

# BeFIT

Decézaris Augusto (damp)

Ivson de Assis (iac)

Gabriel Augusto (gafb)

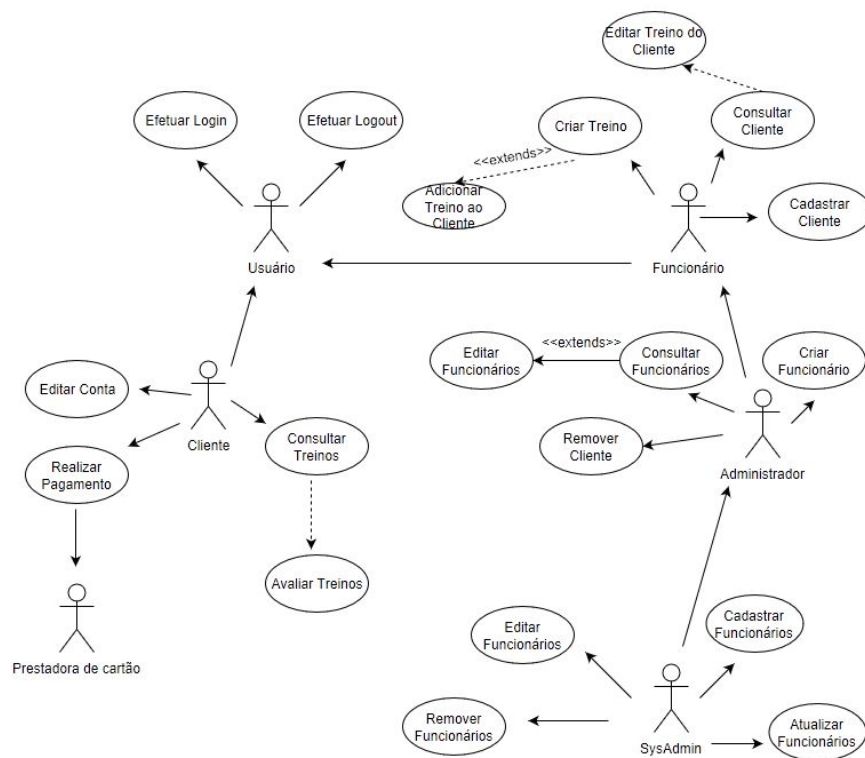
# Roteiro

- Introdução
- Modelagem dos casos de uso
- Casos de uso
- Mapeamento de classes de análise
- Arquitetura
- Diagrama de pacotes

# Introdução

Cada vez mais as pessoas estão procurando por uma melhor qualidade de vida, procurando fazer atividades físicas e ter uma alimentação mais saudável. Visando o fortalecimento deste cenário, propomos o **BeFit**, um sistema de gerenciamento de academias, que facilitará não só a gestão da academia como um todo como também auxiliará os alunos da academia em seu treino.

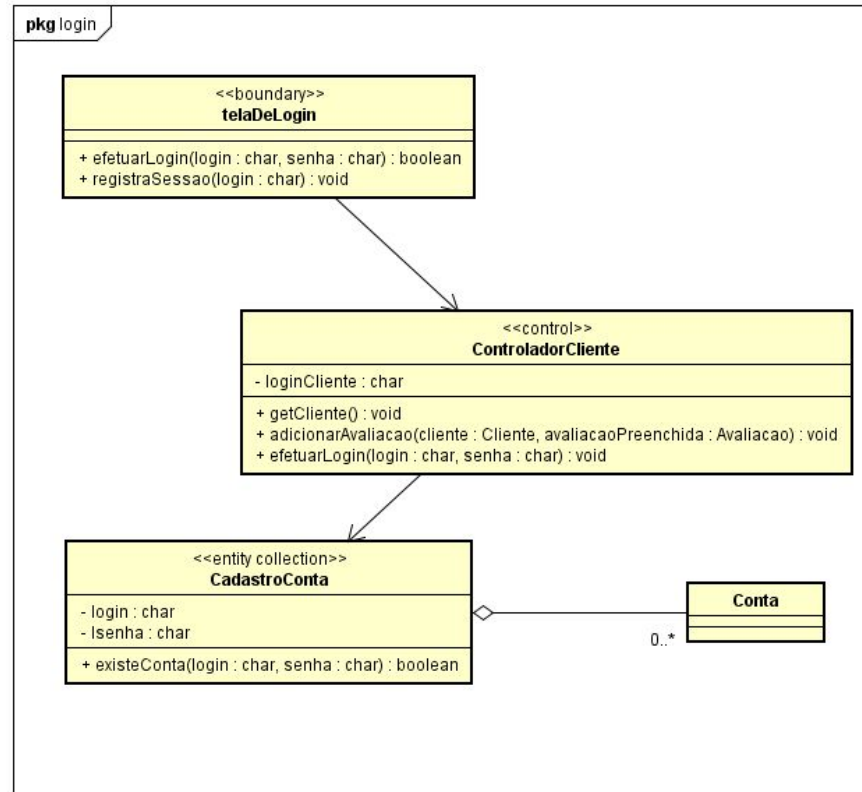
# Modelagem dos casos de uso



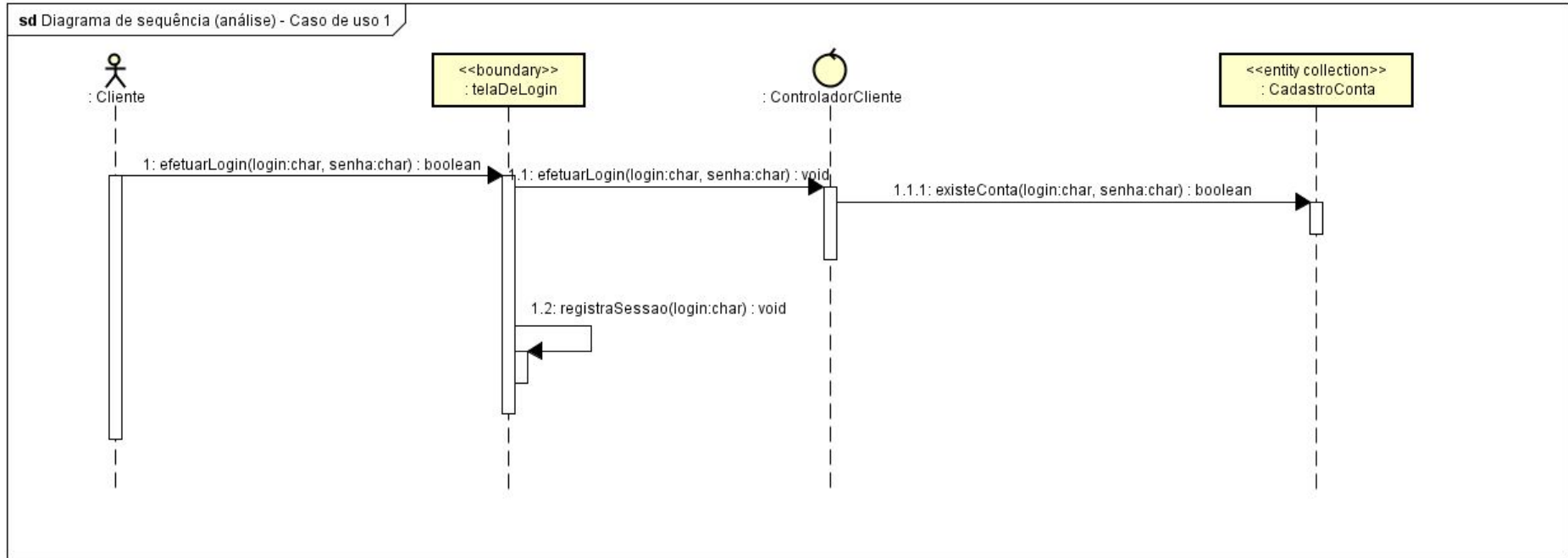
# 1 - Login Cliente

[UC001]	Efetuar Login
Descrição	Caso de uso responsável pelo login do usuário no sistema. Deve haver a validação de login e senha já previamente cadastrados.
Atores	Usuário
Prioridade	Essencial
Pré -condições	Estar cadastrado no sistema
Pós-condições	Ter acesso ao sistema
Fluxo de eventos principais	<ol style="list-style-type: none"><li>1. O usuário preenche os campos de login e senha</li><li>2. O usuário clica no botão de entrar</li><li>3. O sistema valida senha e login</li></ol>
Fluxo de eventos secundários	No passo 1, caso o usuário erre seu login ou senha, não conseguirá ter acesso ao sistema .

# 1 - Login Cliente - Diagrama de classe



# 1 - Login Cliente - Diagrama de Sequência

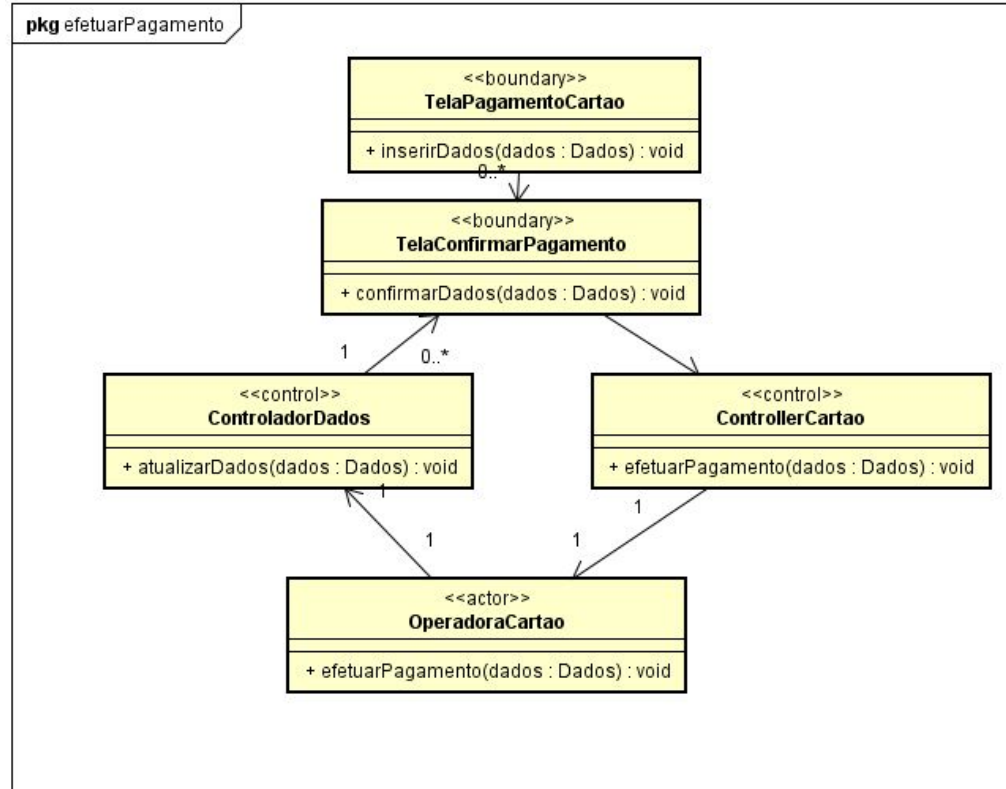


## 2 - Realizar Pagamento

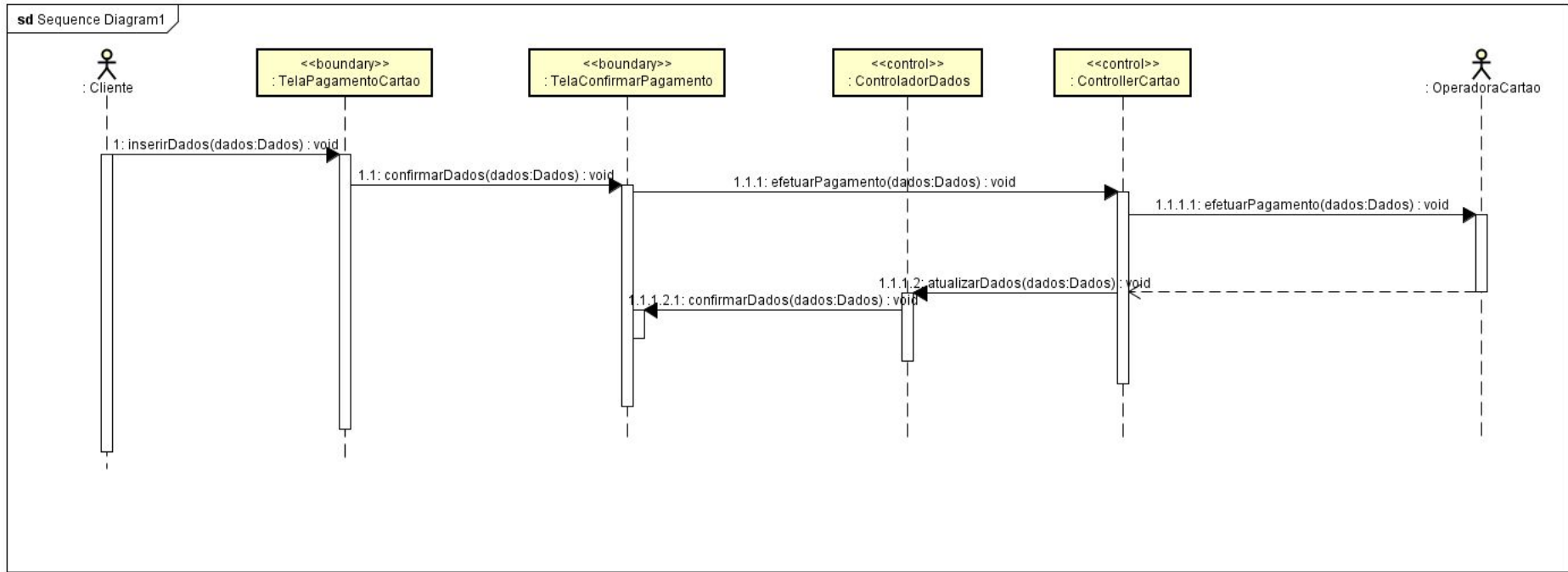
Descrição	Caso de uso responsável pelo pagamento da mensalidade da academia.
Atores	Cliente, Prestadora do Cartão
Prioridade	Essencial
Pré -condições	<ol style="list-style-type: none"><li>1. Cliente matriculado na academia</li><li>2. Cliente faz login no sistema</li><li>3. Dados do cartão</li></ol>
Pós-condições	Recebe um retorno da situação do seu pagamento
Fluxo de eventos principais	<ol style="list-style-type: none"><li>1. O usuário logado clica em realizar pagamento da mensalidade</li><li>2. Preenche os dados da forma de pagamento</li><li>3. Confirma os dados registrados</li><li>4. Salva os dados, caso queira para um próximo pagamento</li><li>5. Recebe um retorno da situação do seu pagamento</li></ol>
Fluxo de eventos secundários	<p>No item 3, caso queira Editar o registro, volta ao item 2, com os campos preenchidos com os dados.</p> <p>No item 5, caso a confirmação de situação venha negatizada, para uso de cartão na compra, aparece mensagem para contatar a operadora do seu cartão, para ter mais detalhes. Ou escolher outra forma de pagamento.</p>



## 2 - Realizar Pagamento - Diagrama de classe



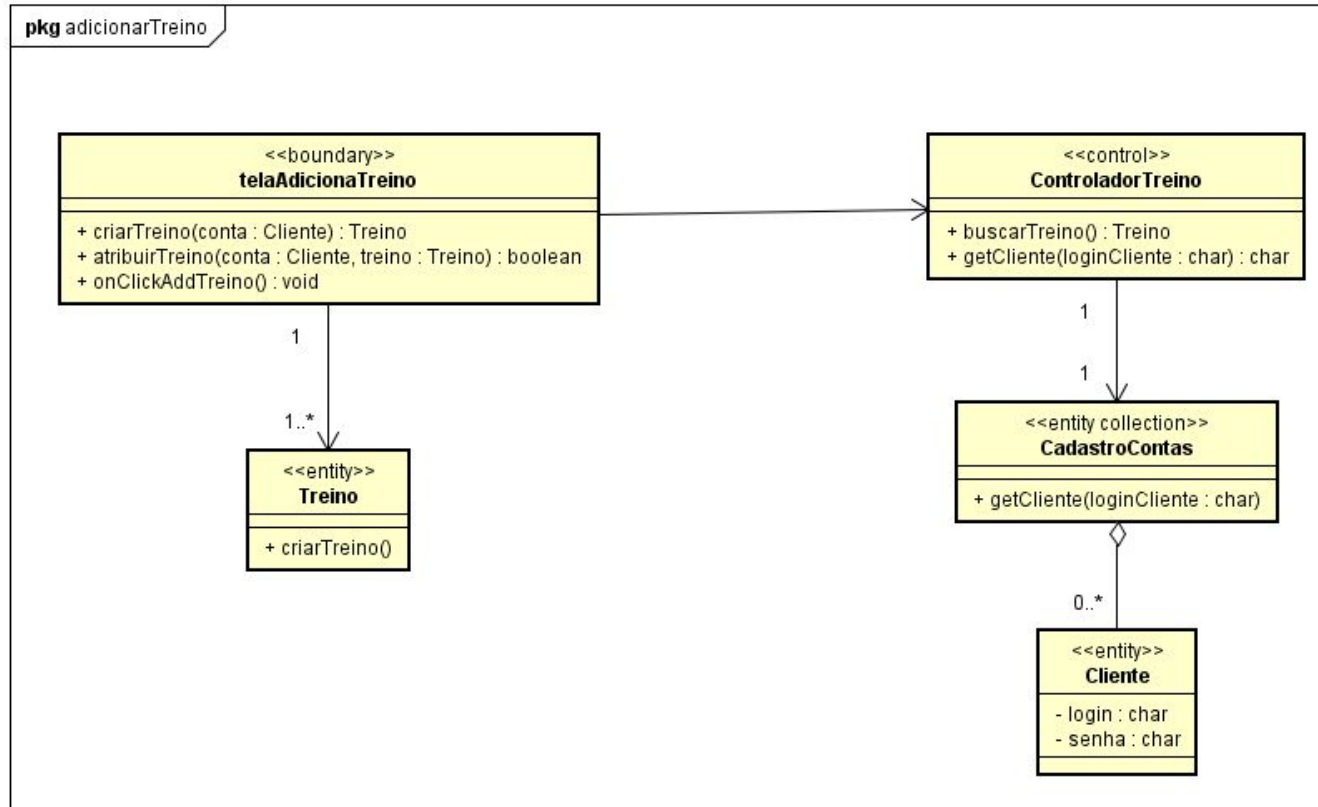
## 2 - Realizar Pagamento - Diagrama de Sequência



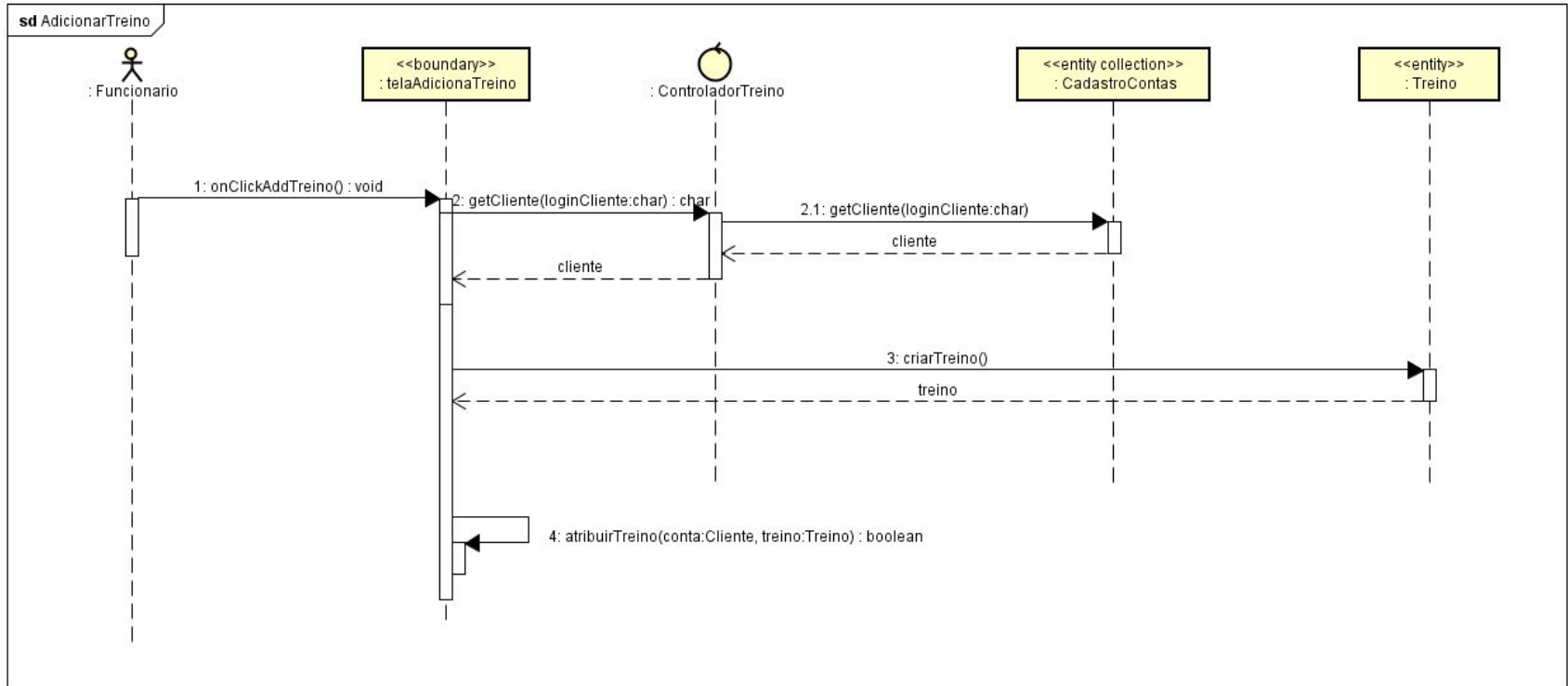
### 3 - Adicionar treino

[UC003]	Adicionar treino
Descrição	Caso de uso responsável pela adição de treinos desejados pelo funcionário para o Cliente.
Atores	Funcionário e Clientes
Prioridade	Importante
Pré -condições	<ol style="list-style-type: none"><li>1. Funcionário estar logado no sistema</li><li>2. Cliente estar cadastrado no sistema</li></ol>
Pós-condições	Cliente ter acesso ao treino
Fluxo de eventos principais	<ol style="list-style-type: none"><li>1. Funcionário faz login no sistema</li><li>2. Funcionário acessa dados do Cliente</li><li>3. Funcionário cria treino para o Cliente</li></ol>
Fluxo de eventos secundários	<ol style="list-style-type: none"><li>1. No passo 2 o funcionário pode acessar dados do cliente e clicar em voltar sem adicionar o treino ao perfil do cliente</li></ol>

### 3 - Adicionar treino - Diagrama de classe



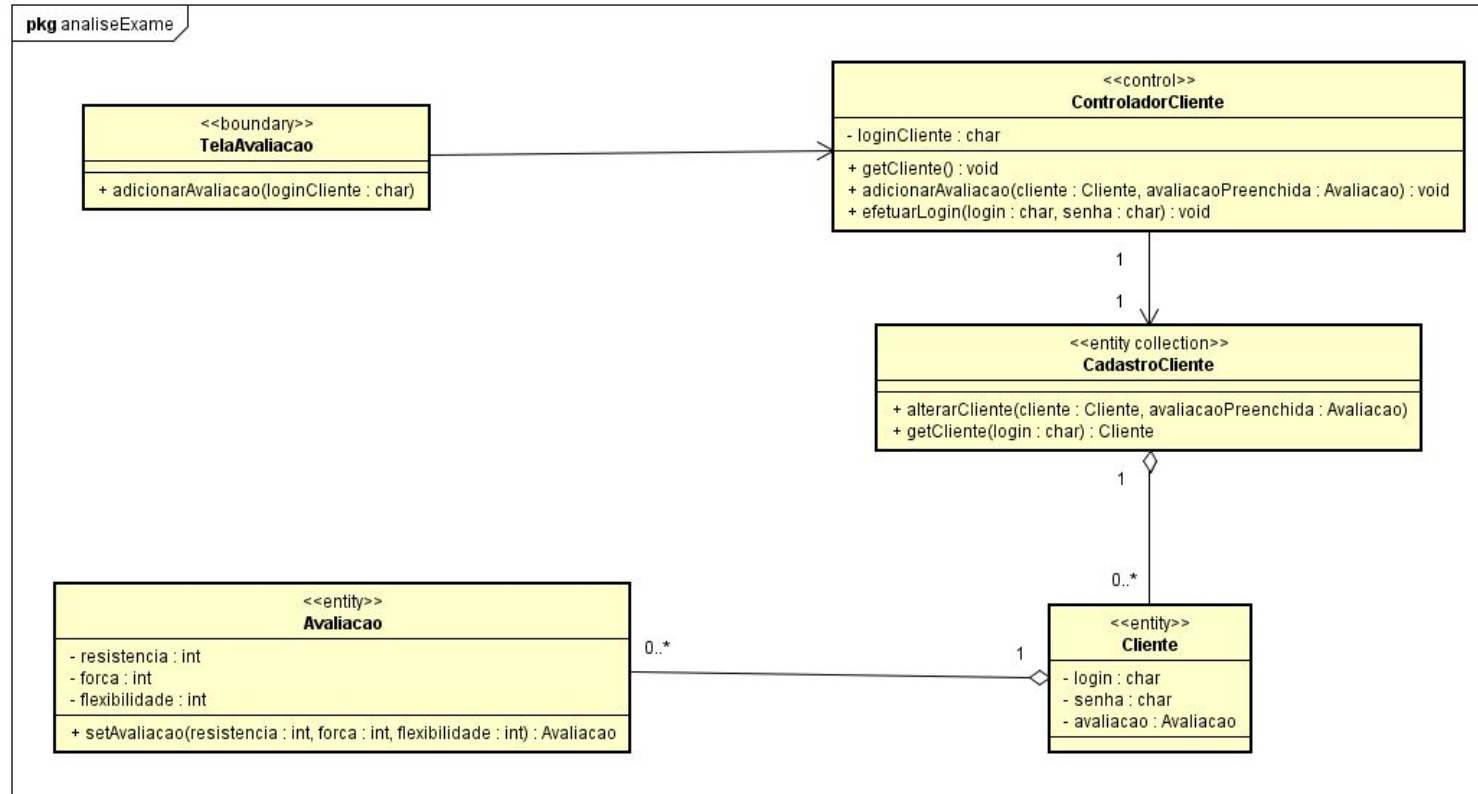
### 3 - Adicionar treino - Diagrama de Sequência



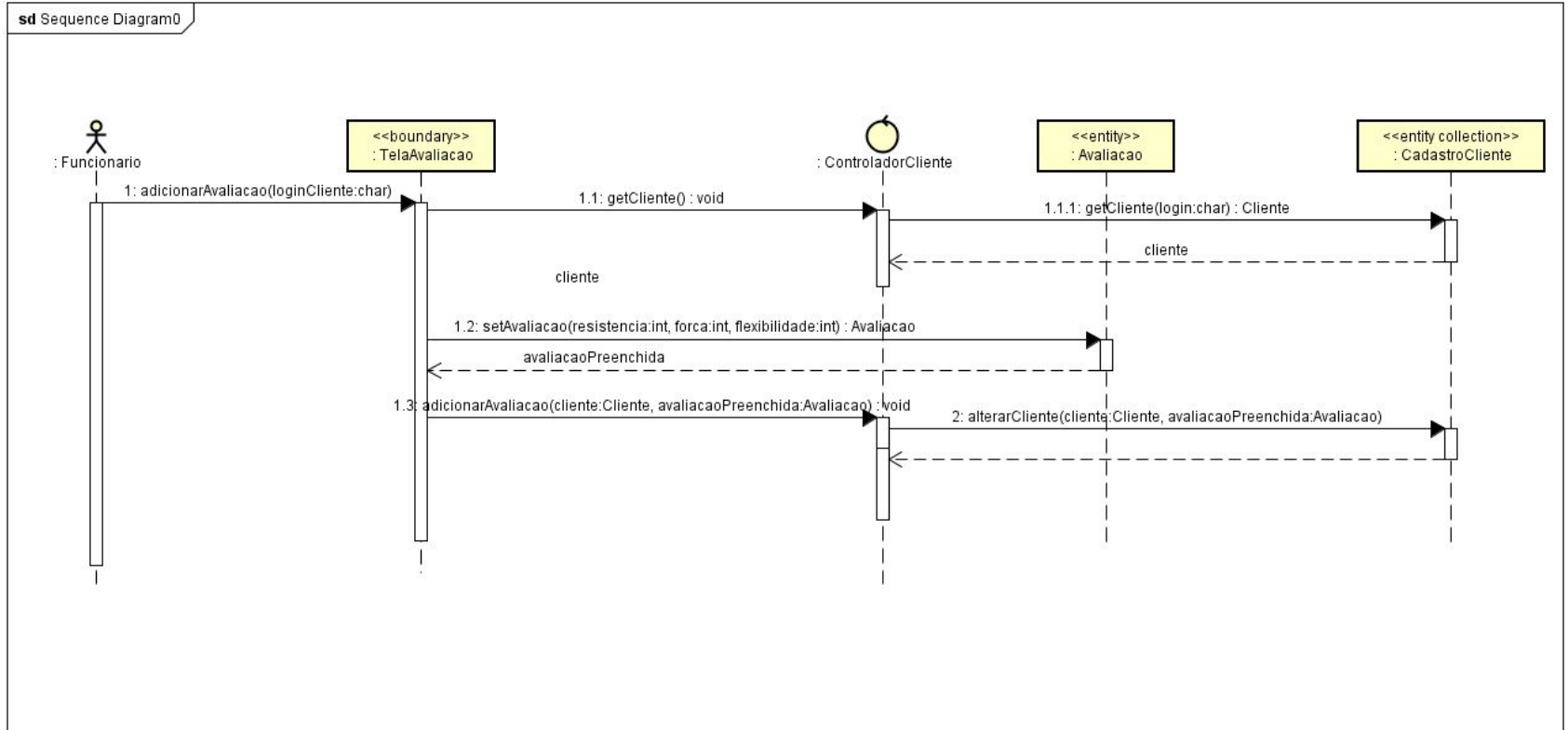
## 4 - Exame físico

[UC004]	Exame físico
Descrição	Cliente realiza exame físico e o funcionário passa as informações para o sistema
Atores	Funcionário e cliente
Prioridade	Baixa
Pré -condições	<ol style="list-style-type: none"><li>1. Cliente cadastrado</li><li>2. Funcionário logado no sistema</li></ol>
Pós-condições	Cliente pode acessar o resultado
Fluxo de eventos principais	<ol style="list-style-type: none"><li>1. Busca informações de um cliente</li><li>2. Funcionário adiciona informações sobre o cliente no sistema</li></ol>
Fluxo de eventos secundários	Caso o funcionário não consiga acessar o sistema ele pode passar as informações depois

## 4 - Exame físico - Diagrama de classe



## 4 - Exame físico - Diagrama de sequência

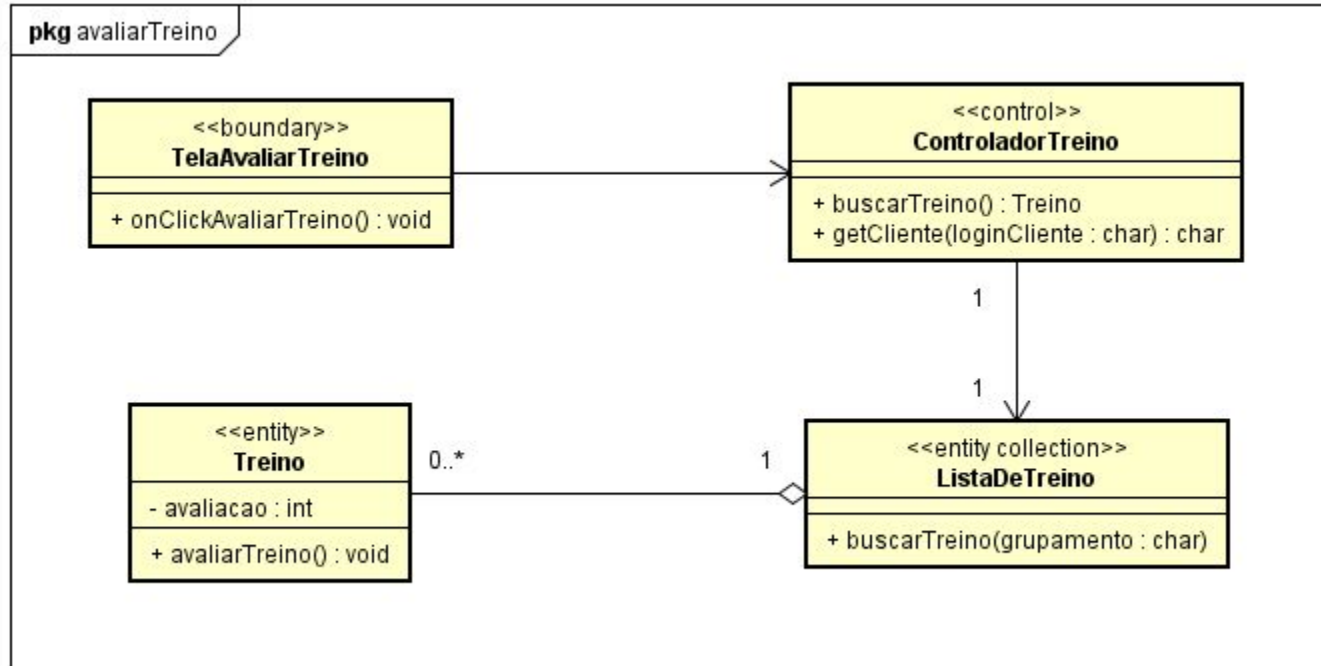




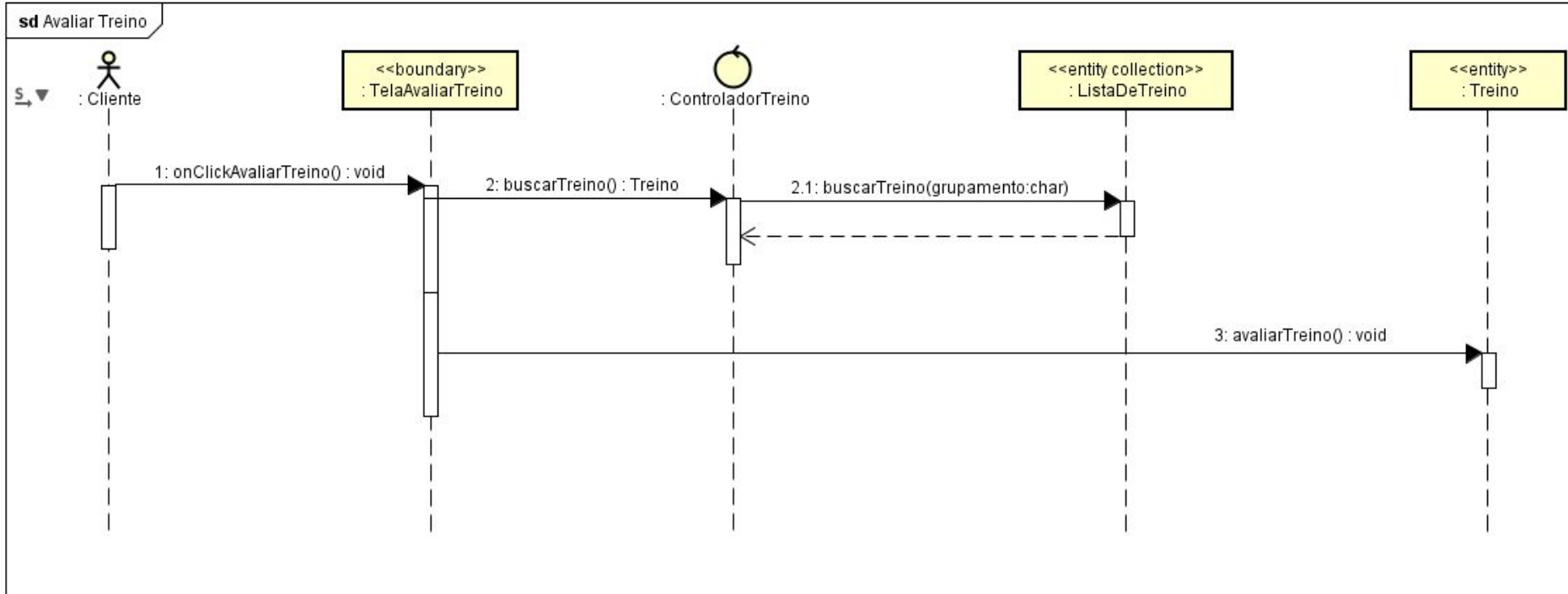
## 5 - Avaliar Treino

[UC005]	Avaliar Treino
Descrição	Caso de uso responsável por avaliar um treino.
Atores	Cliente
Prioridade	Desejável
Pré -condições	<ol style="list-style-type: none"><li>1. Fazer Login</li><li>2. Possuir Treino</li></ol>
Pós-condições	Treino avaliado
Fluxo de eventos principais	<ol style="list-style-type: none"><li>1. Cliente faz login</li><li>2. Cliente na página de treino</li><li>3. Cliente avalia treino(s)</li></ol>
Fluxo de eventos secundários	<ol style="list-style-type: none"><li>1. No passo 3 o cliente pode fazer comentário sobre treino</li></ol>

## 5 - Avaliar Treino - Diagrama de classe



## 5 - Avaliar Treino - Diagrama de sequência



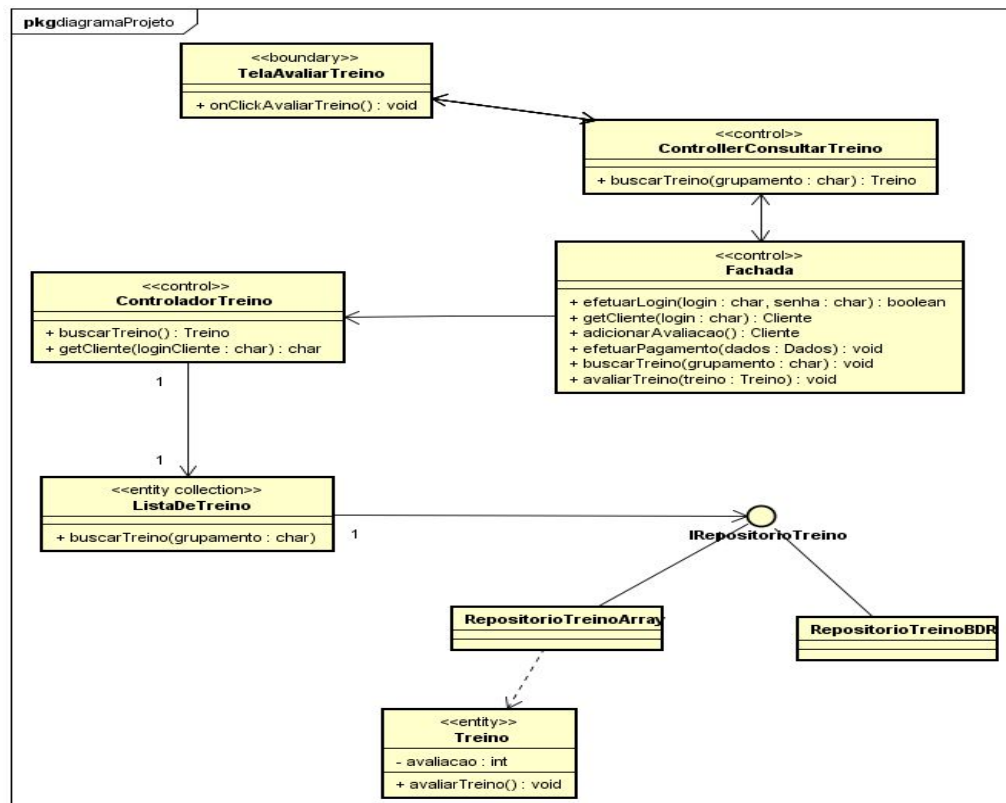
# Mapeamento de classes de análise

Classe de análise	Classe de projeto
Cliente	Cliente
Funcionario	Funcionario
Treino	Treino
Dados	Dados
TelaDeLogin	TelaDeLogin ControllerLogin
TelaAdicionaTreino	TelaAdicionaTreino ControllerTreino
TelaAvaliacao	TelaAvaliacao Controller_avaliacao
TelaConfirmarPagamento	TelaConfirmarPagamento ControllerCartao
TelaAvaliarTreino	TelaAvaliarTreino ControllerConsultarTreino

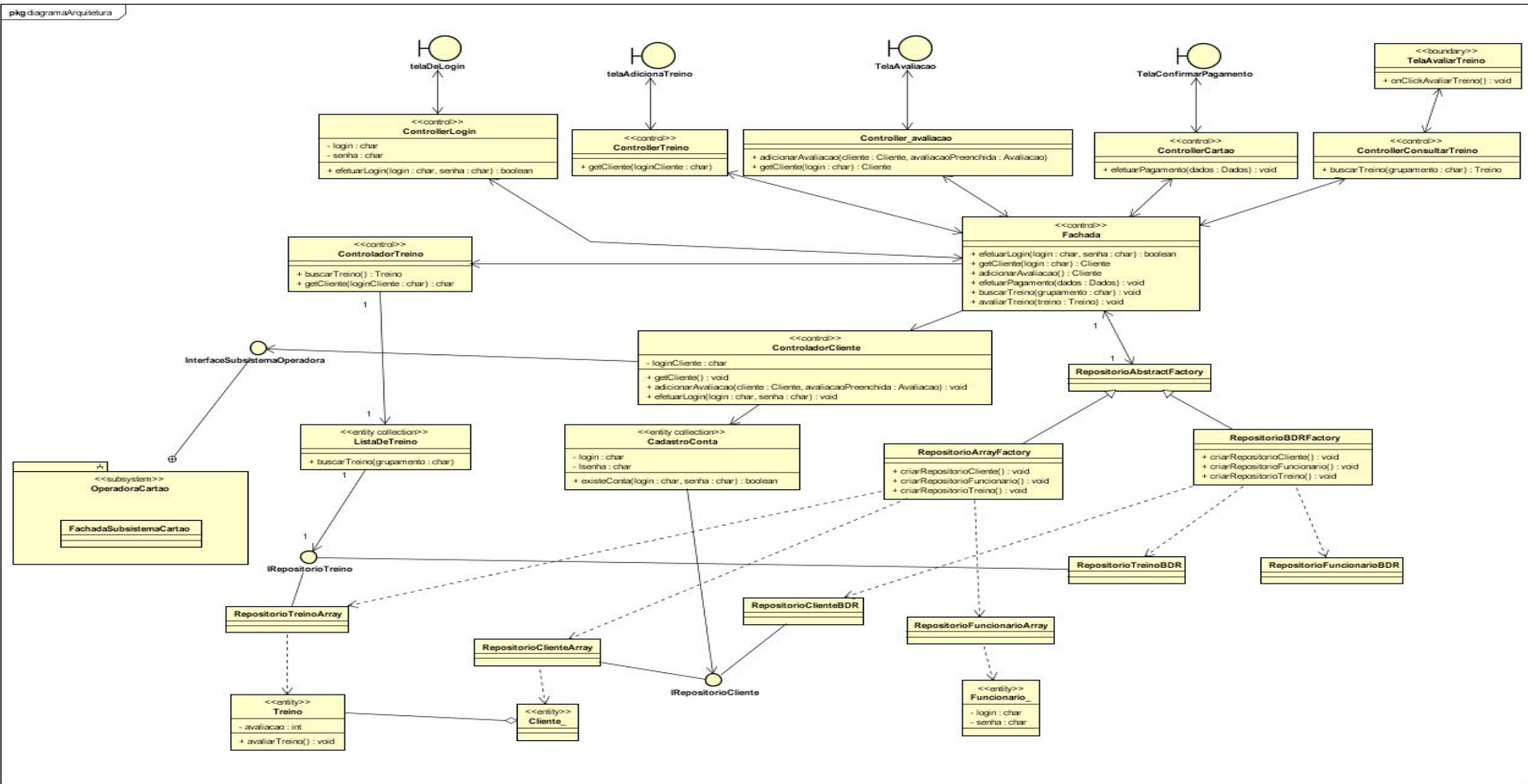
# Mapeamento de classes de análise

<b>ControladorTreino</b>	<b>ControladorTreino</b>
<b>ControladorCliente</b>	<b>ControladorCliente</b>
<b>ControladorDados</b>	<b>ControladorDados</b>
<b>CadastroConta</b>	<b>CadastroConta</b> <b>IRepositorioCliente</b> <b>RepositorioClienteBDR</b> <b>RepositorioClienteArray</b>
<b>ListaDeTreino</b>	<b>ListaDeTreino</b> <b>IRepositorioTreino</b> <b>RepositorioTreinoBDR</b> <b>RepositorioTreinoArray</b>
	<b>RepositorioAbstractFactory</b> <b>RepositorioArrayFactor</b> <b>RepositorioBDRFactory</b>

# Projeto (Adicionar Treino)



# Arquitetura



# Diagrama de pacote

