

# Desenvolvimento de Aplicativo Móvel para Contagem de Produtos com Coletores de Dados: Otimização da Correção de Saldo de Estoque

Ivaney Vieira de Sales<sup>1</sup>, Ricardo Martins Ramos<sup>2</sup>

<sup>1</sup>Instituto Federal de Educação, Ciência e Tecnologia do Piauí  
R. Álvaro Mendes, 94 – Centro (Sul) – 64.000-040 – Teresina – PI – Brasil

<sup>2</sup>Instituto Federal de Educação, Ciência e Tecnologia do Piauí  
R. Álvaro Mendes, 94 – Centro (Sul) – 64.000-040 – Teresina – PI – Brasil

**Resumo.** *Este trabalho apresenta o desenvolvimento de um aplicativo móvel utilizando coletores de dados, focado na contagem de produtos para a correção de saldo de estoque em depósitos. A metodologia Kanban foi empregada para o levantamento de requisitos, permitindo um desenvolvimento ágil e flexível. O aplicativo foi desenvolvido utilizando Flutter para a camada de apresentação e Spring Boot para a camada de serviços, com banco de dados MySQL. As principais funcionalidades incluem a integração com sistemas de gestão existentes, rastreamento de movimentação de produtos, leitura de códigos de barras e alerta automatizados. A interface foi projetada para ser intuitiva, facilitando o uso por gerentes e auditores. Durante o desenvolvimento, foi dado enfoque à experiência do usuário, utilizando o Figma para prototipação e testes de usabilidade. Os resultados indicam que a adoção de tecnologias modernas pode aumentar significativamente a eficiência do processo de inventário, reduzindo erros e melhorando a gestão de estoques. Conclui-se que, além da tecnologia, é essencial estabelecer processos bem definidos e fornecer treinamento adequado aos usuários para garantir a eficácia do sistema. O estudo destaca a importância de uma abordagem integrada, combinando tecnologia com práticas de gestão bem estruturadas para otimizar o controle de inventário em depósitos.*

**Palavras-chave:** *Gestão de Inventário. Aplicativos Móveis. Coletores de Dados, Metodologia Kanban.*

**Abstract.** *This paper presents the development of a mobile application using data collectors, focused on product counting for stock balance correction in warehouses. The Kanban methodology was employed for requirements gathering, allowing for agile and flexible development. The application was developed using Flutter for the presentation layer and Spring Boot for the service layer, with a MySQL database. Key features include integration with existing management systems, product movement tracking, barcode scanning, and automated alerts. The interface was designed to be intuitive, facilitating use by managers and auditors. User experience was a focal point during development, with Figma used for prototyping and usability testing. Results indicate that the adoption of modern technologies can significantly enhance inventory processes, reducing errors and improving stock management. It is concluded that, in addition to technology, well-defined processes and adequate user training are*

*essential for the system's effectiveness. The study emphasizes the importance of an integrated approach, combining technology with well-structured management practices to optimize inventory control in warehouses.*

## **1. Introdução**

No varejo brasileiro, erros de estoque impactam negativamente a eficácia dos processos e a satisfação do cliente. O desafio reside na necessidade de manter um equilíbrio delicado entre a oferta e a demanda, garantindo que os produtos certos estejam disponíveis no momento certo. Nesse contexto, a implementação de um processo de inventário assume um papel estratégico e indispensável.

Realizar inventários é crucial para garantir a precisão das informações de saldo de estoque. Erros no registro de transações e no manuseio físico do estoque podem causar discrepâncias entre o estoque registrado e o real, que só são corrigidas durante verificações físicas esporádicas. Na prática, diversas transações aumentam a possibilidade de erros, resultando em registros imprecisos de estoque. As causas comuns incluem erros de digitação, contagem incorreta de produtos, falhas em registrar corretamente produtos danificados ou destruídos, retirada e retorno de itens sem a devida correção nos registros, atrasos na atualização dos registros após transações e roubos de estoque, frequentes no varejo e também presentes em ambientes industriais e comerciais [Slack et al. 2018].

Segundo a pesquisa realizada por [Silva and Kloeckner 2022], a adoção de inventários cíclicos representa uma prática estratégica que não apenas melhora a acurácia e a gestão dos estoques, mas também promove a eficiência operacional e a redução de custos, fortalecendo a competitividade da empresa no mercado.

Um aplicativo de gestão de inventário emerge como uma solução indispensável para o controle e a organização eficaz do estoque em depósitos. Projetado cuidadosamente para otimizar o gerenciamento de produtos, contagem física, auditoria e geração de relatórios, este aplicativo abrange diversas necessidades cruciais dos gerentes, auditores e equipes de controle de inventário. O aplicativo para dispositivo móvel proporciona funcionalidades específicas para auditores fazerem as contagem de produtos no depósito.

Para maximizar a utilidade do aplicativo, é crucial considerar a integração fluida com sistemas existentes na empresa, garantindo uma operação conjunta e eficiente. Além disso, características como rastreamento de movimentação de produtos, alerta automatizados, segurança robusta, histórico detalhado de alterações e suporte a tecnologias inovadoras, como leitura de códigos de barras, são incorporadas para aprimorar ainda mais a eficácia do sistema.

Os objetivos deste projeto incluem a implementação de um aplicativo de gestão de inventário, com ênfase na contagem e auditoria de estoques em depósitos, com o propósito de facilitar o controle e a organização dos produtos. O aplicativo deve integrar-se de maneira eficiente com os sistemas de gestão já utilizados pela empresa, assegurando a consistência e a atualização em tempo real dos dados de estoque. O escopo do aplicativo é claramente definido, sendo desenvolvido para atender às regras de negócio específicas de uma empresa em particular. Não é o objetivo do projeto criar um aplicativo que seja compatível com os diversos tipos de inventário existentes; o desenvolvimento será focado exclusivamente nas necessidades e processos da empresa em questão. Serão incorpora-

das funcionalidades como leitura de códigos de barras, rastreamento da movimentação de produtos e alerta automatizados, visando aprimorar a eficiência do processo de inventário. Adicionalmente, é imperativo criar uma interface amigável e intuitiva para auditores e gerentes, proporcionando uma experiência de usuário eficiente e adaptada às exigências da empresa. Finalmente, é essencial garantir que os usuários recebam treinamento adequado e suporte técnico contínuo para assegurar uma adoção eficaz do sistema.

## **2. Metodologia**

### **2.1. Abordagem de desenvolvimento**

Os métodos ágeis corrigem deficiências da engenharia de software tradicional. Embora ofereçam benefícios significativos, não são universais e não contradizem completamente as práticas confiáveis de engenharia de software. Eles podem ser aplicados como uma abordagem geral para todos os tipos de projetos de software [Pressman and Maxim 2021].

Na economia atual, é frequentemente difícil prever o desenvolvimento de sistemas computacionais, como aplicativos móveis. Mudanças rápidas ocorrem no mercado, as demandas dos consumidores são alteradas, e novas ameaças competitivas surgem sem aviso. Muitas vezes, é impraticável estabelecer completamente os requisitos antes do início do projeto. É crucial ter agilidade o bastante para se adaptar a um ambiente de negócios dinâmico [Pressman and Maxim 2021].

A metodologia escolhida foi o Kanban, amplamente adotada devido à sua capacidade de visualizar e gerenciar eficientemente fluxos de trabalho. Proporciona transparência sobre o progresso das tarefas e limita o trabalho em andamento, permitindo que a equipe se concentre em concluir tarefas antes de iniciar novas.

Na aplicação do Kanban para o levantamento de requisitos, é fundamental adotar práticas ágeis e integrar processos de iteração e feedback para assegurar um desenvolvimento flexível e eficiente. Inicialmente, compreender as necessidades dos envolvidos é essencial. Esse entendimento foi obtido por meio de entrevistas e questionários, fornecendo uma visão clara das expectativas e exigências do sistema. Com base nessa compreensão, foi elaborado um backlog de requisitos, no qual cada item representa uma funcionalidade ou necessidade específica. Este backlog é então priorizado conforme discutido em reuniões com a equipe de desenvolvimento e as partes interessadas no projeto.

Apesar do Kanban não utilizar ciclos de desenvolvimento fixos, foram estabelecidas reuniões semanais com as equipes de desenvolvimento, permitindo que cada item seja movido através das fases do quadro Kanban: “A Fazer”, “Em Andamento” e “Concluído”.

Reuniões regulares de revisão e retrospectiva foram realizadas para avaliar o progresso e obter feedback das partes interessadas. Esse feedback contínuo permite ajustar os requisitos e o desenvolvimento conforme necessário. O backlog de requisitos é refinado com base nas observações das iterações, incluindo a adição de novos requisitos e a remoção de itens desatualizados. Documentar mudanças e manter uma comunicação clara entre a equipe e as partes interessadas é essencial para alinhar os requisitos e o progresso do projeto.

Em resumo, a combinação da metodologia Kanban com práticas ágeis e processos de iteração e feedback permite um levantamento de requisitos adaptável e eficiente,

garantindo que o desenvolvimento responda rapidamente às mudanças, enquanto mantém um controle rigoroso sobre o progresso e a qualidade do produto final.

## **2.2. Ferramentas e tecnologias**

### **2.2.1. Camada de Apresentação**

No desenvolvimento da camada de apresentação, foi utilizado o Figma (<https://www.figma.com>) para a criação de protótipos de design e interfaces de usuário, permitindo uma visualização clara e interativa do aplicativo antes do desenvolvimento propriamente dito. O Figma é uma ferramenta de design colaborativa baseada na web, empregada para a criação de interfaces de usuário, protótipos e gráficos vetoriais. A ferramenta facilita a comunicação e a iteração entre designers e desenvolvedores em tempo real, assegurando que todos estejam alinhados durante todo o processo de design.

Para a criação da interface do aplicativo, foi empregado o Flutter na versão 3.22 (<https://flutter.dev>), um kit de desenvolvimento de software criado pelo Google. O Flutter possibilitou a construção de aplicativos nativos de alto desempenho para iOS, Android, web e desktop a partir de uma única base de código. Utilizando a linguagem Dart na versão 3.4 (<https://dart.dev>), o Flutter é renomado por sua capacidade de criar interfaces de usuário belas e interativas de forma ágil, proporcionando uma experiência de usuário consistente e responsiva em múltiplas plataformas.

O Android Studio versão 2024.1.1 (<https://developer.android.com/studio>) foi utilizado para desenvolver, testar e depurar o aplicativo. Esta ferramenta oferece um conjunto completo de recursos para o desenvolvimento com Flutter, incluindo um editor de código, ferramentas de depuração e um emulador integrado.

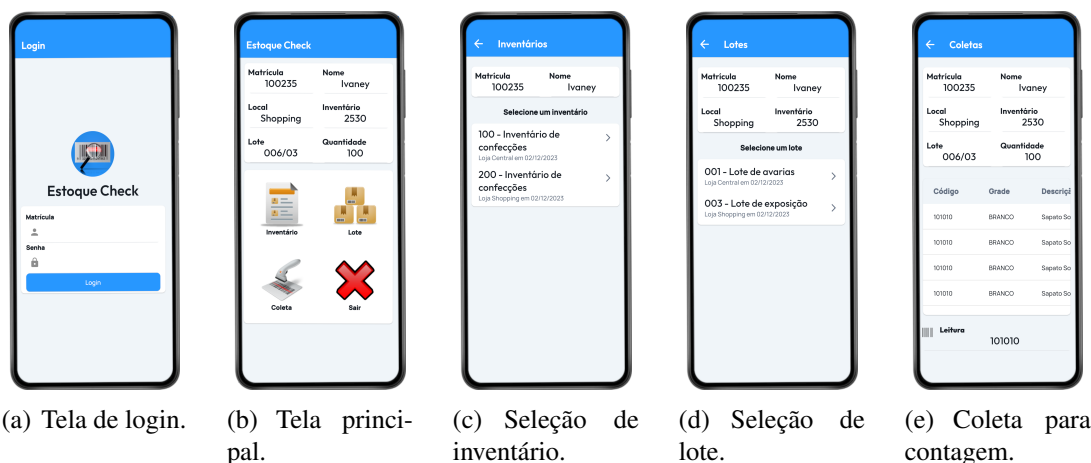
### **2.2.2. Camada de Serviços**

O camada de serviços inclui uma API RESTful, que realiza a integração do aplicativo com o sistema de gestão da organização. Foi desenvolvido na linguagem Java, versão 21 (<https://www.oracle.com/java/>), utilizando o framework Spring Boot, versão 3.2 (<https://spring.io/projects/spring-boot>). O IntelliJ IDEA, versão 2024.1.4 (<https://www.jetbrains.com/idea/>), foi empregado como ambiente de desenvolvimento integrado para a escrita, depuração e teste do código backend. Adicionalmente, o MySQL, versão 8.3 (<https://www.mysql.com>), foi selecionado como sistema de gerenciamento de banco de dados, garantindo armazenamento e recuperação de dados de forma eficiente e segura.

## **3. Resultado e Discussão**

### **3.1. Funcionalidades do aplicativo**

Antes da realização da contagem física dos produtos pelos auditores, munidos de coletores, é responsabilidade do administrador do inventário efetuar a configuração adequada. O inventário deve ser registrado com as seguintes informações: nome, período de realização (data de início e término), identificação do depósito onde ocorrerá a contagem e seleção dos produtos a serem inventariados. A seleção pode ser feita com base no departamento, fornecedor dos produtos ou por uma lista de produtos específicos.



**Figura 1. Layout de telas para o aplicativo**

A figura 1 apresenta o layout das telas do aplicativo conforme descrito a seguir:

- **Tela de Login:** Esta tela destina-se à autenticação do usuário no aplicativo. O usuário deve inserir seu nome de usuário e senha válidos, que correspondem às credenciais de acesso do sistema de gestão da empresa.
- **Tela Principal:** A tela principal apresenta ao usuário as opções para iniciar uma nova contagem de inventário ou retomar uma contagem já em andamento.
- **Seleção de Inventário:** Nesta tela, o usuário visualiza a lista de inventários disponíveis para contagem e seleciona aquele que deseja realizar. Após a seleção do inventário, a lista de produtos associada é carregada no coletor de dados, permitindo o uso offline caso o dispositivo perca a conexão.
- **Seleção de Lote:** A tela de seleção de lote apresenta ao usuário a lista de lotes disponíveis dentro do inventário escolhido. O usuário deve selecionar o lote que deseja contar. Um lote é definido como um conjunto de produtos agrupados em um espaço físico específico, como um palete, uma gôndola ou uma estante. Cada lote é identificado por um código e um código de barras. O auditor pode selecionar o lote desejado lendo o código de barras com o coletor.
- **Coleta para Contagem:** Nesta tela, o usuário visualiza os produtos presentes no lote selecionado e registra a quantidade contada de cada item. Ao finalizar a contagem, os dados coletados são enviados ao sistema de gestão da empresa. Nesse momento, a situação do lote é atualizado para “fechado”, podendo ser reaberto apenas pelo administrador do sistema.

Durante a contagem, o administrador monitora o progresso das contagens por meio de um dashboard e pode configurar notificações para serem enviadas aos auditores, informando sobre a proximidade do término do inventário ou eventuais atrasos na contagem.

Após a conclusão da contagem, o administrador finaliza o inventário. Os dados são validados por meio de uma análise estatística e, em seguida, processados para realizar os ajustes gerenciais e fiscais do estoque.

### 3.2. Aplicativos semelhantes

A tabela 1 uma comparação entre os aplicativos IS Collector, KCollector e Stock e Inventário Simples, que são aplicativos de contagem de inventário disponíveis na loja de aplicativos do Google. A análise abrange suas principais funcionalidades.

**Tabela 1. Funcionalidades dos aplicativos semelhantes.**

Funcionalidade	IS Collector	KCollector	Stock e Inventário Simples
Trabalho online e offline	Sim	Não	Não
Leitura por código de barras e câmera	Sim	Sim	Sim
Modos de trabalho	Coleta avulsa, inventário completo, conferência	Diversos	Coleta avulsa, inventário completo
Visualização da descrição do item	Sim	Não	Não
Automação de processos	Não	Sim	Não
Integração com Excel	Não	Não	Sim
Relatórios personalizáveis	Sim	Sim	Sim

O IS Collector se destaca pela flexibilidade e abrangência de funções, ideal para empresas que exigem um alto nível de controle e automação no processo de inventário. O KCollector é a opção ideal para quem busca uma solução acessível e prática, que transforma o celular em um coletor eficiente. Já o Stock e Inventário Simples se destaca pela facilidade de uso e pelo gerenciamento completo do estoque, sendo uma ótima opção para iniciantes e pequenos negócios.

O aplicativo do presente trabalho se destaca pela integração com o sistema de gestão da empresa, garantindo a consistência e a atualização em tempo real dos dados de estoque. Além disso, a interface intuitiva e as funcionalidades customizáveis proporcionam uma experiência de usuário eficiente e adaptável às necessidades específicas de cada empresa. A integração com tecnologias avançadas, como leitura de códigos de barras e alerta automatizados, eleva a eficácia do aplicativo, tornando-o uma solução completa e moderna para o controle de inventário.

### 4. Considerações finais

Os erros de saldo de estoque no varejo brasileiro representam um desafio considerável que impacta tanto a eficiência operacional quanto a satisfação do cliente. Este estudo propõe a implementação de um processo de inventário robusto e contínuo, integrado à gestão operacional, como uma solução estratégica para este problema.

A adoção de tecnologias avançadas, tais como sistemas automatizados de rastreamento e leitura por código de barras, pode significativamente aumentar a eficácia do processo de inventário, reduzindo o tempo necessário para corrigir discrepâncias. Ferramentas como Flutter e Spring Boot evidenciam a viabilidade de desenvolver sistemas eficientes para a gestão de inventários.

Contudo, a tecnologia por si só não é suficiente para resolver todos os problemas relacionados ao saldo de estoque. É imperativo serem estabelecidos processos bem definidos, oferecido treinamento adequado e cultivada uma cultura organizacional que valorize a precisão. Dessa forma, ao adotar essas práticas, os varejistas não apenas reduzirão os erros de saldo de estoque, mas também aprimorarão a gestão do estoque e a experiência de compra dos clientes.

## **Referências**

- Pressman, R. S. and Maxim, B. R. (2021). *Engenharia de software: uma abordagem profissional*. AMGH, Porto Alegre, 9ª edition.
- Silva, E. F. and Kloeckner, N. V. d. R. (2022). Método DMAIC aplicado ao controle de estoques de telefonia e informática numa empresa de varejo. XXV SEMEAD (Seminários em Administração da Universidade de São Paulo).
- Slack, N., Brandon-Jones, A., and Johnston, R. (2018). *Administração da produção*. Atlas, São Paulo, 8ª edition.