

UNIVERSIDADE ESTADUAL PAULISTA

“JÚLIO DE MESQUITA FILHO”

Instituto de Geociências e Ciências Exatas - IGCE

Curso de Bacharelado em Ciências da Computação

GABRIEL LUIZ

LEO EDUARDO

GUILHERME SIMIONATTO

**SISTEMA DE MANUTENÇÃO DE  
MENSALIDADES DE ACADEMIA COM  
DIVERSAS MODALIDADES**

Orientadora: Profa. Dra. Simone das Graças Domingues Prado

Rio Claro - SP

2019

# **SISTEMA DE MANUTENÇÃO DE MENSALIDADES DE ACADEMIA COM DIVERSAS MODALIDADES**

Relatório de Linguagens Comerciais de Programação, realizado de janeiro à julho de 2019, visando desenvolver uma plataforma java pelo Curso de Bacharelado em Ciências da Computação do Instituto de Geociências e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Câmpus de Rio Claro.

Alunos: Gabriel Luiz  
Guilherme Simionato  
Leo Eduardo

Orientadora: Profa. Dra. Simone das Graças Domingues Prado

Rio Claro - SP

2019

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>3</b>
<b>2</b>	<b>DESENVOLVIMENTO</b>	<b>4</b>
<b>2.1</b>	<b>Telas</b>	<b>4</b>
2.1.1	Tela de Pesquisa	4
2.1.2	Tela de Cadastro	7
<b>2.2</b>	<b>Back-End</b>	<b>9</b>
<b>3</b>	<b>CONCLUSÃO</b>	<b>11</b>
	<b>REFERÊNCIAS</b>	<b>12</b>

# 1 Introdução

O objetivo deste projeto foi realizar um sistema de academia visando o controle de pagamentos. Para isso, o usuário tem a possibilidade de efetuar cadastro dos alunos, modalidades, valores, data de pagamento entre outras informações. Uma vez que os dados estejam cadastrados, o usuário pode procurar a situação (paga/pendente) dos alunos e realizar o controle e baixa no sistema.

Sendo assim, o Capítulo 2 aborda o processo de desenvolvimento do software. Este, por sua vez, é subdividido: Capítulo 2.1 referente às telas e o Capítulo 2.2 referente às ferramentas usadas e particularidades do back-end.

Por fim, o Capítulo 3 destaca as considerações finais a respeito do projeto que engloba e utiliza os conceitos vistos durante as aulas de Linguagens Comerciais de Programação lecionadas no primeiro semestre de dois mil e dezenove.

## 2 Desenvolvimento

Usaremos as tecnologias de Java e Swing (código e telas), MongoDB (estruturar o banco de dados), Spring (comunicação com o banco de dados) e GIT (gerenciamento de versão) para criação deste sistema. Além disso, o atual escopo não prevê um controle financeiro (retorno de troco, quantidade recebida ou ainda fechamento de contas a pagar).

Outras informações pertinentes ao cadastro antes previstas como endereço, contato, desconto nas mensalidades entre outras também foram descartadas nesta versão e podem, futuramente, ser implementadas.

### 2.1 Telas

A aplicação conta com duas telas onde o usuário tem a liberdade de pesquisar e cadastrar novos usuários, conforme vemos a seguir.

#### 2.1.1 Tela de Pesquisa

Assim que a aplicação é inicializada podemos observar (Figura 1) os filtros para a pesquisa de mensalidades dos alunos, um botão para pagamento e outro para efetuar os cadastros.

Gym Application

Cliente:

Modalidade:

Data Inicial:

Data Final:

Status:

☐ Pago

Pesquisar

ID	Cliente	Modali...	Data V...	Valor	Pago
----	---------	-----------	-----------	-------	------

ID:

Pagar

Cadastrar

Figura 1 – Tela de Pesquisa vazia  
Fonte: pessoal.

No exemplo (Figura 2) é feita uma busca utilizando todos os filtros que retorna todas as mensalidades não pagas do cliente Léo vinculadas à modalidade musculação.

Gym Application

Cliente:  Modalidade:

Data Inicial:  Data Final:

Status: ☐ Pago

ID	Cliente	Modali...	Data V...	Valor	Pago
5d...	Leo	musc...	15/08...	100.0	Pende...
5d...	Leo	musc...	15/09...	100.0	Pende...
5d...	Leo	musc...	15/10...	100.0	Pende...
5d...	Leo	musc...	15/11...	100.0	Pende...
5d...	Leo	musc...	15/12...	100.0	Pende...

ID:

Figura 2 – Exemplo de pesquisa na Tela de Pesquisa  
Fonte: pessoal.

Ainda é possível verificar na figura que houve o pagamento das parcelas referente aos meses 06 e 07.

Evidentemente, uma nova pesquisa com o filtro pago aplicado resulta na exibição das duas parcelas pagas (Figura 3).

Gym Application

Cliente:  Modalidade:

Data Inicial:  Data Final:

Status: ☒ Pago

ID	Cliente	Modali...	Data Ve...	Valor	Pago
5d...	Leo	musc...	15/06...	100.0	Pago
5d...	Leo	musc...	15/07...	100.0	Pago

ID:

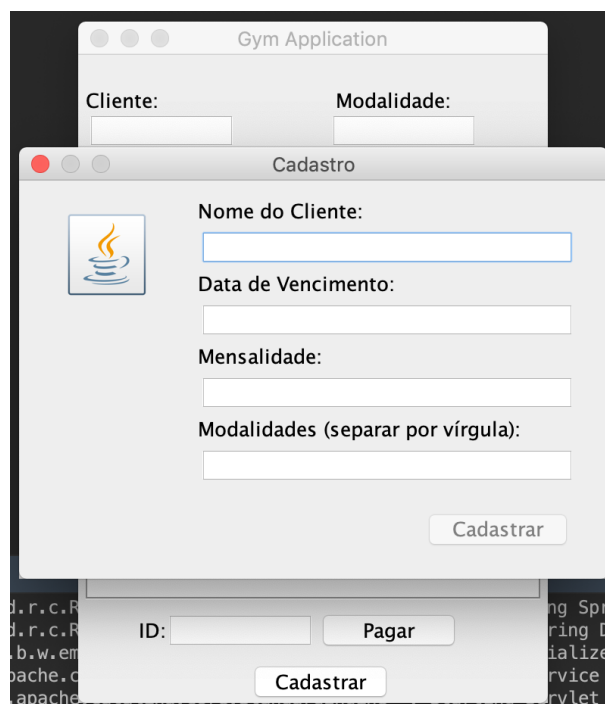
Figura 3 – Exemplo de pesquisa na Tela de Pesquisa

Fonte: pessoal.

### 2.1.2 Tela de Cadastro

A Tela de Pesquisa introduz o acesso ao botão Cadastrar que possibilita o usuário inserir novos alunos, vinculando as modalidades, valores e a data de vencimento da parcela (Figura 4).



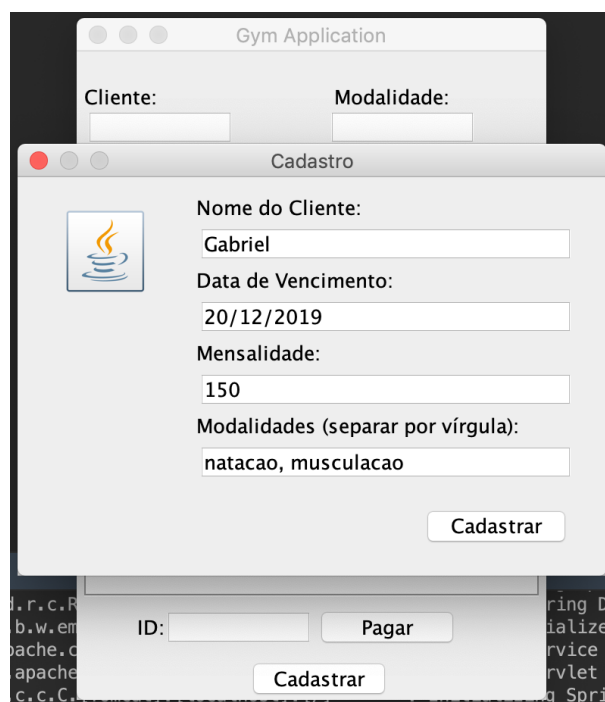


The screenshot shows a window titled "Gym Application" with two sub-windows. The top sub-window has fields for "Cliente:" and "Modalidade:". The bottom sub-window, titled "Cadastro", contains a Java logo on the left and the following fields on the right: "Nome do Cliente:", "Data de Vencimento:", "Mensalidade:", and "Modalidades (separar por vírgula):". Each field has an empty text input box. At the bottom right of the "Cadastro" window is a button labeled "Cadastrar". Below the "Cadastro" window, another sub-window is partially visible with fields for "ID:" and "Pagar", and a "Cadastrar" button at the bottom.

Figura 4 – Tela de Cadastro vazia  
Fonte: pessoal.

Como podemos observar, os quatro campos devem ser preenchidos para que o cadastro seja efetuado.

Uma vez que o preenchimento obrigatório ocorreu, é possível efetuar o cadastro (Figura 5) que poderá ser visualizado com os filtros adequados na Tela de Pesquisa.



This screenshot shows the same "Gym Application" window as Figure 4, but the "Cadastro" sub-window now contains example data. The "Nome do Cliente:" field is filled with "Gabriel", "Data de Vencimento:" with "20/12/2019", "Mensalidade:" with "150", and "Modalidades (separar por vírgula):" with "natacao, musculacao". The "Cadastrar" button remains at the bottom right. The bottom sub-window with "ID:" and "Pagar" fields is also visible.

Figura 5 – Exemplo de cadastro na Tela de Cadastro  
Fonte: pessoal.

Evidentemente, a busca com os filtros adequados resultam no preenchimento da tabela com os dados cadastrados (Figura 6) e possibilita o usuário a realizar as baixas necessárias.

The screenshot shows a window titled "Gym Application". It contains a search form with the following fields and controls:

- Cliente:** Text input with "Gabriel" entered.
- Modalidade:** Empty text input.
- Data Inicial:** Date input with "01/01/2019" entered.
- Data Final:** Date input with "31/12/2019" entered.
- Status:** A checkbox labeled "Pago" is unchecked.
- Pesquisar:** A button to execute the search.

Below the form is a table with the following data:

ID	Cliente	Modali...	Data V...	Valor	Pago
5d...	Gabriel	natac...	20/06...	150.0	Pende...
5d...	Gabriel	natac...	20/07...	150.0	Pende...
5d...	Gabriel	natac...	20/08...	150.0	Pende...
5d...	Gabriel	natac...	20/09...	150.0	Pende...
5d...	Gabriel	natac...	20/10...	150.0	Pende...
5d...	Gabriel	natac...	20/11...	150.0	Pende...
5d...	Gabriel	natac...	20/12...	150.0	Pende...

At the bottom of the window, there are two more controls:

- ID:** Text input.
- Pagar:** Button.
- Cadastrar:** Button.

Figura 6 – Exemplo de pesquisa na Tela de Cadastro  
Fonte: pessoal.

## 2.2 Back-End

A construção das tabelas se dá conforme segue:

Cadastro			
Long idCadastro	String nomeCliente	String modalidades	Date diaVencimento

Figura 7 – Tabela cadastro  
Fonte: pessoal.

Pagamento			
Long idPagamento	Long idCadastro	Long diaVencimento	Boolean pago

Figura 8 – Tabela Pagamento  
Fonte: pessoal.

## 3 Conclusão

Conforme exposto na seção anterior, utilizou-se dos conhecimentos lecionados pela professora Simone das Graças Domingues Prado durante o primeiro semestre de dois mil e dezenove para realização da aplicação que visa permitir a consulta de pagamentos pendentes e controle de matrícula dos alunos de uma academia. Além disso, para o desenvolvimento foi necessário o conhecimento e aprendizado de outras tecnologias não abordadas em aula, mas que complementam o conteúdo como Spring, MongoDB e GIT.

Os realizadores deste projeto acreditam ter aplicado os conhecimentos passados em aula e realizado um projeto pertinente ao proposto.

## Referências