**Sistema de Gestão Acadêmica**

**Linguagem de Programação IV – Internet**

**Projeto MySQL/PHP**

**Integrantes**

JOÃO PEDRO DA SILVA

RENAN MARTARELLI

ANA CAROLINA SANTOS

GABRIELA SOUZA

**Colaboração de cada parte**

JOÃO PEDRO DA SILVA – 25%

RENAN MARTARELLI – 25%

ANA CAROLINA SANTOS – 25%

GABRIELA SOUZA – 25%

**Objetivo**

O sistema proposto nesse documento, busca trazer facilidade e agilidade na gestão acadêmica, imagine uma situação em que você coordena uma instituição de ensino, este sistema trará a você a possibilidade de gerir seus alunos, professores, disciplinas, turmas.

**Escopo**

O sistema busca levar praticidade aos usuários finais, trazendo agilidade em tarefas corriqueiras do dia a dia de alguém inserido no contexto acadêmico.

Por meio de um computador ou celular, o sistema deve possibilitar que três níveis de usuários acessem a plataforma, usuário aluno, usuário professor e usuário gestor ou administrador, uma vez logado na plataforma os usuários terão acesso a ferramentas cabíveis ao seu nível de acesso, ferramentas essas que dizem respeito a sua posição, um aluno possuirá apenas ferramentas de consulta de dados próprios, um professor já pode consultar e alterar dados de alunos, e um gestor tem permissões mais avançadas que o permitira gerir todos os aspectos da instituição.

Devido aos diferentes tipos de acesso, as telas disponíveis ao usuário final podem variar um pouco, já que, dentro do sistema, os diferentes tipos de usuários possuem diferentes tipos de permissão.

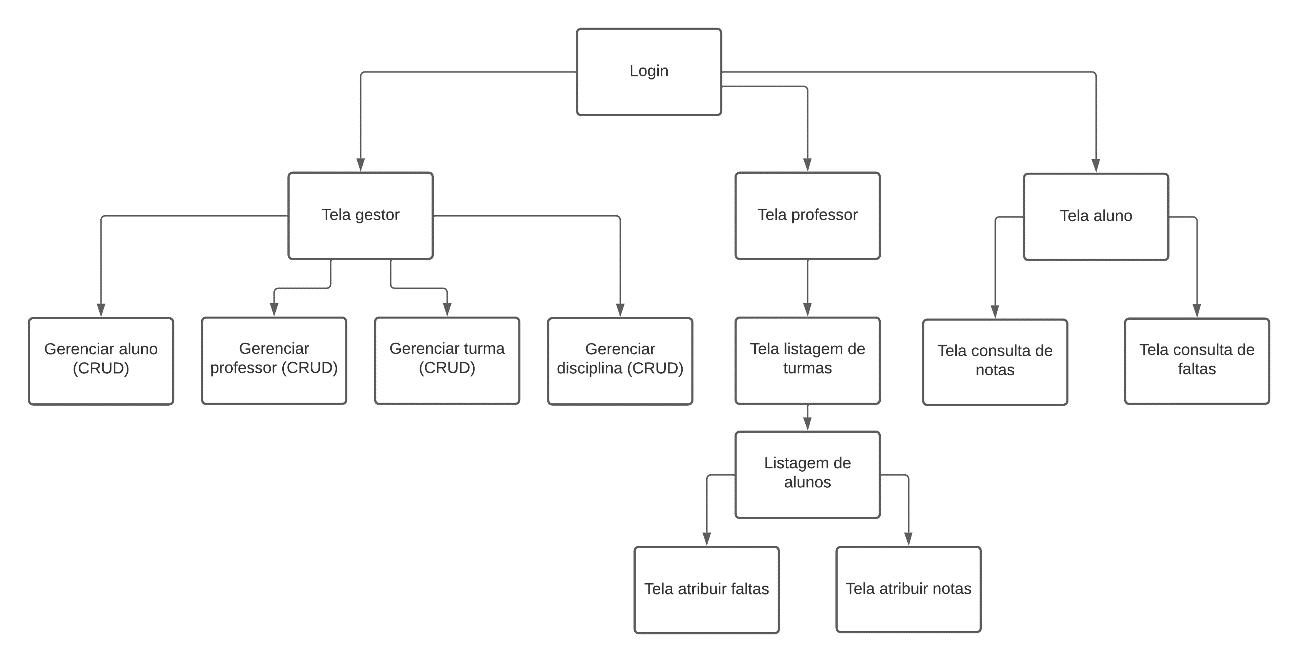
**Regras de negócio**

* Um aluno pode consultar informações sobre si mesmo, suas notas e faltas.
* Todo aluno possui 3 notas no sistema, P1, P2 e P3.
* Todo aluno possui um status, aprovado, reprovado ou exame; após cadastrar 2 notas (P1 e P2) o sistema calculará a média do aluno, se ela for superior a 6 o aluno recebe o status de aprovado, se for inferior a 6 o aluno recebe o status de exame; após ser atribuída a nota P3 o sistema irá verificar se a nota P3 é maior que 6, se sim o aluno recebe o status de aprovado, se não o aluno recebe o status de reprovado.
* Todo aluno possui faltas, se suas faltas ultrapassarem um limite de 25% o aluno recebe o status de reprovado.
* Um professor pode cadastrar notas e faltas aos alunos.
* Um professor pode poder visualizar as turmas em que ministra aulas.
* Um professor pode selecionar uma turma em que ministra aulas e visualizar os alunos daquela turma.
* Um professor pode alterar os valores das notas de um aluno específico de uma de suas turmas.
* Um professor pode atribuir faltas a um aluno específico de uma de suas turmas.
* Um gestor pode criar uma disciplina ou alterar disciplinas já existentes.
* Um gestor pode atribuir um professor a uma determinada disciplina.
* Um gestor pode criar uma turma e atribuir disciplinas a essa turma.
* Um gestor pode cadastrar professores e alunos.
* Um gestor pode atribuir alunos a uma determinada disciplina.

**Requisitos**

* O sistema deve permitir ao aluno fazer login e consultar suas notas e faltas.
* O sistema deve permitir ao professor fazer login e atribuir notas ou faltas aos alunos de suas turmas.
* O sistema deve permitir que o gestor ou administrador faça login, crie disciplinas, crie turmas, atribuas alunos para as turmas, atribua disciplinas para as turmas, cadastre alunos e cadastre professores.

**Mapa navegacional das telas**



**Informações relevantes que serão fornecidas, tratadas e armazenadas**

Entidades:

* **Aluno**: Código (Gerado automaticamente), Cpf, Nome, Data Nascimento, Turma.
* **Professor**: Código (Gerado automaticamente), Cpf, Nome, Data Nascimento, Disciplina, Turmas.
* **Turma**: Código (Gerado automaticamente), Período, Disciplinas.
* **Disciplina**: Código, Nome, Descrição.