Trabalho de Conclusão do Curso Tecnólogo em Sistemas para Internet

Carlos Augusto Esgaravato 13892

Lucas Ferro Ribeiro 13893

Mateus Batista Martins 14197

Docente: Adnys Lougan da Silva

2025

Faculdade ALFA Umuarama - Unialfa / Paraná

**Sumário**

[**1. INTRODUÇÃO 6**](#)

[**2. PROJETO DO SISTEMA 7**](#_ayduijlrjed7)

[2.1 User Story – Tela de Login acesso ao sistema 7](#_8feqd5t0gi30)

[2.1.1 Critérios de Aceitação 7](#_vkwftj62oqdo)

[2.1.2 Regras de Negócio 7](#_7drhkqpgfb0n)

[2.1.3 Teste de Aceitação - Caso de Teste Positivo 7](#_3ru4x642o6ye)

[2.1.4 Teste de Aceitação - Casos de Teste Negativos 8](#_tiskhl1purl5)

[2.1.5 Requisitos Não Funcionais 8](#_dm125d8gt3cv)

[2.2 User Story - Redefinição de Senha 9](#_4bd8o5sl767j)

[2.2.1 Critérios de Aceitação 9](#_vcgqho957757)

[2.2.2 Regras de Negócio 9](#_n8fvv1xj47qn)

[2.2.3 Teste de Aceitação - Caso de Teste Positivo 9](#_3ru4x642o6ye)

[2.2.4 Testes de Aceitação - Casos de Teste Negativo 10](#_wlyn21j80g10)

[2.2.5 Requisitos Não Funcionais 10](#_2x0swje4wfg6)

[2.3 User Story – Cadastro de Alunos 11](#_1yumb3a2u04f)

[2.3.1 Critérios de Aceitação 12](#_1fsv3cj8nvle)

[2.3.2 Regras de Negócio 12](#_aitxzlo62e3x)

[2.3.3 Teste de Aceitação - Caso de Teste Positivo 12](#_u3v9w5pgx4t)

[2.3.4 Testes de Aceitação - Casos de Teste Negativo 12](#_m3qd921310r1)

[2.3.5 Requisitos Não Funcionais 12](#_ne4dsjr4lov0)

[2.4 User Story – Listagem de Alunos 13](#_oqxpuqmjx8qm)

[2.4.1 Critérios de Aceitação 13](#_axmd9pjyy9xw)

[2.4.2 Regras de Negócio 13](#_1tyryhw1dgsb)

[2.4.3 Teste de Aceitação - Caso de Teste Positivo 13](#_f7qalrrv8s2x)

[2.4.4 Testes de Aceitação - Casos de Teste Negativo 14](#_xxf3y1aip841)

[- CT01: Listagem sem alunos cadastrados 14](#_prq4t8f8vgps)

[2.4.5 Requisitos Não Funcionais 14](#_lqfe1opyidpk)

[2.5 User Story – Edição de Alunos 14](#_g7geuzy0o60j)

[2.5.1 Critérios de Aceitação 15](#_lfvxras0w0dv)

[2.5.2 Regras de Negócio 15](#_wcnipyirv0w6)

[2.5.3 Teste de Aceitação - Caso de Teste Positivo 15](#_hk3ha9sprmoo)

[2.5.4 Teste de Aceitação - Caso de Teste Negativo 15](#_qzm8vu7nz072)

[2.5.5 Requisitos Não Funcionais 15](#_ugkuenfhitwe)

[2.6 User Story – Exclusão de Alunos 16](#_5rzaeaigwj9x)

[2.6.1 Critérios de Aceitação 16](#_u3t5upckj13b)

[2.6.2 Regras de Negócio 16](#_e7aibkqs6jpf)

[2.6.3 Teste de Aceitação - Caso de Teste Positivo 17](#_p9lpcdkuqzm4)

[2.6.4 Teste de Aceitação - Caso de Teste Negativo 17](#_v2rqy3c1ai6p)

[2.6.5 Requisitos Não Funcionais 17](#_v6a7beojnlz)

[2.7 User Story – Cadastro de Professores 18](#_n3azizyt9qmg)

[2.7.1 Critérios de Aceitação 18](#_7p50igwqcf8k)

[2.7.2 Regras de Negócio 18](#_6jtf74xislf2)

[2.7.3 Teste de Aceitação - Caso de Teste Positivo 18](#_r8l6a1c6azv5)

[2.7.4 Teste de Aceitação - Caso de Teste Negativo 19](#_g6demu7c7q6)

[2.7.5 Requisitos Não Funcionais 19](#_fqhd6w3t8d97)

[2.8 User Story – Listar Professores 20](#_bolw8o28ql8y)

[2.8.1 Critérios de Aceitação 20](#_e45efvlqa26)

[2.8.2 Regras de Negócio 20](#_74ju5spe5wng)

[2.8.3 Teste de Aceitação - Caso de Teste Positivo 20](#_czo5e9qffz9c)

[2.8.4 Teste de Aceitação - Caso de Teste Negativo 20](#_npt1fuz2a8dg)

[2.8.5 Requisitos Não Funcionais 21](#_t3rrpw7gmlsq)

[2.9 User Story – Edição de Professores 21](#_gthvf1r0ilt)

[2.9.1 Critérios de Aceitação 22](#_4ypptoxnmof6)

[2.9.2 Regras de Negócio 22](#_lpdaenxsdfd0)

[2.9.3 Teste de Aceitação - Caso de Teste Positivo 22](#_84rev58a4v17)

[2.9.4 Teste de Aceitação - Caso de Teste Negativo 22](#_dec44fgzeu0m)

[2.9.5 Requisitos Não Funcionais 23](#_lkfy8fzcooqe)

[2.10 User Story – Exclusão de Professores 23](#_g5fch1kb4fq3)

[2.10.1 Critérios de Aceitação 23](#_f0amz62ovq3a)

[2.10.2 Regras de Negócio 24](#_w3k662dhhmzk)

[2.10.3 Teste de Aceitação - Caso de Teste Positivo 24](#_su9p0r3pu4xt)

[2.10.4 Teste de Aceitação - Caso de Teste Negativo 24](#_h92vitoj4h2)

[2.10.5 Requisitos Não Funcionais 24](#_pn5zhny3l0b4)

[2.11 User Story – Cadastro de Turmas 25](#_gy4dpf567ymg)

[2.11.1 Critérios de Aceitação 25](#_bh1oesjg95bc)

[2.11.2 Regras de Negócio 25](#_gimnmo3vndei)

[2.11.3 Teste de Aceitação - Caso de Teste Positivo 26](#_60qdgmwclgr6)

[2.11.4 Teste de Aceitação - Caso de Teste Negativo 26](#_r54f4320clln)

[2.11.5 Requisitos Não Funcionais 27](#_f5r4yq6rbvqb)

[2.12 User Story – Listar Turmas 28](#_2ekdt5ivpben)

[2.12.1 Critérios de Aceitação 28](#_v2pjmco6bb6z)

[2.12.2 Regras de Negócio 28](#_cgvhs9i6it4v)

[2.12.3 Teste de Aceitação - Caso de Teste Positivo 28](#_w6v6s6d4ed1w)

[2.12.4 Teste de Aceitação - Caso de Teste Negativo 29](#_khp4ukghw7wd)

[2.3.4 Requisitos Não Funcionais 29](#_xbi0pxflfdzc)

[2.13 User Story – Edição de Turmas 30](#_j6gvox79v67p)

[2.13.1 Critérios de Aceitação 30](#_28cr54c1zeio)

[2.13.2 Regras de Negócio 30](#_8bq8jh3mvdu9)

[2.13.3 Teste de Aceitação - Caso de Teste Positivo 30](#_n0jrj8sn7cwi)

[2.13.4 Teste de Aceitação - Caso de Teste Negativo 31](#_uxdypcedsviu)

[2.13.5 Requisitos Não Funcionais 32](#_ft9yrwrpfk9y)

[2.14 User Story – Exclusão de Turmas 32](#_x1pvdm37ahpb)

[2.14.1 Critérios de Aceitação 33](#_f05bl4xntqwx)

[2.14.2 Regras de Negócio 33](#_hrvspnlh8gsa)

[2.14.3 Teste de Aceitação - Caso de Teste Positivo 33](#_ekfetwisqek1)

[2.14.4 Teste de Aceitação - Caso de Teste Negativo 33](#_o0auq4b26g9s)

[2.14.5 Requisitos Não Funcionais 34](#_31t1oivdkeli)

[2.15 User Story – Geração de Relatórios em PDF 35](#_x1qpf51fcxpm)

[2.15.1 Critérios de Aceitação 35](#_6me8j8cu0qii)

[2.15.2 Regras de Negócio 35](#_maqpp2moo9uw)

[2.15.3 Teste de Aceitação - Caso de Teste Positivo 36](#_mgzx6k3apnac)

[2.15.4 Teste de Aceitação - Caso de Teste Negativo 36](#_hhlzec1m8ch1)

[2.15.5 Requisitos Não Funcionais 36](#_dvp6tyunhy9e)

[2.16 User Story – Relatórios de Alunos por Turma 37](#_ezij5movd7zn)

[2.16.1 Critérios de Aceitação 37](#_761t6g12qzt)

[2.16.2 Regras de Negócio 37](#_wqmbp8udm12c)

[2.16.3 Teste de Aceitação - Caso de Teste Positivo 38](#_e6p0nvtymims)

[2.16.4 Teste de Aceitação - Caso de Teste Negativo 38](#_ofwxupwuaah6)

[2.16.5 Requisitos Não Funcionais 39](#_yvh48pgo822v)

[2.17 User Story – Listagem de Professores 39](#_112oixxinh6t)

[2.17.1 Critérios de Aceitação 40](#_mvyzgdg5z17c)

[2.17.2 Regras de Negócio 40](#_utmgekru1ypm)

[2.17.3 Teste de Aceitação - Caso de Teste Positivo 40](#_2i1ed59lnmdx)

[2.17.4 Teste de Aceitação - Caso de Teste Negativo 40](#_8lzvko17hi4r)

[2.17.5 Requisitos Não Funcionais 41](#_xv55jmde7m7v)

[2.18 User Story – Listagem de Turmas 42](#_dgt1y0poyc25)

[2.18.1 Critérios de Aceitação 42](#_xptcasrstvgc)

[2.18.2 Regras de Negócio 43](#_hklnxoocmwsi)

[2.18.3 Teste de Aceitação - Caso de Teste Positivo 43](#_c0ueyjf32pg1)

[2.18.4 Requisitos Não Funcionais 43](#_c2ob3cl8igr0)

[2.19 User Story – Listagem de Projetos 44](#_j0rl94ze2qzl)

[2.19.1 Critérios de Aceitação 44](#_bhal5790jk6l)

[2.19.2 Regras de Negócio 44](#_al67unjn6xbg)

[2.19.3 Teste de Aceitação - Caso de Teste Positivo 44](#_19ku4id1i6k0)

[2.19.4 Teste de Aceitação - Caso de Teste Negativo 45](#_n0868a3kl7ol)

[2.19.5 Requisitos Não Funcionais 45](#_hwlmzso66m8t)

[2.20 User Story – Visualização do Calendário de Eventos 46](#_urm7t7ucrqds)

[2.20.1 Critérios de Aceitação 46](#_lnjm6bmvw9hh)

[2.20.2 Regras de Negócio 46](#_u2dhz2tm72na)

[2.20.3 Teste de Aceitação - Caso de Teste Positivo 46](#_qxr86v1oq2n)

[2.20.4 Teste de Aceitação - Caso de Teste Negativo 47](#_aodam768hy8y)

[2.20.5 Requisitos Não Funcionais 47](#_njrny9963zvx)

[2.21 User Story – Cadastro de Eventos 48](#_ia2ks46bdh6p)

[2.21.1 Critérios de Aceitação 48](#_pbis1pmhlgux)

[2.21.2 Regras de Negócio 49](#_22fy94qtafsh)

[2.21.3 Teste de Aceitação - Caso de Teste Positivo 49](#_21vvhsopfaj)

[2.21.4 Teste de Aceitação - Caso de Teste Negativo 49](#_ncgdvgi630ay)

[2.21.5 Requisitos Não Funcionais 50](#_lwev6g3iz334)

[2.22 User Story – Edição de Eventos 51](#_vwjt8h6lfbfk)

[2.22.1 Critérios de Aceitação 51](#_o4v6emh38baw)

[2.22.2 Regras de Negócio 51](#_wl1jvuiyc0er)

[2.22.3 Teste de Aceitação - Caso de Teste Positivo 52](#_ndkvig3sz2sl)

[2.22.4 Teste de Aceitação - Caso de Teste Negativo 52](#_ugr10ha3g4q3)

[2.22.5 Requisitos Não Funcionais 52](#_u9jvtj6r9ggo)

[2.23 User Story – Exclusão de Eventos 53](#_olh26y0gp2d)

[2.23.1 Critérios de Aceitação 53](#_52x0u2tv975t)

[2.23.2 Regras de Negócio 53](#_5gj01hudsz7t)

[2.23.3 Teste de Aceitação - Caso de Teste Positivo 53](#_yptm3qbkid9j)

[2.23.4 Teste de Aceitação - Caso de Teste Negativo 54](#_kfy7pk6k6krj)

[2.23.5 Requisitos Não Funcionais 54](#_w2ykuyiob9rz)

[2.24 User Story – Envio de Notificações 55](#_z59zwhfqyz5j)

[2.24.1 Critérios de Aceitação 55](#_8ev3025somi0)

[2.24.2 Regras de Negócio 56](#_xpx379raj6tr)

[2.24.3 Teste de Aceitação - Caso de Teste Positivo 56](#_v642gd8p3tw6)

[2.24.4 Teste de Aceitação - Caso de Teste Negativo 57](#_uimm9phyewks)

[2.25 User Story – Botão Sair 58](#_164oij2q3qf3)

[2.25.1 Critérios de Aceitação 58](#_xytey1jwxrgv)

[2.25.2 Regras de Negócio 59](#_m993zzx8l9uh)

[2.25.3 Teste de Aceitação - Caso de Teste Positivo 59](#_s1cr8w5quvvb)

[2.25.3 Teste de Aceitação - Caso de Teste Negativo 59](#_rktmich625q6)

[2.25.5 Requisitos Não Funcionais 60](#_dol6bakvrqr0)

[**3. ANEXOS 60**](#)

# INTRODUÇÃO

A crescente necessidade por soluções tecnológicas eficientes tem impulsionado organizações de todos os tipos a buscarem alternativas que otimizem seus processos e melhorem a gestão de suas atividades. Dentro deste contexto, surgiu a proposta de desenvolvimento de um sistema de controle interno para o Instituto Mário Gazin, uma organização sem fins lucrativos vinculada à empresa varejista Gazin.

O Instituto Mário Gazin atua ativamente na transformação social, oferecendo projetos voltados para crianças e adolescentes, promovendo atividades educacionais, culturais e musicais, como é o caso da Orquestra Jovem. Atualmente, o controle de dados relacionados às crianças atendidas, inscrições em projetos, professores e turmas é realizado por meio de arquivos do Word e planilhas do Excel, o que torna o processo manual, sujeito a erros e com baixa escalabilidade.

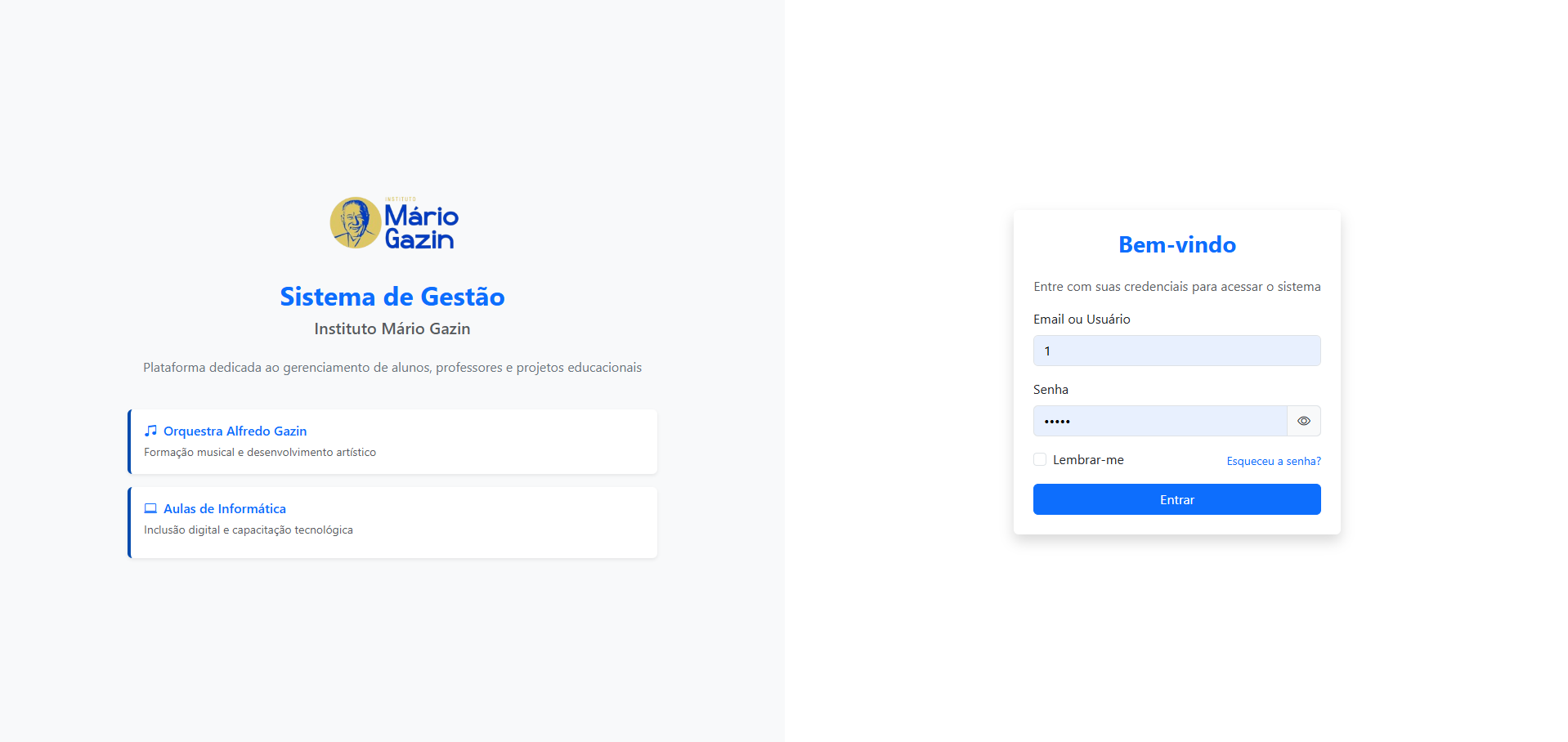
A abordagem de desenvolvimento será iterativa e incremental, permitindo a entrega de funcionalidades em fases e a incorporação de feedback contínuo dos usuários do Instituto. A escolha da metodologia específica de gerenciamento de projetos, como Scrum ou Kanban, será definida em breve, buscando a melhor adaptação à dinâmica do projeto e às necessidades de comunicação com o Instituto Mário Gazin.

A motivação para o desenvolvimento deste sistema parte da necessidade de modernizar e automatizar esse controle, proporcionando mais agilidade, segurança e organização nas atividades administrativas do Instituto. Com a implantação do sistema proposto, espera-se não apenas facilitar o trabalho dos colaboradores envolvidos, mas também permitir que o Instituto amplie seu alcance e sua capacidade de gestão, impactando positivamente a sociedade por meio de uma atuação mais eficiente e estruturada.

# 

# PROJETO DO SISTEMA

## 2.1 User Story – Tela de Login acesso ao sistema

****

**Como** usuário do sistema,  
 **Eu quero** acessar o sistema por meio de uma tela de login,  
 **Para que** eu possa utilizar as funcionalidades.

### 2.1.1 Critérios de Aceitação

* O sistema deve permitir login por e-mail e senha.
* Campos obrigatórios devem ser validados (e-mail em formato válido, senha não vazia).
* O sistema deve exibir mensagem de erro para login inválido.
* Deve existir opção de “Esqueci minha senha” com envio de e-mail para redefinição.
* O login deve redirecionar o usuário a acessar as funcionalidades do sistema

### 2.1.2 Regras de Negócio

* E-mail e senha devem corresponder aos dados cadastrados.
* A redefinição de senha deve expirar após 24 horas.
* O sistema deve bloquear acesso após 5 tentativas mal-sucedidas consecutivas.
* O sistema deve manter o usuário logado por até 30 minutos de inatividade.

### 2.1.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Login com credenciais válidas**
* **Descrição:** Verificar se o usuário consegue acessar o sistema ao informar e-mail e senha válidos.
* **Pré-requisito:** O usuário possui e-mail e senha válidos cadastrados.
  + O sistema está online e disponível.
* **Passos/Ação:** Acessar a tela de login.
  + Inserir e-mail válido no campo de e-mail.
  + Inserir a senha no campo correspondente.
  + Clicar no botão “Entrar”.
* **Resultado Esperado:** O sistema autentica o usuário com sucesso.
  + O sistema redireciona o usuário para a página inicial

### 2.1.4 Teste de Aceitação - Casos de Teste Negativos

* **CT02: Login com campos inválidos ou vazios**
* **Descrição:** Verificar se o sistema exibe mensagens de erro adequadas ao tentar fazer login com dados inválidos ou incompletos.
* **Pré-requisito:** O sistema deve validar os campos de entrada de login.
* **Passos/Ação:** Acessar a tela de login.
  + Informar e-mail com formato inválido → clicar em “Entrar”.
  + Deixar o campo de senha vazio → clicar em “Entrar”.
  + Informar e-mail e senha incorretos → clicar em “Entrar”.
* **Resultado Esperado:**
  + Exibe mensagem “E-mail inválido” para e-mails mal formatados.
  + Exibe mensagem “Senha obrigatória” quando a senha estiver vazia.
  + Exibe mensagem “Usuário ou senha incorretos” para credenciais inválidas.

### 2.1.5 Requisitos Não Funcionais

* **Segurança:** Senhas criptografadas no banco de dados.
* **Performance:** Resposta da tela em até 2 segundos.
* **Compatibilidade:** Funciona em dispositivos móveis e desktops.
* **Usabilidade:** Layout intuitivo com feedback visual para erros.

## 

## 

## 

## 

## 

## 

## 

## 2.2 User Story - Redefinição de Senha

## 

**Como** usuário do sistema  
**Eu Quero** receber um e-mail com instruções para redefinir minha senha  
**Para que** eu possa recuperar o acesso caso esqueça a senha

### 2.2.1 Critérios de Aceitação

* O sistema deve enviar um link de redefinição de senha para o e-mail informado, se ele estiver cadastrado.
* O link enviado deve expirar após um tempo determinado (ex: 24 horas).
* A nova senha deve ser salva com segurança e substitui a anterior.
* Após a redefinição, o usuário deve conseguir acessar o sistema com a nova senha.

### 2.2.2 Regras de Negócio

* O e-mail informado deve estar previamente cadastrado no sistema.
* A senha deve conter no mínimo 6 caracteres e no máximo 20.
* A nova senha não pode ser igual à anterior.
* O link de redefinição é de uso único (não pode ser reutilizado).
* Após o clique no link, o usuário deve ser redirecionado para uma página de criação de nova senha com confirmação.

### 2.2.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Envio de link para redefinição de senha**
* **Descrição:** Verificar se o sistema envia corretamente um e-mail com o link para redefinição de senha.
* **Pré-requisito:** O usuário tem e-mail válido cadastrado no sistema.
  + O servidor de e-mail está operacional.
* **Passos/Ação:** Acessar a tela de login.
  + Clicar em “Esqueci minha senha”.
* **Resultado Esperado:** Exibição da mensagem: "Você receberá um link em seu e-mail para redefinir sua senha."
  + O e-mail é enviado com sucesso com link válido.
  + O link redireciona para a página de nova senha.
* **CT02: Envio de link para redefinição de senha**
* **Descrição:** Verificar se o sistema permite que o usuário redefina sua senha com sucesso, a partir do link enviado por e-mail.
* **Pré-requisito:** O usuário recebeu o e-mail com o link de redefinição válido (token não expirado).
  + O link ainda não foi utilizado.
* **Passos/Ação:** Clicar no link recebido por e-mail.
  + Ser redirecionado para a página “Criar nova senha”.
  + Informar uma nova senha válida (mínimo 6, máximo 20 caracteres).
  + Confirmar a nova senha no segundo campo.
  + Clicar no botão “Salvar nova senha”.
* **Resultado Esperado:** O sistema valida que as senhas coincidem e atendem aos critérios.
  + A senha é atualizada com sucesso no banco de dados.
  + O link é invalidado após o uso.
  + O usuário é redirecionado para a tela de login com mensagem: *"Senha redefinida com sucesso. Faça login com sua nova senha."*

### 2.2.4 Testes de Aceitação - Casos de Teste Negativo

* **CT03: Nova senha igual à anterior**
* **Descrição:** Verificar se o sistema bloqueia a redefinição caso a nova senha seja igual à anterior.
* **Passos/Ação:**
  + Clicar no link de redefinição recebido por e-mail.
  + Informar nova senha igual à atual.
  + Confirmar.
* **Resultado Esperado:**
  + Exibição de mensagem: “A nova senha não pode ser igual à anterior.”
* **CT04: Senha fora dos critérios definidos**
* **Descrição:** Verificar se o sistema bloqueia a criação de senha fora dos critérios de segurança.
* **Passos/Ação:** Acessar a página de nova senha.
  + Informar uma senha com menos de 6 caracteres.
  + Confirmar.
* **Resultado Esperado:**
  + Exibição de mensagem: “A senha deve conter entre 6 e 20 caracteres.”

### 2.2.5 Requisitos Não Funcionais

* **Segurança:**
  + Tokens de redefinição devem ser gerados com criptografia segura.
  + Senhas devem ser criptografadas com hash seguro antes de serem armazenadas.
  + Links devem conter tokens únicos e não previsíveis.
* **Performance:**
  + Envio de e-mail deve ocorrer em até 5 segundos.
  + Validação e atualização de senha em até 3 segundos.
* **Compatibilidade:**
  + E-mails devem ser visualizados corretamente em diferentes clientes de e-mail.
  + Página de redefinição deve funcionar em dispositivos móveis e desktops.
* **Usabilidade:**
  + Interface intuitiva com instruções claras.
  + Feedback visual para validações e erros.
  + Indicador de força da senha durante a digitação.
* **Confiabilidade:**
  + Sistema deve registrar logs de tentativas de redefinição para auditoria.
  + Backup dos tokens de redefinição para recuperação em caso de falha do sistema.

## 2.3 User Story – Cadastro de Alunos

### 

**Como** usuário do sistema,  
**Eu quero** cadastrar os dados dos alunos no sistema,  
**Para que** eu possa organizar, consultar e acompanhar seus dados de forma eficiente.

### 2.3.1 Critérios de Aceitação

* O sistema deve permitir o cadastro de nome, idade, responsável, turma e documentos do aluno.
* Campos obrigatórios devem ser validados.
* Deve ser possível editar e excluir cadastros.
* Aluno deve ser vinculado a uma turma.

### 2.3.2 Regras de Negócio

* Um aluno pode estar em apenas uma turma por projeto.
* É obrigatório o CPF do responsável para alunos menores de idade.
* Alunos devem estar vinculados a um projeto ativo.

### 2.3.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Cadastro realizado com sucesso**
* **Descrição:** Verificar se o sistema realiza corretamente o cadastro completo de um aluno.
* **Pré-requisito:** Usuário logado com permissão de cadastro.
* **Passos/Ação:** Acessar a tela "Alunos"
  + Preencher todos os campos obrigatórios (Nome, Email, Telefone, Responsável, Turma)
  + Clicar em "Cadastrar"
* **Resultado Esperado:** O sistema salva os dados e exibe o aluno na lista de alunos.
  + Mensagem de sucesso exibida: “Aluno cadastrado com sucesso.”

### 2.3.4 Testes de Aceitação - Casos de Teste Negativo

* **CT02: Campo obrigatório não preenchido**
* **Descrição:** Verificar se o sistema impede o cadastro quando campos obrigatórios estão em branco.
* **Pré-requisito:** Usuário logado no sistema.
* **Passos/Ação:** Acessar a tela "Alunos"
  + Preencher apenas alguns campos
  + Clicar em "Cadastrar"
* **Resultado Esperado:** O sistema valida que possuem campos obrigatórios sem preenchimento.
  + O sistema exibe uma mensagem de erro: “Preencha todos os campos obrigatórios.”
  + Nenhum dado é salvo no banco.

### 2.3.5 Requisitos Não Funcionais

* **Desempenho:** O Cadastro deve ser processado em até 2 segundos.
* **Segurança:** Apenas usuários logados com permissão podem cadastrar.
* **Escalabilidade:** Suporte a mais de mil alunos.
* **Manutenibilidade:** Código versionado no GitHub.
* **Compatibilidade:** Compatível com desktop e mobile.
* **Usabilidade:** Interface amigável e validada.

## 2.4 User Story – Listagem de Alunos

### 

**Como** usuário do sistema,

**Eu quero** visualizar uma lista com os alunos cadastrados,

**Para que** eu possa acompanhar os alunos registrados por turma e projeto.

### 2.4.1 Critérios de Aceitação

* O sistema deve exibir uma tabela com todos os alunos cadastrados.
* A listagem deve permitir a ordenação por nome e por turma.

### 2.4.2 Regras de Negócio

* A listagem deve exibir o total de alunos visíveis no rodapé da tabela.
* A exibição deve respeitar a paginação (caso existam muitos alunos).

### 2.4.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Exibição da lista de alunos**
* **Descrição:** Verificar se o sistema exibe corretamente os alunos cadastrados.
* **Pré-requisito:** Ter alunos cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Alunos"
* **Resultado Esperado**: A tabela exibe corretamente os nomes, turmas, responsáveis e status de cada aluno.

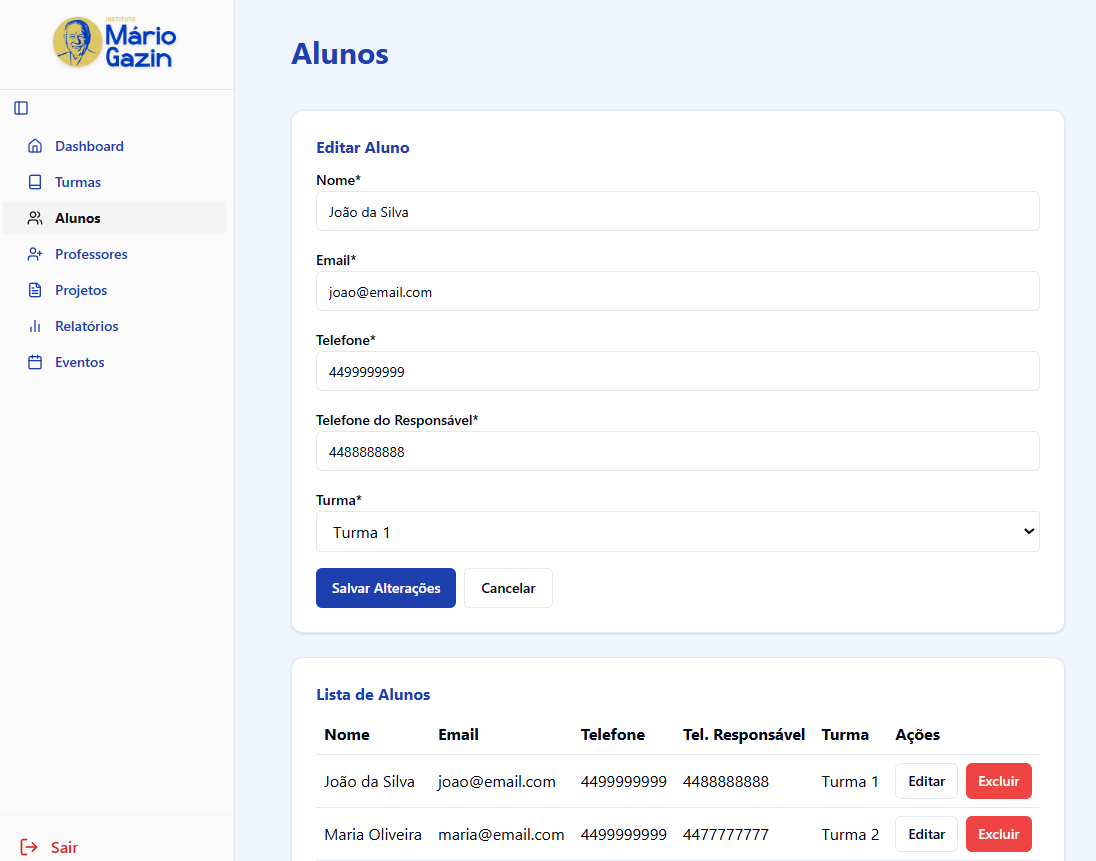
### 2.4.4 Testes de Aceitação - Casos de Teste Negativo

* **CT02: Listagem sem alunos cadastrados**
* **Descrição:** Verificar o comportamento do sistema ao acessar a listagem sem alunos cadastrados.
* **Pré-requisito:** Nenhum aluno cadastrado no sistema.
* **Passos/Ação:** Acessar a tela "Alunos"
* **Resultado Esperado:** O sistema exibe uma mensagem como "Nenhum aluno cadastrado".
  + A tabela aparece vazia.

### 2.4.5 Requisitos Não Funcionais

* **Desempenho:** A listagem deve ser carregada em até 2 segundos, mesmo com grande volume de dados.
* **Segurança:** Apenas usuários logados com permissão podem visualizar a listagem completa dos alunos.
* **Escalabilidade:** O sistema deve suportar a listagem de mais de mil alunos sem perda de performance.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface da listagem deve ser clara, intuitiva e validada com os usuários finais.

## 2.5 User Story – Edição de Alunos



**Como** usuário do sistema,  
**Eu quero** editar os dados de um aluno,  
**Para que** eu possa corrigir ou atualizar suas informações quando necessário.

### 2.5.1 Critérios de Aceitação

* O sistema deve permitir alterar qualquer campo do cadastro.
* Deve ser possível salvar ou cancelar a edição.

### 2.5.2 Regras de Negócio

* Todos os campos obrigatórios devem ser validados novamente na edição.

### 2.5.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Edição realizada com sucesso**
* **Descrição:** Verificar se o sistema permite editar os dados corretamente.
* **Pré-requisito:** Ter alunos cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Alunos"
  + Clicar no botão "Editar" de um aluno
  + Alterar os dados desejados
  + Clicar em "Salvar"
* **Resultado Esperado:** É exibida a mensagem “Dados alterados com sucesso!”
  + Dados são atualizados
* **CT02: Edição cancelada**
* **Descrição:** Verificar se ao clicar em "Cancelar" durante a edição, as alterações não são salvas.
* **Pré-requisito:** Ter alunos cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Alunos"
  + Clicar no botão "Editar" de um aluno
  + Alterar os dados desejados
  + Clicar em "Cancelar"
* **Resultado Esperado:** Nenhuma alteração é aplicada.
  + O cadastro original permanece inalterado.

### 2.5.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Campo obrigatório apagado na edição**
* **Descrição:** Verificar se o sistema trata corretamente a tentativa de edição de um aluno quando um campo obrigatório é apagado.
* **Pré-requisito:** Ter alunos cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Alunos"
  + Clicar no botão "Editar" de um aluno
  + Apagar algum campo obrigatório
  + Clicar em "Salvar"
* **Resultado Esperado:** É exibida a mensagem “Campo obrigatório não preenchido!”
  + O cadastro original permanece inalterado.

### 2.5.5 Requisitos Não Funcionais

* **Desempenho:** A edição dos dados deve ser processada em até 2 segundos após a confirmação.
* **Segurança:** Apenas usuários autenticados com permissão de edição poderão alterar os dados dos alunos.
* **Escalabilidade:** O sistema deve permitir a edição de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de edição deve ser clara, intuitiva e validada com os usuários finais.

## 2.6 User Story – Exclusão de Alunos

**Como** usuário do sistema,  
**Eu quero** excluir o cadastro de um aluno,  
**Para que** eu possa remover registros que não são mais necessários.

### 2.6.1 Critérios de Aceitação

* O sistema deve pedir confirmação antes de excluir.
* Ao excluir, o aluno deve ser removido da listagem.

### 2.6.2 Regras de Negócio

* Um aluno só pode ser excluído se não estiver vinculado a turmas ativas.

### 2.6.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Exclusão realizada com sucesso**
* **Descrição:** Verificar se o sistema exclui corretamente um aluno.
* **Pré-requisito:** Aluno não deve estar vinculado a turma ativa.
* **Passos/Ação:** Acessar a tela "Alunos"
  + Clicar no botão "Excluir" de um aluno
  + Confirmar a exclusão
* **Resultado Esperado:** É exibida a mensagem “Exclusão realizada com sucesso!”
  + O Aluno é removido da listagem padrão.
* **CT02: Exclusão cancelada**
* **Descrição:** Verificar se ao cancelar a exclusão o aluno permanece no sistema.
* **Pré-requisito:** Ter alunos cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Alunos"
  + Clicar no botão "Excluir" de um aluno
  + Clicar em "Cancelar" na confirmação
* **Resultado Esperado:** Nenhuma ação é aplicada.
  + O aluno continua visível na listagem.

### 2.6.4 Teste de Aceitação - Caso de Teste Negativo

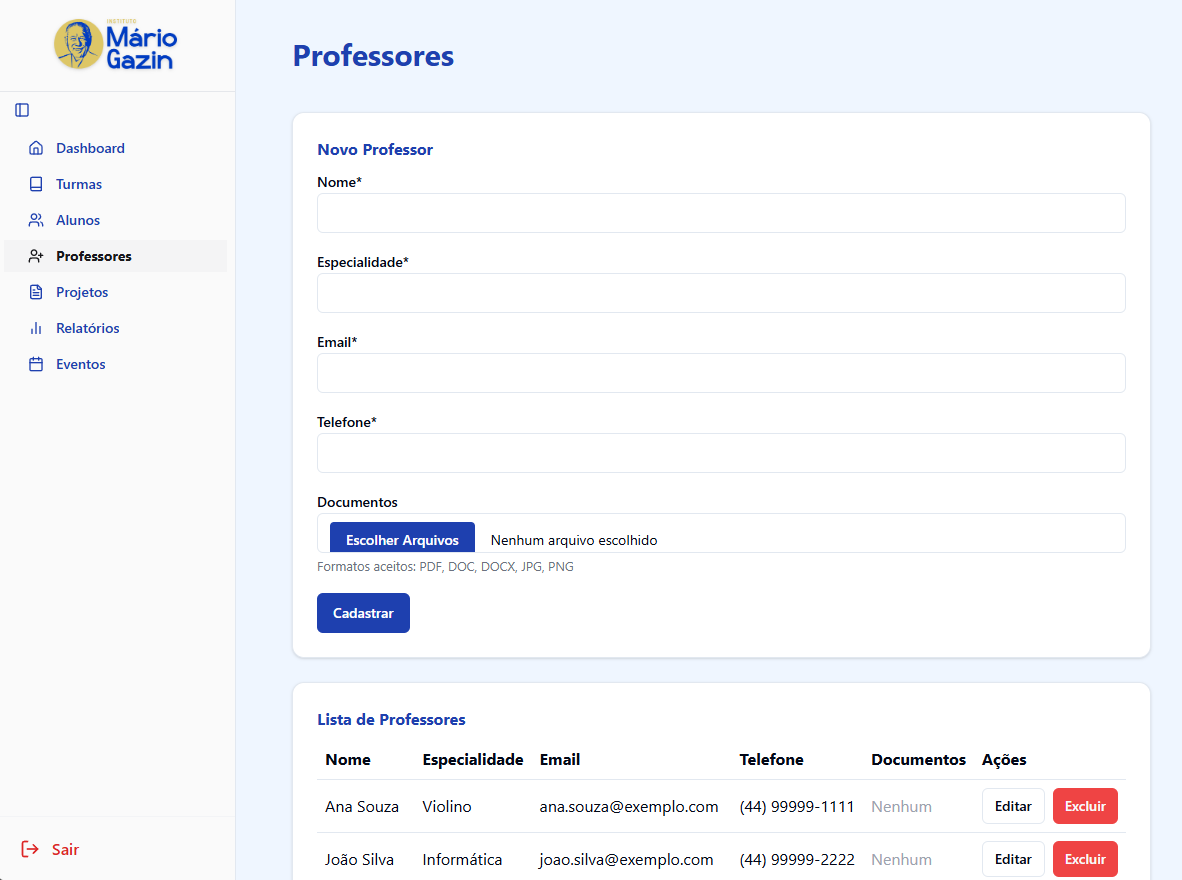
* **CT03: Exclusão de aluno vinculado a turma ativa**
* **Descrição:** Verificar se o sistema impede a exclusão de um aluno que está vinculado a uma turma ativa.
* **Pré-requisito:** O aluno deve estar previamente vinculado a uma turma ativa no sistema.
* **Passos/Ação:** Acessar a tela "Alunos"
  + Clicar no botão "Excluir" de um aluno vinculado a uma turma ativa
  + Confirmar a exclusão
* **Resultado Esperado:** O sistema exibe a mensagem: "Não é possível excluir o aluno. Ele está vinculado a uma turma ativa."
  + Nenhuma exclusão é realizada.

### 2.6.5 Requisitos Não Funcionais

* **Desempenho:** A exclusão dos dados deve ser processada em até 2 segundos após a confirmação.
* **Segurança:** Apenas usuários autenticados com permissão podem realizar a exclusão dos alunos.
* **Escalabilidade:** O sistema deve permitir a exclusão de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de exclusão deve ser clara, intuitiva e validada com os usuários finais.

## 

## 2.7 User Story – Cadastro de Professores



**Como** usuário do sistema,  
 **Eu quero** cadastrar professores e alocá-los às turmas,  
 **Para que** eu possa organizar quem é responsável por cada grupo.

### 2.7.1 Critérios de Aceitação

* O sistema deve permitir cadastro de nome, e-mail, telefone, currículo e documentos do professor.
* O campo de documentos deve aceitar os formatos: PDF, DOC, DOCX, JPG, PNG.
* O sistema deve validar que todos os campos obrigatórios estão preenchidos antes de permitir o cadastro.
* Deve ser possível atribuir o professor a uma ou mais turmas (mesmo que essa ação ocorra em outra etapa/tela).

### 2.7.2 Regras de Negócio

* Um professor pode estar em múltiplas turmas, mas não com conflitos de horários.
* Documentos devem ser obrigatórios para efetivar o cadastro.
* A lista deve apresentar ordenação
* O campo de email deve ser validado para evitar duplicidades e formatos.

### 2.7.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Cadastro de professor com dados válidos**
* **Descrição:** Verificar se o sistema cadastra corretamente um professor com todos os campos obrigatórios preenchidos.
* **Pré-requisito:** Nenhum professor com o mesmo e-mail está cadastrado no sistema.
* **Passos/Ação:** Acessar a tela "Professores"
  + Preencher os campos: Nome, Especialidade, E-mail, Telefone.
  + Fazer upload de um arquivo PDF no campo "Documentos".
  + Clicar no botão "Cadastrar".
* **Resultado Esperado:** O sistema exibe mensagem de sucesso: "Professor cadastrado com sucesso."
  + O professor aparece na listagem de professores.

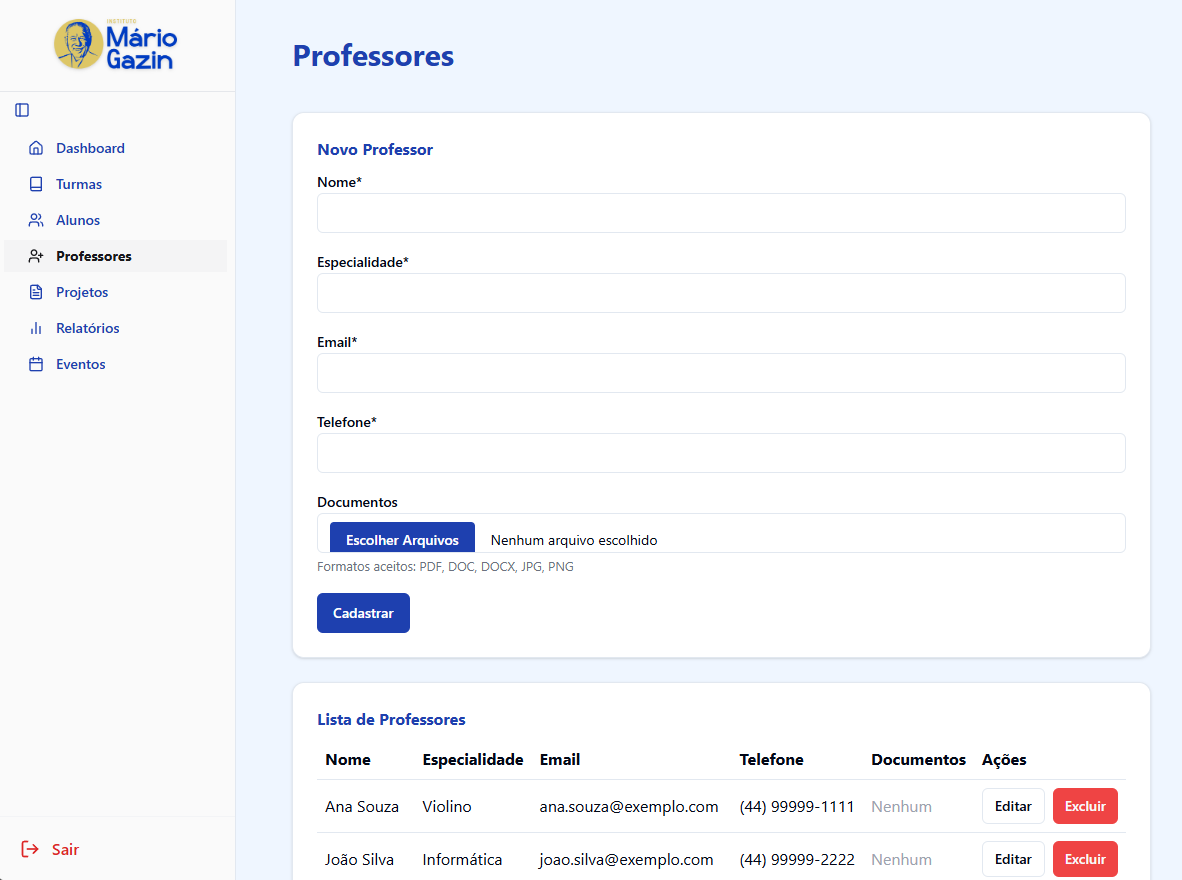
### 2.7.4 Teste de Aceitação - Caso de Teste Negativo

* **CT02: Cadastro sem upload de documentos**
* **Descrição:** Verificar se o sistema impede o cadastro de professor sem anexar um documento.
* **Pré-requisito:** O professor ainda não está cadastrado.
* **Passos/Ação:** Acessar a tela "Professores"
  + Preencher os campos obrigatórios, exceto o upload de documento.
  + Clicar no botão "Cadastrar".
* **Resultado Esperado:** O sistema exibe mensagem de erro: "O campo de documentos é obrigatório."
  + O professor não é cadastrado. Nenhum dado é salvo no sistema.
* **CT03: Upload de documento com formato ou tamanho inválido**
* **Descrição:** Verificar se o sistema impede o envio de arquivos com formato não permitido ou tamanho superior ao limite definido.
* **Pré-requisito:** O usuário está na tela de cadastro de professor.
* **Passos/Ação:** Acessar a tela "Professores"
  + Preencher os campos obrigatórios.
  + Tentar fazer upload de um arquivo com formato inválido (ex: .exe) ou com tamanho superior a 5MB.
  + Clicar no botão "Cadastrar".
* **Resultado Esperado:** O sistema exibe a mensagem: "Formato de arquivo inválido ou tamanho excede o limite permitido (5MB)."
  + Nenhum professor é cadastrado.

### 2.7.5 Requisitos Não Funcionais

* **Desempenho:** Processamento de cadastro em até 3 segundos.
* **Segurança:** Apenas usuários autenticados com permissão podem realizar o cadastro.
* **Escalabilidade:** O sistema deve permitir o cadastro de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de cadastro deve ser clara, intuitiva e validada com os usuários finais.

## 2.8 User Story – Listar Professores



**Como** usuário do sistema,  
 **Eu quero** visualizar a lista de professores cadastrados,  
 **Para que** eu possa consultar, localizar e gerenciar as informações dos docentes de forma prática.

### 2.8.1 Critérios de Aceitação

* O sistema deve exibir uma lista contendo os professores cadastrados
* A listagem deve conter paginação com no máximo 10 registros por página.
* A lista deve ser apresentada por ordem de cadastro.

### 2.8.2 Regras de Negócio

* A listagem deve exibir o total de professores visíveis no rodapé da tabela.
* A exibição deve respeitar a paginação (caso existam muitos alunos).

### 2.8.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Listagem exibida corretamente**
* **Descrição**: Verificar se a listagem de professores aparece com as informações corretas e organizadas.
* **Pré-requisito:** Existem professores cadastrados e ativos no sistema.
* **Passos/Ação:** Acessar a tela de "Professores".
* **Resultado Esperado:** O sistema exibe corretamente os professores cadastrados

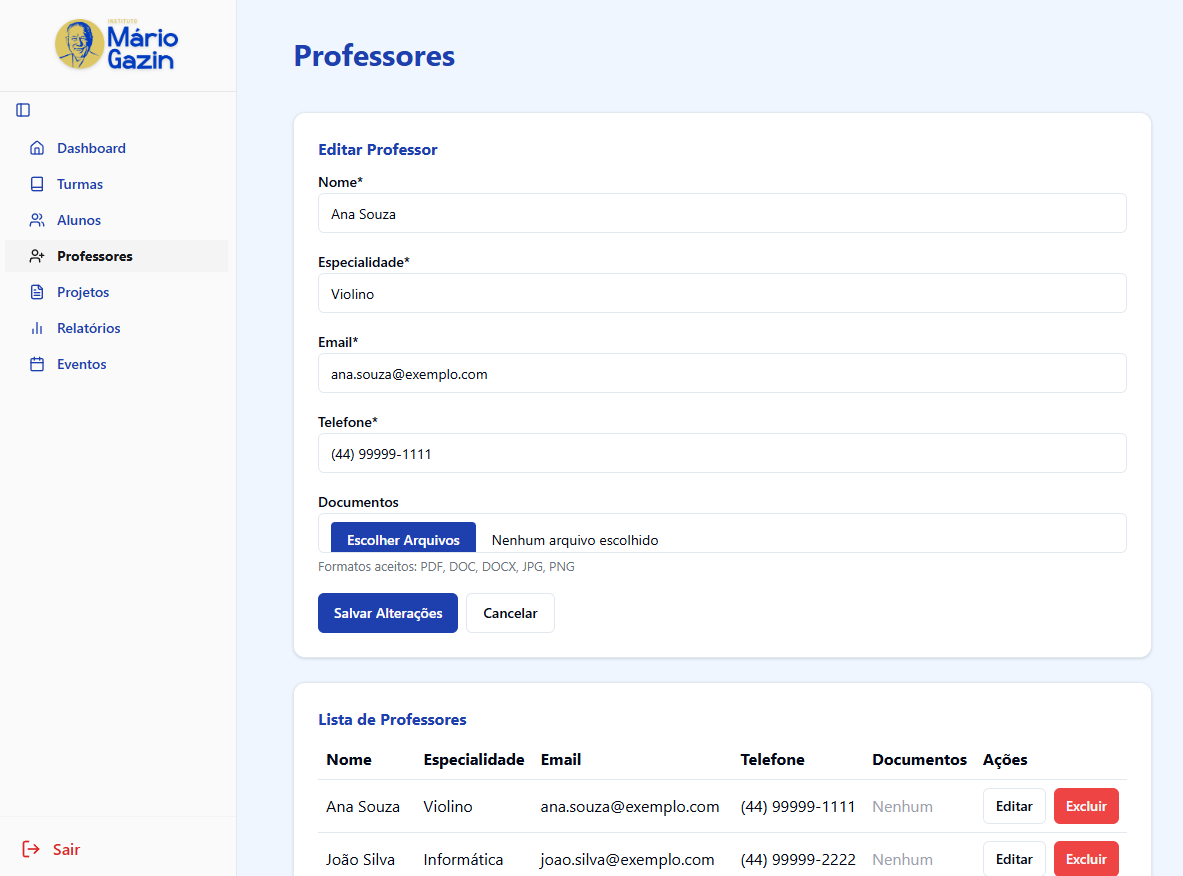
### 2.8.4 Teste de Aceitação - Caso de Teste Negativo

* **CT02: Listagem sem professores cadastrados**
* **Descrição:** Verificar o comportamento do sistema ao acessar a listagem sem professores cadastrados.
* **Pré-requisito:** Nenhum professor cadastrado no sistema.
* **Passos/Ação**: Acessar a tela "Professores"
* **Resultado Esperado:** O sistema exibe uma mensagem como "Nenhum professor cadastrado".
  + A tabela aparece vazia.

### 2.8.5 Requisitos Não Funcionais

* **Desempenho:** Processamento de listagem em até 3 segundos.
* **Segurança:** Apenas usuários autenticados com permissão podem visualizar a listagem.
* **Escalabilidade:** O sistema deve permitir a listagem de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de listagem deve ser clara, intuitiva e validada com os usuários finais.

## 2.9 User Story – Edição de Professores



**Como** usuário do sistema,  
 **Eu quero** editar os dados de um professor cadastrado,  
 **Para que** eu possa corrigir informações ou atualizá-las quando necessário.

### 2.9.1 Critérios de Aceitação

* O sistema deve permitir editar os dados do professor.
* Os campos obrigatórios devem ser validados
* Deve ser possível cancelar a edição antes de salvar.

### 2.9.2 Regras de Negócio

* Todos os campos obrigatórios devem ser validados novamente na edição.
* Todos os campos obrigatórios devem estar preenchidos corretamente para permitir a edição.

### 2.9.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Edição bem-sucedida**
* **Descrição:** Verificar se a edição dos dados de um professor é realizada com sucesso.
* **Pré-requisito:** Professor ativo já cadastrado.
* **Passos/Ação:** Acessar a tela de "Professores".
  + Clicar em "Editar" ao lado de um professor.
  + Alterar os dados
  + Clicar em “Salvar”
* **Resultado Esperado:** É exibida a mensagem “Dados alterados com sucesso!”
  + Dados são atualizados
* **CT02: Edição cancelada**
* **Descrição:** Verificar se ao clicar em "Cancelar" durante a edição, as alterações não são salvas.
* **Pré-requisito:** Ter professores cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Professores"
  + Clicar no botão "Editar" de um professor
  + Alterar os dados desejados
  + Clicar em "Cancelar"
* **Resultado Esperado:** Nenhuma alteração é aplicada.
  + O cadastro original permanece inalterado.

#### 

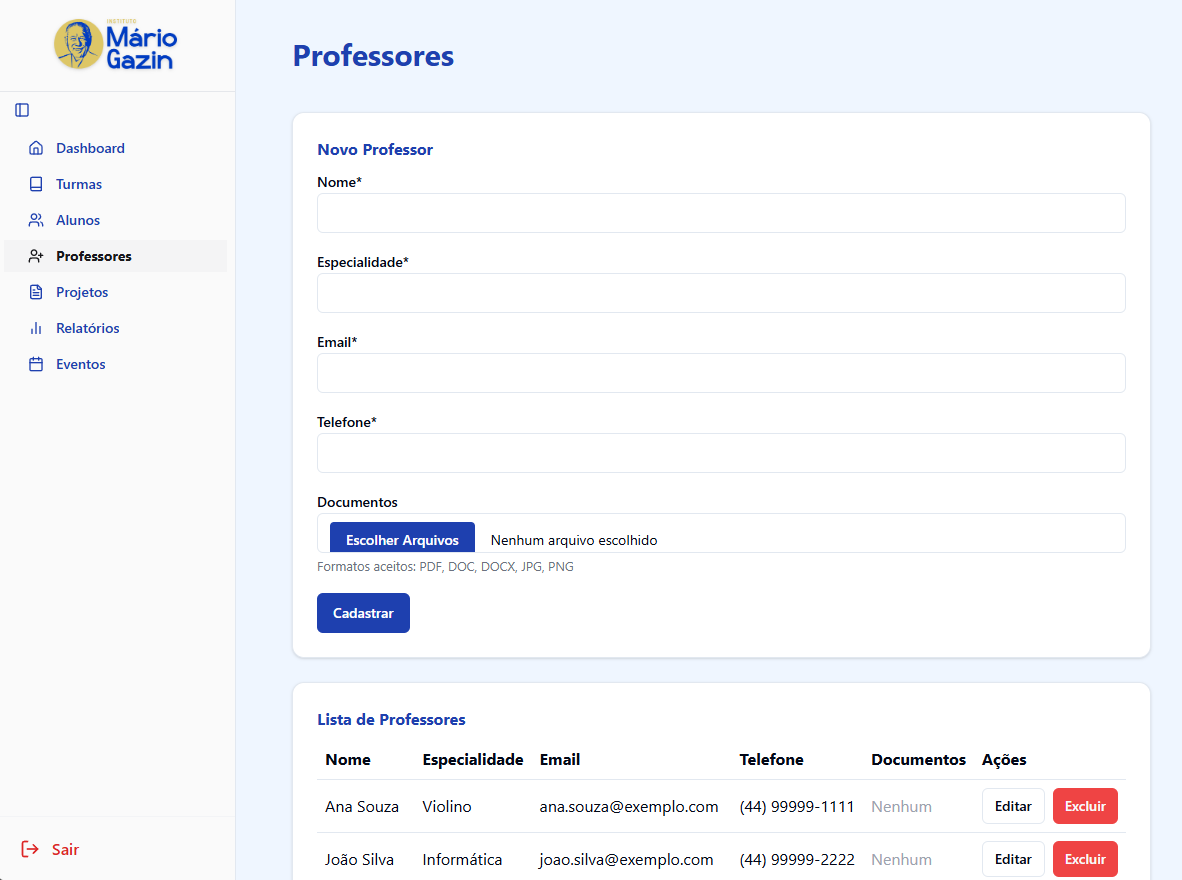
### 2.9.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Campo obrigatório apagado na edição**
* **Descrição:** Verificar se o sistema trata corretamente a tentativa de edição de um professor quando um campo obrigatório é apagado.
* **Pré-requisito:** Ter professores cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Professores"
  + Clicar no botão "Editar" de um professor
  + Apagar algum campo obrigatório
  + Clicar em "Salvar"
* **Resultado Esperado:** É exibida a mensagem “Campo obrigatório não preenchido!”
  + O cadastro original permanece inalterado.

### 2.9.5 Requisitos Não Funcionais

* **Desempenho:** Processamento de cadastro em até 3 segundos.
* **Segurança:** Apenas usuários autenticados com permissão podem editar o cadastro.
* **Escalabilidade:** O sistema deve permitir a edição de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de edição deve ser clara, intuitiva e validada com os usuários finais.

## 2.10 User Story – Exclusão de Professores



**Como** usuário do sistema,  
**Eu quero** poder excluir professores do sistema,  
**Para que** eu possa remover registros que não são mais necessários.

### 2.10.1 Critérios de Aceitação

* O sistema deve pedir confirmação antes de excluir.
* Ao excluir, o professor deve ser removido da listagem.

### 2.10.2 Regras de Negócio

* Um professor só pode ser excluído se não estiver vinculado a turmas ativas.

### 2.10.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Exclusão realizada com sucesso**
* **Descrição:** Verificar se o sistema exclui corretamente um professor.
* **Pré-requisito:** Professor não deve estar vinculado a turma ativa.
* **Passos/Ação:** Acessar a tela "Professores"
  + Clicar no botão "Excluir" de um professor
  + Confirmar a exclusão
* **Resultado Esperado:** É exibida a mensagem “Exclusão realizada com sucesso!”
  + O Professor é removido da listagem padrão.
* **CT02: Exclusão cancelada**
* **Descrição:** Verificar se ao cancelar a exclusão o professor permanece no sistema.
* **Pré-requisito:** Ter professores cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Professores"
  + Clicar no botão "Excluir" de um professor
  + Clicar em "Cancelar" na confirmação
* **Resultado Esperado:** Nenhuma ação é aplicada.
  + O professor continua visível na listagem.

### 2.10.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Exclusão de professor vinculado a turma ativa**
* **Descrição:** Verificar se o sistema impede a exclusão de um professor que está vinculado a uma turma ativa.
* **Pré-requisito:** O professor deve estar previamente vinculado a uma turma ativa no sistema.
* **Passos/Ação:** Acessar a tela "Professores"
  + Clicar no botão "Excluir" de um professor vinculado a uma turma ativa
  + Confirmar a exclusão
* **Resultado Esperado:** O sistema exibe a mensagem: "Não é possível excluir o professor. Ele está vinculado a uma turma ativa."
  + Nenhuma exclusão é realizada.

### 2.10.5 Requisitos Não Funcionais

* **Desempenho:** A exclusão dos dados deve ser processada em até 2 segundos após a confirmação.
* **Segurança:** Apenas usuários autenticados com permissão podem realizar a exclusão dos professores.
* **Escalabilidade:** O sistema deve permitir a exclusão de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de exclusão deve ser clara, intuitiva e validada com os usuários finais.

## 2.11 User Story – Cadastro de Turmas

### 

**Como** usuário do sistema,  
**Eu quero** poder cadastrar novas turmas no sistema,  
**Para que** eu possa organizar os alunos por grupos de atividades específicas.

### 2.11.1 Critérios de Aceitação

* O sistema deve permitir inserir nome da turma obrigatoriamente.
* O sistema deve permitir inserir descrição da turma.
* O sistema deve permitir selecionar um projeto existente.
* O sistema deve permitir selecionar um professor responsável.
* O sistema deve permitir definir horário da turma.
* Após o cadastro, a turma deve aparecer na listagem de turmas.

### 2.11.2 Regras de Negócio

* O nome da turma é obrigatório e deve ser único dentro do mesmo projeto.
* Um professor pode ser responsável por múltiplas turmas.
* A turma deve estar vinculada obrigatoriamente a um projeto existente.

### 2.11.3 Teste de Aceitação - Caso de Teste Positivo

### **CT01: Cadastro realizado com sucesso**

* **Descrição:** Verificar se o sistema cadastra corretamente uma nova turma.
* **Pré-requisito:** Ter pelo menos um projeto e um professor cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Nova Turma"
  + Preencher o campo "Nome" com dados válidos
  + Preencher o campo "Descrição" (opcional)
  + Selecionar um "Projeto" existente
  + Selecionar um "Professor" responsável
  + Preencher o campo "Horário"
  + Clicar em "Cadastrar"
* **Resultado Esperado:** É exibida a mensagem "Turma cadastrada com sucesso!"
  + A nova turma aparece na listagem de turmas.
* **CT02: Cadastro com campos opcionais vazios**
* **Descrição:** Verificar se o sistema permite cadastrar turma sem preencher campos opcionais.
* **Pré-requisito:** Ter pelo menos um projeto e um professor cadastrados no sistema.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Nova Turma"
  + Preencher apenas os campos obrigatórios
  + Clicar em "Cadastrar"
* **Resultado Esperado:** O cadastro é realizado com sucesso.
  + A turma é criada com os campos opcionais em branco.

### 2.11.4 Teste de Aceitação - Caso de Teste Negativo

### **CT0**3**: Cadastro com nome duplicado no mesmo projeto**

* **Descrição:** Verificar se o sistema impede o cadastro de turma com nome já existente no mesmo projeto.
* **Pré-requisito:** Já existir uma turma cadastrada em um projeto específico.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Nova Turma"
  + Inserir nome de turma já existente no mesmo projeto
  + Preencher demais campos obrigatórios
  + Clicar em "Cadastrar"
* **Resultado Esperado:** O sistema exibe a mensagem: "Já existe uma turma com este nome neste projeto."
  + Nenhum cadastro é realizado.
* **CT04: Cadastro sem preencher campos obrigatórios**
* **Descrição:** Verificar se o sistema impede cadastro sem campos obrigatórios.
* **Pré-requisito:** Ter acesso ao formulário de cadastro de turmas.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Nova Turma"
  + Deixar campos obrigatórios em branco
  + Clicar em "Cadastrar"
* **Resultado Esperado:** O sistema exibe mensagens de validação nos campos obrigatórios.
  + Nenhum cadastro é realizado.

### 2.11.5 Requisitos Não Funcionais

* **Desempenho:** Processamento de cadastro em até 3 segundos.
* **Segurança:** Apenas usuários autenticados com permissão podem realizar o cadastro.
* **Escalabilidade:** O sistema deve permitir o cadastro de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de cadastro deve ser clara, intuitiva e validada com os usuários finais.

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 2.12 User Story – Listar Turmas

**Como** usuário do sistema,   
**Eu quero** visualizar todas as turmas cadastradas,   
**Para que** eu possa consultar informações e gerenciar as turmas existentes.

### 2.12.1 Critérios de Aceitação

* O sistema deve exibir lista com todas as turmas cadastradas.
* Cada item da lista deve mostrar: nome, projeto, professor e horário.
* O sistema deve oferecer opções para editar e excluir cada turma.

### 2.12.2 Regras de Negócio

* Apenas turmas ativas devem aparecer na listagem padrão.

### 2.12.3 Teste de Aceitação - Caso de Teste Positivo

### **CT01: Visualização da listagem completa**

* **Descrição:** Verificar se o sistema exibe corretamente todas as turmas cadastradas.
* **Pré-requisito:** Ter turmas previamente cadastradas no sistema.
* **Passos/Ação:** Acessar a tela "Turmas"
* **Resultado Esperado:** São exibidas todas as turmas ativas.
  + Cada turma mostra nome, projeto, professor e horário.
* **CT02: Busca por nome de turma**
* **Descrição:** Verificar se a funcionalidade de busca retorna resultados corretos.
* **Pré-requisito:** Ter turmas cadastradas no sistema.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Digitar nome de uma turma existente no campo de busca
* **Resultado Esperado:** São exibidas apenas as turmas que correspondem ao termo buscado.

### 2.12.4 Teste de Aceitação - Caso de Teste Negativo

### **CT0**3**: Busca sem resultados**

* **Descrição:** Verificar comportamento quando busca não retorna resultados.
* **Pré-requisito:** Estar na tela de listagem de turmas.
* **Passos/Ação:** Digitar termo que não corresponde a nenhuma turma
* **Resultado Esperado:** O sistema exibe mensagem "Nenhuma turma encontrada."
  + A listagem fica vazia ou mostra mensagem informativa.

### 2.3.4 Requisitos Não Funcionais

* **Desempenho:** Processamento de listagem em até 3 segundos.
* **Segurança:** Apenas usuários autenticados com permissão podem visualizar a listagem.
* **Escalabilidade:** O sistema deve permitir a listagem de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de listagem deve ser clara, intuitiva e validada com os usuários finais.

## 

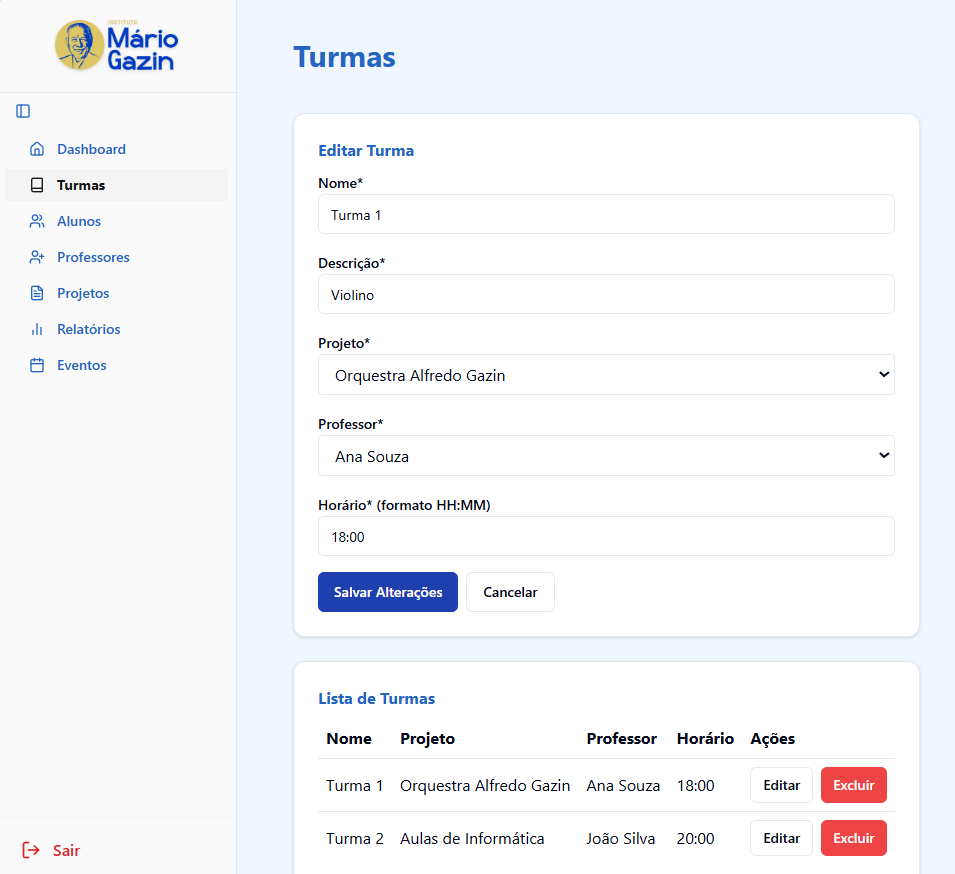
## 

## 

## 

## 

## 2.13 User Story – Edição de Turmas



**Como** usuário do sistema,   
**Eu quero** poder editar informações das turmas cadastradas,   
**Para que** eu possa manter os dados atualizados conforme necessário.

### 2.13.1 Critérios de Aceitação

* O sistema deve permitir alterar nome, descrição, projeto, professor e horário.
* O sistema deve validar se novo nome não duplica com turma existente no mesmo projeto.
* Após a edição, as alterações devem ser refletidas na listagem.
* O sistema deve exibir mensagem de sucesso após salvamento.

### 2.13.2 Regras de Negócio

* Não é possível alterar uma turma para ter nome duplicado dentro do mesmo projeto.
* Todos os campos obrigatórios devem ser mantidos preenchidos.

### 2.13.3 Teste de Aceitação - Caso de Teste Positivo

### **CT01: Edição realizada com sucesso**

* **Descrição:** Verificar se o sistema salva corretamente as alterações em uma turma.
* **Pré-requisito:** Ter turma previamente cadastrada no sistema.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Editar" de uma turma
  + Alterar informações desejadas
  + Clicar em "Salvar"
* **Resultado Esperado:** É exibida a mensagem "Turma editada com sucesso!"
  + As alterações são refletidas na listagem.
* **CT02: Cancelamento da edição**
* **Descrição:** Verificar se ao cancelar a edição nenhuma alteração é salva.
* **Pré-requisito:** Estar editando uma turma.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Editar" de uma turma
  + Fazer alterações no formulário
  + Clicar em "Cancelar"
* **Resultado Esperado:** Nenhuma alteração é salva.
  + Os dados originais são mantidos.

### 2.13.4 Teste de Aceitação - Caso de Teste Negativo

### **CT0**3**: Edição com nome duplicado**

* **Descrição:** Verificar se o sistema impede salvar turma com nome duplicado no mesmo projeto.
* **Pré-requisito:** Existir pelo menos duas turmas no mesmo projeto.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Editar" de uma turma
  + Alterar nome para nome já existente no mesmo projeto
  + Clicar em "Salvar"
* **Resultado Esperado:** O sistema exibe mensagem: "Já existe uma turma com este nome neste projeto."
  + Nenhuma alteração é salva.
* **CT04: Edição removendo campos obrigatórios**
* **Descrição:** Verificar se o sistema impede salvar turma sem campos obrigatórios.
* **Pré-requisito:** Estar editando uma turma.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Editar" de uma turma
  + Remover conteúdo de campos obrigatórios
  + Clicar em "Salvar"
* **Resultado Esperado:** O sistema exibe mensagens de validação.
  + Nenhuma alteração é salva.

### 2.13.5 Requisitos Não Funcionais

* **Desempenho:** Processamento de cadastro em até 3 segundos.
* **Segurança:** Apenas usuários autenticados com permissão podem editar o cadastro.
* **Escalabilidade:** O sistema deve permitir a edição de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de edição deve ser clara, intuitiva e validada com os usuários finais.

## 2.14 User Story – Exclusão de Turmas

### 

**Como** usuário do sistema,   
**Eu quero** poder excluir turmas do sistema,   
**Para que** eu possa remover registros que não são mais necessários.

### 2.14.1 Critérios de Aceitação

* O sistema deve pedir confirmação antes de excluir.
* Ao excluir, a turma deve ser removida da listagem.

### 2.14.2 Regras de Negócio

* Uma turma só pode ser excluída se não possuir alunos matriculados.

### 2.14.3 Teste de Aceitação - Caso de Teste Positivo

### **CT01: Exclusão realizada com sucesso**

* **Descrição:** Verificar se o sistema exclui corretamente uma turma.
* **Pré-requisito:** Turma não deve possuir alunos matriculados.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Excluir" de uma turma
  + Confirmar a exclusão
* **Resultado Esperado:** É exibida a mensagem "Exclusão realizada com sucesso!"
  + A turma é removida da listagem padrão.
* **CT02: Exclusão cancelada**
* **Descrição:** Verificar se ao cancelar a exclusão a turma permanece no sistema.
* **Pré-requisito:** Ter turmas cadastradas no sistema.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Excluir" de uma turma
  + Clicar em "Cancelar" na confirmação
* **Resultado Esperado:** Nenhuma ação é aplicada.
  + A turma continua visível na listagem.

### 2.14.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Exclusão de turma com alunos matriculados**
* **Descrição:** Verificar se o sistema impede a exclusão de uma turma que possui alunos matriculados.
* **Pré-requisito:** A turma deve possuir pelo menos um aluno matriculado.
* **Passos/Ação:** Acessar a tela "Turmas"
  + Clicar no botão "Excluir" de uma turma com alunos matriculados
  + Confirmar a exclusão
* **Resultado Esperado:** O sistema exibe a mensagem: "Não é possível excluir a turma. Existem alunos matriculados nesta turma."
  + Nenhuma exclusão é realizada.

### 2.14.5 Requisitos Não Funcionais

* **Desempenho:** A exclusão dos dados deve ser processada em até 2 segundos após a confirmação.
* **Segurança:** Apenas usuários autenticados com permissão podem realizar a exclusão das turmas.
* **Escalabilidade:** O sistema deve permitir a exclusão de dados em uma base com mais de mil turmas sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de exclusão deve ser clara, intuitiva e validada com os usuários finais.

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

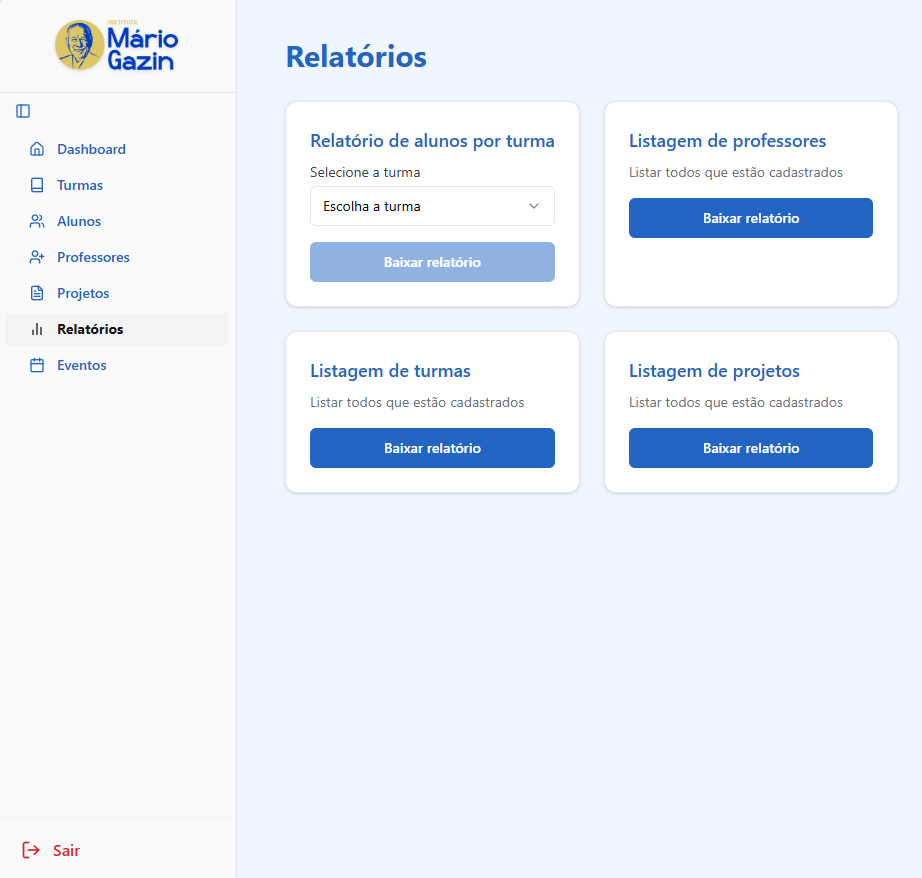
## 

## 

## 

## 

## 2.15 User Story – Geração de Relatórios em PDF



**Como** usuário do sistema,   
**Eu quero** poder gerar relatórios em formato PDF,   
**Para que** eu possa imprimir, compartilhar e arquivar informações do sistema de forma padronizada.

### 2.15.1 Critérios de Aceitação

* O sistema deve permitir gerar relatórios em formato PDF.
* O PDF deve conter cabeçalho com logo e informações do Instituto Mário Gazin.
* O PDF deve incluir data e hora de geração do relatório.
* O sistema deve permitir download direto do arquivo PDF.
* O PDF deve ter formatação clara e profissional.

### 2.15.2 Regras de Negócio

* Todos os relatórios devem seguir layout padrão do Instituto.
* O nome do arquivo PDF deve incluir tipo do relatório e data de geração.
* Relatórios só podem ser gerados por usuários autenticados.

### 2.15.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Geração de PDF realizada com sucesso**
* **Descrição:** Verificar se o sistema gera corretamente relatório em PDF.
* **Pré-requisito:** Estar em qualquer tela de relatório com dados disponíveis.
* **Passos/Ação:** Acessar a tela "Relatórios"
  + Selecionar tipo de relatório desejado
  + Configurar filtros necessários
  + Clicar em "Baixar relatório"
* **Resultado Esperado:** O arquivo PDF é gerado e download é iniciado automaticamente.
  + O PDF contém cabeçalho padronizado, dados solicitados e data/hora de geração.

### 2.15.4 Teste de Aceitação - Caso de Teste Negativo

* **CT02: Tentativa de gerar PDF sem dados**
* **Descrição:** Verificar comportamento quando não há dados para o relatório.
* **Pré-requisito:** Aplicar filtros que não retornem resultados.
* **Passos/Ação:** Acessar a tela "Relatórios"
  + Selecionar tipo de relatório desejado
  + Configurar filtros sem resultados
  + Clicar em "Gerar PDF"
* **Resultado Esperado:** O sistema exibe mensagem: "Não há dados disponíveis para gerar o relatório."
  + Nenhum PDF é gerado.

#### 

### 2.15.5 Requisitos Não Funcionais

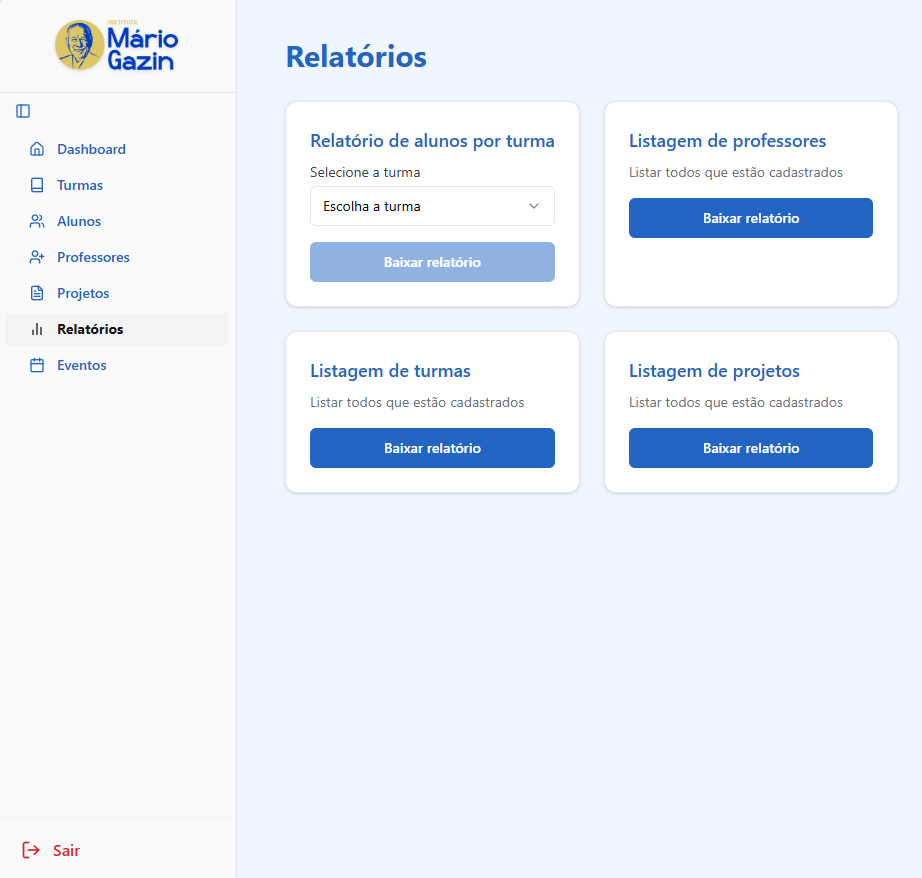
* Compatibilidade: PDFs gerados devem abrir em qualquer visualizador padrão.
* Performance: Geração de relatório em até 5 segundos.
* Segurança: Apenas usuários autenticados com permissão podem realizar o download.
* Usabilidade: Interface simples com opções de filtro e botão direto de exportação.

## 

## 

## 

## 2.16 User Story – Relatórios de Alunos por Turma



**Como** usuário do sistema,   
**Eu quero** gerar relatório de alunos organizados por turma,   
**Para que** eu possa ter visão completa da distribuição de alunos nas diferentes turmas do Instituto.

### 2.16.1 Critérios de Aceitação

* O sistema deve permitir selecionar turma específica ou todas as turmas.
* O relatório deve agrupar alunos por turma mostrando nome da turma, projeto e professor.
* Para cada aluno deve exibir: nome, idade, telefone de contato e data de matrícula.
* O sistema deve mostrar total de alunos por turma e total geral.
* O relatório deve permitir ordenação por nome do aluno ou data de matrícula.

### 2.16.2 Regras de Negócio

* Apenas alunos com status ativo devem aparecer no relatório.
* Turmas sem alunos matriculados devem aparecer com indicação "Nenhum aluno matriculado".
* O relatório deve ser ordenado alfabeticamente por nome da turma.

### 2.16.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Geração de relatório de turma específica**
* **Descrição:** Verificar se o sistema gera relatório para uma turma selecionada.
* **Pré-requisito:** Ter turmas com alunos matriculados no sistema.
* **Passos/Ação:** Acessar "Relatórios"
  + Selecionar "Relatório de alunos por turma"
  + Escolher uma turma específica no filtro
  + Clicar em "Buscar relatório"
* **Resultado Esperado:** São exibidos apenas os alunos da turma selecionada.
  + Relatório mostra cabeçalho da turma com projeto e professor.
  + Lista todos os alunos com informações completas.
* **CT02: Geração de relatório de todas as turmas**
* **Descrição:** Verificar se o sistema gera relatório completo de todas as turmas.
* **Pré-requisito:** Ter múltiplas turmas cadastradas com alunos.
* **Passos/Ação:** Acessar "Relatórios"
  + Selecionar "Relatório de alunos por turma"
  + Manter filtro "Todas as turmas"
  + Clicar em "Buscar relatório"
* **Resultado Esperado:** São exibidas todas as turmas organizadas alfabeticamente.
  + Cada turma mostra seus alunos matriculados.
  + Totais por turma e geral são exibidos.

### 2.16.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Relatório de turma sem alunos**
* **Descrição:** Verificar comportamento quando turma não possui alunos matriculados.
* **Pré-requisito:** Ter turma cadastrada sem alunos matriculados.
* **Passos/Ação:** Acessar "Relatórios"
  + Selecionar "Relatório de alunos por turma"
  + Selecionar turma sem alunos
  + Gerar relatório
* **Resultado Esperado:** O sistema exibe a turma com indicação "Nenhum aluno matriculado".
  + Contador mostra 0 alunos para a turma.

### 2.16.5 Requisitos Não Funcionais

* **Desempenho:** A geração de relatórios deve ser processada em até 5 segundos para bases com até 1000 registros.
* **Segurança:** Apenas usuários autenticados com permissão podem gerar relatórios.
* **Escalabilidade:** O sistema deve gerar relatórios eficientemente mesmo com grandes volumes de dados.
* **Manutenibilidade:** O código de geração de relatórios deve ser modular e facilmente extensível.
* **Compatibilidade:** Os relatórios devem ser compatíveis com diferentes navegadores e dispositivos.
* **Usabilidade:** A interface de relatórios deve ser intuitiva e permitir fácil navegação entre diferentes tipos de relatório.

## 2.17 User Story – Listagem de Professores

**Como** usuário do sistema,   
**Eu quero** gerar relatório com listagem completa dos professores,   
**Para que** eu possa ter visão geral de todo corpo docente do Instituto.

### 2.17.1 Critérios de Aceitação

* O sistema deve listar todos os professores cadastrados no sistema.
* Para cada professor deve exibir: nome, telefone, email e turmas que leciona.
* O relatório deve permitir filtrar por professores ativos/inativos.
* O sistema deve mostrar total de professores e total de turmas por professor.
* A listagem deve ser ordenada alfabeticamente por nome.

### 2.17.2 Regras de Negócio

* Professores inativos devem aparecer apenas quando filtro específico for aplicado.
* Se professor não leciona em nenhuma turma, deve aparecer "Nenhuma turma atribuída".

### 2.17.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Listagem completa de professores ativos**
* **Descrição:** Verificar se o sistema lista todos os professores ativos.
* **Pré-requisito:** Ter professores cadastrados no sistema.
* **Passos/Ação:** Acessar "Relatórios"
  + Selecionar "Listagem de professores"
  + Manter filtro "Professores ativos"
  + Clicar em "Buscar relatório"
* **Resultado Esperado:** São exibidos todos os professores com status ativo.
  + **Cada professor mostra informações completas e turmas que leciona.**
* **CT02: Filtro por professores inativos**
* **Descrição:** Verificar se filtro de professores inativos funciona corretamente.
* **Pré-requisito:** Ter professores com status inativo no sistema.
* **Passos/Ação:** Selecionar filtro "Professores inativos"
  + Gerar relatório
* **Resultado Esperado:** São exibidos apenas professores com status inativo.

### 2.17.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Relatório sem professores no filtro**
* **Descrição:** Verificar comportamento quando filtro não retorna professores.
* **Pré-requisito:** Aplicar filtro que não possui dados.
* **Passos/Ação:** Aplicar filtro específico sem dados
  + Tentar gerar relatório
* **Resultado Esperado:** O sistema exibe "Nenhum professor encontrado com os filtros selecionados."

### 2.17.5 Requisitos Não Funcionais

* **Desempenho:** A geração de relatórios deve ser processada em até 5 segundos para bases com até 1000 registros.
* **Segurança:** Apenas usuários autenticados com permissão podem gerar relatórios.
* **Escalabilidade:** O sistema deve gerar relatórios eficientemente mesmo com grandes volumes de dados.
* **Manutenibilidade:** O código de geração de relatórios deve ser modular e facilmente extensível.
* **Compatibilidade:** Os relatórios devem ser compatíveis com diferentes navegadores e dispositivos.
* **Usabilidade:** A interface de relatórios deve ser intuitiva e permitir fácil navegação entre diferentes tipos de relatório.

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 2.18 User Story – Listagem de Turmas

## 

**Como** usuário do sistema,   
**Eu quero** gerar relatório com listagem completa das turmas,   
**Para que** eu possa ter visão geral de todas as atividades oferecidas pelo Instituto.

### 2.18.1 Critérios de Aceitação

* O sistema deve listar todas as turmas cadastradas no sistema.
* Para cada turma deve exibir: nome, projeto, professor, horário e número de alunos matriculados.
* O sistema deve mostrar total de turmas e total geral de alunos.
* A listagem deve ser agrupada por projeto.

### 2.18.2 Regras de Negócio

* Apenas turmas ativas devem aparecer no relatório padrão.
* Turmas devem ser agrupadas por projeto e ordenadas alfabeticamente.

### 

### 2.18.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Listagem completa de turmas**
* **Descrição:** Verificar se o sistema lista todas as turmas ativas.
* **Pré-requisito:** Ter turmas cadastradas em diferentes projetos.
* **Passos/Ação:** Acessar "Relatórios"
  + Selecionar "Listagem de turmas"
  + Clicar em "Baixar relatório"
* **Resultado Esperado:** São exibidas todas as turmas agrupadas por projeto.
  + Cada turma mostra informações completas e número de alunos.

### 2.18.4 Requisitos Não Funcionais

* **Desempenho:** A geração de relatórios deve ser processada em até 5 segundos para bases com até 1000 registros.
* **Segurança:** Apenas usuários autenticados com permissão podem gerar relatórios.
* **Escalabilidade:** O sistema deve gerar relatórios eficientemente mesmo com grandes volumes de dados.
* **Manutenibilidade:** O código de geração de relatórios deve ser modular e facilmente extensível.
* **Compatibilidade:** Os relatórios devem ser compatíveis com diferentes navegadores e dispositivos.
* **Usabilidade:** A interface de relatórios deve ser intuitiva e permitir fácil navegação entre diferentes tipos de relatório.

## 2.19 User Story – Listagem de Projetos

**Como** usuário do sistema,   
**Eu quero** gerar relatório com listagem completa dos projetos,   
**Para que** eu possa ter visão geral de todas as iniciativas do Instituto.

### 2.19.1 Critérios de Aceitação

* O sistema deve listar todos os projetos cadastrados no sistema.
* Para cada projeto deve exibir: nome, descrição, número de turmas e total de alunos.
* O relatório deve mostrar status do projeto (ativo/inativo).
* O sistema deve calcular totais gerais de projetos, turmas e alunos.

### 2.19.2 Regras de Negócio

* Projetos inativos devem aparecer com indicação visual diferenciada.
* Projetos sem turmas devem aparecer com indicação "Nenhuma turma cadastrada".

### 2.19.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Listagem completa de projetos**
* **Descrição:** Verificar se o sistema lista todos os projetos cadastrados.
* **Pré-requisito:** Ter projetos cadastrados no sistema.
* **Passos/Ação:** Acessar "Relatórios"
  + Selecionar "Listagem de projetos"
  + Clicar em "Buscar relatório"
* **Resultado Esperado:** São exibidos todos os projetos com informações completas.
  + **Totais de turmas e alunos são calculados corretamente.**

### 2.19.4 Teste de Aceitação - Caso de Teste Negativo

* **CT02: Projeto sem turmas cadastradas**
* **Descrição:** Verificar exibição de projeto sem turmas.
* **Pré-requisito:** Ter projeto cadastrado sem turmas.
* **Passos/Ação:** Gerar relatório de projetos
* **Resultado Esperado:** Projeto aparece com indicação "Nenhuma turma cadastrada".
  + **Contador de alunos mostra 0.**

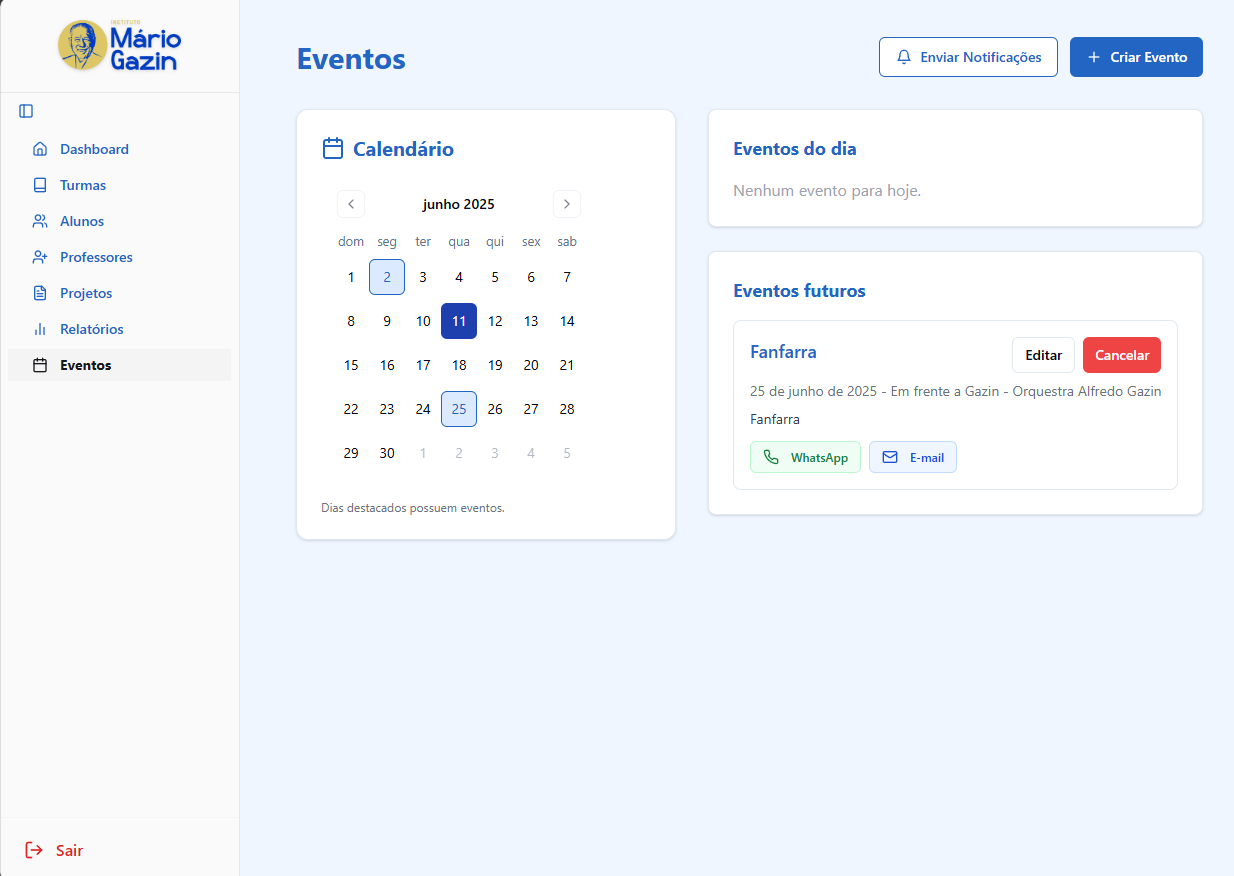
### 2.19.5 Requisitos Não Funcionais

* **Desempenho:** A geração de relatórios deve ser processada em até 5 segundos para bases com até 1000 registros.
* **Segurança:** Apenas usuários autenticados com permissão podem gerar relatórios.
* **Escalabilidade:** O sistema deve gerar relatórios eficientemente mesmo com grandes volumes de dados.
* **Manutenibilidade:** O código de geração de relatórios deve ser modular e facilmente extensível.
* **Compatibilidade:** Os relatórios devem ser compatíveis com diferentes navegadores e dispositivos.
* **Usabilidade:** A interface de relatórios deve ser intuitiva e permitir fácil navegação entre diferentes tipos de relatório.

## 

## 

## 2.20 User Story – Visualização do Calendário de Eventos



**Como** usuário do sistema,   
**Eu quero** visualizar eventos em formato de calendário,   
**Para que** eu possa ter uma visão cronológica das atividades do Instituto.

### 2.20.1 Critérios de Aceitação

* O sistema deve exibir calendário mensal com eventos marcados.
* O usuário deve poder navegar entre meses (anterior/próximo).
* Ao clicar em uma data, devem ser exibidos os eventos daquele dia.
* Eventos devem ter cores diferentes conforme a categoria.
* O sistema deve destacar a data atual.
* Deve haver visualização de eventos do dia atual e eventos futuros.

### 2.20.2 Regras de Negócio

* Apenas eventos com status ativo devem aparecer no calendário.
* Eventos passados devem aparecer com indicação visual diferenciada.
* Máximo de 3 eventos por dia devem ser exibidos diretamente no calendário.

### 2.20.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Visualização do calendário mensal**
* **Descrição:** Verificar se o calendário exibe corretamente os eventos do mês.
* **Pré-requisito:** Ter eventos cadastrados no mês atual.
* **Passos/Ação:** Acessar a tela "Eventos"
  + Visualizar calendário padrão
* **Resultado Esperado:** O calendário mostra o mês atual com eventos marcados.
  + Data atual está destacada.
  + Eventos aparecem nas datas corretas.
* **CT02: Navegação entre meses**
* **Descrição:** Verificar se é possível navegar entre diferentes meses.
* **Pré-requisito:** Estar visualizando o calendário.
* **Passos/Ação:** Acessar a tela "Eventos"
  + Visualizar calendário padrão
  + Clicar na seta "próximo mês"
  + Verificar mudança
  + Clicar na seta "mês anterior"
* **Resultado Esperado:** O calendário navega corretamente entre os meses.
  + Eventos são atualizados conforme o mês selecionado.
* **CT03: Visualização de eventos do dia**
* **Descrição:** Verificar se ao clicar em uma data os eventos são exibidos.
* **Pré-requisito:** Ter eventos cadastrados em uma data específica.
* **Passos/Ação:** Acessar a tela "Eventos"
  + Visualizar calendário padrão
  + Clicar em uma data com eventos
* **Resultado Esperado:** São exibidos todos os eventos da data selecionada.
  + Informações completas de cada evento são mostradas.

### 2.20.4 Teste de Aceitação - Caso de Teste Negativo

* **CT04: Visualização de dia sem eventos**
* **Descrição:** Verificar comportamento ao clicar em data sem eventos.
* **Pré-requisito:** Selecionar data sem eventos cadastrados.
* **Passos/Ação:** Acessar a tela "Eventos"
  + Visualizar calendário padrão
  + Clicar em data vazia no calendário
* **Resultado Esperado:** O sistema exibe "Nenhum evento para hoje."

### 2.20.5 Requisitos Não Funcionais

* **Desempenho:** O calendário deve carregar e exibir eventos em até 2 segundos para meses com até 100 eventos cadastrados.
* **Segurança:** Apenas usuários autenticados podem visualizar eventos no calendário.
* **Escalabilidade:** O sistema deve manter performance adequada mesmo com grandes volumes de eventos, utilizando paginação ou carregamento sob demanda quando necessário.
* **Manutenibilidade:** O componente de calendário deve ser modular e permitir fácil adição de novas funcionalidades como diferentes tipos de visualização (semanal, anual).
* **Compatibilidade:** O calendário deve funcionar corretamente nos principais navegadores e dispositivos móveis.
* **Usabilidade:** A interface de eventos deve ser clara, intuitiva e validada com os usuários finais.

## 2.21 User Story – Cadastro de Eventos

**Como** usuário do sistema,   
**Eu quero** poder cadastrar novos eventos no sistema,   
**Para que** eu possa organizar e divulgar as atividades do Instituto Mário Gazin.

### 2.21.1 Critérios de Aceitação

* O sistema deve permitir inserir título do evento obrigatoriamente.
* O sistema deve permitir inserir descrição detalhada do evento.
* O sistema deve permitir definir data e horário do evento.
* O sistema deve permitir selecionar local do evento.
* Após o cadastro, o evento deve aparecer no calendário.

### 2.21.2 Regras de Negócio

* O título do evento é obrigatório e deve ter pelo menos 5 caracteres.
* A data do evento não pode ser anterior à data atual.
* Eventos no mesmo horário e local devem gerar alerta de conflito.

### 2.21.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Cadastro de evento realizado com sucesso**
* **Descrição:** Verificar se o sistema cadastra corretamente um novo evento.
* **Pré-requisito:** Ter acesso ao formulário de cadastro de eventos.
* **Passos/Ação:** Acessar a tela "Eventos"
  + Clicar no botão "Criar Evento"
  + Preencher os campos obrigatórios
  + Clicar em "Salvar"
* **Resultado Esperado:** É exibida a mensagem "Evento criado com sucesso!"
  + O evento aparece no calendário na data selecionada.

### 2.21.4 Teste de Aceitação - Caso de Teste Negativo

* **CT02: Cadastro com data passada**
* **Descrição:** Verificar se o sistema impede cadastro de evento com data anterior à atual.
* **Pré-requisito:** Estar no formulário de cadastro de eventos.
* **Passos/Ação:** Acessar a tela "Eventos"
  + Clicar no botão "Criar Evento"
  + Preencher dados do evento
  + Selecionar data anterior à atual
  + Tentar salvar evento
* **Resultado Esperado:** O sistema exibe mensagem: "A data do evento não pode ser anterior à data atual."
  + Nenhum cadastro é realizado.
* **CT03: Cadastro sem preencher dados obrigatórios**
* **Descrição:** Verificar se o sistema impede cadastro com dados sem preencher.
* **Pré-requisito:** Estar no formulário de cadastro.
* **Passos/Ação:** Acessar a tela "Eventos"
  + Clicar no botão "Criar Evento"
  + Deixar campos em branco
  + Tentar salvar
* **Resultado Esperado:** O sistema exibe mensagem de campo obrigatório.
  + Nenhum cadastro é realizado.

### 2.21.5 Requisitos Não Funcionais

* **Desempenho:** Processamento de cadastro em até 3 segundos.
* **Segurança:** Apenas usuários autenticados com permissão podem realizar o cadastro.
* **Escalabilidade:** O sistema deve permitir o cadastro de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de cadastro deve ser clara, intuitiva e validada com os usuários finais.

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

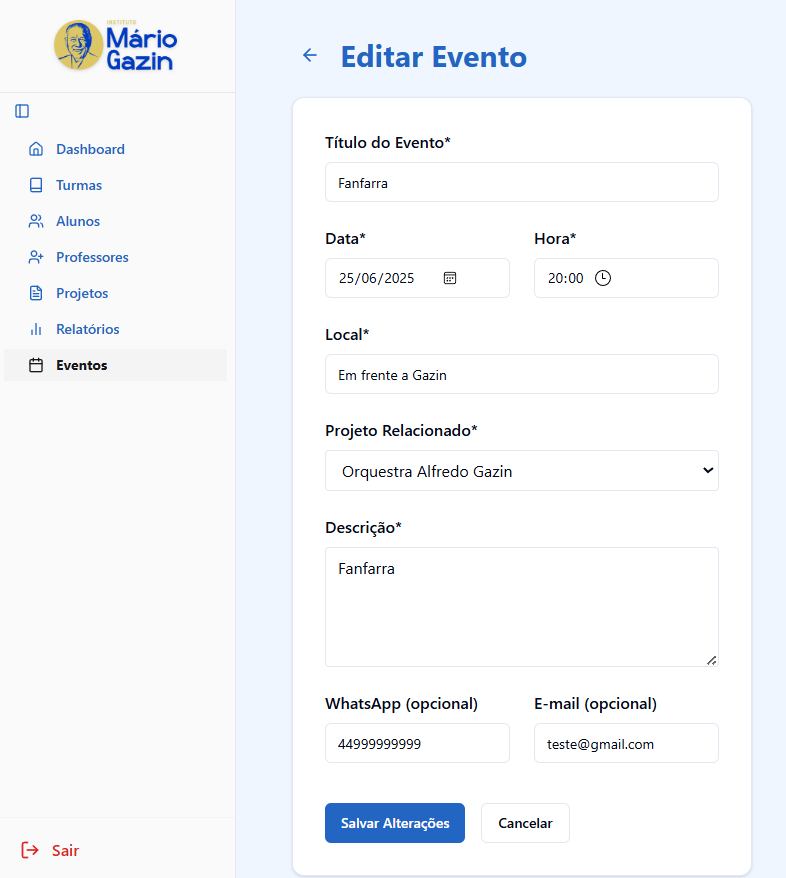
## 

## 

## 

## 

## 2.22 User Story – Edição de Eventos



**Como** usuário do sistema,   
**Eu quero** poder editar informações dos eventos cadastrados,   
**Para que** eu possa manter os dados atualizados e corrigir informações quando necessário.

### 2.22.1 Critérios de Aceitação

* O sistema deve permitir alterar título, descrição, data, horário e local.
* O sistema deve validar se nova data não é anterior à atual.
* O sistema deve verificar conflitos de horário e local.
* Após a edição, as alterações devem ser refletidas no calendário.

### 2.22.2 Regras de Negócio

* Eventos que já ocorreram só podem ter descrição editada.
* Alterações de data devem gerar notificação aos participantes.
* Não é possível alterar evento para data passada.

### 2.22.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Edição de evento futuro**
* **Descrição:** Verificar se o sistema permite editar evento que ainda não ocorreu.
* **Pré-requisito:** Ter evento futuro cadastrado no sistema.
* **Passos/Ação:** Acessar detalhes do evento
  + Clicar em "Editar"
  + Alterar informações desejadas
  + Clicar em "Salvar"
* **Resultado Esperado:** É exibida a mensagem "Evento atualizado com sucesso!"
  + As alterações são refletidas no calendário.
* **CT02: Edição de descrição de evento passado**
* **Descrição:** Verificar se é possível editar apenas descrição de evento já ocorrido.
* **Pré-requisito:** Ter evento com data passada no sistema.
* **Passos/Ação:** Acessar evento passado
  + Clicar em "Editar"
  + Tentar editar descrição
  + Salvar alterações
* **Resultado Esperado:** Apenas a descrição é editável.
  + Demais campos ficam bloqueados.

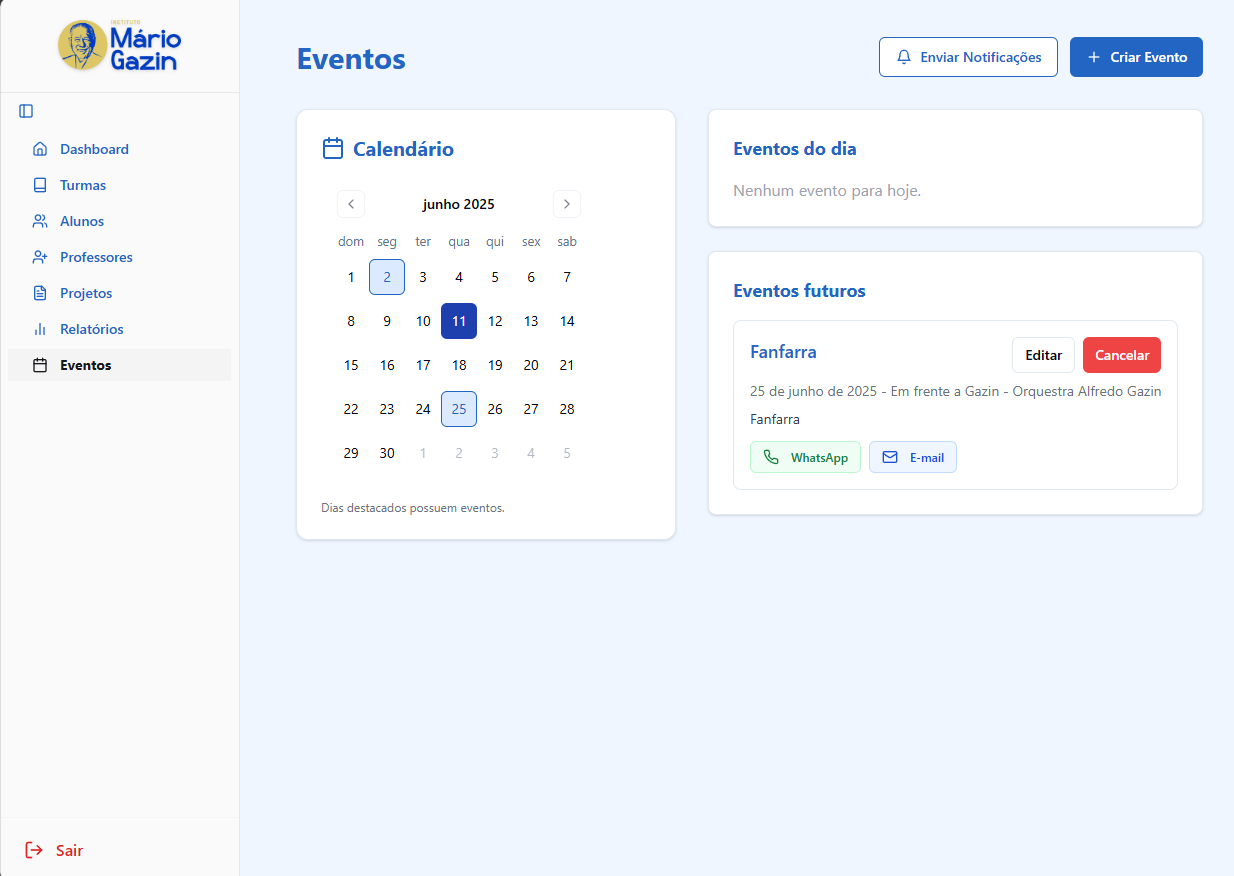
### 2.22.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Tentativa de alterar data para passada**
* **Descrição:** Verificar se o sistema impede alterar evento para data passada.
* **Pré-requisito:** Ter evento futuro cadastrado no sistema.
* **Passos/Ação:** Acessar detalhes do evento
  + Clicar em "Editar"
  + Alterar data para anterior à atual
  + Tentar salvar
* **Resultado Esperado:** O sistema exibe mensagem: "A data do evento não pode ser anterior à data atual."
  + Nenhuma alteração é salva.

### 2.22.5 Requisitos Não Funcionais

* **Desempenho:** Processamento de cadastro em até 3 segundos.
* **Segurança:** Apenas usuários autenticados com permissão podem editar o cadastro.
* **Escalabilidade:** O sistema deve permitir a edição de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de edição deve ser clara, intuitiva e validada com os usuários finais.

## 2.23 User Story – Exclusão de Eventos



**Como** usuário do sistema,   
**Eu quero** poder excluir eventos do sistema,   
**Para que** eu possa remover atividades canceladas ou registros desnecessários.

### 2.23.1 Critérios de Aceitação

* O sistema deve pedir confirmação antes de excluir.
* Ao excluir, o evento deve ser removido do calendário.

### 2.23.2 Regras de Negócio

* Eventos já realizados não podem ser excluídos.

### 2.23.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Exclusão de evento**
* **Descrição:** Verificar se o sistema exclui eventos agendados.
* **Pré-requisito:** Ter evento futuro cadastrado.
* **Passos/Ação:** Acessar detalhes do evento
  + Clicar em "Excluir"
  + Confirmar exclusão
* **Resultado Esperado:** É exibida a mensagem "Evento excluído com sucesso!"
  + O evento é removido do calendário.
* **CT02: Cancelamento da exclusão**
* **Descrição:** Verificar se ao cancelar a exclusão o evento permanece.
* **Pré-requisito:** Ter evento cadastrado.
* **Passos/Ação:** Acessar detalhes do evento
  + Clicar em "Excluir"
  + Clicar em "Cancelar" na confirmação
* **Resultado Esperado:** Nenhuma ação é aplicada.
  + O evento continua no calendário.

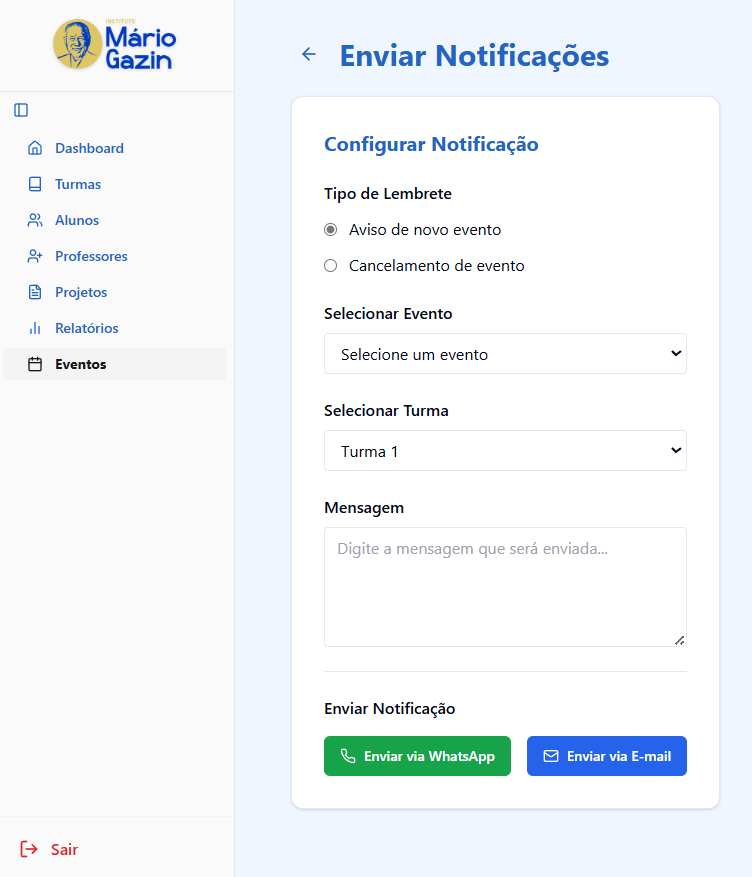
### 2.23.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Tentativa de exclusão de evento realizado**
* **Descrição:** Verificar se o sistema impede exclusão de evento já ocorrido.
* **Pré-requisito:** Ter evento com data passada.
* **Passos/Ação:** Acessar detalhes do evento
  + Tentar excluir evento passado
* **Resultado Esperado:** O sistema exibe mensagem: "Eventos já realizados não podem ser excluídos."

### 2.23.5 Requisitos Não Funcionais

* **Desempenho:** A exclusão dos dados deve ser processada em até 2 segundos após a confirmação.
* **Segurança:** Apenas usuários autenticados com permissão podem realizar a exclusão dos professores.
* **Escalabilidade:** O sistema deve permitir a exclusão de dados em uma base com mais de mil alunos sem perda de desempenho.
* **Manutenibilidade:** O código deve ser versionado e documentado no GitHub, facilitando futuras manutenções.
* **Compatibilidade:** A funcionalidade deve ser totalmente compatível com dispositivos desktop e mobile.
* **Usabilidade:** A interface de exclusão deve ser clara, intuitiva e validada com os usuários finais.

## 2.24 User Story – Envio de Notificações



**Como** usuário do sistema,   
**Eu quero** enviar notificações sobre eventos,   
**Para que** eu possa comunicar efetivamente com alunos, professores e responsáveis sobre as atividades do Instituto.

### 2.24.1 Critérios de Aceitação

* O sistema deve permitir selecionar tipo de lembrete (Aviso de novo evento ou Cancelamento de evento).
* O sistema deve permitir selecionar evento específico da lista de eventos cadastrados.
* O sistema deve permitir selecionar turma específica para envio direcionado.
* O sistema deve permitir personalizar mensagem da notificação.
* O sistema deve oferecer opções de envio via WhatsApp e E-mail.
* O sistema deve mostrar prévia da notificação antes do envio.
* O sistema deve registrar histórico de notificações enviadas.
* O sistema deve permitir reagendar envio de notificações.

### 2.24.2 Regras de Negócio

* Notificações devem ser enviadas apenas para usuários ativos.
* Deve haver limite de notificações por dia para evitar spam.
* Responsáveis devem receber notificações de eventos de seus filhos.
* Notificações de cancelamento só podem ser enviadas para eventos já cadastrados.
* Deve ser obrigatório selecionar um evento antes de enviar a notificação.
* Deve ser obrigatório selecionar uma turma antes de enviar a notificação.

### 2.24.3 Teste de Aceitação - Caso de Teste Positivo

* **CT01: Envio de notificação de novo evento via WhatsApp ou E-mail**
* **Descrição:** Verificar se o sistema envia notificação de novo evento via WhatsApp ou E-mail para turma selecionada.
* **Pré-requisito:** Ter evento cadastrado, turma cadastrada e usuários ativos.
* **Passos/Ação:**
  + Acessar "Enviar Notificações"
  + Selecionar "Aviso de novo evento"
  + Selecionar evento da lista
  + Selecionar turma específica
  + Personalizar mensagem
  + Clicar em "Enviar via WhatsApp" ou “Enviar via E-mail".
* **Resultado Esperado:** É exibida a mensagem "Notificações enviadas com sucesso!"
  + Registro é criado no histórico.
* **CT02: Envio de notificação de cancelamento de evento via WhatsApp ou E-mail**
* **Descrição:** Verificar se o sistema envia notificação de cancelamento via WhatsApp ou E-mail para turma selecionada.
* **Pré-requisito:** Ter evento cadastrado, turma cadastrada e usuários ativos.
* **Passos/Ação:** Acessar "Enviar Notificações"
  + Selecionar "Cancelamento de evento"
  + Selecionar evento da lista
  + Selecionar turma específica
  + Personalizar mensagem de cancelamento
  + Clicar em "Enviar via WhatsApp" ou “Enviar via E-mail".
* **Resultado Esperado:** É exibida a mensagem "Notificações de cancelamento enviadas com sucesso!"
  + Registro é criado no histórico com status "Cancelamento".

### 2.24.4 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Tentativa de envio de novo evento sem selecionar evento**
* **Descrição:** Verificar comportamento quando tipo "Aviso de novo evento" é selecionado mas nenhum evento é escolhido.
* **Pré-requisito:** Estar na tela de configuração de notificações.
* **Passos/Ação:** Acessar "Enviar Notificações"
  + Selecionar "Aviso de novo evento"
  + Deixar campo "Selecione um evento" sem seleção
  + Tentar clicar em "Enviar via WhatsApp" ou "Enviar via E-mail"
* **Resultado Esperado:** O sistema exibe mensagem: "Selecione um evento antes de enviar a notificação."
  + Nenhuma notificação é enviada.
* **CT04: Tentativa de envio de cancelamento sem selecionar evento**
* **Descrição:** Verificar comportamento quando tipo "Cancelamento de evento" é selecionado mas nenhum evento é escolhido.
* **Pré-requisito:** Estar na tela de configuração de notificações.
* **Passos/Ação:** Acessar "Enviar Notificações"
  + Selecionar "Cancelamento de evento"
  + Deixar campo "Selecione um evento" sem seleção
  + Tentar clicar em "Enviar via WhatsApp" ou "Enviar via E-mail"
* **Resultado Esperado:** O sistema exibe mensagem: "Selecione um evento para cancelar antes de enviar a notificação."
  + Nenhuma notificação é enviada.
* **CT05: Tentativa de envio sem selecionar turma**
* **Descrição:** Verificar comportamento quando nenhuma turma é selecionada.
* **Pré-requisito:** Estar na tela de envio de notificações com evento já selecionado.
* **Passos/Ação:** Acessar "Enviar Notificações"
  + Selecionar tipo de lembrete e evento
  + Deixar campo "Turma 1" sem seleção
  + Tentar enviar notificação
* **Resultado Esperado:** O sistema exibe mensagem: "Selecione uma turma antes de enviar a notificação."
  + Nenhuma notificação é enviada.
* **CT06: Tentativa de envio com mensagem vazia**
* **Descrição:** Verificar comportamento quando campo de mensagem está vazio.
* **Pré-requisito:** Estar na tela de envio de notificações.
* **Passos/Ação:** Acessar "Enviar Notificações"
  + Configurar tipo, evento e turma
  + Deixar campo "Mensagem" vazio
  + Tentar enviar
* **Resultado Esperado:** O sistema exibe mensagem: "Digite uma mensagem antes de enviar a notificação."
  + Nenhuma notificação é enviada.

## 2.25 User Story – Botão Sair

### 

**Como** usuário do sistema,   
**Eu quero** poder sair da minha conta com segurança,  
**Para que** minhas informações fiquem protegidas e o acesso ao sistema seja encerrado corretamente.

### 2.25.1 Critérios de Aceitação

* O sistema deve encerrar a sessão ao clicar no botão "Sair".
* Após sair, o usuário deve ser redirecionado para a tela de login.
* Não deve ser possível acessar áreas internas sem autenticação.

### 2.25.2 Regras de Negócio

* O botão "Sair" só deve estar disponível para usuários autenticados.
* A sessão do usuário deve ser totalmente encerrada ao sair.
* As rotas protegidas devem exigir nova autenticação após logout.

### 2.25.3 Teste de Aceitação - Caso de Teste Positivo

### **CT0**1**: Logout realizado com sucesso**

* **Descrição:** Verificar se o sistema encerra corretamente a sessão do usuário.
* **Pré-requisito:** Usuário deve estar autenticado no sistema.
* **Passos/Ação:**
  + Acessar qualquer tela do sistema após login
  + Clicar no botão "Sair**”**
* **Resultado Esperado:**
  + Usuário é redirecionado para a tela de login
  + Sessão é encerrada e acesso restrito sem nova autenticação
* **CT02: Redirecionamento após logout**
* **Descrição:** Verificar se o sistema impede acesso a páginas internas após logout.
* **Pré-requisito:** Sessão encerrada (usuário clicou em "Sair")
* **Passos/Ação:**
  + Clicar no botão "Voltar" do navegador
  + Tentar acessar uma URL protegida
* **Resultado Esperado:**
  + Sistema redireciona automaticamente para a tela de login
  + Nenhum conteúdo interno é exibido

### 2.25.3 Teste de Aceitação - Caso de Teste Negativo

* **CT03: Tentativa de acessar o sistema após logout sem nova autenticação**
* **Descrição:** Verificar se o sistema bloqueia o acesso sem autenticação válida.
* **Pré-requisito:** Sessão encerrada (logout realizado)  
  **Passos/Ação:**
  + Abrir nova aba
  + Tentar acessar uma URL restrita diretamente
* **Resultado Esperado:**
  + Sistema exibe a tela de login
  + Mensagem de acesso negado, se aplicável

### 

### 2.25.5 Requisitos Não Funcionais

* **Desempenho:** O logout deve ocorrer e redirecionar o usuário em até 2 segundos.
* **Segurança:** A sessão deve ser completamente encerrada e os tokens removidos.
* **Escalabilidade:** O sistema deve suportar múltiplas sessões e logouts simultâneos sem impacto.
* **Manutenibilidade:** A lógica de logout deve estar documentada e versionada em repositório (ex: GitHub).
* **Compatibilidade:** A funcionalidade deve funcionar corretamente em desktop e dispositivos móveis.
* **Usabilidade:** O botão "Sair" deve estar claramente visível e acessível na interface do usuário.

### 

# ANEXOS

* Link do repositorio Git Hub : <https://github.com/Mateuscheki/Projeto-extensao>
* .pdf Sebrae Canvas Acesso ao site:

<https://canvas-apps.pr.sebrae.com.br/canvas?id=1510848>

* .pdf das telas de Wireframes e Protótipos