

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA
SOUZA**

Etec DA ZONA LESTE

Análise e Desenvolvimento De Sistemas

Erick Ferreira Lima

Gustavo Rodrigues Leite Da Silva

Hernandes Arthur Da Silva Santos

**SmartShelf: automatizando o controle de estoque com tecnologia
RFID**

São Paulo

2025

Erick Ferreira Lima
Gustavo Rodrigues Leite Da Silva
Hernandes Arthur Da Silva Santos

**SmartShelf: automatizando o controle de estoque com tecnologia
RFID**

**Trabalho de Conclusão de Curso
apresentado à Etec [nome da escola],
como requisito parcial para a obtenção
do título de Técnico em [nome do
curso], sob orientação do(a)
Professor(a) [nome do orientador].**

São Paulo

2025

ERICK FERREIRA LIMA
GUSTAVO RODRIGUES LEITE DA SILVA
HERNANDES ARTHUR DA SILVA SANTOS

**SmartShelf: automatizando o controle de estoque com tecnologia
RFID**

**Trabalho de Conclusão de Curso apresentado à Escola Técnica Estadual
[Nome da Etec], como requisito parcial para a obtenção do título de Técnico
em [nome do curso], sob a orientação do(a) Prof.(a) [nome do orientador(a)].**

Aprovado em ____/____/____.

Banca Examinadora:

Prof.(a) [Nome do Orientador(a)]

Etec [Nome da Etec]

Prof.(a) [Nome do membro 1 da banca]

Etec [Nome da Etec]

Prof.(a) [Nome do membro 2 da banca]

Etec [Nome da Etec]

DEDICATÓRIA

Dedico este trabalho a Deus, que me deu força nos momentos mais difíceis, à minha família, pelo apoio incondicional, e aos meus amigos, que sempre estiveram ao meu lado durante essa jornada.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por me conceder saúde e sabedoria para concluir mais essa etapa.

Ao meu orientador(a) [Nome do Professor], pela paciência, dedicação e orientações ao longo da realização deste trabalho.

Aos meus colegas e professores da Etec [Nome da sua Etec], pelo companheirismo, apoio e conhecimento compartilhado.

À minha família, por todo o suporte emocional, incentivo e amor.

“Escolhe um trabalho de que gostes,
e não terás que trabalhar nem um dia
na tua vida.”

– Confúcio

AULAS PRÁTICAS NO ENSINO REMOTO: CONVERGÊNCIAS E DISRUPTURAS

Resumo

Este artigo busca levantar quais as metodologias utilizadas pelos professores das escolas técnicas nas aulas práticas oferecidas a distância, estabelecendo interlocução entre as necessidades de aprendizagem em oficinas e laboratórios transpostas para as ferramentas disponíveis no ensino remoto. Para tanto, realizou-se uma pesquisa online que levantou os pontos convergentes para as dificuldades impostas pela impossibilidade da oferta de aulas práticas presenciais, ao ponto que também apresentou as possibilidades de ensino e de aprendizagem propiciadas por essas ferramentas. Cabe destacar que os resultados obtidos advêm de práticas e experiências docentes em construção, por isso fica em evidência a sua importância no que diz respeito à validação do cenário atual, que precisa ser encarado como uma oportunidade de ruptura de padrões, conceitos e concepções sobre ensino e aprendizagem em aulas práticas. Nesse sentido, entende-se que o presente artigo pode contribuir para futuros debates e estudos sobre a oferta de Educação Profissional no modelo híbrido.

Palavras-chave: aulas práticas; metodologias ativas; ensino híbrido.

PRACTICAL CLASSES IN REMOTE TEACHING: CONVERGENCES AND DISRUPTIONS

ABSTRACT

This article aims to identify the methodologies employed by technical school teachers in remote practical classes, establishing a dialogue between the learning needs typically addressed in workshops and laboratories and the tools available in distance education. To this end, an online survey was conducted to highlight both the convergences regarding the difficulties imposed by the impossibility of offering in-person practical classes and the teaching and learning possibilities afforded by these tools. It is important to note that the results stem from ongoing teaching practices and experiences, underscoring their significance in validating the current scenario. This situation should be viewed as an opportunity to disrupt established patterns, concepts, and understandings of teaching and learning in practical classes. In this sense, this article seeks to contribute to future debates and studies on the delivery of Vocational Education in a hybrid model.

Keywords: practical classes; active methodologies; hybrid teaching

Lista de Ilustrações

Figura 1 — Placa ESP32 com anotações de componentes	15
Figura 3 — Estrutura básica de uma página em HTML	17
Figura 4 — Exemplo de uma tela HTML para cadastro de produtos.....	18
Figura 5 — Resultado do código HTML de uma página de cadastro de produtos. ...	19
Figura 6 — Exemplo de conexão entre HTML e CSS.	20
Figura 7 — Exemplo de escrita na CSS.	21
Figura 8 — CSS da tela de cadastro de produtos.	23
Figura 9 — Resultado da tela de cadastro de produtos usando HTML e CSS em conjunto.....	24
Figura 10 — Função juntar nome e sobrenome NodeJS	26
Figura 11 — Função juntar nome e sobrenome NodeJS	26

Lista de Abreviaturas e Siglas

CSS – Cascading Style Sheets (Folhas de Estilo em Cascata)

DNS – Domain Name System (Sistema de Nomes de Domínio)

FTP – File Transfer Protocol (Protocolo de Transferência de Arquivos)

HTML – HyperText Markup Language (Linguagem de Marcação de Hipertexto)

HTTP – HyperText Transfer Protocol (Protocolo de Transferência de Hipertexto)

HTTPS – HyperText Transfer Protocol Secure (Protocolo Seguro de Transferência de Hipertexto)

I/O – Input/Output (Entrada e Saída de Dados)

IDE – Integrated Development Environment (Ambiente de Desenvolvimento Integrado)

IoT – Internet of Things (Internet das Coisas)

JS – JavaScript (Linguagem de Programação para Web)

Node.js – Plataforma para execução de JavaScript no servidor

RFID – Radio-Frequency Identification (Tecnologia de Identificação por Rádio Frequência)

URL – Uniform Resource Locator (Localizador Uniforme de Recursos)

SUMÁRIO

1 INTRODUÇÃO	12
2 REFERENCIAL TEÓRICO.....	13
2.1 AUTOMATIZAÇÃO NO CONTROLE DE ESTOQUE	13
2.2 INTERNET DAS COISAS (IOT).....	14
2.3 ETIQUETA RFID	14
2.4 LEITOR RFID	14
2.5 ESP32	15
2.6 ARDUINOIDE.....	15
2.7 HTML	16
2.8 CSS.....	20
2.9 JAVASCRIPT.....	25
2.10 NODEJS.....	25
2.11 FIREBASE	27
REFERÊNCIAS.....	28
HOJE	ERRO! INDICADOR NÃO DEFINIDO.

1 INTRODUÇÃO

O aumento da popularidade de mercados autônomos no Brasil fez do país um dos protagonistas nesse setor. Esse aumento de popularidade é atribuído à facilidade e segurança promovidas por esse modelo de negócio, viabilizadas pelas inovações tecnológicas (LEÃO, 2024).

Dessa forma, o uso de novas tecnologias nesse setor passa a acarretar significativamente um aumento de eficiência, principalmente na parte logística (Rovaroto, 2024). Visando tal aumento de eficiência, podemos então, por meios de auxílios tecnológicos, aumentar a rastreabilidade de estoque, evitando a perda de vendas pela indisponibilidade de itens buscados pelos clientes, mesmo que o item em questão esteja disponível em estoque, mas indisponível na prateleira, ou mesmo notificando previamente o dono do mercado autônomo em questão para que o mesmo reponha determinado item, evitando escassez e eventuais prejuízos.

Esse aumento da rastreabilidade de produtos será possível devido à tecnologia de identificação por radiofrequência (RFID), tecnologia essa que possui em sua gama de aplicações, dispositivos como etiquetas emissoras de radiofrequência e leitores de radiofrequência. Por meio desses, poderemos aumentar a rastreabilidade, identificando a movimentação de saída e entrada de produtos de forma automática, disponibilizando essa informação para fácil uso dos gestores do sistema. Esse aumento de rastreabilidade, em conjunto com o monitoramento do gestor, aumentará a eficiência do sistema, reduzindo a necessidade de verificação manual constante no estoque, aumentando a eficiência do sistema.

Para alcançar tais objetivos, precisaremos investigar a fundo a tecnologia de RFID e as suas formas de implementação, assim como as feições ideais do sistema, que precisará ser de fácil uso e eficaz, tendo como primordial uma comunicação em tempo real entre leitor de estoque e sistema de gestão no computador que exibirá os dados ao gestor, maximizando a eficiência do sistema.

2 REFERENCIAL TEÓRICO

A atual seção visa analisar o cenário atual de pesquisa sobre mercados autônomos em língua portuguesa, visando apresentar de forma clara e respaldada a proposta da integração de um sistema baseado na tecnologia de identificação por rádio frequência (RFID), solucionador da falta de automatização no monitoramento de estoque em mercados autônomos.

2.1 Automatização No Controle De Estoque

No Brasil, o setor varejista enfrenta problemas de administração de estoque que acarretam em prejuízos para as empresas e em perda de qualidade da experiência do consumidor (KPMG, 2024). Segundo Wenceslau (2024), esses problemas podem ser atribuídos a imprecisão de processos orientados à ação humana ou ao mau funcionamento de softwares.

A automação no controle de estoque se torna essencial para o desenvolvimento e a diminuição de problemas de imprecisão em todos os setores. Os tópicos a seguir buscam explicar uma das possíveis soluções para o problema anteriormente mencionado. Com foco em mercados autônomos, a solução irá aumentar a autonomia na reposição de estoque pelos gerentes.

É importante notar também a importância dessa pesquisa para o incentivo à busca e desenvolvimento de soluções e inovações na área de mercados autônomos. Em vista do desenvolvimento e da iminente mudança na atual interação do cliente com o varejo causada pela implementação de novas tecnologias, pesquisas referentes a esse setor da indústria se tornam primordiais para a criação de uma base sólida de conhecimentos em língua portuguesa (Tortorelli, 2018).

2.2 Internet Das Coisas (IoT)

A definição de Internet of Things (IoT) ainda não chegou a um consenso estabelecido. Porém, segundo Magrani (2018), pode-se afirmar que dispositivos IoT possuem características de conectividade com a Internet e compartilhamento de informações entre si, possibilitando cenários de integração entre dispositivos e serviços. Essas características, portanto, possibilitam a criação de dispositivos voltados para a melhoria de processos cotidianos e conectados ao contexto geral da internet, como o que será apresentado ao decorrer dessa monografia.

2.3 Etiqueta RFID

De acordo com os estudos de Santini (2008 apud Oliveira, 2023), a estrutura de uma etiqueta RFID corresponde a um chip e uma antena envolvidos por algum material. A etiqueta faz contato com os sinais transmitidos por meio da antena do leitor RFID.

Consoante Costa (2018), a formatação das informações que estão presentes nas etiquetas podem ser recebidas nas seguintes formas: caracteres hexadecimais, decimais ou ASCII.

Conforme analisa Campos (2021), a etiqueta RFID passiva não possui alimentação própria e, por conta disso, a emissão de dados e a própria alimentação da etiqueta vêm das ondas magnéticas emitidas pela antena do leitor.

2.4 Leitor RFID

Segundo Santini (2008 apud Prediger; Freitas; Silveira, 2014), a comunicação entre o leitor RFID e a tag RFID é feita através de uma antena que pode transmitir essas informações e processá-las para uso em outros sistemas.

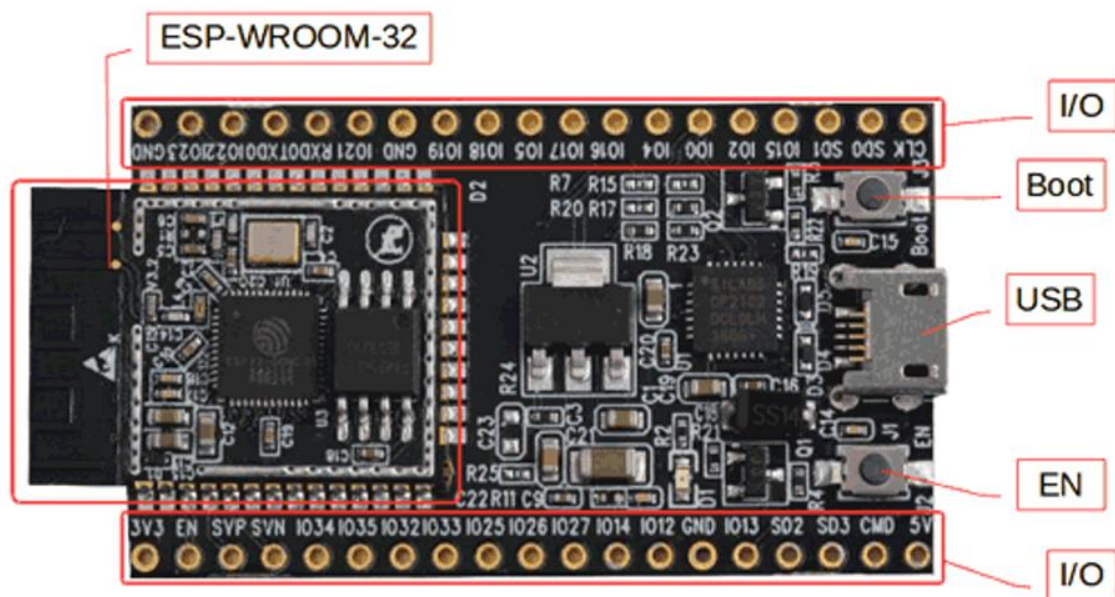
De acordo com Gonçalves (2025), ocorre a emissão de ondas de radiofrequência por meio da antena do leitor RFID.

Conforme Costa (2018), o leitor recebe os dados quando as etiquetas entram dentro da área de alcance do mesmo.

2.5 ESP32

A comunicação entre o leitor e o sistema será feita pela placa ESP32, sendo esse um microcomputador desenvolvido pela empresa chinesa Espressif Systems, capaz de executar o processamento de dados e se comunicar com outros meios via Wi-Fi ou Bluetooth (PET, 2021).

Figura 1 — Placa ESP32 com anotações de componentes



Fonte: PET. Introdução à Programação Embarcada. Minas Gerais: PET, 2021.

2.6 ArduinoIDE

Segundo Silva (2021 apud Damaceno; Fung; Sartori, 2024), o ESP32 é compatível com o ArduinoIDE, um Ambiente de Desenvolvimento Integrado (IDE), que é o meio pelo qual é passado as instruções de funcionamento para microcomputador.

2.7 C++

As instruções passadas para o ESP32 pelo Arduino IDE são feitas na linguagem de programação C++. De acordo com Wiener e Pinson (1991), C++, por ser orientada a objeto, é uma linguagem de programação extremamente eficaz quando se pensa em soluções mais humanizadas. A linguagem também tem um acesso melhor ao hardware, parte física, que vai possibilitar a criação de softwares, parte lógica, melhores.

2.8 HTML

Segundo SILVEIRA e PRATES (2001) O Hyper Text Markup Language (HTML) é a tecnologia para criação da estrutura de WebSites. A navegação de forma dinâmica é possível por conta do HyperText que através de links conecta as páginas entre si como explica DUCKETT (2016). Além disso, DUCKETT (2016) também destaca que Websites são compostos por diversos conteúdos como textos, links, imagens ou vídeos.

Somado a isso, CARDOSO (1999) afirma que para acessar aplicações web pelo navegador basta digitar o endereço (URL) na barra de pesquisa. Como explica DUCKETT (2016) o navegador entende o código HTML e desenha a tela que você vê.

Dessa forma, DUCKETT (2016) destaca que todo esse processo só é possível graças aos elementos HTML, cuja construção se baseia em caracteres colocados entre colchetes angulares (< >), indicando o início e o fim do elemento. Eles são fundamentais na criação de telas, onde cada elemento desempenha uma função específica na estrutura e organização dos conteúdos na página. Segundo SILVEIRA e PRATES (2001), as principais tags usadas para construção da estrutura de um documento HTML básico são:

- “<!DOCTYPE html>” indica o tipo de documento que esta sendo escrito e a versão do HTML. No HTML5, essa declaração é obrigatória e deve ser a primeira linha do código.
- “<HTML>” Representa o elemento raiz de uma página HTML. É composta por duas partes, que delimitam o início e o fim da codificação. Todo conteúdo deve estar entre elas.

- “<HEAD>” Armazena informações que não são diretamente exibidas nas páginas. Contém configurações essenciais para seu funcionamento.
- “<BODY>” Representa o corpo da página, onde ficam todos os elementos visíveis ao usuário, como textos, imagens, vídeos, botões e demais conteúdos interativos.

Um exemplo de utilização dessas tags é apresentado na figura 1 demonstrando sua estrutura básica.

Figura 2 — Estrutura básica de uma página em HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <!-- Todo Conteúdo desta Página -->
</body>
</html>
```

Fonte — Elaborado pelo autor.

Dando continuidade à criação do nosso código, a figura 2 mostra um exemplo de uma estrutura HTML para uma tela simples de cadastro de produtos.

Figura 3 — Exemplo de uma tela HTML para cadastro de produtos.

```
<!DOCTYPE html>
<html lang="pt-BR">

<head>
  <meta charset="UTF-8">
  <title>Cadastro de Produto</title>
</head>

<body>
  <h2>Cadastro de Produto</h2>
  <form action="#" method="post">
    <fieldset>
      <legend>Informações do Produto</legend>

      <label for="id">ID do Produto:</label><br>
      <input type="text" id="id" name="id"><br><br>

      <label for="nome">Nome do Produto:</label><br>
      <input type="text" id="nome" name="nome"><br><br>

      <label for="quantidade">Quantidade:</label><br>
      <input type="number" id="quantidade" name="quantidade"><br><br>

      <label for="marca">Marca:</label><br>
      <input type="text" id="marca" name="marca"><br><br>

      <input type="submit" value="Cadastrar Produto">
    </fieldset>
  </form>
</body>
</html>
```

Fonte — Elaborado pelo autor.

O código mostrado na figura 2, com as tags e dados utilizados, resulta na estrutura visual apresentada na figura 3:

Figura 4 — Resultado do código HTML de uma página de cadastro de produtos.

Cadastro de Produto

Informações do Produto

ID do Produto:

Nome do Produto:

Quantidade:

Marca:

Cadastrar Produto

Fonte — Elaborado pelo autor.

Em sequência, apresenta-se a solução do código, destacando, conforme explica DUCKETT (2016), o funcionamento e a função de cada elemento HTML utilizado.

- “<Form>” elemento que abrange todo formulário, utilizado para coleta das informações
- “<H2>” elemento de títulos. As tags H1, H2...H6. Definem diversos níveis de títulos. Onde “<H6>” indica a menor e “<H1>” o maior nível de título dentro de uma aplicação web.
- “<Fieldset>” agrupa os campos do formulário, organizando o conteúdo com uma borda ao redor deles e simplificando a visualização das informações
- “<Legend>” concede um título para o grupo de campos do <Fieldset>, auxiliando na organização e clareza do formulário.

- “<Label>” define um rótulo descritivo para um campo do formulário, ajudando na organização e acessibilidade.
- “<Input>” criam campos interativos no formulário, como textos, número e botão de envio. Seu comportamento é definido no atributo type, que determina o tipo de dado aceito naquele campo.
- “
” definem uma quebra de linha, organizando os elementos do formulário visualmente.

2.9 CSS

Como destaca KNIGHT (2018) a Cascading Style Sheets (CSS) é uma tecnologia usada para estilizar uma página web. Conforme explica DUCKETT (2016), é possível criar regras que informam como determinado elemento se comporta em uma página web, controlando atributos como cores, tamanhos, fontes e cor de fundo da página. O CSS é um dos principais fatores para uma experiência satisfatória ao usar uma página web, com o intuito, de que seja consumida da melhor forma possível conforme apresentado por EIS e FERREIRA (2012).

Como é abordado por DUCKETT (2016), de início, para estilizar uma tela HTML, é necessário realizar uma conexão com o arquivo CSS. O intermediário desta ligação é o elemento junto de seus atributos, quando adicionado dentro do no código HTML referenciando a folha de estilo, o HTML e o CSS conseguem trabalhar em conjunto. O trecho de código contendo a conexão desses dois documentos:

Figura 5 — Exemplo de conexão entre HTML e CSS.

```
<head>
  <title>Document</title>
  <link rel="stylesheet" href="styles.css">
</head>
```

Fonte — Elaborado pelo autor.

Como apresentado por DUCKETT (2016), esses são os papéis de cada atributo para que essa conexão aconteça:

- “<Link>” serve para comunicar ao navegador onde procurar o arquivo CSS.
- “href” mostra em específico onde está localizado a folha de estilo.
- “type” declara o tipo do documento que o link está referenciando.
- “rel” informa a relação entre a página HTML e o arquivo que referência, quando o link aponta para um documento CSS, o dado declarado deve ser stylesheet.

Conforme explica DUCKETT (2016), ao escrever a CSS, menciona que uma regra CSS é composta por duas partes, um seletor e uma declaração. Os seletores informam a qual tag HTML as regras serão aplicadas. Já as declarações elas incidam como os elementos serão modificados, as declarações também são divididas em duas partes, a propriedade e o seu valor atribuído. O bloco de código a seguir apresenta um exemplo de como escrever essas regras na folha de estilo.

Figura 6 — Exemplo de escrita na CSS.

```
body {  
    font-family: sans-serif;  
    background: ■ #f2f2f2;  
    padding: 30px;  
}  
  
form {  
    background: ■ #fff;  
    padding: 20px;  
    max-width: 350px;  
    margin: auto;  
    border-radius: 6px;  
}
```

Fonte — Elaborado pelo autor.

Conforme explica DUCKETT (2016), se fizermos a desramificação de cada propriedade e seu atributo com o intuito de entendermos seus papéis individualmente, ficaria assim:

- “Body” do HTML, indica tudo que está no corpo da página, realizando o papel de propriedade e recebendo suas declarações na CSS.
- “font-family” usada para declarar a fonte que será utilizada no texto de qualquer elemento HTML em que a regra CSS está agindo.
- “padding” possibilita definir um espaçamento entre o conteúdo e sua borda.
- “background” somente Background determina a cor que será utilizada em determinado elemento.
- “width” relata a largura de um elemento.
- “margin” controla o espaço entre blocos e elementos.
- “Border-radius” permite criar bordas arredondadas, seu valor declara o tamanho do raio.
- “Color” possibilita definir a cor do texto dentro de um elemento.

Prosseguindo, na figura 6 podemos observar o bloco de código completo para estilização do formulário de cadastro de produtos.

Figura 7 — CSS da tela de cadastro de produtos.

```
body {
  font-family: sans-serif;
  background: #f2f2f2;
  padding: 30px;
}

form {
  background: #ffffff;
  padding: 20px;
  margin: 0 auto;
  width: 100%;
  max-width: 350px;
  border-radius: 8px;
}

label {
  display: block;
  margin: 10px 0 5px 0;
}


input[type="text"],
input[type="number"] {
  width: 90%;
  padding: 8px;
  margin-bottom: 10px;
  border-radius: 4px;
  border: 1px solid #ccc;
}

input[type="submit"] {
  width: 100%;
  padding: 10px;
  background: #007bff;
  color: white;
  border-radius: 4px;
  border: none;
}
```

Fonte — Elaborado pelo autor

Somado a isso, com a união de todas as tecnologias apresentadas, o resultado da página de cadastro de produtos apresentado na figura 7:

Figura 8 — Resultado da tela de cadastro de produtos usando HTML e CSS em conjunto.



Informações do Produto

ID do Produto:

Nome do Produto:

Quantidade:

Marca:

Cadastrar Produto

Fonte — Elaborado pelo autor

2.10 JAVASCRIPT

Conforme explica SILVA (2010), com a criação de HTML, responsável pela estrutura dos websites, e o CSS, com o papel de estilizar a página para melhor experiência do usuário final, houve a necessidade de uma tecnologia com a função de adicionar interações para tais páginas.

Na visão de LEPSEN (2018), o Javascript criado pela Netscape com auxílio da Sun Microsystems, é a ferramenta encarregada pela criação das interações da aplicação web com o consumidor final. É a tecnologia capaz de designar funcionalidades aos elementos web, por meio dos campos de formulários, configurações gerais de uma página e interações para salvar nossos dados nos navegadores, o Javascript se comunica com o visitante deixando a página com ar mais dinâmico.

Segundo GRONER (2019), com essa ferramenta, é possível agir tanto na parte visual (Front-end) aplicando animações e manipulando elementos, quanto na parte lógica (Back-end) criando funcionalidades, que podem ser chamados de scripts. Javascript é utilizado principalmente para rodar scripts no lado do cliente, os responsáveis pela interpretação são os navegadores, que entendem e executam as funcionalidades.

2.11 NodeJS

De acordo com Pereira (2014), Node.js, criado em 2009 por Ryan Dahl e com ajuda inicial de 14 colaboradores, se destaca especialmente em aplicações que possuem muitas entradas e saídas de dados (Input/Output - I/O), onde, nesse cenário, ele consegue usufruir o máximo do poder de processamento dos servidores de forma produtiva.

Segundo Moraes (2023), Node.js utiliza o motor V8 da Google e permite criar rotas web usando diversos protocolos de redes (regras de como dispositivos devem se comunicar entre si) como, como HTTPS, DNS, FTP etc., permitindo a construção de sites e apps para sistemas operacionais iOS e Android.

Para Powers (2017), Node.js é uma ferramenta versátil e completa, podendo ser usada em diversos cenários e situações, graças ao seu ambiente com diversas funcionalidades e um bom alcance.

O exemplo a seguir é de um código que junta o nome e o sobrenome do usuário, exibindo ao final o nome completo. Muito usado em sistemas em que é necessário exibir o nome e sobrenome do usuário juntos. Abaixo segue uma descrição do funcionamento do código:

Figura 9 — Função juntar nome e sobrenome NodeJS

A screenshot of a code editor with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The code is written in a light-colored font and is numbered from 1 to 6 on the left side. The code defines a function named 'JuntarNomes' that takes two parameters, 'nome' and 'sobrenome', and returns the concatenation of the two strings with a space in between. The function is then called with the arguments 'Ananda' and 'Holanda'.

Fonte — Elaborado pelo autor

Abaixo segue uma descrição do funcionamento do código:

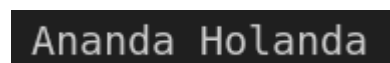
Linha 1: Declara uma função chamada “JuntarNomes” e, dentro dos parênteses, são solicitados dois parâmetros separados por vírgula (nome e sobrenome do usuário).

Linha 2: Retorna o nome e o sobrenome juntos através do sinal de “+”, com um espaço vazio entre eles, declarado entre aspas, que representa o espaçamento entre nome e sobrenome.

Linha 5: Chama a função passando como parâmetros o nome e o sobrenome separados por vírgula.

A imagem a seguir ilustra o que é exibido no terminal após a execução do código:

Figura 10 — Função juntar nome e sobrenome NodeJS

A screenshot of a terminal window with a dark background. The text 'Ananda Holanda' is displayed in a light-colored font, centered on the screen.

Fonte — Elaborado pelo autor

2.12 Firebase

Conforme afirma Andrade (2018 apud Silva; Santos Junior, 2023), sendo parte da infraestrutura Google, o Firebase é uma plataforma que busca ajudar desenvolvedores a construírem aplicações de forma mais rápida, performática e fácil.

Segundo a documentação oficial do Firebase (2025), Firebase é uma plataforma que permite criar projetos que podem usufruir de múltiplas funcionalidades, como hospedagem, banco de dados, autenticação, entre outros serviços.

Consoante Machado (2021), o Firebase entrega uma infraestrutura rica em recursos, possibilitando que o responsável pelo projeto foque em outras atividades, enquanto a plataforma adapta a infraestrutura de forma automática, com qualidade e escalabilidade.

2.13 UML, Linguagem De Modelagem Gráfica

A UML, segundo Guedes (2011), é uma representação visual para explicar processos ou conceitos envolvendo as etapas na produção de um software. O resultado visual segue conceitos estabelecidos no paradigma de orientação ao objeto que parte das linguagens de programação utilizam.

REFERÊNCIAS

CAMPOS, Matheus Santiago de. **Análise da viabilidade de implementação de etiquetas RFID na construção civil**. 2021. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Civil) – Universidade Tecnológica Federal do Paraná, Departamento Acadêmico de Construção Civil, Campo Mourão, 2021.

CARDOSO, Carlos. **HTML4**. Rio de Janeiro: Axcel Books, 1999.

COSTA, Alexsander Muniz da. **RFControl: sistema de gerência de estoque utilizando RFID**. 2018. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) – Universidade Federal de Ouro Preto, Instituto de Ciências Exatas e Aplicadas, Departamento de Computação e Sistemas, João Monlevade, 2018.

COSTA, Alexsander Muniz da. **RFControl: Sistema de gerência de estoque utilizando RFID**. 2018. Monografia (Graduação em Engenharia de Computação) – Instituto de Ciências Exatas e Aplicadas, Universidade Federal de Ouro Preto, João Monlevade, 2018.

DAMACENO, Donizeti; FUNG, Charles Way Hun; SARTORI, Rodrigo Vinicius. **Central de automação residencial de baixo custo com ESP32**. Curitiba: UNINTER, v. 4, n. 8, 2024.

DUCKETT, Jon. **HTML & CSS: projete e construa websites**. Rio de Janeiro: Alta Books, 2016.

EIS, Diego et al. **HTML5 e CSS3: com farinha e pimenta**. São Paulo, Brasil: Tableless, 2012.

GONÇALVES, Luiz Carlos. **Sistema de Controle de Acesso Físico por Dispositivos com Identificação por RFID e Autenticação por Biometria de Impressão Digital**. 2025. Monografia (Graduação em Engenharia de Controle e Automação) – Escola de Minas, Universidade Federal de Ouro Preto, Ouro Preto, 2025.

GOOGLE. Entender os projetos do Firebase. Disponível em: <https://firebase.google.com/docs/projects/learn-more?hl=pt-br>. Acesso em: 20 maio 2025.

GRONER, Loiane. **Estruturas de dados e algoritmos com JavaScript**. 2. ed. São Paulo: Novatec, 2019.

KPMG. Pesquisa Abrappe de Perdas no Varejo Brasileiro 2024. [s.l.], 2024. Disponível em: <https://kpmg.com/br/pt/home/insights/2024/11/pesquisa-abrappe-2024.html?utm_source=chatgpt.com>. Acesso em: 27 mai. 2025.

LEÃO, Leonardo. **Minimercados autônomos ganham força no Brasil**. Belo Horizonte: Diário do Comércio, 2024. Disponível em: <<https://diariodocomercio.com.br/negocios/minimercados-autonomos-ganham-forca-no-brasil/>>. Acesso em: 24 mai. 2025.

LEPSEN, Edécio Fernando. **Lógica de Programação e Algoritmos com JavaScript: UMA INTRODUÇÃO À PROGRAMAÇÃO DE COMPUTADORES COM EXEMPLOS E EXERCÍCIOS PARA INICIANTES**. 1. ed. São Paulo: Novatec, 2018.

MACHADO , Kheronn Khennedy. **Angular 11 e Firebase**: Construindo uma aplicação integrada com a plataforma do Google. Brasil: Casa do Código, 2021. 177 p. ISBN 978-85-7254-036-0.

MAGRANI, Eduardo. **A Internet Das Coisas**. Rio de Janeiro: FGV Editora, 2018

MORAES, William Bruno. **Construindo aplicações com NodeJS**. 4. ed. São Paulo: Novatec Editora, 2023. 304 p. ISBN 978-85-7522-879-1.

OLIVEIRA, Ana Beatriz de Souza Nogueira Rodrigues de. **Estudo paramétrico e análise de impedância de uma etiqueta RFID UHF passiva**. 2023. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Elétrica) – Instituto Federal da Paraíba, João Pessoa, 2023.

PEREIRA, Caio Ribeiro. **Aplicações web real-time com Node.js**. Brasil: Casa do Código, 2013. 186 p. ISBN 978-85-66250-14-5.

PET. **Introdução à Programação Embarcada**. Minas Gerais: Itajubá, 2021.

POWERS, Shelley . **Aprendendo Node**. 1. ed. Brasil: Novatec Editora, 2017. 312 p. ISBN 978-8575225400.

PREDIGER, Daniel. **Modelo de aplicabilidade de sistema RFID para rastreabilidade na indústria alimentícia**. 2014. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Universidade Federal de Santa Maria,

Frederico Westphalen, 2014. Disponível em:
<<http://repositorio.ufsm.br/handle/1/12810>>. Acesso em: 25 mai. 2025.

ROVAROTO, Isabela. **Como esta empresa de calçados deixou de perder vendas por falta de estoque em suas 340 lojas**. Exame, 2024. Disponível em:
<<https://exame.com/negocios/como-esta-empresa-de-calcados-deixou-de-perder-venda-por-falta-de-estoque-em-suas-340-lojas/>>. Acesso em: 24 mai. 2025.

SILVA, Antonio Gabriel Araújo; SANTOS JUNIOR, João Edinaldo Gomes dos. **Vacmonitor: uma aplicação para o monitoramento de vacinas utilizando Flutter e Firebase**. 2023. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Computação) – Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Campus Campina Grande, Campina Grande, 2023.

SILVA, Maurício Samy. **Javascript: Guia do Programador**. 1. ed. São Paulo: Novatec, 2010.

SILVEIRA, Marcelo; PRATES, Rubens. **HTML 4: guia de consulta rápida**. São Paulo: Novatec, 2001.

TORTORELLI, Henrique. **Estratégias de Negócios Internacionais: O caso da empresa Amazon.com**. Covilhã: Universidade Da Beira Interior, 2018.

WENCESLAU, Fernando. **Como a automação e a tecnologia preditiva estão mudando a gestão de estoque**. Santa Catarina: Economia SC, 2024. Disponível em:
<<https://economiasc.com/2024/07/25/como-a-automacao-e-a-tecnologia-preditiva-estao-mudando-a-gestao-de-estoque/>>. Acesso em: 25 mai. 2025.

WIENER, Richard S.; PINSON, Lewis J. **Programação Orientada para Objeto e C++**. São Paulo: Makron, McGraw-Hill, 1991.

Fowler, Martin. **UML Essencial Um breve guia para a linguagem-padrão de modelagem de objetos**. Porto Alegre : Bookman, 2007. Disponível em:
<[https://www.kufunda.net/publicdocs/UML%20Essencial%20\(Martin%20Fowler\).pdf](https://www.kufunda.net/publicdocs/UML%20Essencial%20(Martin%20Fowler).pdf)>
. Acesso em: 25 mai. 2025.

GLOSSÁRIO

ArduinoIDE (Integrated Development Environment): Ambiente de Desenvolvimento Integrado utilizado para programar microcontroladores como o ESP32, permitindo compilar e enviar código para o dispositivo.

Aulas práticas: Atividades presenciais ou remotas com foco na aplicação prática de conhecimentos teóricos, especialmente comuns em cursos técnicos.

Authentication: Serviço de autenticação de usuários oferecido por plataformas como o Firebase, utilizado para controle de acesso em aplicações web.

Automatização: Aplicação de tecnologia para aumentar a eficiência e reduzir a necessidade de ação humana em processos. Essencial no controle de estoque.

Back-end: Parte lógica de uma aplicação web, responsável pelo funcionamento interno, comunicação com banco de dados e execução de scripts do lado do servidor.

C++: Linguagem de programação de alto nível, de propósito geral, e é uma extensão da linguagem C. É utilizada para programar o ESP32 e possui características orientadas a objetos e acesso facilitado ao hardware.

Controle de estoque: Processo de administração de materiais e produtos armazenados. Pode sofrer perdas por erros humanos e falta de rastreabilidade; o projeto busca automatizá-lo usando RFID.

CSS (Cascading Style Sheets): Tecnologia usada para estilizar páginas web. Define regras de exibição como cores, tamanhos, fontes e disposição visual dos elementos.

Database: Serviço de banco de dados que armazena e organiza dados utilizados por aplicações web. No projeto, o Firebase oferece esse recurso.

Ensino híbrido: Modelo educacional que combina atividades presenciais e online, muito utilizado no contexto da Educação Profissional.

Escalabilidade: Capacidade de um sistema ou serviço (como o Firebase) de se adaptar ao aumento da demanda sem perda de desempenho.

ESP32: Microcontrolador da Espressif Systems com recursos de Wi-Fi e Bluetooth, compatível com o Arduino IDE. Utilizado para processar dados e controlar dispositivos físicos.

Etiqueta RFID: Dispositivo composto por um chip e uma antena, capaz de armazenar e transmitir dados quando ativado por um leitor RFID. Pode ser passiva (sem fonte de energia própria).

Firebase: Plataforma da Google para desenvolvimento de aplicações web e móveis. Oferece recursos como banco de dados em tempo real, hospedagem, autenticação e escalabilidade.

Front-end: Parte visual de uma aplicação web, com a qual o usuário interage. É construído com tecnologias como HTML, CSS e JavaScript.

Hosting: Serviço de hospedagem que armazena e disponibiliza um site na internet. O Firebase oferece hosting integrado ao projeto.

HTML (HyperText Markup Language): Linguagem de marcação usada para estruturar páginas web. Utiliza elementos entre colchetes angulares para definir conteúdo como textos, links e imagens.

IDE (Integrated Development Environment): Ambiente de Desenvolvimento Integrado usado para escrever, compilar e enviar códigos para dispositivos como o ESP32.

IoT (Internet of Things): Conceito que envolve a conexão de objetos físicos à internet, permitindo a troca de dados e a automação de processos.

JavaScript (JS): Linguagem de programação utilizada para adicionar interatividade a páginas web. Pode ser executada tanto no front-end quanto no back-end.

Leitor RFID: Equipamento que emite sinais de rádio e lê dados das etiquetas RFID próximas. Responsável por iniciar a comunicação com as etiquetas.

Mercados autônomos: Estabelecimentos sem atendentes, com controle automatizado de entrada, saída e pagamento, geralmente com o uso de tecnologias como RFID.

Metodologias ativas: Estratégias de ensino centradas na participação ativa dos alunos, com foco em experimentação, resolução de problemas e protagonismo no aprendizado.

Node.js: Plataforma que permite a execução de JavaScript no lado do servidor. Utiliza o motor V8 da Google e é usada para desenvolver o back-end de aplicações web.

Protocolos de redes: Conjunto de regras que definem como dispositivos se comunicam em uma rede, como HTTP, HTTPS, FTP e DNS.

Rastreabilidade de estoque: Capacidade de identificar e acompanhar a movimentação de itens no estoque, o que reduz perdas e aumenta a eficiência da gestão.

RFID (Radio-Frequency Identification): Tecnologia que permite identificar objetos por meio de radiofrequência, facilitando rastreamento e controle de estoque de forma automática.

Scripts: Conjunto de instruções escritas em linguagens como JavaScript, executadas para automatizar tarefas ou criar funcionalidades em sistemas web.

URL (Uniform Resource Locator): Endereço usado para acessar recursos na internet, como páginas web, imagens ou arquivos.