

SGI FV - Módulo de Processos

Visão Geral

Este documento descreve a implementação do módulo de Processos com arquitetura SaaS multi-tenant e Row Level Security (RLS).

Arquitetura

Contexto de Autenticação

O sistema utiliza um `AuthContext` que gerencia:

- Sessão do Supabase
- Contexto do usuário (`org_id`, `org_role`, `org_name`, etc.)
- Verificação de permissões (`isAdmin`)

```
// Uso do contexto
const { userContext, isAdmin, signOut } = useAuth();
```

Fluxo de Dados

1. Usuário faz login → Supabase Auth
2. `AuthContext` carrega contexto via `v_user_context` view
3. Componentes acessam `userContext.org_id` para filtrar dados
4. RLS garante isolamento de dados entre organizações

Migrations SQL

Execute as migrations **na ordem** no Supabase SQL Editor:

1. Migration 003: `v_user_context`

```
-- Localização: supabase/migrations/003_v_user_context.sql
-- Cria view para contexto do usuário
```

2. Migration 004: `processes`

```
-- Localização: supabase/migrations/004_processes.sql
-- Cria tabelas: processes, process_events
-- Cria triggers: updated_at, generate_protocol
```

3. Migration 005: `rls_processes`

```
-- Localização: supabase/migrations/005_rls_processes.sql
-- Habilita RLS e cria policies
```

Ordem de Execução das Migrations

1. Abra o Supabase Dashboard → SQL Editor
2. Execute `003_v_user_context.sql`
3. Execute `004_processes.sql`
4. Execute `005_rls_processes.sql`

Importante: As migrations 001 e 002 (multi-empresa e RLS base) já devem estar aplicadas.

Estrutura de Tabelas

processes

Coluna	Tipo	Descrição
id	uuid	PK, auto-gerado
org_id	uuid	FK → organizations
titulo	text	Título do processo
protocolo	text	Auto-gerado (SGI-YYYY-NNN)
status	text	cadastro/triagem/analise/concluido
cliente_nome	text	Nome do cliente
cliente_documento	text	CPF/CNPJ
cliente_contato	text	Telefone/email
responsavel_user_id	uuid	FK → auth.users
created_at	timestamptz	Data de criação
updated_at	timestamptz	Última atualização

process_events

Coluna	Tipo	Descrição
id	uuid	PK, auto-gerado
org_id	uuid	FK → organizations
process_id	uuid	FK → processes
tipo	text	registro/status_change/observacao/documento/atribuicao
mensagem	text	Descrição do evento
created_by	uuid	FK → auth.users
created_at	timestamptz	Data do evento

Políticas RLS

processes

- **SELECT:** Membros da organização (`is_org_member(org_id)`)
- **INSERT/UPDATE/DELETE:** Apenas admin/owner (`is_org_admin(org_id)`)

process_events

- **SELECT:** Membros da organização
- **INSERT/UPDATE:** Apenas admin/owner

Checklist de Testes Manuais

Login e Navegação

- [] Login com credenciais válidas → Entra no AppLayout
- [] Sidebar mostra nome da organização e role do usuário
- [] Topbar mostra nome/email do usuário
- [] Logout funciona e redireciona para /login

Processos (como admin/owner)

- [] /processos mostra lista vazia inicialmente
- [] Botão “Novo Processo” abre modal
- [] Criar processo → Aparece na lista com protocolo auto-gerado
- [] Clicar em processo → Abre detalhes com timeline
- [] Alterar status → Cria evento na timeline
- [] Adicionar observação → Cria evento na timeline

Processos (como client/staff)

- [] Pode visualizar processos da organização
- [] NÃO vê botão “Novo Processo”

- [] NÃO vê botões de alterar status
- [] NÃO vê botão “Registrar Observação”

RLS (multi-tenant)

- [] Usuário da Org A NÃO vê processos da Org B
- [] Mesmo com ID direto, processo de outra org retorna erro

Arquivos Criados/Modificados

Criados

- supabase/migrations/003_v_user_context.sql
- supabase/migrations/004_processes.sql
- supabase/migrations/005_rls_processes.sql
- src/contexts/AuthContext.tsx
- src/lib/processes.ts
- src/lib/userContext.ts
- src/pages/Configuracoes.tsx
- PROCESSES_MODULE.md

Modificados

- App.tsx - Integração com AuthProvider
- src/layouts/AppLayout.tsx - Removido props de autenticação
- src/components/Sidebar.tsx - Exibe contexto da organização
- src/components/Topbar.tsx - Exibe dados do usuário
- src/pages/Dashboard.tsx - Stats reais do banco
- src/pages/Processes/ProcessList.tsx - Dados reais + modal de criação
- src/pages/Processes/ProcessDetails.tsx - Dados reais + ações
- src/pages/Processes/ProcessNew.tsx - Redireciona para lista
- src/pages/Clients/ClientList.tsx - Placeholder atualizado
- pages/Login.tsx - Simplificado para usar AuthContext
- pages/Register.tsx - Simplificado para usar AuthContext

Matriz de Permissões

Ação	Owner	Admin	Staff	Client
Ver processos	✓	✓	✓	✓
Criar processo	✓	✓	✗	✗
Alterar status	✓	✓	✗	✗
Adicionar observação	✓	✓	✗	✗
Ver membros	✓	✓	✓	✗
Gerenciar membros	✓	✓	✗	✗

Comandos

```
# Desenvolvimento
npm run dev
```

```
# Build de produção
npm run build
```

```
# Preview do build
npm run preview
```

Troubleshooting

“Processo não encontrado”

- Verifique se o org_id do usuário está correto
- Verifique se as RLS policies estão habilitadas

“Erro ao criar processo”

- Verifique se o usuário é admin/owner
- Verifique se `is_org_admin()` function existe

“Usuário sem organização”

- Execute registro novamente com org padrão
- Verifique se membership foi criada corretamente