

Arquitetura de Software 4 + 1

Versão 1.0

Histórico da Revisão

Data	Versão	Descrição	Autor
1º de outubro de 2025	1.0	Desenvolvimento inicial do documento contendo estrutura base e seções principais.	Marlon
6 de outubro de 2025	1.1	Revisão geral e ajustes do documento, ajustes de formatação e correções de texto.	Carlos
12 de outubro de 2025	1.2	Auxílio nos ajustes do documento e revisão das seções de requisitos.	Bernardo
15 de outubro de 2025	1.3	Criação do diagrama UML representando os principais casos de uso do sistema.	Matheus
18 de outubro de 2025	1.4	Elaboração do Diagrama de Casos de Uso Significativos com base nos requisitos do projeto.	Carlos
10 de outubro de 2025	1.5	Atualização da seção “Visão Lógica” para correção de conflitos e adequação ao modelo monolítico em camadas.	Marlon e Ricardo
20 de outubro de 2025	1.6	Criação da tabela de referência utilizada na modelagem do sistema.	Matheus
21 de outubro de 2025	1.7	Implantação do sistema e validação das funcionalidades principais.	Bernardo
23 de outubro de 2025	1.8	Alteração e atualização do Diagrama de Casos de Uso.	Carlos
23 de outubro de 2025	1.9	Alteração e atualização da tabela de referência.	Matheus
23 de outubro de 2025	2.0	Atualização e melhorias na implantação do sistema.	Bernardo
23 de outubro de 2025	2.1	Inclusão das referências bibliográficas e atualização da seção “Qualidade”.	Marlon

24 de outubro de 2025	2.2	Revisão estrutural do documento, com reorganização das seções conforme feedback do professor e ajuste na numeração dos requisitos funcionais (RFs).	Carlos
24 de outubro de 2025	2.3	Revisão de padronização visual do documento (margens, espaçamentos, numeração de páginas e legendas de figuras).	Bernardo
24 de outubro de 2025	2.4	Transformação do documento para o formato conforme as normas da ABNT e realização de ajustes finais.	Carlos, Bernardo e Marlon
24 de outubro de 2025	2.5	Elaboração do Diagrama de pacotes	Ricardo
24 de outubro de 2025	2.6	Revisão final de validação acadêmica e técnica, consolidando o documento para entrega e arquivamento institucional.	Marlon

SUMÁRIO

1	Introdução	5
1.1	Objetivo	6
1.2	Escopo	6
1.3	Definições, Acrônimos e Abreviações	7
2	Representação da Arquitetura	8
2.1	Objetivos e Restrições da Arquitetura	9
2.2	Visão de Casos de Uso	11
2.3	Diagrama Casos de Uso Significativos	12
2.3.1	RF08 – Gerar relatório	13
2.3.2	RF02 – Cadastrar Evento	15
2.3.3	RF02 – Inscrever em Evento	16
2.3.4	RF02 – Emitir Certificado	17
2.3.5	RF02 – Validar Presença	18
2.4	Visão Lógica	19
2.4.1	Diagrama de Pacotes e Subsistemas	21
2.4.2	Diagrama de Classes	22
2.4.3	Dicionário de Dados	23
2.5	Visão da Implantação	27
2.6	Qualidade	30
3	Referências	31

1 Introdução

A organização de eventos é uma atividade que envolve múltiplas etapas e exige alto nível de coordenação entre diversos agentes. Desde o planejamento inicial até o encerramento, é necessário lidar com tarefas como divulgação, controle de inscrições, emissão de certificados, comunicação com participantes, gestão de pagamentos e avaliação de resultados. Quando essas atividades são realizadas manualmente ou com ferramentas desconectadas, aumentam-se os riscos de falhas, retrabalho e insatisfação dos envolvidos.

Com o avanço da tecnologia e a popularização de eventos híbridos e online, tornou-se essencial contar com sistemas que automatizem e integrem essas funções, oferecendo uma experiência mais fluida tanto para organizadores quanto para participantes. Nesse contexto, propõe-se o desenvolvimento, um sistema de gestão de eventos completo, voltado para facilitar a administração de eventos acadêmicos, corporativos, culturais e sociais.

Permitirá que organizadores criem eventos com facilidade, definam datas, locais (físicos ou virtuais), categorias de ingresso, limites de participantes e regras de inscrição. Os usuários poderão se inscrever, receber notificações, acessar conteúdos exclusivos, participar de atividades e obter certificados de participação. O sistema também oferecerá recursos para controle de presença, geração de relatórios, integração com plataformas de pagamento e envio de e-mails automáticos.

A arquitetura do foi projetada com base no modelo **4+1 de visões arquiteturais**, que permite representar o sistema sob diferentes perspectivas: lógica, de casos de uso, de implantação, de qualidade e de desenvolvimento. Essa abordagem garante que os requisitos funcionais (como autenticação, inscrição e emissão de certificados) e não funcionais (como desempenho, segurança e disponibilidade) sejam atendidos de forma estruturada, promovendo escalabilidade, confiabilidade e facilidade de manutenção.

1.1 Objetivo

Este documento tem como objetivo **apresentar a arquitetura de software do sistema**, descrevendo suas principais decisões e componentes. São detalhadas as visões de casos de uso, visão lógica, visão de implantação e requisitos de qualidade, conforme o modelo 4+1. O documento busca servir como referência técnica para o desenvolvimento e manutenção do sistema, garantindo alinhamento entre os requisitos do projeto e sua implementação.

1.2 Escopo

O sistema tem como escopo principal oferecer uma solução integrada para o gerenciamento de eventos presenciais, online e híbridos. Ele será utilizado por diferentes perfis de usuários, como organizadores, participantes e palestrantes, permitindo o controle completo do ciclo de vida de um evento, desde o planejamento inicial até o encerramento. O sistema permitirá que os organizadores cadastrem eventos com informações detalhadas, como datas, locais físicos ou virtuais, categorias de ingresso, regras de participação e limites de vagas. Os participantes poderão realizar inscrições, receber notificações, acessar conteúdos exclusivos, validar presença e obter certificados personalizados. Os palestrantes terão acesso a funcionalidades específicas para gerenciar suas atividades e interações com o público.

Além disso, o sistema oferecerá recursos para comunicação automatizada com os inscritos, geração de relatórios gerenciais, controle de presença por meio de QR Code ou login, e integração com plataformas de pagamento para eventos que exigem cobrança de inscrição. A interface será responsiva, permitindo o uso tanto em computadores quanto em dispositivos móveis, garantindo flexibilidade e praticidade para todos os usuários.

Por outro lado, o sistema não incluirá funcionalidades como transmissão ao vivo de eventos, gestão financeira detalhada, suporte técnico em tempo real durante o evento, nem módulos de gamificação ou redes sociais internas. Essas exclusões foram definidas com o objetivo de manter o foco do projeto nas funcionalidades essenciais para a organização e acompanhamento de eventos,

assegurando simplicidade, eficiência e viabilidade técnica dentro do escopo estabelecido.

1.3 Definições, Acrônimos e Abreviações

Para garantir a clareza e o entendimento uniforme dos termos utilizados ao longo deste documento, esta seção apresenta as principais definições, acrônimos e abreviações relacionados ao sistema. O termo “Evento” refere-se a qualquer atividade organizada com data, local e público definidos, podendo ser presencial, online ou híbrida. “Organizador” é o usuário responsável pela criação, configuração e gerenciamento dos eventos dentro do sistema. “Participante” é o usuário que realiza a inscrição em um evento e interage com suas funcionalidades, como acesso aos conteúdos, confirmação de presença e emissão de certificados. “Palestrante” é o usuário convidado para ministrar palestras, oficinas ou atividades durante o evento.

O acrônimo “QR Code” refere-se ao código de resposta rápida utilizado para validar a presença dos participantes de forma prática e segura. “UI” (User Interface) representa a interface do usuário, ou seja, a camada visual com a qual os usuários interagem. “DBMS” (Database Management System) refere-se ao sistema de gerenciamento de banco de dados utilizado para armazenar e organizar as informações do sistema. “API” (Application Programming Interface) é o conjunto de regras e protocolos que permite a comunicação entre sistemas externos, como plataformas de pagamento. “CRUD” (Create, Read, Update, Delete) representa as operações básicas de manipulação de dados realizadas no sistema. “HTTPS” (Hypertext Transfer Protocol Secure) é o protocolo de comunicação utilizado para garantir a segurança na troca de informações entre o cliente e o servidor. Por fim, “Responsividade” refere-se à capacidade da interface do sistema de se adaptar a diferentes tamanhos de tela e dispositivos, como computadores, tablets e smartphones.

2 Representação da Arquitetura

A arquitetura do sistema é apresentada por meio de uma abordagem baseada em múltiplas visões, conforme o modelo 4+1 de arquitetura de software. Essa abordagem permite representar diferentes aspectos do sistema, facilitando a compreensão das decisões técnicas e estruturais adotadas. As visões descritas neste documento incluem a visão de casos de uso, que foca nos principais cenários de interação entre os usuários e o sistema; a visão lógica, que detalha os componentes internos, suas responsabilidades e relacionamentos; a visão de processo, que aborda os aspectos dinâmicos e de concorrência do sistema; e a visão de implantação, que descreve a distribuição dos componentes em ambientes físicos e tecnológicos.

Não será apresentada uma visão de implementação neste documento, pois o foco está nas estruturas conceituais e funcionais que orientam o desenvolvimento do sistema. Todas as visões são baseadas em um modelo UML (Unified Modeling Language), desenvolvido com o apoio de ferramentas de modelagem como o Rational Rose, que permite representar graficamente os elementos da arquitetura e suas interações. Essa estrutura visa garantir que o sistema seja compreendido de forma clara por todos os envolvidos no projeto, desde analistas e desenvolvedores até os stakeholders responsáveis pela validação da solução.

2.1 Objetivos e Restrições da Arquitetura

A arquitetura do sistema foi desenvolvida considerando uma série de requisitos essenciais e restrições técnicas que influenciam diretamente suas decisões estruturais. Esses elementos foram definidos com base nas necessidades dos usuários, nas características do domínio de eventos e nas tecnologias escolhidas para a implementação. A seguir, são apresentados os principais objetivos e restrições que orientam a construção da arquitetura:

1. O sistema deve permitir o gerenciamento simultâneo de múltiplos eventos, garantindo escalabilidade para lidar com grande volume de inscrições e acessos.
2. A plataforma deve ser acessível tanto em computadores quanto em dispositivos móveis, com interface responsiva e compatível com os principais navegadores modernos.
3. A autenticação de usuários deve ser obrigatória, com controle de acesso baseado em perfis (organizador, participante, palestrante), garantindo segurança e privacidade dos dados.
4. O sistema deve integrar-se com plataformas de pagamento externas, como PayPal e Stripe, para permitir a cobrança de inscrições pagas de forma segura e transparente.
5. A arquitetura deve seguir o modelo cliente/servidor, com o frontend operando em navegadores e o backend hospedado em servidores cloud com suporte a escalabilidade horizontal.
6. O tempo de resposta para operações críticas, como login, inscrição e emissão de certificados, não deve ultrapassar dois segundos, garantindo boa experiência ao usuário.

7. O sistema deve estar disponível 24 horas por dia, sete dias por semana, com tolerância a falhas e tempo de inatividade inferior a 4%.

8. A comunicação entre os módulos internos e externos deve ser feita por meio de APIs RESTful, garantindo interoperabilidade e facilidade de manutenção.

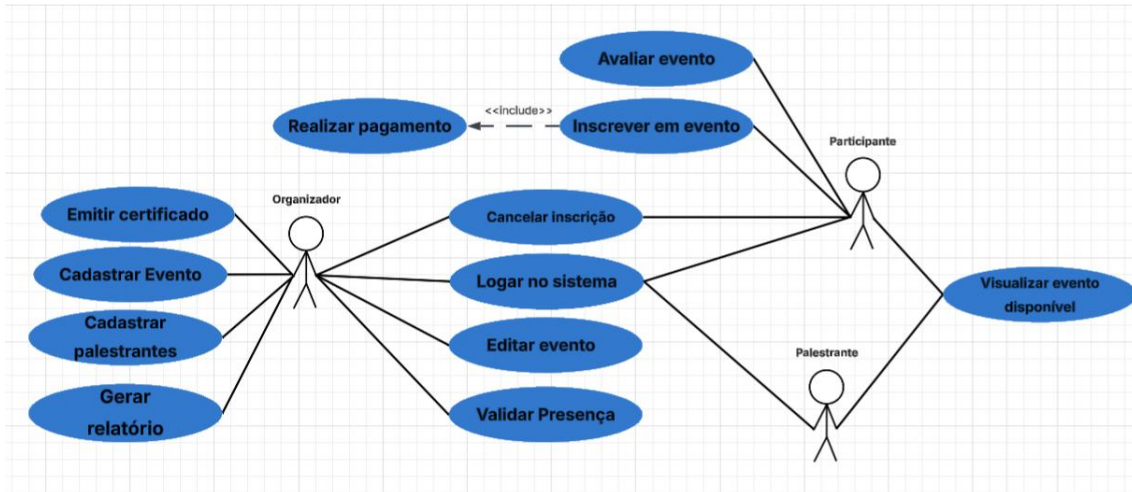
2.2 Visão de Casos de Uso

A visão de casos de uso da arquitetura de software do sistema é fundamental para identificar os principais cenários de interação entre os usuários e o sistema. Essa visão orienta a seleção dos casos de uso que serão foco das iterações de desenvolvimento, destacando funcionalidades centrais e significativas que exercitam diversos elementos da arquitetura. Os casos de uso descritos a seguir representam ações essenciais para o funcionamento do sistema e ilustram pontos críticos que exigem atenção especial na definição da estrutura técnica. Eles envolvem diferentes perfis de usuários, como organizadores, participantes e palestrantes, além de interações com serviços externos, como plataformas de pagamento e sistemas de envio de e-mails.

1. Logar no sistema
2. Cadastrar evento
3. Editar evento
4. Inscrever em evento
5. Cancelar inscrição
6. Emitir certificado
7. Validar presença
8. Gerar relatório
9. Cadastrar palestrante
10. Visualizar evento disponível
11. Realizar pagamento
12. Avaliar evento

Esses casos de uso são iniciados pelos agentes organizador, participante ou palestrante. Além disso, o sistema interage com agentes externos, como gateways de pagamento e serviços de e-mail, que complementam as funcionalidades oferecidas e exigem integração arquitetural adequada.

2.3 Diagrama Casos de Uso Significativos



Link: https://lucid.app/lucidchart/e761ebb0-befa-4677-a1ef-1f56e64608db/edit?viewport_loc=-2688%2C-1926%2C7049%2C2781%2C0_0&invitationId=inv_a621389c-431d-417e-ad82-4b22dedc0d0f

Nome do Diagrama: Casos de Uso Significativos para a Arquitetura

2.3.1 RF08 – Gerar relatório

Nome: Gerar relatório

Atores: Organizador

Descrição: Este caso de uso permite que o Organizador gere relatórios gerenciais sobre os eventos cadastrados no sistema. Os relatórios podem incluir informações como número total de inscritos, número de certificados emitidos, taxa de presença, receita arrecadada (para eventos pagos) e avaliações dos participantes. Os dados são obtidos do banco de dados e apresentados em formato visual e exportável (PDF ou CSV).

Pré-condições

- O Organizador deve estar autenticado no sistema.
- O evento deve estar cadastrado e possuir dados de inscrições ou certificados emitidos.

Fluxo Principal

1. O Organizador acessa o menu “**Relatórios**” no painel administrativo.
2. O sistema exibe as opções de relatório disponíveis (inscrições, certificados, presença, receita).
3. O Organizador seleciona o tipo de relatório desejado e o evento correspondente.
4. O sistema coleta os dados necessários do banco de dados.
5. O sistema gera o relatório e exibe uma visualização prévia.
6. O Organizador pode optar por exportar o relatório em PDF ou CSV.
7. O sistema registra a ação de geração de relatório para auditoria.

Fluxo Alternativo

Credenciais inválidas:

- Se as credenciais inseridas forem inválidas, o sistema exibe uma mensagem de erro informando o usuário sobre a falha no login.
- O usuário pode optar por tentar fazer login novamente ou recuperar sua senha através da funcionalidade de recuperação de senha.

Usuário não registrado:

- Se o sistema identificar que o e-mail não está cadastrado, uma mensagem é exibida informando que o usuário não existe no sistema.
- O usuário pode optar por entrar em contato com o suporte ou realizar o cadastro, se houver essa opção disponível.

2.3.2 RF02 – Cadastrar Evento

Nome do Caso de Uso: Cadastrar Evento

Ator: Organizador

Descrição: Este caso de uso permite que um Organizador cadastre um novo evento no sistema, definindo informações como título, descrição, data, local, número de vagas, tipo de inscrição e regras de participação.

Pré-condições

- O Organizador está autenticado no sistema.

Fluxo Principal

1. O Organizador acessa o painel de gerenciamento de eventos.
2. O Organizador seleciona a opção "Cadastrar Evento".
3. O sistema exibe o formulário de cadastro de evento.
4. O Organizador preenche os campos obrigatórios e confirma a criação.
5. O sistema valida os dados e salva o evento.
6. O sistema disponibiliza o evento para visualização pública.

Fluxo Alternativo

Dados inválidos ou incompletos:

- Se houver erro no preenchimento, o sistema exibe mensagens de validação.
- O Organizador corrige os dados e tenta novamente.

Falha no salvamento:

- Se ocorrer falha na comunicação com o banco de dados, o sistema exibe uma mensagem de erro.
- O Organizador pode optar por tentar novamente ou salvar o evento como rascunho.

2.3.3 RF02 – Inscrever em Evento

Nome do Caso de Uso: Inscrever em Evento

Ator: Participante

Descrição: Este caso de uso permite que um Participante realize sua inscrição em um evento disponível no sistema. O participante pode visualizar os eventos abertos, selecionar aquele de interesse e confirmar sua inscrição, respeitando as regras definidas pelo Organizador.

Pré-condições

- O Participante está autenticado no sistema.
- O evento está com inscrições abertas e possui vagas disponíveis.

Fluxo Principal

1. O Participante acessa a área de eventos disponíveis.
2. O sistema exibe a lista de eventos com inscrições abertas.
3. O Participante seleciona um evento e confirma a inscrição.
4. O sistema valida os dados e registra a inscrição.
5. O sistema envia uma notificação de confirmação ao Participante.

Fluxo Alternativo

Evento com vagas esgotadas:

- O sistema exibe uma mensagem informando que não há mais vagas disponíveis.
- O Participante pode optar por entrar em uma lista de espera, se disponível.

Dados inválidos:

- Se houver erro nos dados do participante, o sistema exibe uma mensagem de validação.
- O Participante corrige os dados e tenta novamente.

2.3.4 RF02 – Emitir Certificado

Nome do Caso de Uso: Emitir Certificado

Ator: Participante

Descrição: Este caso de uso permite que o Participante emita seu certificado de participação após ter sua presença validada no evento. O certificado é gerado automaticamente pelo sistema e disponibilizado em formato PDF.

Pré-condições

- O Participante está autenticado no sistema.
- A presença do Participante foi registrada no evento.

Fluxo Principal

1. O Participante acessa a área de certificados.
2. O sistema exibe os eventos em que o Participante teve presença confirmada.
3. O Participante seleciona o evento desejado.
4. O sistema gera o certificado em formato PDF.
5. O sistema disponibiliza o certificado para download.

Fluxo Alternativo

Presença não registrada:

- O sistema informa que o certificado não pode ser emitido, pois a presença não foi confirmada.
- O Participante pode entrar em contato com o Organizador para verificar a situação.

2.3.5 RF02 – Validar Presença

Nome do Caso de Uso: Validar Presença

Ator: Organizador

Descrição: Este caso de uso permite que um Organizador cadastre um novo evento no sistema, definindo informações como título, descrição, data, local, número de vagas, tipo de inscrição e regras de participação.

Pré-condições

- O Organizador está autenticado no sistema.
- O evento está em andamento.
- O Participante está inscrito no evento.

Fluxo Principal

1. O Organizador acessa a lista de inscritos do evento.
2. O sistema exibe os dados dos Participantes.
3. O Organizador realiza a leitura do QR Code ou marca a presença manualmente.
4. O sistema registra a presença do Participante.
5. O sistema atualiza o status do Participante para “Presença Confirmada”.

Fluxo Alternativo

Participante não inscrito:

- O sistema impede a validação e exibe uma mensagem de erro.
- O Organizador pode verificar os dados ou solicitar inscrição manual.

Falha na leitura do QR Code:

- O sistema exibe uma mensagem de erro.
- O Organizador pode optar pela validação manual.

Visão Lógica

A **Visão Lógica** da arquitetura descreve a estrutura interna do sistema, evidenciando os principais componentes, camadas e relacionamentos entre as partes que o compõem. O objetivo dessa visão é demonstrar como o sistema está organizado em termos de responsabilidades e interações, proporcionando uma compreensão clara do funcionamento da aplicação e de sua divisão lógica. O sistema foi desenvolvido com base em uma arquitetura monolítica em camadas, modelo amplamente utilizado em sistemas corporativos e acadêmicos de médio porte. Essa abordagem garante organização, separação de responsabilidades e facilidade de manutenção, permitindo a evolução e expansão do sistema de forma estruturada.

A arquitetura é composta pelas seguintes camadas:

Camada de Apresentação

A camada de Apresentação é responsável pela interface com o usuário, oferecendo uma experiência responsiva e intuitiva em navegadores web, tanto em computadores quanto em dispositivos móveis. Ela é composta por telas que permitem realizar operações como login, cadastro de eventos, inscrições, emissão de certificados, relatórios e notificações. Essa camada se comunica com a camada de Aplicação por meio de requisições HTTPS, garantindo segurança na troca de informações.

Camada de Aplicação

A camada de Aplicação contém os serviços e controladores que coordenam as regras de negócio e o fluxo das funcionalidades do sistema. Aqui são executados os principais casos de uso, como gerenciar eventos, processar inscrições, emitir certificados, validar presença, enviar notificações e gerar relatórios. Essa camada atua como intermediária entre a Apresentação e o Domínio, garantindo que as regras sejam corretamente aplicadas.

Camada de Domínio

A camada de Domínio representa o núcleo lógico do sistema, onde estão definidas as entidades e regras de negócio que sustentam as funcionalidades. Entre as principais classes, destacam-se: Evento, Participante, Organizador, Palestrante, Inscrição, Pagamento e Certificado. Essas classes encapsulam comportamentos e dados essenciais, assegurando a integridade e consistência das operações realizadas.

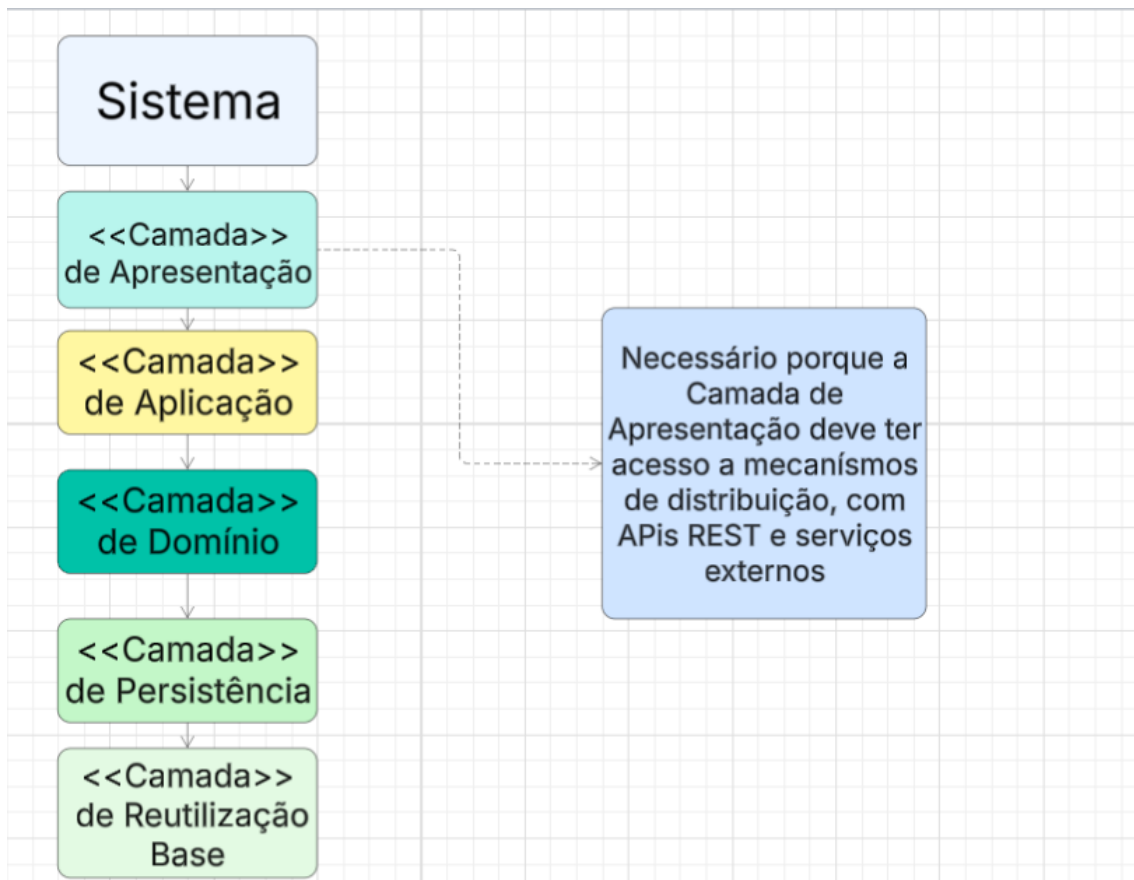
Camada de Persistência

A camada de Persistência é responsável por armazenar e recuperar informações no banco de dados relacional PostgreSQL. Ela utiliza repositórios que abstraem as operações de leitura, escrita, atualização e exclusão de dados (CRUD), além de gerenciar conexões com serviços externos. Nessa camada também estão os adaptadores de integração com plataformas de pagamento (PayPal, Stripe) e serviços de e-mail (SendGrid, Amazon SES).

Camada de Reutilização Base

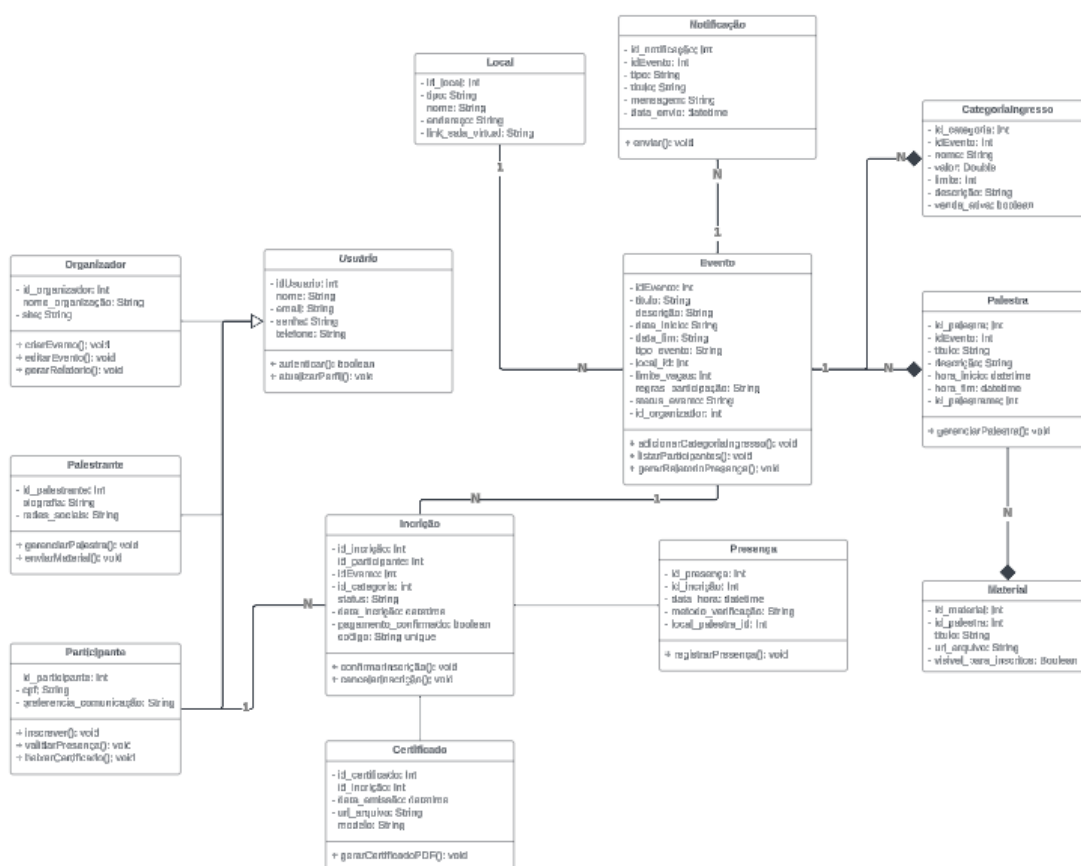
A camada de Reutilização Base contém classes e componentes utilitários compartilhados por todo o sistema. Inclui funções de validação de dados, geração de QR Codes, tratamento de erros, formatação de datas, logs e autenticação. Esses recursos promovem padronização, consistência e reaproveitamento de código, reduzindo retrabalho e aumentando a produtividade no desenvolvimento.

2.3.6 Diagrama de Pacotes e Subsistemas



Link: https://lucid.app/lucidchart/62452437-a368-4e18-8316-439bb4f3f2ac/edit?viewport_loc=-648%2C-38%2C3278%2C1524%2C0_0&invitationId=inv_f3b551ea-77f6-4499-b543-f952b87c8bf7

2.3.7 Diagrama de Classes



Link: https://lucid.app/lucidchart/17891058-caa0-469d-aea8-189f86aa25a3/edit?viewport_loc=-2720%2C-1917%2C6074%2C2373%2C0_0&invitationId=inv_2d5f8387-2c4a-4b05-98e8-7c502df4a850

2.3.8 Dicionário de Dados

Nome da tabela: USUÁRIO			
Descrição da tabela: Cadastro de todos os usuários do sistema			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_USUARIO	VARCHAR(36)	NOT NULL	IDENTIFICADOR ÚNICO DO USUÁRIO
NOME	VARCHAR(255)	NOT NULL	NOME COMPLETO DO USUÁRIO
EMAIL	VARCHAR(255)	NOT NULL	ENDEREÇO DE EMAIL DO USUÁRIO (UNIQUE)
SENHA	VARCHAR(255)	NOT NULL	SENHA CRIPTOGRAFADA DO USUÁRIO
TELEFONE	VARCHAR(30)	NULL	TELEFONE DE CONTATO
TIPO_USUARIO	VARCHAR(50)	NOT NULL	ORGANIZADOR/PARTICIPANTE/PALESTRANTE
CRIADO_EM	DATETIME	NOT NULL	DATA DE CRIAÇÃO DO CADASTRO

Nome da tabela: ORGANIZADOR			
Descrição da tabela: Dados adicionais para usuários que são organizadores.			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_ORGANIZADOR	VARCHAR(36)	NOT NULL	IDENTIFICADOR DO ORGANIZADOR
UUID_USUARIO	VARCHAR(36)	NOT NULL	FK USUÁRIO
NOME_ORGANIZAÇÃO	VARCHAR(255)	NULL	NOME DA ORGANIZAÇÃO/EMPRESA
SITE	VARCHAR(255)	NULL	SITE INSTITUCIONAL
TELEFONE_COMERCIAL	VARCHAR(30)	NULL	TELEFONE COMERCIAL

Nome da tabela: PARTICIPANTE			
Descrição da tabela: Dados específicos de participantes.			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_PARTICIPANTE	VARCHAR(36)	NOT NULL	IDENTIFICADOR DO PARTICIPANTE
UUID_USUARIO	VARCHAR(36)	NOT NULL	FK USUÁRIO
CPF	VARCHAR(14)	NOT NULL	CPF DO PARTICIPANTE (UNIQUE)
PREFERENCIA_COMUNICAÇÃO	VARCHAR(20)	NULL	EMAIL/PUSH/SMS
DATA_NASCIMENTO	DATETIME	NULL	DATA DE NASCIMENTO

Nome da tabela: PALESTRANTE			
Descrição da tabela: Informações dos palestrantes.			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_PALESTRANTE	VARCHAR(36)	NOT NULL	IDENTIFICADOR DO PARTICIPANTE
UUID_USUARIO	VARCHAR(36)	NOT NULL	FK USUÁRIO
BIOGRAFIA	TEXT	NULL	BIOGRAFIA DO PALESTRANTE
REDES_SOCIAIS	VARCHAR(500)	NULL	LINKS DAS REDES SOCIAIS
ESPECIALIDADE	VARCHAR(100)	NULL	ÁREA DE ATUAÇÃO

Nome da tabela: LOCAL			
Descrição da tabela: Locais (físicos ou virtuais) onde eventos ocorrem.			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_LOCAL	VARCHAR(36)	NOT NULL	IDENTIFICADOR DO PARTICIPANTE
TIPO	VARCHAR(10)	NOT NULL	FISICO/VIRTUAL
NOME	TEXT	NOT NULL	NOME DO LOCAL/SALA
ENDEREÇO	VARCHAR(500)	NULL	ENDEREÇO COMPLETO (SE FÍSICO)
LINK_SALA_VIRTUAL	VARCHAR(500)	NULL	LINK DA SALA VIRTUAL (SE VIRTUAL)
CAPACIDADE	VARCHAR(100)	NULL	CAPACIDADE MÁXIMA (SE APLICÁVEL)

Nome da tabela: EVENTO			
Descrição da tabela: Representa um evento criado por um organizador.			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_EVENTO	VARCHAR(36)	NOT NULL	IDENTIFICADOR DO EVENTO
UUID_ORGANIZADOR	VARCHAR(36)	NOT NULL	FK ORGANIZADOR
UUID_LOCAL	VARCHAR(36)	NULL	FK LOCAL
TITULO	VARCHAR(36)	NOT NULL	TÍTULO DO EVENTO
DESCRIÇÃO	TEXT	NULL	DESCRIÇÃO DETALHADA
DATA_INICIO	DATETIME	NOT NULL	DATA/HORA DE INÍCIO
DATA_FIM	DATETIME	NOT NULL	DATA/HORA DE TÉRMINO
TIPO_EVENTO	VARCHAR(36)	NOT NULL	PRESENCIAL / ONLINE / HIBRIDO
LIMITE_VAGAS	INT	NULL	LIMITE DE PARTICIPANTES (NULL = SEM LIMITES)
REGRAS_PARTICIPAÇÃO	TEXT	NULL	TERMOS E REGRAS
STATUS_EVENTO	VARCHAR(36)	NOT NULL	RASCUNHO/PUBLICADO/CANCELADO/ENCERRADO
CRIADO_EM	DATETIME	NOT NULL	TIMESTAMP DE CRIAÇÃO

Nome da tabela: CATEGORIA_INGRESSO			
Descrição da tabela: Tipos de ingresso por evento (VIP, padrão, meia, etc).			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_CATEGORIA	VARCHAR(36)	NOT NULL	IDENTIFICADOR DA CATEGORIA
UUID_EVENTO	VARCHAR(36)	NOT NULL	FK EVENTO
NOME	VARCHAR(100)	NOT NULL	NOME DA CATEGORIA
VALOR	DECIMAL(10,2)	NULL	PREÇO DO INGRESSO
LIMITE	INT	NULL	QUANTIDADE DISPONÍVEL
DESCRIÇÃO	TEXT	NULL	OBSERVAÇÕES SOBRE A CATEGORIA
VENDA_ATIVA	BOOLEAN	NOT NULL	TRUE = DISPONÍVEL À VENDA

Nome da tabela: PALESTRA			
Descrição da tabela: Sessões/palestras dentro de um evento.			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_PALESTRA	VARCHAR(36)	NOT NULL	IDENTIFICADOR DA PALESTRA
UUID_EVENTO	VARCHAR(36)	NOT NULL	FK EVENTO
UUID_PALESTRANTE	VARCHAR(36)	NULL	FK PALESTRANTE
TITULO	VARCHAR(255)	NOT NULL	TITULO DA PALESTRA
DESCRIÇÃO	TEXT	NULL	DESCRIÇÃO DA PALESTRA
HORA_INICO	DATETIME	NOT NULL	INICIO DA PALESTRA
HORA_FIM	DATETIME	NOT NULL	FIM DA PALESTRA
SALA	VARCHAR(255)	NULL	SALA/LINK ESPECIFICO DA SESSÃO

Nome da tabela: MATERIAL			
Descrição da tabela: Materiais disponibilizados por palestra (slides, PDFs, vídeo).			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_MATERIAL	VARCHAR(36)	NOT NULL	IDENTIFICADOR DO MATERIAL
UUID_PALESTRA	VARCHAR(36)	NOT NULL	FK PALESTRA
TITULO	VARCHAR(255)	NOT NULL	TITULO DO MATERIAL
URL_ARQUIVO	VARCHAR(500)	NOT NULL	LINK/CAMINHO DO ARQUIVO
VISIVEL_PARA_INSCRITOS	BOOLEAN	NOT NULL	TRUE = SOMENTE INSCRITOS PODEM ACESSAR

Nome da tabela: INSCRIÇÃO			
Inscrições/ingressos de participantes em eventos.			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_INSCRIÇÃO	VARCHAR(36)	NOT NULL	IDENTIFICADOR DO MATERIAL
UUID_EVENTO	VARCHAR(36)	NOT NULL	FK PALESTRA
UUID_PARTICIPANTE	VARCHAR(36)	NOT NULL	FK PARTICIPANTE
UUID_CATEGORIA	VARCHAR(36)	NULL	FK CATEGORIA
DATA_INSCRIÇÃO	DATETIME	NOT NULL	TIMESTAMP DA INSCRIÇÃO
STATUS	VARCHAR(20)	NOT NULL	PENDENTE/CANCELADA/CONFIRMADA
PAGAMENTO_CONFIRMADO	BOOLEAN	NOT NULL	TRUE SE PAGAMENTO CONFIRMADO
CODIGO	VARCHAR(100)	NOT NULL	CODIGO ÚNICO DO INGRESSO - QR CODE

Nome da tabela: PRESENÇA			
Registro da presença do participante (por inscrição).			
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_PRESENÇA	VARCHAR(36)	NOT NULL	IDENTIFICADOR DA PRESENÇA
UUID_INSCRIÇÃO	VARCHAR(36)	NOT NULL	FK INSCRIÇÃO
DATA_HORA	DATETIME	NOT NULL	DATA/HORA DO REGISTRO
METODO_VERIFICAÇÃO	VARCHAR(20)	NOT NULL	QR CODE/ LOGIN/ MANUAL
LOCAL_PALESTRA_ID	VARCHAR(36)	NULL	FK PALESTRA

Nome da tabela: CERTIFICADO

Certificados emitidos para participantes (por inscrição).

Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_CERTIFICADO	VARCHAR(36)	NOT NULL	IDENTIFICADOR DA PRESENÇA
UUID_INSCRIÇÃO	VARCHAR(36)	NOT NULL	FK INSCRIÇÃO
DATA_EMISSÃO	DATETIME	NOT NULL	DATA/HORA DE EMISSÃO
URL_ARQUIVO	VARCHAR(500)	NULL	LINK PARA DOWNLOAD
MODELO	VARCHAR(100)	NULL	IDENTIFICADOR DO TEMPLATE USADO

Nome da tabela: NOTIFICAÇÃO

Mensagens/alertas enviados aos inscritos ou público.

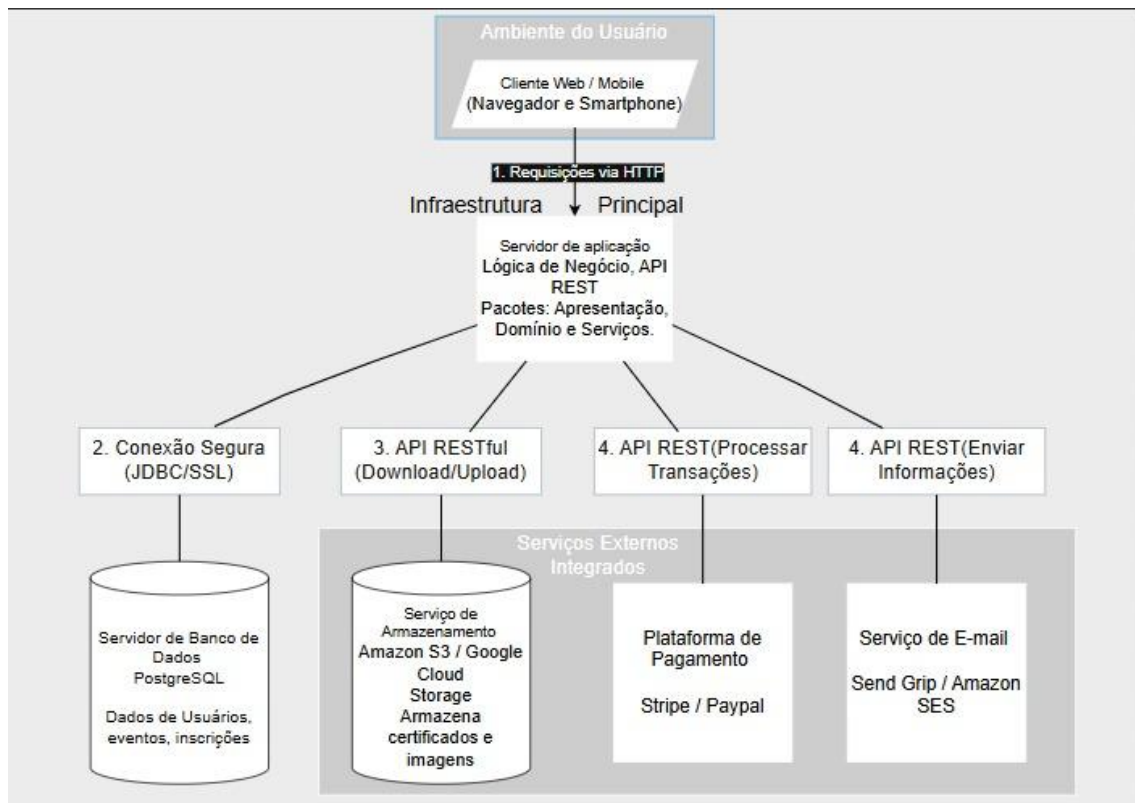
Nome da coluna	Tipo de dado	Nullabe	Descrição da coluna
UUID_NOTIFICAÇÃO	VARCHAR(36)	NOT NULL	IDENTIFICADOR DA PRESENÇA
UUID_EVENTO	VARCHAR(36)	NULL	FK INSCRIÇÃO
TIPO	VARCHAR(20)	NOT NULL	EMAIL/PUSH/SMS
TITULO	VARCHAR(255)	NULL	TITULO DO ASSUNTO
MENSAGEM	TEXT	NOT NULL	CONTEÚDO DA NOTIFICAÇÃO
DATA_ENVIO	DATETIME	NULL	QUANDO FOI ENVIADA

2.4 Visão da Implantação

Uma descrição da visão da implantação da arquitetura que descreve os diversos nós físicos para as mais típicas configurações de plataforma. Também descreve a alocação de tarefas (a partir da Visão do Processo) para os nós físicos.

Esta seção é organizada por configuração de rede física; cada configuração é ilustrada por um diagrama de implementação, seguido de um mapeamento dos processos para cada processador.

Nome do Diagrama: Visão da Implantação



Link:

<https://drive.google.com/file/d/1WVoCFfDrM8CPwUoEkgpAKKgVki6ICgL8/view?usp=sharing>

2.5.1 Cliente Web/Mobile

Os usuários do sistema (Organizadores, Participantes e Palestrantes) acessam a aplicação por meio de navegadores web ou dispositivos móveis conectados à Internet. A interface é totalmente responsiva, permitindo a utilização em computadores, tablets e smartphones. Os clientes comunicam-se com o servidor de aplicação por meio do protocolo **HTTPS**, garantindo a segurança e a integridade das informações transmitidas.

2.5.2 Servidor de Aplicação

O servidor de aplicação é hospedado em um ambiente **cloud**, responsável por processar as requisições dos usuários, aplicar as regras de negócio e coordenar as operações do sistema. Nesta camada são executadas as funcionalidades principais do, como autenticação, gerenciamento de eventos, inscrições, emissão de certificados e envio de notificações. O servidor de aplicação é implementado sobre arquitetura **cliente/servidor**, hospedando os pacotes **Apresentação**, **Serviços de Negócios** e **Domínio**. A comunicação com o banco de dados e serviços externos é feita por meio de APIs REST e conexões seguras.

2.5.3 Servidor de Banco de Dados

O **Servidor de Banco de Dados** armazena todas as informações essenciais do sistema, incluindo dados de usuários, eventos, inscrições, certificados, palestrantes e notificações. Ele é baseado em um **SGBD relacional PostgreSQL**, com suporte a backup automático e replicação para garantir disponibilidade e integridade. A comunicação entre o servidor de aplicação e o banco de dados ocorre através do protocolo **JDBC/SSL**, assegurando o sigilo e a consistência dos dados.

2.5.4 Serviço de E-mail

Utiliza serviços externos de e-mail, como **SendGrid** ou **Amazon SES**, para o envio de notificações automáticas aos usuários. Esses serviços são responsáveis por disparar confirmações de inscrição, lembretes de eventos e

certificados digitais. A integração é feita via **API REST**, autenticada por **chaves de acesso seguras**, garantindo a confiabilidade do envio.

2.5.5 Plataforma de Pagamento

A aplicação se integra com plataformas de pagamento como **PayPal** ou **Stripe** para processar inscrições pagas em eventos. A comunicação com o gateway de pagamento é feita por meio de **requisições HTTPS** autenticadas, seguindo padrões de segurança PCI DSS. O servidor de aplicação do envia as informações de cobrança e recebe o status da transação, atualizando automaticamente as inscrições no sistema.

2.5.6 Serviço de Armazenamento de Arquivos

O sistema utiliza um serviço de **armazenamento em nuvem**, como o **Amazon S3** ou **Google Cloud Storage**, para guardar os certificados PDF e imagens dos eventos. Esse serviço é acessado pelo servidor de aplicação através de APIs RESTful e garante alta disponibilidade, segurança e escalabilidade no gerenciamento de arquivos.

2.5 Qualidade

A arquitetura de software suporta os requisitos de qualidade, conforme estipulado na Especificação:

1. A interface com o usuário deve ser compatível com navegadores modernos e funcionar corretamente em desktops, tablets e smartphones.
2. A interface do sistema foi projetada para facilidade de utilização e deve ser apropriada para uma comunidade de usuários com experiência básica em tecnologia, sem necessidade de treinamento adicional.
3. Cada recurso do sistema deve ter ajuda on-line interna para o usuário. A Ajuda On-line deve incluir instruções passo a passo sobre como realizar login, se inscrever em eventos, validar presença e emitir certificados. A Ajuda On-line também deve incluir definições para termos como “QR Code”, “certificado digital” e “evento pago”.
4. O sistema deve estar disponível 24 horas por dia, 7 dias por semana. Não deve haver mais que 4% de tempo de inatividade mensal.
5. O Tempo Médio Entre Falhas (MTBF) do sistema deve exceder 300 horas, garantindo confiabilidade operacional para organizadores e participantes.

3 Referências

As referências aplicáveis são:

1. PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de Software: Uma Abordagem Profissional**. 8. ed. Porto Alegre: AMGH, 2016.
→ Base conceitual para o modelo 4+1, camadas de arquitetura e casos de uso.
2. SOMMERVILLE, Ian. **Engenharia de Software**. 10. ed. São Paulo: Pearson, 2019.
→ Fundamenta a definição de requisitos, arquitetura lógica e visão de implantação.
3. LARMAN, Craig. **Utilizando UML e Padrões: Uma Introdução à Análise e ao Projeto Orientados a Objetos**. 3. ed. Porto Alegre: Bookman, 2007.
→ Referência principal para diagramas de classes, pacotes e casos de uso.
4. MICROSOFT. **.NET Architecture Guides – Architectural Styles**. Disponível em: <https://learn.microsoft.com/en-us/dotnet/architecture/guide/>. Acesso em: 20 out. 2025.
→ Material técnico sobre padrões arquiteturais, camadas e boas práticas de desenvolvimento web.
5. ALURA. **Arquitetura de Software: Monólitos, Microsserviços e Integrações**. Disponível em: <https://www.alura.com.br/artigos/arquitetura-software-monolitos-microsservicos>. Acesso em: 20 out. 2025.