v7.n11.2026

REVISTA CIENTÍFICA - RECIMA21 ISSN 2675-6218

SISTEMA WEB PARA GESTÃO DE LOCAÇÕES DE ITENS PARA EVENTOS COM CONTROLE DE ESTOQUE E FATURAMENTO

WEB SYSTEM FOR RENTAL MANAGEMENT OF EVENT ITEMS WITH STOCK AND REVENUE CONTROL

SISTEMA WEB PARA LA GESTIÓN DE ALQUILERES DE ARTÍCULOS PARA EVENTOS CON CONTROL DE INVENTARIO Y FACTURACIÓN

Gabriela Fernanda Roque[1], Felipe Diniz Dallilo[2], Fabiana Florian[3]

e7119514
https://doi.org/10.47820/recima21.v7i11.9514

PUBLICADO: 11/2026

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema *web* voltado ao controle de locações de itens para eventos em uma microempresa do setor. O estudo parte da constatação de que negócios desse porte podem depender de processos manuais para o registro de clientes, controle de estoque e cálculo de valores, o que aumenta a chance de falhas operacionais à medida que o volume de locações cresce. A pesquisa foi conduzida de forma aplicada e qualitativa, combinando revisão bibliográfica sobre sistemas de informação, automação de processos e controles internos com o levantamento das necessidades observadas na rotina da empresa estudada. Como resultado, foi desenvolvido um sistema capaz de verificar automaticamente a disponibilidade de produtos, inclusive daqueles compostos por outros itens, calcular o valor das locações sem intervenção manual e acompanhar pagamentos parciais e relatórios de faturamento. Na percepção dos responsáveis pela empresa, a aplicação do sistema em sua rotina real melhorou o controle de cálculos e conferências antes feitos manualmente e proporcionou maior visibilidade sobre o desempenho do negócio. Conclui-se que a informatização desses processos contribui para a organização e a confiabilidade da gestão em microempresas do setor de locação de itens para eventos.

PALAVRAS-CHAVE: Automação de processos. Controle de estoque. Gestão de locações. Microempresas. Sistema *web*. Sistemas de informação.

ABSTRACT

This paper presents the development of a web system aimed at managing the rental of items for events in a microenterprise of this sector. The study starts from the observation that businesses of this size may rely on manual processes to register customers, control stock and calculate rental values, which increases the chance of operational failures as the volume of rentals grows. The research was conducted in an applied and qualitative way, combining a literature review on information systems, process automation and internal controls with a survey of the needs observed in the daily routine of the company studied. As a result, a system was developed that automatically checks product availability, including products composed of other items, calculates rental values without manual intervention and tracks partial payments and revenue reports. According to the perception of those responsible for the company, applying the system to its actual routine improved the control of calculations and checks previously done manually and provided greater visibility over the performance of the business. It is concluded that the computerization of these processes contributes to the organization and reliability of management in microenterprises in the event item rental sector.

[1] Universidade de Araraquara, Graduanda em Sistemas de Informação.
[2] Universidade de São Paulo, Doutor em Engenharia de Software.
[3] UNESP, FCFAr, Doutora em Alimentos e Nutrição.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

KEYWORDS: *Information systems. Inventory control. Microenterprises. Process automation. Rental management. Web system.*

RESUMEN
Este trabajo presenta el desarrollo de un sistema web orientado al control de alquileres de artículos para eventos en una microempresa del sector. El estudio parte de la constatación de que los negocios de este tamaño pueden depender de procesos manuales para el registro de clientes, el control de inventario y el cálculo de valores, lo que aumenta la probabilidad de fallos operativos a medida que crece el volumen de alquileres. La investigación se llevó a cabo de forma aplicada y cualitativa, combinando una revisión bibliográfica sobre sistemas de información, automatización de procesos y controles internos con el levantamiento de las necesidades observadas en la rutina de la empresa estudiada. Como resultado, se desarrolló un sistema capaz de verificar automáticamente la disponibilidad de productos, incluso de aquellos compuestos por otros artículos, calcular el valor de los alquileres sin intervención manual y dar seguimiento a pagos parciales e informes de facturación. Según la percepción de los responsables de la empresa, la aplicación del sistema en su rutina real mejoró el control de cálculos y verificaciones antes realizados manualmente y proporcionó mayor visibilidad sobre el desempeño del negocio. Se concluye que la informatización de estos procesos contribuye a la organización y la fiabilidad de la gestión en microempresas del sector de alquiler de artículos para eventos.

PALABRAS CLAVE: *Automatización de procesos. Control de inventario. Gestión de alquileres. Microempresas. Sistema web. Sistemas de información.*

INTRODUÇÃO

Empresas que atuam no ramo de locação de itens para eventos dependem do gerenciamento eficiente de produtos, estoque, reservas, entregas e pagamentos para garantir a organização de suas atividades e a qualidade dos serviços prestados. À medida que a quantidade de clientes e locações aumenta, o controle manual dessas informações pode se tornar um desafio. Isso dificulta a rotina operacional e aumenta a possibilidade de falhas.

Em muitos pequenos negócios, ainda é comum que os processos sejam realizados de maneira manual, utilizando anotações em papel, agendas físicas e cálculos feitos durante o atendimento ao cliente. Embora esses métodos possam atender às necessidades iniciais da empresa, eles tendem a apresentar limitações à medida que o volume de informações cresce, o que torna mais difícil o acompanhamento das reservas, do estoque e do faturamento. A ausência de um sistema informatizado pode ocasionar problemas como conflitos de datas, indisponibilidade de produtos já reservados, dificuldades no controle financeiro e falhas no cálculo dos valores cobrados.

No setor de locação de itens para eventos, o gerenciamento adequado das reservas e do estoque é fundamental para garantir a organização do negócio e a qualidade do atendimento. Amorim (2023), ao analisar o setor de locação de uma loja de artigos para festas, observou que os contratos eram preenchidos manualmente em formulários físicos e que o controle das

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

devoluções era realizado por meio de uma agenda *online*. Segundo o autor, a ausência de um sistema informatizado para centralizar essas informações dificultava o acompanhamento das locações, a verificação da disponibilidade dos produtos e o controle dos prazos de devolução, além de favorecer falhas como agendamentos de itens indisponíveis e o registro incompleto de informações dos clientes. Diante desse cenário, foi proposto o desenvolvimento de um sistema *web* para auxiliar no gerenciamento das locações e na organização dos processos do setor.

Além das dificuldades encontradas nos processos manuais, cresce entre os pequenos negócios brasileiros a adoção de tecnologias voltadas à gestão das próprias atividades, tema retomado com mais detalhe na fundamentação teórica deste trabalho. Esse movimento reforça a busca por ferramentas que contribuam para a organização das informações, para a melhoria dos processos internos e para a digitalização das atividades administrativas dos pequenos negócios.

Estudos relacionados aos controles internos e à automação empresarial também apontam benefícios na utilização de sistemas informatizados, como é discutido adiante. De forma geral, esse tipo de tecnologia tende a reduzir erros operacionais e a aumentar a confiabilidade dos processos organizacionais.

Diante desse contexto, este trabalho tem como foco de estudo uma microempresa do setor de prestação de serviços de locação de itens para eventos, com o objetivo de contribuir, por meio do desenvolvimento de uma aplicação *web*, para melhorias no controle de estoque, das reservas e do faturamento. Assim, esta pesquisa busca responder à seguinte questão: como essa ferramenta pode auxiliar no controle de reservas, estoque e faturamento em microempresas do setor de locação de itens para eventos?

A hipótese desta pesquisa é que a adoção dessa solução para gerenciamento de locações pode reduzir falhas operacionais, melhorar o controle de estoque e automatizar processos realizados manualmente, o que proporciona maior organização, agilidade e confiabilidade no gerenciamento das informações da empresa.

O objetivo geral deste trabalho é desenvolver um sistema *web* para uma microempresa do setor de locação de itens para eventos, que permita o controle de clientes, produtos, endereços, reservas, estoque e faturamento adequado às necessidades observadas na rotina dessa empresa.

Como objetivos específicos, pretende-se implementar funcionalidades para cadastro e controle de clientes, desenvolver o catálogo de produtos e suas variações, monitorar os períodos de locação dos itens cadastrados, verificar automaticamente a disponibilidade dos produtos para locação, automatizar o cálculo dos valores das locações e gerenciar os pagamentos relacionados às locações realizadas.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

A escolha do tema justifica-se pela necessidade de modernização de pequenos negócios que ainda utilizam métodos manuais em suas rotinas administrativas. A informatização desses processos pode contribuir para a redução de erros humanos, melhoria no controle das informações e maior eficiência operacional. Além disso, o projeto apresenta relevância acadêmica por aplicar, na prática, conceitos relacionados à engenharia de *software*, arquitetura de sistemas, banco de dados e desenvolvimento *web* na resolução de um problema real observado em empresas do setor de locação.

1. REFERENCIAL TEÓRICO

Nesta seção são abordados os principais conceitos e tecnologias que fundamentam o desenvolvimento do sistema proposto.

1.1. Tecnologia e sistemas de informação nos pequenos negócios

A utilização de recursos tecnológicos tem se tornado cada vez mais presente na rotina dos pequenos negócios brasileiros. De acordo com o Sebrae (2025), em levantamento sobre a transformação digital nos pequenos negócios brasileiros, 80% das empresas do comércio e 76% das empresas do setor de serviços utilizam computadores em suas atividades. O mesmo estudo aponta que 47% dos empreendimentos do setor de serviços utilizam *softwares* ou aplicativos capazes de auxiliar na gestão de todas ou quase todas as atividades da empresa. Esses dados indicam que a adoção de tecnologias digitais vem se consolidando como uma alternativa para apoiar a organização e o gerenciamento das atividades empresariais.

A partir dessa realidade, os sistemas de informação assumem papel importante dentro das organizações. De acordo com Laudon e Laudon (2023), os sistemas de informação são formados por componentes inter-relacionados responsáveis por coletar, processar, armazenar e distribuir informações, apoiando a coordenação, o controle, a análise e a tomada de decisões nas empresas. Em pequenos negócios que ainda realizam controles de forma manual, esse tipo de tecnologia pode contribuir para organizar informações relacionadas a clientes, produtos, reservas, estoque e pagamentos. Isso reduz a dependência de registros dispersos.

1.2. Automação de processos

A automação de processos tem sido utilizada pelas organizações como forma de aumentar a eficiência operacional e reduzir a ocorrência de falhas decorrentes da execução manual das atividades. Tarefas antes dependentes da atenção constante de um funcionário, como cálculos, conferências e registros repetitivos, passam a ser executadas de forma automática, o que reduz o tempo gasto em cada operação e diminui a chance de erro humano.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Segundo Trindade et al. (2025), a implementação de mecanismos adequados de controle contribui para a integridade das informações e fortalece os processos organizacionais, além de possibilitar maior rastreabilidade das informações e melhor supervisão das rotinas realizadas pelas organizações. Esse tipo de automação é especialmente relevante em atividades operacionais repetitivas, como o cálculo de valores, a verificação de disponibilidade de itens e o acompanhamento de prazos, situações em que a dependência de conferência manual tende a acumular falhas conforme cresce o volume de operações. No caso deste trabalho, a automação é aplicada principalmente ao cálculo dos valores das locações e à verificação de disponibilidade dos produtos, temas detalhados na seção de resultados e discussão.

1.3. Controles internos

Os controles internos correspondem ao conjunto de mecanismos adotados por uma organização para garantir a confiabilidade das informações, a conformidade dos processos e a proteção dos recursos utilizados em suas operações. Esses mecanismos podem ser reforçados por meio da automação, já que processos automatizados tendem a seguir sempre a mesma lógica de verificação. Isso diminui a variação que ocorre quando cada tarefa depende do julgamento manual de um funcionário.

Nessa direção, Costa e Oliveira (2022 *apud* TRINDADE et al., 2025) apontam que a automação de processos por meio de sistemas ERP e *softwares* de auditoria melhora a rastreabilidade das operações, reduz a ocorrência de erros humanos e contribui para a padronização das informações. No sistema deste trabalho, esse tipo de controle é incorporado à própria lógica da aplicação, por meio da exigência de determinados dados antes do registro de uma operação e da verificação automática de condições que impeçam inconsistências, como a tentativa de reservar um produto já comprometido para o mesmo período. Assim, controles internos bem definidos reduzem a dependência da atenção individual do funcionário e tornam os processos da empresa menos suscetíveis a falhas de registro.

1.4. Controle de estoque e gestão de locações

O controle de estoque exerce papel importante nas empresas que trabalham com locação de produtos, uma vez que a disponibilidade dos itens precisa ser acompanhada continuamente para garantir o atendimento das reservas realizadas pelos clientes. Além do controle das quantidades disponíveis, é necessário monitorar os períodos de locação e devolução dos produtos, evitando conflitos de agendamento e indisponibilidade de materiais.

Ao analisar o setor de locação de uma loja de artigos para festas, Amorim (2023) identificou que a verificação da disponibilidade dos produtos dependia da consulta aos contratos de outros clientes, o que favorecia agendamentos de itens indisponíveis. Um sistema

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

informatizado, como o proposto neste trabalho, permite acompanhar a disponibilidade dos itens de maneira mais organizada, reduzindo a dependência de controles manuais e facilitando o gerenciamento das atividades da empresa.

1.5. Desenvolvimento de sistemas *Web*

Um sistema baseado na *web* é aquele acessado por meio de um navegador, sem exigir a instalação de um programa específico na máquina do usuário nem atualizações manuais em cada equipamento. Segundo Machado (2019), esse modelo permite que o sistema seja utilizado a partir de diferentes tipos de dispositivo, como computadores, *tablets* e *smartphones*, bastando que seja possível executar um navegador no equipamento. O autor destaca ainda que, diferentemente de sistemas tradicionais do tipo cliente/servidor, que exigem a instalação de um programa cliente em cada máquina e limitam o número de acessos, os sistemas baseados na *web* dependem apenas do navegador, o que facilita a atualização e permite atender a um número maior de usuários.

Da mesma forma, o sistema deste trabalho é desenvolvido como uma aplicação *web* voltada ao gerenciamento de locações, com controle de clientes, produtos, reservas, estoque e pagamentos por meio de uma interface acessível via navegador, o que dispensa a instalação de qualquer programa adicional pelos funcionários da empresa.

1.6. Arquitetura de *Software* em camadas

A arquitetura de *software* está relacionada à forma como os componentes de um sistema são organizados e se comunicam entre si. Segundo Pressman (2011), a arquitetura representa a estrutura do *software* e define como seus elementos são organizados para atender aos requisitos do sistema. Existem diferentes abordagens para essa organização, cada uma mais adequada a determinados tipos de aplicação e a diferentes necessidades de manutenção e evolução do *software*.

Entre essas abordagens, a arquitetura em camadas está entre as mais utilizadas. De acordo com Sommerville (2011), esse modelo organiza a aplicação em diferentes responsabilidades, contribuindo para a manutenção do sistema e reduzindo o impacto de alterações futuras. Isso ocorre porque uma alteração feita em uma camada tende a ficar restrita aos componentes daquela camada, sem exigir mudanças nas demais partes do sistema.

Essa característica motivou a escolha da arquitetura em camadas para o sistema deste trabalho, uma vez que a separação entre as responsabilidades de interface, casos de uso, regras de negócio e acesso a dados reduz o risco de que alterações pontuais, como a inclusão de uma nova funcionalidade, comprometam partes não relacionadas do sistema. Assim, o sistema deste trabalho utiliza uma arquitetura em camadas na qual as regras de negócio ficam concentradas

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

nas próprias entidades do domínio, e não distribuídas em outras partes da aplicação, o que ajuda a manter a coerência dos dados manipulados pelo sistema.

1.7. APIs REST e comunicação entre sistemas

As Interfaces de Programação de Aplicações (APIs) permitem a comunicação entre diferentes sistemas, possibilitando a troca de informações de forma padronizada. Segundo Almeida (2025), as APIs desempenham papel importante na integração de aplicações, e entre os modelos mais utilizados está a arquitetura REST (*Representational State Transfer*), cuja ampla adoção está relacionada à simplicidade de implementação e à facilidade de integração entre diferentes tecnologias. Nesse contexto, Aguiar (2024 *apud* ALMEIDA, 2025) destaca que a organização modular dos sistemas e a separação entre lógica de negócio e infraestrutura contribuem para facilitar a manutenção e a evolução do *software*.

Nessa mesma lógica, a comunicação entre o *frontend* em Angular e o *backend* em .NET deste trabalho é realizada por meio de APIs REST, mantendo a separação entre as camadas da aplicação.

1.8. Plataforma .NET e Linguagem C#

A plataforma .NET é utilizada no desenvolvimento de diferentes tipos de aplicações, incluindo sistemas *web*, aplicações desktop, serviços e APIs. De acordo com a Microsoft (2025a), trata-se de uma plataforma de desenvolvimento multiplataforma. Entre as linguagens suportadas está o C#, amplamente utilizado no desenvolvimento de *software*, cuja integração com os recursos do .NET permite a criação de aplicações organizadas e de fácil manutenção.

No caso deste trabalho, a plataforma .NET e a linguagem C# são utilizadas no *backend*, responsável pelo processamento das regras de negócio e pela comunicação com o banco de dados.

1.9. Angular no desenvolvimento *Frontend*

O Angular é um *framework* utilizado no desenvolvimento de aplicações *web*. De acordo com a documentação oficial do Angular (2025), sua arquitetura é baseada em componentes, o que permite dividir a interface em partes reutilizáveis e organizadas, contribuindo para facilitar a manutenção e a evolução do sistema ao longo do tempo.

Já o Angular deste trabalho é utilizado no desenvolvimento da interface com o usuário, dando suporte ao gerenciamento das informações de clientes, produtos, locações e pagamentos.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

1.10. SQL Server

O SQL Server é um Sistema Gerenciador de Banco de Dados Relacional (SGBDR) desenvolvido pela Microsoft, utilizado para o armazenamento, organização e gerenciamento de dados em aplicações computacionais, com recursos para consultas, controle de acesso e manutenção da integridade dos registros. Segundo a Microsoft (2025b), o SQL Server pode ser utilizado em diferentes tipos de aplicações, oferecendo suporte ao armazenamento e ao gerenciamento de dados conforme a necessidade de cada sistema.

O SQL Server deste trabalho, por sua vez, armazena os dados relacionados a clientes, produtos, locações, reservas, endereços e pagamentos, servindo como base para o funcionamento da aplicação.

2. METODOLOGIA

A metodologia adotada nesta pesquisa é classificada como aplicada, pois tem como objetivo solucionar um problema prático identificado em uma microempresa do setor de locações. A pesquisa segue uma abordagem qualitativa e caráter descritivo, sendo desenvolvida por meio de levantamento bibliográfico para fundamentação teórica e análise das necessidades observadas no processo de locação da empresa estudada. O desenvolvimento da aplicação é realizado com base em conceitos de arquitetura em camadas, APIs REST e banco de dados relacional, de forma a atender às necessidades identificadas durante o levantamento dos requisitos.

O levantamento dos requisitos foi realizado por meio da observação direta da rotina da empresa, que é um negócio familiar, e do diálogo com seus responsáveis, a partir das situações enfrentadas no dia a dia, como o controle de estoque de *kits* e a separação do estoque por cor em cada produto. O desenvolvimento seguiu um processo iterativo e incremental: a cada funcionalidade concluída, o sistema era apresentado aos responsáveis pela empresa, e os ajustes apontados eram incorporados antes de avançar para a etapa seguinte. Os testes foram manuais e consistiram em reproduzir no sistema diferentes situações reais de locação da empresa, corrigindo as falhas e lacunas identificadas. A avaliação dos resultados foi qualitativa, baseada no retorno dos usuários e na observação do uso do sistema, sem medição comparativa de indicadores antes e depois da implantação.

Para a construção do sistema, é utilizada a plataforma .NET no *backend*, devido à facilidade na construção de APIs e ao suporte oferecido para aplicações *web*. O *frontend* é desenvolvido utilizando Angular, tecnologia que possibilita a criação de interfaces organizadas e dinâmicas, proporcionando melhor experiência ao usuário. Já o SQL Server é utilizado como

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

sistema gerenciador de banco de dados para armazenamento e manipulação das informações da aplicação.

3. RESULTADOS E DISCUSSÃO

Esta seção descreve como o sistema de gerenciamento de locações foi estruturado e apresenta as funcionalidades implementadas, bem como os resultados obtidos com sua utilização pela empresa.

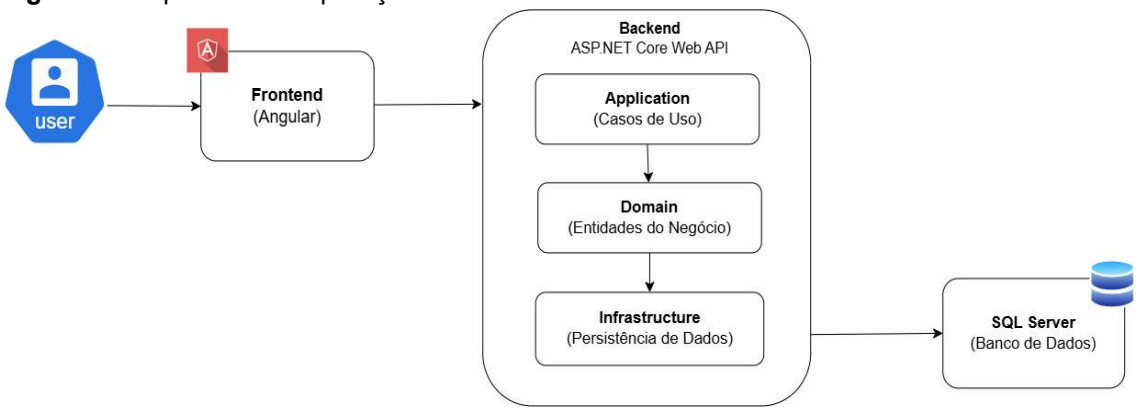
3.1. Arquitetura da aplicação

A arquitetura do sistema é definida a partir de uma estrutura em camadas, buscando separar as responsabilidades da aplicação e facilitar sua manutenção. O sistema é composto por um *frontend* desenvolvido em Angular, responsável pela interação com o usuário, e por um *backend* desenvolvido em ASP.NET Core Web API, responsável pelo processamento das requisições e acesso aos dados.

A camada *Application* é responsável pela execução dos casos de uso da aplicação, enquanto a camada *Domain* concentra as entidades e conceitos relacionados ao negócio. A camada *Infrastructure* realiza a comunicação com o banco de dados SQL Server, responsável pelo armazenamento das informações do sistema.

A Figura 1 apresenta a arquitetura definida para a aplicação e o fluxo de comunicação entre seus componentes.

Figura 1. Arquitetura da Aplicação



Fonte: Autoria própria, 2026.

A separação em camadas busca facilitar a organização do sistema, de modo que diferentes componentes da aplicação possam evoluir de forma mais independente.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

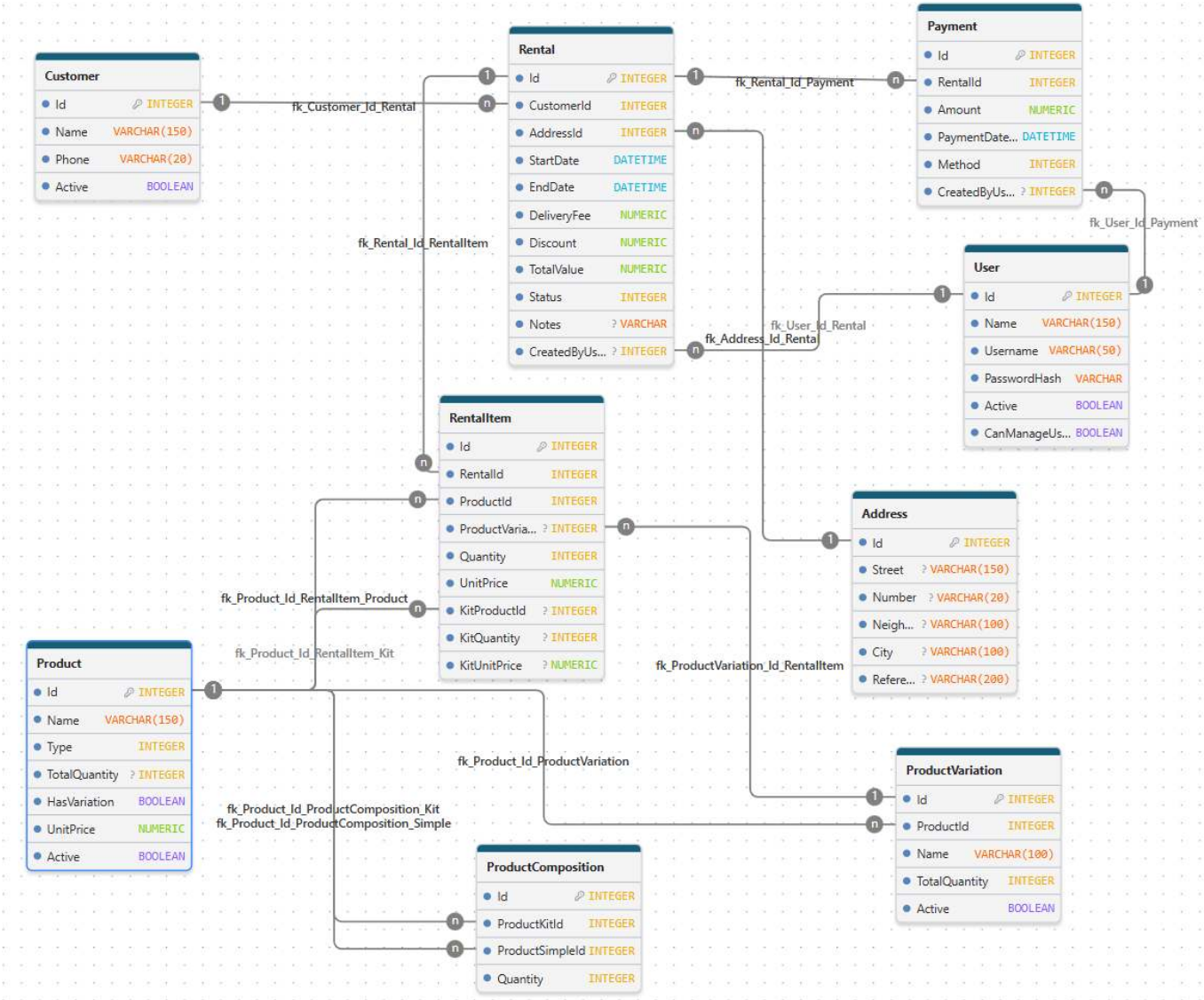
3.2. Modelagem do Banco de Dados

A modelagem do banco de dados tem como objetivo armazenar e organizar as informações necessárias para o funcionamento do sistema de gerenciamento de locações. A estrutura proposta contempla o controle de clientes, produtos, locações, pagamentos, endereços e usuários, com relacionamentos definidos entre os dados utilizados pela aplicação.

O banco de dados foi projetado a partir dos requisitos levantados durante a etapa de análise do sistema, com o objetivo de garantir a integridade das informações e dar suporte às funcionalidades previstas para o gerenciamento das locações.

A Figura 2 apresenta o Diagrama Entidade-Relacionamento desenvolvido para o sistema.

Figura 2. Diagrama entidade-relacionamento do sistema



Fonte: Autoria própria, 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

3.2.1. Principais entidades do sistema

O modelo de dados foi estruturado a partir das principais entidades necessárias para o gerenciamento das locações realizadas pela empresa.

A entidade *Customer* é responsável pelo armazenamento dos dados dos clientes cadastrados na aplicação, enquanto a entidade *Address* registra os endereços vinculados às locações realizadas. Os campos dessa entidade são opcionais, de forma a atender locais sem endereço formal, situação comum na rotina da empresa estudada.

Os produtos disponíveis para locação são armazenados na entidade *Product*. Para permitir o gerenciamento de diferentes variações de um mesmo produto, foi desenvolvida a entidade *ProductVariation*. Já a entidade *ProductComposition* possibilita a criação de *kits* compostos por outros produtos cadastrados no sistema.

As locações realizadas pela empresa são registradas na entidade *Rental*, responsável por armazenar informações relacionadas ao cliente, endereço, período da locação, taxa de entrega, desconto e valor total da operação. Os produtos associados a cada locação são controlados pela entidade *RentalItem*, que registra os itens reservados, suas variações e, no caso de *kits*, a quantidade e o preço próprio do *kit* contratado.

A entidade *Payment* armazena as informações referentes aos pagamentos realizados pelos clientes, mantendo o vínculo com a locação correspondente. Por fim, a entidade *User* controla os usuários autorizados a operar o sistema, registrando qual usuário realizou cada locação e cada pagamento.

Os relacionamentos definidos entre as entidades permitem o controle integrado das informações do sistema, com o gerenciamento dos produtos, reservas, clientes, pagamentos e usuários centralizado em uma única estrutura.

3.3. Fluxo geral de funcionamento do sistema

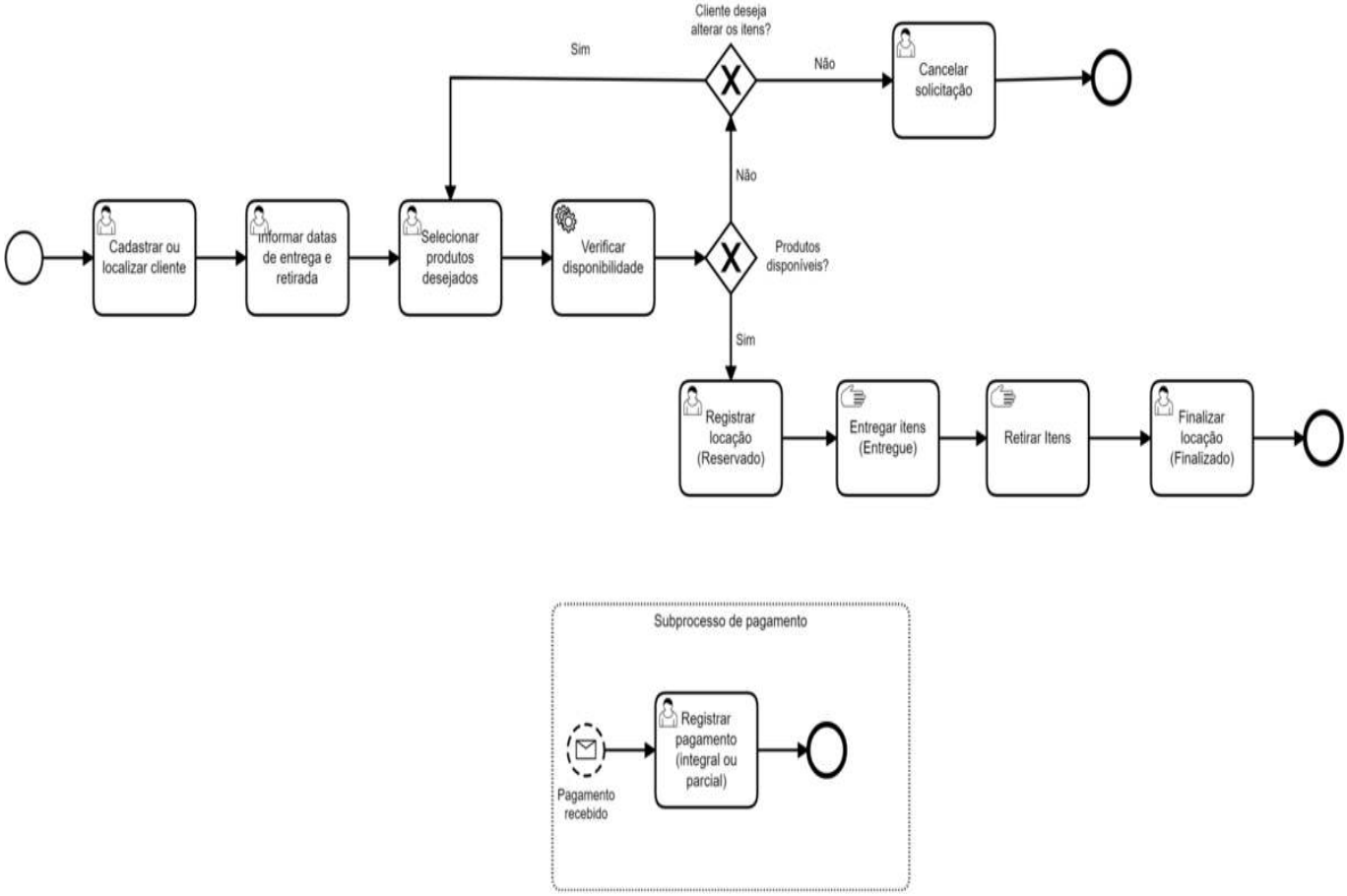
O sistema foi projetado para auxiliar o gerenciamento das locações realizadas pela empresa, desde o atendimento ao cliente até a finalização da locação. O processo inicia-se com o cadastro ou localização do cliente já existente na base de dados. Em seguida, são informadas as datas de entrega e retirada e selecionados os produtos desejados, sendo realizada a verificação automática de disponibilidade para o período informado. Quando um produto não está disponível, o funcionário pode ajustar a seleção de itens ou optar por cancelar a solicitação.

Após a validação dos itens disponíveis, a locação é registrada no sistema como reservada. Com a reserva confirmada, os produtos seguem para entrega, momento em que a locação passa a ser considerada entregue. A retirada dos itens após o evento, embora presente no fluxo operacional da empresa, não corresponde a uma mudança de status registrada pelo

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

sistema, uma vez que este controla apenas os momentos de reserva, entrega, finalização e cancelamento da locação. O pagamento, integral ou parcial, é tratado como um processo independente, podendo ser registrado a qualquer momento entre a reserva e a finalização, sem que isso seja necessário para a confirmação da reserva. Ao término, a locação é finalizada, mantendo o histórico das operações realizadas. A Figura 3 resume essa sequência, do cadastro do cliente até a finalização da locação.

Figura 3. Fluxo geral de funcionamento do sistema



Fonte: Autoria própria, 2026.

3.4. Aspectos de Implementação

A disponibilidade de um produto é recalculada a cada consulta, considerando as demais locações já registradas para o mesmo período. No caso de um *kit*, que não tem estoque próprio, ela é derivada do estoque de cada componente. A Figura 4 apresenta o método que realiza esse cálculo.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 4. Cálculo da disponibilidade de um *kit* a partir dos componentes

```
private async Task<int> CalculateKitAvailability(
    Product kit,
    List<KitComponentSelectionDto>? componentVariations,
    List<Rental> conflictingRentals)
{
    if (kit.Compositions.Count == 0)
        return 0;

    int? minKits = null;

    foreach (var comp in kit.Compositions)
    {
        var componentProduct = await _productRepository.GetById(comp.ProductSimpleId);

        if (componentProduct == null)
            throw new KeyNotFoundException($"Componente do kit '{kit.Name}' não encontrado.");

        int? componentVariationId = null;

        if (componentProduct.HasVariation)
        {
            var selection = componentVariations?.FirstOrDefault(c => c.ProductId == componentProduct.Id);

            if (selection == null)
                throw new InvalidOperationException(
                    $"É necessário informar a variação de '{componentProduct.Name}' para consultar a disponibilidade do kit '{kit.Name}'.");

            componentVariationId = selection.ProductVariationId;
        }

        int componentAvailable = CalculateSimpleAvailability(componentProduct, componentVariationId, conflictingRentals);

        int kitsFromThisComponent = componentAvailable / comp.Quantity;

        minKits = minKits.HasValue ? Math.Min(minKits.Value, kitsFromThisComponent) : kitsFromThisComponent;
    }

    return Math.Max(0, minKits ?? 0);
}
}
```

Fonte: Autoria própria, 2026.

O método percorre cada componente do *kit* e calcula quantos *kits* completos ainda poderiam ser montados com o estoque disponível daquele componente, dividindo a quantidade livre pela quantidade exigida por *kit*. Como a montagem de um *kit* depende de todos os seus componentes ao mesmo tempo, o menor resultado entre eles determina o limite real de montagem, já que a falta de um único componente já é suficiente para reduzir a disponibilidade do conjunto inteiro. Sem essa regra, o sistema não teria como oferecer um *kit* como um produto único na interface, uma vez que seu estoque não existe fisicamente como tal.

Quando um *kit* é adicionado a uma locação, cada componente mantém seu próprio controle de estoque, mas recebe um preço proporcional ao valor combinado para aquele *kit*, calculado anteriormente no mesmo método, de forma que a soma dos componentes corresponda exatamente ao total contratado, mesmo esse valor variando entre locações. Quando o componente também possui variação de cor, como o Cobre-Mancha usado no Jogo de Toalha,

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

a quantidade contratada pode ser dividida entre várias cores em uma mesma locação. O sistema confere se a soma das quantidades escolhidas corresponde ao total exigido pelo *kit* e, em seguida, verifica o estoque de cada cor individualmente antes de registrar o item correspondente, já com esse preço proporcional. A Figura 5 traz o trecho de código que executa essas verificações.

Figura 5. Divisão de um componente de *kit* entre variações de cor

```
var selections = itemDto.ComponentVariations?
    .Where(c => c.ProductId == componentProduct.Id)
    .ToList() ?? new List<KitComponentSelectionDto>();

if (selections.Count == 0)
    throw new InvalidOperationException(
        $"É necessário escolher a variação de '{componentProduct.Name}' para o kit '{product.Name}'.");

var somaSelecionada = selections.Sum(s => s.Quantity);

if (somaSelecionada != requiredQuantity)
    throw new InvalidOperationException(
        $"A soma das quantidades por variação de '{componentProduct.Name}' ({somaSelecionada}) precisa ser igual a {requiredQuantity} para o kit '{product.Name}'.");

foreach (var selection in selections)
{
    if (selection.Quantity <= 0)
        throw new InvalidOperationException(
            $"Quantidade inválida para uma variação de '{componentProduct.Name}'.");

    var componentVariation = componentProduct.Variations
        .FirstOrDefault(v => v.Id == selection.ProductVariationId);

    if (componentVariation == null)
        throw new InvalidOperationException(
            $"A variação informada não pertence ao produto '{componentProduct.Name}'.");

    int alreadyReserved = conflictingRentals
        .SelectMany(r => r.Items)
        .Concat(rental.Items)
        .Where(i => i.ProductId == componentProduct.Id && i.ProductVariationId == componentVariation.Id)
        .Sum(i => i.Quantity);

    int available = componentVariation.TotalQuantity - alreadyReserved;

    if (selection.Quantity > available)
        throw new InvalidOperationException(
            $"Estoque insuficiente de '{componentProduct.Name}' ({componentVariation.Name}) para o kit '{product.Name}'.");

    rental.AddItem(new RentalItem(
        componentProduct,
        selection.Quantity,
        precoComponente,
        componentVariation,
        kit: product,
        kitQuantity: itemDto.Quantity,
        kitUnitPrice: kitUnitPrice
    ));
}
```

Fonte: Autoria própria, 2026.

Esse trecho mostra que a divisão entre cores não é apenas uma conferência de números: cada cor selecionada gera um item de locação próprio, com sua própria verificação de estoque disponível, mas todos compartilham o mesmo preço proporcional combinado para aquele *kit*. Sem essa verificação por variação, seria possível reservar uma cor sem estoque suficiente

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

apenas porque o total geral do componente ainda estava disponível, como no exemplo dos doze jogos de toalha divididos entre quatro toalhas roxas, quatro azuis e quatro amarelas.

Por fim, cada item adicionado a uma locação aciona automaticamente o recálculo do valor total, que soma os itens avulsos e os *kits* já com o preço proporcional atribuído a cada componente, aplica a taxa de entrega e desconta eventuais reduções. Esse recálculo está implementado no método da Figura 6.

Figura 6. Cálculo automático do valor total da locação

```
private void RecalculateTotal()
{
    var kitsTotal = Items
        .Where(i => i.KitProductId.HasValue)
        .GroupBy(i => i.KitProductId!.Value)
        .Sum(g => g.First().KitUnitPrice!.Value * g.First().KitQuantity!.Value);

    var avulsosTotal = Items
        .Where(i => !i.KitProductId.HasValue)
        .Sum(i => i.Quantity * i.UnitPrice);

    TotalValue = kitsTotal + avulsosTotal + DeliveryFee - Discount;
}
```

Fonte: Autoria própria, 2026.

O agrupamento por *KitProductId* evita que o valor do mesmo *kit* seja contado mais de uma vez, já que cada componente do *kit* gera um item de locação próprio, mas todos compartilham o mesmo preço e a mesma quantidade do *kit*. Sem esse agrupamento, um *kit* com três componentes, por exemplo, teria seu preço somado três vezes no valor total. É esse recálculo automático que mantém o valor total atualizado durante o registro da locação, sem exigir nenhuma soma manual por parte do funcionário.

3.5. Funcionalidades implementadas

O acesso ao sistema é feito por meio de *login*, no qual o funcionário informa usuário e senha cadastrados previamente.

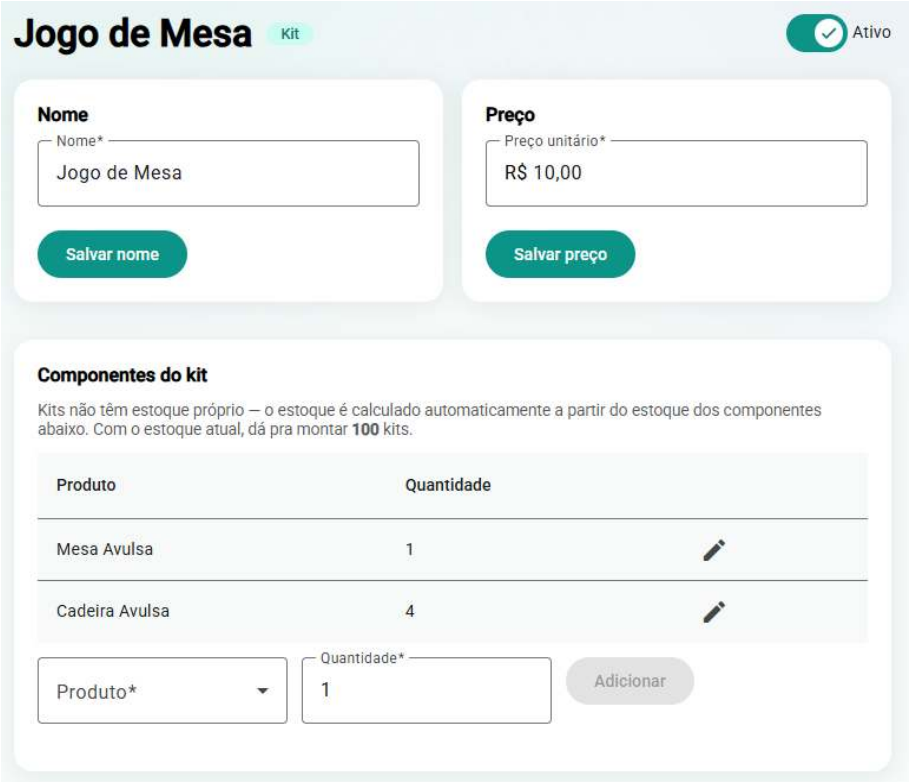
A partir do *login*, o sistema controla quais operações cada usuário pode realizar, sendo possível definir quais funcionários têm permissão para gerenciar outros usuários.

Na aba de Clientes, é possível realizar o cadastro e a edição dos dados de cada cliente. Um mesmo cliente não precisa ser cadastrado novamente ao registrar uma nova locação. Ao informar o telefone na tela de Nova Locação, o sistema busca automaticamente se aquele número já pertence a um cliente cadastrado e preenche o nome correspondente, evitando duplicidade de cadastros. Clientes que já possuem locações associadas não são removidos do sistema, apenas inativados, o que preserva o histórico já registrado.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

O catálogo de produtos foi estruturado para atender três situações da rotina da empresa: produtos simples, como uma mesa avulsa; produtos com variação, como toalhas disponíveis em diferentes cores; e *kits*, formados pela composição de outros produtos cadastrados. Um kit, como o conjunto de mesa com cadeiras, é formado por outros produtos cadastrados e não possui estoque próprio. A Figura 7 exibe a tela de Produtos, na qual um *kit* aparece junto com os componentes que o formam.

Figura 7. Tela de Produtos, com um *kit* e seus componentes



Fonte: Autoria própria, 2026.

Para o registro das locações, o sistema exige a informação do cliente, dos itens desejados, da data de entrega e da data de retirada. O endereço de entrega também pode ser informado, mas nenhum de seus campos é obrigatório, uma vez que parte das locações ocorre em locais sem um endereço formal, como sítios e chácaras. A cada item adicionado, é verificada automaticamente a disponibilidade para o período informado, considerando as demais locações já registradas. Quando o item é um *kit* composto por produtos com variação, o sistema permite dividir a quantidade contratada entre diferentes cores, respeitando o total exigido, como por exemplo doze jogos de toalha divididos entre quatro toalhas roxas, quatro azuis e quatro

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

amarelas dentro da mesma locação. A Figura 8 exibe a tela de Nova Locação, na qual essa divisão é feita.

Figura 8. Tela de Nova Locação, com a divisão de um item de *kit* entre cores



Fonte: Autoria própria, 2026.

O valor total é recalculado automaticamente conforme os itens são inseridos, somando o preço de cada item à taxa de entrega e subtraindo eventuais descontos.

O controle de pagamentos permite o registro de valores parciais, com cálculo automático do saldo devedor. Na lista de locações, um indicador visual mostra se o pagamento está em dia ou quanto ainda falta, o que permite à empresa acompanhar a situação financeira das locações sem precisar abrir cada uma individualmente. Esse indicador aparece na lista de locações da Figura 9.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

Figura 9. Tela de Lista de Locações, com o indicador de pagamento



Fonte: Autoria própria, 2026.

O sistema conta ainda com um módulo de relatórios, que reúne informações de faturamento por período, valores recebidos e pendentes, e locações canceladas dentro do intervalo selecionado. A Figura 10 exibe a tela desse módulo.

Figura 10. Tela de Relatórios, com faturamento por período



Fonte: Autoria própria, 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

É importante observar que o faturamento considera apenas as locações já entregues ou finalizadas, enquanto os valores recebidos e pendentes consideram todas as locações não canceladas, inclusive as reservas ainda não entregues. Por isso, na Figura 10, o valor recebido pode superar o faturamento do período, já que inclui pagamentos feitos antecipadamente em reservas.

3.6. Resultados obtidos

O sistema desenvolvido atendeu aos objetivos propostos, apresentando funcionamento adequado nas principais operações de cadastro, consulta e controle de locações, segundo a avaliação qualitativa realizada com os responsáveis pela empresa.

O cadastro de clientes passou a ser único e centralizado, com busca por telefone diretamente na tela de nova locação. Isso dispensa o preenchimento repetido de dados a cada novo atendimento. O catálogo de produtos passou a contemplar itens simples, itens com variação e *kits* compostos por outros produtos cadastrados, com a disponibilidade de cada *kit* calculada automaticamente a partir do estoque de seus componentes.

O acompanhamento das locações passou a depender do status de cada registro, que evolui da reserva até a finalização, permitindo identificar rapidamente em que etapa cada locação se encontra. A cada item adicionado a uma nova locação, o sistema confere em tempo real se há conflito com outras locações já registradas para o mesmo período.

Segundo os responsáveis pela empresa, esse ponto específico melhorou a rotina, pois reduziu a dependência da conferência manual. Antes, a disponibilidade de um *kit* precisava ser calculada mentalmente a cada pedido, cruzando manualmente o estoque de cada componente com o que já estava reservado para o período. Com o cálculo automático, esse tipo de erro deixou de depender da atenção do funcionário no momento do atendimento. O mesmo vale para o valor cobrado em cada locação, calculado automaticamente a partir do preço dos itens, da taxa de entrega e dos descontos concedidos. O controle financeiro seguiu o mesmo princípio: pagamentos parciais são aceitos e o saldo devedor é calculado sem intervenção manual.

Quanto à visibilidade do negócio, o módulo de relatórios trouxe uma mudança relevante na forma como a empresa passou a acompanhar os próprios números. Informações como faturamento por período e valores já recebidos e pendentes antes exigiam somar manualmente contratos ou anotações, quando eram acompanhadas. Com o sistema, esses dados ficam disponíveis de forma consolidada e visual.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

4. CONSIDERAÇÕES FINAIS

O sistema desenvolvido neste trabalho cumpriu o que se propôs: centralizar o cadastro de clientes e produtos, automatizar a verificação de disponibilidade e o cálculo dos valores das locações, e dar à empresa uma visão dos próprios números que antes não existia de forma organizada.

O maior ganho, na prática, veio da automação de tarefas que antes dependiam de cálculo manual, como a disponibilidade de um *kit* montado a partir de vários componentes ou o valor total de uma locação com desconto e taxa de entrega. Isso vai ao encontro do que se esperava no início do trabalho: que digitalizar esses processos reduziria falhas e traria mais organização para a rotina da empresa.

Uma limitação a destacar é que o sistema foi validado com o uso real de uma única empresa e avaliado de forma qualitativa, sem medição comparativa da quantidade de erros antes e depois, de modo que os resultados não podem ser generalizados automaticamente para negócios de outro porte ou com processos bem diferentes.

Como sugestão para a continuidade do trabalho, destacam-se duas frentes: explorar uma versão para celular, focada em consultar disponibilidade de produtos rapidamente; e ampliar os relatórios com projeções de faturamento e um indicador de quais produtos atingem o limite de disponibilidade com mais frequência, ajudando a empresa a decidir onde vale a pena investir na compra de mais unidades.

Um último ponto a considerar é a capacidade de crescimento do sistema. A aplicação foi desenvolvida para a rotina da empresa estudada e não passou por testes com grande volume de acessos ou de registros. Caso a empresa cresça de forma significativa ou o sistema seja adaptado para outros negócios, seria importante realizar testes de carga para observar o comportamento da aplicação com um volume maior de dados e de usuários, e fazer os ajustes necessários a partir dos resultados.

REFERÊNCIAS

ALMEIDA, Rodrigo Monteiro Guedes de. Boas práticas em desenvolvimento de APIs para integração de sistemas. **Lumen et Virtus**, *São José dos Pinhais*, v. 16, n. 45, p. 1636-1655, fev. 2025. DOI: 10.56238/levv16n45-069. Disponível em: https://periodicos.newsciencepubl.com/LEV/article/view/5213. Acesso em: 23 maio 2026.

AMORIM, Tassio Hipólito de. **Desenvolvimento de um sistema web para gestão de locações em uma loja de artigos para festas**. 2023. 48 f. Trabalho de Conclusão de Curso (Bacharelado em Tecnologias da Informação e Comunicação) – Universidade Federal de Santa Catarina, Araranguá, 2023. Disponível em:

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.

https://repositorio.ufsc.br/bitstream/handle/123456789/253697/Trabalho_de_Conclus%c3%a3o_de_Curso_Tassio_Hip%c3%b3lito.pdf?sequence=1&isAllowed=y. Acesso em: 24 maio 2026.

ANGULAR. **Angular documentation**. 2025. Disponível em: https://angular.dev/overview. Acesso em: 23 maio 2026.

LAUDON, Kenneth C.; LAUDON, Jane P. **Sistemas de informação gerenciais**: administrando a empresa digital. 17. ed. Porto Alegre: Bookman, 2023.

MACHADO, Vinicius Ponte. **Desenvolvimento para web**. Teresina: EDUFPI, 2019. 149 p.

MICROSOFT. **Introduction to .NET**. 2025a. Disponível em: https://learn.microsoft.com/en-us/dotnet/core/introduction. Acesso em: 24 maio 2026.

MICROSOFT. **SQL Server documentation**. 2025b. Disponível em: https://learn.microsoft.com/en-us/sql/sql-server. Acesso em: 24 maio 2026.

PRESSMAN, Roger S. **Engenharia de software**: uma abordagem profissional. 7. ed. Porto Alegre: AMGH, 2011.

SEBRAE. **Transformação digital nos pequenos negócios 2025**: porte, setor, região e UF. 7. ed. *[S. l.]*: Sebrae Nacional, 2025. Disponível em: https://sebraepr.com.br/wp-content/uploads/2025/07/PESQUISA-TIC-2025-Transformacao-Digital-nos-Pequenos-Negocios-Porte-Setor-Regiao-e-UF.pdf. Acesso em: 23 maio 2026.

SOMMERVILLE, Ian. **Engenharia de software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

TRINDADE, Gabriel Hissung; SANTOS, Gabriel Moraes dos; PISTOLATO, Rafael de Oliveira; BORELI, Daniela. A importância dos controles internos na mitigação de erros e fraudes contábeis nas organizações. **Revista Ibero-Americana de Humanidades, Ciências e Educação**, *[S. l.]*, v. 11, n. 11, p. 9988-10004, nov. 2025. DOI: 10.51891/rease.v11i11.22665. Disponível em: https://periodicorease.pro.br/rease/article/view/22665. Acesso em: 23 maio 2026.

Este artigo é publicado em acesso aberto (Open Access) sob a licença Creative Commons Atribuição 4.0 Internacional (CC-BY), que permite uso, distribuição e reprodução irrestritos em qualquer meio, desde que o autor original e a fonte sejam creditados.