

ESPECIFICAÇÃO DE REQUISITOS NÃO FUNCIONAIS

Discentes: BRUNO EDUARDO DOS SANTOS - 211066249
GEILSON DOS SANTOS SA - 231006239
MARCOS ALEXANDRE DA SILVA NERES - 211055334
PEDRO FARIAS DE OLIVEIRA - 211055577
THELMA EVANGELISTA DOS SANTOS - 231003513
WESLEY HENRIQUE FERREIRA - 231021496

Sistema de Gestão de Feiras	
Requisitos não funcionais	Data: 16/06/2025

1. Introdução

Este documento especifica os requisitos não funcionais e de suporte de um sistema de software voltado à organização, divulgação e gerenciamento de eventos do tipo feira. O sistema será responsável por viabilizar a criação, leitura, atualização e exclusão (CRUD) de registros relacionados a feiras, expositores, produtos e ingressos. Seu objetivo é facilitar a interação entre organizadores de feiras e visitantes, permitindo a exibição estruturada de informações sobre os eventos e seus participantes.

O sistema contemplará diferentes níveis de acesso: visitantes poderão consultar dados públicos, como lista de feiras, expositores e produtos; usuários autenticados, por sua vez, terão acesso a funcionalidades administrativas, como a criação e edição de registros, além da emissão de ingressos. Assim, busca-se oferecer uma solução completa e eficiente para a gestão de feiras, promovendo maior visibilidade dos eventos e melhor organização das informações envolvidas.

2. Requisitos funcionais de todo o sistema

- O sistema deve validar permissões de cada usuário para ações como criação, edição e exclusão de registros, conforme seu perfil (administrador, expositor, visitante).
- O sistema deve restringir a exclusão de registros vinculados a outras entidades, garantindo a integridade dos dados.
- O sistema deve permitir a emissão e leitura de ingressos válidos por data e evento, impedindo reutilizações fora do prazo.
- O sistema deve registrar ações administrativas relevantes (criação, edição, exclusão de feiras, usuários e expositores), incluindo data, hora e autor da ação.

- O sistema deve permitir a geração de relatórios em PDF ou CSV contendo informações consolidadas de feiras, expositores e visitas.
- O sistema deve notificar usuários por e-mail em eventos importantes, como aprovações de inscrição e alterações de eventos.
- O sistema deve aplicar validações sistêmicas para impedir alterações que comprometam a consistência entre módulos e dados relacionados.
- O sistema deve permitir agendamento de relatórios periódicos, com envio por e-mail aos administradores.
- O sistema deve impedir a criação de registros duplicados com base em campos-chave (ex: CPF, e-mail, CNPJ, nome de feira).
- O sistema deve invalidar automaticamente o token de autenticação após logout ou expiração da sessão.
- O sistema deve permitir personalização da frequência de notificações por parte do usuário.

3. Qualidades do Sistema

3.1 Usabilidade

ID do requisito	Descrição	Critério de Aceitação/Avaliação
USAB-01	A interface do sistema deverá oferecer uma experiência adaptada aos diferentes perfis de usuário (administrador, expositor e visitante), priorizando a simplicidade para visitantes e ferramentas completas para administradores e expositores.	Testes com usuários reais representando cada perfil, medindo efetividade, eficiência e satisfação.
USAB-02	Padrões de nomenclatura, layout e interação devem ser consistentes.	Inspeção heurística com base nas 10 heurísticas de Nielsen.
USAB-03	O sistema deve permitir execução de tarefas típicas com o mínimo de instrução.	Execução de cenários como “localizar expositor” com medição de cliques e tempo de execução.
USAB-04	Deve haver suporte a testes remotos de usabilidade.	Uso de ferramentas como Maze, UsabilityHub ou Testable com análise de sessões e cliques.
USAB-05	O sistema deve oferecer suporte ao idioma português brasileiro como padrão.	Verificação de uso de arquivos externos .json/.po para textos, separando lógica e conteúdo.
USAB-06	O sistema deve exibir datas, moedas e	Testes em telas de entrada e

6	formatos numéricos conforme as convenções pt-BR.	saída de dados, validando formatos R\$, dd/mm/aaaa, etc.
---	--	--

3.2 Confiabilidade

ID do requisito	Descrição	Critério de Aceitação/Avaliação
REL-01	O sistema deve permitir recuperação de falhas com perda máxima de 5 minutos de dados não persistidos.	Simulações controladas de falhas serão realizadas (ex: desligamento súbito do servidor), com posterior verificação da integridade dos dados restaurados a partir do último backup. O intervalo de perda será validado por meio de análise de logs e timestamp dos dados persistentes.
REL-02	Devem existir mecanismos de backup automáticos diários, com armazenamento seguro e possibilidade de restauração manual ou automatizada.	Verificação técnica dos scripts/agendamentos de backup (cron jobs ou ferramentas específicas). Serão analisados os arquivos gerados, sua periodicidade e a possibilidade de restauração dos dados em ambiente de teste.
REL-03	O sistema deve implementar autenticação segura, utilizando algoritmos de hash robustos para senhas, como bcrypt.	A análise de segurança será feita por meio de revisão de código-fonte para verificar a utilização de algoritmos de hash como bcrypt ou Argon2. Ferramentas como SonarQube ou OWASP Dependency-Check

		podem ser utilizadas para validação automatizada.
REL-04	As sessões de usuários devem expirar automaticamente após 30 minutos de inatividade, como medida de segurança e estabilidade.	Serão conduzidos testes funcionais simulando usuários ociosos. Após 30 minutos de inatividade, será verificado o redirecionamento automático para a tela de login, invalidando a sessão ativa.
REL-05	Deve ser implementado um mecanismo de bloqueio temporário para usuários que realizarem múltiplas tentativas de login inválido consecutivas.	Serão realizados testes de tentativa de login com credenciais inválidas consecutivas. Após um número pré-determinado de falhas (por exemplo, 5 tentativas), o sistema deverá bloquear novas tentativas por um intervalo de tempo definido. A resposta do sistema, o tempo de bloqueio e as mensagens exibidas serão analisados.
REL-06	O sistema deve implementar controle de acesso baseado em perfis de usuário (administrador, expositor e visitante), garantindo que cada perfil tenha permissões específicas para visualização, edição e exclusão de conteúdos.	Serão realizados testes com usuários atribuídos a diferentes perfis, tentando acessar funcionalidades restritas. O sistema deverá permitir ou negar as ações conforme as permissões atribuídas, assegurando o correto isolamento funcional.

3.3 Performance

ID do requisito	Descrição	Critério de Aceitação/Avaliação
-----------------	-----------	---------------------------------

PERF-01	O tempo máximo de resposta deverá ser de até 2 segundos para listagens e consultas.	Testes de desempenho serão realizados utilizando ferramentas como Apache JMeter, Postman ou Locust, simulando requisições às funcionalidades principais do sistema. Os tempos de resposta serão medidos e comparados ao limite estabelecido de 2 segundos.
PERF-02	O sistema deverá oferecer suporte mínimo para 200 usuários simultâneos, mantendo sua estabilidade e desempenho.	Testes de carga e estresse serão executados com ferramentas como JMeter, simulando 200 conexões simultâneas ao sistema. Serão monitorados o tempo de resposta, taxa de erro, uso de CPU/memória e estabilidade do ambiente.
PERF-03	O tempo de inicialização do sistema, desde o início do carregamento até a disponibilidade total da interface, não deverá exceder 10 segundos.	A inicialização será cronometrada manualmente e por meio de logs automáticos, a partir do momento de execução do servidor até o carregamento completo da interface inicial. Esse tempo será avaliado em ambiente padrão de produção.
PERF-04	O sistema deve tratar corretamente acessos simultâneos a um mesmo recurso (como a compra de ingressos), evitando condições de corrida ou conflitos de transações.	Serão realizados testes simulando múltiplos usuários tentando executar operações críticas ao mesmo tempo (ex: adquirir o mesmo ingresso). O sistema será validado quanto à integridade dos dados e à correta emissão de mensagens para os usuários afetados. Situações de race condition e conflitos serão verificadas.
PERF-05	O sistema deve ser capaz de operar simultaneamente com múltiplos eventos ativos e suportar aumento gradual no número de usuários, feiras e expositores sem perda significativa de desempenho.	Durante testes de carga e planejamento de infraestrutura, será avaliado se o sistema pode ser replicado ou redistribuído em múltiplas instâncias (containers, servidores, serviços em nuvem) sem necessidade de reescrever os módulos centrais.

		Documentação técnica ou instruções de deploy escalável devem estar disponíveis para validar esse requisito.
--	--	---

3.4 Suportabilidade

ID do requisito	Descrição	Critério de Aceitação/Avaliação
SUP-01	O sistema deverá adotar uma arquitetura modular, permitindo atualizações independentes por módulo (feira, expositor, produto, ingresso).	Será realizada uma análise técnica da estrutura do sistema para verificar a separação lógica dos módulos. Serão testadas atualizações isoladas em cada módulo, assegurando que mudanças em um não impactem o funcionamento dos demais.
SUP-02	O código deverá ser documentado de forma clara, visando facilitar sua manutenção e evolução.	Serão analisados trechos de código-fonte para verificar a presença de comentários adequados, README, e documentação de APIs ou componentes. Ferramentas como JSDoc, Doxygen ou Typedoc poderão ser utilizadas para geração e verificação da documentação.
SUP-03	O sistema deverá ser compatível com os principais navegadores modernos: Google Chrome, Mozilla Firefox e Microsoft Edge.	Serão conduzidos testes manuais e automatizados de interface utilizando ferramentas como BrowserStack ou Selenium, garantindo que as funcionalidades principais operem corretamente em todos os navegadores especificados.
SUP-04	A infraestrutura do sistema deverá permitir escalonamento horizontal em ambiente de nuvem.	Testes serão realizados em ambiente de nuvem simulando o aumento do número de instâncias

		do sistema (escalonamento horizontal). Será verificado se a aplicação replica corretamente e mantém a integridade das sessões, sem necessidade de reconfiguração manual ou ajustes na lógica de negócio.
SUP-05	O sistema deverá permitir expansão da infraestrutura para atender à demanda crescente sem necessidade de reestruturação completa da arquitetura.	A avaliação incluirá a verificação de que o sistema pode ser implantado em múltiplas instâncias (por exemplo, via containers, balanceadores de carga ou serviços em nuvem), sem exigir alterações significativas na estrutura do código. Testes de carga poderão ser realizados em ambiente simulado para observar o comportamento da aplicação sob aumento progressivo de usuários, validando a capacidade de escalar horizontalmente.

4. Interfaces do sistema

ID do requisito	Descrição	Critério de Aceitação/Avaliação
SI-01	O sistema deverá disponibilizar uma API RESTful para integração com sistemas terceiros, com chamadas HTTP utilizando verbos e rotas semânticas.	A documentação da API (formato Swagger/OpenAPI) deverá listar todos os endpoints, parâmetros, respostas e códigos. Serão realizados testes com Postman/Insomnia para verificação da estrutura e acessibilidade da interface.
SI-02	Todas as respostas da API deverão estar no formato JSON e retornar os	As chamadas de API serão testadas em cenários

	códigos de status HTTP adequados, como 200 OK, 400 Bad Request, 401 Unauthorized.	bem-sucedidos e com erro. As respostas devem conter Content-Type: application/json, dados estruturados e o código de status apropriado.
SI-03	A API deverá incluir os seguintes endpoints: <ul style="list-style-type: none"> • GET /feiras • GET /feiras/{id} • POST /login • GET /expositores • POST /produtos • GET /ingressos/{id}/validar • POST /usuarios 	Cada endpoint será testado individualmente quanto à funcionalidade esperada. As rotas devem estar acessíveis, protegidas se necessário e retornar dados coerentes com os parâmetros fornecidos. Testes manuais e automatizados serão aplicados.
SI-04	A autenticação deverá ser realizada com tokens JWT. Após o login, um token será retornado e deverá ser incluído no cabeçalho Authorization das requisições.	Após autenticação via POST /login, a resposta deverá conter um token válido. As requisições subsequentes só deverão ser aceitas com o cabeçalho Authorization: Bearer <token>. Chamadas sem token deverão retornar 401 Unauthorized.
SI-05	Requisições com tokens ausentes, inválidos ou expirados deverão ser negadas com código 401 Unauthorized, conforme a política de segurança definida.	Serão testadas chamadas à API sem autenticação, com tokens inválidos e expirados. O sistema deverá bloquear o acesso e responder com 401, sem retornar conteúdo confidencial nem permitir avanço de fluxos protegidos.

4.1 Interfaces de usuário

ID do requisito	Descrição	Critério de Aceitação/Avaliação
UI-01	A interface do sistema será acessada por meio de uma aplicação web responsiva, compatível com	Serão realizados testes de responsividade utilizando ferramentas como Chrome

	navegadores modernos (Chrome, Firefox, Edge), acessível por desktop e dispositivos móveis.	DevTools, verificando comportamento em diferentes resoluções. A compatibilidade será testada nos principais navegadores em suas versões suportadas.
UI-02	Serão disponibilizadas telas para login, cadastro de usuários, gerenciamento de feiras, expositores, produtos e ingressos.	Cada funcionalidade será testada individualmente. As telas deverão apresentar formulários completos, mensagens de erro apropriadas e botões de ação funcionais, com navegação coerente entre os módulos.
UI-03	As listagens deverão conter campos de busca, filtros e paginação para facilitar a navegação e consulta de registros.	Serão realizados testes funcionais em listagens, validando a aplicação de filtros, a busca por palavras-chave e a navegação entre páginas. A usabilidade será avaliada com base na efetividade da localização de registros.

4.1.1 Aparência

ID do requisito	Descrição	Critério de Aceitação/Avaliação
LF-01	Estilo moderno e limpo, com paleta de cores baseada em tons neutros e verdes.	Será realizada inspeção visual com base em guia de estilo definido pelo projeto, verificando consistência na paleta de cores, alinhamento de elementos, legibilidade e uso adequado de espaçamentos e ícones. Avaliadores internos ou externos deverão confirmar a aderência ao padrão proposto por meio de revisão entre páginas.
LF-02	O sistema oferecerá acessibilidade básica por meio de compatibilidade mínima com leitores de tela, atendendo usuários com deficiência visual.	Serão conduzidos testes utilizando leitores de tela como NVDA (Windows) e VoiceOver (Mac/iOS), verificando se os elementos da interface possuem

		rótulos adequados, atributos aria-* e navegação sequencial lógica. A navegação por teclado será testada para garantir acesso completo sem uso do mouse.
LF-03	Garantia de responsividade em dispositivos com telas menores que 5”.	Testes de responsividade serão realizados com ferramentas como Chrome DevTools e BrowserStack, simulando dispositivos com largura entre 320px e 480px. A avaliação considerará legibilidade, acessibilidade dos menus e ausência de scroll horizontal inesperado.

4.1.2 Requisitos de layout e navegação

ID do requisito	Descrição	Critério de Aceitação/Avaliação
LNR-01	Menu lateral fixo com navegação hierárquica.	Durante os testes de interface, será verificado se o menu lateral permanece visível e fixo durante a navegação, e se apresenta os itens organizados em hierarquia. A hierarquia será validada por meio de inspeção funcional e usabilidade, garantindo clareza e fácil localização de funcionalidades.
LNR-02	As áreas de conteúdo serão centralizadas e organizadas logicamente de acordo com o contexto de uso (ex: feira, expositor, produto), para facilitar a localização das funcionalidades.	A disposição dos elementos será avaliada por inspeção visual e testes de usabilidade. Serão verificados o alinhamento central dos conteúdos e a organização contextual (ex: campos e ações de produtos visíveis junto aos dados do expositor). Será considerada a clareza na disposição dos blocos visuais e a facilidade de navegação para o usuário.

4.1.3 Consistência

ID do requisito	Descrição	Critério de Aceitação/Avaliação
CO-01	A interface utilizará uma biblioteca de componentes padronizada para garantir consistência visual e funcional em botões, campos, tabelas e outros elementos.	Serão realizados testes exploratórios para validar se os elementos visuais seguem o mesmo estilo, comportamento e hierarquia em todas as páginas. Inconsistências visuais entre componentes semelhantes em diferentes telas deverão ser ausentes ou justificadas.
CO-02	Será mantida terminologia uniforme em todas as telas e mensagens do sistema (ex: uso consistente dos termos “Feira”, “Expositor” e “Produto”).	Será realizada inspeção textual e funcional da interface, verificando se os termos-chave aparecem de forma consistente em botões, rótulos, mensagens de erro e títulos. Termos alternativos, sinônimos ou variações não padronizadas deverão ser corrigidos. Um glossário ou guia de terminologia poderá ser usado como base para verificação.

4.1.4 Requisitos de personalização e personalização do usuário

ID do requisito	Descrição	Critério de Aceitação/Avaliação
UPC-01	O sistema exibirá preferencialmente registros criados pelo usuário autenticado, priorizando informações relevantes ao seu contexto.	Será realizada verificação funcional autenticando usuários diferentes e validando se os registros exibidos nas listagens são, por padrão, aqueles vinculados ao respectivo usuário (usuario_criador_id). A ordem de exibição e a priorização contextual serão analisadas em interfaces como listagens de feiras, expositores e produtos.

UPC-02	Está prevista, em versões futuras, a possibilidade de personalização de notificações e filtros salvos nas listagens, conforme as preferências individuais do usuário.	<p>Quando implementado, serão realizados testes funcionais para verificar:</p> <ul style="list-style-type: none"> • Persistência das preferências do usuário (ex: frequência de notificação por e-mail); • Aplicação automática de filtros previamente salvos nas listagens (ex: expositores favoritos, feiras por localidade). Essas preferências deverão estar armazenadas por usuário e aplicadas automaticamente no carregamento das interfaces relevantes.
--------	---	---

4.2 Interfaces para sistemas ou dispositivos externos

ID do requisito	Descrição	Critério de Aceitação/Avaliação
IE-01	Não há integração prevista com dispositivos físicos (como leitores de código de barras, impressoras térmicas ou sensores) nesta versão do sistema.	Será verificado que não há chamadas, dependências ou bibliotecas vinculadas a dispositivos físicos no código-fonte, nem nas especificações técnicas das funcionalidades. Testes de integração focarão exclusivamente em interações via software, API e interface web.
IE-02	As interfaces externas previstas estão limitadas a sistemas de software e protocolos de comunicação padrão.	Será verificado que todas as integrações externas são realizadas via chamadas HTTP a serviços de terceiros ou internos, com padrões RESTful, sem dependência de hardware

		específico. As chamadas devem retornar dados em formato JSON e seguir convenções de status HTTP. Logs de comunicação e testes com ferramentas como Postman serão usados para validar essa conformidade.
--	--	---

4.2.1 Interfaces de software

ID do requisito	Descrição	Critério de Aceitação/Avaliação
SOE-01	Haverá integração com sistemas de pagamento (via APIs de terceiros) em versões futuras, permitindo a venda de ingressos diretamente pela plataforma.	Os testes envolverão simulações de compra de ingressos, verificação de resposta da API e retorno de status (pagamento aprovado, negado, cancelado). A segurança da transação deverá estar garantida por HTTPS e validação de tokens de acesso da API.
SOE-02	O sistema deverá permitir a integração com APIs de redes sociais (como Facebook, Instagram e X/Twitter) para facilitar o compartilhamento de eventos e páginas de expositores diretamente pelos usuários.	Serão realizados testes de compartilhamento a partir da interface do sistema, verificando se os botões correspondentes redirecionam corretamente para as interfaces das redes sociais, com pré-preenchimento das informações relevantes (ex: nome do evento, link da página). Será avaliado se os parâmetros de integração seguem as diretrizes oficiais das APIs sociais e se há fallback adequado caso o compartilhamento falhe.

4.2.2 Interfaces de hardware

ID do requisito	Descrição	Critério de Aceitação/Avaliação
HI-01	Sistema totalmente baseado na web, sem dependência de hardware específico no lado do cliente ou servidor.	A aplicação deverá funcionar em qualquer dispositivo com navegador moderno (desktop, tablet, smartphone), sem exigir instalação de drivers ou periféricos. Será validado que a execução do sistema ocorre apenas por meio de browser, e que a infraestrutura do servidor é agnóstica quanto a fabricante ou modelo. Serão utilizados ambientes de teste variados (ex: máquinas com Windows, Linux, navegadores diversos) para garantir a independência de hardware.

4.2.3 Interfaces de Comunicação

ID do requisito	Descrição	Critério de Aceitação/Avaliação
CI-01	Todas as comunicações ocorrerão por meio de requisições HTTP/HTTPS sobre rede pública.	<p>Serão utilizados testes com ferramentas como SSL Labs, curl -v, ou o console de rede dos navegadores para confirmar que:</p> <ul style="list-style-type: none"> • O protocolo TLS 1.2 ou superior está ativo; • Não há requisições HTTP sem redirecionamento para HTTPS; • Os cabeçalhos de segurança (Strict-Transport-Security, Content-Security-Policy, etc.) estão

		corretamente configurados, conforme boas práticas da OWASP.
CI-02	Os dados em trânsito deverão ser protegidos com criptografia TLS 1.2 ou superior, garantindo a segurança das informações trocadas entre cliente e servidor.	<p>Ferramentas como SSL Labs, OpenSSL, curl ou extensões de navegador serão utilizadas para confirmar que:</p> <ul style="list-style-type: none"> • A conexão utiliza TLS 1.2 ou superior; • Nenhuma requisição sensível é transmitida em texto plano (HTTP); • Cabeçalhos de segurança como Strict-Transport-Security estão corretamente configurados no servidor.

5. Regras de negócio

5.1 Regras de Validação e Permissão

Código	Descrição	Padrão
R001	Apenas o criador do registro pode editá-lo ou excluí-lo	if [currentUser.id != record.createdBy] then [blockAction()], returnError(403, "Acesso negado: você não é o proprietário deste registro")]
R002	Administradores podem editar qualquer registro mas só excluem registros sem vínculos	if [user.role == "admin" && record.dependencies.count == 0] then [allowAction()]
R003	Bloqueia exclusão de registros com dependências ativas.	if [record.hasActiveDependencies()] then

		[showWarning("Exclusão bloqueada: existem registros vinculados")]
R004	Campos obrigatórios devem ser preenchidos antes do salvamento.	if [form.validateRequiredFields()]== false] then [highlightEmptyFields(), preventSubmission()]

5.2 Regras baseadas em tempo

Código	Descrição	Padrão
R005	Ingressos são válidos apenas na data do evento correspondente.	if [ticket.eventDate != currentDate] then [invalidateTicket(), notifyUser("Ingresso expirado")]
R006	Registros só podem ser editados até 24h antes do evento	if [event.startTime - currentTime < 24.hours] then [disableEditing(), showMessage("Período de edição encerrado")]

5.3 Regras Financeiras

Código	Descrição	Padrão
R007	Reembolsos só são permitidos até 48h antes do evento.	if [event.startTime - currentTime < 48.hours] then [denyRefund(), showPolicy("Reembolso não permitido neste prazo")]
R008	Descontos só aplicam para compras acima	if [cart.tickets.count >= 3]

	de 3 ingressos.	then [applyDiscount(10%)]
--	-----------------	---------------------------

6. Restrições do sistema

- Desenvolvimento em linguagem de programação moderna (JavaScript/TypeScript com Node.js no backend e React no frontend).
- Uso de banco de dados relacional (PostgreSQL).
- Hospedagem em nuvem (AWS, Azure ou similar), sem dependência de hardware proprietário.
- Ferramentas de versionamento como **Git** (via GitHub) e integração contínua serão obrigatórias para rastreamento das alterações.
- O desenvolvimento deverá seguir padrões estabelecidos de codificação (como Airbnb Style Guide para JavaScript), com revisão de código via pull requests.
- As dependências externas devem seguir as regras de licenciamento livre (ex: MIT, Apache 2.0), conforme alinhado com os objetivos de software aberto descritos no Documento de Visão.

7. Conformidade do sistema

7.1 Requisitos de Licenciamento

- O sistema deve implementar controle de licenciamento para usuários premium, se aplicável em versões futuras.

7.2 Avisos legais, de direitos autorais e outros

- Todo conteúdo cadastrado no sistema deve estar sujeito à política de privacidade e termos de uso da plataforma.
- A aplicação deve exibir e exigir aceitação explícita dos termos de uso e política de privacidade no momento do cadastro do usuário.

7.3 Normas aplicáveis

- A aplicação deve adotar princípios de segurança amplamente reconhecidos, como o OWASP Top 10.
- A interface deve seguir diretrizes básicas de acessibilidade e responsividade, conforme WCAG 2.1 AA, para garantir uma experiência positiva a todos os perfis de usuário.

8. Documentação do sistema

- Manual do usuário acessível via interface do sistema.
- Ajuda contextual (tooltips e tutoriais iniciais).
- Documentação da API para desenvolvedores externos.