

## Desafio – Painel de Investimentos com Perfil de Risco Dinâmico

### **Objetivo**

Criar uma aplicação web que analisa o comportamento financeiro do cliente e ajusta automaticamente seu perfil de risco, sugerindo produtos de investimento como CDBs, LCIs, LCAs, Tesouro Direto, Fundos, etc.

### **Front-end – Angular 19**

A CAIXA está evoluindo sua plataforma de investimentos digital. O backend já está pronto e oferece endpoints para retornar perfil de risco com base em dados financeiros, produtos de investimento mais adequados ao perfil e histórico de investimentos e simular investimentos com entrada de valor, prazo e tipo.

- Agora, o desafio é criar uma experiência completa que permita:
  - Visualizar um Dashboard interativo ???
  - Simular investimentos com base nos dados de entrada
  - Exibir os resultados da simulação de forma clara, incluindo visualização de retorno esperado mês a mês.

Imagina que Rafael, um investidor, quer investir em Tesouro Direto. Ele acessa a aplicação, simula a contratação do investimento e entende quanto seu dinheiro renderá mês a mês. Sua aplicação será a ponte entre a necessidade e a decisão do investimento.

### **Objetivo do Desafio**

Crie um aplicativo Angular 19 que permita:

1. **Dashboard Interativo**
  - Gráficos de evolução dos investimentos
  - Distribuição por tipo de produto
  - Histórico de perfil de risco
2. **Visualização do Perfil de Risco**
  - Conservador, Moderado, Agressivo
  - Explicação textual e visual do perfil
3. **Lista de Produtos Recomendados**
  - Nome, tipo, rentabilidade, risco
  - Botão para simular investimento
4. **Simulador de Investimento**
  - Inputs: valor, prazo, tipo

- **Output:** retorno estimado

## 5. Responsividade e Acessibilidade

- Layout adaptável para mobile
- Suporte a leitores de tela

## Requisitos Técnicos

- **Framework:** Angular 19
- **Estilo:** Angular Material ou TailwindCSS
- **Testes:** Karma ou Jest
- **Cobertura mínima de 80%**
- **Integração com API:** HttpClient ou Fetch ou Axios
- **Gráficos:** ngx-charts ou Chart.js
- **Reatividade:** RxJS

## Endpoints Esperados do backend

### 1. GET /perfil-risco/{clientelid}

- **Descrição:** Retorna o perfil de risco atual do cliente.
- **Resposta esperada:**

```
{  
  "clientelid": 123,  
  "perfil": "Moderado",  
  "descricao": "Perfil equilibrado entre segurança e rentabilidade.",  
  "pontuacao": 65  
}
```

### 2. GET /investimentos/{clientelid}

- **Descrição:** Retorna o histórico de investimentos do cliente.
- **Resposta esperada:**

```
[  
 {  
   "id": 1,  
   "tipo": "Ações",  
   "data": "2023-01-01",  
   "valor": 1000000  
 }]
```

```
        "tipo": "CDB",
        "valor": 5000,
        "rentabilidade": 0.12,
        "data": "2025-01-15"
    },
    {
        "id": 2,
        "tipo": "Fundo Multimercado",
        "valor": 3000,
        "rentabilidade": 0.08,
        "data": "2025-03-10"
    }
]
```

### 3. GET /produtos-recomendados/{perfil}

- **Descrição:** Retorna uma lista de produtos de investimento recomendados com base no perfil de risco.
- **Resposta esperada:**

```
[{
    {
        "id": 101,
        "nome": "CDB Caixa 2026",
        "tipo": "CDB",
        "rentabilidade": 0.13,
        "risco": "Baixo"
    },
    {
        "id": 102,
        "nome": "Fundo Agressivo XPTO",
        "tipo": "Fundo",
        "rentabilidade": 0.18,
    }
}]
```

```
"risco": "Alto"
```

```
}
```

```
]
```

#### 4. POST /simular-investimento

- **Descrição:** Simula o retorno de um investimento com base em valor, prazo e tipo.
- **Corpo da requisição:**

```
{
```

```
    "valor": 10000,
```

```
    "prazoMeses": 12,
```

```
    "tipo": "CDB"
```

```
}
```

- Resposta esperada:

```
{
```

```
    "valorFinal": 11200,
```

```
    "rentabilidade": 0.12,
```

```
    "detalhes": "Simulação baseada em CDB com taxa de 12% ao ano."
```

```
}
```

#### 5. POST /autenticacao/login

- **Descrição:** Autentica o cliente e retorna o token JWT.

- **Corpo da requisição:**

```
{
```

```
    "email": "cliente@exemplo.com",
```

```
    "senha": "123456"
```

```
}
```

- **Resposta esperada:**

```
{
```

```
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
```

```
    "clientId": 123
```

```
}
```

## **Exemplo de Estrutura de Módulos**

```
src/app/  
  └── dashboard/  
    ├── dashboard.component.ts  
    ├── investment-chart.component.ts  
    ├── risk-profile.component.ts  
    ├── product-list.component.ts  
    └── investment-simulator.component.ts  
  └── services/  
    ├── investment.service.ts  
    └── profile.service.ts  
  └── models/  
    ├── investment.model.ts  
    ├── product.model.ts  
    └── profile.model.ts
```

### **Critérios de Avaliação**

- Funcionalidade completa e fluida
- Código limpo, modular e reutilizável
- Testes com cobertura mínima de 80%
- Clareza na exibição dos dados
- Qualidade dos gráficos e visualizações
- Boas práticas de componentes, hooks e integração com API
- Responsividade e acessibilidade
- Design de acordo com a marcas.caixa