

Documentação Testes

Sistema Acadêmico Colaborativo com Apoio de IA - PIM

1. Introdução

A presente documentação descreve o processo de planejamento, execução e validação dos testes realizados no Sistema Acadêmico Colaborativo, desenvolvido como parte do Projeto Integrado Multidisciplinar (PIM II) do curso de Análise e Desenvolvimento de Sistemas.

O objetivo desta etapa foi garantir a qualidade, confiabilidade e conformidade do sistema em relação aos requisitos funcionais e não funcionais definidos durante o desenvolvimento.

Foram testados os principais módulos de operação, comunicação e geração de relatórios, assegurando que o sistema atenda corretamente às necessidades propostas e opere de forma estável em ambiente de rede local simulada.

1.1. Objetivo da Documentação de Testes

O objetivo desta documentação é registrar e detalhar os procedimentos de teste aplicados ao projeto, apresentando os resultados obtidos e a análise de conformidade com os requisitos especificados.

Busca-se demonstrar que as funcionalidades implementadas, como cadastro de alunos e turmas, controle de usuários, geração de relatórios e integração entre módulos em C e Python, funcionam corretamente e conforme as expectativas do projeto.

Além disso, este documento visa apoiar a rastreabilidade do desenvolvimento, servindo como evidência da verificação e validação do sistema.

1.2. Escopo dos Testes

Os testes abrangeram as principais funcionalidades do sistema, incluindo o backend em C (manipulação e persistência de dados em arquivos .csv) e o frontend em Python (interface e relatórios digitais).

Foram realizadas verificações de cadastro, autenticação, leitura de dados, geração de relatórios com IA Manual e Gemini, além de simulações de execução em rede local cliente-servidor.

Aspectos de desempenho, integridade dos dados, usabilidade e sustentabilidade digital também foram observados, garantindo o alinhamento com os requisitos não funcionais definidos.

1.3. Metodologia de Testes

A metodologia de testes adotada baseou-se em testes funcionais, integrados e de aceitação, conduzidos ao final de cada sprint do ciclo ágil Scrum.

Os testes foram planejados de forma incremental, acompanhando o avanço das entregas e priorizando as funcionalidades críticas do sistema.

Cada caso de teste foi descrito de maneira estruturada, contendo identificador (CT), ação testada, resultado esperado e status final (OK/Erro).

Os resultados foram documentados em tabela e validados pela equipe de desenvolvimento, garantindo transparência, rastreabilidade e qualidade do produto final.

2. Planejamento de Testes

O planejamento dos testes teve como finalidade definir a estratégia, o ambiente e as ferramentas necessárias para validar as funcionalidades e o desempenho do sistema, garantindo confiabilidade e conformidade com os requisitos estabelecidos.

2.1. Estratégia de Testes

A estratégia adotada baseou-se em testes funcionais, integrados e de aceitação, aplicados ao final de cada sprint do ciclo ágil. Foram priorizados os módulos críticos, como cadastro de usuários, autenticação, registro de notas e geração de relatórios com IA.

Cada teste foi estruturado com entrada controlada e resultado esperado, permitindo verificar o comportamento do sistema em diferentes cenários e validar a integração entre o backend em C e o frontend em Python.

3. Casos de Testes

3.1. Identificação e Estrutura dos Casos

Cada caso de teste foi identificado por um código (CT) e estruturado em quatro campos: Ação Testada, Resultado Esperado, Resultado Obtido e Status.

Os testes foram planejados para validar as principais funcionalidades do sistema, abrangendo operações de cadastro, autenticação, geração de relatórios e integração entre linguagens.

3.2. Tabela de Casos de Teste e Resultados

A seguir são apresentados os casos de teste divididos por categoria funcional, abrangendo as principais operações do sistema acadêmico colaborativo.

A. Testes de Cadastro e Acesso

Caso	Ação Testada	Resultado Esperado	Status
CT01	Cadastro de aluno	Dados gravados corretamente no arquivo .csv	OK
CT02	Cadastro de professor	Registro criado e identificado no arquivo de docentes	OK
CT03	Cadastro de turma	Associação entre turma e professor armazenada corretamente	OK
CT04	Login de usuário	Acesso concedido conforme nível (aluno/professor/admin)	OK
CT05	Edição e exclusão de cadastro	Alterações refletidas corretamente nos arquivos	OK

B. Testes do Módulo em C (Backend)

Caso	Ação Testada	Resultado Esperado	Status
CT06	Criação e leitura de arquivos .csv	Arquivos criados, atualizados e lidos corretamente	OK
CT07	Validação de dados de entrada	Sistema impede gravação de campos vazios ou inválidos	OK
CT08	Funções de busca e ordenação	Dados retornados e organizados conforme critérios	OK
CT09	Persistência de dados após reinício	Dados mantidos no arquivo após fechamento do programa	OK

C. Testes do Módulo em Python (Frontend e Interface)

Caso	Ação Testada	Resultado Esperado	Status
CT10	Inicialização da interface	Tela de login carregada corretamente	OK
CT11	Navegação entre telas	Menus, botões e comandos funcionam sem erros	OK
CT12	Exibição de relatórios digitais	Relatórios exibidos corretamente em formato digital	OK

D. Testes de Inteligência Artificial

Caso	Ação Testada	Resultado Esperado	Status
CT14	Geração de relatório (IA Manual)	Relatório local gerado com análise textual básica	OK
CT15	Geração de relatório (API Gemini)	Relatório online com análise detalhada e recomendações	OK
CT16	Comutação automática IA Manual ↔ Gemini	Sistema alterna para IA Manual ao detectar ausência de internet	OK

3.3. Critérios de Aceitação

Cada teste foi considerado aceito quando o resultado obtido coincidiu com o esperado, sem falhas de execução, erros de leitura ou inconsistências nos dados. Todos os casos listados atingiram os critérios de sucesso definidos.

4. Resultados e Análise

4.1. Avaliação dos Resultados Obtidos

Os resultados demonstraram que o sistema opera de forma estável e cumpre os requisitos funcionais e não funcionais. Todas as funcionalidades principais executaram conforme o esperado, garantindo integridade dos dados e eficiência na comunicação entre os módulos em C e Python.

4.2. Não Conformidades e Ajustes

Durante os testes iniciais, foram identificados pequenos ajustes necessários na validação de login e na atualização automática de arquivos, ambos corrigidos nas etapas seguintes. Após as correções, todos os testes foram concluídos com êxito.

4.3. Validação Final

A validação final confirmou que o sistema está apto para uso acadêmico e demonstração técnica, atendendo integralmente aos requisitos definidos e apresentando comportamento consistente em diferentes cenários de teste.

5. Conclusão dos Testes

Os testes realizados comprovaram a estabilidade, confiabilidade e usabilidade do Sistema Acadêmico Colaborativo.

As integrações entre linguagens e a geração de relatórios com IA, apresentaram resultados satisfatórios.

Conclui-se que o sistema cumpre os objetivos propostos e está tecnicamente validado, podendo evoluir para versões futuras com aprimoramento da IA e banco de dados relacional.