

# **CookBook: Aplicativo para cadastro, organização e busca de receitas culinárias**

Área de Conhecimento de Ciências Exatas e Engenharia – Universidade de Caxias do Sul (UCS) – Caxias do Sul – RS – Brasil

**Resumo.** Este artigo apresenta o desenvolvimento do aplicativo CookBook, criado para facilitar o cadastro, organização e busca de receitas culinárias. O projeto foi implementado com Flutter e Dart, utilizando banco de dados externo PostgreSQL, e segue os princípios da engenharia de software aplicada ao desenvolvimento mobile.

**Abstract.** This article presents the development of CookBook, an application designed to simplify the registration, organization, and search of culinary recipes. The project was built using Flutter and Dart, connected to an external PostgreSQL database, following software engineering principles for mobile development.

## **1. Introdução**

O aplicativo CookBook é uma solução móvel desenvolvida em Flutter, com o objetivo de facilitar o cadastro, organização e busca de receitas culinárias. Ele permite que os usuários armazenem suas receitas de maneira estruturada e intuitiva, categorizando-as e adicionando tags personalizadas. Dessa forma, o aplicativo se torna um repositório pessoal de receitas, substituindo anotações dispersas em papel, capturas de tela ou arquivos desorganizados. A interface amigável permite acessar, organizar e compartilhar receitas com praticidade.

Para o desenvolvimento deste aplicativo, a escolha foi Dart, Flutter e Postgresql para banco de dados. Dart é uma linguagem otimizada para o cliente, voltada à criação de aplicações rápidas em qualquer plataforma (DART, 2024). Seu principal objetivo é proporcionar uma experiência produtiva e consistente para o desenvolvimento multiplataforma, com suporte robusto a estruturas de aplicações, recarregamento rápido (hot reload) e uma interface acessível e intuitiva. O Flutter, por sua vez, é um framework que permite a construção de interfaces responsivas e modernas com alta performance, sendo ideal para o contexto de dispositivos móveis como o Android. A linguagem Dart também se destaca por seu sistema de tipagem segura (type safe), utilizando verificação de tipo estático, o que garante maior confiabilidade no desenvolvimento do código. Essa combinação tecnológica contribui para um fluxo de trabalho ágil e organizado, aspectos essenciais em projetos acadêmicos e profissionais (Flutter documentation, 2024).

Com a crescente popularização de aplicativos móveis e o interesse por uma alimentação mais saudável e prática, muitos usuários têm buscado soluções digitais que auxiliem no gerenciamento de receitas caseiras. O CookBook surge como uma alternativa moderna e acessível para esse público, aliando facilidade de uso, personalização e funcionalidades que otimizam o cotidiano na cozinha.

O principal problema que o CookBook busca resolver é a dificuldade de manter receitas organizadas e facilmente acessíveis. Muitas pessoas acumulam receitas em formatos variados e acabam perdendo ou esquecendo onde as armazenaram. Ao centralizar todas as receitas em um único local e oferecer opções de busca por nome, categorias e tags, o aplicativo otimiza a experiência do usuário, garantindo praticidade e eficiência no planejamento das refeições. Entre os benefícios oferecidos, destacam-se a facilidade para adicionar e organizar receitas, a busca intuitiva por pratos específicos e o armazenamento seguro das informações em um banco de dados externo.

## **2. Compreendendo o Problema e Levantando os Requisitos**

A maioria das pessoas, ao longo do tempo, reúne diversas receitas culinárias provenientes de livros, sites, redes sociais e trocas pessoais. Entretanto, essas receitas muitas vezes acabam sendo esquecidas, desorganizadas ou até mesmo perdidas. Em um contexto onde o tempo é escasso e a praticidade é essencial, a falta de um local centralizado e de fácil acesso compromete a experiência de cozinhar, especialmente para quem busca inovação e variedade no cardápio diário. O CookBook propõe-se a solucionar esse problema oferecendo uma plataforma digital completa e funcional para que o usuário tenha total controle sobre seu acervo culinário.

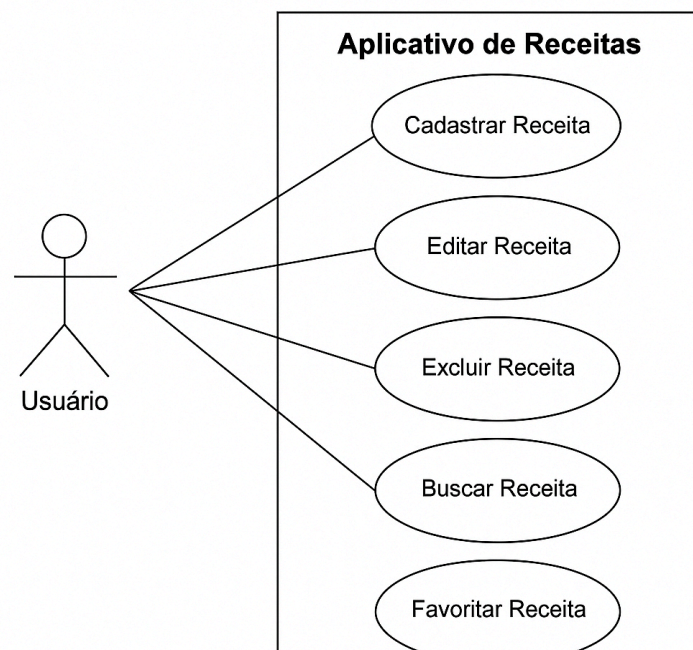
Segundo Wazlawick (2004), os requisitos funcionais são funções realizadas com o conhecimento do usuário, correspondendo a entradas de dados, consultas e interações na interface. Já os requisitos não funcionais dizem respeito a restrições de desempenho, usabilidade ou tecnologias utilizadas, como tempo de resposta, tipo de dispositivo ou banco de dados adotado. Compreender esses dois tipos de requisitos é fundamental para a estruturação do projeto, uma vez que norteiam tanto a arquitetura quanto a experiência do usuário. Ainda conforme o autor, o uso do diagrama de casos de uso auxilia na visualização

do escopo do sistema, facilitando a tomada de decisão quanto ao que deve ou não ser automatizado.

Como requisitos funcionais, o sistema deve permitir o cadastro de receitas contendo nome, lista de ingredientes, passo a passo para preparo, tempo de preparo, observações, categorias e tags. Deve oferecer a funcionalidade de criação de categorias personalizadas e disponibilizar categorias pré-definidas para facilitar a organização. Além disso, o aplicativo deve permitir a busca de receitas por nome, categoria e tags, também é possível fazer a edição e exclusão das receitas e visualizar receitas cadastradas com filtros.

Em relação aos requisitos não funcionais, o aplicativo será desenvolvido utilizando o Flutter e a linguagem Dart, garantindo desempenho e compatibilidade com dispositivos Android. O armazenamento será realizado em banco de dados externo com PostgreSQL. O código do projeto estará disponível no GitHub, e a arquitetura do software seguirá o padrão MVVM, proporcionando uma separação clara entre as camadas de interface, lógica de negócio e persistência de dados.

## Diagrama de Casos de Uso



**Figura 1. Diagrama de casos de Uso**

O diagrama de casos de uso (Figura 1) representa os principais fluxos de interação entre o usuário e o aplicativo CookBook, evidenciando as funcionalidades essenciais do sistema. A figura apresenta um único ator (o usuário), que interage diretamente com o sistema para realizar ações relacionadas ao gerenciamento de receitas culinárias.

### **3. Realização dos casos de uso: (Modelo Larman)**

Essencial para mapeamento dos fluxos principais do aplicativo, o diagrama de casos de uso facilita a comunicação entre as partes interessadas no desenvolvimento do CookBook e planejamentos das ações e funções que o mesmo deve desempenhar a partir da visualização de como o usuário irá interagir com o sistema. Os casos de uso tem como objetivo evidenciar os requisitos e necessidades dos usuários, assegurando uma experiência tanto para implementação quanto para os testes do sistema. Abaixo, podemos ver a definição das principais funcionalidades do nosso aplicativo

### **4. Funcionalidades**

O aplicativo permitirá a criação, edição e exclusão de receitas, possibilitando também a personalização com categorias e tags. As receitas poderão ser listadas e filtradas de acordo com suas categorias ou palavras-chave. Como funcionalidade adicional, o sistema oferecerá suporte à integração com o banco de dados externo PostgreSQL, permitindo sincronização remota de dados, backup automático e integração com versões web futuras.

### **5. Protótipo Visual**

O protótipo do aplicativo foi elaborado com foco na clareza e simplicidade da navegação. A tela inicial apresenta uma lista de receitas cadastradas e permite a busca por nome, categoria ou tag. Um menu inferior permite acesso às telas de criação de receita, categorias, favoritos e configurações.

As receitas podem ser visualizadas em detalhe, com todos os seus ingredientes e o modo de preparo descrito por um passo a passo. O usuário poderá favoritar uma receita diretamente dessa visualização, além de editá-la ou removê-la.

## 6. Diagrama de Classes

O diagrama de classes define a estrutura lógica do sistema, incluindo as principais entidades e seus relacionamentos. A receita contém título, lista de ingredientes, preparo, tempo, observações, categoria e tags. O ingrediente representa os ingredientes de uma receita, com nome e quantidade. A categoria define o tipo da receita. A tag são palavras chaves associadas à receita.

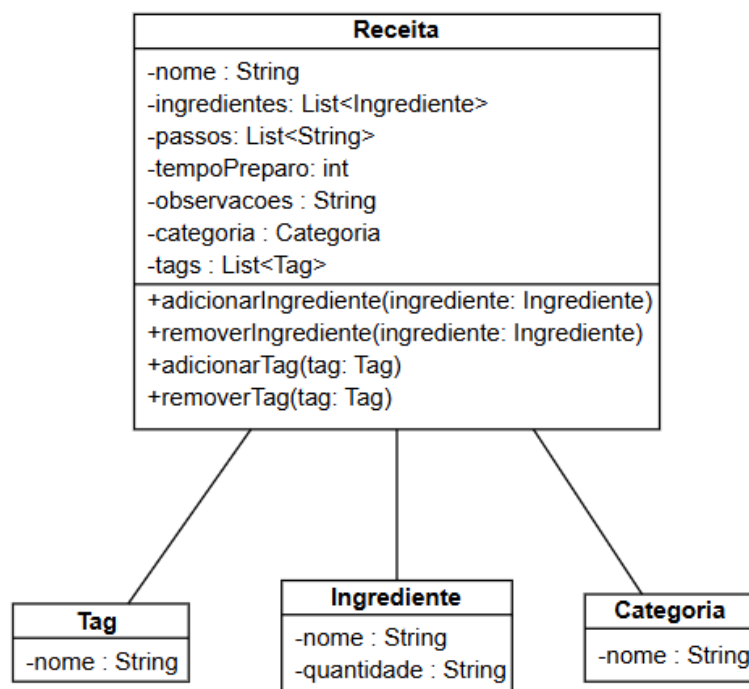


Figura ?. Diagrama de Classes

## 6. Organização de Software

O software será estruturado em três camadas principais: a camada de apresentação (interface do usuário), a camada de lógica de negócio e a camada de persistência de dados. Essa organização segue o modelo arquitetural em camadas proposto por Sommerville (2018), que destaca a importância da separação de responsabilidades para facilitar a manutenção, a extensibilidade e o reuso de componentes do sistema.

A camada de apresentação será desenvolvida com o framework Flutter, conhecido por sua capacidade de produzir interfaces responsivas, intuitivas e visualmente consistentes com padrões modernos de design, como o Material Design. O Flutter atua sobre a linguagem Dart, que é otimizada para aplicações cliente e oferece recursos como tipagem segura, inferência de tipos, e recarregamento a quente, proporcionando agilidade ao ciclo de desenvolvimento (DART, 2024).

A camada de lógica de negócio gerenciará as operações entre a interface do usuário e a base de dados, incluindo validação dos dados inseridos, aplicação das regras do sistema e controle do fluxo das operações.

A camada de persistência, por sua vez, utilizará o banco de dados relacional PostgreSQL, reconhecido por sua robustez, escalabilidade e conformidade com o padrão SQL. Segundo Momjian (2023), o PostgreSQL é um sistema de banco de dados de código aberto que oferece suporte avançado a transações, integridade referencial e extensibilidade por meio de funções personalizadas e tipos de dados definidos pelo usuário, tornando-o uma escolha sólida para sistemas que exigem desempenho e consistência.

Essa arquitetura não apenas facilita a manutenção do sistema, mas também permite que novas funcionalidades sejam adicionadas de forma modular, com baixo acoplamento entre as partes. Isso é fundamental para aplicações que podem evoluir com o tempo, como o CookBook, que poderá futuramente incorporar recursos como autenticação de usuários ou compartilhamento de receitas.

## **7. Conclusão**

O desenvolvimento do aplicativo CookBook representa uma proposta sólida para solucionar um problema cotidiano: a organização de receitas culinárias de maneira prática, personalizada e acessível. Com base em uma arquitetura robusta e na utilização de

tecnologias modernas como Dart, Flutter e PostgreSQL, o projeto se destaca pela clareza na definição dos requisitos e pela estrutura modular que favorece a manutenção e evolução contínua do sistema.

A aplicação dos princípios de engenharia de software garantiu um planejamento bem estruturado, desde o levantamento de requisitos até o desenho da arquitetura e escolha das tecnologias. Além disso, a utilização de bibliotecas específicas e o gerenciamento eficiente de estado reforçam a qualidade do código e a experiência do usuário final.

De maneira geral, o desenvolvimento desta aplicação teve como objetivo colocar em prática os conhecimentos adquiridos nas disciplinas de dispositivos mobile, fundamentos de engenharia de software e organização e arquitetura de software.

Ademais, entende-se que a ideia pode ser aprimorada, com uma interface mais completa e mais botões e funções. Numa possível atualização, o aplicativo poderia incorporar autenticação de usuários, compartilhamento de receitas e sincronização entre dispositivos. Dessa forma, o CookBook apresentaria não apenas como um projeto acadêmico, mas também como uma solução viável e escalável para o mundo real.

Em resumo, os objetivos estabelecidos no início da disciplina foram alcançados com êxito, e o resultado final do aplicativo é satisfatório

## **Referências**

SOMMERVILLE, Ian. Engenharia de software. 10. ed. São Paulo: Pearson, 2018.

DART. Dart overview. Disponível em: <<https://dart.dev/overview>>. Acesso em: 30 de março de 2025.

Flutter documentation. Disponível em: <<https://docs.flutter.dev>>. Acesso em 15 de abril de 2025.

Momjian, B. (2023). *PostgreSQL: Introduction and Concepts*. Addison-Wesley.