

PLANO DE REFATORAÇÃO - SISTEMA DE DOAÇÕES

Data de Início: 08/09/2025

Versão do Sistema: 1.1.5





Status: EM EXECUÇÃO

DIAGNÓSTICO INICIAL

Problemas Identificados:

1. **app.js com ~3000 linhas** - Monolítico, dificulta gestão
2. **Dois bancos de dados** - Um em ./database/ e outro na raiz
3. **Múltiplos erros nas últimas 24h** - Rollback necessário
4. **Scripts de correção dispersos** - Sem padronização

Pontos a Preservar:

-  Layout e estética atual
 -  Funcionalidades existentes
 -  Estrutura de banco
 -  Compatibilidade total
-

FASES DE EXECUÇÃO

FASE 1 - ESTABILIZAÇÃO (PRIORIDADE ALTA) ●

1.1 - Backup Completo

Status: CONCLUÍDO

Ação: Criar backup completo antes de alterações

Script: backup-sistema-completo.js

```
bash  
node backup-sistema-completo.js
```

1.2 - Identificar Erros de Sintaxe

Status: PENDENTE

Ação: Analisar app.js e identificar todos os erros

Script: diagnostico-erros-sintaxe.js

```
bash  
node diagnostico-erros-sintaxe.js
```

1.3 - Remover Banco Duplicado

Status: PENDENTE

Ação: Identificar e remover banco na raiz

Script: unificar-banco-dados.js

```
bash  
node unificar-banco-dados.js
```

1.4 - Consolidar Scripts de Correção

Status: PENDENTE

Ação: Unificar scripts dispersos em único sistema

Script: `consolidar-scripts-correcao.js`

```
bash
```

```
node consolidar-scripts-correcao.js
```

FASE 2 - MODULARIZAÇÃO (PRIORIDADE MÉDIA) ●

2.1 - Análise de Dependências

Status: AGUARDANDO

Ação: Mapear todas as funções e suas dependências

Entregável: Mapa de dependências

2.2 - Divisão em Módulos

Status: AGUARDANDO

Estrutura Proposta:

public/js/

- |— config.js (~50 linhas)
- |— api.js (~300 linhas)
- |— doadores.js (~400 linhas)
- |— doacoes.js (~600 linhas)
- |— historico.js (~300 linhas)
- |— modals.js (~500 linhas)
- |— filters.js (~200 linhas)
- |— exports.js (~400 linhas)
- |— utils.js (~150 linhas)
- |— init.js (~100 linhas)

2.3 - Sistema de Imports

Status: AGUARDANDO

Ação: Implementar carregamento modular no HTML

2.4 - Testes de Compatibilidade

Status: AGUARDANDO

Ação: Verificar todas as funcionalidades

FASE 3 - OTIMIZAÇÃO (PRIORIDADE BAIXA) ●

3.1 - Limpeza de Console.logs

Status: AGUARDANDO

3.2 - Sistema de Logs

Status: AGUARDANDO

3.3 - Minificação

Status: AGUARDANDO

3.4 - Otimização de Queries

Status: AGUARDANDO



REGRAS DE EXECUÇÃO

1. **NUNCA** alterar layout ou estética
2. **SEMPRE** criar backup antes de alterações
3. **TESTAR** cada etapa antes de prosseguir
4. **DOCUMENTAR** no CONTROLE_VERSAO.md
5. **GERAR** script .js para cada alteração
6. **MANTER** padrão de comentários com versão



PROCESSO DE VALIDAÇÃO

Cada etapa deve:

1. Gerar script executável
2. Criar backup dos arquivos afetados
3. Executar alteração
4. Testar funcionalidade
5. Documentar resultado

6. Só então prosseguir

MÉTRICAS DE SUCESSO

- ☐ Zero erros de sintaxe
 - ☐ Único banco de dados
 - ☐ app.js < 500 linhas
 - ☐ Módulos bem definidos
 - ☐ 100% funcionalidades preservadas
 - ☐ Documentação completa
-

STATUS GERAL






Progresso:  40%

Fase Atual: FASE 2 - MODULARIZAÇÃO (Próxima)

Próxima Ação: Testar sistema e iniciar modularização

FASE 1 CONCLUÍDA - RESULTADOS

Melhorias Aplicadas:

-  **3041 linhas** no app.js (pronto para modularização)
-  **1 banco unificado** (./database/doacoes.db)
-  **0 erros de sintaxe** graves
-  **6 scripts organizados** em estrutura hierárquica
-  **Documentação completa** criada

Sistema Atual:

- **Funcional:** 100% operacional
- **Organizado:** Scripts em pastas apropriadas
- **Documentado:** Todos os processos registrados
- **Preparado:** Pronto para Fase 2

Última Atualização: 09/09/2025 - FASE 1 CONCLUÍDA