

# Documentação do Projeto: Financial Tracker

## 1. Introdução

O **Financial Tracker** é uma aplicação web que permite aos utilizadores gerir e monitorizar várias contas bancárias e plataformas de negociação a partir de uma única interface. O objetivo principal da aplicação é centralizar informações financeiras dispersas, oferecendo aos utilizadores uma visão consolidada do seu estado financeiro, transações e investimentos.

### 1.1 Objetivos

- **Centralização de Contas:** Os utilizadores podem vincular diversas contas bancárias e plataformas de negociação para visualizar e gerir todas as suas finanças num único local.
- **Análise Financeira:** Oferecer uma análise detalhada das finanças pessoais, incluindo padrões de despesas, rendimentos, e evolução de investimentos.
- **Segurança de Alto Nível:** Implementação de medidas de segurança avançadas para garantir a proteção de dados financeiros sensíveis.

## 2. Funcionalidades Principais

- **Integração com Múltiplas Contas:** Suporte para a ligação a várias contas bancárias e plataformas de negociação de ações, criptomoedas e outros investimentos.
- **Painel de Controlo Personalizado:** Um dashboard dinâmico onde o utilizador pode visualizar o saldo total, últimos movimentos e um resumo de investimentos.
- **Notificações:** Alertas personalizados sobre limites de despesas, eventos importantes nas plataformas de trading, e transações incomuns.
- **Relatórios:** Geração de relatórios financeiros automáticos com resumos mensais e anuais.
- **Segurança Avançada:** Utilização de métodos de encriptação fortes e autenticação multifator para garantir a proteção dos dados.

## 3. Requisitos de Segurança

Devido à natureza sensível dos dados que serão manipulados, a segurança será uma das principais prioridades no desenvolvimento desta aplicação.

### 3.1 Encriptação de Dados

- **Encriptação de Dados em Trânsito:** Toda a comunicação entre os utilizadores e os servidores será protegida por protocolos de encriptação (TLS/SSL).
- **Encriptação de Dados em Repouso:** As informações financeiras armazenadas no sistema, como dados de contas e transações, serão encriptadas utilizando algoritmos de encriptação simétricos e assimétricos.

### 3.2 Autenticação e Autorização

- **Autenticação Multifator (MFA):** O acesso à aplicação exigirá MFA para garantir que apenas utilizadores autorizados possam aceder às informações.
- **Gestão de Sessões:** As sessões de utilizador terão um tempo de validade limitado e serão automaticamente encerradas em caso de inatividade.
- **Autorização Baseada em Funções:** Diferentes níveis de acesso serão definidos para utilizadores (e.g., utilizadores comuns, administradores) para garantir que apenas as informações necessárias sejam acedidas.

### 3.3 Proteção contra Ameaças

- **Deteção de Ameaças em Tempo Real:** Implementação de sistemas de monitorização para identificar tentativas de intrusão e ataques DDoS.
- **Prevenção de Ataques XSS e CSRF:** Aplicação de mecanismos de proteção contra ataques de Cross-Site Scripting (XSS) e Cross-Site Request Forgery (CSRF).

## 4. Arquitetura da Aplicação

A aplicação seguirá uma arquitetura baseada em microserviços, garantindo escalabilidade, flexibilidade, e facilidade de manutenção. Os principais componentes serão:

- **Frontend:** Desenvolvido com React.js e Tailwind CSS para uma experiência de utilizador moderna e responsiva.
- **Backend:** Implementado com Node.js e Express para lidar com a lógica de negócio e comunicações com os serviços externos.
- **Base de Dados:** Uma base de dados segura (como PostgreSQL ou MongoDB) será utilizada para armazenar dados de utilizador e transações de forma eficiente e segura.
- **Integração com APIs:** Serão utilizadas APIs bancárias abertas (como Open Banking) para aceder aos dados das contas dos utilizadores de forma segura e legal.

## 5. Tecnologias Propostas

- **Frontend:** React.js, Tailwind CSS
- **Backend:** Node.js, Express.js
- **Base de Dados:** PostgreSQL ou MongoDB
- **Segurança:** OpenSSL, Autenticação JWT, OAuth 2.0

- **Infraestrutura:** Docker, Kubernetes para orquestração de serviços, AWS ou Azure para hosting e escalabilidade.