



# Grupo Caramelo: Sistema de gerenciamento de academia

*Ana Clara Zoppi Serpa      Bruno de Marco Apolonio*  
*Gabriel Oliveira dos Santos      Lucas Costa de Oliveira*  
*Vítor Mosso Dario*

Technical Report - IC-18-01 - Relatório Técnico  
June - 2018 - Junho

UNIVERSIDADE ESTADUAL DE CAMPINAS  
INSTITUTO DE COMPUTAÇÃO

The contents of this report are the sole responsibility of the authors.  
O conteúdo deste relatório é de única responsabilidade dos autores.

# Caramelo: Sistema de gerenciamento de academia

Ana Clara Zoppi Serpa - RA 165880      Bruno de Marco Apolonio - RA 195036  
Gabriel Oliveira dos Santos - RA 197460      Lucas Costa de Oliveira - RA 182410  
Vítor Mosso Dario - RA 207024

## Resumo

Quisque accumsan ipsum id tortor laoreet feugiat. In accumsan id risus quis rutrum. Aliquam risus nunc, lacinia ac tincidunt at, accumsan ut purus. Morbi vestibulum et lacus at interdum. Cras pellentesque consectetur sapien, ac lobortis leo aliquam at. In consectetur nibh at bibendum laoreet. Aenean molestie lorem id mattis mattis. Integer rhoncus sem dictum, mattis massa vitae, pretium justo. Curabitur euismod dolor non neque semper tincidunt.

In congue consectetur risus eu pharetra. Vestibulum at lacus auctor, cursus leo et, placerat nibh. Suspendisse vitae enim justo. Praesent feugiat dolor accumsan dui euismod blandit. Maecenas interdum velit dolor, in laoreet urna feugiat finibus. Ut cursus eros id velit laoreet, at fringilla nibh ullamcorper. Donec aliquet elit nisi, quis pellentesque nibh molestie quis. Praesent volutpat hendrerit augue id molestie. Nullam orci ex, auctor non tristique vitae, dignissim at mauris. Fusce tempor eleifend aliquet. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Proin mollis ligula id laoreet dignissim.

## Sumário

1	Etapa 0: Proposta de Software	3
2	Etapa 1: Classes do sistema	4
3	Etapa 2: Release 0	4
4	Etapa 3: Release 1	4
5	Etapa 4: Release 2	4
6	Release final	4

# 1 Etapa 0: Proposta de Software

Nessa etapa, decidimos qual seria o tema do nosso projeto e quais suas funcionalidades. Optamos por implementar um sistema de gerenciamento de academias com cadastro de clientes, atividades e instrutores e a associação destes, por exemplo: o cliente Gabriel faz pilates de segunda e quarta, das 16h às 17h, e o instrutor responsável pelas aulas de pilates desse horário é o Lucas.

Escrevemos um documento descrevendo as funcionalidades que prevíamos para o sistema e entregamos para o professor. Como elas já foram descritas no documento, aqui são mencionadas mais brevemente:

- Cadastro dos clientes que frequentam a academia
- Alteração dos dados de um cliente já cadastrado
- Desativar um cliente
- Cadastro de atividades que a academia oferece
- Alteração dos dados da atividade, por exemplo, preço
- Remoção de uma atividade
- Associar/desassociar horários às atividades oferecidas
- Associar/desassociar clientes aos horários e atividades
- Cadastrar instrutores que trabalham na academia
- Alterar dados de um instrutor cadastrado
- Desativar um instrutor
- Consultar qual atividade possui mais clientes
- Consultar qual atividade possui menos clientes
- Consultar qual atividade tem o maior preço
- Consultar qual atividade tem o menor preço

Em vez de remover clientes e instrutores, optamos por desativá-los. Tivemos essa ideia pensando no seguinte cenário: um cliente decide deixar de frequentar a academia, mas, após algum tempo, retorna. Seria prático se já tivéssemos os dados dele e bastasse reativá-lo. O mesmo ocorre com instrutores que parem de trabalhar na academia, mas depois retornem. Ou, por exemplo, a academia talvez tenha interesse em tentar contactar um instrutor para contratá-lo novamente, ou um cliente antigo para perguntar se ele não deseja retornar.

Nessa etapa do projeto, as coisas ainda não estavam muito bem definidas quanto à implementação e às classes: tínhamos apenas a ideia e a lista de funcionalidades.

## 2 Etapa 1: Classes do sistema

Nessa etapa, pensamos sobre as classes e relacionamentos entre elas que precisaríamos para implementar as funcionalidades consideradas na Etapa 0 e construímos um diagrama UML.

Segue a lista de classes consideradas e o papel que pensamos para elas nessa etapa do projeto (posteriormente tivemos várias mudanças).

Pessoa ha

- Cliente ha ha  
ha
- Instrutor
- Usuario
- Administrador
- Academia
- Turma
- Atividade
- Principal

## 3 Etapa 2: Release 0

as classes e alguns métodos, esqueleto ainda

## 4 Etapa 3: Release 1

estava faltando herança, conceitos de POO e buscas

## 5 Etapa 4: Release 2

colocamos conceitos de POO e buscas

## 6 Release final

colocamos o login que tínhamos tirado