



Prof. Jessen Vidal

TRABALHO CORRESPONDENTE À 70% DA NOTA DA DISCIPLINA

LABORATÓRIO DE ENGENHARIA DE SOFTWARE

PROF. FABRÍCIO GALENDE MARQUES DE CARVALHO

Desenvolver um sistema web que atende aos seguintes requisitos e restrições:

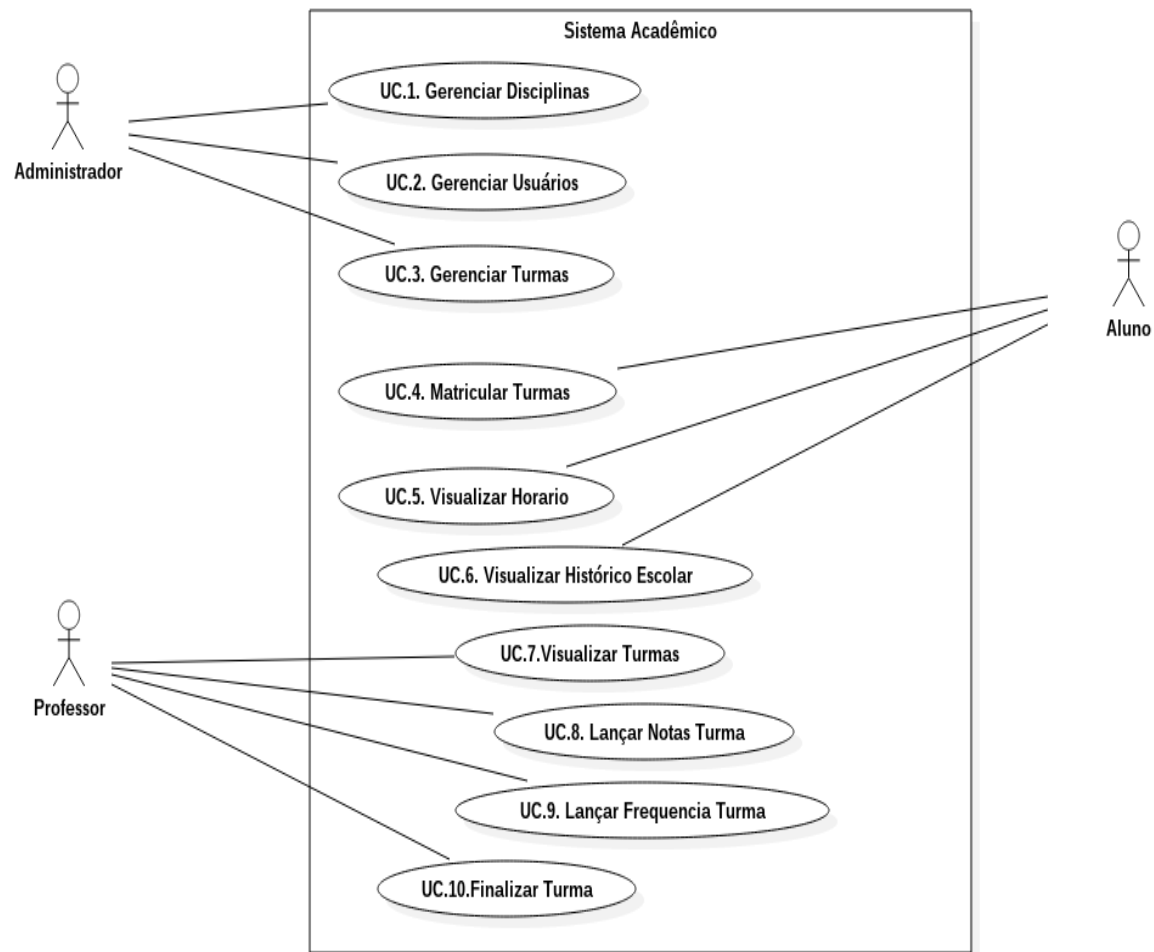
**REQUISITOS E RESTRIÇÕES GERAIS:**

1. Utilizar a linguagem Java (Versão 8 ou superior) no back end;
2. Utilizar DAO ou JPA com provedor Hibernate (5 ou superior) para persistência de dados;
3. Utilizar o padrão de projeto MVC na arquitetura do sistema;
4. Utilizar Servlets anotados ou com descritor de implantação;
5. Utilizar Java Server Pages para criar páginas dinâmicas (tipo templates);
6. Utilizar JavaScript (JS) no front end;
7. Utilizar JS Closures;
8. Utilizar JS DOM Handlers;
9. Utilizar JS event listeners;
10. Utilizar JS para criar dinamicamente elementos no DOM.
11. Utilizar métodos da JQuery;
12. Utilizar marcação HTML 5 com sintaxe e semântica adequados;
13. Utilizar CSS 3 no front end;
14. Utilizar CSS 3 com fonte externa nas interfaces;
15. Utilizar CSS 3 media queries;
16. Utilizar CSS 3 para deixar o sistema “minimamente responsivo”;

17. Utilizar as bases de dados MySQL ou MariaDB para persistência de dados;
18. Utilizar o Apache TomCat como web container;

## REQUISITOS E RESTRIÇÕES ESPECÍFICOS

**Sistema 1: Sistema acadêmico (pode ser feito individualmente, em duplas ou trios);**



**Detalhamento aplicável:**

**Requisitos de dados (mínimos além dos que são evidentes pelo diagrama):**

Disciplina: Deve conter nome e carga horária (variando de 2 a 4h por semana);

Turma: Deve ser específica de disciplina, deve conter horário (dia da semana, hora de início e hora de término);

**Comportamento esperado das funcionalidades (mínimos além dos que são evidentes pelo diagrama):**

UC.1 - UC.3 - Por gerenciar, subentende-se ser capaz de efetuar o CRUD correspondente à todas as funcionalidades.

UC.4. Um aluno não deve ser capaz de se matricular em turmas caso sua carga horária semanal seja superior a 24h. Não deve ser permitido também que um aluno se matricule em uma turma caso já tenha cursado a disciplina.

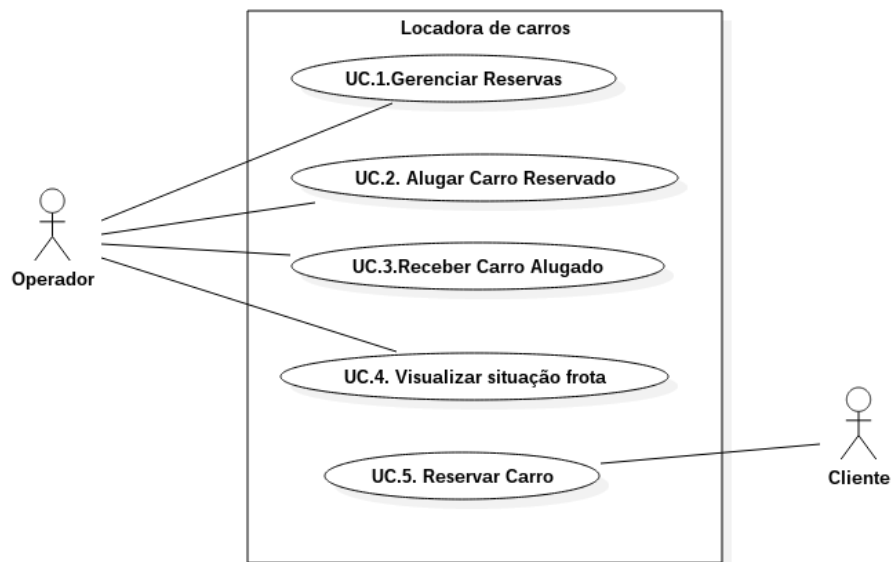
UC.6. O Histórico escolar deve mostrar a diferenciação entre disciplinas cursadas ou em curso. No caso das disciplinas cursadas (turmas fechadas) deve aparecer a nota final do aluno e sua frequência.

UC.7. O professor deve poder visualizar todas as turmas às quais está alocado;

UC.8-UC.9 - Lançamento somente da média e da frequência final.

UC.10 - Encerrar a disciplina faz com que ela apareça como cursada pelo aluno no histórico.

## Sistema 2: Sistema de locadora de veículos (individual ou em duplas);



### Detalhamento aplicável:

#### Requisitos de dados (mínimos além dos que são evidentes pelo diagrama):

Carro: categoria, ano, placa, modelo, fabricante, cor, estado de conservação, quilometragem, tanque de combustível (preenchimento);

Reserva: Código de reserva, dados do usuário, hora início, término previsto, valor total previsto, categoria de carro ou carro específico.

Aluguel de carro (contrato): Dados do cliente, dados do carro, hora de retirada, hora de devolução, valor total da reserva, valor pago antecipadamente, valor pago na devolução.

**Comportamento esperado das funcionalidades (mínimos além dos que são evidentes pelo diagrama):**

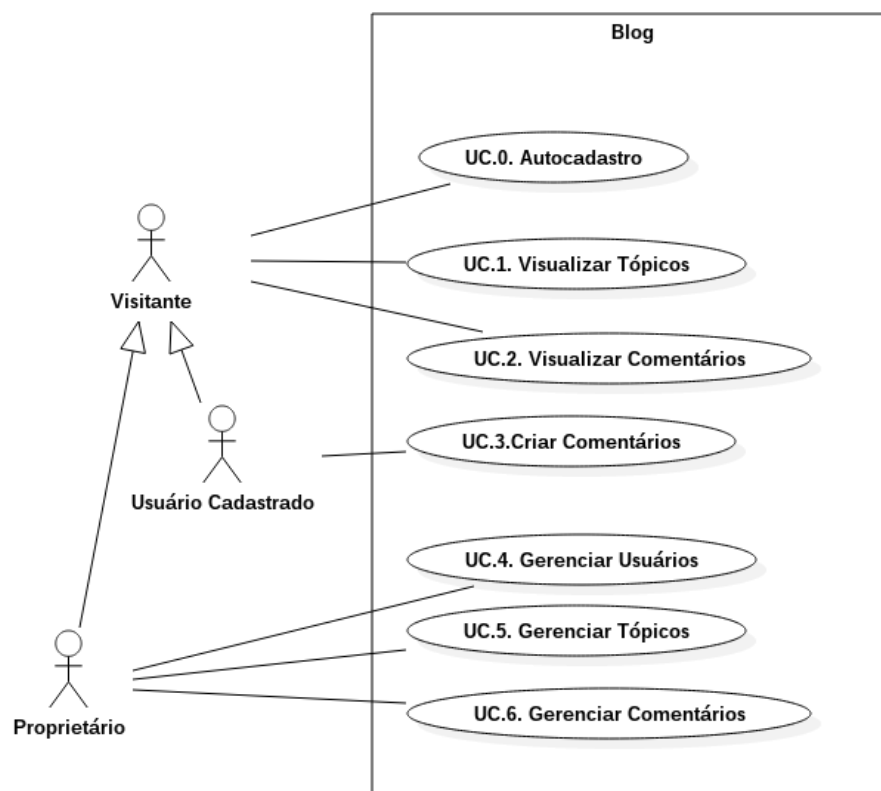
UC.1. CRUD das reservas.

UC.2. Criar e abrir um contrato de locação (deve exigir um valor pago antecipadamente igual ao valor previsto para a reserva);

UC.3. Deve fechar o contrato de locação, inserindo eventuais valores devidos e atualizando os dados do carro no momento da devolução (ex. Estado de conservação, quilometragem, tanque, etc.)

UC.4. Visualizar quais carros estão alugados e quais estão disponíveis.

**Sistema 3: Blog (individual)**



**Detalhamento aplicável:**

**Requisitos de dados (mínimos além dos que são evidentes pelo diagrama):**

Usuário: Nome, email, senha e apelido (deve ser único);

Tópico: Título, texto de conteúdo e data de criação;

Comentário: Autor, texto de conteúdo e data de criação;

**Comportamento esperado das funcionalidades (mínimos além dos que são evidentes pelo diagrama):**

UC.0. Qualquer pessoa pode se autocadastrar;

UC.1 - UC.2. Os tópicos mais recentes devem ser exibidos antes na página de listagem (topo);

UC.3 - Comentários contendo palavras ofensivas não devem ser permitidos no blog.

UC. 4. Um usuário excluído que tenha comentários vinculados à algum tópico, deve ter todos os seus comentários excluídos.

UC.5. Um tópico que tenha sido comentado por um usuário não poderá ser alterado, somente excluído.

UC.6. Um comentário específico pode ser excluído.

**Sistema 4: Tema de TG (individual).**

**Nesse caso, o aluno deve entregar o termo de formalização de orientação (cópia) e a ciência do professor da disciplina TG 1. Além disso, deve fazer uso de todas as tecnologias citadas como requisitos gerais e deve apresentar um diagrama de caso de uso com as funcionalidades do sistema, sendo obrigatória a implementação de pelo menos duas funcionalidades que não sejam do tipo “cadastro de usuários”.**

**ENTREGAS:**

Devem ser via link (por e-mail) de repositório no github para o código-fonte e devem conter todas as bibliotecas e dependências aplicáveis ao projeto. Podem ser também via arquivo em formato .zip, compartilhado em algum drive e enviado o link por e-mail.

1a Entrega – Semana seguinte ao início da primeira semana de provas: Front end contendo HTML com navegação completa entre as páginas (protótipos com conteúdo estático). Lógica de login (com acesso a banco de dados) e visualização das páginas completamente implementadas.

2a Entrega– Semana seguinte ao início da segunda semana de provas. Sistema completo, no mesmo formato da primeira entrega (link para repositórios ou drivers).

**POLÍTICA DE AVALIAÇÃO:**

BASEDA NO NÚMERO DE REQUISITOS GERAIS E ESPECÍFICOS ATENDIDOS (TODOS TEM O MESMO PESO)  
E NA COMPREENSÃO DO ALUNO SOBRE O SISTEMA ENTREGUE (PERGUNTAS E TESTES EFETUADOS PELO  
PROFESSOR DURANTE A APRESENTAÇÃO)