

Sistema de Gerenciamento de Rotinas, Ações e Empresas

1. Visão Geral do Sistema

O sistema tem como objetivo gerenciar **rotinas, ações, delegações e usuários** distribuídos em uma ou várias empresas. Ele fornece controle total sobre:

- Cadastro e manutenção de empresas
- Cadastro e gerenciamento de usuários
- Criação, edição, delegação e execução de rotinas
- Controle de ações dentro das rotinas
- Sistema de tags, projetos e auditoria via logs
- Segurança baseada em níveis de acesso (MASTER, ADMINISTRADOR, FUNCIONÁRIO)

Toda a base segue os requisitos funcionais (RF) e não funcionais (RNF) definidos.

2. Modelo Conceitual (ER)

Entidades Principais

Usuário

Representa qualquer indivíduo que acessa o sistema.

- id, nome, CPF, email, senhaHash
- nível de acesso: MASTER, ADMIN, FUNCIONÁRIO

- status: ATIVO, INATIVO, BLOQUEADO
- empresa à qual pertence
- segredo 2FA

Relacionamentos:

- Pertence a uma empresa
- Pode criar projetos, rotinas, ações e tags
- Pode receber rotinas delegadas

Empresa

Estrutura hierárquica principal.

- id, nome, email, dataCriacao

Relacionamento:

- 1 empresa → N usuários

Projeto

Agrupar rotinas relacionadas.

- 1 projeto → N rotinas

Rotina

Elemento central do sistema.

- nome, datas, prioridade, recorrente, status

Relacionamentos:

- 1 rotina \rightarrow N ações
- 1 rotina \rightarrow N delegações
- 1 rotina \leftrightarrow N tags (RotinaTag)
- Pertence a um projeto (opcional)

Ação

Uma rotina é composta por ações menores.

- status, descrição, datas

Tag & RotinaTag

Sistema de categorização.

- Tag possui id, nome, cor
- RotinaTag faz associação $N \times N$

DelegacaoRotina

Representa uma atribuição de rotina.

- idRotina, idFuncionario, status, progresso

3. Diagrama de Classes — Responsabilidades

Usuário

- autenticar()
- redefinirSenha()

- alterarStatus()

Projeto

- adicionarRotina()
- removerRotina()

Rotina

- iniciar()
- concluir()
- atualizarStatus()

Ação

- executar()
- atualizar()
- marcarConcluida()

DelegacaoRotina

- atualizarProgresso()

Tag

- aplicarEmRotina()
- removerDeRotina()

LogSistema

- registrar()

4. Diagramas de Caso de Uso

Gerenciamento de Empresas

- Atores: MASTER (administração completa), ADMIN (consulta)
- Casos: Cadastrar, Pesquisar, Manter Empresa

Gerenciamento de Usuários

- Atores: MASTER e ADMIN
- Casos: Cadastrar, Pesquisar, Manter Usuário

Login

- Atores: Todos os níveis
- Casos: Login, Logout, Trocar senha

Execução de Rotinas

- Ator: Funcionário
- Casos: Visualizar, Abrir Rotina, Concluir Ação, Fechar Rotina

5. Diagramas de Sequência — Fluxo Interno

Exemplo: Criar Rotina

1. Usuário solicita criação
2. Backend valida permissões

3. Banco registra rotina
4. Sistema retorna confirmação

Inclui validações, logs, ações e delegações.

6. Diagramas de Atividade

- Login com 2FA
- Gerenciamento de Empresas
- Gerenciamento de Usuários
- Criação/edição de Rotinas
- Execução de Rotinas pelos Funcionários

7. Mapeamento RF / RNF

Exemplos:

- **RF007 – Login:** Representado em Casos de Uso, Atividades e Sequência
- **RF010 – Criar Rotina:** Representado em ER, Classes e Fluxos
- **RNF011 – Auditoria:** LogSistema
- **RNF013 – Sincronização Offline:** timestamps, delegações e resolução de conflitos

8. Regras Técnicas e Boas Práticas

- Hash seguro para senhas: bcrypt ou argon2

- 2FA por usuário
- Logs consistentes
- Índices para acelerar consultas
- Endpoints REST padronizados
- Suporte a sincronização offline

9. Estrutura da Base de Dados (MySQL)

9.1. usuario

- idUsuario (PK)
- idEmpresa (FK)
- nome, cpf, email
- senhaHash
- nivel
- status
- segredo2FA

9.2. empresa

- idEmpresa (PK)
- nome
- dataHoraCriacao

9.3. projeto

- idProjeto (PK)
- idCriador (FK)
- nome

9.4. rotina

- idRotina (PK)
- idProjeto
- idCriador
- nome
- datas e status

9.5. acao

- idAcao (PK)
- idRotina (FK)
- nome, status, definicaoConclusao

9.6. delegacaorotina

- idDelegacao (PK)
- idRotina
- idFuncionario
- statusProgresso
- progressoPercentual

