# **Documentação Técnica da Aplicação *Web*** **FinTrack**

Daniel Carneiro Rosa  
Diego Soares Pires Ferreira  
João Victor Cavalcanti Teixeira de Assis

([dan.rosa10@gmail.com](mailto:dan.rosa10@gmail.com))  
([diegospf13@gmail.com](mailto:diegospf13@gmail.com))  
([joao.ctassis@gmail.com](mailto:joao.ctassis@gmail.com))

Recife  
**Março/2022**

## Versionamento do documento

|  |  |  |
| --- | --- | --- |
| **Versão** | **Data da modificação** | **Autor(es)** |
| 1.0 | 01/03/2022 | Daniel Carneiro, Diego Soares, João Victor Assis |
|  |  |  |
|  |  |  |
|  |  |  |

## Índice

1. Introdução..............................................................................................................4
2. Visão Geral da aplicação.......................................................................................5
3. Requisitos Funcionais............................................................................................6  
   3.1. Requisitos Mínimos........................................................................................6  
   3.2. Requisitos Não Mínimos................................................................................6
4. Requisitos Não-Funcionais....................................................................................7
5. Detalhes de implementação...................................................................................8  
   5.1. Fluxos de telas................................................................................................8  
   5.2. Diagrama Entidade Relacionamento do Banco de Dados..............................8  
   5.3. Esquema Relacional do Banco de Dados.......................................................8  
   5.4. Protótipos de Baixa Fidelidade.......................................................................9
6. Considerações Finais...........................................................................................11

## 1. Introdução

Este relatório representa a documentação técnica básica do projeto FinTrack, uma aplicação web desenvolvida por alunos da UFRPE (Universidade Federal Rural de Pernambuco) na disciplina de Desenvolvimento de aplicações para WEB. Nele apresentamos uma descrição inicial do projeto, assim como os requisitos e descrições técnicas que julgamos relevantes.

O objetivo desse documento é proporcionar uma visão ampla dos recursos implementados no FintTack e possibilitar o devido entendimento técnico ou não de como funciona o sistema.

### **Repositório GitHub**: <https://github.com/Danielgol/FinTrack>

## **2. Visão Geral da aplicação**

**Nome da aplicação:**

FinTrack - Acompanhamento de finanças

### **Objetivos**

Pretende-se com esse trabalho construir uma aplicação WEB em que o usuário consiga ter uma visualização das suas finanças e seus investimentos em criptomoedas.

### **Descrição**

Simples, confiável e seguro.

FinTrack é uma plataforma que o irá auxiliar com suas finanças pessoais, trazendo uma interface amigável para quem quer acompanhar o seu caixa pessoal independente do banco ou da moeda.

No FinTrack, o usuário pode registrar sempre que tiver um gasto ou um recebimento, colocando descrição e o valor.

Com esse app, o usuário pode:

* Colocar o quanto gastou e quanto recebeu;
* Separar as finanças por banco;
* Acompanhar seus investimentos em criptomoedas;
* Calcular projeção de rendimento do mês atual.

### **Usuários**

A aplicação devido a sua natureza e objetivos vale para qualquer usuário que tenha uma renda ou que queira acompanhar suas finanças.

Temos 3 níveis de usuários a serem identificados seguindo o Onion Model:

Nível 1 (Sistema): Pessoas que querem organizar suas finanças;

Nível 2 (Sistema de Contenção):

Nível 3 (Ambiente): Concorrentes (Wallet, My wallet).

## **3. Requisitos Funcionais**

### 3.1. Requisitos Mínimos

Requisitos básicos que devem estar presentes no sistema.

* **[RF-01]:** Como função básica, o sistema deve permitir aos usuários entrada/criação manual das finanças.
* **[RF-02]:** Essencialmente, o sistema deverá permitir ao usuário que ele coloque gastos e ganhos sob uma certa finança ou grupo de finanças.
* **[RF-03]:** O sistema permite que o usuário crie grupos de finanças para unificar os saldos das finanças já cadastrados em um cálculo único.
* **[RF-04]:** O sistema permite a edição dos grupos de finanças.
* **[RF-05]:** O sistema permite a edição das finanças individualmente.
* **[RF-06]:** O sistema permite ao usuário visualizar suas finanças em moedas diferentes (físicas ou digitais).
* **[RF-07]:** As funções que permitem a alteração de dados no sistema deverão apenas estar acessíveis/visíveis quando for inserida um login e uma senha válidos.
* **[RF-08]:** A interface básica deve incluir opções para acesso ao sistema (para utilizadores autorizados) e registo de novos utilizadores.
* **[RF-09]:** O sistema só poderá cadastrar CPFs válidos e não repetidos.
* **[RF-10]:** A interface da aplicação deve ser acessível a partir de um navegador de Internet.

### 3.2. Requisitos Não Mínimos

Requisitos que queremos implementar, mas não estão confirmados.

* **[RF-11]:** O sistema permite que o usuário registre um empréstimo feito ou recebido.
* **[RF-12]:** O sistema permite que o usuário visualize suas finanças com o valor possuído e com o valor projetado a partir de um empréstimo realizado.

## **4. Requisitos Não Funcionais**

As categorias embarcadas no nosso projeto são: *Segurança, Confiabilidade, Interoperabilidade, Usabilidade, Integração, Portabilidade, Implementação.*

### 4.1. Segurança

* **[RNF-01]:** Utilização de tokens para autenticação dos usuários.
* **[RNF-02]:** Os dados de um usuário não podem ser visíveis por outros usuários.

### 4.2. **Confiabilidade**

* **[RNF-03]:** O sistema deverá ter disponibilidade em no mínimo 98% do tempo.

### 4.3. **Interoperabilidade**

* **[RNF-04]:** O sistema deverá se comunicar com o MySQL Server para buscar os dados de cada usuário.

### 4.4. **Usabilidade**

* **[RNF-05]:** O sistema deve ser facilmente utilizável a partir da interface limpa dele.

### 4.5. **Integração**

* **[RNF-06]:** O sistema irá se integrar com APIs externas para conseguir os dados de conversão de moedas.

### 4.6. **Portabilidade**

* **[RNF-07]:** O sistema deverá rodar em qualquer plataforma através de um navegador de internet.

### 4.7. **Implementação**

* **[RNF-08]:** Para a persistência dos dados, o sistema irá utilizar o MySQL.
* **[RNF-09]:** Para o desenvolvimento da aplicação no Back-end, o sistema irá utilizar o NodeJS.
* **[RNF-10]:** Para o desenvolvimento da aplicação no Front-end, o sistema irá utilizar o AngularJS.
* **[RNF-11]:** A Integração das partes será feita seguindo o padrão MVC. O servidor Back-end será o módulo intermediário entre o Banco de Dados e as páginas Web.

## **5. Detalhes de implementação**

Nessa sessão estão dispostas as especificações das telas do sistema e do Banco de Dados.

### **5.1. Fluxos de telas**

Interface gráfica do usuário, Diagrama

Descrição gerada automaticamente

### **5.2. Diagrama Entidade Relacionamento do Banco de Dados**

### **5.3. Esquema Relacional do Banco de Dados**

### **5.4.** **Protótipos de Baixa Fidelidade**

#### 5.4.1. Tela inicial

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

#### 5.4.2. Tela de Login

Interface gráfica do usuário, Aplicativo, PowerPoint

Descrição gerada automaticamente

#### 5.4.3. Tela de Cadastro

Interface gráfica do usuário, Aplicativo, PowerPoint

Descrição gerada automaticamente

## **6. Considerações finais**