

# RELATÓRIO

**GUSTAVO BURGOS BITTENCOURT FIGUEIREDO -12723119076, ISAAC DIAS - 12724130145,  
JUAN PABLO DOS SANTOS RODRIGUES - 12723131891, NICOLLE BRASIL DOS SANTOS NERY -  
12723115108, VITORIA MARIA SILVA CARDOSO - 12723110370,**

## TEMA:

**Aplicação para captação de dados de venda de uma rede de loja  
de peças de computadores**

## Introdução

Nossa aplicação apresenta a implementação de um sistema de coleta de dados com ênfase em vendas e controle de estoque, utilizando tecnologias e práticas consolidadas de desenvolvimento. O tema foi introduzido pelo professor e o foco da venda escolhido pelos integrantes do grupo. Definimos o objetivo em venda de peças de computadores, pois imaginamos que, poderíamos fazer uma melhoria em processos de venda de produtos que fazem parte do mundo tecnológico.

Assim, nos aproximamos de elementos que utilizamos para o funcionamento de sistemas, além disso, o desenvolvimento de sistemas de detecção e gerenciamento de dados é uma demanda crescente na era da informação. O sistema foi projetado para gerenciar informações essenciais, como produtos, clientes e transações de vendas, permitindo um controle integrado entre o estoque e o setor financeiro. A escolha do MySQL foi reforçada pelo uso do Node.js, que, aliado ao framework Express.js, viabilizou a implementação de uma API RESTful para a comunicação entre os serviços. A aplicação também foi instalada em contêineres utilizando Docker, garantindo a consistência do ambiente de desenvolvimento e sua replicação.

Os objetivos deste relatório incluem descrever as etapas de planejamento, execução e validação do sistema, evidenciando os desafios encontrados e as soluções definidas. Além disso, buscamos demonstrar como as tecnologias desenvolvidas para alcançar um sistema funcional, escalável e alinhado às necessidades do projeto. A introdução do GitHub como ferramenta de controle de versão possibilitou o trabalho eficiente entre os membros da equipe, sendo um aspecto essencial para o sucesso do desenvolvimento.

## Fundamentação Teórica

Para desenvolvermos um sistema de captação de dados, inicialmente o ideal para o funcionamento é de fato o armazenamento desses dados. “A gestão de bancos de dados desempenha um papel crucial em aplicações modernas, permitindo o armazenamento e a recuperação eficiente de informações.” LAUDON, K. C.; LAUDON, J. P. Sistemas de informações gerenciais. 9. ed. São Paulo: Pearson Prentice Hall, 2011. Com base nisto, utilizamos o MYSQL: Banco de dados relacional usado para armazenar informações como produtos, clientes, vendedores e vendas. Em nossa aplicação, foi utilizada de forma que armazenasse os dados das vendas das peças com função de controle de estoque e financeiro. As linguagens de programação que foram utilizadas foram: JavaScript (Node.js): Para a implementação da API, manipulando requisições HTTP e integração com o banco de dados e SQL: Para a criação e gerenciamento do banco de dados relacional, incluindo a execução de consultas para

manipulação de dados. Os frameworks e bibliotecas, Express.js: Framework para construção da API RESTful, responsável por gerenciar as rotas e facilitar a criação de endpoints e mysql2: Biblioteca para conectar e interagir com o banco de dados MySQL, utilizada para executar consultas SQL e manipular dados. Também, ferramentas para Contêineres, como Docker: Usado para empacotar a aplicação e o banco de dados em containers, permitindo a execução consistente em diferentes ambientes de desenvolvimento, teste e produção. Outra tecnologia indispensável, qual foi necessária para a aplicação é RESTful API: A comunicação entre os serviços é feita através de APIs RESTful, utilizando HTTP para requisições e respostas, e é importante a adição desta tecnologia, pois possui, escalabilidade, flexibilidade e independência, de acordo com o site oficial da **AWS**, plataforma de serviços de computação em nuvem oferecida pela Amazon. E para controle de versão, gerenciando alterações no código-fonte facilitando a colaboração entre a equipe de desenvolvimento, foi válido o manuseamento do GitHub.

## Projeto de implementação

Para o entendimento do projeto, analisamos primeiramente a relação das entidades para arquitetura do banco de dados e qual relação obtinham para que quando buscássemos uma informação, teria uma resposta do sistema. Pois, a ideia do projeto é obter dados de venda dos nossos produtos, sendo assim, o código abaixo demonstra a importância de captar os dados do produto e da venda.

```
1 -- Inserir dados iniciais.sql
2
3 -- Inserir dados iniciais de produtos
4 INSERT INTO produtos (nome, preco, estoque) VALUES
5 ('Teclado', 100.00, 50),
6 ('Mouse', 50.00, 100),
7 ('Monitor', 200.00, 20),
8 ('Placa de Vídeo', 1500.00, 15),
9 ('Processador', 1200.00, 10),
10 ('Memória RAM 8GB', 300.00, 25),
11 ('SSD de 512 GB', 400.00, 30),
12 ('Fonte 500W', 200.00, 20),
13 ('Gabinete', 250.00, 10),
14 ('Mais frio', 80.00, 40);
15
16 -- Inserir dados iniciais dos vendedores
17 INSERT INTO vendedores (nome, email) VALUES
18 ('Vendedor 1', 'vendedor1@gmail.com'),
19 ('Vendedor 2', 'vendedor2@gmail.com');
20
21 -- Inserir dados iniciais de clientes
22 INSERT INTO clientes (nome, email, telefone, endereco) VALUES
23 ('João Silva', 'joao@gmail.com', '111111111', 'Rua A, 123'),
24 ('Maria Souza', 'maria@gmail.com', '222222222', 'Rua B, 456'),
25 ('Carlos Lima', 'carlos@gmail.com', '333333333', 'Rua C, 789'),
26 ('Ana Clara', 'ana@gmail.com', '444444444', 'Rua D, 101'),
27 ('Pedro Costa', 'pedro@gmail.com', '555555555', 'Rua E, 202'),
28 ('Luiza Martins', 'luiza@gmail.com', '666666666', 'Rua F, 303'),
29 ('Rafael Oliveira', 'rafael@gmail.com', '777777777', 'Rua G, 404'),
30 ('Juliana Pereira', 'juliana@gmail.com', '888888888', 'Rua H, 505'),
31 ('Roberto Ferreira', 'roberto@gmail.com', '999999999', 'Rua I, 606'),
32 ('Fernanda Alves', 'fernanda@gmail.com', '1010101010', 'Rua J, 707');
33
34 -- Inserir dados na tabela de vendas
35 INSERT INTO vendas (id_produto, id_vendedor, id_cliente, quantidade, total) VALUES
36 (1, 1, 1, 2, 299.99), -- Venda de 2 unidades do produto 1, realizada pelo vendedor 1 para o cliente 1, com total de R$ 299,99
37 (2, 2, 2, 1, 129.90), -- Venda de 1 unidade do produto 2, realizada pelo vendedor 2 para o cliente 2, com total de R$ 129,90
38 (3, 1, 3, 1, 79.50), -- Venda de 1 unidade do produto 3, realizada pelo vendedor 1 para o cliente 3, com total de R$ 79,50
39 (4, 4, 4, 5, 999.00), -- Venda de 5 unidades do produto 4, realizada pelo vendedor 4 para o cliente 4, com total de R$ 999,00
40 (5, 5, 5, 10, 999.00); -- Venda de 10 unidades do produto 5, realizada pelo vendedor 5 para o cliente 5, com total de R$ 999,00
```

## Considerações Finais

Conseguimos desenvolver o banco de dados, porém apresentamos dificuldades inicialmente, pois não estava sendo criado da forma correta, os dados não estavam sendo relacionados no código, após análises objetivas, finalizamos o que daria o marco inicial para o desenvolvimento do sistema. Priorizamos o funcionamento do sistema, para assim, produzir alinhamentos que deixasse-o harmônico, além disso, o Docker foi um empecilho para a fluidez acelerada da criação, dessa forma, optamos entender o funcionamento do mesmo para que não restasse erros. Finalmente, a aplicação estava em funcionamento e os dados correlacionados, entendemos a funcionalidade do Docker no sistema e finalizamos com o teste de funcionamento.

## Bibliografia

LAUDON, K. C.; LAUDON, J. P. Sistemas de informações gerenciais. 9. ed. São Paulo: Pearson Prentice Hall, 2011

<https://aws.amazon.com/pt/what-is/restful-api/>