

Trabalho de Desenvolvimento Web – 2024.2

1 Introdução

Esse documento tem por objetivo especificar os requisitos da aplicação Escola. O sistema não deve ser desenvolvido “do zero”, **é obrigatório utilizar o exemplo aplicacaoMVC**, explicado em sala, como aplicação base para esse trabalho. Caso o aluno não utilize a aplicação base **aplicacaoMVC** para o trabalho, **será considerado reprovado**.

2 Visão da Solução

Uma aplicação em Java (servlet+jsp) que **apoia** uma escola registrando os dados de professores, alunos, disciplinas e turmas. Apresenta de forma pública as disciplinas disponíveis para a matrícula e as respectivas vagas.

3 Stakeholders

Alunos: São aqueles que visualizam e se matriculam nas disciplinas disponíveis;

Professores: Responsáveis por ministrar uma turma de uma disciplina;

Administradores: usuários autorizados a acessar a área restrita e responsável pelo cadastro de alunos, professores, disciplinas e turmas e outros administradores, além de obter relatórios.

4 Escopo da Solução

4.1 Requisitos Não Funcionais Gerais

RNF1: Sistema deverá ser responsivo sendo obrigatório o uso do Bootstrap

RNF2: Todas as bibliotecas, scripts, imagens etc., necessários para o funcionamento devem estar disponíveis localmente (junto com a aplicação);

4.2 Requisitos Funcionais Gerais

Sempre que o termo cadastrar (e seus sinônimos) for empregado fica compreendido um conjunto de tarefas a serem implementadas na aplicação são: Listar/Consultar, incluir, alterar e excluir - CRUD.

4.2.1 Casos de Usos - Administradores

RF1: O administrador acessa a área privada da aplicação (faz login).

RF2: O administrador obtém os dados do aluno e cadastra-o no sistema (CRUD).

RF3: O administrador obtém os dados do professor e cadastra-o no sistema (CRUD)

RF4: O administrador obtém os dados de outro administrador e cadastra-o no sistema (CRUD).

RF5: O administrador obtém os dados da disciplina e cadastra-a no sistema (CRUD).

RF6: O administrador obtém os dados da turma e cadastra-a no sistema (CRUD).

RF7: O administrador gera relatório (na tela) sobre as disciplinas/turmas e os alunos matriculados em cada uma delas e as suas respectivas notas.

4.2.2 Casos de Usos - Alunos

RF8: O aluno acessa a área privada da aplicação (faz login).

RF9: O aluno obtém os dados da disciplina/turma e se inscreve (inscrição).

RF10: O aluno lista suas notas (histórico – na tela).

4.2.23 Casos de Usos - Professores

RF11: O professor acessa a área privada da aplicação (faz login).

RF12: O professor lança a nota do aluno (edita).

RF13: O professor lista as notas de todos os alunos inscritos em cada disciplina/turma sob sua responsabilidade (na tela).

5 Limites e Restrições da Solução

5.1 As tabelas do Banco de Dados apresentadas abaixo **não podem ser alteradas**; será fornecido o script para o banco de dados.

5.2 Deve existir um administrador previamente cadastrado no sistema (Banco de Dados) com o **cpf= 249.252.810-38 e senha= 111**

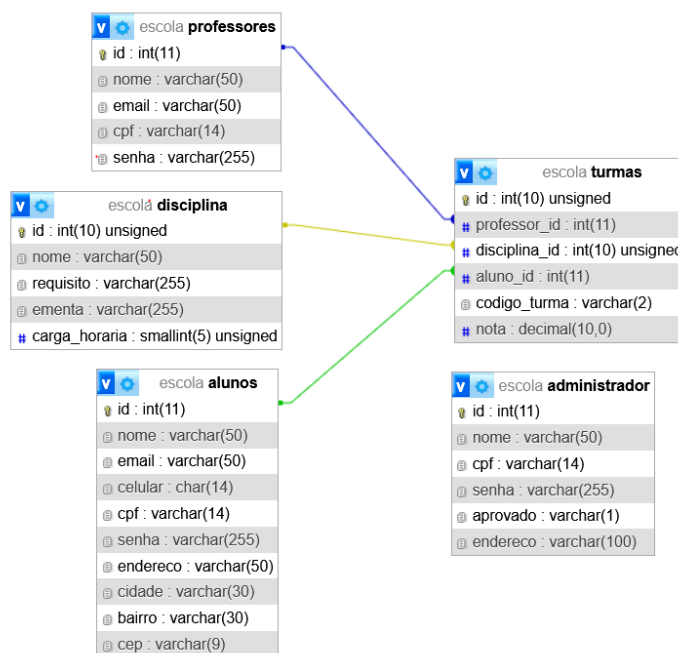
5.3 Todos os logins são realizados usando cpf e senha.

5.4 Deverá possuir validações no servidor (campos vazios, valores negativos etc.).

5.5 Deverá possuir validações no cliente (campos vazios, valores negativos etc.).

5.6 O sistema deverá ser construído a partir da aplicação base fornecida.

6 Diagrama



7 Regras de Negócio

Alunos:

RN1. Um aluno pode se inscrever em qualquer disciplina/turma.

RN2. Um aluno não pode apagar sua inscrição após o lançamento da nota pelo professor.

RN3. Um aluno só pode se inscrever em uma disciplina/turma se houver vaga. Considere o número máximo de vagas = 2.

Professor:

RN4. Um professor pode ministrar apenas duas (2) disciplina/turma.

Administrador:

RN5. Um administrador só fará login caso esteja aprovado.

8 Cronograma

Data Limite Primeira Entrega: 20/12/2024

Requisitos: RF1 até RF7

Data Limite Segunda Entrega: 23/01/2025

Requisitos: RF8 até RF13

A falta de postagem do trabalho nas datas será atribuído zero