

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
CAMPUS DE CAJAZEIRAS
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

THALYSON RIAN MENDES DA SILVA

**Facilita Comanda: UMA APLICAÇÃO MÓVEL PARA AUTOMAÇÃO DE PEDIDOS
E GESTÃO DE COZINHA**

**CAJAZEIRAS-PB
2026**



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA INSTITUTO
FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

THALYSON RIAN MENDES DA SILVA

**Facilita Comanda: UMA APLICAÇÃO MÓVEL PARA AUTOMAÇÃO DE PEDIDOS
E GESTÃO DE COZINHA**

Trabalho de Conclusão de Curso apresentado
junto ao Curso Superior de Tecnologia em
Análise e Desenvolvimento de Sistemas do
Instituto Federal de Educação, Ciência e
Tecnologia da Paraíba - Campus Cajazeiras,
como requisito à obtenção do título de
Tecnólogo em Análise e Desenvolvimento de
Sistemas.

Orientador

Prof. Dr. Francisco Daladier Marques Junior

Aprovada em: **13 de fevereiro de 2026.**

Prof. Dr. Francisco Daladier Marques Junior - Orientador

Prof. Me. Diogo Dantas Moreira - Avaliador IFPB - Campus Cajazeiras

Prof. Me. Fábio Abrantes Diniz - Avaliador IFPB - Campus Cajazeiras

Documento assinado eletronicamente por:

- **Fabio Abrantes Diniz, COORDENADOR(A) DE CURSOS - FUC1 - UNINFO-CZ**, em 13/02/2026 12:46:42.
- **Francisco Daladier Marques Junior, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 13/02/2026 13:55:00.
- **Diogo Dantas Moreira, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 13/02/2026 14:21:47.

Este documento foi emitido pelo SUAP em 13/02/2026. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código 835867
Verificador: 05b636c601
Código de Autenticação:



Rua José Antônio da Silva, 300, Jardim Oásis, CAJAZEIRAS / PB, CEP 58.900-000
<http://ifpb.edu.br> - (83) 3532-4100

IFPB / Campus Cajazeiras
Coordenação de Biblioteca
Biblioteca Prof. Ribamar da
Silva

Catálogo na fonte: Cícero Luciano Félix CRB-15/750

S586v Silva, Thalyson Rian Mendes da.
Facilita comanda : uma aplicação móvel para automação de pedidos e
gestão de cozinha / Thalyson Rian Mendes da Silva.– 2026.

29f. : il.

Trabalho de Conclusão de Curso (Tecnólogo em Análise e
Desenvolvimento de Sistemas) - Instituto Federal de Educação, Ciência e
Tecnologia da Paraíba, Cajazeiras, 2026.

Orientador(a): Prof. Dr. Francisco Daladier Marques Junior.

1. Desenvolvimento de sistemas. 2. Automação comercial. 3. Gestão de
pedidos. 4. Comanda digital. I. Instituto Federal de Educação, Ciência e
Tecnologia da Paraíba. II. Título.

IFPB/CZ

CDU: 004.4(043.2)

RESUMO

A gestão eficiente de pedidos é um dos maiores desafios para pequenos restaurantes e lanchonetes, onde o uso de comandas de papel frequentemente resulta em erros, atrasos e desperdício. Este trabalho apresenta o desenvolvimento do *Facilita Comanda*, uma aplicação móvel voltada para a automação desse fluxo. A solução utiliza uma abordagem híbrida, funcionando tanto como um sistema de exibição digital (KDS) quanto como um gerenciador de impressão térmica via Bluetooth. O sistema foi desenvolvido utilizando a biblioteca *React Native* para o front-end e a plataforma *Google Firebase* (BaaS) para autenticação e banco de dados em tempo real. Os resultados demonstram que a aplicação é capaz de reduzir o tempo de comunicação entre o salão e a cozinha, eliminando a perda de pedidos e oferecendo controle financeiro automatizado ao gestor.

Palavras-chave: Automação comercial, React Native, Firebase, Comanda Digital, Gestão de Restaurantes.

ABSTRACT

Efficient order management is one of the biggest challenges for small restaurants and snack bars, where the use of paper tickets often leads to errors, delays, and waste. This work presents the development of *Facilita Comanda*, a mobile application aimed at automating this workflow. The solution uses a hybrid approach, functioning both as a Kitchen Display System (KDS) and a Bluetooth thermal print manager. The system was developed using the *React Native* framework for the front-end and the *Google Firebase* platform (BaaS) for authentication and real-time database. Results show that the application is capable of reducing communication time between the dining area and the kitchen, eliminating order loss, and offering automated financial control to the manager.

Keywords: Commercial automation, React Native, Firebase, Digital Order System, Restaurant Management.

LISTA DE FIGURAS

Figura 1 - Arquitetura da solução Facilita Comanda.	13
Figura 2 - Documento JSON da coleção "Orders"	14
Figura 3 - Diagrama de classes das coleções do Cloud Firestore.	15
Figura 4 - Diagrama de Sequência de Impressão.	16
Figura 5 - Tela de autenticação do usuário.	17
Figura 6 - Tela inicial com visualização dos produtos.	18
Figura 7 - Detalhamento do pedido e inserção de observações.	19
Figura 8 - Fila de pedidos da cozinha (KDS).....	20
Figura 9 - Relatório financeiro.	21
Figura 10 – Listagem de produtos cadastrados.	22
Figura 11 - Formulário de cadastro de novos produtos.	23
Figura 12 - Identificação do usuário e nível de acesso.	24

LISTA DE ABREVIATURAS E SIGLAS

ABNT - Associação Brasileira de Normas Técnicas

API - *Application Programming Interface*

BaaS - *Backend-as-a-Service*

CRUD - *Create, Read, Update, Delete*

ESC/POS - *Epson Standard Code for Point of Sale*

IFPB - Instituto Federal de Educação, Ciência e Tecnologia da Paraíba

JSON - *JavaScript Object Notation*

KDS - *Kitchen Display System*

MVVM - *Model-View-ViewModel*

NoSQL - *Not Only SQL*

REST - *Representational State Transfer*

SDK - *Software Development Kit*

TCC - Trabalho de Conclusão de Curso

UI - *User Interface*

UX - *User Experience*

SUMÁRIO

1. INTRODUÇÃO	7
1.1 Motivação	8
1.2 Objetivos.....	8
1.2.1 Objetivo Geral	8
1.2.2 Objetivos Específicos	8
1.3 Organização do Documento	9
2. METODOLOGIA	10
3. SOLUÇÃO PROPOSTA	11
3.1 Requisitos Funcionais.....	11
3.2 Arquitetura	12
3.3 Modelagem do banco de dados.....	13
3.4 Implementação	15
3.4.1 Gerenciamento de Estado e Sincronização	15
3.4.2 Serviço de Impressão Térmica.....	15
4. RESULTADOS E DISCUSSÃO	17
4.1 Controle de Acesso e Segurança	17
4.2 Gestão de Cardápio e Tela Inicial.....	18
4.3 Fluxo de Lançamento de Pedidos	19
4.4 Fila da Cozinha (KDS)	20
4.5 Painel Financeiro e Tomada de Decisão	21
4.6 Gestão de Produtos e Estoque	22
4.7 Perfil de Usuário e Controle de Nível.....	23
5. CONSIDERAÇÕES FINAIS	25
5.1 Trabalhos Futuros.....	25
REFERÊNCIAS.....	26
APÊNDICE A	27

1. INTRODUÇÃO

O setor de alimentação fora do lar é um dos mais dinâmicos da economia brasileira, caracterizado pela alta competitividade e pela necessidade constante de agilidade no atendimento. Em restaurantes, lanchonetes e *food trucks*, a gestão eficiente dos pedidos é um fator crítico para o sucesso do negócio. No entanto, muitos desses estabelecimentos ainda operam com processos manuais e analógicos, dependendo excessivamente de comandas de papel e da comunicação verbal entre o salão e a cozinha. Estudos apontam que a automação logística é vital para reduzir erros e aumentar a produtividade (KARNEKE; KEIJO, 2022). No setor de alimentação, o uso de comandas digitais especificamente ajuda a evitar falhas de comunicação entre salão e cozinha (SOUZA, 2016).

Esse modelo tradicional apresenta diversas fragilidades: erros de anotação por parte dos garçons, ilegibilidade das comandas manuscritas, perda de pedidos durante o pico de movimento e falta de controle financeiro preciso. Além disso, a ausência de dados estruturados impede que o gestor tenha uma visão clara sobre os itens mais vendidos ou o faturamento real do dia.

Diante desse cenário, a tecnologia móvel surge como uma aliada estratégica. Porém, as soluções de software existentes no mercado muitas vezes são financeiramente inacessíveis para pequenos empreendedores ou exigem uma infraestrutura de hardware complexa (computadores, servidores locais e cabeamento) que não condiz com a realidade de pequenos espaços.

Este trabalho propõe o desenvolvimento do *Facilita Comanda*, uma aplicação móvel projetada especificamente para democratizar a automação em pequenos estabelecimentos. A solução busca eliminar o uso do papel, digitalizando o fluxo desde a anotação do pedido na mesa até a sua produção na cozinha, utilizando tecnologias modernas de desenvolvimento híbrido e computação em nuvem para oferecer uma ferramenta leve, de baixo custo e alta eficiência.

1.1 Motivação

A motivação para este trabalho nasce da observação direta das dificuldades enfrentadas por microempreendedores do ramo alimentício. Nota-se que, enquanto grandes redes de *fast-food* possuem sistemas sofisticados de KDS (*Kitchen Display System*) e autoatendimento, o pequeno comerciante ainda sofre com a desorganização das "notinhas de papel".

A perda de um único pedido em uma noite movimentada não gera apenas prejuízo financeiro, mas também o desgaste da imagem do estabelecimento perante o cliente. Além disso, a falta de inteligência de dados faz com que o dono do restaurante tome decisões baseadas apenas na intuição, sem saber exatamente qual produto lhe traz maior retorno ou qual é o seu ticket médio real.

O desenvolvimento de uma solução que une a flexibilidade dos dispositivos móveis (smartphones que os garçons já possuem) com a robustez de um backend em nuvem (*Serverless*) apresenta-se como uma resposta viável e necessária para modernizar esse segmento, reduzindo desperdícios e profissionalizando a gestão.

1.2 Objetivos

Esta seção apresenta os objetivos gerais e específicos que nortearam o desenvolvimento deste projeto.

1.2.1 Objetivo Geral

Desenvolver uma aplicação móvel para gerenciamento de pedidos e automação de cozinha, utilizando a plataforma *React Native* e serviços em nuvem (*Firebase*), visando substituir o fluxo de comandas de papel em pequenos restaurantes por um processo digital, híbrido e integrado.

1.2.2 Objetivos Específicos

Para alcançar o objetivo geral, foram definidos os seguintes objetivos específicos:

- Analisar os problemas recorrentes no fluxo de atendimento de pequenos estabelecimentos gastronômicos;

- Implementar um aplicativo móvel capaz de realizar o lançamento de pedidos, categorização de produtos e cálculo automático de valores;
- Desenvolver um módulo de "Fila da Cozinha" que permita o monitoramento dos pedidos em tempo real (KDS) e, simultaneamente, suporte a impressão térmica via Bluetooth para ambientes sem tablets;
- Criar um painel financeiro que forneça métricas essenciais, como faturamento diário, ticket médio e ranking de produtos mais vendidos;
- Utilizar uma arquitetura *Backend-as-a-Service* (BaaS) para garantir sincronização em tempo real e persistência segura dos dados na nuvem.

1.3 Organização do Documento

O restante deste documento está organizado da seguinte forma:

- O Capítulo 2 apresenta a metodologia utilizada no desenvolvimento do projeto, detalhando as ferramentas e tecnologias adotadas;
- O Capítulo 3 descreve a solução proposta, abordando a engenharia de requisitos, a arquitetura do software e a modelagem de dados no *Firestore*;
- O Capítulo 4 demonstra os resultados obtidos, apresentando as telas da aplicação e discutindo como as funcionalidades implementadas atendem aos problemas levantados;
- Por fim, o Capítulo 5 apresenta as considerações finais, limitações do projeto e sugestões para trabalhos futuros.

2. METODOLOGIA

Para o desenvolvimento do *Facilita Comanda*, adotou-se uma abordagem de pesquisa aplicada e tecnológica, com o objetivo de produzir uma solução de software funcional para um problema real. O processo de desenvolvimento seguiu os princípios das Metodologias Ágeis, permitindo iterações rápidas e ajustes constantes com base em testes práticos. Para a organização das tarefas, utilizou-se o método Kanban, que permite a visualização do fluxo de trabalho e a identificação de gargalos (ROBERTO, 2024).

As etapas da pesquisa foram organizadas da seguinte forma:

- **Levantamento Bibliográfico e Documental:** Realizou-se uma investigação sobre as tecnologias de desenvolvimento móvel híbrido (*React Native*) e arquiteturas *Serverless* (*Firebase*), além de uma análise sobre protocolos de impressão térmica (ESC/POS) para dispositivos Android.
- **Análise de Requisitos:** Foram mapeados os fluxos de trabalho de garçons e cozinheiros em lanchonetes locais para identificar os gargalos do processo manual (papel) e definir as funcionalidades essenciais do sistema (RF01 a RF08).
- **Prototipação e Design de Interface:** Criação dos *wireframes* e telas de alta fidelidade, priorizando a usabilidade em dispositivos móveis, conforme os princípios de design do dia a dia (NORMAN, 2013), e a clareza visual para ambientes de cozinha (letras grandes, cores para status).
- **Implementação e Codificação:** Desenvolvimento do aplicativo utilizando *JavaScript* (ES6+) e o ecossistema *React Native*. A integração com o *backend* foi realizada através do SDK do *Firebase*, implementando ouvintes (*listeners*) para sincronização em tempo real.
- **Testes e Validação:** Realização de testes de caixa-preta para validar o fluxo completo: criação do pedido, impressão térmica via Bluetooth e finalização na cozinha.

3. SOLUÇÃO PROPOSTA

Este capítulo descreve o Facilita Comanda, a aplicação móvel desenvolvida neste Trabalho de Conclusão de Curso. A solução tem como objetivo modernizar o processo de atendimento em pequenos restaurantes, substituindo comandas de papel por um fluxo digital integrado.

A aplicação foi desenvolvida utilizando a plataforma React Native (**META PLATFORMS, 2024**) para o *frontend* mobile e a suíte Google Firebase como *Backend-as-a-Service* (BaaS), garantindo sincronia de dados em tempo real entre o salão (garçons) e a cozinha.

3.1 Requisitos Funcionais

Esta seção descreve os requisitos funcionais implementados no sistema, ou seja, as funcionalidades que o software oferece aos seus usuários (garçons, cozinheiros e gerentes). Os requisitos foram levantados com base nas necessidades observadas no ambiente de restaurantes e estão listados no Quadro 1.

Identificador	Requisito	Descrição
RF01	Autenticar Usuários	O sistema deve permitir o login seguro de usuários, diferenciando permissões entre Garçons e Gerentes/Cozinha.
RF02	Gerenciar Produtos	O sistema deve permitir ao gerente cadastrar, editar e remover produtos do cardápio, incluindo preços e categorias.
RF03	Realizar Pedido	O sistema deve permitir que o garçom selecione itens, adicione observações (ex: "sem cebola") e calcule o total do pedido automaticamente.
RF04	Monitorar Fila da Cozinha	O sistema deve exibir, em tempo real, os pedidos lançados na cozinha, permitindo a visualização de detalhes e status ("Pendente", "Em Preparo", "Pronto").
RF05	Imprimir Comprovante	O sistema deve se comunicar com impressoras térmicas via Bluetooth/Rede para emitir a via física do pedido para o cozinheiro (utilizando protocolo RawBT).
RF06	Atualizar Status do Pedido	O sistema deve permitir que a cozinha altere o status do pedido (de "Pendente" para "Finalizado"), notificando a conclusão do preparo.
RF07	Visualizar Histórico	O sistema deve manter um registro de pedidos finalizados e cancelados para conferência financeira posterior.
RF08	Editar Pedido Ativo	O sistema deve permitir que um pedido ainda não finalizado seja alterado (adição/remoção de itens) caso o cliente mude de ideia.

3.2 Arquitetura

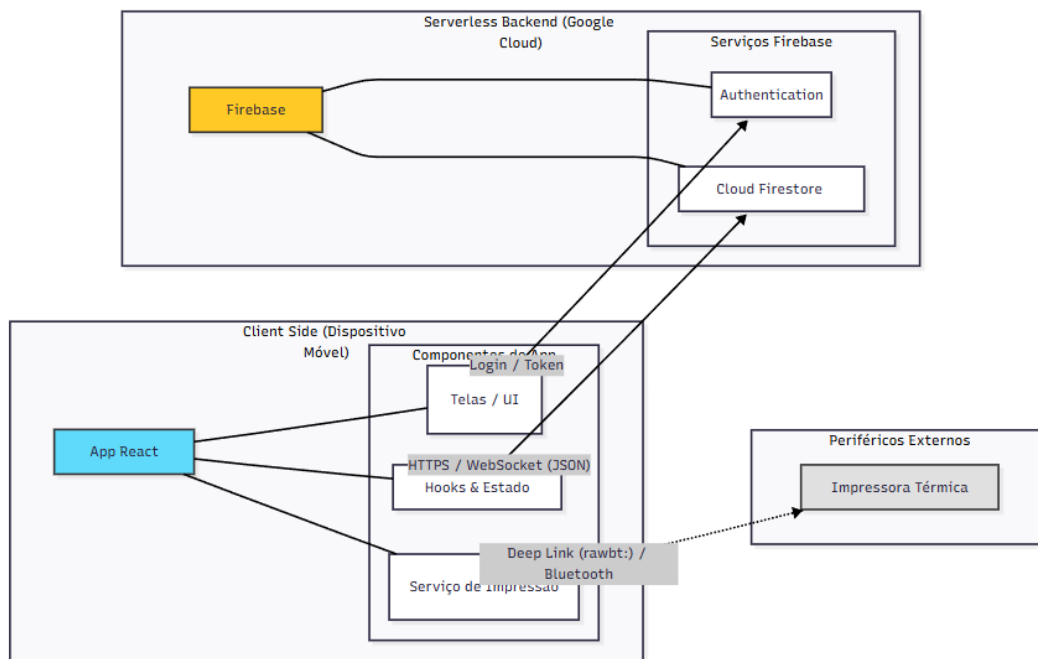
A arquitetura da solução proposta difere dos sistemas tradicionais baseados em servidores monolíticos. O *Facilita Comanda* adota uma arquitetura Cliente-Servidor moderna, utilizando o modelo de *Backend-as-a-Service* (BaaS).

A estrutura é dividida em três camadas principais de interação, conforme ilustrado na Figura 1:

- I. Camada de Apresentação (Frontend Mobile): Desenvolvida em React Native, é responsável pela interface com o usuário (Garçom/Cozinheiro). Nesta camada reside a lógica de interação local, gerenciamento de estado (via *Hooks* do React) e a comunicação direta com os periféricos do dispositivo, como o módulo Bluetooth para impressão térmica.
- II. Camada de Serviços em Nuvem (Backend): Gerenciada pela plataforma Google Firebase. Esta camada abstrai a complexidade da infraestrutura de servidores. O componente Firebase Authentication gerencia a identidade e permissões dos usuários, enquanto o Cloud Firestore atua como banco de dados e servidor de aplicação, processando regras de segurança e sincronização de dados.
- III. Camada de Integração Externa (Hardware): Responsável pela materialização dos pedidos. A comunicação entre o aplicativo móvel e a impressora térmica não ocorre via driver nativo tradicional, mas através de um protocolo de *Deep Linking* (rawbt:), que envia os dados formatados diretamente para o serviço de impressão instalado no dispositivo Android.

Esta arquitetura foi escolhida por permitir a sincronização em tempo real sem a necessidade de implementar *WebSockets* manualmente, garantindo que a tela da cozinha seja atualizada milissegundos após o garçom confirmar um pedido.

Figura 1 - Arquitetura da solução Facilita Comanda.



Fonte: Elaborado pelo autor (2026).

3.3 Modelagem do banco de dados

Diferente dos bancos de dados relacionais tradicionais (SQL), que utilizam tabelas e chaves estrangeiras rígidas, o *Facilita Comanda* utiliza o Google Cloud Firestore (GOOGLE, 2024), um banco de dados NoSQL orientado a documentos. Esta escolha justifica-se pela flexibilidade do esquema e pela capacidade de escalabilidade horizontal. Os dados são organizados em Coleções (equivalente a tabelas) que contêm Documentos (equivalente a linhas), armazenados no formato JSON.

3.3.1 Coleção de Usuários (users)

Armazena os dados de perfil e controle de acesso. A segurança é gerenciada via *Rules* do Firestore, garantindo que apenas gerentes possam editar cardápios.

- Campos principais: uid (ID único), email, role (cargo: 'gerente' ou 'garçom').

3.3.2 Coleção de Produtos (products)

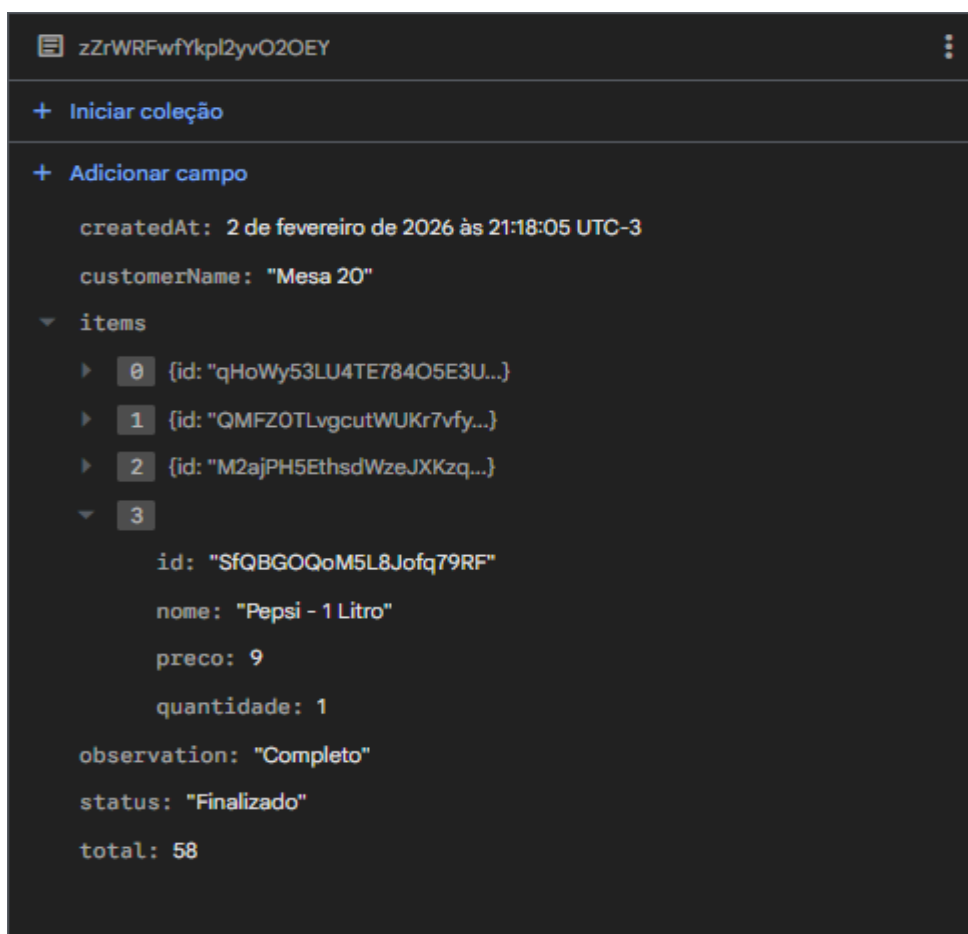
Representa o cardápio digital do restaurante.

- Campos principais: id, nome (String), preco (Number), categoria (String), observation (String).

3.3.3 Coleção de Pedidos (orders)

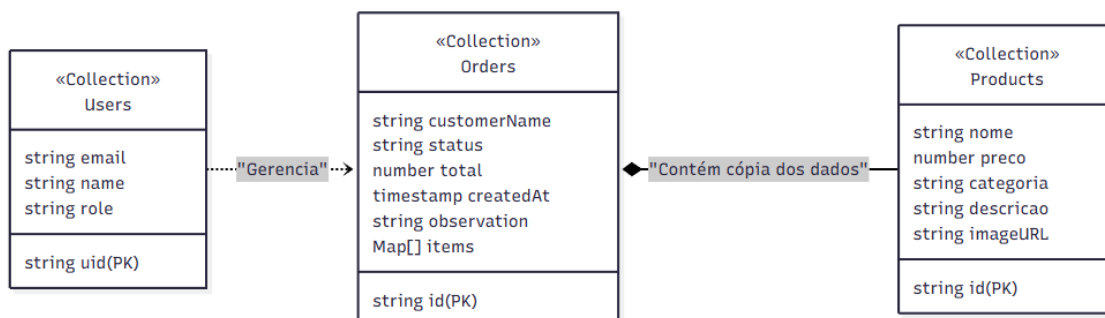
É a coleção central do sistema, onde ocorre a maior parte da transação de dados em tempo real. Cada documento representa uma comanda ativa ou finalizada. Abaixo, apresenta-se a estrutura JSON de um pedido típico no sistema:

Figura 2 - Documento JSON da coleção "Orders".



Fonte: Elaborado pelo autor (2026).

Figura 3 - Diagrama de classes das coleções do Cloud Firestore.



Fonte: Elaborado pelo autor (2026).

3.4 Implementação

A implementação do *Facilita Comanda* priorizou a performance e a experiência do usuário. Diferente de aplicações web tradicionais baseadas em requisições HTTP (Request/Response), o aplicativo mantém uma conexão persistente (WebSocket) com o banco de dados *Firestore*.

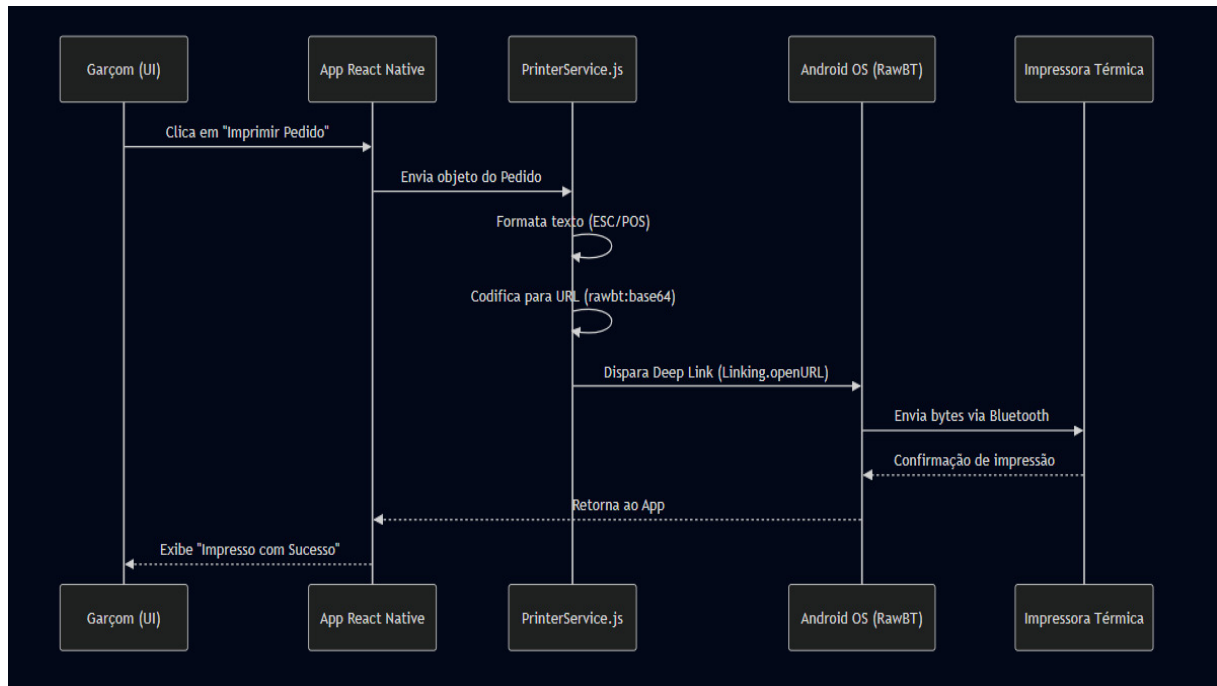
3.4.1 Gerenciamento de Estado e Sincronização

Utilizou-se a Context API do React combinada com *Hooks* personalizados (`useOrders`, `useAuth`) para gerenciar o estado global da aplicação. Quando um garçom lança um pedido, o *listener* `onSnapshot` do Firebase detecta a mudança no servidor e atualiza automaticamente a tela "Fila da Cozinha" em todos os dispositivos conectados, sem necessidade de recarregar a página.

3.4.2 Serviço de Impressão Térmica

Um dos maiores desafios técnicos foi a comunicação com impressoras térmicas genéricas via Bluetooth. Para contornar a falta de *drivers* nativos padronizados, implementou-se um serviço de *Deep Linking*. O aplicativo gera o layout da comanda (formatado com comandos ESC/POS para negrito e alinhamento) e envia esses dados codificados via URL (`rawbt:base64...`) para o serviço de *spooler* do Android, que gerencia a conexão física com a impressora.

Figura 4 - Diagrama de Sequência de Impressão.



Fonte: Elaborado pelo autor (2026).

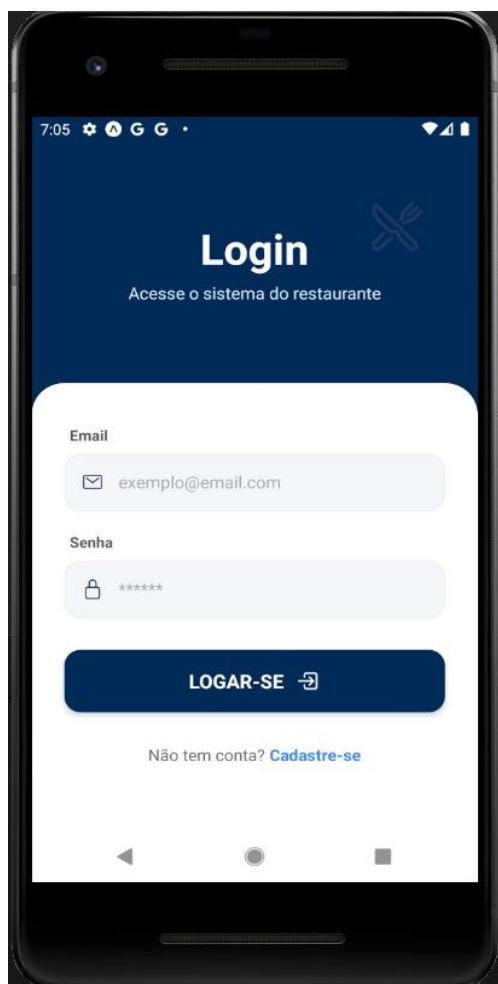
4. RESULTADOS E DISCUSSÃO

Após a finalização da codificação e a configuração do ambiente de *backend* no Firebase, a aplicação foi submetida a testes funcionais simulando o ambiente real de um restaurante. Esta seção apresenta as principais interfaces desenvolvidas e discute como cada uma soluciona os problemas de gestão levantados anteriormente.

4.1 Controle de Acesso e Segurança

A primeira interação do usuário com o sistema ocorre na tela de autenticação. Para garantir a segurança dos dados financeiros e do cardápio, o acesso é restrito. O sistema utiliza o *Firebase Authentication* para validar as credenciais. Conforme observado na Figura 5, a interface foi desenhada para ser limpa e direta, solicitando e-mail e senha.

Figura 5 - Tela de autenticação do usuário.

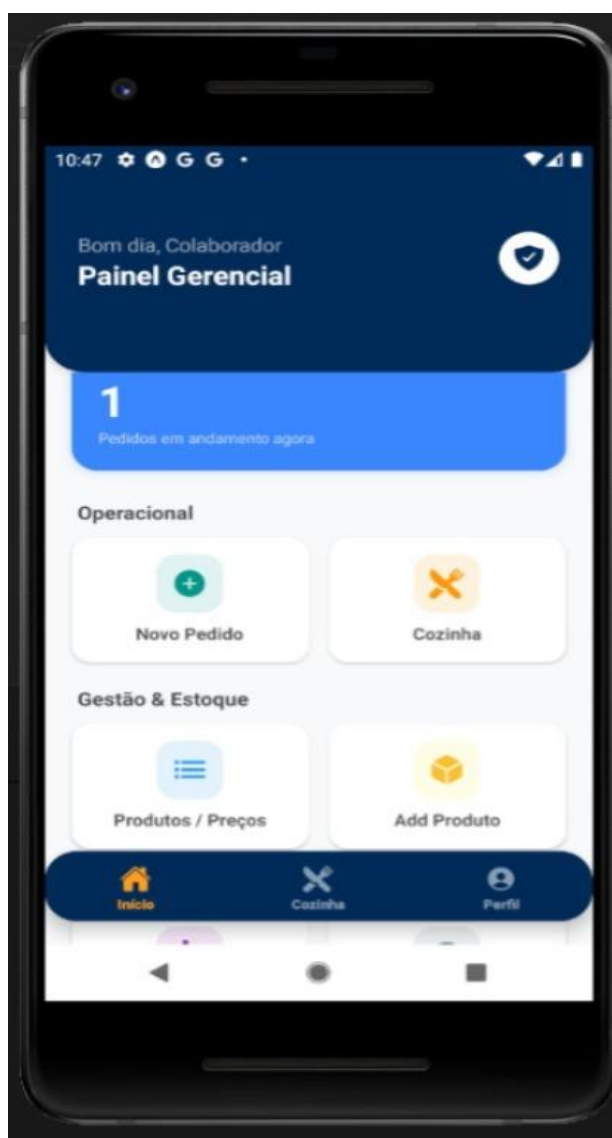


Fonte: Elaborado pelo autor (2026).

4.2 Gestão de Cardápio e Tela Inicial

Após o login, o garçom é direcionado para a Tela Inicial (Home), onde visualiza as categorias de produtos. A Figura 6 demonstra a organização visual dos itens. Diferente de listas de texto simples, optou-se por *cards* com imagens e preços destacados, o que facilita a identificação rápida do produto durante o atendimento pressuroso no salão.

Figura 6 - Tela inicial com visualização dos produtos.

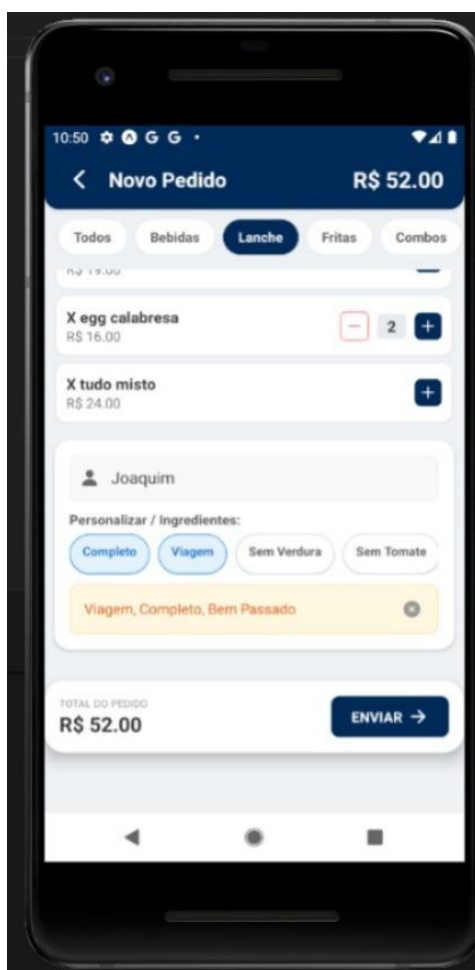


Fonte: Elaborado pelo autor (2026).

4.3 Fluxo de Lançamento de Pedidos

O processo de criação de um pedido foi otimizado para exigir o mínimo de toques na tela. Ao selecionar um produto, o garçom pode definir a quantidade e, crucialmente, adicionar observações específicas (ex: "Sem cebola", "Bem passado"). A Figura 7 ilustra o carrinho de compras antes da confirmação, onde o valor total é calculado automaticamente, eliminando erros de matemática comuns no fechamento manual.

Figura 7 - Detalhamento do pedido e inserção de observações.

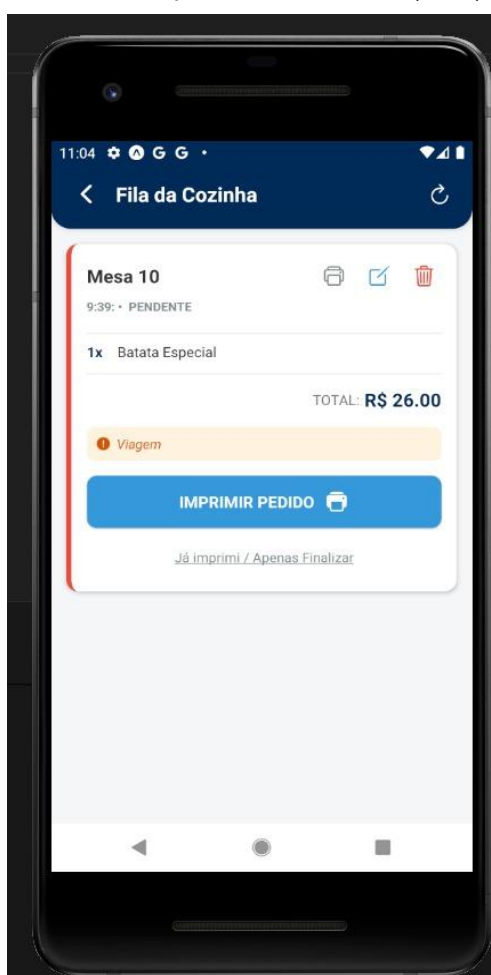


Fonte: Elaborado pelo autor (2026).

4.4 Fila da Cozinha (KDS)

Uma das principais inovações do *Facilita Comanda* é a substituição do grito ou do papel pela "Fila Digital". A Figura 8 apresenta a visão do cozinheiro (tablet ou celular na cozinha). Os pedidos chegam instantaneamente (via sincronização *Realtime*) e são exibidos em ordem cronológica.

Figura 8 - Fila de pedidos da cozinha (KDS).



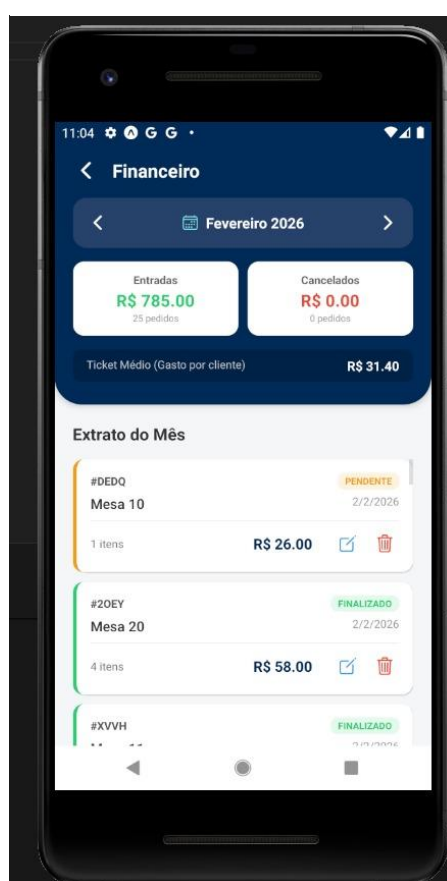
Fonte: Elaborado pelo autor (2026).

Note que cada cartão possui um botão de ação. Ao finalizar o prato, o cozinheiro altera o status, o que notifica o sistema e remove o item da lista de pendências, organizando o fluxo de produção.

4.5 Painel Financeiro e Tomada de Decisão

Por fim, para atender à necessidade de gestão do proprietário, foi desenvolvido o módulo financeiro. A Figura 9 exibe o *dashboard* com o "Ranking de Vendas", destacando os produtos mais populares e o faturamento total. Essa visualização transforma dados brutos em informação estratégica para o negócio.

Figura 9 - Relatório financeiro.



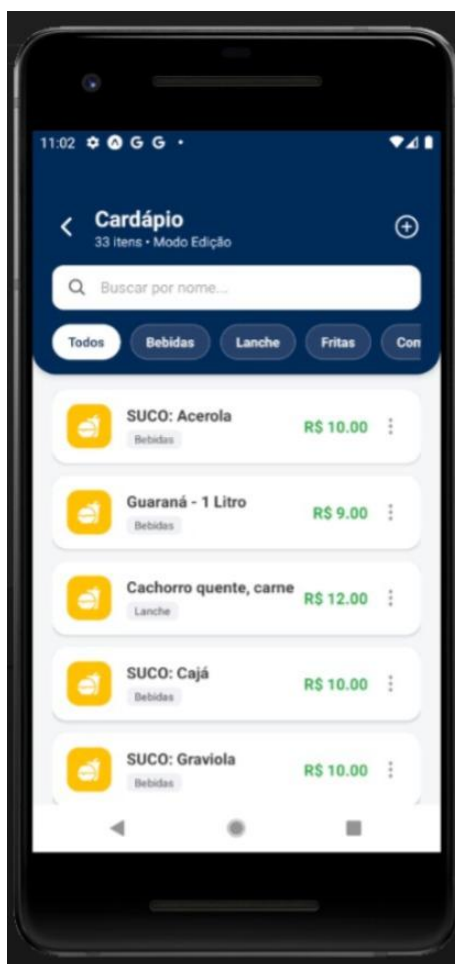
Fonte: Elaborado pelo autor (2026).

A implementação dessas interfaces confirmou a hipótese de que uma interface amigável, combinada com *feedback* visual imediato, reduz a curva de aprendizado da equipe e minimiza erros operacionais.

4.6 Gestão de Produtos e Estoque

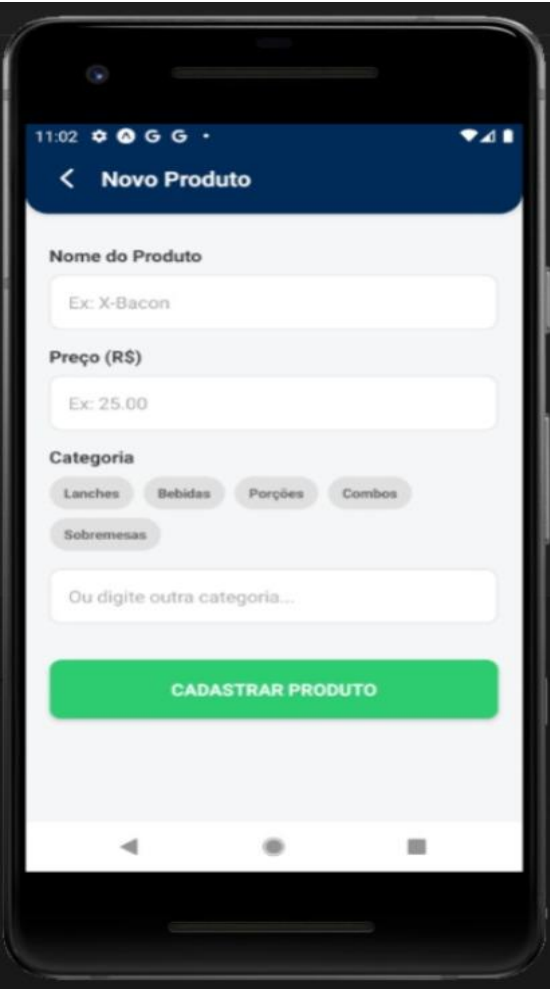
O sistema oferece flexibilidade total para a gestão do cardápio. Através das telas de "Produtos" e "Adicionar Produto", o gerente pode realizar operações de CRUD (*Create, Read, Update, Delete*). É possível cadastrar novos itens definindo nome, preço (seja unitário ou por variação, como "litrinho" ou "lata") e categoria. O sistema permite tanto a seleção de categorias existentes quanto a criação de novas categorias manualmente, garantindo que o cardápio possa crescer conforme a oferta do restaurante se expande (ex: inclusão de novas cervejas ou pratos).

Figura 10 – Listagem de produtos cadastrados.



Fonte: Elaborado pelo autor (2026).

Figura 11 - Formulário de cadastro de novos produtos.

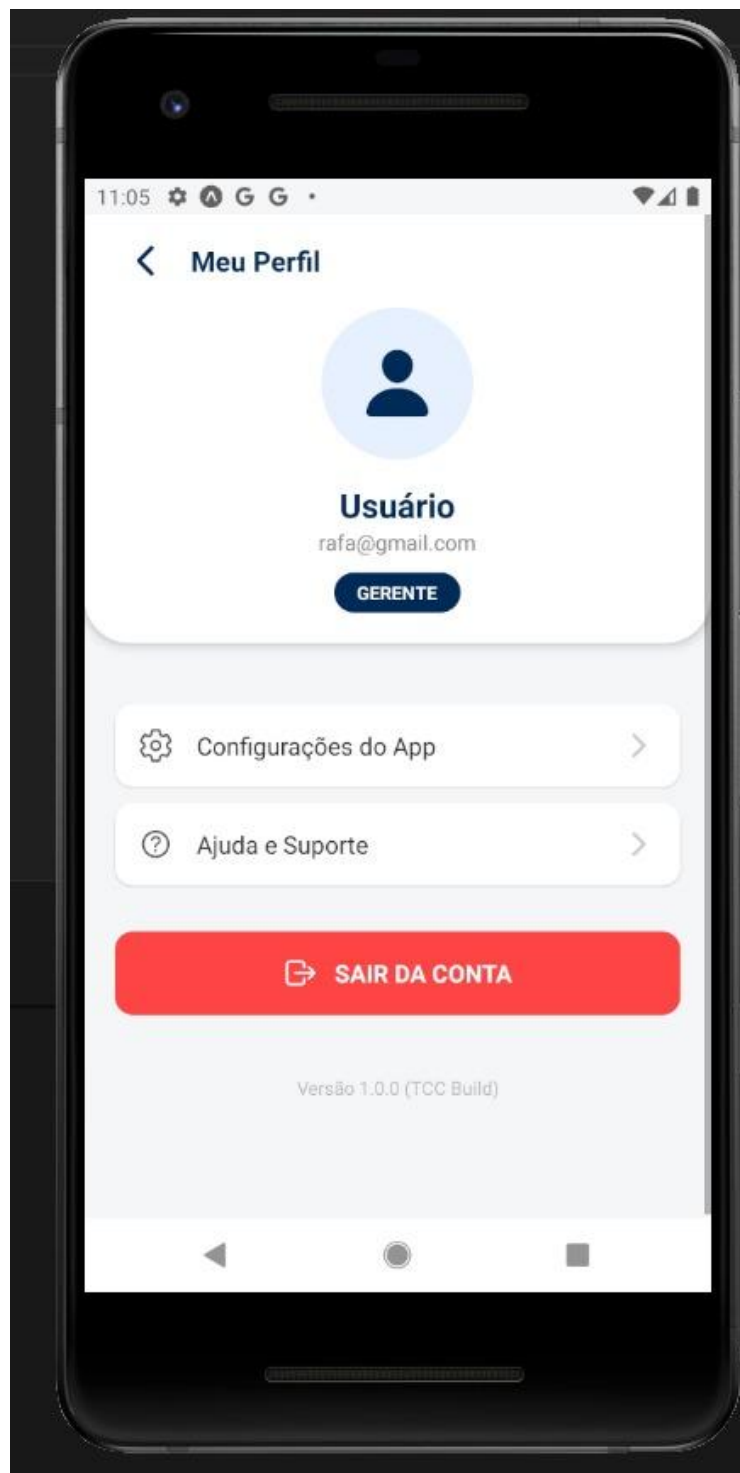


Fonte: Elaborado pelo autor (2026).

4.7 Perfil de Usuário e Controle de Nível

Por fim, a tela de Perfil tem a função crítica de auditoria e segurança. Ela exhibe as credenciais do usuário logado e, principalmente, sua função no sistema (*Role*). Essa distinção visual confirma se o usuário possui privilégios de "Gerente" (acesso total) ou "Garçom" (acesso restrito), validando os requisitos de segurança do software.

Figura 12 - Identificação do usuário e nível de acesso.



Fonte: Elaborado pelo autor (2026).

5. CONSIDERAÇÕES FINAIS

A automação de processos em pequenos estabelecimentos gastronômicos é um passo fundamental para a sobrevivência e crescimento desses negócios. O desenvolvimento do *Facilita Comanda* demonstrou que é possível criar uma solução tecnológica robusta, eficiente e de baixo custo utilizando ferramentas modernas como *React Native* e *Firebase*.

Os resultados obtidos com os testes do aplicativo indicam que a solução atende plenamente aos objetivos propostos. O sistema eliminou a dependência de comandas de papel, reduziu o tempo de comunicação entre o salão e a cozinha e proporcionou, pela primeira vez, uma visão gerencial baseada em dados (Ranking de Vendas) para o administrador.

A abordagem híbrida adotada — permitindo que a cozinha opere tanto com telas digitais (KDS) quanto com comandas impressas — mostrou-se um diferencial importante, garantindo a aderência do sistema a diferentes realidades de infraestrutura.

5.1 Trabalhos Futuros

Alguns trabalhos futuros podem ser desenvolvidos para a evolução do trabalho proposto neste TCC. Exemplos de evoluções importantes incluem:

- **Implementação de Cardápio Digital (QR Code):** Desenvolvimento de uma interface web para que os clientes realizem o pedido na própria mesa, lendo um QR Code, sem necessidade de chamar o garçom.
- **Automação de Delivery via WhatsApp:** Desenvolvimento de um módulo de integração externa (*chatbot*) que permita aos clientes realizarem pedidos diretamente pelo aplicativo de mensagens. A arquitetura proposta prevê que, após a confirmação do pagamento via API, o *bot* insira o pedido diretamente no banco de dados *Firestore*, acionando a impressão automática na cozinha sem intervenção humana.
- **Painel Administrativo Web:** Criação de um *dashboard* para desktop, permitindo ao gerente visualizar relatórios financeiros avançados, curvas de vendas e gestão de estoque em telas maiores.

REFERÊNCIAS

- BARBOSA, Adriley Samuel Ribeiro. **Análise comparativa entre os padrões MVC, MVP, MVVM e MV na plataforma Android**. 2022. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) - Instituto Federal Goiano, Ceres, 2022. Disponível em: https://repositorio.ifgoiano.edu.br/bitstream/prefix/2611/1/tcc_Adriley%20Samuel%20Ribeiro%20Barbosa.pdf. Acesso em: 20 nov. 2024.
- RAWBT. **RawBT print service: API for developers**. 2024. Disponível em: <https://rawbt.ru/api.html>. Acesso em: 06 dez. 2026.
- GOOGLE. **Cloud Firestore: NoSQL Cloud Database**. Documentation. 2024. Disponível em: <https://firebase.google.com/docs/firestore>. Acesso em: 03 fev. 2025.
- GOOGLE. **Firebase Documentation: Fundamentals**. 2024. Disponível em: <https://firebase.google.com/docs>. Acesso em: 03 fev. 2025.
- KARNEKE, Ana Maria; KEIJO, Antonio Roberto. **O impacto da automação dos serviços logísticos na produtividade de uma distribuidora: estudo de caso**. Recife: Grupo Unibra, 2022. Disponível em: <https://www.grupounibra.com/repositorio/ADMIN/2022/o-impacto-da-automacao-dos-servicos-logisticos-na-produtividade-de-uma-distribuidora-estudo-de-caso-karnekeijo57.pdf>. Acesso em: 05 nov. 2024.
- MELO, Caio Felipe de Alcântara. **Bom Delivery: um aplicativo de delivery genérico para cidades do interior**. 2019. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Universidade Federal de Campina Grande, Campina Grande, 2019. Disponível em: <https://dspace.sti.ufcg.edu.br/handle/riufcg/20349>. Acesso em: 11 dez. 2024.
- META PLATFORMS. **React Native: Learn once, write anywhere**. Documentation. 2024. Disponível em: <https://reactnative.dev/docs/getting-started>. Acesso em: 03 fev. 2025.
- NORMAN, Donald Arthur. **O design do dia a dia**. 2. ed. Rio de Janeiro: Rocco, 2013.
- SOUZA, J. A. **Conect Food: um estudo sobre a automação no setor de alimentação**. 2016. Trabalho de Conclusão de Curso - Centro Paula Souza, São Paulo, 2016. Disponível em: https://ric.cps.sp.gov.br/bitstream/123456789/19036/1/20162S_Conect%20Food%20TCC.pdf. Acesso em: 13 nov. 2024.

APÊNDICE A

```

myApp > src > services > PrinterService.js > printOrder
1  import { Linking, Alert } from 'react-native';
2
3  export const printOrder = async (order) => {
4    try {
5      const date = order.createdAt
6        ? new Date(order.createdAt.seconds * 1000).toLocaleString('pt-BR')
7        : new Date().toLocaleString('pt-BR');
8
9      const orderId = order.id ? order.id.slice(-4).toUpperCase() : '-----';
10     const customer = order.customerName || 'Balcão';
11
12     let receipt = '';
13
14     // 1. Cabeçalho
15     receipt += '[C]CARIOCA LANCHES\n';
16     receipt += '[C]-----\n';
17     receipt += '[L]Data: ${date}\n';
18     receipt += '[L]Cliente: ${customer}\n';
19     receipt += '[L]Pedido- COD: ${orderId}\n';
20     receipt += '[C]-----\n';
21     receipt += '[L]QTD  ITEM                VALOR\n';
22
23     // 2. Itens do Pedido
24     if (order.items && Array.isArray(order.items)) {
25       order.items.forEach(item => {
26
27         const qtd = `${item.quantidade}x`.padEnd(4, ' ');
28
29         let name = item.nome ? item.nome.substring(0, 18) : 'Item s/ nome';
30         name = name.padEnd(19, ' ');
31
32         const totalItem = (const totalItem: string = (item.quantidade)).toFixed(2);
33         const price = `R${totalItem}`;
34
35         receipt += `[L]${qtd}${name}${price}\n`;
36       });
37     }
38
39     receipt += '[C]-----\n';
40
41     const totalValue = order.total ? parseFloat(order.total).toFixed(2) : '0.00';
42     receipt += `[R]TOTAL: R$ ${totalValue}\n`;
43     receipt += '[C]-----\n';
44
45     if (order.observation && order.observation.trim().length > 0) {
46       receipt += '[L]--- OBSERVACOES ---\n';
47       receipt += `[L]${order.observation}\n`;
48       receipt += '[C]-----\n';
49     }
50
51     receipt += '[C]Obrigado pela preferencia!\n';
52     receipt += '\n\n';
53
54     const encodedReceipt = encodeURIComponent(receipt);
55
56     const url = `rawbt:${encodedReceipt}`;
57
58     const supported = await Linking.canOpenURL(url);
59
60     if (supported) {
61       await Linking.openURL(url);
62     } else {
63       Alert.alert(
64         "Driver não instalado",
65         "Instale o app 'RawBT' na Play Store para imprimir."
66       );
67     }
68
69   } catch (error) {
70     console.error("Erro na impressão:", error);
71     Alert.alert("Erro", "Falha ao gerar comando de impressão.");
72   }
73 }
74

```

	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
	Campus Cajazeiras - Código INEP: 25008978
	Rua José Antônio da Silva, 300, Jardim Oásis, CEP 58.900-000, Cajazeiras (PB)
	CNPJ: 10.783.898/0005-07 - Telefone: (83) 3532-4100

Documento Digitalizado Restrito

Trabalho de Conclusão de Curso - Thalyson Rian

Assunto:	Trabalho de Conclusão de Curso - Thalyson Rian
Assinado por:	Thalyson Rian
Tipo do Documento:	Anexo
Situação:	Finalizado
Nível de Acesso:	Restrito
Hipótese Legal:	Informação Pessoal (Art. 31 da Lei no 12.527/2011)
Tipo do Conferência:	Cópia Simples

Documento assinado eletronicamente por:

- Thalyson Rian Mendes da Silva, DISCENTE (202022010014) DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - CAJAZEIRAS, em 20/02/2026 16:42:14.

Este documento foi armazenado no SUAP em 20/02/2026. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1772819
Código de Autenticação: 3d0557a694

