

Requisitos de Sistema

Equipe:

Daniel Muller Rezende - 202065020AC

Matheus dos Reis Casarim - 201765512AB

Mathews Edwirds Gomes Almeida - 201765503AB

Pedro Luiz Bonorino Braga - 202065177A

Coordenador da equipe:

Pedro Luiz Bonorino Braga

Sumário

Sumário	2
1. Requisitos de Sistema	3
Requisitos Funcionais	3
Requisitos Não-Funcionais	5
2. Diagrama de Classe	6
3. Diagrama de Caso de Uso	7

1. Requisitos de Sistema

a) Requisitos Funcionais

- RF01: O sistema deve ser capaz de cadastrar jogadores (nome e senha) e professores (nome, senha e código);
- RF02: O sistema deve armazenar corretamente as perguntas (incluindo outros tipos de desafios) e respostas;
- RF03: O sistema deve apresentar diferentes níveis de dificuldade para cada conteúdo da disciplina de algoritmos;
- RF04: É necessário a contabilização de pontuação dos jogadores em cada desafio e o total de pontos atingido até o momento;
- RF05: Os desafios devem conter um sistema de dicas e contextualização em função do conteúdo exigido;
- RF06: O professor deve ser capaz de cadastrar novos desafios ou corrigir desafios antigos;
- RF07: O professor deve ter acesso à pontuação de sua turma na plataforma.
- RF08: O professor também deve ser capaz gerenciar o cadastro dos alunos das suas turmas;
- RF09: O jogo deve conter elementos visuais e animações para prender a atenção do aluno;
- RF10: Ao final da primeira partida de um novo aluno, uma janela deve ser mostrada questionando se ele deseja que sua pontuação seja exibida no ranking geral/público.
- RF11: O sistema poderá ser controlado tanto via *mouse* quanto teclado.

- RF12: Quando o aluno acertar uma questão, o oponente deve levar dano, consequentemente perdendo vida.
- RF13: Quando o aluno errar uma questão ele sofrerá uma punição (dano ou *debuff* temporário), o oponente não sofrerá dano. Após 3 erros consecutivos, o oponente começa a regenerar vida ou recebe um *buff* de dano temporário.
- RF14: O sistema deve definir a quantidade de dano ou cura randomicamente dentro de um intervalo pré-definido.
- RF15: O sistema deve ter trilha sonora e sons de efeito (ataques, cura, *buff*, etc).
- RF16: Quando o aluno errar uma questão, a resposta correta deve ser exibida de forma destacada. Ao clicar na resposta correta, a explicação/resolução deve ser exibida, se necessário.
- RF17: O aluno poderá pausar o jogo, alterar suas configurações ou silenciar a trilha sonora.
- RF18: As perguntas devem ser selecionadas de forma randômica, sendo que elas não devem ser repetidas para um mesmo aluno.
- RF19: As opções de resposta para uma pergunta devem ser embaralhadas.
- RF20: O aluno terá um tempo definido para responder a questão dependendo da dificuldade. Caso não responda a tempo, ele é penalizado.
- RF21: Um formulário de contato para envio de sugestões, elogios e/ou reclamações deve ser disponibilizado numa das opções do menu inicial.
- RF22: O aluno poderá acessar a pontuação geral da turma pelo menu ou ao terminar uma partida.

- RF23: O sistema deverá gerar um histórico de cada aluno para o professor poder acompanhar a evolução da turma.
- RF24: O sistema deverá conter um botão “ajuda” geral que explica as regras do jogo.
- RF25: O sistema deve possuir uma interface gráfica organizada e intuitiva visando facilitar a navegação e melhorar a experiência do usuário.

b) Requisitos Não-Funcionais

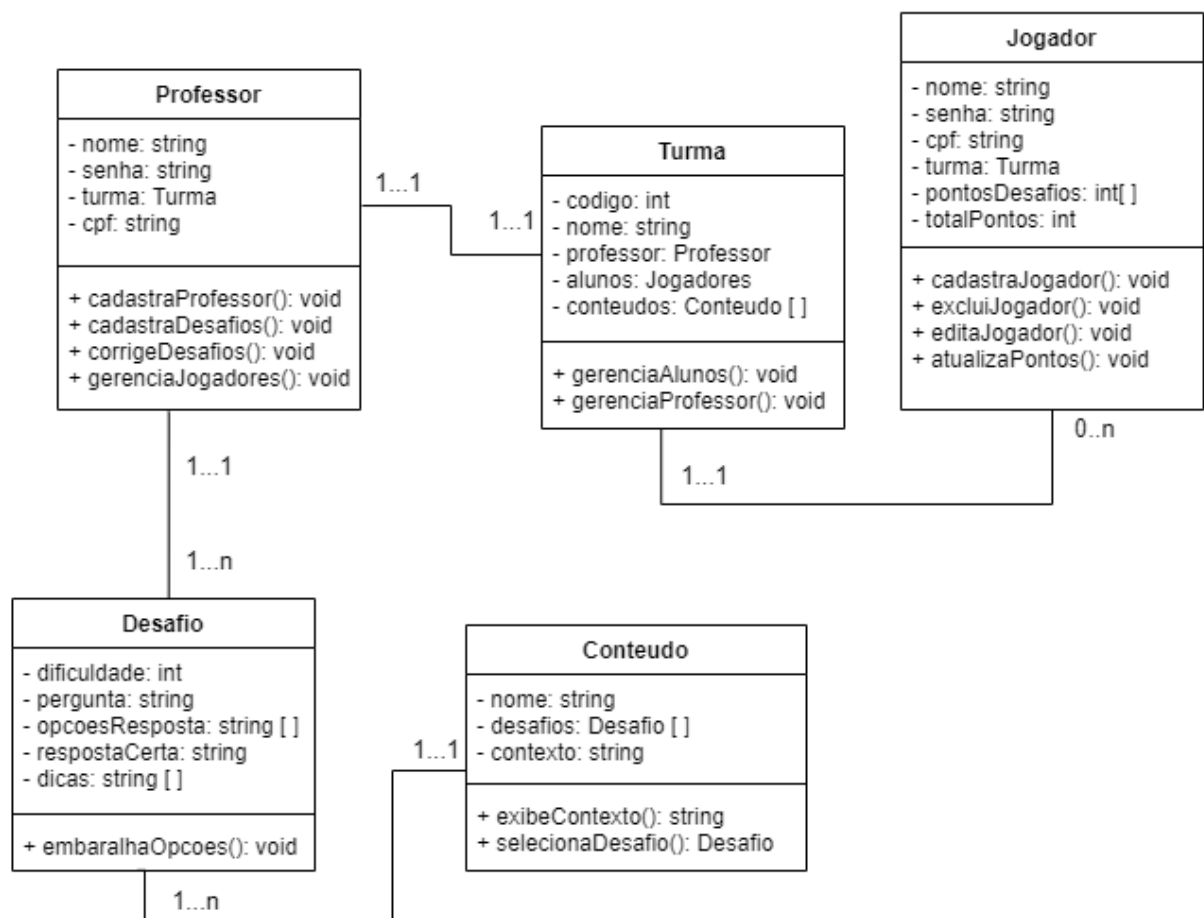
- RNF01: Segurança: controlar acesso dos jogadores e perfis de acesso (professor e aluno).
- RNF02: Segurança: Validar e garantir unicidade dos dados de *login* de cada usuário.
- RNF03: Confiabilidade: Salvar os dados dos usuários e pontuação no banco de dados.
- RNF04: Ética: O sistema não apresentará aos usuários quaisquer dados de cunho privativo (matrícula, cpf, nome completo, etc).
- RNF05: Tempo de resposta: Garantir que as respostas visuais às interações dos usuários (ataques, animações, etc) sejam realizadas em até 750 milissegundos.
- RNF06: Tempo de resposta: Garantir que as consultas das questões e respostas sejam realizadas em até 500 ms.
- RNF07: Acessibilidade: Garantir que o sistema seja intuitivo, com fontes de fácil leitura e de tamanho adequado.
- RNF08: Portabilidade: Assegurar que o sistema possa ser executado em máquinas com qualquer sistema operacional e nos navegadores atuais.

- RNF09: Manutenibilidade: Construir o sistema de forma que novas funcionalidades sejam fáceis de implementar.
- RNF10: Desempenho: Garantir que o jogo funcione a pelo menos 30 *frames* por segundo.

2. Diagrama de Classes

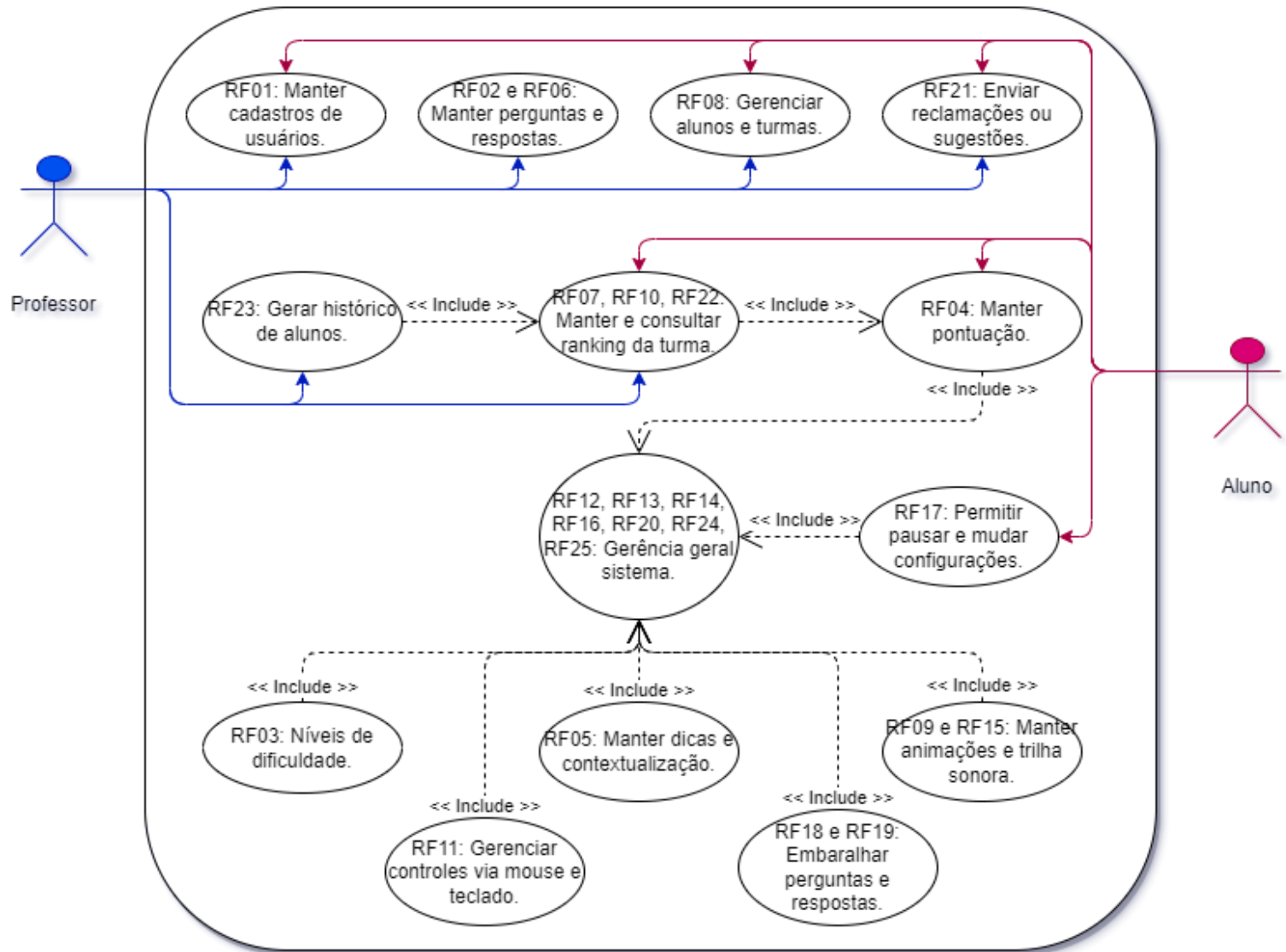
Link de edição do diagrama no *draw.io*: [Diagrama de Classes Grupo 4](#)

Coding Fight - Diagrama de Classes



3. Diagrama de Casos de Uso

Link de edição do diagrama no *draw.io*: [Diagrama de caso de uso grupo 4.drawio](#)



Link de edição deste documento: [Documento de Requisitos de Sistema](#)