

# Acadêmico - Sistema de Matrícula de Alunos

## Disciplina: Testes de Software

**Prof. Willy Tiengo**

### Objetivo

Este exercício tem o objetivo de consolidar o conhecimento teórico visto durante as aulas através da sua aplicação na construção de um sistema.

### Visão geral do sistema

O sistema DRCA consiste do gerenciamento das matrículas de alunos de uma universidade. Este processo é feito pela Departamento de Registro e Controle Acadêmico (DRCA), considerando os pré-requisitos de cada uma das disciplinas oferecidas pelos diversos departamentos da universidade.

A universidade é formada pela DRCA e por vários departamentos que através de suas secretarias oferecem cursos de graduação e pós-graduação. Cada departamento possui diversos professores que lecionam suas disciplinas.

Os departamentos podem ter até duas secretarias que possuem responsabilidades diferentes: secretaria de pós-graduação e secretaria de graduação. Se o departamento oferecer curso de graduação este deve ter a secretaria de graduação e o mesmo ocorre para pós-graduação.

Todo curso tem uma lista de disciplinas, sendo estas obrigatórias e eletivas, que podem ou não estar sendo oferecidas em um determinado período. Assim como os cursos, existem disciplinas de pós-graduação e disciplinas de graduação. Além disso cada disciplina possui:

- um nome e um código;
- um número de créditos associados;
- um pré-requisito correspondente ao número mínimo de créditos que o aluno precisa ter cumprido para que a matrícula deste seja aceita. Existem disciplinas que não possuem este pré-requisito, isto é, o número de créditos necessários para a matrícula do aluno pode ser 0.
- um pré-requisito correspondente ao conjunto de disciplinas que o aluno precisa já ter cursado para que ele tenha o seu pedido de matrícula aceito. Existem disciplinas que não possuem este pré-requisito, isto é, não dependem que o aluno tenha cursado nenhuma outra disciplina ao longo do curso.

### Requisitos Funcionais

1. O sistema deve permitir o processamento da matrícula dos alunos. Na matrícula deve ser oferecida uma listagem com o nome de todos os alunos da universidade. Após escolhido o aluno, devem ser mostradas as disciplinas do departamento do curso do aluno. Após a seleção da disciplina, deve-se informar se o aluno foi ou não matriculado, justificando no caso negativo. A matrícula somente deve ser concretizada se o aluno cumpriu os pré-requisitos (número de créditos mínimo e disciplinas) da disciplina pretendida.
2. Os alunos só podem se matricular em disciplinas do departamento ao qual seu curso pertence e os alunos de pós-graduação não podem cursar disciplinas da graduação. Porém os alunos de graduação podem cursar disciplinas de pós-graduação caso já

tenham cumprido pelo menos 170 créditos. Não existe o conceito de turma. Sendo assim, não existe número de vagas limitado para matrícula em disciplina. Os alunos não podem se matricular em disciplinas que já tenham se matriculado ou cursado.

3. O sistema deve fornecer as seguintes consultas:
  - a. Uma lista por secretaria (graduação e pós-graduação) com os códigos, números de créditos, os códigos dos pré-requisitos, os números de créditos mínimos e os nomes das disciplinas que estão sendo oferecidas neste período por cada departamento.
  - b. Dada uma disciplina, deseja-se uma pauta da mesma, ou seja, uma lista contendo o código, número de créditos, os códigos dos pré-requisitos, o número de créditos mínimo e o nome da disciplina. Além disso devem ser apresentados, o nome do professor responsável e a lista de alunos matriculados na disciplina, com os seus nomes e números de matrícula.
  - c. Dado um aluno, deseja-se um comprovante de matrícula, ou seja, uma lista com o seu nome e número de matrícula, e com os códigos e nomes das disciplinas nas quais o aluno está matriculado.

### Observações

1. O sistema deve ser implementado em Java, utilizando pelo menos as seguintes tecnologias:
  - a. Dropwizard - <https://www.dropwizard.io/>
  - b. Lombok - <https://projectlombok.org/>
  - c. Hibernate – <http://hibernate.org/>
  - d. Jersey - <https://jersey.github.io/>
2. Você deve se basear no projeto fornecido como exemplo.
3. O sistema deve persistir os dados em banco de dados.
4. O projeto deve ser testado com pelo menos o JUnit 5 (pode utilizar outras ferramentas de teste, como, por exemplo, o Mockito - <https://site.mockito.org/> ou Hamcrest - <http://hamcrest.org/JavaHamcrest/>)
5. Utilize assertivas de execução.
6. Crie um plano de teste. Ele deve prever:
  - a. Construção de testes automatizados unitários e funcionais.
  - b. Construção de testes aleatórios.
  - c. Construção de testes automatizados da interface web.
7. Execute o plano de teste projetado. Ao final, meça a cobertura dos testes.

### Créditos

Este exercício foi adaptado da disciplina INF 2125, da PUC-Rio, pelo Prof. Rodrigo Paes.