

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS ARARANGUÁ
CENTRO DE CIÊNCIAS, TECNOLOGIAS E SAÚDE (CTS)
CURSO TECNOLOGIAS DA INFORMAÇÃO E COMUNICAÇÃO

Rafael Rodrigues Ribeiro Júnior

PinArt: Sistema web para integração de eventos e usuários

Araranguá
2026

Rafael Rodrigues Ribeiro Júnior

PinArt: Sistema web para integração de eventos e usuários

Trabalho de Conclusão de Curso submetido ao curso de tecnologias da informação e comunicação do Campus Araranguá da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Bacharel em Tecnologias da Informação e Comunicação

Orientador(a): Prof. Cristian Cechinel, Dr.
Coorientador(a): Felipe Vieira Roque, Dr.

Araranguá
2026

Ribeiro Junior, Rafael Rodrigues
PinArt: Sistema web para integração de eventos e usuários / Rafael Rodrigues Ribeiro Junior ; orientador, Cristian Cechinel, coorientador, Felipe Vieira Roque, 2026.
65 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Campus Araranguá, Graduação em Tecnologias da Informação e Comunicação, Araranguá, 2026.

Inclui referências.

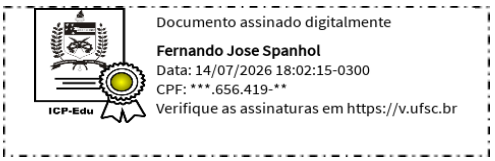
1. Tecnologias da Informação e Comunicação. 2. Eventos. 3. Plataformas Digitais. 4. Desenvolvimento Web. I. Cechinel, Cristian. II. Roque, Felipe Vieira. III. Universidade Federal de Santa Catarina. Graduação em Tecnologias da Informação e Comunicação. IV. Título.

Rafael Rodrigues Ribeiro Junior

PinArt: Sistema web para integração de eventos e usuários

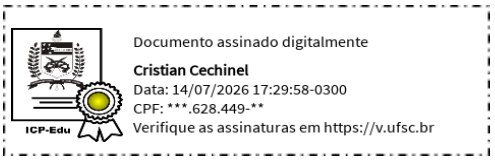
Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do título de Bacharel e aprovado em sua forma final pelo Curso Tecnologias da Informação e Comunicação.

Araranguá, 07 de Julho de 2026.



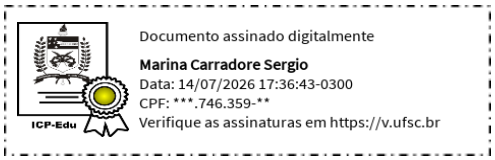
Coordenação do Curso

Banca examinadora



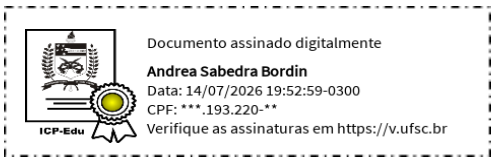
Prof. Cristian Cechinel, Dr.

Orientador(a)



Prof.ª Marina Carradore Sérgio, Dr.ª

Universidade Federal de Santa Catarina



Prof.ª Andréa Bordin, Dr.ª

Universidade Federal de Santa Catarina

Araranguá, 2026

RESUMO

A organização e a divulgação de eventos enfrentam desafios para atingir públicos específicos, gerando uma lacuna de conexão entre organizadores e frequentadores. Diante disso, este trabalho tem como objetivo apresentar o desenvolvimento da plataforma PinArt, um sistema voltado para a integração de usuários a eventos geolocalizados. Como principal contribuição científica, o estudo propõe uma arquitetura web específica voltada para a centralização e distribuição de informações de eventos. A pesquisa foi conduzida sob a diretriz da *Design Science Research Methodology* (DSRM), partindo da pesquisa bibliográfica para a concepção e desenvolvimento do artefato. A avaliação experimental do protótipo funcional foi realizada com 24 participantes de perfil jovem (com ensino superior em andamento ou concluído), empregando métricas de mapas de calor de cliques e profundidade de rolagem (scroll) nas telas do sistema. Os resultados dessa avaliação demonstram que a interface viabiliza a filtragem por categorias e a visualização interativa de forma fluida, apresentando indícios positivos de que pode facilitar a tomada de decisão do usuário. Conclui-se que a plataforma desponta como uma ferramenta com potencial viabilidade para apoiar o acesso à cultura e ao entretenimento, indicando um caminho favorável para a promoção da interação social mediada pela tecnologia.

Palavras-chave: eventos; plataformas digitais; desenvolvimento web.

ABSTRACT

The event organization and promotion face challenges in reaching specific audiences, creating a disconnect between organizers and attendees. Against this backdrop, this study presents the development of the PinArt platform, a system designed to connect users with geolocated events. As its primary scientific contribution, the study proposes a specialized web architecture for centralizing and distributing event information. The research followed the Design Science Research Methodology (DSRM), proceeding from a literature review to the design and development of the artifact. An experimental evaluation of the functional prototype was conducted with 24 young participants (currently pursuing or having completed higher education), utilizing metrics such as click heatmaps and scroll depth across the system's screens. The evaluation results demonstrate that the interface enables seamless category filtering and interactive visualization, showing positive indications that it can facilitate user decision-making. The study concludes that the platform emerges as a potentially viable tool for supporting access to culture and entertainment, pointing to a promising path for fostering technology-mediated social interaction.

Keywords: events; digital platforms; web development.

LISTA DE FIGURAS

Figura 1 - Interface inicial do Sympla 19
Figura 2 - Interface da plataforma Doity 20
Figura 3 - Interface do EventBrite 21
Figura 4 - Etapas da DSRM 26
Figura 5 - Fases do desenvolvimento Iterativo e Incremental 30
Figura 6 - Modelo Conceitual do Banco de Dados 39
Figura 7 - Modelo Lógico do Banco de Dados 42
Figura 8 - Tela de Login 46
Figura 9 - Tela de Cadastro de Usuário 46
Figura 10 - Tela principal de listagem e busca de eventos 48
Figura 11 - Visualização dos eventos via Mapa Interativo 49
Figura 12 - Interface de criação de eventos (Parte 1) 51
Figura 13 - Interface de criação de eventos (Parte 2) 51
Figura 14 - Interface de detalhes do evento e alternância do estado de inscrição 53
Figura 15 - *Dashboard* com informações de usuários 56
Figura 16 - Mapa de calor de cliques na interface principal da plataforma PinArt 57
Figura 17 - Profundidade de rolagem na página de detalhes do evento 58

LISTA DE QUADROS

Quadro 1 - Comparação de ferramentas semelhantes com o PinArt 21
Quadro 2 - Requisitos Funcionais do Sistema 36
Quadro 3 - Requisitos Não Funcionais do Sistema 37
Quadro 4 - Regras de Negócio do Sistema 38
Quadro 5 - Detalhamento das Sprints 44
Quadro 6 - Mapeamento de obrigatoriedade e restrições do modelo Evento 52

LISTA DE ABREVIATURAS E SIGLAS

1FN	Primeira Forma Normal
Abrape	Associação Brasileira dos Promotores de Eventos
ACID	Atomicity, Consistency, Isolation, Durability
APIs	Application Programming Interfaces
CEP	Código de Endereçamento Postal (CEP)
CI/CD	Continuous Integration/Continuous Deployment
CPF	Cadastro de Pessoa Física
CSRF	Cross-Site Request Forgery
CSS	Cascading Style Sheets
DOM	Document Object Model
DSR	Design Science Research
DSRM	Design Science Research Methodology
DRY	Don’t Repeat Yourself
FK	Foreign Key
GPS	Global Positioning System
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
MER	Modelo Entidade-Relacionamento
MVT	Model-View-Template
NIST	National Institute of Standards and Technology
ORM	Object Relational Mapper
PaaS	Platform as a Service
PBKDF2	Password-Based Key Derivation Function 2
PIB	Produto Interno Bruto
SQL	Structured Query Language
SGBD	Sistema Gerenciador de Banco de Dados
TICs	Tecnologias da Informação e Comunicação
VS Code	Visual Studio Code
XSS	Cross-site Scripting

SUMÁRIO

1 INTRODUÇÃO..... 12
1.1 Problemática..... 12
1.2 Objetivos..... 14
1.2.1 Objetivo Geral..... 14
1.2.2 Objetivos Específicos..... 14
1.3 Justificativa..... 14
2 FUNDAMENTAÇÃO TEÓRICA..... 16
2.1 Definição geral sobre eventos..... 16
2.2 Importância dos Eventos na Sociedade Atual..... 17
2.3 Eventos e a Tecnologia..... 17
2.4 Ferramentas semelhantes..... 18
2.4.1 Sympla..... 18
2.4.2 Doity..... 19
2.4.3 EventBrite..... 20
2.4.4 Comparação entre as ferramentas e o PinArt..... 21
2.5 Estrutura do Desenvolvimento Web..... 22
2.5.1 Backend..... 22
2.5.2 Frontend..... 23
2.5.3 Banco de Dados..... 24
2.5.4 Versionamento..... 25
3 METODOLOGIA..... 26
3.1 Identificação do Problema e Motivação..... 27
3.2 Definição dos Objetivos..... 28
3.3 Desenvolvimento do Artefato..... 28
3.3.1 Processo de desenvolvimento iterativo e incremental..... 29
3.3.2 Scrum..... 30
3.3.3 Engenharia de Requisitos e Modelagem de Dados..... 31
3.3.4 Ambiente de Desenvolvimento e Tecnologias Base..... 31
3.3.5 Integração Geográfica e APIs de Localização..... 32
3.3.6 Versionamento..... 33
3.4 Demonstração..... 33
3.5 Validação..... 34
3.6 Comunicação..... 34
4 RESULTADOS E DISCUSSÕES..... 36
4.1 Requisitos..... 36
4.1.1 Funcionais..... 36
4.1.2 Não Funcionais..... 37
4.1.3 Regras de Negócio..... 37
4.2 Modelagem Banco de Dados..... 38

4.2.1 Modelo Conceitual..... 39
4.2.1.1 Herança e Especialização de Usuários.....40
4.2.1.2 Localização..... 40
4.2.1.3 Gestão de Eventos.....40
4.2.1.4 Interações do Participante (Inscrição e Avaliação)..... 41
4.2.1.5 Arquivos e Comunicação..... 41
4.2.2 Modelo Lógico..... 42
4.3 A Aplicação..... 44
4.3.1 Cadastro e Autenticação..... 45
4.3.2 Feed, Busca de Eventos e Mapa.....48
4.3.3 Criação de Eventos, Validação e Persistência de Dados.....50
4.3.4 Inscrição, Presença e Conexão com Eventos.....53
4.4 Validação da Plataforma..... 54
4.4.1 Verificação de Requisitos e Regras de Negócio..... 55
4.4.2 Análise Empírica de Interação.....55
4.5 Síntese dos Resultados e Resposta à Questão de Pesquisa..... 59
5 CONSIDERAÇÕES FINAIS..... 60
REFERÊNCIAS..... 62

1 INTRODUÇÃO

Nos últimos anos, o avanço das tecnologias web transformou significativamente a forma como as pessoas se comunicam, consomem informações e interagem socialmente (Oliveira, et al. 2018). Nesse contexto, o setor de eventos tem se beneficiado do ambiente digital, que possibilita maior alcance, engajamento e integração entre organizadores e participantes. Plataformas online de gestão e divulgação de eventos têm se tornado ferramentas essenciais para promover conexões, facilitar o acesso a informações e otimizar processos de inscrição, divulgação e interação. Diante dessa realidade, o desenvolvimento de sistemas web voltados à conexão de eventos surge como uma solução inovadora para aproximar pessoas com interesses em comum, fortalecer comunidades e impulsionar experiências presenciais e virtuais de forma dinâmica e eficiente (Rodrigues, 2018).

A localização geográfica desempenha um papel decisivo na tomada de decisão sobre o comparecimento a um evento. Nesse sentido, a integração de tecnologias de mapeamento interativo, conhecidas na literatura como Serviços Baseados em Localização, surge como um recurso fundamental. Segundo Küpper (2005), esses serviços permitem uma busca mais contextualizada e visual, filtrando informações de acordo com a posição espacial do usuário e reduzindo o esforço cognitivo necessário para encontrar atividades em regiões de interesse.

1.1 Problemática

A evolução contínua das Tecnologias da Informação e Comunicação (TICs) têm impulsionado a digitalização da maioria dos serviços, resultando em um aumento significativo de alternativas digitais para os usuários conectados. Não obstante, no que concerne à divulgação de eventos, particularmente aqueles de natureza gratuita, as soluções atualmente disponíveis demonstram-se frequentemente insatisfatórias (Hendrickson, 2012). A identificação de eventos em uma localidade específica constitui uma tarefa complexa, e essa dificuldade é amplificada quando o interesse se estende a outras regiões geográficas.

Neste contexto, a dispersão de informações e a ausência de ferramentas integradas para o cadastro e a divulgação centralizada representam desafios

persistentes na busca e no compartilhamento de eventos. Embora plataformas de redes sociais ofereçam mecanismos de divulgação, estas nem sempre proporcionam a estrutura e o controle adequados para organizadores locais e para usuários que buscam uma experiência de descoberta otimizada.

A descoberta de eventos está diretamente atrelada a parâmetros específicos de busca, como localidade, horário e segmento de interesse. Nos métodos de busca atuais, encontrar uma atividade pode ser uma tarefa simples quando o usuário consome conteúdos de sua própria "bolha" social ou restringe a pesquisa à sua região de residência. Contudo, nota-se uma significativa dificuldade na obtenção de informações centralizadas quando há a necessidade de explorar eventos em diferentes cidades ou buscar por categorias inéditas de entretenimento, cenário que justifica a necessidade de novas abordagens tecnológicas para a conexão do público (Malkin, 2015).

Embora existam soluções consolidadas em redes sociais, essas plataformas operam sob lógicas de personalização que frequentemente prejudicam a descoberta orgânica. Conforme aponta Bozdag (2013), as plataformas online contemporâneas atuam como filtros algorítmicos, priorizando o engajamento e confinando os usuários em "bolhas de filtro" ou "câmaras de eco". Esses algoritmos de recomendação reduzem a descoberta acidental e positiva de novos conteúdos, limitando severamente as chances de um usuário encontrar eventos que fujam do seu padrão de consumo habitual.

Atualmente, a busca por atividades relevantes frequentemente exige que o público consulte múltiplas plataformas simultaneamente. Esse cenário de fragmentação, somado à escassez e à ambiguidade das informações disponibilizadas nesses meios, gera frustração e impacta negativamente a experiência do usuário durante o processo de descoberta (Rodrigues, 2018).

Dessa forma, emerge a seguinte questão de pesquisa: **Como a implementação de uma plataforma web baseada em geolocalização pode viabilizar a centralização e a visualização de eventos de forma intuitiva para os usuários finais?**

1.2 Objetivos

Esta seção apresenta os objetivos que norteiam o desenvolvimento deste trabalho, subdivididos em objetivo geral e objetivos específicos.

1.2.1 Objetivo Geral

Desenvolver uma plataforma web baseada em geolocalização para possibilitar a visualização de eventos por meio de um mapa interativo aos usuários finais.

1.2.2 Objetivos Específicos

- Realizar a elicitação e a especificação dos requisitos funcionais, requisitos não funcionais e das regras de negócio;
- Estruturar uma arquitetura de dados web focada na centralização, categorização e distribuição de informações culturais e de entretenimento;
- Implementar funcionalidades de interação e serviços de geolocalização e mapeamento;
- Avaliar comportamentos de usuários testes na aplicação;

1.3 Justificativa

Os eventos desempenham um papel fundamental não apenas como atividades indispensáveis de integração social (Shaw *et al.*, 2009), mas também como motores de desenvolvimento econômico. Para dimensionar essa relevância, dados da Associação Brasileira dos Promotores de Eventos (Abrape) indicam que, em 2023, o setor gerou um faturamento anual expressivo de R$ 291,1 bilhões. No entanto, para que esse potencial econômico e social seja plenamente aproveitado, desde grandes produções até iniciativas locais e gratuitas, é crucial que o público tenha ferramentas eficientes para acessar a programação cultural disponível.

Apesar dessa relevância, o cenário atual de divulgação esbarra nas limitações impostas pelas redes sociais convencionais. A fragmentação das

informações e a dependência de algoritmos que confinam os usuários em suas próprias "bolhas" reduzem drasticamente o alcance orgânico dos organizadores locais e dificultam a descoberta por parte do público. Nesse sentido, Rodrigues (2018) destaca que a criação de plataformas específicas para eventos, apoiadas em conceitos de categorização e integração de dados, é uma medida eficaz para mitigar essas frustrações e melhorar significativamente a experiência do usuário durante o processo de busca.

Do ponto de vista tecnológico e acadêmico, o desenvolvimento de aplicações web modernas apresenta características essenciais para sanar essa lacuna de comunicação. Conforme apontam Oliveira *et al.* (2018), essas tecnologias garantem benefícios como a recuperação ágil da informação, a recomendação inteligente de conteúdos e o acesso independente da localização. Assim, este trabalho justifica-se cientificamente pela proposta de uma arquitetura web específica, baseada em geolocalização, projetada para centralizar informações dispersas. A plataforma PinArt apresenta-se não apenas como uma solução prática para a conexão entre organizadores e frequentadores, mas também como um modelo de estruturação de dados georreferenciados que otimiza a visibilidade cultural.

A partir dessa contextualização, a próxima seção apresenta a fundamentação teórica, abordando conceitos relacionados ao desenvolvimento web, às tecnologias utilizadas e aos modelos de sistemas de gestão de eventos, servindo como base metodológica para o desenvolvimento da aplicação proposta.

2 FUNDAMENTAÇÃO TEÓRICA

A criação do PinArt vai utilizar de fundamentos consolidados nas áreas da computação, além de pesquisas sobre eventos para entender a melhor abordagem em se trabalhar com eles. Esta seção será a fundamentação teórica para sustentar o projeto e será dividida em cinco subseções: a primeira explica a definição geral dos eventos, a segunda explica a importância dos eventos na sociedade atual, a terceira demonstra como a tecnologia pode influenciar positivamente nesses eventos, a quarta é uma comparação de ferramentas semelhantes atualmente no mercado e por último é o detalhamento das tecnologias utilizadas para o desenvolvimento da aplicação web.

2.1 Definição geral sobre eventos

Um evento caracteriza-se como o encontro de pessoas e/ou sociedades, realizado em locais e datas pré-definidos, com o intuito de celebrar acontecimentos especiais. Essas ocasiões podem abranger diferentes temáticas, tais como religiosa, esportiva, social e familiar, entre outras (Zanella, 2003 apud Coutinho, 2010).

De acordo com Danieli (2024), os eventos são formas de conectar indivíduos, proporcionando experiências marcantes que permanecem na memória, gerando satisfação e impactando positivamente aqueles que as vivenciam. Além disso, consistem em acontecimentos voltados à comunicação de mensagens a algum público-alvo, ocorrendo por meio de formatos como shows, excursões, peças teatrais e exposições, entre outros (Armstrong; Kotler, 2003).

Cada evento possui características próprias, como número de participantes, data específica, público-alvo e propósito. Para otimizar os resultados, é fundamental que a organização compreenda profundamente essas particularidades (Allen, 2003 apud Coutinho, 2010). Por fim, ressalta-se que os eventos não ocorrem de maneira espontânea; eles exigem a execução de diversos processos, que se iniciam na definição do tema e se estendem até a sua divulgação (Carneiro; Fontes, 1997).

2.2 Importância dos Eventos na Sociedade Atual

Embora a realização de eventos gere múltiplas consequências para uma região, Small (2007) destaca dimensões altamente positivas que essas atividades proporcionam às comunidades anfitriãs. Entre os benefícios identificados pela pesquisadora, evidencia-se o fortalecimento da identidade e da coesão comunitária, uma vez que os festivais estimulam o sentimento de pertencimento e o orgulho dos residentes ao exporem a singularidade cultural da sua região. Além disso, os eventos atuam como motores de crescimento e desenvolvimento local, viabilizando novas oportunidades de emprego, a aquisição de habilidades pela população e a captação de recursos para grupos da comunidade. Sob a ótica individual, a autora ressalta que essas iniciativas ampliam significativamente as oportunidades de entretenimento e socialização, promovendo a interação interpessoal, a recepção de visitantes e o fortalecimento dos laços familiares.

Carneiro e Fontes (1997) complementam dizendo que os eventos contribuem não apenas para o estreitamento de laços sociais, mas também no conhecimento como um todo sobre determinado assunto, eles também contribuem para o aumento de consumo de determinados produtos ou serviços.

Um outro ponto interessante dos eventos é que eles nunca são os mesmos, os acontecimentos experienciados são únicos. Cardoso (2013) continua, eventos devem ser preparados por profissionais, pois são muito importantes e é necessário a criação de boas estratégias para o sucesso dos eventos. Estudos da Abrape mostram que o Produto Interno Bruto (PIB) do setor de eventos está 2,5% acima da média do Brasil e acima dos principais setores da economia.

Watt (2009) comenta que para os participantes o evento é muito importante, pois se não, não faria sentido estar presente e para o espectador uma oportunidade muito emocionante, ou seja, os organizadores devem lembrar mesmo que pra ele seja outro evento ao longo de muitos, para os outros pode ser o primeiro.

2.3 Eventos e a Tecnologia

Batista (2014) cita em seu trabalho um estudo realizado pela plataforma EventBrite, no qual mostra que aproximadamente 94% dos organizadores de

eventos, estão nas redes sociais para divulgar seus eventos, pois a tecnologia consegue proporcionar uma facilidade na integração das pessoas com o produto final. Ele também cita que é indispensável a relação com o cliente, antes, durante e depois dos eventos, garantindo uma melhor experiência ao cliente e também receber feedbacks sobre o desempenho do evento.

De acordo com Vasconcelos (2023), a comunicação é essencial para o melhor rendimento de um evento e com as TICs o trabalho é muito facilitado, pela dimensão que a comunicação consegue chegar. Além do tempo que a informação chega pela Internet, conseguindo atrair mais pessoas em pouco tempo. Alves (2023) complementa, as TICs e a comunicação são fundamentais para um bom desempenho dos eventos, elas utilizadas de maneira agradável retém mais o público e torna os eventos mais atrativos.

As redes sociais são um dos principais métodos da comunicação atualmente, conseguindo alcançar muitos públicos diferentes e também um público-alvo específico. Outro ponto importante para a comunicação são os *websites*, sendo o primeiro contato dos eventos com os espectadores (Alves, 2023).

Hoje em dia, com a tecnologia a divulgação dos eventos ficou mais fácil, pois a informação consegue chegar mais rápido aos interessados, além disso, o tempo é economizado, organização fica melhor e é possível armazenar coisas de uma maneira mais tranquila (Clericuzi, 2021).

2.4 Ferramentas semelhantes

Essa seção detalha informações relevantes sobre algumas ferramentas semelhantes, também mostra pontos positivos e negativos em cada um deles, e por fim apresenta as melhorias e inovações que o PinArt se propõe a fazer.

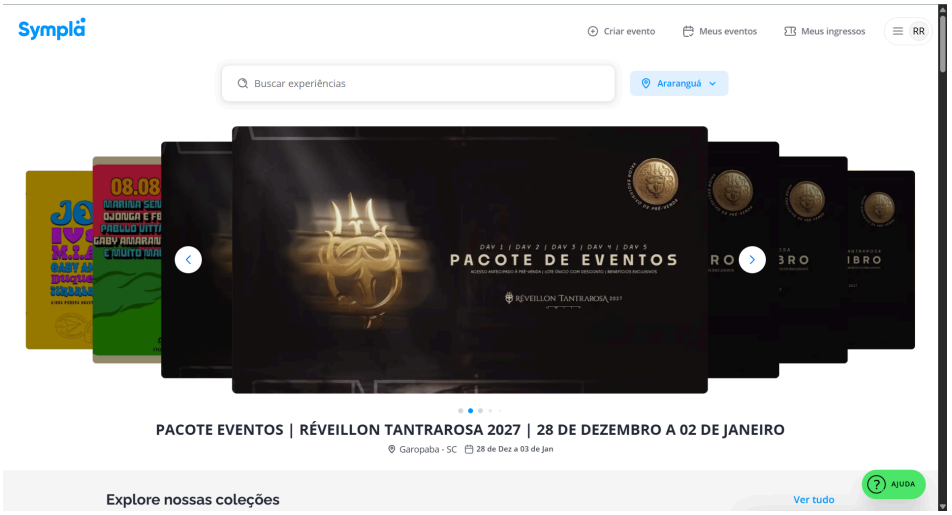
2.4.1 Sympla

O Sympla[1] é um site para venda de ingressos de eventos diversos, em todo o Brasil. É possível buscar por eventos por tipos, valores, localização, data e entre outros, qualquer pessoa pode cadastrar um evento, o site cobra uma taxa entre 10%

[1] https://www.sympla.com.br/

a 15% por ingresso vendido, não possui um mapa interativo para ver com facilidade o local do evento.

Figura 1 - Interface inicial do Sympla



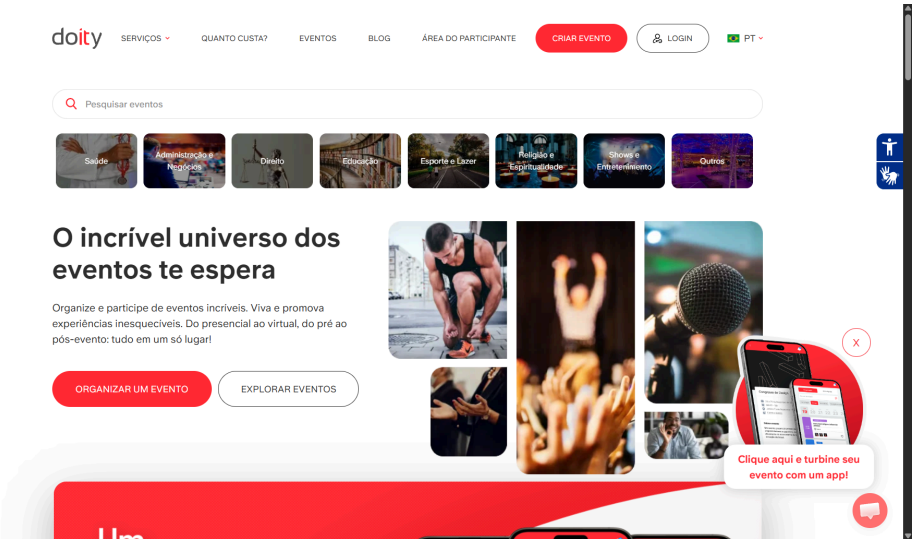
Fonte: Sympla (2026)

2.4.2 Doity

Doity[2] é uma plataforma completa para gestão de eventos, o organizador consegue deixar um site dentro da plataforma, descrevendo mais o evento para os usuários, é possível cadastrar eventos gratuitos com limite de 500 participantes, a plataforma cobra 10% em ingressos vendidos, conseguem emitir certificados, integrações com serviços de análises e emissão de notas. Embora a plataforma apresenta funcionalidades robustas para a gestão de eventos, observam-se inconsistências de *design* em algumas de suas seções. A ausência de padronização na paleta de cores compromete a unidade visual da interface, transmitindo uma percepção de desorganização ao usuário.

[2] https://doity.com.br/

Figura 2 - Interface da plataforma Doity



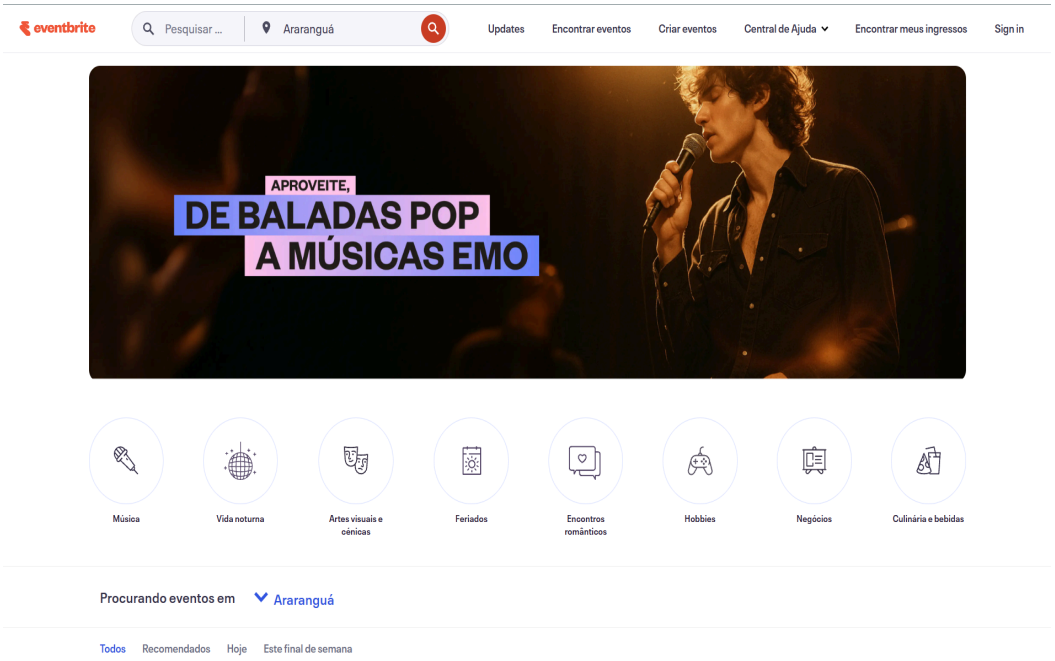
Fonte: Doity (2026)

2.4.3 EventBrite

O Eventbrite[3] atua como uma plataforma em escala global para o cadastro de eventos, disponibilizando interfaces distintas para organizadores e participantes. Contudo, a estrutura dos planos de publicação apresenta pouca clareza, dificultando a compreensão precisa dos valores cobrados. O sistema permite que os usuários acompanhem os perfis dos organizadores; no entanto, durante a navegação, identificaram-se falhas de consistência no sistema, resultando na alternância inesperada entre os idiomas inglês, português e espanhol. Além disso, na página de detalhes da atividade, nota-se que o *design* não prioriza a localização geográfica do evento, o que pode comprometer a experiência de descoberta por parte do usuário.

[3] https://www.eventbrite.com.br/

Figura 3 - Interface do EventBrite



Fonte: EventBrite (2026)

2.4.4 Comparação entre as ferramentas e o PinArt

Quadro 1 - Comparação de ferramentas semelhantes com o PinArt

Plataforma	**Objetivo**	**Pontos Fortes**	**Pontos Fracos**
Sympla	Venda de Ingressos	Criação dinâmica e descentralizada (qualquer usuário pode criar eventos).	Ausência de verificação de qualidade dos eventos, diminuindo a confiabilidade.
Doity	Gestão de eventos	Plataforma completa, páginas dinâmicas, análises, integração com	Inconsistência na interface (mudanças bruscas de cores e padrões

		notas fiscais e plano gratuito atrativo.	durante a navegação).
EventBrite	Venda de Ingressos	Abrangência global, focado em eventos de grande porte.	Visualização de localização insatisfatória; mistura de idiomas durante o uso.
PinArt	Divulgação de Eventos	Mapa interativo, padronização visual consistente e sistema de avaliações/comentários para atestar a qualidade.	

Fonte: elaborado pelo autor (2026)

2.5 Estrutura do Desenvolvimento Web

2.5.1 Backend

O backend representa a camada responsável pelo processamento lógico, pela comunicação com o banco de dados e pela disponibilização de informações para o frontend de uma aplicação web (Sommerville, 2011).

O Python é uma linguagem de programação de alto nível, interpretada e multiparadigma, que se destaca por sua legibilidade e simplicidade sintática. Essas características tornam o desenvolvimento mais ágil e reduzem a probabilidade de erros, além de facilitar a manutenção do código. Sua extensa biblioteca padrão e a grande comunidade de desenvolvedores contribuem para o rápido desenvolvimento de soluções complexas, incluindo integração com bancos de dados, serviços web e ferramentas de análise de dados (Menezes, 2010).

O Django, por sua vez, é um *framework* web baseado no padrão *Model-View-Template* (MVT), que oferece uma estrutura organizada para o

desenvolvimento de sistemas escaláveis e seguros. Ele segue o princípio "*Don't Repeat Yourself*" (DRY), que incentiva a reutilização de código e a redução de redundâncias. Além disso, o Django disponibiliza um *Object Relational Mapper* (ORM) que simplifica a comunicação com bancos de dados relacionais, abstraindo comandos *Structured Query Language* (SQL) em operações de alto nível (Bezerra Junior, 2023).

Entre os benefícios do uso do Django estão o sistema de autenticação integrado, o painel administrativo automático e os recursos de segurança incorporados, como proteção contra ataques de injeção de SQL e *cross-site scripting* (XSS). Tais recursos tornam o *framework* uma escolha sólida para o desenvolvimento de aplicações web que exigem confiabilidade, desempenho e boas práticas de engenharia de software (Django Software Foundation, 2024).

Assim, a combinação entre Python e Django no backend proporciona uma base robusta para o projeto, assegurando produtividade no desenvolvimento, clareza na organização do código e facilidade na manutenção e expansão futura do sistema (Bezerra Junior, 2023).

2.5.2 Frontend

O frontend constitui a camada de interação entre o usuário e o sistema, sendo responsável pela interface visual e pela experiência de uso das funcionalidades disponibilizadas pelo backend. Nesta etapa do desenvolvimento, foram utilizadas as linguagens *Hypertext Markup Language* (HTML), *Cascading Style Sheets* (CSS) e *JavaScript*, que, em conjunto, formam o núcleo das tecnologias da World Wide Web (Sommerville, 2011).

O HTML é a linguagem de marcação padrão utilizada para estruturar o conteúdo das páginas web. Por meio de suas tags e elementos semânticos, o HTML permite organizar textos, imagens, tabelas, formulários e outros componentes que compõem a interface de um site. Sua importância está em estabelecer a base estrutural da aplicação, garantindo acessibilidade e interoperabilidade entre diferentes navegadores e dispositivos (Mazza, 2014; Lucchesi, 2024).

O CSS é responsável pela estilização e pelo *design* visual das páginas, permitindo definir cores, fontes, espaçamentos, animações e o posicionamento dos

elementos. A utilização adequada de CSS contribui significativamente para a responsividade do sistema, assegurando que a interface se adapte de forma dinâmica a diferentes tamanhos de tela e dispositivos. Além disso, boas práticas de separação entre estrutura HTML e estilo CSS favorecem a manutenção, o isolamento de componentes e a reutilização do código-fonte (Mozilla, 2023).

O JavaScript, por sua vez, é uma linguagem de programação interpretada voltada para o comportamento dinâmico das páginas web. Com ele, é possível manipular o *Document Object Model* (DOM), validar formulários, consumir *Application Programming Interfaces* (APIs) e criar interações em tempo real. Atualmente, o *JavaScript* é uma das linguagens mais populares do mundo, sustentando o ecossistema de bibliotecas e *frameworks* como React, Vue.js e Angular. Mesmo utilizando o *JavaScript da forma pura*, ele oferece recursos suficientes para o desenvolvimento de interfaces interativas e responsivas (Flanagan, 2011).

2.5.3 Banco de Dados

O banco de dados é um dos componentes centrais de uma aplicação web, sendo responsável pelo armazenamento, organização e recuperação eficiente das informações que sustentam as funcionalidades do sistema. Neste projeto, optou-se pelo uso do *PostgreSQL*, um SGBD de código aberto, reconhecido por sua confiabilidade, robustez e conformidade com os padrões SQL (Juba; Vannahme; Volkov, 2015).

O *PostgreSQL* é um dos SGBDs mais avançados disponíveis no ecossistema open source, oferecendo suporte a transações *Atomicity, Consistency, Isolation, Durability* (ACID), o que garante integridade e consistência dos dados mesmo em operações complexas ou de alta concorrência. Além disso, ele disponibiliza mecanismos avançados de controle de acesso, replicação, particionamento e extensões que permitem a personalização e o aprimoramento das funcionalidades do banco (Juba; Vannahme; Volkov, 2015).

2.5.4 Versionamento

O Git é um sistema de controle de versão distribuído, criado por Linus Torvalds em 2005, com o objetivo de oferecer alto desempenho, confiabilidade e flexibilidade na gestão de projetos de software. Diferentemente de sistemas centralizados, o Git permite que cada desenvolvedor possua uma cópia completa do repositório, incluindo todo o histórico de commits, o que torna as operações locais mais rápidas e independentes de conexão com servidores remotos (Chacon; Straub, 2014). Os autores complementam que, entre suas principais funcionalidades estão a criação de *branches* e *merges*, que possibilitam o desenvolvimento paralelo de novas funcionalidades sem comprometer a estabilidade da versão principal do sistema.

O GitHub, por sua vez, é uma plataforma online baseada em Git que oferece recursos adicionais de colaboração, armazenamento em nuvem e automação de fluxos de desenvolvimento. Ele permite o gerenciamento de repositórios remotos, controle de permissões, registro de tarefas e bugs, *pull requests* e integração com *pipelines* de *Continuous Integration/Continuous Deployment* (CI/CD). Dessa forma, o GitHub atua como um ambiente centralizado de versionamento e colaboração, favorecendo a transparência e o controle do progresso do projeto (Loeliger; Mccullough, 2012).

A combinação entre Git e GitHub garante um processo de desenvolvimento mais organizado, seguro e rastreável, permitindo que cada modificação seja documentada e revertida quando necessário. Isso proporciona maior qualidade ao código e facilita a manutenção do sistema ao longo de seu ciclo de vida (Loeliger; Mccullough, 2012).

3 METODOLOGIA

Para a consecução dos objetivos propostos neste estudo, faz-se necessário delinear os caminhos metodológicos adotados. Quanto à sua natureza, esta pesquisa classifica-se como aplicada, uma vez que busca gerar conhecimentos voltados para a solução de problemas práticos, culminando no desenvolvimento de um artefato tecnológico funcional.

Em relação aos seus objetivos, a pesquisa possui caráter exploratório, pois visa proporcionar maior familiaridade com o problema da fragmentação da divulgação de eventos e a influência dos algoritmos na descoberta orgânica, explorando a aplicação de mapas geolocalizados como alternativa para a redução da carga cognitiva do usuário.

No que tange à abordagem, o estudo enquadra-se como uma pesquisa de natureza mista, visto que a avaliação do artefato baseia-se na coleta e análise de métricas de interação de usuários finais. Por fim, como procedimento metodológico, a pesquisa fundamenta-se no paradigma da *Design Science Research Methodology* (DSRM) proposto por Peffers *et al.* (2007), estruturando-se em seis etapas lógicas, conforme a Figura 4 e que será detalhado ao longo desta seção.

Figura 4 - Etapas da DSRM



Fonte: elaborado pelo autor baseado em Peffers *et al.* (2007)

A DSRM difere das metodologias tradicionais de pesquisa por ter um caráter eminentemente prescritivo e pragmático, cujo foco central não é apenas observar ou descrever um fenômeno, mas sim projetar e construir um "artefato" computacional utilitário para intervir em um problema real. No contexto do desenvolvimento de software, esse referencial metodológico garante que a arquitetura e as funcionalidades do sistema sejam guiadas por necessidades empíricas, evitando soluções puramente teóricas. O ciclo estruturado por Peffers *et al.* (2007) desdobra-se em um fluxo contínuo onde, inicialmente, mapeia-se a deficiência no cenário atual (Identificação do Problema e Motivação) para, então, estabelecer as metas operacionais e os requisitos da aplicação (Definição dos Objetivos para uma Solução). Com as diretrizes estabelecidas, ocorre a codificação e materialização da plataforma web (Projeto e Desenvolvimento). O artefato gerado é, em seguida, inserido em seu contexto prático de uso (Demonstração) para que o seu desempenho e o atendimento às regras de negócio sejam testados e validados (Avaliação). O processo culmina na documentação e no compartilhamento das descobertas técnicas e teóricas (Comunicação), atestando o rigor científico e a contribuição tecnológica do estudo.

3.1 Identificação do Problema e Motivação

A fase inaugural da Design Science Research visa mapear a problemática e justificar a necessidade da pesquisa. No contexto atual, a busca por atividades de entretenimento e cultura sofre com a fragmentação dos canais de comunicação. No contexto da Interação Humano-Computador (IHC) e da Arquitetura da Informação, sabe-se que a dispersão de dados em múltiplas redes sociais dificulta a localização eficiente de conteúdos. Somado a isso, análises sobre filtragem algorítmica indicam que a dependência de sistemas de personalização fechados atua como uma barreira à descoberta orgânica, restringindo os usuários a “bolhas de filtro” (Bozdag, 2013). O usuário enfrenta uma carga cognitiva excessiva ao tentar localizar eventos quando busca por atividades fora de sua região habitual ou em categorias de entretenimento inéditas. Diante desta lacuna, estabeleceu-se a motivação para o desenvolvimento

da plataforma PinArt, projetada para atuar como um ecossistema centralizado e geolocalizado, mitigando a ambiguidade e a escassez de dados na divulgação de eventos locais.

3.2 Definição dos Objetivos

No contexto desta pesquisa, os objetivos foram projetados para mitigar a fragmentação informacional e reduzir a carga cognitiva dos usuários durante a busca por eventos culturais e de entretenimento.

Diante disso, definiu-se como objetivo geral da pesquisa o desenvolvimento de uma plataforma web baseada em geolocalização, visando possibilitar a visualização e a descoberta de eventos por meio de um mapa interativo aos usuários finais.

Para que este propósito central pudesse ser alcançado de forma sistemática e o artefato fosse validado metodologicamente, estabeleceram-se os seguintes objetivos específicos, que norteiam as etapas subsequentes da DSRM:

- Realizar a elicitação e a especificação dos requisitos funcionais, não funcionais e das regras de negócio;
- Estruturar uma arquitetura de dados web focada na centralização, categorização e distribuição de informações culturais e de entretenimento;
- Implementar funcionalidades de interação e serviços de geolocalização e mapeamento;
- Avaliar os comportamentos dos usuários durante a utilização e interação com a aplicação desenvolvida.

3.3 Desenvolvimento do Artefato

O processo de desenvolvimento do artefato baseia-se em um modelo iterativo e incremental, que permite o desenvolvimento progressivo do sistema por meio de ciclos curtos de planejamento, implementação e validação. Essa abordagem foi escolhida por favorecer a adaptação contínua às necessidades do projeto e a entrega gradual de funcionalidades. Para o gerenciamento das atividades, foram

aplicados princípios da metodologia ágil Scrum, que proporcionou organização, priorização de tarefas e acompanhamento visual do progresso. Além disso, todo o processo de desenvolvimento foi orientado pelos fundamentos da engenharia de software, assegurando a aplicação de boas práticas na análise de requisitos, modelagem, implementação, testes e documentação do sistema. Para materializar esta construção técnica, a codificação ocorreu no ambiente Visual Studio Code (VS Code), adotando o *framework* Django como alicerce arquitetural da aplicação, processos estes resguardados por um rigoroso controle de versionamento, cujas especificidades tecnológicas serão detalhadas nas subseções subsequentes.

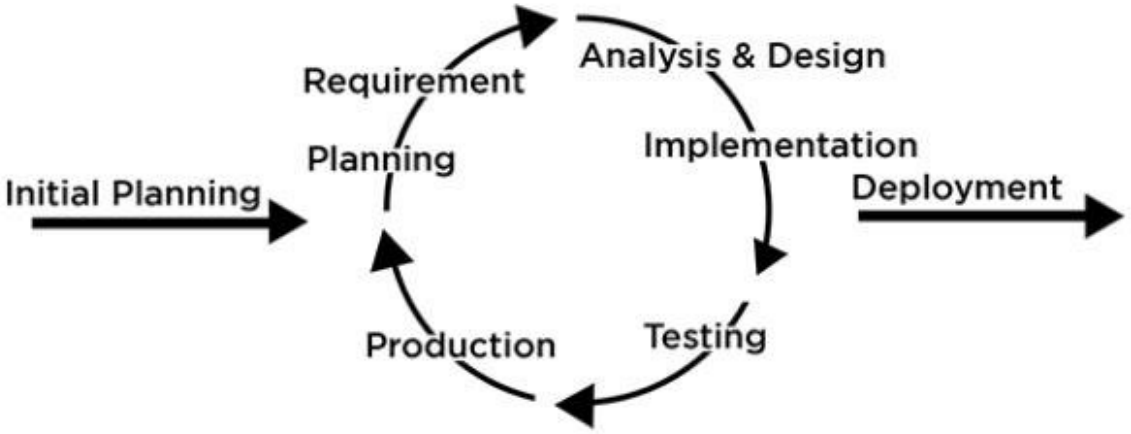
3.3.1 Processo de desenvolvimento iterativo e incremental

O desenvolvimento deste projeto seguirá o modelo iterativo e incremental, uma abordagem amplamente utilizada na engenharia de *software* contemporânea por permitir o aperfeiçoamento contínuo do sistema ao longo de ciclos sucessivos de desenvolvimento (Sommerville, 2011). Nesse modelo, o sistema não é construído integralmente de uma só vez; em vez disso, ele é desenvolvido em iterações (ou ciclos), cada uma responsável por adicionar novas funcionalidades e melhorias ao produto já existente.

Foi adotado esse modelo a fim de possibilitar avaliações constantes, ajustes de requisitos e validação parcial das entregas, o que contribui para reduzir riscos e aprimorar a qualidade do software em cada etapa do processo. Cada incremento representará uma versão mais completa e estável da aplicação, permitindo acompanhar a evolução do sistema de forma controlada e transparente.

Essa escolha metodológica se justifica pela natureza experimental e evolutiva do projeto, que envolve diferentes camadas tecnológicas (*backend*, *frontend* e banco de dados). O modelo iterativo e incremental, conforme a Figura 5, proporcionará, portanto, flexibilidade, adaptabilidade às mudanças e feedback contínuo, alinhando-se às boas práticas de desenvolvimento ágil descritas por Sommerville (2011) e Pressman e Maxim (2021).

Figura 5 - Fases do desenvolvimento Iterativo e Incremental



Fonte: Than (2020)

3.3.2 Scrum

Durante o desenvolvimento deste projeto, foi adotado o *framework* Scrum como metodologia principal para organização e gestão do processo de desenvolvimento. O Scrum foi utilizado para estruturar o trabalho de forma colaborativa e incremental, garantindo planejamento contínuo, entregas parciais e revisões frequentes do produto, em conformidade com os princípios do desenvolvimento ágil.

O Scrum organiza o trabalho em ciclos curtos e repetitivos chamados sprints, nos quais serão definidas metas específicas a serem alcançadas. Cada sprint resultará em um incremento funcional do sistema, possibilitando que o progresso seja avaliado e ajustado conforme o feedback obtido. Realizou-se reuniões regulares a cada quinze dias, as sprints *reviews*, para apresentação dos resultados, junto com a *planning* para atualizar o que vai ser feito nas próximas semanas (Schwaber; Sutherland, 2011).

Essa metodologia foi aplicada para organizar o projeto como um todo, garantindo maior transparência, colaboração entre as partes envolvidas e rapidez na resposta a mudanças de requisitos. Ao adotar o Scrum, o projeto manteve um fluxo de trabalho dinâmico e iterativo, no qual as entregas parciais serviram como base

para validações e refinamentos sucessivos. Essa abordagem foi especialmente útil para um projeto de natureza acadêmica e tecnológica, que requer flexibilidade e constante adaptação.

Assim, o uso do Scrum proporcionou um ambiente ágil e controlado, permitindo um equilíbrio entre planejamento estratégico e execução prática, assegurando que o sistema evolua de forma ordenada, eficiente e conforme as necessidades do projeto.

3.3.3 Engenharia de Requisitos e Modelagem de Dados

Para garantir que a aplicação atendesse às expectativas dos usuários e cumprisse seus objetivos técnicos e operacionais, o processo seguiu boas práticas de engenharia de requisitos, fundamentadas em autores como Sommerville (2011) e Pressman e Maxim (2021). O levantamento dos requisitos foi conduzido de forma iterativa, com validações progressivas a cada incremento do sistema.

Os requisitos funcionais do sistema serviram como base direta para a definição das histórias de usuário aplicadas nas *sprints* do Scrum. Em relação aos requisitos não funcionais, estabeleceram-se métricas e parâmetros pautados em desempenho, segurança, usabilidade e escalabilidade, garantindo uma aplicação eficiente e de fácil manutenção. Por fim, as regras de negócio foram delimitadas com base nas necessidades identificadas durante a análise do domínio e nas diretrizes associadas ao escopo do projeto.

Em paralelo à especificação de requisitos, conduziu-se o projeto arquitetural de dados. Baseando-se nas melhores práticas de modelagem de banco de dados relacionais propostas por Lima, Lopes e Nascimento (2026), elaborou-se o modelo conceitual por meio de um Diagrama Entidade-Relacionamento (DER), seguido da sua conversão para o modelo lógico. Esse procedimento garantiu a normalização estrutural e a persistência eficiente das informações necessárias para o pleno funcionamento da aplicação.

3.3.4 Ambiente de Desenvolvimento e Tecnologias Base

Para a materialização técnica do sistema, a etapa de codificação foi conduzida utilizando o VS Code como ambiente de desenvolvimento. A escolha desta ferramenta justifica-se pela sua arquitetura leve e pela vasta capacidade de integração de extensões. O uso do VS Code otimizou a escrita estruturada do código-fonte, oferecendo suporte avançado para o realce de sintaxe e depuração de erros, além de proporcionar uma interface unificada para a manipulação dos arquivos do projeto.

Como alicerce da plataforma, adotou-se o Django (versão 3.1.4), um robusto framework de desenvolvimento web de alto nível escrito na linguagem Python (versão 3.9.9). O Django foi selecionado por promover o desenvolvimento ágil e o design limpo, operando sob o padrão arquitetural MVT. Essa estrutura permitiu uma separação clara entre a lógica de negócios, a interface do usuário e o acesso aos dados. Além disso, o ORM nativo do *framework* foi fundamental para a aplicação, pois permitiu traduzir de forma fluida os modelos lógicos e conceituais do banco de dados relacional para a camada de código, assegurando a integridade e a segurança no armazenamento das informações.

3.3.5 Integração Geográfica e APIs de Localização

A funcionalidade central da plataforma PinArt baseia-se na exploração espacial de eventos, exigindo uma interface cartográfica precisa e fluida. Para renderizar os mapas interativos no *frontend*, adotou-se a biblioteca JavaScript de código aberto Leaflet.js (versão 1.9.4), acompanhada da extensão Markercluster (versão 1.4.1), utilizada para o agrupamento visual de marcadores (pins) em áreas de alta densidade. A base cartográfica exibida ao usuário é provida pelo sistema colaborativo OpenStreetMap.

Para a conversão de endereços físicos em coordenadas geográficas utilizáveis pelo mapa, o *backend* do sistema foi integrado à OpenStreetMap Nominatim API. Adicionalmente, para otimizar a experiência do usuário e mitigar erros de digitação durante o cadastro de eventos, implementou-se o consumo da ViaCEP API, um serviço de consulta que permite o preenchimento automatizado e validação dos campos de endereço a partir do Código de Endereçamento Postal

(CEP). A convergência dessas tecnologias garante agilidade na inserção de dados e precisão na sua visualização espacial.

3.3.6 Versionamento

Uma vez que os conceitos fundamentais de controle de versão foram abordados na Fundamentação Teórica, a utilização do Git e do GitHub neste projeto concentrou-se na adoção de padrões organizacionais para garantir a rastreabilidade prática do desenvolvimento. O repositório remoto foi estruturado para refletir a evolução iterativa do artefato, atuando como o ambiente centralizador das entregas ao final de cada sprint.

Para manter a legibilidade, a semântica e a clareza do histórico de alterações, o registro do código foi guiado por boas práticas de versionamento, inspiradas na especificação *Conventional Commits*. Esta abordagem consistiu na utilização de prefixos padronizados nas mensagens de commit para indicar a natureza de cada modificação. Utilizou-se, por exemplo, o prefixo *feat* para a introdução de novos requisitos funcionais (como a integração do mapa), *fix* para a correção de anomalias no código e *docs* para atualizações no arquivo de documentação (Al Hasan; Quadir; Tawhid, 2026). Essa estruturação técnica permitiu mapear com exatidão quais módulos foram desenvolvidos em cada etapa cronológica do projeto, conferindo transparência e auditoria ao processo de engenharia do software.

3.4 Demonstração

A demonstração da utilidade de um artefato na DSR exige que este seja submetido a um ambiente onde possa ser interagido. Dessa forma, o projeto superou a fase de desenvolvimento local e foi conduzido à fase de implantação em ambiente de produção. O processo de hospedagem utilizou a plataforma PythonAnywhere, caracterizada como uma solução *Platform as a Service* (PaaS).

Conforme as definições de computação em nuvem estruturadas pelo National Institute of Standards and Technology (NIST) (Mell; Grance, 2011), o modelo PaaS exime o desenvolvedor da complexidade de gerenciamento da infraestrutura subjacente, permitindo o foco exclusivo no provisionamento da aplicação. Esta

escolha viabilizou a disponibilização web pública da plataforma, possibilitando testes reais de geolocalização e cadastros de usuários e eventos de forma contínua.

3.5 Validação

Com a aplicação ativa em ambiente de produção e hospedada na plataforma PythonAnywhere, executou-se a quinta etapa da metodologia, dedicada a mensurar o quão bem o artefato suporta a solução do problema. A estratégia de validação da proposta priorizou a realização de testes de intuitividade, analisando empiricamente a interação dos usuários com a interface. Para a coleta desses dados, a ferramenta analítica Microsoft Clarity foi integrada às páginas do sistema. O recrutamento dos participantes ocorreu de forma espontânea, por meio de convites enviados via aplicativos de mensagens instantâneas, resultando em uma amostra de 24 usuários. Esse grupo apresentou um perfil jovem, composto por estudantes de graduação e profissionais já formados oriundos de diversas áreas do conhecimento, de modo a evitar um viés técnico e garantir que o sistema fosse testado por um público não restrito à área de tecnologia. O monitoramento do comportamento de navegação restringiu-se à geração de mapas de calor de cliques e à medição da profundidade de rolagem nas telas da aplicação. O objetivo dessa instrumentação foi capturar métricas sobre os padrões de interação nos eventos geolocalizados para, posteriormente, verificar se a disposição visual do mapa e a estrutura centralizada proporcionam uma experiência fluida e atendem aos critérios funcionais estabelecidos inicialmente.

3.6 Comunicação

A etapa final da DSRM, segundo Peffers et al. (2007), consiste em comunicar o problema, a sua importância e a novidade estrutural do artefato à comunidade pertinente. Neste projeto, a comunicação consolida-se em duas esferas complementares. Na esfera acadêmica, a publicização ocorre mediante a redação e defesa deste Trabalho de Conclusão de Curso, que expõe todo o arcabouço teórico e as decisões de arquitetura adotadas. Na esfera tecnológica e profissional, a comunicação dá-se pela disponibilização em código aberto do repositório no GitHub.

O código-fonte, acompanhado de sua respectiva documentação de inicialização (*README*), assegura a reprodutibilidade técnica, permitindo que outros desenvolvedores possam auditar, replicar ou expandir o ecossistema do sistema no futuro.

4 RESULTADOS E DISCUSSÕES

4.1 Requisitos

Nesta seção, são descritos os requisitos do sistema, divididos entre funcionais e não funcionais, além das regras de negócio que orientam o comportamento da aplicação. A elicitação foi baseada nas necessidades de interação entre os usuários e a plataforma de eventos.

4.1.1 Funcionais

Os Requisitos Funcionais descrevem as ações que o sistema deve ser capaz de executar, ou seja, as funcionalidades diretas oferecidas aos usuários finais. O Quadro 2 detalha os requisitos funcionais mapeados para o sistema.

Quadro 2 - Requisitos Funcionais do Sistema

Id	**Descrição do Requisito**
RF01	O sistema deve permitir o cadastro de novos usuários e a autenticação na plataforma.
RF02	O sistema deve permitir que usuários com permissão de organizador criem novos eventos, informando título, descrição, datas, capacidade, local, categorias e imagem de capa.
RF03	O sistema deve permitir o cadastro de novos locais (*inline*) durante o fluxo de criação de um evento.
RF04	O sistema deve listar os eventos disponíveis em uma página inicial (em formato de *feed/cards*).
RF05	O sistema deve fornecer uma funcionalidade de busca textual de eventos por título, descrição, local e categoria.
RF06	O sistema deve exibir um mapa interativo para a exploração geográfica e visualização espacial dos eventos.
RF07	O sistema deve permitir a filtragem de eventos por categorias específicas.
RF08	O sistema deve permitir que usuários autenticados confirmem ou cancelem sua presença em um evento.

Id	Descrição do Requisito
RF01	O sistema deve permitir o cadastro de novos usuários e a autenticação na plataforma.
RF02	O sistema deve permitir que usuários com permissão de organizador criem novos eventos, informando título, descrição, datas, capacidade, local, categorias e imagem de capa.
RF09	O sistema deve capturar a geolocalização atual do usuário (via permissão do navegador) para centralizar o mapa.

Fonte: Elaborado pelo autor (2026).

4.1.2 Não Funcionais

Os Requisitos Não Funcionais definem os atributos de qualidade do sistema, como segurança, desempenho, usabilidade e tecnologias adotadas. O Quadro 3 apresenta esses requisitos.

Quadro 3 - Requisitos Não Funcionais do Sistema

Id	Descrição do Requisito
RNF01	Segurança: As senhas dos usuários devem ser submetidas a *hashing* unidirecional utilizando o algoritmo Password-Based Key Derivation Function 2 (PBKDF2) com SHA-256 e aplicação de *salt* criptográfico.
RNF02	Segurança: O sistema deve proteger as requisições de formulários contra ataques de falsificação.
RNF03	Usabilidade: A interface deve ser responsiva e utilizar a abordagem *Mobile-First*, adaptando-se a dispositivos móveis, *tablets* e *desktops*.
RNF04	Desempenho: A filtragem de eventos na visualização do mapa deve ocorrer ao lado do cliente para evitar carregamentos de página.
RNF05	Integração: O sistema deve integrar-se em alguma API para a geocodificação automática de endereços.
RNF06	Persistência: O sistema deve utilizar um banco de dados relacional para gerenciar e garantir a integridade das conexões entre usuários e eventos.

Fonte: Elaborado pelo autor (2026).

4.1.3 Regras de Negócio

As Regras de Negócio consistem nas restrições e políticas lógicas que o sistema deve respeitar para garantir a consistência das operações,

independentemente de como a funcionalidade é implementada. O Quadro 4 sumariza essas regras.

Quadro 4 - Regras de Negócio do Sistema

Id	**Descrição da Regra de Negócio**
RN01	Coerência Cronológica: Não é permitida a criação de um evento cuja data/hora de término seja anterior ou igual à data/hora de início. (RF02)
RN02	Capacidade Válida: O valor numérico correspondente à capacidade máxima de participantes do evento não pode ser negativo. (RF02)
RN03	Exibição Temporal: Por padrão, os *feeds* e mapas do sistema devem exibir apenas eventos futuros (cuja data de início seja superior ao momento atual). (RF04 e RF06)
RN04	Unicidade de Inscrição: Um mesmo usuário não pode confirmar presença mais de uma vez em um único evento. (RF08)
RN05	Autoria Automática: O usuário autenticado que submeter o formulário de criação de evento será automaticamente associado como o "Organizador" daquele evento, sem a possibilidade de manipulação desse campo na interface. (RF02)
RN06	Acesso Restrito: A ação de confirmar presença em um evento é restrita a usuários autenticados que possuam um perfil completo cadastrado no banco de dados. (RF08)

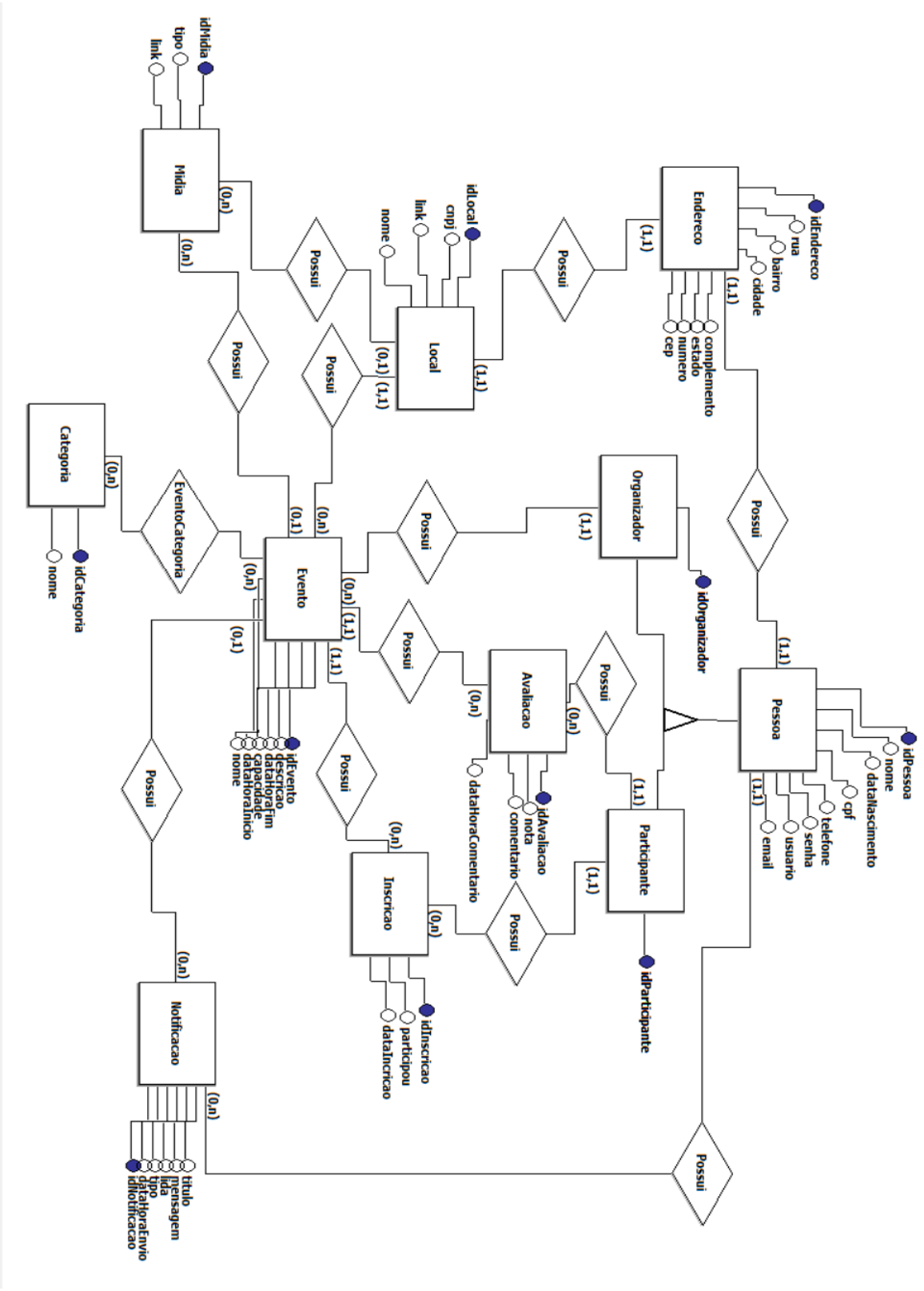
Fonte: Elaborado pelo autor (2026).

4.2 Modelagem Banco de Dados

Esta seção detalha o projeto estrutural do banco de dados, contemplando as fases de modelagem conceitual e lógica. O objetivo é demonstrar como os requisitos e as regras de negócio previamente elicitados foram traduzidos para o esquema de armazenamento relacional, apresentando as principais características de cada modelo e suas transições.

4.2.1 Modelo Conceitual

Figura 6 - Modelo Conceitual do Banco de Dados



Fonte: Elaborado pelo autor (2026).

A estrutura de dados do sistema foi projetada utilizando o Modelo Entidade-Relacionamento (MER). O diagrama ilustra as entidades que compõem o domínio da aplicação, seus respectivos atributos e as cardinalidades que regem as interações entre elas, garantindo a integridade e o suporte às regras de negócio. A seguir, descrevem-se os principais relacionamentos modelados:

4.2.1.1 Herança e Especialização de Usuários

Pessoa, Organizador e Participante: O modelo adota o conceito de generalização/especialização. A entidade Pessoa atua como uma superclasse, concentrando atributos comuns a todos os usuários (como nome, Cadastro de Pessoa Física (CPF), e-mail, senha e data de nascimento). A partir dela, derivam-se duas subclasses: Organizador e Participante. Isso significa que tanto organizadores quanto participantes herdam as informações de Pessoa, mas podem participar de relacionamentos distintos no sistema.

4.2.1.2 Localização

Pessoa e Endereço (1:1): Há um relacionamento de um-para-um, indicando que uma Pessoa possui exatamente um Endereço de residência associado ao seu cadastro.

Local e Endereço (1:1): De forma análoga, cada Local (onde os eventos ocorrem) possui um único Endereço físico atrelado a ele.

4.2.1.3 Gestão de Eventos

Organizador e Evento (1:N): O relacionamento indica que um Organizador pode criar múltiplos Eventos (0,n). No entanto, a cardinalidade garante que cada Evento pertença a apenas um Organizador (1,1).

Evento e Categoria (N:M): Existe um relacionamento de muitos-para-muitos, representado pelo losango EventoCategoria. Isso estabelece que um Evento pode

abranger diversas categorias (ex: Música, Gastronomia) e uma Categoria pode estar associada a inúmeros eventos simultaneamente.

4.2.1.4 Interações do Participante (Inscrição e Avaliação)

Inscrição como Entidade Associativa: O relacionamento natural de muitos-para-muitos entre Participante e Evento foi resolvido transformando Inscrição em uma entidade. Um Participante pode ter várias inscrições (0,n), e um Evento recebe várias inscrições (0,n). A entidade Inscricao armazena os dados dessa transação (data da inscrição e se o usuário de fato participou).

Avaliação como Entidade Associativa: Semelhante à inscrição, a Avaliação conecta o Participante ao Evento. Um participante pode realizar diversas avaliações (0,n), e um evento pode receber várias avaliações (0,n). A entidade guarda a nota e o comentário.

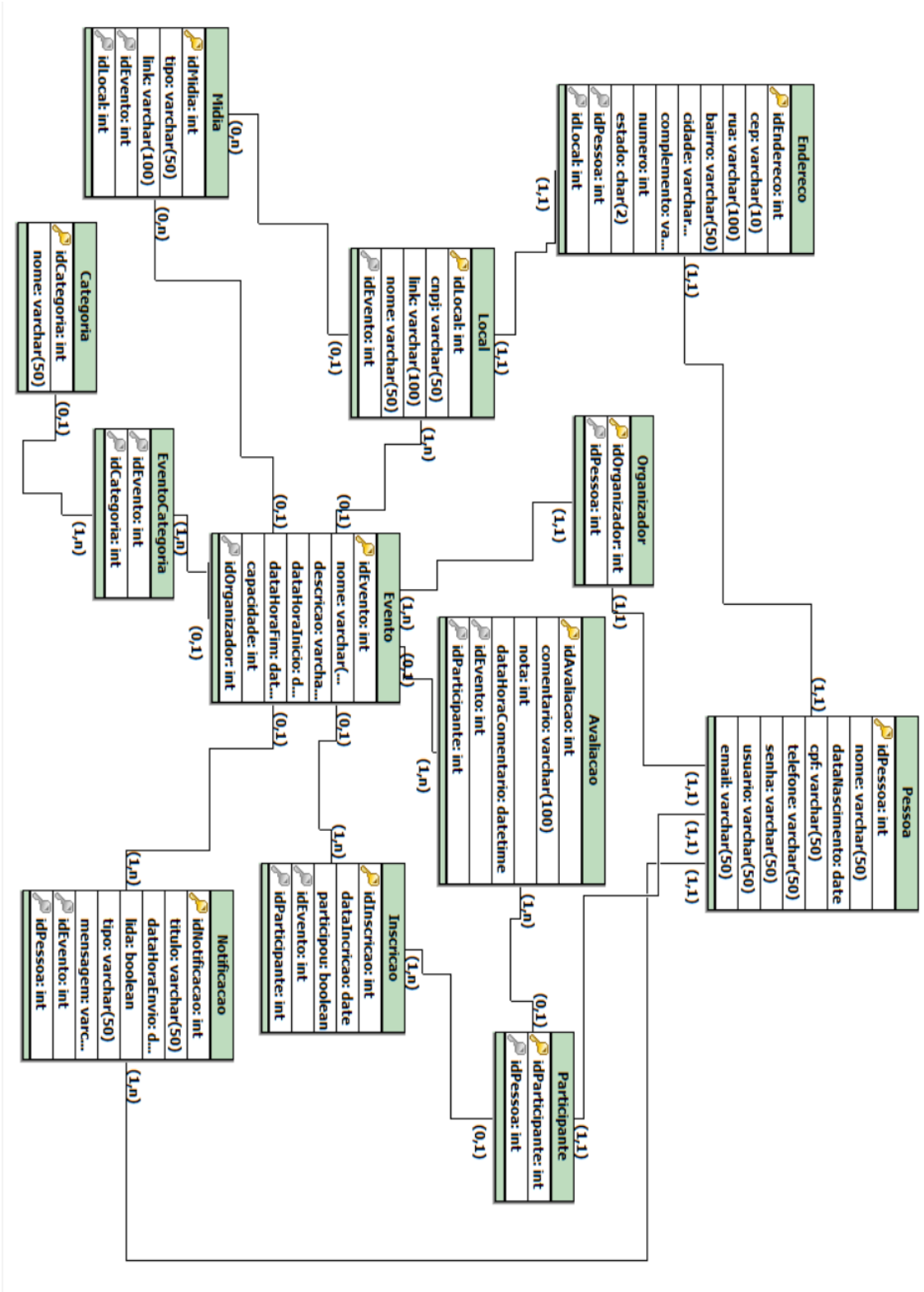
4.2.1.5 Arquivos e Comunicação

Mídia (1:N): A entidade Midia (que armazena fotos, vídeos ou links) relaciona-se tanto com Evento quanto com Local. Um evento ou um local podem possuir múltiplas mídias associadas (0,n) para enriquecer a exibição visual na plataforma.

Pessoa e Notificação (1:N): O sistema prevê comunicação direta com os usuários. Uma Pessoa pode receber múltiplas Notificações (0,n), que armazenam título, mensagem, tipo e o status de leitura.

4.2.2 Modelo Lógico

Figura 7 - Modelo Lógico do Banco de Dados



Fonte: Elaborado pelo autor (2026).

A evolução do modelo conceitual para o modelo lógico (esquema relacional) exigiu a aplicação de regras de mapeamento estrutural para traduzir entidades abstratas e relacionamentos em tabelas e restrições físicas. As principais alterações e implementações técnicas ocorridas nessa transição são detalhadas a seguir:

Materialização da Herança (Pessoa, Organizador e Participante): No modelo conceitual, a herança era representada de forma abstrata. No modelo lógico, essa generalização foi resolvida por meio da estratégia de tabelas mapeadas separadamente. A tabela superclasse Pessoa concentra todos os atributos bioatribuídos (como CPF, dataNascimento, senha). As tabelas subclasses Organizador e Participante foram criadas como entidades físicas distintas, contendo suas respectivas chaves primárias (idOrganizador e idParticipante) e herdando a associação com a superclasse por meio da inclusão obrigatória da chave estrangeira idPessoa.

Unificação Genérica de Relacionamentos 1:1 (Tabela Endereco): No modelo conceitual, os relacionamentos de localização de usuários e locais de eventos eram tratados de forma isolada. No modelo lógico, para otimizar o esquema e evitar a proliferação de tabelas redundantes, a entidade Endereco foi unificada. Ela passou a comportar simultaneamente as chaves estrangeiras idPessoa e idLocal como campos anuláveis. Essa alteração lógica permite que a mesma estrutura de tabela atenda de forma flexível tanto a um indivíduo quanto a um estabelecimento físico.

Conversão de Relacionamentos 1:N em Chaves Estrangeiras: As relações de um-para-muitos deixaram de ser representadas por losangos conceituais e foram convertidas em dependências lógicas diretas, com a migração da chave primária do lado "1" para o lado "N". Como modificações relevantes, a tabela Evento passou a hospedar fisicamente a chave estrangeira idOrganizador; a tabela Midia recebeu as chaves idEvento e idLocal para associar arquivos binários às suas origens; e a tabela Notificacao incorporou idEvento e idPessoa para associar os alertas aos seus respectivos contextos e destinatários.

Resolução de Relações N:M em Tabelas Associativas: O relacionamento de muitos-para-muitos entre eventos e categorias foi decomposto para atender à Primeira Forma Normal (1FN). O elemento conceitual EventoCategoria foi

transformado em uma tabela associativa física contendo exclusivamente o par de chaves estrangeiras idEvento e idCategoria, responsáveis por mapear a cardinalidade múltipla sem gerar redundância de dados.

Formalização de Entidades Associativas com Atributos Próprios: As interações complexas de Inscricao e Avaliacao, que conceitualmente ligavam participantes a eventos, foram consolidadas no modelo lógico como tabelas estruturadas independentes. Diferente de tabelas de junção simples, elas recebem chaves primárias próprias (idInscricao e idAvaliacao) para indexação interna, além de colunas específicas para regras de negócio, como dataInscricao e participou em Inscricao, e nota, comentario e dataHoraComentario em Avaliacao.

4.3 A Aplicação

Nesta seção, são apresentadas as principais funcionalidades da aplicação web desenvolvida. Para melhor compreensão do sistema, o fluxo de interação do usuário é ilustrado por meio de capturas de tela das interfaces, acompanhadas da descrição de seus respectivos comportamentos, regras de negócio e tecnologias aplicadas.

A construção do artefato ocorreu sob as diretrizes do *framework* ágil Scrum, estruturando-se o ciclo de desenvolvimento em cinco *sprints* com durações variáveis. O tempo de execução de cada iteração foi ajustado dinamicamente, adaptando-se à complexidade técnica e aos requisitos das funcionalidades implementadas em cada fase, conforme detalhado no Quadro 5.

Quadro 5 - Detalhamento das Sprints

Sprint	**Período**	**Foco Principal da Iteração**
Sprint 1	Setembro/2025	Elicitação de requisitos.
Sprint 2	Outubro/2025	Modelagem do banco de dados.
Sprint 3	Novembro/2025	Implementação das estruturas no Django.

Sprint 4	Dezembro/2025	Validação dos requisitos, testes internos e *deploy.*
Sprint 5	Fevereiro/2026 até Junho/2026	Testes com usuários selecionados e métricas de experiência do usuário.

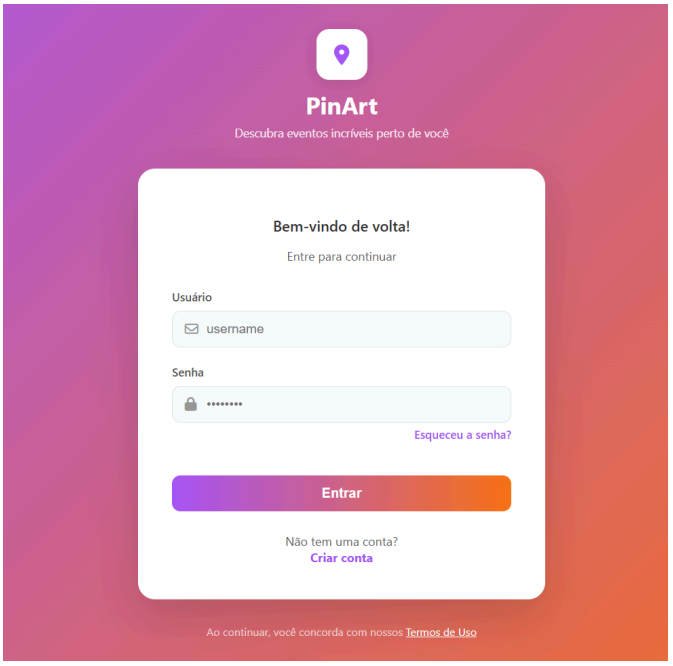
Fonte: Elaborado pelo autor (2026).

O ciclo de desenvolvimento descrito no Quadro 5 foi registrado por meio de controle de versão utilizando o *Git*. O projeto foi armazenado em um repositório na plataforma *GitHub*, com as mensagens de atualização padronizadas sob o formato *Conventional Commits*, garantindo a rastreabilidade entre o código-fonte produzido e as funcionalidades implementadas. O código-fonte integral da aplicação encontra-se disponível para consulta pública por meio do *link*: https://github.com/RqfaelJr/pinart.

4.3.1 Cadastro e Autenticação

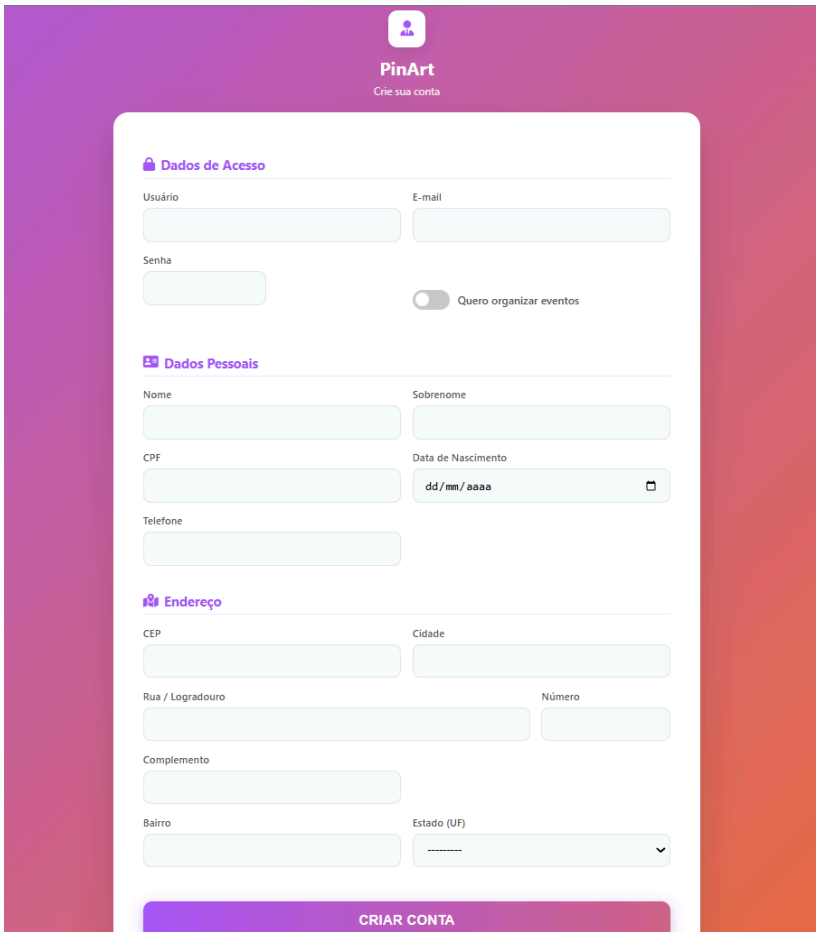
Para acessar as funcionalidades da plataforma e interagir com os eventos, o usuário deve inicialmente realizar sua identificação no sistema ou criar um novo perfil. As Figuras 8 e 9 ilustram a interface desenvolvida para essa finalidade, composta pelos formulários de entrada de credenciais e pela opção de registro de novos usuários. O design desta tela foi projetado para ser intuitivo, garantindo uma porta de entrada simples antes que os mecanismos de segurança de infraestrutura sejam acionados pelo servidor.

Figura 8 - Tela de Login



Fonte: elaborado pelo autor (2026)

Figura 9 - Tela de Cadastro de Usuário



Fonte: elaborado pelo autor (2026)

O gerenciamento de credenciais e o controle de acesso foram desenvolvidos utilizando os recursos de segurança nativos do *framework* Django. No que diz respeito ao armazenamento de senhas, o sistema não utiliza texto plano ou criptografia reversível, adotando, em vez disso, funções de *hashing* unidirecional. O algoritmo empregado é o PBKDF2 associado à função SHA-256, que realiza um alto número de iterações (aproximadamente 220.000), característica que torna inviável a execução de ataques de força bruta (Django Software Foundation, 2024).

Adicionalmente, para cada credencial, o sistema gera um *salt* criptográfico exclusivo, um valor aleatório combinado à senha antes do processo de *hash*. Essa abordagem garante que senhas idênticas resultem em *hashes* diferentes no banco de dados, protegendo a aplicação contra ataques de dicionário e *Rainbow Tables*. O processo de validação de login ocorre de forma segura, extraindo o *salt* e o número de iterações do banco de dados, aplicando o algoritmo na senha informada e verificando a exata correspondência dos hashes.

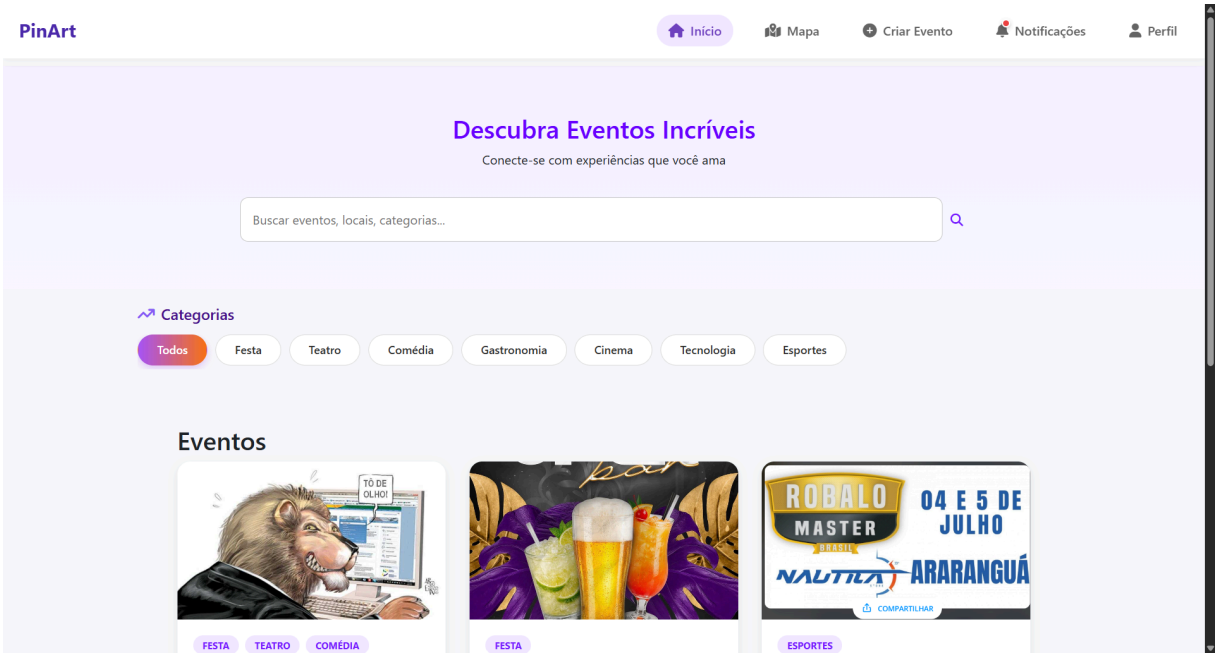
A manutenção do estado de autenticação foi estruturada com base em sessões no lado do servidor. Após a validação das credenciais, o servidor cria um registro de sessão e emite um identificador único (*sessionId*). Esse identificador é enviado ao navegador do usuário via *cookie* Hypertext Transfer Protocol (HTTP) com a diretiva *HttpOnly* habilitada. Essa configuração é uma medida de segurança fundamental, pois impede que scripts no lado do cliente (como *JavaScript*) acessem o *cookie*, mitigando vulnerabilidades de XSS. Em requisições subsequentes, os middlewares da aplicação validam o identificador automaticamente e vinculam o usuário à sessão ativa.

Por fim, visando garantir a integridade das requisições de alteração de dados (como *POST*, *PUT* e *DELETE*), a aplicação implementa proteção nativa contra ataques do tipo *Cross-Site Request Forgery* (CSRF). O mecanismo atua injetando um token de validação dinâmico (*csrfmiddlewaretoken*) em todos os formulários renderizados. Durante o envio dos dados, o servidor exige a correspondência exata entre o token contido no formulário oculto e o *cookie* de segurança do navegador. Caso haja divergência ou ausência do token, a requisição é imediatamente bloqueada (Erro 403 - *Forbidden*), protegendo o banco de dados contra ações forjadas por sites maliciosos de terceiros.

4.3.2 Feed, Busca de Eventos e Mapa

A interface principal da plataforma é responsável por apresentar o catálogo de eventos disponíveis aos usuários, oferecendo múltiplos mecanismos de descoberta. A Figura 10 ilustra esta tela, destacando a organização visual dos *cards* e os componentes de pesquisa.

Figura 10 - Tela principal de listagem e busca de eventos



Fonte: Elaborado pelo autor (2026).

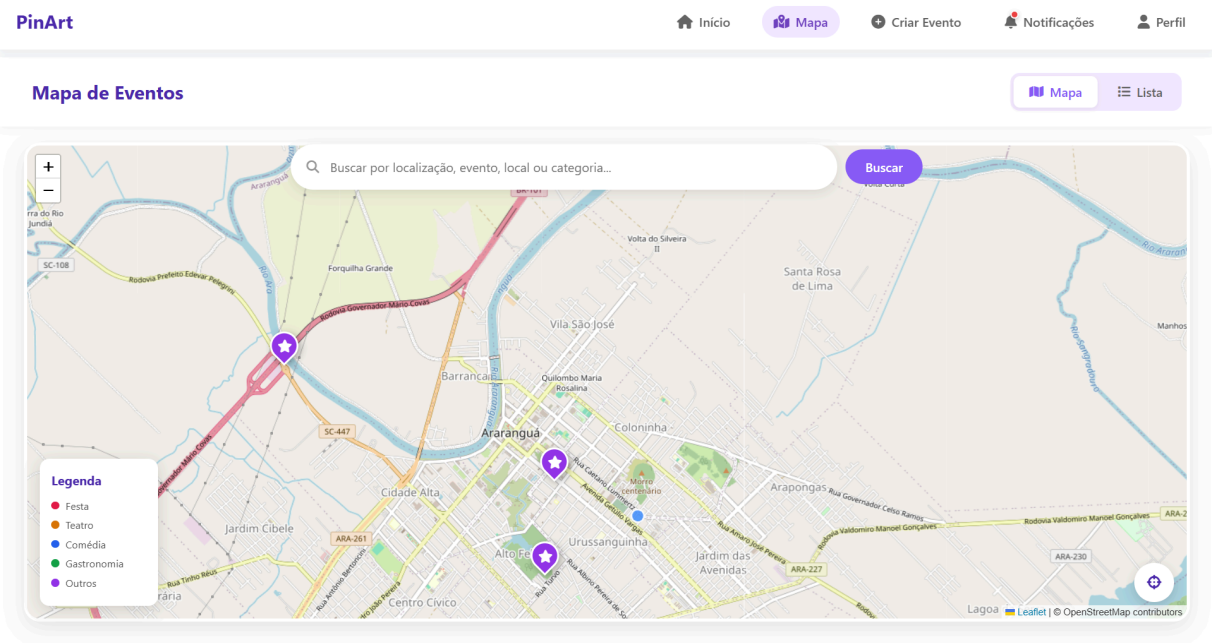
Para otimizar a recuperação da informação, o sistema emprega uma arquitetura de busca híbrida, atuando de formas distintas no servidor (*backend*) e no cliente (*frontend*). Na página inicial, as consultas são processadas diretamente no banco de dados por meio do ORM do *framework* Django. O mecanismo textual utiliza operadores lógicos condicionais para realizar buscas simultâneas e insensíveis a maiúsculas ou minúsculas (*case-insensitive*) nos campos de título, descrição, local e categoria. Visando assegurar a integridade da listagem, um filtro de distinção matemática é aplicado para evitar a exibição duplicada de registros que satisfaçam múltiplos critérios de busca.

Em relação à ordenação e exibição padrão, o sistema aplica uma restrição temporal na camada de dados, filtrando e exibindo estritamente eventos com data

de início superior ao momento da consulta. A ordenação cronológica ascendente garante que os eventos mais próximos sejam priorizados na interface.

Quando a visualização ocorre no contexto do mapa interativo (como ilustra a Figura 11), a busca adota uma abordagem orientada ao cliente. O servidor fornece um conjunto de dados estruturados em formato *JavaScript Object Notation* (JSON), permitindo que a filtragem ocorra dinamicamente via *JavaScript*. Essa técnica elimina a necessidade de recarregamento da página, proporcionando uma experiência contínua. À medida que o usuário insere os termos de pesquisa, o algoritmo redesenha os marcadores e ajusta automaticamente as fronteiras de enquadramento da câmera do mapa para focar exclusivamente nos resultados correspondentes.

Figura 11 - Visualização dos eventos via Mapa Interativo



Fonte: Elaborado pelo autor (2026).

A representação espacial dos eventos é viabilizada pela integração com a API Nominatim do *OpenStreetMap*. Durante o cadastro de um evento, o *backend* realiza a geocodificação automática do endereço em texto, convertendo-o em coordenadas geográficas exatas (latitude e longitude). Para facilitar a leitura do mapa pelo usuário, adotou-se um sistema de legendas dinâmicas: os marcadores assumem cores e ícones específicos com base na categoria do evento (como gastronomia,

cultura ou festas). Adicionalmente, a aplicação consome a API de geolocalização nativa dos navegadores, permitindo que o usuário centralize o mapa em sua posição atual mediante autorização de acesso ao *Global Positioning System* (GPS).

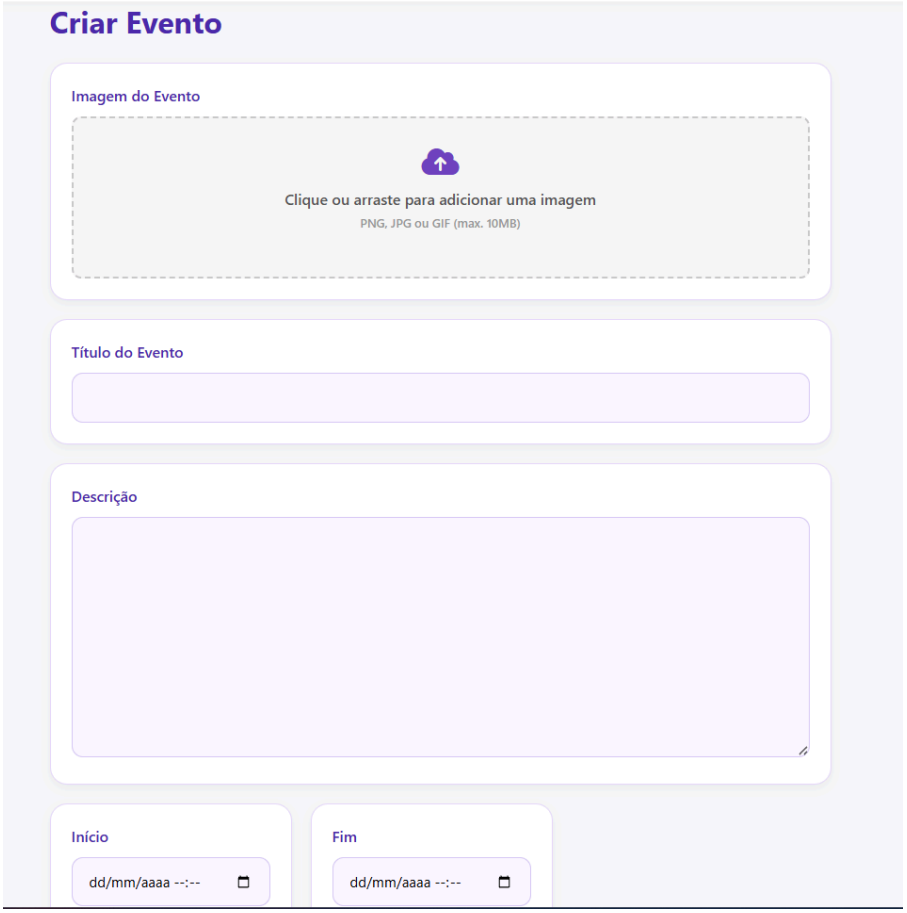
Por fim, o design das interfaces foi fundamentado em princípios de usabilidade e responsividade. O layout emprega estruturas em malha flexível (*CSS Flexbox*), que adapta dinamicamente a exibição dos cartões de três colunas em monitores dos computadores para uma coluna única em dispositivos móveis, maximizando a legibilidade. No ambiente móvel, o mapa adota um comportamento de tela cheia e as categorias utilizam rolagem horizontal nativa, otimizando o espaço para o toque.

Para prevenir a sobrecarga cognitiva e a poluição visual em regiões com alta densidade de eventos, o mapa emprega técnicas de clusterização, agrupando marcadores próximos que se expandem fluidamente ao receberem zoom. Tais características, aliadas a micro animações de interação e botões de alternância de visualização (mapa para lista), garantem uma experiência de uso imersiva e acessível em diferentes dimensões de tela.

4.3.3 Criação de Eventos, Validação e Persistência de Dados

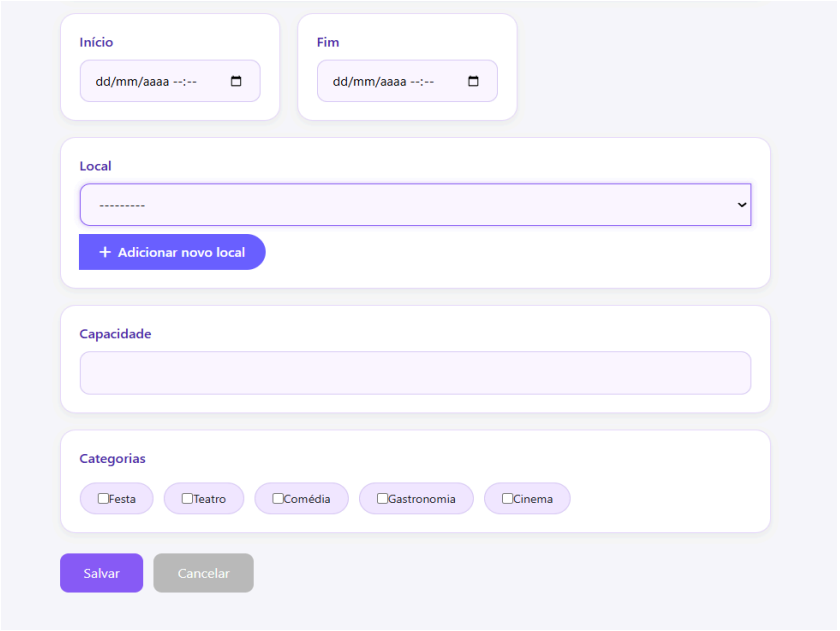
As Figuras 12 e 13 apresentam o formulário de cadastro, o qual incorpora componentes dinâmicos para otimizar a interação. Destaca-se a implementação de uma área interativa para upload de imagens, que utiliza *JavaScript* para processar o arquivo no lado do cliente e fornecer feedback visual imediato antes do envio ao servidor.

Figura 12 - Interface de criação de eventos (Parte 1)



Fonte: Elaborado pelo autor (2026).

Figura 13 - Interface de criação de eventos (Parte 2)



Fonte: Elaborado pelo autor (2026).

Para a seleção de múltiplos itens, como as categorias do evento, o sistema substitui as tradicionais caixas de seleção por componentes visuais clicáveis, melhorando a legibilidade e a interação tátil. Adicionalmente, o preenchimento da localização adota uma abordagem de inserção *inline*. Caso o local desejado não conste no banco de dados, o usuário pode acionar uma janela modal para cadastrá-lo instantaneamente, sem que ocorra a perda do estado atual dos dados já preenchidos no formulário principal.

No que tange à persistência dos dados e às regras de negócio, o mapeamento da obrigatoriedade das informações foi projetado para garantir a consistência do sistema. O Quadro 6 sumariza o comportamento de cada atributo do modelo de domínio na relação entre a interface e o esquema relacional.

Quadro 6 - Mapeamento de obrigatoriedade e restrições do modelo Evento

Campo	**Obrigatório (Interface)**	**Restrição (Banco de Dados)**
Título	Sim	*NOT NULL*
Descrição	Sim	*TEXT, NOT NULL*
Data/Hora Início	Sim	*DATETIME, NOT NULL*
Data/Hora Fim	Sim	*DATETIME, NOT NULL*
Capacidade	Sim	*INTEGER, NOT NULL*
Local	Não (Opcional)	*Foreign Key* (FK), *NOT NULL*
Categorias	Sim	Tabela intermediária
Organizador	Não (Preenchido no *backend*)	*NOT NULL*
Imagem	Não (Opcional)	*NULL*

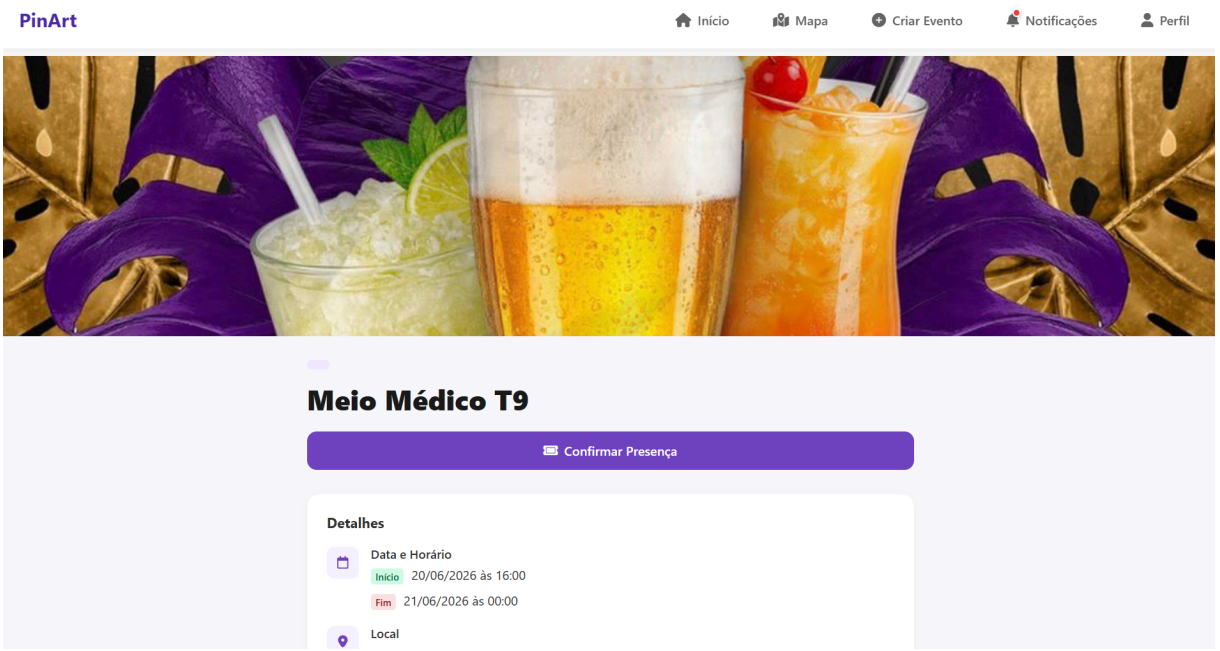
Fonte: Elaborado pelo autor (2026).

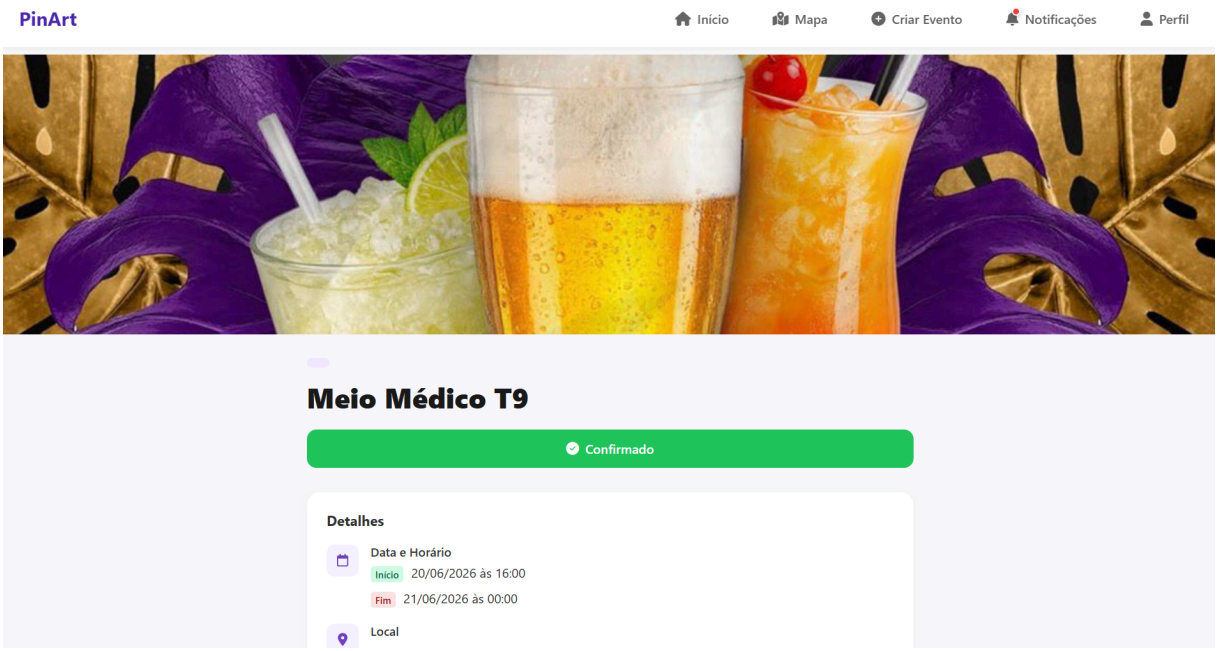
4.3.4 Inscrição, Presença e Conexão com Eventos

A funcionalidade de inscrição constitui o núcleo interativo da plataforma, viabilizando que os participantes cadastrados demonstrem interesse ativo nas atividades oferecidas, ao passo que fornece aos organizadores métricas precisas sobre o engajamento e a ocupação do público.

A interface de confirmação de presença, ilustrada na Figura 14, foi projetada sob a premissa de fluidez, permitindo a execução da ação em um único clique. A tela de detalhes do evento exibe elementos de atração visual, como um banner de destaque e tags de categorização. O principal componente de interação é um botão dinâmico, cujo estado visual e semântico reflete a situação atual do usuário em relação ao evento. Para usuários não inscritos, o componente apresenta-se em tom roxo com a instrução "Confirmar Presença"; uma vez acionado, a interface é atualizada para um estado de confirmação em tom verde, indicando o sucesso da operação. O sistema também exibe um contador atualizado em tempo real da quantidade de participantes.

Figura 14 - Interface de detalhes do evento e alternância do estado de inscrição





Fonte: Elaborado pelo autor (2026).

4.4 Validação da Plataforma

Esta seção destina-se a apresentar os resultados da validação da plataforma PinArt após a sua implementação, com o intuito de comprovar a sua eficácia técnica e funcional. Para garantir o rigor metodológico, a etapa de validação compreendeu a verificação formal do atendimento aos requisitos de software e o monitoramento de métricas de navegação, avaliando a intuição do usuário a partir de seus registros de clique e atenção visual.

A escolha por uma ferramenta de análise comportamental, especificamente o Microsoft Clarity, fundamentou-se na necessidade de observar o comportamento real e espontâneo dos usuários no site, eliminando o viés que frequentemente ocorre em avaliações controladas. Essa abordagem metodológica é respaldada por Albert e Tullis (2013), que apontam que a coleta de métricas de navegação no contexto de uso real confere maior 'validade ecológica' à avaliação, permitindo o registro de ações autênticas em vez de comportamentos simulados.

Apesar da adequação do método, a principal dificuldade encontrada nesta etapa foi o recrutamento de um número suficiente de usuários para gerar um volume de interações consistente. Em contrapartida, os benefícios práticos observados foram muito positivos: a ferramenta permitiu a consolidação e interpretação de todos

os resultados de forma centralizada. A partir da geração de mapas de calor e profundidade de rolagem, foi possível visualizar diretamente como os usuários interagiram com a interface, identificando com precisão os pontos de maior retenção de atenção e aqueles que passaram despercebidos.

4.4.1 Verificação de Requisitos e Regras de Negócio

O processo de verificação funcional atestou que a plataforma construída reflete fielmente o escopo definido na fase de elicitação (Seção 4.1). As validações foram conduzidas por meio de testes de uso simulando os fluxos reais de organizadores e participantes no ambiente de produção.

Conforme detalhado ao longo da Seção 4.3, as interfaces gráficas e os fluxos de navegação demonstrados atestam o cumprimento dos Requisitos Funcionais. De maneira complementar, os Requisitos Não Funcionais relacionados à segurança e responsividade foram validados por meio da própria arquitetura do *framework* Django e testes em múltiplos navegadores e resoluções. Por fim, as Regras de Negócio, como as restrições de feedback e validações de eventos, tiveram seu comportamento devidamente homologado nas camadas de *backend* e banco de dados. Dessa forma, considera-se que o sistema atendeu integralmente aos critérios estabelecidos para esta versão do projeto.

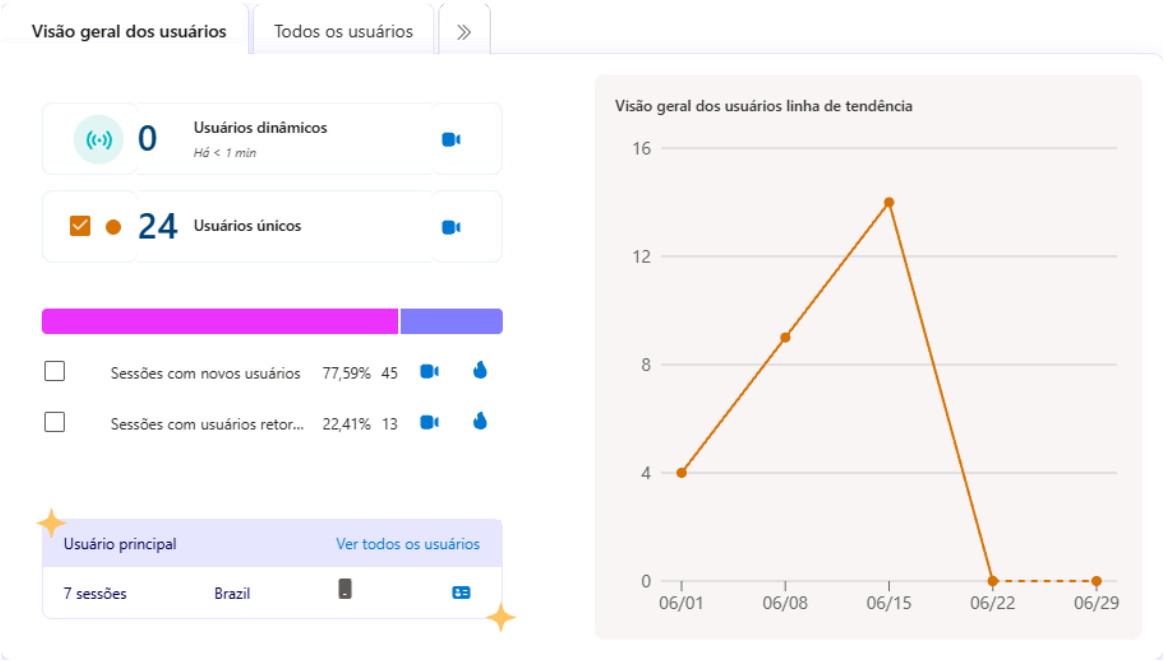
4.4.2 Análise Empírica de Interação

O objetivo desta instrumentação foi capturar métricas sobre os padrões de interação na plataforma para verificar se a disposição visual e os fluxos de navegação são intuitivos e atendem aos critérios funcionais estabelecidos. Para viabilizar esta etapa de forma não intrusiva, a plataforma foi integrada à ferramenta analítica Microsoft Clarity, que registrou sessões reais de uso durante o período de homologação.

A análise concentrou-se primariamente na interatividade e na compreensão visual do sistema. O grupo de participantes foi composto por jovens acadêmicos, cursando graduação ou já graduados em diversas áreas do conhecimento, recrutados de forma orgânica por meio de aplicativos de mensagens.

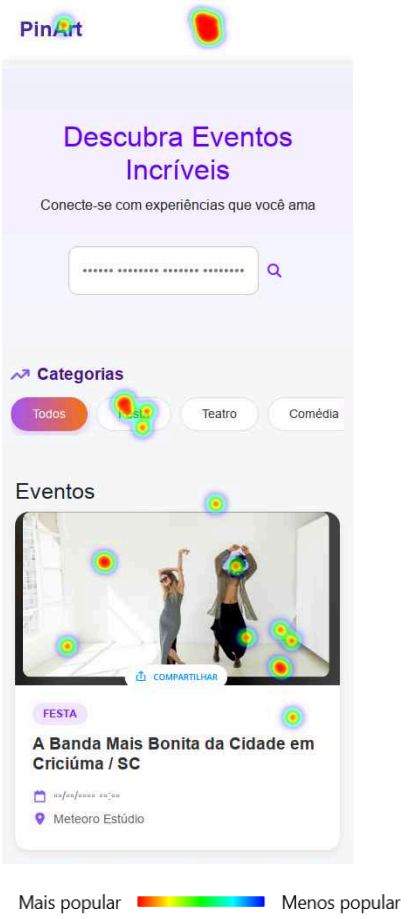
Durante o período de metrificação, compreendido entre os dias 01 de junho e 22 de junho de 2026, registrou-se a participação de 24 usuários únicos, conforme ilustra a Figura 15. A Figura 16 apresenta o mapa de calor de cliques capturado na tela principal do sistema, evidenciando os padrões reais de interação visualizados.

Figura 15 - *Dashboard* com informações de usuários



Fonte: dados retirados do Microsoft Clarity (2026).

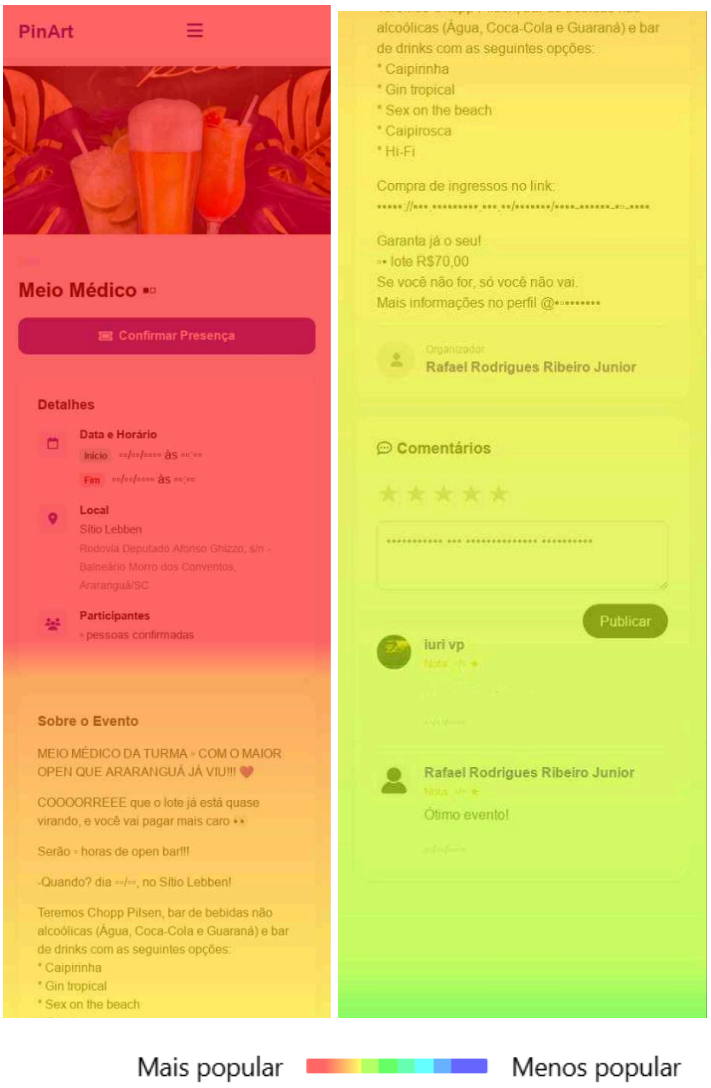
Figura 16 - Mapa de calor de cliques na interface principal da plataforma PinArt



Fonte: dados retirados do Microsoft Clarity (2026).

Após a observação do mapa de calor, é possível identificar três comportamentos principais de usabilidade no sistema. Primeiramente, nota-se uma área de calor extremo na parte superior direita; esta região abriga o menu de navegação, indicando uma forte intenção do público em explorar as opções que a plataforma oferece. Em segundo lugar, observa-se uma área de concentração de cliques na seção de categorias, demonstrando que a filtragem de eventos por temáticas específicas é uma funcionalidade altamente relevante para os usuários. Por fim, o padrão de cliques na listagem de eventos sugere que o público interage intuitivamente com as imagens de capa, na expectativa de que uma tela de detalhamento seja aberta, o que reforça a premissa de que a área clicável deve abranger toda a imagem para otimizar a experiência de navegação. Outra métrica relevante capturada referiu-se à taxa de rolagem nas páginas de detalhes dos eventos, conforme ilustrado na Figura 17.

Figura 17 - Profundidade de rolagem na página de detalhes do evento



Fonte: dados retirados do Microsoft Clarity (2026).

Complementando a análise de navegação da interface mobile, foi analisado o mapa de rolagem na página de detalhamento de um evento específico. Por meio desta métrica, é possível verificar como a hierarquia visual influencia a leitura das informações mais importantes. Os dados demonstraram que a zona superior da tela foi visualizada por todos os usuários, evidenciando que as informações fundamentais, como título, horário e local, devem permanecer nesta região, pois permitem que o usuário capte os dados essenciais sem a necessidade de rolar a página. Observa-se que, à medida que a tela avança para as seções inferiores, há uma redução gradual nas visualizações. Isso indica que as decisões do público são

tomadas predominantemente com base nas informações do topo, resultando em uma subutilização da seção de comentários, o que se configura como um ponto de melhoria para trabalhos futuros na plataforma.

4.5 Síntese dos Resultados e Resposta à Questão de Pesquisa

A implementação da plataforma responde de forma afirmativa à questão de pesquisa deste trabalho. A centralização foi viabilizada pela própria arquitetura do sistema, que atua como um repositório único, agrupando a criação e a descoberta de eventos dispersos em um só lugar. Por sua vez, a visualização intuitiva por parte dos usuários finais foi comprovada empiricamente através da análise do mapa de calor na tela principal. Os dados evidenciaram que os usuários exploraram as funcionalidades de maneira fluida e autônoma, utilizando a filtragem por categorias e os cliques nos *cards* de forma natural, validando a eficácia da interface proposta.

5 CONSIDERAÇÕES FINAIS

O presente trabalho teve como principal objetivo o desenvolvimento da plataforma web PinArt, um sistema voltado à conexão e ao mapeamento de eventos geolocalizados. A proposta buscou criar um ambiente digital moderno, acessível e interativo, capaz de aproximar organizadores e participantes. Dessa forma, o projeto atuou para facilitar a divulgação e o gerenciamento de eventos, promovendo uma experiência mais integrada e eficiente, centralizando informações e estimulando o engajamento da comunidade local.

Durante o desenvolvimento, foi possível alcançar resultados que comprovaram a viabilidade técnica e prática da proposta. Retomando a questão de pesquisa que norteou este estudo, “Como a implementação de uma plataforma web baseada em geolocalização pode viabilizar a centralização e a visualização de eventos de forma intuitiva para os usuários finais?”, conclui-se que o objetivo foi plenamente alcançado. A centralização foi efetivada pela arquitetura do sistema, que unificou eventos dispersos em um único ambiente. Por sua vez, a visualização intuitiva foi atestada por métricas empíricas de usabilidade, como os mapas de calor, que demonstraram uma exploração autônoma e fluida das buscas, filtragens e cliques intuitivos nos *cards*. Além disso, a aplicação da metodologia ágil Scrum e de boas práticas de versionamento de código provou-se fundamental para a evolução do projeto, garantindo entregas incrementais e permitindo ajustes precisos ao longo das etapas de implementação do *backend* e *frontend*.

Como contribuição científica e tecnológica, este estudo propõe a estruturação e validação empírica de uma plataforma de eventos que integra geolocalização interativa e métricas de comportamento real . A pesquisa demonstra como o monitoramento não intrusivo pode guiar o aprimoramento de interfaces voltadas para a descoberta de eventos. No entanto, reconhecem-se limitações metodológicas no estudo. A avaliação de interação foi conduzida com uma amostra de 24 usuários recrutados por conveniência, o que restringe a generalização dos resultados. Além disso, a abordagem de monitoramento remoto, embora confira validade ecológica ao teste, impossibilita a coleta de dados qualitativos profundos (como entrevistas pós-uso) que poderiam elucidar as motivações exatas por trás dos padrões de navegação observados.

Para contornar essas limitações e expandir o escopo do projeto, propõem-se trabalhos futuros específicos focados na evolução arquitetural e funcional da plataforma. Destaca-se a implementação de um sistema de recomendação baseado em Inteligência Artificial, utilizando agentes autônomos ou grandes modelos de linguagem para cruzar o histórico de interação, avaliações passadas e preferências individuais, gerando uma curadoria de eventos altamente personalizada. Complementarmente, sugere-se a criação de um mecanismo de notificações inteligentes baseadas em localização, aproveitando a infraestrutura de mapas já existente para alertar os usuários, em tempo real, sobre eventos de interesse nas suas imediações.

Adicionalmente, no que tange à disposição visual da interface, os dados de profundidade de rolagem indicam a necessidade de reestruturação da página de detalhes do evento. Planeja-se o reposicionamento da seção de avaliações e comentários para uma zona de maior visibilidade (acima da dobra da página), visando mitigar sua subutilização e fomentar a criação de comunidade. Por fim, vislumbra-se a criação de um aplicativo móvel nativo e o desenvolvimento de módulos analíticos avançados para organizadores, fornecendo estatísticas de retenção e visualização para consolidar o sistema no mercado digital.

REFERÊNCIAS

ABRAPE. **Números do Setor**. 2023. Disponível em: https://abrape.com.br/numeros-do-setor/. Acesso em: 15 set. 2025.

ABRAPE. **Setor de eventos têm crescimento do PIB superior à média nacional**. 2024. Disponível em: https://www.abrape.com.br/setor-de-eventos-tem-crescimento-do-pib-superior-a-media-nacional/. Acesso em: 27 out. 2025.

AL HASAN, Sakib; QUADIR, HM Sazzad; TAWHID, Md Nurul Ahad. **Conventional Commit Classification using Large Language Models and Prompt Engineering.** 2026.

ALBERT, William; TULLIS, Thomas. **Measuring the user experience: collecting, analyzing, and presenting usability metrics.** 2. ed. Waltham: Morgan Kaufmann, 2013.

ALVES, Carolina Pereira. **O papel da comunicação e das tecnologias na criação de eventos de sucesso.** The Trends Hub, n. 3, 2023.

ANDERSON, David J. **Kanban: successful evolutionary change for your technology business.** Blue hole press, 2010.

ARMSTRONG, Gary; KOTLER, Philip. **Marketing: an introduction. Pearson Educación**, 2003.

BATISTA, Eduarda Alves Calderon. **Importância da comunicação e mídias sociais em eventos**. 2024.

BEZERRA JUNIOR, Antonio Carlos Rocha. **Desenvolvimento de aplicação web em django como ferramenta de auxílio pedagógico para o ensino de operações com estágios.** 2023.

BOZDAG, E. **Bias in algorithmic filtering and personalization**. Ethics and Information Technology, v. 15, n. 3, p. 209-227, 2013.

CARDOSO, Margarida Lopes. **A importância da organização de eventos no turismo.** 2013. Dissertação de Mestrado. Universidade do Porto (Portugal).

CARNEIRO, Janaina Britto; FONTES, Nena Dantas. **Turismo e eventos: instrumento de promoção e estratégia de marketing.** Revista turismo em análise, v. 8, n. 1, p. 65-74, 1997.

CHACON, S.; STRAUB, B. **Pro Git.** 2. ed. Apress, 2014. Disponível em: https://git-scm.com/book/en/v2. Acesso em: 30 out. 2025.

CLERICUZI, Adriana Zenaide; SOUZA, Emmanuel Gomes; DA SILVA, Dalvanete Cavalcante. **Novas tecnologias para a organização de eventos**. Brazilian Journal of Development, v. 7, n. 4, p. 34492-34507, 2021.

COUTINHO, Helen Rita Menezes. **Organização de eventos.** 2010.

DANIELI, Karen Belleboni. **A importância dos eventos culturais: um estudo sobre a 32ª edição do FestiQueijo.** TCC. Universidade de Caxias do Sul. 2024.

DJANGO SOFTWARE FOUNDATION. ***Django Documentation*.** Disponível em: https://docs.djangoproject.com/. 2024 Acesso em: 30 out. 2025.

Doity, 2013. Disponível em: https://doity.com.br/. Acesso em: 04 jun. 2026.

EventBride, 2006. Disponível em: https://www.eventbrite.com.br/. Acesso em: 04 jun. 2026.

FLANAGAN, David. JavaScript: **The definitive guide: Activate your web pages. "** O'Reilly Media, Inc.", 2011.

HENDRICKSON, Mark. **The Uphill Battle Of Social Event Sharing: A Post-Mortem for Plancast**. 2012. Disponível em: https://techcrunch.com/2012/01/22/post-mortem-for-plancast/. Acesso em: 12 nov. 2025.

JUBA, Salahaldin; VANNAHME, Achim; VOLKOV, Andrey. **Learning PostgreSQL.** Packt Publishing Ltd, 2015.

KÜPPER, Axel. **Location-based services: fundamentals and operation.** Chichester: John Wiley & Sons, 2005.

LIMA, Eugenio Lopes Fernandes; LOPES, Pedro Henrique de Meneses Bittencourt; NASCIMENTO, Francisco Abud. **Boas práticas de modelagem de dados em bancos relacionais: uma revisão rápida de literatura.** Revista Sociedade Científica, v. 9, n. 1, p. 1188-1202, 2026.

LOELIGER, Jon; MCCULLOUGH, Matthew. **Version Control with Git: Powerful tools and techniques for collaborative software development. "** O'Reilly Media, Inc.", 2012.

LUCCHESI, Claudio. **Domine HTML 5 e CSS 3: Do Básico ao Avançado.** 2024.

MALKIN, Hugh. **Why no one has solved event discovery.** 2015. Disponível em: https://www.hughmalkin.com/blogwriter/2015/9/23/why-no-one-has-solved-event-discovery. Acesso em: 27 mai. 2026.

MAZZA, Lucas. **HTML5 e CSS3: domine a web do futuro.** Editora Casa do Código, 2014.

MELL, Peter; GRANCE, Timothy. **The NIST definition of cloud computing**. Gaithersburg: National Institute of Standards and Technology, 2011. (NIST Special Publication 800-145). Disponível em: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf. Acesso em: 28 maio 2026.

MENEZES, Nilo Ney Coutinho. **Introdução a programação com Python.** São Paulo: Novatec, p. 13, 2010.

MOZILLA. **CSS: Cascading Style Sheets.** MDN Web Docs, 2023. Disponível em: https://developer.mozilla.org/pt-BR/docs/Web/CSS. Acesso em: 28 maio 2026.

NIELSEN, Jakob. **Usability 101: Introduction to Usability.** Nielsen Norman Group, 2012. Disponível em: https://www.nngroup.com/articles/usability-101-introduction-to-usability/. Acesso em: 28 maio 2026.

OLIVEIRA, Felipe Rodrigues; MAZIERO, Ronaldo Colucci; DE ARAÚJO, Liriane Soares. **Um estudo sobre a web 3.0: evolução, conceitos, princípios, benefícios e impactos**. Revista Interface Tecnológica, v. 15, n. 2, p. 60-71, 2018.

PEFFERS, Ken. et al. **A design science research methodology for information systems research.** Journal of Management Information Systems, v. 24, n. 3, p. 45-77, 2007.

PRESSMAN, R. S.; MAXIM, B. R. **Engenharia de Software: Uma Abordagem Profissional.** 9. ed. Porto Alegre: AMGH, 2021.

RODRIGUES, Marcelo Aguiar. **What's Happening?–Uma plataforma de eventos.** 2018. Tese de Doutorado. Instituto Superior de Engenharia de Coimbra.

SAUCEDO, Lázaro Lima. **Bora! Sistema de gerenciamento de eventos culturais.** 2025.

SCHWABER, Ken; SUTHERLAND, Jeff. **The scrum guide.** Scrum Alliance, v. 21, n. 1, p. 1-38, 2011.

SHAW, Ryan; TRONCY, Raphaël; HARDMAN, Lynda. **Lode: Linking open descriptions of events.** In: Asian semantic web conference. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. p. 153-167.

SMALL, K. **Social dimensions of community festivals: an application of factor analysis in the development of the Social Impact Perception (SIP) scale**. Event Management, v. 11, p. 45-55, 2007.

SOMMERVILLE, Ian. **Software engineering** (ed. 9). America: Pearson Education Inc, 2011.

Sympla, 2012. Disponível em: https://www.sympla.com.br/. Acesso em: 04 jun. 2026.

THAN, Myint.. **A Comparative Analysis of Traditional and Agile SDLC Models for Software Development.** 2020.

VASCONCELOS, Davina. **USO DA TECNOLOGIA PARA POTENCIALIZAR O ÊXITO DE CERIMONIAIS**. Publicações, 2023.

WATT, David C. **Gestão de eventos em lazer e turismo.** Bookman Editora, 2