RGe ISSN: 2177-3246

WALK: SISTEMA DE GESTÃO WEB PARA CURSINHO PRÉ-UNIVERSITÁRIO UTILIZANDO O FRAMEWORK DJANGO

WALK: WEB-BASED MANAGEMENT SYSTEM FOR PRE-UNIVERSITY COURSES USING THE DJANGO FRAMEWORK

WALK: SISTEMA DE GESTIÓN WEB PARA CURSOS PREUNIVERSITARIOS UTILIZANDO EL FRAMEWORK DJANGO

10.56238/revgeov17n6-119

Daniela Marçal dos Santos
Graduanda em Engenharia de Software
Instituição: Universidade do Estado do Pará (UEPA)
E-mail: dannymarcal2014@gmail.com

Rayda Matias Lima
Graduanda em Engenharia de Software
Instituição: Universidade do Estado do Pará (UEPA)
E-mail: raydaadm@gmail.com

Patrícia Lopes dos Santos
Graduanda em Engenharia de Software
Instituição: Universidade do Estado do Pará (UEPA)
E-mail: patricia.lopes19@outlook.com

Armando José de Sá Santos
Mestre em Ciência e Engenharia de Materiais
Instituição: Universidade do Estado do Pará (UEPA)
E-mail: armando.santos@uepa.br

Wilker José Caminha dos Santos
Mestrando em Propriedade Intelectual e Transferência de Tecnologia para Inovação
Instituição: Universidade do Estado do Pará (UEPA)
E-mail: wilkercaminha@uepa.br

RESUMO
Este trabalho apresenta o desenvolvimento do Sistema Walk, uma plataforma de gestão e suporte pedagógico voltada a cursinhos pré-universitários. O projeto parte da necessidade de qualificar a gestão pedagógica e administrativa de cursinhos populares, reduzindo processos manuais, retrabalho e inconsistências de informação. Metodologicamente, a solução foi desenvolvida com o framework Django, em Python, adotando o padrão arquitetural MVT e recursos de ORM para persistência de dados. Como resultados, obteve-se uma plataforma funcional capaz de automatizar a pré-matrícula, organizar conteúdos por áreas do conhecimento, controlar acessos por perfis e oferecer painéis administrativos e estudantis mais intuitivos. Conclui-se que o sistema constitui uma ferramenta viável

para fortalecer iniciativas de educação popular, ampliando a eficiência da gestão e o suporte ao estudante.

Palavras-chave: Desenvolvimento Web. Django. Engenharia de Software. Gestão Educacional. Cursinho Popular.

ABSTRACT
This paper presents the development of the Walk System, a management and pedagogical support platform designed for pre-university preparatory courses. The project arises from the need to improve the pedagogical and administrative management of community-based courses by reducing manual procedures, rework, and data inconsistencies. Methodologically, the solution was developed with the Django framework in Python, adopting the MVT architectural pattern and ORM resources for data persistence. The results show a functional platform capable of automating pre-enrollment, organizing content by areas of knowledge, controlling access by user profiles, and providing more intuitive administrative and student dashboards. It is concluded that the system is a viable tool to strengthen popular education initiatives and improve educational management.

Keywords: Web Development. Django. Software Engineering. Educational Management. Community Preparatory Course.

RESUMEN
Este trabajo presenta el desarrollo del Sistema Walk, una plataforma de gestión y apoyo pedagógico orientada a cursos preuniversitarios. El proyecto surge de la necesidad de mejorar la gestión pedagógica y administrativa de los cursinhos populares, reduciendo procedimientos manuales, retrabajo e inconsistencias de información. Metodológicamente, la solución fue desarrollada con el framework Django, en Python, adoptando el patrón arquitectónico MVT y recursos ORM para la persistencia de datos. Los resultados muestran una plataforma funcional capaz de automatizar la prematrícula, organizar contenidos por áreas del conocimiento, controlar accesos por perfiles de usuario y ofrecer paneles administrativos y estudiantiles más intuitivos. Se concluye que el sistema constituye una herramienta viable para fortalecer iniciativas de educación popular y mejorar la gestión educativa.

Palabras clave: Desarrollo Web. Django. Ingeniería de Software. Gestión Educativa. Cursinho Popular.

1 INTRODUÇÃO

O acesso ao ensino superior no Brasil ainda representa um desafio para estudantes oriundos de escolas públicas e de famílias de baixa renda, especialmente no que se refere à preparação para processos seletivos de universidades públicas. Nesse contexto, os cursinhos populares desempenham relevante função social ao ofertarem suporte educacional gratuito e contribuírem para a democratização do acesso à educação (GROPPO; OLIVEIRA; OLIVEIRA, 2019).

Na Universidade do Estado do Pará (UEPA), Campus de Marabá, destaca-se o Projeto Social Cursinho Popular AVANTE, organizado por acadêmicos voluntários de diferentes cursos. O projeto oferece aulas preparatórias aos fins de semana para estudantes que não possuem condições financeiras de arcar com cursinhos particulares, utilizando a própria infraestrutura universitária e caracterizando-se como ação de extensão voltada à inclusão educacional (FORPROEX, 2012).

Apesar de sua relevância social, o Cursinho AVANTE enfrenta dificuldades relacionadas à organização administrativa e à gestão das informações. O processo de matrícula é realizado com apoio de ferramentas do Google, como Drive, Docs, Meet e Forms, e também por meio de procedimentos presenciais e manuais, o que favorece atrasos, retrabalho, inconsistência de dados e dificuldades de comunicação entre secretaria, professores e alunos (LAUDON; LAUDON, 2014).

Além disso, a ausência de um sistema informatizado dificulta o acompanhamento das turmas, o controle de frequência, a distribuição de materiais didáticos e a divulgação de avisos, impactando diretamente a dinâmica do cursinho. Estudos apontam que o uso de sistemas de informação na área educacional contribui para a melhoria da organização administrativa e do fluxo de comunicação institucional (BANCO INTERAMERICANO DE DESENVOLVIMENTO, 2021).

Diante desse cenário, este trabalho propõe o desenvolvimento do Sistema Walk, um sistema de gestão web para cursinho pré-universitário integrado a um portal para alunos, professores e equipe administrativa. A proposta busca automatizar a pré-matrícula, facilitar o gerenciamento de turmas e disciplinas e melhorar a comunicação entre os envolvidos, alinhando-se às abordagens tecnológicas aplicadas à gestão educacional descritas na literatura.

O objetivo consiste em projetar e implementar um sistema de gestão web para cursinho pré-universitário utilizando o framework Django. Como objetivos específicos, destacam-se: analisar requisitos funcionais e não funcionais; modelar a estrutura do banco de dados e a arquitetura do sistema; desenvolver funcionalidades para cadastro e gerenciamento de alunos, professores, turmas e conteúdos; implementar mecanismos de autenticação e controle de acesso; realizar testes de funcionalidade e usabilidade; e documentar o desenvolvimento e os resultados obtidos (PRESSMAN; MAXIM, 2016; SOMMERVILLE, 2011).

2 REFERENCIAL TEÓRICO

O referencial teórico deste trabalho articula o contexto da extensão universitária com a adoção de sistemas de informação e de tecnologias contemporâneas de desenvolvimento web, oferecendo base conceitual para a solução proposta.

2.1 EXTENSÃO UNIVERSITÁRIA E INCLUSÃO DIGITAL

A extensão universitária constitui um dos pilares do ensino superior brasileiro, funcionando como ponte entre a produção acadêmica e as demandas sociais (FORPROEX, 2012). Em projetos como o Cursinho Popular AVANTE, a extensão cumpre papel de justiça social ao democratizar o acesso ao conhecimento. Para que essas iniciativas se mantenham sustentáveis, contudo, a gestão eficiente torna-se fundamental. A introdução de ferramentas tecnológicas em ações extensionistas otimiza processos e favorece a inclusão digital de organizadores e beneficiários.

2.2 SISTEMAS DE INFORMAÇÃO NA GESTÃO EDUCACIONAL

Sistemas de Informação são conjuntos de componentes inter-relacionados que coletam, processam, armazenam e distribuem informações para apoiar a tomada de decisão e o controle organizacional (LAUDON; LAUDON, 2014). No ambiente educacional, a transição do registro manual para o digital reduz retrabalho, minimiza erros humanos e permite que a coordenação pedagógica concentre esforços em atividades estratégicas, enquanto tarefas burocráticas passam a ser automatizadas (BANCO INTERAMERICANO DE DESENVOLVIMENTO, 2021).

2.3 DESENVOLVIMENTO WEB COM O FRAMEWORK DJANGO

Para implementar sistemas robustos e escaláveis, a escolha da tecnologia é decisiva. O Django é um framework web de alto nível, escrito em Python, que incentiva desenvolvimento rápido e design limpo. Seu padrão MVT organiza a aplicação em modelos, visualizações e templates; o ORM simplifica a comunicação com bancos relacionais; e o ecossistema do framework oferece recursos nativos de segurança, administração e autenticação que favorecem a construção de aplicações educacionais confiáveis (FORCIER; BISSEX; CHUN, 2008; HOLOVATY; KAPLAN-MOSS, 2009).

3 METODOLOGIA

Este trabalho caracteriza-se como pesquisa aplicada de natureza tecnológica, com abordagem qualitativa e quantitativa. Quanto aos objetivos, classifica-se como exploratória e descritiva, pois busca compreender as necessidades de cursinhos populares e propor uma solução tecnológica adequada ao contexto investigado (GIL, 2017). O método adotado segue princípios do desenvolvimento ágil de

software, permitindo iterações, testes e ajustes contínuos ao longo da construção do sistema (PRESSMAN; MAXIM, 2016).

3.1 LEVANTAMENTO DE REQUISITOS

O levantamento de requisitos foi organizado em requisitos funcionais e não funcionais, conforme apresentado no Quadro 1, evidenciando as funcionalidades e os serviços disponibilizados aos usuários. No Sistema Walk, tais requisitos foram definidos a partir da análise das demandas operacionais do cursinho pré-universitário, considerando diferentes perfis de acesso: administrador, professor e aluno.

Quadro 1. Requisitos funcionais (RF)

Código	Descrição	Categoria
RF01	O sistema deve permitir o cadastro de alunos com informações pessoais e número de matrícula.	Cadastro de usuários
RF02	O sistema deve permitir o cadastro de professores com suas respectivas especializações.	Cadastro de usuários
RF03	O sistema deve possibilitar o gerenciamento de disciplinas e turmas.	Gestão acadêmica
RF04	O sistema deve permitir a matrícula de alunos em disciplinas específicas.	Gestão acadêmica
RF05	O sistema deve possibilitar o upload de materiais didáticos (PDFs e documentos).	Gestão de conteúdo
RF06	O sistema deve permitir o cadastro de links para aulas gravadas (YouTube e Vimeo).	Gestão de conteúdo
RF07	O sistema deve oferecer um blog para publicação de notícias e avisos.	Comunicação
RF08	O sistema deve implementar autenticação diferenciada para três perfis: aluno, professor e administrador.	Segurança e controle de acesso

Fonte: Elaborado pelos autores (2026).

Os requisitos não funcionais apresentados no Quadro 2 definem atributos de qualidade do sistema, isto é, como a solução deve operar. Neste projeto, eles asseguram que a plataforma seja responsiva e acessível via navegador, segura por meio de autenticação e controle de acesso, escalável para suportar crescimento do número de usuários e fácil de usar, com interface intuitiva (PRESSMAN; MAXIM, 2016; SOMMERVILLE, 2011).

Quadro 2. Requisitos não funcionais (RNF)

Código	Descrição	Categoria	Prioridade
RNF01	O sistema deve ser responsivo e acessível via navegadores web.	Usabilidade	Alta
RNF02	O sistema deve garantir a segurança dos dados por meio de autenticação e controle de acesso.	Segurança	Crítica

RNF03	O sistema deve ser escalável para suportar o crescimento do número de usuários.	Performance	Alta
RNF04	O sistema deve apresentar interface intuitiva e de fácil utilização.	Usabilidade	Alta

Fonte: Elaborado pelos autores (2026).

3.2 ARQUITETURA DE DESENVOLVIMENTO

A arquitetura do Sistema Walk foi definida para atender a um contexto educacional com múltiplos perfis de usuários, preservando organização do código, segurança, manutenção e evolução incremental. Para isso, adotou-se uma arquitetura cliente-servidor típica de aplicações web, estruturada em camadas de apresentação, aplicação e dados, abordagem recomendada na Engenharia de Software por favorecer separação de responsabilidades (PRESSMAN; MAXIM, 2016; SOMMERVILLE, 2011).

3.2.1 Organização em Camadas

Na camada de apresentação concentram-se as páginas HTML e os componentes estáticos responsáveis pela interação com o usuário. A camada de aplicação abriga fluxos de uso, validações e regras de negócio. Já a camada de dados contempla o armazenamento relacional e o acesso mediado pelo ORM do Django. Essa separação facilita testes, manutenção e evolução do software.

3.2.2 Padrão MVT do Django e Organização do Projeto

O framework Django adota o padrão MVT (Model-View-Template), equivalente em intenção ao MVC. No Walk, os modelos definem as entidades do domínio; as views implementam casos de uso, validações e fluxos de resposta; os templates estruturam as telas; e o roteamento por URLs organiza o acesso às funcionalidades. O projeto foi particionado em aplicativos por responsabilidade funcional, o que aumenta modularidade e reaproveitamento (HOLOVATY; KAPLAN-MOSS, 2009).

3.2.3 Persistência e Acesso a Dados

A camada de dados utiliza banco relacional com acesso mediado pelo ORM do Django, em alinhamento com a ideia de Data Mapper e com padrões de arquitetura de aplicações empresariais (FOWLER, 2002). Na prática, as migrações versionam o esquema do banco, as operações CRUD são realizadas por modelos e querysets, e os relacionamentos são explicitamente modelados, garantindo integridade e produtividade no desenvolvimento.

3.2.4 Segurança, Autenticação e Controle de Acesso

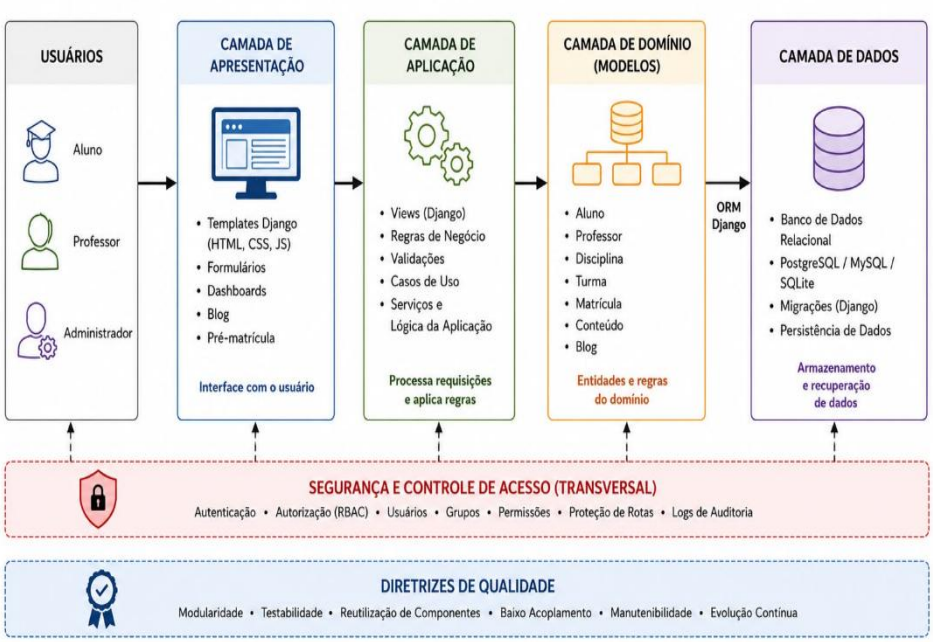
Para tratar dados pessoais e acadêmicos, o Walk implementa mecanismos de autenticação e autorização baseados em perfis, aproximando-se do modelo RBAC, no qual permissões são atribuídas

a papéis e os usuários herdam autorizações conforme sua função (SANDHU et al., 1996). No Django, essa estratégia é operacionalizada por usuários, grupos, permissões, proteção de rotas e restrição de operações sensíveis.

3.2.5 Diretrizes de Qualidade

A arquitetura foi pensada para sustentar evolução incremental, testabilidade, modularidade e manutenibilidade, conforme demonstrado na Figura 1. O isolamento de responsabilidades entre camadas e aplicativos, associado ao uso de padrões de projeto quando necessário, favorece previsibilidade de comportamento, reutilização de componentes e menor acoplamento (GAMMA et al., 1994; MARTIN, 2009).

Figura 1. Arquitetura em camadas do Sistema Walk.

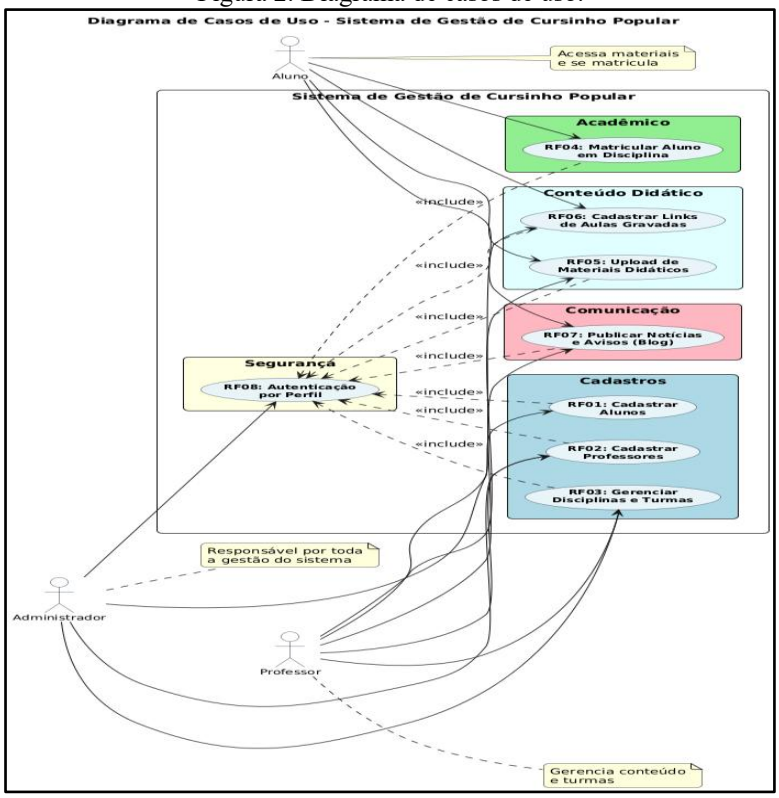


Fonte: Elaborada pelos autores (2026).

3.2.6 Diagrama de Casos de Uso

A especificação dos casos de uso permite compreender os fluxos de eventos e as regras de negócio aplicadas. O diagrama a seguir (Figura 2) resume as principais interações do Sistema Walk entre administradores, professores e alunos (PRESSMAN; MAXIM, 2016).

Figura 2. Diagrama de casos de uso.



Fonte: Elaborada pelos autores (2026).

Os atores representam os papéis desempenhados pelos usuários ao interagirem com o sistema, apresentados no Quadro 3.

Quadro 3. Atores do sistema

Ator	Descrição	Responsabilidades
Administrador	Gestor do sistema com permissões totais.	Gerenciar cadastros de usuários, disciplinas, turmas e supervisionar funcionalidades.
Professor	Docente responsável pelo conteúdo pedagógico.	Realizar upload de materiais, cadastrar links de videoaulas e publicar comunicados no blog.
Aluno	Estudante matriculado em uma das turmas.	Realizar matrículas em disciplinas, acessar materiais didáticos e visualizar aulas gravadas.

Fonte: Elaborado pelos autores (2026).

Com base na definição dos atores do sistema, foram identificados os principais casos de uso necessários para atender aos requisitos funcionais da aplicação. O Quadro 4 apresenta Caso de Uso do Sistema e seus respectivos Requisitos Funcionais.

Quadro 4. Casos de uso do sistema

Caso de uso	Ator(es)	Requisito	Descrição
UC01 - Cadastrar aluno	Administrador	RF01	Registro de novos alunos com dados pessoais e matrícula.

UC02 - Cadastrar professor	Administrador	RF02	Registro de professores e suas especializações.
UC03 - Gerenciar disciplinas	Administrador	RF03	Criação, edição e exclusão de disciplinas e turmas.
UC04 - Matricular em disciplinas	Aluno e administrador	RF04	Vinculação de alunos a turmas específicas.
UC05 - Upload de materiais	Professor e administrador	RF05	Disponibilização de PDFs e documentos didáticos.
UC06 - Cadastrar links	Professor e administrador	RF06	Inclusão de links de aulas gravadas.
UC07 - Publicar no blog	Professor e administrador	RF07	Publicação de notícias e avisos institucionais.
UC08 - Autenticar no sistema	Todos os atores	RF08	Controle de acesso com perfis diferenciados.

Fonte: Elaborado pelos autores (2026).

3.2.7 Hierarquia de Permissões e Segurança

O sistema foi projetado sob uma estrutura hierárquica de acesso. O administrador possui permissão total para executar todos os casos de uso; o professor atua em funcionalidades relacionadas ao gerenciamento pedagógico e à comunicação; e o aluno acessa recursos voltados à matrícula, consulta e consumo de conteúdo. Como regras transversais, nenhum recurso é acessado sem autenticação, a interface adapta-se dinamicamente ao perfil do usuário e ações críticas devem ser rastreáveis por registros de log.

3.3 AVALIAÇÃO HEURÍSTICA DE NIELSEN

A avaliação heurística foi aplicada ao Sistema Walk com o objetivo de identificar problemas de usabilidade capazes de comprometer a interação dos usuários com a plataforma. Foram analisadas as principais interfaces do sistema, incluindo a página inicial, o formulário de pré-matrícula, a tela de autenticação, o blog institucional, o painel administrativo e o painel do aluno.

A avaliação heurística foi conduzida pelos autores do trabalho, que atuaram como avaliadores a partir dos perfis de uso previstos para o sistema. Durante a inspeção, foram simuladas ações reais, como realizar login, preencher o formulário de pré-matrícula, acessar conteúdos, consultar informações, navegar entre menus e observar mensagens de retorno. As inconsistências foram registradas quanto à navegação, clareza das ações, prevenção de erros e mensagens de retorno. A gravidade foi classificada com base na escala de 0 a 4 apresentada no Quadro 5, considerando o impacto de cada problema na execução da tarefa, sua frequência potencial e a dificuldade imposta ao usuário para concluir a ação.

Foram utilizadas como referência as 10 heurísticas de usabilidade propostas por Nielsen: visibilidade do status do sistema; correspondência entre o sistema e o mundo real; controle e liberdade

do usuário; consistência e padrões; prevenção de erros; reconhecimento em vez de memorização; flexibilidade e eficiência de uso; estética e design minimalista; ajuda aos usuários para reconhecer, diagnosticar e corrigir erros; e ajuda e documentação. Segundo Nielsen (1994), a avaliação heurística consiste em um método no qual um pequeno conjunto de avaliadores examina a interface e julga sua conformidade com princípios reconhecidos de usabilidade.

A gravidade dos problemas encontrados foi classificada em uma escala de 0 a 4, conforme apresentado no Quadro 5.

Quadro 5. Escala de gravidade utilizada na avaliação heurística.

Gravidade	Classificação	Descrição
0	Sem problema	Não há falha de usabilidade identificada.
1	Leve	Problema cosmético que não compromete a execução da tarefa.
2	Moderado	Problema que afeta parcialmente a experiência do usuário.
3	Grave	Problema que dificulta a execução da tarefa.
4	Crítico	Problema que impede o usuário de concluir a ação.

Fonte: Elaborado pelos autores (2026).

A partir dessa classificação, cada problema identificado foi relacionado à heurística violada, à tela avaliada, ao impacto na experiência do usuário e à respectiva proposta de melhoria.

4 RESULTADOS E DISCUSSÃO

Esta seção apresenta os resultados obtidos com o desenvolvimento e a implementação do Sistema Walk, com foco nas interfaces e nos módulos efetivamente entregues. As evidências foram organizadas por telas e funcionalidades, permitindo observar a aderência da solução aos requisitos levantados e seu potencial de apoio à gestão de cursinhos populares.

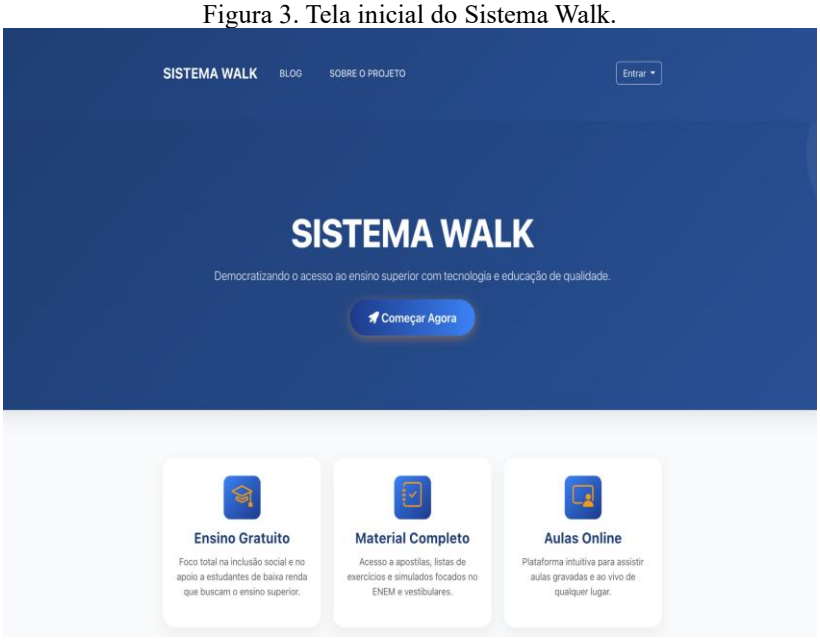
De modo geral, as telas implementadas demonstram aderência aos requisitos funcionais definidos no levantamento: a interface pública contribui para a comunicação institucional; o módulo de pré-matrícula organiza a coleta de dados dos candidatos; a autenticação e os perfis de acesso atendem ao controle de segurança; o blog fortalece a divulgação de avisos e conteúdos; o painel administrativo apoia a gestão de alunos, professores, disciplinas e matrículas; e o painel do aluno centraliza materiais e videoaulas. Assim, os resultados indicam que os módulos desenvolvidos reduzem processos manuais, organizam informações e apoiam a gestão pedagógica do cursinho.

4.1 INTERFACE PÚBLICA E NAVEGAÇÃO INICIAL

A interface pública do sistema apresenta de forma clara a proposta do projeto e os caminhos de acesso aos perfis disponíveis. A página inicial reforça a identidade visual da plataforma e disponibiliza

chamadas para ação, blog e seção sobre o projeto, favorecendo transparência e comunicação institucional, conforme representado na Figura 3.

Figura 3. Tela inicial do Sistema Walk.

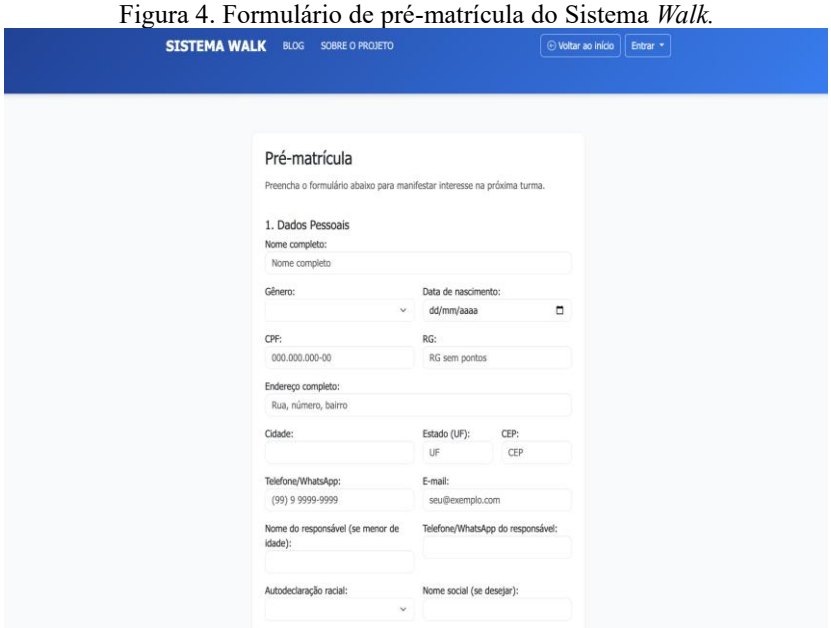


Fonte: Elaborada pelos autores (2026).

4.2 MÓDULO DE PRÉ-MATRÍCULA E COLETA DE DADOS

Um dos resultados centrais do sistema é o módulo de pré-matrícula, que registra o interesse do candidato em uma nova turma por meio de formulário estruturado. Na Figura 4, observa-se a coleta de dados pessoais e de contato, além de campos relevantes para indicadores de inclusão e gestão, como nome social e autodeclaração racial, quando aplicável.

Figura 4. Formulário de pré-matrícula do Sistema *Walk*.



Fonte: Elaborada pelos autores (2026).

4.3 AUTENTICAÇÃO, PERFIS DE ACESSO E SEGURANÇA

Conforme apresentado na Figura 5, a autenticação foi implementada por meio de tela de login objetiva, com fluxo de entrada simples e direcionamento do usuário conforme seu perfil. Essa estratégia reduz riscos de acesso indevido, preserva a integridade das informações e organiza responsabilidades operacionais dentro do ambiente digital.

Figura 5. Tela de autenticação (*login*) do Sistema *Walk*.



Fonte: Elaborada pelos autores (2026).

4.4 BLOG INSTITUCIONAL: NOTÍCIAS E CONTEÚDOS

O módulo de blog, apresentado na Figura 6, foi estruturado para publicação e categorização de notícias, avisos, eventos e conteúdos preparatórios. Mesmo quando não há publicações ativas, a interface comunica o estado do módulo ao usuário, contribuindo para uma experiência consistente e para o fortalecimento da comunicação contínua com os alunos.

Figura 6. Módulo de blog do Sistema *Walk*.

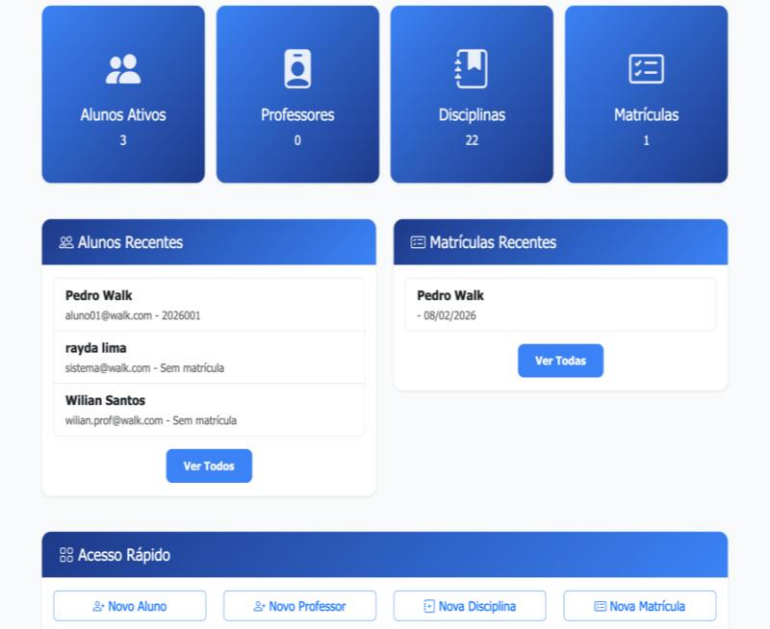


Fonte: Elaborada pelos autores (2026).

4.5 PAINEL ADMINISTRATIVO E INDICADORES OPERACIONAIS

No contexto da gestão, foi desenvolvido um painel administrativo com visão rápida do estado do sistema. O dashboard, demonstrado na Figura 7, apresenta indicadores como quantidade de alunos ativos, professores cadastrados, disciplinas disponíveis e matrículas registradas, além de atalhos para criação de entidades essenciais e acompanhamento de registros recentes.

Figura 7. Dashboard administrativo do Sistema *Walk* (indicadores e atalhos).

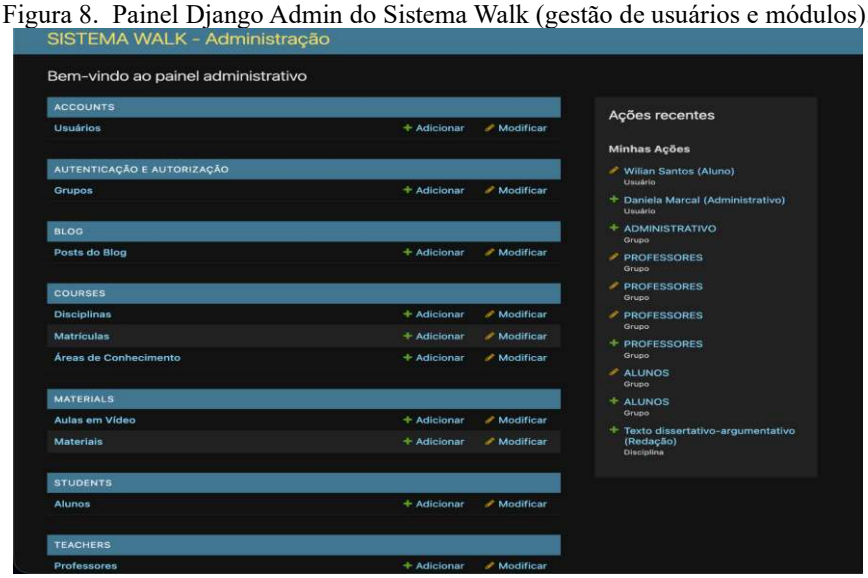


Fonte: Elaborada pelos autores (2026).

4.6 ADMINISTRAÇÃO VIA DJANGO ADMIN

Como resultado da implementação apresentada na Figura 8, o sistema utiliza o Django Admin para administração centralizada de dados, incluindo gerenciamento de usuários, grupos, permissões e entidades do domínio. Esse recurso favorece consistência, rastreabilidade e manutenção, uma vez que integra operações de cadastro, consulta e atualização com os mecanismos nativos do framework.

Figura 8. Painel Django Admin do Sistema Walk (gestão de usuários e módulos).

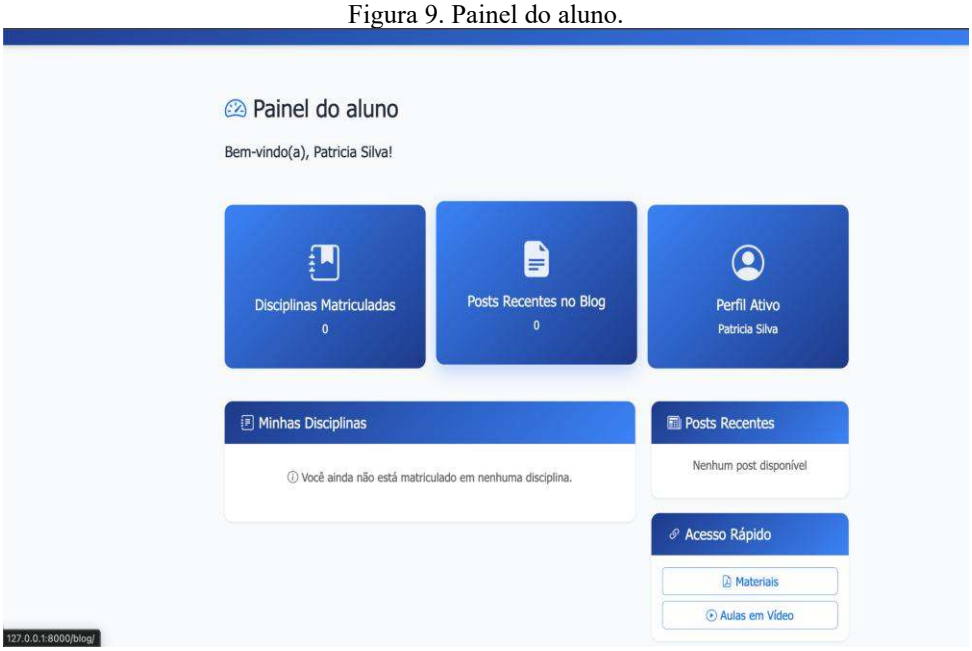


Fonte: Elaborada pelos autores (2026).

4.7 PAINEL DO ALUNO E ORGANIZAÇÃO POR ÁREAS DO CONHECIMENTO

A interface destinada ao estudante centraliza o acesso pedagógico de forma intuitiva. O painel do aluno apresenta indicadores do perfil, disciplinas matriculadas, publicações recentes e atalhos para materiais e videoaulas, conforme Figura 9. A organização dos conteúdos por áreas do conhecimento aproxima a plataforma da lógica de estudos do ENEM e de vestibulares, facilitando a navegação e a construção de rotinas de estudo.

Figura 9. Painel do aluno.



Fonte: Elaborada pelos autores (2026).

4.8 RESULTADOS DO TESTE HEURÍSTICO

A avaliação heurística realizada no Sistema Walk permitiu identificar problemas de usabilidade relacionados à navegação, à comunicação de estados do sistema, à prevenção de erros e à clareza das

ações disponíveis ao usuário. A inspeção contemplou as principais telas implementadas, especialmente aquelas associadas à pré-matrícula, autenticação, comunicação institucional, administração e acesso do estudante aos conteúdos.

Foram identificados oito problemas de usabilidade distribuídos entre as heurísticas analisadas. Observou-se maior concentração de ocorrências na heurística H1, relacionada à visibilidade do status do sistema, indicando a necessidade de reforçar mensagens de confirmação, erro e retorno após ações executadas pelo usuário, conforme apresentado nos Quadros 6 e 7.

Quadro 6. Problemas de usabilidade identificados na avaliação heurística.

Nº	Tela avaliada	Problema identificado	Heurística violada	Gravidade	Sugestão de melhoria
1	Tela de login	O sistema não informa claramente quando o login falha.	H9 - Reconhecer e corrigir erros	3	Exibir mensagem objetiva, como “e-mail ou senha inválidos”.
2	Formulário de pré-matrícula	Alguns campos obrigatórios não estão destacados.	H5 - Prevenção de erros	2	Marcar campos obrigatórios com asterisco e validação visual.
3	Painel do aluno	Algumas opções podem não estar suficientemente visíveis.	H6 - Reconhecimento em vez de memorização	2	Usar ícones e rótulos mais claros para materiais, disciplinas e avisos.
4	Blog institucional	Quando não há publicações, a tela pode parecer vazia.	H1 - Visibilidade do status do sistema	1	Exibir mensagem informando que ainda não há notícias cadastradas.
5	Painel administrativo	Muitos atalhos podem gerar sobrecarga visual.	H8 - Estética e design minimalista	2	Agrupar funcionalidades por categorias.
6	Formulário de pré-matrícula	Ausência de confirmação visual após envio.	H1 - Visibilidade do status do sistema	3	Mostrar mensagem de sucesso após a inscrição.
7	Navegação geral	Falta de botão de retorno em algumas telas.	H3 - Controle e liberdade do usuário	2	Inserir botão “Voltar” ou trilha de navegação.
8	Painel administrativo	Termos técnicos podem dificultar o uso por usuários leigos.	H2 - Correspondência entre sistema e mundo real	2	Substituir termos técnicos por linguagem próxima ao contexto do usuário.

Fonte: Elaborado pelos autores (2026).

Para a análise quantitativa, foi calculada a porcentagem de problemas associados a cada heurística, considerando o total de oito ocorrências identificadas. A fórmula utilizada foi: porcentagem = quantidade de problemas da heurística / total de problemas × 100.

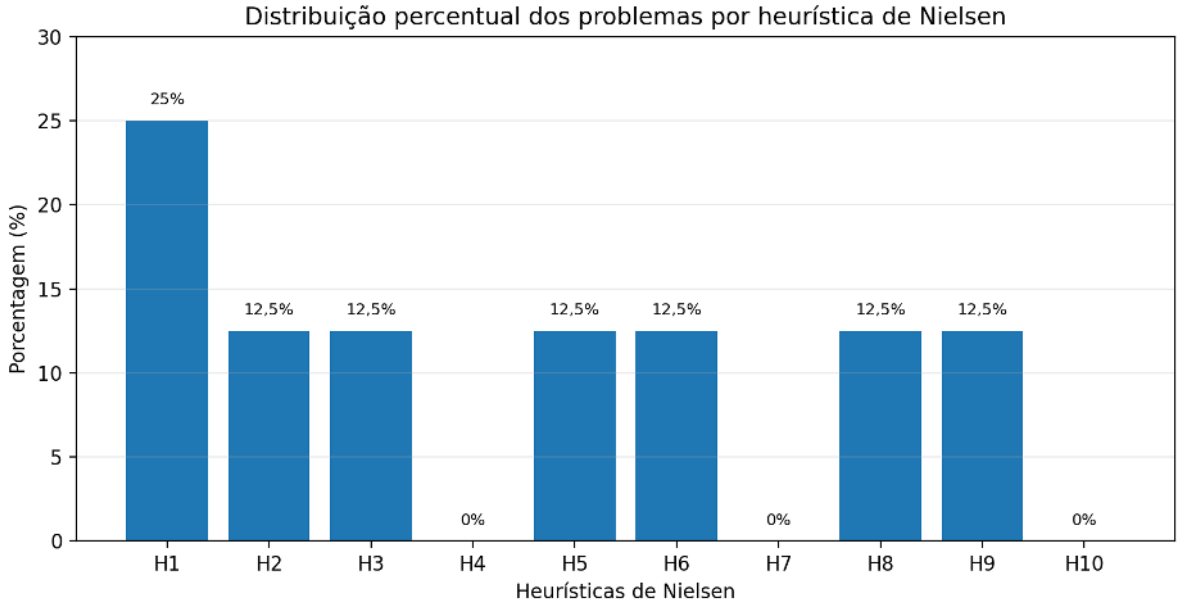
Quadro 7. Distribuição dos problemas por heurística de Nielsen.

Heurística de Nielsen	Quantidade de problemas	Porcentagem
H1 - Visibilidade do status do sistema	2	25%
H2 - Correspondência entre sistema e mundo real	1	12,5%
H3 - Controle e liberdade do usuário	1	12,5%
H4 - Consistência e padrões	0	0%
H5 - Prevenção de erros	1	12,5%
H6 - Reconhecimento em vez de memorização	1	12,5%
H7 - Flexibilidade e eficiência de uso	0	0%
H8 - Estética e design minimalista	1	12,5%
H9 - Ajudar usuários a reconhecer e corrigir erros	1	12,5%
H10 - Ajuda e documentação	0	0%
Total	8	100%

Fonte: Elaborado pelos autores (2026).

A Figura 10 apresenta a distribuição percentual dos problemas por heurística. A concentração em H1 evidencia que o sistema deve aprimorar os retornos visuais ao usuário, especialmente em ações como envio de formulário, autenticação e consulta de informações.

Figura 10. Distribuição percentual dos problemas por heurística de Nielsen.



Fonte: Elaborada pelos autores (2026).

De modo geral, os resultados indicam que o Sistema Walk apresenta uma estrutura funcional adequada, mas pode evoluir em aspectos de feedback, clareza textual e organização visual. As

melhorias propostas contribuem para reduzir dúvidas durante o uso, prevenir erros e tornar a experiência mais intuitiva para administradores, professores e alunos.

4.9 SÍNTESE DOS RESULTADOS

De forma geral, as telas implementadas evidenciam que o Sistema Walk alcançou uma arquitetura funcional orientada à gestão do cursinho e ao suporte ao estudante, contemplando desde o primeiro contato com a plataforma até processos internos de controle, comunicação e administração. Observa-se aderência aos requisitos funcionais e não funcionais, com ênfase em usabilidade, organização pedagógica, segurança e rastreabilidade. Como próximos passos, recomenda-se ampliar a validação com usuários reais por meio de testes de usabilidade, avaliação de satisfação e consolidação de rotinas de relatórios e auditoria.

5 CONCLUSÃO

O Sistema Walk foi desenvolvido para apoiar a gestão de cursinhos populares, articulando tecnologia educacional e compromisso social. A solução implementada no ecossistema Python, com o framework Django, oferece interface pública informativa, módulo de pré-matrícula, autenticação por perfis e painéis específicos para administração e estudantes. Os resultados indicam que o sistema reduz esforço operacional, melhora a organização das informações e amplia o suporte pedagógico, constituindo-se como ferramenta viável para fortalecer iniciativas de educação popular e aproximar universidade e comunidade.

5.1 IMPLEMENTAÇÕES FUTURAS

Considerando a arquitetura escalável em Django, as versões futuras do sistema poderão incorporar otimização do banco de dados para suportar maior volume de acessos simultâneos, módulo de simulados on-line com correção automatizada e sistema de trilhas de aprendizagem personalizadas com base no desempenho dos alunos. Tais melhorias tendem a ampliar o impacto social da plataforma e sua capacidade de replicação em outros cursinhos populares da região amazônica.

REFERÊNCIAS

BANCO INTERAMERICANO DE DESENVOLVIMENTO. Os Sistemas de Informação e Gestão Educacional (SIGED) da América Latina e do Caribe: o caminho para a transformação digital da gestão educacional. Washington, DC: BID, 2021. Disponível em: https://publications.iadb.org/publications/portuguese/document/Os-Sistemas-de-Informacao-e-Gestao-Educacional-SIGED-da-America-Latina-e-do-Caribe-o-caminho-para-a-transformacao-digital-da-gestao-educacional.pdf. Acesso em: 27 abr. 2026.

FORCIER, Jeff; BISSEX, Paul; CHUN, Wesley J. Python web development with Django. Boston: Addison-Wesley Professional, 2008. Disponível em: https://books.google.com/books/about/Python_Web_Development_with_Django.html?id=M2D5nnYlmZoC. Acesso em: 27 abr. 2026.

FORPROEX. Política nacional de extensão universitária. Manaus: Fórum de Pró-Reitores de Extensão das Instituições Públicas de Educação Superior Brasileiras, 2012. Disponível em: https://proex.ufsc.br/files/2016/04/Pol%C3%ADtica-Nacional-de-Extens%C3%A3o-Universit%C3%A1ria-e-book.pdf. Acesso em: 27 abr. 2026.

FOWLER, Martin. Padrões de arquitetura de aplicações empresariais. Boston: Addison-Wesley, 2002. Disponível em: https://books.google.com.br/books?id=8uSN0Qwz2kQC&printsec=copyright. Acesso em: 27 abr. 2026.

GAMMA, Erich et al. Design patterns: elements of reusable object-oriented software. Boston: Addison-Wesley, 1994. Disponível em: https://books.google.com/books/about/Design_Patterns.html?id=6oHuKQe3TjQC. Acesso em: 27 abr. 2026.

GIL, Antonio Carlos. Como elaborar projetos de pesquisa. 6. ed. São Paulo: Atlas, 2017. Disponível em: https://books.google.com/books/about/Como_Elaborar_Projetos_de_Pesquisa.html?id=YCv4zwEACAAJ. Acesso em: 27 abr. 2026.

GROPPO, Luís Antonio; OLIVEIRA, Ana Rosa Garcia de; OLIVEIRA, Fabiana Mara de. Cursinho popular por estudantes da universidade: práticas político-pedagógicas e formação docente. Revista Brasileira de Educação, Rio de Janeiro, v. 24, e240031, p. 1-24, 2019. DOI: https://doi.org/10.1590/S1413-24782019240031. Disponível em: https://www.scielo.br/j/rbedu/a/WCL98kd9VJkHktFSTKwxcmP/. Acesso em: 27 abr. 2026.

HOLOVATY, Adrian; KAPLAN-MOSS, Jacob. The definitive guide to Django. 2. ed. Berkeley: Apress, 2009. Disponível em: https://books.google.com/books/about/The_Definitive_Guide_to_Django.html?id=hc9CAAAAQBAJ. Acesso em: 27 abr. 2026.

LAUDON, Kenneth C.; LAUDON, Jane P. Sistemas de informação gerenciais. 11. ed. São Paulo: Pearson Education do Brasil, 2014. Disponível em: https://books.google.com.br/books?hl=pt-BR&id=NIehEAAAQBAJ&printsec=frontcover&source=gbs_atb. Acesso em: 27 abr. 2026.

MARTIN, Robert C. Clean code: a handbook of agile software craftsmanship. Upper Saddle River: Prentice Hall, 2009. Disponível em: https://books.google.com/books/about/Clean_Code.html?id=_i6bDeoCQzsC. Acesso em: 27 abr. 2026.

NIELSEN, Jakob; MACK, Robert L. (Org.). Usability Inspection Methods. New York: John Wiley & Sons, 1994. Disponível em: https://books.google.com/books?id=cuRQAAAAMAAJ. Acesso em: 27 abr. 2026.

PRESSMAN, Roger S.; MAXIM, Bruce R. Engenharia de software: uma abordagem profissional. 8. ed. Porto Alegre: AMGH, 2016. Disponível em: https://books.google.com/books/about/Engenharia_De_Software.html?hl=pt-BR&id=smNFvgAACAAJ. Acesso em: 27 abr. 2026.

SANDHU, Ravi S. et al. Role-based access control models. Computer, v. 29, n. 2, p. 38-47, 1996. Disponível em: https://csrc.nist.gov/csrc/media/projects/role-based-access-control/documents/sandhu96.pdf. Acesso em: 27 abr. 2026.

SOMMERVILLE, Ian. Engenharia de software. 9. ed. São Paulo: Pearson, 2011. Disponível em: https://books.google.com/books/about/Software_Engineering.html?hl=pt-BR&id=fSYrAAAAQBAJ. Acesso em: 27 abr. 2026.