

Pontifícia Universidade Católica de
Campinas – PUC Campinas

PROJETO INTEGRADOR II – Engenharia de Software

Time 11

Gustavo Antônio Mariano – 25009767

Leonardo Gambaroni Alves – 25003494

Lucas Rodrigues Xavier – 25000508

Mateus Oliveira – 25001369

Wanderley Vieira - 25007121

Campinas

2025

3.1. Autenticação

Requisitos Funcionais:

RF01: O sistema deve permitir o cadastro de professor com nome, e-mail, celular e senha

RF02: O sistema deve permitir a recuperação de senha por e-mail

RF03: O sistema deve impedir acesso anônimo ou de visitantes

RF04: A primeira página exibida deve ser a de login

Requisitos Não Funcionais:

RNF01: O processo de autenticação deve garantir segurança no acesso

RNF02: A recuperação de senha deve ser realizada apenas via e-mail previamente cadastrado

RNF03: A interface de login deve ser simples e intuitiva

RNF04: O sistema deve iniciar sempre na tela de login

3.2. Gerenciar instituições, disciplinas e turmas

Requisitos Funcionais:

RF05: O professor deve informar a instituição e o curso após cadastro

RF06: O sistema deve permitir criar várias disciplinas e turmas

RF07: O sistema deve enviar e-mail de confirmação ao excluir turmas que possuam notas

RF08: O sistema deve impedir a exclusão de uma instituição enquanto houver disciplinas ou turmas associadas

RF09: O sistema deve obedecer à ordem de exclusão: Turmas, Disciplinas e Instituição, respectivamente

Requisitos Não Funcionais:

RNF05: A exclusão deve ser confiável e controlada por e-mail de confirmação

RNF06: A hierarquia de exclusão deve manter a integridade dos dados

3.3. Cadastro e importação de alunos

Requisitos Funcionais:

RF10: O professor deve poder adicionar, editar e remover alunos da turma

RF11: O sistema deve permitir a importação de alunos via arquivo CSV

RF12: O sistema deve permitir a importação de alunos via arquivo JSON RF13:
O sistema deve usar apenas as duas primeiras colunas do CSV, que são
matricula e nome

RF14: O sistema deve exigir matrícula, id ou RA nos arquivos JSON

RF15: O sistema deve manter os dados atuais ao importar alunos já cadastrados

Requisitos Não Funcionais:

RNF07: A importação deve garantir consistência, sem perda de dados existentes

RNF08: A leitura de arquivos deve aceitar apenas os formatos CSV e JSON válidos

3.4. Criar componente de nota

Requisitos Funcionais:

RF16: O professor deve cadastrar atividades, provas e trabalhos como
componentes de nota

RF17: O sistema deve permitir notas de 0 a 10, com duas casas decimais

RF18: O professor deve informar nome, sigla e descrição do componente RF19:
O sistema deve abrir a tabela de notas em modo visualização para evitar erros

Requisitos Não Funcionais:

RNF09: O sistema deve restringir notas dentro do intervalo de 0 a 10 RNF10:

A tabela deve abrir em modo visualização por padrão, priorizando
segurança contra erros

3.5. Apontar notas

Requisitos Funcionais:

RF20: O sistema deve permitir o lançamento de notas de um componente por vez

RF21: O sistema deve permitir edição livre de todas as notas RF22:

O sistema não deve permitir desfazer alterações já salvas

Requisitos Não Funcionais:

RNF11: O sistema deve garantir consistência das notas, mesmo sem função de desfazer

RNF12: O processo de edição deve ser rápido e confiável

3.6. Painel de auditoria

Requisitos Funcionais:

RF23: O sistema deve registrar cada alteração ou lançamento de nota com data, hora e detalhes

RF24: O sistema deve exibir registros em ordem decrescente RF25:

O painel de auditoria pode ser escondido, mas não desativado

Requisitos Não Funcionais:

RNF13: A auditoria deve ser permanente e imutável

RNF14: O sistema deve apresentar os registros de forma clara e organizada

3.7. Cálculo de nota final

Requisitos Funcionais:

RF26: O professor deve informar uma fórmula para cálculo da nota final

RF27: O sistema deve validar se a fórmula utiliza todas as notas

RF28: O sistema deve calcular automaticamente a nota final na tabela RF29:

O sistema não deve permitir edição manual da nota final calculada

Requisitos Não Funcionais:

RNF15: O cálculo da nota final deve ser preciso e automático

RNF16: O sistema deve validar fórmulas de forma confiável para evitar erros

3.8. Coluna de notas finais ajustadas

Requisitos Funcionais:

RF30: O sistema deve permitir ativar uma coluna extra para notas finais ajustadas

RF31: O professor deve poder editar manualmente as notas finais ajustadas
RF32: O sistema deve arredondar notas ajustadas em múltiplos de 0,5 RF33:
O sistema deve aceitar apenas valores arredondados. Exemplo: 6.0 ou 6.5

Requisitos Não Funcionais:

RNF17: O arredondamento deve ser feito automaticamente em múltiplos de 0,5
RNF18: O ajuste deve preservar consistência com as notas originais