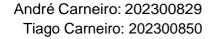
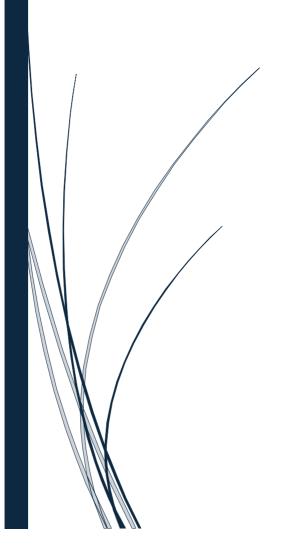


# Recipe Website Manual Técnico

Programação e Integração de Serviços – PIS 2024/2025

Prof. Luís Costa





# 1. Enquadramento

Este projeto é uma aplicação web full-stack para gestão de receitas culinárias, com funcionalidades de autenticação de utilizadores, partilha de receitas, favoritos e integração com a API externa TheMealDB.

## **Funcionalidades Principais:**

- Autenticação de utilizadores (registo, login, logout).
- CRUD (Criar, Ler, Atualizar, Eliminar) de receitas.
- Sistema de favoritos.
- Exploração de receitas externas via API.
- Interface responsiva com React.

# 2. Arquitetura do Sistema

O sistema segue uma arquitetura MVC (Model-View-Controller) e divide-se em:

Camada	Tecnologias		
Frontend	React, React Router, Axios, CSS		
Backend	Node.js, Express, MySQL, Multer (upload de ficheiros)		
Autenticação	JWT (JSON Web Tokens)		
Base de Dados	MySQL		

# 3. Instalação e Configuração

## Pré-requisitos

- Node.js (v18+)
- MySQL (v8+)
- Gestor de Pacotes (npm)

# 4. Passos de Instalação

## Clonar o Repositório

git clone <a href="https://qithub.com/3andredc0/recipe-website.git">https://qithub.com/3andredc0/recipe-website.git</a> cd backend && npm install cd frontend/food-blog-app && npm install

## Configurar a Base de Dados

Criar uma base de dados MySQL chamada recipewebsite Executar o script SQL dbScripts.sql

## Configurar Variáveis de Ambiente em settings.json e .env

```
{
    "server": {
        "port": 3000
},
    "database": {
        "host": "localhost",
        "user": "root",
        "database": "recipewebsite",
        "password": "password",
        "port":3306,
        "connectionLimit": 20
    }
}
SECRET KEY="secretkey"
```

#### Iniciar o servidor

cd backend && npm start # Inicia o servidor Node.js na porta 3000 cd frontend && npm start # Inicia a aplicação React na porta 5173

# 5. Funcionalidades Detalhadas

## 5.1. Gestão de Utilizadores

- Registo: Validação de e-mail único e hash da password com bcrypt.
- Login: Geração de token JWT com duração indefinida (até logout).
- Admin: Utilizadores com admin=true podem editar/eliminar qualquer receita.

#### 5.2. Gestão de Receitas

- Criar Receita: Upload de imagem (armazenada em backend/public/images).
- Editar/Eliminar: Apenas o autor ou administradores têm permissão.
- Formato dos Dados:

```
{
  "title": "Bolo de Chocolate",
  "ingredients": ["farinha", "açúcar", "ovos"],
  "instructions": "Misturar todos os ingredientes...",
  "time": "45 minutos"
}
```

#### 4.3. Favoritos

- Adicionar/Remover: Armazenados na tabela favorites.
- Listagem: Mostra todas as receitas favoritadas pelo utilizador.

## 4.4. Exploração de Receitas Externas

- Integração com TheMealDB:
  - o Pesquisa por nome ou categoria.
  - o Detalhes normalizados para o formato interno da aplicação.

# 6. Documentação da API

Endpoint	Método	Descrição	Autenticação
/signUp	POST	Registo de utilizador	Não
/login	POST	Login de utilizador	Não
/recipe	GET	Listar todas as receitas	Não
/recipe/:id	GET	Detalhes de uma receita	Não
/recipe	POST	Criar uma receita	Sim (JWT)
/recipe/:id	PUT	Atualizar uma receita	Sim (JWT)
/recipe/:id	DELETE	Eliminar uma receita	Sim (JWT)
/favorite/:recipeId	POST	Adicionar receita aos favoritos	Sim (JWT)
/favorite/:recipeId	DELETE	Remover receita dos favoritos	Sim (JWT)

# 7. Segurança

- **JWT**: Tokens incluem id, email e admin para autorização.
- Validação de Inputs: Campos obrigatórios verificados no backend.
- RBAC (Role-Based Access Control): Administradores têm acesso total.