



Sistema de Gerenciamento de Receitas Culinárias



1. Introdução

Este projeto tem como finalidade auxiliar cozinheiros, estudantes de gastronomia e entusiastas da culinária a organizar e gerenciar receitas de forma prática. A aplicação foi desenvolvida em Java, utilizando Swing para a interface gráfica e MySQL como banco de dados.

A integração entre a aplicação e o banco de dados é feita com JDBC (Java Database Connectivity), e o projeto é desenvolvido na IDE Apache NetBeans.



2. Objetivos

Objetivo Geral

Desenvolver um sistema desktop que permita o gerenciamento de receitas culinárias por meio de operações CRUD.

Objetivos Específicos

- Criar uma interface gráfica simples e intuitiva com Java Swing.
- Implementar o cadastro de receitas com os campos:
 - Código
 - Ingredientes
 - Modo de preparo
 - Tempo de preparo
 - Categoria
- Permitir a edição, exclusão e consulta de receitas.
- Garantir que todas as receitas sejam salvas em um banco de dados MySQL.
- Aplicar conceitos de Programação Orientada a Objetos (POO) e tratamento de exceções.



3. Requisitos



3.1 Requisitos Funcionais

- Cadastrar receitas com todos os dados.
- Listar receitas cadastradas.
- Alterar dados de uma receita existente.
- Excluir receitas.
- Pesquisar receitas por nome ou categoria.



3.2 Requisitos Não Funcionais

- Interface gráfica amigável e de fácil uso.
- Linguagem de programação: Java (Swing).
- Banco de dados: MySQL.
- Implementação de tratamento de exceções.
- Estrutura de código com conceitos de POO.



4. Arquitetura do Sistema

O sistema é dividido em dois projetos independentes (backend e frontend) que se comunicam via rede.



Back-end (API RESTful)

- **Responsabilidade:** Gerenciar todas as regras de negócio e o acesso aos dados.
- **Camada de Controle (Controller):** Expõe os endpoints da API (ex: /receitas) usando SparkJava.
- **Camada de Persistência (DAO):** Realiza as operações de CRUD no banco de dados MySQL utilizando JDBC.
- **Formato de Dados:** Comunica-se com o front-end enviando e recebendo dados no formato JSON, com o auxílio da biblioteca Gson.



Front-end (Cliente Desktop)

- **Responsabilidade:** Exibir a interface gráfica e interagir com o usuário.
- **Camada de Apresentação (View):** Telas, tabelas e formulários desenvolvidos com Java Swing.
- **Cliente API:** Uma camada responsável por fazer as chamadas HTTP (requisições) para a API do back-end, buscando e enviando dados. Não possui acesso direto ao banco de dados.

Banco de Dados: Estrutura da Tabela receitas

Campo	Tipo	Descrição
id	INT (PK)	Identificação
nome	VARCHAR	Nome da receita
ingredientes	TEXT	Ingredientes
modo_preparo	TEXT	Modo de preparo
tempo_preparo	INT	Tempo de preparo
categoria	VARCHAR	Categoria



5. Etapas de Desenvolvimento

01



Etapa 1 – Preparação

- Instalar MySQL e criar o banco de dados.
- Instalar e configurar o NetBeans com suporte a Java.
- Baixar o conector JDBC: mysql-connector-java.

02



Etapa 2 – Estrutura do Código

- **Responsabilidade:** Gerenciar todas as regras de negócio e o acesso aos dados.
- **Camada de Controle (Controller):** Expõe os endpoints da API (ex: /receitas) usando SparkJava.
- **Camada de Persistência (DAO):** Realiza as operações de CRUD no banco de dados MySQL utilizando JDBC.
- **Formato de Dados:** Comunica-se com o front-end enviando e recebendo dados no formato JSON, com o auxílio da biblioteca Gson.

03



Etapa 3 – Funcionalidades

- Formulário Swing para cadastrar receitas.
- Tabela para listar receitas do banco.
- Botões para alterar, excluir e consultar.
- Tratamento de exceções com mensagens para o usuário.

04



Etapa 4 – Testes

- Testar todas as operações CRUD.
- Testar falhas de conexão e entradas inválidas.



6. Divisão de Tarefas

Integrante	Responsabilidade
Gabriel N	Frontend,Filtro,Criação de endpoint,Experiencia do us
Victor Cesar	Desenvolvimento,infraestrutur end,integração com JDBC e c com frontend
Liédson	Backend ,Classe Receita e lógica negócio (POO)
Israel	Backend, Dao conexao e inter deletar receita,Slides



7. Conclusão

O Sistema de Gerenciamento de Receitas Culinárias é uma aplicação desktop desenvolvida em Java, com interface gráfica em Swing e persistência de dados em MySQL.

Além de cumprir os requisitos da disciplina de Linguagem de Programação II, o projeto proporciona uma aplicação prática de conceitos como:

- Programação Orientada a Objetos
- Tratamento de Exceções
- Integração com Banco de Dados

Tecnologias Utilizadas



Java Server Faces / client-server

Feito com GAMMA



Swing (Java GUI)



MySQL



JDBC



Apache NetBeans



Git / GitHub



Como Executar o Projeto (Passo a Passo)

Para executar o sistema completo, você precisa rodar o banco de dados, o back-end e o front-end, nesta ordem.

Pré-requisitos

- Java Development Kit (JDK) 11 ou superior
- Apache NetBeans IDE 12 ou superior
- MySQL Server e MySQL Workbench (ou outro cliente SQL)
- Git instalado

Passo 1: Clone o Repositório

Abra o terminal (Git Bash, CMD, etc.) e clone o projeto para sua máquina:

```
`` bash
git clone
https://github.com/VeectorVt/sistema-receitas-java.git
``
```

Passo 2: Prepare o Banco de Dados

Dados

1. Abra o MySQL Workbench e conecte-se ao seu servidor de banco de dados.
2. Abra o arquivo de script SQL que está na pasta do projeto (ex: script_banco.sql).
3. Execute o script. Ele irá criar o banco receitas_db, as tabelas categoria e receita, e inserir alguns dados iniciais.

Passo 3: Execute o Back-end (A API)

1. Abra o NetBeans.
2. Vá em **Arquivo > Abrir Projeto...** e selecione a pasta backend dentro do projeto clonado.
3. **Configuração Crítica:** Abra o arquivo backend/src/main/java/com/mycompany/receitas_culinarias/conexao/Conexao.java e altere o usuário e a senha para as suas credenciais do MySQL.
4. Clique com o botão direito no projeto backend na lista de projetos e selecione **Limpar e Construir**. Aguarde a mensagem BUILD SUCCESS.
5. Encontre o arquivo Main.java (ou Receitas_Culinarias.java) do projeto backend.
6. Clique com o botão direito nele e selecione **Executar Arquivo**.
7. No console de saída do NetBeans, você deverá ver a mensagem: Servidor da API de Receitas iniciado em http://localhost:8080.
8. **Verificação (Opcional):** Abra seu navegador e acesse <http://localhost:8080/receitas>. Você deve ver os dados do banco em formato JSON. Deixe o back-end rodando.

Passo 4: Execute o Front-end (A Interface Gráfica)

1. No NetBeans, vá em **Arquivo > Abrir Projeto...** e selecione a pasta frontend.
2. Clique com o botão direito no projeto frontend e selecione **Limpar e Construir**. Aguarde o BUILD SUCCESS.
3. Certifique-se de que o back-end ainda está rodando! O front-end não funcionará sem ele.
4. Encontre o arquivo Main.java do projeto

4. Encontre o arquivo `main.java` do projeto frontend.
5. Clique com o botão direito nele e selecione **Executar Arquivo**.