT entre esse cliente e o broker na Raspberry Pi. Para isso podem ser usadas a porta 1883 (padrão para o protocolo MQTT do broker) ou 1884 (protocolo MQTT do broker com encriptação SSL). Conforme ilustrado na Figura 3, o EMQX foi selecionado para implantar o serviço do broker. O NGINX foi o software configurado como proxy reverso da aplicação. O dispositivo sensor, entretanto, pode ser implementado na prática através de algum sistema de aquisição de dados simples baseado em uma placa NodeMCU ESP8266, por exemplo.

Figura 3 – Diagrama de uma arquitetura básica para uma rede IoT. Proposta com aplicações elementares instaladas no servidor web baseado em uma Raspberry Pi.



Fonte: autoria própria

Outro dispositivo cliente representado na Figura 3 é um aparelho celular acessando o domínio web desenvolvido. Isto pode ocorrer através de uma página http (menos segura) ou https (mais segura). Esta requisição entra na Raspberry Pi pela porta 80 (conexão web http) ou 443 (conexão web https).

Dentro do servidor web local, o EMQX será responsável por gerenciar a comunicação entre este serviço e a aplicação web ou entre ele e o computador do desenvolvedor. No primeiro caso, podem ser utilizadas as portas 8093 ou 8094, que são para comunicação WebSocket (navegador web) e WebSocket com certificado, respectivamente. Em relação ao intercâmbio de mensagens entre o broker e o computador do estudante, propõe-se o uso das portas 18083 (dashboard do EMQX) e 22 (comunicação SSH entre a Raspberry Pi e o computador do desenvolvedor). Este caminho será exclusivo do desenvolvedor e deverá ser necessariamente resguardado por identificação privada de usuário e senha.

CONCLUSÃO

Neste trabalho foi apresentado um procedimento geral para implementar um servidor web de baixo custo para uma rede IoT. A proposta incluiu um exemplo básico, que pode ser útil como guia didático para desenvolver atividades práticas de ensino, envolvendo conhecimentos em programação, tecnologia e redes de computadores. Espera-se que o trabalho possa promover e estimular a atuação dos alunos como desenvolvedores de sistemas digitais e não simplesmente como consumidores das tecnologias. Esta abordagem colocaria os estudantes como protagonistas no processo de aprendizado, incentivando as capacidades criativas e abrindo possibilidades de atuação futura, inclusive, em programas de pós-graduação e no mercado de trabalho. O atual estudo não constitui um passo a passo rígido para desenvolver um servidor web, e sim uma proposta geral que pode ser utilizada como base para, de acordo com o cenário específico da aplicação, selecionar, instalar e implementar uma rede IoT em atividades de ensino.

REFERÊNCIAS

KORESHOFF, T. L.; ROBERTSON, T.; LEONG, T. W.; Internet of things: A review of literature and products. In Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, OzCHI '13, pages 335–344, New York, NY, USA. ACM, 2013.

MOURSI, A. S.; EL-FISHAWY, N.; DJAHEL, S.; SHOUMAN, M. A. An IoT enabled system for enhanced air quality monitoring and prediction on the edge. *Complex & intelligent systems*, 7(6), 2021.

RAYES, A.; SALAM S. *Internet of Things. From Hype to Reality. The Road to Digitization. Second Edition.* Springer, 2019.

RODRIGO, I. M. S; FERREIRA, S. A. Protocolos de segurança para internet das coisas (IoT). *Revista Ibero-Americana de Humanidades, Ciências e Educação — REASE*, v. 11, n. 12, 2025.

SOARES, B. L. Segurança da informação em ambientes de internet das coisas (IoT): desafios, vulnerabilidades e estratégias de proteção. *Revista Ibero-Americana de Humanidades, Ciências e Educação — REASE*, v. 12, n. 13, 2026.

TOLK, A. Interoperability, composability, and their implications for distributed simulation: Towards mathematical foundations of simulation interoperability. In 2013 IEEE/ACM 17th International Symposium on Distributed Simulation and Real Time Applications, pages 3-9, 2013.

VASSEUR, J-P.; DUNKELS, A. *Interconnecting Smart Objects with IP: The next internet.* Morgan Kaufmann, 2010.

VORMITTAG, E. D. M; CIRQUEIRA, S. S. R.; WICHER, H.; SALDIVA, P. H. N. Análise do monitoramento da qualidade do ar no Brasil. *Estudos Avançados*, 35(102), 7-30, 2021.