

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE
SÃO PAULO**

ANA LÍVIA DA CONCEIÇÃO DE MELO

**ESTUDO DE CASO DE UM CONTROLE DE ESTOQUE EM
OBRAS UTILIZANDO MONGODB**

**CAMPOS DO JORDÃO
ANO 2024**

RESUMO

Este trabalho visa desenvolver um sistema de controle de estoque para um canteiro de obras utilizando o banco de dados não relacional MongoDB. A metodologia incluiu a coleta de requisitos através de entrevistas com gestores, a criação de um modelo de dados utilizando documentos JSON, e a implementação dos modelos de dados no MongoDB. Os resultados abrangem a implementação do sistema de gestão de estoque, demonstrando a flexibilidade e a eficiência do MongoDB para a gestão de materiais em ambientes de construção. Concluímos que o sistema proporcionou uma gestão precisa e eficiente dos materiais, otimizando os processos do almoxarifado e melhorando a eficiência operacional do canteiro de obras.

Palavras-Chave: Sistema, Gestão, Modelagem, MongoDB, Automatização de processos.

ABSTRACT

This work aims to develop an inventory control system for a construction site using the non-relational database MongoDB. The methodology included collecting requirements through interviews with managers, creating a data model using JSON documents, and implementing the data models in MongoDB. The results cover the implementation of the inventory management system, demonstrating the flexibility and efficiency of MongoDB for managing materials in construction environments. We concluded that the system provided accurate and efficient management of materials, optimizing warehouse processes and improving the operational efficiency of the construction site.

Keywords: System, Management, Modeling, MongoDB, Process automation.

LISTA DE FIGURAS

FIGURA 1 – Coleção Materiais	15
FIGURA 2 – Coleção Entrada	16
FIGURA 3 – Coleção Saída	16
FIGURA 4 – Coleção Fornecedores	16
FIGURA 5 – Consulta 1	17
FIGURA 6 – Consulta 2	17
FIGURA 7 – Coleção	17

LISTA DE SIGLAS

JSON JavaScript Object Notation

SUMÁRIO

1	INTRODUÇÃO	7
1.1	Objetivos	8
1.2	Justificativa	8
1.3	Aspectos Metodológicos	9
1.4	Aporte Teórico	10
2.	PROJETO PROPOSTO	10
2.1	Metodologia Aplicada	10
2.2	Considerações Iniciais	15
2.3	Funcionalidades	13
2.4	Notação Modelo	13
3	Resultados Obtidos	14
3.1	Regras de Negócio	15
3.2	Documentos Json	15
4	CONCLUSÃO	18
	Referencias	20

1 INTRODUÇÃO

A gestão eficiente de estoques é crucial em diversos setores, especialmente na construção civil, onde a disponibilidade de materiais pode influenciar diretamente o andamento e o sucesso das obras. Um controle rigoroso de entrada e saída de materiais é essencial para evitar desperdícios, assegurar a qualidade e garantir que os recursos estejam disponíveis no momento certo. Nesse contexto, o uso de sistemas informatizados para a gestão de estoques tem se tornado uma prática comum, proporcionando maior precisão e agilidade aos processos.

Nos últimos anos, os bancos de dados não relacionais, também conhecidos como NoSQL, têm se destacado como uma solução flexível e escalável para o armazenamento e gestão de grandes volumes de dados. Diferentemente dos bancos de dados relacionais tradicionais, que organizam dados em tabelas fixas com colunas e linhas, os bancos de dados NoSQL permitem a manipulação de dados em formatos mais flexíveis e dinâmicos, atendendo melhor às necessidades de aplicações modernas que lidam com dados não estruturados ou semi-estruturados.

Os bancos de dados não relacionais podem ser classificados em diferentes modelos, cada um com suas características específicas e adequadas para diferentes tipos de aplicações. O modelo de documentos, por exemplo, organiza os dados em documentos geralmente armazenados no formato JSON, BSON ou XML. Cada documento é uma coleção de pares chave-valor, que pode conter dados de diferentes tipos e estruturas aninhadas. Esse modelo é altamente flexível, permitindo que cada documento dentro de uma mesma coleção possa ter uma estrutura diferente, o que facilita a adaptação a mudanças nos requisitos de dados sem a necessidade de remodelar o banco de dados. Exemplos de sistemas de banco de dados que utilizam esse modelo incluem MongoDB, CouchDB e Amazon DocumentDB.

Outro modelo importante é o de chave-valor, que armazena dados como pares chave-valor, onde cada chave é única e associada a um valor que pode ser um dado simples ou uma estrutura mais complexa, como um objeto ou um vetor. Esse modelo é extremamente eficiente para operações rápidas de leitura e escrita, pois a busca é realizada diretamente pela chave. É amplamente utilizado em aplicações que requerem acessos rápidos e frequentes a dados, como caches e sistemas de sessão. Alguns dos sistemas de banco de dados mais conhecidos que adotam esse modelo são Redis, Amazon DynamoDB e Riak.

O modelo de colunas organiza os dados em colunas em vez de linhas, o que facilita a leitura e escrita de grandes volumes de dados de forma eficiente. Cada coluna pode ser armazenada de forma distribuída, sendo especialmente útil para operações de leitura massiva de dados, como aquelas realizadas em sistemas de análise de grandes volumes de dados. Este modelo é ideal para aplicações que exigem leituras de grandes volumes de dados de maneira eficiente, como sistemas de análise de dados e big data. Exemplos de bancos de dados que utilizam esse modelo incluem Apache Cassandra, HBase e Google Bigtable.

É o modelo de grafos que armazena dados em forma de grafos, usando vértices para representar entidades e arestas para representar relacionamentos entre essas entidades. Este modelo é ideal para aplicações que requerem a análise de relacionamentos complexos entre os dados, como redes sociais, motores de recomendação e sistemas de gerenciamento de redes. Bancos de dados como Neo4j, Amazon Neptune e OrientDB são exemplos de sistemas que utilizam o modelo de grafos, proporcionando excelente desempenho em consultas que envolvem múltiplos níveis de relacionamento.

Este relatório apresenta o desenvolvimento e implementação de um sistema de gestão de estoques utilizando o banco de dados não relacional MongoDB, que adota o modelo de documentos. A escolha do MongoDB foi motivada por sua capacidade de gerenciar dados de forma flexível e escalável, características essenciais para a gestão eficiente de estoques.

A metodologia seguiu uma abordagem estruturada, abrangendo desde a modelagem dos dados em documentos JSON até a implementação de coleções e a criação de consultas em MongoDB para extrair informações relevantes. Ao longo deste relatório, serão detalhadas as etapas do projeto, apresentados os modelos de dados e exemplos de consultas MongoDB.

1.1 Objetivos

O objetivo deste relatório é documentar o desenvolvimento e a implementação de um sistema de gestão de estoques para um almoxarifado de obra utilizando o banco de dados não relacional MongoDB. O sistema visa proporcionar um controle preciso dos materiais, desde o seu recebimento até a utilização no canteiro de obras, atendendo às necessidades específicas do setor de construção civil.

Este relatório busca fornecer uma visão do processo de desenvolvimento do sistema de gestão de estoques, demonstrando sua importância para a eficiência operacional no setor de construção.

1.2 Justificativa

A gestão eficaz de estoques é crucial para o sucesso de projetos de construção civil. Um controle inadequado dos materiais pode causar falta de insumos essenciais, atrasos nas obras, aumento de custos e desperdício por má conservação ou perda de validade. A implementação de um sistema de gestão de estoques informatizado é,

portanto, essencial para otimizar recursos, garantir a continuidade do trabalho e minimizar perdas financeiras.

O MongoDB foi escolhido como o banco de dados para este sistema devido a diversas vantagens estratégicas. Primeiramente, sua flexibilidade e escalabilidade permitem a expansão horizontal sem dificuldades, além de suportar volumes extensos de dados não estruturados, características cruciais para ambientes dinâmicos como canteiros de obras. Outro ponto crucial é o modelo de dados flexível baseado em documentos JSON, que facilita a adição de novos atributos sem necessidade de mudanças estruturais significativas, adaptando-se facilmente às mudanças nos requisitos de negócios. Em termos de desempenho, o MongoDB se destaca por oferecer consultas rápidas e eficientes, essenciais para operações como análise de estoque e geração de relatórios em tempo real. Além disso, o banco de dados é robusto em termos de segurança, com mecanismos avançados de autenticação, controle de acesso e criptografia de dados, garantindo um ambiente seguro e controlado para as informações sensíveis do sistema. A informatização do controle de estoques no canteiro de obras traz benefícios como redução de desperdício e automação de processos, garantindo uma gestão mais eficiente e precisa dos materiais.

1.3 Aspectos Metodológicos

O presente estudo adotou uma abordagem que combinou pesquisa bibliográfica e prática para o desenvolvimento do sistema de gestão de estoques. As etapas metodológicas incluíram várias fases fundamentais. Inicialmente, foi realizada a coleta de requisitos, onde as necessidades dos usuários foram identificadas e os requisitos funcionais e não funcionais do sistema foram claramente definidos. Em seguida, ocorreu a modelagem de dados, que consistiu na definição das coleções e documentos JSON para representar as entidades e seus relacionamentos dentro do sistema. A implementação no MongoDB seguiu, envolvendo a criação das coleções necessárias e a inserção de dados de teste, juntamente com a implementação de consultas específicas para a gestão eficiente do estoque. Por fim, o sistema foi submetido a uma fase de validação rigorosa, com testes e verificações detalhadas, visando garantir que todas as especificações fossem atendidas e que o sistema operasse de maneira eficaz no contexto dinâmico de um canteiro de obras.

1.4 Aporte Teórico

Para fundamentar teoricamente o desenvolvimento do sistema de controle de estoque utilizando MongoDB, foram consultadas diversas fontes especializadas na área de banco de dados NoSQL e, especificamente, em MongoDB. Uma das principais referências utilizadas é o trabalho de Kristina Chodorow, "MongoDB: The Definitive Guide", um livro que proporcionou uma compreensão aprofundada sobre as funcionalidades avançadas do MongoDB, incluindo técnicas para otimização de consultas e modelagem de dados flexível através de documentos JSON.

Outra referência crucial foi o livro de Kyle Banker, "MongoDB in Action", que não apenas explorou as capacidades fundamentais do MongoDB, como também discutiu casos de uso práticos e estratégias para escalabilidade horizontal, aspecto essencial para ambientes que lidam com grandes volumes de dados não estruturados.

Essas fontes não apenas embasaram o desenvolvimento técnico do sistema, mas também proporcionaram uma base sólida para a tomada de decisões arquiteturais e implementação de melhores práticas no contexto do MongoDB, garantindo assim que o sistema atendesse não apenas aos requisitos operacionais imediatos, mas também estivesse preparado para escalar e adaptar-se a futuras demandas do projeto.

2 PROJETO PROPOSTO

O projeto propõe desenvolver um sistema de controle de estoque baseado no MongoDB, customizado para atender às demandas específicas de um canteiro de obras na indústria da construção civil. A escolha do MongoDB é fundamentada em sua capacidade de lidar com grandes volumes de dados não estruturados e oferecer flexibilidade na modelagem de dados.

2.1 Metodologia Aplicada

A metodologia adotada foi a estruturada que se baseou inicialmente em uma revisão bibliográfica abrangente. Esta revisão permitiu uma compreensão aprofundada das melhores práticas em gestão de estoque e das características distintivas dos bancos de dados NoSQL, com ênfase no MongoDB. Com base nesse conhecimento teórico, procedemos à modelagem dos dados utilizando documentos JSON,

uma escolha que proporcionou flexibilidade na estruturação dos dados e facilitou ajustes conforme as necessidades do sistema evoluíam.

A implementação seguiu as diretrizes teóricas, enfatizando a integridade dos dados e a eficiência operacional. Desenvolvemos consultas no MongoDB para garantir um controle preciso da entrada e saída de materiais, bem como para a geração de relatórios sobre o status do estoque. Essa abordagem metodológica não apenas assegurou a robustez do sistema em termos de funcionalidade e desempenho, mas também possibilitou a adaptação contínua às demandas variáveis de um ambiente dinâmico como o de canteiros de obras.

2.2 Considerações iniciais

Este relatório apresenta o projeto de desenvolvimento de um sistema de controle de estoque otimizado para um canteiro de obras, utilizando MongoDB como o sistema de gerenciamento de banco de dados central. MongoDB foi escolhido devido à sua capacidade de armazenar dados de forma flexível em documentos JSON, o que se mostra ideal para lidar com as complexidades e demandas dinâmicas típicas de ambientes na indústria da construção civil. Este sistema será projetado para fornecer um controle preciso sobre a entrada e saída de materiais no canteiro de obras, assegurando uma visão clara e em tempo real do estoque disponível. Além disso, oferecerá funcionalidades avançadas de relatórios e consultas, facilitando a análise de dados e apoiando decisões estratégicas relacionadas ao planejamento e gestão de materiais.

Na implementação prática, a criação das coleções no MongoDB envolve definir a estrutura dos documentos JSON e inserir os dados relevantes. As consultas serão realizadas utilizando a linguagem de consulta do MongoDB, que suporta operações avançadas como consultas complexas, agregações de dados e indexação eficiente para otimizar o desempenho das buscas. Este sistema não apenas proporciona uma solução robusta para o armazenamento de dados, mas também facilita a gestão eficiente e escalável de estoques em ambientes dinâmicos como canteiros de obras, garantindo flexibilidade na modelagem dos dados e desempenho otimizado nas operações diárias do sistema.

O projeto visa desenvolver um sistema de controle de estoque eficiente e adaptável que possa gerir a entrada e saída de materiais em um canteiro de obras, facilitar o acesso e a manipulação de dados de estoque de forma que gestores possam ter uma visão clara e em tempo real do status dos materiais e proporcionar ferramentas de relatórios e consultas avançadas que auxiliem na tomada de decisão e no planejamento das necessidades de material.

O sistema de controle de estoque desenvolvido para um canteiro de obras utilizando MongoDB oferece diversas funcionalidades essenciais:

- **Gestão de Materiais:** Permite o registro de novos materiais com informações detalhadas, como descrição, unidade de medida, quantidade inicial disponível, além da capacidade de visualizar o estoque atual e detalhes específicos de cada material.
- **Controle de Entrada e Saída:** Facilita o registro de novas aquisições de materiais, especificando a quantidade adquirida, o fornecedor responsável e a data de entrada no estoque. Também registra materiais retirados, incluindo a quantidade utilizada, o destino e a data de saída.
- **Relatórios e Análises:** Possibilita a geração de relatórios detalhados sobre o status do estoque, incluindo histórico completo de entradas e saídas de materiais. Além disso, o sistema oferece notificações automáticas quando o estoque de algum material atinge um nível crítico definido, permitindo uma gestão proativa.
- **Gestão de Fornecedores:** Permite o registro e a manutenção das informações dos fornecedores de materiais, facilitando o acesso rápido e organizado às informações essenciais sobre cada fornecedor.

2.3 Funcionalidades

Na estrutura JSON (JavaScript Object Notation) que é um formato de texto leve usado para armazenar e transmitir dados estruturados de forma legível para humanos e fácil de interpretar para máquinas. Ele é amplamente utilizado em aplicações web e sistemas de banco de dados NoSQL, como o MongoDB, devido à sua simplicidade e flexibilidade.

Em contextos como o desenvolvimento de sistemas de controle de estoque para canteiros de obras, o JSON é utilizado para representar dados estruturados de materiais, entradas, saídas, fornecedores, entre outros. A estrutura básica do JSON consiste em pares chave-valor, onde a chave é sempre uma string e o valor pode ser um número, uma string, um booleano, um array, outro objeto JSON, ou null.

Na modelagem de dados para um sistema de controle de estoque em MongoDB, os documentos JSON podem ser utilizados para representar cada material, entrada de material, saída de material e informações de fornecedores. A flexibilidade do JSON permite adicionar novos campos conforme necessário, sem a necessidade de alterar toda a estrutura do documento, o que é especialmente útil em ambientes onde os requisitos e dados podem evoluir ao longo do tempo.

No MongoDB, os documentos são armazenados em coleções e podem ser consultados e manipulados usando a linguagem de consulta do MongoDB, que permite realizar operações complexas de busca, filtragem e agregação diretamente sobre os documentos JSON armazenados no banco de dados. Isso torna o JSON uma escolha ideal para sistemas que precisam lidar com dados dinâmicos e variáveis, como é o caso de um sistema de controle de estoque.

Essas funcionalidades são projetadas para melhorar a eficiência operacional e garantir um controle preciso e estratégico dos recursos materiais no ambiente dinâmico de um canteiro de obras

Nesta seção, exploraremos exemplos práticos de como os documentos JSON são utilizados para representar e armazenar informações essenciais para o controle de estoque em um canteiro de obras. Veremos como os documentos são estruturados para registrar materiais, entradas e saídas de estoque, informações de fornecedores e outros aspectos relevantes para a gestão eficiente de recursos em projetos de construção.

Ao longo dos exemplos, destacaremos não apenas a sintaxe básica do JSON, mas também como essa estrutura é aproveitada pelo MongoDB para oferecer flexibilidade na modelagem de dados e eficiência nas operações de consulta e análise.

2.4 Notação Modelo Banco de Dados

No desenvolvimento do modelo de dados, foi utilizada a estrutura de documentos JSON, onde cada entidade e relacionamento é representado como um documento. Esta notação facilita a flexibilidade e a expansão dos dados, permitindo uma representação clara e precisa das entidades e seus atributos dentro do sistema de banco de dados.

3. RESULTADOS OBTIDOS

O projeto de controle de estoque para um canteiro de obras, desenvolvido com MongoDB, resultou em um sistema robusto e eficiente que facilita a gestão de materiais e otimiza as operações diárias do almoxarifado. A seguir, detalhamos os principais resultados alcançados e apresentamos exemplos práticos de como o sistema funciona.

3.1 Regras de Negócio

Durante o levantamento inicial para definição das regras de negócio, foi fundamental abordar detalhes específicos que impactam diretamente a gestão de estoque em um canteiro de obras. Isso incluiu uma análise detalhada da vasta gama de materiais utilizados, como concreto, aço, madeira, entre outros, cada um com suas próprias especificações técnicas cruciais para a operação diária. Foi necessário entender as diferentes unidades de medida aplicáveis a cada tipo de material, desde metros cúbicos para concreto até toneladas para aço, garantindo que o sistema pudesse registrar e gerenciar com precisão as quantidades e tipos de materiais em estoque.

Além disso, exploramos a frequência e os volumes típicos de entradas e saídas de estoque, levando em conta não apenas a regularidade das entregas de materiais, mas também a variabilidade nas demandas conforme o progresso das obras. Aspectos críticos da gestão de fornecedores também foram discutidos detalhadamente, incluindo os critérios para seleção de novos fornecedores e a manutenção das informações de contato atualizadas. Esses detalhes foram essenciais para assegurar que o sistema projetado não apenas atendesse às necessidades operacionais imediatas, mas também fosse flexível o suficiente para se adaptar a mudanças futuras nas condições do canteiro de obras e nas demandas do projeto.

A análise dos processos de negócio existentes não só permitiu identificar necessidades atuais, mas também antecipar cenários futuros que poderiam influenciar o sistema de controle de estoque. Isso incluiu considerações sobre a escalabilidade do sistema para lidar com o crescimento contínuo do canteiro de obras, bem como flexibilidade para ajustes conforme novos materiais fossem introduzidos ou processos atualizados. As regras de negócio foram formalmente documentadas em colaboração estreita com os stakeholders para garantir uma compreensão abrangente de todas as perspectivas, resultando em um conjunto claro e completo de diretrizes que orientaram eficazmente o desenvolvimento do sistema de controle de estoque.

3.2 Documentos JSON

Nesta seção, apresentamos exemplos práticos de documentos JSON utilizados na modelagem de dados para o sistema de controle de estoque em canteiro de obras. Os documentos JSON são essenciais para representar de forma flexível e estruturada informações sobre materiais, entradas e saídas de estoque, fornecedores e outras entidades relevantes. Cada exemplo demonstra como o MongoDB é empregado para armazenar e gerenciar esses dados de maneira eficiente, adaptando-se às necessidades dinâmicas e variadas de um ambiente de construção civil. Esses exemplos ilustram a versatilidade do formato JSON para a gestão detalhada e precisa dos recursos essenciais ao desenvolvimento do projeto.

3.2.1 Coleção Materiais

Armazena informações sobre os materiais disponíveis no estoque.

```
1  {  
2      "_id": "mat001",  
3      "nome": "Cimento",  
4      "descricao": "Saco de cimento 50kg",  
5      "quantidade_disponivel": 100,  
6      "unidade_medida": "sacos",  
7      "fornecedor_id": "for001",  
8      "categoria": "Construção",  
9      "data_ultimo_restoque": ISODate("2024-06-10T09:00:00Z")  
10 }
```

FIGURA 1 - Coleção Materiais (Autor)

3.2.2 Coleção Entrada

Registra cada entrada de material no estoque.

```
12  {  
13    "_id": "ent001",  
14    "material_id": "mat001",  
15    "quantidade": 50,  
16    "data_entrada": ISODate("2024-06-15T09:00:00Z"),  
17    "fornecedor_id": "for001",  
18    "nota_fiscal": "NF123456",  
19    "preco_unitario": 20.00,  
20    "responsavel": "João Pereira"  
21  }
```

FIGURA 2 - Coleção Entrada (Autor)

3.2.3 Coleção Saída

Registra cada saída de material do estoque.

```
24  {  
25    "_id": "sai001",  
26    "material_id": "mat001",  
27    "quantidade": 10,  
28    "data_saida": ISODate("2024-06-20T11:00:00Z"),  
29    "destino": "Obra A",  
30    "responsavel": "José Silva",  
31    "motivo": "Uso em fundação"  
32  }
```

FIGURA 3 - Coleção Saída (Autor)

3.2.4 Coleção Fornecedores

Armazena informações sobre os fornecedores de materiais:


```

35  {
36      "_id": "for001",
37      "nome": "Fornecedor A",
38      "contato": {
39          "telefone": "11987654321",
40          "email": "contato@fornecedorA.com.br"
41      },
42      "endereco": {
43          "rua": "Rua Exemplo",
44          "numero": 123,
45          "cidade": "São Paulo",
46          "estado": "SP",
47          "cep": "01234567"
48      }
49  }

```

FIGURA 4 - Coleção Fornecedores (Autor)

3.2.5

Relatório

Entrada

Relatório de Entradas de um Material Específico, esta consulta fornece o histórico de entradas do material "mat001", incluindo data de entrada, quantidade e nota fiscal:

```

62  db.entradas.find(
63      { "material_id": "mat001" },
64      { "data_entrada": 1, "quantidade": 1, "nota_fiscal": 1, "_id": 0 }
65  )

```

FIGURA 5 - Consulta 1 (Autor)

3.2.6 Relatório Saída

Relatório de Saídas de um Material Específico, esta consulta retorna o histórico de saídas do material "mat001", incluindo data de saída, quantidade e destino:

```

69  db.saídas.find(
70      { "material_id": "mat001" },
71      { "data_saida": 1, "quantidade": 1, "destino": 1, "_id": 0 }
72  )

```

FIGURA 6 – Consulta 2 (Autor)

4 CONCLUSÃO

O sistema de controle de inventário desenvolvido utilizando MongoDB demonstrou ser uma solução eficaz e adaptável para gerenciar os recursos em um canteiro de obras. A flexibilidade da estrutura de documentos JSON proporcionou uma gestão ágil e escalável dos dados, facilitando o acesso e a atualização em tempo real. A implementação do sistema resultou em melhorias significativas na rastreabilidade, controle de estoque e manutenção preventiva, reduzindo erros operacionais e aumentando a precisão dos registros. A capacidade do MongoDB de lidar com grandes volumes de dados e a sua capacidade de escalar horizontalmente foram aspectos cruciais para suportar as necessidades dinâmicas do ambiente de construção.

Durante o desenvolvimento e a implementação, foram identificados desafios, como a integração de sistemas legados e o treinamento de usuários. Esses desafios foram abordados com a introdução de estratégias de migração gradual e programas de capacitação contínua. As lições aprendidas incluíram a importância de uma análise detalhada dos requisitos do usuário desde o início do projeto, bem como a necessidade de uma colaboração estreita entre equipes de desenvolvimento e usuários finais.

Recomendações para melhorias futuras foram elaboradas com base na experiência adquirida. Elas incluem a expansão das funcionalidades de relatórios e análises preditivas para melhorar o planejamento e a tomada de decisões, além da exploração de soluções de IoT (Internet das Coisas) para monitoramento em tempo real de ativos. Essas iniciativas visam não apenas aprimorar o sistema atual, mas também prepará-lo para acompanhar as demandas futuras de um setor cada vez mais digitalizado.

Em suma, o uso do MongoDB não apenas otimizou a gestão de inventário, mas também estabeleceu uma base sólida para a eficiência operacional e a sustentabilidade dos recursos no ambiente dinâmico de um canteiro de obras. A contínua evolução e adaptação são essenciais para garantir que o sistema permaneça relevante e eficaz no apoio aos objetivos estratégicos da empresa no longo prazo.

REFERÊNCIAS

A. LIVROS:

BANKER, Kyle. **MongoDB em Ação**. São Paulo: Novatec, 2012.

BANKS, Chris. **Mastering MongoDB 4.x**. Packt Publishing, 2019.

CHODOROW, Kristina. **MongoDB: The Definitive Guide**. O'Reilly Media, 2013.

DOUGLAS, Craig. **The MongoDB Workshop**. Packt Publishing, 2020.

MONGOBOOK. **MongoDB: Guia Prático para Desenvolvedores**. Ed. TechPress, 2021.

B. ONLINE:

MongoDb. **Dados incorporados versus referências**. Disponível em: <https://www.mongodb.com/> Acesso em: 20 jun 2024.